

(51) International Patent Classification:
G06Q 20/00 (2012.01)(21) International Application Number:
PCT/US2012/031762(22) International Filing Date:
31 March 2012 (31.03.2012)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/471,092 1 April 2011 (01.04.2011) US(71) Applicant (for all designated States except US): **VISA INTERNATIONAL SERVICE ASSOCIATION** [US/US];
900 Metro Center Blvd., Foster City, California 94404-2172 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **POURFALLAH, Stacy Sweeney** [US/US]; Oakland, California 94610 (US).
LI, Shaw [—/US].(74) Agent: **HANCHUK, Walter G.**; Chadbourne & Parke LLP, 30 Rockefeller Plaza, New York, New York 10112 (US).

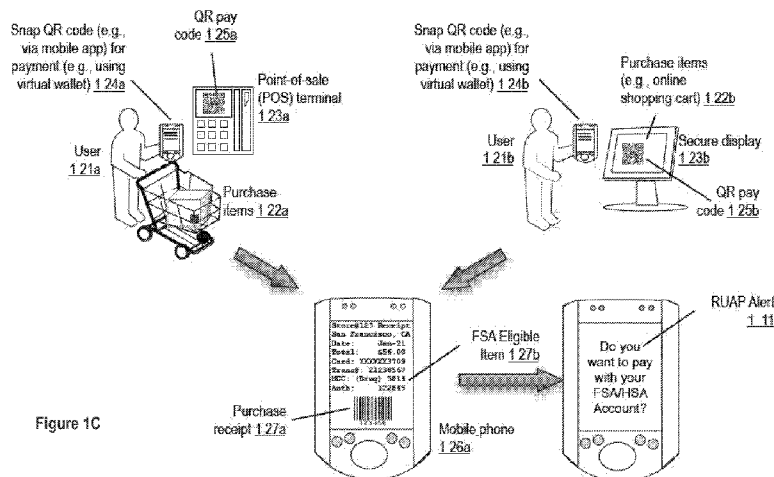
(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LI, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— with international search report (Art. 21(3))

(54) Title: RESTRICTED-USE ACCOUNT PAYMENT ADMINISTRATION APPARATUSES, METHODS AND SYSTEMS



(57) **Abstract:** The RESTRICTED-USE ACCOUNT PAYMENT ADMINISTRATION APPARATUSES, METHODS AND SYSTEMS transform purchase item information inputs or purchase receipt inputs via RUAP components into restricted-use account payment settlement outputs. A method is disclosed, including: receiving a restricted-use account reimbursement request including receipt information related to a purchase transaction from a user; obtaining purchase item information from the receipt information included in the restricted-use account reimbursement request; identifying a restricted-use account of the user; determining a purchase item is eligible for the restricted-account usage; determining a reimbursement amount associated with the purchase item from the purchase item information; generating a reimbursement authorization request message including the purchase item information and the reimbursement amount; transmitting the reimbursement authorization request message to a restricted-account issuer for approval; and transacting the reimbursement amount from the restricted-use account to a user financial account upon the restricted-account issuer approval.

RESTRICTED-USE ACCOUNT PAYMENT ADMINISTRATION APPARATUSES, METHODS AND SYSTEMS

[0001] This patent application disclosure document (hereinafter “description” and/or “descriptions”) describes inventive aspects directed at various novel innovations (hereinafter “innovation,” “innovations,” and/or “innovation(s)”) and contains material that is subject to copyright, mask work, and/or other intellectual property protection. The respective owners of such intellectual property have no objection to the facsimile reproduction of the patent disclosure document by anyone as it appears in published Patent Office file/records, but otherwise reserve all rights.

PRIORITY CLAIM

[0002] Applicant hereby claims priority under Patent Cooperation Treaty and 35 U.S.C. §119 to United States Application serial no. 61/471, 092, entitled "Mobile Claims Administration Apparatuses, Methods and Systems," filed on April 1, 2011.

[0003] The instant application is further related to United States application serial no. _____, entitled "Restricted-Use Account Payment Administration Apparatuses, Methods And Systems," filed on March 31, 2012.

[0004] The aforementioned applications are hereby expressly incorporated by reference.

FIELD

1

2 **[0005]** The present innovations are directed generally to healthcare payment, and
3 more particularly, to RESTRICTED-USE ACCOUNT PAYMENT ADMINISTRATION
4 APPARATUSES, METHODS AND SYSTEMS.

BACKGROUND

5

6 **[0006]** A flexible spending account (FSA) is a financial account which is set up by
7 a user setting aside a portion of their income. For example, the FSA may be used for
8 medical expenses, dependent care, and/or the like. A user needs to collect receipts of
9 payments eligible for FSA reimbursement, and send the receipts to a FSA administer
10 program. Upon verifying eligibility of the expenses, the FSA administer program may
11 return the user payment amount to the user, by an authorized transfer of funds from the
12 user's FSA account to the user.

BRIEF DESCRIPTION OF THE DRAWINGS

13

14 **[0007]** The accompanying appendices and/or drawings illustrate various non-
15 limiting, example, innovative aspects in accordance with the present descriptions:

16 **[0008]** FIGURES 1A-1D provide block diagrams illustrating various scenarios of
17 restricted-use account checkout and/or reimbursement within embodiments of the
18 RUAP;

1 **[0009]** FIGURES 2A-2C provide data flow diagrams illustrating data flows
2 between RUAP and affiliated entities for user PoS checkout with restricted-use account
3 within various embodiments of the RUAP;

4 **[0010]** FIGURES 3A-3E provide logic flow diagrams illustrating user PoS
5 checkout with restricted-use account within various embodiments of the RUAP;

6 **[0011]** FIGURES 4A-4C provide data flow diagrams illustrating data flows
7 between RUAP and affiliated entities for post-purchase restricted-use account
8 reimbursement processing within various embodiments of the RUAP;

9 **[0012]** FIGURES 5A-5C provide logic flow diagrams illustrating data flows
10 between RUAP and affiliated entities for post-purchase restricted-use account
11 reimbursement processing within various embodiments of the RUAP;

12 **[0013]** FIGURES 5D provides a logic flow diagram illustrating restricted-account
13 recommendation within various embodiments of the RUAP;

14 **[0014]** FIGURES 5E-5F provide combined logic flow and user interface(s) flows
15 illustrating healthcare restricted-account (e.g., FSA, HSA, etc.) recommendation within
16 various embodiments of the RUAP;

17 **[0015]** FIGURES 6A-B provide logic flow diagrams illustrating example aspects of
18 processing a bill image comprising a Quick Response code in some embodiments of the
19 RUAP;

20 **[0016]** FIGURES 7A-7C provide data flow diagrams illustrating data flows
21 between RUAP and affiliated entities for user healthcare restricted-use account payment
22 and/or reimbursement within various embodiments of the RUAP;

1 **[0017]** FIGURES 8A-8E provide logic flow diagrams illustrating user healthcare
2 restricted-use account payment and/or reimbursement within various embodiments of
3 the RUAP;

4 **[0018]** FIGURES 9A-9B provide diagrams illustrating restricted-use account
5 enrollment within embodiments of the RUAP;

6 **[0019]** FIGURES 10A-10C provide diagrams illustrating example RUAP aspects in
7 healthcare payment within embodiments of the RUAP;

8 **[0020]** FIGURES 11A-11D provide various exemplary PoS terminal interfaces for
9 user checkout with snap bill/QR code (e.g., see FIGURE 2C) within embodiments of the
10 RUAP;

11 **[0021]** FIGURES 12A-12E provide exemplary UIs of a web-based shopping
12 checkout with RUAP within embodiments of the RUAP;

13 **[0022]** FIGURES 13A-13C provide exemplary mobile UIs illustrating user
14 capturing a barcode/QR code for checkout/reimbursement within embodiments of the
15 RUAP;

16 **[0023]** FIGURES 14A-14G provide exemplary mobile wallet UIs illustrating
17 making a payment within embodiments of the RUAP;

18 **[0024]** FIGURES 15A-15I show user interface diagrams illustrating example
19 features of virtual wallet applications in a snap mode, in some embodiments of the
20 RUAP;

21 **[0025]** FIGURES 16A-16E provide exemplary mobile wallet UIs illustrating wallet
22 account management within embodiments of the RUAP;

1 **[0026]** FIGURES 17A-B show data flow diagrams illustrating an example
2 purchase transaction authorization procedure in some embodiments of the RUAP;

3 **[0027]** FIGURES 18A-B show logic flow diagrams illustrating example aspects of
4 purchase transaction authorization in some embodiments of the RUAP, e.g., a Purchase
5 Transaction Authorization (“PTA”) component;

6 **[0028]** FIGURES 19A-B show data flow diagrams illustrating an example
7 purchase transaction clearance procedure in some embodiments of the RUAP;

8 **[0029]** FIGURES 20A-B show logic flow diagrams illustrating example aspects of
9 purchase transaction clearance in some embodiments of the RUAP, e.g., a Purchase
10 Transaction Clearance (“PTC”) component;

11 **[0030]** FIGURE 21 shows a logic flow diagram illustrating example aspects of
12 transaction data aggregation in some embodiments of the RUAP, e.g., a Transaction
13 Data Aggregation (“TDA”) component;

14 **[0031]** FIGURES 22-23 provide block diagrams illustrating infrastructure within
15 embodiments of the RUAP; and

16 **[0032]** FIGURE 24 shows a block diagram illustrating embodiments of a RUAP
17 controller;

18 **[0033]** The leading number of each reference number within the drawings
19 indicates the figure in which that reference number is introduced and/or detailed. As
20 such, a detailed discussion of reference number 101 would be found and/or introduced
21 in FIGURE 1. Reference number 201 is introduced in FIGURE 2, etc.

DETAILED DESCRIPTION

1
2 **[0034]** The RESTRICTED-USE ACCOUNT PAYMENT ADMINISTRATION
3 APPARATUSES, METHODS AND SYSTEMS (hereinafter “RUAP”) provides a
4 healthcare payment platform which facilitates payment and/or reimbursement from a
5 restricted-use account.

6 **[0035]** In one implementation, RUAP may facilitate a user to engage a restricted-
7 use account for the cost of eligible items. A restricted-use account may be a financial
8 account having funds that can only be used for payment of approved products (e.g.,
9 prescription drugs, vaccine, food, etc.) and/or services (e.g., healthcare treatment,
10 physical examination, etc.). Examples of a restricted use account may comprise Flexible
11 Savings Accounts (FSA), one or more Health Savings Accounts (HSA), Line of Credit
12 (LOC), one or more health reimbursement accounts (HRA), one or more government
13 insurance programs (i.e., Medicare or Medicaid), various private insurance - rules,
14 various other restricted use favored payment accounts such as employment benefit
15 plans or employee pharmacy benefit plans, and income deduction rules, and/or the like.
16 In other examples, the restricted-use account may comprise a food voucher, a food
17 stamp, and/or the like. Within implementations, the approval process of payment with
18 a restricted use account may be administered by a third party, such as, but not limited to
19 FSA/HSA administrator, government unemployment program administrator, and/or
20 the like.

21 **[0036]** For example, the RUAP processing platform may facilitate payment with
22 healthcare restricted-use accounts (e.g., FSA, HSA, etc.) when a user/patient visits a

1 healthcare provider. In one embodiment, a user may operate a payment device (e.g., a
2 mobile wallet component instantiated on a smart phone, a healthcare prepaid card, a
3 web-based application, etc.) for checkout at a merchant, wherein the mobile computing
4 device is web enabled, and may receive a communication from a point of service
5 terminal (POS). The communication may include a balance due bill of a healthcare
6 provider for healthcare to a user. The web enabled mobile computing device may
7 execute an application that derives an optimized payment of the balance due bill that
8 substantially maximizes incentives and minimize penalties in paying the healthcare
9 provider for the healthcare provided to the user. The optimized payment is transmitted
10 from the web enabled mobile computing device to the POS for further authorization
11 processing of one or more currency amounts to be paid from respective accounts to
12 satisfy the balance due bill.

13 **[0037]** Integration of an electronic wallet, a desktop application, a plug-in to
14 existing applications, a stand alone mobile application, a web based application, a smart
15 prepaid card, and/or the like in payment with restricted-use account reduces the
16 number of network transactions and messages that fulfill restricted-use account
17 payment approval and procurement of eligible purchase items (e.g., a user and/or a
18 merchant does not need to file additional application requests for restricted-use account
19 payment/reimbursement, which may require voluminous paper work or manual
20 information fill-in). In this way, with the reduction of network communications, the
21 number of transactions that may be processed per day is increased, i.e., processing
22 efficiency is improved.

RUAP

1
2 **[0038]** FIGURE 1A shows a block diagram illustrating example aspects of mobile
3 claims administration in some embodiments of the RUAP. In some implementations, a
4 user, e.g., 101, may wish to file a claim for a refund of funds paid by the user, file an
5 insurance claim, and/or the like claim for compensation, see, e.g., 102. In one
6 implementation, the user may determine his/her recent purchase comprises a restricted-
7 use account eligible item, and may want to engage the restricted-use account to pay for
8 the eligible items. For example, the user may desire to use his FSA/HSA accounts to pay
9 for purchases of prescription drugs, physician office visits, medical treatments, and/or
10 the like. In another example, the user may desire to use his Medicare, Medicaid account
11 to pay for received medical products/ services received at a qualified provider, e.g., a
12 public hospital. In another example, the user may desire to use a food voucher to pay
13 for grocery goods, and/or the like.

14 **[0039]** In some scenarios, a user may use a non-restricted-use account, e.g., the
15 user's credit card, debit card, personal check, and/or the like to pay for a purchase, and
16 may want to reimburse the cost of eligible items from his restricted-use accounts. In
17 one implementation, the process of filing a claim for such compensation may involve a
18 large amount of paperwork. In such implementations, the user may be discouraged
19 from filing the claim due to the large amount of paperwork. In some implementations,
20 the RUAP may reduce and/or eliminate the need for the user to undertake paperwork to
21 file the claim for compensation.

22 **[0040]** In some implementations, the user may have a user device, e.g., 103. The
23 user may utilize the user device to obtain a snapshot of a receipt for funds previously

1 paid by the user (e.g., a "post-flight" reimbursement claim, etc.), a license plate of a car,
2 and/or like evidence of validity of a claim 105a for compensation. In some
3 implementations, the user may provide the snapshot of the payment receipt for the
4 RUAP, and the RUAP may, see, e.g., 104, generate a claim for compensation on behalf of
5 the user, and coordinate with the claims processor to obtain a claims compensation for
6 the user. The RUAP may also automatically credit funds obtained via claim
7 compensation to an account of the user. In some implementations, the RUAP may
8 require no user intervention beyond providing the snapshot to provide mobile claims
9 administration features.

10 **[0041]** In another implementation, the user may operate the user device 103 at a
11 merchant, e.g., a Point of Sale (PoS) terminal at a pharmacy, a healthcare provider, etc.,
12 to directly pay for eligible items (e.g., an "in-flight" payment during checkout). For
13 example, the user may operate an electronic wallet and select his FSA/HSA account to
14 pay for prescription drugs at a pharmacy, and/or the like.

15 **[0042]** FIGURE 1B provides an exemplary block diagram illustrating RUAP
16 payment at a participating merchant within embodiments of the RUAP. In one
17 implementation, a user 102 may have in his/her shopping cart a list of mixed items
18 including restricted-use account eligible items 121a and non-eligible items 121b. For
19 example, a user 102 may shop at a supermarket, a pharmacy store, etc., to purchase
20 pharmaceutical drugs (e.g., NyQuil cold medicine, Antibiotics, etc.) which are eligible
21 for FSA/HSA usage, and various items (e.g., body wash, shampoo, etc.) that are not
22 eligible for any restricted-use account.

1 **[0043]** In one implementation, a user 102 may engage a mobile wallet 103 to pay
2 for the purchase at a PoS terminal 109 of the merchant. In one implementation, the
3 mobile wallet 103 may provide a prompt alert 111 to the user via a mobile user interface,
4 showing that part or all of the user's purchases are eligible for one or more restricted-
5 use accounts that have been enrolled in the user's mobile wallet. For example, as shown
6 in FIGURE 1B, when the user has purchased pharmaceutical products such as "NyQuil"
7 and "Penicillin," the mobile wallet may notify the users that such drugs are eligible for
8 FSA usage. In one implementation, if the user selects "Yes" to proceed with payment
9 with FSA, the user may need to split the purchase to pay for the eligible items 121a in the
10 shopping cart with his FSA account and submit an amount of the drugs. For example,
11 the user may view a list of the FSA eligible items 113a, and a list of healthcare restricted-
12 use accounts such as FSA, HSA, etc.. If the user selects to pay with FSA, the user may
13 view the remaining balance of the FSA account 112. Upon selecting to pay 113, the user
14 may submit the transaction to pay with FSA account for the healthcare products 113a.

15 **[0044]** In one implementation, the user may be directed to another list of
16 remaining items which are non-eligible for any restricted-use items 121b, e.g., see 113b.
17 The user may then need to select another account (e.g., a regular bank account, etc.),
18 such as a Visa credit card 116 for non-eligible items 121b in a separate transaction.

19 **[0045]** In one implementation, the RUAP alert may be originated at a PoS
20 terminal 109, wherein the merchant may maintain a blacklist/whitelist of eligible
21 product codes for various types of restricted-use accounts, and may send notifications to
22 the wallet via Near Field Communication (NFC) protocol 115.

1 **[0046]** In another implementation, a user's electronic wallet may identify eligible
2 items for a restricted-use account itself, e.g., the PoS terminal 109 may generate a bill
3 and transmit bill information to the user's wallet via NFC 115. In further
4 implementations, the PoS terminal 109 may generate a bill comprise a QR code, and the
5 user may snap a photo of the generated QR code on the bill, which may facilitate the
6 intelligent wallet to capture information of user purchased items.

7 **[0047]** In an alternative implementation, the user may operate a restricted-use
8 prepaid card, and the PoS terminal may inform the user eligibility of the user's purchase
9 to apply his/her restricted-use account.

10 **[0048]** In further implementations, the user's mobile wallet may intelligently
11 recommend an account in the wallet for the instant payment. For example, the mobile
12 wallet may detect the user's location at a healthcare provider 108 via its GPS component,
13 and thus may recommend healthcare benefit accounts for user payment by highlighting
14 the accounts "FSA" 111 and "HSA". When the user selects the FSA account 111, the
15 wallet may display an available balance 112 of the FSA account. The user may then tap
16 on a "pay" button 113 of the mobile wallet to initiate a user payment transaction.

17 **[0049]** FIGURE 1C provides an example showing user checkout with QR code
18 capturing at a merchant PoS terminal within embodiments of the RUAP. In some
19 implementations, a user, e.g., 121a-b, may wish to purchase products at a merchant
20 store, e.g., 123a, or at a merchant website, e.g., 123b. For example, at a merchant store,
21 the user may scan barcodes for a number of products, e.g., 122a, at a PoS terminal in the
22 store, e.g., 123a, and then indicate that the user wishes to checkout the scanned items.
23 In some implementations, the POS terminal may generate a Quick Response ("QR")

1 code, e.g., 125a, including information on the scanned product items, as well as
2 merchant information for processing the purchase transaction via a payment network.
3 The user may capture an image of the QR code generated by the POS terminal using a
4 user device, such as a smartphone. For example, the user device may have executing on
5 it an app for snapping the merchant-product QR code. The user device may utilize the
6 information extracted from the QR code, along with information on a virtual wallet tied
7 to the user device to initiate a purchase transaction. For example, the user device may
8 utilize the product and merchant information extracted from the QR code, and financial
9 payment information from the virtual wallet, to create a purchase transaction request,
10 and submit the request to a payment network (e.g., credit card processing network).

11 **[0050]** In some implementations, the user device may utilize methods alternative
12 to capture of a QR code to obtain information from the POS terminal. For example, the
13 POS terminal may communicate the information required for submitting a purchase
14 transaction request to a payment network to user device via Bluetooth™, Wi-Fi, SMS,
15 text message, electronic mail, and/or other communication methods.

16 **[0051]** In some implementations, a user 121b may wish to checkout items stored
17 in a virtual shopping cart on an online shopping website, e.g., 122b. For example, the
18 user may be viewing the website using a secure display (e.g., that is part of a trusted
19 computing device of the user). Upon indicating that the user wishes to checkout the
20 items in the virtual shopping cart, the website may provide a QR code including
21 information on the products in the virtual shopping cart and merchant information. For
22 example, in the scenario where the user utilizes a secure display, the QR code may be
23 displayed at a random position within the secure display for security purposes. The user

1 may capture a snapshot of the displayed QR code, and utilize payment information from
2 the virtual wallet associated with the user device to create a purchase transaction
3 request for processing by the payment network. Upon completion of the purchase
4 transaction, the payment network may provide a purchase receipt, e.g., 127a directly to
5 the user device 126a, the POS terminal in the store and/or the secure display (for the
6 secure online shopping scenario) as confirmation of completion of transaction
7 processing. Thus, in some implementations, the merchant may be shielded from
8 obtaining personal and/or private information of the user while processing the purchase
9 transaction, while ensuring integrity of the user's virtual wallet using a secure display for
10 presenting the merchant-product QR code.

11 **[0052]** In various implementations, such payment processing may be utilized for
12 a wide variety of transactions. For example, a user dining at a restaurant may obtain a
13 bill including a QR pay code including detail on the dining charges included in the bill,
14 and a merchant ID for the restaurant. The user may take a snapshot of the restaurant
15 bill using the user's smartphone, and utilize the user's virtual wallet to pay for the
16 restaurant bill, without revealing any financial or personal information about the user to
17 the restaurant.

18 **[0053]** In an alternative implementation, when the receipt 127a comprise a FSA
19 eligible item 127b (e.g., prescription drugs, etc.), the user device 126a may receive a
20 RUAP alert 111 to notify user eligibility of the purchase item for FSA usage. For example,
21 in one implementation, the RUAP alert 111 may be received prior to user engage in the
22 final payment so that the user may elect to pay with FSA account. In another example,
23 the RUAP alert may be received after the user has obtained a post-payment purchase

1 receipt and may submit the receipt for reimbursement into the eligible FSA account, as
2 shown in FIGURE 1D.

3 **[0054]** FIGURE 1D shows a block diagram illustrating an example scenario
4 where a user may request reimbursement with his/her restricted-use accounts in some
5 embodiments of the RUAP. In some implementations, a user, e.g., 102, may be required
6 to file a claim to obtain a refund of credit made towards a card-based purchase
7 transaction (e.g., credit card, debit card, prepaid card transaction, hereinafter “charge
8 card”). In one implementation, the user may have a signature debit card for purchase
9 transactions, and as well as a FSA, HSA, HRA, LOC, and/or like accounts. In some
10 scenarios, the user may desire to purchase a number of goods at a store including over-
11 the-counter medication, and may be other items as well, e.g., 131. The user may find it
12 inconvenient to utilize the cards required to charge the medication purchase to the
13 user’s FSA as well as the charge card for expenses ineligible to be paid for using the
14 user’s FSA. For example, continuing on with the example in FIGURE 1B, the user may
15 elect not to pay with FSA at checkout as the user may prefer to checkout all FSA-eligible
16 and non-eligible items in one transaction for convenience, instead of splitting the
17 purchase into different stages.

18 **[0055]** Thus, in some implementations, the user may utilize the charge card to
19 pay for all the expenses, including expenses that may be eligible for payment using the
20 user’s FSA. In such implementations, the user may be paying more that may be needed
21 if the user utilizes the user’s FSA, or the user may be reducing the user’s net income by
22 not utilizing the savings, income, rewards, tax and/or like advantages offered by the FSA.
23 In some implementations, the user may require to substantiate a payment made using a

1 FSA, to prove that the user only utilized the FSA for purposes for which the FSA was
2 issued to the user.

3 **[0056]** In some implementations, the RUAP may provide the user with features to
4 prove that the user only used the user's FSA for legitimate purposes. In some
5 implementations, the RUAP may provide the user with features to transfer charges from
6 a charge card to the user's FSA when the charges are eligible to be paid for using the
7 user's FSA. For example, the user may obtain a purchase receipt for the purchase of the
8 medication items, e.g., 132. The user may obtain a snapshot of the purchase receipt, e.g.,
9 133, e.g., via a mobile device of the user, e.g., 103. The user may provide a snapshot of
10 the purchase receipt 107 to the RUAP for claim processing. The RUAP may obtain the
11 snapshot, extract a list of the items purchased by analyzing the snapshot of the RUAP.
12 The RUAP may determine which of the charges on the purchase receipt are eligible to be
13 paid by the user's FSA. In implementations where the user seeks to substantiate charges
14 made to the FSA, the RUAP may determine whether each charge is eligible to be paid for
15 using the user's FSA, and may provide such substantiation for the user and/or other
16 entities and/or components of the RUAP. In implementations, where the user seeks to
17 transfer charges that are eligible to be paid for using the user's FSA from a charge card
18 to the user's FSA, the RUAP may identify the charges that may be transferred, and
19 credit/debit the account(s) necessary to achieve the transfer. For example, the RUAP
20 may debit the user's FSA account with an amount based on the transaction cost for the
21 items eligible to be paid for using the user's FSA, and credit the user's charge card
22 account appropriately, e.g., at 135. In some implementations, because the user's FSA
23 can provide savings to the user when the user utilizes the FSA to make eligible
24 purchases, the RUAP may provide savings to the user by ensuring that any eligible

1 purchases are always transferred to the appropriate account(s). In some
2 implementations, the RUAP may provide notification(s), e.g., 134, to the user via the
3 user's mobile device indicating the crediting of the user's account.

4 **[0057]** FIGURES 2A-2B provides a data block diagram illustrating data flow
5 between RUAP server, user, merchant and affiliated entities to substantiate an in-flight
6 PoS checkout payment at a merchant within embodiments of the RUAP. Within various
7 embodiments, as shown in FIGURES 2A-2B, one or more user(s) 202, RUAP server 220,
8 RUAP database(s) 219, merchant(s) (PoS terminal(s)) 210, account issuer network 230 ,
9 and/or the like are shown to interact via various communication networks 213 to
10 facilitate the substantiation of a user purchase payment transaction.

11 **[0058]** Within various embodiments, the user (e.g., an individual consumer, etc.)
12 202 may include a wide variety of different communications devices and technologies
13 within embodiments of RUAP operation. For example, in one embodiment, the patient
14 102 may operate devices such as, but not limited to, terminal computers, work stations,
15 servers, cellular telephony handsets, smart phones (e.g., an Apple iPhone, a Google
16 Android, a BlackBerry, etc.), PDAs, and/or the like.

17 **[0059]** In one embodiment, the RUAP server 220 may be equipped at a terminal
18 computer, a mobile device, and/or or the like of the user 202. In another embodiment,
19 the RUAP server 220 may be a remote server which is accessed by the user 202 via a
20 communication network 113, such as, but not limited to local area network (LAN), in-
21 house intranet, the Internet, and/or the like. In a further implementation, the RUAP
22 merchant 210 may communicate with the user 202 via a POS terminal, and/or be
23 integrated with a user 202 at a computer terminal. For example, the user 202 may

1 operate a mobile device as a self-checkout device, e.g., see barcode scanning checkout
2 examples in FIGURE 13A-13C.

3 **[0060]** Within embodiments, a user 202 may shop with a merchant 210, which
4 may be a physical store, an online shopping site, and/or the like. For example, the user
5 202 may walk in a physical merchant store and bring a shopping cart of purchase item
6 to a PoS terminal, which may scan in the purchase item information 203. In another
7 example, the user 202 may engage in online shopping by adding purchase items into a
8 virtual shopping cart and transmit the purchase item information 203 to the shopping
9 site server.

10 **[0061]** For example, in one implementation, a user device may generate a
11 Hypertext Transfer Protocol Secure (HTTPS) POST message to send purchase item
12 information 203 in the form of data formatted according to the XML. Below is an
13 example HTTP(S) POST message including an XML-formatted message of the purchase
14 item information 203 for the merchant 210:

```
15 POST /PurchaseItem\\.php HTTP/1.1
16 Host: 216.15.51.74
17 Content-Type: Application/XML
18 Content-Length: 718
19 <?XML version = "1.0" encoding = "UTF-8"?>
20 <PurchaseItem>
21     <Time> 17:40:40 </Time>
22     <Date> 09-09-2015 </Date>
23     <Item1>
24         <ItemCode> FOOD-9875 </ItemCode>
25         <Category> FOOD </Category>
26         <Sub-Category> Breakfast </Sub-Category>
27         </ItemName> Cereal </ItemName>
28         <Description> whole grain original 10 oz </Description>
29         <Quantity> 1 </Quantity>
30         <UnitPrice> 4.99 </UnitPrice>
31         <TotalAmt> 4.99 </TotalAmt>
32     ...
33 </Item1>
34 <Item2>
35     <ItemCode> DRUG-23401 </ItemCode>
36     <Category> DRUG </Category>
37     <Sub-Category> Non-Prescription </Sub-Category>
38     </ItemName> NyQuil Cold Medicine </ItemName>
```

```

1         <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
2         <Quantity> 1 </Quantity>
3         <UnitPrice> 12.99 </UnitPrice>
4         <TotalAmt> 12.99 </TotalAmt>
5         ...
6     </Item2>
7     <item_3>
8         <ItemCode> SU-Shampoo-001 </ItemCode>
9         <Category> WASH </Category>
10        <Sub-Category> hair product </Sub-Category>
11        </ItemName> SUAVE shampoo </ItemName>
12        <Description> SUAVE shampoo heat treatment 120 mL </Description>
13        <Quantity> 1 </Quantity>
14        <UnitPrice> 8.99 </UnitPrice>
15        <TotalAmt> 8.99 </TotalAmt>
16        ...
17    </item_3>
18    ...
19 </PurchaseItem>
20

```

[0062] Within embodiments, the merchant 210 may generate a bill 208 upon the
 obtained purchase item information. For example, the bill may take a similar data
 structure as the obtained purchase item information 203. In one implementation, the
 merchant may include an intelligent PoS terminal, which may identify whether a
 purchase item qualifies for a restricted-use account. For example, when the merchant is
 equipped with a smart PoS terminal, the terminal may query for a list of eligible
 merchant category code (MCC) associated with a restricted-use account, to determine
 whether the restricted-use account may be applied to purchase the item. In one
 implementation, the PoS terminal may generate a restricted-use account white list
 matching status 206, which may comprise information as to a recommended restricted-
 use account that may be applied to the item. An exemplary XML-formatted message of
 the restricted-use account status 206 for the merchant 210 may take a form similar to
 the following:

```

34 POST /RestrictStatus.php HTTP/1.1
35 Host: 216.15.51.74
36 Content-Type: Application/XML
37 Content-Length: 718
38 <?XML version = "1.0" encoding = "UTF-8"?>
39 <ItemStatus>
40     <Time> 17:40:40 </Time>
41     <Date> 09-09-2015 </Date>

```

```

1      ...
2      <user>
3          <user_id> JS-001 </user_id>
4          <user_name> John Smith </user_name>
5          <wallet_id> js_wallet </wallet_id>
6          <password> ***** </password>
7      ...
8      </user>
9      <Item1>
10         <ItemCode> FOOD-9875 </ItemCode>
11         <Category> FOOD </Category>
12         <Sub-Category> Breakfast </Sub-Category>
13         <RecommendedAct>
14             <Act1> Food Stamp </Act1>
15         ...
16         </RecommendAct>
17     </Item1>
18     <Item2>
19         <ItemCode> DRUG-23401 </ItemCode>
20         <Category> DRUG </Category>
21         <Sub-Category> Non-Prescription </Sub-Category>
22         <RecommendedAct>
23             <Act1> FSA </Act1>
24             <Act2> HSA </Act2>
25         ...
26         </RecommendAct>
27     ...
28     </Item2>
29     <Item3>
30         <ItemCode> MS-Shampoo-01 </ItemCode>
31         <Category> Body Wash</Category>
32         <Sub-Category> shampoo </Sub-Category>
33         <RecommendedAct>
34             <Act1> Regular </Act1>
35         ...
36         </RecommendAct>
37     ...
38     </Item3>
39 ...
40 </ItemStatus>
41
42

```

43 **[0063]** Within implementations, the merchant 210 may provide an account
 44 recommendation message 212 to the user 202, e.g., a message sent to the user's mobile
 45 device via NFC handshake, a message displayed on the PoS terminal, and/or the like.
 46 For example, in one implementation, as shown in the above example of restricted-use
 47 account matching status message 206, when the PoS terminal has determined one or
 48 more of the user's purchase items includes healthcare products that are eligible for
 49 FSA/HSA expenses, the PoS terminal may generate a RUAP alert asking the user
 50 whether the user would like to purchase eligible items using FSA/HSA account, e.g., see

1 111 in FIGURE 1B. An exemplary XML-formatted message of account recommendation
 2 message 212 for the user may take a form similar to the following:

```

3 POST /AccountAlert.php HTTP/1.1
4 Host: 216.15.51.74
5 Content-Type: Application/XML
6 Content-Length: 718
7 <?XML version = "1.0" encoding = "UTF-8"?>
8 <AccountAlert>
9     <Time> 17:40:40 </Time>
10    <Date> 09-09-2015 </Date>
11    ...
12    <Item>
13        <Item1>
14            <ItemCode> DRUG-23401 </ItemCode>
15            <Category> DRUG </Category>
16            <Sub-Category> Non-Prescription </Sub-Category>
17            <ItemAlias> NyQuil </ItemAlias>
18            <RecommendedAct>
19                <Act1> FSA </Act1>
20                <Act2> HSA </Act2>
21                ...
22            </RecommendAct>
23        ...
24    </Item1>
25    </Item>
26    <Message> "Would you like to pay your NyQuil with your FSA/HSA?"
27 </Message>
28    <HardwareID> JS001 </HardwareID>
29    <Address> 01:23:45:67:89:ab </Address>
30    ...
31 </AccountAlert>
  
```

32 **[0064]** In another example, the restricted-use account status 206 may identify
 33 eligible items for each restricted-use account if there is any. For example, an exemplary
 34 XML-formatted message 206 for the user may take a form similar to the following:

```

35 POST /AccountAlert.php HTTP/1.1
36 Host: 216.15.51.74
37 Content-Type: Application/XML
38 Content-Length: 718
39 <?XML version = "1.0" encoding = "UTF-8"?>
40 <AccountStatus>
41     <Time> 17:40:40 </Time>
42     <Date> 09-09-2015 </Date>
43     ...
44     <user>
45         <user_id> JS-001 </user_id>
46         <user_name> John Smith </user_name>
47         <wallet_id> js_wallet </wallet_id>
48         <password> ***** </password>
49         ...
50     </user>
51
52     <ru_account1>
53         <account_name> FSA </account_name>
  
```

```

1      <account_category> healthcare </account_category>
2      <account_type> prepaid </account_type>
3      <item>
4          <Item1>
5              <ItemCode> DRUG-23401 </ItemCode>
6              <Category> DRUG </Category>
7              <Sub-Category> Non-Prescription </Sub-Category>
8              <ItemAlias> NyQuil </ItemAlias>
9              ...
10         </Item1>
11         <item_2>
12             <item_code> DRUG-23402 </item_code>
13             <category> drug </category>
14             <sub_category> antibiotics </sub_category>
15             <item_alias> penicillin </item_alias>
16             ...
17         </item_2>
18         ...
19     </Item>
20     ...
21 </ru_account1>
22
23 <ru_account2>
24     <account_name> food stamp </account_name>
25     <account_category> government </account_category>
26     <account_type> food </account_type>
27     <item>
28         <Item1>
29             <ItemCode> food-2307 </ItemCode>
30             <Category> grocery </Category>
31             <Sub-Category> produce </Sub-Category>
32             <ItemAlias> flour </ItemAlias>
33             ...
34         </Item1>
35         <item_2>
36             <item_code> food-3394 </item_code>
37             <category> grocery </category>
38             <sub_category> bakery </sub_category>
39             <item_alias> cupcake </item_alias>
40             ...
41         </item_2>
42         ...
43     </Item>
44     ...
45 </ru_account2>
46 ...
47 </Accountstatus>
48
49

```

[0065] In one implementation, the user may select an account 214 to proceed with checkout, e.g., by selecting whether to accept payment with a restricted-use account as recommended by the PoS terminal, or to skip the recommendation and proceed with a bank account checkout. In one implementation, the user may submit account information together with the account selection message 214 to the PoS terminal, e.g.,

1 by tapping on an electronic wallet, by swiping a magnetic payment card, and/or the like.
 2 In one implementation, an exemplary XML formatted user account selection message
 3 214 may take a form similar to the following:

```

4 POST /AccountSelection.php HTTP/1.1
5 Host: 216.15.51.74
6 Content-Type: Application/XML
7 Content-Length: 718
8 <?XML version = "1.0" encoding = "UTF-8"?>
9 <AccountSelection>
10     <TimeStamp> 11:11:23 </TimeStamp>
11     <Date> 09-09-2015 </Date>
12     <User>
13         <UserName> John Smith </UserName>
14         <UserID> JS0000 </UserID>
15         ...
16     </User>
17     <Account>
18         <AccountNo> 0000 0000 0000 </AccountNO>
19         <AccountType> FSA </AccountType>
20         <Employer> SuperFast Software Co. </Employer>
21         <AccountHolder> John Smith </AccountHolder>
22         <RoutingNumber> 000000000 </RoutingNumber>
23         ...
24     </Account>
25     ...
26 </AccountSelection>
27

```

28 **[0066]** Within implementation, upon receiving the user's account selection, the
 29 merchant 210 may generate a payment request message 216 to the RUAP server 220,
 30 e.g., a HTTP POST message to the RUAP server 220, wherein the XML-formatted
 31 payment request 216 message may take a form similar to:

```

32 POST /PaymentRequest.php HTTP/1.1
33 Host: www.ruap.com
34 Content-Type: Application/XML
35 Content-Length: 718
36 <?XML version = "1.0" encoding = "UTF-8"?>
37 <PaymentRequest>
38     <RequestID> SHP-0001 </RequestID>
39     <RequestDate> 09-09-2015 </BillDate>
40     <RequestTimeStamp> 14:23:56 </RequestTimeStamp>
41     <User>
42         <UserName> John Smith </UserName>
43         <UserID> JS0000 </UserID>
44         <DeviceID> JS-001 </DeviceID>
45         ...
46     </User>
47     <Account>
48         <AccountNo> 0000 0000 0000 </AccountNO>

```

```

1         <AccountType> FSA </AccountType>
2         <Employer> SuperFast Software Co. </Employer>
3         <AccountHolder> John Smith </AccountHolder>
4         <RoutingNumber> 000000000 </RoutingNumber>
5         ...
6     </Account>
7     <PurchaseItem>
8         <Item1>
9             <ItemCode> DRUG-23401 </ItemCode>
10            <Category> DRUG </Category>
11            <Sub-Category> Non-Prescription </Sub-Category>
12            </ItemName> NyQuil Cold Medicine </ItemName>
13            <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
14            <Quantity> 1 </Quantity>
15            <UnitPrice> 12.99 </UnitPrice>
16            <TotalAmt> 12.99 </TotalAmt>
17            ...
18        </Item1>
19        ...
20    </PurchaseItem>
21    <TotalAmount> 12.99 </TotalAmount>
22 ...
23 </PaymentRequest>
24

```

25 **[0067]** In one implementation, the RUAP server 220 may obtain a routing
26 number 217 from the received payment request 216, based on which the RUAP may
27 determine where to forward the payment authorization request 218, which may take a
28 similar form to the payment request 216. For example, if the user has elected a credit
29 card account for payment, the RUAP server 220 may route the payment authorization
30 request 218 to the credit card issuing bank. In another example, when the user elected a
31 FSA/HSA account for payment, the RUAP server 220 may route the payment
32 authorization request 218 the FSA/HSA account manager.

33 **[0068]** In one implementation, the account issuing network 230 may review and
34 verify the payment request. For example, the account issuer may verify whether the
35 account has sufficient remaining balance to furnish the payment, whether the MCC of
36 the purchase item is eligible for usage of the account, and/or the like, e.g., 221. In one
37 implementation, the account issuer network 230 may generate a payment authorization
38 response message 222, e.g., a HTTPS POST message in the form of data formatted

1 according to the XML. Below is an example HTTP(S) POST message including an XML-
 2 formatted message of the authorization response 222 for the RUAP server 220:

```

3 POST /AuthorizationResponse.php HTTP/1.1
4 Host: www.issuer.com
5 Content-Type: Application/XML
6 Content-Length: 718
7 <?XML version = "1.0" encoding = "UTF-8"?>
8 <AuthorizationResponse>
9     <Time> 17:45:40 </Time>
10    <Date> 09-09-2015 </Date>
11    <User>
12        <UserName> John Smith </UserName>
13        <UserID> JS0000 </UserID>
14        <DeviceID> JS-001 </DeviceID>
15    ...
16    </User>
17    <Issuer>
18        <IssuerID> FSA-CA-001 </IssuerID>
19        <RoutingNo> 00000000 </RoutingNo>
20    ...
21    </Issuer>
22    <Account>
23        <AccountNo> 0000 0000 0000 </AccountNO>
24        <AccountType> FSA </AccountType>
25        <Employer> SuperFast Software Co. </Employer>
26    ...
27    </Account>
28    <PurchaseItem>
29        <Item1>
30            <ItemCode> DRUG-23401 </ItemCode>
31            <Category> DRUG </Category>
32            <Sub-Category> Non-Prescription </Sub-Category>
33            </ItemName> NyQuil Cold Medicine </ItemName>
34            <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
35            <Quantity> 1 </Quantity>
36            <UnitPrice> 12.99 </UnitPrice>
37            <TotalAmt> 12.99 </TotalAmt>
38        ...
39        </Item1>
40        ...
41    </PurchaseItem>
42    <TotalAmount> 12.99 </TotalAmount>
43    <MCCEligibility> Good </MCCEligibility>
44    <Balance> Good </Balance>
45    <Remaining> 1000.00 </Remaining>
46    ...
47 </AuthorizationResponse >
48

```

49 **[0069]** In the above example, the account issuer, e.g., a FSA account manager,
 50 may verify the item category "drug" is eligible for FSA usage, and the remaining balance
 51 "\$1000.00" is sufficient to cover the requested payment amount of "\$14.99."

1 **[0070]** Upon receiving the payment authorization 222, the RUAP server 220 may
2 process the payment by transacting a requested amount from the user account to eh
3 merchant 210, and send an approval notice 223 to the merchant 210. For example, the
4 RUAP server 220 may send the fund transfer request, which may take a form similar to
5 the format in compliance with electronic fund transfers (EFT), and in some
6 embodiments, it may be directly made to the merchant 210 via a third party bank, e.g.,
7 absent the direction of the RUAP server 220. In another implementation, the RUAP
8 server 220 may be integrated with a payment network, e.g., VisaNet, etc., which may
9 facilitate the payment processing. In one implementation, the RUAP server 220 may
10 debit the approved payment amount from the user's account and credit to the merchant
11 210. In another example, the fund transfer message may take a form similar to the Visa
12 Single Message System (SMS) format, Visa Original Credit Transaction (OCT) format,
13 and/or the like. Further implementations of the purchase transaction authorization are
14 discussed in FIGURES 17A-21.

15 **[0071]** In one implementation, upon obtaining the approval notice 223 of the
16 payment transaction, the merchant 210 may generate a receipt 225 to the user. For
17 example, the user may obtain a printed receipt 213 from a PoS terminal. For another
18 example, the user may obtain an electronic receipt (e.g., via online shopping site, via
19 NFC handshake with the PoS terminal from a mobile device, etc.). In one
20 implementation, receipt 225 may list the user's purchased item information, payment
21 account information, merchant information, and/or the like. In another
22 implementation, the RUAP server 220 may incorporate information in the receipt into a
23 transaction record 226 for storage. For example, an example of the transaction record
24 266 for the RUAP server may take a form similar to the following:

```

1 POST /TransactionRecord.php HTTP/1.1
2 Host: www.ruap.com
3 Content-Type: Application/XML
4 Content-Length: 718
5 <?XML version = "1.0" encoding = "UTF-8"?>
6 <Transaction>
7     <TransactionID> 000000 </TransactionID>
8     <TransactionDate> 09-09-2015 </TransactionDate>
9     <RequestTime> 19:30:27 </RequestTime>
10    <ReceiptTime> 19:31:56 </ReceiptTime>
11    <User>
12        <UserName> John Smith </UserName>
13        <UserID> JS0000 </UserID>
14    ...
15    </User>
16    ...
17    <Account>
18        <AccountNo> 0000 0000 0000 </AccountNO>
19        <AccountType> FSA </AccountType>
20        <Employer> SuperFast Software Co. </Employer>
21        <RoutingNo> 000000000 </RoutingNo>
22    ...
23    </Account>
24    <PurchaseItem>
25        <Item1>
26            <ItemCode> DRUG-23401 </ItemCode>
27            <Category> DRUG </Category>
28            <Sub-Category> Non-Prescription </Sub-Category>
29            <ItemName> NyQuil Cold Medicine </ItemName>
30            <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
31            <Quantity> 1 </Quantity>
32            <UnitPrice> 12.99 </UnitPrice>
33            <TotalAmt> 12.99 </TotalAmt>
34        ...
35        </Item1>
36    ...
37    </PurchaseItem>
38    <TotalAmount> 12.99 </TotalAmount>
39    <Verification> Good </Verification>
40    <Merchant>
41        <MerchantID> MC001 </Merhcant>
42        <MerchantName> Walgreens </MerchantName>
43        <MerchantAddress> ... </MerchantAddress>
44        <PoSID> MCC-001-001 </PoSID>
45    ...
46    </Merchant>
47    <TransferLog>
48        <Transfer1>
49            <Amount> 14.99 </Amount>
50            <Payor> 0000 0000 0000 </Payor>
51            <Payee> 0000 0000 0002 </Payee>
52            <Status> Verified </Status>
53        ...
54        </Transfer1>
55    ...
56    </TransferLog>
57 ...
58 </Transaction>

```

59 **[0072]** In another implementation, the database 219 may be a relational database
60 responsive to Structured Query Language ("SQL") commands. The RUAP server may

1 execute a hypertext preprocessor ("PHP") script including SQL commands to query the
 2 database for user, transaction data, and/or the like. An example PHP/SQL command
 3 listing, illustrating substantive aspects of storing a transaction record 266 in the
 4 database:

```

5 <?PHP
6 header('Content-Type: text/plain');
7 mysql_connect("202.155.66.130", $DBserver, $password); // access database server
8 mysql_select("TRANSACTIONS.SQL"); // select database to append
9 mysql_query("INSERT INTO Transactions (transaction_id, transaction_date,
10 requested_time, receipt_time, user_id, user_name, account_no, account_type, employer,
11 routing_no, item_code, category, sub_category, item_name, item_description,
12 item_quantity, unit_price, total_amount, verification_status, merchant_id,
13 merchant_name, PoS_id, transfer_log, payee_id, payor_id, transfer_amount ...)
14 VALUES ($transaction_id$, $transaction_date$, $requested_time$, $receipt_time$,
15 $user_id$, $user_name$, $account_no$, $account_type$, $employer$, $routing_no$,
16 $item_code$, $category$, $sub_category$, $item_name$, $item_description$,
17 $item_quantity$, $unit_price$, $total_amount$, $verification_status$, $merchant_id$,
18 $merchant_name$, $PoS_id$, $transfer_log$, $payee_id$, $payor_id$, $transfer_amount$...);
19 //
20 add data to table in database
21 mysql_close("TRANSACTIONS.SQL"); // close connection to database
22 ?>
23

```

24 **[0073]** In one implementation, the RUAP may access and retrieve information
 25 from one or more online databases 219. Some databases contain a rule for a payment
 26 made towards the balance due bill for the healthcare provided by the healthcare worker
 27 to the user. By way of example, and not by way of limitation, a database can contain a
 28 negative wealth impacting (e.g., deduction, liability, insurance, debt, tax, negative
 29 interests, and/or other wealth reducing factor) rule pertaining to payment to the
 30 healthcare provider for the healthcare to the user. Another database can contains an
 31 insurance rule pertaining to payment for the healthcare to the user. Other online
 32 databases accessible by the RUAP to retrieve information can contain data specific to
 33 the user and an insured entity financially responsible for the user, as well as currency
 34 balances in each of one or more accounts respectively issued by an issuer.

1 **[0074]** FIGURE 2B provides a data flow illustrating alternative implementations
2 of RUAP entity interactions within embodiments of the RUAP. Within implementations,
3 instead of obtaining account recommendation (e.g., 212 in FIGURE 2A) at a smart PoS
4 terminal equipped with the merchant 210, such account recommendation may be
5 provided by the RUAP server 220 upon user submitting purchase item information to
6 the RUAP server 220. In one implementation, in one implementation, upon user
7 submitting purchase item information 203 to the merchant 210, which in turn
8 generates a payment bill 208, the user may obtain a bill 211 from the merchant. For
9 example, in one implementation, the merchant may print a paper bill 211 to the user. In
10 another example, the merchant may transmit an electronic bill of the purchase items 211
11 to the user's electronic wallet, e.g., see 1416 in FIGURE 14B. The user's electronic wallet
12 may then determine whether the purchase item information comprises any restricted-
13 use account eligible items. In further implementations, the merchant may generate a
14 QR code for display, as further discussed in FIGURE 2C.

15 **[0075]** Within alternative implementations, the user may check in with the RUAP
16 server 220 with bill information 227. For example, in one implementation, the user's
17 electronic wallet instantiated on a mobile device may automatically send a notification
18 to the RUAP server 220 upon identifying the user's GPS coordinates reflect user's
19 location at a merchant store 210. In another implementation, the user's browser may
20 send a cookie to the RUAP server 220 indicating the user has entered into a merchant
21 shopping site. The user-merchant check-in may take place prior to, after, or in
22 accordance with the user obtaining a purchase bill from the merchant.

1 **[0076]** Within implementations, the user may also send purchase bill information
 2 227 to the RUAP server 220. For example, in one implementation, the user may
 3 forward an electronic bill to the RUAP server 220 via wallet. In another example, the
 4 user may operate a camera-enabled mobile device to capture an image of a paper bill, a
 5 checkout list displayed at a PoS terminal screen, a QR code generated at a PoS terminal
 6 and/or a checkout page at the user's online shopping site, and/or the like, and
 7 incorporate the captured image in the message 227 to the RUAP server 220. In another
 8 implementation, the user may proceed with 241 in FIGURE 2C to obtain bill
 9 information embedded in a QR code.

10 **[0077]** For example, in one implementation, the user's mobile device may
 11 generate a HTTPS POST message in the form of data formatted according to XML,
 12 wherein the XML-formatted message of the user check-in with bill information message
 13 227 for the RUAP server 220 may take a form similar to the following:

```

14 POST /UserCheckin.php HTTP/1.1
15 Host: 216.15.51.74
16 Content-Type: Application/XML
17 Content-Length: 718
18 <?XML version = "1.0" encoding = "UTF-8"?>
19 <UserCheckin>
20     <Time> 17:45:40 </Time>
21     <Date> 09-09-2015 </Date>
22     <User>
23         <UserName> John Smith </UserName>
24         <UserID> JS0000 </UserID>
25         <DeviceID> JS-001 </DeviceID>
26     ...
27 </User>
28 <GPS> 29.199505,-90.041242 </GPS>
29 <Bill>
30     <Item1>
31         <ItemCode> FOOD-9875 </ItemCode>
32         <Category> FOOD </Category>
33         <Sub-Category> Breakfast </Sub-Category>
34         </ItemName> Cereal </ItemName>
35         <Description> whole grain original 10 oz </Description>
36         <Quantity> 1 </Quantity>
37         <UnitPrice> 4.99 </UnitPrice>
38         <TotalAmt> 4.99 </TotalAmt>
39     ...
40 </Item1>
  
```

```

1         <Item2>
2             <ItemCode> DRUG-23401 </ItemCode>
3             <Category> DRUG </Category>
4             <Sub-Category> Non-Prescription </Sub-Category>
5             </ItemName> NyQuil Cold Medicine </ItemName>
6             <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
7             <Quantity> 1 </Quantity>
8             <UnitPrice> 12.99 </UnitPrice>
9             <TotalAmt> 12.99 </TotalAmt>
10            ...
11        </Item2>
12        ...
13    </Bill>
14    <TotalAmount> 16.99 </TotalAmount>
15    <Merchant>
16        <MerchantID> MC001 </Merhcant>
17        <MerchantName> Walgreens </MerchantName>
18        <MerchantAddress> ... </MerchantAddress>
19        <PoSID> MCC-001-001 </PoSID>
20    ...
21 </Merchant>
22 ...
23 </UserCheckin>
24 [0078]

```

25 **[0079]** In another implementation, the merchant may submit bill information 212
 26 to the RUAP server 220. As such, the RUAP server may forward bill information to the
 27 user via email, SMS, instant messaging, wallet messaging, and/or the like.

28 **[0080]** In one implementation, upon receiving the user check-in information 227,
 29 the RUAP server 220 may query a user database to retrieve user's profile, and determine
 30 a list of user payment accounts registered with the electronic wallet. In one
 31 implementation, the RUAP server 220 may retrieve routing number 229 with user
 32 registered accounts, and submit an update request 231 to the account issuer network
 33 230 for account balance information. The issuer network 230 may in turn retrieve the
 34 account information for status check 233a, and generate a status update 233b to the
 35 RUAP server. For example, in one implementation, the issuer network 230 may
 36 generate a HTTPS POST message in the form of data formatted according to XML,
 37 wherein the XML-formatted message of the account update message 233a-b for the
 38 RUAP server 220 may take a form similar to the following:

```

1 POST /AccountUpdate.php HTTP/1.1
2 Host: www.issuer.com
3 Content-Type: Application/XML
4 Content-Length: 718
5 <?XML version = "1.0" encoding = "UTF-8"?>
6 <AccountUpdate>
7     <Time> 17:45:40 </Time>
8     <Date> 09-09-2015 </Date>
9     <User>
10         <UserName> John Smith </UserName>
11         <UserID> JS0000 </UserID>
12         <DeviceID> JS-001 </DeviceID>
13     ...
14 </User>
15 <Account>
16     <AccountNo> 0000 0000 0000 </AccountNO>
17     <AccountType> FSA </AccountType>
18     <Employer> SuperFast Software Co. </Employer>
19     <RoutingNo> 000000000 </RoutingNo>
20     <Balance> 649.00 </Balance>
21     <Term>
22         <start_date> 01/01/2015 </start_date>
23         <end_date> 12-31-2015 </end_date>
24     </Term>
25     ...
26 </Account>
27 <Status> Good </Status>
28 ...
29 </AccountUpdate>
30

```

[0081] In another implementation, the RUAP server 220 may obtain purchase item information from the user provided bill information and perform item eligibility check 228 to see if any item from the bill is eligible for a user enrolled restricted-account usage. For example, in one implementation, the RUAP server 220 may extract purchase item information from the bill submitted with message 227, e.g., a snap bill image, etc., by performing an optical character recognition (OCR) procedure. The RUAP server 220 may then query a user database for user enrolled account, and the information retrieved by the RUAP from the online databases is processed by an optimization algorithm that operates on the rules in the retrieved information. The RUAP may derive a recommendation that includes a currency payment amount to pay against the balance due bill respectively from each said currency balance of the one or more accounts issued by respective issuers. In further implementations, the recommendation may be performed and rendered on the web enabled mobile computing device for approval by

1 the user. If the recommendation is approved, the enabled mobile computing device
2 transmits the recommendation to the POS. In one implementation, the POS may
3 transmit the recommendation for authorization processing of each currency payment
4 amount to pay against the balance due bill respectively from each currency balance of
5 each account as provided by the recommendation. In a further implementation, the
6 RUAP may substantially maximize currency payments pay against the balance due bill,
7 as well as substantially minimize out-of-pocket payments pay against the balance due
8 bill. Further implementations of account recommendations are illustrated in FIGURES
9 5D-5F.

10 **[0082]** In one implementation, the RUAP server 220 may provide an account
11 listing 235 (add a data structure) to the user, e.g., see 585 in FIGURE 5E, and the user
12 may submit an account selection 214 by tapping on an account. Upon obtaining the user
13 selected account, the merchant PoS terminal may process the payment request 216 in a
14 similar manner as discussed in FIGURE 2A. For example, in one implementation, an
15 exemplary XML-formatted recommended account listing may take a form similar to:

```
16 POST /AccountList.php HTTP/1.1
17 Host: www.issuer.com
18 Content-Type: Application/XML
19 Content-Length: 718
20 <?XML version = "1.0" encoding = "UTF-8"?>
21 <AccountList>
22     <Time> 17:45:56 </Time>
23     <Date> 09-09-2015 </Date>
24     <User>
25         <UserName> John Smith </UserName>
26         <UserID> JS0000 </UserID>
27         <DeviceID> JS-001 </DeviceID>
28     ...
29     </User>
30     <AccountList>
31         <Account1>
32             <AccountType> FSA </AccountType>
33             <AccountBalance> 100.00 </AccountBalance>
34         ...
35         <Account1>
36             <Account2>
37                 <AccountType> HSA </AccountType>
```

```

1           <AccountBalance> 1000.00 </AccountBalance>
2           ...
3           <Account2>
4               <Account3>
5                   <AccountType> HRA </AccountType>
6                   <AccountBalance> 200.00 </AccountBalance>
7               ...
8           <Account3>
9       </AccountList>
10      <Status> Good </Status>
11      ...
12 </AccountList>
13

```

14 **[0083]** FIGURE 2C provides a data flow diagram illustrating user-PoS interaction
15 for capturing bill information from a QR code within embodiments of the RUAP. In
16 some implementations, a user, e.g., 202, may desire to purchase a product, service,
17 offering, and/or the like (“product”), from a merchant, e.g., 210, via a merchant online
18 site or in the merchant’s store. The user may communicate with a merchant server, e.g.,
19 210, via a client such as, but not limited to: a personal computer, mobile device,
20 television, point-of-sale terminal, kiosk, ATM, and/or the like (e.g., 402). For example,
21 the user may provide user input, e.g., checkout input 241, into the client indicating the
22 user’s desire to purchase the product. For example, a user in a merchant store may scan
23 a product barcode of the product via a barcode scanner at a point-of-sale terminal. As
24 another example, the user may select a product from a webpage catalog on the
25 merchant’s website, and add the product to a virtual shopping cart on the merchant’s
26 website. The user may then indicate the user’s desire to checkout the items in the
27 (virtual) shopping cart. The client may generate a checkout request, e.g., 242, and
28 provide the checkout request, e.g., 243, to the merchant server. For example, the client
29 may provide a HTTP(S) GET message including the product details for the merchant
30 server in the form of data formatted according to the XML. Below is an example

- 1 HTTP(S) GET message including an XML-formatted checkout request for the merchant
- 2 server: (change it to the drug example)

```

3 GET /checkout.php HTTP/1.1
4 Host: www.merchant.com
5 Content-Type: Application/XML
6 Content-Length: 718
7 <?XML version = "1.0" encoding = "UTF-8"?>
8 <checkout_request>
9     <session_ID>4NFU4RG94</session_ID>
10    <timestamp>2011-02-22 15:22:43</timestamp>
11    <user_ID>john.q.smith@email.com</user_ID>
12    <client_details>
13        <client_IP>192.168.23.126</client_IP>
14        <client_type>smartphone</client_type>
15        <client_model>HTC Hero</client_model>
16        <OS>Android 2.2</OS>
17        <app_installed_flag>true</app_installed_flag>
18    </client_details>
19    <purchase_details>
20        <Item1>
21            <ItemCode> FOOD-9875 </ItemCode>
22            <Category> FOOD </Category>
23            <Sub-Category> Breakfast </Sub-Category>
24            </ItemName> Cereal </ItemName>
25            <Description> whole grain original 10 oz </Description>
26            <Quantity> 1 </Quantity>
27            <UnitPrice> 4.99 </UnitPrice>
28            <TotalAmt> 4.99 </TotalAmt>
29            ...
30        </Item1>
31        <Item2>
32            <ItemCode> DRUG-23401 </ItemCode>
33            <Category> DRUG </Category>
34            <Sub-Category> Non-Prescription </Sub-Category>
35            </ItemName> NyQuil Cold Medicine </ItemName>
36            <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
37            <Quantity> 1 </Quantity>
38            <UnitPrice> 12.99 </UnitPrice>
39            <TotalAmt> 12.99 </TotalAmt>
40            ...
41        </Item2>
42        <item_3>
43            <ItemCode> SU-Shampoo-001 </ItemCode>
44            <Category> WASH </Category>
45            <Sub-Category> hair product </Sub-Category>
46            </ItemName> SUAVE shampoo </ItemName>
47            <Description> SUAVE shampoo heat treatment 120 mL </Description>
48            <Quantity> 1 </Quantity>
49            <UnitPrice> 8.99 </UnitPrice>
50            <TotalAmt> 8.99 </TotalAmt>
51            ...
52        </item_3>
53    </purchase_details>
54 </checkout_request>
55
56

```

1 **[0084]** In some implementations, the merchant server may obtain the checkout
2 request from the client, and extract the checkout detail (e.g., XML data) from the
3 checkout request. For example, the merchant server may utilize a parser such as the
4 example parsers described below in the discussion with reference to FIGURE 24. The
5 merchant server may extract the product data, as well as the client data from the
6 checkout request. In some implementations, the merchant server may query, e.g., 244,
7 a merchant database, e.g., 219g, to obtain product data, e.g., 245, such as product
8 pricing, sales tax, offers, discounts, rewards, and/or other information to process the
9 purchase transaction. For example, the database may be a relational database
10 responsive to Structured Query Language (“SQL”) commands. The merchant server
11 may execute a hypertext preprocessor (“PHP”) script including SQL commands to query
12 the database for product data. An example PHP/SQL command listing, illustrating
13 substantive aspects of querying the database, is provided below:

```
14 <?PHP
15 header('Content-Type: text/plain');
16 mysql_connect("254.93.179.112", $DBserver, $password); // access database server
17 mysql_select_db("PRODUCTS.SQL"); // select database table to search
18 //create query
19 $query = "SELECT product_info product_price tax_linfo_list offers_list discounts_list
20 rewards_list FROM ProdTable WHERE product LIKE '%" $prod";
21 $result = mysql_query($query); // perform the search query
22 mysql_close("PRODUCTS.SQL"); // close database access
23 ?>
24
25
```

26 **[0085]** In some implementations, in response to obtaining the product data, the
27 merchant server may generate, e.g., 246a, a QR pay code, and/or secure display element
28 according to the security settings of the user. The merchant server may provide the QR
29 code to the client, so that the client may display the QR code, and the user may capture
30 the QR code using the user's device to obtain merchant and/or product data for
31 generating a purchase transaction processing request. In alternate implementations,

1 the merchant server may direct the client to communicate the product and/or merchant
2 data required to process the transaction to the user's device via an alternate
3 communication protocol, such as, but not limited to: Wi-Fi™, Bluetooth™, cellular
4 network, SMS, email and/or like communication protocols. For example, the merchant
5 server may direct the client to initiate a plug-in on its system to provide the alternate
6 communication service, and transmit the product and/or merchant data to the user's
7 device via the communication service.

8 **[0086]** In implementations utilizing a QR code, the merchant server may generate
9 a QR code embodying the product information, as well as merchant information
10 required by a payment network to process the purchase transaction. In some
11 implementations, the QR code may include at least information required by the user
12 device capturing the QR code to generate a purchase transaction processing request,
13 such as a merchant identifier (e.g., a merchant ID number, merchant name, store ID,
14 etc.) and a session identifier for a user shopping session associated with the shopping
15 website/brick-and-mortar store.

16 **[0087]** In some implementations, the merchant server may generate in real-time,
17 a custom, user-specific merchant-product XML data structure having a time-limited
18 validity period, such as the example 'QR_data' XML data structure provided below:

```
19 <QR_data>
20     <order_ID>4NFU4RG94</order_ID>
21     <timestamp>2011-02-22 15:22:43</timestamp>
22     <expiry_lapse>00:00:30</expiry_lapse>
23     <transaction_cost>$34.78</transaction_cost>
24     <alerts_URL>www.merchant.com/shopcarts.php?sessionID=AE4B4356</alerts_URL>
25     <user_ID>john.q.public@gmail.com</user_ID>
26     <client_details>
27         <client_IP>192.168.23.126</client_IP>
28         <client_type>smartphone</client_type>
29         <client_model>HTC Hero</client_model>
30         <OS>Android 2.2</OS>
31         <app_installed_flag>true</app_installed_flag>
```

```

1      </client_details>
2      <secure_element>www.merchant.com/securedyn/0394733/123.png</secure_element>
3 t>
4      <purchase_details>
5
6      <Item1>
7          <ItemCode> FOOD-9875 </ItemCode>
8          <Category> FOOD </Category>
9          <Sub-Category> Breakfast </Sub-Category>
10         </ItemName> Cereal </ItemName>
11         <Description> whole grain original 10 oz </Description>
12         <Quantity> 1 </Quantity>
13         <UnitPrice> 4.99 </UnitPrice>
14         <TotalAmt> 4.99 </TotalAmt>
15         ...
16     </Item1>
17     <Item2>
18         <ItemCode> DRUG-23401 </ItemCode>
19         <Category> DRUG </Category>
20         <Sub-Category> Non-Prescription </Sub-Category>
21         </ItemName> NyQuil Cold Medicine </ItemName>
22         <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
23         <Quantity> 1 </Quantity>
24         <UnitPrice> 12.99 </UnitPrice>
25         <TotalAmt> 12.99 </TotalAmt>
26         ...
27     </Item2>
28     <item_3>
29         <ItemCode> SU-Shampoo-001 </ItemCode>
30         <Category> WASH </Category>
31         <Sub-Category> hair product </Sub-Category>
32         </ItemName> SUAVE shampoo </ItemName>
33         <Description> SUAVE shampoo heat treatment 120 mL </Description>
34         <Quantity> 1 </Quantity>
35         <UnitPrice> 8.99 </UnitPrice>
36         <TotalAmt> 8.99 </TotalAmt>
37         ...
38     </item_3>
39 </purchase_details>
40 <merchant_params>
41     <merchant_id>3FBCR4INC</merchant_id>
42     <merchant_name>Acme Supermarket, Inc.</merchant_name>
43     <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
44 </merchant_params>
45 <QR_data>
46
47

```

[0088] In some implementations, the XML data may include a handle, alias, token, or pointer to information stored on a payment network server, rather than encoding all of the actual data required to initiate the transaction, so that the information encoded into the QR code may be advantageously minimized. In some implementations, the merchant may generate a QR code using the XML data. For example, the merchant server may utilize the PHP QR Code open-source (LGPL) library

1 for generating QR Code, 2-dimensional barcode, available at
2 <http://phpqrcode.sourceforge.net/>. For example, the merchant server may issue PHP
3 commands similar to the example commands provided below:

```
4 <?PHP
5 header('Content-Type: text/plain');
6 // Create QR code image using data stored in $data variable
7 Qrcode::png($data, 'qrcodeimg.png');
8 ?>
```

11 **[0089]** In alternate implementations, the merchant server may provide, e.g., 246b
12 the XML data to a RUAP server 220, along with a request to generate a QR code. For
13 example, the merchant server may utilize an API call to the RUAP server to request
14 generation of the QR code. The RUAP server may generate the QR code for the
15 merchant server, e.g., 246c, and provide, e.g., 246d, the QR code to the merchant server.
16 For example, the RUAP server may encode the information provided by the merchant
17 into the QR code, and may also advantageously encode security information, time
18 validity information, digital certificate information, anonymous shipping information,
19 QR code generation/processing fee information, etc. into the QR code.

20 **[0090]** In some implementations, the RUAP server may provide the merchant
21 server with an encryption key (e.g., a Rivest-Shamir-Adleman (“RSA”) private/public
22 key, digital certificate). The merchant may encrypt the custom, user-specific merchant-
23 product XML data structure using the encryption key to generate encrypted purchase
24 data (e.g., using the RSA algorithm). The merchant server may then encode the
25 encrypted data into the QR code. Such a scheme may be employed advantageously, in
26 various embodiments, by the RUAP server to authenticate the merchant for any
27 transaction processing requests related to the user-merchant shopping session.

1 **[0091]** In some implementations, pre-designed QR codes associated with
2 authenticated with pre-authenticated merchants may be provided to the user device.
3 For example, a user may be browsing an online website on the user's device. The user
4 device may make a HTTP(S) GET request for a webpage from a web server. In some
5 implementations, the web server may, in response to the user device's request for a
6 webpage, generate a query for advertisements to display on the webpage. For example,
7 the web server may search a database, or provide a request to an ad network server (e.g.,
8 Akamai) to provide advertisements for embedding into the webpage. In some
9 implementations, the ad network server may utilize keywords, metadata, etc. obtained
10 from the web server (e.g., keywords or metadata associated with the webpage, user
11 profile information, user ID, user browsing history from a cookie stored on the user
12 device, etc.). The ad network may utilize the keywords to generate a query of database(s)
13 for advertisements associated with the keywords, and may obtain advertisements to
14 provide. In some implementations, the ad network server may provide information (e.g.,
15 via an API call) on such advertisements (e.g., merchant name, merchant ID, product
16 name, product pricing information, related offers, etc.) to a RUAP server. The RUAP
17 server may generate a QR code based on the information provided by the ad network
18 server, such that a user device may snap the QR code to initiate a purchase transaction
19 for the goods and/or services associated with the QR code (e.g., as provided by the ad
20 network server to the RUAP server). The ad network server may provide the QR as part
21 of the advertisement to the web server, which may in turn embed the advertisement
22 including the QR code into the webpage before providing it to the user device. In
23 alternate implementations, the ad network server/web server may transmit a URL or
24 other identifier of the QR code (ultimately) to the user device, and the user device may

1 make a call (e.g., a HTTP(S) GET request) using the URL of the QR code (e.g., hosted on
2 the RUAP server) to obtain the QR code and display it for the user.

3 **[0092]** In some implementations, the merchant server may provide the QR code
4 to the client, e.g., 247. For example, the merchant server may provide a HyperText
5 Markup Language (“HTML”) page including a reference to the QR code image and/or
6 secure element image, such as the example HTML page below:

```
7 <html>  
8             
10            
12 </html>  
13  
14
```

15 **[0093]** In some implementations, the client may obtain the QR pay code, e.g., 247,
16 and display the QR code, e.g., 248 on a display screen associated with the client device.
17 In some implementations, the user may utilize a user device, e.g., 205, to capture the QR
18 code presented by the client device for payment processing. For example, the user may
19 provide payment input into the user device, e.g., 249. In various implementations, the
20 user input may include, but not be limited to: a single tap (e.g., a one-tap mobile app
21 purchasing embodiment) of a touchscreen interface, keyboard entry, card swipe,
22 activating a RFID/NFC enabled hardware device (e.g., electronic card having multiple
23 accounts, smartphone, tablet, etc.) within the user device, mouse clicks, depressing
24 buttons on a joystick/game console, voice commands, single/multi-touch gestures on a
25 touch-sensitive interface, touching user interface elements on a touch-sensitive display,
26 and/or the like. For example, the user device may obtain track 1 data from the user's
27 card (e.g., credit card, debit card, prepaid card, charge card, etc.), such as the example
28 track 1 data provided below:

1 %B123456789012345^PUBLIC/J.Q.^99011200000000000000**901*****?*
2 (wherein '123456789012345' is the card number of 'J.Q. Public' and has a CVV number of
3 901. '990112' is a service code, and *** represents decimal digits which change
4 randomly each time the card is used.)
5
6

7 **[0094]** In some implementation, the user device may determine whether an image
8 it has captured depicts a QR code. Depending on whether or not a QR code has been
9 captured, and also (optionally) depending on contents of the QR code, the user device
10 may redirect the user (e.g., via a web browser application executing on the user device)
11 to: a product, a merchant website, a product at a merchant website, a website and
12 including a command to add an item to a purchasing cart of the user associated with the
13 website, and/or the like. For example, the user device may execute a component such as
14 the example Quick Response Code Processing ("QRCP") component described below in
15 the discussion with reference to FIGURES 6A-B.

16 **[0095]** In some implementations, upon obtaining the user payment input and
17 capturing the QR code, the user device may generate a restricted-account option alert
18 250 (e.g., to notify the user an option to proceed with a restricted-account payment, etc.).
19 In an alternative implementation, upon capturing purchase item information from the
20 QR code, the user may proceed with 227 in FIGURE 2B, e.g., to check-in and submit
21 purchase item information to the RUAP server 220, which may in turn returns a
22 restricted-account option alert 261 for display on the user device 205.

23 **[0096]** In another implementation, upon obtaining the QR code, the user device
24 may provide a card authorization request, on behalf of the user, a HTTP(S) GET
25 message including the product order details for the RUAP server 220, in the form of
26 XML-formatted data. Below is an example HTTP(S) GET message including an XML-
27 formatted card authorization request for the RUAP server:

```

1 GET /purchase.php HTTP/1.1
2 Host: www.merchant.com
3 Content-Type: Application/XML
4 Content-Length: 1306
5 <?XML version = "1.0" encoding = "UTF-8"?>
6 <purchase_order>
7     <order_ID>4NFU4RG94</order_ID>
8     <alerts_URL>www.merchant.com/shopcarts.php?sessionID=AE4BB4356</alerts_URL>
9     <timestamp>2011-02-22 15:22:43</timestamp>
10    <user_ID>john.q.public@gmail.com</user_ID>
11    <client_details>
12        <client_IP>192.168.23.126</client_IP>
13        <client_type>smartphone</client_type>
14        <client_model>HTC Hero</client_model>
15        <OS>Android 2.2</OS>
16        <app_installed_flag>true</app_installed_flag>
17    </client_details>
18    <purchase_details>
19
20        <Item1>
21            <ItemCode> FOOD-9875 </ItemCode>
22            <Category> FOOD </Category>
23            <Sub-Category> Breakfast </Sub-Category>
24            </ItemName> Cereal </ItemName>
25            <Description> whole grain original 10 oz </Description>
26            <Quantity> 1 </Quantity>
27            <UnitPrice> 4.99 </UnitPrice>
28            <TotalAmt> 4.99 </TotalAmt>
29            ...
30        </Item1>
31        <Item2>
32            <ItemCode> DRUG-23401 </ItemCode>
33            <Category> DRUG </Category>
34            <Sub-Category> Non-Prescription </Sub-Category>
35            </ItemName> NyQuil Cold Medicine </ItemName>
36            <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
37            <Quantity> 1 </Quantity>
38            <UnitPrice> 12.99 </UnitPrice>
39            <TotalAmt> 12.99 </TotalAmt>
40            ...
41        </Item2>
42        <item_3>
43            <ItemCode> SU-Shampoo-001 </ItemCode>
44            <Category> WASH </Category>
45            <Sub-Category> hair product </Sub-Category>
46            </ItemName> SUAIVE shampoo </ItemName>
47            <Description> SUAIVE shampoo heat treatment 120 mL </Description>
48            <Quantity> 1 </Quantity>
49            <UnitPrice> 8.99 </UnitPrice>
50            <TotalAmt> 8.99 </TotalAmt>
51            ...
52        </item_3>
53    </purchase_details>
54    <merchant_params>
55        <merchant_id>3FBCR4INC</merchant_id>
56        <merchant_name>Acme Supermarket, Inc.</merchant_name>
57        <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
58    </merchant_params>
59    <account_params>
60        <account_name>John Smith</account_name>
61        <account_type>credit</account_type>
62        <account_num>123456789012345</account_num>
63        <billing_address>123 Green St., Norman, OK 98765</billing_address>

```

```
1         <phone>123-456-7809</phone>
2         <sign>/jqp</sign>
3         <confirm_type>email</confirm_type>
4         <contact_info>john.q.public@gmail.com</contact_info>
5     </account_params>
6     <shipping_info>
7         <shipping_adress>same as billing</shipping_adress>
8         <ship_type>expedited</ship_type>
9         <ship_carrier>FedEx</ship_carrier>
10        <ship_account>123-45-678</ship_account>
11        <tracking_flag>true</tracking_flag>
12        <sign_flag>>false</sign_flag>
13    </shipping_info>
14 </purchase_order>
15
16
```

17 **[0097]** In some implementations, the card authorization request generated by the
18 user device may include a minimum of information required to process the purchase
19 transaction. For example, this may improve the efficiency of communicating the
20 purchase transaction request, and may also advantageously improve the privacy
21 protections provided to the user and/or merchant. For example, in some
22 implementations, the card authorization request may include at least a merchant ID, a
23 session ID for the user's shopping session with the merchant, and a device ID of a device
24 (e.g., smartphone) of the user that is linked to the user's virtual wallet. In some
25 implementations, the QR code and messages sent to/from the QR-code capturing device
26 may include the source ID (e.g., identifier of the device generating the QR code), session
27 ID, merchant ID, item ID (e.g., model number), the charge amount, and/or transacting
28 device ID (e.g., the user's smartphone device).

29 **[0098]** In some implementations, the card authorization request may be provided
30 by the merchant server or point of sale terminal, instead of the user device. In some
31 implementations, the user, desiring security, may request, via the user device, the RUAP
32 server for a dynamically-generated card verification value code (dCVV™) to be utilized
33 along with the user's primary account number ("PAN," e.g., credit card number) in the
34 purchase transaction. In response, the payment network server may generate a dCVV™

1 code (e.g., using random number generation, MD5 hash of an input key, which may be
2 generated using the user ID, merchant ID, session ID, timestamp, combinations thereof,
3 and/or the like), and provide a session-specific dCVV™ code for the user to utilize along
4 with the user's PAN number. For example, the session-specific dCVV™ code may have
5 an expiry time (e.g., expiry in a one minute from issue). The user device may
6 communicate (e.g., via Bluetooth™, NFC, Wi-Fi, cellular, QR code, etc.) the PAN and
7 dCVV to the point-of-sale terminal, which may create the card authorization request.
8 For example, the user device may generate a QR payment code embedding the PAN and
9 dCVV numbers, and the point of sale terminal may snap an image of the user device-
10 generated QR payment code. The point of sale terminal may then generate and provide
11 the card authorization request to the RUAP server. The RUAP server may then be able
12 to validate the transaction by comparing the dCVV obtained from the merchant with the
13 dCVV it provided to the user device before the purchase transaction was initiated. If the
14 dCVV codes from the two sources (RUAP server and merchant) correspond properly to
15 each other, the RUAP server may continue processing the purchase transaction.

16 **[0099]** In some implementations, the card authorization request from a user
17 device may include encrypted data extracted from the QR code, which may have been
18 encrypted by the merchant server as part of a merchant authentication scheme. In some
19 implementations, the RUAP server may obtain the encrypted data from the card
20 authorization request provided by the user device, and attempt to decrypt the encrypted
21 data, e.g., using a RSA private/public that is complementary to the key the RUAP server
22 initially provided to the merchant server for encrypting the purchase data before
23 embedding it into the QR code. If the RUAP server is able to decrypt the purchase data,
24 then the merchant is authenticated as being a valid merchant. In some implementations,

1 the RUAP server may compare the purchase data decrypted from the card authorization
2 with data provided by the user/user device, to determine whether the data from these
3 different sources (user/user device, and merchant) correspond properly to each other.
4 Thus, in some implementations, the RUAP server may be able to authenticate the
5 merchant, and correlate the merchant to a specific user session or user device before
6 processing the transaction.

7 **[00100]** In some implementations, the RUAP server may provide a notification to
8 the user device that the transaction is authenticated and approved for transacting. In
9 alternate implementations, the RUAP server may proceed with transaction processing.
10 In some implementations, upon identifying that the user is in a session with the
11 merchant, the RUAP server may communicate with the user device to provide additional
12 features for the user. For example, in some implementations, the RUAP server may
13 provide a communication to the user device (e.g., via a HTTP(S) POST message) to
14 provide: a virtual storefront of the merchant; a depiction of an aisle of the merchant
15 associated with the products included in the card authorization request, a listing of
16 related items; and/or the like.

17 **[00101]** FIGURES 3A-3D provide logic flow diagrams illustrating various
18 embodiments of restricted-use account payment and reimbursement process flows
19 within embodiments of the RUAP. As shown in FIGURE 3A, in one implementation, a
20 user 302 may submit purchase items to a merchant 305 at a PoS checkout terminal, a
21 checkout page at an online shopping site, etc. The merchant 310 may scan purchase
22 item information 306 and determine whether an item is eligible for a restricted-use

1 account usage 308. For example, the merchant PoS terminal may be installed with a
2 software kit to query a list of eligible MCCs associated with a restricted-use account.

3 **[00102]** In another implementation, the user may submit wallet information to the
4 merchant terminal 310, e.g., by check-in at the merchant at 321, so that the merchant
5 may have wallet information of the user as to which restricted-use accounts the user has
6 enrolled with. The user wallet check-in may further comprise RUAP server updating
7 account balance information 323 to determine whether an account has sufficient funds
8 to fulfill a payment.

9 **[00103]** In one implementation, if an item is eligible for a restricted-use account
10 309, the merchant may generate an account recommendation 311, e.g., by sending a
11 RUAP alert to the user device via NFC, or display the message on a user PoS terminal
12 user interface, 312. The user may elect to submit a checkout preference 313, e.g., by
13 selecting to proceed checkout with the eligible restricted-use account or not. If the user
14 has selected to check out with the recommended restricted-use account 314, the
15 merchant may generate a separate bill for the eligible items 315 exclusively for payment
16 using the corresponding restricted-use account.

17 **[00104]** In another implementation, if no item is eligible for any restricted-use
18 account 309, or the user selects not to check out with the restricted-use account, the
19 merchant may generate a bill of all purchase items 317.

20 **[00105]** In one implementation, the user may submit an account selection 315 in
21 response to the generated bill at 315 or 317 to proceed with purchase payment
22 transaction (e.g., see also 1416 in FIGURES 14A-14B). upon receiving the account
23 selection 324, continuing on with 3B, the RUAP server may determine an issuer routing

1 number 326 from the account selection, and generate a payment authorization message
2 and route to the account issuer 328.

3 **[00106]** In one implementation, the issuer network 330 may receive and process
4 the payment transaction request 331. If the account issuer issues a restricted-account
5 332, the issuer may further inspect the item eligibility 334, e.g., by verifying the MCC in
6 the payment transaction request satisfies the restricted-use account requirement. The
7 issuer may further generate a response message 335 upon verifying item eligibility,
8 account balance, etc., to the RUAP server 320.

9 **[00107]** In one implementation, if the issuer response approves the transaction
10 336, the RUAP server 320 may transact the approved payment amount 337, e.g., by
11 transferring the approved amount from the user account to a merchant account, and
12 generate a transaction receipt message 339 to the merchant. In another implementation,
13 if the transaction is denied by the issuer (e.g., insufficient balance in the selected
14 account, MCC ineligible for a restricted-use account, etc.), the merchant may be notified
15 of the rejection in a transaction denial message 341.

16 **[00108]** In one implementation, upon completing the transaction, the merchant
17 may receive a transaction receipt message 343, based on which the merchant PoS
18 terminal, or the shopping site, may generate a receipt reflecting the purchase 346 to the
19 user. For example, the user may obtain a printed receipt from a PoS terminal. In
20 another example, the user may obtain an electronic receipt via email, SMS, electronic
21 wallet notifications, and/or the like.

22 **[00109]** With reference to FIGURE 3C, the user may check-in at a merchant store
23 via the electronic wallet. In one implementation, the user 302 may submit wallet check-

1 in information 351, e.g., GPS coordinates, user credentials, and/or the like. Upon
2 receiving the wallet check-in information 352, the RUAP server 320 may determine
3 merchant characteristics 353, e.g., based on GPS coordinates, etc., and tentatively
4 retrieve a list of restricted-use use account in the wallet 354. For example, the RUAP
5 server 320 may retrieve accounts the user has enrolled in the electronic wallet. In
6 another example, the RUAP server may retrieve user enrolled restricted-use accounts
7 based on merchant characteristics, e.g., FSA/HSA and other healthcare related
8 restricted-use accounts if the merchant is a hospital, physician office, dental office,
9 and/or the like; food stamp, etc., if the merchant is a grocery store, and/or the like.

10 **[00110]** In one implementation, the RUAP server 320 may extract routing
11 information and send a status update request to account issuer 356. The account issuer
12 may then retrieve user account information 357, e.g., balance information, valid term,
13 etc., and generate account status update information 358 for the user.

14 **[00111]** In one implementation, upon shopping with a merchant, the user 302 may
15 submit purchase information to the merchant 305, which may generate a purchase bill
16 and/or a QR code 360 (e.g., see FIGURE 3E). The user may snap an image of the bill
17 (and/or the QR code) and submit the bill image to the RUAP server 361, e.g., via the
18 electronic wallet.

19 **[00112]** In one implementation, upon receiving bill image information, the RUAP
20 server 320 may perform an OCR procedure to obtain item information 363. For
21 example, the RUAP server 320 may adopt OCR software such as, but not limited to
22 OmniPage, OCRFeeder, Scantron, Java OCR, and/or the like to extract information
23 from a bill image. In an alternative implementation, the user device may perform bill

1 analysis to obtain information as to the purchase item. For example, the user device
2 may decode the QR code and generate an account option based on the purchase item
3 category code 362.

4 **[00113]** With reference to FIGURE 3D, upon OCR scanning of the received bill
5 image, for each item on the bill 364, the RUAP server 320 may determine whether such
6 item is eligible for a restricted-use account. In one implementation, the RUAP server
7 may perform the match 365 based on enrolled restricted-accounts in the user's wallet;
8 such account information may be retrieved at 354. For example, if the user has FSA,
9 HSA accounts enrolled in the wallet, the RUAP server may query each item's MCC to
10 determine whether it is an eligible healthcare product.

11 **[00114]** In one implementation, if an item is eligible for a restricted-use account
12 365, upon obtaining the status update of the restricted-use account 366, the RUAP
13 server may generate a restricted-use account option list 367, e.g., including a
14 recommendation of accounts (as further discussed in FIGURES 5D-5F). Upon receiving
15 account recommendation 368, the user may submit a checkout account, e.g., whether to
16 proceed with a restricted-account checkout or not, and proceed with 324 in FIGURE 3A.

17 **[00115]** With reference to FIGURE 3E, in some implementations, a user may
18 desire to purchase a product, service, offering, and/or the like ("product"), from a
19 merchant via a merchant online site or in the merchant's store. The user may
20 communicate with a merchant server via a client. For example, the user may provide
21 user input, e.g., 371, into the client indicating the user's desire to checkout shopping
22 items in a (virtual) shopping cart. The client may generate a checkout request, e.g., 372,
23 and provide the checkout request to the merchant server. The merchant server may

1 obtain the checkout request from the client, and extract the checkout detail (e.g., XML
2 data) from the checkout request, e.g., 373. For example, the merchant server may utilize
3 a parser such as the example parsers described below in the discussion with reference to
4 FIGURE 24. The merchant server may extract the product data, as well as the client
5 data from the checkout request. In some implementations, the merchant server may
6 query, e.g., 374, a merchant database to obtain product data, e.g., 375, such as product
7 pricing, sales tax, offers, discounts, rewards, and/or other information to process the
8 purchase transaction.

9 **[00116]** In response to obtaining the product data, the merchant server may
10 generate, e.g., 376, a QR pay code, and/or secure display element according to the
11 security settings of the user (see, e.g., 358). For example, the merchant server may
12 generate a QR code embodying the product information, as well as merchant
13 information required by a payment network to process the purchase transaction. For
14 example, the merchant server may first generate in real-time, a custom, user-specific
15 merchant-product XML data structure having a time-limited validity period.

16 **[00117]** In some implementations, the merchant may generate QR code using the
17 XML data. For example, the merchant server may utilize the PHP QR Code open-source
18 (LGPL) library for generating QR Code, 2-dimensional barcode, available at
19 <http://phpqrcode.sourceforge.net/>. For example, the merchant server may issue PHP
20 commands similar to the example commands provided below:

```
21 <?PHP
22 header('Content-Type: text/plain');
23 // Create QR code image using data stored in $data variable
24 QRcode::png($data, 'qrcodeimg.png');
25 ?>
26
27
```

1 **[00118]** The merchant server may provide the QR pay code to the client, e.g., 376.
2 The client may obtain the QR pay code, and display the QR code, e.g., 377 on a display
3 screen associated with the client device. In some implementations, the user may utilize
4 a user device, e.g., 379, to capture the QR code presented by the client device for
5 payment processing. The client device may decode the QR code to extract the
6 information embedded in the QR code. For example, the client device may utilize an
7 application such as the ZXing multi-format 1D/2D barcode image processing library,
8 available at <http://code.google.com/p/zxing/> to extract the information from the QR
9 code. In some implementations, the user may provide payment input into the user
10 device, e.g., 378. Upon obtaining the user purchase input, the user device may generate
11 a card authorization request, e.g., 379, and provide the card authorization request to a
12 RUAP server.

13 **[00119]** In another implementation, upon obtaining information from the QR code,
14 the user device may submit an account selection to the RUAP server (e.g., see 324 in
15 FIGURE 3A) to proceed with a purchase transaction. In further implementation, the
16 user device may submit the QR code to the RUAP server for processing.

17 **[00120]** FIGURES 4A-4C provide data flow diagrams illustrating post-flight snap-
18 receipt restricted-use account reimbursement processing within embodiments of the
19 RUAP. Within alternative embodiments, instead of selecting a restricted-use account to
20 pay for an eligible purchase item as illustrated in FIGURES 2A-3E, a user may select a
21 general purpose bank account (e.g., a credit card, a debit card, a checking account, etc.)
22 for payment, and may submit reimbursement requests for eligible items after the
23 transaction via a communication network 413.

1 **[00121]** In some implementations, a user, e.g., 401, may desire to request a
2 reimbursement for compensation, e.g., a refund and/or reallocation of funds for
3 restricted-use eligible purchase items. The user may have a purchase receipt, e.g., 411,
4 based on conducting a purchase transaction with a merchant, e.g., via the PTA/PTC
5 component discussed further below with reference to FIGURES 17A-19B. The RUAP
6 may provide an app for a user device, e.g., 402, of the user for requesting a
7 reimbursement for compensation. For example, the user may operate a device such as,
8 but not limited to: a personal computer, mobile device, television, point-of-sale terminal,
9 and/or the like, e.g., 402. For example, the app may be an executable application
10 developed using a Software Development Kit (SDK) such as iOS SDK 4, Xcode 4,
11 Android SDK, Visual Studio, Visual C++, Java EE 5 SDK, GTK+, GNUstep, wxWidgets,
12 and/or the like. In some implementations, the app executing on the user device may
13 provide a user interface using which the user may interact with the app executing on the
14 user device. For example, the user may provide various types of input to indicate a user
15 reimbursement request, e.g., 412, including but not limited to: keyboard entry, card
16 swipe, activating a RFID/NFC enabled hardware device (e.g., electronic card having
17 multiple accounts, smartphone, tablet, etc.), mouse clicks, depressing buttons on a
18 joystick/game console, voice commands, single/multi-touch gestures on a touch-
19 sensitive interface, touching user interface elements on a touch-sensitive display, and/or
20 the like. In some implementations, the app may include a security feature to allow a
21 user secure access to the interface. As an example, a user may enter a passcode to access
22 the interface of the app. As another example, the user may present a card (e.g., a credit
23 card, debit card, prepaid card, etc.) at the user device to gain access to the interface of
24 the app. For example, the user may swipe the card through a card reader of the user

1 device, present the card for a NFC card reader, Bluetooth reader, and/or the like. The
 2 user device may obtain, e.g., track 1 data from the user's card such as the example track
 3 1 data provided below:

4 %B123456789012345^PUBLIC/J.Q.^99011200000000000000**901*****?*
 5 (wherein '123456789012345' is the card number of 'J.Q. Public' and has a CVV number of
 6 901. '990112' is a service code, and *** represents decimal digits which change
 7 randomly each time the card is used.)
 8
 9

10 **[00122]** The user device may then authenticate the user based on, e.g., whether the
 11 user identification from the card data matches the identification of the user to whom the
 12 user device is registered, or whether the card number of the user matches the card of the
 13 user stored in a file on the user device, etc. Upon authentication, the app may provide
 14 the interface for the user. In some implementations, the user device executing the app
 15 may provide a number of outputs for the user including, but not limited to: sounds,
 16 music, audio, video, images, tactile feedback, vibration alerts (e.g., on vibration-capable
 17 client devices such as a smartphone etc.), and/or the like.

18 **[00123]** In some implementations, in response to the user's reimbursement
 19 request input, the user device ("client") may capture a receipt snapshot of the receipt,
 20 e.g., 413, and generate a user reimbursement request using the captured receipt
 21 snapshot, e.g., 414. For example, the client may obtain an image, a video, a live stream,
 22 etc. of the receipt. Within various implementations, the receipt image and/or video may
 23 comprise a variety of image/video format, such as, but not limited to JPEG, BMP, TIFF,
 24 MP4, AVI, MOV, and/or the like. In some implementations, the client may provide the
 25 obtained snapshot to a RUAP server, e.g., 404. For example, the client may send a
 26 (Secure) HyperText Transfer Protocol (HTTP(S)) POST/GET message, electronic mail
 27 message, Short Messaging Service (SMS) message, HTTP/Real Time Streaming Protocol

1 (RTSP) video stream, etc., including the captured receipt snapshot as part of a user
2 reimbursement request, e.g., 415. In another implementation, the user 401 may forward
3 an electronic receipt to the RUAP server 404 in the reimbursement request 415. In
4 various implementations, the snapshot of the receipt may be captured by a webcam, a
5 digital camera, a scanner, any image processing device, and/or the like. In a further
6 implementation, the imaging device may automatically send the receipt image to the
7 RUAP server for reimbursement, e.g., via email, SMS, instant messaging, social
8 messaging, wallet messaging, and/or the like.

9 **[00124]** In one implementation, the user may indicate a reimbursement account in
10 the reimbursement request 415, e.g., an account for deposit of the reimbursed funds. In
11 one implementation, the user may configure an account for reimbursement purpose via
12 electronic wallet configuration at account enrollment. In another implementation, the
13 RUAP server 404 may automatically deposit reimbursed funds into the account that has
14 been used for the transaction as a default reimbursement account.

15 **[00125]** In one implementation, the user device/client 402 may generate a HTTPS
16 POST message including the user reimbursement request 415. Below is an example
17 HTTP(S) POST message including an XML-formatted user reimbursement request 415
18 for the RUAP server 404:

```
19 POST /ReimbursementRequest.php HTTP/1.1
20 Host: 216.15.51.74
21 Content-Type: Application/XML
22 Content-Length: 1306
23 <?XML version = "1.0" encoding = "UTF-8"?>
24 <reimbursement_request>
25     <Time> 17:49:40 </Time>
26     <Date> 09-09-2015 </Date>
27     <User>
28         <UserName> John Smith </UserName>
29         <UserID> JS0000 </UserID>
30         <DeviceID> JS-001 </DeviceID>
31     ...
```

```
1         </User>
2         <ReceiptData>
3             <Image1> IMG_0050.JPG </Image1>
4             <Image2> IMG_0050.JPG </Image2>
5             ...
6         </ReceiptData>
7         <RU_account> FSA </RU_account>
8         <Reimburse_account> Chase *0689 </Reimburs_account>
9         ...
10    </reimbursement_request>
11
```

12 **[00126]** In the above example, the user submits a reimbursement request with
13 captured images of a transaction receipt to the RUAP server, and has indicated a
14 preferred restricted-use account is FSA, and has provided a "Chase *0689" bank account
15 for depositing the reimbursed funds. In another implementation, the user may or may
16 not need to indicate a restricted-account for reimbursement request, and the RUAP
17 server may automatically determine a restricted-use account with regard to an eligible
18 item.

19 **[00127]** In one implementation, when there are more than one enrolled restricted-
20 use accounts can be applied for reimbursement for an eligible item, e.g., prescription
21 drugs may be paid by FSA, HSA, LOC, etc., the RUAP server 404 may apply user
22 configured rules to determine which restricted-use accounts to use for reimbursement.
23 For example, the user configured rules may be further illustrated in FIGURES 5E-5F.

24 **[00128]** In some implementations, the RUAP server may parse the user
25 reimbursement request, and extract the captured snapshot, e.g., 416. The RUAP server
26 may generate an extraction request 417 to extract the data fields and values from the
27 snapshot of the purchase receipt. The RUAP server may provide the extraction request
28 to an extraction server, e.g., 405. The extraction server may obtain and parse the
29 extraction request, and may extract the data fields and data values from the snapshot of
30 the purchase receipt, e.g., 418. For example, the extraction server may utilize optical

1 character recognition techniques to extract data fields and their associated values from
 2 images, video frames of the snapshot of the receipt. The extraction server may provide
 3 the receipt data, e.g., 419, to the RUAP server.

4 **[00129]** For example, the extraction processor 405 may generate a HTTPS POST
 5 message including the extracted receipt data 419. Below is an example HTTP(S) POST
 6 message including an XML-formatted receipt data 419 for the RUAP server 404:

```

7 POST /ReceiptData.php HTTP/1.1
8 Host: 216.15.51.00
9 Content-Type: Application/XML
10 Content-Length: 1306
11 <?XML version = "1.0" encoding = "UTF-8"?>
12 <receipt_data>
13     <Time> 17:49:40 </Time>
14     <Date> 09-09-2015 </Date>
15     <receipt_image> IMG_0050.JPG </receipt_image>
16     <source> JS-001 </source>
17     <User>
18         <UserName> John Smith </UserName>
19         <UserID> JS0000 </UserID>
20         <DeviceID> JS-001 </DeviceID>
21     ...
22 </User>
23 <receipt_time> 17:43:40 </receipt_time>
24 <receipt_date> 09-09-2015 </receipt_date>
25 <merchant> Walgreens </merchant>
26 <purchase_item>
27     <item1>
28         <ItemCode> FOOD-9875 </ItemCode>
29         <Category> FOOD </Category>
30         <Sub-Category> Breakfast </Sub-Category>
31         </ItemName> Cereal </ItemName>
32         <Description> whole grain original 10 oz </Description>
33         <Quantity> 1 </Quantity>
34         <UnitPrice> 4.99 </UnitPrice>
35         <TotalAmt> 4.99 </TotalAmt>
36     ...
37 </Item1>
38 <Item2>
39     <ItemCode> DRUG-23401 </ItemCode>
40     <Category> DRUG </Category>
41     <Sub-Category> Non-Prescription </Sub-Category>
42     </ItemName> NyQuil Cold Medicine </ItemName>
43     <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
44     <Quantity> 1 </Quantity>
45     <UnitPrice> 12.99 </UnitPrice>
46     <TotalAmt> 12.99 </TotalAmt>
47     ...
48 </Item2>
49 </purchase_item>
50 <amount> 16.99 </amount>
51 <card_no> *****0689 </card_no>
52 ...

```

1 </receipt_data>
2

3 **[00130]** In some implementations, the RUAP server may parse the user
4 reimbursement request, and determine what information is required to be sent for
5 reimbursements processing based on the user reimbursement request, and perform
6 restricted-account eligibility per each parsed item from the receipt data e.g., 420. For
7 example, the if the user wishes to transfer a charge for a medication purchase from the
8 user's charge card to a restricted-use account, the RUAP server may determine that the
9 reimbursement message sent for reimbursements processing may require the user's full
10 name, residential address, social security number, restricted-use account details ,
11 and/or the like. The RUAP server may ensure that only the user data that is required for
12 reimbursements processing may be sent for reimbursements processing. For example,
13 the RUAP server may protect the user's bank account, charge card account information
14 from being provided to the reimbursements processor, to protect the user's private
15 and/or protected information. In some implementations, the RUAP server may
16 generate a server reimbursement message, e.g., 421, filing a reimbursement on behalf of
17 the user with a reimbursements processor, e.g., 406.

18 **[00131]** For example, the RUAP server may generate a HTTPS POST message
19 including the server reimbursement claim message 421. Below is an example HTTP(S)
20 POST message including an XML-formatted reimbursement message 421 for the RUAP
21 server 404:

```
22 POST /reimbursement_claim.php HTTP/1.1
23 Host: www.ruap.com
24 Content-Type: Application/XML
25 Content-Length: 1306
26 <?XML version = "1.0" encoding = "UTF-8"?>
27 <reimbursement_claim>
28     <Time> 17:49:40 </Time>
```

```

1      <Date> 09-09-2015 </Date>
2      <User>
3          <UserName> John Smith </UserName>
4          <UserID> JS0000 </UserID>
5          <DeviceID> JS-001 </DeviceID>
6      ...
7      </User>
8      <reimburse_item>
9          <Item1>
10             <ItemCode> DRUG-23401 </ItemCode>
11             <Category> DRUG </Category>
12             <Sub-Category> Non-Prescription </Sub-Category>
13             </ItemName> NyQuil Cold Medicine </ItemName>
14             <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
15             <Quantity> 1 </Quantity>
16             <UnitPrice> 12.99 </UnitPrice>
17             <TotalAmt> 12.99 </TotalAmt>
18         ...
19         </Item1>
20     ...
21     <reimburse_item>
22     <amount> 12.99 </amount>
23     <reimburse_account> FSA </reimburse_account>
24     <account_no> 000000000 </account_no>
25     <deposit_card_no> 0000 0000 0000 0000 </deposit_card_no>
26     ...
27 </reimbursement_claim>
28

```

29 **[00132]** The reimbursements process server may parse the server reimbursement
 30 message, extract reimbursement details, and process the reimbursement, e.g., 422. For
 31 example, the reimbursements process server may determine the validity of the
 32 reimbursements and determine an amount of funds to be debited from the user's
 33 restricted-use account, and an amount of funds to be credited to the user's charge card.
 34 The reimbursements process server may provide the reimbursement process results, e.g.,
 35 423, to the RUAP server. For example, an exemplary XML-formatted reimbursement
 36 process results 423 message may take a form similar to the following:

```

37 . POST /reimbursement_results.php HTTP/1.1
38 Host: www.ruap.com
39 Content-Type: Application/XML
40 Content-Length: 1306
41 <?XML version = "1.0" encoding = "UTF-8"?>
42 <reimbursement_results>
43     <Time> 17:49:54 </Time>
44     <Date> 09-09-2015 </Date>
45     <User>
46         <UserName> John Smith </UserName>
47         <UserID> JS0000 </UserID>

```

```

1         <DeviceID> JS-001 </DeviceID>
2     ...
3     </User>
4     <reimburse_item>
5         <Item1>
6             <ItemCode> DRUG-23401 </ItemCode>
7             <Category> DRUG </Category>
8             <Sub-Category> Non-Prescription </Sub-Category>
9             </ItemName> NyQuil Cold Medicine </ItemName>
10            <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
11            <Quantity> 1 </Quantity>
12            <UnitPrice> 12.99 </UnitPrice>
13            <TotalAmt> 12.99 </TotalAmt>
14        ...
15        </Item1>
16    ...
17    <reimburse_item>
18        <amount> 12.99 </amount>
19        <reimburse_account> FSA </reimburse_account>
20        <account_no> 000000000 </account_no>
21        <deposit_card_no> 0000 0000 0000 0000 </deposit_card_no>
22        <status>
23            <item_code> eligible </item_code>
24            <item_category> good </item_category>
25            <receipt_barcode> good </receipt_barcode>
26        ...
27        </status>
28    ...
29 </reimbursement_results>
30 [00133]

```

[00134] Continuing on with FIGURE 4B, in some implementations, the RUAP server may parse the reimbursement process results, and identify the user account(s) from which to debit/credit funds, as well as the amount(s) to debit/credit from the user account(s). In some implementations, the RUAP server may query a database, e.g., 425-26, for addresses (e.g., IP addresses, URLs) of the account(s) server maintaining the user account(s). The RUAP server may use the account server data, e.g., 426, to generate and provide funds transfer message(s), e.g., 427a-n, to the account server(s), e.g., 408a-n. The account server(s) may credit/debit funds from the user account(s) according to the indications in the funds transfer message(s), e.g., 428a-n, and provide funds transfer notifications, e.g., 429a-n, to the RUAP server. The RUAP server may provide notification(s), e.g., 430, of successful reimbursements processing and funds transfer notification(s) to the client, e.g., 402. In some implementations, the client may

1 render and display the obtained notifications from the RUAP server for the user, e.g.,
2 431.

3 **[00135]** Within implementations, the user may have indicated a restricted-use
4 account for reimbursement. For example, the user may have indicated in the
5 reimbursement request that the submitted receipt is for FSA reimbursement. In another
6 implementation, the user may not identify any eligible restricted-use account, and the
7 RUAP server may automatically identify potential restricted-use accounts for
8 reimbursement by extracting item information from the receipt. For example, the
9 RUAP server may retrieve restricted-use accounts the user has enrolled with RUAP, and
10 perform an exhaustive search on the items on the receipt to determine whether any item
11 can be eligible for any registered restricted-use account.

12 **[00136]** Similarly, the user may have indicated in the reimbursement request an
13 account for deposit the reimbursement amount, e.g., a credit card account, a debit
14 account, a checking account, and/or the like. In an alternative implementation, the
15 RUAP server may retrieve a default account configured by the user upon enrollment
16 with RUAP as the account to deposit reimbursement funds. In a further
17 implementation, the RUAP server may identify a payment account from the receipt data
18 and use the identified account as the deposit account. In such cases, as account number
19 appeared on the receipt usually has an abbreviated form, e.g., only the last 4 digits, etc.,
20 the RUAP server may identify whether any enrolled user account matches with the
21 abbreviated account number, and prompt the user to confirm whether the user desire to
22 deposit reimbursement funds into the identified account.

1 **[00137]** Within alternative embodiments, upon obtaining receipt data at 419 and
2 performing restricted-account eligibility match at 420, the RUAP may proceed with
3 Figure 4C to generate a reimbursement request for the restricted-use account issuers.
4 In a further implementation, the RUAP may be part of the account issuer, and may
5 perform claim validity check as to whether the claimed item is eligible for the restricted-
6 use account.

7 **[00138]** As shown in FIGURE 4C, when the RUAP server 404 has determined one
8 or more items identified in the receipt data are eligible for reimbursement for a
9 restricted-use account, the RUAP server may retrieve the restricted-account routing
10 information 432, and generate a balance inquiry 434 to the account issuer 408a-n. The
11 account issuer may then query 437a an issuer database 499 based on the user id and
12 obtain query results 437b reflecting the user account balance and status update 436.
13 The account issuer may then generate an account balance update message 436, which
14 may take a similar form to the status update 233 in FIGURE 2B, to the RUAP server 404.

15 **[00139]** In one implementation, the RUAP server 404 may generate restricted-
16 account recommendation 433a for reimbursement, as further discussed in FIGURES
17 5E-5F, based upon the available balance of the restricted-account balance. In one
18 implementation, the RUAP server 404 may optionally send the generated account
19 recommendation 433b for user confirmation, e.g., see 585 in FIGURE 5E. Upon user
20 affirmation by selecting the reimbursement account 439, the RUAP server 220 may
21 proceed with 424 in FIGURE 4A to generate a reimbursement claim for processing.

22 **[00140]** FIGURES 5A-C show logic flow diagrams illustrating example aspects of
23 processing a user claim via mobile claims administration in some embodiments of the

1 RUAP, e.g., a User Claim Processing (“UCP”) component 500. In some
2 implementations, a user may desire to request a claim for compensation, e.g., a refund.
3 The user may have a purchase receipt, e.g., 501, based on conducting a purchase
4 transaction with a merchant, e.g., via PTA component discussed further below with
5 reference to FIGURES 17A-17B. The user may provide input to request a user claim, e.g.,
6 502. In some implementations, in response to the user’s claim request input, the user
7 device (“client”) may capture a receipt snapshot of the receipt, e.g., 503, and generate a
8 user claim request using the captured receipt snapshot, e.g., 504. The client may
9 provide the obtained snapshot to a RUAP server, e.g., 505. In some implementations,
10 the RUAP server may parse the user claim request, e.g., 505, and extract the captured
11 snapshot, e.g., 506.

12 **[00141]** The RUAP server may generate an extraction request to extract the data
13 fields and values from the snapshot of the purchase receipt, e.g., 507. The RUAP server
14 may provide the extraction request to an extraction server, e.g., 508. The extraction
15 server may obtain and parse the extraction request, e.g., 508, and may extract the data
16 fields and data values from the snapshot of the purchase receipt, e.g., 509. For example,
17 the extraction server may utilize optical character recognition techniques to extract data
18 fields and their associated values from images, video frames of the snapshot of the
19 receipt. The extraction server may determine the data fields present in the receipt, as
20 well as the data values associated with the data fields in the receipt. The extraction
21 server may provide the receipt data, e.g., 512, to the RUAP server. In some
22 implementations, the RUAP server may parse the user claim request, e.g., 513, and
23 determine what information is required to be sent for claims processing based on the
24 user claim request, e.g., 514. In some implementations, the RUAP server may generate

1 a server claim message, e.g., 515, filing a claim on behalf of the user with a claims
2 process server, e.g., 516. The claims process server may parse the server claim message,
3 e.g., 516, extract claim details, and process the claim, e.g., 517. For example, the claims
4 process server may determine the validity of the claims and determine an amount of
5 funds to be debited from the user's FSA account, and an amount of funds to be credited
6 to the user's charge card, e.g., 518. The claims process server may provide the claim
7 process results, e.g., 519, to the RUAP server. In some implementations, the RUAP
8 server may parse the claim process results, e.g., 519, and identify the user account(s)
9 from which to debit/credit funds, e.g., 520, as well as the amount(s) to debit/credit from
10 the user account(s), e.g., 521. In some implementations, the RUAP server may query a
11 database, e.g., 522, for addresses (e.g., IP addresses, URLs) of the account(s) server
12 maintaining the user account(s). The RUAP server may use the account server data
13 obtained from the database, e.g., 523, to generate and provide funds transfer message(s),
14 e.g., 524, to the account server(s). The account server(s) may obtain and parse the funds
15 transfer messages, e.g., 525, and credit/debit funds from the user account(s) according
16 to the indications in the funds transfer message(s), e.g., 526. The account server may
17 generate and provide, e.g., 527, funds transfer notifications to the RUAP server. The
18 RUAP server may generate, e.g., 528, and provide notification(s), e.g., 529, of successful
19 claims processing and funds transfer notification(s) to the client. In some
20 implementations, the client may render, e.g., 530, and display the obtained notifications
21 for the user, e.g., 531.

22 **[00142]** Continuing on with FIGURE 5C, upon obtaining receipt data at 512 in
23 FIGURE 5A, the RUAP server may review each item on the receipt 531 to determine
24 whether it is eligible for a restricted-use account 532. In one implementation, if it is

1 eligible, the RUAP server may send an account status update request 533 to the
2 restricted-use account issuer. The issuer may in turn receive the status update request
3 534, and verify the user credential 535 sent along with the status update request. In one
4 implementation, the account issuer may query on the user account database 536 and
5 retrieve user account status 537, e.g., the remaining balance, etc.

6 **[00143]** In one implementation, the RUAP server may further determine whether
7 the user has configured or indicated a restricted-use account for reimbursement in the
8 original reimbursement request 538. If not, the RUAP server may generate an option
9 list for the user 539. For example, when an item on the receipt comprises healthcare
10 products/services, the RUAP server may suggest FSA/HSA account to the user.

11 **[00144]** In one implementation, upon receiving a restricted-use account option list
12 540, the user may submit a selection 541. Or alternatively, the RUAP server may
13 automatically select an account upon user configured restricted-account usage rules. In
14 further implementations, the user may submit a deposit account for the reimbursed
15 funds 542. The RUAP server may then proceed to generate reimbursement
16 authorization messages for restricted-use account issuers, e.g., 522 in FIGURE 5B. In
17 further implementation, the RUAP may determine an account that has been used to pay
18 for the transaction as showed on the receipt, e.g., an abbreviated credit card number
19 "***1234." The RUAP may retrieve an account enrolled in the wallet that has an account
20 number ends with "1234," and deposit the reimbursement amount into the account.

21 **[00145]** FIGURE 5D-5F provide logic flows and exemplary mobile UIs illustrating
22 restricted account recommendation for "in-flight" real-time PoS payment and/or post-
23 transaction ("post-flight") reimbursement claims within embodiments of the RUAP.

1 Within embodiments, continuing on with receiving receipt data at 521 in FIGURE 5C (or
2 receive user purchase bill information at 364 in FIGURE 3D), the RUAP may parse the
3 purchased item's merchant category code 551 to determine a type of the purchase 552.
4 In further implementations, during a real-time checkout (e.g., a PoS checkout or online
5 checkout, etc.), the RUAP 520 may determine the purchase type based on the GPS
6 information, terminal ID, and/or the like. For example, the user's location at a
7 physician's office may suggest a healthcare purchase, but a location at a grocery store
8 may suggest food purchase, e.g., 553.

9 **[00146]** In one implementation, if the product code and/or terminal ID shows the
10 purchased product comprises food, the RUAP may determine whether food voucher is
11 enrolled in the wallet 554. If there is a food stamp account 561, the RUAP may put the
12 food stamp/voucher account on top of a recommended account list 565.

13 **[00147]** In another implementation, if the product code and/or terminal ID shows
14 a healthcare purchase, the RUAP may determine a recommended plan based on user
15 specified rules. For example, if the user prefers to pay with FSA, the RUAP may
16 determine whether there is FSA 555 enrolled in the wallet. If yes, the RUAP may send a
17 balance inquiry 556 to the account issuer 570, which may verify the account credentials
18 (e.g., a token, etc.) 557 and return the available balance 558. The RUAP may proceed to
19 determine whether there is sufficient balance 560. If yes, the RUAP may put FSA on top
20 of a recommended account list 563. If not, the RUAP, may recommend FSA with the
21 remaining available balance 568. The RUAP may query for other enrolled restricted use
22 accounts 566 in a similar manner.

1 **[00148]** In one implementation, the RUAP may generate a prioritized list of
 2 accounts 572 and presented to the user 573 in the wallet payment page, e.g., as
 3 illustrated in FIGURES 5E-5F.

4 **[00149]** FIGURES 5E-5F provides a dollar amount payment flow illustrating RUAP
 5 account recommendation within embodiments of the RUAP. In one implementation, a
 6 user may set up accounts and spending rules for the enrolled restricted use accounts,
 7 e.g., at 521 in FIGURE 5B. For example, the RUAP rules may be illustrated in one
 8 example in the following table:

Primary Account:	Flexible Spending Account (FSA)
Balance:	\$500.00
End Date:	12/31/2015
Rules:	1. Only use for medical MCC 2. Use for purchases less than \$100.00 until 10/01/2015 3. After 10/01/2015, use available balance for all medical MCC purchases.
Second Account:	Health Savings Account (HSA)
Balance:	\$5000.00
Rules:	1. Use for medical MCC 2. Use for purchases greater than 2000.00 3. Use as tertiary fund for medical MCC purchases
Third Account:	Revolving Line of Credit (LOC)
Credit Line:	\$5000.00
Rules:	1. Use for any MCC 2. Use for purchases greater than \$100 but less than \$2000.00 3. Use as secondary account for medical purchase

9 **[00150]**

10 **[00151]** For example, as shown in FIGURE 5E, if a user 502 goes to a primary
 11 care physician on 06/08/2015, i.e., more than half a year to the end date to his FSA, and
 12 a medical bill indicates a co-pay amount of \$50.00 (e.g., at 581), the user may enter
 13 \$50.00 into the RUAP mobile application and indicate it is medical purchase. Upon
 14 verifying the eligibility of medical purchase, the RUAP 520 may retrieve applicable
 15 healthcare restricted use accounts, and determine the user has his FSA, HSA and LOC

1 accounts enrolled 582. The RUAP may then update the balance information of each
2 account with the account sponsors 570, e.g., see also 527 in FIGURE 5C.

3 **[00152]** In one implementation, the RUAP may assess the rules in the above
4 example, and provide a screen of options showing the remaining balances in the three
5 accounts 584, e.g., ranked as FSA (\$500.00), LOC(\$2900.00), HSA (\$5000.00). In one
6 implementation, the RUAP may list the available accounts in a prioritized order based
7 on the spending rules, showing the balance of each account 585. It should be noted that
8 although mobile user interface elements are depicted, web, desktop and other interfaces
9 are contemplated for all the user interfaces throughout the disclosure. In this example,
10 although LOC is the third account after the HSA, LOC is listed prior to HSA as the rule
11 specifies LOC is applied as secondary account for medical purchase. In one
12 implementation, the RUAP may put a default dollar amount as \$50.00 (e.g., 586) for
13 payment, or the user may change it by type a different amount.

14 **[00153]** For another example, as shown in FIGURE 5F, if the user 502 goes to a
15 physical therapist at 09/27/2015, i.e., approximately three months to the end date of
16 FSA, and the patient's responsibility is \$100.00, the user may enter \$100.00 into the
17 RUAP mobile component and confirm it is medical purchase 587. In FIGURE 5F, the
18 user may press a "pay" button and enter an amount and type of purchase 593. The
19 RUAP 520 may retrieve applicable healthcare restricted use accounts, and determine
20 the user has his FSA, HSA and LOC accounts enrolled 588. The RUAP may then update
21 the balance information of each account with the account sponsors 589, e.g., see also
22 527 in FIGURE 5C, responded by listing the remaining balances, e.g., FSA (\$75.00),
23 LOC (\$3200.00), and HSA (\$5000.00), etc. Upon applying the usage rules 590, in this

1 case, even if the FSA has insufficient funds, it is on top of the prioritized list because it
2 will expire at the end of the year. As the remaining balance in FSA is insufficient to
3 cover the amount due, the user may split the amount by selecting FSA to pay \$75.00 591
4 and LOC to pay the remaining $\$100 - \$75 = \$25$. For example, after paying \$75.00 with
5 FSA, the FSA account may have an updated balance of 0.00 592. The user may elect to
6 pay the remaining \$25.00 593 with the LOC account. The RUAP may send a report
7 summary to the user including the updated remaining balance of the accounts after
8 payment, and/or the like.

9 **[00154]** For another example, if the user received a surgery on 09/30/2015, i.e.,
10 approximately three months to the end date of FSA, and received a medical bill of
11 \$2000.00, but the current accounts have available balances of LOC (\$3000.00), FSA
12 (0), HSA (\$5000.00). In this case, the user may elect to select HSA for the payment.

13 **[00155]** FIGURES 6A-B show logic flow diagrams illustrating example aspects of
14 processing a bill image comprising a Quick Response code in some embodiments of the
15 RUAP, e.g., a user may capture a bill image with a quick response code to facilitate
16 payment as illustrated in FIGURE 2C. With reference to FIGURE 6A, in some
17 implementations, a virtual wallet application executing on a user device may determine
18 whether a QR code has been captured in an image frame obtained by a camera
19 operatively connected to the user device, and may also determine the type, contents of
20 the QR code. Using such information, the virtual wallet application may redirect the
21 user experience of the user and/or initiating purchases, update aspects of the virtual
22 wallet application, etc. For example, the virtual wallet application may trigger the
23 capture of an image frame by a camera operatively connected to the user device, 601.

1 The virtual wallet application may utilize an image segmentation algorithm to identify a
2 foreground in the image, 602, and may crop the rest of the image to reduce background
3 noise in the image, 603. The virtual wallet application may determine whether the
4 foreground image includes a QR code from which data can be reliably read (e.g., this
5 may not be so if the image does not include a QR code, or the QR code is partially
6 cropped, blurred, etc.), 604. For example, the virtual wallet application may utilize a
7 code library such as the ZXing multi-format 1D/2D barcode image processing library,
8 available at <http://code.google.com/p/zxing/> to try and extract the information from
9 the QR code. If the virtual wallet application is able to detect a QR code (605, option
10 “Yes”), the virtual wallet application may decode the QR code, and extract data from the
11 QR code. If the virtual wallet application is unable to detect a QR code (605, option
12 “No”), the virtual wallet application may attempt to perform Optical Character
13 Recognition on the image. For example, the virtual wallet application may utilize the
14 Tesseract C++ open source OCR engine, available at [www.pixel-](http://www.pixel-technology.com/freewarw/tessnet2)
15 [technology.com/freewarw/tessnet2](http://www.pixel-technology.com/freewarw/tessnet2), to perform the optical character recognition, 606.
16 Thus, the virtual wallet application may obtain the data encoded into the image, and
17 may continue if the data can be processed by the virtual wallet application. The virtual
18 wallet application may query a database using fields identified in the extracted data, for
19 a type of the QR code, 608. For example, the QR code could include an invoice/bill, a
20 coupon, a money order (e.g., in a P2P transfer), a new account information packet,
21 product information, purchase commands, URL navigation instructions, browser
22 automation scripts, combinations thereof, and/or the like.

23 **[00156]** In some embodiments, the QR code may include data on a new account to
24 be added to the virtual wallet application (see 609). The virtual wallet application may

1 query an issuer of the new account (as obtained from the extracted data), for the data
2 associated with the new account, 610. The virtual wallet application may compare the
3 issuer-provided data to the data extracted from the QR code, 611. If the new account is
4 validated (611, option “Yes”), the virtual wallet application may update the wallet
5 credentials with the details of the new account, 613, and update the snap history of the
6 virtual wallet application using the data from the QR code, 614.

7 **[00157]** With reference to FIGURE 6B, in some embodiments, the QR code may
8 include data on a bill, invoice, or coupon for a purchase using the virtual wallet
9 application (see 615). The virtual wallet application may query merchant(s) associated
10 with the purchase (as obtained from the extracted data), for the data associated with the
11 bill, invoice, or coupon for a purchase (e.g., offer details, offer ID, expiry time, etc.), 616.
12 The virtual wallet application may compare the merchant-provided data to the data
13 extracted from the QR code, 617. If the bill, invoice, or coupon for a purchase is
14 validated (618, option “Yes”), the virtual wallet application may generate a data
15 structure (see e.g., XML QR_data structure in description above with reference to
16 FIGURES 4-5) including the QR-encoded data for generating and providing a card
17 authorization request, 619, and update the snap history of the virtual wallet application
18 using the data from the QR code, 620.

19 **[00158]** In some embodiments, the QR code may include product information,
20 commands, user navigation instructions, etc. for the virtual wallet application (see 621).
21 The virtual wallet application may query a product database using the information
22 encoded in the QR. The virtual wallet application may provide various features including,
23 without limitation, displaying product information, redirecting the user to: a product

1 page, a merchant website, a product page on a merchant website, add item(s) to a user
2 shopping cart at a merchant website, etc. In some implementations, the virtual wallet
3 application may perform a procedure such as described above for any image frame
4 pending to be processed, and/or selected for processing by the user (e.g., from the snap
5 history).

6 **[00159]** In further implementations, the above mentioned software kits, tools,
7 and/or the like may also be applied to capture textual item information on a bill/receipt
8 image.

9 **[00160]** In further implementations, when a purchase item is related healthcare
10 product/service, the RUAP may automatically submit the purchase information for
11 insurance adjudication. For example, when a user has submitted a receipt including
12 purchase of prescription drugs for FSA reimbursement, the RUAP server may generate
13 an insurance claim in addition to the FSA reimbursement, wherein the RUAP server
14 may engage in an insurance adjudication process with the insurance provider to claim
15 for an insured amount, and may then submit a request for FSA reimbursement of the
16 remaining amount. FIGURE 7A provides a data block diagram illustrating data flows
17 between healthcare payment entities (RUAP server, user and affiliated entities) showing
18 insurance adjudication within embodiments of the RUAP. Within various embodiments,
19 as shown in FIGURE 7A, one or more user(s) (patient(s)) 702, RUAP server 720, RUAP
20 database(s) 719, merchant(s) 710 (e.g., a healthcare provider, etc.), insurance provider
21 750, insurance bank 760, and/or the like are shown to interact via various
22 communication networks 713 to facilitate healthcare insurance adjudication.

1 **[00161]** Within embodiments, prior to receiving healthcare service or purchasing
 2 healthcare products, the user 702 may obtain an insurance program, e.g., by submitting
 3 registration information 703 to an insurance provider 750. For example, the user 702
 4 may fill out an insurance application form via a web based application to the insurance
 5 provider 750. An XML formatted user registration message 703 may take a form similar
 6 to the following:

```

7 POST /InsuranceApp.php HTTP/1.1
8 Host: 64.134.25.22
9 Content-Type: Application/XML
10 Content-Length: 718
11 <?XML version = "1.0" encoding = "UTF-8"?>
12 <InsuranceApp>
13     <TimeStamp> 11:11:23 </TimeStamp>
14     <Date> 09-09-2015 </Date>
15     <InsuranceStartDate> 01-01-2016 </InsuranceStartDate>
16     <InsuranceEndDate> 12-31-2016 </InsuranceEndDate>
17     <InsuranceType> Employer Sponsored </InsuranceType>
18     <ProgramCode> PPO </ProgramCode>
19     <User>
20         <UserName> John Smith </UserName>
21         <UserID> JS0000 </UserID>
22         <AccountNo> 0000 0000 0000 </AccountNO>
23         <Password> 0000 </Password>
24         <PasswordQ>
25             <Question1> Where were you born </Question1>
26             <Answer1> New York </Answer>
27             ...
28         </PasswordQ>
29         <Employer> SuperFast Software Co. </Employer>
30         <AnnualIncome> 100,000.00 </AnnualIncome>
31         ...
32     </User>
33     <Employer>
34         <ID> 092034 </ID>
35         <GroupID> 43498ABC </GroupID>
36         <Name> SuperFast Software Co. </Name>
37         <Address>
38             <Line1> ... </Line1>
39             <Line2> ... </Line2>
40             <Zipcode> 00000 </Zipcode>
41             ...
42         </Address>
43         ...
44     </Employer>
45 ...
46 </InsuranceApp>

```

47 **[00162]** In one implementation, the insurance provider 750 may underwrite an
 48 insurance policy 704 for the user, and issue an insurance device to the user 702, e.g., an
 49 insurance card, etc. For example, the insurance provider 750 may maintain an

1 insurance record of the user 702 at a database. An exemplary XML formatted user

2 insurance record 704 may take a form similar to the following:

```

3 POST /UserInsurance.php HTTP/1.1
4 Host: 64.134.25.00
5 Content-Type: Application/XML
6 Content-Length: 718
7 <?XML version = "1.0" encoding = "UTF-8"?>
8 <UserInsurance>
9     <User>
10         <UserName> John Smith </UserName>
11         <UserID> JS0000 </UserID>
12         <AccountNo> 0000 0000 0000 </AccountNO>
13         <Password> 0000 </Password>
14         <PasswordQ>
15             <Question1> Where were you born </Question1>
16             <Answer1> New York </Answer>
17             ...
18         </PasswordQ>
19         <Employer> SuperFast Software Co. </Employer>
20         <AnnualIncome> 100,000.00 </AnnualIncome>
21         ...
22     </User>
23 <BIN> 0009090fdsfdf </BIN>
24 <Insurance>
25     <InsuranceID> BB0008PPO </Insurance>
26     <InsurnaceStartDate> 01-01-2016 </InsuranceStartDate>
27     <InsuranceEndDate> 12-31-2016 </InsuranceEndDate>
28     <InsuranceType> Employer Sponsored </InsuranceType>
29     <ProgramCode> PPO </ProgramCode>
30     <GroupID> 8943243 </GroupID>
31     <InsuranceCoverage>
32         <ProcedureCode1> 60% </ProcedureCode1>
33         <ProcedureCode2> 60% </ProcedureCode2>
34         ...
35     </Insurance>
36     ...
37 </UserInsurance>
38

```

39 **[00163]** Upon establishing an insurance policy with the insurance provider 750,
40 the user 702 may submit their insurance information 712 (e.g., the insurance ID, user
41 information, etc.) to a merchant 710 upon visiting the office, purchasing a healthcare
42 product at a pharmacy (e.g., at 203 in FIGURE 2A), and/or the like. The merchant 710
43 may perform an insurance provider pre-check, e.g., checking whether the provider is an
44 in-network provider, the coverage of the insurance policy, and/or the like. Based on the

1 retrieved insurance coverage information, the merchant 710 may generate a medical bill
2 713 including a calculated insured amount and a user responsibility amount.

3 **[00164]** For example, the user 702 may receive a payment bill 715, indicating the
4 details of the treatment, and the payment amount due, including an amount of the
5 insurance coverage, and the patient's co-pay amount. In one implementation, the user
6 may receive a printed bill at the POS terminal at the hospital; may receive an electronic
7 bill in the email, instant messaging, a healthcare web portal, a mobile wallet, and/or the
8 like. The merchant 710 may pre-check the insurance information of the patient, and
9 thus make an estimate of the insured amount and user co-payment amount, which may
10 be reflected into the medical bill 715. For example, in one implementation, an
11 exemplary XML implementation of the medical bill 714 may take a form similar to:

```
12 POST /MedBill.php HTTP/1.1
13 Host: 64.134.25.33
14 Content-Type: Application/XML
15 Content-Length: 718
16 <?XML version = "1.0" encoding = "UTF-8"?>
17 <MedBill>
18     <BillID> MD 0000123 </BillID>
19     <BillDate> 09-09-2015 </BillDate>
20     <BillTimeStamp> 14:23:56 </BillTimeStamp>
21     <BillCode> 0543 </BillCode>
22     <Patient>
23         <UserID> 123456789 </UserID>
24         <UserName> John Smith </UserName>
25         </UserAddress> 111 White road </UserAddress>
26         <UserPhoneNumber> 000-000-2222 </UserPhoneNumber>
27         ...
28         </UserDeviceID> 11111111 </UserDeviceID>
29         <UserLicenseInfo> ....</UserLicenseInfo>
30         ...
31     </Patient>
32     ...
33     <Procedure>
34         <ProcedureCode> Sur-Knee-Left </ProcedureCode>
35         <ProcedureDate> 09-09-2011 </ProcedureDate>
36         <ProviderID> SPH001 </ProviderID>
37         <ProviderName> St John's Hospital </ProviderName>
38         ...
39     </Procedure>
40     <Insurance>
41         <InsuranceID> BB0008PPO </Insurance>
42         <InsuranceType> Regular </InsuranceType>
43         <InsuranceCoverage>
```

```

1           <ProcedureCode1> 60% </ProcedureCode1>
2           <ProcedureCode2> 60% </ProcedureCode2>
3           ...
4       </Insurance>
5 <BillSummary>
6     <TotalAmount> 12,000.00 </TotalAmount>
7     <Insured> 5,000.00 </Insured>
8     <PatientResp> 7,000.00 </PatientResp>
9     <AmountDue> 7,000.00 </AmountDue>
10    ...
11 </BillSummary>
12 ...
13 </MedBill>
14

```

15 **[00165]** In one implementation, the merchant may generate a HTTP POST
16 message to the RUAP server 720, seeking for medical claim 716, wherein the XML-
17 formatted message may take a form similar to:

```

18 POST /ClaimRequst.php HTTP/1.1
19 Host: www.Hospital.com
20 Content-Type: Application/XML
21 Content-Length: 718
22 <?XML version = "1.0" encoding = "UTF-8"?>
23 <ClaimRequest>
24   <RequestID> SHP-0001 </RequestID>
25   <RequestDate> 09-09-2015 </BillDate>
26   <RequestTimeStamp> 14:23:56 </RequestTimeStamp>
27   <User>
28     <UserName> John Smith </UserName>
29     <UserID> JS0000 </UserID>
30     <AccountNo> 0000 0000 0000 </AccountNO>
31     <Password> 0000 </Password>
32     <PasswordQ>
33       <Question1> Where were you born </Question1>
34       <Answer1> New York </Answer>
35       ...
36     </PasswordQ>
37     ...
38   </User>
39   <Insurance>
40     <InsuranceID> BB0008PPO </Insurance>
41     <GroupID> 123456789 </GroupID>
42     <InsuranceType> Regular </InsuranceType>
43     <InsuranceCoverage>
44       <ProcedureCode1> 60% </ProcedureCode1>
45       <ProcedureCode2> 60% </ProcedureCode2>
46     </InsuranceCoverage>
47     <BIN> 0009203920390 </BIN>
48     ...
49   </Insurance>
50   ...
51   <Time> 19:20:23 </Time>
52   <Date> 09-01-2011 </Date>
53 <Claim>
54   <Procedure>
55     <ProcedureCode> Sur-Knee-Left </ProcedureCode>
56     <ProcedureDate> 09-09-2011 </ProcedureDate>
57     <ProviderID> SPH001 </ProviderID>

```



```

1         <ProviderName> St John's Hospital </ProviderName>
2         ...
3         </Procedure>
4         <TotalAmount> 12,000.00 </TotalAmount>
5         <EstimatedInsured> 5,000.00 </EstimatedInsured>
6         <PatientResp> 7,000.00 </PatientResp>
7         ...
8     </Claim>
9     ...
10 </ClaimRequest>
11

```

12 **[00166]** In one implementation, the RUAP server 720 may obtain a BIN number
13 718 (e.g., a 16 digit code, etc.) from the received medical claim 716, based on which the
14 RUAP may determine insurance provider routing information 721. For example, the
15 RUAP server 720 may query an insurance provider database (e.g., 2419l in FIGURE 24)
16 and obtain routing destination 221 (e.g., an IP address, port address, gateway, etc.) of
17 the BIN.

18 **[00167]** In one implementation, the RUAP server 720 may generate a payment
19 authorization request 719 and route the message to the insurance provider 750 based on
20 the BIN-based routing destination. For example, the RUAP may generate a HTTPS
21 POST message to make an authorization request 719 in the form of data formatted
22 according to the XML. Below is an example HTTP(S) POST message including an XML-
23 formatted message of the authorization request 723 for the RUAP server:

```

24 POST /AuthorizationRequest.php HTTP/1.1
25 Host: www.RUAP.com
26 Content-Type: Application/XML
27 Content-Length: 718
28 <?XML version = "1.0" encoding = "UTF-8"?>
29 <AuthorizationRequest>
30 <Time> 17:40:40 </Time>
31 <Date> 09-09-2015 </Date>
32
33 <User>
34     <UserName> John Smith </UserName>
35     <UserID> JS0000 </UserID>
36     <Password> 0000 </Password>
37     <PasswordQ>
38         <Question1> Where were you born </Question1>
39         <Answer1> New York </Answer>
40     ...

```

```

1         </PasswordQ>
2         <Employer> ABC Software CO. </Employer>
3         ...
4 </User>
5 <Insurance>
6         <InsuranceID> BB0008PPO </Insurance>
7         <GroupID> 123456789 </Group>
8         <InsuranceType> Regular </InsuranceType>
9         <InsuranceCoverage>
10            <ProcedureCode1> 60% </ProcedureCode1>
11            <ProcedureCode2> 60% </ProcedureCode2>
12            ...
13        </InsuranceCoverage>
14 </Insurance>
15        ...
16 <Procedure>
17         <ProcedureCode> Sur-Knee-Left </ProcedureCode>
18         <ProcedureDate> 09-09-2014 </ProcedureDate>
19         <ProviderID> SPH001 </ProviderID>
20         <ProviderName> St John's Hospital </ProviderName>
21         ...
22 </Procedure>
23 <Claim>
24         <Amount> 5,000.00 </Amount>
25         <TotalAmount> 12000.00 </TotalAmount>
26         ...
27 </Claim>
28 ...
29 </AuthorizationRequest>
30

```

[00168] The insurance provider 750 may review and verify the requested insurance payment. For example, the insurance provider may assess the medical claim of the requested insured amount based on local annual income, economic indicators, market price, living expenses, and/or the like. In one implementation, the insurance provider 750 may send an insurance payment authorization response 736 back to the RUAP to authorize the payment. In another implementation, the insurance provider 750 may approve a payment amount which may not equate the requested amount, and subject to further adjudication and reconciliation.

[00169] Upon reviewing and approving the requested insured amount, the insurance provider 750 may provide a response 736 to the payment authorization request 719, either to approve an entirety or a part of the requested insurance payment, or to reject the payment request when the received medical claim is not eligible. For

1 example, the insurance provider 750 may verify whether the estimated insured amount
 2 in the payment request 719 matches the user's insurance coverage program, whether the
 3 user's year-to-date insurance payment has exceeded a maximum amount if there is any,
 4 whether the user's employment and/or insurance program is in good standing, and/or
 5 the like.

6 **[00170]** In one implementation, the insurance provider may generate a HTTPS
 7 POST message to make an authorization response 736 in the form of data formatted
 8 according to the XML. Below is an example HTTP(S) POST message including an XML-
 9 formatted message of the authorization response 736 for the RUAP:

```

10 POST /AuthorizationResponse.php HTTP/1.1
11 Host: www.RUAP.com
12 Content-Type: Application/XML
13 Content-Length: 718
14 <?XML version = "1.0" encoding = "UTF-8"?>
15 <AuthorizationResponse>
16   <Time> 17:42:40 </Time>
17   <Date> 09-09-2015 </Date>
18   <User>
19     <UserName> John Smith </UserName>
20     <UserID> JS0000 </UserID>
21     <Password> 0000 </Password>
22     <PasswordQ>
23       <Question1> Where were you born </Question1>
24       <Answer1> New York </Answer>
25     ...
26   </PasswordQ>
27   ...
28 </User>
29 <Insurance>
30   <InsuranceID> BB0008PPO </InsuranceID>
31   <GroupID> 123456789 </GroupID>
32   <InsuranceType> Regular </InsuranceType>
33   <InsuranceCoverage>
34     <ProcedureCode1> 60% </ProcedureCode1>
35     <ProcedureCode2> 60% </ProcedureCode2>
36   ...
37 </Insurance>
38 ...
39 <Procedure>
40   <ProcedureCode> Sur-Knee-Left </ProcedureCode>
41   <ProcedureDate> 09-09-2011 </ProcedureDate>
42   <ProviderID> SPH001 </ProviderID>
43   <ProviderName> St John's Hospital </ProviderName>
44   ...
45 </Procedure>
46 <RequestedAmount> 5,000.00 </RequestedAmount>
47 <AnnualMaximum> 6,000.00 </AnnualMaximum>

```

```
1 <YearToDate> 3,000.00 </YearToDate>
2 <Available> 3,000.00 </Available>
3 <ApprovedAmount> 3,000.00 </ApprovedAmount>
4 <Status> Good </Status>
5 ...
6 </AuthorizationResponse >
7
```

8 **[00171]** In the above example, as the user “John Smith” has an insurance policy
9 that cap the yearly insurance payment to be “\$6,000.00” and the user has already
10 received “\$3,000.00” payment year to date. Therefore, the insurance provider may
11 approve an amount capped by the remaining permitted insurance payment as
12 “\$3,000.00.”

13 **[00172]** Upon receiving the insurance payment authorization 136, the RUAP may
14 process the insurance payment 134, and may subject the payment to further
15 adjudication and reconciliation. In one implementation, the RUAP server 720 may
16 obtain an insurance provider approved amount and generate a claim adjustment request
17 737 to the merchant 710. The merchant 710 may in turn generate an adjusted user bill
18 reflecting user co-pay amount 715 for user payment (e.g., the user bill 211 in FIGURE
19 2B).

20 **[00173]** In one implementation, the RUAP may retrieve bank routing information
21 721 (e.g., the insurance bank information, etc.) and generate a fund transfer request 726
22 to transfer the approved insurance amount. For example, the RUAP may send the fund
23 transfer request 136 to a bank 760 (e.g., the insurance bank, etc.), which may take a
24 form similar to the format in compliance with electronic fund transfers (EFT), and in
25 some embodiments, it may be directly made to the merchant via a third party bank, e.g.,
26 absent the direction of the RUAP server. In another implementation, the RUAP server
27 720 may be integrated with a payment network, e.g., VisaNet, etc., which may facilitate
28 the payment processing. In one implementation, the RUAP server 720 may debit the

1 approved insurance amount from the insurance provider's account and credit to the
 2 merchant 710. For example, the RUAP server may generate a HTTPS post for money
 3 transfer, wherein the HTTPS POST message 726 may take a form similar to the
 4 following:

```

5 POST /FundTransfer.php HTTP/1.1
6 Host: www.RUAP.com
7 Content-Type: Application/XML
8 Content-Length: 718
9 <?XML version = "1.0" encoding = "UTF-8"?>
10 <FundTransfer>
11     <Time> 15:39:30 </Time>
12     <Date> 09-09-2015 </Date>
13     <Payor>
14         <ID> BL009 </ID>
15         <Name> Blue Cross </Name>
16         <IssuerID> BOA001 </IssuerID>
17         <AccountNo> 0000 0000 0000 0000 </AccountNO>
18         <Routing> 111111111 </Routing>
19         ...
20     </Payor>
21     <Payee>
22         <ID> SHP009 </ID>
23         <Name> St John's Hospital </Name>
24         <AccountNo> 00000224 </AccountNO>
25         <Routing> 1234555 </Routing>
26         ...
27     </Payee>
28     <Amount> 3000.00 </Amount>
29     ...
30 </FundTransfer>
31

```

32 **[00174]** For another example, the fund transfer message may take a form similar to
 33 the Visa Single Message System (SMS) format, Visa Original Credit Transaction (OCT)
 34 format, and/or the like. The merchant 710 may then received a fund transfer 728 from
 35 the insurance bank 760.

36 **[00175]** In one implementation, the RUAP server 720 may generate a transaction
 37 record 766 for the insurance payment in the database 719. For example, the RUAP may
 38 generate a database record. Below is an example of the transaction record 766 for the
 39 RUAP server:

```

1 POST /TransactionRecord.php HTTP/1.1
2 Host: 00.001.00.00
3 Content-Type: Application/XML
4 Content-Length: 718
5 <?XML version = "1.0" encoding = "UTF-8"?>
6 <Transaction>
7     <TransactionID> 000000 </TransactionID>
8     <TransactionDate> 09-09-2015 </TransactionDate>
9     <RequestTime> 19:30:27 </RequestTime>
10    <ReceiptTime> 19:31:56 </ReceiptTime>
11    <User>
12        <UserName> John Smith </UserName>
13        <UserID> JS0000 </UserID>
14        <AccountNo> 0000 0000 0000 </AccountNo>
15        <Password> 0000 </Password>
16        <PasswordQ>
17            <Question1> Where were you born </Question1>
18            <Answer1> New York </Answer>
19            ...
20        </PasswordQ>
21        ...
22    </User>
23    <TotalAmount> 12,000.00 </TotalAmount>
24    <Insured> 5,000.00 </Insured>
25    <PatientResp> 7,000.00 </PatientResp>
26    <ApprovedInsured> 3,000.00 </ApprovedInsured>
27    <TransferLog>
28        <Transfer1>
29            <Amount> 3,000.00 </Amount>
30            <Payor> Blue Cross </Payor>
31            <Payee> St John's Hospital </Payee>
32            <Status> Verified </Status>
33            ...
34        </Transfer1>
35        ...
36    </TransferLog>
37 ...
38 </Transaction>
39

```

40 In another implementation, the database 719 may be a relational database responsive to
 41 Structured Query Language ("SQL") commands. The RUAP server may execute a
 42 hypertext preprocessor ("PHP") script including SQL commands to query the database
 43 for user, transaction data, and/or the like. An example PHP/SQL command listing,
 44 illustrating substantive aspects of storing a transaction record 766 in the database:

```

45 <?PHP
46 header('Content-Type: text/plain');
47 mysql_connect("202.155.66.130", $DBserver, $password); // access database server
48 mysql_select("TRANSACTIONS.SQL"); // select database to append
49 mysql_query("INSERT INTO Transactions (transaction_id, transaction_date,
50 requested_time, receipt_time, user_id, user_name, user_password, account_no,
51 total_amount, insured_amount, patient_copay, approved_insured, transfer_log, payee_id,
52 payor_id, transfer_amount ...)

```

```
1 VALUES ($transaction_id$, $transaction_date$, $requested_time$, $receipt_time$,  
2 $user_id$, $user_name$, $user_password$, $account_no$, $total_amount$,  
3 $insured_amount$, $patient_copay$, $approved_insured$, $transfer_log$, $payee_id$,  
4 $payor_id$, $transfer_amount$ ...); //  
5 add data to table in database  
6 mysql_close("TRANSACTIONS.SQL"); // close connection to database  
7 ?>  
8
```

9 **[00176]** In an alternative implementation, when the user does not submit
10 insurance information to the merchant at the time of purchase transaction, but seek for
11 insurance reimbursement by submitting receipt information to the RUAP, insurance
12 adjudication may take place after the transaction. FIGURE 7B provides a data flow
13 diagram illustrating post-transaction insurance reimbursement within embodiments of
14 the RUAP. Within implementations, the user may obtain a receipt 741 from a merchant
15 710, which may take a similar form to the receipt 237 in FIGURE 2B.

16 **[00177]** In one implementation, the user may submit a reimbursement request
17 (e.g., including a snap image of the receipt, a reimbursement account indication, etc.)
18 742 to the RUAP server, e.g., see also 227 in FIGURE 2B. Upon obtaining item
19 information 743 by analyzing the receipt image and determining there is healthcare
20 related service and products, the RUAP server 720 may retrieve a BIN number 744 of
21 the user profile for insurance claim. For example, in one implementation, the user may
22 have registered the insurance plan information with the electronic wallet.

23 Within implementations, the RUAP server 720 may generate an insurance payment
24 authorization message 745, and route the message to the insurance provider 750 based
25 on the BIN number. The insurance provider 750 may in turn review the payment
26 request and generate a payment authorization response 746 including an approved
27 insurance payment amount to the RUAP server 720. For example, in one

1 implementation, the insurance payment authorization request message 746 and
2 payment authorization response 746 may take a similar form to 723/736 in FIGURE 7A,
3 respectively.

4 **[00178]** Within implementations, the RUAP server 720 may generate a fund
5 transfer request 747 to the insurance bank 760 to request the approved insurance
6 amount reimbursed to the user 748a-b, which may take a form similar to the fund
7 transfer processing messages 726-728 in FIGURE 7A. Within implementations, upon
8 insurance payment, the RUAP server 720 may proceed to determine whether the
9 insurance partially reimbursed purchase item/service is eligible for healthcare
10 restricted-use account reimbursement, e.g., FSA, HSA, etc. For example, the RUAP
11 server may reimburse the remaining payment amount with a FSA account, e.g., to
12 proceed with 421 in FIGURE 4A.

13 **[00179]** FIGURE 7C provides a data block diagram illustrating data flow between
14 healthcare payment entities (RUAP server, user and affiliated entities) for user payment
15 within embodiments of the RUAP. Within various embodiments, as shown in FIGURE
16 7B, one or more user(s) (patient(s)) 702, RUAP server 720, RUAP database(s) 719,
17 merchant(s) 710, an account issuer 770, insurance bank 780, and/or the like are shown to
18 interact via various communication networks 713 to facilitate healthcare user payment.

19 **[00180]** Continuing on with user 702 receiving a medical bill 715 from the
20 merchant 710 (e.g., as discussed in FIGURE 7A), in response to the received medical bill,
21 e.g., at the POS terminal at the healthcare provider 110, the patient 702 may submit a
22 medical payment request 717 to an acquirer, which may forward the payment request to
23 the RUAP server 720 for processing. For example, the user may submit card

1 information at the POS terminal. For another example, the user may initiate an
2 electronic wallet payment via NFC handshake (e.g., as shown in FIGURE 1A), and
3 selects a payment account via his wallet. In one implementation, the wallet account may
4 comprise any credit card account, debit account, investment account, alternative
5 payment account, loyalty points account, and/or the like. In a further implementation,
6 the user may have added restricted use accounts with the RUAP accounts, such as
7 Flexible Savings Accounts (FSA), one or more Health Savings Accounts (HSA), one or
8 more government insurance programs (i.e., Medicare or Medicaid), various private
9 insurance rules, various other restricted use accounts such as employment benefit plans
10 or employee pharmacy benefit plans, and income deduction rules, and/or the like.
11 Further implementations of restricted use account wallet enrollment are discussed in
12 FIGURE 7C.

13 **[00181]** Within embodiments, the user may select one or more accounts for
14 payment, and enter an amount to be charged with each account. For example, the user
15 may select an account FSA and enter an amount of “1,000.00” and another credit card
16 account with an entered amount of “6,000.00.”

17 **[00182]** In one implementation, the payment request 717 may comprise
18 information such as user profile information, user insurance information, user pre-
19 loaded account information, medical bill information, and/or the like. For example, in
20 one implementation, a POS terminal processing the user payment request may generate
21 a HTTPS POST message including information of the payment request 717 in the form
22 of data formatted according to the XML. Below is an example HTTP(S) POST message
23 including an XML-formatted message for the RUAP server:

```

1 POST /PaymentRequest.php HTTP/1.1
2 Host: www.Hospital.com
3 Content-Type: Application/XML
4 Content-Length: 718
5 <?XML version = "1.0" encoding = "UTF-8"?>
6 <PaymentRequest>
7     <Time> 15:30:30 </Time>
8     <Date> 09-09-2014 </Date>
9     <User>
10         <UserName> John Smith </UserName>
11         <UserWalletID> JS0000 </UserID>
12     ...
13 </User>
14 <Wallet>
15     <WalletID> HW-JS-0001 </WalletID>
16     <WalletToken> 8943243 </WalletToken>
17     <Payment1>
18         <Account> FSA </Account>
19         <AccountNo>
20             <Amount> 1000.00 </Amount>
21     </Payment>
22     <Payment2>
23         <Account> Bank Of America Visa Card </Account>
24         <AccountNo> 0000 0000 0000 0000 </AccountNo>
25         <Amount> 6000.00 </Amount>
26     ...
27 </Wallet>
28 ...
29 </PaymentRequest>
30

```

31 **[00183]** Upon receiving the payment request message 717 from the user, the RUAP
32 server 720 may retrieve wallet information 731 of the user (e.g., account no and user ID,
33 as shown in the payment request 717). For each selected account indicated in the
34 payment request 717, the RUAP server 720 may query for a routing destination 732 for
35 the account. For example, when the user selects “FSA” account, the RUAP server 720
36 may select the routing destination 732 to be the FSA administer/sponsor 770 of the user.

37 **[00184]** In one implementation, the RUAP server 720 may generate and route a
38 payment request 733 to the account issuer 770. For example, the account issuer 770
39 may comprise a restricted use account sponsor, e.g., a FSA/HSA administer, an
40 employer who sponsored employer benefit programs for its employees, a government
41 agency (e.g., Medicare, Medicaid, etc.). In one implementation, the RUAP server 720
42 may generate separate payment request messages to different routing destinations. In

1 the above example payment request 717, when the user has selected a FSA payment
 2 account and a credit card payment, the RUAP server 720 may generate a payment
 3 request message 733 routed to a FSA administering entity (270), and a payment request
 4 message to a user's issuing bank of the credit card account.

5
 6 **[00185]** Upon receiving a payment request 733, the account issuer 770 may
 7 retrieve rules to generate verification messages 734 of the payment request. For
 8 example, the verification message 734 may indicate whether the requested payment
 9 complies with account requirement, whether the requested payment amount has
 10 exceeded the maximum amount, and/or the like. For example, the account issuer entity
 11 770 may generate a HTTPS POST message including a verification message 734 in the
 12 form of data formatted according to the XML. Below is an example HTTP(S) POST
 13 message including an XML-formatted verification message 734:

```

14 POST /FSAverification.php HTTP/1.1
15 Host: www.FSA.com
16 Content-Type: Application/XML
17 Content-Length: 718
18 <?XML version = "1.0" encoding = "UTF-8"?>
19 <FSAverification>
20     <Time> 15:30:56 </Time>
21     <Date> 09-09-2015 </Date>
22     <User>
23         <UserName> John Smith </UserName>
24         <UserWalletID> JS0000 </UserID>
25     ...
26 </User>
27 <AccountStatus>
28     <AccountType> FSA </AccountType>
29     <AccountNo> 123456 </AccountNo>
30     <Amount> 1000.00 </Amount>
31     <Balance> 7000.00 </Balance>
32     ...
33 </AccountStatus>
34 <HealthcareService>
35     <ProviderID> SHP009 </ProviderID>
36     <ProcedureCode> KS-001 </ProcedureCode>
37     <Coverage> OK </Coverage>
38     ...
39 </HealthcareService>
40 <Status> Approve </Status>
41 ...
42 </Verification>

```

1 [00186]

2 [00187] In the above example, the FSA administer program 770 verifies that the
3 user's healthcare service code is eligible for FSA reimbursement, and does not exceed
4 the available balance in the account. As such, the FSA administer program 770 may
5 approve the payment, and generate a fund transfer request 735 to the sponsor bank 780.
6 For example, the fund transfer message may take a form similar to message 726 in
7 FIGURE 7B, which may trigger a fund transfer 737 from the FSA account to the
8 healthcare provider 710.

9 [00188] In one implementation, the account issuer 770 may verify the payment in
10 real time, or a nearly real time manner. In other implementations, the account issuer
11 770 may receive and process payment requests 733 in batch files. In further
12 implementations, the RUAP server 720 may perform an eligibility pre-check if the user
13 submitted account comprises a healthcare restricted-use account (e.g., FSA, HSA, etc)
14 and various rules may be applied to the payment request based on the type of the
15 account.

16 [00189] In one implementation, the RUAP server 720 may generate a transaction
17 record 766b for the insurance payment in the database 719. For example, below is an
18 example XML-formatted message of the transaction record 738 for the RUAP server:

```
19 POST /TransactionRecord.php HTTP/1.1
20 Host: 00.001.00.00
21 Content-Type: Application/XML
22 Content-Length: 718
23 <?XML version = "1.0" encoding = "UTF-8"?>
24 <Transaction>
25     <TransactionID> 000002 </TransactionID>
26     <TransactionDate> 09-09-2015 </TransactionDate>
27     <RequestTime> 19:30:27 </RequestTime>
28     <ReceiptTime> 19:48:56 </ReceiptTime>
29     <User>
30         <UserName> John Smith </UserName>
31         <UserID> JS0000 </UserID>
```

```

1      <Password> 0000 </Password>
2      <PasswordQ>
3          <Question1> Where were you born </Question1>
4          <Answer1> New York </Answer>
5          ...
6      </PasswordQ>
7      ...
8  </User>
9      <PaymentType> FSA </PaymentType>
10     <Status> Approved </Status>
11     <TransferLog>
12         <Transfer1>
13             <Amount> 1,000.00 </Amount>
14             <Payor> FSA administration </Payor>
15             <PayorAccount> 123456 </PayorAccount>
16             <Payee> St John's Hospital </Payee>
17             <Status> Verified </Status>
18             ...
19         </Transfer1>
20         ...
21     </TransferLog>
22 ...
23 </Transaction>
24 [00190]

```

25 In another implementation, the RUAP server may execute a hypertext preprocessor
 26 (“PHP”) script including SQL commands to query the database for user, transaction
 27 data, and/or the like. An example PHP/SQL command listing, illustrating substantive
 28 aspects of storing a transaction record 766 in the database:

```

29 <?PHP
30 header('Content-Type: text/plain');
31 mysql_connect("202.155.66.130",$DBserver,$password); // access database server
32 mysql_select("TRANSACTIONS.SQL"); // select database to append
33 mysql_query("INSERT INTO Transactions (transaction_id, transaction_date,
34 requested_time, receipt_time, user_id, user_name, user_password, account_no,
35 total_amount, account_type, patient_copay, approved_insured, transfer_log, payee_id,
36 payor_id, transfer_amount ...);
37 VALUES ($transaction_id$, $transaction_date$, $requested_time$, $receipt_time$,
38 $user_id$, $user_name$, $user_password$, $account_no$, $total_amount$,
39 $insured_amount$, $account_type$, $approved_insured$, $transfer_log$, $payee_id$,
40 $payor_id$, $transfer_amount$ ...); //
41 add data to table in database
42 mysql_close("TRANSACTIONS.SQL"); // close connection to database
43 ?>

```

44 **[00191]** FIGURES 8A-8E provide logic flow diagrams illustrating healthcare
 45 insurance adjudication and healthcare restricted-use account (e.g., FSA, HSA, etc.)
 46 payment within embodiments of the RUAP. As shown in FIGURE 8A, within

1 implementations, the user 802 may register to the RUAP 820 prior to utilizing the
2 RUAP payment service after healthcare treatment.

3 **[00192]** In one embodiment, the user 802 may submit a request 805 for
4 registration with the RUAP, e.g., via an email, a text message, a telephonic phone call to
5 customer service, and/or the like. The RUAP may then provide a RUAP mobile
6 component 806 to the user. For example, the RUAP may provide an indication, a link,
7 etc. for downloading a mobile payment application to the user's mobile device, via which
8 the user may register one or more multi-purpose accounts with the RUAP and process
9 healthcare claims and payments in real-time.

10 **[00193]** In one embodiment, the user 802 may download and install the RUAP
11 component on a mobile device 807, e.g., an Apple iPhone, etc. In further
12 implementation, the RUAP may comprise a web portal, feature sets in a cloud,
13 downloaded indicia from a cloud, and/or the like.

14 **[00194]** The user 802 may then register with the RUAP via the installed RUAP
15 component, by submitting healthcare insurance information and setting up payment
16 accounts 808. For example, in one implementation, the user 802 may enroll restricted
17 use accounts into their wallet for healthcare user payment. The restricted use rules may
18 include those for one or more HSA/FSA, one or more government insurance programs
19 (i.e., Medicare or Medicaid), various private insurance restricted use rules, various other
20 restricted use accounts such as employment benefit plans or employee pharmacy benefit
21 plans, and income deduction rules.

22 **[00195]** For example, the user may associate his FSA/HSA accounts with the
23 RUAP. For another example, the user may be presented rule choices within agreement

1 and IRS policies, and set up their rules and parameters for usage of their FSA/HSA
2 payment accounts. For example, the user may set up a rule such that any medical
3 purchase less than \$100.00 until the end of the year will be paid by his FSA account.

4 **[00196]** In one embodiment, the RUAP may provide default settings and rules for
5 the user via a user interface, e.g., the mobile component installed on the user's mobile
6 device. In another embodiment, the user may configure a variety of parameters. In the
7 above example, the user may edit the balancing amount of an account, the end date, the
8 rules, and/or the like.

9 **[00197]** In one embodiment, upon receiving user provided registration data and
10 related parameters and spending rules, the RUAP may validate the insurance
11 information with the insurance provider 150, and set up spending rules associated with
12 the user's RUAP account 809 to complete the registration. In another implementation,
13 the RUAP 120 may register the user's mobile device for security, such as, via a hardware
14 ID, MAC address, and/or the like.

15 **[00198]** In one embodiment, after the user is present at a healthcare provider for
16 medical services, the healthcare provider 810 may submit medical claim information 811
17 to an insurance provider 850 at checkout, wherein the insurance provider may approve
18 an insured portion 812 based on the user's insurance policy. In one implementation, the
19 user may send a payment file (e.g., via text message, email, etc.) to the RUAP, including
20 the amount of patient responsibility, NPI, plan membership, SSN, etc. The RUAP may
21 then verify the submitted user data with verify against the healthcare registration record.
22 If the record matches, the RUAP may generate a "please pay an amount \$100.00"
23 message and send to the user.

1 **[00199]** In one implementation, the healthcare provider 810 may send the
2 remaining balance of the medical bill to the RUAP for user payment processing. In
3 another implementation, the user 802 may receive a medical bill, e.g., at a mobile device,
4 etc., in real-time at checkout, and enter the amount due 814 into his mobile device for
5 RUAP.

6 **[00200]** In one implementation, the RUAP 820 may determine a recommended
7 payment plan 815 to the user based on the remaining amount in the user's registered
8 payment accounts with RUAP (e.g., based on the transaction nature, user's GPS location,
9 etc.), and prompt the user to select a payment method 816. Upon receiving a
10 confirmation of payment selection, the RUAP may process payment with the healthcare
11 accounts 817, and the healthcare provider may send confirmation of payment 818.

12 **[00201]** FIGURES 8B-8C provides a logic flow diagram illustrating healthcare
13 insurance adjudication within embodiments of the RUAP. Within embodiment, prior to
14 receiving healthcare treatment, a user 802 may submit insurance registration request
15 821, e.g., to purchase an insurance program, etc. The insurance provider 850 may
16 underwrite the insurance policy for the user 823, and store the information (e.g., see
17 704 in FIGURE 7A) in a database, and send the insurance information 824 to the user,
18 e.g., an insurance card, an insurance electronic entry in the user's electronic wallet,
19 and/or the like.

20 **[00202]** In one embodiment, upon receiving medical services, the user 802 may
21 submit insurance information 826 to healthcare provider 810, e.g., by presenting his
22 insurance card to a representative at a checkout desk. The healthcare provider 810 may
23 pre-check the insurance eligibility 828, such as whether the healthcare provider is in

1 network, whether the insurance term has expired, and/or the like. If it is not eligible
2 (e.g., expired insurance term, healthcare provider out-of-network, etc.), the user may
3 receive an insurance ineligibility notice 831. Otherwise, the healthcare provider 810
4 may generate a medical bill 830, including an estimated insured amount. For example,
5 if the insurance provider has a standard coverage list for a procedure, e.g., 40% for a
6 root canal therapy at an in-network dental provider, etc., the healthcare provider may
7 calculate an estimated insured amount by multiplying the total amount with 40%.

8 **[00203]** Alternatively, the healthcare provider 810 may generate a medical claim
9 832 to the insurance provider for adjudication 833a, e.g., to provide an approved
10 insurance payment amount. In one implementation, the medical claim may be sent to
11 the RUAP server 833b, which may generate an insurance authorization message 834
12 (e.g., see 736 in FIGURE 7A).

13 **[00204]** Continuing on with FIGURE 8C, the RUAP may retrieve a BIN number
14 from the medical claim and determine a routing destination based on the BIN number;
15 and forward the authorization request to the insurance provider 835. Upon receiving
16 the authorization/adjudication request, the insurance provider 850 may parse the
17 request to extract procedure and pricing information 837, and determine an approved
18 insurance amount 838. For example, the insurance provider may assess the procedure,
19 and determine whether the healthcare provider provided pricing is reasonable based on
20 a variety of factors, such as, but not limited to local living expenses, market price,
21 economic indicator, pricing information of peer healthcare providers in the same area,
22 average income, and/or the like.

1 **[00205]** In one implementation, the insurance provider may further verify whether
2 the user's insurance account is in good standing 839. For example, if it is an employer
3 benefit account, the insurance provider may verify the user's employment with the
4 sponsor (e.g., the employer) is in good standing.

5 **[00206]** In one implementation, if the insurance account is no longer eligible for
6 the user, the RUAP may receive a payment denial message and be prompted to re-
7 submit insurance information 840. The RUAP may then provide the denial message to
8 the user 843, who may elect to re-submit an insurance verification request 844.
9 Alternatively, the healthcare provider may be notified of the ineligibility of the insurance,
10 and may adjust the medical bill 841.

11 **[00207]** Upon verification at 839, the insurance provider 850 may compare the
12 claimed amount (e.g., the insured amount field in the medical claim message 716 in
13 FIGURE 7A) with the insurance assessed amount (e.g., at 838) 842. If the approved
14 amount meets with the claimed amount, the insurance provider 850 may authorize a
15 transaction of the medical claim 843 (and withdraw the difference if the insurance
16 approved amount is greater than the claimed 846), and the healthcare provider may
17 received the claimed amount 855. Otherwise, if the insurance assessed amount is less
18 than the requested claim, the insurance provider may re-assess the claim to determine
19 whether to approve the difference 848, e.g., to start a re-adjudication process 841.

20 **[00208]** FIGURES 8D-8E provide logic flow diagrams illustrating healthcare user
21 payment within embodiments of the RUAP. Within embodiments, users may obtain a
22 medical bill which specifies the user responsible portion after the insurance adjudication.
23 Upon receiving a medical bill 851 from a healthcare provider 810, the user 802 may

1 submit a payment request 853, e.g., by swiping a prepaid card at the healthcare provider
2 checkout registry, by operate a mobile wallet, and/or the like. In one implementation,
3 the healthcare provider may determine whether the user submitted payment account is
4 a restricted use account (e.g., a FSA, HSA, LOC, etc.), and/or a sponsor administered
5 account (e.g., Medicare, Medicaid, employment benefit account, etc.) 854. If so, the
6 healthcare provider may perform a pre-check 855 to determine whether it is applicable
7 based on the purchased procedure/product code. For example, a user may not engage
8 his FSA account to pay for cosmetic products, as the product code is not in a FSA eligible
9 category. If not eligible, the transaction may be denied 858 at the healthcare provider.

10 **[00209]** If eligible, the RUAP may receive the payment request 857 including user's
11 account information (e.g., via a healthcare card, an electronic wallet, etc.). The RUAP
12 may retrieve the user's wallet/card information and a routing destination 855, and
13 generate a payment request (e.g., 733 in FIGURE 7B) for the routing destination 859. If
14 the user submitted payment account is not a restricted use account 860, e.g., a bank
15 account, etc., the RUAP may proceed with financial card transaction, e.g., as further
16 discussed in FIGURES 17A-21.

17 **[00210]** If it is a restricted use account, the RUAP may send the payment request to
18 the account sponsor 860, e.g., a FSA/HSA/LOC account manager, Medicare/Medicaid
19 management, employer benefit account manager, and/or the like. The account sponsor
20 870 may verify eligibility of the purchased product/service 863, and verify whether
21 there is sufficient balance in the user's account for the requested payment amount 863.

22 **[00211]** Continuing on with FIGURE 8E, if there is sufficient funds 865, the
23 account sponsor 870 may approve the transaction 866, and generate a fund transfer

1 message for an issuer bank 867, which may be processed in a similar manner as
2 discussed in FIUGRES 20A-23B. The approved amount may be deducted from the user
3 account 869 and the healthcare provider may receive payment 868. Alternatively, if
4 there are insufficient funds in the account, the account manager may elect to approve a
5 payment of the available amount in the account 869.

6 **[00212]** Upon the transaction, the account sponsor 870 may generate a notification
7 of the remaining balance 871 and send it to the user 872. In one implementation, the
8 balance updates may be performed periodically (e.g., weekly, bi-weekly, etc.), and/or on
9 an on-demand basis.

10 **[00213]** FIGURES 9A-9B provide example flows illustrating user restricted-use
11 account enrollment in an electronic wallet within embodiments of the RUAP. In one
12 implementation, a user may download and install a RUAP mobile wallet component on
13 a smart phone or other portable web-enabled computing device. Integration of the
14 electronic wallet reduces the number of network transactions and messages that fulfill
15 healthcare payment approval and procurement of healthcare product and services. In
16 this way, with the reduction of network communications, the number of transactions
17 that may be processed per day is increased, i.e., processing efficiency is improved.

18 **[00214]** Within implementations, the mobile wallet application may be used by a
19 user who is presented with a request to pay for healthcare service charges. When so
20 used by the user, the mobile wallet component makes a recommendation of which the
21 user's account to offers the greatest advantage to the user when used to pay for
22 healthcare service charges. The mobile wallet component provides a real time decision
23 tool for the user to choose one or more healthcare accounts from which to withdraw

1 currency or other funds in order to pay for a healthcare service. To assist the user in
2 making the right choice, the mobile wallet component is programmed to access local
3 restricted use and regulatory business rules for healthcare services. The mobile wallet
4 component is programmed to access: (i) user-specific data and (ii) balances held by the
5 user in multiple payment accounts issued to the user who is financially responsible for
6 the healthcare service charges. The mobile wallet component is further programmed to
7 apply the restricted use and regulatory business rules to the online data (i.e., user-
8 specific data and balances) in order to make a recommendation to the user as to which
9 of the user's payment accounts be used in paying for the healthcare service charges. The
10 user may adopt, or over ride, the mobile wallet component's recommendations.
11 Thereafter, the user's smart phone may then be used at a healthcare service providers
12 POS to make contactless payments from each of the user's account as were
13 recommended by the mobile wallet component.

14 **[00215]** In one implementation, the mobile wallet component may have online
15 access to various information for processing against the restricted use and regulatory
16 business rules. For instance, local negative wealth impacting rules may provide various
17 incentives and penalties as are applicable to: (a) a FSA; (b) a HSA; (c) Government
18 Insurance (e.g.; Medicare); (d) Private Insurance; (e) Other Restricted use Accounts
19 (e.g.; employee benefits plans); (f) Income deduction rules; (g) etc.

20 **[00216]** In one implementation, the mobile wallet component may have online
21 access to various information for processing against insurance payment coverage rules.
22 For instance, insurance payment coverage rules may provide various incentives and
23 penalties as are applicable to: (a) a portion of the healthcare provided by the healthcare

1 provider to the user that are and are not covered and thus will and will not be paid for
2 via insurance coverage; (b) specific spending limit rules for the healthcare provider and
3 health provided by same; (c) annual and life-time limits for spending; (i) by-the person;
4 and (ii) by-the insurance policy; (d) limitations on the type and quantity of healthcare
5 goods and/or services type, including but not limited to: (i) pharmacy; (ii) vision; (iii)
6 dental, (iv) psychological; (v) substance abuse rehabilitation; (vi) etc.; (e) limitation on
7 payments payment to a certain type of merchant, including but not limited to: (i) 'big-
8 box' retailer; (ii) pharmacy retailer; (iii) physician sale of goods and services; (iv) etc.; (f)
9 co-payments by insurance vehicle type; (g) etc.

10 **[00217]** In one implementation, the mobile wallet component may have online
11 access to currency balances available for use, and respective calendar dates of
12 availability, to pay the balance due bill for the healthcare provided by the healthcare
13 provider. For instance, these accounts may include: (a) a Flexible Savings Account
14 (FSA), where data retrieved may include a current currency balance, a date by which all
15 funds in FSA must be spent; (b) a Health Savings Account (HSA) , where data retrieved
16 may include a liquid asset balance and a non-liquid asset balance (e.g.; stocks, mutual
17 funds, Certificates of Deposit, etc.), and an amount charged for a commission to trade
18 an illiquid asset for a liquid asset that may be used to pay the balance due bill from the
19 healthcare provider.; (c) a remaining deductible contribution amount to a healthcare-
20 relates account amount for a specific year; (d) a government insurance prepaid account;
21 (e) a private insurance deductible information; (e) other restricted use accounts (e.g.;
22 employee benefits plans); (f) non-restricted use payment accounts (e.g.; credit, debit,
23 prepaid accounts) including information for these accounts such as loyalty points and
24 awards having currency equivalents that may be made available for payment against the

1 balance due bill and award thresholds for such loyalty points and awards. The mobile
2 wallet component may also have online access to a prediction as to the likely income and
3 local financial bracket of the user for year in which the healthcare provider's balance
4 billing is due.

5 **[00218]** In still another implementation, a healthcare provider provides healthcare
6 services (e.g., medical treatment, etc.) and/or products (e.g., prescription drugs, etc.) to
7 a user. One or more insurance carriers are queried by the healthcare provider to obtain
8 payment for the healthcare provided by the healthcare provider to the user. When the
9 insurance carriers have not paid, or are unlikely to pay, for the healthcare, then the
10 healthcare provider calculates a balance due bill for which the user is financially
11 responsible. A PoS transmits the balance due bill to the user's smart phone. The smart
12 phone executes a mobile wallet component to perform real time online access to various
13 databases. This real time access obtains business rules and user data sufficient for the
14 mobile wallet component to derive a recommendation as to which the user's accounts
15 will provide the greatest advantage to the user when used to pay for healthcare service
16 charges, both goods and services, of the balance due bill. The user's smart phone may
17 then send a transmission to the healthcare provider's POS as a contactless payment
18 from each of the user's recommended accounts. For each account in the
19 recommendation: (i) the healthcare provider's POS sends the user's proposed payment
20 amount to an acquirer for the healthcare provider; (ii) the acquirer sends an
21 authorization request for the amount to a transaction handler (i.e., VisaNet) who sends
22 the authorization request to the corresponding issuer of the user's account. Note that, in
23 addition to the VisaNet network provided by Visa Inc., other networks (such as Discover
24 and American Express) may be used to accept healthcare service payment transactions.

1 Moreover, other payment options may also be made, such as ACH or online bill pay,
2 each of which could be facilitated by either the foregoing implementations or by being
3 routed to another payment portal; and (iii) the issuer sends an authorization response
4 back to the transaction handler who sends the authorization response back to the
5 healthcare provider's acquirer. Assuming that the healthcare provider's payment is
6 authorized by the issuer, the smart phone receives an electronic acknowledgement of
7 payment from each of the issuers 8 for each of the accounts. Clearing and settlement
8 will then follow for each account selected by the user to pay the healthcare provider.

9 **[00219]** In the foregoing implementation, the derivation of the recommendation
10 may rely on an application of mathematical (quantitative) techniques to make a
11 healthcare payment decision recommendation. To do so, the user's financial and
12 insurance penalties and incentives are defined and modeled as a set of mathematical
13 equations. Data that is also made available for the derivation are currency balances, and
14 dates as to availability of same, in one or more accounts to which the user has access for
15 payment of the balance due bill. The equations are subjected to computer analysis to
16 yield an optimized solution as to those user's accounts that will provide the greatest
17 advantage to the user when used to pay for healthcare service charges, both goods and
18 services, as defined in the balance due bill from the healthcare provider. Optimizing the
19 solution may requires one or more iterations to test against various circumstances and
20 situations until the optimized solution is found for making the recommendation. The set
21 of mathematical equations may apply any of several different approaches, including but
22 not limited to dynamic and linear programming, as well as forecasting and simulation
23 techniques such as the Monte Carlo method.

1 **[00220]** FIGURE 9A provides a data block diagram illustrating data flow between
2 healthcare payment entities (RUAP server, user and affiliated entities) for RUAP
3 account management within embodiments of the RUAP. Within various embodiments,
4 as shown in FIGURE 9A, one or more user(s) (patient(s)) 902, RUAP server 920, RUAP
5 database(s) 919, merchant(s) 910, a account issuer 970, and/or the like are shown to
6 interact via various communication networks 913 to facilitate RUAP account
7 management.

8 **[00221]** Within embodiments, the RUAP server 920, or a account issuer 970 may
9 act as a BIN sponsor. For example, the user 902 may submit healthcare benefit
10 program information 942 to the RUAP server 920 in order to add a restricted-use
11 account (e.g., FSA/HSA, LOC, Medicare, Medicaid, employee benefit program, food
12 stamp, etc.) to the electronic wallet. For example, in one implementation, the user may
13 fill in an online application form, may call up a wallet management agent, may send a
14 request via instant messages, emails, and/or the like. In another implementation, the
15 user may be provided the option to register with RUAP service when the user enrolls in
16 a healthcare benefit program. For example, an XML-formatted user registration request
17 including user information 942 may take a form similar to the following:

```
18 POST /RegistrationRequest.php HTTP/1.1
19 Host: 64.255.64.00
20 Content-Type: Application/XML
21 Content-Length: 718
22 <?XML version = "1.0" encoding = "UTF-8"?>
23 <RegistrationRequest[is this the same as Sponsor Program Info? That's what the figure
24 shows it as... I think one or the other is off?]>
25 <Time> 17:42:40 </Time>
26 <Date> 09-01-2014 </Date>
27 <RequestType> Add Account </RequestType>
28 <RequestID> JS09012011 </RequestID>
29 <User>
30     <UserName> John Smith </UserName>
31     <WalletID> JS0000 </UserID>
32     <Password> 0000 </Password>
33     <PasswordQ>
```

```

1           <Question1> Where were you born </Question1>
2           <Answer1> New York </Answer1>
3           ...
4       </PasswordQ>
5       <Phone>
6           <Cell> 000-000-0000 </Cell>
7           <Day> 111-111-1111 </Day>
8       ...
9       </Phone>
10      <Address>
11          <Line1> 122 Apple Ave </Line1>
12          <City> Big City </City>
13          <State> CA </State>
14          <ZipCode> 99920 </Zipcode>
15      </Address>
16      ...
17 </User>
18 <Account>
19     <AccountType> FSA </AccountType>
20     <AccountNo> 123456 </AccountNo>
21     <BIN> FSA-00133 </BIN>
22     ...
23 </Account>
24     ...
25 </RegistrationRequest>
26 [00222]

```

27 **[00223]** In one implementation, the RUAP may provide virtual prepaid card
 28 including a card number without sending physical magnetic cards, e.g., an electronic
 29 mobile wallet entry 951 for the user to download and instantiate on his mobile wallet
 30 (e.g., see electronic wallet UIs in FIGURES 16A-16E). For example, in further
 31 implementations, an additional wallet may be created for general spends.

32 **[00224]** In further implementations, funds on the additional healthcare wallet
 33 account may be separately loaded by the user. For example, fund loading to a FSA/HSA
 34 account may be performed by the user setting aside a portion of his income. In another
 35 implementation, the user may select a “back-up” account (e.g., a credit card account, a
 36 debit account, etc.) as a default account for user co-pay if payment request via the
 37 selected healthcare benefit account is denied by the account issuer 970, e.g., an
 38 ineligible healthcare service, etc.

1 **[00225]** In one implementation, the RUAP server 920 may retrieve the user's
 2 wallet information 943, and determine a routing destination 944 of the added account
 3 from the healthcare benefit program information 942, to generate a verification request
 4 946 to the account issuer entity 970. For example, the verification request 946 may
 5 comprise information fields similar to that of the user submitted restricted-use account
 6 information 942. Below is an example HTTP(S) POST message including an XML-
 7 formatted message of the account access/verification request message 946:

```

8 POST /AccessRequest.php HTTP/1.1
9 Host: www. RUAP.com
10 Content-Type: Application/XML
11 Content-Length: 718
12 <?XML version = "1.0" encoding = "UTF-8"?>
13 <AccessRequest>
14     <Time> 17:42:52 </Time>
15     <Date> 09-09-2014 </Date>
16     <RegistrationID> JS09012011 </RegistrationID>
17     <Account>
18         <AccountType> FSA </AccountType>
19         <AccountNo> 123456 </AccountNo>
20         <BIN> FSA-00133 </BIN>
21         ...
22     </Account>
23     ...
24     <User>
25         <UserName> John Smith </UserName>
26         <UserID> JS0000 </UserID>
27         <AccountNo> 0000 0000 0000 </AccountNo>
28         <Password> 0000 </Password>
29         ...
30         <Question1> Where were you born </Question1>
31         <Answer1> New York </Answer1>
32     </Password>
33     ...
34 </user>
35 ...
36 </AccessRequest>
37

```

38 **[00226]** Within implementations, the account issuer entity 970 may verify the
 39 credentials and authorize the access request from RUAP. For example, the account
 40 issuer 970 may determine whether user credentials, confirmation, etc. are received to
 41 indicate authorization from account owner, whether the benefit sponsor allows the
 42 access, etc. In one implementation, the account issuer 970 may provisionally make a

1 small amount deposit into the account that RUAP attempts to link to, e.g., \$0.65, etc.,
2 and request the user to enter the numeric value of the deposit to prove authorization.
3 For example, the user may receive confirmation request via email, instant messaging,
4 phone calls, text messages, wallet notices, and/or the like, to provide the deposited
5 numeric value. For another example, the account issuer 970 may contact a healthcare
6 benefit program sponsor (e.g., a government agency representative, an employer, etc.)
7 to verify the account access request 946.

8 **[00227]** In one implementation, the healthcare sponsor entity 970 may verify the
9 status 947 of the restricted-use account, and may send a status including the currently
10 available balance 948 to the RUAP server. For example, the account issuer entity 970
11 may generate a HTTPS POST message including an XML-formatted status message 948
12 may take a form similar to the following:

```
13 POST /FSAsstatus.php HTTP/1.1
14 Host: 64.255.64.00
15 Content-Type: Application/XML
16 Content-Length: 718
17 <?XML version = "1.0" encoding = "UTF-8"?>
18 <FSAsstatus>
19   <Time> 17:45:40 </Time>
20   <Date> 09-01-2014 </Date>
21   <User>
22     <UserName> John Smith </UserName>
23     <WalletID> JS0000 </UserID>
24     <Password> 0000 </Password>
25     ...
26   </User>
27   <Account>
28     <AccountType> FSA </AccountType>
29     <AccountNo> 123456 </AccountNo>
30     <BIN> FSA-00133 </BIN>
31     <Token> %^&SDSADFWF </Token>
32     <Balance> 3000.00 </Balance>
33     <LastUpdate> 17:45:00 </LastUpdate>
34     ...
35   </Account>
36   <Rule>
37     <Maximum> 3000.00 </Maximum>
38     <ClearancePeriod> 24 hrs </ClearancePeriod>
39     <Term>
40       <StartDate> 01-01-2015 </StartDate>
41       <EndDate> 12-31-2015 </EndDate>
42       ...
```

```

1         </Term>
2         <ProcedureList>
3             <Code1> 000 </Code1>
4             <Code2> ... </Code2>
5         </ProcedureList>
6 ...
7 </Rule>
8 ...
9 </FSAStatus>
10

```

11 **[00228]** Within implementations, the RUAP server 920 may constantly,
 12 periodically, and/or intermittently send inquiries to the healthcare benefit sponsor
 13 entity 970 for available balance updates.

14 **[00229]** In one implementation, upon verifying with the account issuer entity 970,
 15 the RUAP server 920 may generate an additional entry 949 on the user's electronic
 16 wallet 943, wherein the entry may comprise the added account information, user
 17 verification information, restricted-use account rules, and/or the like that may prompt
 18 the user to provide additional payment method into the electronic wallet. In one
 19 implementation, the RUAP mobile wallet entry 949 including the payment account
 20 entry, may take a form similar to the following XML-formatted data message:

```

21 < RUAPentry>
22     <UserName> John Smith </UserName>
23     <UserID> JS0000 </UserID>
24     <UserContactNo> 000 000 0000 </UserContactNo>
25     <Password> 0000 </Password>
26     <PasswordQ>
27         <Question1> Where were you born </Question1>
28         <Answer1> New York </Answer>
29         ...
30     </PasswordQ>
31     <Account>
32         <AccountType> FSA </AccountType>
33         <AccountNo> 123456 </AccountNo>
34         <BIN> FSA-00133 </BIN>
35         <Token> %^&SDSADFWF </Token>
36         <Balance> 3000.00 </Balance>
37         <LastUpdate> 17:45:00 </LastUpdate>
38         ...
39     </Account>
40     <Rule>
41         <Maximum> 3000.00 </Maximum>
42         <ClearancePeriod> 24 hrs </ClearancePeriod>
43         <Term>
44             <StartDate> 01-01-2015 </StartDate>
45             <EndDate> 12-31-2015 </EndDate>

```

```

1          ...
2          </Term>
3          <ProcedureList>
4          <Code1> 000 </Code1>
5          <Code2> ... </Code2>
6          </ProcedureList>
7          ...
8          </Rule>
9          <Certificate>
10             <UserToken> fdsjreiorrrgr8t9340548 </UserToken>
11             </DigitalCertificate> rfsfsuifuisduifu </DigitalCertificate>
12             <Hash> 00000 </Hash>
13             ...
14          </Certificate>
15          ...
16 </ RUAPentry>
17

```

18 **[00230]** In further implementation, the new wallet account entry 951 may be
19 provided to the user wallet, e.g., the user may view a newly added “FSA” account into his
20 wallet, and the account record 952 may be saved at the database 919. For example, tThe
21 RUAP server may execute a hypertext preprocessor (“PHP”) script including SQL
22 commands to query the database for wallet account data, and/or the like. An example
23 PHP/SQL command listing, illustrating substantive aspects of storing a wallet account
24 entry 952 in the database:

```

25 <?PHP
26 header('Content-Type: text/plain');
27 mysql_connect("202.155.66.130", $DBserver, $password); // access database server
28 mysql_select("WALLET_ENTRY.SQL"); // select database to append
29 mysql_query("INSERT INTO Wallet_Entry (user_id, user_name, user_password, user_contact,
30 user_passwordQ, account_type, account_no, account_BIN, account_token, account_balance,
31 Account_lastupdate, rule_maximum, clear_period, start_date, end_date,..., certificate ...)
32 VALUES ($user_id$, $user_name$, $user_password$, $user_contact$, $user_passwordQ$,
33 $account_type$, $account_no$, $account_BIN$, $account_token$, $account_balance$,
34 $Account_lastupdate$, $rule_maximum$, $clear_period$, $start_date$, $end_date$,...,
35 $certificate$ ...) //
36 add data to table in database
37 mysql_close("Wallet_Entry.SQL"); // close connection to database
38 ?>

```

39 **[00231]** In one implementation, the associated applicability rules 954 may be
40 provided to a list of healthcare providers 910 for pre-check purposes. For example, the
41 RUAP server 920 may provide the applicability rule to healthcare providers within a
42 range of zip codes based on the user’s location, and/or the like. For example, the RUAP

1 server may generate a HTTPS POST message including an XML-formatted applicability
 2 rules message 954, which may take a form similar to the following:

```

3 POST /Rules.php HTTP/1.1
4 Host: 64.255.64.00
5 Content-Type: Application/XML
6 Content-Length: 718
7 <?XML version = "1.0" encoding = "UTF-8"?>
8 <Rules>
9 <Time> 17:45:55 </Time>
10 <Date> 09-05-2014 </Date>
11 <AccountType> FSA </AccountType>
12 <AccountSponsor> FSA-PPA </AccountSponsor>
13 <ApplicableMerchant>
14     <MerchantCode1> MC0001 </MerchantCode1>
15     <MerchantCode2> ... </MerchantCode2>
16     ...
17 </ApplicableMerchant>
18 <ApplicableProducts>
19     <ProductCode1> PA001 </ProductCode1>
20     <ProductCode2> ... </ProductCode2>
21     ...
22 </ApplicableProducts>
23 <Procedures>
24     <ProcedureCode1> KH0938 </ProcedureCode1>
25     ...
26 </Procedures>
27 ...
28 </Rule>

```

29 **[00232]** In the above example, the RUAP may provide a list of applicable
 30 healthcare providers, products, procedures and/or the like, which are applicable for FSA
 31 account usage, to a healthcare provider. In further implementations, a user may
 32 configure user submitted rules for account usage, as further discussed in FIGURE 9B,
 33 9D and 9E.

34 **[00233]** FIGURE 9B provides a logic flow diagram illustrating RUAP restricted use
 35 account enrollment within embodiments of the RUAP. Within embodiments, a user 902
 36 may submit a healthcare sponsor account information 905 (e.g., a FSA/HSA/LOC
 37 account number, Medicare/Medicaid insurance ID, and/or the like), and wallet
 38 information. The RUAP 920 may retrieve user wallet information 908, and determine a
 39 type of the account (e.g., FSA/HSA, food stamp, Medicare/Medicaid, unemployment
 40 benefit, etc.) 911. The RUAP may retrieve restricted use regulation rules 914, and

1 determine whether it is qualified for enrollment with the wallet 916, e.g., whether the
2 regulatory policy permits the enrollment. If not, the user may receive a denial notice
3 928. If yes, the RUAP may route the enrollment request to the account issuer 922 for
4 verification 922.

5 **[00234]** In one implementation, the account issuer 970 may verify the account
6 credentials 925, user profile, account status 927, and/or the like. If the account is in
7 good standing 929, the account issuer may generate and send a token for account access
8 931 to the RUAP, and the most recent balance information and account rules 933 to the
9 RUAP. For example, in one implementation, the account rules may include a list of
10 procedure/product code and/or merchant category code (MCC) applicable for the
11 account usage.

12 **[00235]** In one implementation, the RUAP may store the security token and add a
13 wallet entry 930 to the wallet "my account" list (e.g., 1670 in FIGURE 16D), and the
14 enrolled account is ready to use with wallet payment.

15 **[00236]** FIGURES 10A-10C provide exemplary diagrams illustrating RUAP
16 applications in healthcare related purchases/products within embodiments of the RUAP.
17 FIGURE 10A provides a logic flow diagram illustrating processing healthcare payment
18 within embodiments of the RUAP. In one embodiment, the payment being made by the
19 user is optimized for the user's benefit with respect to considerations of insurance,
20 governmental taxation, and user data so that an optimized payment scheme to be made
21 to satisfy a bill from the healthcare provider for the healthcare.

22 **[00237]** In one embodiment, a user may check in at a kiosk at a healthcare
23 provider's office 1002, e.g., a POS registry a hospital, a clinic, and/or the like. The

1 physician or other healthcare provider may provide healthcare service to the user 1006.
2 In one embodiment, the physician's office determines whether or not the user is insured
3 1010. If the user is insured, then process moves to step 1012. Otherwise, the process
4 moves to step 1016.

5 **[00238]** In one implementation, the physician's Point Of Service terminal (POS)
6 may send a bill to the user's insurance company for the healthcare that was provided to
7 the user. For example, the healthcare provider may send the medical bill directly to an
8 insurance provider via mail, email, instant message, and/or the like. For another
9 example, the healthcare provider may submit information related to the medical bill

10 **[00239]** In one embodiment, at step 1014, the physician's point of service terminal
11 receives partial compensation from the user's insurance company for the healthcare that
12 was provided to the user. At step 1016, the physician's point of service terminal sends a
13 balance due billing to the user's mobile device, for instance, to an email address or as a
14 text message by use of the user's cellular telephone number.

15 **[00240]** In one embodiment, at step 1018, the mobile device renders to the user a
16 description of the bill as received for the balance due billing from the physician. The
17 rendered bill, shown in step number 1018, shows the amount due, the description of the
18 goods and/or services of the healthcare provided to the user by the healthcare provider,
19 and a MCC of the physician or healthcare provider. At step 1020 the user's web-enabled
20 device executes an application, which may also perform the rendering at step 1018,
21 where a decisioning process takes place in order to satisfy the bill rendered at step 1018.

22 **[00241]** In one embodiment, the user may obtain and install a mobile application
23 which determines payment accounts in order to pay the bill shown in step 1018. To

1 make the determination, the mobile application draws upon one or more online
2 databases to which it has access. Arrow 1022 shows online access to a plurality of
3 databases 1024. These databases include a database having miscellaneous data for the
4 user, a database for insurance payment coverage rules, a database for local and
5 government rules, and one or more databases showing various account balances that
6 have been issued by issuers to the user that have credit or currency available to satisfy
7 the bill shown in step 1018. Various rules for incentives and penalties are contained
8 within in the databases as seen at block 1024. For instance, available balances for
9 Medicare Part D provisions can be shown as being available to satisfy the bill in step
10 1018.

11 **[00242]** The various databases can also include considerations for government
12 insurance, pharmacy benefits, employer healthcare considerations, employer pharmacy
13 benefit plans, employer or government subsidizing of healthcare goods and services, and
14 incentives or penalties to use accounts according to code provisions as provided by
15 various business rules. The available deductibles and required deductibles for each of
16 the one or more benefit plans can be found in one or more databases seen at reference
17 numeral 1024, as well as various co-pay requirements, healthcare spending limits, and
18 various restricted use currency amounts. Various forfeiture rules, such as are applicable
19 to FSA can also be found in databases 1024. The relative merits of using an HSA, with
20 its restricted use deposit benefits, as well as the ability to grow its balance in terms of
21 both compounding interest and the probability of a rise in the values of various equity
22 holdings, are also taken into consideration. The various user account balances
23 maintained by the databases of block 1024 can e assessed via one or more issuers of the

1 respective user accounts as seen at 1034. Each issuer is an issuer to an account of the
2 user, who is an account holder on that account that was issued by the issuer.

3 **[00243]** After the mobile application seen at process 1020 receives information,
4 business rules, and data via communication seen at arrow 1022, the process 1020
5 calculates a recommendation of one or more accounts having respective one or more
6 amounts to be paid from each account. This recommendation will provide the most
7 favorable tax, cost, and benefits to the user by using the amounts and respective
8 accounts, while also minimized penalties for such use. The mobile applications
9 recommendations are rendered on the mobile device at step 1028a. The rendering on
10 the web-enabled mobile device may also guard access such as by prompting for, and
11 validating, a user name and the password to enable making withdrawals from respective
12 accounts for respective amounts suggested by process 1020. Each account can be
13 identified by a nickname or identifier, and the nickname will be listed along with the
14 amount that is recommended to be paid from that account toward the balance due
15 billing shown at block 1018.

16 **[00244]** For example, in one implementation, a Visa debit or credit account or a
17 prepaid card may be suggested and identified by a nickname (i.e., a partial account
18 number) along with an amount to be paid from that account. The user has the option to
19 accept or reject the recommendation made as rendered on the web-enabled mobile
20 device at step 1028a. If the user decides to reject the payment recommendation, an
21 override can be submitted by the user to change the account and/or amounts and to
22 make effective the changes or to amend the recommendations as to the amounts to be
23 paid from various accounts by the user to the physician. This payment is seen in step

1 1028b where the physician's POS receives a wireless communication from the user's
2 web-enabled mobile device. This wireless communication will contain data that reflects
3 each account and each corresponding amount to be paid from each account to satisfy
4 the balance due billing shown at step 1018.

5 **[00245]** In one embodiment, at arrows 1030 and 1032, the physician
6 communicates with its acquirer and with a transaction handler (i.e., VisaNet) to send an
7 authorization request for each payment for each account that is designated by the
8 wireless communication from the web-enabled mobile device to the physician's POS.
9 The authorization request is sent from VisaNet via communication 1034 to the issuer of
10 each account from which a payment is to be made. Each issuer, respectively, sends an
11 authorization response to the authorization request back to VisaNet, which is in turn
12 sent from VisaNet to the physician's acquirer as shown by communication arrow 1032,
13 and from there to the physician's acquirer via communication arrow 1030 back to the
14 physician's POS. Once the physician's POS has received an authorization response for
15 the payment from each account, then the physician may deem that the bill, as shown in
16 block 1018, has been satisfied. Thereafter, the physician's office, with its acquirer, will
17 initiate a clearing and settlement process for each authorized payment from each
18 account that was used to satisfy the balance due billing seen at block 1018.

19 **[00246]** FIGURE 10B provides a flow diagram illustrating alternative embodiments
20 of RUAP. A physician has a point of service terminal that sends balance due billing to
21 the patient's web-enabled mobile device 1032, and access to information and data
22 interactively between various online databases and the mobile application executing on
23 a patient's web-enabled mobile device 1034. Block 1036 shows access to retrieve various

1 restricted use rules and benefits that can be input and considered to make a
2 recommendation as to a payment which should be made from one or more accounts. In
3 further implementations, income financial brackets for the patient's year may also be
4 taken into consideration in arriving at a recommendation.

5 **[00247]** In one embodiment, considerations are also input through various online
6 databases to show insurance payment coverage rules 1038. These business rules can
7 include: (i) that portion of healthcare services that are covered or not covered for a
8 healthcare service that is rendered by a physician to a patient; (ii) various specific
9 spending rule limits and forfeiture rules, various annual and lifetime co-payment and
10 maximum insurance payments by the person and/or by the policy, various limits for
11 various goods and services which may or may not be reimbursable under insurance
12 including pharmacy, vision, and dental payments to respective healthcare service
13 providers; (iii) insurance coverage for various types of merchants that are available as
14 benefits and restriction of benefits including big box retailers, retail pharmacy
15 organizations, physician-owned organizations, rehabilitation organizations, various
16 public and private hospitals, as well as various private preferred providers for respective
17 insurance plans; and (iv) copayments that are available for each of several different
18 insurance vehicles.

19 **[00248]** In one embodiment, the various patient account balances may be retrieved
20 to determine availability of currency or funds to pay the balance due bill received from
21 the healthcare provider 1040. These accounts can be assessed by online communication
22 with the respective issuers to determine account balances. By way of example, these
23 balances can include: (i) a balance for one or more Flexible Savings Accounts (FSA),

1 including a current balance and the date by which all funds in each FSA account must be
2 spent; (ii) one or more HSA including a liquid asset balance, a non-liquid asset balance
3 that can including stocks, mutual funds, certificates of deposit, etc. In that some
4 equities must be traded for cash in order to have access to liquid assets to satisfy the
5 healthcare provider's balance due bill, the retrieved information can include various
6 requirements for selling stock or other securities, including commission charges, which
7 information can be taken into consideration by the decisioning application in making
8 the recommendation; (iii) balances for government insurance prepaid accounts, such as
9 Medicare and Medicaid; (iv) private insurance deductibles outstanding and yet to be
10 paid; (v) other restricted use accounts that are available to satisfy the healthcare
11 provider's balance due bill, such as employee benefit plans; (vi) non-restricted use
12 accounts and likely cash flow predictions for in each one of those accounts, such as
13 credit available in credit cards, cash available in debit card accounts, cash available on
14 prepaid card accounts, as well as any currency that is available by converting loyalty
15 points for each one of these accounts so that the loyalty points can be used as currency
16 toward balance due billing payments. Also available are calculations made by the
17 mobile application of award thresholds if a payment is made so as to thereby realize
18 more loyalty points that can then be converted into currency to satisfy the healthcare
19 provider's balance due billing.

20 **[00249]** The various inputs and data that are retrieved are subjected to various
21 calculations as derived from steps 1036 through 1040 so that the mobile application can
22 make a recommendation as to each account 1042, and each amount to be paid from
23 each account, that will be in the patient's best interest to pay a balance due billing

1 received by the web-enabled mobile device from the patient's physician or other
2 healthcare provider via a point of service terminal.

3 **[00250]** FIGURE 10C shows an exemplary screen shot of a display terminal within
4 embodiments of the RUAP. In one implementation, a horizontal and vertical icon is
5 rendered on the screen so that a user can navigate to view vertical and horizontal
6 portions of the display screen, as indicated by icons 1050, 1052. Screen shot shows the
7 total balance due from the physician as well as each of the accounts 1 through N, and
8 respective amounts to be paid from accounts 1 through N, as recommended by the
9 mobile application to satisfy the healthcare provider's balance due billing. As shown in
10 display screen, the patient can accept the recommended payments from each
11 recommended account by clicking in one location. Alternatively, the patient can edit the
12 respective accounts and their respective payments from each account, by 'clicking' on an
13 icon at another location. As a third option, the user can 'click on' an icon to receive a
14 rendering of an explanation on display screen as to why the recommendations from
15 each account for each amount was recommended. The explanation will give the patient
16 an understanding upon which the patient can base an approval, modification, or
17 rejection of the recommended payments from each account.

18 **[00251]** Once the recommendations are accepted, the process taken place as shown
19 in steps 1056 through 1060, where the patient's web-enabled mobile device transmits to
20 the physician's point of service terminal a communication that describes the payment to
21 be made from each account. An e-commerce server, shown at block 1058, processes
22 each payment from each account as is described in FIGURE 10B through the issuer
23 processor, the acquirer processor, and the transaction handler (VisaNet) for a clearing

1 and settlement process by which the physician's accounts receivable satisfied as to the
2 balance due payment made by the patient, as shown in block 1056.

3 **[00252]** In one implementation, the patient may operate a web-enabled mobile
4 computing device that can be executing a World Wide Web browser, or other special
5 purpose software, in order to access databases.

6 **[00253]** In one implementation, the RUAP may allow the patient to view specifics
7 of the balance due billing that the physician is charging the patient, as well as funds for
8 payment of the balance due billing. The patient can provide information to the web-
9 enabled mobile device in order to gain access to financial information stored by each
10 issuer that issued an account to the patient. To access financial information for the
11 patient, a name and password can be required. Once supplied by the patient, financial
12 information can be sent and retrieved. This information can include account issuer
13 identifiers (e.g.; account numbers), the name of the issuer who issued the account
14 numbers, and any amounts that the financially responsible person wishes to pay on
15 balance due billing to the doctor. Specifics of a bill that the patient can view may
16 include: (i) the healthcare organization name that provided healthcare services to the
17 patient, (ii) the name of the physician who treated the patient, (iii) the name of the
18 person who is financially responsible for the patient, (iv) the name of the patient, (v) the
19 date the services were provided by the doctor to the patient, (vi) a description of the
20 healthcare goods and/or services that were rendered to the patient by the doctor, (vii)
21 any amounts paid by the insurance company for the foregoing healthcare goods and
22 services, and (viii) any balance due by the person who is financially responsible for the
23 patient to the healthcare organization.

1 **[00254]** FIGURES 11A-11D provide various exemplary PoS terminal interfaces for
2 user checkout with snap bill/QR code (e.g., see FIGURE 2C) within embodiments of the
3 RUAP. With reference to FIGURE 11A, in some implementations, such merchant-
4 product information embodying QR codes may be utilized by a point-of-sale ("POS")
5 terminal, e.g., 1110a-b. For example, in a brick-and-mortar store, the POS terminal may
6 display a QR code, e.g., 1111a-b, that includes the purchase payment amount, e.g., 1112a-
7 b, upon the user indicating that the user wishes to checkout the items in the user's
8 physical shopping cart.

9 **[00255]** In one implementation, when the PoS terminal comprises a smart
10 component, e.g., as described in FIGURES 2A, the PoS terminal may determine whether
11 an item is eligible for any restricted-use account. For example, the PoS terminal may
12 display a message 1112c showing a list of pharmaceutical items eligible for FSA, and
13 prompt a cashier and/or the user to determine whether they would like to check out
14 with the FSA account. If the user selects "yes" at the screen 1112c, the PoS terminal may
15 generate a separate bill for the FSA eligible items 1112d, and the user may proceed to
16 payment with the FSA account for the eligible items only. In this case, the user may not
17 need a smart phone or electronic wallet to engage in RUAP service, as the PoS terminal
18 may provide a user interface for the user.

19 **[00256]** With reference to FIGURE 11B, in some implementations, the user may
20 obtain a snapshot of the QR code displayed on the screen of the secure display or the
21 POS terminal using a smartphone, e.g., 1113. For example, the user's smartphone may
22 have an app, e.g., 1114, executing on it to detect and capture QR codes, e.g., 1116a. For
23 example, the user may utilize registration features, e.g., 1115, to align the QR code within

1 the display of the smartphone. The app may, in some implementations, provide the user
2 with the ability to zoom in, e.g., 1117, or zoom out, e.g., 1118, of the QR code to ensure
3 that the image of the QR code fits within the dimensions of the screen of the smartphone.
4 Upon aligning the QR code within the display of the smartphone, the user may be able
5 to obtain a snapshot of the QR code using a user interface element, e.g., 1119. The user
6 may cancel the snap mobile payment procedure using a user interface element 1120 on
7 the display of the smartphone.

8 **[00257]** With reference to FIGURE 11C, in some implementations, RUAP
9 reimbursement may be utilized for authentication/verification purposes, and for
10 providing digital consent for disclosure of personal and/or private information. For
11 example, a user 1142 visiting his/her doctor 1143 may be required to provide informed
12 consent to disclosing personal information (e.g., medical records) to the doctor. The
13 doctor's terminal may generate a QR code embodying the doctor's digital certificate as
14 well as information on the type/content of medical records of the user that are
15 requested, e.g., 1144. The user may snap the QR code via the user's mobile device. The
16 user's mobile device may generate a request for records release according to the QR
17 code, and also serve as verification that the request is obtained from a personal trusted
18 device (e.g., the user's mobile device). In alternate implementations, the user may be
19 able to select the personal information that the user would like to reveal to the
20 healthcare provider, and the user's mobile device may generate a QR code for the
21 doctor's terminal to obtain a snapshot for retrieving the user's medical information. In
22 some implementations, the QR code may also include payment information (e.g., the
23 user's pay account information, or the doctor's acquirer information) along with the
24 information on controlled release of personal information.

1 **[00258]** In some implementations, the RUAP may facilitate P2P transactions via
2 pre-filled, modifiable QR payment codes, e.g., 1150. For example, a first user having a
3 public profile page, e.g., 1151, may place an image of a QR code in the public profile, e.g.,
4 1152. For example, the QR code may include a predetermined payment amount for a
5 purchase transaction initiated by capturing a snapshot of the QR code. In some
6 implementations, the predetermined amount may be \$0 (e.g., a \$0 QR pay code). A
7 second user may capture a snapshot of the QR pay code using a mobile device, and may
8 set an amount that the second user would like to pay the first user via the second user's
9 mobile device. The second user's mobile device may provide the information encoded
10 within the QR code along with the second-user-chosen payment amount to a payment
11 network for transaction processing.

12 **[00259]** It is to be understood that the various aspects described herein of snap
13 mobile payment may be utilized for any controlled exchange of information and/or
14 payment. For example, with reference to FIGURE 11D, in some implementations, a user
15 may obtain pay-per-view programming via snap mobile payment, e.g., 1160. For
16 example, a television display may provide an advertisement including programming
17 information, e.g., 1162, as well as a QR pay code for obtain the programming content,
18 e.g., 1161. The QR code may include information identifying the programming
19 information, as well as information identifying the television subscriber account
20 information, television machine address, and/or the like. The user may obtain a
21 snapshot of the QR code, and provide the information embodied in the QR code along
22 with information from the user's mobile device (e.g., subscriber account number linked
23 to the user's virtual wallet, pay account information, and/or the like). Upon processing
24 of the payment information by the payment network, the payment network may provide

1 an indication to the television-programming provider of the payment completion, and
2 the television-programming provider may stream the programming content to the user's
3 television. As another example, a similar flow may be utilized for in-flight
4 entertainment, e.g., 1170, wherein an in-flight screen may provide programming
5 information 1172, as well as a QR pay code 1171 for the user to snap for in-flight
6 entertainment initiation. As another example, a billboard, wall hanging, poster, in-store
7 advertisement, hoarding, etc., e.g., 1180, may include an offer for a product/service, and
8 a QR code including merchant information and product information identifying a
9 purchase amount, and/or the like. The user may snap the QR code with the user's
10 mobile device linked to the user's virtual wallet to purchase the product and/or service,
11 and, if applicable, the product may be directly shipped to the user's address as specified
12 by the purchase information exchanged with the payment network as part of the
13 purchase request sent by the user's mobile device. As another example, newspapers, e.g.,
14 1185, may include offers, advertisements, job postings, and/or the like including QR
15 codes, e.g., 1186, embodying the information necessary for the user to initiate a
16 purchase transaction with the payment network. It is to be understood that any aspects
17 of implementing snap mobile payment discussed in any of the implementations herein,
18 and/or their equivalents, may be utilized in any other implementations discussed herein,
19 and/or their equivalents.

20 **[00260]** FIGURES 12A-12E provide exemplary UIs of a web-based shopping
21 checkout with RUAP within embodiments of the RUAP. With reference to FIGURE 12A,
22 in some implementations, a user may desire to checkout one or items stored in a virtual
23 shopping cart of an online merchant website. For example, the user may be utilizing a
24 browser application, e.g., 1201, to visualize a checkout page of the merchant website, e.g.,

1 1202. The checkout webpage may depict details of the checkout order, e.g., 1203, and
2 may provide one or more options for the user to provide payment for the purchase of the
3 store items. In some implementations, the checkout webpage may include an option to
4 pay for the purchase using a snap mobile payment procedure, e.g., 1204.

5 **[00261]** With reference to FIGURE 12B, in some implementations, upon selecting
6 the option to utilize the snap mobile payment procedure, the merchant checkout
7 webpage, e.g., 1206, may provide via the browser application 1205, a QR code, e.g., 1209,
8 including information on the items in the virtual shopping cart as well as merchant
9 information for the payment network to process the purchase transaction (e.g., a privacy
10 token/alias linked to an acquirer financial account of the merchant). In some
11 implementations, the webpage may be displayed via a secure display of a trusted
12 computing device of the user. For example, as a security measure, the position of the
13 QR code frame, e.g., 1207, within the display may be randomly varied to prevent a
14 snapshot of the QR code from being obtained by fraudulent means (e.g., tampering with
15 the trusted computing device). In some implementations, a security image, e.g., 1208,
16 pre-selected by the user may be displayed on the screen so that the user may verify as
17 being accurate. In some implementations, the image may be encrypted by the RUAP
18 before providing it to the trusted computing device. In some implementations, the
19 trusted computing device may be the only device to hold a decryption key required to
20 decrypt and successfully display the image on the secure display to the user.

21 **[00262]** With reference to FIGURE 12C, in some implementations, upon obtaining
22 a snapshot of the merchant-product QR code, the user's smartphone may extract the
23 product and merchant data stored within the QR code, and utilize an account for the

1 user's virtual wallet linked to the user's smartphone to generate a purchase transaction
2 request for processing by a payment network. Upon completion of processing of the
3 payment transaction by the payment network using the information provided by the
4 user's smartphone, the merchant website 1222 (via the browser application 1221) may
5 provide a purchase receipt 1225 for the user. With reference to FIGURE 12F, in
6 implementations where the user utilizes the snap mobile payment procedure at a brick-
7 and-mortar store, the POS terminal may display a purchase receipt for the user. In
8 some implementations, the payment network may provide a purchaser receipt directly
9 to the smartphone of the user. In further implementations, the user may enter a phone
10 number, email address, instant messenger id, wallet id, social id, and/or the like, to the
11 merchant PoS terminal, so that the terminal may send an electronic receipt to the user
12 via email, SMS, instant messaging, wallet messaging, social messaging, and/or the like.

13 **[00263]** In one implementation, upon providing an electronic receipt, the RUAP
14 may determine whether items in the receipt are eligible for a restricted-use account. For
15 example, the RUAP may provide a RUAP alert 1227, notifying the user that one or more
16 items are eligible for "FSA account usage," and inquire whether the user would like to
17 claim a refund from the FSA account. In one implementation, the user may select to
18 "email e-receipt for refund" 1228, and the RUAP may automatically forward the
19 electronic receipt to the RUAP server for reimbursement processing (e.g., continuing on
20 with 419 in FIGURE 4A).

21 **[00264]** With reference to FIGURE 12D, in some implementations, a user may be
22 advantageously able to provide user settings into a device producing a QR code for a
23 purchase transaction, and then capture the QR code using the user's mobile device. For

1 example, a display device of a point-of-sale terminal may be displaying a checkout
2 screen, such as a web browser executing on a client, e.g., 1231, displaying a checkout
3 webpage of an online shopping website, e.g., 1232. In some implementations, the
4 checkout screen may provide a user interface element, e.g., 1233a-b, whereby the user
5 can indicate the desire to utilize snap mobile payment. For example, if the user activates
6 element 1231a, the website may generate a QR code using default settings of the user,
7 and display the QR code, e.g., 1235, on the screen of the client for the user to capture
8 using the user's mobile device. In some implementations, the user may be able to
9 activate a user interface element, e.g., 1233b, whereby the client may display a pop-up
10 menu, e.g., 1234, with additional options that the user may select from. For example,
11 the website may provide user selection options similar to those discussed above in the
12 description with reference to FIGURES 12B-C. In some implementations, the website
13 may modify the QR code 1235 in real-time as the user modifies settings provided by
14 activating the user interface element 1233b. Once the user has modified the settings
15 using the pop-up menu, the user may capture a snapshot of the QR code to initiate
16 purchase transaction processing.

17 **[00265]** In one implementation, the RUAP may generate a RUAP alert along with
18 the pop-up window when one or more items are eligible for restricted-use accounts 1236.
19 For another example, the RUAP may provide account recommendation by listing
20 eligible restricted-use account (e.g., "FSA" as shown in FIGURE 12D) on top of the
21 account list.

22 **[00266]** With reference to FIGURE 12E, in some implementations, the RUAP may
23 provide the user with a user interface to modify the user's snap mobile payment settings.

1 For example, the RUAP may provide a web interface, e.g., 1241. For example, the user
2 may be able to modify security settings of the user's virtual wallet, e.g., 1242, using the
3 web interface. For example, the user may review a list of trusted device, e.g., 1244, via
4 which the user may access the user's virtual wallet. In some implementations, the web
5 interface may provide a user interface element to add a trusted device, e.g., 1243. The
6 web interface may also provide the user with additional security options. For example,
7 the user be able to set a security passphrase, e.g., 1245, modify settings for when the
8 user should be challenged before authorizing a purchase transaction, e.g., 1246, the
9 type/style of presentation of the security features, e.g., 1247, and a security image to be
10 displayed on the terminal utilized in snap mobile payment, e.g., 1248. In various
11 implementations, the user may be able to access other services including modifying user
12 profiles, accounts, account preferences, adding cards, obtaining offers and coupons,
13 locating ATM machines, etc.

14 **[00267]** FIGURES 13A-13C provide exemplary mobile UIs illustrating user
15 capturing a barcode/QR code for checkout/reimbursement within embodiments of the
16 RUAP. It should be noted that although a mobile wallet platform is depicted, a
17 digital/electronic wallet, a smart/prepaid card linked to a user's various payment
18 accounts, and/or other payment platforms are contemplated embodiments as well; as
19 such, subset and superset features and data sets of each or a combination of the
20 aforementioned payment platforms may be accessed, modified, provided, stored, etc. via
21 cloud/server services and a number of varying client devices throughout the instant
22 specification. Similarly, although mobile wallet user interface elements are depicted,
23 alternative and/or complementary user interfaces are also contemplated
24 including: desktop applications, plug-ins to existing applications, stand alone mobile

1 applications, web based applications (e.g., applications with web objects/frames, HTML
2 5 applications/wrappers, web pages, etc.), and other interfaces are contemplated. It
3 should be further noted that the RUAP payment processing component may be
4 integrated with an digital/electronic wallet (e.g., a Visa V-Wallet, etc.), comprise a
5 separate stand alone component instantiated on a user device, comprise a server/cloud
6 accessed component, be loaded on a smart/prepaid card that can be substantiated at a
7 PoS terminal (e.g., see 109 in FIGURE 1B, 1715 in FIGURE 17A), an ATM, a kiosk, etc.,
8 which may be accessed through a physical card proxy, and/or the like.

9 **[00268]** With reference to FIGURE 13A, in some implementations, the app
10 executing on the device of the user may include an app interface providing various
11 features for the user. In some implementations, the app may be configured to recognize
12 product identifiers (e.g., barcodes, QR codes, etc.), e.g., 1301. In some implementations,
13 the user may be required to sign in to the app to enable its features. Once enabled, the
14 camera may provide in-person one tap purchasing features for the user. For example,
15 the client device may have a camera via which the app may acquire images, video data,
16 streaming live video, and/or the like, e.g., 1303. The app may be configured to analyze
17 the incoming data, and search, e.g., 1301, for a product identifier, e.g., 1304, such as QR
18 codes 209, 211a-b, 216a and 227. In some implementations, the app may overlay cross-
19 hairs, target box, and/or like alignment reference markers, e.g., 1305, so that a user may
20 align the product identifier using the reference markers so facilitate product identifier
21 recognition and interpretation. In some implementations, the app may include
22 interface elements to allow the user to switch back and forth between the product
23 identification mode and product offer interface display screens (see, e.g., 1306), so that a
24 user may accurately study deals available to the user before capturing a product

1 identifier. In some implementations, the app may provide the user with the ability to
2 view prior product identifier captures (see, e.g., 1307) so that the user may be able to
3 better decide which product identifier the user desires to capture. In some
4 implementations, the user may desire to cancel product purchasing; the app may
5 provide the user with a user interface element (e.g., 1308) to cancel the product
6 identifier recognition procedure and return to the prior interface screen the user was
7 utilizing. In some implementations, the user may be provided with information about
8 products, user settings, merchants, offers, etc. in list form (see, e.g., 1309) so that the
9 user may better understand the user's purchasing options. Various other features may
10 be provided for in the app (see, e.g., 1310).

11 **[00269]** With reference to FIGURE 13B, in some implementations, the app may
12 include an indication of the location (e.g., name of the merchant store, geographical
13 location, information about the aisle within the merchant store, etc.) of the user, e.g.,
14 1311. The app may provide an indication of a pay amount due for the purchase of the
15 product, e.g., 1312. In some implementations, the app may provide various options for
16 the user to pay the amount for purchasing the product(s). For example, the app may
17 utilize the GPS coordinates to determine the merchant store within the user is present,
18 and direct the user to a website of the merchant. In some implementations, the RUAP
19 may provide an API for participating merchants directly to facilitate transaction
20 processing. In some implementations, a merchant-branded RUAP application may be
21 developed with the RUAP functionality, which may directly connect the user into the
22 merchant's transaction processing system. For example, the user may choose from a
23 number of accounts (e.g., credit cards, debit cards, prepaid cards, etc.) from various
24 issuers, e.g., under the account type "RU" (restricted-use) at 1313. In some

1 implementations, the app may provide the user the option to pay the purchase amount
2 using funds included in an account of the user, e.g., a checking, savings, money market,
3 current account, etc., e.g., a "FSA" account 1314. In some implementations, the user
4 may have set default options for which card, bank account, etc. to use for the purchase
5 transactions via the app. In some implementations, such setting of default options may
6 allow the user to initiate the purchase transaction via a single click, tap, swipe, and/or
7 other remedial user input action, e.g., 1315a. In some implementations, when the user
8 utilizes such an option, the app may utilize the default settings of the user to initiate the
9 purchase transaction. In some implementations, the app may allow the user to utilize
10 other accounts (e.g., Google™ Checkout, Paypal™ account, etc.) to pay for the purchase
11 transaction, e.g., 1316. In some implementations, the app may allow the user to utilize
12 rewards points, airline miles, hotel points, electronic coupons, printed coupons (e.g., by
13 capturing the printed coupons similar to the product identifier) etc., to pay for the
14 purchase transaction, e.g., 1317-1318. In some implementations, the app may provide an
15 option to provide express authorization before initiating the purchase transaction, e.g.,
16 1319. In some implementations, the app may provide a progress indicator provide
17 indication on the progress of the transaction after the user has selected an option to
18 initiate the purchase transaction, e.g., 1320. In some implementations, the app may
19 provide the user with historical information on the user's prior purchases via the app,
20 e.g., 1321. In some implementations, the app may provide the user with an option to
21 share information about the purchase (e.g., via email, SMS, wall posting on Facebook®,
22 tweet on Twitter™, etc.) with other users and/or control information shared with the
23 merchant, acquirer, payment network etc., to process the purchase transaction, e.g.,
24 1322. In some implementations the app may provide the user an option to display the

1 product identification information captured by the client device (e.g., in order to show a
2 customer service representative at the exit of a store the product information), e.g., 1324.
3 In some implementations, the user, app, device and or RUAP may encounter an error in
4 the processing. In such scenarios, the user may be able to chat with a customer service
5 representative (e.g., VerifyChat 1323) to resolve the difficulties in the purchase
6 transaction procedure.

7 **[00270]** In some implementations, the user may select to conduct the transaction
8 using a one-time anonymized credit card number, see e.g., 1315b. For example, the
9 RUAP may utilize a pre-designated anonymized set of card details (see, e.g.,
10 “AnonCard1,” “AnonCard2”). As another example, the RUAP may generate, e.g., in real-
11 time, a one-time anonymous set of card details to securely complete the purchase
12 transaction (e.g., “Anon It 1X”). In such implementations, the app may automatically
13 set the user profile settings such that the any personal identifying information of the
14 user will not be provided to the merchant and/or other entities. In some
15 implementations, the user may be required to enter a user name and password to enable
16 the anonymization features.

17 **[00271]** With reference to FIGURE 13C, in some implementations, the user
18 interface elements of the snap mobile payment app may be advantageously arranged to
19 provide the user the ability to process a purchase with customized payment parameters
20 with a minimum number of user gestures applied to the user’s mobile device. For
21 example, the user may be provided with an overloaded user interface element, e.g.,
22 1325-326. For example, if the user has a QR pay code within the viewing angle of a
23 camera included in the user’s mobile device, the user may activate element 1325 to snap

1 the QR code and utilize predetermined default settings to process the purchase based on
2 the QR code. However, if the user wishes to customize the payment parameters, the
3 user may activate user interface element 1326 (e.g., press and continue to hold). Upon
4 doing so, the app may provide a pop-up menu, e.g., 1327, providing a variety of payment
5 customization choices, such as those provided previously. The user may, e.g., drag the
6 user's finger to the appropriate settings the user prefers, and release the user's finger
7 from the touchscreen of the user's mobile device to select the setting for payment
8 processing. In alternate implementations, the payment settings options, e.g., 1330, and
9 QR capture activation buttons, e.g., 1328a-b (e.g., 1328b may provide even more settings
10 that those displayed in the initial screen), may be included in the user interface along
11 with a window, e.g., 1329, for capturing the QR code via the mobile device's camera. In
12 alternate implementations, the user's mobile device may generate a hybrid QR code-
13 payment settings graphic, and the POS terminal (or user's trusted computing device)
14 may capture the entire graphic for payment processing.

15 **[00272]** FIGURES 14A-14C provide exemplary mobile wallet UIs illustrating
16 making a payment within embodiments of the RUAP. With reference to FIGURE 14A,
17 some embodiments of the virtual wallet mobile app facilitate and greatly enhance the
18 shopping experience of consumers. A variety of shopping modes, as shown in FIGURE
19 14A, may be available for a consumer to peruse. In one implementation, for example, a
20 user may launch the shopping mode by selecting the shop icon 1410 at the bottom of the
21 user interface. A user may type in an item in the search field 1412 to search and/or add
22 an item to a cart 1411. A user may also use a voice activated shopping mode by saying the
23 name or description of an item to be searched and/or added to the cart into a
24 microphone 1413. In a further implementation, a user may also select other shopping

1 options 1414 such as current items 1415, bills 1416, address book 1417, merchants 1418
2 and local proximity 1419.

3 **[00273]** In one embodiment, for example, a user may select the option current
4 items 1415, as shown in the left most user interface of FIGURE 14A. When the current
5 items 1415 option is selected, the middle user interface may be displayed. As shown, the
6 middle user interface may provide a current list of items 1415a-h in a user's shopping
7 cart 1411. A user may select an item, for example item 1415a, to view product description
8 1415j of the selected item and/or other items from the same merchant. The price and
9 total payable information may also be displayed, along with a QR code 1415k that
10 captures the information necessary to effect a snap mobile purchase transaction.

11 **[00274]** With reference to FIGURE 14B, in another embodiment, a user may select
12 the bills 1416 option. Upon selecting the bills 1416 option, the user interface may display
13 a list of bills and/or receipts 1416a-h from one or more merchants. Next to each of the
14 bills, additional information such as date of visit, whether items from multiple stores are
15 present, last bill payment date, auto-payment, number of items, and/or the like may be
16 displayed. In one example, the wallet shop bill 1416a dated January 20, 2015 may be
17 selected. The wallet shop bill selection may display a user interface that provides a
18 variety of information regarding the selected bill. For example, the user interface may
19 display a list of items 1416k purchased, <<916i>>, a total number of items and the
20 corresponding value. For example, as shown at 1416a, the user may elect to pay for a
21 bill for "Knee Surgery" 1416a performed at 1/20/2015, which comprises details of the
22 healthcare treatment performed for the user 1416k.

1 **[00275]** A user may now select any of the items and select buy again to add
2 purchase the items. The user may also refresh offers 1416j to clear any invalid offers
3 from last time and/or search for new offers that may be applicable for the current
4 purchase. As shown in FIGURE 14A, a user may select two items for repeat purchase.
5 Upon addition, a message 1416l may be displayed to confirm the addition of the two
6 items, which makes the total number of items in the cart 14.

7 **[00276]** In some implementations, the RUAP wallet may provide the RUAP with
8 the GPS location of the user. Based on the GPS location of the user, the RUAP may
9 determine the context of the user (e.g., whether the user is in a store, doctor's office,
10 hospital, postal service office, etc.). Based on the context, the user app may present the
11 appropriate fields to the user, from which the user may select fields and/or field values
12 to send as part of the purchase order transmission. For example, a user may go to
13 doctor's office and desire to pay the co-pay for doctor's appointment. In addition to
14 basic transactional information such as account number and name, the app may provide
15 the user the ability to select to transfer medical records, health information, which may
16 be provided to the medical provider, insurance company, as well as the transaction
17 processor to reconcile payments between the parties. In some implementations, the
18 records may be sent in a Health Insurance Portability and Accountability Act (HIPAA)-
19 compliant data format and encrypted, and only the recipients who are authorized to
20 view such records may have appropriate decryption keys to decrypt and view the private
21 user information.

22 **[00277]** With reference to FIGURE 14C, in one embodiment, the wallet mobile
23 application may provide a user with a number of options for paying for a transaction via

1 the wallet mode. In one implementation, an example user interface 1411 for making a
2 payment is shown. The user interface may clearly identify the amount 1412 and the
3 currency for the transaction. The amount may be the amount payable and the currency
4 may include real currencies such as dollars and euros, as well as virtual currencies such
5 as reward points. The amount of the transaction 1414 may also be prominently displayed
6 on the user interface. The user may select the funds tab 1416 to select one or more forms
7 of payment 1417, which may include various credit, debit, gift, rewards and/or prepaid
8 cards. The user may also have the option of paying, wholly or in part, with reward points.
9 For example, the graphical indicator 1418 on the user interface shows the number of
10 points available, the graphical indicator 1419 shows the number of points to be used
11 towards the amount due 234.56 and the equivalent 1420 of the number of points in a
12 selected currency (USD, for example).

13 **[00278]** In one implementation, the user may combine funds from multiple
14 sources to pay for the transaction. The amount 1419 displayed on the user interface may
15 provide an indication of the amount of total funds covered so far by the selected forms of
16 payment (e.g., Discover card and rewards points). The user may choose another form of
17 payment or adjust the amount to be debited from one or more forms of payment until
18 the amount 1419 matches the amount payable 1414. Once the amounts to be debited
19 from one or more forms of payment are finalized by the user, payment authorization
20 may begin.

21 **[00279]** In one implementation, the user may select a secure authorization of the
22 transaction by selecting the cloak button 1422 to effectively cloak or anonymize some
23 (e.g., pre-configured) or all identifying information such that when the user selects pay

1 button 1421, the transaction authorization is conducted in a secure and anonymous
2 manner. In another implementation, the user may select the pay button 1421 which may
3 use standard authorization techniques for transaction processing. In yet another
4 implementation, when the user selects the social button 1423, a message regarding the
5 transaction may be communicated to one of more social networks (set up by the user)
6 which may post or announce the purchase transaction in a social forum such as a wall
7 post or a tweet. In one implementation, the user may select a social payment processing
8 option 1423. The indicator 1424 may show the authorizing and sending social share data
9 in progress.

10 **[00280]** In another implementation, a restricted payment mode 1429 may be
11 activated for certain purchase activities such as prescription purchases. The mode may
12 be activated in accordance with rules defined by issuers, insurers, merchants, payment
13 processor and/or other entities to facilitate processing of specialized goods and services.
14 In this mode, the user may scroll down the list of forms of payments 1426 under the
15 funds tab to select specialized accounts such as a flexible spending account (FSA) 1427,
16 health savings account (HSA), and/or the like and amounts to be debited to the selected
17 accounts. In one implementation, such restricted payment mode 1429 processing may
18 disable social sharing of purchase information.

19 **[00281]** In one embodiment, the wallet mobile application may facilitate importing
20 of funds via the import funds user interface 1428. For example, a user who is
21 unemployed may obtain unemployment benefit fund 1429 via the wallet mobile
22 application. In one implementation, the entity providing the funds may also configure
23 rules for using the fund as shown by the processing indicator message 1430. The wallet

1 may read and apply the rules prior, and may reject any purchases with the
2 unemployment funds that fail to meet the criteria set by the rules. Example criteria may
3 include, for example, merchant category code (MCC), time of transaction, location of
4 transaction, and/or the like. As an example, a transaction with a grocery merchant
5 having MCC 5411 may be approved, while a transaction with a bar merchant having an
6 MCC 5813 may be refused.

7 **[00282]** With reference to FIGURE 14D, in some other embodiments, a user may
8 select merchants 1418 from the list of options in the shopping mode to view a select list
9 of merchants 1418a-e. In one implementation, the merchants in the list may be affiliated
10 to the wallet, or have affinity relationship with the wallet. In another implementation,
11 the merchants may include a list of merchants meeting a user-defined or other criteria.
12 For example, the list may be one that is curated by the user, merchants where the user
13 most frequently shops or spends more than an x amount of sum or shopped for three
14 consecutive months, and/or the like. In one implementation, the user may further select
15 one of the merchants, Amazon 1418a for example. The user may then navigate through
16 the merchant's listings to find items of interest such as 1418f-j. Directly through the
17 wallet and without visiting the merchant site from a separate page, the user may make a
18 selection of an item 1418j from the catalog of Amazon 1418a. As shown in the right most
19 user interface of FIGURE 14D, the selected item may then be added to cart. The message
20 1418k indicates that the selected item has been added to the cart, and updated number
21 of items in the cart is now 13.

22 **[00283]** With reference to FIGURE 14E, in one embodiment, there may be a local
23 proximity option 1419 which may be selected by a user to view a list of merchants that

1 are geographically in close proximity to the user. For example, the list of merchants
2 1419a-e may be the merchants that are located close to the user. In one implementation,
3 the mobile application may further identify when the user is in a store based on the user's
4 location. For example, position icon 1419d may be displayed next to a store (e.g.,
5 Walgreens) when the user is in close proximity to the store. In one implementation, the
6 mobile application may refresh its location periodically in case the user moved away
7 from the store (e.g., Walgreens). In a further implementation, the user may navigate the
8 offerings of the selected Walgreens store through the mobile application. For example,
9 the user may navigate, using the mobile application, to items 1419f-j available on aisle 5
10 of Walgreens. In one implementation, the user may select corn 1419i from his or her
11 mobile application to add to cart 1419k. In one implementation, the wallet shopping
12 cart may provide a RUAP alert 1420 notifying the purchased items (e.g., grocery items
13 1419f-j) are eligible for food voucher redemption.

14 **[00284]** With reference to FIGURE 14F, in another embodiment, the local
15 proximity option 1419 may include a store map and a real time map features among
16 others. For example, upon selecting the Walgreens store, the user may launch an aisle
17 map 1419l which displays a map 1419m showing the organization of the store and the
18 position of the user (indicated by a yellow circle). In one implementation, the user may
19 easily configure the map to add one or more other users (e.g., user's kids) to share each
20 other's location within the store. In another implementation, the user may have the
21 option to launch a "store view" similar to street views in maps. The store view 1419n may
22 display images/video of the user's surrounding. For example, if the user is about to enter
23 aisle 5, the store view map may show the view of aisle 5. Further the user may

1 manipulate the orientation of the map using the navigation tool 14190 to move the store
2 view forwards, backwards, right, left as well clockwise and counterclockwise rotation.

3 **[00285]** In further implementations, the RUAP may tag the store map view, e.g.,
4 1421, to indicate purchase item categories that are potentially eligible for restricted-
5 account usage. For example, products in the "grocery" area may be eligible for the user's
6 food stamp/voucher usage; products in the "pharmacy" area may be eligible for various
7 healthcare restricted-use accounts, e.g., FSA, HSA, etc.

8 **[00286]** FIGURE 14G provides a mobile user interface diagram illustrating
9 example features of augmented reality in-store scanning 1450 within embodiments of
10 the RUAP. For example, the RUAP wallet component may allow a user to place a
11 camera enabled device at a merchant store (e.g., scanning), and view a camera scene
12 with augmented reality labels to indicate possible items eligible for a restricted-use
13 account.

14 **[00287]** For example, in one implementation, when the user operate the camera
15 enabled device to obtain a view inside the merchant store 1450, the user may also obtain
16 augmented reality labels 1451 which identifies various products/items on the shelf, and
17 show one or more possible eligible restricted-use accounts 1452. For example, over the
18 counter drugs may be labeled as eligible for "FSA, HSA, HRA," etc., 1452; grocery
19 products may be eligible for food stamp usage; and infant food may be eligible for a
20 children nutrition benefit account, and/or the like.

21 **[00288]** FIGURES 15A-15E show user interface diagrams illustrating example
22 features of virtual wallet applications in a snap mode, in some embodiments of the
23 RUAP. With reference to FIGURE 15A, in some embodiments, a user may select a snap

1 mode 1501 to access snap features. In various embodiments, the virtual wallet
2 application may be able to snap and identify a variety of items. For example, the virtual
3 wallet application may be able to snap and identify a purchase invoice 1503, a coupon
4 104, money (e.g., sent in a person-to-person transfer) 1505, a bill (e.g., utilities, etc.)
5 1506, a receipt (e.g., for storing, expense reporting, etc.) 1507, a pay account (e.g., to add
6 a new credit/debit/prepaid card to the virtual wallet application) 1508. The user may be
7 able to return to a shopping screen at any time by activating a graphical user interface
8 element 1502. In some embodiments, the user may be able to set a name of a cart or
9 wishlist stored within the user's virtual wallet application to which the item snapped
10 should be sent (see 1509). In some embodiments, the virtual wallet application may
11 allow a user to create a new cart or wishlist to which the snapped items should be added.

12 **[00289]** In one embodiment, a user may select the snap mode 1510 to access its
13 snap features. The snap mode may handle any machine-readable representation of data.
14 Examples of such data may include linear and 2D bar codes such as UPC code and QR
15 codes. These codes may be found on receipts, product packaging, and/or the like. The
16 snap mode may also process and handle pictures of receipts, products, offers, credit
17 cards or other payment devices, and/or the like. An example user interface in snap
18 mode is shown in FIGURE 15A. A user may use his or her mobile phone to take a picture
19 of a QR code 1515 and/or a barcode 1514. In one implementation, the bar 1513 and snap
20 frame 1515 may assist the user in snapping codes properly. For example, the snap frame
21 1515, as shown, does not capture the entirety of the code 1516. As such, the code
22 captured in this view may not be resolvable as information in the code may be
23 incomplete. This is indicated by the message on the bar 1513 that indicates that the snap
24 mode is still seeking the code. The user may modify the zoom level 1517 of the camera to

1 facilitate snapping the QR code. When the code 1516 is completely framed by the snap
2 frame 1515, the bar message may be updated to, for example, “snap found.” Upon
3 finding the code, in one implementation, the user may initiate code capture using the
4 mobile device camera (see 1520). In another implementation, the snap mode may
5 automatically snap the code using the mobile device camera (see 1519). In some
6 implementations, the virtual wallet application may optionally apply a Global
7 Positioning System tag (see 1518) to the QR code before storing it, or utilizing it in a
8 transaction.

9 **[00290]** With reference to FIGURE 15B, in one embodiment, the snap mode may
10 facilitate payment reallocation post transaction. For example, a user may buy grocery
11 and prescription items from a retailer Acme Supermarket. The user may, inadvertently
12 or for ease of checkout for example, use his or her Visa card to pay for both grocery and
13 prescription items. However, the user may have an FSA account that could be used to
14 pay for prescription items, and which would provide the user tax benefits. In such a
15 situation, the user may use the snap mode to initiate transaction reallocation.

16 **[00291]** As shown, the user may enter a search term (e.g., bills) in the search bar
17 2121. The user may then identify in the tab 1522 the receipt 1523 the user wants to
18 reallocate. Alternatively, the user may directly snap a picture of a barcode on a receipt,
19 and the snap mode may generate and display a receipt 1523 using information from the
20 barcode. The user may now reallocate 1525. In some implementations, the user may also
21 dispute the transaction 1524 or archive the receipt 1526.

22 **[00292]** In one implementation, when the reallocate button 1525 is selected, the
23 wallet application may perform optical character recognition (OCR) of the receipt. Each

1 of the items in the receipt may then be examined to identify one or more items which
2 could be charged to which payment device or account for tax or other benefits such as
3 cash back, reward points, etc. In this example, there is a tax benefit if the prescription
4 medication charged to the user's Visa card is charged to the user's FSA. The wallet
5 application may then perform the reallocation as the back end. The reallocation process
6 may include the wallet contacting the payment processor to credit the amount of the
7 prescription medication to the Visa card and debit the same amount to the user's FSA
8 account. In an alternate implementation, the payment processor (e.g., Visa or
9 MasterCard) may obtain and OCR the receipt, identify items and payment accounts for
10 reallocation and perform the reallocation. In one implementation, the wallet application
11 may request the user to confirm reallocation of charges for the selected items to another
12 payment account. The receipt 1527 may be generated after the completion of the
13 reallocation process. As discussed, the receipt shows that some charges have been
14 moved from the Visa account to the FSA.

15 **[00293]** With reference to FIGURE 15C, in one embodiment, the snap mode may
16 facilitate payment via pay code such as barcodes or QR codes. For example, a user may
17 snap a QR code of a transaction that is not yet complete. The QR code may be displayed
18 at a merchant POS terminal, a web site, or a web application and may be encoded with
19 information identifying items for purchase, merchant details and other relevant
20 information. When the user snaps such as a QR code, the snap mode may decode the
21 information in the QR code and may use the decoded information to generate a receipt
22 1532. Once the QR code is identified, the navigation bar 1531 may indicate that the pay
23 code is identified. The user may now have an option to add to cart 1533, pay with a
24 default payment account 1534 or pay with wallet 1535.

1 **[00294]** In one implementation, the user may decide to pay with default 1534. The
2 wallet application may then use the user's default method of payment, in this example
3 the wallet, to complete the purchase transaction. Upon completion of the transaction, a
4 receipt may be automatically generated for proof of purchase. The user interface may
5 also be updated to provide other options for handling a completed transaction. Example
6 options include social 1537 to share purchase information with others, reallocate 1538 as
7 discussed with regard to FIGURE 15B, and archive 1539 to store the receipt.

8 **[00295]** With reference to FIGURE 15D, in one embodiment, the snap mode may
9 also facilitate offer identification, application and storage for future use. For example, in
10 one implementation, a user may snap an offer code 1541 (e.g., a bar code, a QR code,
11 and/or the like). The wallet application may then generate an offer text 1542 from the
12 information encoded in the offer code. The user may perform a number of actions on the
13 offer code. For example, the user use the find button 1543 to find all merchants who
14 accept the offer code, merchants in the proximity who accept the offer code, products
15 from merchants that qualify for the offer code, and/or the like. The user may also apply
16 the offer code to items that are currently in the cart using the add to cart button 1544.
17 Furthermore, the user may also save the offer for future use by selecting the save button
18 1545.

19 **[00296]** In one implementation, after the offer or coupon 1546 is applied, the user
20 may have the option to find qualifying merchants and/or products using find, the user
21 may go to the wallet using 1548, and the user may also save the offer or coupon 1546 for
22 later use.

1 **[00297]** With reference to FIGURE 15E, in one embodiment, the snap mode may
2 also offer facilities for adding a funding source to the wallet application. In one
3 implementation, a pay card such as a credit card, debit card, pre-paid card, smart card
4 and other pay accounts may have an associated code such as a bar code or QR code.
5 Such a code may have encoded therein pay card information including, but not limited
6 to, name, address, pay card type, pay card account details, balance amount, spending
7 limit, rewards balance, and/or the like. In one implementation, the code may be found
8 on a face of the physical pay card. In another implementation, the code may be obtained
9 by accessing an associated online account or another secure location. In yet another
10 implementation, the code may be printed on a letter accompanying the pay card. A user,
11 in one implementation, may snap a picture of the code. The wallet application may
12 identify the pay card 1551 and may display the textual information 1552 encoded in the
13 pay card. The user may then perform verification of the information 1552 by selecting
14 the verify button 1553. In one implementation, the verification may include contacting
15 the issuer of the pay card for confirmation of the decoded information 1552 and any
16 other relevant information. In one implementation, the user may add the pay card to the
17 wallet by selecting the 'add to wallet' button 1554. The instruction to add the pay card to
18 the wallet may cause the pay card to appear as one of the forms of payment under the
19 funds tab discussed in FIGURE 16C. The user may also cancel importing of the pay card
20 as a funding source by selecting the cancel button 1555. When the pay card has been
21 added to the wallet, the user interface may be updated to indicate that the importing is
22 complete via the notification display 1556. The user may then access the wallet 1557 to
23 begin using the added pay card as a funding source.

1 **[00298]** FIGURES 15F-15G provide user interface diagrams illustrating example
2 aspects of a history mode of a virtual wallet application in some embodiments of the
3 RUAP. With reference to FIGURE 15F, in one embodiment, a user may select the
4 history mode 1561 to view a history of prior purchases and perform various actions on
5 those prior purchases. For example, a user may query on receipts of past transactions
6 to claim a restricted-account reimbursement.

7 **[00299]** The wallet application may query the storage areas in the mobile device or
8 elsewhere (e.g., one or more databases and/or tables remote from the mobile device) for
9 prior transactions. The user interface may then display the results of the query such as
10 transactions 1563. The user interface may identify 1564: a type of the transaction (e.g.,
11 previously shopped for items, bills that have been captured by camera in a snap mode, a
12 person-to-person transfer; the date of the transaction; a description of the transaction,
13 including but not limited to: a cart name, cart contents indicator, total cost, merchant(s)
14 involved in the transaction; a link to obtain a shoptrail (explained further below in
15 greater detail), offers relating to the transaction, and any other relevant information. In
16 some implementation, any displayed transaction, coupon, bill, etc. may be added to a
17 cart for (re)purchase, 1565.

18 **[00300]** In one embodiment, a user may select the history mode 1571 to view a
19 history of filtered prior purchases and perform various actions on those prior purchases.
20 For example, a user may enter a merchant identifying information such as name,
21 product, MCC, and/or the like in the search bar 1572. In another implementation, the
22 user may use voice activated search feature to search the history. In another
23 implementations, the wallet application may display a pop up screen 1576, in which the

1 user may enter advanced search filters, keywords, and/or the like. The wallet application
2 may query the storage areas in the mobile device or elsewhere (e.g., one or more
3 databases and/or tables remote from the mobile device) for transactions matching the
4 search keywords. The user interface may then display the results of the query such as
5 transactions 1563. The user interface may identify 1574: a type of the transaction (e.g.,
6 previously shopped for items, bills that have been captured by camera in a snap mode, a
7 person-to-person transfer; the date of the transaction; a description of the transaction,
8 including but not limited to: a cart name, cart contents indicator, total cost, merchant(s)
9 involved in the transaction; a link to obtain a shoptrail (explained further below in
10 greater detail), offers relating to the transaction, and any other relevant information. In
11 some implementation, any displayed transaction, coupon, bill, etc. may be added to a
12 cart for (re)purchase, 1575.

13 **[00301]** With reference to FIGURE 15G, in one embodiment, the history mode may
14 also include facilities for exporting receipts. The export receipts pop up 1581 may
15 provide a number of options for exporting the receipts of transactions in the history. For
16 example, a user may use one or more of the options 1582, which include save (to local
17 mobile memory, to server, to a cloud account, and/or the like), print to a printer, fax,
18 email, and/or the like. The user may utilize his or her address book to look up email or
19 fax number for exporting. The user may also specify format options for exporting
20 receipts. Example format options may include, without limitation, text files
21 (.doc, .txt, .rtf, iif, etc.), spreadsheet (.csv, .xls, etc.), image files (.jpg, .tff, .png, etc.),
22 portable document format (.pdf), postscript (.ps), and/or the like. The user may then
23 click or tap the export button to initiate export of receipts.

1 **[00302]** FIGURES 15H-I provide exemplary mobile UIs showing augmented reality
2 receipt capturing within embodiments of the RUAP. In one implementation, a user may
3 operate a camera enabled device to capture a view of a receipt 1561, and obtain
4 augmented reality labels 1562 indicating items that are eligible for restricted-use
5 accounts. For example, the RUAP wallet component may perform an instant OCR to
6 extract item information and determine items such as "NyQuil" is eligible for
7 FSA/HSA/HRA 1564 usage, and grocery/food items are eligible for food stamp 1562
8 usage. In one implementation, if the user tap on the displayed account, the RUAP may
9 generate a virtual receipt (e.g., similar to that in FIGURE15C) and proceed to process
10 reimbursement request with the selected restricted-use account.

11 **[00303]** In further implementation, if the RUAP does not automatically determine
12 an item as eligible for any restricted-use accounts, e.g., an "Ester-C" supplement, a user
13 may tap on the screen to select it, and may view a list of accounts 1563 to select a user
14 desired reallocation account, e.g., any restricted-use account, loyalty account, and/or
15 the like.

16 **[00304]** In further implementations, the RUAP may identify a payment account
17 that has been used to fulfill the transaction associated with the receipt, e.g., a Visa
18 account 1566a, and/or obtain account information from the barcode printed on the
19 receipt 1566b. In one implementation, the RUAP may match the "*1234" Visa account
20 with any of user's enrolled account in the wallet, and recommend the user to reimburse
21 funds into an identified "Visa *1234" account if such account is identified from the
22 wallet 1565. In another implementation, the RUAP may prompt the user to select other
23 accounts for depositing reimbursement funds 1565.

1 **[00305]** Continuing on with FIGURE 15I, if the user has tapped on an account, e.g.,
2 "FSA" at 1564 in FIGURE 15H to reimburse an eligible item, the RUAP may generate a
3 reimbursement request 1571, e.g., showing the user is going to reimburse "NyQuil
4 Lipcap" 1572 from the selected "FSA *123" account 1573. In one implementation, the
5 user may indicate an account for depositing the reimbursement funds, e.g., the "Visa
6 *1234" 1574 account auto-identified from the receipt (e.g., at 1566a-b in FIGURE 15H),
7 and/or select other accounts.

8 **[00306]** In another implementation, if the user selects to tap on 1563 in FIGURE
9 15H to reimburse "Ester-C" 1575 for "FSA *123" account 1576, as the RUAP does not
10 identify "Ester-C" as an eligible FSA item, the RUAP may generate a reimbursement
11 request but with a notification to the user that such reimbursement is subject to FSA
12 review and may not be approved 1578.

13 **[00307]** FIGURES 16A-16E provide exemplary mobile wallet UIs illustrating wallet
14 account within embodiments of the RUAP. With reference to FIGURE 16A, in some
15 embodiments, a mobile device 1600 of the user may present a graphical user interface
16 (GUI) 1601 (e.g., a home screen) including a GUI element 1602 to start up a virtual
17 wallet application (e.g., an Apple® iPhone/iPad app, Google Android application,
18 Windows® Mobile application, etc.). For example, the icon 1602 may indicate the
19 wallet is enabled with RUAP service and the wallet has registered a RUAP prepaid
20 account.

21 **[00308]** In some embodiments, when a user activates the GUI element 1601 of
22 FIGURE 16A, the mobile device may provide a virtual mobile wallet application
23 interface 1610. In some embodiments, the application interface may include a GUI

1 element 1611 to initiate a payment communication (e.g., transmitting a
2 credit/debit/prepaid card number via NFC/Bluetooth/cellular communication). In
3 some embodiments, the mobile device may utilize default values for the payment
4 information (e.g., credit/debit/prepaid card information) and communication mode
5 (e.g., NFC). In alternate embodiments, the user may be able to modify the payment
6 information prior to communicating with a PoS terminal to initiate the payment
7 transaction. For example, a GUI element 1614 may provide a mechanism to modify
8 payment information without leaving the “tap to pay” screen provided by application
9 interface 1610 (see, e.g., elements 1620-221 of FIGURE 16B). As another example, a
10 GUI element 1613 may, when activated by the user, present the user an application
11 interface wherein the user may make more detailed adjustments to aspects of the
12 payment mechanism (e.g., card utilized, anonymization/security preferences, etc.). In
13 some embodiments, the user may be able to quickly revert to utilizing default payment
14 settings by activating a GUI element 1612. In some embodiments, the user may be able
15 to stop a communication of payment information that is in progress by activating a GUI
16 element 1615 (“tap to stop”) that the application interface presents during the
17 communication of payment information.

18 **[00309]** With reference to FIGURE 16B, in some embodiments, the user may
19 activate a GUI element 1620 when operating the virtual mobile wallet application in a
20 payment activation application interface to obtain a menu of card selection options
21 1621a-c. For example, the user may add a RUAP prepaid account 1621a to the wallet
22 upon obtaining a card number. The user may activate any of the card selection options
23 to quickly modify the payment information utilized in the communication to trigger the
24 payment transaction. In some embodiments, the user may activate GUI element 1622 to

1 obtain an application interface 1623 (“my accounts”) for a more detailed view, from
2 which the user can make selections of payment options. For example, the wallet
3 accounts 1623 may include a tab for the user's enrolled restricted use accounts. For
4 example, a GUI element 1624 may describe types of payment options (e.g., payment
5 cards, loyalty cards, NFC tags, etc.) available to the user. The specific payment options
6 may be depicted in GUI elements 1625a-b. In some embodiments, the GUI elements
7 1625a-b may be arranged as a column browser, with multiple columns (see 1626). In
8 some embodiments, the interface may provide a GUI element 1627 that the user can
9 activate to add a new payment option to the existing payment options displayed in the
10 GUI elements 1625a-b. For example, as shown at 1625b, the RUAP may list the user's
11 FSA, HSA, LOC accounts enrolled in the wallet with its balance information.

12 **[00310]** With reference to FIGURE 16C, in some embodiments, activating a GUI
13 element corresponding to a payment options may provide a detailed view of the
14 payment option, e.g., 1630 (“manage my card”). The view may include a graphical view
15 1631a of the payment option, providing information including, without limitation: card
16 payment processor (e.g., “Visa”), card number, payment mechanism (e.g., “Visa
17 payWave”), cardholder name, expiration date, and/or the like. The view may also
18 include indications of information such as, without limitation: a current balance 1632, a
19 maximum payment amount currently permissible using the card, a date on which a
20 balance payment is due, and/or the like. The view may include a GUI element 1633 that
21 the user can activate to utilize the payment option in the purchase transaction initiation.
22 In some embodiments, the view may include a GUI element 1634 that the user can
23 activate to view recent purchase transactions initiated using the payment option
24 currently being displayed in 1631a (e.g., a FSA account, etc.). The view may also include

1 a GUI element 1635 that the user can activate to add funds to the payment option
2 currently being displayed in 1631a. In some embodiments, the payment options may be
3 arranged within a column browser interface, such that user can scan through the various
4 payment options (e.g., 1631b-e) by swiping a touchscreen of the mobile device (see
5 1636a). As the user scans through the payment options, GUI element 1636b-e may
6 modify to indicate the user's position within the column browser interface.

7 **[00311]** With reference to FIGURE 16D, the user may be able to view a record of
8 recent transactions 1640 initiated using a payment option (e.g., by activating GUI
9 element 1634 of FIGURE 16C). In the recent transactions view 1640, the user may be
10 provided with a record listing 1645, including information on a time of a purchase
11 transaction ("when" 1641), a description of the transaction ("what" 1642), an amount of
12 the transaction ("amount" 1643), and a location of the transaction ("where" 1644). The
13 user may be able to scroll through a long list of recent transactions by activating a GUI
14 element 1646 ("view more"). In some embodiments, the user may activate a GUI
15 element 1647 to obtain a view of transactions placed on a geographical map, so that the
16 user may understand better where each of the user's transactions originate.

17 **[00312]** FIGURE 16E shows user interface diagrams illustrating example aspects of
18 allocating funds for a purchase payment within a virtual wallet application in some
19 embodiments of the RUAP. In one embodiment, the wallet mobile application may
20 provide a user with a number of options for paying for a transaction via the wallet mode
21 1681. The wallet mode may facilitate a user to set preferences for a payment transaction,
22 including settings funds sources 1682, payee 1683, transaction modes 1684, applying

1 real-time offers to the transaction 1685, and publishing the transaction details socially
2 1686, as described in further detail below.

3 **[00313]** In one implementation, an example user interface 1691 for making a
4 payment is shown. The user interface may clearly identify the amount 1692 and the
5 currency 1693 for the transaction. The amount may be the amount payable and the
6 currency may include real currencies such as dollars and euros, as well as virtual
7 currencies such as reward points. The user may select the funds tab 1682 to select one or
8 more forms of payment 1697, which may include various credit, debit, gift, rewards
9 and/or prepaid cards. The user may also have the option of paying, wholly or in part,
10 with reward points. For example, the graphical indicator 1698 on the user interface
11 shows the number of points available, the graphical indicator 1699 shows the number of
12 points to be used towards the amount due 234.56 and the equivalent 16000 of the
13 number of points in a selected currency (USD, for example).

14 **[00314]** In one implementation, the user may combine funds from multiple
15 sources to pay for the transaction. The amount 1695 displayed on the user interface may
16 provide an indication of the amount of total funds covered so far by the selected forms of
17 payment (e.g., Discover card and rewards points). The user may choose another form of
18 payment or adjust the amount to be debited from one or more forms of payment until
19 the amount 1695 matches the amount payable 1694. Once the amounts to be debited
20 from one or more forms of payment are finalized by the user, payment authorization
21 may begin.

22 **[00315]** In one implementation, the user may select a secure authorization of the
23 transaction by selecting the cloak button 16002 to effectively cloak or anonymize some

1 (e.g., pre-configured) or all identifying information such that when the user selects pay
2 button 16001, the transaction authorization is conducted in a secure and anonymous
3 manner. In another implementation, the user may select the pay button 16001 which
4 may use standard authorization techniques for transaction processing. In yet another
5 implementation, when the user selects the social button 16003, a message regarding the
6 transaction may be communicated to one of more social networks (set up by the user),
7 which may post or announce the purchase transaction in a social forum such as a wall
8 post or a tweet. In one implementation, the user may select a social payment processing
9 option 16003. The indicator 16004 may show the authorizing and sending social share
10 data in progress.

11 **[00316]** In another implementation, a restricted payment mode 16005 may be
12 activated for certain purchase activities such as prescription purchases. The mode may
13 be activated in accordance with rules defined by issuers, insurers, merchants, payment
14 processor and/or other entities to facilitate processing of specialized goods and services.
15 In this mode, the user may scroll down the list of forms of payments 16006 under the
16 funds tab to select specialized accounts such as a FSA, HSA 16007, and/or the like and
17 amounts to be debited to the selected accounts. In one implementation, such restricted
18 payment mode 16005 processing may disable social sharing of purchase information.

19 **[00317]** In one embodiment, the wallet mobile application may facilitate importing
20 of funds via the import funds user interface 16008. For example, a user who is
21 unemployed may obtain unemployment benefit fund 16009 via the wallet mobile
22 application. In one implementation, the entity providing the funds may also configure
23 rules for using the fund as shown by the processing indicator message 16100. The wallet

1 may read and apply the rules prior, and may reject any purchases with the
2 unemployment funds that fail to meet the criteria set by the rules. Example criteria may
3 include, for example, merchant category code (MCC), time of transaction, location of
4 transaction, and/or the like. As an example, a transaction with a grocery merchant
5 having MCC 5411 may be approved, while a transaction with a bar merchant having an
6 MCC 5813 may be refused.

7 **[00318]** FIGURES 17A-B show data flow diagrams illustrating an example
8 purchase transaction authorization procedure in some embodiments of the RUAP. With
9 reference to FIGURE 17A, in some embodiments, a user, e.g., 1701a, may wish to utilize
10 a virtual wallet account to purchase a product, service, offering, and/or the like
11 (“product”), from a merchant via a merchant online site or in the merchant’s store. The
12 user may utilize a physical card, or a user wallet device, e.g., 1701b, to access the user’s
13 virtual wallet account. For example, the user wallet device may be a personal/laptop
14 computer, cellular telephone, smartphone, tablet, eBook reader, netbook, gaming
15 console, and/or the like. The user may provide a wallet access input, e.g., 1711 into the
16 user wallet device. For example, if the user attempts to submit a healthcare payment,
17 the user may tap on the restricted-use account list, and the wallet may return a list of
18 restricted use accounts with updated balance information, e.g., see 585, 591-592 in
19 FIGURES 5E-5F.

20 **[00319]** In various embodiments, the user input may include, but not be limited to:
21 a single tap (e.g., a one-tap mobile app purchasing embodiment) of a touchscreen
22 interface, keyboard entry, card swipe, activating a RFID/NFC enabled hardware device
23 (e.g., electronic card having multiple accounts, smartphone, tablet, etc.) within the user

1 device, mouse clicks, depressing buttons on a joystick/game console, voice commands,
2 single/multi-touch gestures on a touch-sensitive interface, touching user interface
3 elements on a touch-sensitive display, and/or the like. In some embodiments, the user
4 wallet device may authenticate the user based on the user's wallet access input, and
5 provide virtual wallet features for the user.

6 **[00320]** In some embodiments, upon authenticating the user for access to virtual
7 wallet features, the user wallet device may provide a transaction authorization input, e.g.,
8 1714, to a point-of-sale ("PoS") client, e.g., 1702. For example, the user wallet device
9 may communicate with the PoS client via Bluetooth, Wi-Fi, cellular communication,
10 one- or two-way near-field communication ("NFC"), and/or the like. In embodiments
11 where the user utilizes a plastic card instead of the user wallet device, the user may
12 swipe the plastic card at the PoS client to transfer information from the plastic card into
13 the PoS client. For example, the PoS client may obtain, as transaction authorization
14 input 1714, track 1 data from the user's plastic card (e.g., credit card, debit card, prepaid
15 card, charge card, etc.), such as the example track 1 data provided below:

16 %B123456789012345^PUBLIC/J.Q.^99011200000000000000**901*****?*
17 (wherein '123456789012345' is the card number of 'J.Q. Public' and has a CVV number of
18 901. '990112' is a service code, and *** represents decimal digits which change
19 randomly each time the card is used.)
20
21

22 **[00321]** In embodiments where the user utilizes a user wallet device, the user
23 wallet device may provide payment information to the PoS client, formatted according
24 to a data formatting protocol appropriate to the communication mechanism employed
25 in the communication between the user wallet device and the PoS client. An example
26 listing of transaction authorization input 1714, substantially in the form of XML-
27 formatted data, is provided below:

```

1 <?XML version = "1.0" encoding = "UTF-8"?>
2 <transaction_authorization_input>
3     <payment_data>
4         <account>
5             <charge_priority>1</charge_priority>
6             <charge_ratio>40%</charge_ratio>
7             <account_number>123456789012345</account_number>
8             <account_name>John Smith</account_name>
9             <bill_add>987 Green St #456, Chicago, IL 94652</bill_add>
10            <ship_add>987 Green St #456, Chicago, IL 94652</ship_add>
11            <CVV>123</CVV>
12        </account>
13        <account>
14            <charge_priority>1</charge_priority>
15            <charge_ratio>60%</charge_ratio>
16            <account_number>234567890123456</account_number>
17            <account_name>John Smith</account_name>
18            <bill_add>987 Green St #456, Chicago, IL 94652</bill_add>
19            <ship_add>987 Green St #456, Chicago, IL 94652</ship_add>
20            <CVV>173</CVV>
21        </account>
22        <account>
23            <charge_priority>2</charge_priority>
24            <charge_ratio>100%</charge_ratio>
25            <account_number>345678901234567</account_number>
26            <account_name>John Smith</account_name>
27            <bill_add>987 Green St #456, Chicago, IL 94652</bill_add>
28            <ship_add>987 Green St #456, Chicago, IL 94652</ship_add>
29            <CVV>695</CVV>
30        </account>
31    </payment_data>
32    <!--optional data-->
33    <timestamp>2011-02-22 15:22:43</timestamp>
34    <expiry_lapse>00:00:30</expiry_lapse>
35    <secure_key>0445329070598623487956543322</secure_key>
36    <alerts_track_flag>TRUE</alerts_track_flag>
37    <wallet_device_details>
38        <device_IP>192.168.23.126</client_IP>
39        <device_type>smartphone</client_type>
40        <device_model>HTC Hero</client_model>
41        <OS>Android 2.2</OS>
42        <wallet_app_installed_flag>true</wallet_app_installed_flag>
43    </wallet_device_details>
44 </transaction_authorization_input>
45
46

```

[00322] In some embodiments, the PoS client may generate a card authorization request, e.g., 1715, using the obtained transaction authorization input from the user wallet device, and/or product/checkout data. An example listing of a card authorization request 1715, substantially in the form of a HTTP(S) POST message including XML-formatted data, is provided below:

```

52 POST /authorizationrequests.php HTTP/1.1
53 Host: www.acquirer.com
54 Content-Type: Application/XML

```

```

1 Content-Length: 1306
2 <?XML version = "1.0" encoding = "UTF-8"?>
3 <card_authorization_request>
4     <session_ID>4NFU4RG94</order_ID>
5     <timestamp>2011-02-22 15:22:43</timestamp>
6     <expiry>00:00:30</expiry>
7     <alerts_URL>www.merchant.com/shopcarts.php?sessionID=AEBB4356</alerts_URL>
8     <!--optional data-->
9     <user_ID>john.q.public@gmail.com</user_ID>
10    <PoS_details>
11        <PoS_IP>192.168.23.126</client_IP>
12        <PoS_type>smartphone</client_type>
13        <PoS_model>HTC Hero</client_model>
14        <OS>Android 2.2</OS>
15        <app_installed_flag>true</app_installed_flag>
16    </PoS_details>
17    <purchase_details>
18
19    <Item1>
20        <ItemCode> FOOD-9875 </ItemCode>
21        <Category> FOOD </Category>
22        <Sub-Category> Breakfast </Sub-Category>
23        </ItemName> Cereal </ItemName>
24        <Description> whole grain original 10 oz </Description>
25        <Quantity> 1 </Quantity>
26        <UnitPrice> 4.99 </UnitPrice>
27        <TotalAmt> 4.99 </TotalAmt>
28        ...
29    </Item1>
30    <Item2>
31        <ItemCode> DRUG-23401 </ItemCode>
32        <Category> DRUG </Category>
33        <Sub-Category> Non-Prescription </Sub-Category>
34        </ItemName> NyQuil Cold Medicine </ItemName>
35        <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
36        <Quantity> 1 </Quantity>
37        <UnitPrice> 12.99 </UnitPrice>
38        <TotalAmt> 12.99 </TotalAmt>
39        ...
40    </Item2>
41    <item_3>
42        <ItemCode> SU-Shampoo-001 </ItemCode>
43        <Category> WASH </Category>
44        <Sub-Category> hair product </Sub-Category>
45        </ItemName> SUAVE shampoo </ItemName>
46        <Description> SUAVE shampoo heat treatment 120 mL </Description>
47        <Quantity> 1 </Quantity>
48        <UnitPrice> 8.99 </UnitPrice>
49        <TotalAmt> 8.99 </TotalAmt>
50        ...
51    </item_3>
52 ...
53    </purchase_details>
54    <merchant_params>
55        <merchant_id>3FBCR4INC</merchant_id>
56        <merchant_name>Acme Supermarket, Inc.</merchant_name>
57        <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
58    </merchant_params>
59    <account_params>
60        <account_name>John Smith</account_name>
61        <account_type>credit</account_type>
62        <account_num>123456789012345</account_num>
63        <billing_address>123 Green St., Norman, OK 98765</billing_address>

```

```
1         <phone>123-456-7809</phone>
2         <sign>/jqp</sign>
3         <confirm_type>email</confirm_type>
4         <contact_info>john.q.public@gmail.com</contact_info>
5     </account_params>
6     <shipping_info>
7         <shipping_adress>same as billing</shipping_adress>
8         <ship_type>expedited</ship_type>
9         <ship_carrier>FedEx</ship_carrier>
10        <ship_account>123-45-678</ship_account>
11        <tracking_flag>true</tracking_flag>
12        <sign_flag>>false</sign_flag>
13    </shipping_info>
14 </card_authorization_request>
15
16
```

17 **[00323]** In some embodiments, the card authorization request generated by the
18 user device may include a minimum of information to process the purchase transaction.
19 For example, this may improve the efficiency of communicating the purchase
20 transaction request, and may also advantageously improve the privacy protections
21 provided to the user and/or merchant. For example, in some embodiments, the card
22 authorization request may include at least a session ID for the user's shopping session
23 with the merchant. The session ID may be utilized by any component and/or entity
24 having the appropriate access authority to access a secure site on the merchant server to
25 obtain alerts, reminders, and/or other data about the transaction(s) within that
26 shopping session between the user and the merchant. In some embodiments, the PoS
27 client may provide the generated card authorization request to the merchant server, e.g.,
28 1716. The merchant server may forward the card authorization request to a pay gateway
29 server, e.g., 1704a, for routing the card authorization request to the appropriate
30 payment network for payment processing. For example, the pay gateway server may be
31 able to select from payment networks, such as Visa, Mastercard, American Express,
32 Paypal, etc., to process various types of transactions including, but not limited to: credit
33 card, debit card, prepaid card, B2B and/or like transactions. In some embodiments, the
34 merchant server may query a database, e.g., merchant/acquirer database 1703b, for a

1 network address of the payment gateway server, for example by using a portion of a user
2 payment card number, or a user ID (such as an email address) as a keyword for the
3 database query. For example, the merchant server may issue PHP/SQL commands to
4 query a database table (such as FIGURE 24, Pay Gateways 2419h) for a URL of the pay
5 gateway server. An example payment gateway address query 1717, substantially in the
6 form of PHP/SQL commands, is provided below:

```
7 <?PHP
8 header('Content-Type: text/plain');
9 mysql_connect("254.93.179.112", $DBserver, $password); // access database server
10 mysql_select_db("RUAP_DB.SQL"); // select database table to search
11 //create query
12 $query = "SELECT paygate_id paygate_address paygate_URL paygate_name FROM
13 PayGatewayTable WHERE card_num LIKE '%" $cardnum";
14 $result = mysql_query($query); // perform the search query
15 mysql_close("RUAP_DB.SQL"); // close database access
16 ?>
17
18
```

19 **[00324]** In response, the merchant/acquirer database may provide the requested
20 payment gateway address, e.g., 1718. The merchant server may forward the card
21 authorization request to the pay gateway server using the provided address, e.g., 1719.
22 In some embodiments, upon receiving the card authorization request from the
23 merchant server, the pay gateway server may invoke a component to provide one or
24 more services associated with purchase transaction authorization. For example, the pay
25 gateway server may invoke components for fraud prevention, loyalty and/or rewards,
26 and/or other services for which the user-merchant combination is authorized. The pay
27 gateway server may forward the card authorization request to a healthcare collection
28 portal server, e.g., 1705a, for payment processing. For example, the pay gateway server
29 may be able to select from payment networks, such as Visa, Mastercard, American
30 Express, Paypal, etc., to process various types of transactions including, but not limited
31 to: credit card, debit card, prepaid card, B2B and/or like transactions. In some

1 embodiments, the pay gateway server may query a database, e.g., pay gateway database
2 1704b, for a network address of the payment network server, for example by using a
3 portion of a user payment card number, or a user ID (such as an email address) as a
4 keyword for the database query. For example, the pay gateway server may issue
5 PHP/SQL commands to query a database table (such as FIGURE 24, Pay Gateways
6 2419h) for a URL of the healthcare collection portal server. An example payment
7 network address query 1721, substantially in the form of PHP/SQL commands, is
8 provided below:

```
9 <?PHP
10 header('Content-Type: text/plain');
11 mysql_connect("254.93.179.112", $DBserver, $password); // access database server
12 mysql_select_db("RUAP_DB.SQL"); // select database table to search
13 //create query
14 $query = "SELECT payNET_id payNET_address payNET_URL payNET_name FROM PayGatewayTable
15 WHERE card_num LIKE '%" . $cardnum . "%";
16 $result = mysql_query($query); // perform the search query
17 mysql_close("RUAP_DB.SQL"); // close database access
18 ?>
19
20
```

21 **[00325]** In response, the payment gateway database may provide the requested
22 payment network address, e.g., 1722. The pay gateway server may forward the card
23 authorization request to the healthcare collection portal server using the provided
24 address, e.g., 1723.

25 **[00326]** With reference to FIGURE 17B, in some embodiments, the healthcare
26 collection portal server may process the transaction so as to transfer funds for the
27 purchase into an account stored on an acquirer of the merchant. For example, the
28 acquirer may be a financial institution maintaining an account of the merchant. For
29 example, the proceeds of transactions processed by the merchant may be deposited into
30 an account maintained by at a server of the acquirer.

1 **[00327]** In some embodiments, the healthcare collection portal server may
 2 generate a query, e.g., 1724, for issuer server(s) corresponding to the user-selected
 3 payment options. For example, the user's account may be linked to one or more issuer
 4 financial institutions ("issuers"), such as banking institutions, which issued the
 5 account(s) for the user. For example, such accounts may include, but not be limited to:
 6 credit card, debit card, prepaid card, checking, savings, money market, certificates of
 7 deposit, stored (cash) value accounts and/or the like. Issuer server(s), e.g., 1706a, of the
 8 issuer(s) may maintain details of the user's account(s). In some embodiments, a
 9 database, e.g., healthcare collection portal database 1705b, may store details of the
 10 issuer server(s) associated with the issuer(s). In some embodiments, the healthcare
 11 collection portal server may query a database, e.g., healthcare collection portal database
 12 1705b, for a network address of the issuer(s) server(s), for example by using a portion of
 13 a user payment card number, or a user ID (such as an email address) as a keyword for
 14 the database query. For example, the merchant server may issue PHP/SQL commands
 15 to query a database table (such as FIGURE 24, Issuers 2419f) for network address(es) of
 16 the issuer(s) server(s). An example issuer server address(es) query 1724, substantially
 17 in the form of PHP/SQL commands, is provided below:

```

18 <?PHP
19 header('Content-Type: text/plain');
20 mysql_connect("254.93.179.112", $DBserver, $password); // access database server
21 mysql_select_db("RUAP_DB.SQL"); // select database table to search
22 //create query
23 $query = "SELECT issuer_id issuer_address issuer_URL issuer_name FROM IssuersTable
24 WHERE card_num LIKE '%" . $cardnum . "%";
25 $result = mysql_query($query); // perform the search query
26 mysql_close("RUAP_DB.SQL"); // close database access
27 ?>
28
29

```

30 **[00328]** In response to obtaining the issuer server query, e.g., 1724, the healthcare
 31 collection portal database may provide, e.g., 1725, the requested issuer server data to the

1 healthcare collection portal server. In some embodiments, the healthcare collection
2 portal server may utilize the issuer server data to generate funds authorization
3 request(s), e.g., 1726, for each of the issuer server(s) selected based on the pre-defined
4 payment settings associated with the user's virtual wallet, and/or the user's payment
5 options input, and provide the funds authorization request(s) to the issuer server(s). In
6 some embodiments, the funds authorization request(s) may include details such as, but
7 not limited to: the costs to the user involved in the transaction, card account details of
8 the user, user billing and/or shipping information, and/or the like. An example listing
9 of a funds authorization request 1726, substantially in the form of a HTTP(S) POST
10 message including XML-formatted data, is provided below:

```
11 POST /fundsauthorizationrequest.php HTTP/1.1
12 Host: www.issuer.com
13 Content-Type: Application/XML
14 Content-Length: 624
15 <?XML version = "1.0" encoding = "UTF-8"?>
16 <funds_authorization_request>
17     <query_ID>VNEI39FK</query_ID>
18     <timestamp>2011-02-22 15:22:44</timestamp>
19     <transaction_cost>$22.61</transaction_cost>
20     <account_params>
21         <account_type>checking</account_type>
22         <account_num>1234567890123456</account_num>
23     </account_params>
24     <!--optional parameters-->
25     <purchase_summary>
26         <num_products>1</num_products>
27     <Item1>
28         <ItemCode> FOOD-9875 </ItemCode>
29         <Category> FOOD </Category>
30         <Sub-Category> Breakfast </Sub-Category>
31         </ItemName> Cereal </ItemName>
32         <Description> whole grain original 10 oz </Description>
33         <Quantity> 1 </Quantity>
34         <UnitPrice> 4.99 </UnitPrice>
35         <TotalAmt> 4.99 </TotalAmt>
36         ...
37     </Item1>
38     <Item2>
39         <ItemCode> DRUG-23401 </ItemCode>
40         <Category> DRUG </Category>
41         <Sub-Category> Non-Prescription </Sub-Category>
42         </ItemName> NyQuil Cold Medicine </ItemName>
43         <Description> NyQuil Cold&Flu Liquid 80 mL </Description>
44         <Quantity> 1 </Quantity>
45         <UnitPrice> 12.99 </UnitPrice>
46         <TotalAmt> 12.99 </TotalAmt>
```

```

1      ...
2      </Item2>
3      <item_3>
4          <ItemCode> SU-Shampoo-001 </ItemCode>
5          <Category> WASH </Category>
6          <Sub-Category> hair product </Sub-Category>
7          </ItemName> SUAVE shampoo </ItemName>
8          <Description> SUAVE shampoo heat treatment 120 mL </Description>
9          <Quantity> 1 </Quantity>
10         <UnitPrice> 8.99 </UnitPrice>
11         <TotalAmt> 8.99 </TotalAmt>
12     ...
13 </item_3>
14 </purchase_summary>
15 <merchant_params>
16     <merchant_id>3FBCR4INC</merchant_id>
17     <merchant_name>Acme Supermarket, Inc.</merchant_name>
18     <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
19 </merchant_params>
20 </funds_authorization_request>
21
22

```

23 **[00329]** In some embodiments, an issuer server may parse the authorization
24 request(s), and based on the request details may query a database, e.g., user profile
25 database 1706b, for data associated with an account linked to the user. For example, the
26 merchant server may issue PHP/SQL commands to query a database table (such as
27 FIGURE 24, Accounts 2419d) for user account(s) data. An example user account(s)
28 query 1727, substantially in the form of PHP/SQL commands, is provided below:

```

29 <?PHP
30 header('Content-Type: text/plain');
31 mysql_connect("254.93.179.112", $DBserver, $password); // access database server
32 mysql_select_db("RUAP_DB.SQL"); // select database table to search
33 //create query
34 $query = "SELECT issuer user_id user_name user_balance account_type FROM AccountsTable
35 WHERE account_num LIKE '%" . $accountnum . "'";
36 $result = mysql_query($query); // perform the search query
37 mysql_close("RUAP_DB.SQL"); // close database access
38 ?>
39
40

```

41 **[00330]** In some embodiments, on obtaining the user account(s) data, e.g., 1728,
42 the issuer server may determine whether the user can pay for the transaction using
43 funds available in the account, 1729. For example, the issuer server may determine
44 whether the user has a sufficient balance remaining in the account, sufficient credit
45 associated with the account, and/or the like. Based on the determination, the issuer

1 server(s) may provide a funds authorization response, e.g., 1730, to the healthcare
2 collection portal server. For example, the issuer server(s) may provide a HTTP(S) POST
3 message similar to the examples above. In some embodiments, if at least one issuer
4 server determines that the user cannot pay for the transaction using the funds available
5 in the account, the healthcare collection portal server may request payment options
6 again from the user (e.g., by providing an authorization fail message to the user device
7 and requesting the user device to provide new payment options), and re-attempt
8 authorization for the purchase transaction. In some embodiments, if the number of
9 failed authorization attempts exceeds a threshold, the healthcare collection portal server
10 may abort the authorization process, and provide an “authorization fail” message to the
11 merchant server, user device and/or client.

12 **[00331]** In some embodiments, the healthcare collection portal server may obtain
13 the funds authorization response including a notification of successful authorization,
14 and parse the message to extract authorization details. Upon determining that the user
15 possesses sufficient funds for the transaction, e.g., 1731, the healthcare collection portal
16 server may invoke a component to provide value-add services for the user.

17 **[00332]** In some embodiments, the healthcare collection portal server may
18 generate a transaction data record from the authorization request and/or authorization
19 response, and store the details of the transaction and authorization relating to the
20 transaction in a transactions database. For example, the healthcare collection portal
21 server may issue PHP/SQL commands to store the data to a database table (such as
22 FIGURE 24, Transactions 2419i). An example transaction store command, substantially
23 in the form of PHP/SQL commands, is provided below:

```

1 <?PHP
2 header('Content-Type: text/plain');
3 mysql_connect("254.92.185.103",$DBserver,$password); // access database server
4 mysql_select("RUAP_DB.SQL"); // select database to append
5 mysql_query("INSERT INTO TransactionsTable (PurchasesTable (timestamp,
6 purchase_summary_list, num_products, product_summary, product_quantity,
7 transaction_cost, account_params_list, account_name, account_type, account_num,
8 billing_addres, zipcode, phone, sign, merchant_params_list, merchant_id, merchant_name,
9 merchant_auth_key)
10 VALUES (time(), $purchase_summary_list, $num_products, $product_summary,
11 $product_quantity, $transaction_cost, $account_params_list, $account_name,
12 $account_type, $account_num, $billing_addres, $zipcode, $phone, $sign,
13 $merchant_params_list, $merchant_id, $merchant_name, $merchant_auth_key)"); // add
14 data to table in database
15 mysql_close("RUAP_DB.SQL"); // close connection to database
16 ?>
17
18

```

19 **[00333]** In some embodiments, the healthcare collection portal server may forward
20 a transaction authorization response, e.g., 1732, to the user wallet device, PoS client,
21 and/or merchant server. The merchant may obtain the transaction authorization
22 response, and determine from it that the user possesses sufficient funds in the card
23 account to conduct the transaction. The merchant server may add a record of the
24 transaction for the user to a batch of transaction data relating to authorized transactions.
25 For example, the merchant may append the XML data pertaining to the user transaction
26 to an XML data file comprising XML data for transactions that have been authorized for
27 various users, e.g., 1733, and store the XML data file, e.g., 1734, in a database, e.g.,
28 merchant database 404. For example, a batch XML data file may be structured similar
29 to the example XML data structure template provided below:

```

30 <?XML version = "1.0" encoding = "UTF-8"?>
31 <merchant_data>
32     <merchant_id>3FBCR4INC</merchant_id>
33     <merchant_name>Acme Supermarket, Inc.</merchant_name>
34     <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
35     <account_number>123456789</account_number>
36 </merchant_data>
37 <transaction_data>
38     <transaction 1>
39         ...
40     </transaction 1>
41     <transaction 2>
42         ...
43     </transaction 2>
44     .

```

```
1      .  
2      .  
3      <transaction n>  
4      ...  
5      </transaction n>  
6 </transaction_data>  
7  
8
```

9 **[00334]** In some embodiments, the server may also generate a purchase receipt,
10 e.g., 1733, and provide the purchase receipt to the client, e.g., 1735. The client may
11 render and display, e.g., 1736, the purchase receipt for the user. In some embodiments,
12 the user's wallet device may also provide a notification of successful authorization to the
13 user. For example, the PoS client/user device may render a webpage, electronic
14 message, text / SMS message, buffer a voicemail, emit a ring tone, and/or play an audio
15 message, etc., and provide output including, but not limited to: sounds, music, audio,
16 video, images, tactile feedback, vibration alerts (e.g., on vibration-capable client devices
17 such as a smartphone etc.), and/or the like.

18 **[00335]** FIGURES 18A-B show logic flow diagrams illustrating example aspects of
19 purchase transaction authorization in some embodiments of the RUAP, e.g., a Purchase
20 Transaction Authorization ("PTA") component. With reference to FIGURE 18A, in
21 some embodiments, a user may wish to utilize a virtual wallet account to purchase a
22 product, service, offering, and/or the like ("product"), from a merchant via a merchant
23 online site or in the merchant's store. The user may utilize a physical card, or a user
24 wallet device to access the user's virtual wallet account. For example, the user wallet
25 device may be a personal/laptop computer, cellular telephone, smartphone, tablet,
26 eBook reader, netbook, gaming console, and/or the like. The user may provide a wallet
27 access input, e.g., 1801, into the user wallet device. In various embodiments, the user
28 input may include, but not be limited to: a single tap (e.g., a one-tap mobile app
29 purchasing embodiment) of a touchscreen interface, keyboard entry, card swipe,

1 activating a RFID/NFC enabled hardware device (e.g., electronic card having multiple
2 accounts, smartphone, tablet, etc.) within the user device, mouse clicks, depressing
3 buttons on a joystick/game console, voice commands, single/multi-touch gestures on a
4 touch-sensitive interface, touching user interface elements on a touch-sensitive display,
5 and/or the like. In some embodiments, the user wallet device may authenticate the user
6 based on the user's wallet access input, and provide virtual wallet features for the user,
7 e.g., 1802-1803.

8 **[00336]** In some embodiments, upon authenticating the user for access to virtual
9 wallet features, the user wallet device may provide a transaction authorization input, e.g.,
10 1804, to a point-of-sale ("PoS") client. For example, the user wallet device may
11 communicate with the PoS client via Bluetooth, Wi-Fi, cellular communication, one- or
12 two-way near-field communication ("NFC"), and/or the like. In embodiments where
13 the user utilizes a plastic card instead of the user wallet device, the user may swipe the
14 plastic card at the PoS client to transfer information from the plastic card into the PoS
15 client. In embodiments where the user utilizes a user wallet device, the user wallet
16 device may provide payment information to the PoS client, formatted according to a
17 data formatting protocol appropriate to the communication mechanism employed in the
18 communication between the user wallet device and the PoS client.

19 **[00337]** In some embodiments, the PoS client may obtain the transaction
20 authorization input, and parse the input to extract payment information from the
21 transaction authorization input, e.g., 1805. For example, the PoS client may utilize a
22 parser, such as the example parsers provided below in the discussion with reference to
23 FIGURE 24. The PoS client may generate a card authorization request, e.g., 1806, using

1 the obtained transaction authorization input from the user wallet device, and/or
2 product/checkout data.

3 **[00338]** In some embodiments, the PoS client may provide the generated card
4 authorization request to the merchant server. The merchant server may forward the
5 card authorization request to a pay gateway server, for routing the card authorization
6 request to the appropriate payment network for payment processing. For example, the
7 pay gateway server may be able to select from payment networks, such as Visa,
8 Mastercard, American Express, Paypal, etc., to process various types of transactions
9 including, but not limited to: credit card, debit card, prepaid card, B2B and/or like
10 transactions. In some embodiments, the merchant server may query a database, e.g.,
11 1808, for a network address of the payment gateway server, for example by using a
12 portion of a user payment card number, or a user ID (such as an email address) as a
13 keyword for the database query. In response, the merchant/acquirer database may
14 provide the requested payment gateway address, e.g., 1810. The merchant server may
15 forward the card authorization request to the pay gateway server using the provided
16 address. In some embodiments, upon receiving the card authorization request from the
17 merchant server, the pay gateway server may invoke a component to provide one or
18 more service associated with purchase transaction authorization, e.g., 1811. For
19 example, the pay gateway server may invoke components for fraud prevention (see e.g.,
20 VerifyChat, FIGURE 3E), loyalty and/or rewards, and/or other services for which the
21 user-merchant combination is authorized.

22 **[00339]** The pay gateway server may forward the card authorization request to a
23 healthcare collection portal server for payment processing, e.g., 1814. For example, the

1 pay gateway server may be able to select from payment networks, such as Visa,
2 Mastercard, American Express, Paypal, etc., to process various types of transactions
3 including, but not limited to: credit card, debit card, prepaid card, B2B and/or like
4 transactions. In some embodiments, the pay gateway server may query a database, e.g.,
5 1812, for a network address of the payment network server, for example by using a
6 portion of a user payment card number, or a user ID (such as an email address) as a
7 keyword for the database query. In response, the payment gateway database may
8 provide the requested payment network address, e.g., 1813. The pay gateway server may
9 forward the card authorization request to the healthcare collection portal server using
10 the provided address, e.g., 1814.

11 **[00340]** With reference to FIGURE 18B, in some embodiments, the healthcare
12 collection portal server may process the transaction so as to transfer funds for the
13 purchase into an account stored on an acquirer of the merchant. For example, the
14 acquirer may be a financial institution maintaining an account of the merchant. For
15 example, the proceeds of transactions processed by the merchant may be deposited into
16 an account maintained by at a server of the acquirer. In some embodiments, the
17 healthcare collection portal server may generate a query, e.g., 1815, for issuer server(s)
18 corresponding to the user-selected payment options. For example, the user's account
19 may be linked to one or more issuer financial institutions ("issuers"), such as banking
20 institutions, which issued the account(s) for the user. For example, such accounts may
21 include, but not be limited to: credit card, debit card, prepaid card, checking, savings,
22 money market, certificates of deposit, stored (cash) value accounts and/or the like.
23 Issuer server(s) of the issuer(s) may maintain details of the user's account(s). In some
24 embodiments, a database, e.g., a healthcare collection portal database, may store details

1 of the issuer server(s) associated with the issuer(s). In some embodiments, the
2 healthcare collection portal server may query a database, e.g., 1815, for a network
3 address of the issuer(s) server(s), for example by using a portion of a user payment card
4 number, or a user ID (such as an email address) as a keyword for the database query.

5 **[00341]** In response to obtaining the issuer server query, the healthcare collection
6 portal database may provide, e.g., 1816, the requested issuer server data to the
7 healthcare collection portal server. In some embodiments, the healthcare collection
8 portal server may utilize the issuer server data to generate funds authorization
9 request(s), e.g., 1817, for each of the issuer server(s) selected based on the pre-defined
10 payment settings associated with the user's virtual wallet, and/or the user's payment
11 options input, and provide the funds authorization request(s) to the issuer server(s). In
12 some embodiments, the funds authorization request(s) may include details such as, but
13 not limited to: the costs to the user involved in the transaction, card account details of
14 the user, user billing and/or shipping information, and/or the like. In some
15 embodiments, an issuer server may parse the authorization request(s), e.g., 1818, and
16 based on the request details may query a database, e.g., 1819, for data associated with an
17 account linked to the user.

18 **[00342]** In some embodiments, on obtaining the user account(s) data, e.g., 1820,
19 the issuer server may determine whether the user can pay for the transaction using
20 funds available in the account, e.g., 1821. For example, the issuer server may determine
21 whether the user has a sufficient balance remaining in the account, sufficient credit
22 associated with the account, and/or the like. Based on the determination, the issuer
23 server(s) may provide a funds authorization response, e.g., 1822, to the healthcare

1 collection portal server. In some embodiments, if at least one issuer server determines
2 that the user cannot pay for the transaction using the funds available in the account, the
3 healthcare collection portal server may request payment options again from the user
4 (e.g., by providing an authorization fail message to the user device and requesting the
5 user device to provide new payment options), and re-attempt authorization for the
6 purchase transaction. In some embodiments, if the number of failed authorization
7 attempts exceeds a threshold, the healthcare collection portal server may abort the
8 authorization process, and provide an “authorization fail” message to the merchant
9 server, user device and/or client.

10 **[00343]** In some embodiments, the healthcare collection portal server may obtain
11 the funds authorization response including a notification of successful authorization,
12 and parse the message to extract authorization details. Upon determining that the user
13 possesses sufficient funds for the transaction, e.g., 1823, the healthcare collection portal
14 server may invoke a component to provide value-add services for the user, e.g., 1823.

15 **[00344]** In some embodiments, the healthcare collection portal server may forward
16 a transaction authorization response to the user wallet device, PoS client, and/or
17 merchant server. The merchant may parse, e.g., 1824, the transaction authorization
18 response, and determine from it that the user possesses sufficient funds in the card
19 account to conduct the transaction, e.g., 1825, option “Yes.” The merchant server may
20 add a record of the transaction for the user to a batch of transaction data relating to
21 authorized transactions. For example, the merchant may append the XML data
22 pertaining to the user transaction to an XML data file comprising XML data for
23 transactions that have been authorized for various users, e.g., 1826, and store the XML

1 data file, e.g., 1827, in a database. In some embodiments, the server may also generate a
2 purchase receipt, e.g., 1828, and provide the purchase receipt to the client. The client
3 may render and display, e.g., 1829, the purchase receipt for the user. In some
4 embodiments, the user's wallet device may also provide a notification of successful
5 authorization to the user. For example, the PoS client/user device may render a
6 webpage, electronic message, text / SMS message, buffer a voicemail, emit a ring tone,
7 and/or play an audio message, etc., and provide output including, but not limited to:
8 sounds, music, audio, video, images, tactile feedback, vibration alerts (e.g., on vibration-
9 capable client devices such as a smartphone etc.), and/or the like.

10 **[00345]** FIGURES 19A-B show data flow diagrams illustrating an example
11 purchase transaction clearance procedure in some embodiments of the RUAP. With
12 reference to FIGURE 19A, in some embodiments, a merchant server, e.g., 1903a, may
13 initiate clearance of a batch of authorized transactions. For example, the merchant
14 server may generate a batch data request, e.g., 1911, and provide the request, to a
15 merchant database, e.g., 1903b. For example, the merchant server may utilize
16 PHP/SQL commands similar to the examples provided above to query a relational
17 database. In response to the batch data request, the database may provide the
18 requested batch data, e.g., 1912. The server may generate a batch clearance request, e.g.,
19 1913, using the batch data obtained from the database, and provide, e.g., 1914, the batch
20 clearance request to an acquirer server, e.g., 1907a. For example, the merchant server
21 may provide a HTTP(S) POST message including XML-formatted batch data in the
22 message body for the acquirer server. The acquirer server may generate, e.g., 1915, a
23 batch payment request using the obtained batch clearance request, and provide, e.g.,
24 1918, the batch payment request to the healthcare collection portal server, e.g., 1905a.

1 The healthcare collection portal server may parse the batch payment request, and
2 extract the transaction data for each transaction stored in the batch payment request,
3 e.g., 1919. The healthcare collection portal server may store the transaction data, e.g.,
4 1920, for each transaction in a database, e.g., healthcare collection portal database
5 1905b. In some embodiments, the healthcare collection portal server may invoke a
6 component to provide value-add analytics services based on analysis of the transactions
7 of the merchant for whom the RUAP is clearing purchase transactions. For example, the
8 healthcare collection portal server may invoke a component such as the example card
9 transaction-based analytics component discussed above with reference to FIGURE 10.
10 Thus, in some embodiments, the healthcare collection portal server may provide
11 analytics-based value-added services for the merchant and/or the merchant's users.

12 **[00346]** With reference to FIGURE 19B, in some embodiments, for each extracted
13 transaction, the healthcare collection portal server may query, e.g., 1923, a database, e.g.,
14 healthcare collection portal database 1905b, for an address of an issuer server. For
15 example, the healthcare collection portal server may utilize PHP/SQL commands
16 similar to the examples provided above. The healthcare collection portal server may
17 generate an individual payment request, e.g., 1925, for each transaction for which it has
18 extracted transaction data, and provide the individual payment request, e.g., 1925, to
19 the issuer server, e.g., 1906a. For example, the healthcare collection portal server may
20 provide an individual payment request to the issuer server(s) as a HTTP(S) POST
21 message including XML-formatted data. An example listing of an individual payment
22 request 1925, substantially in the form of a HTTP(S) POST message including XML-
23 formatted data, is provided below:

```

1 POST /paymentrequest.php HTTP/1.1
2 Host: www.issuer.com
3 Content-Type: Application/XML
4 Content-Length: 788
5 <?XML version = "1.0" encoding = "UTF-8"?>
6 <pay_request>
7     <request_ID>CNI4ICNW2</request_ID>
8     <timestamp>2011-02-22 17:00:01</timestamp>
9     <pay_amount>$34.78</pay_amount>
10    <account_params>
11        <account_name>John Smith</account_name>
12        <account_type>credit</account_type>
13        <account_num>123456789012345</account_num>
14        <billing_address>123 Green St., Norman, OK 98765</billing_address>
15        <phone>123-456-7809</phone>
16        <sign>/jqp/</sign>
17    </account_params>
18    <merchant_params>
19        <merchant_id>3FBCR4INC</merchant_id>
20        <merchant_name>Acme Supermarket, Inc.</merchant_name>
21        <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
22    </merchant_params>
23    <purchase_summary>
24        <num_products>1</num_products>
25        <product>
26            <product_summary>Book - XML for dummies</product_summary>
27            <product_quantity>1</product_quantity?
28        </product>
29    </purchase_summary>
30 </pay_request>
31
32

```

[00347] In some embodiments, the issuer server may generate a payment command, e.g., 1927. For example, the issuer server may issue a command to deduct funds from the user's account (or add a charge to the user's credit card account). The issuer server may issue a payment command, e.g., 1927, to a database storing the user's account information, e.g., user profile database 1906b. The issuer server may provide an individual payment confirmation, e.g., 1928, to the healthcare collection portal server, which may forward, e.g., 1929, the funds transfer message to the acquirer server. An example listing of an individual payment confirmation 1928, substantially in the form of a HTTP(S) POST message including XML-formatted data, is provided below:

```

42 POST /clearance.php HTTP/1.1
43 Host: www.acquirer.com
44 Content-Type: Application/XML
45 Content-Length: 206
46 <?XML version = "1.0" encoding = "UTF-8"?>
47 <deposit_ack>

```

```
1      <request_ID>CNI4ICNW2</request_ID>
2      <clear_flag>true</clear_flag>
3      <timestamp>2011-02-22 17:00:02</timestamp>
4      <deposit_amount>$34.78</deposit_amount>
5 </deposit_ack>
```

6
7
8 **[00348]** In some embodiments, the acquirer server may parse the individual
9 payment confirmation, and correlate the transaction (e.g., using the request_ID field in
10 the example above) to the merchant. The acquirer server may then transfer the funds
11 specified in the funds transfer message to an account of the merchant. For example, the
12 acquirer server may query, e.g. 1930, an acquirer database 1907b for payment ledger
13 and/or merchant account data, e.g., 1931. The acquirer server may utilize payment
14 ledger and/or merchant account data from the acquirer database, along with the
15 individual payment confirmation, to generate updated payment ledger and/or merchant
16 account data, e.g., 1932. The acquirer server may then store, e.g., 1933, the updated
17 payment ledger and/or merchant account data to the acquire database.

18 **[00349]** FIGURES 20A-B show logic flow diagrams illustrating example aspects of
19 purchase transaction clearance in some embodiments of the RUAP, e.g., a Purchase
20 Transaction Clearance (“PTC”) component 2000. With reference to FIGURE 20A, in
21 some embodiments, a merchant server may initiate clearance of a batch of authorized
22 transactions. For example, the merchant server may generate a batch data request, e.g.,
23 2001, and provide the request to a merchant database. In response to the batch data
24 request, the database may provide the requested batch data, e.g., 2002. The server may
25 generate a batch clearance request, e.g., 2003, using the batch data obtained from the
26 database, and provide the batch clearance request to an acquirer server. The acquirer
27 server may parse, e.g., 2004, the obtained batch clearance request, and generate, e.g.,
28 2007, a batch payment request using the obtained batch clearance request to provide,

1 the batch payment request to a healthcare collection portal server. For example, the
2 acquirer server may query, e.g., 2005, an acquirer database for an address of a payment
3 network server, and utilize the obtained address, e.g., 2006, to forward the generated
4 batch payment request to the healthcare collection portal server.

5 **[00350]** The healthcare collection portal server may parse the batch payment
6 request obtained from the acquirer server, and extract the transaction data for each
7 transaction stored in the batch payment request, e.g., 2008. The healthcare collection
8 portal server may store the transaction data, e.g., 2009, for each transaction in a
9 healthcare collection portal database. In some embodiments, the healthcare collection
10 portal server may invoke a component, e.g., 2010, to provide analytics based on the
11 transactions of the merchant for whom purchase transaction are being cleared. For
12 example, the healthcare collection portal server may invoke a component such as the
13 example card transaction-based analytics component discussed above with reference to
14 FIGURE 10.

15 **[00351]** With reference to FIGURE 20B, in some embodiments, for each extracted
16 transaction, the healthcare collection portal server may query, e.g., 2011, a healthcare
17 collection portal database for an address of an issuer server. The healthcare collection
18 portal server may generate an individual payment request, e.g., 2013, for each
19 transaction for which it has extracted transaction data, and provide the individual
20 payment request to the issuer server. In some embodiments, the issuer server may
21 parse the individual payment request, e.g., 2014, and generate a payment command, e.g.,
22 2015, based on the parsed individual payment request. For example, the issuer server
23 may issue a command to deduct funds from the user's account (or add a charge to the

1 user's credit card account). The issuer server may issue a payment command, e.g., 2015,
2 to a database storing the user's account information, e.g., a user profile database. The
3 issuer server may provide an individual payment confirmation, e.g., 2017, to the
4 healthcare collection portal server, which may forward, e.g., 2018, the individual
5 payment confirmation to the acquirer server.

6 **[00352]** In some embodiments, the acquirer server may parse the individual
7 payment confirmation, and correlate the transaction (e.g., using the request_ID field in
8 the example above) to the merchant. The acquirer server may then transfer the funds
9 specified in the funds transfer message to an account of the merchant. For example, the
10 acquirer server may query, e.g. 2019, an acquirer database for payment ledger and/or
11 merchant account data, e.g., 2020. The acquirer server may utilize payment ledger
12 and/or merchant account data from the acquirer database, along with the individual
13 payment confirmation, to generate updated payment ledger and/or merchant account
14 data, e.g., 2021. The acquirer server may then store, e.g., 2022, the updated payment
15 ledger and/or merchant account data to the acquire database.

16 **[00353]** FIGURE 21 shows a logic flow diagram illustrating example aspects of
17 transaction data aggregation in some embodiments of the RUAP, e.g., a Transaction
18 Data Aggregation ("TDA") component. In some embodiments, a RUAP server may
19 obtain a trigger to aggregate transaction data, e.g., 2101. For example, the server may be
20 configured to initiate transaction data aggregation on a regular, periodic, or continuous
21 basis. As another example, the server may be configured to initiate transaction data
22 aggregation in real-time on-demand. The RUAP server may determine a scope of data
23 aggregation desired to perform the transaction analytics, e.g., 2102. For example, the

1 scope of data aggregation may be pre-determined. As another example, the scope of
2 data aggregation may be determined based on a received request for analytics, in real-
3 time. The RUAP server may initiate data aggregation based on the determined scope.
4 The RUAP server may generate a query for addresses of servers storing transaction data
5 within the determined scope, e.g., 2103. The RUAP server may query a database for
6 addresses of other servers that may have stored transaction data within the determined
7 scope of the data aggregation. The database may provide, e.g., 2104, a list of server
8 addresses in response to the RUAP server's query. Based on the list of server addresses,
9 the RUAP server may generate transaction data requests, e.g., 2105. The RUAP server
10 may issue the generated transaction data requests to the other servers. The other
11 servers may obtain and parse the transaction data requests, e.g., 2106. Based on parsing
12 the data requests, the other servers may generate transaction data queries, e.g., 2107,
13 and provide the transaction data queries to their transaction databases. In response to
14 the transaction data queries, the transaction databases may provide transaction data,
15 e.g., 2108, to the other servers. The other servers may return, e.g., 2109, the transaction
16 data obtained from the transactions databases to the RUAP server making the
17 transaction data requests. The RUAP server may generate aggregated transaction data
18 records from the transaction data received from the other servers, e.g., 2110, and store
19 the aggregated transaction data in a database, e.g., 2111.

20 **[00354]** FIGURE 22 illustrates an exemplary open system payment processing
21 network, depicting a general environment in which a merchant (m) 2210 can conduct a
22 transaction for goods and/or services with an account user (au) or customer on an
23 account (e.g., a prepaid account, credit account, points account, etc.) issued to an
24 account holder (a) 2208 by an issuer (i) 2204, where the processes of paying and being

1 paid for the transaction are coordinated by a transaction handler 2202. The transaction
2 includes participation from different entities that are each a component of the payment
3 processing system . As such, the open system payment processing network can be used
4 by a patient (or agent thereof) to pay a healthcare service provider to who a balance due
5 bill is owed as described above.

6 **[00355]** Payment processing system has a plurality of merchants 2210 that
7 includes merchant (1) 2210 through merchant (M) 2210, where M can be up to and
8 greater than an eight digit integer. Payment processing system has a plurality of
9 accounts 2208 each of which is held by a corresponding account holder (1) 2208
10 through account holder (A) 2208, where A can be up to and greater than a ten eight digit
11 integer.

12 **[00356]** Payment processing system includes account user (1) 2208 through
13 account user (AU) 2208, where AU can be as large as a ten digit integer or larger. Each
14 account user (au) may act as a customer and initiate a transaction for goods and/or
15 services with merchant (m) 2210 using an account that has been issued by an issuer (i)
16 2204 to a corresponding account holder (a) 2208. Data from the transaction on the
17 account is collected by merchant (m) and forwarded to a corresponding acquirer (a)
18 2206. Acquirer (a) 2206 forwards the data to transaction handler 2202 who facilitates
19 payment for the transaction from the account issued by the issuer (i) 2204 to account
20 holder (a) 2208.

21 **[00357]** Payment processing system has a plurality of issuers 2204. Each issuer (i)
22 2204 may be assisted in processing one or more transactions by a corresponding agent

1 issuer (ai) 2204, where 'i' can be an integer from 1 to I, where 'ai' can be an integer from
2 1 to AI, and where I and AI can be as large as an eight digit integer or larger.

3 **[00358]** Payment processing system has a plurality of acquirers 2206. Each
4 acquirer (q) 2206 may be assisted in processing one or more transactions by a
5 corresponding agent acquirer (aq) 2204, where 'q' can be an integer from 1 to Q, where
6 'aq' can be an integer from 1 to AQ, and where Q and AQ can be as large as a eight digit
7 integer or larger.

8 **[00359]** Payment processing system has a transaction handler 2202 to process a
9 plurality of transactions. The transaction handler 2202 can include one or a plurality or
10 networks and switches 2202. Each network/switch (ns) 2202 can be a mainframe
11 computer in a geographic location different than each other network/switch (ns) 2202,
12 where 'ns' is an integer from one to NS, and where NS can be as large as a four-digit
13 integer or larger.

14 **[00360]** Dedicated communication systems 2268 - 2276 (e.g., private
15 communication network(s)) facilitate communication between the transaction handler
16 2202 and each issuer (i) 2204 and each acquirer (a) 2206. The Internet 2212, via e-mail,
17 the World Wide Web, cellular telephony, and/or other optional public and private
18 communications systems, can facilitate communications using communication systems
19 2250 - 2266 among and between each issuer (i) 2204, each acquirer (a) 2206, each
20 merchant (m) 2210, each account holder (a) 2208, and the transaction handler 2202.
21 Alternatively and optionally, one or more dedicated communication systems 2250 -
22 2266 can facilitate respective communications between each acquirer (a) 2206 and each

1 merchant (m) 2210, each merchant (m) and each account holder (a) 2208, and each
2 account holder (a) 2208 and each issuer (i) 2204, respectively.

3 **[00361]** Each acquirer (q) 2206 may be assisted in processing one or more
4 transactions by a corresponding agent acquirer (aq) 2204, where 'q' can be an integer
5 from 1 to Q, where aq can be an integer from 1 to AQ, and where Q and AQ can be as
6 large as a eight digit integer or larger.

7 **[00362]** Merchant (m) 2210 may be a person or entity that sells goods and/or
8 services. Merchant (m) 2210 may also be, for instance, a healthcare service provider, a
9 manufacturer, a distributor, a retailer, a load agent, a drugstore, a grocery store, a gas
10 station, a hardware store, a supermarket, a boutique, a restaurant, or a doctor's office.
11 In a business-to-business setting, the account holder (a) 2208 may be a second
12 merchant making a purchase from another merchant (m) 2210. Merchant (m) 2210
13 may use at least one stationary and/or mobile point-of-sale terminal (POS) that can
14 communicate with acquirer (a) 2206, transaction handler 2202, or issuer (i) 2204. Thus,
15 the POS terminal is in operative communication with the payment processing system .

16 **[00363]** Typically, a transaction begins with account user (au) 2208 presenting a
17 portable consumer device to merchant (m) 2210 to initiate an exchange for a good or
18 service. The portable consumer device may be associated with an account (e.g., a
19 monetary or reward points account) of account holder (a) 2208 that was issued to the
20 account holder (a) 2208 by issuer (i) 2204.

21 **[00364]** The portable consumer device may be in a form factor that can be a
22 payment card, a gift card, a smartcard, a smart media device, a payroll card, a healthcare
23 card, a wrist band, a machine readable medium containing account information, a

1 keychain device, such as a SPEEDPASS® device commercially available from
2 ExxonMobil Corporation, or a supermarket discount card, a cellular phone, personal
3 digital assistant, a pager, a security card, an access card, a wireless terminal, or a
4 transponder. The portable consumer device may include a volatile or non-volatile
5 memory to store information such as the account number or the name of an account
6 holder (a) 2208.

7 **[00365]** Merchant (m) 2210 may use the POS terminal to obtain account
8 information, such as a number of the account of the account holder (a) 2208, from the
9 portable consumer device. The portable consumer device may interface with the POS
10 terminal using a mechanism including any suitable electrical, magnetic, or optical
11 interfacing system such as a contactless system using radio frequency or magnetic field
12 recognition system or contact system such as a magnetic stripe reader. The POS
13 terminal sends a transaction authorization request to the issuer (i) 2204 of the account
14 corresponding to the portable consumer device. Alternatively, or in combination, the
15 portable consumer device may communicate with issuer (i) 2204, transaction handler
16 2202, or acquirer (a) 2206.

17 **[00366]** Issuer (i) 2204 may authorize the transaction using transaction handler
18 2202. Transaction handler 2202 may also clear the transaction. Authorization includes
19 issuer (i) 2204, or transaction handler 2202 on behalf of issuer (i) 2204, authorizing the
20 transaction in connection with issuer (i) 2204's instructions such as through the use of
21 business rules. The business rules could include instructions or guidelines from
22 transaction handler 2202, account holder (a) 2208, merchant (m) 2210, acquirer (a)
23 2206, issuer (i) 2204, a related financial institution, or combinations thereof.

1 Transaction handler 2202 may maintain a log or history of authorized transactions.
2 Once approved, merchant (m) 2210 will record the authorization, allowing account user
3 (au) 2208 to receive the good or service from merchant (m) or an agent thereof.

4 **[00367]** Merchant (m) 2210 may, at discrete periods, such as at the end of the day,
5 submit a list of authorized transactions to acquirer (a) 2206 or other transaction related
6 data for processing through the payment processing system . Transaction handler 2202
7 may compare the submitted authorized transaction list with its own log of authorized
8 transactions. If a match is found, transaction handler 2202 may route authorization
9 transaction amount requests from the corresponding acquirer (a) 2206 to the
10 corresponding issuer (i) 2204 involved in each transaction. Once acquirer (a) 2206
11 receives the payment of the authorized transaction amount from issuer (i) 2204,
12 acquirer (a) 2206 can forward the payment to merchant (m) 2210 less any transaction
13 costs, such as fees for the processing of the transaction. If the transaction involves a
14 debit or prepaid card, acquirer (a) 2206 may choose not to wait for the issuer (i) 2204 to
15 forward the payment prior to paying merchant (m) 2210.

16 **[00368]** There may be intermittent steps in the foregoing process, some of which
17 may occur simultaneously. For example, acquirer (a) 2206 can initiate the clearing and
18 settling process, which can result in payment to acquirer (a) 2206 for the amount of the
19 transaction. Acquirer (a) 2206 may request from transaction handler 2202 that the
20 transaction be cleared and settled. Clearing includes the exchange of financial
21 information between the issuer (i) 2204 and the acquirer (a) 2206 and settlement
22 includes the exchange of funds. Transaction handler 2202 can provide services in
23 connection with settlement of the transaction. The settlement of a transaction includes

1 depositing an amount of the transaction settlement from a settlement house, such as a
2 settlement bank, which transaction handler 2202 typically chooses, into a clearinghouse,
3 such as a clearing bank, that acquirer (a) 2206 typically chooses. Issuer (i) 2204
4 deposits the same from a clearinghouse, such as a clearing bank, which issuer (i) 2204
5 typically chooses, into the settlement house. Thus, a typical transaction involves various
6 entities to request, authorize, and fulfill processing the transaction.

7 **[00369]** Payment processing system will preferably have network components
8 suitable for scaling the number and data payload size of transactions that can be
9 authorized, cleared and settled in both real time and batch processing. These include
10 hardware, software, data elements, and storage network devices for the same. Examples
11 of payment processing system include those operated, at least in part, by American
12 Express, Master Card, Discover Card, First Data Corporation, Diners Club, and Visa Inc.,
13 and agents of the foregoing.

14 **[00370]** Each network/switch (ns) 2202 can include one or more data centers for
15 processing transactions, where each transaction can include up to 100 kilobytes of data
16 or more. The data corresponding to the transaction can include information about the
17 types and quantities of goods and services in the transaction, information about the
18 account holder (a) 2208, the account user (au) 2208, the merchant (m) 2210, financial
19 and incentive treatment(s) of the goods and services, coupons, rebates, rewards, loyalty,
20 discounts, returns, exchanges, cash-back transactions, etc. Of course, in the case of the
21 data corresponding to a healthcare-related transaction, the information would
22 necessarily be limited with respect to the goods and services in the transaction as would

1 be consistent with full regulatory compliance (e.g.; HIPAA compliance in the USA, the
2 Personal Health Information Protection Act (PHIPA) in Canada, etc.)

3 **[00371]** By way of example, network/switch (ns) 2202 can include one or more
4 mainframe computers (e.g., one or more IBM mainframe computers) for
5 communications over systems 2268 - 2276, one or more server farms (e.g., one or more
6 Sun UNIX Superservers), where the mainframe computers and server farms can be in
7 diverse geographic locations.

8 **[00372]** Each issuer (i) 2204 (or agent issuer (ai) 2204 thereof) and each acquirer
9 (a) 2206 (or agent acquirer (aq) 2206 thereof) can use one or more router/switch (e.g.,
10 Cisco routers/switches) to communicate with each network/switch (ns) 2202 via
11 dedicated communication systems 2268 - 2276, respectively.

12 **[00373]** Transaction handler 2202 stores information about transactions processed
13 through payment processing system in data warehouses such as may be incorporated as
14 part of the plurality of networks/switches (ns) 2202. This information can be data
15 mined. The data mining transaction research and modeling can be used for advertising,
16 account holder and merchant loyalty incentives and rewards, fraud detection and
17 prediction, and to develop tools to demonstrate savings and efficiencies made possible
18 by use of the payment processing system over paying and being paid by cash, checks, or
19 other traditional payment mechanisms.

20 **[00374]** The VisaNet® system is an example component of the transaction handler
21 2202 in the payment processing system. The open system payment processing network
22 seen in FIGURE 22 can be enabled via a telecommunications network that may make
23 use of any suitable telecommunications network and may involve different hardware,

1 different software and/or different protocols then those discussed below. FIGURE 22
2 can be deemed to depict a global telecommunications network that supports purchase
3 and cash transactions using any bankcard, travel and entertainment cards, and other
4 private label and proprietary cards. The network also supports ATM transactions for
5 other networks, transactions using paper checks, transactions using smart cards, virtual
6 cards, and transactions using other financial instruments.

7 **[00375]** These transactions are processed through the network's authorization,
8 clearing and settlement services. Authorization is when an issuer approves or declines a
9 sales transaction before a purchase is finalized or cash is dispersed. Clearing is when a
10 transaction is delivered from an acquirer to an issuer for posting to the customer's
11 account. Settlement is the process of calculating and determining the net financial
12 position of each member for all transactions that are cleared. The actual exchange of
13 funds is a separate process.

14 **[00376]** Transactions can be authorized, cleared and settled as either a dual
15 message or a single message transaction. A dual message transaction is sent twice-the
16 first time with only information needed for an authorization decision, an again later
17 with additional information for clearing and settlement. A single message transaction is
18 sent once for authorization and contains clearing and settlement information as well.
19 Typically, authorization, clearing and settlement all occur on-line.

20 **[00377]** FIGURE 22 includes one or more transaction handlers 2202, access points
21 2230, 2232, acquirers 2206, and issuers 2204. Other entities such as payor banks and
22 third party authorizing agents may also connect to the network through an access point.
23 An interchange center is a data processing center that may be located anywhere in the

1 world. In one embodiment, there are two in the United States and one each in the
2 United Kingdom and in Japan. Each interchange center houses the computer system
3 that performs the network transaction processing. The interchange center serves as the
4 control point for the telecommunication facilities of the network, which comprise high
5 speed leased lines or satellite connections based on IBM SNA protocol. Preferably, the
6 communication lines that connect an interchange center (Transaction Handler 2202) to
7 remote entities use dedicated high-bandwidth telephone circuits or satellite connections
8 based on the IBM SNA-LUo communication protocol. Messages are sent over these lines
9 using any suitable implementation of the ISO 8583 standard.

10 **[00378]** Access points 2230, 2232 are typically made up of small computer systems
11 located at a processing center that interfaces between the center's host computer and the
12 interchange center. The access point facilitates the transmission of messages and files
13 between the host and the interchange center supporting the authorization, clearing and
14 settlement of transaction. Telecommunication links between the acquirer (q) 2206 and
15 its access point 2232, and between the access point 2230 and issuer (i) 2204 are
16 typically local links within a center and use a proprietary message format as preferred by
17 the center.

18 **[00379]** A data processing center (such as is located within an acquirer, issuer, or
19 other entity) houses processing systems that support merchant and business locations
20 and maintains customer data and billing systems. Preferably, each processing center is
21 linked to one or two interchange centers. Processors are connected to the closest
22 interchange, and if the network experiences interruptions, the network automatically
23 routes transactions to a secondary interchange center. Each interchange center is also

1 linked to all of the other interchange centers. This linking enables processing centers to
2 communicate with each other through one or more interchange centers. Also,
3 processing centers can access the networks of other programs through the interchange
4 center. Further, the network ensures that all links have multiple backups. The
5 connection from one point of the network to another is not usually a fixed link; instead,
6 the interchange center chooses the best possible path at the time of any given
7 transmission. Rerouting around any faulty link occurs automatically.

8 **[00380]** FIGURE 23 illustrates an integrated payment system 2340 housed within
9 an interchange center to provide on-line and off-line transaction processing within
10 embodiments of the RUAP. For dual message transaction, authorization system 2342
11 provides authorization. Authorization system 2342 supports on-line and off-line
12 functions, and its file includes internal systems tables, a customer database and a
13 merchant central file. The on-line functions of system 2342 support dual message
14 authorization processing. This processing involves routing, cardholder and card
15 verification and stand-in processing, and other functions such as file maintenance. Off-
16 line functions including reporting, billing, and generating recovery bulletins. Reporting
17 includes authorization reports, exception file and advice file reports, POS reports and
18 billing reports. A bridge from system 2342 to system 2346 makes it possible for
19 members using system 542 to communicate with members using system 2346 and
20 access the SMS gateways to outside networks.

21 **[00381]** Clearing and settlement system 2344 clears and settles previously
22 authorized dual message transactions. Operating six days a week on a global basis,
23 system 2344 collects financial and non-financial information and distributes reports

1 between members It also calculates fees, charges and settlement totals and produces
2 reports to help with reconciliation. A bridge forms an interchange between system 2344
3 processing centers and system 2346 processing centers.

4 **[00382]** Single message system 2346 processes full financial transactions. System
5 546 can also process dual message authorization and clearing transactions, and
6 communicates with system 2342 using a bridge and accesses outside networks as
7 required. System 2346 processes Visa, Plus Interlink and other card transactions. The
8 SMS files comprise internal system tables that control system access and processing,
9 and the cardholder database, which contains files of cardholder data used for PIN
10 verification and stand-in processing authorization. System 2346 on-line functions
11 perform real-time cardholder transaction processing and exception processing for
12 authorization as well as full financial transactions. System 2346 also accumulates
13 reconciliation and settlement totals. System 2346 off-line functions process settlement
14 and funds transfer requests and provide settlement and activities reporting. Settlement
15 service 548 consolidates the settlement functions of system 2344 and 2346, including
16 Interlink, into a single service for all products and services. Clearing continues to be
17 performed separately by system 2344 and system 2346.

18 RUAP Controller

19 **[00383]** FIGURE 24 shows a block diagram illustrating example aspects of a RUAP
20 controller 2401. In this embodiment, the RUAP controller 2401 may serve to aggregate,
21 process, store, search, serve, identify, instruct, generate, match, and/or facilitate
22 interactions with a computer through various technologies, and/or other related data.

1 **[00384]** Users, e.g., 2433a, which may be people and/or other systems, may engage
2 information technology systems (e.g., computers) to facilitate information processing.
3 In turn, computers employ processors to process information; such processors 2403
4 may be referred to as central processing units (CPU). One form of processor is referred
5 to as a microprocessor. CPUs use communicative circuits to pass binary encoded signals
6 acting as instructions to enable various operations. These instructions may be
7 operational and/or data instructions containing and/or referencing other instructions
8 and data in various processor accessible and operable areas of memory 2429 (e.g.,
9 registers, cache memory, random access memory, etc.). Such communicative
10 instructions may be stored and/or transmitted in batches (e.g., batches of instructions)
11 as programs and/or data components to facilitate desired operations. These stored
12 instruction codes, e.g., programs, may engage the CPU circuit components and other
13 motherboard and/or system components to perform desired operations. One type of
14 program is a computer operating system, which, may be executed by CPU on a computer;
15 the operating system enables and facilitates users to access and operate computer
16 information technology and resources. Some resources that may be employed in
17 information technology systems include: input and output mechanisms through which
18 data may pass into and out of a computer; memory storage into which data may be
19 saved; and processors by which information may be processed. These information
20 technology systems may be used to collect data for later retrieval, analysis, and
21 manipulation, which may be facilitated through a database program. These information
22 technology systems provide interfaces that allow users to access and operate various
23 system components.

1 **[00385]** In one embodiment, the RUAP controller 2401 may be connected to
2 and/or communicate with entities such as, but not limited to: one or more users from
3 user input devices 2411; peripheral devices 2412; an optional cryptographic processor
4 device 2428; and/or a communications network 2413. For example, the RUAP
5 controller 2401 may be connected to and/or communicate with users, e.g., 2433a,
6 operating client device(s), e.g., 2433b, including, but not limited to, personal
7 computer(s), server(s) and/or various mobile device(s) including, but not limited to,
8 cellular telephone(s), smartphone(s) (e.g., iPhone®, Blackberry®, Android OS-based
9 phones etc.), tablet computer(s) (e.g., Apple iPad™, HP Slate™, Motorola Xoom™, etc.),
10 eBook reader(s) (e.g., Amazon Kindle™, Barnes and Noble's Nook™ eReader, etc.),
11 laptop computer(s), notebook(s), netbook(s), gaming console(s) (e.g., XBOX Live™,
12 Nintendo® DS, Sony PlayStation® Portable, etc.), portable scanner(s), and/or the like.

13 **[00386]** Networks are commonly thought to comprise the interconnection and
14 interoperation of clients, servers, and intermediary nodes in a graph topology. It should
15 be noted that the term “server” as used throughout this application refers generally to a
16 computer, other device, program, or combination thereof that processes and responds to
17 the requests of remote users across a communications network. Servers serve their
18 information to requesting “clients.” The term “client” as used herein refers generally to a
19 computer, program, other device, user and/or combination thereof that is capable of
20 processing and making requests and obtaining and processing any responses from
21 servers across a communications network. A computer, other device, program, or
22 combination thereof that facilitates, processes information and requests, and/or
23 furthers the passage of information from a source user to a destination user is
24 commonly referred to as a “node.” Networks are generally thought to facilitate the

1 transfer of information from source points to destinations. A node specifically tasked
2 with furthering the passage of information from a source to a destination is commonly
3 called a “router.” There are many forms of networks such as Local Area Networks
4 (LANs), Pico networks, Wide Area Networks (WANs), Wireless Networks (WLANs), etc.
5 For example, the Internet is generally accepted as being an interconnection of a
6 multitude of networks whereby remote clients and servers may access and interoperate
7 with one another.

8 **[00387]** The RUAP controller 2401 may be based on computer systems that may
9 comprise, but are not limited to, components such as: a computer systemization 2402
10 connected to memory 2429.

11 **Computer Systemization**

12 **[00388]** A computer systemization 2402 may comprise a clock 2430, central
13 processing unit (“CPU(s)” and/or “processor(s)” (these terms are used interchangeably
14 throughout the disclosure unless noted to the contrary)) 2403, a memory 2429 (e.g., a
15 read only memory (ROM) 2406, a random access memory (RAM) 2405, etc.), and/or an
16 interface bus 2407, and most frequently, although not necessarily, are all interconnected
17 and/or communicating through a system bus 2404 on one or more (mother)board(s)
18 2402 having conductive and/or otherwise transportive circuit pathways through which
19 instructions (e.g., binary encoded signals) may travel to effectuate communications,
20 operations, storage, etc. The computer systemization may be connected to a power
21 source 2486; e.g., optionally the power source may be internal. Optionally, a
22 cryptographic processor 2426 and/or transceivers (e.g., ICs) 2474 may be connected to
23 the system bus. In another embodiment, the cryptographic processor and/or

transceivers may be connected as either internal and/or external peripheral devices via the interface bus I/O. In turn, the transceivers may be connected to antenna(s) 2475, thereby effectuating wireless transmission and reception of various communication and/or sensor protocols; for example the antenna(s) may connect to: a Texas Instruments WiLink WL1283 transceiver chip (e.g., providing 802.11n, Bluetooth 3.0, FM, global positioning system (GPS) (thereby allowing RUAP controller to determine its location)); Broadcom BCM4329FKUBG transceiver chip (e.g., providing 802.11n, Bluetooth 2.1 + EDR, FM, etc.), BCM28150 (HSPA+) and BCM2076 (Bluetooth 4.0, GPS, etc.); a Broadcom BCM4750IUB8 receiver chip (e.g., GPS); an Infineon Technologies X-Gold 618-PMB9800 (e.g., providing 2G/3G HSDPA/HSUPA communications); Intel's XMM 7160 (LTE & DC-HSPA), Qualcomm's CDMA(2000), Mobile Data/Station Modem, Snapdragon; and/or the like. The system clock may have a crystal oscillator and generates a base signal through the computer systemization's circuit pathways. The clock may be coupled to the system bus and various clock multipliers that will increase or decrease the base operating frequency for other components interconnected in the computer systemization. The clock and various components in a computer systemization drive signals embodying information throughout the system. Such transmission and reception of instructions embodying information throughout a computer systemization may be referred to as communications. These communicative instructions may further be transmitted, received, and the cause of return and/or reply communications beyond the instant computer systemization to: communications networks, input devices, other computer systemizations, peripheral devices, and/or the like. It should be understood that in alternative embodiments, any of the above components may be connected directly to

1 one another, connected to the CPU, and/or organized in numerous variations employed
2 as exemplified by various computer systems.

3 **[00389]** The CPU comprises at least one high-speed data processor adequate to
4 execute program components for executing user and/or system-generated requests.
5 Often, the processors themselves will incorporate various specialized processing units,
6 such as, but not limited to: floating point units, integer processing units, integrated
7 system (bus) controllers, logic operating units, memory management control units, etc.,
8 and even specialized processing sub-units like graphics processing units, digital signal
9 processing units, and/or the like. Additionally, processors may include internal fast
10 access addressable memory, and be capable of mapping and addressing memory 2429
11 beyond the processor itself; internal memory may include, but is not limited to: fast
12 registers, various levels of cache memory (e.g., level 1, 2, 3, etc.), RAM, etc. The
13 processor may access this memory through the use of a memory address space that is
14 accessible via instruction address, which the processor can construct and decode
15 allowing it to access a circuit path to a specific memory address space having a memory
16 state/value. The CPU may be a microprocessor such as: AMD's Athlon, Duron and/or
17 Opteron; ARM's classic (e.g., ARM7/9/11), embedded (Cortex-M/R), application
18 (Cortex-A), embedded and secure processors; IBM and/or Motorola's DragonBall and
19 PowerPC; IBM's and Sony's Cell processor; Intel's Atom, Celeron (Mobile), Core
20 (2/Duo/i3/i5/i7), Itanium, Pentium, Xeon, and/or XScale; and/or the like processor(s).
21 The CPU interacts with memory through instruction passing through conductive and/or
22 transportive conduits (e.g., (printed) electronic and/or optic circuits) to execute stored
23 instructions (i.e., program code). Such instruction passing facilitates communication
24 within the RUAP controller and beyond through various interfaces. Should processing

1 requirements dictate a greater amount speed and/or capacity, distributed processors
2 (e.g., Distributed RUAP), mainframe, multi-core, parallel, and/or super-computer
3 architectures may similarly be employed. Alternatively, should deployment
4 requirements dictate greater portability, smaller mobile devices (e.g., smartphones,
5 Personal Digital Assistants (PDAs), etc.) may be employed.

6 **[00390]** Depending on the particular implementation, features of the RUAP may be
7 achieved by implementing a microcontroller such as CAST's R8051XC2 microcontroller;
8 Intel's MCS 51 (i.e., 8051 microcontroller); and/or the like. Also, to implement certain
9 features of the RUAP, some feature implementations may rely on embedded
10 components, such as: Application-Specific Integrated Circuit ("ASIC"), Digital Signal
11 Processing ("DSP"), Field Programmable Gate Array ("FPGA"), and/or the like
12 embedded technology. For example, any of the RUAP component collection (distributed
13 or otherwise) and/or features may be implemented via the microprocessor and/or via
14 embedded components; e.g., via ASIC, coprocessor, DSP, FPGA, and/or the like.
15 Alternately, some implementations of the RUAP may be implemented with embedded
16 components that are configured and used to achieve a variety of features or signal
17 processing.

18 **[00391]** Depending on the particular implementation, the embedded components
19 may include software solutions, hardware solutions, and/or some combination of both
20 hardware/software solutions. For example, RUAP features discussed herein may be
21 achieved through implementing FPGAs, which are a semiconductor devices containing
22 programmable logic components called "logic blocks", and programmable interconnects,
23 such as the high performance FPGA Virtex series and/or the low cost Spartan series

1 manufactured by Xilinx. Logic blocks and interconnects can be programmed by the
2 customer or designer, after the FPGA is manufactured, to implement any of the RUAP
3 features. A hierarchy of programmable interconnects allow logic blocks to be
4 interconnected as needed by the RUAP system designer/administrator, somewhat like a
5 one-chip programmable breadboard. An FPGA's logic blocks can be programmed to
6 perform the operation of basic logic gates such as AND, and XOR, or more complex
7 combinational operators such as decoders or simple mathematical operations. In most
8 FPGAs, the logic blocks also include memory elements, which may be circuit flip-flops
9 or more complete blocks of memory. In some circumstances, the RUAP may be
10 developed on regular FPGAs and then migrated into a fixed version that more resembles
11 ASIC implementations. Alternate or coordinating implementations may migrate RUAP
12 controller features to a final ASIC instead of or in addition to FPGAs. Depending on the
13 implementation all of the aforementioned embedded components and microprocessors
14 may be considered the "CPU" and/or "processor" for the RUAP.

15 **Power Source**

16 **[00392]** The power source 2486 may be of any standard form for powering small
17 electronic circuit board devices such as the following power cells: alkaline, lithium
18 hydride, lithium ion, lithium polymer, nickel cadmium, solar cells, and/or the like.
19 Other types of AC or DC power sources may be used as well. In the case of solar cells, in
20 one embodiment, the case provides an aperture through which the solar cell may
21 capture photonic energy. The power cell 2486 is connected to at least one of the
22 interconnected subsequent components of the RUAP thereby providing an electric
23 current to all the interconnected components. In one example, the power source 2486 is

1 connected to the system bus component 2404. In an alternative embodiment, an outside
2 power source 2486 is provided through a connection across the I/O 2408 interface. For
3 example, a USB and/or IEEE 1394 connection carries both data and power across the
4 connection and is therefore a suitable source of power.

5 **Interface Adapters**

6 **[00393]** Interface bus(es) 2407 may accept, connect, and/or communicate to a
7 number of interface adapters, frequently, although not necessarily in the form of
8 adapter cards, such as but not limited to: input output interfaces (I/O) 2408, storage
9 interfaces 2409, network interfaces 2410, and/or the like. Optionally, cryptographic
10 processor interfaces 2427 similarly may be connected to the interface bus. The interface
11 bus provides for the communications of interface adapters with one another as well as
12 with other components of the computer systemization. Interface adapters are adapted
13 for a compatible interface bus. Interface adapters may connect to the interface bus via
14 expansion and/or slot architecture. Various expansion and/or slot architectures may be
15 employed, such as, but not limited to: Accelerated Graphics Port (AGP), Card Bus,
16 ExpressCard, (Extended) Industry Standard Architecture ((E)ISA), Micro Channel
17 Architecture (MCA), NuBus, Peripheral Component Interconnect (Extended) (PCI(X)),
18 PCI Express, Personal Computer Memory Card International Association (PCMCIA),
19 Thunderbolt, and/or the like.

20 **[00394]** Storage interfaces 2409 may accept, communicate, and/or connect to a
21 number of storage devices such as, but not limited to: storage devices 2414, removable
22 disc devices, and/or the like. Storage interfaces may employ connection protocols such
23 as, but not limited to: (Ultra) (Serial) Advanced Technology Attachment (Packet

1 Interface) ((Ultra) (Serial) ATA(PI)), (Enhanced) Integrated Drive Electronics ((E)IDE),
2 Institute of Electrical and Electronics Engineers (IEEE) 1394, Ethernet, fiber channel,
3 Small Computer Systems Interface (SCSI), Thunderbolt, Universal Serial Bus (USB),
4 and/or the like.

5 **[00395]** Network interfaces 2410 may accept, communicate, and/or connect to a
6 communications network 2413. Through a communications network 2413, the RUAP
7 controller is accessible through remote clients 2433b (e.g., computers with web
8 browsers) by users 2433a. Network interfaces may employ connection protocols such as,
9 but not limited to: direct connect, Ethernet (thick, thin, twisted pair 10/100/1000 Base
10 T, and/or the like), Token Ring, wireless connection such as IEEE 802.11a-x, and/or the
11 like. Should processing requirements dictate a greater amount speed and/or capacity,
12 distributed network controllers (e.g., Distributed RUAP), architectures may similarly be
13 employed to pool, load balance, and/or otherwise increase the communicative
14 bandwidth required by the RUAP controller. A communications network may be any
15 one and/or the combination of the following: a direct interconnection; the Internet; a
16 Local Area Network (LAN); a Metropolitan Area Network (MAN); an Operating
17 Missions as Nodes on the Internet (OMNI); a secured custom connection; a Wide Area
18 Network (WAN); a wireless network (e.g., employing protocols such as, but not limited
19 to a Wireless Application Protocol (WAP), I-mode, and/or the like); and/or the like. A
20 network interface may be regarded as a specialized form of an input output interface.
21 Further, multiple network interfaces 2410 may be used to engage with various
22 communications network types 2413. For example, multiple network interfaces may be
23 employed to allow for the communication over broadcast, multicast, and/or unicast
24 networks.

1 **[00396]** Input Output interfaces (I/O) 2408 may accept, communicate, and/or
2 connect to user input devices 2411, peripheral devices 2412, cryptographic processor
3 devices 2428, and/or the like. I/O may employ connection protocols such as, but not
4 limited to: audio: analog, digital, monaural, RCA, stereo, and/or the like; data: Apple
5 Desktop Bus (ADB), Bluetooth, IEEE 1394a-b, serial, universal serial bus (USB);
6 infrared; joystick; keyboard; midi; optical; PC AT; PS/2; parallel; radio; video interface:
7 Apple Desktop Connector (ADC), BNC, coaxial, component, composite, digital,
8 DisplayPort, Digital Visual Interface (DVI), high-definition multimedia interface
9 (HDMI), RCA, RF antennae, S-Video, VGA, and/or the like; wireless transceivers:
10 802.11a/b/g/n/x; Bluetooth; cellular (e.g., code division multiple access (CDMA), high
11 speed packet access (HSPA(+)), high-speed downlink packet access (HSDPA), global
12 system for mobile communications (GSM), long term evolution (LTE), WiMax, etc.);
13 and/or the like. One output device may be a video display, which may take the form of a
14 Cathode Ray Tube (CRT), Liquid Crystal Display (LCD), Light Emitting Diode (LED),
15 Organic Light Emitting Diode (OLED), Plasma, and/or the like based monitor with an
16 interface (e.g., VGA, DVI circuitry and cable) that accepts signals from a video interface.
17 The video interface composites information generated by a computer systemization and
18 generates video signals based on the composited information in a video memory frame.
19 Another output device is a television set, which accepts signals from a video interface.
20 Often, the video interface provides the composited video information through a video
21 connection interface that accepts a video display interface (e.g., an RCA composite video
22 connector accepting an RCA composite video cable; a DVI connector accepting a DVI
23 display cable, HDMI, etc.).

1 **[00397]** User input devices 2411 often are a type of peripheral device 2412 (see
2 below) and may include: card readers, dongles, finger print readers, gloves, graphics
3 tablets, joysticks, keyboards, microphones, mouse (mice), remote controls, retina
4 readers, touch screens (e.g., capacitive, resistive, etc.), trackballs, trackpads, sensors
5 (e.g., accelerometers, ambient light, GPS, gyroscopes, proximity, etc.), styluses, and/or
6 the like.

7 **[00398]** Peripheral devices 2412 may be connected and/or communicate to I/O
8 and/or other facilities of the like such as network interfaces, storage interfaces, directly
9 to the interface bus, system bus, the CPU, and/or the like. Peripheral devices may be
10 external, internal and/or part of the RUAP controller. Peripheral devices may include:
11 antenna, audio devices (e.g., line-in, line-out, microphone input, speakers, etc.),
12 cameras (e.g., still, video, webcam, etc.), dongles (e.g., for copy protection, ensuring
13 secure transactions with a digital signature, and/or the like), external processors (for
14 added capabilities; e.g., crypto devices 2428), force-feedback devices (e.g., vibrating
15 motors), near field communication (NFC) devices, network interfaces, printers, radio
16 frequency identifiers (RFIDs), scanners, storage devices, transceivers (e.g., cellular, GPS,
17 etc.), video devices (e.g., goggles, monitors, etc.), video sources, visors, and/or the like.
18 Peripheral devices often include types of input devices (e.g., microphones, cameras, etc.).

19 **[00399]** It should be noted that although user input devices and peripheral devices
20 may be employed, the RUAP controller may be embodied as an embedded, dedicated,
21 and/or monitor-less (i.e., headless) device, wherein access would be provided over a
22 network interface connection.

1 **[00400]** Cryptographic units such as, but not limited to, microcontrollers,
2 processors 2426, interfaces 2427, and/or devices 2428 may be attached, and/or
3 communicate with the RUAP controller. A MC68HC16 microcontroller, manufactured
4 by Motorola Inc., may be used for and/or within cryptographic units. The MC68HC16
5 microcontroller utilizes a 16-bit multiply-and-accumulate instruction in the 16 MHz
6 configuration and requires less than one second to perform a 512-bit RSA private key
7 operation. Cryptographic units support the authentication of communications from
8 interacting agents, as well as allowing for anonymous transactions. Cryptographic units
9 may also be configured as part of the CPU. Equivalent microcontrollers and/or
10 processors may also be used. Other commercially available specialized cryptographic
11 processors include: the Broadcom's CryptoNetX and other Security Processors;
12 nCipher's nShield (e.g., Solo, Connect, etc.), SafeNet's Luna PCI (e.g., 7100) series;
13 Semaphore Communications' 40 MHz Roadrunner 184; sMIP's (e.g., 208956); Sun's
14 Cryptographic Accelerators (e.g., Accelerator 6000 PCIe Board, Accelerator 500
15 Daughtercard); Via Nano Processor (e.g., L2100, L2200, U2400) line, which is capable
16 of performing 500+ MB/s of cryptographic instructions; VLSI Technology's 33 MHz
17 6868; and/or the like.

18

Memory

19 **[00401]** Generally, any mechanization and/or embodiment allowing a processor to
20 affect the storage and/or retrieval of information is regarded as memory 2429. However,
21 memory is a fungible technology and resource, thus, any number of memory
22 embodiments may be employed in lieu of or in concert with one another. It is to be
23 understood that the RUAP controller and/or a computer systemization may employ

1 various forms of memory 2429. For example, a computer systemization may be
2 configured wherein the operation of on-chip CPU memory (e.g., registers), RAM, ROM,
3 and any other storage devices are provided by a paper punch tape or paper punch card
4 mechanism; however, such an embodiment would result in an extremely slow rate of
5 operation. In one configuration, memory 2429 may include ROM 2406, RAM 2405, and
6 a storage device 2414. A storage device 2414 may employ any number of computer
7 storage devices/systems. Storage devices may include a drum; a (fixed and/or
8 removable) magnetic disk drive; a magneto-optical drive; an optical drive (i.e., Blu-ray,
9 CD ROM/RAM/Recordable (R)/ReWritable (RW), DVD R/RW, HD DVD R/RW etc.);
10 an array of devices (e.g., Redundant Array of Independent Disks (RAID)); solid state
11 memory devices (USB memory, solid state drives (SSD), etc.); other processor-readable
12 storage mediums; and/or other devices of the like. Thus, a computer systemization
13 generally requires and makes use of memory.

14 **Component Collection**

15 **[00402]** The memory 2429 may contain a collection of program and/or database
16 components and/or data such as, but not limited to: operating system component(s)
17 2415 (operating system); information server component(s) 2416 (information server);
18 user interface component(s) 2417 (user interface); Web browser component(s) 2418
19 (Web browser); database(s) 2419; mail server component(s) 2421; mail client
20 component(s) 2422; cryptographic server component(s) 2420 (cryptographic server);
21 the RUAP component(s) 2435; and/or the like (i.e., collectively a component collection).
22 These components may be stored and accessed from the storage devices and/or from
23 storage devices accessible through an interface bus. Although non-conventional

1 program components such as those in the component collection may be stored in a local
2 storage device 2414, they may also be loaded and/or stored in memory such as:
3 peripheral devices, RAM, remote storage facilities through a communications network,
4 ROM, various forms of memory, and/or the like.

5 **Operating System**

6 **[00403]** The operating system component 2415 is an executable program
7 component facilitating the operation of the RUAP controller. The operating system may
8 facilitate access of I/O, network interfaces, peripheral devices, storage devices, and/or
9 the like. The operating system may be a highly fault tolerant, scalable, and secure system
10 such as: Apple Macintosh OS X (Server); AT&T Plan 9; Be OS; Unix and Unix-like
11 system distributions (such as AT&T's UNIX; Berkley Software Distribution (BSD)
12 variations such as FreeBSD, NetBSD, OpenBSD, and/or the like; Linux distributions
13 such as Red Hat, Ubuntu, and/or the like); and/or the like operating systems. However,
14 more limited and/or less secure operating systems also may be employed such as Apple
15 Macintosh OS, IBM OS/2, Microsoft DOS, Microsoft Windows
16 2000/2003/3.1/95/98/CE/Millennium/NT/Vista/XP (Server), Palm OS, and/or the like.
17 In addition, emobile operating systems such as Apple's iOS, Google's Android, Hewlett
18 Packard's WebOS, Microsofts Windows Mobile, and/or the like may be employed. Any
19 of these operating systems may be embedded within the hardware of the NICK
20 controller, and/or stored/loaded into memory/storage. An operating system may
21 communicate to and/or with other components in a component collection, including
22 itself, and/or the like. Most frequently, the operating system communicates with other
23 program components, user interfaces, and/or the like. For example, the operating

1 system may contain, communicate, generate, obtain, and/or provide program
2 component, system, user, and/or data communications, requests, and/or responses. The
3 operating system, once executed by the CPU, may enable the interaction with
4 communications networks, data, I/O, peripheral devices, program components, memory,
5 user input devices, and/or the like. The operating system may provide communications
6 protocols that allow the RUAP controller to communicate with other entities through a
7 communications network 2413. Various communication protocols may be used by the
8 RUAP controller as a subcarrier transport mechanism for interaction, such as, but not
9 limited to: multicast, TCP/IP, UDP, unicast, and/or the like.

10 **Information Server**

11 **[00404]** An information server component 2416 is a stored program component
12 that is executed by a CPU. The information server may be an Internet information
13 server such as, but not limited to Apache Software Foundation's Apache, Microsoft's
14 Internet Information Server, and/or the like. The information server may allow for the
15 execution of program components through facilities such as Active Server Page (ASP),
16 ActiveX, (ANSI) (Objective-) C (++), C# and/or .NET, Common Gateway Interface (CGI)
17 scripts, dynamic (D) hypertext markup language (HTML), FLASH, Java, JavaScript,
18 Practical Extraction Report Language (PERL), Hypertext Pre-Processor (PHP), pipes,
19 Python, wireless application protocol (WAP), WebObjects, and/or the like. The
20 information server may support secure communications protocols such as, but not
21 limited to, File Transfer Protocol (FTP); HyperText Transfer Protocol (HTTP); Secure
22 Hypertext Transfer Protocol (HTTPS), Secure Socket Layer (SSL), messaging protocols
23 (e.g., America Online (AOL) Instant Messenger (AIM), Apple's iMessage, Application

1 Exchange (APEX), ICQ, Internet Relay Chat (IRC), Microsoft Network (MSN)
2 Messenger Service, Presence and Instant Messaging Protocol (PRIM), Internet
3 Engineering Task Force's (IETF's) Session Initiation Protocol (SIP), SIP for Instant
4 Messaging and Presence Leveraging Extensions (SIMPLE), open XML-based Extensible
5 Messaging and Presence Protocol (XMPP) (i.e., Jabber or Open Mobile Alliance's
6 (OMA's) Instant Messaging and Presence Service (IMPS)), Yahoo! Instant Messenger
7 Service, and/or the like. The information server provides results in the form of Web
8 pages to Web browsers, and allows for the manipulated generation of the Web pages
9 through interaction with other program components. After a Domain Name System
10 (DNS) resolution portion of an HTTP request is resolved to a particular information
11 server, the information server resolves requests for information at specified locations on
12 the RUAP controller based on the remainder of the HTTP request. For example, a
13 request such as `http://123.124.125.126/myInformation.html` might have the IP portion
14 of the request "123.124.125.126" resolved by a DNS server to an information server at
15 that IP address; that information server might in turn further parse the http request for
16 the `"/myInformation.html"` portion of the request and resolve it to a location in memory
17 containing the information "myInformation.html." Additionally, other information
18 serving protocols may be employed across various ports, e.g., FTP communications
19 across port 21, and/or the like. An information server may communicate to and/or with
20 other components in a component collection, including itself, and/or facilities of the like.
21 Most frequently, the information server communicates with the RUAP database 2419,
22 operating systems, other program components, user interfaces, Web browsers, and/or
23 the like.

1 **[00405]** Access to the RUAP database may be achieved through a number of
2 database bridge mechanisms such as through scripting languages as enumerated below
3 (e.g., CGI) and through inter-application communication channels as enumerated below
4 (e.g., CORBA, WebObjects, etc.). Any data requests through a Web browser are parsed
5 through the bridge mechanism into appropriate grammars as required by the RUAP. In
6 one embodiment, the information server would provide a Web form accessible by a Web
7 browser. Entries made into supplied fields in the Web form are tagged as having been
8 entered into the particular fields, and parsed as such. The entered terms are then passed
9 along with the field tags, which act to instruct the parser to generate queries directed to
10 appropriate tables and/or fields. In one embodiment, the parser may generate queries in
11 standard SQL by instantiating a search string with the proper join/select commands
12 based on the tagged text entries, wherein the resulting command is provided over the
13 bridge mechanism to the RUAP as a query. Upon generating query results from the
14 query, the results are passed over the bridge mechanism, and may be parsed for
15 formatting and generation of a new results Web page by the bridge mechanism. Such a
16 new results Web page is then provided to the information server, which may supply it to
17 the requesting Web browser.

18 **[00406]** Also, an information server may contain, communicate, generate, obtain,
19 and/or provide program component, system, user, and/or data communications,
20 requests, and/or responses.

21 **User Interface**

22 **[00407]** Computer interfaces in some respects are similar to automobile operation
23 interfaces. Automobile operation interface elements such as steering wheels, gearshifts,

1 and speedometers facilitate the access, operation, and display of automobile resources,
2 and status. Computer interaction interface elements such as check boxes, cursors,
3 menus, scrollers, and windows (collectively and commonly referred to as widgets)
4 similarly facilitate the access, capabilities, operation, and display of data and computer
5 hardware and operating system resources, and status. Operation interfaces are
6 commonly called user interfaces. Graphical user interfaces (GUIs) such as the Apple
7 Macintosh Operating System's Aqua and iOS's Cocoa Touch, IBM's OS/2, Google's
8 Android Mobile UI, Microsoft's Windows 2000/2003/3.1/95/98/CE/Millennium/
9 Mobile/NT/XP/Vista/7/8 (i.e., Aero, Metro), Unix's X-Windows (e.g., which may
10 include additional Unix graphic interface libraries and layers such as K Desktop
11 Environment (KDE), mythTV and GNU Network Object Model Environment (GNOME)),
12 web interface libraries (e.g., ActiveX, AJAX, (D)HTML, FLASH, Java, JavaScript, etc.
13 interface libraries such as, but not limited to, Dojo, jQuery(UI), MooTools, Prototype,
14 script.aculo.us, SWFObject, Yahoo! User Interface, any of which may be used and)
15 provide a baseline and means of accessing and displaying information graphically to
16 users.

17 **[00408]** A user interface component 2417 is a stored program component that is
18 executed by a CPU. The user interface may be a graphic user interface as provided by,
19 with, and/or atop operating systems and/or operating environments such as already
20 discussed. The user interface may allow for the display, execution, interaction,
21 manipulation, and/or operation of program components and/or system facilities
22 through textual and/or graphical facilities. The user interface provides a facility through
23 which users may affect, interact, and/or operate a computer system. A user interface
24 may communicate to and/or with other components in a component collection,

1 including itself, and/or facilities of the like. Most frequently, the user interface
2 communicates with operating systems, other program components, and/or the like. The
3 user interface may contain, communicate, generate, obtain, and/or provide program
4 component, system, user, and/or data communications, requests, and/or responses.

5 **Web Browser**

6 **[00409]** A Web browser component 2418 is a stored program component that is
7 executed by a CPU. The Web browser may be a hypertext viewing application such as
8 Goofle's (Mobile) Chrome, Microsoft Internet Explorer, Netscape Navigator, Apple's
9 (Mobile) Safari, embedded web browser objects such as through Apple's Cocoa (Touch)
10 object class, and/or the like. Secure Web browsing may be supplied with 128bit (or
11 greater) encryption by way of HTTPS, SSL, and/or the like. Web browsers allowing for
12 the execution of program components through facilities such as ActiveX, AJAX,
13 (D)HTML, FLASH, Java, JavaScript, web browser plug-in APIs (e.g., Chrome, FireFox,
14 Internet Explorer, Safari Plug-in, and/or the like APIs), and/or the like. Web browsers
15 and like information access tools may be integrated into PDAs, cellular telephones,
16 smartphones, and/or other mobile devices. A Web browser may communicate to and/or
17 with other components in a component collection, including itself, and/or facilities of
18 the like. Most frequently, the Web browser communicates with information servers,
19 operating systems, integrated program components (e.g., plug-ins), and/or the like; e.g.,
20 it may contain, communicate, generate, obtain, and/or provide program component,
21 system, user, and/or data communications, requests, and/or responses. Also, in place of
22 a Web browser and information server, a combined application may be developed to
23 perform similar operations of both. The combined application would similarly effect the

1 obtaining and the provision of information to users, user agents, and/or the like from
2 the RUAP equipped nodes. The combined application may be nugatory on systems
3 employing standard Web browsers.

Mail Server

5 **[00410]** A mail server component 2421 is a stored program component that is
6 executed by a CPU 2403. The mail server may be an Internet mail server such as, but
7 not limited to Apple's Mail Server (3), dovect, sendmail, Microsoft Exchange, and/or the
8 like. The mail server may allow for the execution of program components through
9 facilities such as ASP, ActiveX, (ANSI) (Objective-) C (++), C# and/or .NET, CGI scripts,
10 Java, JavaScript, PERL, PHP, pipes, Python, WebObjects, and/or the like. The mail
11 server may support communications protocols such as, but not limited to: Internet
12 message access protocol (IMAP), Messaging Application Programming Interface
13 (MAPI)/Microsoft Exchange, post office protocol (POP3), simple mail transfer protocol
14 (SMTP), and/or the like. The mail server can route, forward, and process incoming and
15 outgoing mail messages that have been sent, relayed and/or otherwise traversing
16 through and/or to the RUAP.

17 **[00411]** Access to the RUAP mail may be achieved through a number of APIs
18 offered by the individual Web server components and/or the operating system.

19 **[00412]** Also, a mail server may contain, communicate, generate, obtain, and/or
20 provide program component, system, user, and/or data communications, requests,
21 information, and/or responses.

Mail Client

[00413] A mail client component 2422 is a stored program component that is executed by a CPU 2403. The mail client may be a mail viewing application such as Apple (Mobile) Mail, Microsoft Entourage, Microsoft Outlook, Microsoft Outlook Express, Mozilla, Thunderbird, and/or the like. Mail clients may support a number of transfer protocols, such as: IMAP, Microsoft Exchange, POP3, SMTP, and/or the like. A mail client may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the mail client communicates with mail servers, operating systems, other mail clients, and/or the like; e.g., it may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, information, and/or responses. Generally, the mail client provides a facility to compose and transmit electronic mail messages.

Cryptographic Server

[00414] A cryptographic server component 2420 is a stored program component that is executed by a CPU 2403, cryptographic processor 2426, cryptographic processor interface 2427, cryptographic processor device 2428, and/or the like. Cryptographic processor interfaces will allow for expedition of encryption and/or decryption requests by the cryptographic component; however, the cryptographic component, alternatively, may run on a CPU. The cryptographic component allows for the encryption and/or decryption of provided data. The cryptographic component allows for both symmetric and asymmetric (e.g., Pretty Good Protection (PGP)) encryption and/or decryption. The cryptographic component may employ cryptographic techniques such as, but not limited

1 to: digital certificates (e.g., X.509 authentication framework), digital signatures, dual
2 signatures, enveloping, password access protection, public key management, and/or the
3 like. The cryptographic component will facilitate numerous (encryption and/or
4 decryption) security protocols such as, but not limited to: checksum, Data Encryption
5 Standard (DES), Elliptical Curve Encryption (ECC), International Data Encryption
6 Algorithm (IDEA), Message Digest 5 (MD5, which is a one way hash operation),
7 passwords, Rivest Cipher (RC5), Rijndael, RSA (which is an Internet encryption and
8 authentication system that uses an algorithm developed in 1977 by Ron Rivest, Adi
9 Shamir, and Leonard Adleman), Secure Hash Algorithm (SHA), Secure Socket Layer
10 (SSL), Secure Hypertext Transfer Protocol (HTTPS), and/or the like. Employing such
11 encryption security protocols, the RUAP may encrypt all incoming and/or outgoing
12 communications and may serve as node within a virtual private network (VPN) with a
13 wider communications network. The cryptographic component facilitates the process of
14 “security authorization” whereby access to a resource is inhibited by a security protocol
15 wherein the cryptographic component effects authorized access to the secured resource.
16 In addition, the cryptographic component may provide unique identifiers of content,
17 e.g., employing and MD5 hash to obtain a unique signature for an digital audio file. A
18 cryptographic component may communicate to and/or with other components in a
19 component collection, including itself, and/or facilities of the like. The cryptographic
20 component supports encryption schemes allowing for the secure transmission of
21 information across a communications network to enable the RUAP component to
22 engage in secure transactions if so desired. The cryptographic component facilitates the
23 secure accessing of resources on the RUAP and facilitates the access of secured
24 resources on remote systems; i.e., it may act as a client and/or server of secured

1 resources. Most frequently, the cryptographic component communicates with
2 information servers, operating systems, other program components, and/or the like.
3 The cryptographic component may contain, communicate, generate, obtain, and/or
4 provide program component, system, user, and/or data communications, requests,
5 and/or responses.

6

The RUAP Database

7 **[00415]** The RUAP database component 2419 may be embodied in a database and
8 its stored data. The database is a stored program component, which is executed by the
9 CPU; the stored program component portion configuring the CPU to process the stored
10 data. The database may be any of a number of fault tolerant, relational, scalable, secure
11 databases, such as DB2, MySQL, Oracle, Sybase, and/or the like. Relational databases
12 are an extension of a flat file. Relational databases consist of a series of related tables.
13 The tables are interconnected via a key field. Use of the key field allows the combination
14 of the tables by indexing against the key field; i.e., the key fields act as dimensional pivot
15 points for combining information from various tables. Relationships generally identify
16 links maintained between tables by matching primary keys. Primary keys represent
17 fields that uniquely identify the rows of a table in a relational database. More precisely,
18 they uniquely identify rows of a table on the “one” side of a one-to-many relationship.

19 **[00416]** Alternatively, the RUAP database may be implemented using various
20 standard data-structures, such as an array, hash, (linked) list, struct, structured text file
21 (e.g., XML), table, and/or the like. Such data-structures may be stored in memory
22 and/or in (structured) files. In another alternative, an object-oriented database may be
23 used, such as Frontier, ObjectStore, Poet, Zope, and/or the like. Object databases can

1 include a number of object collections that are grouped and/or linked together by
2 common attributes; they may be related to other object collections by some common
3 attributes. Object-oriented databases perform similarly to relational databases with the
4 exception that objects are not just pieces of data but may have other types of capabilities
5 encapsulated within a given object. If the RUAP database is implemented as a data-
6 structure, the use of the RUAP database 2419 may be integrated into another
7 component such as the RUAP component 2435. Also, the database may be implemented
8 as a mix of data structures, objects, and relational structures. Databases may be
9 consolidated and/or distributed in countless variations through standard data
10 processing techniques. Portions of databases, e.g., tables, may be exported and/or
11 imported and thus decentralized and/or integrated.

12 **[00417]** In one embodiment, the database component 2419 includes several tables
13 2419a-q. A Users table 2419a may include fields such as, but not limited to: user_id,
14 applicant_id, firstname, lastname, address_line1, address_line2, dob, ssn,
15 credit_check_flag, zipcode, city, state, account_params_list, account_mode,
16 account_type, account_expiry, preferred_bank_name, preferred_branch_name,
17 credit_report, and/or the like. The User table may support and/or track multiple entity
18 accounts on a RUAP. A Clients table 2419b may include fields such as, but not limited to:
19 client_ID, client_type, client_MAC, client_IP, presentation_format, pixel_count,
20 resolution, screen_size, audio_fidelity, hardware_settings_list, software_
21 compatibilities_list, installed_apps_list, and/or the like. An Apps table 2419c may
22 include fields such as, but not limited to: app_ID, app_name, app_type,
23 OS_compatibilities_list, version, timestamp, developer_ID, and/or the like. An
24 Accounts table 2419d may include fields such as, but not limited to: user_id,

1 account_firstname, account_lastname, account_type, account_num,
2 account_balance_list, billingaddress_line1, billingaddress_line2, billing_zipcode,
3 billing_state, shipping_preferences, shippingaddress_line1, shippingaddress_line2,
4 shipping_zipcode, shipping_state, and/or the like. A Claims table 2419e may include
5 fields such as, but not limited to: user_id, claim_id, timestamp claim_type,
6 snapshot_array, receipt_data, process_sent_flag, process_clear_flag, , and/or the like.
7 An Issuers table 2419f may include fields such as, but not limited to: account_firstname,
8 account_lastname, account_type, account_num, account_balance_list, billingaddress_
9 line1, billingaddress_line2, billing_zipcode, billing_state, shipping_preferences,
10 shippingaddress_line1, shippingaddress_line2, shipping_zipcode, shipping_state,
11 issuer_id, issuer_name, issuer_address, ip_address, mac_address, auth_key,
12 port_num, security_settings_list, and/or the like. A Merchants table 2419g may
13 include fields such as, but not limited to: merchant_id, merchant_name, provi
14 merchant_address, ip_address, mac_address, auth_key, port_num,
15 security_settings_list, and/or the like. An Acquirers table 2419h may include fields
16 such as, but not limited to: account_firstname, account_lastname, account_type,
17 account_num, account_balance_list, billingaddress_line1, billingaddress_line2,
18 billing_zipcode, billing_state, shipping_preferences, shippingaddress_line1,
19 shippingaddress_line2, shipping_zipcode, shipping_state, and/or the like. A
20 Transactions table 2419i may include fields such as, but not limited to: order_id,
21 user_id, timestamp, transaction_cost, purchase_details_list, num_products,
22 products_list, product_type, product_params_list, product_title, product_summary,
23 quantity, user_id, client_id, client_ip, client_type, client_model, operating_system,
24 os_version, app_installed_flag, user_id, account_firstname, account_lastname,

1 account_type, account_num, billingaddress_line1, billingaddress_line2,
2 billing_zipcode, billing_state, shipping_preferences, shippingaddress_line1,
3 shippingaddress_line2, shipping_zipcode, shipping_state, merchant_id,
4 merchant_name, merchant_auth_key, rebate_ID, rebate_amount, rebate_user,
5 rebate_sponsor, social_payment_id, and/or the like. A Batches table 2419j may include
6 fields such as, but not limited to: applicant_firstname, applicant_lastname,
7 applicant_address_line1, applicant_address_line2, consumer_bureau_data_list,
8 consumer_bureau_data, applicant_clear_flag, credit_limit, credit_score,
9 account_balances, delinquency_flag, quality_flags, batch_id, transaction_id_list,
10 timestamp_list, cleared_flag_list, clearance_trigger_settings, and/or the like. A
11 Ledgers table 2419k may include fields such as, but not limited to: request_id,
12 timestamp, deposit_amount, batch_id, transaction_id, clear_flag, deposit_account,
13 transaction_summary, payor_name, payor_account, and/or the like. An Insurance
14 Provider table 2419l may include fields such as, but not limited to InsuranceID,
15 InsuranceName, InsuranceProgram, InsuranceBIN, InsuranceCoverageTable,
16 InsuranceVeriCode, InsuranceAuthorization, and/or the like. A Healthcare Provider
17 table 2419m may include fields such as, but not limited to HealthProviderID,
18 HealthProviderName, HealthProviderZipcode, HealthProviderProcedureCode,
19 HealthProviderClaimCode, HealthProviderPricingList, and/or the like. A medical
20 products/services table 2419n may include fields such as, but not limited to
21 authorizedMedProductID, authorizedMedServiceID, ProductCode,
22 ServiceProcedureCode, HealthProviderID, InsuranceID, InsuranceCoverageRate,
23 PricingRate, and/or the like. A Restricted-Use Codes table 2419o may include fields
24 such as, but not limited to ru_type, ru_issuer, ru_category, ru_mcc, ru_sponsor,

1 ru_rule, ru_deduction, ru_alert, ru_account, ru_whitelist, ru_blacklist, and/or the like.
2 A Receipt table 2419p may include fields such as, but not limited to receipt_id,
3 receipt_user_id, receipt_wallet_id, receipt_time, receipt_merchant_id, receipt_image,
4 receipt_item, receipt_mcc, receipt_amount, receipt_transaction_id,
5 receipt_transaction_amount, receipt_claim, receipt_qr_code, receipt_item_ru, and/or
6 the like. A Bill table 2419q may include fields such as, but not limited to bill_id,
7 bill_user_id, bill_wallet_id, bill_time, bill_merchant_id, bill_image, bill_item,
8 bill_mcc, bill_amount, bill_transaction_id, bill_transaction_amount, bill_claim,
9 bill_qr_code, bill_item_ru, and/or the like.

10 **[00418]** In one embodiment, the RUAP database may interact with other database
11 systems. For example, employing a distributed database system, queries and data access
12 by search RUAP component may treat the combination of the RUAP database, an
13 integrated data security layer database as a single database entity.

14 **[00419]** In one embodiment, user programs may contain various user interface
15 primitives, which may serve to update the RUAP. Also, various accounts may require
16 custom database tables depending upon the environments and the types of clients the
17 RUAP may need to serve. It should be noted that any unique fields may be designated as
18 a key field throughout. In an alternative embodiment, these tables have been
19 decentralized into their own databases and their respective database controllers (i.e.,
20 individual database controllers for each of the above tables). Employing standard data
21 processing techniques, one may further distribute the databases over several computer
22 systemizations and/or storage devices. Similarly, configurations of the decentralized
23 database controllers may be varied by consolidating and/or distributing the various

1 database components 2419a-q. The RUAP may be configured to keep track of various
2 settings, inputs, and parameters via database controllers.

3 **[00420]** The RUAP database may communicate to and/or with other components
4 in a component collection, including itself, and/or facilities of the like. Most frequently,
5 the RUAP database communicates with the RUAP component, other program
6 components, and/or the like. The database may contain, retain, and provide
7 information regarding other nodes and data.

8 **The RUAPs**

9 **[00421]** The RUAP component 2435 is a stored program component that is
10 executed by a CPU. In one embodiment, the RUAP component incorporates any and/or
11 all combinations of the aspects of the RUAP discussed in the previous figures. As such,
12 the RUAP affects accessing, obtaining and the provision of information, services,
13 transactions, and/or the like across various communications networks. The features
14 and embodiments of the RUAP discussed herein increase network efficiency by reducing
15 data transfer requirements the use of more efficient data structures and mechanisms for
16 their transfer and storage. As a consequence, more data may be transferred in less time,
17 and latencies with regard to transactions, are also reduced. In many cases, such
18 reduction in storage, transfer time, bandwidth requirements, latencies, etc., will reduce
19 the capacity and structural infrastructure requirements to support the RUAP's features
20 and facilities, and in many cases reduce the costs, energy consumption/requirements,
21 and extend the life of RUAP's underlying infrastructure; this has the added benefit of
22 making the RUAP more reliable. Similarly, many of the features and mechanisms are
23 designed to be easier for users to use and access, thereby broadening the audience that

1 may enjoy/employ and exploit the feature sets of the RUAP; such ease of use also helps
2 to increase the reliability of the RUAP. In addition, the feature sets include heightened
3 security as noted via the Cryptographic components 2420, 2426, 2428 and throughout,
4 making access to the features and data more reliable and secure.

5 **[00422]** The RUAP transforms purchase item information inputs (e.g., see 203 in
6 FIGURE 2A) or purchase receipt inputs (e.g., see 411 in FIGURE 4A) via RUAP
7 components such as user enrollment component 2442 (e.g., FIGURES 8A-8B), card
8 processing component 2443 (e.g., see FIGURES 17A-21), insurance
9 authorization/adjudication component 2444 (e.g., see FIGURES 7A-7B), account
10 recommendation component 2445 (e.g., see FIGURES 5D-5F), receipt/bill image
11 processing component 2446 (e.g., see FIGURES 6A-6B), funds allocation
12 (reimbursement processing) component 2447 (e.g., see FIGURES 4A-5C), restricted-use
13 managing component 2449 (e.g., see FIGURE 3C), PTA component 2451 (e.g., see
14 FIGURES 18A-18B), PTC component 2452 (e.g., see FIGURES 19A-19B), TDA
15 component 2453 (e.g., see FIGURE 21) and/or the like into restricted-use account
16 payment settlement outputs (e.g., see 226 in FIGURE 2A; 766 in FIGURE 7A, and/or
17 the like).

18 **[00423]** The RUAP component enabling access of information between nodes may
19 be developed by employing standard development tools and languages such as, but not
20 limited to: Apache components, Assembly, ActiveX, binary executables, (ANSI)
21 (Objective-) C (++), C# and/or .NET, database adapters, CGI scripts, Java, JavaScript,
22 mapping tools, procedural and object oriented development tools, PERL, PHP, Python,
23 shell scripts, SQL commands, web application server extensions, web development

1 environments and libraries (e.g., Microsoft's ActiveX; Adobe AIR, FLEX & FLASH;
2 AJAX; (D)HTML; Dojo, Java; JavaScript; jQuery(UI); MooTools; Prototype;
3 script.aculo.us; Simple Object Access Protocol (SOAP); SWFObject; Yahoo! User
4 Interface; and/or the like), WebObjects, and/or the like. In one embodiment, the RUAP
5 server employs a cryptographic server to encrypt and decrypt communications. The
6 RUAP component may communicate to and/or with other components in a component
7 collection, including itself, and/or facilities of the like. Most frequently, the RUAP
8 component communicates with the RUAP database, operating systems, other program
9 components, and/or the like. The RUAP may contain, communicate, generate, obtain,
10 and/or provide program component, system, user, and/or data communications,
11 requests, and/or responses.

12 **Distributed RUAPs**

13 **[00424]** The structure and/or operation of any of the RUAP node controller
14 components may be combined, consolidated, and/or distributed in any number of ways
15 to facilitate development and/or deployment. Similarly, the component collection may
16 be combined in any number of ways to facilitate deployment and/or development. To
17 accomplish this, one may integrate the components into a common code base or in a
18 facility that can dynamically load the components on demand in an integrated fashion.

19 **[00425]** The component collection may be consolidated and/or distributed in
20 countless variations through standard data processing and/or development techniques.
21 Multiple instances of any one of the program components in the program component
22 collection may be instantiated on a single node, and/or across numerous nodes to
23 improve performance through load-balancing and/or data-processing techniques.

1 Furthermore, single instances may also be distributed across multiple controllers
2 and/or storage devices; e.g., databases. All program component instances and
3 controllers working in concert may do so through standard data processing
4 communication techniques.

5 **[00426]** The configuration of the RUAP controller will depend on the context of
6 system deployment. Factors such as, but not limited to, the budget, capacity, location,
7 and/or use of the underlying hardware resources may affect deployment requirements
8 and configuration. Regardless of if the configuration results in more consolidated
9 and/or integrated program components, results in a more distributed series of program
10 components, and/or results in some combination between a consolidated and
11 distributed configuration, data may be communicated, obtained, and/or provided.
12 Instances of components consolidated into a common code base from the program
13 component collection may communicate, obtain, and/or provide data. This may be
14 accomplished through intra-application data processing communication techniques
15 such as, but not limited to: data referencing (e.g., pointers), internal messaging, object
16 instance variable communication, shared memory space, variable passing, and/or the
17 like.

18 **[00427]** If component collection components are discrete, separate, and/or
19 external to one another, then communicating, obtaining, and/or providing data with
20 and/or to other components may be accomplished through inter-application data
21 processing communication techniques such as, but not limited to: Application Program
22 Interfaces (API) information passage; (distributed) Component Object Model ((D)COM),
23 (Distributed) Object Linking and Embedding ((D)OLE), and/or the like), Common

Object Request Broker Architecture (CORBA), Jini local and remote application program interfaces, JavaScript Object Notation (JSON), Remote Method Invocation (RMI), SOAP, process pipes, shared files, and/or the like. Messages sent between discrete component components for inter-application communication or within memory spaces of a singular component for intra-application communication may be facilitated through the creation and parsing of a grammar. A grammar may be developed by using development tools such as lex, yacc, XML, and/or the like, which allow for grammar generation and parsing capabilities, which in turn may form the basis of communication messages within and between components.

¹⁰ **[00428]** For example, a grammar may be arranged to recognize the tokens of an

¹¹ HTTP post command, e.g.:

[illegible]

14 **[00429]** where Value1 is discerned as being a parameter because “http://” is part of
15 the grammar syntax, and what follows is considered part of the post value. Similarly,
16 with such a grammar, a variable “Value1” may be inserted into an “http://” post
17 command and then sent. The grammar syntax itself may be presented as structured data
18 that is interpreted and/or otherwise used to generate the parsing mechanism (e.g., a
19 syntax description text file as processed by lex, yacc, etc.). Also, once the parsing
20 mechanism is generated and/or instantiated, it itself may process and/or parse
21 structured data such as, but not limited to: character (e.g., tab) delineated text, HTML,
22 structured text streams, XML, and/or the like structured data. In another embodiment,
23 inter-application data processing protocols themselves may have integrated and/or
24 readily available parsers (e.g., JSON, SOAP, and/or like parsers) that may be employed
25 to parse (e.g., communications) data. Further, the parsing grammar may be used

beyond message parsing, but may also be used to parse: databases, data collections, data stores, structured data, and/or the like. Again, the desired configuration will depend upon the context, environment, and requirements of system deployment.

[00430] For example, in some implementations, the RUAP controller may be executing a PHP script implementing a Secure Sockets Layer (“SSL”) socket server via the information server, which listens to incoming communications on a server port to which a client may send data, e.g., data encoded in JSON format. Upon identifying an incoming communication, the PHP script may read the incoming message from the client device, parse the received JSON-encoded text data to extract information from the JSON-encoded text data into PHP script variables, and store the data (e.g., client identifying information, etc.) and/or extracted information in a relational database accessible using the Structured Query Language (“SQL”). An exemplary listing, written substantially in the form of PHP/SQL commands, to accept JSON-encoded input data from a client device via a SSL connection, parse the data to extract variables, and store the data to a database, is provided below:

```
16 <?PHP
17 header('Content-Type: text/plain');
18
19 // set ip address and port to listen to for incoming data
20 $address = '192.168.0.100';
21 $port = 255;
22
23 // create a server-side SSL socket, listen for/accept incoming communication
24 $sock = socket_create(AF_INET, SOCK_STREAM, 0);
25 socket_bind($sock, $address, $port) or die('Could not bind to address');
26 socket_listen($sock);
27 $client = socket_accept($sock);
28
29 // read input data from client device in 1024 byte blocks until end of message
30 do {
31     $input = "";
32     $input = socket_read($client, 1024);
33     $data .= $input;
34 } while($input != "");
35
36 // parse data to extract variables
37 $obj = json_decode($data, true);
```

```
1
2 // store input data in a database
3 mysql_connect("201.408.185.132",$DBserver,$password); // access database server
4 mysql_select("CLIENT_DB.SQL"); // select database to append
5 mysql_query("INSERT INTO UserTable (transmission)
6 VALUES ($data)"); // add data to UserTable table in a CLIENT database
7 mysql_close("CLIENT_DB.SQL"); // close connection to database
8 ?>
9
```

10 **[00431]** Also, the following resources may be used to provide example
11 embodiments regarding SOAP parser implementation:

```
12 http://www.xav.com/perl/site/lib/SOAP/Parser.html
13 http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.IBMDI.
14 doc/referenceguide295.htm
15
```

16 **[00432]** and other parser implementations:

```
17 http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=/com.ibm.IBMDI.
18 doc/referenceguide259.htm
19
```

20 **[00433]** all of which are hereby expressly incorporated by reference herein.

21 **[00434]** Non-limiting exemplary embodiments highlighting numerous further
22 advantageous aspects include:

23 **[00435]** In order to address various issues and advance the art, the entirety of this
24 application for RESTRICTED-USE ACCOUNT PAYMENT ADMINISTRATION
25 APPARATUSES, METHODS AND SYSTEMS APPARATUSES, METHODS AND
26 SYSTEMS (including the Cover Page, Title, Headings, Field, Background, Summary,
27 Brief Description of the Drawings, Detailed Description, Claims, Abstract, Figures,
28 Appendices and/or otherwise) shows by way of illustration various example
29 embodiments in which the claimed innovations may be practiced. The advantages and
30 features of the application are of a representative sample of embodiments only, and are
31 not exhaustive and/or exclusive. They are presented only to assist in understanding and
32 teach the claimed principles. It should be understood that they are not representative of
33 all claimed innovations. As such, certain aspects of the disclosure have not been

1 discussed herein. That alternate embodiments may not have been presented for a
2 specific portion of the innovations or that further undescribed alternate embodiments
3 may be available for a portion is not to be considered a disclaimer of those alternate
4 embodiments. It will be appreciated that many of those undescribed embodiments
5 incorporate the same principles of the innovations and others are equivalent. Thus, it is
6 to be understood that other embodiments may be utilized and functional, logical,
7 operational, organizational, structural and/or topological modifications may be made
8 without departing from the scope and/or spirit of the disclosure. As such, all examples
9 and/or embodiments are deemed to be non-limiting throughout this disclosure. Also, no
10 inference should be drawn regarding those embodiments discussed herein relative to
11 those not discussed herein other than it is as such for purposes of reducing space and
12 repetition. For instance, it is to be understood that the logical and/or topological
13 structure of any combination of any data flow sequence(s), program components (a
14 component collection), other components and/or any present feature sets as described
15 in the figures and/or throughout are not limited to a fixed operating order and/or
16 arrangement, but rather, any disclosed order is exemplary and all equivalents,
17 regardless of order, are contemplated by the disclosure. Furthermore, it is to be
18 understood that such features are not limited to serial execution, but rather, any
19 number of threads, processes, processors, services, servers, and/or the like that may
20 execute asynchronously, concurrently, in parallel, simultaneously, synchronously,
21 and/or the like are also contemplated by the disclosure. As such, some of these features
22 may be mutually contradictory, in that they cannot be simultaneously present in a single
23 embodiment. Similarly, some features are applicable to one aspect of the innovations,
24 and inapplicable to others. In addition, the disclosure includes other innovations not

1 presently claimed. Applicant reserves all rights in those presently unclaimed
2 innovations, including the right to claim such innovations, file additional applications,
3 continuations, continuations-in-part, divisions, and/or the like thereof. As such, it
4 should be understood that advantages, embodiments, examples, functional, features,
5 logical, operational, organizational, structural, topological, and/or other aspects of the
6 disclosure are not to be considered limitations on the disclosure as defined by the claims
7 or limitations on equivalents to the claims. It is to be understood that, depending on the
8 particular needs and/or characteristics of a RUAP individual and/or enterprise user,
9 database configuration and/or relational model, data type, data transmission and/or
10 network framework, syntax structure, and/or the like, various embodiments of the
11 RUAP may be implemented that allow a great deal of flexibility and customization. For
12 example, aspects of the RUAP may be adapted for various card management and
13 secured payment processing, offer/coupon distribution and redemption, social payment,
14 and/or the like. While various embodiments and discussions of the RUAP have been
15 directed to restricted-use account payment, however, it is to be understood that the
16 embodiments described herein may be readily configured and/or customized for a wide
17 variety of other applications and/or implementations.

CLAIMS

What is claimed is:

1. A processor-implemented restricted-use account reimbursement management method, comprising:

receiving a restricted-use account reimbursement request including receipt information related to a purchase transaction from a user;

obtaining purchase item information from the receipt information included in the restricted-use account reimbursement request;

identifying a restricted-use account of the user;

determining a purchase item is eligible for the restricted-account usage;

determining a reimbursement amount associated with the purchase item from the purchase item information;

generating a reimbursement authorization request message including the purchase item information and the reimbursement amount;

transmitting the reimbursement authorization request message to a restricted-account issuer for approval; and

transacting the reimbursement amount from the restricted-use account to a user financial account upon the restricted-account issuer approval.

2. The method of claim 1, wherein the receipt information related to a purchase transaction comprises an image of a purchase receipt.

3. The method of claim 2, wherein the image of the purchase receipt is capture by a user mobile device.

1
2 4. The method of claim 1, wherein the restricted-use account reimbursement
3 request is received via a user electronic wallet instantiated on a user mobile device.

4
5 5. The method of claim 4, wherein the user mobile device includes any of a
6 smart phone, a laptop computer, and a personal digital assistant.

7
8 6. The method of claim 1, wherein the restricted-use account reimbursement
9 request is received via a web-based application.

10
11 7. The method of claim 1, wherein the restricted-use account reimbursement
12 request comprises user credentials including a user name and a password.

13
14 8. The method of claim 1, wherein the restricted-use account comprises any
15 of: a Flexible Spending Account (FSA), and a Health Savings Accounts (HSA).

16
17 9. The method of claim 1, wherein the restricted use account comprises one
18 or more government sponsored accounts, including any of Medicare and Medicaid.

19
20 10. The method of claim 1, wherein the restricted use account comprises an
21 unemployment benefit account.

22
23 11. The method of claim 1, wherein the restricted-use account comprises any
24 of a food stamp and a food voucher.

1

2 12. The method of claim 1, wherein the purchase item information includes a
3 merchant category code.

4

5 13. The method of claim 1, wherein identifying a restricted-use account of the
6 user comprises retrieving a user enrolled restricted-use account information from a user
7 electronic wallet.

8

9 14. The method of claim 1, wherein obtaining purchase item information
10 comprises performing an optical character recognition procedure on a receipt image.

11

12 15. The method of claim 1, wherein obtaining purchase item information
13 comprises decoding a quick response code included in the receipt information.

14

15 16. The method of claim 1, wherein the purchase item information includes a
16 payment amount associated with the purchase item.

17

18 17. The method of claim 1, wherein the reimbursement amount equals a
19 payment amount obtained from the receipt information.

20

21 18. The method of claim 1, further comprising:
22 obtaining user insurance information;
23 determining the purchase item is a healthcare related product; and
24 generating an insurance claim request to an insurance provider.

1
2 19. The method of claim 1, further comprising:
3 receiving an insurance refund from the insurance provider.
4

5 20. The method of claim 19, further comprising:
6 calculating the reimbursement amount as a payment amount of the purchase
7 item minus the insurance refund.
8

9 21. The method of claim 1, wherein the account issuer verifies the purchase
10 item is eligible for usage with the restricted-use account.
11

12 22. The method of claim 1, wherein the account issuer determines there is
13 sufficient remaining balance to reimburse for the reimbursement amount.
14

15 23. The method of claim 1, further comprising:
16 generating a restricted-use account recommendation list when the purchase item
17 is eligible for more than one restricted-use accounts.
18

19 24. The method of claim 23, wherein the restricted-use account
20 recommendation list is generated based on user configured restricted-use account usage
21 rules.
22

23 25. The method of claim 23, wherein the wherein the restricted-use account
24 recommendation list is generated based on available balance of each restricted-use

1 account.

2

3 26. The method of claim 23, wherein the restricted-use account
4 recommendation list ranks accounts in a prioritized order.

5

6 27. The method of claim 1, wherein the restricted-use account reimbursement
7 account further comprises a user selected account for depositing reimbursed funds.

8

9 28. The method of claim 1, further comprising:
10 obtaining a payment account from the receipt information; and
11 depositing reimbursed funds into the payment account.

12

13 29. The method of claim 28, wherein the payment account comprises any of a
14 credit card account, a debit account and a personal checking account.

15

16 30. The method of claim 1, wherein the purchase item information includes
17 purchase items eligible for the restricted-use account and purchase items ineligible for
18 the restricted-use account.

19

20 31. A restricted-use account reimbursement management system, comprising:
21 a memory;
22 a processor disposed in communication with said memory, and configured to
23 issue a plurality of processing instructions stored in the memory, wherein the processor
24 issues instructions to:

1 receive a restricted-use account reimbursement request including receipt
2 information related to a purchase transaction from a user;
3 obtain purchase item information from the receipt information included in
4 the restricted-use account reimbursement request;
5 identify a restricted-use account of the user;
6 determine a purchase item is eligible for the restricted-account usage;
7 determine a reimbursement amount associated with the purchase item
8 form the purchase item information;
9 generate a reimbursement authorization request message including the
10 purchase item information and the reimbursement amount;
11 transmit the reimbursement authorization request message to a restricted-
12 account issuer for approval; and
13 transact the reimbursement amount from the restricted-use account to a
14 user financial account upon the restricted-account issuer approval.

15

16 32. The system of claim 31, wherein the receipt information related to a
17 purchase transaction comprises an image of a purchase receipt.

18

19 33. The system of claim 32, wherein the image of the purchase receipt is
20 capture by a user mobile device.

21

22 34. The system of claim 31, wherein the restricted-use account reimbursement
23 request is received via a user electronic wallet instantiated on a user mobile device.

24

1 35. The system of claim 34, wherein the user mobile device includes any of a
2 smart phone, a laptop computer, and a personal digital assistant.

3
4 36. The system of claim 31, wherein the restricted-use account reimbursement
5 request is received via a web-based application.

6
7 37. The system of claim 31, wherein the restricted-use account reimbursement
8 request comprises user credentials including a user name and a password.

9
10 38. The system of claim 31, wherein the restricted-use account comprises any
11 of: a Flexible Spending Account (FSA), and a Health Savings Accounts (HSA).

12
13 39. The system of claim 31, wherein the restricted use account comprises one
14 or more government sponsored accounts, including any of Medicare and Medicaid.

15
16 40. The system of claim 31, wherein the restricted use account comprises an
17 unemployment benefit account.

18
19 41. The system of claim 31, wherein the restricted-use account comprises any
20 of a food stamp and a food voucher.

21
22 42. The system of claim 31, wherein the purchase item information includes a
23 merchant category code.

24

1 43. The system of claim 31, wherein identifying a restricted-use account of the
2 user comprises retrieving a user enrolled restricted-use account information from a user
3 electronic wallet.

4
5 44. The system of claim 31, wherein obtaining purchase item information
6 comprises performing an optical character recognition procedure on a receipt image.

7
8 45. The system of claim 31, wherein obtaining purchase item information
9 comprises decoding a quick response code included in the receipt information.

10
11 46. The system of claim 31, wherein the purchase item information includes a
12 payment amount associated with the purchase item.

13
14 47. The system of claim 31, wherein the reimbursement amount equals a
15 payment amount obtained from the receipt information.

16
17 48. The system of claim 31, wherein the processor further issues instructions
18 to:

19 obtain user insurance information;
20 determine the purchase item is a healthcare related product; and
21 generate an insurance claim request to an insurance provider.

22
23 49. The system of claim 31, wherein the processor further issues instructions
24 to:

1 receive an insurance refund from the insurance provider.

2
3 50. The system of claim 49, wherein the processor further issues instructions
4 to:

5 calculating the reimbursement amount as a payment amount of the purchase
6 item minus the insurance refund.

7
8 51. The system of claim 31, wherein the account issuer verifies the purchase
9 item is eligible for usage with the restricted-use account.

10
11 52. The system of claim 31, wherein the account issuer determines there is
12 sufficient remaining balance to reimburse for the reimbursement amount.

13
14 53. The system of claim 31, wherein the processor further issues instructions
15 to:

16 generate a restricted-use account recommendation list when the purchase item is
17 eligible for more than one restricted-use accounts.

18
19 54. The system of claim 53, wherein the restricted-use account
20 recommendation list is generated based on user configured restricted-use account usage
21 rules.

22
23 55. The system of claim 53, wherein the wherein the restricted-use account
24 recommendation list is generated based on available balance of each restricted-use

1 account.

2

3 56. The system of claim 53, wherein the restricted-use account
4 recommendation list ranks accounts in a prioritized order.

5

6 57. The system of claim 31, wherein the restricted-use account reimbursement
7 account further comprises a user selected account for depositing reimbursed funds.

8

9 58. The system of claim 31, wherein the processor further issues instructions
10 to:

11 obtain a payment account from the receipt information; and

12 deposit reimbursed funds into the payment account.

13

14 59. The system of claim 58, wherein the payment account comprises any of a
15 credit card account, a debit account and a personal checking account.

16

17 60. The system of claim 31, wherein the purchase item information includes
18 purchase items eligible for the restricted-use account and purchase items ineligible for
19 the restricted-use account.

20

21 61. A restricted-use account reimbursement management processor-readable
22 non-transitory medium storing processor-executable instructions executable by a
23 processor to:

24 receive a restricted-use account reimbursement request including receipt

1 information related to a purchase transaction from a user;
2 obtain purchase item information from the receipt information included in
3 the restricted-use account reimbursement request;
4 identify a restricted-use account of the user;
5 determine a purchase item is eligible for the restricted-account usage;
6 determine a reimbursement amount associated with the purchase item
7 form the purchase item information;
8 generate a reimbursement authorization request message including the
9 purchase item information and the reimbursement amount;
10 transmit the reimbursement authorization request message to a restricted-
11 account issuer for approval; and
12 transact the reimbursement amount from the restricted-use account to a
13 user financial account upon the restricted-account issuer approval.

14

15 62. The medium of claim 61, wherein the receipt information related to a
16 purchase transaction comprises an image of a purchase receipt.

17

18 63. The medium of claim 62, wherein the image of the purchase receipt is
19 capture by a user mobile device.

20

21 64. The medium of claim 61, wherein the restricted-use account
22 reimbursement request is received via a user electronic wallet instantiated on a user
23 mobile device.

24

1 65. The medium of claim 64, wherein the user mobile device includes any of a
2 smart phone, a laptop computer, and a personal digital assistant.

3
4 66. The medium of claim 61, wherein the restricted-use account
5 reimbursement request is received via a web-based application.

6

7 67. The medium of claim 61, wherein the restricted-use account
8 reimbursement request comprises user credentials including a user name and a
9 password.

10

11 68. The medium of claim 61, wherein the restricted-use account comprises any
12 of: a Flexible Spending Account (FSA), and a Health Savings Accounts (HSA).

13

14 69. The medium of claim 61, wherein the restricted use account comprises one
15 or more government sponsored accounts, including any of Medicare and Medicaid.

16

17 70. The medium of claim 61, wherein the restricted use account comprises an
18 unemployment benefit account.

19

20 71. The medium of claim 61, wherein the restricted-use account comprises any
21 of a food stamp and a food voucher.

22

23 72. The medium of claim 61, wherein the purchase item information includes
24 a merchant category code.

1
2 73. The medium of claim 61, wherein identifying a restricted-use account of
3 the user comprises retrieving a user enrolled restricted-use account information from a
4 user electronic wallet.

5
6 74. The medium of claim 61, wherein obtaining purchase item information
7 comprises performing an optical character recognition procedure on a receipt image.

8
9 75. The medium of claim 61, wherein obtaining purchase item information
10 comprises decoding a quick response code included in the receipt information.

11
12 76. The medium of claim 61, wherein the purchase item information includes
13 a payment amount associated with the purchase item.

14
15 77. The medium of claim 61, wherein the reimbursement amount equals a
16 payment amount obtained from the receipt information.

17
18 78. The medium of claim 61, further storing instructions executable by the
19 processor to:

20 obtain user insurance information;

21 determine the purchase item is a healthcare related product; and

22 generate an insurance claim request to an insurance provider.

23
24 79. The medium of claim 61, further storing instructions executable by the

1 processor to:

2 receive an insurance refund from the insurance provider.

3

4 80. The medium of claim 79, further storing instructions executable by the
5 processor to:

6 calculating the reimbursement amount as a payment amount of the purchase
7 item minus the insurance refund.

8

9 81. The medium of claim 61, wherein the account issuer verifies the purchase
10 item is eligible for usage with the restricted-use account.

11

12 82. The medium of claim 61, wherein the account issuer determines there is
13 sufficient remaining balance to reimburse for the reimbursement amount.

14

15 83. The medium of claim 61, further storing instructions executable by the
16 processor to:

17 generate a restricted-use account recommendation list when the purchase item is
18 eligible for more than one restricted-use accounts.

19

20 84. The medium of claim 83, wherein the restricted-use account
21 recommendation list is generated based on user configured restricted-use account usage
22 rules.

23

24 85. The medium of claim 83, wherein the wherein the restricted-use account

1 recommendation list is generated based on available balance of each restricted-use
2 account.

3

4 86. The medium of claim 83, wherein the restricted-use account
5 recommendation list ranks accounts in a prioritized order.

6

7 87. The medium of claim 61, wherein the restricted-use account
8 reimbursement account further comprises a user selected account for depositing
9 reimbursed funds.

10

11 88. The medium of claim 61, further storing instructions executable by the
12 processor to:

13 obtain a payment account from the receipt information; and
14 deposit reimbursed funds into the payment account.

15

16 89. The medium of claim 58, wherein the payment account comprises any of a
17 credit card account, a debit account and a personal checking account.

18

19 90. The medium of claim 61, wherein the purchase item information includes
20 purchase items eligible for the restricted-use account and purchase items ineligible for
21 the restricted-use account.

22

23 91. A processor-implemented restricted-account payment method, comprising:
24 receiving a purchase checkout request including purchase item information;

1 obtaining purchase item category information from the purchase item
2 information;
3 retrieving a restricted-account registered by a user;
4 determining at least one purchase item qualifies for the restricted-account usage
5 based on the obtained purchase item category information;
6 generating an inquiry to the user notifying eligibility of the purchase item for the
7 restricted-use account;
8 receiving a user account selection for the purchase checkout;
9 determining the user account selection indicates the restricted-use account; and
10 processing a financial transaction with the restricted-use account for the at least
11 one purchase item.

12

13 92. A restricted-account payment system, comprising:
14 a memory;
15 a processor disposed in communication with said memory, and configured to
16 issue a plurality of processing instructions stored in the memory, wherein the processor
17 issues instructions to:
18 receive a purchase checkout request including purchase item information;
19 obtain purchase item category information from the purchase item information;
20 retrieve a restricted-account registered by a user;
21 determine at least one purchase item qualifies for the restricted-account usage
22 based on the obtained purchase item category information;
23 generate an inquiry to the user notifying eligibility of the purchase item for the
24 restricted-use account;

1 receive a user account selection for the purchase checkout;
2 determine the user account selection indicates the restricted-use account; and
3 process a financial transaction with the restricted-use account for the at least one
4 purchase item.

5
6 93. A restricted-account payment processor-readable non-transitory medium
7 storing instructions executable by a processor to:

8 receive a purchase checkout request including purchase item information;
9 obtain purchase item category information from the purchase item information;
10 retrieve a restricted-account registered by a user;
11 determine at least one purchase item qualifies for the restricted-account usage
12 based on the obtained purchase item category information;
13 generate an inquiry to the user notifying eligibility of the purchase item for the
14 restricted-use account;

15 receive a user account selection for the purchase checkout;
16 determine the user account selection indicates the restricted-use account; and
17 process a financial transaction with the restricted-use account for the at least one
18 purchase item.

19
20 94. A processor-implemented restricted-account payment method, comprising:
21 obtaining, at a user device, a user input to initiate a purchase transaction;
22 acquiring an image frame via an image acquisition device operatively connected
23 to the user device;
24 identifying a payment code depicted within the acquired image frame;

1 obtaining purchase item information based on the identified payment code;
2 determining a purchase item is qualified for a restricted-use account;
3 generating an account recommendation indication including the restricted-use
4 account;
5 obtaining a user account selection including the restricted-use account in
6 response to the account recommendation indication;
7 generating, via the user device, a purchase transaction request using the
8 identified payment code;
9 providing the purchase transaction request for payment processing; and
10 obtaining a purchase receipt for the purchase transaction.

11
12 95. A restricted-account payment system, comprising:

13 a memory;

14 a processor disposed in communication with said memory, and configured to
15 issue a plurality of processing instructions stored in the memory, wherein the processor
16 issues instructions to:

17 obtain, at a user device, a user input to initiate a purchase transaction;

18 acquire an image frame via an image acquisition device operatively connected to
19 the user device;

20 identify a payment code depicted within the acquired image frame;

21 obtain purchase item information based on the identified payment code;

22 determine a purchase item is qualified for a restricted-use account;

23 generate an account recommendation indication including the restricted-use
24 account;

1 obtain a user account selection including the restricted-use account in response
2 to the account recommendation indication;
3 generate, via the user device, a purchase transaction request using the identified
4 payment code;
5 provide the purchase transaction request for payment processing; and
6 obtain a purchase receipt for the purchase transaction.
7
8 96. A restricted-account payment processor-readable non-transitory medium
9 storing instructions executable by a processor to:
10 obtain, at a user device, a user input to initiate a purchase transaction;
11 acquire an image frame via an image acquisition device operatively connected to
12 the user device;
13 identify a payment code depicted within the acquired image frame;
14 obtain purchase item information based on the identified payment code;
15 determine a purchase item is qualified for a restricted-use account;
16 generate an account recommendation indication including the restricted-use
17 account;
18 obtain a user account selection including the restricted-use account in response
19 to the account recommendation indication;
20 generate, via the user device, a purchase transaction request using the identified
21 payment code;
22 provide the purchase transaction request for payment processing; and
23 obtain a purchase receipt for the purchase transaction.
24

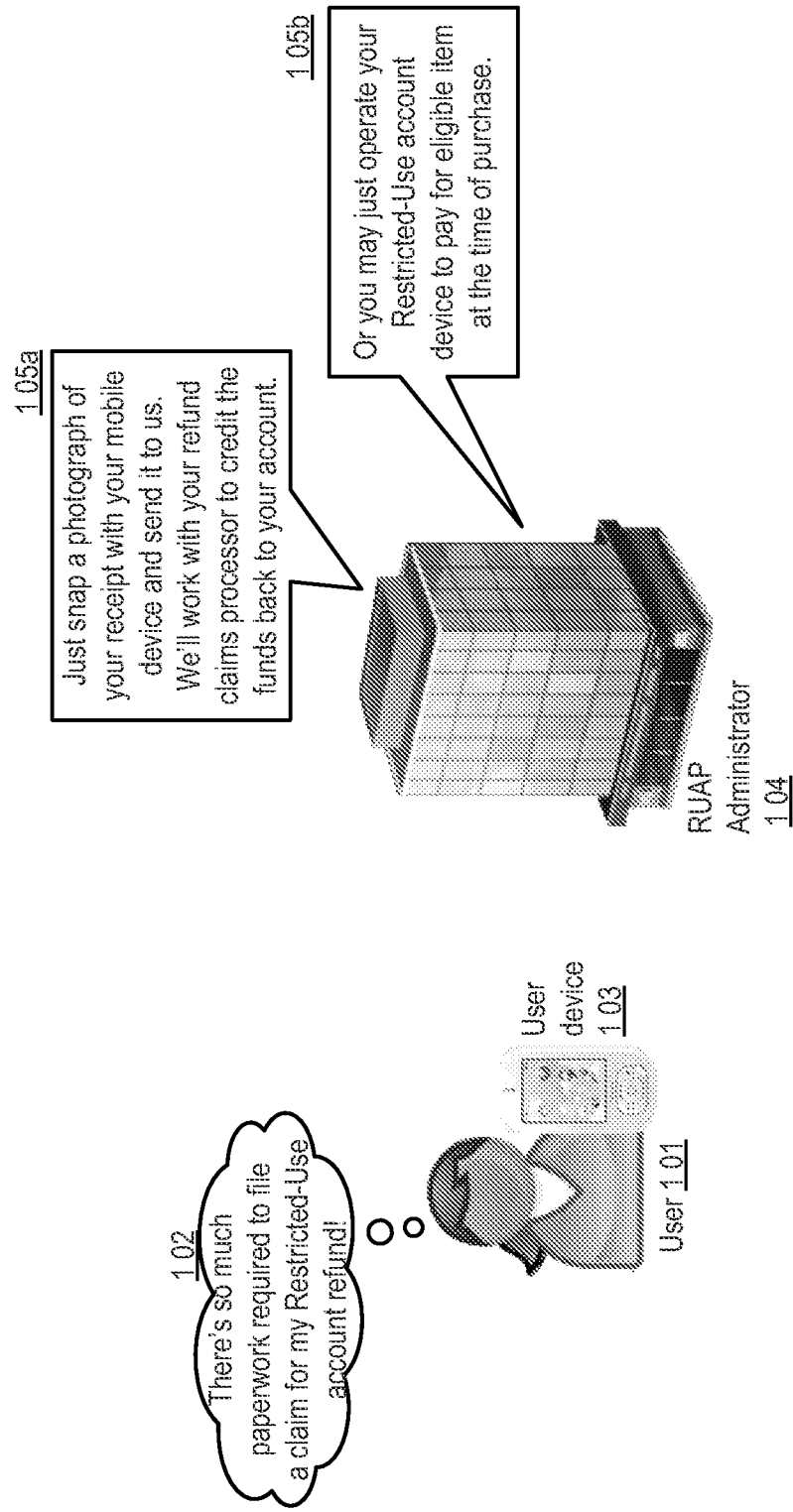
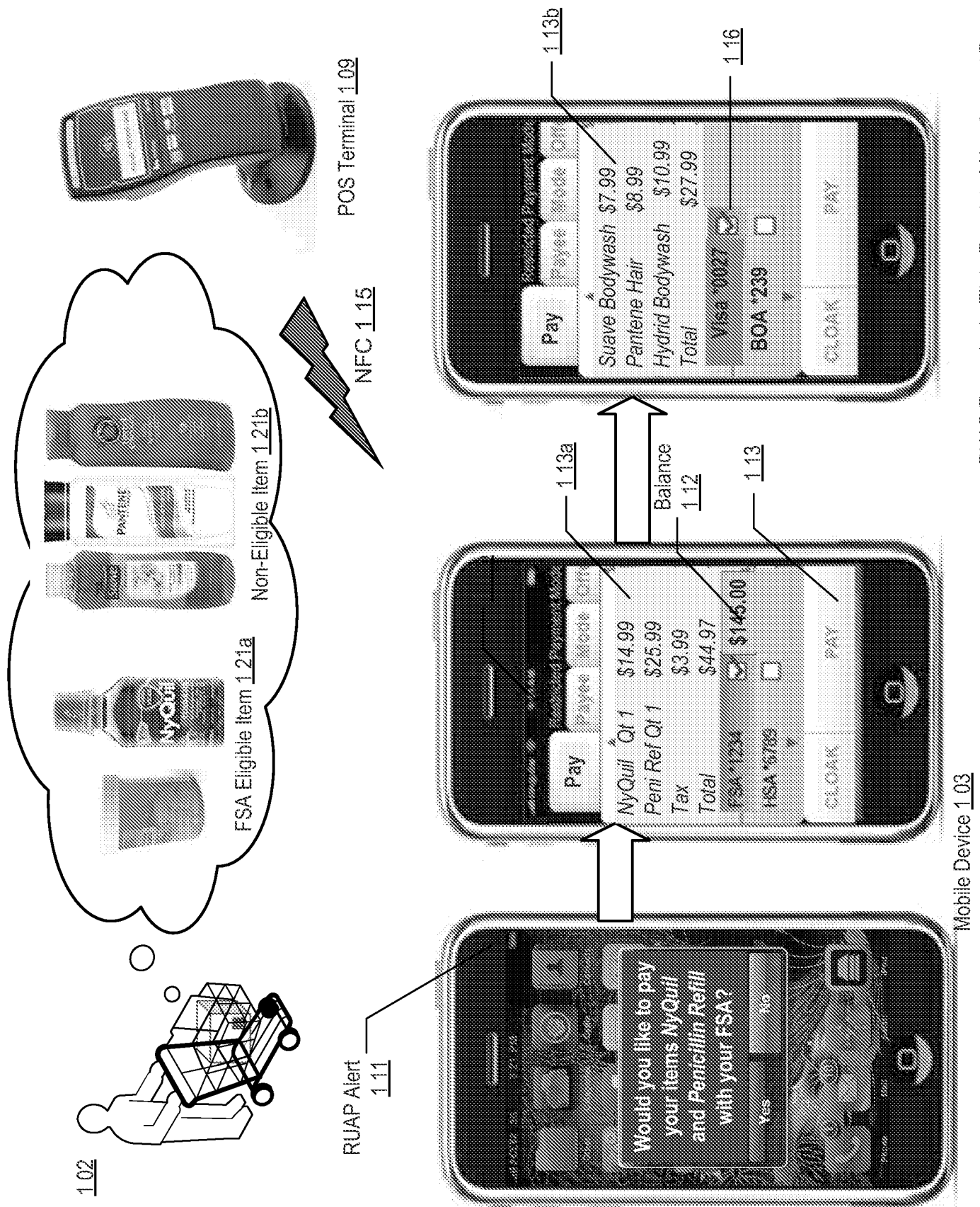


Figure 1A

Example: User Usage of Restricted-Use Accounts



1000

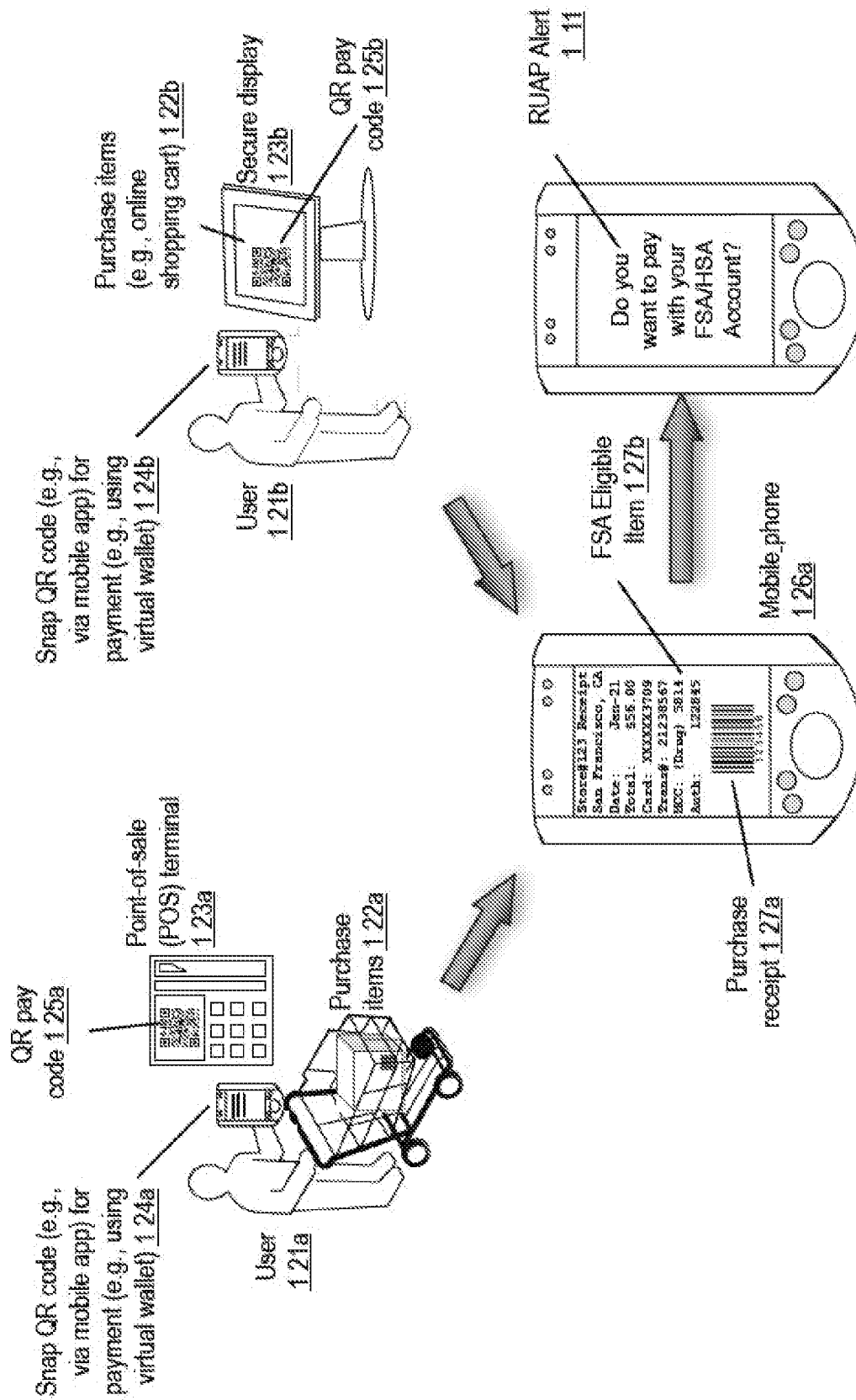


Figure 1C

RUAP Example: In-Flight Restricted-Use Snap Mobile Payment

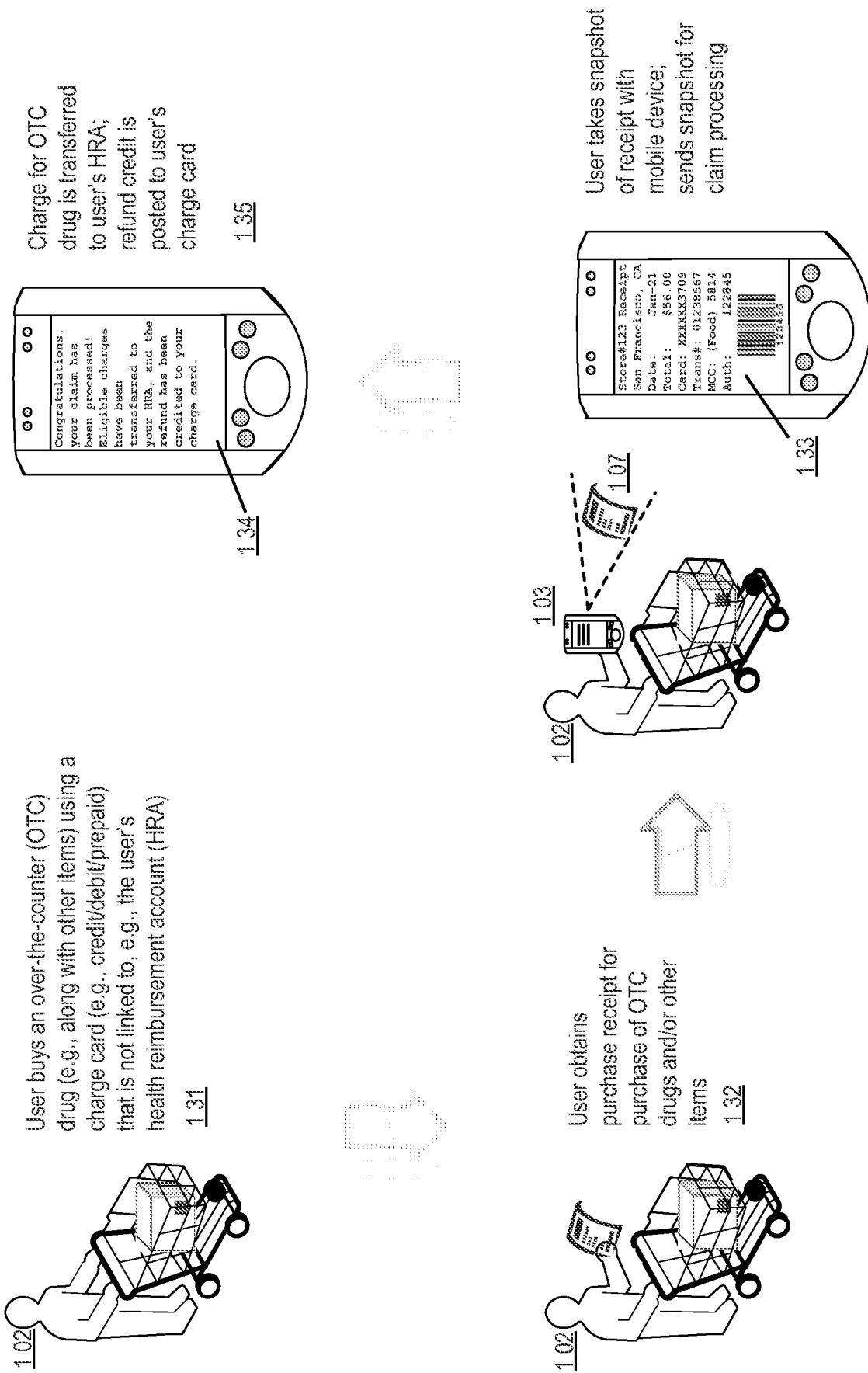
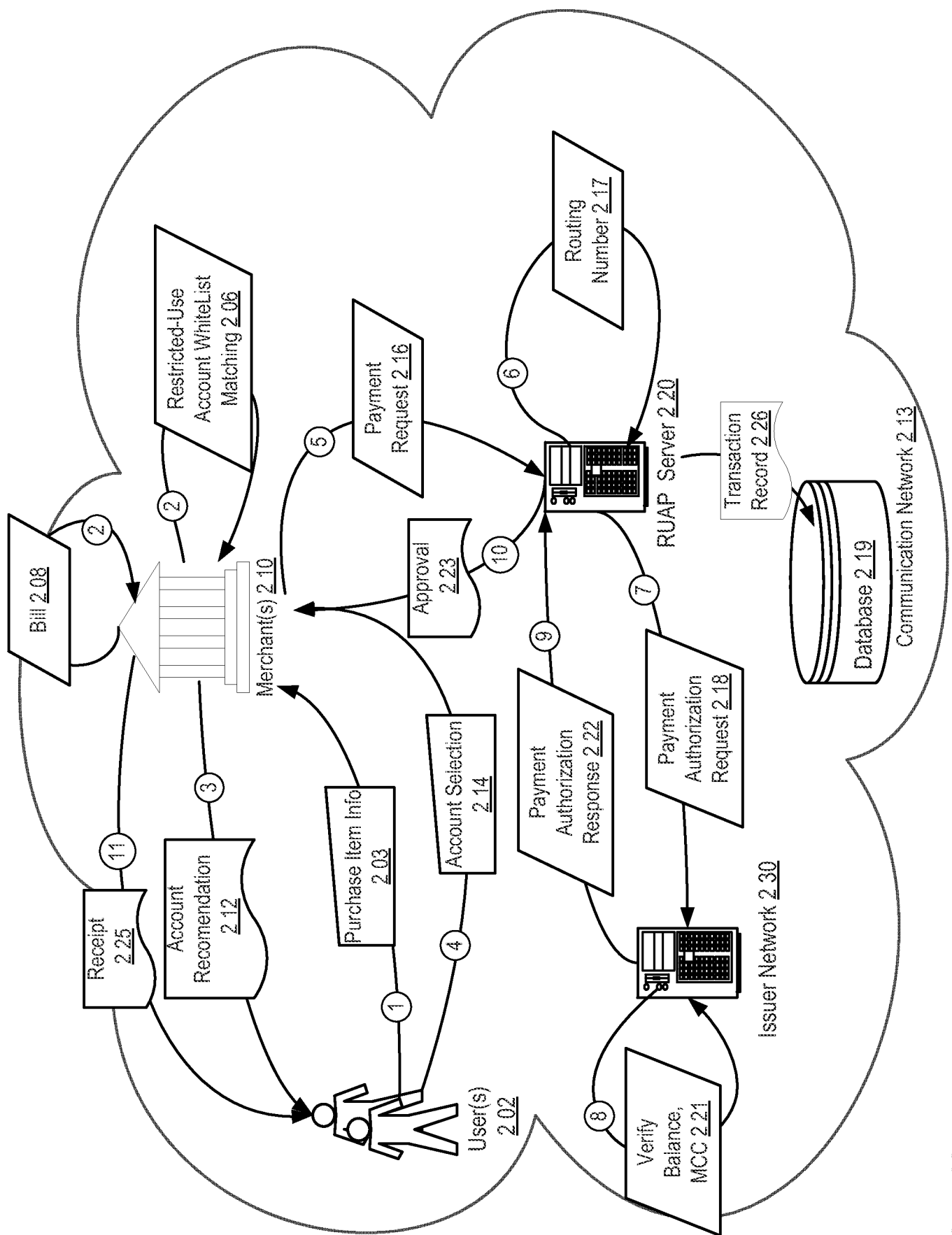


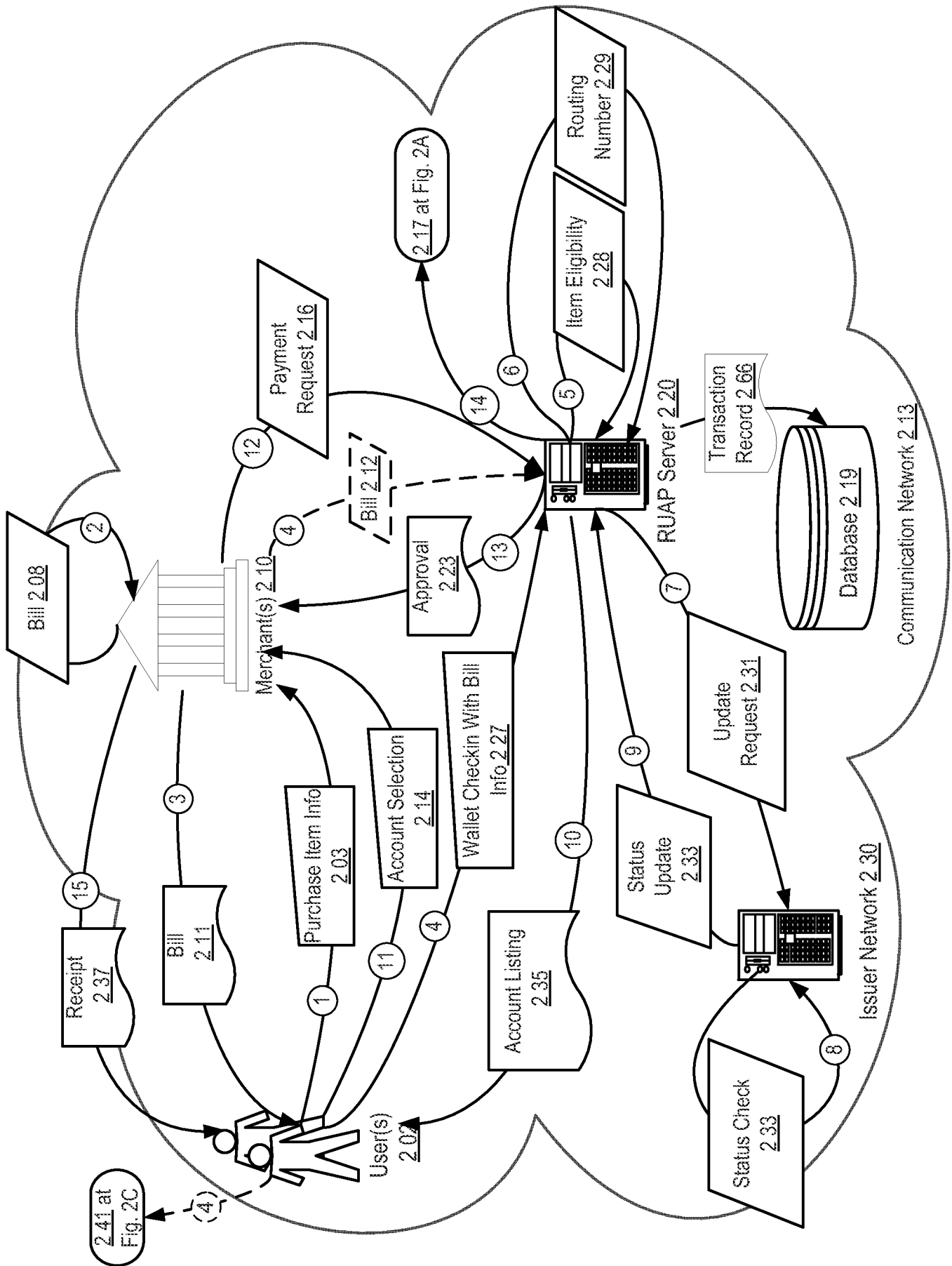
Figure 1D

RUAP Example: Post-Flight Snapshot Claim Scenario



RUAP Example Data Flow: In-Flight Intelligent PoS Checkout

Figure 2A



RUAP Example Data Flow: In-Flight Snap Bill Checkout

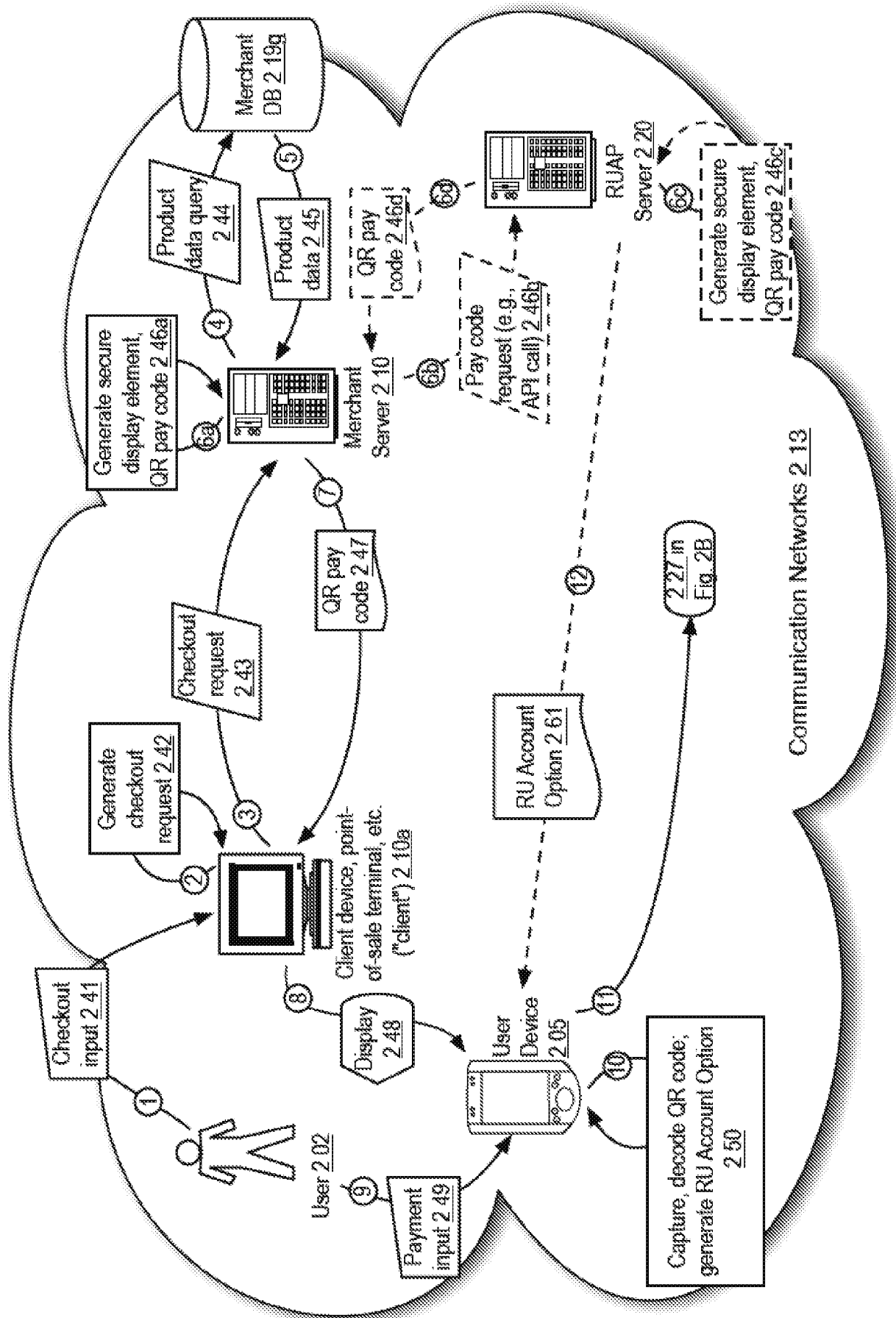


Figure 2C

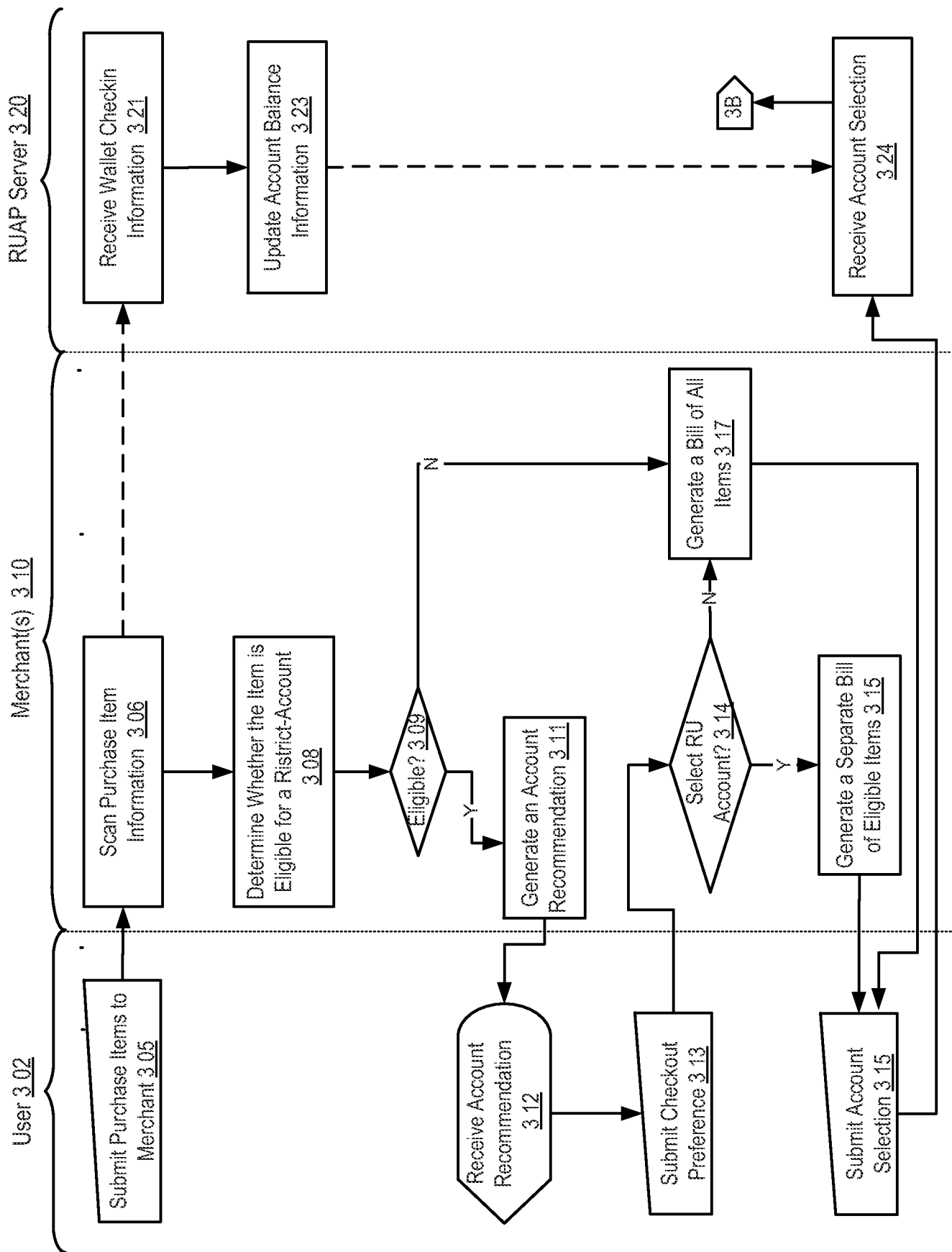
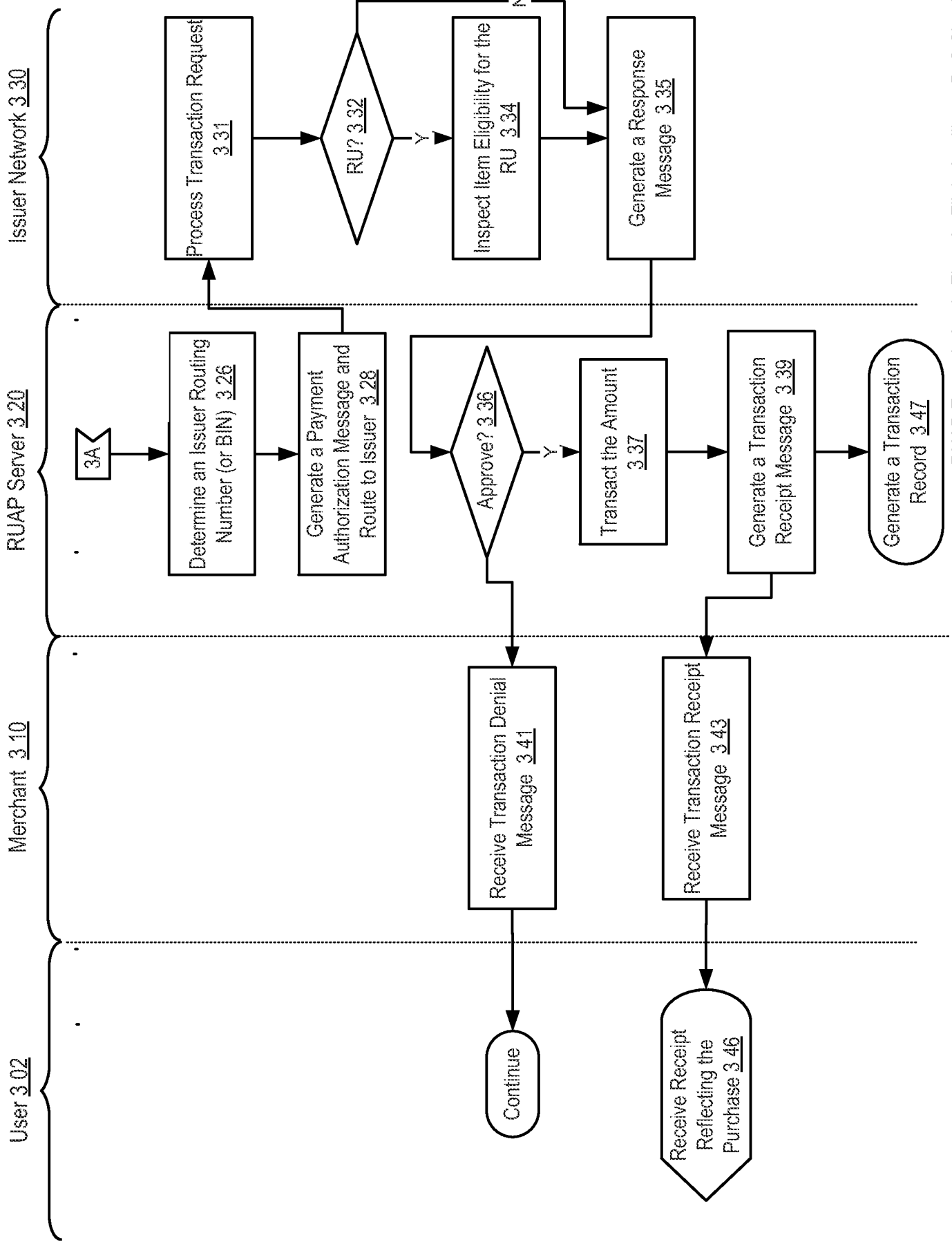


Figure 3A

RUAP Example Logic Flow: In-Flight Intelligent PoS Checkout



RUAP Example Logic Flow: In-Flight Intelligent PoS Checkout

Figure 3B

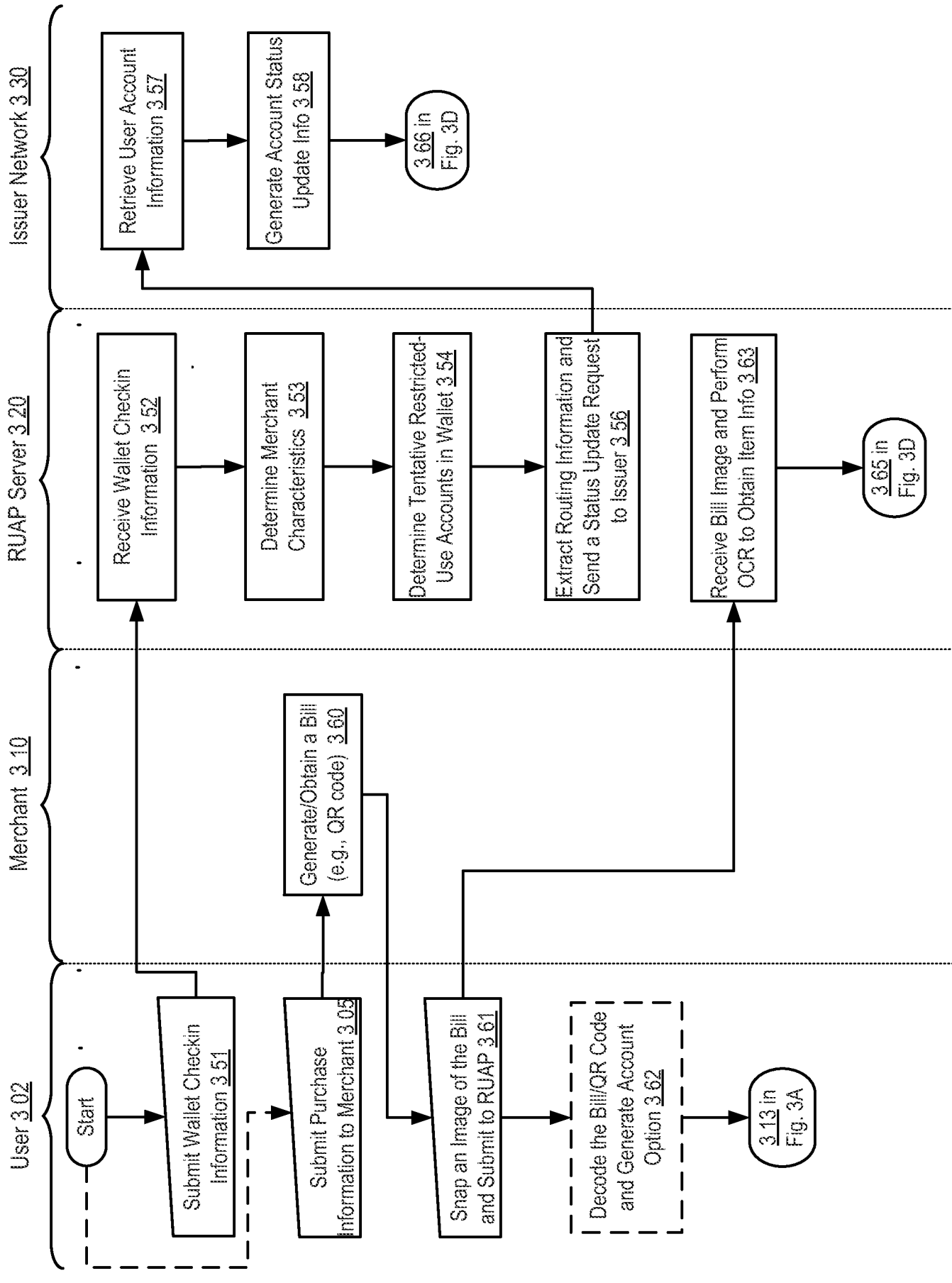


Figure 3C

RUAP Example Logic Flow: User Wallet Check-in at Merchant

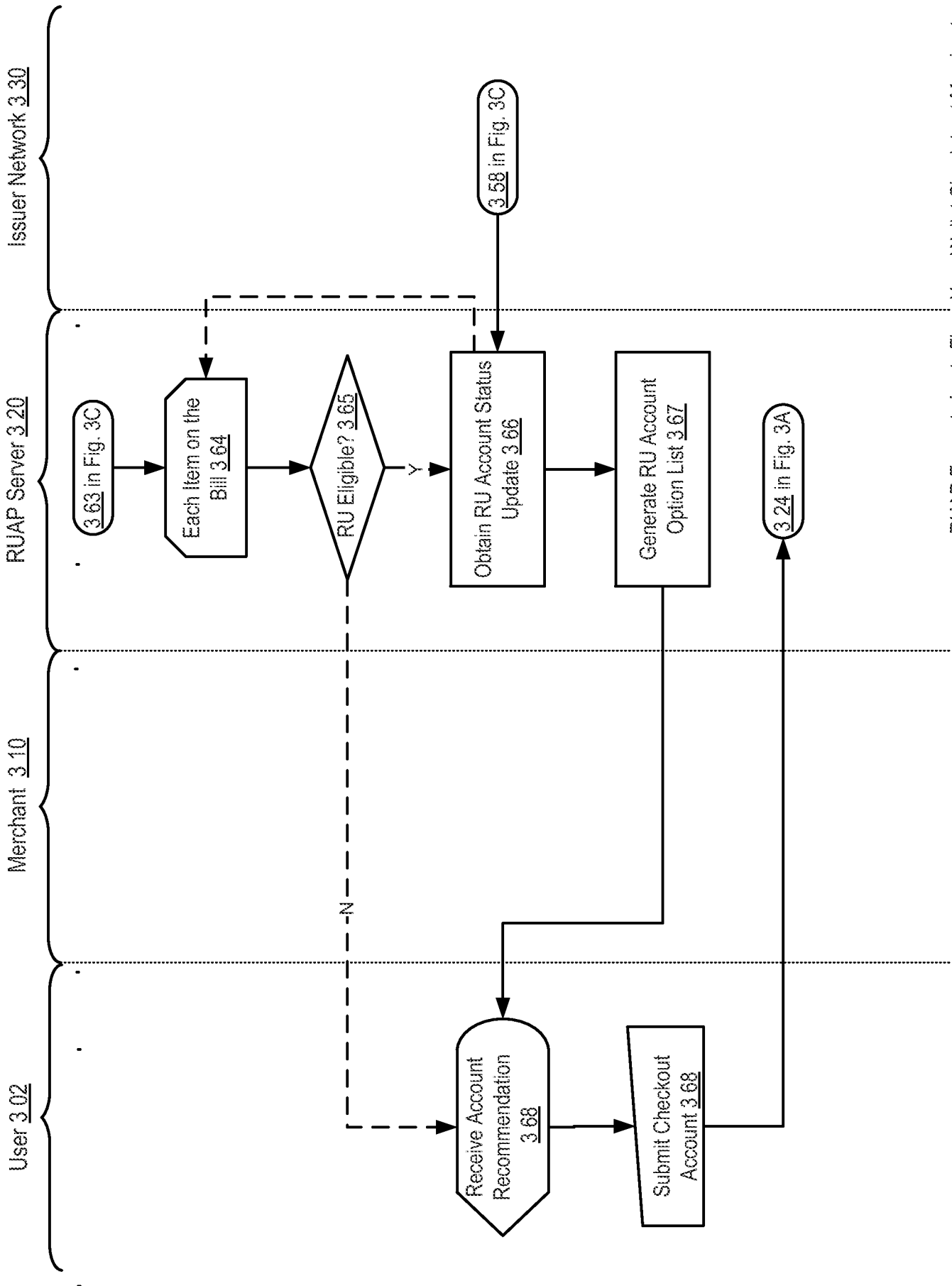


Figure 3D

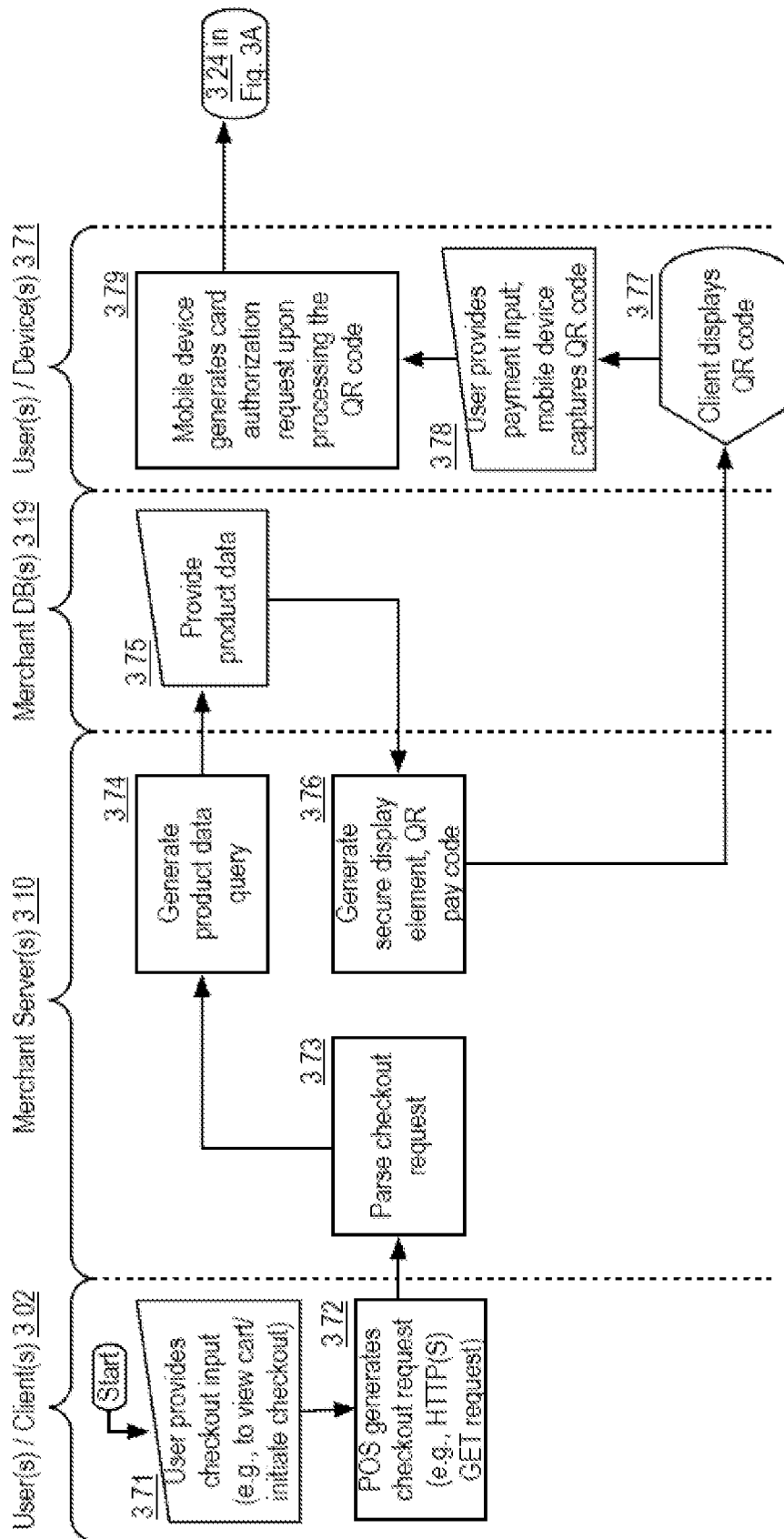


Figure 3E

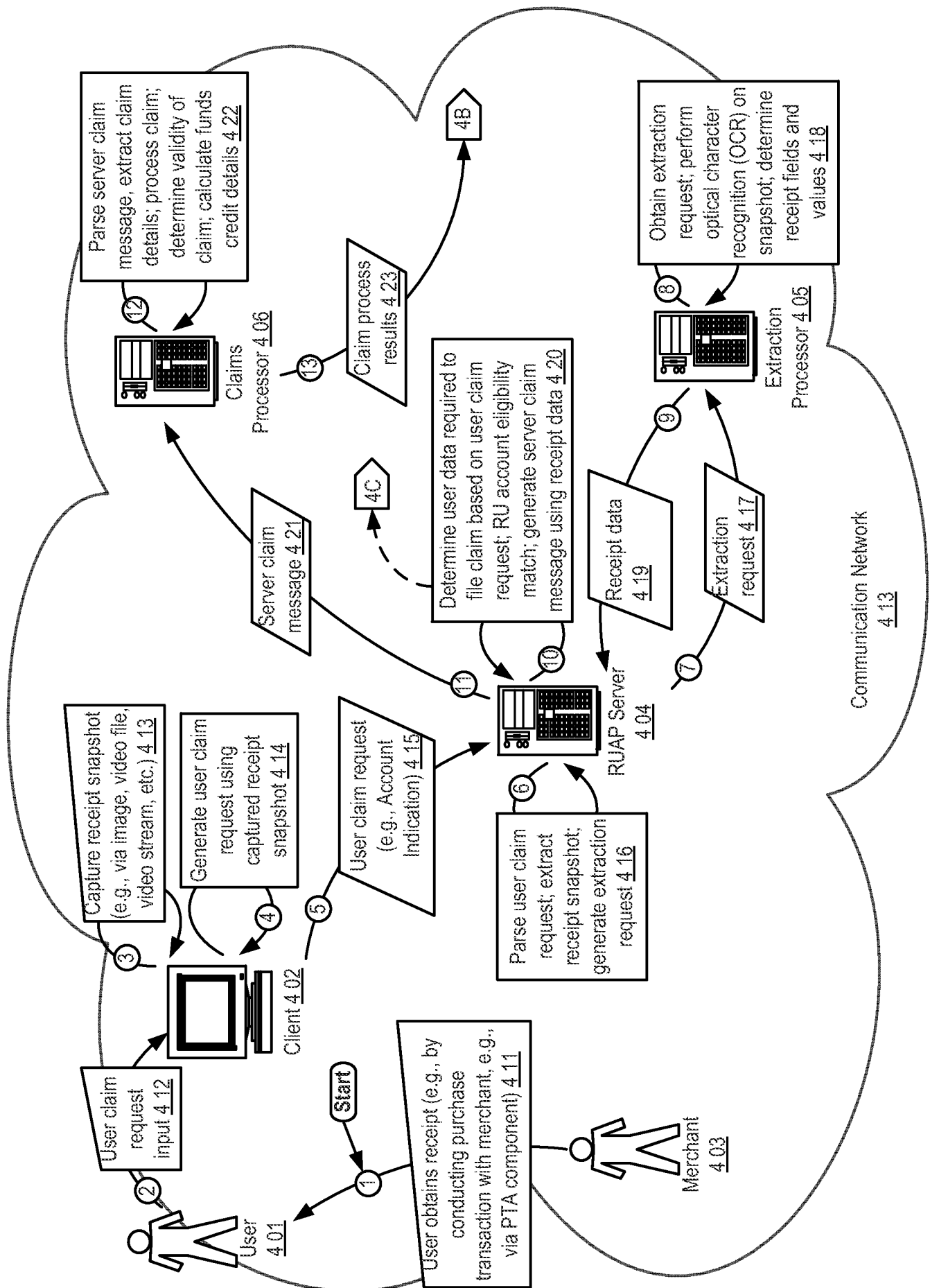


Figure 4A

RUAP Data Flow: Post-Flight Snap-Receipt Claim Processing

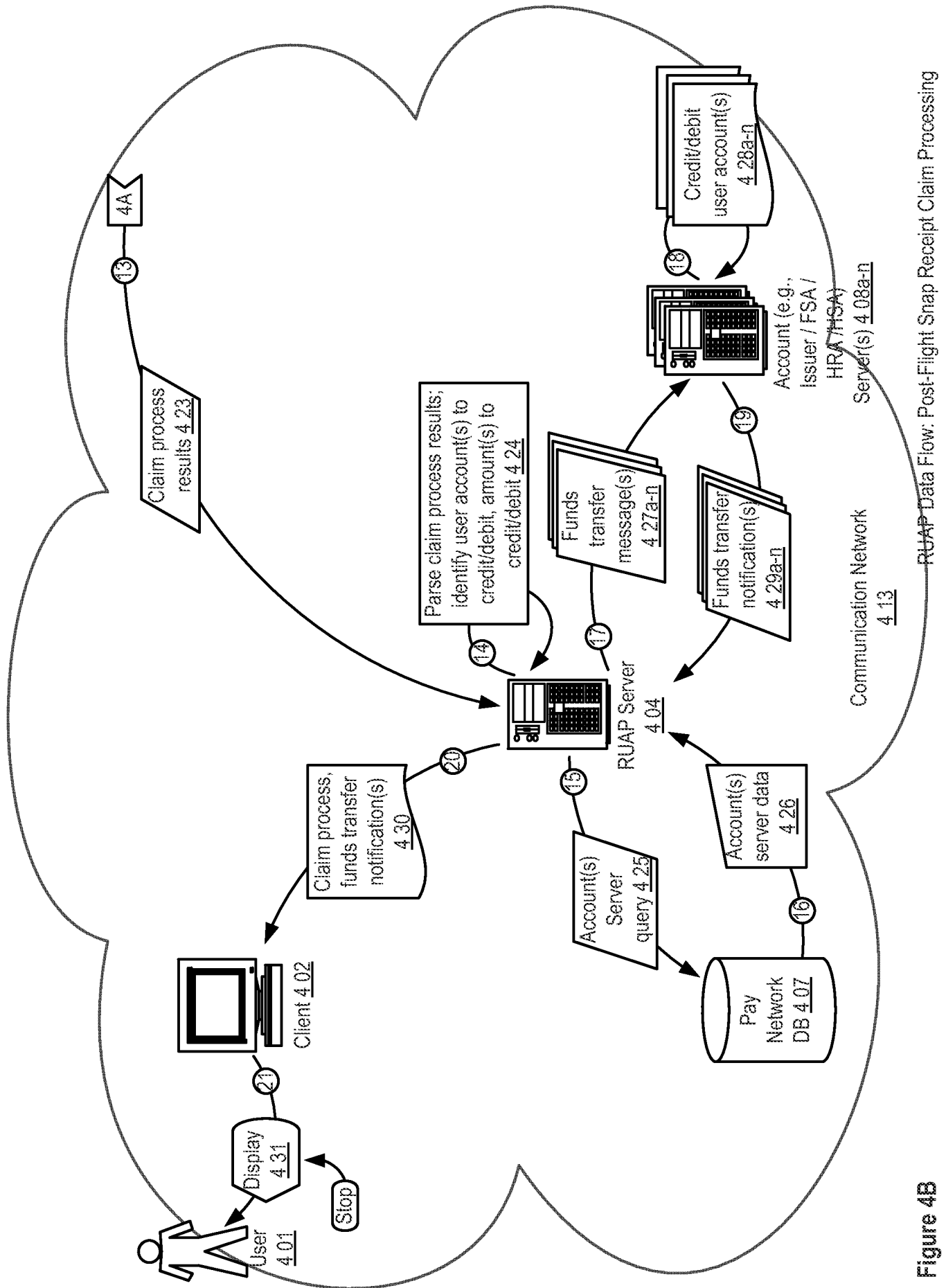
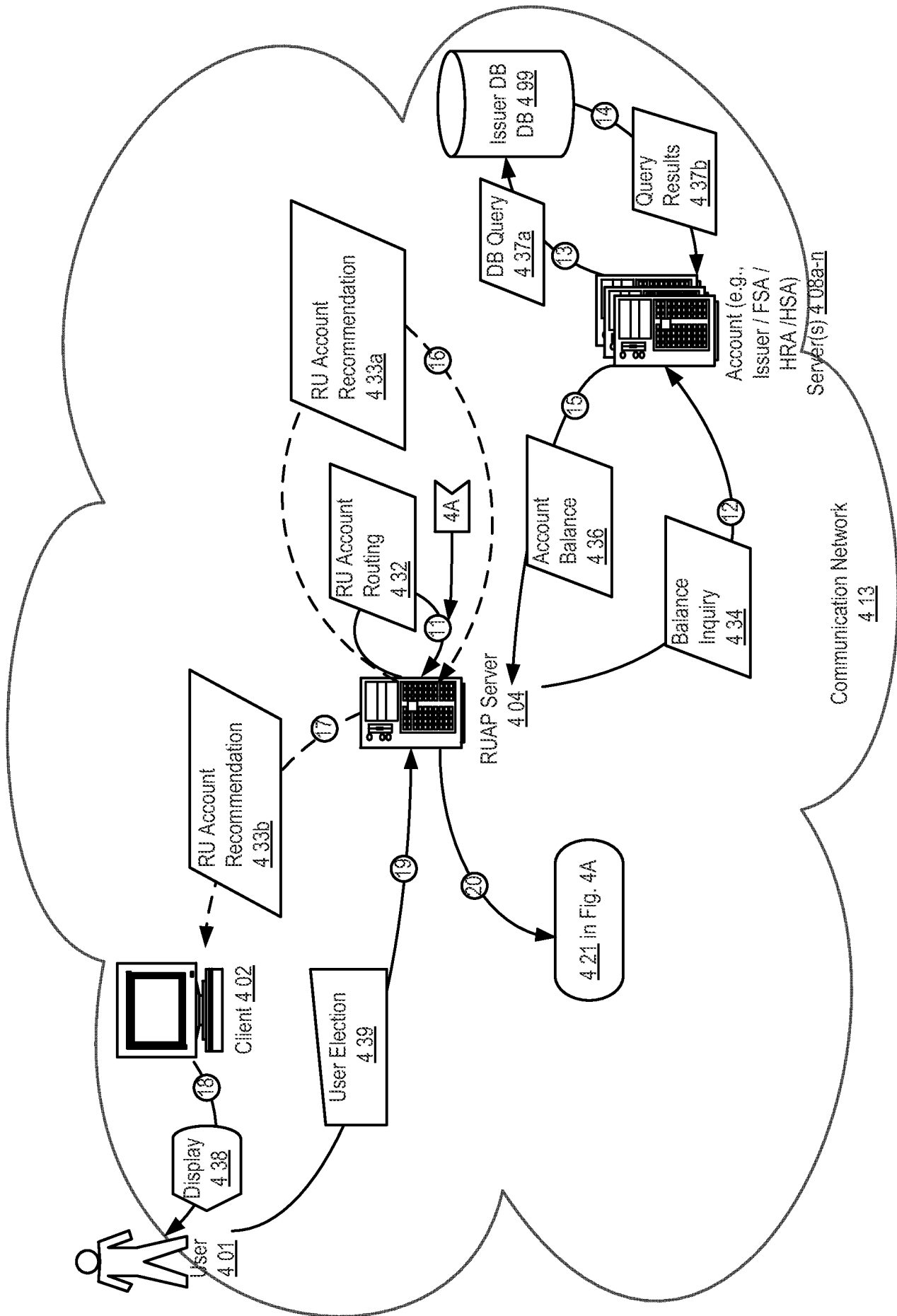


Figure 4B



RUAP Data Flow: Post-Flight Reimbursement Generation

Figure 4C

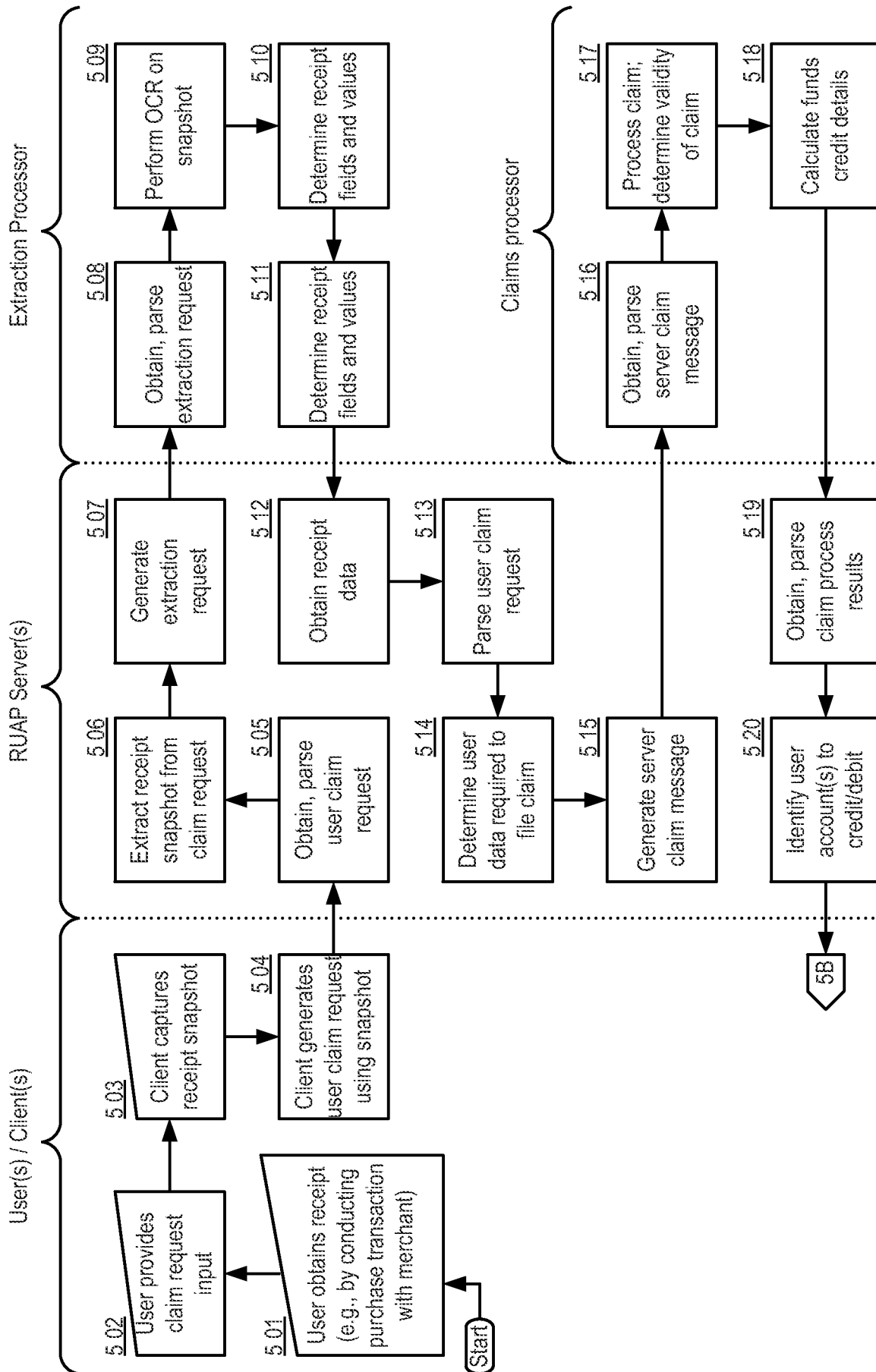
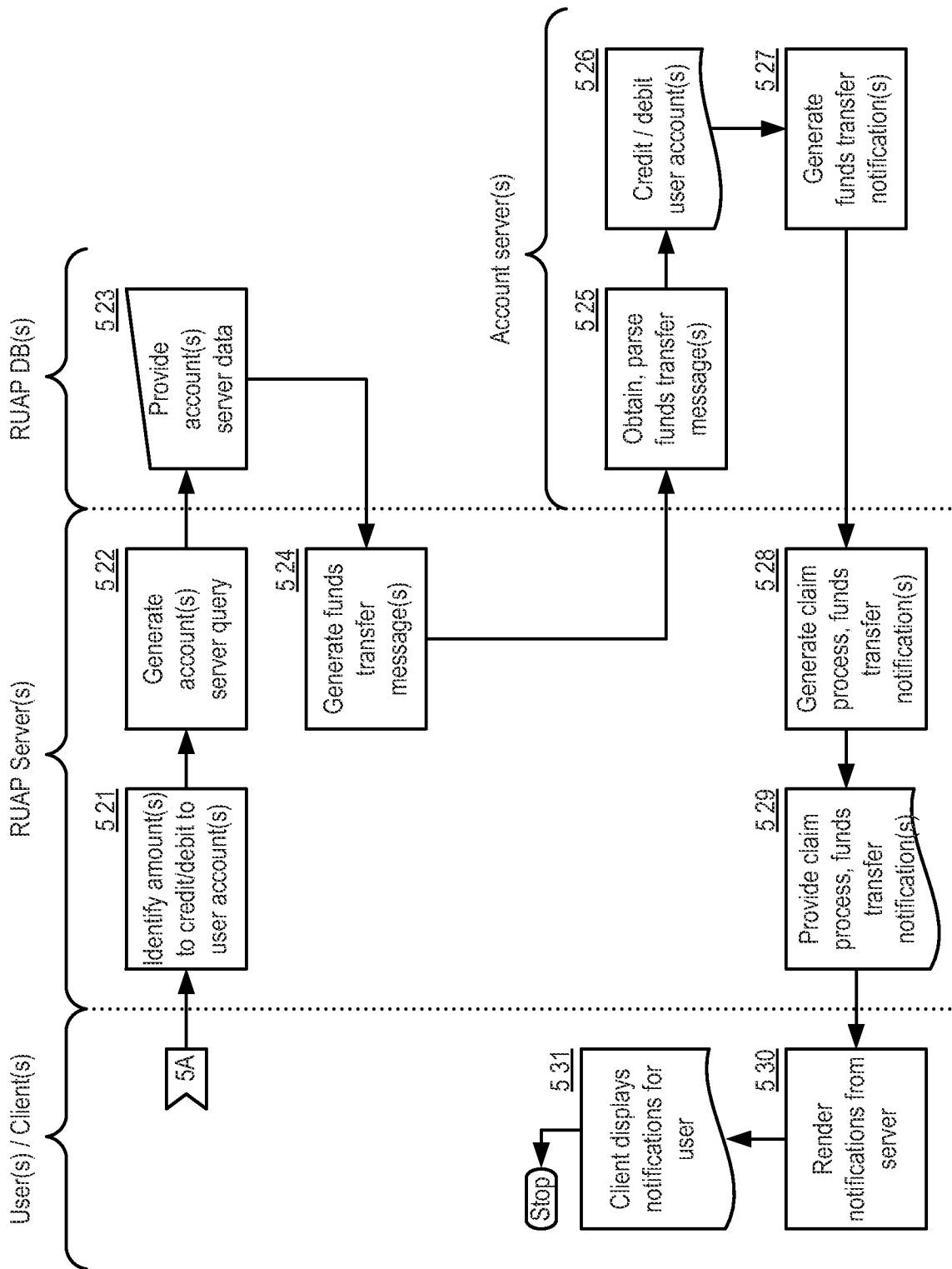


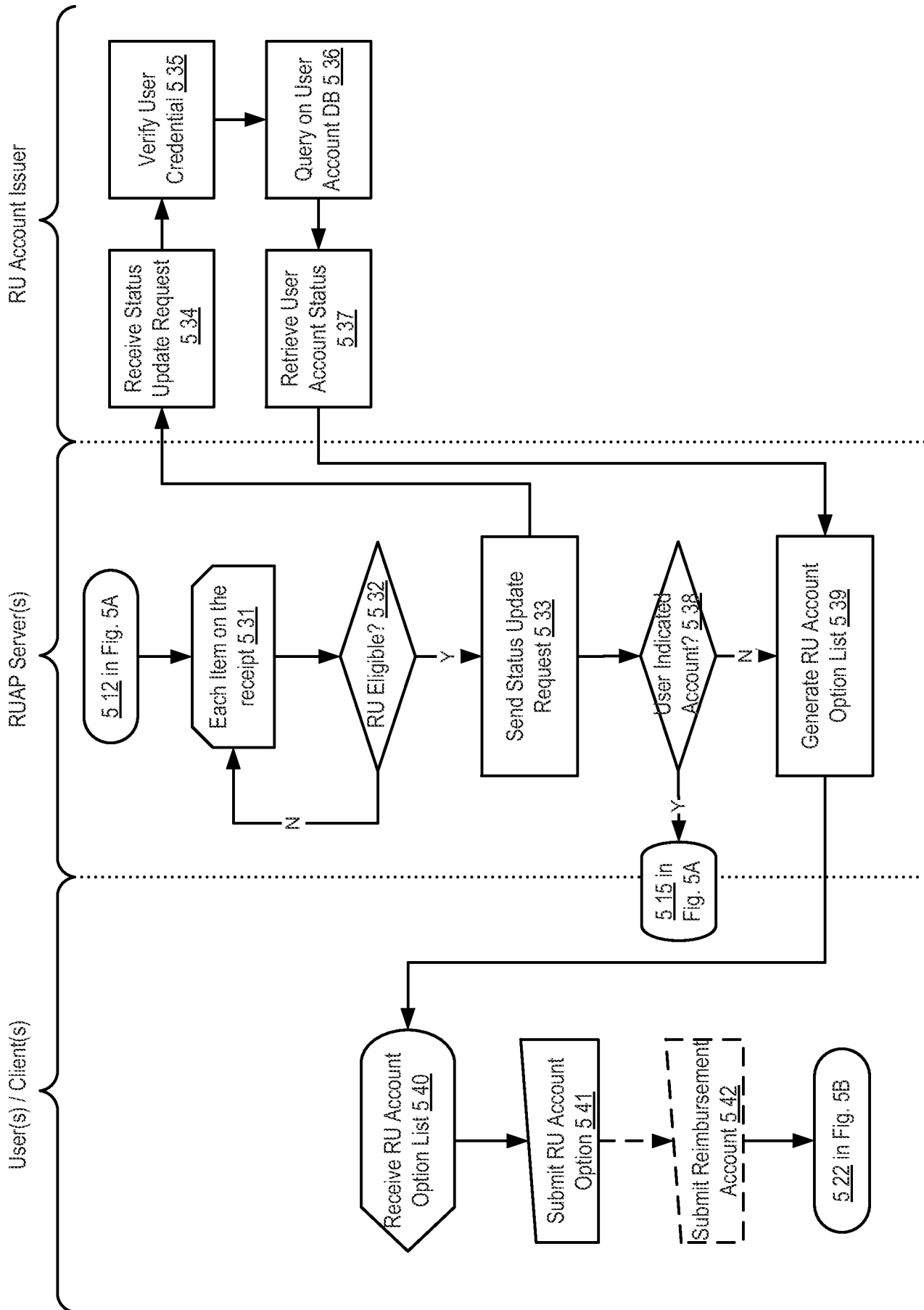
Figure 5A

Example: User Claim Processing (UCP) component 500



Example: User Claim Processing (UCP) component 500

Figure 5B



Example: User Claim Generation

Figure 5C

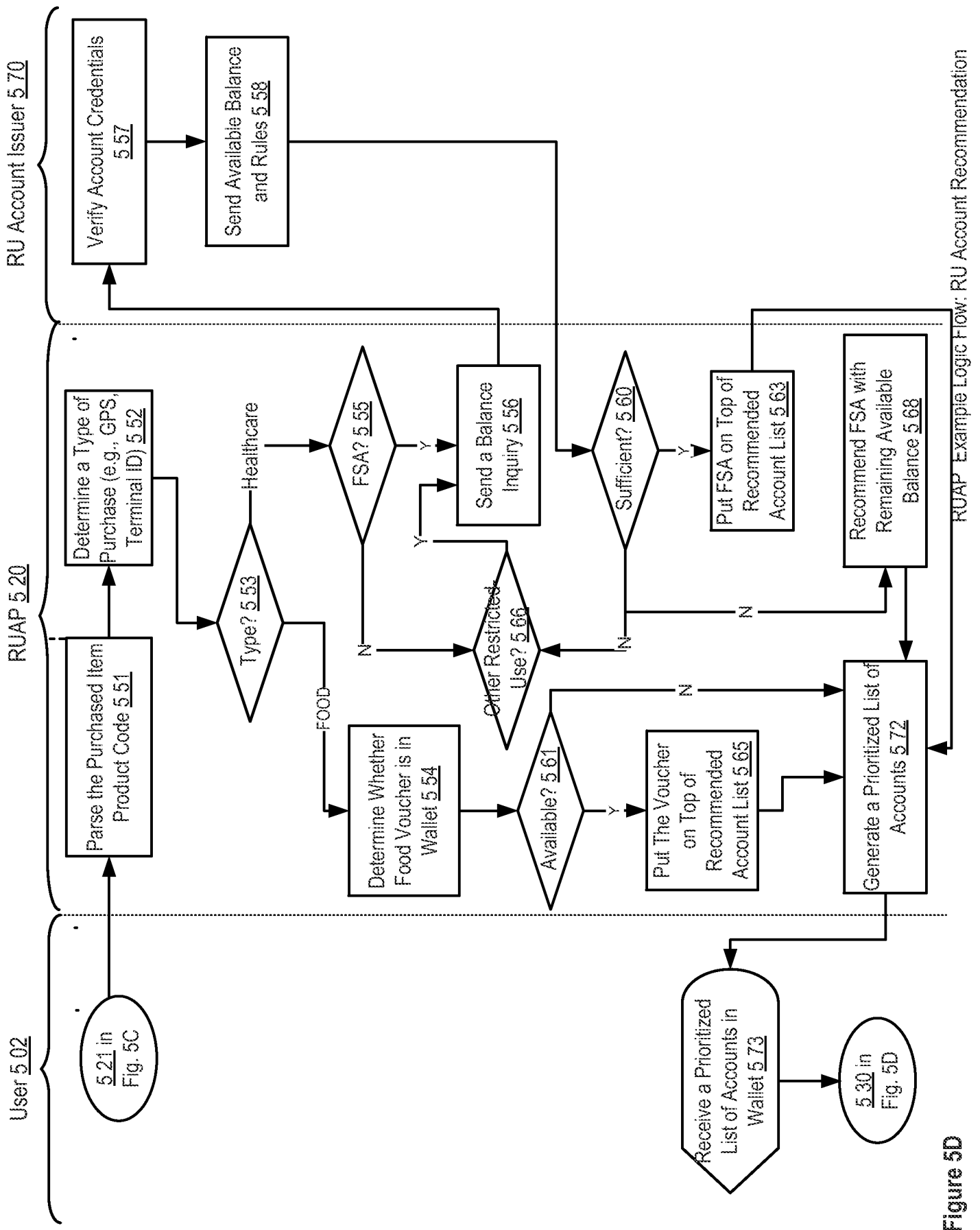
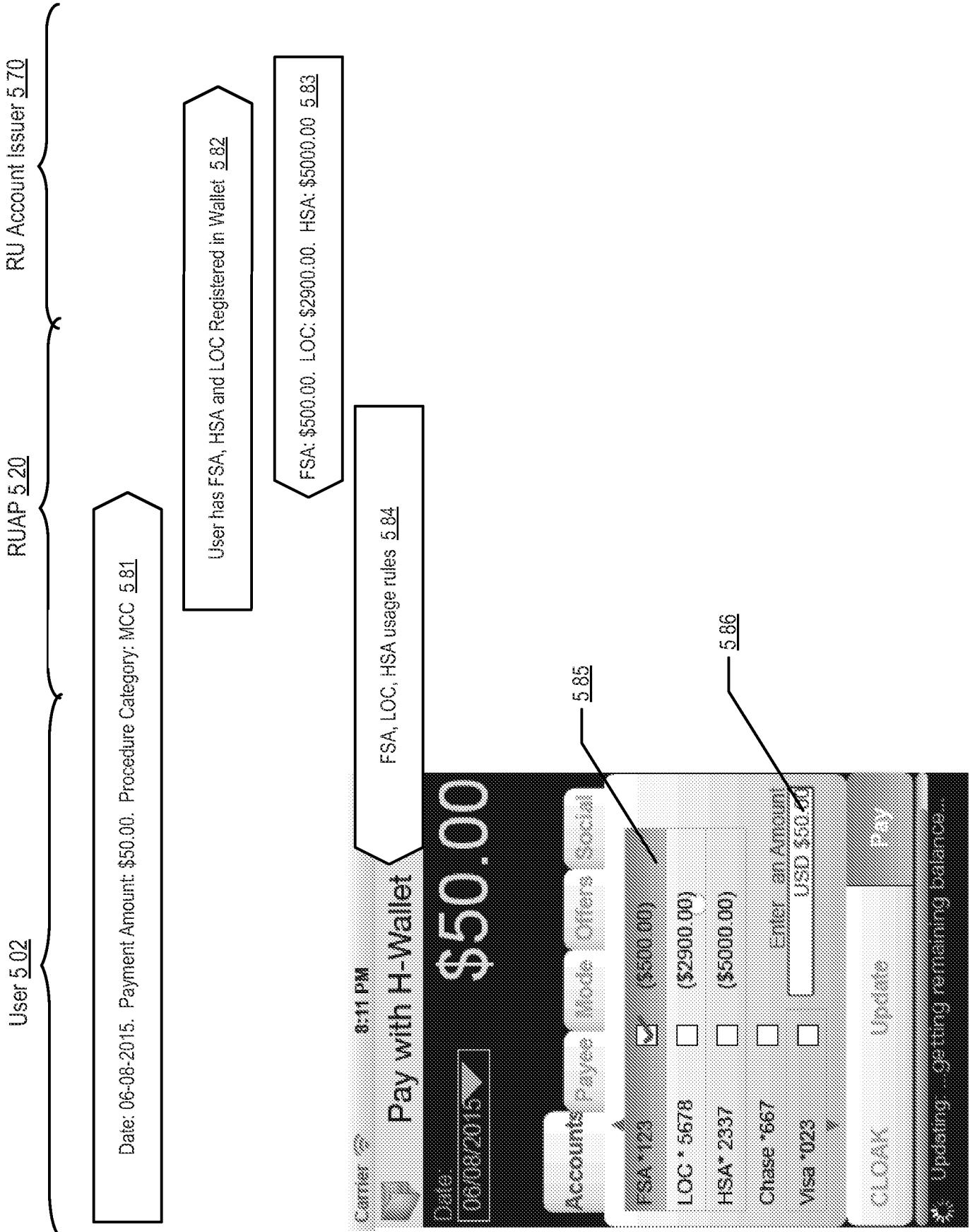


Figure 5D



RUAP Example Dollar Flow: RU Account Recommendation

Figure 5E

User 502

RUAP 520

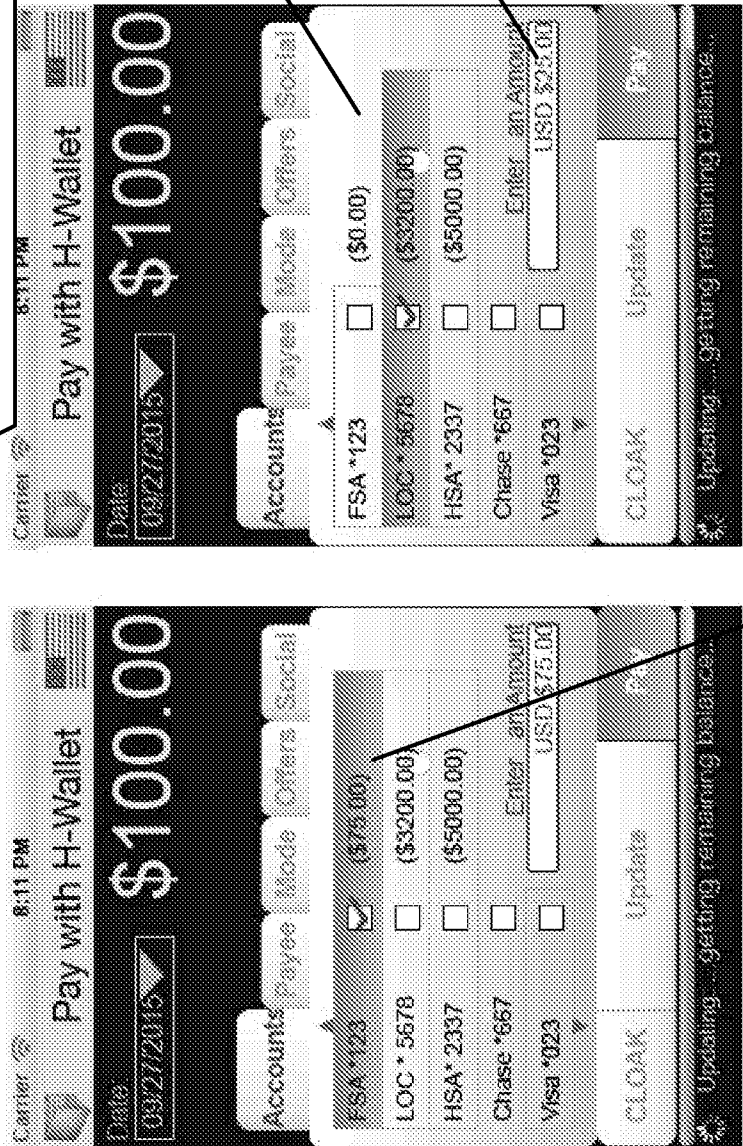
Healthcare Sponsor 570

Date: 09-27-2015. Payment Amount: \$100.00. Procedure Category: MCC 587

User has FSA, HSA and LOC Registered in Wallet 588

FSA: \$75.00. LOC: \$3200.00. HSA: \$5000.00 589

FSA, LOC, HSA usage rules 590



RUAP Example Dollar Flow: RU Account Recommendation

Figure 5F

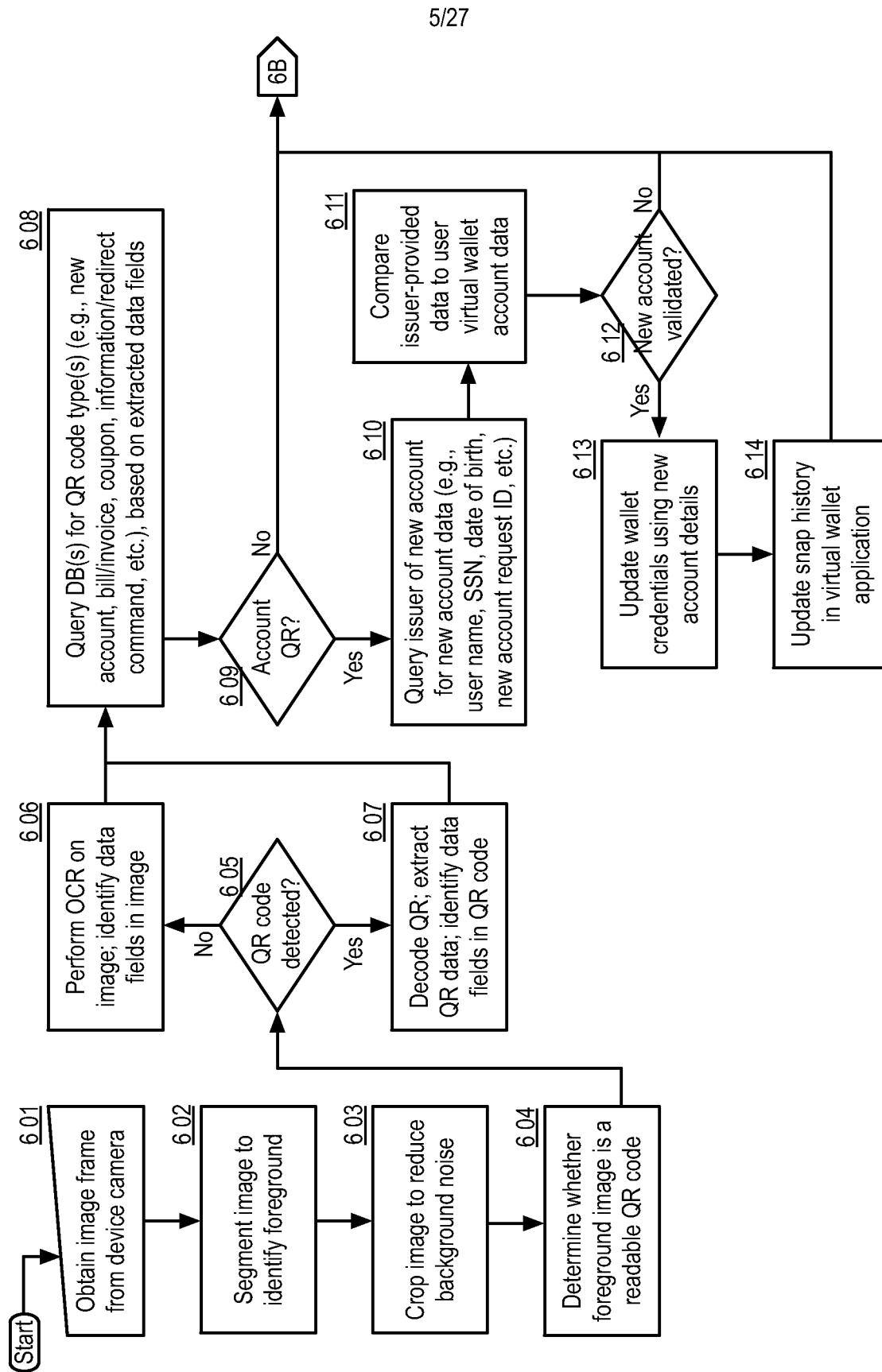


Figure 6A Example: Quick Response Code Processing ("QRCP") component 600

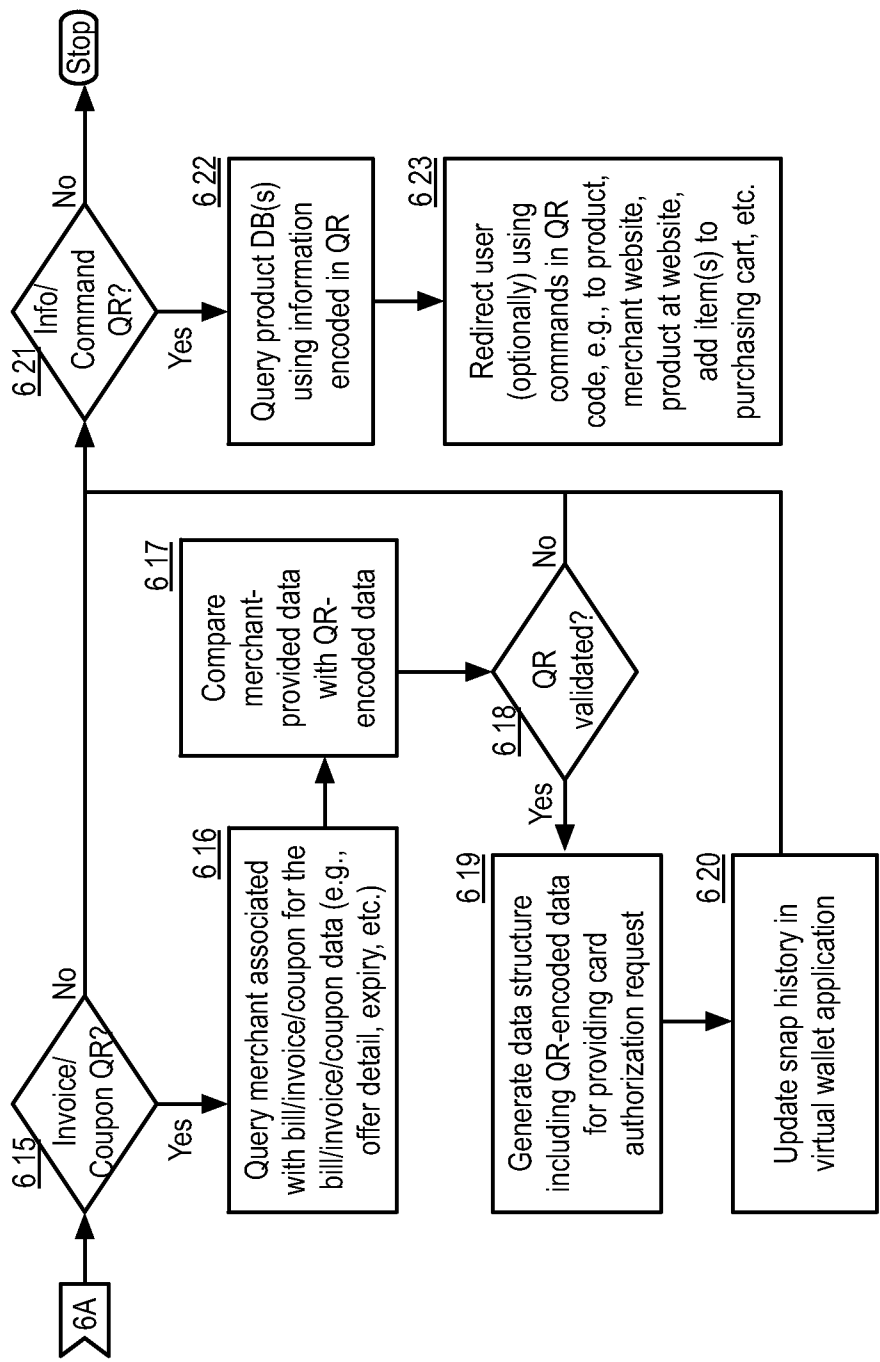
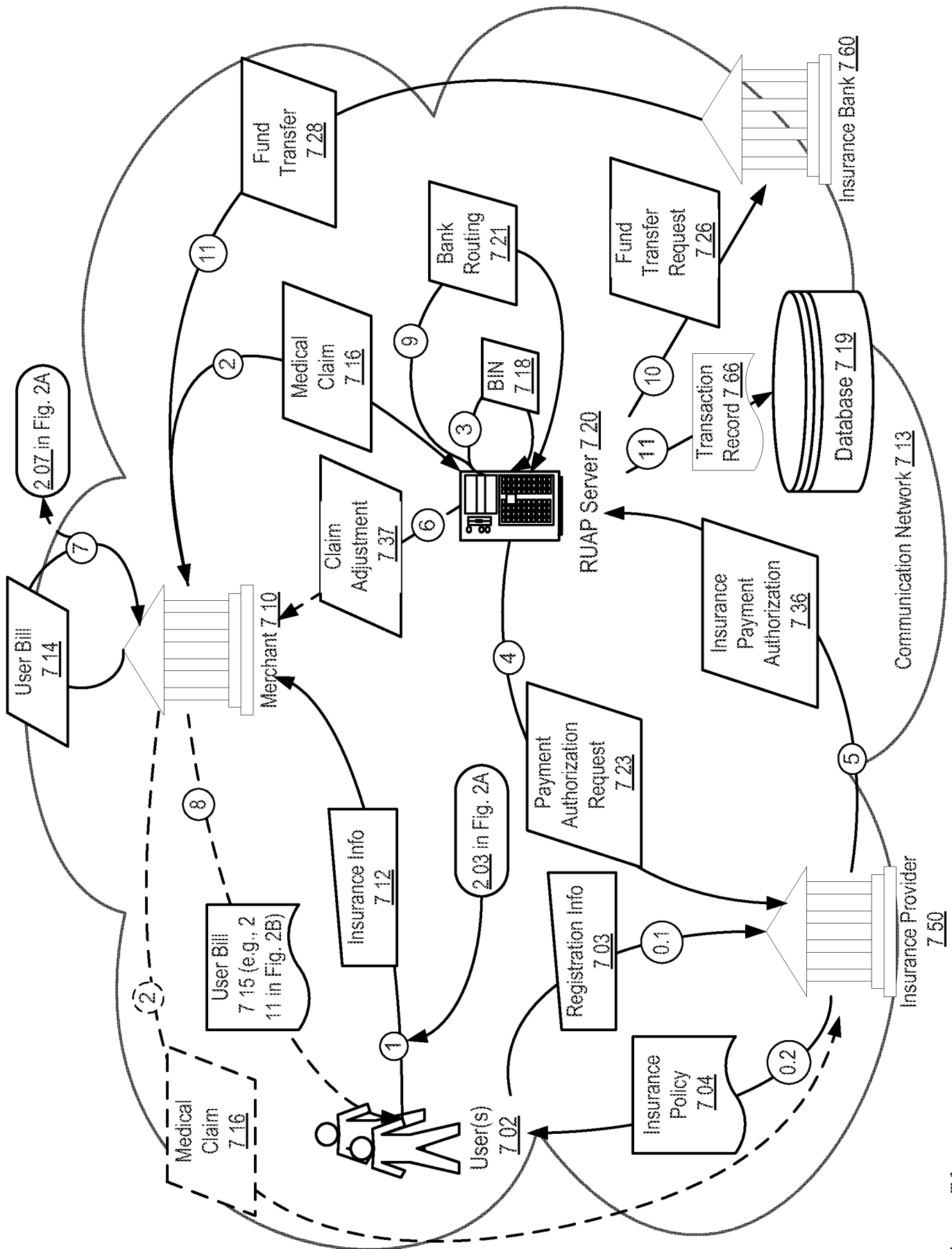


Figure 6B

Example: Quick Response Code Processing ("QRCP") component 600



RUAP Example Data Flow: Insurance Adjudication

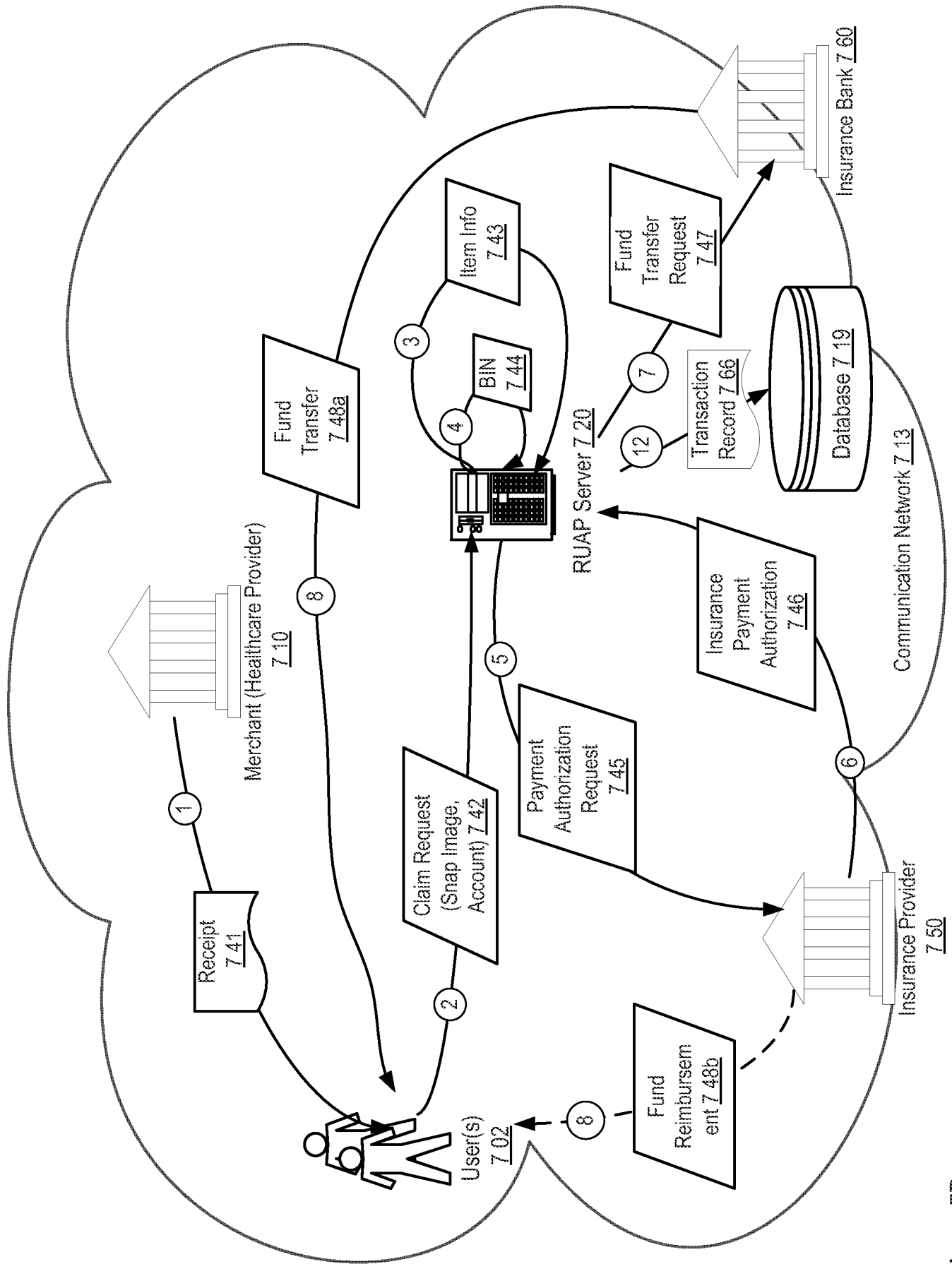


Figure 7B RUAP Example Data Flow: Insurance Reimbursement

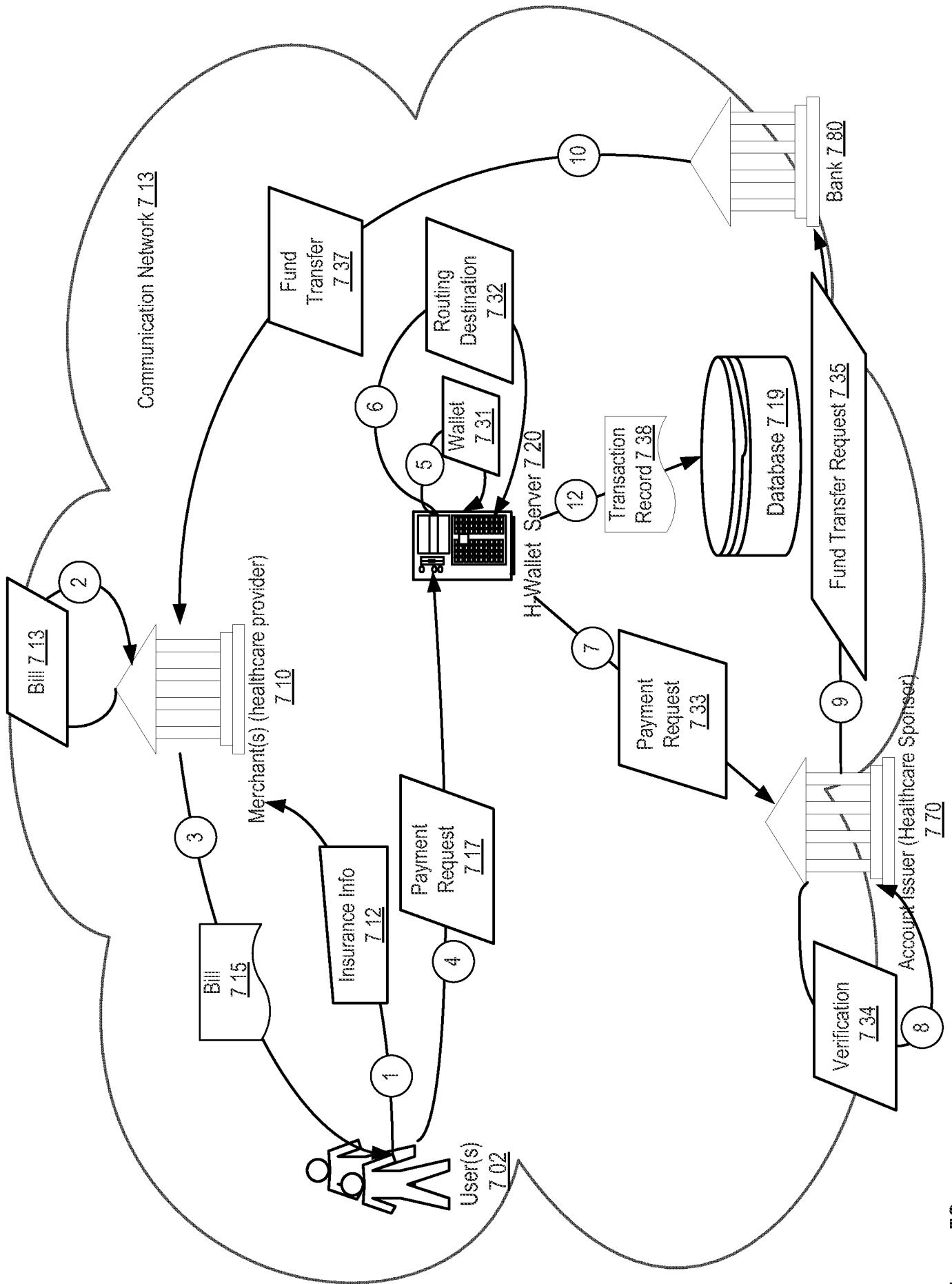


Figure 7C
RUAP Example Data Flow: User Payment

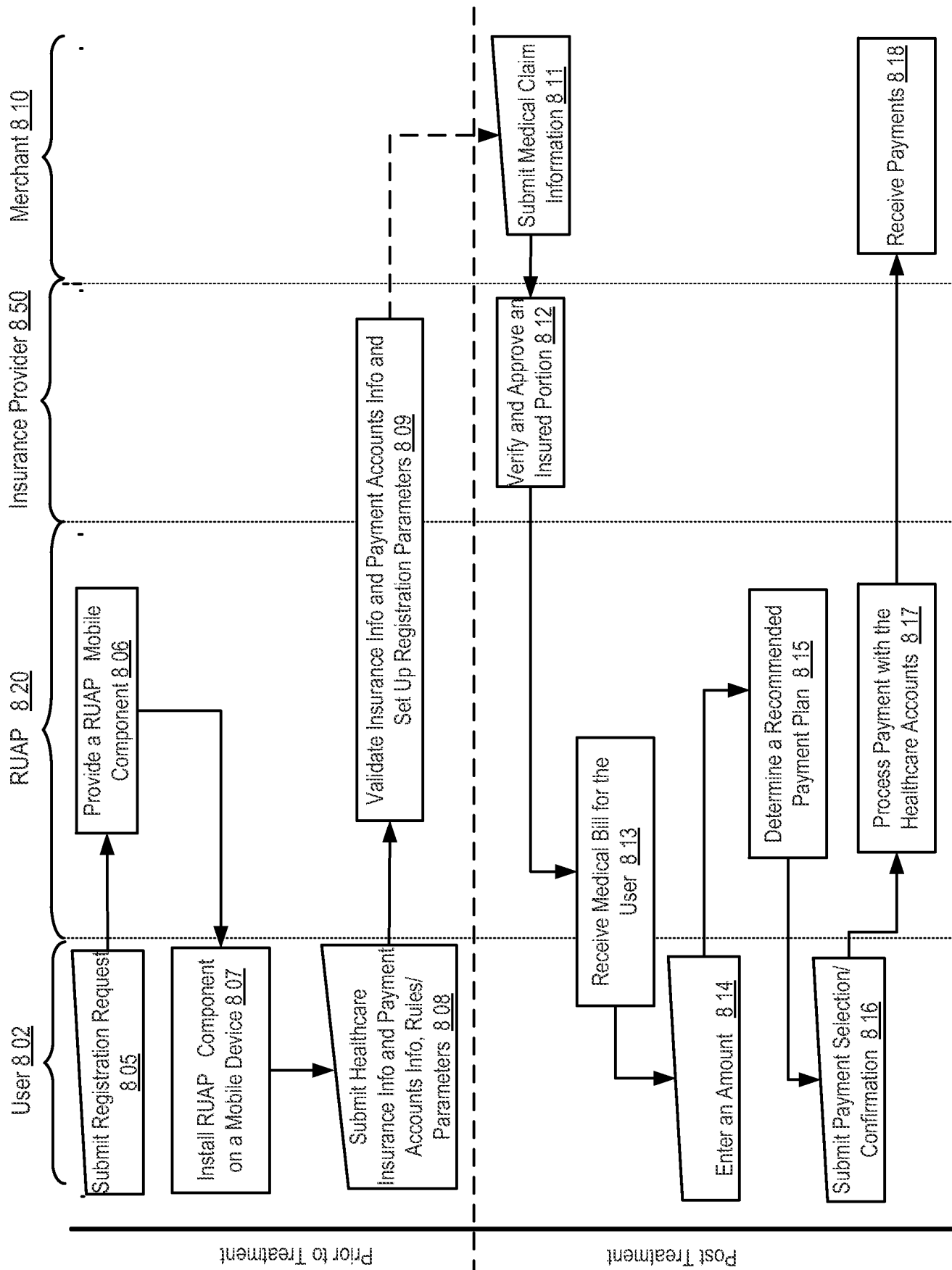


Figure 8A

RUAP Example Logic Flow: Healthcare Wallet Payment Overview

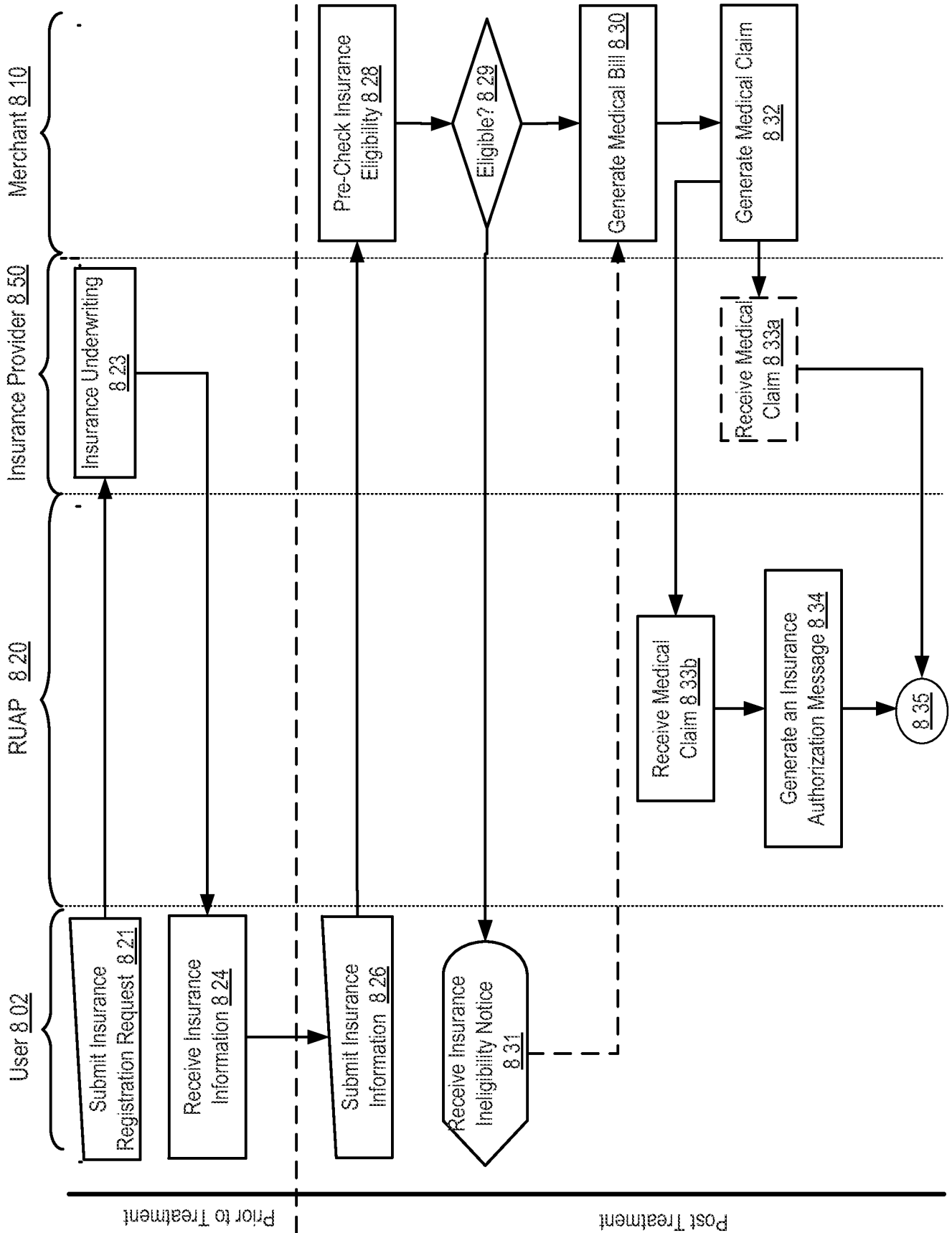
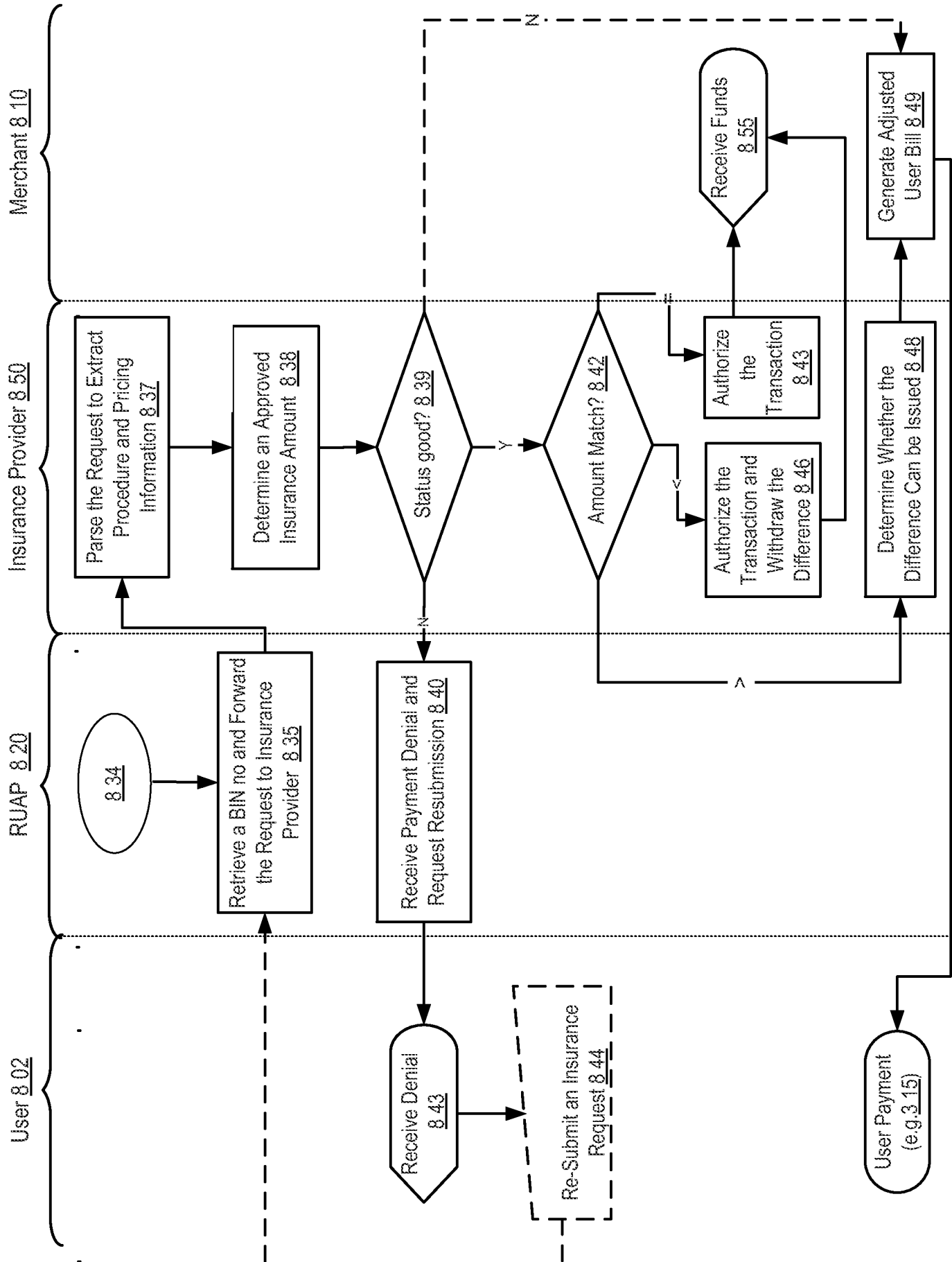


Figure 8B

RUAP Example Logic Flow: Healthcare Insurance Adjudication (1)



RUAP Example Logic Flow: Healthcare Insurance Adjudication (2)

Figure 8C

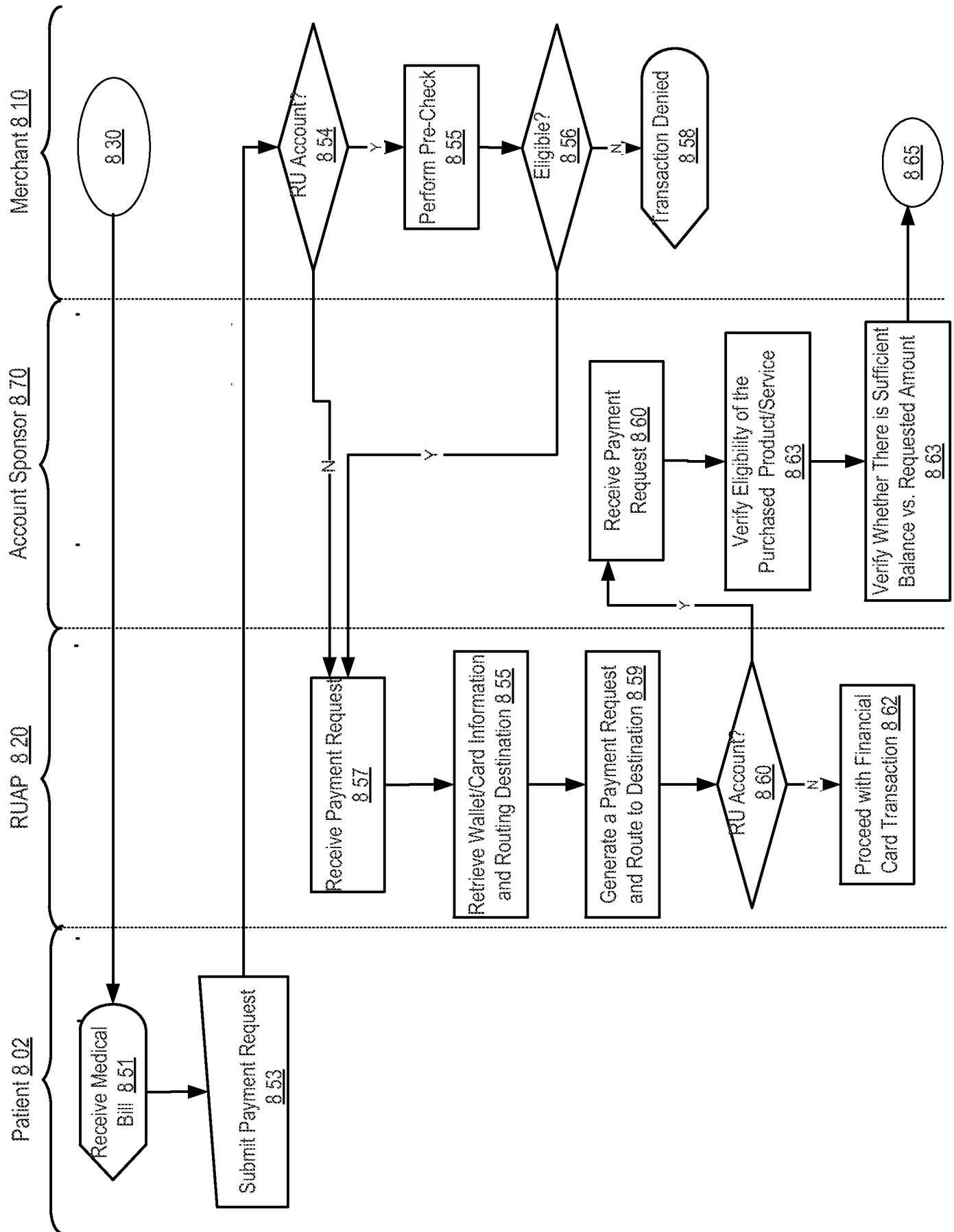
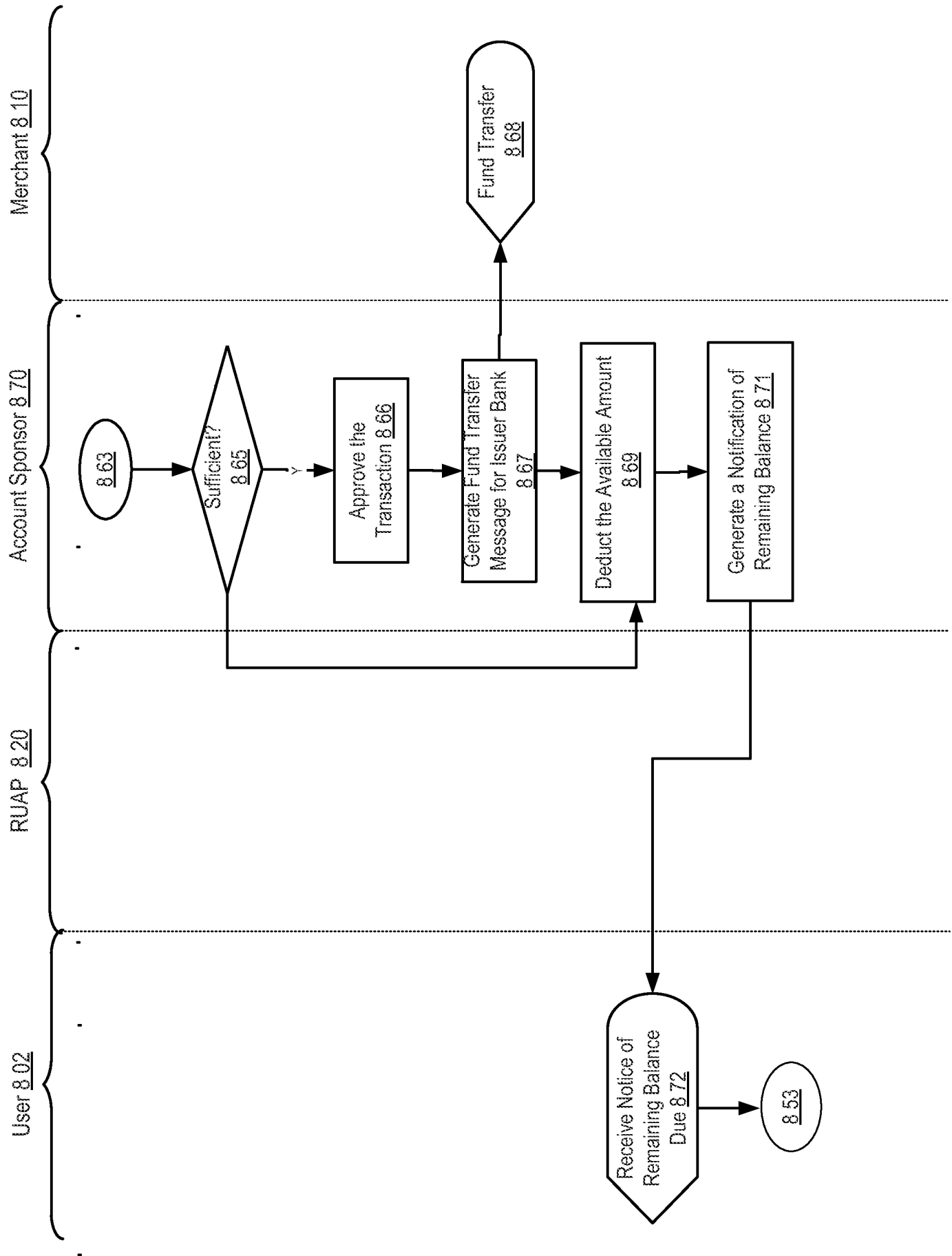


Figure 8D

RUAP Example Logic Flow: Healthcare User Co-Pay (1)

Figure 8E
RUAP Example Logic Flow: Healthcare User Co-Pay (2)

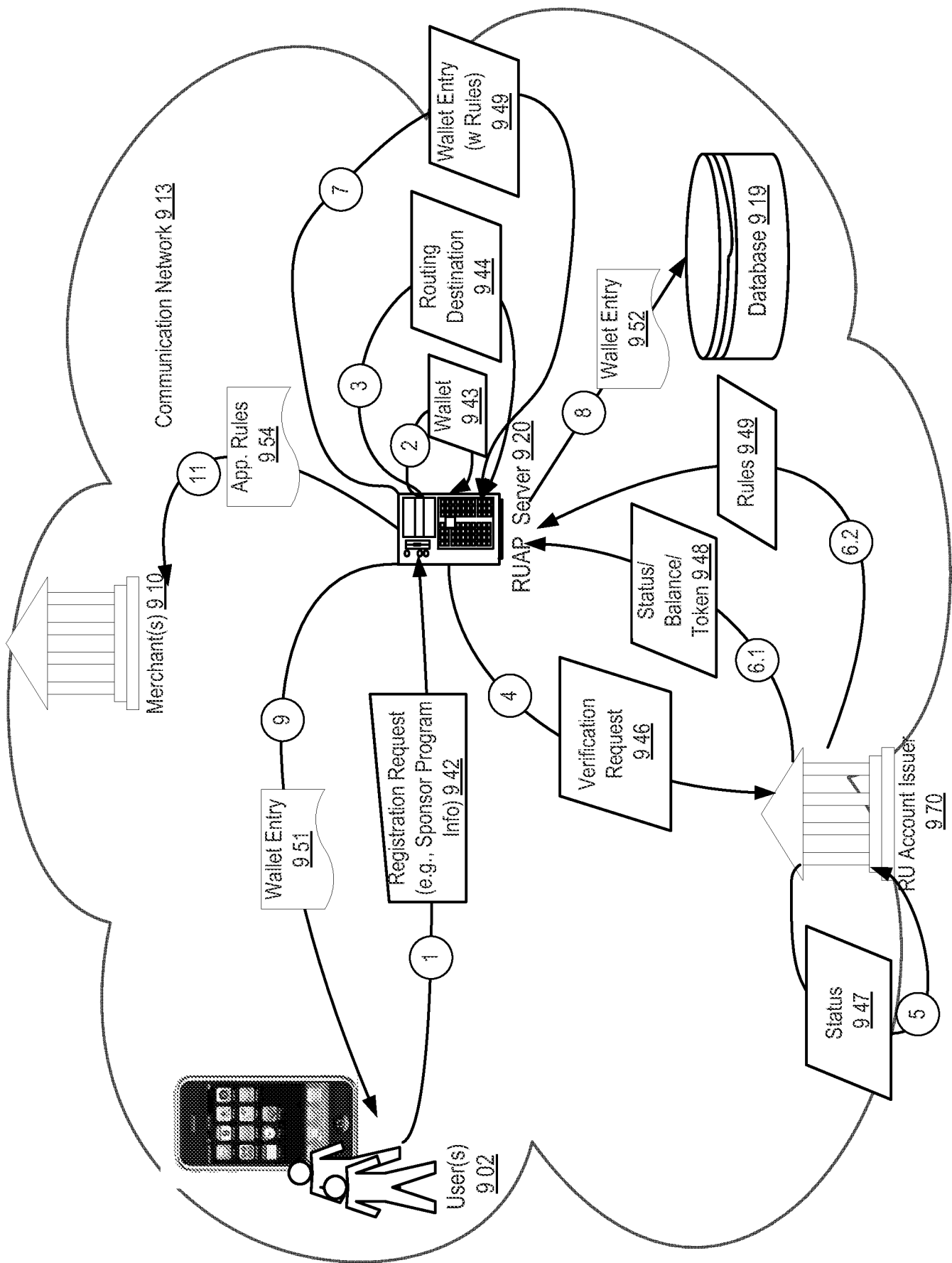


Figure 9A

RUAP Example Data Flow: Healthcare Restricted-Use Account Enrollment

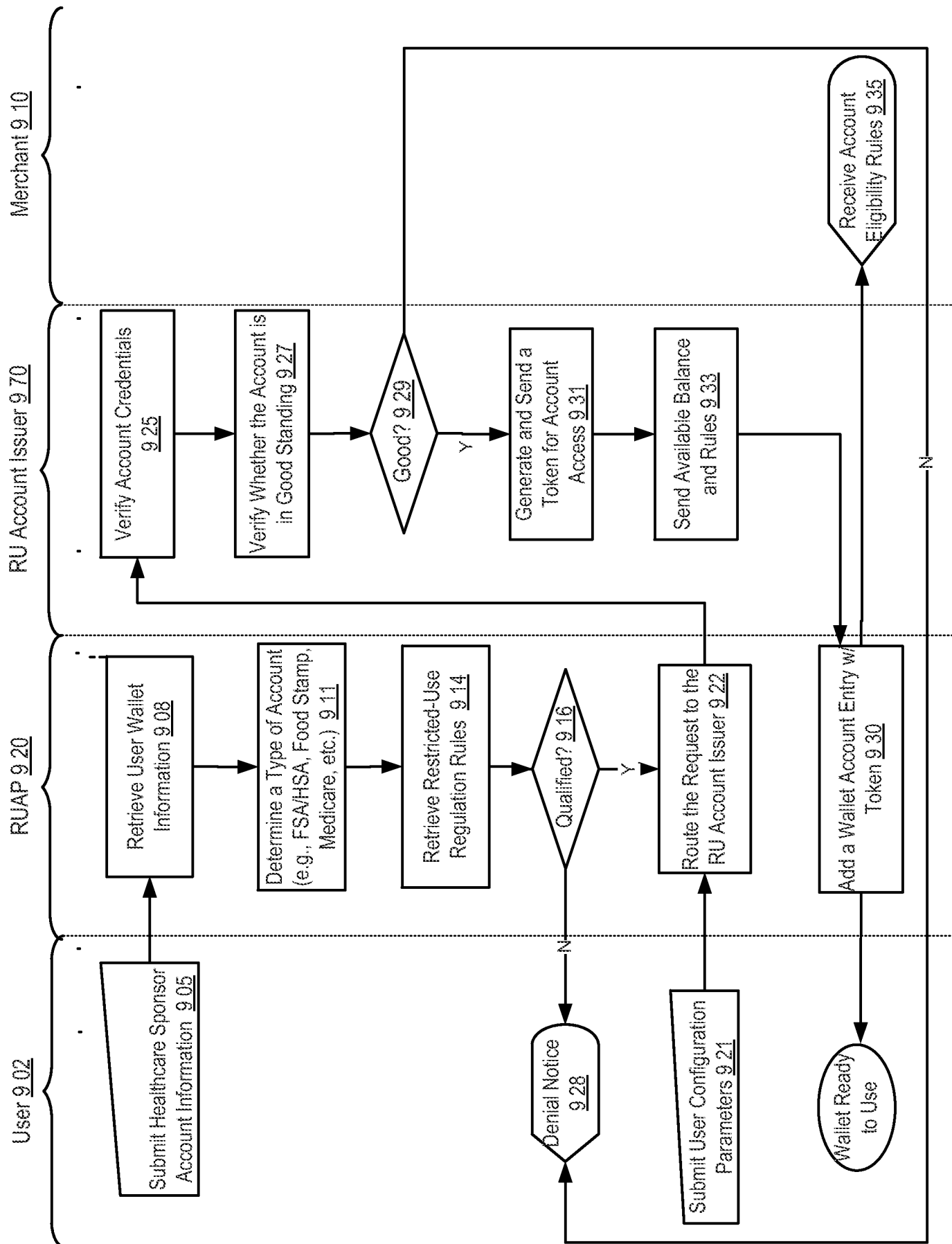


Figure 9B

RUAP Example Logic Flow: Healthcare RU Account Enrollment

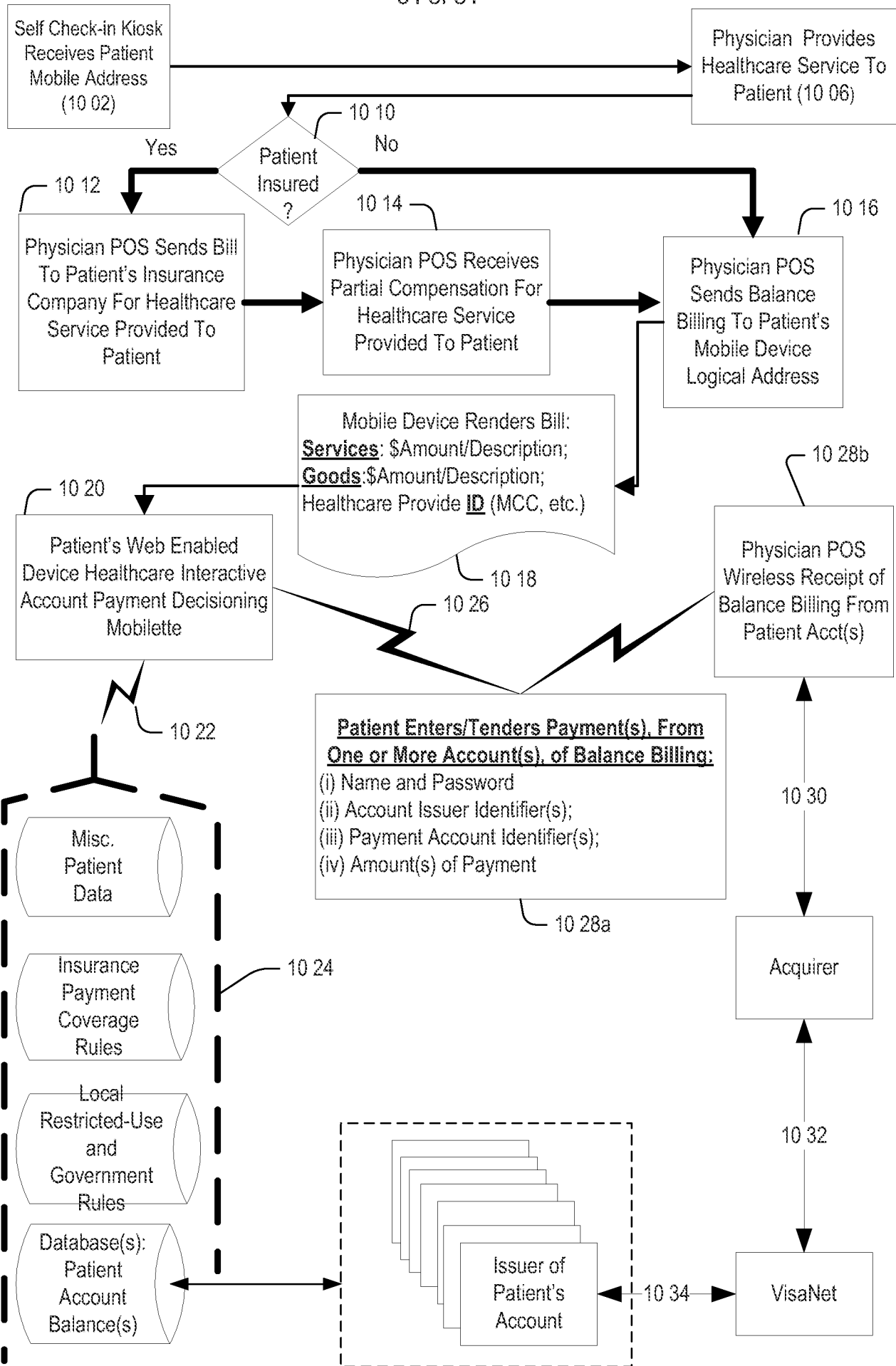


Figure 10A

RUAP Example Logic Flow: Healthcare Payment Overview

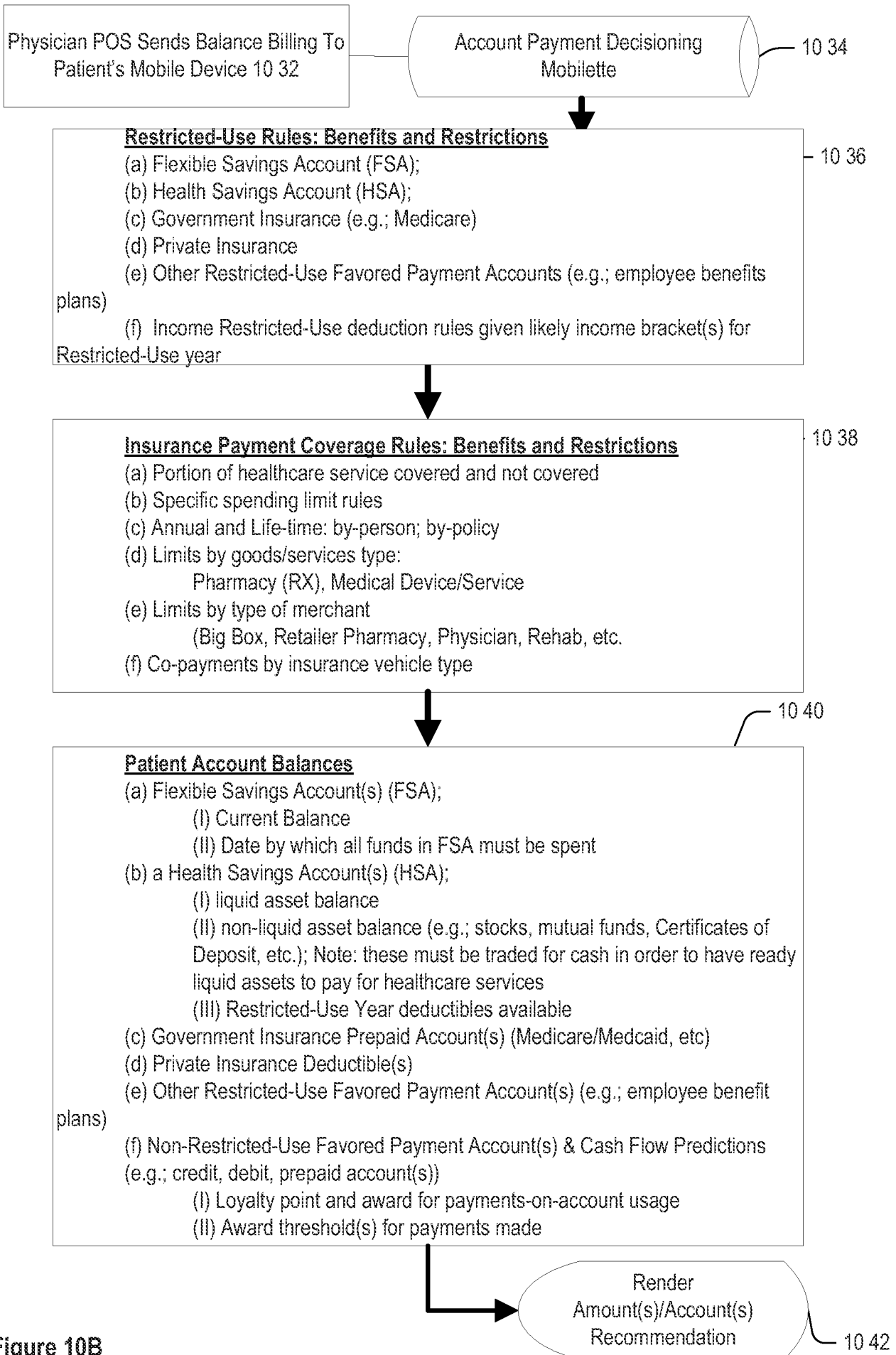


Figure 10B

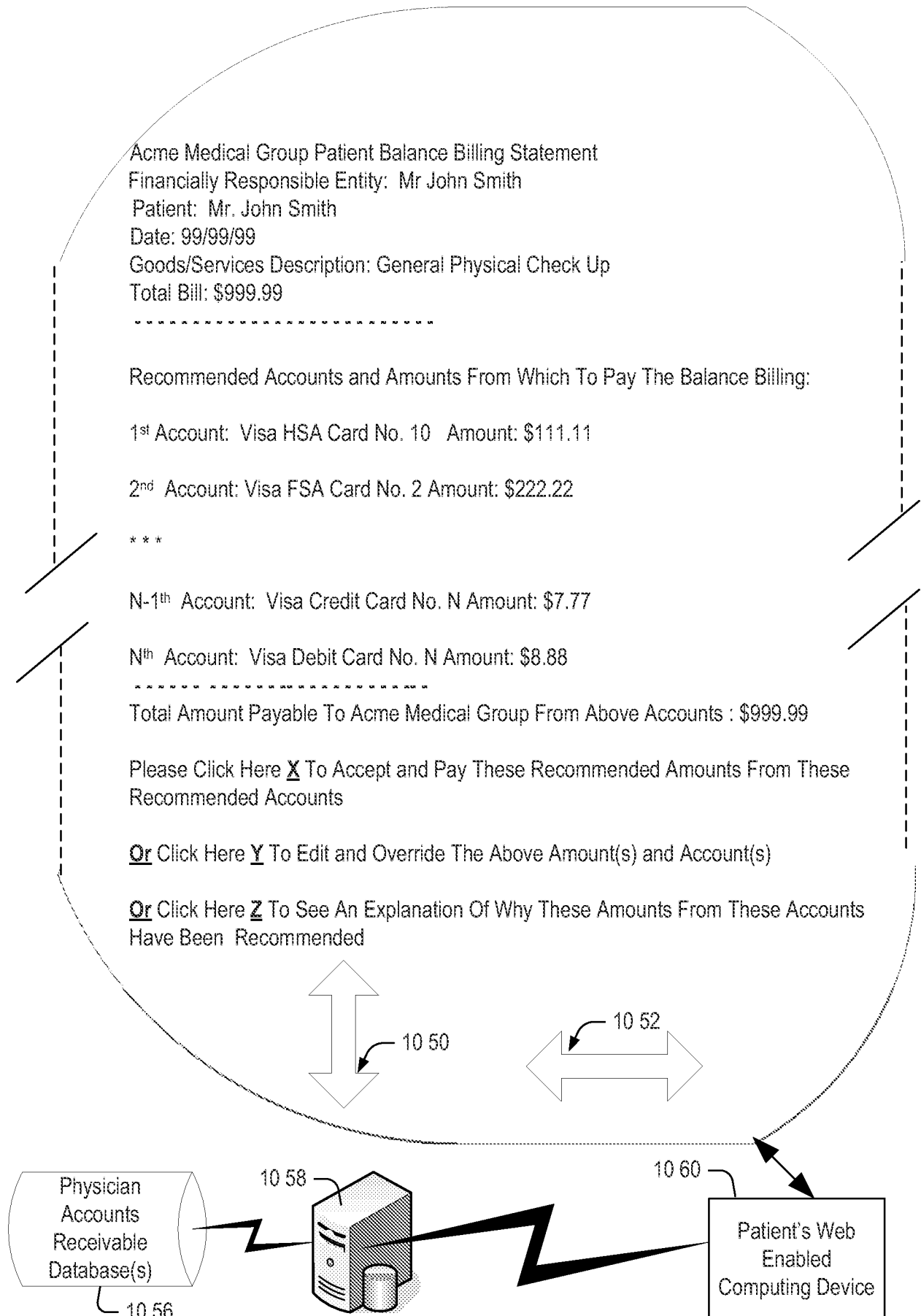


Figure 10C



RUAP Example: Snap Mobile Payment POS Terminal Interface

Figure 11A

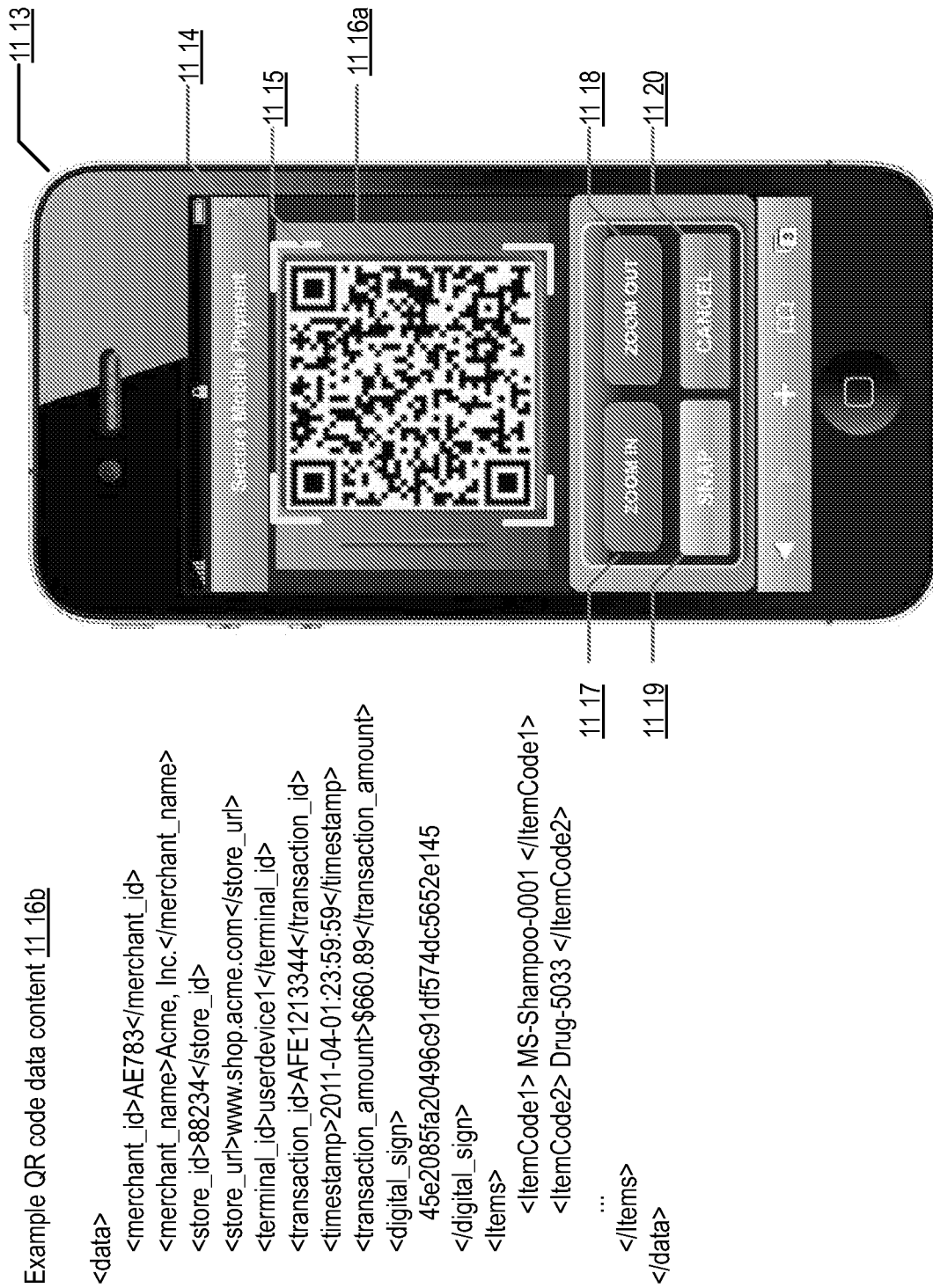


Figure 11B

Example: Snap Mobile App Payment User Interface

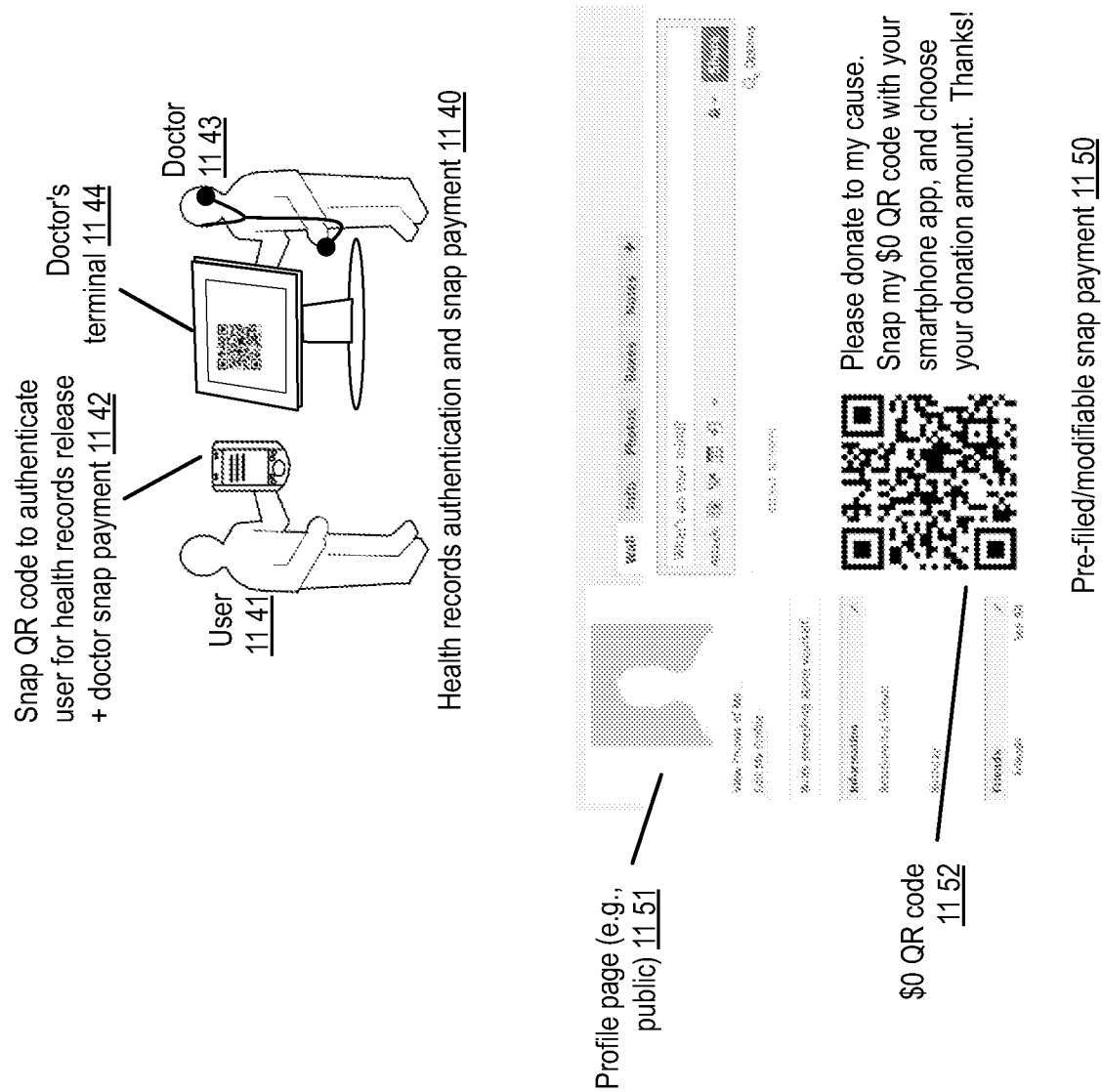


Figure 11C

Example(s): Snap Mobile Payment

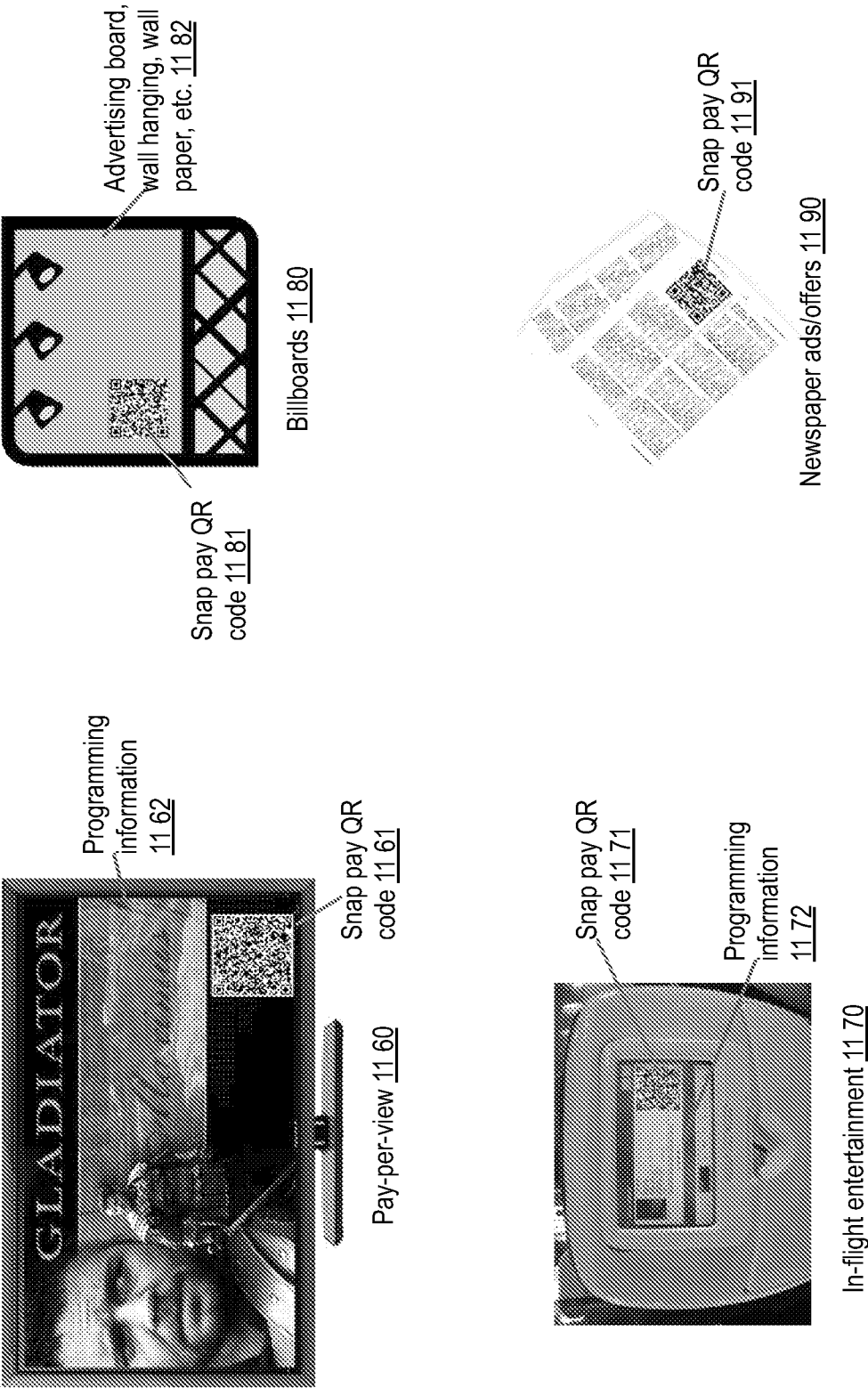


Figure 11D

Example(s): Snap Mobile Applications

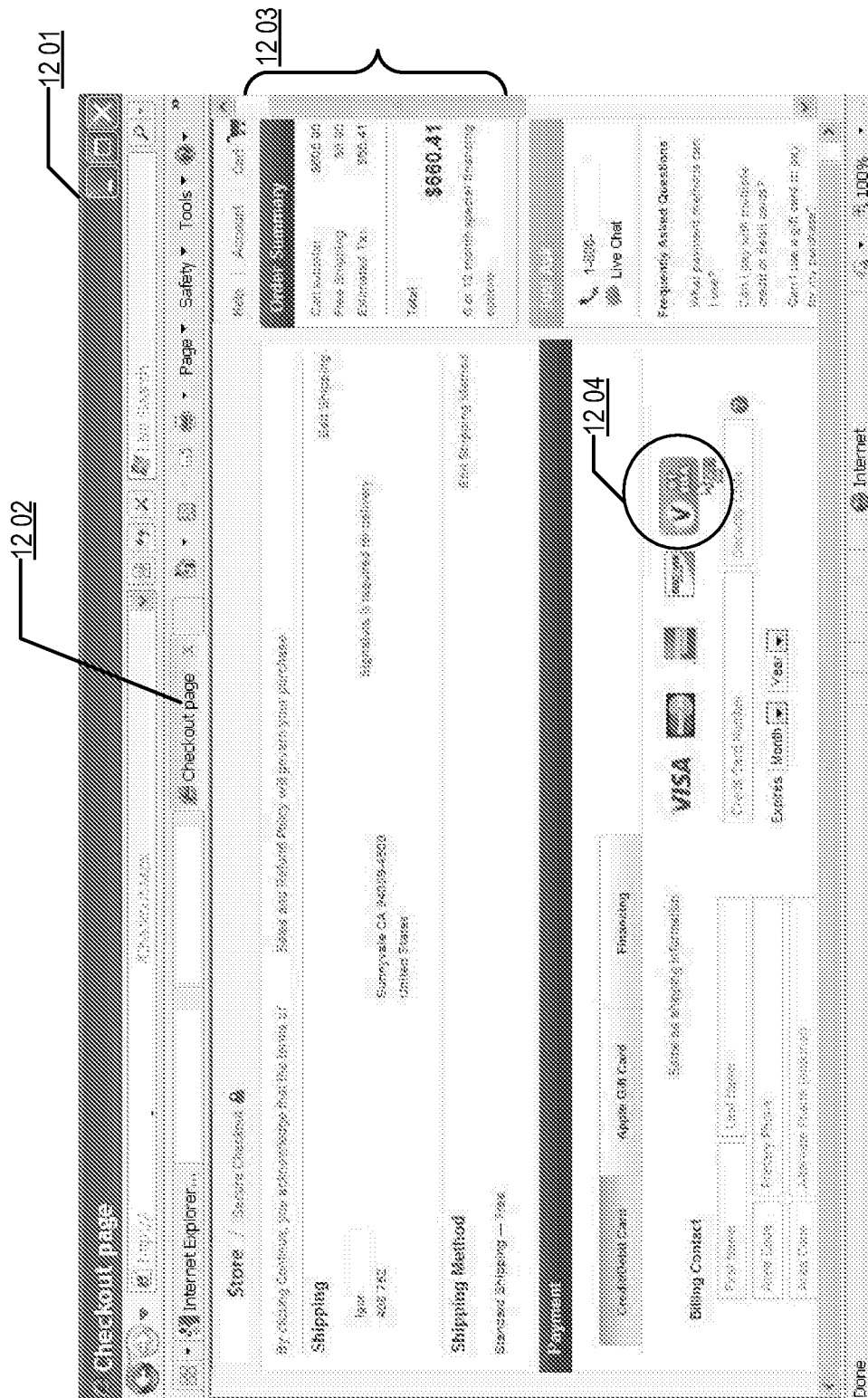


Figure 12A

Example: Snap Mobile Payment User Interface

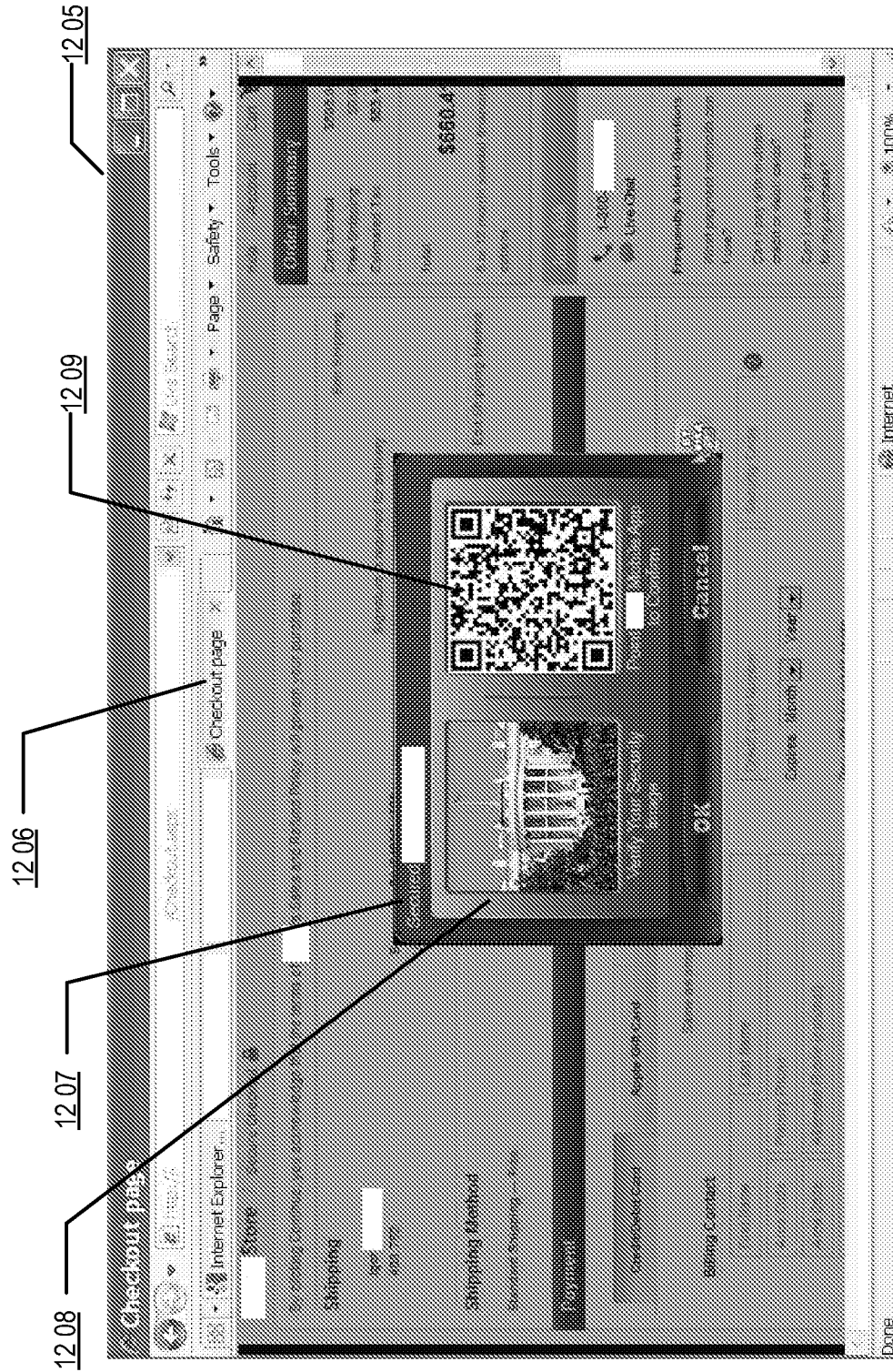


Figure 12B

Example: Snap Mobile Payment Web Interface

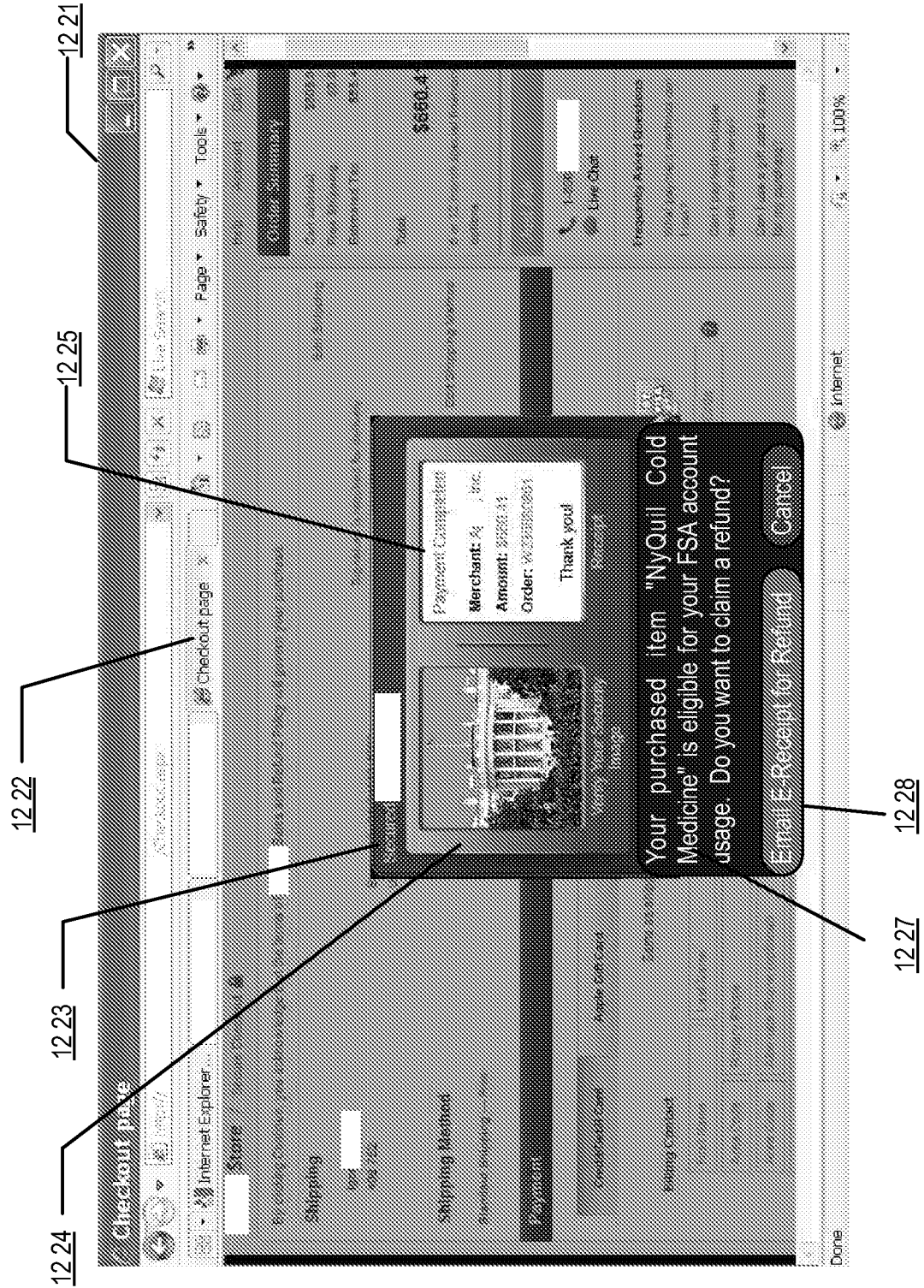


Figure 12C

Example: Snap Mobile App Payment Web Receipt

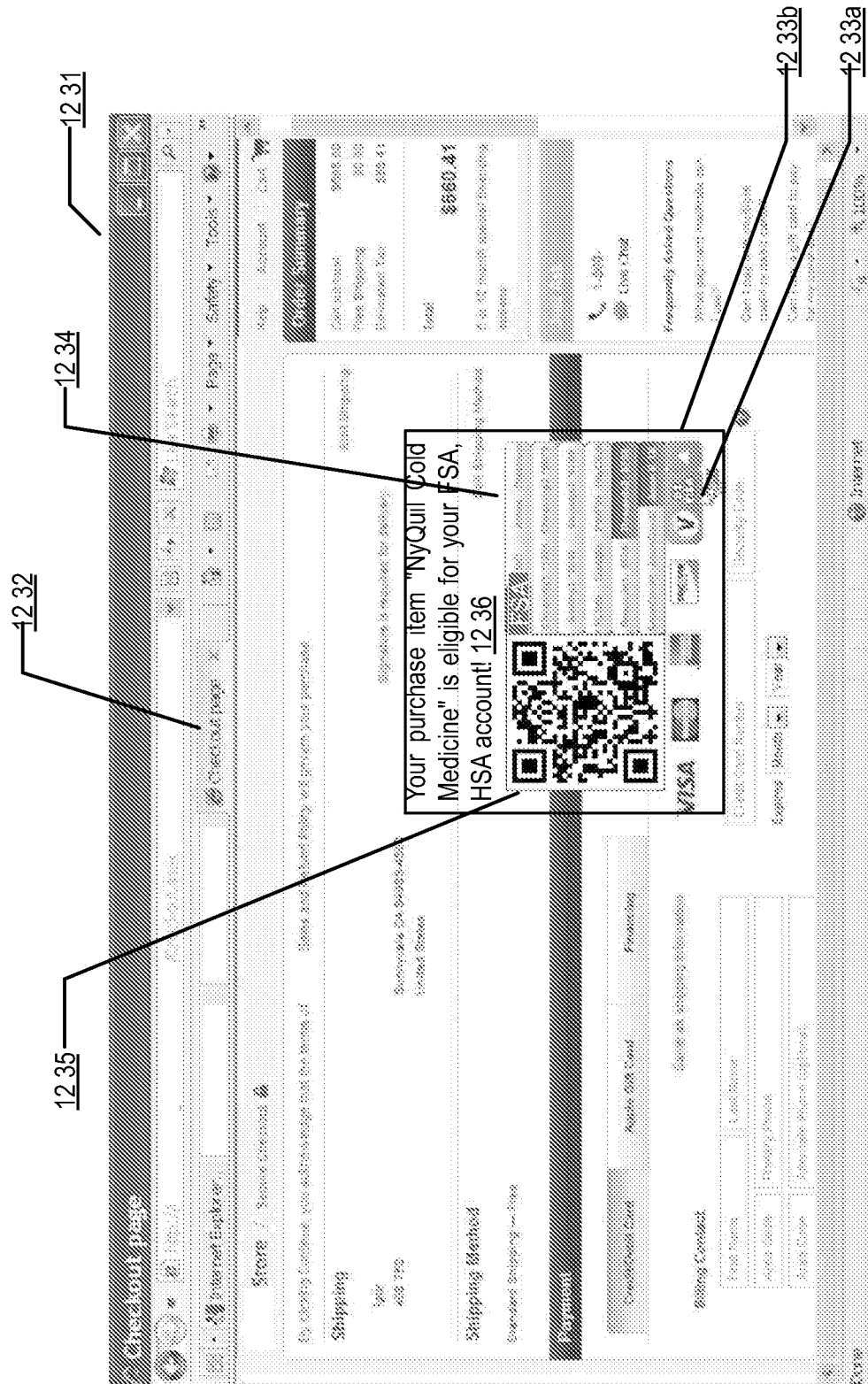


Figure 12D

RUAP Example: Web Checkout

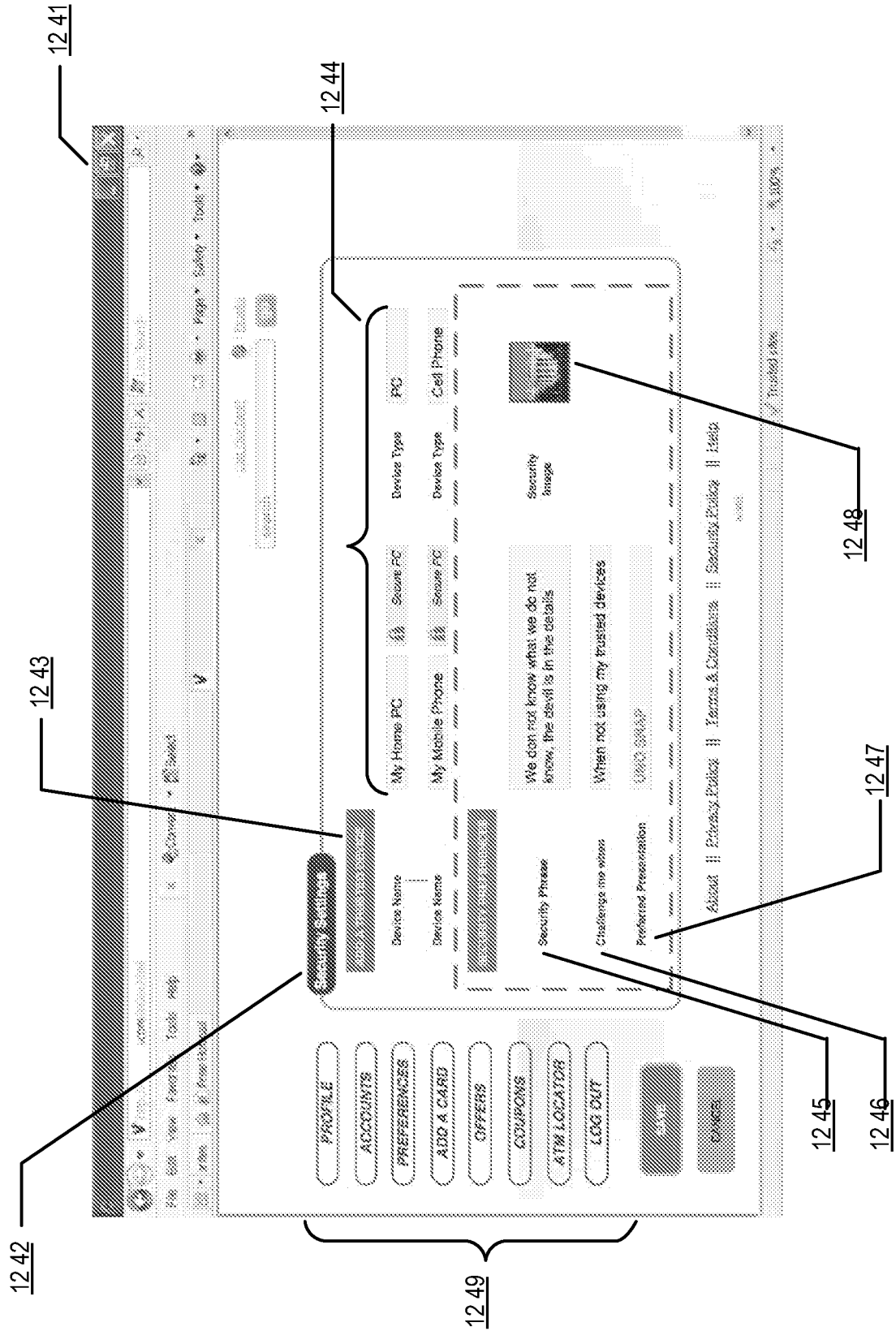


Figure 12E

Example: Snap Mobile Payment: Security / Fraud Prevention

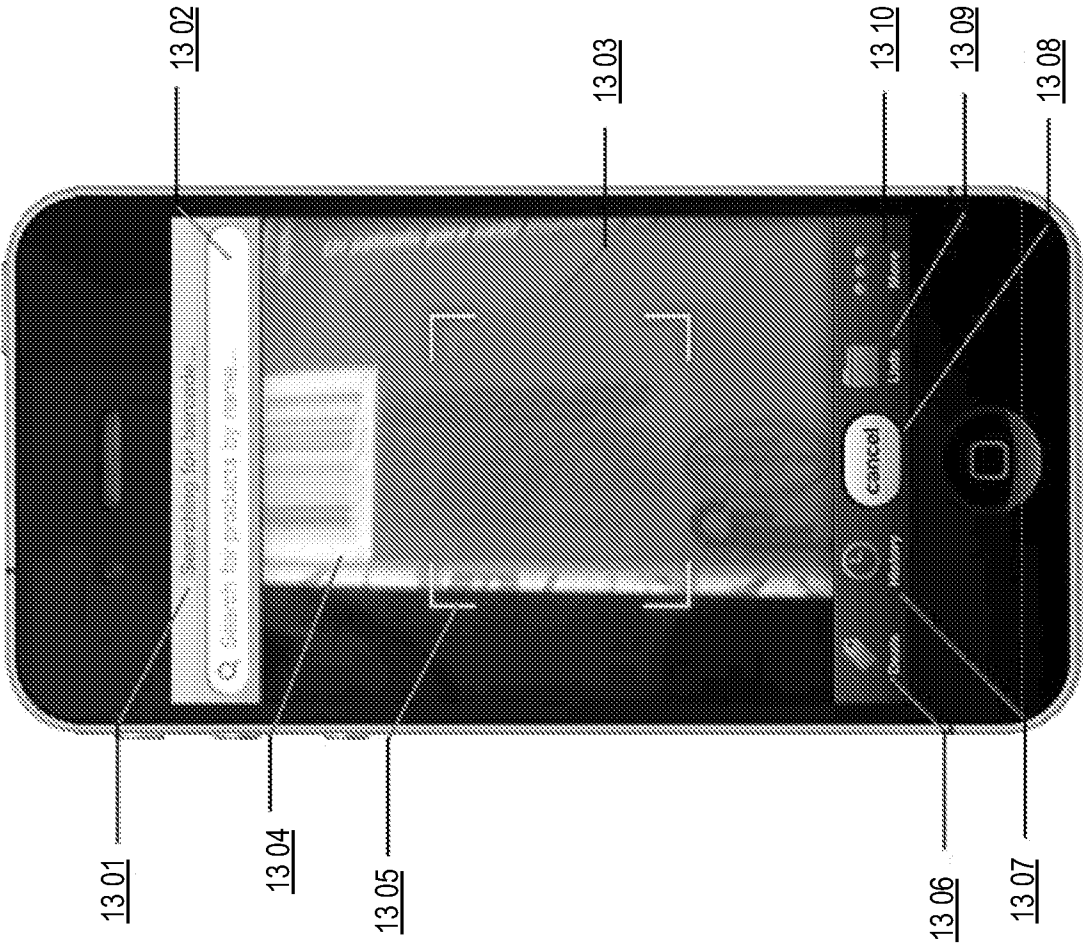


Figure 13A

Example: One-Tap Mobile App: Barcode Capture

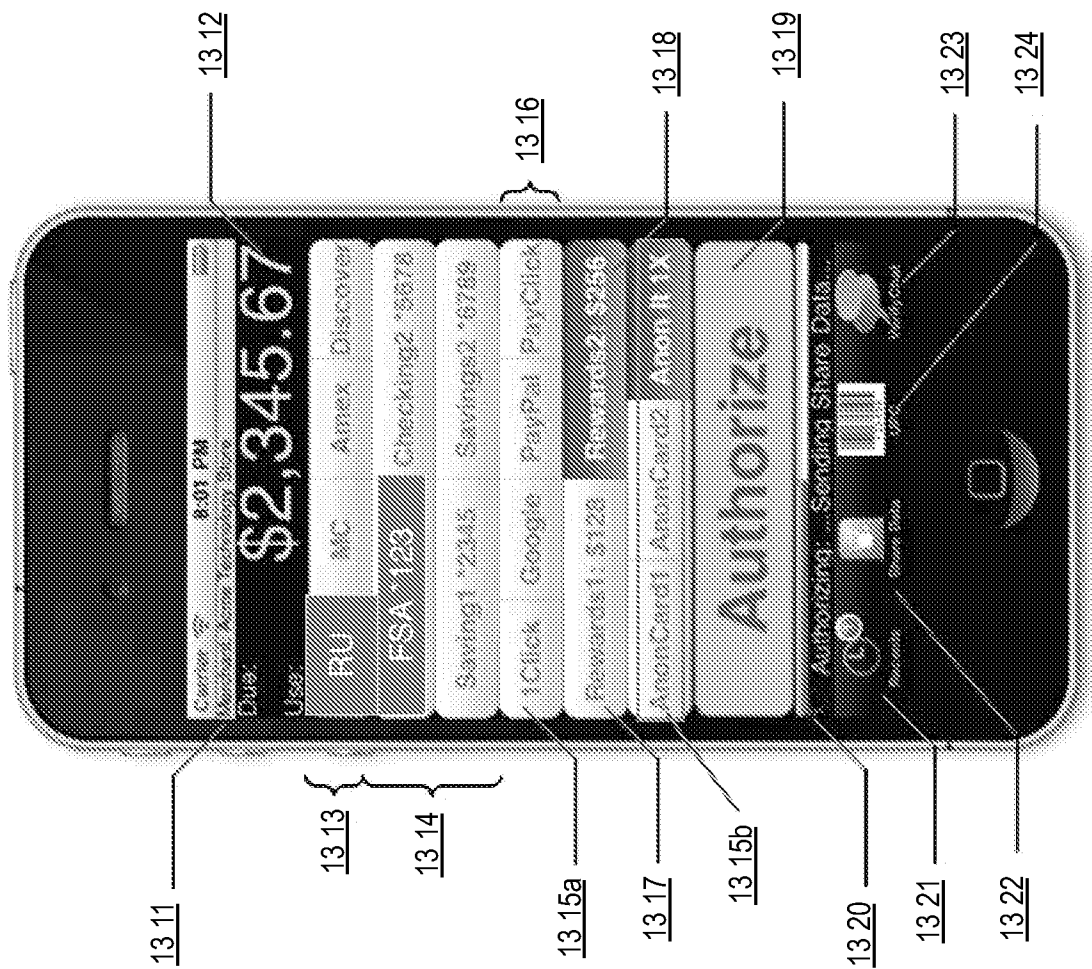


Figure 13B

Example: One-Tap Mobile App: Payment Options

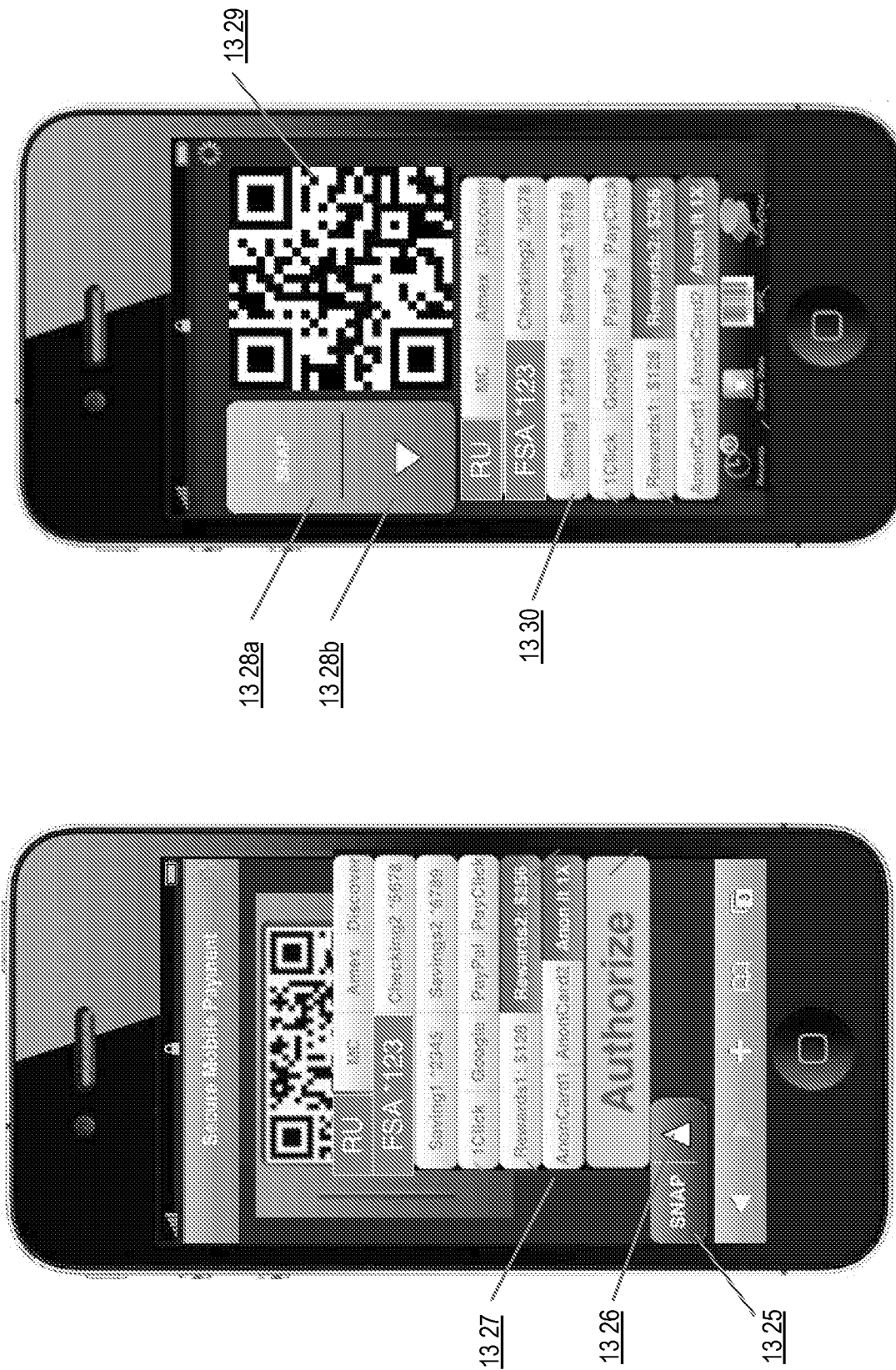


Figure 13C

Example: One-Tap Mobile App: Payment Options

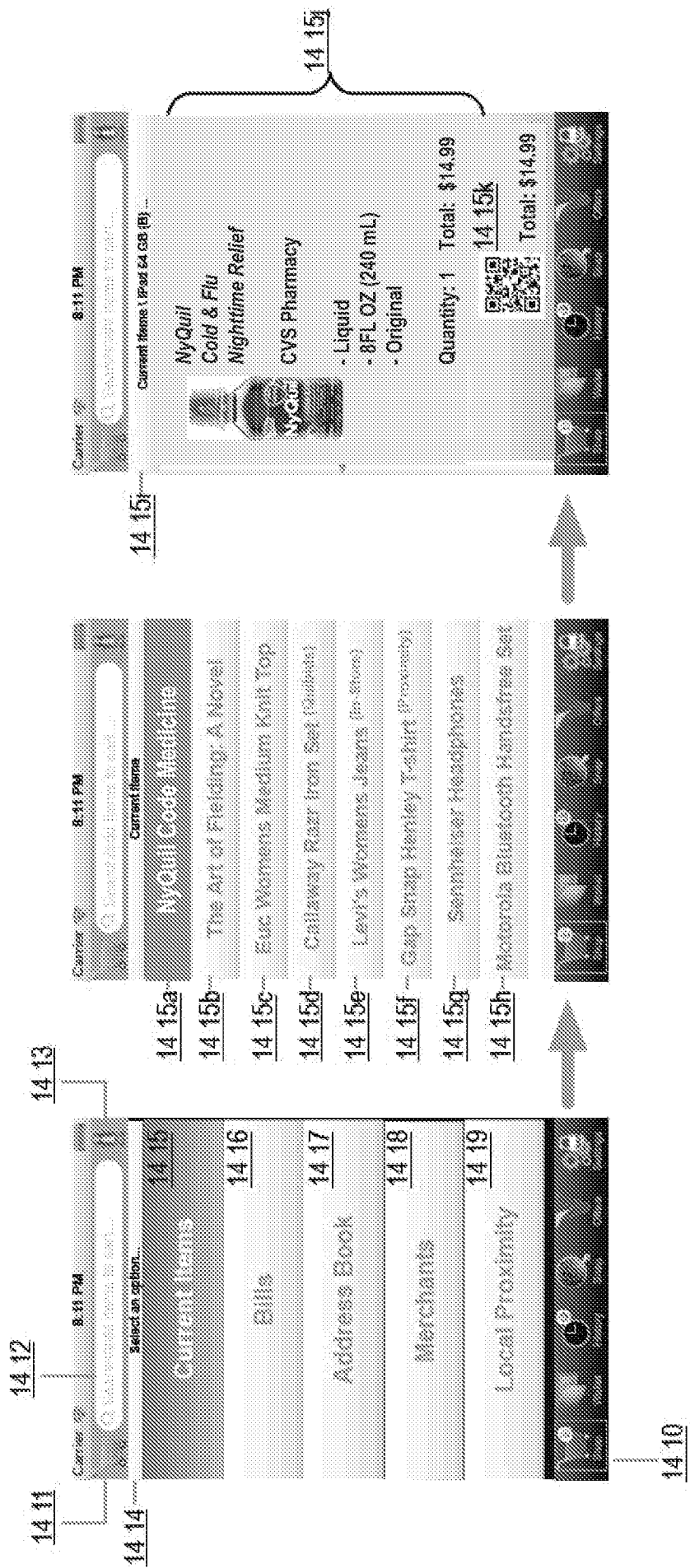


Figure 14A

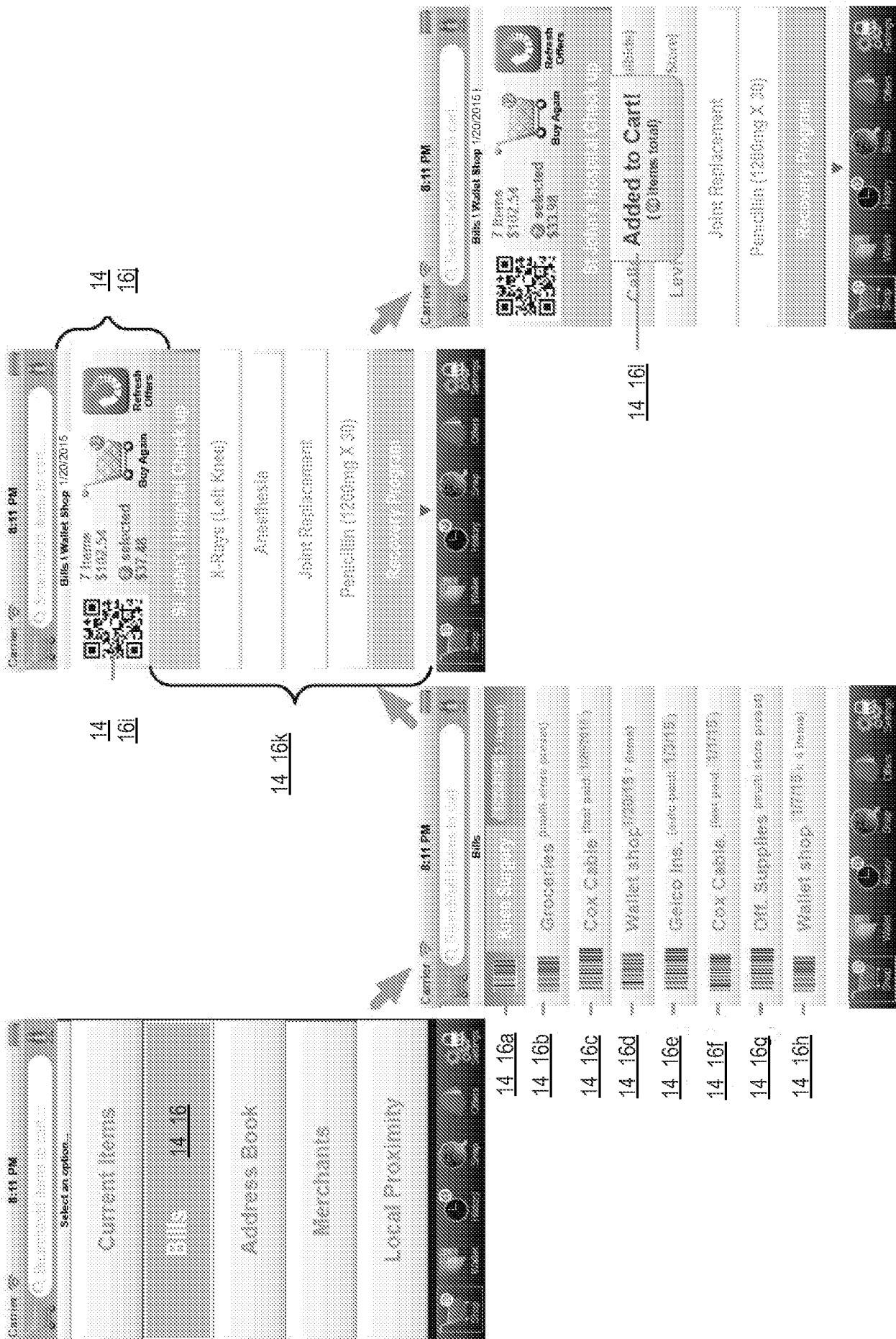


Figure 14B

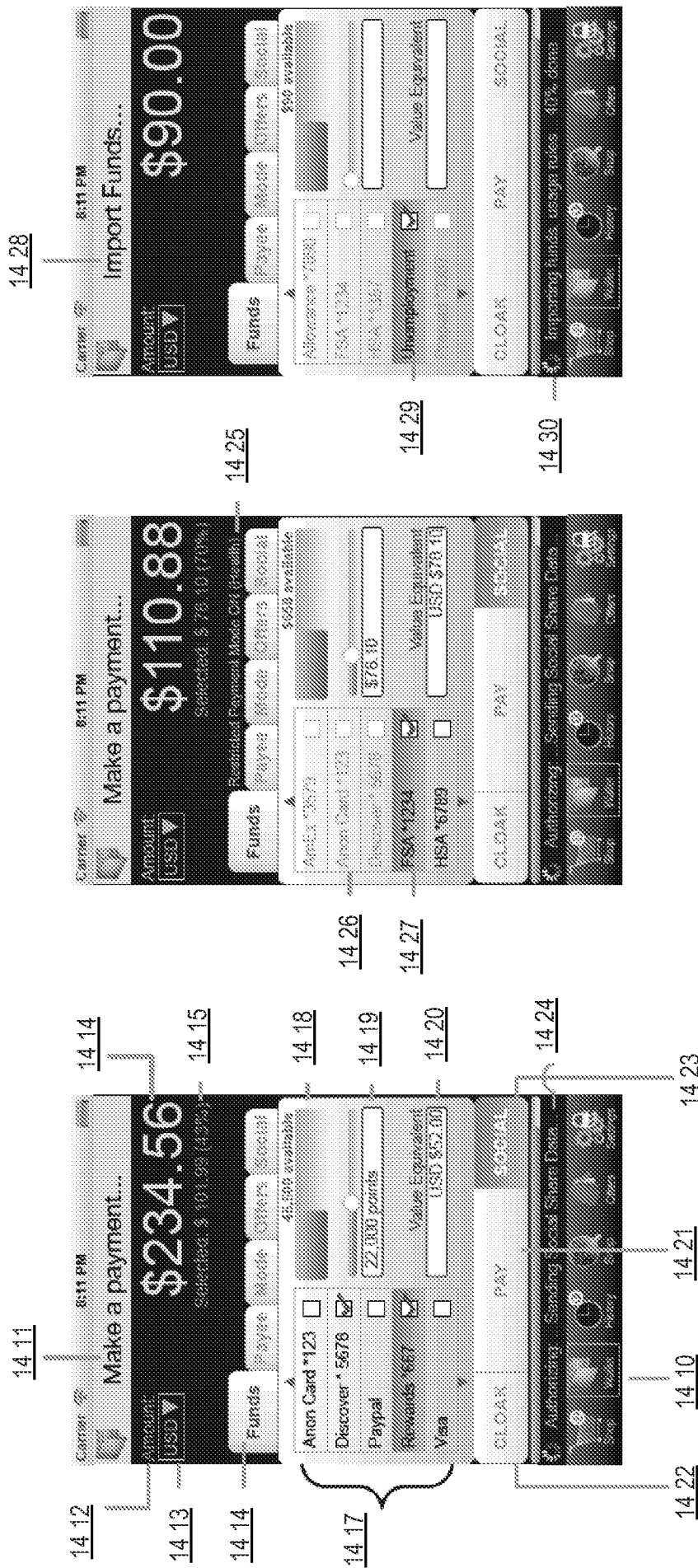


Figure 14C



Figure 14D

Example: Virtual Wallet Mobile App - Shopping Mode

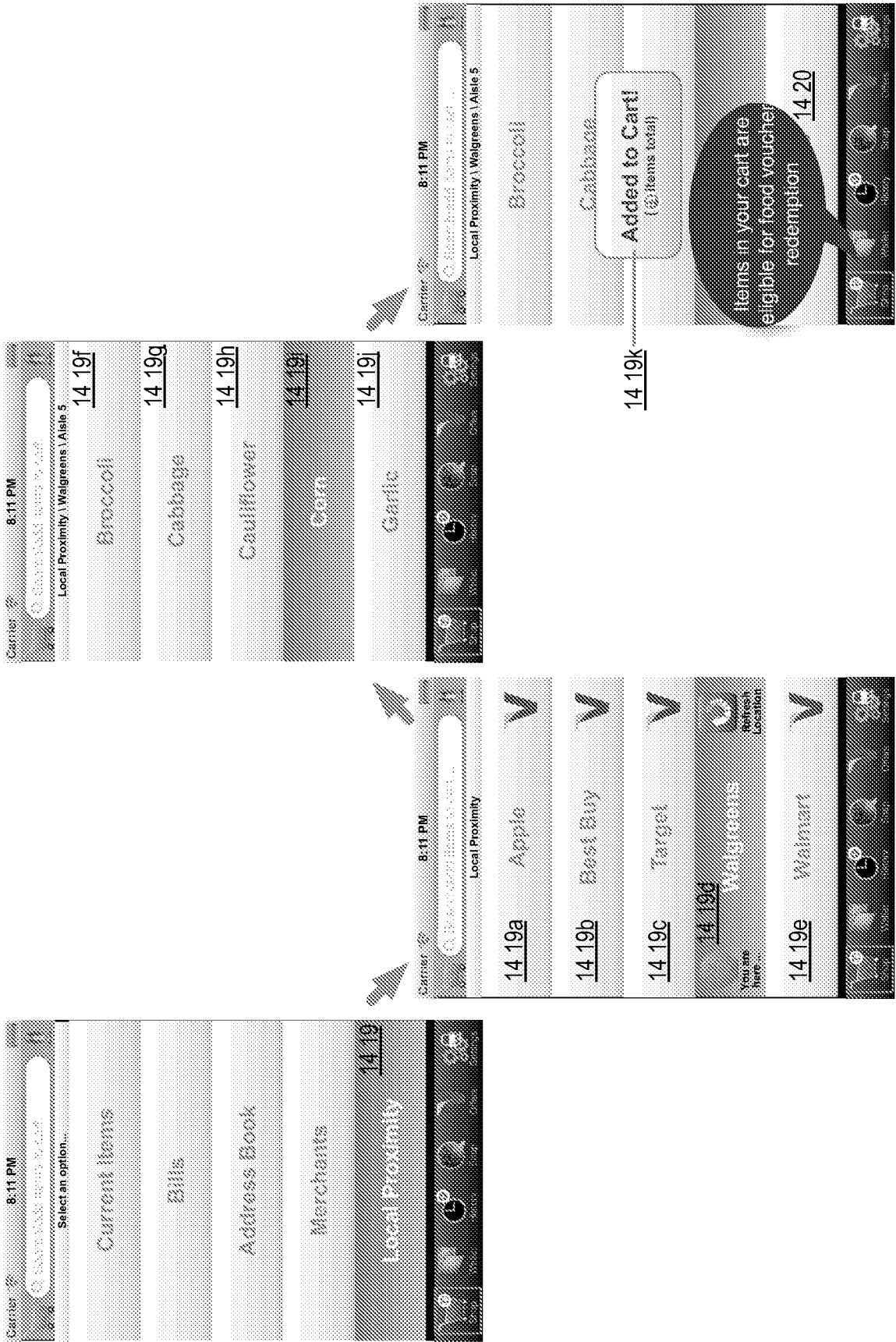
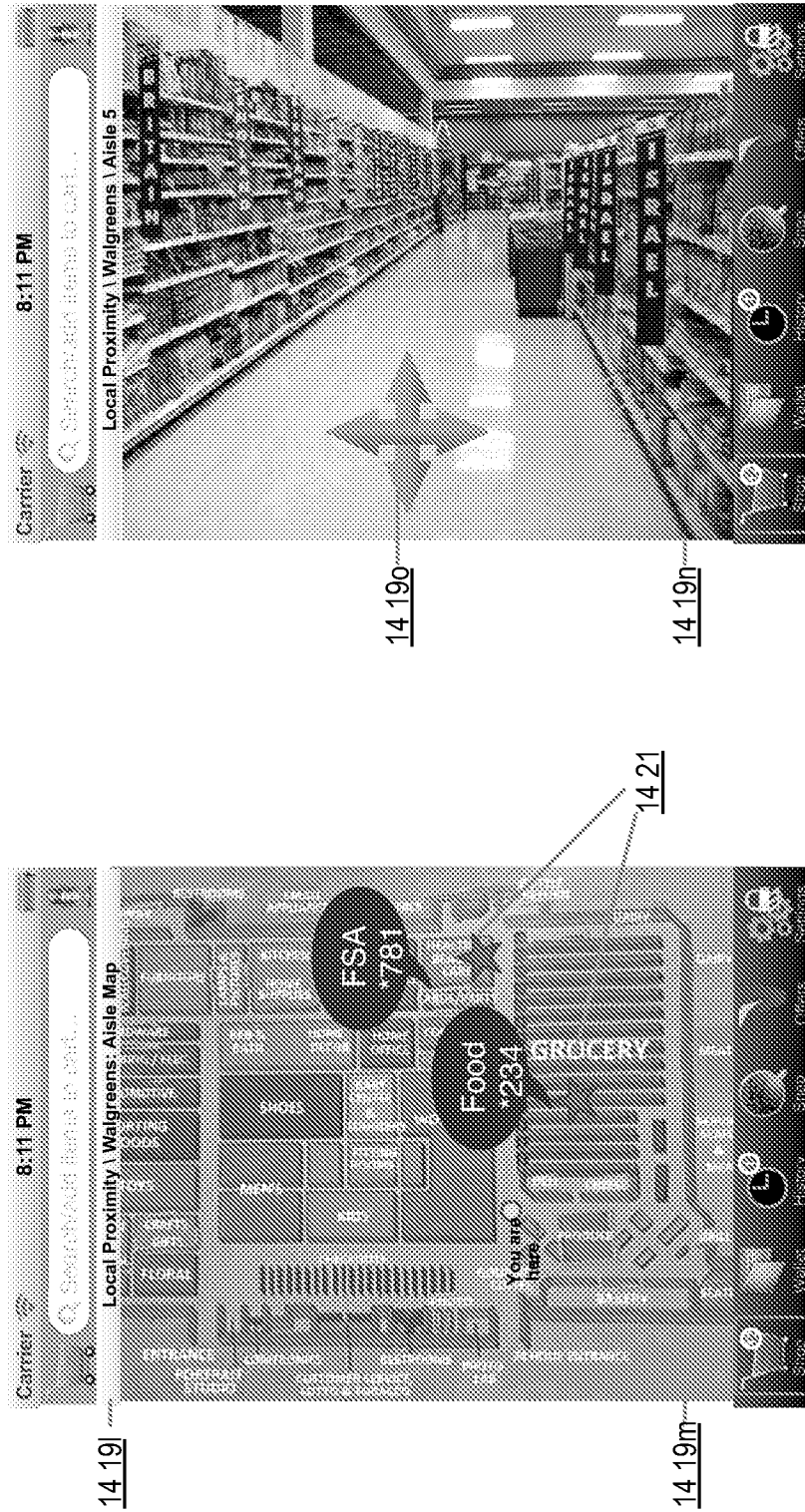


Figure 14E

Example: Virtual Wallet Mobile App - Shopping Mode



Example: Virtual Wallet Mobile App - Shopping Mode

Figure 14F

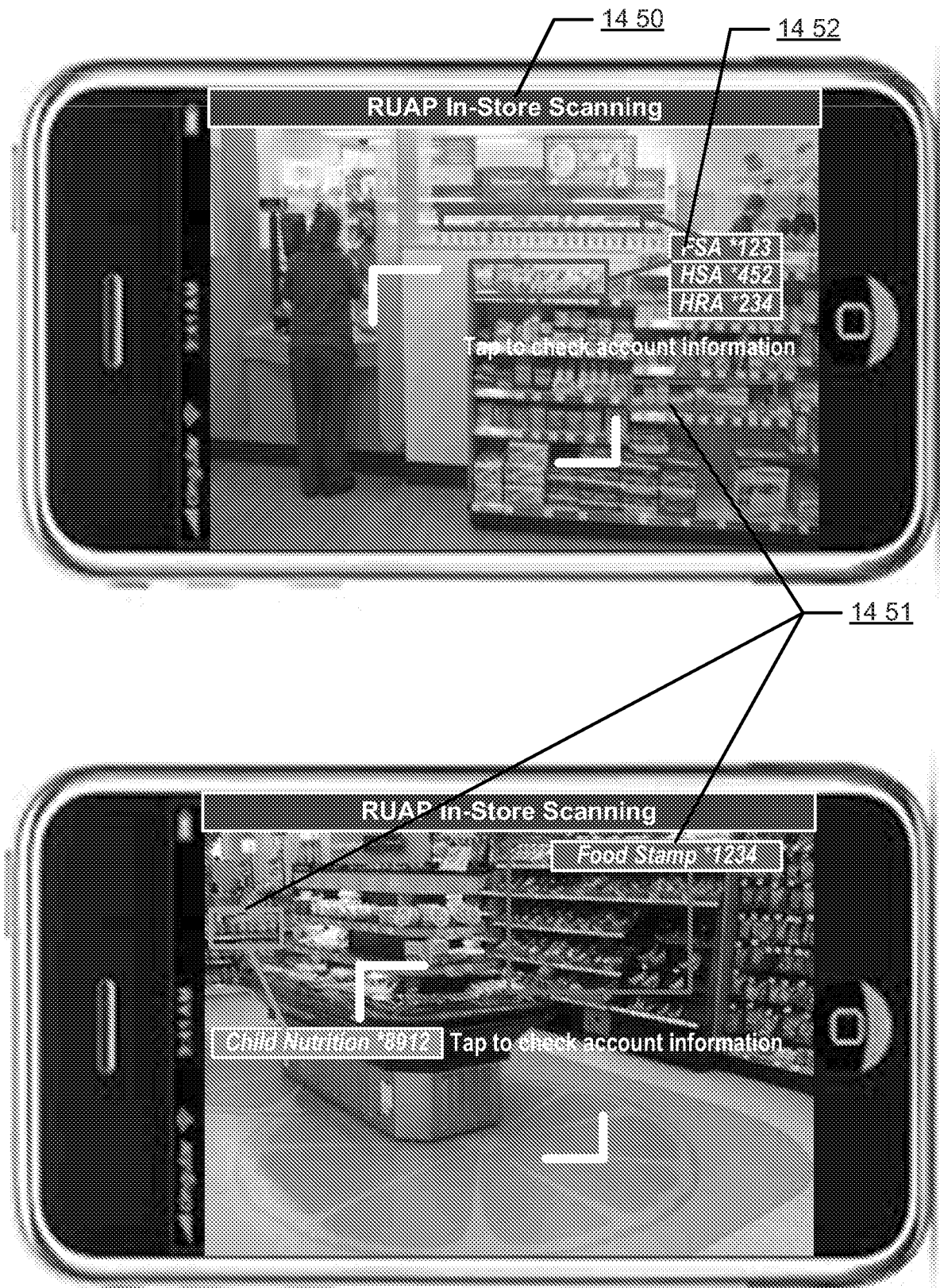


Figure 14G

RUAP Example Mobile Wallet UI: Augmented Reality In-Store

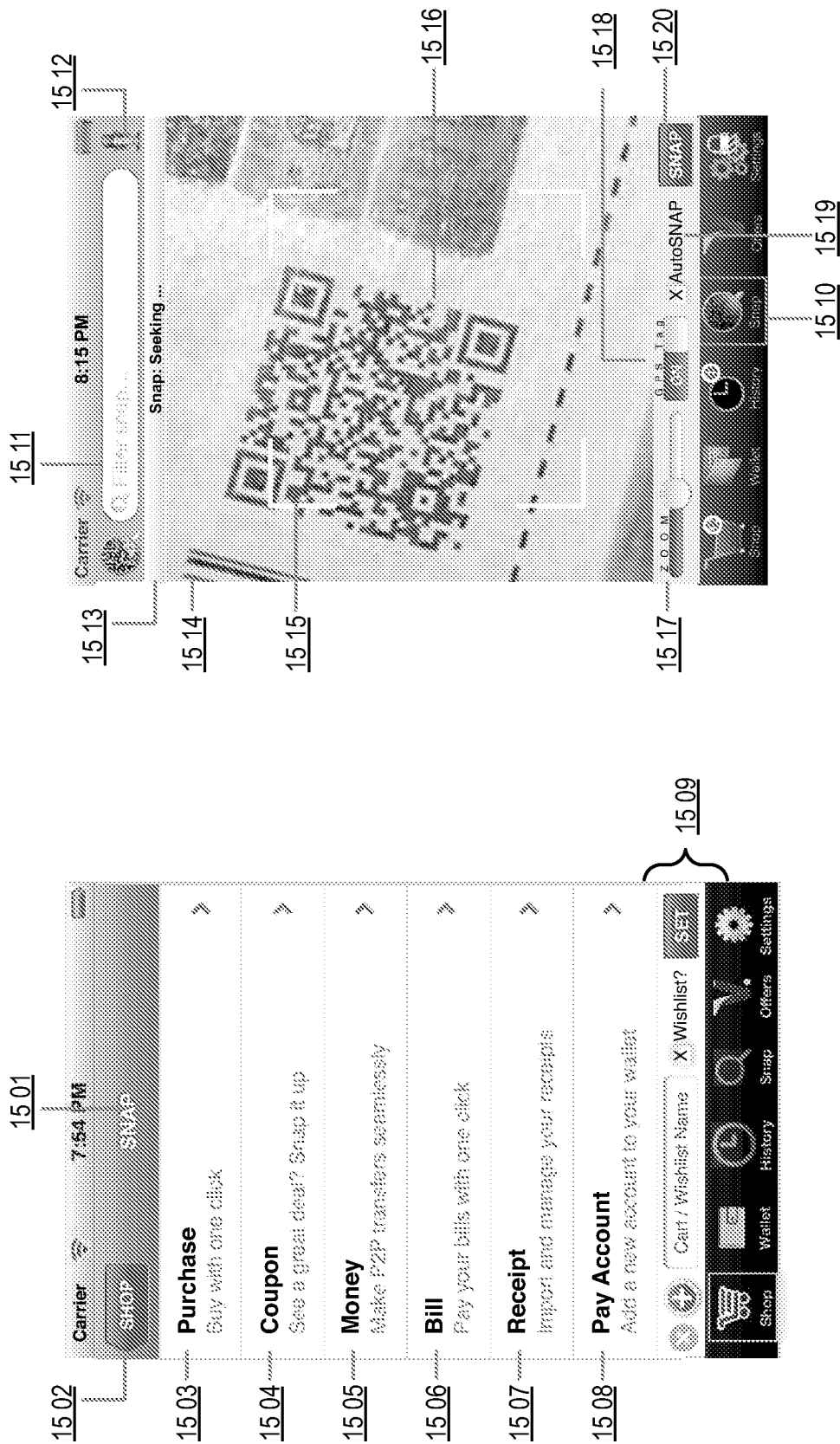


Figure 15A

Example: Virtual Wallet Mobile App - Snap Mode



Figure 15B

Example: Virtual Wallet Mobile App - Snap Mode



Figure 15C

Example: Virtual Wallet Mobile App - Snap Mode



Example: Virtual Wallet Mobile App - Snap Mode

Figure 15D

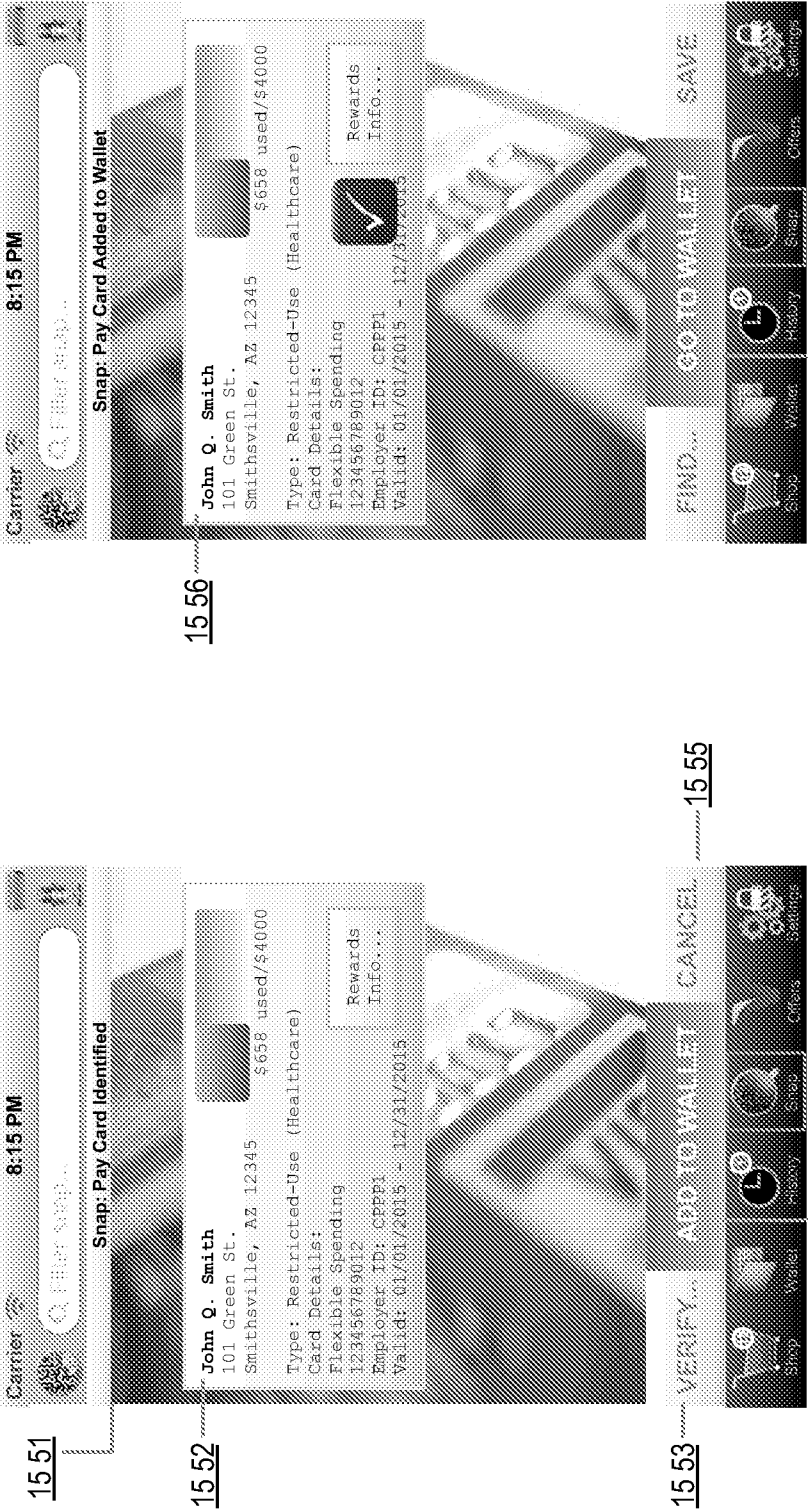


Figure 15E

Example: Virtual Wallet Mobile App - Snap Mode

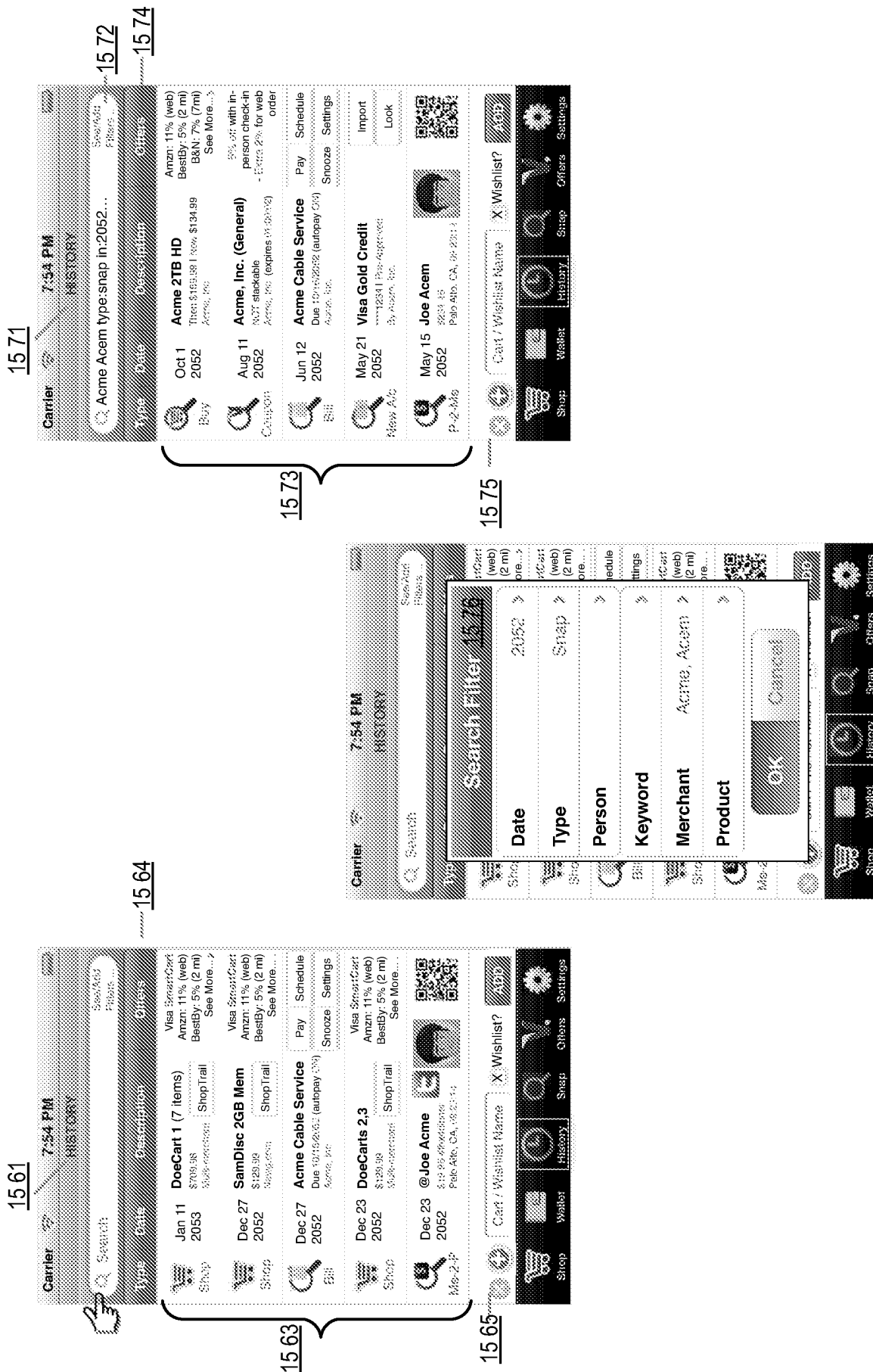


Figure 15F

Example: RUAP Application Embodiment - History Mode

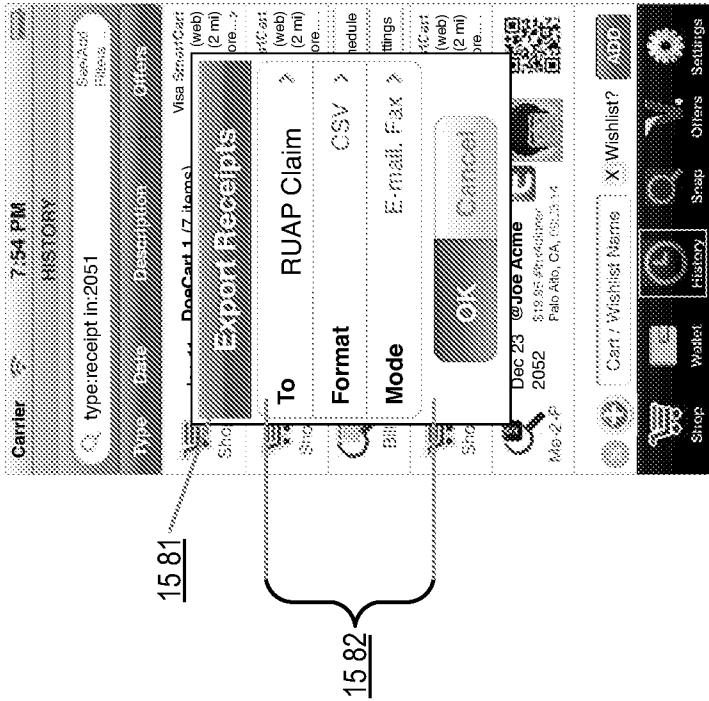
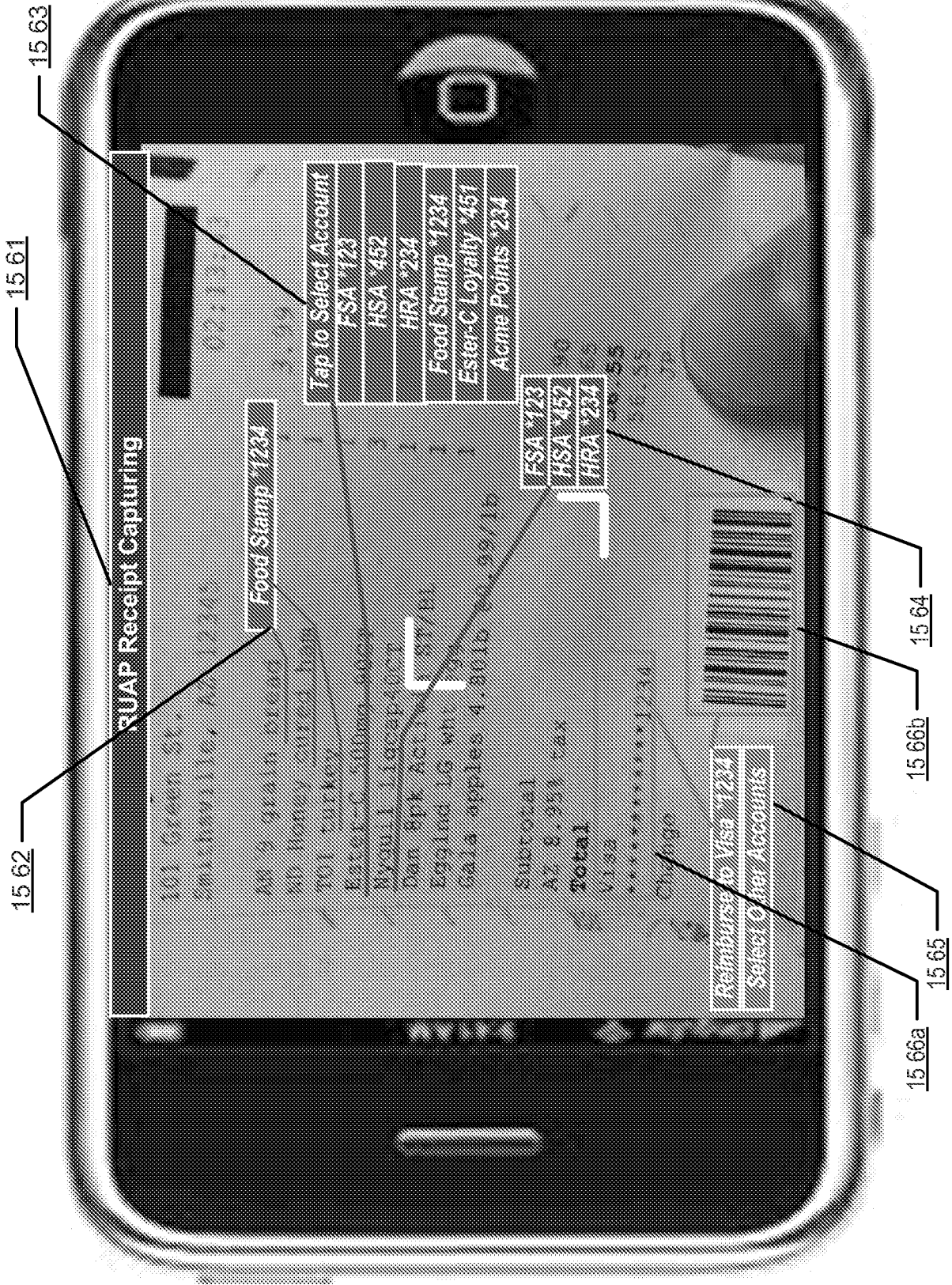


Figure 15G

Example: RUAP Application Embodiment - History Mode

Figure 15H



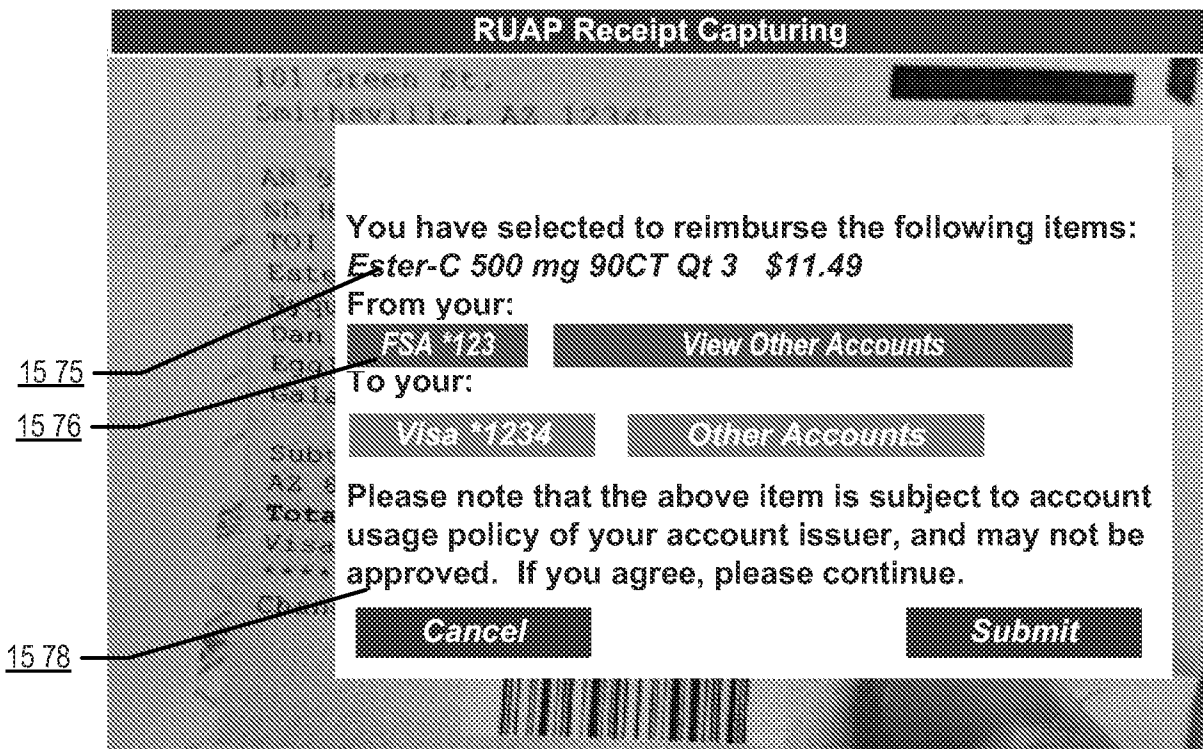
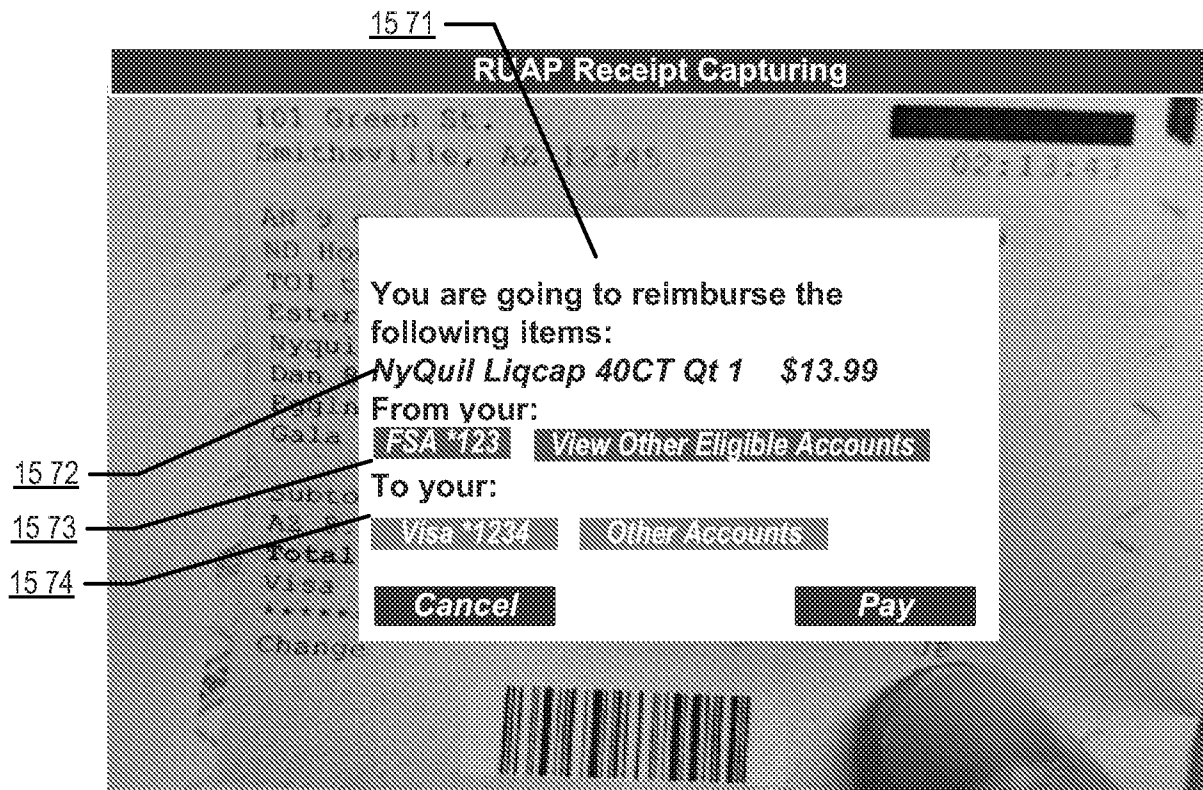
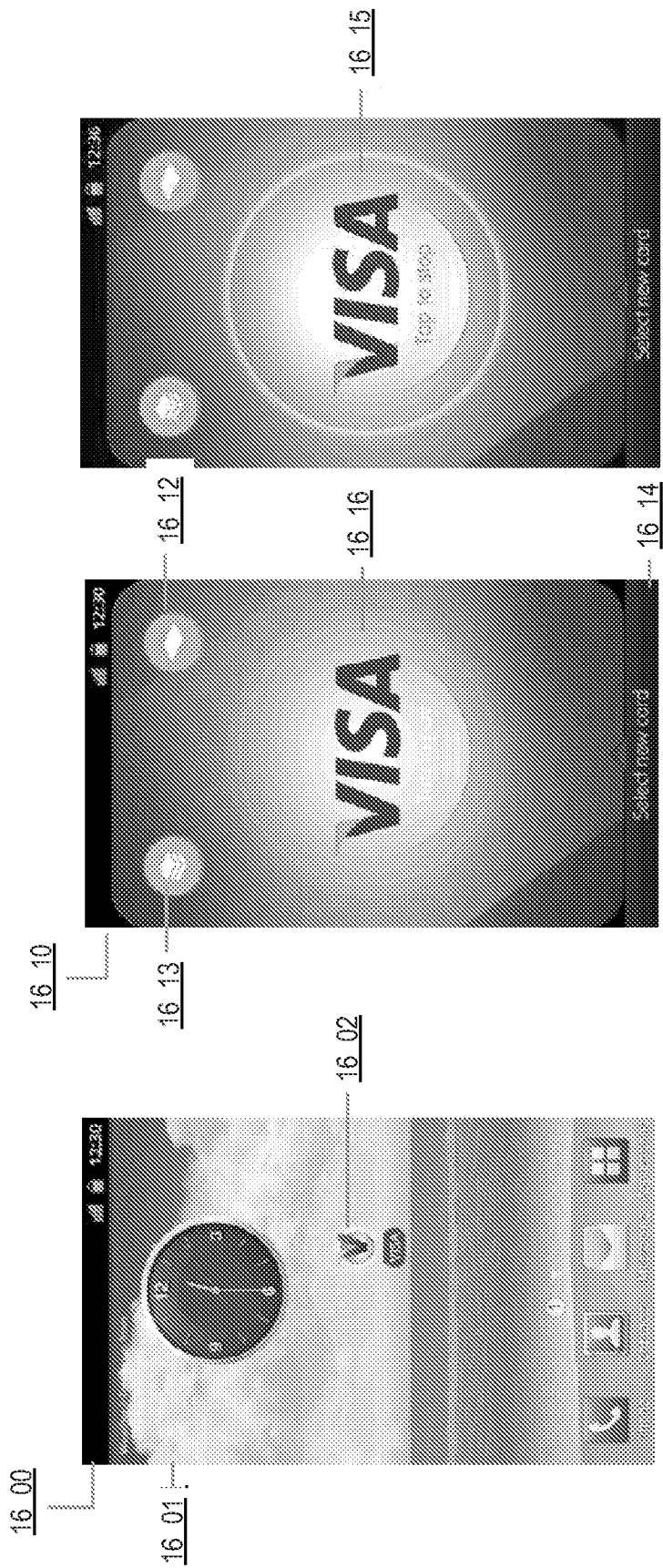


Figure 151

RUAP Example Mobile Wallet UI: Augmented Reality Receipt Capturing



RUAP Example UI: Healthcare Mobile Wallet

Figure 16A

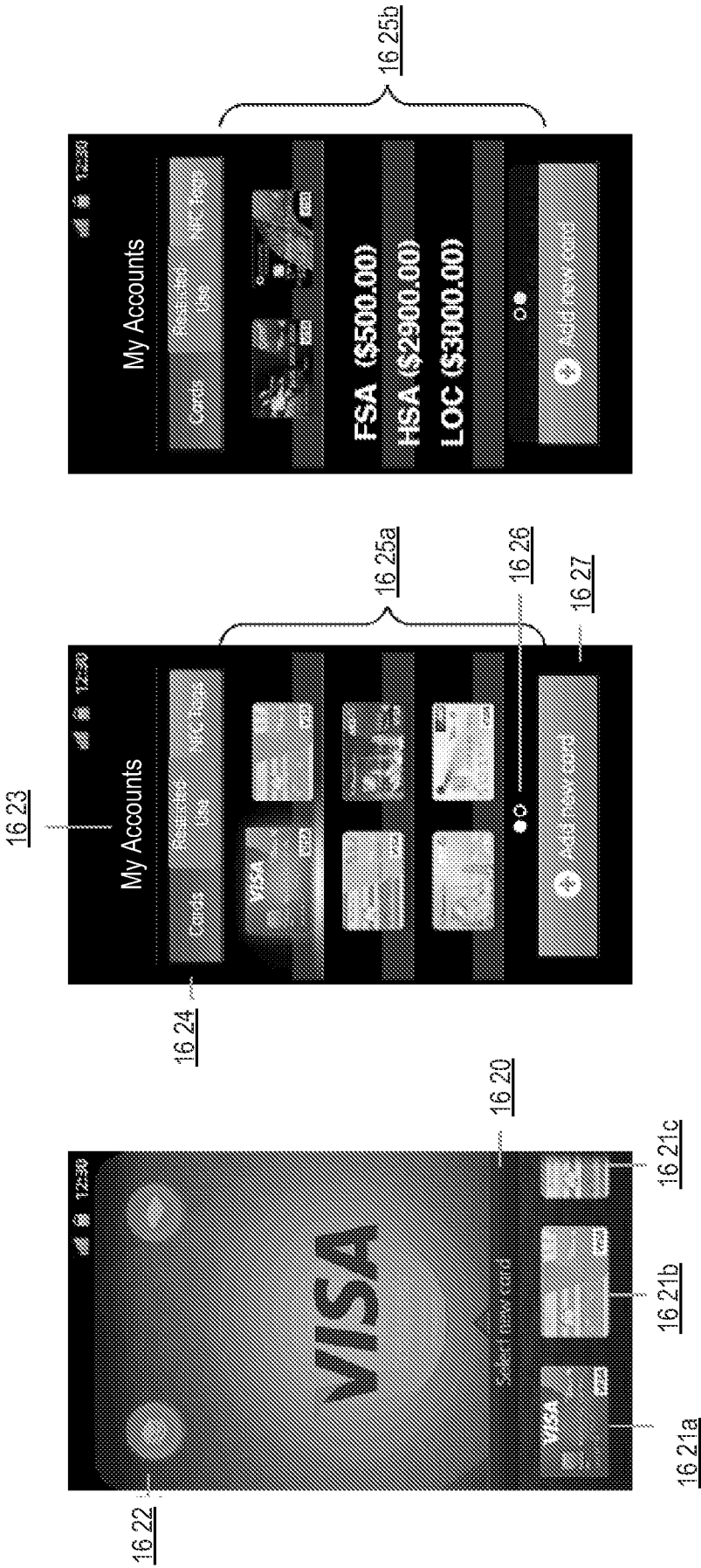
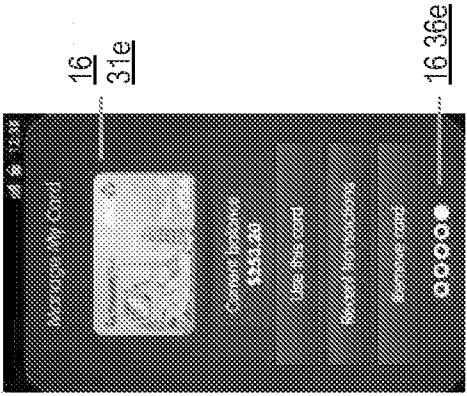
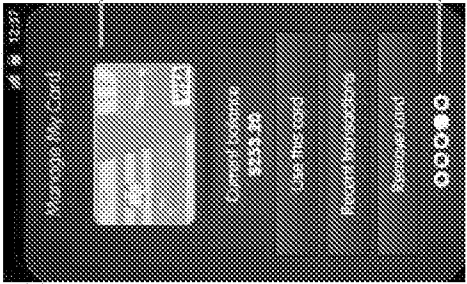
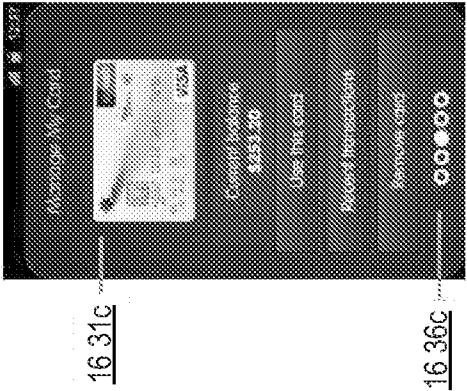
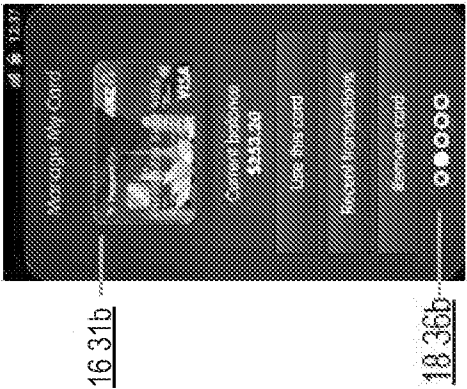
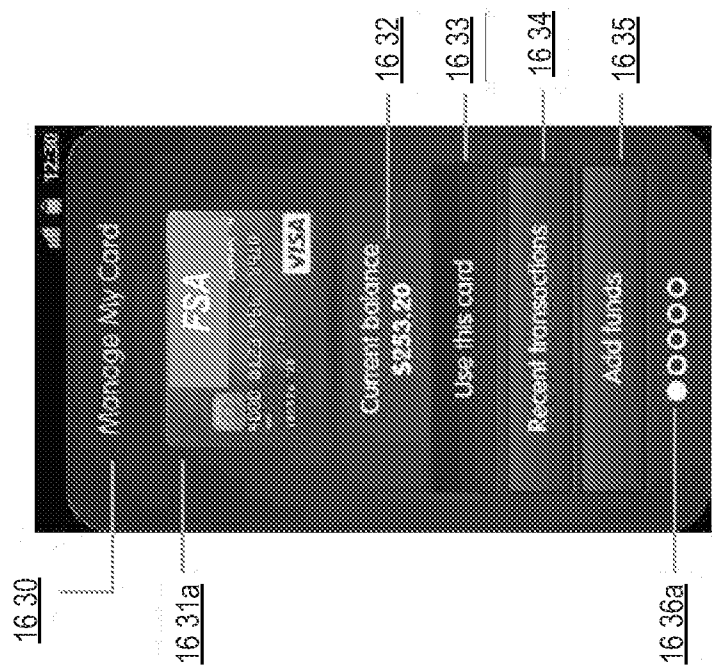


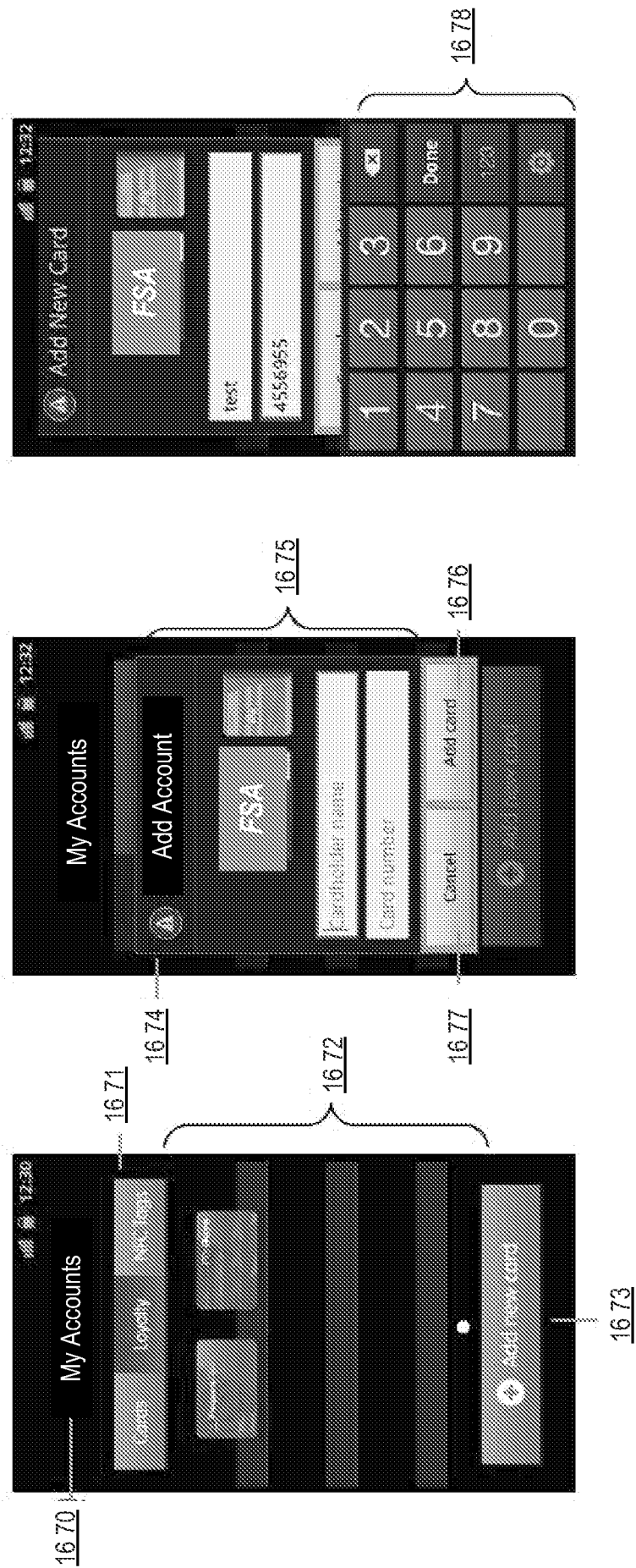
Figure 16B

RUAP Example UI: Account List



RUAP Example UI: Account Management

Figure 16C



RUAP Example UI: Add a New Account

Figure 16D

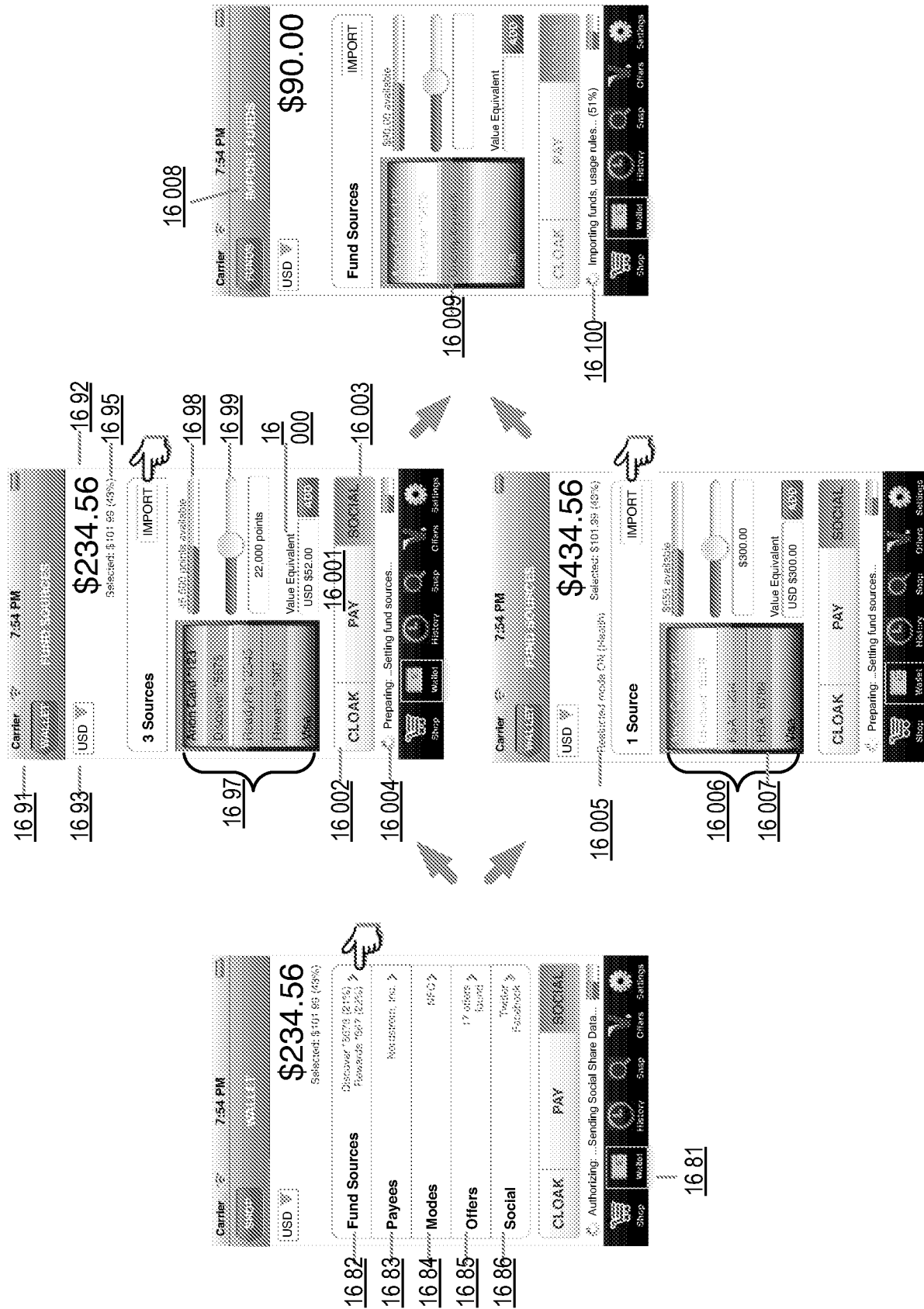


Figure 16E

Example: UEP Application Embodiment - Wallet Funds Mode

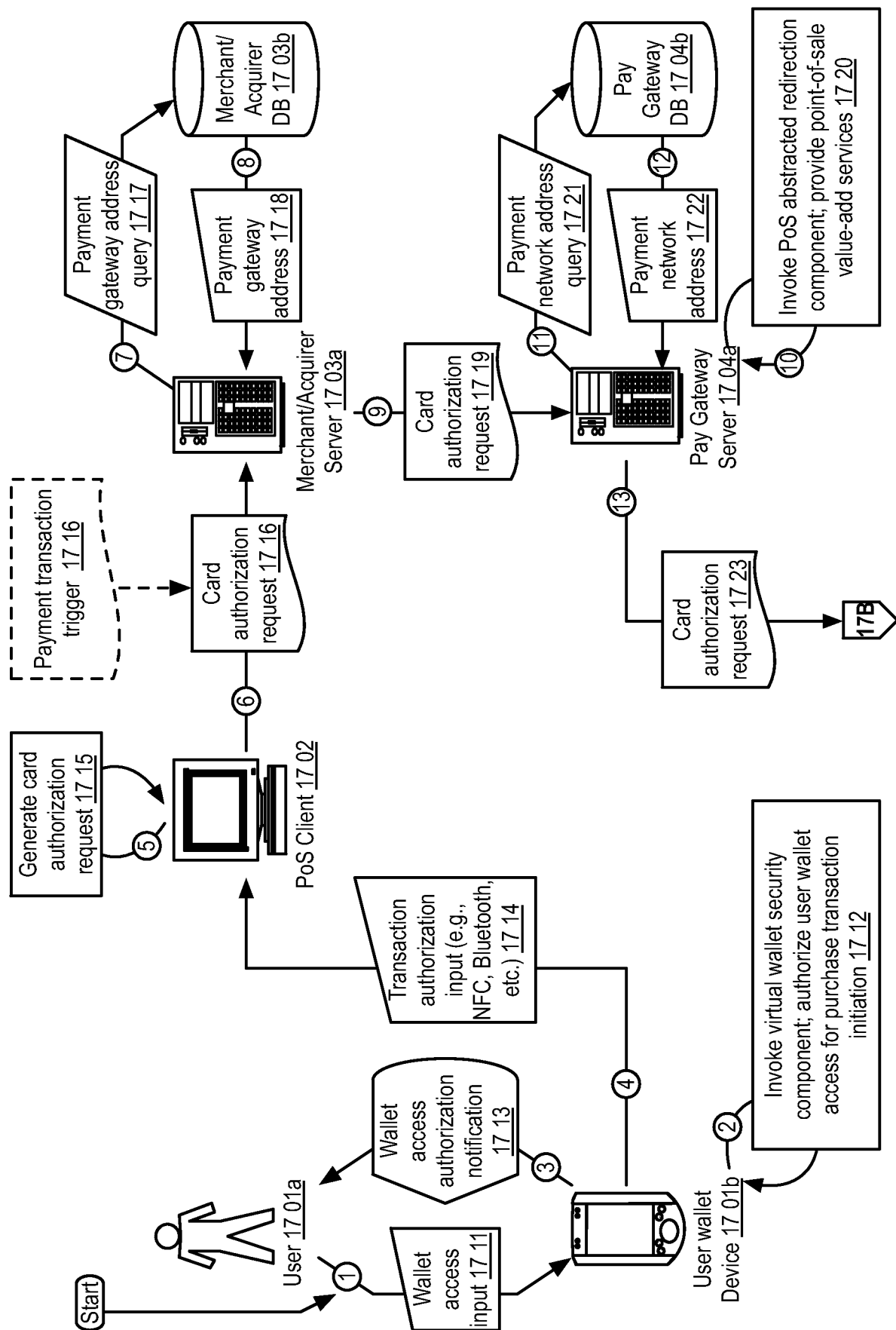
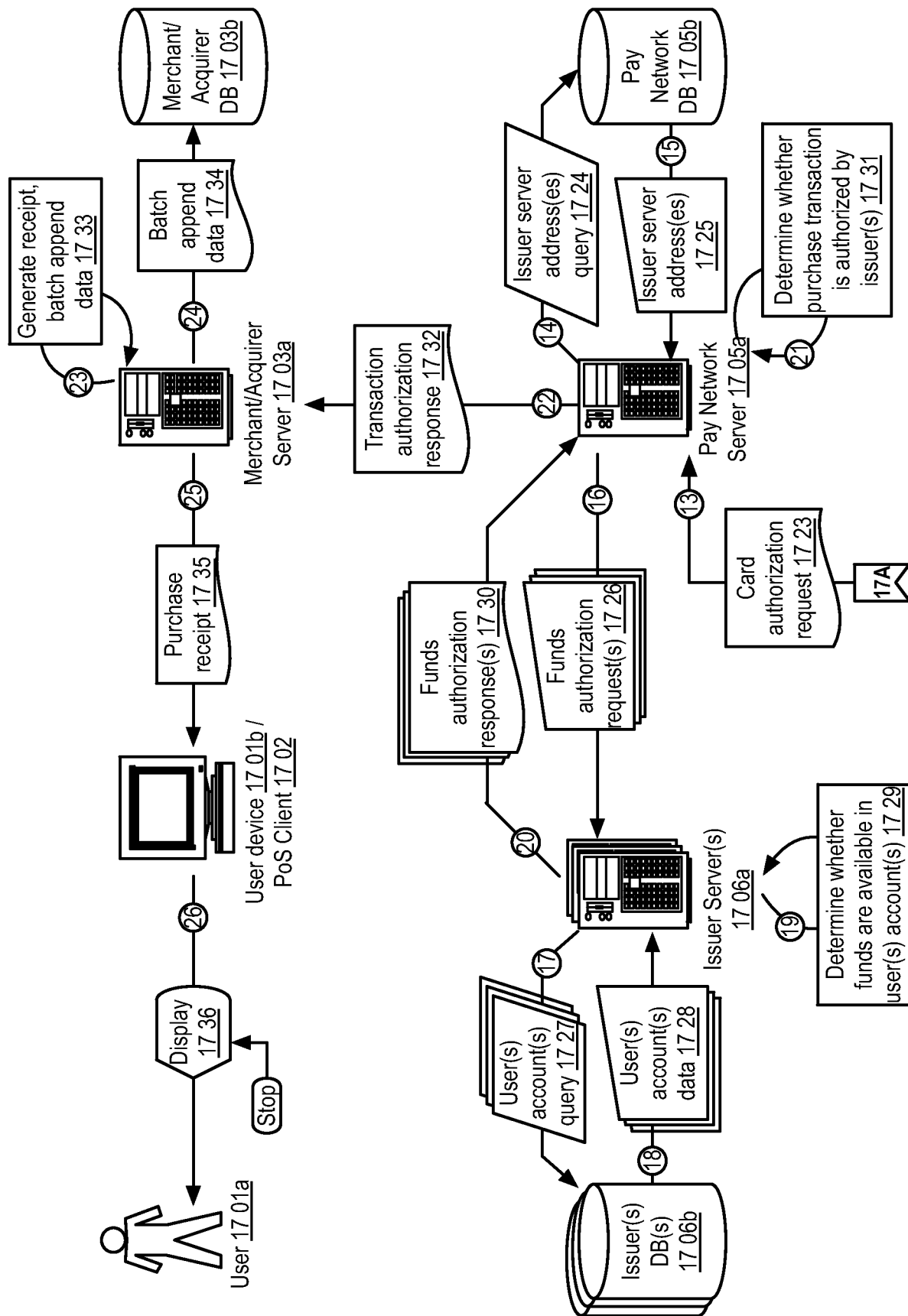


Figure 17A

Example Data Flow: Purchase Transaction Authorization



Example Data Flow: Purchase Transaction Authorization

Figure 17B

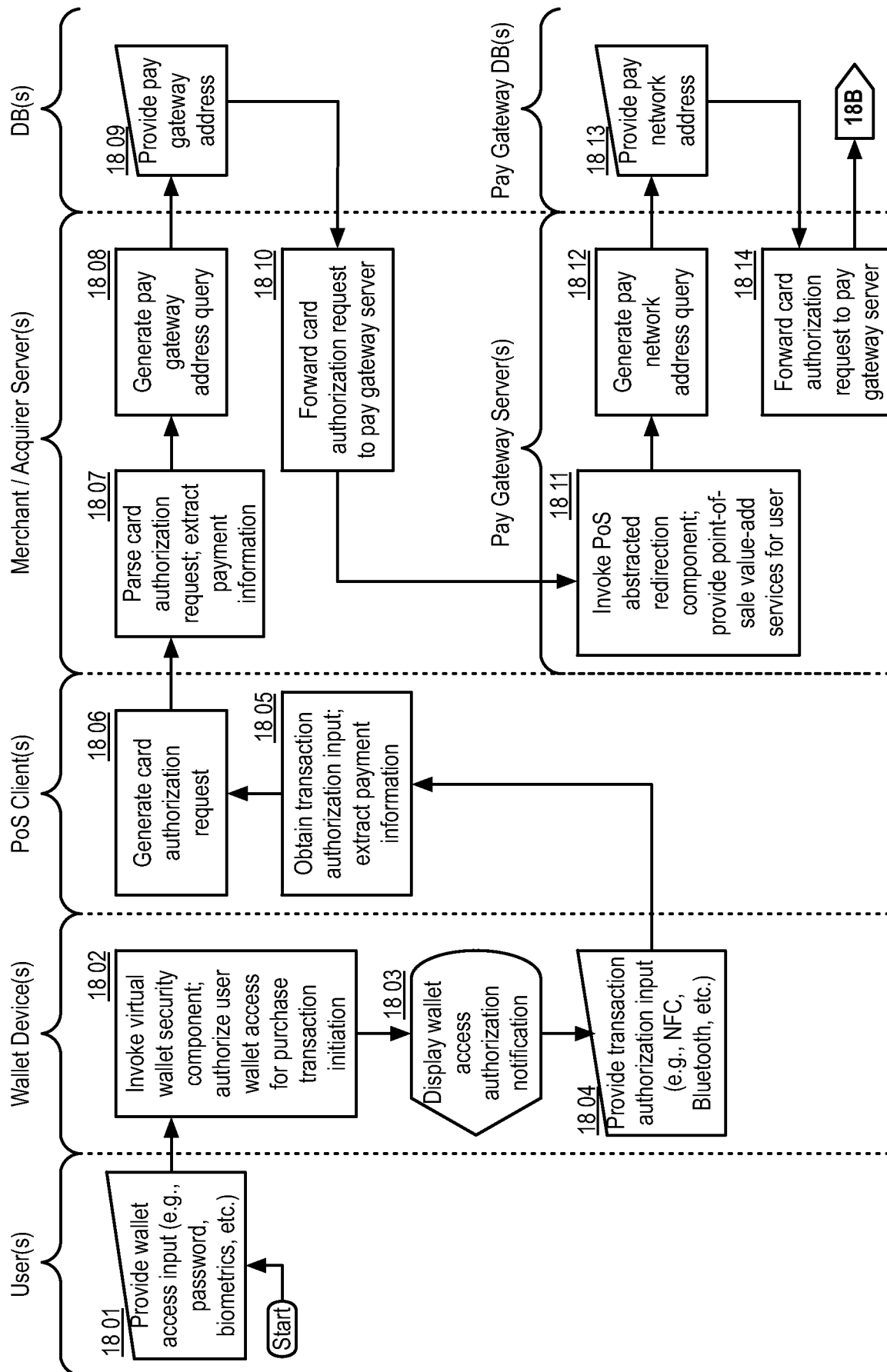


Figure 18A

Example: Purchase Transaction Authorization ("PTA") component

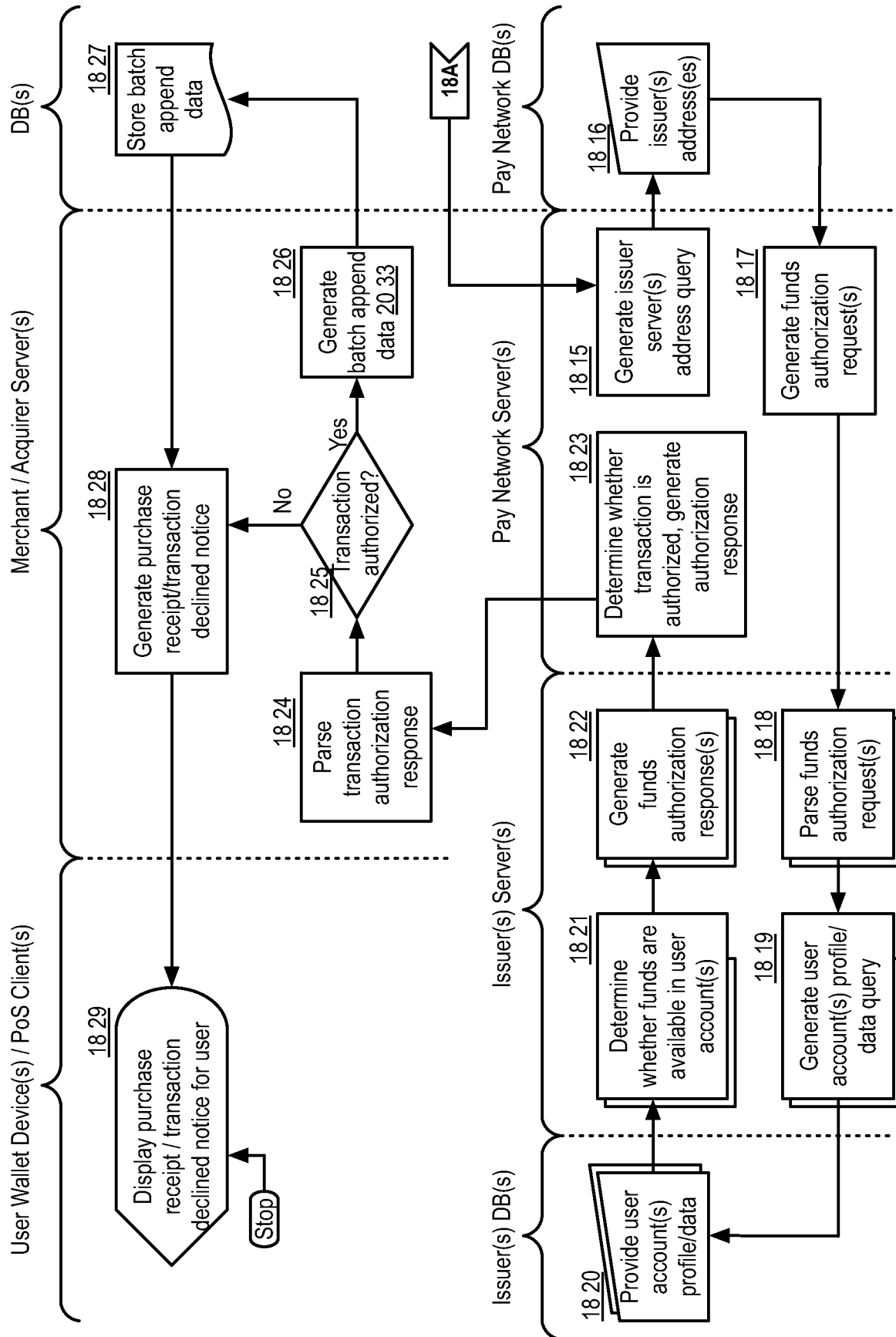


Figure 18B

Example: Purchase Transaction Authorization ("PTA") component

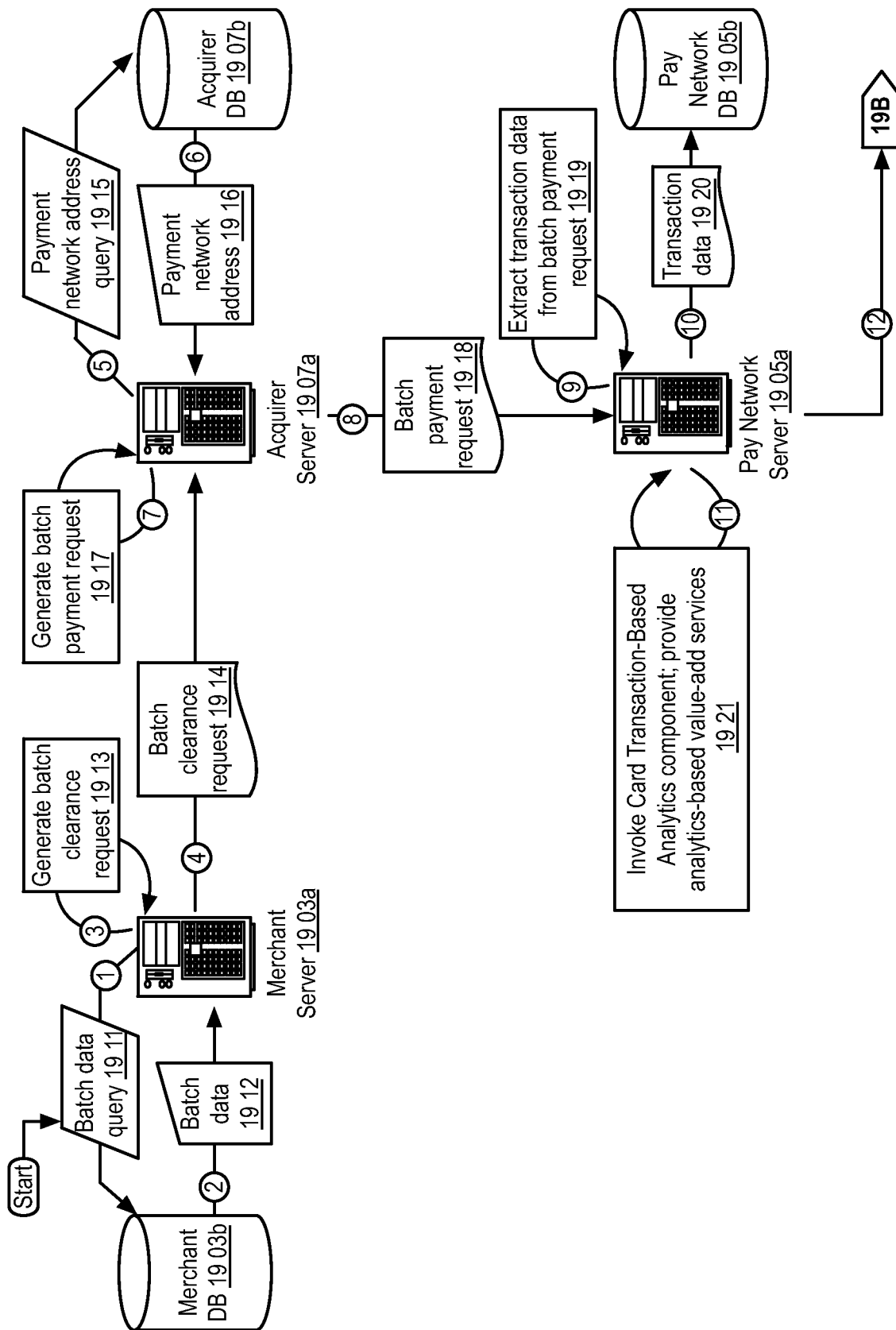
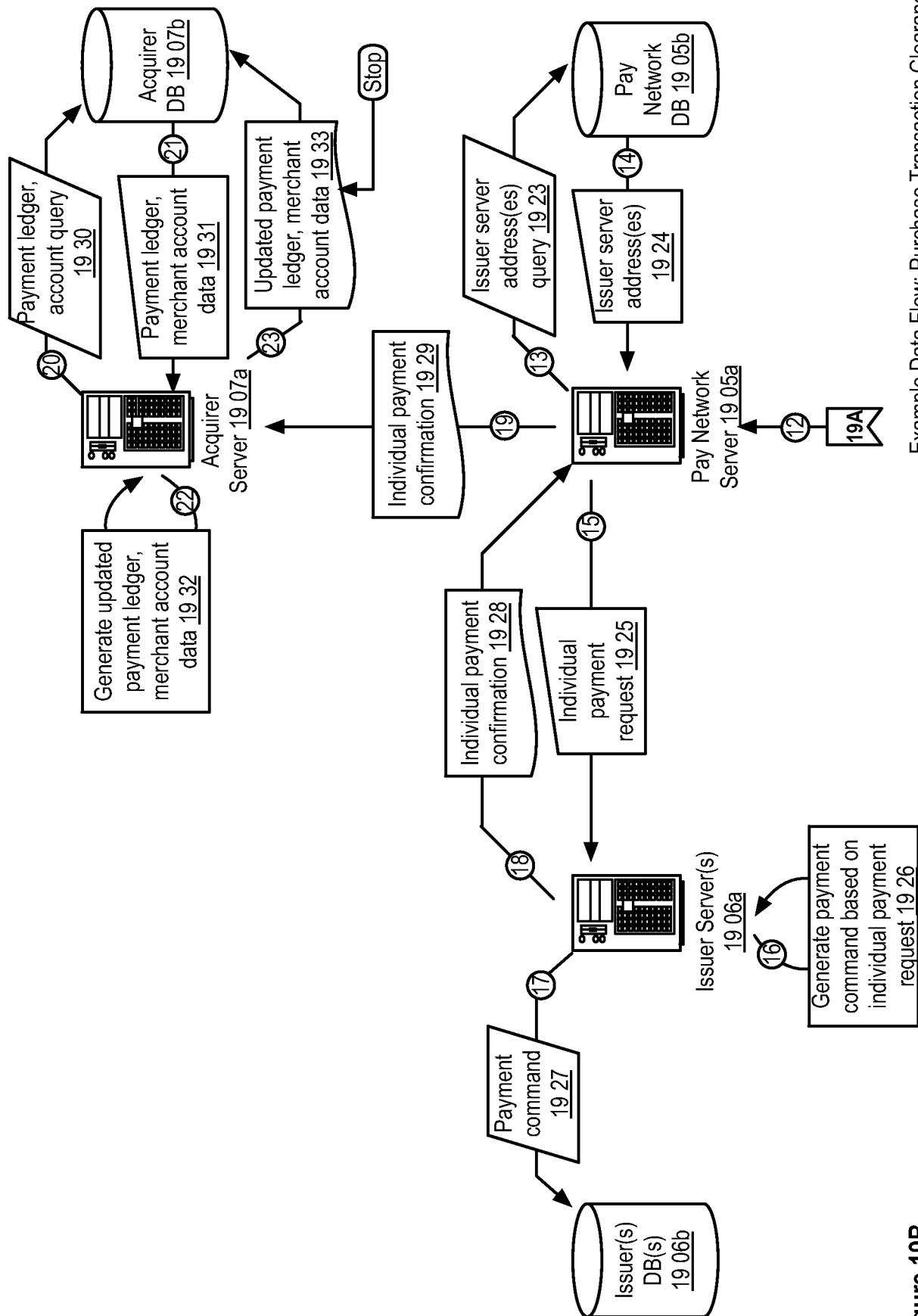


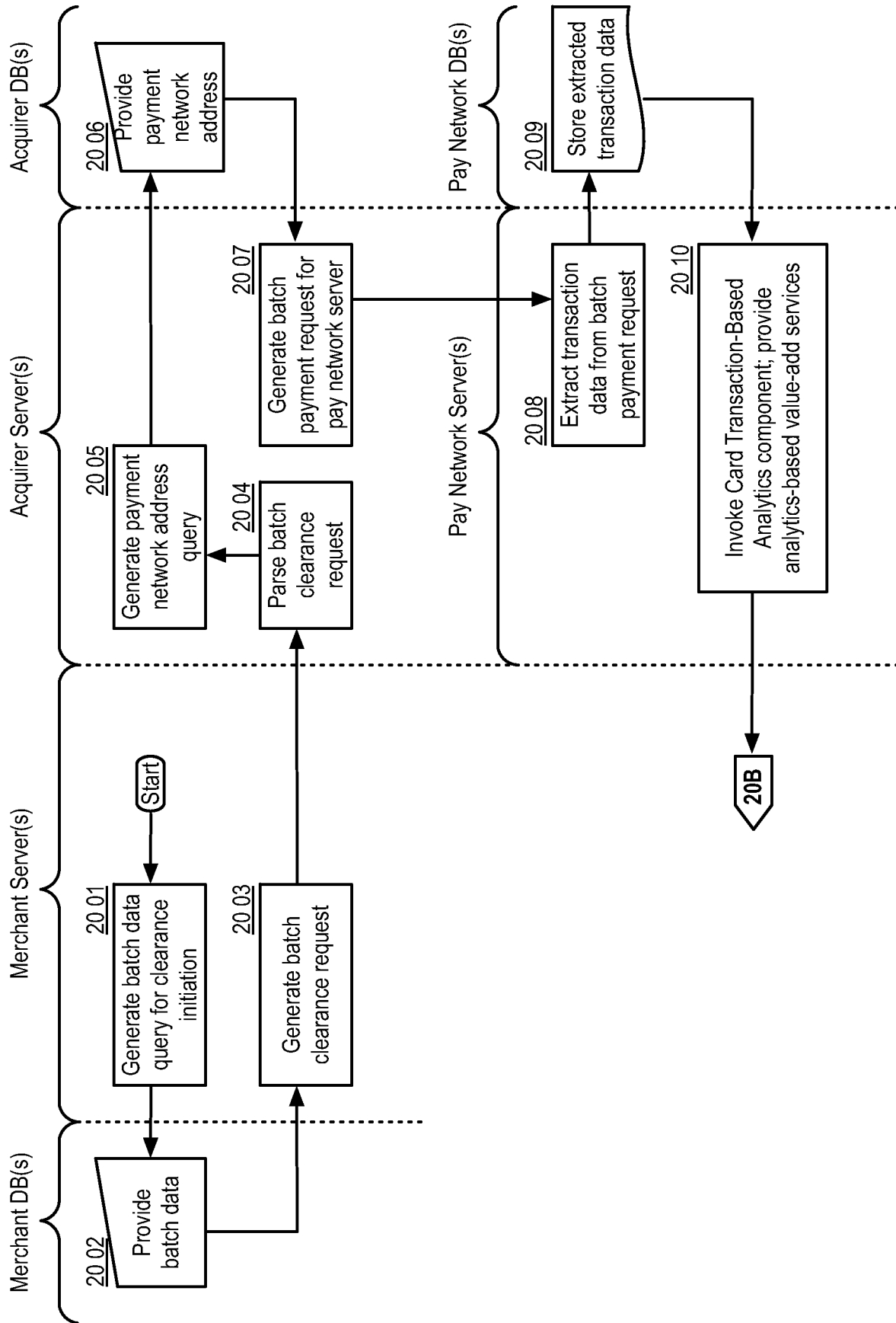
Figure 19A

Example Data Flow: Purchase Transaction Clearance



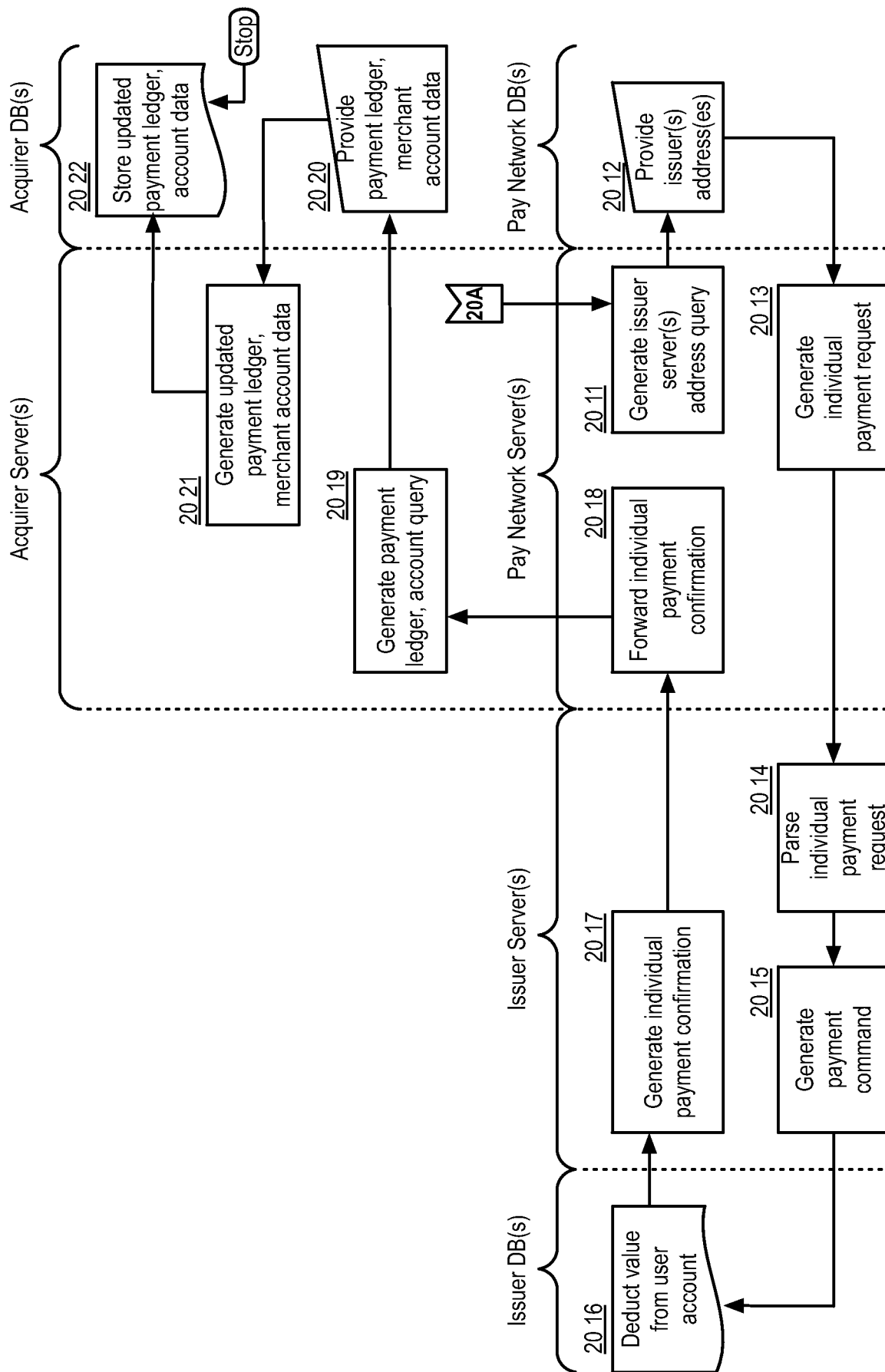
Example Data Flow: Purchase Transaction Clearance

Figure 19B

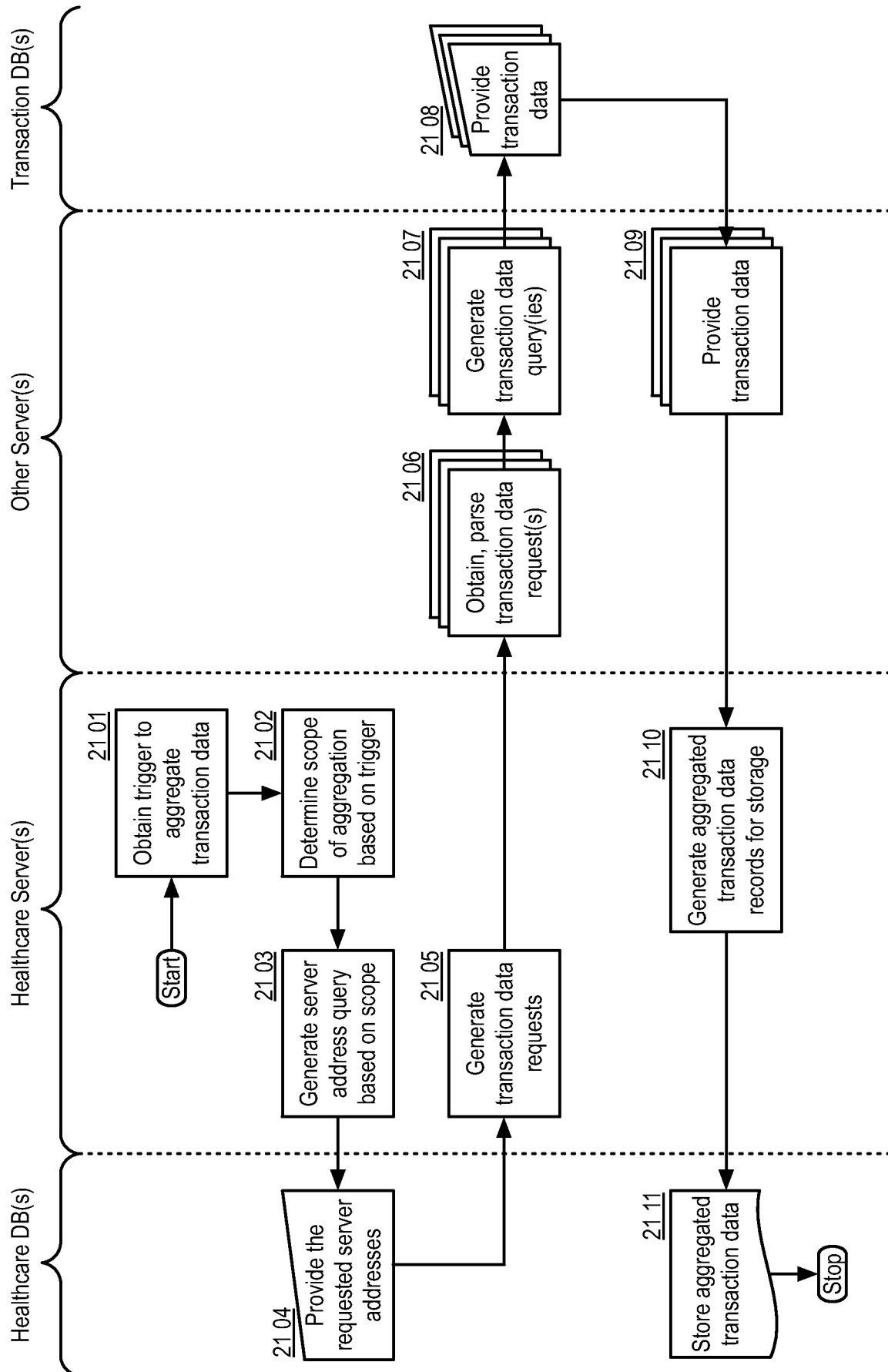


Example Logic Flow: Purchase Transaction Clearance ("PTC") component

Figure 20A



Example Logic Flow: Purchase Transaction Clearance ("PTC") component



Example Logic Flow: Transaction Data Aggregation ("TDA") component

Figure 21

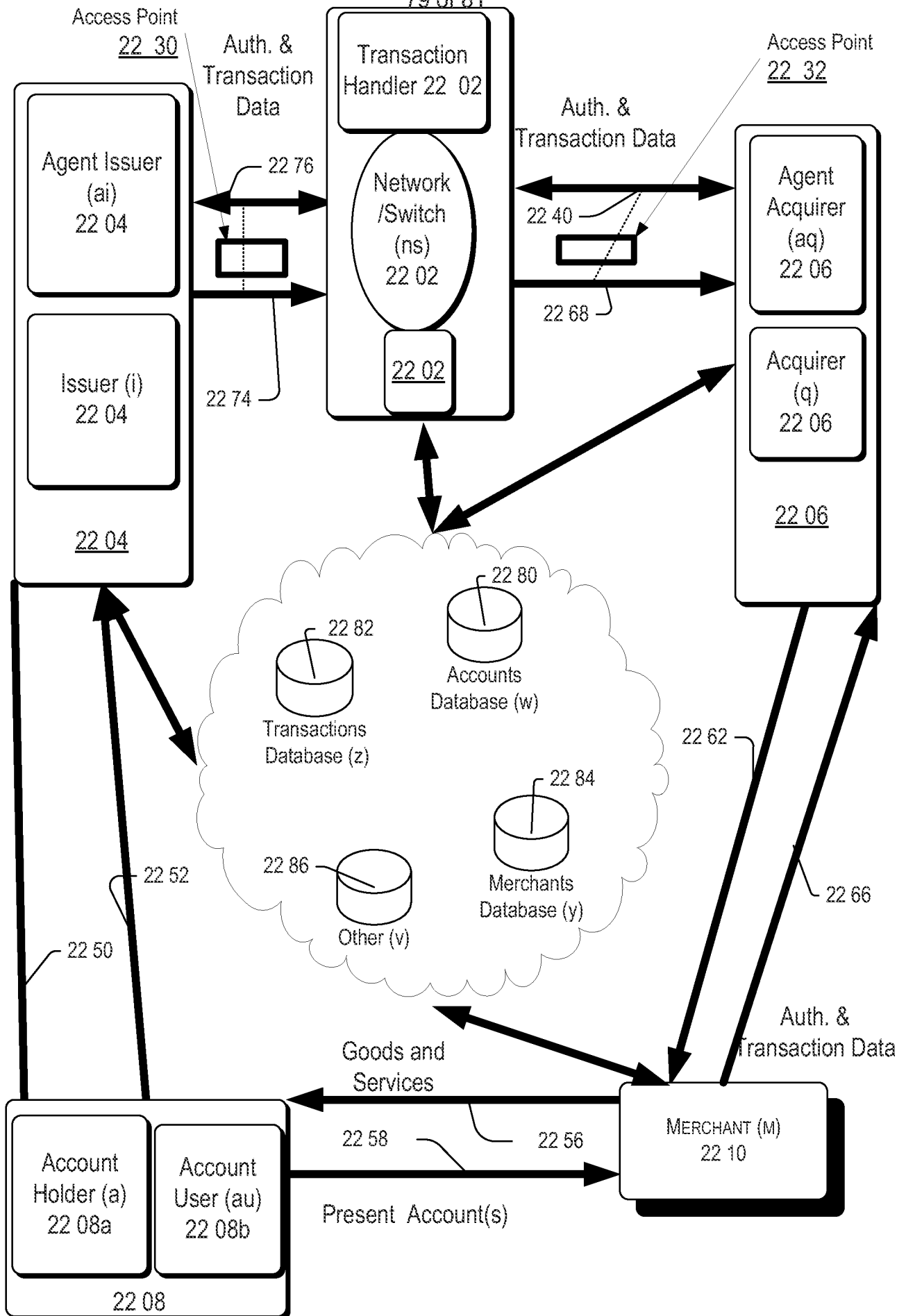


Figure 22

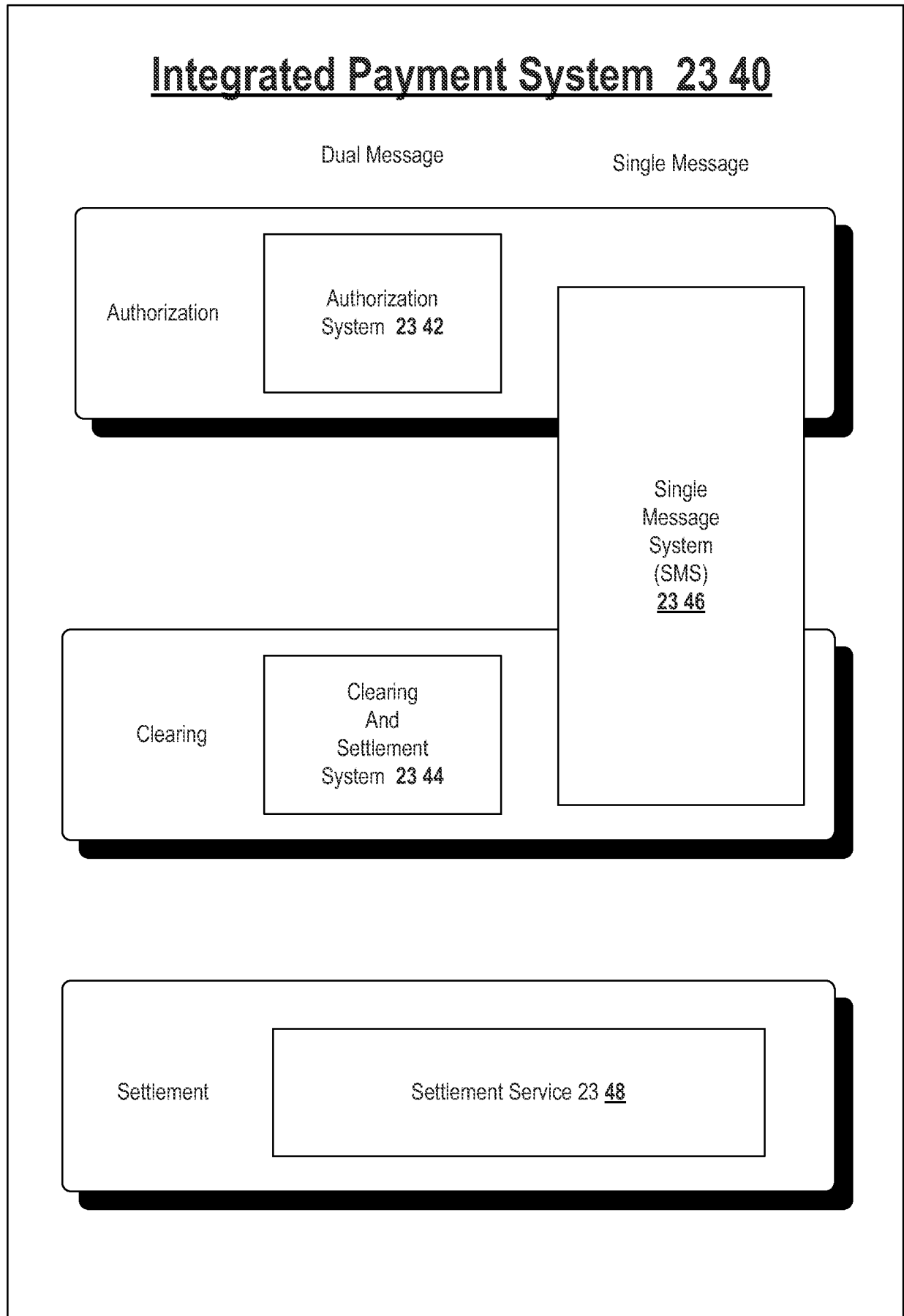


Figure 23

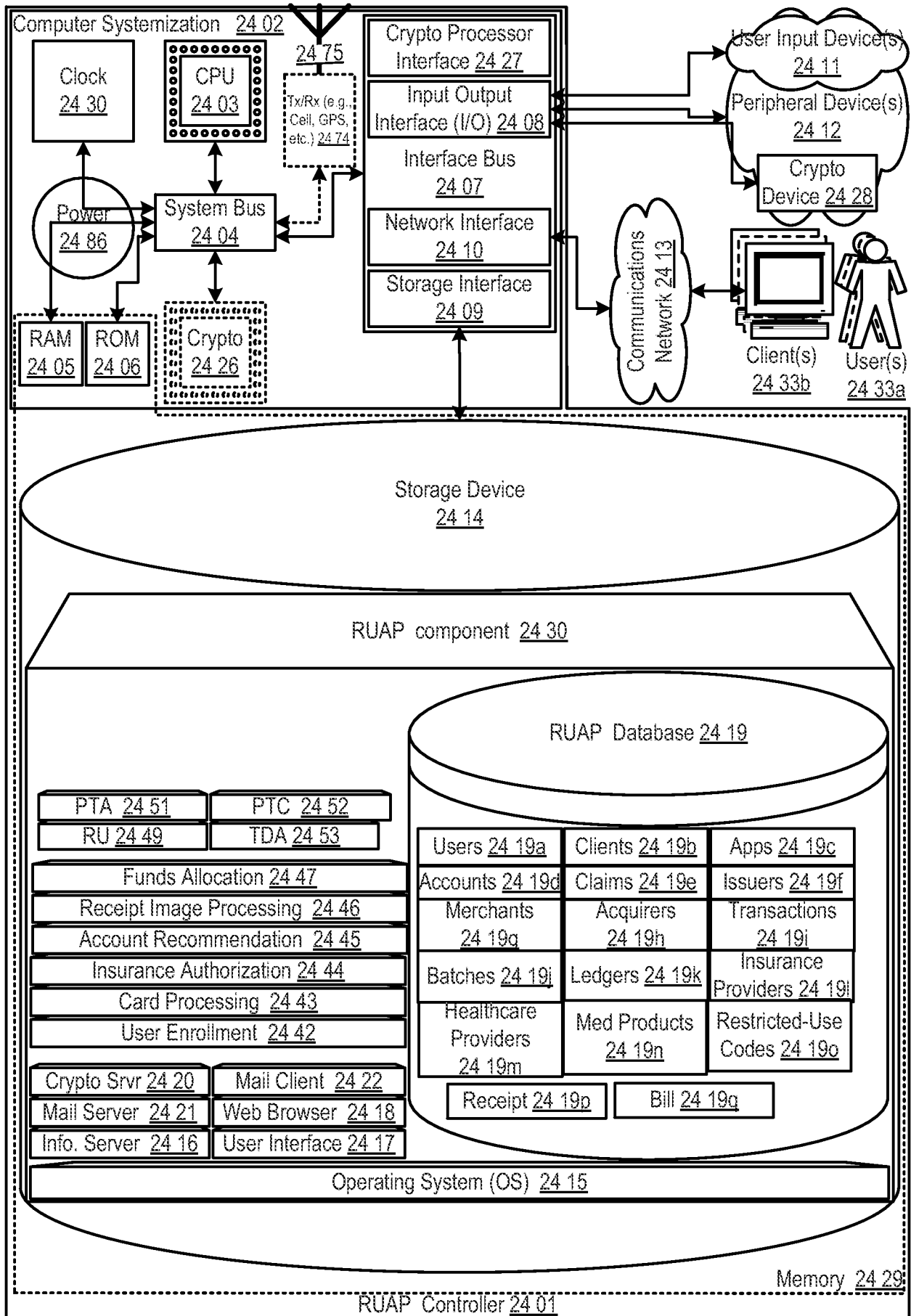


Figure 24

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US 12/31762

A. CLASSIFICATION OF SUBJECT MATTER

IPC(8) - G06Q 20/00 (2012.01)

USPC - 705/64

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC(8)- G06Q 20/00 (2012.01)

USPC - 705/64

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

USPC - 705/77; 705/26.1; 705/39; 713/150 (See Keywords Below)

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

PubWEST (PGPB,USPT,USOC,EPAB,JPAB), Google Scholar; thomson Innovation

Search terms: reimburse, refund, request, claim, application, processing, limited, restricted, benefit, account, card, employment benefit, Medicaid, Medicare, health saving account, HSA, flexible spend account, FSA, employee benefit account, food stamp, food voucher....

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2007/0233525 A1 (BOYLE), 04 October 2007 (04.10.2007), entire document, especially Abstract; para [0006]-[0009], [0011], [0028], [0032]-[0034], [0041]-[0042]	1-90
Y	US 2010/0100484 A1 (NGUYEN et al.), 22 April 2010 (22.04.2010), entire document, especially Abstract; para [0027]-[0029], [0034]-[0035], [0037]-[0040]	1-93
Y	US 2009/0018923 A1 (CHEN et al.), 15 January 2009 (15.01.2009), entire document, especially Abstract; para [0013], [0032]-[0036], [0059]-[0063], [0076]-[0078]	23-26, 53-56, 83-86 and 91-96
Y	US 2006/0122855 A1 (PROROCK), 08 June 2006 (08.06.2006), entire document, especially Abstract; para [0023], [0028]-[0032], [0037], [0047]-[0049]	94-96
Y	US 2007/0271120 A1 (SURESH et al.), 22 November 2007 (22.11.2007), entire document, especially Abstract; para [0021]-[0024], [0045], [0061], [0068]	9, 11, 39, 41, 69 and 71
Y	US 2010/0057611 A1 (ANTOO et al.), 04 March 2010 (04.03.2010), entire document, especially Abstract; para [0022]-[0024], [0029], [0035]	10, 40 and 70
A	US 2010/0145810 A1 (POURFALLAH et al.), 10 June 2010 (10.06.2010), entire document	1-96
A	US 2010/0010909 A1 (MARSHALL et al.), 14 January 2010 (14.01.2010), entire document	1-96
A	US 7,464,859 B1 (HAWKINS), 16 December 2008 (16.12.2008), entire document	1-96
A	US 2008/0046347 A1 (SMITH et al.), 21 February 2008 (21.02.2008), entire document	1-96



Further documents are listed in the continuation of Box C.



* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

03 July 2012 (03.07.2012)

Date of mailing of the international search report

16 JUL 2012

Name and mailing address of the ISA/US

Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, Virginia 22313-1450

Facsimile No. 571-273-3201

Authorized officer:

Lee W. Young

PCT Helpdesk: 571-272-4300
PCT OSP: 571-272-7774