US 20120209749A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2012/0209749 A1**

Hammad et al. (43) **Pub. Date: Aug. 16, 2012**

(54) **SNAP MOBILE PAYMENT APPARATUSES, METHODS AND SYSTEMS**

(76) Inventors: **Ayman Hammad**, Pleasanton, CA (US); **Igor Karpenko**, Sunnyvale, CA (US); **Miroslav Gavrilov**, Santa Clara, CA (US); **Abhinav Shrivastava**, Redmond, WA (US); **Mark Carlson**, Half Moon Bay, CA (US)

(21) Appl. No.: **13/398,817**

(22) Filed: **Feb. 16, 2012**

Related U.S. Application Data

(60) Provisional application No. 61/443,624, filed on Feb. 16, 2011, provisional application No. 61/512,248, filed on Jul. 27, 2011, provisional application No. 61/522,213, filed on Aug. 10, 2011, provisional application No. 61/527,576, filed on Aug. 25, 2011.

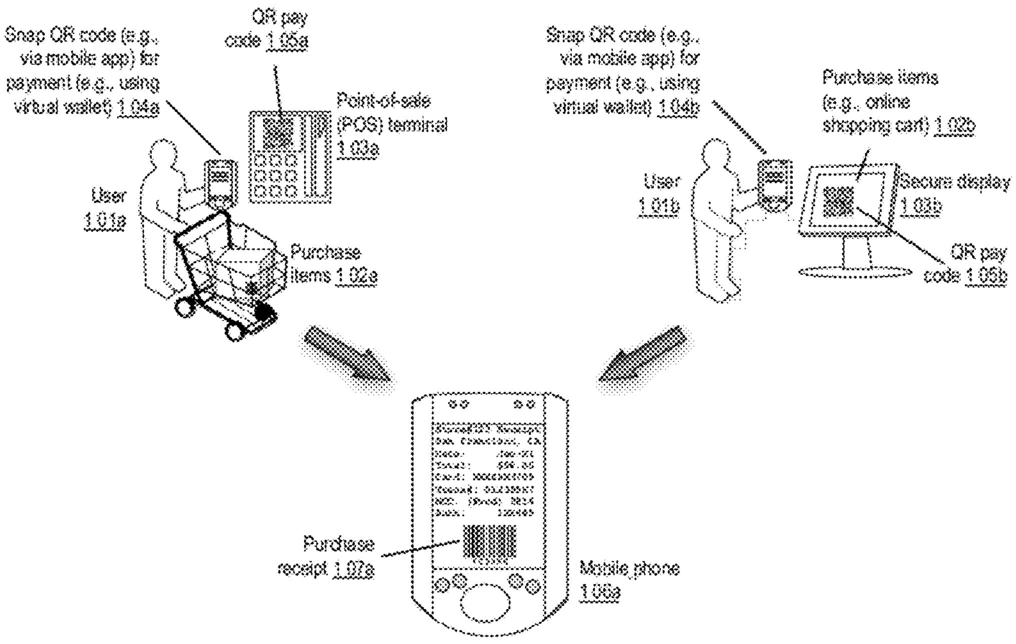
Publication Classification

(51) **Int. Cl.**
G06Q 30/06 (2012.01)

(52) **U.S. Cl.** **705/27.1**

(57) **ABSTRACT**

The SNAP MOBILE PAYMENT APPARATUSES, METHODS AND SYSTEMS ("SNAP") transform real-time-generated merchant-product Quick Response codes via SNAP components into virtual wallet card-based transaction purchase notifications. In one embodiment, the SNAP obtains a snapshot of a QR code presented on a display screen of a point-of-sale device from a mobile device. The SNAP decodes the QR code to obtain product information included in a checkout request of the user, and merchant information for processing a user purchase transaction with a merchant providing the QR code. The SNAP accesses a user virtual wallet to obtain user account information to process the user purchase transaction with the merchant. Using the product information, merchant information and user account information, the SNAP generates a card authorization request, and which the SNAP provides to a payment network for transaction processing. Also, the SNAP obtains a purchase receipt confirming processing of the user purchase transaction.


Snap QR code (e.g., via mobile app) for payment (e.g., using virtual wallet) 104a
QR pay code 105a
Point-of-sale (POS) terminal 103a
User 101a
Purchase items 102a
Snap QR code (e.g., via mobile app) for payment (e.g., using virtual wallet) 104b
Purchase items (e.g., online shopping cart) 102b
User 101b
Secure display 103b
QR pay code 105b
Purchase receipt 107a
Mobile phone 106a

Example(s): Snap Mobile Payment

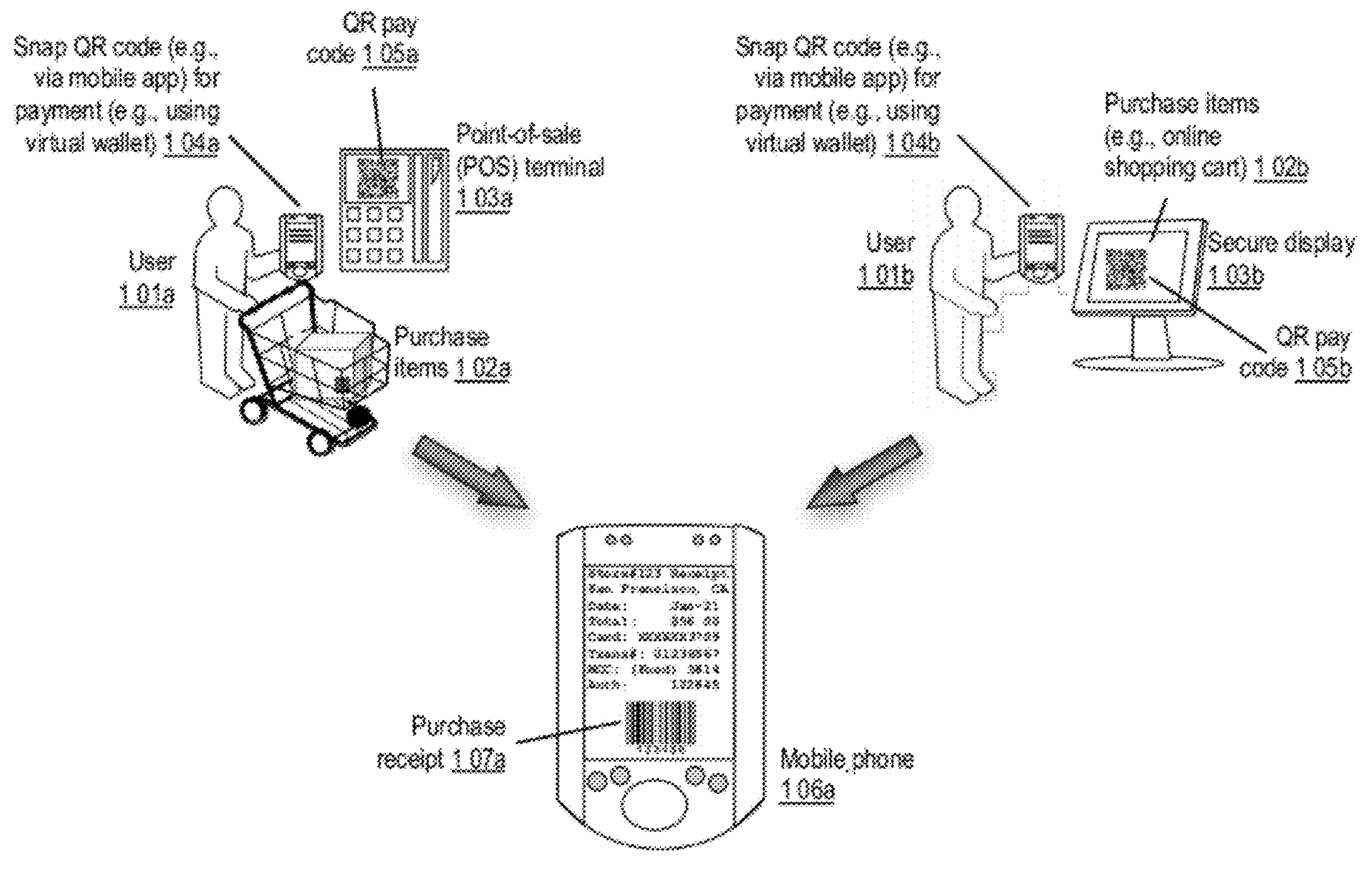
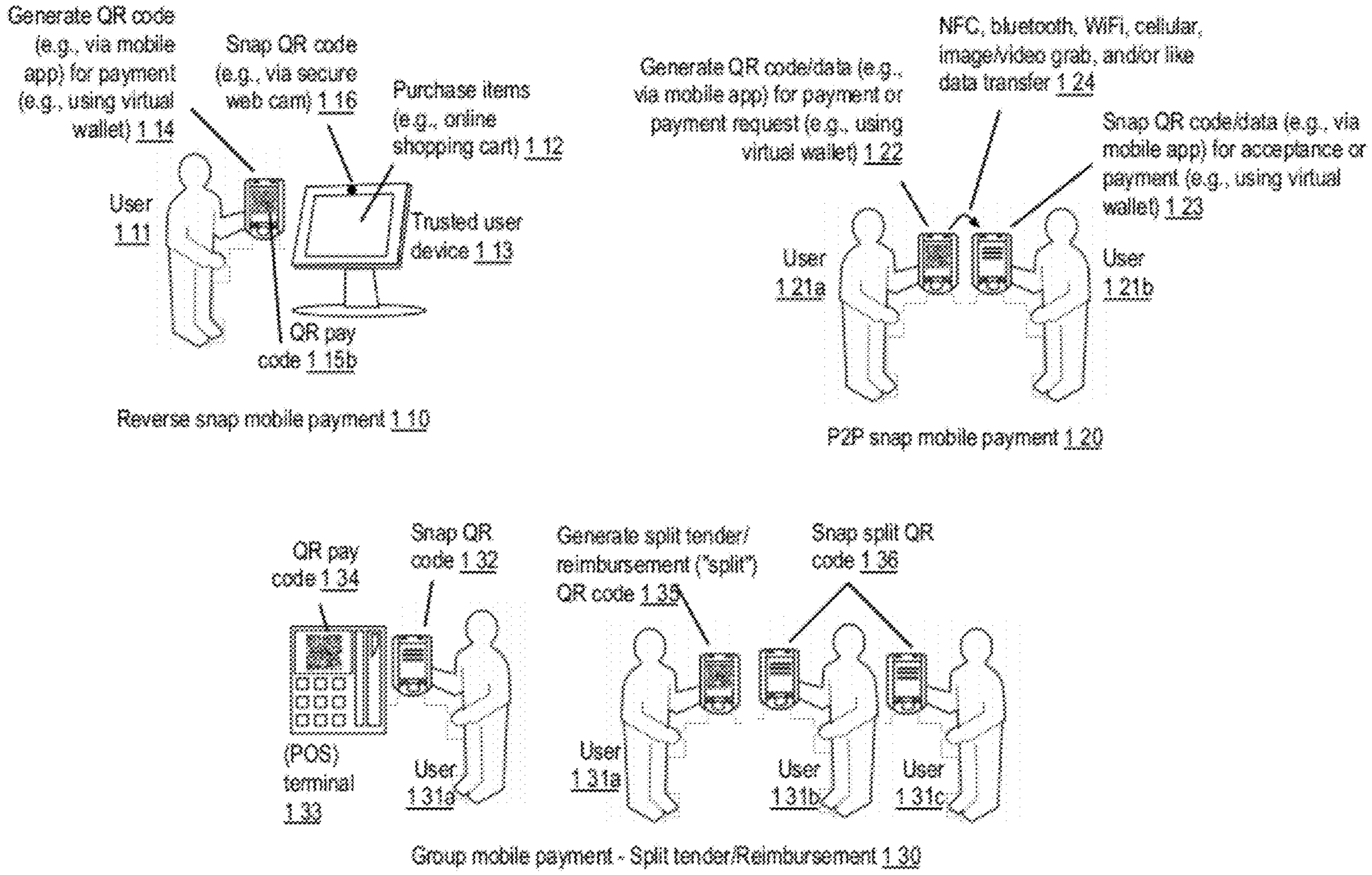


FIGURE 1A

Example(s): Snap Mobile Payment


Generate QR code (e.g., via mobile app) for payment (e.g., using virtual wallet) 114
Snap QR code (e.g., via secure web cam) 116
Purchase items (e.g., online shopping cart) 112
User 111
Trusted user device 113
QR pay code 115b
Generate QR code/data (e.g., via mobile app) for payment or payment request (e.g., using virtual wallet) 122
NFC, bluetooth, WiFi, cellular, image/video grab, and/or like data transfer 124
Snap QR code/data (e.g., via mobile app) for acceptance or payment (e.g., using virtual wallet) 123
User 121a
User 121b
QR pay code 134
Snap QR code 132
Generate split tender/ reimbursement ("split") QR code 135
Snap split QR code 136
(POS) terminal 133
User 131a
User 131a
User 131b
User 131c

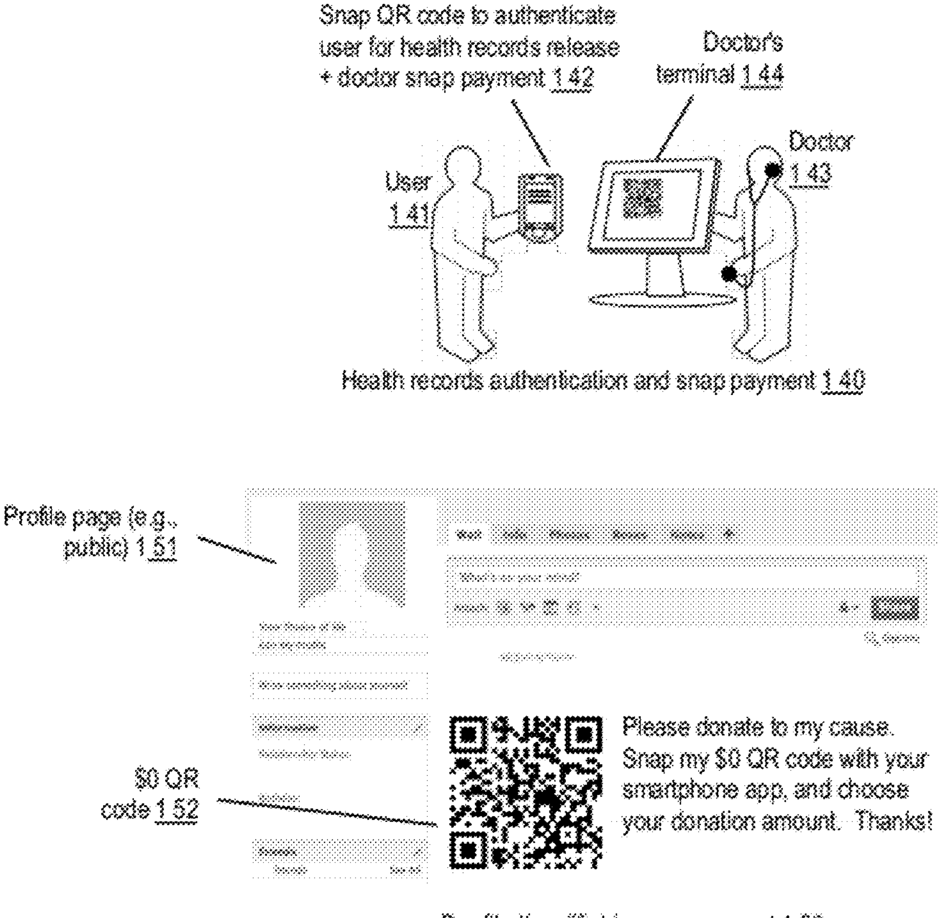
Reverse snap mobile payment 110

P2P snap mobile payment 120

Group mobile payment - Split tender/Reimbursement 130

FIGURE 1B

Example(s): Snap Mobile Payment


Snap QR code to authenticate user for health records release + doctor snap payment 142
Doctor's terminal 144
Doctor 143
User 141
Profile page (e.g., public) 151
$0 QR code 152

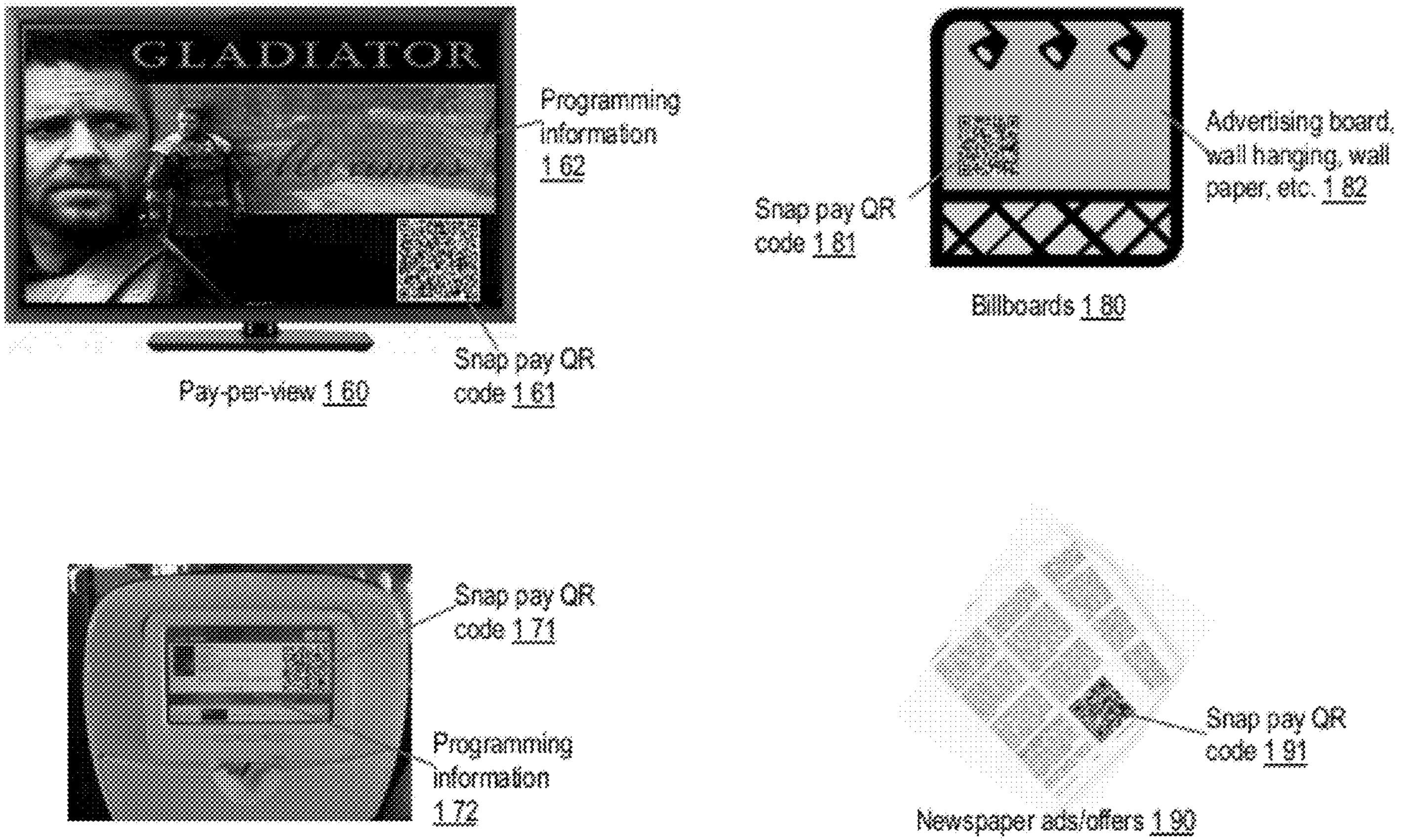
Health records authentication and snap payment 140

Please donate to my cause. Snap my $0 QR code with your smartphone app, and choose your donation amount. Thanks!

Pre-filed/modifiable snap payment 150

FIGURE 1C

Example(s): Snap Mobile Payment


GLADIATOR
Programming information 162
Snap pay QR code 161
Snap pay QR code 181
Advertising board, wall hanging, wall paper, etc. 182
Snap pay QR code 171
Programming information 172
Snap pay QR code 191

Pay-per-view 160

Billboards 180

In-flight entertainment 170

Newspaper ads/offers 190

FIGURE 1D

Example(s): Snap Mobile Applications

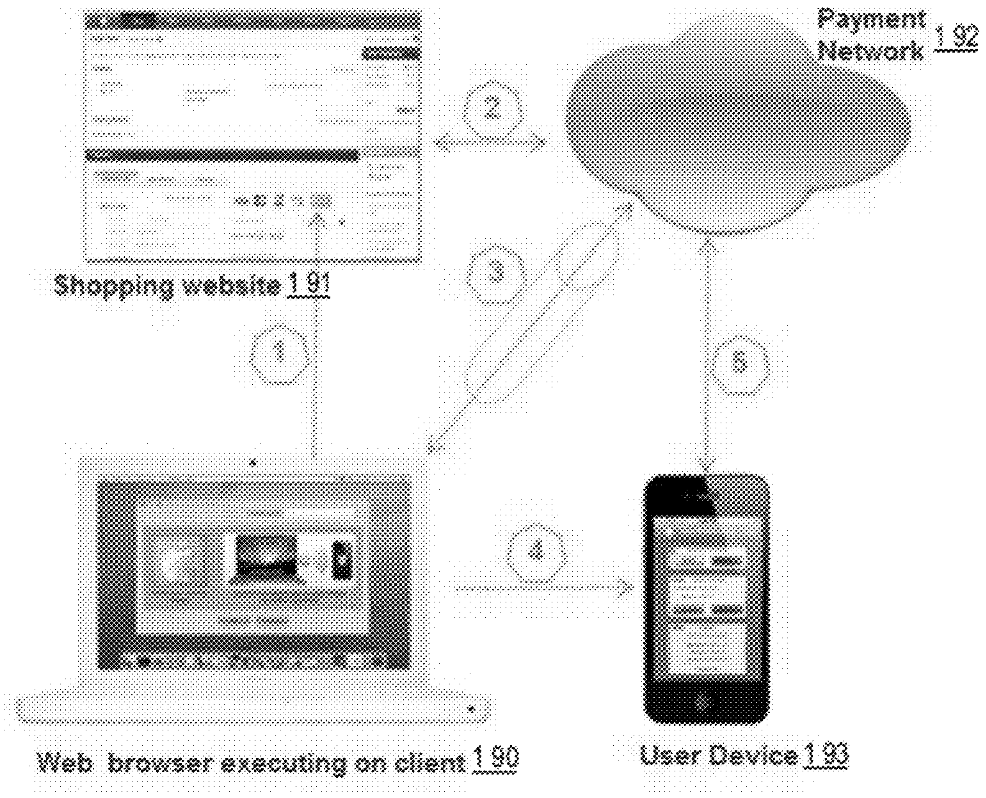
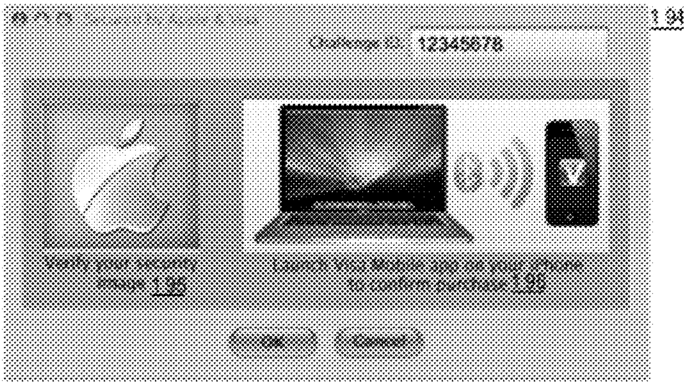


FIGURE 1E

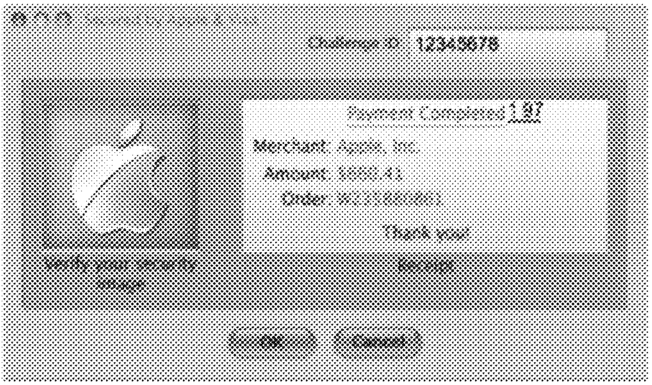
Example(s): Snap Mobile Applications



Challenge ID: 12345678

Verify your security image 1.95

Launch Visa Mobile app on your phone to confirm purchase 1.96



Challenge ID: 12345678

Verify your security image

Payment Completed 1.97

Merchant: Apple, Inc.

Amount: $800.41

Order: W23680861

Thank you!

Receipt

FIGURE 1F

Example(s): Snap Mobile Applications

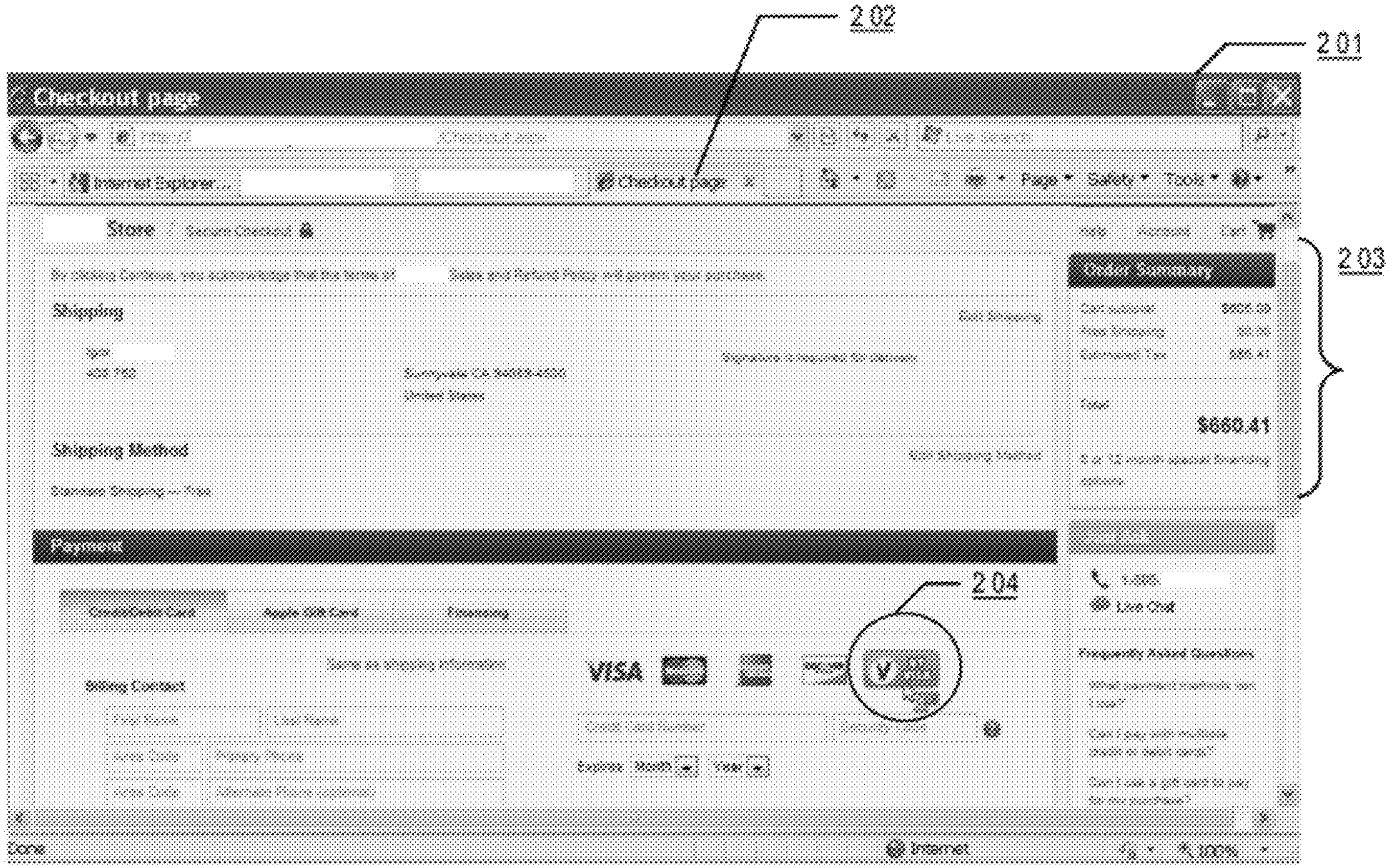
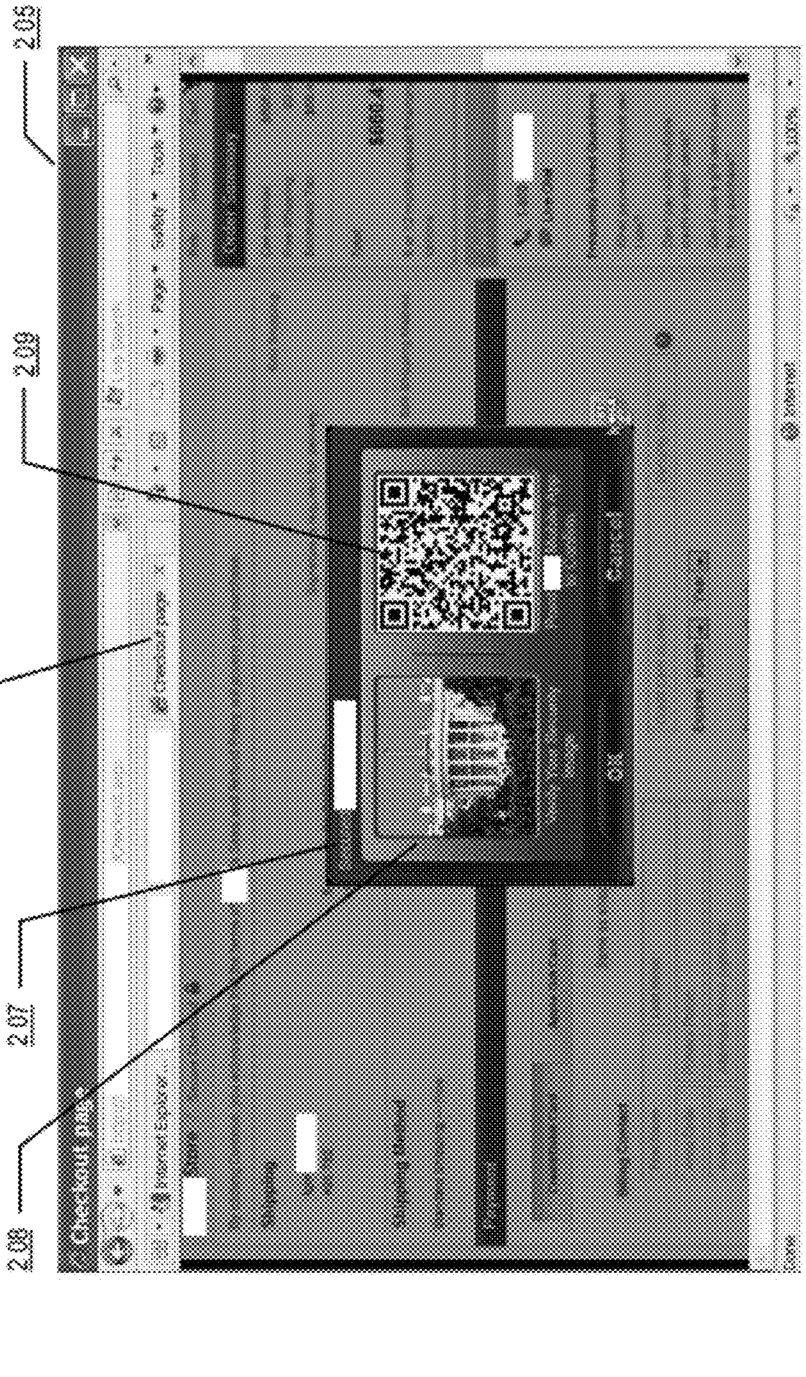


FIGURE 2A

Example: Snap Mobile Payment User Interface



Example: Snap Mobile Payment Web Interface

FIGURE 2B

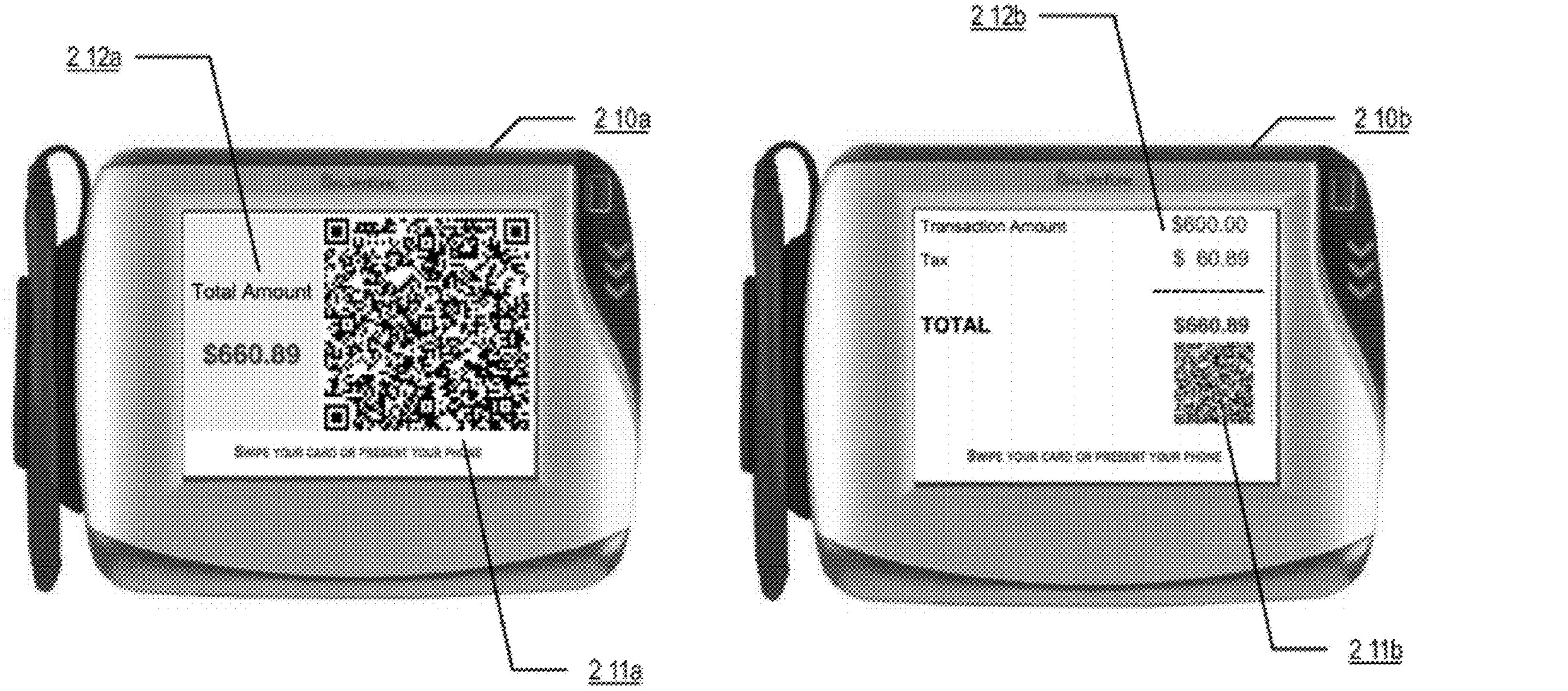


FIGURE 2C

Example: Snap Mobile Payment POS Terminal Interface

Example QR code data content 2.16b

```
<data>
    <merchant_id>AE783</merchant_id>
    <merchant_name>Acme, Inc.</merchant_name>
    <store_id>88234</store_id>
    <store_url>www.shop.acme.com</store_url>
    <terminal_id>userdevice1</terminal_id>
    <transaction_id>AFE1213344</transaction_id>
    <timestamp>2011-04-01:23:59:59</timestamp>
    <transaction_amount>$660.89</transaction_amount>
    <digital_sign>
        45e2085fa20496c91df574dc5652e145
    </digital_sign>
</data>
```

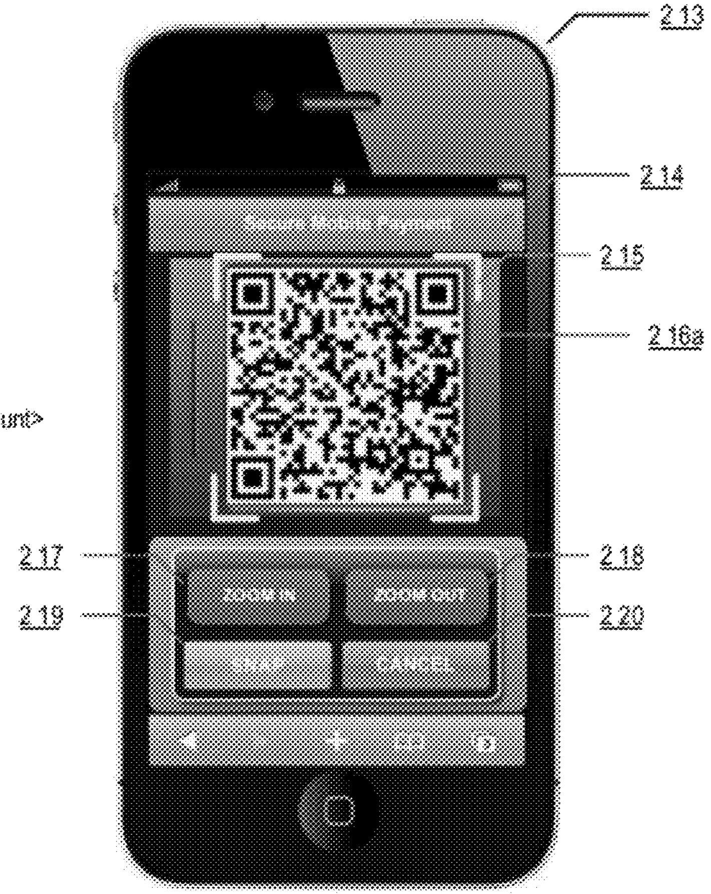
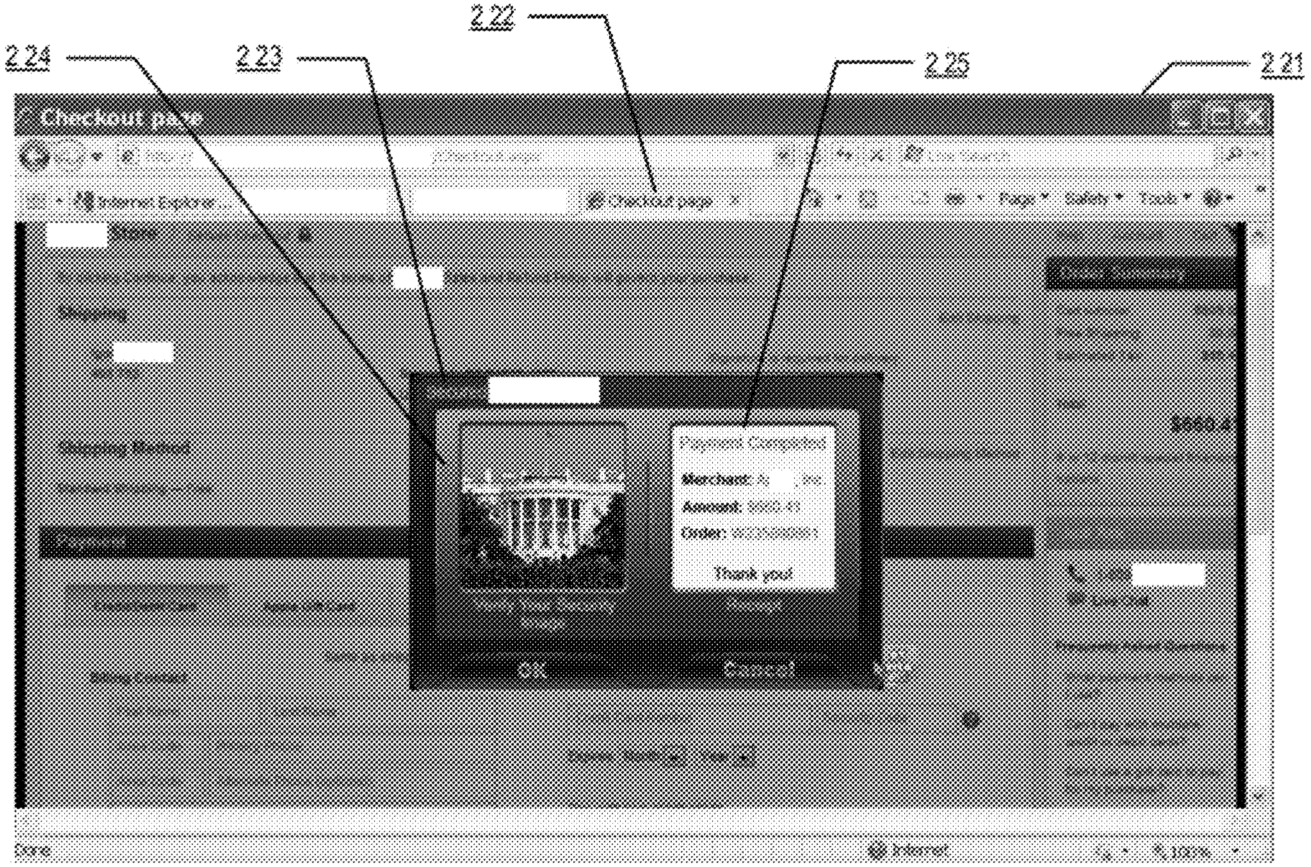


FIGURE 2D

Example: Snap Mobile App Payment User Interface



Payment Completed
Merchant:
Amount:
Order:
Thank you!

FIGURE 2E

Example: Snap Mobile App Payment Web Receipt

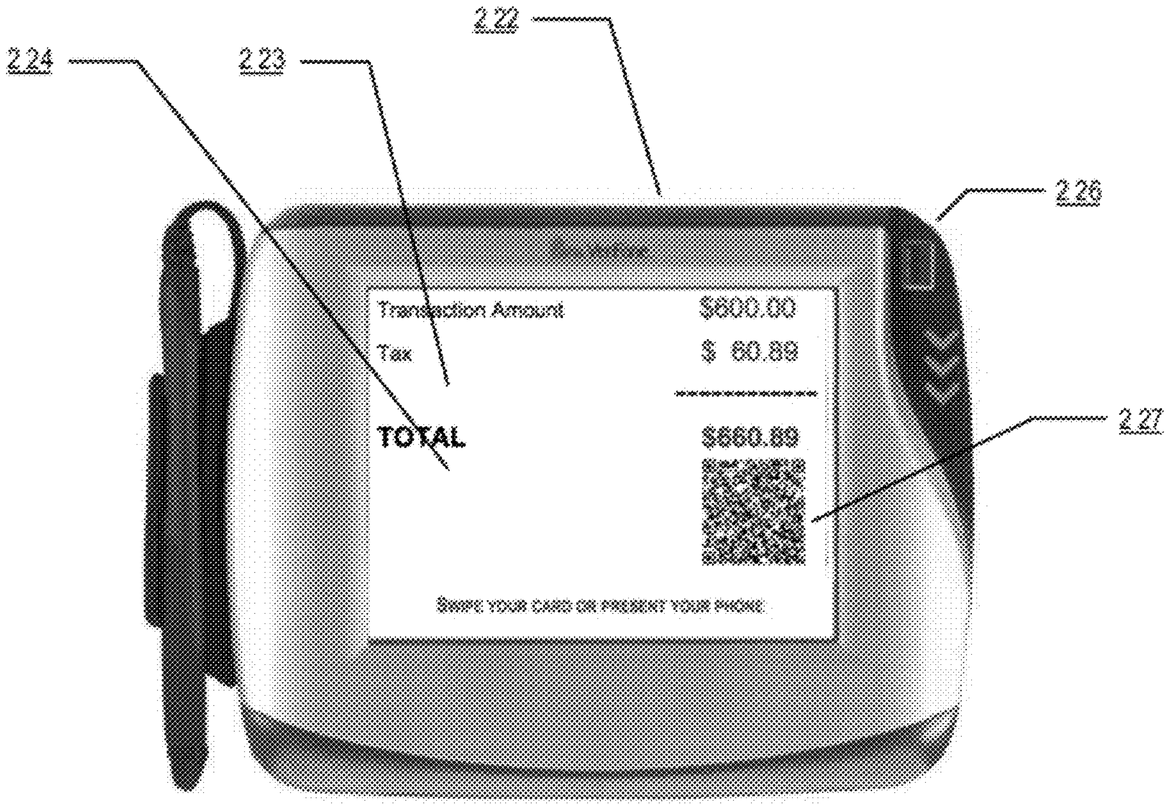
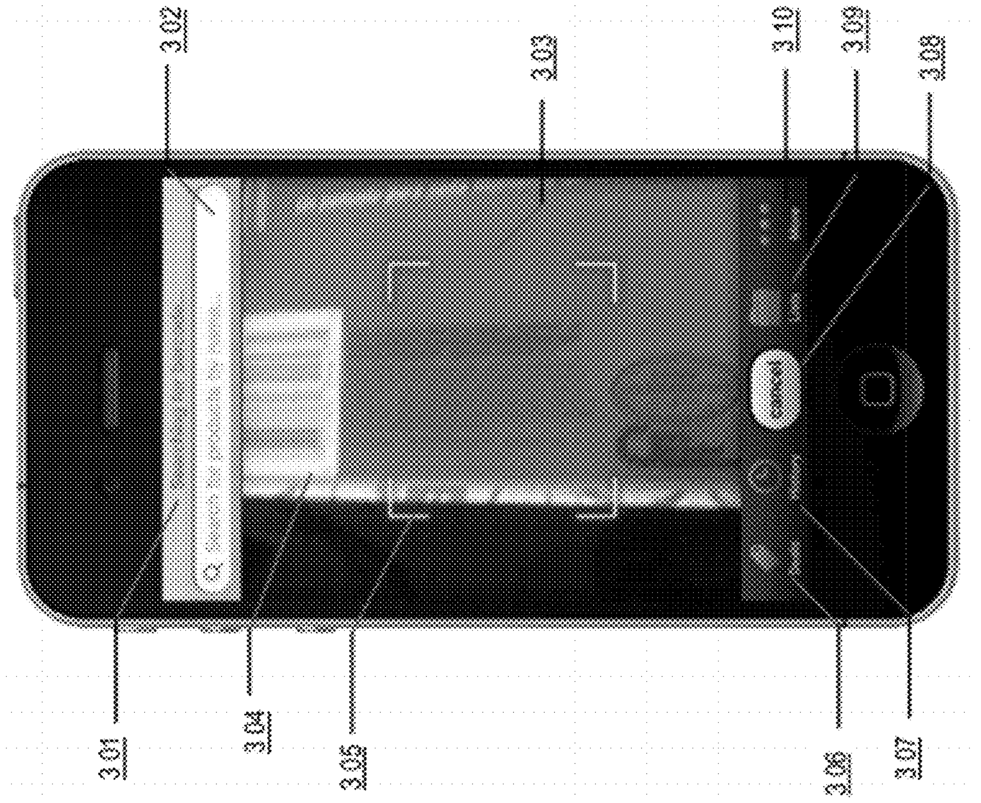

2.24
2.23
2.22
2.26
2.27
Transaction Amount $600.00
Tax $ 60.89
TOTAL $660.89
SWIPE YOUR CARD OR PRESENT YOUR PHONE

FIGURE 2F

Example: Snap Mobile App Payment Web Receipt



Example: One-Tap Mobile App: Barcode Capture

FIGURE 3A

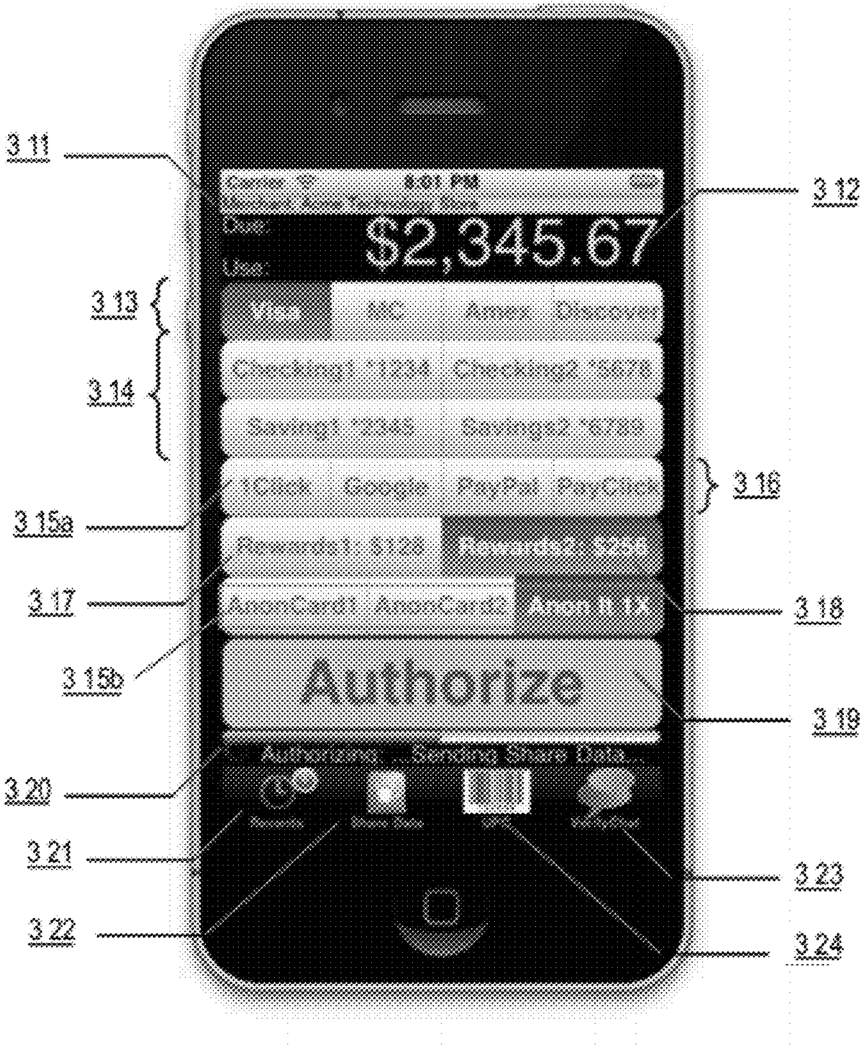
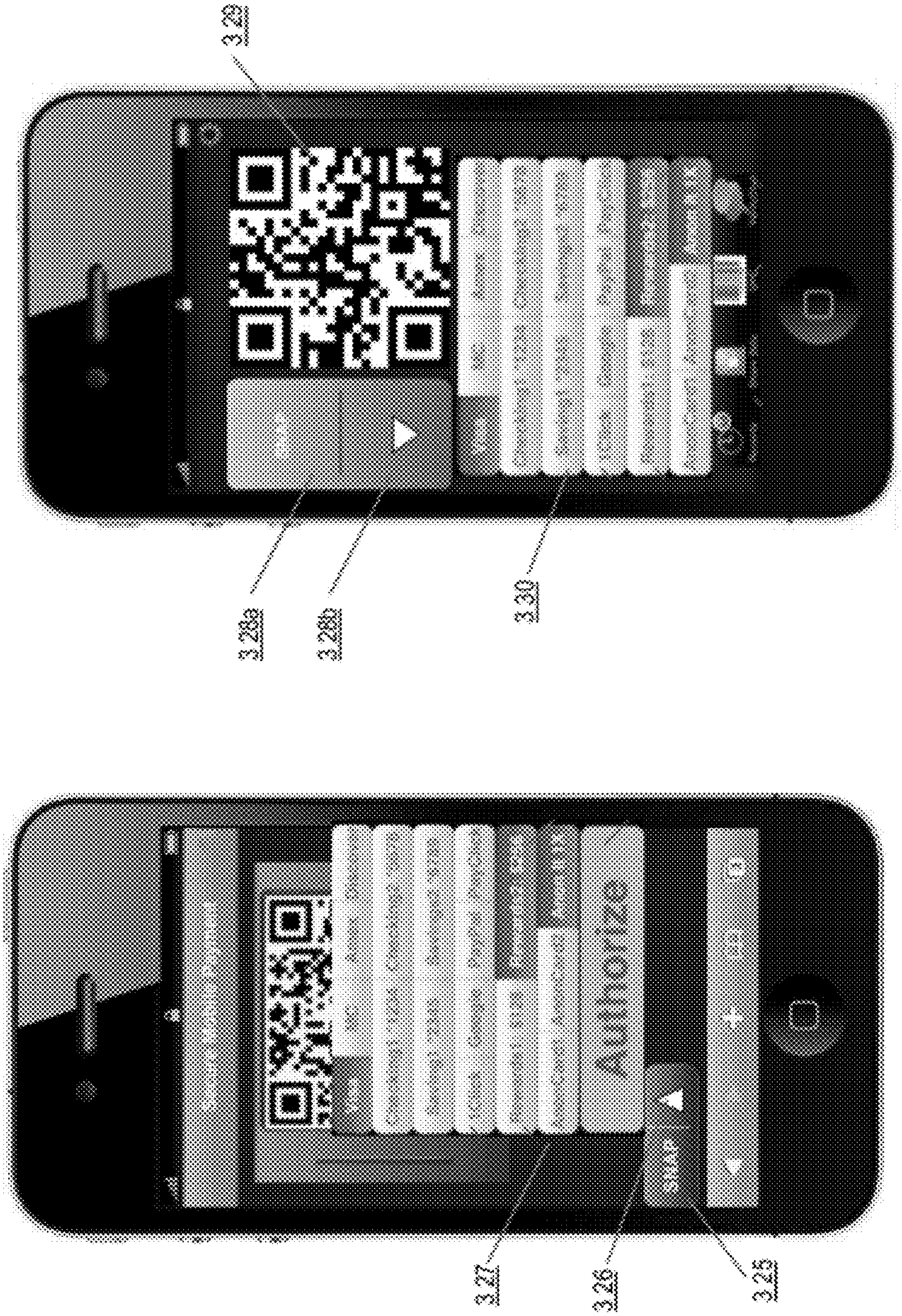


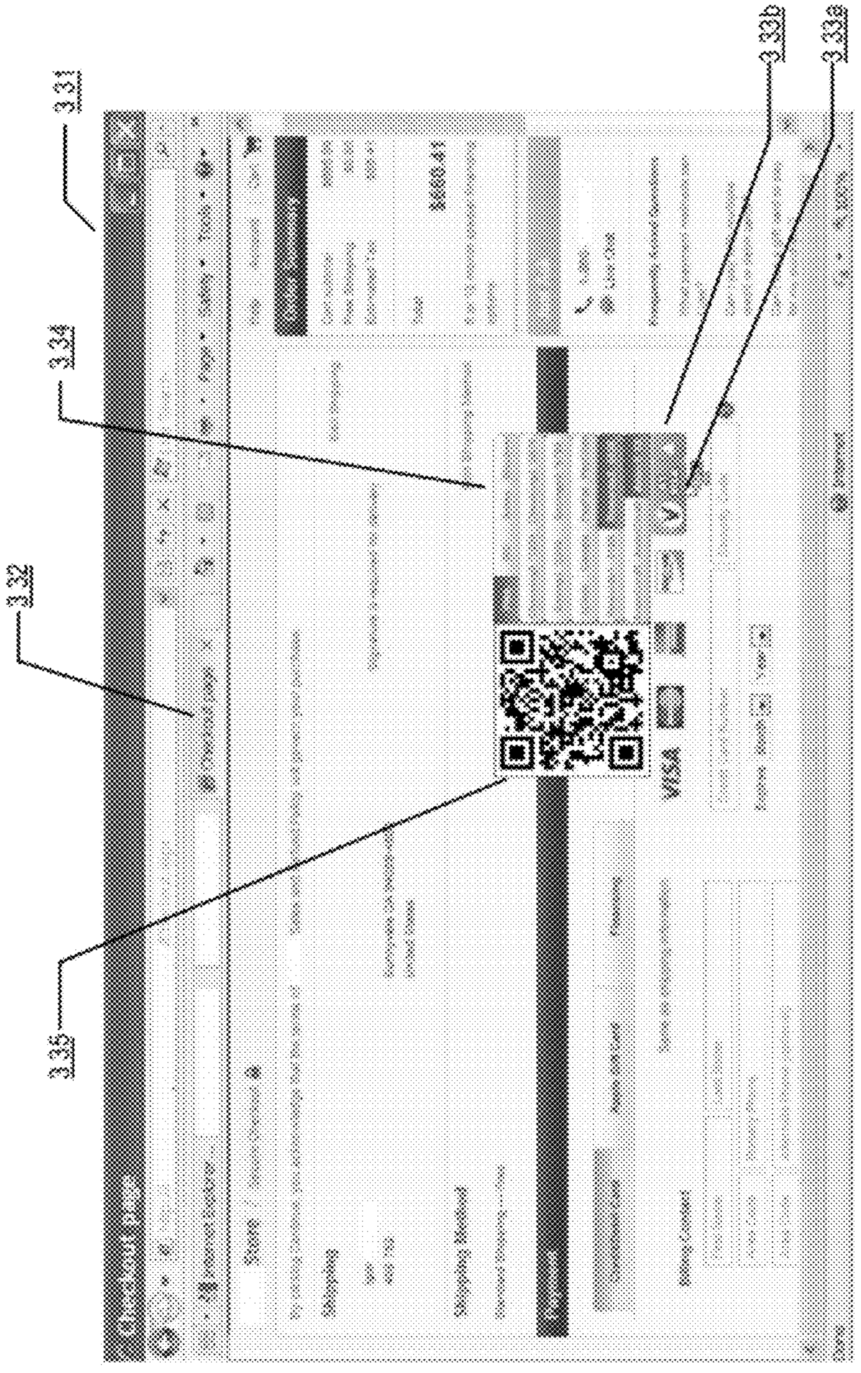
FIGURE 3B

Example: One-Tap Mobile App: Payment Options



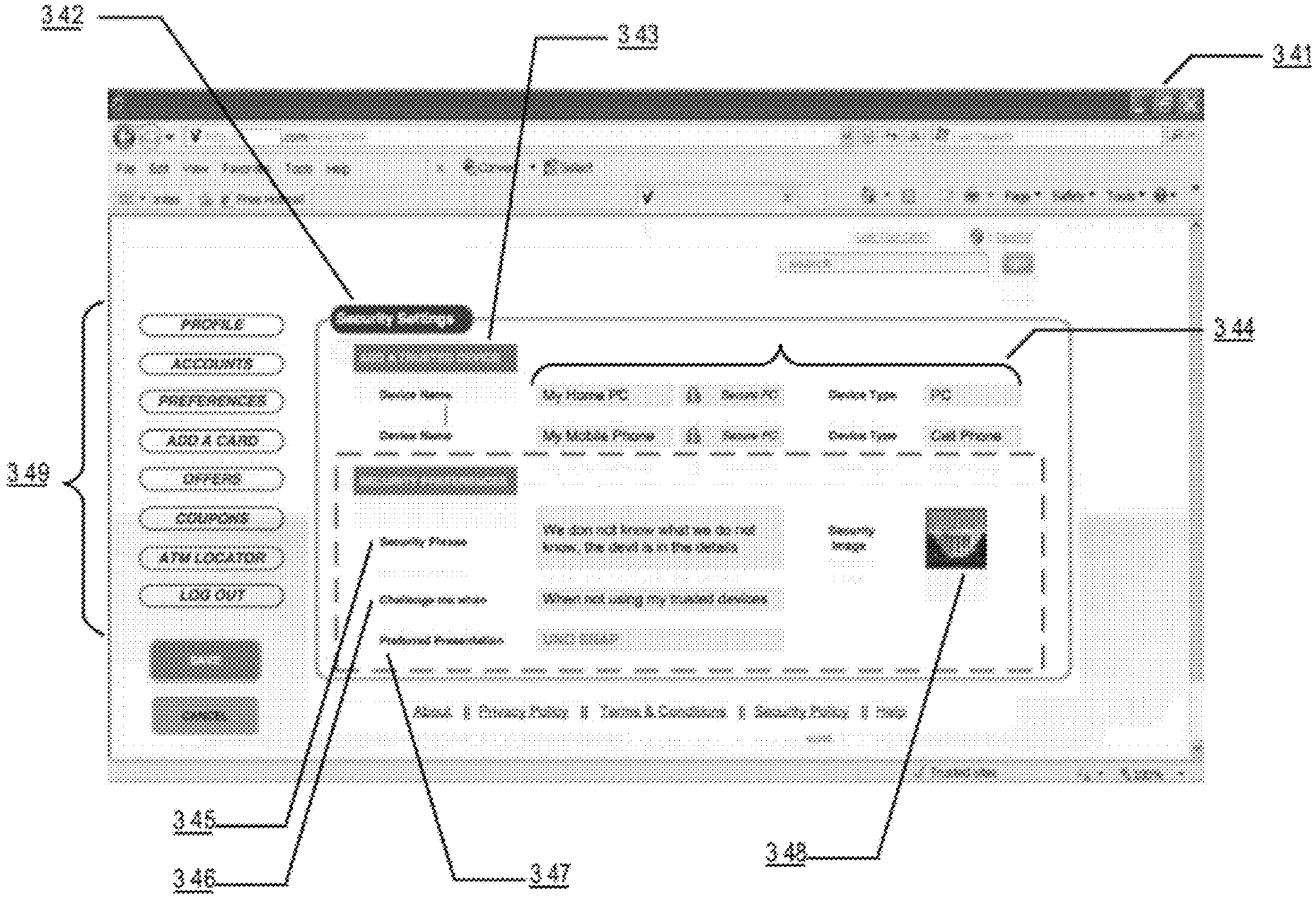
Example: One-Tap Mobile App: Payment Options

FIGURE 3C



Example: One-Tap Mobile App - Payment Options on Website

FIGURE 3D



Device Name My Home PC Secure PC Device Type PC

Device Name My Mobile Phone Secure PC Device Type Cell Phone

Security Phrase We don not know what we do not know, the devil is in the details

Security Image

Challenge me when When not using my trusted devices

Preferred Presentation [illegible]

About | Privacy Policy | Terms & Conditions | Security Policy | Help

FIGURE 3E

Example: Snap Mobile Payment: Security / Fraud Prevention

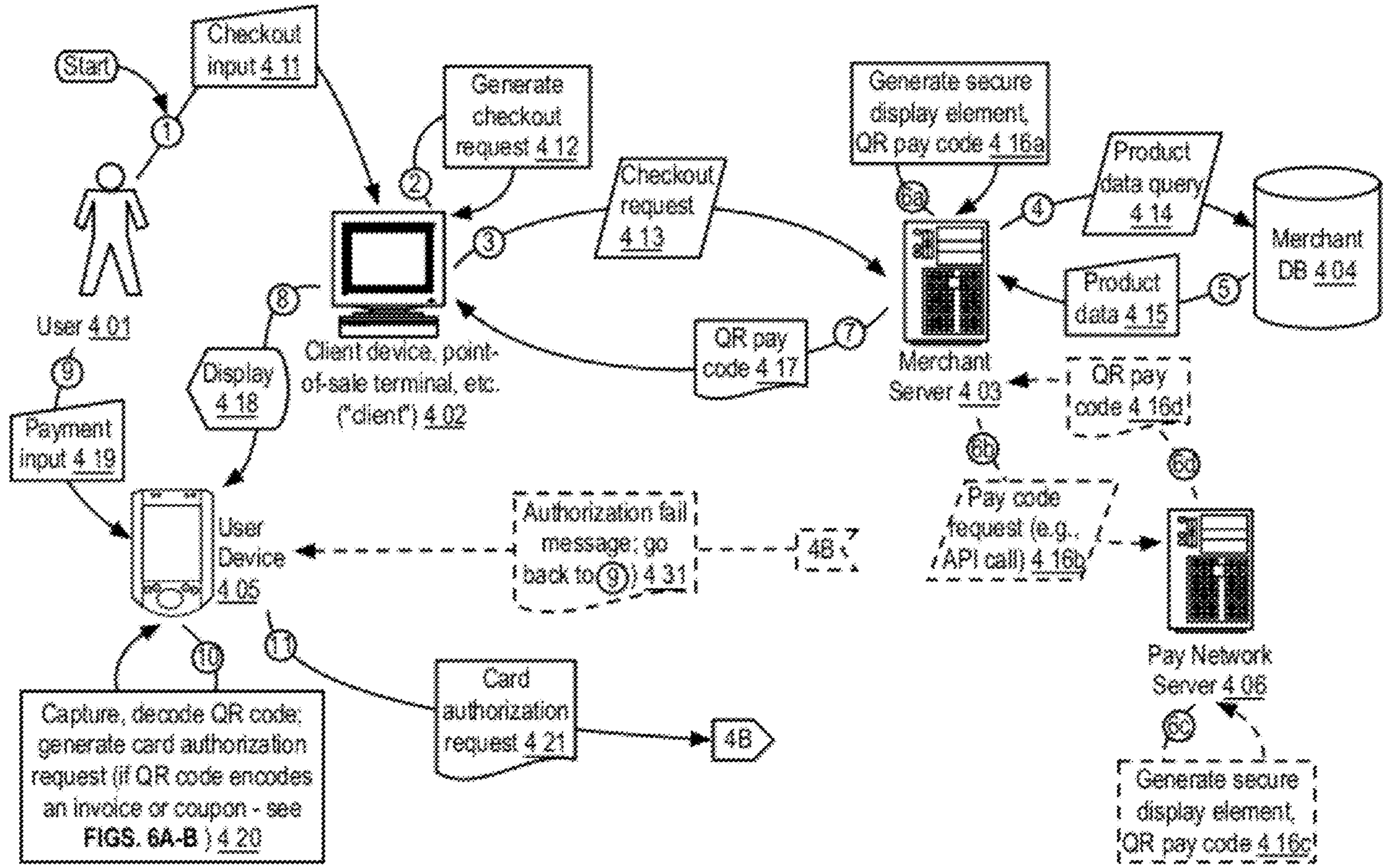
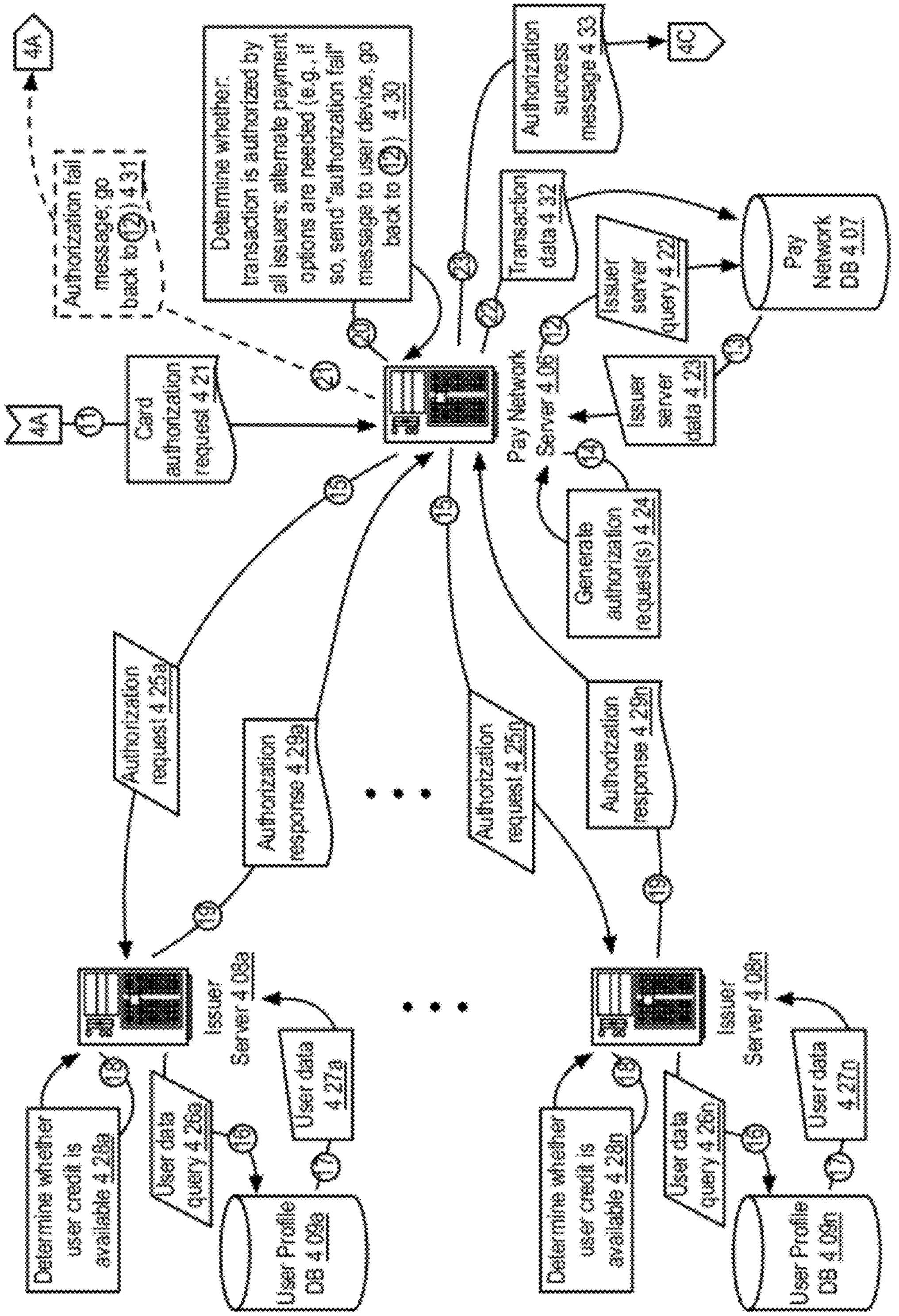

Start
1
Checkout input 4 11
User 4 01
2
Generate checkout request 4 12
3
Checkout request 4 13
Client device, point-of-sale terminal, etc. ("client") 4 02
Generate secure display element, QR pay code 4 16a
6a
4
Product data query 4 14
Merchant DB 4 04
5
Product data 4 15
7
QR pay code 4 17
Merchant Server 4 03
QR pay code 4 16d
6b
6d
Pay code request (e.g., API call) 4 16b
Pay Network Server 4 06
6c
Generate secure display element, QR pay code 4 16c
8
Display 4 18
9
Payment input 4 19
User Device 4 05
Authorization fail message; go back to 9 4 31
4B
10
11
Card authorization request 4 21
4B
Capture, decode QR code; generate card authorization request (if QR code encodes an invoice or coupon - see FIGS. 6A-B) 4 20

FIGURE 4A

Example: Snap mobile payment


4A
11
Card authorization request 4 21
4A
Authorization fail message; go back to 12) 4 31
21
20
Determine whether: transaction is authorized by all issuers; alternate payment options are needed (e.g., if so, send "authorization fail" message to user device, go back to 12) 4 30
22
Authorization success message 4 33
4C
21
Transaction data 4 32
12
Issuer server query 4 22
13
Issuer server data 4 23
Pay Network DB 4 07
Pay Network Server 4 06
14
Generate authorization request(s) 4 24
15
15
Authorization request 4 25a
Authorization response 4 29a
Authorization request 4 25n
Authorization response 4 29n
19
19
Issuer Server 4 08a
Issuer Server 4 08n
User data 4 27a
User data 4 27n
18
18
Determine whether user credit is available 4 28a
Determine whether user credit is available 4 28n
User data query 4 26a
User data query 4 26n
16
16
17
17
User Profile DB 4 09a
User Profile DB 4 09n

Example: Snap mobile payment

FIGURE 4B

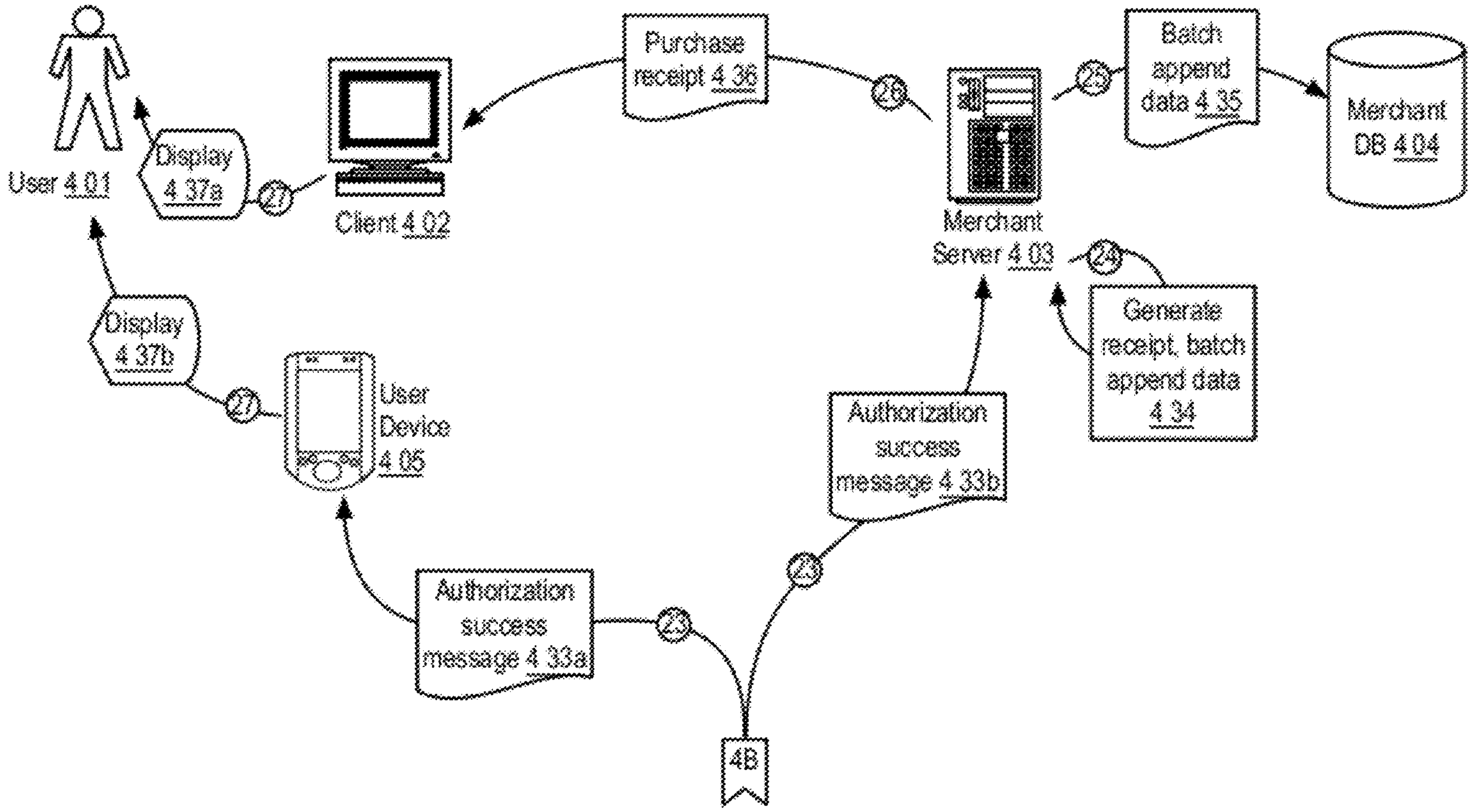


FIGURE 4C

Example: Snap mobile payment

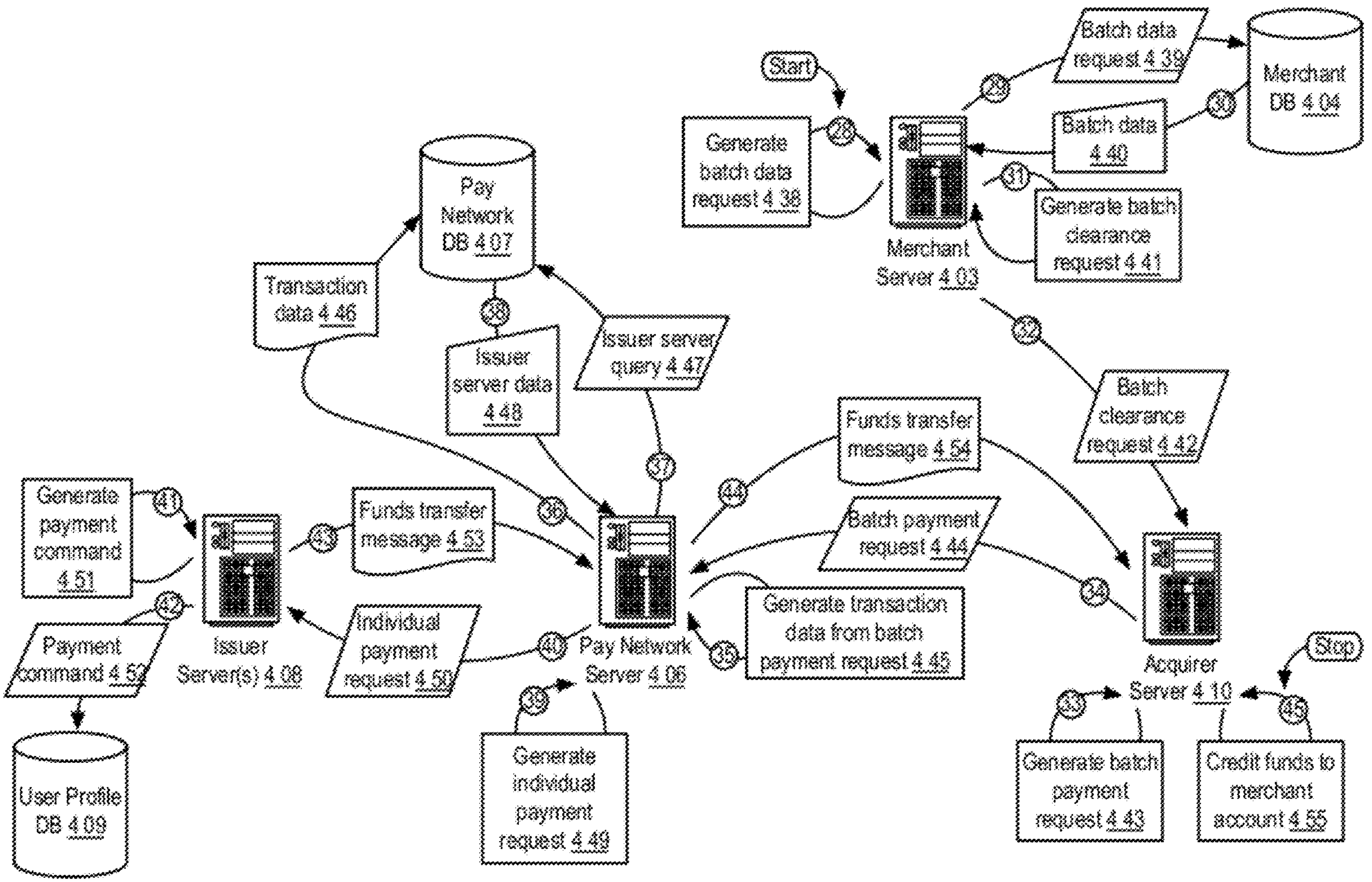

Start
28
Generate batch data request 4 38
Merchant Server 4 03
29
Batch data request 4 39
Merchant DB 4 04
30
Batch data 4 40
31
Generate batch clearance request 4 41
32
Batch clearance request 4 42
Acquirer Server 4 10
33
Generate batch payment request 4 43
34
Batch payment request 4 44
35
Generate transaction data from batch payment request 4 45
Pay Network Server 4 06
36
Transaction data 4 46
Pay Network DB 4 07
37
Issuer server query 4 47
38
Issuer server data 4 48
39
Generate individual payment request 4 49
40
Individual payment request 4 50
Issuer Server(s) 4 08
41
Generate payment command 4 51
42
Payment command 4 52
User Profile DB 4 09
43
Funds transfer message 4 53
44
Funds transfer message 4 54
45
Credit funds to merchant account 4 55
Stop

FIGURE 4D

Example: Snap mobile payment

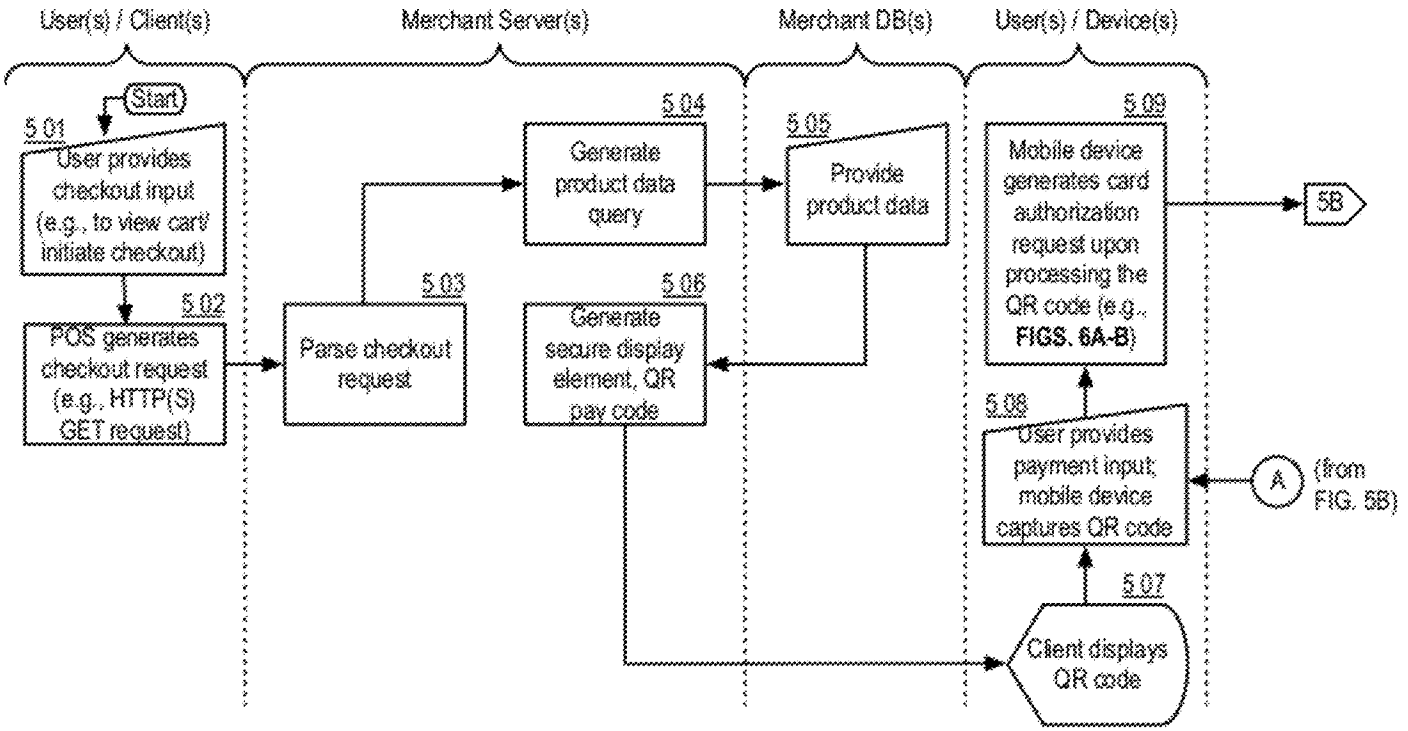
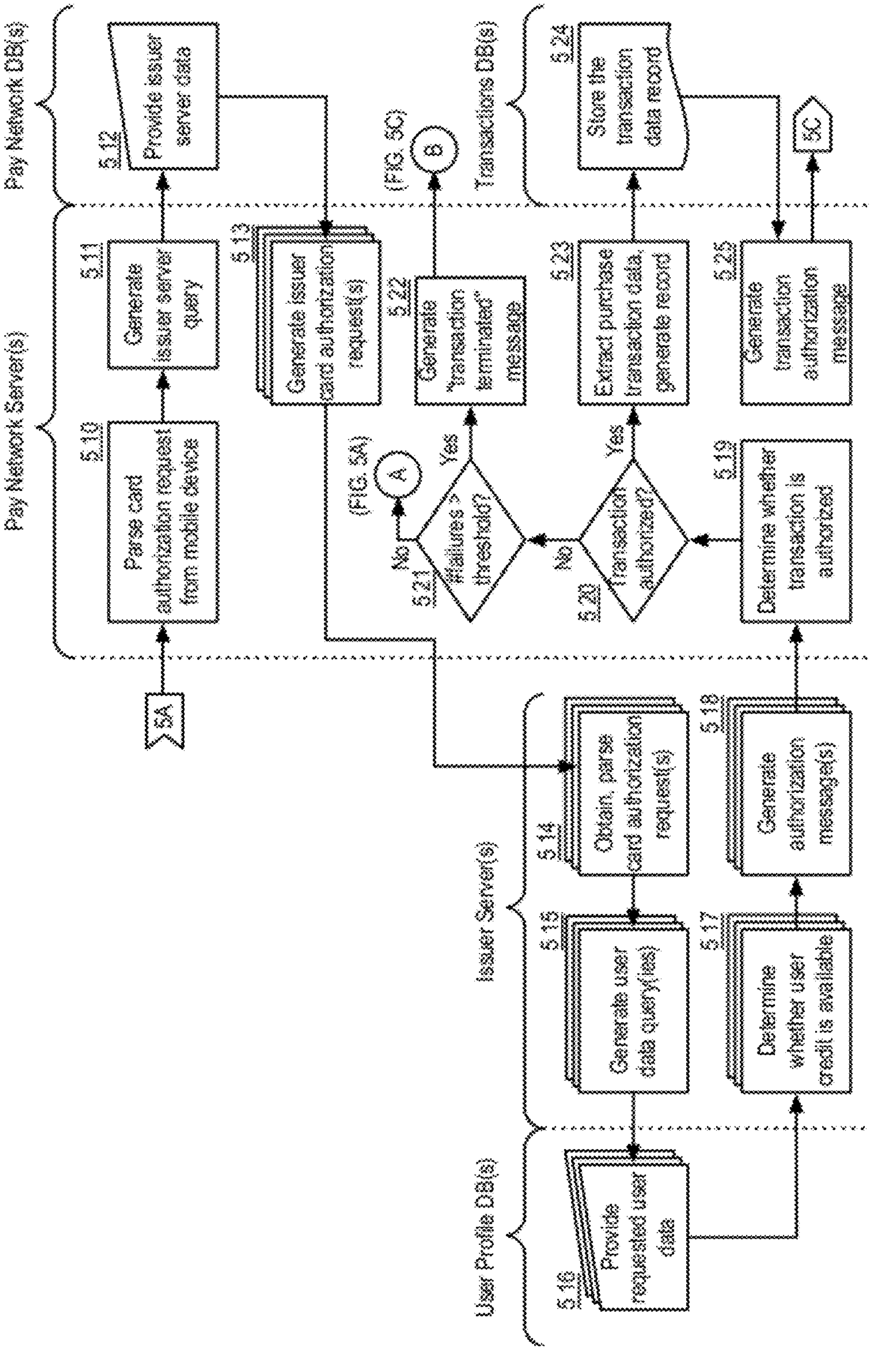

User(s) / Client(s)
Merchant Server(s)
Merchant DB(s)
User(s) / Device(s)
Start
5.01
User provides checkout input (e.g., to view cart/ initiate checkout)
5.02
POS generates checkout request (e.g., HTTP(S) GET request)
5.03
Parse checkout request
5.04
Generate product data query
5.05
Provide product data
5.06
Generate secure display element, QR pay code
5.07
Client displays QR code
5.08
User provides payment input; mobile device captures QR code
5.09
Mobile device generates card authorization request upon processing the QR code (e.g., FIGS. 6A-B)
5B
A
(from FIG. 5B)

FIGURE 5A

Example: Snap Mobile Payment Execution ("SMPE") component 500


Pay Network Server(s)
Pay Network DB(s)
Issuer Server(s)
User Profile DB(s)
Transactions DB(s)
5A
5.10
Parse card authorization request from mobile device
5.11
Generate issuer server query
5.12
Provide issuer server data
5.13
Generate issuer card authorization request(s)
5.14
Obtain, parse card authorization request(s)
5.15
Generate user data query(ies)
5.16
Provide requested user data
5.17
Determine whether user credit is available
5.18
Generate authorization message(s)
5.19
Determine whether transaction is authorized
5.20
Transaction authorized?
No
Yes
5.21
#failures > threshold?
No
Yes
(FIG. 5A)
A
5.22
Generate "transaction terminated" message
(FIG. 5C)
B
5.23
Extract purchase transaction data, generate record
5.24
Store the transaction data record
5.25
Generate transaction authorization message
5C

Example: Snap Mobile Payment Execution ("SMPE") component 500

FIGURE 5B

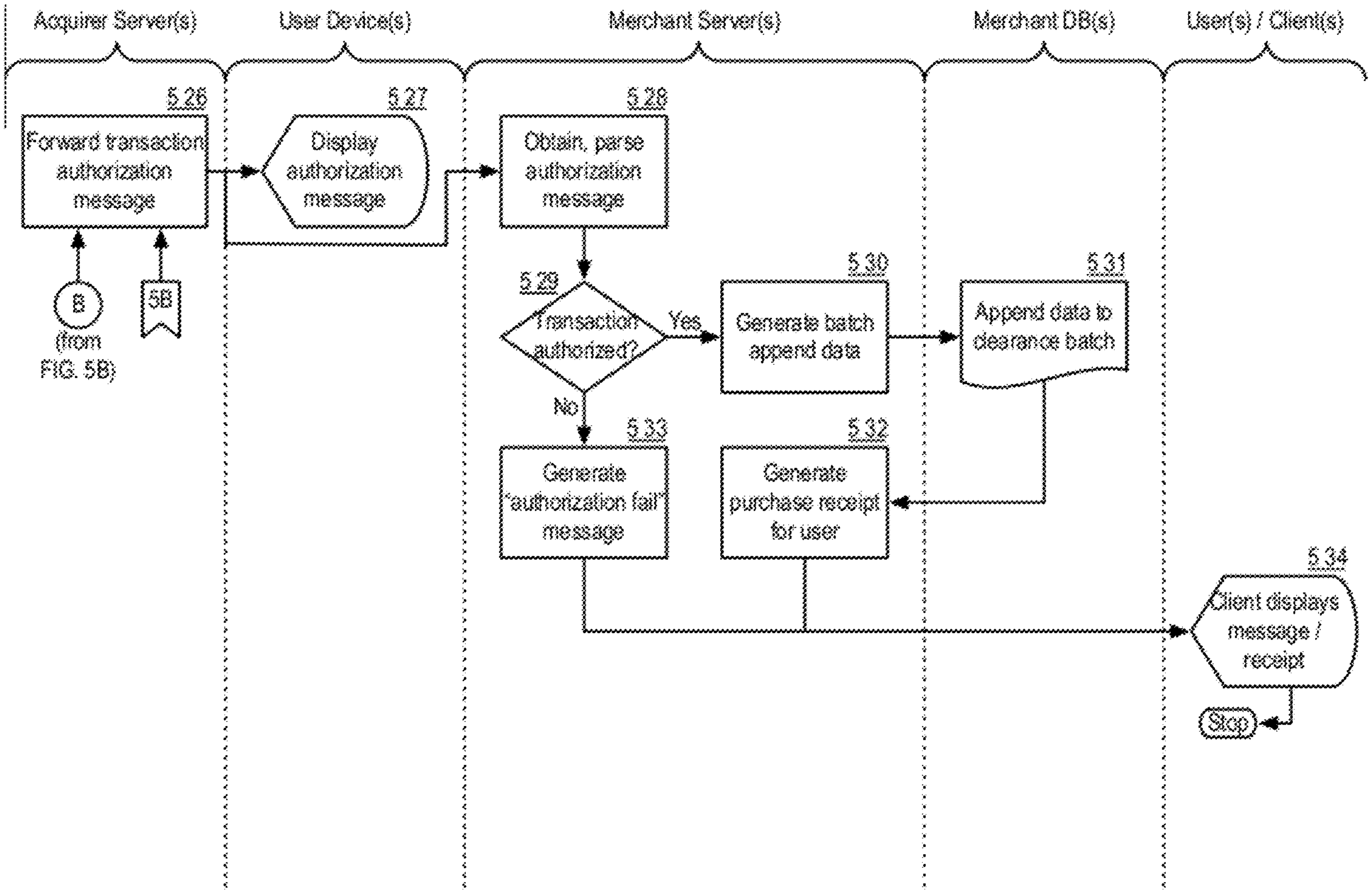

Acquirer Server(s)
User Device(s)
Merchant Server(s)
Merchant DB(s)
User(s) / Client(s)
5.26
Forward transaction authorization message
B
(from FIG. 5B)
5B
5.27
Display authorization message
5.28
Obtain, parse authorization message
5.29
Transaction authorized?
Yes
No
5.30
Generate batch append data
5.31
Append data to clearance batch
5.32
Generate purchase receipt for user
5.33
Generate "authorization fail" message
5.34
Client displays message / receipt
Stop

FIGURE 5C

Example: Snap Mobile Payment Execution ("SMPE") component 500

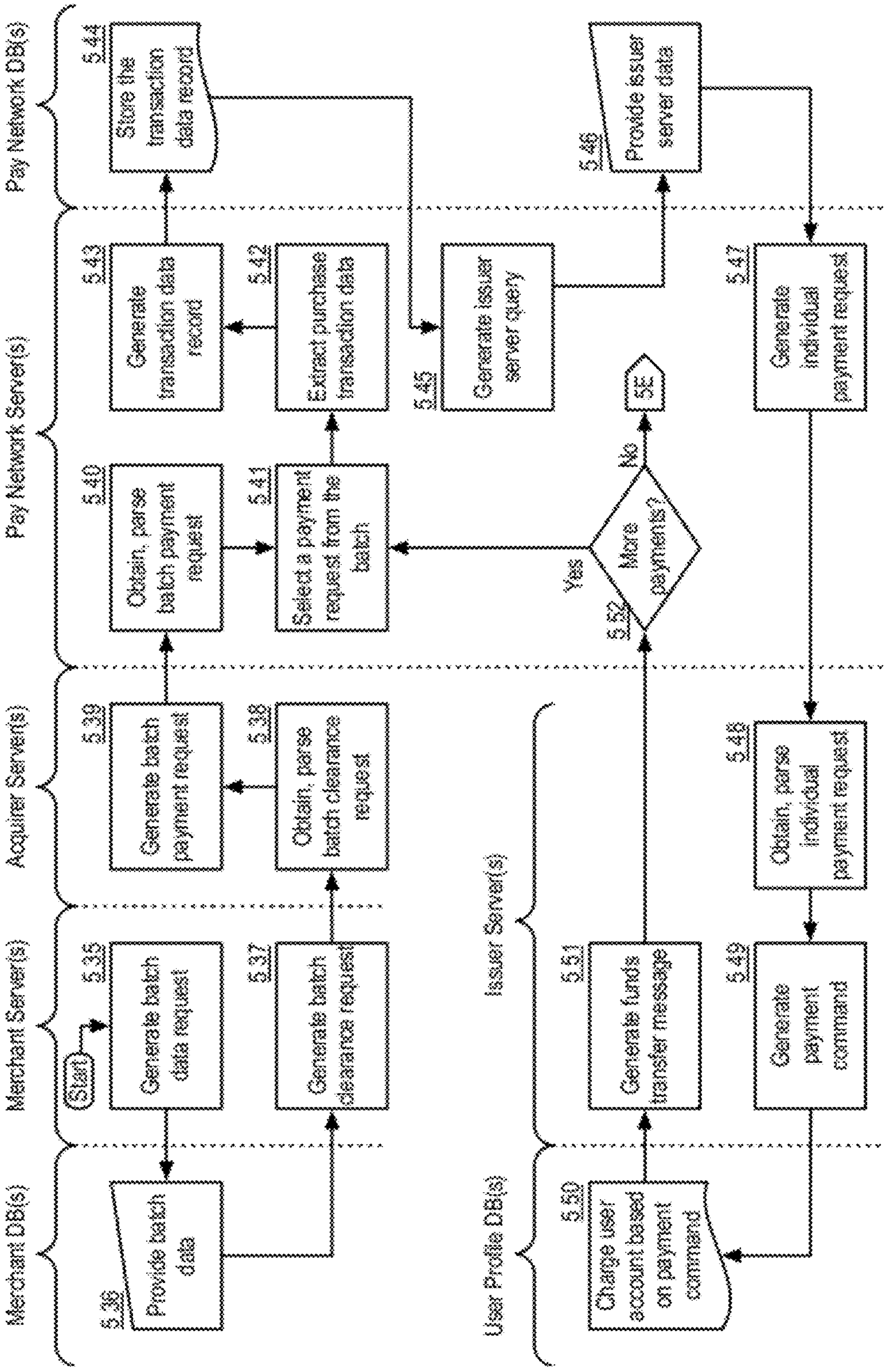
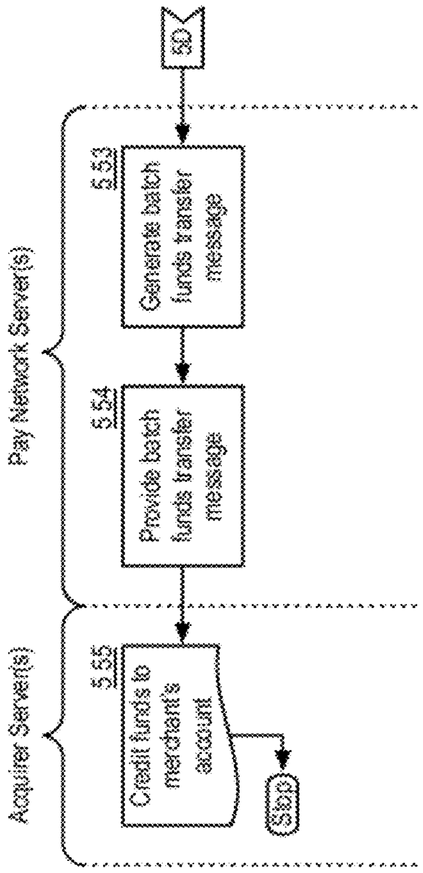

Merchant DB(s)
Merchant Server(s)
Acquirer Server(s)
Pay Network Server(s)
Pay Network DB(s)
Start
5.35
Generate batch data request
5.36
Provide batch data
5.37
Generate batch clearance request
5.38
Obtain, parse batch clearance request
5.39
Generate batch payment request
5.40
Obtain, parse batch payment request
5.41
Select a payment request from the batch
5.42
Extract purchase transaction data
5.43
Generate transaction data record
5.44
Store the transaction data record
5.45
Generate issuer server query
5.46
Provide issuer server data
5.47
Generate individual payment request
5.48
Obtain, parse individual payment request
5.49
Generate payment command
5.50
Charge user account based on payment command
5.51
Generate funds transfer message
5.52
More payments?
Yes
No
5E
User Profile DB(s)
Issuer Server(s)

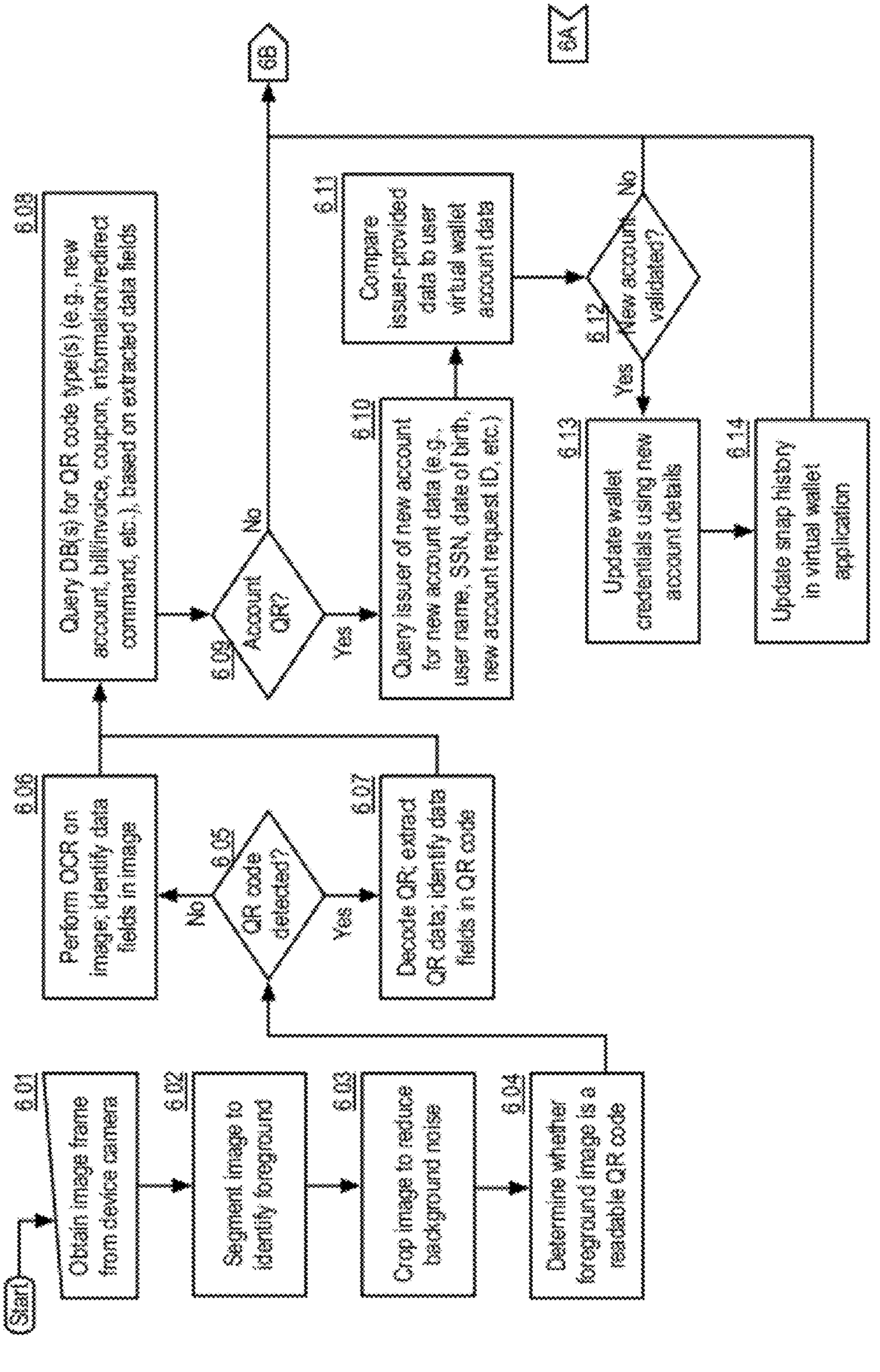
FIGURE 5D

Example: Snap Mobile Payment Execution ("SMPE") component 500


Pay Network Server(s)
Acquirer Server(s)
5D
553
Generate batch funds transfer message
554
Provide batch funds transfer message
555
Credit funds to merchant's account
Stop

Example: Snap Mobile Payment Execution ("SMPE") component 500

FIGURE 5E


Start
6.01
Obtain image frame from device camera
6.02
Segment image to identify foreground
6.03
Crop image to reduce background noise
6.04
Determine whether foreground image is a readable QR code
6.05
QR code detected?
No
Yes
6.06
Perform OCR on image; identify data fields in image
6.07
Decode QR; extract QR data; identify data fields in QR code
6.08
Query DB(s) for QR code type(s) (e.g., new account, bill/invoice, coupon, information/redirect command, etc.), based on extracted data fields
6.09
Account QR?
No
Yes
6B
6.10
Query issuer of new account for new account data (e.g., user name, SSN, date of birth, new account request ID, etc.)
6.11
Compare issuer-provided data to user virtual wallet account data
6.12
New account validated?
No
Yes
6.13
Update wallet credentials using new account details
6.14
Update snap history in virtual wallet application
6A

Example: Quick Response Code Processing ("QRCP") component 600

FIGURE 6A

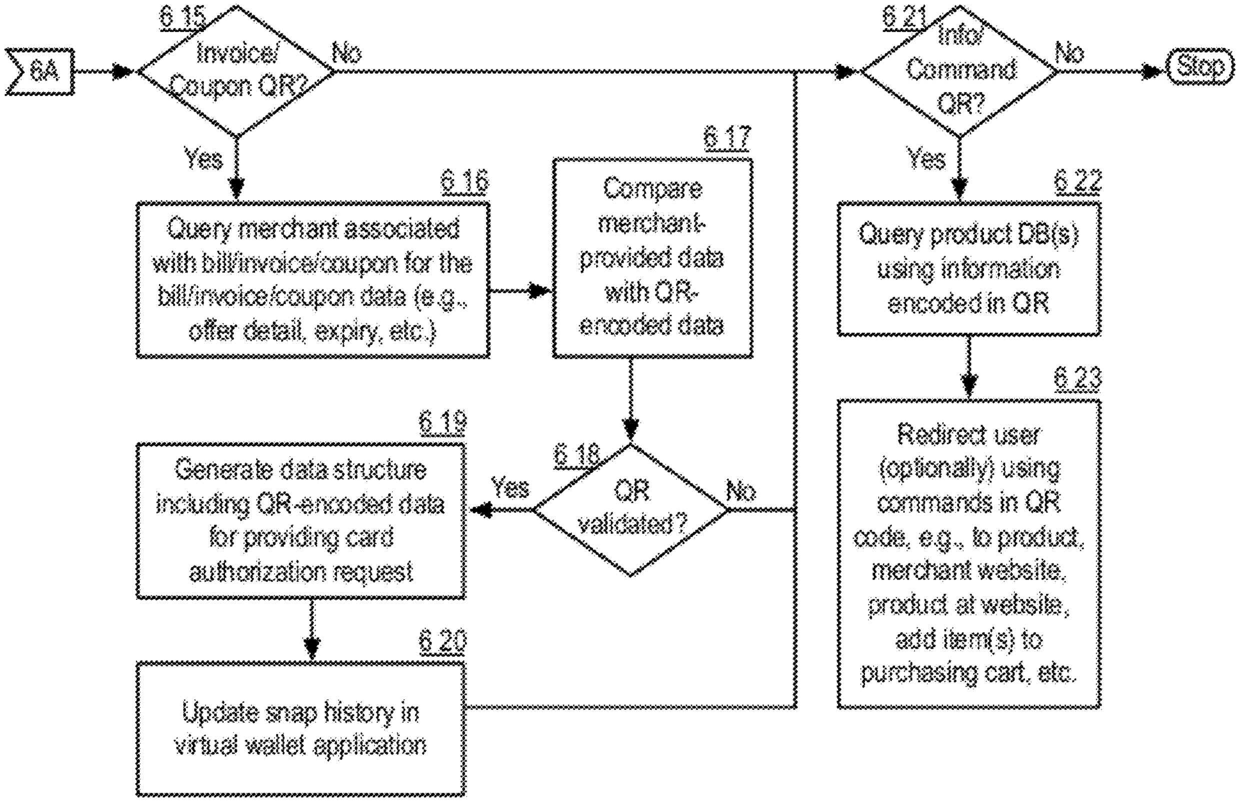

6A
6.15
Invoice/
Coupon QR?
No
Yes
6.16
Query merchant associated with bill/invoice/coupon for the bill/invoice/coupon data (e.g., offer detail, expiry, etc.)
6.17
Compare merchant-provided data with QR-encoded data
6.18
QR validated?
Yes
No
6.19
Generate data structure including QR-encoded data for providing card authorization request
6.20
Update snap history in virtual wallet application
6.21
Info/
Command
QR?
No
Stop
Yes
6.22
Query product DB(s) using information encoded in QR
6.23
Redirect user (optionally) using commands in QR code, e.g., to product, merchant website, product at website, add item(s) to purchasing cart, etc.

FIGURE 6B

Example: Quick Response Code Processing ("QRCP") component 600

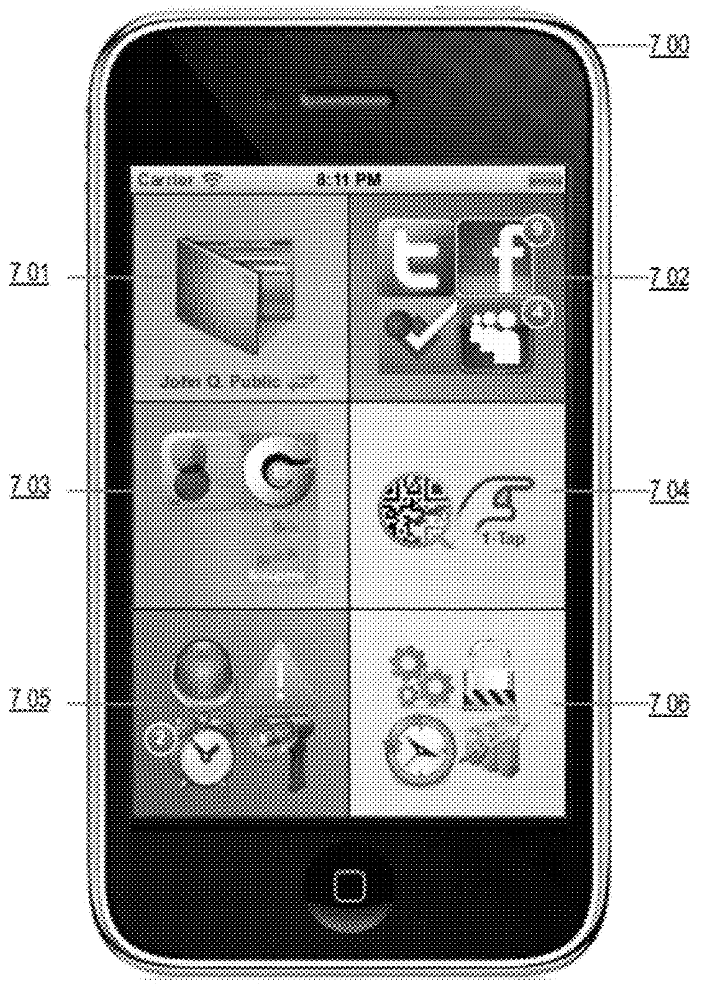


FIGURE 7

Example: Virtual Wallet Mobile App - Feature Overview

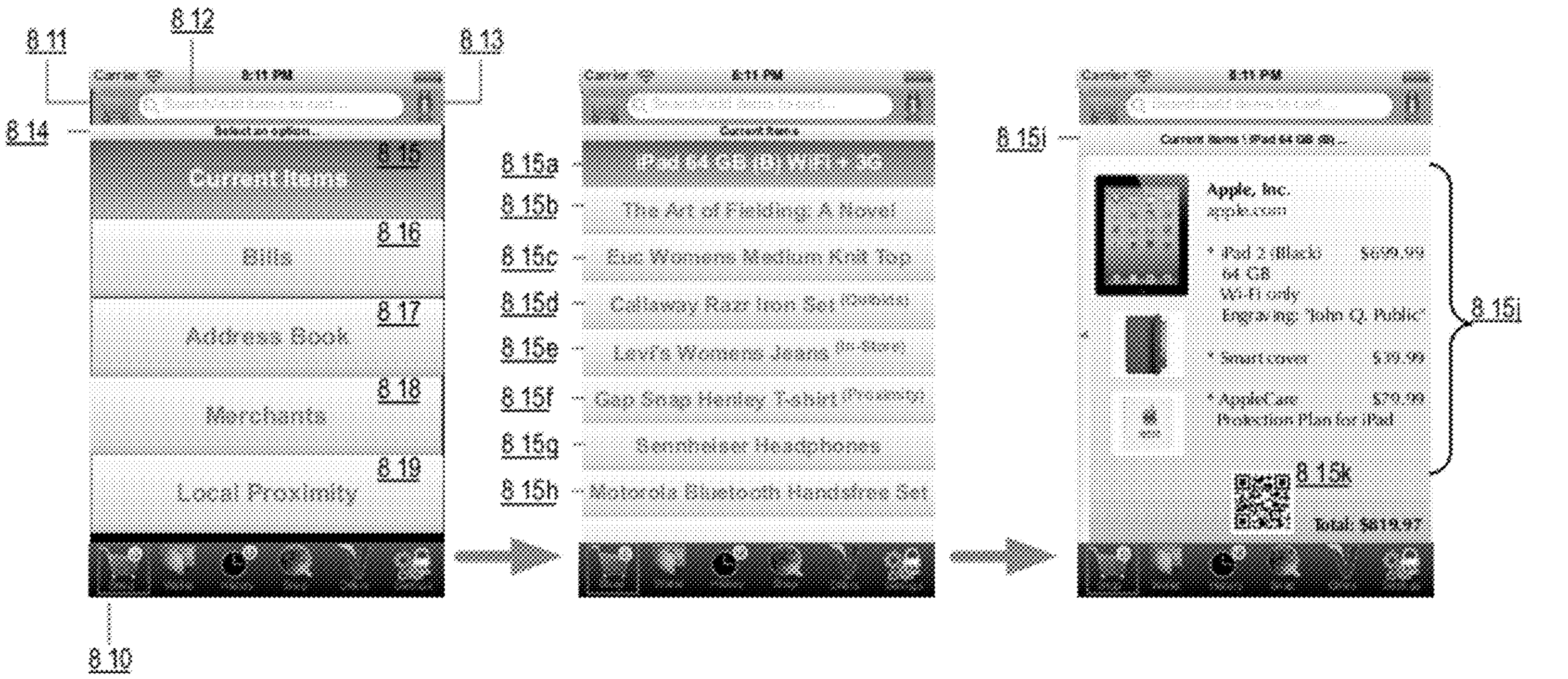
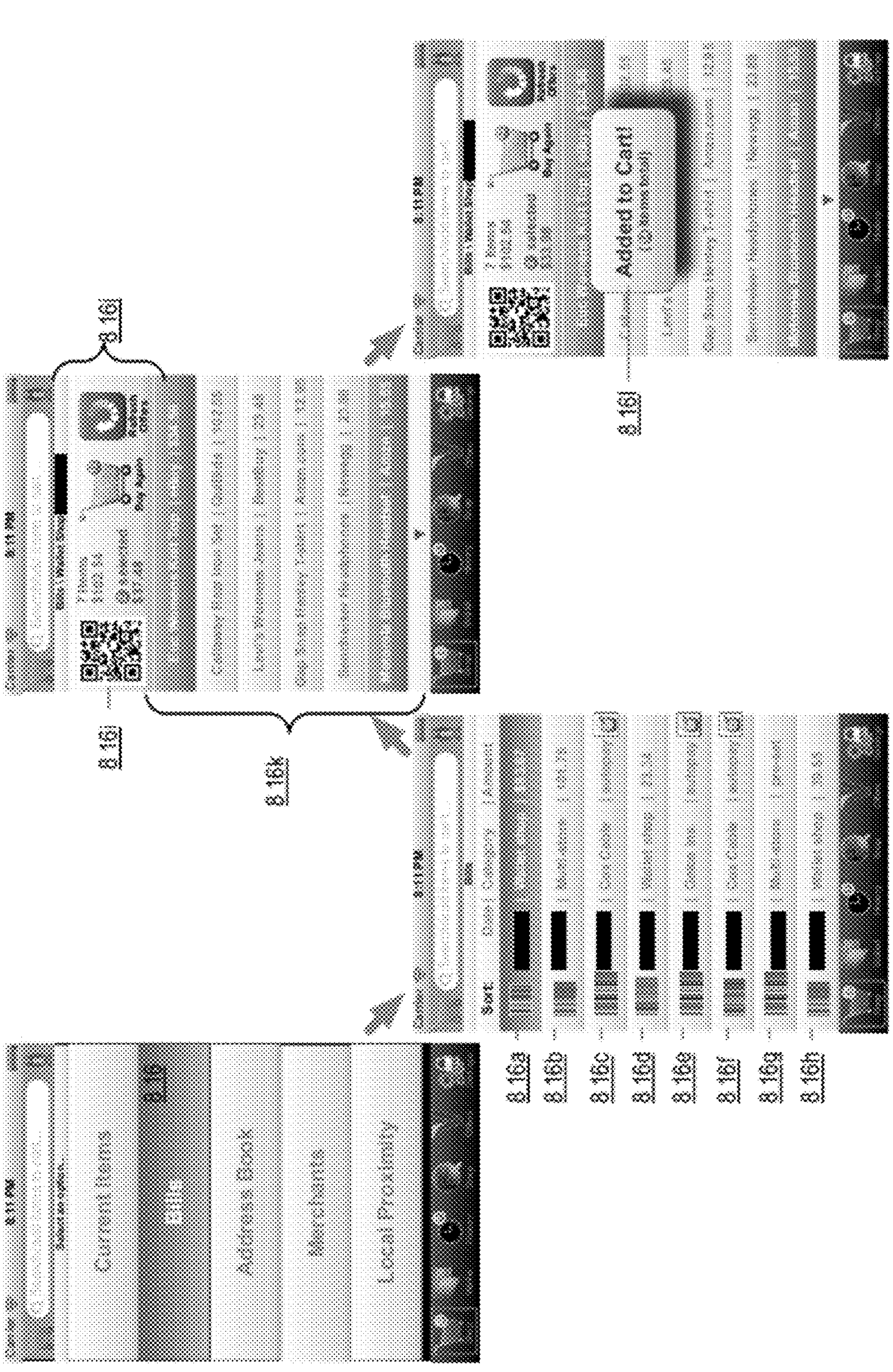


FIGURE 8A

Example: Virtual Wallet Mobile App - Shopping Mode



Example: Virtual Wallet Mobile App - Shopping Mode

FIGURE 8B

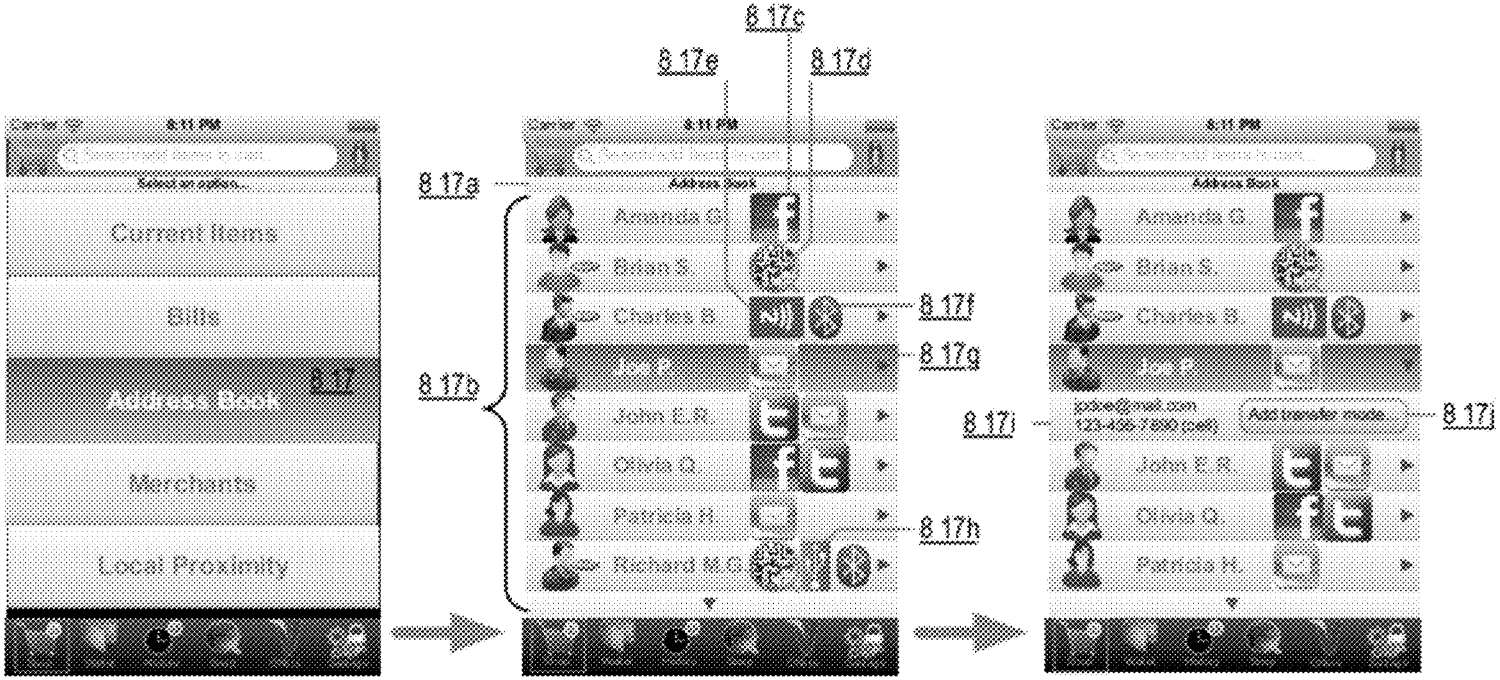


FIGURE 8C

Example: Virtual Wallet Mobile App - Shopping Mode

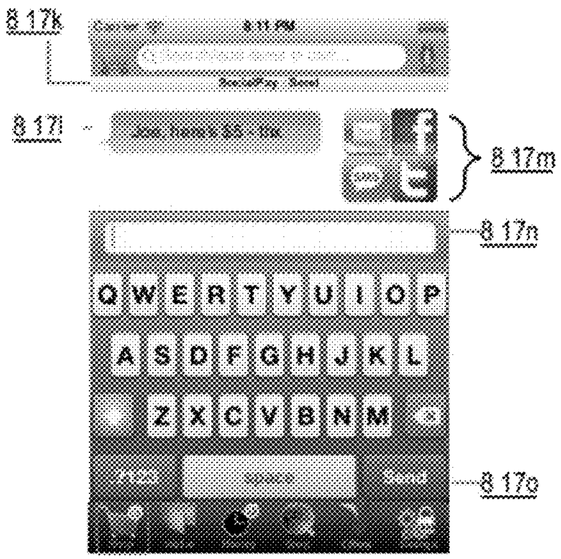


FIGURE 8D

Example: Virtual Wallet Mobile App - SocialPay Mode

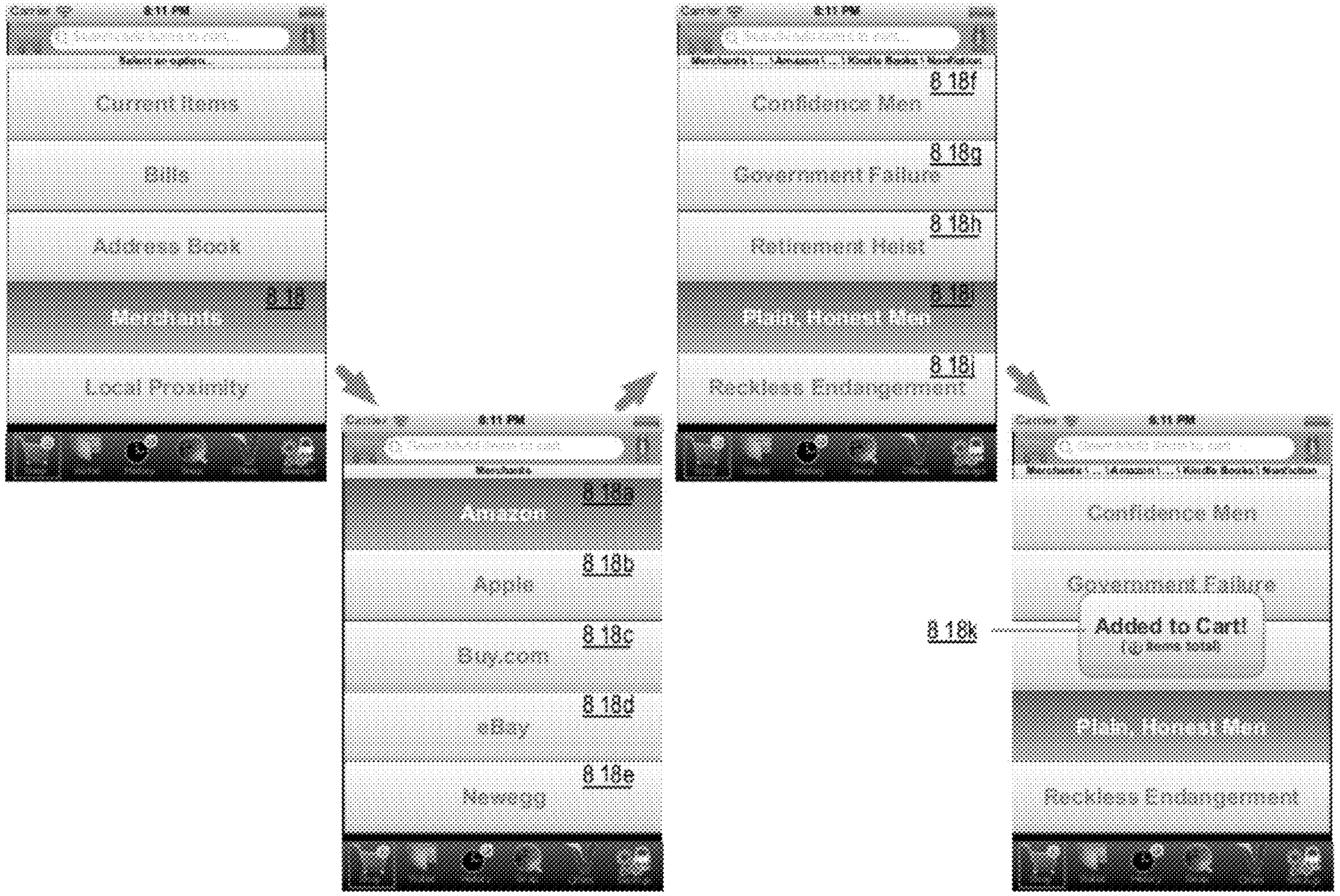


FIGURE 8E

Example: Virtual Wallet Mobile App - Shopping Mode

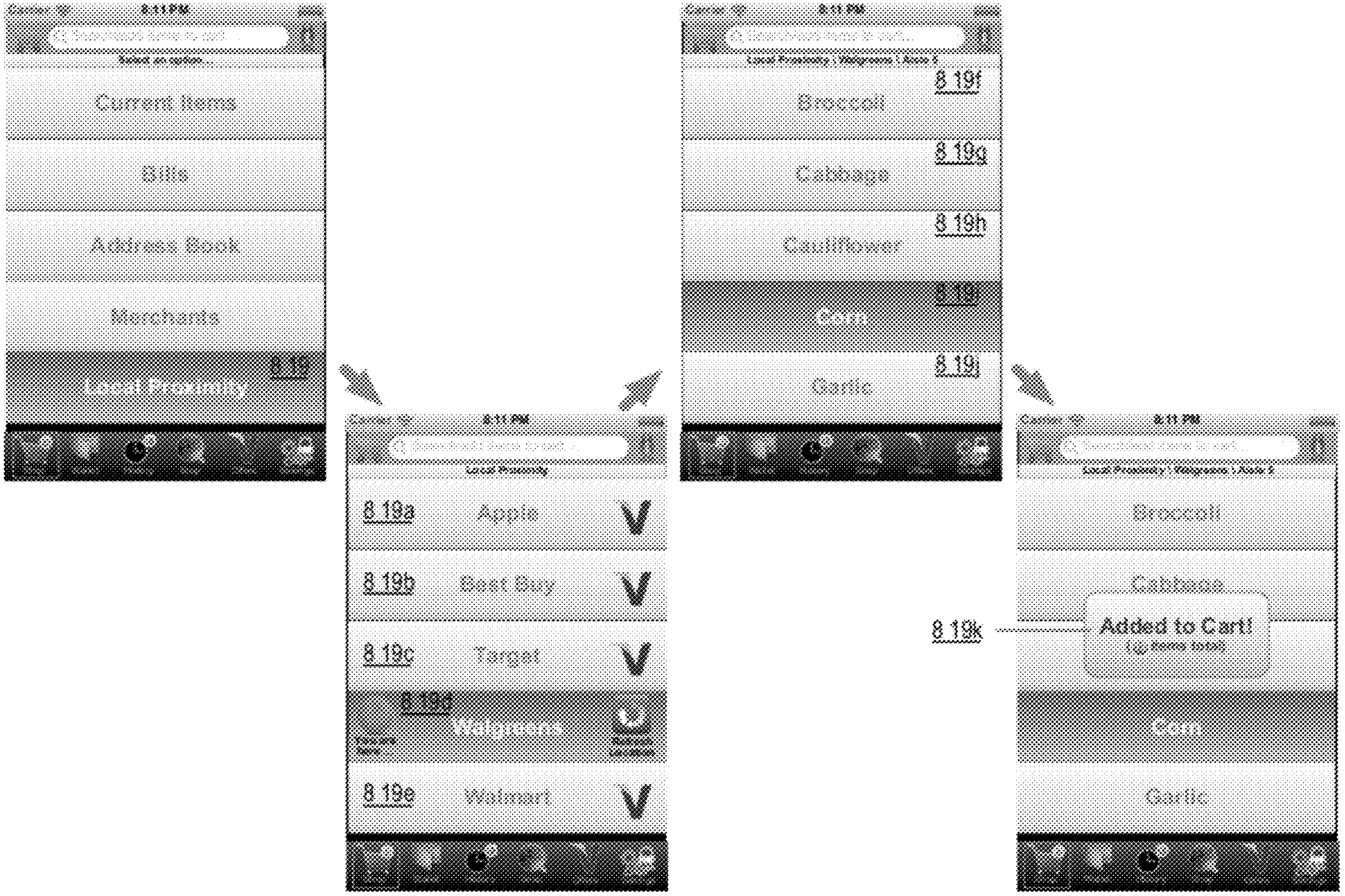


FIGURE 8F

Example: Virtual Wallet Mobile App - Shopping Mode

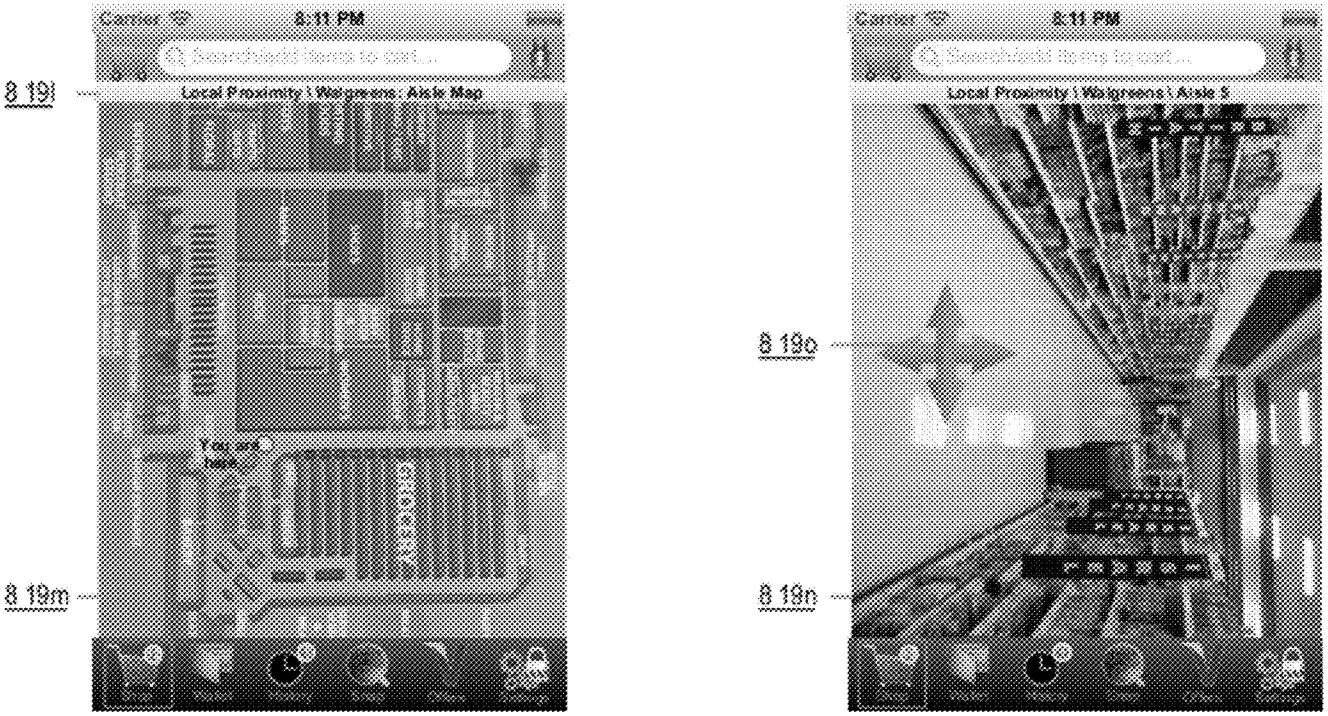

8 19l
8 19m
Carrier
8:11 PM
Search/add items to cart...
Local Proximity | Walgreens | Aisle Map
You are here
GROCERY
8 19o
8 19n
Carrier
8:11 PM
Search/add items to cart...
Local Proximity | Walgreens | Aisle 5

FIGURE 8G

Example: Virtual Wallet Mobile App - Shopping Mode

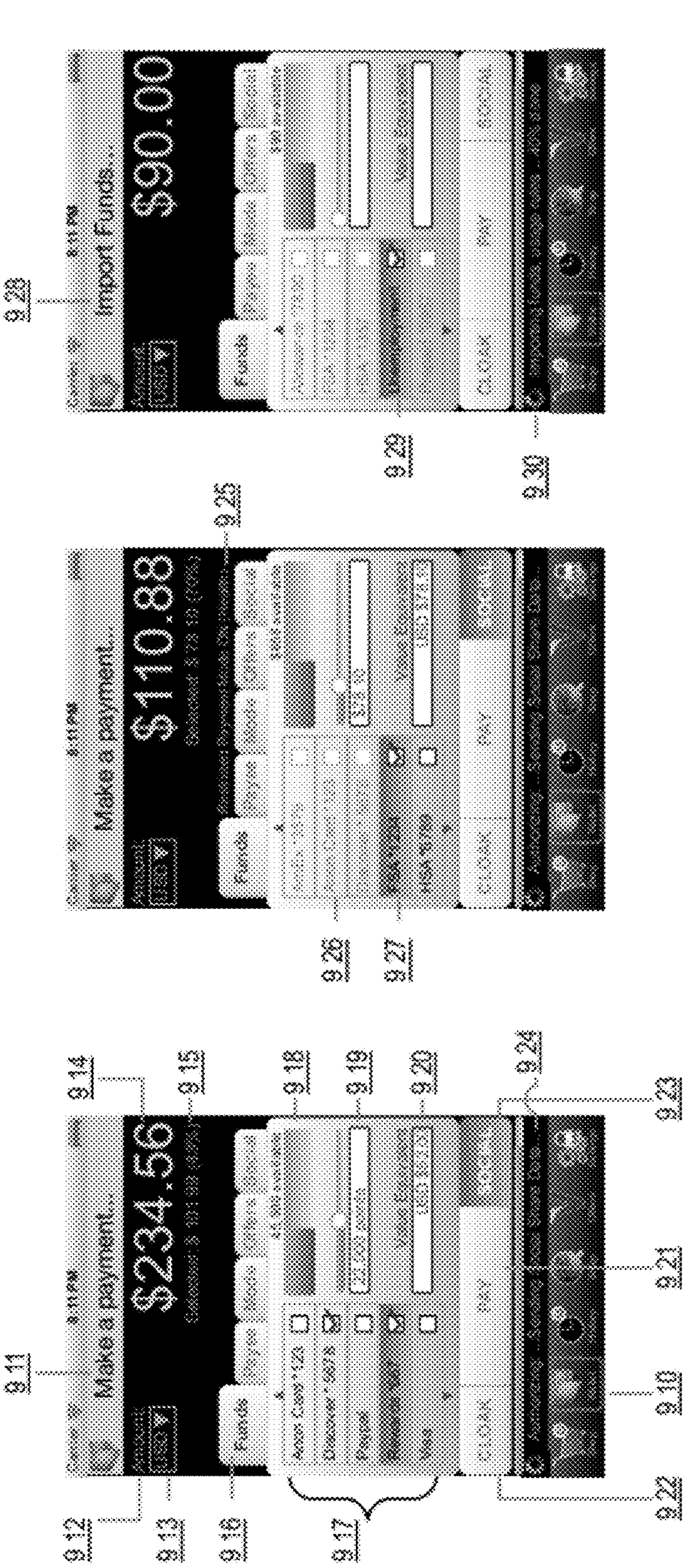


FIGURE 9A

Example: Virtual Wallet Mobile App - Payment Mode

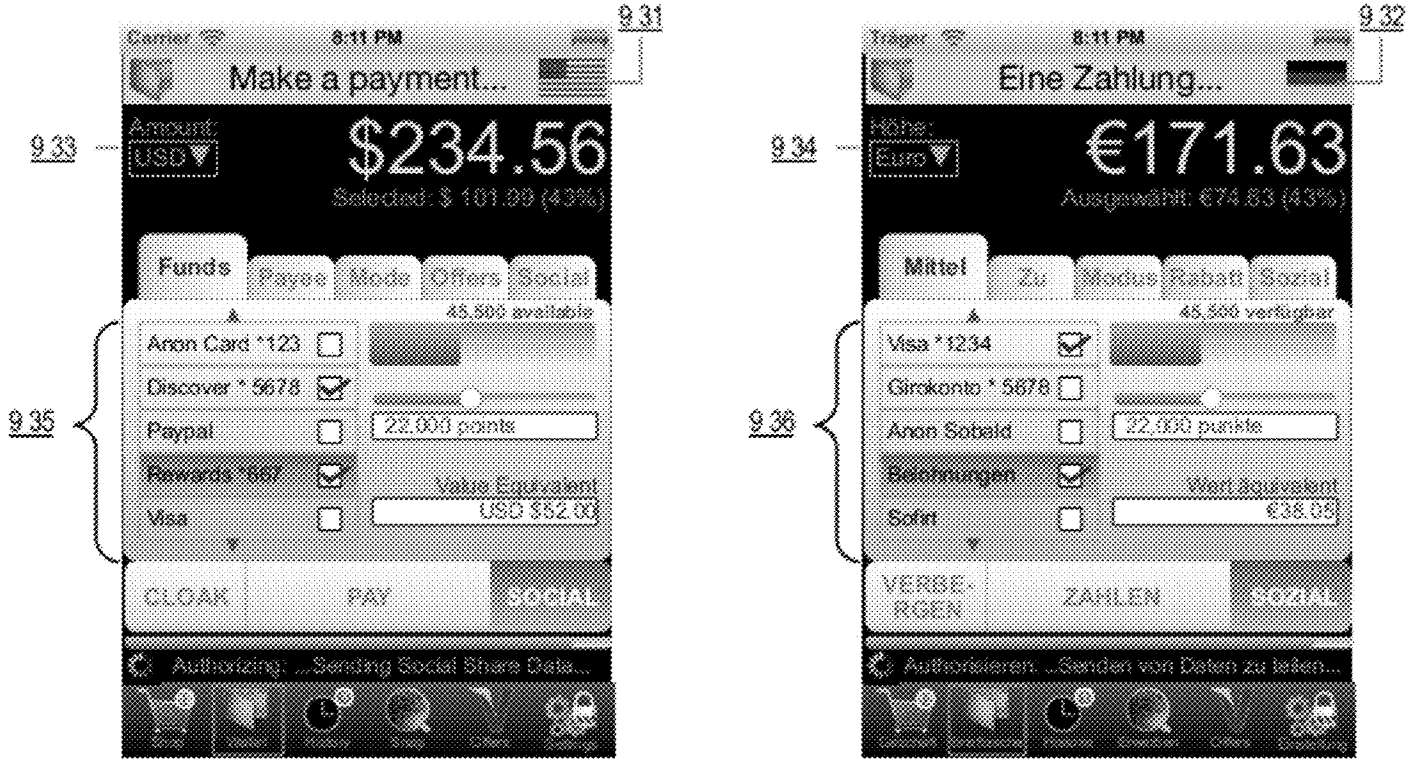
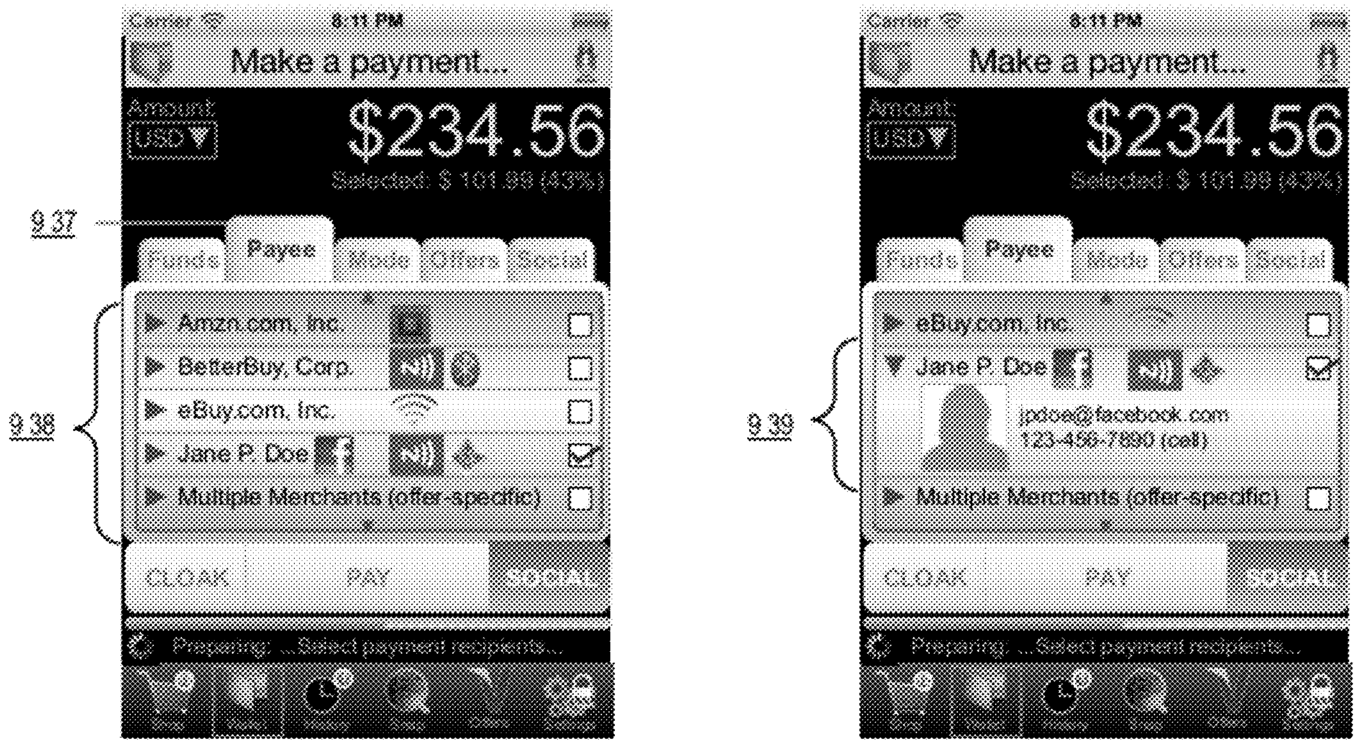


FIGURE 9B

Example: Virtual Wallet Mobile App - Dynamic Payment Optimization



Make a payment...

Amount: USD ▼ $234.56

Selected: $ 101.98 (43%)

- ▶ Amzn.com, Inc. □
- ▶ BetterBuy, Corp. □
- ▶ eBuy.com, Inc. □
- ▶ Jane P. Doe ☑
- ▶ Multiple Merchants (offer-specific) □

Preparing: ...Select payment recipients...

Make a payment...

Amount: USD ▼ $234.56

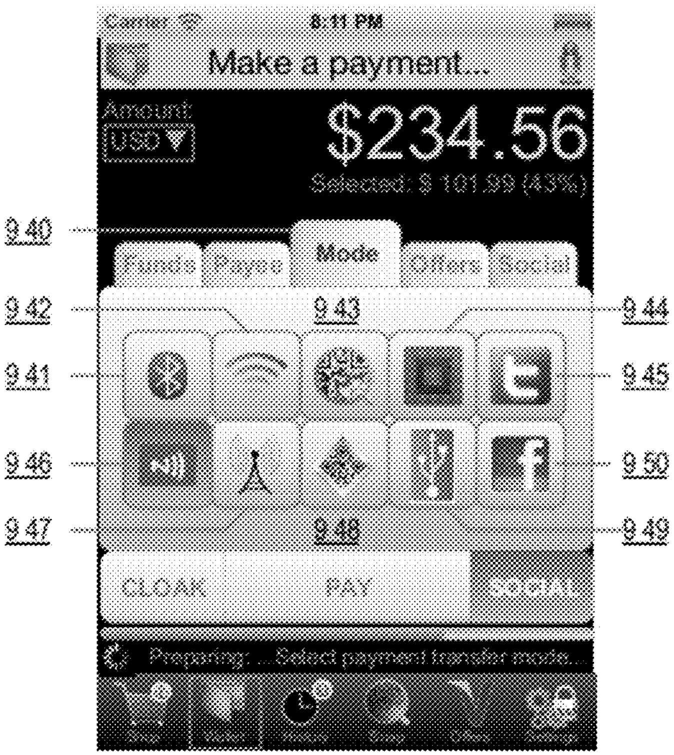
Selected: $ 101.98 (43%)

- ▶ eBuy.com, Inc. □
- ▼ Jane P. Doe ☑
  jpdoe@facebook.com
  123-456-7890 (cell)
- ▶ Multiple Merchants (offer-specific) □

Preparing: ...Select payment recipients...

FIGURE 9C

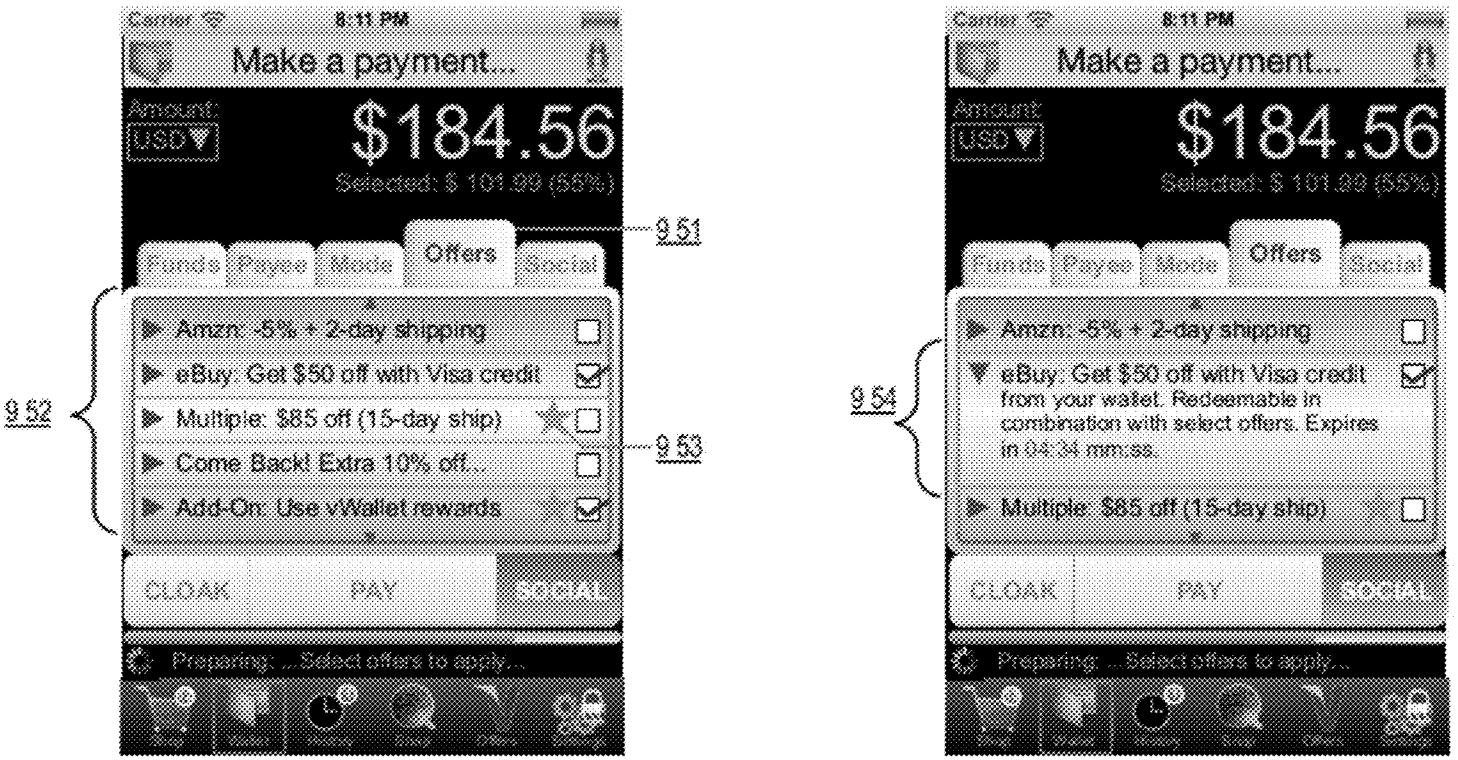
Example: Virtual Wallet Mobile App



Preparing: ...Select payment transfer mode...

Example: Virtual Wallet Mobile App

FIGURE 9D



▶ Amzn: -5% + 2-day shipping □
▶ eBuy: Get $50 off with Visa credit ☑
▶ Multiple: $85 off (15-day ship) □
▶ Come Back! Extra 10% off... □
▶ Add-On: Use vWallet rewards ☑

Preparing: ...Select offers to apply...

▶ Amzn: -5% + 2-day shipping □
▼ eBuy: Get $50 off with Visa credit from your wallet. Redeemable in combination with select offers. Expires in 04:34 mm:ss. ☑
▶ Multiple: $85 off (15-day ship) □

Preparing: ...Select offers to apply...

FIGURE 9E

Example: Virtual Wallet Mobile App

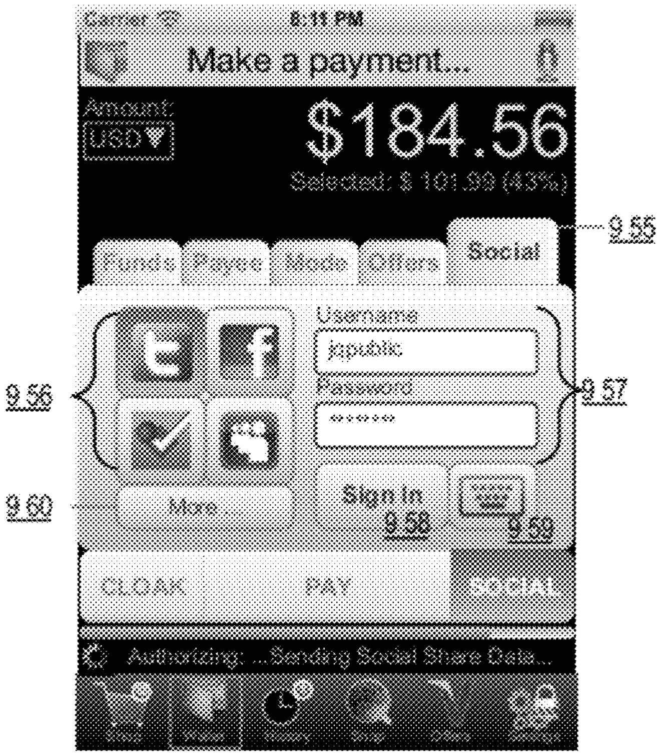


FIGURE 9F

Example: Virtual Wallet Mobile App

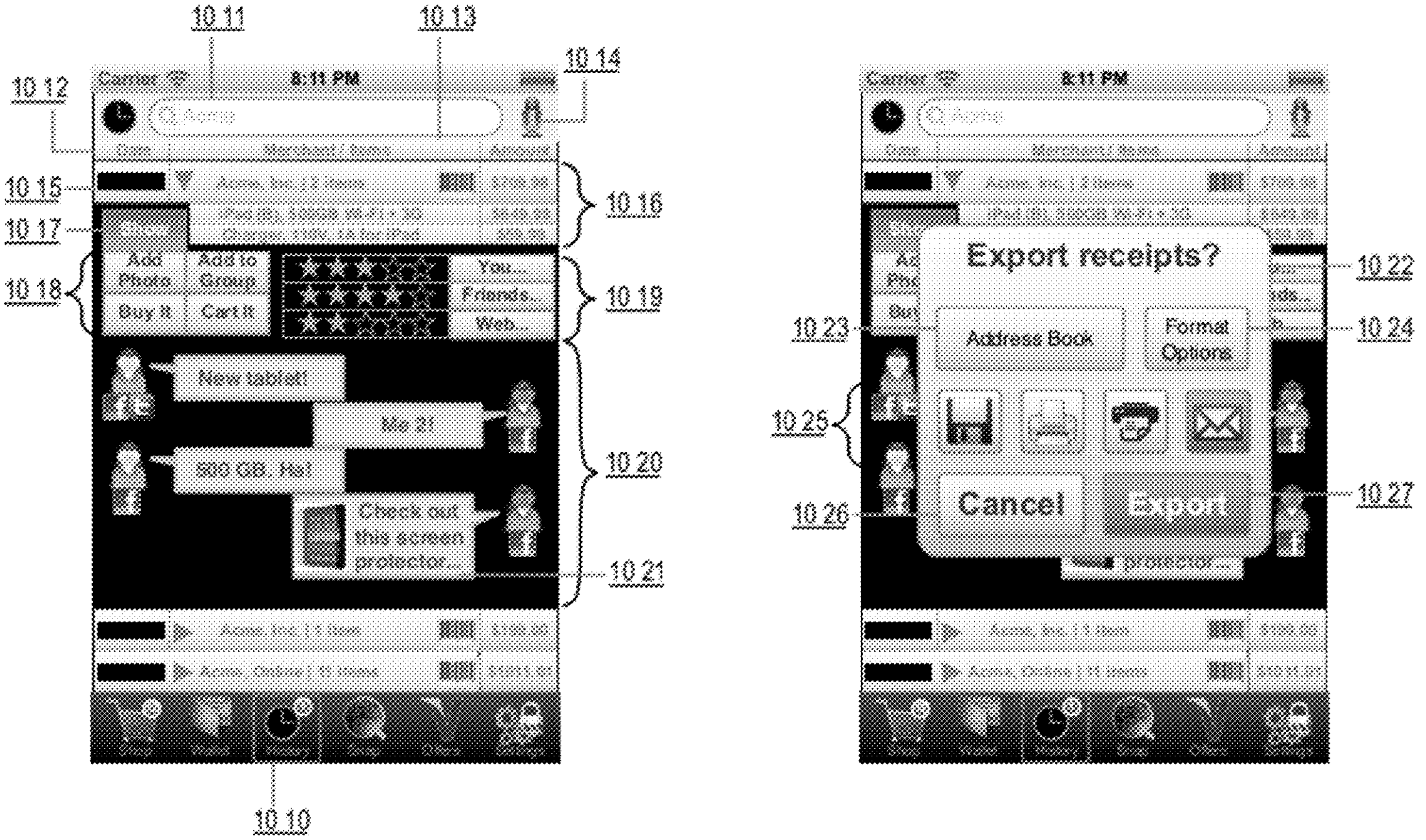
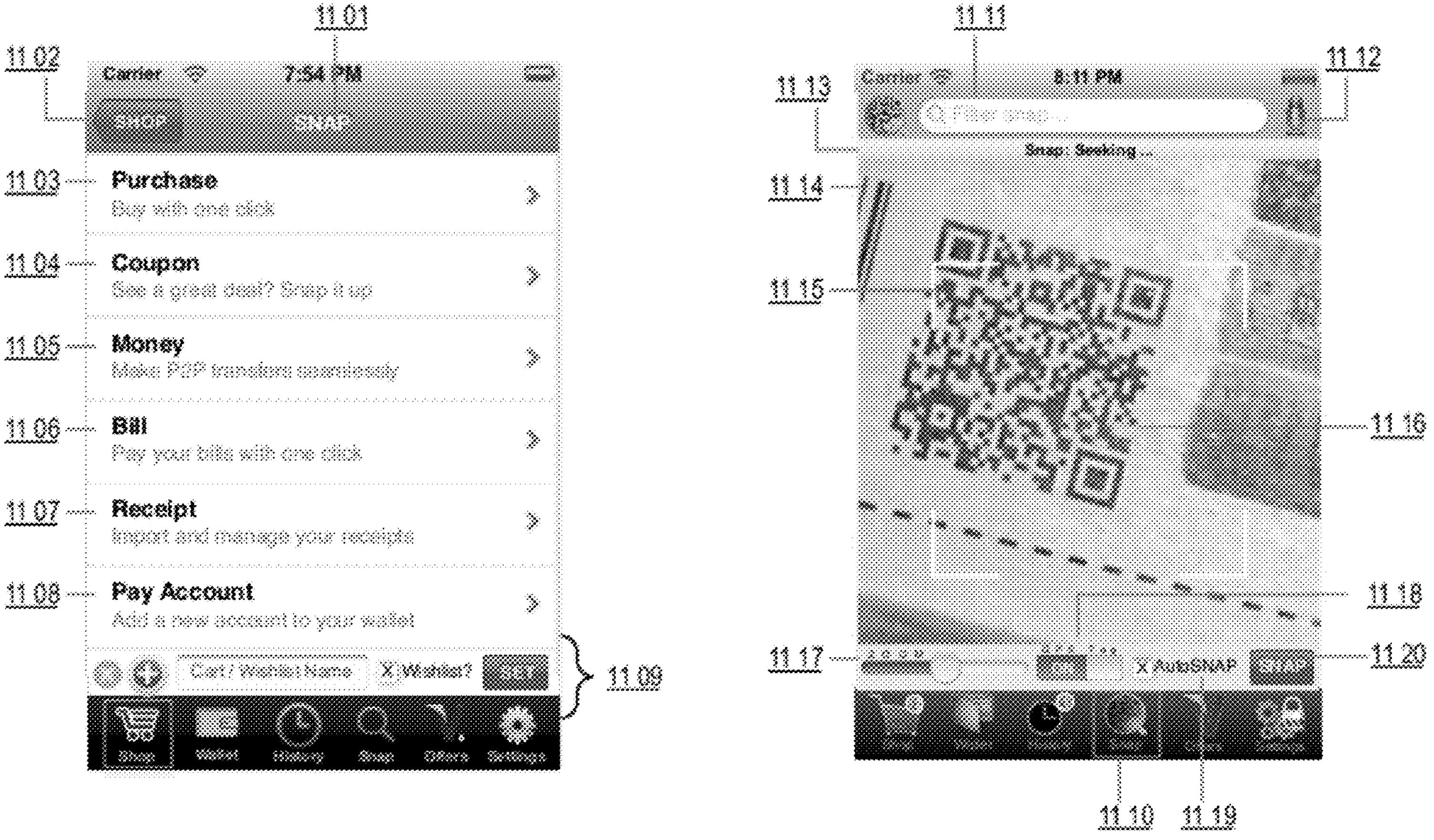


FIGURE 10

Example: Virtual Wallet Mobile App - History



Purchase
Buy with one click

Coupon
See a great deal? Snap it up

Money
Make P2P transfers seamlessly

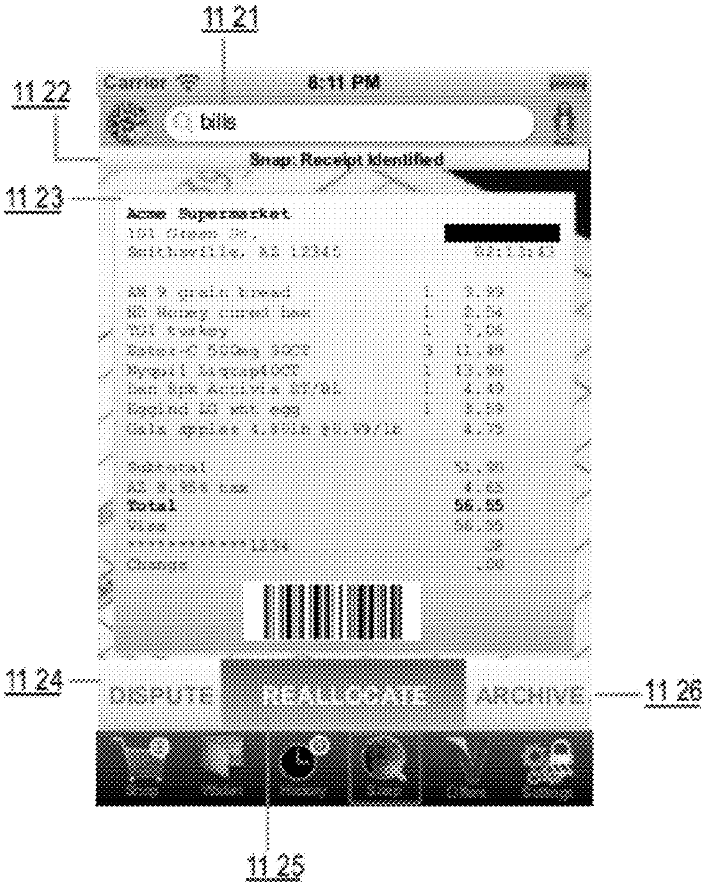
Bill
Pay your bills with one click

Receipt
Import and manage your receipts

Pay Account
Add a new account to your wallet

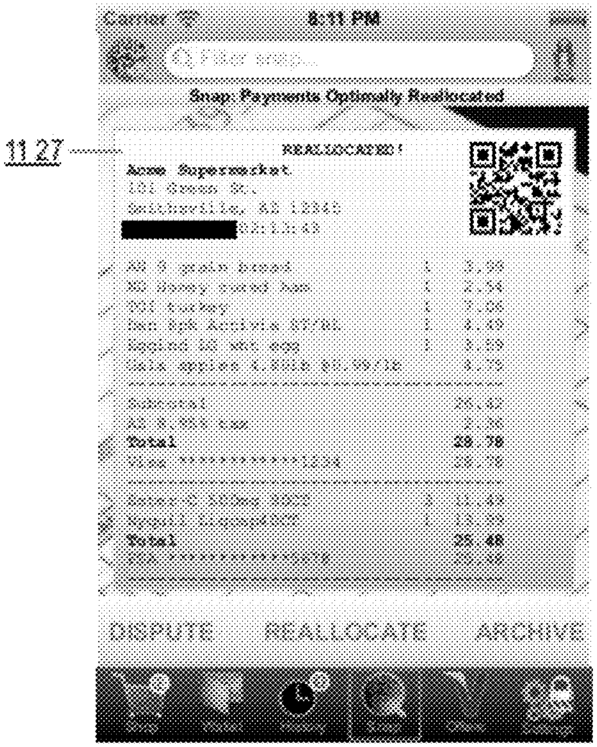
FIGURE 11A

Example: Virtual Wallet Mobile App - Snap Mode



Snap: Receipt Identified

```
Acme Supermarket
101 Green St.,
Smithville, AS 12345                [redacted]
                                      02:13:43

AN 9 grain bread                 1      3.99
HD Honey cured ham               1      2.54
TOI turkey                       1      7.06
Rotex-C 500mg 30CT               3     11.89
Nyquil Liquid4OCT                1     13.99
Dan 6pk Activia 6T/6L            1      4.49
Eggland LG wht egg               1      3.59
Gala apples 1.01lb $0.99/lb             4.75

Subtotal                               51.90
AZ 8.95% tax                            4.65
Total                                  56.55
Visa                                   56.55
**************1234                       .00
Change                                   .00
```

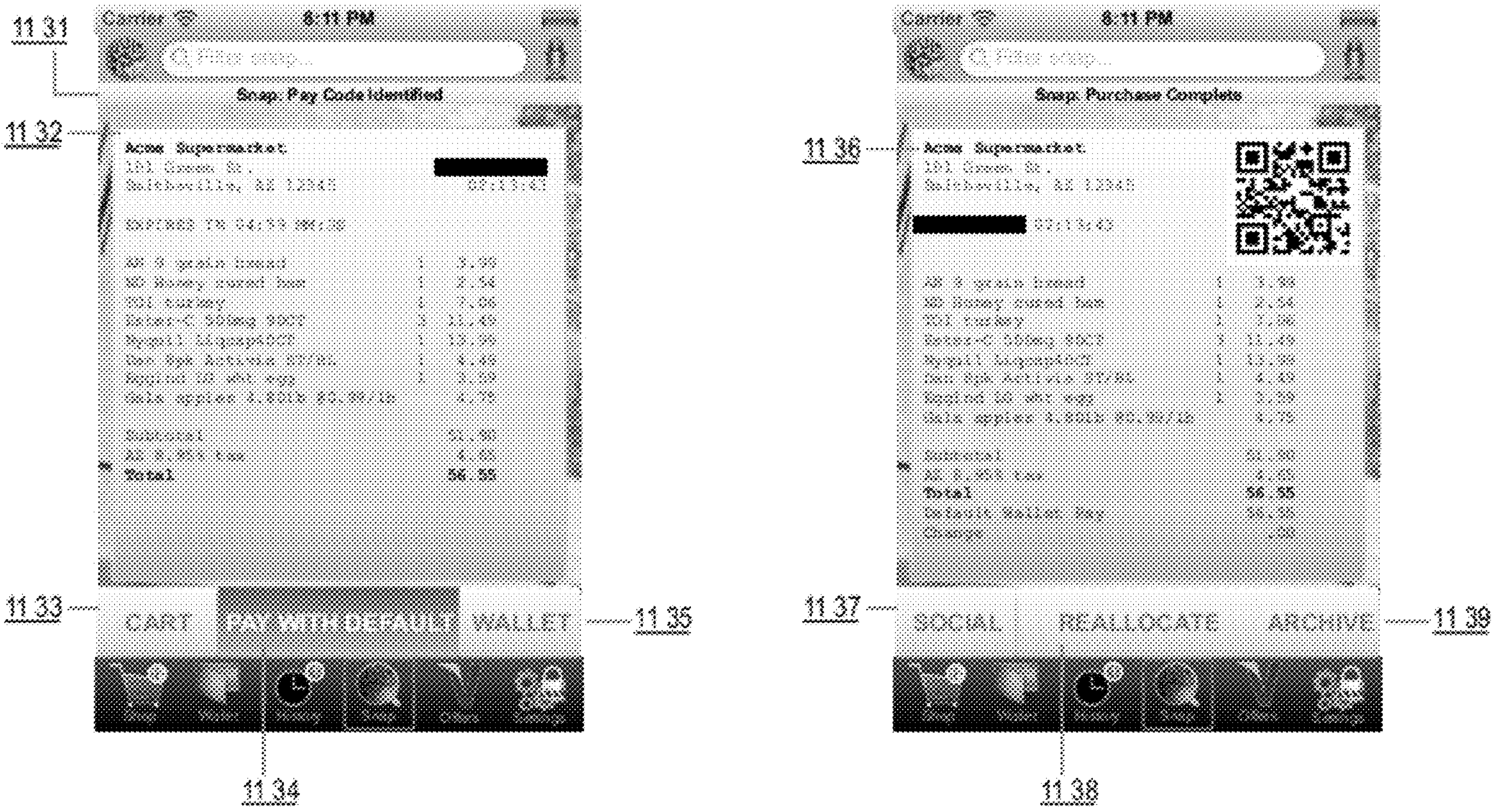


Snap: Payments Optimally Reallocated

```
                    REALLOCATED!
Acme Supermarket
101 Green St.,
Smithville, AS 12345
[redacted] 02:13:43
AN 9 grain bread                 1      3.99
HD Honey cured ham               1      2.54
TOI turkey                       1      7.06
Dan 6pk Activia 6T/6L            1      4.49
Eggland LG wht egg               1      3.59
Gala apples 1.01lb $0.99/lb             4.75
--------------------------------------------
Subtotal                               26.42
AZ 8.95% tax                            2.36
Total                                  28.78
Visa **************1234                28.78
--------------------------------------------
Rotex-C 500mg 30CT               1     11.49
Nyquil Liquid4OCT                1     13.99
Total                                  25.48
FSA **************5678                 25.48
```

FIGURE 11B

Example: Virtual Wallet Mobile App - Snap Mode



Snap: Pay Code Identified

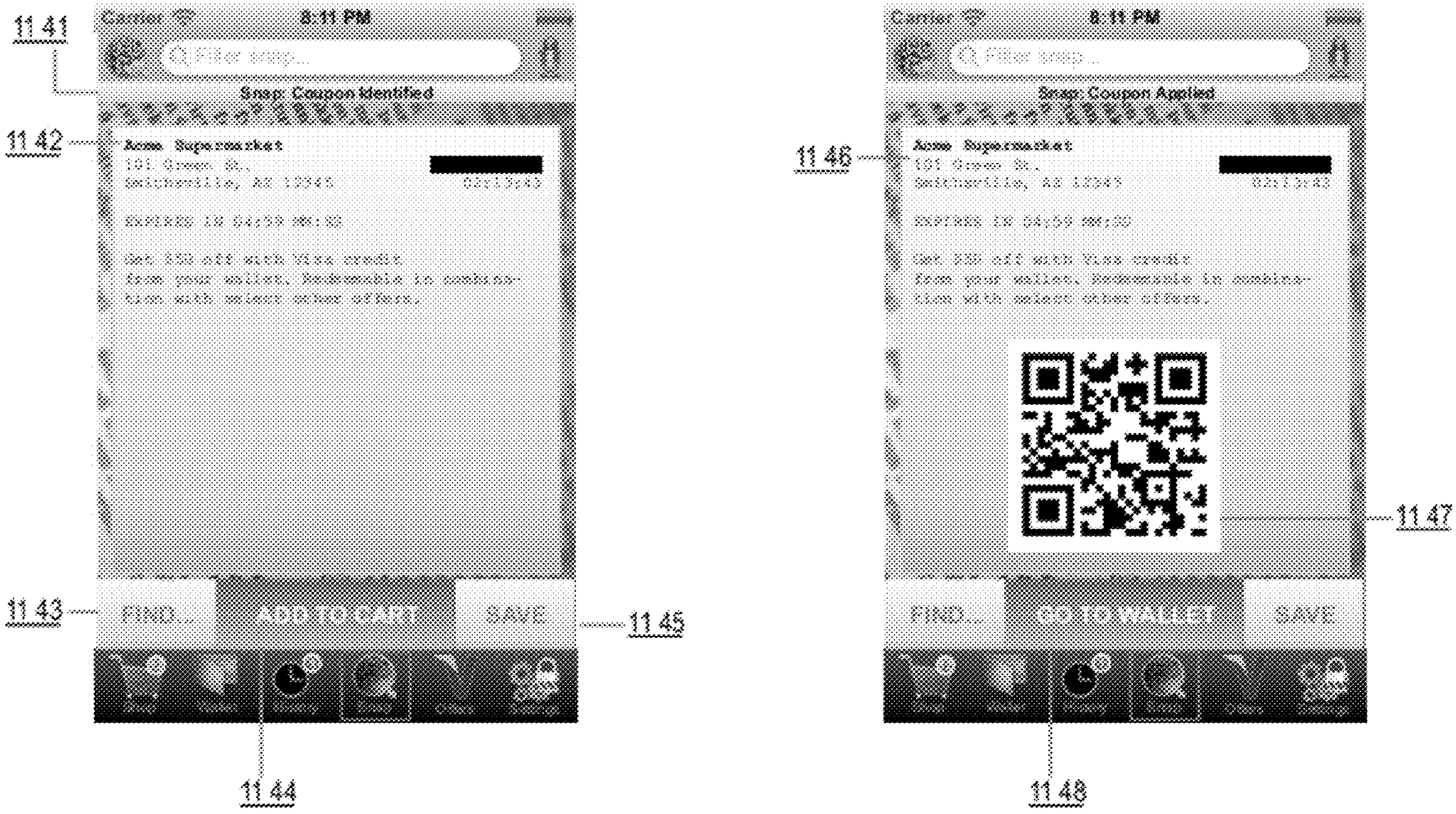
```
Acme Supermarket
101 Green St.                         [redacted]
Smithsville, AZ 12345                   02:13:43

EXPIRES IN 04:59 MM:SS

AN 9 grain bread               1      3.99
BD Honey cured ham             1      2.54
DDI turkey                     1      7.06
Ester-C 500mg 90CT             1     11.49
Nyquil LiquicpADCT             1     13.99
Dan Ygt Activia 8T/BL          1      4.49
Eggland LG wht egg             1      3.59
Gala apples 1.60lb $2.99/lb           4.79

Subtotal                             51.90
AZ 8.95% tax                          4.65
Total                                56.55
```

Snap: Purchase Complete

```
Acme Supermarket
101 Green St.
Smithsville, AZ 12345

[redacted] 02:13:43

AN 9 grain bread               1      3.99
BD Honey cured ham             1      2.54
DDI turkey                     1      7.06
Ester-C 500mg 90CT             1     11.49
Nyquil LiquicpADCT             1     13.99
Dan Ygt Activia 8T/BL          1      4.49
Eggland LG wht egg             1      3.59
Gala apples 1.60lb $2.99/lb           4.79

Subtotal                             51.90
AZ 8.95% tax                          4.65
Total                                56.55
Default Wallet Pay                   56.55
Change                                 .00
```

FIGURE 11C

Example: Virtual Wallet Mobile App - Snap Mode



Carrier 8:11 PM

Q Filter snap...

Snap: Coupon Identified

Acme Supermarket
101 Green St.
Smithville, AZ 12345
[redacted]
02:13:43

EXPIRES IN 04:59 MM:SS

Get $50 off with Visa credit from your wallet. Redeemable in combination with select other offers.

FIND... ADD TO CART SAVE

Carrier 8:11 PM

Q Filter snap...

Snap: Coupon Applied

Acme Supermarket
101 Green St.
Smithville, AZ 12345
[redacted]
02:13:43

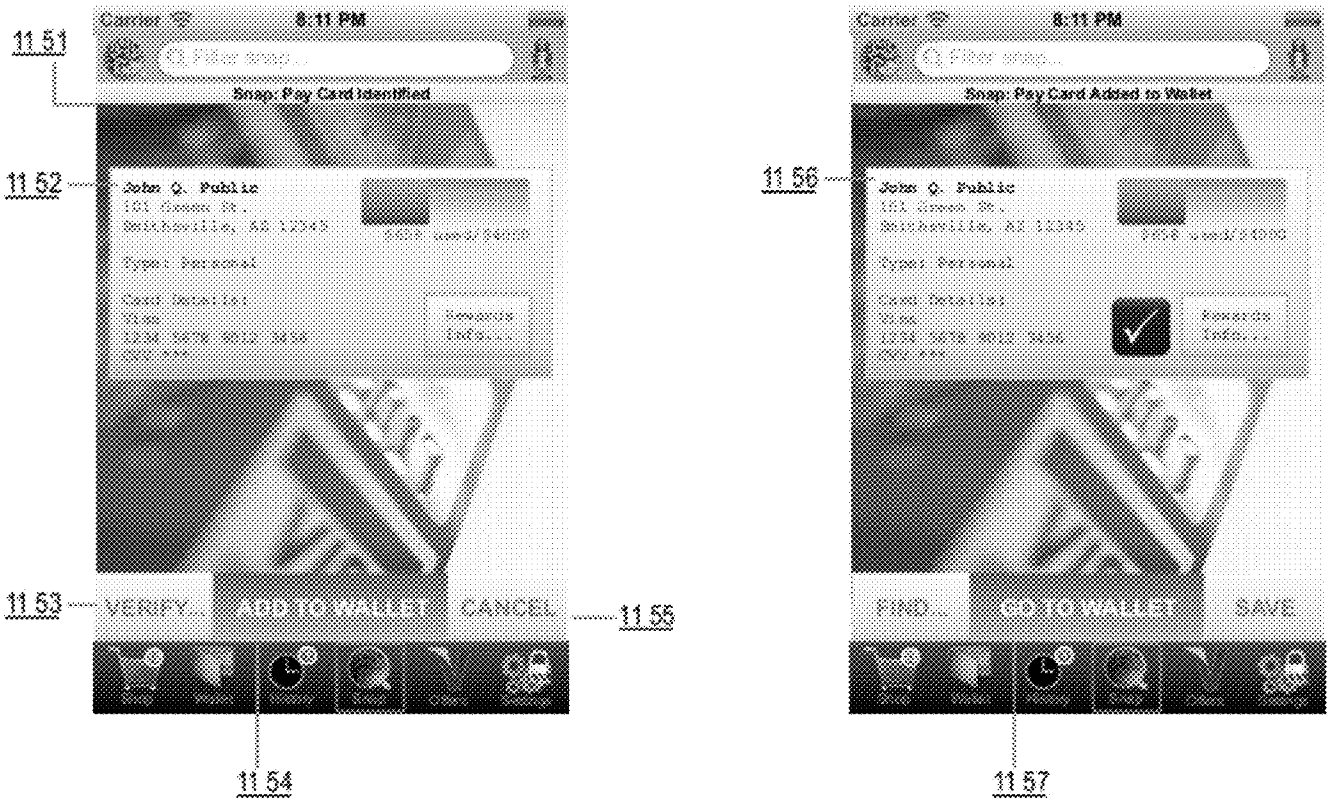
EXPIRES IN 04:59 MM:SS

Get $50 off with Visa credit from your wallet. Redeemable in combination with select other offers.

FIND... GO TO WALLET SAVE

FIGURE 11D

Example: Virtual Wallet Mobile App - Snap Mode


Carrier 8:11 PM

Snap: Pay Card Identified

John Q. Public
101 Green St.
Smithsville, AZ 12345

$450 used/$4000

Type: Personal

Card Details:
Visa
1234 5678 9012 3456

Carrier 8:11 PM

Snap: Pay Card Added to Wallet

John Q. Public
101 Green St.
Smithsville, AZ 12345

$450 used/$4000

Type: Personal

Card Details:
Visa
1234 5678 9012 3456

FIGURE 11E

Example: Virtual Wallet Mobile App - Snap Mode

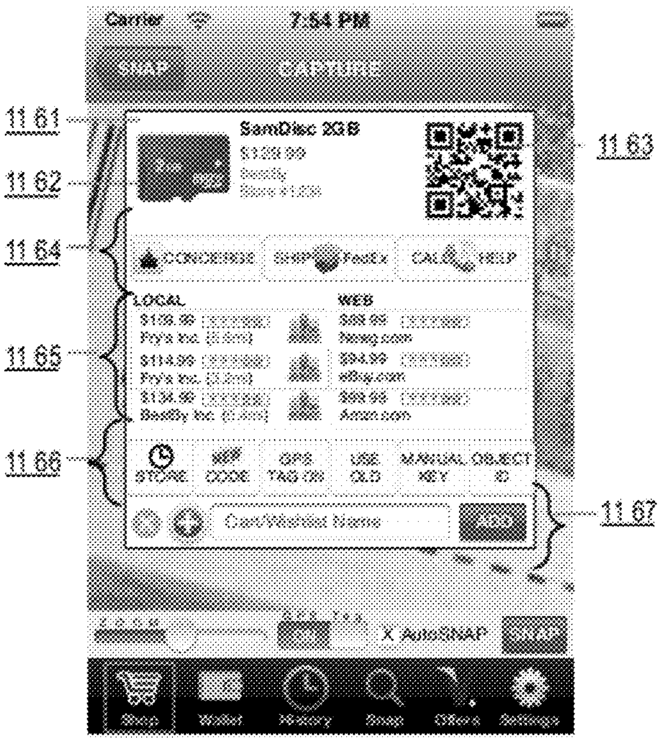
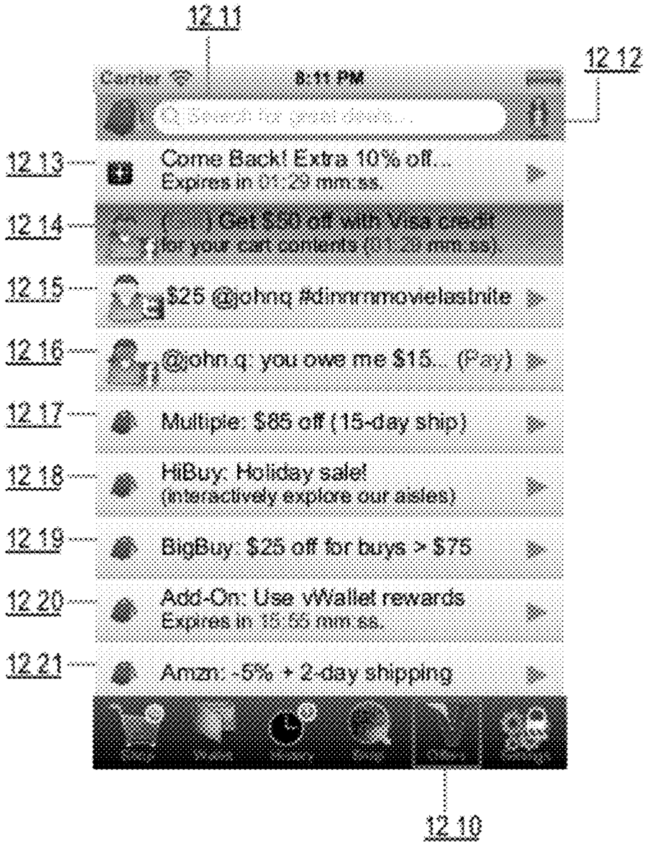


FIGURE 11F

Example: Virtual Wallet Mobile App - Snap Mode



Carrier 8:11 PM

Search for great deals...

Come Back! Extra 10% off...
Expires in 01:29 mm:ss.

Get $50 off with Visa credit
for your cart contents (01:25 mm:ss)

$25 @johnq #dinnrmovielastnite

@john.q: you owe me $15... (Pay)

Multiple: $85 off (15-day ship)

HiBuy: Holiday sale!
(interactively explore our aisles)

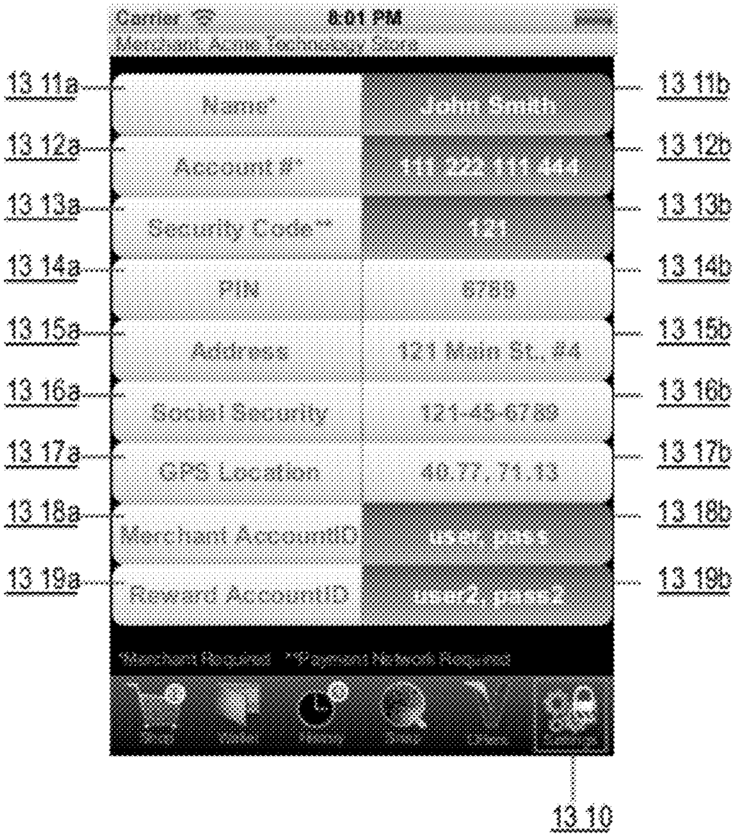
BigBuy: $25 off for buys > $75

Add-On: Use vWallet rewards
Expires in 15:55 mm:ss.

Amzn: -5% + 2-day shipping

FIGURE 12

Example: Virtual Wallet Mobile App - Offers



Carrier 8:01 PM

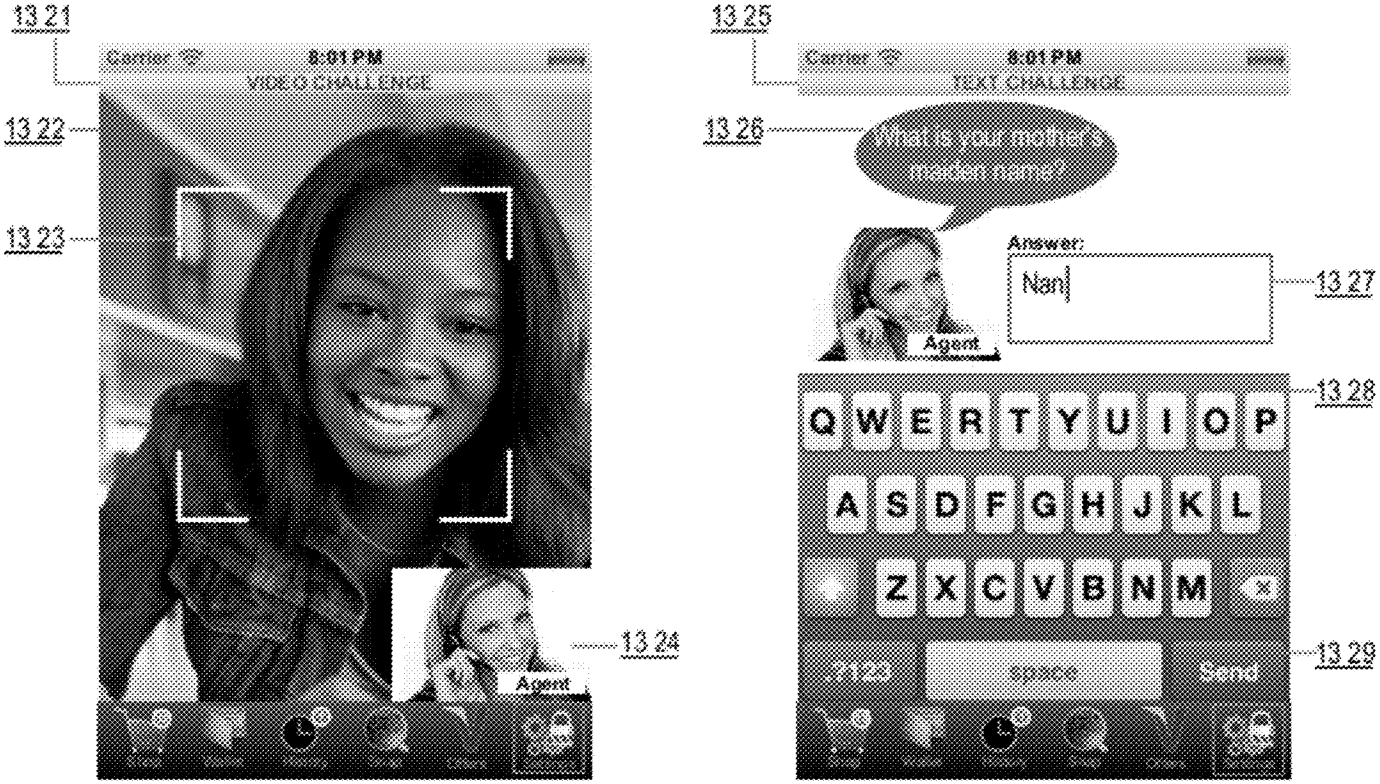
Merchant: Acme Technology Store

Name*	John Smith
Account #*	111 222 111 444
Security Code**	121
PIN	6789
Address	121 Main St., #4
Social Security	121-45-6789
GPS Location	40.77, 71.13
Merchant AccountID	user, pass
Reward AccountID	user2, pass2

*Merchant Required **Payment Network Required

FIGURE 13A

Example: Virtual Wallet Mobile App



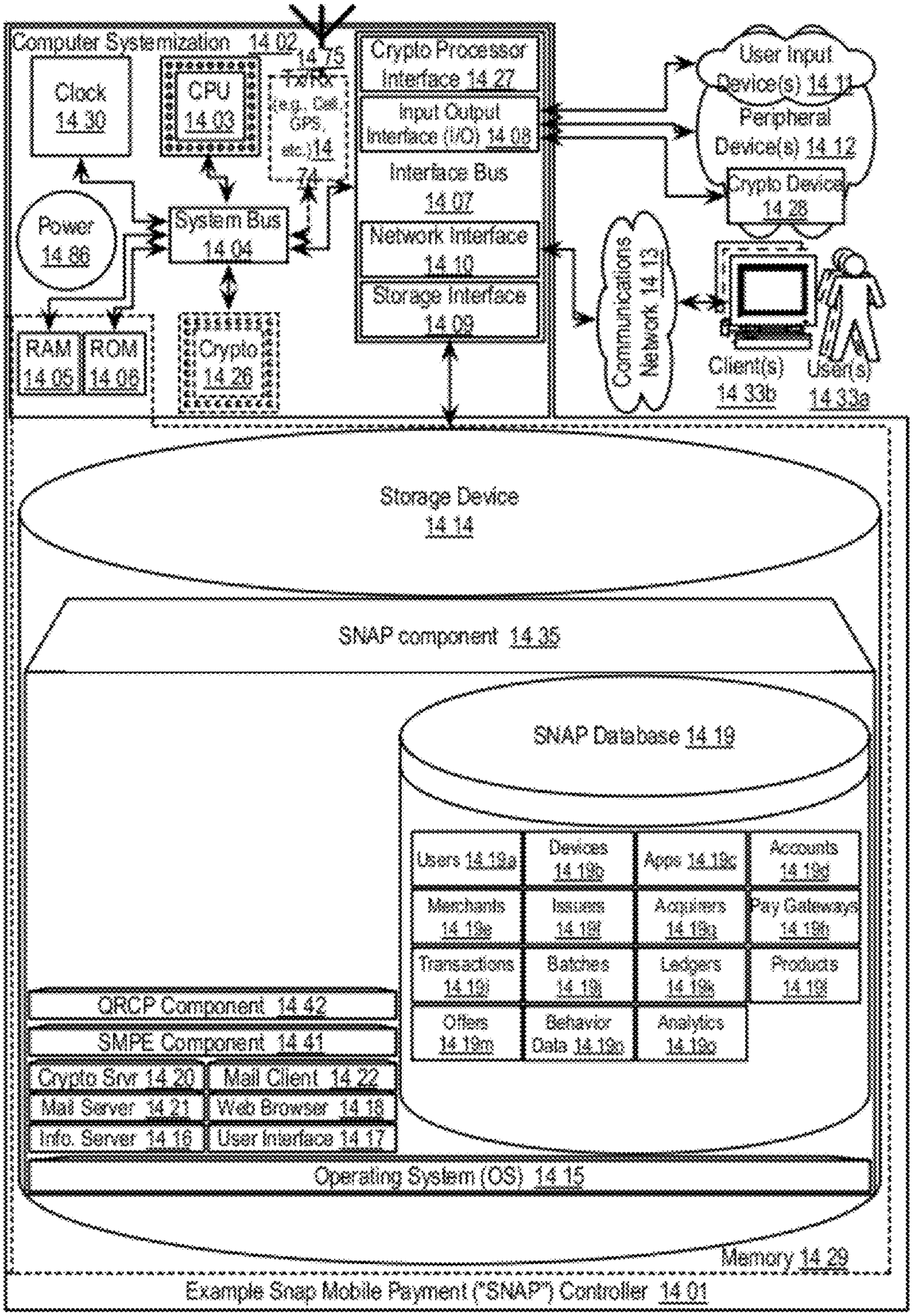
What is your mother's maiden name?

Answer:
Nan

FIGURE 13B

Example: Virtual Wallet Mobile App

FIGURE 14


Computer Systemization 14 02
14 75
Clock 14 30
CPU 14 03
Tx/Rx (e.g., Cell, GPS, etc.) 14 74
Crypto Processor Interface 14 27
Input Output Interface (I/O) 14 08
Interface Bus 14 07
Network Interface 14 10
Storage Interface 14 09
User Input Device(s) 14 11
Peripheral Device(s) 14 12
Crypto Device 14 28
Power 14 86
System Bus 14 04
Communications Network 14 13
Client(s) 14 33b
User(s) 14 33a
RAM 14 05
ROM 14 06
Crypto 14 26
Storage Device 14 14
SNAP component 14 35
SNAP Database 14 19
Users 14 19a
Devices 14 19b
Apps 14 19c
Accounts 14 19d
Merchants 14 19e
Issuers 14 19f
Acquirers 14 19g
Pay Gateways 14 19h
Transactions 14 19i
Batches 14 19j
Ledgers 14 19k
Products 14 19l
Offers 14 19m
Behavior Data 14 19n
Analytics 14 19o
QRCP Component 14 42
SMPE Component 14 41
Crypto Srvr 14 20
Mail Client 14 22
Mail Server 14 21
Web Browser 14 18
Info. Server 14 16
User Interface 14 17
Operating System (OS) 14 15
Memory 14 29
Example Snap Mobile Payment ("SNAP") Controller 14 01

SNAP MOBILE PAYMENT APPARATUSES, METHODS AND SYSTEMS

PRIORITY CLAIM

[0001] This application claims priority under 35 USC §119 to: U.S. provisional patent application Ser. No. 61/443,624 filed Feb. 16, 2011, entitled "MOBILE CAPTURE CHECKOUT APPARATUSES, METHODS AND SYSTEMS," attorney docket no. P-42032PRV|20270-127PV; U.S. provisional patent application Ser. No. 61/512,248 filed Jul. 2, 2011, entitled "SNAP MOBILE PAYMENT APPARATUSES, METHODS AND SYSTEMS," attorney docket no. 10US01|20270-175PV; U.S. provisional patent application Ser. No. 61/522,213 filed Aug. 10, 2011, entitled "UNIVERSAL MOBILE PAYMENT PLATFORM APPARATUSES, METHODS AND SYSTEMS," attorney docket no. 10US03|20270-175PV2; and U.S. provisional patent application Ser. No. 61/527,576 filed Aug. 25, 2011, entitled "SNAP MOBILE PAYMENT APPARATUSES, METHODS AND SYSTEMS," attorney docket no. 10US02|20270-175PV1. The entire contents of the aforementioned applications are expressly incorporated by reference herein.

[0002] 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.

FIELD

[0003] The present inventions are directed generally to apparatuses, methods, and systems for electronic purchase transactions, and more particularly, to SNAP MOBILE PAYMENT APPARATUSES, METHODS AND SYSTEMS ("SNAP").

BACKGROUND

[0004] Consumer transactions typically require a customer to select a product from a store shelf or website, and then to check the out at a checkout counter or webpage. Product information is typically selected from a webpage catalog or entered into a point-of-sale terminal device, or the information is automatically entered by scanning an item barcode with an integrated barcode scanner, and the customer is usually provided with a number of payment options, such as cash, check, credit card or debit card. Once payment is made and approved, the point-of-sale terminal memorializes the transaction in the merchant's computer system, and a receipt is generated indicating the satisfactory consummation of the transaction.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] The accompanying appendices and/or drawings illustrate various non-limiting, example, inventive aspects in accordance with the present disclosure:

[0006] FIGS. **1**A-F show block diagrams illustrating example aspects of a snap mobile payment-based purchase transaction in some embodiments of the SNAP;

[0007] FIGS. **2**A-F shows application user interface diagrams illustrating example features of a snap mobile payment app facilitating snap mobile payment in some embodiments of the SNAP;

[0008] FIGS. **3**A-E show application user interface diagrams illustrating example features of a snap mobile payment app for capturing product barcodes, securing user data and preventing fraud in some embodiments of the SNAP;

[0009] FIGS. **4**A-D show data flow diagrams illustrating an example snap mobile payment procedure in some embodiments of the SNAP;

[0010] FIGS. **5**A-E show logic flow diagrams illustrating example aspects of executing a snap mobile payment in some embodiments of the SNAP, e.g., a Snap Mobile Payment Execution ("SMPE") component **500**;

[0011] FIGS. **6**A-B show logic flow diagrams illustrating example aspects of processing a Quick Response code in some embodiments of the SNAP, e.g., a Quick Response Code Processing ("QRCP") component **600**;

[0012] FIG. **7** shows a user interface diagram illustrating an overview of example features of virtual wallet applications in some embodiments of the SNAP;

[0013] FIGS. **8**A-G show user interface diagrams illustrating example features of virtual wallet applications in a shopping mode, in some embodiments of the SNAP;

[0014] FIGS. **9**A-F show user interface diagrams illustrating example features of virtual wallet applications in a payment mode, in some embodiments of the SNAP;

[0015] FIG. **10** shows a user interface diagram illustrating example features of virtual wallet applications, in a history mode, in some embodiments of the SNAP;

[0016] FIGS. **11**A-F show user interface diagrams illustrating example features of virtual wallet applications in a snap mode, in some embodiments of the SNAP;

[0017] FIG. **12** shows a user interface diagram illustrating example features of virtual wallet applications, in an offers mode, in some embodiments of the SNAP;

[0018] FIGS. **13**A-B show user interface diagrams illustrating example features of virtual wallet applications, in a security and privacy mode, in some embodiments of the SNAP;

[0019] FIG. **14** shows a block diagram illustrating embodiments of a SNAP controller.

[0020] The leading number of each reference number within the drawings indicates the figure in which that reference number is introduced and/or detailed. As such, a detailed discussion of reference number **101** would be found and/or introduced in FIG. **1**. Reference number **201** is introduced in FIG. **2**, etc.

DETAILED DESCRIPTION

Snap Mobile Payment (SNAP)

[0021] The SNAP MOBILE PAYMENT APPARATUSES, METHODS AND SYSTEMS (hereinafter "SNAP") transform real-time-generated merchant-product Quick Response codes, via SNAP components, into virtual wallet card-based transaction purchase notifications. FIGS. **1**A-F show block diagrams illustrating example aspects of a snap mobile payment-based purchase transaction in some embodiments of the SNAP. With reference to FIG. **1**A, in some implementations, a user, e.g., iota-b, may wish to purchase products at a merchant store, e.g., **103***a*, or at a merchant website, e.g., **103***b*. For example, at a merchant store, the user may scan barcodes for a number of products, e.g., iota, at a point-of-sale ("POS") terminal in the store, e.g., **103***a*, and then indicate that the user wishes to checkout the scanned items. In some implementations, the POS terminal may generate a Quick Response ("QR") code, e.g., **105***a*, including information on the scanned product items, as well as merchant information for processing the purchase transaction via a payment network. The user may capture an image of the QR code generated by the POS terminal using a user device, such as a smartphone. For example, the user device may have executing on it an app for snapping the merchant-product QR code. The user device may utilize the information extracted from the QR code, along with information on a virtual wallet tied to the user device to initiate a purchase transaction. For example, the user device may utilize the product and merchant information extracted from the QR code, and financial payment information from the virtual wallet, to create a purchase transaction request, and submit the request to a payment network (e.g., credit card processing network).

[0022] In some implementations, the user device may utilize methods alternative to capture of a QR code to obtain information from the POS terminal. For example, the POS terminal may communicate the information required for submitting a purchase transaction request to a payment network to user device via Bluetooth™, Wi-Fi, SMS, text message, electronic mail, and/or other communication methods.

[0023] In some implementations, a user **101***b* may wish to checkout items stored in a virtual shopping cart on an online shopping website, e.g., **102***b*. For example, the user may be viewing the website using a secure display (e.g., that is part of a trusted computing device of the user). Upon indicating that the user wishes to checkout the items in the virtual shopping cart, the website may provide a QR code including information on the products in the virtual shopping cart and merchant information. For example, in the scenario where the user utilizes a secure display, the QR code may be displayed at a random position within the secure display for security purposes. The user may capture a snapshot of the displayed QR code, and utilize payment information from the virtual wallet associated with the user device to create a purchase transaction request for processing by the payment network. Upon completion of the purchase transaction, the payment network may provide a purchase receipt, e.g., **107** directly to the user device **106**, the POS terminal in the store and/or the secure display (for the secure online shopping scenario) as confirmation of completion of transaction processing. Thus, in some implementations, the merchant may be shielded from obtaining personal and/or private information of the user while processing the purchase transaction, while ensuring integrity of the user's virtual wallet using a secure display for presenting the merchant-product QR code.

[0024] In various implementations, such payment processing may be utilized for a wide variety of transactions. For example, a user dining at a restaurant may obtain a bill including a QR pay code including detail on the dining charges included in the bill, and a merchant ID for the restaurant. The user may take a snapshot of the restaurant bill using the user's smartphone, and utilize the user's virtual wallet to pay for the restaurant bill, without revealing any financial or personal information about the user to the restaurant.

[0025] With reference to FIG. **1**B, in some implementations, e.g., **110**, a user **111** may wish to checkout items stored in a (virtual) shopping cart in a (online) shopping store, e.g., **112**, using a reverse snap mobile payment procedure. For example, the user may be viewing the website using a secure display that is part of a trusted computing device of the user, e.g., **113**, or via a POS terminal in a brick-and-mortar store. Upon indicating that the user wishes to checkout the items in the virtual shopping cart, the user may generate, e.g., **114**, a QR code **115***b* via a mobile app on the user's mobile device including information on the user payment methods, offers, rewards, and/or other aspects connected to the user's virtual wallet. The user may provide the QR code displayed on the user's mobile device to a webcam (or other QR code capture device and/or mechanism) installed on the trusted computing device (or POS terminal). The user's trusted computing device or POS terminal may obtain a snapshot of the QR code generated by the user's mobile device, e.g., **116**, and utilize payment information from QR code generated by the user to create a purchase transaction request for processing by the payment network. Upon completion of the purchase transaction, the payment network may provide a purchase receipt directly to the user mobile device, the POS terminal in the store and/or the secure display (for the online shopping scenario) as confirmation of completion of transaction processing. Thus, in some implementations, the user may be able to utilize a QR code generated by the user's mobile device as a replacement for a plastic payment card (e.g., credit, debit, prepaid card), or as a substitute for other financial information transport mechanisms such as near-field communications, Bluetooth®, etc. In some implementations, the QR code may be representative of a one-time anonymized credit card number (e.g., see the description associated with FIG. **3**B).

[0026] In some implementations, a first user **121***b* may desire to pay a second user **121***a* an amount of money (or a value equivalent, e.g., virtual currency, alternate real currency, rewards, miles, points, etc.), e.g., P2P snap mobile payment **120**. The second user **121***a* may generate a limited-time-validity QR code, e.g., **122**, including information on the amount of money to be transferred, as well as a privacy token/alias linked to a financial account of the second user. The second user may display the QR code generated to the first user (e.g., by holding the second user's mobile phone displaying the QR code to the first user; sending the QR code by email, social network message, tweet, etc.). The first user may take a snapshot of the QR code using the first user's mobile phone, e.g., **123**, and utilize the amount of money, the second user's privacy token/alias linking to a financial account, and the first user's virtual wallet linked to the first user's mobile phone, to generate a purchase transaction request for processing by the payment network. Upon completion of the transaction, the payment network may provide transaction notification receipts to the users who are parties to the transaction. In alternate implementations, the two users may share the data encoded in the QR code via methods alternate to the QR code, including but not limited to: near-field communications (NFC), Wi-Fi™, Bluetooth™, cellular network, SMS, email, text messages and/or the other communication protocols.

[0027] In general, it is to be understood that such tokens, alias and/or handles may be advantageously utilized in various implementations of snap mobile payment. For example, a user wishing to engage in reverse snap mobile payment procedure (see, e.g., FIG. **1**B, element no) may generate a QR code embodying information on a handle pointing to financial payment information stored on a server of a payment network system. For example, some implementations of snap mobile payment may utilize, to generate and/or process handles, a payment tokenization procedure similar to that described in

U.S. application Ser. No. 13/153,301, titled "Payment tokenization apparatuses, methods and systems," the entire contents of which are expressly incorporated by reference herein. Further, in some implementations, the handle may encode information according to a compact messaging protocol, such as described in U.S. Pat. No. 6,837,425, titled "Compact protocol and solution for substantially offline messaging between portable consumer device and based device," the entire contents of which are expressly incorporated by reference herein. In some reverse snap mobile implementations, the user may provide the QR code embodying the handle and displayed on the user's mobile device to a webcam (or other QR code capture device and/or mechanism) installed on the trusted computing device (or POS terminal). The user's trusted computing device or POS terminal may obtain a snapshot of the QR code generated by the user's mobile device, e.g., **116**, and provide the handle extracted from the QR code to a merchant server for purchase transaction request processing by the payment network. The merchant server may generate a card authorization request (such as described further below in the discussion with reference to FIG. **4**A) for processing the purchase transaction using the handle, and may provide the card authorization request to a payment network. Upon completion of the purchase transaction, the payment network may provide a purchase receipt directly to the user mobile device, the POS terminal in the store, and/or the secure display (e.g., for the online shopping scenario) as confirmation of completion of transaction processing using the handle.

[0028] In some implementations, a user alert mechanism may be built into the snap mobile payment purchase transaction process flow. For example, in some implementations, a merchant server may embed a URL specific to the transaction into the card authorization request. For example, in some implementations, a POS terminal, remote device and/or desktop computer may embed the URL into optional level 3 data in the card authorization request. The URL may point to a webpage stored on the merchant's server dedicated to the transaction that is the subject of the card authorization request. For example, the webpage pointed to by the URL may include details on the purchase transaction, e.g., products being purchased, purchase cost, time expiry, status of order processing, and/or the like. Thus, the merchant server may provide to the payment network the details of the transaction by passing the URL of the webpage to the payment network. In some implementations, the payment network may provide notifications to the user, such as a payment receipt, transaction authorization confirmation message, shipping notification and/or the like. In such messages, the payment network may provide the URL to the user device. The user may navigate to the URL on the user's device to obtain alerts regarding the user's purchase, as well as other information such as offers, coupons, related products, rewards notifications, and/or the like.

[0029] In some implementations, a number of users may engage in group pay via snap mobile payment to split a tender, e.g., **130**. In some implementations, one of the users **131***a* may obtain a snapshot, e.g., **132**, of a QR pay code, e.g., **134**, generated at a POS terminal (or, e.g., presented on paper such as a dining bill), e.g., **133**. The user may in turn generate a QR split pay code, embodying information on the amounts that the tender has been split into. The user **131***a* may present the split tender QR code **135** to the other users **131***b-c*, who may obtain snapshots of the split tender QR code, e.g., **136**. In some implementations, the users **131***b-c* may be reimbursing the user **131***a* for payment of the original QR code via the payment network, or the users **131***b-c* may be making direct payments via the split tender QR code to the merchant (e.g., when the user **131***a* took a snapshot of the merchant's QR code, no payment processing occurred immediately). In some implementations, the merchant may directly provide a split tender QR code for the users **131***a-c*.

[0030] In some implementations, group mobile payment may be implemented, instead of utilizing QR codes, via use of an alternate communication mechanism. For example, in some implementations, the POS terminal **133** may utilize a communication protocol such as Bluetooth™ to communicate with the users **131***a-c*. The POS terminal may, serially or in parallel, establish separate communication sessions with each of the users. Via the separate communication sessions that POS terminal may transmit the product and/or merchant data required by the users' devices to generate individual purchase transaction processing requests. Thus, via the separate communication sessions, the POS terminal may split the group tender associated with the users **131***a-c* into individual payment amounts.

[0031] With reference to FIG. **1**C, in some implementations, snap mobile payment methods may be utilized for authentication/verification purposes, and for providing digital consent for disclosure of personal and/or private information. For example, a user **142** visiting his/her doctor **143** may be required to provide informed consent to disclosing personal information (e.g., medical records) to the doctor. The doctor's terminal may generate a QR code embodying the doctor's digital certificate as well as information on the type/content of medical records of the user that are requested, e.g., **144**. The user may snap the QR code via the user's mobile device. The user's mobile device may generate a request for records release according to the QR code, and also serve as verification that the request is obtained from a personal trusted device (e.g., the user's mobile device). In alternate implementations, the user may be able to select the personal information that the user would like to reveal to the healthcare provider, and the user's mobile device may generate a QR code for the doctor's terminal to obtain a snapshot for retrieving the user's medical information. In some implementations, the QR code may also include payment information (e.g., the user's pay account information, or the doctor's acquirer information) along with the information on controlled release of personal information.

[0032] In some implementations, the SNAP may facilitate P2P transactions via pre-filled, modifiable QR payment codes, e.g., **150**. For example, a first user having a public profile page, e.g., **151**, may place an image of a QR code in the public profile, e.g., **152**. For example, the QR code may include a predetermined payment amount for a purchase transaction initiated by capturing a snapshot of the QR code. In some implementations, the predetermined amount may be $0 (e.g., a $0 QR pay code). A second user may capture a snapshot of the QR pay code using a mobile device, and may set an amount that the second user would like to pay the first user via the second user's mobile device. The second user's mobile device may provide the information encoded within the QR code along with the second-user-chosen payment amount to a payment network for transaction processing.

[0033] It is to be understood that the various aspects described herein of snap mobile payment may be utilized for any controlled exchange of information and/or payment. For example, with reference to FIG. **1**D, in some implementations, a user may obtain pay-per-view programming via snap mobile payment, e.g., **160**. For example, a television display may provide an advertisement including programming information, e.g., **162**, as well as a QR pay code for obtain the programming content, e.g., **161**. The QR code may include information identifying the programming information, as well as information identifying the television subscriber account information, television machine address, and/or the like. The user may obtain a snapshot of the QR code, and provide the information embodied in the QR code along with information fro the user's mobile device (e.g., subscriber account number linked to the user's virtual wallet, pay account information, and/or the like). Upon processing of the payment information by the payment network, the payment network may provide an indication to the television-programming provider of the payment completion, and the television-programming provider may stream the programming content to the user's television. As another example, a similar flow may be utilized for in-flight entertainment, e.g., **170**, wherein an in-flight screen may provide programming information **172**, as well as a QR pay code **171** for the user to snap for in-flight entertainment initiation. As another example, a billboard, wall hanging, poster, in-store advertisement, hoarding, etc., e.g., **180**, may include an offer for a product/service, and a QR code including merchant information and product information identifying a purchase amount, and/or the like. The user may snap the QR code with the user's mobile device linked to the user's virtual wallet to purchase the product and/or service, and, if applicable, the product may be directly shipped to the user's address as specified by the purchase information exchanged with the payment network as part of the purchase request sent by the user's mobile device. As another example, newspapers, e.g., **185**, may include offers, advertisements, job postings, and/or the like including QR codes, e.g., **186**, embodying the information necessary for the user to initiate a purchase transaction with the payment network. It is to be understood that any aspects of implementing snap mobile payment discussed in any of the implementations herein, and/or their equivalents, may be utilized in any other implementations discussed herein, and/or their equivalents.

[0034] With references to FIGS. **1**E-F, in some implementations, the data required for processing a purchase transaction may be provided via methods alternate to a QR code including, but not limited to: near-field communications (NFC), Wi-Fi™, Bluetooth™, cellular network, SMS, email, text messages and/or the other communication protocols. For example, in some implementations, a user shopping online via a web browser executing on a client device, e.g., **190**, may desire to pay for a purchase of items from an online shopping website, e.g., **191**. The website may include a user interface element that the user may activate to initiate shopping checkout and payment. Upon the user activating the user element, the client displaying the online shopping website may provide a message to a server of the merchant to initiate secure purchase transaction processing. The server of the merchant operating the online shopping website may establish a secure connection (e.g., a Secure Sockets Layer connection) to a pay network server of a payment network, e.g., **192**. Also, the pay network server may establish a secure connection to the client. For example, the client may include a secure I/O chip that only allows secure connections to be established by the client with pay network servers of the payment network. Via the secure connection, the pay network server may provide an instruction to the client to request the user to launch a virtual wallet mobile app on the user device of the user, see e.g., FIG. **1**F, **196**. The client may accordingly provide a request to the user to launch a virtual wallet mobile app on the user device, e.g., **193**, of the user. Upon the user launching the virtual wallet mobile app on the user device, the user device and the client may establish a secure connection with each other (e.g., via Bluetooth™, Wi-Fi, cellular, etc.) In some implementations, the client and user device may be preconfigured to rapidly establish the secure communication channel with each other. Via the secure communication channel, the client may provide data to the user's mobile device, or vice versa, to facilitate initiation of the purchase transaction. The virtual wallet app on the user's mobile device (or the client) may then generate a purchase transaction initiation message and provide it to the pay network server for processing the purchase transaction. Upon completion of transaction processing, the pay network server may provide a notification of payment completion to the client, e.g., FIG. **1**F, **197**, or to the user device.

[0035] FIGS. **2**A-F show application user interface diagrams illustrating example features of a snap mobile payment app facilitating snap mobile payment in some embodiments of the SNAP. With reference to FIG. **2**A, in some implementations, a user may desire to checkout one or items stored in a virtual shopping cart of an online merchant website. For example, the user may be utilizing a browser application, e.g., **201**, to visualize a checkout page of the merchant website, e.g., **202**. The checkout webpage may depict details of the checkout order, e.g., **203**, and may provide one or more options for the user to provide payment for the purchase of the store items. In some implementations, the checkout webpage may include an option to pay for the purchase using a snap mobile payment procedure, e.g., **204**.

[0036] With reference to FIG. **2**B, in some implementations, upon selecting the option to utilize the snap mobile payment procedure, the merchant checkout webpage, e.g., **206**, may provide via the browser application **205**, a QR code, e.g., **209**, including information on the items in the virtual shopping cart as well as merchant information for the payment network to process the purchase transaction (e.g., a privacy token/alias linked to an acquirer financial account of the merchant). In some implementations, the webpage may be displayed via a secure display of a trusted computing device of the user. For example, as a security measure, the position of the QR code frame, e.g., **207**, within the display may be randomly varied to prevent a snapshot of the QR code from being obtained by fraudulent means (e.g., tampering with the trusted computing device). In some implementations, a security image, e.g., **208**, pre-selected by the user may be displayed on the screen so that the user may verify as being accurate. In some implementations, the image may be encrypted by the SNAP before providing it to the trusted computing device. In some implementations, the trusted computing device may be the only device to hold a decryption key required to decrypt and successfully display the image on the secure display to the user.

[0037] With reference to FIG. **2**C, in some implementations, such merchant-product information embodying QR codes may be utilized by a point-of-sale ("POS") terminal, e.g., **210***a-b*. For example, in a brick-and-mortar store, the POS terminal may display a QR code, e.g., **211***a-b*, that includes the purchase payment amount, e.g., **212***a-b*, upon the user indicating that the user wishes to checkout the items in the user's physical shopping cart. For example, the QR code may include data formatted according to the extensible Markup Language ("XML"), such as the example data structure below:

```
<QR__data>
    <order__ID>4NFU4RG94</order__ID>
    <timestamp>2011-02-22 15:22:43</timestamp>
    <expiry__lapse>00:00:30</expiry__lapse>
    <transaction__cost>$34.78</transaction__cost>
    <user__ID>john.q.public@gmail.com</user__ID>
    <client__details>
        <client__IP>192.168.23.126</client__IP>
        <client__type>smartphone</client__type>
        <client__model>HTC Hero</client__model>
        <OS>Android 2.2</OS>
        <app__installed__flag>true</app__installed__flag>
    </client__details>
    <secure__element>www.merchant.com/securedyn/0394733/123.png</secure__element>
    <purchase__details>
        <num__products>1</num__products>
        <product>
            <product__type>book</product__type>
            <product__params>
                <product__title>XML for dummies</product__title>
                <ISBN>938-2-14-168710-0</ISBN>
                <edition>2nd ed.</edition>
                <cover>hardbound</cover>
                <seller>bestbuybooks</seller>
            </product__params>
            <quantity>1</quantity>
        </product>
    </purchase__details>
    <merchant__params>
        <merchant__id>3FBCR4INC</merchant__id>
        <merchant__name>Books & Things, Inc.</merchant__name>
        <merchant__auth__key>1NNF484MCP59CHB27365</merchant__auth__key>
    </merchant__params>
<QR__data>
```

[0038] With reference to FIG. **2**D, in some implementations, the user may obtain a snapshot of the QR code displayed on the screen of the secure display or the POS terminal using a smartphone, e.g., **213**. For example, the user's smartphone may have an app, e.g., **214**, executing on it to detect and capture QR codes, e.g., **216***a*. For example, the user may utilize registration features, e.g., **215**, to align the QR code within the display of the smartphone. The app may, in some implementations, provide the user with the ability to zoom in, e.g., **217**, or zoom out, e.g., **218**, of the QR code to ensure that the image of the QR code fits within the dimensions of the screen of the smartphone. Upon aligning the QR code within the display of the smartphone, the user may be able to obtain a snapshot of the QR code using a user interface element, e.g., **219**. The user may cancel the snap mobile payment procedure using a user interface element **220** on the display of the smartphone.

[0039] With reference to FIG. **2**E, in some implementations, upon obtaining a snapshot of the merchant-product QR code, the user's smartphone may extract the product and merchant data stored within the QR code, and utilize an account for the user's virtual wallet linked to the user's smartphone to generate a purchase transaction request for processing by a payment network. Upon completion of processing of the payment transaction by the payment network using the information provided by the user's smartphone, the merchant website **222** (via the browser application **221**) may provide a purchase receipt **225** for the user. With reference to FIG. **2**F, in implementations where the user utilizes the snap mobile payment procedure at a brick-and-mortar store, the POS terminal may display a purchase receipt for the user. In some implementations, the payment network may provide a purchaser receipt directly to the smartphone of the user.

[0040] FIGS. **3**A-E show application user interface diagrams illustrating example features of a snap mobile payment app for capturing product barcodes, securing user data and preventing fraud in some embodiments of the SNAP. With reference to FIG. **3**A, in some implementations, the app executing on the device of the user may include an app interface providing various features for the user. In some implementations, the app may be configured to recognize product identifiers (e.g., barcodes, QR codes, etc.), e.g., **301**. For example, the app may be configured to capture merchant-product QR codes for snap mobile payment processing, as discussed above with reference to FIGS. **2**A-F. In some implementations, the user may be required to sign in to the app to enable its features. Once enabled, the camera may provide in-person one tap purchasing features for the user. For example, the client device may have a camera via which the app may acquire images, video data, streaming live video, and/or the like, e.g., **303**. The app may be configured to analyze the incoming data, and search, e.g., **301**, for a product identifier, e.g., **304**, such as QR codes **209**, **211***a-b*, **216***a* and **227**. In some implementations, the app may overlay cross-hairs, target box, and/or like alignment reference markers, e.g., **305**, so that a user may align the product identifier using the reference markers so facilitate product identifier recognition and interpretation. In some implementations, the app may include interface elements to allow the user to switch back and forth between the product identification mode and product offer interface display screens (see, e.g., **306**), so that a user may accurately study deals available to the user before capturing a product identifier. In some implementations, the app may provide the user with the ability to view prior product identifier captures (see, e.g., **307**) so that the user may be able to better decide which product identifier the user desires to capture. In some implementations, the user may desire to cancel product purchasing; the app may provide the user with a user interface element (e.g., **308**) to cancel the product identifier recognition procedure and return to the prior interface screen the user was utilizing. In some implementations, the user may be provided with information about products, user settings, merchants, offers, etc. in list form (see, e.g., **309**) so that the user may better understand the user's purchasing options. Various other features may be provided for in the app (see, e.g., **310**).

[0041] With reference to FIG. **3**B, in some implementations, the app may include an indication of the location (e.g., name of the merchant store, geographical location, information about the aisle within the merchant store, etc.) of the user, e.g., **311**. The app may provide an indication of a pay amount due for the purchase of the product, e.g., **312**. In some implementations, the app may provide various options for the user to pay the amount for purchasing the product(s). For example, the app may utilize the GPS coordinates to determine the merchant store within the user is present, and direct the user to a website of the merchant. In some implementations, the SNAP may provide an API for participating merchants directly to facilitate transaction processing. In some implementations, a merchant-branded SNAP application may be developed with the SNAP functionality, which may directly connect the user into the merchant's transaction processing system. For example, the user may choose from a number of cards (e.g., credit cards, debit cards, prepaid cards, etc.) from various card providers, e.g., **313**. In some implementations, the app may provide the user the option to pay the purchase amount using funds included in a bank account of the user, e.g., a checking, savings, money market, current account, etc., e.g., **314**. In some implementations, the user may have set default options for which card, bank account, etc. to use for the purchase transactions via the app. In some implementations, such setting of default options may allow the user to initiate the purchase transaction via a single click, tap, swipe, and/or other remedial user input action, e.g., **315***a*. In some implementations, when the user utilizes such an option, the app may utilize the default settings of the user to initiate the purchase transaction. In some implementations, the app may allow the user to utilize other accounts (e.g., Google™ Checkout, Paypal™ account, etc.) to pay for the purchase transaction, e.g., **316**. In some implementations, the app may allow the user to utilize rewards points, airline miles, hotel points, electronic coupons, printed coupons (e.g., by capturing the printed coupons similar to the product identifier) etc., to pay for the purchase transaction, e.g., **317-318**. In some implementations, the app may provide an option to provide express authorization before initiating the purchase transaction, e.g., **319**. In some implementations, the app may provide a progress indicator provide indication on the progress of the transaction after the user has selected an option to initiate the purchase transaction, e.g., **320**. In some implementations, the app may provide the user with historical information on the user's prior purchases via the app, e.g., **321**. In some implementations, the app may provide the user with an option to share information about the purchase (e.g., via email, SMS, wall posting on Facebook®, tweet on Twitter™, etc.) with other users and/or control information shared with the merchant, acquirer, payment network etc., to process the purchase transaction, e.g., **322**. In some implementations the app may provide the user an option to display the product identification information captured by the client device (e.g., in order to show a customer service representative at the exit of a store the product information), e.g., **324**. In some implementations, the user, app, device and or SNAP may encounter an error in the processing. In such scenarios, the user may be able to chat with a customer service representative (e.g., VerifyChat **323**) to resolve the difficulties in the purchase transaction procedure.

[0042] In some implementations, the user may select to conduct the transaction using a one-time anonymized credit card number, see e.g., **315***b*. For example, the SNAP may utilize a pre-designated anonymized set of card details (see, e.g., "AnonCard1," "AnonCard2"). As another example, the SNAP may generate, e.g., in real-time, a one-time anonymous set of card details to securely complete the purchase transaction (e.g., "Anon It 1X"). In such implementations, the app may automatically set the user profile settings such that the any personal identifying information of the user will not be provided to the merchant and/or other entities. In some implementations, the user may be required to enter a user name and password to enable the anonymization features.

[0043] With reference to FIG. **3**C, in some implementations, the user interface elements of the snap mobile payment app may be advantageously arranged to provide the user the ability to process a purchase with customized payment parameters with a minimum number of user gestures applied to the user's mobile device. For example, the user may be provided with an overloaded user interface element, e.g., **325-326**. For example, if the user has a QR pay code within the viewing angle of a camera included in the user's mobile device, the user may activate element **325** to snap the QR code and utilize predetermined default settings to process the purchase based on the QR code. However, if the user wishes to customize the payment parameters, the user may activate user interface element **326** (e.g., press and continue to hold). Upon doing so, the app may provide a pop-up menu, e.g., **327**, providing a variety of payment customization choices, such as those provided previously. The user may, e.g., drag the user's finger to the appropriate settings the user prefers, and release the user's finger from the touchscreen of the user's mobile device to select the setting for payment processing. In alternate implementations, the payment settings options, e.g., **330**, and QR capture activation buttons, e.g., **328***a-b* (e.g., **328***b* may provide even more settings that those displayed in the initial screen), may be included in the user interface along with a window, e.g., **329**, for capturing the QR code via the mobile device's camera. In alternate implementations, the user's mobile device may generate a hybrid QR code-payment settings graphic, and the POS terminal (or user's trusted computing device) may capture the entire graphic for payment processing.

[0044] With reference to FIG. **3**D, in some implementations, a user may be advantageously able to provide user settings into a device producing a QR code for a purchase transaction, and then capture the QR code using the user's mobile device. For example, a display device of a point-of-sale terminal may be displaying a checkout screen, such as a web browser executing on a client, e.g., **331**, displaying a checkout webpage of an online shopping website, e.g., **332**. In some implementations, the checkout screen may provide a user interface element, e.g., **333***a-b*, whereby the user can indicate the desire to utilize snap mobile payment. For example, if the user activates element **331***a*, the website may generate a QR code using default settings of the user, and display the QR code, e.g., **335**, on the screen of the client for the user to capture using the user's mobile device. In some implementations, the user may be able to activate a user interface element, e.g., **333***b*, whereby the client may display a pop-up menu, e.g., **334**, with additional options that the user may select from. For example, the website may provide user selection options similar to those discussed above in the description with reference to FIGS. **3**B-C. In some implementations, the website may modify the QR code **335** in real-time as the user modifies settings provided by activating the user interface element **333***b*. Once the user has modified the settings using the pop-up menu, the user may capture a snapshot of the QR code to initiate purchase transaction processing.

[0045] With reference to FIG. **3**E, in some implementations, the SNAP may provide the user with a user interface to modify the user's snap mobile payment settings. For example, the SNAP may provide a web interface, e.g., **341**. For example, the user may be able to modify security settings of the user's virtual wallet, e.g., **342**, using the web interface. For example, the user may review a list of trusted device, e.g., **344**, via which the user may access the user's virtual wallet. In some implementations, the web interface may provide a user interface element to add a trusted device, e.g., **343**. The web interface may also provide the user with additional security options. For example, the user be able to set a security passphrase, e.g., **345**, modify settings for when the user should be challenged before authorizing a purchase transaction, e.g., **346**, the type/style of presentation of the security features, e.g., **347**, and a security image to be displayed on the terminal utilized in snap mobile payment, e.g., **348**. In various implementations, the user may be able to access other services including modifying user profiles, accounts, account preferences, adding cards, obtaining offers and coupons, locating ATM machines, etc.

[0046] FIGS. **4**A-D show data flow diagrams illustrating an example snap mobile payment procedure in some embodiments of the SNAP. With reference to FIG. **4**A, in some implementations, a user, e.g., **401**, may desire to purchase a product, service, offering, and/or the like ("product"), from a merchant, e.g., **403**, via a merchant online site or in the merchant's store. The user may communicate with a merchant server, e.g., **403**, via a client such as, but not limited to: a personal computer, mobile device, television, point-of-sale terminal, kiosk, ATM, and/or the like (e.g., **402**). For example, the user may provide user input, e.g., checkout input **411**, into the client indicating the user's desire to purchase the product. For example, a user in a merchant store may scan a product barcode of the product via a barcode scanner at a point-of-sale terminal. As another example, the user may select a product from a webpage catalog on the merchant's website, and add the product to a virtual shopping cart on the merchant's website. The user may then indicate the user's desire to checkout the items in the (virtual) shopping cart. The client may generate a checkout request, e.g., **412**, and provide the checkout request, e.g., **413**, to the merchant server. For example, the client may provide a (Secure) Hypertext Transfer Protocol ("HTTP(S)") GET message including the product details for the merchant server in the form of data formatted according to the eXtensible Markup Language ("XML"). Below is an example HTTP(S) GET message including an XML-formatted checkout request for the merchant server:

```
GET /checkout.php HTTP/1.1
Host: www.merchant.com
Content-Type: Application/XML
Content-Length: 718
<?XML version = "1.0" encoding = "UTF-8"?>
<checkout_request>
    <session_ID>4NFU4RG94</session_ID>
    <timestamp>2011-02-22 15:22:43</timestamp>
    <user_ID>john.q.public@gmail.com</user_ID>
    <client_details>
        <client_IP>192.168.23.126</client_IP>
        <client_type>smartphone</client_type>
        <client_model>HTC Hero</client_model>
        <OS>Android 2.2</OS>
        <app_installed_flag>true</app_installed_flag>
    </client_details>
    <purchase_details>
        <num_products>1</num_products>
        <product>
            <product_type>book</product_type>
            <product_params>
                <product_title>XML for dummies</product_title>
                <ISBN>938-2-14-168710-0</ISBN>
                <edition>2nd ed.</edition>
                <cover>hardbound</cover>
                <seller>bestbuybooks</seller>
            </product_params>
            <quantity>1</quantity>
        </product>
    </purchase_details>
</checkout_request>
```

[0047] In some implementations, the merchant server may obtain the checkout request from the client, and extract the checkout detail (e.g., XML data) from the checkout request. For example, the merchant server may utilize a parser such as the example parsers described below in the discussion with reference to FIG. **14**. The merchant server may extract the product data, as well as the client data from the checkout request. In some implementations, the merchant server may query, e.g., **414**, a merchant database, e.g., **404**, to obtain product data, e.g., **415**, such as product pricing, sales tax, offers, discounts, rewards, and/or other information to process the purchase transaction. For example, the database may be a relational database responsive to Structured Query Language ("SQL") commands. The merchant server may execute a hypertext preprocessor ("PHP") script including SQL commands to query the database for product data. An example PHP/SQL command listing, illustrating substantive aspects of querying the database, is provided below:

```
<?PHP
header('Content-Type: text/plain');
mysql__connect("254.93.179.112",$DBserver,$password); // access
database server
mysql__select__db("PRODUCTS.SQL"); // select database table to search
//create query
$query = "SELECT product__info product__price tax__linfo__list offers__list
      discounts__list rewards__list FROM ProdTable WHERE product LIKE
      '%' $prod";
$result = mysql__query($query); // perform the search query
mysql__close("PRODUCTS.SQL"); // close database access
?>
```

[0048] In some implementations, in response to obtaining the product data, the merchant server may generate, e.g., **416***a*, a QR pay code, and/or secure display element according to the security settings of the user (see, e.g., **358**). The merchant server may provide the QR code to the client, so that the client may display the QR code, and the user may capture the QR code using the user's device to obtain merchant and/or product data for generating a purchase transaction processing request. In alternate implementations, the merchant server may direct the client to communicate the product and/or merchant data required to process the transaction to the user's device via an alternate communication protocol, such as, but not limited to: Wi-Fi™, Bluetooth™, cellular network, SMS, email and/or like communication protocols. For example, the merchant server may direct the client to initiate a plug-in on its system to provide the alternate communication service, and transmit the product and/or merchant data to the user's device via the communication service.

[0049] In implementations utilizing a QR code, the merchant server may generate a QR code embodying the product information, as well as merchant information required by a payment network to process the purchase transaction. In some implementations, the QR code may include at least information required by the user device capturing the QR code to generate a purchase transaction processing request, such as a merchant identifier (e.g., a merchant ID number, merchant name, store ID, etc.) and a session identifier for a user shopping session associated with the shopping website/brick-and-mortar store.

[0050] In some implementations, the merchant server may generate in real-time, a custom, user-specific merchant-product XML data structure having a time-limited validity period, such as the example 'QR_data' XML data structure provided below:

```
<QR__data>
    <order__ID>4NFU4RG94</order__ID>
    <timestamp>2011-02-22 15:22:43</timestamp>
    <expiry__lapse>00:00:30</expiry__lapse>
    <transaction__cost>$34.78</transaction__cost>
    <alerts__URL>www.merchant.com/shopcarts.php?sessionID=AEBB4356</alerts__URL>
    <user__ID>john.q.public@gmail.com</user__ID>
    <client__details>
        <client__IP>192.168.23.126</client__IP>
        <client__type>smartphone</client__type>
        <client__model>HTC Hero</client__model>
        <OS>Android 2.2</OS>
        <app__installed__flag>true</app__installed__flag>
    </client__details>
    <secure__element>www.merchant.com/securedyn/0394733/123.png</secure__element>
    <purchase__details>
        <num__products>1</num__products>
        <product>
            <product__type>book</product__type>
            <product__params>
                <product__title>XML for dummies</product__title>
                <ISBN>938-2-14-168710-0</ISBN>
                <edition>2nd ed.</edition>
                <cover>hardbound</cover>
                <seller>bestbuybooks</seller>
            </product__params>
            <quantity>1</quantity>
        </product>
    </purchase__details>
    <merchant__params>
        <merchant__id>3FBCR4INC</merchant__id>
        <merchant__name>Books & Things, Inc.</merchant__name>
        <merchant__auth__key>1NNF484MCP59CHB27365</merchant__auth__key>
    </merchant__params>
<QR__data>
```

[0051] 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 for generating QR Code, 2-dimensional barcode, available at http://phpqrcode.sourceforge.net/. For example, the merchant server may issue PHP commands similar to the example commands provided below:

```
<?PHP
header('Content-Type: text/plain');
// Create QR code image using data stored in $data variable
QRcode::png($data, ‘qrcodeimg.png’);
?>
```

[0052] In alternate implementations, the merchant server may provide, e.g., **416***b* the XML data to a pay network server, e.g., **406**, along with a request to generate a QR code. For example, the merchant server may utilize an API call to the pay network server to request generation of the QR code. The pay network server may generate the QR code for the merchant server, e.g., **416***c*, and provide, e.g., **416***d*, the QR code to the merchant server. For example, the pay network server may encode the information provided by the merchant into the QR code, and may also advantageously encode security information, time validity information, digital certificate information, anonymous shipping information, QR code generation/processing fee information, etc. into the QR code.

[0053] In some implementations, the pay network server may provide the merchant server with an encryption key (e.g., a Rivest-Shamir-Adleman (“RSA”) private/public key, digital certificate). The merchant may encrypt the custom, user-specific merchant-product XML data structure using the encryption key to generate encrypted purchase data (e.g., using the RSA algorithm). The merchant server may then encode the encrypted data into the QR code. Such a scheme may be employed advantageously, in various embodiments, by the pay network server to authenticate the merchant for any transaction processing requests related to the user-merchant shopping session.

[0054] In some implementations, pre-designed QR codes associated with authenticated with pre-authenticated merchants may be provided to the user device. For example, a user may be browsing an online website on the user's device. The user device may make a HTTP(S) GET request for a webpage from a web server. In some implementations, the web server may, in response to the user device's request for a webpage, generate a query for advertisements to display on the webpage. For example, the web server may search a database, or provide a request to an ad network server (e.g., Akamai) to provide advertisements for embedding into the webpage. In some implementations, the ad network server may utilize keywords, metadata, etc. obtained from the web server (e.g., keywords or metadata associated with the webpage, user profile information, user ID, user browsing history from a cookie stored on the user device, etc.). The ad network may utilize the keywords to generate a query of database(s) for advertisements associated with the keywords, and may obtain advertisements to provide. In some implementations, the ad network server may provide information (e.g., via an API call) on such advertisements (e.g., merchant name, merchant ID, product name, product pricing information, related offers, etc.) to a pay network server. The pay network server may generate a QR code based on the information provide by the ad network server, such that a user device may snap the QR code to initiate a purchase transaction for the goods and/or services associated with the QR code (e.g., as provided by the ad network server to the pay network server). The ad network server may provide the QR as part of the advertisement to the web server, which may in turn embed the advertisement including the QR code into the webpage before providing it to the user device. In alternate implementations, the ad network server/web server may transmit a URL or other identifier of the QR code (ultimately) to the user device, and the user device may make a call (e.g., a HTTP(S) GET request) using the URL of the QR code (e.g., hosted on the pay network server) to obtain the QR code and display it for the user.

[0055] In some implementations, the merchant server may provide the QR code to the client, e.g., **417**. For example, the merchant server may provide a HyperText Markup Language (“HTML”) page including a reference to the QR code image and/or secure element image, such as the example HTML page below:

```
<html>
    <img src="www.merchant.com/securedyn/0394733/qrcodeimg.png"
    alt=”Merchant- Product QR code”/>
    <img src=" www.merchant.com/securedyn/0394733/123.png"
    alt=”Secure Element”/>
</html>
```

[0056] In some implementations, the client may obtain the QR pay code, e.g., **417**, and display the QR code, e.g., **418** on a display screen associated with the client device. In some implementations, the user may utilize a user device, e.g., **405**, to capture the QR code presented by the client device for payment processing. For example, the user may provide payment input into the user device, e.g., **419**. In various implementations, the user input may include, but not be limited to: a single tap (e.g., a one-tap mobile app purchasing embodiment) of a touchscreen interface, keyboard entry, card swipe, activating a RFID/NFC enabled hardware device (e.g., electronic card having multiple accounts, smartphone, tablet, etc.) within the user device, mouse clicks, depressing buttons on a joystick/game console, voice commands, single/multi-touch gestures on a touch-sensitive interface, touching user interface elements on a touch-sensitive display, and/or the like. For example, the user device may obtain track 1 data from the user's card (e.g., credit card, debit card, prepaid card, charge card, etc.), such as the example track 1 data provided below:

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

[0057] In some implementation, the user device may determine whether an image it has captured depicts a QR code. Depending on whether or not a QR code has been captured, and also (optionally) depending on contents of the QR code, the user device may redirect the user (e.g., via a web browser application executing on the user device) to: a product, a merchant website, a product at a merchant website, a website and including a command to add an item to a purchasing cart of the user associated with the website, and/or the like. For example, the user device may execute a component such as the example Quick Response Code Processing ("QRCP") component **600** described below in the discussion with reference to FIGS. **6**A-B.

[0058] In some implementations, upon obtaining the user payment input and capturing the QR code, the user device may generate a card authorization request **420** (e.g., if the QR code includes a purchasing coupon, offer, invoice, personal payment from another virtual wallet user, etc.), for providing to the pay network server. For example, the user device may provide a card authorization request, e.g., **421**, on behalf of the user, a HTTP(S) GET message including the product order details for a pay network server, e.g., **406**, in the form of XML-formatted data. Below is an example HTTP(S) GET message including an XML-formatted card authorization request for the pay network server:

```
GET /purchase.php HTTP/1.1
Host: www.merchant.com
Content-Type: Application/XML
Content-Length: 1306
<?XML version = "1.0" encoding = "UTF-8"?>
<purchase_order>
    <order_ID>4NFU4RG94</order_ID>
    <alerts_URL>www.merchant.com/shopcarts.php?sessionID=AEBB4356</alerts_URL>
    <timestamp>2011-02-22 15:22:43</timestamp>
    <user_ID>john.q.public@gmail.com</user_ID>
    <client_details>
        <client_IP>192.168.23.126</client_IP>
        <client_type>smartphone</client_type>
        <client_model>HTC Hero</client_model>
        <OS>Android 2.2</OS>
        <app_installed_flag>true</app_installed_flag>
    </client_details>
    <purchase_details>
        <num_products>1</num_products>
        <product>
            <product_type>book</product_type>
            <product_params>
                <product_title>XML for dummies</product_title>
                <ISBN>938-2-14-168710-0</ISBN>
                <edition>2nd ed.</edition>
                <cover>hardbound</cover>
                <seller>bestbuybooks</seller>
            </product_params>
            <quantity>1</quantity>
        </product>
    </purchase_details>
    <merchant_params>
        <merchant_id>3FBCR4INC</merchant_id>
        <merchant_name>Books & Things, Inc.</merchant_name>
        <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
    </merchant_params>
    <account_params>
        <account_name>John Q. Public</account_name>
        <account_type>credit</account_type>
        <account_num>123456789012345</account_num>
        <billing_address>123 Green St., Norman, OK 98765</billing_address>
        <phone>123-456-7809</phone>
        <sign>/jqp/</sign>
        <confirm_type>email</confirm_type>
        <contact_info>john.q.public@gmail.com</contact_info>
    </account_params>
    <shipping_info>
        <shipping_adress>same as billing</shipping_address>
        <ship_type>expedited</ship_type>
```

-continued

```
        <ship__carrier>FedEx</ship__carrier>
        <ship__account>123-45-678</ship__account>
        <tracking__flag>true</tracking__flag>
        <sign__flag>false</sign__flag>
    </shipping__info>
</purchase__order>
```

[0059] In some implementations, the card authorization request generated by the user device may include a minimum of information required to process the purchase transaction. For example, this may improve the efficiency of communicating the purchase transaction request, and may also advantageously improve the privacy protections provided to the user and/or merchant. For example, in some implementations, the card authorization request may include at least a merchant ID, a session ID for the user's shopping session with the merchant, and a device ID of a device (e.g., smartphone) of the user that is linked to the user's virtual wallet. In some implementations, the QR code and messages sent to/from the QR-code capturing device may include the source ID (e.g., identifier of the device generating the QR code), session ID, merchant ID, item ID (e.g., model number), the charge amount, and/or transacting device ID (e.g., the user's smartphone device).

[0060] In some implementations, the card authorization request may be provided by the merchant server or point of sale terminal, instead of the user device. In some implementations, the user, desiring security, may request, via the user device, the pay network server for a dynamically-generated card verification value code (dCVV™) to be utilized along with the user's primary account number ("PAN," e.g., credit card number) in the purchase transaction. In response, the payment network server may generate a dCVV™ code (e.g., using random number generation, MD5 hash of an input key, which may be generated using the user ID, merchant ID, session ID, timestamp, combinations thereof, and/or the like), and provide a session-specific dCVV™ code for the user to utilize along with the user's PAN number. For example, the session-specific dCVV™ code may have an expiry time (e.g., expiry in a one minute from issue). The user device may communicate (e.g., via Bluetooth™, NFC, Wi-Fi, cellular, QR code, etc.) the PAN and dCVV to the point-of-sale terminal, which may create the card authorization request. For example, the user device may generate a QR payment code embedding the PAN and dCVV numbers, and the point of sale terminal may snap an image of the user device-generated QR payment code. The point of sale terminal may then generate and provide the card authorization request to the pay network server. The pay network server may then be able to validate the transaction by comparing the dCVV obtained from the merchant with the dCVV it provided to the user device before the purchase transaction was initiated. If the dCVV codes from the two sources (pay network server and merchant) correspond properly to each other, the pay network server may continue processing the purchase transaction.

[0061] In some implementations, the card authorization request from a user device may include encrypted data extracted from the QR code, which may have been encrypted by the merchant server as part of a merchant authentication scheme. In some implementations, the pay network server may obtain the encrypted data from the card authorization request provided by the user device, and attempt to decrypt the encrypted data, e.g., using a RSA private/public that is complementary to the key the pay network server initially provided to the merchant server for encrypting the purchase data before embedding it into the QR code. If the pay network server is able to decrypt the purchase data, then the merchant is authenticated as being a valid merchant. In some implementations, the pay network server may compare the purchase data decrypted from the card authorization with data provided by the user/user device, to determine whether the data from these different sources (user/user device, and merchant) correspond properly to each other. Thus, in some implementations, the pay network server may be able to authenticate the merchant, and correlate the merchant to a specific user session or user device before processing the transaction.

[0062] In some implementations, the pay network server may provide a notification to the user device that the transaction is authenticated and approved for transacting. In alternate implementations, the pay network server may proceed with transaction processing. In some implementations, upon identifying that the user is in a session with the merchant, the pay network server may communicate with the user device to provide additional features for the user. For example, in some implementations, the pay network server may provide a communication to the user device (e.g., via a HTTP(S) POST message) to provide: a virtual storefront of the merchant; a depiction of an aisle of the merchant associated with the products included in the card authorization request, a listing of related items; and/or the like (see, e.g., FIG. **7**E-G and description below of additional embodiments).

[0063] With reference to FIG. **4**B, in some implementations, the pay network server may process the transaction so as to transfer funds for the purchase into an account stored on an acquirer of the merchant. For example, the acquirer may be a financial institution maintaining an account of the merchant. For example, the proceeds of transactions processed by the merchant may be deposited into an account maintained by at a server of the acquirer.

[0064] In some implementations, the pay network server may generate a query, e.g., **422**, for issuer server(s) corresponding to the user-selected payment options. For example, the user's account may be linked to one or more issuer financial institutions ("issuers"), such as banking institutions, which issued the account(s) for the user. For example, such accounts may include, but not be limited to: credit card, debit card, prepaid card, checking, savings, money market, certificates of deposit, stored (cash) value accounts and/or the like. Issuer server(s), e.g., **408***a-n*, of the issuer(s) may maintain details of the user's account. In some implementations, a database, e.g., pay network database **407**, may store details of the issuer server(s) associated with the issuer(s). For example, the database may be a relational database responsive to Structured Query Language ("SQL") commands. The pay network server may query the pay network database for issuer server(s) details. For example, the pay network server may execute a hypertext preprocessor ("PHP") script including SQL commands to query the database for details of the issuer server(s). An example PHP/SQL command listing, illustrating substantive aspects of querying the database, is provided below:

```
<?PHP
header('Content-Type: text/plain');
mysql_connect("254.93.179.112",$DBserver,$password); // access
database server
mysql_select_db("ISSUERS.SQL"); // select database table to search
//create query for issuer server data
$query = "SELECT issuer_name issuer_address issuer_id ip_address
mac_address
     auth_key port_num security_settings_list FROM IssuerTable
     WHERE account_num
     LIKE '%' $accountnum";
$result = mysql_query($query); // perform the search query
mysql_close("ISSUERS.SQL"); // close database access
?>
```

[0065] In response to obtaining the issuer server query, e.g., **422**, the pay network database may provide, e.g., **423**, the requested issuer server data to the pay network server. In some implementations, the pay network server may utilize the issuer server data to generate authorization request(s), e.g., **424**, for each of the issuer server(s) selected based on the pre-defined payment settings associated with the user's virtual wallet, and/or the user's payment options input, and provide the card authorization request(s), e.g., **425***a-n*, to the issuer server(s), e.g., **408***a-n*. In some implementations, the authorization request(s) may include details such as, but not limited to: the costs to the user involved in the transaction, card account details of the user, user billing and/or shipping information, and/or the like. For example, the pay network server may provide a HTTP(S) POST message including an XML-formatted authorization request similar to the example listing provided below:

[0066] In some implementations, an issuer server may parse the authorization request(s), and based on the request details may query a database, e.g., user profile database **409***a-n*, for data associated with an account linked to the user. For example, the issuer server may issue PHP/SQL commands similar to the example provided below:

```
<?PHP
header('Content-Type: text/plain');
mysql_connect("254.93.179.112",$DBserver,$password); // access
database server
mysql_select_db("USERS.SQL"); // select database table to search
//create query for user data
$query = "SELECT user_id user_name user_balance account_type
FROM UserTable
     WHERE account_num LIKE '%' $accountnum";
$result = mysql_query($query); // perform the search query
mysql_close("USERS.SQL"); // close database access
?>
```

[0067] In some implementations, on obtaining the user data, e.g., **427***a-n*, the issuer server may determine whether the user can pay for the transaction using funds available in the account, e.g., **428***a-n*. For example, the issuer server may determine whether the user has a sufficient balance remaining in the account, sufficient credit associated with the account, and/or the like. Based on the determination, the issuer server(s) may provide an authorization response, e.g., **429***a-n*, to the pay network server. For example, the issuer server(s) may provide a HTTP(S) POST message similar to the examples above. In some implementations, if at least one issuer server determines that the user cannot pay for the transaction using the funds available in the account, see e.g., **430-431**, the pay network server may request payment options again from the user (e.g., by providing an authorization fail message **431** to the user device and requesting the user device to provide new payment options), and re-attempt authorization for the purchase transaction. In some implementations, if the number of

```
POST /authorization.php HTTP/1.1
Host: www.issuer.com
Content-Type: Application/XML
Content-Length: 624
<?XML version = "1.0" encoding = "UTF-8"?>
<card_query_request>
     <query_ID>VNEI39FK</query_ID>
     <timestamp>2011-02-22 15:22:44</timestamp>
     <purchase_summary>
          <num_products>1</num_products>
          <product>
               <product_summary>Book - XML for dummies</product_summary>
               <product_quantity>1</product_quantity?
          </product>
     </purchase_summary>
     <transaction_cost>$22.61</transaction_cost>
     <account_params>
          <account_type>checking</account_type>
          <account_num>1234567890123456</account_num>
     </account_params>
     <merchant_params>
          <merchant_id>3FBCR4INC</merchant_id>
          <merchant_name>Books & Things, Inc.</merchant_name>
          <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
     </merchant_params>
</card_query_request>
```

failed authorization attempts exceeds a threshold, the pay network server may abort the authorization process, and provide an “authorization fail” message to the merchant server, user device and/or client.

[0068] With reference to FIG. **4**C, in some implementations, the pay network server may obtain the authorization message including a notification of successful authorization, see e.g., **430**, **433**, and parse the message to extract authorization details. Upon determining that the user possesses sufficient funds for the transaction, the pay network server may generate a transaction data record, e.g., **432**, from the authorization request and/or authorization response, and store the details of the transaction and authorization relating to the transaction in a transactions database. For example, the pay network server may issue PHP/SQL commands similar to the example listing below to store the transaction data in a database:

```
<?PHP
header('Content-Type: text/plain');
mysql_connect("254.92.185.103”,$DBserver,$password);
// access database server
mysql_select("TRANSACTIONS.SQL"); // select database to append
mysql_query(“INSERT INTO PurchasesTable (timestamp,
     purchase_summary_list, num_products, product_summary,
     product_quantity, transaction_cost, account_params_list,
     account_name, account_type, account_num, billing_addres,
     zipcode, phone, sign, merchant_params_list, merchant_id,
     merchant_name, merchant_auth_key)
VALUES (time( ), $purchase_summary_list, $num_products,
     $product_summary, $product_quantity, $transaction_cost,
     $account_params_list, $account_name, $account_type,
     $account_num, $billing_addres, $zipcode, $phone, $sign,
     $merchant_params_list, $merchant_id, $merchant_name,
     $merchant_auth_key)”);
     // add data to table in database
mysql_close("TRANSACTIONS.SQL"); // close connection
to database
?>
```

[0069] In some implementations, the pay network server may forward an authorization success message, e.g., **433***a-b*, to the user device and/or merchant server. The merchant may obtain the authorization message, and determine from it that the user possesses sufficient funds in the card account to conduct the transaction. The merchant server may add a record of the transaction for the user to a batch of transaction data relating to authorized transactions. For example, the merchant may append the XML data pertaining to the user transaction to an XML data file comprising XML data for transactions that have been authorized for various users, e.g., **434**, and store the XML data file, e.g., **435**, in a database, e.g., merchant database **404**. For example, a batch XML data file may be structured similar to the example XML data structure template provided below:

```
<?XML version = “1.0” encoding = “UTF-8”?>
<merchant_data>
     <merchant_id>3FBCR4INC</merchant_id>
     <merchant_name>Books & Things, Inc.</merchant_name>
     <merchant_auth_key>1NNF484MCP59CHB27365</merchant_auth_key>
     <account_number>123456789</account_number>
</merchant_data>
<transaction_data>
     <transaction 1>
          ...
     </transaction 1>
     <transaction 2>
          ...
     </transaction 2>
          .
          .
          .
     <transaction n>
          ...
     </transaction n>
</transaction_data>
```

[0070] In some implementations, the server may also generate a purchase receipt, e.g., **434**, and provide the purchase receipt to the client, e.g., **436**. The client may render and display, e.g., **437***a*, the purchase receipt for the user. In some implementations, the user device **405** may also provide a notification of successful authorization to the user, e.g., **437***b*. For example, the client/user device may render a webpage, electronic message, text/SMS message, buffer a voicemail, emit a ring tone, and/or play an audio message, etc., and provide output including, but not limited to: sounds, music, audio, video, images, tactile feedback, vibration alerts (e.g., on vibration-capable client devices such as a smartphone etc.), and/or the like.

[0071] With reference to FIG. **4**D, in some implementations, the merchant server may initiate clearance of a batch of authorized transactions. For example, the merchant server may generate a batch data request, e.g., **438**, and provide the request, e.g., **439**, to a database, e.g., merchant database **404**. For example, the merchant server may utilize PHP/SQL commands similar to the examples provided above to query a relational database. In response to the batch data request, the database may provide the requested batch data, e.g., **440**. The server may generate a batch clearance request, e.g., **441**, using the batch data obtained from the database, and provide, e.g., **442**, the batch clearance request to an acquirer server, e.g., **410**. For example, the merchant server may provide a HTTP(S) POST message including XML-formatted batch data in the message body for the acquirer server. The acquirer server may generate, e.g., **443**, a batch payment request using the obtained batch clearance request, and provide the batch payment request to the pay network server, e.g., **444**. The pay network server may parse the batch payment request, and extract the transaction data for each transaction stored in the batch payment request, e.g., **445**. The pay network server may store the transaction data, e.g., **446**, for each transaction in a database, e.g., pay network database **407**. For each extracted transaction, the pay network server may query, e.g., **447-448**, a database, e.g., pay network database **407**, for an address of an issuer server. For example, the pay network server may utilize PHP/SQL commands similar to the examples provided above. The pay network server may generate an individual payment request, e.g., **449**, for each transaction for which it has extracted transaction data, and provide the individual payment request, e.g., **450**, to the issuer server, e.g., **408**. For example, the pay network server may provide a HTTP(S) POST request similar to the example below:

```
POST /requestpay.php HTTP/1.1
Host: www.issuer.com
Content-Type: Application/XML
Content-Length: 788
<?XML version = “1.0” encoding = “UTF-8”?>
<pay__request>
    <request__ID>CNI4ICNW2</request__ID>
    <timestamp>2011-02-22 17:00:01</timestamp>
    <pay__amount>$34.78</pay__amount>
    <account__params>
        <account__name>John Q. Public</account__name>
        <account__type>credit</account__type>
        <account__num>123456789012345</account__num>
        <billing__address>123 Green St., Norman, OK 98765</billing__address>
        <phone>123-456-7809</phone>
        <sign>/jqp/</sign>
    </account__params>
    <merchant__params>
        <merchant__id>3FBCR4INC</merchant__id>
        <merchant__name>Books & Things, Inc.</merchant__name>
        <merchant__auth__key>1NNF484MCP59CHB27365</merchant__auth__key>
    </merchant__params>
    <purchase__summary>
        <num__products>1</num__products>
        <product>
            <product__summary>Book - XML for dummies</product__summary>
            <product__quantity>1</product__quantity?
        </product>
    </purchase__summary>
</pay__request>
```

[0072] In some implementations, the issuer server may generate a payment command, e.g., **451**. 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., **452**, to a database storing the user's account information, e.g., user profile database **409**. The issuer server may provide a funds transfer message, e.g., **453**, to the pay network server, which may forward, e.g., **454**, the funds transfer message to the acquirer server. An example HTTP(S) POST funds transfer message is provided below:

```
POST /clearance.php HTTP/1.1
Host: www.acquirer.com
Content-Type: Application/XML
Content-Length: 206
```

-continued

```
<?XML version = “1.0” encoding = “UTF-8”?>
<deposit__ack>
    <request__ID>CNI4ICNW2</request__ID>
    <clear__flag>true</clear__flag>
    <timestamp>2011-02-22 17:00:02</timestamp>
    <deposit__amount>$34.78</deposit__amount>
</deposit__ack>
```

[0073] In some implementations, the acquirer server may parse the funds transfer message, and correlate the transaction (e.g., using the request_ID field in the example above) to the merchant. The acquirer server may then transfer the funds specified in the funds transfer message to an account of the merchant, e.g., **455**.

[0074] FIGS. **5**A-E show logic flow diagrams illustrating example aspects of executing a snap mobile payment in some embodiments of the SNAP, e.g., a Snap Mobile Payment Execution ("SMPE") component **500**. With reference to FIG. **5**A, in some implementations, a user may desire to purchase a product, service, offering, and/or the like ("product"), from a merchant via a merchant online site or in the merchant's store. The user may communicate with a merchant server via a client. For example, the user may provide user input, e.g., **501**, into the client indicating the user's desire to checkout shopping items in a (virtual) shopping cart. The client may generate a checkout request, e.g., **502**, and provide the checkout request to the merchant server. The merchant server may obtain the checkout request from the client, and extract the checkout detail (e.g., XML data) from the checkout request, e.g., **503**. For example, the merchant server may utilize a parser such as the example parsers described below in the discussion with reference to FIG. **14**. The merchant server may extract the product data, as well as the client data from the checkout request. In some implementations, the merchant server may query, e.g., **504**, a merchant database to obtain product data, e.g., **505**, such as product pricing, sales tax, offers, discounts, rewards, and/or other information to process the purchase transaction.

[0075] In response to obtaining the product data, the merchant server may generate, e.g., **506**, a QR pay code, and/or secure display element according to the security settings of the user (see, e.g., **358**). For example, the merchant server may generate a QR code embodying the product information, as well as merchant information required by a payment network to process the purchase transaction. For example, the merchant server may first generate in real-time, a custom, user-specific merchant-product XML data structure having a time-limited validity period, such as the example 'QR_data' XML data structure provided below:

```
<QR__data>
    <session__ID>4NFU4RG94</session__ID>
    <timestamp>2011-02-22 15:22:43</timestamp>
    <expiry__lapse>00:00:30</expiry__lapse>
    <transaction__cost>$34.78</transaction__cost>
    <user__ID>john.q.public@gmail.com</user__ID>
    <client__details>
        <client__IP>192.168.23.126</client__IP>
        <client__type>smartphone</client__type>
        <client__model>HTC Hero</client__model>
        <OS>Android 2.2</OS>
        <app__installed__flag>true</app__installed__flag>
    </client__details>
    <secure__element>www.merchant.com/securedyn/0394733/123.png</secure__element>
    <purchase__details>
        <num__products>1</num__products>
        <product>
            <product__type>book</product__type>
            <product__params>
                <product__title>XML for dummies</product__title>
                <ISBN>938-2-14-168710-0</ISBN>
                <edition>2nd ed.</edition>
                <cover>hardbound</cover>
                <seller>bestbuybooks</seller>
            </product__params>
            <quantity>1</quantity>
        </product>
    </purchase__details>
    <merchant__params>
        <merchant__id>3FBCR4INC</merchant__id>
        <merchant__name>Books & Things, Inc.</merchant__name>
        <merchant__auth__key>1NNF484MCP59CHB27365</merchant__auth__key>
    </merchant__params>
<QR__data>
```

[0076] In some implementations, the merchant may generate QR code using the XML data. For example, the merchant server may utilize the PHP QR Code open-source (LGPL) library for generating QR Code, 2-dimensional barcode, available at http://phpqrcode.sourceforge.net/. For example, the merchant server may issue PHP commands similar to the example commands provided below:

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

[0077] The merchant server may provide the QR pay code to the client, e.g., **506**. The client may obtain the QR pay code, and display the QR code, e.g., **507** on a display screen associated with the client device. In some implementations, the user may utilize a user device, e.g., **509**, to capture the QR code presented by the client device for payment processing. The client device may decode the QR code to extract the information embedded in the QR code. For example, the client device may utilize an application such as the ZXing multi-format 1D/2D barcode image processing library, available at http://code.google.com/p/zxing/ to extract the information from the QR code. In some implementations, the user may provide payment input into the user device, e.g., **508**. Upon obtaining the user purchase input, the user device may generate a card authorization request, e.g., **509**, and provide the card authorization request to a pay network server.

[0078] With reference to FIG. **5**B, in some implementations, the pay network server may parse the card authorization request, e.g., **510**, and generate a query, e.g., **511**, for issuer server(s) corresponding to the user-selected payment options. In some implementations, a pay network database may store details of the issuer server(s) associated with the issuer(s). In response to obtaining the issuer server query, the pay network database may provide, e.g., **512**, the requested issuer server data to the pay network server. In some implementations, the pay network server may utilize the issuer server data to generate authorization request(s), e.g., 425134, for each of the issuer server(s), and provide the card authorization request(s) to the issuer server(s).

[0079] In some implementations, an issuer server may parse the authorization request(s), and based on the request details may query a user profile database for data associated with an account linked to the user. In some implementations, on obtaining the user data, the issuer server may determine whether the user can pay for the transaction using funds available in the account, e.g., **517**. For example, the issuer server may determine whether the user has a sufficient balance remaining in the account, sufficient credit associated with the account, and/or the like. Based on the determination, the issuer server(s) may provide an authorization response, e.g., **518**, to the pay network server. In some implementations, if at least one issuer server determines, e.g., **519**, that the user cannot pay for the transaction using the funds available in the account, see e.g., **520**, option "No," the pay network server may request payment options again from the user (see e.g., **521**, option "No," by providing an authorization fail message to the user device and requesting the user device to provide new payment options), and re-attempt authorization for the purchase transaction. In some implementations, if the number of failed authorization attempts exceeds a threshold, see, e.g., **521**, option "Yes," the pay network server may abort the authorization process, and provide an "authorization fail" message to the merchant server, user device and/or client, e.g., **522**.

[0080] In some implementations, the pay network server may obtain the authorization message including a notification of successful authorization, see e.g., **520**, option "Yes,", and parse the message to extract authorization details. Upon determining that the user possesses sufficient funds for the transaction, the pay network server may generate a transaction data record, e.g., **523**, from the authorization request and/or authorization response, and store, e.g., **524**, the details of the transaction and authorization relating to the transaction in a transactions database.

[0081] With reference to FIG. **5**C, in some implementations, the pay network server may forward an authorization success message, e.g., **525**, to the user device and/or merchant server, sometimes via the acquirer server, e.g. **526**. The merchant may parse the authorization message, e.g., **528**, and determine from it that the user possesses sufficient funds in the card account to conduct the transaction, see, e.g., **529**. The merchant server may add a record of the transaction for the user to a batch of transaction data relating to authorized transactions, see, e.g., **530-531**. In some implementations, the merchant server may also generate a purchase receipt, e.g., **532**, and provide the purchase receipt to the client. The client may render and display, e.g., **534**, the purchase receipt for the user. In some implementations, the user device **405** may also provide a notification of successful authorization to the user.

[0082] With reference to FIGS. **5**D-E, in some implementations, the merchant server may initiate clearance of a batch of authorized transactions. For example, the merchant server may generate a batch data request, e.g., **535**, and provide the request, e.g., **536**, to a database, e.g., merchant database. In response to the batch data request, the database may provide the requested batch data, e.g., **536**. The server may generate a batch clearance request, e.g., **537**, using the batch data obtained from the database, and provide the batch clearance request to an acquirer server. The acquirer server may generate, e.g., **539**, a batch payment request using the obtained batch clearance request, and provide the batch payment request to the pay network server. The pay network server may parse the batch payment request, and extract the transaction data for each transaction stored in the batch payment request, e.g., **540-542**. The pay network server may store the transaction data, e.g., **543-544**, for each transaction in a database, e.g., pay network database. For each extracted transaction, the pay network server may query, e.g., **545-546**, a database, e.g., pay network database, for an address of an issuer server. The pay network server may generate an individual payment request, e.g., **547**, for each transaction for which it has extracted transaction data, and provide the individual payment request to the associated issuer server.

[0083] In some implementations, the issuer server may generate a payment command, e.g., **548-549**. 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., **549**, to a database storing the user's account information, e.g., user profile database. The issuer server may provide a funds transfer message, e.g., **551**, to the pay network server, which may forward the funds transfer message to the acquirer server. In some implementations, the acquirer server may parse the funds transfer message, and correlate the transaction (e.g., using the request_ID field in the example above) to the merchant. The acquirer server may then transfer the funds specified in the funds transfer message to an account of the merchant, e.g., **553-555**.

[0084] FIGS. **6**A-B show logic flow diagrams illustrating example aspects of processing a Quick Response code in some embodiments of the SNAP, e.g., a Quick Response Code Processing ("QRCP") component **600**. With reference to FIG. **6**A, in some implementations, a virtual wallet application executing on a user device may determine whether a QR code has been captured in an image frame obtained by a camera operatively connected to the user device, and may also determine the type, contents of the QR code. Using such information, the virtual wallet application may redirect the user experience of the user and/or initiating purchases, update aspects of the virtual wallet application, etc. For example, the virtual wallet application may trigger the capture of an image frame by a camera operatively connected to the user device, **601**. The virtual wallet application may utilize an image segmentation algorithm to identify a foreground in the image, **602**, and may crop the rest of the image to reduce background noise in the image, **603**. The virtual wallet application may determine whether the foreground image includes a QR code from which data can be reliably read (e.g., this may not be so if the image does not include a QR code, or the QR code is partially cropped, blurred, etc.), **604**. For example, the virtual wallet application may utilize a code library such as the ZXing multi-format 1D/2D barcode image processing library, available at http://code.google.com/p/zxing/ to try and extract the information from the QR code. If the virtual wallet application is able to detect a QR code (**605**, option "Yes"), the virtual wallet application may decode the QR code, and extract data from the QR code. If the virtual wallet application is unable to detect a QR code (**605**, option "No"), the virtual wallet application may attempt to perform Optical Character Recognition on the image. For example, the virtual wallet application may utilize the Tesseract C++ open source OCR engine, available at www.pixel-technology.com/freewarw/tessnet2, to perform the optical character recognition, **606**. Thus, the virtual wallet application may obtain the data encoded into the image, and may continue if the data can be processed by the virtual wallet application. The virtual wallet application may query a database using fields identified in the extracted data, for a type of the QR code, **608**. For example, the QR code could include an invoice/bill, a coupon, a money order (e.g., in a P2P transfer), a new account information packet, product information, purchase commands, URL navigation instructions, browser automation scripts, combinations thereof, and/or the like.

[0085] In some embodiments, the QR code may include data on a new account to be added to the virtual wallet application (see **609**). The virtual wallet application may query an issuer of the new account (as obtained from the extracted data), for the data associated with the new account, **610**. The virtual wallet application may compare the issuer-provided data to the data extracted from the QR code, **611**. If the new account is validated (**611**, option "Yes"), the virtual wallet application may update the wallet credentials with the details of the new account, **613**, and update the snap history of the virtual wallet application using the data from the QR code, **614**.

[0086] With reference to FIG. **6**B, in some embodiments, the QR code may include data on a bill, invoice, or coupon for a purchase using the virtual wallet application (see **615**). The virtual wallet application may query merchant(s) associated with the purchase (as obtained from the extracted data), for the data associated with the bill, invoice, or coupon for a purchase (e.g., offer details, offer ID, expiry time, etc.), **616**. The virtual wallet application may compare the merchant-provided data to the data extracted from the QR code, **617**. If the bill, invoice, or coupon for a purchase is validated (**618**, option "Yes"), the virtual wallet application may generate a data structure (see e.g., XML QR_data structure in description above with reference to FIGS. **4-5**) including the QR-encoded data for generating and providing a card authorization request, **619**, and update the snap history of the virtual wallet application using the data from the QR code, **620**.

[0087] In some embodiments, the QR code may include product information, commands, user navigation instructions, etc. for the virtual wallet application (see **621**). The virtual wallet application may query a product database using the information encoded in the QR. The virtual wallet application may provide various features including, without limitation, displaying product information, redirecting the user to: a product page, a merchant website, a product page on a merchant website, add item(s) to a user shopping cart at a merchant website, etc. In some implementations, the virtual wallet application may perform a procedure such as described above for any image frame pending to be processed, and/or selected for processing by the user (e.g., from the snap history).

[0088] FIG. **7** shows a user interface diagram illustrating an overview of example features of virtual wallet applications in some embodiments of the SNAP. FIG. **7** shows an illustration of various exemplary features of a virtual wallet mobile application **700**. Some of the features displayed include a wallet **701**, social integration via TWITTER, FACEBOOK, etc., offers and loyalty **703**, snap mobile purchase **704**, alerts **705** and security, setting and analytics **796**. These features are explored in further detail below.

[0089] FIGS. **8**A-G show user interface diagrams illustrating example features of virtual wallet applications in a shopping mode, in some embodiments of the SNAP. With reference to FIG. **8**A, some embodiments of the virtual wallet mobile app facilitate and greatly enhance the shopping experience of consumers. A variety of shopping modes, as shown in FIG. **8**A, may be available for a consumer to peruse. In one implementation, for example, a user may launch the shopping mode by selecting the shop icon **810** at the bottom of the user interface. A user may type in an item in the search field **812** to search and/or add an item to a cart **811**. A user may also use a voice activated shopping mode by saying the name or description of an item to be searched and/or added to the cart into a microphone **813**. In a further implementation, a user may also select other shopping options **814** such as current items **815**, bills **816**, address book **817**, merchants **818** and local proximity **819**.

[0090] In one embodiment, for example, a user may select the option current items **815**, as shown in the left most user interface of FIG. **8**A. When the current items **815** option is selected, the middle user interface may be displayed. As shown, the middle user interface may provide a current list of items **815***a-h* in a user's shopping cart **811**. A user may select an item, for example item **815***a*, to view product description **815***j* of the selected item and/or other items from the same merchant. The price and total payable information may also be displayed, along with a QR code **815***k* that captures the information necessary to effect a snap mobile purchase transaction.

[0091] With reference to FIG. **8**B, in another embodiment, a user may select the bills **816** option. Upon selecting the bills **816** option, the user interface may display a list of bills and/or receipts **816***a-h* from one or more merchants. Next to each of the bills, additional information such as date of visit, whether items from multiple stores are present, last bill payment date, auto-payment, number of items, and/or the like may be displayed. In one example, the wallet shop bill **816***a* dated Jan. 20, 2011 may be selected. The wallet shop bill selection may display a user interface that provides a variety of information regarding the selected bill. For example, the user interface may display a list of items **816***k* purchased, <<**816***i*>>, a total number of items and the corresponding value. For example, 7 items worth $102.54 were in the selected wallet shop bill. A user may now select any of the items and select buy again to add purchase the items. The user may also refresh offers **816***j* to clear any invalid offers from last time and/or search for new offers that may be applicable for the current purchase. As shown in FIG. **8**B, a user may select two items for repeat purchase. Upon addition, a message **8161** may be displayed to confirm the addition of the two items, which makes the total number of items in the cart **14**.

[0092] With reference to FIG. **8**C, in yet another embodiment, a user may select the address book option **817** to view the address book **817***a* which includes a list of contacts **817***b* and make any money transfers or payments. In one embodiment, the address book may identify each contact using their names and available and/or preferred modes of payment. For example, a contact Amanda G. may be paid via social pay (e.g., via FACEBOOK) as indicated by the icon **817***c*. In another example, money may be transferred to Brian S. via QR code as indicated by the QR code icon **817***d*. In yet another example, Charles B. may accept payment via near field communication **817***e*, Bluetooth **817***f* and email **817***g*. Payment may also be made via USB **817***h* (e.g., by physically connecting two mobile devices) as well as other social channels such as TWITTER.

[0093] In one implementation, a user may select Joe P. for payment. Joe P., as shown in the user interface, has an email icon **817***g* next to his name indicating that Joe P. accepts payment via email. When his name is selected, the user interface may display his contact information such as email, phone, etc. If a user wishes to make a payment to Joe P. by a method other than email, the user may add another transfer mode **817***j* to his contact information and make a payment transfer. With reference to FIG. **8**D, the user may be provided with a screen **817***k* where the user can enter an amount to send Joe, as well as add other text to provide Joe with context for the payment transaction **817***l*. The user can choose modes (e.g., SMS, email, social networking) via which Joe may be contacted via graphical user interface elements, **817***m*. As the user types, the text entered may be provided for review within a GUI element **817***n*. When the user has completed entering in the necessary information, the user can press the send button **817***o* to send the social message to Joe. If Joe also has a virtual wallet application, Joe may be able to review **817***p* social pay message within the app, or directly at the website of the social network (e.g., for Twitter™, Facebook®, etc.). Messages may be aggregated from the various social networks and other sources (e.g., SMS, email). The method of redemption appropriate for each messaging mode may be indicated along with the social pay message. In the illustration in FIG. **8**D, the SMS **817***q* Joe received indicates that Joe can redeem the $5 obtained via SMS by replying to the SMS and entering the hash tag value '*1234'. In the same illustration, Joe has also received a message **817***r* via Facebook®, which includes a URL link that Joe can activate to initiate redemption of the $25 payment.

[0094] With reference to FIG. **8**E, in some other embodiments, a user may select merchants **818** from the list of options in the shopping mode to view a select list of merchants **818***a-e*. In one implementation, the merchants in the list may be affiliated to the wallet, or have affinity relationship with the wallet. In another implementation, the merchants may include a list of merchants meeting a user-defined or other criteria. For example, the list may be one that is curated by the user, merchants where the user most frequently shops or spends more than an x amount of sum or shopped for three consecutive months, and/or the like. In one implementation, the user may further select one of the merchants, Amazon **818***a* for example. The user may then navigate through the merchant's listings to find items of interest such as **818***f-j*. Directly through the wallet and without visiting the merchant site from a separate page, the user may make a selection of an item **818***j* from the catalog of Amazon **818***a*. As shown in the right most user interface of FIG. **8**D, the selected item may then be added to cart. The message **818***k* indicates that the selected item has been added to the cart, and updated number of items in the cart is now 13.

[0095] With reference to FIG. **8**F, in one embodiment, there may be a local proximity option **819** which may be selected by a user to view a list of merchants that are geographically in close proximity to the user. For example, the list of merchants **819***a-e* may be the merchants that are located close to the user. In one implementation, the mobile application may further identify when the user in a store based on the user's location. For example, position icon **819***d* may be displayed next to a store (e.g., Walgreens) when the user is in close proximity to the store. In one implementation, the mobile application may refresh its location periodically in case the user moved away from the store (e.g., Walgreens). In a further implementation, the user may navigate the offerings of the selected Walgreens store through the mobile application. For example, the user may navigate, using the mobile application, to items **819***f-j* available on aisle 5 of Walgreens. In one implementation, the user may select corn **819***i* from his or her mobile application to add to cart **819***k*.

[0096] With reference to FIG. **8**G, in another embodiment, the local proximity option **819** may include a store map and a real time map features among others. For example, upon selecting the Walgreens store, the user may launch an aisle map **819***l* which displays a map **819***m* showing the organization of the store and the position of the user (indicated by a yellow circle). In one implementation, the user may easily configure the map to add one or more other users (e.g., user's kids) to share each other's location within the store. In another implementation, the user may have the option to launch a "store view" similar to street views in maps. The store view **819***n* may display images/video of the user's surrounding. For example, if the user is about to enter aisle 5, the store view map may show the view of aisle 5. Further the user may manipulate the orientation of the map using the navigation tool **819***o* to move the store view forwards, backwards, right, left as well clockwise and counterclockwise rotation.

[0097] FIGS. **9**A-F show user interface diagrams illustrating example features of virtual wallet applications in a payment mode, in some embodiments of the SNAP. With reference to FIG. **9**A, in one embodiment, the wallet mobile application may provide a user with a number of options for paying for a transaction via the wallet mode **910**. In one implementation, an example user interface **911** for making a payment is shown. The user interface may clearly identify the amount **912** and the currency **913** for the transaction. The amount may be the amount payable and the currency may include real currencies such as dollars and euros, as well as virtual currencies such as reward points. The amount of the transaction **914** may also be prominently displayed on the user interface. The user may select the funds tab **916** to select one or more forms of payment **917**, which may include various credit, debit, gift, rewards and/or prepaid cards. The user may also have the option of paying, wholly or in part, with reward points. For example, the graphical indicator **918** on the user interface shows the number of points available, the graphical indicator **919** shows the number of points to be used towards the amount due 234.56 and the equivalent **920** of the number of points in a selected currency (USD, for example).

[0098] In one implementation, the user may combine funds from multiple sources to pay for the transaction. The amount **915** displayed on the user interface may provide an indication of the amount of total funds covered so far by the selected forms of payment (e.g., Discover card and rewards points). The user may choose another form of payment or adjust the amount to be debited from one or more forms of payment until the amount **915** matches the amount payable **914**. Once the amounts to be debited from one or more forms of payment are finalized by the user, payment authorization may begin.

[0099] In one implementation, the user may select a secure authorization of the transaction by selecting the cloak button **922** to effectively cloak or anonymize some (e.g., pre-configured) or all identifying information such that when the user selects pay button **921**, the transaction authorization is conducted in a secure and anonymous manner. In another implementation, the user may select the pay button **921** which may use standard authorization techniques for transaction processing. In yet another implementation, when the user selects the social button **923**, a message regarding the transaction may be communicated to one of more social networks (set up by the user) which may post or announce the purchase transaction in a social forum such as a wall post or a tweet. In one implementation, the user may select a social payment processing option **923**. The indicator **924** may show the authorizing and sending social share data in progress.

[0100] In another implementation, a restricted payment mode **925** may be activated for certain purchase activities such as prescription purchases. The mode may be activated in accordance with rules defined by issuers, insurers, merchants, payment processor and/or other entities to facilitate processing of specialized goods and services. In this mode, the user may scroll down the list of forms of payments **926** under the funds tab to select specialized accounts such as a flexible spending account (FSA) **927**, health savings account (HAS), and/or the like and amounts to be debited to the selected accounts. In one implementation, such restricted payment mode **1925** processing may disable social sharing of purchase information.

[0101] In one embodiment, the wallet mobile application may facilitate importing of funds via the import funds user interface **928**. For example, a user who is unemployed may obtain unemployment benefit fund **929** via the wallet mobile application. In one implementation, the entity providing the funds may also configure rules for using the fund as shown by the processing indicator message **930**. The wallet may read and apply the rules prior, and may reject any purchases with the unemployment funds that fail to meet the criteria set by the rules. Example criteria may include, for example, merchant category code (MCC), time of transaction, location of transaction, and/or the like. As an example, a transaction with a grocery merchant having MCC 5411 may be approved, while a transaction with a bar merchant having an MCC 5813 may be refused.

[0102] With reference to FIG. **9**B, in one embodiment, the wallet mobile application may facilitate dynamic payment optimization based on factors such as user location, preferences and currency value preferences among others. For example, when a user is in the United States, the country indicator **931** may display a flag of the United States and may set the currency **933** to the United States. In a further implementation, the wallet mobile application may automatically rearrange the order in which the forms of payments **935** are listed to reflect the popularity or acceptability of various forms of payment. In one implementation, the arrangement may reflect the user's preference, which may not be changed by the wallet mobile application.

[0103] Similarly, when a German user operates a wallet in Germany, the mobile wallet application user interface may be dynamically updated to reflect the country of operation **932** and the currency **934**. In a further implementation, the wallet application may rearrange the order in which different forms of payment **936** are listed based on their acceptance level in that country. Of course, the order of these forms of payments may be modified by the user to suit his or her own preferences.

[0104] With reference to FIG. **9**C, in one embodiment, the payee tab **937** in the wallet mobile application user interface may facilitate user selection of one or more payees receiving the funds selected in the funds tab. In one implementation, the user interface may show a list of all payees **938** with whom the user has previously transacted or available to transact. The user may then select one or more payees. The payees **938** may include larger merchants such as Amazon.com Inc., and individuals such as Jane P. Doe. Next to each payee name, a list of accepted payment modes for the payee may be displayed. In one implementation, the user may select the payee Jane P. Doe **939** for receiving payment. Upon selection, the user interface may display additional identifying information relating to the payee.

[0105] With reference to FIG. **9**D, in one embodiment, the mode tab **1940** may facilitate selection of a payment mode accepted by the payee. A number of payment modes may be available for selection. Example modes include, blue tooth **941**, wireless **942**, snap mobile by user-obtained QR code **943**, secure chip **944**, TWITTER **945**, near-field communication (NFC) **946**, cellular **947**, snap mobile by user-provided QR code **948**, USB **949** and FACEBOOK **950**, among others. In one implementation, only the payment modes that are accepted by the payee may be selectable by the user. Other non-accepted payment modes may be disabled.

[0106] With reference to FIG. **9**E, in one embodiment, the offers tab **951** may provide real-time offers that are relevant to items in a user's cart for selection by the user. The user may select one or more offers from the list of applicable offers **952** for redemption. In one implementation, some offers may be combined, while others may not. When the user selects an offer that may not be combined with another offer, the unselected offers may be disabled. In a further implementation, offers that are recommended by the wallet application's recommendation engine may be identified by an indicator, such as the one shown by **953**. In a further implementation, the user may read the details of the offer by expanding the offer row as shown by **954** in the user interface.

[0107] With reference to FIG. **9**F, in one embodiment, the social tab **955** may facilitate integration of the wallet application with social channels **956**. In one implementation, a user may select one or more social channels **956** and may sign in to the selected social channel from the wallet application by providing to the wallet application the social channel user name and password **957** and signing in **958**. The user may then use the social button **959** to send or receive money through the integrated social channels. In a further implementation, the user may send social share data such as purchase information or links through integrated social channels. In another embodiment, the user supplied login credentials may allow SNAP to engage in interception parsing.

[0108] FIG. **10** shows a user interface diagram illustrating example features of virtual wallet applications, in a history mode, in some embodiments of the SNAP. In one embodiment, a user may select the history mode **1010** to view a history of prior purchases and perform various actions on those prior purchases. For example, a user may enter a merchant identifying information such as name, product, MCC, and/or the like in the search bar **1011**. In another implementation, the user may use voice activated search feature by clicking on the microphone icon **1014**. The wallet application may query the storage areas in the mobile device or elsewhere (e.g., one or more databases and/or tables remote from the mobile device) for transactions matching the search keywords. The user interface may then display the results of the query such as transaction **1015**. The user interface may also identify the date **1012** of the transaction, the merchants and items **1013** relating to the transaction, a barcode of the receipt confirming that a transaction was made, the amount of the transaction and any other relevant information.

[0109] In one implementation, the user may select a transaction, for example transaction **1015**, to view the details of the transaction. For example, the user may view the details of the items associated with the transaction and the amounts **1016** of each item. In a further implementation, the user may select the show option **1017** to view actions **1018** that the user may take in regards to the transaction or the items in the transaction. For example, the user may add a photo to the transaction (e.g., a picture of the user and the iPad the user bought). In a further implementation, if the user previously shared the purchase via social channels, a post including the photo may be generated and sent to the social channels for publishing. In one implementation, any sharing may be optional, and the user, who did not share the purchase via social channels, may still share the photo through one or more social channels of his or her choice directly from the history mode of the wallet application. In another implementation, the user may add the transaction to a group such as company expense, home expense, travel expense or other categories set up by the user. Such grouping may facilitate year-end accounting of expenses, submission of work expense reports, submission for value added tax (VAT) refunds, personal expenses, and/or the like. In yet another implementation, the user may buy one or more items purchased in the transaction. The user may then execute a transaction without going to the merchant catalog or site to find the items. In a further implementation, the user may also cart one or more items in the transaction for later purchase.

[0110] The history mode, in another embodiment, may offer facilities for obtaining and displaying ratings **1019** of the items in the transaction. The source of the ratings may be the user, the user's friends (e.g., from social channels, contacts, etc.), reviews aggregated from the web, and/or the like. The user interface in some implementations may also allow the user to post messages to other users of social channels (e.g., TWITTER or FACEBOOK). For example, the display area **1020** shows FACEBOOK message exchanges between two users. In one implementation, a user may share a link via a message **1021**. Selection of such a message having embedded link to a product may allow the user to view a description of the product and/or purchase the product directly from the history mode.

[0111] In one embodiment, the history mode may also include facilities for exporting receipts. The export receipts pop up **1022** may provide a number of options for exporting the receipts of transactions in the history. For example, a user may use one or more of the options **1025**, which include save (to local mobile memory, to server, to a cloud account, and/or the like), print to a printer, fax, email, and/or the like. The user may utilize his or her address book **1023** to look up email or fax number for exporting. The user may also specify format options **1024** for exporting receipts. Example format options may include, without limitation, text files (.doc, .txt, .rtf, iif, etc.), spreadsheet (.csv, .xls, etc.), image files (.jpg, .tff, .png, etc.), portable document format (.pdf), postscript (.ps), and/or the like. The user may then click or tap the export button **1027** to initiate export of receipts.

[0112] FIGS. **11**A-F show user interface diagrams illustrating example features of virtual wallet applications in a snap mode, in some embodiments of the SNAP. With reference to FIG. **11**A, in some embodiments, a user may select a snap mode **1101** to access snap features. In various embodiments, the virtual wallet application may able to snap and identify a variety of items. For example, the virtual wallet application may be able to snap and identify a purchase invoice **1103**, a coupon **104**, money (e.g., sent in a person-to-person transfer) **1105**, a bill (e.g., utilities, etc.) **1106**, a receipt (e.g., for storing, expense reporting, etc.) **1107**, a pay account (e.g., to add a new credit/debit/prepaid card to the virtual wallet application) **1108**. The user may be able to return to a shopping screen at any time by activating a graphical user interface element **1102**. In some embodiments, the user may be able to set a name of a cart or wishlist stored within the user's virtual wallet application to which the item snapped should be sent (see **1109**). In some embodiments, the virtual wallet application may allow a user to create a new cart or wishlist to which the snapped items should be added.

[0113] In one embodiment, a user may select the snap mode **1110** to access its snap features. The snap mode may handle any machine-readable representation of data. Examples of such data may include linear and 2D bar codes such as UPC code and QR codes. These codes may be found on receipts, product packaging, and/or the like. The snap mode may also process and handle pictures of receipts, products, offers, credit cards or other payment devices, and/or the like. An example user interface in snap mode is shown in FIG. **11**A. A user may use his or her mobile phone to take a picture of a QR code **1115** and/or a barcode **1114**. In one implementation, the bar **1113** and snap frame **1115** may assist the user in snapping codes properly. For example, the snap frame **1115**, as shown, does not capture the entirety of the code **1116**. As such, the code captured in this view may not be resolvable as information in the code may be incomplete. This is indicated by the message on the bar **1113** that indicates that the snap mode is still seeking the code. The user may modify the zoom level **1117** of the camera to facilitate snapping the QR code. When the code **1116** is completely framed by the snap frame **1115**, the bar message may be updated to, for example, "snap found." Upon finding the code, in one implementation, the user may initiate code capture using the mobile device camera (see **1120**). In another implementation, the snap mode may automatically snap the code using the mobile device camera (see **1119**). In some implementations, the virtual wallet application may optionally apply a Global Positioning System tag (see **1118**) to the QR code before storing it, or utilizing it in a transaction.

[0114] With reference to FIG. **11**B, in one embodiment, the snap mode may facilitate payment reallocation post transaction. For example, a user may buy grocery and prescription items from a retailer Acme Supermarket. The user may, inadvertently or for ease of checkout for example, use his or her Visa card to pay for both grocery and prescription items. However, the user may have an FSA account that could be used to pay for prescription items, and which would provide the user tax benefits. In such a situation, the user may use the snap mode to initiate transaction reallocation.

[0115] As shown, the user may enter a search term (e.g., bills) in the search bar **2121**. The user may then identify in the tab **1122** the receipt **1123** the user wants to reallocate. Alternatively, the user may directly snap a picture of a barcode on a receipt, and the snap mode may generate and display a receipt **1123** using information from the barcode. The user may now reallocate **1125**. In some implementations, the user may also dispute the transaction **1124** or archive the receipt **1126**.

[0116] In one implementation, when the reallocate button **1125** is selected, the wallet application may perform optical character recognition (OCR) of the receipt. Each of the items in the receipt may then be examined to identify one or more items which could be charged to which payment device or account for tax or other benefits such as cash back, reward points, etc. In this example, there is a tax benefit if the prescription medication charged to the user's Visa card is charged to the user's FSA. The wallet application may then perform the reallocation as the back end. The reallocation process may include the wallet contacting the payment processor to credit the amount of the prescription medication to the Visa card and debit the same amount to the user's FSA account. In an alternate implementation, the payment processor (e.g., Visa or MasterCard) may obtain and OCR the receipt, identify items and payment accounts for reallocation and perform the reallocation. In one implementation, the wallet application may request the user to confirm reallocation of charges for the selected items to another payment account. The receipt **1127** may be generated after the completion of the reallocation process. As discussed, the receipt shows that some charges have been moved from the Visa account to the FSA.

[0117] With reference to FIG. **11**C, in one embodiment, the snap mode may facilitate payment via pay code such as barcodes or QR codes. For example, a user may snap a QR code of a transaction that is not yet complete. The QR code may be displayed at a merchant POS terminal, a web site, or a web application and may be encoded with information identifying items for purchase, merchant details and other relevant information. When the user snaps such as a QR code, the snap mode may decode the information in the QR code and may use the decoded information to generate a receipt **1132**. Once the QR code is identified, the navigation bar **1131** may indicate that the pay code is identified. The user may now have an option to add to cart **1133**, pay with a default payment account **1134** or pay with wallet **1135**.

[0118] In one implementation, the user may decide to pay with default **1134**. The wallet application may then use the user's default method of payment, in this example the wallet, to complete the purchase transaction. Upon completion of the transaction, a receipt may be automatically generated for proof of purchase. The user interface may also be updated to provide other options for handling a completed transaction. Example options include social **1137** to share purchase information with others, reallocate **1138** as discussed with regard to FIG. **11**B, and archive **1139** to store the receipt.

[0119] With reference to FIG. **11**D, in one embodiment, the snap mode may also facilitate offer identification, application and storage for future use. For example, in one implementation, a user may snap an offer code **1141** (e.g., a bar code, a QR code, and/or the like). The wallet application may then generate an offer text **1142** from the information encoded in the offer code. The user may perform a number of actions on the offer code. For example, the user use the find button **1143** to find all merchants who accept the offer code, merchants in the proximity who accept the offer code, products from merchants that qualify for the offer code, and/or the like. The user may also apply the offer code to items that are currently in the cart using the add to cart button **1144**. Furthermore, the user may also save the offer for future use by selecting the save button **1145**.

[0120] In one implementation, after the offer or coupon **1146** is applied, the user may have the option to find qualifying merchants and/or products using find, the user may go to the wallet using **1148**, and the user may also save the offer or coupon **1146** for later use.

[0121] With reference to FIG. **11**E, in one embodiment, the snap mode may also offer facilities for adding a funding source to the wallet application. In one implementation, a pay card such as a credit card, debit card, pre-paid card, smart card and other pay accounts may have an associated code such as a bar code or QR code. Such a code may have encoded therein pay card information including, but not limited to, name, address, pay card type, pay card account details, balance amount, spending limit, rewards balance, and/or the like. In one implementation, the code may be found on a face of the physical pay card. In another implementation, the code may be obtained by accessing an associated online account or another secure location. In yet another implementation, the code may be printed on a letter accompanying the pay card. A user, in one implementation, may snap a picture of the code. The wallet application may identify the pay card **1151** and may display the textual information **1152** encoded in the pay card. The user may then perform verification of the information **1152** by selecting the verify button **1153**. In one implementation, the verification may include contacting the issuer of the pay card for confirmation of the decoded information **1152** and any other relevant information. In one implementation, the user may add the pay card to the wallet by selecting the 'add to wallet' button **1154**. The instruction to add the pay card to the wallet may cause the pay card to appear as one of the forms of payment under the funds tab **916** discussed in FIG. **9**A. The user may also cancel importing of the pay card as a funding source by selecting the cancel button **1155**. When the pay card has been added to the wallet, the user interface may be updated to indicate that the importing is complete via the notification display **1156**. The user may then access the wallet **1157** to begin using the added pay card as a funding source.

[0122] With reference to FIG. **11**F, in some implementations, the virtual wallet application may identify a product from processing the QR code, and may provide information about the product, as well as information about options for buying the product, assistive services, and/or the like. For example, the virtual wallet application may provide a window **1161**, wherein the virtual wallet application may display images, product specification, prices, merchant information, and/or the like (see **1162**). In some implementations, the virtual wallet application may provide a QR code including the displayed information, so that another user may quickly snap the information to import it into another virtual wallet application. In some implementations, the virtual wallet application may provide features so that a user may request concierge services (e.g., assistance while shopping), shipping services (e.g., so the user may leave a store without carrying the items out), **1164**. In some implementations, the virtual wallet application may provide competitive prices of local merchants (e.g., using the GPS location of the user device) or merchants on the Internet (see **1165**). In some implementations, the virtual wallet application may provide the user with features including, but not limited to: viewing prior snaps, snapping a new code, adding GPS tags to codes, retrieving a previously snapped code for use, entering manual information about a QR code, attribute the QR code to an object (e.g., so that QR codes for home furniture products may be grouped into a "bedroom furniture" object, for organization purposes), etc. (see **1166**). In some embodiments, the user may be able to set a name of a cart or wishlist stored within the user's virtual wallet application to which the item snapped should be sent (see **1167**). In some embodiments, the virtual wallet application may allow a user to create a new cart or wishlist to which the snapped items should be added.

[0123] FIG. **12** shows a user interface diagram illustrating example features of virtual wallet applications, in an offers mode, in some embodiments of the SNAP. In some implementations, the SNAP may allow a user to search for offers for products and/or services from within the virtual wallet mobile application. For example, the user may enter text into a graphical user interface ("GUI") element **1211**, or issue voice commands by activating GUI element **1212** and speaking commands into the device. In some implementations, the SNAP may provide offers based on the user's prior behavior, demographics, current location, current cart selection or purchase items, and/or the like. For example, if a user is in a brick-and-mortar store, or an online shopping website, and leaves the (virtual) store, then the merchant associated with the store may desire to provide a sweetener deal to entice the consumer back into the (virtual) store. The merchant may provide such an offer **1213**. For example, the offer may provide a discount, and may include an expiry time. In some implementations, other users may provide gifts (e.g., **1214**) to the user, which the user may redeem. In some implementations, the offers section may include alerts as to payment of funds outstanding to other users (e.g., **1215**). In some implementations, the offers section may include alerts as to requesting receipt of funds from other users (e.g., **1216**). For example, such a feature may identify funds receivable from other applications (e.g., mail, calendar, tasks, notes, reminder programs, alarm, etc.), or by a manual entry by the user into the virtual wallet application. In some implementations, the offers section may provide offers from participating merchants in the SNAP, e.g., **1217**-**1219**, **1220**. These offers may sometimes be assembled using a combination of participating merchants, e.g., **1217**. In some implementations, the SNAP itself may provide offers for users contingent on the user utilizing particular payment forms from within the virtual wallet application, e.g., **1220**.

[0124] FIGS. **13**A-B show user interface diagrams illustrating example features of virtual wallet applications, in a security and privacy mode, in some embodiments of the SNAP. With reference to FIG. **13**A, in some implementations, the user may be able to view and/or modify the user profile and/or settings of the user, e.g., by activating a user interface element. For example, the user may be able to view/modify a user name (e.g., **1311***a*-*b*), account number (e.g., **1312***a*-*b*), user security access code (e.g., **1313**-*b*), user pin (e.g., **1314**-*b*), user address (e.g., **1315**-*b*), social security number associated with the user (e.g., **1316**-*b*), current device GPS location (e.g., **1317**-*b*), user account of the merchant in whose store the user currently is (e.g., **1318**-*b*), the user's rewards accounts (e.g., **1319**-*b*), and/or the like. In some implementations, the user may be able to select which of the data fields and their associated values should be transmitted to facilitate the purchase transaction, thus providing enhanced data security for the user. For example, in the example illustration in FIG. **13**A, the user has selected the name **1311***a*, account number **1312***a*, security code **1313***a*, merchant account ID **1318***a* and rewards account ID **1319***a* as the fields to be sent as part of the notification to process the purchase transaction. In some implementations, the user may toggle the fields and/or data values that are sent as part of the notification to process the purchase transactions. In some implementations, the app may provide multiple screens of data fields and/or associated values stored for the user to select as part of the purchase order transmission. In some implementations, the app may provide the SNAP with the GPS location of the user. Based on the GPS location of the user, the SNAP may determine the context of the user (e.g., whether the user is in a store, doctor's office, hospital, postal service office, etc.). Based on the context, the user app may present the appropriate fields to the user, from which the user may select fields and/or field values to send as part of the purchase order transmission.

[0125] For example, a user may go to doctor's office and desire to pay the co-pay for doctor's appointment. In addition to basic transactional information such as account number and name, the app may provide the user the ability to select to transfer medical records, health information, which may be provided to the medical provider, insurance company, as well as the transaction processor to reconcile payments between the parties. In some implementations, the records may be sent in a Health Insurance Portability and Accountability Act (HIPAA)-compliant data format and encrypted, and only the recipients who are authorized to view such records may have appropriate decryption keys to decrypt and view the private user information.

[0126] With reference to FIG. **13**B, in some implementations, the app executing on the user's device may provide a "VerifyChat" feature for fraud prevention. For example, the SNAP may detect an unusual and/or suspicious transaction. The SNAP may utilize the VerifyChat feature to communicate with the user, and verify the authenticity of the originator of the purchase transaction. In various implementations, the SNAP may send electronic mail message, text (SMS) messages, Facebook® messages, Twitter™ tweets, text chat, voice chat, video chat (e.g., Apple FaceTime), and/or the like to communicate with the user. For example, the SNAP may initiate a video challenge for the user, e.g., **1321**. For example, the user may need to present him/her-self via a video chat, e.g., **1322**. In some implementations, a customer service representative, e.g., agent **1324**, may manually determine the authenticity of the user using the video of the user. In some implementations, the SNAP may utilize face, biometric and/or like recognition (e.g., using pattern classification techniques) to determine the identity of the user. In some implementations, the app may provide reference marker (e.g., cross-hairs, target box, etc.), e.g., **1323**, so that the user may the video to facilitate the SNAP's automated recognition of the user. In some implementations, the user may not have initiated the transaction, e.g., the transaction is fraudulent. In such implementations, the user may cancel the challenge. The SNAP may then cancel the transaction, and/or initiate fraud investigation procedures on behalf of the user.

[0127] In some implementations, the SNAP may utilize a text challenge procedure to verify the authenticity of the user, e.g., **1325**. For example, the SNAP may communicate with the user via text chat, SMS messages, electronic mail, Facebook® messages, Twitter™ tweets, and/or the like. The SNAP may pose a challenge question, e.g., **1326**, for the user. The app may provide a user input interface element(s) (e.g., virtual keyboard **1328**) to answer the challenge question posed by the SNAP. In some implementations, the challenge question may be randomly selected by the SNAP automatically; in some implementations, a customer service representative may manually communicate with the user. In some implementations, the user may not have initiated the transaction, e.g., the transaction is fraudulent. In such implementations, the user may cancel the text challenge. The SNAP may cancel the transaction, and/or initiate fraud investigation on behalf of the user.

SNAP Controller

[0128] FIG. **14** shows a block diagram illustrating embodiments of a SNAP controller **1401**. In this embodiment, the SNAP controller **1401** may serve to aggregate, process, store, search, serve, identify, instruct, generate, match, and/or facilitate interactions with a computer through various technologies, and/or other related data.

[0129] Typically, users, e.g., **1433***a*, which may be people and/or other systems, may engage information technology systems (e.g., computers) to facilitate information processing. In turn, computers employ processors to process information; such processors **1403** may be referred to as central processing units (CPU). One form of processor is referred to as a microprocessor. CPUs use communicative circuits to pass binary encoded signals acting as instructions to enable various operations. These instructions may be operational and/or data instructions containing and/or referencing other instructions and data in various processor accessible and operable areas of memory **1429** (e.g., registers, cache memory, random access memory, etc.). Such communicative instructions may be stored and/or transmitted in batches (e.g., batches of instructions) as programs and/or data components to facilitate desired operations. These stored instruction codes, e.g., programs, may engage the CPU circuit components and other motherboard and/or system components to perform desired operations. One type of program is a computer operating system, which, may be executed by CPU on a computer; the operating system enables and facilitates users to access and operate computer information technology and resources. Some resources that may be employed in information technology systems include: input and output mechanisms through which data may pass into and out of a computer; memory storage into which data may be saved; and processors by which information may be processed. These information technology systems may be used to collect data for later retrieval, analysis, and manipulation, which may be facilitated through a database program. These information technology systems provide interfaces that allow users to access and operate various system components.

[0130] In one embodiment, the SNAP controller **1401** may be connected to and/or communicate with entities such as, but not limited to: one or more users from user input devices **1411**; peripheral devices **1412**; an optional cryptographic processor device **1428**; and/or a communications network **1413**. For example, the SNAP controller **1401** may be connected to and/or communicate with users, e.g., **1433***a*, operating client device(s), e.g., **1433***b*, including, but not limited to, personal computer(s), server(s) and/or various mobile device(s) including, but not limited to, cellular telephone(s), smartphone(s) (e.g., iPhone®, Blackberry®, Android OS-based phones etc.), tablet computer(s) (e.g., Apple iPad™, HP Slate™, Motorola Xoom™, etc.), eBook reader(s) (e.g., Amazon Kindle™, Barnes and Noble's Nook™ eReader, etc.), laptop computer(s), notebook(s), netbook(s), gaming console(s) (e.g., XBOX Live™, Nintendo® DS, Sony PlayStation® Portable, etc.), portable scanner(s), and/or the like.

[0131] Networks are commonly thought to comprise the interconnection and interoperation of clients, servers, and intermediary nodes in a graph topology. It should be noted that the term "server" as used throughout this application refers generally to a computer, other device, program, or combination thereof that processes and responds to the requests of remote users across a communications network. Servers serve their information to requesting "clients." The term "client" as used herein refers generally to a computer, program, other device, user and/or combination thereof that is capable of processing and making requests and obtaining and processing any responses from servers across a communications network. A computer, other device, program, or combination thereof that facilitates, processes information and requests, and/or furthers the passage of information from a source user to a destination user is commonly referred to as a "node." Networks are generally thought to facilitate the transfer of information from source points to destinations. A node specifically tasked with furthering the passage of information from a source to a destination is commonly called a "router." There are many forms of networks such as Local Area Networks (LANs), Pico networks, Wide Area Networks (WANs), Wireless Networks (WLANs), etc. For example, the Internet is generally accepted as being an interconnection of a multitude of networks whereby remote clients and servers may access and interoperate with one another.

[0132] The SNAP controller **1401** may be based on computer systems that may comprise, but are not limited to, components such as: a computer systemization **1402** connected to memory **1429**.

Computer Systemization

[0133] A computer systemization **1402** may comprise a clock **1430**, central processing unit ("CPU(s)" and/or "processor(s)" (these terms are used interchangeable throughout the disclosure unless noted to the contrary)) **1403**, a memory **1429** (e.g., a read only memory (ROM) **1406**, a random access memory (RAM) **1405**, etc.), and/or an interface bus **1407**, and most frequently, although not necessarily, are all interconnected and/or communicating through a system bus **1404** on one or more (mother)board(s) **1402** having conductive and/or otherwise transportive circuit pathways through which instructions (e.g., binary encoded signals) may travel to effectuate communications, operations, storage, etc. The computer systemization may be connected to a power source **1486**; e.g., optionally the power source may be internal. Optionally, a cryptographic processor **1426** and/or transceivers (e.g., ICs) **1474** may be connected to the system bus. In another embodiment, the cryptographic processor and/or transceivers may be connected as either internal and/or external peripheral devices **1412** via the interface bus I/O. In turn, the transceivers may be connected to antenna(s) **1475**, **13** 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 SNAP controller to determine its location)); Broadcom BCM4329FKUBG transceiver chip (e.g., providing 802.11n, Bluetooth 2.1+EDR, FM, etc.); a Broadcom BCM4750IUB8 receiver chip (e.g., GPS); an Infineon Technologies X-Gold 618-PMB9800 (e.g., providing 2G/3G HSDPA/HSUPA communications); and/or the like. The system clock typically has a crystal oscillator and generates a base signal through the computer systemization's circuit pathways. The clock is typically 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 commonly 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 one another, connected to the CPU, and/or organized in numerous variations employed as exemplified by various computer systems.

[0134] The CPU comprises at least one high-speed data processor adequate to execute program components for executing user and/or system-generated requests. Often, the processors themselves will incorporate various specialized processing units, such as, but not limited to: integrated system (bus) controllers, memory management control units, floating point units, and even specialized processing sub-units like graphics processing units, digital signal processing units, and/or the like. Additionally, processors may include internal fast access addressable memory, and be capable of mapping and addressing memory **1429** beyond the processor itself; internal memory may include, but is not limited to: fast registers, various levels of cache memory (e.g., level 1, 2, 3, etc.), RAM, etc. The processor may access this memory through the use of a memory address space that is accessible via instruction address, which the processor can construct and decode allowing it to access a circuit path to a specific memory address space having a memory state. The CPU may be a microprocessor such as: AMD's Athlon, Duron and/or Opteron; ARM's application, embedded and secure processors; IBM and/or Motorola's DragonBall and PowerPC; IBM's and Sony's Cell processor; Intel's Celeron, Core (2) Duo, Itanium, Pentium, Xeon, and/or XScale; and/or the like processor(s). The CPU interacts with memory through instruction passing through conductive and/or transportive conduits (e.g., (printed) electronic and/or optic circuits) to execute stored instructions (i.e., program code) according to conventional data processing techniques. Such instruction passing facilitates communication within the SNAP controller and beyond through various interfaces. Should processing requirements dictate a greater amount speed and/or capacity, distributed processors (e.g., Distributed SNAP), mainframe, multi-core, parallel, and/or super-computer architectures may similarly be employed. Alternatively, should deployment requirements dictate greater portability, smaller Personal Digital Assistants (PDAs) may be employed.

[0135] Depending on the particular implementation, features of the SNAP may be achieved by implementing a microcontroller such as CAST's R8051XC2 microcontroller; Intel's MCS 51 (i.e., 8051 microcontroller); and/or the like. Also, to implement certain features of the SNAP, some feature implementations may rely on embedded components, such as: Application-Specific Integrated Circuit ("ASIC"), Digital Signal Processing ("DSP"), Field Programmable Gate Array ("FPGA"), and/or the like embedded technology. For example, any of the SNAP component collection (distributed or otherwise) and/or features may be implemented via the microprocessor and/or via embedded components; e.g., via ASIC, coprocessor, DSP, FPGA, and/or the like. Alternately, some implementations of the SNAP may be implemented with embedded components that are configured and used to achieve a variety of features or signal processing.

[0136] Depending on the particular implementation, the embedded components may include software solutions, hardware solutions, and/or some combination of both hardware/software solutions. For example, SNAP features discussed herein may be achieved through implementing FPGAs, which are a semiconductor devices containing programmable logic components called "logic blocks", and programmable interconnects, such as the high performance FPGA Virtex series and/or the low cost Spartan series manufactured by Xilinx. Logic blocks and interconnects can be programmed by the customer or designer, after the FPGA is manufactured, to implement any of the SNAP features. A hierarchy of programmable interconnects allow logic blocks to be interconnected as needed by the SNAP system designer/administrator, somewhat like a one-chip programmable breadboard. An FPGA's logic blocks can be programmed to perform the operation of basic logic gates such as AND, and XOR, or more complex combinational operators such as decoders or simple mathematical operations. In most FPGAs, the logic blocks also include memory elements, which may be circuit flip-flops or more complete blocks of memory. In some circumstances, the SNAP may be developed on regular FPGAs and then migrated into a fixed version that more resembles ASIC implementations. Alternate or coordinating implementations may migrate SNAP controller features to a final ASIC instead of or in addition to FPGAs. Depending on the implementation all of the aforementioned embedded components and microprocessors may be considered the "CPU" and/or "processor" for the SNAP.

Power Source

[0137] The power source **1486** may be of any standard form for powering small electronic circuit board devices such as the following power cells: alkaline, lithium hydride, lithium ion, lithium polymer, nickel cadmium, solar cells, and/or the like. Other types of AC or DC power sources may be used as well. In the case of solar cells, in one embodiment, the case provides an aperture through which the solar cell may capture photonic energy. The power cell **1486** is connected to at least one of the interconnected subsequent components of the SNAP thereby providing an electric current to all subsequent components. In one example, the power source **1486** is connected to the system bus component **1404**. In an alternative embodiment, an outside power source **1486** is provided through a connection across the I/O **1408** interface. For example, a USB and/or IEEE 1394 connection carries both data and power across the connection and is therefore a suitable source of power.

Interface Adapters

[0138] Interface bus(ses) **1407** may accept, connect, and/or communicate to a number of interface adapters, conventionally although not necessarily in the form of adapter cards, such as but not limited to: input output interfaces (I/O) **1408**, storage interfaces **1409**, network interfaces **1410**, and/or the like. Optionally, cryptographic processor interfaces **1427** similarly may be connected to the interface bus. The interface bus provides for the communications of interface adapters with one another as well as with other components of the computer systemization. Interface adapters are adapted for a compatible interface bus. Interface adapters conventionally connect to the interface bus via a slot architecture. Conventional slot architectures may be employed, such as, but not limited to: Accelerated Graphics Port (AGP), Card Bus, (Extended) Industry Standard Architecture ((E)ISA), Micro Channel Architecture (MCA), NuBus, Peripheral Component Interconnect (Extended) (PCI(X)), PCI Express, Personal Computer Memory Card International Association (PCMCIA), and/or the like.

[0139] Storage interfaces **1409** may accept, communicate, and/or connect to a number of storage devices such as, but not limited to: storage devices **1414**, removable disc devices, and/or the like. Storage interfaces may employ connection protocols such as, but not limited to: (Ultra) (Serial) Advanced Technology Attachment (Packet Interface) ((Ultra) (Serial) ATA(PI)), (Enhanced) Integrated Drive Electronics ((E)IDE), Institute of Electrical and Electronics Engineers (IEEE) 1394, fiber channel, Small Computer Systems Interface (SCSI), Universal Serial Bus (USB), and/or the like.

[0140] Network interfaces **1410** may accept, communicate, and/or connect to a communications network **1413**. Through a communications network **1413**, the SNAP controller is accessible through remote clients **1433***b* (e.g., computers with web browsers) by users **1433***a*. Network interfaces may employ connection protocols such as, but not limited to: direct connect, Ethernet (thick, thin, twisted pair 10/100/1000 Base T, and/or the like), Token Ring, wireless connection such as IEEE 802.11a-x, and/or the like. Should processing requirements dictate a greater amount speed and/or capacity, distributed network controllers (e.g., Distributed SNAP), architectures may similarly be employed to pool, load balance, and/or otherwise increase the communicative bandwidth required by the SNAP controller. A communications network may be any one and/or the combination of the following: a direct interconnection; the Internet; a Local Area Network (LAN); a Metropolitan Area Network (MAN); an Operating Missions as Nodes on the Internet (OMNI); a secured custom connection; a Wide Area Network (WAN); a wireless network (e.g., employing protocols such as, but not limited to a Wireless Application Protocol (WAP), I-mode, and/or the like); and/or the like. A network interface may be regarded as a specialized form of an input output interface. Further, multiple network interfaces **1410** may be used to engage with various communications network types **1413**. For example, multiple network interfaces may be employed to allow for the communication over broadcast, multicast, and/or unicast networks.

[0141] Input Output interfaces (I/O) **1408** may accept, communicate, and/or connect to user input devices **1411**, peripheral devices **1412**, cryptographic processor devices **1428**, and/or the like. I/O may employ connection protocols such as, but not limited to: audio: analog, digital, monaural, RCA, stereo, and/or the like; data: Apple Desktop Bus (ADB), IEEE 1394a-b, serial, universal serial bus (USB); infrared; joystick; keyboard; midi; optical; PC AT; PS/2; parallel; radio; video interface: Apple Desktop Connector (ADC), BNC, coaxial, component, composite, digital, Digital Visual Interface (DVI), high-definition multimedia interface (HDMI), RCA, RF antennae, S-Video, VGA, and/or the like; wireless transceivers: 802.11a/b/g/n/x; Bluetooth; cellular (e.g., code division multiple access (CDMA), high speed packet access (HSPA(+)), high-speed downlink packet access (HSDPA), global system for mobile communications (GSM), long term evolution (LTE), WiMax, etc.); and/or the like. One typical output device may include a video display, which typically comprises a Cathode Ray Tube (CRT) or Liquid Crystal Display (LCD) based monitor with an interface (e.g., DVI circuitry and cable) that accepts signals from a video interface, may be used. The video interface composites information generated by a computer systemization and generates video signals based on the composited information in a video memory frame. Another output device is a television set, which accepts signals from a video interface. Typically, the video interface provides the composited video information through a video connection interface that accepts a video display interface (e.g., an RCA composite video connector accepting an RCA composite video cable; a DVI connector accepting a DVI display cable, etc.).

[0142] User input devices **1411** often are a type of peripheral device **1412** (see below) and may include: card readers, dongles, finger print readers, gloves, graphics tablets, joysticks, keyboards, microphones, mouse (mice), remote controls, retina readers, touch screens (e.g., capacitive, resistive, etc.), trackballs, trackpads, sensors (e.g., accelerometers, ambient light, GPS, gyroscopes, proximity, etc.), styluses, and/or the like.

[0143] Peripheral devices **1412** may be connected and/or communicate to I/O and/or other facilities of the like such as network interfaces, storage interfaces, directly to the interface bus, system bus, the CPU, and/or the like. Peripheral devices may be external, internal and/or part of the SNAP controller. Peripheral devices may include: antenna, audio devices (e.g., line-in, line-out, microphone input, speakers, etc.), cameras (e.g., still, video, webcam, etc.), dongles (e.g., for copy protection, ensuring secure transactions with a digital signature, and/or the like), external processors (for added capabilities; e.g., crypto devices **1428**), force-feedback devices (e.g., vibrating motors), network interfaces, printers, scanners, storage devices, transceivers (e.g., cellular, GPS, etc.), video devices (e.g., goggles, monitors, etc.), video sources, visors, and/or the like. Peripheral devices often include types of input devices (e.g., cameras).

[0144] It should be noted that although user input devices and peripheral devices may be employed, the SNAP controller may be embodied as an embedded, dedicated, and/or monitor-less (i.e., headless) device, wherein access would be provided over a network interface connection.

[0145] Cryptographic units such as, but not limited to, microcontrollers, processors **1426**, interfaces **1427**, and/or devices **1428** may be attached, and/or communicate with the SNAP controller. A MC68HC16 microcontroller, manufactured by Motorola Inc., may be used for and/or within cryptographic units. The MC68HC16 microcontroller utilizes a 16-bit multiply-and-accumulate instruction in the 16 MHz configuration and requires less than one second to perform a 512-bit RSA private key operation. Cryptographic units support the authentication of communications from interacting agents, as well as allowing for anonymous transactions. Cryptographic units may also be configured as part of the CPU. Equivalent microcontrollers and/or processors may also be used. Other commercially available specialized cryptographic processors include: the Broadcom's CryptoNetX and other Security Processors; nCipher's nShield, SafeNet's Luna PCI (e.g., 7100) series; Semaphore Communications' 40 MHz Roadrunner 184; Sun's Cryptographic Accelerators (e.g., Accelerator 6000 PCIe Board, Accelerator 500 Daughtercard); Via Nano Processor (e.g., L2100, L2200, U2400) line, which is capable of performing 500+ MB/s of cryptographic instructions; VLSI Technology's 33 MHz 6868; and/or the like.

Memory

[0146] Generally, any mechanization and/or embodiment allowing a processor to affect the storage and/or retrieval of information is regarded as memory **1429**. However, memory is a fungible technology and resource, thus, any number of memory embodiments may be employed in lieu of or in concert with one another. It is to be understood that the SNAP controller and/or a computer systemization may employ various forms of memory **1429**. For example, a computer systemization may be configured wherein the operation of on-chip CPU memory (e.g., registers), RAM, ROM, and any other storage devices are provided by a paper punch tape or paper punch card mechanism; however, such an embodiment would result in an extremely slow rate of operation. In a typical configuration, memory **1429** will include ROM **1406**, RAM **1405**, and a storage device **1414**. A storage device **1414** may be any conventional computer system storage. Storage devices may include a drum; a (fixed and/or removable) magnetic disk drive; a magneto-optical drive; an optical drive (i.e., Blueray, CD ROM/RAM/Recordable (R)/ReWritable (RW), DVD R/RW, HD DVD R/RW etc.); an array of devices (e.g., Redundant Array of Independent Disks (RAID)); solid state memory devices (USB memory, solid state drives (SSD), etc.); other processor-readable storage mediums; and/or other devices of the like. Thus, a computer systemization generally requires and makes use of memory.

Component Collection

[0147] The memory **1429** may contain a collection of program and/or database components and/or data such as, but not limited to: operating system component(s) **1415** (operating system); information server component(s) **1416** (information server); user interface component(s) **1417** (user interface); Web browser component(s) **1418** (Web browser); database(s) **1419**; mail server component(s) **1421**; mail client component (s) **1422**; cryptographic server component(s) **1420** (cryptographic server); the SNAP component(s) **1435**; and/or the like (i.e., collectively a component collection). These components may be stored and accessed from the storage devices and/or from storage devices accessible through an interface bus. Although non-conventional program components such as those in the component collection, typically, are stored in a local storage device **1414**, they may also be loaded and/or stored in memory such as: peripheral devices, RAM, remote storage facilities through a communications network, ROM, various forms of memory, and/or the like.

Operating System

[0148] The operating system component **1415** is an executable program component facilitating the operation of the SNAP controller. Typically, the operating system facilitates access of I/O, network interfaces, peripheral devices, storage devices, and/or the like. The operating system may be a highly fault tolerant, scalable, and secure system such as: Apple Macintosh OS X (Server); AT&T Plan 9; Be OS; Unix and Unix-like system distributions (such as AT&T's UNIX; Berkley Software Distribution (BSD) variations such as FreeBSD, NetBSD, OpenBSD, and/or the like; Linux distributions such as Red Hat, Ubuntu, and/or the like); and/or the like operating systems. However, more limited and/or less secure operating systems also may be employed such as Apple Macintosh OS, IBM OS/2, Microsoft DOS, Microsoft Windows 2000/2003/3.1/95/98/CE/Millenium/NT/Vista/XP (Server), Palm OS, and/or the like. An operating system may communicate to and/or with other components in a component collection, including itself, and/or the like. Most frequently, the operating system communicates with other program components, user interfaces, and/or the like. For example, the operating system may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses. The operating system, once executed by the CPU, may enable the interaction with communications networks, data, I/O, peripheral devices, program components, memory, user input devices, and/or the like. The operating system may provide communications protocols that allow the SNAP controller to communicate with other entities through a communications network **1413**. Various communication protocols may be used by the SNAP controller as a subcarrier transport mechanism for interaction, such as, but not limited to: multicast, TCP/IP, UDP, unicast, and/or the like.

Information Server

[0149] An information server component **1416** is a stored program component that is executed by a CPU. The information server may be a conventional Internet information server such as, but not limited to Apache Software Foundation's Apache, Microsoft's Internet Information Server, and/or the like. The information server may allow for the execution of program components through facilities such as Active Server Page (ASP), ActiveX, (ANSI) (Objective-) C (++), C# and/or .NET, Common Gateway Interface (CGI) scripts, dynamic (D) hypertext markup language (HTML), FLASH, Java, JavaScript, Practical Extraction Report Language (PERL), Hypertext Pre-Processor (PHP), pipes, Python, wireless application protocol (WAP), WebObjects, and/or the like. The information server may support secure communications protocols such as, but not limited to, File Transfer Protocol (FTP); HyperText Transfer Protocol (HTTP); Secure Hypertext Transfer Protocol (HTTPS), Secure Socket Layer (SSL), messaging protocols (e.g., America Online (AOL) Instant Messenger (AIM), Application Exchange (APEX), ICQ, Internet Relay Chat (IRC), Microsoft Network (MSN) Messenger Service, Presence and Instant Messaging Protocol (PRIM), Internet Engineering Task Force's (IETF's) Session Initiation Protocol (SIP), SIP for Instant Messaging and Presence Leveraging Extensions (SIMPLE), open XML-based Extensible Messaging and Presence Protocol (XMPP) (i.e., Jabber or Open Mobile Alliance's (OMA's) Instant Messaging and Presence Service (IMPS)), Yahoo! Instant Messenger Service, and/or the like. The information server provides results in the form of Web pages to Web browsers, and allows for the manipulated generation of the Web pages through interaction with other program components. After a Domain Name System (DNS) resolution portion of an HTTP request is resolved to a particular information server, the information server resolves requests for information at specified locations on the SNAP controller based on the remainder of the HTTP request. For example, a request such as http://123.124.125.126/myInformation.html might have the IP portion of the request "123.124.125.126" resolved by a DNS server to an information server at that IP address; that information server might in turn further parse the http request for the "/myInformation.html" portion of the request and resolve it to a location in memory containing the information "myInformation.html." Additionally, other information serving protocols may be employed across various ports, e.g., FTP communications across port **21**, and/or the like. An information server may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the information server communicates with the SNAP database **1419**, operating systems, other program components, user interfaces, Web browsers, and/or the like.

[0150] Access to the SNAP database may be achieved through a number of database bridge mechanisms such as through scripting languages as enumerated below (e.g., CGI) and through inter-application communication channels as enumerated below (e.g., CORBA, WebObjects, etc.). Any data requests through a Web browser are parsed through the bridge mechanism into appropriate grammars as required by the SNAP. In one embodiment, the information server would provide a Web form accessible by a Web browser. Entries made into supplied fields in the Web form are tagged as having been entered into the particular fields, and parsed as such. The entered terms are then passed along with the field tags, which act to instruct the parser to generate queries directed to appropriate tables and/or fields. In one embodiment, the parser may generate queries in standard SQL by instantiating a search string with the proper join/select commands based on the tagged text entries, wherein the resulting command is provided over the bridge mechanism to the SNAP as a query. Upon generating query results from the query, the results are passed over the bridge mechanism, and may be parsed for formatting and generation of a new results Web page by the bridge mechanism. Such a new results Web page is then provided to the information server, which may supply it to the requesting Web browser.

[0151] Also, an information server may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses.

User Interface

[0152] Computer interfaces in some respects are similar to automobile operation interfaces. Automobile operation interface elements such as steering wheels, gearshifts, and speedometers facilitate the access, operation, and display of automobile resources, and status. Computer interaction interface elements such as check boxes, cursors, menus, scrollers, and windows (collectively and commonly referred to as widgets) similarly facilitate the access, capabilities, operation, and display of data and computer hardware and operating system resources, and status. Operation interfaces are commonly called user interfaces. Graphical user interfaces (GUIs) such as the Apple Macintosh Operating System's Aqua, IBM's OS/2, Microsoft's Windows 2000/2003/3.1/95/98/CE/Millenium/NT/XP/Vista/7 (i.e., Aero), Unix's X-Windows (e.g., which may include additional Unix graphic interface libraries and layers such as K Desktop Environment (KDE), mythTV and GNU Network Object Model Environment (GNOME)), web interface libraries (e.g., ActiveX, AJAX, (D)HTML, FLASH, Java, JavaScript, etc. interface libraries such as, but not limited to, Dojo, jQuery(UI), MooTools, Prototype, script.aculo.us, SWFObject, Yahoo! User Interface, any of which may be used and) provide a baseline and means of accessing and displaying information graphically to users.

[0153] A user interface component **1417** is a stored program component that is executed by a CPU. The user interface may be a conventional graphic user interface as provided by, with, and/or atop operating systems and/or operating environments such as already discussed. The user interface may allow for the display, execution, interaction, manipulation, and/or operation of program components and/or system facilities through textual and/or graphical facilities. The user interface provides a facility through which users may affect, interact, and/or operate a computer system. A user interface may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the user interface communicates with operating systems, other program components, and/or the like. The user interface may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses.

Web Browser

[0154] A Web browser component **1418** is a stored program component that is executed by a CPU. The Web browser may be a conventional hypertext viewing application such as Microsoft Internet Explorer or Netscape Navigator. Secure Web browsing may be supplied with 128 bit (or greater) encryption by way of HTTPS, SSL, and/or the like. Web browsers allowing for the execution of program components through facilities such as ActiveX, AJAX, (D)HTML, FLASH, Java, JavaScript, web browser plug-in APIs (e.g., FireFox, Safari Plug-in, and/or the like APIs), and/or the like. Web browsers and like information access tools may be integrated into PDAs, cellular telephones, and/or other mobile devices. A Web browser may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the Web browser communicates with information servers, operating systems, integrated program components (e.g., plug-ins), and/or the like; e.g., it may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses. Also, in place of a Web browser and information server, a combined application may be developed to perform similar operations of both. The combined application would similarly affect the obtaining and the provision of information to users, user agents, and/or the like from the SNAP enabled nodes. The combined application may be nugatory on systems employing standard Web browsers.

Mail Server

[0155] A mail server component **1421** is a stored program component that is executed by a CPU **1403**. The mail server may be a conventional Internet mail server such as, but not limited to sendmail, Microsoft Exchange, and/or the like. The mail server may allow for the execution of program components through facilities such as ASP, ActiveX, (ANSI) (Objective-) C (++), C# and/or .NET, CGI scripts, Java, JavaScript, PERL, PHP, pipes, Python, WebObjects, and/or the like. The mail server may support communications protocols such as, but not limited to: Internet message access protocol (IMAP), Messaging Application Programming Interface (MAPI)/Microsoft Exchange, post office protocol (POP3), simple mail transfer protocol (SMTP), and/or the like. The mail server can route, forward, and process incoming and outgoing mail messages that have been sent, relayed and/or otherwise traversing through and/or to the SNAP.

[0156] Access to the SNAP mail may be achieved through a number of APIs offered by the individual Web server components and/or the operating system.

[0157] Also, a mail server may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, information, and/or responses.

Mail Client

[0158] A mail client component **1422** is a stored program component that is executed by a CPU **1403**. The mail client may be a conventional mail viewing application such as Apple 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

[0159] A cryptographic server component **1420** is a stored program component that is executed by a CPU **1403**, cryptographic processor **1426**, cryptographic processor interface **1427**, cryptographic processor device **1428**, 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 conventional 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 to: digital certificates (e.g., X.509 authentication framework), digital signatures, dual signatures, enveloping, password access protection, public key management, and/or the like. The cryptographic component will facilitate numerous (encryption and/or decryption) security protocols such as, but not limited to: checksum, Data Encryption Standard (DES), Elliptical Curve Encryption (ECC), International Data Encryption Algorithm (IDEA), Message Digest 5 (MD5, which is a one way hash operation), passwords, Rivest Cipher (RC5), Rijndael, RSA (which is an Internet encryption and authentication system that uses an algorithm developed in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman), Secure Hash Algorithm (SHA), Secure Socket Layer (SSL), Secure Hypertext Transfer Protocol (HTTPS), and/or the like. Employing such encryption security protocols, the SNAP may encrypt all incoming and/or outgoing communications and may serve as node within a virtual private network (VPN) with a wider communications network. The cryptographic component facilitates the process of "security authorization" whereby access to a resource is inhibited by a security protocol wherein the cryptographic component effects authorized access to the secured resource. In addition, the cryptographic component may provide unique identifiers of content, e.g., employing and MD5 hash to obtain a unique signature for an digital audio file. A cryptographic component may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. The cryptographic component supports encryption schemes allowing for the secure transmission of information across a communications network to enable the SNAP component to engage in secure transactions if so desired. The cryptographic component facilitates the secure accessing of resources on the SNAP and facilitates the access of secured resources on remote systems; i.e., it may act as a client and/or server of secured resources. Most frequently, the cryptographic component communicates with information servers, operating systems, other program components, and/or the like. The cryptographic component may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses.

The SNAP Database

[0160] The SNAP database component **1419** may be embodied in a database and its stored data. The database is a stored program component, which is executed by the CPU; the stored program component portion configuring the CPU to process the stored data. The database may be a conventional, fault tolerant, relational, scalable, secure database such as Oracle or Sybase. Relational databases are an extension of a flat file. Relational databases consist of a series of related tables. The tables are interconnected via a key field. Use of the key field allows the combination of the tables by indexing against the key field; i.e., the key fields act as dimensional pivot points for combining information from various tables. Relationships generally identify links maintained between tables by matching primary keys. Primary keys represent fields that uniquely identify the rows of a table in a relational database. More precisely, they uniquely identify rows of a table on the "one" side of a one-to-many relationship.

[0161] Alternatively, the SNAP database may be implemented using various standard data-structures, such as an array, hash, (linked) list, struct, structured text file (e.g., XML), table, and/or the like. Such data-structures may be stored in memory and/or in (structured) files. In another alternative, an object-oriented database may be used, such as Frontier, ObjectStore, Poet, Zope, and/or the like. Object databases can include a number of object collections that are grouped and/or linked together by common attributes; they may be related to other object collections by some common attributes. Object-oriented databases perform similarly to relational databases with the exception that objects are not just pieces of data but may have other types of capabilities encapsulated within a given object. If the SNAP database is implemented as a data-structure, the use of the SNAP database **1419** may be integrated into another component such as the SNAP component **1435**. Also, the database may be implemented as a mix of data structures, objects, and relational structures. Databases may be consolidated and/or distributed in countless variations through standard data processing techniques. Portions of databases, e.g., tables, may be exported and/or imported and thus decentralized and/or integrated.

[0162] In one embodiment, the database component **1419** includes several tables **1419***a-o*. A Users table **1419***a* may include fields such as, but not limited to: user_id, ssn, dob, first_name, last_name, age, state, address_firstline, address_secondline, zipcode, devices_list, contact_info, contact_type, alt_contact_info, alt_contact_type, and/or the like. The Users table may support and/or track multiple entity accounts on a SNAP. A Devices table **1419***b* may include fields such as, but not limited to: device_ID, device_name, device_IP, device_MAC, device_type, device_model, device_version, device_OS, device_apps_list, device_securekey, wallet_app_installed_flag, and/or the like. An Apps table **1419***c* may include fields such as, but not limited to: app_ID, app_name, app_type, app_dependencies, and/or the like. An Accounts table **1419***d* may include fields such as, but not limited to: account_number, account_security_code, account_name, issuer_acquirer_flag, issuer_name, acquirer_name, account_address, routing_number, access_API_call, linked_wallets_list, and/or the like. A Merchants table **1419***e* may include fields such as, but not limited to: merchant_id, merchant_name, merchant_address, ip_address, mac_address, auth_key, port_num, security_settings_list, and/or the like. An Issuers table **1419***f* may include fields such as, but not limited to: issuer_id, issuer_name, issuer_address, ip_address, mac_address, auth_key, port_num, security_settings_list, and/or the like. An Acquirers table **1419***g* may include fields such as, but not limited to: account_firstname, account_lastname, account_type, account_num, account_balance_list, billingaddress_line1, billingaddress_line2, billing_zipcode, billing_state, shipping_preferences, shippingaddress_line1, shippingaddress_line2, shipping_zipcode, shipping_state, and/or the like. A Pay Gateways table **1419***h* may include fields such as, but not limited to: gateway_ID, gateway_IP, gateway_MAC, gateway_secure_key, gateway_access_list, gateway_API_call_list, gateway_services_list, and/or the like. A Transactions table **1419***i* may include fields such as, but not limited to: order_id, user_id, timestamp, transaction_cost, purchase_details_list, num_products, products_list, product_type, product_params_list, product_title, product_summary, quantity, user_id, client_id, client_ip, client_type, client_model, operating_system, os_version, app_installed_flag, user_id, account_firstname, account_lastname, account_type, account_num, account_priority_account_ratio, billingaddress_line1, billingaddress_line2, billing_zipcode, billing_state, shipping_preferences, shippingaddress_line1, shippingaddress_line2, shipping_zipcode, shipping_state, merchant_id, merchant_name, merchant_auth_key, and/or the like. A Batches table **1419***j* may include fields such as, but not limited to: batch_id, transaction_id_list, timestamp_list, cleared_flag_list, clearance_trigger_settings, and/or the like. A Ledgers table **1419***k* may include fields such as, but not limited to: request_id, timestamp, deposit_amount, batch_id, transaction_id, clear_flag, deposit_account, transaction_summary, payor_name, payor_account, and/or the like. A Products table **1419***l* may include fields such as, but not limited to: product_ID, product_title, product_attributes_list, product_price, tax_info_list, related_products_list, offers_list, discounts_list, rewards_list, merchants_list, merchant_availability_list, and/or the like. An Offers table **1419***m* may include fields such as, but not limited to: offer_ID, offer_title, offer_attributes_list, offer_price, offer_expiry, related_products_list, discounts_list, rewards_list, merchants_list, merchant_availability_list, and/or the like. A Behavior Data table **1419***n* may include fields such as, but not limited to: user_id, timestamp, activity_type, activity_location, activity_attribute_list, activity_attribute_values_list, and/or the like. An Analytics table **1419***o* may include fields such as, but not limited to: report_id, user_id, report_type, report_algorithm_id, report_destination_address, and/or the like.

[0163] In one embodiment, the SNAP database may interact with other database systems. For example, employing a distributed database system, queries and data access by search SNAP component may treat the combination of the SNAP database, an integrated data security layer database as a single database entity.

[0164] In one embodiment, user programs may contain various user interface primitives, which may serve to update the SNAP. Also, various accounts may require custom database tables depending upon the environments and the types of clients the SNAP may need to serve. It should be noted that any unique fields may be designated as a key field throughout. In an alternative embodiment, these tables have been decentralized into their own databases and their respective database controllers (i.e., individual database controllers for each of the above tables). Employing standard data processing techniques, one may further distribute the databases over several computer systemizations and/or storage devices. Similarly, configurations of the decentralized database controllers may be varied by consolidating and/or distributing the various database components **1419***a-o*. The SNAP may be configured to keep track of various settings, inputs, and parameters via database controllers.

[0165] The SNAP database may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the SNAP database communicates with the SNAP component, other program components, and/or the like. The database may contain, retain, and provide information regarding other nodes and data.

The SNAPs

[0166] The SNAP component **1435** is a stored program component that is executed by a CPU. In one embodiment, the SNAP component incorporates any and/or all combinations of the aspects of the SNAP discussed in the previous figures. As such, the SNAP affects accessing, obtaining and the provision of information, services, transactions, and/or the like across various communications networks.

[0167] The SNAP component may transform real-time-generated merchant-product Quick Response codes via SNAP components into virtual wallet card-based transaction purchase notifications, and/or the like and use of the SNAP. In one embodiment, the SNAP component **1435** takes inputs (e.g., checkout input **411**; product data **414**; payment input **419**; issuer server data **423**; user data **427***a-n*; and/or the like), and transforms the inputs via SNAP components (e.g., SMPE **1441**; QRCP **1442**; and/or the like), into outputs (e.g., QR pay code **417**; card authorization request **421**; authorization response **429***a-n*; authorization success message **433***a-b*; batch append data **435**; purchase receipt **436**; and/or the like).

[0168] The SNAP component enabling access of information between nodes may be developed by employing standard development tools and languages such as, but not limited to: Apache components, Assembly, ActiveX, binary executables, (ANSI) (Objective-) C (++), C# and/or .NET, database adapters, CGI scripts, Java, JavaScript, mapping tools, procedural and object oriented development tools, PERL, PHP, Python, shell scripts, SQL commands, web application server extensions, web development environments and libraries (e.g., Microsoft's ActiveX; Adobe AIR, FLEX & FLASH; AJAX; (D)HTML; Dojo, Java; JavaScript; jQuery(UI); MooTools; Prototype; script.aculo.us; Simple Object Access Protocol (SOAP); SWFObject; Yahoo! User Interface; and/or the like), WebObjects, and/or the like. In one embodiment, the SNAP server employs a cryptographic server to encrypt and decrypt communications. The SNAP component may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the SNAP component communicates with the SNAP database, operating systems, other program components, and/or the like. The SNAP may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses.

Distributed SNAPs

[0169] The structure and/or operation of any of the SNAP node controller components may be combined, consolidated, and/or distributed in any number of ways to facilitate development and/or deployment. Similarly, the component collection may be combined in any number of ways to facilitate deployment and/or development. To accomplish this, one may integrate the components into a common code base or in a facility that can dynamically load the components on demand in an integrated fashion.

[0170] The component collection may be consolidated and/or distributed in countless variations through standard data processing and/or development techniques. Multiple instances of any one of the program components in the program component collection may be instantiated on a single node, and/or across numerous nodes to improve performance through load-balancing and/or data-processing techniques. Furthermore, single instances may also be distributed across multiple controllers and/or storage devices; e.g., databases. All program component instances and controllers working in concert may do so through standard data processing communication techniques.

[0171] The configuration of the SNAP controller will depend on the context of system deployment. Factors such as, but not limited to, the budget, capacity, location, and/or use of the underlying hardware resources may affect deployment requirements and configuration. Regardless of if the configuration results in more consolidated and/or integrated program components, results in a more distributed series of program components, and/or results in some combination between a consolidated and distributed configuration, data may be communicated, obtained, and/or provided. Instances of components consolidated into a common code base from the program component collection may communicate, obtain, and/or provide data. This may be accomplished through intra-application data processing communication techniques such as, but not limited to: data referencing (e.g., pointers), internal messaging, object instance variable communication, shared memory space, variable passing, and/or the like.

[0172] If component collection components are discrete, separate, and/or external to one another, then communicating, obtaining, and/or providing data with and/or to other components may be accomplished through inter-application data processing communication techniques such as, but not limited to: Application Program Interfaces (API) information passage; (distributed) Component Object Model ((D)COM), (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.

[0173] For example, a grammar may be arranged to recognize the tokens of an HTTP post command, e.g.:

[0174] w3c -post http:// . . . Value1

[0175] where Value1 is discerned as being a parameter because "http://" is part of the grammar syntax, and what follows is considered part of the post value. Similarly, with such a grammar, a variable "Value1" may be inserted into an "http://" post command and then sent. The grammar syntax itself may be presented as structured data that is interpreted and/or otherwise used to generate the parsing mechanism (e.g., a syntax description text file as processed by lex, yacc, etc.). Also, once the parsing mechanism is generated and/or instantiated, it itself may process and/or parse structured data such as, but not limited to: character (e.g., tab) delineated text, HTML, structured text streams, XML, and/or the like structured data. In another embodiment, inter-application data processing protocols themselves may have integrated and/or readily available parsers (e.g., JSON, SOAP, and/or like parsers) that may be employed 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.

[0176] For example, in some implementations, the SNAP 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:

```
<?PHP
header('Content-Type: text/plain');
// set ip address and port to listen to for incoming data
$address = '192.168.0.100';
$port = 255;
// create a server-side SSL socket, listen for/accept incoming
communication
$sock = socket__create(AF__INET, SOCK__STREAM, 0);
socket__bind($sock, $address, $port) or die('Could not bind to address');
socket__listen($sock);
$client = socket__accept($sock);
// read input data from client device in 1024 byte blocks until end of
message
do {
    $input = "";
    $input = socket__read($client, 1024);
    $data .= $input;
} while($input != "");
// parse data to extract variables
$obj = json__decode($data, true);
// store input data in a database
mysql__connect("201.408.185.132",$DBserver,$password); // access
database server
mysql__select("CLIENT__DB.SQL"); // select database to append
mysql__query("INSERT INTO UserTable (transmission)
VALUES ($data)"); // add data to UserTable table in a CLIENT database
mysql__close("CLIENT__DB.SQL"); // close connection to database
?>
```

[0177] Also, the following resources may be used to provide example embodiments regarding SOAP parser implementation:

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

[0178] and other parser implementations:

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

[0179] all of which are hereby expressly incorporated by reference herein.

[0180] Non-limiting exemplary embodiments highlighting numerous further advantageous aspects include:

[0181] 1. A visual snap computer-implemented method, comprising:

[0182] obtaining a user visual item information request (VITR);

[0183] extracting a VITR image frame from the VITR obtained via an image acquisition device operatively connected to a user device;

[0184] extracting VITR field attributes from the VITR image frame through a VITR processing component, wherein the VITR processing component may include barcode recognition and optical character recognition;

[0185] querying a database with the extracted VITR field attributes;

[0186] determining a VITR type from results from the querying;

[0187] providing a visual item information response based on the determined VITR type.

[0188] 2. The method of claim 1, wherein the VITR may be one of a visual coupon addition request, a visual account addition request, a visual bill addition request, a visual purchase request, and a visual purchase information request.

[0189] 3. The method of claim 2, wherein the VITR and visual item information response are storable for later access and retrieval via a VITR history.

[0190] 4. The method of claim 1, wherein the VITR is a visual purchase request, and the provided visual item information response is a purchase intake mechanism, and further comprising:

[0191] obtaining the user's indication to purchase a purchase item identified in the provided visual item information response;

[0192] obtaining a purchase transaction request for payment processing of the purchase item; and

[0193] providing a purchase receipt for the purchase transaction.

[0194] 5. The method of claim 4, wherein the VITR, visual item information response, user's indication, purchase transaction request and purchase receipt are storable for later access and retrieval via a VITR history.

[0195] 6. The method of claim 1, wherein a server performs VITR activities.

[0196] 7. The method of claim 1, wherein a user client device performs VITR activities.

[0197] 8. A snap payment computer-implemented method, comprising:

[0198] obtaining a user visual item information request (VITR);

[0199] acquiring an VITR image frame via an image acquisition device operatively connected to the user device;

[0200] extracting VITR field attributes from the VITR image frame through a VITR processing component, wherein the VITR processing component may include barcode recognition and optical character recognition;

[0201] querying a database with the extracted VITR field attributes;

[0202] determining a VITR type from results from the querying;

[0203] providing a visual item information response based on the determined VITR type.

[0204] 9. The method of claim 8, wherein the VITR may be one of a visual coupon addition request, a visual account addition request, a visual bill addition request, a visual purchase request, and a visual purchase information request.

[0205] 10. The method of claim 9, wherein the VITR and visual item information response are storable for later access and retrieval via a VITR history.

[0206] 11. The method of claim 8, wherein the VITR is a visual purchase request, and the provided visual item information response is a purchase intake mechanism, and further comprising:

[0207] obtaining the user's indication to purchase a purchase item identified in the provided visual item information response;

[0208] providing a purchase transaction request for payment processing of the purchase item; and

[0209] obtaining a purchase receipt for the purchase transaction.

[0210] 12. The method of claim **11**, wherein the VITR, visual item information response, user's indication, purchase transaction request and purchase receipt are storable for later access and retrieval via a VITR history.

[0211] 13. A snap payment computer-implemented method, comprising:

[0212] obtaining, at a user device, a user input to initiate a purchase transaction;

[0213] acquiring an image frame via an image acquisition device operatively connected to the user device;

[0214] identifying a payment code depicted within the acquired image frame;

[0215] generating, via the user device, a purchase transaction request using the identified payment code;

[0216] providing the purchase transaction request for payment processing; and

[0217] obtaining a purchase receipt for the purchase transaction.

[0218] 14. The method of claim 13, further comprising:

[0219] providing an image of the payment code for purchase transaction processing.

[0220] 15. The method of claim 13, further comprising:

[0221] acquiring video including the image frame via the image acquisition device included in the user mobile device;

[0222] extracting the image frame from the acquired video; and

[0223] analyzing the image frame to determine whether the image frame includes the depicted payment code.

[0224] 16. The method of claim 13, wherein the user device is a mobile device.

[0225] 17. The method of claim 13, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

[0226] 18. The method of claim 13, wherein the payment code is a one-dimensional barcode.

[0227] 19. The method of claim 13, wherein the payment code is a two-dimensional barcode.

[0228] 20. The method of claim 19, wherein the payment code is a Quick Response code.

[0229] 21. The method of claim 13, further comprising:

[0230] extracting purchase session data from the payment code; and

[0231] wherein the purchase transaction request is generated, via the user mobile device, using the extracted purchase session data.

[0232] 22. The method of claim 21, wherein the purchase session data varies based on user shopping activity with a merchant.

[0233] 23. The method of claim 22, wherein the merchant is an online merchant.

[0234] 24. The method of claim 13, further comprising:

[0235] providing a portion of the acquired image frame including the depiction of the payment code to a server; and

[0236] obtaining purchase session data from the server in response to providing the portion of the acquired image frame.

[0237] 25. The method of claim 21, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0238] 26. The method of claim 24, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0239] 27. The method of claim 25, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0240] 28. The method of claim 26, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0241] 29. The method of claim 13, further comprising:

[0242] obtaining, for payment processing, payment information associated with a virtual wallet account;

[0243] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0244] 30. The method of claim 29, wherein the payment information includes a dynamically generated card verification value code.

[0245] 31. The method of claim 30, further comprising:

[0246] providing a request for the dynamically generated card verification value code to a server; and

[0247] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[0248] 32. The method of claim 31, wherein the dynamically generated card verification value has an expiration time.

[0249] 33. The method of claim 31, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0250] 34. The method of claim 13, wherein the payment code depicted within the acquired image frame is acquired from the display of a media device, and encodes data to purchase on-demand media content.

[0251] 35. The method of claim 34, wherein the media device is a television.

[0252] 36. The method of claim 35, wherein the television is part of an in-flight entertainment system.

[0253] 37. The method of claim 34, wherein the media device is displaying a webpage.

[0254] 38. A reverse snap payment computer-implemented method, comprising:

[0255] obtaining, at a user device, a user input to initiate a purchase transaction with a merchant;

[0256] obtaining user payment information for processing the purchase transaction;

[0257] generating, via the user device, a payment code image using the payment information for processing the purchase transaction;

[0258] displaying the payment code image, via a display operatively connected to the user device, for a point-of-sale terminal to acquire an image of the payment code image; and

[0259] obtaining a purchase receipt for the purchase transaction.

[0260] 39. The method of claim 38, further comprising:

[0261] obtaining a notification that the point-of-sale terminal has acquired an image of the payment code image; and

[0262] terminating display of the payment code image via the display operatively connected to the user device;

[0263] 40. The method of claim 38, wherein the user device is a mobile device.

[0264] 41. The method of claim 38, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

[0265] 42. The method of claim 38, wherein the payment code is a one-dimensional barcode.

[0266] 43. The method of claim 38, wherein the payment code is a two-dimensional barcode.

[0267] 44. The method of claim 43, wherein the payment code is a Quick Response code.

[0268] 45. The method of claim 38, wherein the merchant is an online merchant.

[0269] 46. The method of claim 38, wherein the purchase receipt includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0270] 47. The method of claim 46, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0271] 48. The method of claim 38, wherein the user payment information is associated with a virtual wallet account;

[0272] 49. The method of claim 48, wherein the payment information includes a dynamically generated card verification value code.

[0273] 50. The method of claim 49, further comprising:

[0274] providing a request for the dynamically generated card verification value code to a server; and

[0275] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[0276] 51. The method of claim 49, wherein the dynamically generated card verification value has an expiration time.

[0277] 52. The method of claim 49, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0278] 53. The method of claim 38, wherein the point-of-sale terminal is a user device.

[0279] 54. The method of claim 38, wherein the point-of-sale terminal is located at physical merchant store.

[0280] 55. A group split snap payment computer-implemented method, comprising:

[0281] obtaining, at a user device of a user, a user input to initiate a group purchase transaction;

[0282] obtaining purchase data for the group purchase transaction;

[0283] generating, via the user device, a split-payment code image using the purchase data for the group purchase transaction;

[0284] wherein the split-payment code image includes information on a payment amount for another user; and

[0285] displaying the split-payment code image, via a display operatively connected to the user device, for another user device of the another user to acquire an image of the split-payment code image.

[0286] 56. The method of claim 55, further comprising:

[0287] generating, via the user device, a purchase transaction request using a payment amount for the user covering a portion of the group purchase transaction;

[0288] providing the purchase transaction request for payment processing; and

[0289] obtaining a purchase receipt for the payment amount for the user for the group purchase transaction.

[0290] 57. The method of claim 56, further comprising:

[0291] obtaining, for payment processing, payment information associated with a virtual wallet account;

[0292] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0293] 58. The method of claim 57, wherein the payment information includes a dynamically generated card verification value code.

[0294] 59. The method of claim 58, further comprising:

[0295] providing a request for the dynamically generated card verification value code to a server; and

[0296] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[0297] 60. The method of claim 58, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0298] 61. The method of claim 55, wherein the split-payment code is a Quick Response code.

[0299] 62. A group split payment computer-implemented method, comprising

[0300] obtaining, at a user device of a user, a user input to initiate a group purchase transaction;

[0301] acquiring an image frame via an image acquisition device operatively connected to the user device;

[0302] identifying a payment code depicted within the acquired image frame, the payment code displayed by another user device of another user;

[0303] generating, via the user device, a purchase transaction request for payment covering a portion of the group purchase transaction using the identified payment code;

[0304] providing the purchase transaction request for payment processing; and

[0305] obtaining a purchase receipt for the purchase transaction.

[0306] 63. The method of claim 62, further comprising:

[0307] extracting purchase session data from the payment code; and

[0308] wherein the purchase transaction request is generated, via the user mobile device, using the extracted purchase session data.

[0309] 64. The method of claim 63, wherein the purchase session data varies based on user shopping activity with a merchant.

[0310] 65. The method of claim 64, wherein the merchant is an online merchant.

[0311] 66. The method of claim 62, further comprising:

[0312] providing a portion of the acquired image frame including the depiction of the payment code to a server; and

[0313] obtaining purchase session data from the server in response to providing the portion of the acquired image frame.

[0314] 67. The method of claim 63, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0315] 68. The method of claim 66, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0316] 69. The method of claim 67, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0317] 70. The method of claim 68, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0318] 71. The method of claim 62, further comprising:

[0319] obtaining, for payment processing, payment information associated with a virtual wallet account;

[0320] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0321] 72. The method of claim 71, wherein the payment information includes a dynamically generated card verification value code.

[0322] 73. The method of claim 72, further comprising:

[0323] providing a request for the dynamically generated card verification value code to a server; and

[0324] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[0325] 74. The method of claim 73, wherein the dynamically generated card verification value has an expiration time.

[0326] 75. The method of claim 73, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0327] 76. A person-to-person snap payment computer-implemented method, comprising:

[0328] obtaining, at a user device of a user, a user input to initiate a person-to-person transaction;

[0329] obtaining a transfer amount for the person-to-person transaction;

[0330] generating, via the user device, a payment code image using the transfer amount for the person-to-person transaction;

[0331] wherein the payment code image includes information on a transfer amount for another user; and

[0332] displaying the payment code image, via a display operatively connected to the user device, for another user device of the another user to acquire an image of the payment code image.

[0333] 77. The method of claim 76, further comprising:

[0334] obtaining, for payment processing, payment information associated with a virtual wallet account;

[0335] wherein the generated payment code image encodes information on the payment information associated with the virtual wallet account.

[0336] 78. The method of claim 77, wherein the payment information includes a dynamically generated card verification value code.

[0337] 79. The method of claim 78, further comprising:

[0338] providing a request for the dynamically generated card verification value code to a server; and

[0339] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[0340] 80. The method of claim 76, wherein the transfer amount is obtained from the another user device of the another user.

[0341] 81. The method of claim 76, wherein the split-payment code is a Quick Response code.

[0342] 82. A person-to-person payment computer-implemented method, comprising

[0343] obtaining, at a user device of a user, a user input to initiate a person-to-person transaction;

[0344] acquiring an image frame via an image acquisition device operatively connected to the user device;

[0345] identifying a payment code depicted within the acquired image frame, the payment code displayed by another user device of another user;

[0346] generating, via the user device and using the identified payment code, a payment transfer request for payment to the another user;

[0347] providing the payment transfer request for payment processing; and

[0348] obtaining a transfer confirmation for the person-to-person transaction.

[0349] 83. The method of claim 82, further comprising:

[0350] extracting transfer account data from the payment code; and

[0351] wherein the payment transfer request is generated, via the user mobile device, using the extracted transfer account data.

[0352] 84. The method of claim 83, wherein the transfer account data includes data on a virtual wallet account.

[0353] 85. The method of claim 82, further comprising:

[0354] providing a portion of the acquired image frame including the depiction of the payment code to a server.

[0355] 86. The method of claim 82, further comprising:

[0356] obtaining, for payment processing, payment information associated with a virtual wallet account;

[0357] wherein the generated payment transfer request includes the payment information associated with the virtual wallet account.

[0358] 87. The method of claim 86, wherein the payment information includes a dynamically generated card verification value code.

[0359] 88. The method of claim 87, further comprising:

[0360] providing a request for the dynamically generated card verification value code to a server; and

[0361] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[0362] 89. The method of claim 87, wherein the dynamically generated card verification value has an expiration time.

[0363] 90. The method of claim 87, wherein the dynamically generated card verification value is specific to a user funds transfer session between the user device and the another user device.

[0364] 91. A snap mobile sales computer-implemented method, comprising:

[0365] obtaining a user checkout request at a point-of-sale device;

[0366] obtaining user shopping cart information with a merchant for processing a purchase transaction related to the user checkout request;

[0367] generating, via the user device, a payment code image using the user shopping cart information;

[0368] displaying the payment code image, via a display operatively connected to the point-of-sale device, for a user device to acquire an image of the payment code image; and

[0369] obtaining a notification of authorization of the purchase transaction.

[0370] 92. The method of claim 91, further comprising:

[0371] obtaining a notification that the user device has acquired an image of the payment code image; and

[0372] terminating display of the payment code image via the display operatively connected to the point-of-sale device;

[0373] 93. The method of claim 91, wherein the user checkout request is obtained via a touchscreen gesture on a touchscreen operatively connected to the point-of-sale device.

[0374] 94. The method of claim 91, wherein the user checkout request is obtained via a communication from the user device.

[0375] 95. The method of claim 91, wherein the payment code is a one-dimensional barcode.

[0376] 96. The method of claim 91, wherein the payment code is a two-dimensional barcode.

[0377] 97. The method of claim 96, wherein the payment code is a Quick Response code.

[0378] 98. The method of claim 91, wherein the merchant is an online merchant.

[0379] 99. The method of claim 98, wherein the point-of sale device is another user device.

[0380] 100. The method of claim 91, wherein the point-of-sale terminal is located at physical merchant store.

[0381] 101. The method of claim 91, wherein the notification of authorization of the purchase transaction includes a session identifier for a user shopping session with the merchant.

[0382] 102. The method of claim 101, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0383] 103. A reverse snap mobile sales computer-implemented method, comprising:

[0384] obtaining a user checkout request at a point-of-sale device;

[0385] acquiring an image frame via an image acquisition device operatively connected to the point-of-sale device;

[0386] identifying a payment code depicted within the acquired image frame;

[0387] generating, via the point-of-sale device, a purchase transaction request using the identified payment code;

[0388] providing the purchase transaction request for payment processing; and

[0389] obtaining a notification of authorization of the purchase transaction.

[0390] 104. The method of claim 103, further comprising:

[0391] providing an image of the payment code for purchase transaction processing.

[0392] 105. The method of claim 103, further comprising:

[0393] acquiring video including the image frame via the image acquisition device operatively connected to the point-of-sale device;

[0394] extracting the image frame from the acquired video; and

[0395] analyzing the image frame to determine whether the image frame includes the depicted payment code.

[0396] 106. The method of claim 103, wherein the payment code is a one-dimensional barcode.

[0397] 107. The method of claim 103, wherein the payment code is a two-dimensional barcode.

[0398] 108. The method of claim 103, wherein the payment code is a Quick Response code.

[0399] 109. The method of claim 103, further comprising:

[0400] extracting purchase payment information from the payment code; and

[0401] wherein the purchase transaction request is generated, via the point-of-sale device, using the extracted purchase payment information.

[0402] 110. The method of claim 109, wherein the purchase payment information includes an expiration time.

[0403] 111. The method of claim 109, wherein the purchase payment information is associated with a virtual wallet account, and wherein the generated purchase transaction request includes the purchase payment data associated with the virtual wallet account.

[0404] 112. The method of claim 103, further comprising:

[0405] providing a portion of the acquired image frame including the depiction of the payment code to a server; and

[0406] obtaining purchase payment information from the server in response to providing the portion of the acquired image frame.

[0407] 113. The method of claim 112, wherein the purchase payment information includes a session identifier for a user shopping session with a merchant.

[0408] 114. The method of claim 113, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0409] 115. A visual snap computer-implemented system, comprising:

[0410] a processor; and

[0411] a memory disposed in communication with the processor and storing processor-executable instructions to:

- **[0412]** obtain a user visual item information request (VITR);
- **[0413]** extract a VITR image frame from the VITR obtained via an image acquisition device operatively connected to a user device;
- **[0414]** extract VITR field attributes from the VITR image frame through a VITR processing component, wherein the VITR processing component may include barcode recognition and optical character recognition;
- **[0415]** query a database with the extracted VITR field attributes;
- **[0416]** determine a VITR type from results from the querying;
- **[0417]** provide a visual item information response based on the determined VITR type.

[0418] 116. The system of claim 115, wherein the VITR may be one of a visual coupon addition request, a visual account addition request, a visual bill addition request, a visual purchase request, and a visual purchase information request.

[0419] 117. The system of claim 116, wherein the VITR and visual item information response are storable for later access and retrieval via a VITR history.

[0420] 118. The system of claim 115, wherein the VITR is a visual purchase request, and the provided visual item information response is a purchase intake mechanism, and further comprising:

[0421] obtain the user's indication to purchase a purchase item identified in the provided visual item information response;

[0422] obtain a purchase transaction request for payment processing of the purchase item; and

[0423] provide a purchase receipt for the purchase transaction.

[0424] 119. The system of claim 118, wherein the VITR, visual item information response, user's indication, purchase transaction request and purchase receipt are storable for later access and retrieval via a VITR history.

[0425] 120. The system of claim 115, wherein a server performs VITR activities.

[0426] 121. The system of claim 115, wherein a user client device performs VITR activities.

[0427] 122. A snap payment computer-implemented system, comprising:

[0428] a processor; and

[0429] a memory disposed in communication with the processor and storing processor-executable instructions to:

- **[0430]** obtain a user visual item information request (VITR);
- **[0431]** acquire an VITR image frame via an image acquisition device operatively connected to the user device;
- **[0432]** extract VITR field attributes from the VITR image frame through a VITR processing component, wherein the VITR processing component may include barcode recognition and optical character recognition;

[0433] query a database with the extracted VITR field attributes;

[0434] determine a VITR type from results from the querying;

[0435] provide a visual item information response based on the determined VITR type.

[0436] 123. The system of claim 122, wherein the VITR may be one of a visual coupon addition request, a visual account addition request, a visual bill addition request, a visual purchase request, and a visual purchase information request.

[0437] 124. The system of claim 123, wherein the VITR and visual item information response are storable for later access and retrieval via a VITR history.

[0438] 125. The system of claim 122, wherein the VITR is a visual purchase request, and the provided visual item information response is a purchase intake mechanism, and further comprising:

[0439] obtain the user's indication to purchase a purchase item identified in the provided visual item information response;

[0440] provide a purchase transaction request for payment processing of the purchase item; and

[0441] obtain a purchase receipt for the purchase transaction.

[0442] 126. The system of claim 125, wherein the VITR, visual item information response, user's indication, purchase transaction request and purchase receipt are storable for later access and retrieval via a VITR history.

[0443] 127. A snap payment computer-implemented system, comprising:

[0444] a processor; and

[0445] a memory disposed in communication with the processor and storing processor-executable instructions to:

[0446] obtain, at a user device, a user input to initiate a purchase transaction;

[0447] acquire an image frame via an image acquisition device operatively connected to the user device;

[0448] identify a payment code depicted within the acquired image frame;

[0449] generate, via the user device, a purchase transaction request using the identified payment code;

[0450] provide the purchase transaction request for payment processing; and

[0451] obtain a purchase receipt for the purchase transaction.

[0452] 128. The system of claim 127, the memory further storing instructions to:

[0453] provide an image of the payment code for purchase transaction processing.

[0454] 129. The system of claim 127, the memory further storing instructions to:

[0455] acquire video including the image frame via the image acquisition device included in the user mobile device;

[0456] extract the image frame from the acquired video; and

[0457] analyzing the image frame to determine whether the image frame includes the depicted payment code.

[0458] 130. The system of claim 127, wherein the user device is a mobile device.

[0459] 131. The system of claim 127, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

[0460] 132. The system of claim 127, wherein the payment code is a one-dimensional barcode.

[0461] 133. The system of claim 127, wherein the payment code is a two-dimensional barcode.

[0462] 134. The system of claim 133, wherein the payment code is a Quick Response code.

[0463] 135. The system of claim 127, the memory further storing instructions to:

[0464] extract purchase session data from the payment code; and

[0465] wherein the purchase transaction request is generated, via the user mobile device, using the extracted purchase session data.

[0466] 136. The system of claim 135, wherein the purchase session data varies based on user shopping activity with a merchant.

[0467] 137. The system of claim 136, wherein the merchant is an online merchant.

[0468] 138. The system of claim 127, the memory further storing instructions to:

[0469] provide a portion of the acquired image frame including the depiction of the payment code to a server; and

[0470] obtain purchase session data from the server in response to providing the portion of the acquired image frame.

[0471] 139. The system of claim 135, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0472] 140. The system of claim 138, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0473] 141. The system of claim 139, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0474] 142. The system of claim 140, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0475] 143. The system of claim 127, the memory further storing instructions to:

[0476] obtain, for payment processing, payment information associated with a virtual wallet account;

[0477] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0478] 144. The system of claim 143, wherein the payment information includes a dynamically generated card verification value code.

[0479] 145. The system of claim 144, the memory further storing instructions to:

[0480] provide a request for the dynamically generated card verification value code to a server; and

[0481] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0482] 146. The system of claim 145, wherein the dynamically generated card verification value has an expiration time.

[0483] 147. The system of claim 145, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0484] 148. The system of claim 127, wherein the payment code depicted within the acquired image frame is acquired from the display of a media device, and encodes data to purchase on-demand media content.

[0485] 149. The system of claim 148, wherein the media device is a television.

[0486] 150. The system of claim 149, wherein the television is part of an in-flight entertainment system.

[0487] 151. The system of claim 148, wherein the media device is displaying a webpage.

[0488] 152. A reverse snap payment computer-implemented system, comprising:

[0489] a processor; and

[0490] a memory disposed in communication with the processor and storing processor-executable instructions to:

- **[0491]** obtain, at a user device, a user input to initiate a purchase transaction with a merchant;
- **[0492]** obtain user payment information for processing the purchase transaction;
- **[0493]** generate, via the user device, a payment code image using the payment information for processing the purchase transaction;
- **[0494]** display the payment code image, via a display operatively connected to the user device, for a point-of-sale terminal to acquire an image of the payment code image; and
- **[0495]** obtain a purchase receipt for the purchase transaction.

[0496] 153. The system of claim 152, the memory further storing instructions to:

[0497] obtain a notification that the point-of-sale terminal has acquired an image of the payment code image; and

[0498] terminate display of the payment code image via the display operatively connected to the user device;

[0499] 154. The system of claim 152, wherein the user device is a mobile device.

[0500] 155. The system of claim 152, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

[0501] 156. The system of claim 152, wherein the payment code is a one-dimensional barcode.

[0502] 157. The system of claim 152, wherein the payment code is a two-dimensional barcode.

[0503] 158. The system of claim 157, wherein the payment code is a Quick Response code.

[0504] 159. The system of claim 152, wherein the merchant is an online merchant.

[0505] 160. The system of claim 152, wherein the purchase receipt includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0506] 161. The system of claim 160, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0507] 162. The system of claim 152, wherein the user payment information is associated with a virtual wallet account;

[0508] 163. The system of claim 162, wherein the payment information includes a dynamically generated card verification value code.

[0509] 164. The system of claim 163, the memory further storing instructions to:

[0510] provide a request for the dynamically generated card verification value code to a server; and

[0511] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0512] 165. The system of claim 163, wherein the dynamically generated card verification value has an expiration time.

[0513] 166. The system of claim 163, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0514] 167. The system of claim 152, wherein the point-of-sale terminal is a user device.

[0515] 168. The system of claim 152, wherein the point-of-sale terminal is located at physical merchant store.

[0516] 169. A group split snap payment computer-implemented system, comprising:

[0517] a processor; and

[0518] a memory disposed in communication with the processor and storing processor-executable instructions to:

- **[0519]** obtain, at a user device of a user, a user input to initiate a group purchase transaction;
- **[0520]** obtain purchase data for the group purchase transaction;
- **[0521]** generate, via the user device, a split-payment code image using the purchase data for the group purchase transaction;
- **[0522]** wherein the split-payment code image includes information on a payment amount for another user; and
- **[0523]** display the split-payment code image, via a display operatively connected to the user device, for another user device of the another user to acquire an image of the split-payment code image.

[0524] 170. The system of claim 169, the memory further storing instructions to:

[0525] generate, via the user device, a purchase transaction request using a payment amount for the user covering a portion of the group purchase transaction;

[0526] provide the purchase transaction request for payment processing; and

[0527] obtain a purchase receipt for the payment amount for the user for the group purchase transaction.

[0528] 171. The system of claim 170, the memory further storing instructions to:

[0529] obtain, for payment processing, payment information associated with a virtual wallet account;

[0530] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0531] 172. The system of claim 171, wherein the payment information includes a dynamically generated card verification value code.

[0532] 173. The system of claim 172, the memory further storing instructions to:

[0533] provide a request for the dynamically generated card verification value code to a server; and

[0534] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0535] 174. The system of claim 172, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0536] 175. The system of claim 169, wherein the split-payment code is a Quick Response code.

[0537] 176. A group split payment computer-implemented system, comprising:

[0538] a processor; and

[0539] a memory disposed in communication with the processor and storing processor-executable instructions to:

[0540] obtain, at a user device of a user, a user input to initiate a group purchase transaction;

[0541] acquire an image frame via an image acquisition device operatively connected to the user device;

[0542] identify a payment code depicted within the acquired image frame, the payment code displayed by another user device of another user;

[0543] generate, via the user device, a purchase transaction request for payment covering a portion of the group purchase transaction using the identified payment code;

[0544] provide the purchase transaction request for payment processing; and

[0545] obtain a purchase receipt for the purchase transaction.

[0546] 177. The system of claim 176, the memory further storing instructions to:

[0547] extract purchase session data from the payment code; and

[0548] wherein the purchase transaction request is generated, via the user mobile device, using the extracted purchase session data.

[0549] 178. The system of claim 177, wherein the purchase session data varies based on user shopping activity with a merchant.

[0550] 179. The system of claim 178, wherein the merchant is an online merchant.

[0551] 180. The system of claim 176, the memory further storing instructions to:

[0552] provide a portion of the acquired image frame including the depiction of the payment code to a server; and

[0553] obtain purchase session data from the server in response to providing the portion of the acquired image frame.

[0554] 181. The system of claim 177, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0555] 182. The system of claim 180, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0556] 183. The system of claim 181, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0557] 184. The system of claim 182, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0558] 185. The system of claim 176, the memory further storing instructions to:

[0559] obtain, for payment processing, payment information associated with a virtual wallet account;

[0560] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0561] 186. The system of claim 185, wherein the payment information includes a dynamically generated card verification value code.

[0562] 187. The system of claim 186, the memory further storing instructions to:

[0563] provide a request for the dynamically generated card verification value code to a server; and

[0564] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0565] 188. The system of claim 187, wherein the dynamically generated card verification value has an expiration time.

[0566] 189. The system of claim 187, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0567] 190. A person-to-person snap payment computer-implemented system, comprising:

[0568] a processor; and

[0569] a memory disposed in communication with the processor and storing processor-executable instructions to:

[0570] obtain, at a user device of a user, a user input to initiate a person-to-person transaction;

[0571] obtain a transfer amount for the person-to-person transaction;

[0572] generate, via the user device, a payment code image using the transfer amount for the person-to-person transaction;

[0573] wherein the payment code image includes information on a transfer amount for another user; and

[0574] display the payment code image, via a display operatively connected to the user device, for another user device of the another user to acquire an image of the payment code image.

[0575] 191. The system of claim 190, the memory further storing instructions to:

[0576] obtain, for payment processing, payment information associated with a virtual wallet account;

[0577] wherein the generated payment code image encodes information on the payment information associated with the virtual wallet account.

[0578] 192. The system of claim 191, wherein the payment information includes a dynamically generated card verification value code.

[0579] 193. The system of claim 192, the memory further storing instructions to:

[0580] provide a request for the dynamically generated card verification value code to a server; and

[0581] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0582] 194. The system of claim 190, wherein the transfer amount is obtained from the another user device of the another user.

[0583] 195. The system of claim 190, wherein the split-payment code is a Quick Response code.

[0584] 196. A person-to-person payment computer-implemented system, comprising:

[0585] a processor; and

[0586] a memory disposed in communication with the processor and storing processor-executable instructions to:

[0587] obtain, at a user device of a user, a user input to initiate a person-to-person transaction;

[0588] acquire an image frame via an image acquisition device operatively connected to the user device;

[0589] identify a payment code depicted within the acquired image frame, the payment code displayed by another user device of another user;

[0590] generate, via the user device and using the identified payment code, a payment transfer request for payment to the another user;

[0591] provide the payment transfer request for payment processing; and

[0592] obtain a transfer confirmation for the person-to-person transaction.

[0593] 197. The system of claim 196, the memory further storing instructions to:

[0594] extract transfer account data from the payment code; and

[0595] wherein the payment transfer request is generated, via the user mobile device, using the extracted transfer account data.

[0596] 198. The system of claim 197, wherein the transfer account data includes data on a virtual wallet account.

[0597] 199. The system of claim 196, the memory further storing instructions to:

[0598] provide a portion of the acquired image frame including the depiction of the payment code to a server.

[0599] 200. The system of claim 196, the memory further storing instructions to:

[0600] obtain, for payment processing, payment information associated with a virtual wallet account;

[0601] wherein the generated payment transfer request includes the payment information associated with the virtual wallet account.

[0602] 201. The system of claim 200, wherein the payment information includes a dynamically generated card verification value code.

[0603] 202. The system of claim 201, the memory further storing instructions to:

[0604] provide a request for the dynamically generated card verification value code to a server; and

[0605] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0606] 203. The system of claim 201, wherein the dynamically generated card verification value has an expiration time.

[0607] 204. The system of claim 201, wherein the dynamically generated card verification value is specific to a user funds transfer session between the user device and the another user device.

[0608] 205. A snap mobile sales computer-implemented system, comprising:

[0609] a processor; and

[0610] a memory disposed in communication with the processor and storing processor-executable instructions to:

[0611] obtain a user checkout request at a point-of-sale device;

[0612] obtain user shopping cart information with a merchant for processing a purchase transaction related to the user checkout request;

[0613] generating, via the user device, a payment code image using the user shopping cart information;

[0614] display the payment code image, via a display operatively connected to the point-of-sale device, for a user device to acquire an image of the payment code image; and

[0615] obtain a notification of authorization of the purchase transaction.

[0616] 206. The system of claim 205, the memory further storing instructions to:

[0617] obtain a notification that the user device has acquired an image of the payment code image; and

[0618] terminating display of the payment code image via the display operatively connected to the point-of-sale device;

[0619] 207. The system of claim 205, wherein the user checkout request is obtained via a touchscreen gesture on a touchscreen operatively connected to the point-of-sale device.

[0620] 208. The system of claim 205, wherein the user checkout request is obtained via a communication from the user device.

[0621] 209. The system of claim 205, wherein the payment code is a one-dimensional barcode.

[0622] 210. The system of claim 205, wherein the payment code is a two-dimensional barcode.

[0623] 211. The system of claim 210, wherein the payment code is a Quick Response code.

[0624] 212. The system of claim 205, wherein the merchant is an online merchant.

[0625] 213. The system of claim 212, wherein the point-of sale device is another user device.

[0626] 214. The system of claim 205, wherein the point-of-sale terminal is located at physical merchant store.

[0627] 215. The system of claim 205, wherein the notification of authorization of the purchase transaction includes a session identifier for a user shopping session with the merchant.

[0628] 216. The system of claim 215, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0629] 217. A reverse snap mobile sales computer-implemented system, comprising:

[0630] a processor; and

[0631] a memory disposed in communication with the processor and storing processor-executable instructions to:

[0632] obtain a user checkout request at a point-of-sale device;

[0633] acquiring an image frame via an image acquisition device operatively connected to the point-of-sale device;

[0634] identify a payment code depicted within the acquired image frame;

[0635] generating, via the point-of-sale device, a purchase transaction request using the identified payment code;

[0636] providing the purchase transaction request for payment processing; and

[0637] obtain a notification of authorization of the purchase transaction.

[0638] 218. The system of claim 217, the memory further storing instructions to:

[0639] providing an image of the payment code for purchase transaction processing.

[0640] 219. The system of claim 217, the memory further storing instructions to:

[0641] acquiring video including the image frame via the image acquisition device operatively connected to the point-of-sale device;

[0642] extract the image frame from the acquired video; and

[0643] analyzing the image frame to determine whether the image frame includes the depicted payment code.

[0644] 220. The system of claim 217, wherein the payment code is a one-dimensional barcode.

[0645] 221. The system of claim 217, wherein the payment code is a two-dimensional barcode.

[0646] 222. The system of claim 217, wherein the payment code is a Quick Response code.

[0647] 223. The system of claim 217, the memory further storing instructions to:

[0648] extract purchase payment information from the payment code; and

[0649] wherein the purchase transaction request is generated, via the point-of-sale device, using the extracted purchase payment information.

[0650] 224. The system of claim 223, wherein the purchase payment information includes an expiration time.

[0651] 225. The system of claim 223, wherein the purchase payment information is associated with a virtual wallet account, and wherein the generated purchase transaction request includes the purchase payment data associated with the virtual wallet account.

[0652] 226. The system of claim 217, the memory further storing instructions to:

[0653] providing a portion of the acquired image frame including the depiction of the payment code to a server; and

[0654] obtain purchase payment information from the server in response to providing the portion of the acquired image frame.

[0655] 227. The system of claim 226, wherein the purchase payment information includes a session identifier for a user shopping session with a merchant.

[0656] 228. The system of claim 227, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0657] 229. A computer-readable tangible medium storing computer-executable visual snap instructions to:

[0658] obtain a user visual item information request (VITR);

[0659] extract a VITR image frame from the VITR obtained via an image acquisition device operatively connected to a user device;

[0660] extract VITR field attributes from the VITR image frame through a VITR processing component, wherein the VITR processing component may include barcode recognition and optical character recognition;

[0661] query a database with the extracted VITR field attributes;

[0662] determine a VITR type from results from the querying;

[0663] provide a visual item information response based on the determined VITR type.

[0664] 230. The medium of claim 229, wherein the VITR may be one of a visual coupon addition request, a visual account addition request, a visual bill addition request, a visual purchase request, and a visual purchase information request.

[0665] 231. The medium of claim 230, wherein the VITR and visual item information response are storable for later access and retrieval via a VITR history.

[0666] 232. The medium of claim 229, wherein the VITR is a visual purchase request, and the provided visual item information response is a purchase intake mechanism, and further storing instructions to:

[0667] obtain the user's indication to purchase a purchase item identified in the provided visual item information response;

[0668] obtain a purchase transaction request for payment processing of the purchase item; and

[0669] provide a purchase receipt for the purchase transaction.

[0670] 233. The medium of claim 232, wherein the VITR, visual item information response, user's indication, purchase transaction request and purchase receipt are storable for later access and retrieval via a VITR history.

[0671] 234. The medium of claim 229, wherein a server performs VITR activities.

[0672] 235. The medium of claim 229, wherein a user client device performs VITR activities.

[0673] 236. A computer-readable tangible medium storing computer-executable snap payment instructions to:

[0674] obtain a user visual item information request (VITR);

[0675] acquire an VITR image frame via an image acquisition device operatively connected to the user device;

[0676] extract VITR field attributes from the VITR image frame through a VITR processing component, wherein the VITR processing component may include barcode recognition and optical character recognition;

[0677] query a database with the extracted VITR field attributes;

[0678] determine a VITR type from results from the querying;

[0679] provide a visual item information response based on the determined VITR type.

[0680] 237. The medium of claim 236, wherein the VITR may be one of a visual coupon addition request, a visual account addition request, a visual bill addition request, a visual purchase request, and a visual purchase information request.

[0681] 238. The medium of claim 237, wherein the VITR and visual item information response are storable for later access and retrieval via a VITR history.

[0682] 239. The medium of claim 236, wherein the VITR is a visual purchase request, and the provided visual item information response is a purchase intake mechanism, and further storing instructions to:

[0683] obtain the user's indication to purchase a purchase item identified in the provided visual item information response;

[0684] provide a purchase transaction request for payment processing of the purchase item; and

[0685] obtain a purchase receipt for the purchase transaction.

[0686] 240. The medium of claim 239, wherein the VITR, visual item information response, user's indication, purchase transaction request and purchase receipt are storable for later access and retrieval via a VITR history.

[0687] 241. A computer-readable tangible medium storing computer-executable snap payment instructions to:

[0688] obtain, at a user device, a user input to initiate a purchase transaction;

[0689] acquire an image frame via an image acquisition device operatively connected to the user device;

[0690] identify a payment code depicted within the acquired image frame;

[0691] generate, via the user device, a purchase transaction request using the identified payment code;

[0692] provide the purchase transaction request for payment processing; and

[0693] obtain a purchase receipt for the purchase transaction.

[0694] 242. The medium of claim 241, further storing instructions to:

[0695] provide an image of the payment code for purchase transaction processing.

[0696] 243. The medium of claim 241, further storing instructions to:

[0697] acquire video including the image frame via the image acquisition device included in the user mobile device;

[0698] extract the image frame from the acquired video; and

[0699] analyze the image frame to determine whether the image frame includes the depicted payment code.

[0700] 244. The medium of claim 241, wherein the user device is a mobile device.

[0701] 245. The medium of claim 241, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

[0702] 246. The medium of claim 241, wherein the payment code is a one-dimensional barcode.

[0703] 247. The medium of claim 241, wherein the payment code is a two-dimensional barcode.

[0704] 248. The medium of claim 247, wherein the payment code is a Quick Response code.

[0705] 249. The medium of claim 241, further storing instructions to:

[0706] extract purchase session data from the payment code; and

[0707] wherein the purchase transaction request is generated, via the user mobile device, using the extracted purchase session data.

[0708] 250. The medium of claim 249, wherein the purchase session data varies based on user shopping activity with a merchant.

[0709] 251. The medium of claim 250, wherein the merchant is an online merchant.

[0710] 252. The medium of claim 241, further storing instructions to:

[0711] provide a portion of the acquired image frame including the depiction of the payment code to a server; and

[0712] obtain purchase session data from the server in response to providing the portion of the acquired image frame.

[0713] 253. The medium of claim 249, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0714] 254. The medium of claim 252, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0715] 255. The medium of claim 253, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0716] 256. The medium of claim 254, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0717] 257. The medium of claim 241, further storing instructions to:

[0718] obtain, for payment processing, payment information associated with a virtual wallet account;

[0719] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0720] 258. The medium of claim 257, wherein the payment information includes a dynamically generated card verification value code.

[0721] 259. The medium of claim 258, further storing instructions to:

[0722] provide a request for the dynamically generated card verification value code to a server; and

[0723] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0724] 260. The medium of claim 259, wherein the dynamically generated card verification value has an expiration time.

[0725] 261. The medium of claim 259, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0726] 262. The medium of claim 241, wherein the payment code depicted within the acquired image frame is acquired from the display of a media device, and encodes data to purchase on-demand media content.

[0727] 263. The medium of claim 262, wherein the media device is a television.

[0728] 264. The medium of claim 263, wherein the television is part of an in-flight entertainment medium.

[0729] 265. The medium of claim 262, wherein the media device is displaying a webpage.

[0730] 266. A computer-readable tangible medium storing computer-executable reverse snap payment instructions to:

[0731] obtain, at a user device, a user input to initiate a purchase transaction with a merchant;

[0732] obtain user payment information for processing the purchase transaction;

[0733] generate, via the user device, a payment code image using the payment information for processing the purchase transaction;

[0734] display the payment code image, via a display operatively connected to the user device, for a point-of-sale terminal to acquire an image of the payment code image; and

[0735] obtain a purchase receipt for the purchase transaction.

[0736] 267. The medium of claim 266, further storing instructions to:

[0737] obtain a notification that the point-of-sale terminal has acquired an image of the payment code image; and

[0738] terminate display of the payment code image via the display operatively connected to the user device;

[0739] 268. The medium of claim 266, wherein the user device is a mobile device.

[0740] 269. The medium of claim 266, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

[0741] 270. The medium of claim 266, wherein the payment code is a one-dimensional barcode.

[0742] 271. The medium of claim 266, wherein the payment code is a two-dimensional barcode.

[0743] 272. The medium of claim 271, wherein the payment code is a Quick Response code.

[0744] 273. The medium of claim 266, wherein the merchant is an online merchant.

[0745] 274. The medium of claim 266, wherein the purchase receipt includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0746] 275. The medium of claim 274, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0747] 276. The medium of claim 266, wherein the user payment information is associated with a virtual wallet account;

[0748] 277. The medium of claim 276, wherein the payment information includes a dynamically generated card verification value code.

[0749] 278. The medium of claim 277, further storing instructions to:

[0750] provide a request for the dynamically generated card verification value code to a server; and

[0751] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0752] 279. The medium of claim 277, wherein the dynamically generated card verification value has an expiration time.

[0753] 280. The medium of claim 277, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0754] 281. The medium of claim 266, wherein the point-of-sale terminal is a user device.

[0755] 282. The medium of claim 266, wherein the point-of-sale terminal is located at physical merchant store.

[0756] 283. A computer-readable tangible medium storing computer-executable group split snap payment instructions to:

[0757] obtain, at a user device of a user, a user input to initiate a group purchase transaction;

[0758] obtain purchase data for the group purchase transaction;

[0759] generate, via the user device, a split-payment code image using the purchase data for the group purchase transaction;

[0760] wherein the split-payment code image includes information on a payment amount for another user; and

[0761] display the split-payment code image, via a display operatively connected to the user device, for another user device of the another user to acquire an image of the split-payment code image.

[0762] 284. The medium of claim 283, further storing instructions to:

[0763] generate, via the user device, a purchase transaction request using a payment amount for the user covering a portion of the group purchase transaction;

[0764] provide the purchase transaction request for payment processing; and

[0765] obtain a purchase receipt for the payment amount for the user for the group purchase transaction.

[0766] 285. The medium of claim 284, further storing instructions to:

[0767] obtain, for payment processing, payment information associated with a virtual wallet account;

[0768] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0769] 286. The medium of claim 285, wherein the payment information includes a dynamically generated card verification value code.

[0770] 287. The medium of claim 286, further storing instructions to:

[0771] provide a request for the dynamically generated card verification value code to a server; and

[0772] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0773] 288. The medium of claim 286, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0774] 289. The medium of claim 283, wherein the split-payment code is a Quick Response code.

[0775] 290. A computer-readable tangible medium storing computer-executable group split payment instructions to:

[0776] obtain, at a user device of a user, a user input to initiate a group purchase transaction;

[0777] acquire an image frame via an image acquisition device operatively connected to the user device;

[0778] identify a payment code depicted within the acquired image frame, the payment code displayed by another user device of another user;

[0779] generate, via the user device, a purchase transaction request for payment covering a portion of the group purchase transaction using the identified payment code;

[0780] provide the purchase transaction request for payment processing; and

[0781] obtain a purchase receipt for the purchase transaction.

[0782] 291. The medium of claim 290, further storing instructions to:

[0783] extract purchase session data from the payment code; and

[0784] wherein the purchase transaction request is generated, via the user mobile device, using the extracted purchase session data.

[0785] 292. The medium of claim 291, wherein the purchase session data varies based on user shopping activity with a merchant.

[0786] 293. The medium of claim 292, wherein the merchant is an online merchant.

[0787] 294. The medium of claim 290, further storing instructions to:

[0788] provide a portion of the acquired image frame including the depiction of the payment code to a server; and

[0789] obtain purchase session data from the server in response to providing the portion of the acquired image frame.

[0790] 295. The medium of claim 291, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0791] 296. The medium of claim 294, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0792] 297. The medium of claim 295, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping 4 session with the merchant.

[0793] 298. The medium of claim 296, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0794] 299. The medium of claim 290, further storing instructions to:

[0795] obtain, for payment processing, payment information associated with a virtual wallet account;

[0796] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0797] 300. The medium of claim 299, wherein the payment information includes a dynamically generated card verification value code.

[0798] 301. The medium of claim 300, further storing instructions to:

[0799] provide a request for the dynamically generated card verification value code to a server; and

[0800] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0801] 302. The medium of claim 301, wherein the dynamically generated card verification value has an expiration time.

[0802] 303. The medium of claim 301, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0803] 304. A computer-readable tangible medium storing computer-executable person-to-person snap payment instructions to:

[0804] obtain, at a user device of a user, a user input to initiate a person-to-person transaction;

[0805] obtain a transfer amount for the person-to-person transaction;

[0806] generate, via the user device, a payment code image using the transfer amount for the person-to-person transaction;

[0807] wherein the payment code image includes information on a transfer amount for another user; and

[0808] display the payment code image, via a display operatively connected to the user device, for another user device of the another user to acquire an image of the payment code image.

[0809] 305. The medium of claim 304, further storing instructions to:

[0810] obtain, for payment processing, payment information associated with a virtual wallet account;

[0811] wherein the generated payment code image encodes information on the payment information associated with the virtual wallet account.

[0812] 306. The medium of claim 305, wherein the payment information includes a dynamically generated card verification value code.

[0813] 307. The medium of claim 306, further storing instructions to:

[0814] provide a request for the dynamically generated card verification value code to a server; and

[0815] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0816] 308. The medium of claim 304, wherein the transfer amount is obtained from the another user device of the another user.

[0817] 309. The medium of claim 304, wherein the split-payment code is a Quick Response code.

[0818] 310. A computer-readable tangible medium storing computer-executable person-to-person payment instructions to:

[0819] obtain, at a user device of a user, a user input to initiate a person-to-person transaction;

[0820] acquire an image frame via an image acquisition device operatively connected to the user device;

[0821] identify a payment code depicted within the acquired image frame, the payment code displayed by another user device of another user;

[0822] generate, via the user device and using the identified payment code, a payment transfer request for payment to the another user;

[0823] provide the payment transfer request for payment processing; and

[0824] obtain a transfer confirmation for the person-to-person transaction.

[0825] 311. The medium of claim 310, further storing instructions to:

[0826] extract transfer account data from the payment code; and

[0827] wherein the payment transfer request is generated, via the user mobile device, using the extracted transfer account data.

[0828] 312. The medium of claim 311, wherein the transfer account data includes data on a virtual wallet account.

[0829] 313. The medium of claim 310, further storing instructions to:

[0830] provide a portion of the acquired image frame including the depiction of the payment code to a server.

[0831] 314. The medium of claim 310, further storing instructions to:

[0832] obtain, for payment processing, payment information associated with a virtual wallet account;

[0833] wherein the generated payment transfer request includes the payment information associated with the virtual wallet account.

[0834] 315. The medium of claim 314, wherein the payment information includes a dynamically generated card verification value code.

[0835] 316. The medium of claim 315, further storing instructions to:

[0836] provide a request for the dynamically generated card verification value code to a server; and

[0837] obtain the dynamically generated card verification value code from the server in response to providing the request.

[0838] 317. The medium of claim 315, wherein the dynamically generated card verification value has an expiration time.

[0839] 318. The medium of claim 315, wherein the dynamically generated card verification value is specific to a user funds transfer session between the user device and the another user device.

[0840] 319. A computer-readable tangible medium storing computer-executable snap mobile sales instructions to:

[0841] obtain a user checkout request at a point-of-sale device;

[0842] obtain user shopping cart information with a merchant for processing a purchase transaction related to the user checkout request;

[0843] generate, via the user device, a payment code image using the user shopping cart information;

[0844] display the payment code image, via a display operatively connected to the point-of-sale device, for a user device to acquire an image of the payment code image; and

[0845] obtain a notification of authorization of the purchase transaction.

[0846] 320. The medium of claim 319, further storing instructions to:

[0847] obtain a notification that the user device has acquired an image of the payment code image; and

[0848] terminate display of the payment code image via the display operatively connected to the point-of-sale device;

[0849] 321. The medium of claim 319, wherein the user checkout request is obtained via a touchscreen gesture on a touchscreen operatively connected to the point-of-sale device.

[0850] 322. The medium of claim 319, wherein the user checkout request is obtained via a communication from the user device.

[0851] 323. The medium of claim 319, wherein the payment code is a one-dimensional barcode.

[0852] 324. The medium of claim 319, wherein the payment code is a two-dimensional barcode.

[0853] 325. The medium of claim 324, wherein the payment code is a Quick Response code.

[0854] 326. The medium of claim 319, wherein the merchant is an online merchant.

[0855] 327. The medium of claim 326, wherein the point-of sale device is another user device.

[0856] 328. The medium of claim 319, wherein the point-of-sale terminal is located at physical merchant store.

[0857] 329. The medium of claim 319, wherein the notification of authorization of the purchase transaction includes a session identifier for a user shopping session with the merchant.

[0858] 330. The medium of claim 329, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0859] 331. A computer-readable tangible medium storing computer-executable reverse snap mobile sales instructions to:

[0860] obtain a user checkout request at a point-of-sale device;

[0861] acquire an image frame via an image acquisition device operatively connected to the point-of-sale device;

[0862] identify a payment code depicted within the acquired image frame;

[0863] generate, via the point-of-sale device, a purchase transaction request using the identified payment code;

[0864] provide the purchase transaction request for payment processing; and

[0865] obtain a notification of authorization of the purchase transaction.

[0866] 332. The medium of claim 331, further storing instructions to:

[0867] provide an image of the payment code for purchase transaction processing.

[0868] 333. The medium of claim 331, further storing instructions to:

[0869] acquire video including the image frame via the image acquisition device operatively connected to the point-of-sale device;

[0870] extract the image frame from the acquired video; and

[0871] analyze the image frame to determine whether the image frame includes the depicted payment code.

[0872] 334. The medium of claim 331, wherein the payment code is a one-dimensional barcode.

[0873] 335. The medium of claim 331, wherein the payment code is a two-dimensional barcode.

[0874] 336. The medium of claim 331, wherein the payment code is a Quick Response code.

[0875] 337. The medium of claim 331, further storing instructions to:

[0876] extract purchase payment information from the payment code; and

[0877] wherein the purchase transaction request is generated, via the point-of-sale device, using the extracted purchase payment information.

[0878] 338. The medium of claim 337, wherein the purchase payment information includes an expiration time.

[0879] 339. The medium of claim 337, wherein the purchase payment information is associated with a virtual wallet account, and wherein the generated purchase transaction request includes the purchase payment data associated with the virtual wallet account.

[0880] 340. The medium of claim 331, further storing instructions to:

[0881] provide a portion of the acquired image frame including the depiction of the payment code to a server; and

[0882] obtain purchase payment information from the server in response to providing the portion of the acquired image frame.

[0883] 341. The medium of claim 340, wherein the purchase payment information includes a session identifier for a user shopping session with a merchant.

[0884] 342. The medium of claim 341, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0885] 343. A visual snap computer-implemented means, comprising means for:

[0886] obtaining a user visual item information request (VITR);

[0887] extracting a VITR image frame from the VITR obtained via an image acquisition device operatively connected to a user device;

[0888] extracting VITR field attributes from the VITR image frame through a VITR processing component, wherein the VITR processing component may include barcode recognition and optical character recognition;

[0889] querying a database with the extracted VITR field attributes;

[0890] determining a VITR type from results from the querying;

[0891] providing a visual item information response based on the determined VITR type.

[0892] 344. The means of claim 343, wherein the VITR may be one of a visual coupon addition request, a visual account addition request, a visual bill addition request, a visual purchase request, and a visual purchase information request.

[0893] 345. The means of claim 344, wherein the VITR and visual item information response are storable for later access and retrieval via a VITR history.

[0894] 346. The means of claim 343, wherein the VITR is a visual purchase request, and the provided visual item information response is a purchase intake mechanism, and further comprising means for:

[0895] obtaining the user's indication to purchase a purchase item identified in the provided visual item information response;

[0896] obtaining a purchase transaction request for payment processing of the purchase item; and

[0897] providing a purchase receipt for the purchase transaction.

[0898] 347. The means of claim 346, wherein the VITR, visual item information response, user's indication, purchase transaction request and purchase receipt are storable for later access and retrieval via a VITR history.

[0899] 348. The means of claim 343, wherein a server performs VITR activities.

[0900] 349. The means of claim 343, wherein a user client device performs VITR activities.

[0901] 350. A snap payment computer-implemented means, comprising means for:

[0902] obtaining a user visual item information request (VITR);

[0903] acquiring an VITR image frame via an image acquisition device operatively connected to the user device;

[0904] extracting VITR field attributes from the VITR image frame through a VITR processing component, wherein the VITR processing component may include barcode recognition and optical character recognition;

[0905] querying a database with the extracted VITR field attributes;

[0906] determining a VITR type from results from the querying;

[0907] providing a visual item information response based on the determined VITR type.

[0908] 351. The means of claim 350, wherein the VITR may be one of a visual coupon addition request, a visual account addition request, a visual bill addition request, a visual purchase request, and a visual purchase information request.

[0909] 352. The means of claim 351, wherein the VITR and visual item information response are storable for later access and retrieval via a VITR history.

[0910] 353. The means of claim 350, wherein the VITR is a visual purchase request, and the provided visual item information response is a purchase intake mechanism, and further comprising means for:

[0911] obtaining the user's indication to purchase a purchase item identified in the provided visual item information response;

[0912] providing a purchase transaction request for payment processing of the purchase item; and

[0913] obtaining a purchase receipt for the purchase transaction.

[0914] 354. The means of claim 353, wherein the VITR, visual item information response, user's indication, purchase transaction request and purchase receipt are storable for later access and retrieval via a VITR history.

[0915] 355. A snap payment computer-implemented means, comprising means for:

[0916] obtaining, at a user device, a user input to initiate a purchase transaction;

[0917] acquiring an image frame via an image acquisition device operatively connected to the user device;

[0918] identifying a payment code depicted within the acquired image frame;

[0919] generating, via the user device, a purchase transaction request using the identified payment code;

[0920] providing the purchase transaction request for payment processing; and

[0921] obtaining a purchase receipt for the purchase transaction.

[0922] 356. The means of claim 355, further comprising means for:

[0923] providing an image of the payment code for purchase transaction processing.

[0924] 357. The means of claim 355, further comprising means for:

[0925] acquiring video including the image frame via the image acquisition device included in the user mobile device;

[0926] extracting the image frame from the acquired video; and

[0927] analyzing the image frame to determine whether the image frame includes the depicted payment code.

[0928] 358. The means of claim 355, wherein the user device is a mobile device.

[0929] 359. The means of claim 355, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

[0930] 360. The means of claim 355, wherein the payment code is a one-dimensional barcode.

[0931] 361. The means of claim 355, wherein the payment code is a two-dimensional barcode.

[0932] 362. The means of claim 361, wherein the payment code is a Quick Response code.

[0933] 363. The means of claim 355, further comprising means for:

[0934] extracting purchase session data from the payment code; and

[0935] wherein the purchase transaction request is generated, via the user mobile device, using the extracted purchase session data.

[0936] 364. The means of claim 363, wherein the purchase session data varies based on user shopping activity with a merchant.

[0937] 365. The means of claim 364, wherein the merchant is an online merchant.

[0938] 366. The means of claim 355, further comprising means for:

[0939] providing a portion of the acquired image frame including the depiction of the payment code to a server; and

[0940] obtaining purchase session data from the server in response to providing the portion of the acquired image frame.

[0941] 367. The means of claim 363, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0942] 368. The means of claim 366, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0943] 369. The means of claim 367, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0944] 370. The means of claim 368, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0945] 371. The means of claim 355, further comprising means for:

[0946] obtaining, for payment processing, payment information associated with a virtual wallet account;

[0947] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0948] 372. The means of claim 371, wherein the payment information includes a dynamically generated card verification value code.

[0949] 373. The means of claim 372, further comprising means for:

[0950] providing a request for the dynamically generated card verification value code to a server; and

[0951] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[0952] 374. The means of claim 373, wherein the dynamically generated card verification value has an expiration time.

[0953] 375. The means of claim 373, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0954] 376. The means of claim 355, wherein the payment code depicted within the acquired image frame is acquired from the display of a media device, and encodes data to purchase on-demand media content.

[0955] 377. The means of claim 376, wherein the media device is a television.

[0956] 378. The means of claim 377, wherein the television is part of an in-flight entertainment means.

[0957] 379. The means of claim 376, wherein the media device is displaying a webpage.

[0958] 380. A reverse snap payment computer-implemented means, comprising means for:

[0959] obtaining, at a user device, a user input to initiate a purchase transaction with a merchant;

[0960] obtaining user payment information for processing the purchase transaction;

[0961] generating, via the user device, a payment code image using the payment information for processing the purchase transaction;

[0962] displaying the payment code image, via a display operatively connected to the user device, for a point-of-sale terminal to acquire an image of the payment code image; and

[0963] obtaining a purchase receipt for the purchase transaction.

[0964] 381. The means of claim 380, further comprising means for:

[0965] obtaining a notification that the point-of-sale terminal has acquired an image of the payment code image; and

[0966] terminating display of the payment code image via the display operatively connected to the user device;

[0967] 382. The means of claim 380, wherein the user device is a mobile device.

[0968] 383. The means of claim 380, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

[0969] 384. The means of claim 380, wherein the payment code is a one-dimensional barcode.

[0970] 385. The means of claim 380, wherein the payment code is a two-dimensional barcode.

[0971] 386. The means of claim 385, wherein the payment code is a Quick Response code.

[0972] 387. The means of claim 380, wherein the merchant is an online merchant.

[0973] 388. The means of claim 380, wherein the purchase receipt includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[0974] 389. The means of claim 388, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[0975] 390. The means of claim 380, wherein the user payment information is associated with a virtual wallet account;

[0976] 391. The means of claim 390, wherein the payment information includes a dynamically generated card verification value code.

[0977] 392. The means of claim 391, further comprising means for:

[0978] providing a request for the dynamically generated card verification value code to a server; and

[0979] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[0980] 393. The means of claim 391, wherein the dynamically generated card verification value has an expiration time.

[0981] 394. The means of claim 391, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[0982] 395. The means of claim 380, wherein the point-of-sale terminal is a user device.

[0983] 396. The means of claim 380, wherein the point-of-sale terminal is located at physical merchant store.

[0984] 397. A group split snap payment computer-implemented means, comprising means for:

[0985] obtaining, at a user device of a user, a user input to initiate a group purchase transaction;

[0986] obtaining purchase data for the group purchase transaction;

[0987] generating, via the user device, a split-payment code image using the purchase data for the group purchase transaction;

[0988] wherein the split-payment code image includes information on a payment amount for another user; and

[0989] displaying the split-payment code image, via a display operatively connected to the user device, for another user device of the another user to acquire an image of the split-payment code image.

[0990] 398. The means of claim 397, further comprising means for:

[0991] generating, via the user device, a purchase transaction request using a payment amount for the user covering a portion of the group purchase transaction;

[0992] providing the purchase transaction request for payment processing; and

[0993] obtaining a purchase receipt for the payment amount for the user for the group purchase transaction.

[0994] 399. The means of claim 398, further comprising means for:

[0995] obtaining, for payment processing, payment information associated with a virtual wallet account;

[0996] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[0997] 400. The means of claim 399, wherein the payment information includes a dynamically generated card verification value code.

[0998] 401. The means of claim 400, further comprising means for:

[0999] providing a request for the dynamically generated card verification value code to a server; and

[1000] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[1001] 402. The means of claim 400, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[1002] 403. The means of claim 397, wherein the split-payment code is a Quick Response code.

[1003] 404. A group split payment computer-implemented means, comprising

[1004] obtaining, at a user device of a user, a user input to initiate a group purchase transaction;

[1005] acquiring an image frame via an image acquisition device operatively connected to the user device;

[1006] identifying a payment code depicted within the acquired image frame, the payment code displayed by another user device of another user;

[1007] generating, via the user device, a purchase transaction request for payment covering a portion of the group purchase transaction using the identified payment code;

[1008] providing the purchase transaction request for payment processing; and

[1009] obtaining a purchase receipt for the purchase transaction.

[1010] 405. The means of claim 404, further comprising means for:

[1011] extracting purchase session data from the payment code; and

[1012] wherein the purchase transaction request is generated, via the user mobile device, using the extracted purchase session data.

[1013] 406. The means of claim 405, wherein the purchase session data varies based on user shopping activity with a merchant.

[1014] 407. The means of claim 406, wherein the merchant is an online merchant.

[1015] 408. The means of claim 404, further comprising means for:

[1016] providing a portion of the acquired image frame including the depiction of the payment code to a server; and

[1017] obtaining purchase session data from the server in response to providing the portion of the acquired image frame.

[1018] 409. The means of claim 405, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[1019] 410. The means of claim 408, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

[1020] 411. The means of claim 409, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[1021] 412. The means of claim 410, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[1022] 413. The means of claim 404, further comprising means for:

[1023] obtaining, for payment processing, payment information associated with a virtual wallet account;

[1024] wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

[1025] 414. The means of claim 413, wherein the payment information includes a dynamically generated card verification value code.

[1026] 415. The means of claim 414, further comprising means for:

[1027] providing a request for the dynamically generated card verification value code to a server; and

[1028] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[1029] 416. The means of claim 415, wherein the dynamically generated card verification value has an expiration time.

[1030] 417. The means of claim 415, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

[1031] 418. A person-to-person snap payment computer-implemented means, comprising means for:

[1032] obtaining, at a user device of a user, a user input to initiate a person-to-person transaction;

[1033] obtaining a transfer amount for the person-to-person transaction;

[1034] generating, via the user device, a payment code image using the transfer amount for the person-to-person transaction;

[1035] wherein the payment code image includes information on a transfer amount for another user; and

[1036] displaying the payment code image, via a display operatively connected to the user device, for another user device of the another user to acquire an image of the payment code image.

[1037] 419. The means of claim 418, further comprising means for:

[1038] obtaining, for payment processing, payment information associated with a virtual wallet account;

[1039] wherein the generated payment code image encodes information on the payment information associated with the virtual wallet account.

[1040] 420. The means of claim 419, wherein the payment information includes a dynamically generated card verification value code.

[1041] 421. The means of claim 420, further comprising means for:

[1042] providing a request for the dynamically generated card verification value code to a server; and

[1043] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[1044] 422. The means of claim 418, wherein the transfer amount is obtained from the another user device of the another user.

[1045] 423. The means of claim 418, wherein the split-payment code is a Quick Response code.

[1046] 424. A person-to-person payment computer-implemented means, comprising

[1047] obtaining, at a user device of a user, a user input to initiate a person-to-person transaction;

[1048] acquiring an image frame via an image acquisition device operatively connected to the user device;

[1049] identifying a payment code depicted within the acquired image frame, the payment code displayed by another user device of another user;

[1050] generating, via the user device and using the identified payment code, a payment transfer request for payment to the another user;

[1051] providing the payment transfer request for payment processing; and

[1052] obtaining a transfer confirmation for the person-to-person transaction.

[1053] 425. The means of claim 424, further comprising means for:

[1054] extracting transfer account data from the payment code; and

[1055] wherein the payment transfer request is generated, via the user mobile device, using the extracted transfer account data.

[1056] 426. The means of claim 425, wherein the transfer account data includes data on a virtual wallet account.

[1057] 427. The means of claim 424, further comprising means for:

[1058] providing a portion of the acquired image frame including the depiction of the payment code to a server.

[1059] 428. The means of claim 424, further comprising means for:

[1060] obtaining, for payment processing, payment information associated with a virtual wallet account;

[1061] wherein the generated payment transfer request includes the payment information associated with the virtual wallet account.

[1062] 429. The means of claim 428, wherein the payment information includes a dynamically generated card verification value code.

[1063] 430. The means of claim 429, further comprising means for:

[1064] providing a request for the dynamically generated card verification value code to a server; and

[1065] obtaining the dynamically generated card verification value code from the server in response to providing the request.

[1066] 431. The means of claim 429, wherein the dynamically generated card verification value has an expiration time.

[1067] 432. The means of claim 429, wherein the dynamically generated card verification value is specific to a user funds transfer session between the user device and the another user device.

[1068] 433. A snap mobile sales computer-implemented means, comprising means for:

[1069] obtaining a user checkout request at a point-of-sale device;

[1070] obtaining user shopping cart information with a merchant for processing a purchase transaction related to the user checkout request;

[1071] generating, via the user device, a payment code image using the user shopping cart information;

[1072] displaying the payment code image, via a display operatively connected to the point-of-sale device, for a user device to acquire an image of the payment code image; and

[1073] obtaining a notification of authorization of the purchase transaction.

[1074] 434. The means of claim 433, further comprising means for:

[1075] obtaining a notification that the user device has acquired an image of the payment code image; and

[1076] terminating display of the payment code image via the display operatively connected to the point-of-sale device;

[1077] 435. The means of claim 433, wherein the user checkout request is obtained via a touchscreen gesture on a touchscreen operatively connected to the point-of-sale device.

[1078] 436. The means of claim 433, wherein the user checkout request is obtained via a communication from the user device.

[1079] 437. The means of claim 433, wherein the payment code is a one-dimensional barcode.

[1080] 438. The means of claim 433, wherein the payment code is a two-dimensional barcode.

[1081] 439. The means of claim 438, wherein the payment code is a Quick Response code.

[1082] 440. The means of claim 433, wherein the merchant is an online merchant.

[1083] 441. The means of claim 440, wherein the point-of sale device is another user device.

[1084] 442. The means of claim 433, wherein the point-of-sale terminal is located at physical merchant store.

[1085] 443. The means of claim 433, wherein the notification of authorization of the purchase transaction includes a session identifier for a user shopping session with the merchant.

[1086] 444. The means of claim 443, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[1087] 445. A reverse snap mobile sales computer-implemented means, comprising means for:

[1088] obtaining a user checkout request at a point-of-sale device;

[1089] acquiring an image frame via an image acquisition device operatively connected to the point-of-sale device;

[1090] identifying a payment code depicted within the acquired image frame;

[1091] generating, via the point-of-sale device, a purchase transaction request using the identified payment code;

[1092] providing the purchase transaction request for payment processing; and

[1093] obtaining a notification of authorization of the purchase transaction.

[1094] 446. The means of claim 445, further comprising means for:

[1095] providing an image of the payment code for purchase transaction processing.

[1096] 447. The means of claim 445, further comprising means for:

[1097] acquiring video including the image frame via the image acquisition device operatively connected to the point-of-sale device;

[1098] extracting the image frame from the acquired video; and

[1099] analyzing the image frame to determine whether the image frame includes the depicted payment code.

[1100] 448. The means of claim 445, wherein the payment code is a one-dimensional barcode.

[1101] 449. The means of claim 445, wherein the payment code is a two-dimensional barcode.

[1102] 450. The means of claim 445, wherein the payment code is a Quick Response code.

[1103] 451. The means of claim 445, further comprising means for:

[1104] extracting purchase payment information from the payment code; and

[1105] wherein the purchase transaction request is generated, via the point-of-sale device, using the extracted purchase payment information.

[1106] 452. The means of claim 451, wherein the purchase payment information includes an expiration time.

[1107] 453. The means of claim 451, wherein the purchase payment information is associated with a virtual wallet account, and wherein the generated purchase transaction request includes the purchase payment data associated with the virtual wallet account.

[1108] 454. The means of claim 445, further comprising means for:

[1109] providing a portion of the acquired image frame including the depiction of the payment code to a server; and

[1110] obtaining purchase payment information from the server in response to providing the portion of the acquired image frame.

[1111] 455. The means of claim 454, wherein the purchase payment information includes a session identifier for a user shopping session with a merchant.

[1112] 456. The means of claim 455, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

[1113] In order to address various issues and advance the art, the entirety of this application for SNAP MOBILE PAYMENT APPARATUSES, METHODS AND SYSTEMS (including the Cover Page, Title, Headings, Field, Background, Summary, Brief Description of the Drawings, Detailed Description, Claims, Abstract, Figures, Appendices and/or otherwise) shows by way of illustration various embodiments in which the claimed innovations may be practiced. The advantages and features of the application are of a representative sample of embodiments only, and are not exhaustive and/or exclusive. They are presented only to assist in understanding and teach the claimed principles. It should be understood that they are not representative of all claimed innovations. As such, certain aspects of the disclosure have not been discussed herein. That alternate embodiments may not have been presented for a specific portion of the innovations or that further undescribed alternate embodiments may be available for a portion is not to be considered a disclaimer of those alternate embodiments. It will be appreciated that many of those undescribed embodiments incorporate the same principles of the innovations and others are equivalent. Thus, it is to be understood that other embodiments may be utilized and functional, logical, operational, organizational, structural and/or topological modifications may be made without departing from the scope and/or spirit of the disclosure. As such, all examples and/or embodiments are deemed to be non-limiting throughout this disclosure. Also, no inference should be drawn regarding those embodiments discussed herein relative to those not discussed herein other than it is as such for purposes of reducing space and repetition. For instance, it is to be understood that the logical and/or topological structure of any combination of any program components (a component collection), other components and/or any present feature sets as described in the figures and/or throughout are not limited to a fixed operating order and/or arrangement, but rather, any disclosed order is exemplary and all equivalents, regardless of order, are contemplated by the disclosure. Furthermore, it is to be understood that such features are not limited to serial execution, but rather, any number of threads, processes, services, servers, and/or the like that may execute asynchronously, concurrently, in parallel, simultaneously, synchronously, and/or the like are contemplated by the disclosure. As such, some of these features may be mutually contradictory, in that they cannot be simultaneously present in a single embodiment. Similarly, some features are applicable to one aspect of the innovations, and inapplicable to others. In addition, the disclosure includes other innovations not presently claimed. Applicant reserves all rights in those presently unclaimed innovations, including the right to claim such innovations, file additional applications, continuations, continuations in part, divisions, and/or the like thereof. As such, it should be understood that advantages, embodiments, examples, functional, features, logical, operational, organizational, structural, topological, and/or other aspects of the disclosure are not to be considered limitations on the disclosure as defined by the claims or limitations on equivalents to the claims. It is to be understood that, depending on the particular needs and/or characteristics of a SNAP individual and/or enterprise user, database configuration and/or relational model, data type, data transmission and/or network framework, syntax structure, and/or the like, various embodiments of the SNAP may be implemented that enable a great deal of flexibility and customization. For example, aspects of the SNAP may be adapted for restaurant dining, online shopping, shopping in brick-and-mortar stores, secure information processing, healthcare information systems, and/or the like. While various embodiments and discussions of the SNAP have been directed to electronic purchase transactions, however, it is to be understood that the embodiments described herein may be readily configured and/or customized for a wide variety of other applications and/or implementations.

What is claimed is:

1. A snap payment computer-implemented method, comprising:

obtaining, at a user device, a user input to initiate a purchase transaction;

acquiring an image frame via an image acquisition device operatively connected to the user device;

identifying a payment code depicted within the acquired image frame;

generating, via the user device, a purchase transaction request using the identified payment code;

providing the purchase transaction request for payment processing; and

obtaining a purchase receipt for the purchase transaction.

2. The method of claim **1**, further comprising:

providing an image of the payment code for purchase transaction processing.

3. The method of claim **1**, further comprising:

acquiring video including the image frame via the image acquisition device operatively connected to the user device;

extracting the image frame from the acquired video; and

analyzing the image frame to determine whether the image frame includes the depicted payment code.

4. The method of claim **1**, wherein the user device is a mobile device.

5. The method of claim **1**, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

6. The method of claim **1**, wherein the payment code is a one-dimensional barcode.

7. The method of claim **1**, wherein the payment code is a two-dimensional barcode.

8. The method of claim **7**, wherein the payment code is a Quick Response code.

9. The method of claim **1**, further comprising:

extracting purchase session data from the payment code; and

wherein the purchase transaction request is generated, via the user device, using the extracted purchase session data.

10. The method of claim **9**, wherein the purchase session data varies based on user shopping activity with a merchant.

11. The method of claim **10**, wherein the merchant is an online merchant.

12. The method of claim **1**, further comprising:

providing a portion of the acquired image frame including the depiction of the payment code to a server; and

obtaining purchase session data from the server in response to providing the portion of the acquired image frame.

13. The method of claim **9**, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

14. The method of claim **12**, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

15. The method of claim **13**, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

16. The method of claim **14**, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

17. The method of claim **1**, further comprising:

obtaining, for payment processing, payment information associated with a virtual wallet account;

wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

18. The method of claim **17**, wherein the payment information includes a dynamically generated card verification value code.

19. The method of claim **18**, further comprising:

providing a request for the dynamically generated card verification value code to a server; and

obtaining the dynamically generated card verification value code from the server in response to providing the request.

20. The method of claim **19**, wherein the dynamically generated card verification value has an expiration time.

21. The method of claim **19**, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

22. The method of claim **1**, wherein the payment code depicted within the acquired image frame is acquired from the display of a media device, and encodes data to purchase on-demand media content.

23. The method of claim **22**, wherein the media device is a television.

24. The method of claim **23**, wherein the television is part of an in-flight entertainment system.

25. The method of claim **22**, wherein the media device is displaying a webpage.

26. A snap payment apparatus, comprising:

a processor; and

a memory disposed in communication with the processor and storing processor-executable instructions to:

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

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

identify a payment code depicted within the acquired image frame;

generate, via the user device, a purchase transaction request using the identified payment code;

provide the purchase transaction request for payment processing; and

obtain a purchase receipt for the purchase transaction.

27. The apparatus of claim **26**, the memory further storing instructions to:

provide an image of the payment code for purchase transaction processing.

28. The apparatus of claim **26**, the memory further storing instructions to:

acquire video including the image frame via the image acquisition device operatively connected to the user device;

extract the image frame from the acquired video; and

analyze the image frame to determine whether the image frame includes the depicted payment code.

29. The apparatus of claim **26**, wherein the user device is a mobile device.

30. The apparatus of claim **26**, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

31. The apparatus of claim **26**, wherein the payment code is a one-dimensional barcode.

32. The apparatus of claim **26**, wherein the payment code is a two-dimensional barcode.

33. The apparatus of claim **32**, wherein the payment code is a Quick Response code.

34. The apparatus of claim **26**, the memory further storing instructions to:

extract purchase session data from the payment code; and

wherein the purchase transaction request is generated, via the user device, using the extracted purchase session data.

35. The apparatus of claim **34**, wherein the purchase session data varies based on user shopping activity with a merchant.

36. The apparatus of claim **35**, wherein the merchant is an online merchant.

37. The apparatus of claim **26**, the memory further storing instructions to:

provide a portion of the acquired image frame including the depiction of the payment code to a server; and

obtain purchase session data from the server in response to providing the portion of the acquired image frame.

38. The apparatus of claim **34**, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

39. The apparatus of claim **37**, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

40. The apparatus of claim **38**, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

41. The apparatus of claim **39**, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

42. The apparatus of claim **26**, the memory further storing instructions to:

obtain, for payment processing, payment information associated with a virtual wallet account;

wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

43. The apparatus of claim **42**, wherein the payment information includes a dynamically generated card verification value code.

44. The apparatus of claim **43**, the memory further storing instructions to:

provide a request for the dynamically generated card verification value code to a server; and

obtain the dynamically generated card verification value code from the server in response to providing the request.

45. The apparatus of claim **44**, wherein the dynamically generated card verification value has an expiration time.

46. The apparatus of claim **44**, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

47. The apparatus of claim **26**, wherein the payment code depicted within the acquired image frame is acquired from the display of a media device, and encodes data to purchase on-demand media content.

48. The apparatus of claim **47**, wherein the media device is a television.

49. The apparatus of claim **48**, wherein the television is part of an in-flight entertainment system.

50. The apparatus of claim **47**, wherein the media device is displaying a webpage.

51. A snap payment means, comprising:

means for obtaining, at a user device, a user input to initiate a purchase transaction;

means for acquiring an image frame via an image acquisition device operatively connected to the user device;

means for identifying a payment code depicted within the acquired image frame;

means for generating, via the user device, a purchase transaction request using the identified payment code;

means for providing the purchase transaction request for payment processing; and

means for obtaining a purchase receipt for the purchase transaction.

52. The means of claim **51**, further comprising:

means for providing an image of the payment code for purchase transaction processing.

53. The means of claim **51**, further comprising:

means for acquiring video including the image frame via the image acquisition device operatively connected to the user device;

means for extracting the image frame from the acquired video; and

means for analyzing the image frame to determine whether the image frame includes the depicted payment code.

54. The means of claim **51**, wherein the user device is a mobile device.

55. The means of claim **51**, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

56. The means of claim **51**, wherein the payment code is a one-dimensional barcode.

57. The means of claim **51**, wherein the payment code is a two-dimensional barcode.

58. The means of claim **57**, wherein the payment code is a Quick Response code.

59. The means of claim **51**, further comprising:

means for extracting purchase session data from the payment code; and

means for wherein the purchase transaction request is generated, via the user device, using the extracted purchase session data.

60. The means of claim **59**, wherein the purchase session data varies based on user shopping activity with a merchant.

61. The means of claim **60**, wherein the merchant is an online merchant.

62. The means of claim **61**, further comprising:

means for providing a portion of the acquired image frame including the depiction of the payment code to a server; and

means for obtaining purchase session data from the server in response to providing the portion of the acquired image frame.

63. The means of claim **59**, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

64. The means of claim **62**, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

65. The means of claim **63**, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

66. The means of claim **64**, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

67. The means of claim **51**, further comprising:

means for obtaining, for payment processing, payment information associated with a virtual wallet account;

wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

68. The means of claim **67**, wherein the payment information includes a dynamically generated card verification value code.

69. The means of claim **68**, further comprising:

means for providing a request for the dynamically generated card verification value code to a server; and

means for obtaining the dynamically generated card verification value code from the server in response to providing the request.

70. The means of claim **69**, wherein the dynamically generated card verification value has an expiration time.

71. The means of claim **69**, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

72. The means of claim **51**, wherein the payment code depicted within the acquired image frame is acquired from the display of a media device, and encodes data to purchase on-demand media content.

73. The means of claim **72**, wherein the media device is a television.

74. The means of claim **73**, wherein the television is part of an in-flight entertainment system.

75. The means of claim **72**, wherein the media device is displaying a webpage.

76. A computer-readable tangible medium storing computer-executable snap payment instructions to:

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

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

identify a payment code depicted within the acquired image frame;

generate, via the user device, a purchase transaction request using the identified payment code;

provide the purchase transaction request for payment processing; and

obtain a purchase receipt for the purchase transaction.

77. The medium of claim **76**, further storing instructions to:

provide an image of the payment code for purchase transaction processing.

78. The medium of claim **76**, further storing instructions to:

acquire video including the image frame via the image acquisition device operatively connected to the user device;

extract the image frame from the acquired video; and

analyze the image frame to determine whether the image frame includes the depicted payment code.

79. The medium of claim **76**, wherein the user device is a mobile device.

80. The medium of claim **76**, wherein the user input is a touchscreen gesture on a touchscreen operatively connected to the user device.

81. The medium of claim **76**, wherein the payment code is a one-dimensional barcode.

82. The medium of claim **76**, wherein the payment code is a two-dimensional barcode.

83. The medium of claim **82**, wherein the payment code is a Quick Response code.

84. The medium of claim **76**, further storing instructions to:

extract purchase session data from the payment code; and

wherein the purchase transaction request is generated, via the user device, using the extracted purchase session data.

85. The medium of claim **84**, wherein the purchase session data varies based on user shopping activity with a merchant.

86. The medium of claim **85**, wherein the merchant is an online merchant.

87. The medium of claim **76**, further storing instructions to:

provide a portion of the acquired image frame including the depiction of the payment code to a server; and

obtain purchase session data from the server in response to providing the portion of the acquired image frame.

88. The medium of claim **84**, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

89. The medium of claim **87**, wherein the purchase session data includes a merchant identifier, and a session identifier for a user shopping session with a merchant associated with the merchant identifier.

90. The medium of claim **88**, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

91. The medium of claim **89**, wherein the session identifier is configured to serve as a token parameter in a uniform resource locator for data on the user shopping session with the merchant.

92. The medium of claim **76**, further storing instructions to:

obtain, for payment processing, payment information associated with a virtual wallet account;

wherein the generated purchase transaction request includes the payment information associated with the virtual wallet account.

93. The medium of claim **92**, wherein the payment information includes a dynamically generated card verification value code.

94. The medium of claim **93**, further storing instructions to:

provide a request for the dynamically generated card verification value code to a server; and

obtain the dynamically generated card verification value code from the server in response to providing the request.

95. The medium of claim **94**, wherein the dynamically generated card verification value has an expiration time.

96. The medium of claim **94**, wherein the dynamically generated card verification value is specific to a user shopping session with a merchant.

97. The medium of claim **76**, wherein the payment code depicted within the acquired image frame is acquired from the display of a media device, and encodes data to purchase on-demand media content.

98. The medium of claim **97**, wherein the media device is a television.

99. The medium of claim **98**, wherein the television is part of an in-flight entertainment system.

100. The medium of claim **97**, wherein the media device is displaying a webpage.

* * * * *