



(51) International Patent Classification:
G06Q 30/06 (2012.01)

(21) International Application Number:
PCT/US2019/051537

(22) International Filing Date:
17 September 2019 (17.09.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
16/133,197 17 September 2018 (17.09.2018) US
62/816,495 11 March 2019 (11.03.2019) US

(71) Applicant: **PIGGY LLC** [US/US]; 225 Primera Boulevard, Suite 160, Lake Mary, FL 32746 (US).

(72) Inventors: **CORRIERI, Nicholas**; 1624 Pebble Creek Dr., Sartell, MN 56377 (US). **GINSBERG, John**; 147A Shaftesbury Avenue, Southend-On-Sea SS1 3AL (GB). **MAHALEC, Nicholas**; 201 Ocean Avenue, Unit 905P, Santa Monica, CA 90402 (US).

(74) Agent: **MATOS, Peter, A.** et al.; Malloy & Malloy, P.L., 2800 S.W. Third Avenue, Miami, FL 33129 (US).

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

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM,

(54) Title: SYSTEMS, METHODS, AND COMPUTER PROGRAMS FOR PROVIDING USERS MAXIMUM BENEFIT IN ELECTRONIC COMMERCE

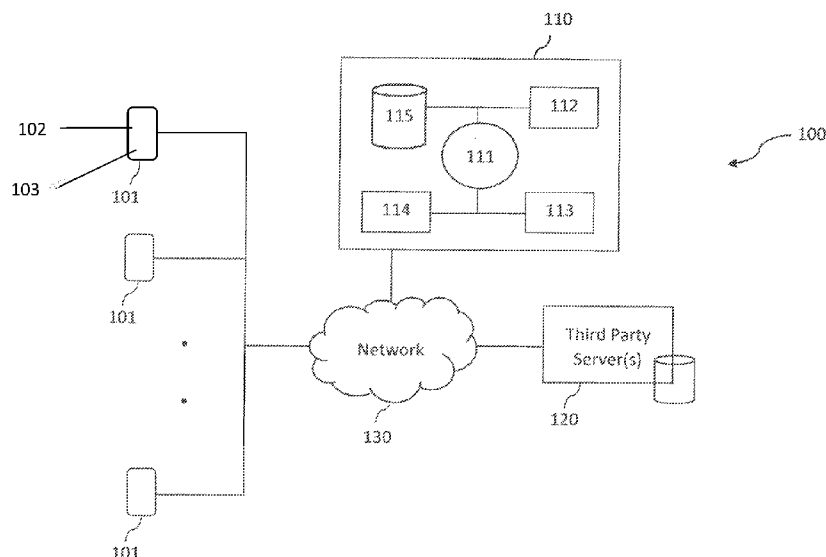


FIG. 1

(57) Abstract: The present invention generally relates to a system, method, and computer program for providing maximum savings in electronic commerce. Specifically, during a shopping experience, several products or service inputs, such as: electronic coupon codes, sales taxes, cashback rebates, customer loyalty programs, etc. are calculated and compared against numerous competitor retailer stores to ascertain the lowest price. Several optimization schemes are also disclosed which enhance a user experience.

TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW,
KM, ML, MR, NE, SN, TD, TG).

Published:

- *with international search report (Art. 21(3))*
- *before the expiration of the time limit for amending the
claims and to be republished in the event of receipt of
amendments (Rule 48.2(h))*

SYSTEMS, METHODS, AND COMPUTER PROGRAMS FOR PROVIDING USERS
MAXIMUM BENEFIT IN ELECTRONIC COMMERCE

BACKGROUND OF THE INVENTION

5 Claim of Priority

The present application claims the domestic benefit, pursuant to 35 U.S.C. § 119(e) to U.S. Provisional Patent Application Serial No. 62/816,495 filed on March 11, 2019, the contents of which are incorporated herein, by reference, in its entirety.

10 The present application is also claims the domestic benefit of a previously filed, now pending application having U.S. Application Serial No. 16/133,197 filed on September 17, 2018, which claims the benefit of U.S. Provisional Patent Application Serial No. 62/559,164, filed on September 15, 2017, and which is a continuation-in-part application of a previously filed, now pending application having U.S. Application Serial No. 15/606,856, which was filed on May 26, 2017, which is a continuation-in-part application of a previously filed, now pending
15 application having U.S. Application Serial No. 15/261,733 filed on September 9, 2016, and which claimed the benefit of U.S. Provisional Serial Application No. 62/216,111 filed on September 9, 2015, the contents of which are incorporated herein, by reference, in their entirety.

20 Field of the Invention

The present invention generally relates to systems, methods, and computer programs for providing the maximum user benefit to a user in electronic commerce. Specifically, multiple price factors are used to compare products and services of various e-commerce merchants in order to automatically alert the user to the best possible price during electronic checkout.
25 Various embodiments of the present invention also provide the user-less tracking of an affiliate purchase in order to apportion user cash back without first requiring the creation of a shopping

experience. Further embodiments are directed to various optimizations of a software embodiment designed to enhance the user experience.

DESCRIPTION OF THE RELATED ART

5 Electronic commerce (“e-commerce”) generally utilizes the internet to conduct retail transactions, or “e-commerce transactions.” As such, it has adopted a number of features that are analogous to brick-and-mortar institutions, such as shopping carts. To this end, most e-commerce websites utilize electronic coupon codes, strings of letters and/or digits which must be input by a consumer, to provide customers with discounts, as opposed to physical coupons.

10 These electronic coupon codes may be issued to the public at large, either via marketing or advertising, or may be issued to individual consumers, or groups of consumers, for various purposes. Either way, the electronic coupon code must be known before it can be utilized during checkout of an e-commerce transaction. Several attempts have heretofore been made to aggregate known electronic coupon codes for the public benefit, usually as listings on websites

15 dedicated to such a purpose. This solution has not been ideal as it requires consumers to first seek out applicable electronic coupon codes, and then manually test each one to determine whether it is still viable.

 Additionally, most e-commerce websites do not provide the user with information regarding competitor’s goods or services. If the user would like to compare product or service

20 information between various merchants (e.g., price) he/she would have to visit each website or open each native application, which takes time. In addition, it would take even more time to calculate the lowest “net” price (i.e., base price +/- taxes, shipping costs, electronic coupon code savings, cashback savings, etc.) of each product or service. As such, there is a need in the art for an automated solution. The present invention answers such a need, and provides several

25 additional benefits to consumers in order to facilitate e-commerce transactions.

SUMMARY OF THE INVENTION

The present invention is generally directed to providing the maximum benefit, such as automatic detection and application of coupons, cash back, and price comparison across multiple vendors, to a user in electronic commerce and accompanying systems, methods, and computer programs thereof. The preferred embodiment of the present invention facilitates the provision of a maximum benefit in electronic commerce (“e-commerce”) settings, such as retail websites and downloadable applications. More specifically, products and/or services of various e-commerce merchants will be automatically compared in order to alert the user to his/her maximum benefit, such as lowest price. In order to compare a product or service of various e-commerce merchants and find the lowest price, multiple inputs may be included, such as but not limited to, availability, price, taxes, fees, electronic coupon codes, rebates or cash back, etc. Accordingly, the user may save money on the purchase of any conceivable e-commerce product or service, including activities, tours, hotel rooms, car rentals, etc. This can be done on exact matches of a product or service, or on a similarity match for the product or service, as described in greater detail below.

As such, at least one embodiment may comprise a browser “plug-in,” or background service, whether in a desktop, mobile, or wearable electronic operating system, which may be implemented for tracking the current website or application the user is viewing and automatically populating at least one electronic coupon or coupon code, as well as providing the user with a lower priced option, as described in greater detail below. For example, in a browser plug-in embodiment, a URL or URI may be tracked, and upon a change or a user’s visit to a recognized store, at least one input, such as a coupon code, may be sent by the browser plug-in to an application server communicating with the plug-in.

For purposes of clarity and without limiting the scope of the present invention, this embodiment may be described as being implemented with a web browser (e.g., mobile, desktop, etc.) for accessing information on the world wide web. More specifically, accessing

retail websites or web pages that are identified by a distinct identifier, such as a URL, enabling the web browser to retrieve and display them on the user's device. However, it is emphasized that this embodiment can be operable with downloadable applications (e.g., mobile apps, desktop apps, etc.) which do not require a URL or access to the world wide web, or on websites
5 through some other means of identifying what product, hotel, flight, service, activity, or other type of listing is being presented to the user on the page.

An algorithm may be implemented in order to detect a website URL or other e-commerce interface identifier or certain product and/or service information, or determine the type of retail store, and accordingly access various data fields, preferably in a uniform format
10 across different stores, in order to determine which input(s) to retrieve from the e-commerce website or application that the user is viewing (this is typically, but not always, initiated when the user is at a "checkout" process – the area of the website or application which allows the user to initiate and/or complete an e-commerce transaction).

DOM parsing rules may be implemented, for example, to access various fields from the
15 website or application via JavaScript code. Individual algorithms may be implemented for different proprietary websites or applications (such as Amazon.com and other large retailers), as well as available out-of-the-box electronic commerce solutions or stores such as Joomla, Magenta, or Wordpress plugins, Shopify, Volusion, Bigcommerce, bigcartel, 3dcart, Squarespace, and the like. Each algorithm specific to a particular store may be established by
20 reverse engineering an existing online store and/or store shopping cart, such as to securely retrieve, submit, and validate responses to the data fields within the online store and/or shopping cart. Certain stores may require the redaction of prices of items, charges for the quantity of shipping, etc., in order to obtain a set of usable data fields for the application server, and as such, the present invention may be configured to comply with same.

25 In at least one embodiment, electronic coupon code(s) may be predetermined to be the "best" code or set of codes in accordance with a rules engine. The rules engine may determine

the best code based on crowd sourced user feedback (i.e. voting), administrator setting at the application server of various coupon priorities, numerical percentage or dollar off, other factors, or combinations thereof. As such, the rules engine is configured to establish a ranking of codes according to at least the aforementioned metrics, with the “best” electronic coupon code, or set of electronic coupon codes, being the highest ranked electronic coupon code or set of electronic coupon codes. A database of electronic coupon codes may be stored at the application server, the database of electronic coupon codes may be manually inserted, user-uploaded, or received from respective APIs and/or third-party application servers of affiliates and partners.

The rules engine may run in part or in whole on the application server and/or on the user or client device as a background process. On an application server embodiment, each time a new electronic coupon code is inserted for a certain store, the application server may run an algorithm to ascertain the numerical discount, percentage discount, and/or combinability of the new code with existing electronic coupon codes, and determine a preset “best” or highest ranked set of codes at the application server. This would then be pulled or pushed from the application server to the client device or browser plug-in running thereon at the time of a customer’s “checkout” or completion of the e-commerce transaction. In the client side embodiment, a plurality of electronic coupon codes may be pulled or pushed from the application server to be tested on the fly on the client device. This embodiment allows customization of the electronic coupon codes to the particular user, as a user may have exhausted certain electronic coupon code already.

In at least one embodiment, if an electronic coupon code is applied a number of times and is unsuccessful in achieving a reduction of the user’s e-commerce transaction amount, the application server may set a flag for human review to ensure that the electronic coupon code is still valid, or, in yet further embodiments, may automatically set the code to be invalid and to therefore be no longer utilized by the rules engine.

The highest ranked electronic coupon codes may be automatically (requiring no user action) or semi-automatically (requiring a single or a few user action(s)) populated into the coupon fields of an online store, in accordance with the rules engine.

Another embodiment of the present invention provides for the user-less, or anonymous,
5 tracking of an affiliate purchase in order to apportion user cash back without first requiring the creation of a user account. The present service receives a percentage of product sales for each product sold through an affiliate network or partnership. This revenue is then divided up or apportioned to the user making the purchase, therefore providing them with a so-called “cash back.” The tracking of this cash back is believed to be particularly novel because the present
10 service and application server are able to track e-commerce transactions, without first requiring the user to create a user account at the application server or service.

In order to support this, an identifier is assigned to the application (application identifier or App ID), such as the browser plug-in, background process, or other desktop or mobile application, upon installation and/or first communication to the application server. A table of
15 Application IDs are kept on the application server, which checks and logs new unique App IDs.

Signed communications are then utilized, i.e. a unique identifier tagged into any links or navigable portions for effecting action at the application server and/or client device. The application server may then track actions, i.e. clicks, and purchases through the plug-in and tie the actions to the App ID.

20 As discussed, this feature allows the application server to track users without first forcing or requiring registration of personal information during either the coupon use or cash back receiving steps. The transaction is preferably kept for a number of days, such as 10, 15, 30, 45, 60 days in order to encourage user registration. Later, when a user uses the installed application having the App ID to create a user account, traditional user registration (or login
25 by 3rd party APIs described below) is used, and the data associated with an App ID is then merged into the user account.

Additional features and embodiments may be combined with the various systems and methods of providing maximum benefit disclosed herein in order to enhance the overall user experience. Initially, various software embodiments of the invention, such as mobile applications, desktop applications, or browser extensions, may be optimized to run various processes as background processes while the user's attention is elsewhere, in order to reduce waiting times for the user. By way of example, as has been disclosed, in at least one embodiment, a method of automatically applying coupon codes and other benefits is as follows:

1. A user lands at cart or checkout page of an e-commerce platform using a computing device;

2. A software embodiment of the invention directs the device to display a modal to the user;

3. The user completes an affirmative interaction with the device, such as a click or tap, to apply coupons, activate cashback, or seek another benefit, such as price comparison;

4. The software embodiment of the invention then searches for and/or applies the desired benefits, such as coupons, which according to current estimates, can take anywhere from 3-15+ seconds depending on the number of coupons that need to be identified and applied;

5. The software embodiment of the invention applies the benefit with the biggest savings to the user, thereby determining the maximum benefit.

However, an optimized embodiment of the foregoing method of automatically applying benefits is capable of reducing wait times by at least 3, but most likely 10 seconds, or more. Such a method may include the steps of:

1. A user lands at cart or checkout page of an e-commerce platform using a computing device;

2. With or without appearing to the user, a software embodiment of the invention automatically starts identifying and applying benefits, such as coupon codes, cash back, and/or price comparison, as a background process while the user may be, e.g., reviewing the contents of the cart;

3. Once the software embodiment identifies which benefits are successful, it further identifies the benefit or combination of benefits providing the most savings, i.e., the maximum benefit;

4. The software embodiment then directs the device to display a notification to the user, which may be in the form of a message such as “We found \$3.07 in savings, click to apply” or “Click to apply coupons”;

5. The user may then complete an affirmative interaction with the device, such as a click or tap, to apply the coupon code or other benefit.

Accordingly, because the software embodiment has already identified which coupon code or other benefit or combination of benefits provides the maximum benefit, it does not need to cycle through all codes after the user completes the affirmative interaction, and instead it simply applies the previously identified benefit(s). Thus the user perceives that the operating time for applying coupon codes is reduced as compared to the first embodiment and indeed, the user’s overall wait time is reduced.

In additional embodiments, the software embodiment can also activate cashback and/or price comparison features as part of the benefits that are determined in a background process if there is an affiliate relationship with the merchant operating the e-commerce platform for cashback, or if there are lower prices available at an alternate vendor.

In yet further embodiments, the software embodiment may not prompt the user at all to apply the identified benefit(s), but may simply check and automatically apply the benefit(s) providing the most savings without any user input whatsoever. Alternatively, the software embodiment can run at least a portion of the software embodiment on an application server

side first (after having identified what is already in the user's cart), if it is not able to run it on a client device for any reason.

Yet another feature that may be deployed with any of the embodiments disclosed herein is the ability for a software embodiment of the present invention to monitor various page elements to determine if a customer is in a cart or checkout page, as opposed to solely relying on monitoring a URL. Such a feature is useful for app-based shopping or in circumstances where a URL may be obscured from the software embodiment of the present invention. Such a page element can include a keyword, image, logo, or other parsable object associated with a checkout interface of an e-commerce interface, and may be operative to trigger the software embodiment of the present invention to begin applying the ruleset to obtain maximum benefit, such as testing and/or application of coupon codes, cashback, or price comparison across multiple vendors.

Yet another feature that may be deployed with virtually any embodiment of the present invention is the incorporation of disparate customer "loyalty" or "perk" programs into the ruleset for analysis of maximum user benefit. By way of non-limiting example, a user of the software embodiment of the present invention may operatively associate a customer shipping program, such as Freeshipping.com, shopranner.com, or even one implemented by the applicant, with the present invention. If it is a program implemented by the applicant, customer access may be identified through a username or a unique app identifier. Third party programs may require one or more solutions, either via participating credit cards that those programs work with (e.g., American Express offers Shoprunner access to its cardholders); customers may be prompted to login to a third-party shipping program directly from a software embodiment of the present invention; or by forwarding the user to the login page for those third-party programs, and then routing the user back to the page they were previously on for the e-commerce merchant.

In any event, such a consideration as free shipping from a particular vendor or other loyalty benefit associated with a merchant or financial institution (such as reward “points” redeemable with the merchant or financial institution) may be considered by the ruleset when considering maximum benefit for a user. By way of non-limiting example, if a user is searching for a particular product on a particular website or mobile application, the software embodiment of the present invention may be operative to search for coupons, cashback, price comparison, and whether the product is available via a participating customer shipping program, in order to assess the least expensive means for the user to obtain the product, whether that means applying a large coupon to a merchant that requires a shipping fee, or possibly ignoring a coupon from one merchant, in favor of free shipping from a participating customer shipping program available from another merchant. Other customer loyalty programs, such as “points” or “perks” can be assessed in a similar manner, and the user may be presented with the option to spend the “points” earned from a particular merchant in order to obtain the maximum benefit on a given purchase.

Yet another feature that may be deployed with virtually any embodiment of the present invention is presentation of customization options to users. Given that the present invention presents a variety of embodiments with various features and services, further embodiments provide users with options to select the particular embodiments, benefits, features, services, etc. that are desirable to the user. Additionally, a user may be able to select which e-commerce platforms to implement the present invention. For example, in the embodiment of automatic coupon application, a user may choose to allow the software embodiment to run only on the top stores in the US (top 10, top 50, etc.), or choose to have the software embodiment run on all stores globally. For price comparison embodiments (both the product and travel price compare products), the user may choose which e-commerce platforms to compare. The user may also select which benefits the ruleset will consider, including, but not limited to, coupons, cashback, price comparison, and loyalty programs. Finally, the user may also select whether

the present invention will automatically search for and apply the best benefits without user intervention, or whether an affirmative action will be required from the user before the invention begins to search for and apply the best benefits.

Yet further embodiments allow the user to choose these preferences in the onboarding
5 process and manage these selections later in a preferences section of their account, which may be presented in a mobile application, browser extension or, on a dedicated website.

These and other objects, features and advantages of the present invention will become clearer when the drawings as well as the detailed description are taken into consideration.

10 BRIEF DESCRIPTION OF THE DRAWINGS

For a fuller understanding of the nature of the present invention, reference should be had to the following detailed description taken in connection with the accompanying drawings in which:

Figure 1 is a diagrammatic representation of an exemplary application and
15 implementation environment for providing electronic coupon codes and cash back in electronic commerce of the present invention.

Figure 2 is an exemplary user interface of a browser application for providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end user functionality.

20 Figure 3 is an exemplary user interface of a browser application for providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end user functionality with a browser extension.

Figure 4 is an exemplary user interface of a browser application for providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end
25 user functionality with a browser extension.

Figure 5 is an exemplary user interface of a browser application for providing a providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end user functionality with a browser extension.

Figure 6 is an exemplary user interface of a browser application for providing a
5 providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end user functionality with a browser extension.

Figure 7 is another exemplary user interface of a browser application for providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end user functionality.

10 Figure 8 is another exemplary user interface of a browser application for providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end user functionality with a browser extension.

Figure 9 is another exemplary user interface of a browser application for providing electronic coupon codes and cash back in electronic commerce of the present invention,
15 illustrating end user functionality with a browser extension.

Figure 10 is another exemplary user interface of a browser application for providing a providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end user functionality with a browser extension.

Figure 11 is another exemplary user interface of a browser application for providing a
20 providing electronic coupon codes and cash back in electronic commerce of the present invention, illustrating end user functionality with a browser extension.

Figure 12 is a flow chart depicting steps of a method according to one embodiment of present invention.

Figure 13 is another exemplary user interface of an application for providing maximum
25 benefit in electronic commerce of the present invention, illustrating end user functionality.

Figure 14 is another exemplary user interface of an application for providing maximum benefit in electronic commerce of the present invention, illustrating end user functionality.

Figure 15 is another exemplary user interface of an application for providing maximum benefit in electronic commerce of the present invention, illustrating end user functionality.

5 Figure 16 is another exemplary user interface of an application for providing maximum benefit in electronic commerce of the present invention, illustrating end user functionality.

Figure 17 is a flowchart depicting operative logic of a method according to one software embodiment of the present invention.

Like reference numerals refer to like parts throughout the several views of the drawings.

10

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

The present invention may be implementable on an application server in a software-as-a-service (SaaS) mobile application, or desktop application environment in accordance with implementation on a system or application 100 as generally represented in Figure 1.

15 Accordingly, the system 100 of the present invention generally comprises at least one client device 101 communicably connected to at least one application server 110 over a network 130. One or more third party server(s) 120 may further be communicably connected to the application server 110 and the at least one client device 101 over the same network 130.

20 The client device 101 may comprise a mobile device, tablet, computer, wearable electronic device, or any other device or combination of circuits structured and configured to communicate with another device, computer, or server over the network 130. The client device 101 may comprise application(s) and user interface(s) (front-end interface) that allows a user to interact with the application server(s) 110 and any third-party server(s) 120 and stored applications and programs thereon (back-end processing). The user interface may be
25 proprietary and may comprise a custom developed mobile or desktop application(s). Alternatively, or in addition to, the user interface may comprise a web browser (including

mobile browsers designed for use on a mobile device) or other application or executable code that allows for communication and visualization of information.

The term “application server” 110, “third party server” 120 refer to at least one computer having appropriate hardware and applications installed thereon for the provision of server services including web and other functional services described herein, such that a user may access, execute, and/or view the applications remotely from a client device 101. More specifically, the application server(s) 110 and third-party server(s) 120 may comprise general-purpose computers, specialized computers, or other hardware components structured and configured to receive, process, transmit, and store information to and from other devices. The application server 110 is further configured with executable or interpretable computer code that allows it to perform the processes described within this application.

For example, the application server 110 may comprise a general-purpose computer comprising a central processing unit (CPU) 111, which may be a single core or multi core processor, memory 114 (random-access memory, read-only memory, and/or flash memory) or primary memory for high-speed storage of executing programs, electronic storage unit 115 (e.g., hard disk) or secondary memory for storing data, communications interface 112 (e.g., network adapter) for communicating with other devices or computers over a network, and/or peripheral device(s) 113 in communication with the CPU 111 that enable input/output of the application server 110.

The application server 110 may implement the methodology of the present invention using any number of solution stacks (a set of software subsystems or components) known to an ordinary computer or web programmer skilled in the art. These solution stacks may include, without limitation, ZEND Server, APACHE Server, NODE.JS, ASP, PHP, RUBY, XAMPP, LAMP, WAMP, MAMP, WISA, LEAP, GLASS, LYME, LYCE, OpenStack, Ganeti, MEAN, MEEN, XRX, JAVASCRIPT and other past, present, or future equivalent solution stacks, or combinations thereof, known to those skilled in the art that allows a programmer to develop

the methods and computer programs described within this application. The software stack might be implemented without third-party cloud platforms, for example using load balancing and virtualization software provided by Citrix, Microsoft, VMware, Map-Reduce, Google Filesystem, Xen, memory caching software such as Memcached and Membase, structured
5 storage software such as MySQL, MariaDB, XtraDB, etc. and/or other appropriate platforms. Of course, these solution stacks may also be deployed in cloud platforms by using known development tools and server hosting services such as GitHub and Rackspace, as well as their equivalents.

The third-party server(s) 120 may comprise any combination of hardware and software
10 (code segments in any number of programmable, executable, or interpretable languages that support the functionality of the methods described herein) configured to host and transmit items of a user. The third-party server(s) 120 may be configured to communicate directly to the application server 110 via application programming interfaces or upon the request of a user.

User account services may be implemented using one or more solution stacks as
15 described above. Alternatively, third-party login services such as Facebook, Twitter, LinkedIn, Google and other related services, may be utilized for user account login and authentication, such as via existing third-party server(s) 120 of other parties.

The network 130 may comprise at least two computers in communication with each other, which may form a data network such as via LAN, WAN, Serial, Z-WAVE, ZIGBEE,
20 RS-485, MODBUS, BACNET, the Internet, or combinations thereof. The connections may be facilitated over various wired and/or wireless mediums or any combination thereof including interconnections by routers and/or gateways. Network 130 may comprise additional hardware components and/or devices appropriate for facilitating the transmission and communication between the various systems and devices of the present invention, such as those directed to
25 integrated authentication, quality control or to improve content delivery such as via a content delivery network (CDN).

Various aspects of the present invention may be thought of as “products” or “articles of manufacture” typically in the form of machine (or processor) executable code, interpretable code, and/or associated data that is carried on or embodied in a machine readable medium. Machine-executable code can be stored on an electronic storage unit, such memory (e.g., read-
5 only memory, random-access memory, flash memory) or a hard disk, as described above.

All or portions of the software may at times be communicated through the Internet or other communication networks. Such communications, for example, may enable loading of the software from one computer or processor onto another, for example, from a management server or host computer onto the computer platform of an application server, or from an application
10 server onto a client computer or device. Thus, another type of media that may bear the software elements includes optical, electrical and electromagnetic waves, such as used across physical interfaces between local devices, through wired and optical landline networks and over various air-links. The physical elements that carry such waves, such as wired or wireless links, optical links or the like, also may be considered as media bearing the software. As used herein,
15 tangible “storage” media, terms such as computer or machine “readable medium”, refer to any medium that participates in providing instructions to a processor for execution. Further, the term “non-transitory” computer readable media includes both volatile and non-volatile media, including RAM. In other words, non-transitory computer media excludes only *transitory propagating signals per se*, but includes at least register memory, processor cache, RAM, and
20 equivalents thereof.

Therefore, a machine-readable medium, such as computer-executable code and/or related data structures, may take many forms, including but not limited to, a tangible storage medium, a carrier wave medium or physical transmission medium. Non-volatile storage media include, for example, optical, magnetic, or solid-state disks, such as any of the storage devices
25 in any computer(s) or the like, such as may be used to house the databases. Volatile storage media include dynamic memory, such as main memory of such a computer platform. Tangible

transmission media may include coaxial cables, copper wire and fiber optics, communication buses. Carrier-wave transmission media may take the form of electric or electromagnetic signals, or acoustic or light waves such as those generated during radio frequency (RF) and infrared (IR) data communications. Common forms of computer-readable media therefore
5 include for example: a floppy disk, a flexible disk, hard disk, magnetic tape, any other magnetic medium, a CD-ROM, DVD or DVD-ROM, any other optical medium, punch cards paper tape, any other physical storage medium with patterns of holes, a RAM, a ROM, a PROM and EPROM, a FLASH-EPROM, any other memory chip or cartridge, a carrier wave transporting data or instructions, cables or links transporting such a carrier wave, or any other medium from
10 which a computer may read programming code and/or data. Many of these forms of computer readable media may be involved in carrying one or more sequences of one or more instructions to a processor for execution.

A web browser or downloadable application may embed and/or access a plug-in or extension 103. As known in the art, an extension may be implemented in different forms,
15 including a callable script implemented using a scripting language, such as JavaScript. For example, a web browser 102 may call or invoke a script built and injected into a webpage and loaded by a browser-type application. As discussed hereinafter, an extension 103 may communicate with the application server 110 so as to send and retrieve e-commerce information relating to products and services of merchant e-retailers.

20 A client device 101 may include device specific storage functionality such as local storage capable of storing certain e-commerce information. For example, device local storage may be configured to store a plurality of electronic coupon codes or response data to codes checked against a website or application's "checkout page". However, local storage is not a necessary requirement of the present invention; e-commerce information, such as a plurality of
25 coupon codes, may be stored and applied directly from the application server 110.

In one embodiment, device local storage may submit response data to the application server 110, so as to store and maintain successful electronic coupon codes. Response submission may be initiated by the browser extension 103 and may be carried out as a background process. Unsuccessful electronic coupon code responses may flag an electronic coupon code to be deleted or human reviewed.

For the ease of illustration, the present disclosure primarily features a web browser application of a client device 101 used to conduct e-commerce website transactions, and a browser extension 103 used by the client device 101 to render client side functionalities of the disclosed method. However, additional embodiments and implementations are envisioned by the present invention.

The present invention provides for a system, method, and computer program for providing, automatically trying, and applying electronic coupon codes and cash back in electronic commerce. Drawing attention to Figures 2 – 6, embodiments of the present invention are illustrated from an end user's perspective.

Drawing attention to Figure 2, the present invention may comprise a browser plug-in or extension that may appear dormant until being 'activated' and injected into the user interface upon navigation to a "checkout page" of a supported e-commerce website, as discussed in detail hereinafter. In at least one embodiment, the user may be provided with an icon plug-in 201, for example, displayed on the browser's toolbar. The icon plug-in 201, may comprise specific e-commerce information regarding the currently navigated website via back-end communications with the application server 110, as discussed in greater detail hereinafter. The icon plug-in 201 may comprise a coupon code icon 202, representing a total amount of electronic coupon codes from the application server 110 for the current website. In another embodiment, the icon plug-in 201 may comprise a cash back icon 203, for example, indicated by dollar signs. The cash back icon 203, may indicate to the user that the current website offers some form of 'cash back' on purchases made via the current e-commerce website.

Upon initially navigating to a website, the application 100 may detect and capture URL information, such as the domain name, to send as a parameter to the application server 110 in order to validate that the website is supported, as known to those skilled in the art. If supported, the application 100 may retrieve specific e-commerce information and store it locally, such as, to the device's temporary local storage until the user navigates to the current website's "checkout page". Upon navigation to a different website, the application 100 will repeat the above process.

For example, the coupon code icon 202, may display the number '5' indicating that there are five merchant electronic coupon codes provided for the user by the application server 110. In one embodiment, the icon plug-in 201 may appear colorless and without the coupon code or cash back icons 202, 203. This may happen for example, upon navigation to an unsupported website or if the application 100 cannot communicate with the website due to operating system limitations. In at least one embodiment, the icons 201, 202, 203 may appear with color upon navigation to a supported website, in order to alert the user to possible savings. In some embodiments, the icon plug-in 201 may be an interactive user interface element, such as a button. Upon click through of this button 201, the user may be provided with e-commerce information, such as individual electronic coupon codes, which may be viewable and/or selectable. In one embodiment, the user may cut and paste coupon code(s) into a "Coupon Code" type input field of the website's checkout page.

Drawing attention to Figure 3, when a "checkout page" of an e-commerce website is detected by the application 100, the dormant or embedded browser extension 103 may be activated or invoked into an already loaded webpage 102. As illustrated, the browser extension 103, here in the form of a user interface 'panel' 300, may be slid or injected into the current website's "checkout page" in order to alert the user to possible savings. The browser extension 103, may take various shapes or colors and be injected into different areas of the current website 102 in order to best alert the user. The browser extension 103 may include a coupon code

message 301 indicating to the user that the application 100 has provided the user with at least one electronic coupon code and an interactive button, such as a “Try Codes” button 302, prompting the user to try the codes against the items currently in the user’s shopping cart. In one embodiment, the browser extension 103 may automatically begin checking and applying
5 electronic coupon codes against cart items upon activation.

In one embodiment, upon detecting a “checkout page” the application 100 may parse the user’s shopping cart and capture item specific information, such as description(s) or product identifier(s), send as parameters to the application server, and retrieve only product specific electronic coupon codes, as known to those skilled in the art.

10 Upon selecting the “Try Codes” button 302, the application 100 will check or test each electronic coupon code saved locally against the items within the user’s shopping cart. In one embodiment, the application 100 may check all the electronic coupon codes within a background process, hidden to the user. In other embodiments, the browser extension 103 may provide the user with the results on the UI panel 300 as each electronic coupon code is checked
15 in sequence. As illustrated in Figure 4, the browser extension 103 may initially display a sequential list 303 of all electronic coupon codes to be tested, each code including a coupon code description 304. As each electronic coupon code is checked against the website, the application 100 may update the UI panel 300 with “live” results to allow the users to have updated feedback for each code tested in the process. For example, if a electronic coupon code
20 is checked and found to be successful in saving the user money, the application 100 will calculate a coupon code savings 305, representing how much money the electronic coupon code will save the user. In one embodiment, the application will display the coupon code savings 305 next to the coupon code description 304 of the UI panel 300. In at least one embodiment, if a electronic coupon code has a negative or failure result, the application 100
25 may draw a line through the coupon code description 304 indicating to the user that the electronic coupon code does not save the user money on any items within the current shopping

cart. In another embodiment, the application 100 may put an 'X' in the coupon code savings 305, indicating the failure of the electronic coupon code.

As illustrated in Figure 5, the application may rank the coupon code descriptions 304 by coupon code savings 305 in order of money that may be saved from highest to lowest. If the website's checkout page only allows one coupon code description 304 per transaction, the application 100 will apply the highest or best ranked coupon code savings 305 into the 'Coupon Code' input field of the website's checkout page. In one embodiment, where the website's checkout page allows more than one coupon code description 304 to be applied, but has a 'maximum' electronic coupon code rule, the application 100 may check all electronic coupon code combinations from the prior 'pool' of the successful electronic coupon codes per the saved results, and apply the best code combination (that saves the user the most money). In another embodiment, where the website's checkout page allows more than one coupon code description 304 to be applied and does not have a 'maximum' electronic coupon code rule, the application 100 may apply all successful electronic coupon codes in order from the best or highest coupon code savings 305 to the lowest.

Figure 6 illustrates the browser 102 and browser extension 103 after the application 100 has applied the best code(s) to the website. The UI panel 300 of the browser extension 103 may include the coupon code description(s) 305 of applied electronic coupon code(s) and the total coupon code savings 306 indicating how much money the user saved per applied electronic coupon code(s). The UI panel 300 may also include a button, such as 'BACK TO CART' button 307, which may put the UI panel back to a dormant state upon click through. In one embodiment, the application 100 may automatically enter the applied code(s) into the user's shopping cart 'Coupon Code' input field of the website's checkout page. In another embodiment, the application 100 may automatically update or reload the user's view to properly reflect the results of applying electronic coupon code(s) and its changes on the 'Discount' and/or 'Total' fields, or equivalents, of the website's checkout page.

In another embodiment, the present invention may be implementable on the application server 110 in a mobile environment in accordance with implementation on an application 100 as represented in Figure 1. This embodiment of the present invention may provide a method for automatically providing and applying e-commerce savings, such as electronic coupon codes and/or “cash back” in electronic commerce transactions, as described in greater detail hereinafter. Drawing attention to Figures 7 – 11, embodiments of the present invention are illustrated from an end user’s perspective.

Upon initially navigating to a website 1102, the application 100 may automatically detect and capture certain website URL information, such as the domain name, to send as a parameter to the application server 110 in order to validate that the website 1102 is supported by the application 100. If supported, the application 100 may automatically retrieve the electronic coupon codes and/or “cash back” associated with the website 1102. The application 100 may detect, capture, and/or retrieve within a background process, hidden to the user. In one embodiment, the application 100 may prompt the user to ‘manually’ detect the website URL information, via an interactive user interface element, such as a button, in order to retrieve the website e-commerce savings.

The application 100 may store the e-commerce savings information locally, such as, to the client device’s 101 temporary local storage until the user navigates to the current website’s “checkout page” which may alert the user to possible savings. Upon navigation to a different website, the application 100 will repeat the above process. A website’s “checkout page” may at least partially comprise a webpage of the website wherein the user is able to initiate an e-commerce transaction. In one embodiment, the “checkout page” may comprise one webpage. In other embodiments, the “checkout page” may comprise more than one webpage with differing names, for example, a “shopping cart page” and a “checkout page”.

In one embodiment, the application 100 may retrieve the website’s ‘ruleset’ from the application server 110. A ruleset may comprise certain website information, such as a webpage

element. For example, the ruleset may comprise the location of a webpage field within a webpage(s), such as a coupon code input field, in order to determine if the user has navigated to the website's "checkout page". The application 100 may monitor the current website 1102 to determine if the user has navigated to the "checkout page" of the website, for example, via the coupon code input field. In another embodiment, the application 100 may retrieve an external provider ruleset from the application server 100 that is at least partially associated with the website's 'ruleset'.

Drawing attention to Figure 7, when the application 100 determines that the user has navigated to the supported website's "checkout page", the application 100 may 'activate' and inject an icon plug-in 1201 into the into an already loaded web browser in order to alert the user that electronic coupon codes and/or "cash back" are available. In one embodiment, the icon plug-in 1201 may automatically be injected on the user's web browser, as is represented in Figure 7. In another embodiment, the icon plug-in 1201 may be injected within the browser and accessible to the user after selecting a web browser button, such as the "share" button.

The icon plug-in 1201 may comprise a coupon code icon 1202, representing a total amount of electronic coupon codes from the application server 110 for the current website. In one embodiment, the icon plug-in 1201 may also comprise a cash back icon, for example, indicated by dollar signs. The cash back icon 1203, may indicate to the user that the current website offers some form of 'cash back' on purchases made via the current e-commerce website. The icon plug-in 1201 may be an interactive user interface element, such as a 'clickable' or selectable button.

More specifically, for example, the coupon code icon 1202 may display the number '2' indicating that there are two merchant electronic coupon codes provided for the user by the application server 110. Upon selection of the icon plug-in 1201, the user may be provided with specific e-commerce information, such as individual electronic coupon codes, which may be

viewable and/or selectable. In one embodiment, the user may ‘cut and paste’ coupon code(s) directly into an input field of the website’s checkout page.

Drawing attention to Figure 8, the present invention may comprise a browser plug-in or extension 1103 that may appear dormant until being ‘activated’ and injected into the web browser. More specifically, upon selection of the icon plug-in 1201, the dormant or embedded browser extension 1103 may be activated or injected into the already loaded “checkout page” 1102. As illustrated, the browser extension 1103, here in the form of a web browser ‘panel’ 1300, may be slid or injected into the current website’s “checkout page” in order to alert the user to specific possible savings.

The browser extension 1103, may take various shapes or colors and be injected into different areas of the current website 1102 in order to best alert the user. The browser extension 1103 may include a coupon code message 1301 indicating to the user that the application 100 has provided the user with at least one electronic coupon code and an interactive button, such as an “AUTO APPLY COUPONS” button 1302, prompting the user to check the coupon code(s) against the items currently in the user’s ‘shopping cart’ of the website’s “checkout page”. In one embodiment, the browser extension 1103 may automatically begin checking electronic coupon codes against cart items upon activation or injection. In another embodiment, upon detecting a “checkout page” the application 100 may parse the user’s shopping cart and capture item specific information, such as description(s) or product identifier(s), to send as parameters to the application server 110 and retrieve only product specific electronic coupon codes.

As illustrated in Figure 9, upon selecting the “AUTO APPLY COUPONS” button 1302, the application 100 will check or test each electronic coupon code against the items within the user’s shopping cart. The browser extension 1103 may provide the user with the results on the ‘panel’ 1300 as each electronic coupon code is checked in sequence. If there is more than one coupon code, the browser extension 1103 may initially display a sequential list 1303 of all

electronic coupon codes to be tested. Each coupon code may include a coupon code description 1304. In one embodiment, the application 100 may check all the electronic coupon codes within a background process, hidden to the user.

As each electronic coupon code is checked against the website 1102, the application
5 100 may update the 'panel' 1300 with "live" results to allow the user to have updated feedback for each code tested in the process. For example, if an electronic coupon code is checked and found to be successful in saving the user money, the application 100 may calculate a coupon code savings, representing how much money the electronic coupon code will save the user. In one embodiment, the application will display the coupon code savings next to the coupon code
10 description 1304 of the panel 1300.

In at least one embodiment, if an electronic coupon code has a negative or failure result, the application 100 may draw a line through the coupon code description 1304 indicating to the user that the electronic coupon code does not save the user money on any items within the current 'shopping cart'. In another embodiment, the application 100 may put an 'X' near the
15 coupon code description 1304, indicating the failure of the electronic coupon code.

The application 100 may check the coupon codes against the website by executing a background process to determine whether the electronic coupon code(s) is successful or fails to achieve a reduction of the user's e-commerce transaction amount. The result of determining whether the electronic coupon code(s) is successful or fails to achieve a reduction in the user's
20 e-commerce transaction amount may comprise "response" information from the website, which may be stored locally on the user's device.

As illustrated in Figure 10, the application 100 may display the total amount the user may save from the coupon code(s) that were checked against the website's checkout page. An interactive button, such as an "UPDATE CART" button 1305, may prompt the user to apply
25 the coupon code(s) against the user's 'shopping cart' of the website's "checkout page". In one embodiment, the application 100 may apply the checked coupon code(s) automatically. Upon

selection, the user's shopping cart will be updated to reflect the savings provided by the applied coupon code(s).

In one embodiment, the application 100 may rank the coupon codes in order of money that may be saved, such as from highest to lowest. If the website's checkout page only allows one coupon code per transaction, the application 100 will apply the highest or best ranked coupon code input field of the website's checkout page. In another embodiment, where the website's checkout page allows more than one coupon code to be applied, but has a 'maximum' electronic coupon code rule, the application 100 may check all electronic coupon code combinations from the prior 'pool' of the successful electronic coupon codes per the saved results, and apply the best code combination (that saves the user the most money).

In another embodiment, where the website's checkout page allows more than one coupon code to be applied and does not have a 'maximum' electronic coupon code rule, the application 100 may apply all successful electronic coupon codes in order from the best or highest coupon code savings to the lowest. In one embodiment, the application 100 may prompt the user to apply at least one coupon code against the website's checkout webpage. In another embodiment, the application 100 may automatically apply at least one coupon code against the website's checkout webpage.

Figure 11 illustrates the e-commerce website 1102 after the application 100 has applied the best coupon code(s) to the shopping cart items of the website. As illustrated, the website 1102 may display a "new" total amount after applying the coupon code(s) and other discounts. In one embodiment, the application 100 may automatically enter the applied code(s) into the user's 'shopping cart' coupon code input field of the website's checkout page. In another embodiment, the application 100 may automatically update or reload the user's view to properly reflect the results of applying electronic coupon code(s) and its changes, for example on a 'Discount' and/or 'Total' fields, of the website's "checkout page", as illustrated in Figure 11.

Additionally, when the user navigates to the checkout page, the application 100 may 'activate' the cash back functionality associated with the e-commerce transaction within a background process, hidden to the user. In one embodiment, while the application 100 'activates' the cash back functionality, the application 100 may concurrently inject a browser
5 plug-in into the checkout page of the web browser. The application 100 may automatically activate the cash back functionality. More specifically, when the user has navigated to the checkout page, as illustrated in Figure 11, the application 100 may capture certain information, such as the website domain name, the total amount of the purchase, user information (for example, user number, user name, user address, etc.) to send as a parameter(s) to the application
10 server 110 in order to validate that the website 1102 is supported or affiliated by the application 100.

If supported, the cash back functionality may be 'activated' or saved to the application server 100 in order to provide the user with his or her cash back savings. If activated, e-commerce transaction information, such as the total amount of the purchase, user information,
15 etc., may be saved to the application server 110. The application 100 may calculate and provide the user with the cash back savings, for example, in the form of a rebate check. In another embodiment, the application 100 may prompt the user to 'manually' activate the cash back functionality, via an interactive user interface element, such as a button. Upon activation, the application 100 may display the cash back savings in the web browser, for example, via a
20 browser plug-in.

As mentioned above, the application server 110 may save or track the user's e-commerce transaction, via user information, such as the user number, user name, etc. In one embodiment, the application 100 may track an 'unknown' user's e-commerce transactions anonymously, in order to offer the user cash back on purchases. Tracking the e-commerce
25 transaction anonymously may comprise calculating a 'cash back' amount earned associated with the e-commerce transaction. The application 100 may track the e-commerce transaction

across a plurality of separate e-commerce transactions, calculating the cash back amount earned associated with each of the plurality of separate e-commerce transactions, and associating each of the cash back amounts with the user via an anonymous client device identifier, as known to those in the art.

5 In another embodiment, the present invention may be implementable on the application server 110 in a desktop or mobile environment in accordance with implementation on an application 100 as represented in Figure 1. Once downloaded to the user's mobile or desktop device 101, the present invention will automatically detect when the user has initiated a shopping experience and activate. As such, it will not be necessary for the user to 'open' the
10 present invention on his/her mobile or desktop device 101 prior to, or during, the shopping experience. This feature will save the user time and reduce "click counts" in their shopping experience, which will allow users to complete purchases faster which increases the chances for a sale.

 Drawing attention to Figure 12, a method of the present invention may initially
15 comprise detecting a URL or domain name of a website to determine if the webpage is supported by an application server by submitting parameters, such as a URL address or domain name, to the application server, as in 901. The application may continuously monitor for URI/URL changes and automatically access the application server to determine if the webpage is supported. The application may access the application server via a mobile application
20 interface installed on the client device, or via a browser accessing the application server as described above. In some embodiments, websites may not be in 'communication' with the application due to operating system limitations and may require a manual detection to determine if the current website is supported by the application server. This manual detection may include another browser extension, designed to capture URL or domain name information,
25 such as by DOM parsing of the website. This information may then be submitted to the application server.

Next, in 902, if the website is supported by the application server, specific stored merchant information, such as but not limited to, a website ruleset, electronic coupon codes, and cash back settings, is retrieved from the application server and saved locally on the user's device. In one embodiment, all electronic coupon codes of the supported website may be
5 retrieved from the application server. In another embodiment, all electronic coupon codes may be sorted by the application server based on factors such as past success or user feedback and retrieved. In other embodiments, electronic coupon codes that are more probable for success may be compiled by the application server into a specific list or queue designed to maximize the user's savings, based on several factors of a predefined algorithm or procedure.

10 For example, some factors may be: success rates reported by users of the application; success of electronic coupon codes applied by the application processes; frequency of appearance of the electronic coupon code(s) on public website(s); or electronic coupon codes containing non-redundant keyword patterns. The electronic coupon codes stored on the application server may be harvested in various ways, such as, programmatically or manually
15 provided by merchant websites via affiliate agreements; collected from general affiliate pools or networks; collected from the internet or other publicly available sources; or submitted by users.

Next, at 903, the application may continuously monitor the URI/URL and webpage elements to detect when a user has navigated to a checkout page or process of the website. If
20 the application detects that the user has navigated to another website, such as by URI/URL changes, the application will repeat step 901. If the application detects certain page elements (based on the website's retrieved ruleset) that indicate the user has navigated to the checkout page, the application may move on to the next step to check electronic coupon codes. In one embodiment, the retrieved ruleset may require item or product specific electronic coupon codes
25 only to test. In such an embodiment, prior to checking the codes, the application may parse the

user's shopping cart, capture item information and submit to the application server, retrieve only the item specific electronic coupon codes, and save locally on the user's device.

In some embodiments, websites may not be in 'communication' with the application due to operating system limitations and may require a manual detection as described above. In
5 such embodiments, the application will immediately detect current page elements from the webpage, in which the manual detection was executed from, in order to determine if the user has navigated to the checkout page.

Users next may be prompted to check electronic coupon codes against the item(s) in the user's electronic shopping cart, as in 904. In one embodiment, the application will
10 automatically begin applying electronic coupon codes by executing a background process. In other embodiments, the application may inject a user interface element into the website's checkout page prompting the user to begin checking all or specific electronic coupon codes. The check codes function of the application server may find the user the best possible electronic coupon code(s) with the most possible total savings. The application may save the current
15 subtotal and/or total order amount of the user's cart prior to checking the codes.

In some embodiments, the ruleset may define website provided coupon or promo codes. In such embodiments, where the codes have already been applied to the user's shopping cart, the application may parse and save those applied codes in order to prevent interference with the electronic coupon codes retrieved from the application server.

20 Electronic coupon codes may be checked against the website's checkout page individually through a defined background process, such as an Ajax request. In one embodiment, some or all electronic coupon codes may be tried all at once by opening and parsing several processes. In another embodiment, some codes may be tried through a "form post" process. In one embodiment, some codes may be tried using device specific functionality,
25 e.g. webview and parsing, such as due to mobile operating system limitations.

In certain embodiments, websites may not be able communicate with the application, or else the application may not be able to detect a URL, due to operating system limitations and may require an alternate way to check electronic coupon codes and determine whether the electronic coupon codes are successful or result in discounts being applied. Accordingly, the application may be configured to build and run a separate script or extension, encapsulating all logic elements and check code functionality as discussed herein, that will pass the electronic coupon codes to the script and check the codes in a submitted backend process, and inject the script into the frontend webpage without requiring the user to reload the webpage.

When a electronic coupon code is checked or validated against the website's checkout page, each response may be saved locally to the user's device and also submitted to the application server for analytic purposes. The application may capture the response, for example, by standard DOM parsing rules or may use site-specific provider rules. In some embodiments, such as websites that require checking electronic coupon codes independently, response information, such as coupon code description and the amount saved, or the rejection reason, may be saved to the local storage of the user's device. In other embodiments, the application may parse the total amount of the user's cart in conjunction with the checked electronic coupon code, and calculate the difference in price, and save this information locally on the user's device. In one embodiment, the application may update the user interface with the results of the check code process in order to allow users to have updated feedback for each code tried in the process(es).

In one embodiment, the ruleset may require the application to copy the user's shopping cart to verify whether the latest electronic coupon code has been applied successfully. In another embodiment, the ruleset may require the application to clear a checked code after the response information has been captured and before checking the next code in the list or queue.

The application may review the local storage on the user's device, determine the best code(s), and apply the code(s) to the user's shopping cart. In one embodiment, the user's cart

may only allow one electronic coupon code. If the cart automatically overrides or erases the applied code per the ruleset, the application may re-apply the best code to the user's cart. If the cart requires that the code be cleared per the ruleset, the application may clear the existing code, and re-apply the best code to the user's cart. In another embodiment, the user's cart may allow
5 more than one electronic coupon codes.

If the cart allows more than one code but has a maximum rule per the ruleset, the application may build a new queue of combinations of successful electronic coupon codes that may be applied together. In such an embodiment, the application may apply a plurality of code combinations, capturing the response information of each grouping as discussed above to
10 accurately apply the best code or combination of codes. The best code or combination of codes may be ranked in order from highest to lowest amount saved. If the ruleset defines a limit to the amount saved by codes or total codes that may be applied to a user's cart, the application may apply the best possible combinations to maximize the amount saved.

Next, in 905, the application may display the electronic coupon codes applied to the
15 user's cart and a "Total Amount Saved" message to the user interface. The application may allow the user to "retry" the codes, for example, after altering the cart. In one embodiment, additional savings may be available by manual application of codes to the user's cart. In such embodiments, the application may display a message or highlight the availability to the user. In some embodiments, the user's cart may require a webpage refresh to display the most recent
20 application of electronic coupon codes applied, for example via a background process. The application may record the total amount saved, after the user completes the checkout, for historical purposes.

Next, optionally, in 906 the application may track the user's e-commerce transactions in order to offer the user cash back on purchases anonymously. More specifically, in step 902,
25 cash back settings of an e-commerce website may be retrieved from the application server. For example, an e-commerce website may offer 3% 'cash back' on purchases to the user, as known

in the art. In the preferred embodiment, the application may track the user's purchases or transactions anonymously, such that the user may use the cash back functionality after installing the application on his or her device, without divulging personal information to the application. For example, after installing the application, the application may generate a unique
5 device identifier. When the user utilizes the application via the device, such as: purchases, API requests, selecting or clicking "Cash back" button(s), or logging in or creating an account with the application, the application server will detect the unique identifier and save the information. Next, the application will retrieve information, such as cash back earnings, from the application server to update the device's local storage with the cash back earnings.

10 The application allows the user to later create an account through the application. On occurrence, the application server will compile all transactions assigned to the unique device identifier and re-assign them to the new user account. In one embodiment, if the user creates an account on another device with the application (or via the website), the user may link the account with the identifier via received "signed communications" such as emails or text links.

15 More specifically, a user may receive signed communications on the device that the application is installed. These signed communications may be detected by application listeners on other devices, wherein the application will store the signed communication. Next, the application may send the signature to the application server (which the signature is attributed to the user) and retrieve user data to the application on the new device. The application may then store the
20 user data in permanent storage.

In another embodiment, if the user logs in via the application website, the application may detect the login cookies and session data that is set by the website on the next visit, and logs in the user through the application by storing a permanent device identifier on the device's local storage and on the application server. The user then may log in/out of the website and
25 clear cookies, but the application will remain assigned to the appropriate user id by saving unique application keys in the device's local storage as long as the application remains

installed. This is because the user's login status is "cookieless" once assigned to the application and stored permanently in the device's storage.

Drawing attention to Figures 13 – 16, embodiments of the present invention are illustrated from an end user's perspective. In order to begin a shopping experience, a user may initially navigate to or open a target e-commerce interface 2102 of a target merchant 'store'. More specifically, the target e-commerce interface 2102 of a target merchant store may comprise the target merchant store's website, such as a mobile website, or a native downloadable application, such as a native mobile application. Accordingly, for purposes of clarity and without limiting the scope of the present invention, this invention will be described with reference to the target e-commerce interface 2102 and a competing e-commerce interface 2202 of at least one competing merchant store being implemented in a desktop environment using a web browser. However, it is emphasized that the target e-commerce interface 2102 and/or the competing e-commerce interface 2202 of the at least one competing merchant store can be operable in addition to and other than a desktop environment (e.g., mobile web or native application environment).

Further, after the user navigates to or opens the target e-commerce interface 2102, the application 100 automatically detects when the user is viewing at least one product or service and automatically captures certain information, such as but not limited to, the product or service item listing. The item listing may contain specific information about the product or service and general information about the target e-commerce interface 2102, which can be sent as parameters to the application server 110 in order to validate that the target e-commerce interface 2102 is supported by the application server 110. If supported, the application server(s) 110 may compare the item listing(s) with item listing(s) of other e-commerce merchant 'stores' in order to find the lowest price available, as described in greater detail below. The item listing may include any conceivable product or service that the user is viewing during the shopping experience, including travel services for booking hotels, airfare, tours, etc.

For purposes of clarity and without limiting the scope of the present invention, this invention will be described with reference to an item price of an item listing being the current lowest price of said product or service *after* all price factors have been applied to the base price of same. As such, an item price may include price factors such as, but not limited to, sales tax
5 amount, shipping cost amount, vendor fees amount, resort fees amount, coupon savings, cashback or rebate savings, or any combination of the above.

It should be noted that a key component of any online transaction cost is sales taxes and shipping costs. In order to properly (and automatically) calculate these items, the user's location needs to be determined. In one embodiment, the user's location may be determined
10 based on the IP address of his/her client device 101. In another embodiment, the user's location may be determined based on his/her preferred shipping address, which may or may not be visually visible on the merchant website or application. After determining the location, the sales tax and shipping costs can be determined without requiring the user to navigate to the competing merchant store. This is unique because sales taxes and shipping costs may vary from
15 merchant to merchant depending on the merchant's location relative to the user (e.g., due to sales tax nexus, lower shipping costs due to warehouse location, etc.). Further, this may be accomplished in a "background" process, hidden from what the user can visually see on the screen.

As an example, referencing the hypothetical table below, Competing Merchant C has
20 the highest base price of the competing merchants, but after applying the cashback savings and coupon savings, the present invention has automatically calculated that the lowest item price would currently come from Competing Merchant C. In addition, since Competing Merchant A and Competing Merchant B also have a lower item price than the Target Merchant, the present invention may retrieve and display all three item listings of the Competing Merchants to the
25 client device 101.

Vendor	Base Price	Cashback	Coupon	Item Price	Total Benefit
Target Merchant	\$110	0%	\$0	\$110	\$0.00
Competing Merchant A	\$100	\$5	\$0	\$95	\$5.00
Competing Merchant B	\$101	5%	\$5	\$90.95	\$19.05
Competing Merchant C	\$102	10%	\$5	\$86.80	\$23.20

In one embodiment, the application 100 may provide the user with various item price factors (tax amount, etc.) that allows the user to manually select what he/she would like the item price to include prior to comparison.

- 5 Each item listing may comprise an e-commerce interface identifier and certain product and/or service information, such as but not limited to: item code, including a product or service identifier; product ID code; service ID code; UPC code; SKU code; product name; service name; item state (e.g., a new or used product); URL information (if a website is used); base price; check in date; check out date; hotel name; hotel address; hotel location based on a map;
- 10 hotel star rating; customer ratings; user reviews; flight number; airline; departure date and time; arrival date and time; car rental pickup date and time; car rental drop-off date and time; car rental supplier; car rental provider; car rental category; car rental type; car make and model; number of travelers; number of passengers; number of bags; number of large bags; number of small bags; departure airport; layover airport; arrival airport; activity name; activity start date
- 15 and time; activity end date and time; activity location; tour name; tour date; shipping from

location; room type; number of beds; bed type; number of guests; number of adults; number of children; number of nights; room category; room name; cancellation policy; and latest cancellable date.

If the target e-commerce interface 2102 is supported by the application 100, the application 100 may automatically compare at least one item listing of the target merchant store with the item listing of at least one competing merchant store in order to determine whether a lower price exists for the product(s) and/or service(s). More specifically, the application server 110 may compare whether the item listing from at least one competing merchant store has a lower item price than the item listing from the target merchant store for the same product(s) and/or service(s). If the target merchant store's item listing has the lowest item price, the application 100 may remain hidden to the user. However, if the item listing from at least one competing merchant store has a lower item price, the application 100 may automatically retrieve the lower-priced item listing(s) and display a message on the client device 101 in order to alert the user.

In at least one embodiment, the present invention may automatically retrieve an item listing from at least one competing merchant store that is similar to, but not the same as, the item listing from the target merchant store. More specifically, if a lower-priced item listing from at least one competing merchant store is found, it will typically be an exact match of the product or service that the user is viewing in the target merchant store (for example, an exact match may be matching the item codes of two televisions). However, a competing merchant store may not have the exact product or service, but does have a "similar" product or service (or has a lower-priced similar product or service in addition to the exact product or service). For example, a television brand may sell one unique model through Walmart and another unique model through Best Buy, but said models may have slight differences (such as one model including an additional HDMI port) and as such, have different item codes. Accordingly, if a "similar" item listing from at least one competing merchant store has a lower item price

than the item listing of the target merchant store, the application 100 may automatically retrieve the “similar” item listing(s) to display on the client device 101 in order to alert the user.

The application 100 may detect, capture, calculate, compare, and retrieve the item listing(s) within a background process, hidden to the user. In one embodiment, the application
5 100 may prompt the user to ‘manually’ detect, capture, compare, and/or retrieve the item listing(s), via an interactive button.

As illustrated in Figures 13 – 14, when the application 100 determines that the item listing(s) from the at least one competing merchant store has a lower item price than the target merchant store, the application 100 may automatically display said item listing(s) within the
10 target e-commerce interface 2102. This ‘display’ within the target e-commerce interface 2102 may be facilitated by a plug-in or extension 2103 that may appear dormant until being ‘activated’ by the application 100 and automatically “injected” into the native application or browser. More specifically, upon retrieval of a lower-priced item listing(s), the dormant or embedded extension 2103 may be automatically activated and displayed into the already loaded
15 target e-commerce interface 2102. Moreover, the extension 2103 may be slid or injected into the target e-commerce interface 2102 in order to alert the user to possible savings at another store.

The extension 2103, may comprise at least one interactive user interface element, such as a button, take various shapes and colors, and be injected into different areas of the target e-
20 commerce interface 2102 in order to best alert the user to possible savings. The extension 2103 may include a price compare message 2301 to alert the user that the application 100 has provided at least one lower-priced item listing. The price compare message 2301 may comprise an interactive button to allow the user to select the lower-priced item listing. As illustrated in Figure 14, if the user is viewing multiple item listings in the target e-commerce interface 2102,
25 the application 100 may retrieve and display multiple extensions 2103 injected therein.

The application server 110 may retrieve and display the item listing of the at least one competing merchant store directly into the target e-commerce interface 2102. If the application 100 retrieves multiple item listings from the at least one competing merchant store, it may rank the retrieved item listings sequentially from lowest item price to highest item price. This may allow the user to “swipe” through the ranked item listings of the at least one competing merchant store, for example one by one, to view all options before making a selection. Moreover, if the user is shopping on a client device 101 that comprises a small screen (e.g., cell phone), it may not be optimal to display all ranked item listings due to the small-screen size. Upon navigation to a different item listing, the application 100 will repeat the above steps.

As illustrated in Figure 14, the application 100 may automatically display at least one extension 2103, each representing a retrieved item listing of an associated competing merchant store. Each extension 2103 may comprise an interactive button, such as the price compare message 2301 in order to prompt the user to select the displayed item listing of the at least one competing merchant store. Upon selection of the price compare message 2301 button, the application 100 may automatically redirect the user to the competing e-commerce interface 2202 (illustrated in Figure 16) associated with said competing merchant store, in order to initiate an e-commerce purchase.

As illustrated in Figure 13, in another embodiment, the application 100 may automatically display the extension 2103 representing a retrieved item listing of the associated competing merchant store. The extension 2103 may comprise an interactive button, such as the “Add to Cart” button 2302 in order to prompt the user to select the displayed item listing of the at least one competing merchant store. By selecting the “Add to Cart” button 2302, representing the lower-priced item listing of the associated competing merchant store, the application 100 may automatically apply the item listing into the associated competing e-commerce interface 2202, in a background process hidden to the user.

More specifically, the selected item listing of the competing merchant store may be automatically applied into the associated competing e-commerce interface 2202, in a background process that may be hidden to the user, to allow the user to continue the shopping experience prior to initiating an e-commerce purchase. The competing e-commerce interface 5 2202 may comprise the competing merchant store's website, including a mobile website, or a native application, including a native mobile application. Accordingly, for purposes of clarity and without limiting the scope of the present invention, this invention will be described with reference to the competing e-commerce interface 2202 being in the form of a desktop website. However, it is emphasized that the competing e-commerce interface 2202 of the present 10 invention can be operable in addition to and other than a desktop website. Moreover, the competing e-commerce interface 2202 may be in the form of a native mobile application, a mobile website, or a desktop application.

The application 100 may apply the selected item listing of the at least one competing merchant store into the associated competing e-commerce interface's 2202 "virtual shopping 15 cart" and display a reference to the applied item listing(s) within the target e-commerce interface 2102. For example, and as illustrated in Figure 15, the application 100 may comprise a cart message 2303 indicating that the application 100 has applied at least one item listing into the competing e-commerce interface 2202 of the at least one competing merchant store. The cart message 2303 may comprise listing item information, such as the item price and the name 20 of the competing merchant store. In one embodiment, the application 100 may prompt the user to view the "virtual shopping cart", via an interactive user interface element, such as a "See Cart" button 2304, in order to view all the selected item listings contained in each "virtual shopping cart".

In another embodiment, the application 100 may apply the selected item listing(s) into 25 the associated competing e-commerce interface's 2202 "electronic shopping cart" and/or "checkout area". As such, if the user independently (for example, not via the application 100)

navigates to the associated competing e-commerce interface 2202, the selected item listing(s) may still be applied therein. A “checkout area” may at least partially comprise a web or native application ‘page’ wherein the user is able to initiate an e-commerce purchase.

As also illustrated in Figure 15, the application 100 may prompt the user to redirect to the competing e-commerce interface 2202 to initiate an e-commerce purchase. The application 100 may prompt the user to redirect to the associated competing e-commerce interface 2202, via an interactive user interface element, such as a “Checkout” button 2305, in order to initiate an e-commerce purchase.

As illustrated in Figure 16, upon redirection to the competing merchant interface 2202, the user may complete his/her e-commerce purchase. If the user has selected item listings from multiple competing merchant stores, the application 100 may direct the user to each competing e-commerce interface 2202 in order to complete all e-commerce purchases for the selected item listings. In one embodiment, after completing an e-commerce purchase at a competing e-commerce interface 2202, the application 100 may redirect the user back to the target e-commerce interface 2102 and prompt the user to ‘manually’ direct, via an interactive button, to the next competing e-commerce interface 2202 to initiate an e-commerce purchase therein.

Turning to Figure 17, a flow chart 1700 depicting operative logic of a software embodiment of the present invention, which optimizes a user experience is disclosed. While the disclosed embodiment contemplates coupons and cash back for purposes of facilitating disclosure, the inventive method may be employed with any user benefit contemplated by the disclosure of this invention. The software embodiment may comprise a mobile application, desktop application, or web extension, as disclosed and described herein. According to 1701, a user initiates a shopping trip by accessing a website or mobile application to engage in an e-commerce transaction as previously disclosed. The user may accomplish this via a web browser or via a merchant’s native retail application. According to 1702 the user determines which

product or service the user will purchase and adds the product or service to a cart, in accordance with previously disclosed embodiments.

According to 1703, the user then navigates to the cart or other checkout interface to begin the transaction. In accordance with previously disclosed embodiments, a software
5 embodiment of the present invention may operatively monitor a URL or page elements as the user navigates the e-commerce interface in order to determine when the user has landed on the page which will begin the transaction. The page elements may comprise keywords, images, or logos that are operatively associated with a checkout interface. Assuming coupons are available, as per 1704, the software embodiment of the method may be operative to initiate a
10 ruleset to identify and apply benefits (coupon codes as depicted in the Figure) in a background process that is not visible to the user, and without prompting the user, as at 1705.

As the software embodiment conducts the identification and application of coupons in a background process 1705, the embodiment may be further operative to determine whether the merchant is affiliated with a cashback benefit 1707, as previously disclosed herein. If so,
15 then the embodiment may present a modal image to the user inquiring as to whether the user would like to obtain coupons or cashback 1709. If the user affirmatively responds to the call to action 1710, then an expedited modal for applying coupons can be presented to the user 1711. Such an expedited modal may include an indication to the user that benefits are being identified and/or applied while the step 1705 completes operation in the background. If step 1705 has
20 already completed by the time the user requests application of a benefit, then presenting a modal to the user 1711 may further include the step of presenting the modal to the user for a predetermined time period, which in a preferred embodiment, may be approximately 3-4 seconds. Accordingly, the user is left with the impression that step 1705, identification and application of coupons, was conducted in a much shorter time frame than may have been
25 necessary. Furthermore, in the event that step 1705, identification and application of coupons, is completed prior to the user affirmatively responding to the call to action 1710, then the step

of presenting an expedited modal for applying coupons to the user 1711 will also serve the purpose of assuring the user that this step was not inadvertently skipped.

Once the modal is presented to the user 1711, then the user may continue the transaction, substantially as disclosed in previous embodiments. This may include presentation
5 of further modals to indicate how much money the user saved by utilizing the present invention. Alternatively, if no benefits were available, a further modal may indicate to a user that the present invention saved the user the time involved to manually search for, and confirm, the fact that no benefits were available with respect to the particular transaction.

Of course, the particular circumstances of a given transaction may not present the same
10 set of factors as are disclosed in Figure 17, which is intended to be exemplary. If no coupons are available at step 1704, then the invention may continue on to ascertain and select other benefits, such as cashback, price comparison, and/or presentation of loyalty perks, substantially as disclosed in other embodiments. Additionally, if the merchant is not affiliated at step 1707, then no cashback may be available for a user. As such, the operative logic would proceed to
15 step 1708, in which the modal presented to the user does not indicate that cash back is available. The call to action presented to the user, as at step 1710, would similarly not indicate that cash back is available.

Since many modifications, variations and changes in detail can be made to the described preferred embodiment of the invention, it is intended that all matters in the foregoing
20 description and shown in the accompanying drawings be interpreted as illustrative and not in a limiting sense. Thus, the scope of the invention should be determined by the appended claims and their legal equivalents.

What is claimed is:

1. A method for providing e-commerce information associated with an e-commerce transaction, to users of an electronic device comprising:

detecting automatically at least one item listing of a target merchant store in a target e-commerce interface;

displaying automatically the at least one item listing of at least one competing merchant store within the target e-commerce interface;

prompting the user automatically to select the at least one item listing of the at least one competing merchant store within the target e-commerce interface;

applying automatically the at least one item listing of the at least one competing merchant store to a competing e-commerce interface associated with the at least one competing merchant store; and,

configuring at least one application server to include at least one database having e-commerce information associated with a plurality of e-commerce interfaces.

2. The method of claim 1 wherein the target e-commerce interface comprises a website of the target merchant store.

3. The method of claim 2 wherein the website of the target merchant store comprises a mobile website of the target merchant store.

4. The method of claim 1 wherein the target e-commerce interface comprises a native application of the target merchant store.

5. The method of claim 4 wherein the application of the target merchant store comprises a native mobile application of the target merchant store.

6. The method of claim 1 wherein the at least one item listing comprises an item code.

7. The method of claim 6 wherein the item code comprises a product identifier.

8. The method of claim 6 wherein the item code comprises a service identifier.

9. The method of claim 1 wherein the at least one item listing comprises an e-commerce

interface identifier.

10. The method of claim 1 wherein the at least one item listing comprises an item price.
11. The method of claim 10 wherein the item price comprises a tax amount.
12. The method of claim 10 wherein the item price comprises a shipping cost amount.
- 5 13. The method of claim 10 wherein the item price comprises a vendor fees amount.
14. The method of claim 10 wherein the item price comprises a resort fees amount.
15. The method of claim 10 wherein the item price comprises coupon savings.
16. The method of claim 10 wherein the item price comprises a cashback savings.
17. The method of claim 1 further comprising:
 - 10 capturing automatically the at least one item listing of the target merchant store;
transmitting automatically the at least one item listing of the target merchant store to at
least one application server;
comparing automatically the at least one item listing of the target merchant store with
the item listing of at least one competing merchant store to determine whether the item listing
15 of the at least one competing merchant store has a lower item price than the item listing of the
target merchant store; and
retrieving automatically the item listing of the at least one competing merchant store
from the application server.
18. The method of claim 17 further comprises retrieving multiple item listings of the at
20 least one competing merchant store from at least one application server.
19. The method of claim 1 wherein applying the at least one item listing of the at least one
competing merchant store further comprises automatically applying the at least one item listing
of the at least one competing merchant store into a checkout area of the at least one competing
merchant store.
- 25 20. The method of claim 1 wherein applying the at least one item listing of the at least one
competing merchant store further comprises automatically applying the at least one item listing

of the at least one competing merchant store into an electronic shopping cart containing the at least one item listing of the at least one competing merchant store.

21. The method of claim 1 wherein applying the at least one item listing of the at least one competing merchant store further comprises automatically applying the at least one item listing
5 of the at least one competing merchant store into a virtual shopping cart containing the at least one item listing of the at least one competing merchant store.

22. The method of claim 21 further comprises prompting the user to view the virtual shopping cart within the target e-commerce interface.

23. The method of claim 1 further comprising redirecting the user to the competing e-commerce interface associated with the at least one competing merchant store.
10

24. The method of claim 23 wherein redirecting the user further comprises prompting the user to redirect from the target e-commerce interface to the competing e-commerce interface.

25. The method of claim 23 wherein redirecting the user further comprises automatically redirecting the user from the target e-commerce interface to the competing e-commerce
15 interface.

26. The method of claim 1 wherein the competing e-commerce interface comprises a website of the at least one competing merchant store.

27. The method of claim 26 wherein the website of the at least one competing merchant store comprises a mobile website of the at least one competing merchant store.

20 28. The method of claim 1 wherein the competing e-commerce interface comprises a native application of the at least one competing merchant store.

29. The method of claim 28 wherein the application of the at least one competing merchant store comprises a native mobile application of the at least one competing merchant store.

30. A method for providing e-commerce information associated with an e-commerce
25 transaction, to users of an electronic device comprising:

monitoring at least one of a URL or a page element to determine whether the user has navigated to a checkout interface associated with the e-commerce transaction;

upon determining that the user has navigated to said checkout interface, and prior to prompting the user, identifying and applying at least one benefit relative to a purchase selected
5 by the user;

after checking for at least one benefit relative to the purchase selected by the user, prompting the user to request application of said at least one benefit;

after receiving input from the user confirming application of said benefit, displaying a modal to the user operative to indicate that benefits are at least being identified; and

10 displaying the result of the user's e-commerce transaction associated with the application of said at least one benefit.

31. The method as recited in claim 30 wherein said page element comprises one element selected from the group consisting of keyword and image within said checkout interface.

32. The method as recited in claim 30 wherein the step of checking for at least one benefit
15 is conducted as a background process which is not visible on the checkout interface.

33. The method as recited in claim 30 wherein the step of prompting the user to request application of said at least one benefit is conducted after completion of the step of checking for at least one benefit relative to a purchase selected by the user.

34. The method as recited in claim 33 wherein the step of displaying a modal to the user
20 operative to indicate that benefits are at least being identified further comprises displaying said modal for approximately between 3 and 4 seconds.

35. The method as recited in claim 30 wherein the step of displaying a modal to the user operative to indicate that benefits are at least being identified further comprises displaying said modal until the step of identifying and applying at least one benefit relative to a product selected
25 by the user is complete.

36. The method as recited in claim 30 wherein said at least one benefit comprises a coupon code operative to reduce the cost of the purchase selected by the user.

37. The method as recited in claim 30 wherein said at least one benefit comprises a cashback percentage provided from a merchant to the user relative to the purchase selected by the user.

38. The method as recited in claim 30 wherein said at least one benefit comprises presentation to the user of at least one competing e-commerce interface associated with the purchase selected by the user.

39. The method as recited in claim 30 wherein said at least one benefit comprises a loyalty benefit associated with a merchant or financial institution.

40. The method as recited in claim 39 wherein said loyalty benefit comprises a customer shipping program.

41. A system for providing benefits to a user during an e-commerce transaction, the system comprising:

a software program operative to monitor a user's interaction with an e-commerce interface;

said software program further operative to determine whether a user has selected at least one purchase and navigated to a checkout interface;

said software program further operative to, prior to prompting the user, determine and apply to the user's transaction, a maximum benefit; and

wherein said maximum benefit comprises at least one benefit selected from the group consisting of coupon, cashback, price comparison, and customer shipping program.

42. The system as recited in claim 41 wherein said software program is configured to be executed at least partially on a client device operated by the user.

43. The system as recited in claim 41 wherein said software program is configured to be executed at least partially on an application server.

44. The system as recited in claim 41 wherein said software program is configured to receive and store user preferences relative to a set of merchants at which the system is operative.

45. The system as recited in claim 41 wherein said software program is configured to
5 receive and store user preferences relative to a set of benefits to be identified by the system.

46. The system as recited in claim 41 wherein monitoring a user's interaction with an e-commerce interface comprises monitoring a URL associated with said e-commerce interface.

47. The system as recited in claim 41 wherein monitoring a user's interaction with an e-commerce interface comprises monitoring a page element associated with said e-commerce
10 interface.

48. The system as recited in claim 47 wherein said page element comprises a keyword associated with said checkout interface.

49. The system as recited in claim 47 wherein said page element comprises an image associated with said checkout interface.

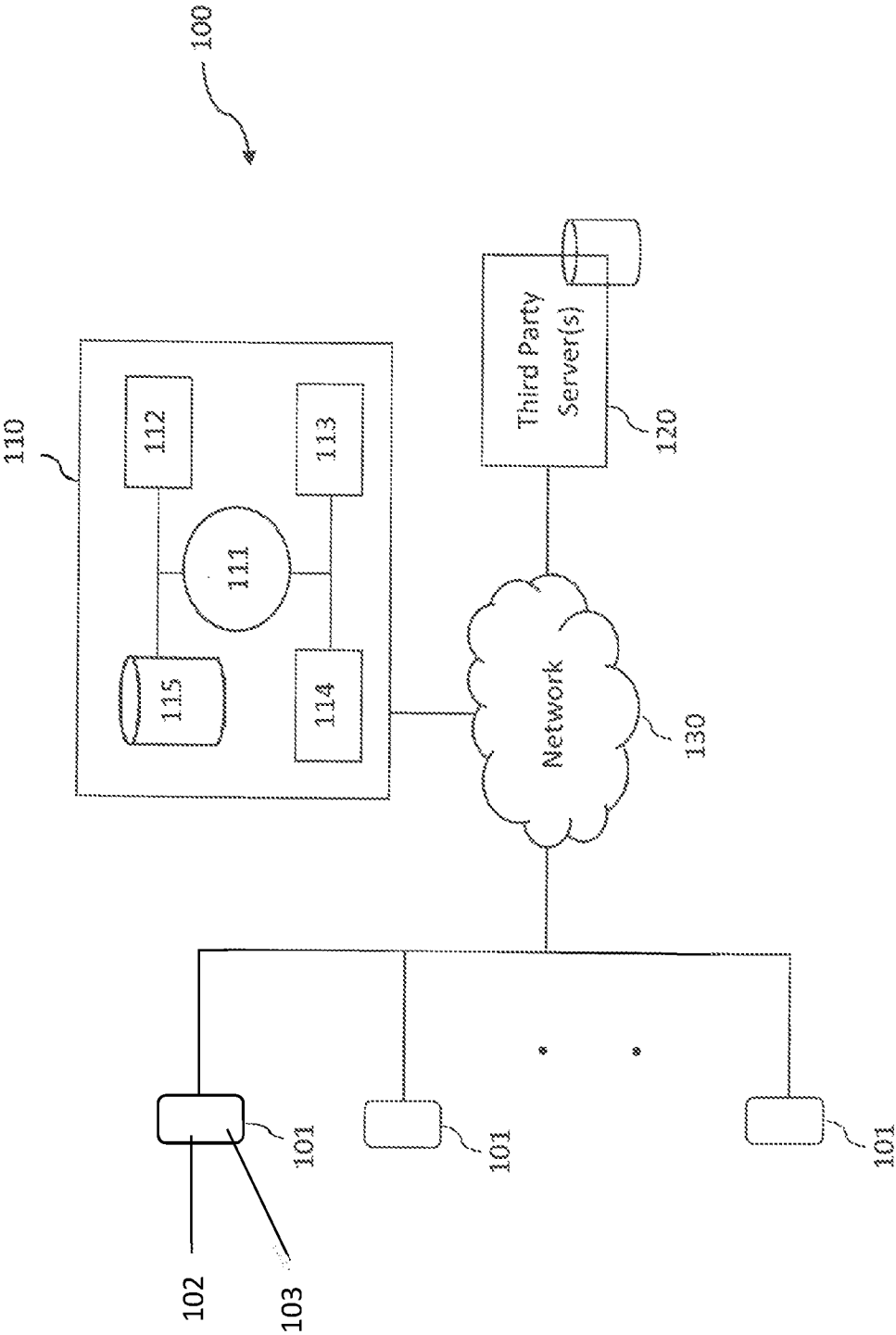
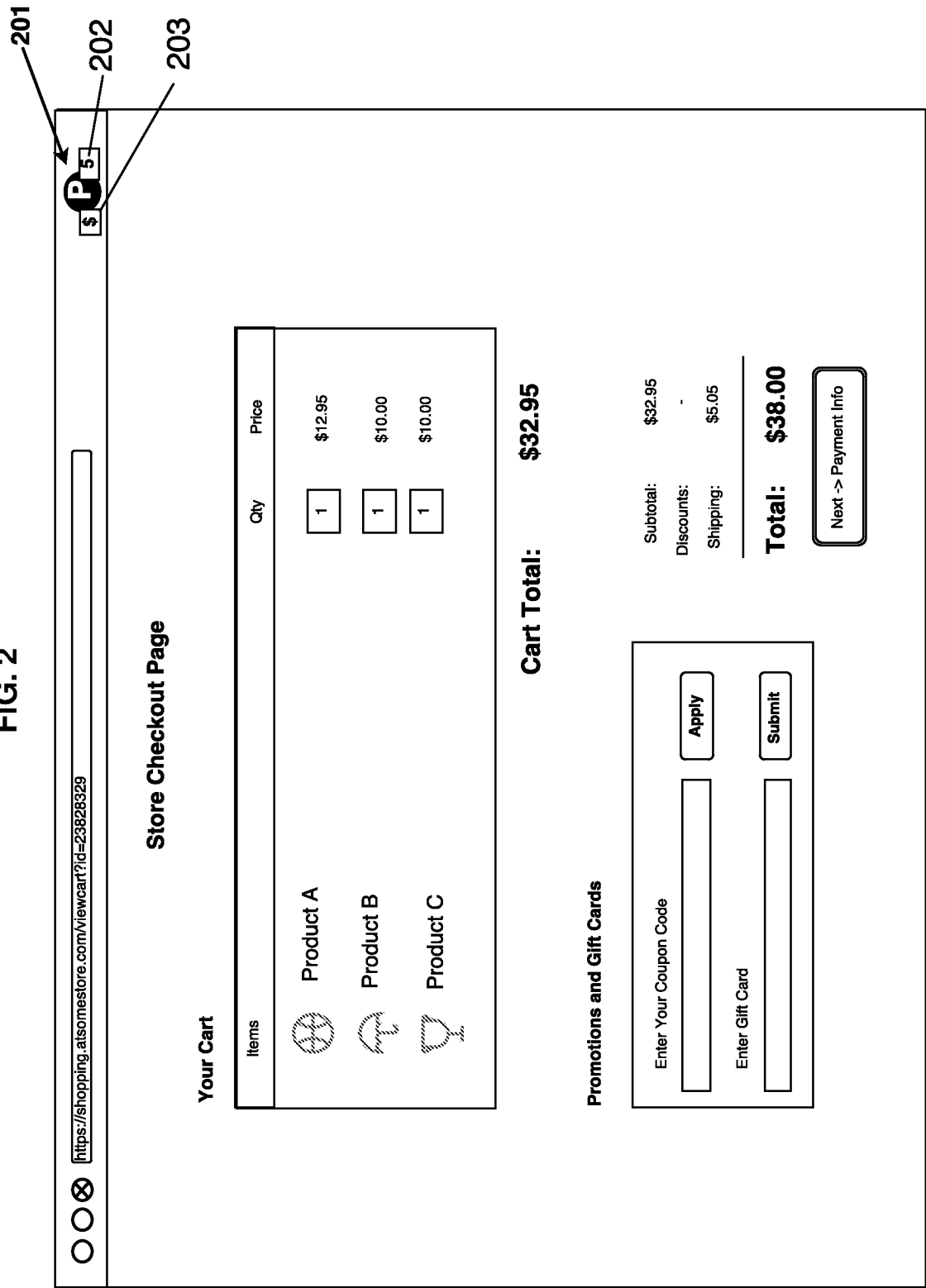


FIG. 1

FIG. 2



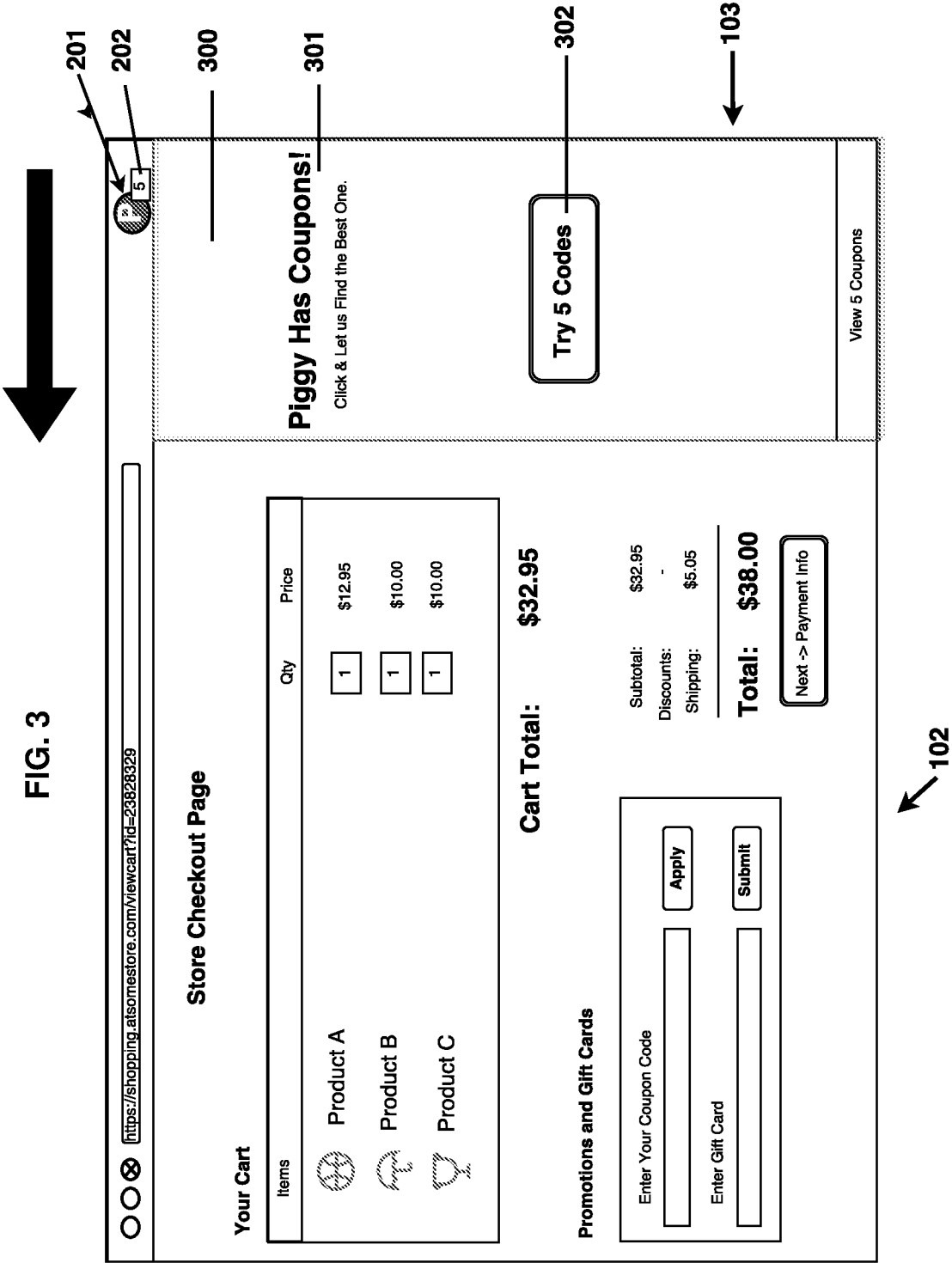


FIG. 4

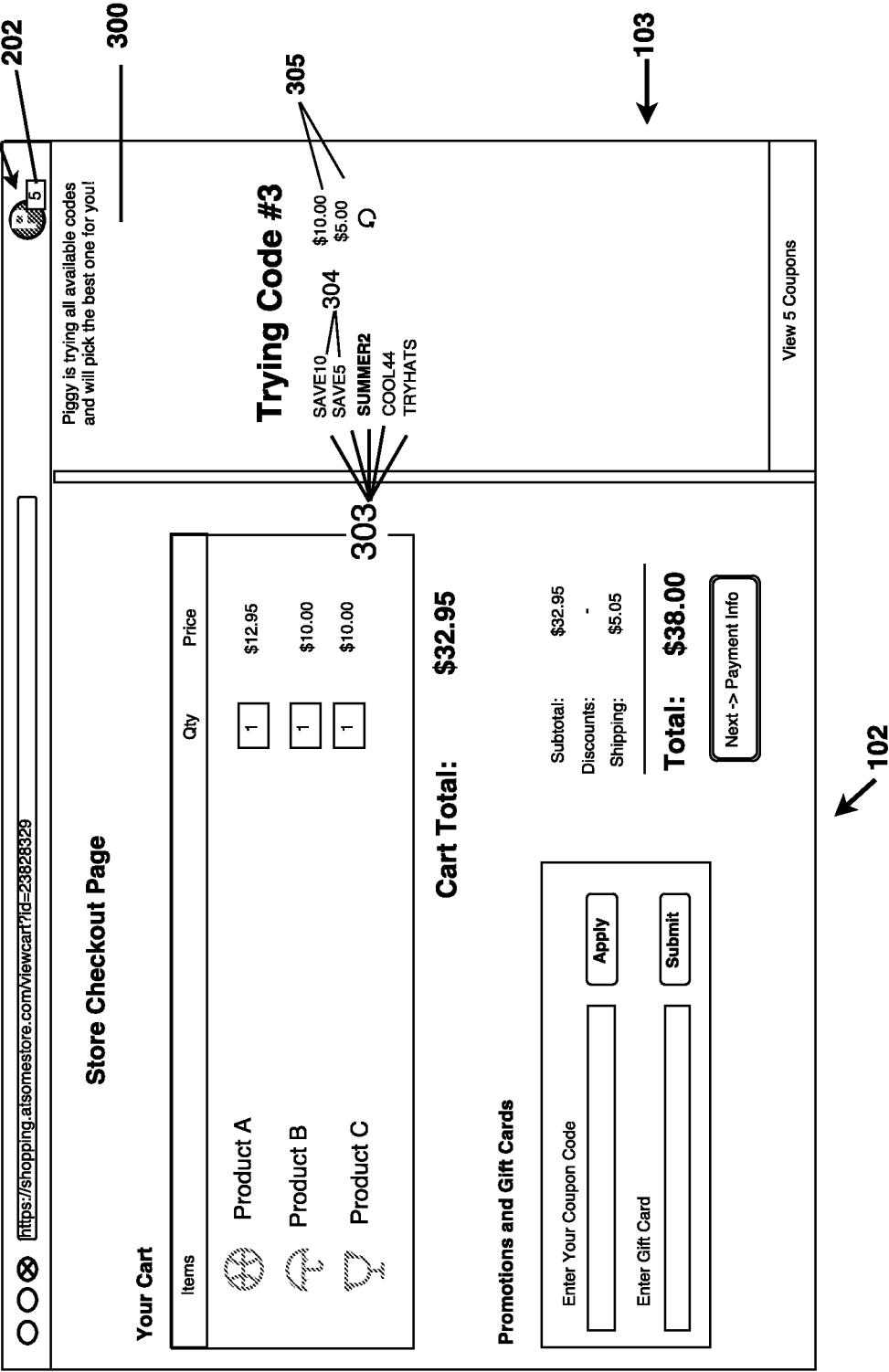


FIG. 5

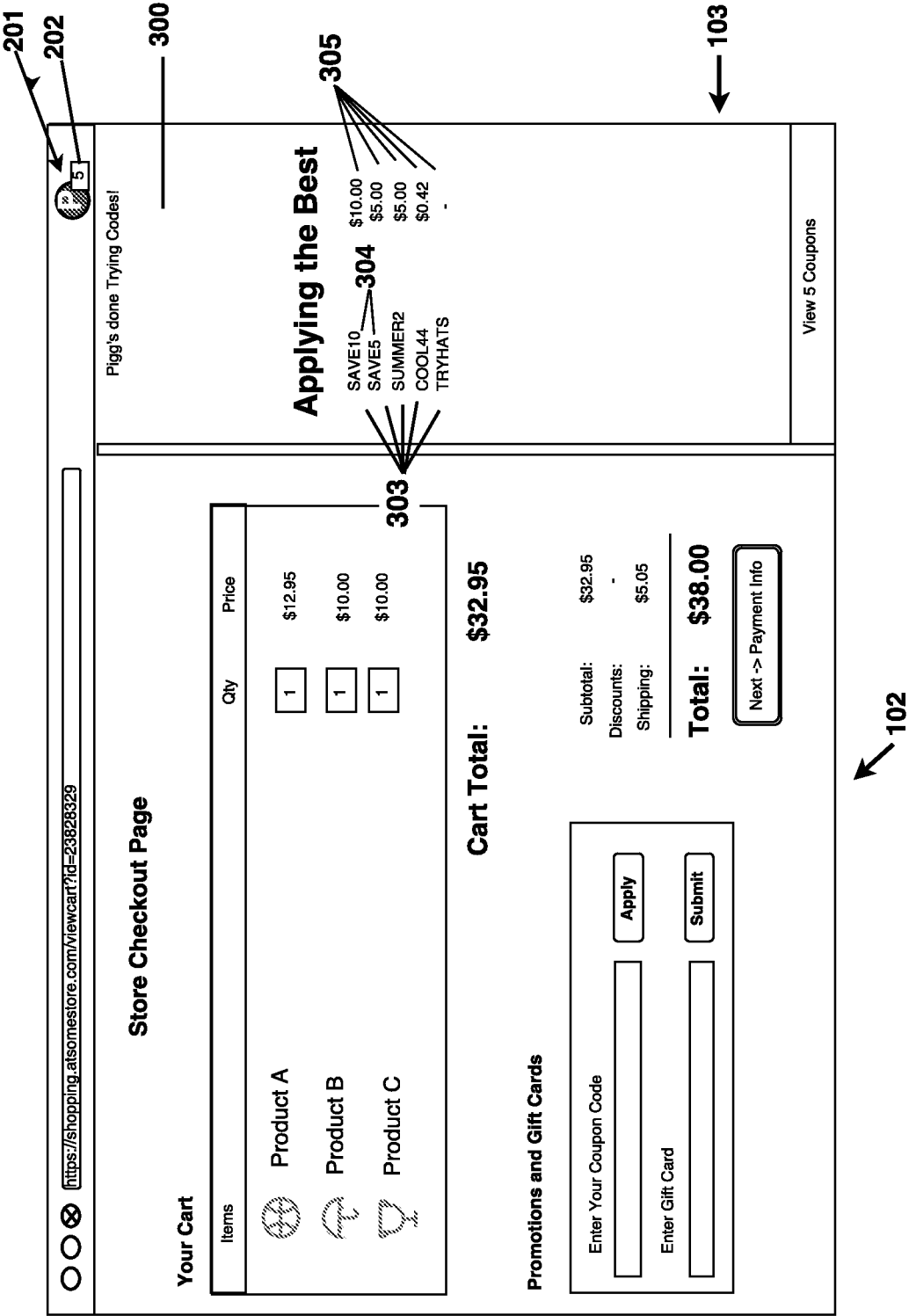


FIG. 6

https://shopping.almoststore.com/viewcart?id=23828329

Store Checkout Page

Your Cart

Items	Qty	Price
Product A	<input type="text" value="1"/>	\$12.95
Product B	<input type="text" value="1"/>	\$10.00
Product C	<input type="text" value="1"/>	\$10.00

Cart Total:

\$32.95

Promotions and Gift Cards

Enter Your Coupon Code
Code Applied: "SAVE10"
[remove](#)

Enter Gift Card

202

5

300

301

306

305

307

103

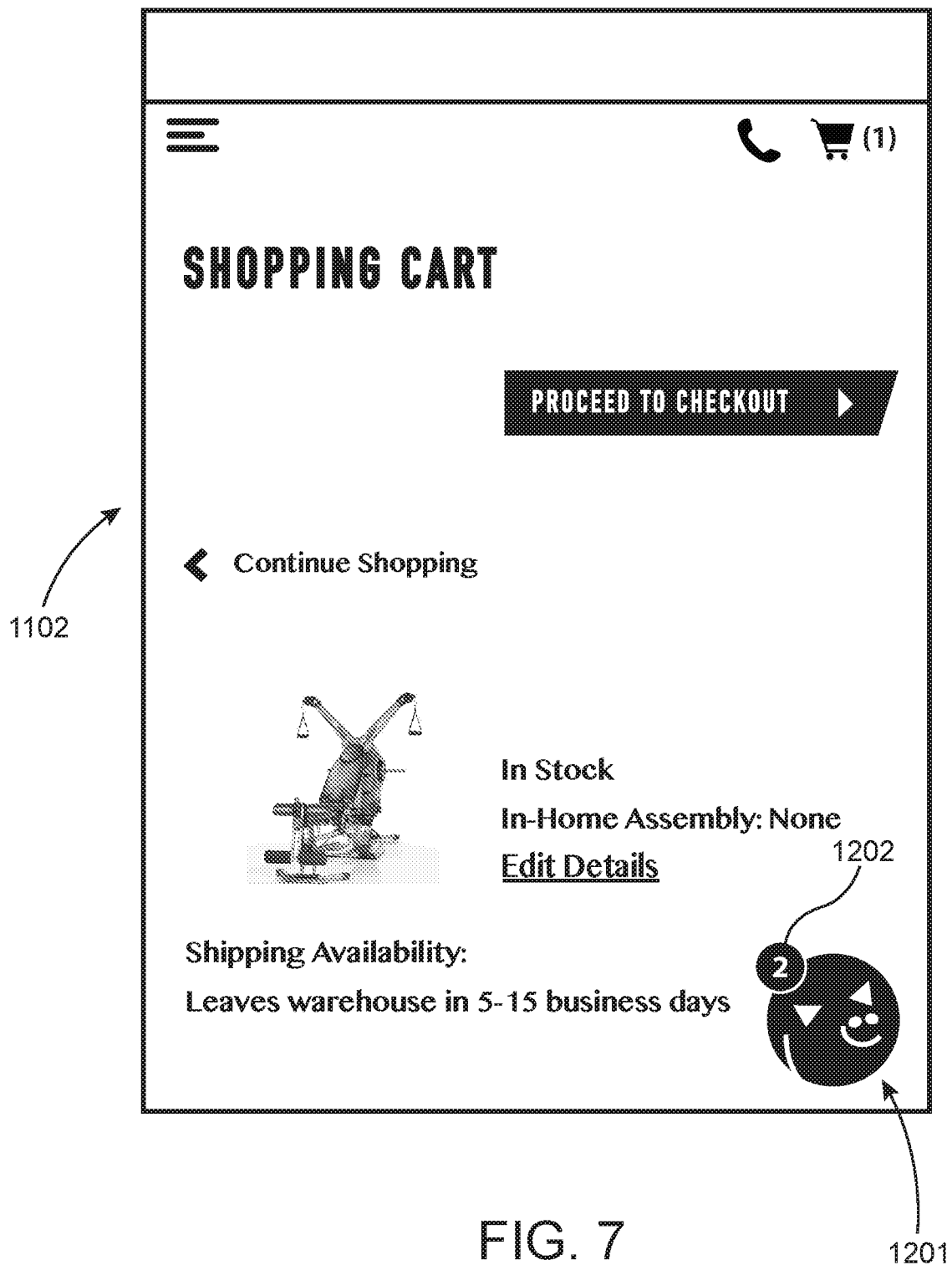
Piggy saved you \$10.00

Using these codes:
SAVE10

BACK TO CART

View 5 Coupons

Next -> Payment Info



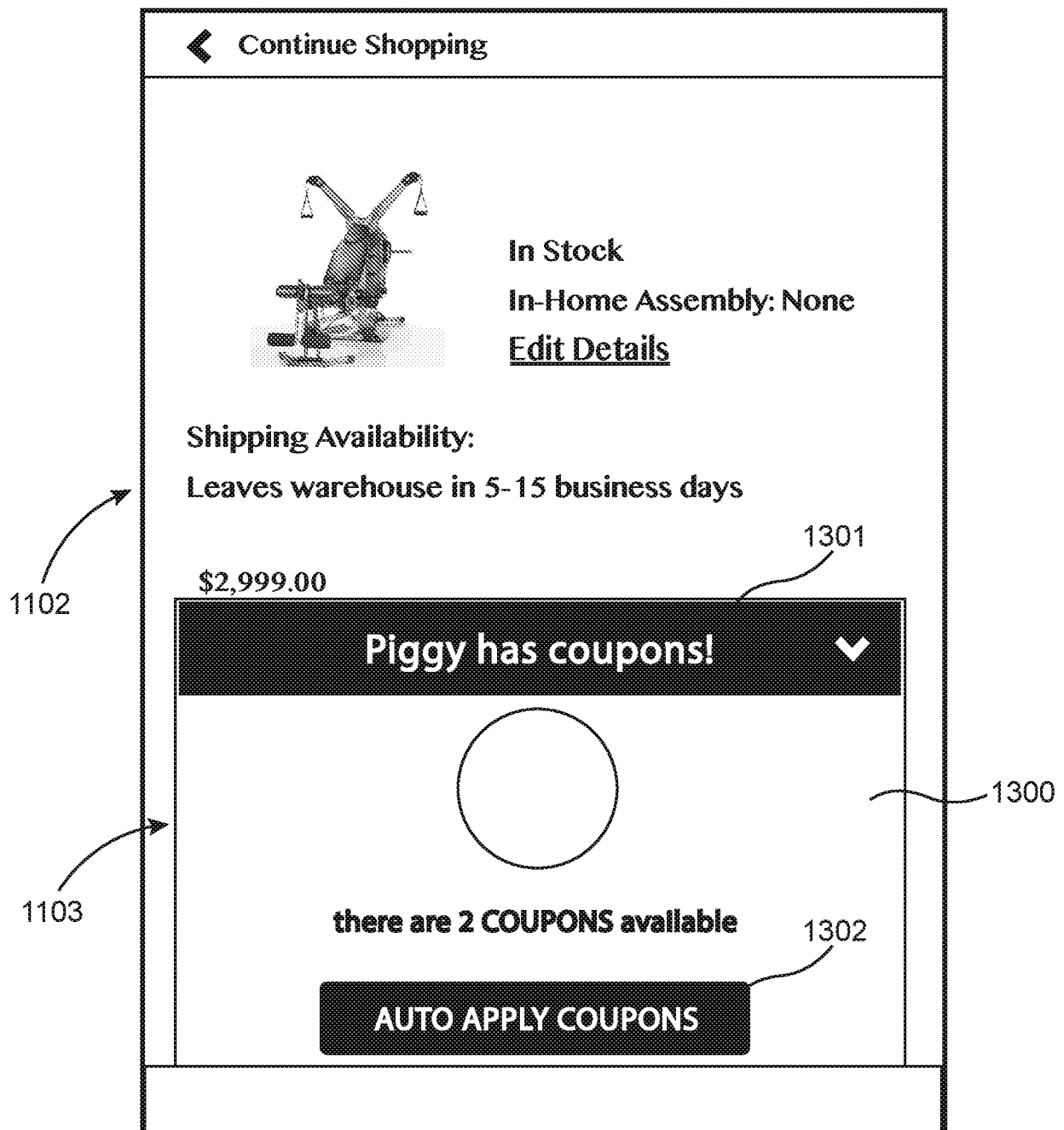


FIG. 8

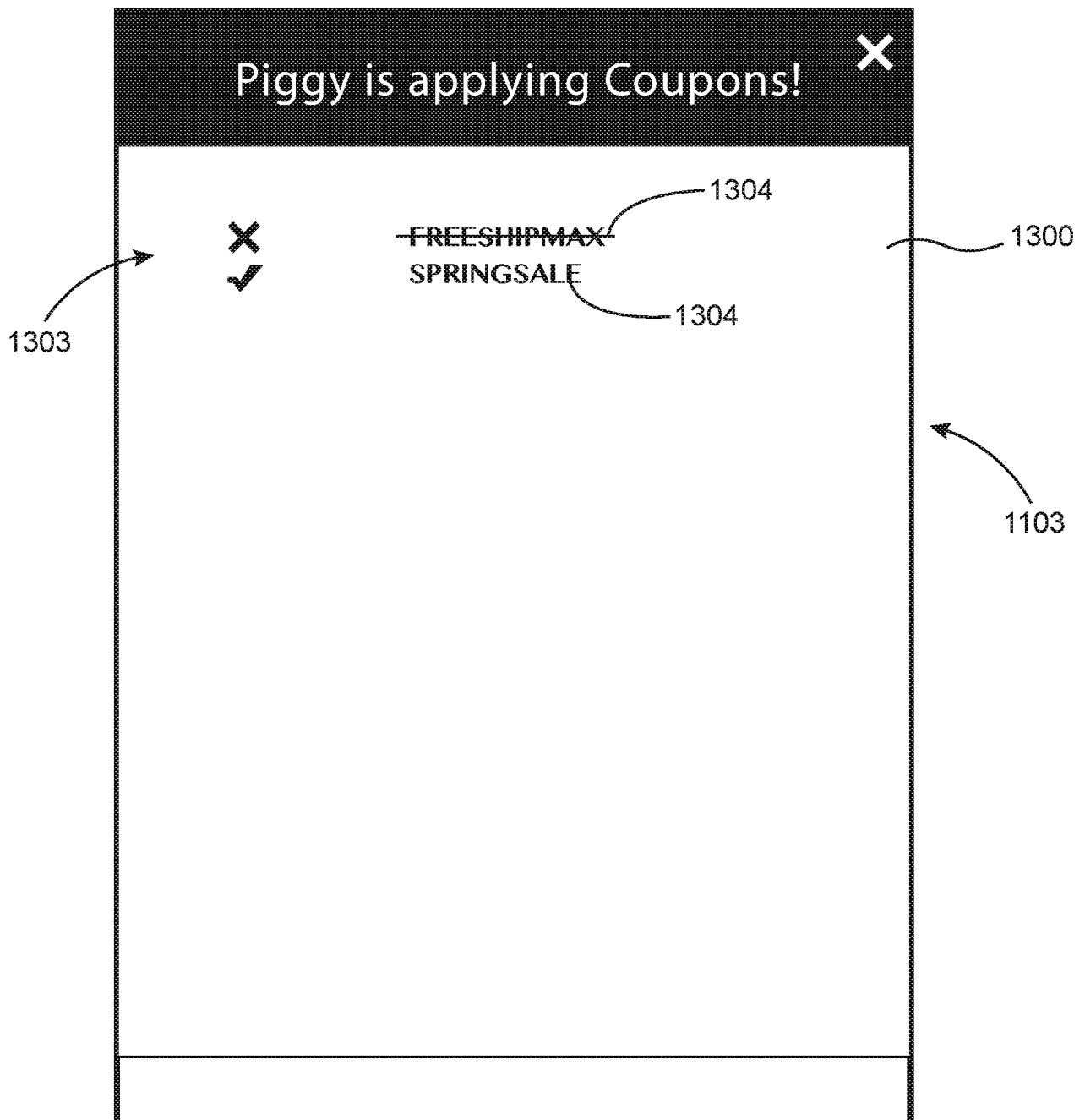


FIG. 9



FIG. 10

1102

ORDER SUMMARY

Edit Cart

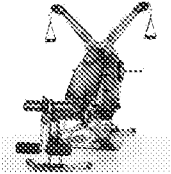
	Qty. 1	-\$2,999.00 \$2,799.00
	Qty. 1	-\$99.00 \$0.00
Subtotal		\$2,799.00
Standard Shipping		\$199.99
Shipping Discount		-\$199.99
Sales Tax		TBD
Estimated Total		2,799.00



FIG. 11

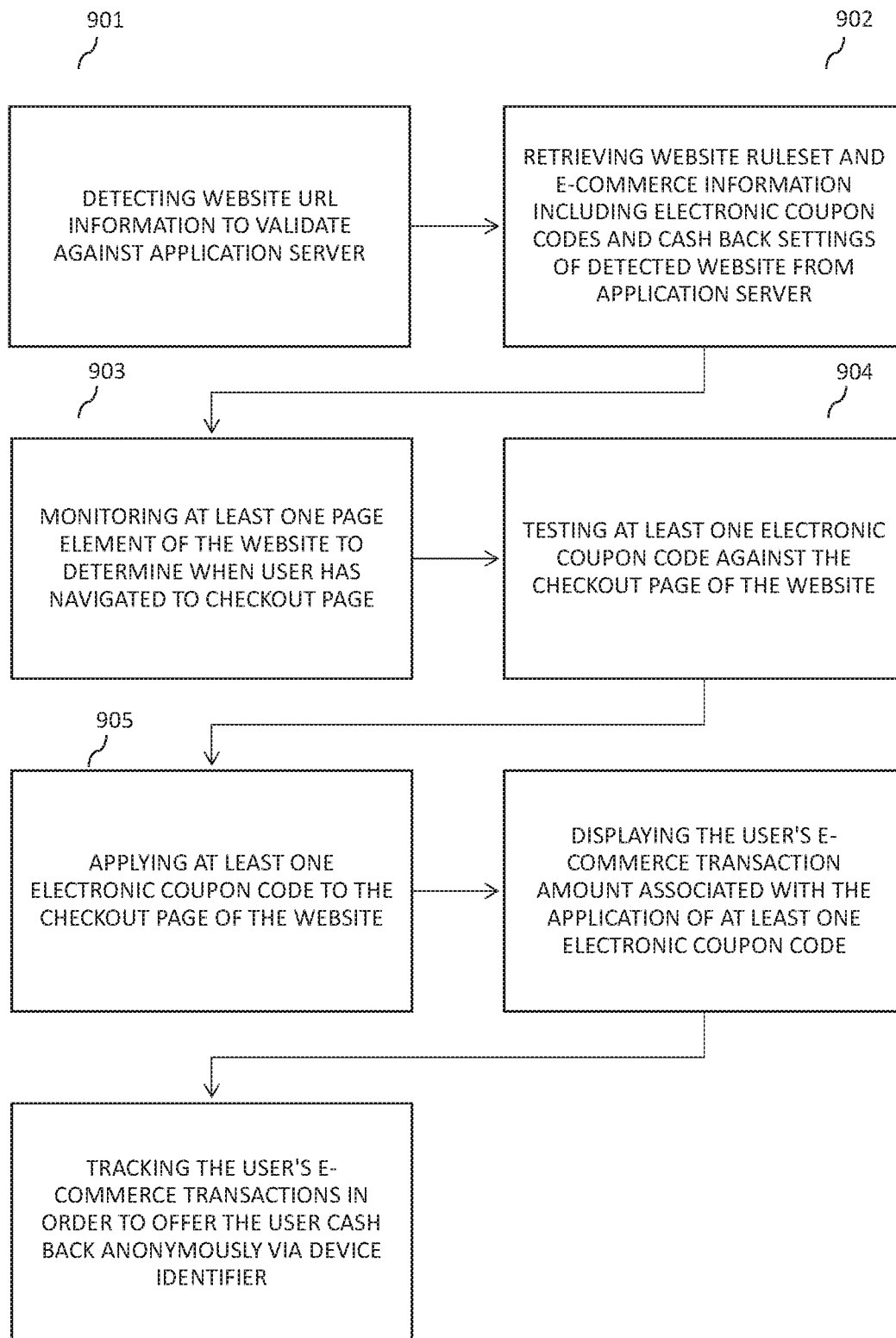
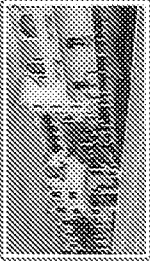


FIG. 12



FIG. 13



Best Western Plus Marina Shores Hotel (Last booked 0 hours ago)
34039 Pacific Coast Hwy, Dana Point, CA 92628 United States, 866-767-4278

☐ Dana Point
• 1.2 miles to city center
• 10 miles to Orange County, CA (USA)
John Wayne
☒ Collect nights

Very Good 4.2/5
618 Hotels.com guest reviews
○○○○○○○○
685 reviews

~~\$498~~
Let's Book
P **\$150**
Plus 8% Cash Back
nightly price per room
✓ free cancellation
✓ pay now or at hotel



Laguna Cliffs Marriott Resort and Spa (Last booked 14 hours ago)
25135 Park Lantana, Dana Point, CA 92626 United States, 866-425-4125

☐ Dana Point
• 1.1 miles to city center
• 18 miles to Orange County, CA (USA)
John Wayne
☒ Collect nights

Fabulous 4.3/5
851 Hotels.com guest reviews
○○○○○○○○
1,372 reviews

~~\$271~~
Let's Book
P **\$200**
Plus 8% Cash Back
nightly price per room
✓ free cancellation
✓ pay now or at hotel



The Ritz-Carlton, Laguna Niguel
1 Ritz Carlton Dr, Dana Point, CA 92620 United States, 866-428-0261

☐ Dana Point
• 0.7 miles to city center
• 18 miles to Orange County, CA (USA)
John Wayne
☒ Collect nights

Very Good 4.2/5
116 Hotels.com guest reviews
○○○○○○○○
2,223 reviews

~~\$852~~
Let's Book
P **\$800**
Plus 8% Cash Back
nightly price per room
✓ pay now or at hotel



The Ritz-Carlton, Laguna Niguel
1 Ritz Carlton Dr, Dana Point, CA 92620 United States, 866-428-0261

☐ Dana Point
• 0.7 miles to city center
• 18 miles to Orange County, CA (USA)
John Wayne
☒ Collect nights

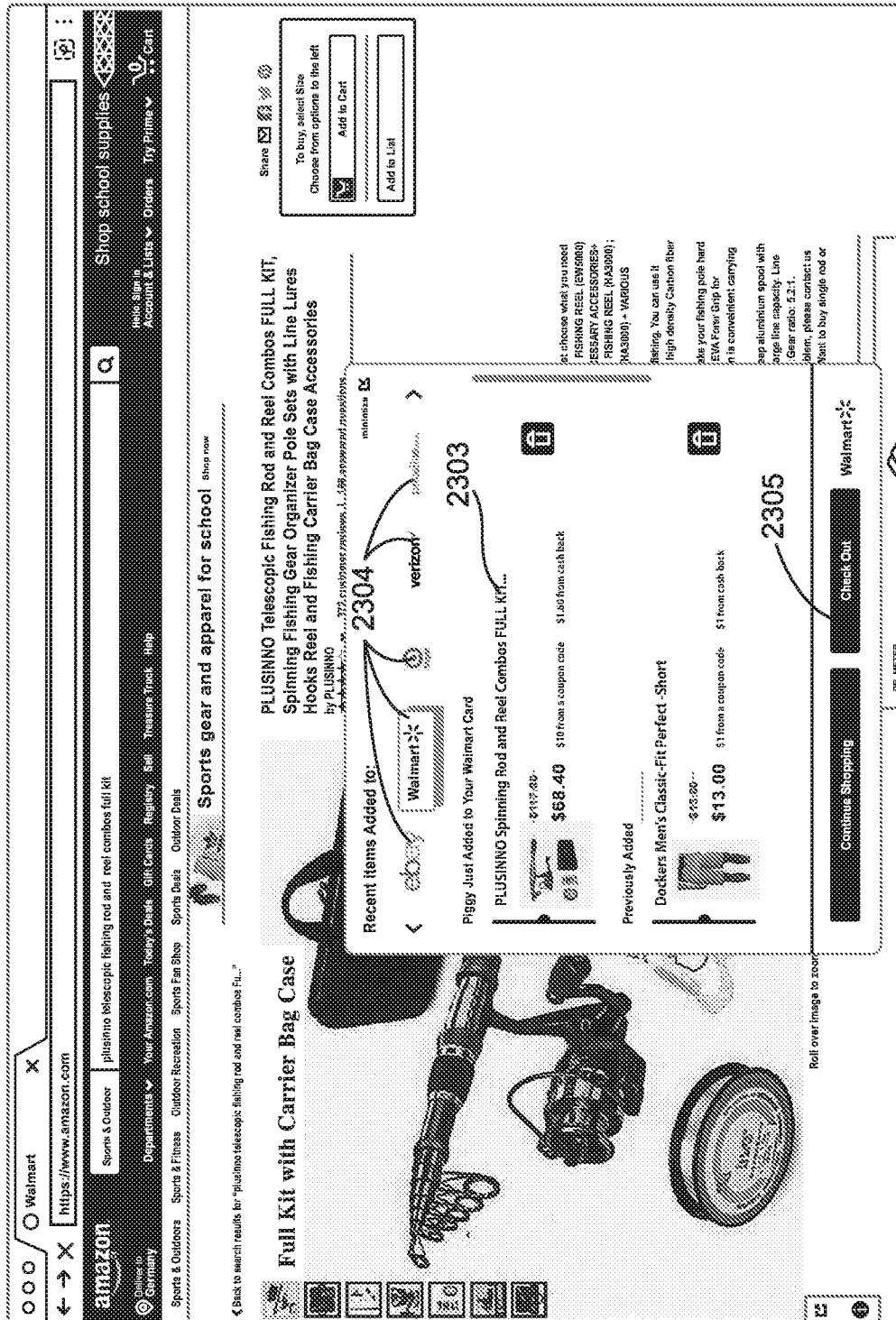
Very Good 4.2/5
116 Hotels.com guest reviews
○○○○○○○○
2,223 reviews

~~\$300~~
Let's Book
P **\$300**
Plus 8% Cash Back
nightly price per room

FIG. 14

2102

SUBSTITUTE SHEET (RULE 26)



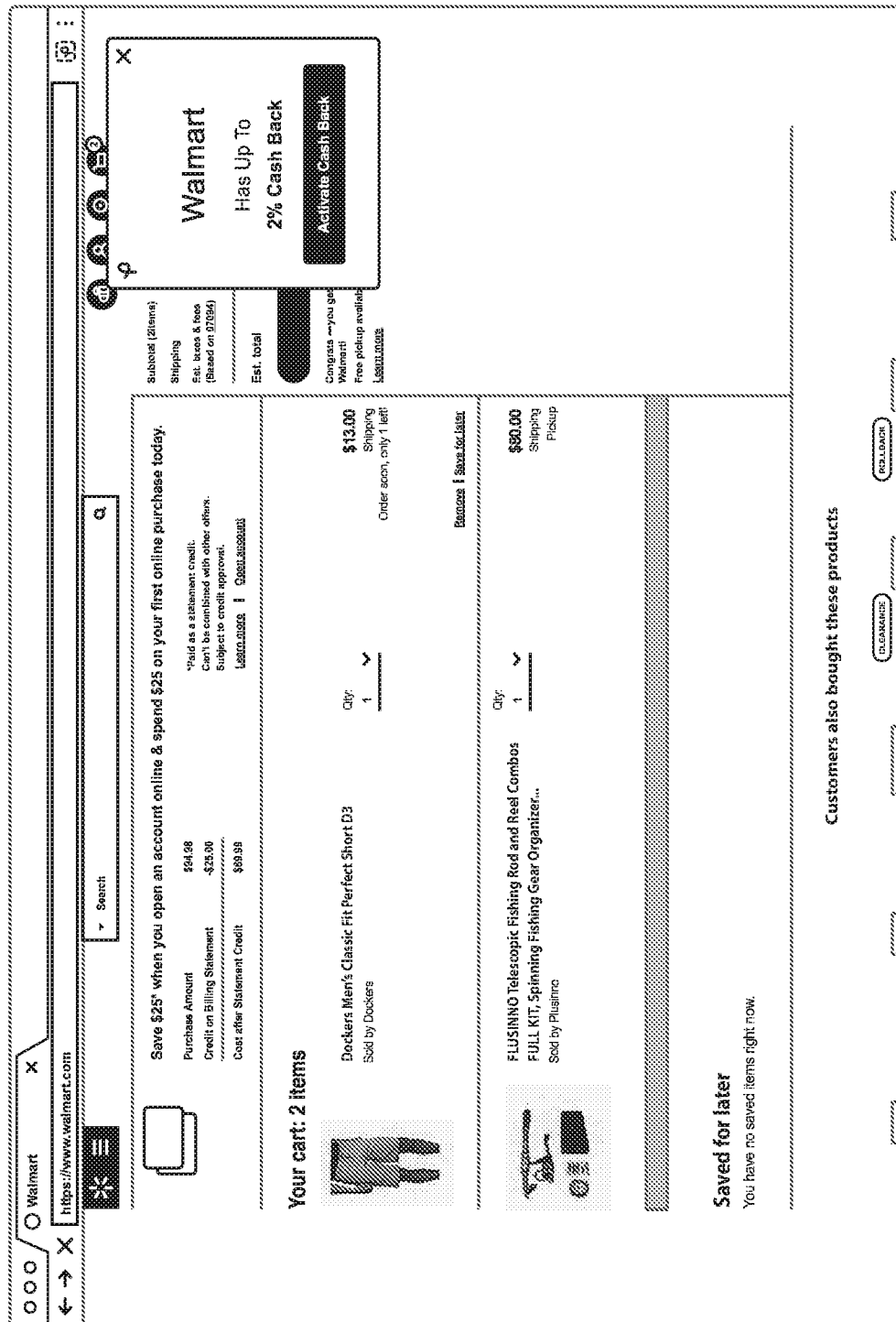


FIG. 16

2202

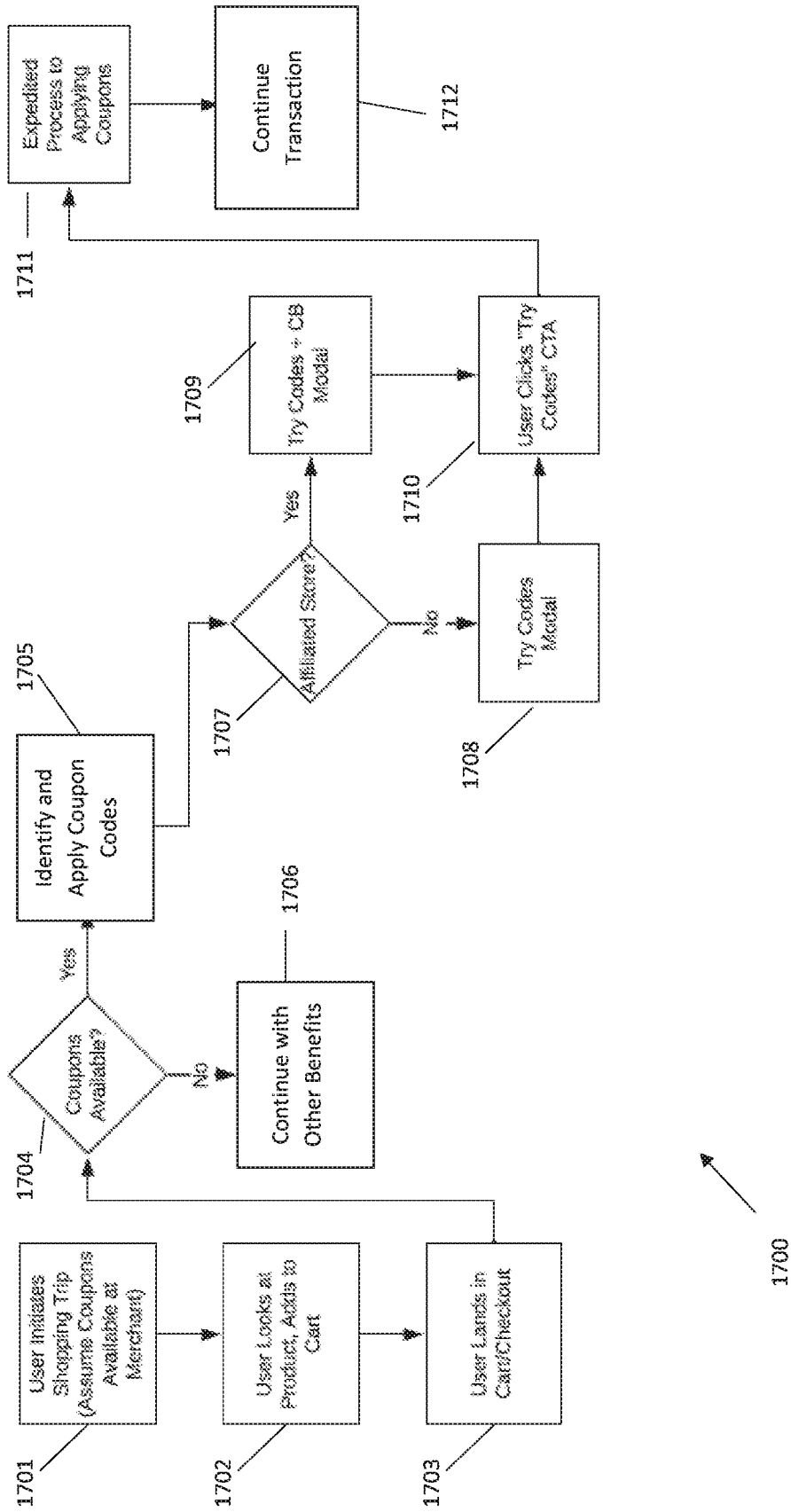


FIG. 17

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2019/051537

A. CLASSIFICATION OF SUBJECT MATTER

IPC(8) - G06Q 30/06; G06Q 30/00; G06Q 30/02 (2020.01)

CPC - G06Q 30/0629; G06Q 10/087; G06Q 30/0234; G06Q 30/0623 (2020.01)

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

See Search History document

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

USPC - 705/14.34; 705/26.1; 705/26.62; 705/26.61 (keyword delimited)

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

See Search History document

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X --- Y	US 2016/0071187 A1 (AMAZON TECHNOLOGIES INC) 10 March 2016 (10.03.2016) entire document	1-13, 17-29 --- 14-16, 38
X --- Y	US 2014/0180790 A1 (COUPONS.COM INCORPORATED) 26 June 2014 (26.06.2014) entire document	30-34, 36, 37, 39, 41-49 --- 15, 16, 35, 38, 40
Y	US 2015/0142610 A1 (GOOGLE INC) 21 May 2015 (21.05.2015) entire document	14
Y	US 2018/0019984 A1 (MONTICELLO ENTERPRISES LLC) 18 January 2018 (18.01.2018) entire document	35
Y	US 10,062,062 B1 (MCGHIE et al) 28 August 2018 (28.08.2018) entire document	40
A	US 2002/0026584 A1 (SKUBIC et al) 28 February 2002 (28.02.2002) entire document	1-49

☐ Further documents are listed in the continuation of Box C.☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

29 January 2020

Date of mailing of the international search report

17 FEB 2020

Name and mailing address of the ISA/US

Mail Stop PCT, Attn: ISA/US, Commissioner for Patents
P.O. Box 1450, Alexandria, VA 22313-1450

Facsimile No. 571-273-8300

Authorized officer

Blaine R. Copenheaver

PCT Helpdesk: 571-272-4300
PCT OSP: 571-272-7774

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2019/051537

Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

See extra sheet(s).

1. ☒ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- ☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- ☒ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2019/051537

Continued from Box No. III Observations where unity of invention is lacking

This application contains the following inventions or groups of inventions which are not so linked as to form a single general inventive concept under PCT Rule 13.1. In order for all inventions to be examined, the appropriate additional examination fees must be paid.

Group I, claims 1-29, are drawn to a method for providing e-commerce information associated with an e-commerce transaction, to users of an electronic device comprising: detecting automatically at least one item listing of a target merchant store in a target e-commerce interface.

Group II, claims 30-49, are drawn to a method for providing e-commerce information associated with an e-commerce transaction, to users of an electronic device comprising: monitoring at least one of a URL or a page element to determine whether the user has navigated to a checkout interface associated with the e-commerce transaction.

The inventions listed as Groups I-II do not relate to a single general inventive concept under PCT Rule 13.1 because, under PCT Rule 13.2, they lack the same or corresponding special technical features for the following reasons: the special technical feature of the Group I invention: detecting automatically at least one item listing of a target merchant store in a target e-commerce interface; displaying automatically the at least one item listing of at least one competing merchant store within the target e-commerce interface; prompting the user automatically to select the at least one item listing of the at least one competing merchant store within the target e-commerce interface; applying automatically the at least one item listing of the at least one competing merchant store to a competing e-commerce interface associated with the at least one competing merchant store; and, configuring at least one application server to include at least one database having e-commerce information associated with a plurality of e-commerce interfaces as claimed therein is not present in the invention of Group II. The special technical feature of the Group II invention: monitoring at least one of a URL or a page element to determine whether the user has navigated to a checkout interface associated with the e-commerce transaction; upon determining that the user has navigated to said checkout interface, and prior to prompting the user, identifying and applying at least one benefit relative to a purchase selected by the user; after checking for at least one benefit relative to the purchase selected by the user, prompting the user to request application of said at least one benefit; after receiving input from the user confirming application of said benefit, displaying a modal to the user operative to indicate that benefits are at least being identified; and displaying the result of the user's e-commerce transaction associated with the application of said at least one benefit as claimed therein is not present in the invention of Group I.

Groups I and II lack unity of invention because even though the inventions of these groups require the technical feature of a method for providing e-commerce information associated with an e-commerce transaction, to users of an electronic device, this technical feature is not a special technical feature as it does not make a contribution over the prior art.

Specifically, US 2002/0026584 to Skubic et al. teaches a method for providing e-commerce information associated with an e-commerce transaction, to users of an electronic device (Paras. [0033-0037]).

Since none of the special technical features of the Group I or II inventions are found in more than one of the inventions, unity of invention is lacking.