Description

Title Of Invention: SERVICE REQUEST ESTIMATED TIME GENERATION SYSTEM AND METHOD

Technical Field

[0001] The present invention relates generally to a system and a method for generation an estimated time, more particularly, the system and method generate an estimated time to fulfil a request for a service which requires a service provider to move from one location to another location.

Background

[0002] In food/grocery delivery industries, accurate prediction of the delivery time or estimated time of arrival (ETA) of a placed order is very important. It will affect the consumers' experiences as the ETA will be shown to the consumers and they will expect the order be delivered on time. Current ETA prediction or generation systems mainly focus on ETA accuracy and ETA promise to achieve on-time deliveries. The systems are designed to maximize the ETA accuracy and ETA promise and consider the direct impact of the ETA.

[0003] During the peak hours when there is a surge in demand, there may be not enough service providers (e.g., drivers) to fulfill the whole orders placed. One conventional way is to control the visibility of merchants by allowing only nearby merchants visible to the consumers so that the average delivery time for orders is shorter due to smaller delivery distance and in turn service providers can fulfil more orders. However, in doing so, this limits the variety of available merchants for some consumers during such times, result in some bad user experience.

[0004] Additionally, it is found that the ETA shown to the consumers may have other implicit or indirect impacts, for example impacts on the demand situations of the delivery platform, which in turns affects the actual time of arrival (ATA) as well as ETA accuracy and ETA promise themselves. For example, the consumer will tend to place the order if the ETA is within their expectations and the ETA expectation will depend on many different factors and market conditions. Generally, the longer the ETA, the consumers have less intention to checkout or place the orders, thus the demand from consumers will reduce; whereas the shorter the ETA, the consumers have more to checkout or place the orders, thus the demand

from consumers will increase. Such market and operating variables are not considered in current ETA generation systems.

[0005] There is thus a need to devise a novel method and system which take the indirect impact of ETA adjustments in shaping demand from consumers, operating variables and accuracy of ETA into consideration in generating an estimated time to fulfil a request for a service (delivery order).

[0006] Furthermore, other desirable features and characteristics will become apparent from the subsequent detailed description and the appended claims, taken in conjunction with the accompanying drawings and this background of the disclosure.

Summary

[0007] In a first aspect, the present disclosure provides a method for generating an estimated time to fulfil a request for a service, the service requiring a service provider to move from one location to another location, comprising: identifying an operating variable relating to a current request for a service and a performance metric, the performance metric comprising an accuracy metric in accurately estimating an actual time required to fulfil the previous request for the service; and generating an estimated time to fulfil the current request for the service to provide a desired number of further requests for the service based on the operating variable and the performance metric..

[0008] In a second aspect, the present disclosure provides a system for generating an estimated time to fulfil a request for a service, the service requiring a service provider to move from one location to another location, the system comprising: at least one processor; and at least one memory including computer program code; the at least one memory and the computer program code configured to, with at least one processor, cause the server at least to, comprising: identify an operating variable relating to a current request for a service and a performance metric, the performance metric comprising an accuracy metric in accurately estimating an actual time required to fulfil a previous request for the service; and generate an estimated time to fulfil the current request for the service to provide a desired number of further requests for the service based on the operating variable and the performance metric.

[0009] Additional benefits and advantages of the disclosed embodiments will become apparent from the specification and drawings. The benefits and/or advantages may be individually obtained by the various embodiments and features of the specification and

drawings, which need not all be provided in order to obtain one or more of such benefits and/or advantages.

Brief Description of the Drawings

[0010] Embodiments and implementations are provided by way of example only, and will be better understood and readily apparent to one of ordinary skill in the art from the following written description, read in conjunction with the drawings, in which:

[0011] Figure 1 shows a block diagram illustrating a system for generating an estimated time to fulfil a request for a service according to various embodiments of the present disclosure.

[0012] Figure 2 shows a block diagram illustrating various components of a system for generating an estimated time to fulfil a request for a service according to an embodiment of the present disclosure.

[0013] Figure 3 shows a flow chart illustrating a method for generating an estimated time to fulfil a request for a service according to an embodiment of the present disclosure.

[0014] Figure 4 shows a flowchart illustrating communications between a service request platform and a system for generating an estimated time to fulfil a request for a service according to an embodiment of the present disclosure.

[0015] Figure 5 shows a flowchart illustrating communications between a service request platform and a system for generating an estimated time to fulfil a request for a service according to another embodiment of the present disclosure.

[0016] Figure 6 shows a flowchart illustrating communications between a service request platform and a system for generating an estimated time to fulfil a request for a service according to yet another embodiment of the present disclosure.

[0017] Figures 7 and 8 show a schematic diagram of a general purpose computer system upon which the coordination server of Figure 1 can be practiced.

[0018] Figure 9 shows an alternative computer device to implement the coordination server of Figure 1.

[0019] Figure 10 shows a schematic diagram of a general purpose computer system upon which the estimated time generation server of Figure 1 can be practiced.

[0020] Figure 11 shows an alternative computer device to implement the estimated time generation server of Figure 1.

[0021] Figure 12 shows a schematic diagram of a general purpose computer system upon which a combined coordination and estimated time generation server of Figure 1 can be practiced.

[0022] Figure 13 shows an alternative computer device to implement a combined coordination and estimated time generation server of Figure 1.

[0023] Skilled artisans will appreciate that elements in the figures are illustrated for simplicity and clarity and have not necessarily been depicted to scale. For example, the dimensions of some of the elements in the illustrations, block diagrams or flowcharts may be exaggerated in respect to other elements to help to improve understanding of the present embodiments.

Detailed Description

[0024] The following detailed description is merely exemplary in nature and is not intended to limit the invention or the application and uses of the invention. Furthermore, there is no intention to be bound by any theory presented in the preceding background of the invention or the following detailed description. It is the intent of this invention to present a service request allocation system and method under lack of available service providers condition.

[0025] Where reference is made in any one or more of the accompanying drawings to steps and/or features, which have the same reference numerals, those steps and/or features have for the purposes of this description the same function(s) or operation(s), unless the contrary intention appears.

[0026] It is to be noted that the discussions contained in the "Background" section and that above relating to prior art arrangements relate to discussions of devices which form public knowledge through their use. Such should not be interpreted as a representation by the present inventor(s) or the patent applicant that such devices in any way form part of the common general knowledge in the art.

[0027] Some portions of the description which follows are explicitly or implicitly presented in terms of algorithms and functional or symbolic representations of operations on data within a computer memory. These algorithmic descriptions and functional or symbolic representations are the means used by those skilled in the data processing arts to convey most effectively the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities, such as electrical, magnetic or optical signals capable of being stored, transferred, combined, compared, and otherwise manipulated.

[0028] Unless specifically stated otherwise, and as apparent from the following, it will be appreciated that throughout the present specification, discussions utilizing terms such as “receiving”, “calculating”, “determining”, “updating”, “generating”, “initializing”, “outputting”, “receiving”, “retrieving”, “identifying”, “dispersing”, “authenticating” or the like, refer to the action and processes of a computer system, or similar electronic device, that manipulates and transforms data represented as physical quantities within the computer system into other data similarly represented as physical quantities within the computer system or other information storage, transmission or display devices.

[0029] The present specification also discloses apparatus for performing the operations of the methods. Such apparatus may be specially constructed for the required purposes, or may comprise a computer or other device selectively activated or reconfigured by a computer program stored in the computer. The algorithms and displays presented herein are not inherently related to any particular computer or other apparatus. Various machines may be used with programs in accordance with the teachings herein. Alternatively, the construction of more specialized apparatus to perform the required method steps may be appropriate. The structure of a computer will appear from the description below.

[0030] In addition, the present specification also implicitly discloses a computer program, in that it would be apparent to the person skilled in the art that the individual steps of the method described herein may be put into effect by computer code. The computer program is not intended to be limited to any particular programming language and implementation thereof. It will be appreciated that a variety of programming languages and coding thereof may be used to implement the teachings of the disclosure contained herein. Moreover, the computer program is not intended to be limited to any particular control flow. There are many other variants of the computer program, which can use different control flows without departing from the spirit or scope of the invention.

[0031] Furthermore, one or more of the steps of the computer program may be performed in parallel rather than sequentially. Such a computer program may be stored on any computer readable medium. The computer readable medium may include storage devices such as magnetic or optical disks, memory chips, or other storage devices suitable for interfacing with a computer. The computer readable medium may also include a hard-wired medium such as exemplified in the Internet system, or wireless medium such as exemplified in the GSM mobile telephone system. The computer program when loaded and executed on such a computer effectively results in an apparatus that implements the steps of the preferred method.

[0032] Request for service – a request for a service refers to an order for a service (e.g., ride haling service, delivery service) which is checked out, placed or purchased by a consumer and is then fulfilled by a service provider. In various embodiments, such request requires a service provider such as a driver to move from one location (e.g., pickup location) to another location (e.g., destination location) to fulfil the order.

[0033] Estimated time – an estimated time refers to a predicted or estimated time of arrival (ETA) for a service provider to arrive at a location to fulfil an order. In various embodiments described in the present disclosure, an ETA is displayed to a consumer, for example, in an application or platform, prior to placement of a service order so that the consumer can decide whether to request for such service based on the ETA, and such ETA is determined and generated based on an operating variable and a performance metric. In alternative embodiments, an ETA generation refers to a generation of an adjustment to an ETA (e.g., a previous ETA generated for a previous service request) based on an operating variable and a performance metric. By considering that the operating variable such as the intention to purchase, the expectations and the demand from the consumers can be shaped by adjusting ETAs, the present disclosure provides a method and a system which can predict and generate an (adjusted) ETA for a service request to provide a desired number of further requests for the service. In an implementation, a prediction/generation of an estimated time is referred to as an “action”.

[0034] Operating variable – an operating variable describes a changing operating condition or a state of the service market such as the intention to purchase, the demand condition of the consumers and the supply conditions of the service providers. Examples of operating variables includes a number or requests or orders for the service (e.g., demand size), a number of available service providers (e.g., available drivers), a real-time condition (e.g., a weather

condition, a traffic condition), a real time (e.g., actual time of servicing a request, actual delivery/travel time), a performance metric (see below paragraph on “performance metric”) or other metric obtained in real time, and a contextual variable indicating a certain hour of a day (e.g., peak or non-peak hour), a certain day of a week (e.g., weekday or weekend), a specific day of a year (e.g., holiday), and a region, or a combination thereof. In an implementation, an operating variable is referred to as a “state”. Such operating variable may comprise or be determined by one or more variables.

[0035] Performance metric – a performance metric relates to an evaluation and a measurement of a performance (quality) of an estimated time for a service request, and such performance and quality of the estimated time may describe how close is the generated estimated time in achieving a desired number of service requests and a desired performance metric. Such performance metric of an estimated time will be used to determine whether an adjustment to the estimated time is required and generate an adjusted estimated time for a new/further request for the service. In particular, when a new request is received or expected, a performance metric of a previously generated estimated time, for example for one or more previous requests generated within a recent time period like 10 minutes or 1 hour, can be a real-time operating variable describing the operating condition or states of the current service market in real time and used for predicting and generating an estimated time for the current request.

[0036] In an implementation, a performance metric of an estimated time (action) may be referred to as a “reward”, while a “cumulative reward” refers to a performance metric of the estimated time (one action) over multiple service request. Once an estimated time (action) is generated, the performance and quality of the estimated time over one or more service requests will be evaluated. A (cumulative) reward is generated from the evaluation, received and utilized to adjust subsequent estimated times generated for subsequent service requests.

[0037] Examples of performance metrics includes accuracy metric, conversion rate metric, allocation rate metric and fulfilment rate metric. An accuracy metric measures how an estimated time has accurately estimated an actual time required to fulfil a service request (i.e., ETA accuracy, ETA promise); a conversion rate metric measures how many service requests (e.g., delivery order for an item) were derived and converted from views by the consumers for such a service, an allocation rate metric which measures how many service requests among all service requests have been allocated to service providers, and a fulfilment rate metric which

measures how many service requests among all service requests have been fulfilled by the service providers.

[0038] Current request (for service) – a current request refers to an order for a service which is currently or going to be checked out, placed or purchased by a consumer, and, prior to the order placement or purchase, an ETA or an adjustment to an ETA is generated specifically for this current request to facilitate/affect the consumer decision to make the order placement or purchase based on an operating variable and a performance metric reflecting the operating condition/state of the current market and the performance metric of recently generated ETAs to fulfil the service.

[0039] Previous request (for service) – a previous request refers to a pervious order for a service (same service or similar service as the current request) which was checked out, placed or purchased by a consumer with reference to a previous ETA displayed to the consumer. The previous ETA was generated based on a previous operating variable and a performance metric reflecting the operating condition/state of the previous market and the performance metric of the previous ETAs at that time.

[0040] Further request (for service) – a further request refers to an order of a service (same service or similar service as the current request) that may or will be checked out, placed or purchased by a consumer with reference to a same, previously generated ETA (e.g., the ETA generated specifically for the current request) or a same, previously generated adjustment to the ETA to fulfil the service.

[0041] Desired number of further requests – a desired number of further requests reflects a desired level of demand from consumers for a particular service. In the present disclosure, such desired level of demand can be achieved by adjusting the ETA to fulfil the service. The longer the ETA, the consumers have less intention to checkout or place the orders, thus the demand (number of further requests) from consumers will reduce; whereas the shorter the ETA, the consumers have more to checkout or place the orders, thus the demand (number of further requests) from consumers will increase. This will allow the service operator to be resilient and adaptive in (i) managing the consumers' demand and expectation on the service in accordance with the changing states and conditions of the service market (operating variables), (ii) maximizing the quality of the service in terms of their allocation rate, conversion rate and fulfilment rate (performance metrics), for example by using it as an input to generate

ETAs and (iii) ultimately improving consumer's and user's experiences towards the service and service platform.

[0042] According to various embodiments of the present disclosure, the process of generation of an estimated time to fulfil a service request can implemented through a system. Figure 1 shows a block diagram illustrating a system 100 for generating an estimated time to fulfil a request for a service according to an embodiment. Further, the system 100 may enable a transaction for a good or service (e.g., ride hailing service, delivery service) including a request for a good or service between a requestor (e.g., consumer) and a provider (e.g., driver). Herein, the requestor and the provider are referred to as "users".

[0043] The system comprises a requestor device 102, a provider device 104, an acquirer server 106, a coordination server 108, an issuer server 110 and an estimated time generation server 112.

[0044] A user may be any suitable type of entity, which may include a consumer, company, corporation or governmental entity (i.e., requestor) who looking to purchase or request for a good or service via a coordination server 108, an application developer, company, corporation or governmental entity (i.e., provider) who looking to sell or provide a good or service via a coordination server 108.

[0045] A requestor device 102 is associated with a customer (or requestor) who is a party to, for example, a request for a good or service that occurs between the requestor device 102 and the provider device 104. The requestor device 102 may be a computing device such as a desktop computer, an interactive voice response (IVR) system, a smartphone, a laptop computer, a personal digital assistant computer (PDA), a mobile computer, a tablet computer, and the like.

[0046] The requestor device 102 may include user credentials (e.g., a user account) of a requestor to enable the requestor device 102 to be a party to a transaction (service request). If the requestor has a user account, the user account may also be included (i.e., stored) in the requestor device 102. For example, a mobile device (which is a requestor device 102) may have the user account of the customer stored in the mobile device.

[0047] In one example arrangement, the requestor device 102 is a computing device in the form of a watch or similar wearable and is fitted with a wireless communications interface (e.g., a NFC interface). The requestor device 102 can then electronically communicate with the

provider device 104 regarding a transaction or coordination request. The customer uses the watch or similar wearable to make a request regarding the transaction or coordination request by pressing a button on the watch or wearable.

[0048] A provider device 104 is associated with a provider who is also a party to the request for a good or service that occurs between the requestor device 102 and the provider device 104. The provider device 102 may be a computing device such as a desktop computer, an interactive voice response (IVR) system, a smartphone, a laptop computer, a personal digital assistant computer (PDA), a mobile computer, a tablet computer, and the like.

[0049] Hereinafter, the term “provider” refers to a service provider and any third party associated with providing a good or service for purchase via the provider device 104. Therefore, the user account of a provider refers to both the user account of a provider and the user account of a third party (e.g., route or service coordinator, platform operator, application developer) associated with the provider.

[0050] If the provider has a user account, details of the user account may also be included (i.e., stored) in the provider device 104. For example, a mobile device (which is a provider device 104) may have user account details (e.g., account number) of the provider stored in the mobile device.

[0051] In one example arrangement, the provider device 104 is a computing device in the form of a watch or similar wearable and is fitted with a wireless communications interface (e.g., a NFC interface). The provider device 104 can then electronically communicate with the requestor to make a request regarding the transaction or coordination request by pressing a button on the watch or wearable.

[0052] An acquirer server 106 is associated with an acquirer who may be an entity (e.g. a company or organization) which issues (e.g. establishes, manages, administers) a payment account (e.g. a financial bank account) of a merchant (e.g., service provider, platform operator). An example of an acquirer is a bank or other financial institution. As discussed above, the acquirer server 106 may include one or more computing devices that are used to establish communication with another server (e.g., the coordination server 108) by exchanging messages with and/or passing information to the other server. The acquirer server 106 forwards the payment transaction relating to a transaction or service request to the coordination server 108.

[0053] A coordination server 108 is configured to carry out processes relating to a user account by, for example, forwarding data and information associated with the transaction and service request to the other servers in the system 100, such as the estimated time generation server 112. In an example, the coordination server 108 may provide operating variable and performance metric that may be used for the estimated time generation and prediction processes by the estimated time generation 112. An image may be determined based on an outcome of the preview process e.g., an image corresponding to a drop-off point for a location based on the information.

[0054] An issuer server 110 is associated with an issuer and may include one or more computing devices that are used to perform a payment transaction. The issuer may be an entity (e.g., a company or organization) which issues (e.g., establishes, manages, administers) a transaction credential or a payment account (e.g., a financial bank account) associated with the owner of the requestor device 102. As discussed above, the issuer server 110 may include one or more computing devices that are used to establish communication with another server (e.g., the coordination server 108) by exchanging messages with and/or passing information to the other server.

[0055] The coordination server 108 may be a server that hosts software application programs for processing transaction and coordination requests, for example, purchasing of a good or service by a user. The coordination server 108 may also be configured for processing coordination requests between a requestor and a provider. The coordination server communicates with other servers (e.g., estimated time generation server 112) concerning transaction and coordination requests. The coordination server 108 may communicate with the estimated time generation server 112 to facilitate generation of estimated times associated with the transaction (service) requests. The coordination server 108 may use a variety of different protocols and procedures in order to process the transaction or coordination requests.

[0056] In an example, the coordination server 108 may receive operating variables and performance metrics associated with service requests and provide the data to the estimated time generation server 112 for use in the estimated time generation and prediction processes.

[0057] Additionally, transactions that may be performed via the coordination server include good or service purchases, credit purchases, debit transactions, fund transfers, account withdrawals, etc. Coordination servers may be configured to process transactions via cash-substitutes, which may include payment cards, letters of credit, checks, payment accounts, tokens, etc.

[0058] The coordination server 108 is usually managed by a service or platform operator that may be an entity (e.g. a company or organization) which operates to process transaction and coordination requests. The coordination server 108 may include one or more computing devices that are used for processing transaction and coordination requests.

[0059] A user account may be an account of a user who is registered at the coordination server 108. The user can be a customer, a service provider (e.g., a ride-hailing or delivery service provider), or any third parties (e.g., route or service coordinator, platform operator, application developer) who want to use the coordination server. A user who is registered to a coordination server 108 or an estimated time generation server 112 will be called a registered user. A user who is not registered to the coordination server 108 or estimated time generation server 112 will be called a non-registered user.

[0060] The coordination server 108 may also be configured to manage the registration of users. A registered user has a user account which includes details and data of the user. The registration step is called on-boarding. A user may use either the requestor device 102 or the provider device 104 to perform on-boarding to the coordination server 108.

[0061] The on-boarding process for a user is performed by the user through one of the requestor device 102 or the provider device 104. In one arrangement, the user downloads an app (which includes, or otherwise provides access to, the API to interact with the coordination server 108) to the requestor device 102 or the provider device 104. In another arrangement, the user accesses a website (which includes, or otherwise provides access to, the API to interact with the coordination server 108) on the requestor device 102 or the provider device 104. The user is then able to interact with the estimated time generation server 112. The user may be a requestor or a provider associated with the requestor device 102 or the provider device 104, respectively.

[0062] Details of the registration include, for example, name of the user, address of the user, emergency contact, next-of-kin contact, vehicle plate number, vehicle type, permissions to retrieve data and information from the requestor device 102 and/or the provider device 104. Alternatively, another mobile device may be selected instead of the requestor device 102 and/or the provider device 104 for retrieving the details/data. Once on-boarded, the user would have a user account that stores all the details/data.

[0063] It may not be necessary to have a user account at the coordination server 108 to access the functionalities of the coordination server 108. However, there may be functions

that are available only to a registered user for example the provision of means to pay after the service request is fulfilled or to pay for a service request using a credit card or an electronic wallet. The term “user” will be used to collectively refer to both registered and non-registered users. A user may interchangeably be referred to as a requestor (e.g. a person who requests for a service) or a provider (e.g. a person who receive and fulfil the service request by moving from one location and another location indicated therein).

[0064] The coordination server 108 may be configured to communicate with, or may include, a database 109 via connection 130. The connection 130 may be over a network (e.g., a local area network, a wide area network, the Internet, etc.). The database 109 stores user details/data as well as data corresponding to a transaction (or transaction data) and a service request. Examples of the transaction data include Transaction identifier (ID), Merchant (Provider) ID, Merchant Name, MCC / Industry Code, Industry Description, Merchant Country, Merchant Address, Merchant Postal Code, Merchant Contact, Aggregate Merchant ID, User ID (e.g., consumer and driver), User Name, User Contact, Payment Account ID, as well as locations, time and date relating to the transaction of goods/ services.

[0065] An estimated time generation server 112 may be a server that hosts estimated time generation software application programs for predicting and generating an estimate time to fulfil a service request. The estimated time generation 112 server may be implemented as shown in Figures 9, 11 and 13.

[0066] The database 113 stores transaction and service request data. The location and route data may be geo-tagged with a geographical coordinate (e.g. latitudinal and longitudinal coordinates) and map-matched to locate a location using location coordinates. The database 113 may be a component of the estimated time generation server 112. In an example, the database 113 may be a database managed by an external entity and the database 113 is a server that, based on a request received from a user device (such as the requestor device 102 or the provider device 104) or the coordination server 108, retrieve data (e.g. from the database 113) and transmit the data to the user device or the coordination server 108. Alternatively, a module such as an estimated time generation module may store the data instead of the database 113, wherein the estimated time generation module may be integrated as part of the estimated time generation server 112 or may be external to the estimated time generation server 112.

[0067] Use of the term ‘server’ herein can mean a single computing device or a plurality of interconnected computing devices which operate together to perform a particular function.

That is, the server may be contained within a single hardware unit or be distributed among several or many different hardware units.

[0068] The requestor device 102 is in communication with the provider device 104 via a connection 114. The connection 114 may be an ad hoc connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet). The requestor device 102 is also in communication with the estimated time generation server 112 via a connection 122. The connections 114, 122 may be a network connection (e.g., the Internet). The requestor device 102 may also be connected to a cloud that facilitates the system 100 for generating an estimated time to fulfil a request for a service. For example, the requestor device 102 can send a signal or data to the cloud directly via an ad hoc connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet).

[0069] The provider device 104 is in communication with the requestor device 102 as described above, usually via the coordination server 108. The provider device 104 is, in turn, in communication with the acquirer server 106 via a connection 116. The provider device 104 is also in communication with the estimated time generation server 112 via a connection 126. The connections 116 and 126 may be network connections (e.g., provided via the Internet). The provider device 104 may also be connected to a cloud that facilitates the system 100 for generating an estimated time to fulfil a request for a service. For example, the provider device 104 can send a signal or data to the cloud directly via an ad hoc connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet).

[0070] The acquirer server 106, in turn, is in communication with the coordination server 108 via a connection 118. The coordination server 108, in turn, is in communication with an issuer server 110 via a connection 120. The connections 118 and 120 may be over a network (e.g., the Internet).

[0071] The coordination server 108 is further in communication with the estimated time generation server 112 via a connection 124. The connection 124 may be over a network (e.g., a local area network, a wide area network, the Internet, etc.). In one arrangement, the coordination server 108 and the estimated time generation server 112 are combined and the connection 124 may be an interconnected bus.

[0072] The estimated time generation server 112, in turn, is in communication with a database 113 via a connection 128. The connection 128 may be a network connection (e.g., provided via the Internet). The estimated time generation server 112 may also be connected to a cloud

that facilitates the system 100 for generating an estimated time to fulfil a service request. For example, the estimated time generation server 112 can send a signal or data to the cloud directly via a wireless ad hoc connection (e.g., via NFC communication, Bluetooth, etc.) or over a network (e.g., the Internet).

[0073] In the illustrative embodiment, each of the devices 102, 104, and the servers 106, 108, 110, 112 provides an interface to enable communication with other connected devices 102, 104 and/or servers 106, 108, 110, 112. Such communication is facilitated by an application programming interface ("API"). Such APIs may be part of a user interface that may include graphical user interfaces (GUIs), Web-based interfaces, programmatic interfaces such as application programming interfaces (APIs) and/or sets of remote procedure calls (RPCs) corresponding to interface elements, messaging interfaces in which the interface elements correspond to messages of a communication protocol, and/or suitable combinations thereof. Examples of APIs include the REST API, and the like. For example, it is possible for at least one of the requestor device 102 and the provider device 104 to receive or submit a request for a service in response to an enquiry shown on the GUI running on the respective API.

[0074] Figure 2 shows a block diagram illustrating various components of a system 200 for generating an estimated time to fulfil a request for a service according to an embodiment of the present disclosure. The system 200 ma comprise a Variable And Metric Identification Module 202, an Estimated Time Generation Module 204, a Number Of Request Determination Module 206, a Weightage Application And Adjustment Module 208.

[0075] As shown in the exemplified method for generating an estimated time to fulfil a request for a service in Figure 3, the system 200, when in operation, is configured to perform the following steps:

- step 302: identifying an operating variable relating to a current request for a service and a performance metric, the performance metric comprising an accuracy metric in accurately estimating an actual time required to fulfil the previous request for the service; and

- step 304: generating an estimated time (ETA) to fulfil the current request for the service to provide a desired number of further requests for the service based on the operating variable and the performance metric.

[0076] In step 302, the Variable And Metric Identification Module 202 is configured to identify an operating variable relating to a current service request and a performance metric. The performance metric which the Variable And Metric Identification Module 202 identifies comprises an accuracy metric in accurately estimating an actual time required to fulfil a previous request for the service. The performance metric may further comprise a conversion rate measure which measures a ratio of a number of previous requests for the service to a number of views for the service based on an initial ETA. The performance metric may further comprise a fulfilment rate measure which measures a number of previous requests for the service fulfilled by one or more service providers which are derived from one or more pervious requests.

[0077] The operating variable comprises a number of current service requests (e.g., level of demand for the service), a number of available service providers to fulfil the service (e.g., number of available drivers), a real-time condition (e.g., a weather condition, a traffic condition), a real time (e.g., actual time of servicing a request, actual delivery/travel time), a region, a period of a day (e.g., peak hour, non-peak hour), a day of a week (e.g., weekday, weekend), a specific day of a year (e.g., school holiday, public holiday) and an allocation rate measure which measures a ratio of a number of previous requests for the service assigned to and to be fulfilled by one or more service providers to a number of previous requests.

[0078] In step 304, The Estimated Time Generation Module 204 generates an ETA to fulfil the current service request to provide a desired number of further service requests based on the operating variable and performance metric identified in step 302. In one implementation, the Estimated Time Generation Module 204 may be configured to generate an initial ETA and then further generate an adjustment to the initial ETA based on the operating variable and performance metric identified in step 302 and apply the adjustment to the ETA to generate the ETA. In such implementation, the accuracy metric is a measure relating to a time difference between the actual time required to fulfil a previous request and the initial ETA to fulfil the current request. Such initial ETA generated by the Estimated Time Generation Module 204 may be based on at least one of the performance metric, the actual time required to fulfil the previous service request and a previous ETA generated for the previous service request. Such previous ETA is generated by the Estimated Time Generation Module 204 for the previous service request based on a previous operating variable relating to the previous service request and a previous performance metric at that time.

[0079] In one alternative embodiment, prior to generating the ETA, the Number Of Requests Determination Module 206 is configured to determine whether the desired number of further

service requests or a desired performance metric can be provided by one or more previous ETAs generated for one or more previous service requests based on the operating variable and performance metric identified in step 302, and generate a result of the determination. The Estimated Time Generation Module 204 will then generate the ETA to fulfil the current service request to provide the desired number of further service requests further based on such result of the determination generated by the Number Of Requests Determination Module 206.

[0080] Alternatively and additionally, the Weightage Application And Adjustment Module 208 is configured to apply and maintain a weightage to the operating variable (if it comprises one or more variables, a weightage to each of them) and a weightage to the performance metric (if it comprises more than one metric, a weightage to each of them). In other words, each of the operating variable and performance metric is weighted differently such that it affects the ETA generation process differently. Additionally, based on the result of the determination generated by the Number Of Requests Determination Module 206, the Weightage And Adjustment Module 208 is further configured to adjust the weightage to the operating variable (if it comprises one or more variables, the weightage to each of them) and the weightage to the performance metric (if it comprises more than one metric, the weightage to each of them).

[0081] Alternatively and additionally, the Variable And Metric Identification Module 202 is configured to identify a performance metric relating to each of the one or more previous estimated times of the one or more previous requests, and the Weightage Application And Adjustment Module 208 is configured to applying a weightage to the each of the one or more previous estimated time according to a time sequence, wherein the generation of the estimated time to fulfil the current request for the service to provide the desired number of further requests for the service is further based on the respective weighted performance metrics of the one or more previous estimated times.

[0082] The Estimated Time Generation Module 204 may be configured to generate the ETA to fulfil the current service request based on the (adjusted) weighted operating variable and performance metric and/or based on the weighted performance metrics of the one or more previous estimated times. Additionally, based on the result of the determination generated by the Number Of Requests Determination Module 206, the Weightage And Adjustment Module 208 is further configured to adjust the weightage to the operating variable (if it comprises one or more variables, the weightage to each of them), the weightage to the performance metric (if it comprises more than one metric, the weightage to each of them) and/or the weightage to the each of the one or more previous estimated time, before they are used to generate the ETA by the Estimated Time Generation Module 204.

[0083] In one implementation, after the ETA is generated by the Estimated Time Generation Module 204, the Variable And Metric Identification Module 202 or another module (not shown) is further configured to evaluate and measure the performance of the ETA generated by the Estimated Time Generation Module 204, for example, whether the desired number of further service requests or the desired performance metric is achieved by the ETA, and generate a new performance metric of the ETA. The new performance metric and/or the ETA will then be used to generate estimated times for subsequent service requests.

[0084] As mentioned earlier, current existing ETA prediction systems mainly rely on accuracy metric, such as the ETA accuracy and ETA promise, to generate new ETAs that are more accurate so as to achieve on-time delivery and improve user experience on the service and platform.

[0085] Where for a specific allowable time gap T minutes, the accuracy metrics are defined as the percentage of total orders (accurate for 80%-90% of total orders). Namely, the ETA accuracy is defined according to the time between the generated ETA and ATA (actual time of arrival) being within the threshold, e.g., within T minutes (shown in Equation (1)) or within a percentage of the ETA (not shown). The ETA promise is defined according to the time gap between the ATA and the generated ETA being less than T minutes ($ATA < ETA$), i.e., the order arrives on time and before the generated ETA (shown in Equation (2)).

Equation (1):

$$|ATA - ETA| <= T$$

Equation (2):

$$ATA - ETA <= T$$

[0086] The current ETA prediction systems are designed to maximize the ETA accuracy and Promise, where they only consider the direct impact of ETA. However, the ETA shown to the consumer could have some other implicit impacts. ETA will also impact the demand-supply situations of the delivery platform. For example, the consumer will tend to check out the order with ETA within their expectations and the ETA expectation will depend on many different factors. Generally, the relationship between ETA and demand is as follows, where conversion rate (CVR) refers to a measure of the number of checkout orders for a good or service derived from the number of views for the good or service:

- Longer ETA → Less intention to checkout → Reducing demands from consumers → Reducing CVR
- Shorter ETA →More intention to checkout → Increasing demands from consumers → Increasing CVR

[0087] Based on the foregoing, it is noted that the demand from consumers can be shaped by adjusting the ETAs. For example, during the peak hours when there would be a surge in demand, there will possibly not be enough drivers to fulfill the whole orders placed.

[0088] Conventionally, a simple way is to control the visibility of merchants by allowing only nearby merchants visible to the consumers. Thus, the average delivery time for orders is smaller (due to smaller delivery distance) and in turn drivers can deliver more orders. This, however, will cause some merchants inaccessible by some consumers and result in some bad experience for such consumers.

[0089] The present disclosure seeks to utilize reinforcement learning, which is a process of training machine learning models to make decisions, to provide reinforcement ETA prediction/generation system and method. The reinforcement ETA prediction/generation system and method would take the ETA's indirect impact into consideration on demand supply shaping, combine ETA accuracy/promise, other business metrics and market situations/conditions in predicting/generating ETAs and apply the predicted ETAs to shape the demand accordingly with an aim to improve user experience towards the service and service platform, the conversion rate and fulfilment rate (measure of a number of completed orders over the number of all orders) as well as to reduce the number of orders unfulfilled due to the market conditions and situations (e.g., due to lack of drivers or long expected delivery time). In other words, such system and method help to impact the demand for a service from consumers, for example, according to the driver supply situations and other changing market conditions and at the same time put on-time delivery promise and expectations from the consumers under control.

[0090] Figure 4 shows a block diagram 400 illustrating a framework, communications between a service request platform and a system and a procedural flow for generating an estimated time to fulfil a request for a service according to an embodiment of the present disclosure. In this framework, the Environment may be referred to as the service request platform (e.g., delivery platform) where transactions of service requests can be initiated, made and completed; while the Agent may be referred to as the system where multiple ETA prediction models and reinforcement learning are applied for generating an estimated time to fulfil a

request for a service displayed to users in the Environment. The Environment can be different in different use cases, while the Agent is configured to make decision and take an action (e.g., generate ETA for a service request) according to policy network or function in order to maximize the cumulative reward (performance metric). The action the Agent takes depends on the state of the Environment (operating variable) which will in turn also change the state of the Environment. The Environment will then evaluate the quality (performance metric) of the action and give a reward value for this action. Such reward will then be input to the Agent for take further actions (e.g., to generate new/adjusted ETAs for further service requests).

[0091] Figure 5 shows a block diagram 500 illustrating another framework communications between a service request platform and a system and a procedural flow for generating an estimated time to fulfil a request for a service according to an embodiment of the present disclosure. This framework is similar to that illustrated in Figure 4 except that the Agent is subdivided into a Policy Network (function) and a Value Network (function) where an Actor-Critic method is applied to train the ETA prediction models used for generating an estimated time to fulfil a request for a service displayed to users in the Environment, where the Policy Network acts as the Actor and the Value Network acts as the Critic in the Actor-Critic method. The Actor makes the decision what Action (e.g., ETA) should be taken and it is controlled by a policy function. The Action (e.g., ETA) also depends on the state of the Environment. The Environment will then evaluate the quality (performance metric) of the Action and give a reward value for this Action. Such reward will then be input to the Critic (Value Network). The Critic with a value function will further evaluate the reward the quality of the Action and input feedback to the Actor to adjust its future Actions.

[0092] In particular, the reward (performance metric) as well as the state (operating variable) may be input and utilized by the Agent to update its Value Network (e.g., weightage applied to the operating variable and performance metric) which in turn cause the policy function to take different Actions, for example even under the same state of the Environment. The interaction between the Environment and the Agent will keep going and the Policy Network and Value Network of the Agent will also evolve.

[0093] For different applications, the definition of Action, State, Reward, Weightage and Return may be different. Table 1 illustrates various parameters used by Reinforcement Learning for ETA prediction according to various embodiments of the present disclosure.

Concept	Description	Variables
Action $a(t)$	The output of Agent (or policy network of Agent): the ETAs will be shown to consumers	Predicted ETA
State $s(t)$	The state of the delivery market is determined by multiple variables: which is the statistical calculator within a specific time window at a market.	demand size, number of available drivers, weather conditions, actual delivery time, real time ETA promise & accuracy, real-time allocated rate, real-time conversion rate, hour of the day day of the week holiday region
Reward $r(a(t), s(t))$	The reward of the action: to evaluate how good the action is	ETA Accuracy, ETA Promise, Fulfilment Rate, Conversion Rate
Weightage a, w	the weightage in the objective applied to Reward(s) according to a time sequence in which the Reward (or Action) is generated or the Type of Reward (or State) which is adjusted according to desired targets and performance metrics	Weightage or multiplier to boost or discount the contribution of a state or a reward
Return $Q(a(t), s(t))$	A cumulative sum of future rewards which considers the long term impact of the action. The value function is applied to evaluate the return.	Sum of weighted current & future reward

Table 1

[0094] A Policy Network is controlled by a policy function (or multiple policy functions) $\pi\,(a(t)/\,s(t),\theta)$, which can be a machine learning model with its parameters as θ. Each policy function will take or predict an Action $a(t)$ (e.g., generate a new ETA) given the current State $s(t)$ of the delivery market. The Environment may generate a new state $s(t+1)$and a Reward $r(t)$ for the action. The Reward $r(t)$ is a function of the Action $a(t)$, i.e., the ETA output of the Agent (or the Policy Network of the Agent) and the State $s(t)$, i.e., the state of the Environment, as illustrated in Equation (3) below.

Equation (3):

$$r(t) \;=\; f((a(t), s(t))$$

[0095] The Reward may be a combination of multiple performance metrics: ETA promise; Fulfillment Rate (FR), Conversion Rate (CVR), each metrics having an independent weightage (e.g., w_1, w_2, w_3, w_4) applied to it, as illustrated in Equation (4) below.

Equation (4):

$$w_1 \times ETA\ Accuracy + w_2 \times ETA\ Promise + w_3 \times FR + w_4 \times CVR$$

[0096] The Critic acting as a judge is to evaluate the Reward generated by the Environment and the quality of the Action $a(t)$ generated by the Actor over a time period (over multiple service requests). A value network is a (multiple) value function(s) $Q(a(t), s(t), \boldsymbol{\beta})$ with parameters $\boldsymbol{\beta}$. The value function will predict the Return of the Action $Q(t)$.

[0097] The Return of the Action $Q(t)$ is the cumulative sum (or discounted sum if a <1) of its current and future Rewards (over multiple service requests), e.g., $r(t), r(t+1), r(t+2)$, which can be calculated using Equation (5), where a is a weightage applied to each of the Rewards. For example, if weightage a is larger than 1 (a >1), future or more recent reward will have a greater weightage and influence in the return value of the action. The objective of reinforcement learning is to find an optimal policy function that is able to generate Actions which can have the maximum return.

Equation (5)

$$Q(a(t), s(t), \boldsymbol{\beta}) \;=\; r(t) \;+\; a\, r(t+1) \;+\; a^2\, r(t+2) + \ldots$$

[0098] By continuously interacting with the service market, collecting and processing data accordingly, the ETA prediction model (Policy Network) of the Agent can be trained and improved through the following steps:

i. Under the current State $s(t)$ based on Environmental operating variables, choose an Action $a(t)$ according to the policy functions in the Policy Network;
ii. Perform $a(t)$ (i.e., to generate and display ETA to users) and then the observe the change in the Environment and get the new State $s(t+1)$ and the Reward $r(t)$;
iii. Generate the next Action $a(t+1)$ by the policy function;
iv. Evaluate the value $q(t)$ of the current Action $a(t)$ and the value $q(t+1)$ of the next Action $a(t+1)$ by the value functions in Value Network;
v. Compute the value gradient and policy gradient given $q(t)$, $q(t+1)$, and $r(t)$;
vi. Update the Value network;
vii. Update the Policy network; and
viii. Repeat steps (i) to (vii) with incrementing time t

[0099] Figure 6 shows a block diagram 600 illustrating an overall framework, communications between a delivery platform and a system and a procedural flow for generating an estimated time to fulfil a request for a service according to yet another embodiment of the present disclosure.

[00100] The delivery platform is any application/website platform allowing consumers to place online orders and will allocate drivers to fulfill those orders accordingly. Some data are collected, recorded, and processed for model training. The main processes for the system are data collection, feature processing, model training and model deployment.

[00101] During the process of data collection, necessary data are collected from the platform and recorded in the offline databases (e.g., Data Base). The data can be also be directly retrieved and streamed from online database or cloud. The database can be hive or presto tables or any other databases. The collected data include:

a. Historical ETA & ATA of orders: timestamps to calculate ATA and predicted ETA
b. Drivers supply: drivers available to deliver orders
c. Demand supply: demands from consumers, including consumers' view & checkout data
d. Order details: Price, item details, quantity etc.
e. Weather data: rain volume

[00102] During the process of feature processing, the historical features and/or real-time features are generated by feature Engine and/or real-time processor respectively and stored in the Feature And Metric Store. The feature engine can be implemented in Spark. The real-time processor can be Flink or Spark streaming platform.

[00103] In the Feature And Metric Store, aggregated features may be further generated to provide real-time statistics, for example by compiling the features from historical orders/features collected within a specific time window in one market (e.g. 10 minutes time window, and these metrics will be updated every 10 minutes, and thus each time window would be a different state). Examples of aggregated features include:

a. ETA Accuracy & Promise: real time metrics;
b. Number of drivers & demand: real time signals about the driver & demand quantities;
c. Allocation rate: real time signal (orders with driver allocated/ orders created);
d. Conversion Rate: real time signal (number of checkout orders /number of views);
e. Weather signal: real time rain volume; and
f. Contextual features: hour of the day, day of the week, holiday, region etc.

[00104] Order level features may also be further calculated in the Feature And Metric Score from the features from individual orders placed by the consumers to provide order statistics. Example of order level features include:

a. ATA statistics: (mean, medium of ATA, driver allocation time, order preparation time, driver picking up time, driver drop off time); and
b. Order information: order price, number of items in the order, delivery distance, driver speed and estimated driver on-road time.

[00105] The Feature And Metric Store can be any key-value database. The aggregated features will be applied in the Value Network, while the aggregated and order level features be the input for the Policy Network for model training.

[00106] During the process of model training, the historical features and real-time features from the feature store are fetched and applied to train the models (e.g., Value Network and Policy Network) following the steps (i) to (viii) described above, for example to adjust the value function, policy function and respective weightages applied to the features for generate next action (e.g., ETA) to better achieve desired targets and metrics.

[00107] For the process of model deployment, the optimized policy network will then be deployed to predict and generate ETAs for the current and further service requests.

[00108] Figure 7 shows a schematic diagram of a general purpose computer system upon which the coordination server 108 of Figure 1 can be practiced. The computer system 700 includes: a computer module 701, input devices such as a keyboard 702, a mouse pointer device 703, a scanner 726, a camera 727, and a microphone 780; and output devices including a printer 715, a display device 714 and loudspeakers 717. An external Modulator-Demodulator (Modem) transceiver device 716 may be used by the computer module 701 for communicating to and from a communications network 720 via a connection 721. The communications network 720 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network, or a private WAN. Where the connection 721 is a telephone line, the modem 716 may be a traditional “dial-up” modem. Alternatively, where the connection 721 is a high capacity (e.g., cable) connection, the modem 716 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 720.

[00109] The input and output devices may be used by an operator who is interacting with the coordination server 108. For example, the printer 715 may be used to print reports relating to the status of the coordination server 108.

[00110] The coordination server 108 uses the communications network 720 to communicate with the provider device 104, the requestor device 102, the estimated time generation server 112 and the database 133 to receive commands and data. The coordination server 108 also uses the communications network 720 to communicate with the provider device 104, the requestor device 102, the estimated time generation server 112 and the database 133 to send notification messages or transaction, service request, location, user and/or route data.

[00111] The computer module 701 typically includes at least one processor unit 705, and at least one memory unit 706. For example, the memory unit 706 may have semiconductor random access memory (RAM) and semiconductor read only memory (ROM). The computer module 701 also includes a number of input/output (I/O) interfaces including: an audio-video interface 707 that couples to the video display 714, loudspeakers 717 and microphone 780; an I/O interface 713 that couples to the keyboard 702, mouse 703, scanner 726, camera 727 and optionally a joystick or other human interface device (not illustrated); and an interface 708 for the external modem 716 and printer 715. In some implementations, the modem 716 may

be incorporated within the computer module 701, for example within the interface 708. The computer module 701 also has a local network interface 711, which permits coupling of the computer system 700 via a connection 223 to a local-area communications network 722, known as a Local Area Network (LAN). As illustrated in Figure 9, the local communications network 722 may also couple to the wide network 720 via a connection 724, which would typically include a so-called “firewall” device or device of similar functionality. The local network interface 711 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 702.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 711.

[00112] The I/O interfaces 708 and 713 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 709 are provided and typically include a hard disk drive (HDD) 710. Other storage devices such as a floppy disk drive and a magnetic tape drive (not illustrated) may also be used. An optical disk drive 712 is typically provided to act as a non-volatile source of data. Portable memory devices, such optical disks (e.g., CD-ROM, DVD, Blu-ray Disc™), USB-RAM, portable, external hard drives, and floppy disks, for example, may be used as appropriate sources of data to the coordination server 108.

[00113] The components 705 to 713 of the computer module 701 typically communicate via an interconnected bus 704 and in a manner that results in a conventional mode of operation of a computer system known to those in the relevant art. For example, the processor 705 is coupled to the system bus 704 using a connection 718. Likewise, the memory 706 and optical disk drive 712 are coupled to the system bus 704 by connections 719. Examples of computers on which the described arrangements can be practised include IBM-PC’s and compatibles, Sun Sparcstations, Apple Mac™ or like computer systems.

[00114] The methods of operating the coordination server 108, as shown in the processes of Figure 3, may be implemented as one or more software application programs 733 executable within the coordination server 108. In particular, the steps of the processes shown in Figure 3 are effected by instructions 731 (see Figure 8) in the software (i.e., computer program codes) 733 that are carried out within the coordination server 108. The software instructions 731 may be formed as one or more code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules performs the operation of the coordination server 108 and a second part and the corresponding code modules manages the API and

corresponding user interfaces in the provider device 104, the requestor device 102, and on the display 714. In other words, the second part of the software manages the interaction between (a) the first part and (b) any one of the provider device 104, the requestor device 102, and the operator of the server 108.

[00115] The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the coordination server 108 from the computer readable medium, and then executed by the computer system 700. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer program product. The use of the computer program product in the coordination server 108 preferably effects an advantageous apparatus for receiving/transmitting data including transaction, service request, location, user and/or route data used for the ETA prediction and generation processes performed by the estimated time generation server 112.

[00116] The software (i.e., computer program codes) 733 is typically stored in the HDD 710 or the memory 706. The software 733 is loaded into the computer system 700 from a computer readable medium (e.g., the memory 706), and executed by the processor 705. Thus, for example, the software 733 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 725 that is read by the optical disk drive 712. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the coordination server 108 preferably effects an apparatus for receiving/transmitting data including transaction, service request, location, user and/or route data used for ETA prediction and generation processes performed by the estimated time generation server 112.

[00117] In some instances, the application programs 733 may be supplied to the user encoded on one or more CD-ROMs 725 and read via the corresponding drive 712, or alternatively may be read by the user from the networks 720 or 722. Still further, the software can also be loaded into the coordination server 108 from other computer readable media. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the coordination server 108 for execution and/or processing by the processor 705. Examples of such storage media include floppy disks, magnetic tape, CD-ROM, DVD, Blu-ray™ Disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computer module 701. Examples of transitory or non-tangible computer readable transmission media that may also

participate in the provision of software, application programs, instructions and/or data to the computer module 701 include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

[00118] The second part of the application programs 733 and the corresponding code modules mentioned above may be executed to implement one or more API of the coordination server 108 with associated graphical user interfaces (GUIs) to be rendered or otherwise represented upon the display 714 or the display of the provider device 104 and the requestor device 102. Through manipulation of typically the keyboard 702 and the mouse 703, an operator of the server 110 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Similarly, on the provider device 104 and the requestor device 102, a user of those devices 102, 104 manipulate the input devices (e.g., touch screen, keyboard, mouse, etc.) of those devices 102, 104 in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an audio interface utilizing speech prompts output via the loudspeakers 717 and user voice commands input via the microphone 780. These other forms of functionally adaptable user interfaces may also be implemented on the provider device 104 and the requestor device 102.

[00119] Figure 8 is a detailed schematic block diagram of the processor 705 and a “memory” 734. The memory 734 represents a logical aggregation of all the memory modules (including the HDD 709 and semiconductor memory 706) that can be accessed by the computer module 701 in Figure 7.

[00120] When the computer module 701 is initially powered up, a power-on self-test (POST) program 750 executes. The POST program 750 is typically stored in a ROM 749 of the semiconductor memory 706 of Figure 7. A hardware device such as the ROM 749 storing software is sometimes referred to as firmware. The POST program 750 examines hardware within the computer module 701 to ensure proper functioning and typically checks the processor 705, the memory 734 (609, 706), and a basic input-output systems software (BIOS) module 751, also typically stored in the ROM 749, for correct operation. Once the POST program 750 has run successfully, the BIOS 751 activates the hard disk drive 710 of Figure 7. Activation of the hard disk drive 710 causes a bootstrap loader program 752 that is resident on the hard disk drive 710 to execute via the processor 705. This loads an operating system 753 into the RAM memory 706, upon which the operating system 753 commences operation.

The operating system 753 is a system level application, executable by the processor 705, to fulfil various high level functions, including processor management, memory management, device management, storage management, software application interface, and generic user interface.

[00121] The operating system 753 manages the memory 734 (609, 706) to ensure that each process or application running on the computer module 701 has sufficient memory in which to execute without colliding with memory allocated to another process. Furthermore, the different types of memory available in the server 108 of Figure 7 must be used properly so that each process can run effectively. Accordingly, the aggregated memory 734 is not intended to illustrate how particular segments of memory are allocated (unless otherwise stated), but rather to provide a general view of the memory accessible by the server 108 and how such is used.

[00122] As shown in Figure 7, the processor 705 includes a number of functional modules including a control unit 739, an arithmetic logic unit (ALU) 740, and a local or internal memory 748, sometimes called a cache memory. The cache memory 748 typically includes a number of storage registers 744 - 746 in a register section. One or more internal busses 741 functionally interconnect these functional modules. The processor 705 typically also has one or more interfaces 742 for communicating with external devices via the system bus 704, using a connection 718. The memory 734 is coupled to the bus 704 using a connection 719.

[00123] The application program 733 includes a sequence of instructions 731 that may include conditional branch and loop instructions. The program 733 may also include data 732 which is used in execution of the program 233. The instructions 731 and the data 732 are stored in memory locations 728, 729, 730 and 735, 736, 737, respectively. Depending upon the relative size of the instructions 731 and the memory locations 728-630, a particular instruction may be stored in a single memory location as depicted by the instruction shown in the memory location 730. Alternately, an instruction may be segmented into a number of parts each of which is stored in a separate memory location, as depicted by the instruction segments shown in the memory locations 728 and 729.

[00124] In general, the processor 705 is given a set of instructions which are executed therein. The processor 705 waits for a subsequent input, to which the processor 705 reacts to by executing another set of instructions. Each input may be provided from one or more of a number of sources, including data generated by one or more of the input devices 702, 703, data received from an external source across one of the networks 720, 702, data retrieved

from one of the storage devices 706, 709 or data retrieved from a storage medium 725 inserted into the corresponding reader 712, all depicted in Figure 7. The execution of a set of the instructions may in some cases result in output of data. Execution may also involve storing data or variables to the memory 734.

[00125] The disclosed association management and payment initiation arrangements use input variables 754, which are stored in the memory 734 in corresponding memory locations 755, 756, 757. The association management and payment initiation arrangements produce output variables 761, which are stored in the memory 734 in corresponding memory locations 762, 763, 764. Intermediate variables 258 may be stored in memory locations 759, 760, 766 and 767.

[00126] Referring to the processor 705 of Figure 7, the registers 744, 745, 746, the arithmetic logic unit (ALU) 740, and the control unit 739 work together to perform sequences of micro-operations needed to perform “fetch, decode, and execute” cycles for every instruction in the instruction set making up the program 733. Each fetch, decode, and execute cycle comprises: a fetch operation, which fetches or reads an instruction 731 from a memory location 728, 729, 730; a decode operation in which the control unit 739 determines which instruction has been fetched; and an execute operation in which the control unit 739 and/or the ALU 740 execute the instruction.

[00127] Thereafter, a further fetch, decode, and execute cycle for the next instruction may be executed. Similarly, a store cycle may be performed by which the control unit 739 stores or writes a value to a memory location 732.

[00128] Each step or sub-process in the processes of Figure 3 is associated with one or more segments of the program 733 and is performed by the register section 744, 745, 747, the ALU 740, and the control unit 739 in the processor 705 working together to perform the fetch, decode, and execute cycles for every instruction in the instruction set for the noted segments of the program 733.

[00129] It is to be understood that the structural context of the coordination server 108 is presented merely by way of example. Therefore, in some arrangements, one or more features of the coordination server 108 may be omitted. Also, in some arrangements, one or more features of the coordination server 108 may be combined. Additionally, in some arrangements, one or more features of the coordination server 108 may be split into one or more component parts.

[00130] Figure 9 shows an alternative computer device to implement the coordination server 108 of Figure 1 (i.e., the computer system 700). In the alternative implementation, the coordination server 108 may be generally described as a physical device comprising at least one processor 902 and at least one memory 904 including computer program codes. The at least one memory 904 and the computer program codes are configured to, with the at least one processor 902, cause the coordination server 108 to facilitate the operations described in the processes of Figure 3. The coordination server 108 may also include a coordination module 906. The memory 904 stores computer program code that the processor 902 compiles to have each of the modules performs their respective functions.

[00131] With reference to Figure 1, the coordination module 906 performs the function of communicating with the requestor device 102 and the provider device 104; and the acquirer server 106 and the issuer server 110 to respectively receive and transmit transaction, service request, location, user and/or route data. Further, the coordination module 906 may provide data and information such as transaction, service request, location, user and/or route data used for the ETA prediction and generation processes performed by the estimated time generation server 112.

[00132] Figures 10 shows a schematic diagram of a general purpose computer system 1000 upon which the estimated time generation server 112 of Figure 1 can be practiced. The computer system 1000 includes: a computer module 1001, input devices such as a keyboard 1002, a mouse pointer device 1003, a scanner 1026, a camera 1027, and a microphone 1080; and output devices including a printer 1015, a display device 1014 and loudspeakers 1017. An external Modulator-Demodulator (Modem) transceiver device 1016 may be used by the computer module 1001 for communicating to and from a communications network 1020 via a connection 1021. The communications network 1020 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network, or a private WAN. Where the connection 1021 is a telephone line, the modem 1016 may be a traditional “dial-up” modem. Alternatively, where the connection 1021 is a high capacity (e.g., cable) connection, the modem 1013 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 1020.

[00133] The input and output devices may be used by an operator who is interacting with the estimated time generation server 112. For example, the printer 1015 may be used to print reports relating to the status of the estimated time generation server 112.

[00134] The estimated time generation server 112 uses the communications network 1020 to communicate with the provider device 104, the requestor device 102, the coordination server 108 and the database 113 to receive commands and data. The estimated time generation server 112 also uses the communications network 1020 to communicate with the provider device 104, the requestor device 102, the coordination server 108 and the database 113 to send notification messages or transaction, service request, location, user and/or route data.

[00135] The computer module 1001 typically includes at least one processor unit 1005, and at least one memory unit 1006. For example, the memory unit 1006 may have semiconductor random access memory (RAM) and semiconductor read only memory (ROM). The computer module 1001 also includes a number of input/output (I/O) interfaces including: an audio-video interface 1007 that couples to the video display 1014, loudspeakers 1017 and microphone 1080; an I/O interface 1013 that couples to the keyboard 1002, mouse 1003, scanner 1026, camera 1027 and optionally a joystick or other human interface device (not illustrated); and an interface 1008 for the external modem 1016 and printer 1015. In some implementations, the modem 1016 may be incorporated within the computer module 1001, for example within the interface 1008. The computer module 1001 also has a local network interface 1011, which permits coupling of the computer system 1000 via a connection 1023 to a local-area communications network 1022, known as a Local Area Network (LAN). As illustrated in Figure 10, the local communications network 1022 may also couple to the wide network 1020 via a connection 1024, which would typically include a so-called “firewall” device or device of similar functionality. The local network interface 1011 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 802.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 1011.

[00136] The I/O interfaces 1008 and 1013 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 1009 are provided and typically include a hard disk drive (HDD) 1010. Other storage devices such as a floppy disk drive and a magnetic tape drive (not illustrated) may also be used. An optical disk drive 1012 is typically provided to act as a non-volatile source of data. Portable memory devices, such optical disks (e.g., CD-ROM, DVD, Blu-ray Disc™), USB-RAM, portable, external hard drives, and floppy disks, for example, may be used as appropriate sources of data to the estimated time generation server 112.

[00137] The components 1005 to 1013 of the computer module 1001 typically communicate via an interconnected bus 1004 and in a manner that results in a conventional mode of operation of a computer system known to those in the relevant art. For example, the processor 1005 is coupled to the system bus 1004 using a connection 1018. Likewise, the memory 1006 and optical disk drive 1012 are coupled to the system bus 1004 by connections 1019. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations, Apple Mac™ or like computer systems.

[00138] The methods of operating the estimated time generation server 112, as shown in the processes of Figure 3, may be implemented as one or more software application programs 1033 executable within the estimated time generation server 112. In particular, the steps of the processes shown in Figure 3 are effected by instructions (see corresponding component 731 in Figure 8) in the software (i.e., computer program codes) 1033 that are carried out within the estimated time generation server 112. The software instructions 731 may be formed as one or more code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules performs the operation of the estimated time generation server 112 and a second part and the corresponding code modules manages the API and corresponding user interfaces in the provider device 104, the requestor device 102, and on the display 1014. In other words, the second part of the software manages the interaction between (a) the first part and (b) any one of the provider device 104, the requestor device 102, and the operator of the server 112.

[00139] The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the estimated time generation server 112 from the computer readable medium, and then executed by the computer system 1000. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer program product. The use of the computer program product in the estimated time generation server 112 preferably effects an advantageous apparatus for generating an ETA to fulfil a request for a service.

[00140] The software (i.e., computer program codes) 1033 is typically stored in the HDD 1010 or the memory 1006. The software 1033 is loaded into the computer system 1000 from a computer readable medium (e.g., the memory 1006), and executed by the processor 1005. Thus, for example, the software 1033 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 1025 that is read by the optical disk drive 1012. A computer readable medium having such software or computer program recorded on it is a computer program

product. The use of the computer program product in the estimated time generation server 112 preferably effects an apparatus for generating an ETA to fulfil a request for a service.

[00141] In some instances, the application programs 1033 may be supplied to the user encoded on one or more CD-ROMs 1025 and read via the corresponding drive 1012, or alternatively may be read by the user from the networks 1020 or 1022. Still further, the software can also be loaded into the estimated time generation server 112 from other computer readable media. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the estimated time generation server 112 for execution and/or processing by the processor 1005. Examples of such storage media include floppy disks, magnetic tape, CD-ROM, DVD, Blu-ray™ Disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computer module 1001. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computer module 1001 include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

[00142] The second part of the application programs 1033 and the corresponding code modules mentioned above may be executed to implement one or more API of the estimated time generation server 112 with associated graphical user interfaces (GUIs) to be rendered or otherwise represented upon the display 1014 or the display of the provider device 104 and the requestor device 102. Through manipulation of typically the keyboard 1002 and the mouse 1003, an operator of the server 112 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Similarly, on the provider device 104 and the requestor device 102, a user of those devices 102, 104 manipulate the input devices (e.g., touch screen, keyboard, mouse, etc.) of those devices 102, 104 in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an audio interface utilizing speech prompts output via the loudspeakers 1017 and user voice commands input via the microphone 680. These other forms of functionally adaptable user interfaces may also be implemented on the provider device 104 and the requestor device 102.

[00143] It is to be understood that the structural context of the coordination server 108 is presented merely by way of example. Therefore, in some arrangements, one or more features of the coordination server 108 may be omitted. Also, in some arrangements, one or more features of the estimated time generation server 112 may be combined. Additionally, in some arrangements, one or more features of the estimated time generation server 112 may be split into one or more component parts.

[00144] Figure 11 shows an alternative computer device to implement the estimated time generation server 112 of Figure 1. In the alternative implementation, the estimated time generation server 112 may be generally described as a physical device comprising at least one processor 1102 and at least one memory 1104 including computer program codes. The at least one memory 1104 and the computer program codes are configured to, with the at least one processor 1102, cause the estimated time generation server 112 of Figure 1 to perform the operations described in the processes of Figure 3. The estimated time generation server 112 may include various modules 202-208 of the system 200 described in Figure 2. The memory 1104 stores computer program code that the processor 1102 compiles to have each of the modules 202-208 performs their respective functions as described in Figure 2 and its accompanying description.

[00145] The estimated time generation server 112 may also include a data module 1106 configured to perform the functions of receiving data including transaction, service request, location, user and/or route data from the requestor device 102, provider device 104, coordination server 108, a cloud and other sources of information to facilitate the processes of Figure 3. For example, the data module 1106 may be configured to receive a request including data including transaction, service request, location, user and/or route data from one user device (such as the requestor device 102 or the provider device 104) the coordination server 108 for use in the ETA prediction and generation processes.

[00146] Figure 12 shows a schematic diagram of a general purpose computer system upon which a combined coordination and estimated time generation server 108, 112 of Figure 1 can be practiced. The computer system 1200 includes: a computer module 1201, input devices such as a keyboard 1202, a mouse pointer device 1203, a scanner 1226, a camera 1227, and a microphone 1280; and output devices including a printer 1215, a display device 1214 and loudspeakers 1217. An external Modulator-Demodulator (Modem) transceiver device 1216 may be used by the computer module 1201 for communicating to and from a communications network 1220 via a connection 1221. The communications network 1220 may be a wide-area network (WAN), such as the Internet, a cellular telecommunications network,

or a private WAN. Where the connection 1221 is a telephone line, the modem 1216 may be a traditional “dial-up” modem. Alternatively, where the connection 1221 is a high capacity (e.g., cable) connection, the modem 1213 may be a broadband modem. A wireless modem may also be used for wireless connection to the communications network 1020.

[00147] The input and output devices may be used by an operator who is interacting with the combined coordination and estimated time generation server 108, 112. For example, the printer 1215 may be used to print reports relating to the status of the combined coordination and estimated time generation server 108, 112.

[00148] The combined coordination and estimated time generation server 108, 112 uses the communications network 1220 to communicate with the provider device 104, the requestor device 102, and the databases 109, 113 to receive commands and data. In one example, the databases 109, 113 may be combined, as shown in Figure 12. The combined coordination and estimated time generation server 108, 112 also uses the communications network 1220 to communicate with the provider device 104, the requestor device 102 and the databases 109, 113 to send notification messages or transaction, service request, location, user and/or route data.

[00149] The computer module 1201 typically includes at least one processor unit 1205, and at least one memory unit 1206. For example, the memory unit 1206 may have semiconductor random access memory (RAM) and semiconductor read only memory (ROM). The computer module 1201 also includes a number of input/output (I/O) interfaces including: an audio-video interface 1207 that couples to the video display 1214, loudspeakers 1217 and microphone 1280; an I/O interface 1213 that couples to the keyboard 1202, mouse 1203, scanner 1226, camera 1227 and optionally a joystick or other human interface device (not illustrated); and an interface 1208 for the external modem 1216 and printer 1215. In some implementations, the modem 1216 may be incorporated within the computer module 1201, for example within the interface 1208. The computer module 1201 also has a local network interface 1211, which permits coupling of the computer system 1200 via a connection 223 to a local-area communications network 1222, known as a Local Area Network (LAN). As illustrated in Figure 12, the local communications network 1222 may also couple to the wide network 1220 via a connection 1224, which would typically include a so-called “firewall” device or device of similar functionality. The local network interface 1211 may comprise an Ethernet circuit card, a Bluetooth® wireless arrangement or an IEEE 802.11 wireless arrangement; however, numerous other types of interfaces may be practiced for the interface 1211.

[00150] The I/O interfaces 1208 and 1213 may afford either or both of serial and parallel connectivity, the former typically being implemented according to the Universal Serial Bus (USB) standards and having corresponding USB connectors (not illustrated). Storage devices 1209 are provided and typically include a hard disk drive (HDD) 1210. Other storage devices such as a floppy disk drive and a magnetic tape drive (not illustrated) may also be used. An optical disk drive 1212 is typically provided to act as a non-volatile source of data. Portable memory devices, such optical disks (e.g., CD-ROM, DVD, Blu-ray Disc™), USB-RAM, portable, external hard drives, and floppy disks, for example, may be used as appropriate sources of data to the combined coordination and estimated time generation server 108, 112.

[00151] The components 1205 to 1213 of the computer module 1201 typically communicate via an interconnected bus 1204 and in a manner that results in a conventional mode of operation of a computer system known to those in the relevant art. For example, the processor 1205 is coupled to the system bus 1204 using a connection 1218. Likewise, the memory 1206 and optical disk drive 1212 are coupled to the system bus 1204 by connections 1219. Examples of computers on which the described arrangements can be practised include IBM-PC's and compatibles, Sun Sparcstations, Apple Mac™ or like computer systems.

[00152] The methods of operating the combined coordination and estimated time generation server 108, 112, as shown in the processes of Figure 3, may be implemented as one or more software application programs 1233 executable within the combined coordination and estimated time generation server 108, 112. In particular, the steps of the processes shown in Figure 3 are effected by instructions (see corresponding component 731 in Figure 8) in the software (i.e., computer program codes) 1233 that are carried out within the combined coordination and estimated time generation server 108, 112. The software instructions 731 may be formed as one or more code modules, each for performing one or more particular tasks. The software may also be divided into two separate parts, in which a first part and the corresponding code modules performs the operation of the combined coordination and estimated time generation server 108, 112 and a second part and the corresponding code modules manages the API and corresponding user interfaces in the provider device 104, the requestor device 102, and on the display 1214. In other words, the second part of the software manages the interaction between (a) the first part and (b) any one of the provider device 104, the requestor device 102, and the operator of the server 112.

[00153] The software may be stored in a computer readable medium, including the storage devices described below, for example. The software is loaded into the combined coordination and estimated time generation server 108, 112 from the computer readable

medium, and then executed by the computer system 1200. A computer readable medium having such software or computer program recorded on the computer readable medium is a computer program product. The use of the computer program product in the combined coordination and estimated time generation server 108, 112 preferably effects an advantageous apparatus for generating an ETA to fulfil a request for a service as well as for receiving/transmitting data including transaction, service request, location, user and/or route data used for the ETA prediction and generation processes performed by the estimated time generation server 112.

[00154] The software (i.e., computer program codes) 1233 is typically stored in the HDD 1210 or the memory 1206. The software 1233 is loaded into the computer system 1200 from a computer readable medium (e.g., the memory 1206), and executed by the processor 1205. Thus, for example, the software 1233 may be stored on an optically readable disk storage medium (e.g., CD-ROM) 1225 that is read by the optical disk drive 1212. A computer readable medium having such software or computer program recorded on it is a computer program product. The use of the computer program product in the combined coordination and estimated time generation server 108, 112 preferably effects an apparatus for generating an ETA to fulfil a request for a service as well as for receiving/transmitting data including transaction, service request, location, user and/or route data used for the ETA prediction and generation processes.

[00155] In some instances, the application programs 1233 may be supplied to the user encoded on one or more CD-ROMs 1225 and read via the corresponding drive 1212, or alternatively may be read by the user from the networks 1220 or 1222. Still further, the software can also be loaded into the combined coordination and estimated time generation server 108, 112 from other computer readable media. Computer readable storage media refers to any non-transitory tangible storage medium that provides recorded instructions and/or data to the combined coordination and estimated time generation server 108, 112 for execution and/or processing by the processor 1205. Examples of such storage media include floppy disks, magnetic tape, CD-ROM, DVD, Blu-ray™ Disc, a hard disk drive, a ROM or integrated circuit, USB memory, a magneto-optical disk, or a computer readable card such as a PCMCIA card and the like, whether or not such devices are internal or external of the computer module 1201. Examples of transitory or non-tangible computer readable transmission media that may also participate in the provision of software, application programs, instructions and/or data to the computer module 1201 include radio or infra-red transmission channels as well as a network connection to another computer or networked device, and the Internet or Intranets including e-mail transmissions and information recorded on Websites and the like.

[00156] The second part of the application programs 1233 and the corresponding code modules mentioned above may be executed to implement one or more API of the combined coordination and estimated time generation server 108, 112 with associated graphical user interfaces (GUIs) to be rendered or otherwise represented upon the display 1214 or the display of the provider device 104 and the requestor device 102. Through manipulation of typically the keyboard 1202 and the mouse 1203, an operator of the server 112 and the application may manipulate the interface in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Similarly, on the provider device 104 and the requestor device 102, a user of those devices 102, 104 manipulate the input devices (e.g., touch screen, keyboard, mouse, etc.) of those devices 102, 104 in a functionally adaptable manner to provide controlling commands and/or input to the applications associated with the GUI(s). Other forms of functionally adaptable user interfaces may also be implemented, such as an audio interface utilizing speech prompts output via the loudspeakers 1217 and user voice commands input via the microphone 680. These other forms of functionally adaptable user interfaces may also be implemented on the provider device 104 and the requestor device 102.

[00157] It is to be understood that the structural context of the coordination server 108 is presented merely by way of example. Therefore, in some arrangements, one or more features of the coordination server 108 may be omitted. Also, in some arrangements, one or more features of the combined coordination and estimated time generation server 108, 112 may be combined. Additionally, in some arrangements, one or more features of the combined coordination and estimated time generation server 108, 112 may be split into one or more component parts.

[00158] Figure 13 shows an alternative computer device to implement a combined coordination and estimated time generation server 108, 112 of Figure 1. In the alternative implementation, the combined coordination and estimated time generation server 108, 112 may be generally described as a physical device comprising at least one processor 1302 and at least one memory 1304 including computer program codes. The at least one memory 1304 and the computer program codes are configured to, with the at least one processor 1302, cause the combined coordination and estimated time generation server 108, 112 to perform the operations described in the processes of Figure 3. The combined coordination and estimated time generation server 108, 112 may include various modules 202-208 of the system 200 described in Figure 2. The memory 1304 stores computer program code that the processor 1302 compiles to have each of the modules 202-208 performs their respective functions as described in Figure 2 and its accompanying description.

[00159] The combined coordination and estimated time generation server 108, 112 may also include a coordination module 1308 configured to perform the function of communicating with the requestor device 102 and the provider device 104; and the acquirer server 106 and the issuer server 110 to respectively receive and transmit data including transaction, service request, location, user and/or route data that are used for the ETA prediction and generation processes.

[00160] The combined coordination and estimated time generation server 108, 112 may also include a data module 1306 configured to perform the functions of receiving data including transaction, service request, location, user and/or route data from the requestor device 102, provider device 104, coordination server 108, a cloud and other sources of information to facilitate the processes of Figure 3. For example, the data module 706 may be configured to receive a request with data including transaction, service request, location, user and/or route data from one user device (such as the requestor device 102 or the provider device 104) to perform the ETA prediction and generation processes.

[00161] The foregoing describes only some embodiments of the present disclosure, and modifications and/or changes can be made thereto without departing from the scope and spirit of the invention, the embodiments being illustrative and not restrictive.

CLAIMS

1. A method for generating an estimated time to fulfil a request for a service, the service requiring a service provider to move from one location to another location, comprising:

identifying an operating variable relating to a current request for a service and a performance metric, the performance metric comprising an accuracy metric in accurately estimating an actual time required to fulfil a previous request for the service; and

generating an estimated time to fulfil the current request for the service to provide a desired number of further requests for the service based on the operating variable and the performance metric.

2. The method of claim 1, further comprising:

generating an initial estimated time to fulfil the current request for the service to provide the desired number of further requests for the service; wherein the accuracy metric is a measure relating to a time difference between the actual time required to fulfil the previous request and the initial estimated time to fulfil the current request.

3. The method of claim 2, wherein the generation of the estimated time to fulfil the current request for the service comprises:

generating an adjustment to the initial estimated time based on the operating variable and the performance metric; and

applying the adjustment to the initial estimated time to generate the estimated time.

4. The method of claim 2 or 3, wherein the initial estimated time is generated based on the performance metric, the actual time required to fulfil the previous request for the service and/or a previous estimate time generated for the previous request for the service based on a previous operating variable relating to the previous request for the service and a pervious performance metric.

5. The method of any one of claims 2-4, wherein the performance metric further comprises a first measure of a number of previous requests for the service to a number of views for the service and/or a second measure of a number of previous requests for the service fulfilled by one or more service providers derived from one or more previous requests based on the initial estimated time generated for the current request.

6. The method of any one of claims 1-5, further comprising:

determining whether the desired number of further requests for the service or a desired performance metric can be provided by one or more previous estimated times generated for one or more previous requests based on the operating variable and the performance metric, wherein the generation of the estimated time to fulfil the current request for the service to provide the desired number of further requests for the service is further based on a result of the determination.

7. The method of claim 6, further comprising:

applying a first weightage to each of one or more variables of the operating variable and/or one or more measures of the performance metric, wherein the generation of the estimated time to fulfil the current request for the service to provide the desired number of further requests for the service is further based on the weighted one or more variables of the operating variable and/or one or more measures of the performance metric.

8. The method of claim 6, further comprising:

identifying a performance metric relating to each of the one or more previous estimated times of the one or more previous requests; and

applying a second weightage to the each of the one or more previous estimated time according to a time sequence, wherein the generation of the estimated time to fulfil the current request for the service to provide the desired number of further requests for the service is further based on the respective weighted performance metrics of the one or more previous estimated times.

9. The method of claim 7 or 8, further comprising:

adjusting the weightage based on the result of the determination.

10. The method of any one of claims 1-9, wherein the operating variable comprises at least one of a number of current requests for the service, a number of available service providers to fulfil the service, a weather condition, a real time, a real-time condition, a real-time signal, a region, a period of a day, a day of a week, a specific day of a year and a third measure of a number of previous requests for the service assigned to and to be fulfilled by one or more service providers to a number of previous requests.

11. A system for adaptively generating an estimated time to fulfil a request for a service, the service requiring a service provider to move from one location to another location, comprising:

at least one processor; and

at least one memory including computer program code;

the at least one memory and the computer program code configured to, with at least one processor, cause the server at least to:

identify an operating variable relating to a current request for a service and a performance metric, the performance metric comprising an accuracy metric in accurately estimating an actual time required to fulfil a previous request for the service; and

generate an estimated time to fulfil the current request for the service to provide a desired number of further requests for the service based on the operating variable and the performance metric.

12. The system of claim 11, wherein the at least one memory and the computer program code configured to, with the at least one processor, cause the server at least to further:

generating an initial estimated time to fulfil the request for the service to provide the desired number of further requests for the service; wherein the accuracy metric is a measure relating to a time difference between the actual time required to fulfil the previous request and the initial estimated time to fulfil the current request.

13. The system of claim 12, wherein the at least one memory and the computer program code configured to, with the at least one processor, cause the server at least to:

generate an adjustment to the initial estimated time based on the operating variable and the performance metric; and

apply the adjustment to the initial estimated time to generate the estimated time.

14. The system of claim 12 or 13, wherein the initial estimated time is generated based on the performance metric and/or the actual time required to fulfil the previous request for the service and/or a previous estimate time generated for the previous request for the service based on a previous operating variable relating to the previous request for the service and a pervious performance metric.

15. The system of any one of claims 12-14, wherein the performance metric further comprises a first measure of a number of previous requests for the service to a number of views for the service and/or a second measure of a number of previous requests for the service fulfilled by one or more service providers derived from one or more previous requests based on the initial estimated time generated for the current request.

16. The system of any one of claims 11-15, The system of claim 11, wherein the at least one memory and the computer program code configured to, with the at least one processor, cause the server at least to further:

determine whether the desired number of further requests for the service or a desired performance metric can be provided by one or more previous estimated times generated for one or more previous requests based on the operating variable and the performance metric; and

generate the estimated time to fulfil the current request for the service to provide the desired number of further requests for the service further based on a result of the determination.

17. The system of claim 16, wherein the at least one memory and the computer program code configured to, with the at least one processor, cause the server at least to further:

apply a weightage to each of one or more variables of the operating variable and/or one or more measures of the performance metric; and

generate the estimated time to fulfil the current request for the service to provide the desired number of further requests for the service further based on the weighted one or more variables of the operating variable and/or one or more measures of the performance metric.

18. The system of claim 16, wherein the at least one memory and the computer program code configured to, with the at least one processor, cause the server at least to further:

identify a performance metric relating to each of the one or more previous estimated times of the one or more previous requests; and

apply a second weightage to the each of the one or more previous estimated time according to a time sequence, and

generate the estimated time to fulfil the current request for the service to provide the desired number of further requests for the service further based on the respective weighted performance metrics of the one or more previous estimated times.

19. The system of claim 17 or 18, wherein the at least one memory and the computer program code configured to, with the at least one processor, cause the server at least to further:

adjust the weightage to the each of one or more variables of the operating variable and/or one or more measures of the performance metric based on the result of the determination.

20. The system of any one of claims 10-19, wherein the operating variable comprises at least one of a number of current requests for the service, a number of available service providers to fulfil the service, a weather condition, a real time, a real-time condition, a real-time signal, a region, a period of a day, a day of a week, a specific day of a year and a third measure of a number of previous requests for the service assigned to and to be fulfilled by one or more service providers to a number of previous requests.