Software Requirements Specification

Amazon Go

Authors

Student 1:
Alperen ÇAYKUŞ
2237170

Student 2:
Aytaç Anıl Durmaz
2237295

Change History

Version	**Date**
1.0	03.03.2020
1.1	07.04.2020

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Purpose of the System

The purpose of this system, namely 'Amazon Go', is to enable customers to shop without making them wait in line when they are ready to pay. To achieve this purpose, Amazon has opened lots of stores, which are highly equipped with cutting edge sensors and cameras. These cameras and sensors track the customer and products, and after the customer is done with shopping, she/he just walks out of store. The amount of his/her shopping will be automatically deduced from his/her specified payment method.

1.2 Scope

- System will have a mobile application, which will enable users (i.e, store customers) to interact with the system. Customers will log in to store by using the QR code provided to them by their mobile application. Mobile application is also the place where customers sign up to the system, see their current cart status, and past shoppings.
- System will use remote servers to keep data of the customers, such as current shopping session, payment information, etc. Remote server will also communicate with the mobile application to enable a customer to see his/her current shopping session, past shoppings, etc.
- System will have physical stores, which are equipped with very sensitive sensors and cameras. Using Artificial Intelligence and Computer Vision algorithms, these cameras and sensors will gather information from the store and communicate with the remote server.
- System will use several APIs to communicate with the specified payment method of the customers, such as bank accounts. By doing so, system will be able to withdraw money from the customer's account when she/he is done with shopping.
- System will use a database to store products' information. Only the system admins are able to make changes and read the database.
- System will keep log of all the shoppings of the users on the remote server for legal purposes. Only system admins can see these logs.
- System will also have a store worker interface to enable the store worker to troubleshoot in case something goes wrong in the store, such as sensor malfunctioning, incorrect amount of money withdrawn from the customer, wrong product was added to a customer's cart, etc.

1.3 System Overview

1.3.1 System Perspective

Amazon Go is a smart store, where you do not have to wait in line to pay what you have bought for. A customer can log in to store by downloading the Amazon Go application to his/her smartphone, and then using his/her Amazon account, he/she can first log in to the mobile application, and then can enter the store by using the QR code produced by the mobile application.

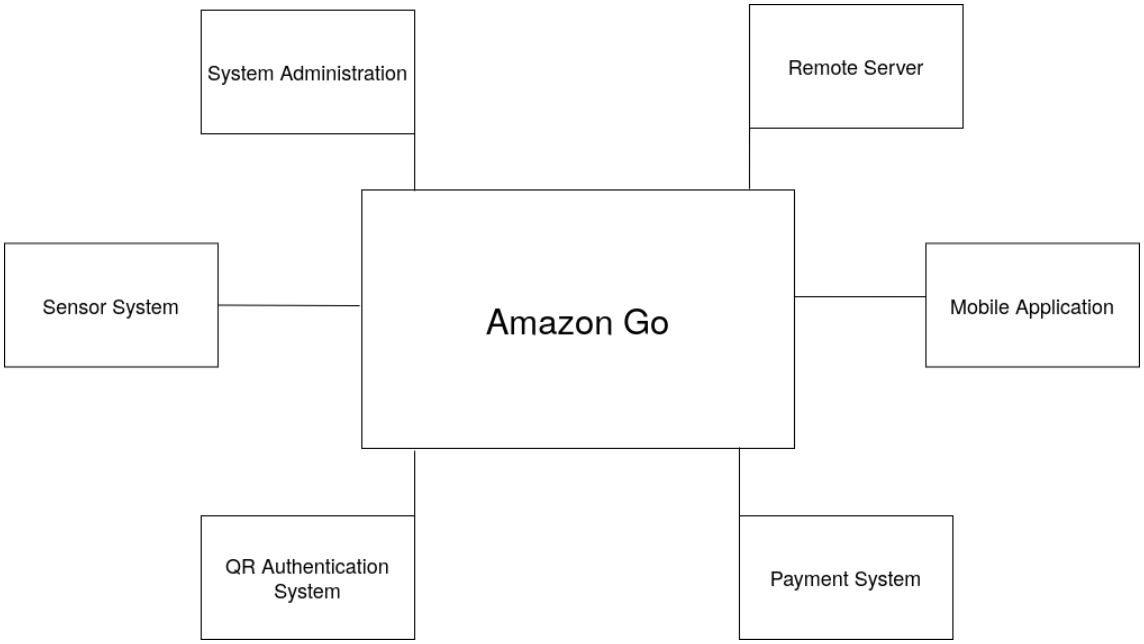

Figure 1: Context Diagram

Using cutting edge Artificial Intelligence and Computer Vision algorithms, the system watches the customers' movements, and based on their acts, the system either adds a product to the customer's cart, or removes it. If a customer is done with shopping, all she/he has to do is just walk out from the store. The products in his/her carts will automatically be paid by his/her connected Amazon account. Figure 1 shows the Context Diagram.

1.3.1.1 System Interfaces

- **Remote Server Interface:** This interface collects data from the sensors, processes them and depending on the output, it communicates with the mobile application. This interface creates log files for each input and output for the legal purposes and

in case of customer complaints.

- **Sensor Management Interface:** This interface is responsible for intercommunication of sensors and cameras. As long as the shop is open, this interface is always active. If a customer is tracked by some sensors, then by communicating with this interface, other sensors will focus on the other customers. This interface also has functions that enables a system administrator to connect these sensors and view current sensor data.

- **Database Management Interface:** One of the most essential part of the system. Every user's information, products' information, store locations, store stocks, customers' current cart information, customers' past shopping information resides on this database. The cases when a user signs up, when a new product arrives, when a product is out of stock in some Amazon Go store will be handled by this interface.

- **Payment API:** When a user is signing up, he/she must provide payment details so that Amazon Go can automatically complete the payment when the user leaves the store. Depending on the customer's payment preference, different APIs will be used. For instance, if user wants to pay via Paypal, then Paypal API must be used.

1.3.1.2 User Interfaces

- **User Sign-up Interface:** When an unregistered user downloads the app for the first time, they will be asked to register. Then, user needs to press the sign-up button. After that, user will be redirected to a sign-up procedure. In the first phase, user will be asked to enter a username, a password and their personal information. In the second phase, user needs to provide a valid payment method. After the validation, user will be registered and an introduction about how to use the system will be shown.

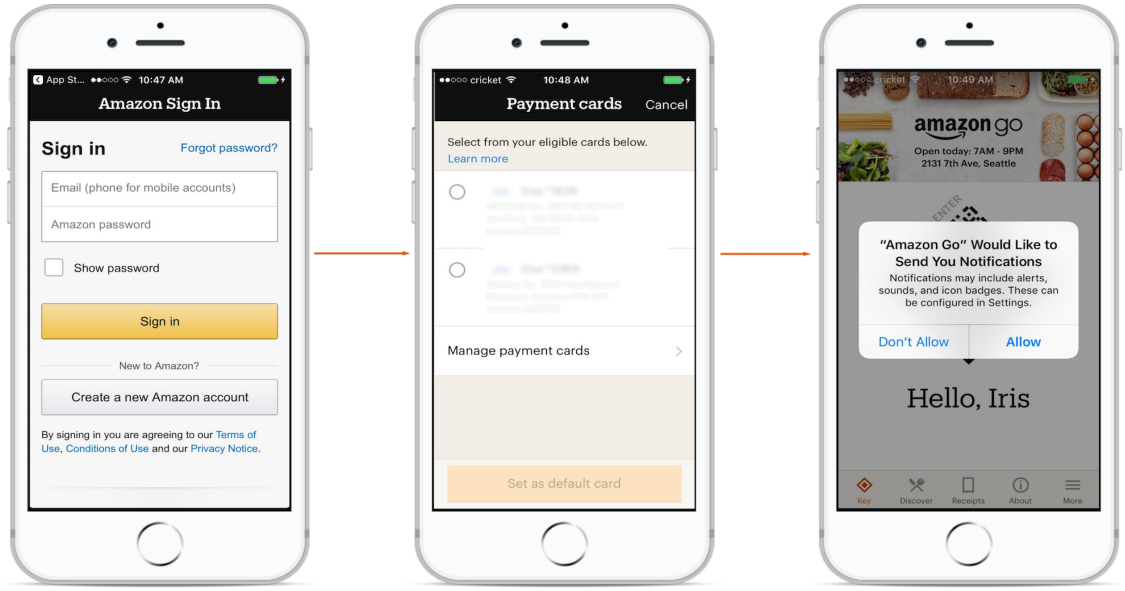

Figure 2: Sign-up Procedure

- **User Cart Interface:** A registered user is able to display their current cart status while shopping. While the customer is shopping, main page will be the cart status. In this interface, location of the store users shopping at, the items they took, the quantities and the prices of the items and the total price will be shown.

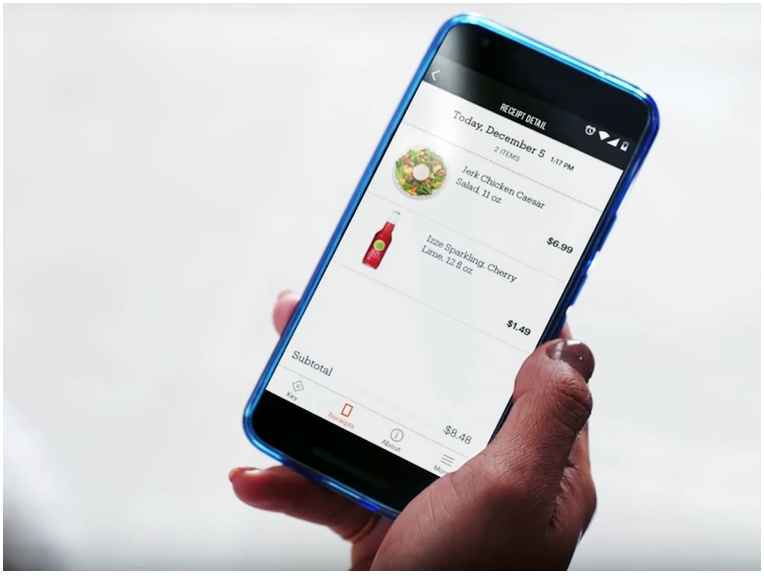

Figure 3: Cart Status

- **User Payment Interface:** A registered user can change their payment method and see their invoices of previous shop sessions. In this interface, there will be a 'Change Payment Method' button at the top and previous sessions will be listed at the below. User will be able to select a session. After selecting a session, he/she will be able to see the full detail about that session and the corresponding payment.
- **System Admin Interface:** Admins are able to see the stock information of every store and the main warehouse. Also, they can see the system logs for investigation purposes.

1.3.1.3 Software Interfaces

- **DBMS**: A DBMS is a must since any information related to customers, stores and products must be stored.
- **Mobile(Smart) Phone:** Since users need to get their QR code scanned when entering the store, they must own a smart phone to use Amazon Go application. Since Amazon Go is a light weight application, there is no need for a fancy smart phone. A smart phone that can browse on the web can also run the Amazon Go app.
- **Operating System:** For server side, a Linux distribution that is made for servers must be used. For the customer side, their smart phone must be running on iOS or Android.

1.3.1.4 Communication Interfaces

Payment information of users are sent over the HTTPS protocol so that it is encrypted and secured. Sensor data is also sent over this protocol as hackers may try to intervene with sensor information, which will result in fraud. System administrators will access to the remote server using SSH.

1.3.1.5 Memory Constraints

Since Computer Vision algorithms will be running, memory can be a big issue. Therefore, bigger memories should be chosen.

1.3.1.6 Operations

The operations can be listed as:
User Operations:

- Sign up

- Log in
- Send personal data
- Send payment data
- Interact with products
- Interact with mobile application
- Get his/her QR code scanned

Store Worker Operations:

- Place products to shelves
- Troubleshoot to system admins in case of a problem

System Admin Operations:

- Setup the store system
- Database maintenance
- View system logs
- Remote server maintenance
- Authorize store worker

System Operations:

- Customer detection and tracking
- Product detection and tracking
- Automatically completing the payment
- QR code generation
- Intercommunication of sensors and cameras
- Updating cart info

1.3.2 System Functions

No	Functionality	Description
1	Customer Recognition	Immediately after the customer gets his/her QR code scanned, environmental cameras track his/her movements and positions, and recognizes him/her continuously.
2	Product recognition	Immediately after a product is marked as exist in the database, product must be tracked via its weight information, visual information and shelf location.
3	Scan Login QR Code	Customers must log in to store by getting their QR codes scanned by the turnstiles.
4	Pick Product	When a customer picks a product from a shelf, this must be recognized by environmental sensors and cameras.
5	Put Back Product	When a customer puts a product back to a shelf, this must be recognized by environmental sensors and cameras.
6	View Cart Info	A customer is able to see his/her current cart info by using the mobile application.
7	Enter Personal and Payment Information	Customers provide personal and payment information during the sign up process via the mobile application.
8	End Shopping Session	When customers leaving the store, sensors and cameras detect this and report this to the remote server, then the server ends the session.
9	Generate Login QR Code	Mobile application generates a unique QR code to enable a customer to log in to store.
10	Withdraw Money	When user leaves the store, by using his/her provided payment information, his cart's total cost will be withdrawn via his/her provided payment method.
11	View Store Stock	System admins are able to see the current stock status of each store and main warehouse.
12	View System Logs	Administrators must be able to see previous shopping session details of customers.

Table 1: System Functions

1.3.3 User Characteristics

The target users of Amazon Go can be divided into three parts. First one is customers(i.e, end users). Customers do not have to have any technical knowledge. They only need to know how to use a smart phone, how to install an application to a smart phone by using

the application store of the phone, and must know how to sign-in and sign-up. They also need to know how to use the store. In other words, they must know that there is no line in the store, they just need to walk out.
The second one is system administrators. System administrators must have deep knowledge in their fields as they will be responsible of system setup, maintenance and take care of troubles in the system. They must also be experts in the IT related jobs, such as database maintenance, server maintenance, etc.
The last one is the store workers. Store workers does not have to know much as system administrators know, but they must know more than customers to help them, or troubleshoot to system administrators whenever there is a problem in the system. They also need to know how the system works, of course not the internal technical details, but via abstractions, such as sensors must see the products, hence, products must be placed to shelves accordingly.

1.3.4 Limitations

- **Regulatory policies**: Any information provided from the user should not be shared with anyone. System should not keep a record of payment details.
- **Hardware limitations:** Since the system completely depends on the sensors, best quality of these sensors are required. A smart phone with iOS or Android is required for the customers to use the application.
- **Interfaces to other applications:** All interfaces are internal in the system. There is no interface with other applications.
- **Parallel operation:** Parallelization is recommended as multiple customers would most probably exist in the same store at the same time.
- **Audit functions:** Financial operations will be handled by the contracted banks. Amazon Go should provide an interface to enter the payment information and to display payment results from the bank only.
- **Control functions:** Controlling functions that have access to server directly must be reachable by the authorized system admins only.
- **Higher-order language requirements:** Since there is a mobile application, for iOS, Swift must be used. For Android, Java must be used. As there are Artificial Intelligence and Computer Vision algorithms running in the background, C++ must be used to implement these algorithms. For the server-side programming, Java must be used.
- **Signal handshake protocols:** For the communication between the sensors and the server, HTTPS protocol must be used. HTTPS protocol must also be used to communicate with the payment method during the payment process. For system administrators, SSH connection must be used to access to the remote server. To access the database, TCP protocol must be used.

- **Quality requirements:** System should keep the personal data safe.System should take backups regularly.
- **Criticality of the application:** It is crucial to track the customers and products continuously. Therefore, constant maintenance should take place and backup servers must be present.
- **Safety and security considerations:** Customers should be informed about QR scans and scanning should also check phone location with GPS to prevent any fraud.Payment details like card number should not be kept. Any critical data like passwords should be encrypted. Database should be safe against any attack.
- **Physical/mental considerations:** Anyone that can use a smart phone and can shop by herself/himself can use the system.

1.4 Definitions

Term	**Definition**
Database	A MySQL database
User	A customer that is signed up to Amazon Go application and has a unique user ID.
User ID	A unique number which can be used to identify a user.
Store Worker	A real person working in an Amazon Go store.
API	Application Programming Interface
HTTPS	Hypertext Transfer Protocol Secure
SSH	Secure Shell
TCP	Transmission Control Protocol

Table 2: Definitions

2 References

This document is written with respect to the specifications of the document below:
29148-2011 - ISO/IEC/IEEE International Standard - Systems and software engineering – Life cycle processes –Requirements engineering.

Other sources:

Cheng, A. (2019, January 13). Why Amazon Go May Soon Change The Way We Shop. Retrieved from *https://www.forbes.com/sites/andriacheng/2019/01/13/why-amazon-go-may-soon-change-the-way-we-want-to-shop*

Wingfield, N. (2018, January 21). Inside Amazon Go, a Store of the Future. Retrieved from *https://www.nytimes.com/2018/01/21/technology/inside-amazon-go-a-store-of-the-future.html*

3 Specific Requirements

3.1 External Interfaces

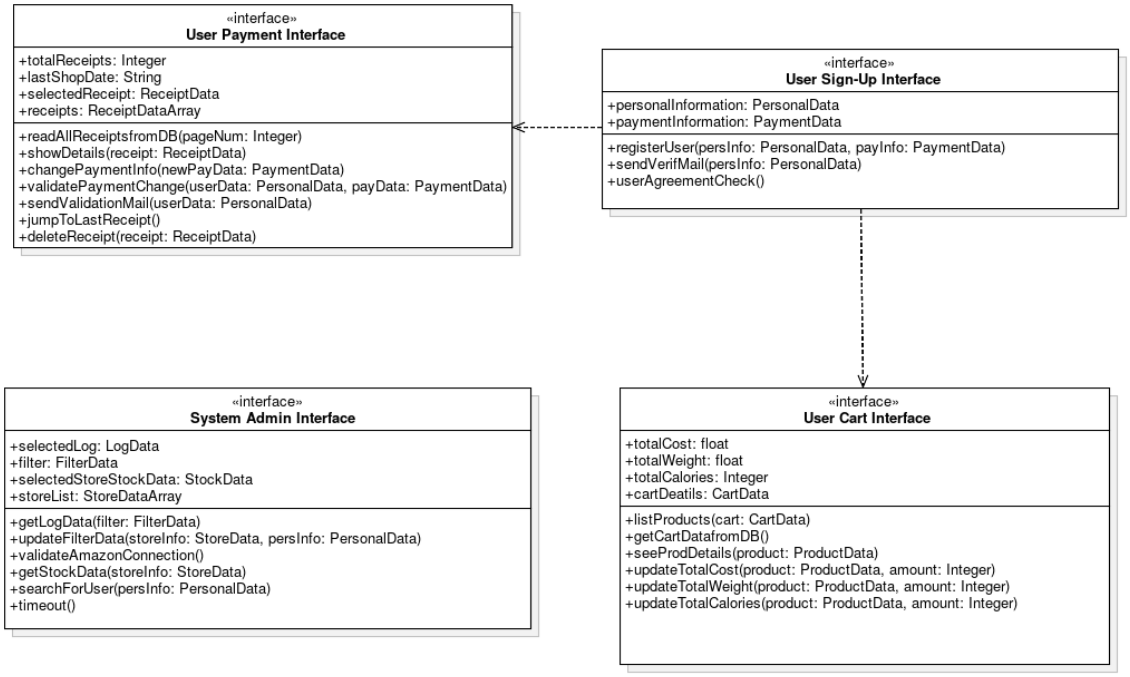

Figure 4: External Interfaces Class Diagram

- **User Sign-Up Interface:** This interface provides a user friendly way for non-registered users to register. Users enter his/her personal and payment information here, then these information are sent to the server and stored in the database. After a valid registration, users will be informed that the registration is successfully completed. By creating an account, users agree to Amazon Go's Conditions of Use and Privacy Notice automatically, which is stated during the registration. There is no time restriction in this part, but closing the application during the registration will require to start the registration from the beginning. Most of the information given in this part can be replaced later, only the username cannot be changed.
- **User Cart Interface:** This interface provides a simple and informative way for the customers to see the details of their current session while shopping. Users need

to open the app during the shopping and this screen will be directly opened when they open the app. Each item they took will be listed with the most important details: price, quantity, size-weight-volume and etc. Users must login to see this interface, but since they can not enter the shop before the login, this will be not the case most of the time. Yet, if they logout after entering, they can login back and see their cart. If they close the app, they can simply open it back and the cart will be shown. This screen will end after the customer leaves the store. This screen will be designed for mobile phones especially.

- **User Payment Interface:** This interface is designed to be fully informative for the customers to see the every detail of their past shopping sessions. Users can see a list of their previous sessions and can select a session to see the every detail including the receipt. Also, there is a button available to change the payment method here, so that customers can reach this method easily. Input form of the payment method is identical to the input form at the registration. User will be informed about the validation of the payment method change after the change attempt. There is no time restriction in this part, receipts can bee seen immediately after a few minutes maximum and will be stored until it is deleted manually.

- **System Admin Interface:** This interface is designed to be informative about the stores and the user actions. System admin must login to the system with a authorized username and password to use this interface. This interface can be only reached from the Amazon facilities, it is unreachable from external. A list of the stores and warehouses will be seen after entering the system. User will be able to select a store or a warehouse to see the stock details of that place. Moreover, there will be a tab to switch to the logs. Here, user can make a search about any user by their username, name and e-mail. Results can be filtered with these action types: Registration time and date, entering to a specific store, Successful and failed payment attempts and blacklisted. User must login again after being idle for 30 minutes due to privacy of customer data. This screen will be implemented for a computer screen.

3.2 Functions

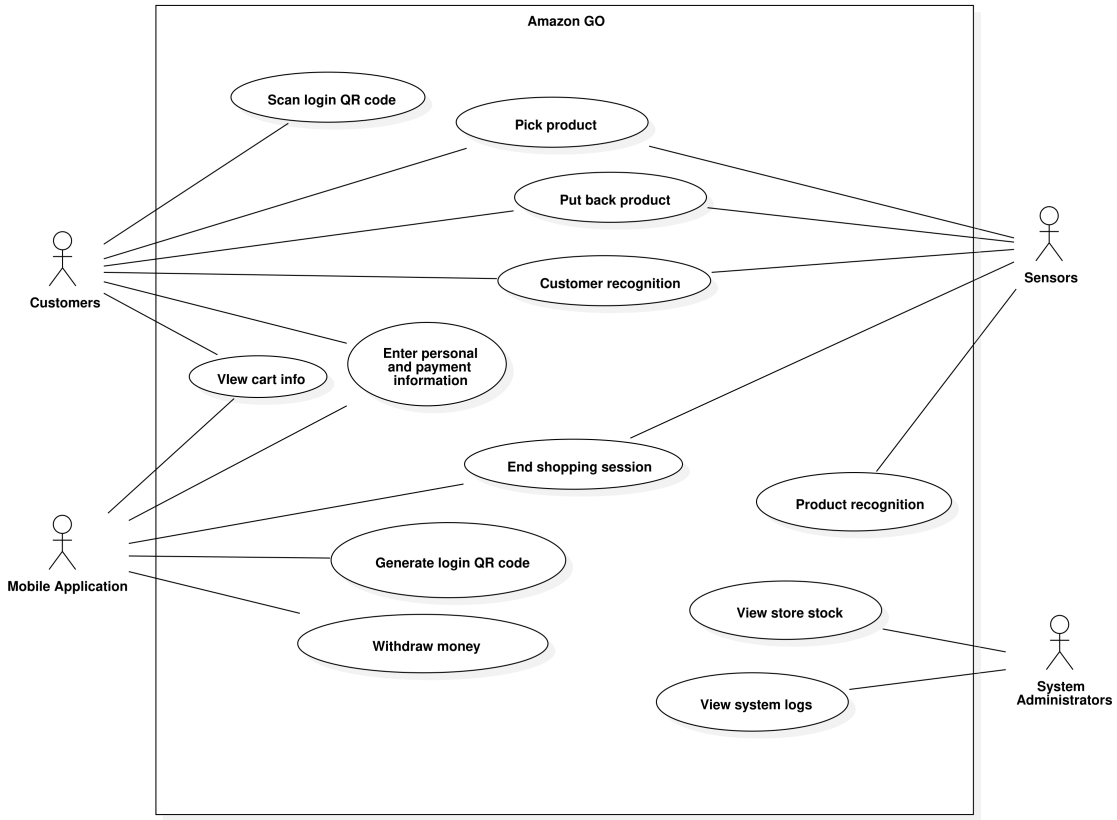

Figure 5: Use Case Diagram

Use case name	Customer recognition
Actors	Customers, Sensors
Description	Immediately after the customer gets his/her QR code scanned, environmental cameras must track his/her movements and positions continuously. Cameras must regularly recognize the customers with their face, clothes etc. and must communicate with the remote server until the customer leaves the store. In order to not to lose visual contact with the customer, every angle in the store must be watched by the cameras and the sensors.
Data	Customers' physical details such as clothes, face, and body movements.
Stimulus	The customer getting his/her QR code scanned.
Response	Match the user ID with the acquired visual data of the customer.
Basic Flow	1- Upon QR scanning, visual sensors receive a user ID and identify the unknown customer with the ID. 2- Visual sensors track customer's movements and position continuously. 3- Customer picks up a product. 4- Customer information is sent to the server. 5- Flow goes back to step 2.
Alternative Flow #1	3- Customer puts a product back to a shelf. 4- Customer information is sent to the server. 5- Flow goes back to step 2.
Alternative Flow #2	3- Customer leaves the store. 4- Customer information is sent to the server. 5- System stops tracking the customer.
Comments	Immediately after a customer logs in to store, sensors must assign a unique customer ID to that user.

Table 3: Customer Recognition

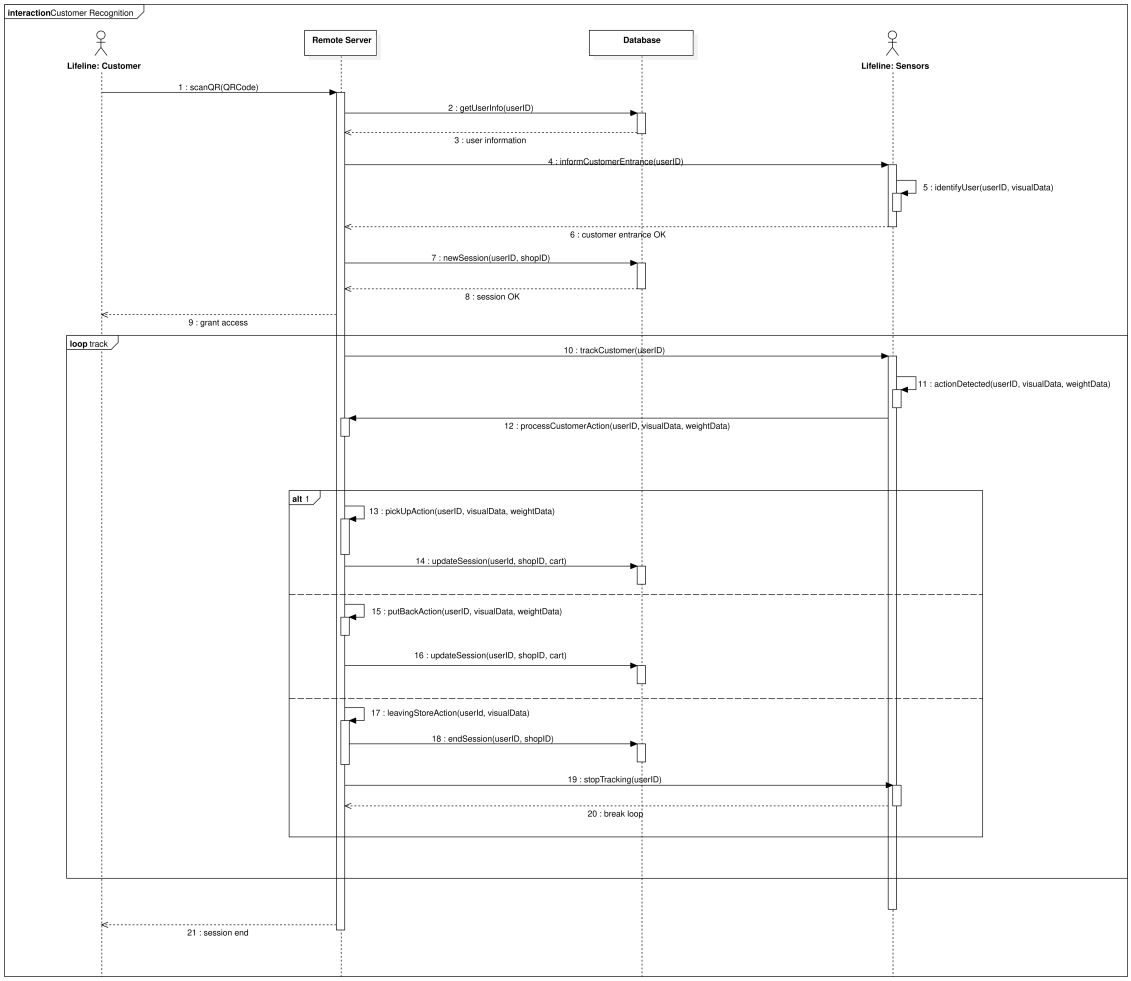

Figure 6: Customer Recognition Sequence Diagram

Use case name	Product recognition
Actors	Sensors
Description	Right after a product is placed to a shelf by the store staff and the product is marked as 'exist' on the store database, the corresponding ID which belongs to the same product set must be assigned to the product. After that point, until someone buys the product and leaves the store, product must be tracked by using its weight information, visual information, and shelf location. A customer is allowed to take a product, put it into his/her bag, and put the product back to any shelf. In that case, for some time, the visual contact with the product would be lost, but after the product is put back to the shelf, system must recover and continue tracking this product. A customer can put a product back to a different shelf, in that case, the product must be recognized by its visual data and weight again.
Data	Weight, physical characteristics, and the location of products.
Stimulus	A product is placed to a shelf by the store worker and marked as exist in the store.
Response	Match the product with the products in the database.
Basic Flow	1- A product is placed to a shelf by the staff. 2- Query the database with the shelf location, visual data and weight. 3- Assign the received product ID to corresponding product set. 4- Sensors work continuously to detect product movements. 5- A product is taken from a shelf. 6- Product information is sent to the server. 7- Flow goes back to step 4.
Alternative Flow	5- A product is placed to a shelf by a customer. 6- Product information is sent to the server. 7- Flow goes back to step 4.
Comments	Initially, products must be registered to the database and must be assigned to a shelf.

Table 4: Product Recognition

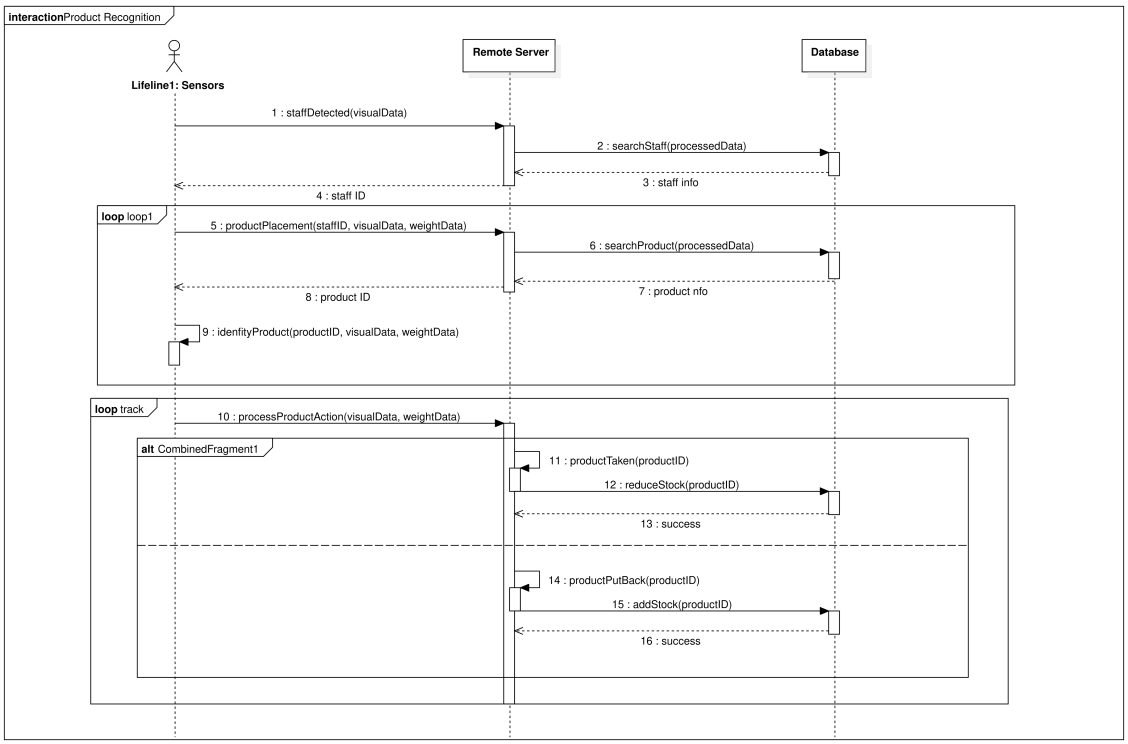

Figure 7: Product Recognition Sequence Diagram

Use case name	Scan login QR code
Actors	Customers
Description	A procedure to log users in to store by scanning their QR code via turnstiles. Turnstiles must allow access after a successful scan. Turnstiles must not allow the customers to enter the store if the scan fails and must display the reason. For security reasons, if the same QR is scanned while there is a shopping session with that QR code, login must not be allowed and QR code must be disabled for security purposes. Additionally, customer must be informed about every login with an SMS.
Data	User information.
Stimulus	Users showing their QR code to turnstiles.
Response	Validation or invalidation of the QR code.
Basic Flow	1- Customer shows the QR code to turnstile QR scanner. 2- Turnstile scans the QR and sends the data to the server. 3- Server processes the store information and the customer information. 4- Server responds with success. 5- Turnstile gives access to the customer. 6- Server sends SMS notification to the customer.
Alternative Flow	4- Server responses with failure. 5- Failure reason is displayed. 6- Server sends SMS notification to the customer.
Comments	Customer must be a registered member. GPS can be used to improve the security.

Table 5: Scan Login QR Code

Use case name	Pick Product
Actors	Customers, Sensors
Description	A procedure to detect customers when they pick up a product from the shelves. When a customer picks a product, weight sensors must detect the change and must send the related data, also visual sensors must recognize the customer who picks up the product and provide additional data about the product. If the customer picks up multiple products at the same time, sensors must be precise enough to handle the situation.
Data	Customer information, product's price and other information related to that product.
Stimulus	Customers picking products.
Response	Add the picked product to customer's cart, update the cart information.
Basic Flow	1- Customer picks a product up from a shelf. 2- Weight sensors detect the change. 3- Visual sensors detect the customer action and assists product recognition. 4- Sensors send data to server. 5- Server processes data, adds the related product to customer's cart.
Alternative Flow	-
Comments	Customers must get their QR code scanned successfully before shopping.

Table 6: Pick Product

Use case name	Put back product
Actors	Customers, Sensors
Description	A procedure to detect customers when they put a product back to shelves. When a customer puts a product back to a shelf, weight sensors must detect the change and must send shelf information, weight information and other related data, also visual sensors must recognize the customer who puts the product back and must provide additional data about the product. System must analyze the cart to be precise about the product. If the customer puts back multiple products at the same time, sensors must be precise enough to handle the situation.
Data	Customer information, product's price and other information.
Stimulus	Customers putting a product back to shelf.
Response	Delete the related product from the customer's cart.
Basic Flow	1- Customer puts a product back to a shelf. 2- Weight sensors detect the change. 3- Visual sensors detect the customer action and assists product recognition. 4- Sensors send data to server. 5- Server processes data, removes the related product from customer's cart.
Alternative Flow	-
Comments	The product must have been picked up beforehand and this must be in the same session.

Table 7: Put Back Product

Use case name	View cart info
Actors	Customers, Mobile Application
Description	A customer must be able to see it's current cart status by using the mobile app. Customer shall see the products she/he picked up, with their quantity and price. Total price must be also displayed.
Data	The customer's cart status.
Stimulus	The customer pressing the 'My Cart' button on the app.
Response	Show the customer's current cart status.
Basic Flow	1- Customer opens the app. 2- Customer logins if not signed in already. 3- Application opens the cart screen automatically. 4- Application sends view cart request to the server. 5- Server responds with the related data. 6- Application shows the cart status.
Alternative Flow	-
Comments	Data must be fetched from the remote server in case of an attempt to cheat. User must be shopping to be able to see the cart screen.

Table 8: View Cart Info.

Use case name	Enter personal and payment information
Actors	Customers, Mobile Application
Description	When a customer signs up, s/he must provide personal information and payment method. If the payment method is invalid, application must request a valid method and must not allow another action to be taken by the customer until a valid payment method is provided.
Data	Personal data and payment data.
Stimulus	When a customer signs up or updates his/her information.
Response	Validation or invalidation of the payment method. If validated, sign up or update the user.
Basic Flow	1- Application shows a personal information form or a payment form. 2- User enters information. 3- Application checks the validity of the information at the same time. 4- If everything is valid, a button is enabled to finish the process. 5- User presses the button. 6- Server inserts the related information to the database.
Alternative Flow	-
Comments	If the customer is a member, then instead of signing him/her up, update the information.

Table 9: Enter Personal and Payment Information

Use case name	End shopping session
Actors	Mobile Application, Sensors
Description	Visual sensors must detect when the customer leaves the store and report to remote server. Details about the shopping session must be visible from the application after session ends.
Data	Customer ID, store information.
Stimulus	Customer leaving the store.
Response	Session information.
Basic Flow	1- Customer walks out from the store. 2- Visual sensors detects the action. 3- End shopping session request is sent to servers with the customer ID. 4- Server processes the request and saves the sessions information to the database. 5- Visual sensors stop tracking that customer.
Alternative Flow	-
Comments	Log the current session to database for possible further use.

Table 10: End Shopping Session

Use case name	Generate login QR code
Actors	Mobile Application
Description	Application must generate a unique QR code to enable customers log in. A generated QR code can be used multiple times until it is disabled due to security or user prompt. QR codes must be user specific.
Data	Customer information.
Stimulus	User pressing the 'QR Code' button on the mobile app.
Response	Produce and show the QR code
Basic Flow	1- A customer signs up. 2- Application requests a QR code for the customer. 3- Server generates a QR code that is unique to the customer.
Alternative Flow	1- Customer presses the Refresh QR button. 2- Application requests a QR code for the customer. 3- Server generates a QR code that is unique to the customer.
Comments	QR code must be generated and sent by the remote server in case of a cheating attempt.

Table 11: Generate Login QR Code

Use case name	Withdraw money
Actors	Mobile Application
Description	When user leaves the store, withdraw money from his/her registered payment method. System must use the payment method provided by the customer. If the payment attempt fails due to lack of money or other reasons, notifications must be sent regularly until customer pays. Customer must not be allowed to shop until she/he pays.
Data	User ID.
Stimulus	Upon the 'End shopping session' procedure.
Response	Confirmation that payment is successful.
Basic Flow	1- System is informed by the End Shopping Session function. 2- System requests a payment from the user's payment method via the Payment API. 3- Payment is successful. 4- Receipt is created.
Alternative Flow	3- Payment is failed. 4- An SMS notification is sent to the customer. 5- The customer is blacklisted until the she/he pays.
Comments	Further actions can be taken by the authorities if a payment is not received for a long time.

Table 12: Withdraw Money

Use case name	View store stock
Actors	System Administrator
Description	System admins must be able to see current stock status of each store and main warehouse. User must be warned about low stocks.
Data	Store ID, admin ID.
Stimulus	User pressing the 'View Stock' button on the mobile app.
Response	Stock information .
Basic Flow	1- System admin opens the administration page. 2- User presses the stock status tab. 3- Overall stock status of each store is listed to the user. Stocks are colored according to the amount. 4- User selects a store. 5- Detailed stock status of the selected store is shown to the user.
Alternative Flow	-
Comments	User must be authorized to view stock information.

Table 13: View Store Stock

Use case name	View system logs
Actors	System Administrators
Description	Administrators must be able to see previous shopping session details of customers. Any inconsistency in sessions must be reported.
Data	Time and date, Store ID, Customer ID
Stimulus	Administrators reaching the log files.
Response	Previous cart information, payment method
Basic Flow	1- System admin opens the administration page. 2- User presses the view logs tab. 3- Several filter options are shown to the user. 4- User fills some of the filtering fields. 5- User presses the search button. 6- Results are listed.
Alternative Flow	-
Comments	Only administrators can see the log files.

Table 14: View System Logs

3.3 Usability Requirements

- A customer must be able to login to mobile application via his/her Amazon account.
- A customer must be able to access the generated QR code with a single click to 'QR Code' button.
- After getting into a store, a customer must be able to see his/her current shopping information by clicking 'My Cart' button on the app.
- After the customer leaves the store and required amount of money is withdrawn from his/her account, the receipt must be sent to him/her via e-mail and a notification must be sent to his/her mobile phone so that customer can immediately see the receipt.
- Since the GPS must be on to get a QR code, if someone tries to enter a store by hacking someone else's QR code, the real owner of the account must be notified via both e-mail and notification.
- A user must be able to see his/her every receipt via clicking the 'Previous Orders' menu on the mobile application.
- Users must be able to ask questions or see the FAQ by clicking to the 'Help' button on the application.
- Users must be able to set a money limit for a shopping session so that when the limit is exceeded, he/she is notified via a notification. For instance, if someone sets a 50$ limit but shops for 51$, he/she must be notified immediately.

3.4 Performance Requirements

- The process of generating a QR code and a user accessing to this QR code must last less than 2 seconds since otherwise, there will be a line in the entrance, which spoils the aim of the whole project.
- Since the sensors and cameras will send image data to the remote server, for a healthy system, an Internet connection with at least 4 Mbps upload and download speed is required.
- As computer vision and image processing algorithms consumes too much memory, server computers must have at least 2 times more and faster RAMs than a regular server computer.
- A customer must be able to see the changed cart information (i.e, when picks a product or puts a product back) in at most 10 seconds so that she/he can continue shopping with peace of mind.

- There is no limitation on the simultaneous users in a store. During the occasions such as 'Black Friday', the number of users may explode significantly. Therefore, parallel processing algorithms must be employed to take care of such situations.
- Android application must run on at least Android 4.1 Jelly Bean and the iOS application must run on at least iOS 8 to reach a wide group of users.

3.5 Logical Database Requirements

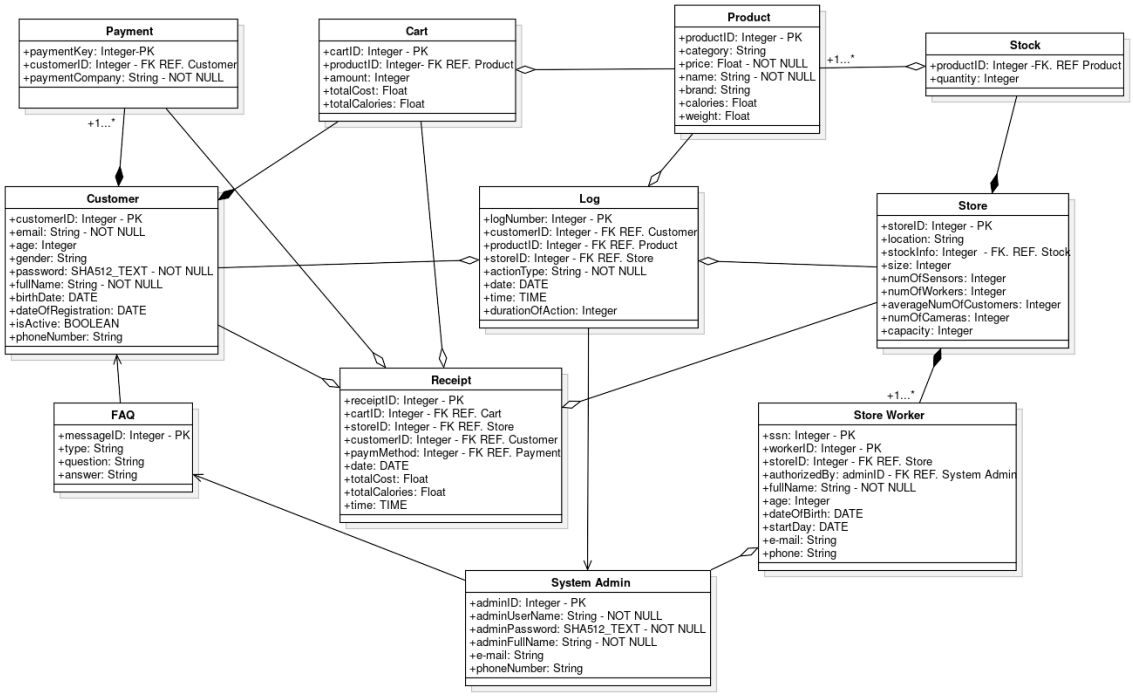

Figure 8: Logical Database Requirements Class Diagram

- A customer entry must be created if and only if the customer signs up via the registration interface.
- A customer can have multiple payment methods.
- Payment details will not be held in the Amazon servers. Instead, a key will be held in the Payment table. This key will be used to communicate with the payment provider, and the payment provider (such as PayPal) will complete the payment.
- A customer can delete himself/herself by using the mobile application, which provides an abstraction via the user interface.

- If a customer is deleted, the associated payment information must also be deleted.
- If a customer is deleted, the associated cart must also be deleted.
- Customers can view the FAQ.
- A customer can delete or update his/her payment entries via mobile application, which provides an abstraction via the user interface.
- Payment and cart tables are weak entities.
- Only system admins can add entries to the FAQ table.
- A store worker entry is created by a system admin. Each store worker entry is associated with a system admin entry.
- A store may have multiple store workers.
- If a store entry is deleted, then the store worker entries associated with this store must also be deleted.
- If a store is deleted, the associated stock information must also be deleted.
- Stock information has multiple products. If no product exists, then stock entry will also not exist.
- Stock table is a weak entity.
- Only system admins can view the logs.
- A log entry is associated with a customer entry, product entry, and a store entry. If one of them is missing, then the log entry cannot exist.
- A receipt entry is created whenever the customer finishes the shopping session, leaves the store and the payment procedure finishes successfully.
- If no product exists, then no cart entry exists.
- If a store worker wants to shop, then he/she must be treated as a customer and he/she must register to the system via the registration interface. Then, the corresponding entry will be created in the customer table. This explanation also applies to the system admins.
- All of the passwords in the system must be stored as SHA512 text for security purposes.
- Database must be backed up every hour.

- Cart information is updated immediately after the sensors catch an interaction between the product and customer. The intermediate information is not held in the cart table. It must only contain the current information. The intermediate information will be held in the log table.
- Once created, log entries are never deleted for legal purposes.

3.6 Design Constraints

- Payment information and passwords of the users must be stored as hashed values for security and privacy.
- Each shopping session must be logged on the remote server. Even the action of a customer that is picking a product and putting it back must be logged; such as when, where and which product he/she picked.
- Each receipt must also be stored on the remote server for legal purposes.
- For security purposes, there must be multiple remote servers in a hierarchy so that each remote server acts as a backup for the one on the higher level.

3.7 Software System Attributes

- **Reliability:** Users' personal information must be kept secure. Sensors must work with very high precision. QR generating and scanning should be quick. Database should be backed up regularly, and there must be multiple copies of the database.
- **Availability:** Mobile application must always be available (except for the maintenance time, which will be announced to users 1 week before hand each time). Store accessibility might be different in different states, but stores must be accessible at 7AM-11PM generally. Authorized users should be able to access the system management any time.
- **Privacy & Security:** Payment information is very critical, it must be transferred with secure protocols and payment method must not be kept at Amazon servers. QR scanning must work with the assistance of GPS in order to prevent QR code thefts. System management must be only accessible at Amazon facilities. Users' passwords must be stored as hashed values, probably with SHA512.
- **Maintainability:** Documentation must be complete and understandable for sustainable maintenance for many years. Using Git is a must to see past changes, and make changes easier on the code.
- **Portability:** Customers will use the mobile application most of the time. Therefore, it shall support the vast majority of the smart phones as possible. It must run on Android 4.1 Jelly Bean and iOS 8.

3.8 Supporting Information

Amazon Go is most certainly a milestone in the sense of a smart store. It is also the very first in its field indeed. Amazon Go is powered by the cutting-edge Computer Vision and Artificial Intelligence algorithms, alongside with the brand-new highly sensitive environmental sensors and cameras. Every information is stored on the Amazon servers, and stored as hashed whenever user privacy is a concern.