AMAZON GO

SOFTWARE DESIGN DESCRIPTION

Middle East Technical University
Software Engineering

NAME	Student ID
Ali Şen	2264661
Şahin Kasap	2264562

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Purpose

Amazon Go is an automated store that doesn't have any lines and checkouts. Having the motto "just grab and go", Amazon Go serves its customers fast and painless shopping experience. Customers use the Amazon Go app to enter the store and take what they want and leave. Amazon Go system serves the customers and workers. The software purposes to make the customers have an error-free, fluent, nice experience.

1.2 Scope

This Software Design Description (SDD) document provides detailed information about the Amazon Go System. It is explained in the architectural design level of the system. It also describes the requirements, functionalities, and necessary definitions of the components in the system. In addition, it is explained how the components of the system and the user and the workers interact with each other and with the system itself, and complete design of the system how the product added, read updated and deleted from the database by the admins and by the system when the necessary condition occurred.

1.3 Stakeholders and their Concerns

Amazon Go is a project based on customers' comfort. There is four types of stakeholders in this project:

Users: Customers who buy stuff and pays via Amazon Go application. Their concerns are being paid automatically for the products which they don't buy and is not being paid by the products for the products they take and try to pay it with extra struggle.

System Developers: System developers develop the programs which lie under the Amazon Go system, these developers will implement the demands so they want to understand all things which are clearly described. Their concern is having no ambiguity in the demands.

Store Staff: Store staff that will add/remove products and helps the customers. Their concerns will have the customers satisfied and have the products in the shelves always fresh, clean, and existing.

IT staff of System: These staff will update the product prices and seek for customer needs, wishes in order to let the store maintain. Their concerns are having a better environment in the store, having a better interface to use the system update the system and to be able to use all the functionalities to solve the problems and errors and keep the system to be maintained easily.

2. References

IEEE standard for information technology--systems design--software design descriptions. (2009). New York, NY: Institute of Electrical and Electronics Engineers.

Other sources:
-Amazon Go main page: https://www.amazon.com/b?ie=UTF8&node=16008589011
-News on Amazon Go: Inside Amazon Go, a Store of the Future https://www.nytimes.com/2018/01/21/technology/inside-amazon-go-a-store-of-the-future.html by Nick Wingfield, January 2018

3. Glossary

Term	Definition
Non-Member User	People that don’t sign up for the Amazon Go app.
Member User	People that signed up for Amazon Amazon Go app.
UI	User Interface
Cameras	Cameras that watch the people inside the store and their interactions with the products in the market.
Sensor Fusion	Sensors that keep track of the products.
Database	A database of products.
QR Code	Unique QR codes that belong to different customers.
Computer Vision	Technology to identify, detect, classify, and measure things.
Deep Learning Algorithm	Algorithm to learn the behaviors in the market.
API	Application programming interface
Shopping cart	The things that customer buys in one entrance and their prices
Friend	The person that enters the store with a customer, which is connected to the shopping cart of the customer.
odtuNumber	The id number of the users in ODTÜ.

Table 1: Glossary

4. Architectural Views

4.1 Context View

In this viewpoint, all use cases of the system are specified and explained in detail. The description tables describe the basic flow of the system in detail including how system should work in different scenarios with an alternative flow as well. Context Diagram shows the general structure of the system, while the Use Case Diagram shows the actors and the system and their interaction with each other.

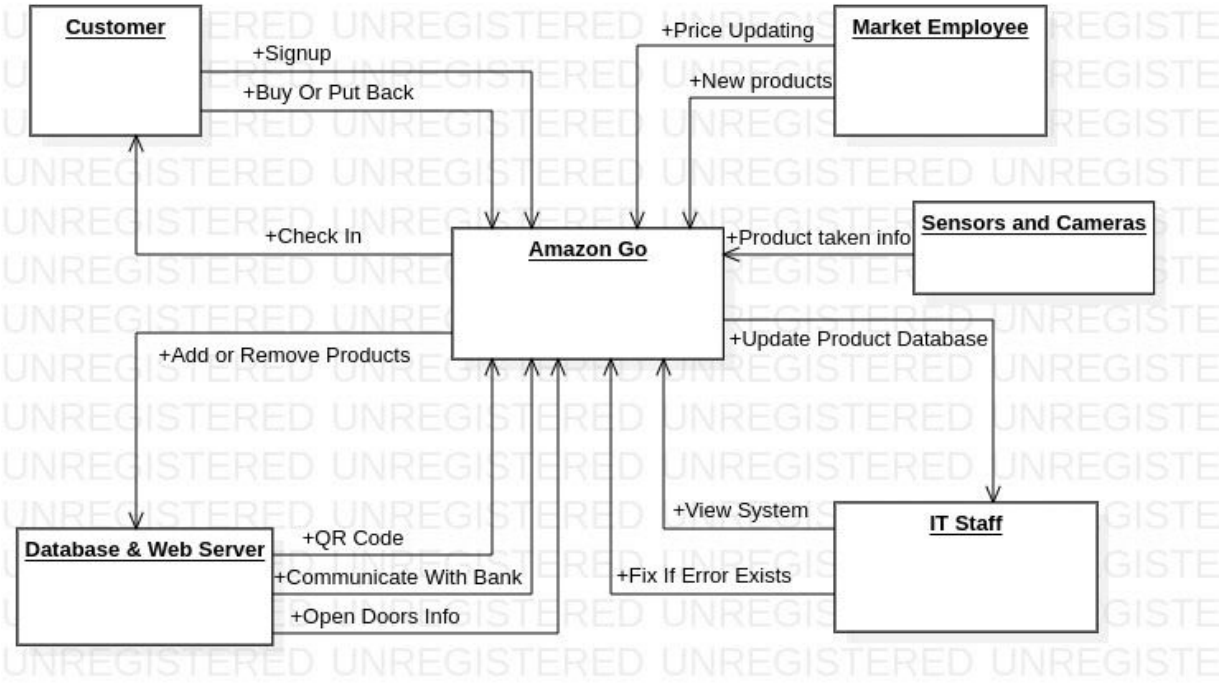

Figure 1: Context Diagram

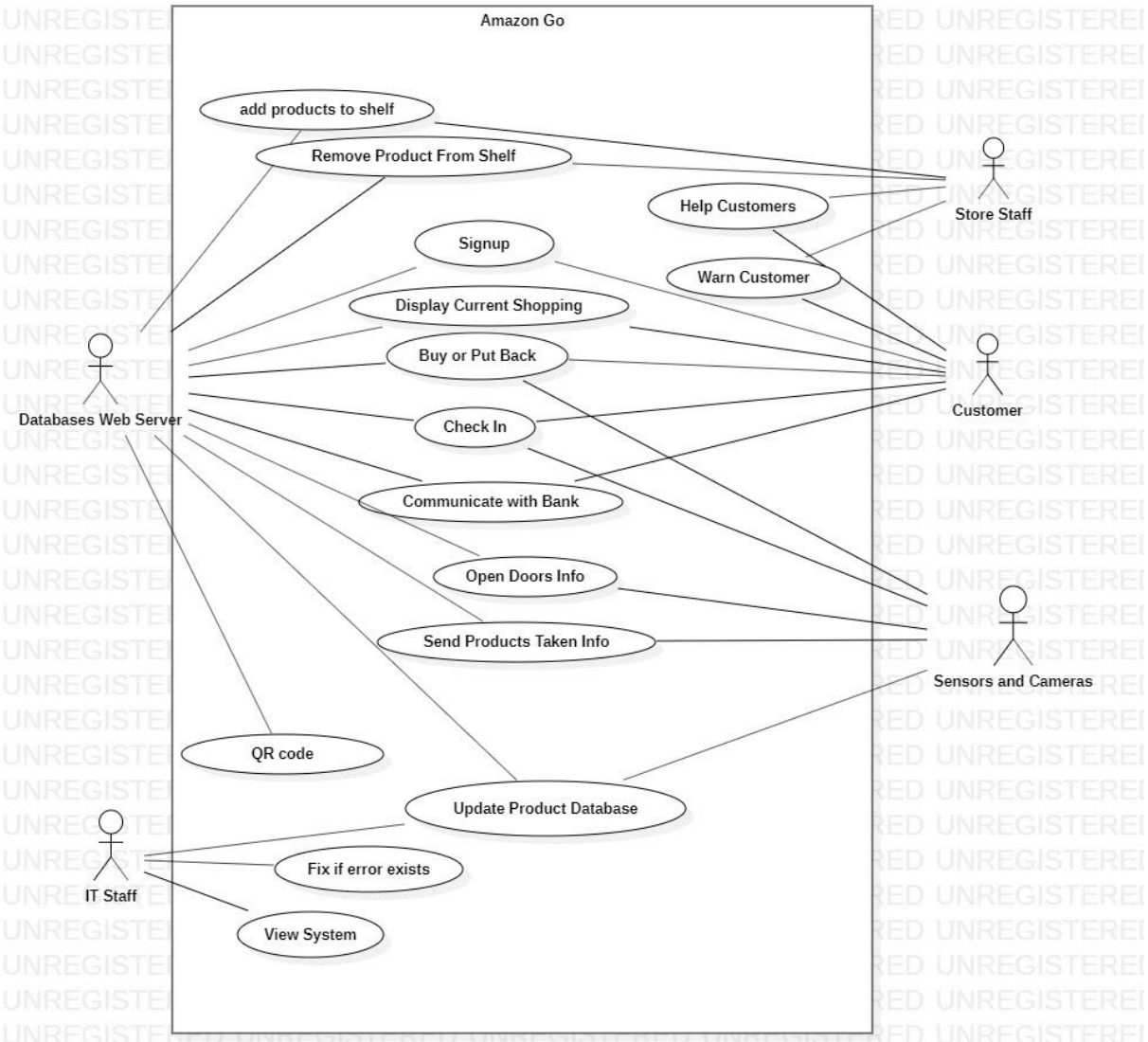

Figure 2: Use Case Diagram

Use case name	Signup
Actors	No-member User and Web Server
Description	No member user uses Amazon to enroll to the system with bank info and name etc important info
Data	Name, email, phone number, bank info, address, picture
Preconditions	Users have to have Amazon app downloaded and have a credit card associated with a bank account
Stimulus	Ads and try it button on Amazon
Basic Flow	Step 1 - User clicks on the ad and gets directed to the Signup page Step 2 - Fills in the required information on the page and continue Step 3 - The user is sent an email to an email account and a message

	to the phone number to confirm.
Alternative Flow	-
Exception Flow	If the user doesn't provide the required information, they get an error message.
Postconditions	User is saved to the database and gets an informative email in order to get used to the situation

Table 2: Signup

Use case name	Buy or Put Back
Actors	Customers, Databases, Sensors, and Cameras
Description	The user takes a product or puts it back.
Data	The view from the cameras and information from the sensors
Preconditions	1-User has to have signed up 2-User has to enter the store.
Stimulus	The good looking of the products, prices and special orders
Basic Flow	Step 1 - User takes the product and puts it somewhere or handles it. Step 2 - Cameras and sensors detect the product being taken using the software
Alternative Flow	Step 1 - User takes a product and puts it somewhere else. Step 2 - Cameras and sensors detect the taken product and the new place it has. Step 3 - The same customer or another customer takes the product. Step 4 - Devices detect the taken thing again.
Exception Flow	-
Postconditions	The user has a bill sent to his/her app.

Table 3:Buy or Put Back

Use case name	Check-In
Actors	User, Web Server
Description	The user reads the QR code in the app to the machines on the entrance of the store and the door gets opened.
Data	The QR code generated by the Web Server
Preconditions	1 - User has to be signed up 2 - User has the app downloaded 3 - The user uses the app to inform the system about his/her entrance.
Stimulus	The ads, cheap and high-quality products
Basic Flow	1-User signs up and has the app downloaded 2-User decides to enter the store and clicks on the "Enter the store"

	button on the app 3-QR code is generated and sent to the app 4-User scans the QR code to the machine and the door gets opened if the QR is as expected.
Alternative Flow	-
Exception Flow	Users may try to enter with another QR or may try to enter by jumping above the entrance. The cameras and staff will inform the system and get the customer out.
Postconditions	-

Table 4:Check-In

Use case name	Help Customers
Actors	Store Staff, Customer
Description	When customers can't use or can't understand the app, store staff will help them to use the app and enter the store.
Data	App
Preconditions	User must come to the store
Stimulus	Shopping is easier on Amazon go.
Basic Flow	1 - Customer comes to the store. 2 - Ask for help from the store staff. 3 - Store staff helps the customer to use the app and enter the store.
Alternative Flow	1 - Customer pass in front of the store. 2 - Customer wonders about the store and asks the store staff. 3 - Store staff inform the customer and customer leaves or want to try the store.
Exception Flow	-
Postconditions	The customer leaves the store with or without the products he/she bought

Table 5:Help Customers

Use case name	Communicate with Bank
Actors	User, Web Server
Description	After a user buys something from the store and leaves the store, the information on what he/she bought and the total price will be sent to the Web Server. Web Server communicates with the bank and withdraws the amount from the bank account using the APIs given by the bank systems.
Data	The products are taken
Preconditions	The user has to be signed up and entered the store.

Stimulus	-
Basic Flow	Step 1-Customer enters to the store Step 2-Customer buys something and his/her virtual shopping cart is updated. Step 3-Customer gets out of the store. Step 4-The web server is informed with the details of the customer. Step 5- The web server communicates with the bank Step 6-Money is withdrawn from the account
Alternative Flow	Cash may be accepted too in some stores.
Exception Flow	-
Postconditions	The user is informed of the products that she/he has taken from the store and the costs of the products.

Table 6:Communicate With Bank

Use case name	Add product to shelf
Actors	Store staff, Databases
Description	When the product on the shelves is decreased store staff gets notified and the staff goes and takes some product from the storage if there is no product in the store too the staff informs the base storage to take some product to the store. When the products come the store staff puts them to the shelves and the databases get updated.
Data	Products
Preconditions	The customers should take the product to decrease the number of the product.
Stimulus	The consumers need the products.
Basic Flow	1 - Store staff takes the product from the storage. 2 - Store staff puts the product to its shelf.
Alternative Flow	-
Exception Flow	There is no product left either in the main storage or in the local storage.
Postconditions	The shelf is filled with the products.

Table 7:Add Product to Shelf

Use case name	Update Product Database
Actors	IT Staff, Database
Description	IT Staff changes the qualifications of the products in case of need, or add new items to the database.
Data	The price, weight, etc qualifications of the existing and new products.
Preconditions	IT Staff must have an admin username and password and the product must exist and should be in the storage for updating purposes. For

	adding new items, products must be bought from other sources.
Stimulus	The changing information, and supply the customer's needs.
Basic Flow	Step 1-IT Staff is informed with the new information that the products have or with a brand new product to be added. Step 2-IT Staff changes the information about the products or add the new item to the database with the quality it has. Step 3-The new qualifications, quantity, and prices are updated for the updated product.
Alternative Flow	-
Exception Flow	-
Postconditions	The product must be available for customers to buy in the store.

Table 8:Update Product Database

Use case name	Open Doors Info
Actors	Web Server, Sensors and Cameras
Description	The entrance of the store has a door and it opens automatically when a customer gets in with a QR and opens again when the customer goes out. These processes will be done with the help of APIs.
Data	QR code, Customer's location information
Preconditions	A customer with a generated QR code or a customer that tries to get out of the store.
Stimulus	Lets customers to get out of the store or get in to store easily.
Basic Flow	Step 1- Customer makes the door to read the QR from his/her mobile phone Step 2- The customer gets in the store when the door opens.
Alternative Flow	Customers get out of the store with just walking out.
Exception Flow	Customers can try to jump over the doors.
Postconditions	Customers get in the store to shop or they are out with the products they bought.

Table 9:Open Doors Info

Use case name	Send Products Taken Info
Actors	Sensors and Cameras, Web Server
Description	When a customer takes a product from any shelf, the cameras and sensors realize this and gives info to the webserver.
Data	Images of customer and product
Preconditions	Customer should be in the store and the products should be in shelves

Stimulus	To know the price of the product
Basic Flow	Step 1- Customer walks in-store and sees the product he/she looks for. Step 2- Customer takes the product
Alternative Flow	-
Exception Flow	-
Postconditions	The product the customer takes is no more on the shelf.

Table 10:Send Products Taken Info

Use case name	QR Code
Actors	Webserver
Description	The web server generates a unique QR code for every member user. And when reading, the web server gets the information of the QR code from the gates of the store and if there is a match with its generated QR code web server understand that to allow customers to enter the store.
Data	Mac address of the mobile phone, QR code
Preconditions	Non-member users should download the Amazon Go app to his/her mobile phone.QR code must be generated and the gates of the store must read it.
Stimulus	To be able to enter to store.
Basic Flow	Step 1- Non-member user downloads the Amazon Go app. Step 2- Non-member user sign ups the app Step 3- Web server generates a QR code for the customer. Step 4- User makes the machine on the entrance read the QR code. Step 5- Web server tells the gate to open the door.
Alternative Flow	Step 1- Non-user members don't know how to get into the store. Step 2- Ask for help from the store staff. Step 3- Store staff help the customer to get in the store by making the app generate a QR code for the customer.
Exception Flow	There may be some erroneous generated QR which doesn't allow the customer to enter the store.
Postconditions	The customer has his/her QR code in his/her mobile phone which allowed him/her to enter the store.

Table 11:QR Code

Use case name	Fix If Error Exists
Actors	System admin
Description	A system admin will be notified or see the errors in the system and fix them by using his/her software skills
Data	Error message

Preconditions	The system's functions must work.
Stimulus	The system should be sustainable
Basic Flow	Step 1- There becomes an error and system admin notified or system admin realized by himself/herself. Step 2- System admin fixes the problem.
Alternative Flow	-
Exception Flow	-
Postconditions	The system works perfectly.

Table 12:Fix If Error Exist

Use case name	View System
Actors	System Admin
Description	System admin view system to see if there is a problem.
Data	Products and their quality information, customers and their location, store staff, and their location.
Preconditions	The system should be working
Stimulus	Sustainability is required for the system to work.
Basic Flow	Step 1- System admin view the system from time to time to see if the system work without any problem
Alternative Flow	-
Exception Flow	-
Postconditions	The system is okay and if there is a problem it is already fixed.

Table 13:View System

Use case name	Remove Product From Shelf
Actors	Store staff, Web server
Description	Store Staff removes product from the shelf.
Data	-
Preconditions	-Product must not be sellable
Stimulus	Selling the most wanted and quality product
Basic Flow	Step 1 - Store staff determines which product to be removed Step 2 - Store staff removes the product from the shelf.
Alternative Flow	

Exception Flow	-
Postconditions	Product is no more in the shelf.

Table 14:Remove Product From Shelf

Use case name	Warn Customer
Actors	Store Staff, Customers
Description	User makes a risky move and staff warns
Data	Store staff views the customers
Preconditions	1-User has to have signed up 2-User has to enter the store.
Stimulus	A wrong move may end up with the customer having to spend money for something they didn't buy.
Basic Flow	Step 1 - User takes a product and gives somebody else Step 2 - Staff sees this and warns, since a product will be paid by the person who takes it first.
Alternative Flow	
Exception Flow	-
Postconditions	User will consider this.

Table 15:Warn Customer

Use case name	Display Current Shopping
Actors	Customers, Databases, Sensors, and Cameras
Description	The shopping is displayed on the user's application on UI.
Data	The products which are taken by the customer.
Preconditions	1-Customer has to enter the store. 2-Customer must take some products.
Stimulus	To give information to the customer about what products he/she take.
Basic Flow	Step 1 - User push the button of the app to show current products. Step 2 - User sees the product that she/he take.
Alternative Flow	-
Exception Flow	-
Postconditions	User will see his/her shopping cart and know what he/she will pay for.

Table 16: Display Current Shopping

4.2 Composition View

This part will show the components of our system from a high-level point.

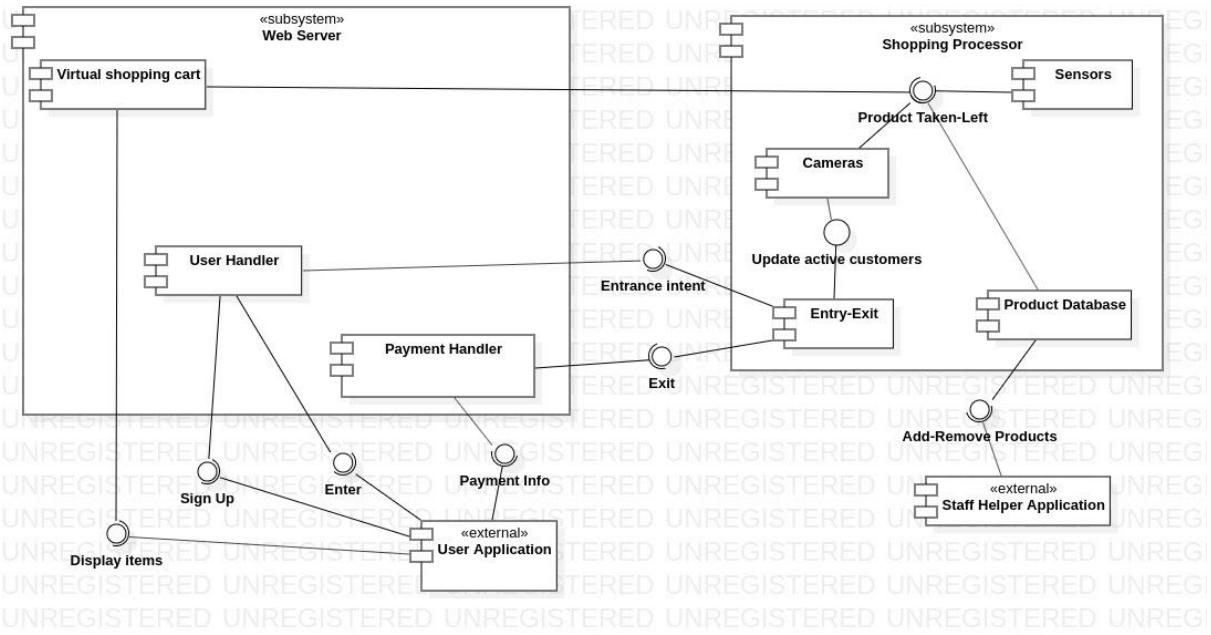

Figure 3: Component Diagram

Design Rationale:

- User applications will be in charge of getting people inside, taking payment info, and displaying the items inside the virtual shopping while and after the shopping. The application will use the interfaces that Virtual shopping cart, User Handler, and Payment Handler components. User applications will be used to sign up too.
- When a user comes to a shopping store, they will enter in different ways. The default way that everybody can use is QR code scanning, but in the later times we can use an NFC reading, or facial recognition in order to let the customers in. User Handler is responsible for signing up and sending entrance intents to the doors.
- Payment Handler will be responsible for sending payment data to the users, which is a virtual bill. Then they will be asked whether the customer thinks there are a problem and the way that they want to pay the bill. In some cases, our image processing system might have some trouble understanding whether a user took an item or not. In this case, the payment handler will ask the customers whether they took the item or not. After that, it will communicate with the bank in order to get the payment.
- Entry-Exit component (which is the door or other access point) will be responsible for getting the intents from the User Handler and inform the Shopping Processor. When a customer leaves the store, the Entry-Exit

component will invoke the Payment Handler so, it will process the users' items and send them the bill.

- Virtual shopping cart component will keep the products that are taken by the customers and will be notified by sensors and cameras if the product that has been taken from shelf put back which will remove the product from the cart which inside the user application.
- Cameras and sensors components are working together to resolve which product is taken by which customer. Sensors keep track of the products and communicate with the camera to identify the customer to put the product to his/her virtual shopping cart. In addition, they keep track of the product until the product is no more in the store to understand the product is put back or not if the product has been put back cameras and sensors communicate and get which customer put the product to remove the product from the customer's virtual shopping cart.
- The product database component will keep the products that the system has, and when the product quantity in the store not enough, the store staff will use the staff helper application component to add the products to the store and will decrease the number from the store of the system. Staff helper application components will also be used by the system admins and IT staff to maintain the system.
- The Shopping Processor Component has different components that are used in order to analyze and detect the movements of the customers. The entry-Exit component will let the cameras know who the new customer is, and then the cameras and sensors will understand who took what. Then when the customer leaves the store, cameras will trigger the Entry-Exit component and this component will start the payment handler.
- Web Server component has different components to communicate with different external systems to make the shopping processor component simpler and make the design more organized to accomplish the task.

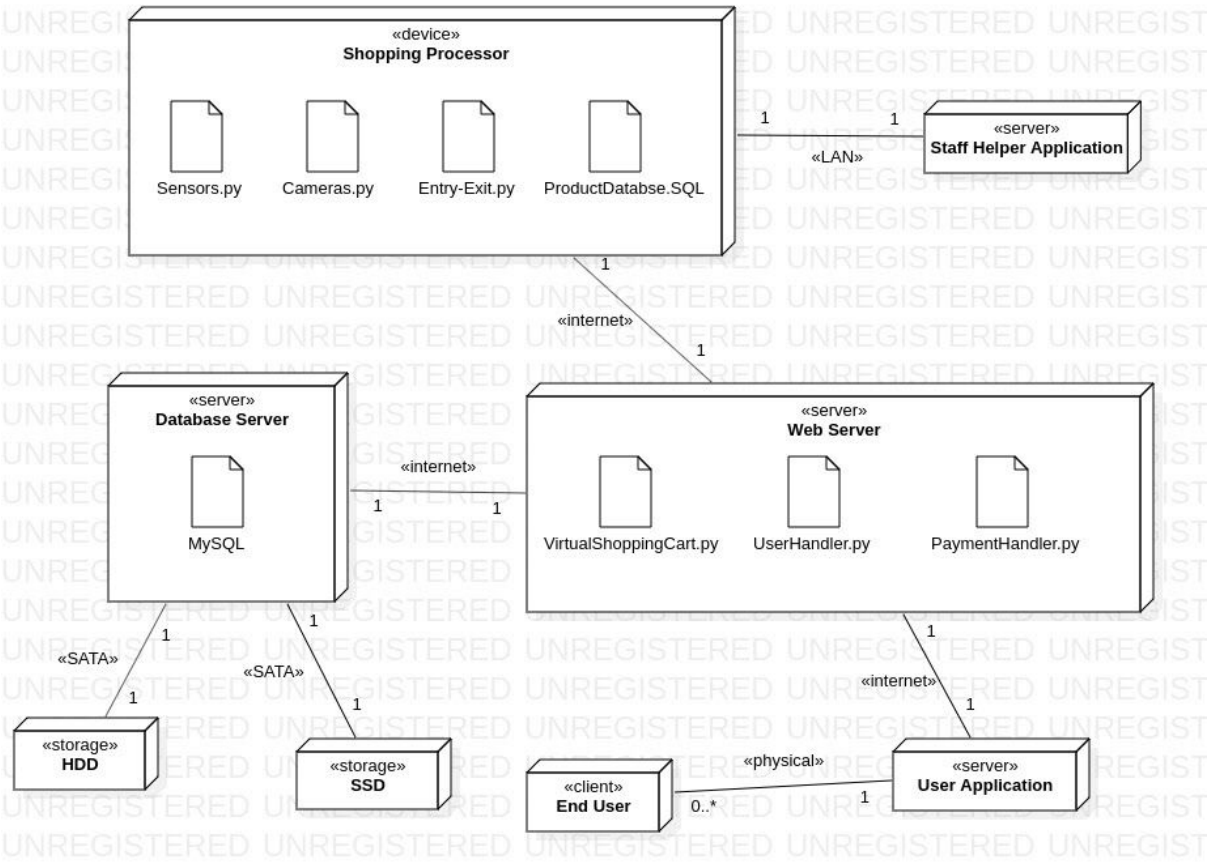

Figure 4: Deployment Diagram

Design Rationale:

- Since we are working with image processing and artificial intelligence, we decided to use Python.
- We use SATA, it is enough for us in any case.
- In the databases, we will store the users and all of their shopping for the last 1 year.
- We use SSD and HDD. In SSD we will store the frequent users, so it needs to be faster, and in HDD we will store more passive users.
- The product database is a local database, we will store the current product list there.
- As a relational database management system, MySQL is used because it is open-source and free.
- Staff Helper Application uses LAN connection to make it reachable by only the store staff. In addition, All internet-based connections will use encrypted communication protocols for increased security (SSL,HTTPS).
- The applications will be programmed in Flutter, in order to make it possible for different devices.

4.3 Information View

In this view, persistent information that is kept in the database and how these map into classes with methods and attributes is shown in a class diagram, along with the relationships of these classes with each other. Create, read, update and delete functionality that is assigned to each database component or class is also specified.

4.3.1 Interfaces

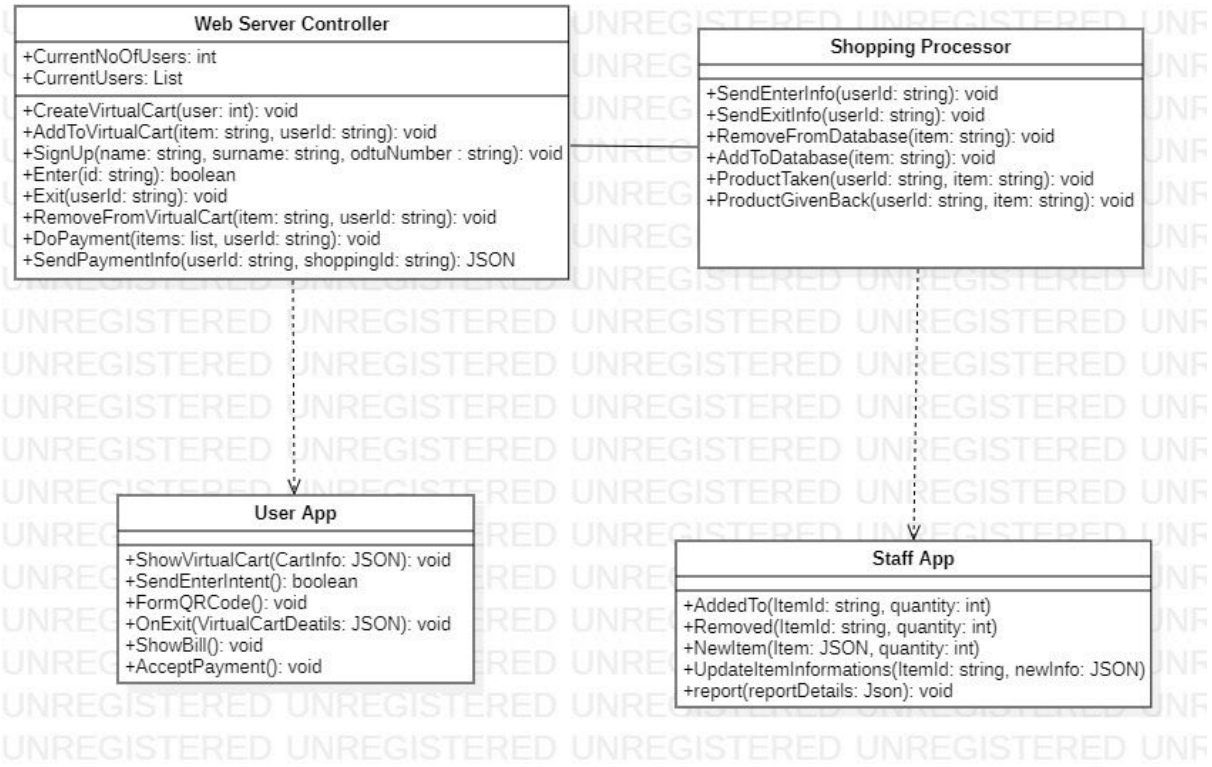

Figure 5: Interface Class Diagram

Operation	**Description**
CreateVirtualCart	Creates a virtual cart for the customer to be able to use in his/her app.
AddToVirtualCart	This operation adds the product taken by the customer to the virtual cart.
SignUp	This operation allows the users to register themselves to the system.
Enter	This operation determines if the customer enters the

	store to shop.
Exit	This operation is called when the customer exits the store and sends needed information to Web Server
RemoveFromVirtualCart	This operation removes the product form the virtual cart which the customer takes before but cancels to buy.
DoPayment	This operation is called when the user accepts doing payment and adjusts the needed things with the Banks, etc.
SendPaymentInfo	This operation sends the payment details to the customer after the automatic payment has done.
SendEnterInfo	This operation is called when somebody enters to the store and matches the person's information with the Web Server. So that web server can set up the virtual cart for the customer
SendExitInfo	This operation is called when a customer leaves the store. The information of the customer is sent to the server and the server finishes the session.
RemoveFromDatabase	The products removed from the database to update the current quantity.
AddToDatabase	New products have been added to the database to update the product database.
ProductTaken	This operation notifies the AddToVirtualCart operation to add the new product to the customer's virtual cart.
ProductGivenBack	This operation notifies the RemoveFromVirtualCart operation to remove the product from the customer's virtual cart.
ShowVirtualCart	This function is called when a session starts and lets the app update the details of the shopping.
SendEnterIntent	When a person wants to enter a store, he/she clicks they want to enter a button, so the application will send the intent of the customer to the webserver.
FormQRCode	This operation generates a QR code to make the customer get into the store.
OnExit	This function adjusts the last things like stopping updating the virtual cart when a customer leaves the store.

ShowBill	Shows the cost of the products and the total price to the customer.
AcceptPayment	Accepts the payment done by the credit card which has been added to the application.
AddedTo	Adds new items to the products database. This takes the product Id and the quantity of the product and updates the total amount after add operation completed.
Removed	When an item is removed from the shelves, this function will be called.
NewItem	Lets the staff add a new type of item to the database and the shelf.
UpdateItemInformations	When an item's information like price changes, this function will be called.
Report	This function lets the staff inform the other staff in order to show a problem they faced.

Table 17 :Operation Description

Operations	**Inputs**	**Outputs**	**Exceptions**
CreateVirtualCart	user	Message operation OK or failed	-Invalid user
AddToVirtualCart	item userId	Message added to a virtual cart	-Product not taken from shelves.
SignUp	name surname odtuNumber	Message sign up operation is successful	-Invalid odtuNumber
Enter	id	void	-The id is not valid or expired -The person did not enter (surrendered maybe)
Exit	userId	void	-The id is not valid or expired
RemoveFromVirtualCart	item userId	Message removed if the operation is successful.	-Not existing item

DoPayment	items userId	Message payment is successful	Lost internet connection cause this operation to fail.
SendPaymentInfo	userId shoppingId	void	-Connection Error
SendEnterInfo	userId	void	-Connection Error
SendExitInfo	userId	void	-Connection Error
RemoveFromDatabase	item	Message: removed successfully.	-Not existing item -Connection Error
AddToDatabase	item	Message: removed successfully.	-Connection Error
ProductTaken	userId item	Message operation OK or failed	-Connection Error
ProductGivenBack	userId item	Message operation OK or failed	-Connection Error
ShowVirtualCart	CartInfo	void	CartInfo is malformed
SendEnterIntent	void	True if the entrance is accepted, false otherwise	-Connection Error -User is blocked
FormQRCode	void	void	-
OnExit	VirtualCartDeatils: JSON	void	Connection Error
ShowBill	void	void	-Connection Error
AcceptPayment	void	True if the process went as expected, false otherwise.	-Card Details Are Wrong -Connection Problem Occurred
AddedTo	ItemId: string quantity: int	void	-ItemId is not found
Removed	ItemId:string	void	-ItemId is not

	quantity: int		found
NewItem	Item: JSON quantity: int	void	-ItemId is used by another one
UpdateItemInformations	ItemId: string newInfo: JSON	void	-ItemId is not found, -Insufficient information
Report	reportDetails:JSON	void	none

Table 18:Operation Design

Design Rationale:

- Web Server Controller is the main controller that is responsible for responding to user interactions on the Amazon Go application. Staff application will use the intranet or Lan connection to make staff interaction not affected by internet traffic.
- Shopping processor doesn't need an internet connection to function, it will be responsible for all of the movements of the customers.
- Staff app will be responsible for informing the shopping processor when staff does something related to the store.
- The website Manager is responsible for querying the database and responding to the users when they take or put back a product.
- User shopping data is kept in the database to show the history of shopping when necessary.

4.3.2 Database Operations

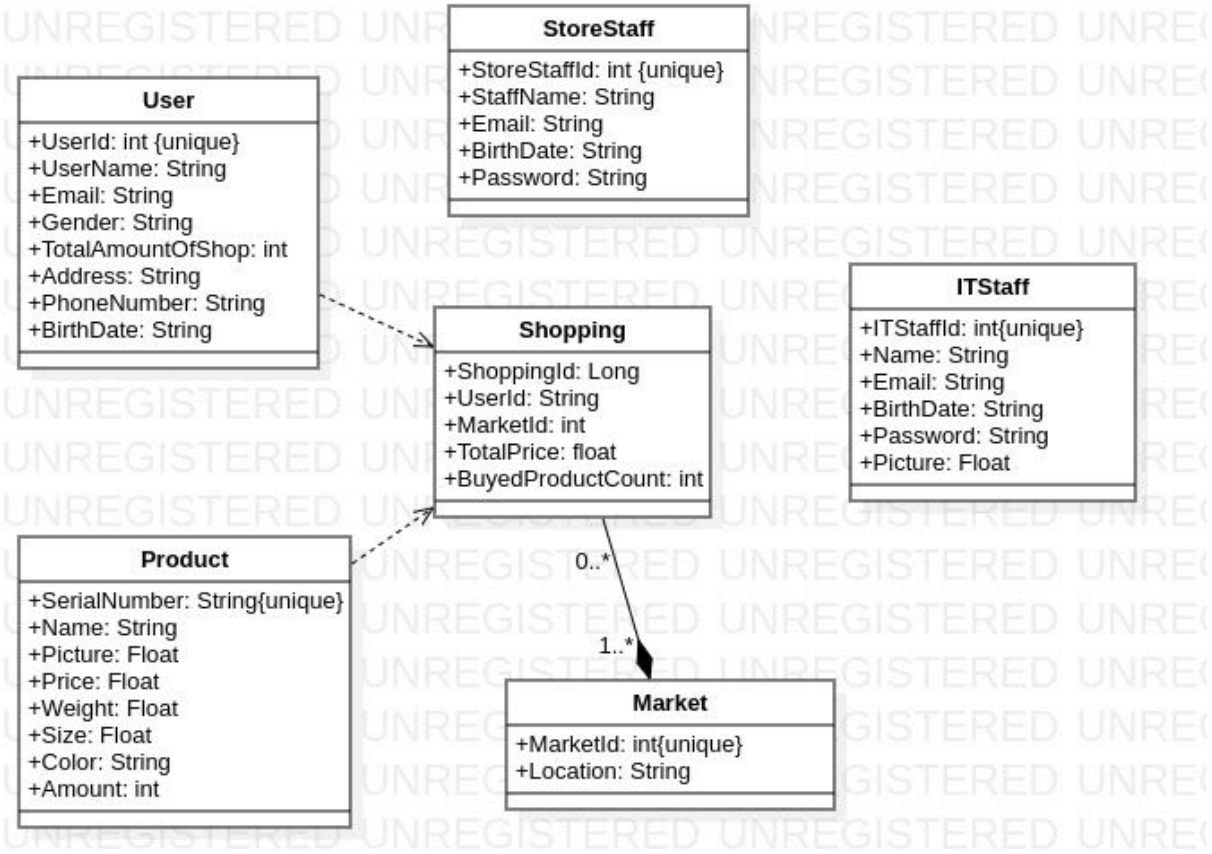

Figure 6: Database Diagram

Operation	CRUD Operations
CreateVirtualCart	Create: Shopping Read: UserId Update: Delete:
AddToVirtualCart	Create: Read:Shopping Update:Shopping Delete:
SignUp	Create: Read: Update: User Delete:
Enter	Create:

	Read: User Update: User Delete:
Exit	Create: Read: Update: Delete: Shopping
RemoveFromVirtualCart	Create: Read: Shopping Update: Shopping Delete:
DoPayment	Create: Read: Market Update: User Delete:
SendPaymentInfo	Create: Read: User, Shopping Update: Delete:
SendEnterInfo	Create: Read: User Update: Delete:
SendExitInfo	Create: Read: User Update: Delete:
RemoveFromDatabase	Create: Read: Update: Product Delete:
AddToDatabase	Create: Product Read: Update: Product Delete:
ProductTaken	Create: Read: User, Product Update: Shopping Delete:
ProductGivenBack	Create: Read:

	Update: Shopping Delete:
ShowVirtualCart	Create: Read: Shopping Update: Delete:
SendEnterIntent	Create: Read: User Update: Delete:
FormQRCode	Create: Read: User Update: Delete:
OnExit	Create: Read: User Update: Delete:
ShowBill	Create: Read: Shopping Update: Delete:
AcceptPayment	Create: Read:User Update: Delete:
AddedTo	Create: Read: Update:Product Delete:
Removed	Create: Read: Update: Delete:Product
NewItem	Create: Product Read: Update: Delete:
UpdateItemInformations	Create: Read: Update: Product

	Delete:
Report	Create: Read: Update: Delete:

Table 19: CRUD Operations

Design Rationale:

- SQL is chosen for the database for its fast and easy usage.
- Shopping is a table for previous shopping that the user made. Current shopping will not be added to this database but will be stored in virtual shopping cart which is a small local database.
- Since this system will be used in different stores, these stores will be saved in market table.
- When a user registers, it is recorded in database.
- The report of the staff will not affect the database.
- When a new item comes, product database will be updated with the amount of new coming products.

4.4 Interface View

In this view, different internal interfaces between components of the Amazon Go system, external interfaces between Amazon Go System and other systems or different types of users, the definitions of these interfaces and use cases they relate to are specified in detail.

4.4.1 Internal Interfaces

The Interface Between the Database Server and the Web Server Controller:
The web server controller will control the database of the users that are registered and their activities. The web server will add the new shopping to the database after every shopping. The activities of the customers will be held in the same database even if the shoppings are done in different stores.

Design Rationale:

- The whole process does not turn around the database, but storing the users is essential to prevent any error and match the real-life users with their information. And it would be better to store the past shoppings of the customers.
- For SQL queries the system will use the web server controller as being responsible.

The Interface Between the Web Server Controller and User App:

The user app will be responsible for interactions between the user and the webserver. Users will be able to register to the system by this app, log in and out, send entrance requests, and see their shoppings. Web Server will supply the data for this app.

Design Rationale:

- The user app has to have easy use. The main page will have the entrance choice.
- The user app will be simple, it will not have many buttons, etc. but will have the choice to add a new payment method, change the information of the user, the choice to show the last shopping.
- Web Server will handle these choices and requests.

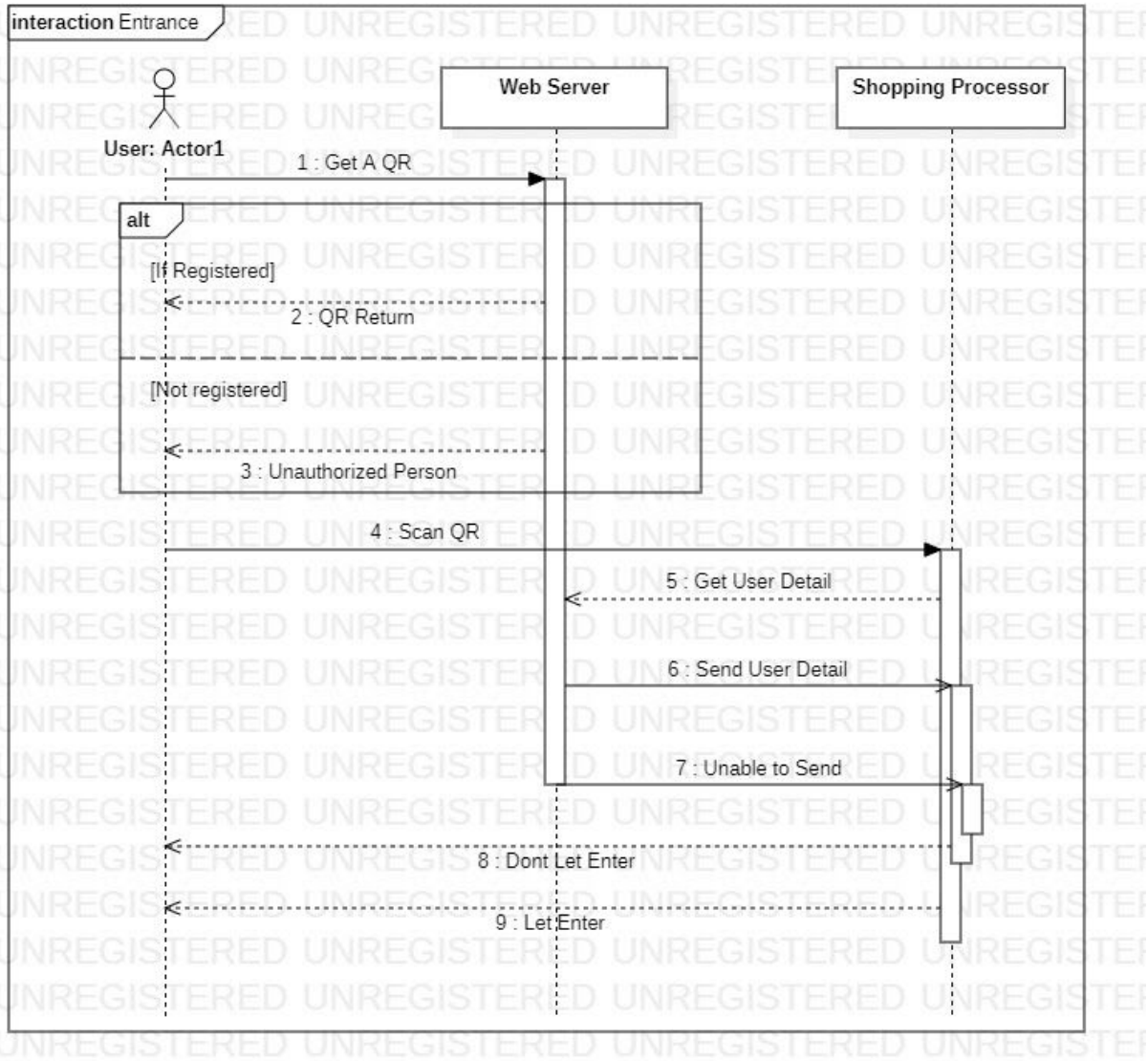

Figure 7: Entrance Sequence Diagram

The Interface Between the Shopping Processor and the Staff App:

Staff application will be used by store staff to add new products to the store when new products are added the Shopping processor will be notified by this app. The product database will be updated.

Design Rationale:

- Store staff will use the Staff App which is downloaded their device by only approval of the IT staff so a random person can't use it to change the data in the store.

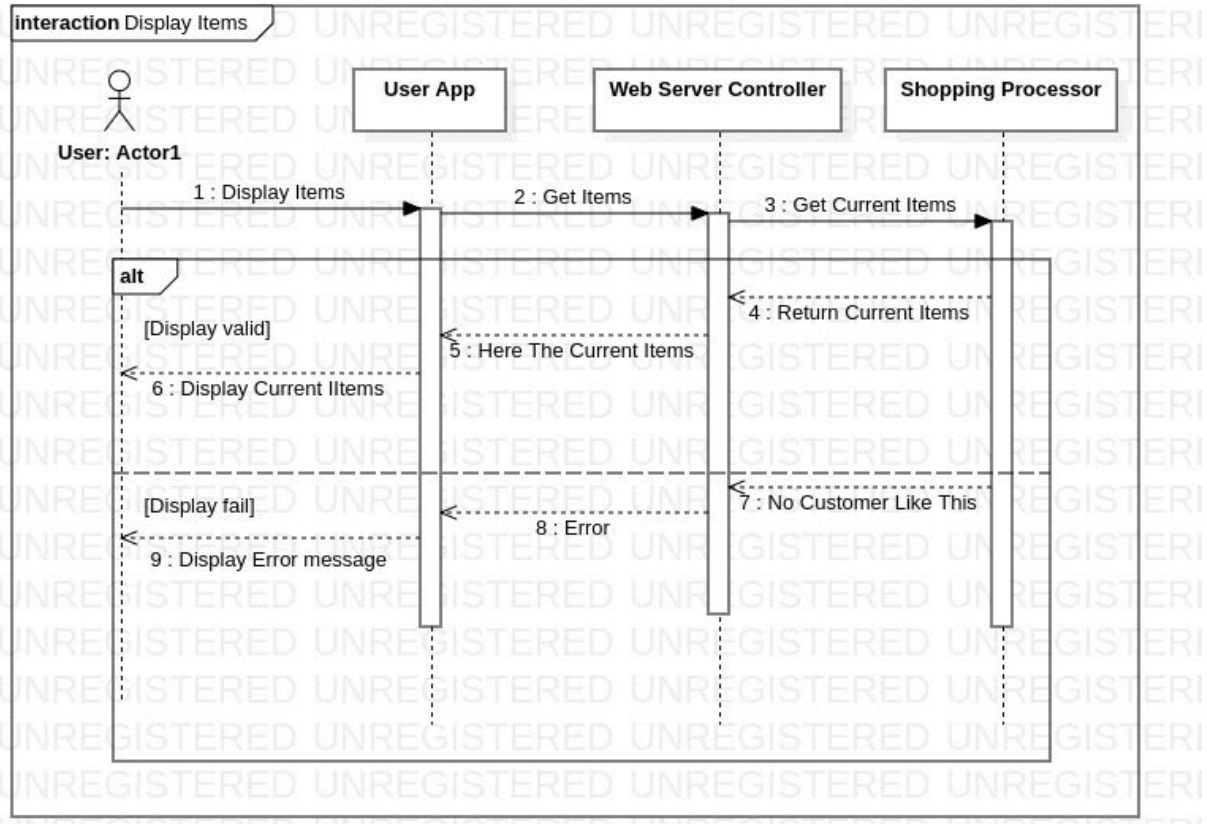

Figure 8: Display Items Sequence Diagram

Figure: Display Items Sequence Diagram

The Interface Between the Web Server Controller And the Shopping Processor:

The shopping processor and the webserver controller will be communicating with each other when a customer ends his/her shopping. So the payment will take place. In addition, when the customer takes a product shopping processor will again communicate with the webserver controller to add the product to the virtual cart.

Design Rationale:

- Shopping processor will communicate only when there is an action on buying an item so the internet traffic will not be crowded other than that the system will use its intranet to accomplish their task like recognizing the item and the person etc.

The Interface Between the Shopping Processor and Product Database:
Product Database is the database to save the product that is currently in the store. Shopping Processor will update this database when customers take stuff and the staff adds products to the shelves.
Design Rationale:

- The shopping processor will not rush to manipulate the product database since it would be hard work to change the database frequently. The shopping processor will change the database when the number of products changed a lot, when their number becomes too low and when the store is getting closed.
- Shopping processor will use the information in the product database to understand the number of elements on the shelves and their grammage, id, price, etc the other information.

4.4.2 External Interfaces

The interfaces that are in the Amazon Go system are the interface for the crowd workers, the interface for the researchers and the interface for the admins, so in other words, the “users” of the Amazon Go system in this context are not the users who ask assistance from the Amazon Go system.

4.4.2.1 User Interfaces

User interfaces are independent of each other. The user app needs to use Web Server to do its job but the staff and IT staff interfaces use Shopping Processor to do something.
Store Staff Interface:
Store staff will use this interface to interact with the product database store staff interface will have basic operations like adding new products and removing products from the database.
Design Rationale:

- The objective of this interface is to have a simple form to use by the store staff to not to complicate operations on the product database.
- The store staff interface will divide the operations into three as adding products, deleting products, and updating product quantity.

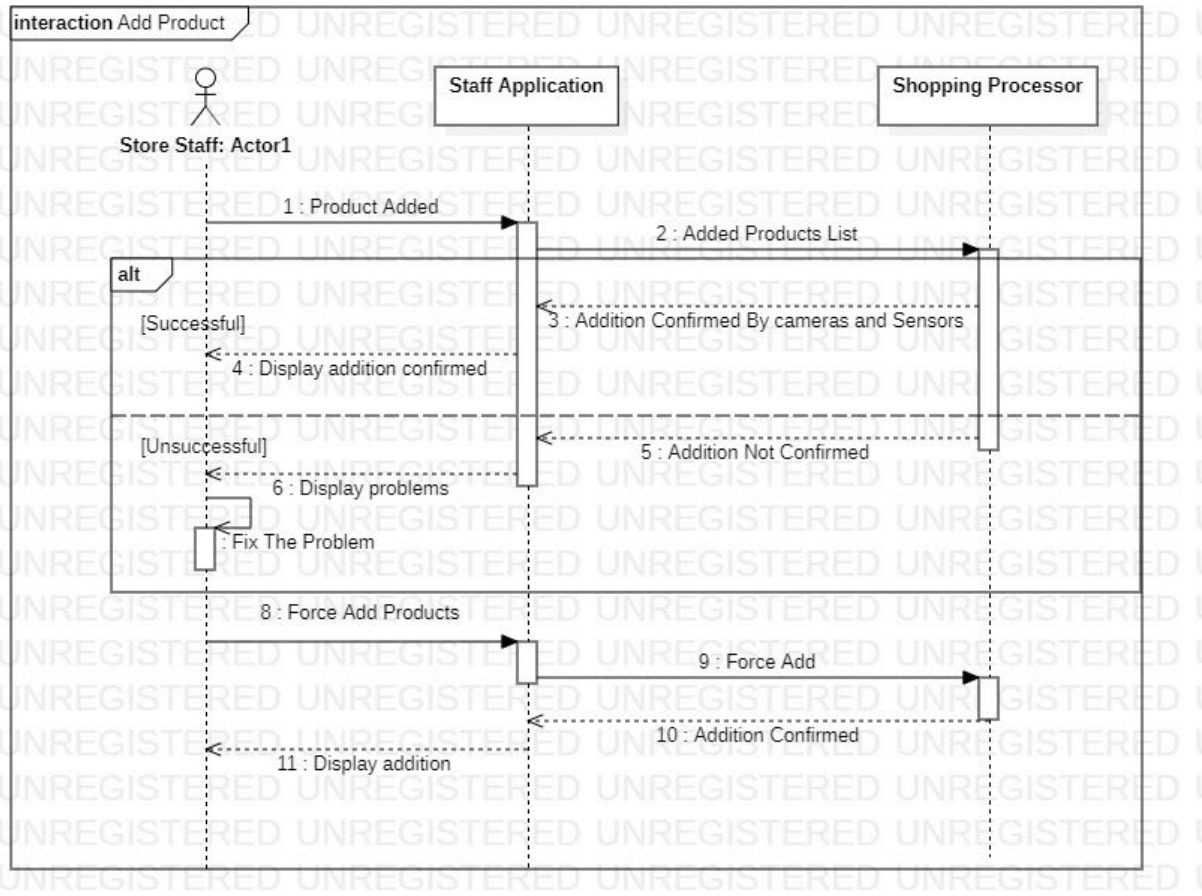

Figure 9: Add Product Sequence Diagram

User Interface:
Users will use this to interact with the webserver and by implication the shopping processor. Users' phones will be used to actually use this interface. The phone is a must for Amazon Go.

Design Rationale:

- Users will be able to enter the system by this app
- In order to let the customers in securely, screen capturing is prohibited in the app.
- We can add extra security like face recognition.
- Users will be able to see their current shopping.
- After the shopping is done, a notification will be sent to the app.

IT Staff Interface:
This interface is only available to the people who have IT staff ID. This is where IT staff can see error logs, and take action to fix them.

Design Rationale:

- Since the IT staff will use this interface the look is not a big issue.
- Reporting a problem will be done via this app.
- Error logs will be seen here.

4.4.2.2 System Interfaces

The Interface between the Web Server and Bank Infrastructure:
Payment will be done by this interface. This interface will calculate the price of the shopping that the customer has made and communicate with the bank to withdraw the amount of money. And in some rare cases, customers may return what they bought and they may want a refund.
Design Rationale:

- Payment handler will be responsible for withdrawing money when a customer with a shopping leaves the store. And sometimes customers may return their items, payment handler will add the money to their account easily.
- The customers may have different types of cards and this interface must handle every situation.
- When a problem occurs, this interface must report it with high importance, since taking money is a huge problem and may damage the customer trust badly.

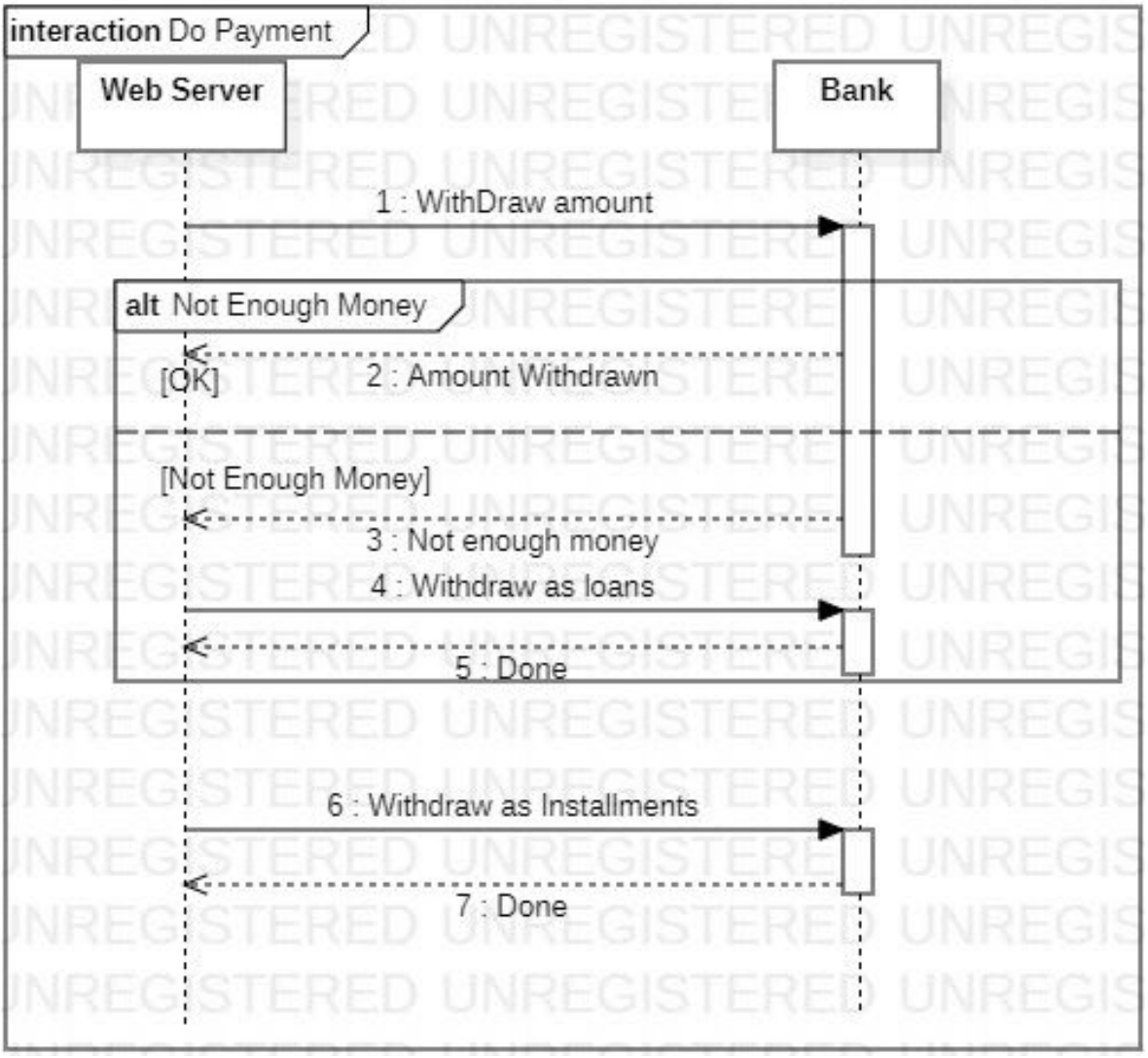

Figure 10: Do Payment Sequence Diagram

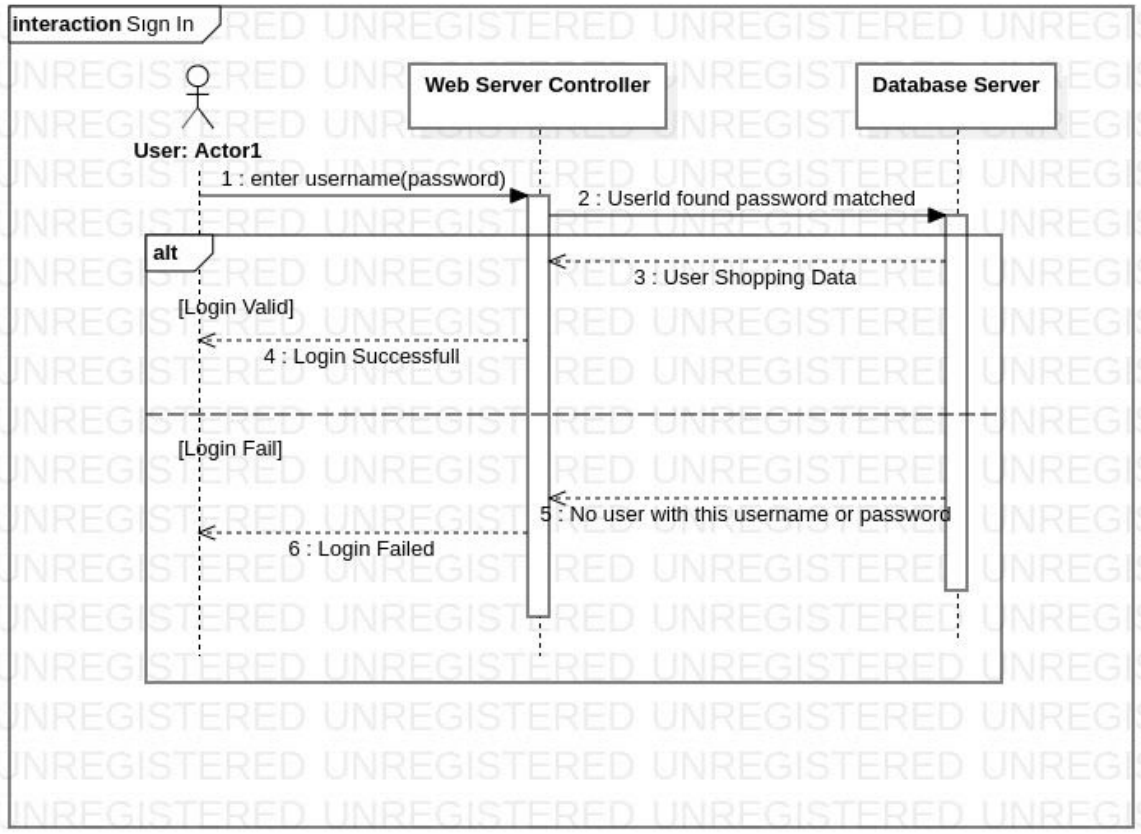

Figure 11: Sign in sequence diagram