1 Introduction

1.1 Purpose

This document focuses on Requirements Analysis and Specification Document (RASD) and contains the description of the main goals, the domain and its representation through some models, the analysis of the scenario with the uses cases that describe them, the list of the most important requirements and specifications that characterize the development of the software described below.

It also includes the research about the interfaces, functional and non-functional requirements and the attributes that distinguish the quality of the system.

This document has the purpose to guide the developer in the realization of the software called CLup, a Customers Line-up application.

Finally, to understand better the development of the document, it contains the history that describes how it is made, with the references used and the description of its structure.

1.2 Scope

The software wants give them the possibility to line up for a super market and notify them when it's their turn.

- **basic function:** Allow application to retrieve a number for users to be in the line waiting for their numbers to be close. Then they will go and stand in the line but in a way that they wont waste their time standing in a queue for long time
- **advanced function 1:** The second functionality point out about booking a visit from application for supermarket which is similar to booking a visit for museums but has some differences. This feature let application for customer to indicate the approximate expected duration of the visit. Alternatively, for long- term customers, this time could be inferred by the system based on an analysis of the previous visits. The application might also allow users to indicate, if not the exact list of items that they intend to purchase, the categories of items that they intend to buy. This would allow the application to plan visits in a finer way, for example allowing more people in the store, if it knows that they are going to buy
- **advanced function 2:** Other feature is that the application might have include a suggestion of alternative slots (in the same day, or in different days) for visiting the store, to balance out the number of people in the store, the suggestion of different stores of the same chain (or even of different chains, if the application is chain- independent) if the preferred one is not available, or the periodic notification of available slots in a day/time range.

1.2.1 World phenomena

WP1	User wants to go shopping
WP2	User get a ticket from manager

1.2.2 Shared phenomena

SP1	Receive a notification that it is user turn for shopping
SP2	Receive a notification that it is better user to approach to shop based on user's location
SP3	Generate QR code
SP4	User choose which category wants to buy
SP5	Manager define his store

SP6	Receive information for booking a visit
SP7	Suggest free slots to user
SP8	Read QR code for entering the store

1.2.3 Goals

G1	Allow user to line up for a specific store
G2	Allow user to book a visit for a specific store
G3	Allow store to generate a ticket for whom have not electronic devices
G4	Estimate the waiting time for each person
G5	Notify user for start to going to store
G6	Notify user when it is their turn to go to store
G7	Suggest people free slots to book a visit
G8	Allow manager to define their store in the system

1.3 Definition, Acronyms, Abbreviations

1.4 Revision history

1.5 Reference Documents

1.6 Document Structure

2 Overall Description

2.1 Product perspective

2.1.1 UML Description

The UML below describes at high-level the model of the system to be developed. It considers the basic service together with the advanced function 1 and advanced function 2 previously specified. The UML does not include every class that will be necessary to define the complete architecture of the system.
CLup more functions than basic service. The manager registers to the application with all necessary information and the manager could activate the advance function or advance function 2 at any time. The user who use mobile could simply download the application on his/her device and use it and user who doesn't have mobile could easily go near the shop and get a ticket from ticket machine.
Here we can identify the main aspect of CLup:

- The user could request to be in line for a shop and application shows estimated waiting time to him/her.
- The user could book a visit for a shop. This booking contains the date and time user wants to go shopping besides, user can add the categories of item he/she has in shopping list. The application could suggest the user free slots and user book them.
- The application based the current location of users must notify them and ask them to approach the shop in a suitable time.
- The application must notify people when it's their turn to go shopping.
- At the entrance time, the QR code generated in the app must be scanned to ensure they come in the right time.
- At the checkout, the cashier must scan the QR code of user and the system must add shopping information (like duration of shopping, category of item which user buy) to user history.

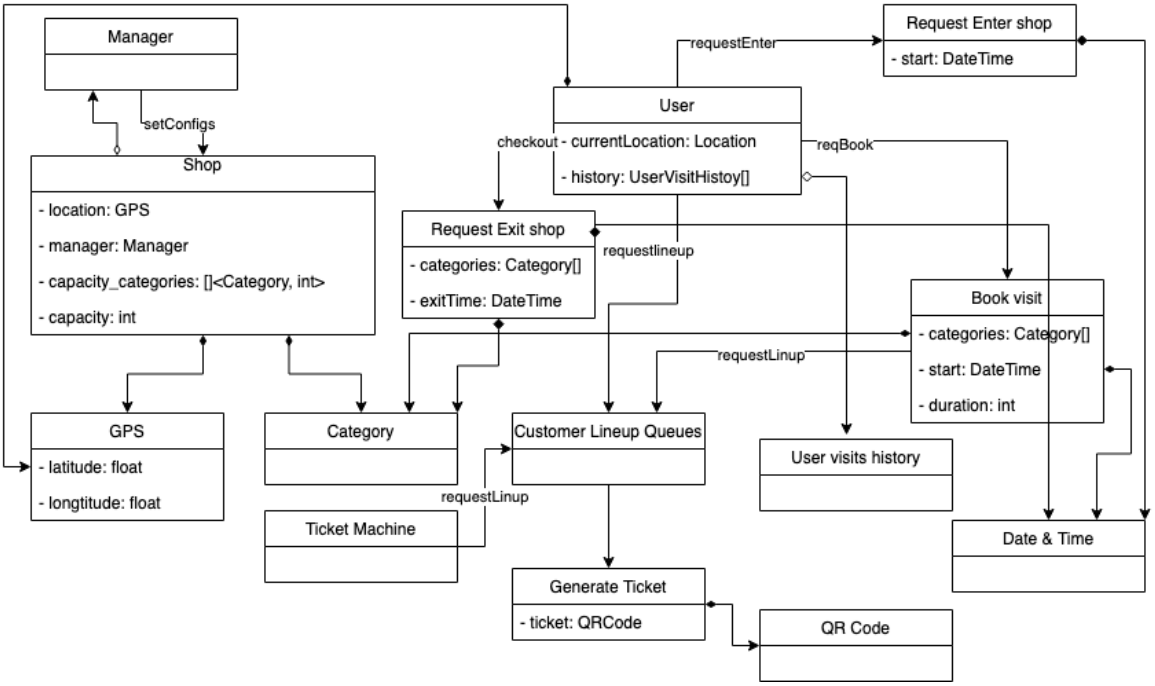

Figure 1. *High level Class Diagram*

2.1.2 State Diagrams

Now we analyse the some important functions of the application, modelling their behaviours and analyze their behaviour to have the expected functionality. we report these diagrams below.

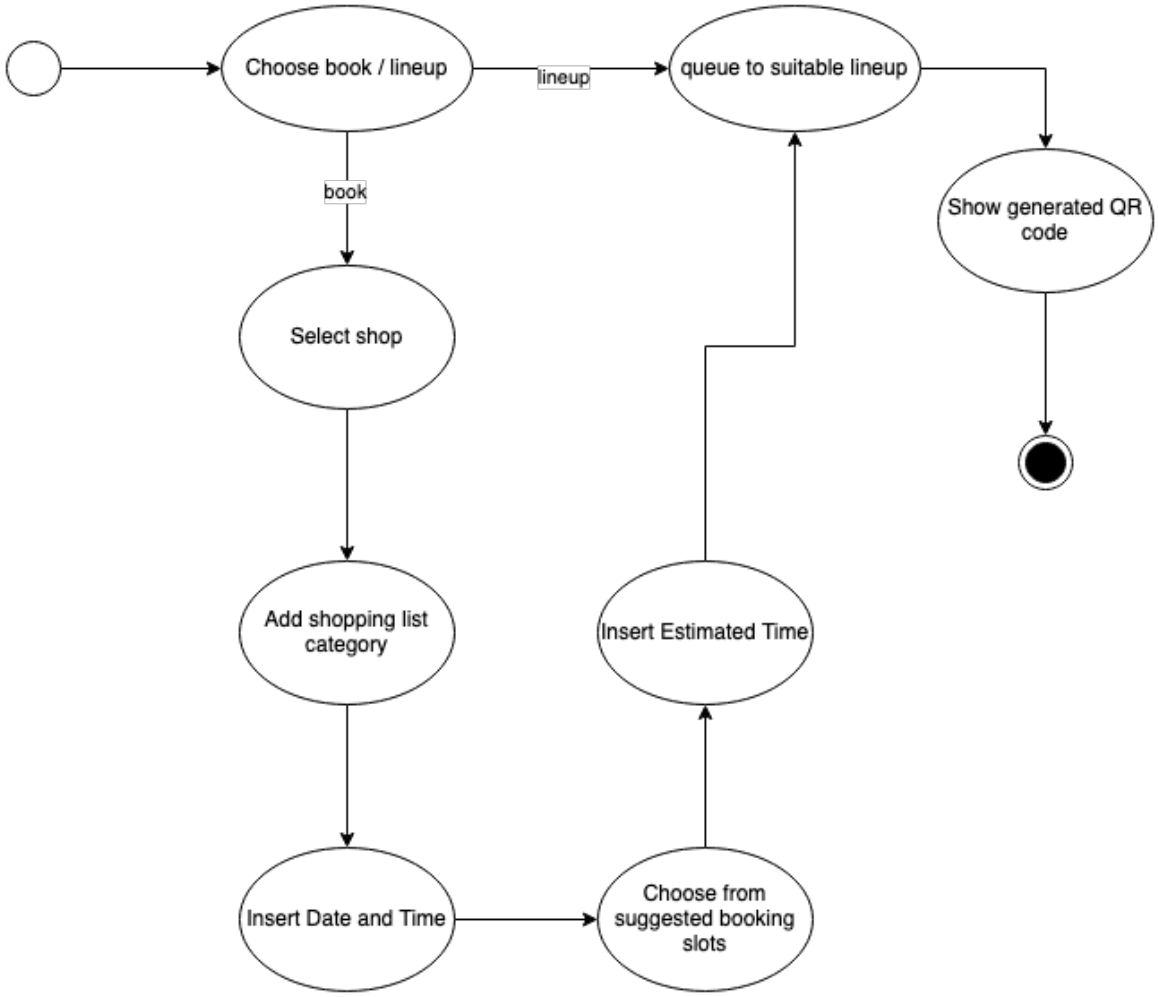

Figure 2. *User book a visit or insert to lineup queue*

In this (Figure 2), we model a user whom has a cell phone and wants to go shopping. As you can see, the user can choose to go to queue or book a visit for a time in future. System could sends him/her list of suggested time and user could choose between them.

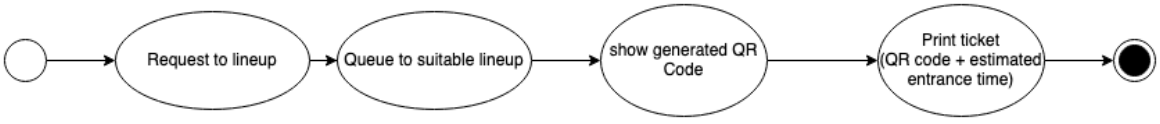

Figure 3. *User get ticket from ticket machine*

In this (Figure 3), we model a user who want to use ticket machine and do not use the app. In this case, user only can add himself/herself to the current line up of the shop.

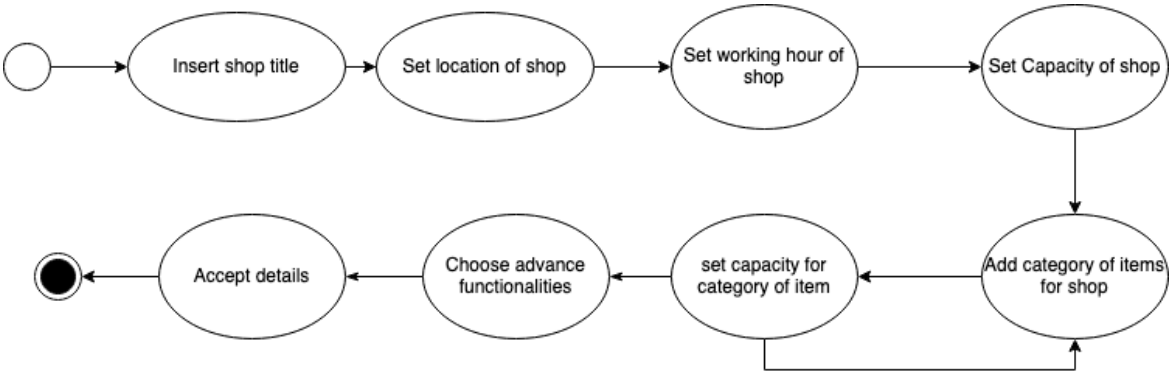

Figure 4. *Manager create a shop*

In this (Figure 4), shows how a manager could create a shop and add necessary information for creating a shop in our system.

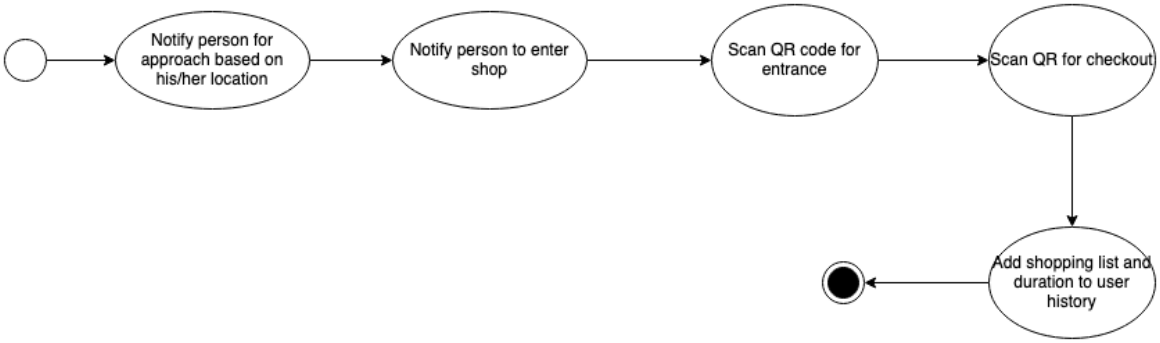

Figure 5. *User shopping*

In this (Figure 5), we model the behaviour of user from entrance to shop until checkout. In the checkout time, we re-scan the QR code of user to insert data user shopping list and duration to user history. we could use, this information to estimate the behaviour of each user and improve waiting time estimation in our system.

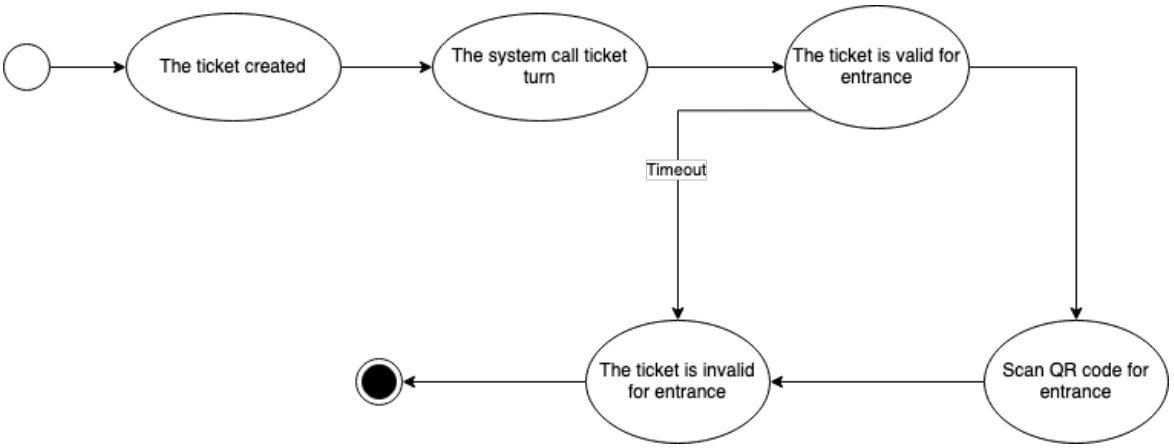

Figure 6. *Ticket states*

In this (Figure 6), the life cycle of a ticket shows. This helps us to understand when a ticket is valid for enter the shop.

2.2 Product functions

1. retrieving a number to line up: the main functionality of the application is the possibility of retrieving a number which gives them their position in the queue. after then it will be easier for customers to access the supermarket without standing in a line for so long. this way forces people to first approach the building and then wait in close proximity (though not in a line) until their number is called.

2. generate QR codes: this feature would be scanned upon entering the store, thus allowing store managers to monitor entrances.

3. estimation process: There is a real risk that the approach does not work in the case the customer arrives to the grocery store after his/her number is called, or too early, as in this case we would get back into a physical line situation. This implies that the system should provide customers with a reasonably precise estimation of the waiting time and should alert them taking into account the time they need to get to the shop from the place they currently are.

4. booking a visit: this function indicate that u can reserve a slot to go to supermarket, its almost like reserving a slot to go to museum or exhibition but they have slight differences like the duration that take to be in a supermarket which we are not able to estimate their time of shopping. so we have to mention a feature which request customer to indicate the approximate expected duration of the visit.

5. system for the customers analysis: there's another option that works for long-term customers in which the time spent by customers is analyzed and the system is gonna predict the average time for that specefic customer based on their pervious visits.

6. indicating the category of items: The application might also allow users to indicate, if not the exact list of items that they intend to purchase, the categories of items that they intend to buy. This would allow the application to plan visits in a finer way, for example allowing more people in the store, if it knows that they are going to buy different things, hence they will occupy different spaces in the store when they visit (thus respecting the requirement that people keep enough distance between them).

2.3 User characteristic

The actors of the application are the following:

1. **user:** customers who download this application to go shopping for the groceries but due to Covid-19 stores and supermarkets shouldn't be full of people and they should follow the social distant rule so without application they used to stand in long lines with social distance but with this application which generate QR codes and with other functionalities, we minimize the time spent by each customer in a line by retrieving a number and send them to be in line if their number is getting closer. also they can book a visit from some days before to go for shopping at supermarkets.

2. **Clerk:** they would scan the QR codes which customers already have by app, and they allow customers to enter. Moreover, stores should also have the possibility to hand out "tickets" on the spot, thus acting as proxies for the customers because maybe some customers do not have the access to the required technology

3. **Manager:** the manager is defining the store on our system and he/she assigns the capacity of either the whole store or each section of the that.

2.4 Assumptions, dependencies and constraints

Domain assumptions:

D1	
D2	
D3	
D4	
D5	

D7	QR code on the ticket of each user must be scan first then let the user to enter the shop.
D8	At the checkout, QR code on the ticket of each user must be scanned.

Constraints:

C1	each ticket is valid for entrance from time the system call it turns until 30 minutes later

3 Specific Requirements

3.1 External Interfaces

3.2 Functional Requirements

3.2.1 List of requirements

R1	Users certified with an authentication
R2	Managers certified with an authentication
R3	Managers should register to the application by a form with mandatory fields
R4	Only managers can create shop
R5	Only managers can update their shop
R6	Managers can activate advance function 1 (allow users booking) on their shop
R7	Managers can activate advance function 2 (send suggestions) on their shop
R8	Users must accept the GPS location for the application
R9	Users could see the list of shops are around them
R10	Users could go to current line-up queue for specific shop
R11	Users could see their ticket (QR code) and estimated waiting time
R12	Users could choose book a time to visit specific shop
R13	Users could choose for which categories they go shopping when they book a visit
R14	The system must generate suitable QR code
R15	The system must estimate the waiting time for each user
R16	The system must make current line-up queue based on users booking
R17	The system sends a notification to user to approach shop based on current location of user and location of shop
R18	The system sends a notification to user to go to the shop when it's him/her turn
R19	After user ticket scanned for entrance, system change status of ticket to invalidate for entrance
R20	The clerk must scan the user ticket
R21	The clerk could add a person to current line-up queue
R22	The clerk could print ticket
R23	After the user checked out, the system analyze shopping list and duration time and add it to the user's visit history
R24	The system could estimate the category of shopping and duration of shopping for specific user
R25	The system estimate the time based on user characteristic
R26	Users can ask a clerk to give them a ticket
R27	If specific time past from user turn to enter his/her ticket will be invalidate for entrance
R28	If managers update a shop the estimated time and book visits will be recalculate
R29	Manager choose the capacity of shop
R30	Manager choose the category of each section in the shop and their capacity
R31	Manager choose working hour of the shop

3.2.2 Mapping

3.2.3 Use cases

3.2.3.1 Use case Description

- **Register**

Name	Register
ID	UC1
Actors	Super User
Entry Condition	Super User has internet connection on his/her device
Event flow	1. Super user see the landing page 2. Super user click on the "sign up" button 3. Super user fill all the mandatory fields 4. Super user choose his type between manager or user 5. Super user fill extra fields based on his type 6. Super user click on "Register" button 7. The system validate the filled data 8. The system confirm the registration 9. The system save the super user on database
Exit Condition	Super user is successfully registered in the system
Exceptions	If the below conditions happened the application return a suitable error in message and return to register page. – Super user is already existed in the system – Super user doesn't fill all mandatory fields – Super user sends invalid data

- **Login**

Name	Login
ID	UC2
Actors	Super User
Entry Condition	Super User has internet connection and it already registered to the application
Event flow	1. Super user see the landing page 2. Super user click on the "sign in" button 3. Super user fill username and password fields 4. Super user click on "Login" button 5. The system check the credential 6. The system confirm the registration 7. The system return the home page of user
Exit Condition	Super user has access to the service of CLup

Exceptions	If the below conditions happened the application return a suitable error in message and return to login page. – Super user is not exist in the system – Super user doesn't fill all mandatory fields – Super user password is invalid

- **Create shop**

Name	**Create Shop**
ID	UC3
Actors	Manager
Entry Condition	Manager has internet connection
Event flow	1. Manager see the home page 2. Manager click on the "add a shop" button 3. Manager set shop title 4. Manager set location of the shop 5. Manager set start and end hour of the shop 6. Manager set capacity of the shop 7. manager click on "new category" 8. manager set capacity and title of category 9. click on "end" button or return to 7. 10. manager choose AF1 or AF2 enable by checkbox. 11. manager click on "Submit" button. 12. The system shows all the detail of shop. 13. manager click on "Accept" button. 14. The system add shop to data base.
Exit Condition	Shop successfully created
Exceptions	If the data of shop is not valid the sytem will return error

- **Update shop**

Name	**Create Shop**
ID	UC4
Actors	Manager
Entry Condition	Manager has internet connection and the shop is exists

Event flow	1. Manager see the home page 2. Manager click on the ”edit” button in specific shop 3. Manager change fields it like 4. manager click on ”Submit” button. 5. The system shows all the detail of shop. 6. manager click on ”Accept” button. 7. The system update the shop in data base. 8. The system update recalculate the estimated time by new values.
Exit Condition	Shop successfully updated
Exceptions	If the below conditions happened the application return a suitable error. – Shop is not existed in the system. – Manager doesn't have access to change that shop. – Fields values are invalid.

- **See list of shops**

Name	**See list of shops**
ID	UC5
Actors	User
Entry Condition	User has internet connection and must accept GPS permission
Event flow	1. User get latitude and longitude from GPS 2. User send a request to get list of shops 3. The system extract latitude and longitude from user request 4. The system get shop list from database 5. The system order shops based on latitude and longitude and characteristic of user. 6. The system shows the list of shops to user.
Exit Condition	User see list of shops
Exceptions	None

- **Generate ticket**

Name	**Generate ticket**
ID	UC6
Actors	User, Clerk
Entry Condition	user or clerk wants to add to a line-up queue

Event flow	1. The system extract the user turn. 2. The system extract the user id. 3. The system extract the shop id. 4. The system extract the line-up queue id. 5. The system generate a QR code based on those information. 6. The system saves generated QR code to database. 7. The system shows the generated QR code to user.
Exit Condition	QR code generated successfully
Exceptions	None

- **Estimate waiting time**

Name	**Estimate waiting time**
ID	UC7
Actors	User, Clerk
Entry Condition	user or clerk wants to add to a line-up queue
Event flow	1. The system extract the users from line-up queue. 2. The system extract the shop. 3. The system extract the new user wants to add in line up. 4. The system get history of users shopping from database. 5. The system extract new user wants shop from which categories. 6. based on previous behavior of user and categories system estimate shopping duration of user. 7. based on previous users in queue, system estimate waiting time for user.
Exit Condition	The system estimate the user waiting time
Exceptions	Check the shops, users and line-up queue are really exists. if they are not exist send a proper message.

- **Add to Line-up Queue**

Name	**Add to Line-up Queue**
ID	UC8
Actors	User, Clerk
Entry Condition	user choose specific shop to go to one of it lines up queue or user book a visit, clerk wants to add person do not use application to a line-up queue

Event flow	1. User choose a shop to add line up or book a visit for that shop or clerk click on add to line-up. 2. The system find that line-up queue 3. The system get the generate number of user in queue. 4. The system pass user, shop and queue to [UC6]. 5. The system get the QR code of ticket. 6. The system pass user, shop and queue to [UC7]. 7. The system get the estimation waiting time for user. 8. The system get current location of user. 9. The system get calculate traveling time estimation for user. 10. The system set interval to send notification for approach a user. 11. The system add user to correct line-up queue. 12. The system send ticket and estimated time to user.
Exit Condition	add user to line-up queue
Exceptions	None

3.2.3.2 Use case Diagram