**1.INTRODUCTION**

**1.1**  **Purpose**

The purpose of the document is to provide a description of CLup application. The idea of developing this application starts from the need to manage the shopping during coronavirus emergency, with the aim to avoid lines in front of stores which creates crowds.

The application is meant to be used by shop managers to regulate the influx of people in the buildings, also through the scanning of QR codes at the entrance, saving the customers from having to stand outside of stores with other people.

The application is also meant to be used by the customers to avoid having to line up outside the building, through a system that gives them the position in a queue through the retrievement of a number. In this way, they should wait until their number is called, or close to being called, to approach the store. For an effective success of the application, the system should provide the customers a reasonably precise estimation of the waiting time

People who do not have access to the required technologies, can hand out tickets on the spot.

CLup provides also the possibility to “book” a visit: either customer can indicate the approximate expected duration of the visit, or in case of long-term customers, this time could be calculated by the system through the analysis of their previous visits.

In addition, to allow more people in the store, or in general, to manage in a better way the affluence of people, the users can indicate the categories of items they intend to buy.

**1.2**  **Scope**

The scope of the application is to allow store managers to regulate the number of people in the building, in order to prevent situations of gathering, that can be dangerous during the period of coronavirus emergency.

On the other side, the application allows customers to do the shopping in a safer way both avoiding them queuing in front of the building with other people and letting them do the shopping (CAMBIAAA) keeping distances from the other customers. Specifically, according to the international rules, the distance between two people must be at least one meter. For this purpose, CLup is able to organize the entrances and the bookings according to the capacity of the building, which is provided by the shop manager during its registration.

To do the shopping, the customer already registered must login the application to take the “ticket” obtaining a position in the queue and the estimation of the waiting time.

In order to make the lining up mechanism effective, either the customer activates the localization, and the system calculates the time needed to get to the shop, or the customer estimates by himself/herself the time required from the place he/she is. And also the customer has to scan the QR code generated with the ticket when arriving/leaving to/from the store to improve the efficiency of the system.

The users can also exploit the advanced functionality of “booking a visit”, indicating an approximate duration of the visit they intend to do. The system can infer this time for long-term customers by analyzing their previous visits.

The customers have also the possibility to indicate which kind of items they think they will buy, to allow the system to better organize the entrances by predicting which spaces will be fully occupied in the store and those who have not reached their maximum capacity. Also in this case the customers have to scan the QR code at the entrances/exits of the shop.

CLup is very simple to use because it includes all demographics, and to take into account visits from people who cannot use the application, the system provides the fallback option of handing out tickets on the spot through the presence of one or more totems positioned around the building.

**1.2.1 World phenomena**

-store managers divide shops in departments

-store managers organize the buildings in order to avoid gathering due to coronavirus pandemic

-user wants to do shopping

-users have a smartphone

-people maintain a one-meter distance between each other

-shops put totem at their entrance to allow people to take tickets on the spot

-people arrive in front of the shop without queuing

-shops put QR scanners at the entrances/exits of the shop

**1.2.2 Shared phenomena**

-user takes the ticket online

-user takes the ticket on the spot through the totem

-user books a visit

-user observes when it is his/her turn

-system assign a position in the queue for each customer

-System analyzes the previous visits of long-terms customers

-application provides the user the estimation of the waiting time

-shops have a maximum capacity

-the application sends a notification to the user when he/she has to leave to arrive in time at the shop

-The day of the booked visit the application reminds it to the customer  through a notification

- user provides to the application the estimated time necessary to reach the shop

- the application uses users GPS to provide the estimated time necessary to teach the shop

-user provides information to the system about the categories of items he/she wants to by

-user scans the QR code when he enters/leaves the shop

**1.2.3 Goals**

-allow people also to take the ticket on the spot

-contrast coronavirus pandemic expansion

-allow people to maintain distance rules while they do shopping

-allow store managers to organize in a more efficient way the store

-allow people to avoid lines in front of the shop

- ? allow people to take a ticket online to do shopping (one shared+one world)

- ? allow people to book a visit online to do shopping (one shared+one world)

**1.3 Definitions, Acronyms, Abbreviations**

**1.3.1 Definitions**

* **Customer**: who signs in the application with the aim to do the shopping.
* **Store Manager**: who provides the application the information about the store and has the purpose of organizing it according to the new rules introduced in order to contrast the coronavirus pandemic.
* **User**: who signs in the application and uses the available services for him/her purposes. The user can be a customer or a store manager.
* **Demographic**: particular sector of population (children, seniors, adults, …)
* **Ticket**: the number received corresponding to the position in the queue.
* **Totem:** multimedial structurewhich allows people who cannot use the application to take a ticket and provides it to them
* **QR Scanner:** digital structure where people have to scan their ticket in order to make the mechanism more efficient
* **QR Code:** bidimensional matrix composed by black modules put in a square schema used to memorize information about a ticket

**1.3.2 Acronym**

§  **QR code:** Quick Response Code

**GPS:** Global Positioning System

**RASD**: Requirement Analysis and Specification Document

**API:** Application Programming Interface

**UI:** user interface

**GDPR:** General Data Protection Regulation

**1.3.3 Abbreviations**

* **WPn**: World Phenomenon number n
* **SPn**: Shared Phenomenon number n
* **Gn**: goal number n
* **AF**: advanced function
* **Rn**: requirement number n
* **Dn**: domain assumptions number n

**1.4 Revision History**

**1.5 Reference Documents**

* Specification Document: **“**R&DD Assignment A.Y. 2020-2021.pdf”
* IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications
* Available slides on beep

**1.6 Document Structure**

* **Chapter 1:** it describes what the scope of the software is, identifying the product, the application domain and the boundaries between the application domain and the external environment.

It sets the goals and explains what is included in the following sections, to guide the readers in the reading process.

**2. Overall description**

**2.1 Product Prospective**

**2.1.1 UML Description**

The UML below shows the requirements, the interfaces of the machine and the interaction between the machine and the world.

It does not contain all the classes useful to describe the complete architecture of the system, it only contains the most significant

CLup is going to be used as a mobile application, and for this reason both the customers and the shop managers will download it on their devices to exploit the services provided by the application.

For this scope they need to be registered and based on their role, different kind of information are requested to them.

·       The shop manager registers his/her shop giving information about:

-        The name and address of the shop (mandatory)

-        Some pictures and additional information of the shop (optional)

-        The departments present in the building with the related capacity and the categories of items that can be find inside

-        The position of the totem where people can take the ticket if they don’t have access to the application

- The scan of the certificate that validates his/her role in the shop

·       The customers have to provide personal information and, if they accept, also their localization.

The UML highlights both the basic service of taking a ticket online in order to do shopping in a safer way and the advanced functionality of booking a visit for the following days.

RICOPIARE DISEGNO UML DA FOGLIO

 INSERT HERE DESCRIPTION OF SCHEDULE AND DB

When the customer already registered and logged in to CLup opens the application, a list of shops divided by categories is shown to him/her. In case of localization provided, this list of shops is sorted from the nearest one.

For each shop in the list open at the moment in which the user enters the application, the number of people already present in its queue is also provided. If the queue is full until the closing time of the shop the system does not allow the customer to take a ticket for the shop.

So, the customer can decide which shop to select to take a ticket.

First of all, if the GPS localization has not been provided yet, the application asks the user if he/she wants to give his/her position by writing his/her actual address (providing the city, the street and the number).

In case of position provided either by GPS or by the customer him/herself, the application computes the necessary time for moving from the provided position to the one of the chosen shop and shows that to the customer. To allow him/her to arrive on time, CLup also sends periodic notifications to him/her.

Otherwise, the customer must estimate by himself/herself the time required to reach the destination.

 Then, the system assigns the ticket to the user and generates a QR code used to monitor the entrances and the exits of him/her from/to the store.

When the customer enters the shop and leaves it scans the QR code at the entrance/exit of the building.

The system saves in the database  this real time data in order to calculate better estimations for the schedule.

The ticket, with the related user, is saved in a schedule that contains all the tickets of the current day and of the following ones (obtained by “book a visit” functionality).

The ticket contains the date and the waiting time, which is computed by the system by using the schedule based on the average duration of customers’ permanence in the shop on the same day of the previous week. For each customer the permanence durations in the shop are saved in the database.

 (asterisco a fine pagine)The duration is calculated through the difference between the exit time and the entry time obtained by the QR code.

The customer who wants to exploit the advanced functionality “book a visit” for the following days, can select the option once logged in to the application and if he/she wants, he/she can provide the category of items he/she has intention to buy and an approximate expected duration of the visit. Alternatively, for long term customers, the system can compute the duration by analyzing the data of their previous visits.

Instead, for new customers who do not provide the duration of the visit, the system uses the average duration of the visits of all the other users of the same day of the previous week.

**2.1.2 State Charts**

**FARE UNO STATE CHART PER LA CANCELLAZIONE DELLA PRENOTAZIONE**

In this section we give an abstract description of the behavior of the system which is represented by a series of events that can occur in the possible states.

 FIGURE 1

The first two state diagrams describe the behavior of the system when a ticket request is received.

The state diagram in figure 1 represents the first part of the behavior of the system when the ticket is requested by a user.

Once the request is received, the system asks the user for permission to know the actual position. The user can accept or refuse.

If the user accepts the system can either receive the GPS localization or the information of the address directly written by the user.

In both cases the system computes the time necessary for the user to arrive to the destination and notifies it to him/her.

Either if the user accepts or refuses the position request, the ‘Information position acquired’ state is reached. This means that either the user provided his/her position or that he/she refused to give it. This leads to the final state.

 FIGURE 2

The state diagram in figure 2 models the behavior of the system from the point in which the information of the position is acquired.

The system, based on the availability of the day (schedule analysis), decides whether to accept the request, or refuse it directly going to the final state.

In the first case the system estimates the waiting time, sends it to the user and once saved the request, reaches the final state too.

  FIGURE 3

The state diagram in figure 3 models the case in which the user decides to exploit the advanced functionality of booking a visit.

The system, once received the AF request, asks the user to provide the date and time of the visit and, once received them, the request of the category of items the user has intention to buy is sent. He/she can decide whether to provide this information – which is saved by the system – or to skip this option.

In both cases the approximate expected duration of the visit is requested. Also in this case the user can decide whether to provide it to the application or not.

In the first case the information is saved in the schedule and the final state is reached.

In the second case, the system checks if the current user is a long term one and if so, it analyzes the previous data in order to come up with an estimated permanence in the shop and saves it in the schedule. Then the system gets to the final state, which is also directly reached if the user is not a long-term customer.

SHOP MANAGER DIAGRAM????

**2.2 Product Functions**

This section highlights the major functions of the software underlining what is able to offer. Here you can also find some of  the requirements which are better explained in the following section “Specific Requirement”.

**Take a ticket**

It is the main functionality of the system.

It gives to the user the possibility of taking a ticket, which corresponds to the position  in the queue identified by a progressive number. With the ticket, the user also receives the QR code he/she has to scan at the entrance/exit of the shop.

Through the list of shops and the number of people already present in the queue of a shop, the user can choose the best option for his/her purposes, take its position in the queue and move from the shop when it is almost his/her turn, also thanks to the periodic notifications sent by the system.

If a customer has already taken a ticket, and so he/she has already a position in a queue,  he/she cannot take another ticket but he has the possibility to book a visit for a following day.

In any moment the user has the opportunity to check the state of the queue and to know about the remaining waiting time.

So the user selects the shop based on how many people are already in the corresponding queue, which categories of items the store provides, the position and the opening and closing time of the store.

After the ticket request:

* If the user during the registration(or in any other moment through the application settings) has given access  for the GPS localization, the system calculates and provides him/her the estimation of the time to reach the destination.
* The user who has not provided the GPS localization, is asked to write the address  (city, street, civic number) in order to receive the estimated time by the system.
* The user can also skip the localization request without giving any information about his/her position and so he/she has to calculate by him/herself the necessary time.

In any moment the customer can cancel the ticket and quit the queue he/she is positioned in. In this way the following tickets move up one position in the queue.

**Book a visit**  
The user has also the possibility to “book a visit” for the following days.   
Once selected the shop, the system asks him/her to provide the category of items he/she has intention to buy and an estimated approximate time of the visit. The user also in this case can skip the request and directly select the day and the time slot of the visit between the available ones shown by the application. At this point the user receives the ticket and the related QR code.

The user has the possibility to cancel his/her reservation until one hour before the start of the slot time selected.

The system sends a notification to the user two hours before the visit to remind him/her the reservation made.

**Register a shop**

When the shop manager decides to use the service provided by Clup, register himself as “Store manager”. With this kind of registration the system requires:  
-Personal information of the shop manager  
-Departments present in the building  
-Capacity (number of people allowed) in each department  
-Category of items in each department  
-Opening and closing time  
-position of the Totem  
-the certificate that validates the role of the shop manager   
  
The store manager once registered must equip the shop with the necessary technologies: -Totem to allow people to take the ticket on spot. It must be connected to the system to insert the ticket in the schedule of the day

-QR code scanners positioned at the entrance and exit of the shop to track the number of people inside the building and to modify the schedule real time.

**Show the status of the queue**

The user, after receiving the ticket, has always the possibility to enter the application and check the status of the queue in the dedicated field (“Check Status” field).   
In particular the system provides the user the number of people before him/her in the queue and the estimation of the time he/she has to wait.

**Show list of shops and related info**

Once opened the application the customer can see a list of shops divided by categories. If he/she has provided the localization the list is sorted from the closest to the farthest one. For each shop the application shows how many people are already in the corresponding queue, which categories of items the store provides, the position and the opening and closing time of the store.

**2.3 User Characteristic**

This section provides the list of the actors and how the system interacts with them

1. **User-store customer:** who downloads Clup application on his mobile phone, registers on it and logs in to *take a ticket* to do the shopping in a safer way  and *book a visit* in a shop for the following days.
2. **User-Store manager:** who exploits CLup Application and register his/him shop to it, to give the customer the possibility to avoid queuing in front of the store with other people and letting them do the shopping keeping the right distances from the other customers.

**2.4 Assumptions dependencies and constraints**

in this section it is explained everything that cannot be controlled by the system, but it is required by it to work properly.

D1. When a user requests a ticket or books a visit the internet connection must work properly.

D2. There are totems at the entrances of the registered store

D3 There are QR scanners at the entrances/exits of the registered store

D4. The Shop’s address, the pictures, the categories of items, the various departments with their associated capacities and any other kind of shop’s information are correctly provided by the store manager.

D5. The various information that can be provided by the users (position address, category of items that he/she wants to buy, approximate duration of the visit, time slot selected for the visit) is reliable.

D6: the user-customer arrives at the shop almost at the time associated with his/her ticket by the system estimations.

D7. User has the QR code associated with his/her ticket once arrived at the shop either on the device or in any other way (for instance he/she has printed it).

D8. Each person scans his/her QR code when entering and leaving the store.

D9. Each person who does not take the ticket online, always takes the ticket through the totem before entering the store.

D10. When active, the GPS provides the exact position with an error of ten meters at most.

(?)D11. when a customer decides to take a ticket, the internet connection works properly.

(?) D12. when a customer decides to book a visit, the internet connection works properly.

1. **Specific requirements**

**3.1 External interface requirements**

**3.1.1 User interfaces**

Add mock up here

**3.1.2 Hardware interfaces**

* **Mobile phone:** the customer needs to have a mobile phone in order to take a ticket online (instead if the customer wants to take the ticket on the spot, having a mobile phone is not necessary), while for the store manager it is required in order to register the shop in CLup Application.
* **Totem:** the stores need to have at least one totem positioned in a visible place around the shop so that people who take tickets on the spot can be managed by the system. In this way the totem represents a proxy between the customers and the system.
* **QR scanner:** they must be placed at the entrance and exit of the stores to make people scan the QR code associated with their tickets. Thanks to them it is possible to monitor the entrances and exits and update the schedule.

**3.1.3 Software interfaces**

* **City Map:** the use of a map of the city is necessary for the customer who decides to provide the position to choose a shop near him/her and calculate the time and route to reach the destination. To do this the system uses a public API.
* **Calendar:** the application allows customers to “book a visit” and for this reason it needs to show him/her a calendar. It is used to choose the day of the planned visit. For this purpose, there are several available APIs, like Google Calendar.

**3.1.4 Communication interfaces**

**3.2 Functional requirements**

**3.2.1 List of Requirements**

**R1.** The system should ask the user(customer/store manager) to register himself/herself to the application filling the form with mandatory fields

**R2.** The system should ask the store managers to provide documents that certify his/her association with the shop

**R3.** The system should provide the report of daily entrances/exits only to the shop managers

**R4.** The system should ask the user for authorization of GPS position

**R5**. The system should ask the user for authorization of the cookies

**R6.** The system should ask the user to tick privacy Terms & Conditions box

**R7.** The system should require the customer-user to be logged in to use the service *Take a ticket* of the application

**R33.** The system should require the customer-user to be logged in to use the *book a visit* of the application

**R8.** The system should require the shop manager-user to be logged in to use the services *Register the shop* and *Show Report of the day*

**R9.** The system should provide the list of the shop even when the users are not logged in

**R10.** The system requires the shop managers to give access to the data about their shops

**R11.** The system notifies the user (about the remaining waiting time) real time

**R12.** The system should ask the user who want to book a visit to fill the mandatory field of date and time of the visit

**R13.** The system builds statistics on data whichare stored in the database

**R14.** The system must save the data provided by the users when they register

**R15.** The system must save the generated ticket in the database real time

**R16.** the system should save the ticket of Book a visit real time in the schedule

**R17.** The system should generate automatically the QR code associated to the ticket **(??)**

**R18.** The system should save real time the ticket taken through the totem in the schedule

**R19**. The system should allow the user to decide between *Take a ticket* and *book a visit* options

**R20.**The system should allow the user to provide in an optional field the category of items they want to buy (in Book a visit option)

**R21.**The system should allow the user to provide in an optional field the estimated duration of the visit (in Book a visit option)

**R22.** The system should allow the user to cancel the reservation of a visit within the selected time slot

**R23.** The system should allow the user to cancel a ticket within his/her turn

**R24.** The system shall allow users to take a ticket online   ????REQ O GOAL

**R25.** The system shall allow users to book a visit               ????REQ O GOAL

**R26.** The system should allow the user to check the status of a queue of a shop

**R27.** The system should allow the user to check in each moment the remaining time for his/her turn

**R29.** The system should assign a position in the queue for each customer who takes a ticket

**R30.** The day of the booked visit the system should notify the user that he has a visit reservation

**R31.**The system should reserve a place in the selected time slot for each customer who books a visit.

**R32.** The system by analyzing data updates the interfaces which show real time information every 10 seconds

**3.2.2 Mapping**

**Goals DA REQ**

**G1 d2, d9 (r13?), r15,r17,18,r29**

**G2 d6**

**G3 d3,d4?,d6?,d8?, r11,r13,r14,r16,r18,r19?,r20?,**

**r21?,r26,r27,r29,r31,r32**

**G4 d2,d3,d4,d8,d9,d10 r2,r3,r8,(r13?),**

**G5 d2,d3,d6,d7,d8,d9, r11,r21,,r26,r27,r29,r32**

**G6 (?d4),d5,d11   r1,r4,r5,r6,r7,r9,r14,r15,r17,r19,23,r24,r26,r27,r29,r32**

**G7 d4,d5,d1,d10, r1,r5,r6,r33,r9,r12,r14,r16,r17,r19,**

**r20,r21,r22, r25,r30,r31**

**3.2.3 Use Cases**

**3.2.3.1 Use Cases Description**

**name**: Signup-customer

**actor**: user-customer **entry condition**: the user has the internet connection available and the application downloaded and installed in his/her device  
**Event Flow:**

1. the user enters the application that  shows the homepage
2. the user clicks on the “Sign up” button
3. the user choose the “Customer” option (between “customer” and “shop manager” options)
4. the user fills the mandatory fields
5. the user fills some/all of the optional fields
6. the user ticks the “I agree to the Terms of Services and Privacy Policy” box
7. the users clicks on the “Continue” button
8. the user  chooses between “Don’t Allow” and “Allow” options with relation to the the access of his/her position
9. the system stores the user’s data

**Exit Conditions:** the user is correctly registered to Clup Application and so he/she has the possibility to exploit the functionalities for the customer  
**Exceptions:**

1. the user was already registered and the system asks the user to login
2. the username already exists and the system warns the user asking him/her to write another one
3. the user does not tick on the “I agree to the Terms of Services and Privacy Policy” box and the system warns the user to do it in order to confirm the registration
4. the user does not fill all the mandatory fields and the system asks him/her to insert  missing information.

-------------------------------------------------------------------------------------------------------------

**name**:  Signup-shop manager

**actor**: user-shop manager **entry condition**: the user has the internet connection available and the application downloaded and installed in his/her device  
**Event Flow:**

1. the user enters the application that  shows the homepage
2. the user clicks on the “Sign up” button
3. the user chooses the “Shop Manager” option (between “Customer” and “Shop Manager” options)
4. the user fills the mandatory fields
5. the user fills some/all of the optional fields
6. the user ticks the “I agree to the Terms of Services and Privacy Policy” box
7. the users clicks on the “Continue” button
8. the user uploads the certificate that  validates his/her role
9. the user fills the mandatory fields about the store (Name and Address)
10. the system validates the certification
11. the user upload the image of the store (not mandatory)
12. the users clicks on the “Finish” button
13. the user fills the mandatory fields about his/her shop(Name, Address, Departments’ list with capacity and category of items, position of totems)
14. the user fills the optional fields about the additional information about his/her shop (description of the departments and /or of the product)
15. The user clicks on the “End Registration” button
16. The system stores the user’s data

**Exit Conditions:** the user is correctly registered to Clup Application and so he/she has the possibility to exploit the functionalities for the shop manager. Now the shop is visible in the list of the stores of the application.

**Exceptions:**

1. the user was already registered and the system asks the user to login
2. the username already exists and the system warns the user asking him/her to write another one
3. the user does not tick on the “I agree to the Terms of Services and Privacy Policy” box and the system warns the user to do it in order to confirm the registration
4. the user does not fill all the mandatory fields and the system asks him/her to insert  missing information
5. the certification uploaded by the user is not valid and the system asks him/her to upload it again.

-------------------------------------------------------------------------------------------------------------

In the following use case we consider the general User (both Customer and Shop Manager) because the login phase is the same. The User HomePage is different for the customer and the shop manager

**Name:** Log-in **Actor:** User **Entry Condition:** The application is correctly installed, the internet connection works properly and the user has already signed up **Event Flow:**

1. the user opens the application on his/her device
2. the user enters the credentials (Email and Password)
3. the user clicks on the “Sign In” button
4. The system opens the User’s homepage

**Exit Conditions:** the user is successfully logged in and he/she has access to the offered services  
**Exceptions:**

1. the user enters a wrong email and the system asks him/her to re-insert it.
2. the user enters a wrong password and the system asks him/her to re-insert it.

-------------------------------------------------------------------------------------------------------------

**Name:** take a ticket **Actor:** User-customer  **Entry Condition:** theuser has logged in, the application is open in the home page and internet connection works properly **Event Flow:**

1. the user selects a shop from the list of the available ones
2. (????)the user visualizes the information about the selected shop (people already in the queue, category of items, address, opening and closing time)
3. the user chooses  “Take a Ticket” between “Take a ticket” and “Book a visit” options
4. the system saves the ticket in the schedule
5. if the user has not already given access to the GPS localization, he/she decides whether to provide or not the address.
6. if the system has access to the position of the user (either GPS or address), the user receives the estimated time to reach the shop.
7. the user visualizes the estimated time, the number of people before him in the queue and the QR code associated to the ticket

**Exit Conditions:** the user has successfully taken a ticket

**Exceptions:**

1. the user takes a ticket while he/she is already in the queue and the system warns him/her by sending the message “You are already in the queue of another shop”
2. the user provides a nonexistent address and the system asks him/her to write another one.
3. the user wants to take the ticket for a shop whose queue is full until the closing time and the system warns him/her by sending the message “This shop is unfortunately full! Please choose another one”

-------------------------------------------------------------------------------------------------------------

**Name:** Book a visit **Actor:** User customer **Entry Condition:** theuser has logged in, the application is open in the home page and internet connection works properly

**Event Flow:**

1. the user selects a shop from the list of the available ones
2. (????)the user visualizes the information about the selected shop (people already in the queue, category of items, address, opening and closing time)
3. the user chooses “Book a visit” between the “Take a ticket” and “Book a visit” options
4. the user selects the day he/she wants to do the visit in a calendar
5. the user selects between the available time slots (mandatory)
6. the user fills some/all of the optional fields (category of items/estimated duration of the visit)
7. the system saves the information in the schedule
8. the user receives the associated QR code and a recup of the visit (date and shop)
9. (????)the day of the visit the user receives an alert notification

**Exit Conditions:** the user has successfully booked a visit   **Exceptions:**

1. the user selects a day which does not have available time slots and the system warns him/her by sending the message “This day has not available time slots. Please select another one!” and giving him/her the possibility to select another day
2. the user gives an estimation of the duration of the visit which is longer than the remaining time before the closing of the shop. The system warns him/her with a notification and does not store the provided value of the duration.

**Name:** check ticket’s status   
**Actor:** User-customer **Entry Condition:** theuser has logged in, the application is open in the home page  
**Event Flow:**

1. the user clicks on the “Menu” button
2. the user clicks on the “Check Status” button
3. the user visualizes the estimated remaining waiting time, the number of people above him/her in the queue, and the “Delete” button.

**Exit Conditions:** the user visualizes a page which contains all the information about his/her taken ticket)

**Exceptions:**

1. the user has not taken a ticket and the system shows a  screen with  “Take a ticket and then come back here to see all the related information!” message
2. The internet connection suddenly drops and the system sends the “the connection is lost” message.

-------------------------------------------------------------------------------------------------------------

**Name:** download QR code

**Actor:** User-customer

**Entry Condition:** theuser has logged in, the application is open in the home page   
**Event flow:**

1. the user clicks on the “Menu” button
2. the user clicks on the “My QR codes” button
3. the user visualizes the list of his/her QR code(s)
4. the user clicks on “download” button associated to the QR code he/she want to save in his/her device

**Exit Condition:** the user has  QR code(s) saved in his/her device

**Exception:**

1. **the user has neither taken a ticket nor booked a visit and the system**  shows a  screen with  “Take a ticket and/or book a visit and then come back here to download the associated QR code(s)!” message
2. The internet connection suddenly drops and the system sends the “the connection is lost” message.

-------------------------------------------------------------------------------------------------------------

**Name:** Cancel ticket

**Actor:** user-customer

**Entry Condition:** The user is in the “check ticket’s status” page

**Event Flow:**

1. the user clicks on the “Cancel Ticket” button
2. the user clicks “Yes” on the “Do you want to confirm the cancellation of the ticket?” button
3. the system removes the reservation from the schedule

**Exit Conditions:** the ticket is no longer present in the schedule

**Exceptions:**

1. The internet connection suddenly drops and the system sends the “the connection is lost” message.

-------------------------------------------------------------------------------------------------------------

**Name:** Check booked visit(s)

**Actor:** User-customer **Entry Condition:**theuser has logged in, the application is open in the home page  
**Event Flow:**

1. the user clicks on the “Menu” button
2. the user clicks on the “My booked visits” button
3. the user visualizes the list of his/her booked visit(s)
4. the user selects one booked visit
5. the user visualizes the Name of the Shop (and the related information), the Date, slot time, the duration of the visit and the category of items he/she wants to buy(if provided) and the “Delete” button.

**Exit Conditions:** the user knows all the information about his/her booked visit(s)

**Exceptions:**

1. the user has not booked any visit and the system shows a  screen with  “Book a visit  and then come back here to see all the related information!” message.
2. The internet connection suddenly drops and the system sends the “the connection is lost” message.

-------------------------------------------------------------------------------------------------------------

**Name:** Cancel Visit

**Actor:** user-customer

**Entry Condition:** The user is in the “Check booked visits” page  
**Event Flow:**

1. the user clicks on the “Cancel Visit” button
2. the user clicks “Yes” on the “Do you want to confirm the cancellation of the ticket?” button
3. the system removes the reservation from the schedule

**Exit Conditions:** the visit is no longer in the schedule

**Exceptions:**

1. The internet connection suddenly drops and the system sends the “the connection is lost” message.

-------------------------------------------------------------------------------------------------------------

**Name:** Modify shop’s information   
**Actor:** User-shop manager **Entry Condition:** theuser has logged in, the application is open in the home page   
**Event Flow:**

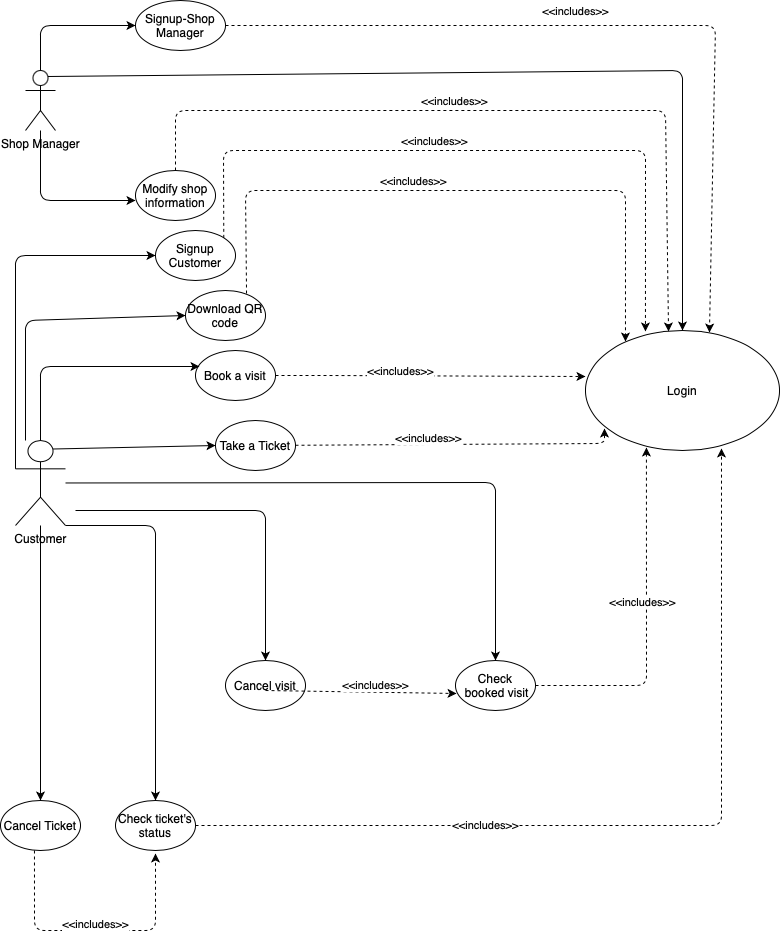
1. The user clicks on the “Menu” button
2. the user clicks on the “About my Shop” button
3. the user clicks on “Edit” button
4. the user modifies some/all the information about his/her shop
5. the user clicks on “Confirm” button
6. the system saves the updated data

**Exit Conditions:** the shop information are modified

**Exceptions:**

1. the user provides a nonexistent address and the system asks him/her to write another one.
2. The internet connection suddenly drops and the system sends the “the connection is lost” message.

**3.2 Use Cases Diagram**



**3.3 performance requirements - Non-Functional Requirements**

The system should be able to simultaneously serve 80000 individuals and to manage 200 shops. This is the start idea, but the system should be flexible to changes and the number of requests of registration (both from the customers and from the shop manager users).

The response time for any action must be less or equal to 1 second. In addition, the system must generate the QR code within 3 seconds from the generation of the ticket or the “book a visit” request.

The system should be able to run the algorithms implemented to manage the data in the schedule within two seconds from the action of the user.

**3.4 design constraints - Non-Functional Requirements**

**3.4.1  standards compliance**

* The application is able to preserve the state of the system avoiding the accidental loss of data: the user should be able to have his/her state always preserved.
* According to the privacy of data, the application processes sensitive ones according and so the project is subject to the GDPR (General Data Protection Regulation). These private data are stored in the internal storage of the application.
* To exploit all the services of the application, it only requests absolute minimum permissions.

**3.4.2 hardware limitations**

All the shops must have totems and QR scanners in their building to manage the requests of tickets/entrances/exits of people from/to the shops.

Both the user-customer(who decides not to take the ticket on the spot) and the user-shop manager must own a device to exploit the functionalities of the application. To register and log in themselves, they also need a properly working internet connection. This must be available also in the moment in which the user takes a ticket or books a visit and in any other action that requires the analysis of data in the database:

- for the customers: check the status of the queue, check the remaining waiting time...

- for the shop managers: check daily reports for the shop manager…

Moreover, the device should have GPS sensors to provide the correct position of the user in case he decides to allow the system to access it.

**3.4.3 Any other constraints**

The application must respect all the privacy policies with relation to the user, according to the privacy Terms & Conditions: the personal information and the telephone number are not used for commercial scopes; the position, if provided, is used only to compute the estimation of the time necessary to reach the destination from the address of the user.

**3.5 Software System Attributes**

**3.5.1 Reliability**

The system must be up 24 hours per day, every day without any interruption. In case of failure an error message must be displayed on the application within 15 minutes.  To avoid data loss the system should periodically backup the data and the core services must be duplicated.

In this way the system  guarantees high reliability.

**3.5.2 Availability**

To make the system available full time and enable it to continue operating properly in the event of failure, it has a strong fault tolerance architecture.

Redundancy (switching to the duplicated system) must be ensured to make data available in case of breakdown of a portion of the system.

**3.5.3 Security**

Since the users (both customers and shop managers) have to provide sensitive data, the system should protect it from any possible internal and external attack. For this purpose users’ passwords and personal information must be encrypted to be protected during transmission. In this way data becomes useless if hacked from a server and the protection of privacy is guaranteed.

**3.5.4 Maintainability**

The system must have a high level of maintainability. For this scope the code should be fully commented in order to better explain all the components present in the system.

Moreover the associated documentation should be extremely clear to make the project fully understandable by everyone.

Tests associated with the code must cover at least  80% of it. Specifically, automated tests should be used to make it easy to validate changes and integration tests must be continuous in order to build the code easily.

**3.5.5 Portability**

The software must be implemented as a multiplatform application. In particular, it should support Android and iOS operating systems for mobile devices.