**3 Specific requirements**

**3.1 External interface requirements**

**3.1.1 User interfaces**

The following mockups <<there is a water here; every human reading it must understand it>>.

**3.1.2 Hardware user interfaces**

<<The description of the hardware interface>>

**3.1.3 Software interfaces**

The system does not provide any API to external applications, but it uses some proven services and ready-made functionality for stable performance.

* Yandex Maps API

Yandex Maps API is a set of services that allow to use Yandex map up-to-date information such as public transport schedules, traffic jams, reconstruction in a project. In our case, as mentioned, it is used to lay out the route and display the stores available in the area and their additional information such as opening hours, load, approximate waiting time.

* Calendar and alert system

Planning a trip to the store is not possible without a calendar and specifying the exact time of arrival. After viewing the available dates and time of booking, the user assigns the most comfortable one for them. For convenience, it is recommended using a notification system about planned visit

* QR code reading/generation

An important attribute of the program is the QR code. It is generated on the server side after the request to visit the store is confirmed. It is used to enter and exit the store after scanning by the staff

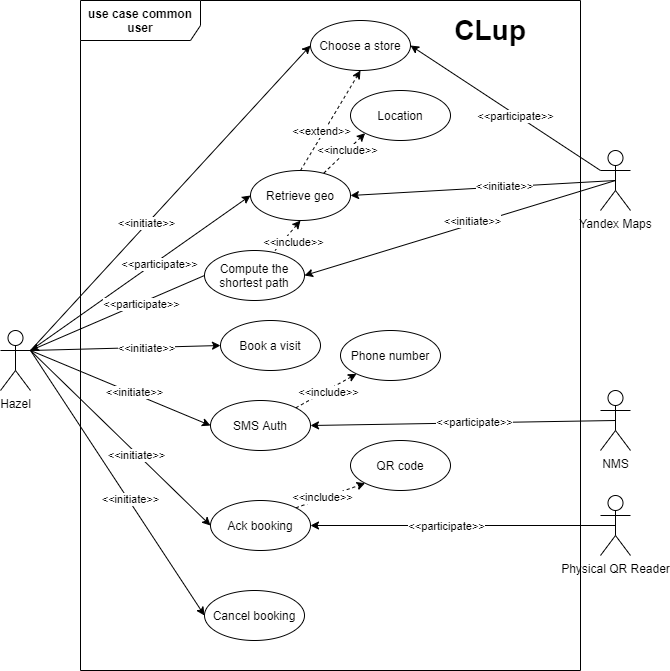
**3.2. Functional requirements**

**3.2.1 Common User**

**Scenario 1**

Because of the pandemic Hazel does not want to endanger her own life and lives of the others, and she does not leave the home. But using food delivery services soon became too expensive for her. And she started to look for the other ways to get food supplies, so as an active app user she downloaded and installed CLup. This service helped her to get to the store, buy all the supplies she needed and reduce contact with people to a minimum.

**Use case diagram**



*Figure – Use case diagram: Common User*

**Use cases**

|  |  |
| --- | --- |
| Name | Choose a Store |
| Actor | Hazel, Yandex Maps |
| Entry conditions | 1. Hazel has opened the application on her device  2. (Optionally) She could turn on the geo location |
| Events flow | 1. Hazel chooses a store on the map  2. Hazel clicks on the “Choose” button  3. The app saves the data |
| Exit conditions | Hazel has chosen the store where she would buy the supplies |
| Exceptions | 1. Hazel forgot to turn on the Internet connection, so she would get the Connection Error  2. Hazel has chosen the store that is closed this time, then the app would display the closed store status |

|  |  |
| --- | --- |
| Name | Location |
| Actor | Hazel |
| Entry conditions | Hazel has opened the application on her device |
| Events flow | 1. The app displays the geolocation banner that asks the user to turn on the geolocation  2. Hazel approves or rejects  3. The app saves the data |
| Exit conditions | Hazel approved or rejected the usage of geolocation by this service |
| Exceptions | 1. Unavailability of Yandex Maps API Servers |

|  |  |
| --- | --- |
| Name | Retrieve geo |
| Actor | Yandex maps, Hazel |
| Entry conditions | 1. Hazel has opened the application on her device  2. The location must be approved |
| Events flow | 1. Yandex maps retrieve Hazel geo position  2. Yandex maps display Hazel geo position on the map |
| Exit conditions | 1. The app is closed  2. The Internet connection is lost.  3. Hazel rejects the location. |
| Exceptions | 1. Unavailability of Yandex Maps API Servers |

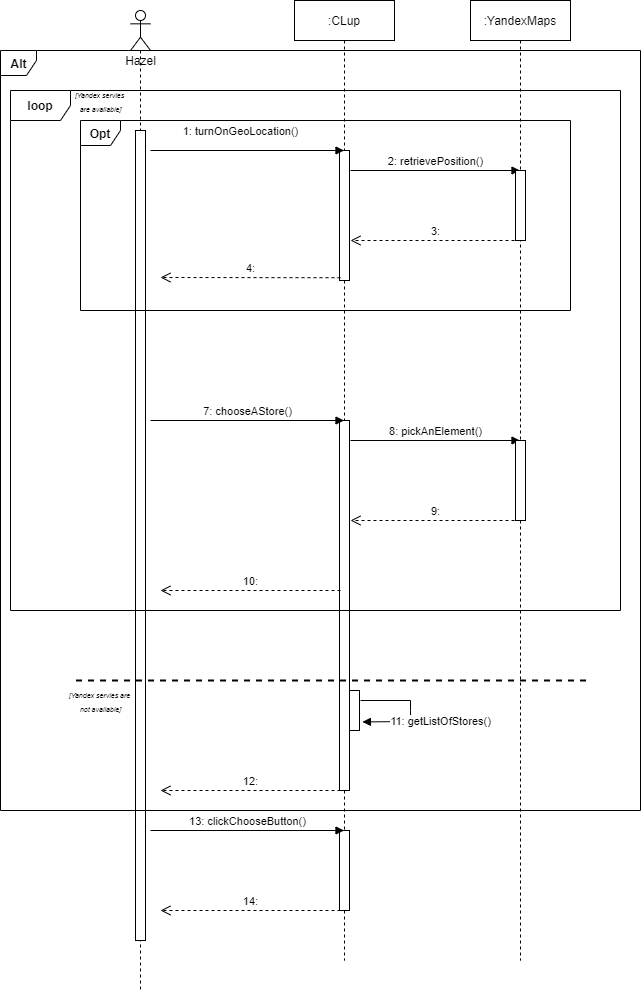
|  |  |
| --- | --- |
| Name | Compute the shortest path |
| Actor | Yandex maps, Hazel |
| Entry conditions | 1. Hazel has opened the application on her device  2. The location must be approved  3. Yandex maps must retrieve Hazel geo position |
| Events flow | 1. Yandex maps retrieve Hazel geo position  2. Hazel has chosen the store  3. Yandex maps return the shortest path to this store |
| Exit conditions | 1. The app is closed  2. The Internet connection is lost.  3. Hazel rejects the location  4. Hazel cancels the choice of this store |
| Exceptions | 1. Unavailability of Yandex Maps API Servers  2. Unexpected road works or accidents |

|  |  |
| --- | --- |
| Name | Book a visit |
| Actor | Hazel |
| Entry conditions | 1. Hazel has opened the application on her device  2. Hazel has chosen the store |
| Events flow | 1. Hazel chooses the departments she wants to visit  2. Hazel chooses the date and the time of the visit  3. Hazel enters the phone number  4. Hazel clicks “Send SMS” button |
| Exit conditions | 1. The app is closed  2. Hazel clicks “Cancel” button |
| Exceptions | 1. The Internet Connection is lost  2. The occurrence of “Nonrepeatable Read” |

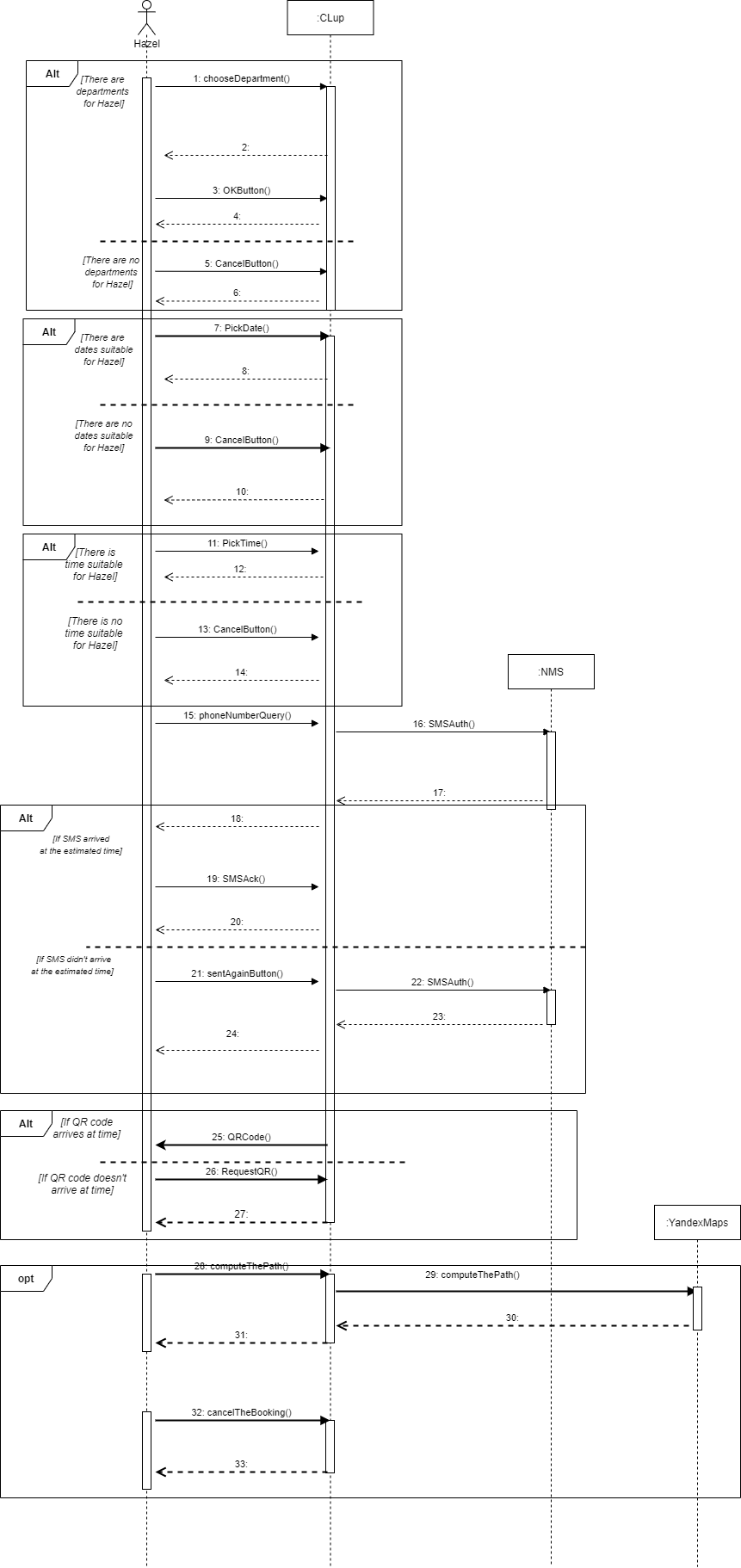
|  |  |
| --- | --- |
| Name | SMS Auth |
| Actor | Hazel, NMS |
| Entry conditions | 1. Hazel has opened the application on her device  2. Hazel has chosen the store  3. Hazel has finished choosing the booking entry data  4. Hazel has entered her phone number |
| Events flow | 1. Hazel verifies the SMS code  2. Hazel gets the QR code acting as a ticket to the store  3. Hazel clicks “Ok” button |
| Exit conditions | 1. The app is closed |
| Exceptions | 1. The Internet Connection is lost  2. Hazel hasn’t got the QR code  3. Hazel hasn’t got the SMS code |
| Name | Ack booking |
| Actor | Hazel, Physical QR reader |
| Entry conditions | 1. Hazel has booked a visit  2. Hazel has arrived to the store |
| Events flow | 1. Hazel scans the QR code when she arrives  2. Hazel scans the QR code when she leaves |
| Exit conditions | 1. Hazel leaves the store |
| Exceptions | 1. Hazel didn’t arrive at the store  2. Hazel accidently scans the QR code twice (maybe even thrice) |
| Special Requirements | 1. Hazel receives a reminder notification 15 minutes before the estimated arrival at the store |

|  |  |
| --- | --- |
| Name | Cancel booking |
| Actor | Hazel |
| Entry conditions | 1. Hazel has booked a visit |
| Events flow | 1. Hazel goes to the account tab  2. Hazel cancels the visit |
| Exit conditions | 1. The app is closed |
| Exceptions | 1. The Internet connection is lost |

**Sequence diagrams**

****

*Sequence Diagram: Choose A Store*

****

*Sequence Diagram: Book A Visit*

**3.2.2 Prioritized User**

**Scenario 2**

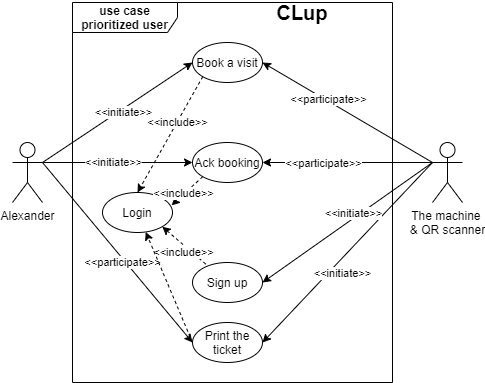
Alexander is an elderly Afghanistan veteran but the pandemic knocked him down. He has lost all the joys of his life: he couldn’t see his relatives and he couldn’t even go to the store without the risk of getting sick. He doesn’t have a smartphone, and new technologies seem too complicated for him. So Alexander goes to the store and he sees some machine giving tickets. Thanks to the presence of this machine the risks of getting sick for him have decreased.

**Scenario 3**

Julie is a student and as every young person, she doesn’t like planning. Walking along the embankment (yes, she doesn’t want to observe self-isolation) she realized that she would like to have a snack. So she goes to the nearest store and books a visit via the machine using the "CLup" service. This service will help the people concerning their health to minimize the risk of illness.

**Use case diagram**

The machine & QR scanner is the one device. This division was made in the diagram just to make it clear that the QR scanner is embedded into the machine.



*Figure – Use case diagram: Prioritized User*

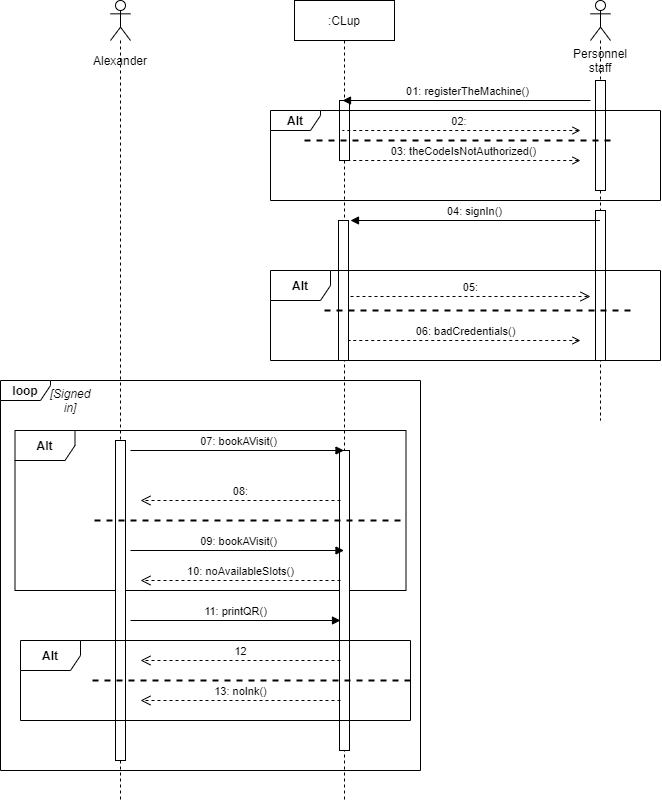
|  |  |
| --- | --- |
| Name | Book a visit |
| Actor | Alexander, the machine & QR scanner |
| Entry conditions | 1. Alexander is in the store and near the machine  2. Machine has been registered in the system |
| Events flow | 1. Alexander clicks the button “book a visit”  2. The machine shows the QR code and Alexander click “Ok” |
| Exit conditions | 1. “OK” button is pushed |
| Exceptions | 1. There is no available slots the next hour |
| Special Requirements | 1. Availability of a consultant to help and control the flow of people |

|  |  |
| --- | --- |
| Name | Ack booking |
| Actor | Alexander, the machine & QR scanner |
| Entry conditions | 1. Alexander has booked a visit  2. Alexander has arrived at the store |
| Events flow | 1. Alexander scans the QR code when he arrives  2. Alexander scans the QR code when he leaves |
| Exit conditions | 1. Alexander leaves the store |
| Exceptions | 1. Alexander didn’t arrive at the store  2. Alexander accidently scans the QR code twice (maybe even thrice) |

|  |  |
| --- | --- |
| Name | Sign up |
| Actor | The machine & QR scanner, staff personal |
| Entry conditions | 1. The machine has turned on |
| Events flow | 1. Personnel staff enters the required data to register the machine in the system |
| Exit conditions | 1. The machine has been registered |
| Exceptions | 1. The Internet connection is lost  2. The serial number doesn’t exist |

|  |  |
| --- | --- |
| Name | Login |
| Actor | The machine & QR scanner, staff personal |
| Entry conditions | 1. The machine has turned on |
| Events flow | 1. Personnel staff inputs the serial number of the machine |
| Exit conditions | 1. The machine has been logged in |
| Exceptions | 1. The Internet connection is lost  2. The serial number doesn’t exist  3. Power failure |

|  |  |
| --- | --- |
| Name | Print the ticket |
| Actor | The machine & QR scanner, Alexander |
| Entry conditions | 1. The visit has been booked |
| Events flow | 1. The machine prints the ticket |
| Exit conditions | 1. Alexander has the ticket |
| Exceptions | 1. The ink is out  2. The Internet connection is lost  3. Out of print paper  4. The user loses the received ticket |



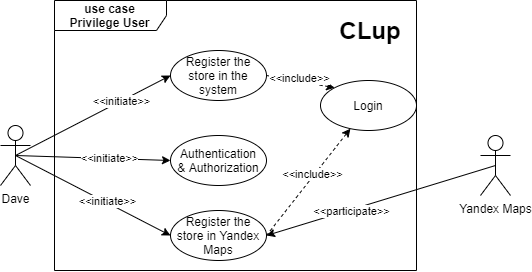
*Sequence Diagram: Book A Visit via the machine*

**3.2.3 Privilege User**

**Scenario 4**

Working in a store during the pandemic is a big risk, Dave figured it out the hard way. He had COVID-19, and when he getting on his feet, he decided to secure the store in which he works as a manager. So his choice fell on the CLup because he doesn’t need to recruit couriers, it will help him to save some money at this difficult time. All he needs to do is to estimate the number of people for each department and in line at the cash register and provide this data to the system.

**Use case diagram**



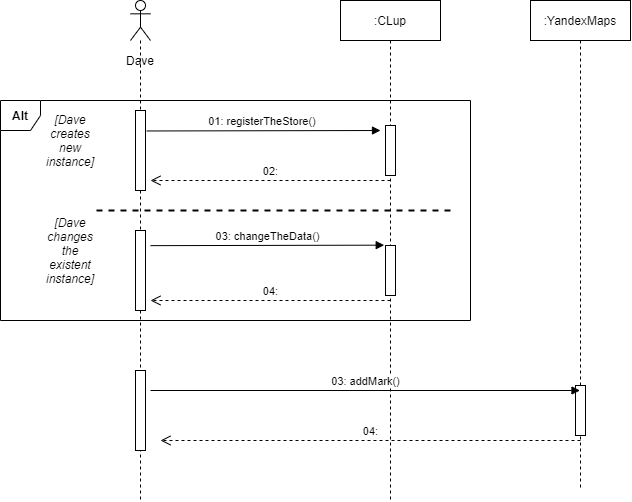
*Figure – Use case diagram: Privilege User*

|  |  |
| --- | --- |
| Name | Authentication & Authorization |
| Actor | Dave |
| Entry conditions | Dave has opened PC and started the app |
| Events flow | 1. Dave chooses the “Create” option  2. Dave enters login, password and phone number  3. Dave chooses the “OK” button  4. The system saves it |
| Exit conditions | 1. Dave is authenticated and authorized as a manager  2. System saves the data |
| Exceptions | 1. He inputs incorrect data (not correct phone number, long login, not strong password)  2. The session was interrupted for technical reasons |

|  |  |
| --- | --- |
| Name | Login |
| Actor | Dave |
| Entry conditions | Dave has signed in |
| Events flow | 1. Dave chooses the “Sign in” option  2. Dave enters login, password  3. Dave chooses the “OK” button |
| Exit conditions | Dave is logged in |
| Exceptions | 1. He inputs incorrect data (a login or a password)  2. The session was interrupted for technical reasons |

|  |  |
| --- | --- |
| Name | Register the store in the store system |
| Actor | Dave |
| Entry conditions | Dave has logged in |
| Events flow | 1. Dave enters the name of the store  2. Dave enters the address of the store  3. Dave inputs the personnel staff of the store and authorizes them to register the machines  4. Dave inputs the area of the store  5. Dave inputs the number of the departments and their area  6. Dave inputs the number of cash registers  7. Dave clicks “Ok” button  8. The system saves the information about this store |
| Exit conditions | The system has the information about the store. Dave logs out |
| Exceptions | 1. He inputs invalid data  2. He misprints some data and wants to correct it  3. The session was interrupted for technical reasons |

|  |  |
| --- | --- |
| Name | Register the store in Yandex Maps |
| Actor | Dave, Yandex Maps |
| Entry conditions | Dave has logged in |
| Events flow | 1. Dave creates the mark on Yandex Maps  2. Dave enters the name, address and the description of the store  3. Dave marks the store as active in the system  4. The system saves the information |
| Exit conditions | The store and its description are displayed on Yandex maps |
| Exceptions | 1. He inputs invalid mark  2. He misprints some data and wants to correct it  3. He enter invalid name or address  4. He forgot to mark the store as active  5. The session was interrupted for technical reasons |



*Sequence Diagram: Register the store in the system*

3.2.3 Requirements

**G1: 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.**

[D2] - The device must have stable access to the Internet to establish a user status and queue calculation.

[D3] – The user goes to the selected shop according to the notification on the mobile device.

[D11] – The user is going to the store by shortest path.

[D12] – If the user is late more than 10 minutes (according to his/her appointed time) he/she is removing from the queue.

[D14] – The location of the user is obtained by GPS. The locations of the stores are retrieved by API.

R1: The system should allow users to go to the store by shortest path calculated by Yandex Maps.

R2: The all information about visit (arrival/departure time, the selected departments) should be analyzed by system, to estimate the time of waiting more precisely.

**G2: To give people opportunity to stay in queue remotely to avoid crowds in stores and in real queues.**

[D1] – Registration is via the phone number.

[D4] - The user should follow the rules that will be notified on the device (i.e. social distance, the presence of a mask, allocated time for purchase).

[D6] - The customer will enter/exit the store using a QR code

[D8] – The user should denote the supposed departments to visit.

[D9] – Store owner must specify the area of the shop and departments.

[D12] – If the user is late more than 10 minutes (according to his/her appointed time) he/she is removing from the queue.

[D13] – It is possible to cancel the booking, if user plans are changed.

R3: The system should send the data of the visit to the server to organize the queue correctly.

R4: The system should provide concurrency to insert multiple request in short period of time in correct order.

R5: The system should balance the human flow in the store by analyzing the departments and store capacity.

R6: The system generates QR codes by UUID, telephone number/device id and suggested time of the visit.

**G3: Stores should have the possibility to hand out “tickets” on the spot.**

[D2] - The device must have stable access to the Internet to establish a user status and queue calculation.

[D5] - People who cannot use the device should easily integrate into the queue by registering on the spot.

[D6] - The customer will enter/exit the store using a QR code.

[D8] – The user should denote the supposed departments to visit.

[D10] – The ticket given by machine will contain information about the customer arrival/departure time.

[D12] – If the user is late more than 10 minutes (according to his/her appointed time) he/she is removing from the queue.

R7: The system should distinguish users of mobile phones and local devices to prioritize the second ones in the queue.

R8: QR readers should be installed at the entrance and exit of the store sending data of arrival and departure time to system.

R9: The machine should be authorized and logged in the system by the manager of the store.

**G4: The system should allow customers to book a visit to the supermarket to optimize quantity of people in it by time of their visit, and category of products which they want to purchase.**

[D1] - Registration is via the phone number.

[D3] - The user goes to the selected shop according to the notification on the mobile device.

[D6] - The customer will enter/exit the store using a QR code.

[D8] – The user should denote the supposed departments to visit.

[D9] – Store owner must specify the area of the shop and departments.

[D12] – If the user is late more than 10 minutes (according to his/her appointed time) he/she is removing from the queue.

[D13] – It is possible to cancel the booking, if user plans are changed.

R10: The system automatically extends the queue if the user books a visit for particular date and time.

**G5: The system (application and “tickets” on the spot) should include alternative slots (for another day), suggest to the customer the location of the nearest “safe” store based on his location.**

[D3] - The user goes to the selected shop according to the notification on the mobile device.

[D5] - People who cannot use the device should easily integrate into the queue by registering on the spot.

[D6] - The customer will enter/exit the store using a QR code.

[D9] – Store owner must specify the area of the shop and departments.

R11: The system should provide algorithms to assess the stores human flow.

**G6: The system should allow the third party to get the statistical information to perform better management of the store.**

[D5] - People who cannot use the device should easily integrate into the queue by registering on the spot.

[D6] - The customer will enter/exit the store using a QR code.

[D7] - Store owners will have access to an expanded version of the app for more control and tracking of the flow of customers.

[D9] – Store owner must specify the area of the shop and departments.

R12: The staff of the store have to login into the system to obtain statistic to manage the store in a better way.

R13: The system provides to the store owner only statistical information.

R14: The system must have instruments for visualization.