

17/11/2020

### World phenomena:

someone wants/needs to go to the supermarket  
order in which a person does the grocery shopping  
person gets to/approaches the supermarket by car/bike/on foot, ...  
a person decides to buy an item  
a person pays at the check-out  
a person exits without buying (contained in the next one)  
*a person enters/exits in/from the supermarket (maybe Shared: our system sees people exiting and controls entrance flow)*

### Shared phenomena:

available slots for shopping (World controlled: it depends on people deciding to go or not to the shopping centre)  
person books a visit (World controlled)  
person gets a ticket (World controlled)  
person cancels his booked visit (World controlled)  
person cancels his booked ticket (World controlled)  
a person waits to be called for his turn (Machine controlled)  
*a person knows approximate time to spend in the supermarket (World controlled)*  
*a person knows approximate categories of items to buy (World controlled)*  
a person is notified for entrance (Machine controlled)  
a person buys a ticket / books a visit but doesn't go to the grocery shop (World controlled)  
a person buys a ticket / books a visit at a certain date/time but doesn't respect the scheduled date/time (World controlled)  
a customer is notified to leave from home (Machine controlled)  
a customer receives a suggestion to book a visit in its habitual day of the week and time, if any slot is available (Machine controlled)  
a customer receives a suggestion to book a visit in a different shop if the one selected hasn't available slots in the selected date/time (Machine controlled)  
a customer receives a suggestion to book a visit in a different date/time in the same shop if the one selected is not available (Machine controlled)  
a customer receives a suggestion to get a ticket in a different shop if the one selected hasn't available slots (Machine controlled)

### GOALS:

avoid physical lining up (*maybe this is a sub-goal of the following one*)  
grant social distance outside shopping centre  
manage entrances in the shopping centre (?)  
avoid having crowds inside (avoid too many people inside the shopping centre)  
*enter in the supermarket respecting safety rules*  
*respect security rules imposed by government (probably too general: it implies all the others)*

## DOMAIN ASSUMPTIONS:

1. GPS provides the exact location with an error of 10 metres at most
2. people inside the supermarket respect social distance as imposed by safety rules of the country/region
3. a ticket is associated to only one person
4. a visit is associated to only one person
5. only one person enters the shop for every ticket/visit
6. the supermarket has a maximum capacity for people inside
7. a very little part of customers uses ticket machines for reservations, because mobile application and call center guarantee a more comfortable service accessible to almost everyone (people with a smartphone use the application, people without it can call and book)

## REQUIREMENTS:

1. the system allows users to book one or more visits
  - A. the system allows users to book one visit with a ticket machine only if no slots for tickets are available for that supermarket in the current day **(dividerlo, non fare più sub-requirement ma 3 separati, specificando meglio questo)**
  - B. the system allows users to book one or more visits with the mobile application
  - C. the system allows users to book one or more visits calling the call center
2. the system does not allow a person to enter if shopping centre's maximum capacity has been reached
3. the system does not allow for entrance a person who has not booked a visit or got a ticket
4. a user application is associated to a profile (personal profile)
5. the system allows "ticket-machine" application to get a ticket
6. the system allows a user application to get a ticket only when no other ticket is "pending" (booked but not called yet) for that profile
7. the system sends a notification two minutes before entrance (only in case of user application and call center reservation)
8. the system allows a customer to have the possibility to enter only when it is his turn and for the next ten minutes
9. the system allows the user to book slots for shopping for the next month from current date and time
10. the system registers time spent inside the supermarket for every customer
11. the system allows to insert a list of items (that user would like to buy) when booking a visit (in case of mobile application)
12. the system is able to build statistics (time spent in the supermarket, items/categories of items purchased)
13. the system generates a unique code for entrance associated to every ticket/visit
  - a. the system generates a unique QR code associated to every ticket/visit booked from user application
  - b. the system generates a unique QR code associated to every ticket/visit booked from a ticket machine
  - c. the system generates a unique numeric code associated to every ticket/visit booked calling the call center
14. the system, in case of mobile application, sends notifications basing on user's

- actual position and time remaining before his turn
15. the system generates a unique numeric code for the position in the waiting "queue" associated to every ticket/visit booked from a ticket machine
  16. the system allows users to get information about estimated time before being called.
    - a. the system allows app user to get information about estimated time before being called in the mobile app
    - b. the system allows ticket machine user to get information about estimated time before being called through the display in the supermarket
  17. the mobile application enables 100% screen brightness when a QR code is displayed
  18. call centers are open in the supermarket's opening hours
  19. the system allows users to get a ticket reservation calling the call center
  20. the system allows users to book a visit calling the call center
  21. the system manages available slots depending on statistics built on specific user's habits (usual time spent in the supermarket, items/categories of items purchased)
  22. the system tracks user habits in order to detect its habitual shop and date/time of visit, in order to provide booking suggestions in advance if any slot is available
  23. the system is able to check availability for all the supermarket included in it
  24. the system allows user to delete a reservation
  25. ticket machines are available in the supermarket's opening hours

**Time:** 2h30:

1h World and Shared phenomena

30min goals

30min domain assumptions

30min requirements

Andrea and Roberto

## SCENARIOS

- **Delete a ticket reservation**

Alessandro, because of a sanitary emergency in his country, would like to avoid waiting for his turn in the supermarket's proximity. He has a smartphone and, through CLup app, decides to get a ticket: the expected time before being called is 2 hours. But in the meanwhile, he receives an important job call: he will be busy for the next four hours, so he deletes his ticket reservation through the application; the system notifies him that this operation went fine.

- **Delete a visit reservation**

Marta works from Monday to Friday as a teacher, so she usually goes to do the grocery shopping on Saturday afternoon. For this reason, she decides, using *CLup* mobile application, to book a visit to the supermarket for next Saturday, at 2pm. But as soon as she remembers that next Saturday she must pick her nephew up from school, she decides to delete her visit reservation (always with the app): the system notifies that this operation went fine, and the 2pm time slot for Saturday is now free for some other customer(s).

- **Entrance in the grocery**

Matteo is waiting in his car just outside his favourite grocery. Using *CLup* app on his smartphone, he has already got a ticket and his turn is approaching: he knows it thanks to the notification, received a few minutes ago. When it is time to enter, a new notification arrives and Matteo opens the app: he sees the QR code for entrance and a message telling him that it is his turn; QR code will expire in 10 minutes, as shown by another message (in the upper part of the smartphone's screen).

Matteo gets out of the car and enters the supermarket by showing his QR code to the scanner at the entrance: doors open and he can get into the grocery.

- **Book a visit from CLup app**

Stefania really likes planning all her activities for the coming week on Monday morning. She works in the city hospital and according to her timetable, she has only time to go to the supermarket on Wednesday. Therefore, in order to have the access to the BCD shop granted, on Monday she decides to use CLup app to book a visit for Wednesday at 10am. The requested time slot for the visit is available so the reservation is confirmed and she can go on the day desired.

- **Suggest an alternative, the requested visit is not available**

Andrea would like to go to his usual grocery, Esselunga (very close to his house), tomorrow at 5pm. It is very important to go tomorrow, because he has almost nothing in the fridge and his wife is complaining a lot about this fact.

Andrea uses *CLup* app on his smartphone to book a visit for tomorrow: he selects his usual supermarket, date and time for the visit. Unfortunately, a message tells him that there is no availability for his request; anyway, the system suggests him another date for his favourite Esselunga (in two days' time, always at 5pm) or another Esselunga shop (a little bit further from his house, but not very distant), available tomorrow at 5pm. Since it is very important for him to do the grocery shopping tomorrow, he accepts this latter advice: he books a visit for tomorrow at

5pm at the suggested Esselunga shop.

- **Get a ticket from call center (HO MESSO GET A TICKET CON GABRIELE)**

Francesco, a seventy five-year old man without a smartphone, before going to the grocery shop wants to be sure about the fact that there is availability for a ticket and does not want to wait outside the market if the estimated waiting time is too long. Thus, calling at the call center he can get the next ticket available (estimated time: 3 hours) with a certain reservation code and therefore he can avoid waiting outside the shop and get tired from physical lining up. After having waited comfortably from home, Francesco approaches the market and arrives at the exact time previously indicated to him by phone. When it is time to enter an SMS is sent to him. Then he shows its code and enters into the grocery shop.

- **Get a ticket from app**

Alice, on Thursday afternoon, on the way back home from her office, decides to go to the supermarket to buy the necessary for dinner. For this reason, she uses the CLup app to check if there is any ticket available with less than 15 minutes of estimated waiting time at her favorite shop (ABC shop). The app shows her that the estimated waiting time for the next ticket in the ABC shop is 10 minutes. So she gets the ticket and continues her way to the shop.

- **Get a ticket from call center**

Gabriele, a single forty eight-year old man, wants to go to the grocery shop, on the way home from work by car, to buy the necessary food for the next few days. When he tries to log in the CLup app, he notices that he has no mobile connection available at that moment. So, he decides to call at the call center in order to get a ticket and be able to buy something for the current day. The call center operator tells him that the next ticket available in the requested supermarket has an expected waiting time of 20 minutes. After Gabriele's confirmation, the operator tells him a numeric code that he will use for the entrance. Moreover the call center operator reminds Gabriele that he will be notified, with a SMS, at the moment he can access the shop. Therefore, Gabriele decides to approach the supermarket and waits for his turn on the car. At the moment he receives the SMS notification he approaches the entrance, shows its numeric code and enters the supermarket.

- **Book a visit from call center**

Lucia is a seventy-year-old woman without a smartphone. On Wednesday, she decides to go to the grocery shop to buy food for the next few days and, given the current rules about safety distance, she knows that she must get a ticket (for entrance) at one of the ticket machines. She approaches the ticket machine, but she reads on its screen a message saying "Expected waiting time for tickets is: 4 hours". Of course, 4 hours are much more than what Lucia was expecting, so she decides to come back home and call the call center to book a visit for another day. On the phone, she establishes a reservation next Friday at 3p.m., with the call center operator which confirms her reservation telling her a numeric code that she will use to enter the market on Friday. Lucia will come to the supermarket on Friday.

- **Available slot suggestion**

Marco is used to going to the grocery shop close to his house on Saturday

afternoon; he always books a visit using *CLup* application on his smartphone, in order to avoid crowds and respect social distance. Due to his high number of reservations in the last months, the system knows Marco's habits about grocery shopping: in fact, the man receives a notification on Monday about an available slot for next Saturday at 4pm. As soon as Marco sees the notification on his smartphone, he fills all requested data for the reservation and books the suggested visit for Saturday: in this way, his habits will continue to be respected also for this week.

- **Notification based on GPS position**

Antonella has a very busy life, with two little children and a full-time job. Three days ago, she used *CLup* application for booking a visit to her usual supermarket: her turn is today at 4pm. However, her busy life has made her forget about the reservation, and at 3.30pm she is still totally unaware of it. But her smartphone has GPS services running, so at 3.32pm she receives a notification about the approaching turn: so, she immediately gets dressed and goes to the supermarket (approximate time from home to the shop is 10 minutes by car: this information is known by the system thanks to GPS position). She is able to get there at 3.55pm, totally in time for her turn.

- **Get a ticket from ticket machine**

Antonio, an eighty-year-old man without a smartphone, approaches a grocery shop on foot, with the intention to buy all necessary for next week. Given the current rules about safety distance, he knows that he must get a ticket (for entrance) at one of the ticket machines outside the supermarket: he selects "Ticket" option on the touchscreen (wearing disposable gloves available near the ticket machine itself) and a ticket with a QR code is printed, together with a 6-digits number identifying his turn; in addition, the screen shows a message telling him that the expected waiting time before his turn is 15 minutes. Antonio decides to walk around and comes back after 10 minutes; two (more) minutes later, his turn comes and the identification number appears on the waiting-screen at the top level of the queue. The man approaches the shop's entrance and shows his QR code (on paper) to the scanner positioned there: doors open and he can get into the supermarket.

- **Ticket machine suggests to reserve the next available slot: no more ticket for the current day are available**

Giuseppe, an eighty-year old man without a smartphone, approaches a shop on foot, with the intention to buy all necessary for next week. Given the current rules about safety distance, he knows that he must get a ticket (for entrance) at one of the ticket machines outside the supermarket: he selects the "Ticket" option on the touchscreen (wearing disposable gloves available near the ticket machine itself). Unfortunately there are no tickets available for the current day anymore. Thus, the screen shows a message which invites him to reserve the first available slot in the shop in the next few days. The proposal refers to a reservation for the next day at 3pm. Giuseppe accepts the suggestion and reserves a slot for the next day.

19/11/2020

## USE-CASES

### 1) Registration of mobile app user

<b>ID</b>	UC1
<b>Name</b>	Registration of mobile app user
<b>Actors</b>	Mobile app user*
<b>Entry condition</b>	<ul style="list-style-type: none"><li>• User has the internet connection available</li><li>• User has downloaded <i>CLup</i> app on his smartphone/tablet</li></ul>
<b>Event flow</b>	<ol style="list-style-type: none"><li>1. User unlocks his smartphone</li><li>2. User opens <i>CLup</i> app</li><li>3. User sees the home page of the app</li><li>4. User clicks on "Sign up" button</li><li>5. User inserts email, name and surname in appropriate fields</li><li>6. User chooses his password for the service, according to security standards</li><li>7. User accepts <i>CLup</i> "Terms and conditions"</li><li>8. User clicks on "Continue" button</li><li>9. User sees the page for registering his mobile phone number</li><li>10. User inserts his mobile phone number and clicks on "Send a SMS" button</li><li>11. The system sends a SMS to the specified phone number, containing a 8-digits code</li><li>12. User receives the SMS and fills the appropriate field in <i>CLup</i> page with the received code</li><li>13. User clicks on "Confirm" button</li><li>14. The system confirms registration with an email; in the meanwhile, the app goes back to the home page</li><li>15. User opens the email and validates his registration by opening the suggested link</li></ol>
<b>Exit condition</b>	<ul style="list-style-type: none"><li>• The user is now able to login and use the application services</li></ul>
<b>Exceptions</b>	<ol style="list-style-type: none"><li>1. User does not fill some of the mandatory fields in the first page (email, name, surname, password, "Terms and conditions" acceptance)</li><li>2. Password does not match security standards</li><li>3. Email is already registered in the system's DB</li><li>4. User inputs a non-valid email address</li><li>5. Email does not exist</li><li>6. User is not the owner (or, in any case, has no visibility) of the specified email address</li><li>7. User does not fill some of the mandatory fields in</li></ol>

	<p>the second page (mobile phone number, received code)</p> <ol style="list-style-type: none"> <li>8. Mobile phone number is already registered in the system's DB</li> <li>9. User inputs a code different from the one sent by the system</li> <li>10. Mobile phone number does not exist</li> <li>11. User is not the owner (or, in any case, has no visibility) of the specified mobile phone number</li> </ol> <p>In cases 1,2,3,4, the system does not let the "Continue" button to be pressed until all fields are correctly filled in.  In cases 5, 6, the user never receives an email, so he will never be able to complete registration.  In cases 7,8, the system does not let the "Confirm" button to be pressed until all fields are correctly filled in.  In case 9, the user can ask for a new code with a click on the appropriate button.  In cases 10,11, the user does never receive a SMS, so he will very likely not be able to proceed.</p>
--	--

*\* until the end of UC1 specification, references to "mobile app user" will be made using the word "user"*

## 2) Login of mobile app user

<b>ID</b>	UC2
<b>Name</b>	Login of mobile app user
<b>Actors</b>	Mobile app user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has the internet connection available</li> <li>• User has downloaded <i>CLup</i> app on his smartphone/tablet</li> <li>• User is registered to the service</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his smartphone</li> <li>2. User opens <i>CLup</i> app</li> <li>3. User sees the home page of the app</li> <li>4. User clicks on "Sign in" button</li> <li>5. User inserts email and password in appropriate fields</li> <li>6. User clicks on "Sign in" button</li> </ol>



	<ol style="list-style-type: none"> <li>The system checks for a matching with all registered users (with their passwords) in the DB</li> <li>The system confirms login and shows the home page</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>User is logged in</li> <li>User can use all <i>CLup</i> services</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>User does not fill some of the mandatory fields (email, password)</li> <li>The system does not find a matching with registered users (and corresponding passwords) in the DB</li> <li>User account has been blocked by <i>CLup</i></li> </ol> <p>In case 1, the system does not let the "Confirm" button to be pressed until all fields are correctly filled in.</p> <p>In cases 2,3, the system notifies an appropriate error message and goes back to the login page (where to insert again email and password).</p>

*\* until the end of UC2 specification, references to "mobile app user" will be made using the word "user"*

### 3) Mobile app user gets a ticket

<b>ID</b>	UC3
<b>Name</b>	Mobile app user gets a ticket
<b>Actors</b>	Mobile app user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>User has the internet connection available</li> <li>User has downloaded <i>CLup</i> app on his smartphone/tablet</li> <li>User is registered to the service</li> <li>User is logged in</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>User unlocks his smartphone</li> <li>User opens <i>CLup</i> app</li> <li>User sees the home page of the app</li> <li>User clicks on "Get a ticket" button</li> <li>User selects a supermarket from the list, directly or after searching</li> <li>User clicks "Confirm" button</li> <li>The system notifies "Success" message and displays expected waiting time before being called</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>User has a ticket reservation</li> <li>User sees the ticket in "All reservations" page of <i>CLup</i> application</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>User has already a "pending" (booked but not called yet) ticket reservation</li> </ol>

	<p>2. A “pending” (booked but not called yet) ticket is already associated to the user’s phone number, by a call center reservation</p> <p>3. No more tickets are available for current day and selected supermarket</p> <p>In cases 1,2, the system does not let the “Confirm” button to be pressed.</p> <p>In case 3, refer to use-case 4) ^</p>
--	--

*\* until the end of UC3 specification, references to “mobile app user” will be made using the word “user”*

#### Mobile app user receives suggestions after failure in getting a ticket

<b>ID</b>	UC4
<b>Name</b>	Mobile app user receives suggestions after failure in getting a ticket
<b>Actors</b>	Mobile app user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has the internet connection available</li> <li>• User has downloaded <i>CLup</i> app on his smartphone/tablet</li> <li>• User is registered to the service</li> <li>• User is logged in</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his smartphone</li> <li>2. User opens <i>CLup</i> app</li> <li>3. User sees the home page of the app</li> <li>4. User clicks on “Get a ticket” button</li> <li>5. User selects a supermarket from the list, directly or after searching</li> <li>6. User clicks “Confirm” button</li> <li>7. The system shows a message, telling the user about no availability for the current day and inviting him to book a visit</li> <li>8. The system suggests to get a ticket for the closest supermarket (with respect to the selected one) still available for current day</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• A suggestion is shown to the user</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. Some tickets are available for current day and selected supermarket</li> <li>2. User has already a “pending” (booked but not called yet) ticket reservation</li> </ol> <p>In case 1, refer to use-case 3) ^</p>

	In case 2, the system does not let the "Confirm" button to be pressed.
--	--

*\* until the end of UC4 specification, references to "mobile app user" will be made using the word "user"*

#### 4) Mobile app user books a visit

**LOOK FOR NOTE #**

<b>ID</b>	UC5
<b>Name</b>	Mobile app user books a visit
<b>Actors</b>	Mobile app user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has the internet connection available</li> <li>• User has downloaded <i>CLup</i> app on his smartphone/tablet</li> <li>• User is registered to the service</li> <li>• User is logged in</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his smartphone</li> <li>2. User opens <i>CLup</i> app</li> <li>3. User sees the home page of the app</li> <li>4. User clicks on "Book a visit" button</li> <li>5. User selects desired day on the appeared calendar, then the time for the visit</li> <li>6. User selects a supermarket from the list, directly or after searching**</li> <li>7. User clicks "Confirm" button</li> <li>8. The system notifies "Success" message</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User has a visit reservation</li> <li>• User sees booked visit in "All reservations" page of <i>CLup</i> application</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. No availability for selected supermarket, day and time</li> <li>2. User has already a reservation for the selected day (in any grocery shop)</li> <li>3. A reservation is already associated to the the selected day and the user's phone number, by a call center reservation</li> </ol> <p>In case 1, refer to use-case 6) ^ In cases 2,3, the system does not let the "Confirm" button to be pressed and shows a proper message to the user.</p>

*\* until the end of UC5 specification, references to "mobile app user" will be made using the word "user"*

*\*\* note that, at 6), the customer also has the option to specify the items he/she plans to buy and the expected duration for the visit*

5) Mobile app user receives suggestions after failure in booking a visit

<b>ID</b>	UC6
<b>Name</b>	Mobile app user receives suggestions after failure in booking a visit
<b>Actors</b>	Mobile app user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has the internet connection available</li> <li>• User has downloaded <i>CLup</i> app on his smartphone/tablet</li> <li>• User is registered to the service</li> <li>• User is logged in</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his smartphone</li> <li>2. User opens <i>CLup</i> app</li> <li>3. User sees the home page of the app</li> <li>4. User clicks on "Book a visit" button</li> <li>5. User selects desired day on the appeared calendar, then the time for the visit</li> <li>6. User selects a supermarket from the list, directly or after searching</li> <li>7. User clicks "Confirm" button</li> <li>8. The system shows a message telling the user about no availability for selected day, time and supermarket</li> <li>9. The systems suggests the user available slots for: same supermarket, same day and different hour (<math>\pm 2</math> hours); same supermarket, same hour and different day (<math>\pm 2</math> days); same day, same hour, different supermarket (among the five closest to the selected one)</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User has a visit reservation</li> <li>• User sees booked visit in "All reservations" page of <i>CLup</i> application</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. Availability for selected supermarket, day and time</li> <li>2. User has already a reservation for the selected day (in any grocery shop)</li> </ol> <p>In case 1, refer to use-case 5) ^  In case 2, the system does not let the "Confirm" button to be pressed and shows a proper message to the user.</p>

\* until the end of UC6 specification, references to "mobile app user" will be made using the word "user"

6) Mobile app user deletes a ticket reservation

<b>ID</b>	UC7
<b>Name</b>	Mobile app user deletes a ticket reservation
<b>Actors</b>	Mobile app user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has the internet connection available</li> <li>• User has downloaded <i>CLup</i> app on his smartphone/tablet</li> <li>• User is registered to the service</li> <li>• User is logged in</li> <li>• User has got a ticket for current day and his turn has not come yet</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his smartphone</li> <li>2. User opens <i>CLup</i> app</li> <li>3. User sees the home page of the app</li> <li>4. User clicks the button to see all his reservations in the home page of the application</li> <li>5. User clicks the deletion button for the ticket he wants to delete</li> <li>6. The system shows "Are you sure you want to delete your ticket?" message dialog</li> <li>7. User clicks "Yes" button</li> <li>8. The system notifies "Success" message</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User has no tickets for the current day</li> <li>• User does not see any ticket in "All reservations" page of <i>CLup</i> application</li> </ul>
<b>Exceptions</b>	

*\* until the end of UC7 specification, references to "mobile app user" will be made using the word "user"*

#### 7) Mobile app user deletes a visit reservation

<b>ID</b>	UC8
<b>Name</b>	Mobile app user deletes a visit reservation
<b>Actors</b>	Mobile app user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has the internet connection available</li> <li>• User has downloaded <i>CLup</i> app on his smartphone/tablet</li> <li>• User is registered to the service</li> <li>• User is logged in</li> <li>• User has booked a visit and his turn has not come yet</li> </ul>

<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his smartphone</li> <li>2. User opens <i>CLup</i> app</li> <li>3. User sees the home page of the app</li> <li>4. User clicks the button to see all his reservations in the home page of the application</li> <li>5. User clicks the deletion button for the visit he wants to delete</li> <li>6. The system shows "Are you sure you want to delete your visit?" message dialog</li> <li>7. User clicks "Yes" button</li> <li>8. The system notifies "Success" message</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User does not have the selected visit reservation anymore</li> <li>• User does not see the selected visit in "All reservations" page of <i>CLup</i> application</li> </ul>
<b>Exceptions</b>	

*\* until the end of UC8 specification, references to "mobile app user" will be made using the word "user"*

#### 8) Call center user gets a ticket

<b>ID</b>	UC9
<b>Name</b>	Call center user gets a ticket
<b>Actors</b>	Call center user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has a mobile device enabled to phone calls, with a SIM card</li> <li>• User's phone number is not blocked neither by <i>CLup</i> call center nor by <i>CLup</i> services (in general)</li> <li>• User is in a place where there is signal</li> <li>• User knows <i>CLup</i> call center telephone number</li> <li>• Date and time correspond to one of <i>CLup</i> call center's opening hours</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his mobile device</li> <li>2. User opens the program/app to make phone calls</li> <li>3. User dials <i>CLup</i> call center telephone number</li> <li>4. The IVR system lists a set of options: "press 1 to get a ticket"; "press 2 to book a visit", "press 3 to delete a reservation", "press 0 to talk to an operator"</li> <li>5. User presses 1 on the keypad</li> <li>6. The IVR system proposes the following options: "after the beep sound, pronounce the name, the town and the street of the grocery shop (if there is only one shop in the selected town, street is</li> </ol>

	<p>optional)”; “press 0 to talk to an operator”</p> <ol style="list-style-type: none"> <li>7. User hears a beep sound</li> <li>8. User pronounces name, town and (optionally) street, as suggested by the IVR system</li> <li>9. The system looks for the grocery shop which better fits the one pronounced by the user, and repeats it together with the expected waiting time (before the turn comes)</li> <li>10. The IVR system proposes the following options: “press 1 to get the ticket”; “press 2 to pronounce another grocery shop”; “press 0 to talk to an operator”</li> <li>11. User presses 1 on the keypad</li> <li>12. The IVR system proposes the following options: “press 1 to confirm you would like to receive text messages for entrance to the phone number you are using for this call”; “press 2 to insert another phone number”</li> <li>13. User presses 1 on the keypad</li> <li>14. The IVR system confirms the ticket, repeats the expected waiting time and tells the code for entrance</li> <li>15. The system ends the phone call</li> <li>16. User receives a SMS confirming the ticket reservation, together with a code for entrance</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User has a ticket reservation</li> <li>• User can see all the information about the ticket reservation (grocery shop, expected waiting time) in the received SMS</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. No more tickets are available for current day and selected supermarket</li> <li>2. The phone call ends before the user explicit on which phone number he would like to receive text messages</li> <li>3. User presses 0, in order to talk to an operator</li> <li>4. The IVR system does not recognize the grocery shop pronounced by the user</li> <li>5. User inputs a non-valid number, with respect to the options given by the IVR system (at any moment)</li> <li>6. User inputs while the IVR system is talking</li> <li>7. User inputs a non-valid phone number for SMS notifications</li> <li>8. User inputs a phone number for SMS notifications which has been blocked by <i>CLup</i></li> <li>9. A “pending” (booked but not called yet) ticket is already associated to the inserted phone number (either from a call center reservation or by a mobile app user with the same phone number)</li> </ol> <p>In case 1, the IVR system tells the user about unavailability</p>

	<p>and proposes only to pronounce another grocery shop or to talk to an operator.</p> <p>In case 2, the procedure to get a ticket is not completed and the user does not have a ticket reservation.</p> <p>In case 3, the user is put on hold until the first free operator is able to talk to him.</p> <p>In case 4, the IVR system still proposes to pronounce another grocery shop or to talk to an operator, so the user can repeat.</p> <p>In cases 5,7, the system waits until the first valid input has been written (for phone number, a specific termination sequence is given, so that the system can automatically know when the input terminates).</p> <p>In case 6, all inputs received before the sentence has finished are ignored.</p> <p>In cases 8,9, the user is invited to input another phone number in order to complete the reservation request; an information message about the cause of the fault follows.</p>
--	--

*\* until the end of UC9 specification, references to "call center user" will be made using the word "user"*

#### 9) Call center user books a visit

<b>ID</b>	UC10
<b>Name</b>	Call center user books a visit
<b>Actors</b>	Call center user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has a mobile device enabled to phone calls, with a SIM card</li> <li>• User's phone number is not blocked neither by <i>CLup</i> call center nor by <i>CLup</i> services (in general)</li> <li>• User is in a place where there is signal</li> <li>• User knows <i>CLup</i> call center telephone number</li> <li>• Date and time correspond to one of <i>CLup</i> call center's opening hours</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his mobile device</li> <li>2. User opens the program/app to make phone calls</li> <li>3. User dials <i>CLup</i> call center telephone number</li> <li>4. The IVR system lists a set of options: "press 1 to get a ticket"; "press 2 to book a visit", "press 3 to delete a reservation", "press 0 to talk to an operator"</li> <li>5. User presses 2 on the keypad</li> <li>6. The IVR system proposes the following options: "after the beep sound, pronounce the name, the</li> </ol>



	<p>town and the street of the grocery shop (if there is only one shop in the selected town, street is optional)"; "press 0 to talk to an operator"</p> <ol style="list-style-type: none"> <li>7. User hears a beep sound</li> <li>8. User pronounces name, town and (optionally) street, as suggested by the IVR system</li> <li>9. The system looks for the grocery shop which better fits the one pronounced by the user, and repeats it</li> <li>10. The IVR system proposes the following options: "press 1 to book a visit"; "press 2 to pronounce another grocery shop"; "press 0 to talk to an operator"</li> <li>11. User presses 1 on the keypad</li> <li>12. The IVR system proposes the following options: "if you want to complete the reservation soon, write with your keypad four digits, in order to specify the day and month of your visit (for example, if you would like to visit on the third of July, write zero three zero seven)"; "press # to talk to an operator"</li> <li>13. User dials a four digits number on the keypad</li> <li>14. The IVR system proposes the following options: "if you want to complete the reservation soon, write with your keypad four digits, in order to specify the exact hour of your visit (for example, if you would like to visit at half past nine, write zero nine three zero)"; "press # to talk to an operator"</li> <li>15. User dials a four digits number on the keypad</li> <li>16. The system looks for the closest earlier (in time, since supermarket is fixed) available time slot (<math>\leq</math>) and the closest later (<math>&gt;</math>) to the one selected by the user, and repeats them</li> <li>17. The IVR system proposes to press 1 to book the closest earlier (<math>\leq</math>) visit, to press 2 to book the closest later (<math>&gt;</math>) visit, to press 3 to input another date, to press 4 to input another time (keeping the date fixed), to press 4 to select another grocery shop, to press 0 to talk to an operator</li> <li>18. User presses 1 on the keypad</li> <li>19. The IVR system proposes the following options: "press 1 to confirm you would like to receive text messages for entrance to the phone number you are using for this call"; "press 2 to insert another phone number"</li> <li>20. User presses 1 on the keypad</li> <li>21. The IVR system confirms the visit reservation, repeats date, time and tells the code for entrance</li> <li>22. The system ends the phone call</li> <li>23. User receives a SMS confirming the visit reservation, together with a code for entrance</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User has a visit reservation</li> <li>• User can see all the information about the visit</li> </ul>

	reservation (grocery shop, date, time) in the received SMS
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. The phone call ends before the user explicit on which phone number he would like to receive text messages</li> <li>2. User presses the digit for talking to an operator</li> <li>3. The IVR system does not recognize the grocery shop pronounced by the user</li> <li>4. User inputs a non-valid number, with respect to the options given by the IVR system (at any moment)</li> <li>5. User dials a number with less than four digits, when he is asked to specify date/hour for the visit</li> <li>6. User dials a number with more than four digits, when he is asked to specify date/hour for the visit</li> <li>7. User inputs while the IVR system is talking</li> <li>8. User inputs a non-valid phone number for SMS notifications</li> <li>9. User inputs a phone number for SMS notifications which has been blocked by <i>CLup</i></li> <li>10. A booked visit is already associated to the inserted date and phone number (either from a call center reservation or by a mobile app user with the same phone number)</li> </ol> <p>In case 1, the procedure to book a visit is not completed and the user does not have a visit reservation.</p> <p>In case 2, the user is put on hold until the first free operator is able to talk to him.</p> <p>In case 3, the IVR system still proposes to pronounce another grocery shop or to talk to an operator, so the user can repeat.</p> <p>In cases 4,5,8, the system waits until the first valid input has been written (for phone number, a specific termination sequence is given, so that the system can automatically know when the input terminates).</p> <p>In case 6, the system ignores all digits dialed after the required four and before the IVR has listed all the options of the following step.</p> <p>In case 7, all inputs received before the sentence has finished are ignored.</p> <p>In cases 9,10, the user is invited to input another phone number in order to complete the reservation request; an information message about the cause of the fault follows.</p>

*\* until the end of UC10 specification, references to "call center user" will be made using the word "user"*

#### 10) Call center user deletes a reservation

<b>ID</b>	UC11
<b>Name</b>	Call center user deletes a reservation
<b>Actors</b>	Call center user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User has a mobile device enabled to phone calls, with a SIM card</li> <li>• User's phone number is not blocked neither by <i>CLup</i> call center nor by <i>CLup</i> services (in general)</li> <li>• User is in a place where there is signal</li> <li>• User knows <i>CLup</i> call center telephone number</li> <li>• Date and time correspond to one of <i>CLup</i> call center's opening hours</li> <li>• User has got a ticket for current day / booked a visit and his turn has not come yet</li> <li>• User has at his disposition the code associated to his reservation</li> <li>• User has at his disposition the mobile device associated to his reservation</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User unlocks his mobile device</li> <li>2. User opens the program/app to make phone calls</li> <li>3. User dials <i>CLup</i> call center telephone number</li> <li>4. The IVR system lists a set of options: "press 1 to get a ticket"; "press 2 to book a visit", "press 3 to delete a reservation", "press 0 to talk to an operator"</li> <li>5. User presses 3 on the keypad</li> <li>6. The IVR system proposes the following options: "if you want to delete your reservation soon, write with your keypad the code associated to it"; "press # to talk to an operator"</li> <li>7. User dials the code on the keypad</li> <li>8. The IVR system repeats the input code and all data connected to the reservation; then, it proposes the following options: "press 1 to delete your reservation"; "press 2 if you want to keep (not delete) this reservation, but you want to input another reservation code for deletion"; "press 0 to talk to an operator"</li> <li>9. User presses 1 on the keypad</li> <li>10. The system sends a SMS to the phone number connected to the reservation, then informs the user to input the security code just received; the option to talk to an operator is present, too</li> <li>11. User receives a SMS with a security code for deletion</li> <li>12. User dials the security code</li> <li>13. The system confirms deletion and ends the phone call</li> <li>14. User receives a SMS confirming the reservation deletion</li> </ol>

<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User does not have the selected reservation anymore</li> <li>• The code connected to the selected reservation is no more active</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. The phone call ends before the user dials the security code</li> <li>2. User presses the digit for talking to an operator</li> <li>3. User inputs a non-valid reservation code</li> <li>4. User inputs a wrong security code</li> <li>5. User inputs a non-valid number, with respect to the options given by the IVR system (at any moment)</li> <li>6. User dials a number with less digits than expected, when he is asked to specify reservation/security code</li> <li>7. User dials a number with more digits than expected, when he is asked to specify reservation/security code</li> <li>8. User inputs while the IVR system is talking</li> </ol> <p>In case 1, the procedure to delete a reservation is not complete and ticket/visit is still valid for the user.  In case 2, the user is put on hold until the first free operator is able to talk to him.  In case 3, if the code is associated with an existing reservation, the IVR system proposes to delete it or to digit another code (so, a user who dialed in a wrong way can repeat his right code); if not, the system only proposes to input another code.  In case 4, the system notifies the error and asks for a new security code, after sending another SMS. After 5 errors in the same phone call, the system ends it and the calling number is blocked for the following 30 minutes.  In cases 5,6, the system waits until the first valid input has been written.  In case 7, the system ignores all digits dialed after the required ones and before the IVR has completed the sentence of the following step.  In case 8, all inputs received before the sentence has finished are ignored.</p>

*\* until the end of UC11 specification, references to "call center user" will be made using the word "user"*

#### 11) Ticket machine user gets a ticket

<b>ID</b>	UC12
<b>Name</b>	Ticket machine user gets a ticket
<b>Actors</b>	Ticket machine user*

<b>Entry condition</b>	<ul style="list-style-type: none"> <li>User is approaching the grocery shop during its opening hours</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>User waits for its turn in order to use the ticket machine</li> <li>User clicks on the "Get a Ticket" button</li> <li>The ticket machine shows the expected waiting time before being called</li> <li>User clicks on "Confirm" button</li> <li>A paper with the QR code and a unique queue number gets printed</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>User has a ticket reservation</li> <li>User can wait for its queue number to show on the display, as its turn to enter begins</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>There are no tickets available for the current day</li> <li>After clicking on the "Get a Ticket" button, the user clicks the "Cancel" button</li> <li>After clicking on the "Get a Ticket" button, the user remains inactive for more than 3 minutes</li> </ol> <p>In case 1, refer to use-case 13) ^  In case 2 or 3, the ticket machine returns to its initial state</p>

*\* until the end of UC12 specification, references to "ticket machine user" will be made using the word "user"*

#### 12) Ticket machine user books a visit

<b>ID</b>	UC13
<b>Name</b>	Ticket machine user books a visit
<b>Actors</b>	Ticket machine user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>User is approaching the grocery shop during its opening hours</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>User waits for its turn in order to use the ticket machine</li> <li>User clicks on the "Get a Ticket" button</li> <li>The ticket machine shows an error message telling the user that there are no available tickets for the current day</li> <li>User clicks on "Book the first available slot" button</li> <li>The ticket machine shows the date and time of the visit</li> <li>User clicks on "Confirm" button</li> <li>A paper with the QR code and a unique queue number gets printed</li> </ol>

<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User has a visit reservation</li> <li>• User can wait for its visit date and time to come</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. After clicking on the "Get a Ticket" button, the user remains inactive for more than 3 minutes</li> <li>2. After clicking on the "Get a Ticket" button, the user clicks on the "Cancel" button</li> <li>3. After clicking on the "Book the first available slot" button, the user remains inactive for more than 3 minutes</li> <li>4. After clicking on the "Book the first available slot" button, the user clicks on the "Cancel" button</li> </ol> <p>In case 1-2-3-4, the ticket machine returns to its initial state</p>

*\* until the end of UC13 specification, references to "ticket machine user" will be made using the word "user"*

### 13) Ticket machine user deletes a reservation

<b>ID</b>	UC14
<b>Name</b>	Ticket machine user deletes a reservation
<b>Actors</b>	Ticket machine user*
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>• User is approaching the grocery shop during its opening hours</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>1. User waits for its turn in order to use the ticket machine</li> <li>2. User clicks on the "Delete a reservation" button</li> <li>3. The ticket machine shows a message telling the user to put the reservation QR code in front of the barcode reader of the machine</li> <li>4. The ticket machine shows a message asking the user if he/she really wants to delete the reservation, showing the reservation details</li> <li>5. User clicks on "Confirm" button</li> <li>6. The ticket machine shows a confirmation message telling the user that the reservation has been deleted</li> </ol>
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• The QR code of the reservation deleted by the user is no longer valid</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>1. After clicking on the "Delete a reservation" button, the user remains inactive for more than 3 minutes</li> <li>2. After clicking on the "Delete a reservation" button, the user clicks on the "Cancel" button</li> <li>3. When the ticket machine asks the user if he/she really wants to delete the reservation, the user</li> </ol>

	<p>remains inactive for more than 3 minutes</p> <ol style="list-style-type: none"> <li>When the ticket machine asks the user if he/she really wants to delete the reservation, the user clicks on the "Cancel" button</li> <li>The ticket machine couldn't find a reservation associated to the QR code showed by the user</li> </ol> <p>In case 1-2-3-4, the ticket machine returns to its initial state In case 5, the ticket machine shows an error message</p>
--	--

*\* until the end of UC14 specification, references to "ticket machine user" will be made using the word "user"*

#### 14) User enters the grocery shop

<b>ID</b>	UC14
<b>Name</b>	User enters the grocery shop
<b>Actors</b>	User
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>User has a ticket/visit reservation</li> <li>User has received a notification (mobile app and/or SMS) for entering, or has seen the call for the number associated to his reservation on the display (outside the supermarket)</li> <li>User is outside the grocery shop's entrance, in front of the code reader</li> <li>User has the code for entrance at his immediate disposition</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>User activates the code reader with a movement near the sensor, as shown by an information plaque</li> <li>Code reader screen shows a message, inviting the user to scan/input the code</li> <li>User shows the QR code for entrance to the reader (mobile app user, ticket machine user), or inputs the code with the reader's keypad after wearing disposable gloves (call center user, mobile app user who prefers/has the necessity to use the code received by SMS)</li> <li>The system processes the code, checking for the possibility to enter</li> <li>Doors open and code reader screen shows a blank page</li> <li>User gets into the grocery shop</li> <li>Doors close</li> </ol>

<b>Exit condition</b>	<ul style="list-style-type: none"> <li>User is inside the supermarket and can do the grocery shopping</li> </ul>
<b>Exceptions</b>	<ol style="list-style-type: none"> <li>The movement near the sensor does not turn the code reader on</li> <li>The code for entrance is not recognized (impossibility to read QR code and/or wrong code)</li> <li>The code for entrance expired</li> </ol> <p>In case 1, the user can try again with a movement near the sensor.</p> <p>In case 2, the code reader shows an error message, turns off and the user can repeat the whole procedure.</p> <p>In case 3, the code reader screen shows a message to inform about the expiration.</p>

#### 15) User pays and exits the grocery shop

<b>ID</b>	UC15
<b>Name</b>	User pays and exits the grocery shop
<b>Actors</b>	User, cashier (in case of non-automatic check-out)
<b>Entry condition</b>	<ul style="list-style-type: none"> <li>User has a ticket/visit reservation</li> <li>User has already received a notification (mobile app and/or SMS) for entering, or has already seen the call for the number associated to his reservation on the display (outside the supermarket); he has already entered the supermarket</li> <li>User is at the check-out inside the grocery shop</li> <li>User has the code for entrance at his disposition</li> </ul>
<b>Event flow</b>	<ol style="list-style-type: none"> <li>User (in case of automatic cashier) / cashier (in case of traditional check-out) scans all the products the customer would like to buy</li> <li>The cashier asks the user to show/input the code associated to his reservation to the code reader</li> <li>The cashier activates the code reader</li> <li>User picks the code associated to his reservation</li> <li>User shows the QR code to the reader (mobile app user, ticket machine user), or inputs the code with the reader's keypad after wearing disposable gloves (call center user, mobile app user who prefers/has the necessity to use the code received by SMS)</li> <li>The system processes and checks the code</li> <li>The system recognizes the code and notifies the cashier</li> <li>The cashier asks the user to pay</li> </ol>



	9. User pays 10. User leaves the check-out
<b>Exit condition</b>	<ul style="list-style-type: none"> <li>• User is going to exit the grocery shop</li> <li>• The system knows that the user is going to exit soon</li> </ul>
<b>Exceptions</b>	1. The code associated to the reservation is not recognized (impossibility to read QR code and/or wrong code)  In case 1, the user is asked to show/input the code again, after making sure that the code itself is the right one.

In case of a user who is going to buy nothing, the exit way turnstyle opens after the customer shows/inputs the code to the code reader placed there; the way is analogous to UC14 for entrance (but here it is valid for exiting).

## NOTES

° the situation is completely described in another use-case, so read and refer only to it  
 # should we add to the event flow the possibilities users have to include the estimated time of his visit and the categories of product he's going to buy?

**Time:** 1h30 19/11/2020  
 Andrea and Roberto

**Time:** 30min 20/11/2020  
 Andrea

**Time:** 30min 20/11/2020  
 Roberto

## PRODUCT FUNCTIONS

- **Get a ticket**

Users who want to reserve a ticket for accessing the supermarket, without lining up physically, can get it using *CLup* service. The functionality of getting a ticket can be achieved in two ways: the first one is using a smartphone/tablet, the second one with a ticket machine (outside the grocery shop).

Tickets correspond to "virtual" lining up: when someone gets a ticket, he becomes the last one to wait in the queue; when time slots finish for the current day (according to the supermarket's opening hours), no more tickets are assigned. A relevant aspect is that visits fill the same queue as tickets, but they can be reserved in advance (so, also some days/weeks before the date of the visit itself). Every customer can download *CLup* application on a mobile device; in order to get a ticket with the app, the guest needs to be registered and specify which grocery shop he wants to visit.

In alternative, a person can go directly to the supermarket and get his ticket (if available for that day) using a ticket machine: in this case, there is no need to specify the grocery shop (implicit: it is the closest one), nor to be registered; tickets are printed on paper in the form of QR code, with the addition of a 6-digits number identifying the turn in the waiting queue.

- **Book a visit**

Users, through *CLup* service, can book a visit for accessing the supermarket, avoiding physical lining up. The functionality of booking a visit can be achieved in two ways: the first one is using a smartphone/tablet, the second one with a ticket machine (outside the grocery shop).

Every guest can download *CLup* mobile application; in order to book a visit with the app, he needs to be registered and specify which grocery shop he wants to visit, in which day and time. A user might also specify the approximate duration of his visit and the categories of items he is going to buy, in order to help the system to coordinate other customers' visits/entrances with tickets.

Alternatively, a visit can be booked directly from outside the supermarket, using a ticket machine: in this case, there is no need to be registered, nor to specify the grocery shop; visit reservations are printed on paper in the form of QR code (with the addition of a reservation receipt containing all relevant information: supermarket, date, time).

- **Delete a reservation**

Users can delete reservations connected to a ticket or to a booked visit. In order to do so with the mobile application, the user should open the page regarding information about his tickets/visits and select the one he wants to delete.

The equivalent operation can be performed for tickets or visits booked using a ticket machine: the customer should scan the printed QR code with the machine's scanner.

- ***CLup* suggestion mechanism**

The system, for registered customers using the mobile application, is able to suggest time slots (for visits) based on specific users' habits. In particular, *CLup* stores all data about customers, days and time for visits, as well as most visited supermarkets; analyzing this data, the application can send customized

notifications when an attractive visit can be booked.

Other features for suggestions include giving alternatives after failures in getting a ticket or booking a visit (because no slots are available): for tickets, the system proposes to get one in the closest supermarket (with respect to the selected one) still available for current day; for visits, suggestion coincides with the proposal of a slot for a different hour/a different day/a close supermarket. These features are available only for customers using the mobile application.

- ***CLup* avoids people crowding inside the supermarket**

The system can manage slots according to known (or inferred) duration of customers' visits and categories of items to buy. The general idea is that, knowing more information, *CLup* can more precisely stagger reservations in order to avoid crowds inside the supermarket. Data is collected by the system; the user can specify it (approximate visit duration and categories of items) when booking a visit. This feature is customized for mobile application users; for customers without the app, estimations are performed (based on average data).

- ***CLup* sends notifications (based on GPS position)**

The system, for registered customers using the mobile application and having an active GPS connection on their smartphones, is able to notify them when their reservation (ticket or visit) is approaching. In particular, *CLup* computes time to get to the supermarket from the user's actual position, and sends him a notification so that he can get on time to the grocery shop.

If GPS connection is not active, the user still receives notifications about an approaching reservation, but only in predefined times.

Which transportation means should be considered? Car, walking, ...? Maybe ask the user (or calculate only car time, but in any case we have to specify).

**Time:** 1h30

Andrea and Roberto

## Users Characteristics

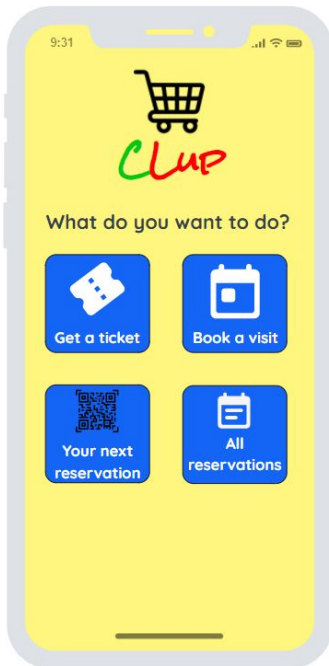
Considering that supermarkets usually sell primary goods, the range of possible users of CLup is very wide, basically any person is a potential user of the system. If we consider the various types of customers, we can identify the following three categories:

- **Mobile App User:** a user who can take advantage of all the functionality of CLup through the use of the mobile application. Once the user is registered and logged in, he/she is able to get a ticket or book a visit at any shop, and handle all the reservations that have been made. By taking advantage of the real time GPS position and mobile app push notifications, the user can be notified to know when it's time to leave from the current position and when is its turn to enter the shop.
- **Ticket Machine User:** a user who does not have the mobile app, therefore is not registered and wants to get a ticket as he arrives at the supermarket. This kind of user is only able to get tickets or, if there are no tickets available, it can book a visit for the first available slot.
- **Call Center User:** a user who does not have the mobile app, therefore is not registered but still wants to get a ticket or book a visit before getting to the supermarket. Giving its mobile number to the call center operator, the user can be notified by SMS when its turn has come, avoiding the need to line up outside the shop.

21/11/2020

### 3.1.1 USER INTERFACES

#### 3.1.1.1 CLup mobile application interface



- **Home page**

The home page of *CLup* mobile application for logged users shows the following features:

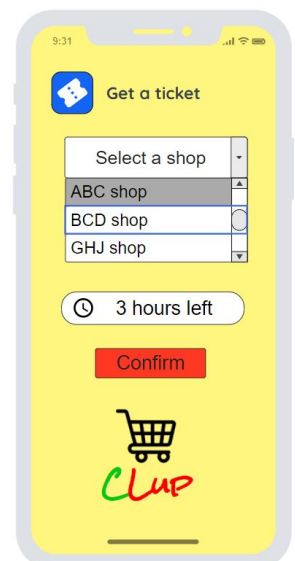
- Get a ticket
- Book a visit
- See the list of reservations
- Manager user (login/logout)

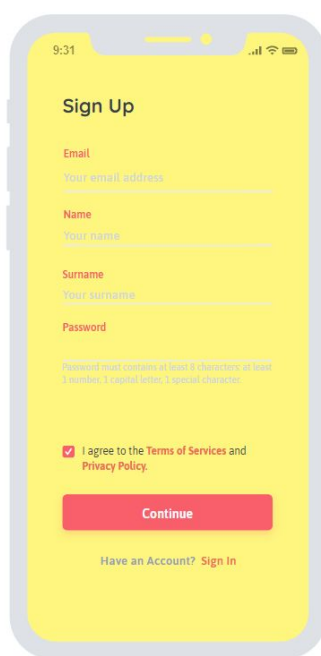
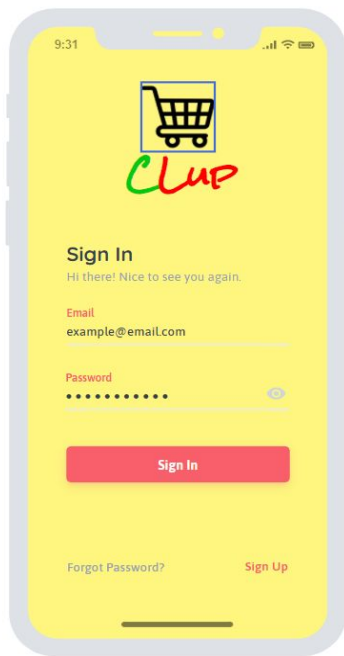
If a user is not logged in, CLup only shows a button to sign in/sign up.

- **Get a ticket**

This page allows users to choose the supermarket from a list (shops can also be searched using associated keywords). Once the grocery store is chosen, users should only confirm to get the ticket reservation.

When users select the shop, a message shows expected time before being called for entrance.



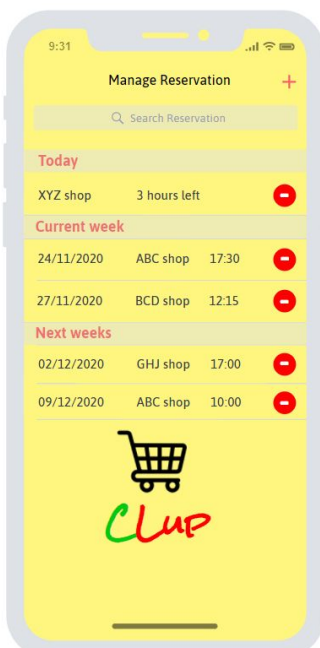
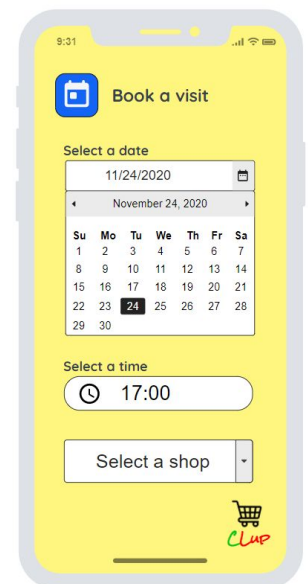


- **Sign up/sign in**

This page allows users to register to the system or log in (if already registered). The user interface allows to insert the email and password; in case of sign up, also name and surname must be added.

- **Book a visit**

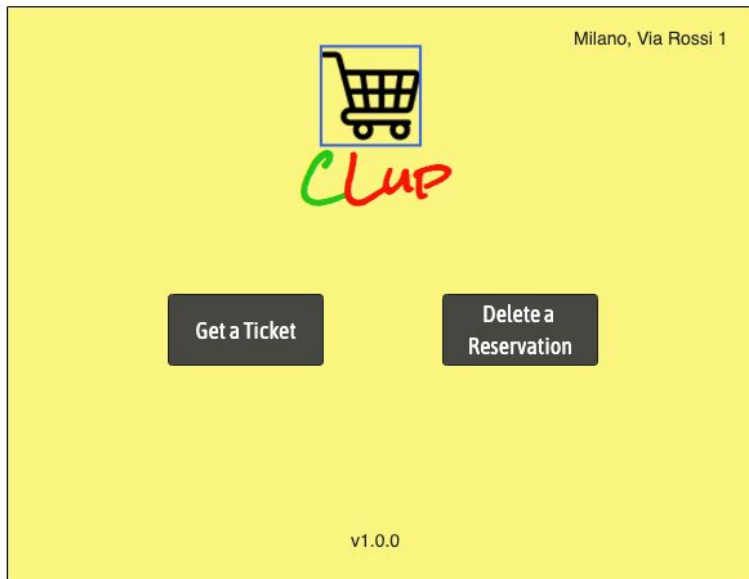
This page allows users to choose the supermarket from a list (where customers can also search), the day and the time for the visit from a calendar. In addition, there are two optional fields for the duration of the visit and the categories of items to buy: both duration and categories should be chosen from a predefined list.



- **List of reservations**

The page allows users to get information about estimated remaining time before the turn (in case of ticket) or day and time of the visit. In addition, there is an option for deleting any reservation. Clicking on a single reservation allows the user to update the estimated duration and the categories of products it is going to buy.

### 3.1.1.1 CLup ticket machine interface



- **Home page**

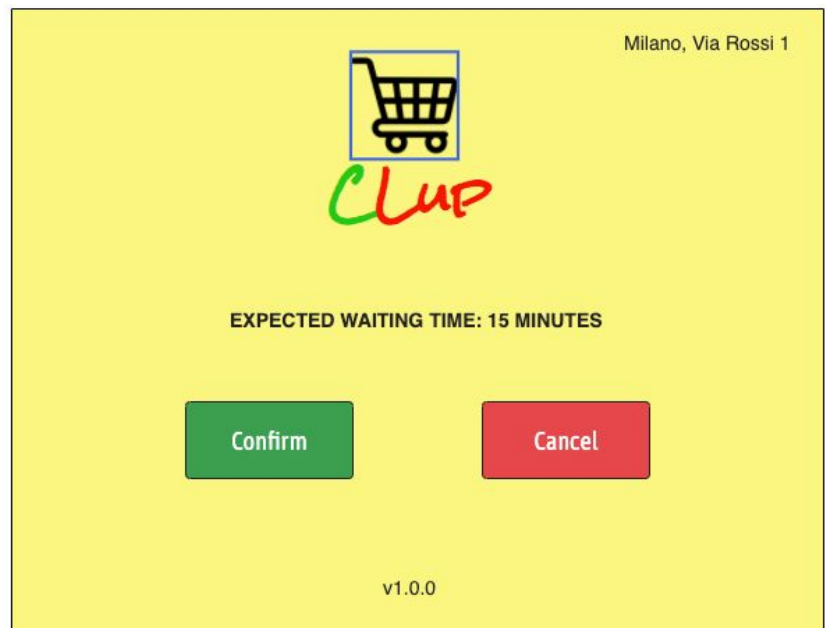
The home page of CLup ticket machine system shows the following options:

- Get a ticket
- Delete a reservation

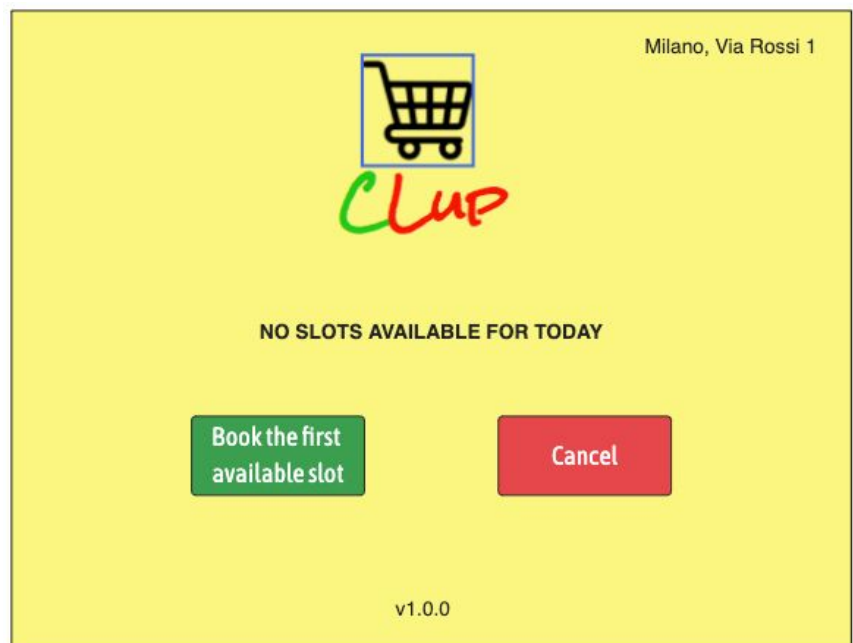
Customers do not need to register or log in (in fact, there is no option for it).

- **Get a Ticket**

This page shows the expected waiting time if any slot is available for the current day.



If no slots are available, an option to book the first available slot is given to the user.





- **Book a visit**

This page shows the expected scheduled date and time for the visit the user is booking.

### 3.1.2 HARDWARE INTERFACES

Users must have a mobile device (smartphone/tablet) equipped with a GPS system and mobile data (Internet connection).

Ticket machines are equipped with a touchscreen display, a QR code scanner and Internet access.

A QR code scanner is present at the grocery store's entrance: it is a scanner which can be activated through a movement sensor on the upper part of it. The scanner is connected with the supermarket's doors, which open when an active QR code has been scanned.

There are screens outside the grocery store which are used for displaying turns, and they must be big enough so that reading from 15 metres away is possible.

*Same for QR code scanners at exit doors? We must decide whether to have them (maybe useful for some reasonments about crowded areas, based on known categories of items users buy), or rely only on numbers (and a person exits after he pays at the check-out: wait two minutes and call the next). -> NO, QR CODE IS SCANNED BY THE CASHIER WHEN THE CUSTOMER PAYS*

### 3.1.4 COMMUNICATION INTERFACES

HTTPS protocol is used to grant secure data transmission over the Internet.

### 3.2 SOFTWARE INTERFACES

Regarding the frontend for the mobile application, the app should be compatible with the following operating system:



- Android (from version 5.0 Lollipop in order to be compatible with the 94.1% of the devices [data regarding December 2020, source: [android.com](https://developer.android.com/about/versions/lollipop)])
- iOS (from version 12.0 in order to be compatible with the 94% of the devices [source: [apple.com](https://www.apple.com/newsroom/2018/09/180920-ios120/)])

The mobile app must interact with the following external software component:

- Geolocalization service (e.g Google Maps SDK)
- Push Notification Service (e.g Google Firebase Cloud Messaging)

Moreover, the CLup's mobile application interacts with CLup's backend services. In particular:

- User authentication services
- Reservation handling services (also used by the operators in call center and the ticket machine)

The backend components make use of a database (relational database or NoSQL) to store data about Users and Reservations. Using a database implies the integration of a software library to interact with the chosen DBMS.

### **3.3 PERFORMANCE REQUIREMENTS**

The system should be able to answer a ticket/visit reservation request in less than 10 seconds. In case there is no availability for the requested reservation, it should provide appropriate suggestions in a reasonable time (together with the requested answer or at most 3 seconds later). Moreover, the system should delete a reservation (when requested) in less than 5 seconds.

CLup, after monitoring client's visit time and duration habits, should be able to mine them within 3 days.

The system should monitor GPS positions of users who have a reservation in the next hours, in order to be able to send notifications according to requirements.

### **3.4 DESIGN CONSTRAINTS**

#### **3.4.1 Standards compliance**

There is no particular necessity to deviate from usual web-standards suggested by the World Wide Web Consortium (W3C); for this reason, the development of the system should follow W3C recommendations.

Any further compliance, if needed, will be specified during the design analysis.

#### **3.4.2 Hardware limitations**

The mobile application can be installed only on a smartphone/tablet equipped with a GPS system and the possibility to activate mobile data.

**Time:** 2h  
Andrea and Roberto

## 3.5 SOFTWARE SYSTEM ATTRIBUTES

### 3.5.1 Reliability

*Customers Line-up* should be available 24/7 in order to book visits and manage reservations whenever users want. However, the crucial time in which the system should work coincides with supermarkets' opening hours: for this reason, 4-6 hours of maintenance do not represent a problem, if programmed outside opening hours.

### 3.5.2 Availability

*CLup* does not have a very critical nature (it is not connected with emergency situations); however, in case the system is down during opening hours, supermarket's business becomes much more complicated: entrances have to be managed directly from its employees and ticket reservations become impossible; the only thing employees could do is to guarantee the entrance to an appropriate number of people that are waiting outside, without realizing any other feature that the software system guarantees. For these reasons, 95% availability of the system should be guaranteed.

### 3.5.3 Security

The system uses HTTPS for a secure communication between users and the server. Moreover, all passwords are encrypted and, in case of recovery, they are not sent in clear.

Possible additional choices will be discussed in the *Design Document*.

### 3.5.4 Maintainability

Code should follow good software engineering practices and be properly commented. In addition, the use of proper design patterns, as suggested in the *Design Document*, is mandatory.

### 3.5.5 Portability

*Customers Line-up's* frontend is composed mainly by a mobile application: to ensure an high grade of portability between Android and iOS an hybrid framework should be used to build the frontend, so to be able to share the most of the code between the two platforms.

**Time:** 1h

Andrea and Roberto

**Time:** 1h Roberto images

25/11 2h mapping goal-requirements-domain assumptions

Andrea, Cosimo and Roberto

26/11 2h UML class-diagram

Andrea, Cosimo and Roberto

28/11 2h review

Andrea, Cosimo and Roberto

29/11/2020:

Roberto 1h: UML use-cases

01/12/2020:

Roberto 1h: UML use-cases

02/12/2020:

Roberto 1h: UML use-cases

03/12/2020:

Cosimo 2h30: UML use-cases, user characteristics, software interfaces, review

Roberto 2h: scenarios, UML review

Andrea 2h: sections 2.A, 2.D, UML use-cases, review

04/12/2020:

Roberto 2h30 mapping use cases-requirements

Cosimo 2h30 Ticket Machine UI, review

## Mapping USE CASES - REQUIREMENTS

<b>UC1</b>	Registration of mobile app user
<b>REQ</b>	A user application is associated to a profile (personal profile)
<b>REQ</b>	The system allows users to book one or more visits with the mobile application
<b>REQ</b>	The system, in case of mobile application, sends notifications basing on user's actual position and time remaining before his turn
<b>REQ</b>	The system tracks user habits in order to detect its habitual shop and date/time of visit, in order to provide booking suggestions in advance if any slot is available
<b>ALTR O?</b>	

<b>UC2</b>	Login of mobile app user
<b>REQ</b>	A user application is associated to a profile (personal profile)
<b>REQ</b>	The system allows users to book one or more visits with the mobile application
<b>REQ</b>	The system allows a user application to get a ticket only when no other ticket is "pending" (booked but not called yet) for that profile

<b>REQ</b>	The system allows to insert a list of items (that user would like to buy) when booking a visit
<b>ALTR O?</b>	

<b>UC3</b>	Mobile app user gets a ticket
<b>REQ</b>	A user application is associated to a profile (personal profile)
<b>REQ</b>	The system does not allow for entrance a person who has not booked a visit or got a ticket
<b>REQ</b>	The system allows a user application to get a ticket only when no other ticket is "pending" (booked but not called yet) for that profile
<b>REQ</b>	The system sends a notification two minutes before entrance (only in case of user application and call center reservation) ( <b>METTIAMO?</b> )
<b>REQ</b>	The system generates a unique QR code associated to every ticket/visit booked from user application
<b>REQ</b>	The system, in case of mobile application, sends notifications basing on user's actual position and time remaining before his turn
<b>REQ</b>	The system allows users to get information about estimated time before being called
<b>REQ</b>	The system is able to check availability for all the supermarket included in it
<b>ALTR O?</b>	

<b>UC5</b>	Mobile app user books a visit
<b>REQ</b>	A user application is associated to a profile (personal profile)
<b>REQ</b>	The system allows users to book one or more visits with the mobile application
<b>REQ</b>	The system does not allow for entrance a person who has not booked a

	visit or got a ticket
<b>REQ</b>	The system sends a notification two minutes before entrance (only in case of user application and call center reservation) ( <b>METTIAMO?</b> )
<b>REQ</b>	The system allows the user to book slots for shopping for the next month from current date and time
<b>REQ</b>	The system allows to insert a list of items (that user would like to buy) when booking a visit (in case of mobile application)
<b>REQ</b>	The system generates a unique QR code associated to every ticket/visit booked from user application
<b>REQ</b>	The system, in case of mobile application, sends notifications basing on user's actual position and time remaining before his turn
<b>REQ</b>	The system manages available slots depending on statistics built on specific user's habits (usual time spent in the supermarket, items/categories of items purchased)
<b>REQ</b>	The system is able to check availability for all the supermarket included in it

<b>UC6</b>	Mobile app user receives suggestions after failure in booking a visit
<b>REQ</b>	A user application is associated to a profile (personal profile)
<b>REQ</b>	The system does not allow a person to enter if shopping centre's maximum capacity has been reached
<b>REQ</b>	The system allows users to book one or more visits with the mobile application
<b>REQ</b>	The system allows the user to book slots for shopping for the next month from current date and time
<b>REQ</b>	The system is able to build statistics (time spent in the supermarket, items/categories of items purchased)
<b>REQ</b>	The system manages available slots depending on statistics built on specific user's habits (usual time spent in the supermarket, items/categories of items purchased)
<b>REQ</b>	The system is able to check availability for all the supermarket included in it
<b>ALTR O?</b>	

<b>UC7</b>	Mobile app user deletes a ticket reservation
<b>REQ</b>	A user application is associated to a profile (personal profile)
<b>REQ</b>	The system allows user to delete a reservation
<b>ALTR O?</b>	

<b>UC8</b>	Mobile app user deletes a visit reservation
<b>REQ</b>	A user application is associated to a profile (personal profile)
<b>REQ</b>	The system allows user to delete a reservation
<b>ALTR O?</b>	

<b>UC9</b>	Call center user gets a ticket
<b>REQ</b>	The system does not allow for entrance a person who has not booked a visit or got a ticket
<b>REQ</b>	The system sends a notification two minutes before entrance (only in case of user application and call center reservation)
<b>REQ</b>	The system generates a unique numeric code associated to every ticket/visit booked calling the call center
<b>REQ</b>	Call centers are open in the supermarket's opening hours
<b>REQ</b>	The system allows users to get a ticket reservation calling the call center
<b>REQ</b>	The system is able to check availability for all the supermarket included in it
<b>ALTR O?</b>	

<b>UC10</b>	Call center user books a visit
<b>REQ</b>	The system allows users to book one or more visits calling the call center

<b>REQ</b>	The system does not allow for entrance a person who has not booked a visit or got a ticket
<b>REQ</b>	The system sends a notification two minutes before entrance (only in case of user application and call center reservation)
<b>REQ</b>	The system allows the user to book slots for shopping for the next month from current date and time
<b>REQ</b>	The system generates a unique numeric code associated to every ticket/visit booked calling the call center
<b>REQ</b>	Call centers are open in the supermarket's opening hours
<b>REQ</b>	The system allows users to book a visit calling the call center
<b>REQ</b>	The system is able to check availability for all the supermarket included in it
<b>ALTR O?</b>	

<b>UC11</b>	Call center user deletes a reservation
<b>REQ</b>	The system allows users to delete a reservation
<b>REQ</b>	Call centers are open in the supermarket's opening hours
<b>ALTR O?</b>	

<b>UC12</b>	Ticket machine user gets a ticket
<b>REQ</b>	The system does not allow for entrance a person who has not booked a visit or got a ticket
<b>REQ</b>	The system allows "ticket-machine" application to get a ticket
<b>REQ</b>	The system generates a unique numeric code associated to every ticket/visit booked calling the call center
<b>REQ</b>	The system generates a unique QR code associated to every ticket/visit booked from a ticket machine
<b>REQ</b>	The system generates a unique numeric code for the position in the waiting "queue" associated to every ticket/visit booked from a ticket machine
<b>REQ</b>	The system allows ticket machine user to get information about estimated time before being called through the display in the



	supermarket
<b>REQ</b>	The system is able to check availability for all the supermarket included in it
<b>ALTR O?</b>	

<b>UC13</b>	Ticket machine user books a visit
<b>REQ</b>	The system allows users to book one visit with a ticket machine only if no slots for tickets are available for that supermarket in the current day
<b>REQ</b>	The system does not allow for entrance a person who has not booked a visit or got a ticket
<b>REQ</b>	The system allows the user to book slots for shopping for the next month from current date and time
<b>REQ</b>	The system generates a unique QR code associated to every ticket/visit booked from a ticket machine
<b>REQ</b>	The system generates a unique numeric code for the position in the waiting "queue" associated to every ticket/visit booked from a ticket machine
<b>REQ</b>	The system allows ticket machine user to get information about estimated time before being called through the display in the supermarket
<b>REQ</b>	The system is able to check availability for all the supermarket included in it
<b>ALTR O?</b>	

<b>UC14</b>	Ticket machine user deletes a reservation
<b>REQ</b>	The system allows users to delete a reservation
<b>REQ</b>	Ticket machines are available in the supermarket's opening hours

06/12/2020:

Roberto: 1h30 UC sequence diagrams and account suspension

Cosimo: 2h review RASD (product functions)

### **AGGIUNZIONE CACCIATI DAL SISTEMA**

The system can suspend accounts for those people who did not show up at the shop despite having reserved a ticket/visit several times. The same idea works for call center users too. In fact, call centers will not take any reservation by people related to phone numbers that previously reserved tickets/visits and never showed up at the shop.

## 5.EFFORT SPENT

### Cosimo Sguanci

Date	Topic	Hours
25/11/20	Review	1:30h
02/12/20	State Charts	1:30h
03/12/20	Ticket machine use cases	1:30h
03/12/20	Sequence Diagrams	1h
04/12/20	Use-cases sequence diagrams, user characteristics and software interfaces	2:30h
05/12/20	Review data collection	1h
06/12/20	Review data collection	1h
09/12/20	Section 2.D.2 (Dependencies)	45m
10/12/20	Alloy in final RASD WIP	1:30h

### Roberto Spatafora

Date	Topic	Hours
20/11/20	Use cases	30m
25/11/20	User interfaces	1:30h
26/11/20	User interfaces	30m
29/11/20	Use cases and sequence diagrams	1h
01/12/20	Use cases and sequence diagrams	1h
02/12/20	Use cases and sequence diagrams	1h
03/12/20	Scenarios	1:30h
03/12/20	UML review	30m
04/12/20	Mapping UseCases-Requirements	2:30h
06/12/20	Use cases sequence diagrams	1h
06/12/20	Account suspension function	30m

09/12/20	Review	1:30h
----------	--------	-------

## Andrea Vergani

Date	Topic	Hours
16/11/20	Review of notes of COSA meeting	15m
17/11/20	Revision of document	30m
18/11/20	Revision of scenarios	30m
21/11/20	Chapter 1: sections 1.A, 1.B, 1.C	2:30h
24/11/20	Chapter 1 completed and review of all the document	1:30h
27/11/20	Update use-cases and section 2.B	2:45h
28/11/20	Shared phenomena	15m
29/11/20	Use-cases	30m
30/11/20	Use-cases	45m
01/12/20	Chapter 2: section 2.A.1	1h
04/12/20	Use-cases sequence diagram	45m
05/12/20	Use-cases and sequence diagrams	1:15h
06/12/20	Use-cases sequence diagrams	3h
07/12/20	Use-cases and sequence diagrams	15m

## Group

Date	Topic	Hours
29/10/20	Initial brainstorming	2h
17/11/20	World and Shared phenomena	1h
17/11/20	Goals, Domain Assumptions, Requirements	1:30h
18/11/20	Scenarios	1:30h
19/11/20	Use-Cases	1:30h
20/11/20	Scenarios	30m

20/11/20	Product functions	1:30h
21/11/20	Chapter 3: sections 3.1.1, 3.1.2, 3.1.4, 3.3, 3.4	2h
22/11/20	Chapter 3: section 3.5	1h
22/11/20	Review	15m
25/11/20	Mapping Goal-Requirements-Domain assumptions	2h
26/11/20	UML Class Diagram	2h
28/11/20	Review and Updates	2h
29/11/20	Use cases and Sequence Diagrams	2h
01/12/20	Use cases	1h
03/12/20	Discussion and Review	1h
03/12/20	Chapter 2: sections 2.A.1, 2.A.2	1h
04/12/20	Use cases and Sequence Diagrams	2:30h
07/12/20	Alloy	4:30h
08/12/20	Alloy	5h
09/12/20	Alloy	45m
12/12/20	Use-case diagram	1h