



POLITECNICO
MILANO 1863

RASD
**Requirement Analysis and Specification
Document**

AY 2020-2021

Students:

Matteo Makovec
Lorenzo Male
Gabriele Morelli

Contents

1	Introduction	1
1.1	Purpose	1
1.2	Scope	1
1.3	Definitions, acronyms, abbreviations	1
1.4	Revision history	3
1.5	Effort	4
1.6	Reference Documents	5
2	Overall description	5
2.1	Product perspective	5
2.1.1	Scenarios	6
2.2	Phenomena	8
2.2.1	Class diagram	9
2.2.2	State charts	9
2.3	Product functions	10
2.4	User characteristics	13
2.5	Assumptions, dependencies and constraints	13
3	Specific requirements	14
3.1	External interface requirements	14
3.1.1	User interfaces	14
3.1.2	Hardware interfaces	16
3.1.3	Software interfaces	16
3.1.4	Communication interfaces	16
3.2	Functional requirements	16
3.2.1	Goals	16
3.3	Functional requirements	17
3.4	Goal mapping	18
3.5	Use cases and sequence diagrams	22
3.5.1	Use cases diagram	22
3.6	Performance requirement - Non-functional requirements	34
3.7	Design constraint	34
3.8	Software system attributes	34
3.8.1	Reliability	34
3.8.2	Availability	34
3.8.3	Security	35
3.8.4	Maintainability	35
3.8.5	Portability	35
4	Alloy	35

1 Introduction

1.1 Purpose

The present document represents the Requirement Analysis and Specification Document (RASD) for the CLup system.

The goals of this document are to analyze real needs of the customers in order to model the system to be, to describe it in terms of functional and nonfunctional requirements, to show the constraints and the limit of the software and indicate the typical use cases that will occur after the release.

This document is addressed to the stakeholders and could be used as a contractual basis.

1.2 Scope

In order to face the problem of social distancing, the software wants to give the possibility to users to queue and book visits for stores from home, avoiding crowds outside of them; at the same time, the software regulates the influx of people inside the building, avoiding crowds inside stores, giving to store managers the possibility to monitor entrances.

Finally, the software can speed up the process, always guaranteeing safety distance between people.

1.3 Definitions, acronyms, abbreviations

DEFINITIONS:

TERM	DEFINITION
Employee	He/she works in a store and he/she does the functionality of “mediator”, performs the “physical booking” functionality for customers and scans the QR codes at the entrance and at the exit
Store Manager	He/she is in charge of monitoring his/her store and subscribes the store to the CLup system sending to the CLup team the proper documentation
Booking	It allows the user to book a visit for a specific store
Ticket	It allows the user to queue for a specific store
Physical booking	It allows the user to book physically a visit for a specific store
Visit	It represents the “user journey” internally of the store, tracking his/her entrance and the exit
Memo	It is the software interface provided to the user to simplify his/her user experience. It is an interface which aggregate all the information needed by the user to enter the store after

	having performed the “take a ticket”, the “book a visit” or the “physical booking” functionalities
EstimatedDuration	It is the user's estimation of his/her visit duration
Store	It is the supermarket subscribed to the <u>CLup</u> system
Device	It is a hardware component used by the actors to use our software, it can be a smartphone or a tablet
CLup application	It defines the mobile application downloaded in the user's device
System	It defines the whole back-end architecture
CLup system	The whole software, both the mobile app and the system
Customer	It is the client of a store who uses the “physical booking” functionality
User	It is the client of a store that uses the <u>CLup</u> application
Long-term customer	An User that has already taken a ticket or booked a visit
Appointment	It is a visit that is already scheduled
Estimated waiting time	It is a estimation of the time that the user has to wait for his/her turn
Recommended departure time	It suggests the user the best time to begin moving from the starting address in order to arrive right on time at the store
Starting address	It is the address from which the user will begin moving to go to the store
Maximum capacity	It is the maximum number of people admitted at the same time in a store
Queue	Virtual sequence of users waiting for their turn
Line-up	Physical sequence of people waiting for their turn
Slot	It is the unity of measure of the “book a visit” functionality which represents an interval of time

ACRONYMS:

TERM	DEFINITION
GPS	Global Positioning System
RASD	Requirement Analysis and Specification Document
QR code	Quick Response code

SYNOMYS:

TERM	SYNONYM
Supermarket, Grocery shop	Synonym of Store
Mobile app	Synonym of Clup application

1.4 Revision history

DATE	MODIFICATION
23/11	Release

1.5 Effort

In this section the tables of each team member will follow, indicating their work done in the project. It should be taken into account that of the hours declared, 35 hours were carried out in a group and therefore are counted in each table

Matteo Makovec

TOPIC	HOURS
Analysis of the assignment	5 hours
Purpose & Scope	2 hours
Scenarios and Class diagram	11 hours
Use cases and Sequence diagrams	10 hours
Phenomena, Goals, Functional Requirements and Domain Assumptions	10 hours
External Interfaces Requirements	3 hours
Non-functional requirements	2 hours
Revision after the discussion with the tutor	5 hours
Alloy	4 hours
Revisions	8 hours
Latex document composition	5 hours

Lorenzo Male

TOPIC	HOURS
Analysis of the assignment	5 hours
Purpose & Scope	2 hours
Scenarios and Class diagram	8 hours
Use cases and Sequence diagrams	11 hours
Phenomena, Goals, Functional Requirements and Domain Assumptions	10 hours
Non-functional requirements	2 hours
Revision after the discussion with the tutor	5 hours
Alloy	16 hours
Revisions	8 hours

Gabriele Morelli

TOPIC	HOURS
Analysis of the assignment	5 hours
Purpose & Scope	2 hours
Scenarios and Class diagram	12 hours
Use cases and Sequence diagrams	12 hours
Phenomena, Goals, Functional Requirements and Domain Assumptions	7 hours
External Interfaces Requirements	3.5 hours
Non-functional requirements	0.5 hours
Revision after the discussion with the tutor	5 hours
Alloy	10 hours
Revisions	8 hours

1.6 Reference Documents

Specification document: "R&DD Assignment AY 2020-2021.pdf"

RASD standard: http://didattica.agentgroup.unimore.it/wiki/images/7/7a/IEEE830_ENG.pdf

2 Overall description

2.1 Product perspective

The product we're going to develop comprises two main parts: the mobile app, which works as an interface to the User, and the System, which is the business side of this application responsible for all the operations and of memorizing all the information. The CLup system is intended to provide three main functionalities.

First, to take a ticket, in order to start queuing. It's crucial the fact that the CLup application provides also to the users, the estimated waiting time for your turn to come, calculated by the system using some parameters including the starting address of the user.

Second, to do a booking, in order to reserve your visit to do grocery. If the user tries to book a slot of time which is already busy, the system gives a suggestion of possible alternatives.

Third, the system allows stores to book a visit for customers who go physically to the store to book their visit. A printable memo can be obtained through the "physical booking" procedure performed by the stores, this functionality requires the store to print them.

The system uses QR codes to uniquely identify users inside the CLup system and to aid the work of the store manager in regulating and managing the influx of people. This

last feature is actually performed by the store employees which controls the memos at the entrance by simply scanning the QR codes.

The System also provides suggestions to the users by sending them a notification about available slots to book a visit.

Finally, the system, through some mechanisms, can enhance the maximum capacity of the store allowing more people in the store.

2.1.1 Scenarios

- **Take a ticket**

Giancarlo is an employee of an Insurance Company in Milan. Since he's at the beginning of his career, he has to work full day to get promoted. Because of this, he has very little free time. The only window of time in which he can go grocery shopping is during the lunch break, which lasts 1 hour. Therefore, he needs to schedule a visit to the supermarket in order to be sure to be able to go grocery shopping. In order to schedule a visit, he opens the application, chooses the option "book a visit", then selects the preferred supermarket, and, between the appointments still available, chooses the appointment he prefers. Finally, he inserts the duration that he thinks his visit would last and the categories of the items he intends to buy. On the chosen day, he reaches the supermarket at the chosen time and a store employee scans his QR code to let him access the supermarket.

- **Take a ticket**

Sergio has the fridge empty, so in order to have lunch he has to go grocery shopping. Therefore he opens the CLup application, selects "Take a ticket", chooses the supermarket nearby his home and starts queuing; the system returns him a memo with the estimated waiting time and the recommended departure time. Then he waits at home and begins moving, following the recommended departure time suggested by the application. He arrives at the supermarket right on time, lets a store employee scan his QR code and then he enters the store and takes what he needs. When he finishes, he lets scan his QR code again at the exit and, finally, he goes home to have his lunch.

- **Do a physical bookingt**

Gildo is an 80 years old person who doesn't have a family that can assist him, yet he's still able to manage himself, that's why he still does grocery shopping by himself. Unfortunately, he's too old to be able to learn how to use a smartphone, that's why he can't take a ticket through the CLup application. So, when he needs to buy some products, he directly goes to the supermarket. Once on the spot, he asks a store employee to book the first available slot for a visit, the employee then asks for the categories of the products he's going to buy and how much time he estimates his visit will last. Once he has chosen the appointment and provided the information, the employee prints him the memo.

- **Increase maximum capacity**

The Lidl store in Rozzano has a maximum capacity of 20 people. Referring to this threshold, 6 slots (30% of the maximum number) are reserved only for booked visits. On Friday 15th December, from 2 pm to 3 pm, the store already has 6 bookings and the queue is full. All people that have booked the visit, have also indicated the categories of the items they intend to buy: 1 person needs to buy only fruit and vegetables, 2 people need to buy personal hygiene products, the last 2 have to buy correspondingly frozen foods and cold cuts. The system, evaluating the situation (considering the equal distribution of the people), allows other 2 people from the queue to enter the store.

- **User does a wrong estimation**

Ruggero has booked a visit for the supermarket near his home, where he is a long-term customer. While booking, he declares that his stay will last 15 minutes and books a single 30-minutes time slot. Once at the store, right on time, he enters it after letting a store employee scan his QR code. While doing the groceries, he decides to also buy the ingredients for a pizza. He has never made one so he checks the ingredients on his phone and starts looking for them on the shelves. After 25 minutes, he lets an employee scan his QR code and leaves the store. The system updates the estimated waiting time for the people who have taken a ticket. The system updates time statistics associated with Ruggero's QR code.

- **User doesn't follow the recommended departure time**

Piero has already taken a ticket. Since he is in a hurry, instead of waiting at home for his turn he leaves home earlier. When he arrives there, he notices that he isn't the only one with that idea and so he immediately finds himself in a crowded place, evanescing the purpose of the CLup application.

- **System gives a suggestion**

Daniele wants to buy a just-baked cake for his birthday, which will be on Friday. To ensure being able to do grocery on Friday, he tries to book an appointment for that day. When he opens the application, inserts the preferred store and chooses Friday as day, discovers that all appointments are already booked. Nevertheless, he tries to click on a specific time slot for that day. The system in response gives him a suggestion which consists of the next available visit, providing the next available slot after the one he chose and also, if there's one, the nearest supermarket of the same chain which still has an appointment available at the time of the specific appointment he had clicked on.

- **System sends a notification**

Sara has already used the application when she needed to do grocery shopping , so the application starts sending her weekly notifications saying that a slot she had booked previously is free also for the following week, at the same time and for the same store.

- **User cancels a ticket**

It's 5 pm and Luca has already taken a ticket for the store near his house. The estimated waiting time is 20 minutes. After 10 minutes, the estimated waiting time is still 18 minutes, probably due to someone being late. Since the estimated waiting time is still too high, he decides to cancel his ticket from the application. The system then updates the estimated waiting time for people who were queuing after Luca.

- **User arrives late at the store**

Carlo is at home and wants to go grocery shopping. So he opens the CLUp application, selects "take a ticket", inserts, through the GPS, his current position and starts queuing. The system returns him a memo in which it is written that he will take 22 minutes to arrive at the store. However Carlo doesn't trust the suggestion made by the application because he thinks that it will take him only 10 minutes to arrive at the store. So he decides to begin moving only 10 minutes before his turn, but when he arrives at the store he has accumulated a delay of 12 minutes and so he loses his turn in the queue because his ticket has expired.

2.2 Phenomena

WORLD PHENOMENA:

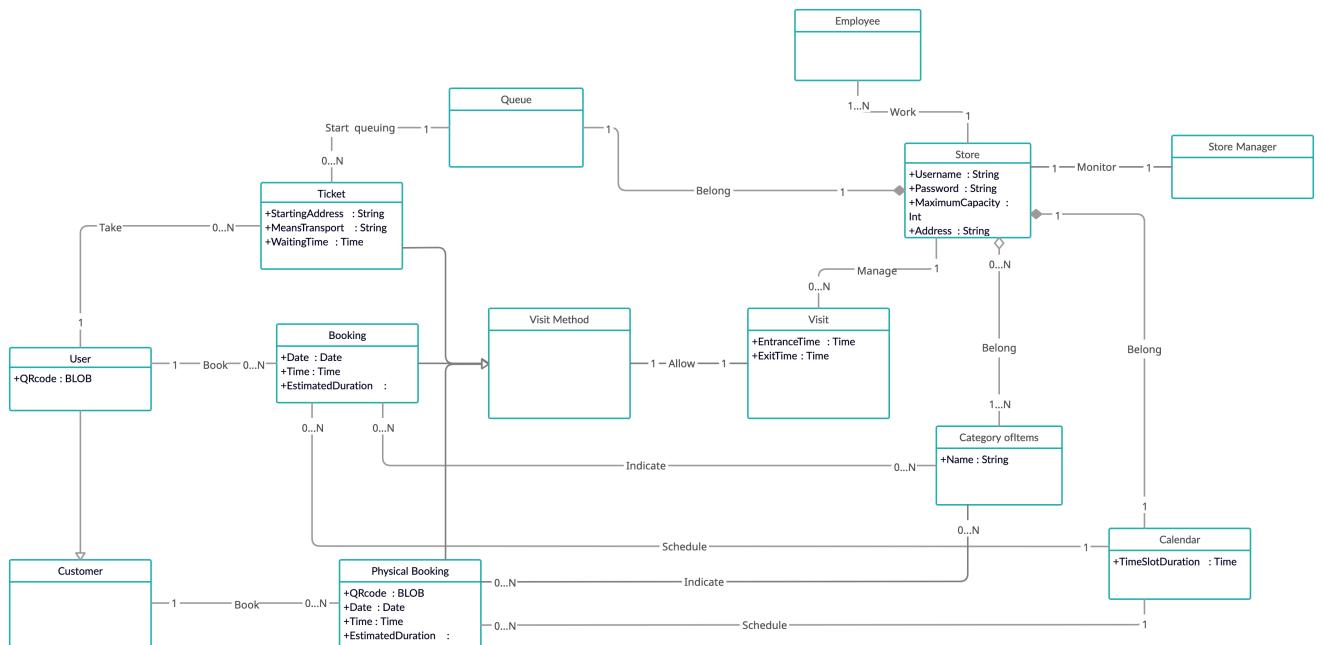
- Limited access to the store results in queuing outside
- People need to go grocery shopping
- Not everyone has the required technology
- People can arrive early or late
- People can skip a fixed appointment
- People may do wrong estimations

SHARED PHENOMENA:

- Take a ticket through the application (Machine)
- The store hands out memos on the spot (World)
- Book a visit (Machine)
- Scan the QR code (World)
- People get a notification (Machine)
- Receive a suggestion (Machine)

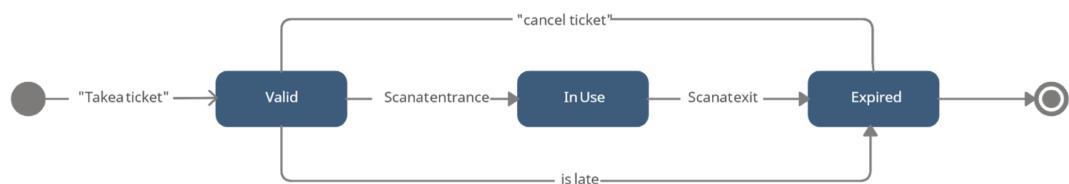
- Show the estimated waiting time (Machine)
- Indicate the starting address (World)
- The store prints a memo (World)
- The system enhance the maximum capacity of the store (Machine)

2.2.1 Class diagram

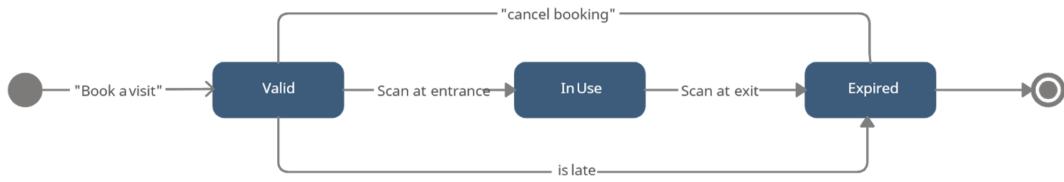


2.2.2 State charts

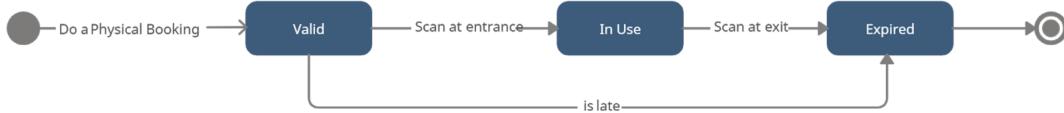
TICKET:



BOOKING:



PHYSICAL BOOKING:



2.3 Product functions

Here are described the most important requirements to achieve the most prominent functions of the CLUp system. In our system users don't have to register or authenticate themselves, every time a user downloads and installs the CLUp app, the system generates a new unique QR code and associates it with such a user, as if it were his/her identification inside the CLUp system.

Take a ticket:

The "take a ticket" procedure requires that the user opens the CLUp application and clicks on the "take a ticket" button displayed in the homepage. Next, the user provides the starting address, which can be provided by allowing the CLUp application to access the GPS or inserting it manually, and the means of transport through which he'll reach the store. Then, the user can choose a specific store by choosing from a list of available stores provided by the system, based on the stated starting address, or by searching manually a specific store. Each store for which it is possible to take a ticket, is accompanied with the estimated waiting time. Now the user can start queuing and a "memo" is generated by the system and will be made available to be seen from a dedicated menu in the CLUp application. This memo contains the User's QR code, the recommended departure time and the estimated waiting time.

Book a visit:

The "book a visit" procedure requires that the user opens the CLUp application and clicks on the "book a visit" button displayed in the homepage. Next, the user can choose from a list of stores provided by the system and also to search manually for a specific store. If the user doesn't allow the CLUp application the access to the GPS, only the manually search function for a store is provided. Then, the CLUp application displays a weekly calendar divided in time slots for the specific store. In particular, this calendar shows all the slots still available and those which are busy. Once the User has chosen

a free slot, the CLUp application displays a new page which contains the chosen date and time and two optional fields. The first field allows to specify his estimation of the duration of his/her visit. If the duration of his/her visit is more than the duration of the time slot, the system takes into account this and asks whether the user wants to book also the next time slot, if available. Otherwise, if the following slot isn't available, the CLUp application won't allow the user to provide such estimated duration. The second field, allows to specify the categories of the items he's going to buy. Stating in advance the categories of the products he/she intends to buy would be useful in order to allow the system to increase the maximum capacity of the store up to 10% if users are going to buy different things, hence occupy different spaces in the store. Now the user can book his/her visit and a "memo" is generated by the system and will be made available to be seen from a dedicated menu in the CLUp application. This memo contains User's QR code, day, time and eventually the categories and the declared estimated duration of his/her visit.

Suggestion of alternative slots or stores:

If the user chooses a busy slot, besides the fact that he/she won't be able to book it, the CLUp application displays a suggestion which contains tips to book a visit for a different store of the same chain at the same time, and for the same store at a different time.

Store's Sign Up:

A Store will be present on the CLUp application after the Store Manager sends to the CLUp team the proper documentation about the store (including the store maximum capacity, the list of the categories of the products they sell and the duration of a time slot. Will be the responsibility of the store manager to choose an effective duration). After the approval of the store's documentation by the CLUp team, the Store will be visible on the CLUp application and the credentials will be provided to the Store Manager. How this procedure is implemented is out of the scope of this document.

Time statistics:

The time statistics takes into account: the time that a long-term user declared that his/her visit would have lasted when he/she booked it and the time actually spent during such visit. These information are then elaborated and used by the system to estimate how accurate is the estimated duration such long-term user states when books visits. The system then uses these information to manage the queue in a finer way. How the system calculates these statistics and how this feature is concretely implemented is out of the scope of this document.

Dynamic managing of the queue:

By default, each booked visit lasts as the duration of the time slot. If the user didn't declare an estimated duration for his/her visit, two cases can occur when he/she exits the store. If the duration of his/her visit was less than the duration of the time slot, the system takes into account this and takes actions to manage the queue more efficiently, for example, allowing users to enter the store earlier by lowering their estimated waiting

time. Actually, even when users declare the estimate of the duration, and this is less than the duration of the time slot, the same mechanism to speed-up the queue occurs. For example, let's suppose that the time slot lasts 1 hour and a user has booked his/her visit at 15:00 stating that the duration of his visit will be 40 minutes, the system can decide whether to use those 20 minutes to let the next user from the queue enter the store. Besides, if the user enters the store at 15:00 and exits at 15:20, the system can decide whether to use those 40 minutes to let the next user from the queue enter the store. Whereas, if the duration of his / her visit was more than the duration of the time slot, the system takes into account this and regulates the estimated waiting times of the other users in the queue.

Notifications:

For long-term customers, the system can notify them by suggesting to book a visit based on his/her previous tickets and visits. These suggestions are generated based on the behaviour of the user taking into account parameters like the most visited stores or the most frequent days and times in which the user goes grocery shopping.

Estimated waiting time:

The estimated waiting time is used in the “take a ticket” functionality. It is calculated by the system taking into account how many people are before him/her in the queue and two other parameters provided by the User: the starting address and the means of transport. This works like a timer in which the time decreases until the turn comes. Pay attention to the fact that if someone before him/her in the queue cancels his/her turn, this time can decrease until the minimum time necessary for the user to reach the store. This time doesn't decrease if the people remain inside the store more time than expected or arrive late to the spot. It is used together with the recommended departure time in order to not have people that line-up outside.

Do a physical booking:

The employee signs into the CLup application accessing the functionalities provided by the application for a store. Since it's not possible to manage the estimated waiting time through this physical procedure, the system allows the store, but concretely the employees, to act as a mediator in the “book a visit” procedure. The “take a ticket” procedure is not available going physically to the spot. The employee can ask whether the customer wants to book the first available slot or a specific free slot for a different moment, then asks for the categories of the products he's going to buy and how much time he estimates his visit will last. Besides, the system generates a temporary QRcode for the customer. The system takes into account that these QR codes are only temporary and it won't be possible to do any time statistic using them. Provided these informations, a printable version of the memo as the memo obtained through the “book a visit” procedure, is generated.

2.4 User characteristics

The actors of our application are the following:

- **User:** it is the client that is using the CLup application. He/she doesn't have to insert any personal information after installing the application; in fact, after the CLup application is downloaded into a mobile phone, the system will automatically assign to it a unique QR code that will be used to identify the user. Through the mobile application he/she can perform 3 main functionalities:
 - take a ticket
 - book a visit
 - look at his/her memos
- **Customer:** it is the physical client of the store that hasn't the possibility to access the technology that is needed to take a ticket or book a visit. Since we want the system to be usable by all demographics, this client can go to the store and physically book a visit.
- **Employee:** he/she is a person who works for a store. He/she can act as a mediator physically booking visits for customers that don't have access to the device that is needed to take a ticket or book a visit. In addition, he/she scans the users' QR code at the entrance and at the exit of the store. In addition, he/she scans the users' QR code at the entrance and at the exit of the store.
- **Store manager:** he/she is in charge of monitoring his/her store and is responsible for sending the documents to the CLup team to subscribe his/her store to the system.

2.5 Assumptions, dependencies and constraints

DOMAIN ASSUMPTIONS:

- Every employee has a device
- Every QR code is scanned at the entrance and at the exit of the store
- Every store has a printer
- People that have booked a visit or taken a ticket through the application have to bring the device with themselves
- People stick to the categories they have specified
- Whoever doesn't use the GPS states the truth about his/her starting address
- Users follow recommended departure time given by the CLup application

-
- The store manager specifies the right maximum capacity of the store
 - Users wait at home until their turn comes

3 Specific requirements

3.1 External interface requirements

3.1.1 User interfaces



Figure 1: Homepage of the mobile phone with the application downloaded

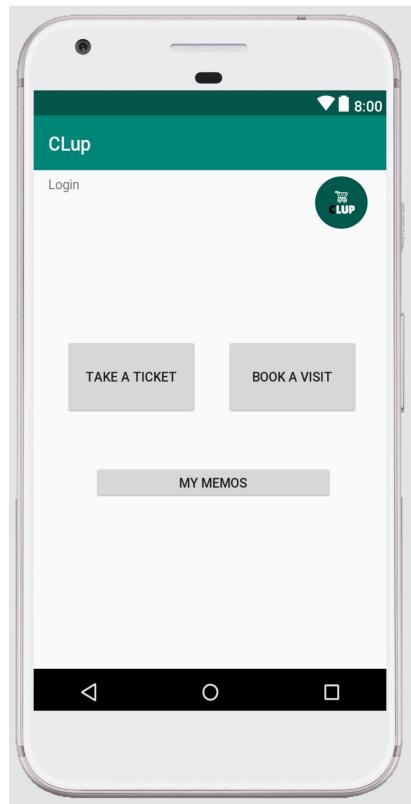


Figure 2: Homepage of the application

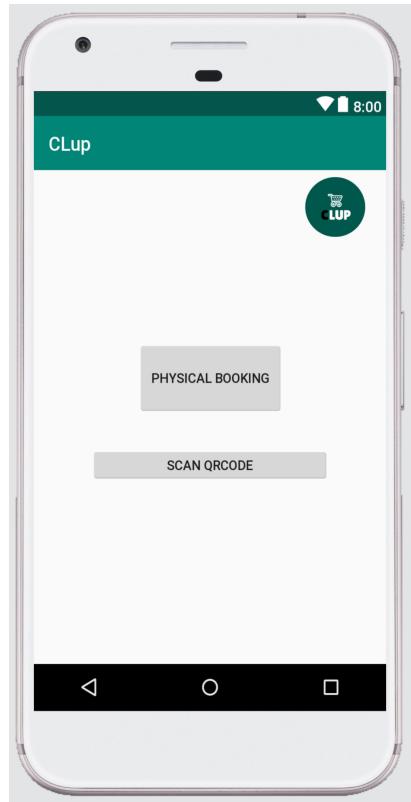


Figure 3: Homepage after the Store's login

3.1.2 Hardware interfaces

Since our project is a piece of software, the hardware interfaces will be those that allow our software to interact with the real world. The most important is the device, that is the physical component used by the two main actors: the user and the employee. The user can use all the functionalities provided by the CLUp application through it and also, when he/she is at the store, can let the QR code to be scanned from it. The employee, using a device can act as a mediator performing the physical procedure of booking a visit, this implies that the CLUp application provides a printable version of the memo that has to be printed by the store, and can also scan the QR codes by using the functionality provided to him which allow to use the device as a scanner.

3.1.3 Software interfaces

The memo is actually an interface provided to the user, since there are three procedures through which it is possible to take it. These three procedures differ also on what the memo comprises.

The calendar is an interface provided to the user to make easier the choice of a slot to book. It aggregates all the information about the busy and available slots for a specific store through an user-friendly interface and the information needed to give suggestions when the user chooses a busy slot.

The procedures, that are “book a visit”, “physical booking”, “take a ticket” and scan a QR code, communicate with the same data storage, since it’s needed to cross all the information in order to accomplish features like: checking if the ticket (or booking) is valid, or to book a visit in a slot which is still available.

3.1.4 Communication interfaces

The device which is executing the CLUp application communicates with the remaining parts of the system using the standard protocols used in the Internet.

3.2 Functional requirements

3.2.1 Goals

1. Allow store manager to regulate the influx of people in the building
2. Save people from having to line up outside the store
3. Allow store manager to monitor entrances
4. Allow customers to schedule their visits to the grocery shop
5. Handle the maximum capacity of a store efficiently

3.3 Functional requirements

1. The system provides, for every store, a calendar to book visits, also indicating whether a slot is available or not
2. User can book a visit in a free slot for a specific store
3. The System allows the User to specify the estimated time of the booked visit
4. User who have indicated their starting address are allowed to take a ticket
5. The System allows User to specify categories of items he/she intends to buy during his visit
6. The system creates a new unique QR code for each downloaded mobile app
7. The system saves the ticket taken by the user and associates it to the user's QR code
8. The system saves the user's booking and associates it to the user's QR code
9. User can take a ticket for a specific store
10. The system provides an up-to-date estimated waiting time for the "take a ticket" functionality
11. The CLup application manages and displays only the stores signed up in the CLup system
12. The system provides a sign-in function for the stores
13. The system should reserve 70% of the store's maximum capacity in a time slot for people that want to take a ticket;
14. The system should reserve the 30% of the store's maximum capacity in a time slot for people that want to book a visit
15. The system should allow the stores to behave as if it were a mediator in the "book a visit" procedure
16. The system has to create temporary QR codes for physical tickets taken from the store's profile
17. The system provides the printable version of the memo associated to the physical ticket
18. The system should allow the employee to scan the QR codes directly from his interface

-
19. The system makes statistics on the time actually spent in the store by every long-term user, using their QR code
 20. The system can decide to enhance up to 10% the maximum capacity of the store in the same time slot when every customer inserts item's category he/she intends to buy
 21. The user can't take a ticket while the store is closed, that is outside of its working time
 22. The user can't take more than one ticket for a specific store until the previous one is used or expired

3.4 Goal mapping

- **G1: Allow store manager to regulate the influx of people in the building**
 - **R2.** User can do a booking for a visit in a free slot for a specific store
 - **R9.** User can take a ticket for a visit to a specific store
 - **R15.** The system should allow the stores to behave as if it were a mediator in the "book a visit" procedure
 - **R12.** The system provides a sign-in function for the stores
 - **R3.** The System allows the user to specify the estimated duration of the booked visit
 - **R10.** The system provides an up-to-date estimated waiting time for the "take a ticket" functionality
 - **R11.** The CLUp application manages and displays only the stores signed up in the CLUp system
 - **DA7.** Users follow recommended departure time given by the CLUp application
 - **DA8.** The store manager specifies the right maximum capacity of the store
- **G2: Save people from having to line up outside the store**
 - **R9.** User can take a ticket for a visit to a specific store
 - **R4.** User who have indicated their starting address are allowed to take a ticket
 - **DA6.** Whoever doesn't use the GPS states the truth about his/her starting address
 - **DA7.** Users follow recommended departure time given by the CLUp application

-
- **R6.** The system creates a new unique QR code for each downloaded app
 - **R7.** The system saves the ticket taken by the user and associates it to the user's QR code
 - **DA1.** Every employee has a device
 - **R10.** The system provides an up-to-date estimated waiting time for the “take a ticket” functionality
 - **DA4.** People that have booked a visit or taken a ticket through the application have to bring the device with themselves
 - **DA2.** Every QR code is scanned at the entrance and at the exit of the store
 - **R11.** The CLup application manages and displays only the stores signed up in the CLup system
 - **R13.** The system should reserve 70% of the store's maximum capacity in a time slot for people that want to take a ticket
 - **R18.** The system should allow the employee to scan the QR codes directly from his/her interface
 - **R21.** The user can't take a ticket while the store is closed, that is outside of its working time
 - **R22.** The user can't take more than one ticket for a specific store until the previous one is used or expired
 - **DA9.** Users wait at home until their turn comes

- **G3: Allow store manager to monitor entrances**

- **R6.** The system creates a new unique QR code for each downloaded app
- **R7.** The system saves the ticket taken by the user and associates it to the user's QR code
- **R8.** The system saves the user's booking and associates it to the user's QR code
- **R11.** The CLup application manages and displays only the stores signed up in the CLup system
- **R12.** The system provides a sign-in function for the stores
- **DA1.** Every employee has a device
- **DA2.** Every QR code is scanned at the entrance and at the exit of the store
- **DA4.** People that have booked a visit or taken a ticket through the application have to bring the device with themselves

- **G4: Allow customers to schedule their visits to the grocery shop by booking them**

- **R1.** The system provides, for every store, a calendar to book visits, also indicating whether a slot is available or not
- **R2.** User can do a booking for a visit in a free slot for a specific store
- **R3.** The System allows the user to specify the estimated duration of the booked visit
- **R5.** The System allows the user to specify categories of items he/she intends to buy during his visit
- **R6.** The system creates a new unique QR code for each downloaded app
- **R8.** The system saves the user's booking and associates it to the user's QR code
- **R11.** The CLup application manages and displays only the stores signed up in the CLup system
- **R14.** The system should reserve the 30% of the store's maximum capacity in a time slot for people that want to do a booking
- **R15.** The system should allow the stores to behave as if it were a mediator in the "book a visit" procedure
- **R16.** The system has to create temporary QR codes for physical tickets taken from the store's profile
- **R17.** The system provides the printable version of the memo associated to the physical booking
- **R18.** The system should allow the employee to scan the QR codes directly from his/her interface
- **DA1.** Every employee has a device
- **DA2.** Every QR code is scanned at the entrance and at the exit of the store
- **DA3.** Every store has a printer
- **DA4.** People that have booked a visit or taken a ticket through the application have to bring the device with themselves
- **DA5.** People stick to the categories they have specified

- **G5: Handle the maximum capacity of a store efficiently**

- **R11.** The CLup application manages and displays only the stores signed up in the CLup system
- **R13.** The system should reserve 70% of the store's maximum capacity in a time slot for people that want to take a ticket

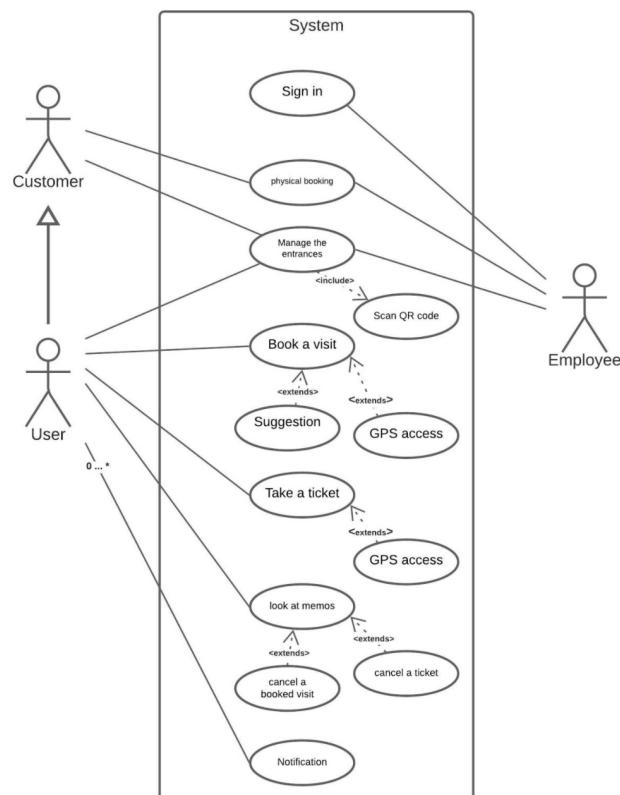
-
- **R14.** The system should reserve the 30% of the store's maximum capacity in a time slot for people that want to do a booking
 - **R19.** The system makes statistics on the time actually spent in the store by every long-term user
 - **R20.** The system can decide to enhance up to 10% the maximum capacity of the store in the same time slot when every customer inserts item's category he/she intends to buy
 - **DA5.** People stick to the categories they have specified

	G1	G2	G3	G4	G5
R1				X	
R2	X			X	
R3	X			X	
R4		X			
R5				X	
R6		X	X	X	
R7		X	X		
R8			X	X	
R9	X	X			
R10	X	X			
R11	X	X	X	X	X
R12	X		X		
R13		X			X
R14				X	X
R15	X			X	
R16				X	
R17				X	
R18		X		X	
R19					X
R20					X
R21		X			
R22		X			

	G1	G2	G3	G4	G5	
DA1		X	X	X		
DA2		X	X	X		
DA3					X	
DA4		X	X	X		
DA5					X	X
DA6		X				
DA7	X	X				
DA8	X					
DA9		X				

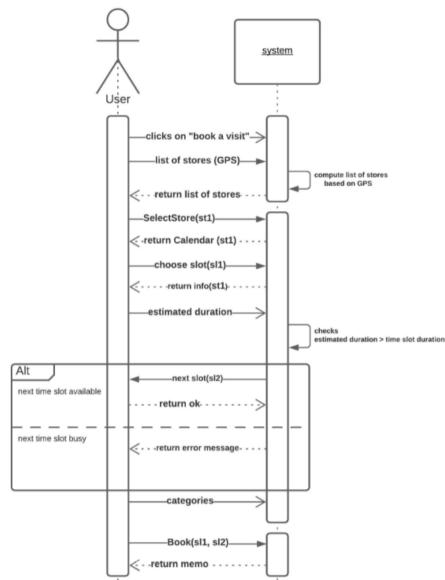
3.5 Use cases and sequence diagrams

3.5.1 Use cases diagram



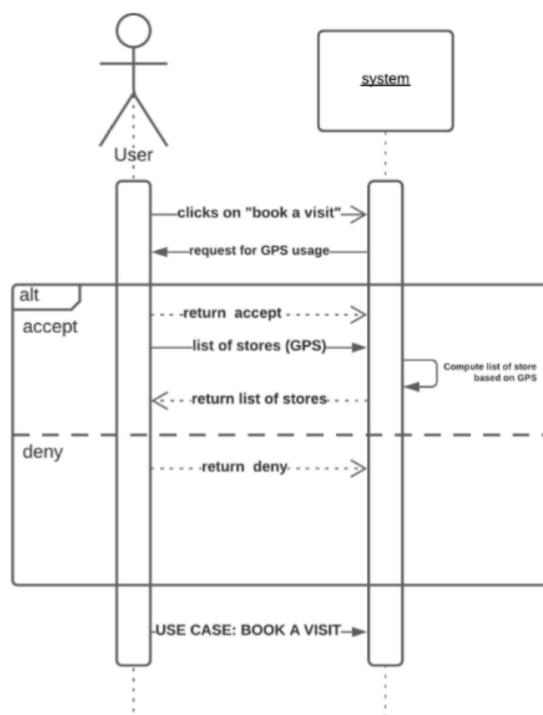
Book a visit

Name	Book a visit
Actor	User
Entry condition	The user has already installed the application and wants to schedule his/her visit to a store
Event flow	<ol style="list-style-type: none"> 1. In the homepage, the user clicks on the "book a visit" button; 2. The CLup application displays a list of stores based on the GPS location and also the option to search manually for a specific store; 3. The User chooses the desired store. 4. The CLup application displays a weekly calendar divided in time slots for the specific store. In particular, this calendar shows all the slots still available and those which are busy. 5. The User chooses a free slot from the calendar; 6. The CLup application displays a new page in which the chosen date and time are stated and also two optional fields in which the User specifies his estimation of the duration of his/her visit and the categories of the items he's going to buy; 7. The User's estimation of the duration of his/her visit is greater than the duration of the time slot 8. The system asks whether the user wants to book also the next slot; 9. The user clicks on a button in order to book also the next slot; 10. The system saves these information assigning them to the user's QR code
Exit condition	The system provides a memo to the User
Exception	<p>If the user chooses a "busy slot" he/she won't be able to book it.</p> <p>If the following time slot is busy, the system won't allow the user to book it.</p>



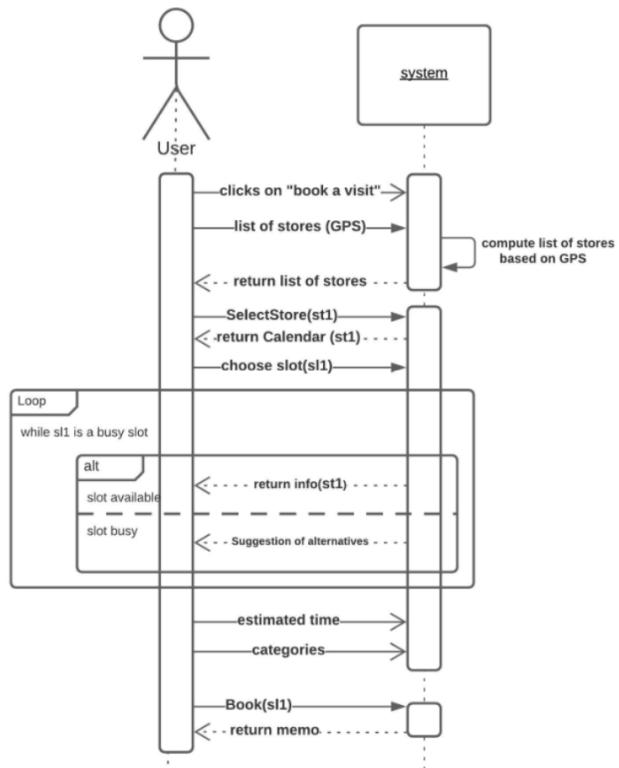
Book a visit - first time

Name	Book a visit - First time
Actor	User
Entry condition	The user has already installed the application and wants to schedule his/her visit to a store
Event flow	<ol style="list-style-type: none"> 1. In the homepage, the user clicks on the "book a visit" button; 2. The CLUp application asks to the user to accept the usage of the GPS; 3. continue from the point 3 of the use case Book a visit
Exit condition	The system provides a memo to the User
Exception	<p>If the user chooses a "busy slot" he/she won't be able to book it.</p> <p>If the user doesn't allow the CLUp application the access to the GPS, then only the manually search function for a store is provided</p>



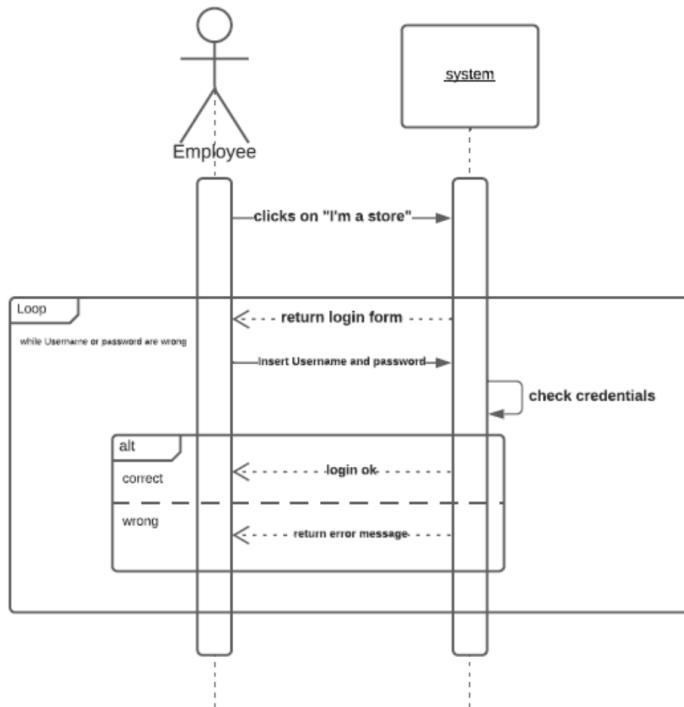
Suggest of alternative slots or stores

Name	Suggest alternative slots or stores
Actor	User
Entry condition	The user, during the “book a visit” procedure, clicks on a busy slot in the calendar
Event flow	<p>1. The CLUp application displays to the User a suggestion which contains tips to book a visit for:</p> <ul style="list-style-type: none"> ○ A different store of the same chain at the same time ○ The same store but on a different date and time
Exit condition	The system displays the suggestion
Exception	None



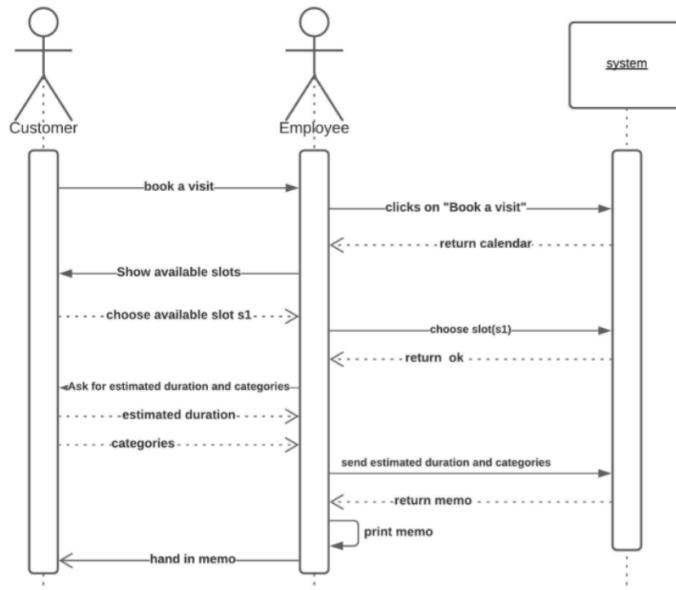
Sign-in the Clup application

Name	Sign-in the Clup application
Actor	Employee
Entry condition	The Employee has the credentials and a device with the Clup application installed
Event flow	<ol style="list-style-type: none"> 1. In the homepage, the customer clicks on the "I'm a store" button; 2. The CLup application displays two fields in which the Employee specify the Username and the password 3. The Employee clicks on the "submit" button
Exit condition	The employee is logged in
Exception	The credentials provided by the Employee are wrong, so the system responds with an error message



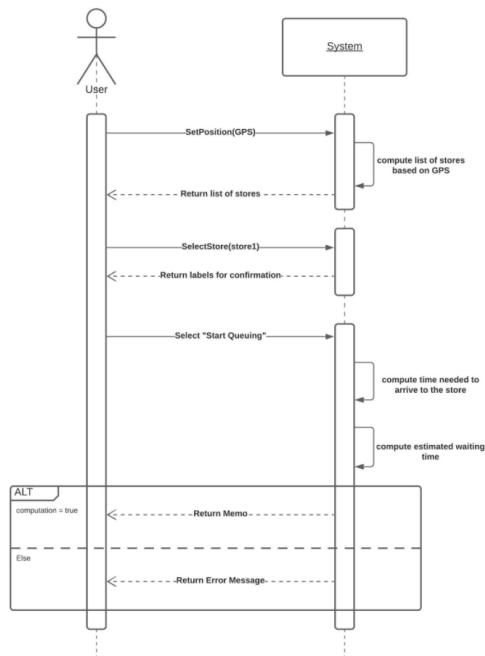
Book physically a visit at the store

Name	Book physically a visit at the store
Actor	Customer, Employee
Entry condition	The customer is physically at the store and the Employee is already signed into the CLUp application
Event flow	<ol style="list-style-type: none"> 1. The customer asks to an employee to book a visit; 2. The employee asks whether the customer wants to book the first available slot or a specific free slot for a different moment; 3. The customer chooses a free slot to book; 4. The employee asks whether the customer wants to specify the categories of item he intends to buy and the estimated duration of his visit; 5. The customer provides these information; 6. The employee takes on the choices by inserting them in the CLUp application; 7. The system generates a temporary QR code 8. The system provides a printable version of the memo.
Exit condition	The Employee hands in the printed memo to the customer
Exception	None



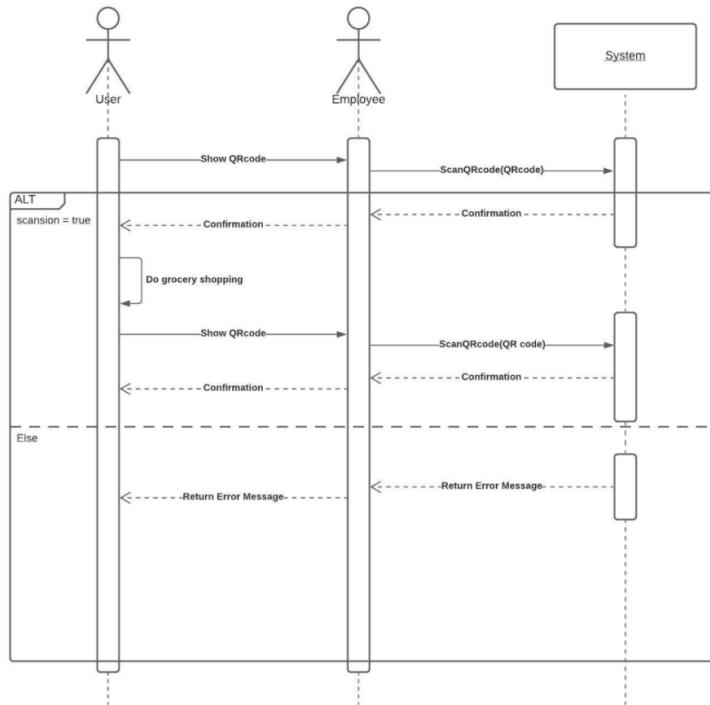
Take a ticket

Name	Take a ticket
Actors	User
Entry Condition	User has the CLUp application installed on her/his smartphone
Event Flow	<ol style="list-style-type: none"> 1. In the homepage, the User clicks on the "Take a ticket" button; 2. The User indicates her/his starting address by GPS or inserting it manually and her/his means of transport 3. The CLUp application displays a list of stores based on the address provided, with the respective estimated waiting time, and a search bar for inserting manually the name of the desired store 4. The User chooses the desired store by clicking on it 5. The User visualises the selected store description and a button to start queuing 6. The user clicks the button to start queuing 7. The system saves these information assigning them to the user's QR code
Exit Condition	The system provides a memo to the User
Exception	The Store is closed and so the User can't take the ticket. In this case, if the User searches the store and clicks on it, a pop-up is displayed with a "The store is closed, you can't start queuing" message.



Manage the entrances

Name	Manage the entrances
Actors	User, Employee
Entry Condition	The Employee is signed in and the turn of the User has come
Event Flow	<p>1) The Employee clicks on the button to access the function to scan QR codes</p> <p>2) The Employee scans the QR code of the User</p> <p>3) User enters the store</p> <p>4) After having done her/his groceries, the User lets the Employee scan again her/his QR code at the exit</p>
Exit Condition	User exits the store
Exception	The memo is invalid for that schedule or store. In this case, the Employee scans the QR code of the User discovering that the memo is invalid, so the User can't enter the store.



Arrive late to the store (“Take a ticket”)

Name	Arrive late to the store (“Take a ticket”)
Actors	User
Entry Condition	The User has taken a ticket and has the memo
Event Flow	<ol style="list-style-type: none">1) The estimated waiting time in the memo goes to zero2) The User doesn't scan his QR code at the store3) The system waits the scan of the user's QR code for 10 minutes4) The user doesn't scan his QR code at the store
Exit Condition	The system expires the ticket and eliminates the memo of the user and reduces the estimated waiting time of the following users
Exception	User scans his QR code at the store within the 10 minutes in which the system is waiting so he can enter in the store

Arrive late to the store (“Book a visit”)

Name	Arrive late to the store (“Book a visit”)
Actors	User
Entry Condition	The User has booked a visit and has the memo
Event Flow	<ol style="list-style-type: none">1) The time of the appointment comes2) The user doesn't scan his QR code at the store3) The system waits the scan of the user's QR code for 10 minutes4) The user doesn't scan his QR code at the store
Exit Condition	The system cancels user's booking and eliminates the memo
Exception	User scans his QR code at the store within the 10 minutes in which the system is waiting so he can enter in the store

Look for a memo

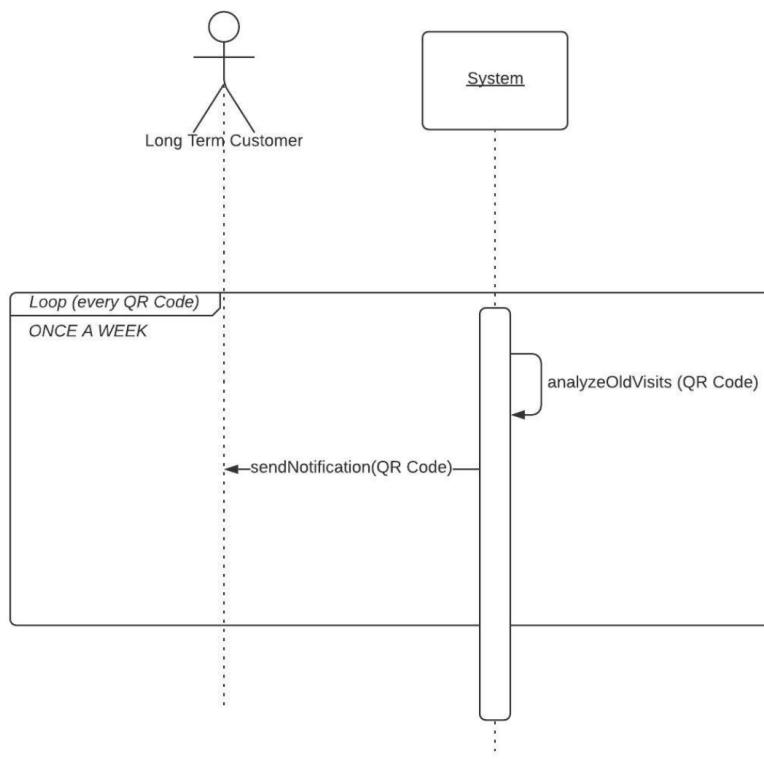
Name	Look for a memo
Actors	User
Entry Condition	The User has already installed the application and wants to look for a memo
Event Flow	<ol style="list-style-type: none">1. The User opens the CLUp application2. The User goes to the menu dedicated to the memos3. The CLUp application displays the list of all the user's memos;4. The User clicks on a specific memo;5. The CLUp application displays all the information about such memo;
Exit Condition	Information about a memo are provided
Exception	None
Special Requirements	Only the memos for the days to come can be seen in the application, memos about past days disappear from the Clup application

Increase maximum capacity of a store

Name	Increase maximum capacity of a store
Actors	User
Entry Condition	Users have booked all possible visits for a specific time slot of a store.
Event Flow	<ol style="list-style-type: none">1. Every User has specified all the categories of items they intend to buy2. System analyses categories that have been specified by the users
Exit Condition	System decides whether to increase the maximum capacity of the store up to 10%
Exception	None

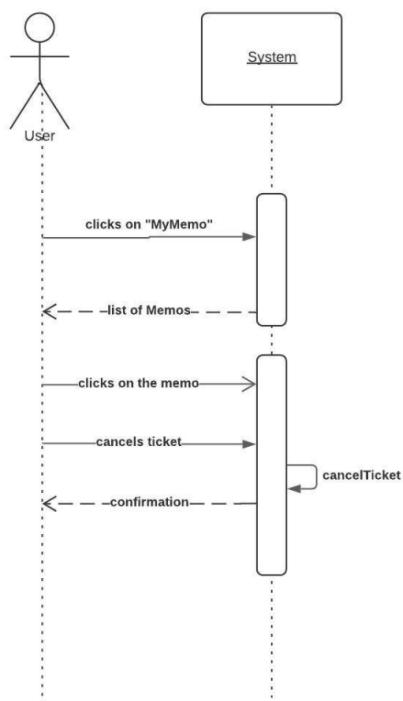
Receive a notification for a free slot to book

Name	Receive a notification for a free slot to book
Actors	User
Entry Condition	User has already had a booked visit or taken a ticket for a certain store
Event Flow	1. System sends a notification to user suggesting to book a visit based on his/her previous tickets and visits
Exit Condition	User receives the notification
Exception	None



Cancel a ticket

Name	Cancel a ticket
Actors	User
Entry Condition	User has taken a ticket for a store
Event Flow	<ol style="list-style-type: none"> 1. User opens the CLup application 2. User goes to the menu dedicated to the memos 3. User clicks on a specific memo 4. User visualizes the memo description 5. User cancels the ticket 6. User confirms his/her choice 7. System eliminates user's position from the queue 8. System updates the estimated waiting time for other users in line for the same store
Exit Condition	User's ticket is cancelled
Exception	None



3.6 Performance requirement - Non-functional requirements

1. We have assumed that an user or a customer can arrive late on his/her turn at the store. When this happens, her/his position in the queue of the store is saved for 10 minutes, while the rest of positions in the queue continue to flow.
2. The visits have to be booked at least 30 minutes in advance. In case the store's slots available for visits are not fully booked, then the system will use these slots for the "take a ticket" functionality, in order to speed up the queue. i.e.: let's take for example the slot at 10:00 a.m. If it comes 9:30 a.m. and for a store there still are some slots not booked, then these will not be anymore reserved for the "book a visit" but the system will use them for the "take a ticket" functionality.

3.7 Design constraint

HARDWARE LIMITATIONS

In order to provide the physical tickets to customers, the system has to provide a printable version for the memos obtained through the "get a physical ticket" functionality that has to be printed with the printer of the store.

ANY OTHER CONSTRAINT

A user can't take a ticket or book a visit for a store in which he/she already has a ticket or a booked visit which is still valid.

3.8 Software system attributes

3.8.1 Reliability

The system must guarantee a 24/7 reliable service. Errors during the process should be managed and users should be alerted if anything goes wrong in the process of taking a ticket/booking a visit.

3.8.2 Availability

The system must guarantee a continuous service, since avoiding physical lining up is crucial also for one single day. Very small deviations are acceptable since the most important functionality (taking a ticket) isn't needed out of working hours. A 2-nine availability is needed, so that the average time between the occurrence of a fault and service recovery should be contained around at 3.65 days per year.

3.8.3 Security

Users are anonymous. Memos should be stored and users can cancel tickets or visits only if their QR code corresponds to the one saved in the memo. Store's log-in is protected with a password, and in case of password loss this should be reset only with previous validation of the Store manager's identity.

3.8.4 Maintainability

The development of the CLup system must be done so that in the future it will be easy to fix and modify it, according to the circumstances, i.e capable of facilitating addition of new features and options.

3.8.5 Portability

CLup application should be developed for many devices, for mobile phone and tablet operating systems as iOS, Android, Windows Mobile.

4 Alloy

This section is dedicated to a model of the project, developed with the Alloy tool. For sake of a better understanding, the model doesn't represent the entire system, some details are left out and some are simplified (i.e. the time slot of a calendar gives the opportunity of only one booking instead of the 30% of the capacity of the Store).

The model focuses mostly on the main entities of Store, Ticket, Booking and Physical Booking, on the behavior of them in relation with their states and on their constraints. Entities like Queue, Calendar, Time Slot are modelled accordingly to their use in relation to the main entities. Some secondary details are added.

The features that our model is going to show are the function of booking a time slot for a store, the function of taking a ticket, the event of entering a store after having taken a ticket and the event of exiting from a store, showing the entities involved and the relations created. Lastly, a creation of a new visit in a full store is tried, proving that it's not possible.

Snapshots of instances are attached at the end of the code.

```

open util/Time

sig Customer{
    physBook: set PhysicalBooking
}

sig QRcode{}

sig User extends Customer{
    userQRcode: one QRcode,
    take: set Ticket,
    book: set Booking
}

abstract sig VisitMethod{
    vmState: State lone -> Time,
    haveVisit: lone Visit
}

abstract sig State{}

one sig VALID extends State{}
one sig INUSE extends State{}
one sig EXPIRED extends State{}

sig Ticket extends VisitMethod{
    queue: Queue one -> one Store
}#queue<=1

sig Booking extends VisitMethod{
    indicateBook: set CategoryOfItems,
    scheduleBook: TimeSlot one -> lone Calendar
}

sig PhysicalBooking extends VisitMethod{
    randomQRcode: one QRcode,
    indicatePBook: set CategoryOfItems,
    schedulePBook: TimeSlot one -> lone Calendar
}

sig Queue {
    belongQ: one Store
}

sig Calendar {
    timeSlots: some TimeSlot,
    belongC: one Store,
}

abstract sig TimeSlotState{}
one sig AVAILABLE extends TimeSlotState{}
one sig BUSY extends TimeSlotState{}

sig TimeSlot{
    TSstate: TimeSlotState lone -> Time,
}

```

```

sig Store {
    sell: set CategoryOfItems,
    haveC: one Calendar,
    haveQ: one Queue,
    works : some Employee,
    monitors : one StoreManager,
    maxCapacity:Int,
    haveV:some Visit
}
#maxCapacity>0
#haveV<=maxCapacity

sig Visit {
    entranceTime: one Time,
    exitTime: lone Time,
    storeVisit: one Store
}

sig CategoryOfItems {}

sig Employee {}

sig StoreManager{}

```

////////// FACTS //

////////// TIMESLOTS //

//Each time slot is related to only one calendar

```

fact everyTimeSlotToOneCalendar{
    all ts:TimeSlot | one c:Calendar | ts in c.timeSlots
}

fact timeSlotStateChart{
    //ogni ticket creato come valid
    all ts:TimeSlot | one t0:Time | ts.TSstate.t0=AVAILABLE
    all ts:TimeSlot, t1:Time |
        (ts.TSstate.t1=BUSY implies
            all t2:Time | gte[t2,t1] implies ts.TSstate.t2=BUSY)
}
```

```

|||||||||||||||||QRCODES||||||||||||||||||

//every User's QR code is unique and belongs to only one user
fact noEqualUserQRcode{
    all disj u, u' : User | u.userQRcode != u'.userQRcode
}

//every Physical booking's QR code is unique and belongs to only one Physical booking
fact noEqualPBookingQRcode{
    all disj pb, pb' : PhysicalBooking | pb.randomQRcode != pb'.randomQRcode
}

//User's QR codes and Physical bookings QR codes are different
fact noUserQRforPhysicalBook{
    all u: User | all pb: PhysicalBooking |      no (u.userQRcode & pb.randomQRcode)
}

// every QR code must be related to an User or a Physical Booking
fact noQRcodeWOUserOrPB{
all qr:QRcode | let u=User, pb=PhysicalBooking | ((qr in u.userQRcode) or (qr in pb.randomQRcode))
}

|||||||||||||||||STORE||||||||||||||||||

fact {
haveC=~belongC and haveQ=~belongQ
and
    all v:Visit | one s:Store | s in v.storeVisit implies v in s.haveV
and
    all s:Store | all v:Visit | v in s.haveV implies s in v.storeVisit
}

//every Category is sold at least by one store
fact noCategoryWOStore{
    all coi:CategoryOfItems | some s:Store | coi in s.sell
}

fact storeManagerStore{
    // no Store Manager that works for different stores
    all sm : StoreManager | (no disj s1,s2: Store | sm in s1.monitors and sm in s2.monitors)
and //no store without a StoreManager
    all s : Store | one sm: StoreManager | sm in s.monitors
and //no StoreManager without a store
    all sm: StoreManager | one s:Store | sm in s.monitors
}

fact employeeStore{
    // no Employee that works for different stores
    all e : Employee | (no disj s1,s2: Store | e in s1.works and e in s2.works)
and //no store without an employee
    all s : Store | some e: Employee | e in s.works
and //no Employee without a store
    all e: Employee | one s:Store | e in s.works
}

```

|||||||||VISIT METHOD|||||||

```
//If a Visit Method has as state; "INUSE", then it is linked to one and only one Visit  
fact allINUSEVMhasVisit{
```

```
all vVisit| one vm:VisitMethod | v in vm.haveVisit  
}
```

```

fact visitMethodStateChart{
  //ogni visitMethod creato come valid
  all vm:VisitMethod | one t0:Time | vm.vmState.t0=VALID
  and
  all vm:VisitMethod, t1:Time |
    (vm.vmState.t1=EXPIRED implies
     all t2:Time | gte[t2,t1] implies vm.vmState.t2=EXPIRED)
  and
    (vm.vmState.t1=INUSE implies
     all t2:Time | gte[t2,t1] implies vm.vmState.t2!=VALID)
}

```

|||||||TICKETS|||||||

```

//each Ticket is taken by only one User
fact ticketUser{
    (all t:Ticket | no disj u1, u2:User | t in u1.take and t in u2.take)
and //No ticket without an User
    (all t: Ticket | one u: User | t in u.take)
}

```

||||||BOOKINGS||||||

```
//every Category of Items that is indicated in a booking, is sold by the store related to that booking
fact indicateOnlyStoreCategories{
    all b:Booking, ts:TimeSlot | all coi:CategoryOfItems | (coi in b.indicateBook) implies (coi in sell[belongC[ts.(b.scheduleBook)]]))
}
```

```
//each Booking is booked by only one User
fact bookingUser{
    (all b: Booking | no disj u1,u2: User | b in u1.book and b in u2.book)
and //No booking without an User
    (all b: Booking | one u: User | b in u.book)
}
```

```
// if Booking has a relation that maps Timeslot to calendar, that timeslot has to be in that calendar
fact allBookingsSlotRelatedCalendar{
all b:Booking, c:Calendar | all ts:TimeSlot | (ts in (b.scheduleBook).c) implies (ts in c.timeSlots)
}
```

||||||PHYSBOOKING||||||

```

//every Category of Items that is indicated in a physical booking is sold by the store related to that physical booking
fact indicateOnlyStoreCategoriesPB{
    all pb:PhysicalBooking, ts:TimeSlot | all coi:CategoryOfItems | (coi in pb.indicatePBook)
    implies (coi in sell[belongC[ts.(pb.schedulePBook)]]))
}

//each physical Booking is booked by only one customer
fact physicalBookingCustomer{
    (all pb: PhysicalBooking | no disj cu1, cu2: Customer | pb in cu1.physBook and pb in cu2.physBook)
and //no physical booking without customer
    (all pb: PhysicalBooking | one cu: Customer | pb in cu.physBook)
}

//user can't take a physical booking
fact userNoPhysicalBooking{
all u:User | #u.physBook=0
}

// if PhysicalBooking has a relation that maps Timeslot to calendar, that timeslot has to be in that calendar
fact allPBookingsSlotRelatedCalendar{
all pb:PhysicalBooking, c:Calendar | all ts:TimeSlot | (ts in (pb.schedulePBook).c) implies (ts in c.timeSlots)
}

```

||||| predicates |||||

```
//////////booking of a slot
pred isTimeSlotAvailable[ts:TimeSlot, t:Time]{  
  
ts.TSstate.t=AVAILABLE  
}
```

```

pred makeABooking[u:User, b:Booking, ts:TimeSlot, c:Calendar, t:Time]{
  //preconditions
  isTimeSlotAvailable[ts, t]
  //postconditions
  u.book=b
  b.scheduleBook.c = ts
  not isTimeSlotAvailable[ts,t.next]
  ts.TSstate.(t.next)=BUSY
}

```

||||||||||||||||||||taking a ticket

```
pred userTakesATicket[u:User, tick:Ticket, q:Queue, s:Store]{
    //preconditions
    u.take=tick
    tick.queue.s = q
    #Booking=0
    #PhysicalBooking=0
}
```

```

///////////user enters a store

pred userReadyToEnter[u:User, tick:Ticket, q:Queue, s:Store, t:Time]{
    u.take=tick
    tick.queue.s=q
    tick.vmState.t=VALID
}

pred userEntersAStoreWATicket[u:User, tick:Ticket, q:Queue, s:Store, v:Visit, t:Time]{
    //preconditions
    userReadyToEnter[u, tick, q, s, t]
    //postconditions
    v.entranceTime=t
    #v.exitTime=0
    not userReadyToEnter[u, tick, q, s, t.next]
    tick.vmState.(t.next)=INUSE
    tick.haveVisit=v
}

////////// user exits from a store

pred userReadyToExit[u:User, tick:Ticket, v:Visit, t1:Time, t:Time]{
    u.take=tick
    tick.vmState.t=INUSE
    tick.haveVisit=v
    gte[t, t1]
    v.entranceTime=t1
}

pred userExitsAStore[u:User, tick:Ticket, v:Visit, t1:Time, t:Time]{
    //
    userReadyToExit[u, tick, v, t1, t]
    //

    v.exitTime=t.next
    not userReadyToExit[u, tick, v, t1, t.next]
    tick.vmState.(t.next)=EXPIRED
}

////////no entrances allowed

pred storeIsFull [s:Store, disj v1, v2:Visit]{
    s.maxCapacity=2 // to change to "3" to prove the visit is allowed in that case
    v1.storeVisit=s
    v2.storeVisit=s
}

pred tryToEnterFullStore[ s:Store, disj v1,v2, v3:Visit]{
    //precondition
    storeIsFull[s, v1, v2]
    //postcondition
    v3.storeVisit=s
}

pred show{
}

```

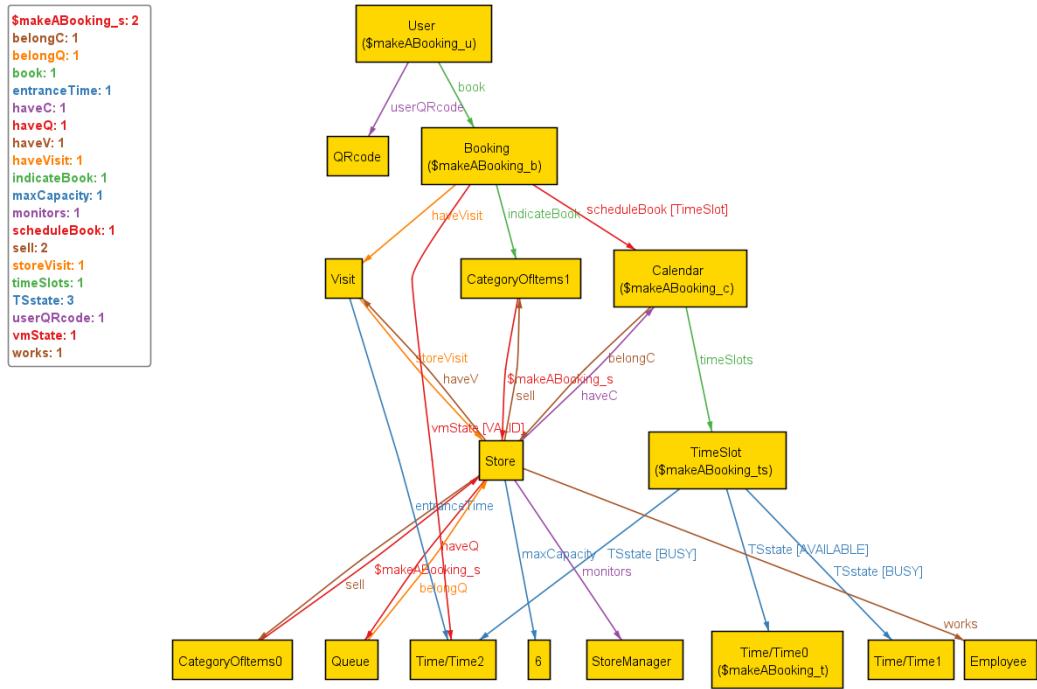


Figure 4: Make a booking

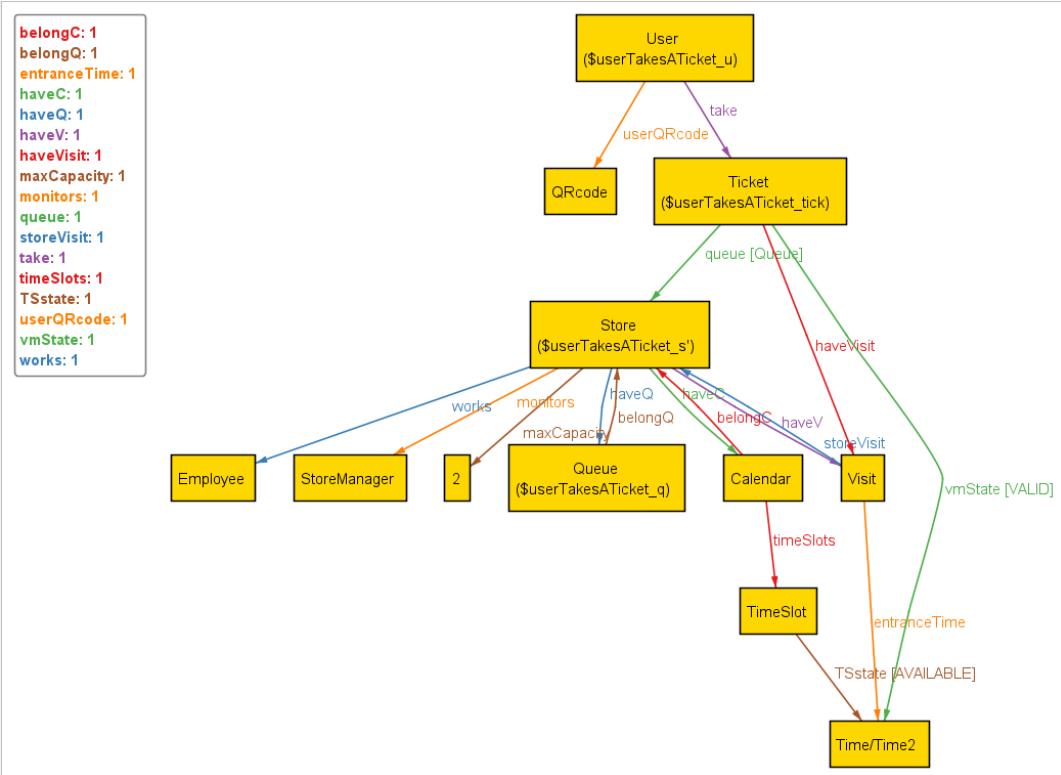


Figure 5: User takes a ticket

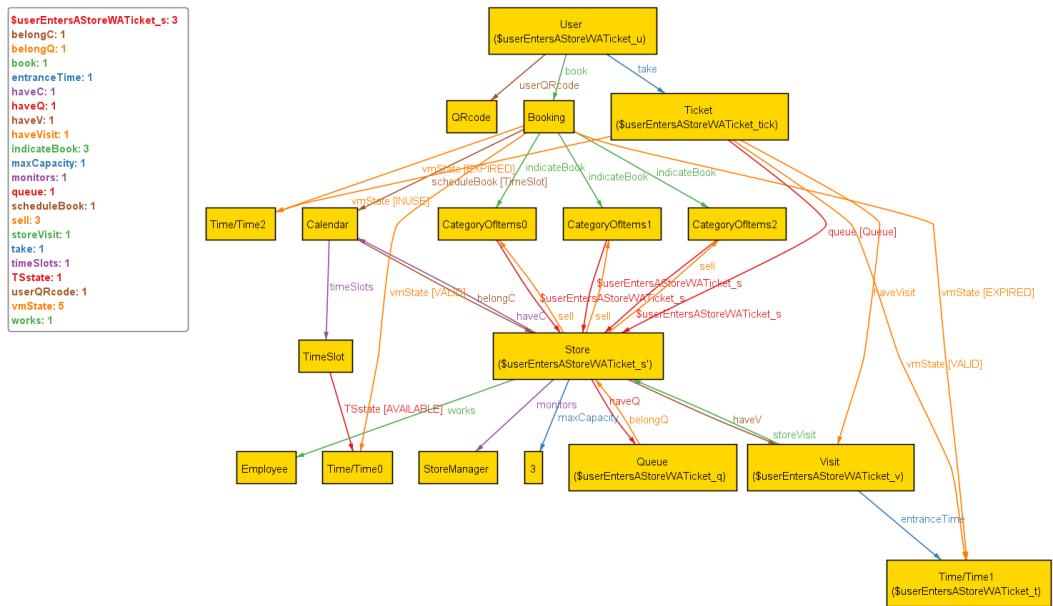


Figure 6: User enters the store with a ticket

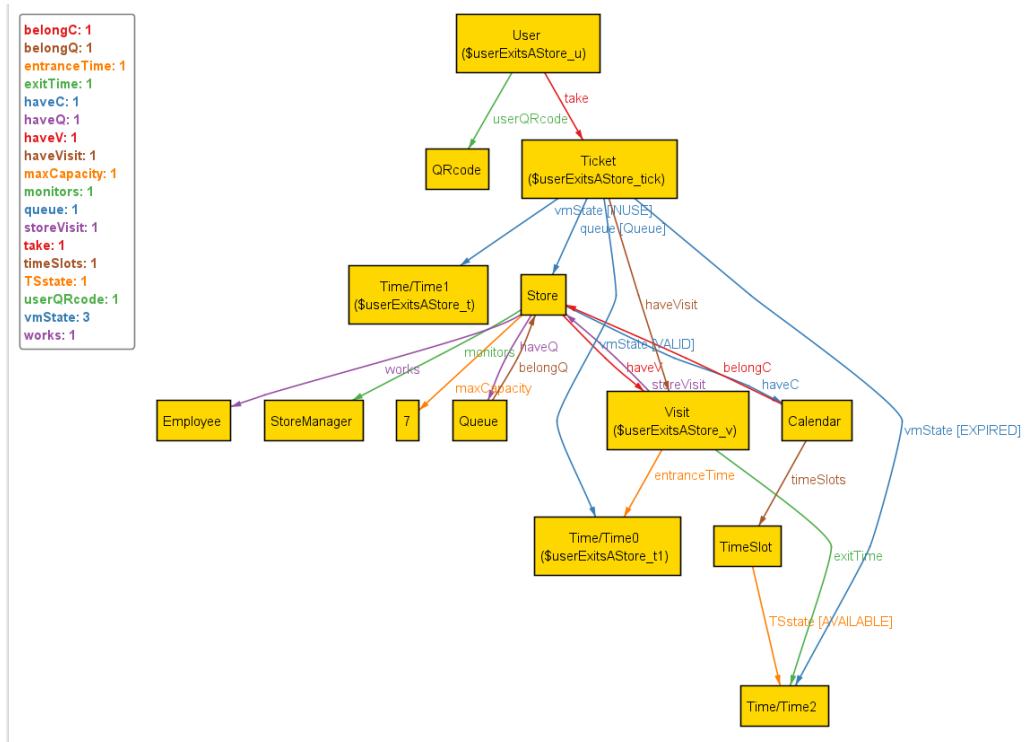


Figure 7: User exits the store

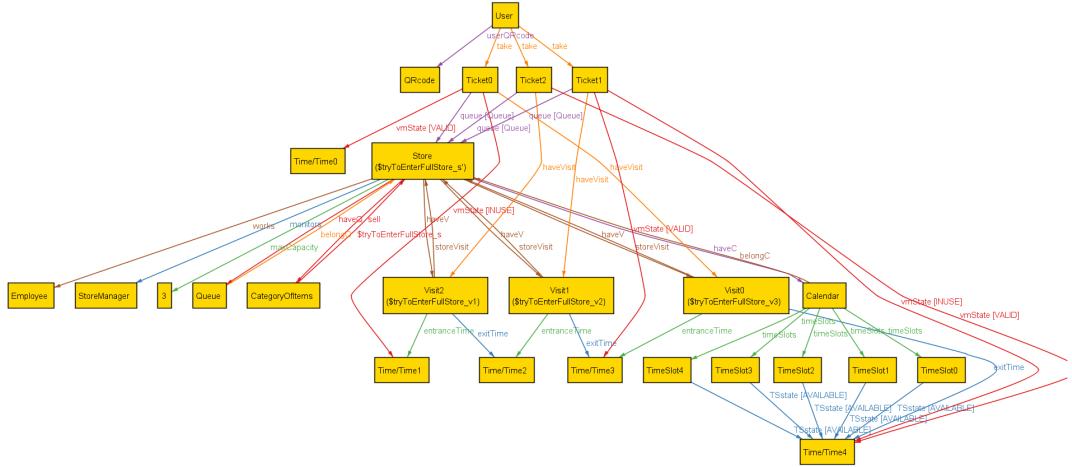


Figure 8: Try to enter full store

The user tries to enter the store and the latter hasn't already reached its maximum capacity:

```
Executing "Run tryToEnterFullStore for 5 but 1 Store"
Solver=sat4j Bitwidth=4 MaxSeq=5 SkolemDepth=1 Symmetry=20
11298 vars. 823 primary vars. 19758 clauses. 51ms.
Instance found. Predicate is consistent. 42ms.
```

The user tries to enter the store but the latter has already reached its maximum capacity:

```
Executing "Run tryToEnterFullStore for 5 but 1 Store"
Solver=sat4j Bitwidth=4 MaxSeq=5 SkolemDepth=1 Symmetry=20
11298 vars. 823 primary vars. 19758 clauses. 42ms.
No instance found. Predicate may be inconsistent. 14ms.
```