POLITECNICO
MILANO 1863

Software Engineering 2 Project

Politecnico di MIlano AA 2020-2021

Computer Science and Engineering

Requirements Analysis and Specifications Document

Professor: Di Nitto Elisabetta

Group Components

Amato Nunzio (Matricola: ***970618***)

Ciriello Giovanni (Matricola: ***963100***)

Colombrino Fulvio (Matricola: ***966097***)

Contents

1. INTRODUCTION

The main objective of this document is the analysis of the requirements and of the specifications of the developed system. It provides an overview of the system through a description of its main goals, of its main functions together with a brief description of the context in which it has been developed. Moreover, the document gathers all the functional and non-functional requirements, elicited through the study of scenarios, and their application towards the achievement of its main goals.

Also, the document reports the development process of the system together with the referenced resources and a description of its structure.

This document has the purpose to guide the developer in the realization of the system called Customers Lineup(CLup), an application developed to fight the Covid pandemic.

1.A Purpose

The world has been undergoing a new pandemic, that has been putting a strain on society on many levels, like in the case of the imposed lockdowns that allow people to exit their homes only for essential needs. Also, grocery shops have been restricting access to their stores to avoid having a great influx of people, which typically results in long lines forming outside, which are themselves a source of hazard.

The goal of this project is to develop a user-friendly application that not only allows store managers to regulate the influx of people in the building but moreover saves people from having to line up and stand outside of stores for hours on end, putting their health at stake.

Customers Line up offers three main functionalities:

1. The system allows a customer to book the first available slot for the line of a selected supermarket. The application manages the virtual lines

avoiding any kind of overcrowding inside the building and it notifies the user about the estimated waiting time, according to his location.

Also, the system takes into account the people who don't have the proper technology, giving them the possibility to book a visit outside the market itself.

2. The system allows people to book a visit to the supermarket, indicating the day, the time, the expected duration of the visit, and the categories of items they intend to buy. In this way, the system is able to plan visits in a finer way and without overcrowding the store.

3. The system suggests alternative slots for visiting the store if the preferred one is not available or too crowded. This would allow us to balance out the number of people in the store.

The goals are represented in the table below:

G1	Allow store managers to regulate the influx of customers inside the supermarket
G2	Let people avoid to line up and stand outside of stores creating assemblage through the creation of a virtual line
G3	Ensure social distancing between customers into different store department
G4	Allow customers to book a reservation to do shopping in a store
G5	Distribute the flow of the customers in multiple stores avoiding over-crowd them

1.B Scope

1.B.1 World Phenomena

- The user decides to shop in a supermarket

- The user does shopping in the supermarket
- The user leaves the supermarket
- The user takes a smartphone with himself

1.B.2 Shared Phenomena

- The system receives a request for a reservation for a specific store and for a selected time slot.
- The system returns a possible free time.
- The system receives a request for a reservation in an available supermarket in the user area.
- The system suggests a less crowded store
- The **online user** receives the available slots for the chosen supermarket
- The **online user** confirms the suggested/estimated time and confirms the optional store creating the reservation
- The **online user** receives the ticket from the system
- The system accepts the reservation
- The **online user** places his smartphone display in front of the QR-code reader at the store entry.
- The system sends the **online user** the confirmation of a reservation
- The system sends the **online user** the notification that his reservation has been cancelled
- The system sends the **online user** the notification that his turn is approaching
- The system sends the **online user** the notification that his preferred supermarket and time slot is available
- The system sends the **online user** the notification that he didn’t go to the right supermarket or on the right day/time
- The **offline user** places his printed ticket in front of the QR-code reader of the store entry.
- The **online use**r places his smartphone display in front of the QR-code reader at the cash register when he ends up with a shopping
- The store manager sets the open days and time of the supermarket

- The store manager sets the max capacity of customers that can shop at the same time.
- The store manager receives statistics about reservations

1.C Definitions, Acronyms, Abbreviations

1.C.1 Definitions

Confirm notification	Whenever a user books a reservation for a supermarket, he receives a confirmation message in which there are the number of the ticket, the QR code, the expected waiting time and a precise time and day(in the case the user decides at what time and day he wants to go shopping)
Delete notification	Whenever a booked slot is cancelled either by the user or by the system the user receives a message describing the cause of cancellation.
Wrong notification	Whenever a user scans his QR code but at a different time or supermarket he receives a message reminding him of his bookings (if present).
Free notification	The message sent by the system that notifies the user that a new slot for the time/day and supermarket he selected as preferred is available.
Approaching notification	Whenever the turn of a user is approaching, he receives a notification.(It depends on his current position and his number in the queue)
Queue	It defines the virtual line.

To book	To take a slot in the queue of a supermarket as soon as possible or on a specific date and time.
Turn	The user slot in the queue
Ticket	A reservation for a slot in the queue
Online user	User that makes a online reservation with a internet connected device
Offline user	User that makes a reservation outside a supermarket through automatic ticket machine
Time slot	Range time(1: 00 H) in which you can book a turn
ASAP turn	The first available turn
Safety Margin	The number of available customer slots inside the supermarket that are used only in case of multiple delays or for a finer affluence of people.
Success ring	Sound coming from the QR reader

1.C.2 Acronyms

RASD	Requirement Analysis and Specification Document
CLup	Customers Line-up
GPS	Global Positioning System
QR	Quick Response

1.C.3 Abbreviations

CONNOT	Confirm Notification
DELNOT	Delete Notification
WRONOT	Wrong Notification
FREENOT	Free Notification
APNOT	Approaching Notification
WPn	World Phenomena number n
SPn	Shared Phenomena number n
Gn	Goal number n
Rn	Requirement number n
ASAP	As soon as possible
Cn	Constraint number n

1.D Revision History

21/12	❖ Mockup of the structure
6/11	❖ First component diagram ❖ FIrst interfaces
13/11	❖ Changes to the requirements. ❖ Addition of the definitions, acronyms and abbreviations. ❖ Addition of the main goals.
16/11	❖ Domain assumptions

	❖ Specification of world and shared phenomena.
22/11	❖ Added use cases ❖ Added non functional requirements (security,…)
24/11	❖ Added use case diagram ❖ Added flow diagrams ❖ Added object diagram
29/11	❖ Added chapter 3
5/12	❖ First full revision ❖ Added mapping of the requirements
10/12	❖ Added Alloy code ❖ Added metamodels
15/12	❖ First full revision
20/12	❖ Second full revision

1.E Reference Documents

- Specification Document: “CustomerLineUpProject Assignment.pdf”.
- Slides of the lectures.
- I. Sommerville, Software Engineering 10th edition

1.F Document Structure

- **Chapter 1:** It gives an introduction to the purpose of the document and the developed system. In particular, the specifications such as the definitions, acronyms, abbreviation, revision history of the document and the references, are gathered and described. As one of the main focuses, the chapter also specifies the main goals and world and shared phenomena of the system.

- **Chapter 2:** It focuses on the full description of the system and how it works. First of all the document contains a brief description of the main actors that interact with the developed system. Later on, there is the model description through a class diagram together with the statechart of the main function of the application. Lastly, the assumptions, dependencies and constraints that help define the limits and necessary functions of our system.
- **Chapter 3:** It describes the user, hardware, software and communication interfaces and their requirements. Moreover, the document contains the functional and non-functional requirements together with their sequence diagrams and their correlations with the domain assumption and the main goals that help us understand how the latter are satisfied. In addition, there are some possible scenarios and use cases that show users' interactions and intentions towards the system and its the relative response.
- **Chapter 4:** It includes the alloy code needed to test the feasibility of the system and the corresponding metamodels generated from it.
- **Chapter 5:** Shows the individual effort put towards the development of the document and of the project in general.
- **Chapter 6:** Includes the referenced documents for the production of the chapters above and the project altogether.

2. OVERALL DESCRIPTION

2.A Product Perspective

2.A.1 Scenarios

Scenario 0: General without choosing a slot

Caterina is a young woman. She decides to reserve a ticket at the supermarket ,where she usually goes to, with the Line-up mobile app. This one sends her a digital ticket in which there is the number of the ticket, the QR code, the expected waiting time. When her turn is approaching, she receives a notification and so she walks towards the store. Once she arrives, she notices on the outside screen that it is her turn. Then, she enters the store, scans her QR code and starts shopping. After buying groceries, she goes to the cashier. After that, she scans her QR code and eventually she pays.

Scenario 1: General

Nunzia is a mum of three children and she goes shopping for her family every Saturday. Since she doesn't want to go through crowded places, she decides to use Customers Line-up to avoid waiting in line in presence. Therefore, after accessing the app, she decides the supermarket she wants to go to, selects a time slot among the ones available on that day, and adds the expected duration of the visit. After that, she receives a digital ticket in which there is the number of the ticket, the QR code, the precise time and day when she can go shopping. She will receive multiple notifications, when her turn is approaching. Before accessing the building, she searches for her number on the outside screen and sees that she is able to enter. Afterward, she scans her personal QR code and begins buying groceries. When she goes to the machine cashier, firstly she scans her QR code again and then she pays.

Scenario 2: (double possibility: choose the supermarket or the date)

Giovanni is an offside student that lives in Milan and, due to the pandemic, wants to book his visit to the supermarket so he can shop in safety for his covid-free party. He decides the day and time slot in which he has some free time and the app shows which supermarkets are less crowded in the nearby area. In addition, after deciding the store, he checks the exact categories of items that he intends to buy, that is delicacies and liquors. CFR scenario 1.

Scenario 3: Wrong market and turn

Giampiero is quite a scatterbrain person. He has booked a spot for his usual supermarket but forgets the day and time and presents himself a day before the appointment. Once arrived, he does not see his number on the screen but tries to scan his QR code at the entrance anyway. At that time Giampiero's phone receives a notification reminding him that his turn is the next day at the appointed time. Giampiero dams himself for his careless behavior and goes back home.

Scenario 5: User who is too late or does not show up

Marianne is an extremely busy person. She decides to book a slot at the supermarket with the CLup. She notices that there is an available slot at 17:15 and decides to book it. The time looks perfect for Marianne's personal schedule, but when she is ready to go downstairs, she receives a work call from her boss and she won't in time for her appointment. Marianne's Boss keeps her on call for 30 minutes after which Marienne notices she received in the meanwhile a notification that informs her that her reservation has been cancelled and she needs to make another reservation.

Scenario 6: User who has to wait outside at the supermarket

Camilla is a young businesswoman. She decides to book a ticket at the supermarket where she usually goes with the CLup mobile app. This one gives her a number, a QR code and the expected waiting time. When her turn is approaching, she receives a notification and so she walks towards the store. Once she arrives, she notices on the outside screen that it isn't her

turn yet. This is why some people inside the supermarket are taking longer than expected. Therefore, she has to wait outside until one person finishes shopping. When her turn comes, she scans her personal QR code and enters the store. She buys what she needs and eventually goes to the cashier. Then, she scans her QR code and after doing that, she pays and finishes her shopping.

Scenario 7: User who has no technology

Antonio is an elderly person. Like every week, he goes shopping at his usual supermarket. Once out at the supermarket, he retrieves a ticket with an estimated time for his turn, a number and QR code from the automatic machine on the spot. He looks at time on the ticket and he realizes that he has enough time to go to collect his pension before his turn comes. After that, he comes back to the supermarket and he realizes from the screen that he can enter the supermarket. So he scans the QR code present on the ticket and starts shopping. After buying some groceries, he goes to the cashier, scans his QR code and pays.

Scenario 8: System gives alternatives

Maria wants to book the first available slot for the supermarket around the corner. However, before completing the reservation the application advises her that the supermarket is quite full at the moment and that this could cause an hazard for Maria's health. For this reason, the application gives Maria some alternative markets that are less crowded and that are in her neighborhood. Maria, who's a responsible person, chooses one of the suggested supermarkets and heads towards it to do her grocery shopping. CFR scenario 1.

Scenario 9: System gives periodic notifications

Bianca is a mother of two and today, like most of the week days, she would like to go to the supermarket afterwork. Due to the pandemic she only goes grocery shopping if she can do it in safety. Luckily the application notifies her that in the time slot that she selected as her preferred, after her

registration in the application, there are still some open positions and that she should book one as soon as possible. Bianca quickly gets a ticket and relaxes knowing that she will be able to buy dinner for her sons without taking any risks.

Scenario 10: Modification of a reservation

Fiore is a young man that decides to go shopping to help his sick grandmother. So he books a slot with CLup at the supermarket near his grandmother's house at 17: 30. After that, he receives a ticket. An hour before the reservation, he realizes that he isn't able to go shopping since he does not finish homework yet. Therefore, he decides to modify his reservation. He opens the app and checks if there is a slot available at 18: 30. Fortunately, there is the last available slot for that time. So, he can modify his booking. After that, he receives another ticket. Everything than proceeds as described in Scenario 1.

Scenario 11: Store Manager

Luca is the store Manager of the Pam supermarket present in the Città Studi neighborhood. Due to renovations he had to change the shelves disposition. However, this brought a decrease in the walkable space for the customers and for this reason he needs to update the max number of people that can enter the store. So he uses the CLup to see the statistics of the past months and then changes the max feasible q.ty of people inside the market.

2.A.2 Class Diagram

The main structure and functions of the system are described through the UML below. This gives a high level model of the CLup where all the main components are specified. A deeper explanation of the different relations is given below. However for simplicity sake only the main entities of the system are present and carefully described.

- Whenever a user books a turn a reservation is created that saves the date, time, the number and QR code assigned and the optional values like the estimated time spent shopping. Moreover a customer can specify in which departments he intends to visit and in what time slot he would like to shop. The reservation is then assigned to the unique queue of the selected supermarket.
- Each supermarket has a defined max quantity of customers that can shop at the same time and a safety margin that can be exploited whenever the system feel s necessary. Each supermarket has one or more managers and at least one display and multiple QR readers.
- Every store has his own coordinates. That is, the system can then calculate the distance and time needed by the user to arrive at the supermarket, in case he has the GPS enabled. The supermarket can be composed of different departments that an online user can specify in his reservation to allow a finer management of the influx of customers.

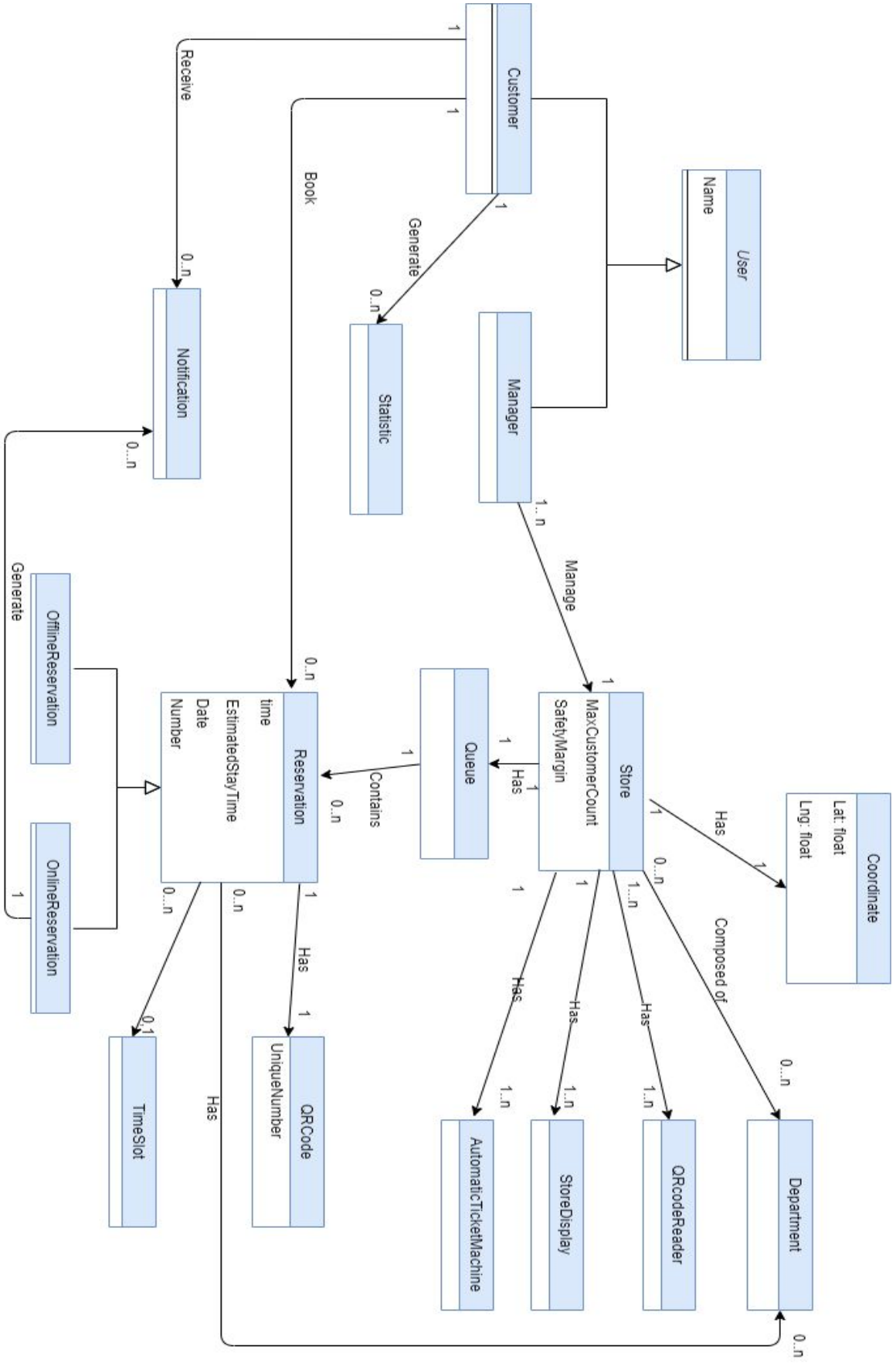

Image 0: CLup-Class diagram

2.A.3 State Charts

This paragraph describes and analyzes through statecharts some of the main functionalities of the developed system. As a matter of fact, it's shown their development through time and the responses given according to specific inputs.

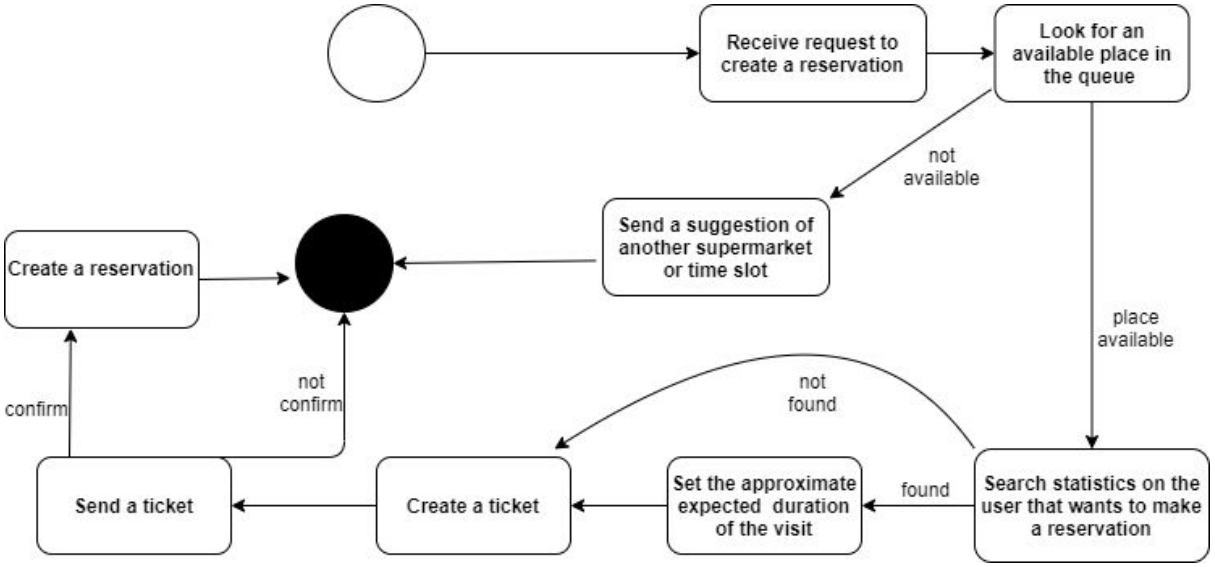

Image 1: State diagram 1

In the first statechart we model the creation of a reservation made by the system. The diagram starts with a request by the user to make a reservation in a specific supermarket(the request refers either to a specific time slot or to an ASAP turn). If there is a place available in the queue, the system checks, when possible, if the user has already visited the supermarket. If so the system assigns the customer an expected duration of the visit based on an analysis of his previous visits. Later on, the system creates a ticket and sends it to the user. Finally, if the latter confirms the ticket, the system creates the reservation. In case there isn't an available turn, the system sends the user some suggestions of other available supermarkets or available time slots.

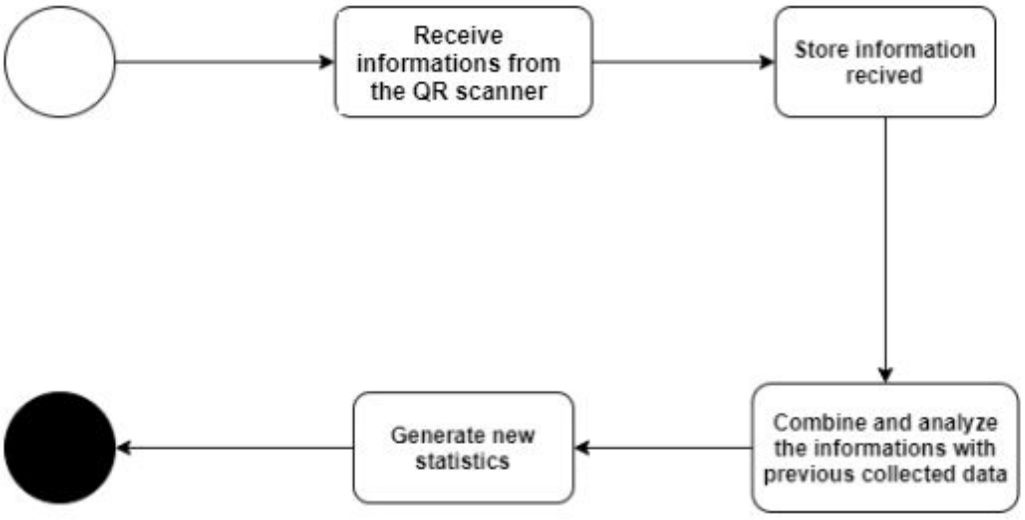

Image 2: State diagram 2

In the second statechart it's described the process of gathering and analyzing data of the customers. Every time that an online user scans his QR code before exiting the store, the system will collect infos like the time spent inside the building and the selected departments to better understand his behaviour. In this way the system is able to better manage the influx of customers inside the supermarket and to give more precise suggestions to the store manager.

2.B Product Functions

Since January 2020 the world has been fighting a strenuous battle against the new Covid pandemic. Thanks to its low death rate and high spreading capacity the virus was able to reach many of the world nations in a short period of time, putting all governments on alert. As a matter of fact, all the hit nations have seen themselves constrained to find solutions capable of containing the exponential growth as soon as possible. They found the best ones in modern technology like in the case of the contact tracing systems. However with the reopening of the activities and the pandemic still ongoing other applications must be developed to maintain control over the virus' spread and to protect the population's health. One of the many hazardous situations in which a person is often put through is standing in

line outside shops and more specifically, outside the supermarkets. As a matter of fact, especially for small markets, it could be difficult to maintain a line where everyone respects the proper social distance especially due to the large amount of people that go grocery shopping daily.
The main function of the developed system is to allow customers to line up for a supermarket in a digital way avoiding the presence of a fiscal line and all the hazards that are connected to it. A person can book a turn in line in two main ways:

Online booking is available to all customers that have the proper technology like a tablet or a smartphone and a working internet connection. With the mobile app a registered customer can start his booking procedure either by choosing a supermarket or a day/time slot. Regarding the shops the user can search for a specific one or, if he enables his GPS, he can look up what supermarkets are near his location. Regarding the time of day, the shopper can decide to book an ASAP turn (limited to the day in which the reservation is done) or a turn in a future time slot. After selecting the time/shop he can specify the estimated shopping time and the categories of product he intends to buy. Once the consumer has made his choice the system can either advise him that the building could be crowded, and suggest other time slots or shops, or send him directly a CONNOT. An APNOT is sent to the user 15 minutes before his turn. The time is calculated either according to the current position of the customer and the persons already inside the shop, if he has enabled his GPS, or just by the latter one. If a problem of any kind should assurge then the customer is able to modify or cancel his booking at any time.

Offline booking is available to anyone but specifically targeted to those without the proper devices to make an online booking. Outside all the affiliated supermarkets there is an automatic ticket dispenser that can book an ASAP turn and print the needed QR code, number of the place in line and expected time of entrance.

The queue is managed by the application and to work properly needs different parameters. First of all the store manager should insert both the maximum number of customers that can stay at the same time inside the shop and the Safety Margin. These parameters can be suggested by the system if the proper planimetry is given. Secondly the application needs an estimation of the time that the customer will probably spend inside the supermarket. The value can either be inserted by the user or calculated on the past visits if the latter is a common customer. In absence of the previous estimations then the system will assign the booking a default time of 20 minutes. Moreover, if many customers select the category of the product they intend to buy, the system is able to use this notion, increasing the affluence of customers inside the supermarket.

In the end, thanks to the parameters above the system is able to give an estimated turn time to the online and offline customer and he is able to use part of the Safety Margin in case multiple customers have a delay or they select the category of the product they intend to buy.

The system is able to track the persons inside the building thanks to the QR code given for each booking that must be scanned at the entrance of the supermarket by the customers whenever their turn comes up on the outside screen of the shop. In addition, the scan of the code is necessary to enable the payment at the cash registers. As a matter of fact the clientele needs to scan a second time the unique code whenever they are ready to finish their shopping so they can either activate the automatic checkout machine or be given the permission to put the groceries on the automatic belt of the serviced cash register. In this way the system discourages any malicious subject to enter the supermarket freely and/or jump the line.

As an additional feature any online user can receive daily notifications about the available turns for a preferred time slot and supermarket.

In the list below there are the main requisites that the system satisfies in order to work as just described.

List of requirements:

R1 The system should allow the user to register

- **Ra** Through OAuth platforms (Google, Apple, Facebook…)
- **Rb** Through filling email and password fields

R2 The system should allow the user to retrieve the lost password.

R3 The system should be able to show the nearby supermarkets based on the location of the user(if the GPS is on).

R4 The system should allow the user to choose the supermarket where he wants to shop.

R5 The system should provide users with a reasonably precise estimation of the waiting time.

R6 The system should allow users to confirm a reservation.

R7 The system should generate a unique QR for each reservation.

R8 The system should generate a unique line number for each reservation.

R9 The system should send a ticket to the user through a notification .(CONNOT)

R10 The system should be able to notify the user about the approaching of his turn based on his geographical position(if his GPS is on) and his turn in the line.(APNOT)

R11 The system should be able to maintain a count on the people inside the supermarket.

- **Ra** After scanning the QR code, the system should notify the user if he is able to enter the supermarket.
- **Rb** If the supermarket is fuller than allowed(number of customers > max - Safety Margin), the system must be able to invalidate the user's QR-code until the number of people inside is equal to the max inserted minus the Safety Margin.

R12 The system should notify the user if the chosen supermarket/date/time slot is too crowded and should give an alternative.

R13 The system should allow the user to choose the day for a booking to the supermarket.

R14 The system should allow the user to select the time slot for a booking to the supermarket.

R15 The system should allow the user to indicate the approximate expected duration of the visit at the supermarket.

R16 The system should be able to assign users a default time of 20 minutes in case he doesn't choose the expected duration of the visit.

R17 The system should be able to estimate a duration time for a frequent customer based on his previous visits.

R18 The system should be able to make statistics based on the daily reservations and the time spent inside the supermarkets by the customers.

R19 The system should allow users to indicate the departments of items that they intend to visit to allow a finer management of the influx of customers.

R20 The system should notify users periodically if there are any available turns, in the nearby supermarkets, in a selected day/time range . (FREENOT)

R21 The system should cancel a reservation if the user doesn't scan the QR code at the entrance after 15 minutes from the set time and notify him.(DELNOT)

R22 The system should allow the user to cancel his reservation.

R23 The system should allow the user to modify his reservation.

R24 After scanning the QR-code, the system should notify the user if he makes a mistake(WRONOT).

R25 The system should be able to manage bookings from different sources (smartphone, outsidex box).

R26 The system should allow the store manager to choose the maximum number of customers that can enter the supermarket.

R27 The system should not allow customers to book a turn in case the chosen time slot has reached the established threshold.

R28 The system should allow people without device to reserve a ticket

R29 The system should invalidate any reservation which QRcode isn't scanned after 15 minutes of his turn.

R30 The system should give suggestions to the store manager regarding the feasible max amount of customers inside the shop.

R31 The system should allow store manager to manage the department of his supermarket

2.C User characteristics

The actors of the application are the following:

1. **Customer:** someone who wants to go shopping at a supermarket that makes available the possibility to use CLup. Since there is the possibility to reserve a slot also through an automatic machine outside the supermarket, there are two types of customers:

- Online customer that makes a reservation through the use of a device
- Offline customer that makes a reservation outside a supermarket through automatic ticket machine

2.Store Manager: The responsible for overlooking that the right influx of customers is present inside the supermarket. Thanks to CLup he can:

- Study the statistics provided by the application through the study of the bookings received. Thanks to this data, the manager can make more precise decisions towards the change in the influx of customers inside the shop and the disposition of the shelves inside the shop so as to keep his clientele in a safe environment.
- Modify the quantity of customers that can enter the building at once so to give them enough space to enable social distancing. He could refer to the number given by the application itself according to its estimation or could give a precise quantity if he prefers so.

- Modify the available slots and/or move the already present bookings in case there is an emergency or interruption of any kind.

2.D Assumptions, Dependencies and Constraints

D1	Each **online user** has a smartphone and a working internet connection during the entrance and during the payment in the supermarket
D2	Each offline user preserves his ticket
D3	Each smartphone has a display capable of showing a clear qr-code to be scanned
D4	Each partner store has a working internet connection
D5	Every partner store has a QR-code scanner at the entry and one at each cash register (automatic or with personnel)
D6	Each partner store has a display to show the current and incoming turn numbers, outside of the store
D7	Each partner store has a ticket machine for **offline users** outside the supermarket
D8	The QR-code readers in the store work properly.
D9	Only **online users** and **offline users** enter the store
D10	When users scan their QR-code at exit, they actually go out from the supermarket

3. SPECIFIC REQUIREMENTS

3.A External Interface Requirements

3.A.1 User interfaces

The app must be designed to be coherent to modern mobile app guidelines.

Its usability must permit non-usual smartphone users to make a reservation

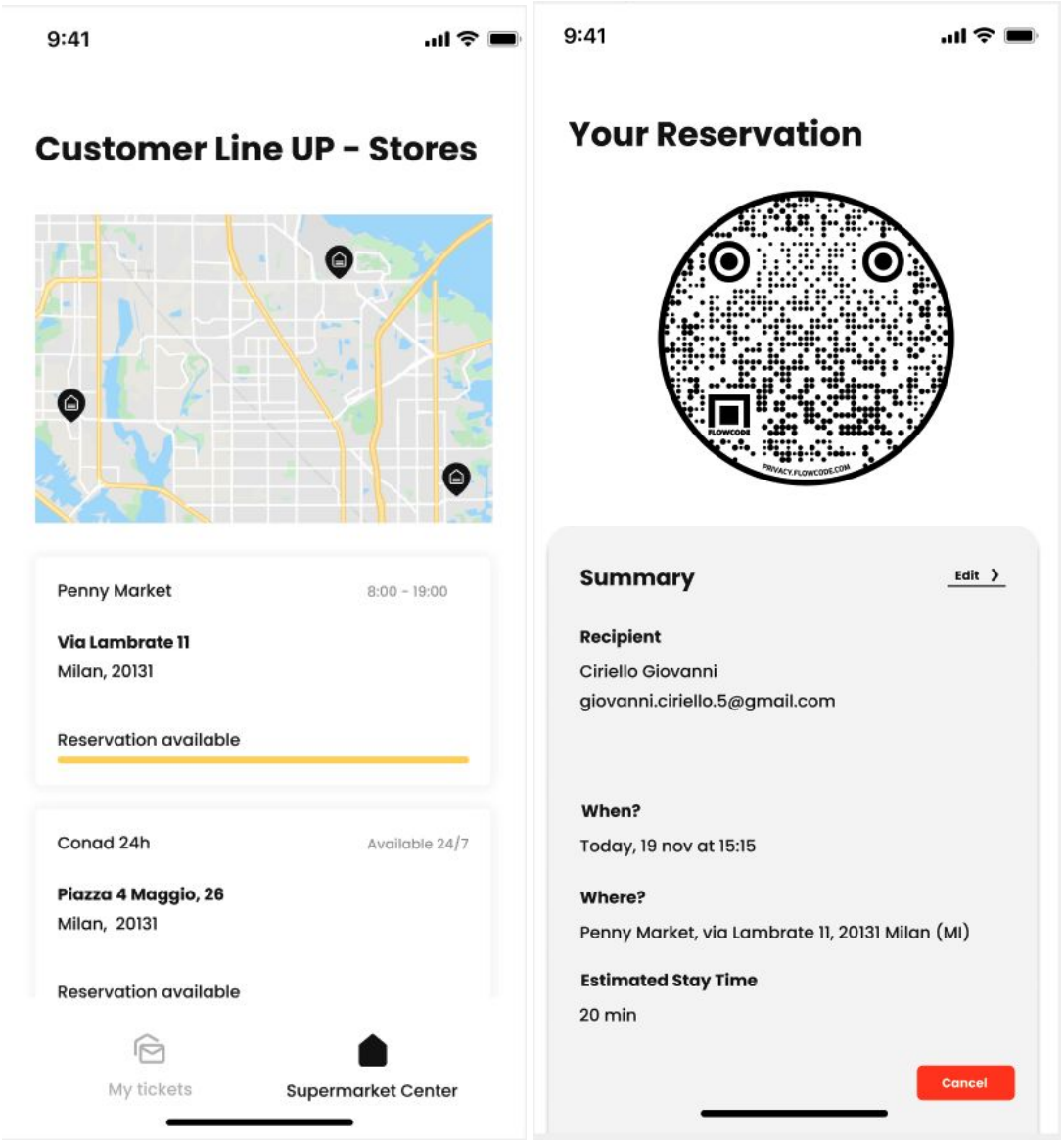

Image 3: Mockup user interface

3.A.2 Hardware interfaces

- Regarding the store customer that makes an online reservation, he must own a smartphone connected to the internet, optionally with GPS sensor if he wants to receive suggestions about near supermarkets based on his location.
- Regarding the store, it must be equipped with a QR code reader at store entry and a QR code reader for each cash register(automatic or with personnel). Each QR code must be integrated in a connected chip that makes a secure http request once it reads the encoded URL.
  The store must be equipped with a display outside the supermarket that shows the served number of the line. In this way, the customer has the possibility to know precisely, once he arrives outside the supermarket, when he can enter the store.
  Finally the store must have a ticket machine outside the supermarket to allow people without smartphones to book a turn.
- Regarding the store manager, he can access the dashboard with his smartphone, tablet or computer.

3.A.3 Software interfaces

The application is using external services in order to simplify the phase of development.

- **Google, Facebook, Apple Sign-in API** through OAuth 2.0 to authenticate users in a quicker way.
- **Google Maps API** to show to store customer map frames containing nearest supermarkets as map pins (markers)
- **Google Geocoding API** to convert string-like address inserted by users in geocoded coordinates. They will be suitable to calculate distances between places and show to user the minutes-walking distance
- **Intercomm API** to provide the user a easy way to contact the CLup Customer Support in case via a real-time chat

3.A.4 Communication Interfaces

Both Customers' mobile app and Manager web app use HTTPS protocol to send and receive data. This protocol, which supports TLS, makes secure the data exchange between client apps and server.

3.B Functional requirements

3.B.1 Use Cases

The paragraph collects the use cases that describe, through the interaction between man and system, the main functionalities of the developed application. The relation among all the use cases is specified through the use case diagrams in image 4. For a cleaner raffiguration, the graphs also include the use case "Book a turn". It is a generalization of the three cases that describe how a customer can reserve a place in line. Moreover, it specifies how the booking of a turn is related to the other main use cases like "Log in of user" and "Shop in the supermarket".

1.Registration of user

Name	Registration of user
Actors	User
Entry condition	User has installed the application on his device
Event flow	1. User launches the app 2. User clicks on ‘Sign up’ button 3. He fills all fields or signs in through another platform (Google, Facebook…) 4. User chooses the city in which he lives 5. User accept privacy conditions 6. The user on ‘Confirm’ button 7. The system confirms the registration of the user 8. The system saves his personal data 9. The system asks the user for GPS permission r 10. The user makes a choice about the GPS permission
Exit conditions	User is registered and is logged in the application
Exception	1. User is already present in the system 2. User did not fill up all mandatory fields with valid data If one of these situations happens, the application throws an error message and will return to the registration form page.

2. Login of User

Name	Login of the user
Actors	User
Entry condition	User is already registered to the application service
Event flow	1. User accesses the application through its device 2. User compiles the fields ‘Username’ and ‘Password’ 3. User clicks on ‘Login’ button 4. The system opens the application home page
Exit conditions	The user has successfully logged in and can access the CLup
Exception	1. User enters invalid Username 2. User enters invalid Password If one of the above conditions is detected, the application notifies the User taking it back to the login screen

3. Book online an ASAP Turn

Name	Book online an ASAP turn
Actors	Online user
Entry condition	Online user has already logged in the application
Event flow	1. The user click on “book a turn” 2. The user click on “choose a supermarket” 3. The system shows all the available supermarkets 3.1. if the user granted the GPS permission, the interactive map shows only nearby supermarket based on his current location 3.2. if the user didn’t grant the GPS permission, the map shows supermarkets based on the city inserted during the registration 4. The user chooses a supermarket 4.1. optionally he can select the category of products that he intends to buy 5. The system looks for the first available turn in line 6. The system shows the user the waiting time 6.1. if the supermarket is crowded in that estimated time the system advises the user and suggest other time slots or stores 7. The user clicks on “Confirm” button 8. The system sends him a ticket (CONNOT)
Exit conditions	The user has successfully booked a turn in line
Exception	1. Supermarket is full for the whole day. If it happens, the system notifies him that all reservations are taken, and suggest other time slots or store 2. If the user want to reject the suggested time, he clicks on the “Cancel” button and he doesn’t receive any ticket

4. Book a turn for a time slot/day

Name	Book a turn for a time slot/day
Entry condition	Online user has already logged in the application
Event flow	1. The user click on 'book a turn' 2. The user clicks on 'choose a supermarket'' 3. The system shows all the available supermarkets 3.1. if the user granted the GPS permission, the interactive map shows only nearby supermarket based on his current location 3.2. if the user didn't grant the GPS permission, the map shows supermarkets based on the city inserted during the registration 4. The user chooses a supermarket 5. The user chooses a preferred day and time slot 6. Optionally the user selects the category of the products that he intends to buy and the estimated time he will spend inside the store 7. The user clicks on "Reserve" button 8. The system looks for a available turn inside the chosen time slot 9. The system shows the user the day and the calculated time 9.1. if the supermarket is crowded in that estimated time the system advises the user and suggest other time slots or stores 10. The user confirms the reservation. 11. The system sends him a ticket (CONNOT)
Exit conditions	The user has successfully booked a turn in line in the preferred time slot
Exception	3. Supermarket is completely full in the chosen time slot If it happens, the system notifies him that all reservations are taken, and suggest other time slots or store

5. Book offline an ASAP turn

Name	Book offline an ASAP turn
Actors	Offline user
Entry condition	User outside the supermarket wants to go grocery shopping in the partner store
Event flow	1. The offline user clicks on ‘Reserve’ button on the ticket machine outside the supermarket 2. The system looks for the first available turn in line in that supermarket 3. The system launches the print through the ticket machine 4. The user picks and conserves the printed ticket
Exit conditions	The offline user has successfully booked a turn in line
Exception	1. Supermarket is full for the whole day and the user can’t obtain his reservation

6. Shop in the supermarket

Name	Shop in the supermarket
Actors	Online or offline user
Entry condition	User with his reservation, is outside the chosen supermarket and his turn has come up
Event flow	1. The user looks at the outside monitor for his number and sees it. 2. The user places his smartphone or his ticket with the QRcode in front of the reader. 3. The system validates the read QRcode and approves it playing a “success ring”. 4. The system updates the count of the people inside the building. 5. The user enter in the supermarket and goes grocery shopping 6. The user show his QRcode again during purchase process a. If the user uses an automatic cash register, he shows the QRcode to the barcode reader. b. If the user uses a cash register with personnel, the cashier scans the QR code of the customer. 7. The system updates the count of the people inside the building. 8. User takes his purchased grocery and goes out from the supermarket
Exit conditions	The User has finished his grocery shopping and has exited the building
Exception	1. The user goes to the wrong supermarket or at the wrong time If it happens, the system, as soon as it reads the QR code, it plays the “error ring”, and only for the online user the application shows a WRONOT.

7. Cancellation of an online reservation

Name	Cancellation of an online reservation
Actors	Online user
Entry condition	User has successfully taken a reservation and wants to edit the time or the supermarket
Event flow	1. User goes to “my tickets” section in the app 2. User selects a reservation and clicks on “cancel” button 3. The system removes his reservation updating the queue 4. The system sends the user a DELNOT
Exit conditions	The User has successfully canceled his reservation
Exception	The reservation refers to a day and time in the past and it can’t be canceled anymore

8. Setting the periodic notifications

Name	Set the periodic notifications
Actors	Online user
Entry condition	Online user has already logged in the application
Event flow	1. User launches the app 2. User goes to settings > custom notification section 3. The system shows the custom notification page 4. The user chooses the preferred supermarket 5. The user chooses the preferred time slot 6. The user chooses the frequency or the week days in which he desires to receive the availability notification (FREENOT).
Exit conditions	User has set a periodic notification and he will update when the chosen supermarket has an available turn in the chosen time slot.
Exception	User chooses to receive the FREENOT for a day in which the chosen supermarket is closed

9. Modification of the influx of customers

Name	Modification of the influx of customers
Actors	Store manager
Entry condition	The store manager has logged in
Event flow	1. The store manager pushes the “change max. q.ty” button on the console 2. The system gives the manager a form to compile 3. The store manager inserts the new blueprint of the supermarket 4. The system analyzes the print and calculates an optimal max quantity of customers and an alternative optimal disposition of the shelves 5. the system shows the suggested results to the manager 6. The store manager inputs the chosen maximum quantity and the **Safety margin** 7. The manager confirms the changes
Exit conditions	The manager has successfully updated the max amount of customers that can shop at the same time
Exception	1. The value inserted by the manager is invalid or the Safety margin is more than 50% of the max amount The system shows an error message that indicates the fields to modify.

10. Build statistics

Name	Build statistics
Actors	User
Entry condition	The user places his smartphone with the QRcode in front of the reader.
Event flow	1. The system validates it and starts to count the time spent by the user inside the supermarket. 2. The user scans his QRcode again during purchase process 3. The system estimates the time spent by the user inside the supermarket
Exit conditions	The system build statistics

11. Receive an approaching notification

Name	Receive an approaching notification
Actors	User
Entry condition	The user has a ticket for a supermarket
Event flow	1. The system estimates, according to his position(if the GPS is on) and his number in the queue, that his turn is approaching 2. The system sends the user a notification (APNOT) 3. The user receives the notification
Exit conditions	The user has successfully received the notification

3.B.2 Use Cases Diagrams

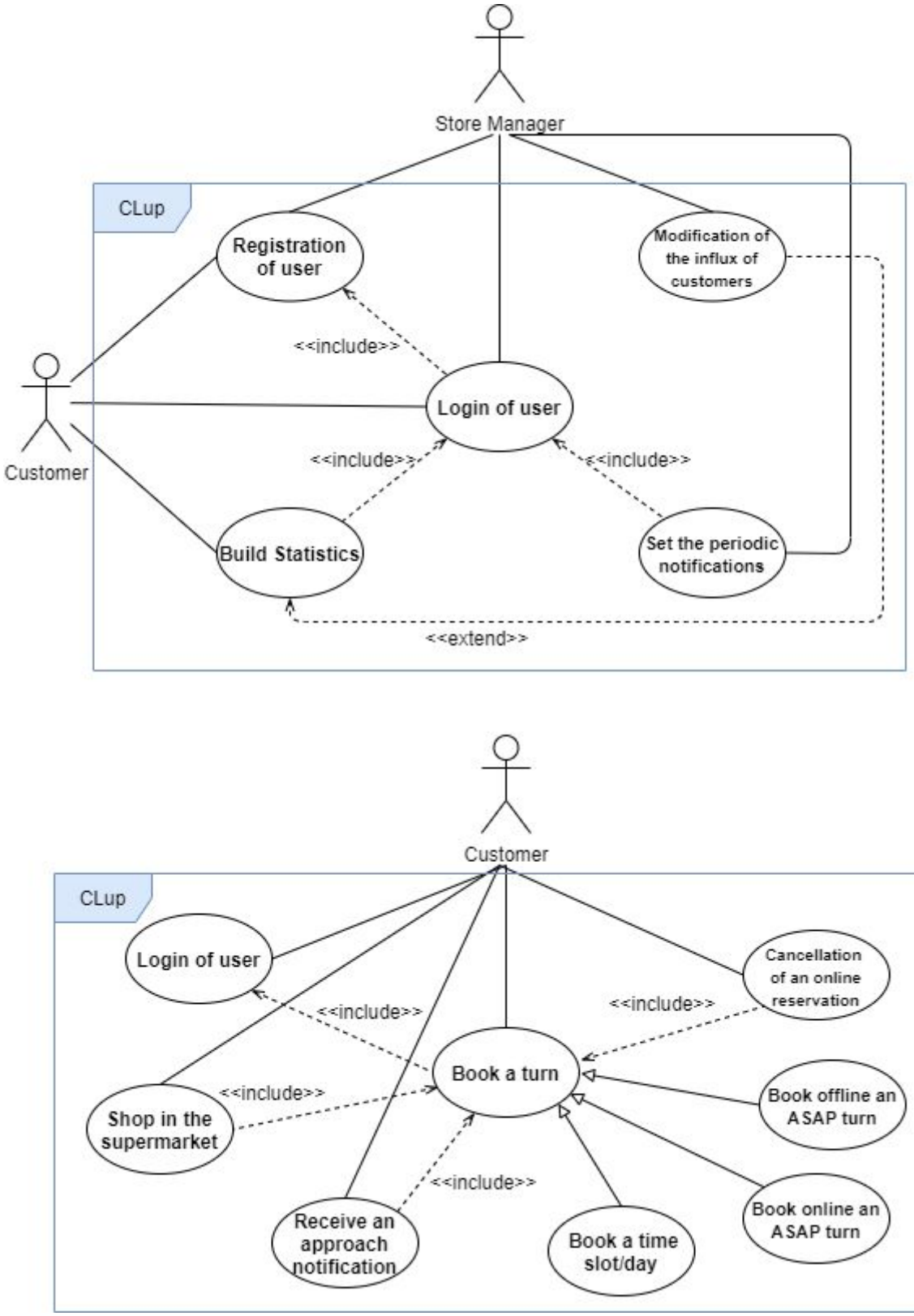

Image 4: CLup-Use Cases Diagrams

3.B.3 Sequence Diagrams

1. Registration of user

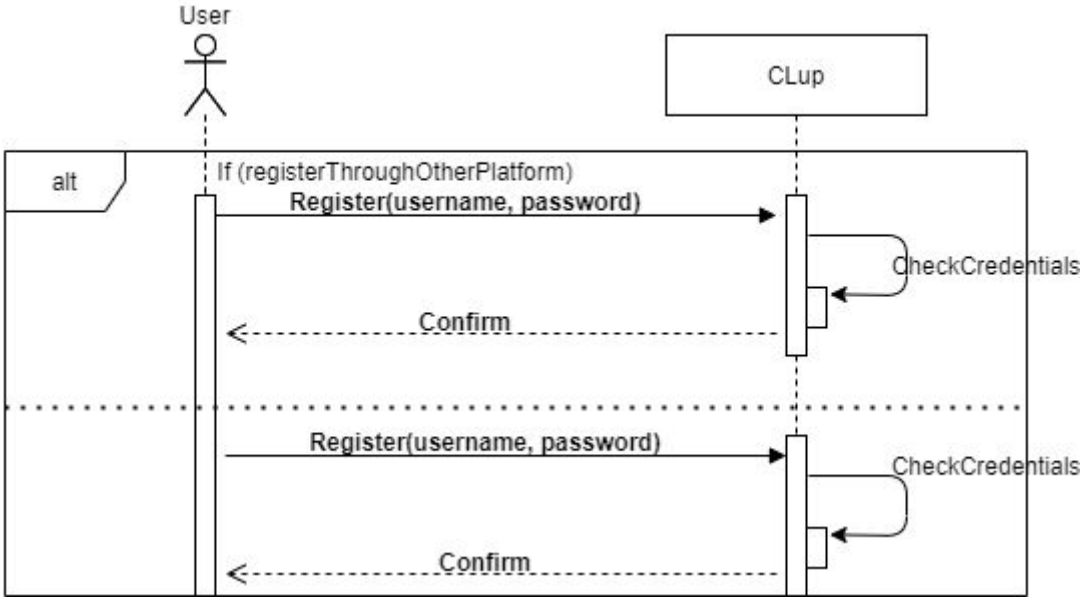

Image 5: Use Case 1

2. Login of user

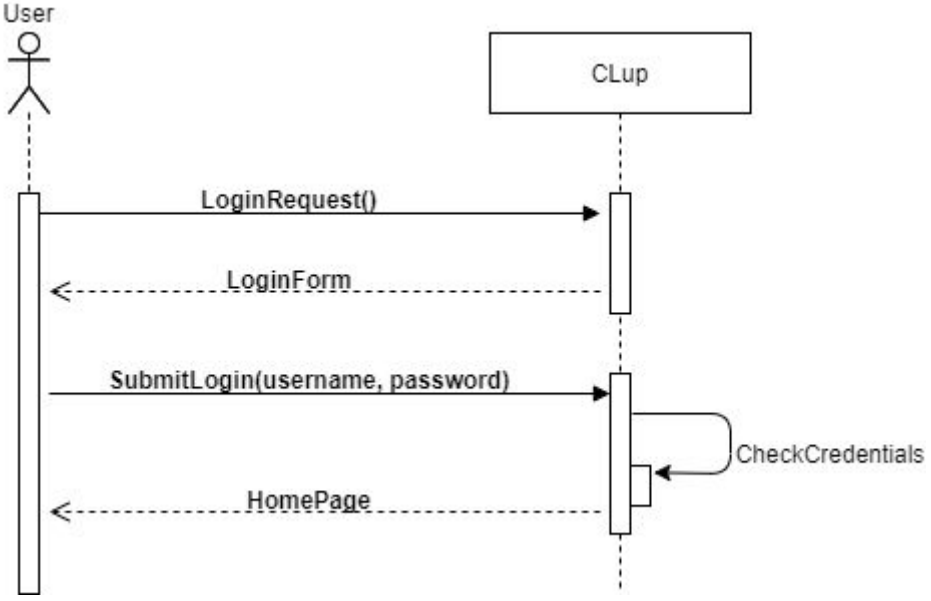

Image 6: Use Case 2

3.Book online and ASAP turn/ 4.Book a turn in a time slot/day

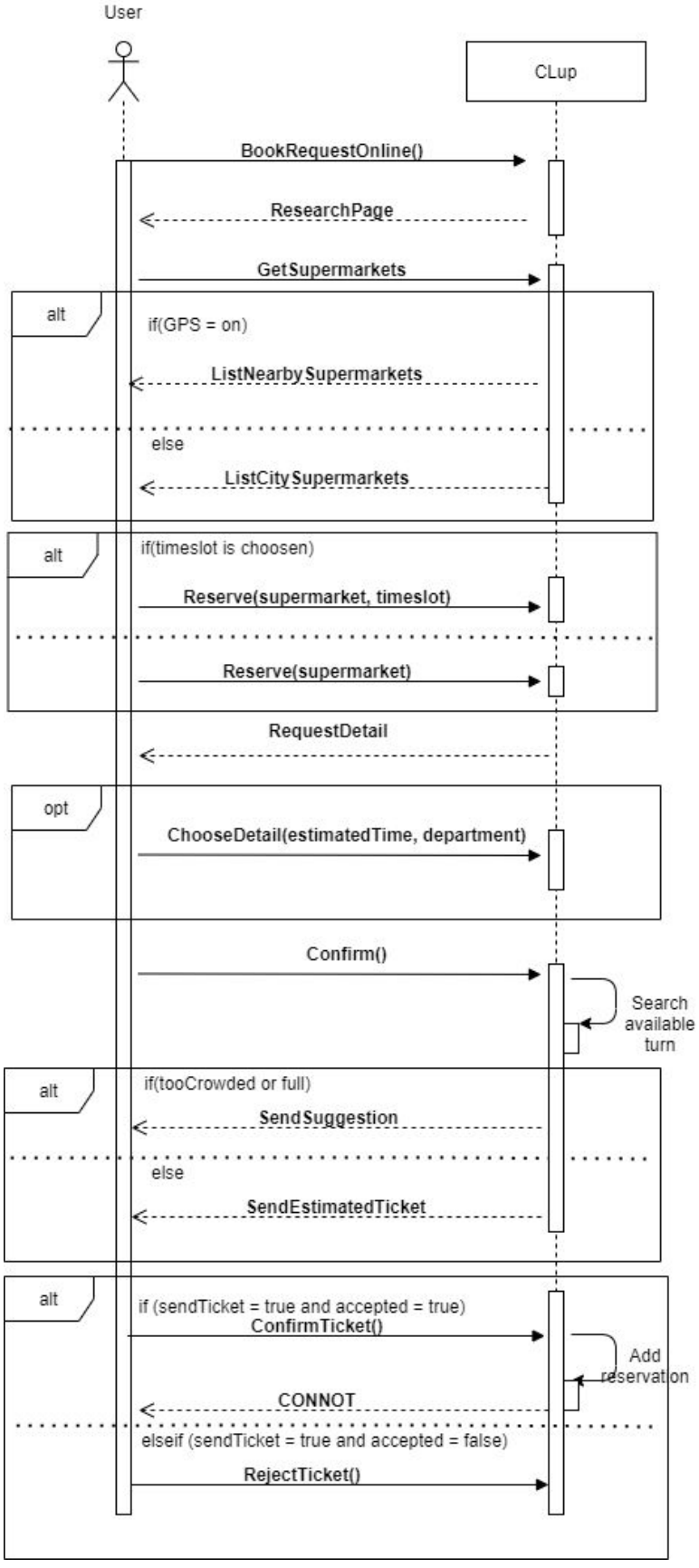

Image 7: Use Case 3/4

5. Book offline an ASAP turn

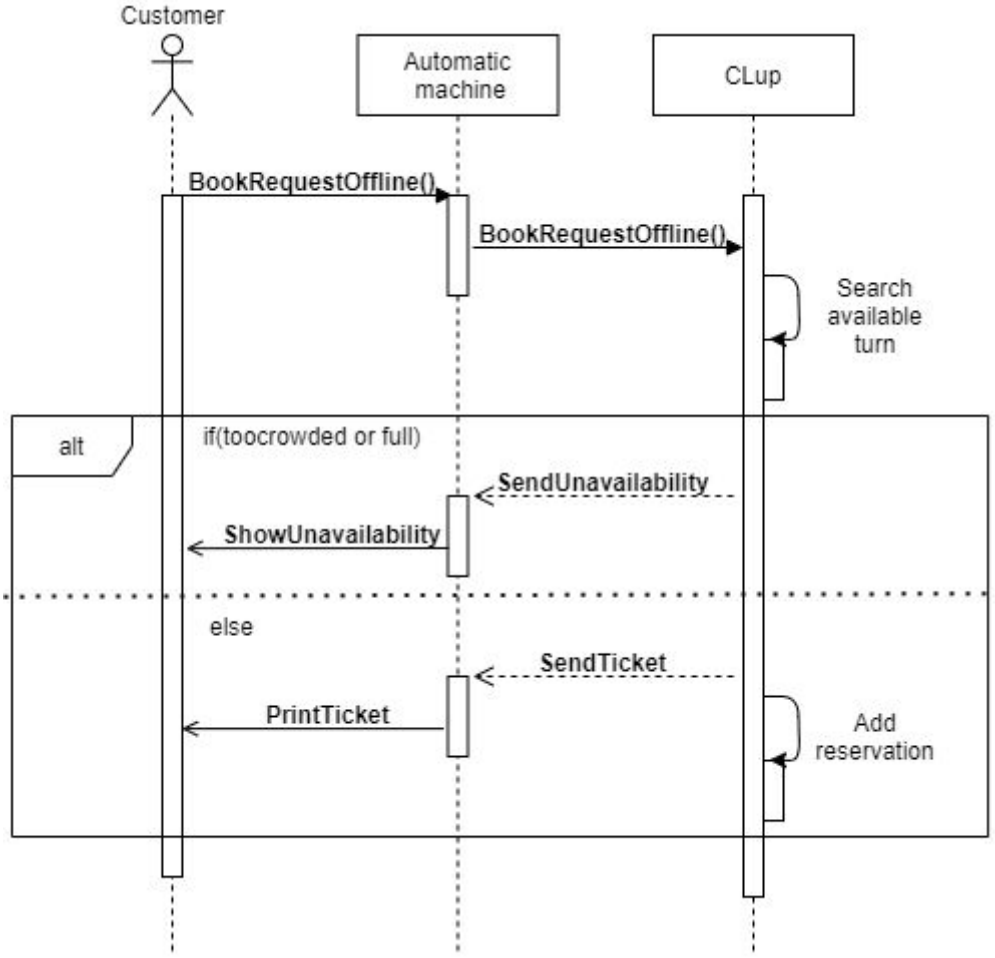

Image 8: Use Case 5

6. Shop in the supermarket

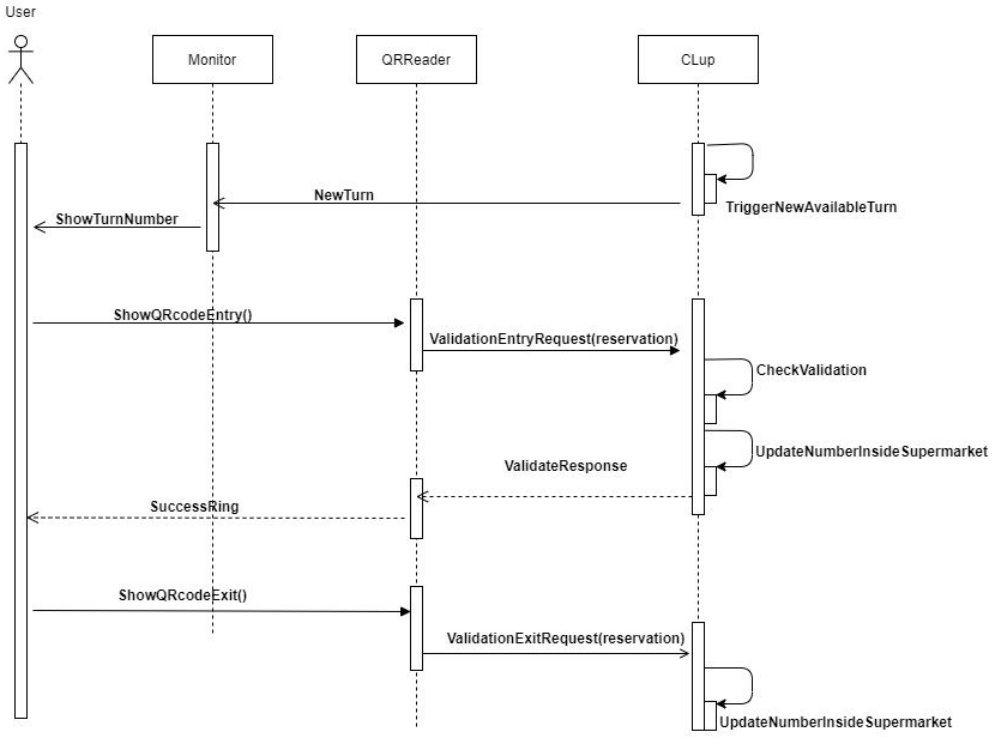

Image 9: Use Case 6

7. Cancellation of a reservation

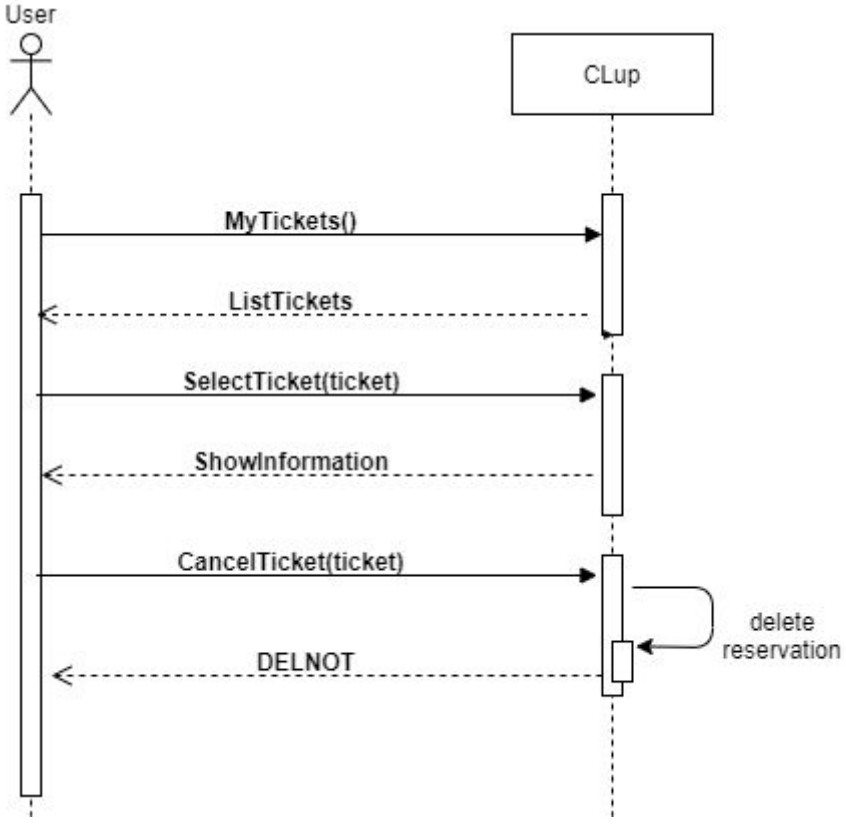

Image 10: Use Case 7

8. Setting the periodic notification

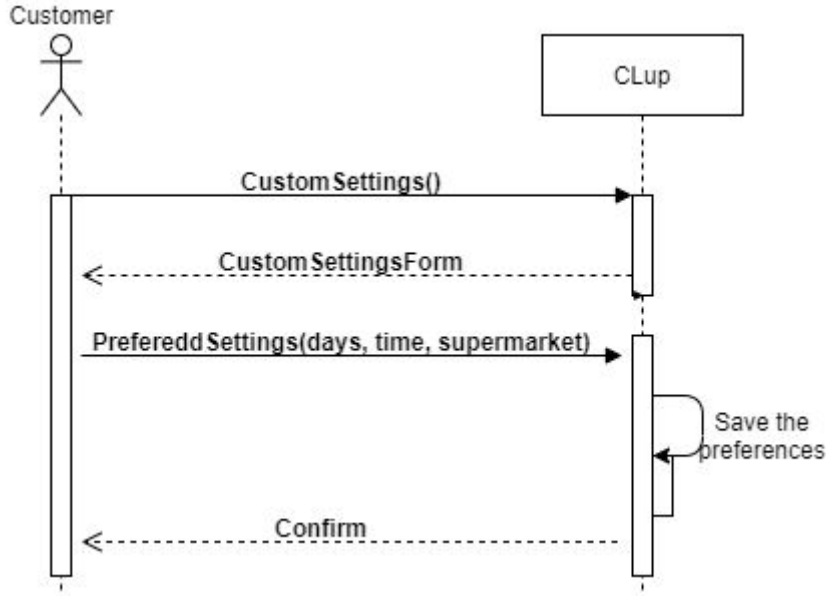

Image 11: Use Case 8

9. Modification of the influx

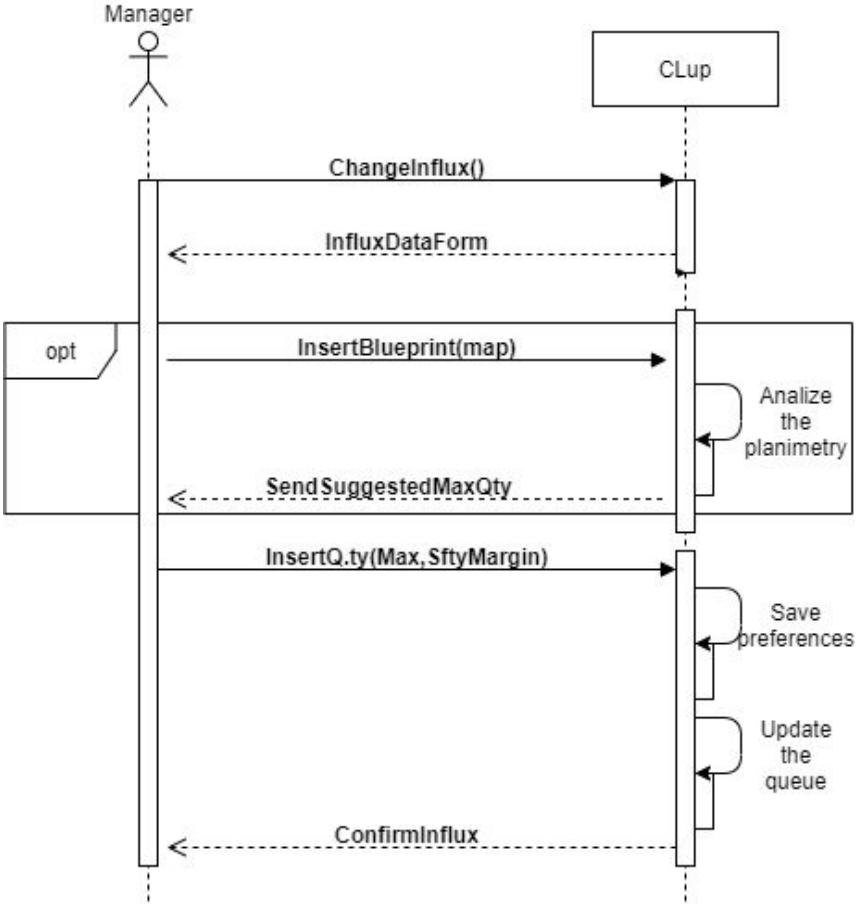

Image 12: Use Case 9

10. Build statistics

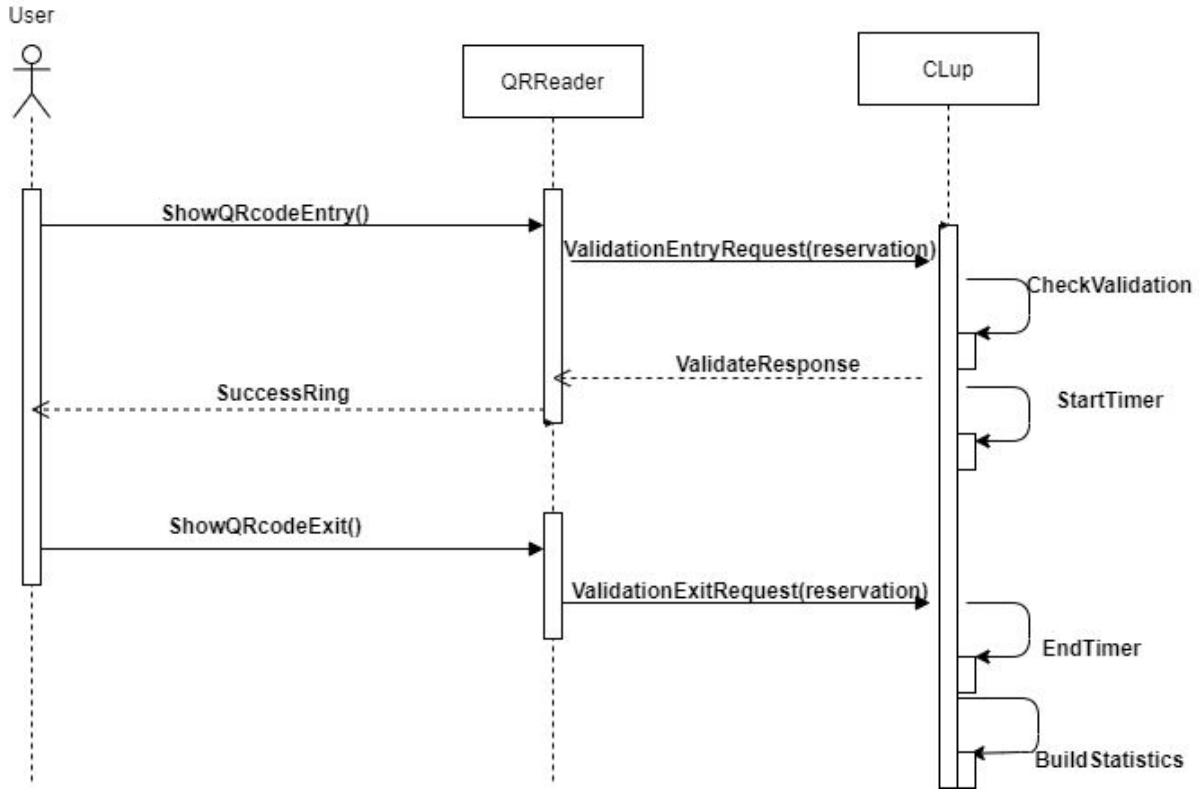

Image 13: Use Case 10

11. Receive an approaching notification

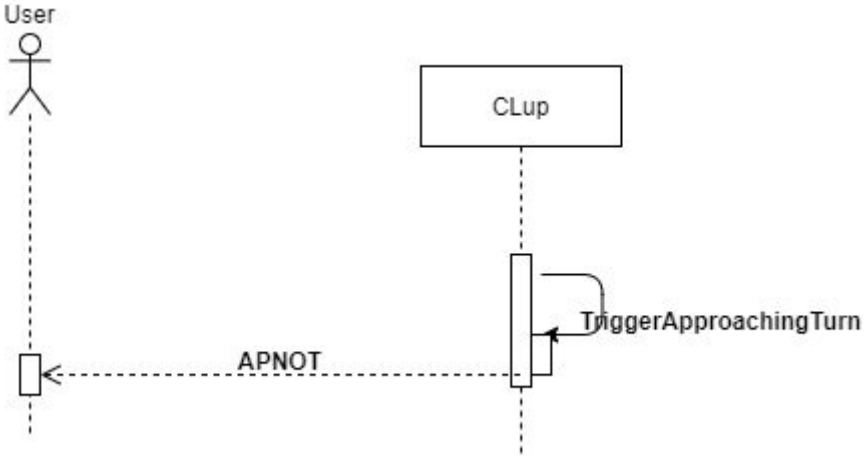

Image 14: Use Case 11

3.B.4 Mapping on requirements

GOALS	DOMAIN ASSUMPTIONS	REQUIREMENTS
G1	D1,D3, D5,D6, D8, D9, D10	R1, R5 ,R7, R8, R10, R11, R15, R16, R17, R18, R19, R21, R24, R25, R26, R27, R29, R30
G2	D4, D5, D6, D7, D8	R1, R2, R6, R7, R8, R9, R12, R13, R14, R15, R16, R17, R18, R21, R22, R23, R24, R25, R29
G3	D4, D10	R11, R12, R19, R26, R27, R29, R30, R31
G4	D1, D2, D4, D5, D7	R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R12, R13, R14, R15, R19, R20, R21, R22, R23, R25, R27, R28
G5	D1, D4	R1, R2, R3, R11, R12, R13, R14, R15, R15, R17, R19, R20, R26, R27, R30,

G1. Allow store managers to regulate the influx of customers inside the supermarket

D1. Each online user has a smartphone and a working internet connection during the entrance and during the payment in the supermarket
D3. Each smartphone has a display capable of showing a clear qr-code to be scanned
D5. Every partner store has a QR-code scanner at the entry and one at each cash register (automatic or with personnel)
D6. Each partner store has a display to show the current and incoming turn numbers, outside of the store
D8. The QR-code readers in the store work properly.
D9. Only online users and offline users enter the store

R1. The system should allow the user to register
R5. The system should provide users with a reasonably precise estimation of the waiting time.
R7. The system should generate a unique QR for each reservation.
R8. The system should generate a unique line number for each reservation.
R10. The system should be able to notify the user about the approaching of his turn based on his geographical position(if his GPS is on) and his turn in the line.(APNOT)
R11. The system should be able to maintain a count on the people inside the supermarket.
R15. The system should allow the user to indicate the approximate expected duration of the visit at the supermarket.
R16. The system should be able to assign users a default time of 20 minutes in case he doesn't choose the expected duration of the visit.
R17. The system should be able to estimate a duration time for a frequent customer based on his previous visits.
R18. The system should be able to make statistics based on the daily reservations and the time spent inside the supermarkets by the customers.
R19. The system should allow users to indicate the departments of items that they intend to visit to allow a finer management of the influx of customers.
R21. The system should cancel a reservation if the user doesn't scan the QR code at the entrance after 15 minutes from the set time and notify him.(DELNOT)
R24. After scanning the QR-code, the system should notify the user if he makes a mistake(WRONOT).
R25. The system should be able to manage bookings from different sources (smartphone, outsidex box).
R26. The system should allow the store manager to choose the maximum number of customers that can enter the supermarket.
R27. The system should not allow customers to book a turn in case the chosen time slot has reached the established threshold.
R29. The system should invalidate any reservation which QRcode isn't scanned after 15 minutes of his turn.
R30. The system should give suggestions to the store manager regarding the feasible max amount of customers inside the shop.

G2. Let people avoid to line up and stand outside of stores creating assemblage through the creation of a virtual line

D4. Each partner store has a working internet connection
D5. Every partner store has a QR-code scanner at the entry and one at each cash register (automatic or with personnel)
D6. Each partner store has a display to show the current and incoming turn numbers, outside of the store
D7. Each partner store has a ticket machine for offline users outside the supermarket
D8. The QR-code readers in the store work properly.

R1. The system should allow the user to register
R2. The system should allow the user to retrieve the lost password.
R6. The system should allow users to confirm a reservation.
R7. The system should generate a unique QR for each reservation.
R8. The system should generate a unique line number for each reservation.
R9. The system should send a ticket to the user through a notification .(CONNOT)
R12. The system should notify the user if the chosen supermarket/date/time slot is too crowded and should give an alternative.
R13. The system should allow the user to choose the day for a booking to the supermarket.
R14. The system should allow the user to select the time slot for a booking to the supermarket.
R15. The system should allow the user to indicate the approximate expected duration of the visit at the supermarket.
R16. The system should be able to assign users a default time of 20 minutes in case he doesn't choose the expected duration of the visit.
R17. The system should be able to estimate a duration time for a frequent customer based on his previous visits.
R18. The system should be able to make statistics based on the daily reservations and the time spent inside the supermarkets by the customers.
R21. The system should cancel a reservation if the user doesn't scan the QR code at the entrance after 15 minutes from the set time and notify him.(DELNOT)
R22. The system should allow the user to cancel his reservation.
R23. The system should allow the user to modify his reservation.
R24. After scanning the QR-code, the system should notify the user if he makes a mistake(WRONOT).
R25. The system should be able to manage bookings from different sources (smartphone, outsidex box).
R29. The system should invalidate any reservation which QRcode isn't scanned after 15 minutes of his turn.

G3. Ensure social distancing between customers into different store department

D4. Each partner store has a working internet connection
D1. Each online user has a smartphone and a working internet connection during the entrance and during the payment in the supermarket

R11. The system should be able to maintain a count on the people inside the supermarket.
R12. The system should notify the user if the chosen supermarket/date/time slot is too crowded and should give an alternative.
R19. The system should allow users to indicate the departments of items that they intend to visit to allow a finer management of the influx of customers.
R26. The system should allow the store manager to choose the maximum number of customers that can enter the supermarket.
R27. The system should not allow customers to book a turn in case the chosen time slot has reached the established threshold.
R29. The system should invalidate any reservation which QRcode isn't scanned after 15 minutes of his turn.
R30. The system should give suggestions to the store manager regarding the feasible max amount of customers inside the shop.
R31. The system should allow store manager to manage the department of his supermarket

G4. Allow customers to book a reservation to do shopping in a store

D1. Each online user has a smartphone and a working internet connection during the entrance and during the payment in the supermarket
D2. Each offline user preserves his ticket
D4. Each partner store has a working internet connection
D5. Every partner store has a QR-code scanner at the entry and one at each cash register (automatic or with personnel)
D7. Each partner store has a ticket machine for offline users outside the supermarket

R1. The system should allow the user to register
R2. The system should allow the user to retrieve the lost password.
R3. The system should be able to show the nearby supermarkets based on the location of the user(if the GPS is on).
R4. The system should allow the user to choose the supermarket where he wants to shop.
R5. The system should provide users with a reasonably precise estimation of the waiting time.

R6. The system should allow users to confirm a reservation.
R7. The system should generate a unique QR for each reservation.
R8. The system should generate a unique line number for each reservation.
R9. The system should send a ticket to the user through a notification .(CONNOT)
R10. The system should be able to notify the user about the approaching of his turn based on his geographical position(if his GPS is on) and his turn in the line.(APNOT)
R12. The system should notify the user if the chosen supermarket/date/time slot is too crowded and should give an alternative.
R13. The system should allow the user to choose the day for a booking to the supermarket.
R14. The system should allow the user to select the time slot for a booking to the supermarket.
R15. The system should allow the user to indicate the approximate expected duration of the visit at the supermarket.
R19. The system should allow users to indicate the departments of items that they intend to visit to allow a finer management of the influx of customers.
R20. The system should notify users periodically if there are any available turns, in the nearby supermarkets, in a selected day/time range . (FREENOT)
R21. The system should cancel a reservation if the user doesn't scan the QR code at the entrance after 15 minutes from the set time and notify him.(DELNOT)
R22. The system should allow the user to cancel his reservation.
R23. The system should allow the user to modify his reservation.
R25. The system should be able to manage bookings from different sources (smartphone, outsidex box).
R27. The system should not allow customers to book a turn in case the chosen time slot has reached the established threshold.
R28. The system should allow people without device to reserve a ticket

G5. Distribute the flow of the customers in multiple stores avoiding over-crowd them

D1. Each online user has a smartphone and a working internet connection during the entrance and during the payment in the supermarket
D4. Each partner store has a working internet connection

R1. The system should allow the user to register
R2. The system should allow the user to retrieve the lost password.
R3. The system should be able to show the nearby supermarkets based on the location of the user(if the GPS is on).
R11. The system should be able to maintain a count on the people inside the supermarket.

R12. The system should notify the user if the chosen supermarket/date/time slot is too crowded and should give an alternative.
R13. The system should allow the user to choose the day for a booking to the supermarket.
R14. The system should allow the user to select the time slot for a booking to the supermarket.
R15. The system should allow the user to indicate the approximate expected duration of the visit at the supermarket.
R15. The system should allow the user to indicate the approximate expected duration of the visit at the supermarket.
R17. The system should be able to estimate a duration time for a frequent customer based on his previous visits.
R19. The system should allow users to indicate the departments of items that they intend to visit to allow a finer management of the influx of customers.
R20. The system should notify users periodically if there are any available turns, in the nearby supermarkets, in a selected day/time range . (FREENOT)
R26. The system should allow the store manager to choose the maximum number of customers that can enter the supermarket.
R27. The system should not allow customers to book a turn in case the chosen time slot has reached the established threshold.
R30. The system should give suggestions to the store manager regarding the feasible max amount of customers inside the shop.

3.C Performance Requirements

- The server system must calculate a suitable time for the user reservation based on input data (suggested time slot by user, supermarket opening time, current reservation with relative estimated stay time) in no more than 5 seconds.
- The server system will be developed in order to handle a concurrent data flow from users trying to reserve in a popular supermarket in a rush hour, specially in discount periods.

3.D Design constraints

3.D.1 Standards compliance

The developed code for this system should not only follow the requirements contained in this document but moreover all the variables should respect the standard nomenclature for the chosen framework. In addition, the code should be clearly commented and documented in order to allow further and future development of the system itself.

3.D.2 Hardware limitations

In order to work properly, the system requires some technological devices:

1. QR-reader at the entrance of the supermarket and at every cash machine
2. an automatic ticket distributor outside the supermarket
3. a display that shows the current turn

In addition, the affiliated supermarkets should have a working internet connection to allow the system to manage properly the influx of the customer in the queue.

In case of the online customers, the mobile devices used should have an internet connection and should be able to support the application.

3.D.3 Any other constraints

All selectable time slots have a duration of 1 H(i.e. 8:30 - 9:30, 9:30 - 10: 30).

3.E Software system attributes

3.E.1 Reliability

The system must be the most reliable possible.

To achieve high levels of reliability, the system will be replicated using a cluster of multiple servers.

To avoid data loss, the database is redundant and backupped. These requirements are met by using a Cloud Database.

3.E.2 Availability

The system should be available 24/7.

The average time between the occurrence of a fault and service recovery (MTTR or downtime) should be contained around 3.65 days per year.

The system should be able to manage a huge amount of data deriving from different users at the same time.

3.E.3 Security

The system should ask the user for permission for the use of his smartphone GPS and should use it only to estimate a time arrival and to show the nearby supermarkets.

Moreover, all the personal customer data should be preserved and protected on encrypted databases and in case of password recovery all the exchanged data between the user and system should be encrypted as well. In addition, the system should be protected against any attack that aims to steal the private information of both customers and supermarkets.

3.E.4 Maintainability

The system must guarantee a high level of maintainability.
In order to do that:

- Code (mobile, web, server) will be written with modern frameworks guaranteeing the design patterns and programming standards.
- Code must be written with a high level of abstractions without hard-code as well.
- Code must be written with a large use of comments that cover all aspects of code itself.
- Code must provide a testing routine that covers at least 70% of the entire code, excluding software interface.

3.E.5 Portability

The system should be able to run on different platforms and OSs like Windows, Linux and Mac. In addition, the system must support and be available for the mobile operating system such as iiOS and Android.

3.E.6 Scalability

Within a pandemic setup it is expected that data, i.e., of new online customers, can grow exponentially in a very short amount of time. CLup needs to be able to cope with this exponential growth in order to be useful, when it is needed the most. One of the scalability dimensions is determined in terms of regions covered. Without any doubt, achieving an application suitable also across multiple countries will be inherently more useful.

4. FORMAL ANALYSIS WITH ALLOY

4.1 Alloy Code

```
// Signatures

abstract sig User {}

sig Customer extends User {

	ownReservations: set Reservation

}

sig Manager extends User {}{
	// a Manager can manage just a Store
	one this.~relatedManagers
}

sig Store {
	relatedQueue: Queue,
	relatedManagers: some Manager,
	relatedDepartments: set Department,

	maxCustomerCount: Int,
	safetyMargin: Int,

	setQrCodeReaders: some QRCodeReader,
	setStoreDisplays: some StoreDisplay,
	setAutomaticTicketMachines: some AutomaticTicketMachine

}{
	maxCustomerCount > 0
	safetyMargin > 0
	// Safety margin can't be greater or equal than the allowed
customer count
	safetyMargin < maxCustomerCount
}


sig Queue {

	includedReservations: set Reservation

}{
	// A Queue must belong to just a Store
```

```
      one this.~relatedQueue
}

abstract sig ReservationType{}
one sig ONLINE extends ReservationType{}
one sig OFFLINE extends ReservationType{}

sig Reservation {

      relatedQrCode: QrCode,
      type: ReservationType,
      date: one Date,
      time: one Time,
      requestedTimeSlot: lone TimeSlot,
      referredDepartments: set Department,
     relatedNotifications: set Notification
}{

      // A reservation must belong to just a Customer
      one this.~ownReservations

      // A reservation must be in one and just one queue
      one this.~includedReservations

      // if the reservation refers to specific departments
      // the chosen ones must be departments of the store
     // that owns the relative queue
#referredDepartments > 0 implies all d:referredDepartments | d in
((this.~includedReservations).~relatedQueue).relatedDepartments


}

sig QrCode{
      number: Int
}{
      // QrCode Number is non-negative
      number > 0

      // A QrCode is associated to one and only one Reservation
      one this.~relatedQrCode
}
```

```
sig Department {}{
      // A department belongs only to a specific Store
      one this.~relatedDepartments
}


abstract sig StoreHardware {}{}

sig QRCodeReader extends StoreHardware{
} {
      // a QR Code Reader can be installed in a single Store
      one this.~setQrCodeReaders
}

sig StoreDisplay extends StoreHardware{
} {
      // an Store DisplayMachine can be installed in a single Store
      one this.~setStoreDisplays
}

sig AutomaticTicketMachine extends StoreHardware{
}{
      // a  Automatic Ticket can be installed in a single Store
      one this.~setAutomaticTicketMachines
}

sig Date{}
sig Time{}
sig TimeSlot{}

abstract sig NotificationType{}
one sig CONNOT extends NotificationType{}
one sig DELNOT extends NotificationType{}
one sig WRONOT extends NotificationType{}
one sig APNOT extends NotificationType{}
one sig FREENOT extends NotificationType{}


sig Notification{
      type: NotificationType
}{
      // A notification si related to one and only one reservation
      one this.~relatedNotifications
}
```

// Facts

```
// id of reservations for QR-code generator are unique
fact{
	no disj qr1, qr2: QrCode | qr1.number = qr2.number
}

// Only Online Reservation has related dispatched notifications
fact {
	all r: Reservation | #r.relatedNotifications > 0 implies
r.type = ONLINE
}

// Only Online Reservation has, optionally, referred department
chosen by user
fact {
	all r: Reservation | #r.referredDepartments > 0 implies
r.type = ONLINE
}

// Only Online Reservation has, optionally, a time slot chosen by
user
fact {
	all r: Reservation | #r.requestedTimeSlot = 1 implies r.type
= ONLINE
}
```

// Assertions

```
// only online reservation has referred departments
assert onlyOnlineReservationHasReferredDepartments {
	no r: Reservation | r.type = OFFLINE &&
#r.referredDepartments > 0
}
check onlyOnlineReservationHasReferredDepartments


// only online reservation has requested time slot
assert onlyOnlineReservationHasRequestedTimeSlot {
	no r: Reservation | r.type = OFFLINE && #r.requestedTimeSlot
> 0
}
```

```
check onlyOnlineReservationHasRequestedTimeSlot


// only online reservation has notifications
assert onlyOnlineReservationHasNotifications {
      no     r:     Reservation     |     r.type     =     OFFLINE     &&
#r.relatedNotifications > 0
}
check onlyOnlineReservationHasNotifications
```

// Predicates

```
pred  onlineUserDoesShopping(r:  Reservation,  s:  Store,  q:  Queue,
m:Manager, c: Customer){
      r.type = ONLINE
      s.relatedQueue = q
      s.relatedManagers = m
      r in c.ownReservations
      #Reservation > 2
      #Department > 0
      #Time > 2

}
run onlineUserDoesShopping


pred  onlineUserDoesShoppingChosingDepartments(r:  Reservation,  s:
Store, q: Queue, m:Manager, c: Customer){
      r.type = ONLINE
      s.relatedQueue = q
      s.relatedManagers = m
      r in c.ownReservations
      #Department > 0
}
run onlineUserDoesShoppingChosingDepartments


pred  offlineUserDoesShopping(s:  Store,  q:  Queue,  m:Manager,  c:
Customer){
      s.relatedQueue = q
      s.relatedManagers = m
      all r: Reservation | r.type = OFFLINE
      all r: Reservation | r in c.ownReservations

     // we enforce the existence of some departments just to show
that they will not be related at any offline reservation
```

```
      #Department > 0
}
run offlineUserDoesShopping
```

Full code available here (access required):

https://github.com/giovanniciriello/AmatoCiriielloColombrino/blob/main/AlloyModels.als

4.2 Metamodels

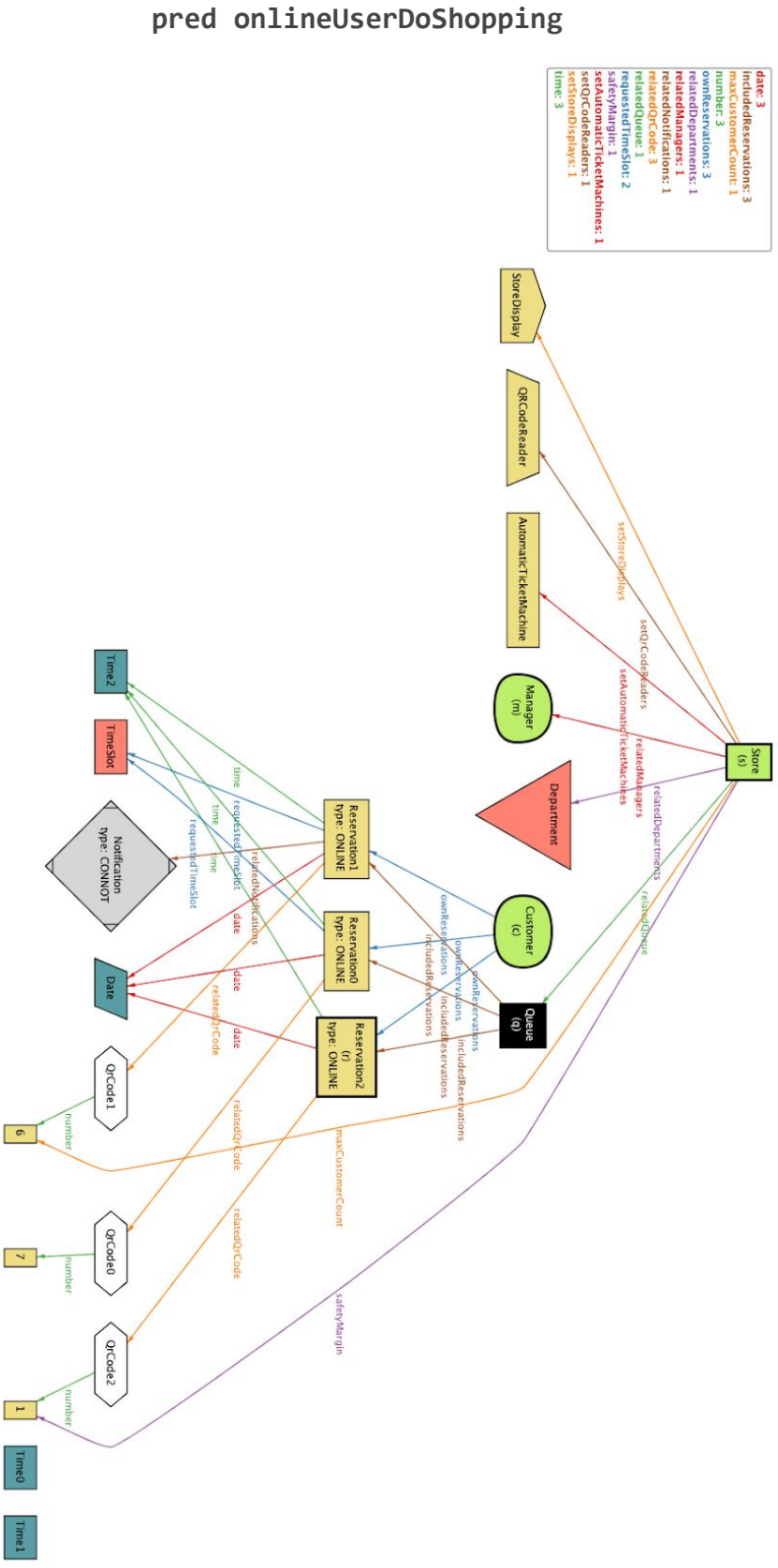

Image 15: MetaModel 1

```
pred onlineUserDoesShoppingChosingDepartments
```

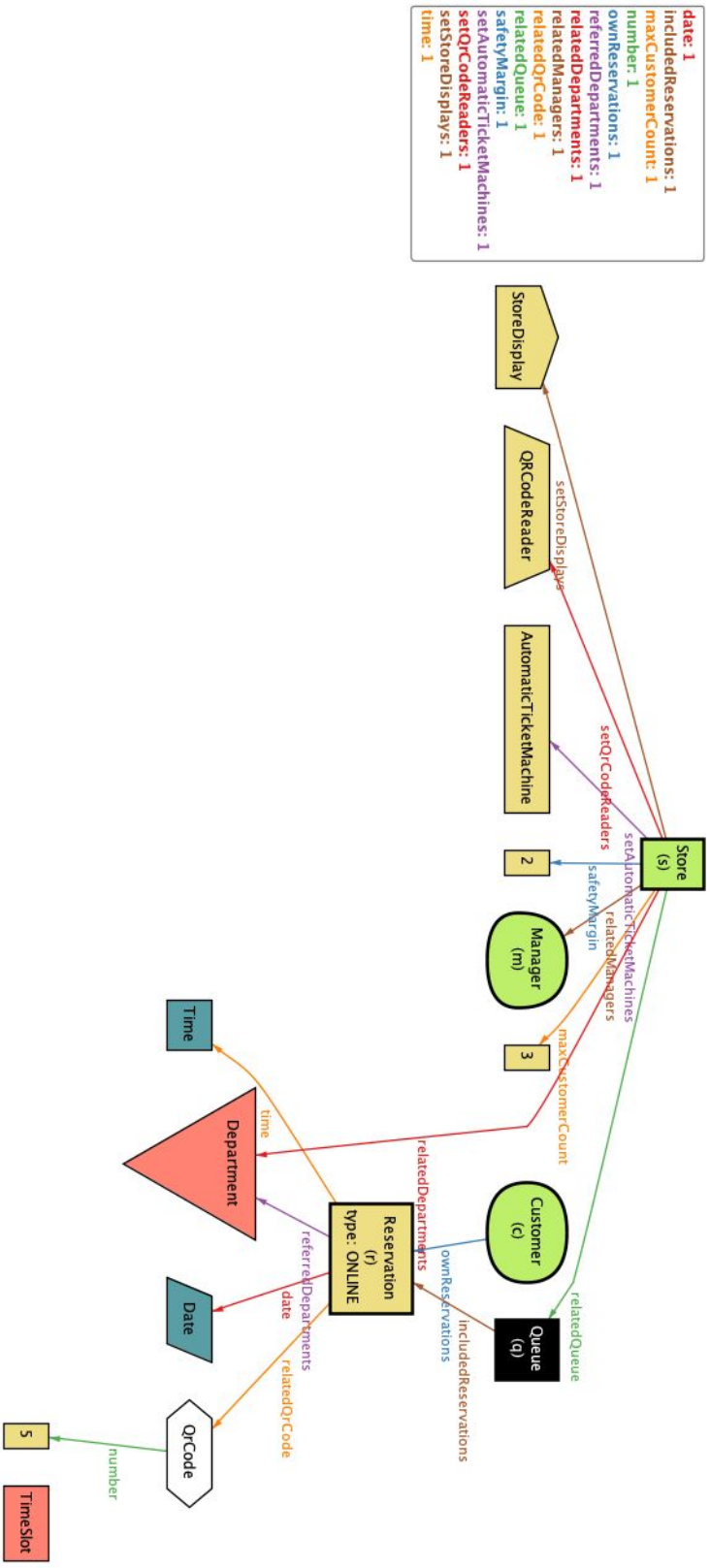

Image 16: Metamodel 2

```
pred offlineUserDoesShopping
```

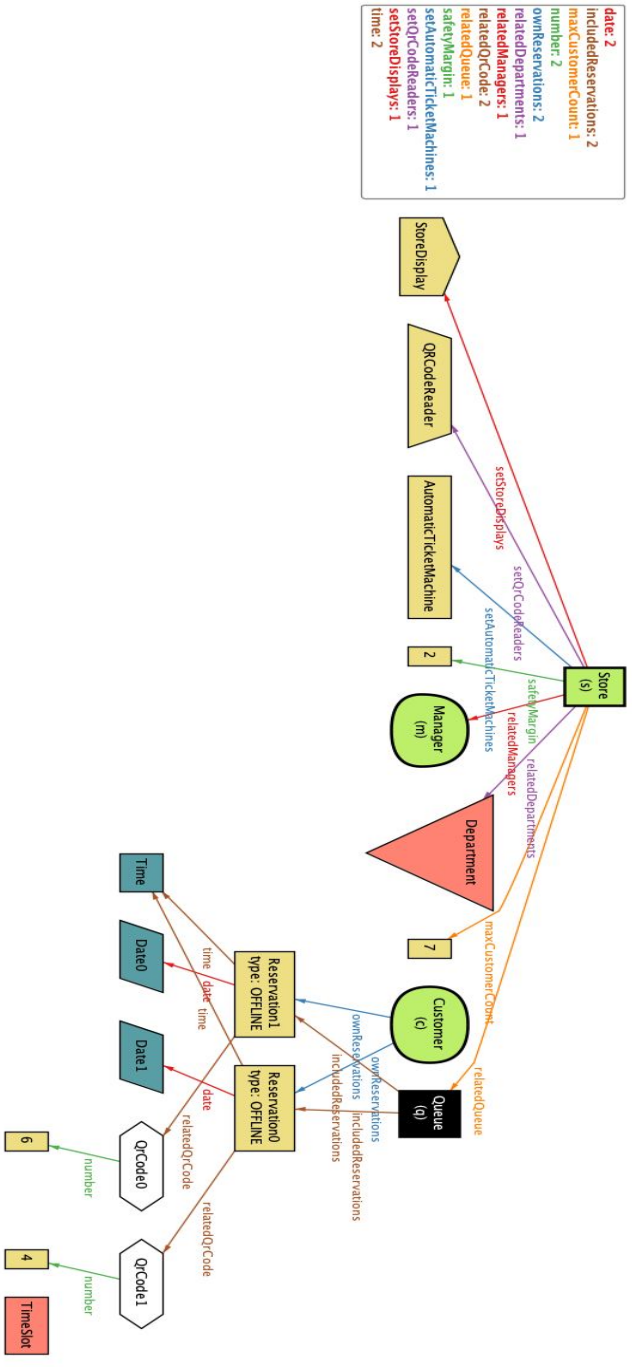

Image 17: Metamodel 3

4.3 Assertions' results

```
Executing "Check onlyOnlineReservationHasReferredDepartments"
   Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
   3759 vars. 336 primary vars. 9037 clauses. 10ms.
   No counterexample found. Assertion may be valid. 2ms.

Executing "Check onlyOnlineReservationHasRequestedTimeSlot"
   Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
   3759 vars. 336 primary vars. 9037 clauses. 9ms.
   No counterexample found. Assertion may be valid. 1ms.

Executing "Check onlyOnlineReservationHasNotifications"
   Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
   3759 vars. 336 primary vars. 9037 clauses. 10ms.
   No counterexample found. Assertion may be valid. 1ms.
```

4.4 Predicates' results

```
Executing "Run onlineUserDoShopping"
   Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
   3912 vars. 348 primary vars. 9319 clauses. 8ms.
   Instance found. Predicate is consistent. 11ms.

Executing "Run onlineUserDoesShoppingChosingDepartments"
   Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
   3888 vars. 348 primary vars. 9261 clauses. 7ms.
   Instance found. Predicate is consistent. 9ms.

Executing "Run offlineUserDoesShopping"
   Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
   3890 vars. 345 primary vars. 9269 clauses. 9ms.
   Instance found. Predicate is consistent. 9ms.
```

5. EFFORT SPENT

Amato Nunzio

Topic	Hours
Purpose and Scope	2 h
Domain assumption	2 h
Product functions	2 h
World and Shared Phenomena	1 h
Requirements	3 h
Scenario	3 h
Mapping(goal, domain assumption, requirement)	2 h
Use cases	3 h
Sequence diagram	2 h
UML and state diagrams	4 h
Alloy code	7 h

Ciriello Giovanni

Topic	Hours
Purpose and Scope	3 h
Domain assumption	2 h
Product functions	1 h
World and Shared Phenomena	2 h
Requirements	2 h

Scenario	3 h
Mapping(goal, domain assumption, requirement)	2 h
Use cases	1 h
Sequence diagram	4 h
UML and state diagrams	2 h
Alloy code	10 h

Colombrino Fulvio

Topic	**Hours**
Purpose and Scope	2 h
Domain assumption	2 h
Product functions	4 h
World and Shared Phenomena	2 h
Requirements	2 h
Scenario	4 h
Mapping(goal, domain assumption, requirement)	3 h
Use cases	3 h
Sequence diagram	3 h
UML and state diagrams	4 h
Alloy code	5 h

6. REFERENCES

- All the diagrams have been made with https://app.diagrams.net/
- Alloy code development has been supported by https://alloytools.org/documentation.html
- Wikipedia https://www.wikipedia.org/
- Software engineering, Global edition, Ian Sommerville

Image Index

Changelog

- 26/12/2020
    - Changed 3.4 Sequence Diagram Title
- 10/01/2021
    - Added departments-store fact to Alloy code