Politecnico di Milano
AA 2020 – 2021
Computer Science and Engineering

Software Engineering 2 - Project

RASD

Requirement Analysis and Specification Document

Claudio Alfredo Emanuele – 963293
Antonio Guadagno – 966763

1. Introduction

1.1. Purpose

The present document is the Requirement Analysis and Specification Document (RASD) of "Clup", a software designed to assist store managers and customers during the coronavirus emergency.
The document focuses on the description of the system goals, it contains the list of both functional and non-functional requirements needed to ensure a proper operability of the software itself and presents the accurate description of the application domain through the use of formal models such as Alloy and UML.
In this document, use cases will be formally described and different scenarios will explain how the system faces the challenge of allowing grocery shopping while avoiding overcrowding and long queue in front of the stores.
This document is addressed to the developers who have to realize the software implementing a list of requirements and specifications.

1.2. Scope

Due to the coronavirus emergency, many countries have imposed several restrictions which allow people to leave their houses only to satisfy essential needs. Even in cases in which people are justified in going out, they are requested to respect strict rules such as to keep a distance of at least one meter from person to person and to respect limitations in the number of people who can contemporarily access a building.

Grocery shopping is considered one of the essential activities that authorize people to leave their houses. The problem is that grocery stores can register a high affluence of people and so store managers have to deal with the restrictions concerning the limited number of customers that can contemporarily access the building.

Another problem is that the respect of the restrictions concerning the limited number of people inside a building can result in long queues outside the store: people could have to wait for several time before their turn to enter the store and so the problem of overcrowding is not solved, it's just that people don't crowd inside the store but outside it.

The goal of this project is to design and develop a software that allows store managers to regulate the influx of people inside the stores and to save people from lining up outside of stores for hours on end, avoiding also overcrowding outside of the groceries.

To do that, the system should provide different features such as:

- **Lining up from home**: customers can get a ticket from home using their smartphones. In this way the problem of long queue outside the store is solved because people can wait from home. Then, customers will receive a notification when their turn is close to come so that they can head towards the store in time.

- **Book a visit to the store**: customers can book a visit in advance using their smartphones and specifying the date and the time in which they intend to visit the grocery store.
Customers can also specify the name of the street and the civic number of the place where they will be located just before the moment of the visit so that they will be able to choose a grocery store that is near the place where they effectively will be (that can of course be different from the place where they are at the moment of the booking). Furthermore, if customers want, they can also specify the categories of items they intend to buy. This is useful because it can allow more people to contemporarily access the store: if the system knows that different people are going to buy different things, it means that they will occupy different spaces of the grocery store and so it's possible to manage the affluence of a larger number of people still respecting social distance rules.

The application should be easy to use because the application domain of grocery stores involves people from all ages and from all social categories. It is also to be considered that there are people who do not have the possibility to access the required technology and so that the system should allow them to benefit of the service in a different way, for example, stores should also offer the possibility to hand out tickets on the spot (using ticket distributors), thus acting as proxies for customers.

1.2.1. World Only Phenomena

WP1	The user has a smartphone with himself/herself
WP2	The user reaches the grocery store
WP3	The user lines up outside the grocery store
WP4	The user enters the grocery store
WP5	The user does grocery shopping
WP6	The user leaves the grocery store

1.2.2. Shared Phenomena

SP1	The System receives a request for a ticket *[Controlled by the world]*
SP2	The System issues a ticket *[Controlled by the machine]*
SP3	The System receives a booking for a visit *[Controlled by the world]*
SP4	The customer scans the QR code when enters the store *[Controlled by the world]*
SP5	The customer scans the QR code when leaves the store *[Controlled by the world]*

1.2.3. Goals

G1	Avoid overcrowding both inside and outside (queues) grocery stores
G2	Allow customers to do the grocery shopping safely
G2.1	Allow customers to get a ticket to enter the grocery store without physically approaching the store
G2.2	Allow customers to book a visit to enter the grocery store without physically approaching the store
G2.3	Allow customers to line up from home and to physically approach the store just when their turn is close to come
G2.4	Allow customers to not lose their turn even if they are not physically in queue in front of the store
G2.5	Allow customers to be informed about the queue situation in real time even if they are lining up from home
G3	Allow Store managers to respect the restrictions imposed and to make grocery stores safer for customers
G3.1	Allow Store managers to limit the number of people contemporarily visiting the store
G3.2	Allow Store managers to limit the number of people contemporarily waiting in queue in front of the store
G3.3	Allow Store managers to not lose customers because of the restrictions on the limited number of people that can contemporarily line up in front of the store and access it

1.3. Definitions, Acronyms, Abbreviations

1.3.1. Definitions
- **Registered Customer:** a Customer who has already installed the application "Clup" on his/her smartphone and is already registered to the service.
- **Unregistered Customer:** a Customer who does not have access to the required technology or who does not want to sign up to the service.
- **Store manager:** manager of a grocery store who signs up to the service.

1.3.2. Acronyms
- **RASD**: Requirement Analysis and Specification Document.
- **GPS**: Global Positioning System

1.3.3. Abbreviations
- **Gn**: goal number n
- **Dn**: domain assumption number n
- **Rn**: functional requirement number n

1.4. Revision History
16/12/2020 – First version of RASD
26/12/2020 – Second version of RASD (Requirement 28 added)

1.5. Reference Documents
- Clup specification document
- Slides on the course website

1.6. Document Structure
- *Chapter 1*: in this chapter, the problem to be solved is identified and a solution is provided. Then, also the goals of the software, the domain application and the World and Shared Phenomena are presented and explained.
- *Chapter 2*: the interfaces and the boundaries between the system and the external environment are described. We also clarify who are the actors interacting with the software (humans, software, hardware) using state charts and class diagrams. In the product function section, the most important requirements that the software must offer and the domain assumptions are listed.
- *Chapter 3*: contains the details and the core of the RASD. We describe user interfaces, hardware interfaces, software interfaces and communication interfaces. In the functional requirements section, we define use case diagrams, use cases, the associated sequence diagrams and mapping on requirements. Finally, there is a description of possible scenarios and the description of non-functional requirements such as performance requirements, reliability, availability, security, maintainability and portability
- *Chapter 4*: includes the alloy code and the corresponding metamodels generated from it
- *Chapter 5*: shows the effort spent for each member of the group.
- *Chapter 6*: includes the reference documents.

2. Overall Description

2.1. Product perspective

2.1.1. UML description

The following UML Diagram contains the most relevant elements of the Clup system and describes the relationships between them.

The System offers to Registered Customers the possibility to get a ticket to visit a grocery store either using the ticket distributor in front of the store or online, using their smartphones.

Unregistered Customers can only get tickets using the ticket distributors that are in front of the stores.

A ticket is associated to a specific grocery store but many tickets can be associated to the same grocery's.

Each ticket is also associated to a unique status: the ticket is "Issued" if the Registered Customer has got the ticket but has not yet entered the store, is in the "Entrance Scanned" status if the customer has entered the store scanning the QR code associated to the ticket and is still inside the store, is in the "Exit Scanned" status if the customer has finished the grocery shopping, so has scanned again the ticket in order to leave the store and then is in the "Expired" state if the customer has lost his/her turn or has cancelled the ticket request.

Each ticket is associated to an unique QR Code and this QR Code is scanned, using the QR Reader of the store, by customers, both when they enter and when they leave the store.

Each ticket is associated to zero or one Registered Customer because also Unregistered Customers can get a ticket using a ticket distributor. Only Registered Customers can book a visit online using their smartphones and the booked visit will be associated to exactly one ticket (a ticket can be or not be associated to a visit).

Then, a Store Manager can run one or more grocery stores, each grocery store is associated to exactly one queue and owns exactly one ticket distributor and one QR Reader.
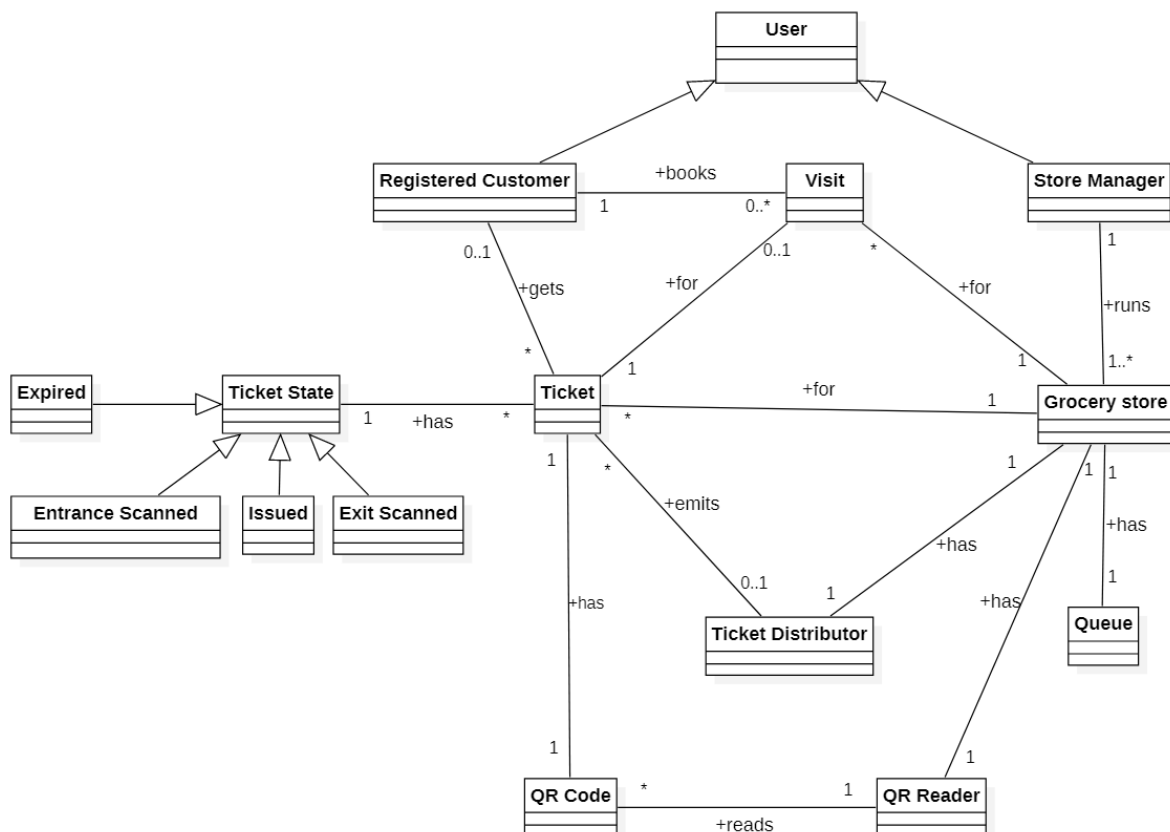
2.1.2. State Charts

In this paragraph, most important aspects of the System will be analysed by the use of state charts.

1) The following state diagram models the request of a ticket online made by a Registered Customer
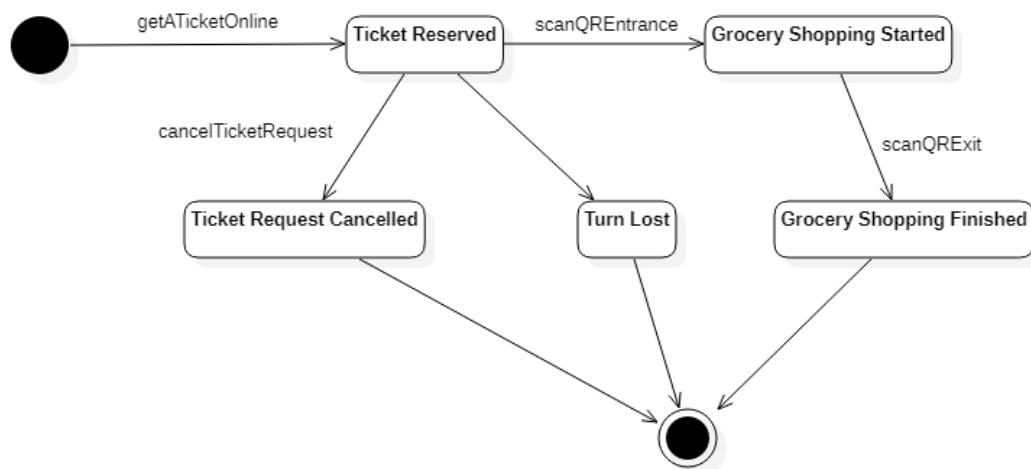
2) The following state diagram models the booking of a visit online made by a Registered Customer. The un-named arrow that links "Visit Booked" to the final state is used in cases in which a Registered Customer books a visit but then doesn't go to the grocery store on the day of the visit.
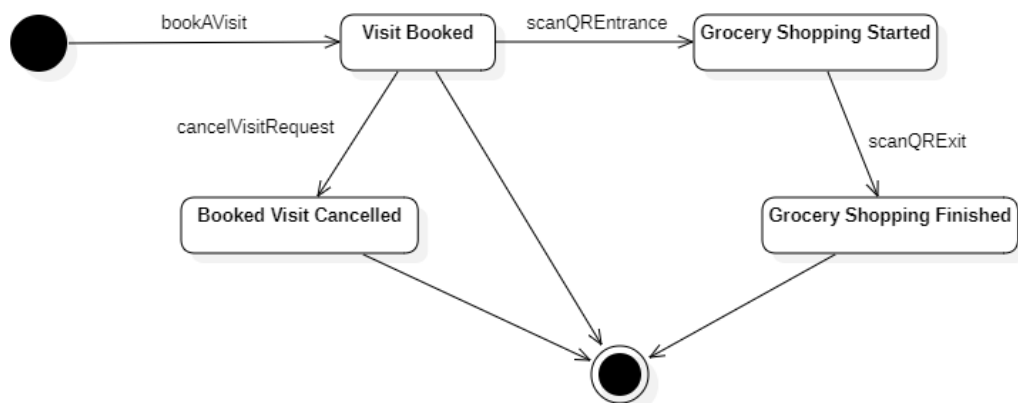
3) The following state diagram models the request of a ticket in presence
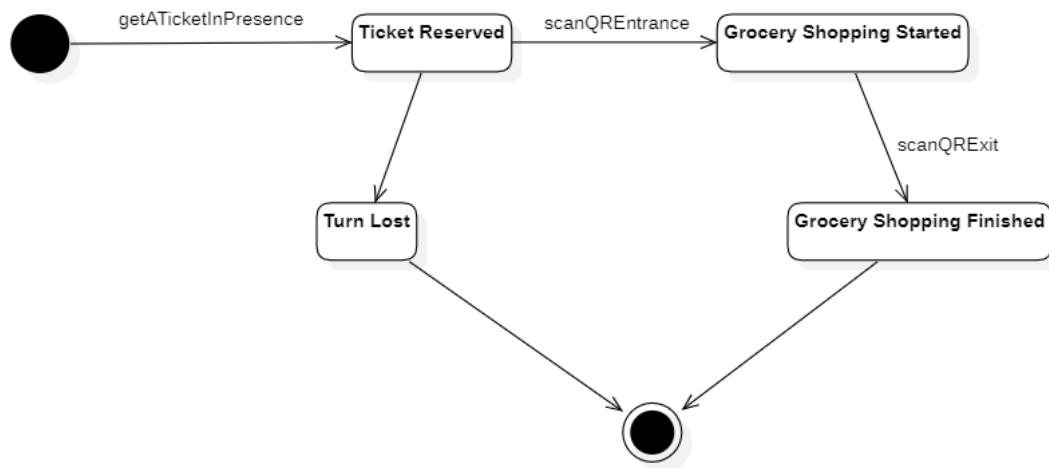
2.2. Product functions

Getting a ticket online

The main functionality that the System provides to customers is the possibility to use a smartphone to reserve a ticket online and so to line up from home. The main goals are to regulate the flux of people inside grocery's and to avoid overcrowding both inside and outside the stores: letting people to line up from home and sending them the notification to head to the store in the right moment, the System avoids overcrowding due to long queues and also manages to limit the number of people that are allowed to be contemporarily inside the building. Using GPS, the System finds grocery stores nearby the customer's current position, then the customer is asked to choose if he/she wants to reach the grocery store on foot, by car or with public transport. This information is used by the system, combined with the information about the estimated waiting time and the information about the traffic condition in real time, to send to the customer the notification to head to the store in the right moment: different modes of transport mean different amounts of time needed to reach the store. Then the customer is asked to choose the grocery store that he/she prefers and to get a ticket. To help the customer in his/her choice, the System shows, for each grocery's, the distance and the estimated waiting time in the queue. The System also provides information about the situation of the queue in real time so that the customer can line up from home without the risk to lose his/her turn. Customers can also choose to cancel the request for the ticket if they change their mind, but they cannot get multiple tickets at a time.

According to the limitations imposed by the Government, only grocery stores that are in the same municipality of the customer's current position at the moment of the reservation are showed to the user.

Anyway, only stores that are open at the moment of the reservation are selectable by the user. Stores that will close before that the Customer will be allowed to enter are not selectable. We decide to show to the user all the stores of the municipality, also unselectable once, in order to let the customer understand if there are stores that are registered to the system in the municipality where he/she currently is.

Book a visit online

Another important functionality that the System provides to Registered Customers is the possibility to book a visit to a grocery store in advance.

The customer enters data and time that he/she prefers for the visit and then the System asks for the address where the customer will be located just before the moment of the visit (that could be different from the current position of the user at the moment of the booking) in order to find all the grocery stores nearby. The System shows to the user that grocery stores that are in the same municipality of the inserted address and that have a bookable ticket for the selected date and time with a tolerance of 30 minutes (for instance, if the time selected is "19:00", the System also shows that grocery's that have bookable tickets from 18:30 to 19:30 if all the tickets bookable for the 19:00 are already finished). The customer chooses a grocery store, books a visit and then receives a confirmation from the System. The customer will receive a second notification sent by the System that reminds the booked visit: if the visit is scheduled for the 10:00 AM or earlier, the System sends the notification at 6:00 PM of the day before, otherwise the notification is sent 2 hours before the time of the visit.

If Customer's smartphone is turned off when the notification arrives, then he/she will see it when he/she turns on the phone. Customers can also select the kind of goods that they intend to buy. This is helpful to allow more people to contemporarily access the store: if the System knows that different people will be in different areas of the grocery's, it can to regulate the influx of a greater number of people still respecting social distance rules.

A Customer can book more than one visit at a time but not in the same date and the same time.

Store manager analytics

The WebApp offers to store managers the possibility to insert and eventually update important information about the store such as

1) the opening hours
2) the number of people that are allowed to contemporarily access the building (this number will depend on the dimensions of the store and on the restrictions imposed by the government)
3) the number of people that can contemporarily line up physically outside of the store (this number will depend also on the space available for the queue) *(*)*
4) the kind of goods sold by the grocery's (useful information to let people to specify what they intend to buy when they book a visit)

The WebApp also provides the possibility to analyse important information such as the number of people that enter the store day by day and the average time of the visit, both general and for a specific registered customer. Information about the average time of the visit is also used by the System to estimate the waiting time in the queue and so to send a notification to users that are lining up from home when it's time for them to head to the store. This information is collected by the system because customers are asked to scan a QR code both when they enter and when they leave the store.
Using the WebApp, a store manager can also see the visits booked to his/her grocery store.

(*) Note that our System cannot guarantee that the constraint on the number of people that can contemporarily line up physically outside of the store will be perfectly respected because of Unregistered Customers (that customers who don't have access to the required technology): they need to reach the store to get a ticket so our system cannot control their arrival to the grocery's via the notification system.
Our system can only act on Registered Customers (customers who get a ticket using their smartphones) sending them the notification to head to the store in the right moment.
Anyway, knowing the information about the tickets got online and the information about tickets got from ticket distributor, our System can estimate the number of Unregistered Customers that go to the store in a certain moment of a certain day, in order to regulate the sending of the notification to head to the store to that customers that are lining up from home, of course always respecting the order of the queue.
For these reasons, the number of people that can contemporarily line up physically outside of the store will be used just as a useful indication to regulate the sending of the notifications, but the system doesn't ensure the perfect respect of the constraint.

2.3. User characteristics

2.3.1. Actors

1) **Customer**
   There are 2 kinds of Customers:
   - **Registered Customers**: a Registered Customer is a customer who has already installed the application "Clup" on his/her smartphone and is already registered to the service.
     A Registered Customer can get tickets online and can book visits to grocery stores using his/her smartphone.
   - **Unregistered Customers**: an Unregistered Customer is a customer who does not have access to the required technology or a customer who does not want to sign up to the service.
     Unregistered Customers have to go physically to the store to get a ticket from the ticket distributor and cannot book visits.

2) **Store manager**
   A store manager is the manager of a grocery store who signs up to the service.
   A store manager signs up using the WebApp and providing all the information that are necessary to certify that he/she really runs an activity that is a grocery store (the list of documents that the store manager as to provide is described later in the present document).

3) **Ticket distributor**
   In front of each grocery store, there is a ticket distributor that provides tickets on the spot for customers who can't access the required technology or for customers who are not registered to the service.

4) **QR code reader**
   In front of each grocery store, there is a QR code reader used by Customers (both Registered and Unregistered) to scan the QR code present on their tickets both when they enter and when they leave the store.

2.4. Assumptions, dependencies and constraints

2.4.1. Domain assumptions

DA1	If a Registered Customer books a visit it's because he/she has already organized his/her commitments so is able to tell the System the address of the place where he/she will be just before the moment of the visit
DA2	When a Registered Customer provides to the System the information on how he/she wants to reach the store (on foot, by car, with public transport), then the customer really reaches the store using the selected method.
DA3	All the grocery stores have a ticket distributor in front of the store that is connected to the System and gives the possibility to get a ticket on the spot
DA4	All the grocery stores have a QR code reader in front of the store that is connected to the System and gives the possibility to scan QR codes present on tickets
DA5	Customers always really scans the QR code associated to the ticket both when they enter and when they leave the store
DA6	Information provided by Store managers about their grocery stores are always correct, up to date and in accordance with the laws issued by the government to face the emergency.
DA7	Internet connection works properly
DA8	Ticket distributors work properly
DA9	QR readers work properly
DA10	GPS provides the position with an error of 100 meters at most

3. Specific Requirements

3.1. External Interface Requirements

3.1.1. User Interfaces

If a customer wants to benefit of the service, his/her device must have a GPS sensor integrated.
The GPS sensor is used by Clup to estimate the time needed by a customer to reach the store from his/her current position and consequently to send the notification when it's time for the customer who has requested a ticket to head to the store in order to not lose his/her turn.
Usually, all modern smartphones have a GPS sensor integrated and so are the optimal devices to benefit of the service offered by Clup.

In the following, most important mockups are briefly presented.

Home page. This page is presented to the user when he/she opens the application.

Page in which a Registered Customer can choose to get a ticket or to book a visit

Page in which a Registered Customer provides information on how he/she wants to reach the store
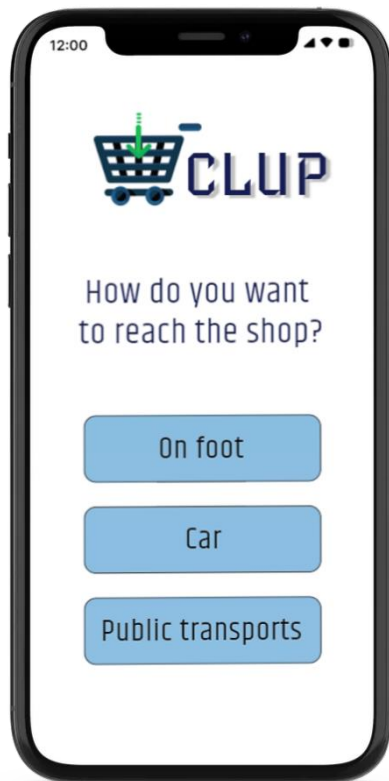
Page in which a Registered Customer selects the store in which he/she wants to do grocery shopping

12:00

CLUP

How do you want to reach the shop?

On foot

Car

Public transports

12:00

CLUP

COOP - Via Garibaldi 21
Distance: 0,5 km
Estimated waiting
time: 35 min

CONAD - Via Messina 35
Distance: 1,5 km
Estimated waiting
time: 18 min

PAM - Via Buonarroti 72
Distance: 2,5 km
Estimated waiting
time: 50 min
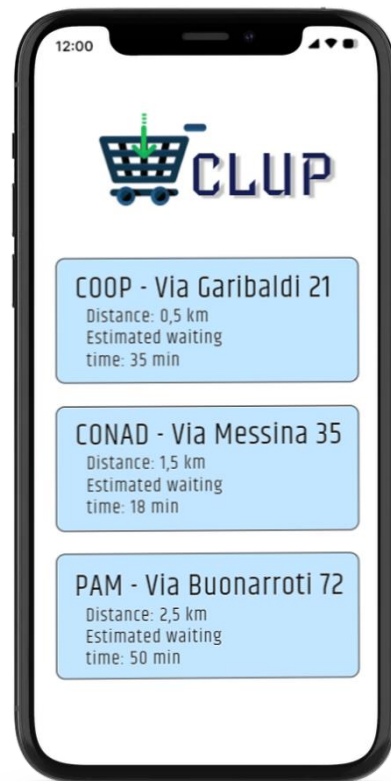
Page in which a Registered Customer can see information about his/her ticket and the estimated waiting time

12:00

CLUP

Your ticket:

COOP - Via Garibaldi 21

Estimated waiting
time: 35 min
Your ticket number is: 47
Number of people in queue
before you: 10

Distance: 0,5 km
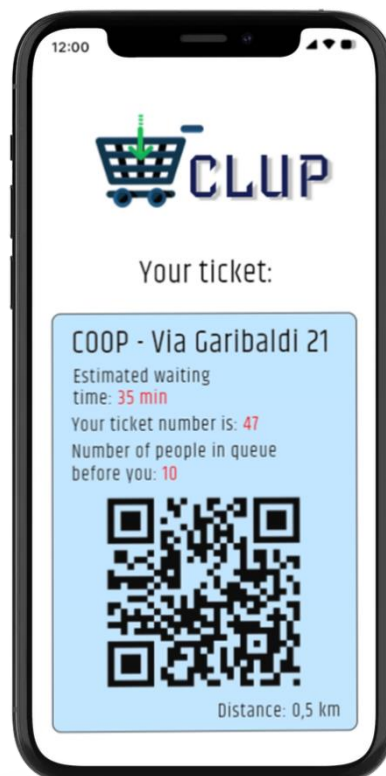
3.1.2. Hardware Interfaces

The System involves 3 types of human actors: Store managers, Registered customers and Unregistered customers. Each of them requires different kinds of hardware interfaces in order to benefit of services offered by the System.

Store managers need a computer in order to access the dedicated WebApp. Using the WebApp, store managers can modify information about the store (such as the opening hours, the kinds of goods that are available in the store, the number of people allowed contemporarily to access the building) but they can also analyze information such as the number of people that enter the store day by day, the booked visits and the average time of the visit (both general and for a specific registered customer).

Registered Customers need a smartphone with a GPS sensor integrated in order to access the mobile app. Using the mobile app, a Registered Customer can get a ticket or book a visit. Thanks to the mobile application, Registered Customers also receives notifications that inform them when it's time to head to the store or to remind them of a booked visit.

Unregistered Customers are, by definition, customers who don't have the required technology to benefit of the service or customers who don't want to sign up to the System. They don't need specific hardware interfaces, they just use ticket distributors present in front of grocery stores.

Both Registered and Unregistered Customers have to interact with a QR code reader when they enter and when they leave a grocery store.

3.1.3. Software Interfaces

To work properly, the System needs external software interfaces. In particular, the System needs Google Maps to find grocery stores near the customer current position and to estimate the time needed by the customer to reach the store (considering the mode of transport selected by the user - on foot, by car, with public transport - and the real-time traffic conditions).

3.1.4. Communication Interfaces

Computers and Smartphones are connected to the System via internet, QR code readers and Ticket distributors are connected to the grocery store computer via wireless connection.

3.2. Functional Requirements

3.2.1. List of Requirements

In the following table, "Customer" means "Both a Registered or an Unregistered Customer". Otherwise, the kind of Customer is explicitly specified.

R1	When a Customer tries to use the application Clup, the System controls if the device used has a GPS sensor integrated. If not, the Customer is notified that he/she cannot use the application because he/she doesn't have the required technology.
R2	The System shall ask to the Customer the permission to access the GPS position. If the Customer doesn't want to give the authorization, the System informs him/her that he/she can't use the service.
R3	A Registered Customer must be able to log in to the System using his/her credential
R4	A Registered Customer can use the system only if he/she is logged in
R5	A Registered Customer cannot take more than one ticket at a time
R6	A Registered Customer can book more than one visit at a time but not in the same date and the same time
R7	A Registered Customer can cancel the request of a ticket losing his/her turn and also the request of a booked visit
R8	A Registered Customer must be able to see the real time situation of the queue using his/her smartphone
R9	Everyone (both Registered and Unregistered Customers) must have the possibility to get a ticket, or using the dedicated app on the smartphone or using the ticket distributor that all the stores subscribed to the System must have
R10	When a Registered Customer wants to get a ticket online, the System asks him/her if he/she intends to reach the store by car, using public transports or on foot in order to better calculate the estimated time needed by the Registered Customer to reach the store
R11	When a Registered Customer wants to get a ticket online, the System asks him/her to choose from a list of stores that are currently open and located in the same municipality of the current Registered Customer position. Stores that will close before that the customer will be allowed to enter are not selectable.
R12	When a Customer gets a Ticket (both online and in presence), the System generates an unique QR code associated with the ticket
R13	When a Registered Customer books a visit, he/she can select the category of goods that he/she intends to buy
R14	When a Registered Customer books a visit, the System asks him/her to insert data and time of the visit and also the address of the place where the Registered Customer will be just before the moment of the visit in order to identify and then show on screen the list of stores having a bookable visit on the selected date and time and near the selected address

R15	When a Registered Customer gets a ticket using his/her smartphone, then the System sends him/her a notification 10 minutes before that the estimated waiting time is equal to the time needed by Registered Customer to reach the store in order to notify him/her that is time to head to the store
R16	When a Registered Customer books a visit, 2 hours before the time of the visit the System sends a notification to Registered Customer to remember him/her the booked visit. If the visit is scheduled for the 10:00 AM or earlier, the System sends the notification at 6:00 PM of the day before.
R17	If a Registered Customer booked a visit online and then his/her smartphone is turned off when the System sends the notification to remind the booked visit, then the Registered Customer will be able to see the notification when he/she turns on the device
R18	If a Customer is not in front of the store when it's his/her turn, the System waits for 2 minutes then Customer loses his/her turn and the turn passes to the next Customer in line
R19	In order to sign up, a Store manager has to prove that he/she they really runs a grocery shop sending documents related to the management of the store itself and licenses (required documents are later specified in the Use cases section)
R20	A registered Store manager must be able to log in to the System using his/her credentials
R21	A Store manager can use the system only if he/she is logged in
R22	Using the WebApp, a Store manager can select the number of customers allowed to be contemporarily in queue in front of the store
R23	Using the WebApp, Store managers can select the number of customers allowed to be contemporarily inside the store
R24	Using the WebApp, Store managers can see information about their grocery store such as the average time of the visit (both general and for a specific Registered Customer), the number of people that entered the store day by day and the booked visits
R25	The System is able to use GPS, information about traffic condition in real time and information about movement methods chosen by Registered Customer to calculate the estimated time needed by a Registered Customer to reach the store
R26	The System must provide to a Registered Customer the estimated waiting time for the line up in real time
R27	When a Registered Customer wants to get a ticket online, the System shows also stores that cannot be selected (because they are closed or they will close before that the customer will be allowed to enter) in order to let the Customer understander if there are stores registered to the system in the municipality where he/she currently is. Of course, unselectable stores cannot be selected and they are showed at the bottom of the list, after the once that are selectable (if exists).
R28	Customers should be able to scan the QR Code on their tickets both when they enter and when they leave the store. The system should be able to update queues and stores information with data taken from the scan of the QR Codes.

3.2.2. Mapping

(All requirements and all domain assumptions have been used)

Goals	Requirements	Domain assumptions
G1	R1, R2, R3, R4, R5, R6, R7, R8, R10, R12, R13, R15, R16, R17, R22, R23, R24, R25, R26	DA2, DA4, DA5, DA7, DA9, DA10
G2	R1, R2, R3, R4, R5, R6, R7, R8, R10, R11, R14, R15, R16, R17, R18, R25, R26, R28	DA1, DA2, DA3, DA7, DA8, DA10
G3	R5, R6, R7, R9, R12, R13, R15, R16, R17, R19, R20, R21, R22, R23, R24, R27, R28	DA3, DA6, DA8

G1	Avoid overcrowding both inside and outside (queues) grocery stores
R1	When a Customer tries to use the application Clup, the System controls if the device used has a GPS sensor integrated. If not, the Customer is notified that he/she cannot use the application because he/she doesn't have the required technology.
R2	The System shall ask to the Customer the permission to access the GPS position. If the Customer doesn't want to give the authorization, the System informs him/her that he/she can't use the service.
R3	A Registered Customer must be able to log in to the System using his/her credential
R4	A Registered Customer can use the system only if he/she is logged in
R5	A Registered Customer cannot take more than one ticket at a time
R6	A Registered Customer can book more than one visit at a time but not in the same date and the same time
R7	A Registered Customer can cancel the request of a ticket losing his/her turn and also the request of a booked visit
R8	A Registered Customer must be able to see the real time situation of the queue using his/her smartphone
R10	When a Registered Customer wants to get a ticket online, the System asks him/her if he/she intends to reach the store by car, using public transports or on foot in order to better calculate the estimated time needed by the Registered Customer to reach the store
R12	When a Customer gets a Ticket (both online and in presence), the System generates an unique QR code associated with the ticket
R13	When a Registered Customer books a visit, he/she can select the category of goods that he/she intends to buy
R15	When a Registered Customer gets a ticket using his/her smartphone, then the System sends him/her a notification 10 minutes before that the estimated waiting time is equal to the time needed by Registered Customer to reach the store in order to notify him/her that is time to head to the store
R16	When a Registered Customer books a visit, 2 hours before the time of the visit the System sends a notification to Registered Customer to remember him/her the booked visit. If the visit is scheduled for the 10:00 AM or earlier, the System sends the notification at 6:00 PM of the day before.
R17	If a Registered Customer booked a visit online and then his/her smartphone is turned off when the System sends the notification to remind the booked visit, then the Registered Customer will be able to see the notification when he/she turns on the device
R22	Using the WebApp, a Store manager can select the number of customers allowed to be contemporarily in queue in front of the store
R23	Using the WebApp, Store managers can select the number of customers allowed to be contemporarily inside the store
R24	Using the WebApp, Store managers can see information about their grocery store such as the average time of the visit (both general and for a specific Registered Customer), the number of people that entered the store day by day and the booked visits

R25	The System is able to use GPS, information about traffic condition in real time and information about movement methods chosen by Registered Customer to calculate the estimated time needed by a Registered Customer to reach the store
R26	The System must provide to a Registered Customer the estimated waiting time for the line up in real time
DA2	When a Registered Customer provides to the System the information on how he/she wants to reach the store (on foot, by car, with public transport), then the customer really reaches the store using the selected method.
DA4	All the grocery stores have a QR code reader in front of the store that is connected to the System and gives the possibility to scan QR codes present on tickets
DA5	Customers always really scans the QR code associated to the ticket both when they enter and when they leave the store
DA7	Internet connection works properly
DA9	QR readers work properly
DA10	GPS provides the position with an error of 100 meters at most

G2	Allow customers to do the grocery shopping safely - Allow customers to get a ticket to enter the grocery store without physically approaching the store - Allow customers to book a visit to enter the grocery store without physically approaching the store - Allow customers to line up from home and to physically approach the store just when their turn is close to come - Allow customers to not lose their turn even if they are not physically in queue in front of the store - Allow customers to be informed about the queue situation in real time even if they are lining up from home
R1	When a Customer tries to use the application Clup, the System controls if the device used has a GPS sensor integrated. If not, the Customer is notified that he/she cannot use the application because he/she doesn't have the required technology.
R2	The System shall ask to the Customer the permission to access the GPS position. If the Customer doesn't want to give the authorization, the System informs him/her that he/she can't use the service.
R3	A Registered Customer must be able to log in to the System using his/her credential
R4	A Registered Customer can use the system only if he/she is logged in
R5	A Registered Customer cannot take more than one ticket at a time
R6	A Registered Customer can book more than one visit at a time but not in the same date and the same time
R7	A Registered Customer can cancel the request of a ticket losing his/her turn and also the request of a booked visit
R8	A Registered Customer must be able to see the real time situation of the queue using his/her smartphone
R10	When a Registered Customer wants to get a ticket online, the System asks him/her if he/she intends to reach the store by car, using public transports or on foot in order to better calculate the estimated time needed by the Registered Customer to reach the store
R11	When a Registered Customer wants to get a ticket online, the System asks him/her to choose from a list of stores that are currently open and located in the same municipality of the current Registered Customer position. Stores that will close before that the customer will be allowed to enter are not selectable

R14	When a Registered Customer books a visit, the System asks him/her to insert data and time of the visit and also the address of the place where the Registered Customer will be just before the moment of the visit in order to identify and then show on screen the list of stores having a bookable visit on the selected date and time and near the selected address
R15	When a Registered Customer gets a ticket using his/her smartphone, then the System sends him/her a notification 10 minutes before that the estimated waiting time is equal to the time needed by Registered Customer to reach the store in order to notify him/her that is time to head to the store
R16	When a Registered Customer books a visit, 2 hours before the time of the visit the System sends a notification to Registered Customer to remember him/her the booked visit. If the visit is scheduled for the 10:00 AM or earlier, the System sends the notification at 6:00 PM of the day before.
R17	If a Registered Customer booked a visit online and then his/her smartphone is turned off when the System sends the notification to remind the booked visit, then the Registered Customer will be able to see the notification when he/she turns on the device
R18	If a Customer is not in front of the store when it's his/her turn, the System waits for 2 minutes then Customer loses his/her turn and the turn passes to the next Customer in line
R25	The System is able to use GPS, information about traffic condition in real time and information about movement methods chosen by Registered Customer to calculate the estimated time needed by a Registered Customer to reach the store
R26	The System must provide to a Registered Customer the estimated waiting time for the line up in real time
R27	When a Registered Customer wants to get a ticket online, the System shows also stores that cannot be selected (because they are closed or they will close before that the customer will be allowed to enter) in order to let the Customer understander if there are stores registered to the system in the municipality where he/she currently is. Of course, unselectable stores cannot be selected and they are showed at the bottom of the list, after the once that are selectable (if exists).
R28	Customers should be able to scan the QR Code on their tickets both when they enter and when they leave the store. The system should be able to update queues and stores information with data taken from the scan of the QR Codes.
DA1	If a Registered Customer books a visit it's because he/she has already organized his/her commitments so is able to tell the System the address of the place where he/she will be just before the moment of the visit
DA2	When a Registered Customer provides to the System the information on how he/she wants to reach the store (on foot, by car, with public transport), then the customer really reaches the store using the selected method.
DA3	All the grocery stores have a ticket distributor in front of the store that is connected to the System and gives the possibility to get a ticket on the spot
DA7	Internet connection works properly
DA8	Ticket distributors work properly
DA10	GPS provides the position with an error of 100 meters at most

G3	Allow Store managers to respect the restrictions imposed and to make grocery stores safer for customers - Allow Store managers to limit the number of people contemporarily visiting the store - Allow Store managers to limit the number of people contemporarily waiting in queue in front of the store - Allow Store managers to not lose customers because of the restrictions on the limited number of people that can contemporarily line up in front of the store and access it
R5	A Registered Customer cannot take more than one ticket at a time
R6	A Registered Customer can book more than one visit at a time but not in the same date and the same time
R7	A Registered Customer can cancel the request of a ticket losing his/her turn and also the request of a booked visit
R9	Everyone (both Registered and Unregistered Customers) must have the possibility to get a ticket, or using the dedicated app on the smartphone or using the ticket distributor that all the stores subscribed to the System must have
R12	When a Customer gets a Ticket (both online and in presence), the System generates an unique QR code associated with the ticket
R13	When a Registered Customer books a visit, he/she can select the category of goods that he/she intends to buy
R15	When a Registered Customer gets a ticket using his/her smartphone, then the System sends him/her a notification 10 minutes before that the estimated waiting time is equal to the time needed by Registered Customer to reach the store in order to notify him/her that is time to head to the store
R16	When a Registered Customer books a visit, 2 hours before the time of the visit the System sends a notification to Registered Customer to remember him/her the booked visit. If the visit is scheduled for the 10:00 AM or earlier, the System sends the notification at 6:00 PM of the day before.
R17	If a Registered Customer booked a visit online and then his/her smartphone is turned off when the System sends the notification to remind the booked visit, then the Registered Customer will be able to see the notification when he/she turns on the device
R19	In order to sign up, a Store manager has to prove that he/she they really runs a grocery shop sending documents related to the management of the store itself and licenses (required documents are later specified in the Use cases section)
R20	A registered Store manager must be able to log in to the System using his/her credentials
R21	A Store manager can use the system only if he/she is logged in
R22	Using the WebApp, a Store manager can select the number of customers allowed to be contemporarily in queue in front of the store
R23	Using the WebApp, Store managers can select the number of customers allowed to be contemporarily inside the store
R24	Using the WebApp, Store managers can see information about their grocery store such as the average time of the visit (both general and for a specific Registered Customer), the number of people that entered the store day by day and the booked visits
R28	Customers should be able to scan the QR Code on their tickets both when they enter and when they leave the store. The system should be able to update queues and stores information with data taken from the scan of the QR Codes.
DA3	All the grocery stores have a ticket distributor in front of the store that is connected to the System and gives the possibility to get a ticket on the spot
DA6	Information provided by Store managers about their grocery stores are always correct, up to date and in accordance with the laws issued by the government to face the emergency.
DA8	Ticket distributors work properly

3.2.3.Use Cases

3.2.3.1. Use Cases Description

USE CASE 1 – STORE MANAGER SIGNS UP

ACTORS	Store manager
INPUT CONDITIONS	Store manager has already installed a browser application on his/her device
EVENTS FLOW	1) The Store manager opens the browser 2) The Store manager opens the initial page of the Clup WebApp 3) The Store manager clicks on the "Sign up" button 4) The Store manager fills in the sign-up form *(1)* 5) The Store manager clicks on the "Sign Up!" button 6) The Store manager receives a confirmation of the registration via e-mail
OUTPUT CONDITIONS	The grocery store is now registered to the service and can log in
EXCEPTIONS	1) The Store manager has already registered his/her grocery store to the service 2) The Store manager inserts not valid data for at least one of the mandatory fields of the sign-up form Exception 1 is handled informing the Store manager and redirecting him/her to the home page where he/she can log in Exception 2 is handled informing the Store manager and redirecting him/her to the empty registration page

(1) Mandatory fields of the sign-up form

- Name and Surname of the Store manager

- Username and Password of the Store manager

- E-mail of the Store manager

- Phone number of the Store manager

- Name, city and address of the grocery store

- Documents certifying that the person who is registering really runs an activity that is a grocery store

- Scan of an identity document of the Store manager

- Maximum number of people allowed to be simultaneously inside the store and maximum number of people allowed to physically line up outside the store

- Opening days and hours of the grocery store

- Kind of goods that are sold in the store (he has to select from the following list and have to select at list one type of good: meat – fruit – vegetables – fish – cold cuts – pasta - products for breakfast and for snacks - household products)

USE CASE 2 – STORE MANAGER LOGS IN

ACTORS	Store manager
INPUT CONDITIONS	The Store manager has already registered to the service
EVENTS FLOW	1) The Store manager opens the browser on his/her device 2) The Store manager opens the initial page of the Clup WebApp 3) The Store manager clicks on the "Sign In" button 4) The Store manager inserts his/her credentials and presses the "Log In" button
OUTPUT CONDITIONS	The Store manager is now logged in and can check information such as the number of people that entered the grocery store day by day, the booked visits and the average time of the visit (both general and for a single registered user). He/She can also change information such as opening hours of the grocery store and the number of allowed people inside the store or physically in queue.
EXCEPTIONS	1) The Store manager inserts wrong credentials The exception is handled informing the Store manager and redirecting him/her to the login page where he can try again to log in.

USE CASE 3 – STORE MANAGER UPDATE GROCERY STORE INFORMATION

ACTORS	Store manager
INPUT CONDITIONS	The Store manager has logged in
EVENTS FLOW	1) The Store manager access the control panel (his/her home page) 2) If the Store manager wants, clicks on the "update opening hours" button, inserts the new opening hours and then confirms the changes 3) If the Store manager wants, clicks on the "update store capacity" button, inserts the new number of people that are allowed to contemporarily access the store or to physically lining up in front of it and then confirms the changes 4) If the Store manager wants, clicks on the "update kind of goods" button, selects the goods from the given list and then confirms the changes
OUTPUT CONDITIONS	Store information are now updated
EXCEPTIONS	1) The Store Manager deselects all the kind of goods 2) The Store Manager selects a store capacity equal to 0 3) The Store Manager select a closing hour that is before the opening hour All the exceptions are handled informing the Store manager that if he/she wants to do this operation, he/she has to close the store (no opening hours).

USE CASE 4 – UNREGISTERED CUSTOMER SIGNS UP

ACTORS	Unregistered Customer
INPUT CONDITIONS	Unregistered Customer has already installed the app "Clup"
EVENTS FLOW	1) Unregistered Customer opens the application "Clup" on his/her smartphone 2) Unregistered Customer clicks on the "Sign Up" button, inserts his/her data (Name, Surname, Username, Password and Phone Number, all mandatory fields) *(1)*, and signs up becoming a Registered Customer 3) The System saves the Registered Customer's data
OUTPUT CONDITIONS	Unregistered Customer is successfully registered to the service, becomes a Registered Customer and he/she is redirected to the Home Page where he/she can log in
EXCEPTIONS	1) "Unregistered" Customer is already a Registered Customer (his phone number is already associated to another account) 2) Unregistered Customer inserts not valid data for at least one of the mandatory fields Exception 1 is handled informing the "Unregistered" Customer and redirecting him/her to the home page where he/she can log in Exception 2 is handled informing the Unregistered Customer and redirecting him/her to the empty registration page

(1) Email is not asked because the software should be an easy-to-use application and especially older people may not have an email. Instead, if they have installed the app, they have for sure a smartphone, and so, in most of the cases, a phone number (they may have a smartphone without a SIM).
The username must be unique because it's used by the store manager to find the average time of the visit of a Registered Customer.

USE CASE 5 – REGISTERED CUSTOMER LOGS IN

ACTORS	Registered Customer
INPUT CONDITIONS	None
EVENTS FLOW	1) Registered Customer opens the application Clup on his/her smartphone 2) Registered Customer clicks on the "Sign In" button 3) Registered Customer inserts his/her credentials and press the "Log In" button
OUTPUT CONDITIONS	Registered Customer is logged in and can get a ticket or book a visit
EXCEPTIONS	1) Registered Customer inserts wrong credentials The exception is handled informing the Registered Customer and redirecting him/her to the login page where he can try again to log in.

USE CASE 6 – REGISTERED CUSTOMER GETS A TICKET ONLINE

ACTORS	Registered Customer
INPUT CONDITIONS	Registered Customer has already logged in
EVENTS FLOW	1) Registered Customer clicks on the "Get a ticket" button 2) The System uses the GPS to locate the current Registered Customer position 3) The System asks to Registered Customer if he/she wants to reach the store by car, using public transports or on foot and Registered Customer makes a choice 4) The System finds all the grocery stores near the current Registered Customer position and shows them on screen sorted by increasing order of distance from the current Registered Customer position *(1)* 5) Registered Customer selects the store where he/she wants to do the grocery shopping among the available. (During the selection, Registered Customer can see the current estimated waiting time and the number of people in queue for each store) *(2)* 6) The System shows on screen the Registered Customer ticket and real time information such as the number of people in line before Registered Customer and the estimated waiting time 7) Using the GPS and the selected moving mode, the System calculates the time that Registered Customer needs to reach the Store 8) 10 minutes before that the estimated waiting time is equal to the time needed by Registered Customer to reach the store, the System sends a notification to Registered Customer *(3)*
OUTPUT CONDITIONS	Registered Customer reaches the store and scans the QR Code in order to enter
EXCEPTIONS	No grocery store is found because they are all closed, because there are no grocery store that are registered to the system and that are in the same municipality of the customer or because they will close before that the Registered Customer will be allowed to enter. The exception is handled informing Registered Customer and asking him/her to try again the next day

(1) The System shows all the stores in the municipality, but only open grocery stores are selectable. Using information about the mode of transport, the Customer current position and the estimated waiting time of each store, the System will not let the customer to select also that stores that will close before that the Registered Customer will be allowed to enter. Unselectable stores are always showed at the bottom of the list, after the once that are selectable.

(2) The estimated waiting time is calculated considering the average time of the visit for the specific grocery store, the number of people that can contemporarily access the grocery shop and also the booked visits.

(3) At the moment in which the ticket is taken, if (estimated waiting time - time needed by Registered Customer to reach the store) < 10 minutes, the notification is sent immediately

[The sent of the notification can also be modelled with a separate Use Case that has as input condition "(estimated waiting time - time needed by Registered Customer to reach the store) <= 10 minutes"]

USE CASE 7 – CUSTOMER GETS A TICKET IN PRESENCE

ACTORS	Customer (Registered or Unregistered), Ticket distributor
INPUT CONDITIONS	Customer is in front of the store
EVENTS FLOW	1) Customer reaches the ticket distributor 2) Customer clicks the "Get a Ticket" button that appears on the screen of the Ticket Distributor 3) The System emits a ticket and Customer takes it 4) The System shows the information concerning the estimated waiting time and the number of people in queue on the screen and Customer reads it
OUTPUT CONDITIONS	Customer has a ticket and can decide to line up and wait for his/her turn or to go away
EXCEPTIONS	1) The System cannot emit a ticket because the estimated time in which Customer will be able to access the grocery store is after the closing time The exception is handled informing Customer that there are no more tickets available and asking him/her to choose another grocery shop or to try again the next day

USE CASE 8 – REGISTERED CUSTOMER BOOKS A VISIT

ACTORS	Registered Customer
INPUT CONDITIONS	Registered Customer has already logged in
EVENTS FLOW	1) Registered Customer clicks on the "Book a Visit" button 2) Registered Customer chooses a date and a time for the booking 3) Registered Customer inserts the name of the street and the civic number of the place where he/she will be just before leaving to reach the store the day of the booked visit 4) The System uses the GPS to locate the selected place, then finds all the grocery stores near it 5) The System selects only that stores that have a bookable ticket available for the date and the time selected by Registered Customer (with a tolerance of 30 minutes) and then shows the identified stores on screen 6) Registered Customer selects a grocery store among those shown on screen, books a visit and receives a confirmation message from the System 7) If Registered Customer wants, he/she can select the category of products that intends to buy 8) 2 hours before the time of the visit *(1)* the System send a notification to Registered Customer to remember him the booked visit 9) If the visit is scheduled for the 10:00 AM or earlier, the System sends the notification at 6:00 PM of the day before.
OUTPUT CONDITIONS	Registered Customer reaches the store and scans the QR Code in order to enter
EXCEPTIONS	1) There isn't any store with a ticket available on the date and time selected The exception is handled informing Registered Customer that there are no more tickets available and asking him/her to choose date and time.

(1) 2 hours is a reasonable time because, considered the situation of emergency, Registered Customer will most likely be at home. Furthermore, Registered Customer can only do the grocery shopping in stores situated inside his/her municipality, so he/she is unlikely to walk or drive for more than 2 hours to reach the store. If the System sends the notification too early, is more likely that Registered Customer forgets it.

[The sent of the notification can also be modelled with a separate Use Case that has as input condition "the visit is scheduled before the 10:00 AM and it's 6:00 PM of the day before the visit orit's 2 hours before the moment of the visit"]

USE CASE 9 – CUSTOMER ENTERS THE STORE

ACTORS	Customer (Registered or Unregistered), QR code reader
INPUT CONDITIONS	Customer has already got a ticker or booked a visit Customer has reached the store and it's now his/her turn QR Code scanner works properly
EVENTS FLOW	1) Customer opens the application Clup on his/her smartphone (or has a ticket emitted from the distributor in his/her hands) 2) Customer scans the QR code displayed on his screen
OUTPUT CONDITIONS	Customer enters the store
EXCEPTIONS	None

USE CASE 10 – CUSTOMER LEAVES THE STORE

ACTORS	Customer (Registered or Unregistered), QR code reader
INPUT CONDITIONS	Customer has already entered the grocery store Customer has finished his/her grocery shopping Customer is at the shop cashier and has already paid QR Code scanner works properly
EVENTS FLOW	1) Customer opens the application Clup on his/her smartphone or has a ticket emitted from the distributor in his/her hands) 2) Customer scans the QR code displayed on his screen
OUTPUT CONDITIONS	Customer leaves the store
EXCEPTIONS	None

USE CASE 11 – REGISTERED CUSTOMER CANCELS TICKET REQUEST

ACTORS	Registered Customer
INPUT CONDITIONS	Registered Customer has already got a ticket
EVENTS FLOW	1) Registered Customer opens the application Clup on his/her smartphone 2) Registered Customer clicks on the "Profile" button 3) Registered Customer clicks on the "Active Ticket" button 4) Registered Customer clicks on the Ticket and then clicks on "Cancel request"
OUTPUT CONDITIONS	Registered Customer is not lining up anymore
EXCEPTIONS	None

USE CASE 12 – REGISTERED CUSTOMER CANCELS A BOOKED VISIT

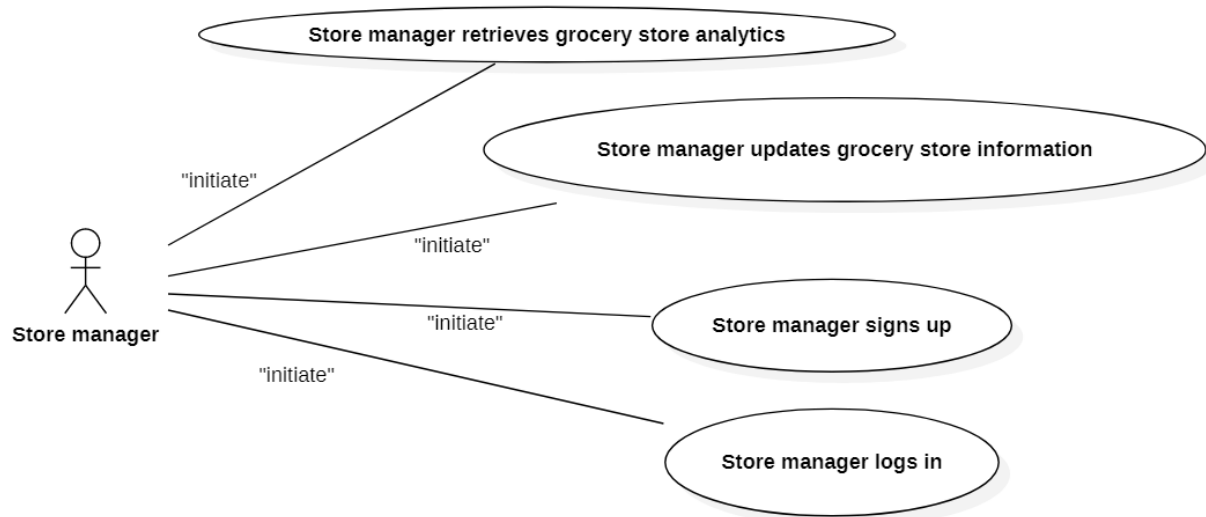| ACTORS	Registered Customer
INPUT CONDITIONS	Registered Customer has already booked a visit
EVENTS FLOW	1) Registered Customer opens the application Clup on his/her smartphone 2) Registered Customer clicks on the "Profile" button 3) Registered Customer clicks on the "Booked visits" button 4) Registered Customer clicks on the booked visit that he wants to cancel and then clicks on "Cancel request"
OUTPUT CONDITIONS	Registered Customer has successfully cancelled the request for the booked visit
EXCEPTIONS	None

Use cases about the Store manager retrieving the general average time of the visit and the average time of the visit of a specific user have not been reported in the RASD because he/she only needs to click a botton to retrieve the desired information, so these use cases are not significative.

Anyway, in the DD, there is a dedicated mockup (Store Manager control panel) that shows the bottons needed for these two operations.
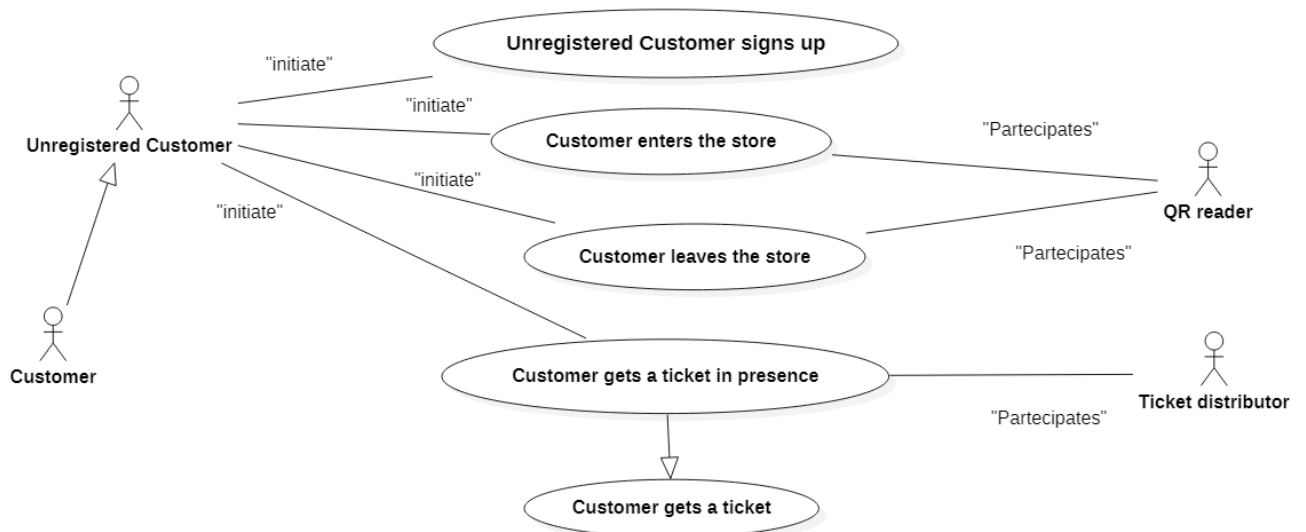
3.2.3.2. Use Cases Diagram

For legibility reasons, the use cases of "Registered Customer", "Unregistered Customer" and "Store manager" are divided in 3 different images.
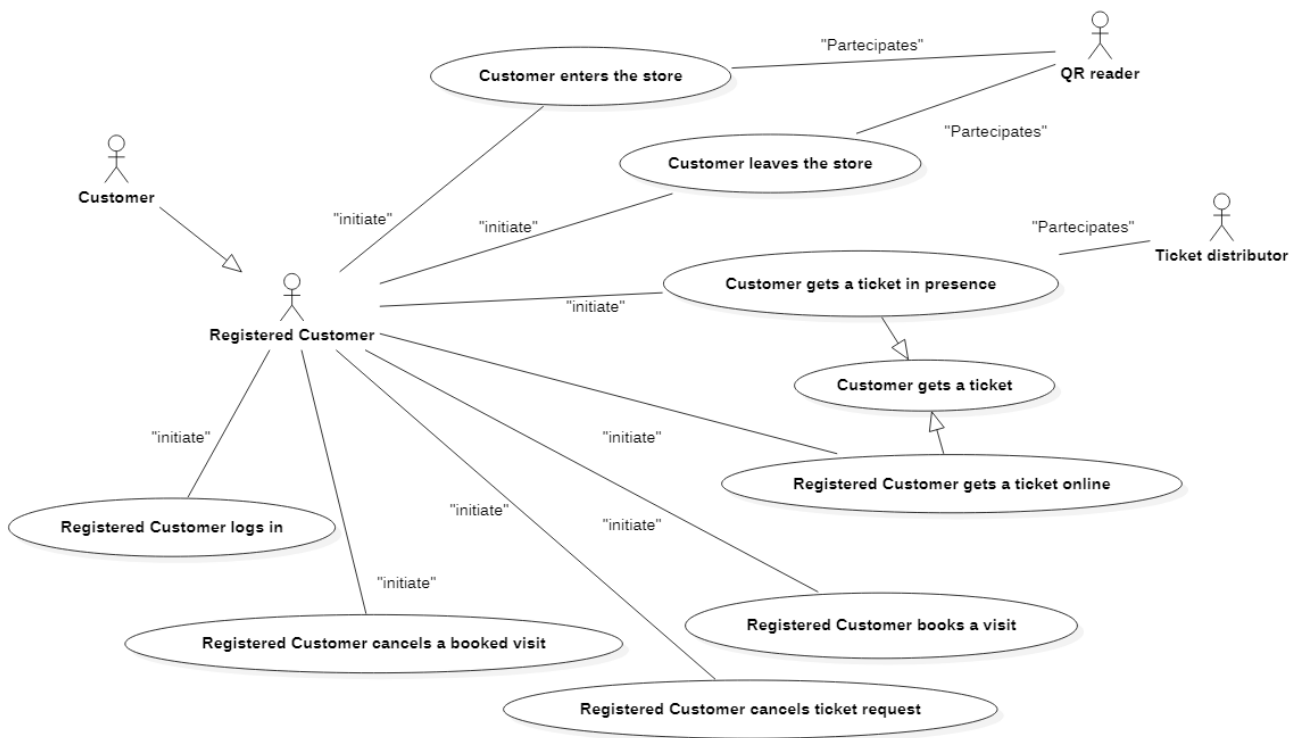
Store manager use cases diagram
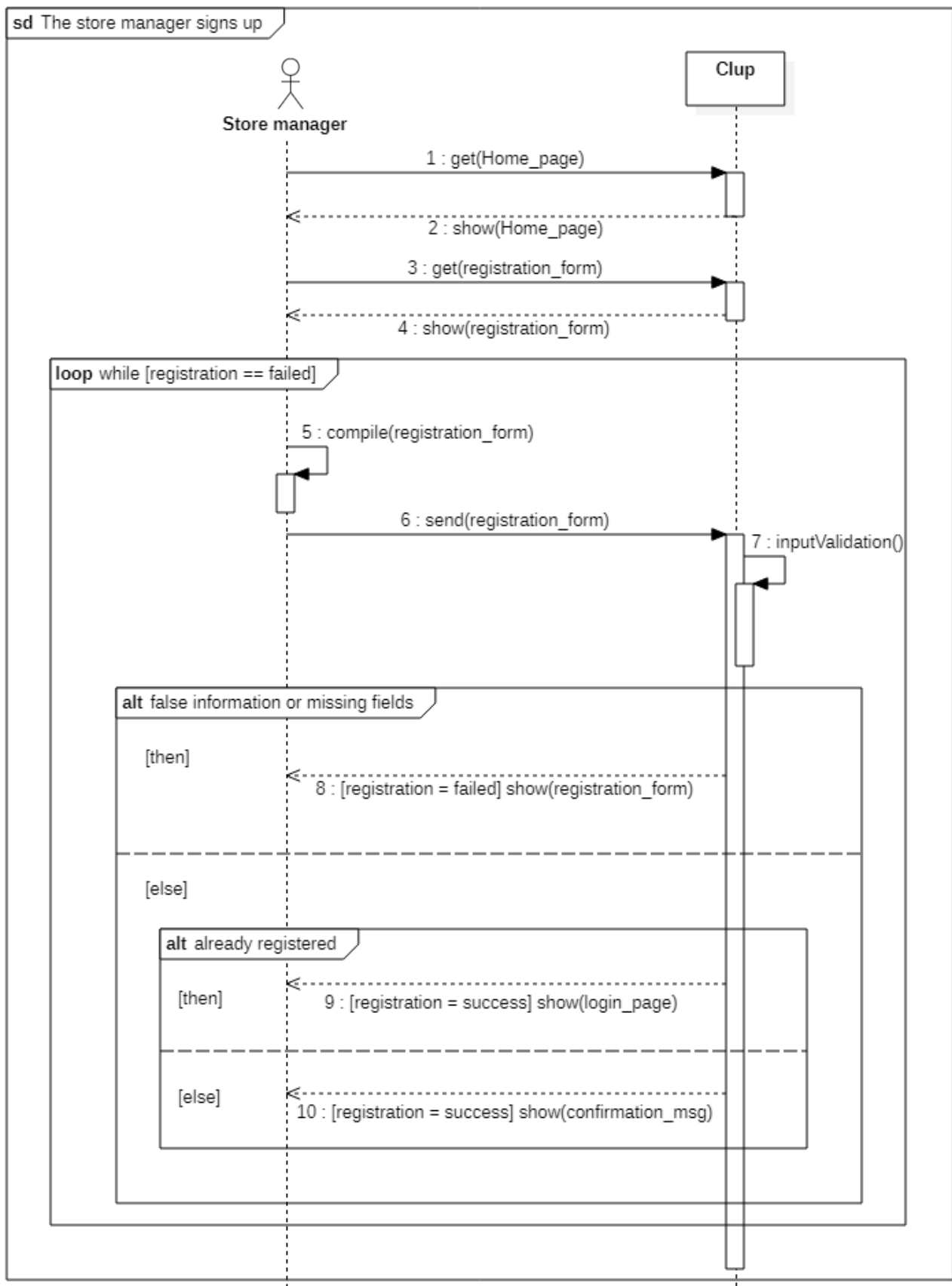

Unregistered customer use cases diagram

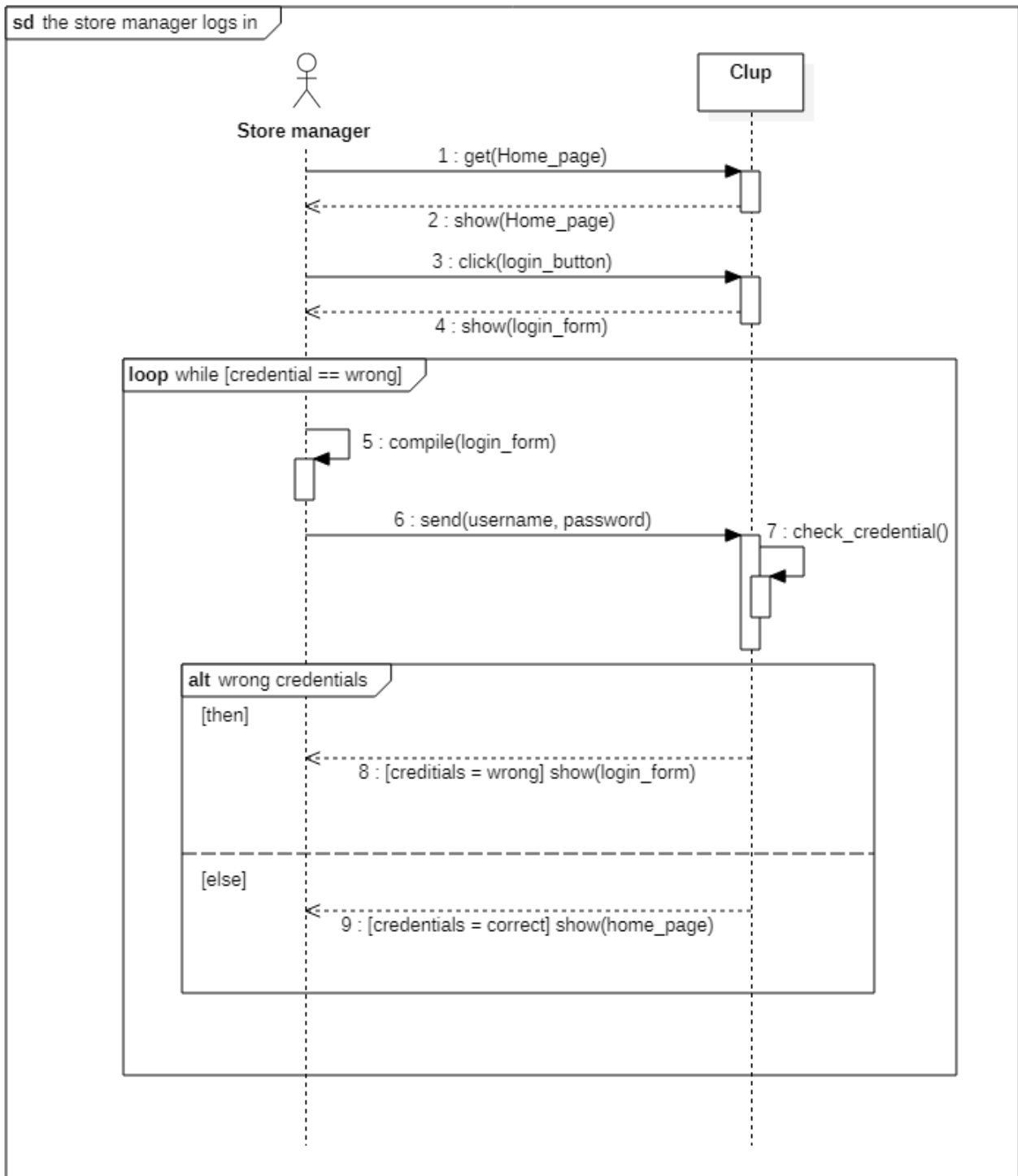## Registered customer use cases diagram

3.2.4. Sequence Diagrams

1) Store manager signs up

2) Store manager logs in

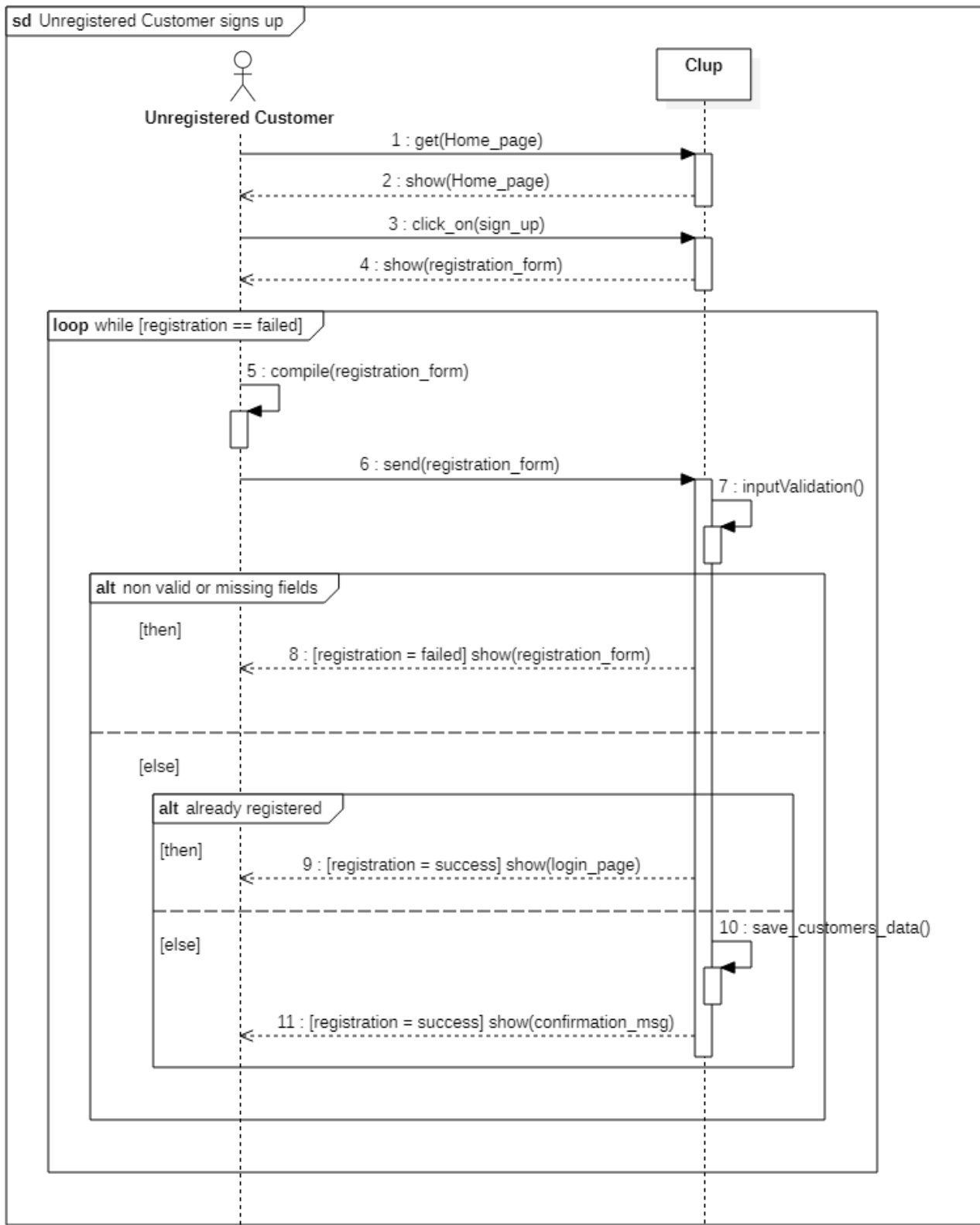### 3) Store manager updates grocery store information

inputValidation() controls that new information inserted by the store manager are correct and so that:

1) There is at least one kind of good that is sold by the store
2) The number of people that are allowed to enter the store is greater then 0
3) The closing hour is after the opening hour (strictly greater)
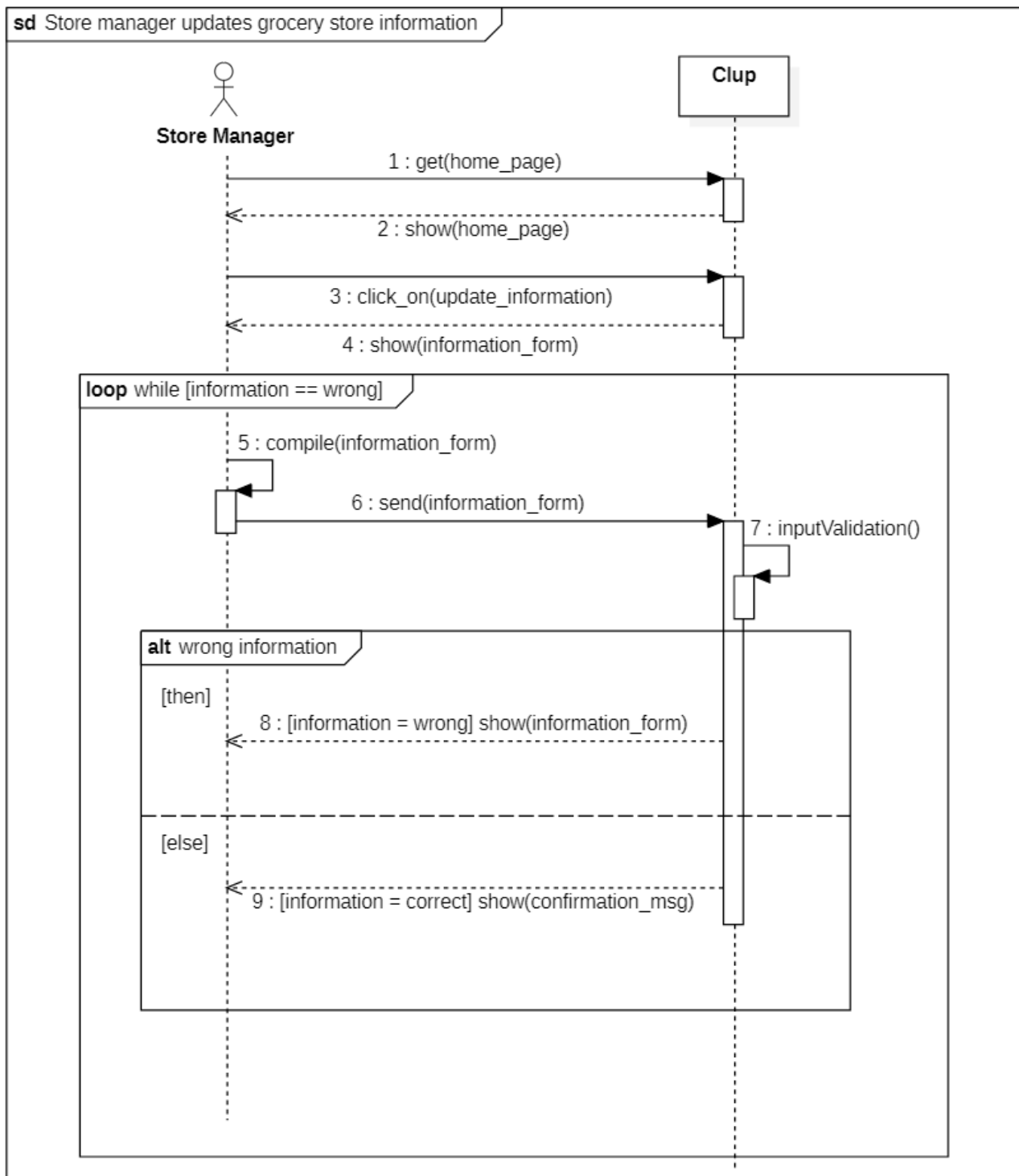
4) Unregistered Customer signs up

5) Registered Customer logs in

sd registered customer logs in

Registered Customer

Clup

1 : get(Home_page)

2 : show(Home_page)

3 : click(login_button)

4 : show(login_form)

loop while [credential == wrong]

5 : compile(login_form)

6 : send(username, password)

7 : check_credential()

alt wrong credentials

[then]

8 : [creditials = wrong] show(login_form)

[else]

9 : [credentials = correct] show(home_page)
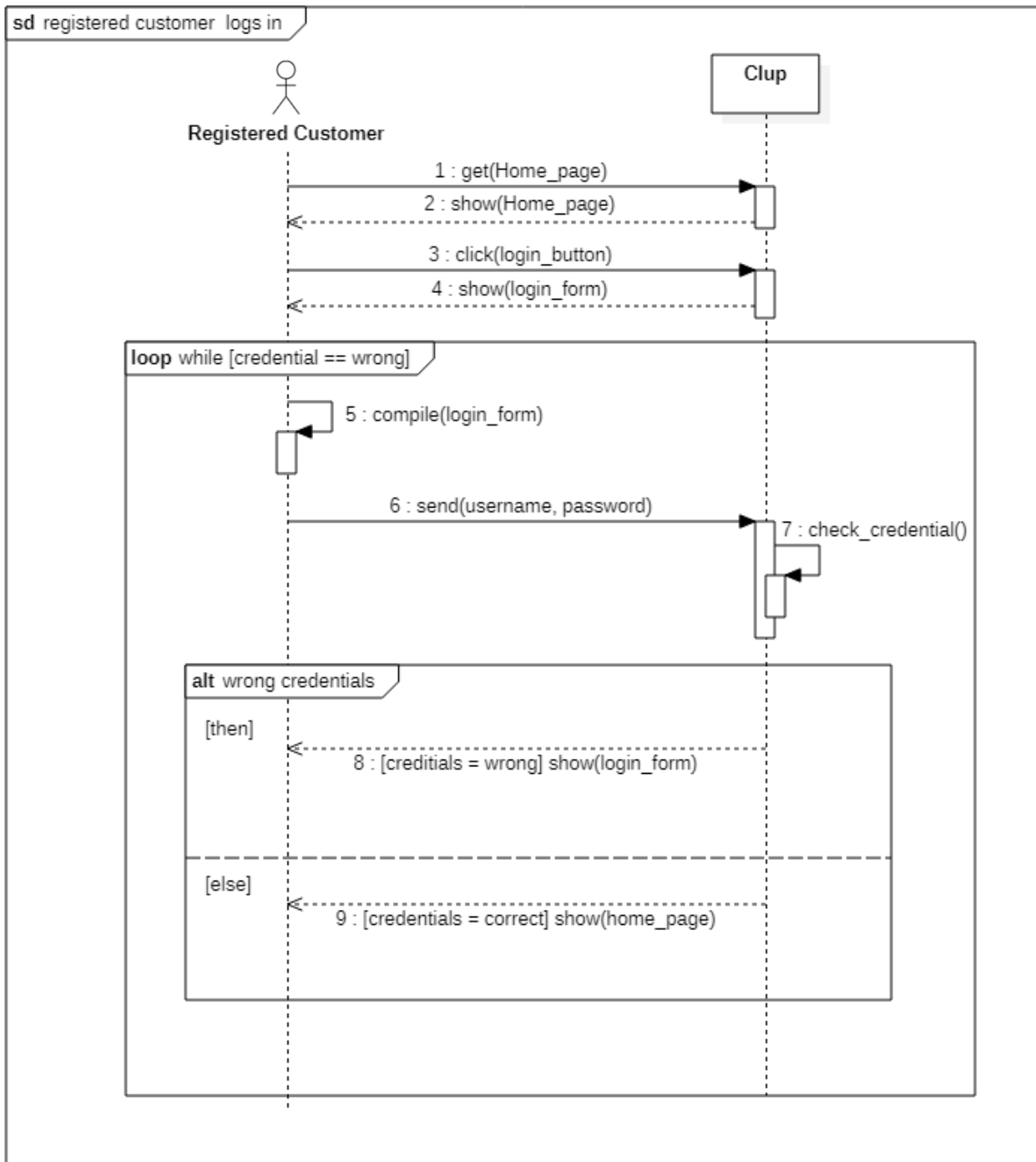
6) Registered Customer gets a ticket online

7) Customer gets a ticket in presence

sd Customer gets a ticket in presence

Customer — Ticket Distributor — Clup

1 : click_on(get_ticket)

2 : get_ticket()

3 : calculate_estimated_waiting_time

alt estimated_waiting_time < closing time

[then]

4 : emit_ticket()

5 : show_informations()

[else]

6 : reject_ticket()

7 : show(error_msg)
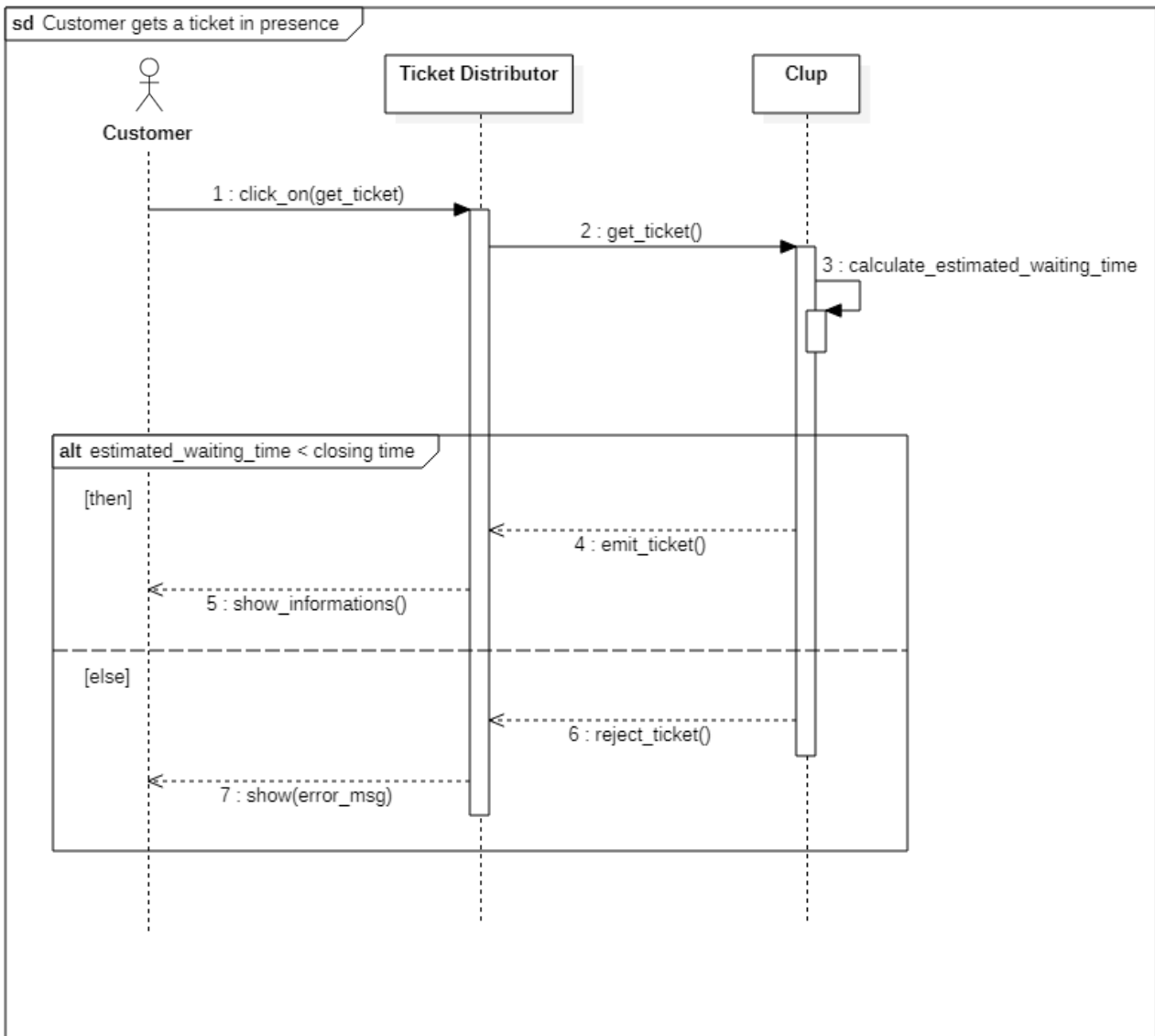
8) Registered customer books a visit

booking_info contains the data of the visit, the time of the visit and the address where the Registered Customer will be located just before the moment of the visit

nodtification_condition = if the visit is scheduled before the 10:00 AM the notification_condition is "it's 6:00 PM of the day before the visit", otherwise notification_condition is "it's 2 hours before the moment of the visit"
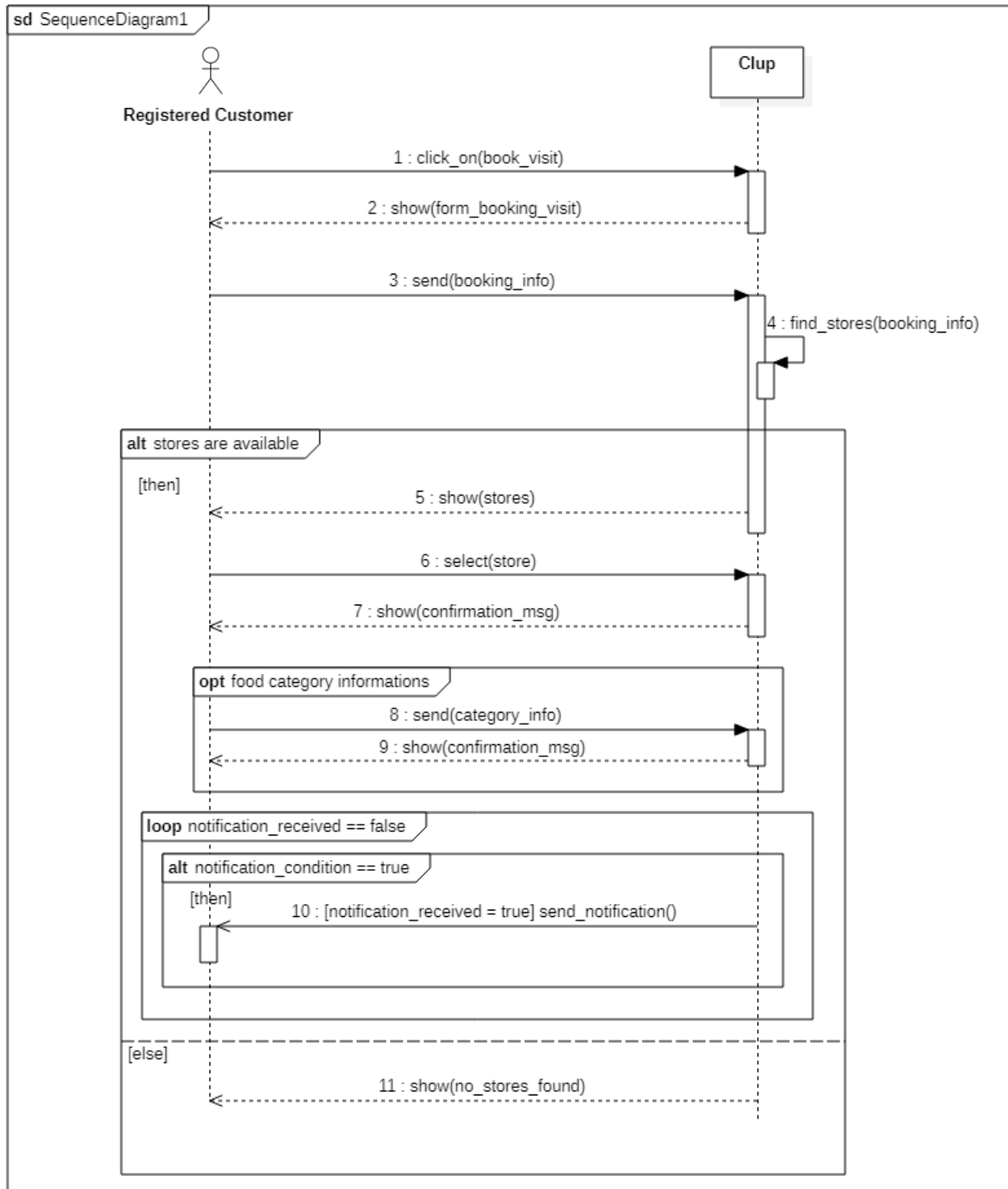
9) Customer enters the store
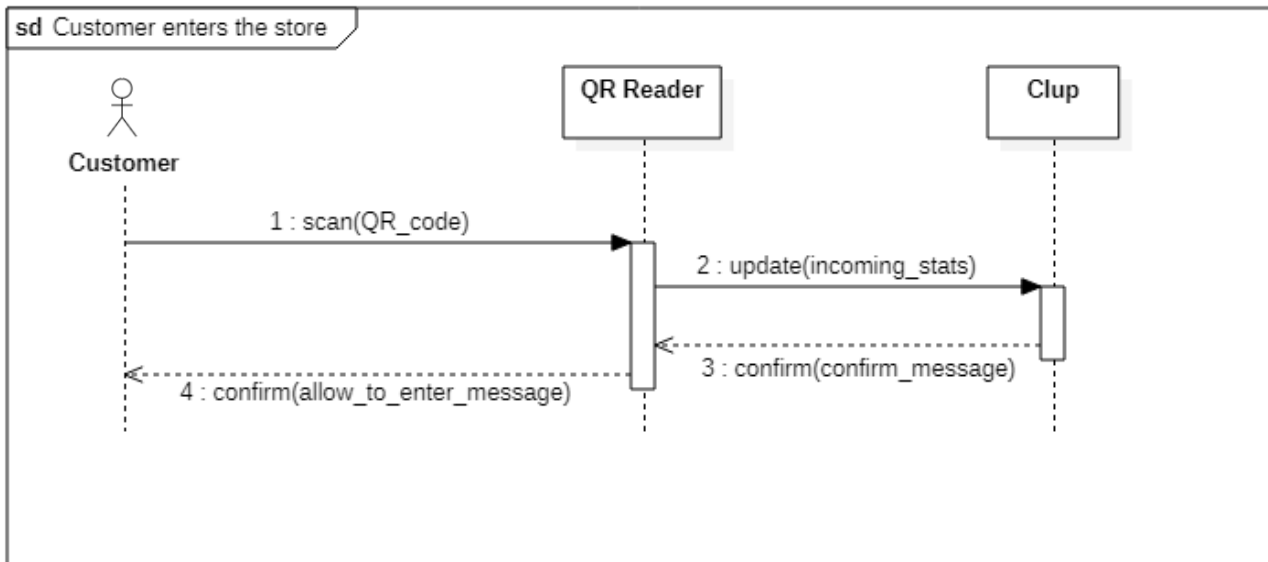
10) Customer leaves the store
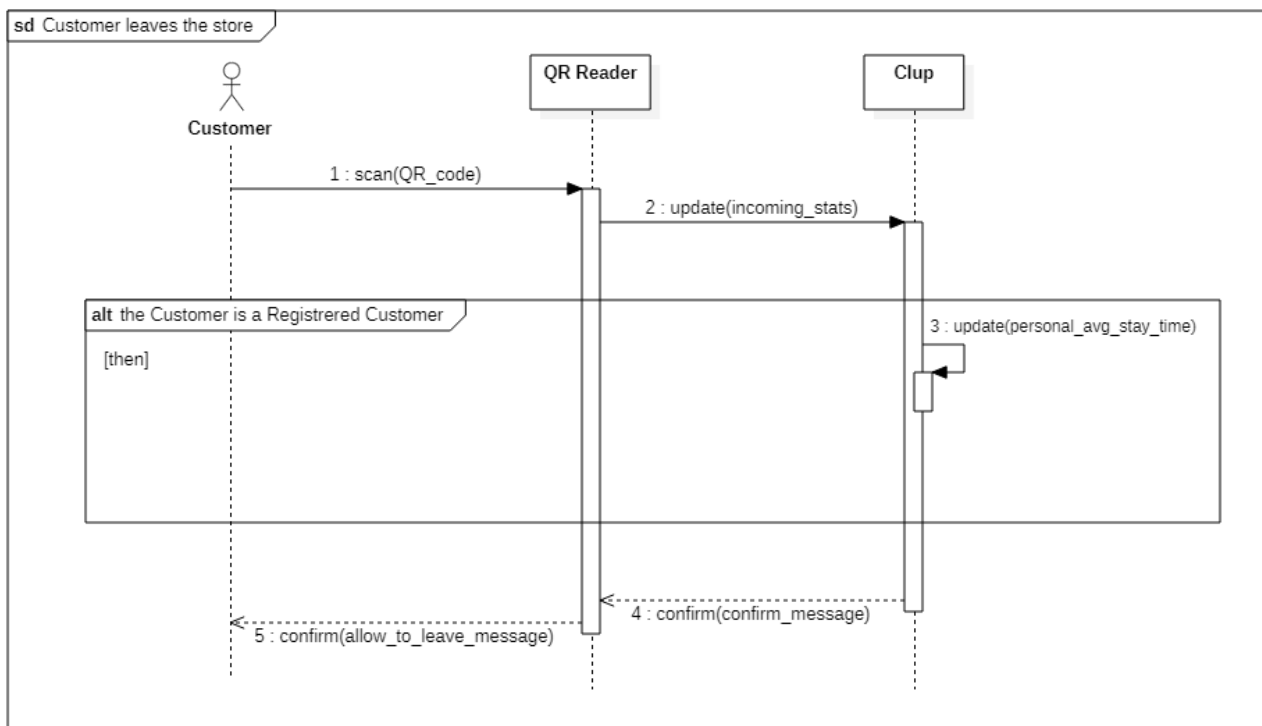
11) Registered customer cancels ticket request

sd Registered customer cancels ticket request

Registered Customer — Clup

1 : click_on(profile)

2 : show(profile_page)

3 : view_active_ticket()

4 : show(active_ticket_info)

5 : cancel(ticket)

6 : cancel_ticket(ticket)

7 : show(confirmation_msg)
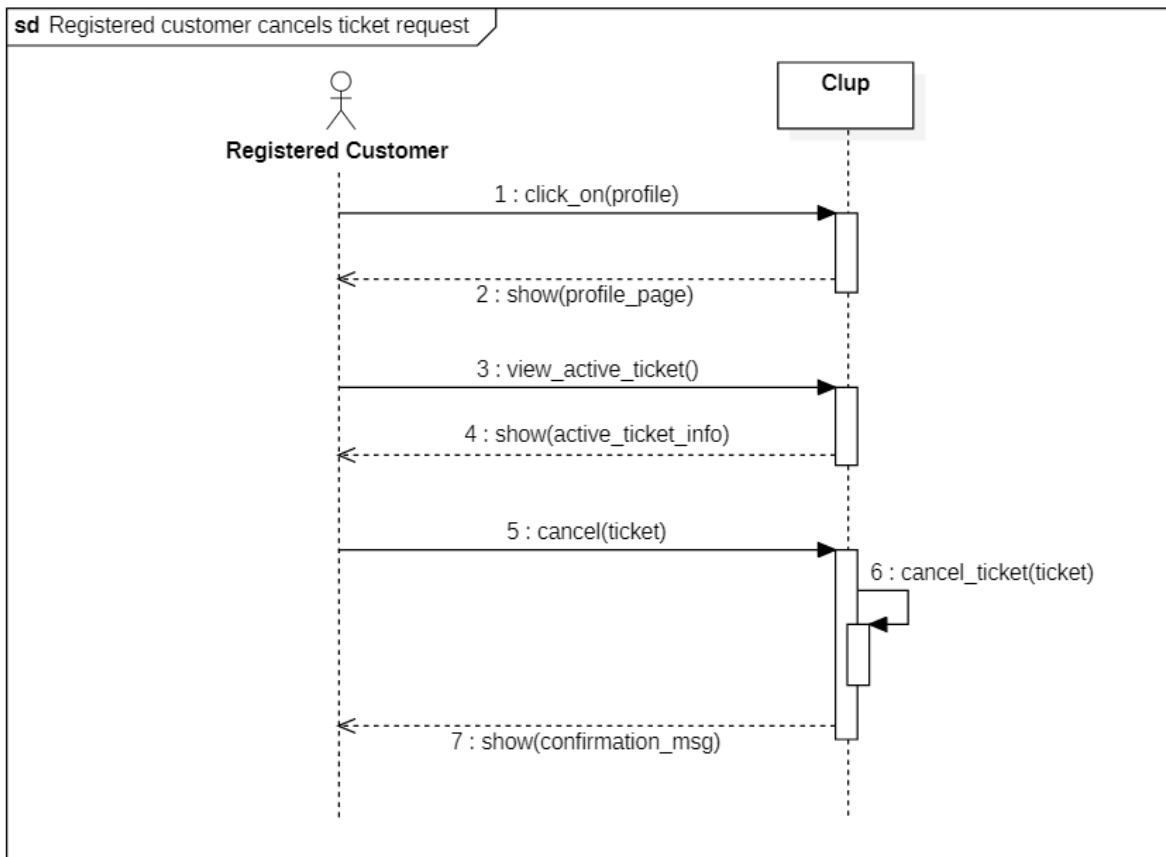
12) Registered customer cancels a booked visit
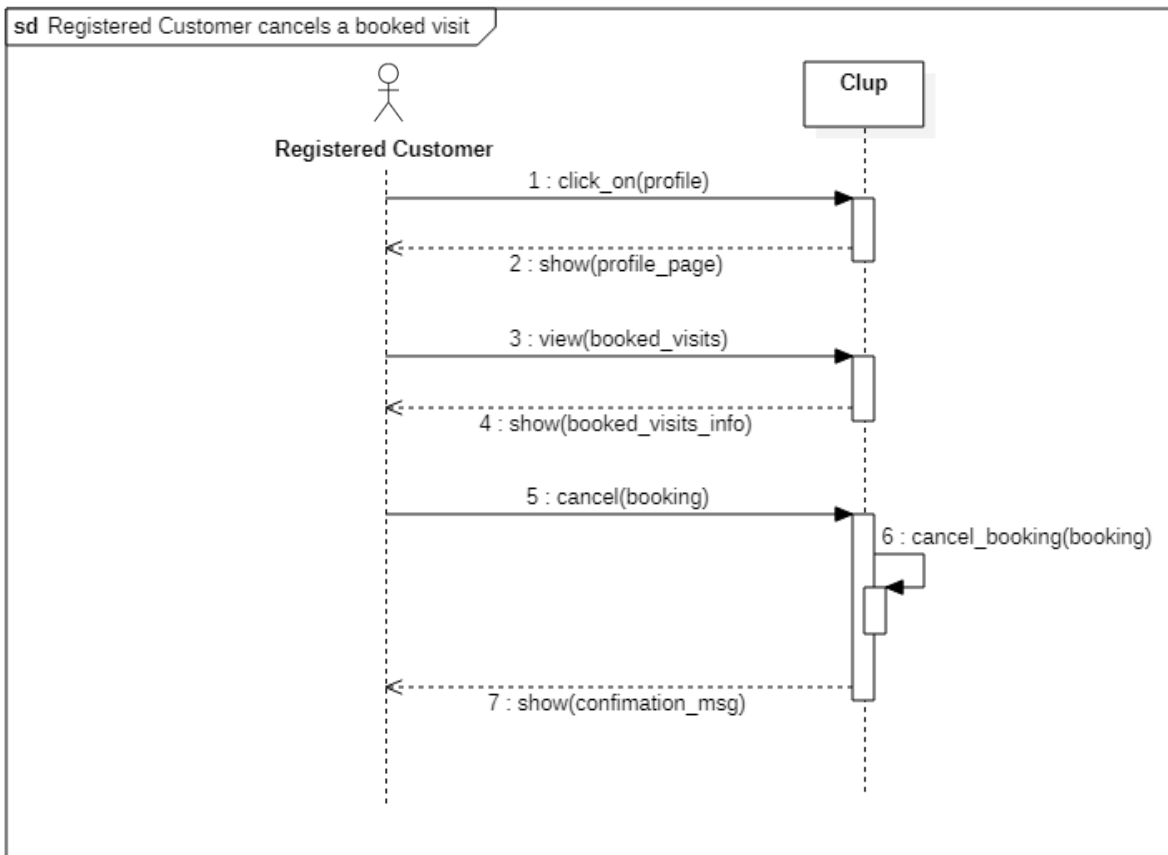
3.2.5. Scenarios

Scenario 1 – Getting a ticket online and lining up from home

Matteo is 22 years old and he's an off-site student: he studies in Milan while his family is in Rome.
Matteo decided to not come back in Rome during the coronavirus emergency and to stay in Milan passing the quarantine with his girlfriend Sara. Among all the other things, Matteo and Sara have to do the grocery shopping.
Matteo downloads and installs the app "Clup" on his smartphone, he easily signs up by entering his credentials and after the registrations he logs in and starts to look for a grocery store nearby.
Matteo discovers that the grocery store where he usually goes is registered to the service and sees that the current estimated waiting time is 2 hours.
2 hours it's a lot of time, anyway Matteo decides to get a ticket because that store is the closest one to his house, has everything that he needs and because, thanks to Clup, he knows that he doesn't need to wait standing in front of the store but he can line up from home while studying for Software Engineering 2 exam.
Matteo has to choose between 3 options: "Go to the store on foot", "Go to the store by car" and "Go to the store using public transports". He chooses the first one. Using GPS and the information regarding the movement method, the System calculates that the time it takes to Matteo to reach the store from his current position is 10 minutes. The System will send a notification to Matteo when his turn is close to come.
Once the notification arrives, Matteo walks towards the store, waits for his turn, scans the QR code to enter the store, does the grocery shopping, exits the store scanning the QR code again and then he comes back home.

Scenario 2 – A store manager signs up to the service

Alberto is a 45 years old man, he lives in Milan and he runs an important grocery store of the city, usually frequented by many people.
Due to the Coronavirus emergency, Alberto has to respect strict rules such as the limitations on the number of people that can contemporarily access the building. Guaranteeing the respect of the rules it's not an easy task, especially seen the large number of customers that need to do the grocery shopping every day, and Alberto is worried about losing that customers that cannot immediately enter the store.
To solve this problem, Alberto decides to signs up to Clup, a system that can really help him during this difficult situation. Using his computer, Alberto opens the Clup WebApp, he clicks on the signs up button and fills the dedicated form to register to the service.
Alberto is asked to insert personal information such as his name, surname, email address, phone number but also general information about the store such as the city where the store is located, the address, the opening hours and the kind of goods that are sold inside the store (the types of goods are selected checking from a pre-compiled list).
Alberto has also to upload a document that certifies that he really is the store manager of a grocery store and a personal document to certify his identity.
Finally, Alberto is asked to insert the number of people that are allowed to be contemporarily inside the store and the number of people that can simultaneously be physically in queue outside the store.
This two information are asked to the store manager because they depend on many things such as the dimensions of the store (considering also the disposition and the quantity of shelves), on the situation of the emergency (the rules can change very quickly because of different DPCM) and on the average time of the visit, information that Alberto can retrieve using the WebApp.
Once Alberto has filled all the fields of the registration form, he can click on the "registration" button: from this moment on, customers can see Alberto's grocery store among the available stores on the Clup app, so they can get a ticket (or book a visit) to enter the store and do grocery shopping.

Scenario 3 – Booking a visit

Ludovica is the founder of a growing start up, she is always very busy and she uses an agenda to annotate all her work commitments.
Ludovica needs to do the grocery shopping but this week she's very busy because of all the meetings with the new investors.
Looking at her agenda, Ludovica discovers that she will be free only on Wednesday evening, from 19 onwards.
Ludovica, which he has already installed the app Clup, opens the app on her smartphone, she presses the "Book a visit" button and then inserts the name of the street and the civic number of the place where she will be on Wednesday, so the place from which she intends to leave in order to reach the grocery store.
Ludovica also inserts the date and the time in which she intends to visit the grocery store and then the System shows on screen a list of grocery stores that are near the place that Ludovica has indicated and that have bookable ticket for the date and the time selected.
Ludovica selects a store and, because she's vegetarian, she selects the kind of products that she wants to buy. The System understands that she will not go to the butcher's area. She confirms the booking and then receives a notification from the System. On Wednesday, at 17 o'clock, the System sends to Ludovica a new notification to remind her of the booked visit.
Carlo, a colleague of Ludovica that will be with her on Wednesday, also wants to book a visit for Wednesday at 19 o'clock and in the same store. Carlo is organizing a barbeque so he specifies that he will only buy meat. The System understands that Carlo will only visit the butcher's area and so organizes his visit contemporarily to Ludovica's once because they will visit different areas of the store and so the social distance will be respected.

Scenario 4 – Unregistered Customer

Giuseppe is 74 years old and he does not have a smartphone. He needs to do grocery shopping, so he decides to go to the grocery store near his house. He leaves his house and walks towards the store. When he arrives, he takes a ticket from the machine in front of the store (ticket distributor).
Giuseppe reads on the display of the ticket distributor that the estimated waiting time is 20 minutes, so he decides to line up. When his turn arrives, he enters the store scanning the QR code printed on the ticket. Finally, he can do the grocery shopping, he leaves the store scanning the QR code again to notify that he's leaving the store and then come back home.

Scanerio 5 – Cancellation of a ticket request

Giulio and Marta are married, they have two children, full time jobs and they are always very busy.
They need to do the grocery shopping but they are not very well organized.
Giulio arrives at home and suddenly remembers that he has to do the grocery shopping: he opens the Clup application on his smartphone, he gets a ticket online and waits for his turn. While he's waiting at home, Marta arrives with 2 big bags: she has already done the grocery shopping because she finished work earlier. Giulio thanks his wife and uses his smartphone to cancel the ticket request.

3.3. Performance Requirements - Non-Functional Requirements

3.3.1. Performance

Considering that grocery shopping is an essential need and considering also the strict limitations imposed by governments during the period of the Coronavisur emergency, it is expected that many store managers will rely on Clup and consequently many customers.
The System must provide information on the queue situation in real time and has to find grocery's stores near the customer's current position in under 10 seconds.

3.4. Design Constraints - Non-Functional Requirements

3.4.1. Standards compliance

The System must require to customers the permission to get their position. If the customer doesn't provide to the system the authorization to access his/her position, he/she cannot benefit of the service.
The System must manage sensible data (position, phone number, e-mail addresses) coming from customers and store managers, respecting their privacy. Telephone numbers, e-mail addresses and other personal information won't be used for commercial uses.
Store managers must obey the regulations imposed by the authorities regarding the maximum number of people admitted at the same time in the shop according to the size of the shop.

3.4.2. Hardware limitations

Registered customer must own a smartphone. The store managers can use a computer to interact with the web app. In both situations an internet connection is required.

In details:

Registered Customers:

- Mobile App
  - iOS or Android smartphone
  - 2G/3G/4G/5G connection
  - GPS

Store managers:

- Web App:
  - Computer with a browser

Moreover, to control accesses, every store must have a QR-reader in front of the entrance and a ticket distributor to provide tickets on the spot to Unregistered customers.

3.4.3. Any other constraint

Considering only tickets that are not associated to a visit, a registered customer can have at most one active ticket at a time.

3.5. Software System Attributes - Non-Functional Requirements

3.5.1. Reliability and availability

The availability of the system should be up to 99%. That means that the average time between the occurrence of a fault and the service recovery (MTTR or downtime) should be contained around 3.65 days per year. Customers who want to book a visit outside the opening hours of the grocery stores must be able to do it, however, it is not necessary to provide a higher level of availability considering that some of the functionality, such as the emission of a ticket, can be offered only during the opening hours of grocery stores. In order to achieve this percentage, it is necessary to add or to build redundancy into the System, including multiple power supplies and processors and backups, so that the failure of a component does not imply the failure of the entire System. The System has to detect failures as they occur and has to warn users about the failure within 10 minutes with a specific message addressed to the devices.

3.5.2. Security

Each Registered Customer has his/her own access credentials which must be stored in a database by the System after being hashed. Moreover, the System must ensure the protection of sensitive information: mechanisms against disclosure, alteration or destruction of data must be provided. In case of password recovery, this should never be sent in clear.

3.5.3. Maintainability

The system must be easy to repair, improve and understand. To achieve this, the code must have the following characteristics:

- Readable
- Well documented
- Modular

Automated tests should be provided in order to make it easy to validate changes.

3.5.4. Portability

The software is mainly developed to be used by customers through their smartphones, so Android and iOS have to support it. Furthermore, considering that the System is also used by store managers through the dedicated WebApp, the software must be able to run also on the most used operating systems for PCs, namely Windows, Mac and Linux.

4. Formal Analysis Using Alloy

```
abstract sig Customer{
        tickets: set Ticket
}

sig UnregisteredCustomer extends Customer{}

sig RegisteredCustomer extends Customer{
        visits: set Visit
}

sig StoreManager{
        stores: set GroceryStore
}{ #stores > 0 }

sig Ticket {
        qr_code: one QRCode,
        status: one TicketState
}

sig QRCode{}

abstract sig TicketState{}

one sig Issued extends TicketState{}

one sig EntranceScanned extends TicketState{}

one sig ExitScanned extends TicketState{}

one sig Expired extends TicketState{}

sig Visit{
        ticket: one Ticket,
        grocery_store: one GroceryStore,
        schedule: one Schedule
}

sig Schedule{
        day: one Int,
        month: one Int,
        year: one Int,
        hour: one Int
}{
        day > 0
        month > 0
        year > 0
        hour > 0
}
```

```
sig GroceryStore{
        distributor: one TicketDistributor,
        tickets: set Ticket,
        qr_reader: one QRReader,
        queue: one Queue
}

sig TicketDistributor{
        tickets: set Ticket
}

sig QRReader{
        tickets: set Ticket
}

sig Queue{
        customers: set Customer
}

// A queue can exist only if it is associated to a grocery store
fact OneGroceryStoreForEachQueue{
        all q: Queue | one g: GroceryStore | q = g.queue
}

// A ticket distributor can exist only if it is associated to a grocery store
fact OneGroceryStoreForEachDistributor{
all d: TicketDistributor | one g: GroceryStore | d = g.distributor
}

// A QR reader can exist only if it is associated to a grocery store
fact OneGroceryStoreForEachQRReader{
        all qr: QRReader | one g: GroceryStore | qr = g.qr_reader
}

// A grocery store can exist only if it is associated to a store manager
fact OneStoreManagerForGroceryStore{
        all g: GroceryStore | one s: StoreManager | g in s.stores
}

// A ticket can exist only if it is associated to a grocery store and to a customer
fact OneGroceryStoreForEachQueue{
        all t: Ticket | one g: GroceryStore | one c: Customer | t in g.tickets and t in c.tickets
}

// A schedule can exist only if it is associated to a visit
fact OneScheduleForAVisit{
        all s: Schedule | one v: Visit | v.schedule = s
}
```

```
// Different tickets have different QR Codes
fact DifferentTicketDifferentQRCode{
        all t1:Ticket, t2: Ticket |  t1 != t2 implies t1.qr_code != t2.qr_code
}

// Different visits have different tickets
fact DifferentVisitDifferentTicket{
        all v1:Visit, v2:Visit | v1 != v2 implies v1.ticket != v2.ticket
}

// Different grocery stores has different queues
fact DifferentGroceryStoreDifferentQueue{
        all gs1: GroceryStore, gs2: GroceryStore | gs1 != gs2 implies gs1.queue != gs2.queue
}

// Different grocery stores has different ticket distributors
fact DifferentGroceryStoreDifferentTicketDistributor{
        all gs1: GroceryStore, gs2: GroceryStore | gs1 != gs2 implies gs1.distributor != gs2.distributor
}

// Different grocery stores has different QR readers
fact DifferentGroceryStoreDifferentQRReader{
        all gs1: GroceryStore, gs2: GroceryStore | gs1 != gs2 implies gs1.qr_reader != gs2.qr_reader
}

// A ticket belongs to exactly one customer
fact OneCustomerForEachTicket{
        all c1,c2: Customer | c1 != c2 implies no (c1.tickets & c2.tickets)
}

// A ticket can belong to the set of tickets of only one grocery store
fact OneTicketOneGroceryStore{
        all g1, g2: GroceryStore, t: Ticket | g1 != g2 and t in g1.tickets implies t not in g2.tickets
}

// A visit can belong to the set of visits of only one registered customer
fact OneTicketOneGroceryStore{
        all rc1, rc2: RegisteredCustomer, v: Visit | rc1 != rc2 and v in rc1.visits implies v not in rc2.visits
}

// A visit for Grocery Store g must be associated to a ticket for Grocery Store g
fact VisitForGroceryStoreGAssociatedToTicketForGroceryStoreG{
        all v: Visit, g: GroceryStore, t: Ticket | v.ticket = t and v.grocery_store = g implies t in g.tickets
}
```

```alloy
// A ticket issued by the ticket distributor of the grocery store g belongs to the set of tickets of the grocery store
g
fact TicketIssuedFromDistributorOfGroceryStoreGAreForGroceryStoreG{
        all t: Ticket, td: TicketDistributor, g: GroceryStore | t in td.tickets and
        g.distributor = td implies t in g.tickets
}

//The ticket associated to a visit associated to a registered customer must be in that registered customer's
tickets
fact TicketForAVisitInRegisteredCustomersTickets{
        all rc: RegisteredCustomer, v: Visit | v in rc.visits iff v.ticket in rc.tickets
}

// A Customer cannot be associated simultaneously to 2 different queues
fact NoCustomerInMoreThanOneQueue{
        all q1, q2: Queue | q1 != q2 implies no (q1.customers & q2.customers)
}

// It cannot exist different visits booked by the same customer at the same time in the same day
fact NoTwoVisitAtTheSameHour{
        all rc: RegisteredCustomer, v1,v2: Visit  | (v1 != v2 and v1 in rc.visits and v2 in rc.visits and
        v1.schedule.day = v2.schedule.day
        and  v1.schedule.month = v2.schedule.month and v1.schedule.year = v2.schedule.year) implies
        v1.schedule.hour != v2.schedule.hour
}

// If a Customer has a ticket which status is "Issued" and that is not associated to a visit, it cannot exist another
ticket associated to that Customer that is not associated to a visit and which status is "Issued" or
"EntranceScanned"
fact OnlyOneIssuedTicketNotAssociatedToAVisit{
        all c: Customer, t1: Ticket, t2: Ticket | (t1 in c.tickets and t2 in c.tickets and t1 != t2
        and t1.status = Issued and (no v:Visit | t1 in v.ticket or t2 in v.ticket)) implies (t2.status != Issued and
        t2.status != EntranceScanned)
}

// If a Customer has a ticket which state is "EntranceScanned", it cannot exist another ticket associated to that
Customer that is not associated to a visit and which status is "Issued"
fact NoIssuedTicketNotAssociatedToVisitIfTickerInEntranceScanned{
        all rc: RegisteredCustomer, t1: Ticket, t2: Ticket | t1 != t2 and t1 in rc.tickets and t2 in rc.tickets and
        t1.status = EntranceScanned and (no v:Visit | v in rc.visits and t2 in v.ticket) implies t2.status != Issued
}
```

// If a Customer has a ticket which state is "EntranceScanned", it cannot exist another ticket associated to that Customer which state is "EntranceScanned"
fact OnlyOneEntranceScannedTicket{
      all c: Customer, t1: Ticket, t2: Ticket | (t1 in c.tickets and t2 in c.tickets and t1 != t2 and t1.status = EntranceScanned) implies (t2.status != EntranceScanned)
}

// A Customer is in the queue of Grocery Store g iff he/she has a ticket associated to the Grocery Store g which status is "Issued"
fact CustomerNotInQueueIWithoutIssuedTicket{
      all g:GroceryStore, q:Queue, c: Customer | (one t: Ticket | t.status = Issued and t in c.tickets and t in g.tickets) and g.queue = q iff (c in q.customers)

}

// It cannot exist a ticket associated to the grocery store g which state is "EntranceScanned" or "ExitScanned" which QR Code is not read by the QR Reader of the Grocery Store g
fact NoEntranceScannedTicketNotReadByQRReader{
      all t: Ticket, g: GroceryStore, qr:QRReader | (t.status = EntranceScanned or t.status = ExitScanned) and t in g.tickets and g.qr_reader = qr implies t in qr.tickets
}

// It cannot exist a ticket which status is "Expired" which QR Code is read by a QR Reader
fact NoExpiredTicketsReadByQRReader{
      all t: Ticket, q: QRReader | t.status = Expired implies (t not in q.tickets)
}

// It cannot exist a ticket that has simultaneously "distributor != none" and "visit != none" because a ticket issued by a ticket distributor cannot be associated to a visit
fact NoVisitTicketsEmittedFromDistributors{
      all t: Ticket, v: Visit | v.ticket = t implies (no td: TicketDistributor | t in td.tickets)
}

// If an Unregistered Customer is associated to a ticket, this ticket is not associated to a visit (because Unregistered Customers cannot book visits) and it's associated to the ticket distributor of the grocery store (because Unregistered Customers cannot get a ticket using the app)
fact NoVisitsForUnregisteredCustomers{
      all uc: UnregisteredCustomer, t: Ticket, g: GroceryStore, ts: TicketDistributor|
      t in uc.tickets and t in g.tickets and g.distributor = ts implies (no v:Visit | v.ticket = t) and t in ts.tickets
}

// If a ticket is issued by a ticket distributor belonging to a grocery store g, that ticket belongs to grocery store g. Viceversa is not true because a ticket for the grocery store g can be issued online and not from the ticket distributor

```
fact TicketDistributorOfStoreGEmitsTicketsForStoreG{
        all t: Ticket, d: TicketDistributor, g: GroceryStore | (t in d.tickets and g.distributor = d)
        implies t in g.tickets
}
```

// If QR Reader qr reads the QR Code of Ticket t, t and qr both belong to the same grocery store

```
fact TicketScannedByQRReaderOfItsStore{
        all t: Ticket, q: QRReader, g: GroceryStore | (t in g.tickets and (t.status = EntranceScanned or
        t.status = ExitScanned) and g.qr_reader = q) iff t in q.tickets
}
```

// A customer gets a ticket

```
pred IssuedTicket(c, c': Customer,  t: Ticket, qr: QRCode){
        //pre-condition
        t not in c.tickets
        // post-condition
        t.qr_code = qr
        t.status = Issued
        c'.tickets = c.tickets + t
}
```

// A registered custome books a visit

```
pred BookingVisit(c,c': RegisteredCustomer, v: Visit, t: Ticket, g: GroceryStore, s: Schedule){
        //pre-condition
        v not in c.visits
        //post-condition
        v.ticket = t
        v.grocery_store = g
        v.schedule = s
        c'.visits = c.visits + v
}
```

// A RegisteredCustomer cancels a tickets

```
pred CancelTicket(c,c': RegisteredCustomer, t: Ticket){
        //pre-condition
        t in c.tickets
        t.status = Issued
        //post-condition
        c'.tickets = c.tickets – t
}
```

```
// A registeredCustomer cancels a visit
pred CancelBookingVisit(c,c': RegisteredCustomer, v: Visit){
        //pre-condition
        v in c.visits
        v.ticket.status = Issued
        //post-condition
        c'.visits = c.visits – v
        c'.tickets = c.tickets - v.ticket
}

pred show{}

run show for 8
```

Proof of validity

Executing "Run IssuedTicket for 4"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  13554 vars. 556 primary vars. 30112 clauses. 110ms.
  Instance found. Predicate is consistent. 42ms.

Executing "Run BookingVisit for 4"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  13651 vars. 564 primary vars. 30338 clauses. 95ms.
 Instance found. Predicate is consistent. 69ms.

Executing "Run CancelTicket for 4"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  13494 vars. 552 primary vars. 29978 clauses. 94ms.
  Instance found. Predicate is consistent. 57ms.

Executing "Run CancelBookingVisit for 4"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  13570 vars. 552 primary vars. 30214 clauses. 97ms.
  Instance found. Predicate is consistent. 56ms.

Example of the world generated
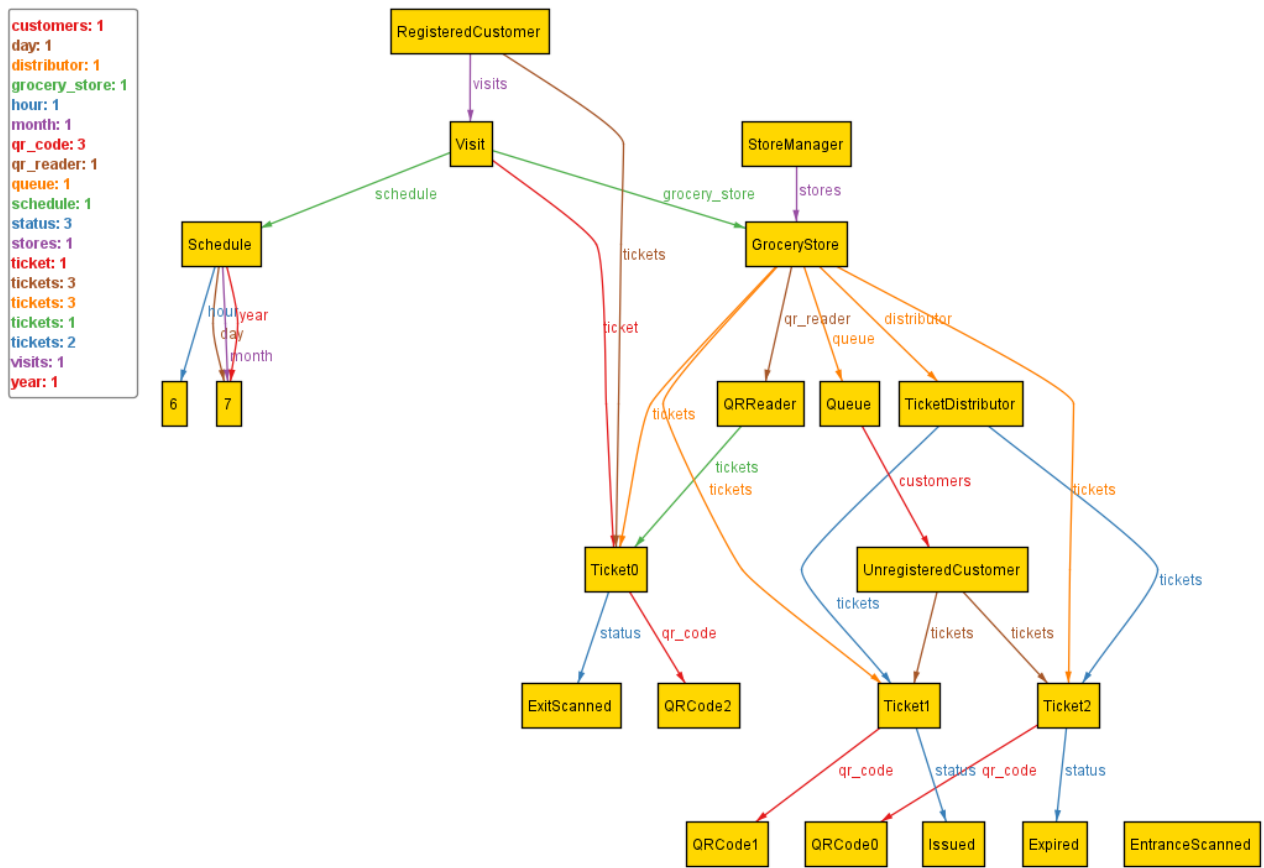
5. Effort Spent

Claudio Alfredo Emanuele

Discussion about the problem	**4h**
Purpose & Scope	**2h**
World and Shared Phenomena	**1h**
Definitions, Acronyms, Abbreviations, Document Structure	**2h**
UML Description	**3h**
User characteristics	**1.5h**
Domain Assumptions	**2h**
Requirements	**3h**
Mapping	**1h**
Use Cases	**2h**
Sequence Diagrams	**6h**
Performance Requirements	**2h**
Alloy	**10h**
Review	**3h**

Antonio Guadagno

Discussion about the problem	**4h**
Purpose & Scope	**2h**
Goals	**2h**
State Charts	**1h**
Product functions	**2h**
Domain Assumptions	**1.5h**
External Interface Requirements	**3h**
Requirements	**3h**
Mapping	**1h**
Use Cases & Use Cases Diagrams	**6h**
Sequence Diagrams	**2h**
Scenarios	**2h**
Alloy	**10h**
Review	**3h**

6. References
Course slides

Wikipedia