CLup

Customers Line-Up

RASD

Requirement Analysis and Specification Document

Software Engineering II Project
Computer Science and Engineering
Politecnico di Milano
Version 1.1 - Date

POLITECNICO
MILANO 1863

Davide Merli - 10578363
Dario Passarello - 10566467

Deliverable:	RASD
Title:	Requirement Analysis and Verification Document
Authors:	Davide Merli, Dario Passarello
Version:	1.1
Date:	23-December-2020
Download page:	https://github.com/davidemerli/MerliPassarello/
Copyright:	

Contents

List of Figures

List of Tables

1 Introduction

1.1 Purpose

1.1.1 Problem Analysis

Due to the recent coronavirus global pandemic, many of the human activities have been drastically affected and restricted by the need to contain the virus diffusion. Norms and regulations may vary from contry to country, but almost everywhere the main focus is to mantain social distancing to avoid diffusion from one person to another. One of the most difficult activity to fullfil (yet absolutely essential) is grocery shopping.

Stores are forced to restrain access to avoid too many people inside the building, and this produces endless lines out of the stores. This can both increase the danger for people waiting for their turn and force the shops to regulate customers even outside the structure.

CLup aims to reduce heavily the issues involving customer queues outside of stores by permitting clients to keep track of their position in the store queue or book in a visit in advance with an easy to use application.

On the other hand, CLup also will provide a simple (but very customizable and scalable) system for the stores to integrate the queue monitoring tools into their everyday workflow, without the absolute need to buy expensive hardware.

Moreover, CLup will take in consideration the fact that not every customer has the same familiarity with today's technology. In addiction to a straightforward interface and UX, the system will permit the stores to hand out tickets on the spot. It should be also possible to improve the overall experience in line even to the customers that do not use the CLup application, thanks to the monitoring tools that will be made available.

CLup has the goal to ease the struggles for customers and store workers and save a lot of time that would have been lost waiting for hours in queues. This is an important improvement, even in times not as trying as the current coronavirus emergency. A well integrated system taking advantage of the CLup service can even boost sales: clients are more willing to go shopping if they know they won't be loosing time; store owners will also have the possibility to check and profit from statistics about user entrances and average shopping time.

1.1.2 Document Purpose

The purpose of this document is to analyze the problem taking in consideration the real needs of the customers and shop workers. This RASD describes in detail the functional and non-functional requirements of the S2B and includes exhaustive descriptions about typical use cases from the actors that will take part of the system. Hence, this document is addressed to the developers of the S2B as well as the companies that want to integrate CLup services into their workflow.

1.1.3 Goals

Table 1: Goals table

Label	Goal Description
G1	Avoid exceeding the maximum number of customer inside the store in each store
G2	Reduce the number of customer waiting physically in line in front of the store entrance
G3	Try to distribute people uniformly inside the store to ease maintaining social distancing
G4	Allow customers to book for a visit to the store at a desired time and day
G5	Let every customer have the possibility to access the store regardless of the technology available to them
G6	Give stores adopting CLup access to anonymous statistics regarding the people coming to the store
G7	Provide a simple and user-friendly interface to book tickets
G8	Provide an interface for the access controller to check tickets and monitor the occupancy
G9	Provide an estimation of the waiting time to every customer that waits in line (physical or virtual)
G10	Notify a customer when it's time to approach the store entrance

1.2 Scope

CLup positions itself as an intermediary between stores and customers. Clients can book entrances and retrieve tickets through the application, and the stores communicate with the CLup backend to update entrances, leavings and the capacity of the building.

The mobile application can be used by customers to monitor their time in queue, get notifications when they should approach the entrance and get a time estimation before it's their turn to enter; it can also be used by store employees to manually update the store live information.

Furthermore, CLup will provide a simple but powerful REST API that can be easily exploited to automate completely the process of updating live store information.

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Acronyms

- **S2B**: Software To Be
- **RASD**: Requirement Analysis and Specifications Documents
- **REST**: REpresentational State Transfer
- **API**: Application Programming Interface
- **UX**: User Experience
- **UI**: User Interface
- **SSO**: Single sign-on
- **QR code**: Quick Response code
- **OS**: Operating System
- **RAM**: Random Access Memory
- **LAN**: Local Area Network
- **GPS**: Global Positioning System
- **GB**: GigaByte
- **TCP/IP**: Transmission Control Protocol/Internet Protocol
- **HTTPS**: Hypertext Transfer Protocol Secure
- **IoT**: Internet of Things
- **MQTT**: Message Queuing Telemetry Transport
- **RAID**: Redundant Array of Independent Disks
- **UML**: Unified Modeling Language

1.3.2 Definitions

- **Access controller**: a subsystem that permits the entrance of customers into the store. It can be a device like a smart turnstile that reads customers tickets or just a person of the store staff manually scanning tickets.

- **Business account**: a CLup account that is destined to store managers or operators and therefore the 'business' side of CLup

- **In-Site ticket**: a ticket that is taken by a customer near the store. It can be both a virtual paperless ticket or a ticket printer by an emitter near the store premises

- **Virtual ticket**: a ticket issued through the CLup application

- **Physical/Paper ticket**: a printed physical ticket, emitted by a printer near the store premises

- **Valid Ticket** (at time X): a ticket that has a code recognized by the CLup system and valid for the specified time

- **Time slot**: a time delta that is associated with a number of bookable tickets (which varies and is customizable from store to store)

- **People-Counting System**: a subsystem that permits the counting of the number of people inside the store. It can comprehend a device like a proximity sensor or a smart turnstile, or it can be a person of the store staff manually counting people.

- **Customer Application**: the CLup mobile application destined to customers that want to shop inside stores adopting CLup

- **Operator Application**: the CLup mobile application destined to store staff to monitor entrances and statistics

- **Store Main System**: the store main server that communicates directly with CLup servers. All store subsystems and smart devices should communicate with it through an Intranet

- **Geocoding API**: Geocoding converts addresses into geographic coordinates to be placed on a map. A Geocoding API allows the use of their services to permit translation between textual addresses and Latitude/Longitude coordinates

- **Map API**: An external services that provides operations of geographics maps and the download of map information, usually of the places in proximity of given geographics coordinates

- **Hashed Password**: When a password has been "hashed" it means it has been turned into a scrambled representation of itself. A user's password is taken and – using a key known to the site – the hash value is derived from the combination of both the password and the key, using a set algorithm.

1.4 Revision History

- 1.0: First Revision
- 1.1: Fixed Some Incoherences in the Requirements numbering

1.5 Reference Documents

- The World and the Machine - Michael Jackson - ICSE95: 17th International Conference on Software Engineering Seattle Washington USA April, 1995
- R&DD Assignment A.Y. 2020-2021 - Elisabetta di Nitto, Matteo Giovanni Rossi
- Software Engineering II slides and material - Elisabetta di Nitto, Matteo Giovanni Rossi
- ISO/IEC/IEEE 29148:2018 Systems and software engineering - Life cycle processes - Requirements engineering
- Queueing Theory And Modeling - Linda Green - Graduate School of Business, Columbia University, New York

1.6 Document structure

Chapter 1 introduces the problems that the S2B will resolve framing them into the current world situation. It describes purpose of the project and the goals that the software will need to accomplish. Moreover, it reports reference documents and describes the lexicon used in the whole document (i.e. Definitions, Acronyms and Abbreviations).
Chapter 2 lays out an in-depth analysis of the functionalities that will be provided by the S2B starting from the examination of real world phenomena, user characteristics and details about demographics differences on the customers side.
In detail are shown:

- specific description of likely scenarios of customers everyday life and their relation with CLup
- functional and non-functional requirements of the S2B
- domain assumptions about the environment in which the system behaves
- a formal representation of the whole system

Chapter 3 examines specific requirements considered together with current technologies, hardware, systems and existing interfaces. It provides insights about CLup relationship with the external world from a technology point of view. It lists the constraints that needs to be respected, and delineates system attributes like Reliability, Availability, Security, Maintainability and Portability
Common use cases are meticulously described, divided into smaller actions and highlighting the actors taking place in them. Sequence Diagrams are provided for every use case in order to further explain and visualize them.
Requirements are mapped with the goals they accomplish, together with the domain assumptions they are related to. This improves the understanding of how the work can later be divided and parallelized, and will define the development process.
Chapter 4 provides an abstract static Alloy model for the S2B. In this section the goals of the modeling activity are explained. The commented Alloy code is provided to the reader, this code is further explained with some instances generated from the alloy engine.
Chapter 5 shows the time spent from each member of the team for writing this document.

2 Overall Description

2.1 Product Perspective

2.1.1 User scenarios

In this section are shown some scenarios in order to present some in real life settings and explain how CLup fits in and how it behaves in different situations.

Scenario 1: The Jet Market
With the "Stay home" ordeal due to COVID-19 the "Jet Market", a mid sized grocery store in the city of Springfield experienced a sudden increase of customers per day. At the same time its not too big capacity was halved due to social distancing measures. These events lead to the formation of long queues of people out of the store premises in peak hours.

The store manager, in order to improve the service quality, decided to adopt the CLup queueing system. Upon signing the contract he received a business account, and set up the shop systems with the help of CLup advisors and technicians. In the very first days with CLup the queue length did not change much, but thanks to CLup adverts exposed outside the shop, some people downloaded the application.

In the following days more and more people noticing that CLup allowed to skip the line by booking the visit beforehand, downloaded the application too.

After some weeks the "Jet Market" can reserve up to the 60 % of the store capacity to CLup booked customers. The remaining capacity is filled with non CLup customer. The "Jet Market" manager is very happy because the queues are a lot shorter and his customers started to distribute their visit more uniformly during the day.

Scenario 2: Bartolomeo
Bartolomeo is an old grandfather. Due to his age he's not very familiar with the modern technology and he doesn't have a smartphone. His favorite supermarket decided recently to adopt CLup, as long queues were forming outside the store.

Before CLup adoption Bartolomeo waited up to one hour to enter the supermarket on peak times. After the installation, for customers not confident with the technologies used by CLup, like Bartolomeo, things did not become more complex. Bartolomeo has only to retrieve a paper ticket at the entrance and wait until his number is called. If there are a lot of people in queue before him, he could take a sit on the bench at the park near the supermarket and return in time for when his number is called. An estimation of the entrance hour is printed on every ticket.

Scenario 3: Slightly late
Marcello booked the 5:45 PM slot at the supermarket near his workplace. This slot allows him to enter his chosen market from 5:45 PM to 6:00 PM without having to wait in line.

He normally ends his workday at 5:30 PM, but his colleague asked him some help just before him leaving. He arrived at the store late and scanned the ticket at 6:05 PM, just two minutes after the 3 minute tolerance (configured by the store using the CLup configuration panel).

The CLup system, considering that there are no people waiting in line and that the store is not full, lets Marcello enter anyway. The application then sends him a notification

reminding him that he could have possibily lost his priority, but saves him from the struggle of having to create an In-Site ticket and scan the code another time.

Scenario 4: Paperless
For various reasons Adriana could not plan in advance when she could go grocery shopping at her favorite store this week.

On Thursday morning she was going to work by car, and received a call from her boss telling her that the scheduled meeting was cancelled. She had her morning free and decided to go for grocery shopping since she was already out of the house.

Even if she didn't book her entrance in advance, she used CLup to check how crowded the supermarket was. As she arrived near the supermarket, she created a ticket from the CLup application to enter as soon as possible. This kind of ticket is equivalent to the one printed by the emitter out of the store but it's paperless; moreover, the application notifies about live changes in the estimated entrance time. To avoid staying in front of the entrance Adriana waits in her car until CLup notifies her. Some minutes later, she receives a notification on her phone that she should approach the entrance because in a short time she will enter the supermarket.

Scenario 5: No-show
Luigi booked an entrance for the 4:00 - 4:15 PM slot last Saturday. He was busy cleaning the house and didn't pay attention to the time neither to his smartphone. At 4:30 PM he picks up the phone and notices CLup notifications about approaching the entrance, and even about that his picked time slot to enter had expired.

Since he forgot to cancel his booking, at 4:20 PM the late tolerance time ended. The store at the time is unusually full and some people are waiting outside. CLup system allowed one more person to enter the store replacing Luigi's reservation.

2.1.2 The world and the machine

In this section the system will be described using "The World and The Machine" model proposed by M. Jackson. At a coarse grain CLup is composed of a central system (the machine) which relates to components that allow shared phenomena to happen between the world and the machine.

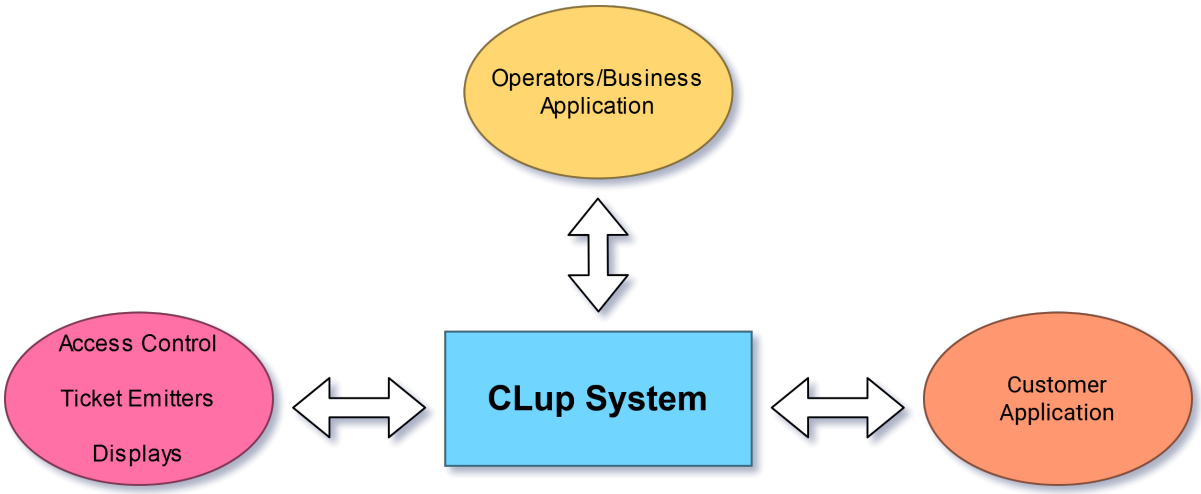


Figure 1: "The Machine" and the components interfacing with "The world"

2.1.3 Shared phenomena

Here is a list of shared phenomena happening in CLup

Table 2: Shared Phenomena table

Label	Phenomena	Controlled by
SP1	The gate opens and lets the customer inside the store	Machine
SP2	The customer scans his ticket at the gate	World
SP3	The customer leaves the store and their exit is counted by a People-Counting System	World
SP4	The customer books a ticket to enter the shop with the Customer Application	World
SP5	The customer receives a notification when it's almost his turn to enter the store	Machine
SP6	The customer presses the button to request a paper ticket	World
SP7	The customer receives a ticket from the ticket printer and an estimated waiting time	Machine
SP8	The display at the entrance shows which ticket numbers are allowed to enter and the other numbers in queue	Machine

2.1.4 Interfacing with external systems

The S2B will expose some interfaces to communicate with external systems used to provide data to CLup.
Physical ticket emitters should be placed outside the store near the entrance, to allow users that do not have CLup installed on their smartphones to wait for their entrance. The ticket emitter prints tickets on a paper support when a button is pressed. The emitter is connected to the store main system and via a dedicated API request retrieves information for a new valid ticket. The response will contain the information needed to print the ticket (e.g. the ticket number/QR code and the estimated time before entering). The queue statistics in the system are then updated accordingly.
Optionally, the emitter can show on a small screen the current waiting time estimation, so customers can know in advance if they have the possibility to wait for the displayed time.
A **Smart gate** is a composed of a mechanical actuator that opens and closes the gate and a QR code scanner. It is connected to the store main system. When a QR code is scanned and decoded, a request is sent to the CLup API with the ticket information stored in the code. The system checks the validity of the ticket and sends a response to the gate. If the response is affirmative the gate will open. In case the scanned QR doesn't represent a valid ticket, an error will be shown to the customer via visual and/or auditive feedback. Even if the gate replaces a big part of access-control staff's work (see next paragraph), some human intervention could be required for security reasons, or to open the gate in exceptional cases.
People counters located at shop exits, monitor people passing through them. They are connected to the store network and will communicate the count of people that left the shop. This counter can be a turnstile or a sensor; it could even be another smart gate that scans the same tickets used to enter (or a barcode printed on the receipt) before letting people out of the store.
CLup system should works fine without them, but these counters allow to have precise data about people inside the shop, and this can be useful to estimate more precisely the waiting times. In particular, CLup can calculate and store the shopping time of customers if it's given the possibility to associate the time of entrance with the time of leaving of the said customer. This could, for example, be done through the scan of the same QR code for both entering and leaving.
Ticket number screens are located at shop entrances. They display the latest batch of ticket numbers allowed to enter the store, and the ordered list of the other numbers in line.

2.1.5 Interfacing with users

Access-control operators have a portable device equipped with a camera and internet connectivity. This device has installed the staff access-control app. The user interface for operators provides these functionalities:

- Authentication through a CLup operator account or corporation SSO service
- Checking (and adjusting) estimated occupancy of the whole shop and its departments
- Checking length of the queues, and bookings in the time slots
- Scan a ticket QR code to manually allow a person to enter
- Notify the next customers in the virtual ticket queue alerting them to reach the entrance

The **CLup Customer Application** allows some operations even to unauthenticated users, though only logged-in ones can book tickets for In-Site queues or for a desired time slot.

- Every user can:
  - Create a new account
  - Login to an existing account
  - Check which nearby stores are adopting CLup system (on a map) and an option for listing them sorted by distance
  - See real-time and projected occupancy of all stores
- An authenticated user can also:
  - Create a virtual ticket to enter a store immediately (if space is available) or join a queue and get update notifications
  - Check store occupancy in future time slots
  - Book a visit to a selected store in an available time slot

2.2 User Characteristics

There are two macro-groups of people that will use :, the businesses adopting the system and the customers of these business using CLup to plan their visits to the shop.

2.2.1 Business-side Characteristics

CLup is addressed mainly to big grocery shops, but could be used from every medium to big sized shop. CLup is very flexible, and will work adapting to a lot of different scenarios, thanks to these features:

- Parameters of the system are tunable: The business is able to customize booking time-slots duration and capacity
- Different access controls methods can be employed: the shop can install smart gates with a QR scanner, or delegate an employee to manually check the tickets with the CLup operator application.
- Different precision levels of estimated data are accessible based on the data sources available. For example, if the shop counts the number of people exiting the premises (using a turnstile or a QR scanner), CLup can provide the exact live occupancy of the store. If this is not possible, the live occupancy will be estimated based on the average permanence time in the shop.

2.2.2 Customer side characteristics

Everyone needs to do grocery shopping, but in the situation of a pandemic, it is encouraged to stay safe at home. This can lead people to go for longer shopping sessions to last more days without going out; it can also drastically increase the number of customers per day at store, inevitably creating long queues. Nobody should be excluded from accessing the shops, and also people without CLup application need to be taken into account.
The S2B solves this problem by allowing those customers to retrieve paper tickets and wait in a physical line before entering the shop if the shop is full. The physical line of customer is unavoidable if the store cannot satisfy the increased demand of groceries in its area, even if CLup can alleviate this problem allowing a better distribution of the client visits. The possibility of booking visits in advance will push more and more users to download CLup in order to avoid queues; avoiding cramming at the entrance and maintaining social distancing can decrease the spread of the virus.

2.3 Assumptions and dependencies

2.3.1 Domain Assumptions

Table 3: Domain Assumptions table

Label	Domain Assumption
DA1	Customers that created a shoplist will buy approximately all the products in that list, so they will visit for the greater part of their permanence the departments where the products are located
DA2	Customer will stay approximately the time they have declared when booking the ticket
DA3	The access-control system works properly and won't allow unauthorized customer entrances
DA4	Customers with a numbered digital ticket try to avoid staying near the entrance until they receive the "go to entrance" notification
DA5	Customers with a booked ticket in a given time slot will not show up until few minutes before the start of their time slots
DA6	If people counters are installed they will provide the exact count of the customers that enter or leave the shop
DA7	No customer are present at the shop opening hour, and no customer will be present at the shop closing hour
DA8	The store manager will insert correct data about the shop and the departments maximum capacities

2.4 Product Functions

In this section are presented the main functions of the S2B, introduced by the state machines representing the two main scenarios for the customers.

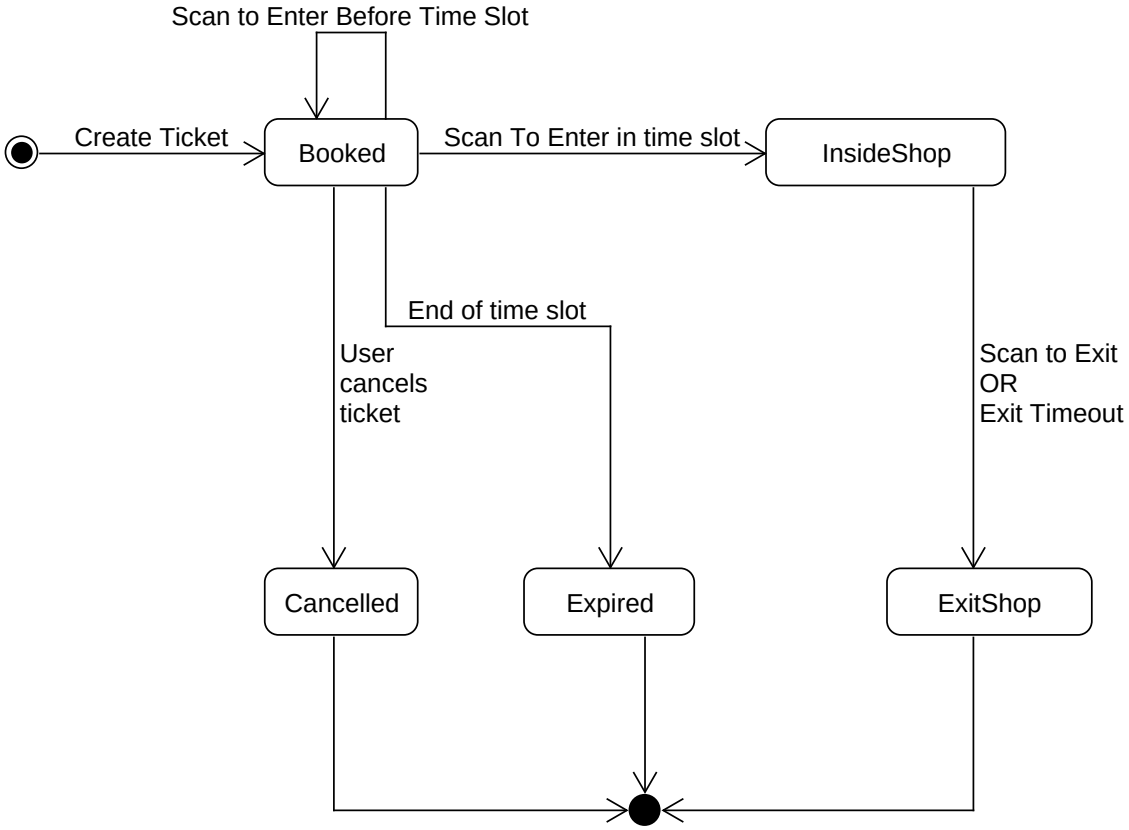


Figure 2: Booked ticket state machine

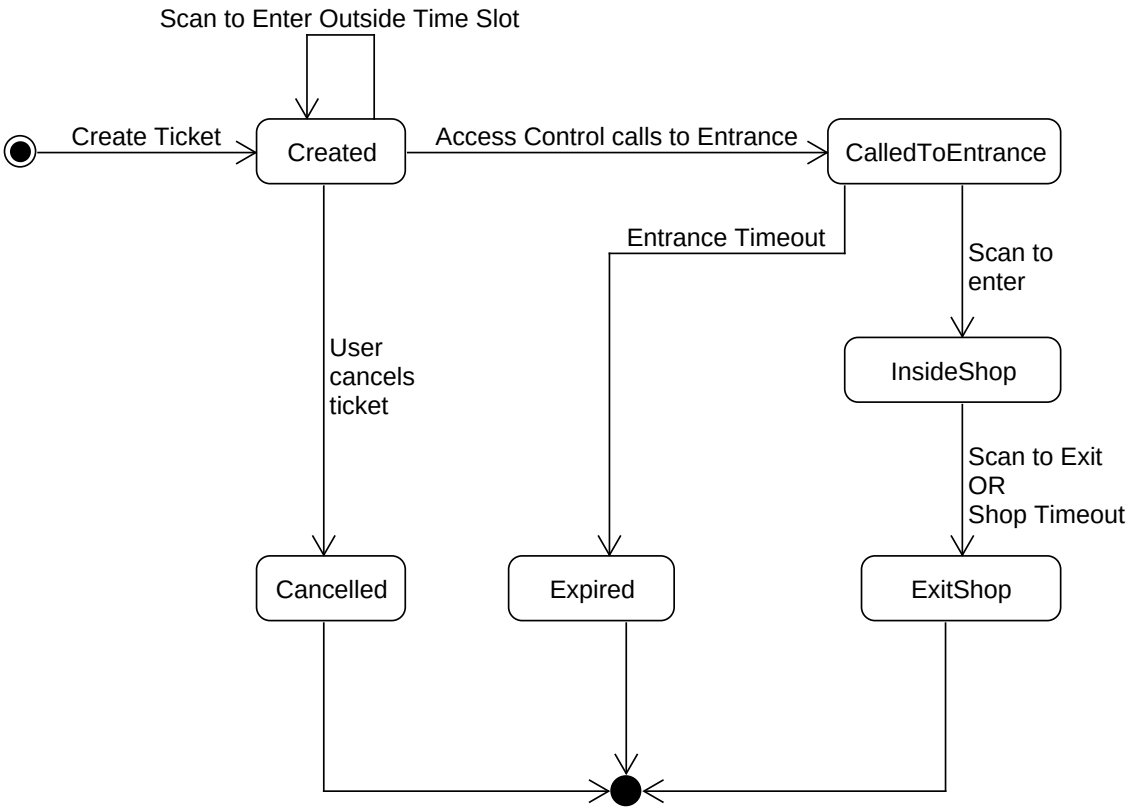


Figure 3: In-place virtual ticket state machine

Keeping in mind this two state machines it is provided a list of requirements for the CLup system. Every interaction should be taken into account and implemented in the most simple and user friendly way. The effort put into realizing the best user experience should be priorized towards the customers that will use the application. Stores adopting the CLup system should instruct their workers on how to use the application; customers should receive the most immediate and intuitive UX as possible.
Requirements list both functional and non-functional requirements, and span every aspect the system will have to implement and respect to achieve all the previously described goals. Further in the document, goals will be mapped with the requirements needed to fullfil them, to provide a convenient reference useful to understand how much work will be needed to accomplish goals and how to divide teams for development.

2.4.1 System Requirements

Table 4: Requirements table

Label	Requirement description
R1	The system must keep general information and contacts about the store chains adopting CLup
R2	The system must provide each store a store-admin account
R3	The store-admin account must allow the creation of store-operator accounts
R4	For each store the system must allow the users to retrieve information about location and business hours
R5	The system stores information about capacity of each market
R6	The system won't let anyone enter the store if the maximum capacity has been reached
R7	The system will let a customer enter the store if and only if they have a a valid ticket
R8	The system will use the occupancy data retrieved from the store to control the store access
R9	The system must provide an interface to communicate with the store access control
R10	The system must provide an interface for user to compile a shopping list
R11	The system must take in consideration shopping list data and historic data from previous user visits to reduce store crowdedness per department
R12	The system must allow the store-admin account to create and edit entrance time intervals
R13	Each time interval must have a number of bookable slots fewer than the store capacity
R14	The system must allow authenticated users to book a visit in a desired time interval
R15	The system must not allow a user to book a slot in an already full time interval
R16	The system must not allow a user to book a visit if he has already reserved another visit
R17	The system must allow a customer to create a numbered virtual queue ticket and notify them if he can enter immediately (if the store is not full) or provide them a waiting time estimation
R18	The system must notify the customers with a virtual queue ticket when it's time to approach the store entrance
R19	The store operator application must allow an authenticated operator to manually admit customers

Label	Requirement description
R20	The system must ask the customer to provide the estimated visit time when booking a time slot
R21	The system must allow stores to hand out numbered physical queueing tickets to those that do not use the CLup application
R22	The system must allow the access to the store to customer with numbered tickets using a 'First Come First Served' logic, treating virtual and physical ticket owner equally
R23	The system must try to estimate waiting time based on store capacity, reservations and the current number of people with numbered tickets waiting in line
R24	The system should interface with an screen placed at the entrance of the store to notify customer which ticket numbers will enter in the next called batch
R25	The system must let the store-admin accounts retrieve statistics collected from CLup regarding their store
R26	The system must record periodically and store statistics about the occupancy of each store
R27	The customer CLup application must show brief statistics about average occupancy of each stores during different days of the week
R28	The operator CLup application must show to an authenticated operator the real time occupancy of the store
R29	The customer CLup application must be cross-platform and must work on the majority of the devices
R30	The stores adopting CLup must be displayed on a map
R31	The CLup customer application allows user to mark stores as favorite in order to access them quickly
R32	The CLup customer app after the login allows immediately to book tickets right from the homepage
R33	The system must provide an interface for automated control devices to communicate to CLup data about store entrances, store leavings and crowdedness in the various departments
R34	The system must push notifications to user devices with update information on the store he has a ticket for
R35	The system must associates tickets with line numbers

2.4.2 System Representation

Use cases and scenarios, elaborated and expanded into functional and non-functional requirements for the software, require a structure defining all the entities involved in the system. Below it is presented the CLup class diagram, with interaction between entities and the cardinality of such associations.

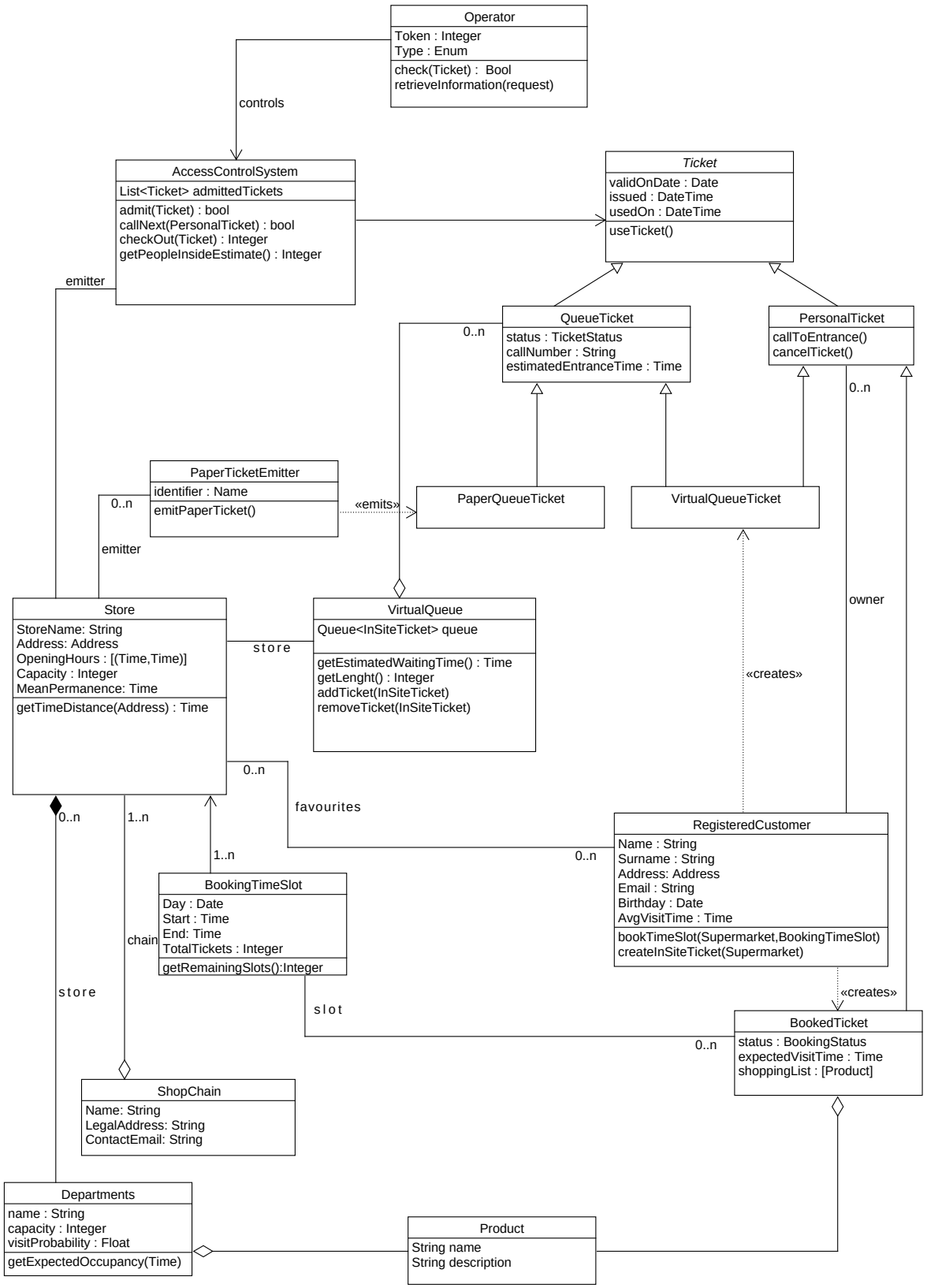


Figure 4: Class Diagram

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The user application has to be developed, as stated numerous times, for every kind of demographics. This includes visually impaired users and people not familiar with modern technologies.
Hence, the interface must:

1. remain simple on every page and menu
2. follow standards in design to accomodate users into finding which are the interactible parts of the application; for example Material Design by Google and/or Apple iOS design standards could be applied
3. communicate via short and concise messages
4. show important information without having to navigate multiple times into sub-menus
5. integrate correctly with OS screen reader technologies to be usable by blind people
6. use an adaptive layout to support devices with different resolutions and aspect ratios

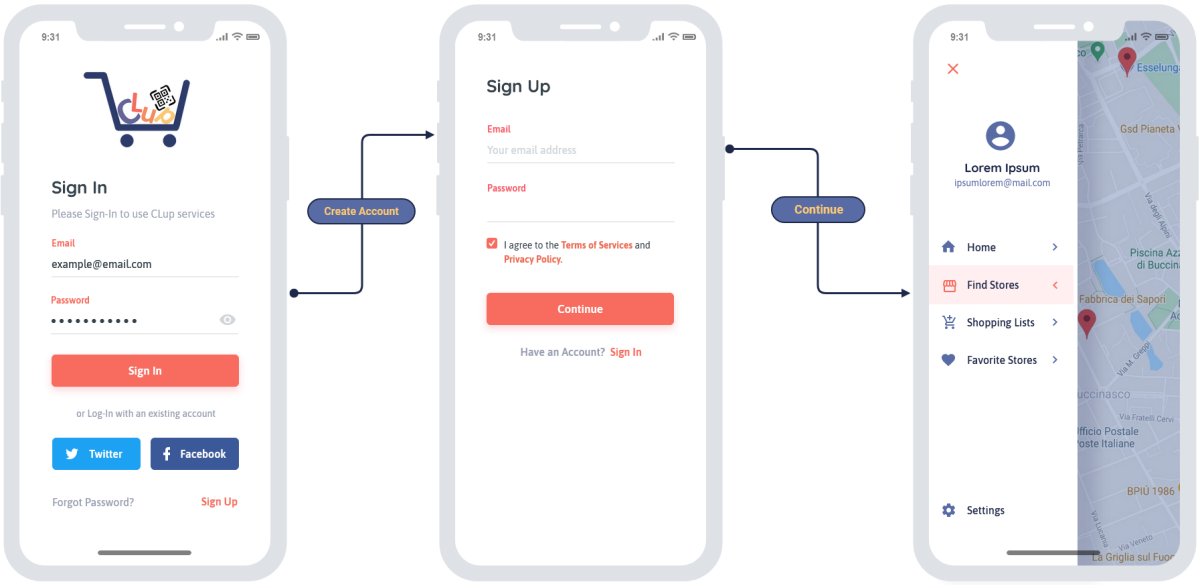


Figure 5: CLup's Customer Application Mockups

3.1.2 Hardware Interfaces

CLup software runs on different platforms and different kinds of hardware depending on the part of the system services it is required to run.
To run the CLup mobile application it is required a device with mobile internet access (e.g. a smartphone or a tablet) and at least 1GB of ram (to be able to not experience

lags due to memory used by other applications). Most of today's smartphones start with 2GB of RAM anyway. GPS is used to retrieve the current location but can be replaced with manual insertion of the needed data by the user.
Automation systems communicating in a LAN do not have specific requirements: many different solutions can be employed and can vary from different kind of store. The store main system must be able to communicate with the CLup APIs, and it is required to have a working internet connection.
CLup main servers on the other hand can exploit newest standards on Serverless computing, modularizing functions in microservices and be able to keep up with an increase of requests and stores adopting the S2B.

3.1.3 Software Interfaces

CLup will expose REST APIs in order to exchange data from a local store server to CLup main servers.
Automation systems will communicate with the local store server without any constraints, even though standard REST APIs for internal information exchange will also be provided by the S2B.

3.1.4 Communication Interfaces

Connections from remote to local servers will be done over TCP/IP and HTTPS. CLup REST API's will require valid tokens to verify authentication.
Local subsystems interfacing with the local main server should be connected to a private Intranet to lower the risk of possible cracking attempts.
Local subsystems can also be developed and connected with IoT standards, for example employing communication through protocols like MQTT.

3.2 Functional Requirements

3.2.1 Use cases

Here is a list of relevant use cases for the S2B. These use cases are also shown in Figure 6

Label	Use Case
UC1	Customer Registration
UC2	Customer/Operator Authentication
UC3	Customer search for the store page
UC4	Customer adds/removes a store from their favorites list
UC5	Customer books a visit in a store
UC6	Customer creates/edits a shopping list
UC7	Customer creates a numbered virtual ticket to enter a store as soon as there is a place available
UC8	Customer creates a physical numbered ticket
UC9	Customer cancels a previously created ticket
UC10	Customer scans the ticket through an access control system to enter
UC11	An user leaves the store through an exit with a people counter installed
UC12	A store operator checks statistics about the store
UC13	User resets his password

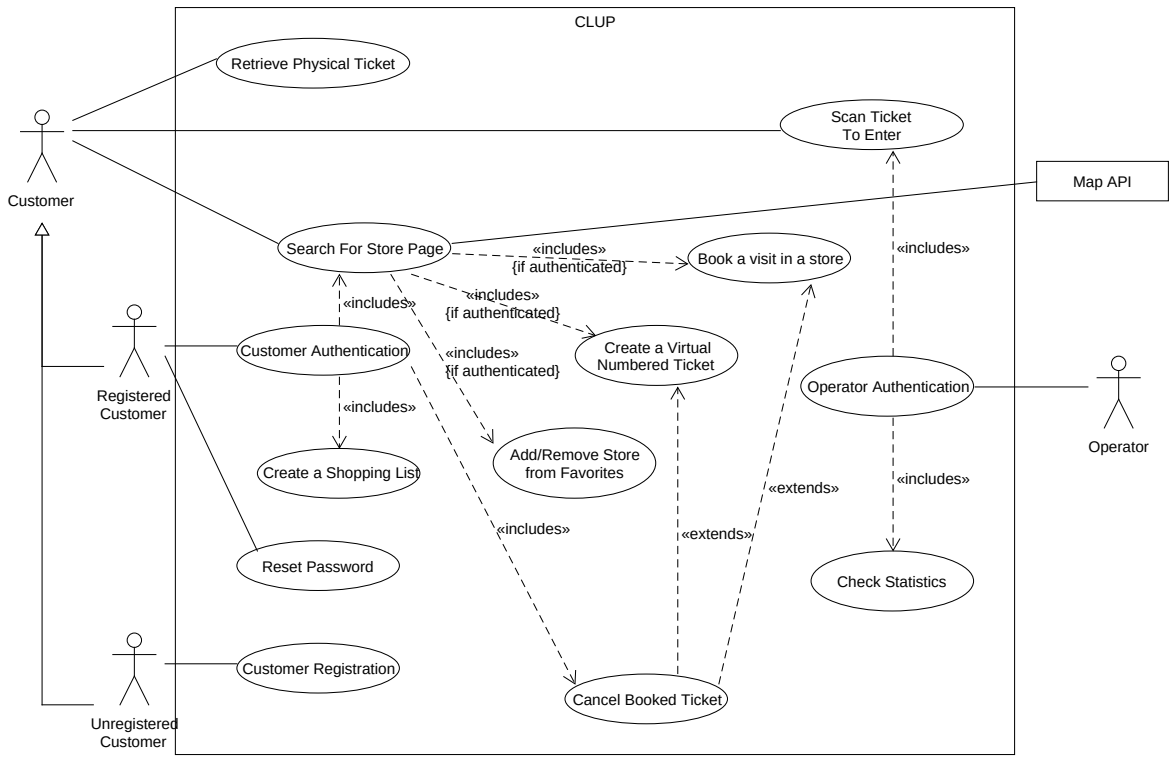


Figure 6: Use Case Diagram for the S2B

In the following use case description tables, if not explicitly stated otherwise, the term "user" is used interchangeably with the term "customer".

Use case 1: User Registration

Name	**User Registration**
Actors	Unregistered customer
Entry conditions	The user requests the system to register
Flow of events	1. The system shows the form with the required fields to register 2. The user inserts his e-mail address and his password 3. The user inserts his name, surname, birth date and his preferred home address 4. The user is shown the recap of the information used to fill the form 5. The user confirms the information, and the form is sent to the system 6. The system sends a e-mail to the address provided by the user in the form. The e-mail contains a verification link 7. The users open the e-mail and clicks on the verification link 8. The system sends an e-mail to the user stating that their registration process ended successfully
Exit conditions	The unregistered customer now is a registered customer and after authentication he can access all CLup customer functionalities
Exceptions	• If the e-mail inserted is already registered in the system an error message tells the user that the provided e-mail is already in use by another account. The systems asks the user if they want to reset the password (See UC13 - User resets password) • While typing the password the system checks that the password is compliant with the password requirements listed in the section 3.6.1. If the password requirements are not met the system shows an error listing the non respected password requirements.

Table 5: *Customer registration* use case description

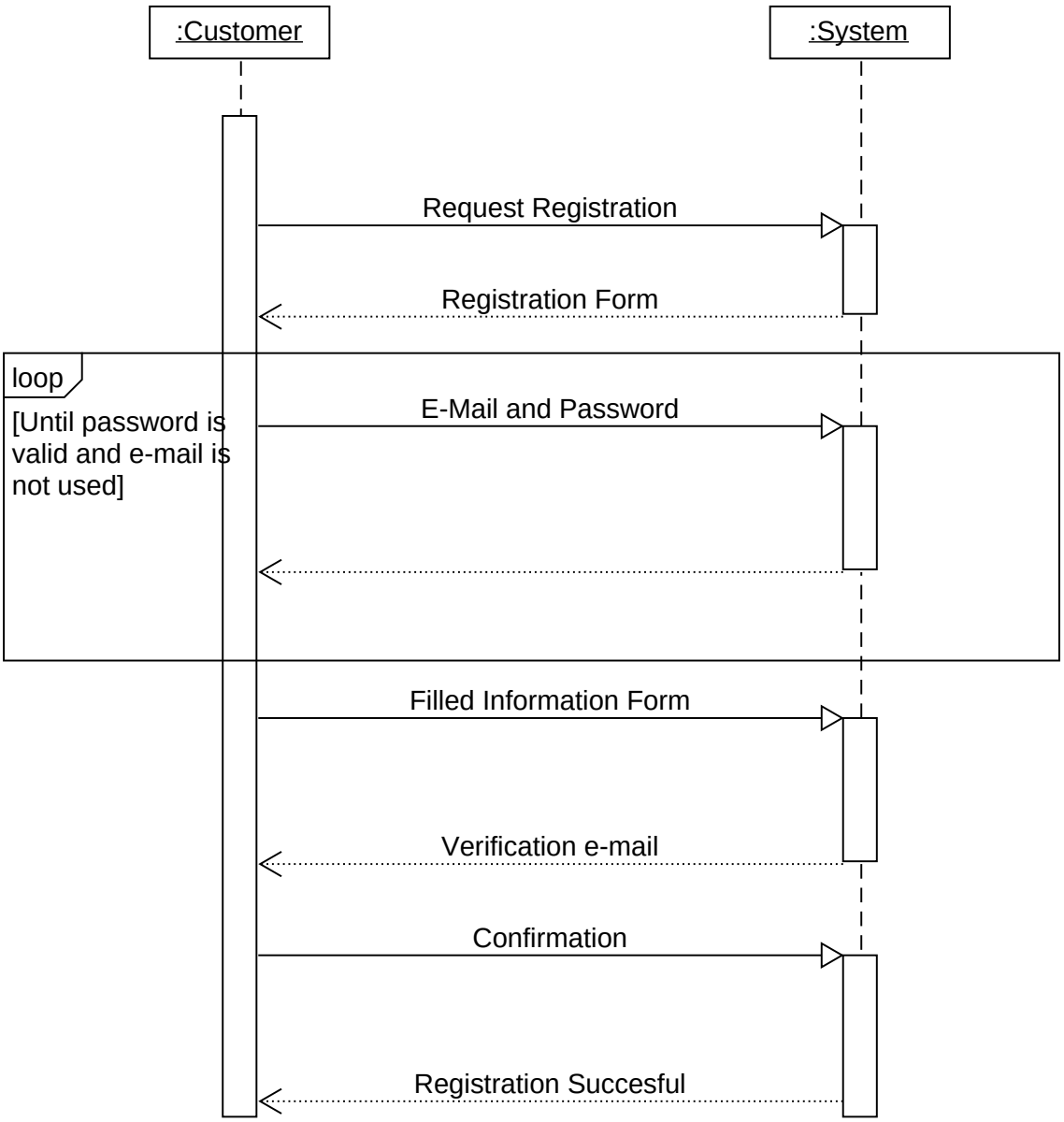


Figure 7: Sequence Diagram for Use Case 1

Use case 2: User Authentication

Name	**User Authentication**
Actors	Registered Customer or Store Operator
Entry conditions	The user requests the system to log-in
Flow of events	1. The system shows the authentication form 2. The user fills the form with his e-mail address used to register and his password 3. The system checks if there is an account registered with the provided e-mail 4. The system checks that the account's hashed password matches with the hashed password provided in the form 5. The initial CLup page is shown to the user
Exit conditions	The user can use the CLup customer/operator application with all the functionalities available to authenticated users
Exceptions	• If the e-mail or the password are incorrect then an error message is shown to the user

Table 6: *Customer authentication* use case description

Use case 3: Customer searches for store details

Name	**Customer searches for store details**
Actors	Customer
Entry conditions	The user started the CLup customer application
Flow of events	1. The system checks the position of the user with the GPS 2. The system interfaces with an external map API downloading from it a map of the surroundings of the user position 3. The system decorates the map with the positions of all store adopting CLup 4. The system sends the map to the user 5. The user applies filters on the store list 6. The system updates the map displaying only the stores complying with the filter 7. The user selects one store 8. The system retrieves all the details about the store from his databases 9. The system loads and displays the store view
Exit conditions	The customer now views the store pages and can: • **book a visit in that store** • **create a numbered ticket for entering the store as soon as possible** • **add the store to their favorites list**

Exceptions	• If the GPS position is not available or the user doesn't want to provide it to the system, the system will center the map view at the user's home address specified in the registration, making a call to a Geocoding API to retrieve home coordinates • If the Geocoding API call fails or returns no result the map will be centered to the last known user position • If the other fallback option fails and the Map API is available the map will be shown centered in a default position • If the Map API call fails default list view containing all CLup stores will be shown to the user • If no store matches the user filters an error message will be prompted to the user
Additional Requirements	• The store details shown in the store view are: store name, store chain, address, opening hour, occupancy statistics at every hour • The filters criteria available to the user are: Store name, city, currently open stores, maximum distance from the store

Table 7: *Customer searches for store details* use case description

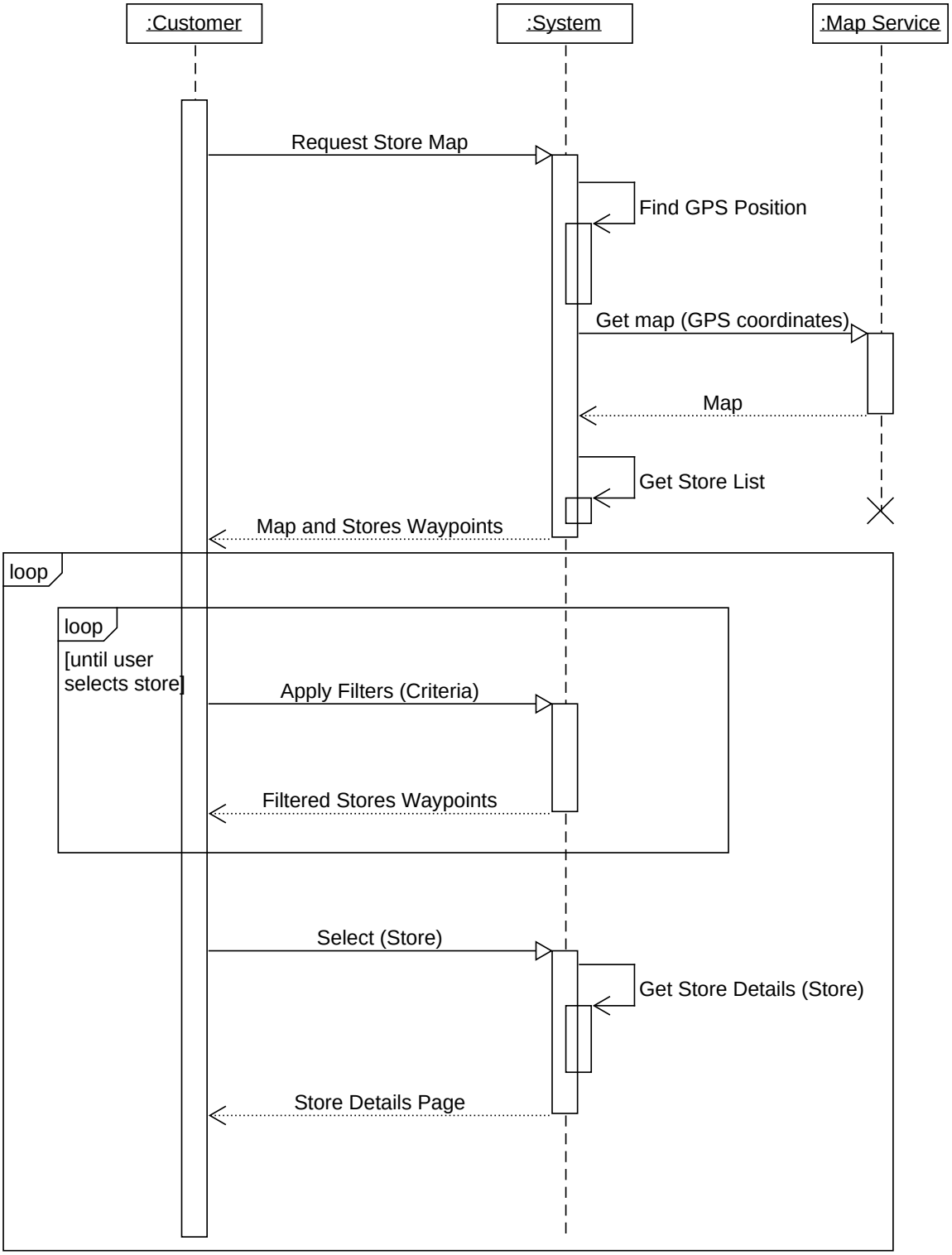


Figure 8: Sequence Diagram for Use Case 3

Use case 4: Customer adds/removes a store from their favorites list

Name	**Customer adds/removes a store to their favorites list**
Actors	Registered customer
Entry conditions	The user is authenticated and has loaded a store view
Flow of events	1. The user clicks on the button to add/remove the store from the favorites 2. If the store is on the user's favorites list the store is removed from that list; else the store is added to that list
Exit conditions	The user sees the change on their favourite list
Exceptions	

Table 8: *Customer adds/removes a store to their favorites list* use case description

Use case 5: Customer books a visit in a store

Name	**Customer books a visit in a store**
Actors	Registered customer
Entry conditions	The user is authenticated and has loaded a store view. The user has no other visits or ticket active
Flow of events	1. The user starts the procedure selecting the option to book a visit from the store page 2. The user selects a date 3. The system retrieves the time slots with at least one free bookable slot and shows them to the user 4. The user selects his preferred slots sending the information to the System 5. The systems checks if the slot is currently available, asks the user to select the expected duration of his visit, showing to them a default value 6. The user (optionally) edits the value and then confirm the booking 7. The system confirms the booking of the visit 8. The system asks the customer if they want to **create a shopping list**
Exit conditions	The user can enter the store during the booked time-slot
Exceptions	If the selected slot is not available anymore, the system will show an error message inviting the customer to select another available slot

Table 9: *Customer books a visit in a store* use case description

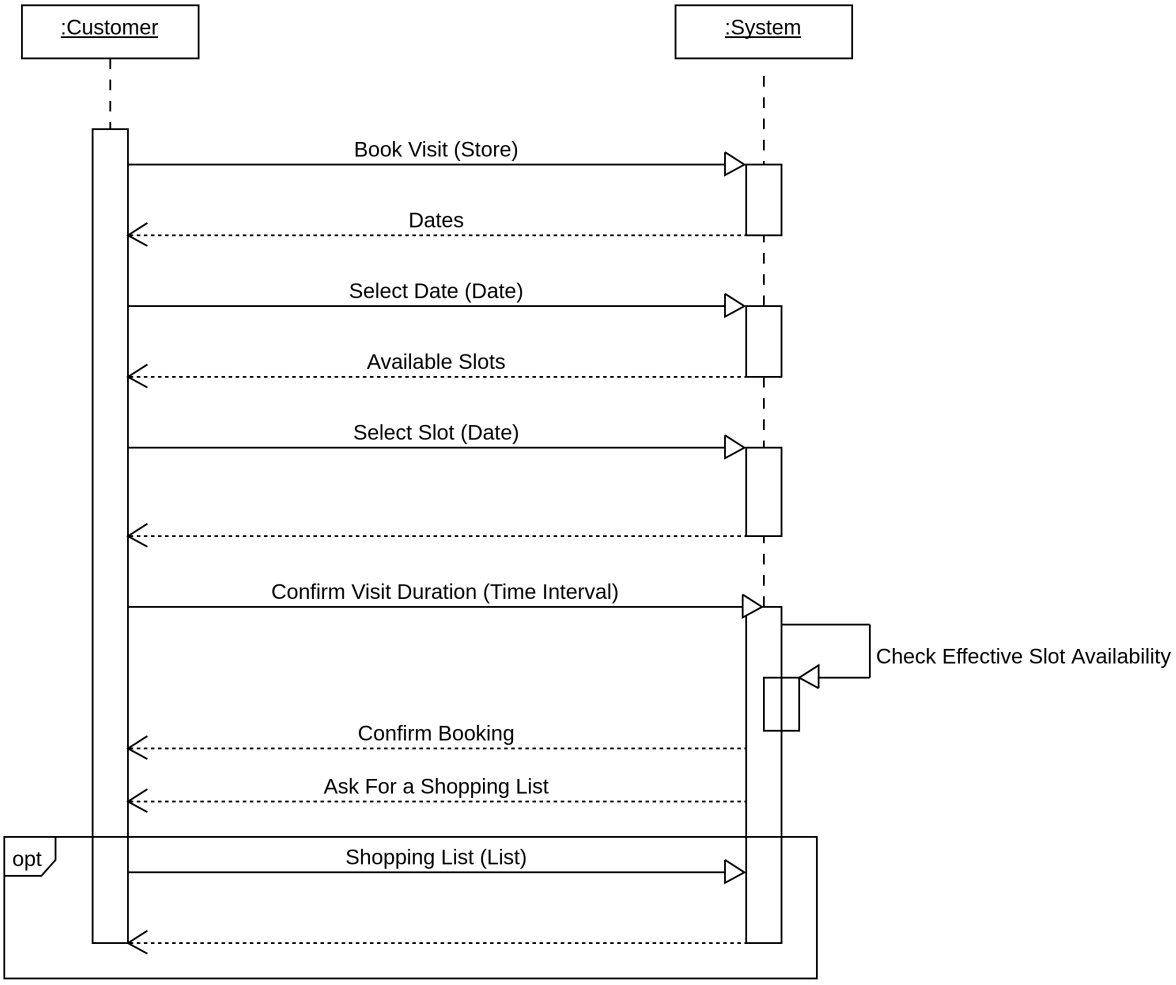
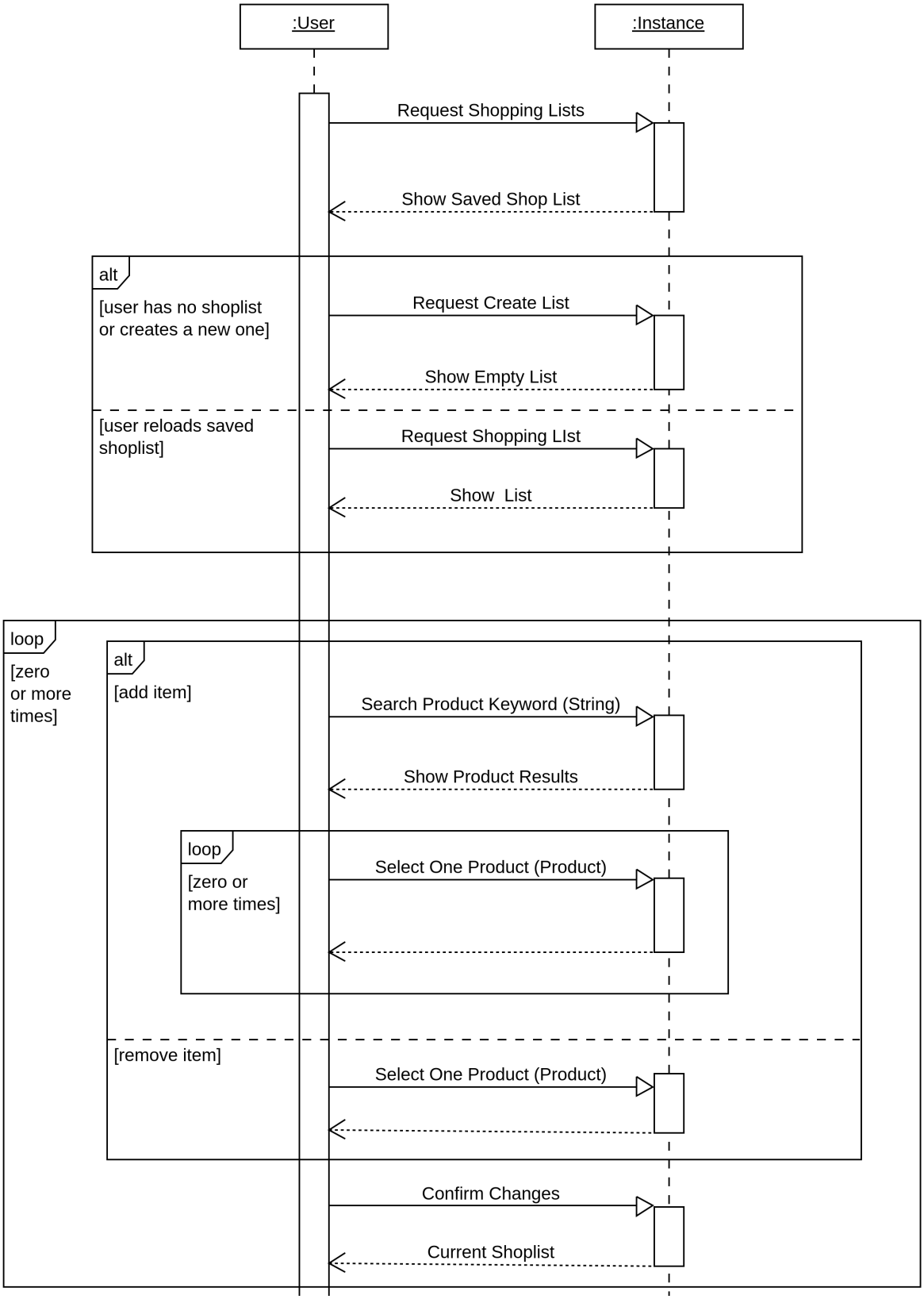


Figure 9: Sequence Diagram for Use Case 5

Use case 6: Customer creates/edits a shopping list

Name	**Customer creates/edits a shopping list**
Actors	Registered customer
Entry conditions	The user is authenticated
Flow of events	1. The user reaches the shopping list view 2. If the user added a shopping list before, the system asks the user if they want to edit the old list or if they want to start from a blank one 3. The user inputs a search key in a search bar 4. The system searches products matching the key and shows them ordered by closeness to the search key 5. The user choses zero or more than one product from the search results and adds to the shopping list 6. The user views the complete shopping list and can continue to add or remove product until they confirm the list. 7. When the user confirms the list, the system will save it in order to make it available to the user the next time they want to compile a shopping list 8. If the user has booked a visit the system can update the estimates on store departments occupancy on the time of the visit thanks the data on the shopping list 9. The system acknowledges the user that the list has been saved
Exit conditions	
Exceptions	

Table 10: *Customer creates/edits a shopping list* use case description


:User
:Instance
Request Shopping Lists
Show Saved Shop List
alt
[user has no shoplist
or creates a new one]
Request Create List
Show Empty List
[user reloads saved
shoplist]
Request Shopping LIst
Show List
loop
[zero
or more
times]
alt
[add item]
Search Product Keyword (String)
Show Product Results
loop
[zero or
more times]
Select One Product (Product)
[remove item]
Select One Product (Product)
Confirm Changes
Current Shoplist

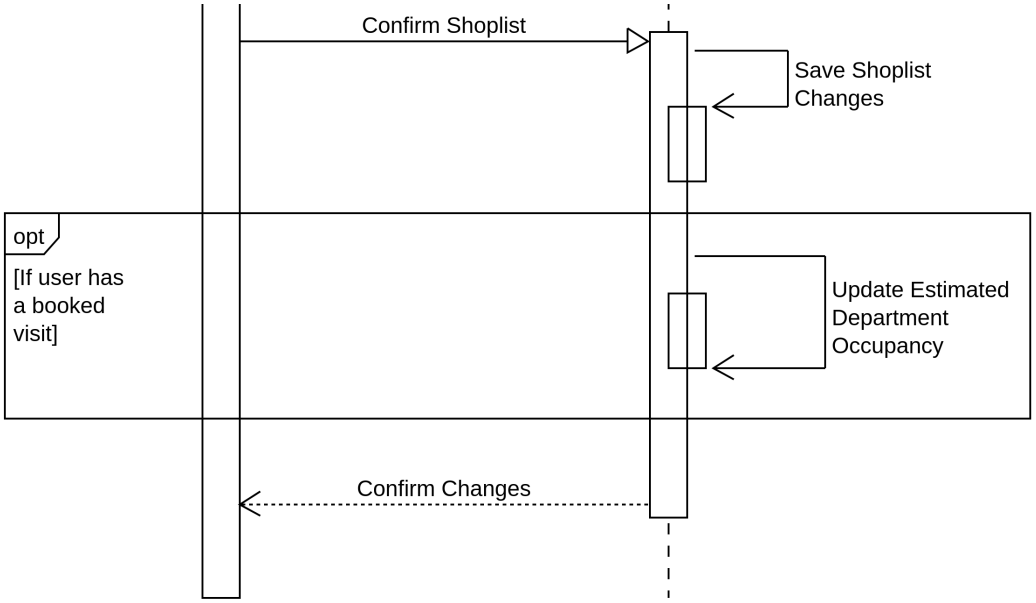


Figure 10: Sequence Diagram for Use Case 6

Use case 7: Customer creates a numbered virtual ticket to enter a store as soon as there is place available

Name	**Customer create a numbered virtual ticket to enter a store as soon as there is place available**
Actors	Registered customer
Entry conditions	The user is authenticated and has loaded a store view. The user has no other visits or ticket active
Flow of events	1. The user starts the procedure selecting the option to retrieve a numbered ticket from the store page 2. The system estimates the waiting time based on the number of people queued and the people actually inside the store 3. The system confirms the emission of the ticket to the user and shows them the number and other ticket details 4. If the waiting time changes in a significant way the system notifies the user with the new waiting time 5. The system alerts the user when it's time to approach the entrance
Exit conditions	The user goes to the entrance and scans the ticket
Exceptions	• If the waiting time is long the system shows an alert to the user asking if they are sure of creating the ticket despite the long queue • If the user takes too long to enter the store after being called to enter an alert is shown stating that the entry is no more guaranteed with that ticket

Table 11: *Customer creates a numbered ticket to enter a store as soon as there is place available* use case description

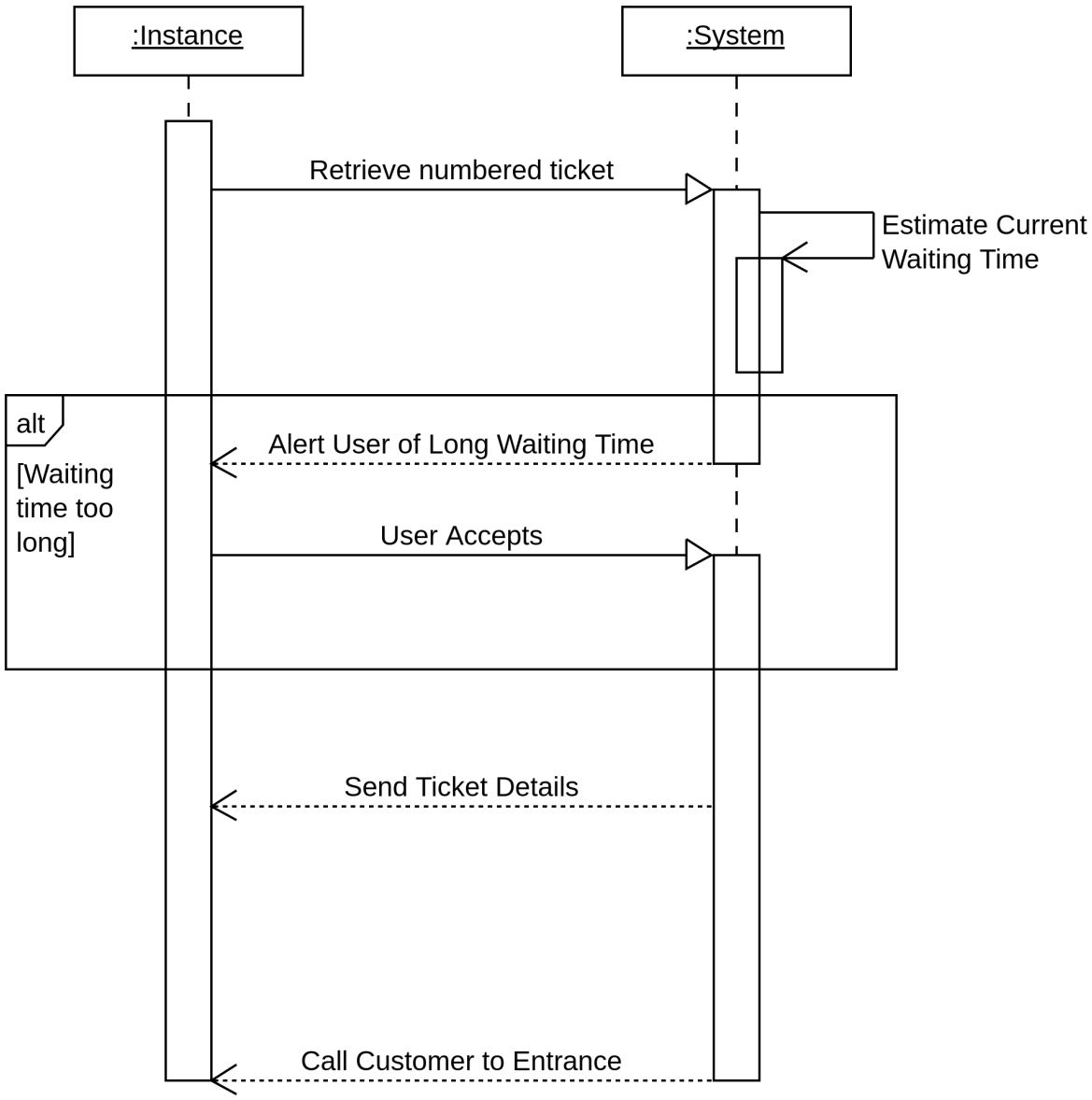


Figure 11: Sequence Diagram for Use Case 7

Use case 8: Customer create a numbered physical ticket to enter the store

Name	**Customer create a numbered physical ticket to enter the store**
Actors	Customer, Physical Ticket Emitter
Entry conditions	The customer is in front of the ticket emitter
Flow of events	1. The user selects a button to create a new ticket 2. The emitter request the system to create a new numbered ticket 3. The system estimates the waiting time based on the number of people queued and the people actually inside the store 4. The system returns the details of the ticket to the emitter 5. The emitter prints the ticket 6. The user retrieves the ticket and waits until the number on his ticket is called at the entrance of the store
Exit conditions	The user enters the store scanning the ticket
Exceptions	If the user takes too long to enter the store after being called to enter, the ticket no more guarantees the entrance to the store

Table 12: *Customer create a numbered physical ticket to enter the store* use case description

Use case 9: Customer cancels a previously created ticket

Name	**Customer cancels a previously created ticket**
Actors	Registered Customer
Entry conditions	The customer has a visit to a store booked or a numbered virtual ticket previously created
Flow of events	1. The customer from the virtual ticket/booking view selects the option to cancel the ticket/booking 2. The system deletes the item from the database and triggers a reevaluation of the estimations about the occupancy of the store and the departments 3. The system acknowledges the user that the operation was successful
Exit conditions	The user can now create another ticket or book another visit
Exceptions	

Table 13: *Customer cancels a previously created ticket* use case description

Use case 10: Customer scans the ticket on an access control system to enter the store

Name	**Customer scans the ticket on an access control system to enter the store**
Actors	Customer, Access Control System
Entry conditions	The customer has a ticket or a visit reservation
Flow of events	1. The user scans the machine readable identifier on the ticket at the Access Control System 2. The access control system sends the scanned ticket identifier to the system 3. The system determines if the customer can enter, retrieving information about the ticket and the current occupancy of the supermarket 4. The system sends a confirmation to the Access Control System 5. The Access Control System lets the customer enter
Exit conditions	The user is inside the store
Exceptions	• If the Access Control System fails to recognize the ticket identifer, the user will be asked to scan again the ticket • If the ticket is not valid, the Access Control System won't let the customer enter. This decision can be overridden by an human operator if the system is manned • If the supermarket occupancy is over the legal capacity, the access control won't let the customer enter even with a valid ticket. The system will ask the customer to wait and will consider the booking of the current time slot valid also after the expiration • If the Access Control System fails to communicate with the system, the customer is asked to scan again the ticket and if fails again asks the customer to notify the store personnel (only if the system is not manned).

Additional Requirements	• A booked ticket is valid only in the booked date within the time slot indicated in the ticket. • A numbered ticket is valid only for a limited time after the ticket number has been called to the entrance. • All the tickets are bound to a specific store and could be used only on that store. • The store can allow customer to enter with an expired ticket if the store is far from being full.

Table 14: *Customer scans the ticket in an access control system to enter the store* use case description

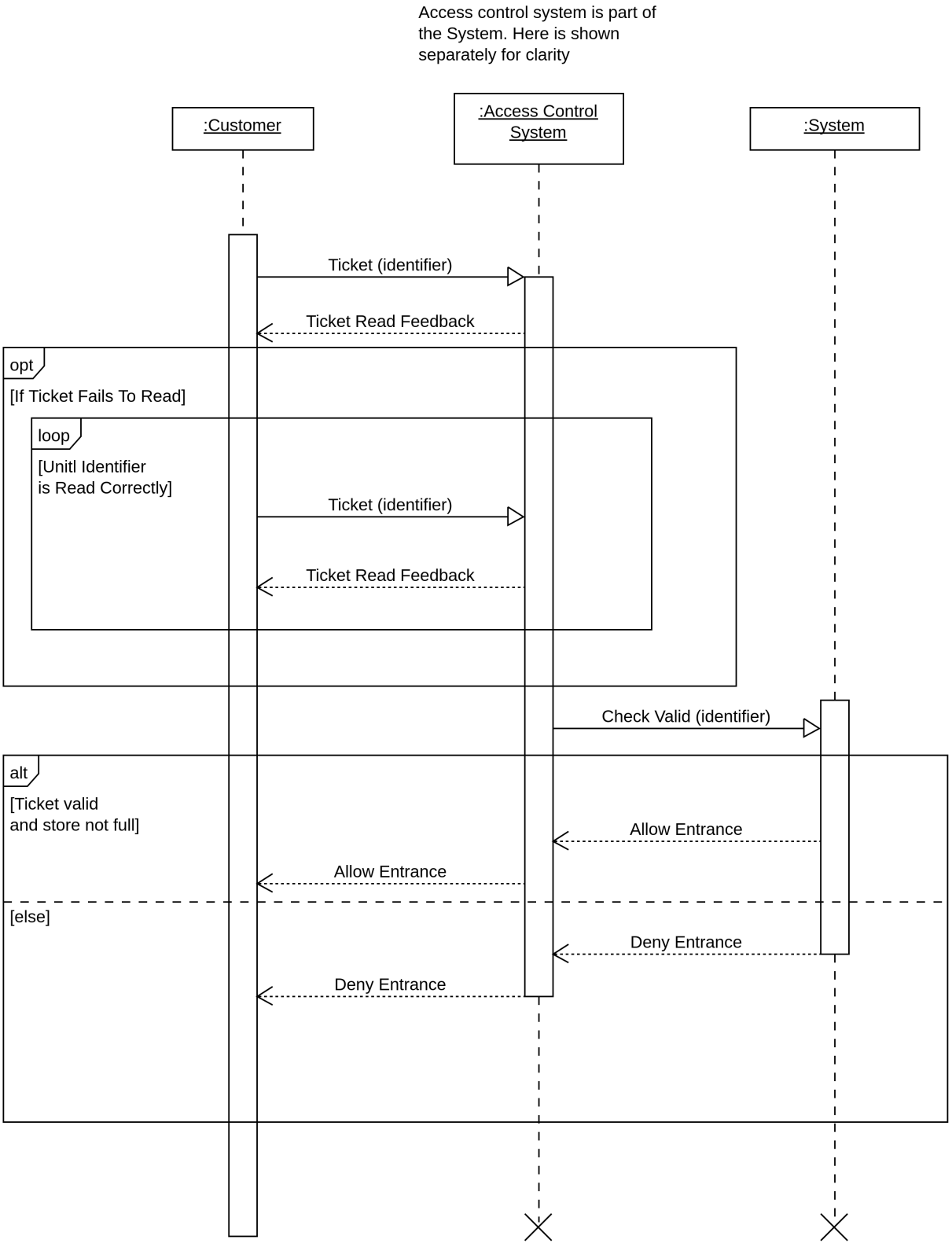


Figure 12: Sequence Diagram for Use Case 10

Use case 11: An user leaves the store through an exit with a people counter installed

Name	**An user leaves the store through an exit with a people counter installed**
Actors	Customer, People Counter
Entry conditions	The customer is inside the store
Flow of events	1. The customer walks out of the store through a gate with a people counter installed 2. The people counter registers that a person passed through the gate 3. The people counter contacts the system, updating the occupancy real time information
Exit conditions	The customer is out of the store and the count of the people of the store is updated
Exceptions	

Table 15: *An user exits the store through an exit with a people counter installed* use case description

Use case 12: A store operator checks statistics about the store

Name	**A store operator checks statistics about the store**
Actors	Store Operator/ Store Manager
Entry conditions	The Store Operator/Store Manager is authenticated
Flow of events	1. The store operator requires to see statistics or real time information about the store collected by CLup 2. The system retrieves or compute that information 3. The system present the data to the Operator/Manager
Exit conditions	The Operator/System can now see the data presented by the System
Exceptions	The system throws an error to the operator if he has no privilege to see the requested information

Table 16: *A store operator checks statistics about the store* use case description

Use case 13: User resets the password

Name	**A store operator checks statistics about the store**
Actors	Registered Customer/ Store Manager
Entry conditions	The user requests a password reset
Flow of events	1. The user inputs his e-mail 2. The system checks that the e-mail corresponds to a registered account 3. The system sends an e-mail to the user with a temporary link to the reset password page 4. The user opens the link and inputs his new password 5. The user confirms his new password 6. The system applies the changes and redirects the user to the proper login page
Exit conditions	The password of the user account is changed
Exceptions	• If the e-mail inserted from the user is not associated with any of the accounts an error message is shown to the user. • While typing the password the system checks that the password is compliant with the password requirements listed in the section 3.6.1. If the password requirements are not met the system shows an error listing the non respected password requirements.

Table 17: *User resets the password* use case description

3.2.2 Requirements Mapping

In this sections each goal is mapped to the functional requirements and the domain assumptions that combined satisfy that goal. Each goal also includes the use cases that practically demonstrate part of the goal and/or its related requirements.

G1: Avoid exceeding the maximum number of customer inside the store in each store

- R1 The system must keep general information and contacts about the store chains adopting CLup
- R2 The system must provide each store a store-admin account
- R3 The store-admin account must allow the creation of store-operator accounts
- R4 For each store the system must allow the users to retrieve information about location and business hours
- R5 The system stores information about capacity of each market
- R6 The system won't let anyone enter the store if the maximum capacity has been reached
- R7 The system will let a customer enter the store if and only if he has a a valid ticket
- R8 The system will use the occupancy data retrieved from the store to control the store access
- R9 The system must provide an interface to communicate with the store access control
- DA2 Customer will stay approximately the time they have declared when booking the ticket
- DA3 The access controller works properly and won't allow unauthorized customers entrances
- DA6 If people counters are installed they will provide the exact count of the customers that enter or leave the shop
- DA7 No customer are present at the shop opening hour, and no customer will be present at the shop closing hour
- DA8 The store manager will insert correct data about the shop and the departments maximum capacities
- UC10 Customer scans the ticket through an access control system to enter
- UC11 An user leaves the store through an exit with a people counter installed

G2: Reduce the number of customer waiting physically in line in front of the store entrance

- R12 The system must allow the store-admin account to create and edit entrance time intervals
- R13 Each time interval must have a number of bookable slots fewer than the store capacity
- R14 The system must allow authenticated users to book a visit in a desired time interval
- R17 The system must allow a customer to create a numbered virtual queue ticket and notify them if he can enter immediately (if the store is not full) or provide them a waiting time estimation
- R18 The system must notify the customers with a virtual queue ticket when it's time to approach the store entrance
- R19 The store operator application must allow an authenticated Notify a customer with the CLup app a reasonably precise estimation of the waiting time
- DA4 Customers with a numbered digital ticket try to avoid staying near the entrance until they receive the "go to entrance" notification
- DA5 Customers with a booked ticket in a given time slot will not show up until few minutes before the start of their time slots
- UC5 Customer books a visit in a store
- UC7 Customer creates a numbered virtual ticket to enter a store as soon as there is a place available

G3: Try to distribute people uniformly inside the store to ease maintaining social distancing

- R10 The system must provide an interface for user to compile a shopping list
- R11 The system must take in consideration shopping list data and historic data from previous user visits to reduce store crowdedness per department
- R36 The system allows customer to register an user account
- R37 The system allows registered customers to authenticated
- DA1 Customers that created a shop list will buy approximately all the products in that list, so they will visit for the greater part of their permanence the departments where the products are located
- DA2 Customer will stay approximately the time they have declared when booking the ticket
- UC6 Customer creates/edits a shopping list

G4: Allow customers to book for a visit to the store at a desired time and day

- R12 The system must allow the store-admin account to create and edit entrance time intervals
- R13 Each time interval must have a number of bookable slots fewer than the store capacity
- R14 The system must allow authenticated users to book a visit in a desired time interval
- R15 The system must not allow a user to book a slot in an already full time interval
- R16 The system must not allow a user to book a visit if he has already reserved another visit
- R20 The system must ask the customer to provide the estimated visit time when booking a visit to the store
- R30 The stores adopting CLup must be displayed on a map
- R32 The CLup customer app allows to book visit and retrieve tickets directly from the store page
- R36 The system allows customer to register an user account
- R37 The system allows registered customers to authenticated
- DA7 No customer are present at the shop opening hour, and no customer will be present at the shop closing hour
- DA8 The store manager will insert correct data about the shop and the departments maximum capacities
- UC1 Customer Registration
- UC2 Customer/Operator Authentication
- UC3 Customer search for the store page
- UC5 Customer books a visit in a store
- UC9 Customer cancels a previously created ticket
- UC10 Customer scans the ticket through an access control system to enter
- UC13 User resets his password

G5: Let every customer have the possibility to access the store regardless of the technology available to them

- R21 The system must allow stores to hand out numbered physical queueing tickets to those that do not use the CLup application
- R22 The system must allow the access to the store to customer with numbered tickets using a 'First Come First Served' logic, treating virtual and physical tickets owner equally
- R23 The system must try to estimate waiting time based on store capacity, reservations and the current number of people with numbered tickets waiting in line
- R24 The system should interface with an screen placed at the entrance of the store to notify customer which ticket numbers will enter in the next called batch
- R29 The customer CLup application must be cross-platform and must work on the majority of the devices
- UC8 Customer creates a physical numbered ticket
- UC10 Customer scans the ticket through an access control system to enter

G6: Give stores adopting CLup access to anonymous statistics regarding the people coming to the store

- R2 The system must provide each store a store-admin account
- R3 The store-admin account must allow the creation of store-operator accounts
- R25 The system must let the store-admin accounts retrieve statistics collected from CLup regarding their store
- R26 The system must record periodically and store statistics about the occupancy of each store
- R27 The customer CLup application must show brief statistics about average occupancy of each stores during different days of the week
- R28 The operator CLup application must show to an authenticated operator the real time occupancy of the stor
- DA8 The store manager will insert correct data about the shop and the departments maximum capacities
- UC2 Customer/Operator Authentication
- UC12 A store operator checks statistics about the store

G7: Provide a simple and user-friendly interface to book tickets

- R4 For each store the system must allow the users to retrieve information about location and business hours
- R29 The customer CLup application must be cross-platform and must work on the majority of the device
- R30 The stores adopting CLup must be displayed on a map
- R31 The CLup customer application allows user to mark stores as favorite in order to access them quickly
- R32 The CLup customer app allows to book visit and retrieve tickets directly from the store page
- R34 The system must push notifications to user devices with update information on the store he has a ticket for
- R36 The system allows customer to register an user account
- R37 The system allows registered customers to authenticated

- UC1 Customer Registration
- UC2 Customer/Operator Authentication
- UC3 Customer search for the store page
- UC4 Customer adds/removes a store from their favorites list

G8: Provide an interface for the access controller to check tickets and monitor the occupancy

- R2 The system must provide each store a store-admin account
- R3 The store-admin account must allow the creation of store-operator accounts
- R8 The system must provide an interface to communicate with the store access control
- R9 The system must provide an interface to communicate with the store access control
- R19 The store operator application must allow an authenticated operator to manually admit customers
- R23 The system must try to estimate waiting time based on store capacity,reservations and the current number of people with numbered tickets waiting in line
- R28 The operator CLup application must show to an authenticated operator the real time occupancy of the store
- R33 The system must provide an interface for automated control devices to communicate to CLup data about store entrances, store leavings and crowdedness in the various departments
- DA3 The access-control system works properly and won't allow unauthorized customer entrances
- DA6 If people counters are installed they will provide the exact count of the customers that enter or leave the shop
- DA8 The store manager will insert correct data about the shop and the departments maximum capacities
- UC10 Customer scans the ticket through an access control system to enter
- UC12 A store operator checks statistics about the store

G9: Provide an estimation of the waiting time to every customer that waits in line (physical or virtual)

- R8 The system will use the occupancy data retrieved from the store to control the store access
- R17 The system must allow a customer to create a numbered virtual queue ticket and notify them if he can enter immediately (if the store is not full) or provide them a waiting time estimation
- R20 The system must ask the customer to provide the estimated visit time when booking a visit to the store
- R21 The system must allow stores to hand out numbered physical queueing tickets to those that do not use the CLup application
- R22 The system must allow the access to the store to customer with numbered tickets using a 'First Come First Served' logic, treating virtual and physical ticket owner equally
- R23 The system must try to estimate waiting time based on store capacity, reservations and the current number of people with numbered tickets waiting in line
- R24 The system should interface with an screen placed at the entrance of the store to notify customer which ticket numbers will enter in the next called batch
- R34 The system must push notifications to user devices with update information on the store he has a ticket for
- DA2 Customer will stay approximately the time they have declared when booking the ticket
- UC7 Customer creates a numbered virtual ticket to enter a store as soon as there is a place available
- UC8 Customer creates a physical numbered ticket

G10: Notify a customer when it's time to approach the store entrance

- R8 The system will use the occupancy data retrieved from the store to control the store access
- R17 The system must allow a customer to create a numbered virtual queue ticket and notify them if he can enter immediately (if the store is not full) or provide them a waiting time estimation
- R18 The system must notify the customers with a virtual queue ticket when it's time to approach the store entrance
- R21 The system must allow stores to hand out numbered physical queueing tickets to those that do not use the CLup application
- R24 The system should interface with an screen placed at the entrance of the store to notify customer which ticket numbers will enter in the next called batch
- R35 The system must associates tickets with line numbers
- DA2 Customer will stay approximately the time they have declared when booking the ticket
- DA6 If people counters are installed they will provide the exact count of the customers that enter or leave the shop
- UC7 Customer creates a numbered virtual ticket to enter a store as soon as there is a place available
- UC8 Customer creates a physical numbered ticket

3.3 Performance Requirements

Considering **CLup customer application** and **CLup operator application**, an user action that implies a communication with a remote server should take no longer than 5 seconds, from the start of the interaction to the moment when the user receives some feedback. When there is no need to connect to a server, this time should be lower than 1 second. This requirements holds in a context where the underlying operative system is not overloaded and the user internet connection is working properly. When a store admin account requests elaborate statistics of the store, the response time must be lower than 10 seconds.

3.4 Design Constraints

- General Data Protection Regulation (GDPR, EU 2016/679) must be followed in order to collect data from users.

3.5 Software System Attributes

3.5.1 Reliability

All the data stored in the CLup databases must be periodically backed up. Data redundancy policies, like RAID, must be enacted to hande failures promptly and lowering the risk of losing data.

3.5.2 Availability

The system must have multiple nodes handling requests from the customers and the operators. These nodes, working in parallel, reduce the risk of a total failure of the system. Data redundancy policies also help to reduce downtimes in case of a data storage unit fault.

The system uptime should be as close as possible to 100%. Since that is impossible, an uptime of 99.9% is more than acceptable and corresponds to less than 9 hours of down time every 365 days. Ordinary maintenance that needs shutting down part of the system should be planned, if possible, during nights; in this way it is more likely that the majority of the users won't experience disruptions in the use of CLup services.

The access control system must continue working even if an automated, external system installed in the store fails. In the following table are listed the measures that must be taken after a failure of an external system:

Table 18: External Systems fails table

External System that fails	Measures
People counter	The CLup system will estimate the number of people inside the store using the average visit times. This value will also be manually adjustable
Smart Gates	An operator with the CLup operator application will check tickets and admit people manually
Paper Ticket Emitter	An operator will form a queue and admits people manually if there is space in the store (information that can still be tracked by the CLup app)
Ticket number screens	An operator announces manually the ticket numbers of the people that should enter the store, looking them up on the CLup operator app

3.5.3 Security

All the data sent via insecure communication channels (i.e. Internet, Local Networks) must be secured with modern encryption standards. This requirement applies to all data exchanged by CLup customer and operator applications and for all traffic generated by the external hardware systems installed on the store.

The external hardware systems installed in the store should be connected in a secure and dedicated LAN with a single access point to the internet with proper firewall settings. This network must be separated from other networks especially the ones accessible freely from the customers (i.e. a shop free Wi-Fi service).

The customer accounts are separated from the business accounts. Each store manager has permissions to manage the operator accounts and has access to all the store data; the manager can decide which account can view which statistics.

3.5.4 Maintainability

The system is clearly divided in a lot of independent subsystems that communicate with each other using some interfaces. As long as the interface between different subsystems is preserved, it is possible to maintain and update single subsystems transparently.

3.5.5 Portability

The CLup customer mobile application compatibility should cover at least 95% of the iOS and Android devices. To make CLup even more accessible, a mobile friendly web interface can be provided to the users.

The CLup operator application must be available via a web application. The functionalities needed for the Access Control Staff must also be available on a mobile application that will be installed on devices given to the staff.

3.6 Other Requirements

3.6.1 Required fields for registration

When a CLup customer creates a CLup customer account, they have to fill a form. Here are specified requirements regarding the information needed to register to CLup. If not otherwise specified every field is compulsory.

- First name
- Last name
- Address, composed of:
  - Country
  - Region/State
  - Town
  - CAP
  - Street and number
  - Additional address lines (optional)
- Gender (optional)
- Date of birth
- Valid email address
- Password, must respect this criteria:
  - Must have at least a digit
  - Must have at least a lowercase character
  - Must have at least a special symbol in this list

    ```
    *.!@#$%^&(){}[]:;<>,.?/~_+-=|\
    ```

  - Must have a number of characters between 8 and 32

4 Formal Analysis Using Alloy

A model written using a modeling language like Alloy is useful to better understand the problem. In particular, it is beneficial to check if the modeling of the problem is correct or needs to be refined. Alloy generates some instances starting from the model code in order to catch modeling errors. Fixing these errors in the early stages of the system lifecycle will be significantly cheaper than fixing them at later stages of the development cycle.

4.1 Goals of the model

The Alloy model proposed in the next subsection of this document will focus on the formalization of the customers queueing and the slot booking system. This model is static, meaning that each instance generated by the alloy engine depicts the state of the system at a specific time instant. The goal of a static model is to generate instances of that model: these represent only the states that the system can assume. The transition between these states are instead described by a dynamic model. Finding an invalid state means that something is wrong in the system or in the model; however even if all states analyzed are correct this does not necessarily mean that the system static model is also correct.

4.2 Alloy Code

```
enum Bool {False, True}

sig Store {
    // List of all numbered tickets in order of issue time
    queue : lone NumberedTicket,
    // List of all bookable timeslots
    slots : lone TimeSlot,
    inside: set Customer
}


sig Customer {
} {
    // A customer retrieves only one ticket (Simplification)
    lone t: Ticket | t.holder = this
}

abstract sig Ticket {
    issued : one TimeStamp,
    used : lone TimeStamp,
    noShow : one Bool,
    holder : one Customer
} {
    // A ticket must be used after it has been issued
    isUsed[this] implies precedesStamp[issued, used]
    // If the customer doesn't show the ticket is not used
    noShow = True implies not isUsed[this]
}

sig BookedTicket extends Ticket {
    timeSlot : one TimeSlot,
} {
    // A ticket must be booked before the timeslot starts
    precedesStamp[issued, timeSlot.start]
    // A ticket must be used within its timeslot
    precedesStamp[timeSlot.end, used] or used = timeSlot.start or noShow = True
    precedesStamp[used, timeSlot.end] or used = timeSlot.end or noShow = True
}
```

```
abstract sig NumberedTicket extends Ticket {
    next : lone NumberedTicket
}

sig PhysicalTicket extends NumberedTicket {}

sig VirtualTicket extends NumberedTicket {}

sig TimeSlot {
    nextSlot : lone TimeSlot,
    start : one TimeStamp,
    end : one TimeStamp
} {
    // The timeslot end timestamp is greater than the slot's start timestamp
    precedesStamp[start,end]
}

sig TimeStamp {
    nextStamp : lone TimeStamp
}

fact timeFlow {
    // A first TimeStamp does exists
    one first : TimeStamp | no t: TimeStamp | t ≠ first and first in t.*nextStamp
    // All timestamps follow the first TimeStamp
    one first : TimeStamp | all t: TimeStamp | t ≠ first implies precedesStamp[first, t]
    // No cycles in time
    no t : TimeStamp | t in t.nextStamp.*nextStamp
}

fact slotFlow {
    // A first TimeStamp does exists
    all s: Store | no t: TimeSlot | t ≠ s.slots and s.slots in t.*nextSlot
    // All timestamps follow the first TimeStamp
    all t: TimeSlot | one s: Store | s.slots = t or precedesSlot[s.slots, t]
    // No cycles in time
    no t : TimeSlot | t in t.nextSlot.*nextSlot
}

fact timeSlotNotOverlapping {
    // A previous timeslot's start and end timestamps are before the current timeslot,
    // there is no overlapping of timeslots
    all s1: TimeSlot | all s2 : TimeSlot |
        sameStoreSlot[s1,s2] implies
        (precedesSlot[s1,s2]
        iff (precedesStamp[s1.start, s2.start] and
             precedesStamp[s1.end, s2.end] and
             precedesStamp[s1.end, s2.start]))
}

fact queueFlow {
    // A first TimeStamp does exists
    all s: Store | no t: NumberedTicket | t ≠ s.queue and s.queue in t.*next
    // All timestamps follow the first TimeStamp
    all t: NumberedTicket | one s: Store | s.queue = t or precedesQueue[s.queue, t]
    // No cycles in time
    no t : NumberedTicket | t in t.next.*next
}

fact firstComeFirstServed {
    // A ticket with a smaller timestamp is ahead in the queue
    all t1: NumberedTicket | all t2: NumberedTicket |
        sameStoreQueue[t1,t2] implies
        (precedesStamp[t1.issued,t2.issued] implies precedesQueue[t1,t2])
    // A customer enters the store only if all people before him have entered
    // the store or no-showed
    all s: Store | all t1: NumberedTicket | all t2: NumberedTicket |
        (inStoreQueue[s,t1] and inStoreQueue[s,t2] and precedesQueue[t1,t2]) implies
        ((t2.holder in s.inside) implies (t1.holder in s.inside or t1.noShow = True))
}

fact ticketNeededToEnter {
```

```
    // No one can enter without a ticket
    all s : Store | all c : Customer | c in s.inside iff (one t: Ticket | t.holder = c
        ↪ and isUsed[t])
}


pred isUsed (t: Ticket) {
    #t.used = 1
}


pred precedesStamp (t1, t2: TimeStamp) {
    t2 in t1.nextStamp.*nextStamp
}


pred precedesSlot (t1, t2: TimeSlot) {
    t2 in t1.nextSlot.*nextSlot
}

pred sameStoreSlot (t1, t2: TimeSlot) {
    precedesSlot[t1,t2] or precedesSlot[t2,t1] or t1 = t2
}

pred sameStoreQueue (t1, t2: NumberedTicket) {
    precedesQueue[t1,t2] or precedesQueue[t2,t1] or t1 = t2
}

pred precedesQueue (t1, t2: NumberedTicket) {
    t2 in t1.next.*next
}

pred inStoreQueue (s : Store, t : NumberedTicket) {
    t = s.queue or precedesQueue[s.queue,t]
}

pred inSlot (t: TimeStamp, s: TimeSlot) {
    t = s.start or t = s.end or
    (precedesStamp[s.start,t] and precedesStamp[t, s.end])
}

pred show {

}

//RUN
run show for 7 but 10 TimeStamp
```

4.3 Instance Analysis

In this section some generated instances by the alloy engine are shown in order to better explain the model. For clarity some atoms and some relations between them are hidden.

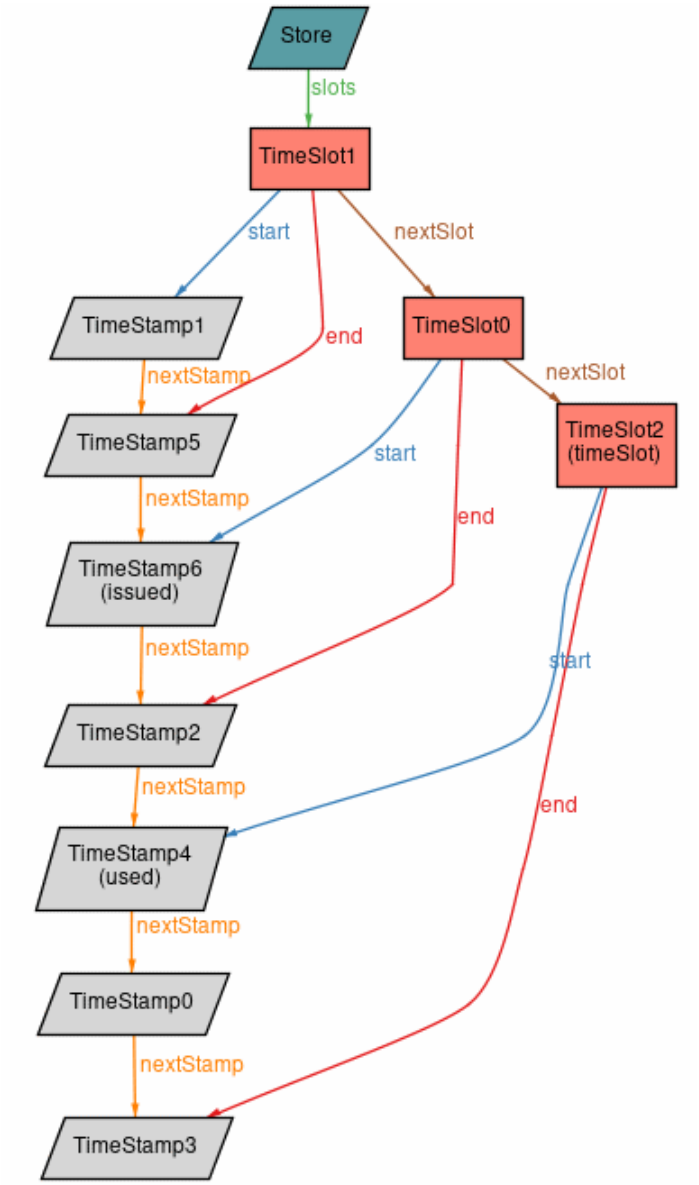


Figure 13: Instance diagram showing the Time Slots

As shown in Figure 13 the continuos time is modeled using discrete, ordered timestamps. A timestamp is an instant of time when something relevant for the system happens (i.e. a time slot starts, a ticket is issued or used to enter the store). Each instance offers a view of a valid system state at the last (largest) timestamp, which is **TimeStamp3** in Figure 13.
In this instance diagram, time slots atoms are shown. Every couple of TimeSlots created by the same store must not overlap and the next TimeSlot should start after the end of the current one.

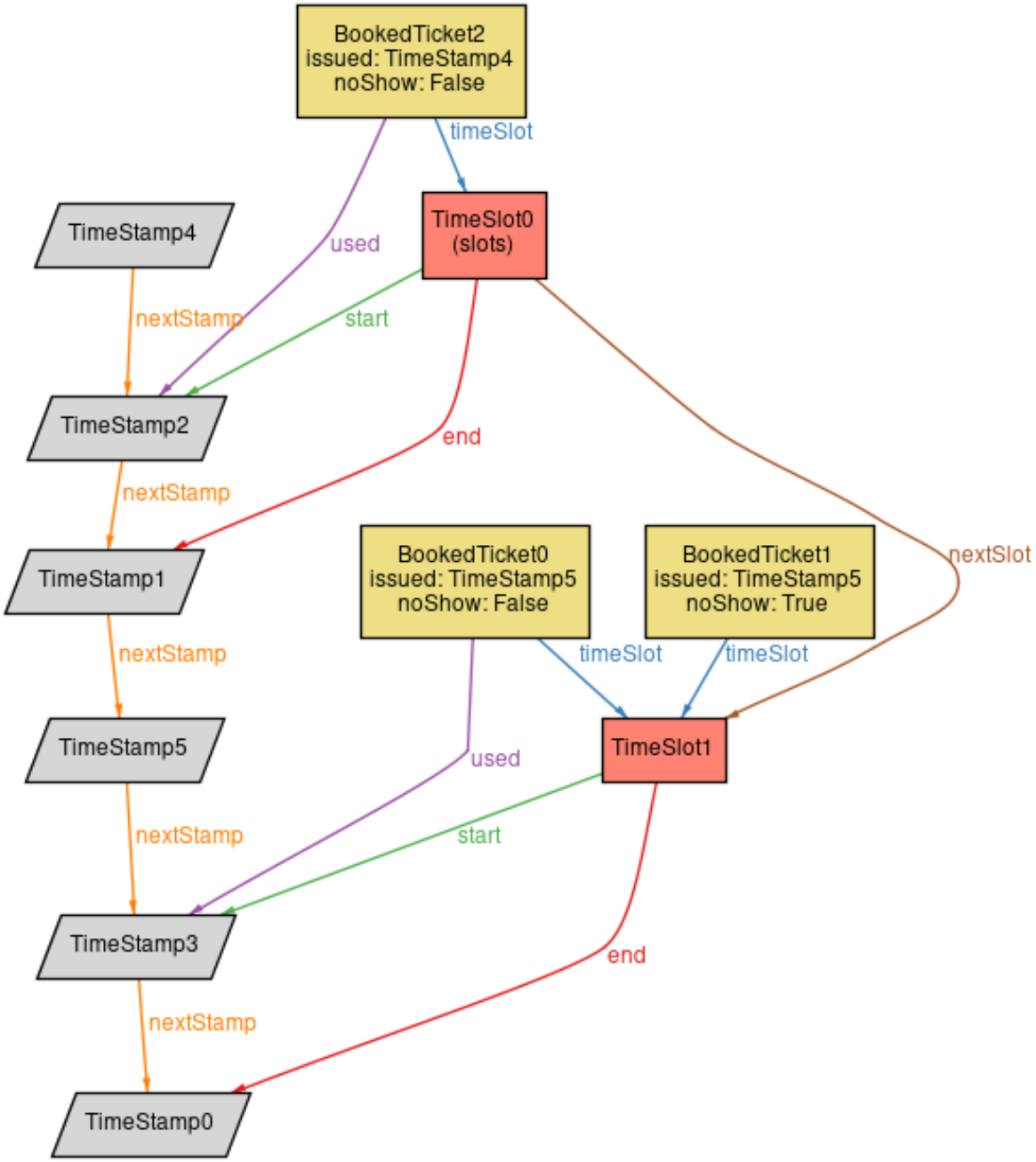


Figure 14: Instance diagram showing Booked Tickets

In Figure 14 booked tickets are shown in the diagram. BookedTickets are a special type of ticket that is bound to a TimeSlot. A Booked Ticket must be issued before the start of the time slot, and it has to be used within its associated TimeSlot. A BookedTicket not used in its TimeSlots counts as a **no-show** ticket (which represents a customer that books a ticket but does not show up to the store in time).

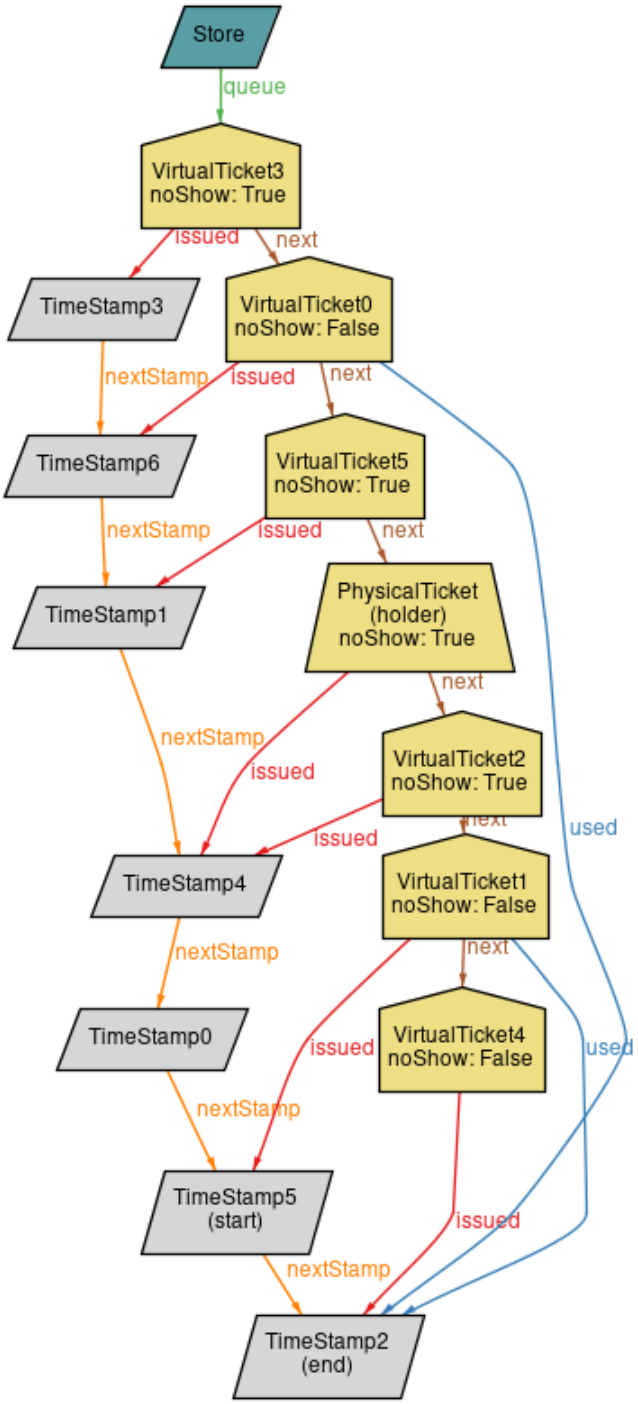


Figure 15: Instance diagram showing the queueing system

The Figure 15 explains how the store queueing system is modeled. If a customer doesn't book a ticket to enter the store, they have to retrieve a NumberedTicket, either Virtual (using the application) or Physical (using the ticket emitter). These tickets are handed using a **First Come First Served** Logic: a ticket with a lower TimeStamp should be called to the entrance before a ticket with an higher one. The TimeStamp representing when the ticket is used should also reflect this order unless no-shows happen (i.e. A customer that retrieves a ticket and then does not show when called to the entrance).

5 Effort Spent

Dario Passarello

- Discussion about specific: 5 hours
- Discussion about Requirements and Use Cases: 4 hours
- Other Meetings: 5 hours
- Document Drafting: 3 hours
- Class and State Diagrams: 3 hours
- Use Cases: 4 hours
- Use Case & Sequence Diagrams: 4 hours
- Requirements Mapping : 2 hours
- Alloy Coding: 4 hour
- Alloy Review: 2 hour
- Final Review: 3 hours

Total: 39 hours

Davide Luca Merli

- Discussion about specific: 5 hours
- Repository Setup and Structure: 1 hour
- Discussion about Requirements and Use Cases : 4 hours
- Other Meetings : 5 hours
- Introduction: Scope, Purpose: 3.5 hours
- Introduction: Document Structure, Definitions and Acronyms: 3 hours
- Chapter 2: Refactor and Style changes: 3 hours
- Specific Requirements: 0.5 hours
- Mockups: 3 hours
- Use Case Review: 1 hour
- Alloy Review, Code style, Minor Refactors: 1.5 hours
- Final Refactor: 6 hours

Total: 36.5 hours