Politecnico di Milano

A.A. 2020/2021

Computer Science and Engineering

Software Engineering 2

RASD

Requirement Analysis and Specification Document

CLup: Customers Line-up

Group members:

Andrea Razza (968483)

Loris Panza (967915)

Version: 1.0

Chapter 1

Introduction

1.1 Purpose

In recent months, many countries have introduced security measures to contain the Coronavirus pandemic. One of the most strict actions has been for sure the lockdown that allows people to exit their homes only for essential needs, such as going to grocery stores.

In order to avoid having crowds, the CLup (Customer Line Up) application purpose is to manage the accesses into these shops. Specifically, the scope is to regulate the influx of people into and outside the building and saves people from having to line up and create gatherings. The CLup is designed as a software application used to line up the customers from their home: so they can reach the shop only when their number is called.

In addition, the grocery shop owner can monitor the entrances thanks to scanned QR codes generated by the software: in this way he can trace information about his clients.

The application allows customers to book their visit to the grocery in a specific hour slot of a certain day, so he/she can program shopping and occupy his turn in advance. Hence, upon booking a visit, the system allows the users to insert some useful information to better manage the visits: customers can insert the estimated needed time and, if they don't specify it, the software could calculate an approximate period based on the duration of previous visits. Specifying also the kind of products to purchase, the software can organize contemporary customer accesses to let them occupy different spaces in the grocery guaranteeing enough distance.

1.1.1 Goals

G1	The system prevents the creation of physical queues at the entrance of shops.
G2	The system offers the opportunity to visit shops avoiding wasting time in the queue.
G3	The system helps grocery’s owners to manage the client's influx to shops.
G4	The system allows the respect of the social distancing policy into the shop.

1.2 Scope

The software wants to give users the possibility of waiting in a virtual queue for the access in grocery stores.

CLup offers two different services:

- Basic service: the system allows users to sign up and wait their turn from home visualizing how many people precede him. Every customer in the queue has a unique QRcode. The system should provide customers with a reasonably precise estimation of the waiting time and should alert them taking into account the time they need to get to the shop from the place they currently are. This place can be specified in two ways: either through GPS or indicating a specific address. In order to manage in an optimal way the queue waiting time, every user can specify an approximate expected duration of the visit.
- Booking a visit: users can also submit a reservation indicating the same information of the basic service plus the specific time and date. The customer can insert various types of products that he is going to purchase in order to allow the software to coordinate optimally and, eventually, contemporary clients in the market.

In order to be available in the application, the grocery shop should sign up to the software indicating specific information.

For people that have no access to this technology there are ticket machines which physically distribute tickets. The customer can queue thanks to these tickets: specifically, the user will indicate the required information to the grocery's ticket machine. This machine, acting as a proxy, will send the information to the software and will print automatically a ticket where the user can visualize all the details of the required service. In this way, CLup is constantly updated in order to manage both types of clients (physical and online ones).

It is necessary also to distinguish physical and virtual interactions: for instance, the application should be aware of sending notifications only for digital users and not for the physical ones.

If the user doesn't specify the expected time, CLup infer it from an analysis of the client's previous visits. If the client has never gone to that grocery or has never indicated any expected period, the software estimates a time based on the average period spent by the previous user in that specific shop.

When a customer arrives at the market, QR code generated by the application or printed on physical tickets must be scanned to make sure that the person is allowed to enter, through reservations or queue. This scan is useful also to keep track of customers' generalities and to understand the effective time spent in the shop. Specifically the QR code has to be scanned when the user enters the shop and when he leaves it to define the effective time.

With respect to laws, the number of people that can simultaneously access the grocery is calculated according to the shop's dimensions.

1.2.1 World phenomena

WP1	One or more users need to visit a grocery shop
WP3	A user wants to purchase a specific kind of product
WP4	A user approaches a grocery shop
WP5	A user knows how much time he will spend in a shop
WP6	A user buys a product in a shop
WP7	A shop is open

1.2.2 Shared phenomena

SP1	A user join the queue for a shop using CLup- World controlled, Machine observed
SP2	A user books a visit to the grocery shop - World controlled, Machine observed
SP3	A user examines the details of his booking - World Controlled, Machine Observed
SP4	A user checks how much time remain before going to the shop - World Controlled, Machine Observed
SP5	The user scans QR code - Machine controlled, World observed
SP6	The system alerts user to go to the shop - Machine controlled, World observed
SP7	The system updates the amount of time remained for an user based on the number of people in queue - Machine Controlled, World Observed

SP8	The system estimates an approximate expected duration of the visit for a customer - Machine Controlled, World Observed
SP9	The system generates a QR code - Machine Controlled, World Observed

1.3 Definitions, acronyms, abbreviations

1.3.1 Definitions

Queue	Sequence of clients that wait their turn keeping distance from the shop.
Digital client	User who takes advantages from services offered by the application through the digital software.
Physical client	Users who have no possibility to use the software and go physically to the grocery store to access the application service.
Ticket	Certification for physical clients of having benefited the service.
Client scanning operation	QR code scanning upon client arrival at the shop.
Alert / Notification	Message sent by CLup app to alert the customer of going to the shop.
Estimated time	Time calculated by the application if the user does not insert any expected time.
Expected time	Time specified by the user that indicates how long that customer will stay in the shop.
Remaining time	Time calculated by the application and visualizable by a digital user that indicates how much time is remaining before being into the shop.
Effective Time	Effective time passed between the entrance of the shops and the exit.
To line up / To queue/ To join the queue	Being part of a virtual queue and waiting for the turn.

Overflow user	A booked user who is allowed to enter a full shop because it has indicated in his request a type of product different from all the other simultaneous booked clients in the shop.

1.3.2 Acronyms

- QR Code: Quick Response Code;
- RASD: Requirement Analysis and Specification Document;
- GPS: Global position system.

1.3.3 Abbreviations

- Gn: n-th goal;
- Dn: n-th domain assumption;
- Rn: n-th requirement;
- WPn: n-th world phenomena;
- SPn: n-th shared phenomena.

1.4 Reference documents

- Specification document: “R&DD Assignment AY 2020-2021”;
- Slides from the lectures.

1.5 Overview

Chapter 1 gives an introduction to the purpose of our assignment indicating main goals, some of the world and shared phenomena. Moreover, in the section are defined the limits of our software, specifically there is a general description about all the services provided and the scope of these ones. In addition, the chapter indicates definitions, abbreviations and acronyms used in the document.

Chapter 2 contains the overall description of the project. In the product perspective are included the statecharts of the major function of the application and the UML diagram. The product function clarified the functionalities of the application. Furthermore, in user characteristics are explained the types of actors that can use the different services. Finally, it includes the domain assumption and the constraints that can be deducted from the assignment.

Chapter 3, in the first part, shows some aspects of user interfaces, thanks to mockups, and explains hardware and software interfaces by specifying all the components, physical and not, that CLup is made of.

The second part is dedicated to use cases: a use case diagram is at the beginning and then there are use cases, sequence diagrams and functional requirements ordered by scenarios. Going down, the traceability matrix can be found: a matching between goals, requirements and domain assumptions.

The fourth and fifth paragraphs talk about various performance requirements and domain constraints, such as software and hardware limitations and standard compliance.

The 3.6 part, treats system attributes that the software is required to have: reliability, availability, security, maintainability and portability. Lastly, the 3.7 part is about other various requirements.

Chapter 4 includes the alloy code and the corresponding metamodels generated from it, with a brief introduction about the main purpose of the alloy model.

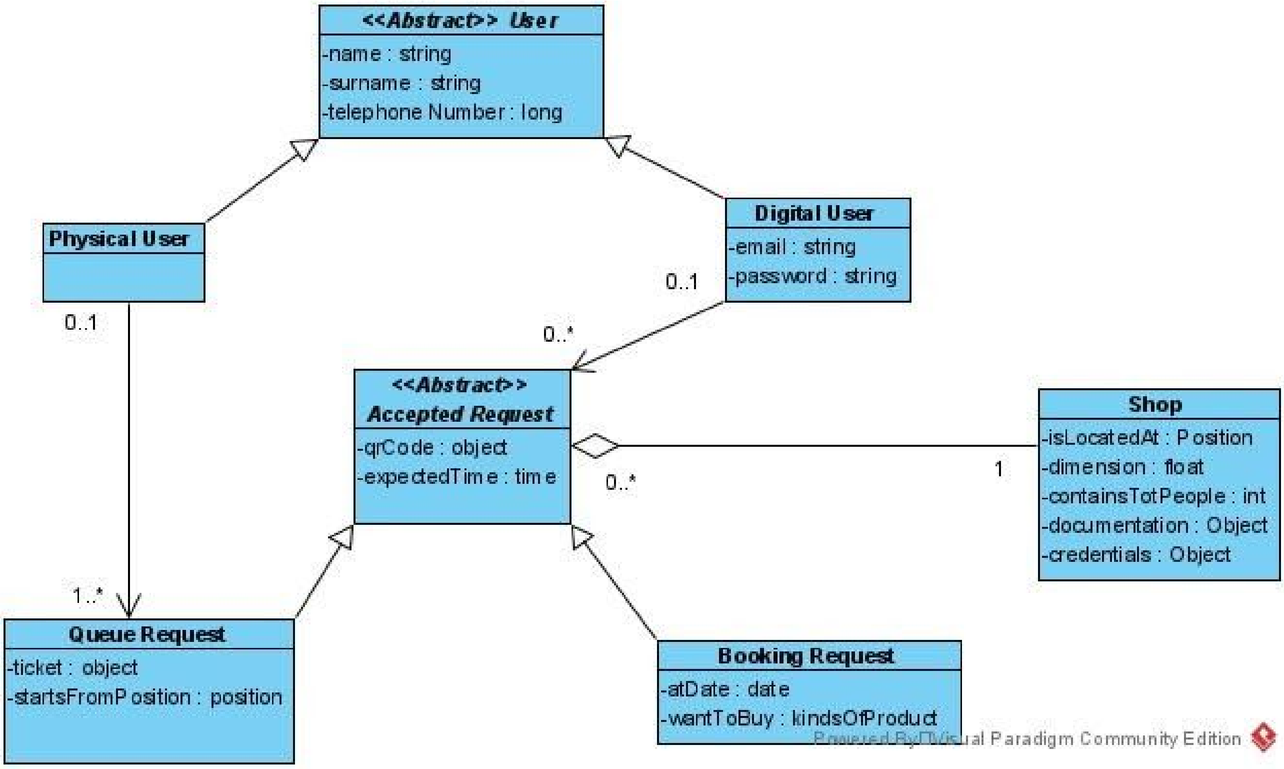
Chapter 5 shows the effort spent for each member of the group.

Chapter 2

Overall description

2.1 Product perspective

2.1.1 UML diagram



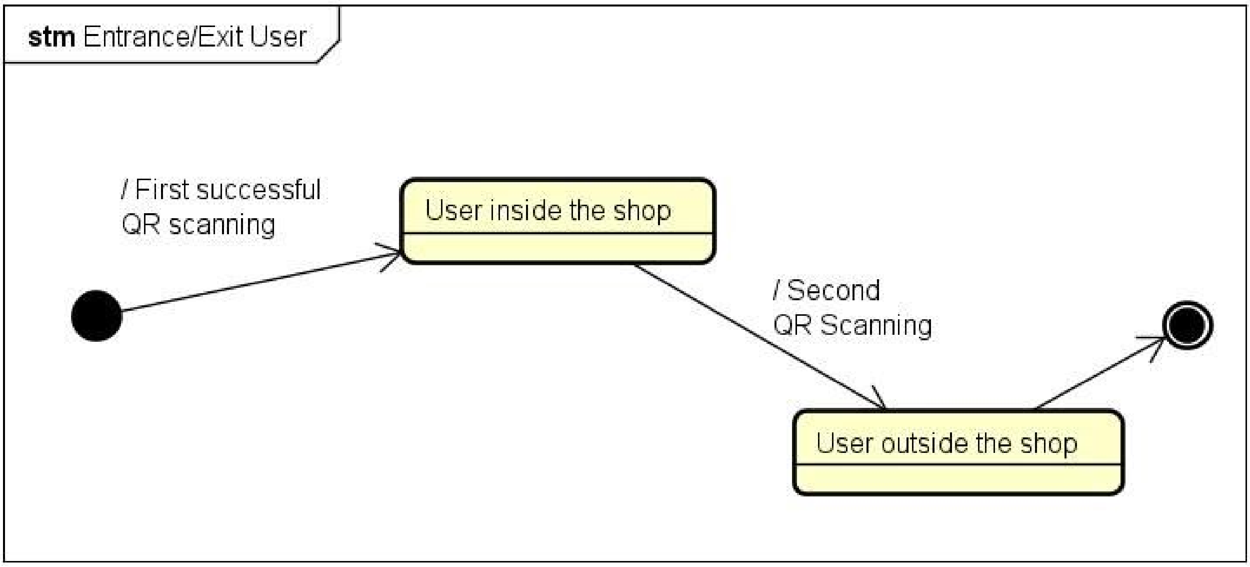
The UML describes at high-level the model of the system to be developed. It does not include all the necessary classes and attributes to completely define the system architecture: the class diagram is useful to underline all the main components which actively participate in the different systems' functions.

Digital and physical users share some common characteristics but digital users have information linked to the relative account. Physical users, by means of ticket machines, can only join a queue while the digital ones can take advantage also of the booking service. Each user is considered physical if he performs at least one queue request while the digital is considered as an user even if he does not request any service: it's sufficient to be registered. Each type of user can make one or more requests and each of them has to be related to one shop and consequently to one user that can be either physical or digital. A shop can receive none or more requests.

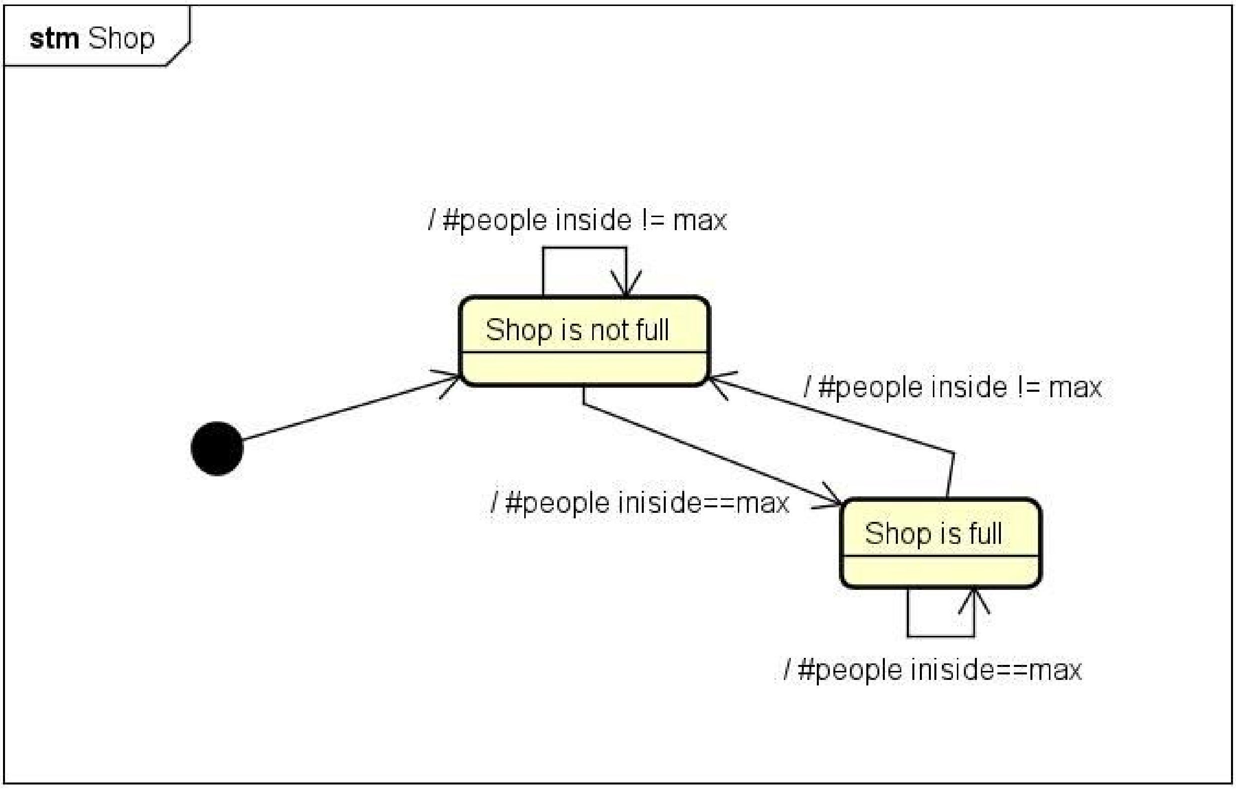
2.1.2 State and Activity Diagrams

Through the following state charts, we are going to explain critical processes related to some functionalities of the system. Specifically, these charts model the behaviour of software over the time.

For estimating the queue delay and the user’s estimated time who has not defined any expected time, the system needs to be aware of the previous visits effective time spent into the shop. In order to do that, the QR code must be scanned at the customer entrance and exit. These two times are sent to CLup that changes in real time the remaining time for each person in the queue. Every QR code is unique for each customer and for each request so the systems understand if the scanning is the first for entering or the second for leaving. If the first scanning operation is successful, the user is able to enter the store otherwise the type of error raised will be communicated through the application. Specifically, if the user approaches the store too much time later with respect to the remaining time or in relation to the booked visit, the scanning operation will be rejected and the relative reason will be displayed on the app. For the physical users which do not have any device but just the ticket, they can visualize the result of the evaluation of the scanning through a small display located near the scanner.



The contemporary number of people into the grocery store is communicated by the shop manager. CLup has to be able to understand when the shop is already full counting the customers entering and leaving. This functionality is based on QR scanning at the entrance and at the exit that allow CLup to update the shop state. If the shop is momentarily full, the user is warned to wait at the entrance of the shop.



2.1.3 Scenarios

Scenario 1: Joining a queue

Andrea, a young student, is terrified of the coronavirus because he lives with his family and doesn’t want to infect them, so he avoids getting into queues and crowds. Nevertheless, he needs to go to the supermarket and, so he downloads CLup, registers for it and compiles the form to join the closest grocery shop’s queue. By means of CLup, he is able to visit the shop in safe conditions sending a simple request. Now he can wait his turn from home with his family and can check the time remaining before going in every moment.

Tom is an eighty years old retiree and lives with his wife Emily. Covid-19 has changed their life and they used to visit Alex’s supermarket that recently joined CLup. Tom and his wife Emily aren’t able to use smartphones and are scared of contracting a disease,so, the only way to visit Alex's grocery is getting a ticket from the ticket machine outside the shop. It’s 10:00 a.m. and Tom leaves his house to approach the market, the machine requires him to compile the queue form. His request goes well and the machine generates his queue ticket. Even though Tom is not very used with these new technologies, he can continue to visit his preferred store.

Scenario 2: Booking a visit

Loris, a very busy engineer and habitual CLup user, has only 30 minutes free so he decides to join the closest grocery's queue. Unfortunately, he realizes that the remaining time is about forty minutes, so he decides to cancel his request and to book a visit for the day after. So, he visualizes the booking form, compiles it and sends the request. Not having inserted the expected time, CLup infer it from an analysis of the previous visits and confirms the reservation. By doing so, he can avoid wasting his time and can program his routine in an optimal way.

Scenario 3: A shop subscribes to CLup

Matteo is the owner of the one of the most popular supermarkets in Avellino and he is worried because of the pandemic, especially for the big number of customers that usually buy at his shop. In the last days Matteo heard of a new app that's right for his activity: CLup. So he decides to join it and, after having downloaded it, he accesses the store managers' dedicated section. He fills a form where he specifies information related to his grocery and sends the request that will be accepted. In this way, the store can continue to sell its product in safe conditions. One week later, Matteo wants to consult the clients flow of his grocery, so he logs in to CLup application and access to the dedicated section.

Scenario 4: Receiving a notification and approaching the shop

Madison is an important business woman and she needs to buy something to eat as soon as possible because her fridge is empty. She joins a queue for the nearest shop that is about 30 minutes from her home and realizes that she has the necessary time to work on another project because the remaining time is about 2 hours. The time goes on and she completely forgot about that reservation because of her commitments.

While she was working, Madison sees on her smartphone the CLup notification and in a moment she quickly gets in her car. She fortunately approaches the shop at the right time, can the QR code and the application stop her from entering the shop because it is momentarily full. This means that as soon as the next customer comes out Madison will be able to get in. Once finished shopping, she re-scans the code to go out.

2.2 Product Functions

In the following document section, the system functionalities are described in detail.

2.2.1 Main Functions

Joining a shop queue

One of the main functionalities of the system is to organize grocery shops' customers in a virtual queue. In this way, the user can wait his turn from home and then approach the shop without creating crowds outside of it. In order to take advantage of the services, the customer has to sign up entering these informations: name, surname, e-mail, password and a telephone number. Upon submitting the queue request related to a specific shop, the user can fill an optional form in which he defines his current position indicating an address or using the GPS and the expected time.

If the user doesn't specify the expected time, CLup infer it from an analysis of the client's previous visits: it estimates a time based on the average effective period spent by the previous user in that specific shop.

If the client has never gone to that grocery, the software estimates an approximate period based on the number of people in the queue and the shop's dimension .

Thanks to the expected time and effective, the software can precisely calculate in real time the remaining time between each customer in the queue. Obviously, the remaining time is also based on the shop's dimensions and the consequent number of people that can simultaneously enter.

Through the application, the user can visualize the number of people in the queue before him, the shop, QR code, the remaining and expected time. If the user specifies the position, the application will automatically calculate the required time to reach the shop from that position. When that time is equal to the remaining time, the software will send a notification to the user. If the user doesn't define the position, none of the notifications are sent.

Booking a shop visit

The booking service allows customers to visit shops on a specific date and time. In this way, the user can avoid waiting for his turn in the queue and can programm his shopping session in advance. In order to take advantage of the booking service, the user has to be registered for CLup and could fill the optional form previously discussed in the basic service. Moreover, the insertions of the date and time are mandatory. The expected time can be expressed by the user or estimated by the application. Nevertheless, the remaining time is not shown: the booked user takes precedence over the queue. If the shop is momentarily full at the time when a user has booked his visit, he is the next to get in independently from the current queue. An alert will be sent at ‘00:00’ of the booked day.

Upon booking a visit, the user has the possibility to insert the types of product that is going to purchase: in this way, contemporary clients in the shop can be managed basing on how the product sections are located.

The shop has a specific maximum number *n* of people inside it: for this reason *n* users are able to book a visit to the store in the same date and hour interval but there is a specific condition in which other users can access the shop exceeding the limited number.

Specifically, if all the *n* contemporary booked users have indicated the kinds of products to buy and another customer specifies different types of goods from all the others, the shop will evaluate the reservation request. If the products are different, the shop will confirm the extra-reservation through CLup. Specifically, the “overflow user” will be able to enter the full shop if and only if inside of it there will be only booked clients that have indicated in their bookings different types of product. If the shop is not full, he will be normally allowed to get in.

Getting a ticket

Every registered shop in CLup should provide a ticket machine for the customers that don't have the possibility to use the required technology. The user will not provide some of the information mentioned before, such as position and e-mail. In this way the user can join the queue through a ticket generated by a machine with a printed QR code and the shop’s name. The ticket will show the right time when the user will have to approach the shop (instead of the remaining time) and the expected time calculated by CLup or defined by himself. A ticket is valid only for the shop where it has been generated.

2.3 User characteristics

The system provides a registration form for new users. Each type of user needs to provide different information, as discussed before; shops are required to testify their role providing the necessary identification documentation that will be checked before the acknowledgement. In addition, the store manager has to communicate the dimension of the shop and the consequent number of people that can safely stay inside of it at the same time. Creating an account for both the users and shops is mandatory in order to exploit the system’s functionalities. Moreover, a digital user can take advantage of the ticket machine for getting a ticket; so, in this case, he is considered a physical and digital user with the same generalities.

1. Digital User: a person who registers for CLup.
    - Basic Service: user interested in visiting the shops waiting his turn in a virtual queue;
    - Booking Service: user interested in visiting the shops in a specific date;
2. Physical User: a person who takes advantage of the queue service getting physically a ticket from the ticket machine located at the grocery shop.
3. Grocery shop: shop registered for the software interested in accepting all the customers in safety condition inside and outside the structure. The shop manager has the possibilities to check all the active and past customers which have visited the shop using CLup.

2.4 Constraints, Assumptions and Dependencies

2.4.1 Domain assumptions

D1	When a user inserts in a booking service the product required, this one is sold by the grocery shop.
D2	Each shop respects the opening period.
D3	Each shop communicates the correct dimension of the structure.
D4	Each shop communicates the correct number of contemporary people inside of it respecting the physical distancing policy.
D5	Each user scans the QR code only one time before entering and before leaving the shop.
D6	Each user can't spend more than 5 minutes over the expected time expressed by himself or estimated by the system inside the sop.
D7	Every digital user disposes of a working internet connection to join a queue or to book a visit.
D8	Each booked user that specifies the kinds of product to purchase visits only the related departments of the shop.
D9	If a user scans the unique QR code for getting in the shop and the operation is successful, he enters the shop.
D10	If the user scans the unique QR code for leaving the shop, he actually goes outside of it.
D11	If the scanning operation is rejected, the user doesn't get into the shop.

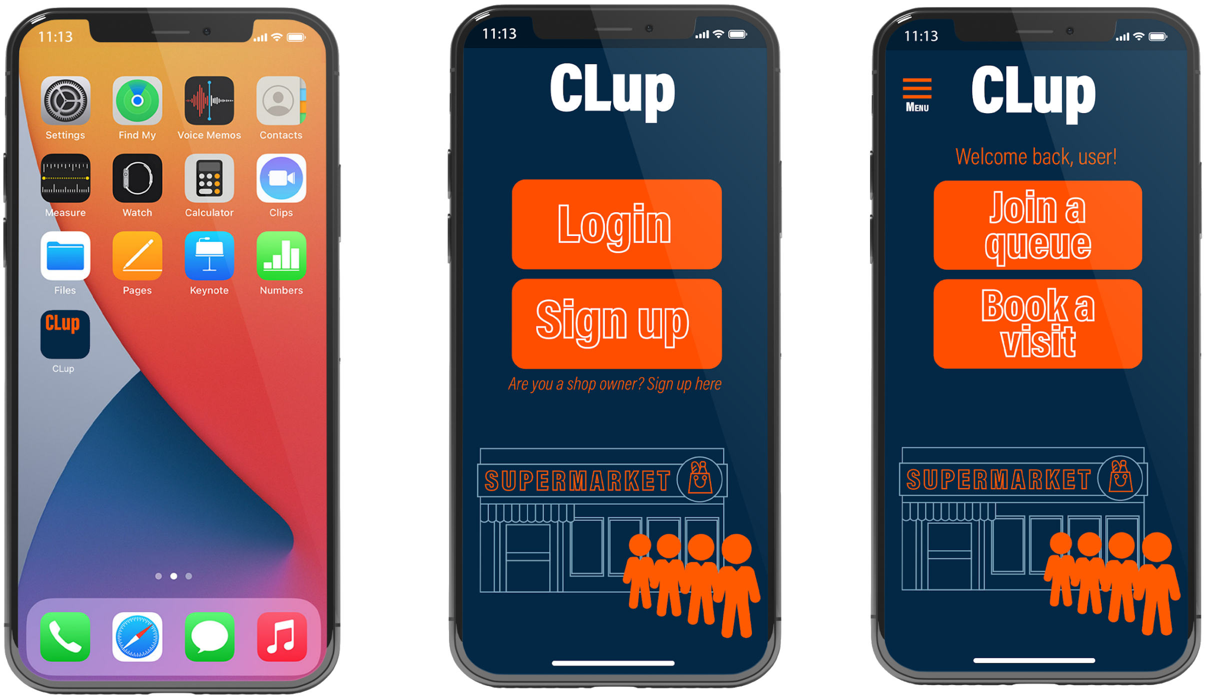
Chapter 3

Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The following mock- ups give a general idea on how the icon of the application will look like and the starting home page of CLup in a mobile device. Other mock-ups will be described in the Design Document.



3.1.2 Hardware Interfaces

The digital users can use their smartphones to take advantage of the services. Using these devices, he can be informed about all the relevant information of the service required.

The physical users can benefit from CLup by means of tickets machines. All the useful information is printed on the tickets.

Shops can sign up using the smartphone indicating all the documentation required.

At the entrance of the shop, there must be a QR scanner in order to detect the QR code when the user enters and leaves the shop. For physical users that own paper tickets, there must be a related display that shows the scanning result.

3.1.3 Software Interfaces

The system uses the following external interfaces:

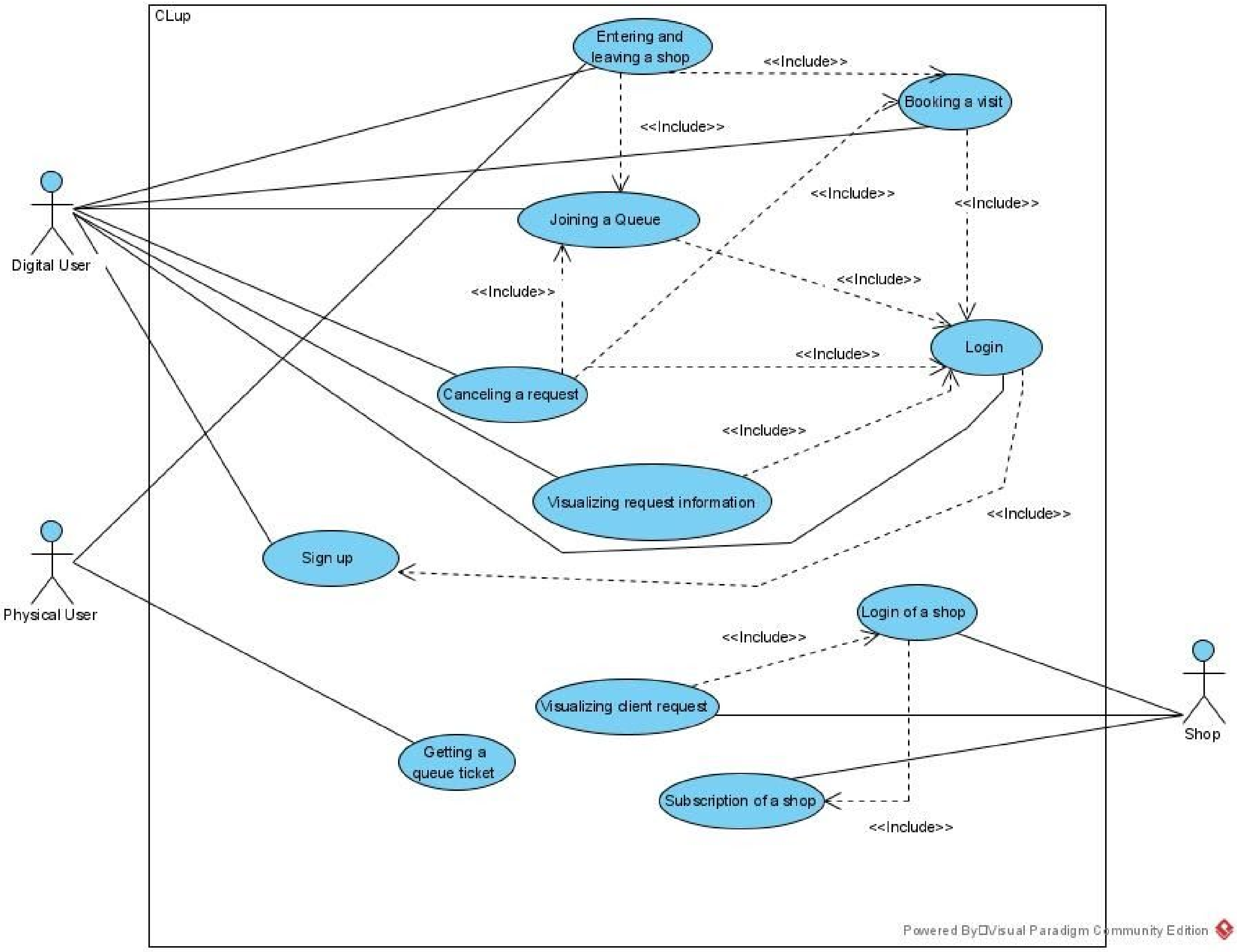
- **Real time position**: We assume that the system uses a public API to provide to the user his position in real time for the basic service compilation.

3.1.4 Communication Interfaces

The device connects to CLup system via internet connection.

3.2 Functional Requirement

The following use cases, use case diagrams and associated sequence diagrams are related to the scenarios previously described.



3.2.1 Use Cases - Scenario 1: Joining a queue

Use Cases

1) Sign up

Name	Sign up
Actor	Digital user
Entry Condition	The user opens the application CLup on his device.
Event Flow	1. The user clicks on "Sign up" button 2. The user fills the mandatory fields 3. The user clicks on "Confirm" button 4. The system saves the new account
Exit Condition	The user is registered and the system has its data stored.
Exception	1. The user doesn't fill the form completely or not in a correct way. The system notifies which fields are not compiled yet or shows the errors occurred. 2. The inserted credentials are already used. The system invites the user to change them.

2) Login

Name	Log in
Actor	Digital user
Entry Condition	The user opens the application CLup on his device and is already signed up.
Event Flow	1. The user clicks on the "LogIn" Button. 2. The user inserts in the form his "Email" and "Password" credentials . 3. The user clicks on the "Enter" button. 4. The system communicates to the user the request's success.
Exit Condition	The user is logged in and he is redirected to the homepage where he can manage his data and take advantage of the services.

Exception	1. The inserted email doesn't correspond to any existing account. 2. The inserted password is not correct. In both cases, the system notifies the user to re-fill the login form.

3) Joining a queue

Name	Joining a queue
Actor	Digital user
Entry Condition	The user is already logged in CLup and needs to visit a specific grocery store.
Event Flow	1. The user visualizes the homepage. 2. The user clicks on the "Join a queue" button. 3. The user searches a grocery store, by indicating its name, from the list of registered shops that displays their names and addresses. 4. The user clicks on the shop he wants to visit. 5. If the user wants to insert a position he can do it by GPS or by inserting an address: 5.1. If the user selects the option "Insert an address": 5.1.1. The user inserts city, province and address in the relative fields. 5.2. Else the user selects the option "GPS": 5.2.1. User clicks "Yes" on "Allow CLup to access the position?" request. 6. The user can write in the optional field the expected time of his visit. 6.1. If the user does not insert any expected time, it is calculated by the application analyzing statistics. 7. The user clicks on the "Send Request" button. 8. The system confirms the request and saves the data.

Exit Condition	The user is inserted in the queue and visualizes the number of people that precede him, the shop, the remaining and expected time and the QR code for the visit.
Exception	1. The user clicks "No" on "Allow CLup to access the position?" request. In this case the only way to insert the position will be indicating manually the address. 2. The inserted shop is not found. The system will show an error message offering the possibility to choose another store. 3. The sent request is rejected because the selected grocery store is actually closed. The system reacts blocking the request and showing an error notification.

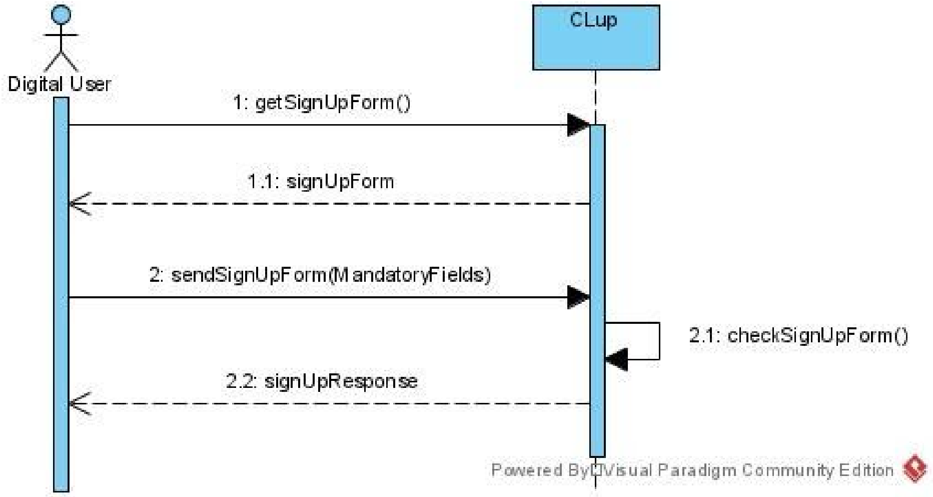
4) Getting a queue ticket

Name	Getting a queue ticket
Actor	Physical user
Entry Condition	The user approaches the market he wants to visit and finds the ticket machine.
Event Flow	1. The user visualizes the homepage. 2. The user selects the "Join the queue" option. 3. The user fills the mandatory form inserting his generalities, such as name, surname, a telephone number 4. The user can write in the optional field the expected time of his visit. 4.1. If the user does not insert any expected time, it is calculated by the application analyzing statistics. 5. The user clicks on the "Send Request" button. 6. The system confirms the request and saves the data.
Exit Condition	The ticket machine generates a ticket on which is printed the QR code, the shop,the expected time and the right

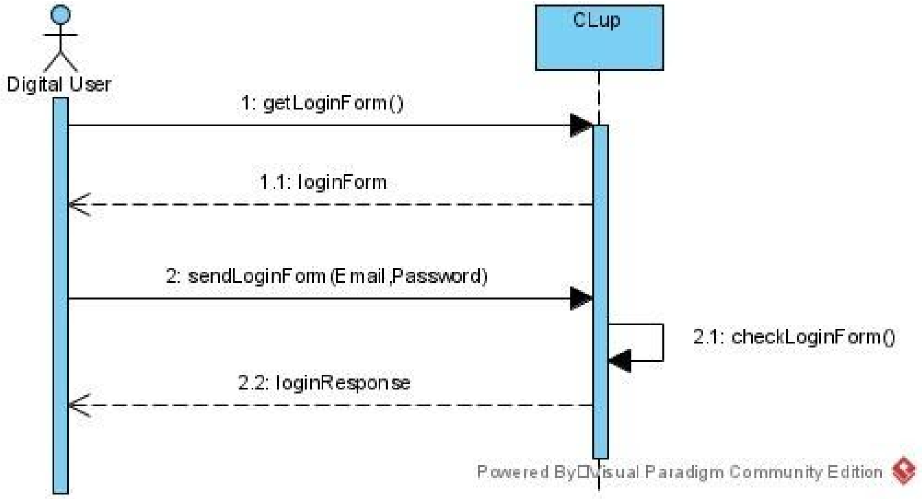
	time when the user will have to approach the shop.
Exception	1. The user doesn't fill the form completely or not in a correct way. The system notifies which fields are not yet compiled or shows the errors occurred.

3.2.1.1 Sequence Diagrams - Scenario 1

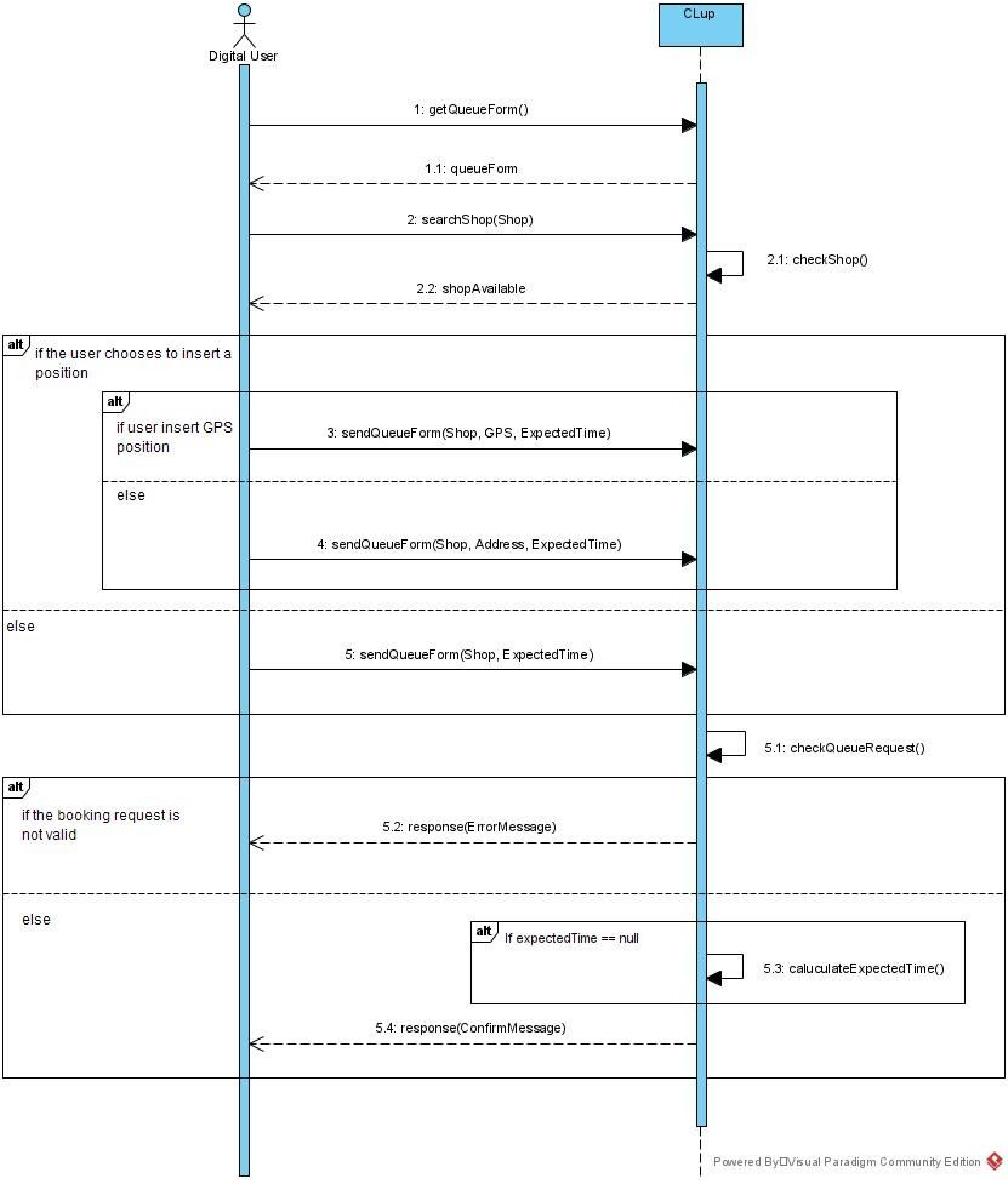
1) Sign up



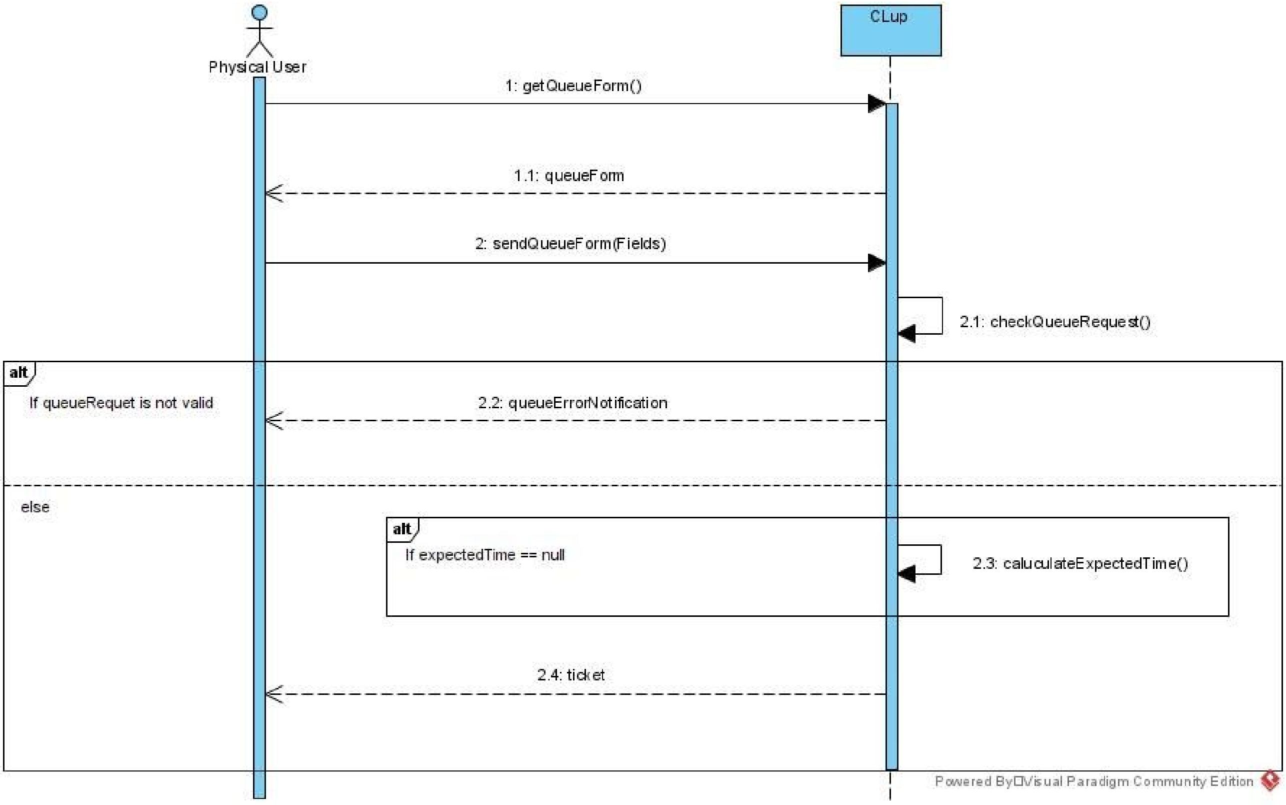
2) Login



3) Joining a queue


Digital User
CLup
1: getQueueForm()
1.1: queueForm
2: searchShop(Shop)
2.1: checkShop()
2.2: shopAvailable
alt
if the user chooses to insert a position
alt
if user insert GPS position
3: sendQueueForm(Shop, GPS, ExpectedTime)
else
4: sendQueueForm(Shop, Address, ExpectedTime)
else
5: sendQueueForm(Shop, ExpectedTime)
5.1: checkQueueRequest()
alt
if the booking request is not valid
5.2: response(ErrorMessage)
else
alt
If expectedTime == null
5.3: caluculateExpectedTime()
5.4: response(ConfirmMessage)
Powered By Visual Paradigm Community Edition

4) Getting a ticket



3.2.1.2 Functional Requirements - Scenario 1

- **R1**: the system shall allow the user to register for the application and log in it.
- **R2**: the system shall allow the user to send a request for joining a queue.
- **R3**: the system shall allow the user to choose the shop that he wants to visit.
- **R4**: the system shall allow the user to join a queue using a ticket machine.
- **R5**: the system shall allow the digital user who joins a queue to optionally communicate his position using GPS or specifying an address.
- **R6**: the system shall allow the user to optionally communicate an expected duration of the visit inside the shop.
- **R7**: If the user does not insert any expected time spent into the shop, the system calculates it analyzing statistics.
- **R8**: the system stores the data about users and their active and past requests.
- **R9**: the system provides a unique QR code for the specific user and request.
- **R10**: the system calculates dynamically the remaining time for each digital user in the queue using the expected time specified or estimated by CLup.
- **R11**: the system shall allow the user to check the state and the details of his queue active requests.

3.2.2 Use Cases - Scenario 2: Booking a visit

Use Cases

5) Visualizing request information

Name	Visualizing request information
Actor	Digital User
Entry Condition	The user has successfully performed at least or one “Joining the queue” request or a “Book a visit” request. The user is already logged in and is interested in knowing information of a request.
Event Flow	1. The user clicks on the ”Menu” button. 2. The user chooses the “Your request” option. 3. The user clicks on the interested booking or joining request. 4. The system displays all the related information.
Exit Condition	The user visualizes all the useful details about the service chosen.
Special Requirement	The information related to services, as the remaining time for the lining up is updated every minute.

6) Canceling a request

Name	Canceling a request
Actor	Digital User
Entry Condition	The user has successfully performed at least or one “Joining the queue” request or a “Book a visit” request. The user is already logged in and wants to cancel a specific request.
Event Flow	1. The user clicks on the ”Menu” button. 2. The user visualizes the “Your request” page. 3. The user clicks on the “Cancel” button related to the service that wants to cancel. 4. The user clicks “Yes” on “Do you really want to cancel this request?” question.

	5. The system confirms the request and updates the "Your Request" page.
Exit Condition	The user is correctly removed from the queue or his booing is deleted. Then he's redirected to the homepage.
Exception	1. The remaining time is less than 5 minutes and the cancel request is not available.

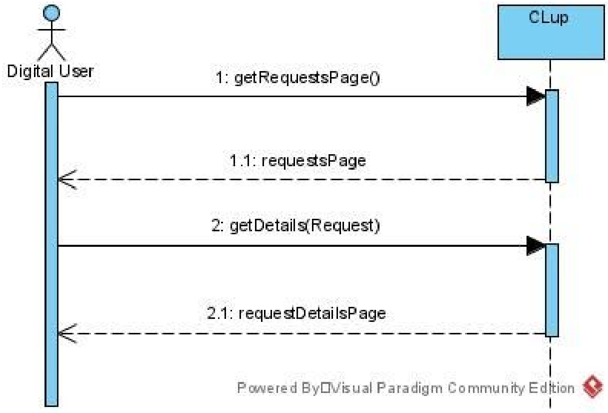
7) Booking a visit

Name	Booking a visit
Actor	Digital User
Entry Condition	The user is already logged in and needs to visit a grocery store without waiting his turn in a queue.
Event Flow	1. The user visualizes the homepage. 2. The user clicks on the "Book a visit" button. 3. The user searches a grocery store, by indicating its name, from the list of registered shops that displays their names and addresses. 4. The user clicks on the shop he wants to visit. 5. The user can write in the optional field kinds of products that he is going to purchase. 6. The user can write in the optional field the expected time of his visit. 6.1. If the user does not insert any expected time, it is calculated by the application analyzing statistics. 7. The user has to insert in the mandatory field the date and the hour at which he wants to book the visit. 8. The user clicks on the "Send Request" button. 9. The system confirms the request and saves the data.
Exit condition	The system notifies the user the booking request success and displays him the date and hour, the expected time and the shop and the QR code to access it.
Exception	1. The inserted shop is not found. The system will show an error message offering the possibility to

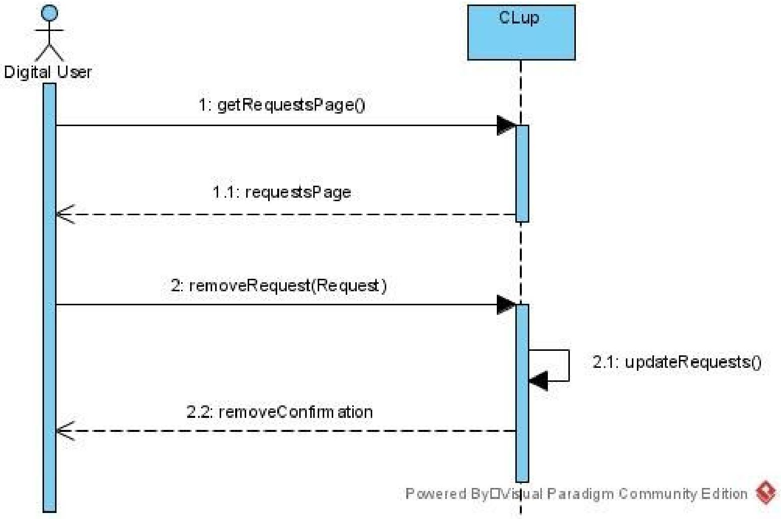
	choose another store. 2. The sent request is rejected because at the specified date and hour the selected grocery store is closed. The system reacts blocking the request and showing an error notification. 3. The sent request is rejected because for the specified date and hour the store is full yet, all the possible spots are booked. The system reacts blocking the request and showing an error notification. 4. The user doesn’t fill the form completely or not in a correct way. The system notifies which fields are not yet compiled or shows the errors occurred.

3.2.2.1 Sequence Diagrams - Scenario 2

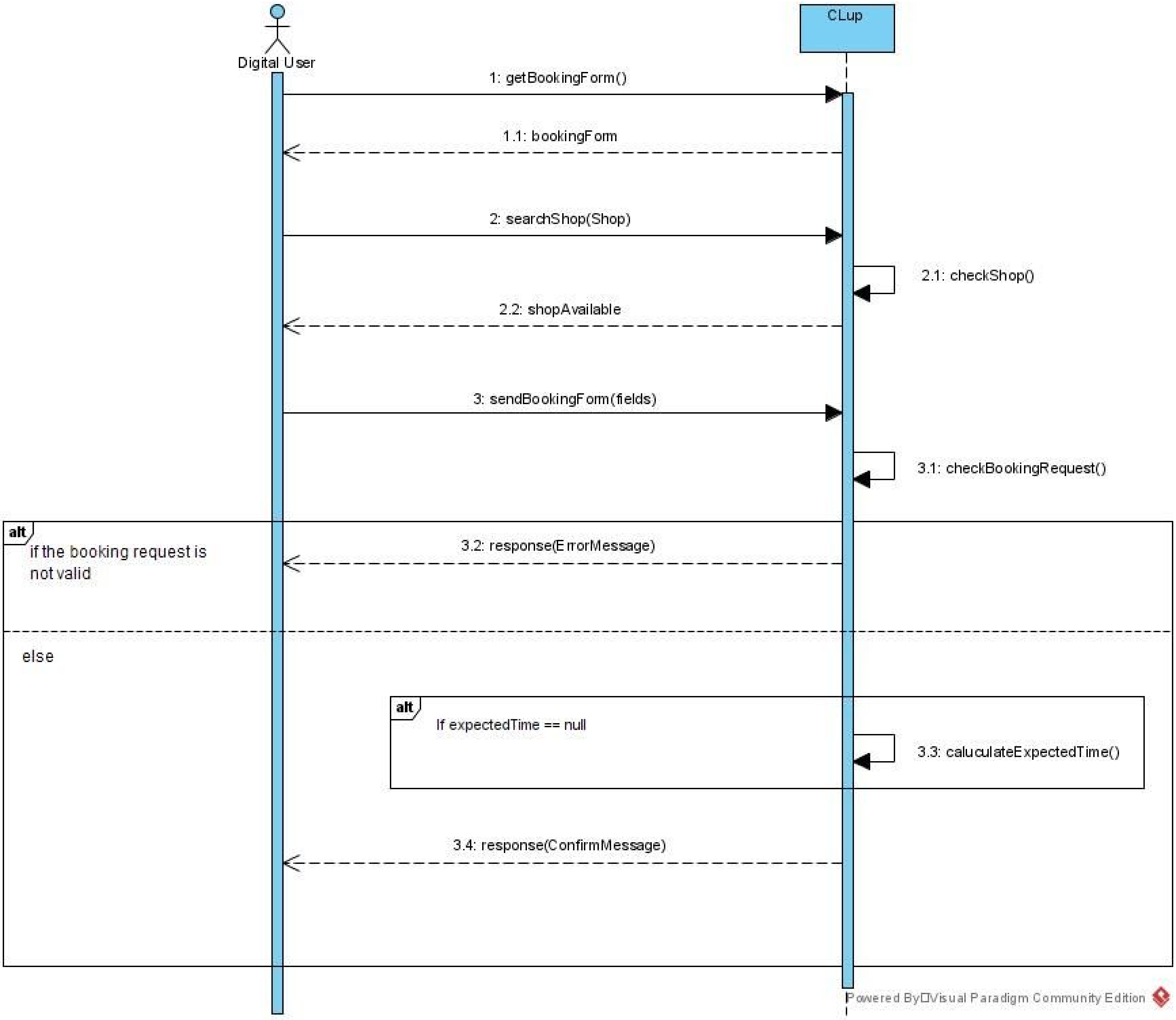
5) Visualizing request information



6) Canceling a request



7) Booking a visit


Digital User
CLup
1: getBookingForm()
1.1: bookingForm
2: searchShop(Shop)
2.1: checkShop()
2.2: shopAvailable
3: sendBookingForm(fields)
3.1: checkBookingRequest()
alt
if the booking request is not valid
3.2: response(ErrorMessage)
else
alt
If expectedTime == null
3.3: caluculateExpectedTime()
3.4: response(ConfirmMessage)
Powered By Visual Paradigm Community Edition

3.2.2.2 Functional Requirements - Scenario 2

- **R1, R3, R6, R7, R8, R9**
- **R12**: the system shall allow the user to check the state and the details of his booking active requests, in particular the shop and the booking date and hour.
- **R13**: the system shall allow the digital user to cancel the required service.
- **R14**: the system shall allow the user to book a visit to a grocery shop indicating the specific date and hour.
- **R15**: the system shall allow the booked user to optionally specify the kind of products he wants to purchase.
- **R16**: the system shall allow a user to book a reservation if, for the chosen date and hour, the shop is full yet but, specifying types of product, the user has to visit different departments than all the other customers. This user is considered as an “Overflow User”.

3.2.3 Use Cases - Scenario 3: A shop subscribes to CLup

Use Cases

8) Subscription of a shop

Name	Subscription of a shop
Actor	Shop
Entry Condition	The shop owner opens the application CLup on his device.
Event Flow	1. The owner clicks on “Menu” button 2. The owner clicks on “Do you own a shop? Sign up here”. 3. The owner fills all the mandatory fields. 4. The owner attaches necessary documentation that proves the authenticity and characteristics of his activity. 5. The owner clicks on the “Confirm” button. 6. The request is sent to the system that analyzes it. 7. The system sends a confirmation notification and stores the relative data.
Exit Condition	The shop is registered and the system stores its data.
Exception	1. The shop owner does not fill all the mandatory fields. The systems display an error and underline the fields not yet compiled. 2. The request to sign up is rejected because the documentation is not valid. The system displays the motivations. 3. The inserted credentials are already used. The system invites the shop to change them.

9) Login of a shop

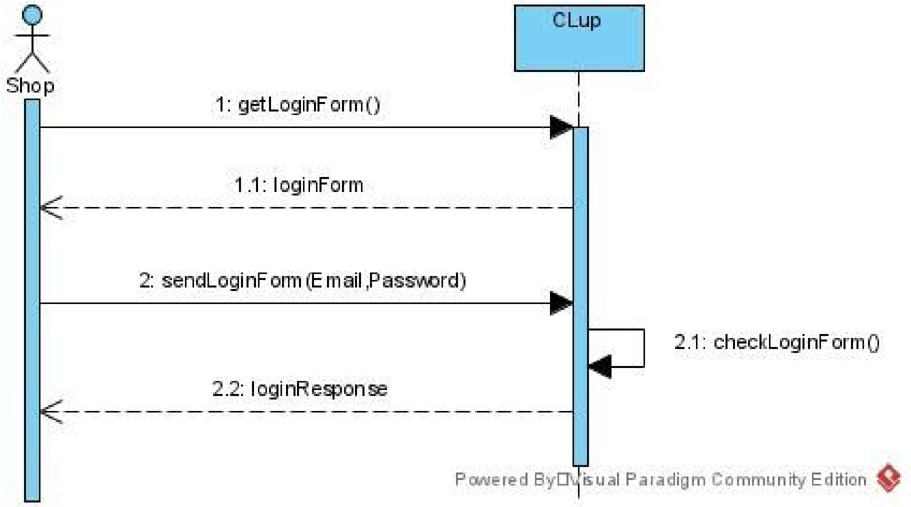
Name	Login of a shop
Actor	Shop
Entry Condition	The shop owner opens the app CLup on his device and his shop is already signed up.
Event Flow	1. The owner clicks on the “Menu” button. 2. The owner clicks on the “Login” button. 3. The owner inserts in the form his “Email” and “Password” credentials . 4. The owner clicks on the “Enter” button. 5. The system confirms the access.
Exit Condition	The owner is logged in and he is redirected to the homepage where he can take advantage of CLup thanks to special functionality reserved to shop’s accounts.
Exception	1. The inserted email doesn’t correspond to any existing account. 2. The inserted password is not correct. In both cases, the system notifies the owner to refill the login form.

10) Visualizing clients’ requests

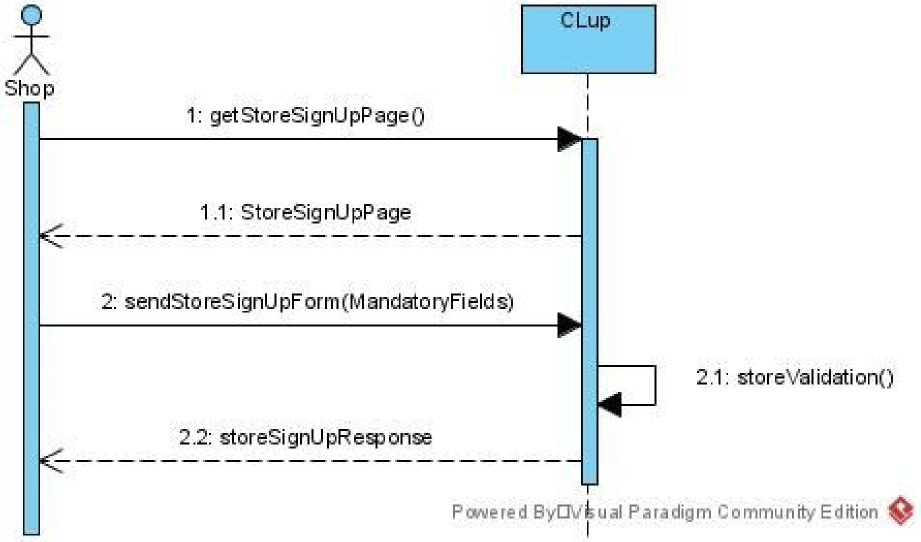
Name	Visualizing clients’ requests
Actor	Shop
Entry Condition	The shop owner is already logged into CLup.
Event Flow	1. The owner clicks on the “Menu” button. 2. The owner clicks on the “Clients’ requests” button 3. The owner chose what he wants to visualize between the booked clients, the clients in queue and a list of the previous visits’ clients related to that specific shop.
Exit Condition	The shop owner visualizes requests of the chosen type.

3.2.3.1 Sequence Diagrams - Scenario 3

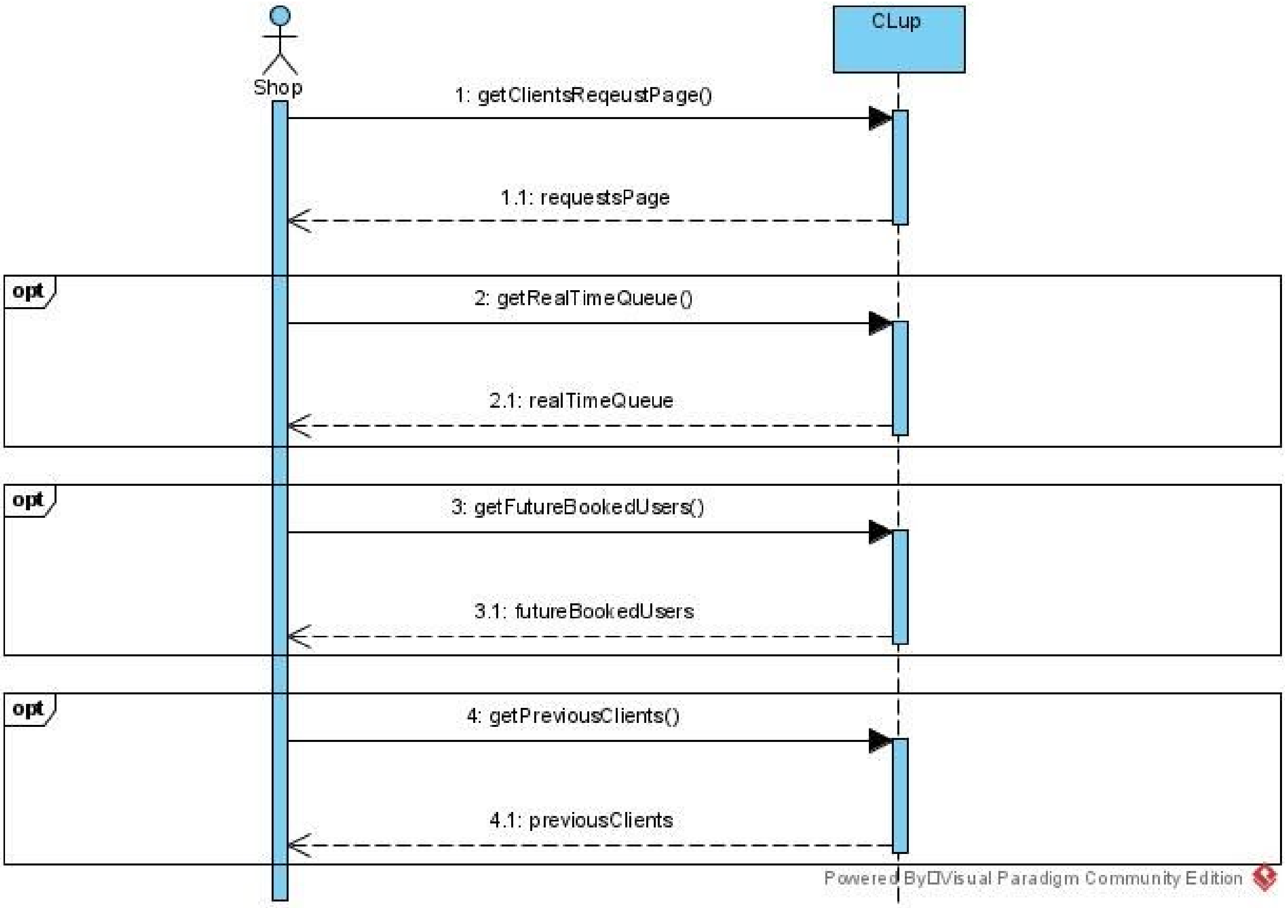
8) Subscription of a shop



9) Login of a shop



10) Visualizing clients request



3.2.3.2 Functional Requirements - Scenario 3

- **R8**.
- **R17**: The system shall allow the shop owner to register their shop to the application and log in it.
- **R18**: the system shall allow the shop to communicate the necessary documentation, store dimensions, maximum number of people inside of it and other information.
- **R19**: the system stores the data about the registered shops.
- **R20**: the system shall allow the shop to visualize the related real time queue, booked clients and the past requests.

3.2.4 Use Cases - Scenario 4: Receiving a notification and approaching the shop

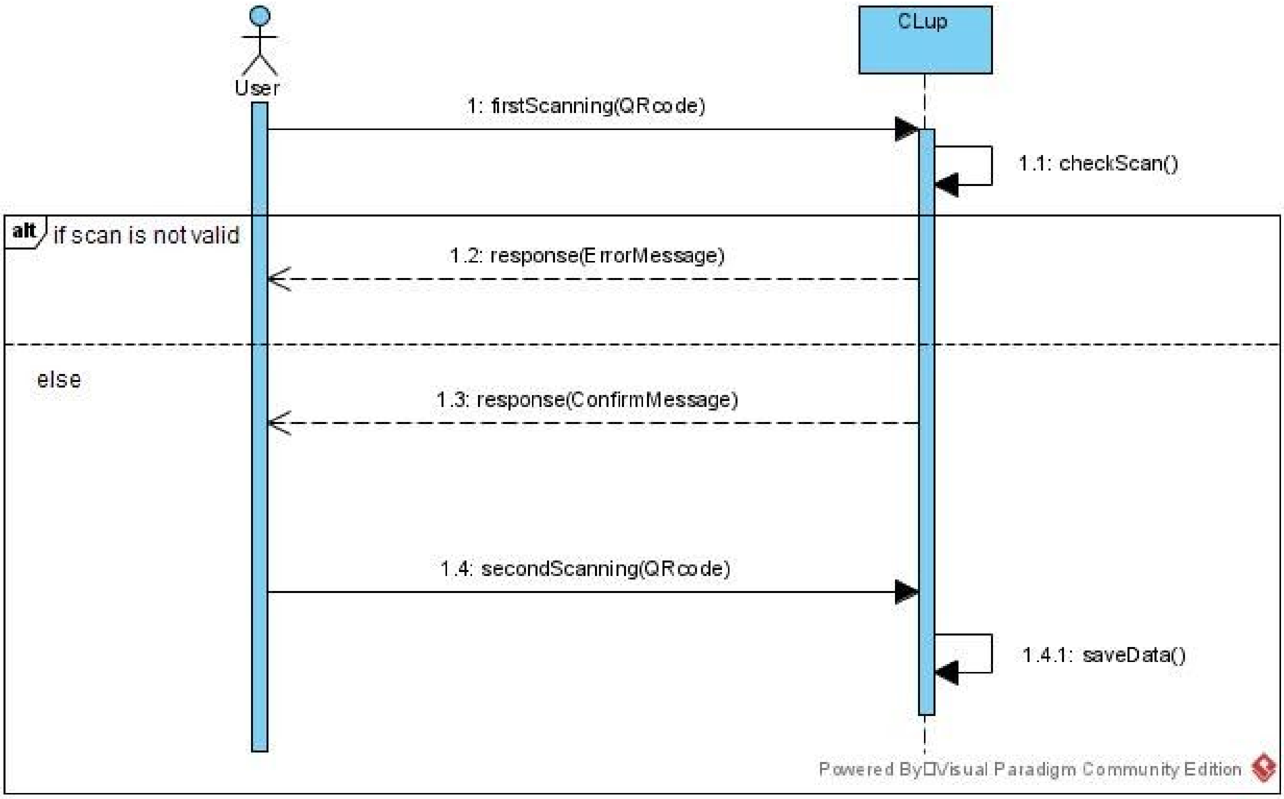
Use Cases

11) Entering and leaving the shop

Name	Entering and leaving the shop
Actor	Digital User
Entry Condition	The user is at the entrance of the shop and wants to enter. The user has his QR code.
Event Flow	1. The user scans the QR code by means of an appropriate scanner. 2. The system evaluates the scanning operation and sends a confirmation. 3. The user enters the shop. 4. The user rescan the QR code before leaving. 5. The system stores the visit data and calculates the effective time spent in the store.
Exit Condition	The system communicates the scanning success and the user leaves.
Exception	1. The system can negatively evaluate the operation because the shop is full. CLup invites the user to wait near the shop. 2. The system can negatively evaluate the operation because it is not yet the user's turn. CLup invites the user to wait near the shop. 3. The system can negatively evaluate the operation because the user has arrived too much time later with respect to the remaining time or the date of the booking.

3.2.4.1 Sequence Diagrams - Scenario 4

11) Entering and leaving a shop


User
CLup
1: firstScanning(QRcode)
1.1: checkScan()
alt
if scan is not valid
1.2: response(ErrorMessage)
else
1.3: response(ConfirmMessage)
1.4: secondScanning(QRcode)
1.4.1: saveData()
Powered By Visual Paradigm Community Edition

3.2.4.2 Functional Requirements - Scenario 4

- **R1, R2, R3, R5, R6, R7, R8, R9, R10, R11.**
- **R21**: If the user scans the QR code for entering when the shop is full but it is his turn the system allows him to enter if and only if he’s an “overflow” user, otherwise the system rejects the scanning operation and shows an error message.
- **R22**: If the user scans the QR code for entering when it is not his turn, the system does not allow the user to enter by rejecting the scanning operation and showing an error message.
- **R23**: If the user scans the QR code for entering when the shop is not full and it’s his turn, the system allows the user to enter it confirming the scanning operation.
- **R24**: the system sends notifications to make the digital user aware of the incoming shop visit considering the specified position and the remaining time.

3.2.5 All requirements

R1	The system shall allow the user to register for the application and log in it.
R2	The system shall allow the user to send a request for joining a queue.
R3	The system shall allow the user to choose the shop that he wants to visit.
R4	The system shall allow the user to join a queue using a ticket machine.
R5	The system shall allow the digital user who joins a queue to optionally communicate his position using GPS or specifying an address.
R6	The system shall allow the user to optionally communicate an expected duration of the visit inside the shop.
R7	If the user does not insert any expected time spent into the shop, the system calculates it analyzing statistics.
R8	The system stores the data about users and their active and past requests.
R9	The system provides a unique QR code for the specific user and request.
R10	The system calculates dynamically the remaining time for each digital user in the queue using the expected time specified or estimated by CLup.
R11	The system shall allow the user to check the state and the details of his queue active requests.
R12	The system shall allow the user to check the state and the details of his booking active requests, in particular the shop and the booking date and hour.
R13	The system shall allow the digital user to cancel the required service.
R14	The system shall allow the user to book a visit to a grocery shop indicating the specific date and hour.
R15	The system shall allow the booked user to optionally specify the kind of products he wants to purchase.
R16	The system shall allow a user to book a reservation if, for the chosen date and hour, the shop is full yet but, specifying types of product, the user has to visit different departments than all the other customers. This user is considered as an “Overflow User”.

R17	The system shall allow the shop owner to register their shop to the application and log in it.
R18	The system shall allow the shop to communicate the necessary documentation, store dimensions, maximum number of people inside of it and other information.
R19	The system stores the data about the registered shops.
R20	The system shall allow the shop to visualize the related real time queue, booked clients and the past requests.
R21	If the user scans the QR code for entering when the shop is full but it is his turn the system allows him to enter if and only if he's an "overflow" user, otherwise the system rejects the scanning operation and shows an error message.
R22	If the user scans the QR code for entering when it is not his turn, the system does not allow the user to enter by rejecting the scanning operation and showing an error message.
R23	If the user scans the QR code for entering when the shop is not full and it's his turn, the system allows the user to enter it confirming the scanning operation.
R24	The system sends notifications to make the digital user aware of the incoming shop visit considering the specified position and the remaining time.

3.3 Goals, requirements and assumptions

G1: The system prevents the creation of physical queues at the entrance of shops.

- ❖ D2**:** Each shop respects the opening period.
- ❖ D5: Each user scans the QR code only one time before entering and before leaving the shop.
- ❖ D6: Each user can't spend more than 5 minutes over the expected time expressed by himself or estimated by the system inside the shop.
- ❖ D7: Every digital user disposes of a working internet connection to join a queue or to book a visit.
- ❖ D9: If a user scans the unique QR code for getting in the shop and the operation is successful, he enters the shop.

- R1: the system shall allow the user to register for the application and log in it.
- R2: the system shall allow the user to send a request for joining a queue.
- R3: the system shall allow the user to choose the shop that he wants to visit.
- R4: the system shall allow the user to join a queue using a ticket machine.
- R5: the system shall allow the digital user who joins a queue to optionally communicate his position using GPS or specifying an address.
- R6: the system shall allow the user to optionally communicate an expected duration of the visit inside the shop.
- R7: If the user does not insert any expected time spent into the shop, the system calculates it analyzing statistics.
- R9: the system provides a unique QR code for the specific user and request.
- R10: the system calculates dynamically the remaining time for each digital user in the queue using the expected time specified or estimated by CLup.
- R11: the system shall allow the user to check the state and the details of his queue active requests.
- R13: the system shall allow the digital user to cancel the required service.
- R17: The system shall allow the shop owner to register their shop to the application and log in it.
- R22: If the user scans the QR code for entering when it is not his turn, the system does not allow the user to enter by rejecting the scanning operation and showing an error message.
- R23: If the user scans the QR code for entering when the shop is not full and it’s his turn, the system allows the user to enter it confirming the scanning operation.
- R24: the system sends notifications to make the user aware of the incoming shop visit considering the specified position and the remaining time.

G2: The system offers the opportunity to visit shops avoiding wasting time in the queue.

- ❖ D1: When a user inserts in a booking service the product required, this one is sold by the grocery shop.
- ❖ D2: Each shop respects the opening period.
- ❖ D5: Each user scans the QR code only one time before entering and before leaving the shop.
- ❖ D7: Every digital user disposes of a working internet connection to join a queue or to book a visit.
- ❖ D9: If a user scans the unique QR code for getting in the shop and the operation is successful, he enters the shop.

- R1: the system shall allow the user to register to the application and log in it.
- R3: the system shall allow the user to choose the shop that he wants to visit.
- R6: the system shall allow the user to optionally communicate an expected duration of the visit inside the shop.
- R7: If the user does not insert any expected time spent into the shop, the system calculates it analyzing statistics.
- R9: the system provides a unique QR code for the specific user and request.
- R12: the system shall allow the user to check the state and the details of his booking active requests, in particular the shop and the booking date and hour.
- R13: the system shall allow the digital user to cancel the required service.
- R14: the system shall allow the user to book a visit to a grocery shop indicating the specific date and hour.
- R15: the system shall allow the user to optionally specify the kind of products he wants to purchase.
- R16: the system shall allow a user to book a reservation if, for the chosen date and hour, the shop is full yet but, specifying types of product, he has to visit different departments than all the other customers. This user is considered as an "Overflow User".
- R17: The system shall allow the shop owner to register their shop to the application and log in it.
- R21: If the user scans the QR code for entering when the shop is full but it is his turn the system allows him to enter if and only if he's an "overflow" user, otherwise the system rejects the scanning operation and shows an error message.
- R22: If the user scans the QR code for entering when it is not his turn, the system does not allow the user to enter by rejecting the scanning operation and showing an error message.
- R23: If the user scans the QR code for entering when the shop is not full and it's his turn, the system allows the user to enter it confirming the scanning operation.

G3: The system helps grocery's owners to manage the client's influx to shops.

- ❖ D2: Each shop respects the opening period.
- ❖ D5: Each user scans the QR code only one time before entering and before leaving the shop.
- ❖ D9: If a user scans the unique QR code for getting in the shop and the operation is successful, he enters the shop.
- ❖ D10: If the user scans the unique QR code for leaving the shop, he actually goes outside of it.

- ❖ D11: If the scanning operation is rejected, the user doesn't get into the shop.

- R1: the system shall allow the user to register for the application and log in it.
- R2: the system shall allow the user to send a request for joining a queue.
- R3: the system shall allow the user to choose the shop that he wants to visit.
- R4: the system shall allow the user to join a queue using a ticket machine.
- R8: the system stores the data about the users and their active and past requests.
- R9: the system provides a unique QR code for the specific user and request.
- R14: the system shall allow the user to book a visit to a grocery shop indicating the specific date and hour.
- R17: The system shall allow the shop owners to register their shop to the application and log in it.
- R18: the system shall allow the shop to communicate the necessary documentation, store dimensions, the maximum number of people inside of it and other information.
- R19: the system stores the data about the registered shops.
- R20: the system shall allow the shop to visualize the related real time queue, booked clients and the past requests.
- R21: If the user scans the QR code for entering when the shop is full but it is his turn the system allows him to enter if and only if he's an "overflow" user, otherwise the system rejects the scanning operation and shows an error message.
- R22: If the user scans the QR code for entering when it is not his turn, the system does not allow the user to enter by rejecting the scanning operation and showing an error message.
- R23: If the user scans the QR code for entering when the shop is not full and it's his turn, the system allows the user to enter it confirming the scanning operation.

G4: The system allows the respect of the social distancing policy into the shop.

- ❖ D1: When a user inserts in a booking service the product required, this one is sold by the grocery shop.
- ❖ D3: Each shop communicates the correct dimension of the structure.
- ❖ D4: Each shop communicates the correct number of contemporary people inside of it respecting the physical distancing policy.
- ❖ D5: Each user scans the QR code only one time before entering and before leaving the shop.
- ❖ D8: Each booked user that specifies the kinds of product to purchase visits only the related departments of the shop.
- ❖ D9: If a user scans the unique QR code for getting in the shop and the operation is

successful, he enters the shop.

- ❖ D10: If the user scans the unique QR code for leaving the shop, he actually goes outside of it.
- ❖ D11: If the scanning operation is rejected, the user doesn't get into the shop.

- R1: the system shall allow the user to register for the application and log in it.
- R2: the system shall allow the user to send a request for joining a queue.
- R3: the system shall allow the user to choose the shop that he wants to visit.
- R4: the system shall allow the user to join a queue using a ticket machine.
- R9: the system provides a unique QR code for the specific user and request.
- R14: the system shall allow the user to book a visit to a grocery shop indicating the specific date and hour.
- R15: the system shall allow the user to optionally specify the kind of products he wants to purchase.
- R16: the system shall allow a user to book a reservation if, for the chosen date and hour, the shop is full yet but, specifying types of product, he has to visit different departments than all the other customers. This user is considered as an "Overflow User".
- R17: The system shall allow the shop owners to register their shop to the application and log in it.
- R18: the system shall allow the shop to communicate the necessary documentation, store dimensions and the maximum number of people inside of it.
- R21: If the user scans the QR code for entering when the shop is full but it is his turn the system allows him to enter if and only if he's an "overflow" user, otherwise the system rejects the scanning operation and shows an error message.
- R22: If the user scans the QR code for entering when it is not his turn, the system does not allow the user to enter by rejecting the scanning operation and showing an error message.
- R23: If the user scans the QR code for entering when the shop is not full and it's his turn, the system allows the user to enter it confirming the scanning operation.

3.3.1 Traceability matrix

Goal	Requirements	Domain Assumption
G1	R1, R2, R3, R4, R5, R6, R7, R9, R10, R11, R13, R17, R22, R23, R24	D2, D5, D6, D7, D9
G2	R1, R3, R6, R7, R9, R12, R13, R14, R15, R16, R17, R21, R22, R23	D1, D2, D5, D7, D9
G3	R1, R2, R3, R4, R8, R9, R14, R17, R18, R19, R20, R21, R22, R23	D2, D5, D9, D10, D11
G4	R1, R2, R3, R4, R9, R14, R15, R16, R17, R18, R21, R22, R23	D1, D3, D4, D5, D8, D9, D10, D11

3.4 Performance Requirements

The system should be able to receive, check and save the sending request by the user in less than 10 second. Being a service that guarantees the safety of people, CLup has to be reactive and reliable. For these reasons it must be able to handle and serve a large number of users and shops simultaneously.

3.5 Design Constraints

3.5.1 Standards compliance

Since CLup manages sensitive information regarding users, the project is subject to the General Data Protection Regulation (GDPR). It was designed to protect the privacy and security of all personal data collected for or about EU citizens, especially as it relates to processing, using or exchanging such data. These data protection rules applied not only when responsible parties are established or operated within the EU, but also when the controller used equipment located inside the EU to process personal data.

3.5.2 Hardware limitations

Users and shops can take advantage of CLup using their mobile devices. So, it is necessary to have an Internet connection when using the system.

Each registered shop has a ticket machine located near the structure. At the entrance of all the registered shops, there must be a scanner with an integrated display. The display is used to communicate when the physical user scans his QR code printed on the ticket before entering.

3.5.3 Any other Constraints

When a user joins a queue, he can't make other queue requests towards any shop until the programmed visit is not finished or he cancels it.

3.6 Software System Attributes

3.6.1 Reliability

The system should be available for an agreed period without interruption. Reliability depends on the component used to provide the service. Therefore it is necessary, preventive maintenance to prevent downtime but it is also important to handle them. For this reason, the system has a fully backup system in different places that replicates the running processes and the central server.

3.6.2 Availability

The service should be continuously available for the user. The system should be up for 99.99% of time. The functionalities offered are extremely important to avoid the virus expansion, so the downtimes should be as short as possible. In this case the average time between the occurrence of a fault and its recovery should be no more than 52 minutes/year.

3.6.3 Security

The data provided by the user contains sensitive information so it is extremely important to adopt all the defensive measures to eliminate software weaknesses. Therefore, passwords and other user data stored are encrypted.

3.6.4 Maintainability

The development of the application has to follow predetermined patterns in order to make the system easy to fix and update. Therefore, also a good level of abstraction is required.

3.6.5 Portability

The software must support Android and iOS operating systems for mobile devices.

3.7 Other Requirements

The real-time queue and the relative remaining time is updated every 1 minute.

Chapter 4

Formal analysis using Alloy

In this section, the analysis of some critical aspects is provided exploiting alloy. The focus is on some static constraint:

- Each digital and physical user has unique generalities but it can happen that a registered user could take advantage of the ticket machine located at the shop. In this specific case the user is considered as digital and physical at the same time; he will communicate the same personal information to use both services.
- Each user can join only one queue at a time.
- Only digital users can insert a position when joining a queue.
- Each user has to specify a starting position different from the shop that he wants to visit.

Then, other structural constraints of the UML diagram previously shown are expressed and properly commented in the Alloy formal notation. It is important to note that we have not defined in a precise way all the attributes of the signatures to make the Alloy model clearer and readable. Moreover, not all the system constraints are considered because Alloy has a different purpose and his notation does not allow to express them.

4.1 Alloy Code

```
-- SIGNATURES

abstract sig User{
   generalities: one Generalities
}


abstract sig AcceptedRequest{
   expectedTime: one Int,
   qrCode: one QR
}{expectedTime>0}


abstract sig Position{}
```

```
-- Generalities is a unique attribute made up of a name,surname and
-- a telephone number
sig Generalities{}
sig GPSPosition extends Position{}
sig Address extends Position{}
sig QR{}
sig Date{}
sig Email{}
sig Password{}
sig KindsOfProduct{}
sig Ticket{}

-- Digital Users are not related to any tickets.
sig DigitalUser extends User{
   performs: set AcceptedRequest,
   email: one Email,
   pw: one Password
}{ performs.ticket = none}

--Physical user can' insert a position when joining a queue.
sig PhysicalUser extends User{
   joins: some QueueRequest
} {joins.startsFromPosition = none}

sig QueueRequest extends AcceptedRequest{
   ticket: lone Ticket,
   remainingTimeQueue: one Int,
   startsFromPosition: lone Position
}{remainingTimeQueue>0}

sig BookingRequest extends AcceptedRequest{
   atDate: one Date,
   wantToBuy: lone KindsOfProduct
}

sig Shop{
   receives: set AcceptedRequest,
   isLocatedAt: one Address,
   containsTotPeople: one Int
}{containsTotPeople>0}

--FACTS
--Each request has to be related to one user.
```

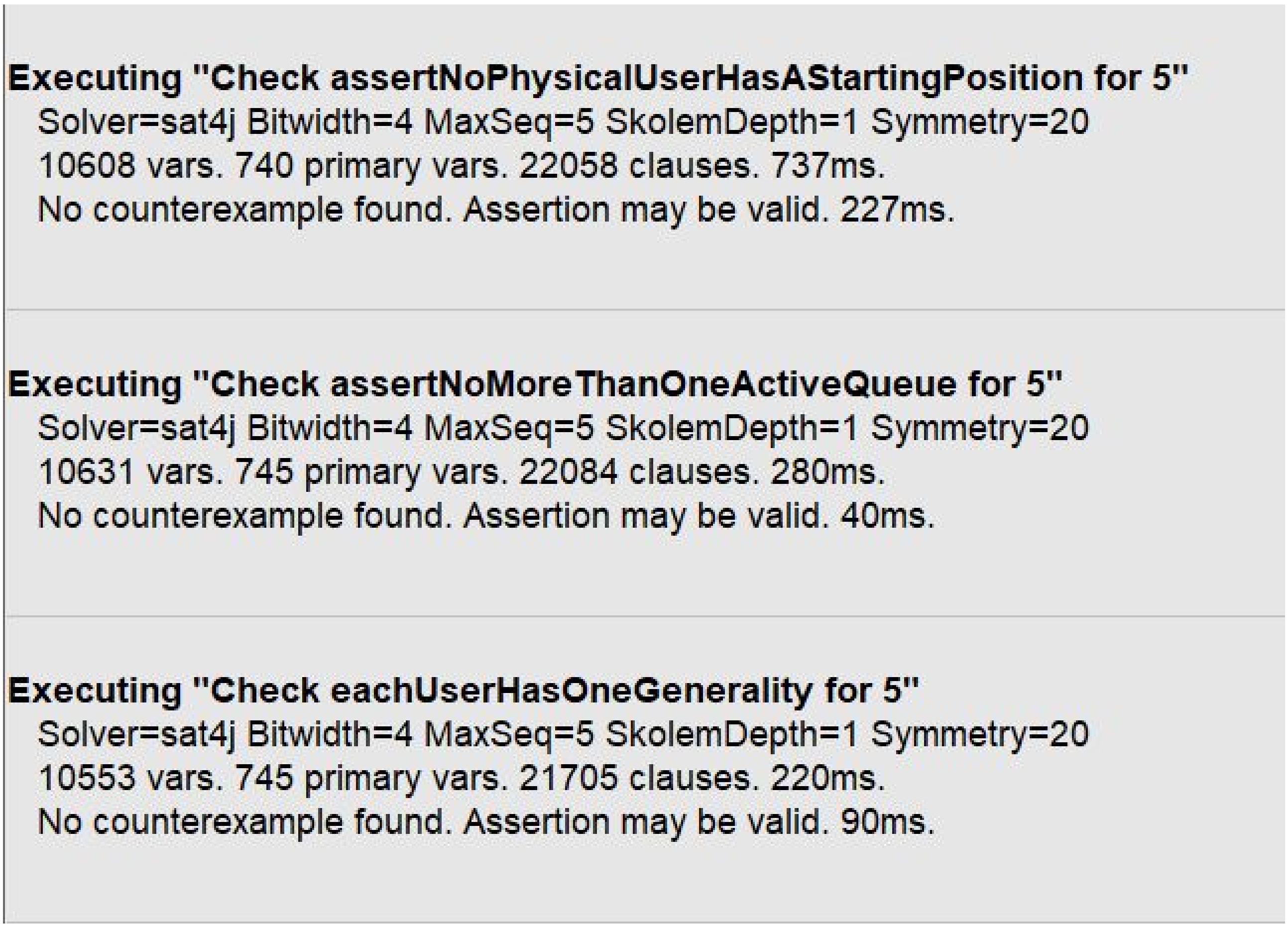
```
fact eachRequestIsRelatedToOneUser{
   all r: AcceptedRequest |  one u:User |
   isAUserRequest[r,u]
}


-- Each request has to be related to one shop

fact eachRequestIsRelatedToOneShop{
   all r: AcceptedRequest | one s:Shop |
   isAShopRequest[r,s]
}


--Each Qr is unique
fact eachQrCodeIsUnique{
   all disj q1,q2 : QR | q1!=q2
}

--Each Qr code has to be related to one request
fact eachQrCodeIsRelatedToOneRequest{
   all q: QR | one r:AcceptedRequest |
   q in r.qrCode
}

-- Each generality is unique for each kind of user.
fact eachGeneralitiesIsUniqueForEachDigitalUser{
   no g: Generalities | some u1,u2: DigitalUser|
   u1!=u2 and g in u1.generalities and g in u2.generalities
}

fact eachGeneralitiesIsUniqueForEachPhysicalUser{
   no g: Generalities | some u1,u2: PhysicalUser|
   u1!=u2 and g in u1.generalities and g in u2.generalities
}

--Each generalities has to be related to at least one user. One
physical User
-- could be associated to the generalities of a digital one if he is
registered the
-- application and he also takes advantage of the ticket machine.
fact eachGeneralityIsRelatedToUser{
   all g: Generalities | some u: User|
   g in u.generalities
```

```
}


--Each position has to be related to at least one digital queue request
-- or one shop.
fact eachPositionIsrelatedToAShopOrToAQueueRequest {
   all p: Position | some q: QueueRequest |
   p in q.startsFromPosition
      or
   all p1:Position| one s1:Shop|
   p1 in s1.isLocatedAt
}


--Each mail has to be related to only one digital user
fact eachEMailIsRelatedToOneDigitalUser{
   all e: Email| one d: DigitalUser |
   e in d.email
}

--Each password has to be related to at least one digital user
fact eachPasswordIsRelatedToDigitalUser{
   all p: Password| some d: DigitalUser|
   p in d.pw
}

--Each date has to be related to at least one booking
fact eachDateIsRelatedToBooking{
   all d: Date| some b: BookingRequest|
   d in b.atDate
}

-- Each user can join at most one queue at the same time
fact oneActiveQueueForEachUser{
   no u: User | some r1,r2: QueueRequest|
   r1!=r2 and  isAUserRequest[r1,u] and isAUserRequest[r2,u]
}

-- An user who use a ticket to join a queue can't also use the
application as digital
-- to join another queue.
fact oneActiveQueue{
   no q1,q2: QueueRequest | some u1,u2:User|
   q1!=q2 and u1!=u2  and isAUserRequest[q1,u1] and
```

```
u1.generalities=u2.generalities
   and isAUserRequest[q2,u2]
}

--Each kind of product has to be related to at least one booking
request.
fact eachKindsOfProductIsRelatedToBooking{
   all k: KindsOfProduct| some b1: BookingRequest|
   k in b1.wantToBuy
}

-- Each user queue specifies a starting position different from the
shop that he
-- wants to visit.
fact eachStartingPositionIsDifferentFromTheShopTheyWantToVisit{
   no q: QueueRequest | one s:Shop |
   q in s.receives and q.startsFromPosition=s.isLocatedAt
}

-- Each shop has a different address.
fact eachShopHasAnUniqueAddress{
   no a: Address| some s1,s2: Shop|
   s1!=s2 and a in s1.isLocatedAt and a in s2.isLocatedAt
}

--Each ticket has to be related to one queue request.
fact eachTicketIsRelatedToOneQueueRequest{
   all t: Ticket | one q: QueueRequest |
   t in q.ticket
}

--PREDICATES
pred isAShopRequest[ r: AcceptedRequest,s:Shop]{
   r in s.receives
}

pred isAUserRequest[r:AcceptedRequest, u:User]{
   r in u.performs or r in u.joins
}

--ASSERTIONS
assert assertNoPhysicalUserHasAStartingPosition{
   no q: QueueRequest | one p: PhysicalUser| one pos: Position|
   (q in p.joins and pos in q.startsFromPosition)
```

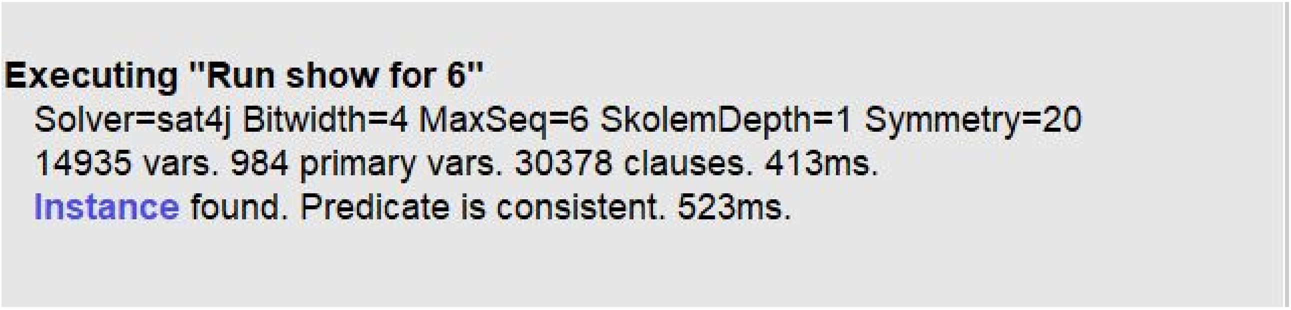
```
}

check assertNoPhysicalUserHasAStartingPosition for 5


assert assertNoMoreThanOneActiveQueue{
   no disj r1,r2: QueueRequest| one u:User|
   not(isAUserRequest[r1,u] and isAUserRequest[r2,u])
}

check assertNoMoreThanOneActiveQueue for 5


assert eachUserHasOneGenerality{
   no disj g1,g2: Generalities | all u:User|
   g1 in u.generalities and g2 in u.generalities
}

check eachUserHasOneGenerality for 5



pred show{
   #KindsOfProduct>0
   #Date>0
   #Password>0
   #Email>0
   #Generalities>0
   #QR>0
   #Address>0
   #GPSPosition>0
   #DigitalUser>0
   #PhysicalUser>0
   #QueueRequest>0
   #BookingRequest>0
   #Shop>0
   #Ticket>0
}

run show for 9
```

4.2 Assertions Results

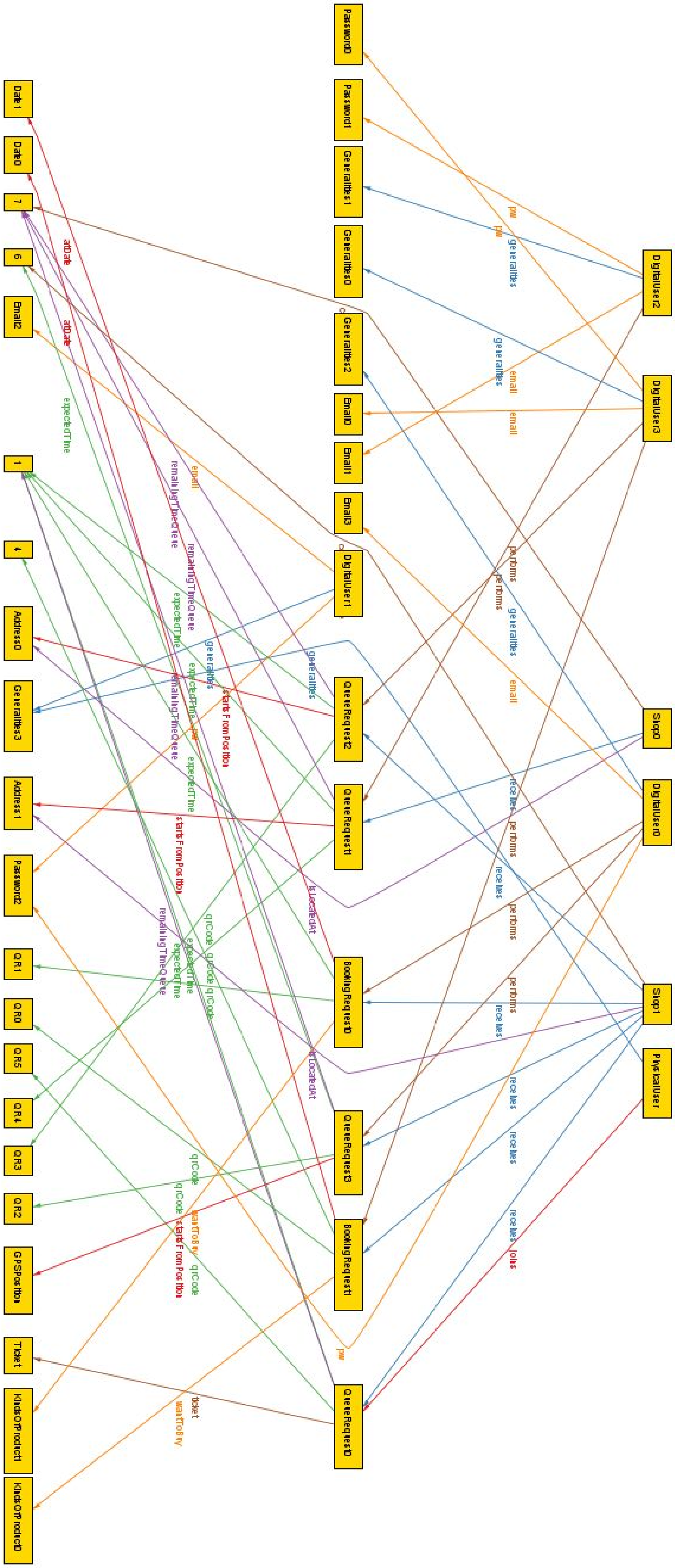


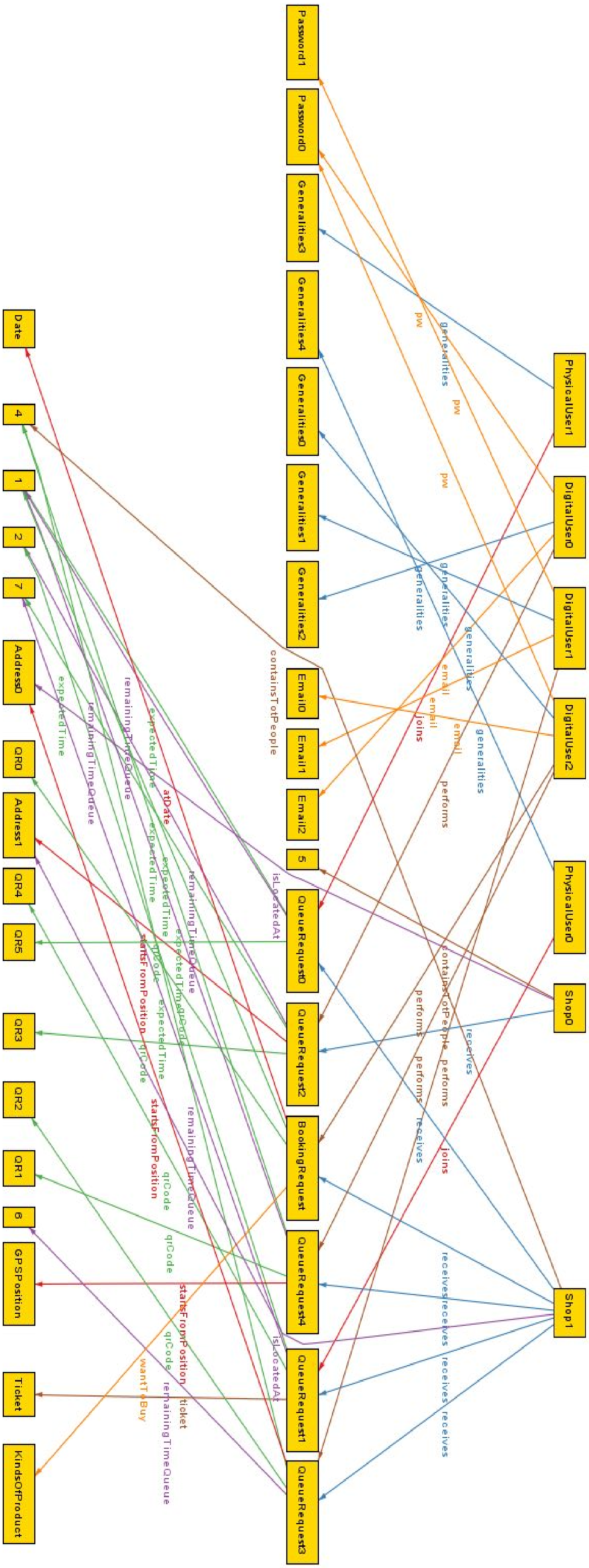
```
Executing "Check assertNoPhysicalUserHasAStartingPosition for 5"
   Solver=sat4j Bitwidth=4 MaxSeq=5 SkolemDepth=1 Symmetry=20
   10608 vars. 740 primary vars. 22058 clauses. 737ms.
   No counterexample found. Assertion may be valid. 227ms.

Executing "Check assertNoMoreThanOneActiveQueue for 5"
   Solver=sat4j Bitwidth=4 MaxSeq=5 SkolemDepth=1 Symmetry=20
   10631 vars. 745 primary vars. 22084 clauses. 280ms.
   No counterexample found. Assertion may be valid. 40ms.

Executing "Check eachUserHasOneGenerality for 5"
   Solver=sat4j Bitwidth=4 MaxSeq=5 SkolemDepth=1 Symmetry=20
   10553 vars. 745 primary vars. 21705 clauses. 220ms.
   No counterexample found. Assertion may be valid. 90ms.
```

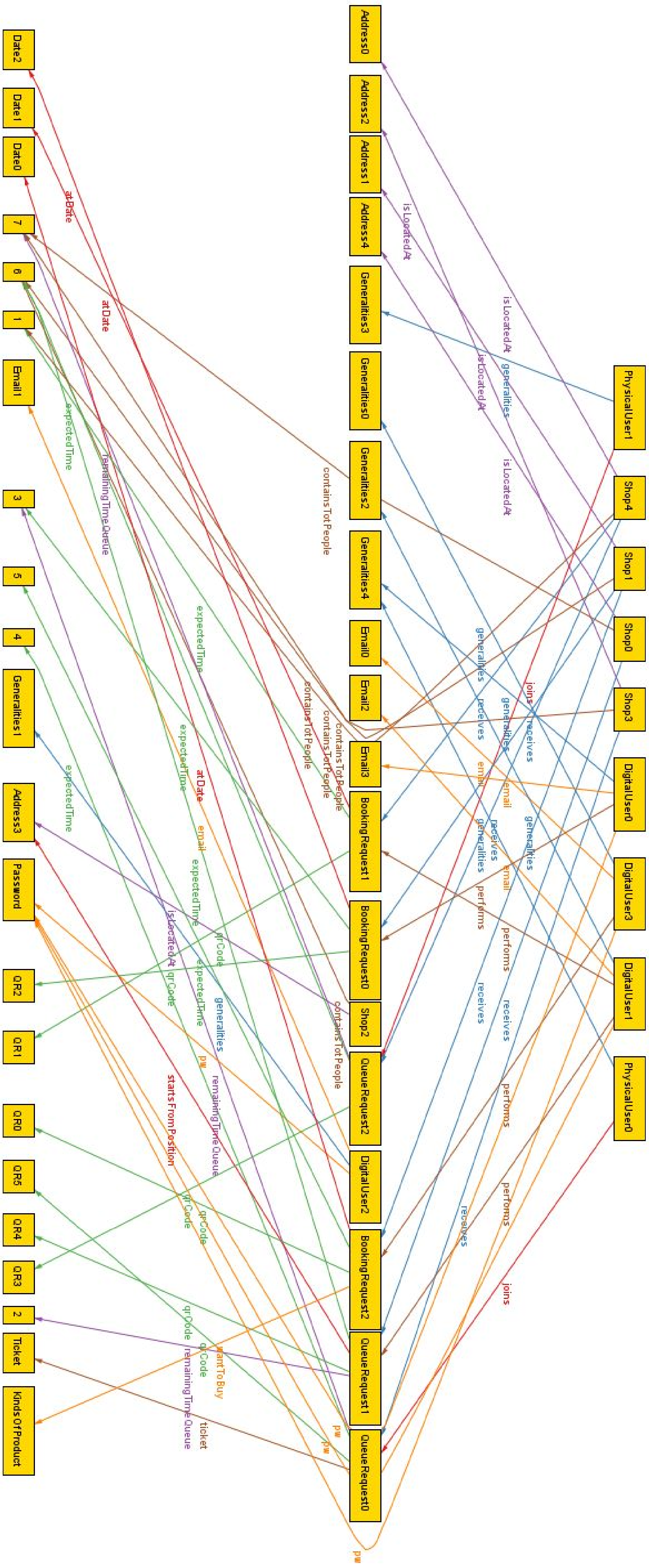
4.3 MetaModels



```
Executing "Run show for 6"
   Solver=sat4j Bitwidth=4 MaxSeq=6 SkolemDepth=1 Symmetry=20
   14935 vars. 984 primary vars. 30378 clauses. 413ms.
   Instance found. Predicate is consistent. 523ms.
```


DigitalUser2
DigitalUser3
Shop0
DigitalUser0
Shop1
PhysicalUser
Password0
Password1
Generalities1
Generalities0
Generalities2
Email0
Email1
Email3
DigitalUser1
QueueRequest2
QueueRequest1
BookingRequest0
QueueRequest3
BookingRequest1
QueueRequest0
Date1
Date0
7
6
Email2
1
4
Address0
Generalities3
Address1
Password2
QR1
QR0
QR5
QR4
QR3
QR2
GPSPosition
Ticket
KindsOfProduct1
KindsOfProduct0
pw
generalities
email
performs
receives
joins
atDate
expectedTime
remainingTimeQueue
startsFromPosition
isLocatedAt
qrCode
wantToBuy
ticket


Password1
Password0
Generalities3
Generalities4
Generalities0
Generalities1
Generalities2
Email0
Email1
Email2
5
QueueRequest0
QueueRequest2
BookingRequest
QueueRequest4
QueueRequest1
QueueRequest3
PhysicalUser1
DigitalUser0
DigitalUser1
DigitalUser2
PhysicalUser0
Shop0
Shop1
Date
4
1
2
7
Address0
QR0
Address1
QR4
QR5
QR3
QR2
QR1
8
GPSPosition
Ticket
KindsOfProduct
pw
generalities
email
joins
performs
receives
containsTotPeople
isLocatedAt
remainingTimeQueue
expectedTime
qrCode
atDate
startsFromPosition
ticket
wantToBuy



Address0
Address2
Address1
Address4
Generalities3
Generalities0
Generalities2
Generalities4
Email0
Email2
Email3
BookingRequest1
BookingRequest0
Shop2
QueueRequest2
DigitalUser2
BookingRequest2
QueueRequest1
QueueRequest0
PhysicalUser1
Shop4
Shop1
Shop0
Shop3
DigitalUser0
DigitalUser3
DigitalUser1
PhysicalUser0
Date2
Date1
Date0
7
6
1
Email1
3
5
4
Generalities1
Address3
Password
QR2
QR1
QR0
QR5
QR4
QR3
2
Ticket
KindsOfProduct
isLocatedAt
generalities
receives
joins
email
performs
containsTotPeople
atDate
expectedTime
remainingTimeQueue
qrCode
pw
startsFromPosition
wantToBuy
ticket

Chapter 5

Effort spent

Member 1: Andrea Razza (968483):

Topic	**Hours**
Chapter 1 - Introduction	4h
2.1 - Product perspective	8h
2.2, 2.3 - Product function and user characteristics	2.5h
2.4 - Constraints, assumptions and dependencies	1.5h
3.1 - External interface requirements	3h
3.2 - Functional requirements	15h
3.3 - Goals, requirements and assumptions	4h
3.4, 3.5, 3.6, 3.7 - Performance requirements, design constraints, software system attributes and other requirements	1h
Chapter 4 - Formal analysis using alloy	15h

Member 2: Loris Panza (967915):

Topic	**Hours**
Chapter 1 - Introduction	4h
2.1 - Product perspective	8h
2.2, 2.3 - Product function and user characteristics	2.5h
2.4 - Constraints, assumptions and dependencies	1.5h
3.1 - External interface requirements	3h
3.2 - Functional requirements	15h
3.3 - Goals, requirements and assumptions	4h
3.4, 3.5, 3.6, 3.7 - Performance requirements, design constraints, software system attributes and other requirements	1h
Chapter 4 - Formal analysis using alloy	15h

Chapter 6

References

- All the diagrams have been made using “Visual Paradigm 16.2”.
- The alloy code has been executed and developed using Alloy 4.2 (platform independent).