POLITECNICO
MILANO 1863

# RASD

Requirements Analysis and Specification Document

Authors: Gabriele DIGREGORIO
Enrico MASSARO
Vanessa TAMMA

Version: 0.5
Date: 07 December 2020
Professor: Elisabetta Di Nitto

# CONTENTS

1 INTRODUCTION ..... 4
1.1 Purpose ..... 4
1.2 Scope ..... 4
1.2.1 World Phenomena ..... 5
1.2.2 Shared Phenomena ..... 5
1.3 Definitions, Acronyms, Abbreviations ..... 6
1.3.1 Definitions ..... 6
1.3.2 Acronyms ..... 6
1.3.3 Abbreviations ..... 6
1.4 Revision history ..... 6
1.5 Reference Documents ..... 7
1.6 Document Structure ..... 7
2 OVERALL DESCRIPTION ..... 10
2.1 Product perspective ..... 10
2.1.1 Class Diagram ..... 11
2.2.2 State Charts ..... 11
2.2.2 State Charts ..... 11
2.2 Product functions ..... 11
2.3 User characteristics ..... 12
2.3 Assumptions, dependencies and constraints ..... 12
2.3.1 Domain Assumptions ..... 12
3 SPECIFIC REQUIREMENTS ..... 13
3.1 External Interface Requirements ..... 13
3.1.1 User interfaces ..... 13
3.1.2 Hardware interfaces ..... 14
3.1.3 Software interfaces ..... 14
3.1.4 Communication interfaces ..... 14
3.2 Functional Requirements ..... 14
3.2.1 Use cases ..... 15
3.2.2 Mapping ..... 28
3.3 Performance Requirements ..... 33
3.4 Design Constraints ..... 33
3.4.1 Standards Compliance ..... 33
3.4.2 Hardware Limitations ..... 34
3.4.3 Any Other Constraints ..... 34

3.5 Software System Attributes ........ 34
3.5.1 Reliability ........ 34
3.5.2 Availability........ **Errore. Il segnalibro non è definito.**
3.5.3 Security........ 34
3.5.4 Maintainability ........ 34
3.5.5 Portability........ 34
**4 FORMAL ANALYSIS USING ALLOY**........ 34
**5 EFFORT SPENT**........ 43
**6 FORMAL ANALYSIS USING ALLOY**........ **Errore. Il segnalibro non è definito.**

# 1 INTRODUCTION

## 1.1 Purpose

The purpose of the project CLup (Customer Line-up) is to develop a digital system of line up that saves people from having to stand outside of stores for hours, avoids crowds inside the store, and, more in general, allows to regulate the influx of people in the stores.

The idea is to create a digital version of the traditional mechanism of lining up that is easy to use by everyone. In this way, the system would help to deal with the strict rules imposed by the government due to the global pandemic.

The system should give customers the possibility to line up from their home and approach to store only when their number is close to being called. This mechanism should avoid the situation in which the customers wait for their shift in the proximity of the store that is not an acceptable scenario in a lockdown situation.

| | |
|---|---|
| **G.1** | Allows to regulate the influx of people that enter the building. |
| **G.2** | Avoids that customers must line up and wait outside of stores for hours. |
| | |
| **G.4** | Everyone must be able to shop, even people who do not have access to the required technology. |
| **G.5** | Provides smart managing of lining up and booking with a digital system. |

## 1.2 Scope

The software should represent a digital alternative to the situation in which people retrieve a physical number that gives their position in the queue when they want to enter a store.

C-Lup should provide three main features:

- **Lining up**: allows customers to line up from their home and avoids crowds outside the stores. It should include tools to regulate the influx of people into the store and a notification system that alerts people when their number is close to being called. These alerts should consider the time customers need to get to the shop from the place they currently are and should be based on precise estimation of the waiting time. Moreover, C-Lup must provide effective fallback options for people who do not have access to the required technology.
- **Booking**: allows customers to book a visit to the supermarket. Since the time that it takes to visit a supermarket is not uniform, the system should give to user the possibility to specify an estimation of the duration of the visit. Alternatively, it might infer this information analysing the previous visits, if any.

- **Suggestions**: suggests different time slots for visiting the store (also on different days) to deal better with the restriction in the number of people inside the store. Alternatively, the system should propose to the customers other available supermarkets and alerts them in case a new time slot becomes available (e.g. after the deleting of a booking by another customer).

The customer that wants to use the service must be registered. Thanks to this, the system would be able to track the lining up, the booking, and the duration of the previous visits and use this information to manage better the influx of people and estimates with acceptable accuracy the waiting time.

### 1.2.1 World Phenomena

| | |
|---|---|
| **W.1** | Limitation on the number of entrances into buildings. |
| **W.2** | Distance of at least one meter between people. |
| **W.3** | Prohibition of gathering and long lines near the groceries. |
| **W.4** | The required technology is not accessible to everyone. |
| **W.5** | Variable visit time inside the shops. |

### 1.2.2 Shared Phenomena

| | |
|---|---|
| **S.1** | Regulation of the influx of people in the shop. |
| **S.2** | Management of the lines outside of stores. |
| **S.3** | Line up at the store from a smartphone. |
| **S.4** | Monitoring of the entrances by store managers. |
| **S.5** | Estimation of the waiting time. |
| **S.6** | Notification that alerts customers when their shift will begin shortly. |
| **S.7** | Line up for people that do not have access to the required technology. |
| **S.8** | Book a visit to the supermarket. |
| **S.9** | Estimation of visit time inside the shops. |
| **S.10** | Suggestion of different stores of different chains. |
| **S.11** | Suggestion of different time slots to visit the store. |
| **S.12** | Periodic notification of available time slots in a day/time range. |

## 1.3 Definitions, Acronyms, Abbreviations

### 1.3.1 Definitions

| | |
|---|---|
| **Time slot** | Period or day that can be chosen for a booking by the customers. |
| **Store data** | Data about the store like the number of people allowed in each thematic sector, opening and closing times, address, name, and photo. |
| **Reservation** | A word that might indicate either a booking or a lining up in a specific store. |
| **Active reservation** | Lining up or booking that is not yet expired. It means that the reservation has been taken but customers still have to wait for their shift. |
| **Store manager** | Manager, cashier, or employee of a store. |

### 1.3.2 Acronyms

| | |
|---|---|
| **CLup** | Customer Line-up |
| **RASD** | Requirements Analysis and Specification Document |
| **SSL** | Secure Sockets Layer |
| **TLS** | Transport Layer Security |
| **UI** | User interface |

### 1.3.3 Abbreviations

| | |
|---|---|
| **Wn** | World phenomena n-th |
| **Sn** | Shared phenomena n-th |
| **Gn** | Goal n-th |
| **Rn** | Requirement n-th |

## 1.4 Revision history

| DATE | DESCRIPTION |
|---|---|
| **31/10/2020** | First version and goals definition. |

| | |
|---|---|
| **07/11/2020** | World and shared phenomena definition. |
| **25/11/2020** | Mock-ups, textual descriptions, definitions, acronyms, and abbreviations. |
| **29/11/2020** | Mapping between goals, requirements, and domain assumptions. General improvement. |
| **02/12/2020** | Adding Product Functions and Product Perspective descriptions. |
| **07/12/2020** | Adding sequence diagrams and their descriptions |
| **13/12/2020** | Performance Requirements, Design Constraints, and Software System Attributes |
| **16/12/2020** | Use cases mapping |
| **18/12/2020** | Alloy integration |
| | |

## 1.5 Reference Documents

- *Requirement Engineering and Design Project: goal, schedule, and rules*
- *I&T assignment goal, schedule, and rules*
- Slides of the course *Software Engineering 2*

## 1.6 Document Structure

The document is composed of six chapters. They are the following:

- **Chapter 1**: provides an introduction to the purposes and the whole scenario of the software. First, it includes the general description of the system and the goals. Then, there is a sufficiently detailed specification of the main features that the system should provide and the analysis of the world and shared phenomena. Lastly, it includes the list of abbreviations, acronyms, and definitions used in the document, the revision history, and the reference documents.
- **Chapter 2**: specifies the most relevant function of the system to be and provides its representation as class diagram and state charts. Lastly, in this chapter there is the clarification of the different users of the system and the list of domain assumptions of the scenario.
- **Chapter 3**: shows a high level description of the interfaces of the system and specifies the functional requirements. Here are present also the use cases diagrams and the mapping between requirements, domain assumptions, and use cases.

- **Chapter 4**: includes the metamodel generates with Alloy and the code used for this scope. Also, there is present a description of the main objectives of this analysis.
- **Chapter 5**: shows the amount of time that each member has spent to produce the document.
- **Chapter 6**: specifies the reference documents and online resources used during the production of this document.

| | |
|---|---|
| **R.1** | The system generates a single QR code to enter and exit the store for each booking or lining up. |
| **R.2** | The system allows to get a reservation for the supermarket. |
| **R.3** | The system provides customers a precise estimation of the waiting time. |
| **R.4** | The system uses the customers' current position to estimate the waiting time. |
| **R.5** | The system alerts the customers before their shift according to the geolocation information. |
| **R.6** | The system allows people (who do not have access to the required technology) digitally line up directly when they are at the store. |
| **R.7** | The system suggests alternative time slots for visiting the store when the desired one is not available. |
| **R.8** | The system suggests alternative stores when the desired one is not available. |
| **R.9** | The system allows customers to insert the approximate expected duration of the visit. |
| **R.10** | The system infers customers' expected duration of the visit based on an analysis of the previous visits. |
| **R.11** | The system allows store managers to get a lining up only for their own store. |
| **R.12** | The system provides periodic notifications of available time slots in a day/time range. |
| **R.13** | The system shows the list of shops. |
| **R.14** | The system shows the available time slots for each grocery. |
| **R.15** | The system provides a QR code printing service. |
| **R.16** | The system requires a sign up/login. |
| **R.17** | The system shows active bookings. |
| **R.18** | The system shows the history of bookings. |
| **R.19** | The system allows customers to delete a booking. |

| | |
|---|---|
| **R.20** | The system allows customers only one lining up at a time for each shop. |
| **R.21** | The system uses information about the customer that exit the store to infer better the waiting time. |
| **R.22** | The system allows the store manager to scan the QR codes. |
| | |
| **R.24** | The system allows customers only one booking at a time for each shop. |

# 2 OVERALL DESCRIPTION

## 2.1 Product perspective

C-Lup helps to manage the influx of people both inside and outside the stores avoiding crowding. It represents the digital counterpart of the traditional mechanism of lining up that is not suitable in a lockdown situation.

In the following, a summary of the problem is presented.

- The customers who want to avoid standing outside of stores for hours requires a digital lining up using the C-Lup platform. They select the option for lining up and specify the store and the thematic sectors that they want to visit. Furthermore, they indicate the expected duration of the visit if it is not automatically inferred from the previous visits by the system. Then, the system provides an estimation of the waiting time and shows this and the other active reservations in a specific section of the software. According to the time that the customers need to get to the shop from the place they currently are and with the waiting estimation, the system sends a notification when the customers' number is near to be called. Lastly, the customers can delete a lining up at any time and they can have only one active line up at a time for each store.
- The customers can also book a visit to the supermarket. As seen for the lining up, even for the booking they specify the store and the thematic sectors that they want to visit and the expected duration of the visit if it is not automatically inferred. Moreover, in this case, they need to indicate the time slots when they want to visit the store. The system shows this and the other active reservations in a specific section of the software. Lastly, the customers can delete a booking at any time, and they can have only one active booking at a time for each store.
- If the desired shop is not available in the selected time slot for any reason, the system gives the customers some suggestions about different time slots or different stores. Moreover, the system can alert customers when a new time slot becomes available.
- For each reservation, C-Lup generates a QR code that the customers must scan at the entrance and exit of the shop. The system gives the possibility to print the QR code too.
- People who do not have access to the required technology are also allowed to line up at the store. The store managers help them and give them a printout of the QR code and the estimated waiting time.
- The store managers own a special version of C-Lup that has no limits on the number of reservations. This allows to manage the lining up also for people who do not have access to the required technology. They can also scan a QR code to allow customers to enter and exit from the store.
- C-Lup requires registration to access its functions. This allows to track the duration of the previous visits and use this information to better manage the influx of people and estimates with acceptable accuracy the waiting time.

### 2.1.1 Class Diagram

The UML class diagram below represents a conceptual, high-level model of the software to be. Givens its nature, it may model objects that will not be represented in the system that will be developed. Moreover, at this level, it should not include any references to methods and other low-level details that will be detailed during the design phase.

### 2.2.2 State Charts

### 2.2.2 State Charts

## 2.2 Product functions

As described above in this document, C-Lup offers several functions that help to manage the influx of people both inside and outside the stores. In the following, a list of the basic and advanced functions is presented.

- **Line up**. The system allows to:
  - specify the store and the thematic sectors that customers want to visit;
  - indicate the expected duration of the visit;
  - delete a lining up at any time;
  - see the list of the active reservations;
  - see the history of the reservations;
  - send a notification according to the time that the customers need to get to the shop from the place they currently are and with the waiting estimation.
- **Booking**. The system allows to:
  - specify the store and the thematic sectors that customers want to visit;
  - indicate the expected duration of the visit;
  - delete a booking at any time;
  - see the list of the active reservations;
  - see the history of the reservations;
  - suggestions about different time slots or different stores if the desired shop is not available in the selected time slot for any reason. The system can also alert customers when a new time slot becomes available;
  - indicate the time slots when they want to visit the store.
- **Fallback option**. People who do not have access to the required technology can line up at the store too.

Moreover, for each reservation, the system generates a QR code that can be scanned by the store manager at the entrance and exit of the shop. This code can be also printed.

C-Lup requires a sign up to access its functions. This allows to track the reservations from a specific customer and the duration of the previous visits.

## 2.3 User characteristics

It is possible to distinguish two different types of actors who use the system:

1. **Customer**: someone who wishes to have access to the supermarket. She/He wants to get smart reservations, manage them, see active ones, and explore shop visit history. He/She cannot have more than one active reservation for the same store at a time.
2. **Store Manager**: someone who helps people who do not have access to the required technologies to get a reservation. She/he might also help in the QR code scanning operation. He/She can have an unrestricted number of active reservations at a time.

## 2.4 Assumptions, dependencies and constraints

### 2.4.1 Domain Assumptions

| | |
|---|---|
| **D.1** | The customers wait until their number is close to being called to approach the store. |
| **D.2** | The time that it takes to visit the supermarket is not uniform. |
| **D.3** | The approximated expected duration of the visit indicated by the customer is reliable. |
| | |
| **D.5** | Customers can get into the store if and only if they scan a valid QR code. |
| **D.6** | All customers that want to access the store use the digital system of lining up. |
| **D.8** | Customers exit from the store only after they had scanned a valid QR code. |
| **D.9** | Store data are provided by the store owner and are already present in the DB. |
| **D.10** | The store manager with the digital system helps people who do not have access to the required technology to line up directly when they are at the store. |
| **D.11** | The store manager prints the QR code for people who do not have access to the required technology. |
| **D.12** | The system is applied to medium and large shops[1]. |
| **D.13** | One and only one customer can enter with one QR code. |

[1] Shops that have a dimension greater than 400 square meters (warehouses, laboratories and offices are not included).

# 3 SPECIFIC REQUIREMENTS

## 3.1 External Interface Requirements

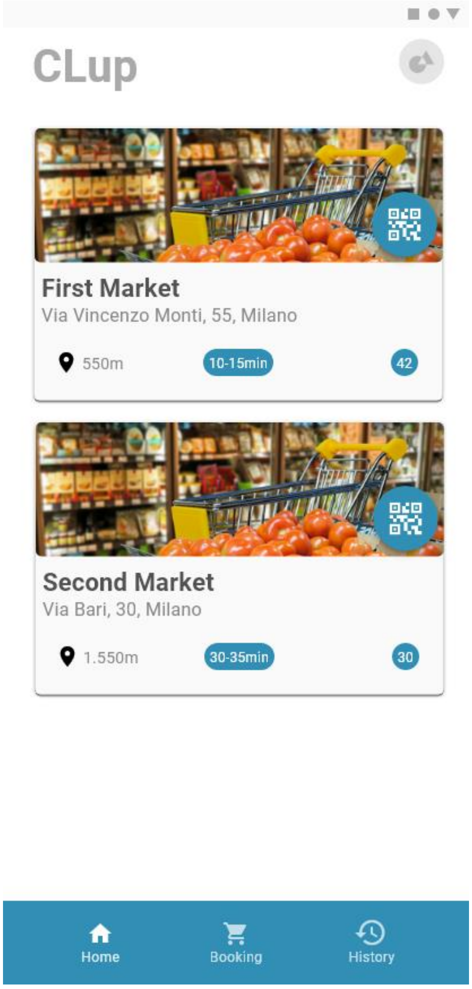
### 3.1.1 User interfaces

The system allows store managers and customers to manage the reservations and create new ones at any time. It also uses customers' location information to alert them when their number is close to being called. Lastly, the history of the reservations and a summary of the active ones are other functions that should be provided by the system.

Given these and the other purposes described above in this document, a smartphone is clearly a suitable device to use all necessary functionalities. In general, it satisfies all the hardware requirements that will be described below. Hence, a mobile application might be adequate to allow customers and store managers to access the system.

Let us assume that the User Interface (UI) is divided into several sections dedicated to the different functionalities. The following mock-up shows an idea of two of these sections.

The first one describes the Home Page of the application with the list of the active reservations. On the other hand, the second one shows the Booking Page with an example of possible choices and information that should be provided to the customers and store managers.



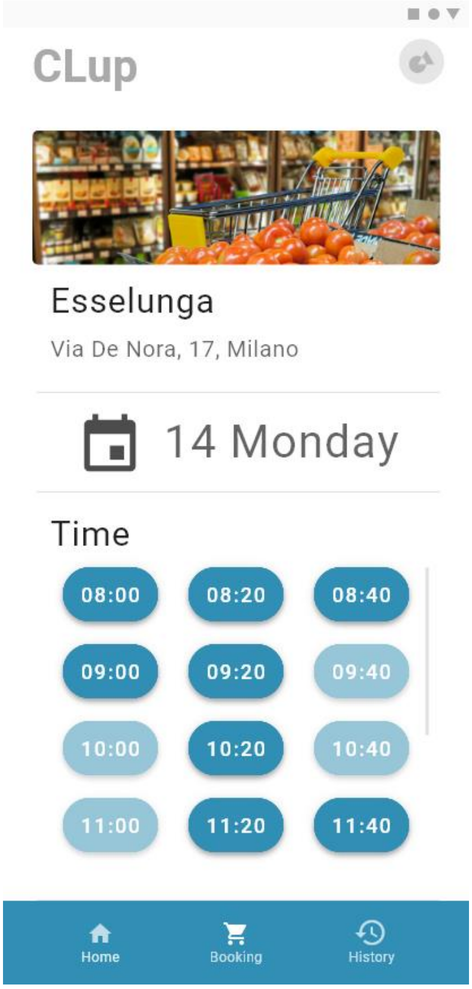


*Figure 1 - Home Page mock-up*

*Figure 2 – Booking Page mock-up*

### 3.1.2 Hardware interfaces

To use the system, both Customers and Store Managers must use a mobile device. Due to the communication capabilities needed, a smartphone might be a good example of a suitable device.

### 3.1.3 Software interfaces

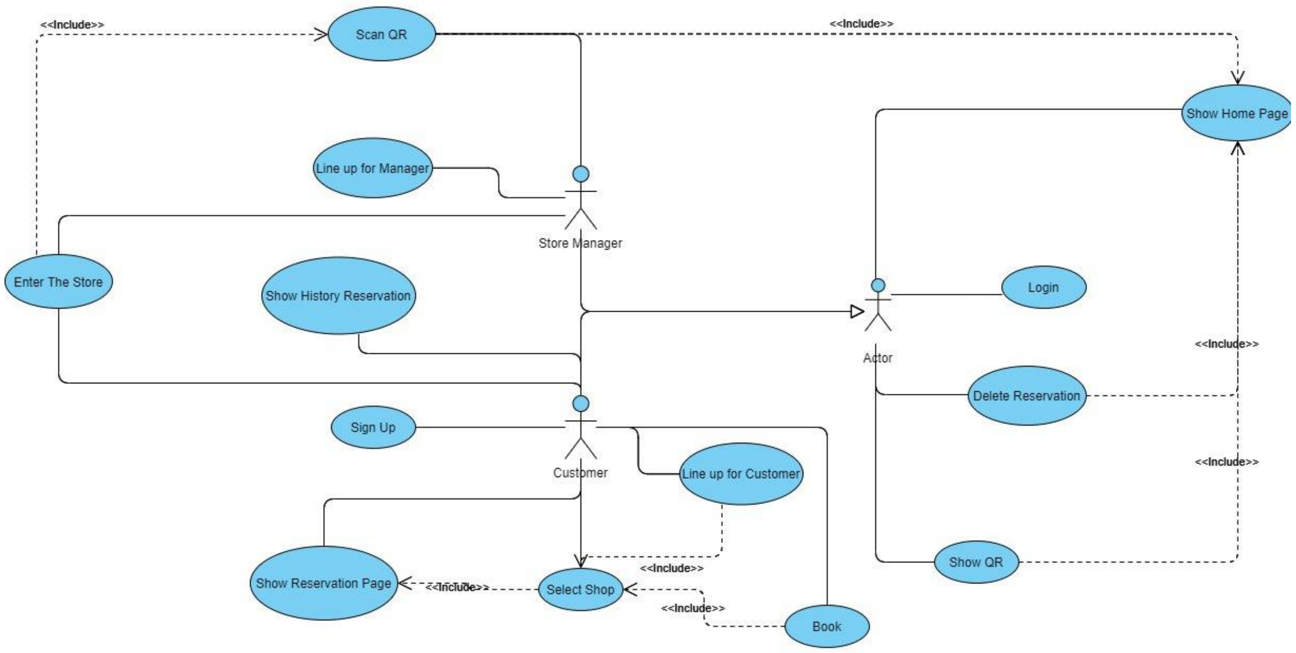
QR Service:

- Generate
- Scan

### 3.1.4 Communication interfaces

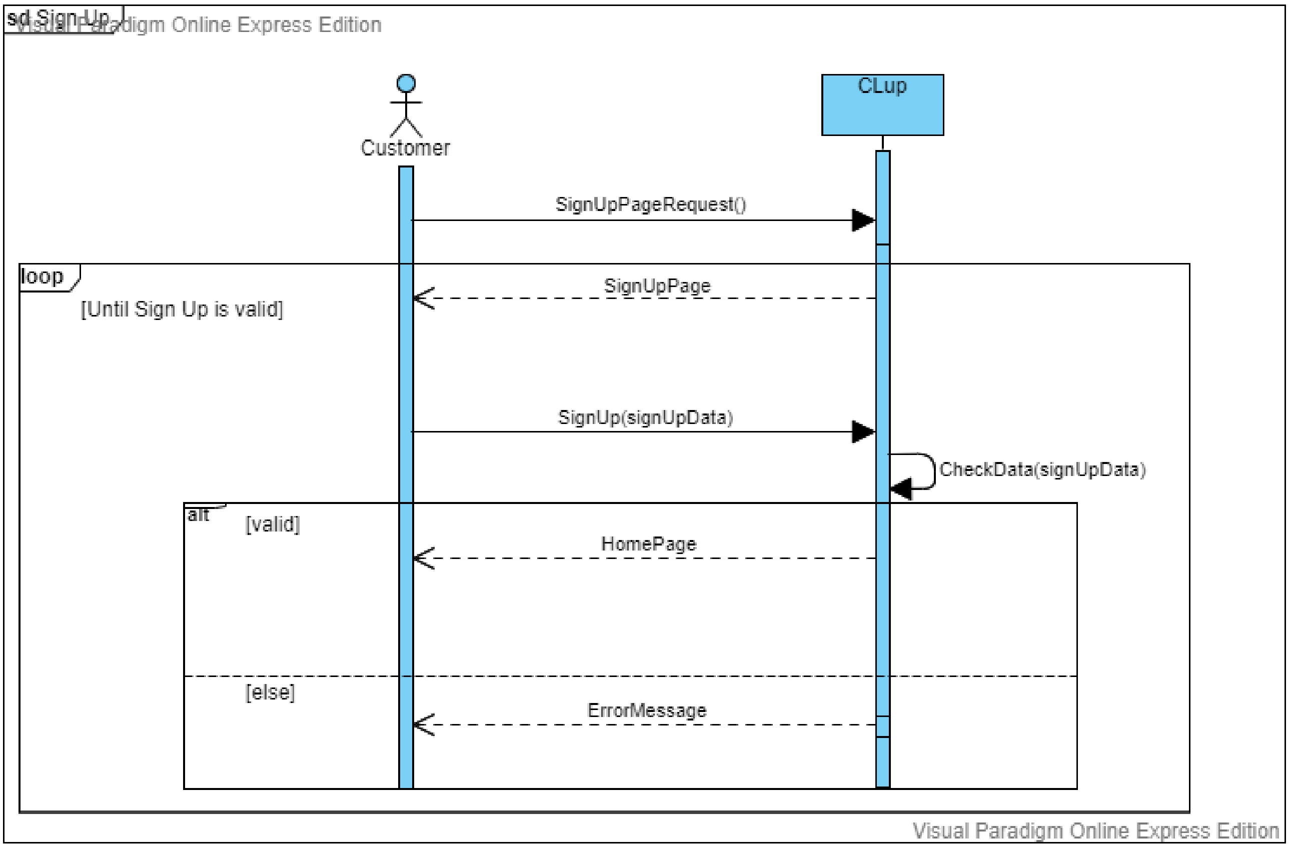
The system requires a stable internet connection to work properly. This connection is used to exchange data between the mobile devices and a central database about the status of the reservation, the list of available stores, and the available time slots.

## 3.2 Functional Requirements



### 3.2.1 Use cases

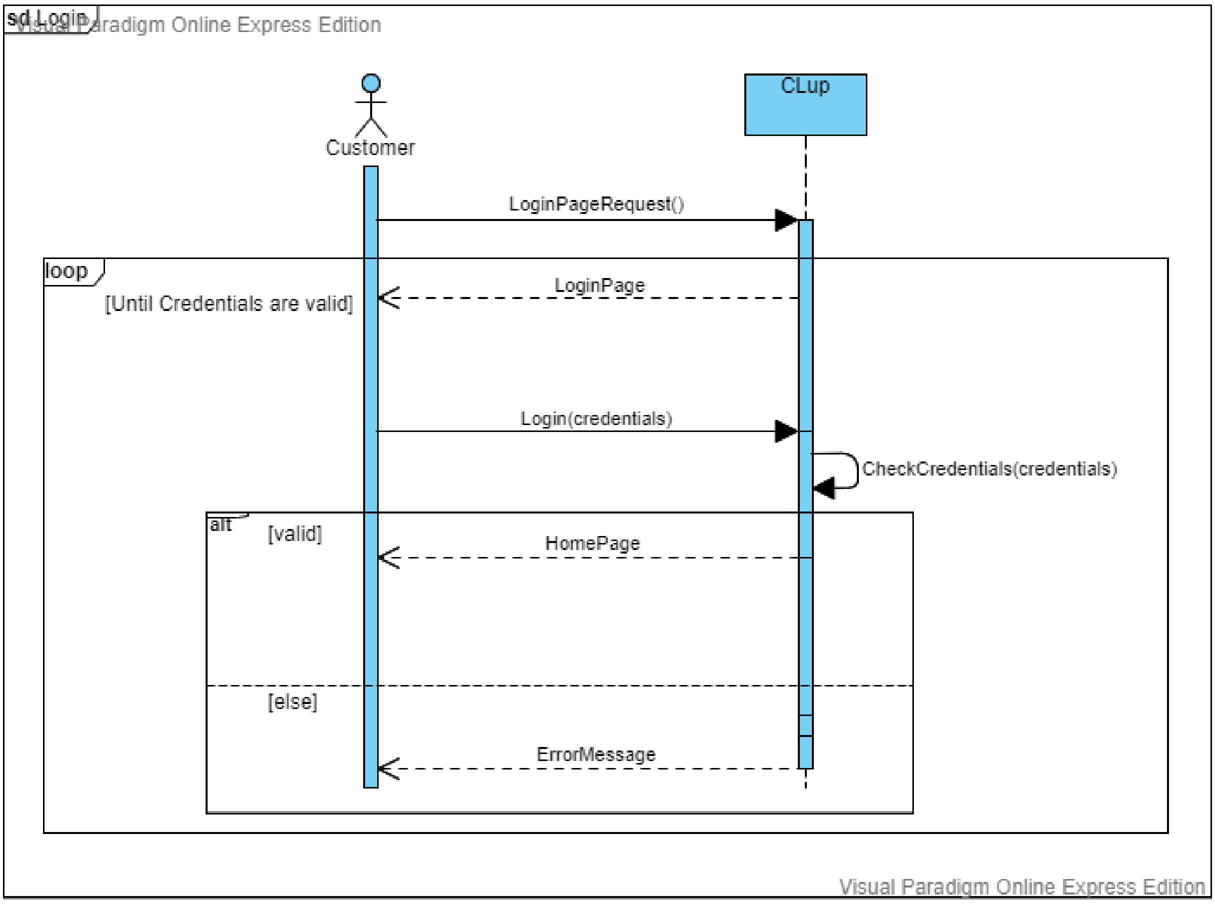
1. Sign Up

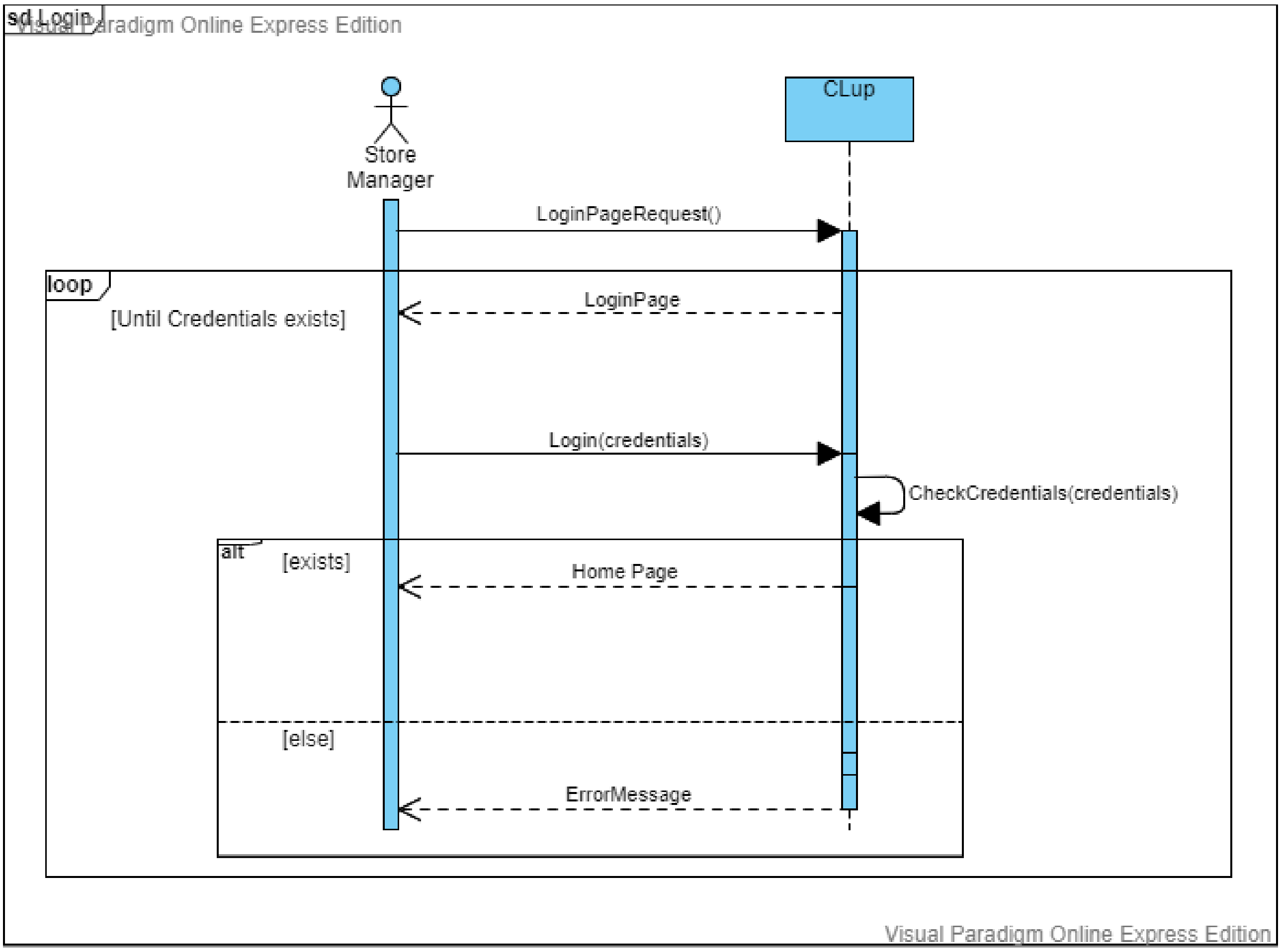


| | |
|---|---|
| **Name** | Sign Up |
| **Actors** | Customer |
| **Entry Condition** | The actor is not already registered at the system |
| **Event Flow** | 1. The actor requires the Sign Up Page<br>2. The system shows the Sign Up Page at the actor<br>3. The actor fills all the mandatory fields<br>4. The system checks the information<br>5. The system saves the information<br>6. The system shows the Home Page at the actor |
| **Exit Condition** | The actor is successfully registered to the system and the home page is shown |
| **Exception** | 1. Any mandatory fields are not filled<br>2. Password and Confirm Password fields are different<br>3. The actor is already present into the system<br>4. Loss of internet connection |

| | |
|---|---|
| | 5. The actor cancels the operation before confirming<br>At this level, we assume that the Sign Up process requires some basic information such as name, email, and password. The password must be inserted twice in two different fields to prevent typing mistakes. |

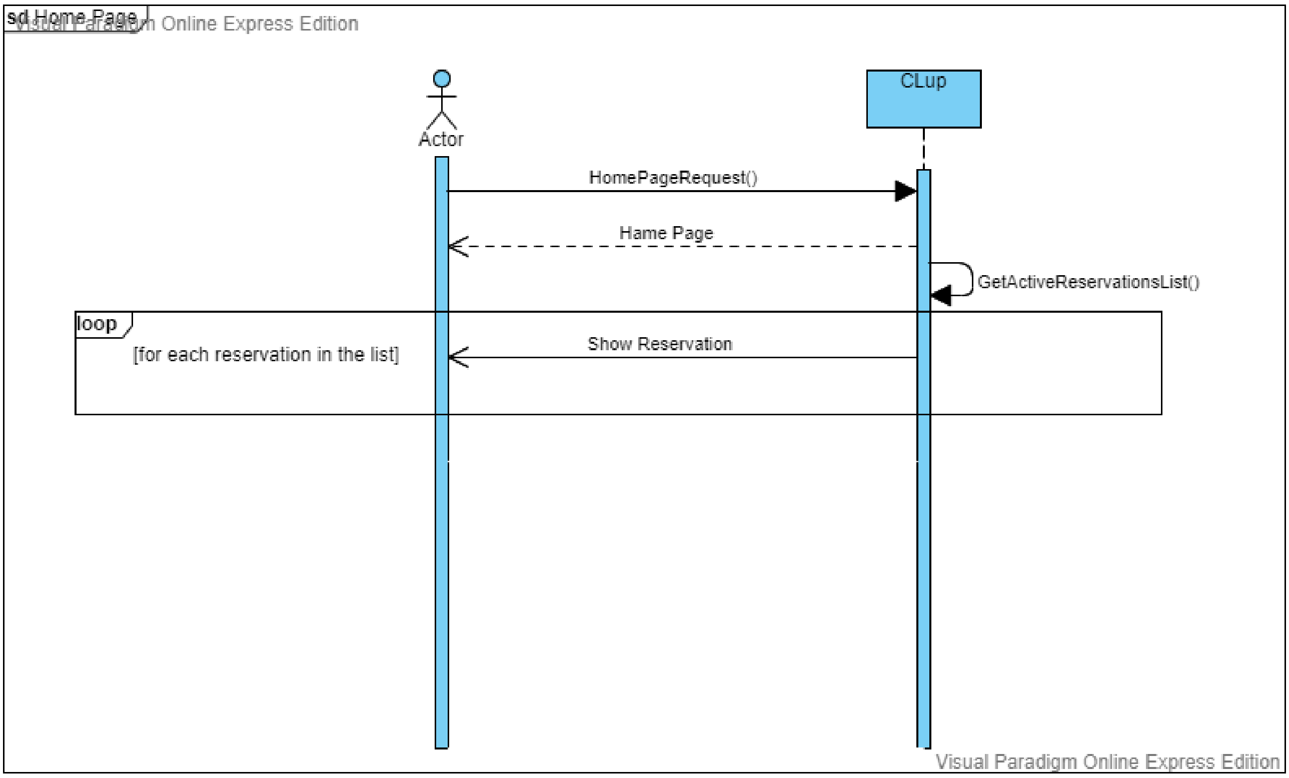
## 2. Login


sd Login
Visual Paradigm Online Express Edition
Customer
CLup
LoginPageRequest()
loop
[Until Credentials are valid]
LoginPage
Login(credentials)
CheckCredentials(credentials)
alt
[valid]
HomePage
[else]
ErrorMessage
Visual Paradigm Online Express Edition


sd Login
Visual Paradigm Online Express Edition
Store Manager
CLup
LoginPageRequest()
loop
[Until Credentials exists]
LoginPage
Login(credentials)
CheckCredentials(credentials)
alt
[exists]
Home Page
[else]
ErrorMessage
Visual Paradigm Online Express Edition

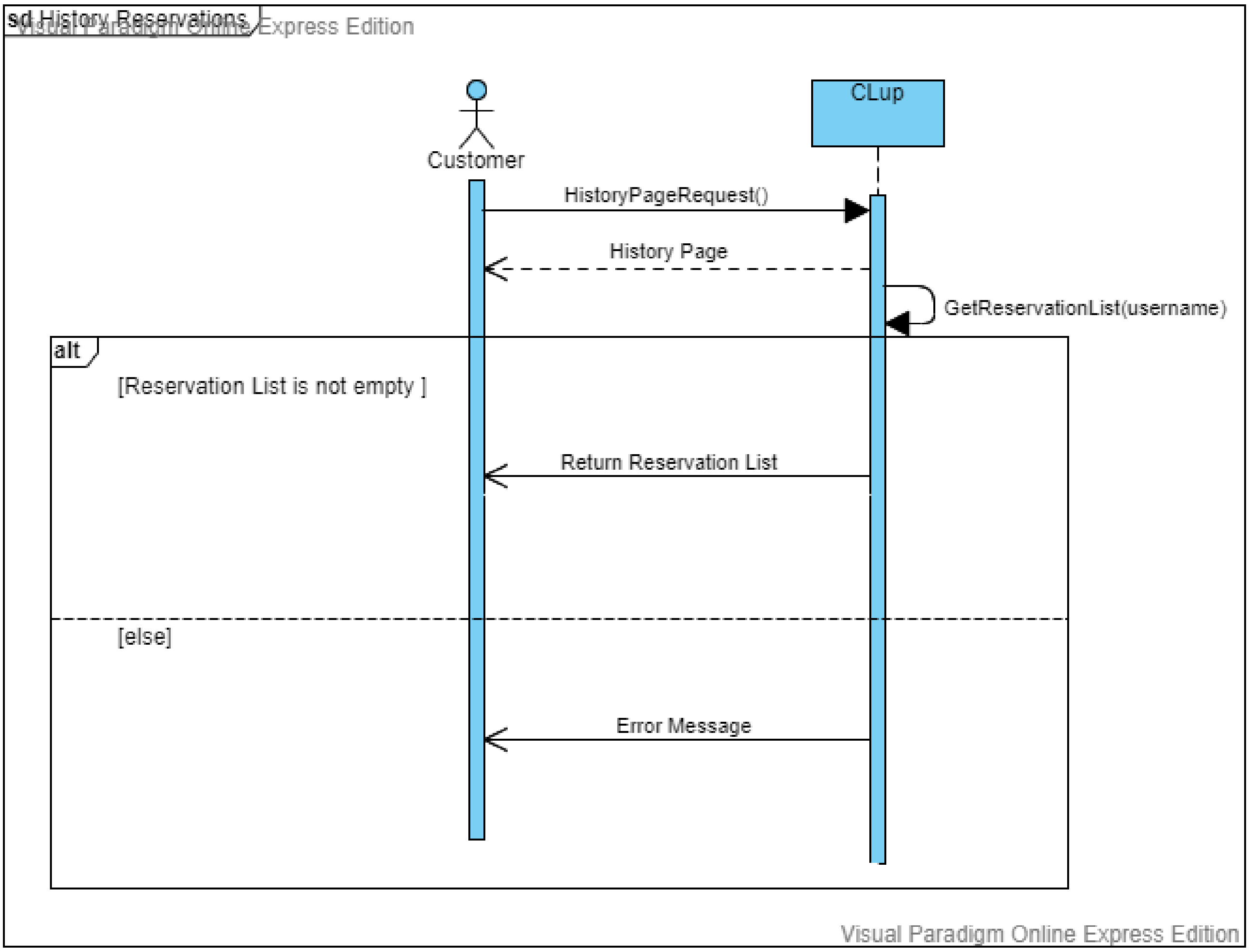
| | |
|---|---|
| **Name** | Login |
| **Actors** | Customer, Store Manager |
| **Entry Condition** | The actor is already registered at the system |
| **Event Flow** | 1. The actor requires the Login Page<br>2. The system shows the Login Page at the actor<br>3. The actor inserts the credentials<br>4. The system checks the information<br>5. The system shows the Home Page at the actor |
| **Exit Condition** | The actor has successfully logged into the system and the home page is displayed |
| **Exception** | 1. The username is not valid<br>2. The password is not valid<br>3. Loss of internet connection<br>4. The actor cancels the operation before confirming |

### 3. Home Page


sd Home Page
Visual Paradigm Online Express Edition
Actor
CLup
HomePageRequest()
Hame Page
GetActiveReservationsList()
loop
[for each reservation in the list]
Show Reservation
Visual Paradigm Online Express Edition

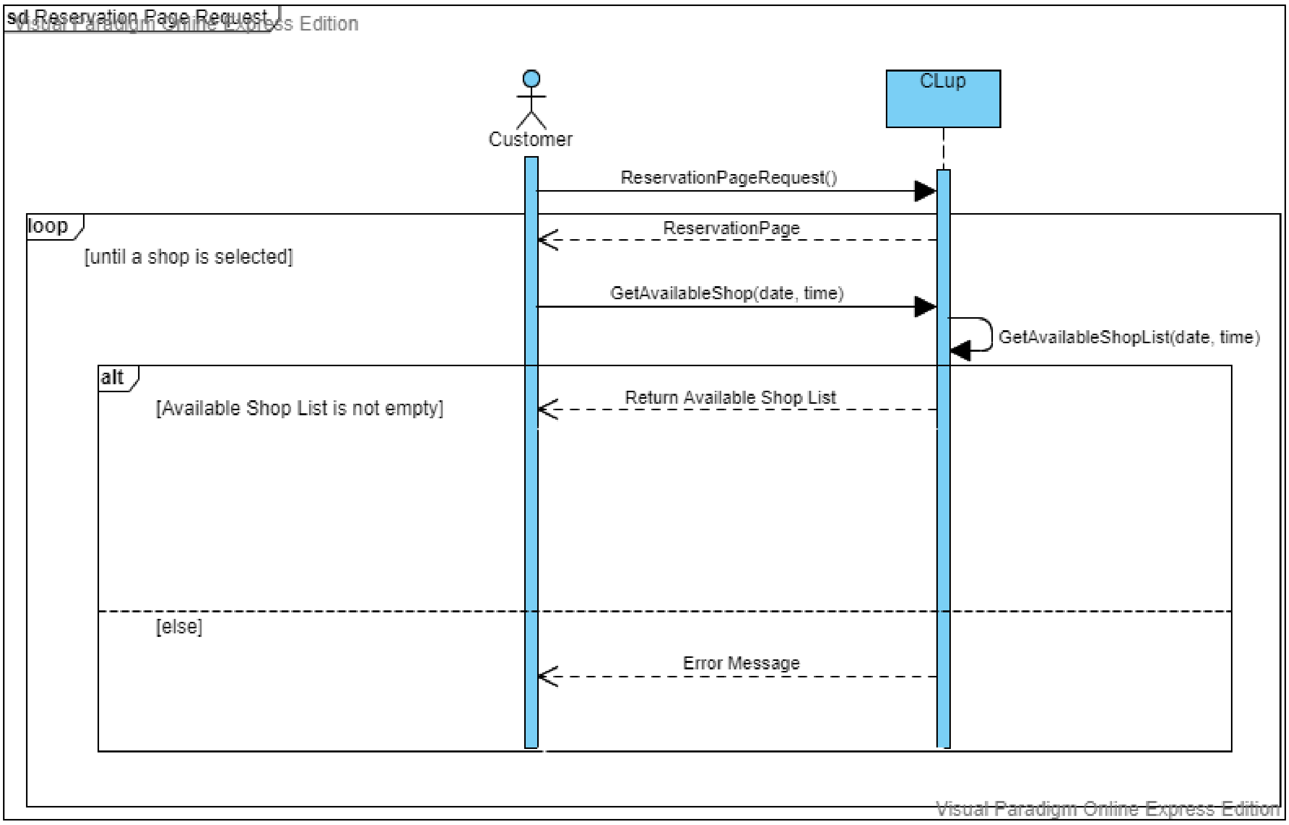
| | |
|---|---|
| **Name** | Home Page |
| **Actors** | Customer, Store Manager |
| **Entry Condition** | The actor is already logged into the system |
| **Event Flow** | 1. The actor requires the Home Page<br>2. The system shows the Home Page at the actor<br>3. The system retrieves the list of active reservations<br>4. The system displays the list of active reservations in the Home Page |
| **Exit Condition** | The Home Page and the list of active reservations are displayed |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation<br>3. The list of active reservations is empty |

4. History Reservations


sd History Reservations
Visual Paradigm Online Express Edition
Customer
CLup
HistoryPageRequest()
History Page
GetReservationList(username)
alt
[Reservation List is not empty ]
Return Reservation List
[else]
Error Message
Visual Paradigm Online Express Edition

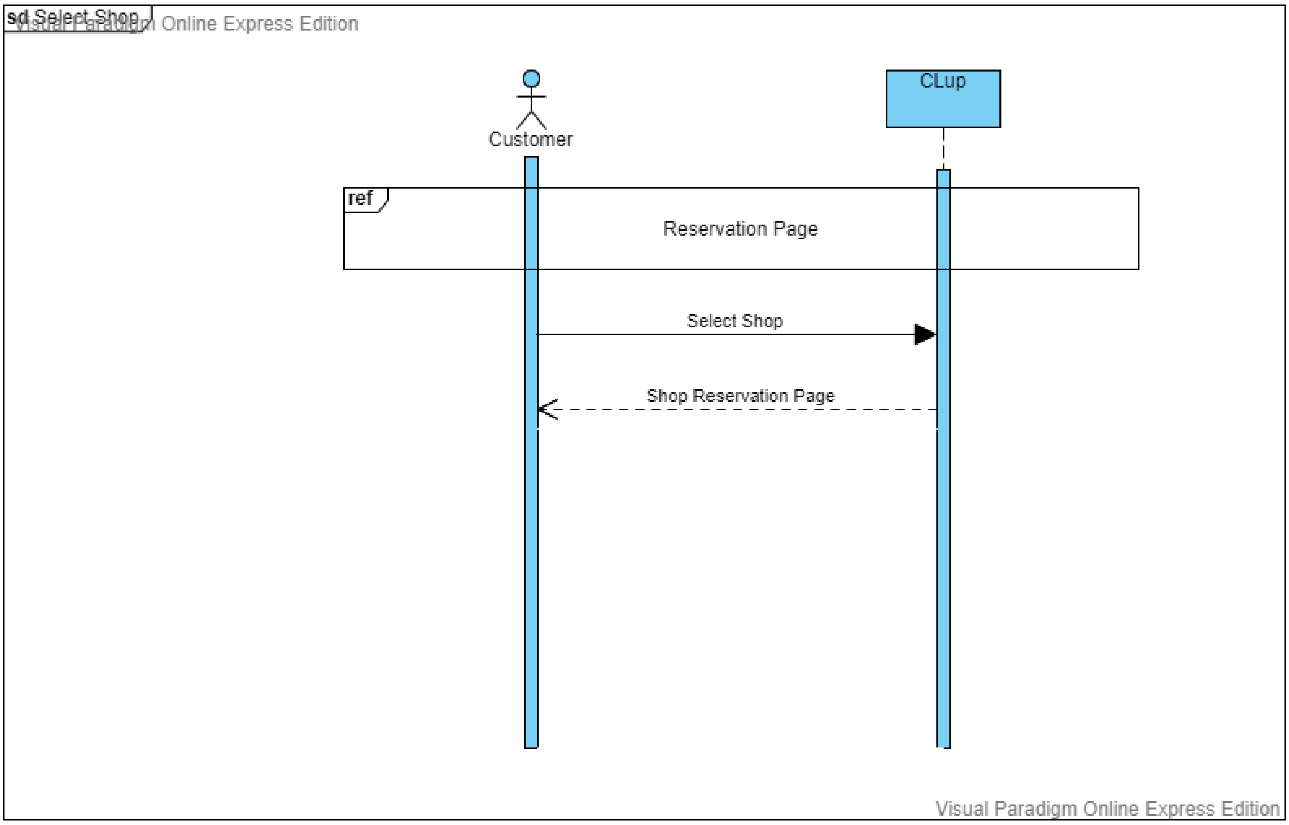
| | |
|---|---|
| **Name** | History Reservation |
| **Actors** | Customer |
| **Entry Condition** | The actor is already logged into the system |
| **Event Flow** | 1. The actor requires the History Page<br>2. The system shows the History Page at the actor<br>3. The system retrieves the list of reservations<br>4. The system displays the list of reservations in the History Page |
| **Exit Condition** | The History Page and the list of reservations are displayed |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation<br>3. The reservation list is empty |

5. Reservation Page


sd Reservation Page Request
Visual Paradigm Online Express Edition
Customer
CLup
ReservationPageRequest()
loop
[until a shop is selected]
ReservationPage
GetAvailableShop(date, time)
GetAvailableShopList(date, time)
alt
[Available Shop List is not empty]
Return Available Shop List
[else]
Error Message
Visual Paradigm Online Express Edition

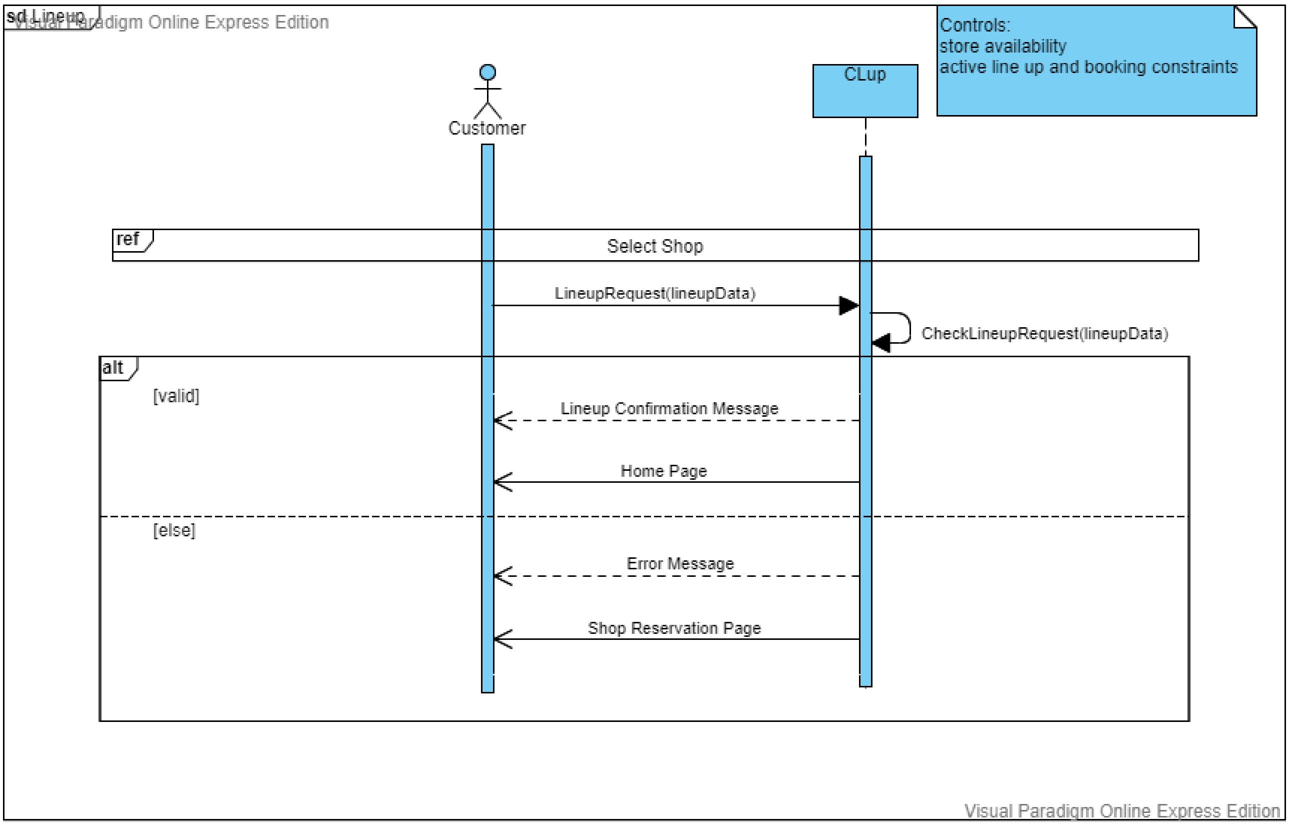
| | |
|---|---|
| **Name** | Reservation Page |
| **Actors** | Customer |
| **Entry Condition** | The actor is already logged into the system |
| **Event Flow** | 1. The actor requires the Reservation Page<br>2. The system shows the Reservation Page at the actor<br>3. The actor requires the list of available shops<br>4. The system retrieves the list of available shops<br>5. The system displays the list of available shops in the Reservation Page |
| **Exit Condition** | The Reservation Page and the list of reservations are displayed |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation<br>3. The list of available shops is empty |

6. Select Shop



| | |
|---|---|
| **Name** | Select Shop |
| **Actors** | Customer |
| **Entry Condition** | The actor is already logged into the system and the system already shows the Reservation Page |
| **Event Flow** | 1. The actor selects the desired shop<br>2. The system shows the Reservation Page |
| **Exit Condition** | The Reservation Page is shown, and the shop is correctly selected |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation |

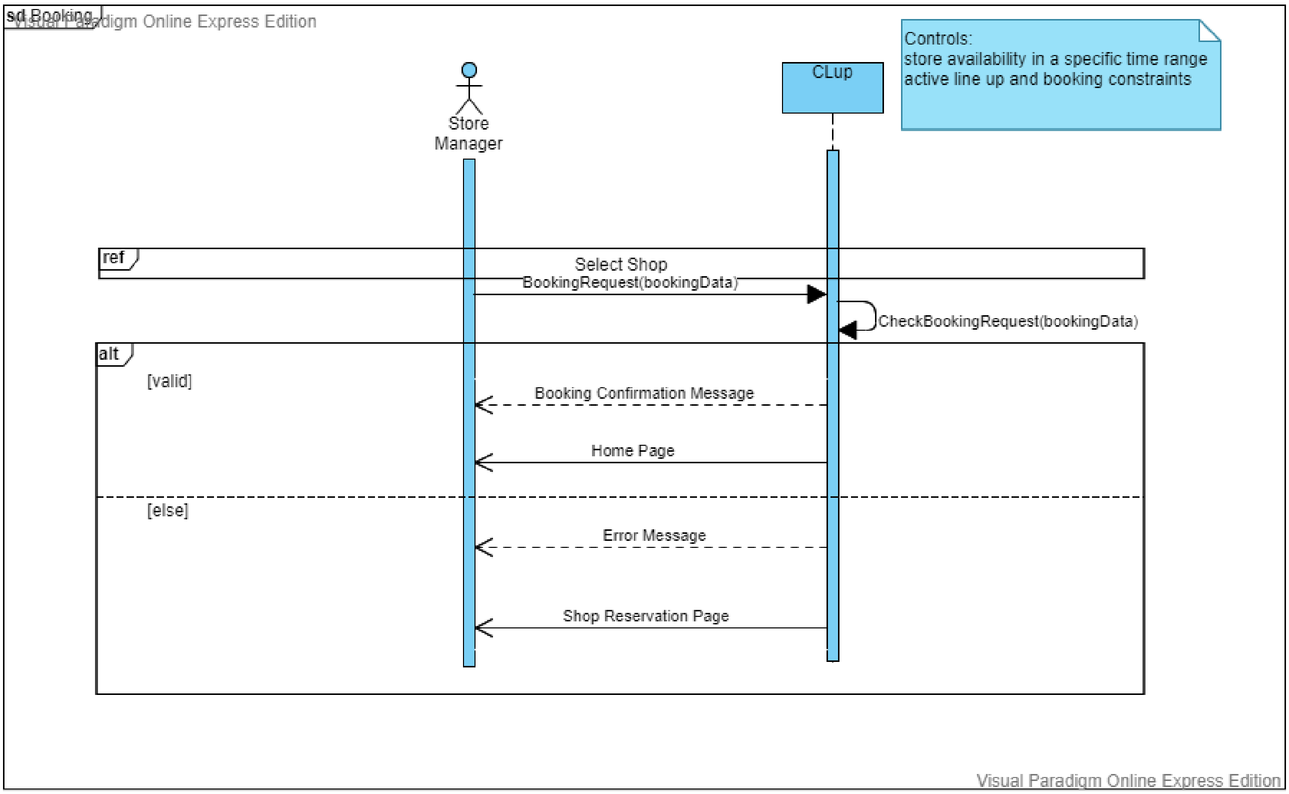
7. Line Up


sd Lineup
Visual Paradigm Online Express Edition
Customer
CLup
Controls:
store availability
active line up and booking constraints
ref
Select Shop
LineupRequest(lineupData)
CheckLineupRequest(lineupData)
alt
[valid]
Lineup Confirmation Message
Home Page
[else]
Error Message
Shop Reservation Page
Visual Paradigm Online Express Edition

| | |
|---|---|
| **Name** | Line Up |
| **Actors** | Customer |
| **Entry Condition** | The actor is already logged into the system and has already selected the shop |
| **Event Flow** | 1. The actor requires a Line Up<br>2. The system checks the request<br>3. The system shows a confirmation message<br>4. The system displays the Home Page |
| **Exit Condition** | The Home Page and the confirmation message are displayed |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation<br>3. The store is not available in a specific time range<br>4. The customer has already another active lining up<br>If an error occurs, the system shows the Reservation Page again to allow a new attempt. |

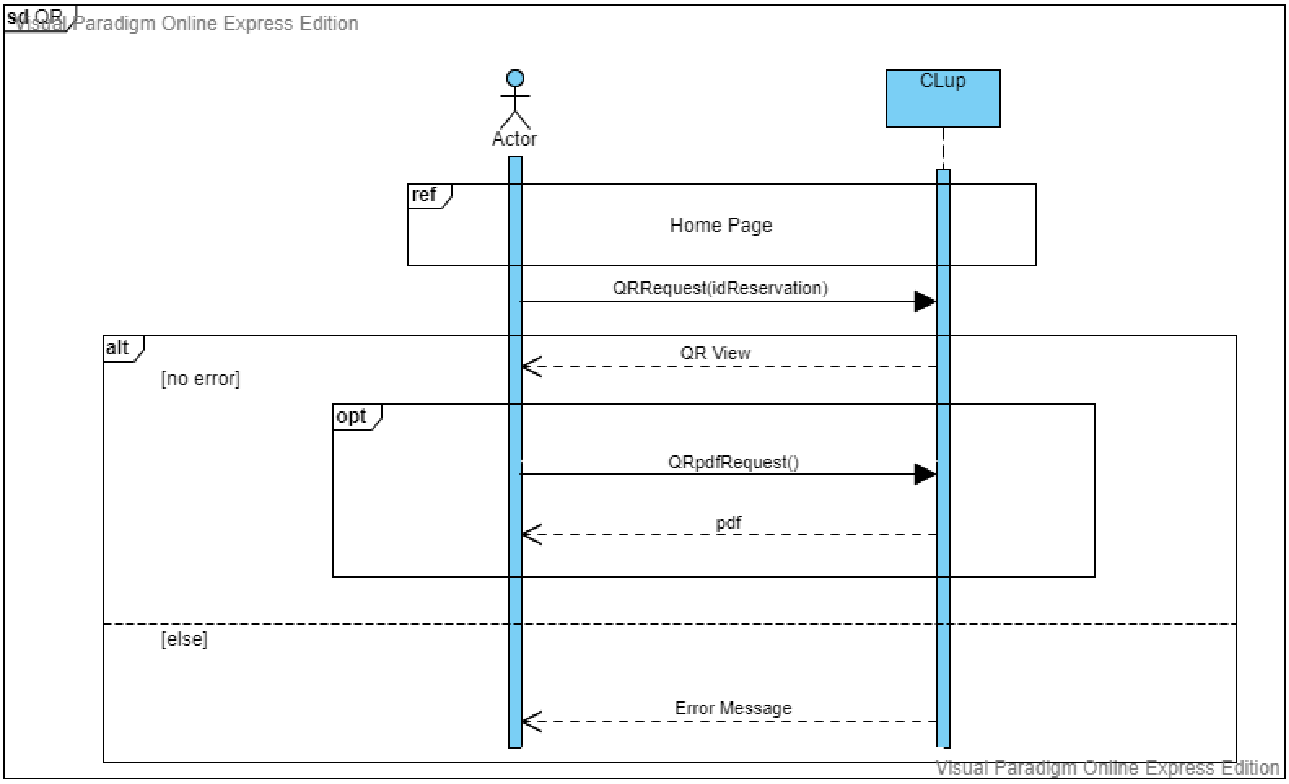
### 8. Manager Line Up

### 9. Booking


sd Booking
Visual Paradigm Online Express Edition
Store Manager
CLup
Controls:
store availability in a specific time range
active line up and booking constraints
ref
Select Shop
BookingRequest(bookingData)
CheckBookingRequest(bookingData)
alt
[valid]
Booking Confirmation Message
Home Page
[else]
Error Message
Shop Reservation Page
Visual Paradigm Online Express Edition

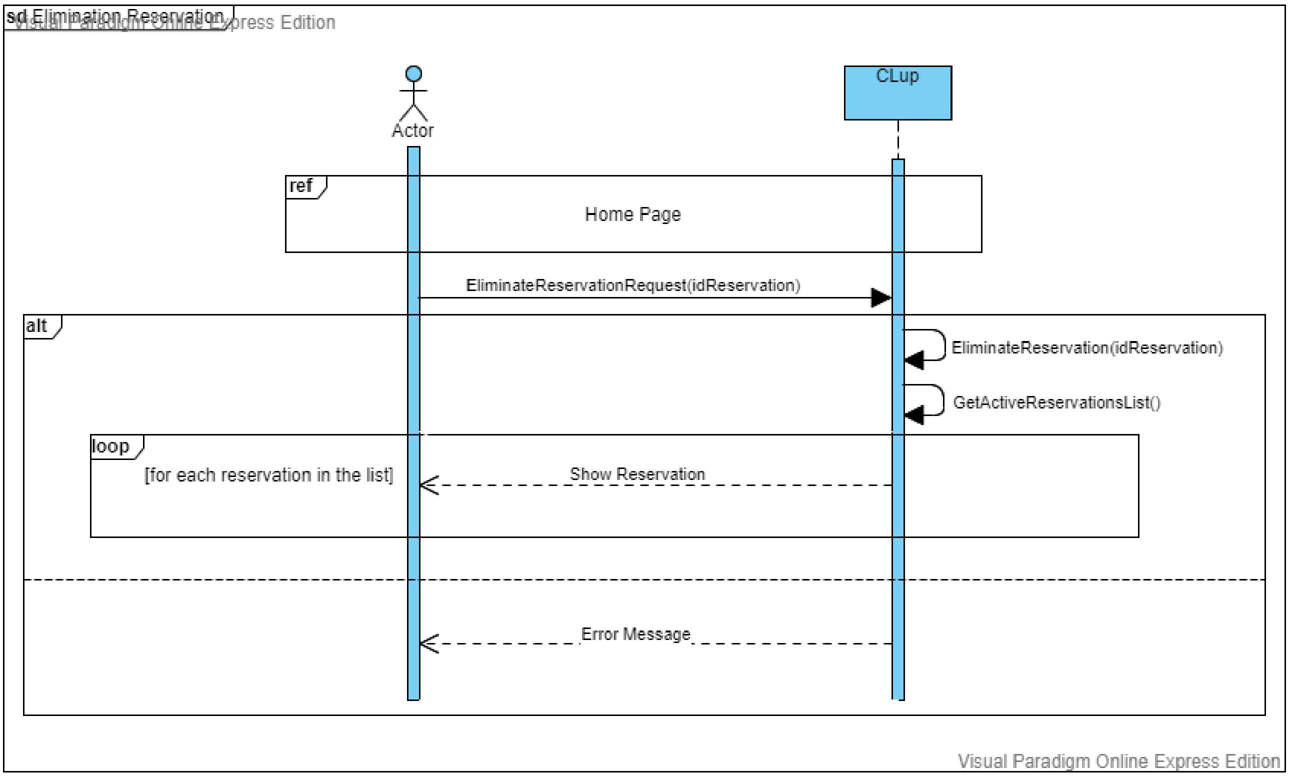
| | |
|---|---|
| **Name** | Booking |
| **Actors** | Customer |
| **Entry Condition** | The actor is already logged into the system and has already selected the shop |
| **Event Flow** | 1. The actor requires a Booking<br>2. The system checks the request<br>3. The system shows a confirmation message<br>4. The system displays the Home Page |
| **Exit Condition** | The Home Page and the confirmation message are displayed |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation<br>3. The store is not available in a specific time range<br>4. The customer has already another active booking<br>If an error occurs, the system shows the Reservation Page again to allow a new attempt. |

## 10. QR View



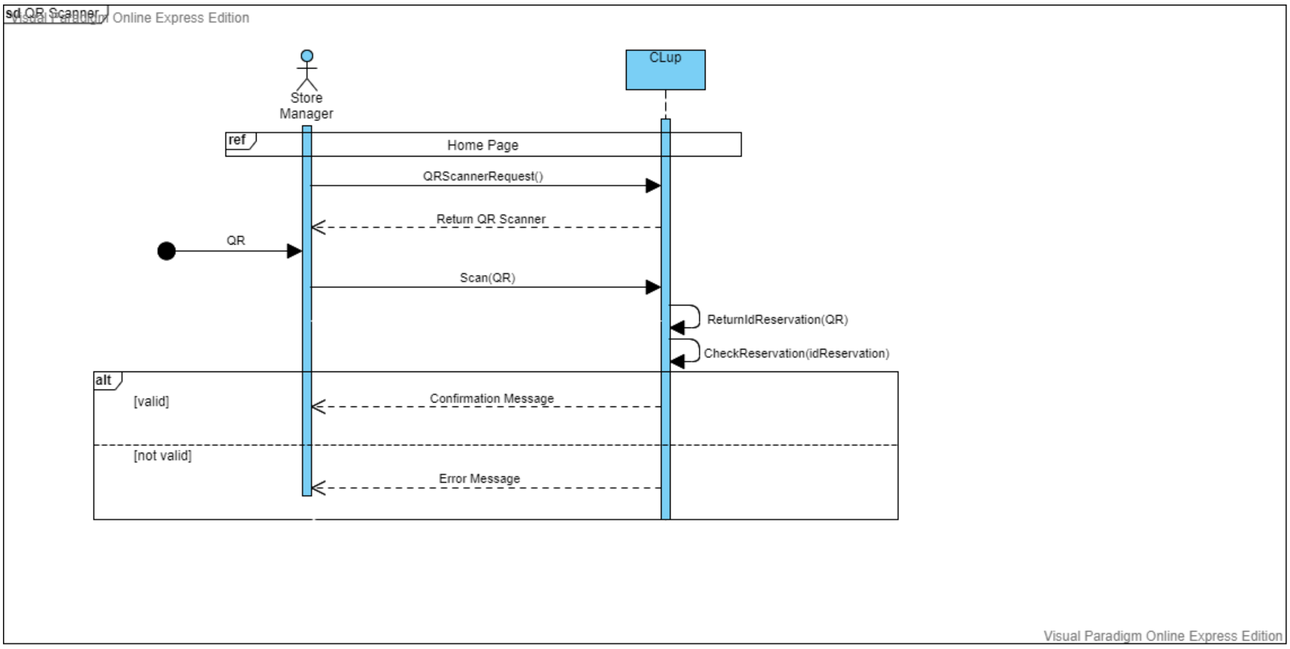
| | |
|---|---|
| **Name** | QR View |
| **Actors** | Customer, Store Manager |
| **Entry Condition** | The actor is already logged into the system and the system already shows the Home Page |
| **Event Flow** | 1. The actor requires the visualization of the QR code associated to a reservation<br>2. The system shows the QR code<br>3. The actor requires a PDF version of the QR code<br>4. The system returns the PDF version of the QR code |
| **Exit Condition** | The QR code is displayed |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation |

## 11. Delete Reservation


sd Elimination Reservation
Visual Paradigm Online Express Edition
Actor
CLup
ref
Home Page
EliminateReservationRequest(idReservation)
alt
EliminateReservation(idReservation)
GetActiveReservationsList()
loop
[for each reservation in the list]
Show Reservation
Error Message
Visual Paradigm Online Express Edition

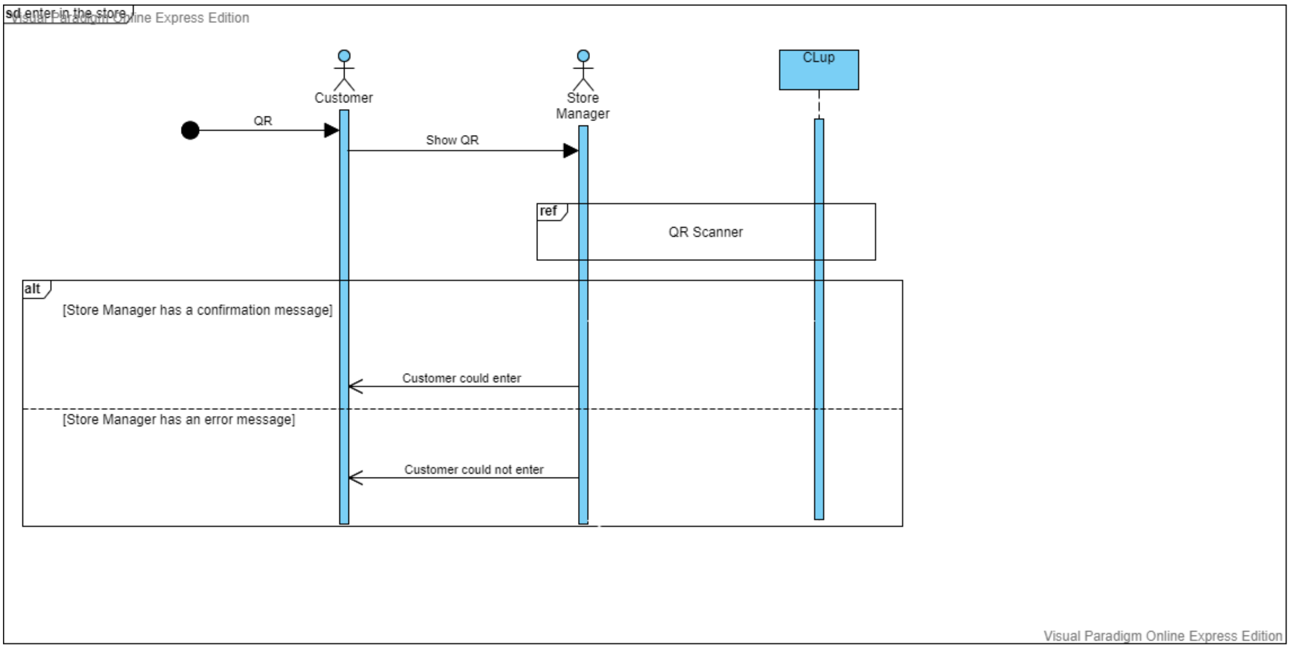
| | |
|---|---|
| **Name** | Delete Reservation |
| **Actors** | Customer, Store Manager |
| **Entry Condition** | The actor is already logged into the system and the system already shows the Home Page |
| **Event Flow** | 1. The actor requires the elimination of a reservation<br>2. The system deletes the reservation<br>3. The system requires the list of the active reservations<br>4. The system shows the list of the active reservations in the Home Page |
| **Exit Condition** | The QR code is displayed |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation<br>3. The list of the active reservations is empty |

## 12. QR Scanner



| | |
|---|---|
| **Name** | QR code Scanner |
| **Actors** | Store Manager |
| **Entry Condition** | The actor is already logged into the system and the system already shows the Home Page |
| **Event Flow** | 1. The Store Manager requires the QR code Scanner<br>2. The system shows the QR code Scanner<br>3. A QR code to scan is given<br>4. The Store Manager scans the QR code using the QR code Scanner<br>5. The system checks the reservation associated to the QR code<br>6. The Store Manager receives a confirmation message |
| **Exit Condition** | A confirmation message is displayed |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation<br>3. The QR code is not valid |

13. Enter the Store


sd enter in the store
Visual Paradigm Online Express Edition
Customer
Store Manager
CLup
QR
Show QR
ref
QR Scanner
alt
[Store Manager has a confirmation message]
Customer could enter
[Store Manager has an error message]
Customer could not enter
Visual Paradigm Online Express Edition

| | |
|---|---|
| **Name** | Enter the Store |
| **Actors** | Customer, Store Manager |
| **Entry Condition** | The actor is already logged into the system and the system already shows the QR code |
| **Event Flow** | 1. The actor shows the QR code at the Store Manager<br>2. The Store Manager scans the QR code<br>3. The Customer receives the authorization to enter the store |
| **Exit Condition** | The Customer can enter the store |
| **Exception** | 1. Loss of internet connection<br>2. The actor cancels the operation<br>3. The QR code is not valid |

### 3.2.2 Mapping

| Goals | Requirements | Domain Assumptions | Use Cases |
|---|---|---|---|

| | | | |
|---|---|---|---|
| G.1 | R.1 R.2 R.6 R.17 R.19 R.22 | D.5 D.6 D.9 D.12 | U.3 U.5 U.6 U.7 U.8 U.9 U.10 U.12 U.13 |
| G.2 | R.1 R.2 R.3 R.4 R.5 R.6 R.9 R.10 R.16 R.21 R.22 | D.1 D.2 D.3 D.6 D.8 D.9 D.12 | U.1 U.2 U.3 U.5 U.6 U.7 U.8 U.9 U.10 U.12 |
| G.3 | R.12 R.13 R.20 | D.9 D.12 | |
| G.4 | R.6 R.7 R.8 R.14 R.15 R.16 R.22 R.24 | D.5 D.6 D.8 D.10 D.11 | |
| G.5 | R.2 R.6 R.7 R.8 R.9 R.10 R.11 R.12 R.13 R.15 R.18 | / | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

| | |
|---|---|
| **G.1** | Allows to regulate the influx of people that enter the building. |
| **D.5** | Customers can get into the store if and only if they scan a valid QR code. |
| **D.6** | All customers that want to access the store use the digital system of line up. |
| **D.9** | Store data are provided by the store owner and are already present in the DB. |
| **D.12** | The system is applied to medium and large shops. |
| **R.1** | The system generates a single QR code to enter and exit the store for each booking or lining up. |
| **R.2** | The system allows to get a reservation for the supermarket. |
| **R.6** | The system allows people (who do not have access to the required technology) digitally line up directly when they are at the store. |

| | |
|---|---|
| **R.17** | The system shows active bookings. |
| **R.19** | The system allows customers to delete a booking. |
| **R.20** | The system allows customers only one lining up at a time for each shop. |
| **R.22** | The system allows the store manager to scan the QR codes. |
| **U.3** | Show Home Page |
| **U.5** | Show Reservation Page |
| **U.6** | Select Shop |
| **U.7** | Line Up for Customer |
| **U.8** | Line Up for Manager |
| **U.9** | Book |
| **U.10** | Show QR |
| **U.12** | Scan QR |
| **U.13** | Enter the Store |

| | |
|---|---|
| **G.2** | Avoids that customers must line up and wait outside of stores for hours. |
| **D.1** | The customers wait until their number is close to being called to approach the store. |
| **D.2** | The time that it takes to visit the supermarket is not uniform. |
| **D.3** | The approximated expected duration of the visit indicated by the customer is reliable. |
| **D.6** | All customers that want to access the store use the digital system of lining up. |
| **D.8** | Customers exit from the store only after they had scanned a valid QR code. |
| **D.9** | Store data are provided by the store owner and are already present in the DB. |
| **D.12** | The system is applied to medium and large shops. |
| **R.1** | The system generates a single QR code to enter and exit the store for each booking or lining up. |
| **R.2** | The system allows to get a reservation for the supermarket. |

| | |
|---|---|
| **R.3** | The system provides customers a precise estimation of the waiting time. |
| **R.4** | The system uses the customers' current position to estimate the waiting time. |
| **R.5** | The system alerts the customers before their shift according to the geolocation information. |
| **R.6** | The system allows people (who do not have access to the required technology) digitally line up directly when they are at the store. |
| **R.9** | The system allows customers to insert the approximate expected duration of the visit. |
| **R.10** | The system infers customers' expected duration of the visit based on an analysis of the previous visits. |
| **R.16** | The system requires a sign up/login. |
| **R.21** | The system uses information about the customer that exit the store to infer better the waiting time. |
| **R.22** | The system allows the store manager to scan the QR codes. |
| **U.1** | Sign Up |
| **U.2** | Lineup |
| **U.3** | Show Home Page |
| **U.5** | Show Reservation Page |
| **U.6** | Select Shop |
| **U.7** | Lineup for Customer |
| **U.8** | Lineup for Manager |
| **U.9** | Book |
| **U.10** | Show QR |
| **U.12** | Scan QR |

| | |
|---|---|
| **G.4** | Everyone must be able to shop, even people who do not have access to the required technology. |
| **D.5** | Customers can get into the store if and only if they scan a valid QR code. |

| | |
|---|---|
| **D.6** | All customers that want to access the store use the digital system of lining up. |
| **D.8** | Customers exit from the store only after they had scanned a valid QR code. |
| **D.10** | The store manager with the digital system helps people who do not have access to the required technology to line up directly when they are at the store. |
| **D.11** | The store manager prints the QR code for people who do not have access to the required technology. |
| **R.6** | The system allows people (who do not have access to the required technology) digitally line up directly when they are at the store. |
| **R.7** | The system suggests alternative time slots for visiting the store when the desired one is not available. |
| **R.8** | The system suggests alternative stores when the desired one is not available. |
| **R.14** | The system shows the available time slots for each grocery. |
| **R.15** | The system provides a QR code printing service. |
| **R.16** | The system requires a sign up/login. |
| **R.22** | The system allows the store manager to scan the QR codes. |
| **R.24** | The system allows customers only one booking at a time for each shop. |
| **U.3** | Show Home Page |
| **U.8** | Lineup for Manager |
| **U.10** | Show QR |
| **U.12** | Scan QR |
| **U.13** | Enter the Store |

| | |
|---|---|
| **G.5** | Provides smart managing of lining up and booking with a digital system. |
| **R.1** | The system generates a single QR code to enter and exit the store for each booking or lining up. |
| **R.2** | The system allows to book a visit to the supermarket. |
| **R.6** | The system allows people (who do not have access to the required technology) digitally line up directly when they are at the store. |

| | |
|---|---|
| **R.7** | The system suggests alternative time slots for visiting the store when the desired one is not available. |
| **R.8** | The system suggests alternative stores when the desired one is not available. |
| **R.9** | The system allows customers to insert the approximate expected duration of the visit. |
| **R.10** | The system infers customers' expected duration of the visit based on an analysis of the previous visits. |
| **R.11** | The system allows store managers to get a lining up only for own store. |
| **R.12** | The system provides periodic notifications of available time slots in a day/time range. |
| **R.13** | The system shows the list of shops. |
| **R.15** | The system provides with a QR code printing service. |
| **R.18** | The system shows the history of bookings. |

## 3.3 Performance Requirements

Due to the non-critical nature of the system, too strict performance requirements are not required. However, in order to offer the best possible user experience, the system should provide:

- the list of available stores in 7 seconds, or less;
- the list of active reservations and their information in less than 5 seconds;
- the QR code generation in roughly 5 seconds;
- the history of reservations in at most 10s;
- the validation of the QR code at the entrance of the shop in less than 12s;
- the lining up and booking confirmation in 10s, or less;
- the loading of the available time slots for a specific store in less than 5s;
- push app notification with e delay that is imperceptible to the user

Notice that a good internet connection is assumed in the previous estimations.

The system should also guarantee simultaneous access to 50 000 Customers and Store Managers.

## 3.4 Design Constraints

### 3.4.1 Standards Compliance

All the specifications that have been described in this document must be respected by the system. Also, the source code of the application must be commented on and documented adequately.

Moreover, the system should respect the line guides described by the European RGPD.

### 3.4.2 Hardware Limitations

The system requires a mobile device and a stable internet connection to work properly.

### 3.4.3 Any Other Constraints

## 3.5 Software System Attributes

### 3.5.1 Reliability and Availability

The system should offer its functionalities with an availability equal to 99.5%, or more. In other words, the system must be inaccessible for less than two days every year. To achieve this goal, the system should provide a high redundancy for the most critical components.

Furthermore, in order to guarantee better reliability performances, all the scheduled maintenance intervents on the system should be done during the night or public holidays.

### 3.5.2 Security

The connection between the Customers and Store Managers application and the server must be safe. To keep a good level of security, the system should use the SSL/TLS (*Secure Sockets Layer /Transport Layer Security*) standard technology.

Moreover, all passwords must be encrypted.

### 3.5.3 Maintainability

The source code must be commented on as well as possible and the correlated documentation must be kept updated during the whole life cycle of the system.

Modularity and low coupling between components must be a focus during the developing phases.

### 3.5.4 Portability

In the first phases of developing and launch of the system, the only supported platform will be Android. Further, future developments could take into account other platforms such as iOS and web applications.

# 4 FORMAL ANALYSIS USING ALLOY

## 4.1 Objectives of the analysis

In this section a presentation of the formal modelling activity that has been done using the Alloy formal notation. Goals of this activity is formally describing of the domain and properties of the system to be.

In particular, the main objectives of this activity are model and formally represents the following entities:

- actors of the system to be;
- reservations and their subtypes (lining up and booking);
- status of a reservation;
- shops and their elements;
- QR codes.

Moreover, the purpose is also to model:

- that each customer can have only one active reservation at a time;
- that each store manager is associated to only one store;
- the linking between QR codes and reservations;
- the possibilities for the store managers to get a reservation only for their own shops;
- the limits on the number of people into the shop.

Further constraints and details are directly provided as comment in the following code. There are present also the definition of several assertions and predicates.

## 4.2 Alloy Code

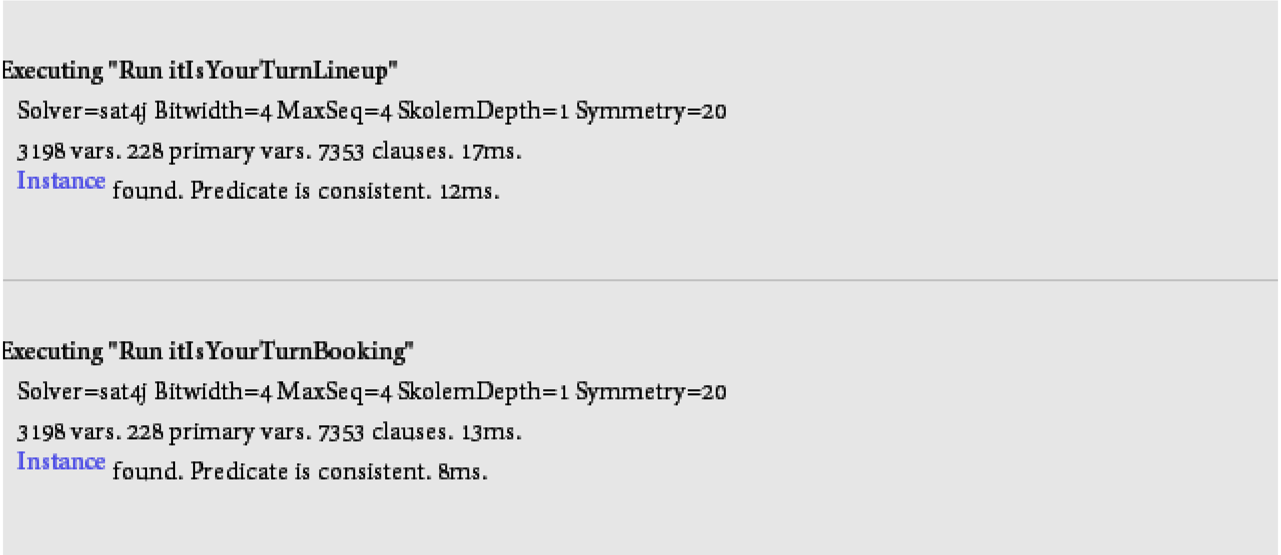
```
//+++++ DECLARATION +++++

// Definition of boolean type

abstract sig Bool {}

one sig True extends Bool {}

one sig False extends Bool {}


// Definition of status of reservation

abstract sig Status{}

one sig Wait extends Status{} // the customer is waiting

one sig Enter extends Status{}// the customer has entered the shop but that has not yet exit

one sig Exit extends Status{} // the customer has left the shop


// Definition of actors

abstract sig Person{}

sig Customer extends Person{

lineup: set Lineup,

booking: set Booking,
}
sig StoreManager extends Person{
lineup: set Lineup,
shop: one Shop,
}

// Definition of reservation
abstract sig Reservation{
status: one Status,
qrcode: one QRCode,
}{}
sig Lineup extends Reservation{
customer: lone Customer ,
storemanager: lone StoreManager,
shop: one Shop,
}{#customer = 1 <=> #storemanager =0}
sig Booking extends Reservation{
customer: one Customer,
shop: one Shop,
}

// Definition of shop
// where an enter line up (booking) is a line up (booking) associated to a customer that
//  has entered the shop but that has not yet exit
sig Shop{
storemanager: one StoreManager,
enterLineup: set Lineup,
enterBooking: set Booking,
lineup: set Lineup,
booking: set Booking,
capacity: Int,
}{capacity>0}
```

```
// Definition of QR code used to enter and exit the shop
sig QRCode{
reservation: one Reservation,
valid: Bool,
}{(reservation.status = Exit implies valid = False) and (reservation.status != Exit implies valid = True)}

//+++++ FACT +++++
// Only one active lining up for each customer at a time for each shop
fact OneLineUp{
all disj l1,l2: Lineup | (l1.status != Exit and l2.status != Exit) implies
(l1.customer != l2.customer or l1.shop != l2.shop)
}

// Only one active booking for each customer at a time for each shop
fact OneBooking{
all disj b1,b2: Booking | (b1.status != Exit and b2.status != Exit) implies (
b1.customer != b2.customer or b1.shop != b2.shop)
}
// Each store manager belongs to only one shop
fact ShopUnique {
all disj s1,s2: Shop | s1.storemanager != s2.storemanager }

// If a store manager belongs to a shop, then that shop has that store manager
fact StoreManagerCoherence {
all st: StoreManager, sh: Shop | st.shop = sh iff sh.storemanager = st }

// If a customer book a visit, then that booking is associated to that customer
fact BookingCustomerCoherence {
all b: Booking, c: Customer | b.customer = c iff c.booking = b }

// If a customer lines up, then that lining up is associated to that customer
```

```
fact LineupCustomerCoherence {
all l: Lineup, c: Customer | l.customer = c iff c.lineup = l }

// If a store manager lines up, then that lining up is associated to that store manager
fact LineupManagerCoherence {
all l: Lineup, s: StoreManager | l.storemanager = s iff s.lineup = l }

// If a reservation is associated to a QR code, then that QR code is associated to that reservation
fact QRReservationCoherence {
all r: Reservation, q: QRCode | r.qrcode = q iff q.reservation = r }

// If a lining up is associated to a shop, then that shop is associated to that lining up
fact ShopLineupCoherence {
all s: Shop, l: Lineup | s.lineup = l iff l.shop = s }

// If a booking is associated to a shop, then that shop is associated to that booking
fact ShopBookingCoherence {
all s: Shop, b: Booking | s.booking = b iff b.shop = s }

// A store manager can line up only on own shop
fact StoreManagerOnlyLineupOwnShop {
all sh: Shop, st: StoreManager | st.lineup.shop = sh implies st.shop = sh }

// An enter lineup is a lineup associated to a customer that has entered the shop
// but that has not yet exit
fact EnterLineupOwner{
all s:Shop | s.enterLineup.status= Enter
}

// If an enter line up is associated to a shop, then that shop contains that enter line up
fact EnterLineupOwner{
all s: Shop, l: Lineup | s.enterLineup = l iff l.shop = s
}

// An enter lineup is a lineup associated to a customer that has entered the shop
// but that has not yet exit
fact EnterBookingOwner{
all s:Shop | s.enterBooking.status=Enter
}

// If an enter booking is associated to a shop, then that shop contains that enter booking
fact EnterLineupOwner{
all s: Shop, b:Booking | s.enterBooking = b iff b.shop = s
}

// The sum of people into the shop is less or equal than the capacity of the shop
fact CapacityConstraint{
all s:Shop | (#s.enterLineup+#s.enterBooking) <= s.capacity
}

//+++++ PREDICATE +++++
// The customer who lined up can enter the shop
pred itIsYourTurnLineup(l:Lineup){
l.status = Enter
}
run itIsYourTurnLineup

// The customer who booked a visit can enter the shop
pred itIsYourTurnBooking(b: Booking){
b.status = Enter
}
run itIsYourTurnBooking
```

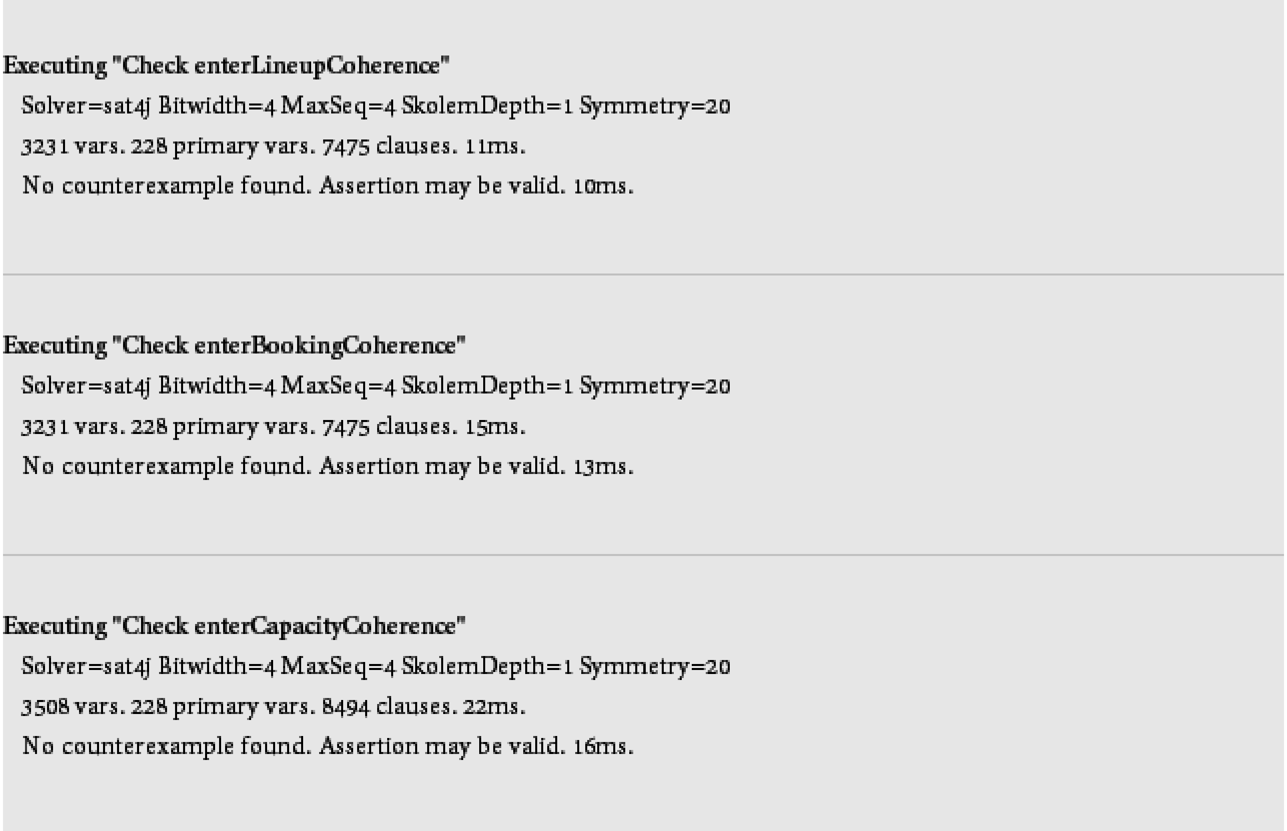
```
//Predicate show

pred show {}

run show for 5


//+++++ ASSERTION +++++

// Checks if the number of enter lineup is less or equal than the total number of lining up

// for each shop

assert enterLineupCoherence {

all s:Shop | #(s.enterLineup) <= #(s.lineup)

}

check enterLineupCoherence


// Checks if the number of enter booking is less or equal than the total number of lining up

// for each shop

assert enterBookingCoherence {

all s:Shop | #(s.enterBooking) <= #(s.booking)

}

check enterBookingCoherence


// Checks if the sum of all customers into all shops in less or equal than the sum of

// all capacities

assert enterCapacityCoherence {

all s:Shop | sum(#(s.enterBooking)) + sum(#(s.enterLineup))<= sum(s.capacity)

}

check enterCapacityCoherence
```

## 4.3 Predicates Results



```
Executing "Run itIsYourTurnLineup"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  3198 vars. 228 primary vars. 7353 clauses. 17ms.
  Instance found. Predicate is consistent. 12ms.

Executing "Run itIsYourTurnBooking"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  3198 vars. 228 primary vars. 7353 clauses. 13ms.
  Instance found. Predicate is consistent. 8ms.
```

## 4.4 Assertions Results



```
Executing "Check enterLineupCoherence"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  3231 vars. 228 primary vars. 7475 clauses. 11ms.
  No counterexample found. Assertion may be valid. 10ms.

Executing "Check enterBookingCoherence"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  3231 vars. 228 primary vars. 7475 clauses. 15ms.
  No counterexample found. Assertion may be valid. 13ms.

Executing "Check enterCapacityCoherence"
  Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
  3508 vars. 228 primary vars. 8494 clauses. 22ms.
  No counterexample found. Assertion may be valid. 16ms.
```

## 4.4 Models

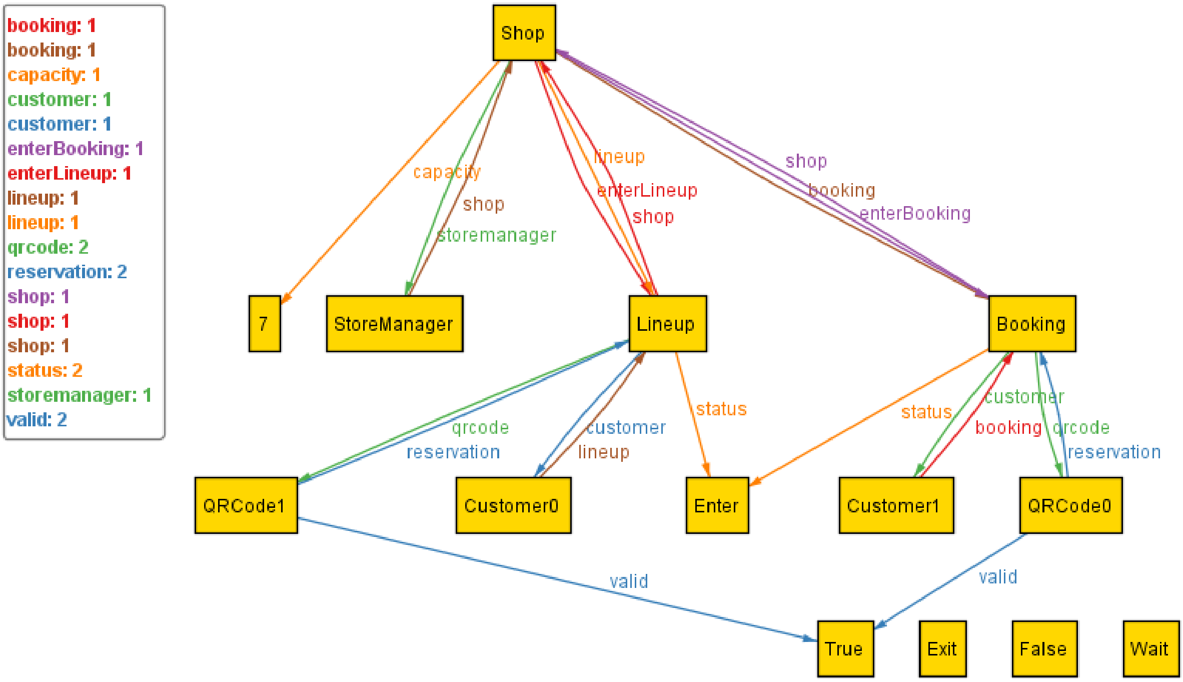


*Figure 3: predicate show*

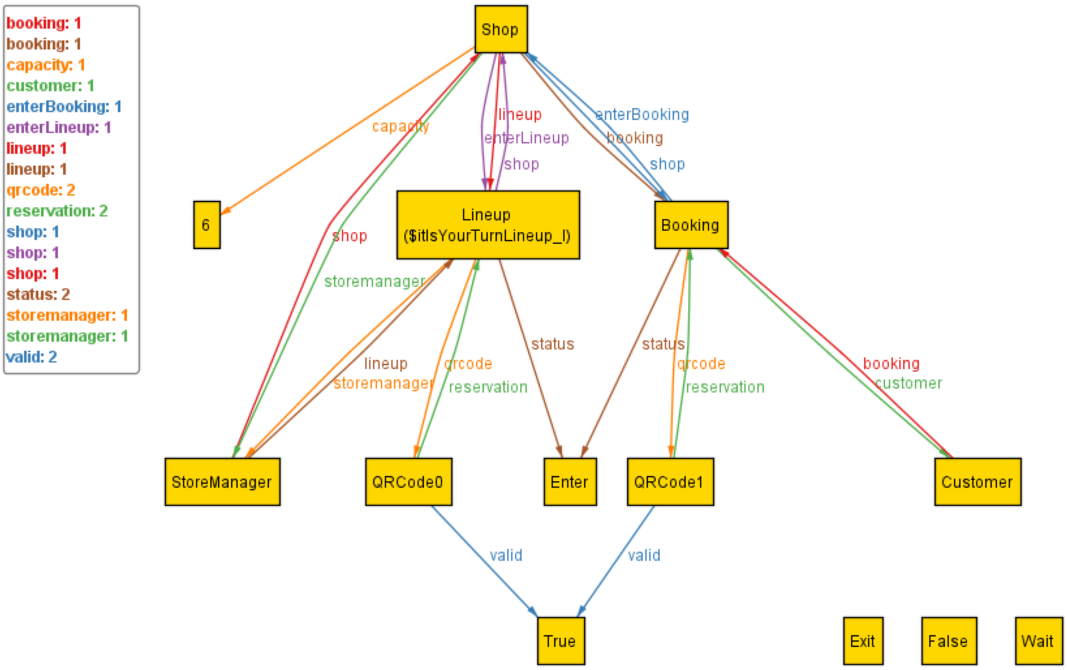


*Figure 4: predicate itIsYourTurnLineup*

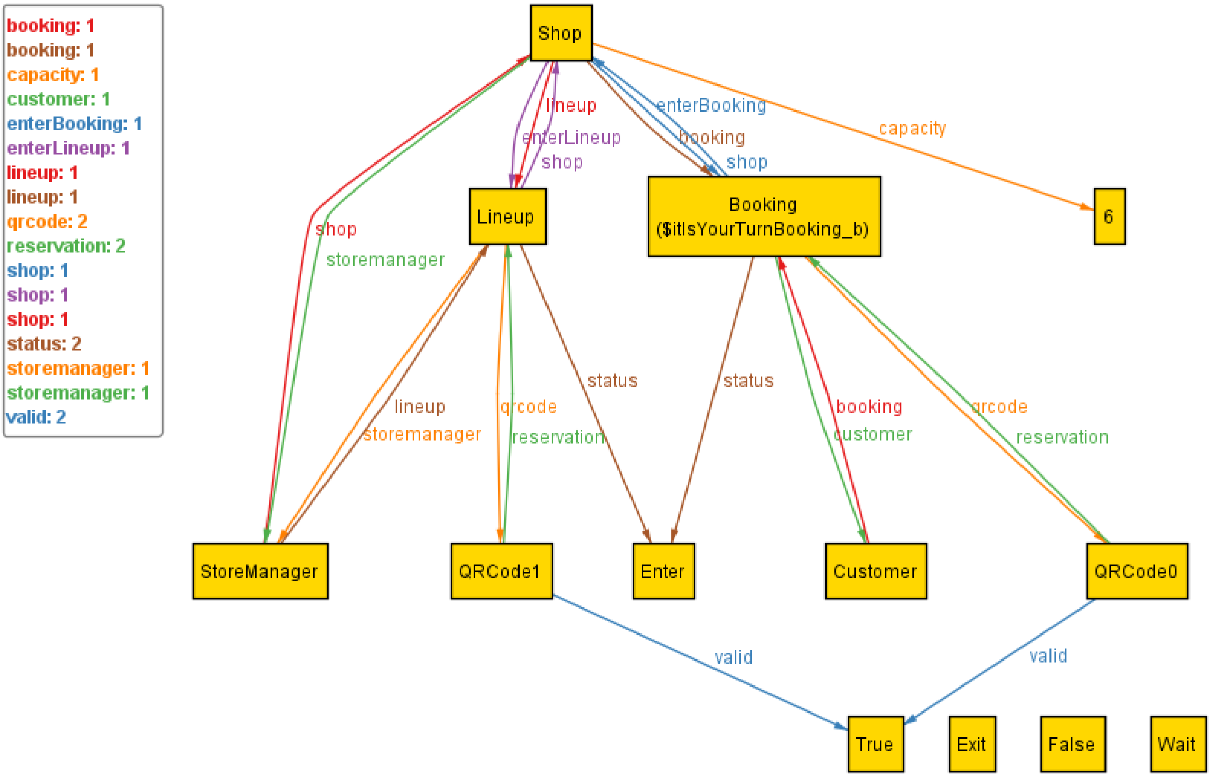


*Figure 5: predicate itIsYourTurnBooking*

# 5 EFFORT SPENT

This section shows the amount of time that each member has spent to produce the document. Please notice that each section, diagram, and specification is the result of a coordinated work. The column *Member* specifies only the main contributor (or contributors, if more than one) for each topic but should not be interpreted as a lack of participation by other team members.

| TOPIC | MEMBER | HOURS |
|---|---|---|
| General initial brainstorming and interpretation of the domain | Digregorio, Massaro, Tamma | 5h |
| Creation of the document and identification of the goals | Digregorio, Massaro, Tamma | 4h |
| World and shared phenomena | Digregorio, Massaro, Tamma | 4h |
| Requirements | Digregorio, Massaro, Tamma | 7h |
| Mock-up | Digregorio | 3.5h |

| | | |
|---|---|---|
| Mapping between goals, requirements, and domain assumptions | Digregorio, Massaro, Tamma | 6h |
| Improvement on goals, requirements, and domain assumptions | Digregorio, Massaro, Tamma | 3.5h |
| Product Perspectives, Product Functions and User Characteristics | Digregorio, Massaro | 3h |
| Brainstorming and discussion of some key points | Digregorio, Massaro, Tamma | 5h |
| Document formatting and other improvements | Digregorio, Massaro | 2h |
| Sequence Diagrams | Tamma | 15h |
| Sequence Diagrams descriptions | Digregorio | 6h |
| Class Diagram | Massaro | 2h |
| New brainstorming and new discussion of some key points | Digregorio, Massaro, Tamma | 3h |
| Alloy | Digregorio, Massaro, Tamma | 8h |
| Inserting of comments and explanations on Alloy | Massaro | 1.5h |
| Definition of Performance Requirements and Design Constraints | Digregorio, Massaro, Tamma | 1h |
| Drafting of Performance Requirements and Design Constraints | Digregorio | 1h |
| Definition of Software System Attributes | Digregorio, Massaro, Tamma | 1h |
| Drafting of Software System Attributes | Massaro | 1h |
| Alloy improvements and corrections | Massaro | 4h |
| Alloy description, integration into the document and final adjustments | Digregorio | 2h |

# 6 REFERENCES