



POLITECNICO
MILANO 1863

CLup

Requirement Analysis and Specification Document

Simone Abelli
Stefano Azzone

Deliverable:	RASD
Title:	Requirement Analysis and Verification Document
Authors:	Simone Abelli, Stefano Azzone
Version:	1.0
Date:	7 November 2016
Download page:	https://github.com/StefanoAzzone/AbelliAzzone
Copyright:	Copyright © 2017, Simone Abelli, Stefano Azzone – All rights reserved

Contents

1	Introduction	4
1.1	Purpose	4
1.2	Scope	4
1.2.1	World Phenomena	5
1.2.2	Shared Phenomena	5
1.2.3	World Phenomena	5
1.3	Definitions, Acronyms, Abbreviations	5
1.3.1	Definitions	5
1.3.2	Acronyms	5
1.3.3	Abbreviations	5
1.4	Revision history	5
1.5	Reference Documents	6
1.6	Document Structure	6
2	Overall Description	7
2.1	Product perspective	7
2.1.1	Customer	7
2.1.2	Store owner	7
2.1.3	State charts	8
2.2	Product functions	9
2.3	User characteristics	10
2.4	Assumptions, dependencies and constraints	10
2.4.1	Domain Assumptions	10
3	Specific Requirements	11
3.1	External interface requirements	11
3.1.1	User interfaces	11
3.1.2	Hardware interfaces	11
3.1.3	Software interfaces	11
3.1.4	Communication interfaces	12
3.2	Functional requirements	12
3.2.1	Use cases	12
4	Formal Analysis Using Alloy	17
5	Effort Spent	18
5.1	Simone Abelli	18
5.2	Stefano Azzone	18
	References	19

1 Introduction

1.1 Purpose

The coronavirus emergency has put a strain on society on many levels, in particular, grocery shopping can become a challenge in the presence of such strict rules: supermarkets need to restrict access to their stores to avoid having crowds inside and long lines outside. The goal of this project is to develop an easy-to-use application that, on the one side, allows store managers to regulate the influx of people in the building and, on the other side, saves people from having to line up and stand outside of stores for hours on end.

The application will allow customers to “line up” (i.e., retrieve a number) from their home, and then wait until their number is called (or is close to being called) to approach the store. In addition, the application could be used to generate QR codes that would be scanned upon entering the store, thus allowing store managers to monitor entrances.

Follow the goals:

- G1 CLup should allow customers to queue up remotely and on premise in such a way that they don't need to form a physical line
- G2 CLup should allow store owners to regulate how many customers can be simultaneously in their stores
- G3 CLup should provide the customer with a reasonably precise estimate of waiting time
- G4 CLup should alert the customers when it is time to get to the shop taking into account travel time
- G5 CLup should allow customers to book future visits to stores
- G6 CLup should allow customers to specify estimated visit duration and desired objects in order to provide a better guess of waiting time for all the customers
- G7 CLup should be able to infer an approximate duration of the visit from an analysis of the previous ones to plan visits and manage the queue in a finer way

1.2 Scope

The system to be allows to avoid creating queues in front of stores. This is accomplished by enabling the customers to queue up remotely. Moreover, the shop owners can oversee customers entering and exiting stores.

The system offers the following functionalities:

- F1 it allows customers to line up remotely or on premise
- F2 it allows customers whose position in queue allows it to enter and exit the store
- F3 it schedules customers in order to minimize overcrowding inside and outside of the store
- F4 it alerts customers when they should head to the store
- F5 it allows customers to book a visit and optionally specify duration and desired categories of products
- F6 it allows store owners to set how many customers can be simultaneously in their stores

F7 it allows store owners to register stores

F8 it uses statistics build on entrance and exit data, and preferred categories to better evaluate duration of visits

1.2.1 World Phenomena

WP1 Customer reaches a store

WP2 Customer enters or exits a store

WP3 Store owner controls current number of customers in one of his stores

WP4 Customer buys products

1.2.2 Shared Phenomena

SP1 Customer queues up

SP2 Customer is identified in order to allow entrance or exit from a store

SP3 Customer is alerted when his turn is close

SP4 Customer books a visit to a store

1.2.3 World Phenomena

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

Ticket	Virtual or physical artifact used to identify the position of a customer in a queue
Reservation	A place in the queue
Enqueued	A customer is enqueued when he has provided the system with a means of identification and requested a reservation
Authorized	A customer is authorized when he has been enqueued and is allowed temporary access access the store.
Occupation	Number of customers currently present in the store

1.3.2 Acronyms

RASD	Requirement Analysis and Specification Document
GPS	Global Positioning System
S2B	Software to be
UI	User Interface

1.3.3 Abbreviations

Gn	Goal number n
Rn	Requirement number n
Dn	Domain Assumption number n

1.4 Revision history

Not yet defined.

1.5 Reference Documents

1. IEEE Std 830-1998 Recommended Practice for Software Requirements Specifications
2. Specification Document: R&DD Assignment A.Y. 2020/2021

1.6 Document Structure

- Chapter 1: gives an introduction about the project, describing the purpose of the system informally and defining its scope, its main goals, world and shared phenomena. Moreover this section contains specifications such as the definitions, acronyms, abbreviation, revision history of the document and the references.
- Chapter 2: contains the overall description of the project, with a more in-depth look at its functionalities. Here are identified the main actors involved in the application's usage lifecycle, some scenarios that point out the major features of the S2B, and all the necessary domain assumptions, dependencies and constraints. This section also provides a class diagram, which aid to better understand the general structure of the project, and some state diagrams, to make the evolution of the crucial objects clear.
- Chapter 3: This section contains the core of the document: first it presents the interface requirement including user, hardware, software and communication interfaces. Then it offers the specification and the description of all the functional requirements necessary in order to reach the goals; is also provided a list of use cases, with their corresponding sequence diagrams and their mapping on the requirements, as long as some scenarios, useful to identify specific cases in which the application can be utilized. Finally non-functional requirements are defined, including performance, design and the software systems attributes.
- Chapter 4: includes the alloy code and the corresponding metamodels generated from it, in order to show how the project has been modeled and represented through the language.
- Chapter 5: shows the effort which each member of the group spent working on the project.
- Chapter 6: includes the reference documents.

2 Overall Description

2.1 Product perspective

The S2B will be used by two kinds of users: customers and store owners, and offers to each of them different functionalities:

2.1.1 Customer

The customer can create reservations for a desired store: such reservation is then inserted into a queue. This queue is managed in order to avoid overcrowding. To achieve this, statistics about visits' durations are exploited, taking also into account information that customers may provide (e.g. which department of the store they want to visit).

There are multiple types of reservations that can be requested:

1. Immediate reservation: the customer wants to queue up immediately to a store. He/she is also provided with an estimation of the time he has to wait before his/her turn.
2. Future reservation: the customer wants to book a future visit at a desired time and store. When creating the reservation the customer can specify how long he/she intends to stay and which departments of the market he/she plans to go to, in order to provide a better plan. If he/she does not specify the duration the system can infer it using some statistic built on his/her previous visits. Then the customer will be provided with the actual time he will be able to access the store (considering the bookings from other customers).
3. On premise reservation: the customer is allowed to enqueue directly at the store. Each of them will provide a system to print tickets so that those who do not have access to the required technology for the previous options can still line up. These tickets contain a reservation number and an estimation of the waiting time before being able to enter.

Customers who requested an immediate or a future reservation receive an alert when they need to depart to reach the store. This alert is based on the location of the customer, so that the time needed to reach the store is taken into account. Moreover immediate and future reservations can be deleted by who created them.

2.1.2 Store owner

The store owner can

1. register a store to the system so that it can be accessed by customers
2. set the maximum occupation threshold for each of his stores.
3. monitor the number of customers inside at any time
4. visualize statistics about the flow of clients and department occupation

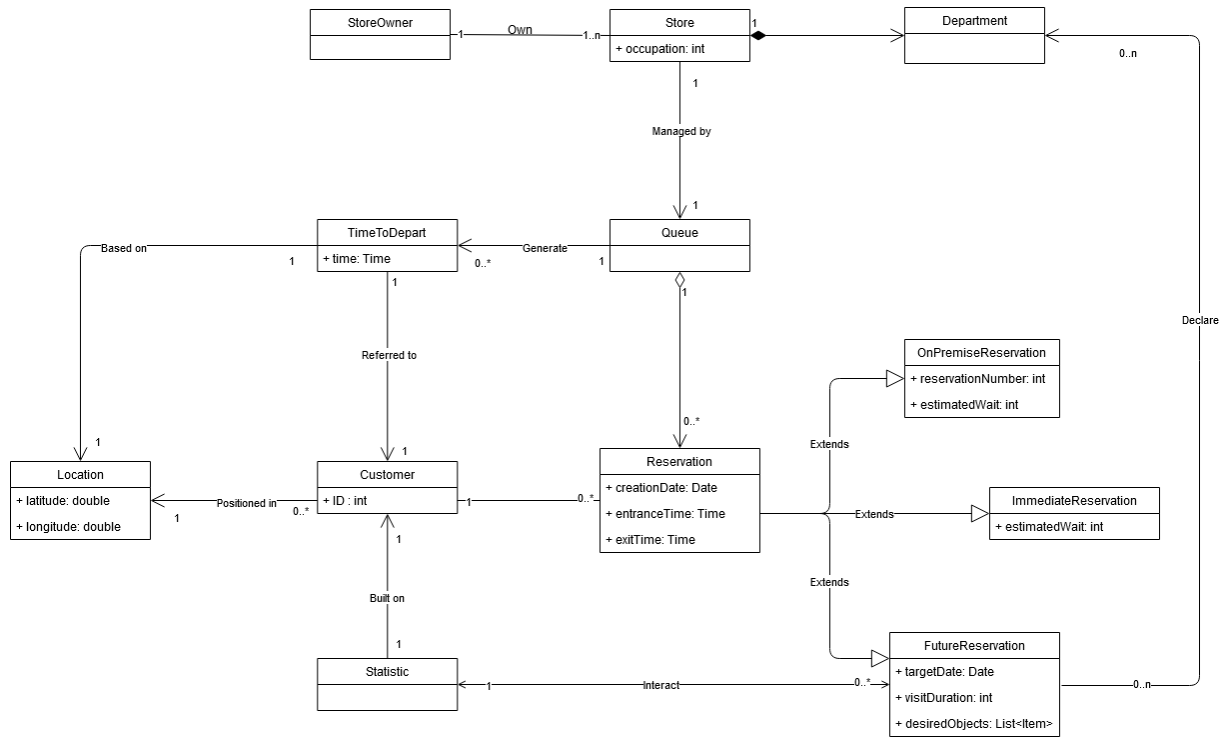


Figure 1: Class Diagram.

2.1.3 State charts

Here we show the main processes that the system will manage, and the states in which the system will find itself.

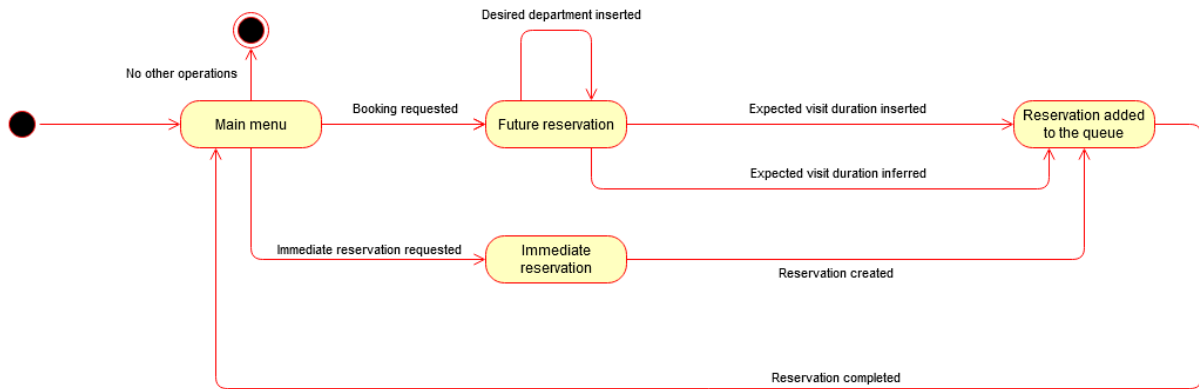


Figure 2: State Diagram 1: Creation of a reservation.

The figure above illustrates how the customer creates a reservation (virtual customer). The customer accesses the main menu. From there he/she can request an immediate reservation or a future reservation. If he/she requests an immediate reservation, it is created and the process terminates. If he/she requires a future reservation he can add the departments he/she intends to visit, and how much time he/she is going to spend at the store. Then the reservation is created and the process terminates.

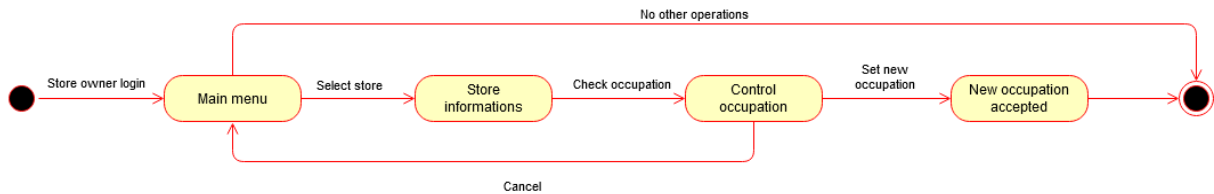


Figure 3: State Diagram 2: Store owner sets occupation.

The figure above illustrates how the store owner sets the maximum occupation of one of his stores. The store owner logs in and is provided a main menu. He/she selects one of his stores, and is presented with the current maximum occupation of his selected store. He can change it to a value to his liking, or go back to the main menu.

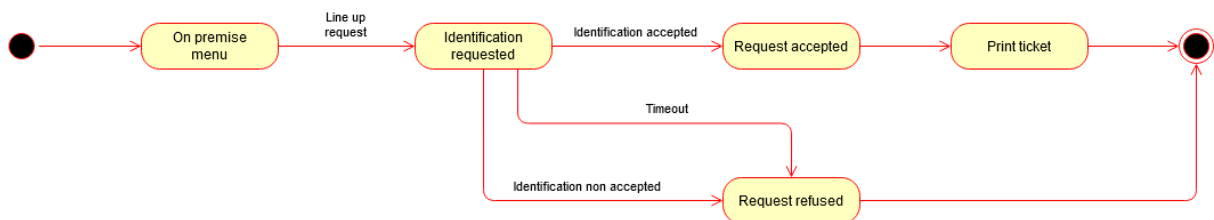


Figure 4: State Diagram 3: On premise creation of a reservation.

The figure above illustrates how a customer can queue up on premise (physical customer). The customer is presented with an on premise main menu. From there he/she can request to queue up (create a reservation) immediately. He will be prompted to identify himself (e.g. social security card). Once identified the customer will be provided a ticket with a number that represents his position in queue and an estimate of the waiting time.

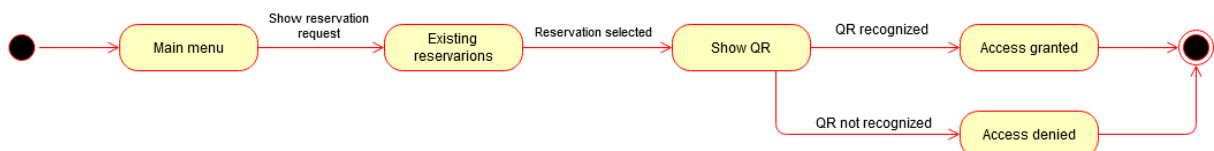


Figure 5: State Diagram 3: Customer enters a store.

The figure above illustrates how a customer who is authorized can access the store. At the entrance of his/her store of choice he is enabled to demonstrate his authorization. Once he/she does, he is allowed to enter the store.

2.2 Product functions

We describe here functions that the S2B will support.

1. Customers can queue up virtually or physically at one of the stores registered to the service.
2. Customers can access the desired store when its population reaches acceptable levels.
3. Customers that queued up virtually can be notified when it is time for them to depart to reach the store.
4. Store owners can monitor the population in each of their stores.
5. Store owners can define the maximum population of each of their stores.
6. Store owners can register stores to the system.

2.3 User characteristics

1. **Customers:** a physical person that needs to access any of the stores registered to the system. The customers belong to all demographics, thus the need for a user-friendly interface for both virtual and physical customers. The customer needs a reasonably precise estimate of waiting time and time to get to the store.
2. **Store owners:** a physical or legal person that owns any number of stores and needs to enable customer access through a queue system.

2.4 Assumptions, dependencies and constraints

2.4.1 Domain Assumptions

- D1 The stores have means to allow entrance or exit to customers that are authorized (e.g. with QR/NFC connected turnstiles).
- D2 Outside stores a device can identify customers (e.g. with social security card) and print tickets.
- D3 When provided, user location has maximum error of 5 meters.
- D4 There is no way for a customer to enter a store except from entrance and exit.
- D5 Each virtual and physical customer has means of identification unique to that user (e.g, telephone number, social security card).

3 Specific Requirements

3.1 External interface requirements

3.1.1 User interfaces

There are two categories of users that have different interface requirements:

- **Customers**

Customers belong to all demographics so a user friendly interface is needed. The customer is presented with a main menu which allows him/her to:

- line up immediately (immediate reservation) at a specific store
- book a visit (future reservation) at a specific store
- view and delete existing reservations

The customer will receive a notification when it is time for him/her to depart to reach the shop.

- **Store owners**

The store owner is presented with a main menu from which he/she can:

- register a store to the system
- delete a store from the system
- view and edit occupation for currently registered stores

3.1.2 Hardware interfaces

The S2B requires the following hardware interfaces:

1. **Computer or smartphone**

Users will need a computer or a smartphone to access the system's services which are provided via an application (only for smartphone owners) and a web application. Only users with an application will be able to receive notifications that alert them when it is time for them to depart and reach the store.

2. **Turnstiles**

Turnstiles allow authorized customers to enter or exit the store by providing a means of identification. (i.e. QR, NFC)

3. **Ticket printer**

The ticket printers located outside stores allows potential customers to queue up on premise provided that they identify themselves (e.g. social security card).

4. **Monitor**

A monitor located outside stores allows customers that queue up on premise to know when it is time for them to access the store.

3.1.3 Software interfaces

The system uses a public API to locate the customer and provide him/her with notifications about the time he will need to depart to reach the store.

3.1.4 Communication interfaces

Customers can access the system through a working internet connection.

3.2 Functional requirements

3.2.1 Use cases

3.2.1.1 Customer registration

Name	Customer registration
Actors	Customer
Entry condition	Customer has opened the smartphone application or the web app on his computer but has not logged in
Event flow	<ol style="list-style-type: none">1. a registration menu is provided to the customer2. from such menu the customer selects the option to sign up as customer3. the customer is then prompted to insert identifying information4. the customer inserts requested information5. the system validates provided information6. the system confirms the registration of the customer and saves information provided
Exit conditions	Customer has registered to the system
Exceptions	<ol style="list-style-type: none">1. a customer with same identifying information already exists2. validation of identifying information is not successful3. the customer decides to cancel the registration <p>If one of the events described above occur, the application will alert the customer and provide him with the possibility to retry or go back to the initial menu</p>

3.2.1.2 Store owner registration

Name	Store owner registration
Actors	Store owner
Entry condition	Store owner has opened the smartphone application or the web app on his computer but has not logged in
Event flow	<ol style="list-style-type: none"> 1. a registration menu is provided to the store owner 2. from such menu the store owner selects the option to sign up as store owner 3. the store owner is then prompted to insert identifying information 4. the store owner inserts requested information 5. the system validates provided information 6. the system confirms the registration of the store owner and saves information provided
Exit conditions	Store owner has registered to the system
Exceptions	<ol style="list-style-type: none"> 1. a store owner with same identifying information already exists 2. validation of identifying information is not successful 3. the store owner decides to cancel the registration <p>If one of the events described above occur, the application will alert the store owner and provide him with the possibility to retry or go back to the initial menu</p>

3.2.1.3 Customer queuing up remotely

Name	Customer queuing up remotely
Actors	Customer
Entry condition	Customer has logged in on the smartphone application or the web app on his computer
Event flow	<ol style="list-style-type: none"> customer selects the option to queue up from main menu customer selects a store from the list of stores registered to the system customer selects to queue up <ul style="list-style-type: none"> with an immediate reservation with a future reservation (and optionally inserts desired categories of item he/she intends to buy and how much time he/she intends to spend at the store) customer submits reservation request
Exit conditions	Customer reservation is confirmed
Exceptions	<ol style="list-style-type: none"> customer decides to cancel the reservation <p>If one of the events described above occur, the application will alert the customer and provide him with the possibility to retry or go back to the initial menu</p>

3.2.1.4 Customer queuing up on premise

Name	Customer queuing on premise
Actors	Customer
Entry condition	Customer has reached the store ticket printer
Event flow	<ol style="list-style-type: none"> customer selects the option to queue up from main menu of the ticket printer customer provides the ticket printer with a means of identification (e.g. social security card)
Exit conditions	<p>Customer reservation is confirmed and a ticket is printed containing the following information:</p> <ul style="list-style-type: none"> how much time he needs to wait before being able to enter the store a progressive number that will allow him to know when his/her turn is
Exceptions	<ol style="list-style-type: none"> customer decides to cancel the reservation <p>If one of the events described above occur, the application will alert the customer and provide him with the possibility to retry or go back to the initial menu</p>

3.2.1.5 Customer entering store

Name	Customer entering store
Actors	Customer
Entry condition	Customer has logged in on the smartphone application and is at the entrance of a store
Event flow	<ol style="list-style-type: none"> 1. customer selects the option to show existing reservations from main menu 2. customer selects an existing reservation and if authorized (i.e. it is his turn to enter the store) he/she is given the means to identify him/herself (e.g. display QR or activate NFC) 3. customer identifies him/herself at the turnstiles
Exit conditions	Customer enters the store
Exceptions	<ol style="list-style-type: none"> 1. customer is not authorized to enter the store <p>If one of the events described above occur, the application will alert the customer and provide him with the possibility to retry or go back to the initial menu</p>

3.2.1.6 Customer exiting store

Name	Customer exiting store
Actors	Customer
Entry condition	Customer has logged in on the smartphone application and is in a store
Event flow	<ol style="list-style-type: none"> 1. customer selects the option to show existing reservations from main menu 2. customer selects an existing reservation and if authorized (i.e. it is his turn to enter the store) he/she is given the means to identify him/herself (e.g. display QR or activate NFC) 3. customer identifies him/herself at the turnstiles
Exit conditions	Customer exits the store
Exceptions	

3.2.1.7 Store owner registering a store

Name	Store owner registering a store
Actors	Store owner
Entry condition	Store owner has logged in on the smartphone application or the web app on his computer
Event flow	<ol style="list-style-type: none"> 1. store owner selects the option register a store from main menu 2. store owner inserts necessary information and sets up equipment 3. store owner submits store registration request
Exit conditions	Store registration is confirmed
Exceptions	<ol style="list-style-type: none"> 1. store owner decides to cancel the store registration 2. information is missing or incorrect 3. equipment is not working properly <p>If one of the events described above occur, the application will alert the customer and provide him with the possibility to retry or go back to the initial menu</p>

3.2.1.8 Store owner setting maximum occupation of one of his stores

Name	Store owner setting maximum occupation of one of his stores
Actors	Store owner
Entry condition	Store owner has logged in on the smartphone application or the web app on his computer
Event flow	<ol style="list-style-type: none"> 1. store owner selects one of his stores from the list reachable from the main menu 2. store owner views current occupation of his store, and current occupation threshold 3. sets the new desired occupation threshold
Exit conditions	The new occupation threshold is set
Exceptions	<ol style="list-style-type: none"> 1. the threshold value is inadequate <p>If one of the events described above occur, the application will alert the customer and provide him with the possibility to retry or go back to the initial menu</p>

4 Formal Analysis Using Alloy

```
sig Customer{
    opens: set Reservation
}

sig StoreOwner{
    owns: set Store
}

sig Store{}
{
    -- store has at most one owner
    no s: Store, disj o1, o2: StoreOwner | s in o1.owns and s in o2.owns
    --store has at least one owner
    all s: Store | some o: StoreOwner | s in o.owns
}

abstract sig Reservation{
    refersTo: one Store
}
{
    -- reservation is opened by at most one customer
    no r: Reservation, disj c1, c2: Customer | r in c1.opens and r in c2.opens
    -- reservation is opened by at least one customer
    all r: Reservation | some c: Customer | r in c.opens
}

sig ImmediateReservation extends Reservation{}
sig FutureReservation extends Reservation{}
sig OnPremiseReservation extends Reservation{}

pred show{
    #Customer >=3
}

run show for 5
```

5 Effort Spent

Provide here information about how much effort each group member spent in working at this document. We would appreciate details here.

5.1 Simone Abelli

Writing goals	3
UML class diagram	4
Product perspective	3
State charts, External interface requirements	3

5.2 Stefano Azzone

Writing goals	3
UML class diagram	3
Product perspective	3
Product functions, user characteristics, domain assumptions	2
State charts, External interface requirements	3
Use cases	2
Started alloy section	2

References

- [1] S. Bernardi, J. Merseguer, and D. C. Petriu. A dependability profile within MARTE. *Software and Systems Modeling*, 10(3):313–336, 2011.