POLITECNICO
MILANO 1863

Requirements analysis and specification document (RASD)

Davide Rossetto 894029, Alessandro Tatti 883861

Delivery date: 2017 May 07

v1.3

Contents

Section 1

Introduction

1.1 Purpose

The Requirements Analysis and Specification Document for the Travlendar+ system management is intended to describe the system itself, its functional and non-functional requirements, its components, its constraints, the relationship with the real world, and users by providing several use cases and scenarios.

A part of the documentation uses Alloy, a language to describe structures and a tool to explore and provide a formal specification of some features of the system.

This document is predisposed primarily to developers and programmers who must meet the requirements, testers who need to determine if the requirements are met, project managers, who control the development process, and users who validate the goals of the system.

1.2 Scope

The product is a digital management system to support the creation of a calendar-based application that automatically computes and accounts for travel time between appointments to make sure that users are not late for appointments and support users in their travels.

The system consists of two back – end server applications:

- An application that handles requests for entering appointments or trips, managing schedules and routes;
- An application that communicates with the systems of transportation companies. ***

And two front – end applications:

- The web-based application to provide the end user with a friendly interface to take advantage of services of Travlendar+;
- A mobile application that allows the user to easily access the service wherever he needs.***

The system is intended for users who must be allowed to register and access the system via username and password, to make the appointment and management process easier and quicker.

Users can create meetings, and when meetings are created at locations that are unreachable in the allotted time, a warning is created; the application must also take into account possible issues related to the request (e.g. public service strikes on the scheduled day for the meeting).

The system should allow users to define various kind of user preferences: user can activate or deactivate each travel means, should be able to provide constraints on different travel means and select combinations of transportation means that minimize carbon footprint. In addition, the user must specify a flexible lunch: the system must handle this, allowing the user to have half an hour to have lunch within the set time interval.

1.3 Goal

The goals of Travlendar+ are the followings:

1. Let the user register to the service and login via provided credentials;
2. Let the user manage his/her own profile;
3. Let the user insert his/her meeting in the schedule application;
4. Let the system work efficiently by generating an alarm when a meeting is not possible within the specified time range;
5. Let the system indicate which best travel means is to be used for a given meeting;
6. Let the user indicate his/her preferences on the travel means;
7. Automatically the system searches for the shortest path to reach the meeting site;
8. Automatically the system searches the cheapest means of transport to reach the meeting site;
9. Let the user specify a flexible lunch, i.e. a period of time to eat;
10. Let the user specify its intent to minimize his carbon footprint;
11. Automatically the system must find at least half an hour to have lunch (within the ”flexible lunch” time).

1.4 Definitions, Acronyms, Abbreviations

Actor: Specifies a role played by a user or any other system that interacts with the system;

API: Application Programming Interfaces.

Back – end application: Computer program that remains in the background, or resides on a server located in a back room. A user, generally, interfaces only with a front ? end application.

Front – end application: Any application the users interact with directly. It provides the so called presentation layer.

GPS: Global Positioning System

Guest: Any person who is not registered or logged in to the Travlendar+ service.

JEE: Java Enterprise Edition

Mobile application: Computer program designed to run on a mobile device such as smartphone or tablet.

OS: Operative system.

RASD: Requirements Analysis and Specification Document.

System:

User: Any person subscribed and logged in to the service who hence can insert a meeting using Travlendar+.

User Interface: It is the way through which a user interacts with an application or a website.

Web application: Client – server application accessible by an user through a browser.

1.5 Revision history

v1.0 Construct basic document's structure.

v1.1 Add *Purpose*, *Scope*, *Goal*, *Definitions, Acronyms, Abbreviations*, *Reference documents*, *Document structure.*

v1.2 Add *Product Perspective*, *Product Functions*, *User Characteristics*, *Assumptions and Dependencies*, *Constraints*, *World and Machine model interpretation.* Modify *Definition, Acronyms, Abbreviations* and document structure. Create a simple Appendix, that will be completed at the end.

v1.3 Add *External Interface Requirements.* Modify *Goal*, *Definitions, Acronyms, Abbreviations*, *Product Functions*, *Use Case Model*, *Assumption and Dependences*, *World and Machine model.*

1.6 Reference documents

This document is based on the specifications concerning the RASD assignment for the Software Engineering II project, part of the course held by professors Elisabetta Di Nitto and Matteo Giovanni Mottola at the Politecnico di Milano, A.Y. 2017/2018.

1.7 Document structure

This document consists of three sections:

Section 1: Introduction A general introduction and overview of the system-to-be purpose, scope and goals, along with some important information about this document.

Section 2: Overall description It describes the general factors that affects the product and its requirements. The section provides a background for those requirements which are defined in detail in Section 3 and makes them easier to figure out.

Section 3: Specific Requirements All the software requirements are specified to a level of detail which is sufficient to let the designers satisfy them. Both functional and non-functional requirements are mentioned.

There are two additional parts, Appendix and Bibliography that provide another information about the sections of this document.

Section 2

Overall Description

2.1 Product Perspective

2.1.1 User interfaces

The user have two main ways to access the system:

- through a web application accessible from any modern browser
- through a mobile application that can run on any modern smartphone

Although they are two different platforms, the user interface must be unified and intuitive, allowing anyone to use it without any training needed.

2.1.2 Hardware interfaces

The web application can be executed on any modern computer that meets the minimum system requirements (link al paragrafo).

The mobile application can run on any modern mobile device (i.e. smartphone, tablet) with mobile data connectivity, GPS and meets the minimum system requirements (link al paragrafo).

2.1.3 Software interfaces

The web application must support most of the modern browsers e.g., IE, Firefox, Chrome, Safari.

The mobile application must be supported by the most widespread mobile OSs such as iOS and Android that meets the minimum system requirements (link al paragrafo).

The backend application must rely on a commercial DBMS to store data and must be implemented in Java. The backend also have to interface with the APIs of a public transportation and traffic informations provider.***

2.2 Product Functions

The system allows the users to create meetings, helps them to reach the location.

The users can:

- register to the service;
- login in to the service;
- manage personal information and delete their accounts;
- create meetings;
- manage their preferences, such as activate/deactivate travel mean, provide constraints on travel mean, specify their intent to minimize his carbon footprint;
- specify lunch time.

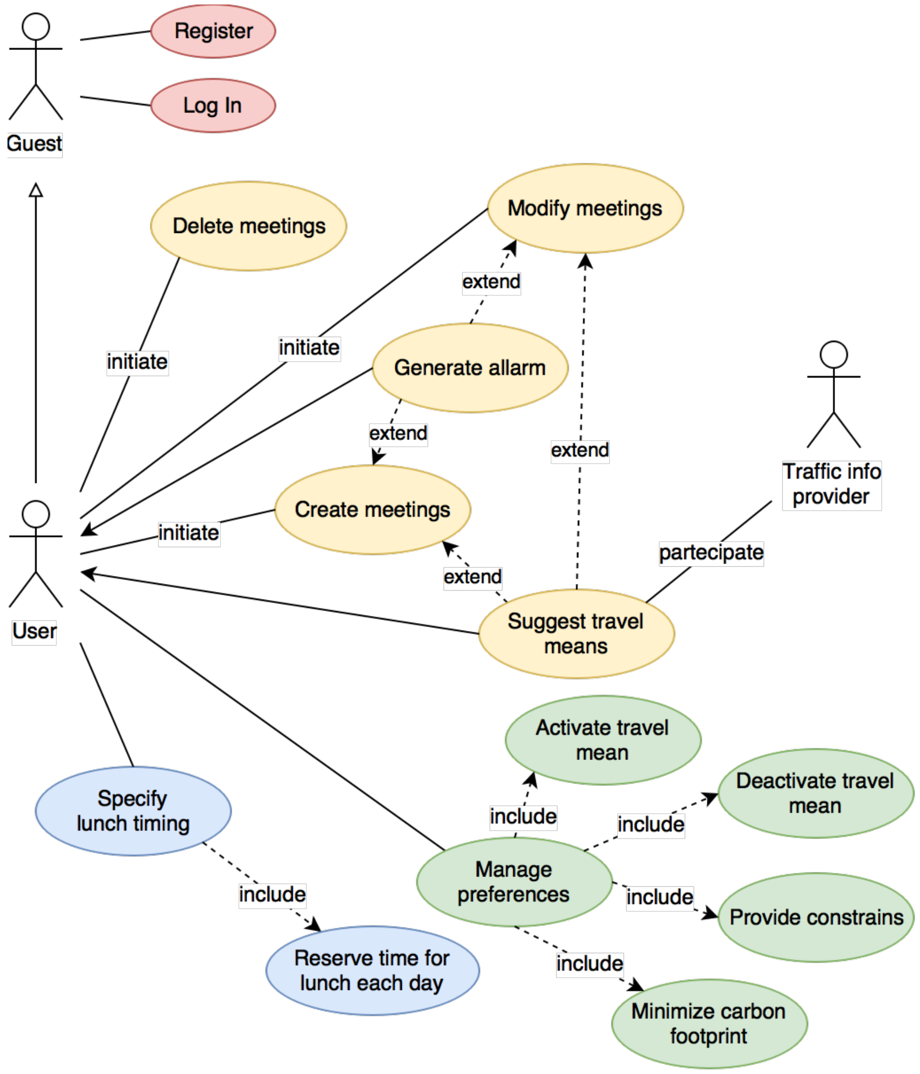

Figure 1: Use Case Model. Explains actors and functionalities offered by the system.

2.3 User Characteristics

The target user is someone who needs an automated system to manage appointments that also handle the scheduling of travels.

The meeting should not overlap with any pre-existing one and should allow the user to move from his/her current position to the meeting position, according with preferred travel means and traffic conditions.

2.4 Assumptions and Dependences

The following assumptions are given for granted:

- Transport means are complied with the user?s request.
- Device is always connect to the server.
- All users provided correct and valid data at time the registration.
- GPS shows the actual position of the owner.
- Provider Information shows correct and update data.
- The event, when it?s inserted, must not be in the past.

2.5 Constraints

2.5.1 Regulatory policies

The application must be allowed by the user to collect his/her position, through GPS.

2.5.2 Hardware limitations

- Web application:
  - Internet connection;
  - 800x600 screen resolution;
  - JavaScript enabled.
- Mobile application:
  - Internet connection;
  - 50 MB of available storage space;
  - 1GB of RAM;
  - GPS module.

2.5.3 Reliability requirements

The system reliability, that is the probability to operate without a failure for a specific period of time, must be at least 99%.

2.6 World and Machine model interpretation

In this part of RASD, a description of the system-to-be is provided following the World and Machine model introduced by Jackson and Zave.

They indicate as the Machine the portion of the system to be developed, typically software ? to ? be plus hardware. The Machine domain is the set of phenomena located entirely in the machine and that the machine control (e.g., machine algorithms, controlled device,?)

Opposite, the World domain is a set of phenomena that the machine cannot observe.

The World is connected with the Machine through Shared Phenomena ? part can be observable both by the Machine and by the World. The Shared Phenomena can be controlled by the world and observed by the Machine or controller by the machine and observe by the world.

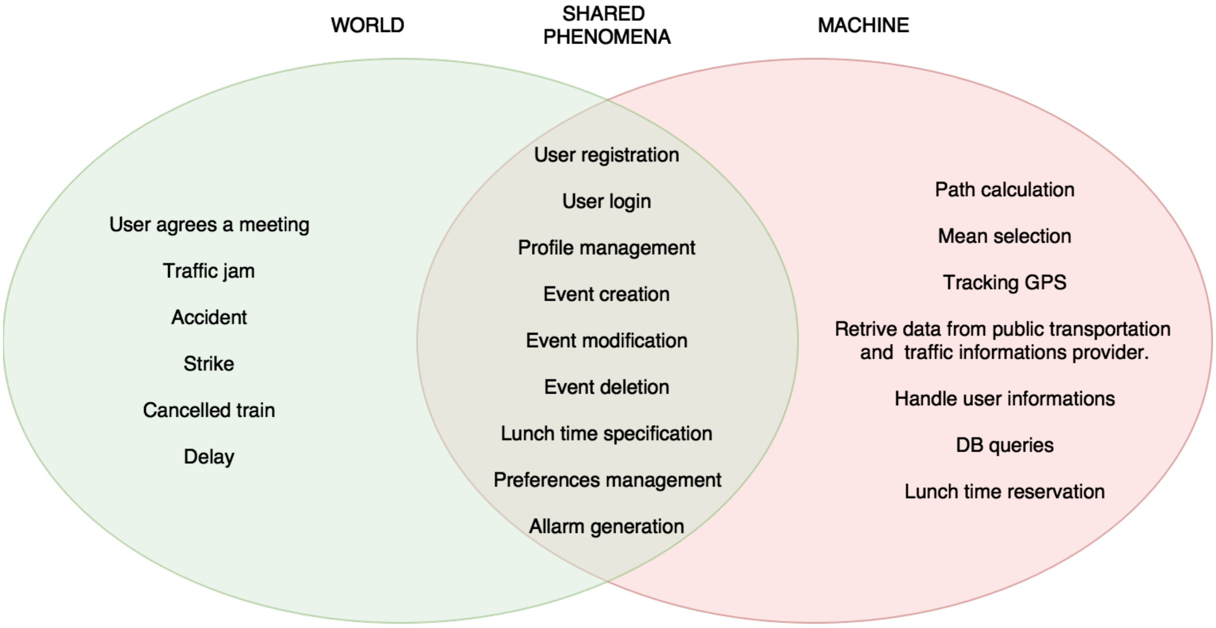

Figure 2: World and Machine model for the main functionalities offered by the system.

Section 3

Specific Requirements

3.1 External Interface Reqirements

3.1.1 User Interfaces

The user interface must be intuitive and unified, granting to the user a pleasant experience. In order to make it possible both web and mobile application must satisfy the following requirements:

- If the session is not already active the user must be redirected to a Login page. If he/she is not already registered he/she can create a new account within the same page. Moreover, the user must be able to Sign In/Sign Up not only with username and password but also with his/her social accounts (e.g. Google, Facebook, Twitter);
- Once logged, the user must be redirected to his/her personal page;
- A toolbar must allow the user to navigate through the pages described below;
- The user?s personal page must display an overview of user?s future events (display them in a list or in a calendar view) and must also offer the possibility to insert new ones;
- Clicking on an event the user must be redirected to the event?s detail page;
- The event?s detail page must display detailed information such as location of the event, time and suggested mean (including the path) needed to reach it;
- The event?s detail page must allow the user to modify or delete the event;
- If an user tries to inserted or modified an event which location is unreachable in the remaining time an alert should be displayed;
- The Account settings page must allow the user to modify its/her personal informations or delete the account;
- The user must be allowed to select between different languages (en, it, de, fr, es, ru, zh, ja, ar);
- The UI must comply the Flat Design principles;
- Both web and mobile applications must use the same graphic objects for the same interface elements.

Specific constraints must be satisfied by specific application:

- Web application:
  - The user interface must be responsive i.e. adapt to screen size;
  - All pages must comply W3C standards.
- Mobile application:
  - Must run on iOS 9.3 or greater and Android 5.0 or greater.

3.1.2 Hardware Interfaces

As already mentioned in section 2.1.2, the web application can be executed on any computer that meets the basic requirements described in the "Hardware Limitation" section.

The mobile application must exchange data with the GPS module located on any type of smartphone or tablet. You must also have an internet connection to communicate with the main system server.

3.1.3 Software Interfaces

The backend application requires the following software products:

- Java EE 7 - http://www.oracle.com/technetwork/java/javaee/overview/index.html
- MySQL 5.7 - http://dev.mysql.com/

As mentioned in Section 2.1.3, the backend must interfaced with the APIs of a public transportation and traffic informations provider, to have the information useful to plan the path.

The mobile applications requires the following software products:

- (iOS) Swift 4 - https://developer.apple.com/swift/
- (Android) Java SE 7 - http://www.oracle.com/technetwork/java/javase/overview/index.html

3.1.4 Communications Interfaces

Every communication between application server and client must comply the HTTPS protocol.

If the back-end application and the DBMS runs on different servers the communication between them must be SSL/TLS encrypted.

3.2 Functional requirements

3.3 Performance requiremens

3.4 Design constrains

3.4.1 Standards compliance

3.4.2 Hardware limitations

3.4.3 Any other constrain

3.5 Software System Attributes

3.5.1 Reliability

3.5.2 Availability

3.5.3 Security

3.5.4 Mantainability

3.5.5 Portability

Section 4

Formal Analysis Using Alloy

Section 5

Effort Spent

Section 6

References