Travlendar+ project

Requirement Analysis and Specification Document

RICCARDO FACCHINI

ANDREA GUGLIELMETTI

October 9, 2017

Deliverable specific information

Deliverable:	RASD
Title:	Requirement Analysis and Verification Document
Authors:	Riccardo Facchini - Andrea Guglielmetti
Version:	0.1
Date:	October 9, 2017
Download page:	https://github.com/Riccardo95Facchini/FacchiniGuglielmetti.git
Copyright:	Copyright © 2017, Riccardo Facchini - Andrea Guglielmetti – All rights reserved

Contents

List of Figures

List of Tables

1   Introduction

1.1   Purpose

This Requirement Analysis and Specification Document (RASD for short) document has the purpose of fully describing to a wide range of potential readers the system and to function as a base for legal agreements between developers and other parties.

In the following pages there will be precise descriptions of all the functional and non-functional requirements, the different scenarios and cases of interaction between the system and the users, with attention to what the users need from it, the domain of the system and the constraints that it implies.

The readers of this document comprise the developers of the system and its applications and agents from the local public transportation agencies or independent company in similar professions such as taxi businesses or bike/car sharing companies.

1.2   Scope

The scope of this project is to develop a system called Travlendar+ that will allow in the most possible efficient way the paring of daily commutes and the management of scheduled appointments and meetings, by providing for each situation the best alternatives of moving throughout the city both for work related reasons and for personal motives.

Users of Travlendar+ can create a calendar with every appointment paired with time and place, the system will then compute the best way of reaching each location in time by choosing between every commuting option available at the moment and taking into account the preferences expressed by the user in the customization settings, possible strikes of the local transportation services and the weather, if the location is too far and cannot be reached in time a warning is going to pop up on the screen of the user.

Each time a scheduled appointment is coming up a notification is going to appear in advance by a configurable amount of time, the user will then be able to confirm, change or refuse the proposed trip.

1.3   Stakeholders

Here are listed all the potential stakeholders with a brief description and how they may be affected by the system:

- **User**: All the individuals that will use the system to schedule their daily commute.

- **Public transportation companies**: Local and international companies that handle public transportation may have an advantage in giving an easy way to integrate their schedules with Travlendar+ as it could mean a higher number of sold tickets.

- **Taxi and Car/Bike sharing companies**: Given that is not always possible for each type of user to walk long distances and public transport does not reach every possible destination they may be interested in a partnership between their service and the system.

- **Mobile network carriers**: They have an interest in providing a network and contracts to connect devices to the service.

1.4   Definitions, acronyms and abbreviations

What follows is the list of all the main terms and abbreviations used in the document.

1.4.1   Definitions

- **User**: who is using the system to schedule their calendar.

- **Trip**: the plan to move from point A to point B done using one or more means.

- **Travel**: synonymous of trip.

- **System**: All the software needed to deliver all the functionalities desired, often used as a synonymous of Travlendar+.

- **RASD**: Requirement Analysis and Specification Document

- **SRS**: Software Requirement Specification. Synonymous of RASD.

- **ETA**: Estimated Time of Arrival.

- **GPS**: Global Positioning System

1.4.2   Abbreviations

No other abbreviations were used.

1.4.3   Revision History

- Introduction.

1.5   Reference documents

- Document of the assignment: Mandatory Project Assignments.pdf

2   Product Perspective

The system will have three main parts:

1. Mobile application version for phones and tablets.

2. Web browser version.

3. Backend structure to support the functioning of the service.

3   Specific Requirements

3.1   External Requirements

3.1.1   User Interfaces

The user interfaces must be user-friendly, to provide an easy access to the system functionalities.
To perform this the followings constrains must be satisfied:

- The first page must always ask the user to login or register to service.

- After the login, the system redirects the user to his/her home page.

- (Web) A toolbar must be present in every page, except login and registration page.

- (Mobile) A toolbar must be present in the homepage.

- The *Create a meeting* page must provide a guided process to set-up a meeting and clearly show if the created meeting is not reachable in time from the location of a previous appointment.

- The *Manage meetings* page must show a list of user's meetings divided by day and hour, and allows the user to choose a meeting to obtain further information.

- The interface must offer the possibility to choose between a set of different languages.

- The user interface must dynamically adapt to the screen size.

- The Mobile and Web application must use the same graphical elements.

In addition to these constraints, other platform-dependent constraints are provided:

- Web Application

  All the pages must submit to W3C standards.

- Mobile Application

  All mobile versions must follow the design guidelines provided by the respective platform manufacturer (Android, iOS, Windows . . .).

3.1.2   Hardware Interfaces

The system must be able to communicate with the GPS module of the smartphone to track the user's movements.

3.1.3   Software Interfaces

The server side of the application requires:

- Java EE, to write the server application that perform the travel computation and the database access.

- MySQL, to memorize user information and meetings.

The client side of the application requires the latest version of the platform SDK.

3.1.4   Communication Interfaces

The client communicates to the server via HTTPS protocol using TCP.
In addition, the system must be able to use the API of other application in order to retrieves weather and news about road conditions or strike.