POLITECNICO
MILANO 1863

Travlendar+ project

Design Document

RICCARDO FACCHINI

ANDREA GUGLIELMETTI

November 10, 2017

Deliverable specific information

Deliverable:	Design Document
Title:	Requirement Analysis and Verification Document
Authors:	Riccardo Facchini - Andrea Guglielmetti
Version:	1.0
Date:	November 10, 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 document aims to detail the design of the software and of the architecture regarding the system of Travlendar+. To do so it will be taken a more detailed approach for the description of each component and the overall architecture of the system, by also pointing out the relations between each module and giving a description of such relationships.

1.2 Scope

Travlendar+ is a time/travel management web-based system, designed to help the users to keep track of their daily routine by scheduling for them the best way to move from one place to the other using all the information given by the users themselves and external data in order to deliver a tailored experience for everyone.
After the user enters all the needed data to register an event, the system will automatically alert him/her when it's time to leave and will give directions to reach each one of the means of travel as specified in the options, taking into account also factors like the weather and possible public transportation strikes.

1.3 Definitions, Acronyms, Abbreviation

- DD : Design Document
- RASD: Requirement Analysis and Specification Document
- API: Application Programming Interface
- DB: DataBase
- DBMS: DataBase Management System
- GPS: Global Position System

1.4 Revision History

1.5 Reference Documents

- Document of the assignment: Mandatory Project Assignments.pdf
- Requirements and Specification Document

1.6 Document Structure

2 Architectural Design

2.1 Overview

We need to design a system in which the user asks to the system to store an appointment and calculate the best path from a starting location to the appointment location.
Since this interaction between user and system can be summarize as:

1. User request a service to the system.
2. System responds to the user with the requested service.

Based on this, we decide to use a client-server architectural approach.

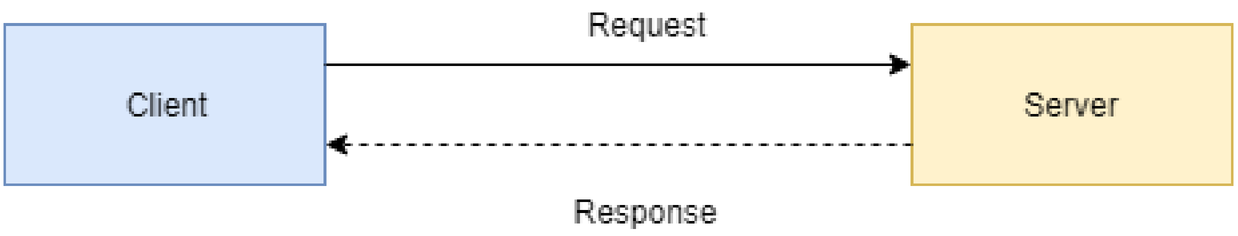

Figure 1: Client Server architecture

Furthermore, the system can be divided into three different subsystems: the presentation layer, the application layer and the data layer as we can see in Figure 2.

- The *Presentation Layer* provides the GUI of the system. This layer contains the mobile application and the web pages.
- The *Application Layer* contains the logic of the application,that receives the requests from the user, computes the best path to reach the appointment, checks the weather and the road conditions and executes the dynamic web pages of the web site.
- The *Data Layer* stores and maintains the data needed from the system to works properly, i.e. user's information and user's appointment information.

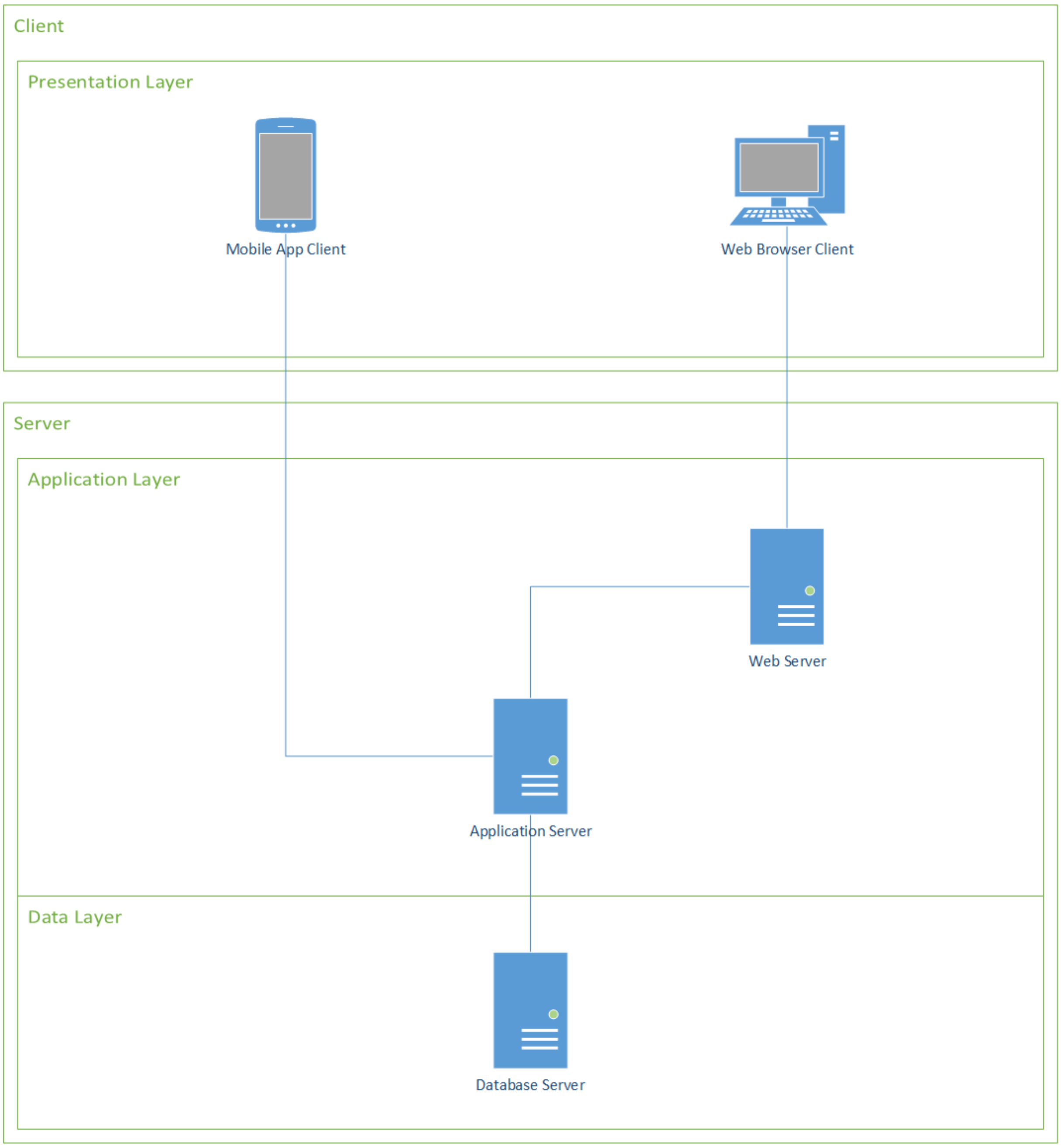

Figure 2: Overview of the system architecture

2.2 Component View

2.3 Deployment View

2.4 Runtime View

2.5 Component Interfaces

2.6 Selected architectural styles and patterns

2.7 Other design decision

3 Algorithm Design

Once the system has the couple time-place for both the departure and the arrival, it must take into account all the preferences of the user, weather conditions and public transportation strikes in order to formulate the best possible query that will be forwarded and handled by Google Maps.

4 User Interface Design

4.1 UserInterfaces

5 Requirements Traceability

5.1 requirements traceability

6 Implementation, Integration and Test Plan

6.1 implementation

7 Appendix

7.1 Effort Spent

7.2 References