



**POLITECNICO**  
MILANO 1863

# **Travlendar+ project**

## **Design Document**

RICCARDO FACCHINI

ANDREA GUGLIELMETTI

October 26, 2017

## Deliverable specific information

---

<b>Deliverable:</b>	Design Document
<b>Title:</b>	Requirement Analysis and Verification Document
<b>Authors:</b>	Riccardo Facchini - Andrea Guglielmetti
<b>Version:</b>	1.0
<b>Date:</b>	October 26, 2017
<b>Download page:</b>	<a href="https://github.com/Riccardo95Facchini/FacchiniGuglielmetti.git">https://github.com/Riccardo95Facchini/FacchiniGuglielmetti.git</a>
<b>Copyright:</b>	Copyright © 2017, Riccardo Facchini - Andrea Guglielmetti – All rights reserved

---

## Contents

<b>Deliverable specific information</b>	<b>1</b>
<b>Table of Contents</b>	<b>2</b>
<b>List of Figures</b>	<b>3</b>
<b>List of Tables</b>	<b>4</b>
<b>1 Introduction</b>	<b>5</b>
1.1 Purpose	5
1.2 Scope	5
1.3 Definitions, Acronyms, Abbreviation	5
1.4 Revision History	5
1.5 Reference Documents	5
1.6 Document Structure	5
<b>2 Architectural Design</b>	<b>6</b>
2.1 Overview	6
2.2 Component View	6
2.3 Deployment View	6
2.4 Runtime View	6
2.5 Component Interfaces	6
2.6 Selected architectural styles and patterns	6
2.7 Other design decision	6
<b>3 Algorithm Design</b>	<b>8</b>
<b>4 User Interface Design</b>	<b>9</b>
4.1 UserInterfaces	9
<b>5 Requirements Traceability</b>	<b>10</b>
5.1 requirements traceability	10
<b>6 Implementation, Integration and Test Plan</b>	<b>11</b>
6.1 implementation	11
<b>7 Appendix</b>	<b>12</b>
7.1 Effort Spent	12
7.2 References	12

## List of Figures

1	Client Server architecture . . . . .	6
2	Overview of the system architecture . . . . .	7

## List of Tables

## **1 Introduction**

### **1.1 Purpose**

### **1.2 Scope**

### **1.3 Definitions, Acronyms, Abbreviation**

### **1.4 Revision History**

### **1.5 Reference Documents**

### **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.

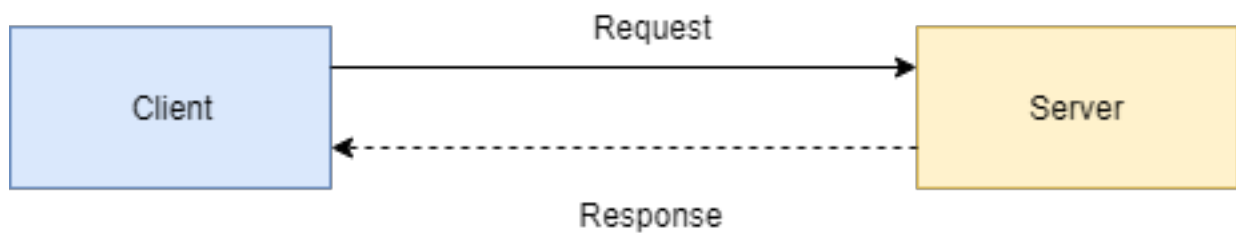


Figure 1: Client Server architecture

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](#). Additional information are in [subsection 2.6](#).

### 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

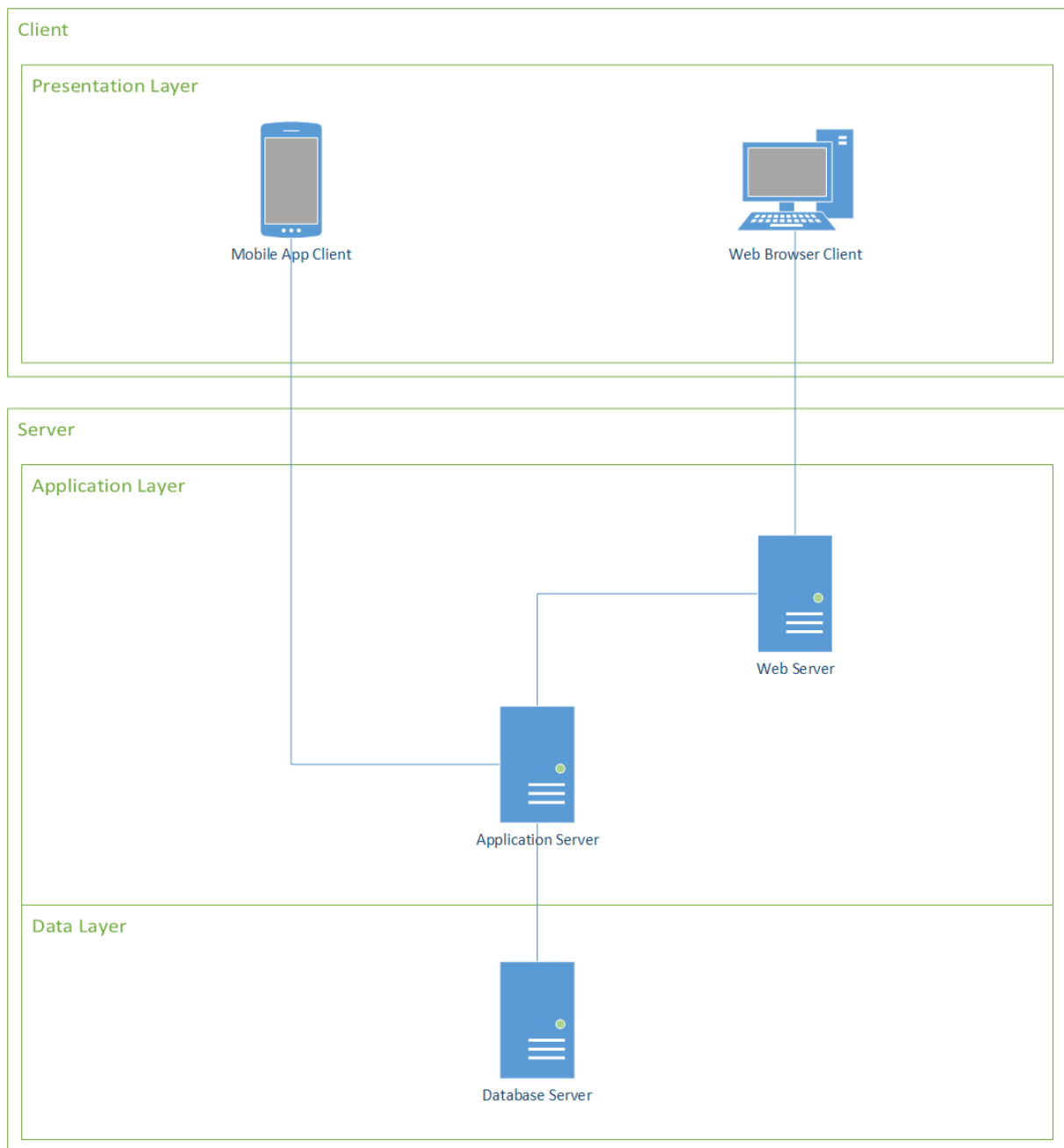


Figure 2: Overview of the system architecture



## 3 Algorithm Design

### Algorithm

## **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**