

Politecnico di Milano

A.Y. 2017/2018

Software Engineering 2: ***Travlendar+***

**Design Document**

Matteo Biasielli - Emilio Capo - Mattia Di Fatta

v. 0.1

**Table of Contents**

**1. Introduction**

1.1. Document purpose..………………………………………………...………………….3

1.2. Definitions, Acronyms, Abbreviations………………………………………….3

1.2.1 Definitions……………………………………………………………………….3

1.2.2 Acronyms………………………………………………………………………..3

1.2.3 Abbreviations…………………………………………………………………..3

1.3. Reference Documents…………………………………………………………………4

1.4. Document Structure……………………………………………………………………5

1.5. Revision History…………………………………………………………………………5

**2. Architectural Design**

**3. Algorithm Design**

**4. User Interface Design**

**5. Requirements Traceability**

**6. Implementation, integration and test plan**

**7. Effort Spent**

**1. Introduction**

* 1. **Document purpose**

This document has to be intended as a general guide for the correct development of the Travlendar+ application. The content of this document follows and is based on the content of the RASD document. This document is meant to be a reference for any person who has an interest in the project. This includes, but is not limited to, development team members, stakeholders and end users.

* 1. **Definitions, Acronyms, Abbreviations**
     1. **Definitions**
* **User**: actor that is using the application and may want to access all functionalities.
* **Application**: with the term application we are talking about the desktop version, the website and mobile version of the Travlendar+ system.
* **Scheduling**: action performed by a user that is adding a new activity to his personal calendar.
* **Flexible Activity**: An activity with starting and ending time larger than the duration.
* **Fixed Activity**: An activity with fixed starting and ending time.
  + 1. **Acronyms**
* **RASD:** Requirements Analysis and Specification Document
* **DD:** Design Document
* **UI:** User Interface
* **API:** [Application programming interface](https://en.wikipedia.org/wiki/Application_programming_interface)
* **UXD**: User Experience Diagram
* **UML**: Unified Modeling Language
* **GPS**: Global Positioning System
  + 1. **Abbreviations**
* **[Gn]:** the n-th goal
* **[Rn]:** the n-th requirement
* **[NFRn]:** the n-th non-functional requirement
* **[An]:** the n-th assumption
* **[Cn]:** the n-th constraint
* **[UIn**]: the n-th user interface example
  1. **Reference Documents**
* Mandatory project assignments for the A.Y. 2017/2018 available on the beep’s page of the Software Engineering 2 course.
* Projects examples and other documents available on the beep’s page of the Software Engineering 2 course.
  1. **Document Structure**
* **Introduction:** This is the very first part of the document.

In this section it’s possible to retrieve general information about the Design Document. The purpose and intended audience of the document are specified here.

In addition, Acronyms, Definitions and Abbreviations are defined in this section in order to make it easier, more concise and clearer to read the rest of the Design Document.

* **Architectural Design:** This part represents the second chapter of the document. Here the reader can find the architecture of the system components at various levels and contexts.

First of all, a high-level overview of the components and the way they’re connected is provided in this section. Following this, some components will be analysed in detail and their internal architectures will be showed for a matter of clarity.

* **Algorithm Design:** The most important algorithms that will be implemented in our application are described here, both with natural language and with java code/pseudocode.
* **User Interface Design:** Some User Interface samples have already been provided in the RASD document but they’ll be extended and some will be added in this section of the Design Document.

In addition, further explanation about the already existing UI will be added here, together with a detailed mapping of the User Interfaces into functional requirements and non-functional requirements.

* **Requirements Traceability:** Design choices are mapped into functional and non-functional requirements here.
  1. **Revision History**
* **v. 0.1 [28 Oct 2017]**: added the whole “Introduction” section.

**2 Architectural Design**

In this section, we first provide a general overview on Travlendar+

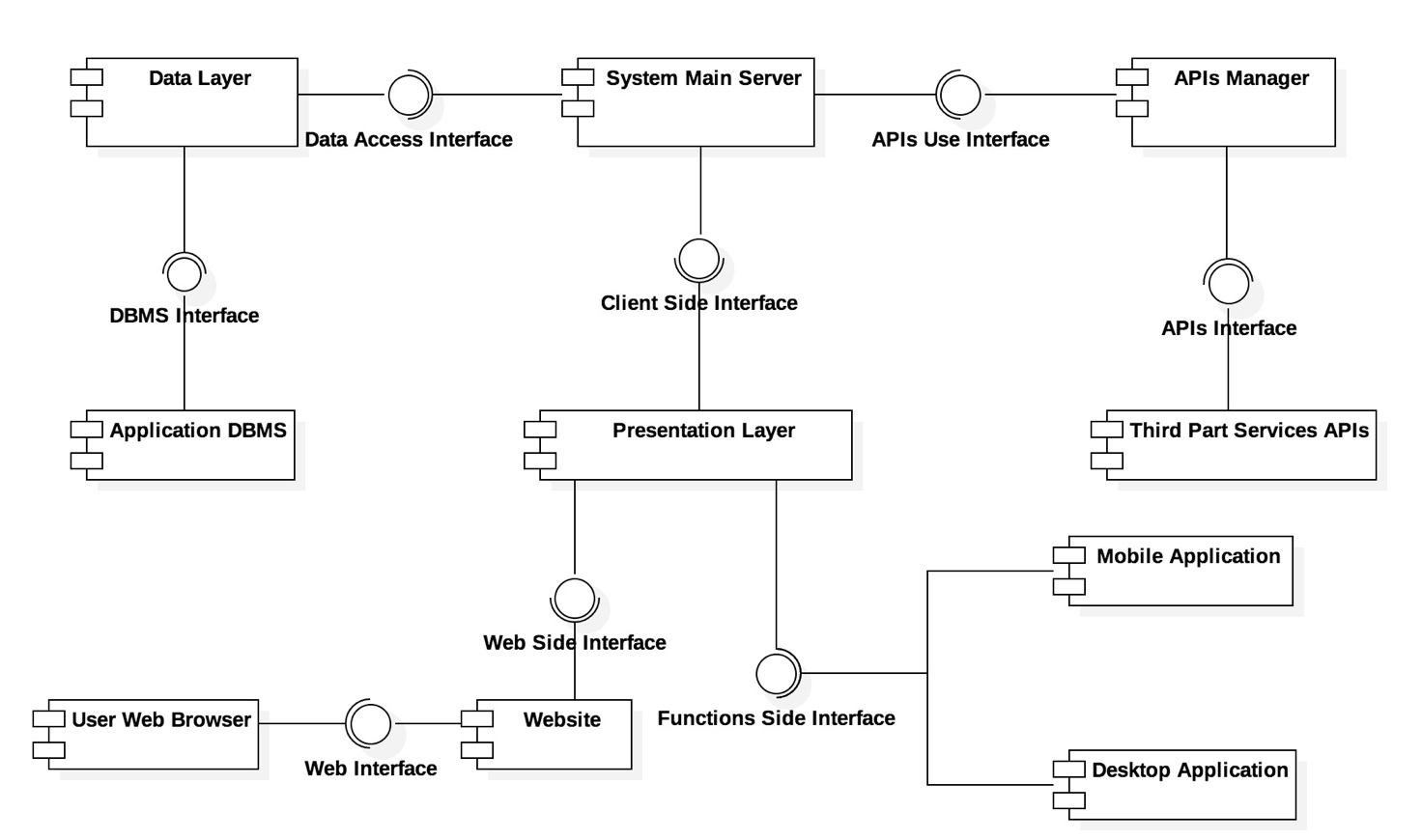
system by means of a general Component Diagram. Then it is given a Component View on some important components again by means of Component Diagrams.

Moreover, a Deployment View and a Runtime View of the system is provided in order to describe nodes with their components, protocol for their interaction and how they are expected to work (by means of Sequence Diagram).

Eventually, we describe in detail all components’ interfaces that can be found in the general Component Diagram.

**2.1 General Overview on System’s Components**

The following diagram describes the whole Travlendar+ system with all its components (i.e. software modules).



***Data Layer***

This component deals directly with system’s DBMSs allowing other components to write, read and update data in the DBs properly (i.e. in a secure, consistent way) using the proper DBMS. It provides a single interface outward in order to systematize access to data stored in DBs and hide the internal complexity and implementation of DBMSs and DBs.

***Application DBMSs***

This particular component is used as generalization of the two DBMSs used in Travlendar+: one to manage users’ personal data and calendar, one to manage all relevant internal information used by Travlendar’s server to works properly (for further information see Component View section).

It provides an appropriate interface to allow the Data Layer component, and only it, to query DBMSs.

***System Main Server***

This is the core component of the server side of the system and of the application in general. It’s composed by three essential components: the Computation Unit, the User Side Unit and the Notification System.

It manages, with the aid of the APIs Manager and the Data Layer, all the Travlendar’s functionalities described in the RASD document, with the exception of the presentation side, managed entirely by the Presentation Layer (see below for further information).

***APIs Manager***

**7. Effort Spent**

This section will provide detailed information about the number of hours spent on this document.

**Matteo Biasielli**, matr. 893590

|  |  |
| --- | --- |
| **Section(s)** | **Number of hours** |
| 28-oct-17 Introduction | 1 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **TOTAL:** |  |

**Mattia Di Fatta,** matr. 893608

|  |  |
| --- | --- |
| 28-oct-17 | 1 |
| 29-oct-17 | 2.5 |
| 30-oct-17 | 2 |
| 31-oct-17 | 2 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **TOTAL:** | 7.5 |

**Emilio Capo,** matr. 899842

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| **TOTAL:** |  |