Implementation Plan

# Introduction

This document aims to present the implementation plan of the web application according to the architecture proposed and detailed in the specifications.

# Plan

The Google APIs (Calendar and Authentication) are not shown as they will not be implemented in this project but just used by several components. However, it will be important to be sure that these APIs are ready to use before to use them, and that the technologies which will be chosen allow to use these APIs. The technology are also chosen according to my knowledge and what is the best to do web application at the moment. About the priority, 1 equal “to do first” and 2 equal “to do once one is finished” but there is not particularly a hierarchy in the development of these components (even if it is better to set the connection before to work on the booking of a room).

|  |  |  |
| --- | --- | --- |
| Name of the component | Description | Implementation Plan |
| API | The API will have to support the requests made by the front-side and will have to send information to the front-side. This API will also have to call the functions of the libraries. | **Technologies**: Express and Node.js  **Priority**: 2 |
| Server | The server will have to support the request from the front-side through the API. | **Technologies**: Express and Node.js  **Priority**: 1 |
| Booking | This library will have to send information to the database to store it or to request the database to obtain information; will have to send information to the Google Calendar through the G.C.API or to request the G.C.API to obtain information (especially for the verification). | **Technology**: JavaScript  **Priority**: 2 |
| IoT Access | This library will have to send information to the database to store it or to request the database to obtain information. | **Technology**: JavaScript  **Priority**: 2 |
| Occupancy, Status and Calendar | This library will have to send information to the database to store it or to request the database to obtain information; will have to send information to the Google Calendar through the G.C.API or to request the G.C.API to obtain information. Know the position with the reader can be more difficult than the other things, then I will finish by this task in the project. | **Technology**: JavaScript  **Priority**: 2 |
| Database and Make the connection | This database will have to support the requests from different sources. This database do not need to be very complex as there are few information to store and these information do not need a particular schema. However, depending of tables and their importance, some tables can be fuelled later in the project. | **Technologies**: MySQL and JavaScript (to make the connection)  **Priority**: 1 |
| Web Application | The web application will have to make HTTP request to the server-side (API) and will have to receive information from this API. It will also display the information to the user. | **Technologies**: HTML/CSS and Angular or Angular.JS  **Priority**: 2 |