Airplane Checkup

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 19/03/2020 | 1.0 | Inception | Rus Tudor |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

2. Non-functional Requirements 4

2.1 Availability 4

2.2 Performance 4

2.3 Security 4

2.4 Testability 4

2.5 Usability 4

3. Design Constraints 4

# Introduction

The application must be a web application. The environment must be Java and Maven capable for the backend and Spring Boot will be used. The RESTful Api will be used for frontend-backend communication. The frontend will be done in VS Code using Vue.js. MySQL Workbench will be used for creating and storing persistent data. The system must be available, usable, secure, performant and testable.

# Non-functional Requirements

## Availability

The application requires an Internet connection.

## Performance

The application should sustain multiple users and simultaneously operations without lag.

## Security

The user accounts and their contents must be stored securely in a database. The accounts are password protected. The security is very important taking into account that human lives are at stake.

## Testability

The application must be testable using Unit Tests for both the frontend and the backend.

## Usability

The application’s interface must be user-friendly and as simple as possible so there is no confusion.

# Design Constraints

* The software languages must be Java, HTML, CSS, Javascript, SQL
* The development tools must be IntelliJ, Maven Project, RESTful Api, MySQL Workbench, VS Code, Vue.js, Spring Boot
* The system architecture must follow the Layered Pattern and the Domain Driven Design style
* The system must include at least three design patterns