<Company Name>
30432

MedPat: a Doctor-Patient platform
Supplementary Specification

Version <1.0>

Revision History

Date	Version	Description	Author
20/03/2018	1.0	First rendition.	Turcu Lucian Andrei

Table of Contents

Supplementary Specification

1. Introduction

*[The introduction of the **Supplementary Specification** provides an overview of the entire document.*

*The **Supplementary Specification** captures the system requirements that are not readily captured in the use cases of the use-case model. Such requirements include:*

Legal and regulatory requirements, including application standards.

Quality attributes of the system to be built, including usability, reliability, performance, and supportability requirements.

Other requirements such as operating systems and environments, compatibility requirements, and design constraints.]

2. Non-functional Requirements

[Define system quality attributes in terms of scenarios according to the following template:
- *Quality attribute definition*
- *Source of stimulus: the entity (human or another system) that generated the stimulus or event*
- *Stimulus: a condition that determines a reaction of the system*
- *Environment: the current condition of the system when the stimulus arrives*
- *Artifact: is a component that reacts to the stimulus. It may be the whole system or some pieces of it*
- *Response: the activity determined by the arrival of the stimulus*
- *Response measure: the quantifiable indication of the response*
- *Tactics*

]

2.1 Availability

The system will be available 24/7 to both the patients and the users. However, it is up to the doctor to manage his own schedule in which he or she is available to the patients.

2.2 Performance

a) Simultaneous users
The number of users that the system supports at any given time highly depends on the hardware that the server is installed on and the provided internet connection. It should support around 500 users without any considerable hardware requirements.

b) Database access response time
The system shall provide access to the database with no more than 10 seconds latency.

c) Transaction response time
The system must be able to complete all transactions in 2 minutes.

2.3 Security

All user passwords will be encrypted using the Spring Security 5 modules, which are primarily focused on OAuth2.

2.4 Testability

The system will be tested using Junit tests and manual tests on a local database with dummy accounts.

2.5 Usability

a) Android compliance
The application will comply to the Android design standards, including the Material UI design, to be in line with the current trends.

b) Ease-of-Use
The application will be designed to be easily used by a basic smartphone user, by having clearly named items and descriptions where necessary.

3. Design Constraints

The main programming language used is Java and the application will be based on Spring Boot to provide a platform for designing a reliable client-server architecture. The database engine will be SQL. Dependencies will be managed using Maven.