

# Software Requirements Specification

Version 1.0

<<Annotated Version>>

27/5/2021

Doctor Bookings Manager

Mohamad Belal Al-Shmalya

Anas Al-bokaie

Nawras Audi

Mohamad Yasser Raaed

<<Any comments inside double brackets such as these are *not* part of this SRS but are comments upon this SRS example to help the reader understand the point being made.

## Table of Contents

Table of Contents .....	3
1.0. Introduction .....	4
1.1. Purpose : .....	4
1.2. Scope of Project .....	4
1.3. Glossary .....	5
1.4. References .....	5
1.5. Overview of Document .....	5
2.0. Overall Description .....	6
2.1 System Environment .....	6
2.2 Functional Requirements Specification .....	7
2.2.1 Patient Use Case .....	7
2.2.2 Patient Use Case .....	8
2.2.3 Doctor Use Case .....	9
2.2.4 Doctor Use Case .....	10
2.3 User Characteristics .....	خطأ! الإشارة المرجعية غير معروفة.
2.4 Non-Functional Requirements .....	11
3.0. Requirements Specification .....	12
3.1 External Interface Requirements .....	خطأ! الإشارة المرجعية غير معروفة.
3.2 Functional Requirements .....	12
3.2.1 Book an appointment .....	12
3.2.2 Edit Profile .....	12
3.2.3 Approve/Reject a booking .....	12
3.2.4 Block Patient .....	13

## **1.0. Introduction**

### **1.1. Purpose :**

The purpose of this document is to present a detailed description of the Patient Booking System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

### **1.2. Scope of Project**

This software system will be a Patient Booking Appointments for a local clinic. This system will be designed to help the doctor and patients automating the booking process, which would otherwise have to be performed manually when the patient visits the doctor. The system will meet the doctor's and patient's needs while remaining easy to understand and use.

More specifically, this system is designed to allow a doctor to manage and communicate with a group of patients to manage their bookings to a public website. The software will facilitate communication between doctors and patients via the website. The system also contains a relational database containing a list of patients and their bookings.

### 1.3. Glossary

Term	Definition
Doctor	Person manage the appointments and patient accounts
Database	Collection of all the information monitored by this system.
Patient	Person who make appointmets
Field	A cell within a form.
Historical Society Database	The existing membership database (also HS database).
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
Stakeholder	Any person with an interest in the project who is not a developer.

### 1.4. References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications*.

IEEE Computer Society, 1998.

### 1.5. Overview of Document

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

## 2.0. Overall Description

### 2.1 System Environment

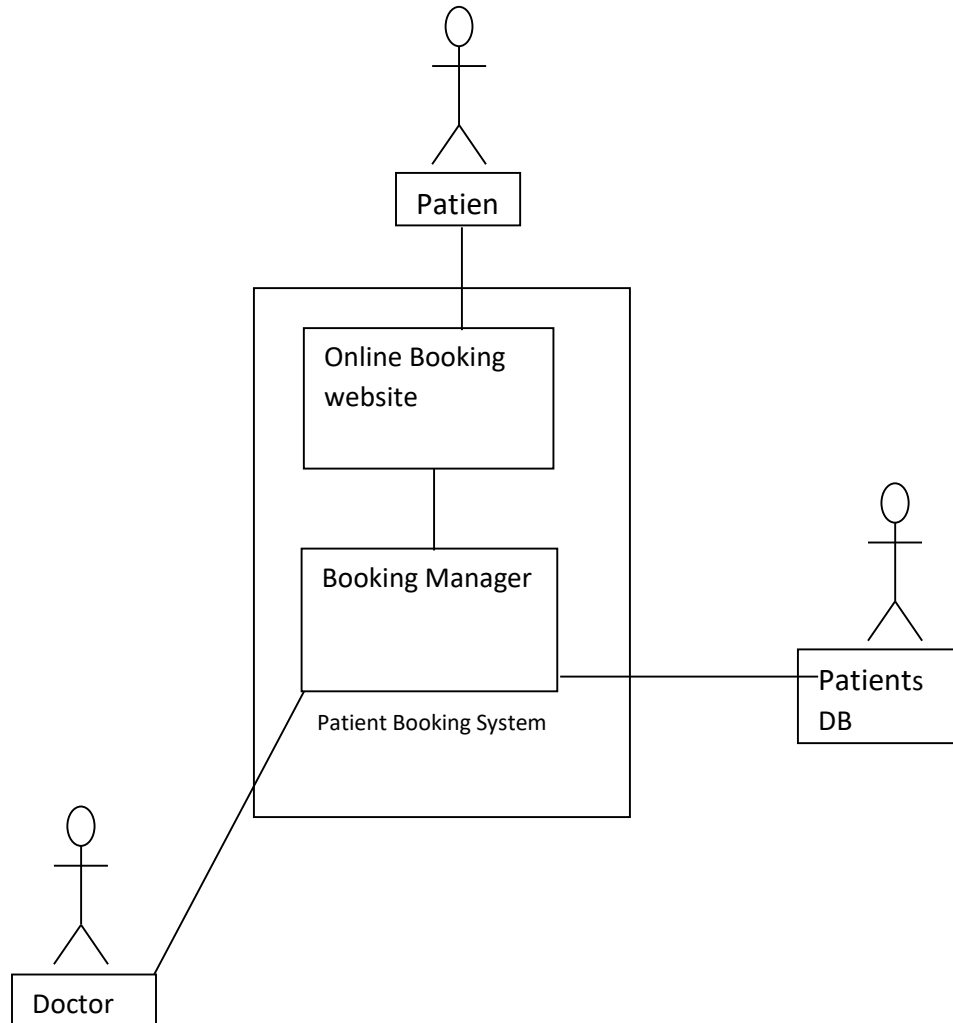


Figure 1 - System Environment

The Patient Booking System has two active actors and one cooperating system.

The Patient accesses the Online Booking System through the Internet. The Doctor accesses the entire system directly.

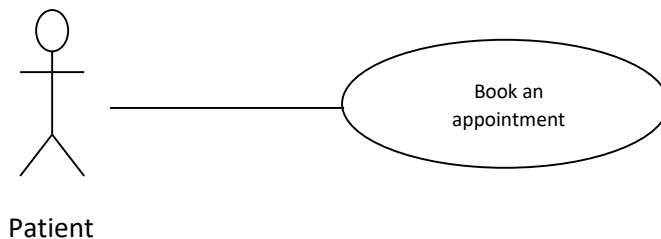
## 2.2 Functional Requirements Specification

This section outlines the use cases for each of the patients separately. The patient has only one use case apiece while the doctor is main actor in this system.

### 2.2.1 Patient Use Case

*Use case: Book an appointment*

**Diagram:**



#### **Brief Description**

The Patient accesses the Online Booking Website, searches for an appointment and book it.

#### **Initial Step-By-Step Description**

Before this use case can be initiated, the Patient has already accessed the Online Booking Website and made an account on it.

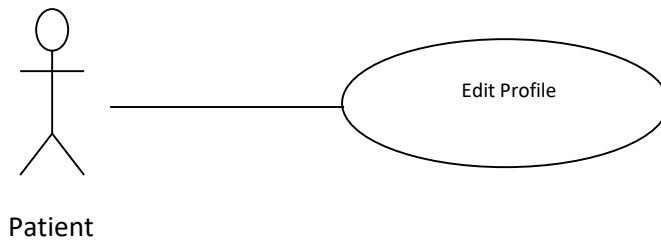
1. The Patient chooses to search by date.
2. The system displays the choices to the Patient.
3. The Patient selects the booking date desired.
4. The system provides the requested booking and book it, After that , The doctor will approve or reject this book.

**Xref:** Section 3.2.1, Book an appointment.

### 2.2.2 Patient Use Case

*Use case: Edit his Profile*

**Diagram:**



#### **Brief Description**

The Patient accesses the Online Booking Website and edit his personal information like:

Name, email, password, phone number and photo.

#### **Initial Step-By-Step Description**

Before this use case can be initiated, the Patient has already accessed the Online Booking Website and login for his account.

1. The Patient update his information that he/she need to edit.
2. The system displays message that tell the patient if updated was success or not.

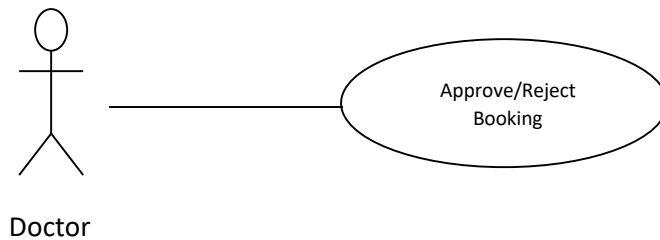
**Xref:** Section 3.2.2, Edit Profile



### 2.2.3 Doctor Use Case

*Use case: Approve/Reject a booking*

**Diagram:**



#### **Brief Description**

The Doctor accesses the booking manager website and approve/reject the booking.

#### **Initial Step-By-Step Description**

Before this use case can be initiated, the Doctor already login to his account.

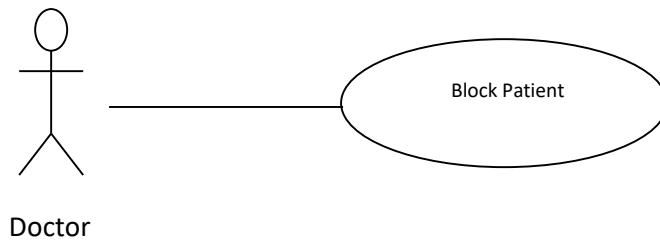
1. The Doctor must be browse all appointments that need to approve/reject.
2. The Doctor chose an appointment that he want.
3. The Doctor approve or reject it.

**Xref:** Section 3.2.3, Search Article

## 2.2.4 Doctor Use Case

### *Use case: Block Patient*

#### **Diagram:**



#### **Brief Description**

The Doctor accesses the booking manager website and block patient that make fake appointments.

#### **Initial Step-By-Step Description**

Before this use case can be initiated, the Doctor already login to his account.

1. The Doctor select the patient that make fake appointment.
2. The Doctor block this account.

**Xref:** Section 3.2.4, Block Patient

## 2.3 Non-Functional Requirements

The Online Booking Website will be on a server with high speed Internet capability. The physical machine to be used will be determined by the patients DB. The software developed here assumes the use of a tool such as Tomcat for connection between the Web pages and the database.

The Booking Manager will run on the doctor's PC and will contain an Access database. Access is already installed on this computer and is a Windows operating system.

### 3.0. Requirements Specification

#### 3.1 Functional Requirements

##### 3.1.1 Book an appointment

<b>Use Case Name</b>	Book an appointment
<b>XRef</b>	Section 2.2.1, Book an appointment
<b>Trigger</b>	The patient assesses the Online Booking Website
<b>Precondition</b>	The Web is displayed the available bookings.

##### 3.1.2 Edit Profile

<b>Use Case Name</b>	Edit Profile
<b>XRef</b>	Section 2.2.2, Edit Profile
<b>Trigger</b>	The Patient update his information or reviewer and the Patient is already in the database.
<b>Precondition</b>	The Patient has accessed the Personal Information Manager.

##### 3.1.3 Approve/Reject a booking

<b>Use Case Name</b>	Approve/Reject a booking
<b>XRef</b>	Section 2.2.3, Approve/Reject a booking
<b>Trigger</b>	The Doctor selects the booking to update the status of it in the database.
<b>Precondition</b>	The Doctor has accessed the Booking Manager main screen and the booking is already in the database.

#### 3.1.4 Block Patient

<b>Use Case Name</b>	Block Patient
<b>XRef</b>	Section 2.2.4, Block Patient
<b>Trigger</b>	The Doctor selects the patient to block the patient in the database.
<b>Precondition</b>	The Doctor has accessed the Patient Manager main screen and the Patient is already in the database.