SOFTWARE REQUIREMENTS ANALYSIS

for

E-APPOINTMENT

E-Prescription

Version 1.0 approved

Prepared by

Masum Ahmed Esha

Mushtaq Shahriyar Rafee

Shamihul Islam Khan Limon

Partho Sarker

SUST_1729 Inc.

13 April 2019

Table of Contents

1

Introduction

1.1 Purpose

The purpose of this Software Requirement Specification (SRS) document is to gather information about the E-Appoinment system.

E-Appoinment system is an android based application that is created for redusing time between the doctors and the patients.Normally the existing appointment system of the doctors portrays scenery of an ill person waiting for hours and hourse just to get a glimpse of doctors face.In this application we wanted to reduse this gap.

1.3 Intended Audience and Reading Suggestions

Expected users of this applications are people of all ages and backgrounds. The user does not have any proper idea in this application.

1.4 Product Scope

The system initially provides for a doctor/patient online appointment system. It is created from information submitted by the doctor through online forms. The patient uses the system to find a doctor and view their schedule. The patient then uses the system to request an appointment at an available time. The doctor looks at a master view of the schedule and approves the requested appointments. An email message is sent to the patient to confirm the appointment that they requested

1.5 References

This document conforms to the IEEE 830 Standard for Software Requirements Specifications. The UML artifacts are compliant with UML Version 1.1.

2 Overall Description

The remainder of this document is organized as follows: Section 2 presents a General Description of the system. Specific requirements are presented in Section 3. The specific requirements are categorized as functional, external interface, performance, design constraints, attributes, or other requirements. The functional requirements section includes the UML analysis diagrams. The UML analysis artifacts are use case diagrams, sequence diagrams, and an analysis-level class diagram. A second iteration would include use case texts and operations specifications (pre/post-conditions).

2.1 Product Perspective

He proposed system allows doctors to make their schedules available online so that patients can find doctors and make appointments from the convenience and privacy of their web browser.

2.2 Product Functions

- Find Doctor – A patient uses this interface to search for a doctor. Ultimately, the patient might be able to search for doctors based on appointment availability time (e.g. I need to find a doctor that takes appointments early in the morning).

- Patient Schedule – This is a limited view of the schedule in which patients request appointments. It does not show what other patients have made appointments for. Approved and requested appointments are represented as unavailable appointment times.

- Doctor Schedule – This is a master view of the schedule that shows detailed appointment information. Some of this information should be hidden from the patient view. Doctors can use this view to approve requested appointments.

2.3 User Classes and Characteristics

The users of this system are assumed to have a basic understanding of the Internet.

3. Specific Requirements

3.1 **Functional Requirements**

The functional requirements were derived from the following use case diagram. Use case diagrams represent the functional interactions of a system. The stick figures represent the actors, which are external to the system and interact with the system through interfaces. The actors of the online appointment system are the Patient and the Provider. The Patient uses the system to schedule appointments to obtain services from Providers. The ovals are individual use cases that represent the functions the system performs to provide the services that the actors desire. The primary use cases of the online appointment system are: Find Provider, Manage Schedule, Manage Appointment, and Register. The Patient interacts with

the following use cases: Register, Manage Appointment, Find Provider. The Provider interacts with the following use cases: Register, Manage Appointment, Manage Schedule.

3.1.1.1 *Introduction*

Customers shall be allowed to search for providers by a variety of criteria. This criteria shall be by zip code and specialty but must be extensible so that any it is possible to sort or search on any data field.

3.1.1.2 *Inputs*

The user would input the searching criteria/values and the sorting criteria.

3.1.1.3 *Processing*

The system performs a search on the database of provider

3.1.1.4 *Outputs*

The user is presented with a list of providers that match the searching criteria sorted as defined by the sorting criteria. By clicking on any doctor listed the user is taken to that doctor's main appointment page.

3.1.3 Cancel Appointment

3.1.3.1 *Introduction*

This allows patients or providers to cancel an appointment. The following sequence diagram represents the patient initiating to cancel.

3.1.3.2 *Inputs*

The patient selects to delete his or her appointment.

3.1.3.3 *Processing*

The system marks the appointment as cancelled.

3.1.3.4 *Outputs*

The provider is emailed a notification of the cancellation, and the patient is displayed a confirmation that it has been cancelled. The patient might ensue some fee for cancelling after a certain time period.

4. Other Nonfunctional Requirements

4.1 User interfaces

All user (Patient, Provider) interfaces are web-based HTML pages which can be accessed through any standard web browser. Web pages which ask for user input will include HTML forms. These HTML forms need to include JavaScript code to verify that the appropriate information has been input before allowing the information to be submitted to the web server. Web pages which provide confirmation information will be dynamically generated static HTML pages.

4.2 Hardware interfaces

There are not direct hardware interfaces in the system. The operating systems on both the client and server side will be configured to handle the hardware interfaces.

4.3 Software interfaces

The client side software interface will be through any standard web browser. The server side interface will be through a Java web server.

4.4 Communication interfaces

Communication between the client and servers will occur between the standard internet protocol HTTP. Mail transfers will be accomplished through SMTP.

4.5 Performance Requirements

Customers will be using this system to make important appointments. If the system is unresponsive customers will be lost. The response time for any operation shall be no longer than two (2) seconds not including internet latency. To test the response time a computer will be placed on a LAN with the server and perform operations.

4.6 Standards compliance

The system complies with the internet standards of HTML, HTTP, and MIME. These standards support the largest number of online customers.

4.7 Hardware limitations

There are no hardware limitations to be considered.

5 Other Requirements

.5.1 Database

A database is not required from the offset. Files will be used to handle data storage.

5.2 Operations

Backups will be performed by both the owners and the administrators of the web hosting service.

5.3 Site adaptation

The site will be adaptable to the different users of the system, according to the access associated with the login that they provide.