**Interactive** **Health** **Care** **System**

**Software Requirements Specification**

***Version 3.0***

***March 17, 2015***

**The Development Team** Bharadwaj Aryasomayajula Clinton Jarboe

Arpan Chatterjee

Thaynara Regina Silva Alves

Elexus Rangel

Travis Finell

Submitted in partial fulfillment

Of requirements of the Course Project of

CSE 360 Introduction to computer Science.

**Table** **of** **Contents**

Table of Contents..........................................................................i

List of Figures..............................................................................ii

1.0 Introduction............................................................................1

1.1 Purpose...................................................................................1

1.2 Scope of Project.....................................................................1

1.3 Glossary..................................................................................2

1.4 Overview of Document..........................................................2

2.0 Overall Description................................................................3

2.1 System Environment..............................................................3

2.2 Requirement Specifications....................................................4

2.2.1 Login........................................................................4

2.2.2 View Reports............................................................4

2.2.3 Submit Report...........................................................5

2.2.4 View Appointments...................................................5

2.2.5 Suggest care..............................................................6

2.2.6 New Appointment.....................................................6

2.2.7 Modify Appointment.................................................7

2.3 User Characteristics................................................................7

2.4 Non-Functional Requirements................................................8

3.0 Requirements Specifications...................................................8

3.1 External Interface Requirements.............................................8

3.2 Functional Requirements.........................................................8

3.2.1 User Login................................................................8

3.2.2 Submit Report...........................................................9

3.3.3 New Appointment.....................................................9

3.3.4 View Appointments.................................................10

3.3.5 New Appointment....................................................11

3.3.6 Modify Appointment...............................................11

3.3.7 Suggest Care............................................................12

3.2 Functional Requirements.........................................................8

3.3 Logical Structure of the Data.................................................13

**List** **of** **Figures**

Figure 2.1 – Interactive Healthcare System.................3

Figure 2.2 – Login Use Case........................................4

Figure 2.3 – View Reports Use Case............................5

Figure 2.4 – Submit Report Use Case..........................5

Figure 2.5 – View Appointments Use Case..................5

Figure 2.6 – Suggest Care Use Case.............................6

Figure 2.7 – New Appointment Use Case.....................6

Figure 2.8 – Modify Appointment Use Case................7

**Revisions/Updates to SRS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version #** | **Implemented**  **By** | **Revision**  **Date** | **Approved**  **By** | **Approval**  **Date** | **Reason** |
| **1.0** | **Clinton Jarboe** | ***02-23-2015*** | ***Arpan*** | ***02-23-2015*** | **Test Plan draft** |
| **2.0** | **Bhardwaj** | **03-17-2015** | **Arpan** | **03-20-2015** | **Fine tuning of requirements and grouping of functionalities.** |
| **3.0** | **Bharadwaj** | **03-23-2015** | **Clinton** | **03-23-2015** | **Adding Requirements for Extra Credits** |

Changes made Version 3.0 in brief:

* Adding Requirements for Extra Credits.

Changes made Version 2.0 in brief:

* Added table grouping the functionalities by user
* Refined Login Feature.
* Refined Naming of some features.
* Added some Non functional Requirements.

**1.0 – Introduction**

**1.1 - Purpose**

The purpose of this software will be for the everyday use of medical professionals and their patients. It will be designed with ease of use being at the forefront. This means that anyone will be able to use the software with little effort, thereby improving efficiency. The vast improvement that the ability to access and create records automatically, without having to manually copy out and file all given information, cannot be overstated. This software will also allow for the filing of other medical information such as prescriptions and also provides patients with a convenient way to add new ailments, change contact information, and request a different doctor. A database is stored of all critical information.

**1.2 – Scope of Project**

This software system is an Interactive Healthcare System designed to improve communication and scheduling between healthcare practitioners, with the goal of improving overall efficiency for the healthcare facility. The system allows for quick communication of the patient's symptoms, as well as quick treatment suggestion for non-emergency cases.

Specifically, the system allows patients to submit their healthcare concerns, which are evaluated by a healthcare provider, who then suggests treatment or notifies administrative staff to schedule an appointment with the patient. The system contains a relational database between symptom reports, treatments, patients, and practitioners.

**1.3 – Glossary**

|  |  |
| --- | --- |
| Appointment | An appointment for a specific patient to meet with his doctor. |
| Client | The front-end portion of the system, which runs on individual devices. |
| Database | The back-end portion of the system, stored on a central server. |
| Doctor | A medical doctor in the system. |
| Healthcare practitioner | Any doctor or nurse in the system. |
| Nurse | A medical nurse in the system. |
| Receptionist | An administrative worker at the healthcare facility. |
| Patient | A patient of the healthcare facility in the system. |
| Suggested treatment | A course of treatment recommended by a healthcare practitioner through the system. |
| Symptom report | A report of symptoms submitted by a patient to his doctor through the system. |
| User | Any doctor, nurse, patient, or receptionist in the system. |

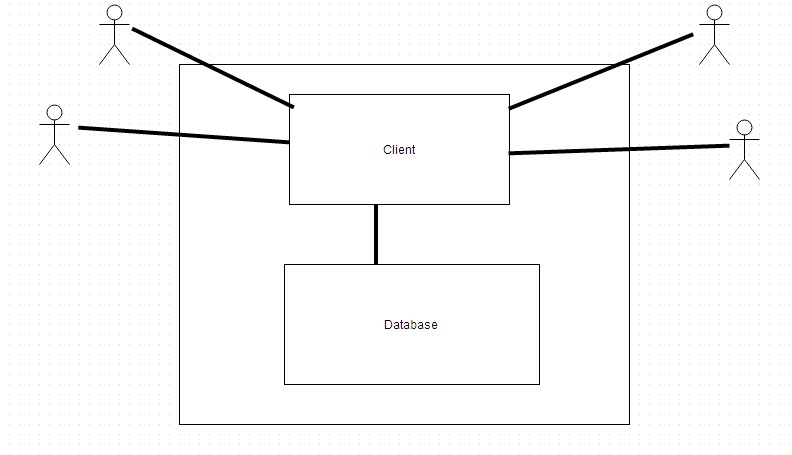
**1.4 – Overview of Document**

The next section, Overall Description, contains a breakdown of use cases for users of the system. This is used to establish a basis for understanding the technical specifications provided in the following chapter.

The third chapter is Requirements. These sections break down the functional and database features of the system for the purpose of reference by the software's developers.

**2.0 – Overall Description**

**2.1 – System Environment**

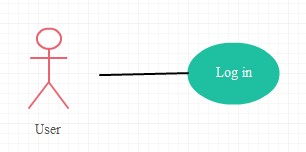


*Fig. 2.1 - Interactive healthcare system*

The symptom report system has four actors and one central system. The patient submits reports and views appointments through the client. The nurse and doctor view symptom reports, and the receptionist and doctor set and modify appointments through the client. The client also automatically schedules appointments for the patient if the situation is severe enough.

**2.2 - Functional Requirement Specification**

**2.2.1 Log in**



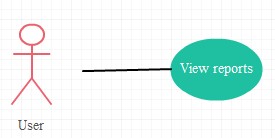
*Fig 2.2 - Login Use Case*

The User accesses the client by entering his user ID and password. Initial step-by-step description

1. The User enters his user type,user ID and password

2. He is able to view a 3-pane homepage window with one pane containing his report history and a button to create a new report, and another containing his appointments. The third pane contains any suggested treatments from their healthcare practitioners

**2.2.2 View reports**



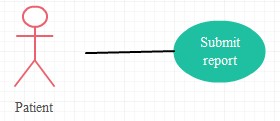
*Fig 2.3 - View reports use case*

The User is able to view his past reports. Initial step-by-step description

Before this step has been initiated, the user has already loaded the client and logged in.

1. The User is shown a list of his past reports in descending chronological order.

**2.2.3 Submit Report**



*Fig. 2.4 - Submit report use case*

The patient submits a report of their current symptoms and severity for use by healthcare practitioners.

Initial step-by-step description:

Before this step has been initiated, the patient has already loaded the client and logged in.

1. the patient selects the symptoms that they are currently experiencing

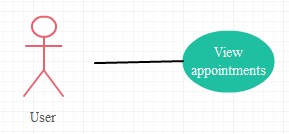
2. the patient rates the severity of those symptoms

3. if the situation is severe, the patient is advised to seek emergency care

4. the patient chooses to request an appointment, sending an automated email to the

Receptionist and doctor

**2.2.4 View appointments**



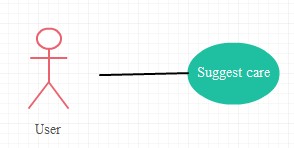
*Fig 2.5 - View appointments use case*

The patient views their past and upcoming appointments. Initial step-by-step description:

Before this step has been initiated, the patient has already loaded the client and logged in.

1. the patient is shown a list of appointments in descending order, with upcoming appointments highlighted at the top

**2.2.5 Suggest care**



*Fig. 2.4 - Suggest Case Use Case*

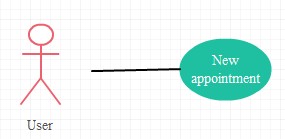
The Authorized user suggests care for the patient's symptoms. Initial step-by-step description:

Before this step has been initiated, the user must be viewing a patient's record.

1. The User may write a note to the patient suggesting care (ie ice, rest, schedule an appointment)

2. The User may check a box to submit an appointment request to the receptionist for the patient

**2.2.6 New appointment**



*Fig. 2.7 - New Appointment use case*

The authorized User can schedules an appointment. Initial step-by-step description:

Before this step has been initiated, the receptionist has already loaded the client and logged in.

1. The user selects 'schedule new appointment' from their homepage

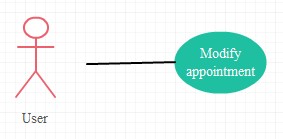
2. The user enters his name, ID, time and a date or for a different patient with proper

Authorization to visit their doctor. This must be more than 15 minutes after any existing appointment for that doctor

3. The User submits the new appointment and it is added to the schedule

4. any existing appointment request for that patient is marked fulfilled

**2.2.7 Modify appointment**



*Fig. 2.8 - Modify Appointment use case*

The User modifies an existing appointment for a patient. Initial step-by-step description:

Before this step has been initiated, the receptionist has already loaded the client and logged in.

1. The User selects an appointment from the list

2. The User may modify the time or date of the visit

3. The User may cancel the appointment altogether

4. The User submits all changes.

**Functionalities as per users**

So summarizing the use cases as per the users as follows

|  |  |  |
| --- | --- | --- |
| Functionality | Patient | Health Care Professional |
| Login | Yes | Yes |
| View Reports | Yes | Yes |
| Submit Symptom Report | Yes | No |
| New Appointment | No | Yes |
| Modify Appointment | No | Yes |
| View Appointments | Yes | Yes |
| Suggest Care | No | Yes |

**2.3 User characteristics**

The doctor, nurse, receptionist, and patient are expected to be able to fill out forms on a computer. All are expected to be familiar with basic UI features like dropdown menus, checkboxes, and form submissions.

**2.4 Non-functional requirements**

The user software will run as an application or applet on user devices.

The SQL database must be hosted on a server with high-speed internet in order to synchronize efficiently.

The Application will provide best results With JDK (8.0 or more) in a resolution of 640X480 or more

With more than 512 Mb Heap Memory.

The User can install the Application using the installer Provided.

**3.0. Requirements Specification**

**3.1 External Interface Specification**

We are not using any external interface at this moment. If we decide to add any external interface at a later stage of software development we will add it to the documents.

**3.2 Functional Requirements**

**3.2.1 User Login**

|  |  |
| --- | --- |
| Use Case Name | User Login |
| Xref | Section 2.2.1 |
| Trigger | The user a **health care professional or an patient** wants to  Login |
| Precondition | The user has valid username and password. |
| Basic Path | 1. User Navigates to the login page.  2. Selects Health care professional or patient from user type.  3. User enters username and password.  4. User clicks on submit button.  5. User navigates to his homepage |
| Alternate Path | There are no alternate paths available. |
| Post Condition | If User has entered Wrong password more than thrice his session is locked. |
| Exception Path | Gets an invalid credentials error for improper credentials  If user Enters wrong password more than 3 times his session is locked |
| Others | The passwords are protected in Db using Hash function. |

**3.2.2 Submit Report**

|  |  |
| --- | --- |
| Use Case Name | Submit Report |
| Xref | Section 2.2.3 |
| Trigger | **Patient** clicks a link on a homepage(Submit Report) |
| Precondition | The Patient is able to login |
| Basic Path | 1. User Logs into his home page.  2. User Clicks on Submit Report Button.  3. User Describes his conditions  4. User rates his symptoms on a scale of 1 to 10.  5. User clicks submit button.  6. System prompts the User to make an appointment based his symptoms. |
| Alternate Path | There are no alternate paths available. |
| Post Condition | Patient is able to submit his form |
| Exception Path | If patient doesn’t rate any of the symptom he gets an error prompt |
| Other | NA |

**3.2.3 View Reports**

|  |  |
| --- | --- |
| Use Case Name | View Reports |
| Xref | Section 2.2.2 |
| Trigger | Health care professional or Patient clicks views report history on homepage. |
| Precondition | The user is able to login |
| Basic Path | 1. User Logs into his home page.  2. User views report history of himself if a patient, or his patients if a healthcare practitioner. |

|  |  |
| --- | --- |
|  | 3. User has the option to interact with past reports in  Different ways depending on his user profile. |
| Alternate Path | There are no alternate paths available. |
| Post Condition | There are no post condition available. |
| Exception Path | There is no reports |
| Other | NA |

**3.2.4 View Appointments**

|  |  |
| --- | --- |
| Use Case Name | View Appointments |
| Xref | Section 2.2.4 |
| Trigger | Heath Care Personal or Patient Logs into his Homepage. |
| Precondition | The Patient has at least one scheduled Appointments |
| Basic Path | 1. User Logs into his home page.  2. His Appointments are displayed on chronological order  3. The Appointment Date, Time, doctor and hospital are shown in a table. |
| Alternate Path | 1. User Makes a new Appointment.  2. After Successful creation all the appointments are shown. |
| Post Condition | User is able to submit his form |
| Exception Path | Gets an error you have no appointments if he has no appointments |
| Other | NA |

**3.2.5 New Appointment**

|  |  |
| --- | --- |
| Use Case Name | New Appointment |
| Xref | Section 2.2.6 |

|  |  |
| --- | --- |
| Trigger | Health care professonal Logs into his Homepage. |
| Precondition | The user has authorization to create an appointment and is  able to login. |
| Basic Path | 1. The Receptionist selects 'schedule new appointment'  from their homepage  2. The Receptionist selects a patient as well as a time and  a date for the patient to visit their doctor. This must be more than 15 minutes after any existing appointment for that doctor  3. The Receptionist submits the new appointment and it  is added to the schedule  4. Any existing appointment request for that patient is  marked fulfilled |
| Alternate Path | NA |
| Post Condition | Receptionist is successful in submitting the Request. |
| Exception Path | NA |
| Other | NA |

**3.2.6 Modify Appointment**

|  |  |
| --- | --- |
| Use Case Name | Modify Appointment |
| Xref | Section 2.2.7 |
| Trigger | The Health care professional Logs into their Homepage. |
| Precondition | The Patient has at least one scheduled Appointments |
| Basic Path | 1. The receptionist selects an appointment from the list.  2. the receptionist may modify the time or date of the visit  3. the receptionist may cancel the appointment altogether  4. the receptionist submits all changes |

11

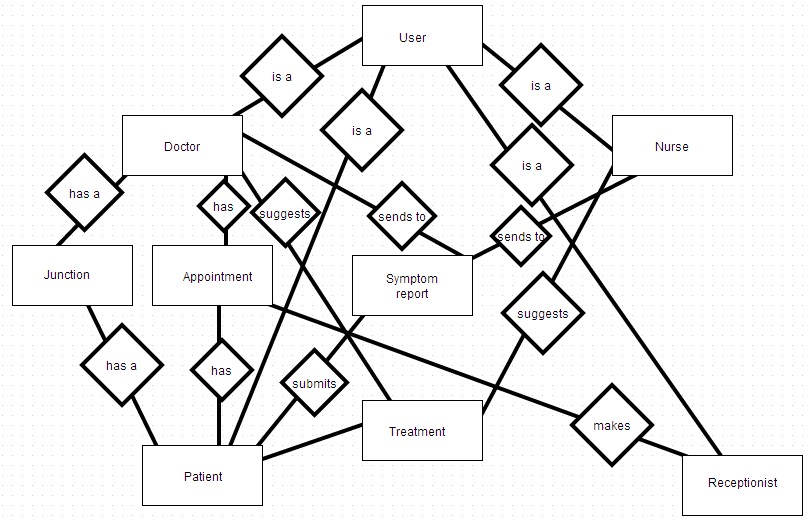
|  |  |
| --- | --- |
| Alternate Path | NA |
| Post Condition | Receptionist is successful in submitting the Request. |
| Exception Path | NA |
| Other | NA |

**3.2.7 Suggest Care**

|  |  |
| --- | --- |
| Use Case Name | Suggest Care |
| Xref | Section 2.2.5 |
| Trigger | Health care user Logs into their Homepage. |
| Precondition | The User is able to log into the System. |
| Basic Path | 1. The User may write a note to the patient suggesting  care (ie ice, rest, schedule an appointment)  2. The User may check a box to submit an appointment  request to the receptionist for the patient |
| Alternate Path | The User can click on Submitted report and write care. |
| Post Condition | The User is able to Complete the session successfully. |
| Exception Path |  |
| Other |  |

12

**3.3 - Logical structure of the data**



**User data entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Username | Text | First letter of first time and last name, plus a number if needed | Primary key - unique |
| Password | Text | The user's password | Hashed |
| Last Name | Text | The user's last name |  |
| First Name | Text | The user's first name |  |
| Email | Text | The user's email |  |
| Phone number | Int | The user's phone number |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Text | The type of user | Patient, Doctor, Receptionist, Nurse |

**Symptom report data entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Primary key | Int | For indexing | No functional use |
| Symptoms | Text | A list of symptoms being experienced | Symptoms separated by commas |
| Severity | Text | A list of the severity of those symptoms | Ints separated by commas |
| Score | Int | Used for determining whether to recommend an ER visit. |  |
| Date | Datetime | The time and date of the report submission |  |

**Patient-Doctor Junction data entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Primary key | Int | For indexing | No functional use |
| Patient | Text | The patient | Corresponds to patient username |
| Doctor | Text | The doctor | Corresponds to doctor username |

**Appointment data entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Primary key | Int | For indexing | No functional use |
| Patient | Text | The patient | Corresponds to a patient username |
| Doctor | Text | The doctor | Corresponds to a doctor username |
| Date and time | Datetime | The date and time of the appointment |  |
| Date made | Datetime | The date and time the appointment was made |  |

**Suggested treatment data entity**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Item** | **Type** | **Description** | **Comment** |
| Primary key | Int | For indexing | No functional use |
| Patient | Text | The patient for whom the treatment is suggested | Corresponds to a patient username |
| Symptom report | Int | The symptom report the suggestion is in response to | Corresponds to primary key of a symptom report |
| Date | Datetime | The date and time the suggestion was made |  |
| Recommendation | Text | The doctor or nurse's recommended course of action for the patient |  |

**3.3.2 Security**

The central database is stored on a secured server at or leased by the healthcare facility. The server will have its own security to prevent unauthorized access of any type.

The software will run on individuals' computers, and thus will use a hashed password to protect the confidential information accessed. Different user profile types will have different levels of read, write, and delete access to the server data.