**Software Requirements Specification**

**1. Introduction**

This section describes the scope and the overview of everything included in our project WeCare in this SRS document. The abbreviations and definitions that are used in this SRS are also described.

**1.1 Purpose**

The purpose of this Software Requirements Specification (SRS) is to define and describe the functions and specifications of our project WeCare. This Software Requirements Specification illustrates, in clear terms, the system’s primary uses and required functionality so that the next developers can easily understand what we did and how to change anything if it is required.

**1.2 Intended Audience**

The SRS is intended for software developers, designers, project managers and testers. The document is intended to be read sequentially as it is.

**1.2 Intended Use**

The SRS is intended to use for doctors and patients as well as medical site people like pharmacy.

**1.4 Product Scope**

WeCare is for patients who have difficulty recognized doctors handwriting and for doctors who can scheduling his/her appointment as well as easy way to find the medicine name with details to suggest patient. It is a software-based application. We will develop its android version later. A doctor can view selective information for each patient, they can search for medicines by their name or generics and type out the prescriptions with a very convenient way as this E- prescription will allow the doctor to organize the prescription in several specific section. A prescription will have a neat and informative interface and can be printed out too, if needed. A patient can view his/her records and prescriptions. We are also planning to add such an environment where a patient can easily find an appropriate doctor according to their symptoms. Though we are not sure about the stability of this function.

For admins, they will have all the access to spectate and modify the database. They will be able to create a new account as well as new admin in the database also will be able to manipulate them. Currently its available in Bangladesh as known as ‘Prescription writing software’ but have very few or less details or poor management system in this software. Our product will allow fast search medicine with auto details and auto scheduling appointment list, also have a speech to text functionality. We will add other features later.

**1.5 Risk Definition**

Monitoring and controlling risks as well as auto update information. Scheduling and appointment update. Try to detect speech more accurately. Above all, management handling issues.

**2. Overall Description**

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders that will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

**2.1 User Classes and Characteristics**

There are three types of users of the platform. The doctors, the patients and the administrators.  
The doctors and the patients will be able to the web app. They can use all the features.

The administrator will take all data from the doctors and patients will upload them on the web portal. They have all access to modify the whole system.

**2.2 User Needs**

This system will consist of one part: one web portal. The web portal will be used by both the doctors and the patients. The doctors can prescribe medicine and other details and the patients can take appointment, see doctor’s details and prescription details.  
The web application will both have a database to store medicine, prescription data and user data. The web portal will have read, write and modify access of the database. All database communication will happen through internet.The web portal will be accessible from any device as it won’t have any hardware constrain. The admins of the portal will be able to add new medicine with details, can add discount etc.

**2.3 Operating Environment**

Operating System: Windows any version

Software: Atom, Pycharm

**2.4 Constraints**

As this an online application, internet connection is a constraint. Internet connection is crucial for the application to fetch data from the database over the internet.  
Since the database is shared between both application it may be forced to queue incoming requests and therefor increase the time it takes to fetch data.

**2.5 Assumptions**

Proper browser should be used for the web app and a stable internet connection is required. We will develop the security issue. We will make it more efficient and reliable.

**3. Requirements**

All inputs and outputs necessary for the requirements will be specified in this section. Hardware, Software and communication interfaces will also be characterized here.

**3.1 Functional Requirement**

This section includes the requirements that describe all the essential functions of the system.

**3.1.1 Functional requirement 1.1**  
**ID: PFR1**  
TITLE: User Sign up  
ACTOR: Patient  
PURPOSE: Register the patient information  
OVERVIEW: Given that a user has downloaded the mobile application, or accessed the website then the patient should be able to register into the system. The patient must provide Name, Password, Address, Email, Gender, NID, Age.  
TYPE: Primary function  
PRIORITY: High  
DEP: none  
COURSE OF EVENTS:  
I) Open the application or website  
II) Enter all the information  
III) Click on Submit  
ALTERNATIVE COURSE OF EVENTS:  
I) Email not match with regex, show error: “Please provide a valid email”  
II) Password is less than 6 digits, show error: “Password must include at least 6 digits”  
III) Any field left blank, show error: “All the fields must be filled up”  
IV) NID fields must be filled up carefully.

**3.1.2 Functional requirement 1.2**  
**ID: PFR2**  
TITLE: User Registration  
ACTOR: Doctor  
PURPOSE: Register the Doctor information  
OVERVIEW: Given that a user has downloaded the mobile application, or accessed the website then the patient should be able to register into the system. The patient must provide Name, Password, Address, Email, Gender, NID, Age, Educational Qualification (completely), Specialized sector, working place and address (if any), personal chamber, timing etc.  
TYPE: Primary function  
PRIORITY: High  
DEP: none  
COURSE OF EVENTS:  
I) Open the application or website  
II) Enter all the information  
III) Click on Submit  
ALTERNATIVE COURSE OF EVENTS:  
I) Email not match with regex, show error: “Please provide a valid email”  
II) Password is less than 6 digits, show error: “Password must include at least 6 digits”  
III) Any field left blank, show error: “All the fields must be filled up”  
IV) NID fields must be filled up carefully.  
  
**3.1.3 Functional requirement 1.3**  
**ID: PFR3**  
TITLE: User Login  
ACTOR: Patient  
PURPOSE: Sign in the patient  
OVERVIEW: Given that the patient is already registered, the patient will be able to login by providing the email and password.  
TYPE: Primary function  
PRIORITY: High  
DEP: PFR1  
COURSE OF EVENTS:  
I) Select “Already Registered” button from the registration page.  
II) Enter email and password  
III) Click on login  
ALTERNATIVE COURSE OF EVENTS:  
I) Email not exist, show error: “This email is not registered yet. Please Register first.”  
II) Password does not match: “Password or email is not correct. Please retry!”  
III) Any field left blank, show error: “All the fields must be filled up”

**3.1.4 Functional requirement 1.4**  
**ID: PFR4**  
TITLE: User Login  
ACTOR: Doctor  
PURPOSE: Sign in the doctor  
OVERVIEW: Given that the doctor is already registered, the doctor will be able to login by providing the email and password.  
TYPE: Primary function  
PRIORITY: High  
DEP: PFR2  
COURSE OF EVENTS:  
I) Select “Already Registered” button from the registration page.  
II) Enter email and password  
III) Click on login  
ALTERNATIVE COURSE OF EVENTS:  
I) Email not exist, show error: “This email is not registered yet. Please Register first.”  
II) Password does not match: “Password or email is not correct. Please retry!”  
III) Any field left blank, show error: “All the fields must be filled up”

**3.1.5 Functional requirement 1.5**  
**ID: PFR5**  
TITLE: Doctor’s List  
ACTOR: User - patient  
PURPOSE: Want to search which kind of specialist needed  
OVERVIEW: Patient can search his desire kind of specialist and able to submit his needed then the doctor’s list will be shown with details  
TYPE: Primary function  
PRIORITY: Medium  
DEP: PFR3  
COURSE OF EVENTS:  
I) Type the search keywords  
II) Provide Tags (Hearts/Psychology etc)  
III) Click on submit  
ALTERNATIVE COURSE OF EVENTS:  
I) If desire kind of doctor’s aren’t in the database, show error: “No record matching your query were found”  
  
**3.1.6 Functional requirement 1.6**  
**ID: PFR6**  
TITLE: Prescription  
ACTOR: User - patient  
PURPOSE: Want to see prescription  
OVERVIEW: This section will include the past on going prescription  
TYPE: Primary function  
PRIORITY: High  
DEP: PFR3  
COURSE OF EVENTS:  
I) Submit prescription ID  
ALTERNATIVE COURSE OF EVENTS:  
I) If id not match send message “Not Found”

**3.1.6 Functional requirement 1.6**  
**ID: PFR6**  
TITLE: Prescription Query  
ACTOR: User - patient  
PURPOSE: Want to see reports   
OVERVIEW: This section will show patients reports as well as doctor’s suggestion  
TYPE: Primary function  
PRIORITY: High  
DEP: PFR3  
COURSE OF EVENTS:  
I) Go back ALTERNATIVE COURSE OF EVENTS:  
I) If there is no prescription, show error: “Please copy your desired prescription id for further query no matching your query were found”  
  
**3.1.7 Functional requirement 1.7**  
**ID: PFR7**  
TITLE: Settings  
ACTOR: User - Patient  
PURPOSE: Modify the profile page  
OVERVIEW: Show the user information to the user. This information will include, name, email, phone number, address and user type. User will be able to modify any of this information except for the mail.  
TYPE: Primary function  
PRIORITY: Low  
DEP: PFR3  
COURSE OF EVENTS:  
I) Select Profile button from home  
II) Select any field  
III) Modify the selected field  
IV) Click on Save Changes  
ALTERNATIVE COURSE OF EVENTS:  
I) Data not downloaded properly, show error: “Couldn’t connect to server, please check your internet connection and try again later!”  
  
**3.1.8 Functional requirement 1.8**  
**ID: PFR8**  
TITLE: Patient’s list  
ACTOR: User - doctor  
PURPOSE: View patient list  
OVERVIEW: Doctor can know patient’s details with preferable condition TYPE: Primary function  
PRIORITY: Medium  
DEP: PER4  
COURSE OF EVENTS:  
I) Select Go back  
ALTERNATIVE COURSE OF EVENTS:  
I) No Patient Found

**3.1.9 Functional requirement 1.9**  
**ID: PFR9**  
TITLE: Existing Patient’s list  
ACTOR: User - doctor  
PURPOSE: View patient prescription  
OVERVIEW: Doctor can edit or give a new prescription by submitting patients ID TYPE: Primary function  
PRIORITY: High  
DEP: PER4  
COURSE OF EVENTS:  
I) Select Submit Patient ID II) Select E- Prescription  
ALTERNATIVE COURSE OF EVENTS:  
I) No Patient ID Found

**3.1.10 Functional requirement 1.10**  
**ID: PFR10**  
TITLE: New Patient’s list  
ACTOR: User - doctor  
PURPOSE: patient prescription  
OVERVIEW: Doctor can add new patient details , then he can go to e-prescription page TYPE: Primary function  
PRIORITY: High  
DEP: PER4  
COURSE OF EVENTS:  
I) Select Submit Patient sign up form II) Select E- Prescription  
  
**3.1.11 Functional requirement 1.11**  
**ID: PFR11**  
TITLE: Settings  
ACTOR: User - Doctor  
PURPOSE: Modify the profile page  
OVERVIEW: Show the user information to the user. This information will include, name, email, phone number, address and user type. User will be able to modify any of this information except for the mail.  
TYPE: Primary function  
PRIORITY: Low  
DEP: PFR4  
COURSE OF EVENTS:  
I) Select Profile button from home  
II) Select any field  
III) Modify the selected field  
IV) Click on Save Changes  
ALTERNATIVE COURSE OF EVENTS:  
I) Data not downloaded properly, show error: “Couldn’t connect to server, please check your internet connection and try again later!”  
  
**3.1.12 Functional requirement 1.12**  
**ID: PFR12**  
TITLE: Administrator Login  
ACTOR: Admin  
PURPOSE: Login the admin  
OVERVIEW: Given that the admin is already registered the admin will be able to login in trough a web portal.  
TYPE: Primary function  
PRIORITY: High  
DEP: none  
COURSE OF EVENTS:  
II) Enter email/username and password  
III) Click on login  
ALTERNATIVE COURSE OF EVENTS:  
I) Email not exist, show error: “This email/user is not registered yet. Please Register first.”  
II) Password does not match: “Password or email is not correct. Please retry!”  
III) Any field left blank, show error: “All the fields must be filled up”  
  
**3.1.13 Functional requirement 1.13**  
**ID: PFR13**  
TITLE: Admin Home Page  
ACTOR: Admin  
PURPOSE: View the admin dashboard  
OVERVIEW: The admin dashboard will show doctors and patients list as well as settings and new admin creator.  
TYPE: Primary function  
PRIORITY: High  
DEP: PFR12  
COURSE OF EVENTS:  
I) View Doctor’s list  
II) View Patient’s list  
ALTERNATIVE COURSE OF EVENTS:  
I) No doctors are in the list II) No patients are in the list  
  
**3.1.14 Functional requirement 1.14**  
**ID: PFR14**  
TITLE: Doctor’s List  
ACTOR: Admin  
PURPOSE: View Doctor’s list with doctor’s details  
OVERVIEW: The admin will be able to view all doctor’s details as well as information TYPE: Primary function  
PRIORITY: Low  
DEP: PFR12  
  
**3.1.15 Functional requirement 1.15**  
**ID: PFR15**  
TITLE: Patient’s list  
ACTOR: Admin  
PURPOSE: View Patients list with patient’s details  
OVERVIEW: The admin will be able to view all patient’s list as well as information.  
TYPE: Primary function  
PRIORITY: Low  
DEP: PFR12

**3.1.16 Functional requirement 1.16**  
**ID: PFR16**  
TITLE: Admin Creator  
ACTOR: Admin  
PURPOSE: New admin create  
OVERVIEW: Admin can create new admin for increase management system and make things easier.  
TYPE: Primary function  
PRIORITY: High  
DEP: none  
COURSE OF EVENTS:  
I) Fill up sign up form  
II) Click on submit  
ALTERNATIVE COURSE OF EVENTS:  
I) If want to reset all , click on reset  
II) Any field left blank, show error: “All the fields must be filled up”

**3.1.17 Functional requirement 1.17**  
**ID: PFR17**  
TITLE: Settings  
ACTOR: Admin  
PURPOSE: Login the admin  
OVERVIEW: Admin can modify his record  
TYPE: Primary function  
PRIORITY: Medium  
DEP: PFR12  
COURSE OF EVENTS:  
II) Enter email/username and password  
III) Click on save changes  
ALTERNATIVE COURSE OF EVENTS:  
I) Any field left blank, show error: “All the fields must be filled up”

**3.1.18 Functional requirement 1.18**  
**ID: PFR18**  
TITLE: Prescription  
ACTOR: User - Doctor  
PURPOSE: Want to generate prescription  
OVERVIEW: This section will include the prescription details  
TYPE: Primary function  
PRIORITY: High  
DEP: none  
COURSE OF EVENTS:  
I) Submit chief complains II) Submit measurement and bp  
III) Submit body check-up if needed  
IV) Submit medicine with desire time  
V) Submit duration  
VI) Submit suggestion  
VII) Generate Prescription  
  
ALTERNATIVE COURSE OF EVENTS:  
I) Medicine not found “Not Found” I II) Empty Prescription submitted or prescription has no data

**3.2 Functional Requirement 2**

**3.2.1 Functional requirement 2.1**  
**ID: SFR1**  
TITLE: Download mobile application  
ACTOR: User  
OVERVIEW: A user should be able to download the mobile application through either an application store or similar service on the mobile phone. The application should be free to download.  
PURPOSE: In order for a user to download the mobile application.  
TYPE: Secondary function  
Priority: High  
DEP: None  
COURSE OF EVENTS:  
I) Downloads the app from google play store  
  
**3.2.2 Functional requirement 2.2**  
**ID: SFR2**  
TITLE: Check Password  
ACTOR: System  
RAT: Match the password  
DESC: Match password entered by the user during login. The inserted password will be matched against the one stored in the database.  
TYPE: Secondary function  
PRIORITY: High  
DEP: PFR2  
COURSE OF EVENTS:  
I) Get password from user  
II) Get password from databases  
III) Match if both the passwords are the same.  
ALTERNATIVE COURSE OF EVENTS:  
I) The errors will be handled by the view using this function

**3.2.3 Functional requirement 2.3**  
**ID: SFR3**  
TITLE: Check if User is logged in  
ACTOR: System  
PURPOSE: Check if user is already logged in  
OVERVIEW: The system may need to check if the user is properly logged in in different scenarios. The login state may be cached in the running device.  
TYPE: Secondary function  
PRIORITY: High  
DEP: PFR2  
COURSE OF EVENTS:  
I) Get user login status from device  
ALTERNATIVE COURSE OF EVENTS:  
I) Couldn’t access device storage, show error “Couldn’t access device memory. Check the permissions for this app”

**Other Nonfunctional Requirements**

**4.1 Performance Requirements**

The requirements in this section provide a detailed specification of the user interaction with the software and measurements placed on the system performance.  
  
**4.1.1 Prominent search feature**  
**ID: NFR1**  
TITLE: Prominent doctor’s search feature  
OVERVIEW: The search feature should be prominent and easy to find doctor for the patient.  
PURPOSE: In order to for a patient to find the search doctor’s details with time easily.  
DEP: none  
  
  
**4.1.2 Usage of the result in the list view**  
**ID: NFR2**  
TITLE: Usage of the result in the doctor’s list view  
OVERVIEW: The results displayed in the doctor’s list view should be user friendly and easy to understand. PURPOSE: In order to for a patient to view the desire doctor’s list easily.  
DEP: none  
  
  
**4.1.3 Response time**  
**ID: NFR3**  
TAG: Response Time  
GIST: The fastness of the search  
SCALE: The response time of a search  
METER: Measurements obtained from 1000 searches during testing.  
MUST: No more than 2 seconds 100% of the time.  
WISH: No more than 1 second 100% of the time.  
  
**4.1.4 System dependability**  
**ID: NFR4**  
TAG: System Dependability  
GIST: The fault tolerance of the system.  
SCALE: If the system loses the connection to the Internet or missing any sensor, the user should be informed.  
METER: Measurements obtained from 1000 hours of usage during testing.  
MUST: 100% of the time.

**4.2 Security Requirements**

**4.2.1 Communication Security**  
**ID: NFR5**  
TAG: Communication Security  
GIST: Security of the communication between the system and server.  
SCALE: The messages should be encrypted for log-in communications, so others cannot get user-name and password from those messages.  
METER: Attempts to get user-name and password through obtained messages on 1000 log-in session during testing.  
MUST: 100% of the Communication Messages in the communication of a log-in session should be encrypted.  
Communication Messages: Defined: Every exchanged of information between client and server.  
  
**4.2.2 Admin Login Security**  
**ID: NFR6**  
TAG: Admin Login Account Security  
GIST: Security of accounts.  
SCALE: If an admin tries to log in to the web portal with a non-existing account then the admin should not be logged in. The admin should be notified about log-in failure.  
METER: 1000 attempts to log-in with a non-existing user account during testing.  
MUST: 100% of the time.  
  
**4.2.3 Admin Account Security**  
**ID: NFR7**  
TAG: Admin Account Security  
GIST: Security of admin accounts.  
SCALE: An admin and IP address should not be able to log-in to the web portal for a certain time period after three times of failed log-in attempts.  
METER: 1000 attempts to log-in during the lock period after user account has been locked because of failed log-in attempts of three times.  
MUST: The locking period should be half an hour, and during that period the log-in function is disabled.  
  
**4.2.4 User Account Security**  
**ID: NFR8**  
TAG: User Create Account Security  
GIST: The security of creating account for users of the system.  
SCALE: If a user wants to create an account and the desired user name is occupied, the user should be asked to choose a different user name.  
METER: Measurements obtained on 1000 hours of usage during testing.  
MUST: 100% of the time.

**4.3 Software Quality Attributes**

The requirements in this section specify the required reliability, availability, and maintainability of the software system.  
  
**4.3.1 Reliability**  
**ID: NFR9**  
TAG: System Reliability  
GIST: The reliability of the system.  
SCALE: The reliability that the system gives the right result on a search.  
METER: Measurements obtained from 1000 searches during testing.  
MUST: More than 98% of the searches.  
PLAN: More than 99% of the searches.  
WISH: 100% of the searches.  
  
**4.3.2 Availability**  
**ID: NFR10**  
TAG: System Availability  
GIST: The availability of the system when it is used.  
SCALE: The average system availability (not considering network failing).  
METER: Measurements obtained from 1000 hours of usage during testing.  
MUST: More than 98% of the time.  
PLAN: More than 99% of the time.  
WISH: 100% of the time.  
  
**ID: NFR11**  
**TITLE: Internet Connection**  
OVERVIEW: The application should be connected to the Internet.  
PURPOSE: In order for the application to communicate with the database.  
DEP: none  
  
**4.3.3 Maintainability**  
**ID: NFR12**  
TITLE: Application extendibility  
OVERVIEW: The application should be easy to extend. The code should be written in a way that it favors implementation of new functions.  
PURPOSE: In order for future functions to be implemented easily to the application.  
DEP: none  
  
**ID: NFR13**  
TITLE: Application testability  
OVERVIEW: Test environments should be built for the application to allow testing of the application’s different functions.  
PURPOSE: In order to test the application.  
DEP: none  
  
**4.3.4 Portability**  
**ID: NFR14**  
TITLE: Application portability  
OVERVIEW: The application should be portable with Android.  
PURPOSE: The adaptable platform for the application to run on.  
DEP: none

**4.4 Business Rules**

Sales of illegal products and non-furniture products are not allowed. Administrator can view all user information obtained from user registration.

**5. Other Requirements**

*Backup and Recovery:* Keep proper system wide backups and the backup cycle should be regular and frequent enough.

**Appendix A: Glossary**

Definitions, acronyms, and abbreviations are listed below:

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Admin/Administrator | System administrator who is given specific permission for managing and controlling the system |
| Patient | Someone who can see doctor’s list and prescription view |
| Doctor | Someone who gives prescription and see patient’s list with details |
| DEP | Dependency |
| PFR | Primary Functional Requirements |
| SFR | Secondary Functional Requirements |
| TAG | A unique, persistent identifier contained in a PLanguage statement [2] |
| GIST | A short description to help understanding [2] |
| SCALE | The scale of measurement used to quantify the statement [2] |
| METER | The process or device used to measure using the SCALE [2] |
| MUST | The minimum level required to avoid failure [2] |
| PLAN | The level at which success can be claimed [2] |
| WISH | A desirable level of achievement [2] |

**Appendix B: Analysis Models**

We plan to include class diagrams after the prototype is finished.

**Appendix C: To Be Determined List**

Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.