

Software Requirements Specification

Project Name:

DocPlus

Project Members:

Dorothy Mtaty

Justin Esau

Abstract: This Software Requirements Specification (SRS) describes the system requirements, and system actions. The Software Requirements Specification includes;

- Introduction,
- Use Cases,
- Functional, Non-functional and Environmental Requirements,

For the software;

Keywords: DocPlus doctors' community system, uml, use case diagram, use case, software requirements specification, system requirements specification, Web 2.0, Symfony, Postgres, PHP, Ubuntu Linux, doctor, patient, doctor's assistant, administrator.

Software Requirements Specification

1.0 Introduction

A doctor's appointment management system is prepared for South African health sector medical practitioners. Using this system both doctors and their assisting personnel can prepare and easily manage their patients' appointment schedule. This management system is a good way of arranging the appointments, which help doctors work efficiently.

1.1 Purpose

DocPlus system should respond to the requests of doctors, their helping staff and patient. This system enables doctors' secretaries or themselves to give appointments to patients according to doctors' availability status. The system provides placing a new appointment, modifying and deleting an existing appointment & showing weekly schedule.

This system should facilitate adding & deleting and updating schedule seamlessly. Moreover, it should be user-friendly and understandable.

The aim of the project is to prepare a management system, which arranges the most appropriate time for patients, their doctor's ratings, recommendations. The system should have a high usability level with its user-friendly interface (a professional graphics design firm will be contracted to do this), simplicity in usage and success in practice.

1.2 Scope

The scope of the project can be summarized as follows:

- To prevent getting lost of appointment and patients information.
- To prepare suitable weekly schedule for the doctors.

- To present a good user interface for creating, deferring, cancelling, editing, and updating the appointment schedule.
- To help patients in locating the nearest doctor (or dentist).
- To cut costs associated with appointment placements.
- To create a community of medical practitioners and their patients.
- To report concisely the doctor-patient relationship by keeping every tiny detail of every appointment that takes (will take) place between them.
- To provide reporting for doctors and system administrators.
- To handle billing of doctors.
- To allow doctors to upload their profile and showcase their businesses.

1.3 Definitions

- Symfony (1.4) Framework, a high-level PHP Web framework that encourages rapid development and clean, pragmatic design.
- SRS - Software Requirements Specification
- SPMP - Software Project Management Plan
- SDD - Software Design Description
- User - The person who uses the software
- Doctor – A medical practitioner
- Secretary – A medical practitioner’s assistant
- Administrator – A person who manages the system, billing doctors etc.
- Web 2.0 – An approach to web application development that focuses on the interactivity of the system with the user. AJAX technology is severely employed here.
- Postgres – The most advanced open database that will be used to persistently store all the system data.
- PHP - The high-level, object-oriented programming language that is used in the coding the software
- TSP - Team Software Process

1.4 Overview

At the end of the 2011, Graphifox Software Services will prepare a system, which is a doctors' community system. The application will include all features mentioned in 'Purpose, Scope and Objective' part of this document. Graphifox will deliver a Project Planning Report on November 01, 2011; a Draft SRS report on November 04, 2011, and a SDD on November 17, 2011. As a part of the deliverables, the Graphifox team will also give an overall system presentation on January 09, 2011 to the client and other audience. The schedule of the project phases, milestones and corresponding documents are given in the following table:

Phase	Deliverables	Deadline
Project Planning	Project Planning Report	November 01, 2011
Software Requirements Specification	SRS Report	November 04, 2011
Design	SDD Report Design Presentation	November 17, 2011
Implementation	Interface Design Code Implementation + Presentation	December 05, 2011 January 09, 2011

2.0 Use Cases

UC101	Secretary gives an appointment to patient in first visit.
Actors	Secretary, system.
Goal(s)	<ol style="list-style-type: none"> 1) To register the patient. 2) To check doctor's free slots 3) To give appointment based on the availability of the doctor
Step(s)	<ol style="list-style-type: none"> 1) Secretary enters the patient info to the system. 2) System registers the patient. 3) Secretary gives an appointment date (morning or afternoon) for patient.
Result	<ol style="list-style-type: none"> 1) System saves the patient information to the database and sends a username and password to the patient's email. 2) Secretary assigns an appropriate appointment date. 3) System saves the appointment date. 4) System sends an appointment confirmations message (SMS or email) to the patient.
UC102	Patient comes to register on the system website (www.docplus.co.za)
Actors	Patient, system.
Goal(s)	<ol style="list-style-type: none"> 1) To be registered with the DocPlus system. 2) To activate their mobile number. 3) To activate the email.
Step(s)	<ol style="list-style-type: none"> 1) The patient goes to the DocPlus website (www.docplus.co.za) and registers with the system. 2) The system sends a registration confirmation message (SMS and email) 3) Patient completes registration process by strictly confirming their details (both mobile and email)
Result	<ol style="list-style-type: none"> 1) Patient has now been satisfactorily registered with the system 2) Patient can now make doctors' appointment using DocPlus with no hassle.
UC103	Patient searches for doctors by region, city and area, (or even by name)
Actors	Patient, system.
Goal(s)	<ol style="list-style-type: none"> 1) To peruse doctors' profiles 2) To choose a doctor based on their proximity, ratings and recommendations of other patients. 3) To fully ready the doctor's profile and other details.

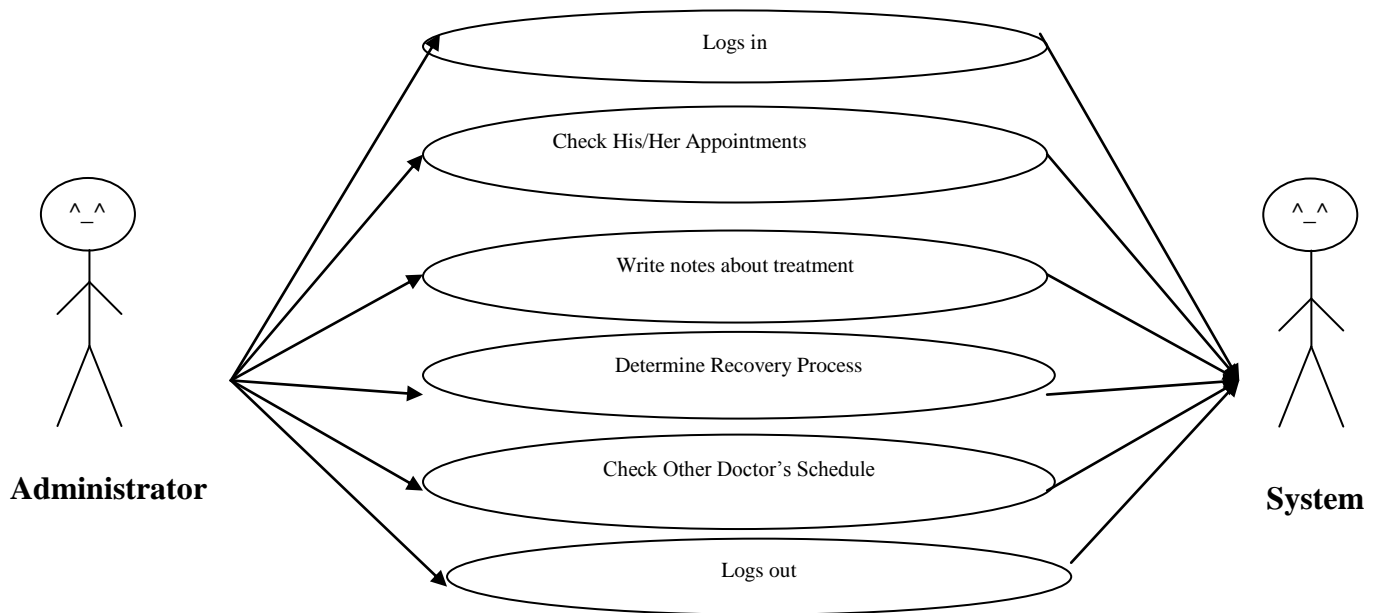
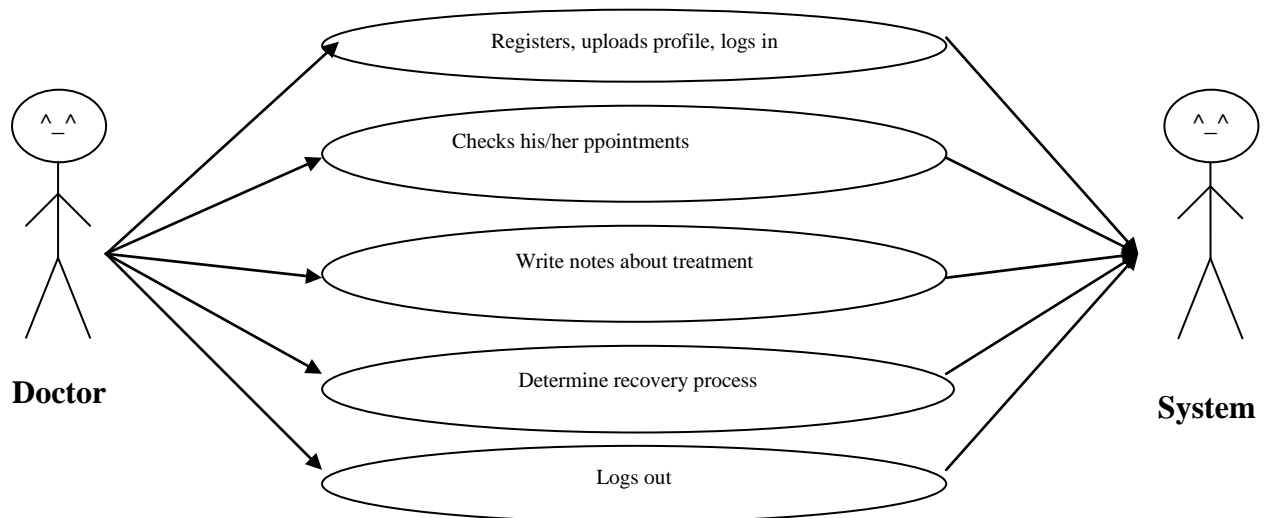
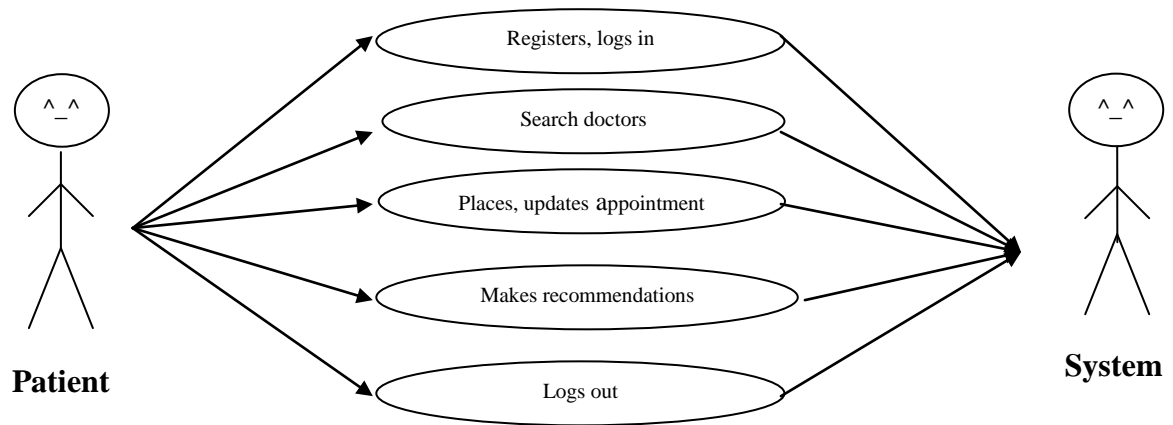
	<ol style="list-style-type: none"> 4) To request a doctor's appointment. 5) To receive their appointment placement confirmation.
Step(s)	<ol style="list-style-type: none"> 1) The patient searches for a doctor, by selecting region, entering city, entering area, or their specific doctor's name. 2) The patient reads the search returned doctors' profiles and subsequently chooses the doctor whom they prefer to the rest. 3) The patient reads other patients' ratings and recommendations about that specific doctor. 4) The patient can return to their search result to see other doctors' profiles. 5) Finally, the patient decides to make an appointment with one of the doctors. 6) The patient visually views the doctor's entire upcoming appointment schedule. 7) The patient selects a doctor's available slot and makes an appointment with that him/her.
Result	<ol style="list-style-type: none"> 1) An appointment placement notification reaches the doctor's office (by SMS or email). 2) The patient also gets system appointment confirmation message to confirm their placed appointment (this should not be confused with the actual doctor's confirmation). 3) The patient waits for the doctor's appointment confirmation
UC104	Doctor comes to register on the system website (www.docplus.co.za)
Actors	Doctor, system.
Goal(s)	<ol style="list-style-type: none"> 4) To be registered with the DocPlus system. 5) To activate their mobile number. 6) To activate their email. 7) To upload their profile details (contact details, business and personal photos).
Step(s)	<ol style="list-style-type: none"> 4) The doctor goes to the DocPlus website (www.docplus.co.za) and registers with the system. 5) The system sends a registration confirmation message (SMS and email). 6) Doctor completes registration process by strictly confirming their details (both mobile and email).
Result	<ol style="list-style-type: none"> 3) Doctor has now been satisfactorily registered with the system. 4) Doctor can now be searched and contacted by the patients using DocPlus with no hassle.
UC105	Doctor adds his/her assistants to the system
Actors	Doctor, system.

Goal(s)	<ol style="list-style-type: none"> 1) To add his/her assistants to the system 2) To delegate appointment schedule management to their assistants.
Step(s)	<ol style="list-style-type: none"> 1) Doctor enters his/her assistant's username. 2) Doctor enters his/her assistant's password. 3) Doctor enters other assistant's details (First name, Last name, Mobile, etc.) 4) System validates these details.
Result	<ol style="list-style-type: none"> 1) Doctor's assistant has been registered successfully. 2) The assistant gets a confirmation (SMS and email) of their registration.
UC303	Doctor or his/her assistant logs in.
Actors	Doctor, system.
Goal(s)	<ol style="list-style-type: none"> 3) To be logged in. 4) To check his/her patient appointments. 5) To enter notes to patients. 6) To determine recovery process.
Step(s)	<ol style="list-style-type: none"> 5) Doctor enters his/her username. 6) Doctor enters his/her password. 7) System validates the doctor. 8) Doctor sees his/her appointment schedules. 9) Doctor accepts the patient's appointment. 10) Doctor rejects or reschedules the patient's appointment.
Result	<ol style="list-style-type: none"> 3) Doctor is logged in successfully. 4) The patient gets a confirmation (whether accepted or rescheduled) of their appointment with the doctor. 5) Doctor determines recovery process.
Alternative	<ol style="list-style-type: none"> 1) Doctor enters wrong username or/and password.
Result	<ol style="list-style-type: none"> 1) System rejects the username and password. 2) Doctor re-enters the username and password.
UC106	Doctor or his/her assistant updates the patient appointment result.
Actors	Doctor, system.
Goal(s)	<ol style="list-style-type: none"> 1) To update how the appointment went with the patient. 2) To mark the appointment complete.
Step(s)	<ol style="list-style-type: none"> 1) Doctor selects the just ended appointment from their schedule calendar. 2) Doctor adds notes to the appointment for the patient to see on their side, and marks the appointment complete.
Result	<ol style="list-style-type: none"> 1) Doctor has successfully executed the appointment with the patient.

UC107	The patient fails to make it for the appointment.
Actors	System and doctor.
Goal(s)	1) To record the appointment as cancelled or missed. 2) To maintain a clear representation of their appointment schedules.
Step(s)	1) Doctor selects the missed appointment and updates it accordingly. 2) Doctor checks for the upcoming appointments on the schedule list.
Result	1) The system sends an SMS or email about the missed appointment to the patient.
UC108	Secretary logs in.
Actors	Secretary, system.
Goal(s)	1) To be logged in. 2) To register patient 3) To give, check, update an appointment.
Step(s)	1) Secretary enters his/her username. 2) Secretary enters his/her password. 3) System validates the secretary.
Result	1) Secretary is logged in.
Alternative	Secretary enters wrong username or/and password.
Result	1) System rejects the username and password. 2) Secretary re-enters his/her username and password.
UC109	Secretary updates an appointment date.
Actors	System and secretary.
Goal(s)	1) Secretary updates patient's appointment date.
Step(s)	1) Secretary finds the new appointment date from the system. 2) Secretary cancels the old appointment date from the system. 3) Secretary gives the new appointment date from the system. 4) System saves changes.
Result	1) New appointment date is successfully saved. 2) The patient gets notified of the modifications to their doctor's appointment.

UC110	The patient refers their doctor to a friend
Actors	System and patient.
Goal(s)	1) To refer his/her doctor to a friend.
Step(s)	1) Patient visits the doctor's profile and decides to send him/her to a friend. 2) The patient enters their friend's full name and email.
Result	1) The system sends an SMS or email about the referral to the referred friend.
UC111	System administrator logs in.
Actors	Administrator, system.
Goal(s)	1) To be logged in. 2) To manage doctor's billing activities. 3) To view doctors-patients appointments conducted in the system for a specific period. 4) To activate/deactivate patients or doctors.
Step(s)	1) Administrator enters his/her username. 2) Administrator enters his/her password. 3) System validates the administrator. 4) Administrator pulls a report of doctors-patients appointments. 5) Administrator deactivates the doctor's or patient's profile.
Result	1) Administrator is logged in. 2) Administrator now has appointments report. 3) A doctor's or patient's profile has been activate/deactivated.
Alternative	Administrator enters wrong username or/and password.
Result	1) System rejects the username and password. 2) Administrator re-enters his/her username and password.

2.1 Use Case Diagrams:



2.2 Use Case Suite

2.2.1 Use Cases by Functional Area

Patient To-Do list management

- Patient registers with the system.
- Patient logs into the system.
- Patient registers a patient
- Patient searches for the nearest doctor.
- Patient places a doctor's appointment
- Patient updates an appointment
- Patient checks their upcoming appointments
- Patient rates system doctors
- Patient refers a friend to his/her doctor.
- Patient logs out from the system

Assistant To-Do list management

- Assistant logs into the system
- Assistant registers a patient
- Assistant places an appointment for the patient
- Assistant updates an appointment
- Assistant checks all the appointments in the system
- Assistant logs out from the system

Doctor To-Do List

- Doctor registers with the system.
- Doctor checks his/her appointment schedule.
- Doctor writes notes/comments about the patient's treatment.
- Doctor determines the recovery process for the patient.
- Doctor registers their assistant with the system.

Administrator To-Do List

- Administrator does the general upkeep of the system.
- Administrator draws reports from the system.
- Administrator manages doctor's billing accounts.
- Administrator activates or deactivates doctor's or patient's accounts.

2.3 User Story

Doctor's assistant Victoria Debyshire enters his username and password to log into the system. Onur is a patient so he goes to the www.DocPlus.co.za website. To be completed...

3.0 Functional Requirements

Our program's functions;

- The system allows secretary to see appointment information of the patients.
- The system allows secretary to give, cancel or update appointment.
- The system allows doctors to see their patients' appointments and see their weekly schedule.
- The system allows doctors to determine the status of the treatment and can take notes about the treatment.
- The system allows patients to search and view doctors profiles.

4.0 Non-Functional Requirements

4.1 Usability Requirements

The DocPlus system provides medical practitioners to have appointments well scheduled. The only requirement is to have computers with Internet connection. The program provides doctors to see their weekly appointments while they are not at their offices. Accessibility of the information and usability of the program is easy. With few clicks the user can reach the destination information.

4.2 Reliability and up-time Requirements

Any reliability problem will not take place throughout the lifecycle of the software system.

Every data can be accessed and seen just after data entrance. The system will provide at least 99% uptime on Web hosting sites. Reliability factors will be supplied through:

- Success track record.
- Physical server security.
- Disaster recovery plan

4.3 Security Requirements

Since we use PHP and Symfony, all the security precaution that PHP and Symfony provides is our security policy. Apart from it, the login system will protect the information in the system from outside users. All user types such as secretary, doctors and head doctor have distinct pages to which only they can access. The password safety will be provided by the encryption of the passwords saved in the database.

Addition to all; Symfony Security Procedures will be provided.

4.4 Performance and Scalability Requirements

The DocPlus system can be applied to any community of doctors and their patients. The performance of the system will be appropriate for medical practitioners, which requires a high speed of interaction, and so all tasks will be carried out within a few clicks and seconds. The scalability requirements of the system are another important issue as well as the performance requirements. The scheduled appointment management system will have ability to provide all involved participants with efficient support, which will not be broken down.

4.5 Maintainability & Upgradeability Requirements

Making changes or upgradeability in the system will not be that much difficult. By having some knowledge of programming, some features of the system might be converted to a new version. According to the needs of upgrade, system requirements might change such as change in hardware or operating system or not.

4.6 Supportability & Operability Requirements

Supportability will be provided over the whole product life of the system. System will be quite easy to use but educational support will be given if needed. Although the system is completely open-source, there will be a price for developing the system. Some maintenance (5% of system development fee in price) will be provided without any request of payment in the first year of system installation. If this period of time exceeds or system is older than one year, maintenance will be supplied with the charge of money, which varies according to the type of maintenance. The scheduled appointment management system is a web based system and can be run on every computer with an Internet access. The system will can be installed for any operating system e.g. Microsoft Windows XP/Vista/7, or Linux. The system will be easily accessible to all South African medical practitioners and their patients.

4.7 Business Life-Cycle Requirements

The scheduled appointment management system is designed for coordinating doctor-patient appointment in CHCs. However, the scheduled appointment management system can be adjusted to many business areas requiring appointment. Hence, this system is feasible for a range group of business and great number of people in any sector. Some innovations in the system may be performed and can have a greater range of business life.

5.0 Environmental Requirements

5.1 System Hardware Requirements

- PC 1.6 GHz or higher.
- 512 Mb RAM or higher.
- 40 Gb Disc Space or higher.
- Video-Graphic Card (800 x 600) 128 Mb or more.
- Internet Access.

5.2 System Software Requirements

Software requirement to run the system:

- Microsoft Windows XP/Vista/7 or Linux.
- Any version of Internet Explorer, Mozilla Firefox, Opera etc.
- PHP, Symfony.
- Postgres.

5.3 Application program interfaces (APIs)

There will be no API for this application because there will be no additional software planned to be added to the program.

5.4 Data Import and Export Requirements

Application is a web based one and users can reach their data from any computer with the Internet.

- Import will not be performed.
- Export will not be performed.