SOFTWARE REQUIREMENT SPECIFICATION

1. ***Introduction***

A web platform for a hospital management system meant to organize the data and manage the events described in this documentation while taking advantage of the wide base of platforms that support this hospital web site.

1.1 Purpose

The main purpose of our system is to make hospital tasks easy and to develop software that replaces the manual hospital system with an automated hospital management system. This document serves as the unambiguous guide from the developers of this software system.

1.2 Feedback to the Customer

This software requirement specification assures the project management stakeholders and **-** client that the development team has really understood the business requirements documentation properly. This also provides confidence that the team will develop the functionality which has been detailed.

1.3 Breaking the Requirements Down

This document is documented in such a way that it breaks the deliverables into smaller components which makes the participants in this project to understand what is to be done clearly. The information is organized in such a way that all the developers within the team will not only understand the boundaries within which we need to work, but also what functionality needs to be developed and in what order.

Understanding what order the functionality will be developed in means that we, the developers will have the "big picture" view of the development. This gives us an opportunity to plan ahead which saves both project time and cost.

1.4 Facilitating other Documentation

The SRS forms the basis for a load of other important documents such as the Software Design Specification.

1.5 Product Validation

This document helps in validating with the client that the product which is being delivered, meets what they asked for. Which means that the product we have output is Equal to the standards of the documentation in the SRS which the client satisfied and agreed on.

1.6 Scope

The document only covers the requirement specification for the hospital management system. This document does not provide any references to the other component of the hospital management system. All the external interfaces and the dependencies are also identified in this document.

1.7 **Feasibility Study**

The overall scope of the feasibility study was to provide sufficient information to allow a decision to be made as to whether the hospital management system project should proceed and so, its relative priority in the context of the other existing hospital management system.

The feasibility study of this project had undergone through various steps which as describe as under:

a) Identify the origin of the information at different level.

b) Identify the expectation of user from computerized system.

c) Analyze the drawback of existing system.

1.8 **Definition, Acronyms, Abbreviations**

CFD: - Context Flow Diagram

DFD: - Data Flow Diagram

IDE: - Integrated Development Environment Java:-Platform Independent, Object oriented programming language

SQL: - Structured Query Language

SRS: - Software Requirement Specification

1.9 **Overview**

Hospital Management System is a process of implementing all the activities of the hospital in a computerized automated way to fasten the performance. This project is to add a new patient, search for existing patients by their Name, ID, make an appointment for a patient with a certain doctor, view all patients who have appointments, view the patient's details page that page contains the basic information of the patient in addition to his history ordered by date, and add a history record to the patient, these records should contain the date, doctor's name and description.

2 ***Overall Description***

2.1 **Product perspectives**

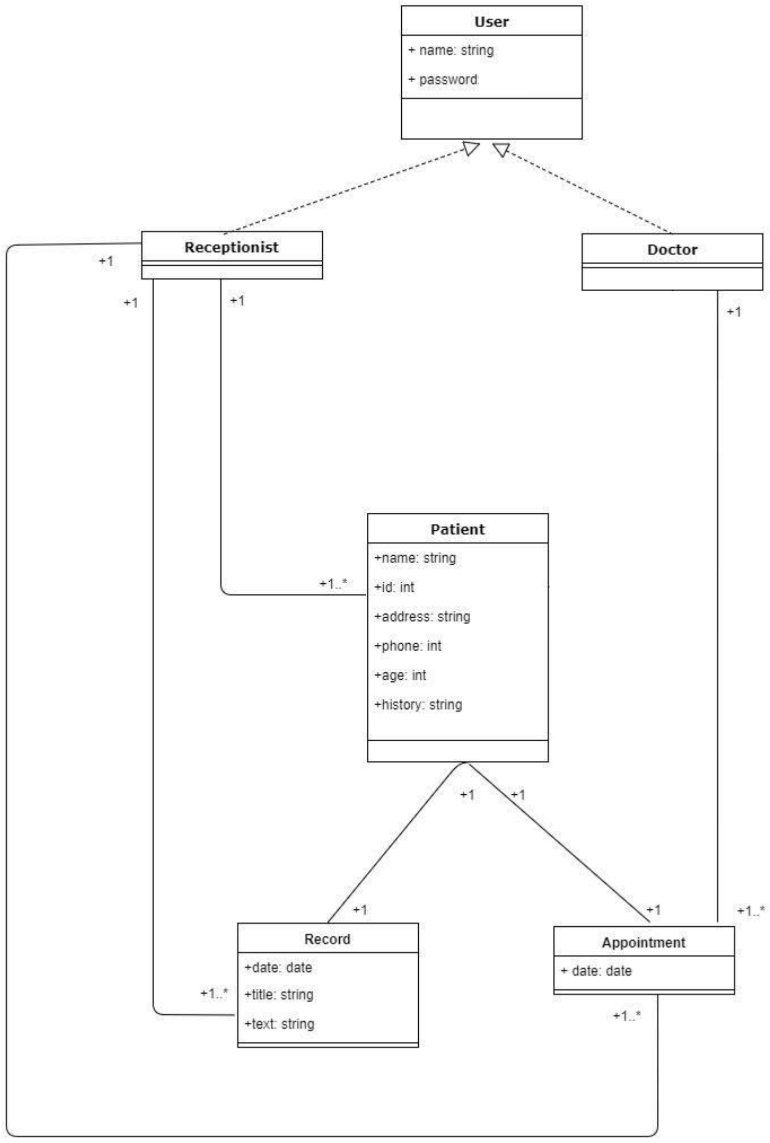
This project gives the procedural approach how a doctors gets details about date of patient who have appointments and add a history record to the patient.

2.2 Product Function

The data represented in hospital website will perform the following major function:- Patient Details: - It includes inpatient and outpatient details ,appointment for certain doctors , and history record to the patient. This enables the organization to keep the information in efficient and systematic way.

2.3 User Characteristics

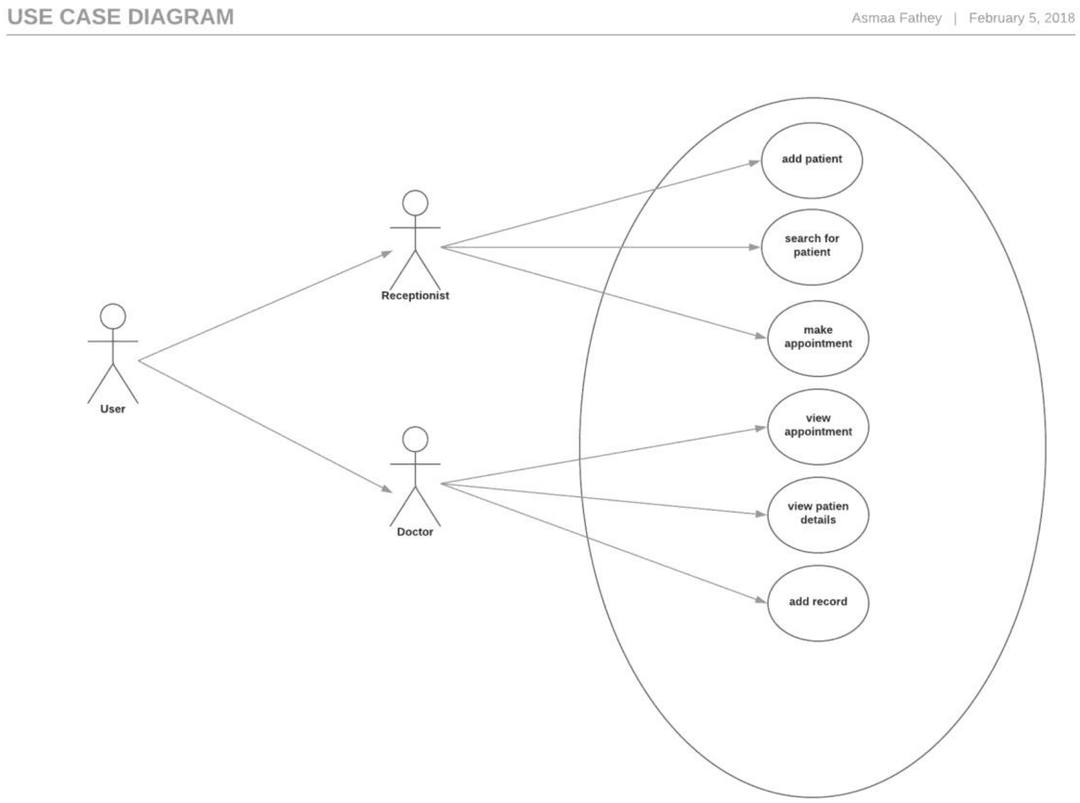
This software is developed such that total appearance of the product to make it more user friendly.



Admin has the full access to the system which means he is able to manage any activity with regard to the system. He is the highest privileged user who can access to the system.

2.3.2 Key functions

The key functions are best shown by this UML use case diagram.



2.4 Design and Implementation Constraints

Any update regarding the patients' information from the hospital are to be recorded to have updated and correct values

- System is only accessible within the hospital premises only.

- Database is password protected.

- Should use less RAM and processing power.

- Each system user should have individual ID and password.

2.5 Assumption and Dependencies

All the data entered will be correct and up to date .This software package is developed using python.

2.6 **Operating Environment**

- Software Requirements

Any machine with a web browser should be able to access this system .

- Hardware Requirements

Minimal hard ware is needed to access the system.

2.8 **Assumptions and Dependencies**

- Server must be running for the system to function

- Users must log in to the system to access any record

- Only the Administrator can delete records

3 ***Specific Requirements***

It describes all the details that the software developer need to know for designing and developing the system. This is typically the largest and most important part of the document

3.1 **External Interface Requirements**

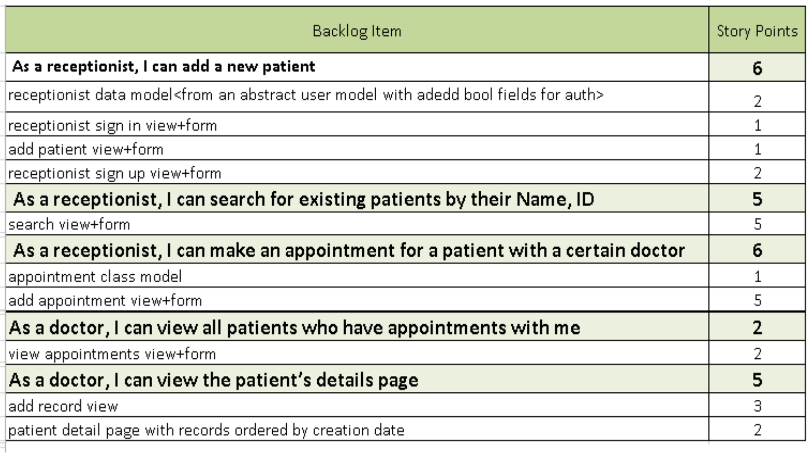
-Hardware Interface

The mainframe should be able to handle the data storage and should have ample IO and pressing power to keep up with the user requests.

-Software Interface

The server should have a platform that support Django deployment and maintenance.

3.2 **Functional Requirements**



Backlog Item	Story Points
As a receptionist, I can add a new patient	**6**
receptionist data model<from an abstract user model with adedd bool fields for auth>	2
receptionist sign in view+form	1
add patient view+form	1
receptionist sign up view+form	2
As a receptionist, I can search for existing patients by their Name, ID	**5**
search view+form	5
As a receptionist, I can make an appointment for a patient with a certain doctor	**6**
appointment class model	1
add appointment view+form	5
As a doctor, I can view all patients who have appointments with me	**2**
view appointments view+form	2
As a doctor, I can view the patient's details page	**5**
add record view	3
patient detail page with records ordered by creation date	2

This WBS clarifies the main functional requirement body.

3.2.1 Administration module

This module enables the user to insert, update, view and delete the patient information.

3.2.2 Patient module

Patient Id, Name, Age, Sex, Address, Phone Number, Weight.

This module has following 2 sub modules:

3.2.2.1 Inpatient module

This sub module is used to store information about patients who were admitted in the hospital on doctors' advice. Patient Id, Dept. depending on disease, Doctor, Room no, Date of admitted, Advance, Date of discharge.

3.2.2.2 Outpatient module

Patient Id, New Case, Old Case, Date, Department disease, Doctor.

3.2.3 Doctor module

Can view all patients who have appointments, can view the patient's details page, and can add a history record to the patient.

3.3 Nonfunctional Requirements

3.3.1 Performance Requirements

The capability of the computer depends on the performance of the software. The software can take any number of input provided the database size is large enough. This would depend on the available memory space.

3.3.2 Design Constraints

This will help the doctors or users to view the records of the patients immediately whenever necessary. This software also has the ability to add, update and delete the record whenever needed. This project will help to smoother the process of the hospital activates.

3.3.3 Safety Requirements

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

3.3.4 Security Requirements

All the administrative and data entry operators have unique logins so system can understand who is login in to system right now no intruders allowed except system administrative nobody cannot change record and valuable data.

3.3.5 Software Quality Attributes

-AVAILABILITY: The system shall be available all the time.

-CORRECTNESS: A bug free software which fulfill the correct need/requirements of the client.

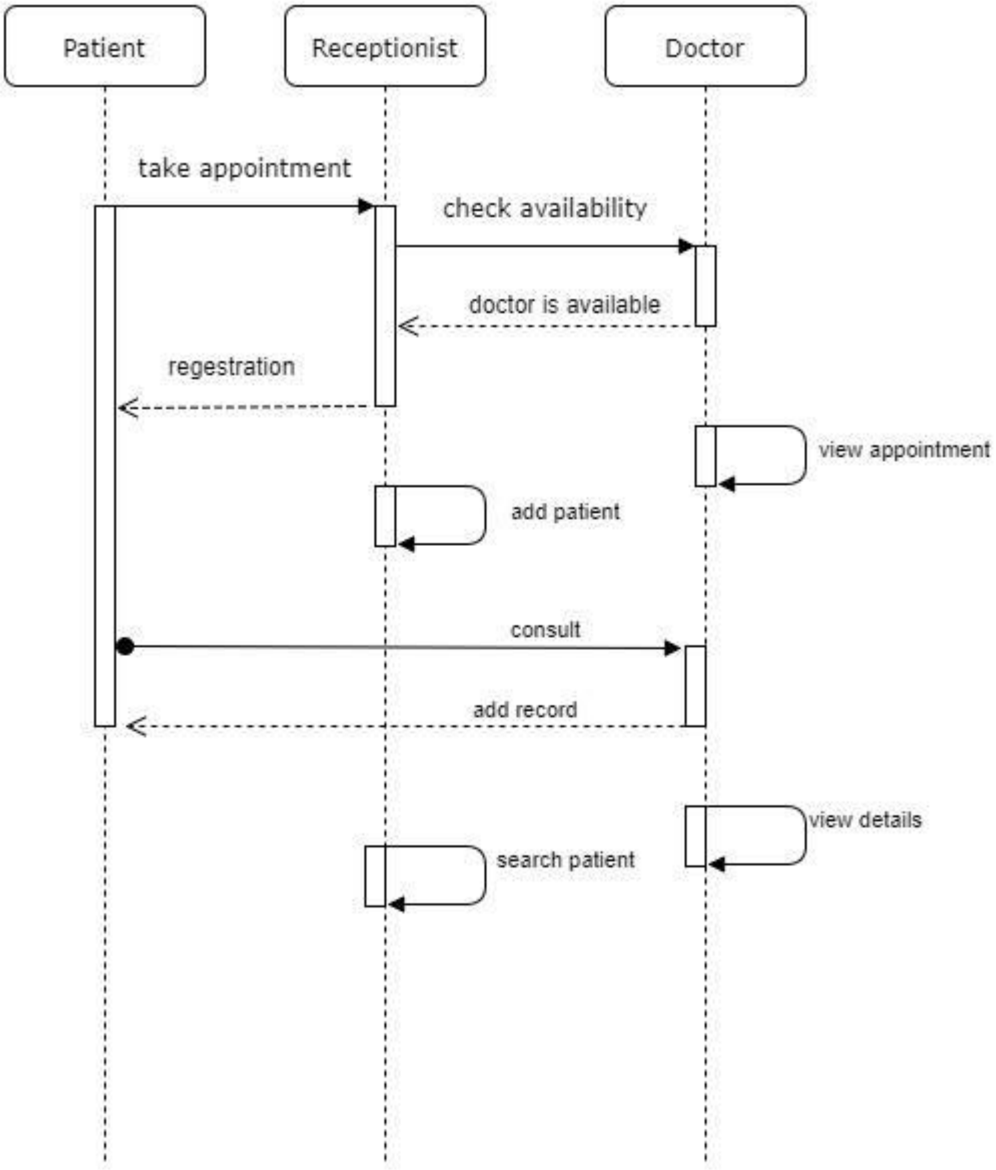
-MAINTAINABILITY: The ability to maintain, modify information and update fix problems of the system.

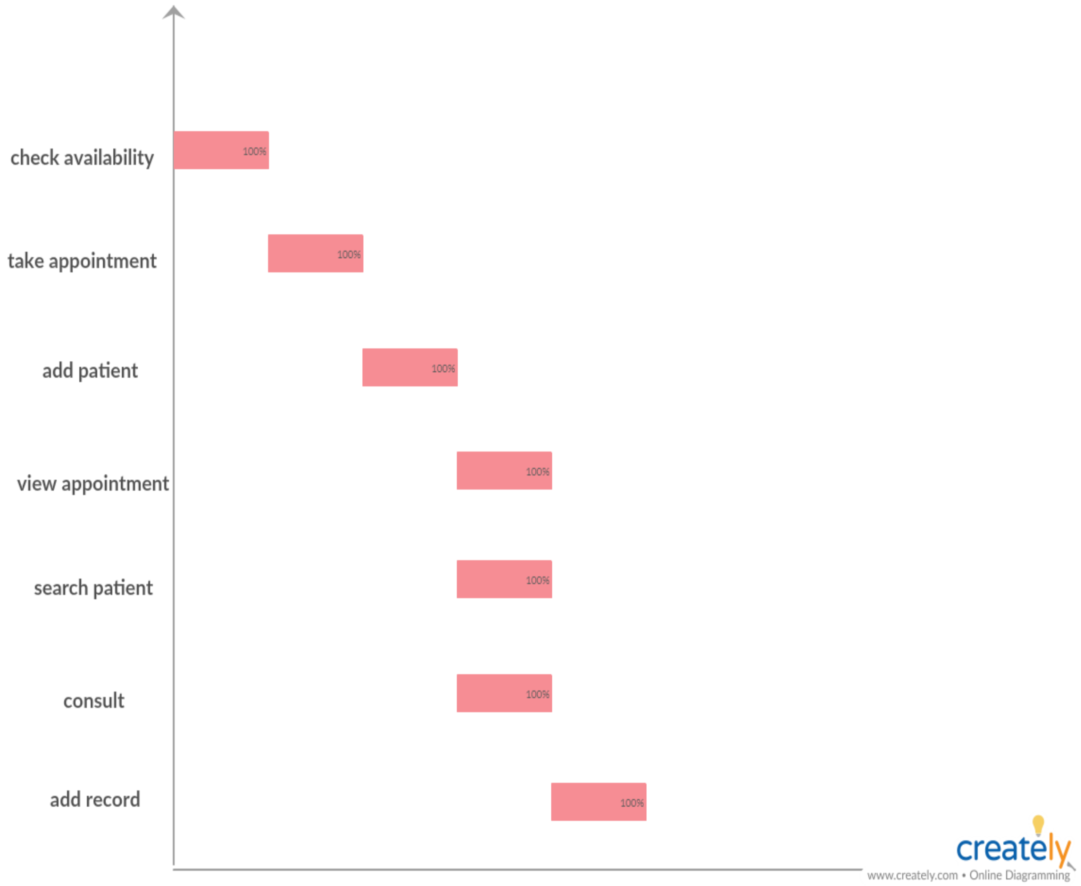
-USABILITY: software can be used again and again without distortion.

-ACCESSIBILITY: Administrator and many other users can access the system but the access level is controlled for each user according to their work scope.

-ACCURACY: The reliability on the information/output. Can depend/be sure of the outcome.

-STABILITY: The system outcome/output won't change time to time. Same output will be given always for a given input.


Patient
Receptionist
Doctor
take appointment
check availability
doctor is available
regestration
view appointment
add patient
consult
add record
view details
search patient


check availability
take appointment
add patient
view appointment
search patient
consult
add record
100%
100%
100%
100%
100%
100%
100%
creately
www.creately.com • Online Diagramming