
System Requirements Specification

for

Hospitome

Prepared by

Jaagrit Arora (20103021)

Kanika Kaur (20103057)

Tabish Khan (20103070)

Chhabra Utkarsh (20103084)

Table of Contents

1. Introduction

- 1.1 Purpose
- 1.2 Intended Audience
- 1.3 Project Scope

2. Overall Description

- 2.1 Product Perspective
- 2.2 Product Features
- 2.3 User Classes and Characteristics
- 2.4 Design and Implementation Constraints

3. System Features

- 3.1 System Feature 1
- 3.2 System Feature 2
- 3.3 System Feature 3

4. External Interface Requirements

- 4.1 User Interfaces
- 4.2 Hardware Interfaces
- 4.3 Software Interfaces
- 4.4 Communications Interfaces

5. Other Nonfunctional Requirements

- 5.1 Availability
- 5.2 Accuracy
- 5.3 Reliability
- 5.4 Portability

Appendix A: Analysis Models

1. Introduction

This document is prepared in order to determine the software requirement specification for Hospitome - the at-home disease prediction system. In remote areas, the penetration of hospitals and doctors is low. This project aims to create a solution to increase the reach of medical facilities. It is a system that can easily detect a range of diseases by taking user input in the form of medical reports, test results & patient details from the user and give reliable results using machine learning models. In order to gain an overview about the report, firstly the purpose and scope of this document will be given, followed by an overall description of the Hospitome System. In addition to these, system features such as uploading reports, disease prediction system and booking appointment are described deeply.

1.1 Purpose

The SRS is needed to evolve as the development of the software product processes. The purpose of this document is to give a complete description about how the Hospitome System can be developed. This document is to provide information about what the software product is to do to help patients and doctors. In addition to these, it provides a basis for validation and verification.

1.2 Intended Audience

This document is intended for the developers working on this project and the college instructors to evaluate the project.

1.3 Project Scope

The name of the software is Hospitome System. Hospitome is a website that predicts a patient's diagnosis among a range of diseases based on the medical records of the patient & then facilitates interaction with a specialist of the disease by booking an appointment. The aim of this system is to facilitate at-home healthcare services and make the system efficient and quick. The users can be of two types, patients and doctors. The patients can use the system to upload their medical reports and receive a reliable result based on the well-trained ML models. They can then book a consultation with the disease specialist who will be able to access the patient's current and previous records.

2. Overall Description

In this section, background information about what type of requirements the system should have will be provided briefly.

2.1 Product Perspective

Hospitome System is an independent website which patients and doctors can use as an application of Machine Learning to diagnose diseases with the click of a button . It is not a part of a larger system, it is an independent system.

2.2 Product Features

After creating an account, the first thing the patient needs to do is upload their medical records and reports. Then the model predicts whether the patient suffers from the particular disease or not. Patient details are stored for future reference & consultations. If the test result is positive for the disease, the patient is notified along with the details of the specialist so that an appointment can be booked, based on the doctor's availability. The predicted results are shared with the specialist as well as the patient. The doctor's prescriptions are stored in the database.

2.3 User Classes and Characteristics

Hospitome System doesn't require any specific computer knowledge to use it except the developers and the administrators of it. Standard users are thought to be from any age, any gender, any nationality who can just use the computer's browser.

On the other hand, administrators and potential developers need a high level of expertise to understand technologies.

2.4 Design and Implementation Constraints

Health is something that can affect not only a person's life but also the lives of people around him/her and by using ML algorithms, there might be cases where a disease is misdiagnosed and false positives or false negatives are recorded. The Hospitome System solves this problem by using a highly accurate disease recognition system that can correctly identify diseases with the help of symptoms and its reliable algorithms.

3. System Features

This section describes the key features and services provided by the product.

3.1 System Feature 1

Check Patients Disease Status.

3.1.1 Description

The user can Upload their reports and symptoms for evaluation.

3.1.2 Stimulus/Response Sequences

The user opens the website and clicks the **Testing** button.

3.1.3 Functional Requirements

html, css, js, firebase, python

3.2 System Feature 2

Find a Suitable Doctor for an appointment.

3.1.1 Description

The user can search a particular specialist for the disease in the Doctor's record and the corresponding doctor will be shown there.

3.1.2 Stimulus/Response Sequences

The user opens the website and clicks the disease specialist button. Then the user needs to click on the search bar present on the top of the page and enter the disease name there.

3.1.3 Functional Requirements

html, css, js, firebase, python

3.3 System Feature 3

Book Appointments

3.1.1 Description

The patient can book appointments with the doctor and find the suitable cure and prescription for the disease.

3.1.2 Stimulus/Response Sequences

The user can book an appointment using the appointment booking button.

3.1.3 Functional Requirements

html, css, js, firebase, python

4. External Interface Requirements

4.1 User Interfaces

Various interfaces for Hospitome System could be:

1. Home Page
2. Patient Login Page
3. Test Upload And Evaluation Page
4. Appointments Management Page

4.2 Hardware Interfaces

The System must run over the internet, all the hardware is required to be connected to the internet. As for e.g. Modem, WAN-LAN, Ethernet Cross-Cable.

4.3 Software Interfaces

The end user needs a web browser to interact with the system.

4.4 Communications Interfaces

A system with a web browser and stable internet connection is required for the communication purposes.

Communication standards and network communication protocol:

HTTP

HTTPS

FTP

5. Other Nonfunctional Requirements

5.1 Availability

The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs. In case of a hardware failure or database corruption, a replacement page will be shown. Also in case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the administrator. Then the service will be restarted. It means 24 X 7 availability.

5.2 Accuracy

The system should be accurate in identifying Diseases and there should be no Diseases recorded incorrectly. Diseases Detection is a very crucial process and it needs to be done accurately as it can affect a person's life.

5.3 Reliability

The system provides storage of all databases on redundant computers with automatic switch over. The reliability of the overall program depends on the reliability of the separate components. The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes. Thus the overall stability of the system depends on the stability of the container and its underlying operating system.

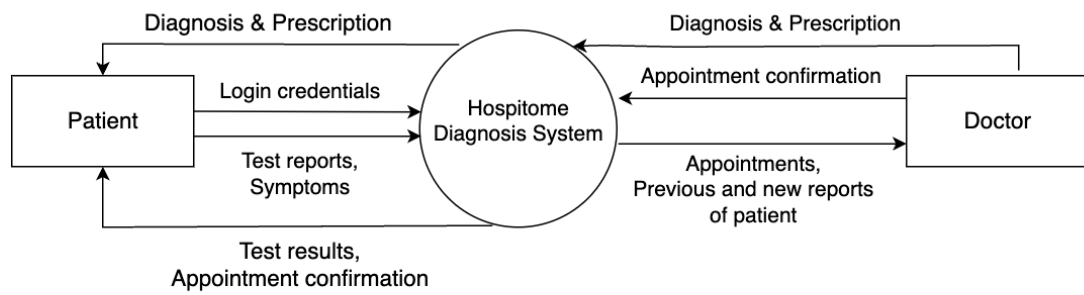
5.4 Portability

The application is HTML and scripting language based. So that end user part is fully portable and any system using any web browser should be able to use the features of the system, including any hardware platform that is available or will be available in the future. An end-user can use this system on any OS; either it is Windows or Linux. The system shall run on PC, Laptops etc.

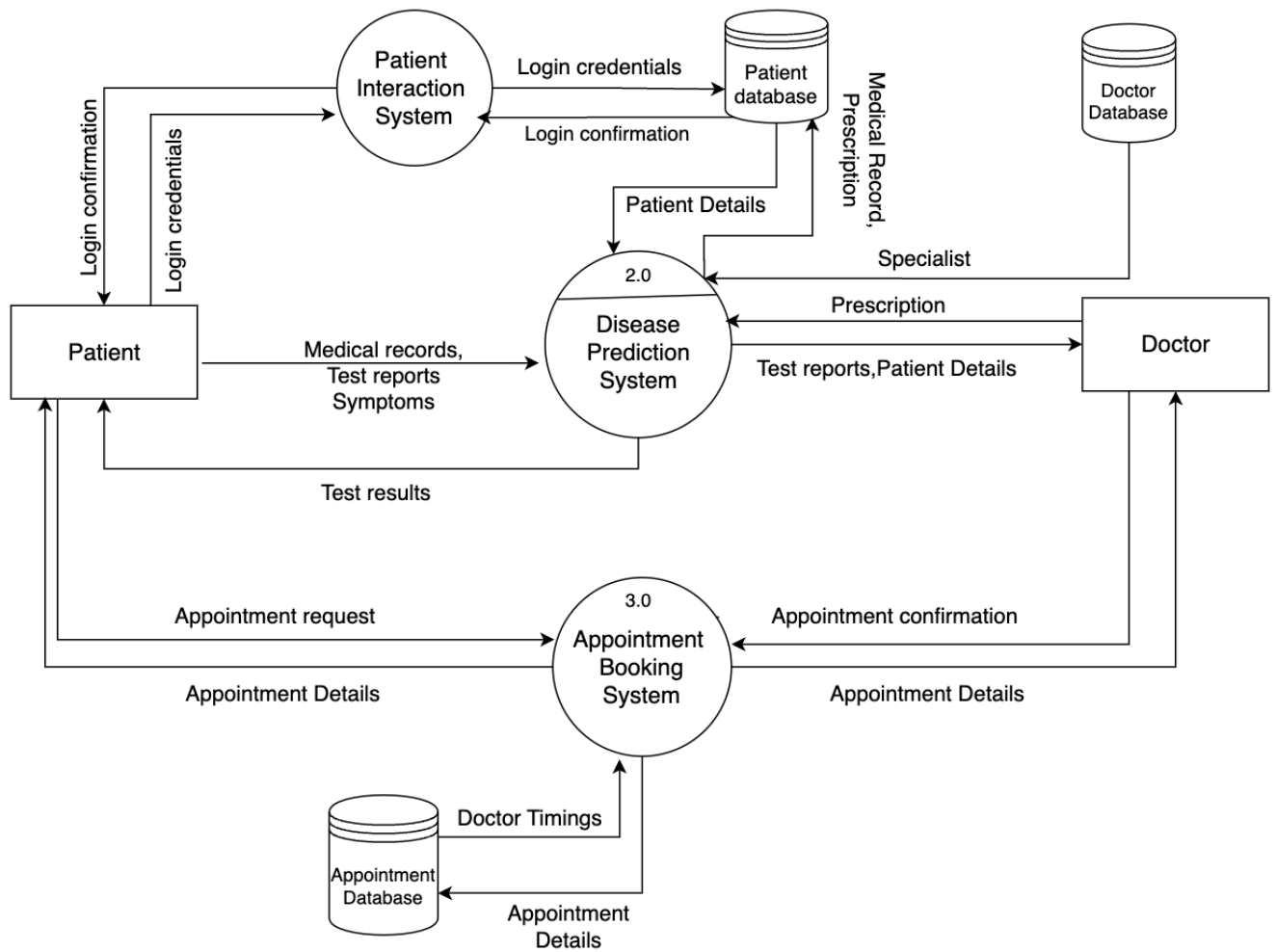
Appendix A: Analysis Models

A.1 Level Zero Data Flow Diagram

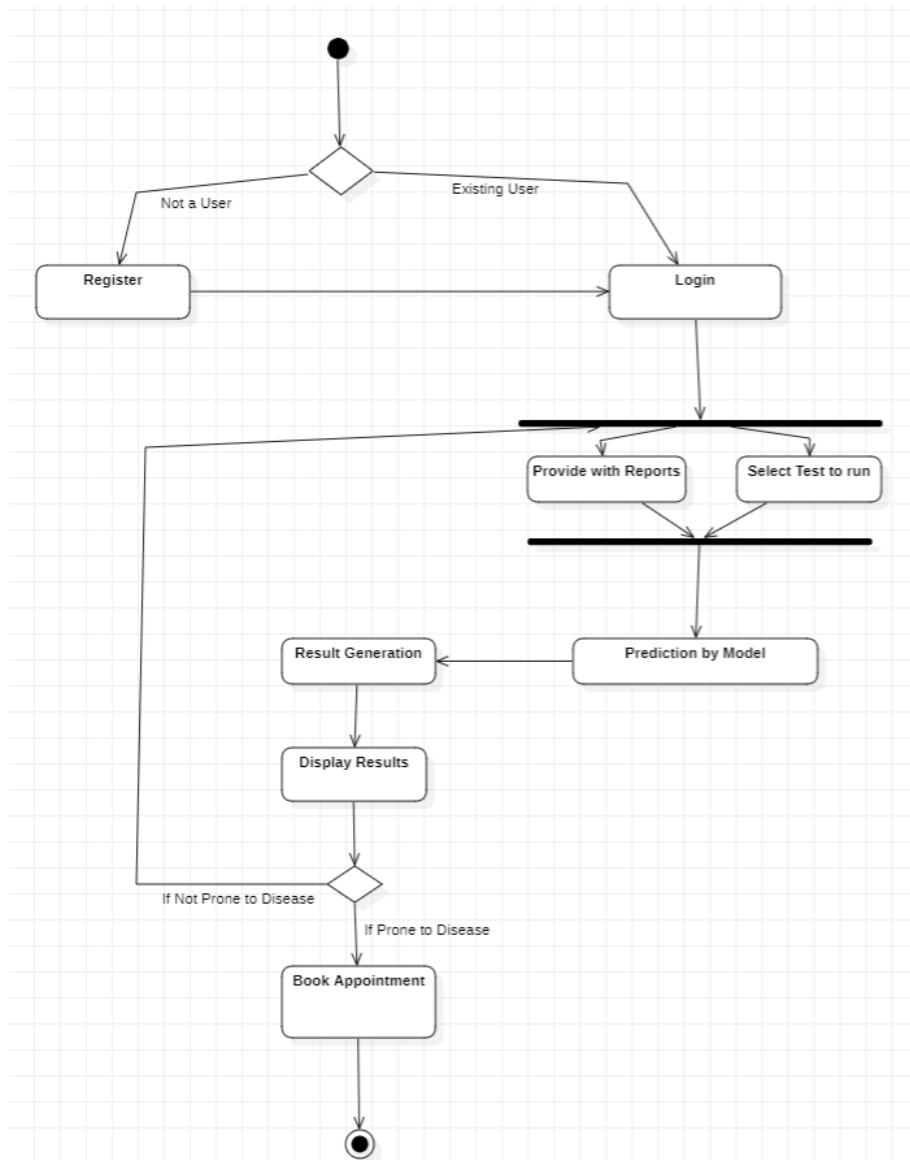
LEVEL 0



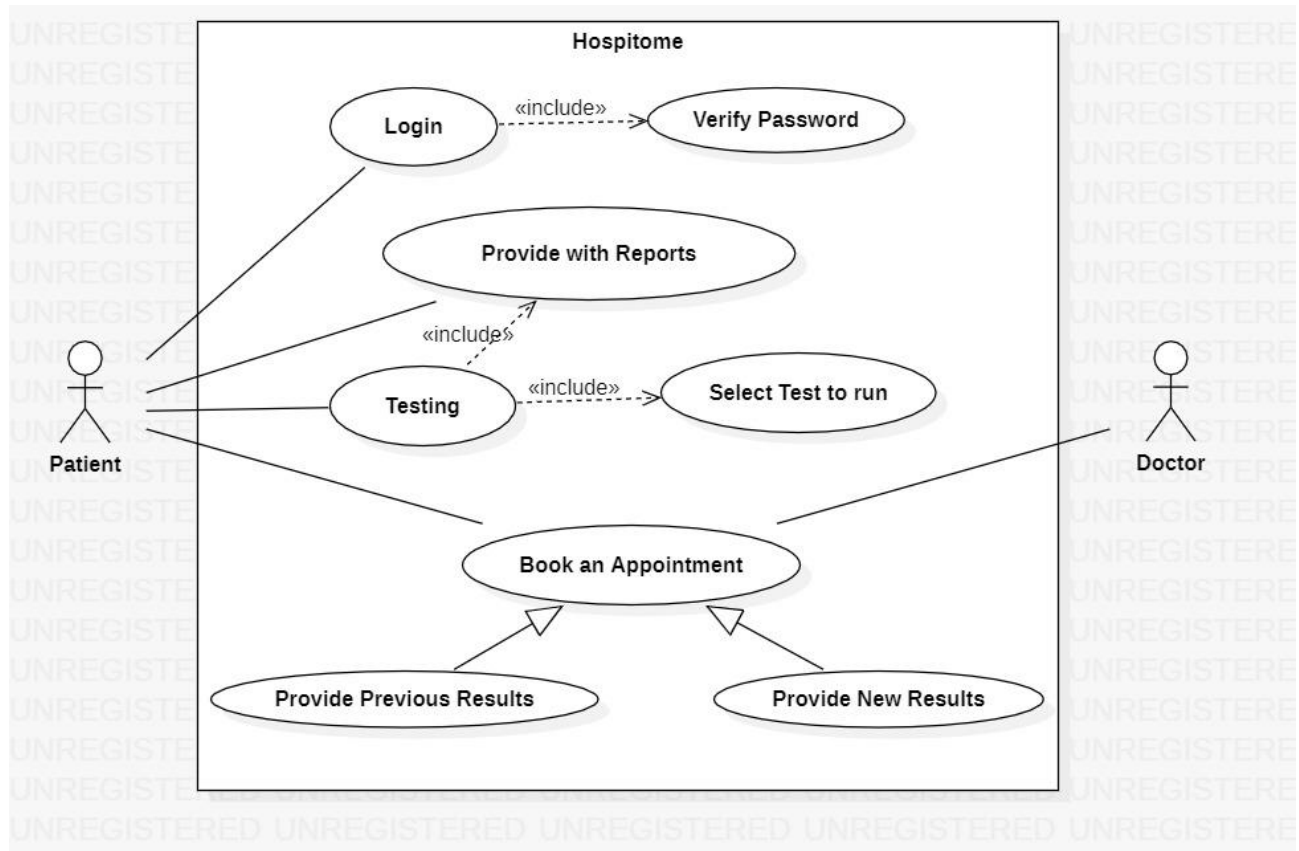
A.2 Level One Data Flow Diagram



A.3 Activity Diagram



A.4 Use Case Diagram



A.5 Class Diagram

