

# **PHARMA GUIDE (SRS)**

## **Group Members**

Rahul Sahu    1705157  
Rishika Sinha   1705159

## **Under Guidance of**

Dr. Suresh C. Sathapathy

## **CONTENTS**

### **1. INTRODUCTION**

#### **1.1 PURPOSE**

#### **1.2 PROJECT SCOPE**

#### **1.3 INTENDED AUDIENCE**

### **2. OVERALL DESCRIPTION**

#### **2.1 PRODUCT PERSPECTIVE**

#### **2.2 USER CLASSES AND CHARACTERISTICS**

#### **2.3 OPERATING ENVIRONMENT**

#### **2.4 ASSUMPTIONS AND DEPENDENCIES**

### **3. SYSTEM FEATURES**

#### **3.1 FUNCTIONAL REQUIRMENTS**

### **4. EXTERNAL INTERFACE REQUIREMENTS**

#### **4.1 USER INTERFACE**

#### **4.2 SOFTWARE INTERFACE**

#### **4.3 HARDWARE INTERFACE**

### **5. NON-FUNCTIONAL REQUIREMENTS**

#### **5.1 PERFORMANCE REQUIREMENTS**

#### **5.2 SAFETY REQUIREMENTS**

#### **5.3 SECURITY REQUIREMENTS**

## **INTRODUCTION**

### **1.1 PURPOSE**

The purpose of this project is to build a virtual doctor to predict the disease based on the symptoms and recommend a drug for the disease. Top suitable drugs based on review and rating are displayed to the user. **The major purpose is to crosscheck doctor's prescription and diagnosis. And to diagnose some minor disease.**

### **1.2 PROJECT SCOPE**

The purpose of the project is to create a convenient and easy-to-use application for patients. The prediction model is based on a kaggle database which has drug name with the disease and symptoms. We will have a database server supporting the major cities in India. Above all, we hope to provide a comfortable user experience and to extend the scope of this project across the world.

### **1.3 INTENDED AUDIENCE**

Our target audience are the people who are showing symptoms of illnesses that can be cured at home by taking either medicine or precautions. We let people know about their illness and whether they need to see a doctor. The patients have the convenience of checking their symptoms from anywhere like home, workplace etc.

## **OVERALL DESCRIPTION**

### **2.1 PRODUCT PERSPECTIVE**

Our database stores the following information:

1. Symptoms of diseases
2. Drugs available for diseases
3. Rating and review of customers for the drugs

### **2.2 USER CLASSES AND CHARACTERISTICS**

The system is to support the User who has abnormal symptoms and might be suffering from a disease. Customers will have access to customer functions. The customer should be able to do the following functions:

1. Check if they are suffering from a disease by verifying their symptoms and the probability of having the disease.
2. The patients can enter the disease and know the top drugs for the disease based on rating and review.

### **2.3 OPERATING ENVIRONMENT**

- Web Browser

## **SYSTEM FEATURES**

### **3.1 FUNCTIONAL REQUIRMENT**

- Predict disease by symptoms.
- Find Medicine by disease.
- Contact to admin.

## **INTERFACE REQUIREMENTS**

### **5.1 USER INTERFACE**

- Front End - HTML, CSS, JAVASCRIPT
- Back End- Python,Mongo DB, flask

### **5.2 SOFTWARE INTERFACE**

- OPERATING SYSTEM: Any OS.
- Database: MongoDB.
- Language: Python (flask).

### **5.3 HARDWARE INTERFACE**

- 1 GB RAM or more (55 MB needed for app execution)
- 2.4GHz Processor or more.
- 4 GB RAM
- Stable Internet
- Screen Resolution : Any (16:9 for best view)

## **NON-FUNCTIONAL REQUIREMENTS**

### **6.1 PERFORMANCE REQUIREMENT**

#### **DATA CLEANING**

Data cleaning is one of the important parts of machine learning. It plays a significant part in building a model. Data Cleaning is one of those things that everyone does but no one really talks about. It surely isn't the fanciest part of machine learning and at the same time, there aren't any hidden tricks or secrets to uncover. However, proper data cleaning can make or break your project. Professional data scientists usually spend a very large portion of their time on this step.

#### **Steps involved in Data Cleaning**



### **6.2 SAFETY REQUIREMENT**

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 (typically tape) and reconstructs a more current state by reapplying or redoing the operations up to the time of failure.