Software Requirements Specification

for

AROGYAM : The Disease Prediction App

Government Engineering College, Wayanad

GROUP NO: 15
15 ASEEMMUHAMMAD A R
43 NAZEEB ABDULLA
59 VISHNU K T
65 JISHNU K SANJEEVAN

November 9, 2019

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to give a detailed description of “AROGYAM : The Disease Prediction App”. This document will explain purpose and features of the system and identifies its users. It is intended for both the designers, developers as well as end-users of the system.

1.2 Scope

’AROGYAM’ is a disease prediction system consisting of two mobile applications(Primarily focusing on the android platform), web applications and a machine learning model that improves the prediction by pre-collected data. The system will be designed to :

- Collect data from the user.
- Collect data by the Asha workers.
- Monitor data for the admin.
- Predict the individuals vulnerable to Tuberculosis.

The product allows the user to check their health condition and admin to monitor the users.

1.3 Document Conventions and Acronyms

Term	**Defenition**
Product / Software	Product and software is used interchangeably to denote the ’AROGYAM’ application
User	A person who checks his/her health condition
Data	The health conditions of a user.
Disease	Tuberculosis
ML	Machine Learning
DMO	District Medical Officer
App	Application
Patient / Subject	A person diagnosed with Tuberculosis

1.4 References

[1] Ajay Kumar Shrivastava, Akash Rajak, and Shashank Bhardwaj. Detection of tuberculosis based on multiple parameters using anfis. *2018 3rd International Innovative Applications of Computational Intelligence on Power, Energy and Controls with their Impact on Humanity (CIPECH)*, 2018.

1.5 Overview

Section 2 of this document gives an overall description about the product. It describes the general functionalities, expected user groups, constraints of the product. Section 3 gives more specific information about the functionalities specified in Section 2. Section 2 is a general view of the system and should be used as a guidance to section 3. Section 3 is intended for developers and testers and may be skipped by end users.

2 Overall Description

2.1 Product Perspective

This product is envisaged as a complete solution to support the efforts of the Govt of Kerala to eradicate Tuberculosis from Wayanad The application will help the stake holders to manage socio-economic data of the residents of wayanad and predict the possibility of Tuberculosis. It is an independent and self-contained solution. To the best of our knowledge there exists no other application that performs the intended task of this app.

2.1.1 User Interfaces

The User app and Survey app will have registration, login and data collection forms. User interfaces are made as user-friendly as possible with optimal use of multile choices, check boxes and drop down lists. Dialog boxes are used for providing info and alerts.

2.1.2 Software Interfaces

This project needs the following services and software.

1. Python 3.7 or Higher.
   Python is the main programming language we will be using in our project because it provides all the functionality we need.
   Source: https://www.python.org/

2. NumPy
   NumPy is the fundamental package for scientific computing with Python.
   Source: https://numpy.org/

3. Pandas
   Pandas is an open-source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language.
   Source: https://pandas.pydata.org/

4. TensorFlow
   The core open-source library to help you develop and train ML models.
   Source: https://www.tensorflow.org/

5. Firebase
   Firebase is a Back end-as-a-Service — BaaS. We use it to build our database.
   Source: https://firebase.google.com/

6. Google Cloud Platform
   It is a suite of public cloud computing services offered by Google. The platform includes a range of hosted services for compute, storage and application development that run on Google hardware.
   Source: https://cloud.google.com/

7. Django
   Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. We use it for creating the server code in python.
   Source: https://www.djangoproject.com/

8. HTML
   It is the standard markup language for Web pages. We use it for creating our structure of web application.
   Source: https://www.w3schools.com/html/

9. CSS
   It is a language that describes the style of an HTML document. In our project we use it to style the web application such that it will be appeal and easy to understand by the admins.
   Source : https://www.w3schools.com/css/

10. JavaScript
    An object-oriented computer programming language commonly used to create interactive effects within web browsers. We used it in our project for client side scripting.
    Source : https://www.w3schools.com/js/

11. Android Studio
    Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. We use it for creating our android application and also linking to various services to application.
    Source : https://developer.android.com/studio

2.2 Product Functions

This section outlines the use cases of the user. Specific functionalities will be discussed in section 3.

2.2.1 User Registration

The user must register first, to use the application.

2.2.2 User Login

Registered users can log in with their Mobile no./Email and pin to the app.

2.2.3 Health Checkup - Data Entry

The user feeds data to the applications to check the health. The user data is saved permanently in a database so that it can be used to run the scoring or the machine learning algorithm for predicting the health conditions of the user. The data can be updated later by the user himself or the Asha workers.

2.2.4 Prediction

Based on the user's data, the ML model predicts the health condition of the person and sends the information to health department for further action. The user is referred to the nearby hospital.

2.2.5 Admin Login

The admin can log in to their web apps using their username and password.

2.2.6 Asha Worker registration

The admins can register an account for each Asha worker for the survey app. The Asha workers will receive their credentials to login to the system.

2.2.7 Survey Login

The preregistered Asha workers can log in to the survey app.

2.2.8 Health Checkup - Data Collection

Asha workers can use the survey app to collect survey data.

2.2.9 Leaderboard

Administrators and doctors can monitor the users / subjects using the leader board. Statistical analysis of the user data is also provided in the leader board.

2.3 User Characteristics

The users of this system are those who wish to check their health condition, conduct a survey on health issues and those who monitor the health conditions. The user must have the basic knowledge to use mobile phones. Three types of users are

1. Residents of Wayanad
   These are the major user group for 'AROGYAM' app. They are the naive users who have only Login, Health Checkup- Data Entry operations and to view the result.

2. Asha Workers
   Asha Workers are the ones who use the Survey app. The preregistered Asha workers have login , Health Checkup- Data Collection operations.

3. Admin
   Admins are the doctors, health inspectors, the district medical officers of Wayanad. They can monitor both the user data and survey data statistically. They are the only user who can view the result of ML model prediction.

2.4 Constraints

The user of this application must be restricted to Wayanad District.

2.5 Assumptions and Dependencies

The residents of Wayanad are informed about the application through media. The accuracy of the prediction is dependent on the honesty of the users and the correctness of the data collected by the Asha workers.

2.6 Apportioning of requirements

The following requirements should be provided as increments on future versions and need not be included in the first release. This is to ensure accuracy of the system. No more specific details for these requirements will be provided in section 3.

1. Machine Learning Model For Disease Prediction
   To develop an ML Model for Disease prediction can only be developed using a large amount of user data, which is currently unavailable. So, until sufficient data is not obtained, the ML model cannot be fully relied upon.

3 Specific Requirements

3.1 Functional Requirement Specification

This section describes each functionality and their flow of events in detail. The actor in each use case is always the ’User’, unless otherwise specified.

3.1.1 User registration

Use case name	:	Registration of new user
Objective	:	To obtain basic details of the user and to ensure the validity of the user
Priority	:	High
Precondition	:	An android phone with an internet connection
Postconditions	:	Valid Email address/Mobile Number
Flow of Events	:	1. Basic Flow

1.1. The user clicks the *Register* button in the home page

1.2. The application then opens a *data collection form* that collects basic data required for user validation

1.3. The user clicks the *Register* button

1.4. The application validates the user data, if invalid - asks the user to enter valid data using an alert box

1.5. If valid, user is taken back to the home page and asked to login

3.1.2 User Login

Use case name	:	Login of registered user
Objective	:	To validate and identify a registered user
Priority	:	High
Precondition	:	An android phone with an internet connection
PostConditions	:	Registered Email address/Mobile Number and Pin
Flow of Events	:	1. Basic Flow

1.1. The user clicks the *Login* button

1.2. The user is taken to the login page

1.3. The user enters the registered Email address/Mobile Number and PIN

1.4. The user clicks on *Login* button

1.5. The application validates the Login credentials and if invalid - asks the user to enter valid login details

1.6. If valid, then the user is successfully logged and can use the application

3.1.3 Data Entry

Use case name : Data entry by user
Objective : The user enters data to check their health condition
Priority : High
Precondition : An android phone with an internet connection and user login
Flow of Events :

1. Basic Flow

    1.1. The user selects clicks the *Health Checkup* button

    1.2. Application shifts to a new page, consisting of *data collecting form*

    1.3. The user fills the *form* and submits it

    1.4. The form data is validated if invalid - asks the user to enter valid data

    1.5. If valid, data is stored in the database for Admin to monitor and ML model to predict

    1.6. The data is then processed by the ML model, and if disease is predicted, the result is sent to the admin and the user is referred to a nearby hospital

3.1.4 Admin Login

Use case name : Admin Login
Objective : To validate and identify an admin
Priority : High
Precondition : A Computer with an internet connection
Flow of Events :

1. Basic Flow

    1.1. Admin enters enters the login credentials - Email Id/Mobile Number and password and clicks the *Login* button

    1.2. Admin login credentials are checked with the predefined admin credentials

    1.3. If verified, admin can proceed and view the user data statistics else admin is requested to enter valid login details

3.1.5 Asha Worker Registration

Use case name : Registration of Asha Workers for Survey app
Objective : To register Asha Workers into Survey app
Priority : High
Precondition : Computer with an internet connection
Flow of Events :

1. Basic Flow

    1.1. Admin logins to the web app

    1.2. Clicks the *Register Asha Worker* button

    1.3. The details and login credentials of Asha Worker is entered to the web app and is stored in the database

3.1.6 Asha Worker Login

Use case name : Login of registered Asha Worker into the Survey app
Objective : To validate and identify an registered Asha Worker
Priority : High
Precondition : An android phone with an internet connection
PostConditions : Registered Email address/Mobile Number and Pin
Flow of Events : 1. Basic Flow

1.1. Asha Worker clicks the *Login* button

1.2. App changes to the login page

1.3. Registered Email address/Mobile Number and PIN is entered

1.4. The *Login* button is clicked

1.5. The application validates the Login credentials and if invalid - asks the user to enter valid login details

1.6. If valid, then the Asha worker is successfully logged and can use the application

3.1.7 Data Collection

Use case name : Data collection by Survey App
Objective : The Asha Worker enters data collected from people to check their health condition
Priority : High
Precondition : An android phone with an internet connection and user login Flow of Events : 1.

3.1.8 Data Statistics

Use case name	:	Data Statistics View
Objective	:	The admin can view data statistics so that he can monitor the users effectively
Priority	:	High
Precondition	:	Computer with an internet connection

Flow of Events :

1. Basic Flow

    1.1. Admin logins to the web app

    1.2. Admin clicks the *View User Data* button

    1.3. The user data is retrieved from the database and is processed to represent in a statistics

    1.4. Data is shown in statistics

3.1.9 Disease Prediction

Use case name	:	Disease Prediction
Objective	:	To predict the disease using ML model and user entered data
Priority	:	High
Precondition	:	Valid user data

Flow of Events :

1. Basic Flow

    1.1. The user logins to the app

    1.2. The user enters data

    1.3. The user clicks *Submit* button

    1.4. The data is stored in the database and the ML model processes the data

    1.5. If the disease is predicted, the result is sent to the admin and the user is referred to a nearby hospital

    1.6. If no disease is predicted, the user is notified that he/she is okay

3.2 Non-functional Requirements

3.2.1 Performance

The mobile applications, web app, and database must be fast. No more than 2 seconds should be taken. The processing of machine learning model can be slow but not more than 5 seconds.

3.2.2 Portability

The product is not portable to other platforms.