Software Requirements Specification

Department of Information Technology

Faculty of Engineering

MEDI-CAPS UNIVERSITY, INDORE- 453331

Team Members -

1. **Nimra Khan**
2. **Aniket Manke**
3. **Anjalee Jaiswal**

Project-

Heart Disease Prediction System

Table of Contents

1. Introduction

The health care industries collect huge amounts of data that contain some hidden information, which is useful for making effective decisions. For providing appropriate results and making effective decisions on data, some advanced data mining techniques are used. In this study, a Heart Disease Prediction System (HDPS) is developed using Logistic Regression and Decision Tree algorithms for predicting the risk level of heart disease. The system uses 15 medical parameters such as age, sex, blood pressure, cholesterol, and obesity for prediction. The HDPS predicts the likelihood of patients getting heart disease. It enables significant knowledge. E.g. Relationships between medical factors related to heart disease and patterns, to be established. We have employed the multilayer perceptron neural network with backpropagation as the training algorithm. The obtained results have illustrated that the designed diagnostic system can effectively predict the risk level of heart diseases.

1.1 Purpose

This SRS document details the software requirement for Heart Disease Prediction System.
The Purpose of this document is to verify that all the specification are correct and are verified.
It is intended for the Project Team , Business Owner , Teacher Supervising thisProject ,and all other Stack-holder's.

1.2 Scope

Scope of this Project-
Provides new approach to concealed patterns in the data.
- Helps avoid human biasness.

- To implement Naïve Bayes Classifier that classifies the disease as per the input of the user.
- Reduce the cost of medical tests.

1.3 References

- https://archive.ics.uci.edu/ml/datasets/heart+Disease
- https://www.kaggle.com/ronitf/heart-disease-uci

1.4 Overview

Section 2 is for Potential User or Employee's.
Section 3 is for Developer's.

1.5 Definitions, acronyms, and abbreviations

- *age: The person's age in years*

- *sex: The person's sex (1 = male, 0 = female)*

- *cp: The chest pain experienced (Value 1: typical angina, Value 2: atypical angina, Value 3: non-anginal pain, Value 4: asymptomatic)*

- *trestbps: The person's resting blood pressure (mm Hg on admission to the hospital)*

- *chol: The person's cholesterol measurement in mg/dl*

- *fbs: The person's fasting blood sugar (> 120 mg/dl, 1 = true; 0 = false)*
restecg: Resting electrocardiographic measurement (0 = normal, 1 = having ST-T wave abnormality, 2 = showing probable or definite left ventricular hypertrophy by Estes' criteria)

- *thalach: The person's maximum heart rate achieved*

- *exang: Exercise induced angina (1 = yes; 0 = no)*

- *oldpeak: ST depression induced by exercise relative to rest*

- *slope: the slope of the peak exercise ST segment (Value 1: upsloping, Value 2: flat, Value 3: downsloping)*

- *ca: The number of major vessels (0-3)*

- *thal: A blood disorder called thalassemia (3 = normal; 6 = fixed defect; 7 = reversable defect)*

- *target: Heart disease (0 = no, 1 = yes)*

1.6 Assumptions and dependencies

- We assume that sensors will be able to give data frequently.
- We assume that sensors' accuracy will be enough to meet our needs.
- Computing power of the fixed device will be adequate.
- Internet connection will be available for our system to work.
- Users will have a mobile phone compatible with the app.
- Users will have WiFi connection in their homes.

2. The Overall Description

Clinical decisions are often made based on doctor's insight and experience rather than on the knowledge rich data hidden in the dataset. This practice leads to unwanted biases, errors and excessive medical costs which affects the quality of service provided to patients. The proposed system will integrate clinical decision support with computer-based patient records (Data Sets). This will reduce medical errors, enhance patient safety, decrease unwanted practice variation, and improve patient outcome. This suggestion is promising as data modeling and analysis tools, e.g., data mining, have the potential to generate a knowledge rich environment which can help to significantly improve the quality of clinical decisions.
There are voluminous records in medical data domain and because of this, it has become necessary to use data mining techniques to help in decision support and prediction in the field of healthcare. Therefore, medical data mining contributes to business intelligence which is useful for diagnosing of disease.

2.1 Product Perspective

In this system we are implementing effective heart attack prediction system using Logistic Regression algorithm. We can give the input as in CSV file or manual entry to the system. After taking input the algorithms apply on that input that is Logistic Regression. After accessing data set the operation is performed and effective heart attack level is produced.

- The proposed system will add some more parameters significant to heart attack with their weight, age and the priority levels are by consulting expertise doctors and the medical experts.
- The heart attack prediction system designed to help the identify different risk levels of heart attack like normal, low or high and also giving the prescription details with related to the predicted result.

2.1.1 Hardware Interfaces

This Project will run on any device (Mobile,Tablet,Laptop,Desktop ,etc) which support Browser and steady internet connection.

2.1.2 Operations

Specify the normal and special operations required by the user such as:
(1) Data Preprocessing.

Importing Dependencies
(2) Exploratory Data Analysis.
(3) Model Creation.
(4)Finding accuracy and making predictions.

3. Specific Requirements

3.1 HARDWARE REQUIREMENTS

v System : Pentium 4, Intel Core i3, i5, i7 and 2 GHz Minimum

v RAM : 512Mb or above

v Hard Disk : 10 GB or above

v Input Device : Keyboard and Mouse

v Output Device : Monitor or PC

3.2 SOFTWARE REQUIREMENTS

v Operating System : Windows 7, 10 or Higher Versions

v Platform : Jupiter Notebook

v Front End : Python

v Back End : Python and Files

Programming Lang : Python

3.3 Functional Requirement

A Functional requirement defines a function of a system or its component. A function is described as a set of inputs, the behaviour, and outputs. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describing all cases where the system uses the functional requirements are captured in use cases. Functional requirements are supported by non-functional requirements (also known as quality requirements), which impose constraints on the design or implementation (such as performance requirements, security, or reliability).

As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.So, Functional requirements are statements of the services that the system must provide.

- *Functional Requirements concerns with the specific functions delivered by the system.*

• *The functional requirements of the system should be both complete and consistent*

• *Completeness means that all the services required by the user should be defined.*

• *Consistency means that requirements should not have any contradictory definitions.*

• *The requirements are usually described in a fairly abstract way. However, functional system requirements describe the system function in details, its inputs and outputs, exceptions and so on.*

- *The data is cleaned and preprocessed before it is submitted to the proposed algorithm for training and testing.*
- *work is to predict more accurately the presence of heart disease.*
- *The "goal" field refers to the presence of heart disease in the patient.*
- *0 Represents Person not Having Heart Disease 1 present Person Having Heart disease.*
- *Exploratory Data Analysis Should conclude fruitful results.*
- *Divide our data set into two parts: a training set, which is used to create the model, and a test set, which is used to verify that the model is accurate and not over fitted.*
- *The system should allow administrator to monitor and remove inappropriate datasets and code.*

3.3.1 Non-Functional Requirements-

• *Non-functional Requirements refer to the constraints or restrictions on the system. They may relate to emergent system properties such as reliability, response time and store occupancy or the selection of language, platform, implementation techniques and tools.*

• *The non-functional requirements can be built on the basis of needs of the user, budget constraints, organization policies and etc.*

- *The system should provide values used during prediction to the user.*

- *The system should be responsive and consistent.*

1. *Performance requirement: All data entered shall be up to mark and no flaws shall be there for the performance to be 100%.*
2. *Platform constraints: The main target is to generate an intelligent system to predict the adult height.*
3. *Accuracy and Precision: Requirements are accuracy and precision of the data*
4. *Modifiability: Requirements about the effort required to make changes in the software. Often, the measurement is personnel effort (person- months).*
5. *Portability: Since mobile phone is handy so it is portable and can be carried and used whenever required.*
6. *Reliability: Requirements about how often the software fails. The definition of a failure must be clear. Also, don't confuse reliability with availability which is quite a different kind of requirement. Be sure to specify the consequences of software failure, how to protect from failure, a strategy for error Prediction, and a strategy for correction.*
7. *Security: One or more requirements about protection of your system and its data.*
8. *Usability: Requirements about how difficult it will be to learn and operate the system. The requirements are often expressed in learning time or similar metrics.*

ACCESSIBILITY:

Accessibility is a general term used to describe the degree to which a product, device, service, or environment is accessible by as many people as possible. In our project people who have registered with the cloud can access the cloud to store and retrieve their data with the help of a secret key sent to their email ids. User interface is simple and efficient and easy to use.

MAINTAINABILITY:

In software engineering, maintainability is the ease with which a software product can be modified in order to include new functionalities can be added in the project based on the user requirements just by adding the appropriate files to existing project using .net and programming languages. Since the programming is very simple, it is easier to find and correct the defects and to make the changes in the project.

SCALABILITY:

System is capable of handling increase total throughput under an increased load when resources (typically hardware) are added. System can work normally under situations such as low bandwidth and large number of users.

PORTABILITY:

Portability is one of the key concepts of high-level programming. Portability is the software code base feature to be able to reuse the existing code instead of creating new code when moving software from an environment to another. Project can be executed under different operation conditions provided it meet its minimum configurations. Only system files and dependant assemblies would have to be configured in such case.

VALIDATION:

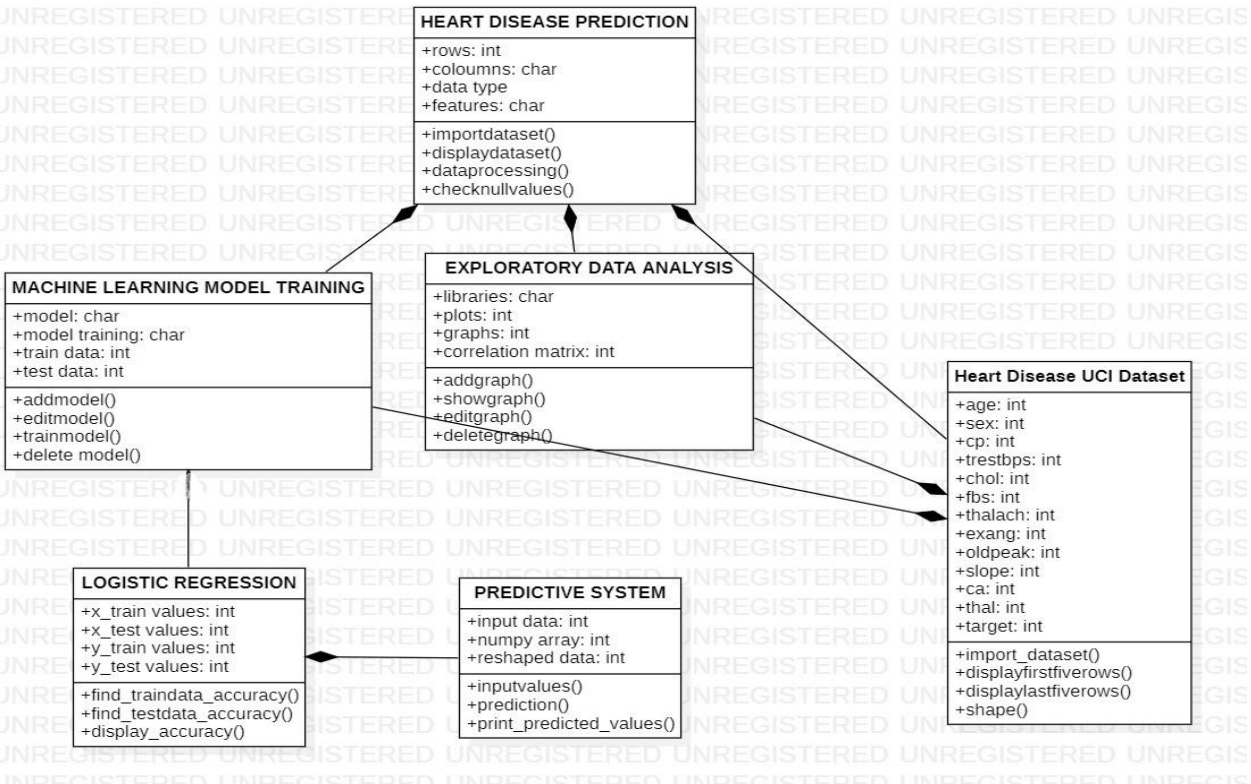
It is the process of checking that a software system meets specifications and that it fulfils its intended purpose. It may also be referred to as software quality control. It is normally the responsibility of software testers as part of the software development lifecycle. Software validation checks that the software product satisfies or fits the intended use (high-level checking), i.e., the software meets the user requirements, not

as specification artefacts or as needs of those who will operate the software only; but, as the needs of all the stakeholders.

3.4 UML Diagrams

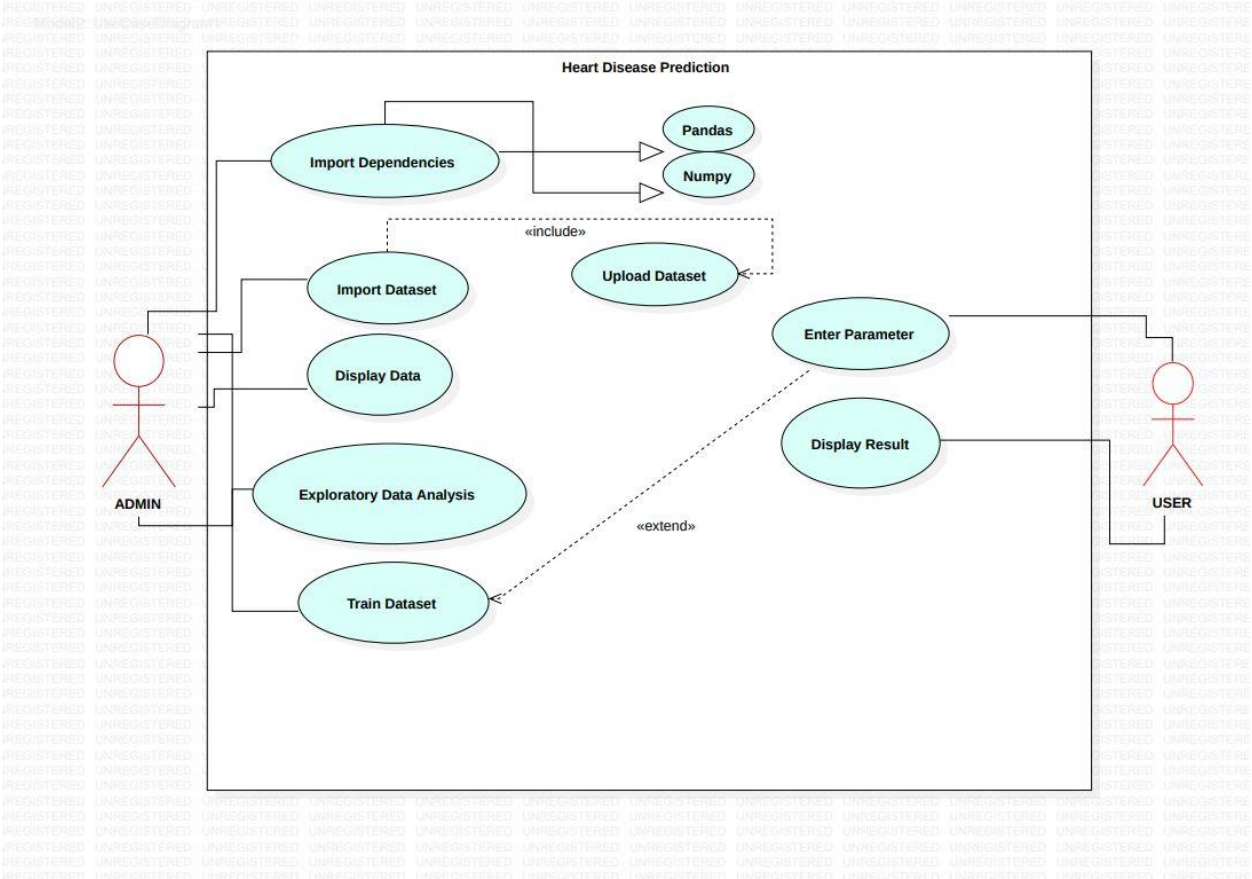
CLASS DIAGRAM

Disease prediction using machine learning consist of class diagram that all the other application that consists the basic class diagram, here the class diagram is the basic entity that is required in order to carry on with the project. Class diagram consist information about all the classes that is used and all the related datasets, and all the other necessary attributes and their relationships with other entities, all these information is necessary in order to use the concept of the prediction, where the user will enter all necessary information such as user name, email, phone number, and many more attributes that is required in order to login into the system and using the files concept we will store the information of the users who are registering into the system and retrieves those information later while logging into the system.



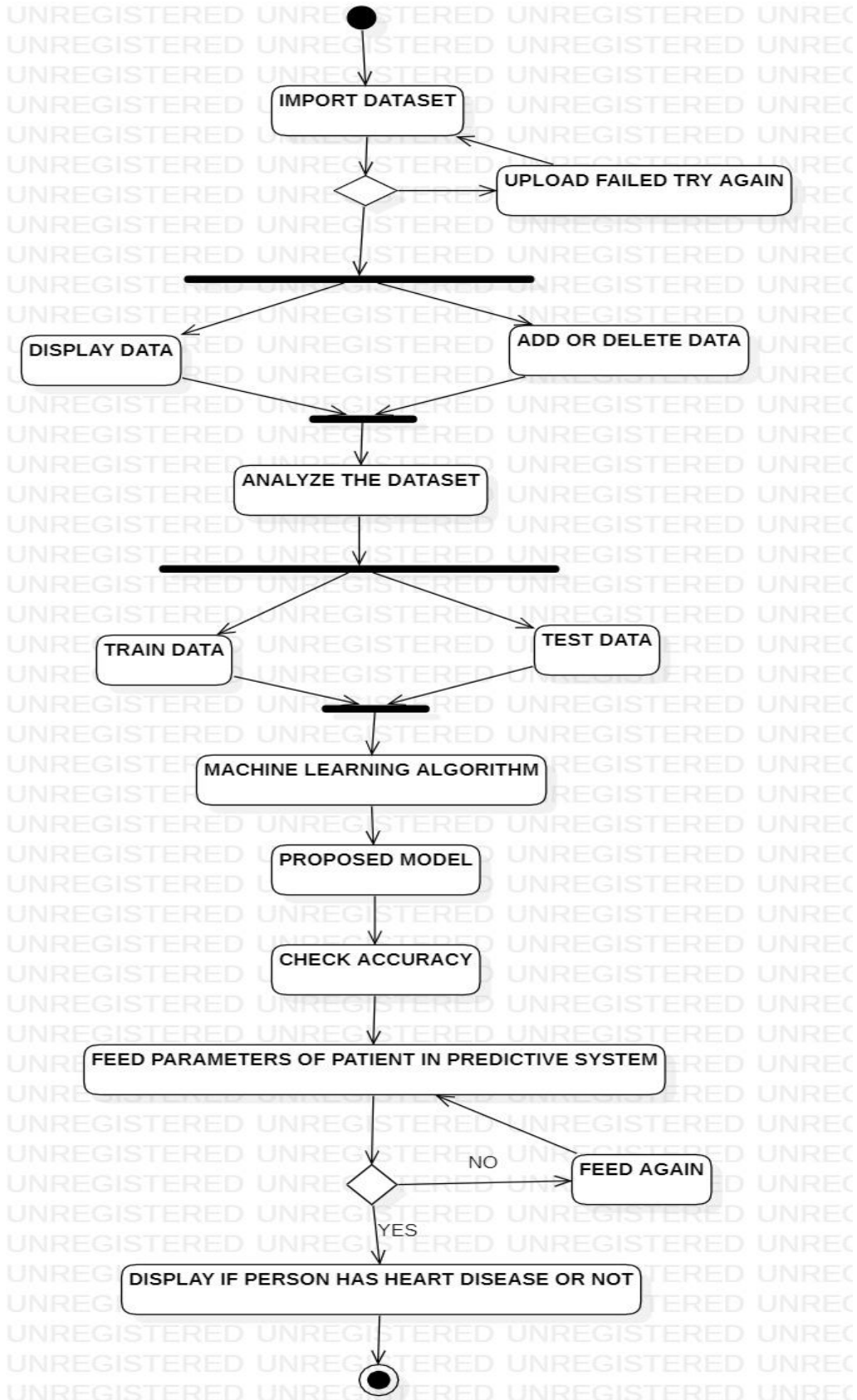
USE CASE DIAGRAM

The Use Case diagram of the project disease prediction using machine learning consist of all the various aspects a normal use case diagram requires. This use case diagram shows how from starting the model flows from one step to another, like he enter into the system then enters all the information's and all other general information along with the symptoms that goes into the system, compares with the prediction model and if true is predicts the appropriate results otherwise it shows the details where the user if gone wrong while entering the information's and it also shows the appropriate precautionary measure for the user to follow. Here the use case diagram of all the entities are linked to each other where the user gets started with the system.



ACTIVITY DIAGRAM

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. Here in this diagram the activity starts from user where the user registers into the system then login using the credentials and then the credentials are matched in the system and if its true, then the user proceeds to the prediction phase where the prediction happens. Then finally after processing the data from datasets the analysis will happen then the correct result will be displayed that is nothing but the Output.


IMPORT DATASET
UPLOAD FAILED TRY AGAIN
DISPLAY DATA
ADD OR DELETE DATA
ANALYZE THE DATASET
TRAIN DATA
TEST DATA
MACHINE LEARNING ALGORITHM
PROPOSED MODEL
CHECK ACCURACY
FEED PARAMETERS OF PATIENT IN PREDICTIVE SYSTEM
NO
FEED AGAIN
YES
DISPLAY IF PERSON HAS HEART DISEASE OR NOT