Institute of Technology University of Moratuwa

Diagnosis Of Heart Patients Using Deep Neural Network

Project II Final Report

DMN Sangeeth	19/IT/IT/527
DSWijethunga	19/IT/IT/543
KAM Chathuranga	19/IT/IT/467
UGABK Bandara	19/IT/IT/463
TN Ranaweera	19/IT/IT/521

Division of Information Technology

Institute of Technology

University of Moratwa

March 2021

Declaration

We declare that this thesis is our own work and has not been submitted in any form for another degree or diploma at any university or other institution of tertiary education. Information derived from the published or unpublished work of others has been acknowledged in the text and a list of references is given.

Name of Student (s) Signature of Student (s)

DMN Sangeeth
DS Wijethunga
KAM Chathuranga
UGABK Bandara
T.N.Ranaweera

Date:

Supervised by

Mrs. K. Galapaththi
Head of the Division
Division of Information Technology
Institute of Technology University of Moratuwa

Signature of Supervisor(s)

Date:

Acknowledgment

We would like to express our special thanks of gratitude to our project supervisor Dr. (Mrs.) Kalpana Galappaththi and our project instructor Mrs. Sapeepa Kuruppu as well as all lecturers and instructors of the IT division of ITUM gave us massive support and encouragement to do this wonderful project. Secondly, we should be thankful to Dr. Prabath and the staff in Base Hospital, Embilipitiya for supporting us. Finally, we would also like to thank our parents and friends who helped us by providing the necessary advice and other requirements.

Abstract

Cardiovascular diseases are the leading cause of death globally, taking an estimated 17.9 million lives each year as WHO said. It is a very unfortunate situation. If heart diseases can be identified earlier, a lot of lives will be saved. One of the most common heart disease detection techniques is based on ECG. When a doctor suspects that a person has heart disease, firstly he or she is referred by the doctor for an ECG test, and after making a decision. That is the current procedure. Here we hope to develop a system that will decide whether or not a person is at risk of heart disease based on the ECG image. The first user should input a digital 12 lead-based ECG image into the proposed system and after the system will display the result. The main technology used for it is Machine Learning. The things we expected by building this system are minimizing the time it takes for a doctor to make a decision, reducing human errors, using it even in the absence of a cardiologist, and, giving more accurate results. In addition to this system, we hope to develop a mobile application for heart patients. Here we expected to provide health tips to the patient based on the data entered by him. These tips will help the patient to maintain diabetes, reduce overweight and obesity, improve physical activity, and keep a good diet plan. There is a chance for a patient to contact a doctor via the mobile app.

Table of Contents

Contents

List of Abbreviations

ECG- Electrocardiogram

UML-Unified Modeling Language

ID- Identity document

API- Application programming interface

BMI- Body mass index

KDD- Knowledge discovery in databases

KNN- K-Nearest Neighbors

SDD- Single Shoot Detection

IDE- integrated development environment

TCP- Transmission Control Protocol

AI - Artificial intelligence

HTML- Hypertext Markup Language

CSS- Cascading Style Sheets

SQL- Structured Query Language

IT – Information Technology

ER- Entity Relation

DNN- Dynamic Neural Network

SVM- Support Vector Machine

NB- Naive Bayes

ANN- Artificial Neural Network

DT- Decision Tree Classifier

PHD- Doctor of Philosophy

LE- Low Energy

Chapter 1

Introduction

1.1 Introduction

The main focus of this project is to develop a system to determine if a person is at risk for heart disease or not. In addition, hope to develop a mobile application to help the patient. The system will display whether the person has the ability can have heart disease or not with the relevant percentage. This can be gotten by the doctor and the patient can get it by mobile application. And also, both doctor and patient can see the digital ECG image in the system.

A lot of benefits are given by this system to heart patients. It expresses a person's high risk of developing heart disease as a percentage.

- When a person has heart pain, they may be taken to the nearest hospital, but there may not be a cardiologist at that time in the hospital. In such a case, our software is very useful so that the heart disease can be quickly identified and referred for treatment.
- A large number of heart patients are reported daily. Cardiologists are not able to monitor every patient in such a situation, so this is very useful.

This paragraph, the description of our project is described in three parts. 1.2 describes the background of our project and the motivation we received to carry out the project. Under 1.3 we describe the aims of the project and the relevant objectives. The Proposed solution under 1.4 describes how we can achieve the aims & objectives in the project. Users, inputs, outputs, and processes are described here.

1.2 Background and Motivation

1.2.1 Background

Heart diseases are the leading cause of death globally, taking an estimated 17.9 million lives each year according to the World Health Organization. The most important behavioral risk factors of heart disease are an unhealthy diet, physical inactivity, tobacco use, and the harmful use of alcohol. It is very important to be able to diagnose heart disease at an early stage. The most common heart disease detection technique is based on an electrocardiogram (ECG), angiography screening, and blood test. In this study, we mainly focus on ECG.

Cessation of tobacco use, reduction of salt in the diet, eating more fruit and vegetables, regular physical activity, and avoiding harmful use of alcohol have been shown to reduce the risk of heart disease. It is important to be able to educate the patient on the steps to be taken for this. Here it is hoped to give medical advice and health tips that are specific to each patient.

If there are a large number of patients in the hospital, the patients have to wait a long time to meet a doctor. But in Sri Lanka, the patient has to travel long distances sometimes. it costs the patient a lot of time and money. It is also important to reduce the distance between the doctor and the patient. So, hope to introduce a mobile application for both doctors and patient

1.2.2 Motivation

A large number of patients will die due to heart disease per year. Another factor contributing to this is the inability to accurately diagnose heart disease in the beginning. With the development of technology, Machine learning can be used to solve the above problems effectively.

The situation of heart patients is recognized by doing ECG testing. It was the first step done by the doctor. In some cases, a general doctor may need to have the help of a specialist to examine the ECG report and draw conclusions. But the smallest point of change in the report could be missed by the specialist doctor also.

Another important thing is that if the strip of ECG had been got wet or damaged It is unable to read the data. Then the doctor's decision may be wrong, and it can be harmful to the patient.

There is still no clear solution to this in the world. But the proposed system can solve that problem. The proposed system can determine if a person has an ability can have heart disease or not by using ECG data. Then no need for Special knowledge of reading ECG

1.3 Aim and Objectives

1.3.1 Aim

Designing a software application that assists doctors in the diagnosis of heart patients accurately and with a minimum amount of time by reducing the time of reading an ECG report and by using a machine learning model and creating a mobile application to contact the doctor and patient for minimizing the gap between them.

1.3.2 Objectives

- Requirement gathering by conducting interviews with doctors and ECG machine operators.
- To conduct a literature review and identify existing systems and their features.
- To define the system features and system boundaries.
- To design the system process with the help of UML diagrams.
- To conduct a feasibility study about the technologies such as programming languages and databases.
- To find the proper data set.
- To preprocess data.
- To train the model using preprocessed data with the most appropriate machine learning model to improve the accuracy.
- To validate and improve the prediction.
- To test the model using test data.
- To develop a mobile application.
- To integrate the system.

1.4 Proposed solution

The main solution is developing a system that can determine if a person is at risk for heart disease or not. After the doctor input a digital ECG image of the person into the system, the system determines whether the person is at risk for heart disease or not. This decision will be given very accurately and quickly.

In addition, hope to develop a mobile application to help the patient. Here, the patient can get medical advice and health tips through the mobile app. It will be very helpful for the patient to maintain a good health pattern and reduce the risk of heart disease. In addition, the patient has the opportunity to contact the doctor when needed.

1.4.1 Users

Doctor

In the beginning, the doctor should be logged into the system, and he has to input a 12-lead-based digital ECG image and patient ID into the system. And also get the result supplied by the system. He has a responsibility to help the patient via a mobile application.

Patient

In the beginning, the patient should be logged into the mobile application, and after he can get the prediction. It means whether he can have heart disease or not. The patient can get help from the doctor via a mobile app. For that, he can use two options available in the system. There are call and chat options. In addition, he has an opportunity to create a new profile or update an existing profile. The important thing is, he can get feedback supplied by the system.

Admin

The system allows the administrator to monitor and remove inappropriate datasets and code. He has to create logins for doctors and patients, and they used them to login into the system. He can update and edit existing information in the system, so he can access the database. And also, can back up data of the system. Admin will be able to log in to both system and mobile applications.

1.4.2 Inputs

Username and password

The doctor and admin have to input their username and password to login into the system. The doctor and patient have to input their username and password to login into the mobile application.

Digital ECG Image

The doctor has to input 12 lead-based digital ECGs into the system.

Patient ID

The doctor has to input the patient's ID into the system.

Patient Profile Form

The patient has to input his information into the system via the mobile application.

1.4.3 Outputs

Result

The system will display whether the person has the ability can have heart disease or not with the relevant percentage. This can be gotten by the doctor and the patient can get it by mobile application. And also, both doctor and patient can see the digital ECG image in the system.

Medical advice and health tips

The mobile application will display medical advice and health tips to the patient. It includes how the patient controls their diet, the Foods that need to get, and how much exercise per day. Etc.

1.4.4 Process

ECG image uploaded into the system by the doctor is stored in the image details database. Then these details are used to process the image and extract features. Then extracted feature is stored in the extracted feature database. The data model is classified by using extracted features of the binary image. The patient login system for authentication. Patient and Admin authentication data is stored in the login info database. Admin can add patient

data stored in the patient's registration info database. Then patients add patient data stored into the database through a mobile application. Classify image phase is a classified disease then a doctor can see the disease report on the monitor. And also, patients get this report through the mobile application. If patients are wanted to contact doctors, patients can be sent a message via API and that message will be stored in the chat database. Next, that message will be displayed on the doctor's API interface. Then Doctor can be replied to the patients according to the same process via API.

the patient should update their profiles. Patients can input their Weight, Height, Cholesterol level, etc. Then the system will be compared the patient's data with standard parameters that a normal person should have. So, patients can be followed those tips. Then they can be protected from heart disease. Then the system will be out some medical advice and health tips. Like Diet Plans, how much exercise do you need to do per day, BMI with doctor's comments?

E.g.-:
If patients input Blood Glucose levels, the system will be compared them with the standard normal Blood Glucose range that some particular patients should have. If the patient's Blood Glucose range is high or low than the standard level, the system will be identified that as an issue. The next system will be out some health tips to manage normal Blood Glucose levels.

1.5 Summary

The proposed system can determine whether the person is at risk for heart disease or not. This decision will be given very accurately and quickly. It is a great way to diagnose heart disease without wasting time. Many lives can be saved by being able to detect heart disease early. Sometimes the doctor's decision may be wrong, and it can be detrimental to the patient. the proposed system will be able to avoid those mistakes and help doctors to make the right decision.

The mobile application minimizes the gap between the patient and the doctor. This allows the patient to contact the doctor and get help when needed. In addition, the app gives some health tips and medical advice to patients. These interfaces are also designed to be easy for the patient to use.

-

Chapter 2

Literature Review

2.1 Introduction

This chapter contains Literature review studies for this project. It indicates that similar approaches are available, as well as how those systems are developed and how existing systems are similar and different to our proposed system, among other systems. This section can be divided into two parts. The First includes related research article papers and applications. The first part of this section will be discussed similar projects to the proposed system using research papers. The summary section aims to demonstrate how relevant projects are beneficial, as well as what are the similarities and differences with this approach.

2.2 Available approaches

As our first step, mainly we forced on how to detect heart diseases by using Machine Learning. Here we could find study some article papers and research papers related to our proposed system and they are listed below.

1. Cardiac Disorder Classification by Electrocardiogram Sensing Using Deep Neural Network [1]
2. Prediction of Heart Disease Using Machine Learning Algorithms [2]
3. A Hybrid Intelligent System Framework for the Prediction of Heart Disease Using Machine Learning Algorithms [3]
4. Effective Heart Disease Prediction using Hybrid Machine Learning Techniques [4]
5. Heart Disease Prediction using Machine Learning [5]

In this phase, we hope to discuss the what are things each article and research papers focus on. The first research article is Cardiac Disorder Classification by Electrocardiogram Sensing Using Deep Neural Network. It mainly focuses to provide a novel automatic detection tool relatively similar and adaptable for cardiac hospitals to process the 12-lead-based ECG images. This paper discusses how to detect Automatically cardiac disorders via a deep neural network using 12-lead-based ECG image processing. The second

research article is Prediction of Heart Disease Using Machine Learning Algorithms. This research provides a detailed description of Naïve Bayes and decision tree classifier. They have been applied in this research, particularly in the prediction of heart disease. And also, some experiment has been conducted to compare the execution of predictive data mining technique on the same dataset. The third research article is A Hybrid Intelligent System Framework for the Prediction of Heart Disease Using Machine Learning Algorithms. This research proposed a hybrid intelligent machine-learning-based predictive system for the diagnosis of heart disease. Here the performances of different machine learning predictive models for heart disease diagnosis on full and selected features have been tested. The fourth research article is Effective Heart Disease Prediction using Hybrid Machine Learning Techniques. They have proposed a novel method that aims at finding significant features by applying machine learning techniques resulting in improving the accuracy in the prediction of cardiovascular disease. The fifth research article is Heart Disease Prediction using Machine Learning. They hope to predict the chances of heart disease and classifies patients' risk level by implementing different data mining techniques. This paper presents a comparative study by analyzing the performance of different machine learning algorithms.

The first research article uses Single shot detector (SSD) MobileNet v2 to design the model. As this article describe, the Deep neural network is best suited for medical images problem; that is the main reason to choose SSD MobileNet V2, a deep neural network-based architecture to detect cardiac disorder on 12-lead-based ECG images. The second research paper analyzes heart disease predictions using classification algorithms. The techniques mainly the Nave Bayes classifier and Decision Tree classifier are used here. Naive Bayes classifiers are a family of simple probabilistic classifiers based on Bayes' theorem with strong independence assumptions between features in data mining and also, they are intensely scalable, requiring some parameters linear in the number of variables (features/predictors) in a learning problem. A decision tree is a decision-making aid that employs a tree-like graph or model of decisions and their potential outcomes, which includes chance event outcomes, resource costs, and utility. It's one technique to show how an algorithm works. The third research paper has applied relatively a large number of methodologies. They have used three well-known Feature Selection algorithms for feature selection. They have considered it as a necessary part of the machine learning process because sometimes irrelevant features affect the classification performance of the machine learning classifier. The used feature selection algorithms are the relief feature selection

algorithm, minimal- redundancy-maximal-relevance feature selection algorithm, and, the least absolute shrinkage and selection operator. In this research, some popular classification algorithms and their theoretical background also have been briefly discussed. The important thing is, they have applied six machine learning classifiers. They are Logistic Regression, Support Vector Machine (SVM), Naive Bayes (NB), Artificial Neural Network (ANN), Decision Tree Classifier (DT), and KNearest Neighbor (K-NN). The fourth research paper has applied an R studio rattle to perform heart disease classification. It has provided an easy-to-use visual representation of the dataset, wrought environment, and building predictive analytics. The multiclass variable and binary classification were introduced for the attributes of the given dataset in preprocessing step. Several machine learning techniques have been used to do the feature selection. In this study, the HRFLM algorithm has been used to classify. HRFLM includes 4 algorithms. Algorithm 1 has a Decision tree-based partition. Algorithm 2 Applies machine learning to find less error rate. Algorithm 3 does feature extraction using less error Classifier. Finally, Algorithm 4 Applies Classifier to extracted features. The proposed hybrid HRFLM approach is used combining the characteristics of Random Forest (RF) and Linear Method (LM). HRFLM proved to be quite accurate in the prediction of heart disease. In the fifth research paper, different machine learning algorithms are used to classify the model such as Random Forest, Decision Tree, Logistic Regression, and Naive Bayes classification techniques. The metrics used to carry out performance analysis of the algorithm are Accuracy score, Precision (P), Recall (R), and F-measure.

According to the first research article, one of the big challenges faced by them is collecting data.in this study, they used a 12-lead-based ECG Images data set collected from Ch. Pervaiz Elahi Institute of Cardiology Multan, Pakistan. This data set consists of four classes and 929 12-lead-based ECG images. Those classes are Myocardial infarction (MI), Abnormal heartbeat, Previous history of MI, and Normal. The second research article used the data set obtained from the Cleveland Heart Disease database which has collected a significant amount of information about patients and their medical conditions. The data set can be divided into parts. They are training dataset and testing dataset and also a total of 303 records with 76 medical attributes were obtained and all attributes are numeric-valued. The third research article also used the “Cleveland heart disease dataset 2016”. It can be accessed from the online data mining repository of the University of California, Irvine. The fourth research article also used the Cleveland Heart Disease Database. because it is a commonly used database for ML researchers with comprehensive and complete records.

13 attributes feature in the prediction of heart disease, where only one attribute serves as the output of the predicted attribute to the presence of heart disease in a patient in the fifth research article, they used the already processed UCI Cleveland dataset available on the Kaggle website for analysis. This database consists of a total of 76 attributes but all published experiments refer to using a subset of only 14 features. The input dataset is split into 80% of the training dataset and the remaining 20% of the test dataset.

In this phase, hope to discuss the procedures of each system. According to the first article, firstly Data included in the dataset should be preprocessed. The second step is, Training the Model by using Preprocessed data and Configuring the SSD mobile V2 model. Finally, when the user inputs an unknown ECG image into the system, the system will determine which class it belongs to by using the learned model. According to the second research paper, in the system, a dataset is included. It contains patient details. In the phase of the attribute selection process, it selects useful attributes for the prediction of heart disease. After identifying the available data resources, they are further selected, cleaned, and made into the desired form. That process is called preprocessing. Different classification techniques will be applied to preprocessed data to predict the accuracy of heart disease. Finally, the accuracy measure process will compare the accuracy of different classifiers. According to the third research paper, the system has a database. Data obtained from the database should be preprocessed. For those preprocessing techniques such as removing missing values, standard scalar, and MinMax Scalar have been applied to the dataset for effective use in the classifiers. To classify heart patients and healthy people, six machine learning classification algorithms are used here. Finally, the system has to do a validation method. According to the fourth research paper, they used an R studio rattle to perform the classification of heart disease from the Cleveland UCI repository. The first step was loading the UCI dataset and the data became ready for preprocessing. After The subset of 13 attributes is selected from the pre-processed data set of heart disease. The three existing models for heart disease prediction (DT, RM, LM) were used to develop the classification. The evaluation of the model is performed with the confusion matrix. Four outcomes are generated by the confusion matrix, namely TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative). These measures can be used for the calculation of accuracy, sensitivity, and specificity. According to the fifth research, as the first step, the collected dataset should be preprocessed. After it classifies into two parts namely, Training data and Test data. After, classification techniques are used to test the model. Finally, the proposed system will give the result.

The first research article said it achieved high accuracy results (98%) in detecting cardiac abnormalities, and several cardiologists have verified the accuracy result manually. In this second article, the heart disease prediction system has been introduced with a Bayes classifier and a decision tree classifier for the prediction of heart disease. When we analyzed the result, the tree has better accuracy as compared to the naive Bayes classifier. In this third article, the result given is the presence of heart disease or the absence of heart disease. the system used three FS algorithms, seven classifiers, one cross-validation method, and performance evaluation metrics for heart disease diagnosis. The system has been tested on the Cleveland heart disease dataset to classify heart diseases and healthy subjects. So, the result will be more accurate. In this fourth article, the result given by the proposed system is more accurate. The fifth article compares the accuracy score of Decision Tree, Logistic Regression, Random Forest, and Naive Bayes algorithms for predicting heart disease using the UCI machine learning repository dataset. The result of this study indicates that the Random Forest algorithm is the most efficient algorithm with an accuracy score of 90.16% for the prediction of heart disease.

As our second step, mainly we forced on existing mobile applications, there we could find and study some applications.

Heart disease is regarded as one of the world's leading causes of death. Some unhealthy behaviors, like being overweight, eating a poor diet, smoking, and consuming alcohol, are risk factors for this devastating disease. Patients, on the other hand, can live a healthy lifestyle if they are guided by persuasive-emotional technologies. In the past few decades, the research community has focused on Mobile cloud computing, augmented reality, and IoT with iOS technology to provide a platform between human and software applications. This work is widely recognized by several companies and medical fields for identification and communication between the system and heart patient to give medical tips and health feedback.

Advancement in healthcare technologies and biomedical equipment leads to accurate diagnosis of heart-related diseases. The major challenges associated with telehealth care technologies are the complex computational requirement and large amount of data processing in continuous monitoring. A mobile cloud computing approach [6] is presented in this work to overcome the issues involved in ECG telemonitoring. The mobile cloud approach is superior to telehealth monitoring techniques due to the access to centralized

cloud data and report delivery to the mobile application. BeHabbit [7] proposed a system based on an emotional-persuasive habit change of Heart disease Patients. It is supported to change the lifestyle of heart patients. The proposed system is designed, implemented, tested, and evaluated by 10 users. In conclusion, the users are satisfied to used BeHabit to change their bad habits. Emotional and persuasive features which are integrated into BeHabit are the key to helping patients to change their bad habits. BeHabit and the integrated feature can be used as a guideline for healthcare developers and providers for the improvement of health services. An augmented reality-supported mobile application [8] based on the deal with the problem of heart sound analysis and disease diagnosis by using a mobile application that can perform the mentioned tasks by having also support from both virtual reality- and augmented reality-oriented components. Thanks to the using features and functions provided by the application, it is possible to analyze signals instantly and have rapid feedback over the interface supported with virtual or mixed reality objects combining both real and virtual worlds in a common ground. iOS mobile application system [9] connects with a microcontroller (Prototyped Ph.D.) over Bluetooth LE and receives raw analog data from the microcontroller. Then it formats and displays the data that comes through the serial port in the proper way like graphs. It also connects with Health services for saving and receiving user health records. Additionally, it provides many more functionalities to fulfill system requirements. iOS mobile application with IoT architecture and implementation of machine leaning model to identify absence or presence of heart diseases. In the context of heart diseases monitoring system, consists prototyped body sensors and smart gateway (smart mobile phone) where mobile application is installed. Smart gateways at fog layer facilitate flow of health data from cloud and body sensors. Also, mobile application installed in smart gateway helps to process and display health data obtained from devices.

2.3 Summary

The proposed system is capable of accurately determining whether a person is at risk of having a heart attack or not. Our proposed system is the best application for doctors because it makes it easier for doctors to make decisions. And our mobile application will reduce the gap between patients and doctors, it uses to give medical advice and health tips. Our mobile application interfaces are very simple and the system will give results very efficiently. The common point between our proposed system and other system Diagnosis of heart patients. The majority of those projects are using machine learning techniques for that. But our system is used deep learning and prosed system uses image processing. 12-Lead Based Digital ECG images including a database are used to implement the proposed system. But others used a database that includes a significant amount of information about patients and their medical conditions. In our system, the patient has a chance to contact a doctor for getting instructions and advice. But other systems do not allow that.

According to the table given below, the comparison between the proposed system and other approaches can be seen.

A. Cardiac Disorder Classification by Electrocardiogram Sensing Using Deep Neural Network
B. Prediction of Heart Disease Using Machine Learning Algorithms
C. A Hybrid Intelligent System Framework for the Prediction of Heart Disease Using Machine Learning Algorithms
D. Effective Heart Disease Prediction using Hybrid Machine Learning Techniques
E. Heart Disease Prediction using Machine Learning
F. A mobile cloud computing approach
G. BeHabbit H. An augmented reality-supported mobile application
I. iOS mobile application system
J. Heart disease detection system using deep learning.

System / Feature	A	B	C	D	E	F	G	H	I	J
Using Deep learning algorithms	√	X	X	X	X	X	X	X	X	√
Accuracy above 95%	√	X	X	X	X	X	X	X	X	√
Patients can contact a doctor	X	X	X	X	X	X	√	√	X	√
Instance reply	√	√	√	√	√	X	√	√	√	√
App gives medical advice and health tips	X	X	X	X	X	√	√	√	√	√
Give prediction about risk	√	√	√	√	√	X	√	√	√	√

Chapter 3

Approach

3.1 Introduction

This approach has been arranged into different parts. This chapter describes the architectures that are going to implement in system design, image processing, data training models, and online database system in detail

3.2 Approach

1. Create a Trained Data model & image processing

Anaconda navigator software and python Jupiter notebook have used this module. This IDE used its way of coding. The code has to be arranged that suits that IDE. Two-process models are implemented by Jupyter with a kernel and a client. The client is the interface offering the user the ability to send code to the kernel. The notebook communicates with the underlying kernel using a WebSocket, a TCP-based protocol implemented in modern web browsers. All libraries like TensorFlow, NumPy, Keras, and MATLAB are going to use in this module.

2. Availability to Sign in and sign out system

The Doctor can be signed into the system. Then doctor authentication data is stored in the login info database. ECG image is uploaded by the doctor and then the image is stored in the image details database. The doctor can reply to patients' messages coming through the mobile application.

Also, Admin can be signed into the system. Admin can be added patient data stored in the patient's registration info database.

3. Patients connected to the system through the Mobile application

Patients can be added to patient data stored in the database through mobile applications.

Also, patients can be got reports through the mobile application. If patients are wanted to contact a doctor, patients can be sent a message via API and that message will be stored in the chat database.

4. Upload ECG images and store them in a database system

For this section, component-based architecture is used. All ECG images are stored in the system database. Through this, Easy and understandable interfaces are given for the doctors and patients.

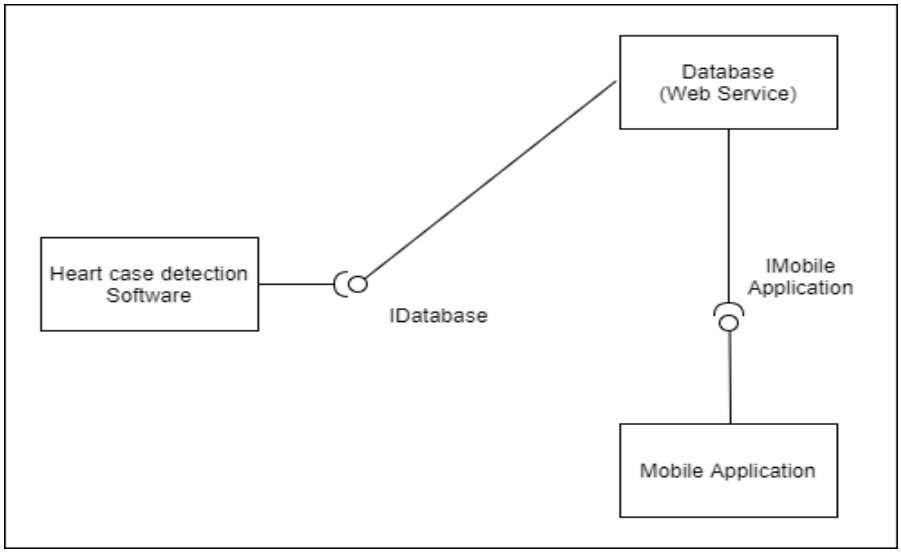
5. Use an online database to store patients' data

Patients can be created a profile using personal details. Also, they can input about below kind of data.

- Current situation of the disease
- Have you ever had a disease like this before? If so, details.
- If you have chronic diseases? You will mention them.
  (High blood pressure, Diabetics, etc.)

6. Module Showing all the output in a clear way that all users can identify

This part is very useful. All outputs are controlled by this module. A component-based design is being used to construct this module. It will be made easier to effectively handle all the outputs. This will also be given a simple approach to code.



3.3 Summary

We addressed our strategy for implementing the solution in this chapter, as well as the many architectures and design patterns that we would apply in our application to improve its output. All the benefits and reasons for selecting this architecture for our system are highlighted.

Chapter 04

Technology

4.1 Introduction

There are two software products to be developed, first one is a heart disease detection system using machine learning. The second one is a mobile application that helps patients. To develop a heart disease detection system, python is chosen as a programming language. Hope to use TensorFlow, Keras, NumPy, and Matplotlib as libraries and frameworks. Jupyter notebook may be used as IDE in the Anaconda environment and GitHub is used as a version control system. Moqups application will be used to create user interfaces as a graphic Tool. To develop mobile applications, hope to use PhoneGap as a mobile app development framework. SQLite will be used to create a database.

4.2 Technology adapted

1. Python

Python is a programming language that may be used to build a variety of applications. It's a popular choice among developers for projects involving artificial intelligence (AI), machine learning, and deep learning. Python language consists of many libraries and frameworks that make coding easy. Another important thing, Python projects can be integrated with other systems coded in different programming languages, Python has a syntax that is easy to understand and provides a lot of code review and test tools. That’s why Python is used to build the heart disease detection system.

2. TensorFlow

TensorFlow is an open-source artificial intelligence library. It uses data flow graphs to build models. It helps developers to create large-scale neural networks with many layers. Classification, perception, understanding, discovery, prediction, and creation are some of the most common uses for TensorFlow. So, this is used to train and run the deep neural network for image recognition.

3. Keras

Keras is an open-source highly modular neural network library coded in Python and a Python-based framework that makes it easy to debug and explore. It provides a Python interface for artificial neural networks. Keras acts as an interface for TensorFlow. So, in this project, Keras is used to define and train neural network models in just a few lines of code.

4. NumPy

NumPy is one of the most powerful Python libraries. it is adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. Moreover, NumPy forms the foundation of the Machine Learning stack. To train the data model, some algorithms should be used. As a result of that, NumPy is will be used in this project.

5. Matplotlib

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. Machine learning, helps to understand a huge amount of data through different visualizations. In this project, Matplotlib is used to create graphs such as Line Plots, Histograms, Three-dimensional plots, Steam plots, Bar charts, Pie charts, Tables, Scatter plots, etc.

6. Anaconda

Anaconda is an open-source distribution of the Python and R programming languages for scientific computing. The most common data science and machine learning tools, such as Jupyter, pandas, NumPy, Matplotlib, and scikit-learn come with Anaconda. So, the proposed heart disease detection system is implemented within the Anaconda environment.

7. Jupyter notebook

The Jupyter Notebook is an open-source web application that allows developers to create and share documents. Jupiter notebook works in most web browsers and allows developers to mix code with descriptive text in a simple and easy-to-understand way. This also comes with an Anaconda navigator. In this project, Jupyter notebook is used as IDE. Here codes can be run line by line and can create graphs, notes, and comment proper manner. that’s why the Jupyter notebook is used.

8. GitHub

GitHub is a provider of Internet hosting for software development and version control using Git. It simplifies the process of working with team members and makes it easy to collaborate on projects. In this project, we hope to work on files and easily merge our changes with the master branch of the project.

9. Moqups

Moqups is a visual collaboration tool that includes whiteboard, diagram, and design features in a single. it is an online app and can be for free. It is used to create a wide range of visual materials for any project. In this project, Moqups will be used to create both mobile applications and heart disease detection system user interfaces.

10. PhoneGap

PhoneGap is an open-source mobile app development framework by Adobe Systems. It is also known as Apache Cordova. Using PhoneGap the developer can create an app easier, does not require to know mobile programming language but only web-development languages like HTML, CSS, and Jscript. In this project, hope to use PhoneGap to develop a mobile application.

11. SQLite

SQLite is a SQL-based embedded database management system. An SQLite database with full API is provided by iOS, Android, and Windows Phones. the SQLite Cordova plugin offers a simple API for creating databases and running SQLite queries. This is used to create a database that is connected to the mobile application.

4.3 Summary

A lot of technology has to be used to develop both heart disease detection systems and mobile applications. In this study, we have to use the most suitable and popular technologies. It makes software development easier. The proposed heart disease detection system depends on machine learning. It is special technology used here. Also, the technologies used to develop the mobile application are special. Hope to create more accurate, faster, and more efficient systems using the technologies discussed above.

Chapter 5

Analysis

5.1 Introduction

The process of identifying the expectations of users for an application that is to be built or modified is known as requirements analysis. This includes all of the tasks that are conducted to determine the needs of various stakeholders. This phase is a significant and essential activity. This chapter hopes to discuss how we gathered and analyzed the requirements and what are the requirements of the suggested solution.

5.2 Requirement gathering and analysis

Requirement gathering and analysis was important phase in this project. Because this project is related to the medical field. This phase was a big challenge for us. As the first step, we were able to identify who are the Stakeholders in this project. (a doctor, a patient, an admin, and, a development team) The second step was capturing stakeholder requirements. Some mechanisms had to be used to gather requirements. The first one is interviewing. We had to chance to visit the hospital and arrange a meeting with the Cardiologist. It was a good opportunity to ask questions and know-how does a doctor determine whether a person has heart disease or not. After discussing with the ECG machine operator, we had to see and know how it works and its features. Here we could gather requirements and get a good idea of what we hope to develop father. The second method used is asking questions. Several heart patients were selected for this. There some questions were asked of them and we were able to understand the problems they are facing and what they were expecting. For that, the designed questionnaires were used, and they included various questions related to the project. We studied the existing system and drew some diagrams to understand the process. It was very helpful to use to fulfill the requirements. In addition to the Internet was used to find some facts. In this phase use-case diagram also was created to understand how the system or service would work.

After gathering requirements, the third step was categorizing the gathered requirements. To make analysis easier, considered grouping the requirements into these two categories:

- Functional Requirements
- Non-Functional Requirements

The fourth step was Interpret and Record Requirements. In this phase, determined and listed in sufficient detail which requirements are achievable or not, and how the system or product can deliver them. Although many requirements are important, some are more important than others. So, we finally prioritized the selected requirements.

5.3 Requirements of the suggested solution

Functional Requirements

- The doctor and admin shall be able to login into both the heart disease detection system and mobile application.
- The doctor shall be able to input digital ECG images into the heart disease detection system.
- The doctor shall be able to input the patient's ID (if it is possible) into the heart disease detection system.
- The heart disease detection system shall be able to determine whether or not a person is at risk of having a heart disease by itself.
- The heart disease detection system shall be able to display results.
- The admin shall be able to create logins for doctors to use both the heart disease detection system and mobile applications.
- The admin shall be able to create logins for patients to use only a mobile application.
- The admin shall be able to update both the heart disease detection system and mobile application.
- The patient shall be able to login into the mobile application.
- The patient shall be able to create a new profile or update an existing profile.
- The patient shall be able to contact the doctor by using chat and call options via mobile application.
- The doctor shall be able to respond to calls and chat with a patient via mobile application.
- The mobile application shall be able to store patients' data using a database.

- The mobile application shall be able to display medical advice and health tips to patients.

Non-Functional Requirements

- Both the mobile application and heart disease detection system shall take a short time without any delay to load user interface screens.
- The system shall verify the login information within five seconds.
- The heart disease detection system shall give a result with the highest accuracy.
- The system shall be able to access their resume without failure.
- The doctors can use the heart disease detection system at any time during the day and the patients can use the mobile application at any time during the day. In the case of unplanned system downtime, all features will be available again after one working day.
- The system shall be able to store patients' data forms within the database up to a large amount.
- Only the admin can access data in the database. Because it includes more sensitive patient data.
- The admin shall be able to maintain backups to ensure the system's database security.
- The system's interface has to be user-friendly and easy to use. because patients don't have IT-related knowledge.

5.4 Summary

The gathering requirements phase was not easy, because we had to visit the hospital and discuss even with the specialist doctor. There we have been informed about the way of making discissions by doctors. Other important facts are the functionality of the ECG machine and getting ECG images to form it. These factors mainly affect the proposed system. Another one is the problems faced by patients. Some of the distance between the patient and the doctor, patients' unwillingness to seek treatment, etc.

Chapter 06

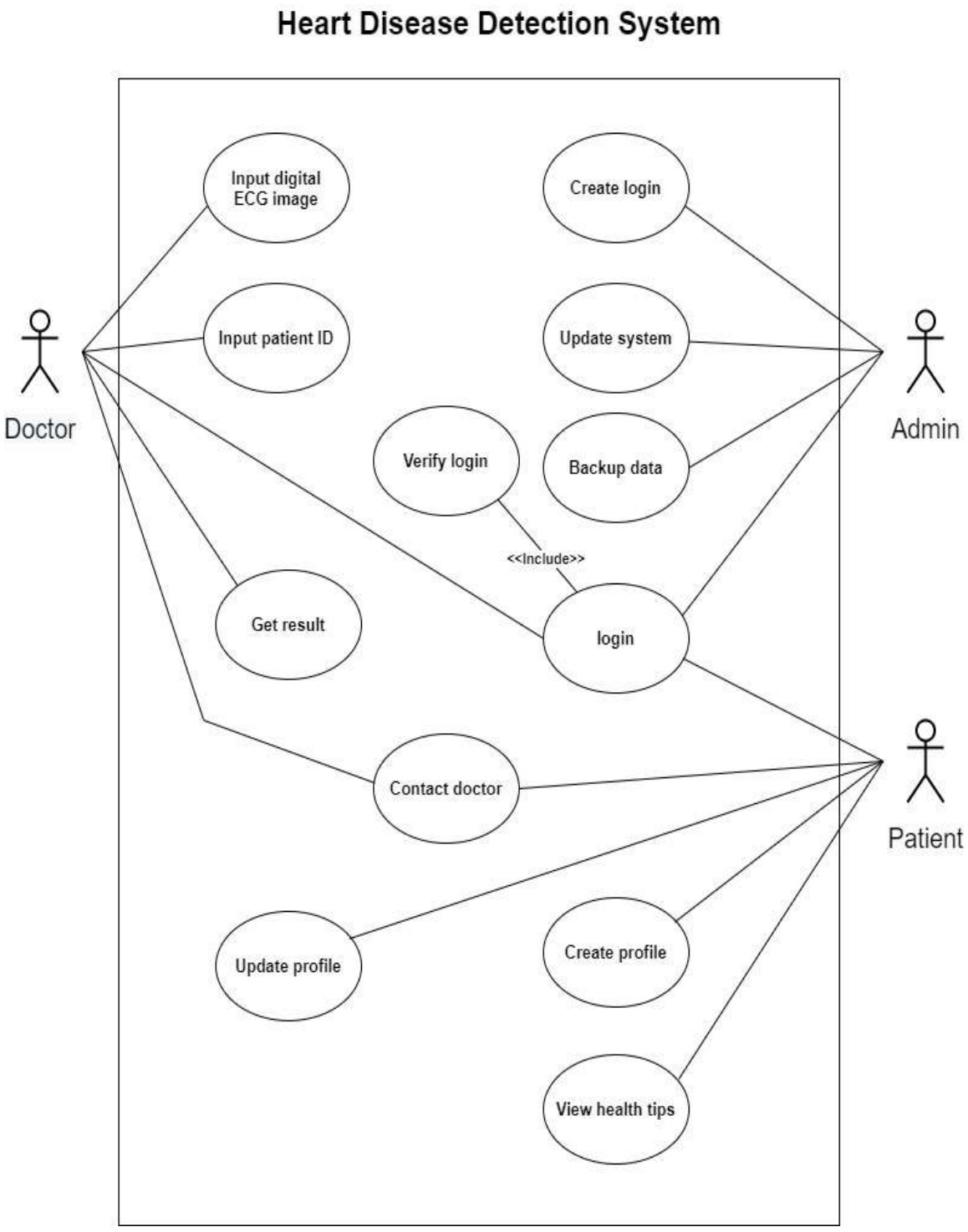
Design

6.1 Introduction

The purpose of the Design phase is to convert the requirements into complete and detailed system design specifications. The design of a system is likely the most important aspect impacting software quality. It has a significant impact on the project later on, especially during testing and maintenance. That’s why this phase is important. In this chapter, we hope to discuss several diagrams. If they are use-case diagrams, entity-relationship diagrams, class diagrams, activity diagrams, sequence diagrams, and state chart diagrams. Several diagrams were created for this phase and included a brief commentary with each diagram. These diagrams are really helpful in the implementation part of the proposed system that’s why we gave a lot of focus on this part. That's why we put so much emphasis on this section.

6.2 Design

6.2.1 Use-case Diagram

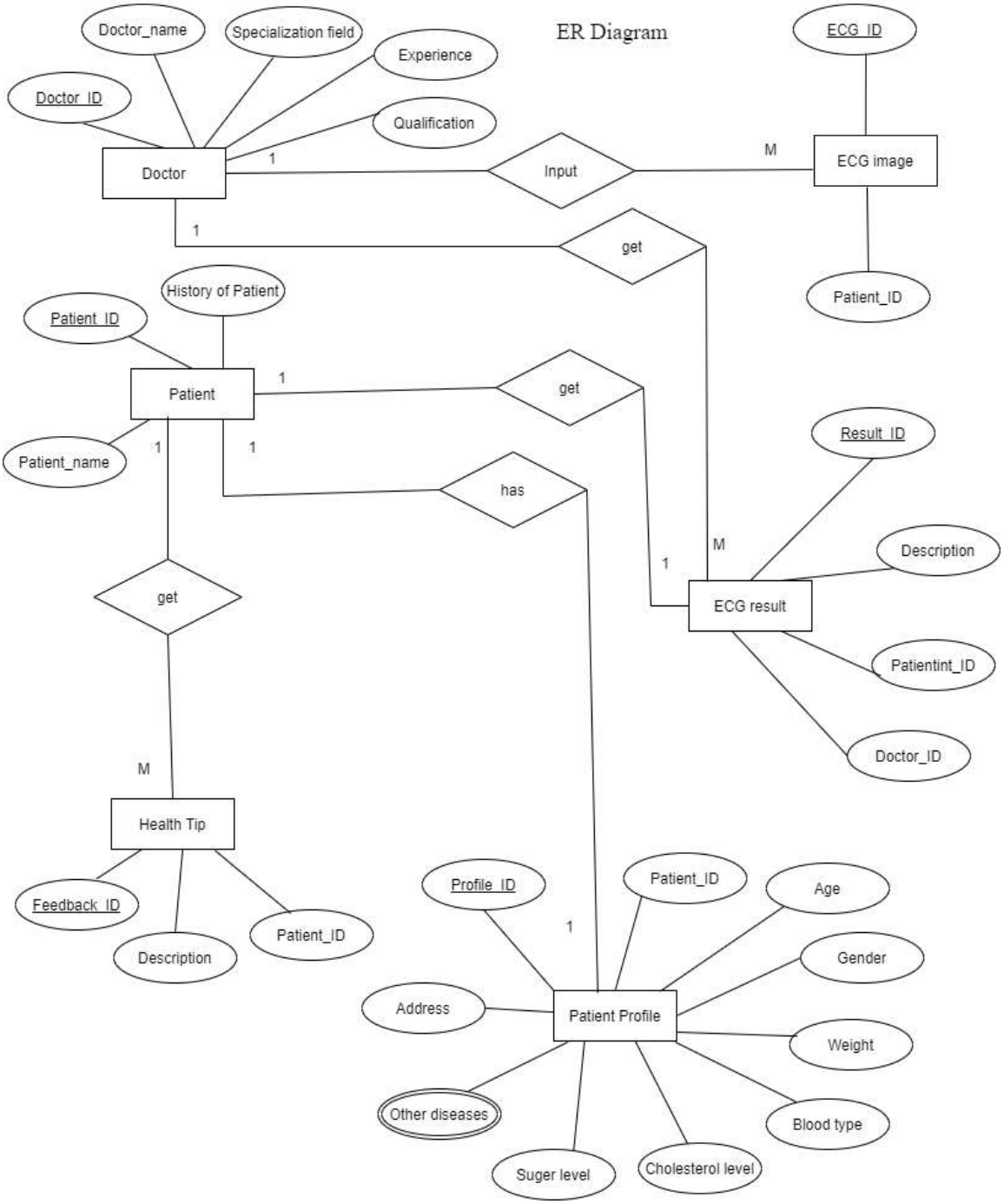


The use-case diagram explains the high-level functions and scope of the proposed system. This also identifies the interactions between the system and its actors. The use cases and actors in diagrams describe what the system is going to do and how the actors use it. The users that interact with the system are considered actors. There are three actors in this use-case diagram.

- Doctor
- Patient
- Admin (Administrator)

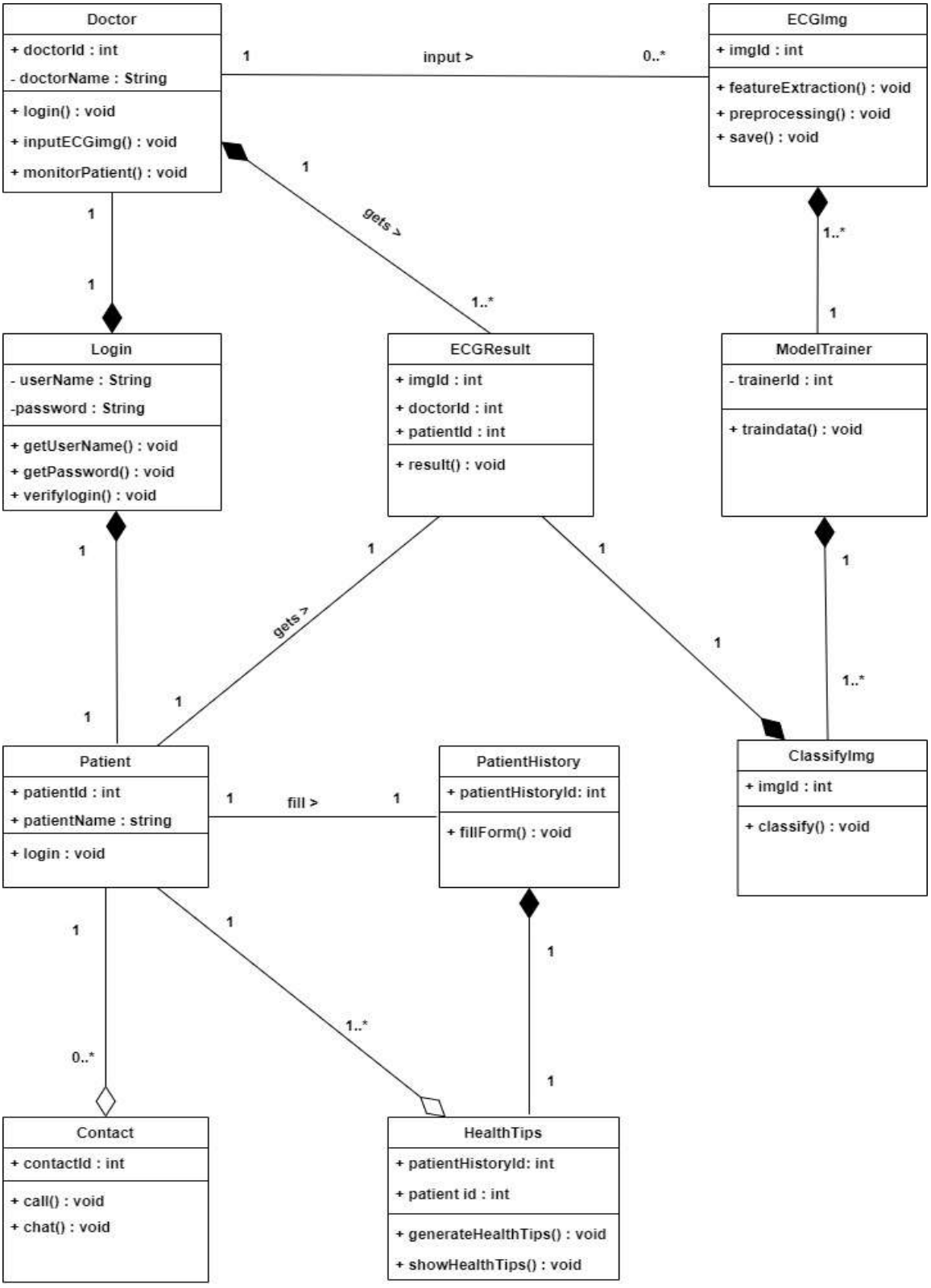
The actors should do the use cases. In the begging, the doctor has to login into the system and upload the ECG image with the patient ID into the heart disease detection system. After he can get the result. Another factor is the patient. In the begging the patient also has to login into the mobile application and after, can create a new profile or update the existing profile. By using the app patients can get medical advice and health tips. If he wants to contact the doctor, he can contact him by using chat or call options. The admin also has to do several use cases. There are logging into the system, creating logins for both doctors and patients, updating the system, and getting backup data.

6.2.2 Entity-Relationship Diagram



ER Diagrams that display the relationship of entity sets stored in a database. Also, ER Diagram helps to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes, and relationships. Our system has six entities. These are Doctor, ECG image, Patient, ECG result, Feedback, and Patient Profile. A doctor has five attributes. They are Doctor_ID and Doctor_name, Specialization field. Experience and Qualification. A doctor is made a one-to-many relationship with ECG image. ECG image has two attributes. They are ECG _ID and Patient_ID. ECG_ID is the primary key. The primary key is a key in a relational database that is unique for each record. It's a unique identifier. The doctor made one-to-many relationships with ECG results. It has four attributes. The patient has a one-to-one relationship with the ECG result. Also, the patient has a one-to-many relationship with Feedback and Patient Profile. In the patient profile, the patient can input their details and update the profile. They can input Address, Age, Gender, Weight, Blood type, Cholesterol level, Sugar level, and other diseases. Then the patient can get health tips and medical advice via the system.

6.2.3 Class Diagram

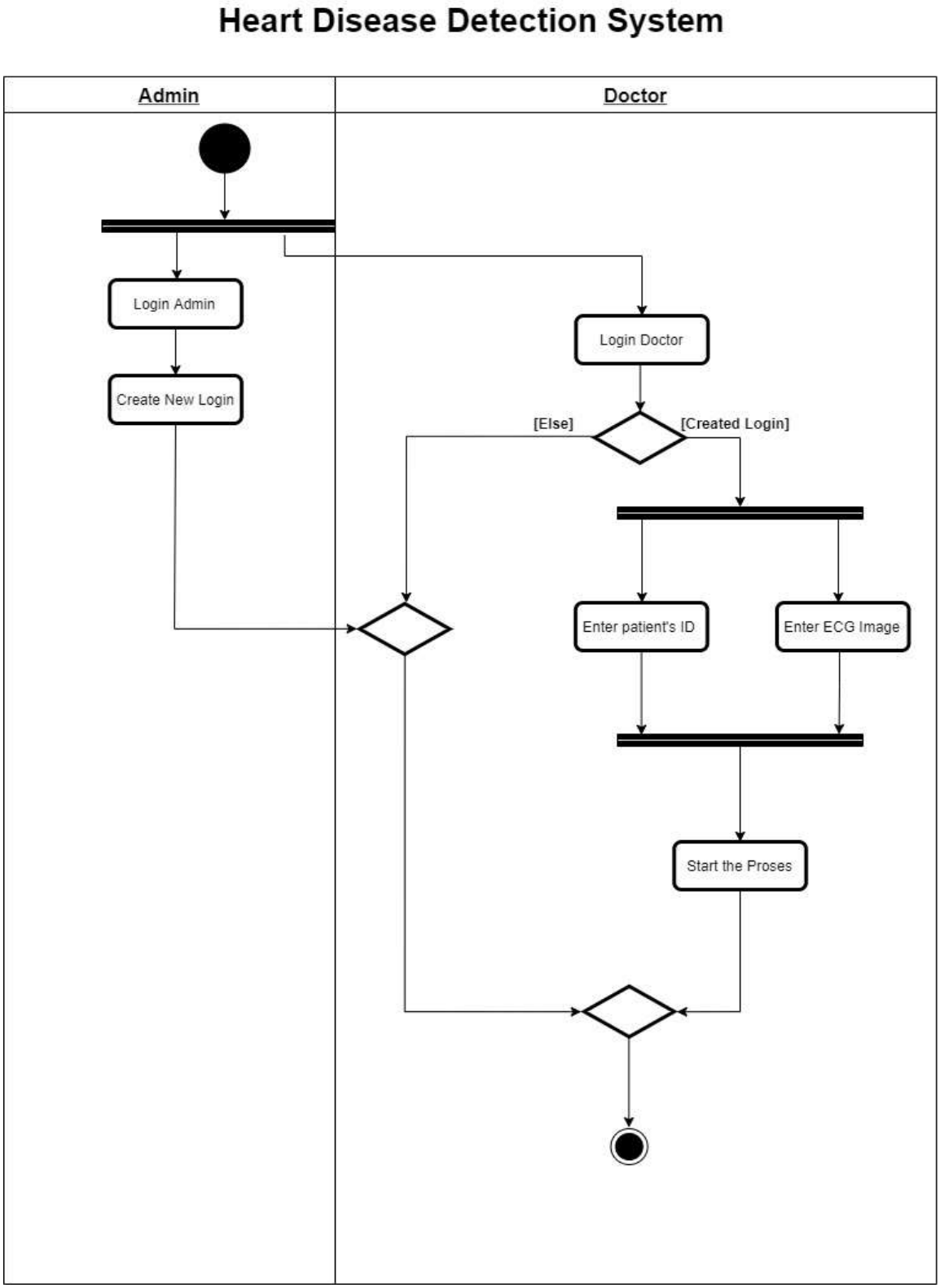


rectangles inside each box. The top rectangle has the name of the class; the middle rectangle contains the attributes of the class; and the bottom rectangle contains the methods of the class, also known as operations. Consider the doctor class as an example. The doctor is the class name. doctorId and doctorName are the attributes of the class and also login(), inputECGImg() and, monitorPatient() are the methods.

This diagram shows 10 classes and the relationships between them. There is a composition relationship between the doctor class and ECGresult class. According to the diagram, the ECG result class can't be available without a doctor class. Other than that, the composition relationship can be found between the doctor class and login class, ECGresult class and ClassifyImg class, login class and Patient class, ClassifyImg class, and ModelTrainer class, etc.

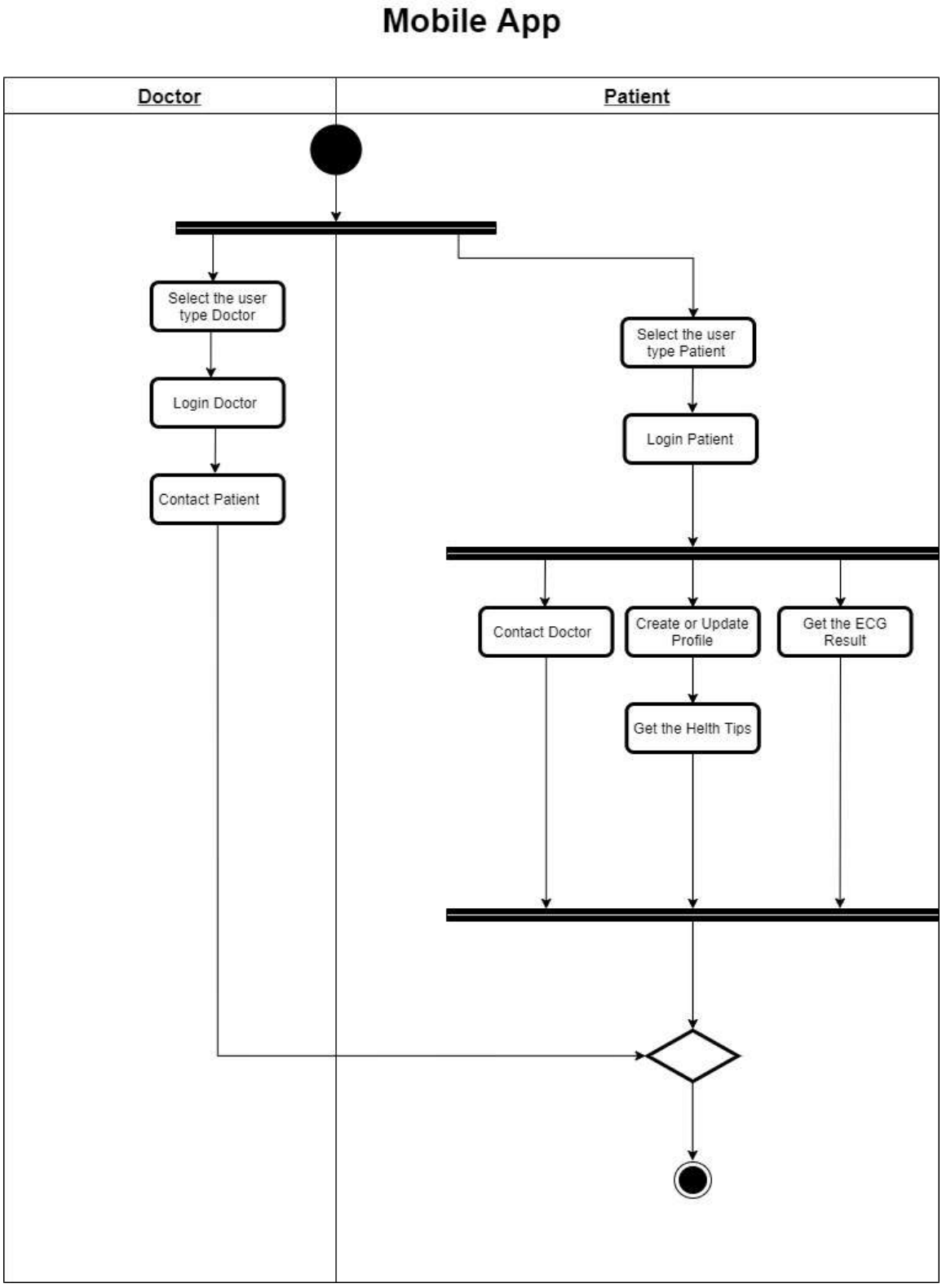
There are aggregation relationships between the Patient class and Contact class and between the Patient class and HealthTips class. According to the diagram, the Patient class does not depend on the Contact class. That is the behavior of the aggregation relationship. Other than that, Association relationships are here, between the Doctor class and ECGimg class, Patient class and PatientHistory class, and ECGresult class and Patient class.

6.2.4 Activity Diagram



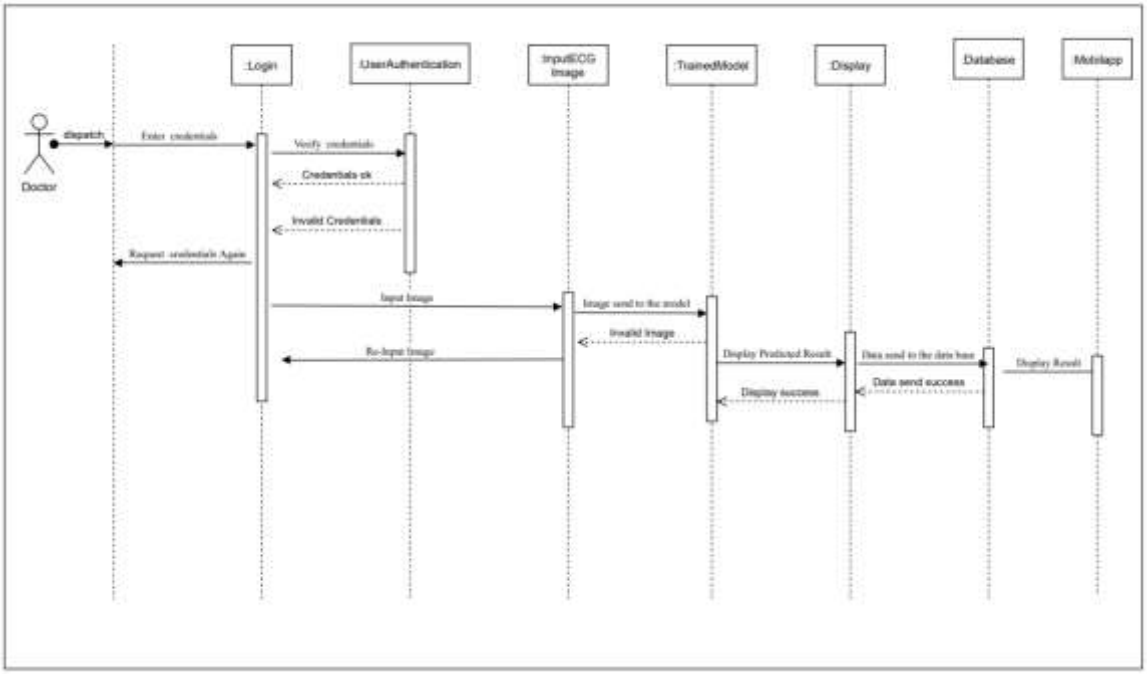
Activity diagrams show how activities are organized to create a service. Typically, some operations are required to complete an event, especially when the operation is intended to accomplish a few different things that necessitate coordination, or how the events in a single use case relate to one another, particularly in use cases where activities overlap, and coordination is required.

After starting the diagram, Doctor and the Admin can Login. If Admin Login, he can create a new Logger. If the Doctor Login in to it, he can use his account when the account has been created by the admin only. After entering the ID and ECG image of the Patient the Doctor can Start the process.

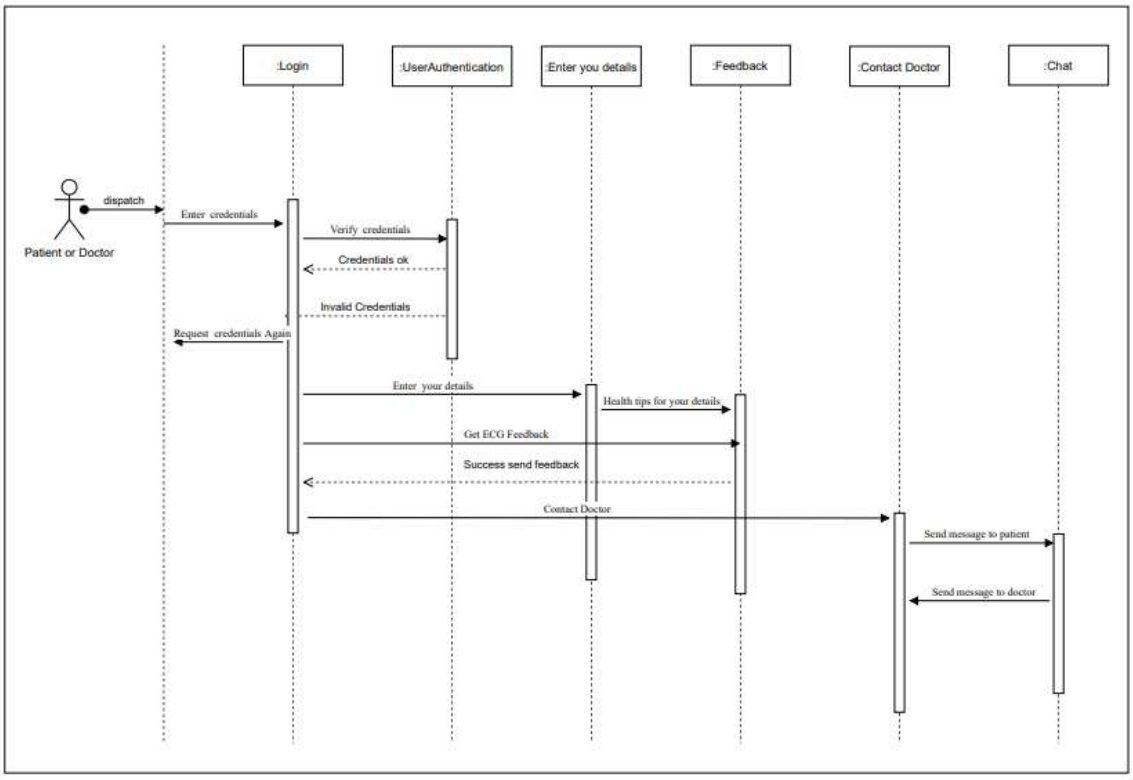


After starting the diagram, the Patient or Doctor can be login. If the Doctor login in, he has to select the user type “Doctor” and can be login there. And he can contact the Patient. If the Patient login he has to select the user type “Patient” and he can be Login. After doing it he can Contact the Doctor, Receive the Result of the ECG, Create and Update his Profile and Receive health tips.

6.2.5 Sequence Diagram

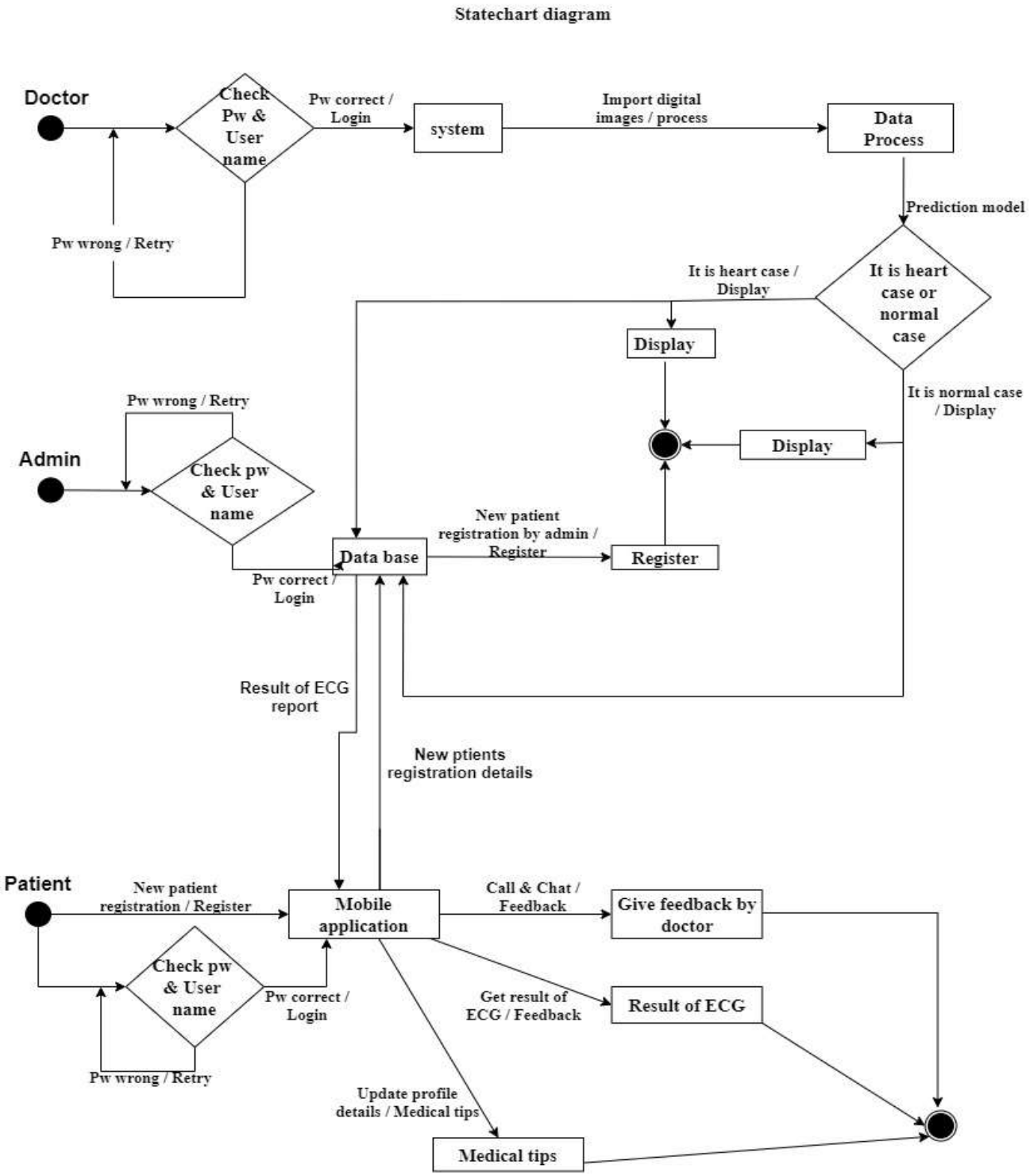


According to this sequence diagram, the first step is the doctor login into the system. Here username and password should be verified by the system. If the username or password is invalid, the user has to request the credentials again. The long rectangular section of this diagram shows the time required to perform this task. In an emergency, the doctor can use the system without logging. The second step is to input a digital ECG image into the system. Here if the user inputs an invalid ECG image, the user has to input a valid one. After inputting the ECG image, the prediction is made by the trained model. After the predicted result will be displayed by the system. In the next step, the result is sent to the database and if it was done successfully, the system displays that by message. the mobile app also requests that data in the database.



This diagram is designed for the mobile application. As before the doctor and patient should be login into the system. In this phase, the system should verify the username and password. After the patient can input health details into the form given by the system. The system will supply health tips decided based on the patient's health information to the patient. The patient can get that feedback after logging into the system. If it was done by the patient successfully, the system displays that by message. In this phase, the patient will be able to contact the doctor also. For that, there are two options. If they are chat and call options.

6.2.6 State Chart Diagram



The statechart diagram is one of the five UML diagrams used to model the dynamic nature of a system. They define different states of an object during its lifetime and these states are changed by events. Statechart diagrams are useful to model reactive systems.

This diagram clearly shows the state of " Diagnosis of heart patients using machine learning". There are three actors.

- Doctor
- Admin
- Patient

The doctor can enter the system using a username and password. then the doctor can import a patient's digital ECG image into the system. Then, the system checks it to see if the patient has heart disease. The result is display on the system interface.

The admin can enter the database using a username and password. then the admin can register new patients.

The patients can enter the mobile application using a username and password. if the patient hasn't registered, the patient can register as a new patient. then, patients have an opportunity to get advice or feedback by contacting the doctor, there have two opportunities chat & call. the patient can get ECG report result through the mobile application. patients can update their profiles and get medical tips.

6.3 Summary

The design phase is a very important step in a software development project. Because the implementation phase depends on this phase. Here we discussed several UML diagrams. The use case diagram describes a graphical depiction of a user's possible interactions with a system. The ER diagram describes relationships among uses, objects, places, concepts, or events. The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The activity diagram describes the dynamic aspects of the system. The sequence diagram describes how—and in what order—a group of objects works together. Finally, the state chart diagram describes the flow of control from one state to another state.

Chapter 7

Implementation and Evaluation

7.1 Introduction

There is a software product we developed, the Diagnosis Heart Patient using Deep Neural Network. To develop a heart patient detection system, python is chosen as a programming language. We used JavaScript to design front-end web pages. We used TensorFlow, Keras, React, and FastAPI as Deployment tools and frameworks. Jupyter notebook is used as IDE in the Anaconda environment and GitHub is used as the version control system. Also, Spyder, VS Code, and Postmen are used as the Development tool. Moqups application is used to create user interfaces as a graphic Tool. Firebase is used as the database.

7.2 Technology Adapted

1. Python

   Python is a programming language that may be used to build a variety of applications. It's a popular choice among developers for projects involving artificial intelligence (AI), machine learning, and deep learning. Python language consists of many libraries and frameworks that make coding easy. Another important thing, Python projects can be integrated with other systems coded in different programming languages, Python has a syntax that is easy to understand and provides a lot of code review and test tools. That’s why Python is used to build the diagnosis heart patient detection system.

2. TensorFlow

   TensorFlow is an open-source artificial intelligence library. It uses data flow graphs to build models. It helps developers to create large-scale neural networks with many layers. Classification, perception, understanding, discovery, prediction, and creation are some of the most common uses for TensorFlow. So, this is used to train and run the deep neural network for image recognition.

3. Keras

   Keras is an open-source highly modular neural network library coded in Python and a Python-based framework that makes it easy to debug and explore. It provides a Python interface for artificial neural networks. Keras acts as an interface for TensorFlow. So, in this project, Keras is used to define and train neural network models in just a few lines of code.

4. Anaconda Navigator

   Anaconda is an open-source distribution of the Python and R programming languages for scientific computing. The most common data science and machine learning tools, such as Jupyter, pandas, NumPy, Matplotlib, and sci-kit-learn come with Anaconda. So, the proposed heart disease detection system is implemented within the Anaconda environment.

5. Jupyter notebook

   The Jupyter Notebook is an open-source web application that allows developers to create and share documents. Jupiter notebook works in most web browsers and allows developers to mix code with descriptive text in a simple and easy-to-understand way. This also comes with an Anaconda navigator. In this project, Jupyter notebook is used as IDE. Here codes can be run line by line and can create graphs, notes, and comment proper manner. That's why the Jupyter notebook is used.

6. GitHub

   GitHub is a provider of Internet hosting for software development and version control using Git. It simplifies the process of working with team members and makes it easy to collaborate on projects. In this project, we hope to work on files and easily merge our changes with the master branch of the project.

7. Moqups

   Moqups is a visual collaboration tool that includes a whiteboard, diagram, and design features in a single. it is an online app and can be for free. It is used to create a wide range of visual materials for any project. In this project, Moqups will be used to create both mobile applications and heart disease detection system user interfaces.

8. JavaScript

JavaScript is a programming language that is commonly used in web development. It is a high-level, interpreted language that is designed to be easy to use and understand. JavaScript is primarily used to create dynamic and interactive websites, but it is also used in a variety of other applications, such as mobile apps and desktop applications. JavaScript is often used in combination with HTML and CSS to create web pages that are interactive and engaging for users.

9. React

React is a JavaScript library for building user interfaces. React is a popular choice for building user interfaces because it is easy to learn, has a strong developer community, and is well-documented.

10. Fast API

FastAPI is a modern, fast, web framework for building APIs with Python based on standard Python-type hints. It is designed to be easy to use and provide high performance and adapt to the latest Python developments.

FastAPI is suitable for building a wide range of APIs, from simple to complex, and is particularly well-suited for building high-performance microservices. It has a strong developer community and is actively maintained and developed. If someone is looking to build a fast and efficient API with Python, FastAPI is a great choice. According to our Project, we use a python-based model and a React-based front-end. Also, we want to deal with large datasets and fast, efficient API connections. That's why we selected a fast API.

11. Firebase

Firebase is a cloud-based NoSQL database service that is provided by Google. It is designed to be easy to use and to provide a flexible and scalable solution for storing and syncing data across multiple devices and platforms. Real-time synchronization, Offline support, and Security are key features of the Database. In our project, we use the collection of the dataset and store user information and E.C.G. images. That's why we selected Firebase as a database.

12. HTML & CSS

    We must use HTML and CSS to implement interfaces of the web application. HTML and CSS are the two most fundamental programming languages used in web development. Without them, we would not have websites. Knowing a bit of HTML and CSS can help even if you don't want to become a web developer. A web application is created to carry out the project and it should be designed in such a way that it is easy for anyone to use from anywhere. HTML & CSS are also chosen because the webpage is easy to maintain and integrates with other languages.

13. Visual Studio Code

    Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux, and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Visual Studio Code was found while focusing on an application that has an easy-to-use and friendly interface with a developer for web page development.

7.3Challenges faced during the implementation

- Difficulty in finding a dataset in the Sri Lanka context

As patients' data are sensitive data to get them, we had to follow a long procedure at which Sri Lanka government hospital.

- Difficulty to get domain knowledge

Since we are programmers, the knowledge of doctors we had to learn the patterns that doctors used to identify a normal heartbeat from a person who has a heart disease.

- Difficulty to find a suitable database

In the first stage, we selected an SQL database for the project. But when we are implementing the project, we realize it's not matched our project. Because we wanted a database model to make fast communication with interfaces. So, we selected MongoDB as the database model. But when we store images in that model, it's difficult to convert form data. So finally, we moved the firebase database to solve all these problems. Firebase database provides a fast and efficient database service for our project.

7.4 Summary

.

There are so many technologies to implement our application. Among them, we had to decide the most tending and easiest way to implement our application. On other vises it may be very difficult, it must take more time to implement, and it may be not a useful full application. We had got a good idea about the technologies for implementing this application. By using these technologies, we think our application may be very useful and performance may be very high.

Chapter 8

Testing & Evaluation Chapter

8.1 Introduction

We will discuss the testing and assessment portion of our project in this chapter. We cover two primary subjects in this chapter's discussion. Test cases, test coverage, and test plans. The primary goal of testing and assessment is to create a flawless product that meets the needs of end customers. Additionally, a development company should place greater emphasis on it. Therefore, a more important phase of the software development life cycle is product testing. So let's talk about how the chapter on testing and assessment is carried out.

The application was tested using both manual and automated testing methods. To acquire a comprehensive understanding of how the web application was implemented, each functionality was tested independently, and each associated screenshot was provided. To test the application, sample test data were used. The source code was tested to find flaws using the white box testing methodology. Utilizing several dimensions, the database, some functionalities' capabilities, and the user interface was assessed. Testing was given precedence after the development cycle since it is a crucial stage in the software development lifecycle.

8.2. Testing types

8.2.1 Testing types

There are two types of testing called Manual testing and automation testing. Manual testing was done by humans, they may be developers, testers, or end users. Automation testing is done by using specialized automation tools. So, Manual testing was used by us instead of automation testing.

In manual testing, key features of our project were validated by us such as User registration and User Sign-in.

8.2.2 Testing methods

Blackbox testing

In our case, many complex modules are integrated so, some testing actions were difficult for us. A black box test was performed without code scanning.

As well as in our system some modules include more complex coding segments, So black box testing was performed by us for that kind of complex module such as the CNN model.

Whitebox testing

White box testing is also called clear box testing, open box testing, and code base testing. White box testing was performed by us for some modules which have less complex code segments. Code structure, design, and input-output flow were considered.

Firstly we understood the source code of the particular module. Then we identify if there are any security issues with code segments. After that, we make sure proper data flow is performed through the selected module.

So bugs were identified by us and re-assigned the code to get only numbers from users to this field. Otherwise, it will occur error msg.

unit testing was performed by us for testing purposes. Blocks of code were selected by us from selected modules and performed tests.

8.3. Testing levels

Functional testing

Unit testing

It was done at the development stage by testing a small block of codes. White box testing was used by our team to perform unit testing because we could identify errors in the initial stages. As well as we can do changes at this level.

Each module was selected by us and modules were divided into small units and then tested.

Integration testing

After performing the unit testing process, Our modules were integrated by us. At this level ensured by us that all components are working perfectly and that they have good interaction between modules.

We can find several methodologies to perform integration testing

- Top-down integration
- Bottum-up integration
- Sandwich integration
- Big bang integration

The bottom-up methodology was used by our integration for testing. Firstly it tests lower-level modules and gradually moves to higher-level modules. We use this methodology because all our modules are available for testing.

Non-functional testing

At this testing level, we test our system for its behavioral factors such as

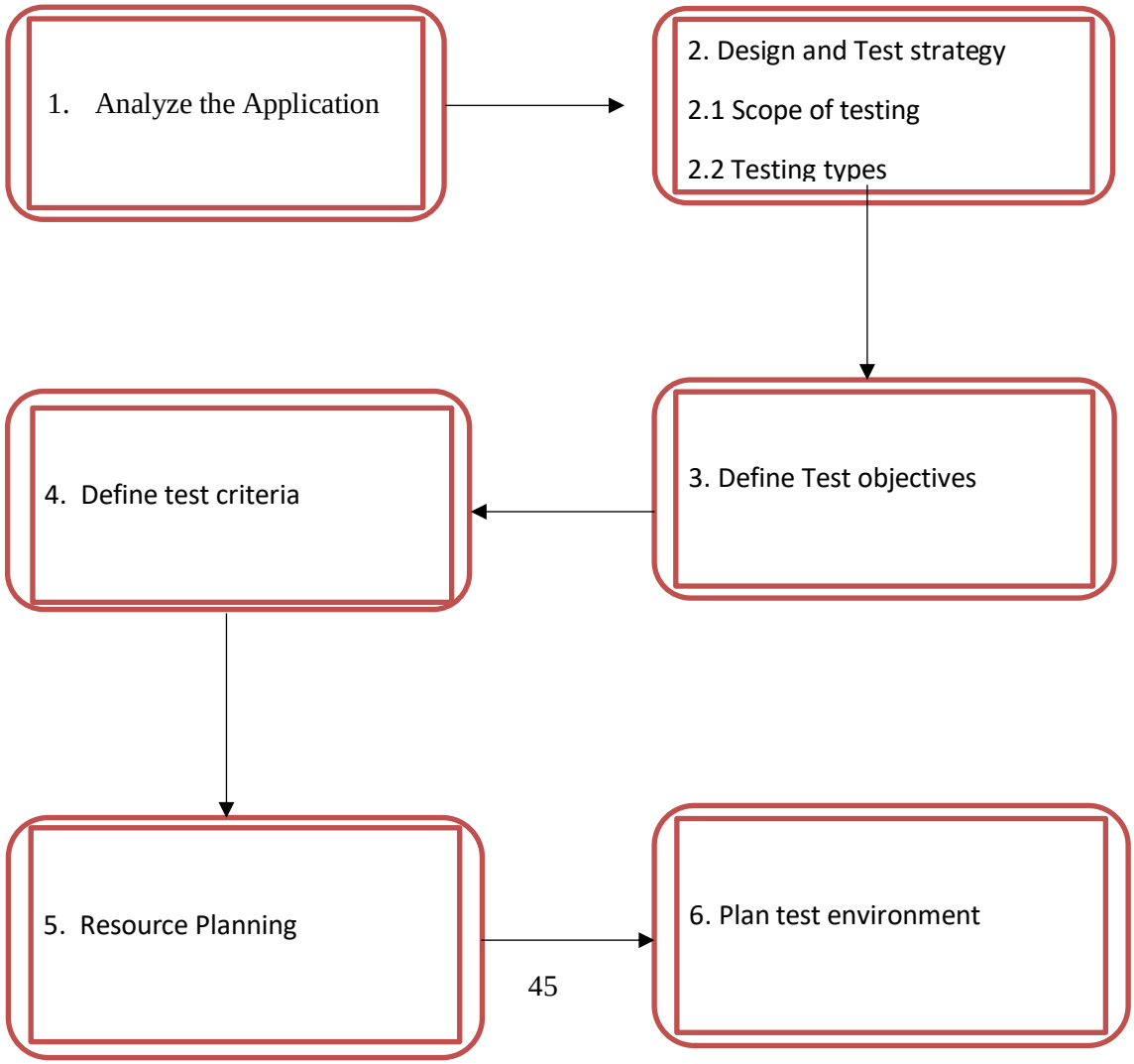
- Performance
- Usability
- Availability
- Compatibility
- Security

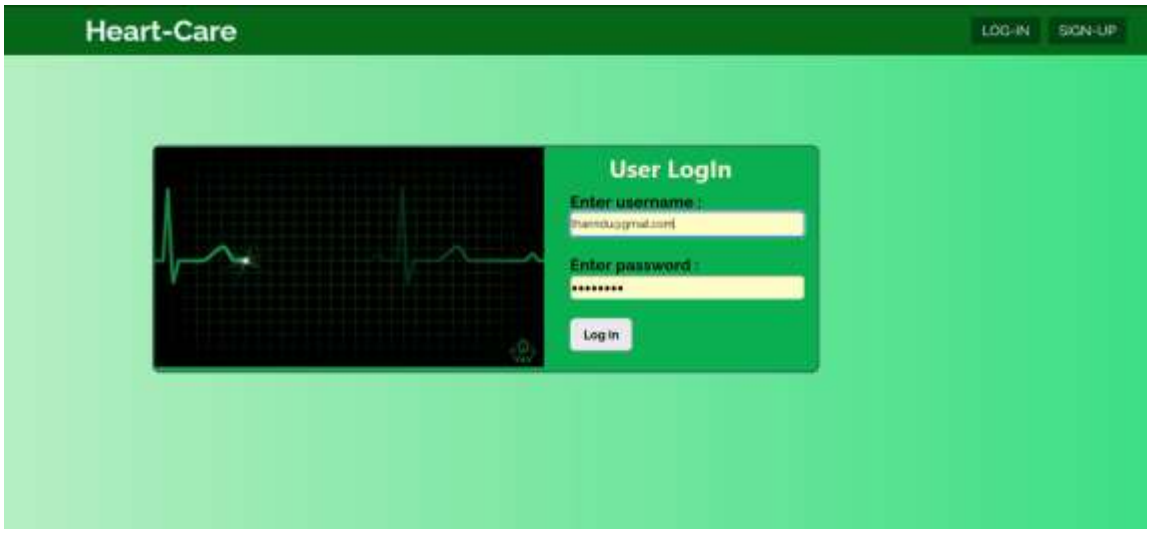
Security testing

Security Testing is a type of Software Testing that identifies vulnerabilities, threats, and risks in a software application and prevents malicious attacks from outsiders. To perform that we use some test scenarios for security testing. Some of them are,

- A password should be in an encrypted format
- Application or System should not allow invalid users

Test Plan





- Testing the Login function in the web application.

  Test data

  Username – tharindu @gmail.com
  Password – tharindu

- Testing the Signup function in the web application

  Test Data

  User Email- tharindu @gmail.com
  Username – tharindu
  User Password – 123&abc
  User Contact- 0766008863

Chapter 9

Conclusion

9.1 Introduction

The use of a deep neural network (DNN) architecture to diagnose heart patients through analysis of electrocardiogram (ECG) images. The DNN is trained to accurately determine whether a person has a normal or abnormal heartbeat based on the ECG image. The software allows a general practitioner to input an ECG image and receive a predicted result, which is also stored in a database This chapter aims to describe Limitations and Future works.

9.2 Limitation

• Difficulty in finding a dataset in the Sri Lanka context

As patients' data are sensitive data to get them, we had to follow a long procedure at which Sri Lanka government hospital.

• Difficulty to get domain knowledge

Since we are programmers, the knowledge of doctors we had to learn the patterns that doctors used to identify a normal heartbeat from a person who has a heart disease.

9.3 Future works

- In stage 2 we try to develop our model to identify what is the type of disease (Myocardial Infarction, Abnormal heartbeat,)
- Add more data and train the model for more accurate results
- Implement a mobile application to connect doctors and patients for monitoring patients

Doctors and Patients can log into the mobile app. Doctors can contact the patients remotely. The mobile app allows the patient to get predictor results. The patient can also enter the age, sex, weight, blood glucose, cholesterol level, etc.…and get health tips on those matters through the mobile app. And the system is very efficient and easy to handle. We hope to create more accurate, faster, and more efficient systems in future works.

9.4 Summery

This project proposes the use of a deep neural network (DNN) architecture to diagnose heart patients through the analysis of electrocardiogram (ECG) images. The DNN is trained to accurately determine whether a person has a normal or abnormal heartbeat based on the ECG image. The software allows a general practitioner to input an ECG image and receive a predicted result, which is also stored in a database. As future works of the project, hope to develop the mobile app associated with the system that allows patients to receive the predicted results and contact a doctor, as well as input their personal health information (such as age, sex, weight, blood glucose, cholesterol level) and receive health tips. The goal of the system is to create a more accurate, efficient, and faster method of diagnosing heart patients and contact the patients to give some health tips.

Reference

[1] A. H. Khan, M. Hussain, and M. K. Malik, "Cardiac disorder classification by electrocardiogram sensing using deep neural network," *Complexity*, vol. 2021. 2021, doi: 10.1155/2021/5512243.

[2] P. Kalpana, S. Shiyam Vignesh, L. M. P. Surya, and V. Vishnu Prasad, "Prediction of Heart Disease Using Machine Learning," *J. Phys. Conf. Ser.*, vol. 1916, no. 1, 2021, doi: 10.1088/1742-6596/1916/1/012022.

[3] A. U. Haq, J. P. Li, M. H. Memon, S. Nazir, R. Sun, and I. Garciá-Magarinõ, "A hybrid intelligent system framework for the prediction of heart disease using machine learning algorithms," *Mobile Information Systems*, vol. 2018. 2018, doi: 10.1155/2018/3860146.

[4] S. Mohan, C. Thirumalai, and G. Srivastava, "Effective heart disease prediction using hybrid machine learning techniques," *IEEE Access*, vol. 7, pp. 81542–81554, 2019, doi: 10.1109/ACCESS.2019.2923707.

[5] J. Patel, A. A. Khaked, J. Patel, and J. Patel, "Heart Disease Prediction Using Machine Learning," *Lect. Notes Networks Syst.*, vol. 203 LNNS, pp. 653–665, 2021, doi: 10.1007/978-981-16-0733-2_46.

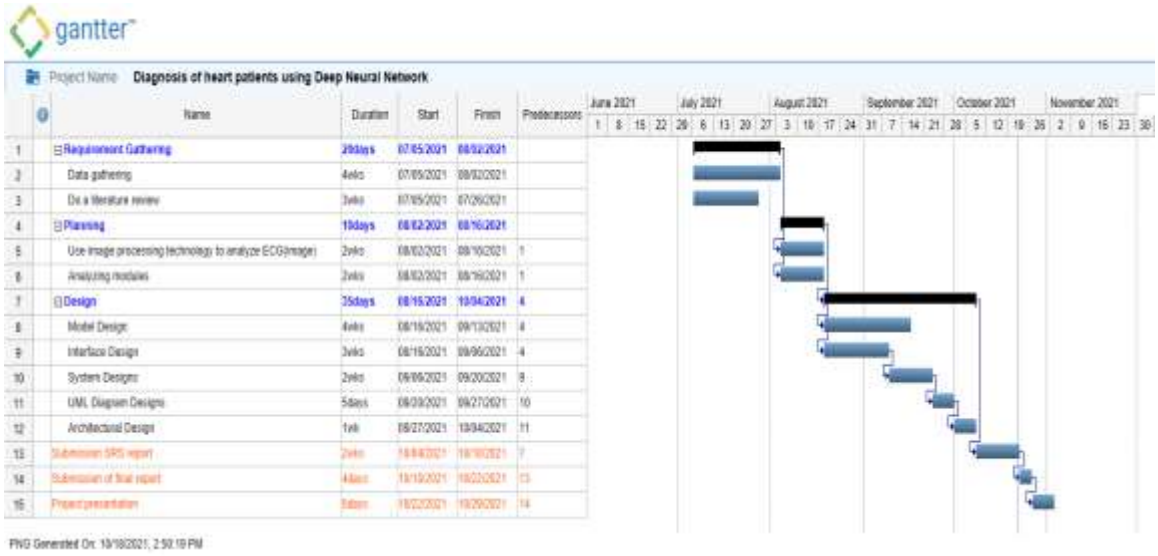
[6] C. Venkatesan, P. Karthigaikumar, and S. Satheeskumaran, "Mobile cloud computing for ECG telemonitoring and real-time coronary heart disease risk detection," *Biomedical Signal Processing and Control*, vol. 44. pp. 138–145, 2018, doi: 10.1016/j.bspc.2018.04.013.

[7] B. D. Ravichandran and P. Keikhosrokiani, "An Emotional-Persuasive Habit-Change Support Mobile Application for Heart Disease Patients (BeHabit)," *Lecture Notes on Data Engineering and Communications Technologies*, vol. 72. pp. 252–262, 2021, doi: 10.1007/978-3-030-70713-2_25.

[8] J. D. Hemanth, U. Kose, O. Deperlioglu, and V. H. C. de Albuquerque, "An augmented reality-supported mobile application for diagnosis of heart diseases," *Journal of Supercomputing*, vol. 76, no. 2. pp. 1242–1267, 2020, doi: 10.1007/s11227-018-2483-6.

[9] N. D. K. G. Dharmasiri and S. Vasanthapriyan, "Approach to heart diseases diagnosis and monitoring through machine learning and iOS mobile application," *18th International Conference on Advances in ICT for Emerging Regions, ICTer 2018 - Proceedings*. pp. 407–412, 2019, doi: 10.1109/ICTER.8615504.

Appendix


gantter
Project Name: Diagnosis of heart patients using Deep Neural Network
Name
Duration
Start
Finish
Predecessors
June 2021
July 2021
August 2021
September 2021
October 2021
November 2021
Requirement Gathering
Data gathering
Do a literature review
Planning
Use image processing technology to analyze ECG(Image)
Analyzing modules
Design
Model Design
Interface Design
System Designs
UML Diagram Designs
Architectural Design
Submission SRS report
Submission of final report
Project presentation
PNG Generated On: 10/18/2021, 2:50:19 PM

National Diploma in Information Technology IT2407 Project II Supervisor certification for the final report submission.

The final report of the group no 3 titled Diagnosis of Heart Patients Using Deep Neural Network was submitted for supervision and it is observed that all the given comments are successfully added in the final version. (Note: Please indicate the group number and name in the above spaces.) I checked and the hard copy of the certification with the signature will be provided to the process Dr.Mrs. K. Galappaththi……. Signature of the Supervisor Additional remarks that can be given by the supervisor to support the examiner evaluation. It was asked to add an abstract, acknowledgment, and more test cases to the draft version. All the changes have been added to the new version successfully.