MVJ COLLEGE OF ENGINEERING, BENGALURU – 560067

(An Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

PROJECT REPORT ON

REAL TIME ARDUINO-BASED CARDIAC DISEASE DETECTION USING NEURAL NETWORK

Submitted By:

KOMMA SANDEEP REDDY	**1MJ19EC063**
LAKSHMAN REDDY M	**1MJ19EC064**
M SANTOSH	**1MJ19EC065**
MOHAMMED MAAZ	**1MJ19EC077**

Under the Guidance of

Prof. Sayantam Sarkar

Assistant Professor

Department of Electronics and Communication Engineering

MVJ College of Engineering

In Partial fulfillment for the award of degree

of

Bachelor of Engineering

In

Electronics and Communication Engineering

2022-23

MVJ COLLEGE OF ENGINEERING, BENGALUR-560067

(An Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the major project titled **"REAL TIME ARDUINO-BASED CARDIAC DISEASE DETECTION USING NEURAL NETWORK"** was carried out by **KOMMA SANDEEP REDDY (1MJ19EC063), LAKSHMAN REDDY M (1MJ19EC064), M SANTOSH (1MJ19EC065) and MOHAMMED MAAZ (1MJ19EC077)** in partial fulfilment for the award of degree of Bachelor of Engineering in Electronics and Communication Engineering of the Visvesvaraya Technological University, Belagavi during the year 2022-2023. It is certified that all corrections / suggestions indicated during the internal assessment have been incorporated in the Report deposited in the department library. The report has been approved as it satisfies the academic requirements in respect of project work prescribed by the institution for the said degree.

Signature of Guide	Signature of HOD	Signature of Principal
Prof. Sayantam Sarkar	**Dr. I. Hameem Shanavas**	**Dr. M. Brindha**
Assistant Professor	Professor & HOD	Principal in charge
Department of ECE,	Department of ECE	MVJ College of Engineering
MVJ College of Engineering	MVJ College of Engineering	Bengaluru
Bengaluru	Bengaluru	

Name of Examiners	Signature with Date
1.	
2.	

MVJ COLLEGE OF ENGINEERING, BENGALURU-560067

(An Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

DECLARATION

We, **KOMMA SANDEEP REDDY (1MJ19EC063), LAKSHMAN REDDY M (1MJ19EC064), M SANTOSH (1MJ19EC065) and MOHAMMED MAAZ (1MJ19EC077)** student of Eight Semester B.E., Department of Electronics and Communication Engineering, MVJ College of Engineering, Bengaluru-560067, hereby declare that the Technical major project Titled **"REAL TIME ARDUINO- BASED CARDIAC DISEASE DETECTION USING NEURAL NETWORK"** has been carried out by us and submitted in partial fulfilment for the award of the degree of Bachelor of Engineering in Electronics and Communication Engineering during the year 2022-2023. Further we declare that the content of the report has not been submitted previously by anybody for the award of any degree or diploma to any other University.

Place: Bengaluru

Date:

NAME	USN	Signature
KOMMA SANDEEP REDDY	**1MJ19EC063**	
LAKSHMAN REDDY M	**1MJ19EC064**	
M SANTOSH	**1MJ19EC065**	
MOHAMMED MAAZ	**1MJ19EC077**	

ACKNOWLEDGEMENT

The satisfaction and euphoria that accomplishes the successful completion of any task would be incomplete without the mention of the people who make it possible.

I am thankful to **The Management** of MVJ College of Engineering, Bengaluru, for their continuous support and encouragement for carrying out the major project work.

I thank **Dr. M. Brindha**, Principal, MVJCE, Bengaluru for the encouragement and valuable suggestions.

I would like to express my sincere gratitude to **Dr. M. A. Lourdu Antony Raj,** Registrar & Controller of Examination, MVJCE, Bengaluru, for persistent guidance.

I thank **Dr. I. Hameem Shanavas,** Professor and HOD, Department of Electronics and Communication Engineering, MVJCE, Bengaluru, for being a constant support and providing all the facilities that was needed in the major project work.

I avail this opportunity to express profound sense of gratitude and thank to our guide. We would like to thank **Prof. Sayantam Sarkar,** Associate Professor, Department of Electronics and Communication Engineering, MVJCE, Bengaluru, for her wholehearted support, suggestions, and invaluable advice throughout our major project work and also for the help in the preparation of this report.

I also express our gratitude to our panel members **Dr. Utsab Banerjee,** Assistant Professor and Prof. **Sayantam Sarkar,** Assistant Professor, Department of Electronics and Communication Engineering, MVJCE, Bengaluru, for their valuable comments and suggestions.

I am in debt to the support of all the teaching and non-teaching members of the Department of ECE, MVJCE, for their kind help and co-operation, throughout the major project.

M SANTOSH

ABSTRACT

In order to detect and treat cardiovascular illnesses, electrocardiograms (ECGs) are commonly utilized. These tests record the real-time electrical activity of the heart through leads attached to the patient's body. However, the existing ECG monitoring equipment is often bulky and expensive. While some portable ECG devices are available, they do not provide information about the probable condition of the heart. To address this, a novel approach is proposed in this paper, which utilizes ECG signals to classify heart conditions. The real-time ECG signals are acquired using an AD8232 sensor and then classified using a Random Forest Neural Network [1]. The implementation of this approach has shown an impressive accuracy of 99.43%. This research offers a promising advancement in the field of ECG monitoring, providing a more efficient and accurate method for detecting and diagnosing heart conditions.

TABLE OF CONTENTS

PAGE NO

CERTIFICATE……………………………………………………………………........I

DECLARATION…………………………………………………………………….II

ACKNOWLEDGEMENT………………………………………………………….III

ABSTRACT…………………………………………………………………………IV

TABLE OF CONTENTS………………………………………………………....V

LIST OF FIGURES……………………………………………………………...VII

REAL TIME ARDUINO- BASED CARDIAC DISEASE DETECTION USING NEURAL NETWORK

CHAPTER 1 INTRODUCTION …...…………………………………………………...1

1.1 Introduction……………………………………………………………… 2

1.2 Cardiovascular diseases.……………………………………………….2

1.2.1 Coronary Artery Disease (CAD)…………………………2

1.2.2 Stroke……………………………………………………..2

1.2.3 Heart Failure……………………………………………...3

1.2.4 Arrhythmias………………………………………………3

1.2.5 Heart Valve Disease……………………………………...3

1.2.6 Congenital Heart Disease………………………………...3

1.2.7 Myocardial infarction…………………………………….3

1.3 WHO report on cardiovascular diseases………………………………4

1.4 Electrocardiogram (ECG)……………………………………………..4

1.4.1 ECG Waveform………………………………………….5

1.4.2 ECG Data Application…………………………………...6

CHAPTER 2 LITERATURE SURVEY……………………………………………… 7

CHAPTER 3 NEURAL NETWORK…....…………………………………………….10

3.1 Introduction……………………………………………………………...11

3.2 Algorithms and its Types……………………………………………...12

3.2.1 Definition of Algorithm……………………………...12

3.2.2 Types of Algorithms…………………………………….12

3.3 Comparing all Algorithms……………………………………………14
3.3.1 Logistic Regression……………………………………..15
3.3.2 K-Nearest Neighbor……………………………………15
3.3.3 Support Vector Machine………………………………..15
3.3.4 Naive Bayes…………………………………………….16
3.3.5 Random Forest …………………………………………16

CHAPTER 4 HARDWARE…...…………………………………………………… 18
4.1 Methodology……………………………………………………….19
4.2 Hardware Specification……………………………………………19
4.2.1 AD8232 Sensor………………………………………20
4.2.2 NodeMCU ESP8266…………………………………20
4.2.3 OLED Display Module………………………………21
4.3 Hardware Implementation…………………………………………... 22

CHAPTER 5 SOFTWARE………………………………………………………… 23
5.1 Introduction to Visual Studio ……………………………………… 24
5.2 Visual Studio Applications……………………………………………25
5.3 Software Implementation…………………………………………….26
5.3.1 Visual Studio Interface………………………………… 26
5.3.2 Data Preparation and Preprocessing…………………… 27
5.3.3 Data feature Extraction & Data Engineering………….. 28

CHAPTER 6 RESULT AND ANALYSIS…………………………………………… 30
6.1 Result….…………………………………………………………...31
6.2 Analysis….………………………………………………………...31

CHAPTER 7 CONCLUSION AND FUTURE SCOPE………………………………33
7.1 Conclusion….……………………………………………………...34
7.1 Future Scope….……………………………………………………34

CHAPTER 8 CERTIFICATE...……………………….…………………………….35

CHAPTER 9 REFERENCES...…………………………..………………………….36

LIST OF FIGURES

Fig No.	Name	Page No.
1.1	ECG Waveform	05
3.1	Basic architecture of Artificial Neuron	11
3.2	Jupyter Notebook Software	14
3.3	Logistic Regression	15
3.4	K-Nearest Neighbor	15
3.5	Support Vector Machine	16
3.6	Naive Bayes	16
3.7	Random Forest	17
4.1	Block Diagram of the Project	19
4.2	AD8232 ECG Sensor	20
4.3	NodeMCU ESP8266	20
4.4	1.3-inch OLED Display Module	21
4.5	Hardware Setup	22
4.6	ECG Graph Data	22
5.1	Visual Studio	24
5.2	Importing Libraries for Streamlit Application	26
5.3	Importing Libraries for ECG Data Extraction	26
5.4	Uploading the ECG Data	27
5.5	Converting ECG Data into Gray Scale Image	27
5.6	Dividing the Leads	28
5.7	Preprocessed Leads	28
5.8	Extracting Signals	29
5.9	Converting to 1D Signal	29
5.10	Performing Dimensionality Reduction	29
6.1	Comparing the result of all the Algorithms	31
6.2	The ECG Signal	32
6.3	Extracting Signals	32
6.4	Predicted Output	32

CHAPTER-1
INTRODUCTION

CHAPTER-1

INTRODUCTION

1.1 Introduction

Cardiovascular diseases (CVDs) are the leading cause of death worldwide, accounting for millions of lives lost each year. Early detection and timely intervention play a critical role in preventing fatalities and improving patient outcomes. The advancements in technology, particularly in the fields of electronics and artificial intelligence (AI), have opened up new possibilities for developing innovative solutions to address this global health challenge.

One such promising approach is the real-time Arduino-based cardiac disease detection system using neural networks. This project aims to leverage the power of Arduino microcontrollers and neural networks to create an efficient and accurate tool for diagnosing cardiovascular conditions. By combining hardware and software components, this solution enables continuous monitoring of vital signs, early detection of abnormalities, and prompt medical intervention, ultimately saving lives.

1.2 Cardiovascular diseases

Cardiovascular diseases (CVDs) are a group of disorders that affect the heart and blood vessels. These diseases can involve various components of the cardiovascular system, including the heart, arteries, veins, and capillaries. They are characterized by abnormalities in the structure and function of these components, which can lead to serious health consequences.

There are several types of cardiovascular diseases, including:

1.2.1 Coronary Artery Disease (CAD)

This is the most common type of cardiovascular disease. It occurs when the coronary arteries, which supply oxygen and nutrients to the heart muscle, become narrowed or blocked due to the buildup of plaque (atherosclerosis). CAD can lead to angina (chest pain) and heart attacks.

1.2.2 Stroke

A stroke occurs when the blood supply to the brain is interrupted or reduced, leading to the death of brain cells. Ischemic stroke, the most common type, is caused by a blocked or narrowed blood vessel supplying the brain. Hemorrhagic stroke occurs when a blood vessel in the brain ruptures and causes bleeding.

1.2.3 Heart Failure

Heart failure happens when the heart's pumping capacity becomes weakened, making it difficult for the heart to effectively pump blood to meet the body's needs. It can result from various conditions such as coronary artery disease, high blood pressure, or heart muscle damage.

1.2.4 Arrhythmias

Arrhythmias are abnormalities in the heart's rhythm or rate. They can cause the heart to beat too fast (tachycardia) or too slow (bradycardia) or have irregular rhythms. Arrhythmias can disrupt the heart's normal pumping function and increase the risk of complications such as blood clots or cardiac arrest.

1.2.5 Heart Valve Disease

Heart valve diseases involve abnormalities in the valves that control blood flow within the heart. These valves can become damaged or malfunction, leading to conditions such as valve stenosis (narrowing) or valve regurgitation (leakage).

1.2.6 Congenital Heart Disease

Congenital heart diseases are structural abnormalities of the heart that are present at birth. They can involve defects in the heart's walls, valves, or blood vessels. Congenital heart diseases can range from mild, requiring no treatment, to severe, requiring surgical interventions.

1.2.7 Myocardial infarction

Myocardial infarction, commonly known as a heart attack, is a serious medical condition that occurs when the blood supply to a part of the heart is blocked, usually due to a blood clot. This blockage prevents oxygen and nutrients from reaching the affected area, leading to damage or death of the heart muscle. Myocardial infarction can manifest with symptoms such as chest pain, shortness of breath, nausea, and sweating. Immediate medical attention is crucial to minimize the extent of damage and prevent life-threatening complications.

1.3 WHO report on cardiovascular diseases

Cardiovascular diseases are a group of disorders that affect the heart and blood vessels, including conditions such as coronary artery disease, heart attacks, stroke, and heart failure. These diseases have been a major global health concern for several decades, causing a significant burden on individuals, families, communities, and healthcare systems worldwide.

According to the WHO, cardiovascular diseases are the leading cause of death globally. In its most recent report published in 2019, cardiovascular diseases accounted for approximately 17.9 million deaths, representing 32% of all global deaths. This staggering number highlights the urgent need for prevention, early detection, and effective management of CVDs.

It is important to note that the burden of cardiovascular diseases is not evenly distributed across the globe. The WHO reports that around 85% of cardiovascular disease-related deaths occur in low- and middle-income countries. This disparity can be attributed to various factors, including differences in access to healthcare, socioeconomic conditions, lifestyle choices, and the prevalence of risk factors such as tobacco use, unhealthy diets, physical inactivity, and high blood pressure.

The impact of cardiovascular diseases extends beyond mortality. Survivors of heart attacks and strokes often experience long-term disability and a reduced quality of life. Additionally, CVDs place a significant economic burden on individuals, families, and healthcare systems, with costs related to treatment, hospitalization, rehabilitation, and lost productivity.

To address the global burden of cardiovascular diseases, the WHO and other health organizations emphasize the importance of comprehensive strategies that focus on prevention, early detection, and management. These strategies include promoting healthy lifestyles, raising awareness about risk factors, improving access to healthcare services, strengthening healthcare systems, and conducting research to advance medical knowledge and interventions.

1.4 Electrocardiogram (ECG)

ECG (Electrocardiogram) data refers to the electrical signals produced by the heart during its contraction and relaxation cycles. It is recorded using specialized medical devices called electrocardiographs or ECG machines. These machines measure the electrical activity of the heart and generate a visual representation of the heart's electrical impulses over time, known as an ECG waveform or ECG tracing.

The ECG data is obtained by placing multiple electrodes on specific locations of the body, typically on the chest, arms, and legs. These electrodes detect the electrical signals generated by the heart, which are then amplified, filtered, and displayed on the ECG machine or transmitted to a computer system for further analysis.

1.4.1 ECG Waveform

The ECG waveform consists of different components that represent specific events during the cardiac cycle.

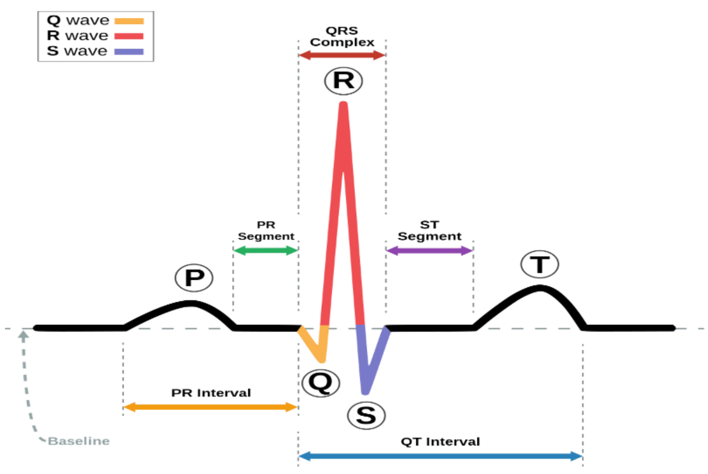


Fig 1.1 ECG Waveform

The main features of the ECG waveform include:

1. P Wave: The P wave represents the depolarization (contraction) of the atria, the heart's upper chambers.
2. QRS Complex: The QRS complex indicates the depolarization of the ventricles, the heart's lower chambers. It includes the Q wave, R wave, and S wave.
3. T Wave: The T wave represents the repolarization (relaxation) of the ventricles.

Additional components and intervals on the ECG waveform provide valuable information about the heart's electrical conduction, rhythm, and any abnormalities that may be present.

1.4.2 ECG Applications

ECG data is widely used in medical settings for various purposes:

1. **Diagnosis:** ECG data plays a crucial role in diagnosing a wide range of heart conditions. It helps identify abnormalities in the heart's rhythm (arrhythmias), signs of myocardial ischemia (reduced blood flow to the heart), conduction abnormalities, structural abnormalities, and other cardiac disorders. By analyzing the characteristics of the ECG waveform, healthcare professionals can make accurate diagnoses and determine appropriate treatment plans.
2. **Monitoring:** ECG data is utilized for continuous or intermittent monitoring of patients with known or suspected heart conditions. This includes long-term monitoring with devices such as Holter monitors or event recorders to capture and analyze ECG data over extended periods. Monitoring ECG data helps detect and evaluate transient or intermittent abnormalities that may not be captured during a standard ECG test, aiding in the management of various cardiac conditions.
3. **Risk Assessment:** ECG data is used as part of risk assessment protocols to evaluate an individual's susceptibility to certain cardiac events or diseases. By analyzing specific parameters in the ECG waveform, healthcare providers can assess the risk of developing conditions such as coronary artery disease, arrhythmias, or sudden cardiac arrest. This information guides preventive strategies and lifestyle modifications to mitigate risks.
4. **Treatment Monitoring:** ECG data is employed to assess the effectiveness of certain cardiac treatments, such as medications or interventions. Serial ECG measurements can track changes in the ECG waveform, helping healthcare professionals evaluate the response to treatment and make adjustments as necessary.
5. **Fitness Tracking:** The system can be integrated into wearable fitness devices to provide real-time cardiac health monitoring during exercise or physical activities. It can help individuals optimize their workouts and prevent overexertion.
6. **Research and Education:** The system can be utilized in research studies to collect data on cardiac health parameters in real-world scenarios. It can also be used in educational settings to teach students about cardiac physiology and diseases.
7. **Ambient Assisted Living:** In elderly care or assisted living environments, the system can contribute to the monitoring of residents' cardiac health, alerting caregivers or medical professionals in case of emergencies or abnormalities.

CHAPTER-2
LITERATURE SURVEY

CHAPTER-2

LITERATURE SURVEY

The purpose of literature review is to gain an understanding of the existing research and debates relevant to a particular topic or area of study.

Kenil Shah et al [1] "Detection of Heart Defects using Electrocardiogram (ECG)" investigate the utilization of electrocardiogram (ECG) for the identification of heart defects. The authors conduct a thorough examination of previous research pertaining to ECG-based heart defect detection, with a particular emphasis on the techniques, algorithms, and methodologies employed. The survey offers a summary of the progress made in this area, outlining the obstacles, constraints, and potential avenues for future development.

Published in March 2020, "ECG Monitoring System" by Mohamed Adel et al [2] explores the creation of a monitoring system specifically designed for electrocardiogram (ECG) analysis. The paper offers a comprehensive review of current research and progress in the field of ECG monitoring. It delves into several key aspects including wearable ECG devices, wireless communication protocols, techniques for signal processing, and methods for data analysis.

Ray et al [3] the author provides a comprehensive literature survey on machine learning techniques. In the paper "A Quick Review of Machine Learning Algorithms". The survey covers various algorithms, their applications, and highlights their strengths and limitations. It offers valuable insights into the theoretical foundations, learning mechanisms, and optimization strategies of popular algorithms such as decision trees, support vector machines, neural networks, and ensemble methods. The paper serves as a concise yet informative resource for researchers and practitioners seeking to understand the diverse landscape of machine learning algorithms.

"Wireless ECG Monitoring System for Telemedicine Application", Ahmed Aboalseoud, et al [4] present a novel wireless ECG monitoring system designed for telemedicine purposes. Published in the 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS), the paper includes a comprehensive literature survey that investigates the current state of research and technological advancements in the field of ECG monitoring and its applications in telemedicine.

"IoT: Electrocardiogram (ECG) Monitoring System" by Kristine Joyce P. Ortiz et al [5] explores the development of an Internet of Things (IoT) based ECG monitoring system. The authors discuss the importance of remote ECG monitoring for timely diagnosis and treatment of cardiovascular diseases. They present a comprehensive literature survey, covering various aspects of ECG monitoring systems, including signal acquisition, processing, transmission, and analysis.

"Real-Time ECG Monitoring System Using Arduino and LabVIEW", Deepak Yadav et al [6] provide an overview of the relevant literature. The authors concentrate on the creation of an electrocardiogram (ECG) monitoring system utilizing Arduino and LabVIEW. They explore existing research and technologies associated with ECG monitoring systems, while also discussing the strengths and limitations of different approaches in the field.

Published in the Journal of Physics: Conference Series in 2020, the paper titled "Development of ECG sensor using Arduino Uno and e-health sensor platform: Mood detection from heartbeat" [7] investigates the utilization of an ECG sensor, Arduino Uno, and the e-health sensor platform for the analysis of heartbeats to detect mood variations. The paper includes a comprehensive literature survey that encompasses a range of pertinent research in the field. This includes investigations on mood detection using ECG signals, wearable health monitoring devices, and the development of Arduino-based sensors.

In the May 2018 issue of JETIR, Mr. Ved Prakasha and Mr. Manoj Kumar Pandey, Assistant Professors at the ECE Department, Amity University, Gurgaon, published a paper titled "Heart Rate Monitoring System" [8]. The paper focuses on conducting a literature survey on heart rate monitoring. They explore different approaches, including wearable devices, optical sensors, and electrocardiogram (ECG) systems, while also highlighting the advantages and limitations associated with each approach.

In their 2018 paper titled "Arduino-based ECG Monitoring System" [9] K. Lakshmi Bhanu et al [9] Bhargavi conduct a comprehensive literature survey on the subject. The authors explore a range of existing studies that focus on ECG monitoring systems utilizing Arduino technology. The survey encompasses various aspects, including the design and implementation of Arduino-based ECG systems, techniques for signal processing, wireless transmission approaches, and methods for data visualization.

CHAPTER-3
NEURAL NETWORK

CHAPTER-3

NEURAL NETWORK

3.1 Introduction

A neural network is a computational model inspired by the structure and functioning of the human brain. It is a collection of interconnected nodes, called artificial neurons or "neurons," that work together to process and analyze complex data. Neural networks are widely used in machine learning and artificial intelligence applications.

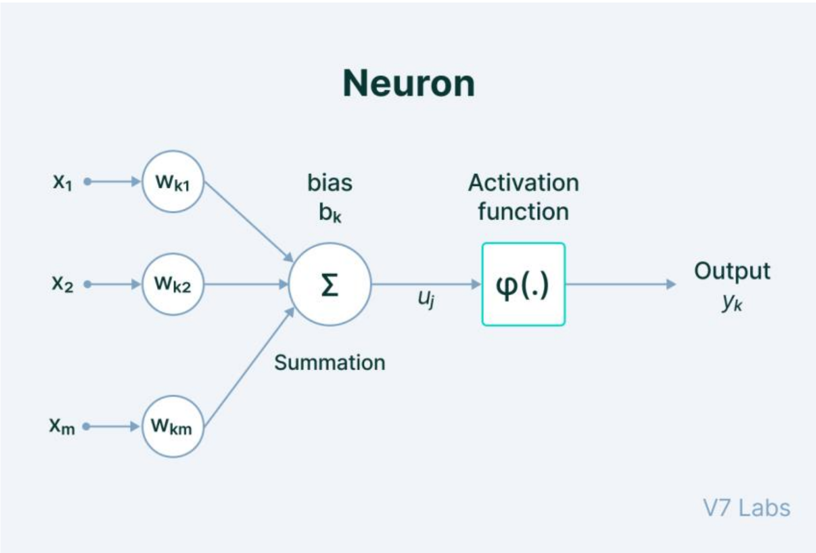


Fig 3.1 Basic architecture of Artificial Neuron [1]

The basic building block of a neural network is the artificial neuron, which receives input signals, performs calculations on them, and produces an output signal. The input signals are weighted and combined, and then a non-linear activation function is applied to the sum. The resulting output is passed on to other neurons in the network.

Neural networks consist of multiple layers of interconnected neurons. The first layer is called the input layer, which receives the initial data. The last layer is the output layer, which produces the final output or prediction. In between, there can be one or more hidden layers, which perform intermediate computations and help the network learn complex patterns and relationships in the data.

During the training process, a neural network learns from a set of labeled examples or training data. It adjusts the weights of its connections iteratively using a process called backpropagation, which involves calculating the error between the predicted output and the desired output. By minimizing this error, the network gradually improves its ability to make accurate predictions or classifications.

Neural networks have shown great success in various tasks such as image and speech recognition, natural language processing, and recommendation systems. They have the ability to learn from large amounts of data, generalize from examples, and make predictions on unseen data.

3.2 Algorithms and its Types

Neural networks have gained significant popularity in the field of artificial intelligence and machine learning due to their ability to learn complex patterns from data. At the heart of a neural network lies a set of algorithms that define how the network processes information, adjusts its parameters, and makes predictions. These algorithms provide the computational framework that enables neural networks to learn and generalize from examples.

3.2.1 Definition of Algorithm

An algorithm in a neural network is a step-by-step procedure or a series of mathematical operations that govern the flow of data through the network and guide the learning process. These algorithms are responsible for tasks such as feeding input data through the network, computing weighted sums, applying activation functions, updating weights during training, and producing output predictions or classifications.

3.2.2 Types of Algorithms

There are different types of algorithms available for us to use in this project, and here are some of them are which we used to compare in this project:

(a). **Logistic Regression:** Logistic regression is a popular algorithm used for binary classification tasks [3]. It models the relationship between input features and the probability of belonging to a particular class. The algorithm uses a sigmoid function to map the linear combination of input features into a probability value between 0 and 1. During training, it adjusts the weights through techniques like gradient descent to minimize the difference between predicted probabilities and true class labels.

(b). **K-Nearest Neighbor:** The k-nearest neighbor (k-NN) classifier is a simple and intuitive algorithm used for classification tasks. It operates on the principle of similarity, where an unlabeled example is assigned, a label based on the labels of its k-nearest neighbors in the feature space [3]. The k-NN classifier works by computing the distance between the unlabeled example and all labeled examples in the dataset using a distance metric, commonly the Euclidean distance. It then selects the k examples with the smallest distances as the nearest neighbors. The predicted label for the unlabeled example is determined by a majority vote among the labels of these neighbors.

(c). **Support Vector Machine:** The Support Vector Machine (SVM) algorithm is a powerful and versatile machine learning algorithm used for both classification and regression tasks [3]. It aims to find an optimal hyperplane that separates data points belonging to different classes or predicts a continuous target variable. SVM operates by constructing a decision boundary in a higher-dimensional feature space, where each data point is represented as a vector. The algorithm seeks to maximize the margin, which is the distance between the decision boundary and the nearest data points of each class. This allows SVM to generalize well to unseen data and handle outliers effectively.

(d). **Naive Bayes:** The Naive Bayes algorithm is a probabilistic classifier that is widely used in machine learning for classification tasks [3]. It is based on Bayes' theorem, which calculates the probability of a certain event given prior knowledge or evidence. The algorithm calculates the probability of each class based on the feature values and then assigns the class with the highest probability as the predicted class. It is particularly useful when dealing with high-dimensional datasets and large feature spaces.

(e). **Random Forest:** The Random Forest algorithm is a versatile and powerful ensemble learning method widely used in machine learning for both classification and regression tasks [3]. It operates by combining multiple decision trees and making predictions based on their collective output. Each decision tree in the random forest is trained on a different subset of the training data and uses a random selection of features. During training, the algorithm creates a multitude of decision trees and aggregates their predictions to obtain the final result. This ensemble approach helps to improve the accuracy and robustness of the model while reducing the risk of overfitting.

3.3 Comparing all Algorithms

When comparing different algorithms in the field of machine learning and data analysis, the Jupyter software environment serves as a powerful and versatile tool. Jupyter provides an interactive computing platform that allows researchers, data scientists, and analysts to create and share documents containing live code, visualizations, and explanatory text. Its flexibility and integration with various programming languages, such as Python, R, and Julia, make it an ideal choice for algorithm comparison tasks.

Fig 3.2 Jupyter Notebook Software

One of the main advantages of using Jupyter for algorithm comparison is its ability to provide a reproducible and transparent analysis environment. With Jupyter notebooks, researchers can document and share their entire workflow, including code, data, visualizations, and textual explanations.

.

Jupyter also offers the convenience of an interactive interface, allowing users to execute code cells individually and see immediate results. This feature is particularly valuable when comparing different algorithms since it enables users to observe and compare their performance in real-time. Furthermore, Jupyter provides a wide range of libraries and tools specifically designed for machine learning and algorithm comparison.

In order to find the model with the highest accuracy for spotting cardiac problems, the Kaggle dataset was obtained and trained using various Algorithms. The most accurate model was chosen for diagnosing.

3.3.1 Logistic Regression

By Using Logistic Regression algorithm, we got an accuracy of 84% on training data and 80% on testing data as shown in fig 3.3.

```
Out[21]: LogisticRegression()

In [22]: X_train_prediction=model.predict(X_train)
         training_data_accuracy=accuracy_score(X_train_prediction,Y_train)

In [23]: print('accuracy on training data',training_data_accuracy)

         accuracy on training data 0.848780487804878

In [24]: X_test_prediction=model.predict(X_test)
         testing_data_accuracy=accuracy_score(X_test_prediction,Y_test)

In [25]: print('accuracy on testing data',testing_data_accuracy)

         accuracy on testing data 0.8048780487804879
```

Fig 3.3 Logistic Regression

3.3.2 K-Nearest Neighbor

By Using K-Nearest Neighbor algorithm, we got an accuracy of 90% on training data and 71% on testing data as shown in fig 3.4.

```
Out[26]: KNeighborsClassifier()

In [27]: X_train_prediction1=model1.predict(X_train)
         training_data_accuracy1=accuracy_score(X_train_prediction1,Y_train)

In [28]: print('accuracy on training data',training_data_accuracy1)

         accuracy on training data 0.9060975609756098

In [29]: X_test_prediction1=model1.predict(X_test)
         testing_data_accuracy1=accuracy_score(X_test_prediction1,Y_test)

In [30]: print('accuracy on testing data',testing_data_accuracy1)

         accuracy on testing data 0.7219512195121951
```

Fig 3.4 K-Nearest Neighbor

3.3.3 Support Vector Machine

By Using Support Vector Machine algorithm, we got an accuracy of 86% on training data and 82% on testing data as shown in fig 3.5.

```
In [32]: clf=SVC(kernel='linear')
         clf.fit(X_train,Y_train)

Out[32]: SVC(kernel='linear')

In [33]: X_train_prediction2=clf.predict(X_train)
         training_data_accuracy2=accuracy_score(X_train_prediction2,Y_train)

In [34]: print('accuracy on training data',training_data_accuracy2)

         accuracy on training data 0.8695121951219512

In [35]: X_test_prediction2=clf.predict(X_test)
         testing_data_accuracy2=accuracy_score(X_test_prediction2,Y_test)

In [36]: print('accuracy on testing data',testing_data_accuracy2)

         accuracy on testing data 0.824390243902439
```

Fig 3.5 Support Vector Machine

3.3.4 Naive Bayes

By Using Naive Bayes algorithm, we got an accuracy of 83% on training data and 78% on testing data as shown in fig 3.6.

```
In [37]: from sklearn.naive_bayes import GaussianNB
         model2 = GaussianNB()
         model2.fit(X_train, Y_train);

In [38]: X_train_prediction3=model2.predict(X_train)
         training_data_accuracy3=accuracy_score(X_train_prediction3,Y_train)

In [39]: print('accuracy on training data',training_data_accuracy3)

         accuracy on training data 0.8390243902439024

In [40]: X_test_prediction3=model2.predict(X_test)
         testing_data_accuracy3=accuracy_score(X_test_prediction3,Y_test)

In [41]: print('accuracy on testing data',testing_data_accuracy3)

         accuracy on testing data 0.7804878048780488
```

Fig 3.6 Naive Bayes

3.3.5 Random Forest

By Using Naive Bayes algorithm, we got an accuracy of 99% on training data and 96% on testing data as shown in fig 3.7.

```
In [61]: from sklearn.ensemble import RandomForestRegressor
         rf_random=RandomForestRegressor()

In [62]: from sklearn.model_selection import RandomizedSearchCV

In [63]: rf = RandomForestRegressor()

In [64]: rf_random.fit(X_train,Y_train)

Out[64]: RandomForestRegressor()

In [67]: predictions = rf_random.predict(X_test)
         predictions

In [68]: from sklearn.metrics import r2_score
         r2_score(Y_test,predictions)

Out[68]: 0.9700504761904762

In [69]: predictions = rf_random.predict(X_train)
         predictions
         r2_score(Y_train,predictions)

Out[69]: 0.9943217664112776
```

Fig 3.7 Random Forest

CHAPTER-4

HARDWARE

CHAPTER-4
HARDWARE

4.1 Methodology

In this project the first step is to collect ECG Data from human body using AD8232 Sensor, where three ECG electrodes are attached to the body. This data is sent to Arduino where we can see the output ECG signal in the serial monitor, which is further processed for Heart Beat per Minute and then displayed on OLED display module.

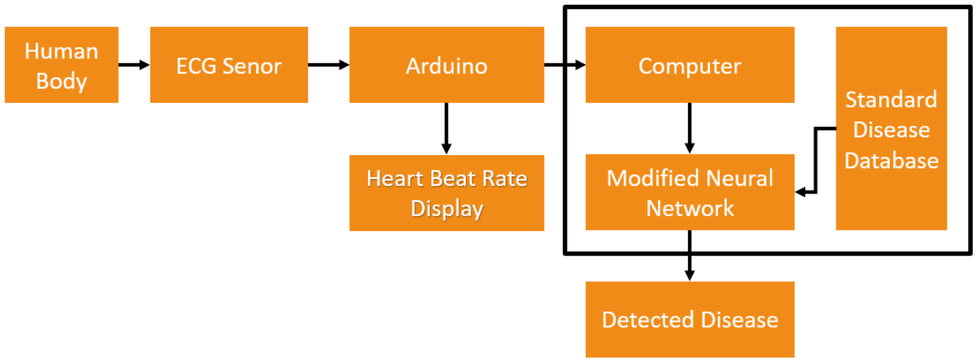


Fig 4.1 Block Diagram of the automatic cardiac disease detection

The signal received is then sent to the computer where the signal is further processed and is compared with the standard disease database, if there is any deviation from the standard database then computer displays that the patient has an abnormal Heart beat, if there is no deviation from the standard database then computer displays that the patient has a normal Heart beat.

4.2 Hardware Specification

The hardware setup for a real-time Arduino-based cardiac disease detection using a neural network project would typically involve several components in which the main hardware is the ECG sensor that will collect the ECG graph from the human body and the Central Processing Unit i.e., Arduino in this project. Here is the specification of the essential hardware components required for this project:

4.2.1 AD8232 Sensor

The AD8232 ECG (Electrocardiogram) sensor is a specialized component designed for monitoring the electrical activity of the heart. It is widely used in various applications, including medical devices, fitness trackers, and research projects. The AD8232 sensor provides a convenient and reliable means of acquiring ECG signals for analysis and monitoring.

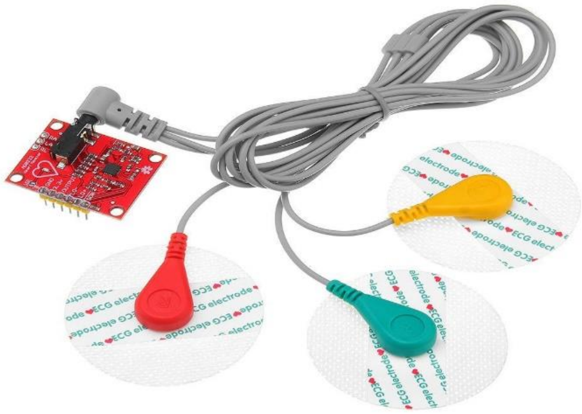


Fig 4.2 AD8232 ECG Sensor

The AD8232 ECG sensor is based on a single-lead architecture, which means it captures the electrical signals from one lead of the ECG. It features a three-electrode configuration consisting of two electrodes for signal acquisition and one electrode for the right leg drive (RLD) circuitry, which helps in reducing common-mode noise.

4.2.2 NodeMCU ESP8266

The NodeMCU ESP8266 is a popular and versatile development board that combines the power of the ESP8266 Wi-Fi module with an integrated microcontroller unit (MCU). It provides an easy and cost-effective way to connect devices to the internet and build IoT (Internet of Things) projects.

Fig 4.3 NodeMCU ESP8266

The ESP8266 module at the heart of the NodeMCU board is a Wi-Fi system-on-a-chip (SoC) that features a Tensilica Xtensa LX106 microcontroller unit. It provides built-in Wi-Fi connectivity, making it capable of connecting to wireless networks and accessing the internet. The module supports both 2.4 GHz Wi-Fi standards (802.11 b/g/n) and can function as a Wi-Fi access point or a client device.

4.2.3 OLED Display Module

A 1.3-inch OLED (Organic Light-Emitting Diode) display is a compact and versatile display module that offers high contrast, vibrant colors, and excellent viewing angles. The display typically features a resolution of 128x64 pixels or similar, providing a clear and sharp display for its compact size. The small size makes it suitable for devices with limited space requirements, such as smartwatches, fitness trackers, digital thermometers, and portable devices.

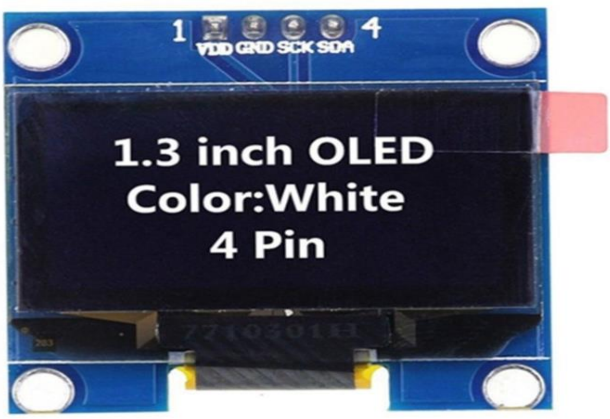


Fig 4.4 1.3-inch OLED Display Module

The OLED display module often comes with an integrated controller chip, such as the SSD1306 or SH1106, which simplifies the interface and allows for easy integration with microcontrollers or development boards. It usually supports common communication protocols like I2C or SPI, making it compatible with a wide range of microcontrollers, including Arduino and Raspberry Pi.

4.3 Hardware Implementation

We attached three lead pins of AD8232 ECG sensor to the human body, which is then connected to the Arduino Uno. The Arduino Uno Constantly collects data and displays it on the OLED display module. This data can also be seen in the serial plotter window of the Arduino IDE.

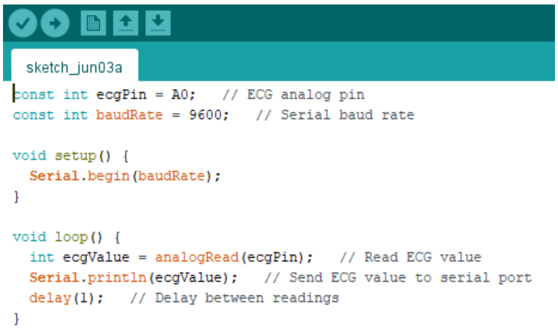


Fig 4.5 Hardware Setup

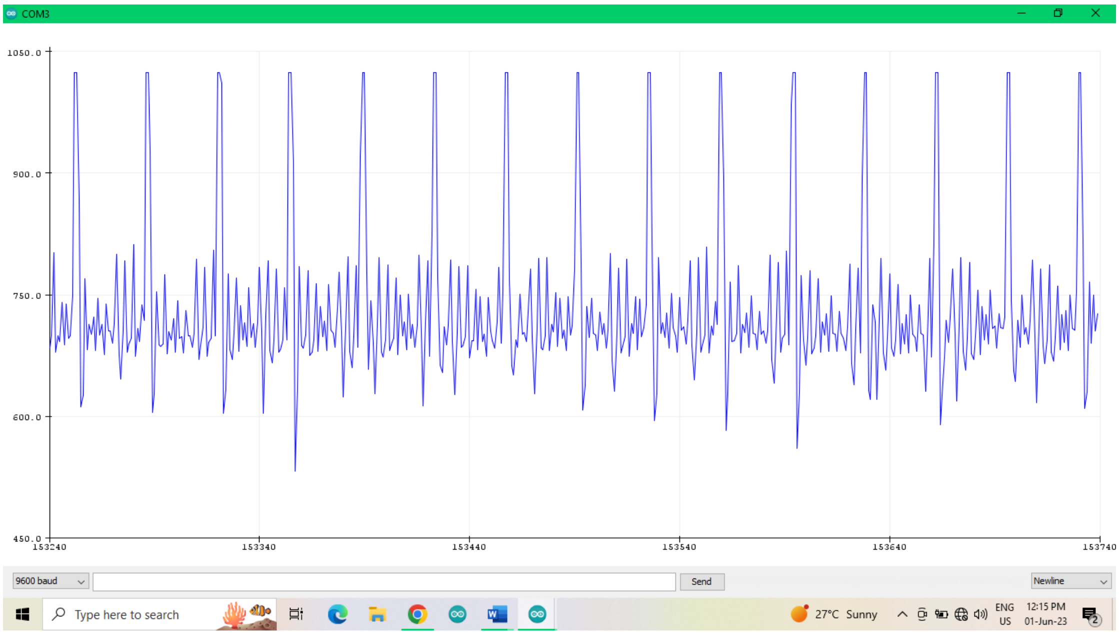


Fig 4.6 ECG Graph Data

CHAPTER-5

SOFTWARE

CHAPTER-5

SOFTWARE

5.1 Introduction to Visual Studio

Visual Studio is a powerful and versatile integrated development environment (IDE) developed by Microsoft. It serves as a comprehensive toolkit for software developers, providing a wide range of tools, features, and frameworks to streamline the development process. With Visual Studio, developers can create applications for various platforms, including desktop, web, mobile, and cloud. It supports multiple programming languages such as C#, Visual Basic, C++, F#, JavaScript, and more, enabling developers to choose the language that best suits their project requirements.

Fig 5.1 Visual Studio

Visual Studio offers a rich set of code editing and debugging capabilities, along with intuitive design tools and templates, making it easier to build visually appealing and user-friendly applications. It also integrates with popular frameworks, libraries, and platforms, such as .NET, Azure, Xamarin, Unity, and GitHub, facilitating seamless development, deployment, and collaboration. Overall, Visual Studio empowers developers to efficiently build high-quality software solutions with its comprehensive set of features and extensive ecosystem of tools and extensions.

5.2 Visual Studio Applications

Visual Studio has numerous applications and can be used for various types of software development projects. Here are some of the common applications of Visual Studio:

(a). **Desktop Application Development:** Visual Studio allows you to develop Windows desktop applications using languages like C#, Visual Basic, C++, and F#. It provides a rich set of libraries, controls, and design tools to build user-friendly and feature-rich desktop applications.

(b). **Web Application Development:** Visual Studio supports the development of web applications using popular frameworks and technologies like ASP.NET, ASP.NET Core, HTML, CSS, and JavaScript. It includes built-in templates, code editors, debugging tools, and deployment capabilities for efficient web development.

(c). **Mobile App Development:** Visual Studio offers tools for building cross-platform mobile apps for iOS, Android, and Windows using frameworks like Xamarin and React Native. You can write code in C# or JavaScript and leverage platform-specific APIs to create native-like mobile applications.

(d). **Cloud Development:** Visual Studio integrates with Azure, Microsoft's cloud computing platform, allowing you to develop, deploy, and manage cloud-based applications and services.

(e). **Game Development:** Visual Studio supports game development through integration with game engines like Unity and Unreal Engine. You can write code, debug, and deploy games for multiple platforms using Visual Studio's game development tools.

(f). **IoT (Internet of Things) Development:** Visual Studio offers tools for developing IoT applications, connecting devices, and processing data from sensors. You can use frameworks like Windows IoT Core, Arduino, and Raspberry Pi to build IoT solutions.

(g). **AI and Machine Learning:** Visual Studio provides tools and frameworks for developing artificial intelligence (AI) and machine learning (ML) applications. It supports frameworks like TensorFlow and PyTorch and includes features for model training, deployment, and inference.

(h). **Team Collaboration:** Visual Studio includes features for version control, code collaboration, and project management. It supports popular version control systems like Git and provides integration with Azure DevOps for managing software development workflows.

5.3 Software Implementation

The single use of Visual Studio in the project of Real-Time Arduino Based Cardiac Disease Detection Using Neural Network is to serve as a development environment for coding and managing the project. Visual Studio provides a powerful integrated development environment (IDE) that supports multiple programming languages, including C++ and C#. It offers features such as code editing, debugging tools, and project management, allowing developers to efficiently write, test, and organize the codebase.

5.3.1 Visual Studio Interface

The Visual Studio interface for ECG detection offers an intuitive setting for creating and putting into practice algorithms. It provides project management, debugging, and code editing capabilities to facilitate effective development and testing. Machine learning, feature extraction, and signal processing are made easier by integration with libraries and frameworks.

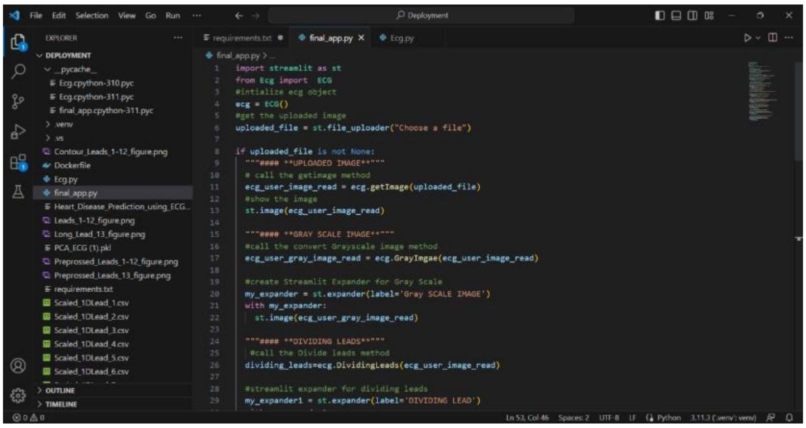


Fig 5.2 Importing Libraries for Streamlit Application

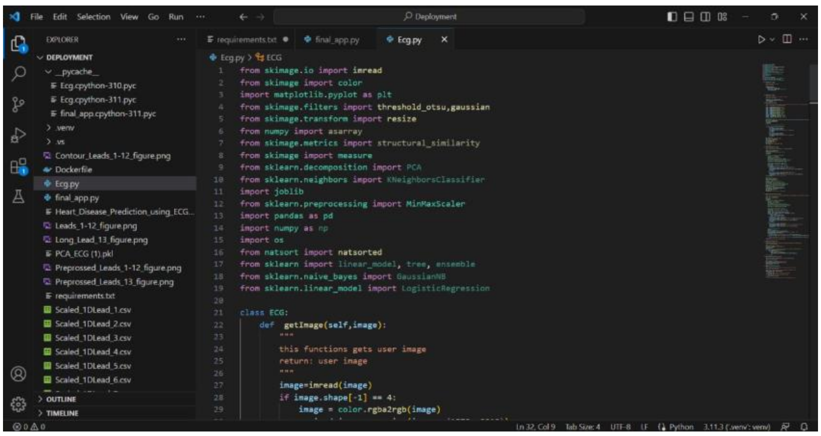


Fig 5.3 Importing Libraries for ECG Data Extraction

5.3.2 Data Preparation and Preprocessing

1. Convert all ECG images to grayscale
2. Resize image per requirements
3. Divide the images into 12 sections for each image extracting 12 lead values (1-12)
4. Remove Grid lines from each lead image and convert to binary image

Fig 5.4 Uploading the ECG Data

Fig 5.5 Converting ECG Data into Gray Scale Image

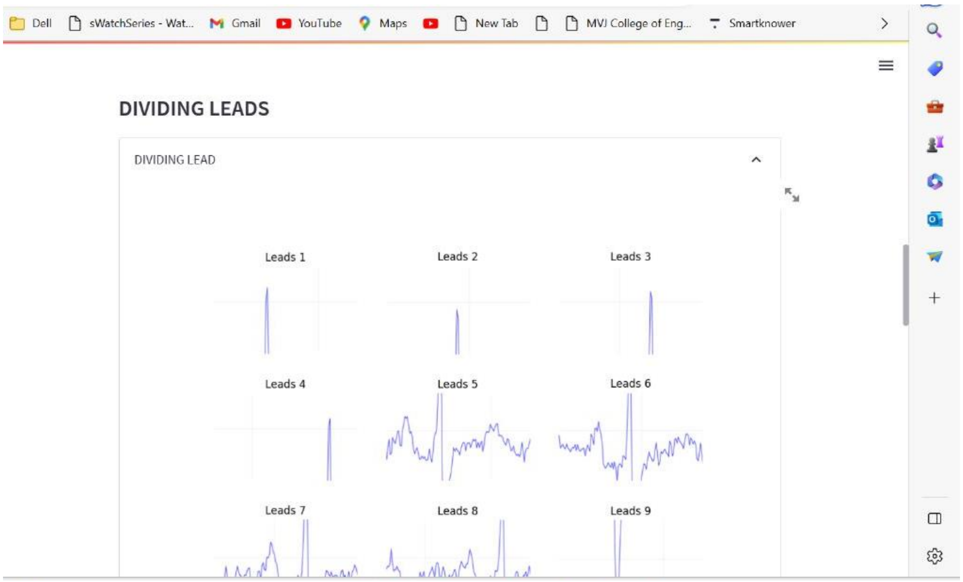


Fig 5.6 Dividing the Leads

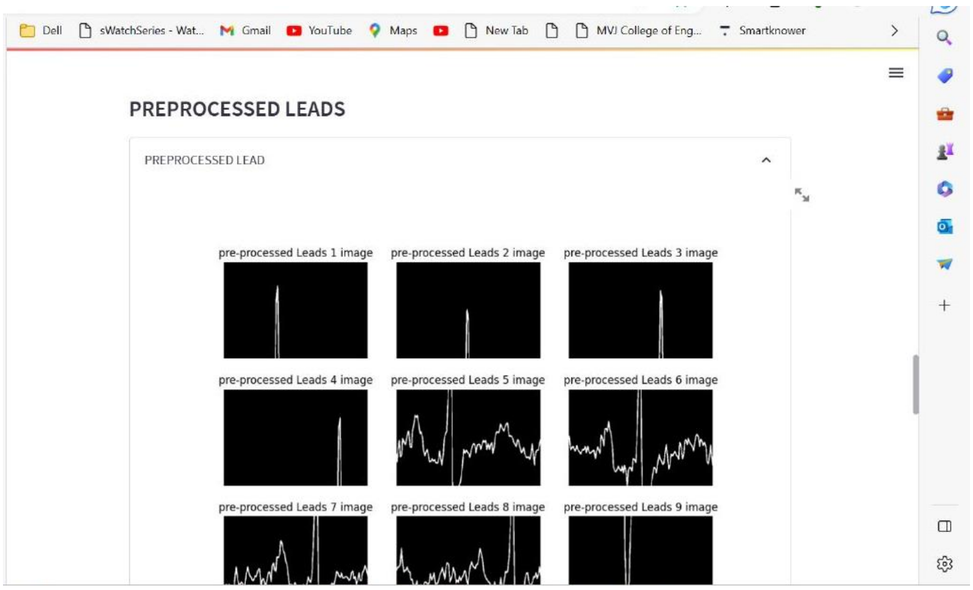


Fig 5.7 Preprocessed Leads

5.3.3 Data feature Extraction & Data Engineering

1. To trace and extract only the necessary signal from images, we used the contour technique
2. The image is converted to a one-dimensional signal
3. After that MinMaxScaler is used to scale the image
4. For each lead1-12) signal in all ecg image, save the 10 signal values in a.csv.file
5. In a single csv file, combine all 12 lead values with the target label added.

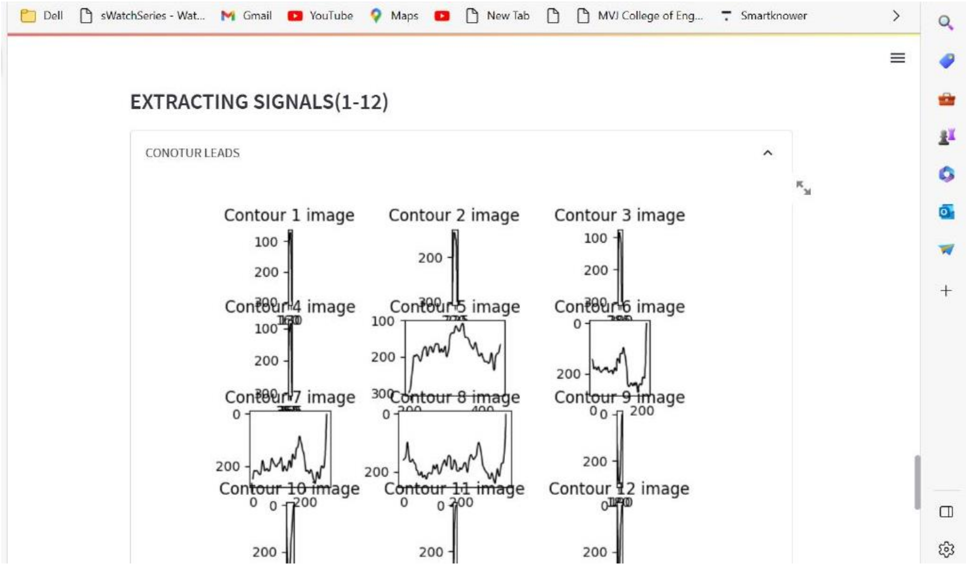
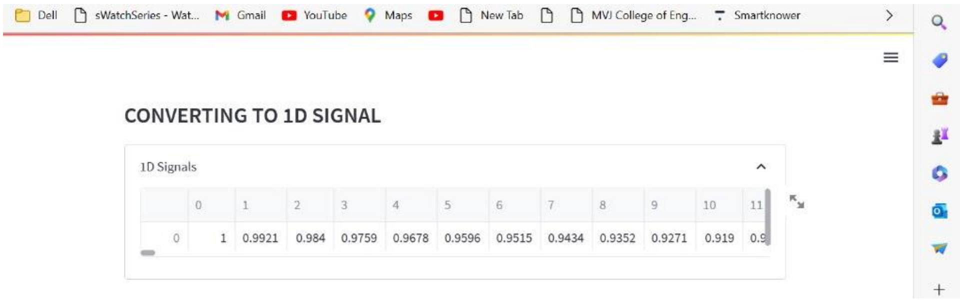
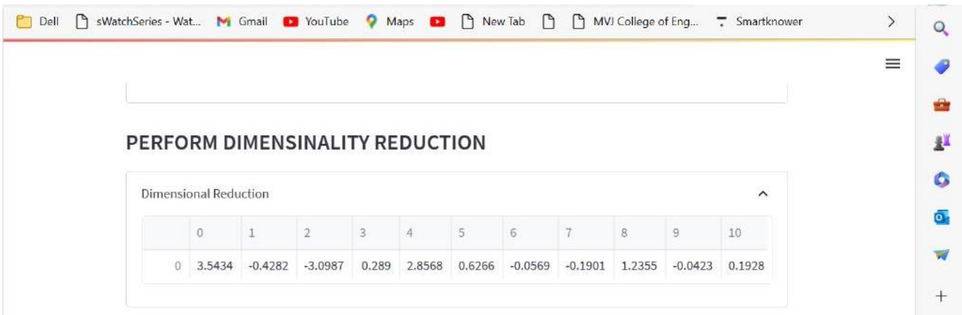


Fig 5.8 Extracting Signals



	0	1	2	3	4	5	6	7	8	9	10	11
0	1	0.9921	0.984	0.9759	0.9678	0.9596	0.9515	0.9434	0.9352	0.9271	0.919	0.9

Fig 5.9 Converting to 1D Signal



	0	1	2	3	4	5	6	7	8	9	10
0	3.5434	-0.4282	-3.0987	0.289	2.8568	0.6266	-0.0569	-0.1901	1.2355	-0.0423	0.1928

Fig 5.10 Performing Dimensionality Reduction

CHAPTER-6

RESULT AND ANALYSIS

CHAPTER-6

RESULT AND ANALYSIS

6.1 Result

The creation of a software programme that makes use of algorithms and machine learning strategies to analyse real-time data from cardiac sensors or devices is required for real-time cardiac disease detection using Visual Studio Code. The popular integrated programming environment (IDE) Visual Studio Code, which offers a strong foundation for coding, debugging, and testing, is used to create the application. To identify patterns or abnormalities suggestive of heart disorders, the software collects and interprets data from the cardiac sensors, extracts pertinent information, and applies machine learning models. The outcomes of this real-time detection system can yield vital data for early diagnosis and action, thereby enhancing patient outcomes and lowering the likelihood of problems. In order to assess the success of the system and improve the algorithms used for detection, an analysis of the system's performance, including accuracy, sensitivity, and specificity, is necessary. This will ensure dependable and accurate cardiac disease detection in real-time settings.

6.2 Analysis

On analyzing the results of all the algorithms using the accuracy showed by each algorithm for both training and testing data.

Table 6.1 Comparing the result of all the Algorithms

Sl no.	Models	Training Data Accuracy	Testing Data Accuracy
1.	Logistic Regression	84.8%	80.48%
2.	K-Neighbor Classifier	90%	72.1%
3.	SVM	86.9%	82.4%
4.	Naive Bayes	83.9%	78%
5.	Random Forest	99.4%	96.6%

As seen in Table 6.1, Compared to other algorithms, Random Forest has the best accuracy in both training and test data results. Therefore, for this particular project Random Forest is the most suitable algorithm when compared to other algorithms.

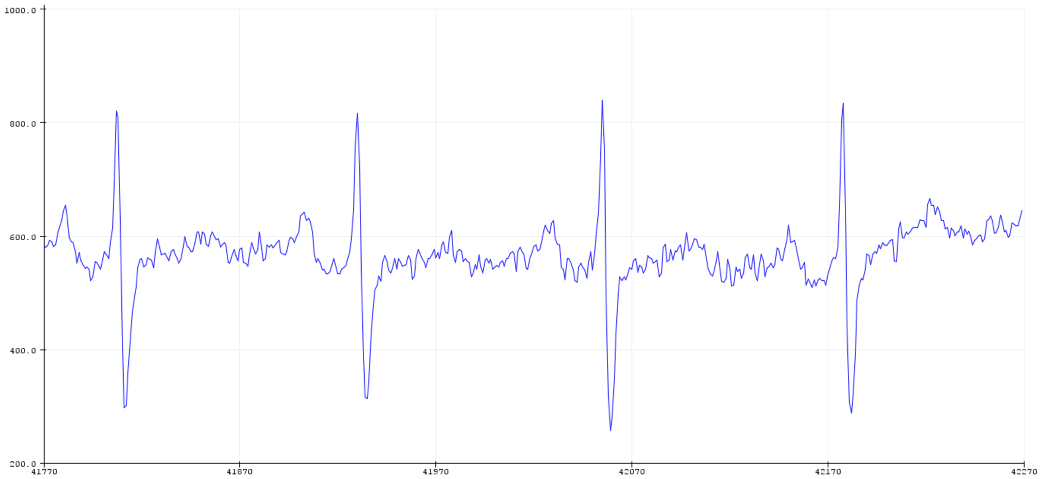


Fig 6.2 The ECG Signal

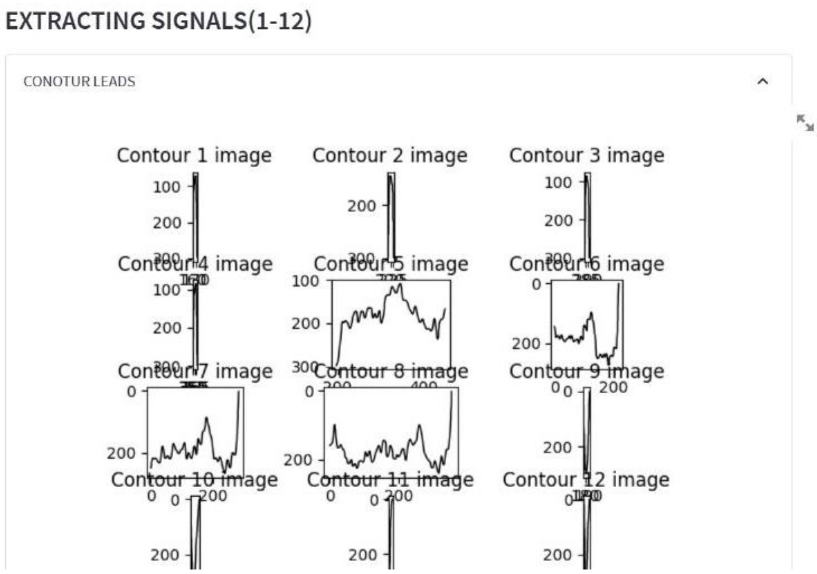


Fig 6.3 Extracting Signals

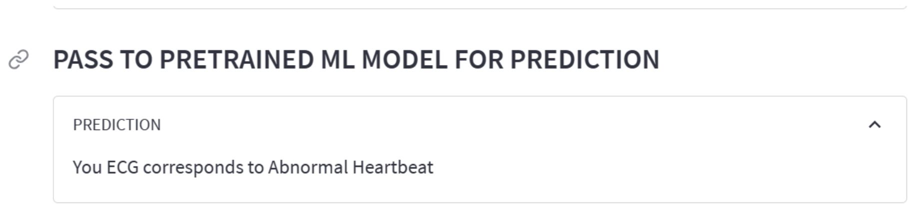


Fig 6.4 Predicted Output

CHAPTER-7

CONCLUSION AND FUTURE SCOPE

CHAPTER 7
CONCLUSION

7.1 Conclusion

Heart disease is a major issue for the general public because it affects such an important organ of the body. Consequently, using machine learning algorithms on information the early detection of cardiovascular disease will help prevent this dangerous condition and save many lives. Additionally, early detection of any heart condition abnormalities can aid researchers in conducting studies more efficiently, which will ultimately benefit healthcare. In this project, we applied a variety of machine learning algorithms to predict the development of heart disease. In order to find the model with the highest accuracy for spotting cardiac problems, the Kaggle dataset was obtained and trained using various methods. The most accurate model was chosen for diagnosing cardiac problems in real-time patient data after each model's accuracy was assessed. It is possible to determine if a patient has a cardiac issue or not by applying the real-time data from casualties to the highest accuracy model. This procedure can assist medical personnel in providing patients with heart diseases with more thorough, quicker diagnosis as well as higher-quality care overall. Overall, applying machine learning models to the healthcare industry can yield insightful information and enhance patient outcomes.

7.2 Future Scope

The future scope of the Real Time Arduino Based Cardiac Disease Detection Using Neural Network project is promising. One potential avenue for expansion is the integration of more advanced machine learning techniques and algorithms to enhance the accuracy and reliability of cardiac disease detection. Additionally, the project can benefit from the incorporation of wearable devices with real-time ECG monitoring capabilities, enabling continuous monitoring and immediate detection of cardiac abnormalities. Another exciting direction is the integration of data from multiple sensors and health monitoring devices to provide a comprehensive assessment of an individual's cardiovascular health. Overall, the future scope of the project lies in refining the detection algorithms, incorporating real-time monitoring capabilities, embracing predictive analytics, and forging partnerships to make significant contributions to the field of cardiac disease detection and prevention.

CERTIFICATE

MVJ COLLEGE OF ENGINEERING Since 1982

Engineering A Better Tomorrow

An Autonomous Institute

Approved by AICTE, New Delhi
Affiliated to Visvesvaraya Technological University, Belagavi
Recognized by UGC with 2(f) and 12(B) status
Accredited by NBA and NAAC

Certificate of Participation

This is to certify that,

M Santosh

has presented a paper titled,

" Real Time ECG Monitoring System Using Arduino "

in Sixth National Conference on "Design innovation for 3-C's Complite, Communicate, Control" (NCDI3C-2023) organized by the Department of ECE, EEE & IIOT on 26th and 27th May 2023, held at MVJ College of Engineering, Bengaluru.

Principal

REFERENCES

[1] Shah, K., Rane, M. and Emamian, D.V. "Detection of Heart Defects using Electrocardiogram (ECG)". International Journal of Science and Engineering Invention, (Volume 6 – Issue 13), pp 15 – 19, 2018

[2] Serhani MA, T. El Kassabi H, Ismail H, Nujum Navaz A. "ECG Monitoring Systems: Review, Architecture, Processes, and Key Challenges." Sensors, pp 7 – 11, 2020

[3] Ray, S.A Quick Review of Machine Learning Algorithms. 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon) (Volume 6 – Issue 13), pp 1 – 3, 2019

[4] Ahmed Aboalseoud; Akram Youssry; Mohamed El-Nozahi; Ahmed El-Rafei; Ahmed ElBealy; Ayman Wahba, Hani Ragaai , "Wireless ECG Monitoring System for Telemedicine Application", Ninth International Conference on Intelligent Computing and Information Systems (ICICIS), pp 5 – 12, 2019

[5] Kristine Joyce P. Ortiz, John Peter O. Davalos, Elora S. Eusebio, Dominic M. Tucay "IoT: Electrocardiogram (ECG) Monitoring System", Indonesian Journal of Electrical Engineering and Computer Science Vol. 10, No. 2, pp. 480 – 489, 2018

[6] Deepak Yadav, KartikKaul, Aman Sharma, Janakkumar B.Patel, Manoj Pandey "Real Time Ecg Monitoring System Using Arduino And Labview" Amity University Haryana, India. JETIR May 2018, Volume 5, Issue 5. Pp 446 – 450, 2018

[7] Rumpa, Lantana & Suluh, Sallolo & Ramopoly, Irene & Jefriyanto, Wilson. Development of ECG sensor using arduino uno and e-health sensor platform: mood detection from heartbeat. Journal of Physics: Conference Series. 1528. 012043. 10.1088/1742-6596/1528/1/012043. pp 1- 6, 2020

[8] Mr. Ved Prakasha, Mr. Manoj Kumar Pandey "Heart Rate Monitoring System", ECE Department, Amity University Gurgaon, Haryana, JETIR May 2018, Volume 5, Issue 5 pg 879 – 886, 2018

[9] K.Lakshmi Bhanu Prakash Reddy, M.Komalaakshi, T.Bhargavi3 "Arduino Based Ecg Monitoring System", National Conference on Emerging Trends in Information, Management and Engineering Sciences , ISSN: 2395-1303, pp 1 – 3, 2018