CLASSIFICATION OF ARRHYTHMIA BY USING DEEP LEARNING WITH 2-D ECG SPECTRAL IMAGE REPRESENTATION

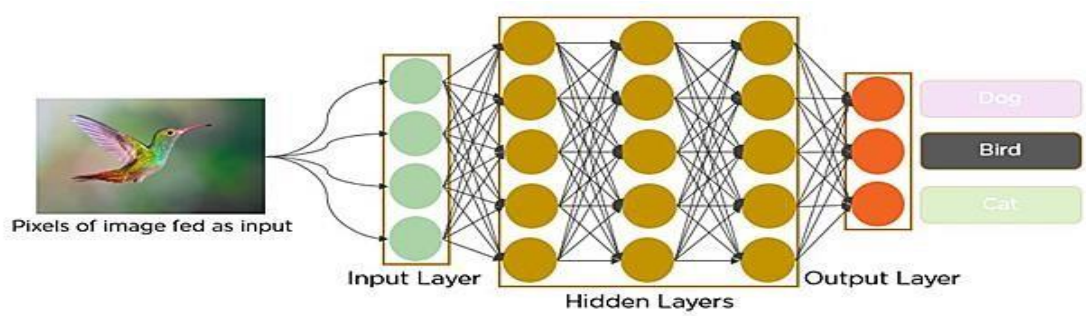
INTRODUCTION:

1.1 OVERVIEW:

According to the World Health Organization (WHO), cardiovascular diseases (CVDs) are the number one cause of death today. Over 17.7 million people died from CVDs in the year 2017 all over the world which is about 31% of all deaths, and over 75% of these deaths occur in low and middle-income countries. Arrhythmia is a representative type of CVD that refers to any irregular change from the normal heart rhythms. There are several types of arrhythmia including atrial fibrillation, premature contraction, ventricular fibrillation, and tachycardia. Although a single arrhythmia heartbeat may not have a serious impact on life, continuous arrhythmia beats can result in fatal circumstances. In this project, we build an effective electrocardiogram (ECG) arrhythmia classification method using a convolution al neural network (CNN), in which we classify ECG into seven categories, one being normal and the other six being different types of arrhythmia using deep two-dimensional CNN with grayscale ECG images. We are creating a web application where the user selects the image which is to be classified. The image is fed into the model that is trained and the cited class will be displayed on the webpage.

1.2 PURPOSE:

In the past few decades, Deep Learning has proved to be a compelling tool because of its ability to handle large amounts of data. The interest to use hidden layers has surpassed traditional techniques, especially in pattern recognition. One of the most popular deep neural networks is Convolution al Neural Networks.

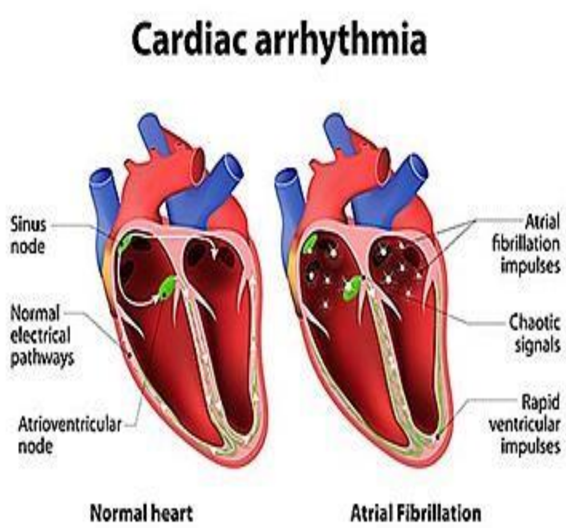


In deep learning, a convolution al neural network (CNN/ConvNet) is a class of deep neural networks, most commonly applied to analyze visual imagery. Now when we think of a neural network we think about matrix multiplications but that is not the case with ConvNet. It uses a special technique called Convolution. Now in mathematics convolution is a mathematical operation on two functions that produces a third function that expresses how the shape of one is modified by the other.

LITERATURE SURVEY:

2.1 EXISTING PROBLEM:

Cardiovascular diseases (CVDs) are the number one cause of death today. Over 17.7 million people died from CVDs in the year 2017 all over the world which is about 31% of all deaths, and over 75% of these deaths occur in low and middle-income countries. Arrhythmia is a representative type of CVD that refers to any irregular change from the normal heart rhythms. There are several types of arrhythmia including atrial fibrillation, premature contraction, ventricular fibrillation, and tachycardia.



2.2 PROPOSED SOLUTION:

An "ambulatory electrocardiogram" or an ECG) about the size of a postcard or digital camera that the patient will be using for 1 to 2 days, or up to 2 weeks. The test measures the movement of electrical signals or waves through the heart. These signals tell the heart to contract (squeeze) and pump blood. The patient will have electrodes taped to your skin. It's painless, although some people have mild skin irritation from the tape used to attach the electrodes to the chest.They can do everything but shower or bathe while wearing the electrodes. After the test period, patient will go back to see your doctor. They will be downloading the information.

2.3 PROBLEM STATEMENT DEFINITION:

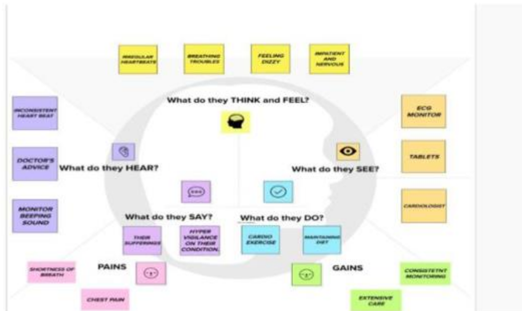
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	A Lab Technician	Find a software to classify Arrhythmia beats based on ECG Output.	Accuracy in pattern detection is not too high.	Convolutional neural networks has some limitations.	Challenging.
PS-2	A cardiac patient	Find an ECG device to identify arrhythmia beats.	The device is not classifying the beats properly.	It takes lot of time to identify .	Scary.

IDEATION & PROPOSED SOLUTION:

3.1 Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

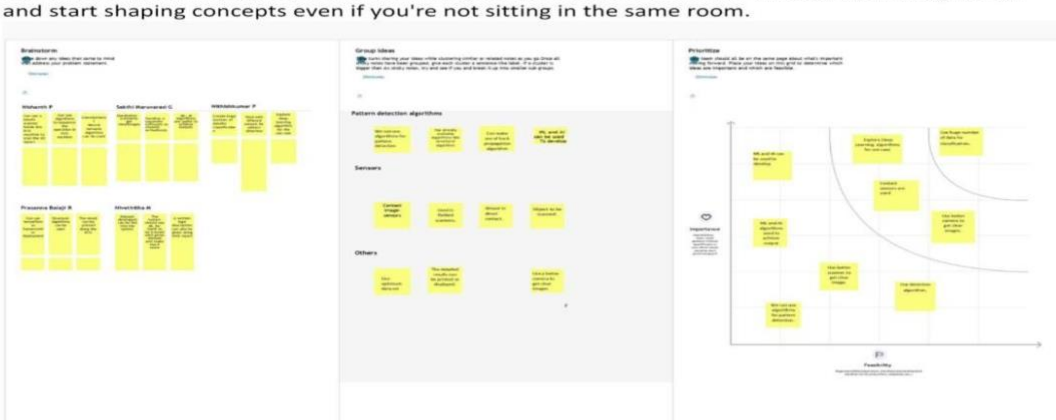
It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.



3.2 Ideation & Brainstorming:

Brainstorm & Idea Prioritization Template:

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.



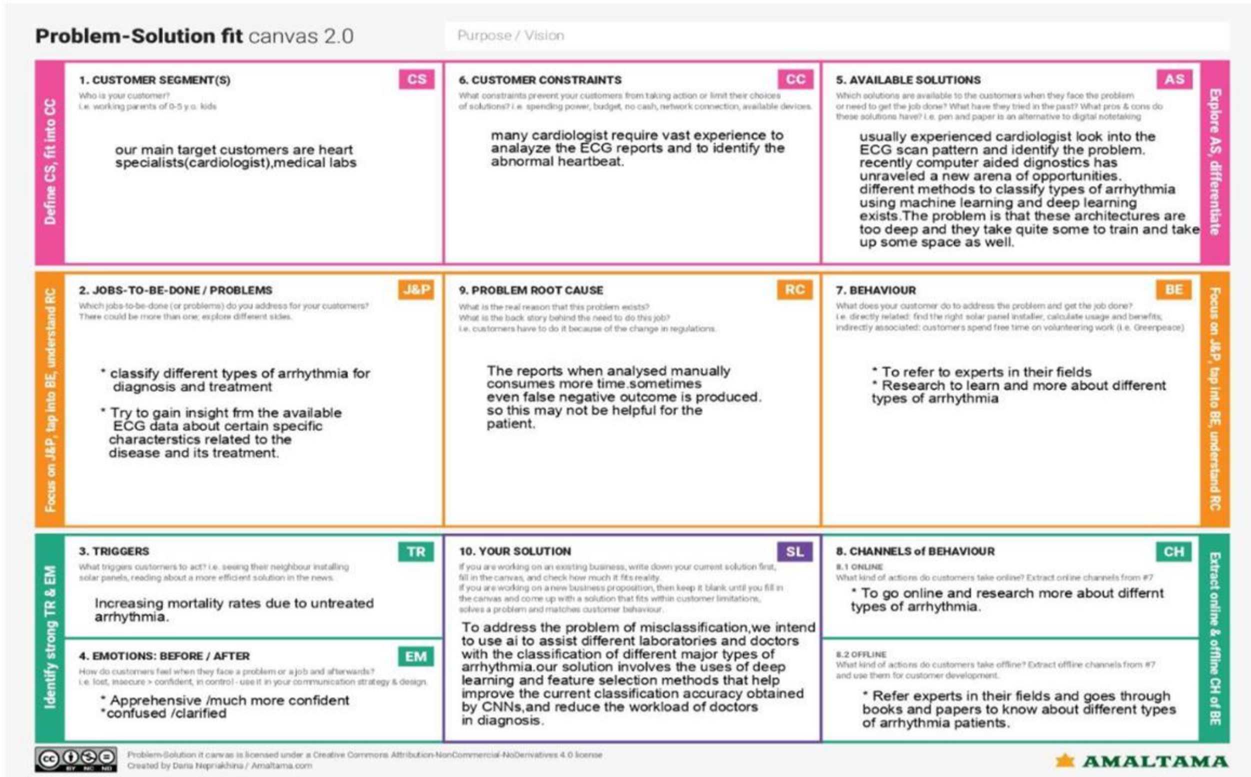
3.3 Proposed Solution:

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	ECG signals is crucial for precise diagnoses of patients' acute and chronic heart conditions. In this study, we propose a two-dimensional (2-D) convolutional neural network (CNN) model for the classification of ECG signals into eight classes; namely, normal beat, premature ventricular contraction beat, paced beat, right bundle branch block beat, left bundle branch block beat, atrial premature contraction beat, ventricular flutter wave beat, and ventricular escape beat. The one-dimensional ECG time series signals are transformed into 2-D spectrograms through short-time Fourier transform. The 2-D CNN model consisting of four convolutional layers and four pooling layers is designed for extracting robust features from the input spectrograms. Our proposed methodology is evaluated on a publicly available MIT-BIH arrhythmia dataset. We achieved a state-of-the-art average classification accuracy of 99.11\%, which is better than those of recently reported results in classifying similar types of arrhythmias. The performance is significant in other indices as well, including sensitivity and specificity, which indicates the success of the proposed method.
2.	Idea / Solution description	we propose a two-dimensional (2-D) convolutional neural network (CNN) model for the classification of ECG signals into eight classes; namely, normal beat, premature ventricular contraction beat, paced beat, right bundle branch block beat, left bundle branch block beat, atrial premature contraction beat, ventricular flutter wave beat, and ventricular escape beat. The one-dimensional ECG time series signals are transformed into 2-D spectrograms through short-time Fourier transform. The 2-D CNN model consisting of four convolutional layers and four pooling layers is designed for extracting robust features from the input spectrograms. Our proposed methodology is evaluated on a publicly available MIT-BIH arrhythmia dataset. We achieved a state-of-the-art average classification accuracy of 99.11%, which is better than those of recently reported results in classifying similar types of arrhythmias. The performance is significant in other indices as well, including sensitivity and specificity, which indicates the success of the proposed method.
3.	Novelty / Uniqueness	We achieved a state-of-the-art average classification accuracy of 99.11%, which is better than those of recently reported results in classifying similar types of arrhythmias. The performance is significant in other indices as well, including sensitivity and specificity, which indicates the success of the proposed method.
4.	Social Impact / Customer Satisfaction	1. Upgradeable Software 2. Works well with Unstructured Data 3. Better Self-Learning Capabilities 4. Supports Parallel and Distributed Algorithms. 5. Cost Effectiveness 6. Low cost maintenance
5.	Business Model (Revenue Model)	Drivers of medical device growth -India •Higher disposable incomes •Increase in public spend on healthcare •Increase in penetration of health insurance •Models of healthcare emerging •Many avenues for funding
6.	Scalability of the Solution	In this study, we proposed a 2-D CNN-based classification model for automatic classification of cardiac arrhythmias using ECG signals. An accurate taxonomy of ECG signals is extremely helpful in the prevention and diagnosis of CVDs. Deep CNN has proven useful in enhancing the accuracy of diagnosis algorithms in the fusion of medicine and modern machine learning technologies. Using 2-D images, can classify eight kinds of arrhythmia, namely, NOR, VFW, PVC, VEB, RBB, LBB, PAB, and APC, and it achieved 97.91% average sensitivity, 99.61% specificity, 99.11% average accuracy, and 98.59% positive predictive value (precision). These results indicate that the prediction and classification of arrhythmia with 2-D ECG representation as spectrograms and the CNN model is a reliable operative technique in the diagnosis of CVDs. The proposed scheme can help experts diagnose CVDs by referring to the automated classification of ECG signals. The present research uses only a single-lead ECG signal. The effect of multiple lead ECG data to further improve experimental cases will be studied in future work

3.4 Problem Solution fit:



REQUIREMENT ANALYSIS:-

4.1 Functional requirement:

Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through gmail Registration through mobile number.
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP.

4.2 Non-Functional requirements:

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution.

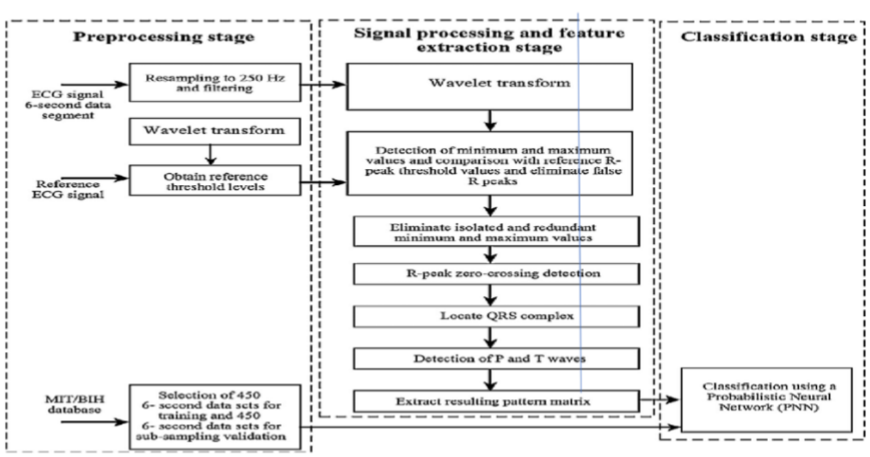
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Our main target are heart specialists(cardiologist), medical labs. Our software is used by our customers in an easy manner.
NFR-2	**Security**	The ECG images are encrypted. Our Application is secured using various layers of firewall. Only the Authorized person can able to access the images.
NFR-3	**Reliability**	We achieved a state-of-the-art average classification accuracy of 99.11%, which is better than those of recently reported results in classifying similar types of arrhythmias.
NFR-4	**Performance**	The performance is significant in other indices as well, including sensitivity and specificity, which indicates the success of the proposed method.
NFR-5	**Availability**	Our Software is available 24*7 for registered authentic users. The images are only available to the authorized Medical Specialists.
NFR-6	**Scalability**	We proposed a 2-D CNN-based classification model for automatic classification of cardiac arrhythmias using ECG signals. An accurate taxonomy of ECG signals is extremely helpful in the prevention and diagnosis of CVDs. Deep CNN has proven useful in enhancing the accuracy of diagnosis algorithms in the fusion of medicine and modern machine learning technologies. Using 2-D images, can classify eight kinds of arrhythmia,

PROJECT DESIGN

5.1 Data Flow Diagrams:

Data Flow Diagrams:

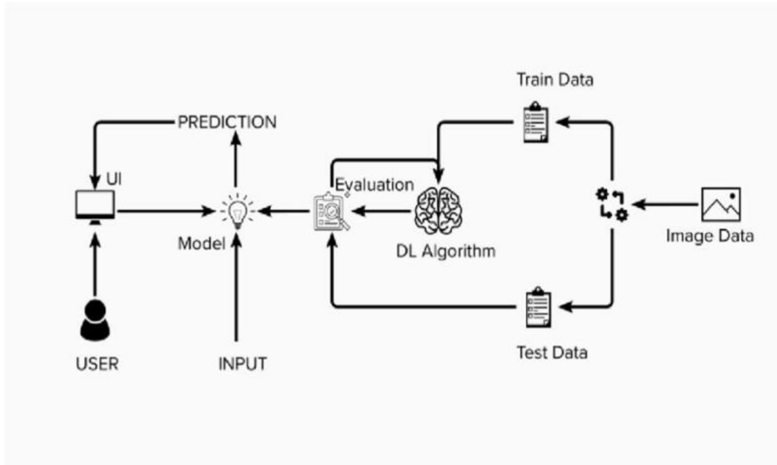
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Solution & Technical Architecture:

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



5.3 User Stories:

User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Individual patient	To know or verifythe arrhythmia condition	USN-1	As a user, I can log in to the website using user name and password(credentials) or create one if am new, can proceed with filling out the details as individual and uploading the scanned copy of ECG report	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive the report throughemail or download it on the site itself	I can receive output inpdf format	Low	Sprint-1
		USN-3	As a user, I can see my previous reports of arrhythmia (stages) on my account	I can access it if its a authorised log in	High	Sprint-2
	Login	USN-4	As a user, I can log into the application by entering email & password		High	Sprint-1
Lab Technicians(Hospitals)	To print summaryalong with ECG ifthe patient is observed with arrhythmia	USN-5	As a user I need an application to run along with my ECG machine to classify the type of arrhythmia and provide a report with results	It can run along with itsoperation every time	High	Sprint-3

PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation:

S.NO	MILESTONE	ACTIVITIES	DATE
1.	Preparation Phase	Pre-requisites	3 Nov 2022
		Prior knowledge	3 Nov 2022
		Project Structure	3 Nov 2022
		Project Flow	5 Nov2022
		Project Objectives	3Nov 2022
		Registrations	
		Environment Set-up	
2.	Ideation Phase	Literature Survey	7 Oct 2022
		Empathy Map	7 Oct 2022
		Problem Statement	5 Oct 2022
		Ideation	7 Oct 2022
3.	Project Design Phase -I	Proposed Solution	13 Oct 2022
		Problem Solution Fit	13 Oct 2022
		Solution Architecture	13 Oct 2022
4.	Project Design Phase -II	Customer Journey	14 Oct 2022
		Requirement Analysis	14 Oct 2022
		Data Flow Diagrams	19 oct 2022
		Technology Architecture	19 Oct 2022
5.	Project Planning Phase	Milestones & Tasks	5 Nov 2022
		Sprint Schedules	5 Nov 2022
6.	Project Development Phase	Sprint-1	
		Sprint-2	
		Sprint-3	
		Sprint-4	

6.2 Sprint Delivery Schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	HomePage	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Prasanna Balaji R
Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Nithishkumar P
Sprint-2	Registrarion	USN-3	As a user, I can register for the application through Facebook	2	Low	Nivethitha M
Sprint-3	Uploading Image	USN-4	As a user, I could upload ECG Image	2	Medium	Sakthi Maruvarasi G
Sprint-4	Result	USN-5	The convolutional neural network identifies and classify arrythmia and gives out the result	1	High	Nishanth P

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	04 Nov 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	08 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	13 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	16 Nov 2022

CODING & SOLUTIONING

7.1 Feature 1:

7.1.1 APP.PY:-

```
impo
rt os
    import numpy as np #used for numerical analysis

    from flask import Flask,request,render_template

      # Flask-It is our framework which we are going to use to run/serve our application.
      #request-for accessing file which was uploaded by the user on our application.

      #render_template- used for rendering the html pages
```

```
 from tensorflow.keras.models import load_model#to load our trained model   from
tensorflow.keras.preprocessing import image

        app=Flask(__name__)#our flask app
        model=load_model('ECG.h5')#loading the model

 @app.route("/") #default route   def
about():
            return render_template("home.html")#rendering html page

        @app.route("/home") #default route
        def home():
            return render_template("home.html")#rendering html page

        @app.route("/info") #default route
        def information():
            return render_template("info.html")#rendering html page

        @app.route("/upload") #default route
        def test():
            return render_template("predict.html")#rendering html page

        @app.route("/predict",methods=["POST","GET"]) #route for our prediction
        def upload():

            if request.method=='POST':

              f=request.files['file'] #requesting the file
              basepath=os.path.dirname('__file__')#storing the file directory
              filepath=os.path.join(basepath,"uploads",f.filename)#storing the file in uploads folder
             f.save(filepath)#saving the file
print("file save")

             img=image.load_img(filepath,target_size=(64,64)) #load and reshaping the image
x=image.img_to_array(img)#converting image to array           x=np.expand_dims(x,axis=0)#changing the
dimensions of the image
```

```
##      pred=model.predict(x)#predicting classes
##      y_pred = np.argmax(pred)
##      print("prediction",y_pred)#printing the prediction      preds=model.predict(x)#predicting classes
pred=np.argmax(preds,axis=1)#predicting classes      print("prediction",pred)#printing the prediction


index=['Left Bundle Branch Block','Normal','Premature Atrial Contraction','Premature Ventricular Contractions', 'Right Bundle Branch Block','Ventricular Fibrillation']
##      result=str(index[y_pred])
    result=str(index[pred[0]])


return result#resturing the result      return None


#port = int(os.getenv("PORT"))
if __name__=="__main__":


    app.run(debug=False)#running our app
  #app.run(host='0.0.0.0', port=8000)
```

7.2 Feature 2:

7.2.1 HOME.HTML:

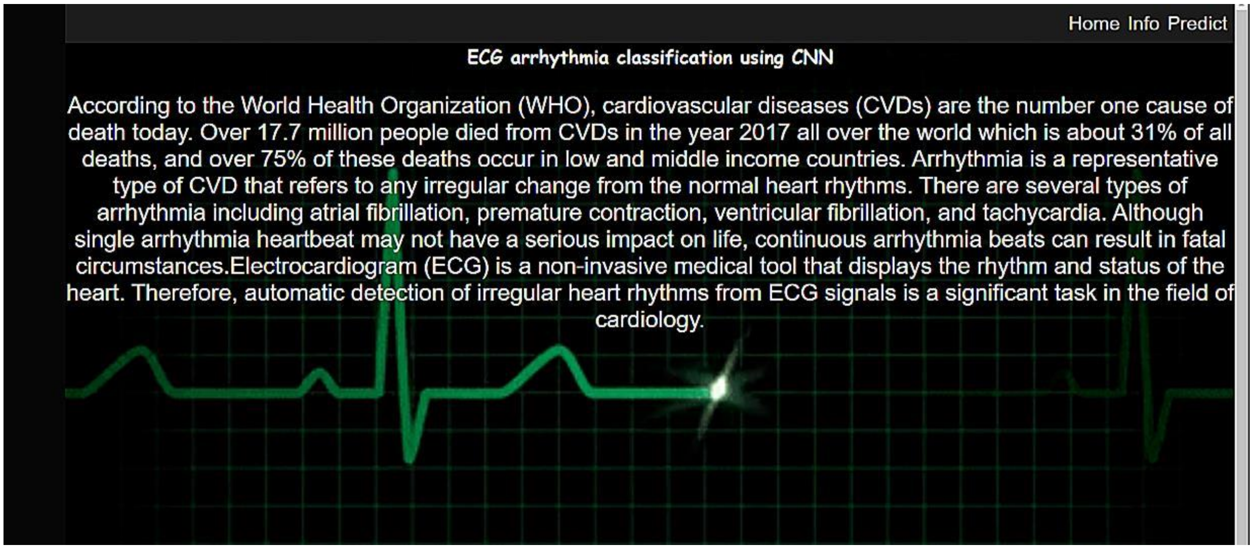
```
<!DOCTY
PE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">


    <title>Home</title>

<style>

body {
```

```
            margin:                         0px;
padding: 0px;             font-family:
sans-serif;
              }



              /* .pd {
                 padding-bottom: 100%;
              } */

              .navbar {
                 padding: 20px 0px 40px;
                   ;
background-color: #222;


                   font-size: 25px;
                   text-align: center;


              }


                   .navbar a {
color: #eee;                  float:
right;             text-decoration:
none;              font-style: normal;
                   font-family: sans-
serif;             padding-right:
10px;
              }


                   .navbar a:hover {
background-color: rgb(0, 0, 0);
color: rgb(17, 194, 238);
border-radius: 5px;
padding: 5px;
                   }              .content{                     background-image:
url("https://thumbs.gfycat.com/ChiefHeftyBasil-small.gif");                 background-size:
cover;                            background-repeat: no-repeat;                         height: 87vh;
                      margin-top: -21px;
```

```
            }

                .dic p {
color: white;          text-align:
center;                        font-
family: sans-serif;
font-size: 30px;
            }

            footer{          display: flex;
      justify-content: center;
      background-color: #222;
              margin-top: -10px;
              color: white;



            }
        </style>
      </head>

      <body>
        <div class="navbar">
            <a href="/upload">Predict</a>
            <a href="/info">Info</a>
            <a href="/home">Home</a>

        </div>

        <div  class="content">

         <h2  style="display: flex;justify-content: center;  color:white;size:15;font-family:comic Sans MS">ECG
     arrhythmia classification using CNN</h2>




            <div class="dic">
```

```
<p>According to the World Health Organization (WHO), cardiovascular diseases (CVDs) are the
number one cause of death today. Over 17.7 million people died from CVDs in the year 2017 all
over the world which is about 31% of all deaths, and over 75% of these deaths occur in low and
middle income countries. Arrhythmia is a representative type of CVD that refers to any irregular
change from the normal heart rhythms. There are several types of arrhythmia including atrial
fibrillation, premature contraction, ventricular fibrillation, and tachycardia. Although single
arrhythmia heartbeat may not have a serious impact on life, continuous arrhythmia beats can result in
fatal circumstances.Electrocardiogram (ECG) is a non-invasive medical tool that displays the rhythm
and status of the heart. Therefore, automatic detection of irregular heart rhythms from ECG signals
is a significant task in the field of cardiology.
        </p>
      </div>
    </div>
    <footer>
      <h4>@All Rights Reserved</h4>
    </footer>
  </body>


  </html>
```

OUTPUT:



7.2.2 INFO.HTML:

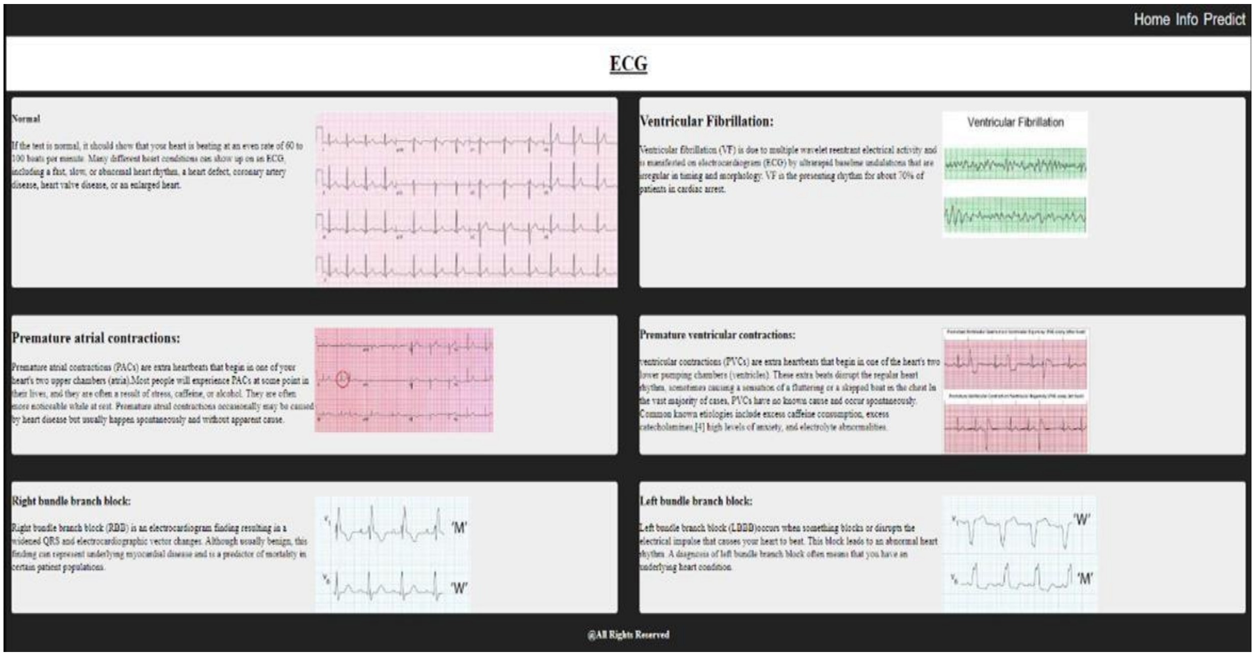
```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Info</title>

    <style>
        body {

            margin: 0px;
            padding: 0px;
        }



        .navbar {

            padding: 20px 0px 40px;
            ;
            background-color: #222;

            font-size: 25px;
            text-align: center;


        }

        .navbar a {
            color: #eee;
            float: right;
            text-decoration: none;
            font-style: normal;
            font-family: sans-serif;

```

```
                padding-right: 10px;
            }

            .navbar a:hover {
                background-color: rgb(0, 0, 0);
                color: rgb(17, 194, 238);
                border-radius: 5px;
                padding: 5px;
            }

            .content{

                display: grid;
 grid-template-columns: 1fr 1fr;
 justify-content: space-between;
    gap: 20px;
    background-color: #222;

            }
          .content .info{
                display: grid;
              grid-template-columns: 1fr 1fr;
              margin: 10px;
              background-color: #eee;
              border-radius: 5px;
height: auto;
            }

            .info img{
                margin-top: 20px;        }
                footer{
                display: flex;
                justify-content: center;
                background-color: #222;
    margin-top: -10px;
    color: white;
```

```



        }



    </style>
</head>

<body>
    <header class="navbar">
        <a href="/upload">Predict</a>
        <a href="/info">Info</a>
        <a href="/home">Home</a>
    </header>
    <div>
        <h1 style="text-align: center; text-decoration: underline;">ECG</h1>
        <div class="content">
            <div class="info">
                <span>
                    <h4>Normal</h4>
                    If the test is normal, it should show that your heart is beating at an even rate of 60 to 100 beats per minute. Many different heart conditions can show up
                </span>

                <img src="data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD/2wCEAAoHCBUSFRgVEhYZGBgYHBoYGBgZHB4SGRgYGBkcGRkYGRgcJS41HB4rIRgYJjgmKy8xNTU1GiQ7QD00Py40NTEBDAwMEA
            </div>
            <div class="info">
                <span>
                   <h2>Ventricular Fibrillation: </h2> Ventricular fibrillation (VF) is due to multiple wavelet reentrant electrical activity and is manifested on electrocardi
                </span>
                <img src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcRy4T8SLf6JpguXbOnLbmzE2L6sTmH3LKX8og&usqp=CAU" alt="">
            </div>
            <div class="info">
                <span>

                    <h2>Premature atrial contractions:</h2> Premature atrial contractions (PACs) are extra heartbeats that begin in one of your heart's two upper chambers (atr


```

```html
                </span>
                <img src="data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD/2wCEAAoHCBYVFRgWFRYZGRgYHB8YGBgaHB8aHBwYHBolGhoaHBgcJC4lHB4rHxocJjgmKy8xNTU1GiQ7QDs0Py40NTEBDAwMEA
            </div>
            <div class="info">
                <span>
                    <h3> Premature ventricular contractions: </h3> ventricular contractions (PVCs) are extra heartbeats that begin in one of the heart's two lower pumping chamb


                </span>
                <img src="data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD/2wCEAAoGBxQTExYTFBQYFhYWGiIWGRkWGx8gFhkaGSAaGhkZGhkaICsiGh8oHxwfIzQjKCwuMTExGSE3PDcwOysMS4BCwsLDw
            </div>
            <div class="info">
                <span>
                    <h3> Right bundle branch block:</h3> Right bundle branch block (RBB) is an electrocardiogram finding resulting in a widened QRS and electrocardiographic vec
                </span>
                <img src="data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD/2wCEAAoGBxMQExIQEBQTERAWGhgWFhAWGRkWFhAWFhoYFxYWFhYZHyoiGRwnHRcWIzQjJysuMTExGCE2OzYwOiowMS4BCwsLDw
            </div>

            <div class="info">
                <span>
                    <h3>  Left bundle branch block:</h3> Left bundle branch block (LBBB)occurs when something blocks or disrupts the electrical impulse that causes your heart t


                </span>
                <img src="data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD/2wCEAAoGBxQRExQQEBQRFhEQGhsWEBAWFBoWEBARFhsaGRYWFhYaICsiGhwoHxYWIzQjKCwuMTExGSE3PDcwOysMS4BCwsLDw
            </div>



        </div>
    </div>
    <footer>
        <h4>@All Rights Reserved</h4>
    </footer>

</body>

</html>
```

OUTPUT:



7.2.3 PREDICT. HTML:

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Predict</title>
</head>

<style>
    body {

        margin: 0px;
        padding: 0px;
    }



    .navbar {

        padding: 20px 0px 40px;
        ;
        background-color: #222;

        font-size: 25px;
        text-align: center;


    }

    .navbar a {
        color: #eee;
        float: right;
        text-decoration: none;
        font-style: normal;
        font-family: sans-serif;
```

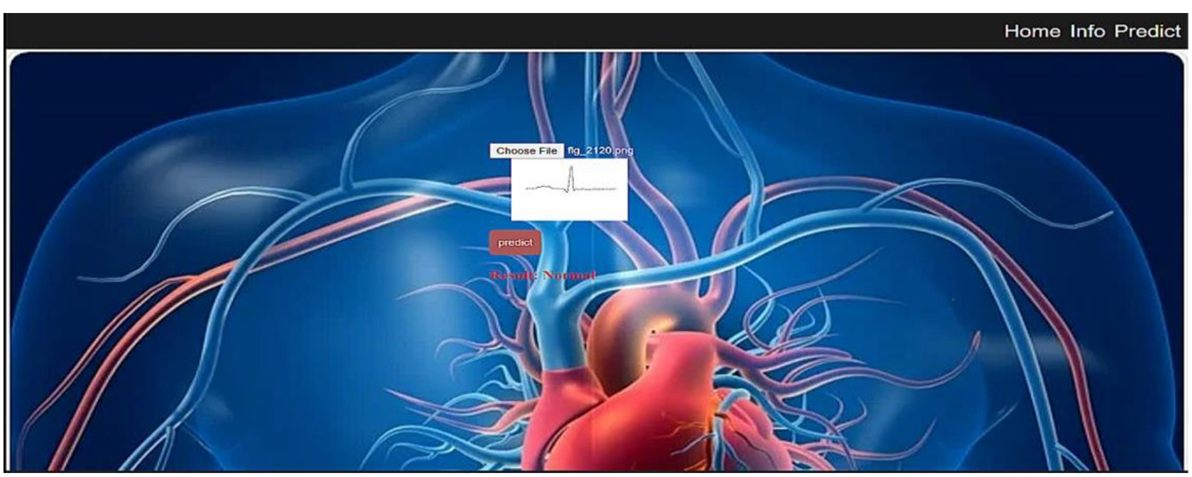
```css

    padding-right: 10px;
}

.navbar a:hover {
    background-color: rgb(0, 0, 0);
    color: rgb(17, 194, 238);
    border-radius: 5px;
    padding: 5px;
}

.content{

background-image: url("https://www.news-medical.net/image.axd?picture=2021%2F1%2Fshutterstock_1576424071.jpg");
height: 90vh;
background-repeat: no-repeat;
background-size:cover;
width: 100%;
position: relative;
}



.wrapper {
    display: grid;
    position: absolute;
    justify-content: center;
    align-items: center;

    width: 40%;
    margin: 10% auto;
    color: #eee;
    margin-left: 30%;


}

```

```
        input[type=button]{
            padding: 10px;
            background-color: rgb(170, 89, 89);
            border: 0;
            border-radius: 5px;
            color: white;
        }

        input[type=button]:hover{
            padding: 12px;
            background-color: rgb(7, 7, 7);
            color: white;


        }
        footer{
                display: flex;
                justify-content: center;
                background-color: #222;
        margin-top: -10px;
        color: white;


            }
    </style>

    <body>
        <header class="navbar">
            <a href="/upload">Predict</a>
            <a href="/info">Info</a>
            <a href="/home">Home</a>
        </header>
        <div  class="content">
            <div class="wrapper">
            <form method="post" id="upload-file" enctype="multipart/form-data">

                <div>
```

```html
            <div>

                <input type="file" name="file" id="imageUpload" required="true">
            </div>
            <div class="image-section" style="display:none;">
            <div class="holder" style="margin-left: 10%">
                <img id="imgPreview" src="#" alt="pic" width="60%" />
            </div>
            </div>
            <div style="margin-top:10px; ">
                <input type="button" value="predict" id="btn-predict" >
            </div>

            <div class="loader" style="display:none;"></div>

           <h3 style="color:rgb(211, 23, 23)" id="result">
        <span> </span>
    </h3>


        </form>
    </div>
    </div>

    <div>
        <div >

        </div>
    </div>
    <footer>
        <h4>@All Rights Reserved</h4>
    </footer>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js">
    </script>

    <script>
        $(document).ready(() => {
            $("#imageUpload").change(function () {
```

```
        <div >

        </div>
    </div>
    <footer>
        <h4>@All Rights Reserved</h4>
    </footer>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js">
    </script>

    <script>
        $(document).ready(() => {
            $("#imageUpload").change(function () {
                const file = this.files[0];
                if (file) {
                    let reader = new FileReader();
                    reader.onload = function (event) {
                        $("#imgPreview")
                            .attr("src", event.target.result);
                    };
                    reader.readAsDataURL(file);
                }
            });
        });
    </script>
    <script src="{{ url_for('static', filename='js/main.js') }}" type="text/javascript"></script>

</body>

</html>
```

OUTPUT:



TESTING

8.1 Test Cases:

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Home page	3	0	3	3
Information page	6	1	1	5
Predict page	2	0	0	2
Final Report Output	4	0	0	4
Version Control	2	0	0	2

8.2 User Acceptance Testing:

Date	15-Nov-22
Team ID	PNT2022TMID19938
Project Name	Classification of Arrhythmia by Using Deep Lea
Maximum Marks	4 marks

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status
Navigation	Functional	Home Page	Validate all the tabs in the navigator		1.Enter URL and click go		All the three tabs should be visible	Working as	pass
Home	Functional	Home page	Verify the visibility of the video		1.Enter URL and click go		User should able to view the video	Working as	Pass
Home pageTC_OO2	Functional	Home Page	Validate the description of the image		1.Enter URL and click go		Description should be visible on	Working as	Pass
Home pageTC_OO3	Functional	Home Page	Verify the user is able to navigate to		1.Enter the URL and click go		It should redirect the user to the	Working as	pass
Inforation_page_T	Functional	Introductio	Verify the user is in the introduction		1.Enter the URL and click go		User should be in the introduction	Working as	pass
Information_	Functional	Introductio	Verify the page title and information		1. Enter the URL and click go		User should able to view the	Working	pass
Predict page	Functional	predict	Verify the working of predict page		1.Enter the URL and click go		User should be able to visit the	Working	pass
Predict page	Functional	predict	Verify the upload image option		1.Enter the URL and click go		Make sure the option works	Working	Pass
Predict	Functional	predict	Verify the choose button is enabled		1.Enter the URL and click go		The choose button option should	Working as	Pass
Predict	Functional	predict	Verify the user is able to access		1.Enter the URL and click go		Image should be uploaded	Working as	pass
Predict	Functional	predict	Verify the selected image is same		1.Enter the URL and click go		Selected image should be an ECG 2D	Working as	pass
Predict	Functional	predict	Verify the working condition of the		1.Enter the URL and click go		The type of arrhythmia should be	Working as	pass

FLOW CHART & RESULTS WITH SCREENSHOTS:

9.1 FLOW CHART & RESULTS BY TRAINING MODEL IN LOCAL MACHINE:-

A. DATASET COLLECTION:

The dataset contains six classes:

1. Left Bundle Branch Block
2. Normal
3. Premature Atrial Contraction
4. Premature Ventricular Contractions
5. Right Bundle Branch Block
6. Ventricular Fibrillation

B. IMAGE PREPROCESSING:

Image Pre-processing includes the following main tasks

- **Import ImageDataGenerator Library:**

Image data augmentation is a technique that can be used to artificially expand the size of a training dataset by creating modified versions of images in the dataset.
The Keras deep learning neural network library provides the capability to fit models using image data augmentation via the ImageDataGenerator class.

```
In [3]:
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Convolution2D
from tensorflow.keras.layers import MaxPooling2D
from tensorflow.keras.layers import Flatten
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

- **Configure ImageDataGenerator Class:**

There are five main types of data augmentation techniques for image data; specifically:

1. Image shifts via the width_shift_range and height_shift_range arguments.
2. Image flips via the horizontal_flip and vertical_flip arguments.
3. Image rotates via the rotation_range argument 4. Image brightness via the brightness_range argument.

5. Image zooms via the zoom_range argument.

```
In [4]: train_datagen = ImageDataGenerator(rescale = 1./255,shear_range = 0.2,zoom_range = 0.2,horizontal_flip = True)
```

```
In [5]: test_datagen = ImageDataGenerator(rescale = 1./255)
```

- **Applying ImageDataGenerator functionality to the trainset and test set:**

We will apply ImageDataGenerator functionality to Trainset and Testset by using the following code

This function will return batches of images from the subdirectories Left Bundle Branch Block, Normal, Premature Atrial Contraction, Premature Ventricular Contractions, Right Bundle Branch Block and Ventricular Fibrillation, together with labels 0 to 5{'Left Bundle Branch Block': 0, 'Normal': 1, 'Premature Atrial Contraction': 2, 'Premature Ventricular Contractions': 3, 'Right Bundle Branch Block': 4, 'Ventricular Fibrillation': 5}

```
In [6]: x_train = train_datagen.flow_from_directory("/content/data/train",target_size = (64,64),batch_size = 32,class_mode = "categorical")
```

Found 15341 images belonging to 6 classes.

```
In [7]: x_test = test_datagen.flow_from_directory("/content/data/test",target_size = (64,64),batch_size = 32,class_mode = "categorical")
```

Found 6825 images belonging to 6 classes.

```
In [8]: x_train.class_indices

Out[8]: {'Left Bundle Branch Block': 0,
         'Normal': 1,
         'Premature Atrial Contraction': 2,
         'Premature Ventricular Contractions': 3,
         'Right Bundle Branch Block': 4,
         'Ventricular Fibrillation': 5}
```

We can see that for training there are 15341 images belonging to 6 classes and for testing there are 6825 images belonging to 6 classes.

C. MODEL BUILDING

We are ready with the augmented and pre-processed image data,we will begin our build our model by following the below steps:

- **Import the model building Libraries:**

In [9]:
```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten
```

- **Initializing the model:**

Keras has 2 ways to define a neural network:

1. Sequential

2.Function API

The Sequential class is used to define linear initializations of network layers which then, collectively, constitute a model. In our example below, we will use the Sequential constructor to create a model, which will then have layers added to it using the add () method. Now, will initialize our model.

- **Adding CNN Layers:**

We are adding a convolution layer with an activation function as "relu" and with a small filter size (3,3) and a number of filters as (32) followed by a max-pooling layer.

The Max pool layer is used to downsample the input. The flatten layer flattens the input.

Initialize the model

In [10]:
```
model=Sequential()
```

Adding CNN layers

```
In [11]:
model.add(Convolution2D(32,(3,3),activation="relu",strides=(1,1),input_shape=(64,64,3)))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 62, 62, 32)        896

 max_pooling2d (MaxPooling2D  (None, 31, 31, 32)       0
 )

 flatten (Flatten)           (None, 30752)             0

=================================================================
Total params: 896
Trainable params: 896
Non-trainable params: 0
_________________________________________________________________
```

- **Adding Hidden Layers:**

Dense layer is deeply connected neural network layer. It is most common and frequently used layer.

```
In [12]:
model.add(Dense(500,activation="relu"))
model.add(Dense(500,activation="relu"))
```

- **Adding Output Layer:**

```
In [13]:
model.add(Dense(6,activation="softmax"))
```

Understanding the model is very important phase to properly use it for training and prediction purposes. Keras provides a simple method, summary to get the full information about the model and its layers.

- **Configure the Learning Process:**

■ The compilation is the final step in creating a model. Once the compilation is done, we can move on to the training phase. The loss function is used to find error or deviation in the learning process. Keras requires loss function during the model compilation process.

■ Optimization is an important process that optimizes the input weights by comparing the prediction and the loss function. Here we are using adam optimizer

■ Metrics is used to evaluate the performance of your model. It is similar to loss function, but not used in the training process.

```
In [14]: model.compile(loss="categorical_crossentropy",optimizer="adam",metrics=['accuracy'])
         len(x_train)
```

- **Training the model:**

We will train our model with our image dataset. fit_generator functions used to train a deep learning neural network.

```
In [15]: model.fit(x_train,epochs=5,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test))

Epoch 1/5
480/480 [==============================] - 140s 289ms/step - loss: 1.0137 - accuracy: 0.6721 - val_loss: 0.6701 - val_accuracy: 0.7538
Epoch 2/5
480/480 [==============================] - 142s 296ms/step - loss: 0.3082 - accuracy: 0.9022 - val_loss: 0.5529 - val_accuracy: 0.8101
Epoch 3/5
480/480 [==============================] - 140s 291ms/step - loss: 0.1732 - accuracy: 0.9475 - val_loss: 0.4951 - val_accuracy: 0.8379
Epoch 4/5
480/480 [==============================] - 137s 286ms/step - loss: 0.1226 - accuracy: 0.9627 - val_loss: 0.5920 - val_accuracy: 0.8503
Epoch 5/5
480/480 [==============================] - 139s 290ms/step - loss: 0.0892 - accuracy: 0.9727 - val_loss: 0.4156 - val_accuracy: 0.8752
```

- **Saving the model:**

The model is saved with .h5 extension as follows

An H5 file is a data file saved in the Hierarchical Data Format (HDF).It contains multidimensional arrays of scientific data.

```
In [16]: model.save('ECG.h5')
```

- **Testing the model:**

Load necessary libraries and load the saved model using load_model

Taking an image as input and checking the results

Note: The target size should for the image that is should be the same as the target size that you have used for training.

In [17]:

```
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
model=load_model('ECG.h5')
img=image.load_img("/content/data/test/Right Bundle Branch Block/fig_101.png",target_size=(64,64))
img
```

Out[17]:

In [18]:

```
x=image.img_to_array(img)
x
```

In [22]:

```
pred=model.predict(x)
```

```
1/1 [==============================] - 0s 79ms/step
```

In [23]:

```
pred
```

Out[23]:

```
array([[0., 0., 0., 0., 1., 0.]], dtype=float32)
```

In [24]:

```
index=['Left Bundle Branch Block',
 'Normal',
 'Premature Atrial Contraction',
 'Premature Ventricular Contractions',
 'Right Bundle Branch Block',
 'Ventricular Fibrillation']
index[np.argmax(pred)]
```

Out[24]:

```
'Right Bundle Branch Block'
```

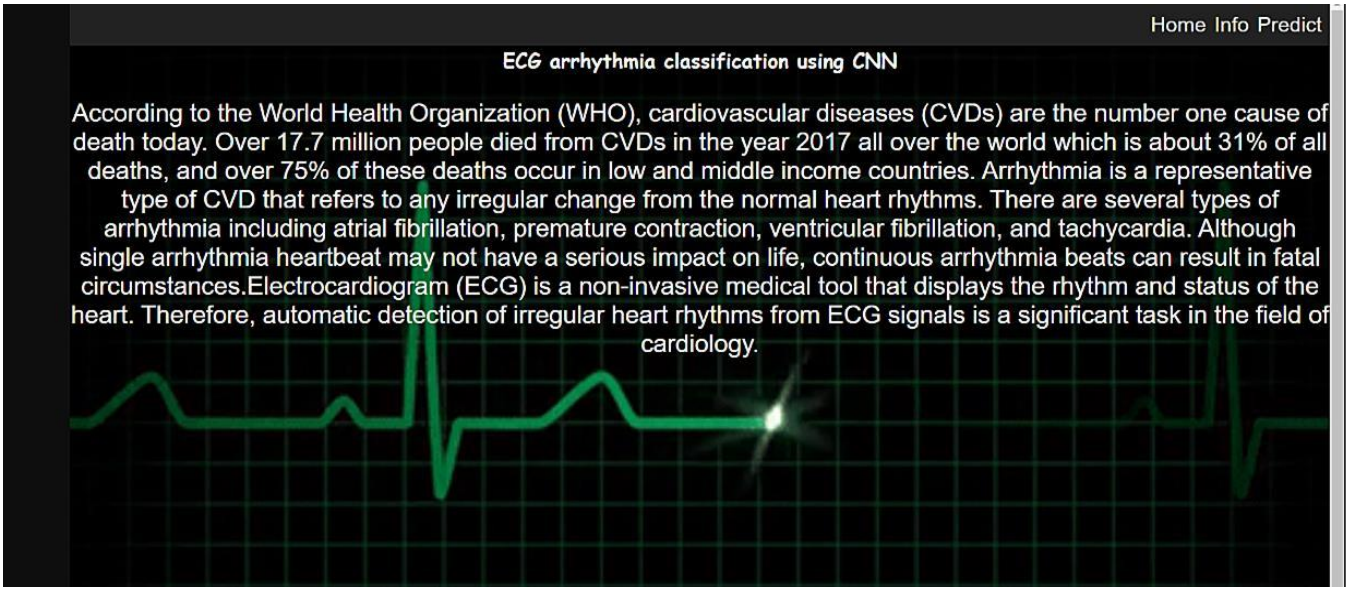
D. APPLICATION BUILDING:

In this section, we will be building a web application that is integrated into the model we built. A UI is provided for the uses where he has uploaded an image. The uploaded image is given to the saved model and prediction is showcased on the UI.

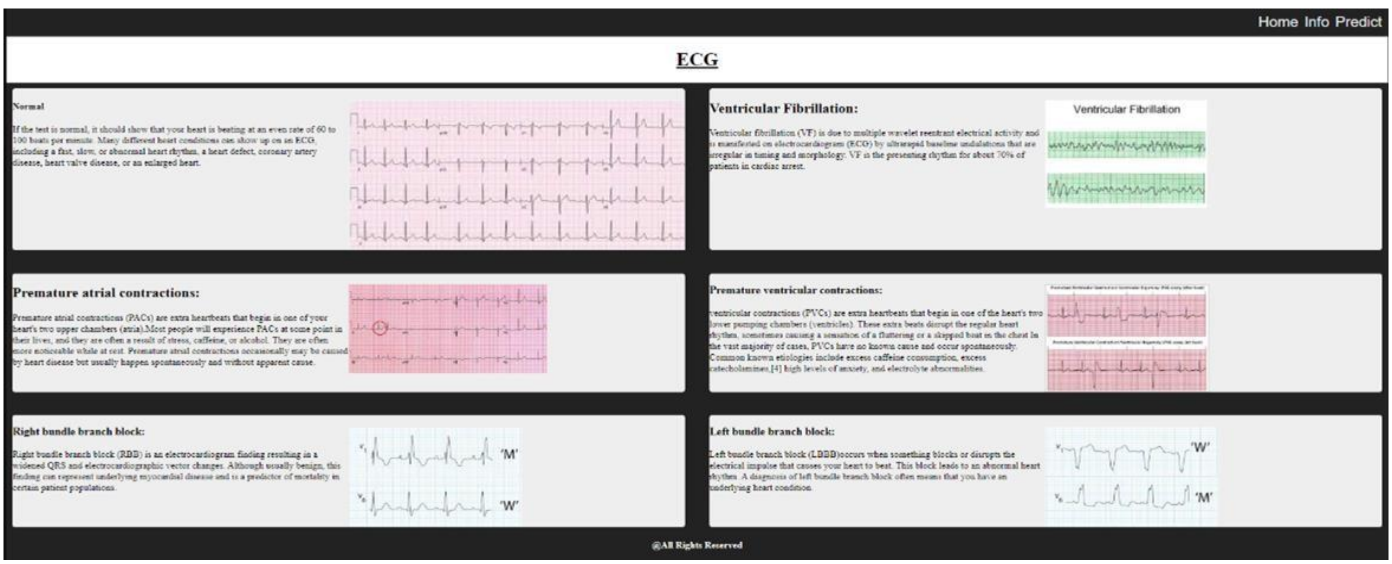
This section has the following tasks

- **Building HTML Pages:**
  - We use HTML to create the front end part of the web page.
  - Here,we created 4 html pages- home.html, predict_base.html,predict.html, information.

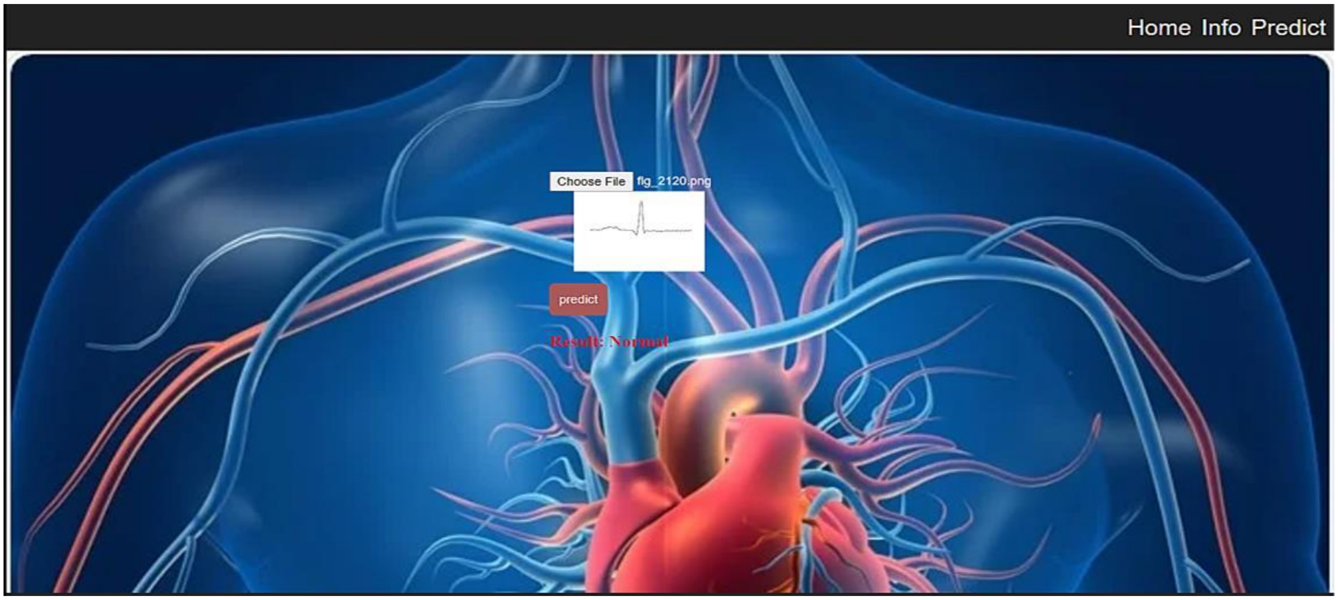
- home.html displays the home page.



- information.html displays all important details to be known about ECG.



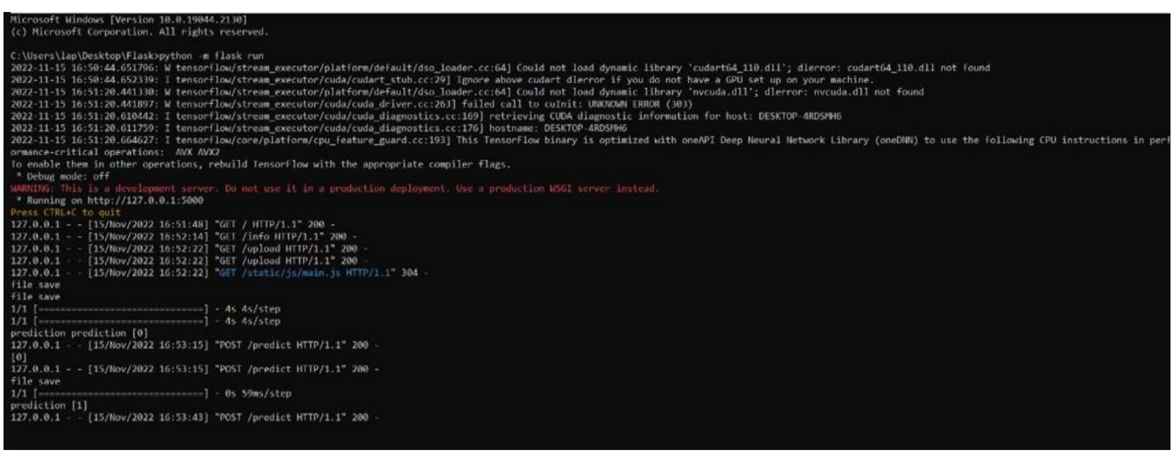
■ Predict-base.html and predict.html accept input from the user and predicts the values.



- **Building server-side script:**

We will build the flask file ‘app.py’ which is a web framework written in python for server-side scripting.

■ The app starts running when the “__name__” constructor is called in main.
■ render_template is used to return HTML file.
■ “GET” method is used to take input from the user.
■ “POST” method is used to display the output to the user.



```
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\lap\Desktop\Flask>python -m flask run
2022-11-15 16:50:44.651796: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'cudart64_110.dll'; dlerror: cudart64_110.dll not found
2022-11-15 16:50:44.652339: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart dlerror if you do not have a GPU set up on your machine.
2022-11-15 16:51:20.441330: W tensorflow/stream_executor/platform/default/dso_loader.cc:64] Could not load dynamic library 'nvcuda.dll'; dlerror: nvcuda.dll not found
2022-11-15 16:51:20.441897: W tensorflow/stream_executor/cuda/cuda_driver.cc:263] failed call to cuInit: UNKNOWN ERROR (303)
2022-11-15 16:51:20.610442: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:169] retrieving CUDA diagnostic information for host: DESKTOP-4RDSMH6
2022-11-15 16:51:20.611759: I tensorflow/stream_executor/cuda/cuda_diagnostics.cc:176] hostname: DESKTOP-4RDSMH6
2022-11-15 16:51:20.664627: I tensorflow/core/platform/cpu_feature_guard.cc:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network Library (oneDNN) to use the following CPU instructions in perf
ormance-critical operations:  AVX AVX2
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [15/Nov/2022 16:51:48] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [15/Nov/2022 16:52:14] "GET /info HTTP/1.1" 200 -
127.0.0.1 - - [15/Nov/2022 16:52:22] "GET /upload HTTP/1.1" 200 -
127.0.0.1 - - [15/Nov/2022 16:52:22] "GET /upload HTTP/1.1" 200 -
127.0.0.1 - - [15/Nov/2022 16:52:22] "GET /static/js/main.js HTTP/1.1" 304 -
file save
file save
1/1 [==============================] - 4s 4s/step
1/1 [==============================] - 4s 4s/step
prediction prediction [0]
127.0.0.1 - - [15/Nov/2022 16:53:15] "POST /predict HTTP/1.1" 200 -
[0]
127.0.0.1 - - [15/Nov/2022 16:53:15] "POST /predict HTTP/1.1" 200 -
file save
1/1 [==============================] - 0s 59ms/step
prediction [1]
127.0.0.1 - - [15/Nov/2022 16:53:43] "POST /predict HTTP/1.1" 200 -
```

- **Running The App:**

```
C:\Users\lap\Desktop\Flask>python -m flask run
```

```
To enable them in other operations, rebuild TensorFlow with the appropriate compiler flags.
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
```

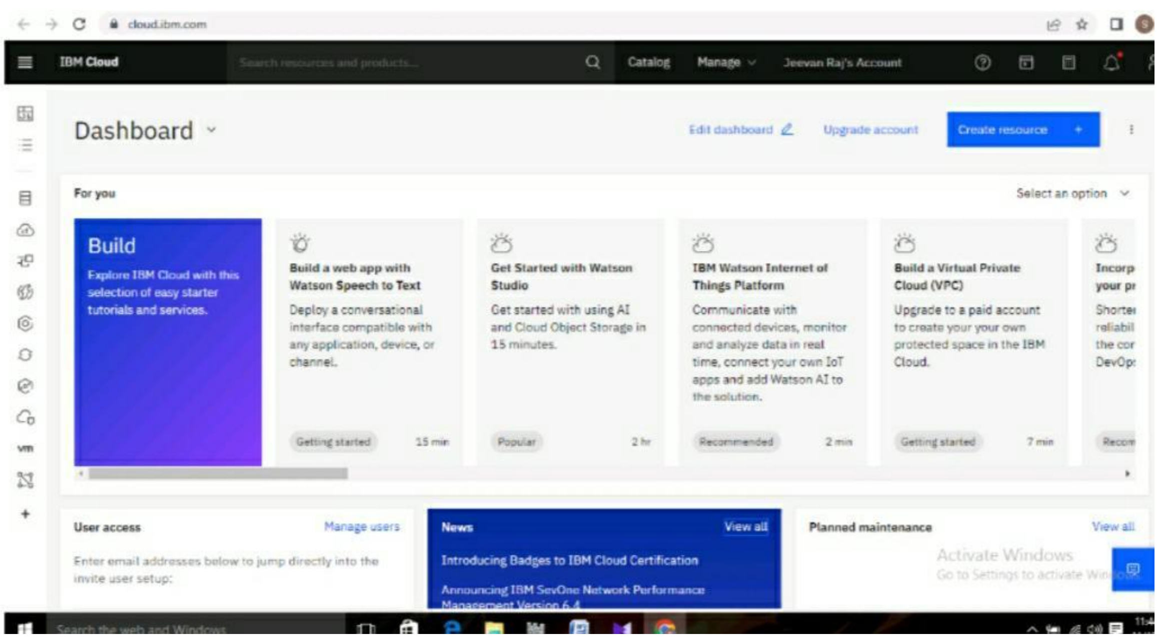
Navigate to the localhost (http://127.0.0.1:5000/)where you can view your web page.

9.2 Flow Chart & Results by training model in IBM WATSON STUDIO:

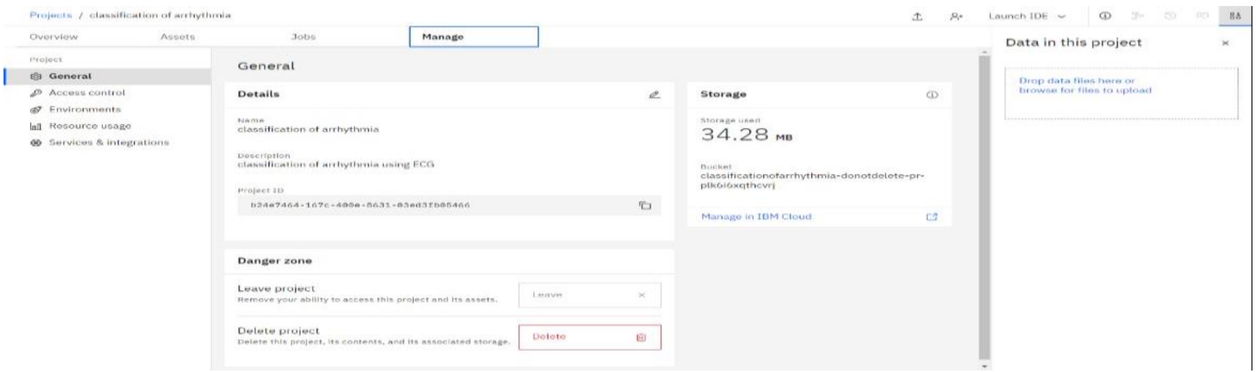
A. Creating IBM cloud account:-

We have to create an IBM Cloud Account and should log in.

B. Register for IBM Cloud account:-



C. Deployment space in the watson studio:-



D. Apply CNN algorithm and save the model and deploy it using API key generated:-

In [120…

```
model.save('ECG.h5')
```

In [121…

```
!tar -zcvf ECG-model_new.tgz ECG.h5
```

```
ECG.h5
```

In [134…

```
ls -1
```

```
data/
ECG.h5
ECG-model_new.tgz
my_modell.tar1.gz
```

In [95]:

```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
                    "url": "https://us-south.ml.cloud.ibm.com",
                    "apikey":"m6NbiyMRA29yQcTssg5Wi0HkkZHyK-gib4hSGEJppyVy"
                  }
client = APIClient(wml_credentials)
```

In [124…

```
client = APIClient(wml_credentials)
```

In [125…

```
client.spaces.list()
```

```
Note: 'limit' is not provided. Only first 50 records will be displayed if the number of records exceed 50
------------------------------------  --------------  ------------------------
ID                                    NAME            CREATED
d68afc27-adb2-4919-a9d0-cd6acb8700f3  Model building  2022-11-11T06:11:54.400Z
------------------------------------  --------------  ------------------------
```

In [126…

```
def guid_from_space_name(client, space_name):
    space = client.spaces.get_details()
    #print(space)
    return(next(item for item in space['resources'] if item['entity']['name'] == space_name)['metadata']['id'])
```

In [127…

```
space_uid = guid_from_space_name(client, 'Model building')
print("Space UID = "+ space_uid)
```

```
Space UID = d68afc27-adb2-4919-a9d0-cd6acb8700f3
```

In [128…

```
client.set.default_space(space_uid)
```

Out[128…

```
'SUCCESS'
```

```
In [130…
software_spec_uid = client.software_specifications.get_uid_by_name("tensorflow_rt22.1-py3.9")
software_spec_uid

Out[130… 'acd9c798-6974-5d2f-a657-ce06e986df4d'

In [131…
model_details = client.repository.store_model(model='ECG-model_new.tgz',meta_props={
    client.repository.ModelMetaNames.NAME:"ECG_Model",
    client.repository.ModelMetaNames.TYPE:"tensorflow_2.7",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid})
model_id=client.repository.get_model_uid(model_details)

This method is deprecated, please use get_model_id()
/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/ibm_watson_machine_learning/repository.py:1453: UserWarning: This method is deprecated, please use get_model_id()
  warn("This method is deprecated, please use get_model_id()")

In [132…
model_id

Out[132… '7bc41052-6999-4af0-ba61-aebca5eaa72e'

In [133…
client.repository.download(model_id,'my_modell.tar1.gz')
```

E. For downloading the model we have to run the last part of the above code in the local jupyter notebook:

```
In [133…
client.repository.download(model_id,'my_modell.tar1.gz')
```

Hence we trained the model using IBM Watson.

ADVANTAGES & DISADVANTAGES:

6.1 ADVANTAGES:

i. The proposed model predicts Arrhythmia in images with a high accuracy rate of nearly 96%
ii. The early detection of Arrhythmia gives better understanding of disease causes, initiates therapeutic interventions and enables developing appropriate treatments.

6.2 DISADVANTAGES:

i. Not useful for identifying the different stages of Arrhythmia disease.
ii. Not useful in monitoring motor symptoms

APPLICATIONS :

- It is useful for identifying the arrhythmia disease at an early stage. ● It is useful in detecting cardiovascular disorders

CONCLUSION:

- Cardiovascular disease is a major health problem in today's world. The early diagnosis of cardiac arrhythmia highly relies on the ECG.
- Unfortunately, the expert level of medical resources is rare, visually identify the ECG signal is challenging and time-consuming.
- The advantages of the proposed CNN network have been put to evidence.
- It is endowed with an ability to effectively process the non-filtered dataset with its potential anti-noise features. Besides that, ten-fold cross-validation is implemented in this work to further demonstrate the robustness of the network.

FUTURE SCOPE:

For future work, it would be interesting to explore the use of optimization techniques to find a feasible design and solution. The limitation of our study is that we have yet to apply any optimization techniques to optimize the model parameters and we believe that with the implementation of the optimization, it will be able to further elevate the performance of the proposed solution to the next level.

APPENDIX:

DEMO LINK:
https://drive.google.com/file/d/13_X6D4PJbSlZpsCW6cJ9C2Eefr07phGm/view?usp=drivesdk

GITHUB LINK: https://github.com/IBM-EPbL/IBM-project-51748-1660982847

THE END

TEAM MEMBERS:

Jeevan Raj. V
Bala Murugan. B
Venkatesh Prasath. S
. **Prithivraj. C**