1. INTRODUCTION

1.1 Project Overview

An electrocardiogram (ECG) measures the electric activity of the heart and has been widely used for detecting heart diseases due to its simplicity and non-invasive nature. By analysing the electrical signal of each heartbeat, i.e., the combination of action impulse waveforms produced by different specialised cardiac tissues found in the heart, it is possible to detect some of its abnormalities. In the last decades, several works were developed to produce automatic ECG-based heartbeat classification methods. In this work, we survey the current state-of-the-art methods of ECG-based automated abnormalities heartbeat classification by presenting the ECG signal preprocessing, the heartbeat segmentation techniques, the feature description methods and the learning algorithms used. In addition, we describe some of the databases used for evaluation of methods indicated by a well-known standard developed by the Association for the Advancement of Medical Instrumentation (AAMI) and described in ANSI/AAMI EC57:1998/(R)2008 (ANSI/AAMI, 2008). Finally, we discuss limitations and drawbacks of the methods in the literature presenting concluding remarks and future challenges, and also we propose an evaluation process workflow to guide authors in future works.

1.2 Purpose

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 convolutional 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.

2. LITERATURE SURVEY

2.1 Existing problem

Previous studies on arrhythmia were used to diagnose the abnormally fast, slow, or irregular heart rhythm through ECG (Electrocardiogram), which is one of the biological signals. ECG has the form of P-QRS-T wave, and many studies have been done to extract the features of QRS-complex and R-R interval. However, in the conventional method, the P-QRS-T wave must be accurately detected, and the feature value is extracted through the P-QRS-T wave. If an error occurs in the peak detection or feature extraction process, the accuracy becomes very low. Therefore, in this paper, we implement a system that can perform PVC (Premature Ventricular Contraction) and PAC (Premature Atrial Contraction) classification by using P-QRS-T peak value without the feature extraction process using deep neural networks. The parameters were updated for PVC and PAC classification in the learning process using P-QRS-T peak without feature value. As a result of the performance evaluation, we could confirm higher accuracy than the previous studies and omit the process of feature extraction, and the time required for the pre-processing process to construct the input data set is relatively reduced.

2.2. REFERENCES

[1] ***Po-Ya Hsu*** Department of Computer Science & Engineering, University of California San Diego, The USA

Chung-Kuan Cheng Department of Computer Science & Engineering, University of California, San Diego, The USA

https://ieeexplore.ieee.org/document/9176679

[2] ***A. Rajkumar*** Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, India

M. Ganesan Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, India

R. Lavanya Department of Electronics and Communication Engineering, Amrita School of Engineering, Coimbatore, India

https://ieeexplore.ieee.org/abstract/document/8728362

[3]*EunKwang Jeon* Dept. of Computer Science & Engineering, Soonchunhyang University, Asan, South Korea

MinSu Chae Dept. of Computer Science & Engineering, Soonchunhyang University, Asan, South Korea

Sangwook Han Dept. of Computer Science & Engineering, Soonchunhyang University, Asan, South Korea

https://ieeexplore.ieee.org/document/8989066

[4] Amin Ullah University of Engineering and Technology

Taxila Syed Anwar University of Engineering and Technology

Taxila Muhammad Bilal Hankuk University of Foreign Studies

Raja Majid Mehmood Xiamen University Malaysia

https://arxiv.org/abs/2005.06902

[5] ***Rémi Dekimpe,***ICTEAM Institute, Université Catholique de Louvain, Louvain-la-Neuve, Belgium

David Bol ICTEAM Institute, Université Catholique de Louvain, Louvain-la-Neuve, Belgium

https://ieeexplore.ieee.org/document/9795058

2.3 Problem Statement Definition

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 convolutional neural network (CNN),in which we classify ECG into seven categories,one being normal and the other side 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.

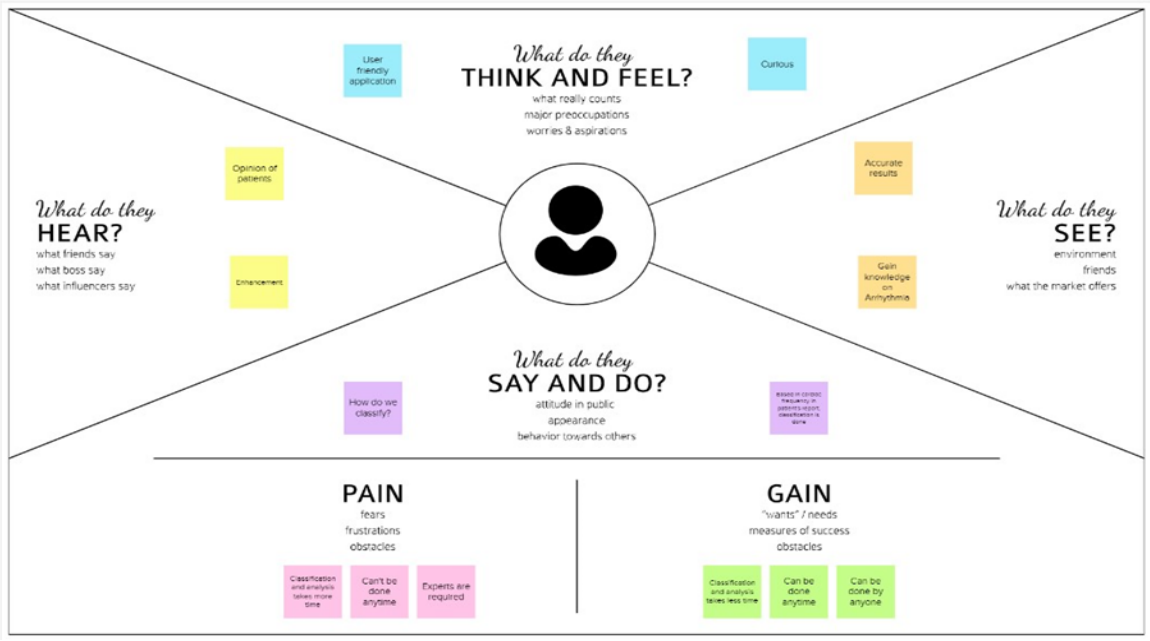
3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

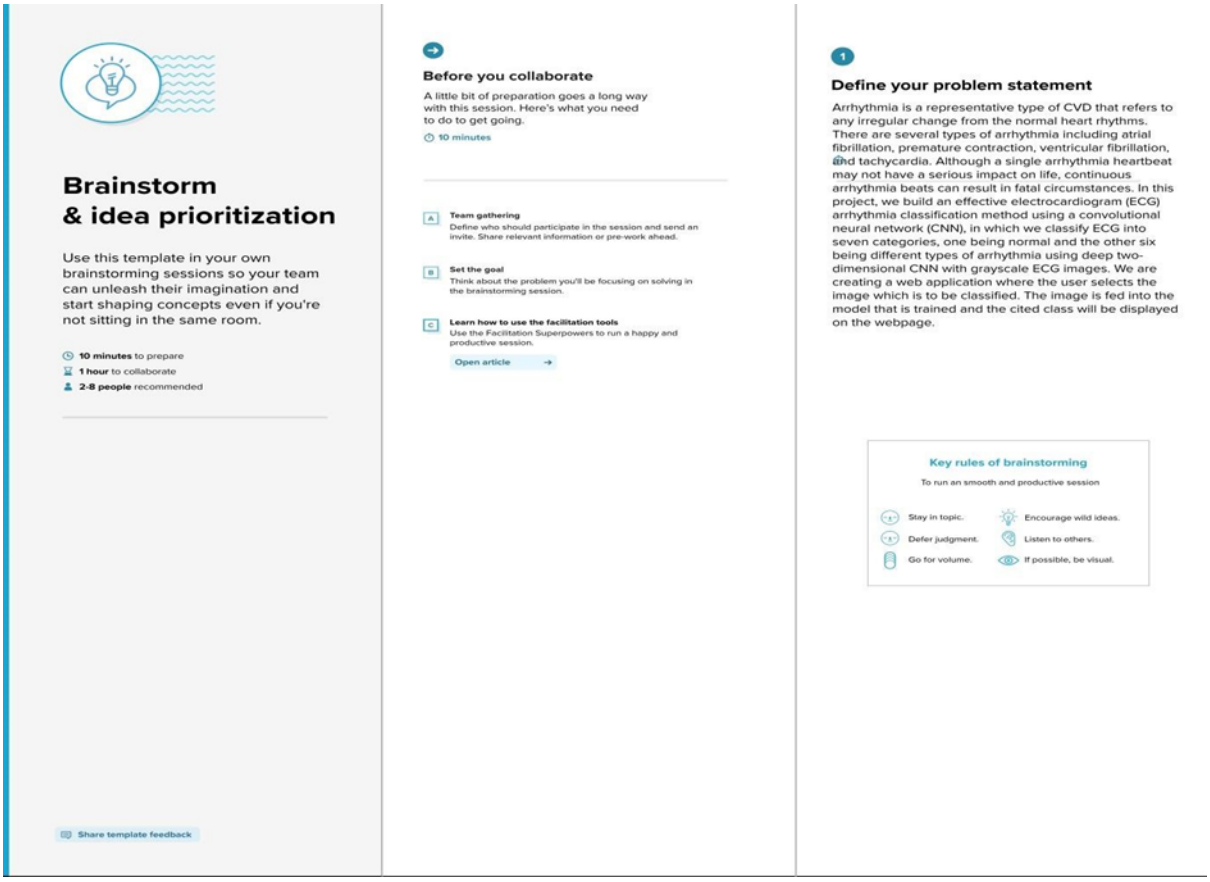
An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to helps teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

1

Build empathy and keep your focus on the user by putting yourself in their shoes.



3.2 Ideation & Brainstorming



Brainstorm & idea prioritization

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.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Share template feedback

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

1

Define your problem statement

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 convolutional 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.

Key rules of brainstorming

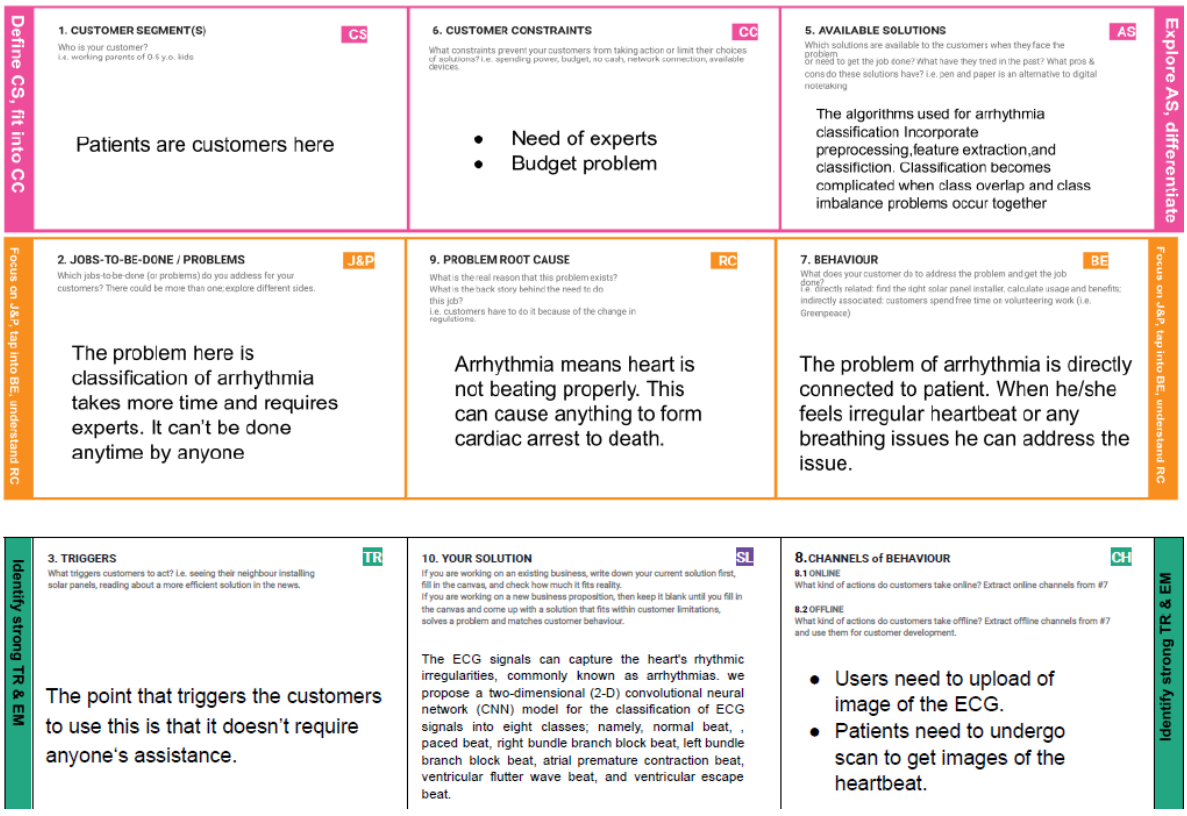
To run an smooth and productive session

Stay in topic.
Defer judgment.
Go for volume.
Encourage wild ideas.
Listen to others.
If possible, be visual.

3.3 Proposed Solution

Classification of arrhythmia using deep learning with 2d ECG spectral image representation.To create an application that is used to classify the arrhythmia and provide more detailed information about it. Here we use deep learning techniques and with the help of 2D ECG spectral image to classify the arrhythmia.Provides accurate results and detailed information required by the users or patients.Users or customers can easily use the app because of its user friendly interface and simplicity. Can be used by anyone at any time.As this application can be very useful for the earlier and fast classification of arrhythmia it we be used by many patients suffering by it.Experts guidance is not required when we have a app that can be used by anyone. Data of the patient will be securely stored and maintained for future purposes.

3.4 Problem Solution fit



4.REQUIREMENT ANALYSIS

4.1 Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Get User Input	Upload image as jpeg Upload image as png
FR-4	Save Image	Images are saved in the uploads folder
FR-5	Chat with Doctor	Consult with Doctor
FR-6	Report Generation	Get complete Report

4.2 Non-Functional requirements

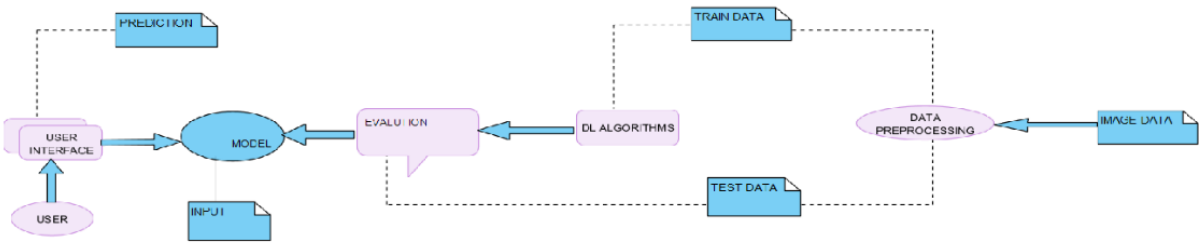
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Classification of Arrhythmia with the help of AI.
NFR-2	**Security**	User's data cannot be accessed by unauthorised people.
NFR-3	**Reliability**	The system performs without failure.
NFR-4	**Performance**	High accuracy.
NFR-5	**Availability**	Anyone who is authorised.
NFR-6	**Scalability**	Does not affect the performance even though used by many users.

5.PROJECT DESIGN

5.1 Data Flow Diagrams

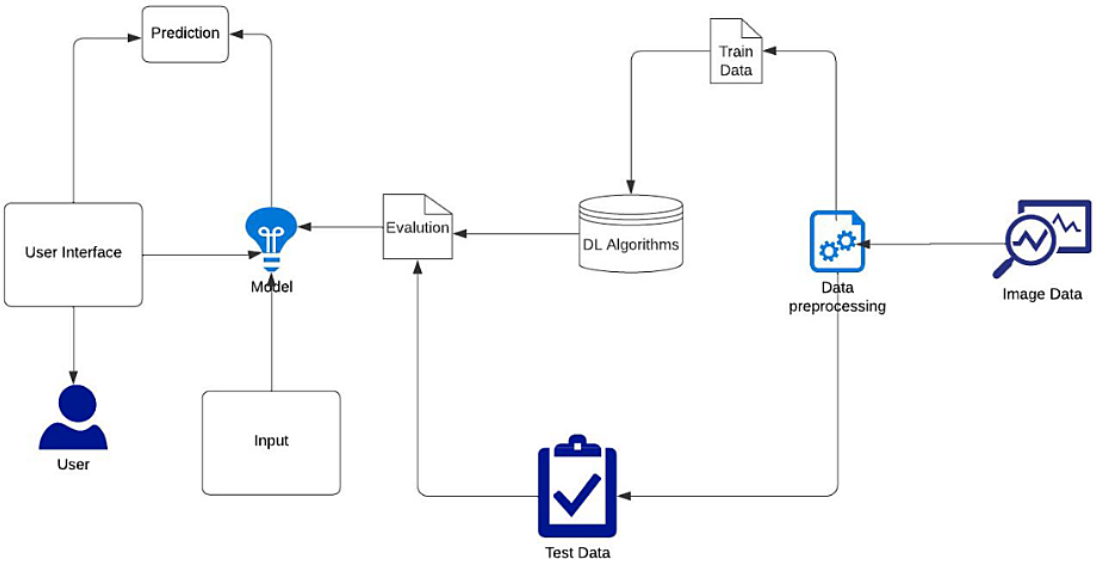
A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs,

outputs, storage points and the routes between each destination.

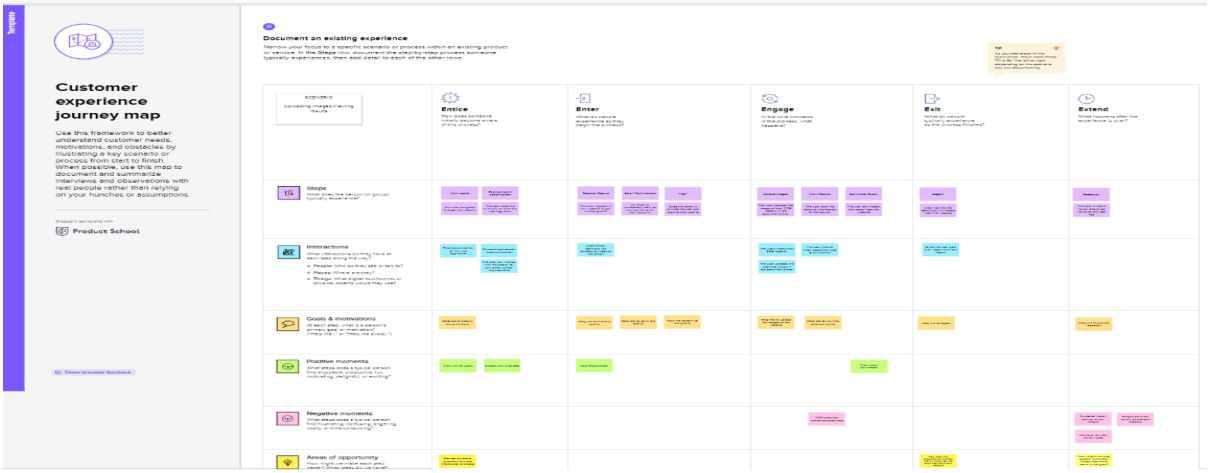


5.2 Solution & Technical Architecture

Technical architecture—which is also often referred to as application architecture, IT architecture, business architecture, etc.—refers to creating a structured software solution that will meet the business needs and expectations while providing a strong technical plan for the growth of the software application through its lifetime. IT architecture is equally important to the business team and the information technology team.Technical architecture includes the major components of the system, their relationships, and the contracts that define the interactions between the components. The goal of technical architects is to achieve all the business needs with an application that is optimised for both performance and security.



5.3 User Stories

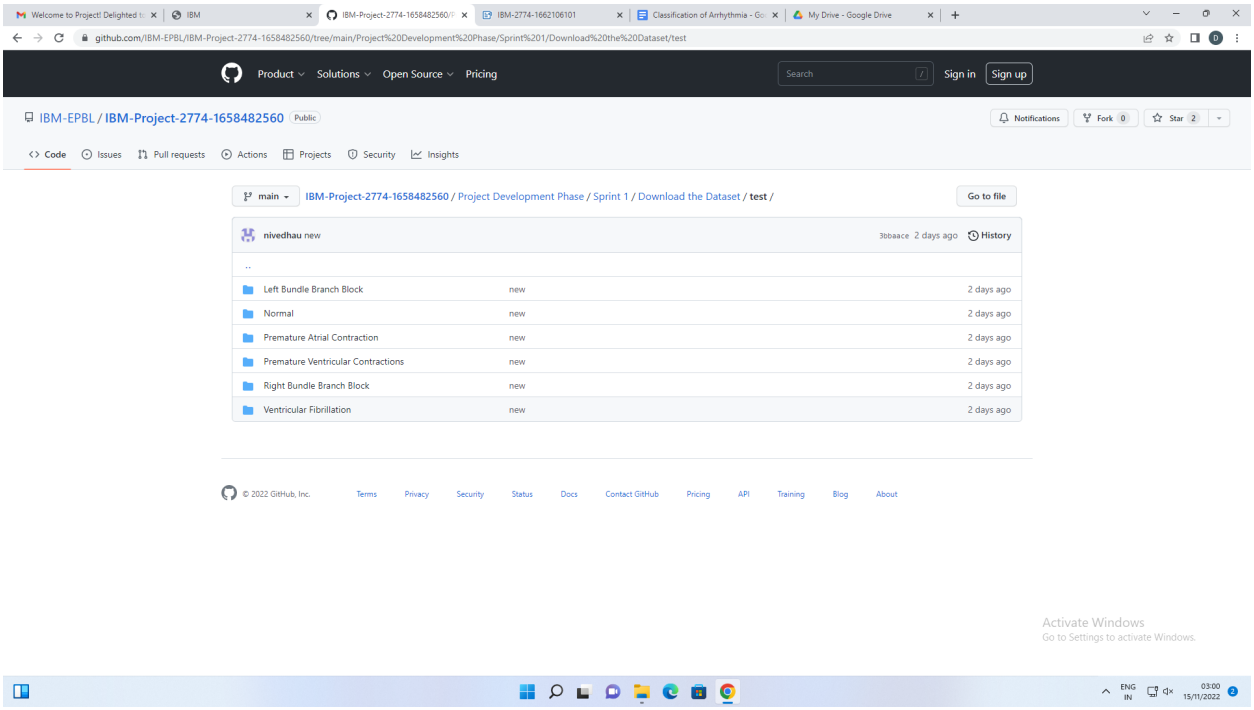


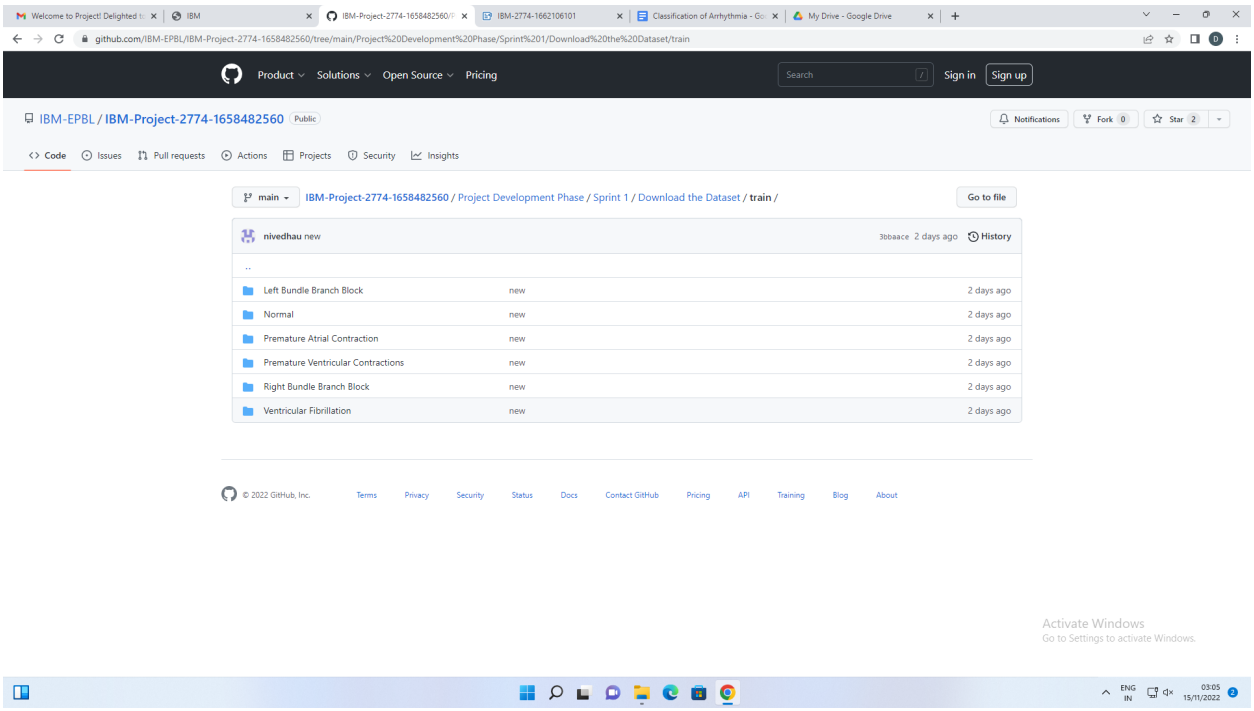
6.PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

SPRINT 1

1. Download the dataset





2. Image Preprocessing

A. Import the ImageDataGenerator Library

Import the ImageDataGenerator library

```
[ ] from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

B. Configure ImageDataGenerator Class

Configure ImageDataGenerator Class

```
[ ] train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,vertical_flip=True,horizontal_flip=True)
```

```
[ ] test_data=ImageDataGenerator(rescale=1./255)
```

C. Apply ImageDataGenerator functionality to trainset and test set

Apply ImageDataGenerator functionality to trainset and testset

```
[ ] x_train=train_datagen.flow_from_directory(r"/content/drive/MyDrive/Project Development Phase/data/train",target_size=(64,64),class_mode="categorical",batch_size=
```

```
Found 15341 images belonging to 6 classes.
```

```
[ ] x_test=test_data.flow_from_directory(r"/content/drive/MyDrive/Project Development Phase/data/test",target_size=(64,64),class_mode="categorical",batch_size=128)
```

```
Found 6825 images belonging to 6 classes.
```

```
[ ] x_train.class_indices
```

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

3. Model Building

A. Import the libraries

MODEL BUILDING

Import the libraries

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

B. Initialize the model

Initialize the model

```
[ ] model=Sequential()
```

C. Adding CNN layers

Adding CNN layers

```
[ ] 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
```


D. Adding Dense layer

Adding Dense layer

Hidden layer

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

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

Output layer

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

E. Train the model

```
[ ] Epoch 1/5
    120/120 [==============================] - 140s 1s/step - loss: 0.1920 - accuracy: 0.9401 - val_loss: 0.4968 - val_accuracy: 0.8731
    Epoch 2/5
    120/120 [==============================] - 147s 1s/step - loss: 0.1607 - accuracy: 0.9512 - val_loss: 0.5703 - val_accuracy: 0.8727
    Epoch 3/5
    120/120 [==============================] - 142s 1s/step - loss: 0.1358 - accuracy: 0.9572 - val_loss: 0.4914 - val_accuracy: 0.8831
    Epoch 4/5
    120/120 [==============================] - 140s 1s/step - loss: 0.1181 - accuracy: 0.9640 - val_loss: 0.5450 - val_accuracy: 0.8794
    Epoch 5/5
    120/120 [==============================] - 133s 1s/step - loss: 0.1109 - accuracy: 0.9666 - val_loss: 0.4703 - val_accuracy: 0.8801
    <keras.callbacks.History at 0x7f527adbd750>
```

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

    Epoch 1/5
    120/120 [==============================] - 127s 1s/step - loss: 0.1022 - accuracy: 0.9692 - val_loss: 0.5888 - val_accuracy: 0.8659
    Epoch 2/5
    120/120 [==============================] - 139s 1s/step - loss: 0.0934 - accuracy: 0.9710 - val_loss: 0.5789 - val_accuracy: 0.8689
    Epoch 3/5
    120/120 [==============================] - 128s 1s/step - loss: 0.0862 - accuracy: 0.9729 - val_loss: 0.4989 - val_accuracy: 0.8848
    Epoch 4/5
    120/120 [==============================] - 128s 1s/step - loss: 0.0778 - accuracy: 0.9765 - val_loss: 0.6542 - val_accuracy: 0.8759
    Epoch 5/5
    120/120 [==============================] - 137s 1s/step - loss: 0.0793 - accuracy: 0.9745 - val_loss: 0.5369 - val_accuracy: 0.8844
    <keras.callbacks.History at 0x7f527adc0a50>
```


F. Save the model

Save the model

```
[ ] model.save('arrhythmia.h5')
```

G. Testing the model

Testing the model

```
[ ] import numpy as np
    from tensorflow.keras.models import load_model
    from tensorflow.keras.preprocessing import image
```

```
[ ] model=load_model('arrhythmia.h5')
```

```
[ ] img=image.load_img("/content/drive/MyDrive/Project Development Phase/data/test/Right Bundle Branch Block/fig_101.png",target_size=(64,64))
```

```
[ ] img
```

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

```
[ ] x
```



```
    [255., 255., 255.]]], dtype=float32)
```

```
[ ] pred=model.predict(x)
    1/1 [==============================] - 0s 43ms/step
```

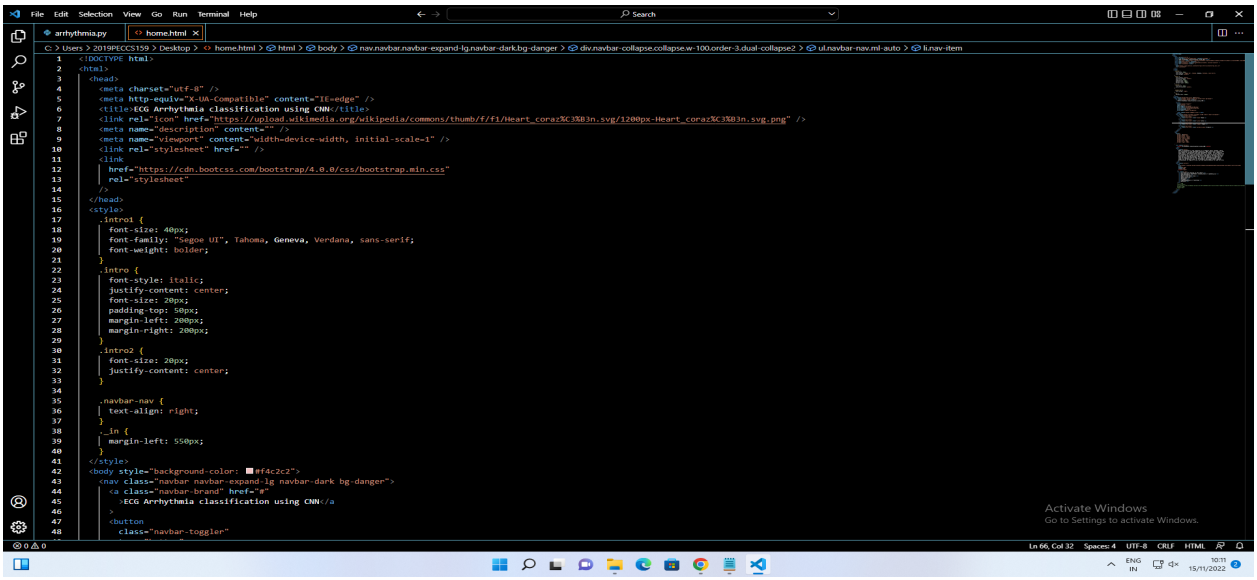
```
[ ] pred
    array([[0., 0., 0., 0., 1., 0.]], dtype=float32)
```

```
[ ] index=['Left Bundle Branch Block',
     'Normal',
     'Premature Atrial Contraction',
     'Premature Ventricular Contractions',
     'Right Bundle Branch Block',
     'Ventricular Fibrillation']
```

```
[ ] index[np.argmax(pred)]
    'Right Bundle Branch Block'
```

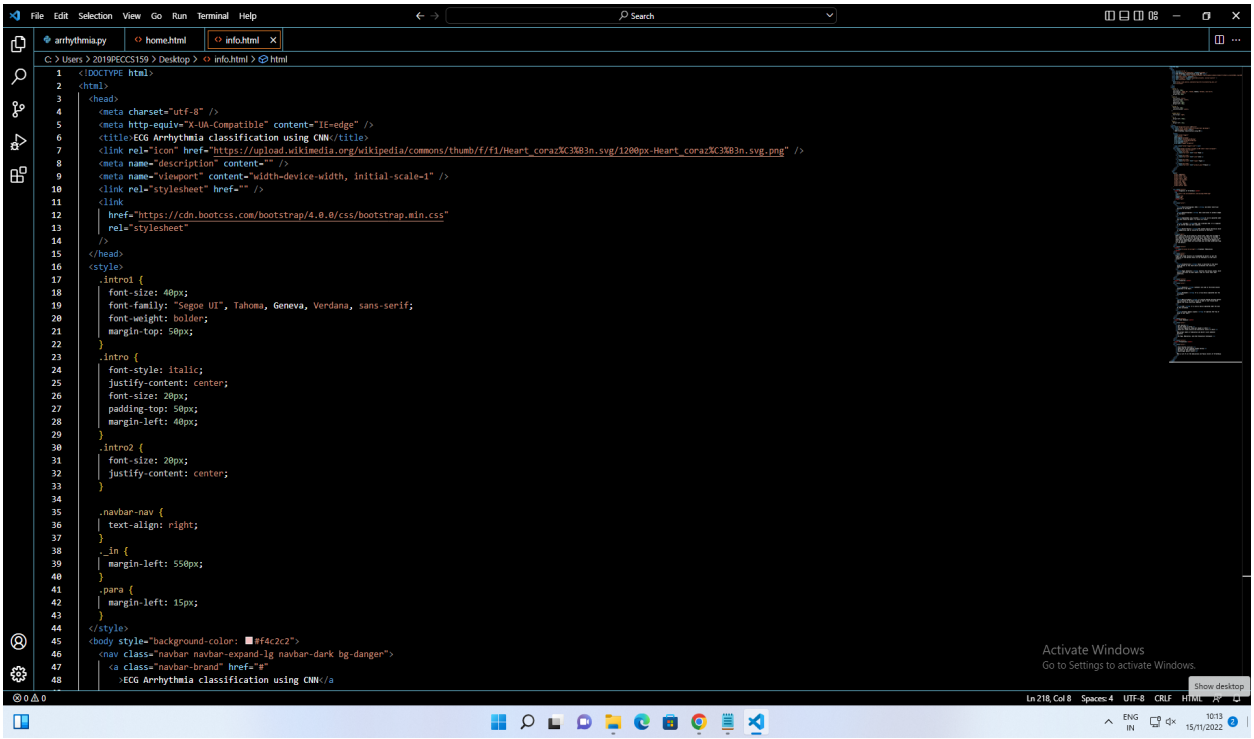
SPRINT 2

A. home.html



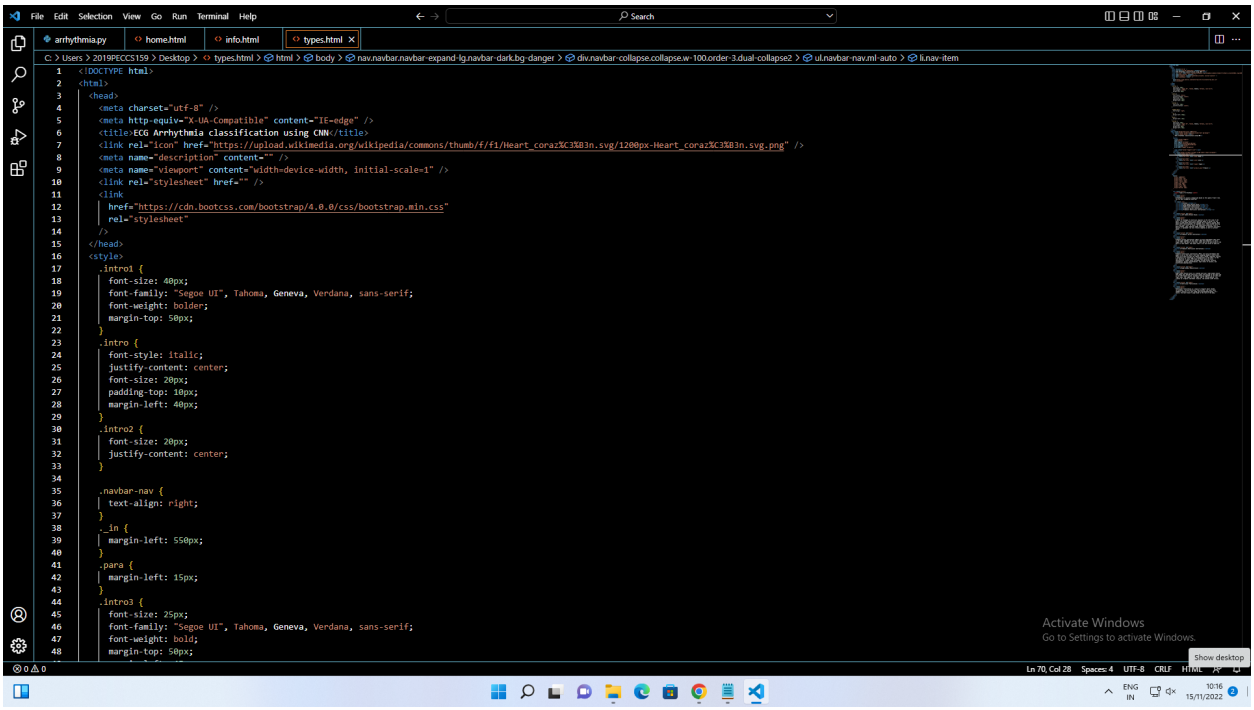
```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <title>ECG Arrhythmia classification using CNN</title>
    <link rel="icon" href="https://upload.wikimedia.org/wikipedia/commons/thumb/f/f1/Heart_coraz%C3%B3n.svg/1200px-Heart_coraz%C3%B3n.svg.png" />
    <meta name="description" content="" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="" />
    <link
      href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
      rel="stylesheet"
    />
  </head>
  <style>
    .intro1 {
      font-size: 40px;
      font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
      font-weight: bolder;
    }
    .intro {
      font-style: italic;
      justify-content: center;
      font-size: 20px;
      padding-top: 50px;
      margin-left: 200px;
      margin-right: 200px;
    }
    .intro2 {
      font-size: 20px;
      justify-content: center;
    }

    .navbar-nav {
      text-align: right;
    }
    ._in {
      margin-left: 550px;
    }
  </style>
  <body style="background-color: #f4c2c2">
    <nav class="navbar navbar-expand-lg navbar-dark bg-danger">
      <a class="navbar-brand" href="#"
        >ECG Arrhythmia classification using CNN</a
      >
      <button
        class="navbar-toggler"
```

B. info.html

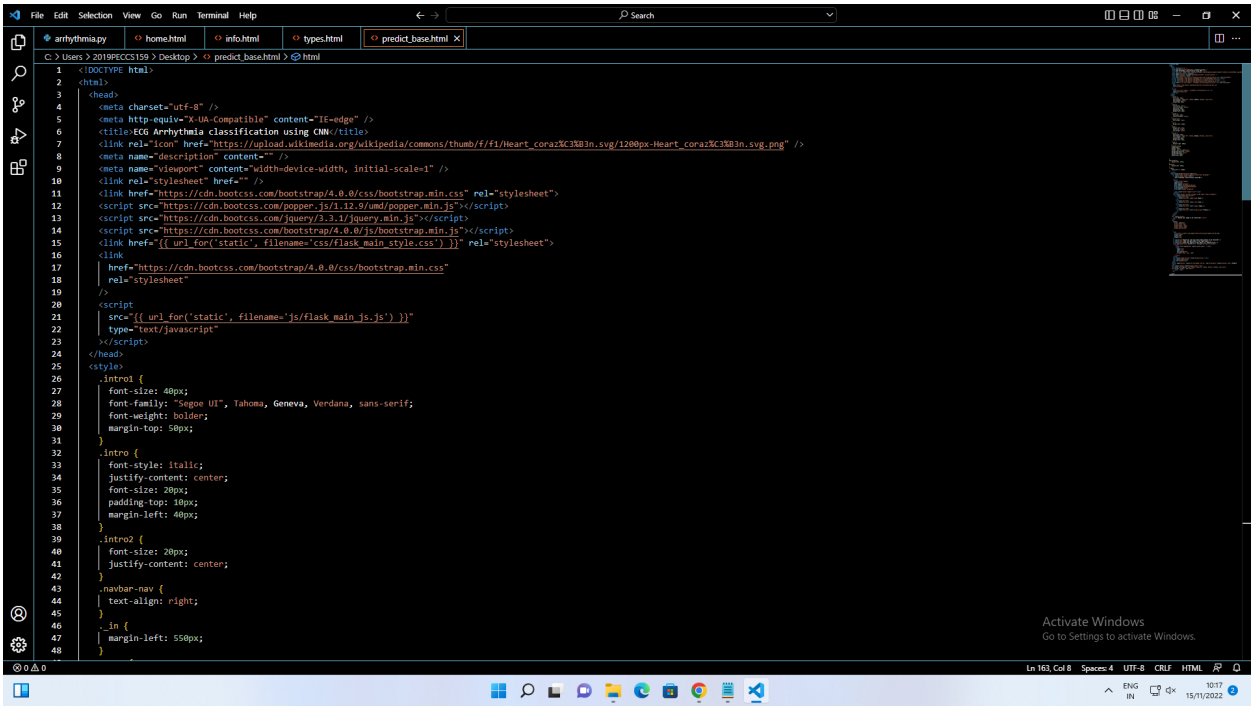


```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <title>ECG Arrhythmia classification using CNN</title>
    <link rel="icon" href="https://upload.wikimedia.org/wikipedia/commons/thumb/f/f1/Heart_coraz%C3%B3n.svg/1200px-Heart_coraz%C3%B3n.svg.png" />
    <meta name="description" content="" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="" />
    <link
      href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
      rel="stylesheet"
    />
  </head>
  <style>
    .intro1 {
      font-size: 40px;
      font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
      font-weight: bolder;
      margin-top: 50px;
    }
    .intro {
      font-style: italic;
      justify-content: center;
      font-size: 20px;
      padding-top: 50px;
      margin-left: 40px;
    }
    .intro2 {
      font-size: 20px;
      justify-content: center;
    }

    .navbar-nav {
      text-align: right;
    }
    ._in {
      margin-left: 550px;
    }
    .para {
      margin-left: 15px;
    }
  </style>
  <body style="background-color: #f4c2c2">
    <nav class="navbar navbar-expand-lg navbar-dark bg-danger">
      <a class="navbar-brand" href="#"
        >ECG Arrhythmia classification using CNN</a
```

C. types.html

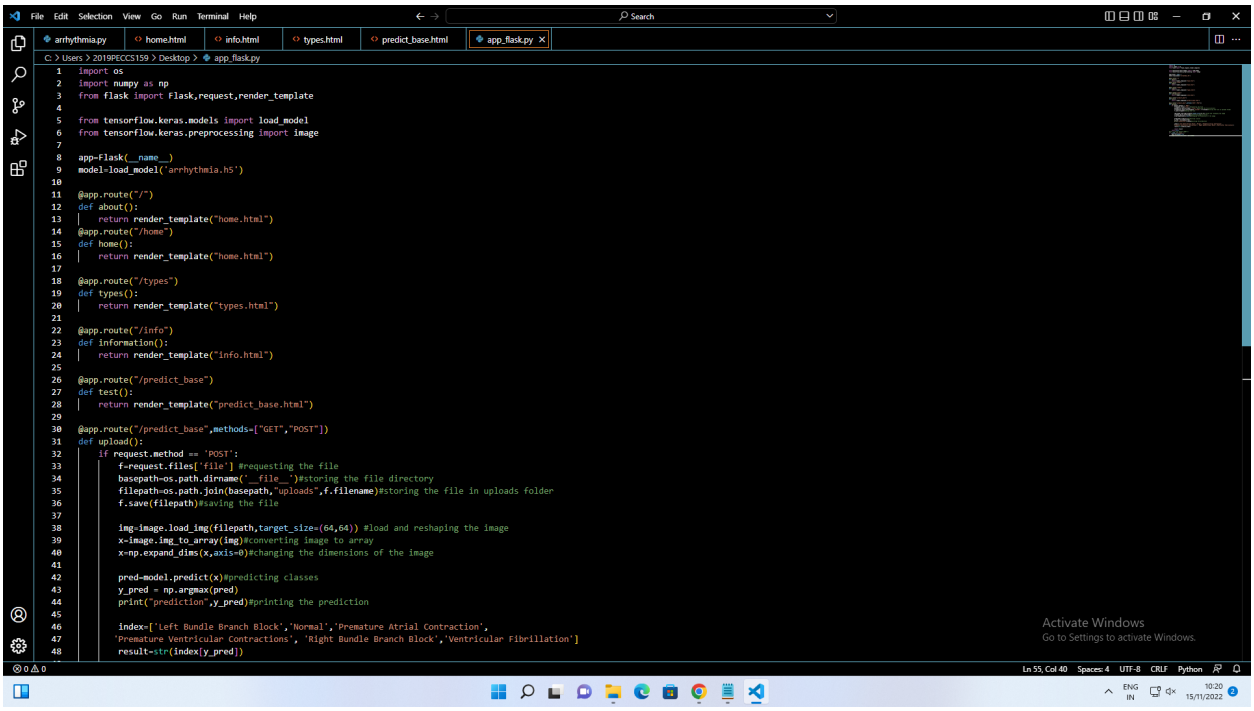


D. predict_base.html



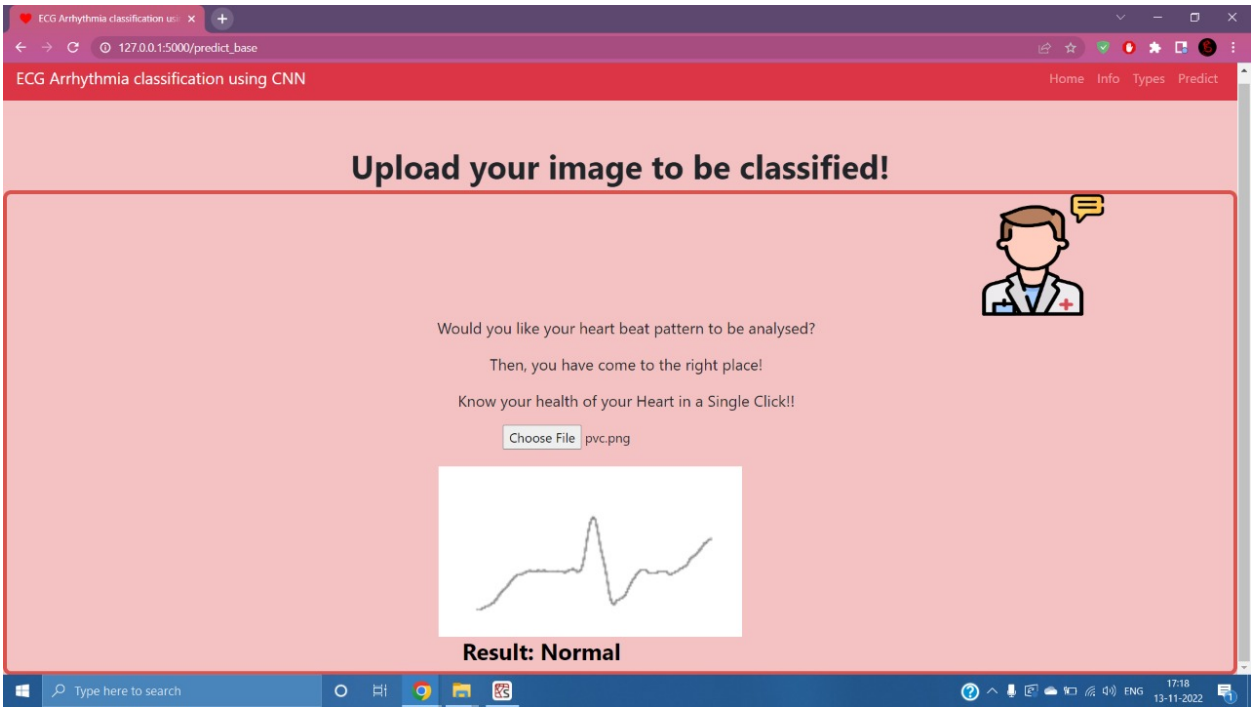
SPRINT 3

app_flask.py



```python
import os
import numpy as np
from flask import Flask,request,render_template

from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image

app=Flask(__name__)
model=load_model('arrhythmia.h5')

@app.route("/")
def about():
    return render_template("home.html")
@app.route("/home")
def home():
    return render_template("home.html")

@app.route("/types")
def types():
    return render_template("types.html")

@app.route("/info")
def information():
    return render_template("info.html")

@app.route("/predict_base")
def test():
    return render_template("predict_base.html")

@app.route("/predict_base",methods=["GET","POST"])
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

        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

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

SPRINT4



6.2 Sprint Delivery Schedule

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint Start Date	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	2 Days	04 Nov 2022	06 Nov 2022	20	06 Nov 2022
Sprint-2	20	2 Days	06 Nov 2022	08 Nov 2022	40	08 Nov 2022
Sprint-3	20	2 Days	08 Nov 2022	10 Nov 2022	60	10 Nov 2022
Sprint-4	20	2 Days	10 Nov 2022	12 Nov 2022	80	12 Nov 2022

6.3 Reports from JIRA

7. CODING & SOLUTIONING

7.1 Feature 1

1. Classifies arrhythmia only with an ECG image within seconds

Source Code

```
import os
import numpy as np
from flask import Flask,request,render_template
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
app=Flask(__name__)
model=load_model('arrhythmia.h5')
@app.route("/")
def about():
        return render_template("home.html")
@app.route("/home")
def home():
        return render_template("home.html")
@app.route("/types")
def types():
        return render_template("types.html")
@app.route("/info")
def information():
        return render_template("info.html")
@app.route("/predict_base")
def test():
        return render_template("predict_base.html")
@app.route("/predict_base",methods=["GET","POST"])
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 file in uploads folder
            f.save(filepath)#saving the file
            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
            index=['Left Bundle Branch Block','Normal','Premature Atrial Contraction',
            'Premature Ventricular Contractions', 'Right Bundle Branch Block','Ventricular
             Fibrillation']
             result=str(index[y_pred])
        return result
    return None
if __name__=="__main__":
        app.run(debug=True)
```

2. Easy user interface

Source code

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<title>ECG Arrhythmia classification using CNN</title>
<linkrel="icon"href="https://upload.wikimedia.org/wikipedia/commons/thumb/f/f1/Heart
_coraz%C3%B3n.svg/1200px-Heart_coraz%C3%B3n.svg.png" />
<meta name="description" content="" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<link rel="stylesheet" href="" />
  <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
    <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
```

```
 <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
 <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
 <link href="{{ url_for('static', filename='css/flask_main_style.css') }}" rel="stylesheet">
 <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
 rel="stylesheet"/>
 <script src="{{ url_for('static', filename='js/flask_main_js.js') }}"type="text/javascript"
></script>
</head>
<style>
.intro1 {
font-size: 40px;
font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
font-weight: bolder;
margin-top: 50px;
}
.intro {
 font-style: italic;
justify-content: center;
font-size: 20px;
padding-top: 10px;
margin-left: 40px;
}
.intro2 {
font-size: 20px;
justify-content: center;
}
.navbar-nav {
text-align: right;
}
._in {
margin-left: 550px;
}
```

```
.para {
margin-left: 15px;
font-size: larger;
text-align: center;
}
.intro3 {
font-size: 25px;
font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
font-weight: bold;
margin-top: 50px;
margin-left: 45px;
}
.right {
margin-right: 600px;
}
#imagePreview{
width:375px;
height: 211px;
border: 1px solid #f4c2c2;
background-position: center;
background-size: cover;
margin-left:530px;
margin-top: 20px;
}
#btn-predict{
margin-left: 670px;
}
#result{
margin-left: 560px;
}
.right{
margin:0 0 0  1200px;
```

```
}
</style>
<body style="background-color: #f4c2c2">
<nav class="navbar navbar-expand-lg navbar-dark bg-danger">
<a class="navbar-brand" href="#">ECG Arrhythmia classification using CNN/>
<button
class="navbar-toggler"
type="button"
data-toggle="collapse"
data-target="#navbarNavAltMarkup"
aria-controls="navbarNavAltMarkup"
aria-expanded="false"
aria-label="Toggle navigation"
>
<span class="navbar-toggler-icon"></span>
</button>
<div class="navbar-collapse collapse w-100 order-3 dual-collapse2">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link" href="/home">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="/info">Info</a>
</li>
<li class="nav-item">
<a class="nav-link" href="/types">Types</a>
</li>
<li class="nav-item">
<a class="nav-link" href="/predict_base">Predict</a>
</li>
</ul>
</div>
```

```
</nav>
<div class="intro1">
<center>Upload your image to be classified!</center>
</div>
<div
style="
border: #d9534f;
border-width: 5px;
border-style: solid;
border-radius: 10px;
height:max-content;">
<img
src="https://tikotra.com/images/Tikotra-Vaccination-page-icons-02.png"
class="right"
width="150"
height="150"
/>
<p class="para">Would you like your heart beat pattern to be analysed?</p>
<p class="para">Then, you have come to the right place!</p>
<p class="para">Know your health of your Heart in a Single Click!!</p>
<form id="upload-file" method="post" enctype="multipart/form-data">
<center>
<label for="imageUpload" class="upload-label"> </label>
<input
type="file"
name="file"
id="imageUpload"
accept=".png, .jpg, .jpeg"
/>
</center>
</form>
<div class="image-section" style="display:none;"></div>
```

```
        <div class="img-preview">
        <div id="imagePreview">
        </div>
        <button type="button" class="btn btn-danger btn-lg " id="btn-predict" style="display: none;">Predict
    </div>
    <div class="loader" style="display:none;"></div>
    <h3 style="color:Black; font-family: Segoe UI, Tahoma, Geneva, Verdana, sans-serif;
    font-weight: bolder;" id="result">
       <span> </span>
    </h3>
  </body>
</html>
```

3. Provides more information about arrhythmia and it's types

Source Code

```
        <!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <title>ECG Arrhythmia classification using CNN</title>
                                                    <link rel="icon"
href="https://upload.wikimedia.org/wikipedia/commons/thumb/f/f1/Heart_coraz%C3%B3n.svg/1200px-Heart_coraz%C3%B3n.svg.png" />
    <meta name="description" content="" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="" />
    <link
      href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
      rel="stylesheet"
    />
  </head>
```

```
<style>
  .intro1 {
    font-size: 40px;
    font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
    font-weight: bolder;
    margin-top: 50px;
  }
  .intro {
    font-style: italic;
    justify-content: center;
    font-size: 20px;
    padding-top: 10px;
    margin-left: 40px;
  }
  .intro2 {
    font-size: 20px;
    justify-content: center;
  }

  .navbar-nav {
    text-align: right;
  }
  ._in {
    margin-left: 550px;
  }
  .para {
    margin-left: 15px;
  }
  .intro3 {
    font-size: 25px;
    font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
    font-weight: bold;
```

```
    margin-top: 50px;
    margin-left: 45px;
  }
</style>
<body style="background-color: #f4c2c2">
  <nav class="navbar navbar-expand-lg navbar-dark bg-danger">
    <a class="navbar-brand" href="#"
      >ECG Arrhythmia classification using CNN</a
    >
    <button
      class="navbar-toggler"
      type="button"
      data-toggle="collapse"
      data-target="#navbarNavAltMarkup"
      aria-controls="navbarNavAltMarkup"
      aria-expanded="false"
      aria-label="Toggle navigation"
    >
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="navbar-collapse collapse w-100 order-3 dual-collapse2">
      <ul class="navbar-nav ml-auto">
        <li class="nav-item">
          <a class="nav-link" href="/home">Home</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/info">Info</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/types">Types</a>
        </li>
        <li class="nav-item">
```

```
        <a class="nav-link" href="/predict_base">Predict</a>
      </li>
    </ul>
  </div>
</nav>
<div
  style="
    border: #d9534f;
    border-width: 5px;
    border-style: solid;
    border-radius: 10px;
    height: max-content;
    margin-top: 30px;
    margin-left: 10px;
    margin-right: 10px;
    margin-bottom: 10px;
  "
>
  <div class="intro1">
    <center>Types of Arrhythmia</center>
  </div>
  <div class="intro">
    <p class="para">
      Arrhythmias are usually categorized based on the speed of heart rate.
      The six main categories would be-
    </p>
    <ol class="_in">
      <li><strong>Normal</strong></li>
      <li><strong>Left Bundle Branch Block</strong></li>
      <li><strong>Right Bundle Branch Block</strong></li>
      <li><strong>Ventricular Fibrillation</strong></li>
      <li><strong>Premature Atrial Contraction</strong></li>
```

```
    <li><strong>Premature Ventricular Contractions</strong></li>
  </ol>
</div>
<div class="intro3" id="types">
  <section>1.Left Bundle Branch Block:</section>
</div>
<div class="intro">
  <p class="para">
    A delay or blockage of electrical impulses to the left side of the
    heart. Left bundle branch block sometimes makes it harder for the
    heart to pump blood efficiently through the circulatory system. Most
    people don't have symptoms. If symptoms occur, they include fainting
    or a slow heart rate. If there's an underlying condition, such as
    heart disease, that condition needs treatment. In patients with heart
    failure, a pacemaker can also relieve symptoms as well as prevent
    death.
  </p>
</div>
<div class="intro3" id="types">
  <section>2.Premature Atrial Contraction:</section>
</div>
<div class="intro">
  <p class="para">
    Premature atrial contractions (PACs) are extra heartbeats that start
    in the upper chambers of your heart. When the premature, or early,
    signal tells the heart to contract, there may not be much blood in the
    heart at that moment. That means there's not much blood to pump out.
  </p>
</div>
<div class="intro3" id="types">
  <section>3.Premature Ventricular Contractions:</section>
</div>
```

    Premature ventricular contractions (PVCs) are extra heartbeats that
    begin in one of the heart's two lower pumping chambers (ventricles).
    These extra beats disrupt the regular heart rhythm, sometimes causing
    a sensation of a fluttering or a skipped beat in the chest. In the
    vast majority of cases, PVCs have no known cause and occur
    spontaneously. Common known etiologies include excess caffeine
    consumption, excess catecholamines, high levels of anxiety, and
    electrolyte abnormalities.

  <section>4.Right Bundle Branch Block:</section>

    Right bundle branch block is a problem with your right bundle branch
    that keeps your heart’s electrical signal from moving at the same time
    as the left bundle branch. Instead of moving together on the left and
    right sides, the signal on the right side is running behind. This
    creates an irregular heartbeat. Usually, the problem isn’t serious.

  <section>5.Ventricular Fibrillation:</section>

    Ventricular fibrillation is a type of irregular heart rhythm
    (arrhythmia). During ventricular fibrillation, the lower heart
    chambers contract in a very rapid and uncoordinated manner. As a

result, the heart doesn't pump blood to the rest of the body.

```
        </p>
      </div>
    </div>
  </body>
</html>
```

8.TESTING

8.1. Performance Testing

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

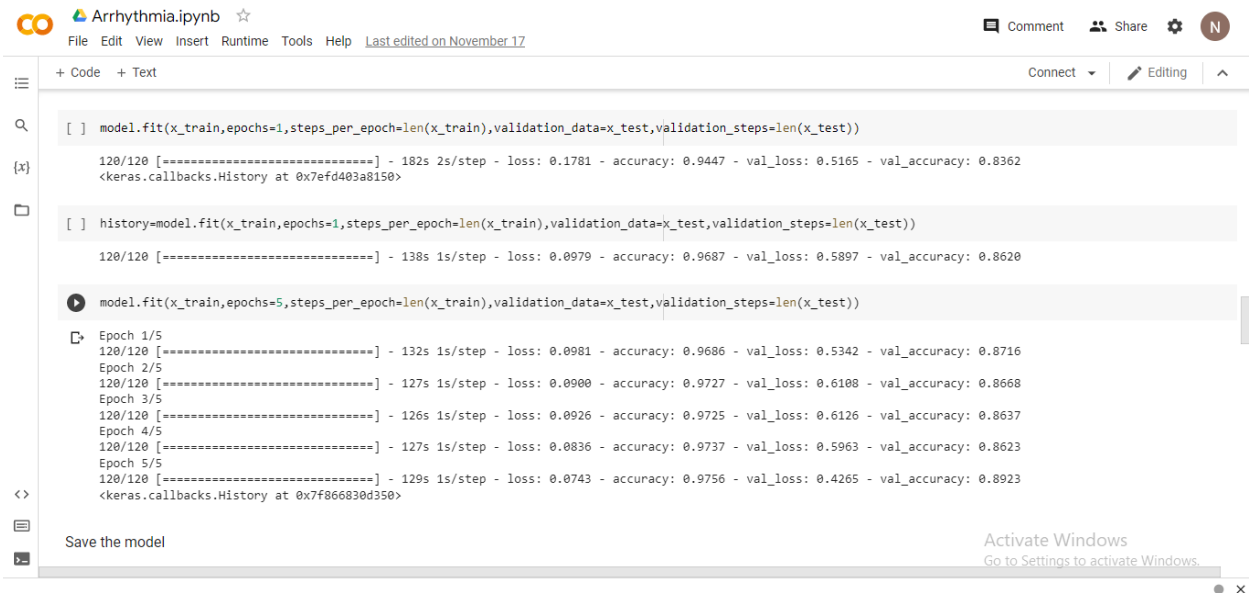
S.No.	Parameter	Values	Screenshot
1.	Model Summary	-	https://drive.google.com/file/d/1jmRk6-t2BYYywLNwihLPtM-BuX13BAZP/view?usp=share_link
2.	Accuracy	Training Accuracy - Validation Accuracy -	https://drive.google.com/file/d/1KvV1YlEwYW4sx46vqdYmuaTTCYHRGWIS/view?usp=share_link

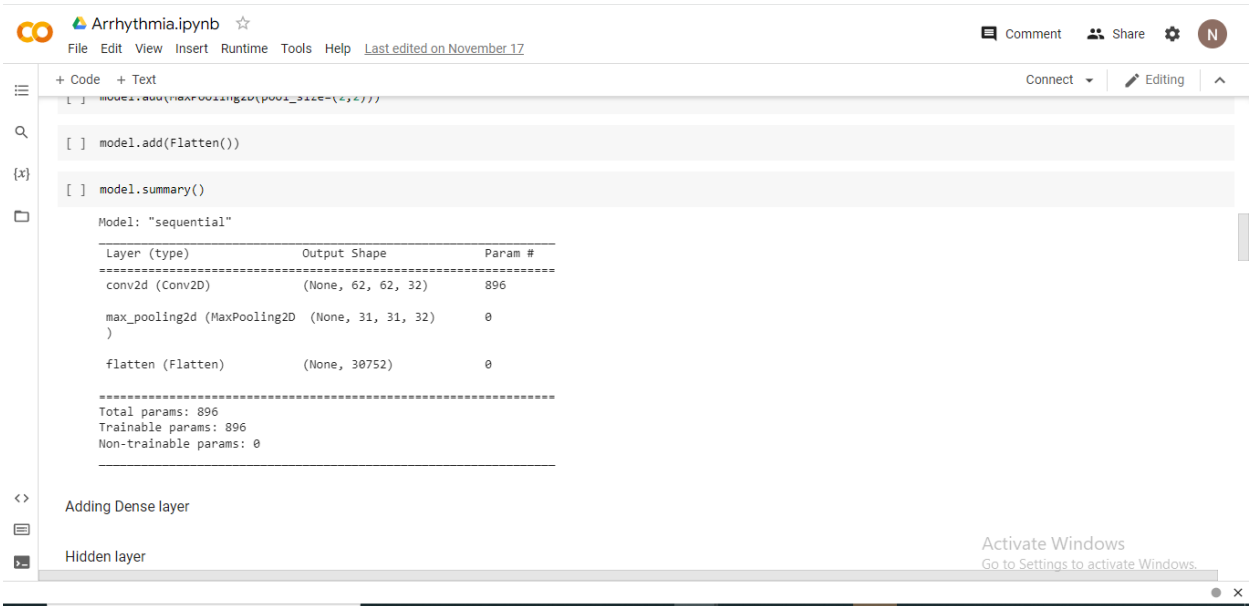
8.2.User Acceptance Testing

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	19
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	10	2	4	10	26
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	4	1	1	6
Totals	22	15	10	30	77

9.RESULTS

9.1.Performance Metrics





10.ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Provides accurate results and detailed information required by the users or patients.
- Users or customers can easily use the app because of its user-friendly interface and simplicity.
- Can be used by anyone at any time.
- As this application can be very useful for the earlier and fast classification of arrhythmia it we be used by many patients suffering from it.
- Experts guidance is not required when we have a app that can be used by anyone.
- Data of the patient will be securely stored and maintained for future purposes.

11.Conclusion

Arrhythmia is a severe CVD that can be predicted via ECG segment processing. Arrhythmia must be accurately diagnosed and prevented early to reduce cardiac disease. Our proposed system model met the study's primary goal of assisting doctors in swiftly determining the kind of ECG or verifying their diagnostics in a medical context while maintaining a high level of precision and cost. In this work, a CNN-Bi-LSTM model is proposed to categorise five categories of ECG fragments to construct an effective and resilient autonomous computer-aided diagnosis system. The developed network achieved maximum accuracies of 100%, 98.0%, and 98.0% of training, validation, and testing using MIT-BIH data set. In comparison, the St-Petersburg data set achieved 98.0%, 95.0%, and 95.0% accuracies of training, validation, and testing in identifying arrhythmia.

This research showed many advantages, including its ability to help clinicians reliably make ECG recording-related clinical decisions. Moreover, it was intended to be as simple as possible while delivering the most significant performance. The described method is straightforward for health professionals and does not involve signal modification or feature extraction. Additionally, this research focused only on one kind of CVD, namely, arrhythmia, whereas the manifestations of cardiac disease are often complex and varied. As a result, more types of ECG data will need to be added to broaden the scope of the planned network.

12.FUTURE SCOPE

1. To enable online doctor consultation
2. To use database and store patients data

13.APPENDIX

Source Code

1) Model Building Code

```
from google.colab import drive

drive.mount('/content/drive')

cd /content/drive/MyDrive/Project Development Phase

#Import the ImageDataGenerator library

from tensorflow.keras.preprocessing.image import ImageDataGenerator

#Configure ImageDataGenerator Class

train_datagen=ImageDataGenerator(rescale=1./255,zoom_range=0.2,vertical_flip=True,horizont

al_flip=True)

test_data=ImageDataGenerator(rescale=1./255)

#Apply ImageDataGenerator functionality to trainset and testset

x_train=train_datagen.flow_from_directory(r"/content/drive/MyDrive/ProjectDevelopment

Phase/data/train",target_size=(64,64),class_mode="categorical",batch_size=128)


x_test=test_data.flow_from_directory(r"/content/drive/MyDrive/ProjectDevelopment

Phase/data/test",target_size=(64,64),class_mode="categorical",batch_size=128)

x_train.class_indices

#MODEL BUILDING

#Import the libraries

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense,Convolution2D,MaxPooling2D,Flatten

#Initialize the model
```

```
model=Sequential()
#Adding CNN layers
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()
#Adding Dense layer
#Hidden layer
model.add(Dense(500,activation="relu"))
model.add(Dense(500,activation="relu"))
#Output layer
model.add(Dense(6,activation="softmax"))
#Configure the learning process
model.compile(loss="categorical_crossentropy",optimizer="adam",metrics=['accuracy'])
len(x_train)
#Train the model
model.fit(x_train,epochs=5,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test))
model.fit(x_train,epochs=5,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test))
model.fit(x_train,epochs=5,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test))
model.fit(x_train,epochs=5,steps_per_epoch=len(x_train),validation_data=x_test,validation_steps=len(x_test))
```

#Save the model

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

#Testing the model

```
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image

img=image.load_img("/content/drive/MyDrive/Project Development Phase/data/test/Premature Ventricular Contractions/VEBfig_11.png",target_size=(64,64))

x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
pred=model.predict(x)

index=['Left Bundle Branch Block', 'Normal', 'Premature Atrial Contraction', 'Premature Ventricular Contractions', 'Right Bundle Branch Block', 'Ventricular Fibrillation']

index[np.argmax(pred)]
```

2) HTML Code

a) home.html

```
<!DOCTYPE html>
<html>
  <head>
```

```
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <title>ECG Arrhythmia classification using CNN</title>
                                                                    <link rel="icon"
href="https://upload.wikimedia.org/wikipedia/commons/thumb/f/f1/Heart_coraz%C3%B3n.svg/
1200px-Heart_coraz%C3%B3n.svg.png" />
    <meta name="description" content="" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="" />
    <link
      href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
      rel="stylesheet"
    />
  </head>
  <style>
    .intro1 {
      font-size: 40px;
      font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
      font-weight: bolder;
    }
    .intro {
      font-style: italic;
      justify-content: center;
      font-size: 20px;
```

```
    padding-top: 50px;
    margin-left: 200px;
    margin-right: 200px;
   }
   .intro2 {
    font-size: 20px;
    justify-content: center;
   }

   .navbar-nav {
    text-align: right;
   }
   ._in {
    margin-left: 550px;
   }
</style>
<body style="background-color: #f4c2c2">
  <nav class="navbar navbar-expand-lg navbar-dark bg-danger">
    <a class="navbar-brand" href="#"
      >ECG Arrhythmia classification using CNN</a
    >
    <button
      class="navbar-toggler"
      type="button"
```

```
  data-toggle="collapse"
  data-target="#navbarNavAltMarkup"
  aria-controls="navbarNavAltMarkup"
  aria-expanded="false"
  aria-label="Toggle navigation"
>
  <span class="navbar-toggler-icon"></span>
</button>
<div class="navbar-collapse collapse w-100 order-3 dual-collapse2">
  <ul class="navbar-nav ml-auto">
    <li class="nav-item">
      <a class="nav-link" href="/home">Home</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="/info">Info</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="/types">Types</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="/predict_base">Predict</a>
    </li>
  </ul>
</div>
```

```
</nav>
<div
  style="
    border: #d9534f;
    border-width: 5px;
    border-style: solid;
    border-radius: 10px;
    height: max-content;
    margin-top: 30px;
    margin-left: 10px;
    margin-right: 10px;
  "
>
  <div class="intro1">
    <center>Arrhythmia classification using CNN</center>
  </div>
  <div>
    <p class="intro">
      ‘Hear Arrhythmia’ is the condition of irregular heart rhythms. There
      are variations in the heartbeat patterns. When electrical signals that
      were meant to coordinate with the heartbeats falter, this condition
      takes place. The heart could beat faster, or slower, or any other form
      of irregularity is usually noticeable. Some basic fluttery feeling is
      harmless. It may speed up during active periods of the body and slow
```

down during relaxing periods. The level of symptoms is important to look for because some of them can even be life-threatening. There might not be obvious signs of Arrhythmia. One might need a medical professional to figure it out, but the subtle details to look for are;

```
</p>
</div>
<div class="intro2">
<img
src="https://uxwing.com/wp-content/themes/uxwing/download/relationship-love/heart-care-icon.png"
alt=""
height="200"
width="200"
align="right"
/>
<ol class="_in">
<li>A fluttering feeling in the chest</li>
<li>The feeling of heartbeats slowing down or speeding up</li>
<li>Breathlessness</li>
<li>Pain in the chest area</li>
<li>Heavy sweating</li>
<li>Dizziness</li>
<li>Fatigue</li>
<li>Lightheadedness or fainting</li>
```

```
          <li>Anxiety</li>
        </ol>
        <!-- <img
        class="heart"
              src="https://www.kindpng.com/picc/m/164-1645384_heart-pulse-heart-medical-health-doctor-medicine-football.png"
        height="200"
        width="200 "
       /> -->
      </div>
    </div>
  </body>
</html>
```

b) info.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <title>ECG Arrhythmia classification using CNN</title>
                                                                <link                rel="icon"
```

```
href="https://upload.wikimedia.org/wikipedia/commons/thumb/f/f1/Heart_coraz%C3%B3n.svg/
1200px-Heart_coraz%C3%B3n.svg.png" />
    <meta name="description" content="" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <link rel="stylesheet" href="" />
    <link
      href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
      rel="stylesheet"
    />
  </head>
  <style>
    .intro1 {
      font-size: 40px;
      font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
      font-weight: bolder;
      margin-top: 50px;
    }
    .intro {
      font-style: italic;
      justify-content: center;
      font-size: 20px;
      padding-top: 50px;
      margin-left: 40px;
    }
```

```
  .intro2 {
    font-size: 20px;
    justify-content: center;
  }

  .navbar-nav {
    text-align: right;
  }
  ._in {
    margin-left: 550px;
  }
  .para {
    margin-left: 15px;
  }
</style>
<body style="background-color: #f4c2c2">
  <nav class="navbar navbar-expand-lg navbar-dark bg-danger">
    <a class="navbar-brand" href="#"
      >ECG Arrhythmia classification using CNN</a
    >
    <button
      class="navbar-toggler"
      type="button"
      data-toggle="collapse"
```

```
    data-target="#navbarNavAltMarkup"
    aria-controls="navbarNavAltMarkup"
    aria-expanded="false"
    aria-label="Toggle navigation"
  >
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="navbar-collapse collapse w-100 order-3 dual-collapse2">
    <ul class="navbar-nav ml-auto">
      <li class="nav-item">
        <a class="nav-link" href="/home">Home</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/info">Info</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/types">Types</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/predict_base">Predict</a>
      </li>
    </ul>
  </div>
</nav>
```

```
<div
  style="
    border: #d9534f;
    border-width: 5px;
    border-style: solid;
    border-radius: 10px;
    height: max-content;
    margin-top: 30px;
    margin-left: 10px;
    margin-right: 10px;
    margin-bottom: 10px;
  "
>
  <div class="intro1">
    <center>Diagnosis of Arrhythmia</center>
    <img
      src="http://cdn.onlinewebfonts.com/svg/img_573539.png"
      alt=""
      height="100"
      width="100"
      align="right"
    />
  </div>
  <div class="intro">
```


Electrocardiograms (ECG): Can detect electrical activity of the heart.

Echocardiograms: Uses sound waves to produce images of the heart.

Implantable loop recorder:It can be implanted under the skin around the heart, to record its status.

 recorder:It’s like a wearable ECG. It is supposed to be alerted when you have symptoms.

Holter Monitor:It’s another mobile ECG device which is temporarily used to record the activity of the heart.

<p class="para">
Other tests that can be used are, Stress test, where you are made to

```
    do a physical activity and the activity of your heart is recorded.
    Tilt-table test is used for fainting cases, where you lie flat on the
    table while recording your heart status. EP testing and mapping can
    also be used where tubes with electrode ends are used inside your body
    to find details.
  </p>
</div>
<div class="intro1">
  <center>
    <i class="fa-solid fa-syringe"></i>Treatment -Medications
  </center>
</div>
<div class="intro">
  <p class="para">
    Drugs like blood thinners are recommended by doctors as per the
    condition of the patient.There are two therapies to treat heart
    issues---
  </p>
  <ol>
    <li>
      <strong>Cardioversion:</strong> Shock is delivered to the heart
      using patches on the chest which coordinates the electrical
      impulses.
    </li>
```

```
    <li>
      <strong>Vagal maneuvers:</strong> Controls the nervous system, which
      inturn slows down the heart beats. Used for faster heart beat
      conditions.
    </li>
  </ol>
</div>
<div class="intro1">
  <center>Surgeries</center>
</div>
<div class="intro">
  <ol>
    <li>
      <strong>Ablation:</strong> Catheters are used in the blood vessels
      connected to the heart.
    </li>
    <li>
      <strong>Pacemaker:</strong> It is a tiny device implanted near the
      collarbone.
    </li>

    <li>
      <strong>Maze procedure:</strong> It involves making multiple pattern
      like incisions of the heart tissue to make it scar tissue which
```

```
    reduces the stray electrical impulses.
   </li>
   <li>
    <strong>ICD:</strong> It is also a device implanted under the skin
    of the collarbone.
   </li>
   <li>
    <strong>Coronary bypass surgery:</strong> It improves the flow of
    blood to your heart.
   </li>
  </ol>
</div>
<div class="intro1">
  <center>Home Remedies</center>
</div>
<div class="intro">
  <ol>
   <li>Eat healthy</li>
   <li>Do not smoke or drink</li>
   <li>Exercise regularly, keep your weight in check</li>
   <li>Keep your blood pressure and cholesterol levels in check</li>
   <li>
    Keep proper habits of medication and doctor visits whenever
    necessary
```

```
        </li>
        <li>Do Yoga, Meditation, and other Relaxation techniques</li>
       </ol>
      </div>
      <div class="intro1">
       <center>Precautions</center>
      </div>
      <div class="intro">
       <ol>
        <li>Have healthy practices</li>
        <li>Keep note of any symptoms you’re having</li>
        <li>Be aware of the conditions</li>
        <li>Have proper doctor visits</li>
        <li>
         Make a list of all the medications and family history of Arrhythmias
        </li>
       </ol>
      </div>
    </div>
  </body>
</html>
```

c) types.html

```
<!DOCTYPE html>
<html>
```

```
<head>
  <meta charset="utf-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <title>ECG Arrhythmia classification using CNN</title>
                                                    <link rel="icon"
href="https://upload.wikimedia.org/wikipedia/commons/thumb/f/f1/Heart_coraz%C3%B3n.svg/
1200px-Heart_coraz%C3%B3n.svg.png" />
  <meta name="description" content="" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <link rel="stylesheet" href="" />
  <link
    href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
    rel="stylesheet"
  />
</head>
<style>
  .intro1 {
    font-size: 40px;
    font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
    font-weight: bolder;
    margin-top: 50px;
  }
  .intro {
    font-style: italic;
```

```
  justify-content: center;
  font-size: 20px;
  padding-top: 10px;
  margin-left: 40px;
}
.intro2 {
  font-size: 20px;
  justify-content: center;
}

.navbar-nav {
  text-align: right;
}
._in {
  margin-left: 550px;
}
.para {
  margin-left: 15px;
}
.intro3 {
  font-size: 25px;
  font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
  font-weight: bold;
  margin-top: 50px;
```

```
    margin-left: 45px;
  }
</style>
<body style="background-color: #f4c2c2">
  <nav class="navbar navbar-expand-lg navbar-dark bg-danger">
    <a class="navbar-brand" href="#"
      >ECG Arrhythmia classification using CNN</a
    >
    <button
      class="navbar-toggler"
      type="button"
      data-toggle="collapse"
      data-target="#navbarNavAltMarkup"
      aria-controls="navbarNavAltMarkup"
      aria-expanded="false"
      aria-label="Toggle navigation"
    >
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="navbar-collapse collapse w-100 order-3 dual-collapse2">
      <ul class="navbar-nav ml-auto">
        <li class="nav-item">
          <a class="nav-link" href="/home">Home</a>
        </li>
```

```
      <li class="nav-item">
        <a class="nav-link" href="/info">Info</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/types">Types</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/predict_base">Predict</a>
      </li>
    </ul>
  </div>
</nav>
<div
  style="
    border: #d9534f;
    border-width: 5px;
    border-style: solid;
    border-radius: 10px;
    height: max-content;
    margin-top: 30px;
    margin-left: 10px;
    margin-right: 10px;
    margin-bottom: 10px;
  "
```

```
>
  <div class="intro1">
    <center>Types of Arrhythmia</center>
  </div>
  <div class="intro">
    <p class="para">
      Arrhythmias are usually categorized based on the speed of heart rate.
      The six main categories would be-
    </p>
    <ol class="_in">
      <li><strong>Normal</strong></li>
      <li><strong>Left Bundle Branch Block</strong></li>
      <li><strong>Right Bundle Branch Block</strong></li>
      <li><strong>Ventricular Fibrillation</strong></li>
      <li><strong>Premature Atrial Contraction</strong></li>
      <li><strong>Premature Ventricular Contractions</strong></li>
    </ol>
  </div>
  <div class="intro3" id="types">
    <section>1.Left Bundle Branch Block:</section>
  </div>
  <div class="intro">
    <p class="para">
      A delay or blockage of electrical impulses to the left side of the
```

heart. Left bundle branch block sometimes makes it harder for the heart to pump blood efficiently through the circulatory system. Most people don't have symptoms. If symptoms occur, they include fainting or a slow heart rate. If there's an underlying condition, such as heart disease, that condition needs treatment. In patients with heart failure, a pacemaker can also relieve symptoms as well as prevent death.

<section>2.Premature Atrial Contraction:</section>

Premature atrial contractions (PACs) are extra heartbeats that start in the upper chambers of your heart. When the premature, or early, signal tells the heart to contract, there may not be much blood in the heart at that moment. That means there's not much blood to pump out.

<section>3.Premature Ventricular Contractions:</section>

```
  <p class="para">
    Premature ventricular contractions (PVCs) are extra heartbeats that
    begin in one of the heart's two lower pumping chambers (ventricles).
    These extra beats disrupt the regular heart rhythm, sometimes causing
    a sensation of a fluttering or a skipped beat in the chest. In the
    vast majority of cases, PVCs have no known cause and occur
    spontaneously. Common known etiologies include excess caffeine
    consumption, excess catecholamines, high levels of anxiety, and
    electrolyte abnormalities.
  </p>
</div>
<div class="intro3" id="types">
  <section>4.Right Bundle Branch Block:</section>
</div>
<div class="intro">
  <p class="para">
    Right bundle branch block is a problem with your right bundle branch
    that keeps your heart’s electrical signal from moving at the same time
    as the left bundle branch. Instead of moving together on the left and
    right sides, the signal on the right side is running behind. This
    creates an irregular heartbeat. Usually, the problem isn’t serious.
  </p>
</div>
<div class="intro3" id="types">
```

```
      <section>5.Ventricular Fibrillation:</section>
    </div>
    <div class="intro">
      <p class="para">
        Ventricular fibrillation is a type of irregular heart rhythm
        (arrhythmia). During ventricular fibrillation, the lower heart
        chambers contract in a very rapid and uncoordinated manner. As a
        result, the heart doesn't pump blood to the rest of the body.
      </p>
    </div>
  </div>
 </body>
</html>
```

d) predict_base.html

```
<!DOCTYPE html>
<html>
 <head>
  <meta charset="utf-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <title>ECG Arrhythmia classification using CNN</title>
                                                    <link rel="icon"
href="https://upload.wikimedia.org/wikipedia/commons/thumb/f/f1/Heart_coraz%C3%B3n.svg/
1200px-Heart_coraz%C3%B3n.svg.png" />
  <meta name="description" content="" />
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <link rel="stylesheet" href="" />
  <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
  <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
  <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
  <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
  <link href="{{ url_for('static', filename='css/flask_main_style.css') }}" rel="stylesheet">
  <link
    href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css"
    rel="stylesheet"
  />
  <script
    src="{{ url_for('static', filename='js/flask_main_js.js') }}"
    type="text/javascript"
  ></script>
</head>
<style>
  .intro1 {
    font-size: 40px;
    font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
    font-weight: bolder;
    margin-top: 50px;
  }
  .intro {
```

```
  font-style: italic;
  justify-content: center;
  font-size: 20px;
  padding-top: 10px;
  margin-left: 40px;
}
.intro2 {
  font-size: 20px;
  justify-content: center;
}
.navbar-nav {
  text-align: right;
}
._in {
  margin-left: 550px;
}
.para {
  margin-left: 15px;
  font-size: larger;
  text-align: center;
}
.intro3 {
  font-size: 25px;
  font-family: "Segoe UI", Tahoma, Geneva, Verdana, sans-serif;
```

```
    font-weight: bold;
    margin-top: 50px;
    margin-left: 45px;
  }
  .right {
    margin-right: 600px;
  }
  #imagePreview{
  width:375px;
  height: 211px;
  border: 1px solid #f4c2c2;
  background-position: center;
  background-size: cover;
  margin-left:530px;
  margin-top: 20px;



}
#btn-predict{
  margin-left: 670px;
}
#result{
  margin-left: 560px;
}
```

```
.right{
    margin:0 0 0  1200px;
}
  </style>
  <body style="background-color: #f4c2c2">
    <nav class="navbar navbar-expand-lg navbar-dark bg-danger">
      <a class="navbar-brand" href="#"
        >ECG Arrhythmia classification using CNN</a
      >
      <button
        class="navbar-toggler"
        type="button"
        data-toggle="collapse"
        data-target="#navbarNavAltMarkup"
        aria-controls="navbarNavAltMarkup"
        aria-expanded="false"
        aria-label="Toggle navigation"
      >
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="navbar-collapse collapse w-100 order-3 dual-collapse2">
        <ul class="navbar-nav ml-auto">
          <li class="nav-item">
            <a class="nav-link" href="/home">Home</a>
```

```
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/info">Info</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/types">Types</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="/predict_base">Predict</a>
      </li>
    </ul>
  </div>
</nav>
<div class="intro1">
  <center>Upload your image to be classified!</center>
</div>
<div
  style="
    border: #d9534f;
    border-width: 5px;
    border-style: solid;
    border-radius: 10px;
    height:max-content;
  "
```

```
>
  <img
    src="https://tikotra.com/images/Tikotra-Vaccination-page-icons-02.png"
    class="right"
    width="150"
    height="150"
  />
  <p class="para">Would you like your heart beat pattern to be analysed?</p>
  <p class="para">Then, you have come to the right place!</p>
  <p class="para">Know your health of your Heart in a Single Click!!</p>
  <form id="upload-file" method="post" enctype="multipart/form-data">
    <center>
      <label for="imageUpload" class="upload-label"> </label>
      <input
        type="file"
        name="file"
        id="imageUpload"
        accept=".png, .jpg, .jpeg"
      />
    </center>
  </form>
  <div class="image-section" style="display:none;"></div>
  <div class="img-preview">
  <div id="imagePreview">
```

```
    </div>
        <button type="button" class="btn btn-danger btn-lg " id="btn-predict" style="display:
none;">Predict
    </div>
    <div class="loader" style="display:none;"></div>
    <h3 style="color:Black; font-family: Segoe UI, Tahoma, Geneva, Verdana, sans-serif;
    font-weight: bolder;" id="result">
        <span> </span>
    </h3>
  </body>
</html>
```

3) Flask Code

```
import os
import numpy as np
from flask import Flask,request,render_template
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
app=Flask(__name__)
model=load_model('arrhythmia.h5')
@app.route("/")
def about():
    return render_template("home.html")
@app.route("/home")
def home():
```

```
    return render_template("home.html")
@app.route("/types")
def types():
    return render_template("types.html")
@app.route("/info")
def information():
    return render_template("info.html")
@app.route("/predict_base")
def test():
    return render_template("predict_base.html")


@app.route("/predict_base",methods=["GET","POST"])
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
        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
```

```
        index=['Left Bundle Branch Block','Normal','Premature Atrial Contraction',
       'Premature Ventricular Contractions', 'Right Bundle Branch Block','Ventricular Fibrillation']
        result=str(index[y_pred])
        return result
    return None
#port = int(os.getenv("PORT"))
if __name__=="__main__":
    app.run(debug=True)
```

GitHub

https://github.com/IBM-EPBL/IBM-Project-2774-1658482560

Project Demo Link

https://github.com/IBM-EPBL/IBM-Project-2774-1658482560/blob/main/Video%20recording/Project%20Demo.mp4