Classification Of Arrhythmia by Using Deep Learning With 2-D ECG Spectral Image Representation

A PROJECT REPORT

Submitted By

KANDULA NAGA REDDAIAH

SREERAMANENI GIRISH

THOPUGUNTA PAVAN KUMAR

YALAGALA VENKATA RAMANA

in partial fulfilment for the award of the degree

of

Bachelor of Engineering

IN

Electronics and Communication Engineering

SAVEETHA ENGINEERING COLLEGE – THANDALAM

CHENNAI-602 105

DECEMBER-2022

SAVEETHA ENGINEERING COLLEGE: CHENNAI-602 105

BONAFIDE CERTIFICATE

Certified that this project report "**Classification Of Arrhythmia By Using Deep Learning With 2-D ECG Spectral Image Representation**" is the bonafide work of "KANDULA NAGA REDDAIAH, SREERAMANENI GIRISH, THOPUGUNTA PAVAN KUMAR, YALAGALA VENKATA RAMANA" who carried out the project work under my supervision.

SIGNATURE

Dr. SRIGITHA S. NATH,

HEAD OF THE DEPARTMENT
ECE DEPARTMENT,
SAVEETHA ENGINEERING COLLEGE

SIGNATURE

Dr. S. KUMARAN

SUPERVISOR
ECE DEPARTMENT,
SAVEETHA ENGINEERING COLLEGE

TABLE OF CONTENTS

TABLE OF CONTENTS

Classification of Arrhythmia by Using Deep Learningwith 2-D ECG Spectral Image Representation

1. INTRODUCTION:

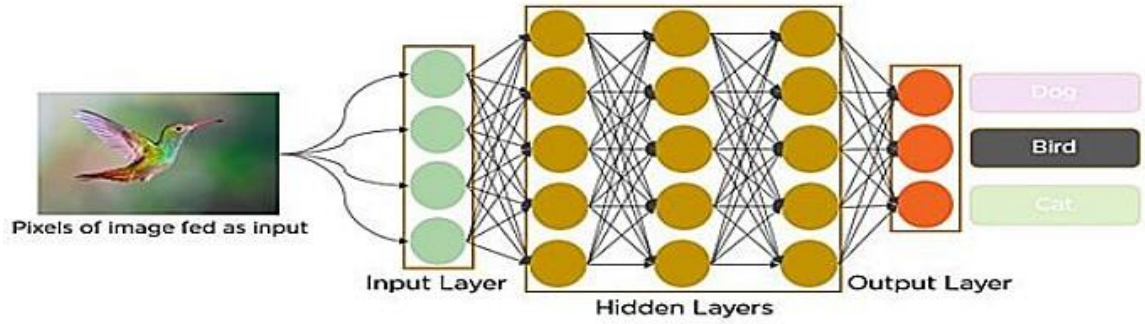
1.1. PROJECT OVERVIEW:

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 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.

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.

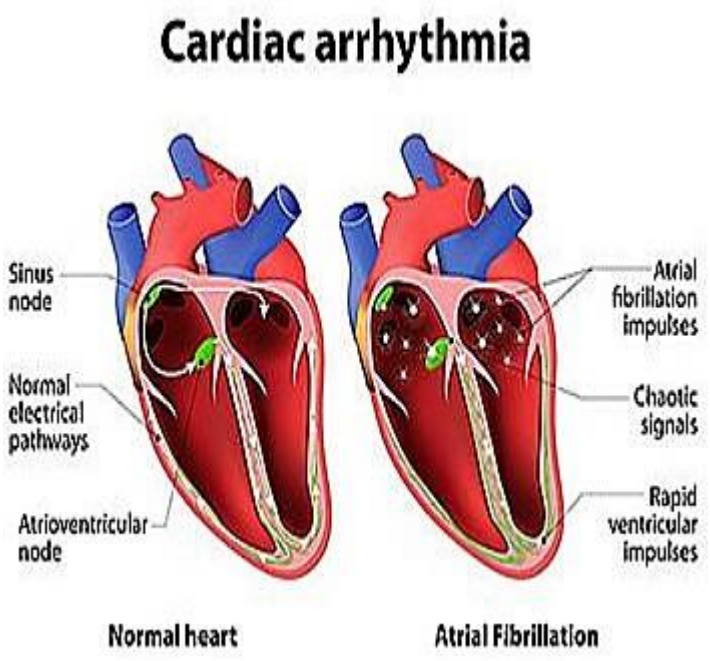


In deep learning, a convolution al neural network (CNN/ConvNet) is a class of deep neural networks, most commonly applied to analyze visual imagery. Now when we think of a neural network we think about matrix multiplications but that is not the case with ConvNet. It uses a special technique called Convolution.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

2.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. REFERENCES:

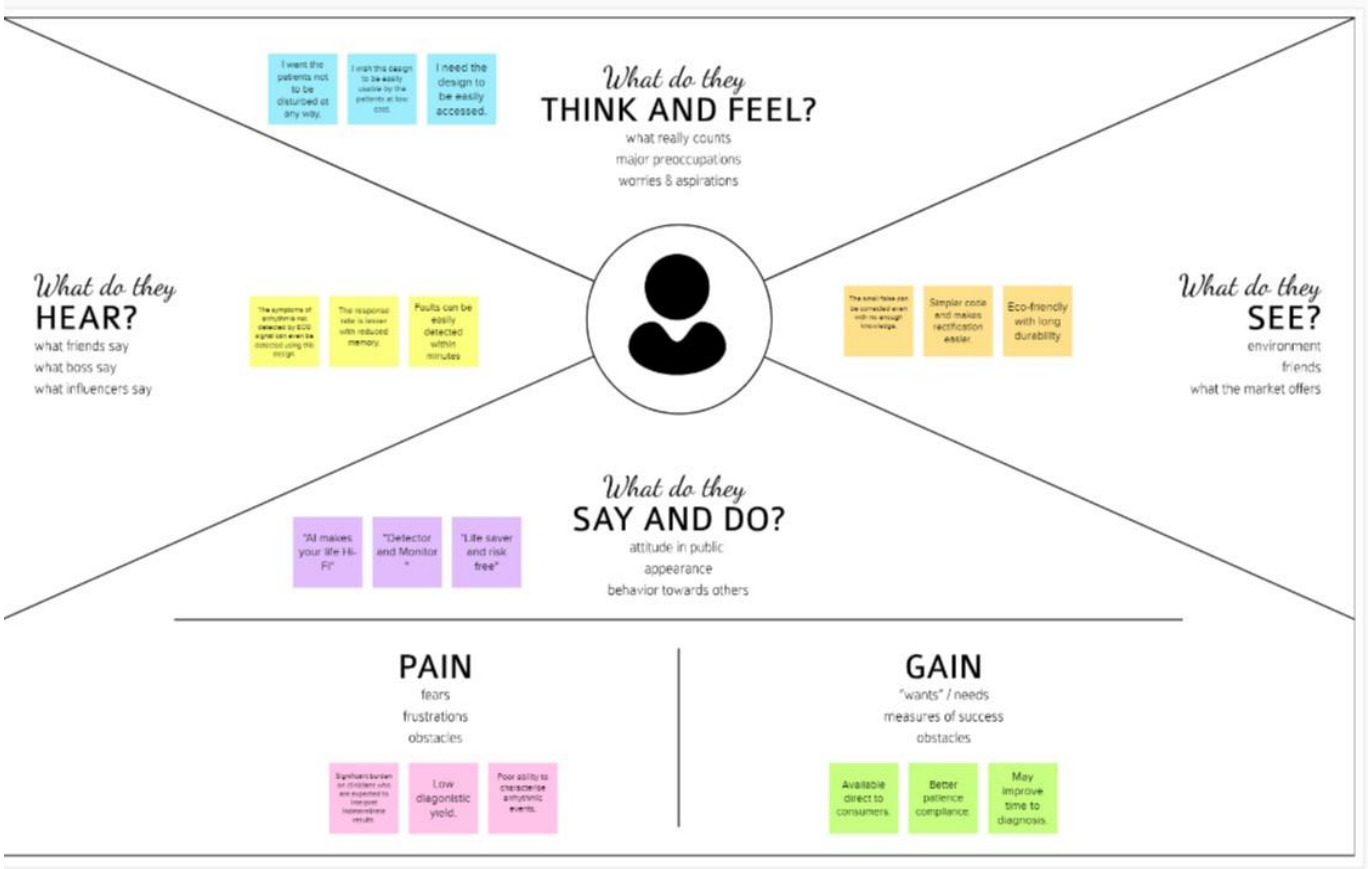
- Amin Ullah Syed Muhammad Anwar, Muhammad Bilal, and Raja Majid Mehmood (2020)
- Bazi, Haikel AlHichri, Naif Alajlan, Farid Melgani, Ronald R Yager (2022)
- Faezeh Nejati Hatamian, Nishant Ravikumar, Sulaiman Vesal(2020)
- Fatma Murat, Ozal Yildirim, Muhammed Talo, Ulas Baran Baloglu, Yakup Demir, U Rajendra Acharya (2020)
- Han Lia, Xinpei Wanga, Changchun Liua, Peng Lib Yu Jiaoa (2021)
- Jagdeep Rahul Lakhan Devi Sharma (2022)
- Kurniawan, I Ketut Eddy Purnama, Mpu Hambyah Syah Bagaskara Aji (2021)
- Ozal Yildirima, MuhammedTaloa, BetulAybUlas BaranBalogluc, GalipAydinbU, RajendraAcharya (2020)
- Rashidah Funke Olanrewaju, S. Noorjannah Ibrahim, Ani Liza Asnawi, Hunain Altaf (2021)
- Rui Hu, Jie Chen, Li Zhou (2022)

2.3. PROBLEM STATEMENT DEFINITION:

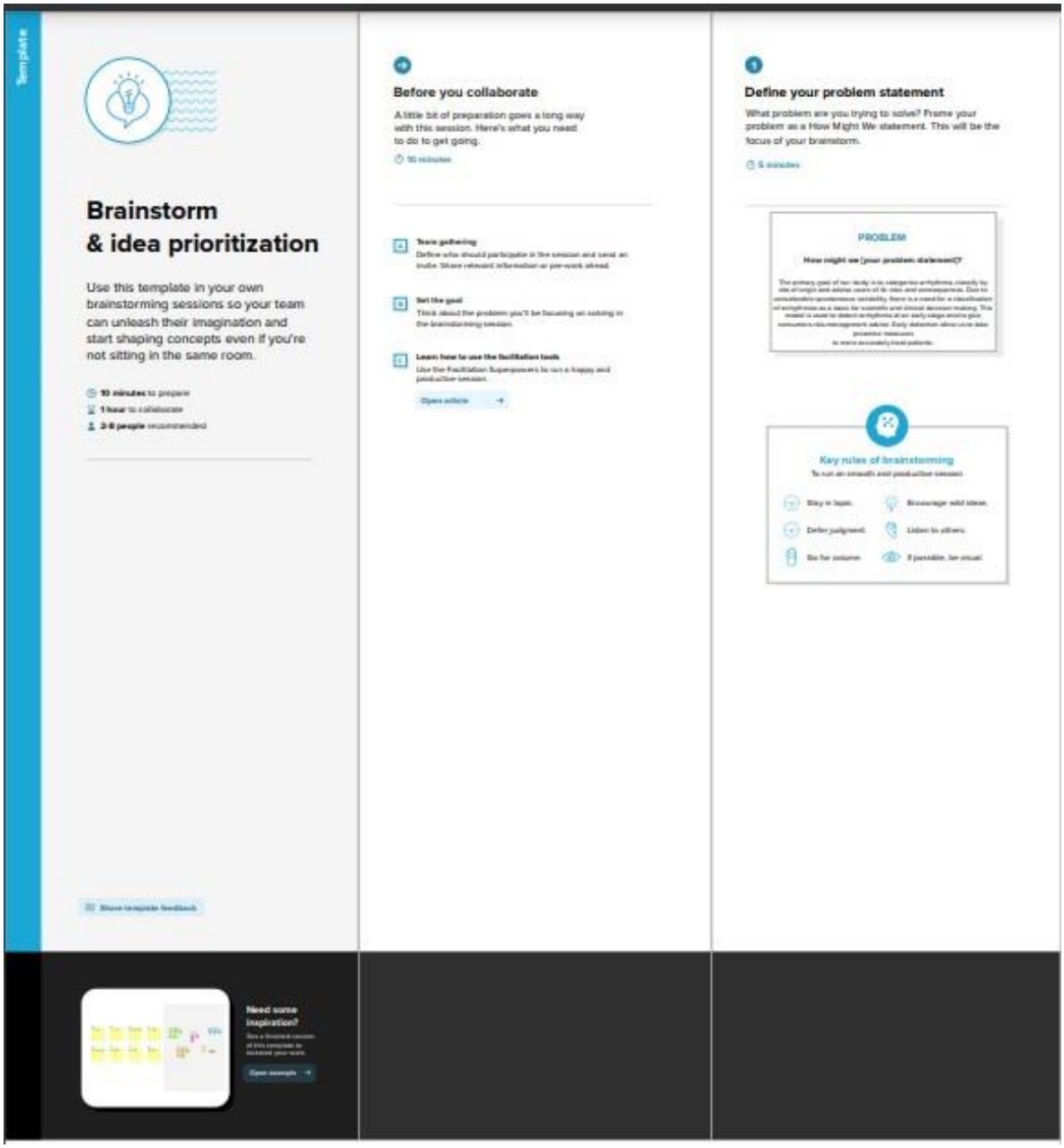
More than four million of people, mostly over age sixty, are suffering from various kinds of arrhythmias that cause discomfort or even sudden cardiac death (SCD).Fast and accurate classification of large set of Electrocardiogram (ECG) beats containing both normal and arrhythmic categories is still a challenging task for the state-of-the art classification algorithms. The ECG signals can capture the heart's rhythmic irregularities, commonly known as arrhythmias. A careful study of ECG signals is crucial for precise diagnoses of patients' acute and chronic heart conditions. A two-dimensional (2-D) convolutional neural network (CNN) model is helpful for the classification of ECG signals into eight classes.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. Using Deep Learning CNN we can enhance the accuracy of diagnosis algorithms in the fusion of medicine and modern machine learning technologies. The proposed CNN-based classification algorithm, using 2-D images, can classify eight kinds of arrhythmia.

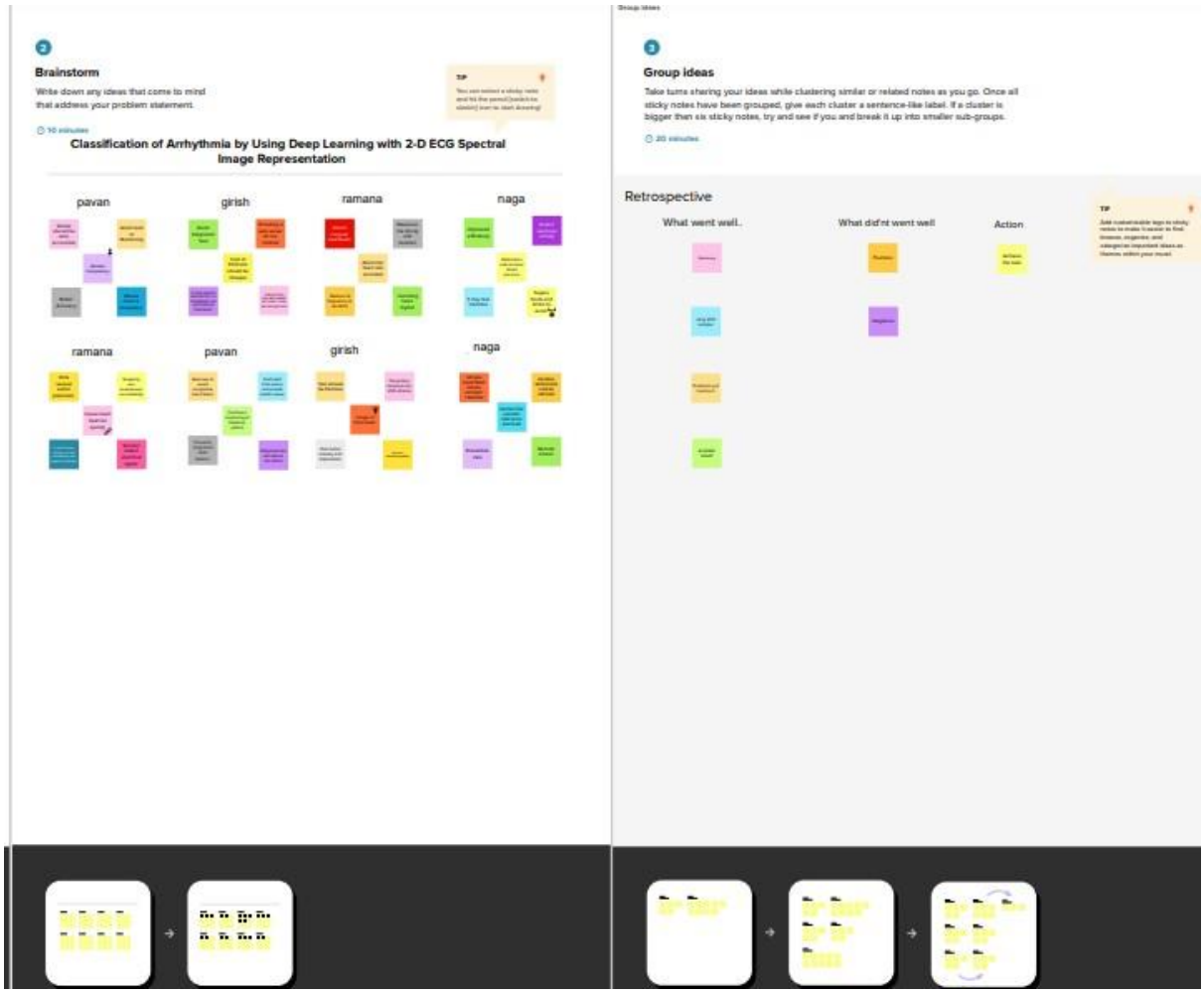
3. IDEATION & PROPOSED SOLUTION:

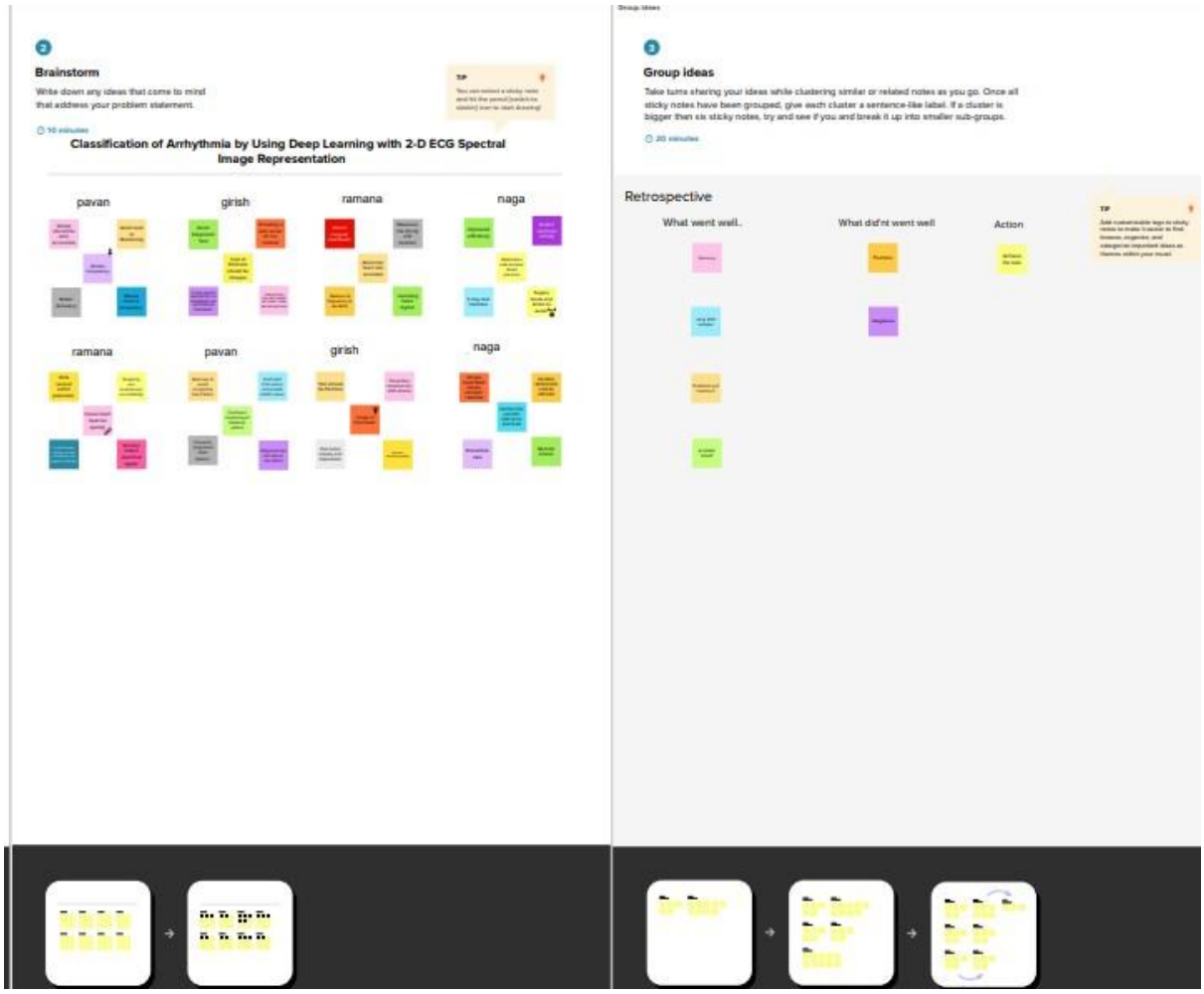
3.1. EMPATHY MAP CANVAS:



3.2. IDEATION & BRAINSTORMING:



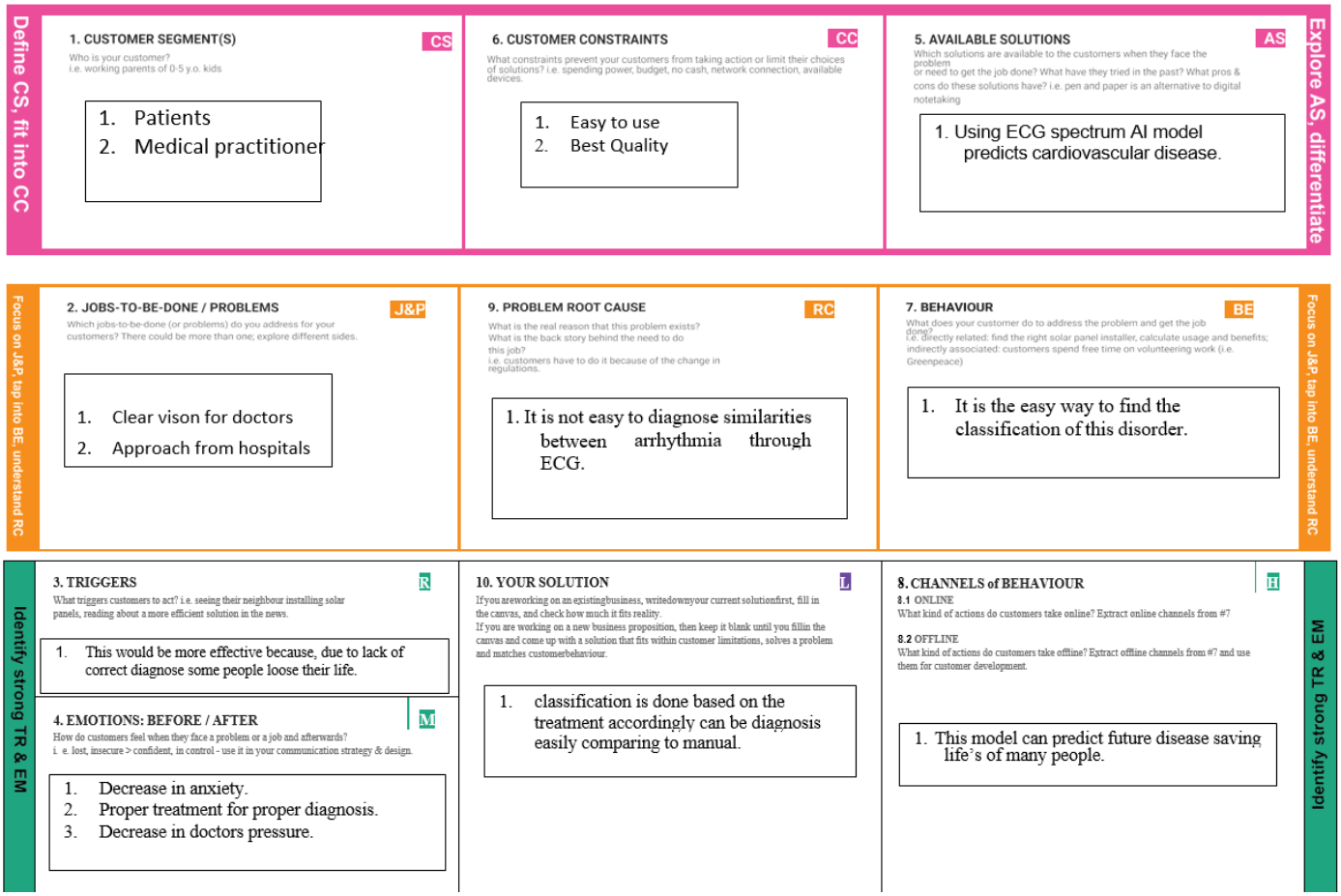

2
Brainstorm
Write down any ideas that come to mind that address your problem statement.
10 minutes
Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation
pavan
girish
ramana
naga
ramana
pavan
girish
naga
3
Group ideas
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.
20 minutes
Retrospective
What went well..
What did'nt went well
Action


Brainstorm
Write down any ideas that come to mind that address your problem statement.
Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation
pavan
girish
ramana
naga
ramana
pavan
girish
naga
Group ideas
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.
Retrospective
What went well..
What did'nt went well
Action

3.3. PROPOSED SOLUTION:

S.No.	Parameter	Description
1.	Problem Statement (Problem to besolved)	Classifying this disorder for the ECG will the best challenge because only few dissimilar variations are there between every type of thisdisorder. We know that Arrhythmia is a heart related disorder and can either be curable or uncurable.
2.	Idea / Solution description	Our vision is to create a model using deeplearning that can read the ECG and can disperse the Arrhythmia even by those similarities.
3.	Novelty / Uniqueness	The classification is developed by using deeplearning with 2-D ECG Spectral Image representation helping to diagnose the difference between each and every Arrhythmia which can be differentiatedby ECG.
4.	Social Impact / Customer Satisfaction	This furtherly helping doctors to diagnose simply and the patients do not need to be panic and worry about their living.
5.	Business Model (Revenue Model)	Artificial Intelligence is today's most heralded technologies and this model arefor health care this application is most likely to be usedby income.
6.	Scalability of the Solution	expansion of the model depends on the training and models accuracy which helps themodel to train itself. More training of the model leads to a goodprediction of the model.

3.4. PROBLEM SOLUTION FIT:



4. REQUIREMENT ANALYSIS:

4.1. FUNCTIONAL *REQUIREMENT:*

Functional Requirements:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration is done through Gmail
FR-2	User Confirmation	Confirmation via Email Confirmation through OTP
FR-3	Image upload	The user can be able to upload the pictures on the web application.
FR-4	Results	The user should be able to view the results i.e the type of arrythmia when they upload the image of ECG.
FR-5	Authentication	The website should authenticate the users once they sign in.
FR-6	Data pre-processing	The data should be pre-processed.

4.2. NON-FUNCTIONAL REQUIREMENTS:

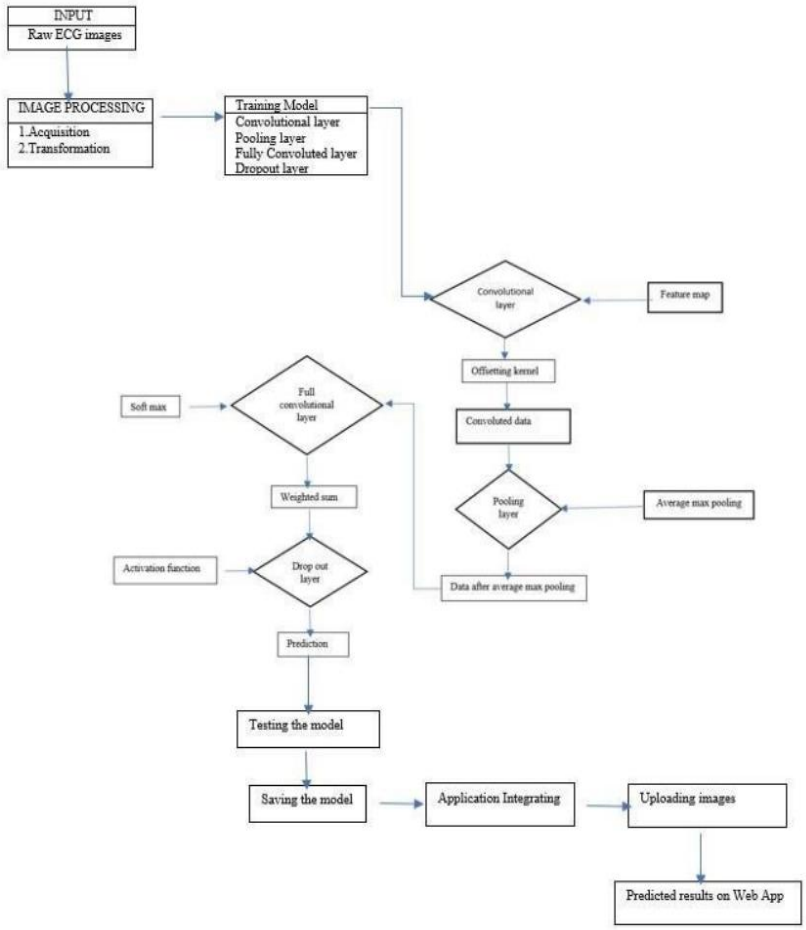
Non-functional Requirements:

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

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The web app should be easy to learn, navigate, use and it should be user-friendly and error tolerant.
NFR-2	**Security**	The web application should be enough to protect the images uploaded by the users and should protect the personal details provided by the users when they sign-up.
NFR-3	**Reliability**	The web application should be consistent thought the results over time between users and it should provide the accurate results to each and every user.
NFR-4	**Performance**	The web page is to load within three seconds and be able to shoe the results within two thirds of the second after the images are uploaded by the users.
NFR-5	**Availability**	The web application should be able to show output at each all times of the day. It can be accessed through the globe.
NFR-6	**Scalability**	The web app should be able sustain the rapid growth for the user count and also showing in time.

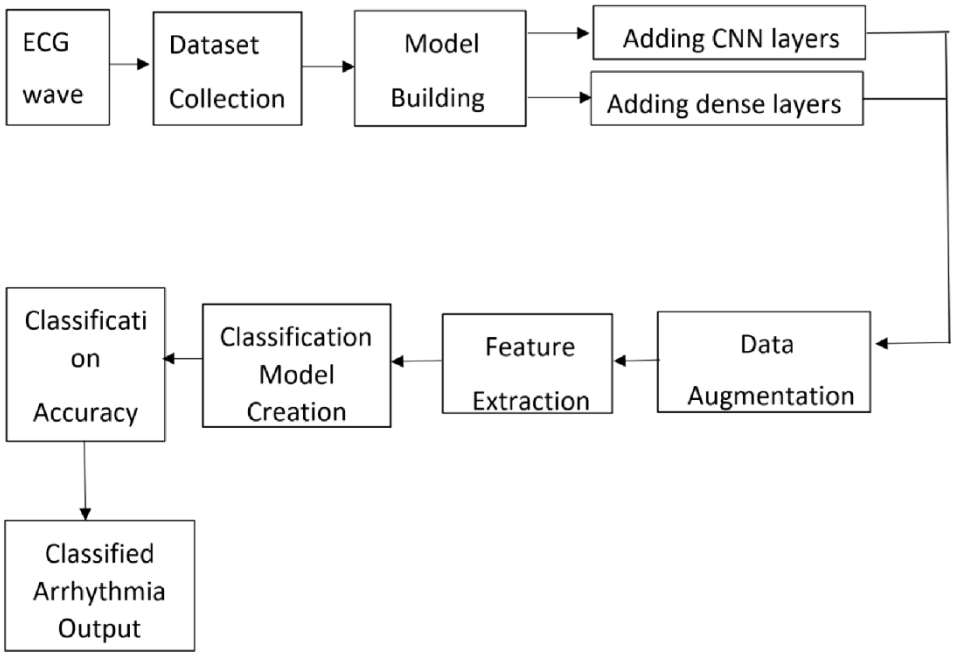
5. PROJECT DESIGN:

5.1. DATA FLOW DIAGRAMS:



5.2. SOLUTION & TECHNICAL ARCHITECTURE:

Solution Architecture:



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
Customer (Web user)	Registration	USN-1	As a web user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a web user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a web user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a web user, I can register for the application through Gmail	I can register & access the dashboard with Gmail	Medium	Sprint-1
	Login	USN-5	As a web user, I can log into the application by entering email & password	I can access my account	High	Sprint-1
	Dashboard	USN-6	As a web user, I can view the picture on the webpage once I uploaded it.	I get and "Upload Successful" pop-up	Medium	Sprint-1
		USN-7	As a web user, I can view the results for the image I uploaded.	Result is displayed.	High	Sprint-1
Customer Care Executive	Login	USN-1	As a Customer Care Executive, I can login with my credentials.	I can access my account	High	Sprint-1
	Dashboard	USN-2	As a Customer Care Executive, I can see all the information, I can view the picture on the webpage once I uploaded it	I get and "Upload Successful" pop-up	High	Sprint-1
	Responsibilities	USN-3	As a Customer Care Executive, I am able to resolve the customers' complaints.	Manage and resolve complaints	High	Sprint-1
Administrator	Login	USN-1	As an administrator, I can login with my credentials.	I can access my account/dashboard	High	Sprint-1
	Dashboard	USN-2	As an administrator, I can see all the information, I can view the picture on the webpage once I uploaded it.	I can see all the information in the dashboard.	Low	Sprint-2

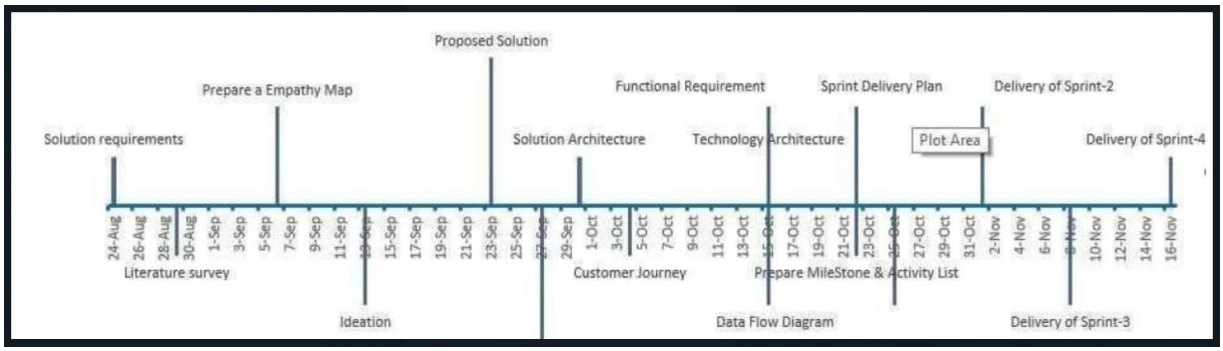
6. PROJECT PLANNING & SCHEDULING:

6.1. SPRINT PLANNING & ESTIMATION:

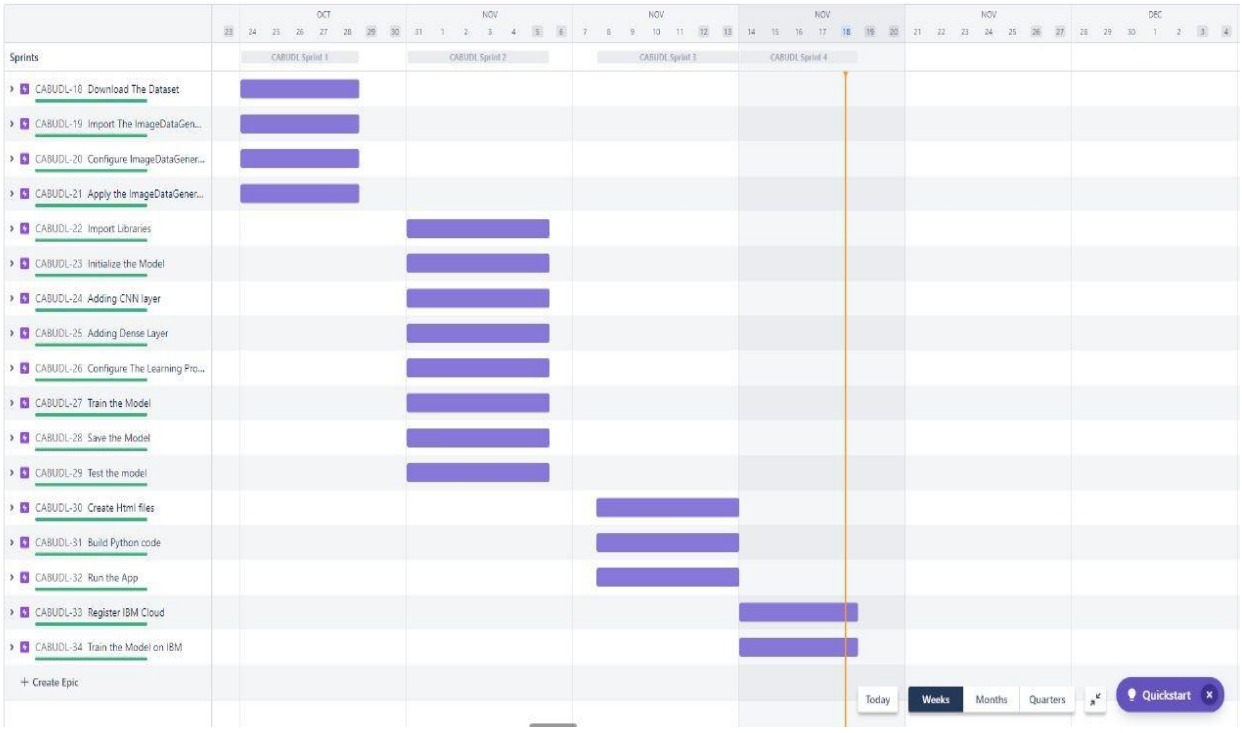
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Download The Dataset	USN-1	We can download the Dataset contains Six classes	4	Low	Pritha R
Sprint-1	Import The ImageDataGenerator Library	USN-2	We can import ImageDataGenerator	4	Low	Ruhie N
Sprint-1	Configure ImageDataGenerator class	USN-3	We can configure the ImageDataGenerator class	6	Medium	Shyam Praveen Singh
Sprint-1	Apply the ImageDataGenerator functionality to Train Set and Dataset	USN-4	We can apply ImageDataGenerator to train dataset	6	Medium	Vijay Anand M
Sprint-2	Import Libraries	USN-5	We can import required Libraries	1	Low	Pritha R
Sprint-2	Initialize the Model	USN-6	Initializing the Image recognition model	2	Medium	Ruhie N
Sprint-2	Adding CNN layer	USN-7	We can add Convolutional Neural Network(CNN) used for image/object recognition and classification	3	High	Shyam Praveen Singh
Sprint-2	Adding Dense Layer	USN-8	We can add Dense Layer in which each neuron receives input from all the neurons of previous layer	3	High	Vijay Anand M
Sprint-2	Configure The Learning Process	USN-9	We can configure The Learning process which is a method, mathematical logic or algorithm that improves the network's performance and/or training time.	4	High	Pritha R
Sprint-2	Train the Model	USN-10	We can train our model with our image dataset. fit_generator functions used to train a deep learning neural network	4	High	Vijay Anand M
Sprint-2	Save the Model	USN-11	We can save The model with .h5 extension	2	Medium	Shyam Praveen Singh
Sprint-2	Test the model	USN-12	We can Test the model through Loaded necessary libraries, the saved model	1	Medium	Ruhie N
Sprint-3	Create Html files	USN-13	We use HTML to create the front end part of the web page.	8	High	Pritha R
Sprint-3	Build Python code	USN-14	We build the flask file 'app.py' which is a web framework written in python for server-side scripting.	8	High	Vijay Anand M
Sprint-3	Run the App	USN-15	We can run the App	4	Medium	Shyam Praveen Singh
Sprint-4	Register IBM Cloud	USN-16	We can register IBM Cloud	8	Medium	Vijay Anand M
Sprint-4	Train the model on IBM	USN-17	We can Train Out model on IBM	12	High	Ruhie N

6.2. SPRINT DELIVERY SCHEDULE:

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	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022



6.3. REPORTS FROM JIRA:



7. CODING & SOLUTIONING:

7.1. FEATURE 1:

7.1.1 App.py

```python
import 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
@app.route("/home") #Home page set to default page
def default():
    return render_template('index.html') #rendering index.html

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

@app.route("/about") #route to about us page
def about_us():
    return render_template('about.html')  #rendering about.html

@app.route("/contact") #route to contact us page
def contact_us():
    return render_template('contact.html')  #rendering contact.html

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

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

        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

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

```
        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[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:

7.2.1Homepage.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>Self Care - Heart Prediction Online</title>
  <link rel="shortcut icon" href="{{url_for('static', filename='images/fevicon.png' )}}" type="image/x-icon">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/aos/2.3.1/aos.css" />
  <link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@600&display=swap" rel="stylesheet" />
  <link rel="stylesheet" href="{{url_for('static', filename='css/style.css' )}}" />
  <script src="https://kit.fontawesome.com/64d58efce2.js" crossorigin="anonymous">
  </script>
</head>

<body>
  <div class="wrapper">
    <!--Navigation Bar-->
    <div class="nav">
      <div class="logo">
        <a href="/">
          <img src="static\images\SELF1 11.png" style="width:190px"/>
        </a>
      </div>
      <div class="links">
        <a href="/home" class="mainLink">Home</a>
        <a href="/info">Info</a>
        <a href="/about">About Us</a>
        <a href="/contact">Contact Us</a>
        <a href="/upload" class="btn1">Predict</a>
      </div>
    </div>
    <!--Landing Page-->
    <div class="landing">
      <div class="landingText" data-aos="fade-up" data-aos-duration="1000">
        <h1>
          Classification of Arrhythmia
          <span style="color: #e0501b; font-size: 4vw">Prediction</span>
        </h1>
        <h3>
          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...
        </h3>

        <div class="btn2"><a href="/info">Read more</a>
        </div>
```

```
    </div>
    <div class="landingImage" data-aos="fade-down" data-aous-duration="2000">
      <img src="static/images/banner_img.jpg" alt="bannerImg" style="width: 500px; height:360px" />
    </div>
  </div>

  <!--Service Section-->
  <div class="about">
    <div class="aboutText" data-aos="fade-up" data-aous-duration="1000">
      <h1 style="margin: 20px;">
        Our Patients Are at Centre
        <span style="color: #2f8be0; font-size: 3vw">of Every We Do</span>
      </h1>
      <div class="image-container">
        <img src="/static/images/connsultPationt.png" alt="consultPationt"
          style="width:400px; margin:100px 0px 0px 90px;"></img>
      </div>
    </div>
    <div class="aboutList" data-aos="fade-left" data-aous-duration="1000">
      <ol>
        <li>
          <span>01</span>
          <p>99.8% accurate result.</p>
        </li>
        <li>
          <span>02</span>
          <p>No need to go hospital.</p>
        </li>
        <li>
          <span>03</span>
          <p>No need to login</p>
        </li>
        <li>
          <span>04</span>
          <p>24/7 Support.</p>
        </li>
      </ol>
    </div>
  </div>

  <!--Info Section-->
  <div class="infoSection">
    <div class="infoHeader" data-aos="fade-up" data-aous-duration="1000">
      <h1>
        We Analyse Youe Health states <br /><span style="color: #e0501b">In Order to Top Service.</span>
      </h1>
    </div>
    <div class="infoCards">
```

```
                </a>
              </div>
            </div>
          </div>
        </div>

        <!--Banner And Footer-->
        <div class="banner">
          <div class="bannerText" data-aos="fade-right" data-aous-duration="1000">
            <h1>
              Download the SelfCare App Today <br /><span style="font-size: 1.6vw; font-weight: normal"
                class="bannerInnerText">Stay Updated and get all your medical needs taken care of!</span>
            </h1>
            <a href="/"><img src="static/images/AndroidPNG.png" alt="" /></a>
            <a href="/"><img src="static/images/iosPNG.png" alt="" /></a>
          </div>
          <div class="bannerImg" data-aos="fade-up" data-aous-duration="1000">
            <img src="static/images/app.png" alt="" />
          </div>
        </div>

        <div class="footer">
          <h1>SelfCare</h1>
          <div class="footerlinks">
            <a href="/home" class="mainLink">Home</a>
            <a href="/info">Info</a>
            <a href="/about">About Us</a>
            <a href="/contact">Contact Us</a>
          </div>
        </div>
      </div>
      <script src="https://cdnjs.cloudflare.com/ajax/libs/aos/2.3.1/aos.js"></script>
      <script>
        AOS.init();
      </script>
    </body>

    </html>
```

7.2.2. about.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">
    <link rel="shortcut icon" href="{{url_for('static', filename='images/fevicon.png' )}}" type="image/x-icon">
    <title>Self Care - About Us</title>
    <link rel="stylesheet" href="{{url_for('static', filename='css/about.css')}}">
    <link rel="stylesheet" href="{{url_for('static', filename='css/style.css')}}">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/aos/2.3.1/aos.css" />
    <link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@600&display=swap" rel="stylesheet" />
    <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css" rel="stylesheet">
</head>
<style>
    .footer {
        margin-bottom: 20px;
    }

    h1 {
        margin: 21.44px 0px;
    }
</style>

<body>
    <div class="wrapper">
        <!--Navigation Bar-->
        <div class="nav">
            <div class="logo">
                <a href="/"><img src="static/images/SELF1 11.png" alt="Website Title" style="width:190px" /></a>
            </div>
            <div class="links">
                <a href="/">Home</a>
                <a href="/info">Info</a>
                <a href="/about" class="mainLink">About Us</a>
                <a href="/contact">Contact Us</a>
                <a href="/upload" class="btn1">Predict</a>
            </div>
        </div>
        <div class="landing">
            <div class="landingText" data-aos="fade-up" data-aous-duration="1000">
                <h1>
                    We are a team of
                    <span style="color: #e0501b; font-size: 4vw">Arrthymia Prediction</span>
                </h1>
                <h3>
                    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.
            </h3>
        </div>
        <div class="landingImage" data-aos="fade-down" data-aous-duration="2000">
            <img src="static/images/about_us.png" alt="aboutImg" style="width: 450px; height:450px" />
        </div>
    </div>
    <div class="main">
        <div class="profile-card">
            <div class="img">
                <img src="static/images/vjaavatar.jpg">
            </div>
            <div class="caption">
                <h3>Vijay Anand </h3>
                <p>Web Developer, Full Stack Developer, AI Learning Engineer</p>
                <div class="social-links">
                    <a href="#"><i class="fab fa-facebook"></i></a>
                    <a href="#"><i class="fab fa-instagram"></i></a>
                    <a href="#"><i class="fab fa-twitter"></i></a>
                </div>
            </div>
        </div>
        <div class="profile-card">
            <div class="img">
                <img src="static/images/profile_avatar.png">
            </div>
            <div class="caption">
                <h3>Pritha V </h3>
                <p>Front End Developer, Machine Learning Engineer</p>
                <div class="social-links">
                    <a href="#"><i class="fab fa-facebook"></i></a>
                    <a href="https://www.instagram.com/the_._._champ/"><i class="fab fa-instagram"></i></a>
                    <a href="#"><i class="fab fa-twitter"></i></a>
                </div>
            </div>
        </div>
        <div class="profile-card">
            <div class="img">
                <img src="static/images/profile_avatar.png">
            </div>
            <div class="caption">
```

```
                    <div class="caption">
                        <h3>Ruhie N</h3>
                        <p>Back End Developer, AI Learning Engineer</p>
                        <div class="social-links">
                            <a href="#"><i class="fab fa-facebook"></i></a>
                            <a href="#"><i class="fab fa-instagram"></i></a>
                            <a href="#"><i class="fab fa-twitter"></i></a>
                        </div>
                    </div>
                </div>
                <div class="profile-card">
                    <div class="img">
                        <img src="static/images/sh.jpg">
                    </div>
                    <div class="caption">
                        <h3>Shyam Praveen Singh</h3>
                        <p>Full Stack Developer, Machine Learning Engineer</p>
                        <div class="social-links">
                            <a href="#"><i class="fab fa-facebook"></i></a>
                            <a href="#"><i class="fab fa-instagram"></i></a>
                            <a href="#"><i class="fab fa-twitter"></i></a>
                        </div>
                    </div>
                </div>
            </div>
            <div class="footer">
                <h1>SelfCare</h1>
                <div class="footerlinks">
                    <a href="/home">Home</a>
                    <a href="/info">Info</a>
                    <a href="/about" class="mainLink">About Us</a>
                    <a href="/contact">Contact Us</a>
                </div>
            </div>
        </div>

        </div>
        <script src="https://cdnjs.cloudflare.com/ajax/libs/aos/2.3.1/aos.js"></script>
        <script>
            AOS.init();
        </script>
    </body>

    </html>
```

7.2.3. Predict.html

```
{% extends "predict_base.html" %} {% block content %}

<center>
    <h2 style="font-size: 40px;">
        ECG Arrhythmia
        <span style="color: #2f8be0; font-size: 3vw">Classification</span>
    </h2>
</center>

<div>
    <form id="upload-file" method="post" enctype="multipart/form-data">
        <center> <label for="imageUpload" class="upload-label">
                Choose...
            </label>
            <input type="file" name="file" id="imageUpload" accept=".png, .jpg, .jpeg">
        </center>
    </form>

    <center>
        <div class="image-section" style="display:none;">
            <div class="img-preview">
                <div id="imagePreview">
                </div>
            </div>
        </div>
    </center>
</div>
<center>
    <div class="btn3" id="btn-predict"
        style="padding: 8px 34px; width: 120px; margin-top: 30px; padding: 14px 20px 12px 20px; background-color: #007bff; border-radius: 45px; text-align: center; color: #fff; cu
        Predict</div>
    <div class="loader" style="display:none;"></div>
</center>

<h3 style="color:Black" id="result">
    <span> </span>
</h3>

</div>

</div>



{% endblock %}
```

8. TESTING:

8.1. TEST CASES:

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status
EditorPage_TC_OO 1	Functional	Visual Studio Editor	Open the Visual Studio Editor and click the app.py	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		Working correctly	Working as expected	Pass
Navigation_TC_O O2	Functional	Home Page	Validate all the tabs in the navigator	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		All the 3 tabs should vissible	Working as expected	Pass
Home_TC_OO3	Functional	Home page	Verify the Visibility of the image	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		User should able to see the image	Working as expected	Pass
HomePage_TC_OO 4	Functional	Home page	Validate the description of the image	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		Description should be visible on the window	Working as expected	Pass
HomePage_TC_OO 5	Functional	Home page	Verify the user is able to navigate	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		It should redirect the user to the same window	Working as expected	Pass
InfoPage_TC_OO_1	Functional	Info Page	Verify the user is in the introduction	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		User should be in the introduction	Working as expected	Pass
InfoPage_TC_OO_2	Functional	Info Page	verify the page title and information	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		User should able to view the page	Working as expected	Pass
PredictPage	Functional	Predict	verify the working of predict page	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		User should be able to visit the page	Working as expected	Pass
PredictPage	Functional	Predict	verify the upload image option	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		Make sure the option works	Working as expected	Pass
Predict	Functional	Predict	Verify the choose button is enable	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		The Choose button option should	Working as expected	Pass
Predict	Functional	Predict	Verify the user is able to access	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		Image should be Uploaded	Working as expected	Pass
Predict	Functional	Predict	verify the selected image is same	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		Selected image should be ECG 2D array structure	Working as expected	Pass
Predict	Functional	Predict	verify the working condition	Knowledge about Python, keras, tensorflow	1.Enter URL and click go		The Type of Arrhythmia should be found and result is displayed in the predict window	Working as expected	Pass

8.2. USER ACCEPTANCE TESTING:

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

3. Test Case Analysis

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

Section	Total Cases	Not Tested	Fail	Pass
Predict Page	7	0	0	7
Different Types of ECG Images	10	0	0	10
Further Classification	4	0	0	4
Home Page	3	0	0	3
Info Page	6	0	0	6
Final Report Output	1	0	0	1

9. RESULTS:

9.1. PERFORMANCE METRICS:

The two significant optimization parameters in the proposed 2-D CNN model are the learning rate and the batch size of the data used. To improve the performance, these two optimization parameters must be selected carefully to obtain the best accuracy in the automatic classification of arrhythmia using the ECG signals. The proposed model was evaluated in different experiments with various values of learning parameters. For a smaller value of the learning rate (i.e., less than 0.0005), the speed of the convergence was very slow. However, when the value of the learning rate was large (i.e., greater than0.001), the speed of convergence improved. At the same time, asymmetrical changes were observed in the accuracy rate. Henceforth, we selected an optimum value of 0.001 for the learning rate, as this value can attain better accuracy for the proposed model (i.e., optimum value).

Batch sizes and average accuracy for a learning rate of 0.001.

Learning Rate	Batch Size	Average Accuracy
0.001	2800	99.11
0.001	2000	98.96
0.001	1000	99.00
0.001	500	98.95
0.001	100	98.93

Learning rate and average accuracy for a batch size of 2800.

Batch Size	Learning Rate	Average Accuracy
2800	0.001	99.11
2000	0.001	98.96
1000	0.001	99.00
500	0.001	98.95
100	0.001	98.93

We compared the proposed CNN-based model with recent techniques for the automatic classification of arrhythmia where the algorithm achieved 97.88% average sensitivity, 99.61% specificity, 99.11% average accuracy, and 98.59% positive predictive value (precision). These values indicate improved performance when compared with recent methods using of 1-D and 2-D CNNs, given the same arrhythmia classification. The results also show that the proposed CNN algorithm has better results in terms of accuracy with both the augmented and without augmented data.

The proposed model has attained the highest sensitivity among all the compared CNN algorithms.It is pertinent to note that detecting these cardiac arrhythmias is a labor intensive task, where a clinical expert needs to carefully observe recordings that can go for up to hours. With such automated methods,the artificially intelligent system could augment the performance of clinical experts by detecting these patterns and directing the observer to look more closely at regions of more significance. This would ultimately improve the clinical diagnosis and treatment of some of the major CVDs.

Model Performance Testing:

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

S.No.	Parameter	Values	Screenshot
1.	Model Summary	-	
2.	Accuracy	Training Accuracy - Validation Accuracy -	
3.	Confidence Score (Only Yolo Projects)	Class Detected - Confidence Score -	

10. ADVANTAGES & DISADVANTAGES:

10.1. ADVANTAGES :

- 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.
- To overcome the challenge for the visual and physical explanation of the ECG signal, computer-aided study has been developed to automatically identify such signals automatically.
- A careful study of ECG signals is crucial for precise diagnoses of patients' acute and chronic heart conditions.
- The performance is significant in other indices as well, including sensitivity and specificity, which indicates the success of the proposed method.
- A simple User Interface for the users to classify their ECG report.
- Great User Experience to make is simple and easy to use for users and patients while accessing the website

10.2 DISADVANTAGES:

- The efficiency and accuracy of could be negatively affected by the increasing size of data.
- The techniques presented been applied to smaller datasets.
- For the purpose of generalization, the performance should betested on larger datasets.
- Most methods have been tested on only a few types of arrhythmia and must be evaluated on all major types of arrhythmia.
- It should be noted that the performance of methods developed for 1-D ECG signals can be further improved.

11. CONCLUSION:

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 heart diseases. Deep CNN has proven useful in enhancing the accuracy of diagnosis algorithms in the fusion of medicine and modern machine learning technologies. The proposed CNN-based classification algorithm, using 2-D images, can classify seven kinds of arrhythmia. 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 heart diseases. The proposed scheme can help experts diagnose heart diseases 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.

12. FUTURE SCOPE:

For further study, it would be intresting 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.

These are the further improvements that can be made in the future: -

- Further classification for upto 8-10 different types of arrhythmia will be possible.

- Improved accuracy for bigger data size.
- The techniques presented been applied to smaller datasets can be made much faster and efficient.
- Improved UI & UX for the users.

13. APPENDIX:

SOURCE CODE LINK:

https://github.com/IBM-EPBL/IBM-Project-23797-1659930232/tree/main/Final%20Deliverables

PROJECT DEMO LINK:

https://youtu.be/PUv5gO0hMpU

GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-23797-1659930232