



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



Nalaiya Thiran

Professional Readiness for Innovation, Employability & Entrepreneurship

A Project Report

Submitted by

SHAKTHI. S (311419106024)

SHRUTHI. I (311419106026)

CHELSIA STELLA. P (311419106007)

LAVANYA. G (311419106013)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING

MEENAKSHI COLLEGE OF ENGINEERING,

WEST K.K. NAGAR

ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

FACULTY MENTOR

R.ANANDHA PRABA

FACULTY EVALUATOR

D.SATHEESWARI

INTRODUCTION

OVERVIEW:

As cardiovascular diseases (CVDs) are a serious concern to modern medical science to diagnose at an early stage, it is vital to build a classification model that can effectively reduce mortality rates by treating millions of people in a timely manner. An electrocardiogram (ECG) is a specialized instrument that measures the heart's physiological responses. To accurately diagnose a patient's acute and chronic heart problems, an in-depth examination of these ECG signals is essential. The proposed model consists of a convolutional neural network having three convolutional, two pooling, and two dense layers. The proposed model is trained and evaluated on the MIT-BIH arrhythmia and PTB diagnostic datasets. The classification accuracy is 99.16%, which is higher than state-of-the-art studies on similar arrhythmias. Recall, precision, and F1 score of the proposed model are 96.53%, 95.15%, and 99.17%, respectively. The proposed model can aid doctors explicitly for the detection and classification of arrhythmias.

LITERATURE SURVEY:

REFERENCES:

- [1]. A Hybrid Deep Learning Approach for ECG-Based Arrhythmia Classification
<https://www.mdpi.com/2306-5354/9/4/152> Parul Madan, Vijay Singh, Devesh Pratap Singh, Manoj Diwakar, Bhaskar Pant, Avadh Kishor 2022
- [2]. Automated ECG multi-class classification system based on combining deep learning features with HRV and ECG measures <https://link.springer.com/article/10.1007/s00521-022-06889-z> Ahmed S Eltrass, Mazhar B Tayel, Abeer I Ammar 2022
- [3]. Analysis of Digitalized ECG Signals Based on Artificial Intelligence and Spectral Analysis Methods Specialized in ARVC
<https://arxiv.org/abs/2203.00504> Vasileios E Papageorgiou, Thomas Zegkos, Georgios Efthimiadis, George Tsaklidis ---2022
- [4]. Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation. Amin Ullah , Syed Muhammad Anwar, Muhammad Bilal and Raja Majid Mehmood ---2020
- [5]. Detecting stress through 2D ECG images using pretrained models, transfer learning and model Compression techniques. Syem Ishaque , Naimul Khan, Sri Krishnan---2022

[6]. Deep learning-based classification of multichannel bio-signals using directedness transfer learning. Nooshin Bahador , Jukka Kortelainen ----2022

[7]. A clinical study on Atrial Fibrillation, Premature Ventricular Contraction, and Premature Atrial Contraction screening based on an ECG deep learning model Jianyuan Hong a,b, Hua-Jung Li a , Chung-chi Yang c, Chih-Lu Han d, Juichien Hsieha

[8].Two-dimensional ECG-based cardiac arrhythmia classifcation using DSEResNet Jiahao Li , Shao-peng Pang, Fangzhou Xu, Peng Ji, Shuwang Zhou, & Minglei Shu

[9]. A clinical study on Atrial Fibrillation, Premature Ventricular Contraction, and Premature Atrial Contraction screening based on an ECG deep learning model Jianyuan Hong a,b, Hua-Jung Li a , Chung-chi Yang c, Chih-Lu Han d, Juichien Hsieha

[10].Two-dimensional ECG-based cardiac arrhythmia classifcation using DSEResNet Jiahao Li , Shao-peng Pang, Fangzhou Xu, Peng Ji, Shuwang Zhou, & Minglei Sh

Ideation Phase

Define the Problem Statements

Date	14 October 2022
Team ID	PNT2022TMID27741
Project Name	Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation
Maximum Marks	2 Marks

Customer Problem Statement Template:

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

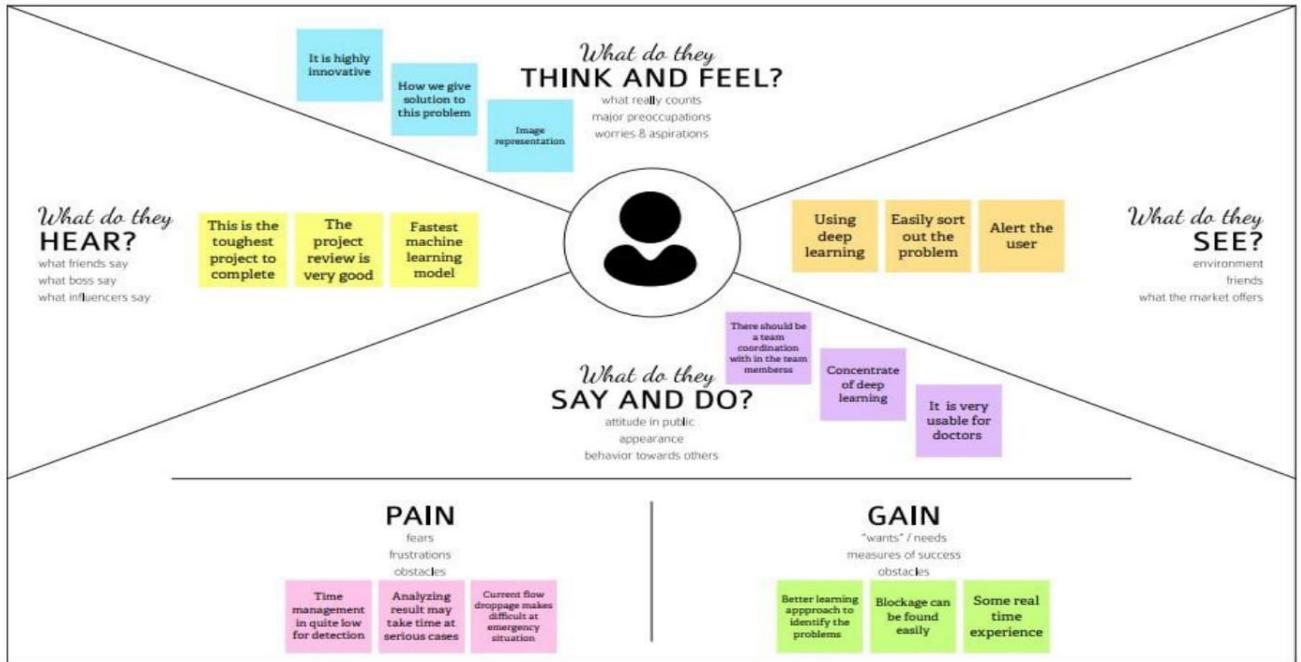


Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me Feel
PS-1	A student	To score first mark in my public exam	Some times pain in my heart	I am working too much	Scared
PS-2	Parent	Take care about my child	I am too weak	Sometimes my heart beat is too faster continuously	Anxious

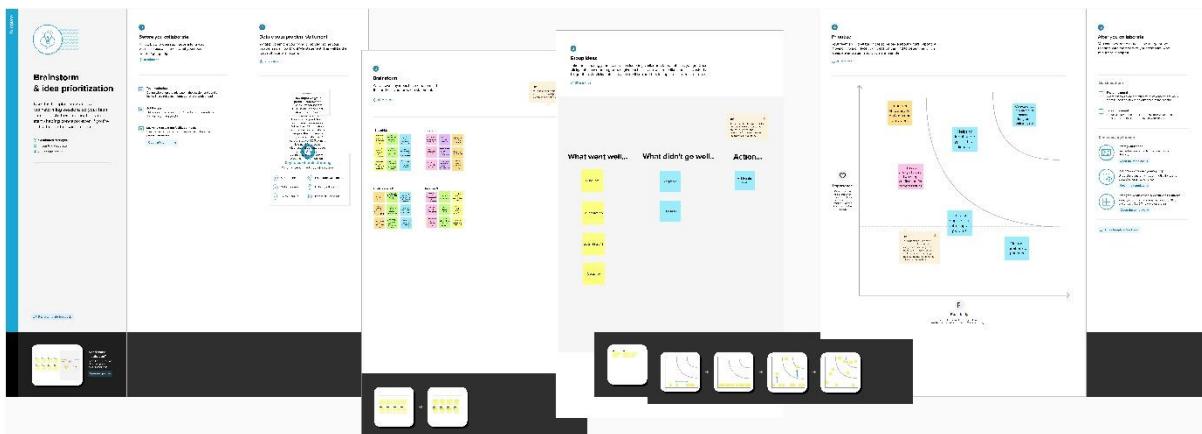
PS-3	A volleyball player	Has power to hit the ball	I feel sometimes I don't have strength	My aim is distracted sometimes	Sadly
PS-4	Person	Healthy and wont have heart diseases	My heart is hurting	I think I have a heart disease	Frustrated

+

Classification of arrhythmia by using deep learning with 2-d ecg spectral image representation



BRAINSTORMING & IDEA PRIORITISATION



Project Design Phase-I

Proposed Solution Template

Date	14 October 2022
Team ID	PNT2022TMID427741
Project Name	Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation
Maximum Marks	2 Marks

Proposed Solution Template:

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

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	Build an effective electrocardiogram (ECG) arrhythmia classification method using a convolutional neural network (CNN)
2.	Idea / Solution description	Classify ECG using deep two-dimensional(2-D) CNN with grayscale ECG images
3.	Novelty / Uniqueness	When the image is fed into the model, the cited class will be displayed on the webpage
4.	Social Impact / Customer Satisfaction	Using this Method, We can get classification accurate
5.	Business Model (Revenue Model)	Creating a web application where the user selects the image which is to be classified
6.	Scalability of the Solution	It can classify into seven categories, one being normal and the other six being different types of arrhythmia

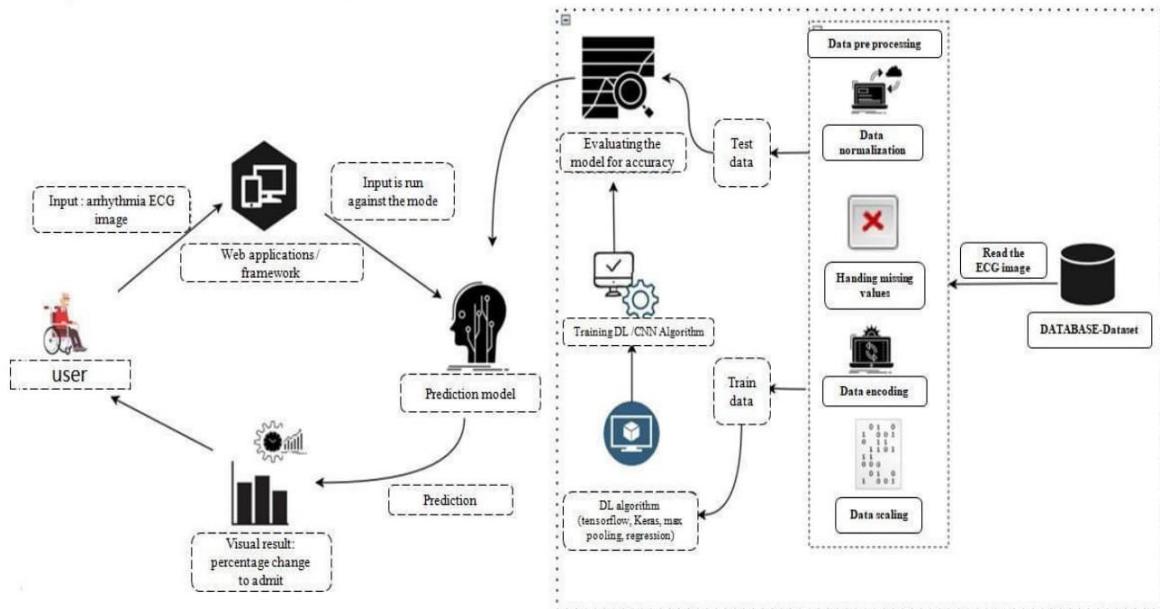
1. CUSTOMER SEGMENT(S) CS A Plumber who have heart disease but he don't have the time to go to hospital	6. CUSTOMER CONSTRAINTS CC Identify heart disease because of several contributory risk factors such as diabetes, high blood pressure, high cholesterol, abnormal pulse rate	5. AVAILABLE SOLUTIONS AS Healthy lifestyle habits such as eating a low-fat, low-salt diet, getting regular exercise and good sleep, and no smoking
2. JOBS-TO-BE-DONE / PROBLEMS J&P Find heart problem to cure the disease	9. PROBLEM ROOT CAUSE RC Risk factors include a poor diet, lack of exercise, obesity and smoking. Healthy lifestyle choices can help lower the risk of atherosclerosis	7. BEHAVIOUR BE Protect you from type 2 diabetes, asthma, joint pain, and a number of other chronic diseases and conditions
3. TRIGGERS TR Symptoms : Symptoms may include chest pain, nausea, shortness of breath, sweating, dizziness, palpitations.	10. YOUR SOLUTION SL Vitamin C. Arrhythmias and other heart conditions are associated with oxidant stress and inflammation. Antioxidants like vitamin C and vitamin E appear to be effective in reducing these. You can use vitamin C to treat colds, the flu, and even cancer, and it can also help with arrhythmia.	8. CHANNELS OF BEHAVIOUR CH 8.1 ONLINE Customer will Find their heart disease online rather than going hospital 8.1 OFFLINE Customer will collect their ecg image offline going hospital
4. EMOTIONS: BEFORE / AFTER EM Before : Especially negative emotions, such as hostility, anger, depression and anxiety, precipitate coronary heart disease After : Temporary feelings of sadness and a depressed mood are common for the first few weeks		

Project Design Phase-I

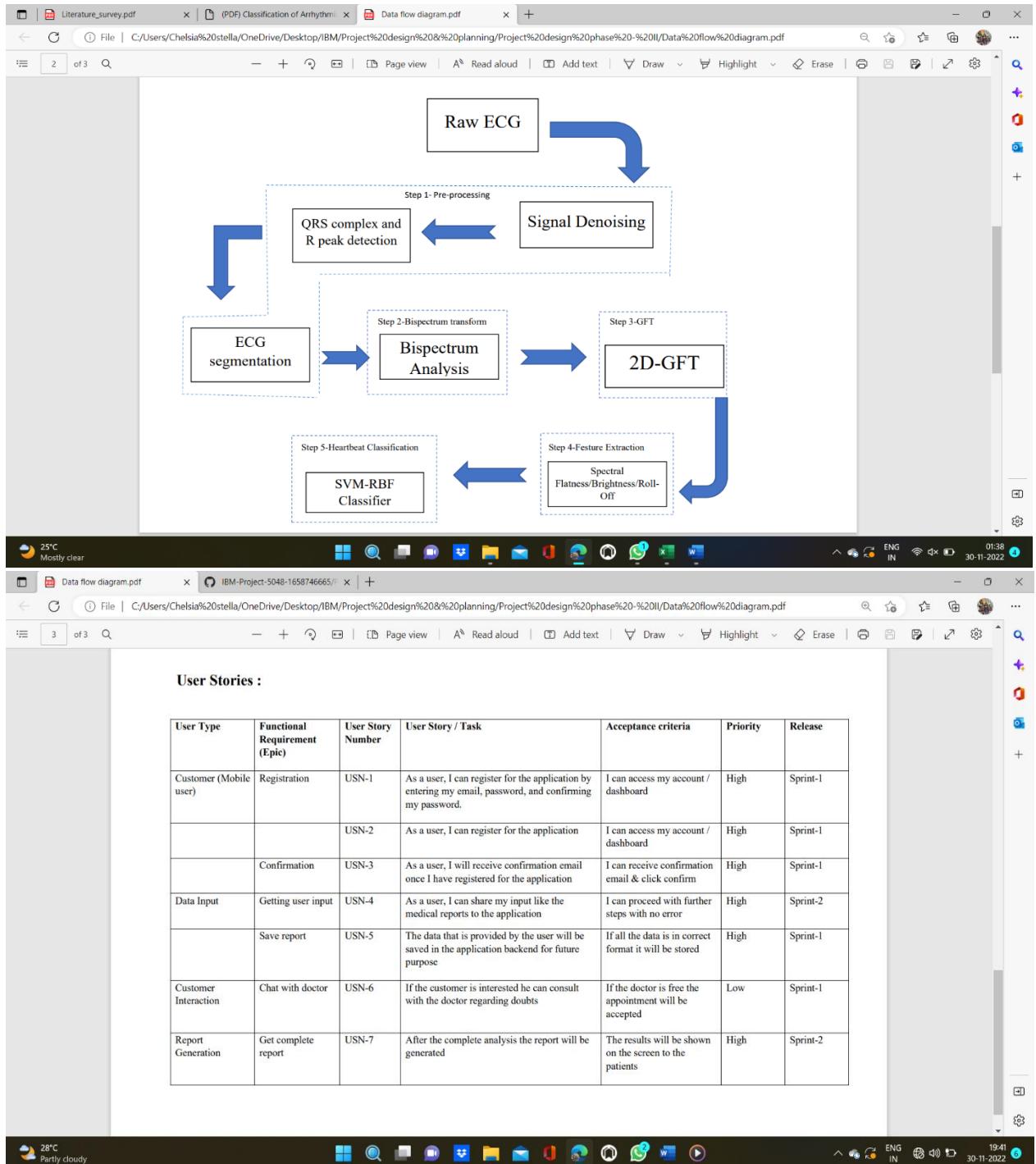
Solution Architecture

Date	14 October 2022
Team ID	PNT2022TMID27741
Project Name	Classification of arrhythmia by using deep learning with 2-d ecg spectral image representation
Maximum Marks	4 Marks

Minimum Viable Architecture for MVP



DATAFLOW DIAGRAM



FUNCTIONAL REQUIREMENTS

Project Design Phase-II
Solution Requirements (Functional & Non-functional)

Date	17 October 2022
Team ID	PNT202TMID27741
Project Name	Classification of arrhythmia using deep learning with 2-D ECG spectral image Representation
Maximum Marks	4 Marks

Functional Requirements:
Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIn
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User selection	Knowledge about ECG images select the image to be classified
FR-4	User input	Images need to be uploaded
FR-5	Save image	Images are saved in uploaded folder
FR-6	Predict ECG image	User ECG images in our web application Collection of data sets Database read ECG images

Non-functional Requirements:
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	An user friendly and simple UI Web application . Easy drag and drop options
NFR-2	Security	No third party web and UI is used for prediction of data Details about user interaction with the web application are protected
NFR-3	Reliability	Higher accuracy rate Defect free

NFR-4	Performance	Fast and quick classification of the required class is done
NFR-5	Availability	Availability describes how likely the system is accessible to a user at a given point in time and the periodically for a solutions.
NFR-6	Scalability	The ability of the user problem in arrhythmia disease to handle an increase in workload without performance degradation

TECHNOLOGY ARCHITECTURE

IBM-Project-5048-1658746665/ Functional requirement.pdf Technology Architecture.pdf

1 of 3

Project Design Phase-II
Technology Architecture

Date	17 October 2022
Team ID	PNT2022TMID27741
Project Name	Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation
Maximum Marks	4 Marks

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

```
graph LR; User((User)) -- "Browsing" --> Connected((Connected with website)); Connected --> Watson((IBM Watson studio)); Watson --> Dataset((Dataset)); Dataset --> TrainedData((Trained Data)); TrainedData --> Testing((Testing)); Testing --> Prediction((Prediction)); Prediction --> UI((UI design)); Admin((Admin)) -- "UI design" --> UI;
```

IBM-Project-5048-1658746665/ Functional requirement.pdf Technology Architecture.pdf

2 of 3

Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	Web UI, Mobile UI.	HTML, CSS, JavaScript / React Js.
2.	Application Logic-1	Python is used for backend	Python
3.	Application Logic-2	It's a symbolic math toolkit that performs a variety of tasks including deep neural network training and inference using dataflow and differentiable programming	Tensorflow
4.	Cloud Database	A global technology company that provides hardware, software, cloud-based services and cognitive computing.	IBM Cloud
5.	File Storage	Breaks up data into blocks and then stores those blocks as separate pieces, each with a unique identifier.	IBM Block
6.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
7.	External API-2	Purpose of External API used in the application	Aadhar API, etc.
8.	Machine Learning Model	Object recognition is a subfield of computer vision, artificial intelligence, and machine learning	Object Recognition Model

PROJECT PLANNING-MILESTONE & ACTIVITY

IBM-Project-504b-1658746655/ Milestone & Activity List.pdf

Project Planning Phase

Project Planning Template (Milestone & Activity List)

Date	22 October 2022
Team ID	PNT2022TMID27741
Project Name	Classification Of Arrhythmia By Using Deep Learning With 2-D ECG Spectral Image Representation
Maximum Marks	8 Marks

Product Backlog, Sprint Schedule, and Estimation (4 Marks)
Use the below template to create product backlog and sprint schedule

28°C Partly cloudy

IBM-Project-504b-1658746655/ Milestone & Activity List.pdf

SNO	MILESTONE	ACTIVITIES	DATE
1.	Preparation Phase	Pre-requisites Prior knowledge Project Structure Project Flow Project Objectives Registrations Environment Set-up	24 Aug 2022 25 Aug 2022 25 Aug 2022 25 Aug 2022 22 Aug 2022 26 Aug 2022 27 Aug 2022

SNO	MILESTONE	ACTIVITIES	DATE
2.	Identification Phase	Literature Survey Entity Map Problem Statement Identification	29 Aug 2022 - 01 Sept 2022 01 Sept 2022 - 7 Sept 2022 08 Sept 2022 - 18 Sept 2022 12 Sept 2022 - 16 Sept 2022
3.	Project Design Phase -I	Proposed Solution Problem Solution Fit Solution Architecture	19 Sept 2022 - 23 Sept 2022 24 Sept 2022 - 26 Sept 2022 27 Sept 2022 - 30 Sept 2022

28°C Partly cloudy

IBM-Project-504b-1658746655/ Milestone & Activity List.pdf

4.	Project Design Phase -II	Customer Journey	03 Oct 2022 - 08 Oct 2022
		Requirement Analysis	09 Oct 2022 - 11 Oct 2022
		Data Flow Diagrams	11 Oct 2022 - 14 Oct 2022
		Technology Architecture	15 Oct 2022 - 16 Oct 2022
5.	Project Planning Phase	Milestones & Tasks	17 Oct 2022 - 18 Oct 2022
		Sprint Schedules	19 Oct 2022 - 22 Oct 2022
6.	Project Development Phase	Sprint-1	24 Oct 2022 - 28 Oct 2022
		Sprint-2	30 Oct 2022 - 04 Nov 2022

28°C Partly cloudy

IBM-Project-5048-1658746665/ Milestone & Activity List.pdf

			Sprint-3	06 Nov 2022- 11 Nov 2022
			Sprint-4	13 Nov 2022 - 18 Nov 2022
	Sprint-1		<ul style="list-style-type: none"> • Download The Dataset • Import ImageDataGenerator Library • Configure ImageDataGenerator class • Import Libraries • Initialize the Model 	24 Oct 2022 – 28 Oct 2022
	Sprint – 2		<ul style="list-style-type: none"> • Register IBM Cloud • Apply ImageDataGenerator functionality to Trainset and Dataset • Test the model 	30 Oct 2022 – 04 Nov 2022

IBM-Project-5048-1658746665/ Milestone & Activity List.pdf

			Sprint-3	06 Nov 2022- 11 Nov 2022
			Sprint-4	13 Nov 2022 - 18 Nov 2022
	Sprint-1		<ul style="list-style-type: none"> • Download The Dataset • Import ImageDataGenerator Library • Configure ImageDataGenerator class • Import Libraries • Initialize the Model 	24 Oct 2022 – 28 Oct 2022
	Sprint – 2		<ul style="list-style-type: none"> • Register IBM Cloud • Apply ImageDataGenerator functionality to Trainset and Dataset • Test the model 	30 Oct 2022 – 04 Nov 2022
	Sprint – 3		<ul style="list-style-type: none"> • Train the model on IBM • Create Html files • Train the Model 	06 Nov 2022 – 11 Nov 2022
	Sprint – 4		<ul style="list-style-type: none"> • Configure The Learning Process • Build Python code • Adding Dense Layer • Adding CNN layer 	13 Nov 2022 – 18 Nov 2022

PROJECT PLANNING PHASE

Sprint-2 Apply the ImageDataGenerator USN-4 We can apply ImageDataGenerator to train dataset 2 Medium Lavanya.G Shruthi.I

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	functionality to Trainset and Dataset					Chelsia Stella.P
Sprint-1	Import Libraries	USN-5	We can import required Libraries	1	Low	Chelsia Stella.P
Sprint-1	Initialize the Model	USN-6	Initializing the Image recognition model	2	Medium	Shakthi.S Shruthi.I
Sprint-4	Adding CNN layer	USN-7	We can add Convolutional Neural Network (CNN) used for image/object recognition and classification	4	High	Shakthi.S Chelsia Stella.P
Sprint-4	Adding Dense Layer	USN-8	We can add Dense Layer in which each neuron receives input from all the neurons of previous layer	4	High	Shakthi.S Lavanya.G Chelsia Stella.P
Sprint-4	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	Shakthi.S Lavanya.G Shruthi.I Chelsia Stella.P

Project Planning Phase
Project Planning Template (Product Backlog, Sprint Planning, Stories, Story points)

Date	22 October 2022
Team ID	PNT2022TMID27741
Project Name	Classification Of Arrhythmia by Using Deep Learning With 2-D ECG Spectral Image Representation
Maximum Marks	8 Marks

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

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	1	Low	Shruthi.I Chelsia Stella.P
Sprint-1	Import The ImageDataGenerator Library	USN-2	We can import ImageDataGenerator	1	Low	Shruthi.I Chelsia Stella.P
Sprint-1	Configure ImageDataGenerator class	USN-3	We can configure the ImageDataGenerator class	1	Low	Shakthi.S Lavanya.G Chelsia Stella.P

Sprint-3	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	3	High	Shakthi.S Chelsia Stella.P
Sprint-1	Save the Model	USN-11	We can save The model with .h5 extension	2	Medium	Shruthi.I Chelsia Stella.P
Sprint-2	Test the model	USN-12	We can Test the model through Loaded necessary libraries, the saved model	2	Medium	Shakthi.S Shruthi.I

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-3	Create Html files	USN-13	We use HTML to create the front-end part of the web page.	3	High	Shakthi.S Lavanya
Sprint-4	Build Python code	USN-14	We build the flask file 'app.py' which is a web framework written in python for server-side scripting.	4	High	Shruthi.I Chelsia Stella.P
Sprint-1	Run the App	USN-15	We can run the App	2	Medium	Shakthi.S Chelsia Stella.P
Sprint-2	Register IBM Cloud	USN-16	We can register IBM Cloud	2	Medium	Shakthi.S Chelsia Stella.P
Sprint-3	Train the model on IBM	USN-17	We can Train Out model on IBM	3	High	Shakthi.S Chelsia Stella.P

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

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	5 Days	24 Oct 2022	28 Oct 2022	20	28 Oct 2022
Sprint-2	20	5 Days	30 Oct 2022	04 Nov 2022	20	04 Nov 2022
Sprint-3	20	5 Days	06 Nov 2022	11 Nov 2022	20	11 Nov 2022
Sprint-4	20	5 Days	13 Nov 2022	18 Nov 2022	20	18 Nov 2022

Velocity:
To calculate the team's **average velocity (AV)** per iteration unit

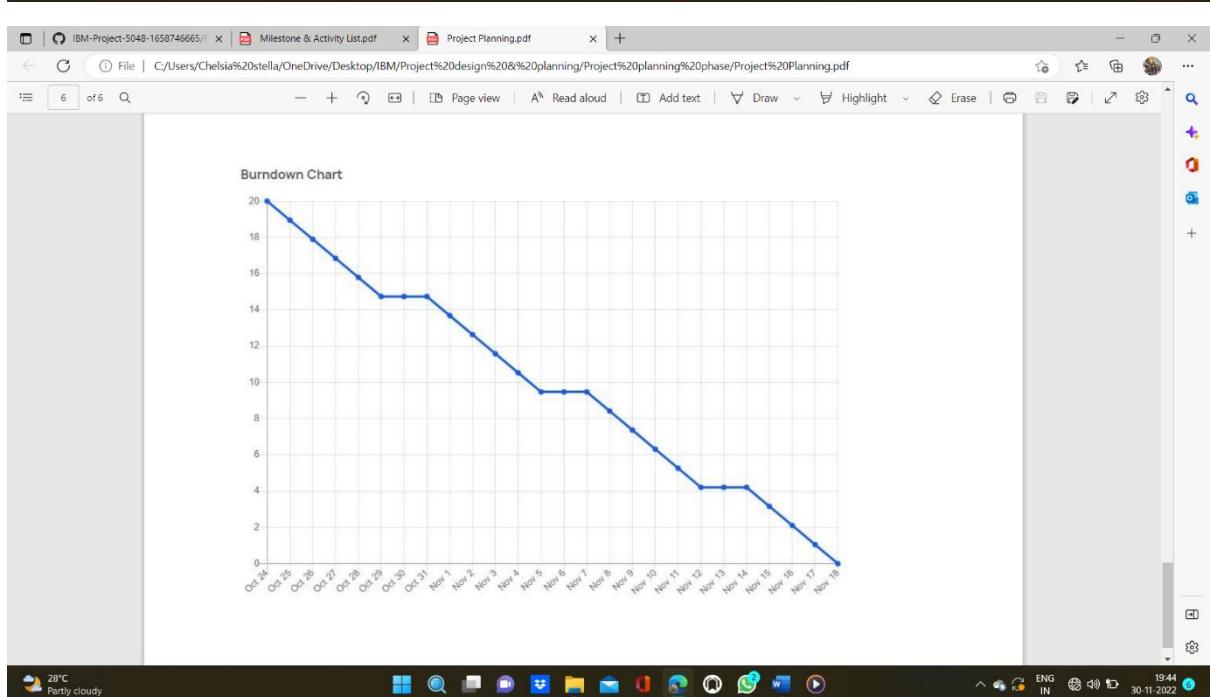
$$Av = \frac{\text{Total Story Points}}{\text{Sprint duration}}$$

Where,

Average Velocity - Story points per day
 Sprint duration - Number of days (Duration) for Sprints
 Velocity - Points per Sprint

$$Av = \frac{20}{5} = 4$$

Average Velocity is 4 points per Sprint



DEVELOPMENT PHASE

SPRINT 1

IBM-Project-5048-1658746665/ Sprint 1.pdf

1 of 3

SPRINT 1: Classification of Arrhythmia by Using Deep Learning With 2-D ECG Spectral Image Representation

Team ID: PNT2022TMID27741
Team members: Shakthi, Shruthi, Chelsia stella, Lavanya

Code: Updated in GitHub in the Deliverables section in Sprint1 folder.

Description of USN and Screenshots:

USN-I:
As a user, I can register for the application by entering my name, email, user name,password, and confirming my password though registered mail id.

Screenshot:



28°C Partly cloudy

ENG IN 19:44 30-11-2022

IBM-Project-5048-1658746665/ Sprint 1.pdf

2 of 3

USN-2:
As a user, I will receive confirmation email once I have registered for the application.

USN-3:
As a user, I can log into the application by entering email/username & password.

Screenshot:



28°C Partly cloudy

ENG IN 19:44 30-11-2022

TEMPLATE CODE FOR LOGIN

login_style.css - Visual Studio Code

```
# login_style.css X
C:\Users\Chelsia stella>OneDrive>Desktop>IBM>Project Development phase>Sprint 1>Code># login_style.css > ...
1 @import url('https://fonts.googleapis.com/css?family=Roboto:300');
2
3 .body{
4     background-image: url('https://ordinaryfaith.net/wp-content/uploads/2016/03/Gray_plain_website-background.jpg');
5     background-repeat: no-repeat;
6     background-size: cover;
7 }
8
9 .login-page {
10    width: 360px;
11    padding: 0 0 0;
12    margin: auto;
13 }
14
15 .form {
16    position: relative;
17    z-index: 1;
18    background: #f8f8f8;
19    max-width: 360px;
20    margin: 0 auto 100px;
21    padding: 45px;
22    text-align: center;
23    box-shadow: 0 0 20px 0 rgba(0, 0, 0, 0.2), 0 5px 5px 0 rgba(0, 0, 0, 0.24);
24 }
25
26 .form input {
27   font-family: FontAwesome, "Roboto", sans-serif;
28   outline: 0;
29   background: #fff;
30   width: 100%;
31   border: 0;
32   margin: 0 0 15px;
33   padding: 15px;
34   box-sizing: border-box;
35   font-size: 14px;
36   border-radius: 10px;
37 }
```

Restricted Mode 0 0 0

28°C Partly cloudy

File Edit Selection View Go Run Terminal Help login_style.css - Visual Studio Code

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF CSS R Q

19:46 30.11.2022

login_style.css - Visual Studio Code

```
.container .info {
101   margin: 50px auto;
102   text-align: center;
103 }
104
105 .container .info h1 {
106   margin: 0 0 15px;
107   padding: 0;
108   font-size: 36px;
109   font-weight: 300;
110   color: #1a1a1a;
111 }
112 .container .info span {
113   color: #ddd;
114   font-size: 12px;
115 }
116 .container .info span a {
117   color: #000000;
118   text-decoration: none;
119 }
120 .container .info span .fa {
121   color: #EF3B3A;
122 }
123 body {
124   background: #76b852; /* fallback for old browsers */
125   background: -webkit-linear-gradient(right, #76b852, #80C26F);
126   background: -moz-linear-gradient(right, #76b852, #80C26F);
127   background: -o-linear-gradient(right, #76b852, #80C26F);
128   background: linear-gradient(to left, #76b852, #80C26F);
129   font-family: "Roboto", sans-serif;
130   -webkit-font-smoothing: antialiased;
131   -moz-osx-font-smoothing: grayscale;
132 }
```

Restricted Mode 0 0 0

28°C Partly cloudy

File Edit Selection View Go Run Terminal Help login_style.css - Visual Studio Code

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF CSS R Q

19:46 30.11.2022

SPRINT 2

IBM-Project-5048-1658746665/ Sprint 2.pdf

SPRINT 2: Classification of Arrhythmia by Using Deep Learning With 2-D ECG Spectral Image Representation

Team ID: PNT2022TMID27741

Team Members: Shakthi, Shruthi, Chelsia stella, Lavanya

Code: Updated in GitHub in the Deliverables section in Sprint 2 folder.

Description of USN and Screenshots:

USN-4:

As a user, I want quality data to be collected for the purposes of training the model. Also, image processing methods must be employed to pre-process the dataset.

Screenshot:

The PC - Desktop > IBM > data > test

Name Date modified Type Size

- Left Bundle Branch Block 08-11-2022 20:03 File folder
- Normal 08-11-2022 20:04 File folder
- Premature Atrial Contraction 08-11-2022 20:04 File folder
- Premature Ventricular Contractions 08-11-2022 20:04 File folder
- Right Bundle Branch Block 08-11-2022 20:04 File folder
- Ventricular Fibrillation 08-11-2022 20:04 File folder

28°C Party cloudy

IBM-Project-5048-1658746665/ Sprint 2.pdf

SPRINT 2: Classification of Arrhythmia by Using Deep Learning With 2-D ECG Spectral Image Representation

Team ID: PNT2022TMID27741

Team Members: Shakthi, Shruthi, Chelsia stella, Lavanya

Code: Updated in GitHub in the Deliverables section in Sprint 2 folder.

Description of USN and Screenshots:

USN-4:

As a user, I want quality data to be collected for the purposes of training the model. Also, image processing methods must be employed to pre-process the dataset.

Screenshot:

Image Split:

Left Bundle Branch Block – 504 images

Normal – 7436 images

Premature Atrial Contraction – 2054 images

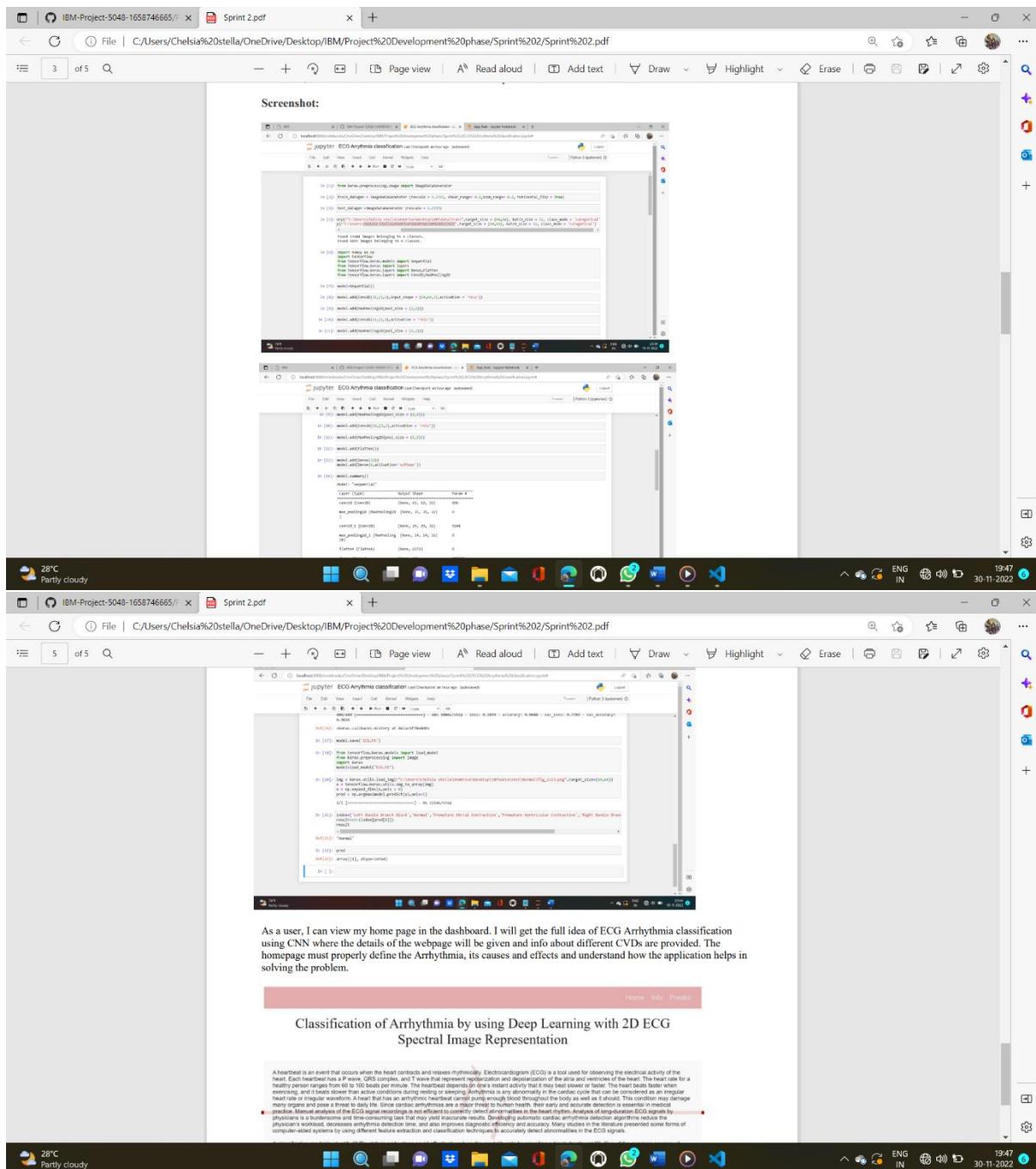
Premature Ventricular Contractions – 2759 images

Right Bundle Branch Block – 2239 images

Ventricular Fibrillation – 439 images

For reducing skewness in the dataset, ImageDataGenerator class was used for both processing and handling with data imbalance

28°C Party cloudy



SPRINT 4

IBM-Project-5048-1658746655 | Sprint 4.pdf | Sprint 4.pdf | +

File | C:/Users/Chelsia%20stella/OneDrive/Desktop/IBM/Project%20Development%20phase/Sprint%204/Sprint%204.pdf

1 of 3 Q Page view Read aloud Add text Draw Highlight Erase

SPRINT 4: Classification of Arrhythmia by Using Deep Learning With 2-D ECG Spectral Image Representation

Team ID: PNT2022TMID27741

Team Members: Shakthi, Shruthi, Chelsia stella, Lavanya

Code: Updated in GitHub in the Deliverables section in Sprint 4 folder.

Description of USN and Screenshots:

USN-6:

As a user, I can upload an ECG image and view the result. The type of Arrhythmia such as Left Bundle Branch Block, Normal, Premature Atrial Contraction, Premature Ventricular Contractions, Right Bundle Branch Block and Ventricular Fibrillation is displayed.

Screenshot:

ECG Arrhythmia Classification

28°C Partly cloudy

IBM-Project-5048-1658746655 | Sprint 4.pdf | Sprint 4.pdf | +

File | C:/Users/Chelsia%20stella/OneDrive/Desktop/IBM/Project%20Development%20phase/Sprint%204/Sprint%204.pdf

2 of 3 Q Page view Read aloud Add text Draw Highlight Erase

USN-7:

As a user, I can upload any ECG image and view the result. The algorithm is designed to denote the type of Arrhythmia such as Left Bundle Branch Block, Normal, Premature Atrial Contraction, Premature Ventricular Contractions, Right Bundle Branch Block and Ventricular Fibrillation.

The HTML file used to build the Info page includes:

```
index.html
...
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>ECG Arrhythmia Classification</title>
  </head>
  <body>
    <div>
      <form action="#" id="uploadForm" method="post" enctype="multipart/form-data">
        <label for="fileInput">Choose an ECG image</label>
        <input type="file" name="image" id="imageInput" accept="img, jpg, jpeg" />
      </form>
      <div class="image-section" style="display:none;">
        <div class="image-preview">
          <img alt="Image Preview" id="imgPreview" />
        </div>
        <div>
          <button type="button" class="btn btn-primary hvr-hover" id="tsr-predict">Predict</button>
        </div>
      </div>
      <div class="lower" style="display:none;"></div>
    </div>
  </body>
</html>
```

28°C Partly cloudy

IBM-Project-5048-1658746655 | Sprint 4.pdf | Sprint 4.pdf | +

File | C:/Users/Chelsia%20stella/OneDrive/Desktop/IBM/Project%20Development%20phase/Sprint%204/Sprint%204.pdf

3 of 3 Q Page view Read aloud Add text Draw Highlight Erase

The Python code behind classification of Arrhythmia :

```
app.py
...
from flask import Flask, render_template, request, send_file
app = Flask(__name__)

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

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/predict', methods=['GET', 'POST'])
def predict():
    if request.method == 'POST':
        f = request.files['file']
        basepath = os.path.dirname(__file__)
        filename = os.path.join(basepath, 'uploads', f.filename)
        f.save(filename) # Saving the file
        img = Image.open(imgfile, target=_blank, mode='r') # Read and reusing the image
        img = np.array(img) # Converting image to array
        predmodel = joblib.load('arrhythmia.pkl') # Loading the model
        predmodel.predict_classes() # Predicting classes
        print('Prediction:', pred) # Printing the prediction
        if pred == 0:
            result = 'Normal'
        elif pred == 1:
            result = 'Premature Atrial Contraction'
        elif pred == 2:
            result = 'Premature Ventricular Contractions'
        else:
            result = 'Right Bundle Branch Block'
    return result

if __name__ == '__main__':
    app.run(debug=True)
```

28°C Partly cloudy

TESTCASES REPORT

Testcases Report - Excel													
Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps to Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	Bug ID	Executed By
LoginPage_TC_OO_1	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button	Create HTML and CSS files to Login and Signup page to integrate with SQL for Database management.	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Signup popup displayed below the dropdown.	https://shopener.com/	Login/Signup popup should display	Working as expected	Pass	The Login/Signup page opens once the URL is clicked	Y	Nil	User 1 - Shakthi
LoginPage_TC_OO_2	UI	Home Page	Verify the UI elements in Login/Signup popup	The HTML page for Login/Signup popup contains User name, mail ID, Password.	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Signup popup with below UI elements: a.email text box b.password text box c.Login button d.New customer? Create account link e.last password? Recovery password link	https://shopener.com/	Application should show below UI elements: a.email text box b.password text box c.Login button with orange colour d.New customer? Create account link e.last password? Recovery password link	Working as expected	Pass	The user is able to login or signup. The confirmation mail is also received.	Y	Nil	User 2 - Shruthi
LoginPage_TC_OO_3	Functional	Home page	Verify user is able to log into application with Valid credentials	A user with valid mail ID and password can login without any errors.	1.Enter URL and click go 2.Click on My Account dropdown button 3.Enter Valid username/email in Email text box 4.Enter valid password in password text box 5.Click on login button	Username: chalam@gmail.com password: Testing123	User should navigate to user account homepage	Working as expected	Pass	The user was able to view the home page of the webpage.	Y	Nil	User 3 - Chelsia stella
LoginPage_TC_OO_4	Functional	Login page	Verify user is able to log into application with Invalid credentials	The webpage is designed to allow only the user with login credentials.	1.Enter URL(https://shopener.com/) and click go 2.Click on My Account dropdown button 3.Enter Invalid username/email in Email text box 4.Enter invalid password in password text box 5.Click on login button	Username: chalam@gmail.com password: Testing123	Application should show 'incorrect email or password' validation message.	Incorrect email or password	Pass	The validation box was successfully displayed when an user logs with invalid credentials.	Y	Nil	User 3 - Chelsia stella

Testcases Report - Excel													
Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps to Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	Bug ID	Executed By
LoginPage_TC_OO_4	Functional	Login page	Verify user is able to log into application with Invalid credentials	The webpage is designed to allow only the user with login credentials.	1.Enter URL(https://shopener.com/) and click go 2.Click on My Account dropdown button 3.Enter Invalid username/email in Email text box 4.Enter invalid password in password text box 5.Click on login button	Username: chalam@gmail.com password: Testing123	Application should show 'incorrect email or password' validation message.	Incorrect email or password	Pass	The validation box was successfully displayed when an user logs with invalid credentials.	Y	Nil	User 3 - Chelsia stella
LoginPage_TC_OO_4	Functional	Login page	Verify user is able to log into application with Invalid credentials	The webpage is designed to allow only the user with login credentials.	1.Enter URL(https://shopener.com/) and click go 2.Click on My Account dropdown button 3.Enter Invalid username/email in Email text box 4.Enter invalid password in password text box 5.Click on login button	Username: chalam@gmail.com password: Testing12367886786768768	Application should show 'incorrect email or password' validation message.	Incorrect email or password	Pass	The validation box was successfully displayed when an user logs with invalid credentials.	Y	Nil	User 3 - Chelsia stella
LoginPage_TC_OO_5	Functional	Login page	Verify user is able to log into application with Invalid credentials	The webpage is designed to allow only the user with login credentials.	1.Enter URL(https://shopener.com/) and click go 2.Click on My Account dropdown button 3.Enter Invalid username/email in Email text box 4.Enter invalid password in password text box 5.Click on login button	Username: chalam@gmail.com password: Testing12367886786768768	Application should show 'incorrect email or password' validation message.	An account with this user name already exists	Fail	The user has to choose a different user name	N	ERR	User 4 - Lavanya

ACCEPTANCE TESTING

The purpose of this document is to briefly explain the test coverage and open issues of the Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation project at the time of the release to User Acceptance Testing (UAT).

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	15	4	2	3	24
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	10	2	4	20	36
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	28	14	13	26	81

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51

Category	Count	Count	Count	Count
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2