

Project Planning Phase

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

| | |
|---------------|--|
| Date | 28 October 2022 |
| Team ID | PNT2022TMID26020 |
| 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

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|----------|---|-------------------|---|--------------|----------|-----------------------------------|
| Sprint-1 | Download The Dataset | USN-1 | We will download the Dataset contains Six classes | 2 | Low | Madheshwaran. R, Ajay.D |
| Sprint-1 | Import The Image Data Generator Library | USN-2 | We will import Image Data Generator | 1 | Low | Logesh.D, Rithick Roy.A.S |
| Sprint-1 | Configure Image Data Generator class | USN-3 | We will configure the Image Data Generator class | 1 | Low | Ajay.D, Logesh.D, Rithick Roy.A.S |
| Sprint-2 | Apply the Image Data Generator | USN-4 | We will apply Image Data Generator to train dataset | 2 | Medium | Madheshwaran. R, |

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|----------|---------------------------------------|-------------------|---|--------------|----------|---|
| | functionality to Trainset and Dataset | | | | | Ajay.D |
| Sprint-2 | Import Libraries | USN-5 | We will import required Libraries | 1 | Low | Madheshwaran.R Logesh.D |
| Sprint-2 | Initialize the Model | USN-6 | Initializing the Image recognition model | 2 | Medium | Rithick Roy.A.S, Logesh.D, |
| Sprint-4 | Adding CNN layer | USN-7 | We will add Convolutional Neural Network (CNN) used for image/object recognition and classification | 4 | High | Ajay.D, Madheshwaran.R |
| Sprint-4 | Adding Dense Layer | USN-8 | We will add Dense Layer in which each neuron receives input from all the neurons of previous layer | 4 | High | Rithick Roy.A.S, Ajay.D, Madheshwaran.R |
| Sprint-3 | Configure The Learning Process | USN-9 | We will configure The Learning process which is a method, mathematical logic or algorithm that improves the network's performance and/or training time. | 3 | Medium | Logesh.D Ajay.D, Madheshwaran.R |
| Sprint-4 | Train the Model | USN-10 | We will train our model with our image dataset. Fit generator functions used to train a deep learning neural network | 4 | High | Madheshwaran.R, Logesh.D |
| Sprint-4 | Save the Model | USN-11 | We will save The model with .h5 extension | 2 | Medium | Ajay.D,' Rithick Roy.A.S |
| Sprint-4 | Test the model | USN-12 | We will Test the model through Loaded necessary libraries, the saved model | 3 | Medium | Madheshwaran.R, Logesh.D |

| 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 | Ajay.D, Logesh.D |
| 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. | 4 | High | Madheshwaran. R, Rithick Roy.A.S |
| Sprint-4 | Run the App | USN-15 | We can run the App | 3 | Medium | Ajay.D, Madheshwaran.R |
| Sprint-1 | Register IBM Cloud | USN-16 | We can register IBM Cloud | 2 | Medium | Madheshwaran.R, Logesh.D, Ajay.D, Rithick Roy.A.S |
| Sprint-3 | Train the model on IBM | USN-17 | We can Train Out model on IBM | 3 | High | Logesh.D, Madheshwaran.R, Ajay.D |

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 | 29 Oct 2022 |
| Sprint-2 | 20 | 5 Days | 30 Oct 2022 | 04 Nov 2022 | 20 | 4 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

$$Av = \frac{Velocity}{Sprint\ duration}$$

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