**Project Planning Phase**

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

|  |  |
| --- | --- |
| Date | 29 October 2022 |
| Team ID | PNT2022TMID09796 |
| 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 can download the Dataset contains Six classes | 4 | Low | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-1 | Import The  Image Data Generator  Library | USN-2 | We can import Image Data Generator | 4 | Low | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-1 | Configure  Image Data Generator class | USN-3 | We can configure the ImageDataGenerator class | 6 | Medium | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-1 | Apply the  Image Data Generator functionality to Train Set and Dataset | USN-4 | We can apply ImageDataGenerator to train dataset | 6 | Medium | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-2 | Import Libraries | USN-5 | We can import required Libraries | 1 | Low | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |

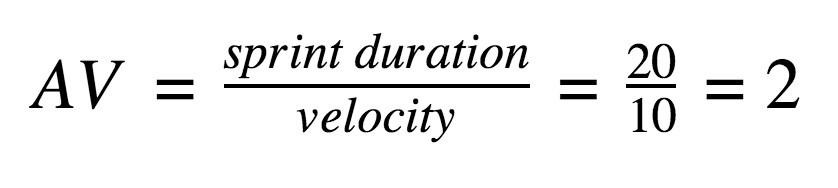
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Functional**  **Requirement (Epic)** | **User Story**  **Number** | **User Story / Task** | **Story Points** | **Priority** | **Team**  **Members** |
| Sprint-2 | Initialize the Model | USN-6 | Initializing the Image recognition model | 2 | Medium | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-2 | Adding CNN layer | USN-7 | We can add Convolutional Neural  Network(CNN) used for image/object recognition and classification | 3 | High | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| 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 | 2 | High | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| 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 | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| 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 | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-2 | Save the Model | USN-11 | We can save The model with .h5 extension | 2 | Medium | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-2 | Test the model | USN-12 | We can Test the model through Loaded necessary libraries, the saved model | 2 | Medium | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-3 | Create Html files | USN-13 | We use HTML to create the front end part of the web page. | 8 | High | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| 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 | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| **Sprint** | **Functional**  **Requirement (Epic)** | **User Story**  **Number** | **User Story / Task** | **Story Points** | **Priority** | **Team**  **Members** |
| Sprint-3 | Run the App | USN-15 | We can run the App | 4 | Medium | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-4 | Register in IBM Cloud | USN-16 | We can register in the cloud | 8 | Medium | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |
| Sprint-4 | Train the model on IBM | USN-17 | We can train the model on IBM | 12 | High | Aasha Sapkota  Rinshu Kumari  Ritu Jha  Nitu Jha |

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

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Sprint** | **Total Story Points** | **Duration** | **Sprint Start Date** | **Sprint End Date (Planned)** | **Story Points**  **Completed (as on**  **Planned End Date)** | **Sprint Release Date (Actual)** |
| Sprint-1 | 20 | 6 Days | 24 Oct 2022 | 29 Oct 2022 | 20 | 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 |

**Velocity:**

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)



**Average Velocity = Story Points per Day**

**Sprint Duration = Number of (Duration) days per Sprint**  **Velocity = Points per Sprint**

# 𝟐𝟎

**AV =**

## Therefore, the AVERAGE VELOCITY IS 4 POINTS PER SPRINT

**Burndown Chart:**

