**Project Planning Phase**

**Project Planning Template (Milestone Activity)**

|  |  |
| --- | --- |
| Date | 15 NOV 2022 |
| Team ID | PNT2022TMID29172 |
| Project Name | Real-Time Communication System Powered By AI For Specially Abled |

Milestone Activity Plan.

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone** | **Function (Epic)** | **Milestone Story Number** | **Story / Task** |
| **Milestone** 1 | Data collection | M1 | we're collecting dataset for building our project and creating two folders, one for training and another one for testing. |
| **Milestone 2** | Image preprocessing | M2 | Importing image data generator libraries and applying image data generator functionality to train the test set. |
| **Milestone 3** | Model building | M3 | Importing the model building libraries, Initializing the model, Adding Convolution layers, Adding the Pooling layers, Adding the Flatten layers, Adding Dense layers, Compiling the model Fit and Save the model. |
| **Milestone 4** | Testing the model | M4 | Import the packages first. Then we save the model and Load the test image, preprocess it and predict it. |
| **Milestone 5** | Application layer | M5 | Build the flask application and the HTML pages. |
| **Milestone 6** | Train CNN model | M6 | Register for IBM Cloud and train Image Classification Model. |
| **Milestone 7** | Final result | M7 | To ensure all the activities and resulting the final output. |

|  |  |  |
| --- | --- | --- |
|  | **Technolo gy**    **MySQL & NoSQL** |  |
|  |  |  |
|  | **JAVA**  **& PYTHO N** |  |
|  |  |  |
| **Machine Learning, CNN & RNN** |
|  |



**Train CNN model**

**Application layer**

**Testing the model**

**Model Building**

**Image preprocessing**

**Data Collection**

**Load the test model image, preprocessing and predict it.**

**Importing the model building libraries**

**Importing the image libraries**

**Collecting dataset for building project**

**Ensure all activities & getting final output**

**Final Result**

Register for train image classification

Building flask applicatio