SPRINT- 1

TEAM ID	PNT2022TMID27588
PROJECT NAME	Real-Time Communication System Powered by AI for Specially Abled
DATE OF THE MEETING	26 -10-2022

MINUTES OF THE MEETING:

- The process of data collection was discussed and procedure was implemented in the local system.
- The training images were tested and verified.
- The image processing procedure was detailly examined and discussed .
- The required python modules were installed and issues in the model were rectified .

Data collection:

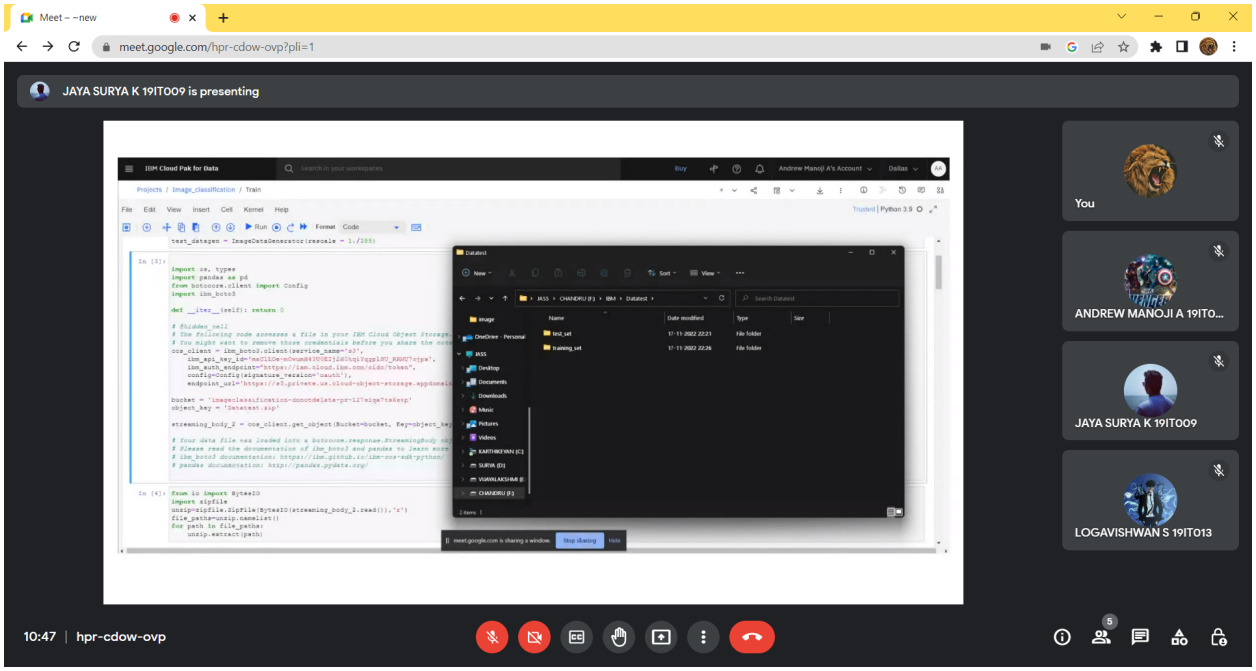
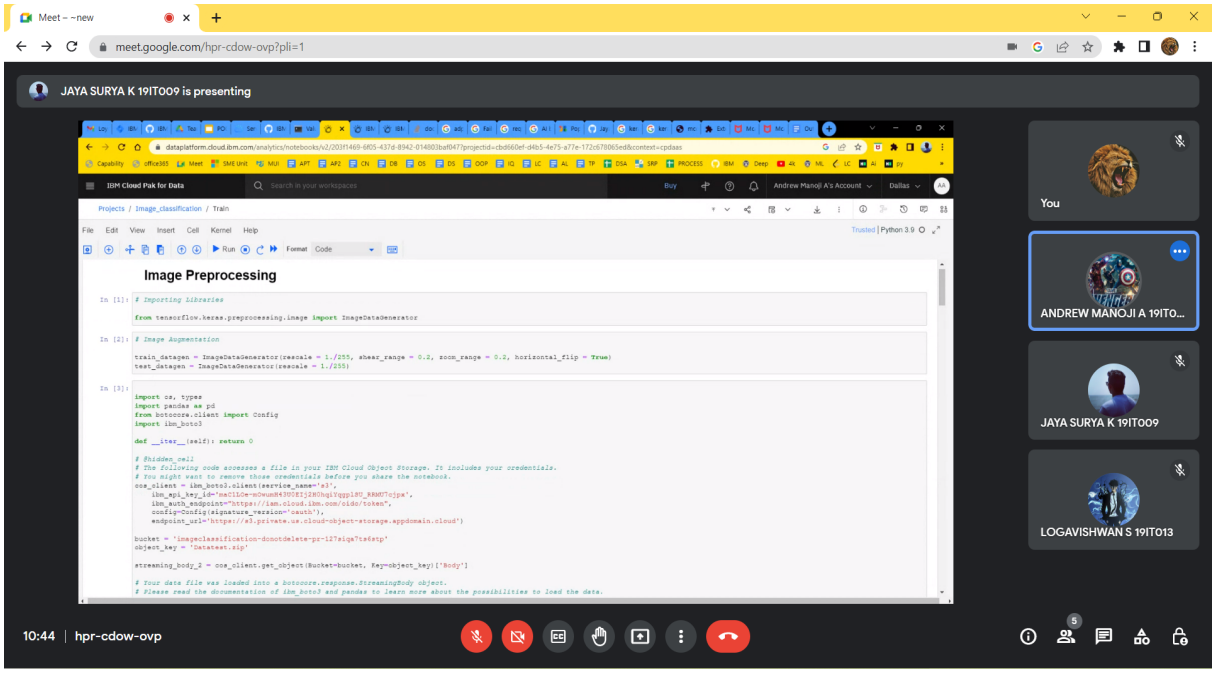


Image preprocessing:



SPRINT– 2

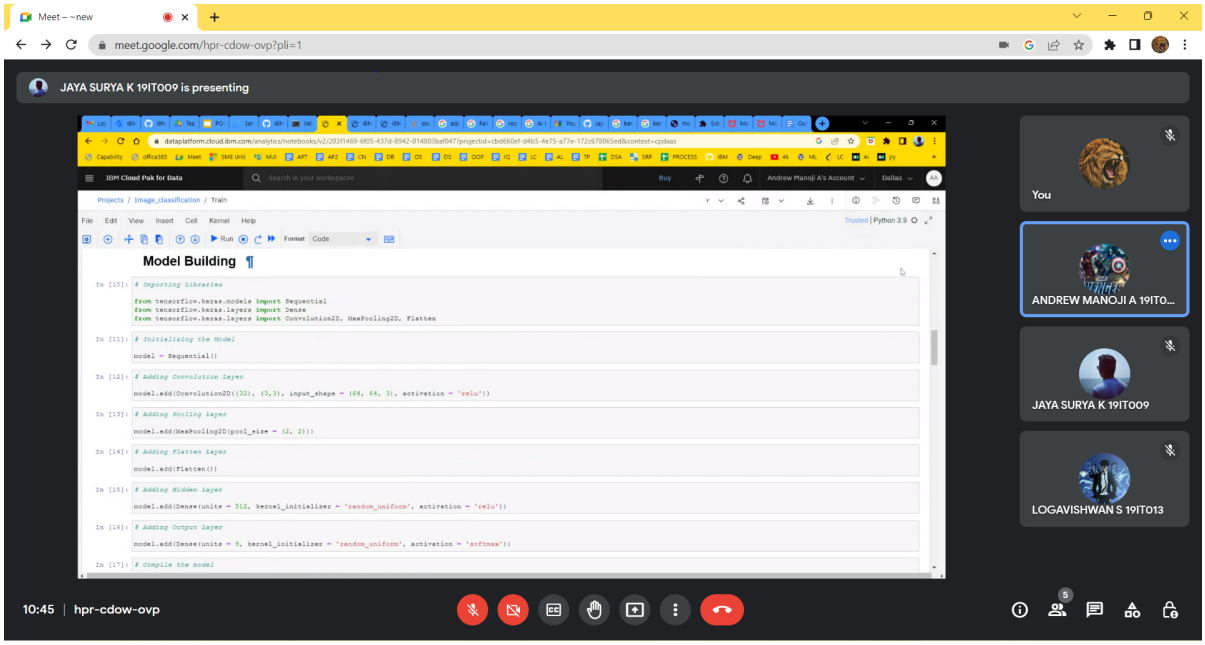
TEAM ID	PNT2022TMID27588
PROJECT NAME	Real-Time Communication System Powered by AI for Specially Abled
DATE OF THE MEETING	02-11-2022

MINUTES OF THE MEETING:

- **The model building of the module was discussed and implemented in the Ibm cloud.**
- **The required packages were installed and imported .**
- **The issues in installation of packages were cleared.**

- The dataset was imported in Ibm cloud and it was unzipped using python code.
- The model was trained using the given training images in the dataset.

Model Building and training:



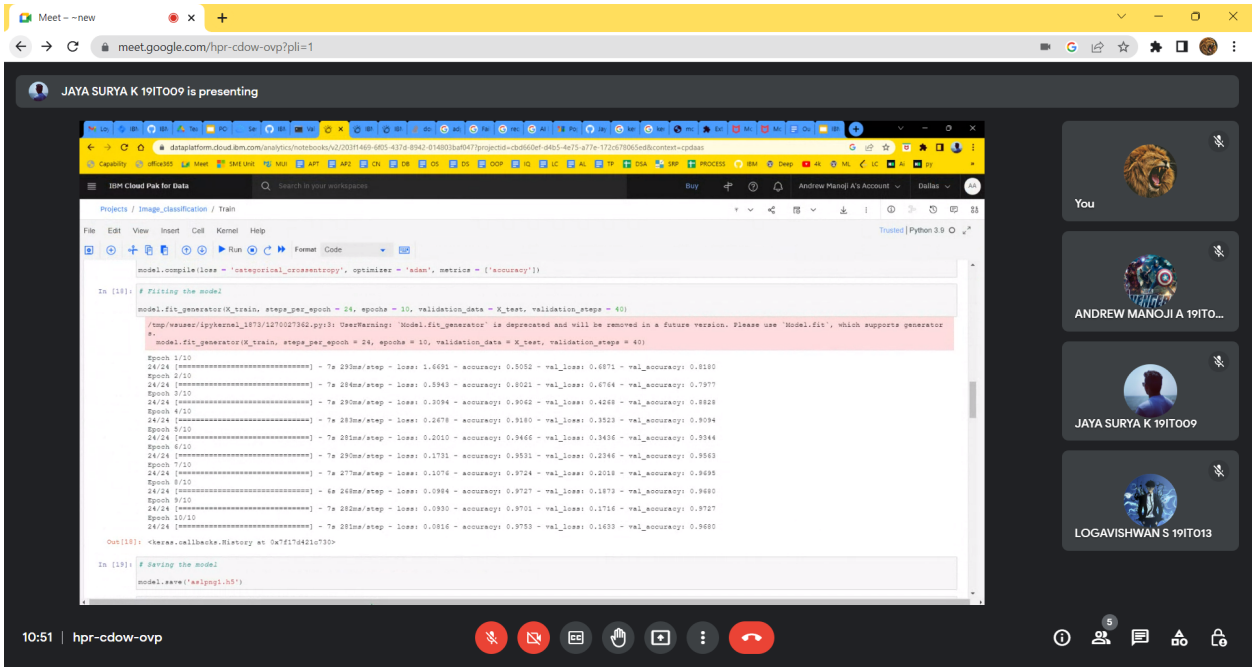
SPRINT– 3

TEAM ID	PNT2022TMID27588
PROJECT NAME	Real-Time Communication System Powered by AI for Specially Abled
DATE OF THE MEETING	7-11-2022

MINUTES OF THE MEETING:

- The model trained in the earlier sprint was saved and loaded using the model.load() function.
- The loaded model was tested using the sample images provided in the testset of the dataset packages.
- The errors in the testing the model was rectified and the correctness of the model was improved.

Training and testing:



SPRINT– 4

TEAM ID	PNT2022TMID27588
PROJECT NAME	Real-Time Communication System Powered by AI for Specially Abled
DATE OF THE MEETING	14-11-2022

MINUTES OF THE MEETING:

- The train and test model was integrated with an HTML page with Flask application.
- The flask application development was discussed and done with video reference provided in the Ibm cloud.
- The API routes were built in the flask application and the python code for the opencv module was built under the prediction route.
- The model was loaded and it was made to read the real time human signals with the help of opencv2 and keras.
- The output was then tested and verified.

Implementation of the application: