# Insight App with Smart Video Distortion Correction

Microsoft Engage 2021

## ❖ Project

### ➤ Process:

- **SCRUM**, a subset of Agile, framework was used for developing this Project.

- Time period of 01 month given for the completion of the project was divided into 4 Sprints, each consisting of 1 week.

- The goals, requirements and planning for the week was decided on the first day of each week and the **tasks were broken and scheduled** in advance.

- At the start of each day, last day's progress was noted and backlogs were carried forward to this day's tasks. The activities of the rest of the day was planned.

- At the end of the each day, goals achieved were marked and any mistake made was noted down so that we never repeat it.

- At the end of each week, the progress of the week was noted. The **web app was kept ready to be hosted** with whatever features we had worked on so far.

- Demonstration:
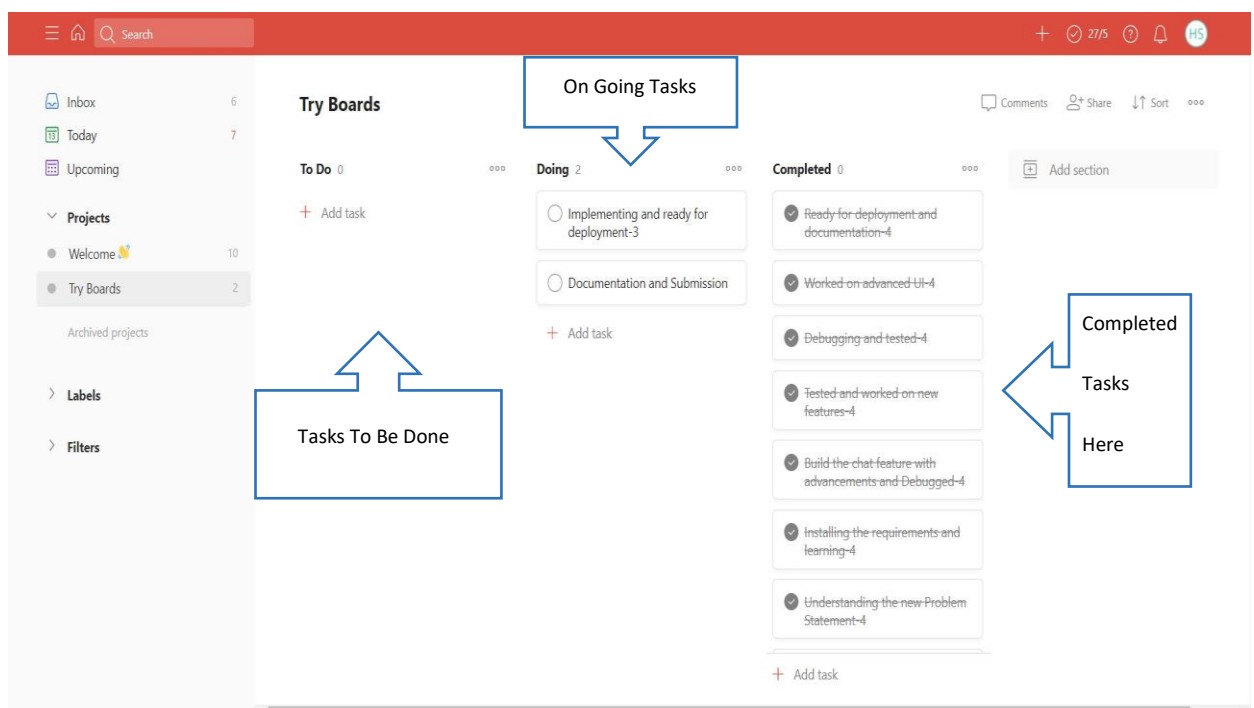
# ❖ Special Feature: <u>Video Distortion Correction</u>

## ➢ Idea:

A Machine Learning Model has been implemented to **correct the distorted video in the recording**, due to poor internet connection during the meeting.

## ➢ Libraries:

Pytorch and Tensorflow in Python

## ➢ Description:

Usually **during a meeting due to poor connection**, a user in the meeting faces video distortion but his audio is clear, in that case the model would correct that part of the video to give a seamless experience to anyone watching the recording of the meeting.

This was done using the Wav2Liv Deep Fake algorithm using the Generative Adversarial Networks (GAN's) in Deep Learning. It would pick video from a few seconds before the user encountered the distortion. When the user faces distortion, the model would pick the audio from that part. When the meeting ends, it would generate accurate lip syncing for that audio and implement it in the lips of the user in the video recorded just before he faced the distortion. This would result in a seamless final recorded video of the meeting.

## ➢ Algorithm:

Wav2lip and GANs and is trained on the Oxford-BBC Lip Reading Sentences 2 (LRS2) Dataset.



*Research Paper: https://arxiv.org/pdf/2008.10010.pdf*
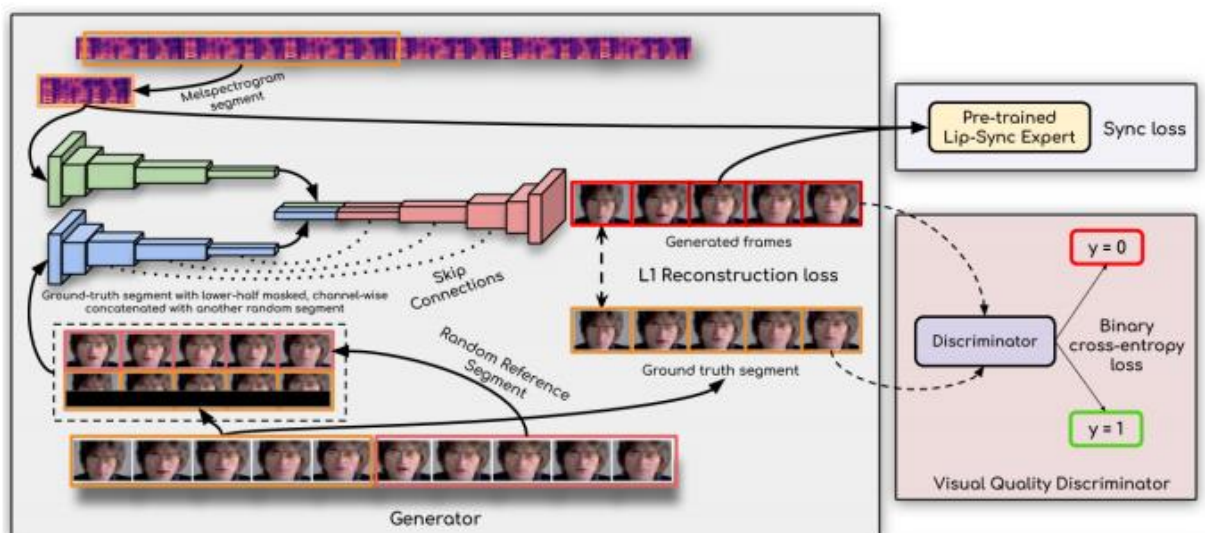
## ❖ Product

### ➢ Objective:

- To create a Video Conference Web application to at least connect 2 people.

- Chat which can be started before the meeting, will continue during and after the meeting ends.

### ➢ Languages:

- ReactJS

- NodeJS

- ExpressJS

- Html

- CSS

- Python

### ➢ Libraries and Dependencies:

- React Chat Engine

- Webrtc

- Socket.io

- React-dom

- Peerjs

### ➢ Features:

- Video Conference 1:1

- Invite people

- Live Chat

- Authentication

- Registration and Login

- Group Chat up to 25 people which stays before and after the Meeting

- Show active status

- Set display Picture or Avatar

- See who read the message

- Share Images in chat

- See shared images separately

- See people present in the group

- Make new groups

- Delete members from groups

- Delete group

## ❖ Future of the Application

➢ The web application will be hosted using Azure App Services to work with group video calls and multiple servers simultaneously and with the Video distortion correction feature explained above.