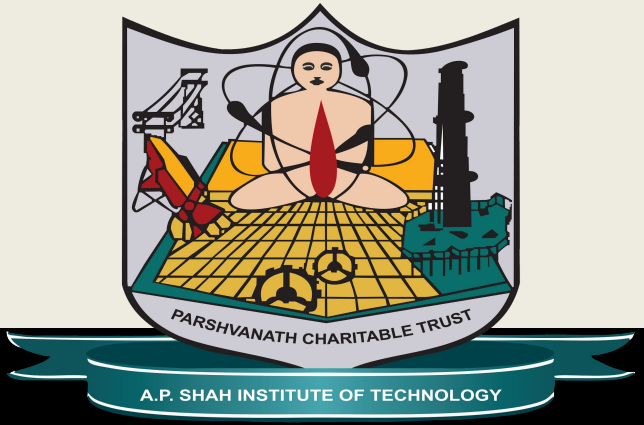


Department of Information Technology
NBA Accredited

A.P. Shah Institute of Technology

G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic Year 2022-2023

A Project Report on

Designing ML Based Comprehensive Web Framework for Soft Skills Enhancement

Submitted in partial fulfillment of the degree

in

INFORMATION TECHNOLOGY

By

Akshay Bura(19104041)

Sourav Joshi(19104068)

Sakshi Deshpande(19104002)

Under the Guidance of

Mrs. Rujata Chaudhari

1.Project Conception and Initiation

1.1 Abstract

- Many students lack confidence and verbal inability in social environment like placement drives and that may affect their job interviews.
- There are abundant of sources and mediums to detect lack of self-confidence but most of them don’t help in building confidence.
- Since we will be automating ‘Xpression Club’ activity which was introduced by our organization.
- We are providing a platform where all these problems can be solved.

1.2 Objectives

- To automate Xpression Club (Xpression Club is an initiative which was introduced by our organization to boost the confidence level of students).
- To Improve logical thinking and verbal ability of the students.
- To predict the confidence level of students and help them in improve it.
- To create an user friendly UI using flutter.
- To perform feature extraction on the video using mediapipe.
- To Predict the confidence in speech using NLP.

1.3 Literature Review

Sr. No.	Title	Key Findings	Year
1.	Multimodal Emotion Recognition.	Here they are providing definitions of affective computing and multimodal sentiment analysis.	2020
2.	Emotion Recognition Based on Facial Expressions Using Convolutional Neural Network (CNN).	They tried different parameters and architectures of the CNN in order to detect the seven emotions in human faces.	2020
3.	BLSTM-Based Confidence Estimation for End-to-End Speech Recognition.	To tackle the imbalance dataset problem, they employed BLSTM-model as a strong binary-class sequence labeler.	2021

1.4 Problem Definition

- Nowadays students are facing lack of confidence while speaking to others. Most of the apps on market do not provide a solution to anxiety and verbal ability. The apps that do try are costly and their free versions don't have the features. They also contain lots of advertisements.
- Most of the apps use a Support Vector Machine (SVM) algorithm to predict the confidence of the person. The accuracy of SVM algorithm is low, and this may lead to false predictions that can lower the self-esteem of the user.
- Apps in the market only do activities which might tackle anxiety and low-esteem, like breathing and meditating, which solve problem temporary. They also charge a couple of dollars in the form of subscriptions.

1.5 Scope

- Many companies notice employers who got placed in their company are good in technical skills but lack in communication skills. So to overcome this problem our institute had started an activity named Xpression Club. Our project automates the process of Xpression Club.
- Can be useful in institutes or colleges to improve self confidence and soft skills of students.
- Can be useful in rural areas where psychiatrist can not be reached.

1.6 Technology stack

- **Front End**
  - Flutter 3.3.2
  - Dart 2.18.1
- **Back End**
  - AI / ML
    - MediaPipe - To Identify the Posture
    - NLP - To Predict the confidence in speech
  - Cloud (Google) - To store data on cloud

1.7 Benefits for environment & Society

- Developing a comprehensive application to enhance soft skills using artificial intelligence and machine learning which can reduce costs, and increase efficiency.
- The application can enhance soft skills which is useful in our day to day life.
- The benefits can be applied to educational institutions and rural areas.

2. Project Design

2.1. Proposed System

Functions of the Proposed System are :

- User Authentication
- Attempt Quiz
- Upload Video
- Video Analysis
- Speech Analysis
- Perform Activities

2.2. Design(Flow Of Modules)

- User needs to create an account in the application.
- Then user has to upload a video, and ML model will give score based on the posture and the way user speaks in video.
- Then user needs to attempt a Quiz.
- Scores generated by a trained ML model and the Quiz will be stored in the cloud and these scores will be used to assign users to different confidence levels : Low, Medium, and High.
- Each level will have various activities, including logical, verbal, and memory games, to enhance users' soft skills.
- High Confidence users will receive a certificate after completion.

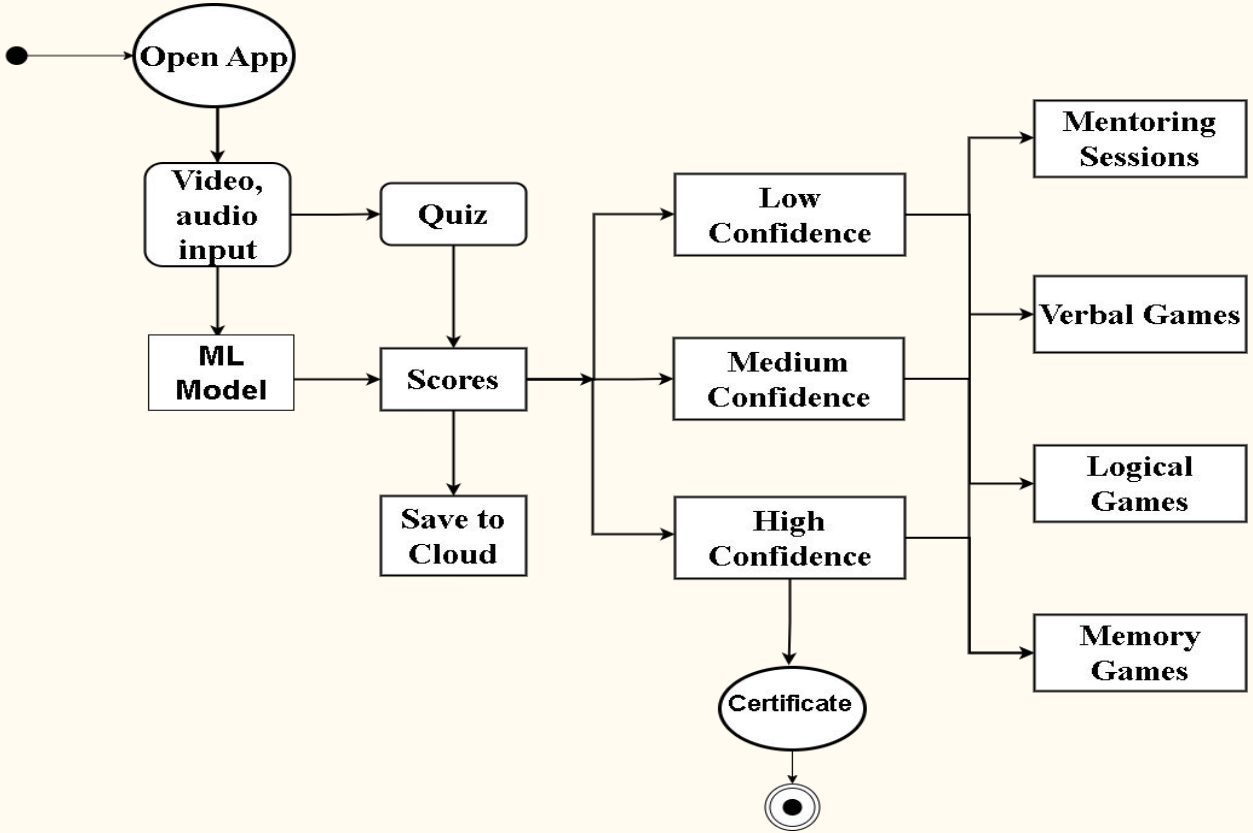
2.3. Description Of Use Case

The use case diagram shows how user will interact with the application. The user has three roles :

- Log In
- Upload Video
- Attempt Quiz

After giving Quiz & Uploading Video user will get aggregate score. This score will be stored in the backend, and based on the score, user will be assigned to specific confident level. User can perform various activities which helps in enhancing their confidence.

2.4. Activity diagram



3. Implementation

```python
import cv2
import os
import mediapipe as mp
import numpy as np
import base64
from moviepy.video.io.VideoFileClip import VideoFileClip
import speech_recognition as sr
import requests
import flask from Flask,requrst

app=Flask(__name__)

speech_score = 0


mp_drawing = mp.solutions.drawing_utils
mp_pose = mp.solutions.pose

```

```
filename = "output.mp4"

@app.route('/',methods=['POST'])
def confidence():
    video_base64=request.json['base64']
    with open('output.mp4', 'wb') as f:
        video_data = base64.b64decode(video_base64)
        f.write(video_data)
    cap = cv2.VideoCapture(filename)


    cv2.namedWindow("Sitting Posture")


    while cap.isOpened():
        success, image = cap.read()
        if not success:
            print("End of video.")
            break
```

```
        image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)

        with mp_pose.Pose(min_detection_confidence=0.5, min_tracking_confidence=0.5) as pose:
            results = pose.process(image)

            image = cv2.cvtColor(image, cv2.COLOR_RGB2BGR)
            mp_drawing.draw_landmarks(image, results.pose_landmarks, mp_pose.POSE_CONNECTIONS)

            left_hip = results.pose_landmarks.landmark[mp_pose.PoseLandmark.LEFT_HIP]
            right_hip = results.pose_landmarks.landmark[mp_pose.PoseLandmark.RIGHT_HIP]
            left_knee = results.pose_landmarks.landmark[mp_pose.PoseLandmark.LEFT_KNEE]
            right_knee = results.pose_landmarks.landmark[mp_pose.PoseLandmark.RIGHT_KNEE]
            left_ankle = results.pose_landmarks.landmark[mp_pose.PoseLandmark.LEFT_ANKLE]
            right_ankle = results.pose_landmarks.landmark[mp_pose.PoseLandmark.RIGHT_ANKLE]
            score = 0
            if left_hip.y < left_knee.y < left_ankle.y and right_hip.y < right_knee.y < right_
                score += 50
                finalscore.append(score)
            if abs(left_knee.x - left_hip.x) < abs(right_knee.x - right_hip.x):
```

```
                score += (left_knee.x - left_hip.x) * 50 // (right_knee.x - right_hip.x)
                finalscore.append(score)
            cv2.putText(image, f"Sitting Score: {score}", (50, 50), cv2.FONT_HERSHEY_SIMPLEX,

            cv2.imshow("Sitting Posture", image)

        if cv2.waitKey(1) == ord('q'):
            break

    avg = round(np.mean(finalscore),1)
    print(avg)
    cap.release()
    cv2.destroyAllWindows()


    video = VideoFileClip(filename)

    audio = video.audio
```

```
78      audio.write_audiofile("output.wav")
79
80
81      r = sr.Recognizer()
82
83      audio_file = sr.AudioFile("output.wav")
84
85      with audio_file as source:
86          audio = r.record(source)
87          transcript = r.recognize_google(audio)
88
89      print(transcript)
90
91      api_key = 'a6558ffdf5mshb040b6de6a19840p135346jsnba7122a2de16'
92
93      text = transcript
94
95
96      url = 'https://api.languagetool.org/v2/check'
```

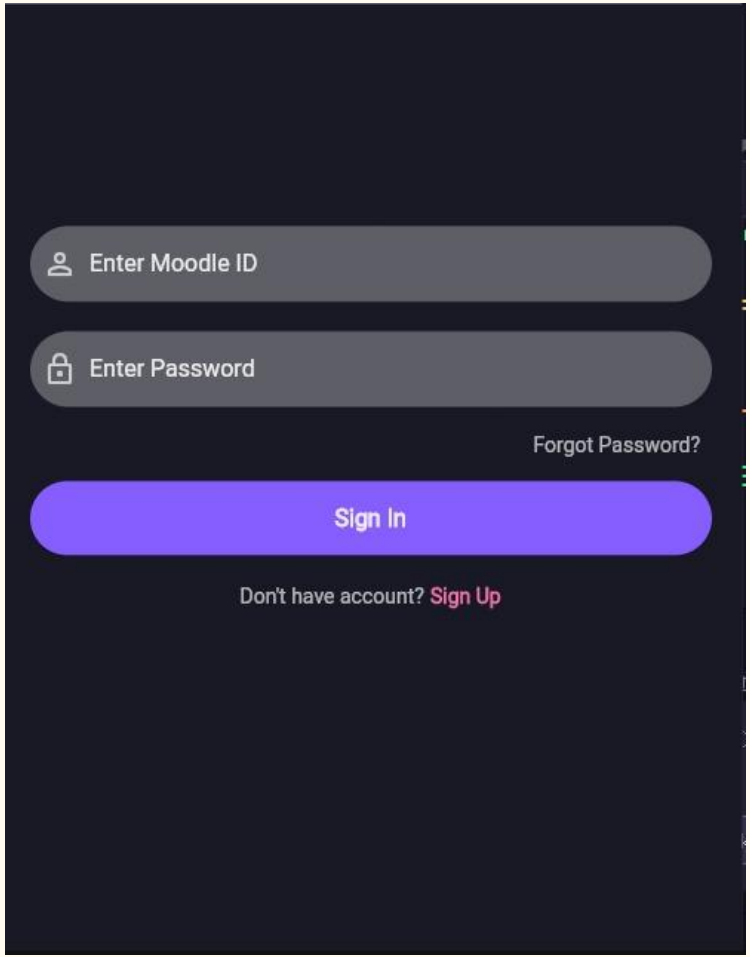
```
78      audio.write_audiofile("output.wav")
79
80
81      r = sr.Recognizer()
82
83      audio_file = sr.AudioFile("output.wav")
84
85      with audio_file as source:
86          audio = r.record(source)
87          transcript = r.recognize_google(audio)
88
89      print(transcript)
90
91      api_key = 'a6558ffdf5mshb040b6de6a19840p135346jsnba7122a2de16'
92
93      text = transcript
94
95
96      url = 'https://api.languagetool.org/v2/check'
```

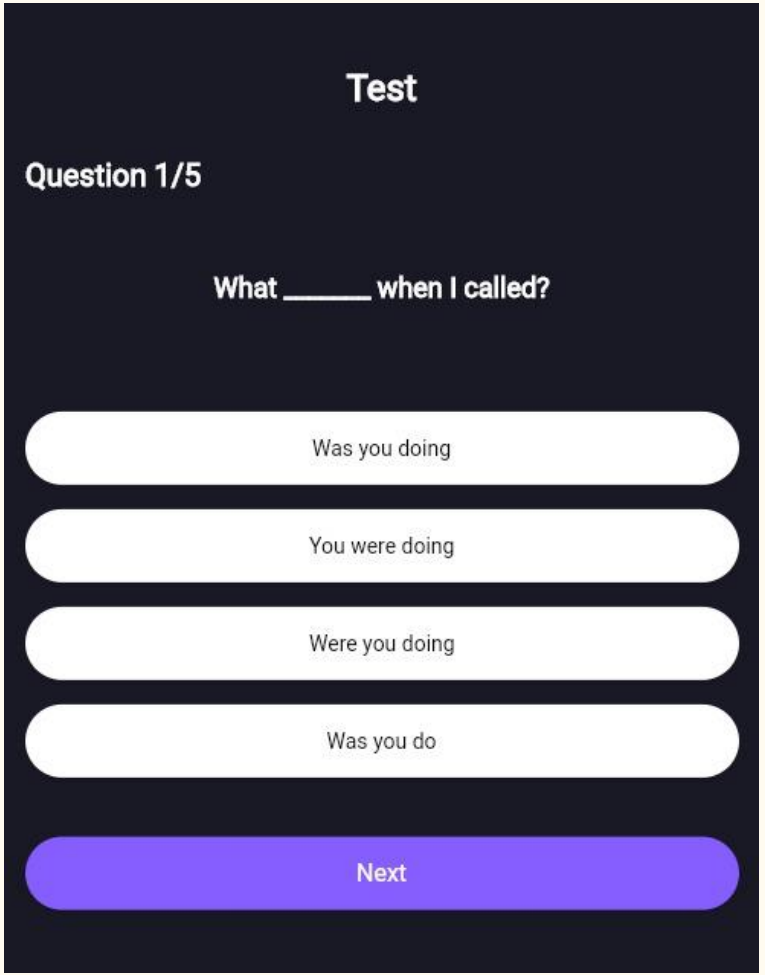
```
    url = 'https://api.languagetool.org/v2/check'
    data = {
    'text': text,
    'language': 'en-US'
    }
    headers = {
    'Content-Type': 'application/x-www-form-urlencoded',
    'Accept': 'application/json',
    'Authorization': f'Bearer {api_key}'
    }
    response = requests.post(url, data=data, headers=headers)


    json_data = response.json()
    matches = json_data['matches']
    if len(matches) > 0:
        print(f'{len(matches)} grammar error(s) found:')
        for match in matches:
```

```
            message = match['message']
            offset = match['offset']
            length = match['length']
            error_text = text[offset:offset+length]
        print(f'- {message}: {error_text}')
        if (len(matches) <= 1):
            speech_score = 85
            print("Person is confident")
        elif(len(matches) <= 4):
            speech_score = 60
            print("Person is low confident")
        elif(len(matches) > 7):
            speech_score = 40
            print("Person is not confident at all")

        finalscore = (avg + speech_score) / 2
        return({"Video":avg,"Audio":speech_score,"Final":finalscore})
```


Enter Moodle ID
Enter Password
Forgot Password?
Sign In
Don't have account? Sign Up


Test
Question 1/5
What _____ when I called?
Was you doing
You were doing
Were you doing
Was you do
Next

Select Video

Selected video: output.mp4

Upload Video

4. Testing

4.1. Functional Testing

Sr. No.	Video File	Expected Result	Actual Result	Pass or Fail
1.	Output.mp4	Confident	Confident	Pass
2.	Output.mp4	Less Confident	Less Confident	Pass
3.	Output.mp4	Confident	Confident	Pass

4.1. Test Case: ML Model Testing

4.2. Integration Testing

Sr. No.	Test Name	Expected Result	Actual Result	Pass or Fail
1.	Empty MoodleID Field	Enter MoodleID	MoodleID cannot be Empty	Pass
2.	Empty Password Field	Enter Password	Password cannot be empty	Pass
3.	User already Exists	Enter MoodleID	Moodle ID is already registered	Pass

4.2. Test Case : Sign Up Page

Sr. No.	Test Name	Expected Result	Actual Result	Pass or Fail
1.	Empty MoodleID Field	Enter MoodleID	MoodleID cannot be Empty	Pass
2.	Empty Password Field	Enter Password	Password cannot be empty	Pass
3.	Incorrect Password	Enter Password	Password doesn’t match	Pass

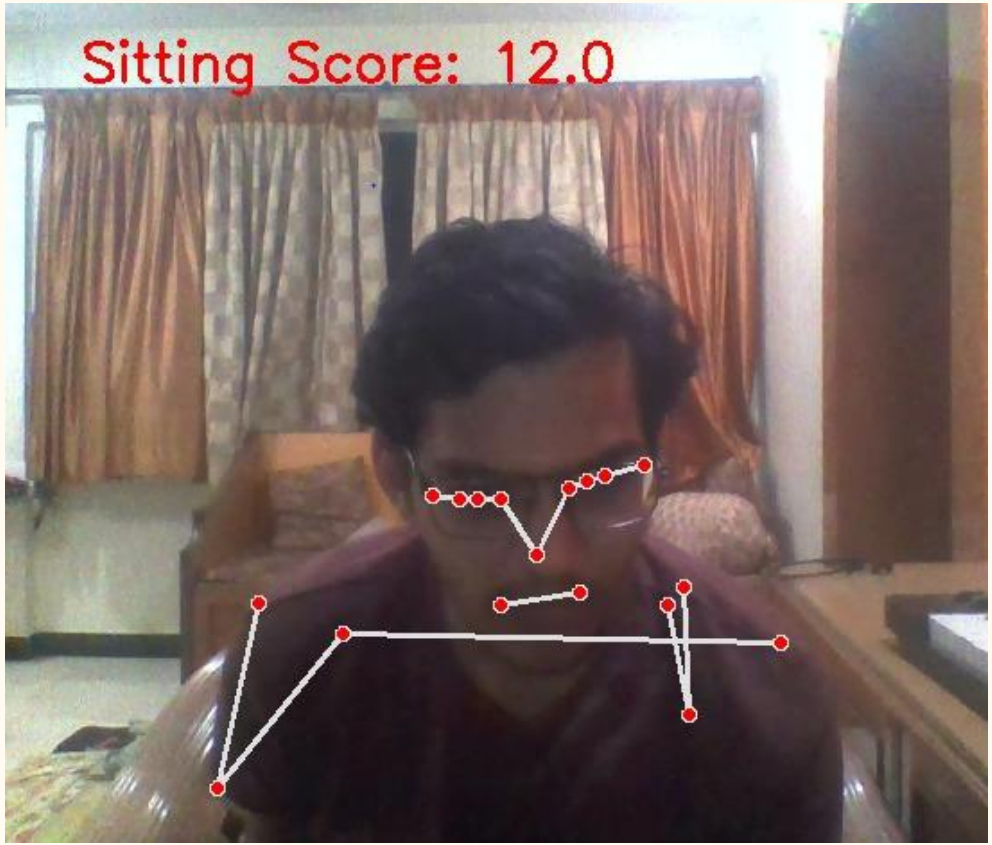
4.3. Test Case : Sign In Page

Sr. No.	Test Name	Expected Result	Actual Result	Pass or Fail
1.	Wrong Format File	Upload Video	Invalid File Format	Pass
2.	Empty Field	Upload Video	Please Upload MP4 File	Pass

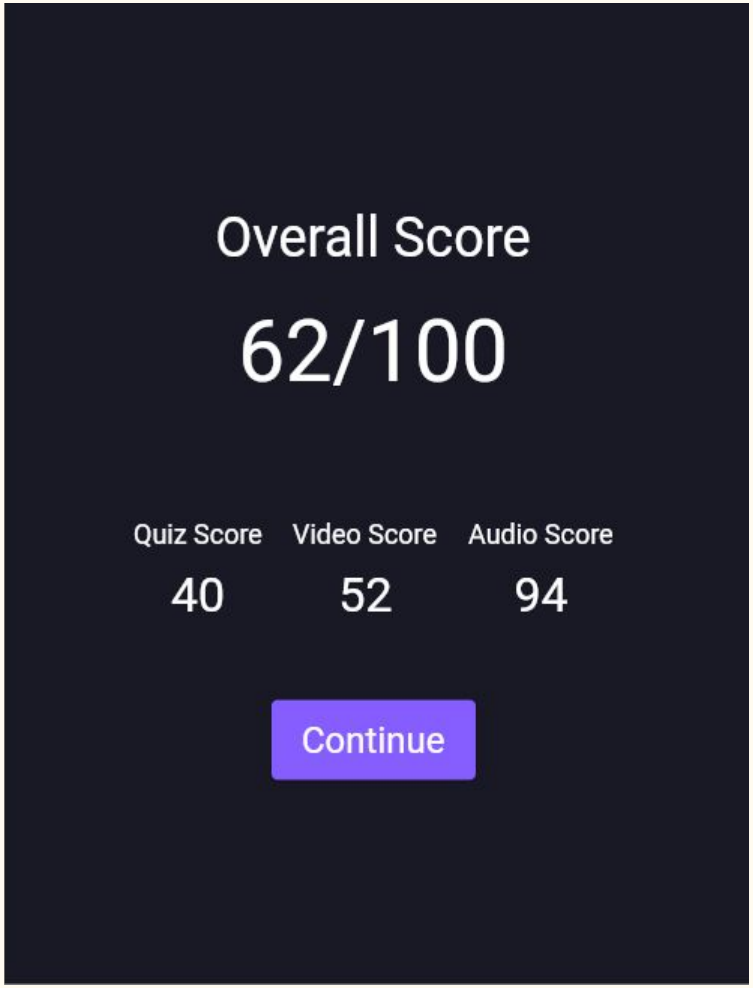
4.4. Test Case : Video Uploading Page

5. Result


```
Person is confident
>>>
```




```
Person is low confident
>>>
```


Overall Score
62/100
Quiz Score
Video Score
Audio Score
40
52
94
Continue

6. Conclusion and Future Scope

Conclusion

Our primary objective was to empower students to express themselves with confidence in front of a large audience. To achieve this goal, we had developed a cross-platform application that will enhance students' communication abilities and self-assurance. We believe that this project has the potential to be a valuable resource for colleges seeking to help their students improve their self-confidence.

Future Scope

In future, the application could incorporate more sophisticated machine learning algorithms and natural language processing to provide personalized feedback and recommendations to each student based on their unique communication style and learning preferences. Another potential direction is to integrate the application with virtual or augmented reality technologies to provide students with realistic simulated environments in which to practice and develop their communication skills.

References

1. Wei, Gou, Li Jian, and Sun Mo. ”Multimodal (audio, facial and gesture) based emotion recognition challenge.” 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020). IEEE, 2020.
2. Begaj, Sabrina, Ali Osman Topal, and Maaruf Ali. ”Emotion Recognition Based on Facial Expressions Using Convolutional Neural Network (CNN).” 2020 International Conference on Computing, Networking, Telecommunications & Engineering Sciences Applications (CoNTESA). IEEE, 2020.
3. A. Ogawa, N. Tawara, T. Kano and M. Delcroix, ”BLSTM-Based Confidence Estimation for End-to-End Speech Recognition,” ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 2021, pp. 6383-6387, doi: 10.1109/ICASSP39728.2021.9414977

Thank You