# DEPARTMENT OF APEX INSTITUTE OF TECHNOLOGY

## PROJECTPROPOSAL

## 1. Project Title: -Voice Emotion Recognition for Call Centres

## 2. Project Scope: -

### INTRODUCTION:

Emotion recognition using speech is a cutting-edge machine learning application with enormous potential for improving a variety of businesses, including call centres, mental health care, and market research. This project intends to create a robust emotion detection system that uses machine learning techniques to analyse audio data and infer the speaker's emotional state. The project comprises two major phases: model development and real-time emotion detection implementation.

**Phase 1: Developing an Emotion Detection Model using Voice Data**

The major goal of this research is to create a robust and reliable emotion recognition model based on speech data. During Phase 1, we train and test the machine learning model on labelled audio data to detect emotions including anger, pleasure, sorrow, and neutrality. This phase consists of the following main steps:

1. **Data collection and preprocessing:**
   We collect a varied range of audio recordings with distinct emotional expressions. This dataset will serve as the basis for training our emotion recognition algorithm. We use Python's librosa package to preprocess audio data and extract key features including Mel-frequency cepstral coefficients (MFCCs), which capture the spectral properties of voice sounds.
2. **Model development with TensorFlow and Keras:**
   We use TensorFlow and Keras to create a convolutional neural network (CNN) or hybrid model capable of successfully learning and classifying emotions from pre-processed audio information.
   The model architecture will be developed to accommodate the sequential nature of audio input while also capturing temporal and spectral information required for emotion assessment.
3. **Training and Evaluation:**
   The model is trained on the pre-processed audio dataset, with a part set aside for validation to track model performance and prevent overfitting. We use relevant assessment measures including accuracy, precision, recall, and F1-score to analyse the model's performance in recognizing various emotions.
4. **Iterative refinement:**
   We improve the model architecture and training process by iterative experimentation and hyperparameter optimization to maximize performance and generalization capabilities. Data augmentation techniques can be used to increase model resilience and generalizability to previously unknown data.

Following the successful design and validation of the emotion detection model in Phase 1, we move on to Phase 2, which involves real-time emotion detection.

**Phase 2: Real-time Emotion Detection with Server Socket Integration.**

In Phase 2, we enhance the created model's capacity to recognize emotions in real time from live audio feeds. This phase consists of the following extra steps:

1. **Integration With Server Socket:**
   To communicate between the emotion recognition module and a phone or audio input device, we use a server-client architecture with Python's socket library. The server component receives incoming audio streams from the client (phone or audio input device) and analyses them in real time with the learned machine learning model.
2. **Real-time Emotion Detection:**
   Upon receiving audio data from the client, the server preprocesses the incoming audio stream using the same feature extraction techniques employed during model training. The pre-processed audio features are fed into the trained emotion detection model to predict the emotional state of the speaker in real-time.
3. **Deployment and Testing:**
   The integrated system is deployed in a real-world setting for testing and validation, ensuring its reliability and accuracy in real-time emotion detection scenarios. Comprehensive testing is conducted to evaluate the system's performance under various conditions, including different environments and speaker demographics.

## 3. Requirements: -

- Hardware Requirements
  1. High-quality microphones
  2. PC/Laptop
- Software Requirements
  1.Lebrosa
  2.Keras
  3.Tensorflow

## STUDENTS DETAILS

| Name | UID | Signature |
|---|---|---|
| Ashrith Ravikanti | 20BCS4378 | |

## APPROVAL AND AUTHORITY TO PROCEED

We approve the project as described above, and authorize the team to proceed.

| Name | Title | Signature (With Date) |
|---|---|---|
| | | |