Capstone Project Report – Emotion Detection API

1. Executive Summary

This capstone project involves developing and deploying an emotion detection API using machine learning techniques. The primary goal is to train a model that can accurately classify speech features into emotional categories, and deploy it via a FastAPI application hosted on Google Cloud Run.

2. Problem Statement and Dataset

Emotions in human speech can provide vital cues in applications like virtual assistants, customer service, or mental health diagnostics. The dataset used was a subset of EmoV-DB, which contains extracted audio features mapped to emotions like neutral, calm, or angry. After preprocessing, the data was split for training and evaluation.

3. Model Development and Evaluation

Two models were developed:

- **Logistic Regression:** Achieved 100% accuracy on the test set during final evaluation.
- **Random Forest Classifier:** Achieved 91% accuracy on the same test set.
  Due to better generalization and simpler architecture, Logistic Regression was selected for deployment.

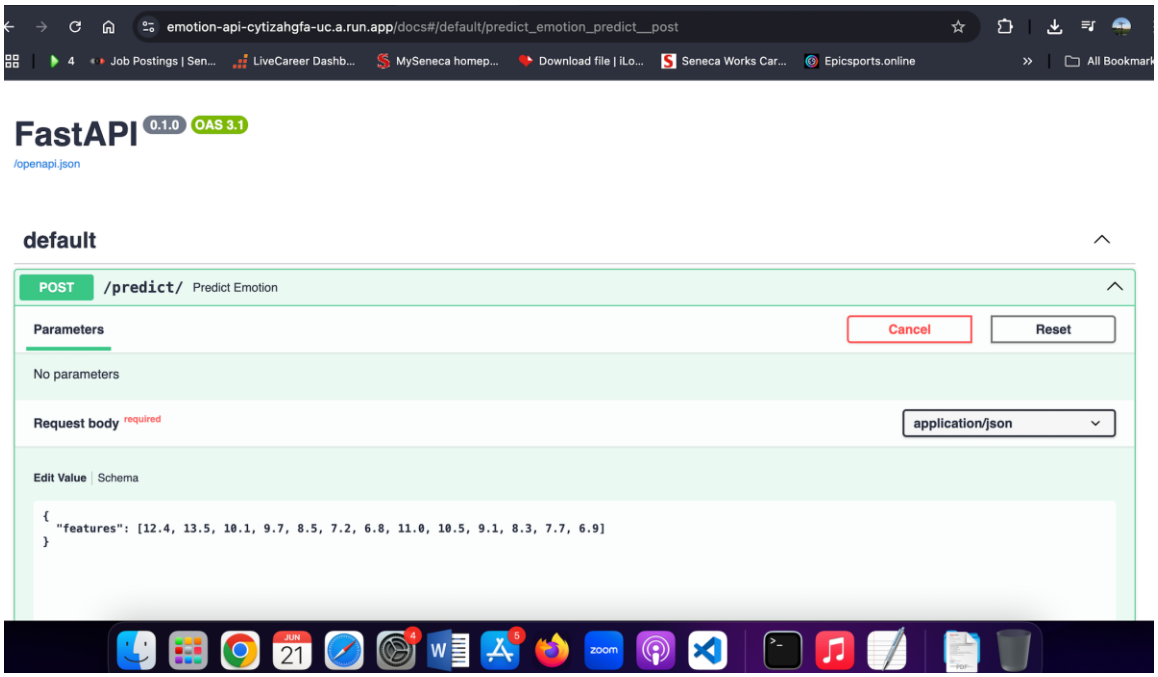
4. API Design and Architecture

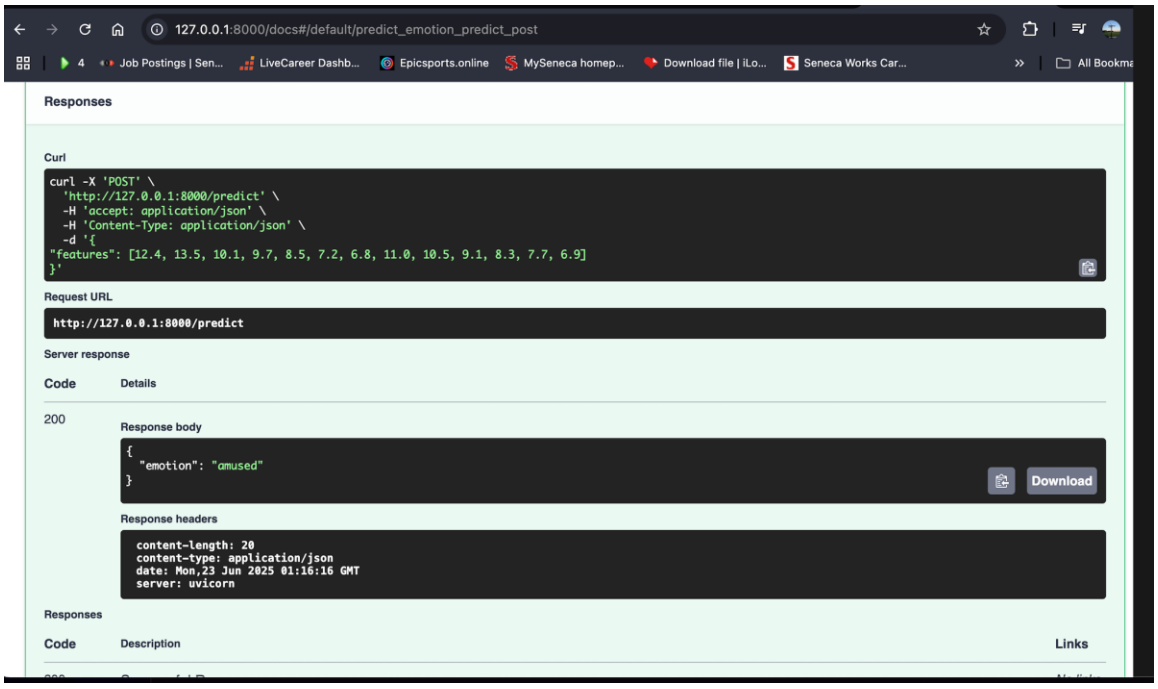
The final API was built using **FastAPI** and deployed to **Google Cloud Run** via Docker. It exposes a single `/predict` endpoint that receives 13 numerical features and returns the predicted emotion label.
Technologies used:

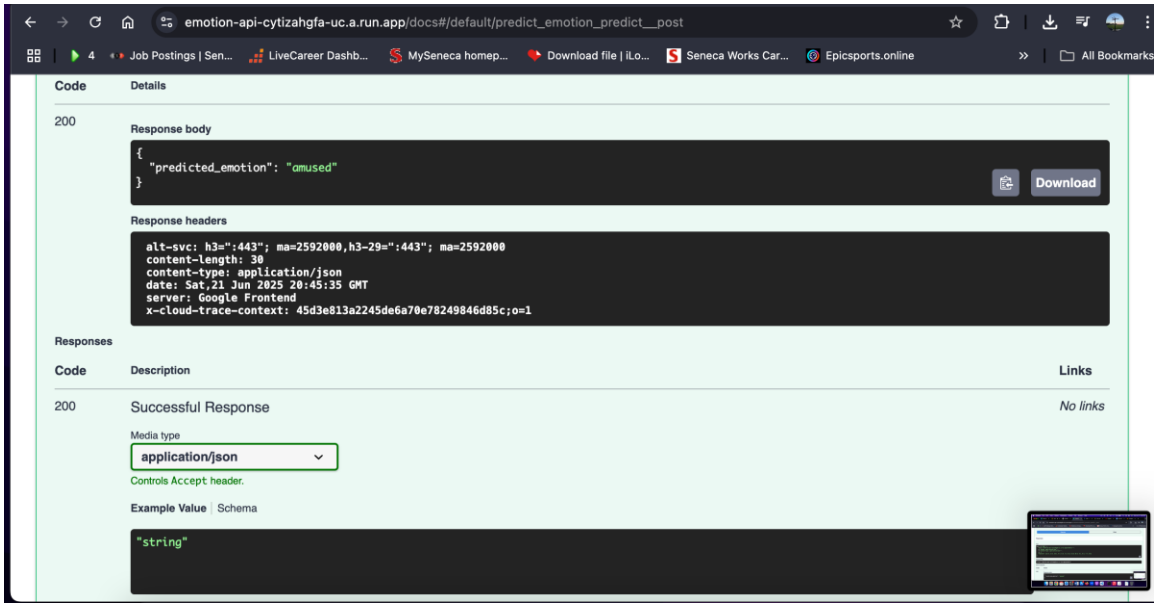
- FastAPI for backend
- Uvicorn as ASGI server
- Docker for containerization
- Google Cloud Run for deployment

5. Screenshots of Working API

The screenshots showing successful API execution and prediction from Swagger UI should be included here. These were captured from FastAPI's interactive docs (`/docs`) with example inputs producing accurate emotion outputs.







6. Challenges and Resolutions

Several challenges were encountered:

- **Model performance discrepancies**: Solved by removing stratified sampling and rebalancing the dataset.
- **Cloud Run container timeout**: Resolved by correctly configuring `uvicorn` in Dockerfile and ensuring port 8080 was used.
- **File loading issues**: Fixed by confirming relative paths in code and ensuring **`.pkl`** model files were copied to the Docker container.

7. Future Work

This project can be extended in several ways:

- Accepting audio files and extracting features in real-time.
- Adding multi-language support for emotion classification.
- Deploying a web interface for end-user testing.
- Integrating with chatbots or voice assistants.

8. Live API Endpoint

https://emotion-api-<project_id>.run.app/predict (http://127.0.0.1:8000/docs).