Sample Prepper: Cloud-Based Audio Processing

A Cloud Computing Implementation Project

Project Access

- Notebook: Google Colab
- Repository: GitHub
- Technologies: Google Colab, Python 3.10, Flask, ngrok, torchaudio

Business Understanding

- Problem: Manual audio sample preparation is time-consuming
- Solution: Cloud-based automated processing service
- Cloud Benefits: Scalable processing power, no local installation needed

Data Understanding

- Digital Waveforms:
  - Sample rate
  - Time & Frequency Domains
  - Amplitude
  - Noise Levels
  - Harmonics
- Audio file formats / MIME types
- Metadata
- Cloud considerations:
  - Data transfer optimization
  - Storage requirements
  - Processing requirements

Implementation Status

- Functional REST API endpoint accepting WAV files
- Audio analysis pipeline implemented
- Automated deployment via Google Colab
- Testing completed with sample WAV files

API Documentation

POST /process-audio

- Input: WAV file
- Output: Processed WAV file
- Status: Implemented and tested

Cloud Architecture

- Google Colab for serverless compute
- Flask REST API for service interface
- ngrok for public endpoint tunneling
- Future scaling possibilities with containerization

Deployment

- Serverless deployment on Google Colab
- REST API endpoints for processing
- Automated dependency management
- Containerization-ready architecture

Evaluation & Results

- Successfully implemented cloud-based API micro-service
- Integration with existing project
- Demonstrates understanding of:
    - PaaS implementation
    - RESTful service architecture
    - Cloud resource utilization
    - API endpoint management

Future Work

- Container deployment options
- Continous integration and fine-tuning
- Batch processing capabilities
- Load balancing implementation

Conclusion

This project demonstrates practical implementation of cloud computing concepts including:

- Platform as a Service (PaaS)
- RESTful microservice architecture
- Cloud resource utilization
- Scalable architecture design