Problem Statement 1

AI-Generated Voice Detection (Tamil, English, Hindi, Malayalam, Telugu)

1. Introduction

AI systems can now generate very realistic human-like voices. Because of this, it is difficult to identify whether a voice recording was spoken by a real human or generated by an AI system.

In this problem, students must build an API-based solution that detects whether a given voice sample is AI-generated or Human, across five supported languages.

2. Supported Languages (Fixed)

Your system must support only these five languages:

- Tamil
- English
- Hindi
- Malayalam
- Telugu

Each request will contain one audio file in one of the above languages.

3. What You Need to Build

You must design and deploy a REST API that:

- Accepts one MP3 audio file at a time
- Audio will be sent only as Base64
- Analyzes the voice
- Returns whether the voice is:
  - AI_GENERATED
  - HUMAN
- Responds in JSON format
- Is protected using an API Key

4. Input Rules

- Audio format: MP3
- Input type: Base64 encoded
- One audio per request
- Audio must not be modified

5. API Authentication

Your API must validate an API Key sent in request headers.

API Key Header Format

- **x-api-key:** YOUR_SECRET_API_KEY

Requests without a valid API key must be rejected.

6. API Request (cURL Example)

Endpoint Example

- POST https://your-domain.com/api/voice-detection

cURL Request Example

```
curl -X POST https://your-domain.com/api/voice-detection \
 -H "Content-Type: application/json" \
 -H "x-api-key: sk_test_123456789" \
 -d '{
  "language": "Tamil",
  "audioFormat": "mp3",
  "audioBase64": "SUQzBAAAAAAAI1RTU0UAAAAPAAADTGF2ZjU2LjM2LjEwMAAAAAAA..."
 }'
```

7. Request Body Fields

Field	Description
language	Tamil / English / Hindi / Malayalam / Telugu
audioFormat	Always mp3
audioBase64	Base64-encoded MP3 audio

8. API Response Body (Success)

Example Response

- {
- "status": "success",
- "language": "Tamil",
- "classification": "AI_GENERATED",
- "confidenceScore": 0.91, // out of 1.0
- "explanation": "Unnatural pitch consistency and robotic speech patterns detected"
- }

9. Response Field Explanation

Field	Meaning
status	success or error
language	Language of the audio
classification	AI_GENERATED or HUMAN
confidenceScore	Value between 0.0 and 1.0
explanation	Short reason for the decision

10. Classification Rules (Strict)

Only one classification field is required:

- AI_GENERATED → Voice created using AI or synthetic systems
- HUMAN → Voice spoken by a real human

👉 voiceSource is removed because it is logically the same as classification.

11. Error Response Example

{

"status": "error",

"message": "Invalid API key or malformed request"

}

12. Evaluation Process

1. System sends one Base64 MP3 per request
2. Language will be one of the 5 supported languages
3. Your API analyzes the voice

4. JSON response is returned
5. Multiple requests are made for evaluation

13. Evaluation Criteria

Participants will be evaluated on:

- 🎯 Accuracy of AI vs Human detection
- 🌍 Consistency across all 5 languages
- 📦 Correct request & response format
- ⚡ API reliability and response time
- 🧠 Quality of explanation

14. Rules & Constraints

- ❌ Hard-coding results is strictly prohibited
- ❌ Misuse of data leads to disqualification
- ⚠️ External detection APIs may be restricted
- ✅ Ethical and transparent AI usage is mandatory

15. One-Line Summary

> Build a secure REST API that accepts one Base64-encoded MP3 voice in Tamil, English, Hindi, Malayalam, or Telugu and correctly identifies whether it is AI-generated or Human.

16. Sample Reference Voice:

> Drive link - sample voice 1.mp3

Endpoint Submission Rules

Participants must submit one public API endpoint URL

The endpoint must correspond to the selected problem statement

The endpoint must be live, accessible, and stable during evaluation

Participants must provide a valid API key for authentication

Late or non-working endpoints will not be evaluated

What the Evaluation System Will Do

Send official test inputs to your endpoint

Validate authentication using the API key

Check request handling and response structure

Evaluate the correctness and stability of your solution

Applicable for Both Problems

Problem 1: AI-Generated Voice Detection — Your API must accept audio input and return classification results

Problem 2: Agentic Honey-Pot — Your API must accept scam messages and return extracted intelligence

Evaluation Readiness

Ensure your API handles multiple requests reliably

Ensure correct JSON response format as defined in the problem statement

Ensure low latency and proper error handling

Outcome of This Level

Your endpoint moves to the automated evaluation stage

Results and scores will be generated based on API performance

Problem Description

Participants must build an API-based system that detects whether a given voice sample is AI-generated or spoken by a real human. The solution must support five specified languages and return results in a structured JSON format.

Problem Definition & Input Guidelines

Participants must design and deploy a **REST API** that analyzes voice samples and classifies them as **AI-generated** or **Human**.

Supported Languages: Tamil, English, Hindi, Malayalam, Telugu.

Sample Audio: A Google Drive link containing **one sample MP3 audio** will be shared for reference.

Input: One Base64-encoded MP3 audio per request.

Submission Requirement

Participants must submit a publicly accessible **API endpoint** along with an **API key**.

Reference Documentation

Complete API specifications, request/response examples, authentication rules, and evaluation details are available in the official documentation.

View Reference Documentation

Expected Output

The API response must include the classification result (AI_GENERATED or HUMAN) along with a confidence score between 0.0 and 1.0.

Rules & Evaluation

- Hard-coding is strictly prohibited
- External detection APIs may be restricted
- Accuracy, stability, and explainability will be evaluated

This endpoint tester allows participants to validate their API for the AI-Generated Voice Detection problem. Participants can test authentication, request handling, audio input processing, and response structure by sending a sample voice input to their deployed API endpoint before final evaluation.

How to Use the Endpoint Tester

This tool helps participants verify that their API endpoint is working as expected for the AI-Generated Voice Detection (Multi-Language) problem.

Steps:

- Enter your deployed API endpoint URL
- Provide the required **Authorization / API key** in the header
- Add a short message describing the test request (for reference)
- Provide a valid **audio file URL** pointing to an MP3 voice sample
- Click **Test Endpoint** to send the request

What This Tests:

- API authentication using headers
- Ability to receive and process audio input
- Correct request parsing and validation
- Proper JSON response formatting
- API stability and response behavior

Note: This tester is for validation purposes only. The final evaluation will use a separate automated system with official voice samples.

sample voice input to their deployed API endpoint before final evaluation.

Headers *

x-api-key *

Enter x-api-key

Endpoint URL *

Enter endpoint url

Request Body *

Language *

Enter language

Audio Format *

Enter audio format

Audio Base64 Format *

Enter audio base64 format