Audio Recorder API - Technical Documentation

February 13, 2025

Overview

The Audio Recorder API provides real-time audio streaming capabilities via WebSockets and device information via HTTP GET. It supports system audio, microphone audio, and mixed audio recording modes. Audio is streamed in WAV format chunks.

API Endpoints

1. `WebSocket Endpoint: /ws/{mode}`

Purpose

Real-time audio streaming.

Method

WebSocket.

Path Parameter: {`mode`}

- `system`: System audio capture.
- `mic`: Microphone audio capture.
- `mixed`: Mixed system and microphone audio capture.

WebSocket Communication Flow

1. **Connection:** Initiate WebSocket connection to `/ws/{mode}`.
2. **Configuration:** Send `RecordingConfig` JSON as text immediately after connection.
3. **Data Stream:** Receive WAV audio chunks as binary data.
4. **Stop Signal:** Send `"stop"` as text to terminate recording.
5. **Disconnection:** Close WebSocket connection.

`RecordingConfig` **(JSON - Text Message upon WebSocket Connection)**

```
{
  "target_url": "string",    // Unused
  "min_split": float,        // Minimum chunk duration (seconds)
  "max_split": float,        // Maximum chunk duration (seconds)
  "sample_rate": integer,    // Audio sample rate (Hz)
  "channels": integer        // Number of audio channels (1 or 2)
}
```

Data Format (Streamed via WebSocket)

WAV audio chunks (binary).

2. GET Endpoint: /devices

Purpose

Retrieve available audio input devices.

Method

HTTP GET.

Request

```
GET /devices
```

Response (JSON)

```
{
  "devices": [
    {
      "id": integer,          // Device ID
      "name": string,         // Device Name
      "channels": integer,     // Max input channels
      "default": boolean      // Is default device
    },
    // ... more device objects ...
  ]
}
```

Address and Port

Default Address

`127.0.0.1` or `localhost`

Default Port

`8000`

Port Configuration

Port is configurable during API startup using command-line arguments (e.g., `--port`).