

AI Reel Generator - Project Overview

An intelligent Flask web application that transforms your content into professional Instagram reels with AI-generated voiceovers. Upload media files, describe your reel, and let the AI handle the rest.

Key Features

- **Automated Processing** – Upload files → AI voiceover → Professional video generation (no manual work required)
- **Non-Blocking Operations** – Continue browsing while reels render in the background
- **Instant Gallery Updates** – Auto-refreshing gallery shows your new reels as soon as they're ready
- **Batch Upload Support** – Upload multiple images/videos for a single reel
- **Professional Voiceovers** – ElevenLabs AI generates natural-sounding narration
- **Instagram Optimized** – All outputs are formatted as 1080x1920 (9:16) vertical videos perfect for Reels
- **Responsive Design** – Works seamlessly on desktop and mobile devices

Tech Stack

Component	Technology
Backend	Flask (Python)
Audio Generation	ElevenLabs API
Video Processing	FFmpeg
Frontend	HTML5, CSS3, JavaScript
Storage	Local filesystem

Prerequisites

Before you begin, ensure you have:

- **Python 3.8+** installed
- **FFmpeg** installed and added to system PATH
 - Windows: Download from ffmpeg.org
 - Mac: `brew install ffmpeg`
 - Linux: `sudo apt install ffmpeg`

- **ElevenLabs API Key** (sign up free at elevenlabs.io)
-

Quick Start

1. Clone & Setup

```
bash
git clone https://github.com/AbdullahMansuri-Codes/AI-Reel-Generator---Fully-Working-Version
cd AI-Reel-Generator---Fully-Working-Version
pip install -r requirements.txt
```

2. Configure API Key

- Open `test_to_audio.py`
- Replace `ELEVENLABS_API_KEY` with your actual key
- Alternatively, set it as an environment variable: `ELEVENLABS_API_KEY=your-key-here`

3. Run the Application

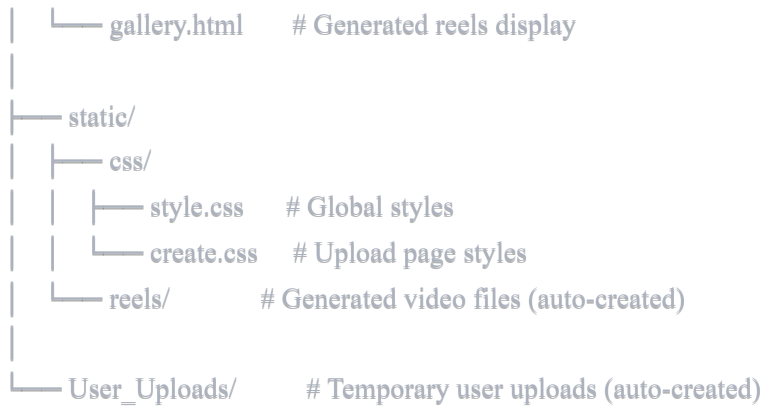
```
bash
python main.py
```

4. Access the Web App

- Open browser and go to `http://localhost:5000`
 - Navigate to "Create Reel" tab
 - Start creating!
-

Project Structure

```
project/
├── main.py           # Flask server & routing
├── generate_process.py # Core video generation logic
├── test_to_audio.py  # ElevenLabs API integration
├── requirements.txt   # Python dependencies
├──
├── templates/
│   ├── base.html     # Base template layout
│   ├── index.html    # Home page
│   └── create.html    # Upload & creation interface
```



💡 How It Works

User Workflow:






1. Upload image/video files
2. Enter descriptive text for voiceover
3. Click "Create Reel"
4. Processing overlay appears (non-blocking)
5. Page redirects to gallery
6. Gallery auto-refreshes every 5 seconds
7. Reel appears when ready

Backend Process:

1. **Upload Phase** – Files saved to `User_Uploads/{unique_id}/`
2. **Audio Generation** – ElevenLabs converts description text to speech → `audio.mp3`
3. **Video Creation** – FFmpeg combines media files + audio → 1080x1920 video
4. **Output** – Final reel saved to `static/reels/{unique_id}.mp4`

✅ Features Implemented

- ✅ Asynchronous background processing with threading
- ✅ Robust file upload and validation
- ✅ Cross-platform path compatibility (Windows/Mac/Linux)
- ✅ Auto-creating required directories
- ✅ Processing overlay & progress feedback

-  Auto-refreshing gallery interface
-  Support for multiple file formats (PNG, JPG, MP4, MOV, AVI)
-  Error handling and logging
-  Download buttons for generated reels
-  Mobile-responsive design

Troubleshooting

Issue	Solution
FFmpeg not found	Verify FFmpeg is installed and in PATH by running <code>ffmpeg -version</code>
API key invalid	Double-check your ElevenLabs API key and usage quota
Video not generating	Check console logs, verify file permissions, wait 30-60 seconds
File upload fails	Ensure files are under 500MB and use allowed formats
Gallery not updating	Clear browser cache or manually refresh the page

Configuration

To use environment variables for your API key:

```
python

import os
ELEVENLABS_API_KEY = os.getenv('ELEVENLABS_API_KEY', 'default-key')
```

Set environment variable before running:

```
bash

export ELEVENLABS_API_KEY=your-actual-key-here
python main.py
```

Important Notes

- This is a demo project—test thoroughly before production use
- Generated videos are stored locally in `static/reels/`
- Monitor your ElevenLabs API usage to avoid unexpected charges
- Ensure proper folder permissions for file operations

- Large video files may take longer to process

Contributing

Found a bug or have an idea? Feel free to open an issue or submit a pull request!

Created by: Abdullah Mansuri | [GitHub](#)