ChatVid AI

Ever wish your video could answer back? Now it can.

A multimodal video-analysis system that lets users chat with YouTube videos

Headstarter SWE Residency Sprint 2 | May 2025

by Misbah Ahmed Nauman

Problem & Motivation

Why We All Struggle With Videos

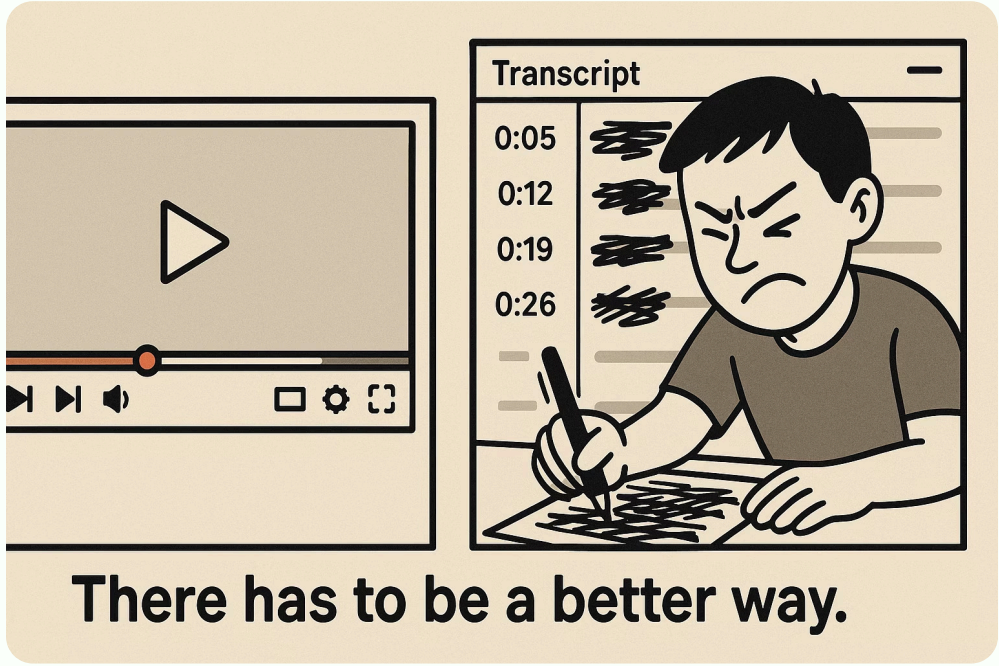
Ever watched a 10-minute lecture and wasted 3 minutes scrubbing around just to find that one key concept?

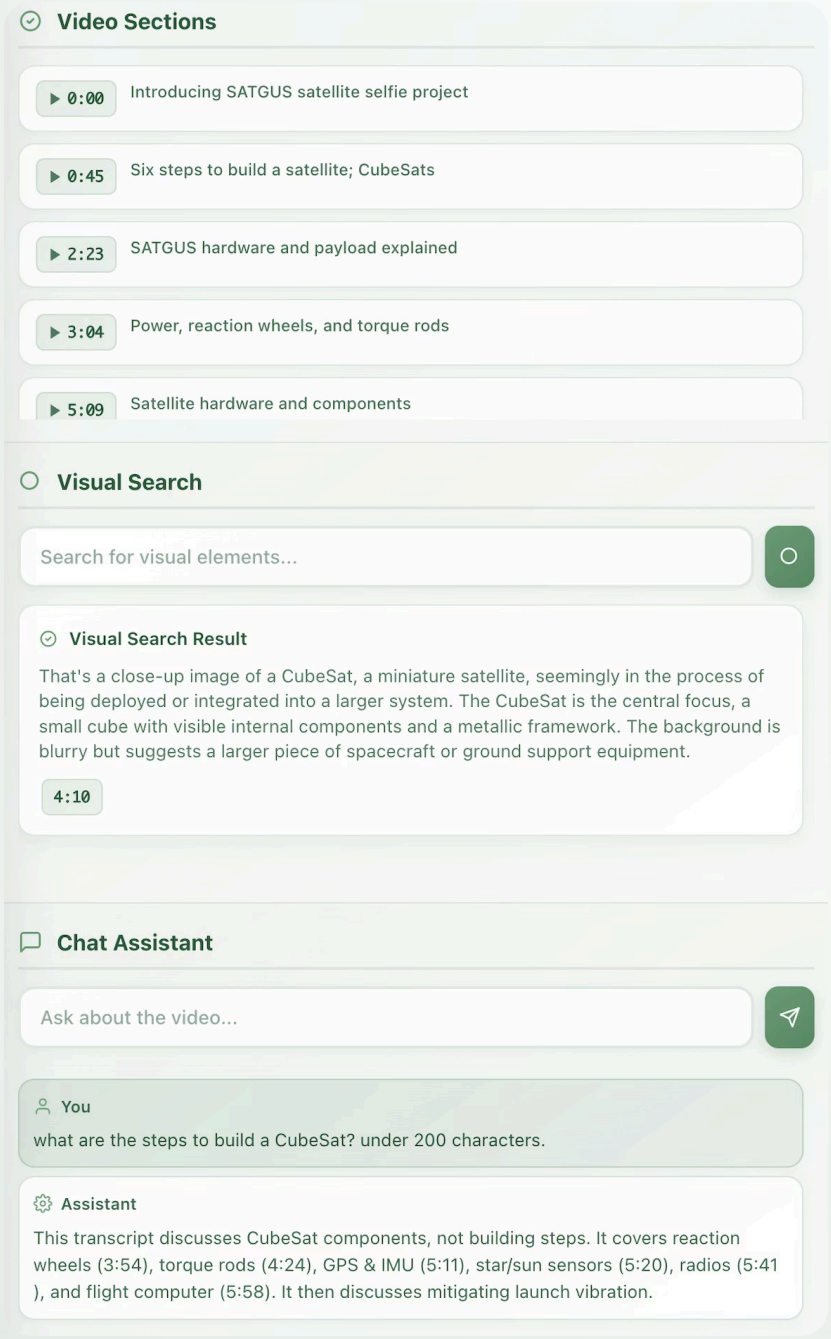
Current Frustrations

• Endless Scrubbing: You drag the bar back and forth, hoping to land on the right moment.

• Disconnected Tools: You might open a transcript-only site –but then you can't click a timestamp to watch.

• Missing Visual Search: "I know I saw a red car at 2:15... but where?"





Solution Overview

Fetch YouTube Transcripts

Using youtube-transcript-api to extract text content

AI-Powered Summaries & Q&A

Generate timestamped summaries and handle questions

Visual Snapshot Embedding

Extract frames with OpenCV and embed them using Gemini’s multimodal API

Present in React Frontend

Next.js + TypeScript + Tailwind for seamless experience

System Architecture

Frontend (Next.js + TypeScript)

Two main routes: Single-page app with two views

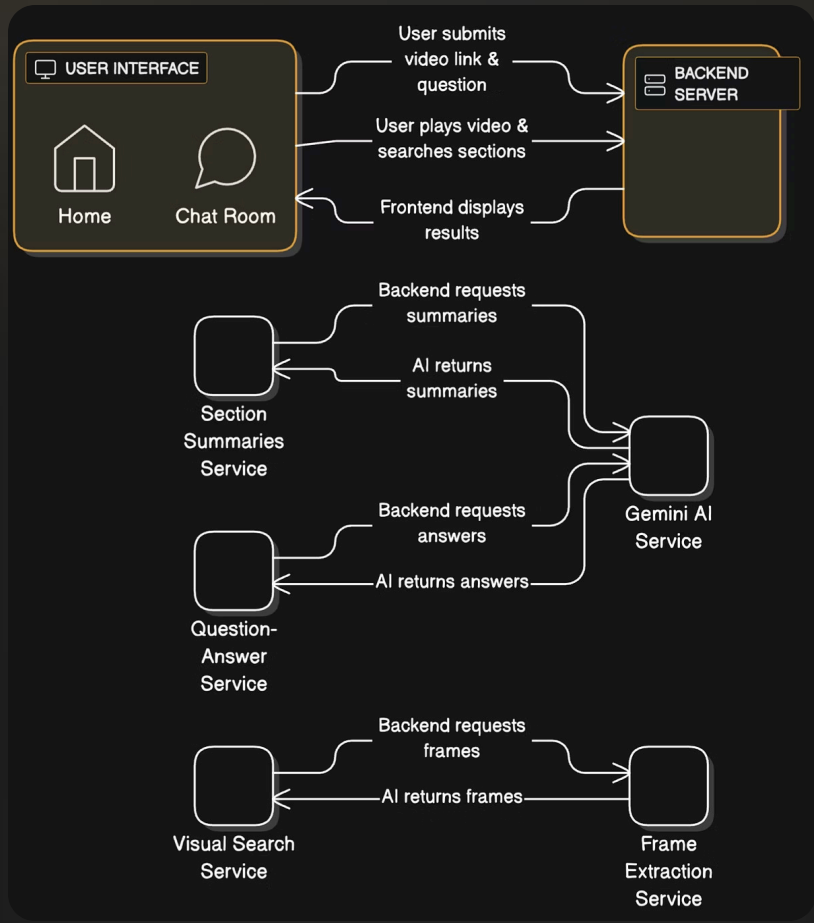
- **Home**: paste your YouTube link + AI key
- **Chat**: YouTube player, clickable section list, chat interface, and visual search

Backend (FastAPI)

Routes: Transcript Service, Section Summaries Service, Question-Answer Service, Visual Search Service

Third-party Services

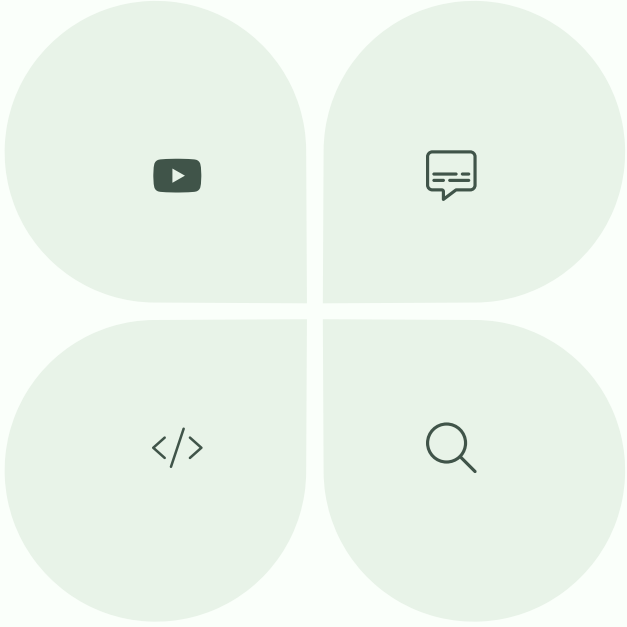
Gemini API (text, image, video), youtube-transcript-api, yt-dlp + OpenCV



Key Features

Paste YouTube Link

→ Upload any public video and extract its transcript automatically



Chat with Video

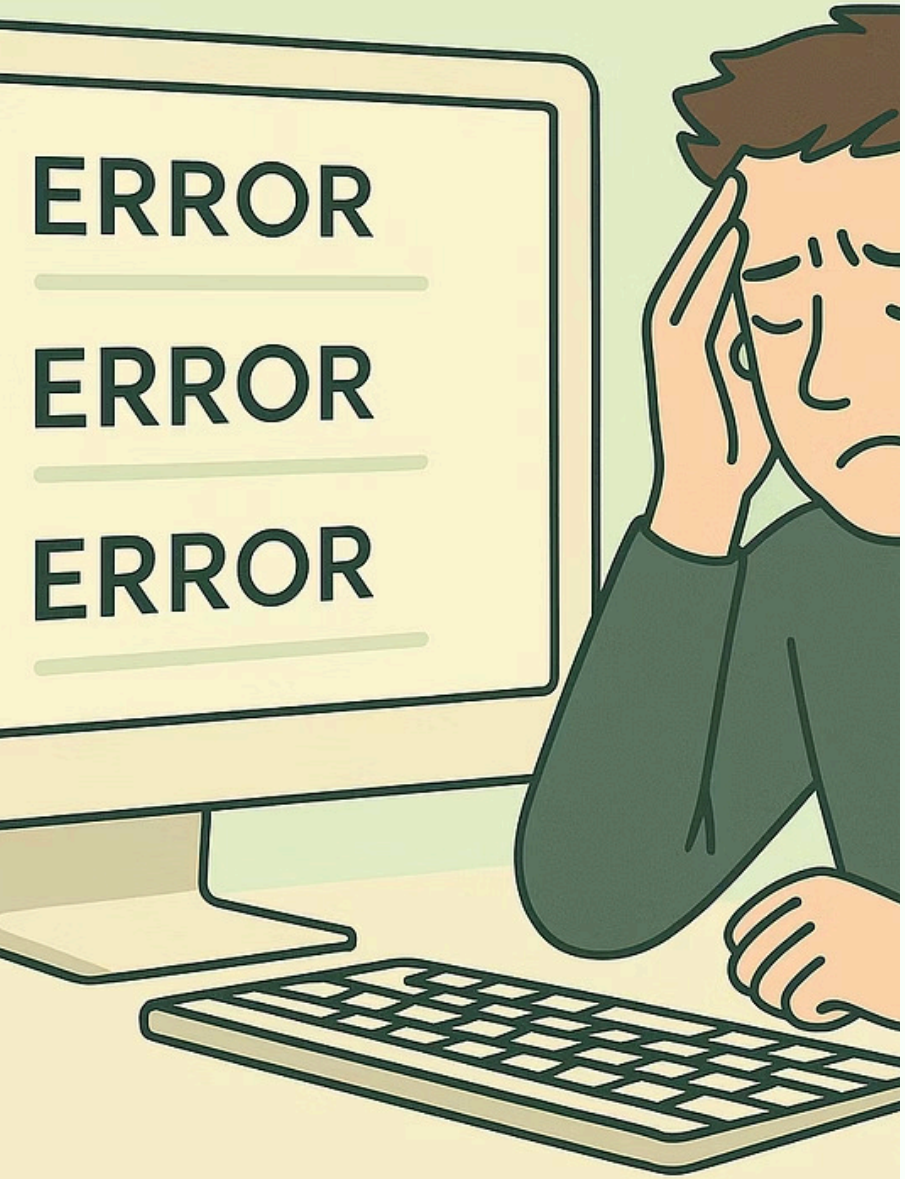
→ Ask questions like “What’s happening at 1:23?” with timestamped answers

Timestamped Summaries

→ Gemini breaks the video into structured, clickable chapters

Visual Frame Search

→ Search objects like “red car” and jump to matching frames



Challenges & Solutions

Async Frame Embeddings

Switched from blocking CPU to asyncio.gather() with httpx.AsyncClient

Timestamp Seeking

Replaced unreliable ReactPlayer with native iframe

Gemini Integration

Leveraged multimodal capabilities instead of juggling separate models

Prompt Engineering

Crafted precise prompts for structured JSON responses

What's Next?

Can we tell you who's speaking?

Yes–speaker labels are coming soon

Want to compare two videos?

Future updates will let you query multiple URLs at once

Upload Local MP4s?

Support file upload in addition to YouTube links

Contact: misbahahmed2005@gmail.com | MisbahAN.com