

UTSAV V CHAUDHARY

utsavmaan28@gmail.com | portfolio | github/UttU28

Cover Letter

I'm excited to introduce myself and express my genuine **enthusiasm** for the opportunity to join your team. With a **strong passion** for problem-solving and creating meaningful solutions, I approach every challenge with a **fresh perspective** and a **commitment to excellence**. My **expertise** includes software development, web and app creation, backend engineering, and working with advanced technologies like large language models (LLMs).

I love working on personal projects that challenge the status quo and allow me to approach problems from a **fresh perspective**. Whether it's building an AI-powered home assistant or creating automation tools to simplify everyday tasks, I enjoy diving into ideas that blend **innovation** with **practicality**. These projects are not only opportunities for technical growth but also a reflection of my **passion** for creativity and learning.

Beyond my technical skills, I take **pride** in taking on **leadership roles** and fostering **collaboration** within teams. I approach my work with **confidence**, knowing that the effort I invest will lead to meaningful results. From brainstorming ideas to implementing solutions, I **thrive** on bringing people together and driving projects forward with clarity and purpose.

I'm a **quick learner** who thrives on turning challenges into opportunities to grow and evolve. Each project I've worked on has taught me invaluable lessons, shaping my **ability** to tackle complex problems with **creativity** and **determination**. I'm excited about the opportunity to bring this mindset to your team and explore how we can innovate and grow together.

Thank you for considering my application. I look forward to the chance to discuss how my skills and experiences align with your goals. Let's connect to explore the possibilities!

Projects

Apeksha (Personal Assistant + Home Assistant)

Personal Project

GitHub: [github/UttU28/Apeksha](https://github.com/UttU28/Apeksha)

- Fine-tuned and hosted a custom LLM (**LLama 3.2 Vision**) locally with **GPU acceleration**, ensuring high-performance functionality while keeping all data secure and local.
- Configured **Ollama** as the primary **Conversation Agent**, routing queries and commands to the locally running LLM, with unrelated queries seamlessly forwarded to external LLMs via API.
- Built a personal home assistant on **Raspberry Pi**, integrating intent recognition, task automation, and voice control, powered by the custom LLM and external APIs.
- Trained & implemented custom wake-word detection model using **OpenWakeWord** & **TensorFlow** for real-time detection.
- Deployed containers with **Wyoming Satellite** to integrate **Whisper (Speech-to-Text)** and **Piper (Text-to-Speech)** modules, enabling seamless voice interaction capabilities.
- Designed a **web dashboard** showcasing Apeksha's capabilities, featuring integrations with custom automation projects (available on GitHub):
 - **Movie Controller:** Remote media control system using PC and smart devices. [link](#)
 - **Job Application Helper:** Automated job search and application tracking. [link](#)
 - **Bhashini:** API-based translation and recognition of 22+ Indian languages. [link](#)
 - **Ashwathama:** Car control system utilizing OBD2 sensor data for real-time monitoring and analytics. [link](#)
- Integrated **Firebase Database** for **authentication** and storing data related to **To-Do tasks**, **Ashwathama metrics**, and other project features. As this is a **personal project**, no external user data is stored, ensuring privacy and security.
- Architected a unified API-based communication framework to connect all applications, enabling cross-project voice automation and control with AI intelligence.
- Configured automation workflows to recognize intents and trigger actions, enhancing usability and convenience.
- Actively expanding features and functionality, with ongoing updates to enhance the project's capabilities and user experience.

LinkedIn Reverse Search

Client Project

- Developed a LinkedIn scraping bot using **Python**, **BS4**, **Selenium**, and **Requests** to automate **data extraction** and validation from diverse web sources.
- Fine-tuned a **Hugging Face model** to detect features from messy **HTML structures**, enabling accurate scraping, & validation.
- Designed and implemented a pipeline to process input Excel sheets with headers (`firstName`, `lastName`, `companyName`) and output enriched data with additional details, including LinkedIn URLs and verified Contact Emails.
- Built a **threaded data enrichment pipeline** to process data in parallel across LinkedIn, SalesQL APIs, and the fine-tuned LLM, improving efficiency by **30%**.

- Developed a client web app using **Next.js**, integrating user auth with **Firestore Auth** & data storage via **Firestore DB**.
- Maintained a database of prev data to eliminate redundancies, reducing processing time for recurring requests by **40%**.
- Set up a secure **SMTP server** for automated email delivery of Excel sheets to recipients, ensuring reliable data sharing.

Eventbrite & LUMA Scraping

Client Project

- Developed a Python bot to scrape events from **Eventbrite** and **LUMA platforms** hourly, extracting key details such as Event Name, Description, Start Date, End Date, and Registration URL.
- Designed an optimized **SQL schema** to store and manage structured event data, ensuring seamless retrieval and scalability.
- Built a dynamic **keyword filtering system** to identify events related to **Cryptocurrency**, **RWA**, and **Tokenization**, with additional filters for **Date Time** and **Location**.
- Automated **data cleaning and preprocessing** pipelines in Python to ensure relevant and accurate event information.
- Created a **web application** to display event data, incorporating **CRM functionalities** for tracking and managing events.
- Implemented a robust ingestion pipeline to handle continuous updates from event sources, improving processing speed **35%**.
- Delivered tailored solutions for **keyword-based event categorization**, aligning with client-specific business goals and improving event relevance.

Instagram Teaching Reels' Automation

Personal Project **Instagram:** [instagram/that_vocab_girl/](#)

- Developed a Python-based pipeline to dynamically create and upload English teaching reels on Instagram, showcasing vocabulary lessons and language tips.
- Automated raw video collection using **PlayPhrase.me API**, storing initial files on **Google Drive** for free & scalable storage.
- Utilized **FFMPEG** for advanced video editing, including transitions, overlays, and image insertions, combined with **Photoshop** for logo and design customizations.
- Integrated **WhisperAI** for generating subtitles with precise timestamps, enhancing accessibility & viewer engagement.
- Stored and managed processed video URLs on **Azure Blob Storage**, ensuring efficient and scalable media handling.
- Orchestrated end-to-end workflow with **Azure DevOps** & **CRON jobs**, achieving automated pipeline from creation to upload.
- Automated Instagram uploads using **Selenium** in **Docker containers** on **Ubuntu OS**, with session persistence to avoid repeated logins and added a preview dashboard for content review.

Movie Controller Application

Personal Project **GitHub:** [github/UttU28/Movie_Controller_2](#)

- Developed a cross-platform media controller as a Python-based client-server application, enabling remote control of a PC for streaming platforms like **YouTube**, **Netflix**, and **Prime Video**.
- Designed a Smart TV-style remote interface using **Vite** and **React**, ensuring an intuitive and responsive user experience.
- Configured **POST requests** from **React client** to **Flask** backend, enabling seamless communication for command execution.
- Integrated **PyAutoGUI** to emulate PC controls like typing, searching, & navigating through browser and media like Netflix, Prime, YouTube, and other UNdisclosed sites :) for media and content control on device.
- Utilized **OpenCV (CV2)** for image recognition to detect screen objects and perform targeted actions, including ad skipping and automated playback adjustments.
- Enabled browser-based remote access to the React client, allowing to control PC from any device within network securely.
- Provided a streamlined media control experience, enabling users to interact with streaming platforms through their **personal devices**, improving convenience and accessibility.

AssignmentX

Personal Project **GitHub/YouTube:** [github/UttU28/AssignmentX](#)

- Designed & developed an Android & web application that generates handwritten-like assignments, utilizing **Python**, **NumPy**, **Pillow**, **Django**, & **Android Studio**. Achieved over **5000+ downloads** & maintained **200+ daily active users (DAUs)**.
- Leveraged **Pillow** and **OpenCV2** for **OCR-based image processing**, incorporating a custom **human behavioral algorithm** to replicate natural handwriting variations with realistic stroke simulation.
- Developed and deployed a scalable **Python RESTful API** backend on **Azure**, integrating **Azure App Services**, **Azure SQL Database**, and **Blob Storage** for seamless performance and data management.
- Integrated **Django** to facilitate seamless interaction between mobile, web, and backend services, ensuring a unified user experience across platforms.
- Enabled real-time email functionality using **SMTP**, allowing users to generate and send dynamically created PDFs of assignments directly through the app.
- Optimized performance using **multi-threading** & caching techniques, reducing response times & improving user satisfaction.
- Ensured a robust and scalable architecture, maintaining balance between high performance and ease of use for all platforms.