UTSAV CHAUDHARY

Texas | utsavmaan28@gmail.com | (607) 296-9583 | portfolio | in/utsavmaan28 | github/UttU28

Hello there! a developer with ~3 years of experience in tech, working with **Python** for **automation**, **API development**, and **backend systems** using **Python (FastAPI, Flask, Django)** or **Node.js**. On the **frontend**, I build applications with **TypeScript** and **Node.js**, leveraging frameworks like **React** and **Next.js**. I also have experience in **data scraping**, building **data pipelines** for **ML models**, and **fine-tuning LLMs** to enhance **AI-driven applications**.
I actively contribute to **open-source projects** to refine my skills and stay ahead of **emerging technologies**. My **personal projects** always keep me on my toes to implement **new technologies**. I'm always looking to bring a **fresh perspective** to projects, approaching problems from **new angles** to create more **innovative** and **effective solutions**.
If this aligns with your needs, I'd love to connect for an interview and discuss how we can help each other grow.

Technologies

Languages: Python, JavaScript, Java, TypeScript, C++, C#, Rust, Bash, PowerShell, R, Go, Ruby, Swift
Web Technologies: React, Flask, NextJS, NodeJS, Django, FastAPI, Express, ASP.NET, Angular, HTML, CSS, Jinja, YAML
Databases: SQL, PostgreSQL, Azure SQL, Redis, AWS RDS, MongoDB, DynamoDB, Firebase, Firestore
Cloud & DevOps: Azure, AWS, CI/CD, Kubernetes, Docker, Jenkins, Terraform, Ansible, PowerShell, Yarn, NPM, Azure Functions VM Blob, AWS Lambda S3 EC2, RBAC.
Operating Systems: Windows, Linux (CentOS, Ubuntu), macOS, Embedded Systems, RaspberryPi, Arduino
APIs and Protocols: REST APIs, OpenAPI, Swagger, WebRTC, SOAP, GraphQL, MQTT, WebSocket, OAuth, JSON-RPC
Other Tools & Technologies: Kafka, Jira, GitHub, Nginx, VSCode, Azure AKS, Metamask, Fireblocks, Cron, WebSockets

Experience

Software Engineer, Midh Technologies, TX — Feb 2024 - Present

- Fine-tuned the **Llama 3.2 model** on **Azure Kubernetes Service (AKS)**, optimizing for **scalable**, **low-latency inference**, and **real-time data processing**, reducing inference latency by **35%**.
- Designed and deployed secure **RESTful API endpoints** using **FastAPI**, integrated with **Azure API Management** for **authentication**, **rate-limiting**, and **monitoring**, supporting **50k+ API requests/day**.
- Built a **Retrieval-Augmented Generation (RAG) pipeline** with a **vector database**, improving **model response accuracy** by **20%** and reducing **inference times** by **30%**.
- Developed a **server-side communication framework** for seamless interaction with **LLM**, ensuring **efficient integration**.
- Setup **API documentation**, detailing all **functions** of the LLM and web application, and tested endpoints using **Postman**.
- Assisted in developing and deployed the company's official website using **Next.js**, integrating a **dynamic CMS** with admin functionalities, hosted on **Azure App Services**, reducing update times by **40%**.
- Integrated **Fireblocks SDK** and **Metamask** to develop secure **user profile dashboards** for **on-chain ID creation** and **blockchain asset management**, increasing transaction speed by **25%**.
- Established robust **CI/CD pipelines** for **automated building**, **validation**, and **deployment** of infrastructure, cutting production rollout times by **50%** and ensuring **zero downtime deployments**.
- Built and configured **cloud-based pipelines** using **Azure DevOps**, streamlining **versioning**, **testing**, and **deployment workflows**, improving developer efficiency by **30%**.
- Created an **interactive analytics dashboard** with **React.js** and **D3.js**, featuring **advanced visual analytics** and **RBAC**.
- Developed a **CI/CD pipeline** for the website using **Azure DevOps**, automating deployment processes, and utilized **Terraform** for provisioning scalable **Azure App Services**, ensuring **reproducibility** and reducing provisioning time by **50%**.
- Leveraged **Azure Functions** to create **scalable APIs** for **real-time data interaction** and **seamless system communication**, handling **10k+ concurrent users** with **99.9% uptime**.
- Led cross-functional efforts to resolve high-priority **bugs** and manage **support tickets** across distributed teams, utilizing tools like **Jira** and **Slack** to ensure clear communication and timely resolutions, achieving a **94% resolution rate** within SLA.

Python Web Developer, Compendious Medialabs Pvt. Ltd., IN — June 2021 - July-2022

- Designed and developed an in-house **data scraping application** using **Python**, **Selenium**, **RestAPI**, and **Beautiful Soup**, automating **data extraction** from diverse web sources with dynamic content structures, reducing collection time by **60%**.
- Optimized **Selenium-based scrapers** by implementing **multi-threading**, reducing data scraping time by **40%**, and ensuring adaptability to changes in web structures using advanced **XPath** and **CSS selectors**.
- Integrated robust error-handling mechanisms to identify and recover from failures during scraping sessions, improving scraping reliability and ensuring a **93% success rate**.
- Built a robust **ETL pipeline** for **data cleaning**, **transformation**, and **storage** in **DynamoDB**, increasing data reliability by

30% and seamlessly integrating processed data into marketing workflows.

- Leveraged **AWS Cloud Services** to host and manage pipeline infrastructure, ensuring high availability and scalability, while enabling integration with CRM systems for personalized, data-driven campaigns.
- Conducted advanced **data preprocessing** using **Pandas** and implemented **NLP tasks** with **NLTK** and **spaCy**, improving **text classification** and **entity recognition** accuracy by **20%**.
- Developed an interactive **data dashboard** with **Django**, featuring **real-time updates** from **DynamoDB** and integrated **visualizations** using **D3.js** and **Chart.js**.
- Collaborated with cross-functional teams using **Jira** for bug tracking and agile sprint planning, enhancing team productivity and ensuring timely resolution of issues with a **92% bug resolution rate**.

Python ML / Game Developer, VRoKCs, Kansas, US March 2020 - Sept 2020

- Developed a **Sign Language translation and augmentation system** using **machine learning**, incorporating extensive **data collection**, **labeling**, and **manual cleaning** to create a high-quality dataset.
- Utilized **Kaggle datasets** and generated a **custom dataset** by capturing raw images, transforming them with **Pillow** and **OpenCV2**, and applying **filters** to enhance classification diversity.
- Evaluated and optimized multiple **ML algorithms** (**Logistic Regression**, **Naïve Bayes**, **Decision Tree**) and achieved **92% accuracy**, selecting the most performant model for deployment.
- Designed and deployed a **TensorFlow Lite model** for **efficient, real-time inference**, optimized for edge devices.
- Integrated the deployed model with **Unity**, enabling seamless **(ASL) American Sign Language translation** within **Augmented Reality (AR)** environments for real-time user interaction.
- Augmented the dataset with trans4matons like **rotation** and **scaling**, enhancing the model for **real-world scenarios**.
- Applied **OpenCV2** for image preprocessing, including **noise reduction** and **normalization**, ensuring consistent data quality.

Game Developer Intern, underDogs Gaming Studios, IN March 2019 - July 2019

- Completed a **POC** for integrating **Quarters** (a digital cryptocurrency for gaming) into multiple games, enabling seamless **universal gaming currency transactions**.
- Implemented **API integrations** to support **account-specific balances**, adding in-game functionality for **buying coins**, **managing transactions**, and converting **points to quarters**.
- Integrated & optimized **Quarters API** for secure wallet management & seamless transactions across platforms.
- Assisted in **bug fixing** and improving game stability, adhering to **API best practices** and enhancing overall user experience.
- Improved **real-time data synchronization** and communication between **clients** and **servers** using **Python** and **C#**.
- Collaborated on implementing **secure transaction methods** in **Unity3D**, enabling safe and reliable player interactions within the gaming environment.

Projects

Apeksha (Personal Assistant + Home Assistant) Personal Project **GitHub:** `github/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 frm 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 **Firebase 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.

Education

State University of New York at Binghamton Aug 2022 – Dec 2023
Master of Science in Computer Science 3.44/4.00 GPA