Swasti Singh

(346) 521-9034 | sssingh4@wisc.edu | LinkedIn | Personal Portfolio | Madison, WI

OBJECTIVE & EDUCATION

- I'm a senior seeking full-time software/AI engineering roles; Open to relocation.
- **University of Wisconsin-Madison** • Bachelor of Science: **Computer Science and Data Science (Double Major)**
- December 2025 • GPA: **3.86**/4.0 • ***Dean's List*** **x4**

WORK & RESEARCH EXPERIENCE

AidenAI, Software/AI Engineering Intern **Princeton, NJ (May 2025 – Aug. 2025)**

- Built a full-stack, 4-agent Development Agent system (Config, Task, Deploy, Tool Selection) using LangGraph and Hugging Face endpoints to convert plain-English prompts into Unqork UI components, cutting dev time by 60%.
- Developed a responsive frontend with Vite, React, and TypeScript to support real-time, chat-based interactions and display generated UI configurations.
- Implemented real-time streaming backend APIs using FastAPI and AsyncIO, achieving <200 ms first-token latency for seamless user experience.
- Integrated 30+ backend services via Model Context Protocol (MCP) adapters, covering 90% of Low-code/No-code UI component types with dynamic and semantic tool invocation.
- Engineered agent-specific prompts and role templates to guide LLM behavior and improve output consistency.
- Managed session state with PostgreSQL, supporting short-term (in-session) and long-term (persistent) memory for multi-turn interactions.
- Orchestrated agent workflows with LangGraph's state-passing model, enabling sequential, branching, and memory-aware execution across agents for robust LLM-based decision making.
- Presented bi-weekly demos to stakeholders, incorporating feedback into agent improvements.

UW Madison College of Engineering, ML Research Assistant **Madison, WI (Sept. 2024 - Present)**

- Built scalable image classification models using Mixture-of-Experts (MoE) in PyTorch, improving generalization, and reducing compute for large datasets like ImageNet and CIFAR-100.
- Automated 200+ training experiments on a high-throughput cluster, using Weights & Biases for real-time performance tracking, model comparisons, and version control.
- Fine-tuned expert selection and load balancing strategies, boosting model performance on out-of-distribution test sets by 5% while reducing training resource usage.
- Explored LLM-inspired gating strategies in MoE models to improve routing and performance on large-scale vision tasks.

Wisconsin Institute for Discovery, Artificial Intelligence Intern **Madison, WI (Mar. 2024– Dec. 2024)**

- Analyzed urinary tract simulation data to uncover key trends, using PCA to reduce dimensionality and identify top 5 predictive features from 30+ variables.
- Applied correlation analysis and hypothesis testing to validate feature importance and derive clinical insights.
- Engineered a reusable Python toolkit with Pandas, NumPy, and Matplotlib to streamline data exploration and visualization, cutting manual graphing effort by 40%; presented findings to lab members and faculty to promote adoption.
- Cleaned up and modularized code into reusable Python files and added unit tests.

Division of Information Technology (DoIT), Information Technology Specialist **Madison, WI (Jun. 2024– Apr. 2025)**

- Resolved Cisco VoIP issues for ~12 customers weekly (≈570 total) via Cherwell ticket closures, ensuring prompt and effective support.

PROJECTS

Optimized Transit Routing System | *Python, GTFS, Geopandas, AWS EC2, Google BigQuery, Docker, GCP, Kafka,*

- Built a real-time Wisconsin transit routing prototype as a consulting project for Dr. Caraza-Harter.
- Ingested GTFS updates via Kafka and gRPC/Protobuf, delivering sub-10s network refreshes and enabling dynamic schedule adjustments for 100% of routes.
- Clean and enrich a 1 million+ record GTFS dataset using Python, Pandas, and Docker, staging processed feeds in Google Cloud Storage and loading into BigQuery, slashing data-quality issues by 90%.
- Implemented Dijkstra's algorithm, BFS-based methods, and matrix optimization to compute shortest paths, integrating walking connections, transit delays, and time constraints for real-life optimal route planning.

Animify: RAG-Powered Prompt-to-Animation Platform (demo) | *LangChain, React, Typescript, FastAPI, Manim, Docker, Chroma, JavaScript*

- Built a RAG-powered prompt-to-animation platform, integrating a LangChain pipeline with OpenAI embeddings and Chroma vector store to semantically retrieve SVG templates and FastAPI endpoints powered by Manim for on-the-fly SVG/MP4 rendering.
- Developed a responsive React & TypeScript SPA, featuring live SVG previews, interactive parameter controls, and seamless calls to the retrieval API for template suggestions and animation triggers.
- Implemented comprehensive testing with Jest (frontend) and PyTest (backend), achieving > 80% code coverage to ensure reliability and catch regressions across the animation workflow.

CERTIFICATIONS & SKILLS

- **AWS Certified Developer Associate, Amazon Web Services –** Issued on 08/2024 **(credential link)**
- **AWS Certified Cloud Practitioner, Amazon Web Services –** Issued on 05/2024 **(credential link)**
- **Languages, Tools, Databases & Frameworks:** Apache Kafka, Spark, ETL, Docker, Python, R, Java, Ruby, Linux, Git, React, SQL, Node.js, PostgreSQL, MongoDB, Julia, OOP, Data Structures & Algorithms, Optimization, Big Data Systems, AWS, GCP, Agile.
- **Machine Learning:** Computer Vision, Deep Learning, Scikit-learn, TensorFlow, NumPy, Statistics, Probability.