Smit Mahesh Panchal

spanch19@asu.edu | +1 (480) 853-9400 | linkedin.com/in/smit-panchal-9a83ba1b7 | github.com/SmitPanchal61

Education

Master of Science in Software Engineering — *Graduating May 2026*
Arizona State University, Arizona, USA — *Current GPA – 3.93 / 4.00*

B.E. in Computer Engineering, AI & ML Honors — *Dec 2020 – Apr 2024*
University Of Mumbai, Maharashtra, India — *Overall GPA – 3.58 / 4.00*

Technical Skills

Languages: Python, Java, JavaScript, TypeScript, C/C++, SQL
Frontend & Web: React, Vue.js, HTML, CSS, Tailwind CSS, REST APIs
Backend & Frameworks: FastAPI, Flask, Django, Spring Boot, Node.js, Express.js
Databases: PostgreSQL, MySQL, MongoDB, Firebase, SQLite
Cloud & DevOps: AWS (EC2, S3, Lambda, ECR, SQS, IoT Greengrass), Docker, Kubernetes, CI/CD
Applied AI & Data: NLP, LLMs, RAG, Embeddings, CNNs, TensorFlow, Scikit-learn, Pandas, NumPy

Experience

Machine Learning Engineering Intern — *Escape LLC, USA* — *Oct 2025 – Jan 2026*
- Built FastAPI endpoints to ingest raw business data, trigger enrichment workflows, and return structured JSON responses.
- Integrated third-party APIs (OpenAI, Google Places) to enrich lead records with summaries, scores, and metadata.
- Implemented Firestore data models to persist enriched records and support efficient querying by downstream services.
- Containerized backend services with Docker to ensure reproducible local development and cloud deployment.
- Collaborated with engineers in agile workflows to debug API failures, improve request handling, and stabilize backend services.

NLP & Machine Learning Intern — *Blue Clay Health, USA* — *Aug 2025 – Sep 2025*
- Built document ingestion pipelines to accept uploaded files, preprocess content, and generate embeddings for search.
- Implemented backend APIs to support semantic document retrieval using vector similarity search.
- Developed React components for file upload, document listing, and user-triggered actions.
- Integrated frontend components with backend APIs to enable end-to-end document processing and retrieval.

Projects

Web-Based Software Metrics Calculator (SER 516) — *Java, Spring Boot, Docker, Vue.js, MongoDB, GitHub Actions*
- Developed a modular Java-based web application to compute software metrics using distributed Spring Boot microservices.
- Designed REST APIs to ingest code metadata and return computed metrics such as defect density and coupling.
- Built a Vue.js frontend with interactive dashboards to visualize metrics and historical trends.
- Implemented CI/CD pipelines using GitHub Actions to automate builds, testing, and Docker-based deployments.
- Collaborated in a 24-member Agile team, scaling the system to 16 microservices and delivering 118 story points.

Carpool: Ride-Sharing Web Platform — *Django, Python, JavaScript, SQL, HTML, CSS*
- Built a full-stack web application enabling authenticated users to create, join, and manage ride-sharing trips.
- Designed relational SQL schemas to store user profiles, trip details, and ride requests with data integrity constraints.
- Implemented backend REST endpoints in Django to handle trip creation, search, and request workflows.
- Integrated frontend views with backend APIs to support dynamic form submission and real-time data updates.

RaceLine AI – Formula 1 Analytics Platform — *FastAPI, React, PostgreSQL, Docker*
- Built a full-stack analytics platform ingesting multi-season Formula 1 data and serving results through FastAPI APIs.
- Designed backend services to support filtering, pagination, and aggregation of race and driver statistics.
- Implemented PostgreSQL schemas to store structured race data and enable efficient query performance.
- Developed React dashboards with tables and charts to visualize driver standings and championship performance.
- Dockerized backend, database, and frontend services using Docker Compose for reproducible local deployment.

Cloud-Based Face Detection & Recognition System (CSE 546) — *AWS Lambda, SQS, Docker, IoT Greengrass*
- Built a cloud-based face detection and recognition pipeline using AWS Lambda and SQS to process IoT video frames.
- Implemented event-driven workflows to asynchronously trigger face recognition and return classification results.
- Containerized Lambda functions using Docker and optimized images for CPU-based inference.
- Extended the pipeline to edge computing using AWS IoT Greengrass and MQTT for low-latency processing.

Extracurricular Experience

Smart India Hackathon 2023 — Software Edition Winner (PS - SIH1453) — *Dec 2023*
- Analyzed open-source codebase(OpenVPN) using Snyk and SonarQube to identify CVEs, security vulnerabilities, and code quality issues.
- Collaborated in a cross-functional team of 5 to document findings and assess potential impact on production systems.
- Presented vulnerability reports and remediation recommendations to company stakeholders with clear technical explanations.