Soham Nag

Email: sohamnag@asu.edu | LinkedIn: linkedin.com/in/sohamnag | Github: github.com/SohamNag

EDUCATION

Master of Science in Computer Science   *August 2022 – May 2024*
Arizona State University, Tempe, AZ   **GPA: 4.0 / 4.0**

Bachelor of Technology in Electrical and Electronics Engineering   *August 2013 – May 2017*
National Institute of Technology Meghalaya, Shillong, India   **GPA: 8.6 / 10**

TECHNICAL SKILLS

Languages: Java, Python, C, C++, JavaScript, TypeScript, HTML, CSS, Kotlin
Database: PostgreSQL, MySQL, MongoDB, Redis, Elasticsearch, SQL, NoSQL, NewSQL
Cloud: AWS (EC2, S3, Lambda, DynamoDB, RDS), Google Cloud (GCP), Microsoft Azure
Libraries and Misc.: Spring Boot, Next.js, React.js, Node.js, Express.js, Django, Redux, GraphQL, Kibana, Git, Github, NPM, Yarn, Docker, Kubernetes, Ansible, Terraform, Firebase, Apache Kafka, Mockito, Postman, Jenkins, Agile, SDLC, CI/CD, Jira, GitLab, Gradle, Maven, Linux, Data Structures and Algorithms, Object Oriented Programming, Event Driven Architecture and Software Design Patterns

RELEVANT WORK EXPERIENCE

Graduate Research and Teaching Assistant   *August 2022 – May 2024*
Arizona State University, Tempe, AZ

- **Augmented a benchmark ID dataset by 5900%** for a **Department of Homeland Security** funded research project of developing a Generative Adversarial Network based privacy preserving fake ID detection system
- Performed an in-depth comparison of the performance of PostgresML (an in-database machine learning inference framework) against Sklearn, ONNX, Hummingbird, Treelite and **published the results in a paper in ACM SOCC'23**
- Facilitated grading for **500+ students** in courses 'Database Management', 'Data Intensive Systems for Machine Learning', 'Principles of Programming Languages', 'Engineering Blockchain Applications' and 'Distributed Software Development'

PROJECTS

- **Thesis Project: GAN based counterfeit identity document detection |** *OpenCV2, Pillow, Pytorch, Python*
    - Developed a workflow to create an identity document dataset containing 20 different types of IDs pertaining to 10 US states and 10 European nations. Each type further has 4 sub-types containing different fraud patterns
    - Produced an entire dataset of 600k images which is about 60x times bigger than the similar existing dataset
- **TripSafe Safety Companion Android App |** *Android Studio, Kotlin, Chaquopy, Maps API*
    - Designed and developed an Android application using Kotlin to **enhance the safety measures** for long-haul drivers, integrating multiple real-time functionalities, as listed below.
    - Implemented **overspeeding detection** algorithm and nearest gas stations suggestions for timely refueling
    - Engineered an **automatic crash detection** feature utilizing sensor data, coupled with emergency service coordination functionality, enabling swift response in critical situations
    - Employed ML techniques to develop **driver drowsiness detection system** and implemented real-time monitoring of driver physiological parameters, including **heart rate and breathing rate**, to gauge driver wellness and stress levels
- **Spotify Clone Application |** *Next 13, React, Supabase, PostgreSQL, Zustand, Tailwind, Stripe API, Vercel*
    - Implemented **user registration, authentication**, song and album art upload, liked/disliked songs and music playback functionalities in a polished UI with a sub 150ms loading time
    - Leveraged the efficiency of Supabase and Zustand for authentication and state management. Integrated **Stripe API for secure monthly subscription** payments and hosted the app on Vercel
- **Cloud-Native Trivia Application |** *Java, Spring Boot, PostgreSQL, AWS EC2, RDS, S3, Postman*
    - Architected a trivia application using Java and Spring Boot, leveraging **microservices** architecture to ensure scalability and modularity to handle 1000+ players simultaneously
    - Implemented **REST APIs** for user authentication, retrieving questions, submitting answers, and tracking scores. Utilized PostgreSQL as the backend database managed through Amazon RDS for 40% lesser database management overhead
- **Distributed Content Delivery Network (CDN) |** *PostgreSQL, MongoDB, Python, Docker*
    - Developed a SQL and NoSQL based distributed CDN leveraging Docker containers and Docker Networking
    - Devised efficient **database replication and failover strategies** with 1 master and 2 slave nodes