Samihan Nandedkar

Chicago, IL - 60612 | 872-218-4076| svn2998@gmail.com | [Linkedin](#) | [Github](#) | [Portfolio Website](#)

EDUCATION

Master of Science in **Computer Science** | *University of Illinois at Chicago, Chicago, IL* | **GPA**: *3.6/4.0*   *Aug 2021 – May 2023*
Relevant Coursework: Computer Algorithm, Engineering Distributed Objects in Cloud Computing, *Object-Oriented Languages and Environments, Parallel Processing, Natural Language Processing, Data Mining and Text Mining*

Bachelor of Technology | *Malaviya National Institute of Technology, Jaipur, India* | **GPA**: *7.89/10*   *Aug 2016 – May 2020*

SKILLS

LANGUAGES: JavaScript ES6, Typescript, Python, HTML5, CSS
FRAMEWORKS: React.js, Vue.js, Redux, Jest, SuperTest | Apache Hadoop, Kafka, Spark | Bootstrap, Materialize
DATABASES: SQL, MySQL, MariaDB, DynamoDB
CLOUD: Amazon Web Services (AWS), Azure
TOOLS, TECHNOLOGIES, METHODOLOGIES & OS: Docker, Kubernetes, Jenkins | Git, Git Actions | Continuous Integration and Continuous Delivery | Agile, Scrum | Linux (RHEL, OEL, IBM AIX), Windows | LDAP, Azure AD, JWT

CERTIFICATIONS

AWS: [AWS Certified Developer – Associate](#), [AWS Certified Cloud Practitioner (CLF)](#) | **AZURE**: [Azure Fundamentals Certified](#)

PROFESSIONAL EXPERIENCE

Technology Solutions, University of Illinois at Chicago - Software Developer-Graduate Employee   *Feb 2022 to Present*
- Developed core backend functionality for Employee Leave Tracking System in **Node.js** with over **3000+ active daily users.**
- Designed API architecture that facilitated retrieval of data from various data sources and APIs, in multi-tenant environment.
- Implemented over **500+ unit test cases in Jest & SuperTest** to assert the core functionalities of exposed API endpoints & backend system.
- Optimized report generation by implementing **concurrent processing using JavaScript native Promises** to assist stakeholders in analyzing leave summary data of their employees which resulted in a **40% improvement in productivity.**
- Reinforced **code quality using Husky & GitHub Actions** by setting up test cases & lint checks to validate each code push to align with the project's code standardization.

Gap Inc. – Software Engineer (Compute Unix - Infrastructure Department)   *June 2020 - July 2021*
- Designed a full-stack web app in **React** and **Express** to facilitate the product teams to monitor and manage their infrastructure footprint and engage with compute teams as required.
  - Analyzed & showcased current infrastructure reach across multiple cloud vendors (Azure & OCI) and data centers using Python script to interact with cloud service exposed APIs.
  - Implemented end-to-end automation to patch a Linux server to the latest approved version at a designated time with minimum human intervention, **reducing 20 manual hours per week**.
- Automated operation testing on over **10,000+ Linux cloud instances** to make them production-ready by leveraging Ansible.
- Provided Level 3 support for Linux (OEL, RHEL 6/7, IBM Aix) servers to plan and implement critical infrastructure change tickets which impacted production-sensitive environments **without any downtime.**

Gap Inc. – Summer Intern (Compute Unix - Infrastructure Department)   *May 2019 - July 2019*
- Developed a web application in **MERN** stack to enable team members to interact with AIX HMC using Rest APIs.
- Introduced a layer of security over HMC dashboard to limit actions to users according to their authorization level in AD.
- Built a deployment pipeline in Jenkins to facilitate continuous deployment of the project.

PROJECTS

Portfolio Website: https://www.samihann.com/   HTML/CSS/JavaScript/Bootstrap CSS
Cloud Management Dashboard & Automation Scripts   React/Express/Chef/MySQL/Python/Azure APIs
Developed a front-end dashboard in React facilitating users to perform tasks in the cloud from multiple providers, such as provisioning a new cloud resource, starting, stopping the resource, reshaping the resource, and custom automation. Utilized custom ARM templates to provision the new resources required and set up a Chef environment to maintain the configuration across all the resources

Data Streaming Pipeline   AWS/Akka Actor/Kafka/Spark/Docker/ Scala
Structured an end-to-end pipeline to analyze the application logs from multiple servers and alert if any specific log entries are detected. The logs generated over multiple EC2 instances are pushed to the S3 bucket which triggers a Lambda function thereby interacting with Akka Actor System to fetch the newly generated log file and transmit it to the Spark Program through Kafka messaging system.

Domain-Specific Language for Set Theory   Scala
Created a domain-specific language in Scala to perform Set Theory operations. Implemented the major object-oriented language concepts such as variables, conditionals, classes that support inheritance, methods, scopes, and defined access restrictions in form of public, private & protected.