Samihan Nandedkar

Chicago, IL - 60612 | 872-218-4076| svn2998@gmail.com | Linkedin | Github

SKILLS

LANGUAGES: JavaScript (ES6+), Typescript, Rust, Python, HTML5, CSS
FRAMEWORKS: React.js, Vue.js, Pinia, Jest, SuperTest | Bootstrap, Materialize
DATABASES: SQL, MySQL, Oracle DB, ibm DB2, DynamoDB
CLOUD: AWS (EC2, ECS with Fargate, S3, RDS, ALB, Cloudfront, IAM), Azure (APIM, Entra ID, Key Vault, AKS)
DEVOPS: Docker, Kubernetes, Github Actions, Linux System Administration, Jenkins, Ansible
TOOLS, TECHNOLOGIES, METHODOLOGIES & OS: Salesforce Marketing Cloud (Data APIs) | Data Integration & ETL | Git | Continuous Integration and Continuous Delivery (CICD) | Agile, Scrum | Linux (RHEL, OEL, IBM AIX), Windows
AUTHENTICATION: LDAP, OAuth 2, JWT, SAML, Azure Entra ID, Single Sign On (SSO)

PROFESSIONAL EXPERIENCE

University of Illinois at Chicago – Full Stack Developer (Jun 2023 – Present) | Graduate Student Developer (Feb 2022 – May 2023)

- Developed core backend services for a high-traffic Employee Leave Tracking System using Node.js, supporting **7,000+ daily active users** and improving operational efficiency.
- Led backend development for a new enterprise-grade Employee Time Management platform, establishing scalable API contracts and development standards for student employee workflows.
- Architected and deployed a distributed system in **AWS using ECS Fargate, ALB, and RDS**, enabling autoscaling, secure service communication, and high-availability API infrastructure.
- Built automated data integration pipelines for Salesforce Marketing Cloud & Emma, improving contact sync accuracy and significantly boosting targeted campaign effectiveness.
- Built secure **CI/CD pipelines on GitHub Actions** with AWS Secrets Manager & Azure Key Vault integration; achieved **zero-downtime deployments** and eliminated manual release steps.
- Developed a comprehensive **test automation suite** (Jest & SuperTest), delivering **1,000+ automated tests** and ensuring high code reliability across REST APIs and AWS-deployed microservices.
- Integrated Azure AD-based authentication and SSO for internal and partner systems, securing access for **10,000+ users** across university applications.
- Contributed to UIC's standardized Vue.js design system, incorporating centralized auth patterns and accelerating new application onboarding.
- Applied robust security practices across data pipelines, SSO flows, and third-party integrations, ensuring compliance with university policies and regulatory standards.
- Performed performance and scalability tuning, implementing caching, load balancing, and architectural optimizations to ensure consistent user experience during peak load.

Gap Inc. – Software Engineer (Compute Unix - Infrastructure Department) *June 2020 - July 2021*

- Designed and deployed a **React/Express** full-stack web application enabling product teams to effectively monitor and manage their infrastructure footprint, facilitating improved cross-team engagement.
- Developed Python scripts for analyzing and visualizing infrastructure data across Azure and OCI platforms, significantly enhancing insights into resource distribution and utilization.
- Automated the **Linux server patching process**, reducing manual intervention by 20 hours per week and ensuring system updates during optimal maintenance windows.
- Automated **operational readiness tests** across over **10,000 Linux** cloud instances using Ansible, significantly bolstering production stability and readiness.
- Delivered expert **Level 3 technical support** for Linux servers (OEL, RHEL, IBM AIX), implementing critical updates and changes with zero downtime.

EDUCATION

Master of Science in **Computer Science** | *University of Illinois at Chicago, Chicago, IL* *Aug 2021 – May 2023*
Bachelor of Technology | *Malaviya National Institute of Technology, Jaipur, India* *Aug 2016 – May 2020*

CERTIFICATIONS

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

PROJECTS

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.