Nikhil Gande

Email: nikhilgande61@gmail.com
LinkedIn: https://www.linkedin.com/in/nikhil-gande-728862171/
GitHub: https://github.com/NikhilGande/
MyPortfolio: https://nikhilgande.github.io/my-portfolio/
Phone: +1 (816)-462-2894

SUMMARY

A highly skilled software engineer with a strong foundation in problem-solving, system design, and application development. Proven ability to collaborate effectively with cross-functional teams, delivering high-quality solutions that meet business objectives. Expertise in optimizing application performance, ensuring scalability, and maintaining robust, reliable systems. Known for attention to detail, efficient time management, and a passion for continuous learning and improvement. Strong communicator who thrives in both independent and team-oriented environments, with a focus on delivering solutions that enhance user experience and meet client need

TECHNICAL SKILLS

- **Programming Languages**: JavaScript, Java, Python, TypeScript
- **Frameworks**: ReactJS, Spring Boot, Express (Node.js)
- **Web Technologies**: HTML, CSS, Bootstrap, Tailwind
- **Databases**: SQL(Postgres, MySQL), NOSQL(MongoDB)
- **Cloud Platforms**: AWS
- **DevOps & Tools**: Docker, Git, JIRA, Maven, Postman, Apache Kafka, GraphQL

PROFESSIONAL EXPERIENCE

Associate Software Engineer. **Nov 2020 – Dec 2022**

Bank of Montreal (BMO -TCS)

Chicago, USA

Responsibilities:

- **Migrated APIs from legacy Java Servlets to AWS Lambda using Java**, improving scalability, reducing infrastructure overhead, and leveraging serverless architecture for cost efficiency and performance optimization.
- **Leveraged AWS API Gateway as the front-end to handle all incoming requests**, offloading authentication, authorization, and throttling management, while triggering AWS Lambda functions for serverless processing, reducing infrastructure complexity and enhancing security and performance.
- **Optimized Account Management System** by refactoring database queries and implementing asynchronous processing using Spring Boot's `@Async` annotation, resulting in a 40% improvement in transaction processing speed and a 25% reduction in customer complaints related to delays.
- Implemented cache layer with **AWS Redis**, reducing latency by 20% and decreasing database load, enhancing overall application performance and responsiveness.
- **Dockerized** Spring Boot application, enhancing portability and simplifying deployment, which increased deployment frequency by 30% and facilitated easier feature additions and testing, resulting in faster iteration cycles
- Deployed containerized applications to **AWS ECS Fargates**, to manage container orchestration without managing servers. This reduced infrastructure overhead, enabled seamless scaling, high availability and fault tolerance.
- Used **JIRA** for tracking tasks and managing **agile** workflows, and **Bitbucket** for version control and collaboration on code repositories.

Junior Software Engineer , Jan 2019 - Oct 2020
Coreobject Soft Systems And Services Private Limited
Bangalore, India
Responsibilities:

- **Utilized Redux to manage global authentication state** with JWT tokens stored in local Storage, ensuring login persistence across page reloads and enhancing security while reducing session-related issues by 40%
- **Implemented debouncing and Redux caching in React** to optimize API calls for search functionality, reducing unnecessary requests by 60%, improving server performance, and providing a faster, more responsive user experience
- Developed serverless **AWS Lambda functions** in **Java** to handle synchronous and asynchronous events from various sources like API Gateway, SQS, Event Bridge etc.,
- Wrote unit tests for Lambda function using **JUnit** and **Mockito** to stub external calls (95% of code coverage) to validate the code and to find edge cases and reduce production bugs.
- Created and configured **API Gateway** using **Swagger** to define HTTP methods, URL endpoints, request validation, and authentication with **API keys**. Implemented **usage plans** for rate limiting, ensuring secure and efficient API traffic management
- Used **AWS SQS** to store events from various sources and deliver them to Lambda for processing (with automatic scaling). Leveraged long polling to fetch batch data, reducing the number of calls and improving efficiency
- Used **AWS S3** to store object data in a hierarchical structure and implemented **CORS** for secure access from front-end applications. Leveraged **DynamoDB** for efficient data storage and retrieval, ensuring fast, scalable access to application.
- Leveraged AWS **CDK (Typescript)** for Infrastructure as Code (**IaC**) and implemented **Blue Green Deployment** to automate traffic distribution(V1 to V2) during new release.
- Used AWS CloudWatch to store logs, create dashboards, and generate alarms, integrated with **EventBridge** to trigger events and handle real-time errors in production.
- Used **Bitbucket** as the version control system and implemented **webhooks** to automate deployments based on repository events, ensuring efficient and streamlined CI/CD processes(**CodeBuild, CodeDeploy**).
- Utilized **SoapUI** for testing and validating the APIs, ensuring functionality, reliability, and compliance with business requirements.

EDUCATION

Master of Science in Computer Science, University of Missouri-Kansas City, MO ***Jan 2023 – Dec 2024***

CERTIFICATIONS & ACHIEVEMENTS

- **Certified Developer – Associate (Amazon Web Services)**

.

.