



# NITISH SHARMA

Software Engineer - 1

 [LinkedIn@nitishsharma](#)

 [Leetcode@nitishsharma](#)

## About Me

An experienced software engineer with expertise in Core Java, Spring Boot, AWS Services, and various frameworks. Recognised for developing highly scalable microservices and serverless applications.



+91-8427417663



n.sharma8427@gmail.com



#12830 St 20 Vishwakarma Colony  
Ludhiana, Punjab (141003)

## TECH STACK

**Core Java** (collection framework, Multithreading), **AWS**(EC2, Fargate, Lambda, API Gateway, DynamoDB, S3, SQS, SNS), **Apache Camel**, **Docker**, PostgreSQL, Redis, REST Protocol, MySQL, Design Patterns

## METHODOLOGIES

Microservices Maven, Gradle, Test Driven Development, Agile Scrum, Features breakdown, Plan Sprints for continuous Delivery, Security Scans(Blackduck, AquaSec)

## FRAMEWORKS

Spring Boot, Spring Cloud, Spring Data, Spring MVC, Spring Security, JPA, Junit, Mockito

## EXPERIENCE

### Software Engineer – 1, FICO

Bengaluru - India

June 2023 - **Present**

- Implemented APIs using **JOSE**(Javascript Object Signing and Encryption)
- **Containerisation** of the old classic monolith application. Wrote **cloudformation** template to deploy **AWS** services.
- Created a **lambda** to automate the process of creating the **production** issue tickets.
- Created **Jenkins** jobs and wrote **Ansible** playbooks to automate the deployment process.
- Upgrade multiple projects to **spring Boot 3**.

### Software Engineer Associate, FICO

Bengaluru - India

July 2022 - May 2023 (**11 Months**)

- I develop and maintain applications that provide a process management mechanism to invoke some external operations, process the data, and send responses to other microservices using **SQS**.
- Worked as a part of the **Architecture Security** team, **Designed**, and built **APIs**.
- Created a **security** feature for the Integration Module Using **JWT** authentication.
- Created **CloudWatch** matrix using **EMF** to check success and failure counts of requests in production through some widgets.

### Software Engineer Intern, FICO

Bengaluru - India

July 2021 - June 2022 (**1 Year**)

- Hands-on experience with **Microservices**.
- Wrote many **APIs**.
- Learning **AWS** fundamentals.
- Wrote **Integration Tests**.
- Wrote **Unit Test** using **Junit5**, **Mockito**.

## PROJECTS

### ProxyIM(FICO).

**Java** with **Spring boot** microservice has **4** different microservices. Provides a **mechanism** for process management to invoke some external operations, process the **transaction-related** data, and send responses to other **microservices**.

With **DynamoDB** for storing the configs, **SQS** for the async communication b/w the microservices.

### Digital-direct(FICO).

**Java** with **Spring boot** microservice provides a common communication framework across channels.

Offers one single **API** endpoint for sending and receiving **messages** across the most popular channels (**SMS**, **WhatsApp**, **Email**, **Push Notification**).

Uses 3rd party vendors like **Twilio**, **Sinch**, and **Amazon SES**.

### ServiceNow Alarm(FICO).

**Python lambda** uses a **subscription filter** that provides functionality to automatically create **servicenow** ticket for any error logged 5 times or more in an hour and then create an **AWS Cloudwatch Alarm**.

Provides the ticket where all the service-related and error-related information available.

### Compliance-UI(FICO).

**Java** with **Spring boot Serverless** application provides a view of the **security scan** results(**BlackDuck**, **AquaSec** etc) of all the microservices in a single place.

With **DynamoDB** to store the scan results.

## TECH CHANGES AND IMPACT

### ServiceNow Lambda & Alarm(FICO).

I Created a **lambda** that **reduces** the **time** to create the tickets for the **production** issues, as it **automates** the process.

All the service-related & error-related **information** comes in that ticket. **Reduces** the **manual effort** and **reduces** any human **errors**.

### SQS Grouping(FICO).

As a part of **optimizing** the critical flow on high risk and **high latency**, I introduced **SQS grouping** to group the same client case in the same queue.

It results **significantly reduce the consumer queues by 50%** and **reduce** the **latency** by a significant amount.

## AWARDS AND RECOGNITION

- Got **outstanding** rating (5 Stars) in all the appraisal cycles in FICO.
- Got the **SPOT Award** 4 times in FICO.
- Awarded **Human firewall** title for following best practices and security measures so that user's PII/PCI data can be secured.

## EDUCATION

### **Chitkara University**

B.Tech (Computer Science and Engineering), 2022  
**CGPA: 9.6**