



SIVA NARAYANA REDDY

Spring Boot / Micro Services / AWS / Docker / Cassandra / PCF

Fast paced Spring Boot Developer with 6 years of experience on the very latest versions of spring modules and Java 8.

I specialize mainly in Micro Services, the most successful and widely adopted architectural style, with Spring Boot and Spring Cloud.

Looking for challenging roles like technical or module lead to mentor rookies and working towards high quality delivery in Java and Cloud space.



CONTACT ME



Telecom Nagar, Hyderabad, IN



+91 81066 24427



narayan-sambireddy@outlook.com



narayan-sambireddy.github.io



github.com/narayan-sambireddy



bitbucket.org/narayan-sambireddy



linkedin.com/in/narayan-sambireddy



stackoverflow.com/users/9678169

INTERESTS

Mythologies

Entrepreneurship

STRENGTHS

Trustworthy

Persistence

INSPIRATION

My Dad & Mom

SIGNIFICANT ACHIEVEMENTS

- ❖ Star Performer at Apple.
- ❖ Kick-started and driven a critical project from E2E and delivered with impregnable quality.

EDUCATION

2012 – 2014

Master of Software Engineering
BITS PILANI

Pursued MS Software Engineering while working at Wipro.

AREA OF EXPERTISE

JAVA [1.7 - 1.8 & 1.9]

Worked on the latest versions of Java with all the latest features of functional programming and modularity in the implementation avoiding monolithic design.

MICROSERVICES

Developed microservices to accelerate delivery by minimizing communication and coordination between the teams while reducing the scope and risk of change.

SPRING BOOT [1.2 – 2.0.2] / SPRING CLOUD / 12-FACTOR APP

Implemented many microservice applications leveraging spring boot and its starter modules: integration, batch, web-services, jdbc, data-jpa, data-rest, web, webflux, security.

Worked with spring cloud, to enable coordination between the microservices in a distributed middleware system and to make these resilient and scalable by implementing most of the 12 factors.

AMAZON WEB SERVICES

Worked towards migrating legacy batch applications to AWS using the services: Lambda, EC2, Batch, S3 and RDS.

APACHE CASSANDRA (DATASTAX DIST.)

Worked on complex modelling in Cassandra migrating DB2 and Oracle schemas and the read and write queries.

PIVOTAL CLOUD FOUNDRY

Worked on implementing PASS and CAAS solutions using PCF with on premise DC.

WORK HISTORY

Aug 2012 – Apr 2017

Wipro Technologies

Client:



1. BRUCKE PLATFORM

DESCRIPTION:

Brucke enables various business processes and functions to operate on an integrated platform. Various groups in and out of Apple Online Store (AOS) can take advantage of Brucke's Security, Messaging and all the reusable platform components to support their business.

BRUCKE ADMIN:

- Authentication via Apple Connect
- API and Admin console for Module specific authorization

BRUCKE MESSAGING CLIENT (BMC):

- Common Interface for Solace and Geneva message brokers.
- Multi-threaded Publishing and Consuming
- Queue Browser to View Payload and Message count.

BRUCKE CORE:

- Hosts all the database related configuration
- Entity and Repository mappings
- Includes Espresso Alerting to raise alerts on failure

BRUCKE FEED PROCESSOR (BFP):

- Microservice implementation of BMC Consumer.
- Designed to consume messages from a given endpoint and process them
- The statistics are published to Cassandra Database.
- Multiple instances of this microservice each of a specific processing logic.
- The processing logic can be plugged in as a Maven dependency.

ROLE: Spring Developer

CONTRIBUTION

- Built reusable components and libraries for future use across entire Online Store.
- Completely involved in the development and testing of each and every modules with Spring and Maven.
- Written unit and integration test cases using Junit, spring-test and Mockito.
- Implemented integration gateways using spring integration for inspecting the headers and routing the messages to specific channels for further processing.
- Contributed in utility APIs for Cassandra that resemble Spring JPARepository.
- Implemented security layer using spring security LDAP with (Apple Directory)
- Migrated spring security from LDAP to Apple Connect SSO through VPN.
- Written Kafka implementation for Brucke Messaging Client.

2. QUOTE MACHINE / QUOTES CENTRAL

DESCRIPTION:

QuoteMachine is the master data management tool for Commit Code / Shipping Availability and NPI (New Product Innovation) readiness events.

It is the system for assigning initial quote (Shipping Availability) for products during an NPI.

QUOTE MACHINE SERVICE (QMS):

- Simplify the Commit Code setup including all messaging, translations and related master data to support all corresponding systems execution during NPI.
- Collect all relevant master data into one management system with allows for a comprehensive setup and validation process.
- Decrease the time to production for creating, updating and disseminating information to downstream execution systems.
- Increase the reporting and confirmation process to ensure correct data management.

QMS CORE:

- Database related configuration and implementation to connect to Oracle.
- Written using Spring Data JPA

QMS CASSANDRA CORE:

- Configuration and implementation to operate on Cassandra keyspaces.
- Written using Datastax APIs

QMS STATS COLLECTOR:

- Built on top of BFP with specific processing logic each for commit-code, translations, etc.
- This component listens to Solace endpoints for messages and processes them.
- It handles the status updates during message processing and updates the same in Cassandra through QMS Cassandra Core.
- Collates the processing status from various subscribers.

ROLE: Spring Developer

CONTRIBUTION

- Kick-started the project from scratch which involved modules viz. QuoteMachineService, QMSCore, QMSStatsCollector, QMSCassandraCore.
- Independently developed the entire QMSStatsCollector module, which listens to Queue (Geneva/Solace) endpoints for messages and processes with the provided processing logic.
- Independently developed the entire module of QMSCassandraCore.
- Automated the Cassandra querying and filtering by building a Cassandra easy filter component.
- Migrated the sources from Java 7 to Java 8 and did refactoring wherever necessary.
- The entire application was written in Spring Boot and Java8.
- Built docker images for the modules to improve delivery speed.
- Contributed in doing release cut for the project through jgit plugin.

Apr 2017 – Aug 2017

Cognizant Technology Solutions

Client: **verizon**

DISTRIBUTED VISION SYSTEMS (DVS): Order Re-Wire

DESCRIPTION:

Verizon, being the 3rd largest customer of IBM's Mainframe Systems, had to pay a huge sum for the infrastructure and support. It has lately realized that now is the time to move on and started liberating itself from the clutches of legacy and monolithic systems by focusing on Open Source Frameworks and Micro Services and Cloud Space.

The end goal of the assignment is to bring down the MIPS of the Vision System for the Order Re-wire API that involves several other internal API calls.

ROLE: Micro Service Developer

CONTRIBUTION

- Identify the internal APIs and targeted tables of main frame DB2.
- Reverse engineer the mainframe DB2 schema and replicate the closest possible Cassandra schema (keyspaces with similar tables)
- Breaking down the existing monolithic services that are interacting with Vision Systems into smaller APIs so they can be implemented as microservices.
- Implementing microservices for the identified APIs using spring boot.
- Composing Docker file for the microservices to build docker images.
- Converting the DB2/Vision queries into Cassandra queries leveraging TypeDef framework written in groovy.
- Implemented Service Discovery using Eureka for collaboration between Microservices.
- Implemented Hystrix and Feign Client to improve resiliency.

1. Transaction Data Store (TDS)

DESCRIPTION:

TDS is all about decomposing the existing monolithic transaction systems maintained by third-party vendors viz. MoneyThor, Tibco, Finacle into microservices.

The goal is to create a caching layer around Finacle and decouple / eliminate MoneyThor and reduce the dependency on Tibco / SOI.

MODULES:

Moneythor Resiliency, Transaction History Finacle Event, Account History Finacle Event

2. Banca (Banking Agent)

DESCRIPTION:

Banca is an agency system where we collaborate with other service providers and sell their products to our account holders. As a result DBS earns commission of agreed percentage for each product sale.

MODULES:

Application Data Service, Banca Execution Service, Banca Commons, Banca Audit Service

ROLE: Micro Service Developer

CONTRIBUTION

- Kickstarted the Banca project with microservice approach.
- Implemented config server, config client, service discovery and circuit breaker features.
- Migrated the batch workflows from using TWS (IBM's Tivoli Workload Scheduler) to AWS Batch.
- Implementing an auditing micro service for Banca project to track the prospect data.
- Written microservices to work in parallel with SOI APIs and some to replace SOI APIs.
- Written a batch for retrying to obtain the acknowledgement on payment successful.
- Written a batch for pushing the purchase / dropout prospect details as EOD batch file to Auditing systems.
- Worked on PCF for deploying the microservices.
- Written Lambda functions to trigger batch execution on the arrival of EOD batch file.