

Abhinaya D (Sr. JAVA FULL STACK DEVELOPER)

☎ : 404-939-0395 ✉ : dabhinaya3@gmail.com

PROFESSIONAL SUMMARY

- ✓ Professional IT experience in delivering enterprise solutions in multiple domains like **Health, Finance, Enterprise**, and other **related domains**. Expertise in conceptualizing, designing and coding technical solutions using **Java/J2EE** and UI technology stacks to satisfy complex business problem statements.
- ✓ Extensive experience in designing UI web applications using front-end technologies like **HTML5, CSS3, JavaScript, XML, DHTML, jQuery, AJAX, JSON, AngularJS, Angular, React, Bootstrap**.
- ✓ Experience in developing User Interface (UI) Rich Web Applications, and Web Service Applications using **Angular modules, services, templates, directives, pipes, services**.
- ✓ Experienced in **React JS** and working with **React Flux** architecture. Experienced in working with **Redux** architecture.
- ✓ Experience in Test driven development (TDD) using unit testing frameworks such as **Jasmine Protractor, Karma, JUnit, QUnit** and **Selenium**.
- ✓ Expert in Core Java with strong understanding of **OOP concepts, Collections, Multithreading, Event handling and Exception handling**.
- ✓ Hands on experience in using Java 8 features like Lambda Expressions, Functional Interface and Stream API.
- ✓ Designed and implemented an event-driven microservices architecture using **Apache Kafka**.
- ✓ Strong understanding of **Microservice patterns Hystrix, Proxy chaining**.
- ✓ Hands-on experience in 3 competing and cross-pollinating paradigms of microservices - Spring Boot, JEE(Micorprofile) and Nodejs - as well as their Reactive offshoots (Spring 5 **WebFlux**) and event-driven variants.
- ✓ Extensive Knowledge of how Web Semantics translate into **REST API's** and Developing REST API's using Java and Spring.
- ✓ Strong experience in **Spring Framework** modules such as Spring MVC, AOP, JDBC, Spring Boot, Spring Microservices, Spring REST, Spring Ribbon, Spring Netflix.
- ✓ Expertise in J2EE Architecture using **Java, JDBC, Servlets, JMS, JSP, Hibernate, Maven, Spring, Web services (JAX WS, JAX RS)** and Rule Engines (Drools, Easy Rules, Rule Book).
- ✓ Experience in Messaging services like **(JMS) MQ's Rabbit MQ, IBM MQ, Apache Kafka and Active MQ**.
- ✓ Deployed the Application into **Docker** container and made easily accessible at runtime using **Pivotal Cloud Foundry** and other cloud services like **AWS, Netflix Eureka, Spring Hystrix** as fallback method, **Spring Ribbon** for load balancing and a broker service to maintain status using **Spring Kafka** and **Zookeeper**.
- ✓ Strong Knowledge on different databases like **DB2, Oracle, My SQL, NoSQL Database (MongoDB and Cassandra)** and writing **PL/SQL** packages, procedures, Functions, Views, triggers using **SQL Plus**.
- ✓ Good knowledge in employing database caching strategies using **MongoDB Elastic search**.
- ✓ Experience in unit testing the applications using **JUnit, Mockito, Postman, SOAP UI** and developed logging and standard mechanism based on **Log4J, Slf4j** and good experience in shell scripting.
- ✓ Hands on experience with **build tools** like **Maven, Gradle** and logging tools like **Log4J**.
- ✓ Experience in using DevOps tools such as **Jenkins, Docker** for Continuous Integration & Deployment for Microservices.
- ✓ Good knowledge of designing, deploying, and operating highly available, scalable and fault tolerant systems using **Amazon Web Services (AWS)** and **Azure**.
- ✓ Expertise on advanced software development methodologies such as **Agile/Scrum**, Waterfall, Test Driven Development (TTD), Incremental and Iteration methodology.

TECHNICAL SKILLS

Languages	Java, SQL, PL/SQL
Web Technologies	JSP, JavaBeans, Java Servlets API
Frameworks	Spring, Hibernate, Angular, Spring Security, Spring MVC, Spring Boot.
Design Patterns	Singleton, DTO Pattern, DAO Pattern, Delegate Pattern, Service Facade, Factory Pattern, Front Controller Pattern, MVC Pattern.
Web servers	Apache Tomcat, IBM WebSphere, JBOSS
NoSQL Databases	Oracle, SQL server, MySQL, DB2, Mongo DB, Cassandra.
Cloud Environments	Azure, AWS, Kubernetes, Docker, PCF.
Messaging	JMS MQ, IBM MQ, Rabbit MQ, IBM MQ, Apache Kafka
Testing Tools	JUNIT, JMeter, Mockito, TestNG, Jasmine, Karma
Build Tools	Ant, Maven, Gradle, Jenkins
Source Control	GIT, Bit Bucket, GitLab, Github.

PROFESSIONAL EXPERIENCE

SOUTHERN COMPANY, ATLANTA, GA
Sr. Java Full Stack Developer

Jan 2021 To Present

Responsibilities:

- ✓ Developed UI pages using HTML5, AngularJS, Bootstrap, CSS3, JavaScript, jQuery, AJAX and Nodes.
- ✓ Used **React JS** for templating for faster compilation and developing reusable components.
- ✓ Develop various screens for the front end using React JS and used various predefined components from NPM (Node Package Manager) and redux library.
- ✓ Used **ReactJS** to implement rich functions in various pages: form validation, grid list, search, sort, pagination, multi-selection, tags-input, advanced/custom directives, authentication, unit tests, etc.
- ✓ Used **ReactJS** to create views to hook up models to the DOM and synchronize data with server as a Single Page Application (SPA).
- ✓ Integrated **Redux framework** to scale and maintain the application better while handling large amounts of data.
- ✓ Worked on security updates and issues found with **Vue.js** and all project dependencies.
- ✓ Developed forms for user event handling and validations using **JavaScript** and **AJAX**.
- ✓ Integrated testing methods implemented by using **Karma** and **Jasmine**.
- ✓ Implementing or exposing the **Micro service** architecture with **Spring Boot** based services interacting through a combination of **REST** and **Apache Kafka** message brokers.
- ✓ **Spring cloud** is implemented along with the **microservices architecture**.
- ✓ Worked on Spring Ribbon for load balancing and used fail safe methodology using Circuit Breaker pattern with **Hystrix/Dashboard** on designing and implementing **Microservice** applications.
- ✓ Improved data retrieval by implementing **RESTful** API and interfacing with corresponding **OLTP** and **OLAP**.
- ✓ Worked on **RESTful** web services, exclusively consuming REST API with **Asynchronous behaviour**.
- ✓ Designed and Developed Custom Connector Software in Java to fetch data from third-party security applications using **RESTful** webservices, storing in **Kafka** Message Queues and performing analytics for streaming data in real time using **Apache Spark**.

- ✓ Developed **Rest API's** to processes the data from Database to another Rest Service and created containers for the API's using Docker.
- ✓ Used Annotations for **Spring DI, Auto wiring** and Spring MVC for REST API s and **Spring Boot** for **Micro services**.
- ✓ Implemented pre-authentication and data base security using **Spring Security** and securing application access and **Spring Batch** for scheduling and maintenance of batch jobs.
- ✓ Developed the applications using **Spring** framework including **Spring Core, Spring MVC, and Spring Integrations design pattern**.
- ✓ Created **Micro services** using **spring boot, spring security** and deployed to **AWS cloud**.
- ✓ Worked on creation of custom **Docker** container images, tagging, pushing images, integration of **spring boot**.
- ✓ Worked on the **OAuth** for accessing the cloud storages like Box, one drive, Google Drive, Drop box.
- ✓ Implemented **Kafka** producer and consumer applications on **Kafka** cluster setup with help of **Zookeeper**.
- ✓ Implemented **MongoDB** as data storage and utilized aspects like replica sets, sharding and clever document design to make service extensible scale and feature wise.
- ✓ Using **Swagger UI** registered micro services, monitored service health check from Spring Boot admin console.
- ✓ Auto generated documentation for Rest API calls using **Swagger**.
- ✓ Worked with several testing frameworks such as **Junit** and **Mockito**.
- ✓ Used **Log4j** for logging and debugging and JUnit extensively for testing. Deployed application using AWS services.
- ✓ Used **Maven** scripts for build and deployment of the application and **GIT** as the secure Versioning software. Utilized **Bitbucket** to manage the software version and **Rally** for issue tracking.
- ✓ Worked on generating **Docker Images** and running them on Docker container.
- ✓ Experience with **DevOps** methodologies, implementing **CI/CD Pipelines** using tool sets like **GIT** and **Jenkins**. Setup the continuous Integration (CI) process and continuous deployment (CD) jobs in **Jenkins's** brew master instance, which reduces the time for the deployment of **micro services**.
- ✓ Implemented **AWS** solutions using **EC2, S3, RDS, EBS, Elastic Load Balancer, Auto scaling groups, Optimized volumes** and **EC2 instances** and created **monitors, alarms** and **notifications** for **EC2 hosts** using **Cloud Watch**.
- ✓ Used **AWS** with the **AWS SDK** for Java for taking the complexity out of coding by providing Java APIs for AWS services including Amazon S3, ECS, DynamoDB, AWS Lambda, and more.

Environment: HTML, CSS, React, Java, JavaScript, Karma, Jasmine, Microservices, REST API, Spring Boot, Spring, JWT, Kafka, MongoDB, Swagger, Junit, Mockito, Maven, Git, Jenkins, Docker, AWS, JSON, XML.

QUEST DIAGNOSTICS, SAN DIEGO, CA

Mar 2020 To Dec 2020

Java Full Stack Developer

Responsibilities:

- ✓ Developed dynamic web pages using **HTML5, CSS3, Bootstrap** and **AJAX**.
- ✓ Developed multiple **Angular** components which are compatible with latest versions of **typescript, angular**.
- ✓ Developed **single page applications (SPA)**, components using **Angular directives, controllers, view and routing, service modules**.
- ✓ Involved in web application development for backend system using **Angular, Angular CLI, Webpack, HTML5, TypeScript, SCSS**.
- ✓ Used **Angular** Reactive forms to implement complex **forms** and performed the validations using Form Builder.

- ✓ Defined **Angular routes** using **Router** module for components at module level to enable lazy loading for modules and components.
- ✓ Extensively worked on **Java 8** features like **lambda expressions** for communicating between business layer and database, **Stream API** for Bulk Data Operations on Collections using both streams and parallel streams.
- ✓ Implemented fail safe and **circuit breaker** patterns in **Spring Boot Microservice** applications using **Hystrix**, configured **Swagger-UI** to registered Microservice with **Eureka server** and monitored service health check from **Spring Boot** admin console and **Spring Ribbon** for load balancing.
- ✓ Defined **Hibernate** Named Queries for Batch related operations while defined Criteria Queries for **Micro services**.
- ✓ Developed **Spring Boot REST API** end points implementing messaging queue for a supply chain delivery system
- ✓ Worked on building reusable framework for consuming and providing **REST API's** using Java, Swagger.
- ✓ Worked on **Spring** Framework for **dependency injection**, **integration** of **Hibernate** using **Spring ORM support**, development of **RESTful web services** to give **JSON** output.
- ✓ Developed and Integrated of REST APIs and services using Fetch API.
- ✓ Worked on **Spring cloud** for Routing, Service-to-service calls, Load balancing.
- ✓ Developed Application using **Spring MVC Framework** by implementing Controller, Service classes. Implemented spring framework **Bean Factory** and application Context for bean initialization and configurations.
- ✓ Used **Spring MVC Transaction Management and Hibernate cache** concepts.
- ✓ Extensively worked on **Hibernate** concepts such as Lazy loading, Dirty checking, Locking, and Transactions.
- ✓ Worked with **Hibernate** in **persistence layer**, **Hibernate** mappings using **Annotations, HQL, Criteria and Queries**.
- ✓ Used **OAuth** authentication protocol respectively for **security** and **authorization**.
- ✓ Implemented **RabbitMQ** message queuing for post processing after submission.
- ✓ Planned and programmed a work queue system on **Angular** with **RabbitMQ**. It uses Express to create a **REST API** allowing users to create jobs and retrieve data from **MongoDB**.
- ✓ Worked on Unit testing of application and actively participated in debugging using **JUnit, Mockito**, and built the packages using Maven built tools and deployed the application on **Oracle WebLogic** Application Server locally, as well as coordinated with the DevOps team to deploy the application onto cloud using **AWS**.
- ✓ Used **Postman** to test the RESTful API for HTTP requests such as **GET, POST, and PUT**.
- ✓ Developed Unit test cases using **JUnit, Mockito** and Easy mock.
- ✓ Developed unit test cases using **Mockito framework** for testing accuracy of code and **logging** is done using **SLF4j** and **Log4j**.
- ✓ Worked with version control systems like GIT and used Source code management client tools like **GitLab**.
- ✓ Integrated GitLab into **Jenkins** to automate the code checkout process.
- ✓ Worked Hands on with AWS products such as **AWS API Gateway, EC2, EBS, AWS Lambda, ECS, EKS**.
- ✓ Used Amazon Cloud Watch to monitor **AWS** services and Amazon Cloud Watch logs to monitor application
- ✓ Designed REST-full XML Web Service for handling AJAX requests.

Environment: HTML, CSS3, Angular, Java8, JavaScript, Karma, Jasmine, Microservices, REST API, Spring Boot, Spring, Hibernate, JWT, RabbitMQ, MongoDB, Postman, Junit, Mockito, Git, Tomcat, Maven, Jenkins, AWS, JSON, XML.

Responsibilities:

- ✓ Worked with Reducers, React Containers to update the state of an App produced by Redux and by mapping those to the React component using React-Redux.
- ✓ Created **React JS** components using **JSX**, Followed **Redux Architecture** for unidirectional data flow and manage your data in React and Make HTTP requests in **React & Redux**.
- ✓ Worked on configuring Spring MVC with Angular using Request Mapping.
- ✓ Expertise in creating User Controls, Web Forms and client-side validations using JavaScript, jQuery Validate and Angular-JS. Experienced in using Backbone-JS to connect the API's which enforces the communication to the server.
- ✓ Used JavaScript for the web page validation and Struts Validator for server-side validation of data.
- ✓ Used advanced java 8 features like **Lambda expressions** for array operations, **Streams**, **Method reference**, **collections**. Enhanced concurrency methods etc. throughout the application for well-defined programming.
- ✓ Worked on testing Web Services using **SoapUI** tool, request, and response xml.
- ✓ Designed and developed Microservices based business components and RESTful service endpoints using **Spring boot**
- ✓ Implemented Spring boot microservices to process the messages into the **Kafka** cluster setup.
- ✓ Developed Micro services using RESTful web services to provide all the **CRUD** capabilities.
- ✓ Developed **several REST web services** which produces **both XML and JSON** to perform tasks, leveraged by both web and mobile applications. **RESTful web services** using Jersey for JAX-RS implementation.
- ✓ Integrated with the back-end Restful API calls to make the application more dynamic and responsive with the best user experience.
- ✓ Exposed and consumed **SOAP based Web Services** to communicate between systems.
- ✓ Configured **Swagger** for auto-generated documentation for all REST API calls.
- ✓ Developed and deployed Java **REST** web services using **Spring Framework**.
- ✓ Worked with **SOAP UI, JAX-WS** to create Web Services and consuming the existing services.
- ✓ Developed batch processing using **Spring Batch** in the persistence layer & improved performance of batch processes using **Multithreading** with **Executor Framework** for control of asynchronous tasks in parallel threads.
- ✓ Developed **micro services** architecture using Rest APIs, **spring boot** and **pivotal cloud foundry**.
- ✓ Worked on **Hibernate** as an Object Relational Mapping (ORM) Tool for the backend data persistence.
- ✓ Developed and maintained the data layer using Hibernate and used **Spring ORM** to integrate with Hibernate.
- ✓ Developed the Hibernate mapping files and involved in writing **JUnit** test cases and suits using **Eclipse IDE**.
- ✓ Implemented Spring boot microservices to process the messages into the **Kafka** cluster setup.
- ✓ Developed data transformer for converting legacy data from Source tables to Avro format and fed into **Kafka** using **Camel**.
- ✓ Worked on Single node/Multi-cluster setup using technology **NoSQL** and **MongoDB**.
- ✓ Extensively used **JUnit, JMeter, SoapUI** for testing.
- ✓ Conducted a testing of various unit layers using JUnit with Easy Mock and created automated test scripts for integration testing using Selenium and configured unit test cases using Junit
- ✓ Experience in using **Maven, Gradle, Ant** builds and in Visualization tools like **Tableau**.
- ✓ **GitHub** has been used as a Version Controlling System.
- ✓ Deployed, Scaled, Configured, wrote manifest file for various Microservices in **PCF**.
- ✓ Experience in Micro services development using spring boot and deployment in Pivotal Cloud Foundry (PCF).

Environment: HTTP, CSS3, Angular, JavaScript, SOAP, Spring MVC, Hibernate, JMS, DB2, Oracle, Git, WebLogic, PCF, AWS.

Java Developer**Responsibilities:**

- ✓ Involved in creating dynamic web site design and programming using **HTML5, CSS3, JavaScript and jQuery**.
- ✓ Built **Web pages** that are rich user-interface using **AJAX, JavaScript and AngularJS**.
- ✓ Created custom Plugins, templates, and functions for Angular JS application.
- ✓ Developed Angular services making use of resource service to perform REST API calls to the backend.
- ✓ Involved in developing **HTML and JavaScript** for client-side presentation and, data validation within the forms.
- ✓ Used **Hibernate** framework for back-end development and **Spring DI** for middle layer development.
- ✓ Implemented **Spring MVC** (Model View Controller) to handle/intercept the user requests.
- ✓ Used **JMS** for asynchronous exchange of message by applications on different platforms.
- ✓ Uses the **DAO** pattern for data access layer and JMS API for the messaging requirement.
- ✓ Configured the Web sphere application server to connect with **DB2, Oracle and SQL** Server in the back end by creating JDBC data source and configured **MQ Series with IBM RAD and WAS** to create new connection factories and queues.
- ✓ Created new **RESTful** web service operations, modified the existing web service, and used **SOAP UI tool** to test the **RESTful** web service operation.
- ✓ Monitoring and managing the **WebLogic** server instance using WSLT and for the automation purpose.
- ✓ Extensively used **Git** for version controlling and regularly pushed the code to **GitHub**.
- ✓ Involved in the configuration of **Spring, Pivotal cloud foundry (PCF), Micro Services Framework and Hibernate mapping**.

Environment: HTML, CSS3, Angular, Java8, JavaScript, Microservices, REST API, Spring Boot, Spring, Hibernate, JWT, RabbitMQ, Oracle, Postman, Junit, Mockito, Git, Tomcat, Maven, Jenkins, AWS, JSON, XML.

Java Developer**Responsibilities:**

- ✓ Involved in complete software development life cycle including **design, development, testing, monitoring logs, debugs and trouble shoot and fix the problems in various environments in AGILE Methodology**.
- ✓ Server-side programming is done using **java servlets, and EJB**.
- ✓ Translated the entire business model of the **Mutual Funds Operations** into Object Oriented models.
- ✓ Converted **XML** data into HTML specification to be displayed in the webpage.
- ✓ Coded **Java Beans with Eclipse IDE and implemented the logic for navigation between JSP pages using MVC design patterns**.
- ✓ Implemented **Struts framework (Action & Controller classes)** for **dispatching** request to appropriate classes.
- ✓ Used simple **Struts Validation** for validation of user input as per the business logic and initial data loading.
- ✓ Used **Spring Beans** to encapsulate business logic and Implemented Application MVC Architecture using **Spring MVC framework**.
- ✓ Designed & developed various **business services and methods** utilized by online web applications for fulfilling their functionalities using **Spring**.
- ✓ Developed **Controller's and Validator components** using **Spring MVC framework**.
- ✓ Used **REST** methodology using **HTTP Handlers** in developing **Web Service** replication interface for setting up data feeds for content transportation in **XML and JSON formats**.

- ✓ Developed the code through different Layers presentations layer using **Struts**, **Business Layer** using **Spring IOC**, **Service Layer** using **Hibernate ORM framework**.
- ✓ **HQL queries** were developed to retrieve data using **Hibernate**.
- ✓ Developed **Controller Servlets**, **Action** and **Action Form objects** for process of interacting with **MySQL** database using **Struts**.
- ✓ Develop back-end business/application logic through **Session EJBs**.
- ✓ Developed **Database Procedures** for **updating, Inserting, and deleting table for a complete logical Unit of Work (LUW)** using **PL/SQL**.
- ✓ Involved in **system testing and integration testing**, and trouble shoot issues in multiple environments.
- ✓ Deployment and developing **Enterprise Java Beans** with **WebLogic Application server** in the **UNIX Environment**.
- ✓ Involved in **JUnit** testing for various components of applications like Service layer and DAO layer.
- ✓ Used **Putty** and **WinSCP** tools to check the log files on Unix server in case of debugging an issue
- ✓ Prepared **complete documentation** for developed portion of the system.
- ✓ Used **Rational Clear Case** as version control system.
- ✓ Used **SVN** for version control and **LOG4J** for debugging.

Environment: Java, J2EE, EJB, HTML, CSS, AJAX, JQuery, Angular JS, XML, JSON, JSP, Servlets, EJB, Struts, Hibernate, Spring, PL/SQL, JDBC, WebLogic, My Eclipse, MAVEN, JUnit, Notepad++, SVN, Pivotal Tracker.