Devashri Gaikwad

Sr.Full Stack Developer

 [devashri.gaikwad.n@gmail.com](mailto:devashri.gaikwad.n@gmail.com) **Mob : 9842623024**

Goal Driven Sr Full Stack Developer with 8 Years of experience in design, development and implementation of software applications using **Java, React, AWS** technologies with expertise in the analysis, design, and coding phases of the Software Development Life Cycle (SDLC)

PROFESSIONAL SUMMARY

* Front-end technologies with practical expertise include **Angular12**, ReactJS, React Native, Bootstrap, HTML5, NodeJS, Graph-QL, JavaScript, ES6, Vanilla JS, CSS3, Kafka, jQuery.
* Proficiency in leveraging the Mongo, Express, Angular, and **Node JS.** Full Stack development toolkit to create Dynamic Single Page Applications (SPA).
* Extensive experience in working with Promises and mastering asynchronous programming. P
* Proficient in handling asynchronous operations, such as API calls and database interactions, to ensure smooth and non-blocking execution of code
* Proficient in JavaScript design patterns, employing industry best practices to ensure scalable and maintainable code.
* Extensive experience with modern JavaScript libraries and frameworks, such as React or AngularJS, to develop robust and efficient web applications.
* Utilized third-party libraries such as Redux and React Navigation to simplify state management and navigation within mobile applications.
* Proficient in React Native framework, **Redux state management**, and React Native component libraries like Native Base and React Native Elements.
* Skilled at using the Document Object Model to create JavaScript data capture and event **processing (DOM**).
* Collaborated with designers and product managers to create user interfaces that provided a smooth and intuitive experience for users.
* Hands-on experience in integrating Apache Kafka with ReactJS for real-time data updates and event-driven communication.
* Skilled in creating J2EE enterprise applications using Micro Service Architecture and Service Oriented Architecture.
* J2EE Technology expertise, including Servlets, Struts, Spring, JDBC, JTA, JPA, Hibernate, and JMS
* Practical knowledge of how to create web applications using the Spring framework components Spring IOC, Spring AOP, Spring MVC, Spring Security, Spring Micro services, Spring REST, Spring Boot, Spring Batch, and Spring Data.
* A thorough understanding of recommended methods for Software Configuration Management (SCM) in the Agile, SCRUM, Waterfall, and Test-Driven Development TDD approaches.
* Proven ability to build REST (Representational State Transfer) proxy microservices that call SOAP (Simple Object Access Protocol) applications on the back end.
* Expertise with Docker images using a Docker file. creation of master and node for the Kubernetes orchestration of Docker and Container images.
* Experienced in designing and building highly available high-volume messaging infrastructure with Apache Kafka on
* Experience working with modern IDEs, including Visual Studio Code and IntelliJ.

**Professional Experience**

**Client: Strategic Healthcare Programs Jan 2021- Present**

**Location: Atlanta, GA**

**Role: Sr Full Stack developer**

**Responsibilities:**

* Adhered to SAFe agile methodologies to encourage iterations, collaboration, and process adaptation.
* Used **React16 and Node JS** to build a single page application that loads numerous views using route services to make it dynamic.
* Worked with **React JS and React views**, rendering utilizing parts that contain other parts known as custom HTML tags.
* Created client-side user interfaces using React JS and Redux Saga for building SPA.
* Developed a responsive and user-friendly single-page application (SPA) for a payment and billing system using Vue.js and React.
* Worked with the product team to define and prioritize payment processing and billing management features, ensuring the Java based microservices are meeting the requirements.
* Implemented secure authentication and authorization using React Router 6, ensuring role-based access control for different user types.
* Developed responsive and visually appealing user interfaces using **SCSS** styling, enhancing the overall user experience.
* Strong knowledge of React and Redux principles, including React Hooks, state management, and action creators.
* Developed high-quality, reusable, and scalable components and a front-end solution using **React.**
* Implemented the payment feature, utilizing ReactJS chips to create a visually appealing and user-friendly interface for selecting payment options.
* Migrated from rest to GraphQL and improved GraphQL APIs by adding new features and fixing bugs based on feedback from users and stakeholders.
* Integrated third-party payment gateway s PayPal into the software, enabling secure and efficient payment processing and authorization.
* Developed a billing summary component that dynamically calculates and displays the total amount, taxes, and discounts applied to provide users with a clear overview of their charges before finalizing the payment.
* Implemented a server-driven UI architecture using frameworks like React to dynamically render UI components based on server-side configuration.
* Leveraged JSON -based configuration files to define the structure and behaviour of UI components, enabling easy customization and flexibility without code changes.
* Implemented caching mechanisms to reduce server load and improve response times for frequently accessed UI components.
* Designed and implemented an interactive data visualization using D3.js, providing an intuitive representation of complex financial data.
* Developed custom D3 components and implemented various data visualization techniques, such as bar charts, line graphs, and heatmaps, to present data insights effectively.
* Collaborated with designers and backend developers to ensure seamless integration of frontend and backend.
* Conducted code reviews and provided feedback to team members to ensure code quality and consistency.
* Optimized the design process for a B2B software platform, using Figma and Figma Library to streamline design handoffs and improve collaboration between the design and development teams.
* Conducted design reviews and provided feedback to other designers on the team, using **Figma and Figma Library** to ensure that designs were consistent with the overall design system and best practices.
* Web services and Restful interfaces were used to produce and consume data from other subsystems.
* Implemented responsive layouts to ensure seamless user experiences across various devices and screen sizes.
* Developed and maintained a robust design system, streamlining UI components and design patterns for efficient and consistent development.
* Spearheaded the creation and evolution of a future design system to support the team's long-term project goals and scalability.
* Successfully implemented localization/internationalization systems, allowing the application to cater to a global user base while addressing potential cultural and language-related challenges.
* Troubleshot and resolved issues related to internationalization, ensuring a seamless experience for users in different regions.
* Worked on Java 1.8 features like Lambda Expressions, which use Functional Interfaces to make anonymous inline function calls, making code more legible and concise.
* Worked with fundamental Java concepts like memory management, the Collections Framework, and multithreading.
* Implemented authentication and authorization mechanisms using OAuth 2.0 and JW tokens, ensuring secure access to microservices and protecting sensitive data.
* Demonstrated expertise in writing clean, efficient, and scalable code using the latest ECMAScript standards. Well-versed in leveraging the power of JavaScript to build robust and dynamic web applications.
* Implemented a microservice architecture using Java, Spring Boot, and Docker to improve scalability and maintainability of the application.
* Developed a payment billing software using Java, employing both functional and reactive programming paradigms to enhance code readability and maintainability.
* Designed and developed a RESTful API microservice using Java and Spring Framework, enabling seamless integration between different components of the system.
* Using Spring Boot, a microservices architecture was implemented to make the application more autonomous and compact.
* Spring Integration workflow files were used to integrate the application with Microservices.
* Various Spring Framework modules, including Spring MVC, Spring IOC, Spring AOP, Spring Boot, and Spring Security, were used to develop the application.
* To test the REST (Representational State Transfer) APIs (Applications) for the GET, POST, PUT, and DELETE methods and to document the Microservices, we used the POSTMAN tool.
* Worked on deploying Micro Services using AWS EC2 and ECS instances.
* Utilized AWS Elastic Container Service (ECS) to containerize the payment microservices, allowing easy deployment, scaling, and management.
* Employed Docker to package the application and utilize Implemented AWS API Gateway to provide a secure and controlled access point for the payment services.
* Utilized Delta Lake to build robust and reliable data pipelines, enabling versioning, ACID transactions, and data quality maintenance.
* Designed and implemented a data lake architecture using AWS Lake Formation, enabling centralized storage and management of diverse data sources.
* Orchestrated data ingestion processes using AWS Glue ETL jobs, transforming and cataloging data for easy access and analysis
* Integrated AWS Simple Notification Service (SNS) and Simple Queue Service (SQS) to enable asynchronous communication between microservices. This pattern ensured loose coupling and fault tolerance, allowing the payment system to handle varying workloads efficiently.
* Developed ETL pipelines using Databricks notebooks, leveraging Apache Spark's capabilities for data transformation and analysis.
* Leveraged WebSocket technology to enable real-time payment updates for users. This feature allowed users to receive instant notifications about successful payments or any payment-related events.
* Provided technical support for Java-based applications running on a GKE cluster on GCP, ensuring smooth operation and resolving any issues that arose.
* Executed the implementation of new functionalities to enhance the Java application in the GCP environment, utilizing relevant tools and technologies.
* Implemented cloud-native security measures, including encryption, identity and access management (**IAM**), and network security groups, to protect the application and its data from unauthorized access.
* Designed and developed a microservices-based application using cloud-native principles, leveraging containerization with Docker and orchestration with Kubernetes.
* Created pipelines, used Jenkins to run tests, and used AWS (Amazon Web Services) Lambda to deploy applications.
* Collaborated with cross-functional teams to design, develop, and deliver high-quality serverless applications using AWS Lambda.
* Utilized AWS services, including AWS Kinesis, to implement real-time data streaming and processing capabilities within the payment billing software, ensuring timely and accurate transaction updates.
* Extensive experience in designing and deploying applications on AWS Lambda and EC2 instances.
* Utilized KSQL to query and analyse real-time streaming data, performing transformations and aggregations to derive valuable insights from the Kafka stream.
* Demonstrated proficiency in Java and extensive knowledge of Kafka API, enabling seamless integration of Kafka into Java-based applications and leveraging its advanced features.
* Implemented features such as push notifications, in-app purchases, and social media integration to enhance the functionality of mobile applications.
* Docker Container deployment, scaling, and administration were orchestrated using Kubernetes.
* Designed and developed containerized applications using Docker and Kubernetes, ensuring portability and scalability.
* Experience with log4j, including configuration, customization, and integration with various Java applications to facilitate logging and debugging.
* Demonstrated ability to build dashboards, alerts, and reports in Splunk for real-time monitoring and analysis.
* Used LogRhythm and Splunk for monitoring and troubleshooting production systems, including error tracking and performance optimization.
* Extensive experience in unit testing using JUnit and Jest, including the setup and maintenance of test cases, fixtures, and test suites.
* Developed and implemented a comprehensive TDD strategy for a payment software project, utilizing React and Java technologies.
* Created test plans, test cases, and test scripts based on business requirements and user stories, ensuring comprehensive coverage of all functionalities.
* Utilized JUnit and Jest frameworks to write unit tests for Java and React components, respectively, ensuring code quality and reliability
* Perform shallow and deep rendering of React components for testing purposes using Enzyme

**Environment:** Agile Methodology, **Java**1.8, J2EE, HTML5, CSS3, jQuery, Ajax, Bootstrap, JavaScript, Angular, **React,** **NodeJS**, ES6, VannilaJS, J2EE, Hibernate, Spring MVC, Spring IOC, Spring boot, Spring security, Python, OAuth, XML, GIT, JSP, JUnit, Maven, AWS, Apache AXIS, JAX-RS, Restful, Amazon Web Services (AWS), Micro Services, Log4J, Oracle, Docker, Mockito, JIRA, Jenkins and Spring Tool Suite, Postman, DynamoDB.

**Client: Philips Dec, 2017 – Dec 2020**

**Location: Hyderabad, India**

**Role: Sr Full Stack Developer**

**Responsibilities:**

* Used TypeScript to organize and deliver notifications in an Angular application.
* Participated in the application's requirements collection and analysis using the Agile development approach.
* Using **Angular**, HTML5, CSS3, and Bootstrap, designed and constructed a responsive web application that is dynamic and interoperable with multiple browsers.
* Developed responsive user interfaces using Material UI in Angular and React applications, ensuring a seamless user experience across different devices and screen sizes.
* Utilized Angular's component-based architecture to build scalable and modular UI components, facilitating code reuse and improving team collaboration during development.
* Integrated and customized Material UI components, such as buttons, cards, and modals, to match the design
* Developed Middle tier with Spring Boot and Angular for the user interface.
* Used maven build tool to configure the application.
* Used the **Angular CLI** extensively to create components, services, pipes, and directives.
* Utilized UML to model system architecture and database schema, including class diagrams, sequence diagrams, and ER diagrams.
* Developed single page applications using **Angular, Typescript, Web Pack, and Grunt.**
* Worked with stakeholders to design, develop, and deploy database solutions that aligned with project objectives.
* Ensured data integrity and security through the implementation of appropriate Oracle database security measures.
* Enhanced legacy application by adding new Typescript components.
* Created a Node.js server in the backend to handle requests coming from Angular Ajax calls in the front end.
* In a fast-paced setting, web applications and cross-platform runtime were created using Node.js, Angular and Bootstrap.
* Managed application state using RxJS and NgRx, enabling efficient state management and synchronization across components.
* Used Spring Boot to implement the application, and Spring Security was used to handle the authorization and authentication functions.
* Employed Tailwind CSS to streamline and expedite the front-end development process, maximizing productivity and ensuring consistent styling across the application.
* Demonstrated expertise in page optimization principles, implementing techniques to enhance website performance, loading speed, and search engine optimization (SEO) factors, resulting in improved user engagement and conversion rates.
* Created custom exceptions and error handling mechanisms.
* Designed and implemented OAuth 2.0 authentication and authorization mechanism for securing the system.
* Developed and executed automated tests using JUnit and Mockito to ensure the quality of the code.
* Used variety of Angular UI Bootstrap components, including calendars, masks, maps, sliders, alarms, and modal dialogs for UI development.
* Designed and implemented AWS Lambda functions using Python and Boto3 to automate data processing and ingestion tasks, reducing manual intervention and improving overall system efficiency.
* Leveraged Apache Spark to efficiently process and analyze large datasets, optimizing data pipelines and improving application performance. Employed Spark's distributed computing capabilities to handle complex transformations and aggregations.
* Implemented and optimized object-relational mapping (ORM) using Hibernate to facilitate seamless communication and interaction between Java objects and relational databases.
* Developed efficient Hibernate mappings, utilizing annotations or XML configurations, to establish a clear and effective relationship between database tables and Java entities.
* Designed and executed complex queries using Hibernate Query Language (HQL) to retrieve and manipulate data from relational databases, improving application performance and response time.
* **Implemented transaction management using Hibernate's built-in support for ACID (Atomicity, Consistency, Isolation, Durability) properties, ensuring data integrity and concurrency control in multi-user environments.**
* Created high-quality, scalable, and reusable components and front-end solutions using **Node.js framework.**
* **Developed RESTful APIs using Express.js for an Angular application.**
* Spring Boot was used to implement the application, and Spring Security was used to handle the authorization and authentication functions.
* **Developed and consumed web services using SOAP and Restful APIs, including WSDL and JSON formats.**
* Singleton, Factory, DAO, and Composite View Design Patterns were used to build the application.
* To develop REST controllers, map URLs for the request, and present with various views for various use cases, Spring MVC module was used. involved in creating web applications with Thyme leaf and Spring MVC.
* Collaborated with cross-functional teams to design and implement new features, utilizing Kotlin's concise syntax and functional programming features.
* Conducted code reviews and provided constructive feedback to team members, leading to an overall improvement in code quality and adherence to best practices.
* Implemented logging in the application using the Spring Framework AOP Module to track the status of the application.
* Used React Native's built-in components and APIs to create the user interface of your app, including screens for account management, transaction history, and other key features.
* Utilized Next.js's built-in data fetching methods, such as getStaticProps and getServerSideProps, to fetch and display dynamic data from external APIs or databases.
* Integrated MongoDB with a Node.js application, leveraging the MongoDB Node.js driver for seamless data operations and CRUD functionalities.
* Implemented data aggregation pipelines to perform complex data transformations and generate meaningful insights from large datasets stored in MongoDB
* Developed automated testing scripts using tools like Jest, Mocha, and Chai to ensure application and infrastructure reliability and quality.
* Developed automated test scripts using Cucumber and Selenium WebDriver to test web applications.
* Implemented Behaviour-Driven Development (BDD) practices and frameworks, with Cucumber to improve collaboration and communication between developers, testers, and stakeholders.
* Continuously optimized system performance, scalability, and cost-efficiency by monitoring and analyzing system metrics and implementing system enhancements and cost optimizations.
* Designed and implemented backup and disaster recovery strategies for Linux-based infrastructure, utilizing appropriate tools and techniques to safeguard critical data and minimize downtime.
* Resolved Linux server issues and performance bottlenecks by analyzing logs, monitoring system resources, and applying optimization techniques, resulting in improved system stability and uptime.
* Participated in all test cases using the Junit framework and corrected any faults or problems found throughout the testing phase.
* Designed and executed backup and recovery strategies in PostgreSQL, ensuring data integrity and minimizing downtime in case of system failures.
* Conducted database performance tuning in PostgreSQL by analyzing query execution plans, identifying bottlenecks, and optimizing SQL statements.
* Implemented data security measures in PostgreSQL, including user authentication, role-based access control, and encryption of sensitive data.
* Contributed to the creation of a Java API for Amazon Lambda to control several AWS services. utilized Elastic Search to search the document and AWS S3 to store and quickly retrieve the data.
* Worked on AWS Elastic load balancing for high availability application deployment. created EC2 instances and connected to Jenkins for integration using the public domain.
* Demonstrated hands-on development expertise in building applications and solutions using AWS services such as EC2, Lambda, EMR, and DynamoDB.
* Proficiently managed the end-to-end Continuous Integration and Continuous Deployment (CICD) pipeline for both AWS and on-premises applications
* Experience with setting up, configuring, and maintaining Jenkins CI/CD pipelines.
* Leveraged Kafka's scalability features to handle increasing data volumes and processing demands, ensuring efficient message processing and minimizing latency.
* Developed custom connectors using Kafka's connector framework, enabling seamless integration with external systems and facilitating data ingestion and extraction processes.
* Conducted performance monitoring and analysis using tools such as AppDynamics and Splunk, providing insights into system behaviour and identifying opportunities for optimization.
* Mentored and trained junior team members on automation and performance testing techniques and tools, improving the overall testing efficiency and quality of the team.
* Implemented end-to-end tests using Protractor for Angular applications to validate the entire software stack.
* Developed Protractor test scripts to simulate user interactions and validate the application's behavior.
* Expertise in integrating Jenkins with other tools such as Git, Maven, JIRA, and Selenium for end-to-end software delivery process.
* Experience with building and deploying applications using Jenkins on cloud platforms like AWS.

**Environment:** Java, Core Java, AWS, EC2, JUnit, Agile, Spring, JavaScript, Angular, React Native, Spring boot, Spring Security, Webservices, HTML, CSS, Oracle, VueJS, MongoDB, Maven, Bootstrap, JMS, Kafka, Jenkins, GIT, Nodejs, Log4j, RESTful.

**Client: Larsen & Tourbo Info Limited Mar 2016 – Nov 2017**

**Location: Hyderabad, India**

**Role: Software Developer**

**Responsibilities:**

* Front-end Angular development of a single-page, responsive banking web application.
* Used the Agile Methodology (Scrum) to produce high-quality products on schedule while meeting client expectations.
* Implemented a custom build process using Webpack to bundle and optimize the application's assets, reducing load times and improving overall performance.
* Integrated RESTful APIs with Angular services, enabling seamless communication between the frontend and backend systems.
* Designed and implemented a dynamic user interface using Angular components, directives, and templates, ensuring a consistent and responsive user experience across different devices.
* Utilized Angular Material UI components to create a visually appealing and user-friendly interface, following Material Design guidelines.
* Utilised AEM (DAM) system, to manage rich media assets, including images, videos, and audio files.
* Made use of Mockito to carry out testing and develop test cases.
* Used SonarLint to find and correct quality problems, and Maven for managing dependencies.
* The User Interface was created using HTML5, CSS3, JavaScript, **jQuery**, and Twitter Bootstrap, which improved the response.
* Developed and executed comprehensive end-to-end test scenarios using Jasmine, Karma, and Cypress, ensuring high-quality software delivery.
* Utilized Bootstrap's CSS and JavaScript libraries to create a consistent and modern look and feel across the application's UI

**Environment:** JAVA/J2EE, JAX-WS, EJB, Web Logic, JUnit, JMS, Oracle, Jenkins, Spring, Spring Boot, MongoDB, Angular, PHP, React Native, Agile Methodology, Log4j, SVN, Eclipse, Maven

**Education:**

**Master’s in Computer science**

Northern Illinois University