Vamsi Asa
Sr. Full Stack Developer
LinkedIn:www.linkedin.com/in/vamsiasa
Hackerrank**:**https://www.hackerrank.com/profile/Asa_Vamsi
Email: asavamsi22@gmail.com
Phone: +1 (205) 202-0316

SUMMARY:

- Highly accomplished Sr. Java Full Stack Developer with over 10+ years of experience designing, developing, and deploying complex enterprise applications. Proven ability to deliver across the entire development lifecycle, from front-end UI creation to back-end server architecture.
- Proven ability to mentor junior developers and guide teams toward successful project completion.
- Proficient in Java development with expertise in Spring Framework (Core, Batch, Boot, Transaction, Security), Hibernate, Struts, JSP, Servlets, and EJB, ensuring robust and scalable backend solutions.
- Experienced in application server technologies such as Apache Tomcat, JBoss, and WebSphere, ensuring smooth deployment and management of Java applications.
- Skilled in message brokers such as RabbitMQ, Kafka, and GraphQL, and developing and consuming RESTful APIs, SOAP services, and JMS for seamless communication between components.
- Familiar with authentication and authorization protocols like OAuth2 and OpenID Connect, ensuring secure access to applications and APIs.
- Experienced in working with various databases, including MySQL, PostgreSQL, SQL Server, Oracle, MongoDB, Redis, DynamoDB, and CosmosDB, managing data effectively for applications.
- Experienced in working with data formats like JSON, XML, XSLT, XSD, and Avro, ensuring efficient data exchange and processing.
- Proficient in frontend technologies including HTML, CSS, JavaScript, jQuery, AJAX, React, Redux, Angular, Vue.js, Material UI, Bootstrap, Express.js, Node.js, RxJS, TypeScript, and Tailwind CSS, delivering engaging and user-friendly interfaces.
- Skilled in cloud platforms such as AWS (EC2, S3, Lambda, RDS, EBS, Auto Scaling, API Gateway, ELB, CloudFormation, CloudWatch, CodePipeline, CodeBuild, CodeDeploy, CloudFront), Microsoft Azure (Azure App Service, Azure Functions, Azure Logic Apps, ADF, Azure AD, Azure DevOps), and GCP (Google App Engine, Cloud Functions, Cloud Pub/Sub), ensuring seamless deployment and scalability of applications.
- Knowledgeable in containerization technologies like Docker, Kubernetes, AWS ECS, and AWS EKS, ensuring efficient deployment and management of microservices-based applications.
- Familiar with infrastructure-as-code tools like CloudFormation, Ansible, and Terraform, automating infrastructure provisioning and management.
- Proficient in building automation tools like Maven, Gradle, ANT, and continuous integration/continuous deployment (CI/CD) pipelines using Jenkins, GitHub Actions, AWS CodePipeline, and Azure DevOps, ensuring efficient and automated software delivery.
- Experienced in logging and monitoring tools such as Log4J, CloudWatch, Splunk, New Relic, and ELK Stack, ensuring high availability and performance of applications.
- Skilled in reporting tools like BIRT, Jasper Reports, Power BI, and collaboration platforms like JIRA, Confluence, ServiceNow, and SharePoint, facilitating effective project management and reporting.
- Proficient in testing frameworks and tools, including JUnit, Mockito, Cucumber, Postman, and Swagger, ensuring the quality and reliability of applications.
- Proficient in utilizing version control methods such as BitBucket, Git, and GitHub and IDEs like Eclipse IDE, IntelliJ IDEA, and NetBeans, facilitating efficient development workflows and collaboration.
- Possess strong written and verbal communication abilities, adept at articulating technical concepts clearly to diverse audiences, including technical and non-technical stakeholders.

TECHNICAL SKILLS

Java/ J2EE & Frameworks: Spring Framework (Core, Batch, Boot, Transaction, Security), Hibernate, Struts, JSP, Servlets, EJB
Web Technologies: HTML5, CSS3, JavaScript (ES6+) jQuery, AJAX, React, Redux, Angular, Vue.js, Material UI, Bootstrap, Express.js, Node.js, RxJS, TypeScript, Tailwind CSS
Data Transformation: JSON, XML, XSLT, XSD, Avro
Messaging and Communication: RabbitMQ, Kafka, flink, IBM MQ,GraphQL, RESTful APIs, SOAP, JMS
Database Technologies: MySQL, PostgreSQL, SQL Server, Oracle, MongoDB, Redis, Cassandra,DynamoDB, Hibernate, CosmosDB,pl/sql triggers
Cloud Platforms and Services: AWS (EC2, S3, Lambda, RDS, EBS, Auto Scaling, API Gateway, ELB, CloudFormation, CloudWatch, CodePipeline, CodeBuild, CodeDeploy, CloudFront,Glue), Microsoft Azure (Azure App Service, Azure Functions, Azure Logic Apps, ADF, Azure AD, Azure DevOps), GCP (Google App Engine, Cloud Functions, Cloud Pub/Sub)
Containerization: Docker, Kubernetes, AWS ECS, AWS EKS
Build Tools & CI/CD: Maven, Gradle, ANT, Jenkins, SonarQube,GitHub Actions, AWS CodePipeline, Azure DevOps
Logging and Monitoring: Log4J, CloudWatch, Splunk, New Relic, ELK Stack

IaC & Automation: CloudFormation, Ansible, Terraform
Testing Frameworks and Tools: JUnit, Mockito, Cucumber, Postman, Swagger
Security: OAuth2, OpenID Connect,
Servers: Apache Tomcat, JBoss, WebSphere
IDEs & Version Control: Eclipse IDE, IntelliJ IDEA, NetBeans, Git, GitHub, BitBucket,npm, bower, grunt, gulp,Webpack
Monitoring & Logging: Splunk, New Relic, ELK Stack
Reporting & Collaboration: BIRT, Jasper Reports, Power BI, JIRA, Confluence, ServiceNow, SharePoint, Agile, Scrum, Kanban
Operating Systems: Windows, Linux, UNIX, macOS

PROFESSIONAL EXPERIENCE:

Client: Collins Aerospace, Binghamton, New York **Nov 2022 – Present**
Role: Sr. Full Stack Developer
Responsibilities:

- Led the software development life cycle (**SDLC**) from conception to deployment, ensuring high-quality deliverables and timely project completion.
- Practiced **Agile** and **Scrum** methodologies for iterative development, collaboration, and continuous improvement of project processes.
- Provided training sessions and ongoing technical support to engineers and administrative staff, enhancing their proficiency and confidence in using **aerospace systems** and **software**.
- Tracked project progress, tasks, and issues using **JIRA**, ensuring transparency, accountability, and alignment with project goals.
- Utilized **Spring Core** and **Spring Batch** frameworks to develop scalable and efficient backend data processing and management processes.
- Implemented **Hibernate ORM** for seamless database interaction, optimizing data retrieval and storage operations through efficient query handling and entity mapping.
- Developed and maintained scalable backend applications using **Java** with **Spring Boot**, ensuring high performance and reliability by leveraging **Spring Boot**'s auto-configuration and dependency management features.
- Implemented and optimized data access layers using **Spring Boot** with **JDBC** and **Hibernate**, improving database interaction efficiency by designing and refining **DAOs** and repositories.
- Integrated authentication and authorization mechanisms using **Spring Boot Security**, ensuring secure access to backend services and APIs with **OAuth2** and **JWT**.
- **Developed RESTful APIs** using **Spring Boot** and integrated them with **AWS API Gateway**, providing secure and scalable access to microservices, and ensuring smooth traffic management with features like throttling and caching.
- Designed and implemented **RESTful APIs** and web services using **Spring Boot**, facilitating seamless integration with frontend and third-party systems through consistent and well-documented endpoints.
- Gained expertise in **NoSQL databases** like **DynamoDB and MongoDB**, implementing flexible and high-performance storage solutions.
- Implemented event-driven architecture using **Kafka** messaging system for real-time data processing and communication between microservices, ensuring asynchronous processing and improved system resilience.
- Implemented **GraphQL** for efficient data querying and manipulation, improving overall application performance and providing clients with flexible data retrieval capabilities.
- Utilized **Java Streams API** for streamlined data processing and manipulation, optimizing performance and resource utilization through functional programming techniques.
- Developed responsive and user-friendly web interfaces using **HTML5**, **CSS3**, and **JavaScript ES6+**, enhancing user experience and accessibility with modern web standards.
- Implemented **Bootstrap** framework to ensure consistent and responsive UI across different devices and screen sizes, utilizing grid systems and pre-built components.
- Integrated **Express.js** framework for building robust and scalable backend APIs, ensuring efficient data processing and communication with frontend applications.
- Developed frontend components using **React**, **React Router**, and **Redux** for state management and dynamic UI rendering, enhancing user interaction and performance.
- Developed dynamic and responsive web applications using **Angular** and **TypeScript**, implementing state management with **NgRx** and ensuring type safety and modular code.
- Implemented caching techniques with frameworks like **EhCache** or **Redis** within **Spring Boot** to optimize application performance.
- Utilized **Docker** for containerization of microservices, enabling consistent and portable deployment across environments.
- Employed **Kubernetes** for orchestration and management of microservices, ensuring high availability and scalability.
- Utilized **Behavior-Driven Development (BDD)**: Employed **BDD** practices to ensure software features align with business requirements.
- Developed Executable Specifications: Used tools like **Cucumber** and **SpecFlow** to write and maintain **Gherkin** syntax specifications for clear, understandable test scenarios.

- **Container Orchestration and Microservices Deployment:** Deployed Java-based microservices on OpenShift, leveraging its Kubernetes-based orchestration capabilities to ensure high availability, scalability, and robust management of containerized applications.
- **Behavior-Driven Development (BDD):** Applied BDD practices to ensure that software features align with business requirements. Created executable specifications using tools like Cucumber and SpecFlow to write and maintain clear, understandable test scenarios.
- **Azure Cloud Deployment:** Deployed applications on Azure cloud infrastructure, utilizing Azure Virtual Machines, Azure Blob Storage, Azure Functions, and Azure SQL Database to ensure reliable and scalable cloud solutions.
- **Elastic Block Storage (EBS) Management:** Managed storage using Azure Disk Storage to ensure reliable and persistent storage for applications and databases, contributing to system stability and data integrity.
- **Autoscaling and Performance Monitoring on Azure:** Implemented autoscaling strategies on Azure to adjust resources dynamically based on application demand. Monitored and managed application performance using Azure Monitor, optimizing resource utilization and identifying performance bottlenecks.

Environment: Spring Core, Spring Batch, Hibernate, Azure Virtual Machines, Azure Blob Storage, Azure Functions, Azure SQL Database, Azure Monitor, Auto Scaling, API Gateway, Load Balancer, Kafka, Spring Boot, React, TypeScript, Bootstrap, Express.js, MySQL, PostgreSQL, MongoDB,Redis, EhCache, Azure CloudFormation, Docker, Kubernetes, Jenkins, Maven, Ansible, Git, Azure DevOps, Eclipse, Apache Tomcat, Spring Security, OAuth2, JWT, SAML, OpenID Connect, Identity Management, MFA, Zero-Trust Architecture, Unix/Linux, Shell Script, PowerShell, Perl Script, JUnit, Cucumber, Swagger, Agile, Scrum, JIRA..

Client: Ross Stores, Bay Area, CA. **June 2020 – Oct 2022**
Role: Sr. Full Stack Developer
Responsibilities:

- Managed project dependencies and build configurations using **Gradle**.
- Implemented infrastructure as code using **Terraform** for cloud resource provisioning and management.
- Utilized **Git** and **GitHub** for efficient version control and collaborative software development, ensuring code integrity and facilitating seamless teamwork.
- Utilized **Spring Framework** for building robust and scalable applications, including **Inversion of Control (IoC)** and **Dependency Injection (DI)** for modular and maintainable code.
- Implemented **Core Java** principles and best practices in application development for efficient and optimized codebase.
- Developed and maintained **microservices architecture** using **Spring Boot**, facilitating modular and scalable application development.
- Applied **Aspect-Oriented Programming (AOP)** concepts to modularize cross-cutting concerns such as logging and security.
- Integrated **Hibernate ORM** and **Data Access Objects (DAO)** for seamless database interactions and data persistence.
- **Real-Time Data Analytics**: Leveraged NoSQL databases such as Cassandra for real-time analytics, optimizing queries and data storage for low-latency insights.
- Developed scalable and maintainable applications using **TypeScript**, ensuring type safety and reducing runtime errors.
- Implemented **object-oriented programming** principles in **TypeScript** to create reusable and modular code.
- Developed and debugged applications using **IntelliJ IDEA** for efficient coding and debugging.
- Designed and implemented efficient **inventory management systems** for Ross Stores, ensuring accurate tracking of stock levels and compliance with retail industry standards.
- Developed dynamic and responsive web applications using **Angular** and **TypeScript**.
- Implemented frontend components using **TypeScript**, **AJAX**, **Node.js**, **Angular**, and **RxJS** for interactive user experiences.
- Implemented **state management** in **Angular** applications using **NgRx** for complex data handling.
- Integrated **RESTful APIs** and **GraphQL** endpoints with **Angular** for efficient data retrieval and manipulation.
- Utilized **Angular Router** for declarative routing and navigation within single-page applications.
- Created reusable components, directives, and services using **Angular's component-based architecture**.
- Implemented **reactive forms** and **template-driven forms** with **Angular** for efficient form handling and validation.
- Developed responsive and interactive user interfaces using **React.js**.
- Implemented **state management** using **React's** built-in state and **Redux** for complex applications.
- Integrated **RESTful APIs** and **GraphQL** endpoints with **React** for data fetching and management.
- Utilized **React Router** for declarative routing and navigation within single-page applications.
- Developed reusable UI components and libraries using **React's component-based architecture**.
- Implemented **responsive web design** principles using **CSS-in-JS** solutions like **Styled Components** or **Emotion** with **React**.
- Implemented **multithreading** techniques for concurrent processing and improved application performance.
- Utilized **caching techniques** with frameworks like **EhCache** or **Redis** within **Spring Boot** to optimize application performance.
- Implemented **messaging systems** using **Spring Boot** and integrated with message brokers like **RabbitMQ** or **Kafka** for asynchronous communication.

- Implemented **transaction management** and **error handling strategies** with **Spring Boot**, ensuring data integrity and application stability.
- Implemented **scheduled jobs** and **batch processing** using **Spring Boot's scheduling capabilities**, automating routine tasks and improving operational efficiency.
- Utilized **Docker** for containerization of microservices, enabling consistent and portable deployment across environments.
- Employed **Kubernetes** for orchestration and management of microservices, ensuring high availability and scalability.
- **Application Scaling and Monitoring:** Configured horizontal and vertical scaling on OpenShift, enabling the application to respond to workload changes efficiently. Monitored performance metrics and resource usage to proactively manage scalability and optimize application response times.
- **CI/CD Integration:** Integrated OpenShift with CI/CD pipelines, allowing automated deployment of code changes. This setup enabled iterative development and ensured the rapid, reliable release of updates.
- Automated **build**, **test**, and **deployment processes** using **Jenkins** for continuous integration and delivery.
- Deployed applications on **Azure cloud infrastructure**, including **Azure Virtual Machines**, **Azure Blob Storage** for storage, **Azure Functions** for serverless computing, and **Azure SQL Database** for **database management**.
- Implemented **autoscaling strategies** on **Azure** to adjust resources based on dynamic application demand.
- Utilized **Apache Kafka** for efficient **event-driven messaging** and **data processing**.
- Worked with **relational databases** such as **PostgreSQL** and **NoSQL databases** like **MongoDB** for **data storage** and **retrieval**.
- Implemented **OAuth2** for secure **authentication** and **authorization mechanisms**.
- Managed **XML configurations** and **data transformations** for application integration.
- Deployed applications on the **JBoss application server** for production environments.
- Utilized **ELK stack** (**Elasticsearch**, **Logstash**, **Kibana**) for **log monitoring** and **analysis**.
- Generated **reports** and **data visualizations** using **Jasper Reports** for **business intelligence**.
- Developed and debugged applications using **IntelliJ IDEA** for efficient **coding** and **debugging**.

Environment: Spring, IoC, DI, Core Java, Hibernate, RESTful APIs, Spring Boot, Spring Transaction, Azure, Azure Virtual Machines, Azure Blob Storage, Azure Functions, Azure SQL Database, Apache Kafka, PostgreSQL, MongoDB, TypeScript, AJAX, Node.js, Angular, RxJS, HTML5, CSS3, OAuth2, JBoss, XML, ELK, Docker, Kubernetes, Terraform, Jenkins, Jasper Reports, Gradle, GitHub, IntelliJ.

Client: State of California, Department of Public Health ,Los Angeles **Sep 2018 – June 2020**
Role: Full Stack Java Developer
Responsibilities:

- Practiced **Agile methodologies**, including **Kanban**, for iterative and collaborative project management.
- Implemented **Spring framework** components, including **Inversion of Control (IoC)**, **Dependency Injection (DI)**, and **Aspect-Oriented Programming (AOP)** for a modular and maintainable codebase.
- Utilized **Lambda expressions** for concise and functional programming paradigms, improving code readability and maintainability.
- Implemented robust **Exception Handling** mechanisms to ensure application stability and graceful error recovery.
- Designed and optimized data access layer using **Hibernate ORM** for seamless integration with **SQL Server** and **CosmosDB**.
- Developed **RESTful APIs** using **Spring Boot** and **Microservices architecture** to enhance system scalability and performance.
- Implemented authentication and authorization across microservices using **OAuth2**, **JWT**, and **API Gateway** solutions.
- Utilized **API Gateway** and service mesh technologies like **Istio** and **Envoy** for traffic management, security, and observability of microservices.
- Conducted unit testing, integration testing, and end-to-end testing of microservices using frameworks like **JUnit**, **Mockito**, **Mocha**, and **Jest**.
- Integrated with external identity providers (IdPs) and **LDAP directories** using **Spring Security**.
- Implemented custom authentication providers and user details services to meet specific application requirements with **Spring Security**.
- Configured secure method-level access control using **Spring Security** annotations.
- Implemented security headers and **CORS policies** using **Spring Security** to mitigate security risks.
- Implemented messaging services using **JMS** for reliable communication between distributed components.
- Utilized **Java Streams API** for streamlined data processing and manipulation, optimizing performance and resource utilization through functional programming techniques.
- Managed code repositories and version control using **BitBucket**, ensuring code integrity and collaboration.
- Developed and maintained scalable backend applications using **Java** with **Spring Boot**, ensuring high performance and reliability by leveraging **Spring Boot**'s auto-configuration and dependency management features.
- Implemented and optimized data access layers using **Spring Boot** with **JDBC** and **Hibernate**, improving database interaction efficiency by designing and refining DAOs and repositories.

- Integrated authentication and authorization mechanisms using **Spring Boot Security**, ensuring secure access to backend services and APIs with **OAuth2** and **JWT**.
- Implemented event-driven architecture using **Kafka** messaging system for real-time data processing and communication between microservices, ensuring asynchronous processing and improved system resilience.
- Implemented **GraphQL** for efficient data querying and manipulation, improving overall application performance and providing clients with flexible data retrieval capabilities.
- Experienced in working with various **databases**, including **MySQL**, **PostgreSQL**, **SQL Server**, **Oracle**, **MongoDB**, **Redis**, **DynamoDB**, and **CosmosDB**, managing data effectively for applications.
- Serialized and deserialized data using **JSON** for efficient data exchange between services and clients.
- Developed responsive and user-friendly web interfaces using **HTML5**, **CSS3**, and **JavaScript** frameworks like **Vue.js** and **Material UI**.
- Utilized **TypeScript** with modern **JavaScript** features (ES6+) to build efficient and powerful applications.
- Implemented asynchronous programming with **TypeScript** using **Promises**, **async/await**, and **Observables**.
- Conducted unit testing and integration testing of **TypeScript** code using testing frameworks like **Jest**, **Mocha**, and **Jasmine**.
- Developed and maintained **TypeScript type definitions** for third-party libraries to ensure seamless integration.
- Conducted unit testing and end-to-end testing using **Jest** and **React Testing Library** to ensure robust code quality.
- Collaborated closely with UX/UI designers to translate wireframes and prototypes into high-quality **React components**.
- Optimized application performance by implementing code splitting and lazy loading techniques in **React**.
- Integrated authentication and authorization features using **React Context API** or third-party libraries like **Auth0**.
- Implemented internationalization (i18n) and localization (l10n) features using **React**'s support for internationalization libraries.
- Integrated asynchronous data retrieval and updates using **AJAX** and **Node.js** for real-time user interactions.
- Utilized **Docker** for containerization of microservices, enabling consistent and portable deployment across environments.
- Employed **Kubernetes** for orchestration and management of microservices, ensuring high availability and scalability.
- Managed cloud-based solutions on **Microsoft Azure**, including **Azure App Service**, **Azure Functions**, **Azure Logic Apps**, and **Azure Data Factory (ADF)**, ensuring scalability and reliability.
- Implemented **Azure AD** for secure authentication and authorization mechanisms in enterprise applications.
- Implemented **CI/CD pipelines** using **Azure DevOps** to automate build, test, and deployment processes.
- Deployed and managed applications on **WebSphere** and **OpenShift platforms** for high availability and scalability.
- Utilized **Splunk** for log monitoring and analysis, ensuring system performance and reliability.
- Created interactive data visualizations and reports using **Power BI** for data-driven decision-making.
- Automated build and deployment processes using **Maven**, improving development productivity and consistency.
- Managed cloud-based solutions on **Microsoft Azure**, including **Azure App Service**, **Azure Functions**, **Azure Logic Apps**, and **Azure Data Factory (ADF)**, ensuring scalability and reliability.

Environment: RESTful APIs, Spring Boot, Vue.js, Material UI, AJAX, Node.js, SQL Server, CosmosDB, Microsoft Azure, ADF, Azure AD, Azure DevOps, JSON, BitBucket, JMS, WebSphere, Splunk, Power BI, NetBeans, JUnit, JIRA, OpenShift, Maven, Agile, Kanban.

Client: BMO USA **Oct 2017 – Sep 2018**
Java developer

- Implemented **J2EE technologies** to develop scalable and robust enterprise applications, leveraging **Spring framework** for **dependency injection** and **MVC architecture**.
- Integrated **RabbitMQ** for asynchronous message queuing, ensuring efficient communication and decoupling of components.
- Developed dynamic web pages using **JSP** and **Servlets**, facilitating user interaction and data processing.
- Utilized **Spring MVC** and **Spring Data JPA** to build **RESTful APIs** and interact with **Oracle** and **Redis databases**, ensuring data persistence and efficient CRUD operations.
- Implemented logging and monitoring using **Spring Boot**'s integration with logging frameworks like **Logback** or **SLF4J**, and monitoring tools like **Prometheus** or **Micrometer**.
- Contributed to the development of security guidelines and best practices within the organization with **Spring Security**.
- Continued professional development through training and certifications in **cybersecurity** and **Spring Security**.
- Actively participated in security communities and forums to stay updated on emerging threats and security trends with **Spring Security**.
- Collaborated with cross-functional teams including **UX/UI designers**, **product managers**, and **QA engineers** to deliver high-quality software solutions using **Spring Boot**.
- Followed **agile development practices** including daily stand-ups, sprint planning, and retrospective meetings to ensure timely delivery of features and continuous improvement with **Spring Boot**.

- Designed and implemented responsive user interfaces with **HTML**, **CSS**, **JavaScript**, **jQuery**, and **AJAX**, enhancing user experience and interactivity.
- Deployed **Angular** applications to cloud platforms like **AWS**, **Azure**, or **Firebase** using **CI/CD pipelines** (e.g., **Jenkins**, **GitLab CI**).
- Implemented real-time updates and data synchronization using **Angular** in conjunction with **WebSocket** technologies (e.g., **Socket.io**).
- Collaborated in **Agile** development environments, participating in daily stand-ups, sprint planning, and retrospective meetings for **Angular projects**.
- Implemented **Avro serialization** for efficient data exchange and storage, optimizing performance and resource utilization.
- Deployed and managed applications on **Google Cloud Platform (GCP)**, utilizing **Google App Engine** for scalable and reliable hosting.
- Proficiency in designing and managing scalable infrastructure using **GCP services** like **Compute Engine** and **Kubernetes Engine**.
- Experience with **GCP**'s data analytics and machine learning tools, such as **BigQuery** and **AI Platform**, for processing large datasets and building intelligent solutions.
- Expertise in implementing robust security measures on **GCP**, including **IAM**, **encryption**, and compliance with industry standards, ensuring data protection and regulatory adherence.
- Developed **serverless functions** using **Cloud Functions**, enabling event-driven and cost-effective application architecture.
- Integrated **Google Cloud Pub/Sub** for reliable messaging and event-driven communication between microservices.
- Utilized **New Relic** for application monitoring and performance optimization, ensuring high availability and responsiveness.
- Conducted unit and integration testing using **Mockito**, ensuring code quality and reliability.
- Collaborated on project documentation and knowledge sharing using **Confluence**, facilitating team collaboration and information sharing.
- Automated build and deployment processes using **ANT** and **GitHub Actions**, streamlining development workflows and ensuring code consistency.
- Integrated **OpenID Connect** for authentication and authorization, enhancing application security and user privacy.
- Developed and debugged applications using **Eclipse IDE**, ensuring efficient coding practices and software quality.
- Managed version control and collaborative development using **Git**, promoting code quality and teamwork.

Environment: Angular,J2EE, Spring, JSP, Servlets, Spring MVC, Spring Data JPA, RabbitMQ, RESTful APIs, Oracle, Redis, AJAX, Avro, GCP, New Relic, Mockito, Confluence, ANT, OpenID Connect, Eclipse IDE, Git.

Client: Cybage Software, Pune, India. **May 2013 - July 2016**
Java Developer

- Developed web applications using **Java**, **Struts**, **JSP**, and **Servlets** to enhance user experience and streamline operations.
- Followed **agile development practices** including daily stand-ups, sprint planning, and retrospective meetings to ensure timely delivery of features and continuous improvement.
- Implemented **J2EE technologies** to develop scalable and robust enterprise applications, leveraging **Spring framework** for **dependency injection** and **MVC architecture**.
- Implemented robust **Exception Handling** mechanisms to ensure application stability and graceful error recovery.
- Designed and optimized **data access layer** using **Hibernate ORM** for seamless integration with **SQL Server** and **CosmosDB**.
- Utilized **Spring MVC** and **Spring Data JPA** to build **RESTful APIs** and interact with **Oracle** and **Redis databases**, ensuring data persistence and efficient CRUD operations.
- Developed **RESTful APIs** using **Spring Boot** and **Microservices architecture** to enhance system scalability and performance.
- Developed and maintained scalable backend applications using **Java** with **Spring Boot**, ensuring high performance and reliability by leveraging Spring Boot's auto-configuration and dependency management features.
- Implemented and optimized **data access layers** using **Spring Boot** with **JDBC** and **Hibernate**, improving database interaction efficiency by designing and refining **DAOs** and **repositories**.
- Integrated authentication and authorization mechanisms using **Spring Boot Security**, ensuring secure access to backend services and APIs with **OAuth2** and **JWT**.
- Implemented **event-driven architecture** using **Kafka** messaging system for real-time data processing and communication between microservices, ensuring asynchronous processing and improved system resilience.
- Integrated **RabbitMQ** for asynchronous message queuing, ensuring efficient communication and decoupling of components.
- Implemented **GraphQL** for efficient data querying and manipulation, improving overall application performance and providing clients with flexible data retrieval capabilities.
- Utilized **Java Streams API** for streamlined data processing and manipulation, optimizing performance and resource utilization through functional programming techniques.
- Experienced in working with various databases, including **MySQL**, **PostgreSQL**, **SQL Server**, **Oracle**, **MongoDB**, **Redis**, **DynamoDB**, and **CosmosDB**, managing data effectively for applications.
- Serialized and deserialized data using **JSON** for efficient data exchange between services and clients.
- Developed dynamic web pages using **JSP** and **Servlets**, facilitating user interaction and data processing.
- Designed and implemented responsive user interfaces with **HTML**, **CSS**, **JavaScript**, **jQuery**, and **AJAX**, enhancing user experience and interactivity.
- Created responsive and interactive user interfaces using **HTML5**, **CSS3**, and **JavaScript frameworks** like **Vue.js** and **Material UI**.

- Utilized **TypeScript** with modern **JavaScript** features (**ES6+**) to build efficient and powerful applications.
- Developed frontend components using **React**, **React Router**, and **Redux** for state management and dynamic UI rendering, enhancing user interaction and performance.
- Implemented internationalization (**i18n**) and localization (**l10n**) features using **React**'s support for internationalization libraries.
- Deployed **Angular** applications to cloud platforms like **AWS**, **Azure**, or **Firebase** using **CI/CD pipelines** (e.g., **Jenkins**, **GitLab CI**).
- Implemented real-time updates and data synchronization using **Angular** in conjunction with **WebSocket** technologies (e.g., **Socket.io**).
- Collaborated closely with **UX/UI designers** to translate wireframes and prototypes into high-quality **Angular components**.
- Implemented **Avro serialization** for efficient data exchange and storage, optimizing performance and resource utilization.
- Utilized **Docker** for containerization of microservices, enabling consistent and portable deployment across environments.
- Employed **Kubernetes** for orchestration and management of microservices, ensuring high availability and scalability.
- Managed cloud-based solutions on **Microsoft Azure**, including **Azure App Service**, **Azure Functions**, **Azure Logic Apps**, and **Azure Data Factory (ADF)**, ensuring scalability and reliability.
- Implemented **Azure AD** for secure authentication and authorization mechanisms in enterprise applications.
- Implemented **CI/CD pipelines** using **Azure DevOps** to automate build, test, and deployment processes.
- Deployed and managed applications on **Google Cloud Platform (GCP)**, utilizing **Google App Engine** for scalable and reliable hosting.
- Implemented **AWS Lambda** for serverless computing, reducing infrastructure costs and improving application scalability.
- Managed scalable applications on **AWS EC2**, leveraging **AWS S3** for storage and **AWS RDS** for relational database services.
- Integrated **Google Cloud Pub/Sub** for reliable messaging and event-driven communication between microservices.
- Integrated **ServiceNow** for IT service management, enhancing incident tracking and resolution processes.
- Utilized **New Relic** for application monitoring and performance optimization, ensuring high availability and responsiveness.
- Utilized **Splunk** for log monitoring and analysis, ensuring system performance and reliability.
- Automated build and deployment processes using **ANT** and **GitHub Actions**, streamlining development workflows and ensuring code consistency.

Environment: Java, angular,Struts, JSP, Servlets, Hibernate, EJB, MongoDB,DB2, SOAP, WSDL, AWS, Log4J, GIT, Eclipse, ServiceNow, Sharepoint, Mockito, aws, ActiveMQ, Maven.

Education:

- B-Tech, Computer Science from Vasireddy Venkatadri Institute of Technology, May 2013
- Masters, Computer Science from University of Alabama at Birmingham, Aug 2016 - Dec 2017