**ABHISHEK**

***E-mail****:* *naiduabhishek1091@gmail.com*

**PROFESSIONAL SUMMARY:**

* Around 10+ years of experience working as a Java/J2EE developer in designing, developing, deploying, and maintaining web-based, user-centric, client/server applications along with Enterprise and Distributed applications.
* Expertise in Java programming and have good understanding on OOPs, Collections, Exceptions Handling, Input/output (I/O) System, Annotations, Concurrency, Multi-Threading, Lambda and Generics.
* Expert level experience with Web based Application design & development using Java, J2EE, Spring Boot, Spring, Hibernate, Oracle, SQL.
* Designed and implemented RESTful APIs using Java 8 features such as Lambda expressions and Streams for cleaner, more efficient code.
* Implemented enterprise-level microservices architecture using Java 11, improving performance, and reducing response times.
* Utilized Java 17’s improved garbage collection (G1, ZGC) for optimizing memory management in high-performance applications.
* Employed enhanced JVM features in Java 17 to improve application performance and reduce latency in microservices architecture.
* Proficient in developing front-end systems with JavaScript, JQuery, Ajax, Bootstrap, HTML5, CSS3 and JavaScript frameworks such as Angular and React JS library.
* Experience on Object Oriented Design Patterns such as Singleton Pattern, MVC Pattern, Factory Pattern, Builder Pattern and DAO Pattern with exposure to UML Modeling.
* Good experience working with the Java1.8 and JDK1.8 and implemented the new feature like Lambda Expressions, Functional Interface, Streams, Java Time API.
* Well Experienced with Micro services in implementing client-server business applications to build independent task and flexible applications.
* Refactored existing codebases with Java 8's functional programming constructs, reducing code complexity by 40%.
* Designed and implemented MongoDB schemas for scalable and flexible data storage in microservices.
* Leveraged Java 11's new API enhancements and Stream API to simplify code and increase maintainability.
* Experience in working on various Spring modules (Spring Core, Spring MVC, Spring Boot, DAO and AOP), along with Hibernate as the back end ORM tool for implementation of persistence layer, Microservices using Spring and mapping of POJOs.
* Worked with various python libraries to create and manipulate excel reports. software development and design, testing of web based and client server multi-tier applications using Python, Django & Backend Technologies.
* Expertise in Object Oriented concepts like Creating Classes, Inheritance, Constructors, Polymorphism, Encapsulation, Overloading and Overriding.
* Migrated legacy systems from Java 8 to Java 11, optimizing resource management and reducing memory leaks.
* Experience as a Python Developer in design, Development, testing, maintenance and documentation of Enterprise applications using various technologies, analyzing complex business requirements & mapping them to system specifications.
* Knowledge of XML-based standards and protocols such as WSDL and SOAP, Service Oriented Architecture SOA, development of XSD schemas and implementing both SOAP and REST web services.
* Extensively worked on Micro Services using Spring Boot, Spring Data and REST
* Hands on experience in Amazon Web Services (AWS), services like EC2, S3, IAM, ELB (Load Balancers), RDS, VPC, Route53, Cloud Watch, Cloud Formation, IAM, SNS, SQS, Lambda, API gateways, Security groups.
* Experience with AWS Cloud Formation Scripts and launch configurations to automate repeatable provisioning of AWS resources for applications.
* Deep hands-on experience on building scalable web service architectures with various relational MySQL, OracleDB.
* Experience in working with NoSQL databases like Cassandra, MongoDB and ElasticSearch.
* Experience with messaging systems like Kafka, Kafka event sourcing, ActiveMQ and RabbitMQ.
* Extensive experience in working with Unit testing frameworks Junit and integration testing with Mockito.
* Good experience working with Continuous Integration and Deployment (CI/CD) tools like Jenkins, Docker.
* Worked on CI/CD tool Jenkins to automate the build process from version control tool into testing and production environment.
* Used Kubernetes to orchestrate the deployment, scaling, and management of Docker containers.
* Experience in using version control and configuration management tools like GIT, GitHub and CVS.
* Expertise in using Agile Software Development methodologies like Scrum, Sprint and Traditional software models like Waterfall and Test-Driven Development (TDD).
* Strong analytical, interpersonal and communications skills. Demonstrated track record of analyzing situations, implementing in a fast-paced environment.

**TECHNICAL SKILLS:**

|  |  |
| --- | --- |
| **Languages** | Java 1.8, Java 11, Java 17, J2EE, C, C++, PL/SQL, Python 3.6.0, Drools, SQL, Kotlin, |
| **Java/J2EE Technologies** | Servlets, JSP, JSTL, JDBC, JSF, JavaBeans, MVC, Log4j, XML, JAXP, DOM, JNDI, JUNIT, Web services, Multithreading, JTA, |
| **Web Technologies** | HTML, JavaScript (ES5, ES6), PHP, Node.js, Angular.js, Angular 4/5/6/8/9/11/13/15/17, CSS, AJAX, DOJO, XML, Web Services (SOAP, REST, WSDL) jQuery, |
| **Frameworks and IDE’s** | Struts, Spring version 4, 5, Hibernate, JPA, Spring Core, Spring ORM, Spring MVC, Spring AOP, Spring Boot, Maven, Gradle, Jenkins, SonarQube |
| **Web/Application Servers** | Web logic, Apache, WebSphere 8.x/9.x, IBM, Jetty, JBOSS, Tomcat 8 |
| **Automation Testing** | Selenium, Cucumber TestNG, JUnit5, Jest, Mocha |
| **Web Service Technologies** | SOAP, REST, WSDL, SAX/DOM parsers, Jersey, XSD, JAX -WS, JAX -RPC, |
| **Version Control** | GIT, Tortoise SVN, CVS, SVN, VSS, GIT |
| **IDE’s** | Eclipse, NetBeans, IntelliJ. |
| **Patterns** | MVC, Core J2EE design patterns (MVC, business delegate, service locator, session façade, DAO, DTO, Singleton, etc.) |
| **Methodologies** | Agile, Iterative, SCRUM, Kanban |
| **Database Environments** | SQL Server, MySQL, Postgres, MongoDB. |
| **Cloud Technologies** | AWS, Microsoft Azure, Google Cloud Netflix Eureka, Mesos, IBM Cloud Private. |
| **Micro Services** | Docker, Kubernetes |

**EDUCATIONAL DETAILS:**

* Bachelor ’s of Technology from TKR College of Engineering & Technology.

**PROFESSIONAL EXPERIENCE:**

**Client: Wells Fargo, New York June 2022 – Till Date**

**Role: Sr. Full Stack Java Developer**

**Responsibilities**:

* Developing and deploying web-app using Java, Angular and SQL Server stack for Enterprise Management in Legal Domain.
* Developing single page application and reusable components using Angular and fine tuning the SQL queries in SQL Server.
* Used Java 11's HttpClient API for streamlined HTTP requests and modernized REST API communication.
* Applied Java 8's Optional API to handle null values safely, preventing runtime null pointer exceptions.
* Developed middleware service and data access layer using Spring and Hibernate and used Spring Declarative Transaction management, JDBC queries.
* Managed WebLogic application server clusters for high availability and scalability in enterprise applications.
* Extensively used Angular, NodeJS, JQUERY, JSON, AJAX and DOM scripting to create interactive web applications like message posting and auto complete form validations.
* Implemented data aggregation pipelines in MongoDB to process large datasets efficiently.
* Automated application testing and deployment processes with GitLab CI/CD, improving release frequency and reliability.
* Implemented modules in Java 11 to improve encapsulation and enhance security across microservices.
* Proficient in Kubernetes, orchestrated containerized applications to ensure scalability, resilience, and efficient resource management.
* Configured and managed Kubernetes clusters, deploying applications across multiple nodes for high availability.
* Utilized Java 8 features such as lambda expressions and streams to write more efficient and concise code, improving readability and performance.
* Developed and implemented RESTful APIs to enable seamless communication between frontend and backend services.
* Designed REST APIs adhering to industry best practices for scalability, security, and maintainability in distributed systems.
* Architected and deployed scalable infrastructure on Azure using services like Azure Virtual Machines, Azure Blob Storage, and Azure SQL Database, resulting in reduced hosting costs.
* Implemented real-time, bi-directional communication between clients and servers using WebSocket, enhancing user engagement in live applications.
* Developed WebSocket endpoints in Java/Spring to handle persistent connections, enabling efficient data streaming and low-latency updates.
* Designed and developed scalable and resilient backend solutions using Azure Functions, enabling seamless integration with data sources and business logic for Wells Fargo's financial applications.
* Engineered data storage solutions utilizing DynamoDB and Serverless Aurora to store and retrieve financial data securely and efficiently, meeting the stringent compliance and regulatory requirements of the financial domain for the Wells Fargo project.
* Integrated Bloomberg Terminal data feeds into Wells Fargo's trading platform, providing real-time market data, analytics, and trade execution capabilities for traders.
* Leveraged Bloomberg's API to automate data retrieval and streamline trading operations, improving decision-making efficiency within the platform.
* Designed and implemented Kafka-based messaging systems to handle high-throughput data streams, improving real-time data processing capabilities.
* Configured and managed Kafka clusters, ensuring optimal performance, scalability, and fault tolerance in production environments.
* Designed and implemented behavior-driven development (BDD) test cases using Cucumber, enhancing collaboration between developers, testers, and business stakeholders.
* Integrated Cucumber with testing frameworks like JUnit and TestNG to automate end-to-end testing of web applications.
* Proficient in managing and troubleshooting Linux/Unix servers, ensuring system stability and optimal performance.
* Automated system administration tasks using shell scripting in Linux/Unix environments, improving operational efficiency.
* Integrated DocuSign API into web applications to automate electronic signature workflows, improving document management efficiency.
* Developed custom solutions using the DocuSign API for sending, tracking, and managing legally binding electronic documents.
* Built RESTful APIs using Flask and Python, enabling efficient interaction between microservices and databases.
* Managed MongoDB replica sets for high availability and fault tolerance in distributed applications.
* Deployed Java EE applications on WebLogic servers, ensuring optimal performance and high uptime.
* Managed Spark clusters using Apache YARN and Spark Standalone mode, ensuring resource efficiency and stability of data processing workloads.
* Automated the deployment and monitoring of Spark jobs in production, ensuring continuous operation and quick resolution of any issues.
* Developed React components and user interfaces for Wells Fargo project within the financial domain, ensuring compliance with industry standards and regulations.
* Implemented complex data visualizations and interactive dashboards using React to provide comprehensive financial insights for Wells Fargo clients.
* Developed responsive web pages using HTML5 and CSS3, ensuring compatibility across multiple browsers and devices.
* Implemented best practices in HTML for SEO and accessibility, improving search engine rankings and user experience.
* Optimized CSS for performance by minimizing file sizes, using CSS pre-processors like SASS/LESS, and implementing efficient selectors.

**Environment**: Java 8, Java 11, Angular, SQL Server, Spring, Spring Boot, Hibernate, Azure, Kubernetes, Docker, GitLab CI/CD, WebLogic, MongoDB, Kafka, Python, Flask, DynamoDB, Spark, React.js, HTML5, CSS3, SASS, Linux/Unix, JUnit, Maven, Cucumber, Apache YARN.

**Client: Samsung, New York. Nov 2020 to May 2022**

**Role: Java Full Stack Developer.**

**Responsibilities**:

* Actively involved in Analysis, Design, Development, System Testing and User Acceptance Testing. Successfully followed agile methodology in the Scrum Cycle model to help, respond quickly to business needs.
* Used Spring Inversion of Control (IOC) for injecting the beans and reduced the coupling between the classes. Implemented the Model View Controller using Spring MVC.
* Managed Kubernetes clusters across multiple environments using tools like Rancher and OpenShift for centralized control.
* Optimized multithreading processes using CompletableFuture and ExecutorService APIs in Java 11.
* Used spring actuator configuring Spring Boot application, monitoring, and invoking HTTP endpoints.
* Automated backup and recovery processes for MongoDB databases, ensuring data integrity and availability.
* Implemented the application using Spring Boot Framework and handled the security using Spring Security. Used XML Web Services using SOAP to register agents using non-java application.
* Leveraged Java 8’s parallel streams for processing large datasets, achieving performance gains of up to 50%.
* Designed and developed Microservices using synchronous protocols like HTTP, REST(JAX-RS).
* Integrated third-party APIs using REST protocols to enhance application functionality and provide extended service capabilities.
* Configured and optimized WebLogic JDBC data sources for improved database connectivity and transaction management.
* Configured secure webhook endpoints for receiving real-time event data from external APIs, improving monitoring and alerting capabilities.
* Developed custom webhook handlers to process incoming data for various automation workflows, reducing manual intervention.
* Developed microservices-based architecture using Spring Cloud for distributed systems, enabling seamless communication and scalability.
* Integrated Spring Cloud Config Server to manage centralized configuration across multiple services, ensuring consistency and flexibility in environment-specific settings.
* Utilized AWS Lambda for serverless computing, reducing operational overhead and enabling cost-efficient execution of application logic.
* Managed and orchestrated containerized applications using AWS ECS and EKS, ensuring seamless deployment and scaling of microservices.
* Automated data extraction and transformation processes using Python libraries such as Pandas and NumPy.
* Built dynamic, data-driven forms and interfaces in Unqork to capture and process user input, enhancing user experience.
* Utilized Cucumber to perform functional testing, regression testing, and acceptance testing, ensuring application reliability and performance.
* Integrated Cucumber with continuous integration (CI) pipelines, enabling automated test execution in tools like Jenkins for faster feedback on code quality.
* Troubleshooted and resolved issues related to WebSphere configuration, application deployment, and runtime errors, ensuring smooth operation of Samsung’s WebSphere environments.
* Utilized DocuSign API to automate document generation, signature requests, and form submissions, reducing manual processes.
* Implemented DocuSign API for secure, real-time document tracking, ensuring full visibility and control over the signing process.
* Utilized WebSocket for live chat applications, financial trading platforms, and real-time notifications, improving user experience with instant feedback.
* Integrated WebSocket with front-end frameworks like Angular and React to provide seamless real-time updates in dynamic web applications.
* Integrated Java 8 features with Spring and Hibernate frameworks, ensuring seamless implementation of modern Java capabilities in enterprise applications.
* Utilized the enhanced var keyword in Java 11 to write cleaner and more readable code across multiple projects.
* Engineered data storage solutions utilizing DynamoDB and Serverless Aurora to store and manage critical data with high availability and scalability for Samsung's cloud-based platforms.
* Involved in writing SQL and Stored Procedures for handling complex queries with help of TOAD and access them through Java Programs from Oracle Database.
* Developed CI/CD pipelines using OpenShift’s integrated Jenkins, automating build, test, and deployment processes.
* Integrated Prometheus and Grafana for monitoring Kubernetes clusters, enhancing observability and issue resolution.
* Developed Kafka producers and consumers in Java/Scala, enabling efficient data ingestion and processing across distributed systems.
* Implemented Kafka topic management and partitioning strategies to balance load and enhance message throughput.
* Automated infrastructure provisioning using AWS CloudFormation and Terraform to manage cloud resources effectively.
* Troubleshooted WebLogic server performance issues using JVM tuning and garbage collection analysis.
* Integrated Kafka with Apache Spark for real-time stream processing, facilitating advanced data analytics and insights
* Implemented and monitored fault-tolerant message routing in Apache Camel, ensuring high availability and reliability of critical business processes.
* Implemented unit testing for Python applications using unittest and pytest, increasing code coverage by 35%.
* Developed and maintained complex data migration scripts in PL/SQL to facilitate seamless transfer of data between heterogeneous databases and platforms.
* Implemented NoSQL databases such as MongoDB within Samsung's projects, leveraging their flexible data models and horizontal scalability to store and manage large volumes of unstructured data, enabling agile development and seamless integration with modern applications and services.

**Environment**: Java 8, Java 11, Spring Boot, Spring Security, Spring MVC, Kubernetes, AWS ECS, AWS EKS, DynamoDB, AWS Lambda, Docker, MongoDB, WebSphere, WebLogic, Prometheus, Grafana, Kafka, Apache Camel, Jenkins, OpenShift, Python, Terraform, Cucumber, PL/SQL, Unqork, HTML5, CSS3, jQuery, React.js.

Client: Charter Communications, Charlotte, NC. July 2018 to Oct 2020

Role: Java Full Stack Developer

**Responsibilities**:

* Involved in various phases of Software Development Life Cycle (SDLC) such as requirements gathering, modelling, analysis, Design, Development, and Testing.
* Created reusable functions and predicates with Java 8’s functional interfaces, increasing code reusability.
* Followed agile methodology, interacted with the client, provide/take feedback on the features, suggest/ implement optimal solutions and tailored application to customer needs.
* Used java 1.8 Stream and Lambda expressions available as part of to store and process the data.
* Involved in development, design and implementation front end part of the widget-based application using HTML5, CSS3-LESS, AJAX, Bootstrap 4.0.
* Implemented security best practices in WebLogic using SSL, TLS, and role-based access control for enterprise applications.
* Developed and integrated unit and integration tests with JUnit 5 for Java 11-based applications.
* Experience with the new features of Angular 7 new if else syntax, ng-templates, form validators, Routers.
* Highly used Angular 5 to build Single Page Application for navigation through the different status and multiple modals.
* Configured Kubernetes Ingress controllers for load balancing and external access to services, improving availability.
* Created AWS Lambda functions in Python for serverless architecture, reducing infrastructure management overhead.
* Managed CI/CD pipelines on AWS CodePipeline and CodeDeploy to automate the software delivery process.
* Implemented Services and Dependency Injection in Angular 7 to connect the web application to back-end APIs and for sharing the code between the components.
* Implemented Micro services using Spring Batch, Spring Boot, and Spring Security.
* Designed a MongoDB sharding and replication strategy that enhanced database scalability and availability for a cloud service platform.
* Implemented data aggregation pipelines in MongoDB that reduced data processing time by 30% for real-time analytics applications.
* Implemented webhook to automate real-time notifications and updates between systems, improving response times and operational efficiency.
* Integrated webhook with third-party services to enable seamless data exchange, enhancing overall application functionality.
* Implemented business layer using Core java, Spring Beans using dependency injection, spring annotations.
* Implemented Microservices using Spring Boot, Spring Cloud, Spring Microservices.
* Used Micro services architecture, with Spring Boot-based services interacting through a combination of REST and MQ to build, test and deploy Microservices.
* Optimized database performance by leveraging JPA features like lazy loading, caching, and batch fetching, improving application response times.
* Used JPA Criteria API to create dynamic and type-safe queries, ensuring robust and flexible data access logic.
* Integrated Apache Kafka with microservices to enable asynchronous communication and event-driven architectures.
* Enhanced legacy applications by incorporating Java 8’s CompletableFuture for asynchronous processing.
* Explored and applied garbage collection improvements in Java 11 to boost application performance and stability.
* Implemented Continuous Integration and Continuous Delivery (CI & CD) Process stack using DevOps tools like Jenkins.
* Integrated AWS SQS and SNS for building event-driven architectures and decoupling system components.
* Documented ELT processes and workflows, providing clear guidelines and reference materials for team members and stakeholders.
* Automated deployment processes using Web Logic Scripting Tool (WLST) to streamline application delivery.
* Integrated third-party APIs using Python, simplifying data retrieval and processing from external systems.
* Used Swagger as medium for modifying the web services URL to feed the input to UI.
* Collaborated with teams to upgrade enterprise applications to Java 11, ensuring backward compatibility and successful deployments.

**Environment**: Java 8, Spring Boot, JPA, Spring Batch, Spring Cloud, Spring Security, Microservices, Angular 7, Angular 5, Kubernetes, Docker, AWS Lambda, AWS SQS, AWS SNS, PostgreSQL, MongoDB, Kafka, Jenkins, Redis, Swagger, JUnit, WebLogic.

**Client: State of Mississippi, Mississippi Mar 2017 to June 2018**

**Role: Java Developer**

**Responsibilities:**

* Developed VG infra website and responsible for Database administration, content updates and ensured website performance and user experience. Understanding website requirements and participating in activities like gathering business requirements.
* Created POJO’s and DAO’s for the database entities using Spring JDBC annotation mappings.
* Worked in Agile development environment.
* Used Spring Core annotations for Spring Dependency Injection, Spring MVC for Rest API’s and Spring Boot for microservices.
* Secured AWS environments with IAM roles, policies, and multi-factor authentication (MFA) to enforce best practices.
* Automated Kubernetes cluster provisioning and deployment using Terraform and Helm charts.
* Monitored WebLogic server health and performance using Oracle Enterprise Manager (OEM), identifying and resolving bottlenecks.
* Hands on experiences with Core Java with Multithreading, Concurrency, Generics, Data Structures and Collections.
* Led the transition of multiple services from Java 7 to Java 8, unlocking new features like method references and the forEach loop.
* Developed and maintained data persistence layers using JPA to interact with relational databases, ensuring efficient data storage and retrieval.
* Implemented complex entity relationships, such as one-to-one, one-to-many, and many-to-many associations, using JPA annotations for seamless data mapping.
* Optimized Angular applications by using lazy loading, reducing the initial load time and improving performance.
* Participated in code reviews, Sprint Demos, Sprint Retrospective Grooming with the sprint team.
* Developed Front-end UI using JSP, Servlets, HTML5 and Java Script.
* Developed robust JavaScript and Node.js applications for the State of Mississippi project, delivering scalable and high-performance solutions to meet governmental requirements efficiently.
* Leveraged Python’s boto3 library to interact with AWS services, automating tasks such as EC2 instance management.
* Utilized Node.js for server-side scripting and API development, ensuring seamless communication between frontend interfaces and backend systems for the State of Mississippi project.
* Involved in Unit Testing, Integration Testing and UAT Testing.
* Deployed the application on the WebLogic Application Server and integrated applications using Apache Camel.
* Actively involved in code review and bug fixing for improving the performance.

**Environment**: Java 8, JPA, Spring MVC, Spring Boot, Kafka, Angular, Python, Node.js, AWS (EC2, S3, IAM), Kubernetes, Terraform, MongoDB, WebLogic, Apache Camel, JSP, HTML5, Servlets.

**Client: POLARIS, Hyderabad, India. Nov 2013 to Dec 2016**

**Role: Java Developer**

**Responsibilities:**

* Involved in the Requirements collection & Analysis from the business team.
* Created the design documents with use case diagram, class diagrams, and sequence diagrams using Rational Rose.
* Deployed and managed containerized applications in Kubernetes clusters, ensuring efficient orchestration and scaling.
* Implemented the MVC architecture using Apache Struts1.2 Framework.
* Implemented Action Classes and server-side validations for account activity, payment history and Transactions.
* Implemented views using Struts tags, JSTL2.0 and Expression Language.
* Implemented Tiles Framework for the view’s layout.
* Utilized the java.time package to manage date and time operations, eliminating errors related to older date-handling libraries.
* Leveraged AWS Elastic Load Balancer (ELB) and Route 53 to manage traffic distribution and improve application availability.
* Implemented session beans to handle business logic for fund transfer, loan, credit card & fixed deposit modules.
* Worked with JAXB, SAXP and XML Schema for exporting data into XML format and importing data from XML format to data base and JAXB in the web service's request response data marshalling as well as marshaling process.
* Utilized Kafka Streams for real-time analytics, transforming and processing event streams with low latency.
* Integrated MongoDB with Python and Java applications to handle unstructured and semi-structured data.
* Configured WebLogic server clusters with load balancing and failover mechanisms for high availability.
* Integrated Java 8’s default methods into interface-driven architectures, improving maintainability and flexibility.
* Implemented web services client to consume the third-party service API for validating credit cards. Used XML Web Services using SOAP to transfer application that is remote and global to different financial institutions.
* Used JMS for reliable and asynchronous exchange of important information such as payment status report.
* Integrated JMS with enterprise message brokers like ActiveMQ and RabbitMQ to handle large volumes of transactional messages.

**Environment**: Java 8, Apache Struts, WebLogic, MongoDB, Kafka, AWS ELB, Route 53, JMS, Spring Boot, JAXB, XML, SOAP, SQL, JSTL, Tiles Framework.