**Vikas K**

vikaskoduri.java@gmail.com

+1(913)-732-0043

**PROFESSIONAL EXPERIENCE:**

* Over **8 years** of professional full stack development in IT industry with exposure to all the stages in Software Development Life Cycle (**SLDC**) including gathering requirements, analysis of requirements, software design, development, testing and maintenance of **web applications** using **JAVA/J2EE** technologies.
* Expertise in Designing and Developing **User Interface** using **HTML5**, **CSS3, Bootstrap**, **JQuery, ECMASscript6(ES6), ECMAScript5(ES5), Javascript, Typescript**.
* Well understanding the features of **ES6** like arrows, promises, modules, block scoping, enhances object literals to build highly interactive pages.
* Expertise in implementing **MVC** architecture on client side using **Angular JS, React with redux, React with flux, Backbone JS, Ext JS, knockout JS.**
* Developed view rendering pages using **React.js.**
* Experience in implementing **MVVM** architecture on client side using **Angular 2, Angular 4 , Angular 5 with typescript 2.7**.
* In **Angular** implemented various concepts like **components, directives, Data Binding, Event Binding, written services for dependency injection, Observables, Shadow DOM.**
* Implemented angular **observables** and listeners (**RxJS** libraries) to simplify making asynchronous calls to backend **REST** **API’s.**
* Experience in using **Angular 4 router** to build **single page application (SPA)** to navigate through the different components.
* Experience in developing server-side **JavaScript** frameworks using **NodeJS** along with **Express JS** to build **event driven, non-blocking i/o model** web applications.
* Experience in developing and publishing **RESTful** **web services** using **Node Js Asynchronous programming** and **express JS.**
* Experience in developing **Restful API’S** with **Hapi.js** and followed Event driven approach with **Node.js**.
* Expertise in working with **Hapi.JS** to build reusable **application logic** by avoiding spending time on **building infrastructure**.
* Developed applications using **node JS Socket.io programming** for persistence connection from client to the server.
* Experience in developing applications with **Angular 2, Node Js, Express Js and Mongo DB** to keep application synchronized with the database.
* Expertise in using of Java Script build tools and task runners like **Webpack** for React, **Grunt, Gulp** for Angular 2.0/4.0, **Browserify** for packaging Node JS modules for browsers.
* Good amount of working experience with web application’s **UI testing frameworks** such as **Karma, Jasmine,** **Protractor** for Angular JS, **Mocha and Chai** for **React JS** and **Backbone JS**.
* Worked extensively with **JAVA 8** features like **Stream API,** **Time API,** **Functional Interfaces Multithreading, Transaction Management, Exception Handling, Collection API.**
* Extensively worked on **Microservices** for creating API’s services with **Spring Boot** to modularize code.
* Experience in developing and deploying **Micro Services based** on **Spring boot** and **Spring Cloud**.
* Expertise in using **Spring Boot** to create **stand-alone** applications and for building **web service.**
* Good understanding of **Spring annotations** as well as **XML configuration** for **dependency injection**.
* Experience in implementing **Swagger** for **dynamic API documentation.**
* Expert in integrating **JMS (Java Message Service)** with **SPRING BOOT** by providing an instance which is **embed** with **Active** **MQ.**
* Experience in implementing **Service Oriented Architecture** (SOA) with web services like **Simple Object Access Protocol** (SOAP) and service based technology like **Representational State Transfer** (REST).
* Extensively worked on **SOAP** protocol using **JAX WS** and **Apache CXF** and documented services using **WSDL**.
* Developed projects with **RESTful** services (**JAX-RS, Apache CXF**) and REST resources that include components like **Jersey** and used **WADL** for **documenting REST API’S.**
* Extensively worked on **servlets, JSP, Structs, JSF, Spring MVC, Spring IOC, Spring AOP, Spring JDBC, Spring Data, Spring I/O, Spring REST, Spring JTA, Spring security, Spring batch** to implement **Monolithic Architecture.**
* Experience in Deploying and Configuring **Application servers and Web Servers** such as **Oracle WebLogic, IBM WebSphere, JBoss, Glassfish and Apache Tomcat**.
* Good knowledge in Object Relational Mapping (**ORM**) framework which includes **Hibernate**, **Spring JPA, Apache iBATIS, Spring Data, Spring JDBC.**
* Experience in using with **Hibernate’s second level cache** by **Ehcache**.
* Experience in Relational Database Management System (**RDBMS**) databases **like Oracle 10g/11g/12g, MySQL5.x, DB2, PostgreSQL.**
* Experience in **NoSQL** database like **MongoDB, Apache Cassandra, Apache CouchDB, Graph database**.
* Proficient in writing stored **procedures, functions and triggers using. PL/SQL in JDBC, ODBC** in **WebLogic** **server.**
* Experience using **Object Document Mapper** (**ODM)** frameworks to connect asynchronous environment with **Mongoose Connector** and **Cassandra Node Driver.**
* Expert command on implementing **Spring Cloud- Netflix OSS Stack, Zuul (API Gateway), OAuth 2.0. (authentication framework), Hystrix (Circuit Breaker)** and **Eureka** for service registration
* Expertise in applying **JMS** **(Java Messaging Service)** for reliable information exchange across Java applications using **Apache Kafka, RabbitMQ, ActiveMQ and IBM MQ** tocommunicate from Client to server.
* Experience in using **Apache Kafka (Message Queues)** for reliable and asynchronous exchange of vital information between multiple business applications
* Good knowledge in using **Apache Kafka** in cluster as messaging system between the APIs and micro services.
* Experience in maintaining log delivery system by creating producer, consumer and broker architecture using **Apache Kafka**.
* Experience in building real-time **dashboard applications** for visualizing, analyzing and exploring data on **Apache Web servers** using **Elastic Search, Logstash** and **Kibana 4.0 (ELK), Splunk.**
* Experience in using various **AWS (Amazon Web Services) IaaS (Infrastructure as a Service), PaaS (Platform as a Service)** Components like **EC2** for running applications on virtual machines, **Lambda** for serverless program functions, **S3** for storage, **Amazon DynamoDB** which provides seamless scalability and fast performance, **Amazon Elastic Load Balancing** for distributing traffic on the network.
* Expertise in using **Microsoft Azure** which is cloud computing service for building, testing, deploying and managing applications and services through a global network of Microsoft-managed data centers.
* Expertise in **Pivotal Cloud Foundry (PCF)** which is an open source **cloud** **platform as a service (PaaS)** on which developers can build, deploy, run and scale applications.
* Used **Docker Container** to address every application across the **Hybrid Cloud and Docker** for better collaboration and innovation.
* Hands on experience in managing this **Docker Container** using **Kubernetes** with zero downtime deployment.
* Experience in **Oauth 2.0** providing security for authorizing the users, **single sign on** for authenticating, **JWT** for producing tokens.
* Implemented **Docker** container service, build pipe line like **Jenkins, Bamboo and GoCD** for Continuous Integration and Continuous Deployment that can run on the cloud or on-premises.
* Experience in automation tools like **MAVEN, ANT, Gradle** for project builds, documentation and dependency.
* Strong Experience in working with **JAVA** based testing frameworks **like JUint, Selenium, TestNG, Mockito, Spock, Cucumber**, **Powermock**, **MockMvc, etc.**
* Experience in Web Application testing using **Selenium, Cucumber** (**Body Driven Development).**
* Experience in working with various version controlling tools like **Git, SVN, Bitbuckets** and different Project management tools like **JIRA, Rally, Microsoft Project.**
* Experience in using **Drools** which is **Business Rule Management System (BRMS**) for defining business rules and Validations.
* Experience in using **Mule soft** which is light weight Java based **Enterprise Service Bus** **(ESB)** which allows connecting applications together quickly and easily, enabling them to exchange data.
* Experience in using open source framework **Apache Camel** which for message-oriented middleware with a rule-based routing and mediation engine that provides a Java object-based implementation of the **Enterprise Integration Patterns (EIP)** using an application programming interface to configure routing and mediation rules.

## technical skills

|  |  |
| --- | --- |
| **Programming Languages** | C, C++, Java, PL/SQL, UNIX Shell Script |
| **JavaScript Frameworks** | HTML5, CSS3, JavaScript, AJAX, JSON, Bootstrap, jQuery, Angular2.0/4.0, React JS with Flux, Redux, Ember JS, Knockout JS, Back Bone JS, Handlebar JS, Node JS, Express JS, Hapi JS |
| **Java/J2EE Technologies** | Servlets, Spring, EJB, JPA, JTA, JDBC, JSP, JSTL, |
| **Frameworks** | Struts 2.0/1.0, Hibernate 4.x/3.x, Spring 3.0, Spring Batch, Spring Security, Spring AOP, Spring Core, JSF, IBatis, Spring IOC. |
| **Web Technologies** | HTML, HTML5, CSS/CSS3, AJAX, jQuery, Bootstrap, XML, Grunt, Gulp, JSON. |
| **Database** | Oracle 8i/9i/10g/11i, DB2, SQL Server 2000/2005/2008, MYSQL, MongoDB, Raven DB, Cassandra |
| **Web Services** | XSD, WSDL, Apache Axis, SOAP, Restful, GraphQL, JAX RS, JAX WS |
| **Messaging** | JMS (Java Message Service), IBM MQ, Apache Active MQ, Apache Kafka |
| **Web/Application Servers** | Oracle Web logic, IBM Web Sphere, JBoss, Glassfish, Apache Tomcat |
| **Messaging Services** | JMS, Active MQ, Rabbit MQ, IBM MQ, Apache Kafka |
| **Methodologies** | Agile, Waterfall, OOAD, SCRUM |
| **Testing Tools** | Junit, TestNG, Mockito, Jasmine, Karma, Mocha, Chai, Spock, Protractor |
| **Version Control** | SVN, CVS, GIT and Clear Case |
| **Build& Configuration Tools** | Ant, Maven, Gradle, Jenkins, Bamboo Gulp, Grunt, Webpack, NPM, Bower |
| **Project Management Tools** | JIRA, Rally, Microsoft Project |
| **Cloud Technologies & DevOps Tools** | AWS, PCF, EC2, S3, Docker, ECS, EBS, EKS, SQS**,** RDS, S3, Glacier, Lambda, Kubernetes |
| **IDEs** | Eclipse IDE, RAD 6.0/7.0, Intellij IDEA, Spring Tool Suite (STS), Sublime Text, Putty |
| **Operating Systems** | Windows, Linux and UNIX |
| **Software** | Visual Studio, Rational Rose and Microsoft Office |

## professional experience

**Client**: **AVERA Health, Sioux Falls, SD**. **(Feb 2017-Current)**

**Role:** **Full Stack Java Developer**

**Description**: Avera Health is a major health care system in upper Midwest and provides top-quality health services. It serves more than 1 million people through hospitals, clinics, senior living facilities, sports and wellness facilities in more than 300 locations. My role in this project is to develop web applications to quicken health record functionalities, check the patient details, pull the insurance and patient record and process the health services.

**Responsibilities:**

* This application is built on **Micro Service Architecture** using **MEAN** (**Mongo, Express, Angular 4.0 and Node**) stack and developed **Single Page Applications (SPA).**
* Designed efficient Interaction pages using **HTML5, CSS3** and **Bootstrap CSS** framework and **media queries** to create responsive web pages.
* To allow fully encapsulate custom elements, created custom elements, **HTML imports** and **Shadow Dom, pipes** using **Angular 4** and implemented **Angular Router** to enable **navigation** from one view to the next as the user performs application task.
* Implemented form validation on **HTML** forms using the built-in validation properties and methods in **Angular 4.0**.
* Have worked with **reactive forms, template driven forms** and written **pipes** in **Angular 4.0** to transform the data.
* Developed **client-side** **validation** code using **JavaScript** and **JQUERY**.
* Worked on various features in **ECMAScript 6 (ES6)** such as Arrow functions, Classes and enhanced object literals.
* Created **cross-browser compatible** and standards-compliant CSS-based page layouts.
* Used **Angular 4 Lazy Loading** and ahead of time compilation to optimize the performance of application.
* Developed angular **views** to hook up models to the **DOM** and synchronize data with server as a **SPA.**
* Used Built-in **AJAX HTTP Service** with **observables** extensively for making quick and incremental updates to **UI** of the application.
* Developed **RESTFUL** web services using **Spring Boot.**
* Used **Microservices** for creating API’s services with **Spring Boot** to modularize code and implemented **Swagger** for **dynamic API** documentation.
* Used complete annotation driven approach to develop the application using **Spring Boot** and was involved in setting up the **Spring bean profiling**.
* Extensively worked in integrating **JMS** **(Java Messaging Service)** with **SPRING BOOT** for reliable information exchange across Java applications using **ActiveMQ** to **communicate** from **Client to Server.**
* Developed **Microservices** using **Spring Cloud Netflix OSS** stack to address **Load balancing** using **Ribbon,** API Gateway using **Zuul,** Fault Tolerance using **Hystrix.**
* Unit testing of web services is done using **Spock** framework by extending **Groovy** syntax.
* Used **OAuth 2.0 protocol** for authorization of server and **JWT** to generate to access a protected resource on a resource server.
* Developed **Mean Stack application** from scratch including **Mongo DB** server setup and **Express-JS server** development.
* Developed **RESTful** web services using **Node JS** and **Express JS** and used **Node JS** server to interact with the **Rest services** and **database**, hosted on multiple load balanced cloud instances.
* Implemented services call from **Angular** at client to **Node** **JS** using **Express JS**.
* Implemented **Node JS** as a **server-side proxy** for an event-driven, non-blocking I/O model to make the application lightweight and efficient.
* Configured **Express JS** node http framework to set up middle-wares to respond to HTTP requests and used its front-end tooling like **Web packs, Starter packs, Boiler plates** efficiently.
* Used **Postman** to test the API workflow tested on the **JSON files** which are generated by the **NodeJS, Express JS.**
* Focused on **Test Driven Development (TDD**) for UI testing we have used **Jasmine** and **Karma** for **Angular** and for back end we used **Mocha** and **chai** for **Node JS, Junit** and **Mockito** for **Spring Boot** and embedded databases like **HSQL** and function testing we use **Spring MVC** mock.
* Handled unstructured data using **MongoDB** and used **Mongoose** **connector** to connect to database.
* Composed and executed **MongoDB scripts** and queries to fetch and test data from **NoSQL databases.**
* Performed configuration, deployment and support of cloud services including **Amazon Web Services (AWS)**. Launch various **AWS EC2** instance to deploy the application.
* Implemented **AWS** solutions using **E2, 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 Lambda** to run the code and manage the servers in **AWS** with the help of **API**.
* Deployed application into **Docker container** and **Docker consoles** for managing the application life cycle on **Kubernetes deployment** manager and monitored on **Git**.
* Experience in maintaining log delivery system by creating producer, consumer and broker architecture using **Apache Kafka.**
* Used **Apache Kafka** in cluster as messaging system between the APIs and microservices.
* Worked on **Splunk** search processing language (SPL) **queries, reports, alerts, dashboard, visualizing and logging** using **Log4j**.
* Worked with the automation team to understand the **Selenium scripts** and review the code, run the tests and review the results and provide inputs to the team.
* Used **Maven** as **build tool** and **Jenkins** for **CD/CI deployment** process and used **GIT** for version management.

**Environments:**

Angular 4, Java 1.8, Node JS, Express JS, HTML5, CSS3,MongoDB, Mongoose, jQUERY, ES6,Spring 4.2.0, AWS, REST services, OAuth 2.0, Swagger 2.0, Jasmine, Karma, Mockito, Apache Kafka 0.11.0.1, Junit 4, Log4j, Selenium 3.0, GIT, Kubernetes, Maven 3.3 and Jenkins.

**Client:** **Nike, Hillsboro, OR (Jan 2016- Feb 2017)**

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

**Description:**

Nike is a multinational corporation, design and manufactures athletic footwear and apparel. It is the most valuable sports brand in the country. My role in the project includes developing UI components for various modules, design and developing services and applicationsfor online ecommerce website which provides customer better experience in placing the order and managing status of the orders.

**Responsibilities:**

* Created **UI** specification documents by interacting with business users and design documentation and data modelling for the entire project.
* Designed and developed efficient web pages using **HTML5, CSS3-SASS** and **Twitter Bootstrap** **CSS** framework and **media queries** to create responsive web pages.
* Expertise in using **React.js/Redux** to build User Interface, strong knowledge of **state store, middleware, action creator, reducer and container.**
* Written several **Axios** service calls like **get** and **post** to make **REST service** calls.
* Developed user interface by using the **React JS, Redux for SPA** development and implemented client-side Interface using **React JS.**
* Created **responsive design** and developed a **single responsive** website that could be served to desktop, Tablets and mobile users using **React JS**.
* Implemented **Redux** Form to build Report **Data Grid, tree grid, and developed DataGridItems** which can be dragged by **IRD system** users.
* Implemented **Virtual Dom** for Client-Side view rendering services using **REACT JS.**
* Worked on Redux/**NGRX** and understanding of the Redux pattern.
* Wrote Client-Side code using **React JS** and used **NPM,** directories to generate the server environments used **browserify**, **flux** to manage with targeted URL's (Uni-Directional data flows) and to monitor the application.
* **REACT JS** Virtual **DOM** used for client-side view rendering services, **React-Redux** for state management **and React-Router** for programmatic navigation.
* Building cloud **Microservices** and implemented back-end using **Spring Boot.**
* Worked with **Microservice** architecture using **Spring Boot** to modularized code and implemented Rest API’s using **Spring Rest** and integrated **Swagger API** for documentation.
* Implementation domain base classes providing basic properties using **Spring Data**.
* Worked with **JAVA8** features like **Functional Interfaces, Stream API for Bulk Data Operations, Time API, Transaction Management, Concurrency API, Collection API**, **Calendar API**.
* Created reusable templates using Angular directives and worked with NPM package manager tools (**Node JS).**
* Experience in maintaining log delivery system by creating producer, consumer, and broker architecture using **Apache Kafka.**
* Used **Apache Kafka** in cluster as messaging system between the API’S and microservices.
* Experience in **Docker** Engine and **Docker** Machine environment, to deploy the micro services-oriented environments for scalable application.
* Proficient in writing react code with help of **ECMAScript (ES6) and JSX.**
* Deployed our Application on **Pivotal Cloud Foundry** which is used to reduce the development overhead by providing a ready to use platform.
* Deployed and Monitored Micro Services Using **Spring Cloud Foundry** Managed Domains and Routes with the Spring Cloud Service Registry.
* Implemented **Spring Cloud**- **Netflix ZUUL** to building microservices to communicate with each other, **Hystrix** for circuit breaker and **Eureka** to provide weighted load balancing based on several factors like traffic, recourse usage, authentication, monitoring services and routing.
* Implemented **Netflix OSS(Open Source Software)** for **Feasible changes** after the deployment.
* Implemented platform modules through **spring Java micro** and **Node.js services** by employing **Hapi.js**, as well as **front-end React JS.**
* Involved in writing **Promises** and **call back** function in **Node Js** to build **microservices** using **Hapi.JS**.
* Developed reusable application logic by avoiding building infrastructure by **Hapi.JS**.
* Worked on**Docker hub**, creating **Docker images** and handling multiple images primarily for middleware installations and domain configurations.
* Implemented **Spring Circuit breaker** pattern, integrated **Hystrix** dashboard to monitor Spring micro services.
* Configured **JBOSS** Application server and used **Oracle** database with **Hibernate ORM framework** and used the functionalities of **Hibernate’s** second level **EHCache.**
* Implemented Relational Data Base like **Oracle and PL/SQL** store procedures.
* Worked with **NoSQL** based **ORM** frameworks like **Cassandra Db** and used **Cassandra node driver**.
* Worked with **Cassandra Node Connector** for database persistence to interact with **Cassandra NOSQL DB** with **unify and ease** the access to **Cassandra** persistence stores.
* Performed Code building using **Gradle.**
* Implemented a metrics collection mechanism for the server by utilizing **Apache Kafka** to stream server-side events.
* Configured **Kibana** data visualization plugin **for Elasticsearch, Logstash** and created bar, line and scatter plots, or pie charts and maps on top of large volumes of data.
* Worked with **Web pack** for bundling application source code in convenient chunks and for loading that code from a server into a browser
* Worked with **OpenShift** for **Platform as a service with** configuration management.
* Authenticating Users and to provide the security to **RESTFul** web services using **OAUTH** **2.0**
* Implemented unit test cases using **Karma** and **Jasmine** on client side.

**Environment:**  Swagger, Node-JS, React, Redux, Web pack, Karma and jasmine, Rest,Spring 4.x, Spring Boot, Spring cloud like Ribbon, Hystrix, Netflix ZUUL, Micro services architecture, Docker, Cassandra Db, OAUTH, Apache Kafka, Kibana, OpenShift.

**Client: Quicken Loans, Campus Martius Park, Detroit (Sep 2014-Jan 2016)**

**Role:** **Full Stack Java / J2EE Developer**

**Description:**

Quicken Loans provides mortgages and is the second largest overall retail lender in the U.S. (it is also the largest online retail mortgage lender). My role in the project is to develop web applications to meet the mortgage needs of client, quicken loan deposits, pulling the credit report, ordering an appraisal and processing of loan.

## Responsibilities:

* Designed the user interface screen and its functionality as per user requirements and developed **RESTful** Web services in **Mule** **ESB** based on **Service Oriented Architecture (SOA)**.
* Developed dynamic web pages using **HTML5, CSS3, Bootstrap** and **AJAX.**
* Effectively migrated the **Angular JS** application to **Angular 2.0** to use the enhanced features of **type script, Angular Routers** and **Angular Components**.
* Used Angular 2.0 to create views to hook up models to the **DOM** and synchronize data with server as a **Single Page Applications**.
* Effectively used **Angular Directives, filters, declarative templates, service providers** and **context aware communication**.
* Performed Angular JS end to end testing using **Protractor** framework.
* Worked on using **Grunt** used for **concatenation**, **minification** and running of **unit test** cases.
* **Legacy** **Web Services** Systems using **SOAP (AXIS-2)** is migrated and implemented using **REST** (**Jersey** extending the **JAX-RS**) with data representation in different formats such as **XML** and **JSON**.
* Implemented **WADL** configuration and API documentation for the understanding of **Rest** based Web services.
* Wrote **custom marshaller** for webservices in the application for converting in-memory object into persisting or transportable format.
* **JAX-RS** API generic and pluggable interface called **Message Body Writer** for doing the custom marshalling.
* Written **IBatis** Criteria queries for data retrievals and have performed the **ORM** mappings to map **java entities** to database tables.
* Implemented second level cache using **Oracle Coherence** for **IBatis**.
* Worked extensively with persistence context (JPA) implemented by **IBatis.**
* **DB2** database used to load the application data and written **PL/SQL** procedures, complex queries to perform various data operations
* **Message broker services** are achieved in the application using **Active MQ** with **JMS** framework
* Used **Maven** build tool to deploy the application on web servers i.e. **JBoss.**
* Deployed and monitored services using **Open Shift** and managed Domains and Routes
* Configured **Jenkins** to handle application deployment in **Pivotal Cloud Foundry(PCF)** server and to integrate with **Git Hub** version control
* Implemented **Drools Business Logic Integration Platform** which provides a unified and integrated platform for Rules, Workflow and Event Processing
* **Business Unit, functional, end-to-end testing** is achieved using **TestNg** framework
* Integration testing methods and Behavioral driven testing in the application are implemented by using **Selenium** and **Cucumber** respectively

**Environment**: Angular JS, HTML5, CSS3, AJAX, Bootstrap, JSON, Active MQ, JMS, Hibernate, DB2, SOAP-AXIS2, Restful services, JAX-RS SOA, Eclipse Java EE IDE Neon.3,Jenkins, Git, Log4j, DB2, Grunt, Jasmine, JBoss, Jenkins, Open Shift, Drools, Maven, TestNg, Protractor, Grunt, Selenium, Cucumber, WADL.

**Client: COX Communications (Oct 2013-Sep 2014)**

**Role: Java Full Stack Developer**

**Description:**

Cox communications (also known as COX cable and formerly Cox Broadcasting Corporation, Dimension Cable Services and Times-Mirror Cable) is an American privately-owned subsidiary of Cox Enterprises. Providing digital cable television, telecommunications and home automation services in the United States. My role is to develop various applications like account management of users and processing technical issues raised by user.

## Responsibilities:

* Developed User Interface view with **Spring MVC** and **Backbone JS** as a **Service Oriented Architecture (SOA)** application.
* Used **HTML5, CSS, jQuery, Ajax** and performed client-side validations using **JavaScript.**
* Developed responsive UI modules using **Backbone.JS, Handlebar JS.**
* Designed and developed various UI features using **Backbone JS,** performed routing using **Backbone Router**.
* Used **Backbone JS** to utilize its rendering functions and create controllers to handle events triggered by clients and send request to server.
* Developed various building blocks such as **models, views, events, routers and collections** for assembling the client-side web applications using **backbone.js.**
* Made **Ajax calls** using **backbone.js** to render the view from the server side.
* Used **jQuery libraries** to query the DOM and handled the Events using backbone.js
* Implemented **Spring** Beans using IOC and Transaction management features to handle the transactions and business logic, **spring AOP** to increase code modularity by cross cutting concerns.
* Implemented Internationalization using **Spring MVC Interceptors** and spring core modules.
* Expertise in developing flexible and loosely coupled Application using **Spring MVC** and handled different **HTTP Request** and **Responses** using **Dispatcher Servlet**.
* Implemented **Spring MVC Validation** to validate the form inputs at the server side.
* Redirected the pages based on the Http Request and **handled Exceptions** using **Spring MVC**.
* Implemented spring **batch framework** to create custom readers and writers to work with the application and Spring security for authentication and authorization.
* Used **Hibernate** as **ORM framework** and **Hibernate JPA** implementation, annotated the classes with JPA annotations.
* Involved in multi-tiered J2EE design utilizing **Spring Inversion of Control (IOC)** architecture and **Spring JPA**.
* These applications need to maintain privacy and restrict illegal users. To achieve this functionality, we used **Spring Security**.
* Worked with Hibernate **session objects** to interact with database and used **HQL** to query the database.
* Worked with message queueing using **RabbitMQ** for exchanging data between processes, applications, and servers and involved in setting up a connection and publishing/consuming messages from a queue.
* Developed and enhanced data processing tool in Java by implementing **connection pooling** and **multithreading.**
* Created stored procedures, triggers, functions, views, indexes using **PostgreSQL** and written complex queries using JOINS and Subqueries to retrieve data.
* Worked with Maven for project build, dependency and documentation.
* Used **Jenkins** for continuous integration of the build and deployment for release process.
* Used **Drools** rules engine for defining business rules and Validations.
* Involved in performing Unit and Integration testing **Junit.**

**Environment**: Java, backbone.js, Handlebar.js, Hibernate JPA, Spring MVC, Spring JPA, Spring Security, PostgreSQL, Maven, RabbitMQ, Junit, Jenkins, Drools.

**Client: AXIS Bank, India (Dec 2012- Sep 2013 )**

**Role**: **Java Developer**

**Description:** Axis Bank is an India’s best private banking and financial services company. This is in house application portal and an Internet Banking product, designed and developed using Struts and J2EE to meet the demands of the Bank customer for transactions through Internet. In this we have integrated online services and made it is a powerful platform for business owners and customers.

**Responsibilities:**

* Deloped UI pages using **HTML5, CSS3, JavaScript, jQuery, AJAX and JSON**.
* Developed client side validation code using **Javascript and JQuery.**
* Involved in various phases of Software Development Life Cycle (**SDLC**) on the application like Requirement gathering, Design, Analysis and Code Development.
* Developed and implemented the **MVC Architectural** pattern using **STRUTS 2.x** framework.
* Implemented POJO based action by **STRUTS** and also utilized various Tag Support given by the **STRUTS 2.x**.
* Implemented **interceptor** which is invoke at preprocessing and post processing of a request in **STRUTS 2.x.**
* Implemented **OGNL(Object Graph Navigation Language) of STRUTS 2.x** to simplify the accessibility of data stored in the **ActionContext**.
* Loading data from Text Files to Database by using **SQL Loader.**
* Implemented **IBM WebSphere** Application Server to deploy the application.
* Worked with **Log4J** to capture the log that includes runtime exceptions.

**Environment**: HTML5, CSS3, JavaScript, JQuery, AJAX, JSON, SDLC, MVC Architectural, STRUTS 2.x, SQL Loader, IBM WebSphere, Log4J.

## Client: Flipkart, India (May 2010-Dec 2011)

**Role:** **Java Developer**

**Description:** Flipkartis an ecommerce company in Bangalore, India. It is the holds most of Indian ecommerce market share. Developed an application using Java Servlets and JSP which gives the customer account information like order details.

## Responsibilities:

* Developed a **Monolithic** application and generated UI screens using **JSP, HTML, CSS, JavaScript**.
* Worked on **JSP, Servlets, Struts framework** and production support issues of the existing applications.
* Involved in development of **Action Forms, Action Servlets**, **Action** and **Validating Action Forms** in **Struts framework.**
* Implemented **Struts Dispatch Action** class and form bean classes using **struts** framework.
* Client-server validations were handled using **Struts** **Validator Framework’s** validate plug-in.
* Implemented **Multithreading** for handling multiple requests and for high performance.
* Developed model components using session beans and used local message driven beans (MDB) to interact to session using **EJB**.
* Responsible in creating **Servlets**, which route submittals to appropriate Enterprise Java Bean (**EJB**) components and render, retrieved information and **EJB** Session Beans were used to process request from user interface using OSS.
* **Java message queue** (JMS) is used for exchanging the information and messaging services.
* Designed and developed **Message Driven Beans** (MDB) that consumed the messages from the **JMS**.
* Develop a business continuity plan for the **SQL Server Databases** using **JDBC** drivers.
* Created many **Stored procedures** and scheduled jobs to support our applications for to create reports for customers**.**
* Experience in installing, configuring and deploying application on **Web Logic Server**.
* Developed **DAO** pattern to retrieve the data from database.
* Developed **SOAP** **web services** (JAX-WS) for sending and getting data from different applications **WSDL** generation and exposed the service at server-side over the Internet and tested the services using **JUnit**.
* Deployed application on **Client Linux server**.
* Used **ANT** built tool for compiling and generating war file.

**Environment**: Core Java, J2EE, JSP, HTML, MVC, Servlets, Struts, JMS, MDB, EJB3.1, JUnit, PL/SQL, DAO, Web Logic Server, Ant, SOAP, JAX-WS, WSDL, Junit, SQL.

**eDUCATION**

**Bachelor’s in Information Technology Aggregate: 3.8/4.0**

**GITAM University, Visakhapatnam, India.**