Java Full Stack Developer Java Full Stack Developer Java Full Stack Developer - Sentury Tire Hialeah, FL   Around 7 years of full stack development experience in all phases of Software Development Life Cycle (SDLC) which includes Analysis, Design, Development, Documentation, Testing, Deployment, Version Control and production support of web based, enterprise business applications using the JAVA/J2EE technologies.   Mastery in UI Design and Development utilizing HTML5, CSS3 (SASS, LESS), Bootstrap and AJAX that takes after W3C Web Standards. Experienced in using ES6 feature's like arrows, modules, enhanced object literals, promises and block scoping.   Proficient in developing Dynamic Single Page Application (SPA) using MEAN. Sound knowledge on implementing Angular 2.x/4.x with NGRX to accomplish MVVM design flow. Expertise on using Router ParamMap in Angular 4 which is useful to query a ParamMap in router itself and also gives advantage on security aspects with all types.   Well Experienced in using Component features like Modules, Templates, Directives, and Meta Data.   Good amount of Knowledge on using directives, observables, promises, dependency injection, pipes, shadow Dom.   Expertise in writing Typescript which is useful in Angular 4.x applications that advanced features like auto completion, navigation, and refactoring.   Strong knowledge in React JS framework to develop the SPA.   Experience in using React JS components, Keys, Router, Forms, Events plus Redux, Animations and Flux concept.   Well Experienced for creating interactive UI's in React JS using Virtual DOM, One-way data flow, JSX, React Native concepts.   Expertise in making Custom Reusable React Components Library.   Well experienced in Angular JS, Backbone JS, Ember JS, Knockout JS, and Ext JS for making customize front end app development and to achieve MVC design flow.   Strong knowledge in developing applications for view rendering using JQuery, React JS, Handlebar JS.   Developed and deployed Micro Services based applications using Spring Boot along with Spring Rest and Node JS with Express JS and Hapi JS.   Expertise in Using Grunt, Gulp, Web pack, Bower.   Strong Experience on working with UI testing frameworks like Karma, Jasmine, Protractor for Angular JS, Mocha and Chai for React JS and Backbone JS.   Good Experience in Bug tracking tools like JIRA, HP Quality Center and Bugzilla.   Expertise in working with JAVA 8 features like Collection API, Stream API, Time API, Multithreading, Transaction Management, and

Exception Handling.   Expertise in making Stand-alone, production grade applications using Spring Boot.   Developed controllers and services for handling the requests using Spring Boot.   Well experienced in using Spring Boot feature's like Spring Boot Actuator, Spring Boot JPA, Spring Boot JDBC.   Good amount of experience in integrating Swagger documentation into a Spring Boot based Rest web service mainly by using spring fox implementation for generating the swagger documentation.   Expertise in SOA using Spring REST to create REST resources which has various components like JAX-RS, Jersey Framework, Rest Easy and WADL(for documenting REST API'S).

   Good working knowledge on integration of Drools Rule Engines which corresponds to validating BRMS (Business Rules Management System) with Spring Bean.   Strong knowledge in Amazon Web Services like Amazon S3(Amazon Simple Storage Service),Amazon EC2(Elastic Compute Cloud),Amazon Simple DB, Amazon EBS, Amazon cloud front, Amazon SQS, Amazon Elastic Load Balancing.   Worked with Jenkins for continuous integration and Performance improvements for the application.   Knowledge on various Docker components like Docker Engine, Hub, Machine, Compose and Docker Registry.   Well experienced with build tools like Gradle, Maven and ANT. Good knowledge on package manager tools such as NPM and Bower for dependency management.   Hands on experience in software configuration/change control process and tools like Subversion (SVN), Git, and CVS.   Expertise in logging tools like Log4j, slf4j.   Experience in writing and executing unit test cases using JUnit and JMeter, Mokito Testing Framework.   Strong Knowledge on automated java-based integration testing and behavioral testing of the applications using Selenium and Cucumber respectively.   Good experience in using Software development methodologies like Agile(Scrum), Water Fall and Rational Application Development methodologies Authorized to work in the US for any employer Work Experience Java Full Stack Developer Sentury Tire - Hialeah, FL November 2018 to Present Description: Sentury Tire is a subsidiary of Sentaida Group Co. Ltd., born in Qingdao, Shandong, China, in 2008. Sentury Tire Co. Ltd started mass production in 2009, principally engaged in passenger, crossover, sport utility vehicle, commercial van and aircraft tires. Sentury's facilities cover an area of 9.6 million square feet with an annual capacity of 9 million tires in 2013   Responsibilities: Implemented Micro Service Architecture using

Mongo DB, Hapi JS, Angular 4 and Node JS stack and developed Single Page Applications (SPA).

Designed the front-end using web technologies likeHTML5, CSS3 and Twitter Bootstrap framework and Angular 4 for creating grid layout, responsive screen sizes and devices.   Developed reusable Angular 4components which are compatible with latest versions of typescript and angular CLI.

Configured various routes, directives for grid, pagination, conditional validations, templating, dynamic loading, lazy loading, Shadow DOM, NGRX store.   Created services to consume RESTAPI's using Component based architecture provided by Angular4. Made use of Angular UI grid to display tree-based grids   Converted the code base to use modern ECMAScript (ES6 / ES7) used Babel, Typescript and Web pack as a module loader and used along with Angular CLI.   Developed Angular views to hook up models to the DOM and synchronize data with server as a SPA.   Made use of Angular4 Observables to make HTTP Requests and leveraged its functions to loop around results.   Built-in AJAX HTTP Service with observables is used extensively for making quick and incremental updates to UI of the application.   Developed server-side proxy using Node JS (Hapi JS) for an event-driven, non-blocking I/O model to make an efficient application while keeping it lightweight.   Used NodeJS with HapiJS server to interact with the Rest services and database, hosted on multiple load balanced cloud instances.   Made use of authentication and authorization techniques available with HapiJS and used its tools like web packs to avoid boiler plate code. Utilized auto configurable documentation functionality provided by HapiJS to configure the Open API specification-Swagger using Hapi-swaggered and Hapi-swaggered-ui.   Karma test runner and Jasmine test framework tool are used to perform behavioral driven testing for Angular 4 code. Pushed new features live after passing tests implemented with Karma Test Runner.   Testing for Node JS is done using Mocha and chai.   Unit testing of web services is done using Spock framework by extending Groovy syntax.   Configured JBOSS Application server and used Oracle database to write complex PL/SQL queries.   Designed dynamic schema through MongoDB by incorporating rich queries syntaxes and achieved data (collections) distributed across multiple systems for horizontal scalability as data volumes increase through Automatic Sharding. Mongoose Connector is used for implementing ODM framework for Mongo DB.   For dependency

management, used the built in NPM package manager to configure the right versions of the packages.   Used Jenkins CI/CD tool for continuous integrating and deploying application on servers.   Configured Docker container service and build pipe line. Handled all the port mapping/binding things that happen with Dockers and Containers.   Used Rabbit MQ for transport of data in Queue in between components of the application.   Configured Kibana data visualization plugin for Elastic search, Log stash and created bar, line and scatter plots, pie charts and maps on top of large chunks of data.   Used Redis configuration with node server along with Cat box for caching the responses.   Implemented OAUTH2.0 authentication and authorization framework to provide the security to rest web services.   Installed, configured, and administered NoSQL database clusters in AWS Virtual Private Cloud Network for available data on cloud.   Implemented AWS solutions using E2C, 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.   Implemented Netflix Zuul API gateway for all the requests from the devices and websites by registering services to Eureka.   Configured Kubernetes in the application for automating deployment, scaling, and management of applications.   Configured DynaTrace to measure and monitor the performance of application.   Environment: HTML5, CSS3, Twitter Bootstrap, Angular4, Typescript, Angular CLI, NGRX, ECMA(ES6/ES7), Babel, Typescript, Web Pack, DOM, HTTP, Ajax, Node JS, Hapi JS, REST, Mongo DB, Oracle, Redis, Swagger, Karma, Jasmine, Mocha, Chai, Spock, Groovy, JBoss, ODM, Mongoose Connector, Kibana, Elastic Search, Logstach, NPM, Jenkins, Docker, Redis, AWS, Netflix Zuul, Eureka, Kubernetes, DynaTrace, Rabbit MQ, Cat box, AWS. Java Full Stack Developer Royal Caribbean cruise - Miami, FL April 2017 to November 2018 Description: Royal Caribbean International is known for driving innovation at sea. Each successive class of ships is an architectural marvel and pushes the envelope of what is possible on a cruise ship such as rock climbing, ice skating and surfing at sea to capture the imagination of families and adventurous vacationers alike. I involved with a team to develop the application which collects and stores the Customers data and display the different questions from the various dumps according to the specific module.   Responsibilities: Involved in developing the

Single Page Micro Service Oriented Architecture Application using the Mongo Database, React with Redux, Angular 2.0 and Node JS with Express JS.   Involved in developing the efficient UI using HTML5 for DOM representation and CSS3-LESS for good styling of the application.   Used the Bootstrap for the custom templating of the web pages.   Involved in developing custom directives for making reusable components in Angular 2.0.   Used Angular 2.0 to create views to hook up models to the DOM and synchronize data with server as a Single Page Applications.   Developed the objects including the components using Typescript, ng Modules, interfaces and classes using the Arrow function.   Implemented Angular JS observables and listeners (RxJS reactive libraries) to simplify the making asynchronous calls to backend REST API's.   ES6 generators were used for maintaining the concurrency between the components.   Multiple views were rendered by using the React JS and implemented JSX for adding XML syntax to the java script.   Implemented various screens for the front end using React.JS and used various predefined components from NPM (Node Package Manager) and redux library.   Implemented React JS components, Forms, Events, Keys, Router, and Animations.   Implemented React with Redux for state-management and Router for the navigation.   Implemented Redux to cut some complexity corners by using functional composition. Implemented Store-Redux for holding the whole state tree of application and Reducer-Redux for describing actions.   Reducers are implemented for specifying how the application state changes in response to the actions sent to the Store.   Implemented Web pack for bundling the modules and maintaining the libraries efficiently.   Angular testing was performed using the Protractor and React unit testing was performed using Mocha and Chai.   Developed the Rest Services on the server side using the Node JS with 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.   Developed Node.js Readable and Writable streams to process the data from the external source of the application.   Implemented modules into Node JS to coordinate with plans and necessities. Developing 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 Express JS which is fast, assertive, essential and moderate web framework of Node

JS that helps manage a sever and routes. Implemented Express JS which respond to HTTP Requests and dynamically render HTML Pages based on passing arguments to templates. Implemented Axios for calling the developed RESTful services that is for making the Ajax Calls. Implemented the Node Package Manager Build tool for managing the dependencies in the project. OpenID Connect (OIDC) was implemented on the top of OAuth 2.0 for authorizing the Rest Services and JWT for token based authentication. Used Spring IOC for the life cycle management of the bean and Java Transaction API (JTA) for the service layer. Developed some standalone Micro services using Spring Boot micro service architectural patterns. Implemented Spring Boot properties like actuators and health checkers to enhance micro service implementations. Mongo database was used for handling the unstructured data and used the Mongoose Connector for the connection. Mongo DB was used for heterogeneous collections of content types. Implemented Log4J for logging the information and Rabbit MQ for the service of Message broker. Implemented Docker for creating the custom Docker container images. Implemented the Docker for packaging the application in virtual container and then the contained application is deployed to Pivotal Cloud Foundry using CLI (Command Line Interface) tool where service is as Infrastructure. Jenkins was implemented for the Continuous delivery and for Continuous Integration. Used the Junit for unit testing and Selenium for the integral testing of the application. Implemented the Agile methodology for the efficient and effective completion of project. Environment: Redux, Angular 2.0, HTML5, CSS3-LESS, Bootstrap, Typescript, ng Modules, Arrow function, Angular JS observables, listeners, ES6, React JS, JSX, React.JS, NPM, React JS components, Forms, Events, Keys, Router, Animations, Store-Redux, Reducer-Redux, Web pack, Protractor, Mocha and Chai, Rest Services, Node JS with Express JS, Axios, OpenID Connect (OIDC), OAuth 2.0, JWT, Spring IOC, Java Transaction API (JTA), Spring Boot, Mongoose Connector, Mongo DB, Log4J, Rabbit MQ, Docker, Pivotal Cloud Foundry, Jenkins, Junit, Selenium and Agile methodology. Java Full stack Developer Edison, NJ July 2016 to April 2017 Description: Consolidated Edison, Inc., commonly known as con Edison or con Ed, is one of the largest investor-owned energy companies in the United States. I was involved in developing UI pages that helps customers retrieve online

quotes depending on service. Also worked on login field validations that help customers who are already members to retrieve their online account.   Responsibilities: UI Web pages were developed by using HTML5, CSS3 and JavaScript frameworks like AngularJS1.6   Angular JS 1.6 is used to create Controllers to handle events triggered by clients and send request to server. Angular Directives like Ng-Model, Ng-App, and Ng-repeat etc. were used to develop views and data binding in our application.   Worked with built-in Angular Filters and developed custom Angular Filters in application as per business requirements.   Angular scopes ($scopes) are used to glue View and controller to send data from controller to view.   Angular Routes were used to navigate through views in our application.   Angular Directives are used to develop customized components.   Rest API calls and data binding is done by using Angular JS   Spring MVC frame work is used to develop enterprise application and it helps to maintain loosely coupling. Spring IOC is used to inject dependencies and manage the life cycle of a bean.   Implemented Spring Annotations, Spring AOP to develop Business layer components.   Experienced with distributed transaction management with isolation levels with Spring JTA.   Exposed Stateless session beans as RESTful web services using JAX-RS.   Developed Restful web service using JaxRs with Jersey framework.   Swagger is used to create interactive documentation of Rest API and to automate the test scripts.   OAUTH 2.0 authentication frame work is used to provide security to Restful web services by generating access token in order to access protected resources.   Spring security is used for authentication and authorization and to provide protection against attacks like session fixation, clickjacking and cross site request forgery.   Spring batch programming is executed by using quartz scheduler.   Test cases are developed and implemented using Junit testing.   Developed our application using Service Oriented Architecture.   Apache camel framework is used for message-oriented middleware with rule-based routing and mediation engine.   Apache camel framework is used for concrete implementations of Enterprise Integration Patterns and connectivity to great variety of transports and API's   Messaging queues were developed using JMS frame work with IBM-MQ to communicate between applications by sending messages.   Maven is used as a build tool and SVN is used to maintain version control.   GULP is used to build Angular.js application and protractor for testing

Hibernate ORM is used to achieve persistence and make connections with MySQL Database. HQL is used to retrieve data from database. Hibernate EHcache is used at second level to boost the performance of application. Backend interfaces are developed using stored procedures and PL/SQL packages. Log4jand JIRA is used for logging and debugging. Pivotal Cloud foundry is used to build and deploy applications Environment: AngularJS 1.0, HTML5, CSS3, JSON,, Bootstrap, MAVEN, , Spring Boot, Spring IOC, Spring Batch, Spring cloud, , AWS, EHcache, Jax Rs, Jersey, Jax WS Apache CXF, Spring Batch, IBM-MQ,SVN,GULP,Protractor,1 Pivotal cloud foundry, Log4j, JIRA Sr. Java Developer Delta Airlines - Atlanta, GA September 2015 to April 2017 Description: In Delta Airlines e-commerce project my role is to facilitate the enhanced features to book or cancel tickets and to do check-ins through smart devices which will help Delta Airlines customers to make bookings, check-ins by providing them various benefits such as Ease of access, Convenience, make bookings faster and saving the check -in time, Reduce the use of sales assistant. Responsibilities: Developed Monolithic application based on HTML5, CSS3 (LESS), JQuery, Bootstrap, AJAX for building dynamic web pages. Create the frontend JavaScript components using Backbone JS and frontend templates and used Underscores hard coded dependency in combination with Handlebar JS for frontend templating. Developing standalone forms for cross-browser and cross-Key Accomplishment as Single Page Applications (SPA) with the use of Backbone JS. Data-structuring (models and collections) and user interfaces are developed using Backbone JS to maintain the synchronization and to achieve MVC pattern. Used Ajax for sending and retrieving data from server by giving URL to get JSON data for models and to populate models from the server. Used QUnit testing framework to perform unit test on Backbone JS code and developed validations using custom generic JavaScript functions. Load testing is done using JMeter for Ajax (JSF/Rich Faces) application. Produced SOAP web services using Metro JAX-WSRI for building the business application layer. Created WSDL (Web Services Description Language) document for client application web services. Designed and implemented Business Delegate, Session Facade and DTO Design Patterns. Used Hibernate along with spring for implementing Spring IOC, Spring AOP and Spring ORM for back end tiers. Developed beans from

the existing parent beans using Spring Inheritance and extensively used various Spring Framework modules like MVC, JDBC Templates.   Extensively used Hibernate mappings, HQL, EHCache, Query, Lazy loading Transactions and Locking's   Implemented concepts of Hibernate Transaction Management, Hibernate Batch Transactions, and cache.   DB2 database used to load the application data and written PL/SQL procedures, complex queries to perform various data operations.   Experienced in implementation of JMS to exchange information over reliable channel in an asynchronous way by using Rabbit MQ as a message queue.   Implemented Spring Security for providing authentication and authorization for web services.   Focused on Test Driven Development (TDD), thereby creating detailed Junit along with Mockito for every single functionality before writing the functionality.   Responsible for Continuous Integration (CI) and Continuous Delivery (CD) process implementation using Jenkins along with Shell scripts to automate routine jobs.   Used Maven to build the application.   Performed logging using Log4j.   Environment: HTML5, CSS3(LESS), JQuery, Bootstrap, Ajax, Backbone JS, Underscore JS, Handlebar JS, Native JS, Qunit, JMeter, Junit, JAX-WS, SOAP, WSDL, JSON, Spring IOC, Spring AOP, Spring ORM, Spring MVC, JDBC, Hibernate, HQL, EHCache, Hibernate Transaction Management, Hibernate Batch Transactions, Jenkins, Maven, Log4J , DB2, PL/SQL, JMS, Rabbit MQ. Jr. Java Developer UPS July 2012 to August 2015 Description: UPS is the world's largest package delivery company. Using advanced technology, UPS provides a powerful competitive advantage that can help you earn repeat customers and grow your business. This is a web application that was built to satisfy the need for developing a micro service architecture that brings in new feature and enhances the process of creating and retrieving report and existing reference data that was collected from the user.   Responsibilities: Involved in developing the Monolithic architecture application using EJB, Struts, jQuery, JSP by following the Waterfall Methodology.   Implemented the EJB and Struts Framework for developing the Form and Action Classes.   Prepared EJB deployment descriptors using XML and Used JAXB components for transferring the objects between the application and the database.   Dynamics from generation auto completion of forms and user-validation functionalities using jQuery.   Configured Struts-config.xml, Tiles-def.xml and properties file provided by Struts

Framework for the implemented modules.   Implemented connectivity to databases using JDBC from servlets and JSP through Enterprise Java Beans (EJB).   Developed Web pages using JSP and implemented Struts Validation framework for form input validations.   Build PL/SQL functions, stored procedures, views with JDBC data source with connection pooling enabled.   Implemented JavaScript Library for simplifying the HTML document traversing, and event handling.   The application is deployed to JBoss Application Server.   Environment: EJB, Struts, XML, jQuery, JDBC, Servlets, JSP, PL/SQL, HTML, Java Script and JBoss. Education Bachelor's Skills Eclipse, Ejb, J2ee, Java, Hibernate, HTML5, CSS3, Twitter Bootstrap, Angular4, Typescript, Angular CLI, NGRX, ECMA(ES6/ES7), Babel, Typescript, Web Pack, DOM, HTTP, Ajax, Node JS, Hapi JS, REST, Mongo DB, Oracle, Redis, Swagger Additional Information TECHNICAL SKILLS: Programming Languages C, C++, Java, PL/SQL, .Net Web Technologies HTML5, SASS, CSS3, JavaScript, PHP, JSON, JQUERY, AJAX, XML, SOAP, REST, WSDL, JAXP, JAXB JavaScript Frameworks JQuery, Angular 2.0/4.0, React JS, Express.js, Backbone.js, Node.js, Ember.js, Handlebars.js J2EE Technologies & frameworks Servlets, EJB, JSF, JSP, JDBC, REST, SOAP, WSDL, JAXB, JAXP, JPA, JMS, Spring3.x/4.x/5.x, Hibernate, Spring Data, Spring Cloud, Dump Analysis, Spring MVC, Spring Boot, Spring Batch.   Web service frameworks JAX-WS, JAX-RS, Apache CXF, Axis, Jersey Testing technologies/tools JUnit, JMeter, Selenium, Cucumber, SPock, Mockito, TestNg, Karma, Jasmine, Protractor, Mocha, Chai Databases MySQL, Oracle, SQL Server, NoSQL (Mongo, Cassandra) Messaging Services JMS, Active MQ, Rabbit MQ, IBM MQ, Apache Kafka IDE's Eclipse, Visual Studio, IntelliJ, Spring Tool Suite, Sublime Text, WinSCP, Putty Cloud Technologies & DevOps Tools AWS, PCF, EC2, EBS, S3, Docker, ECS, EBS, EKS, SQS, RDS, S3, Glacier, Lambda, Kubernetes Build Tools ANT, Maven 3.0, Gradle, NPM, Grunt, Gulp, Bower, Jenkins Testing Tools Junit, TestNG, Mockito, Jasmine, Karma, Mocha, Chai, Spock, Protractor Version Control CVS, GIT, SVN Software Methodologies Agile, Scrum, TDD, Waterfall Project Management Tools JIRA, Rally, Microsoft Project Operating System Windows, Linux, Unix, MS-DOS

Name: Helen Odom

Email: vcarter@example.net

Phone: +1-339-745-1310x075