**Aashish Sapkota**

**Full Stack Software Developer**

**Email: Satya@takeo.ai / Contact No.: 347-903-1896‬ / Location: Minneapolis, MN 55409   
LinkedIn**: https://www.linkedin.com/in/sapkota-aashish/

**SUMMARY**

Professional Full Stack Software Developer with **7+ years** of strong experience in building web applications using **React**, **Redux, Flux, Babel, ES6**, **Web-Pack,** and Java technologies for Application development. For the build of **JSX and ES2015(ES6) used Babel, webpack**. Expertise in Various Spring Modules **(Spring MVC, DAO, AOP, Spring LDAP, Spring Scheduler)** along with Hibernate as the back and skilled in developing **SPA** (Single Page Application) with React features such as**class-based and functional components, props, states, routing, React Hooks, and lifecycle hooks.**

**Technical Proficiency**

|  |  |
| --- | --- |
| **Languages:** | **JavaScript** |
| **Web Technologies:** | HTML/HTML5, CSS2/CSS3, XML, JavaScript, AJAX, jQuery, JSON, Bootstrap |
| **Frameworks/Tools:** | React.js, Bootstrap, Node.js |
| **IDE** | Sublime Editor, Visual Code Studio, Atom |
| **Source code control tools:** | SVN, GitHub |
| **Bug tracking Tools:** | JIRA |
| **Operating System:** | Windows, Linux, OS X |

**Experiences**

**UnitedHealth Group, Minneapolis-St. Paul, MN** **Jun 2021 - Present**

**Full Stack Software Developer**

**Responsibilities:**

* Developed all the UI pages **using HTML5, CSS3, Bootstrap, JavaScript, React.JS, Typescript, AJAX, JSON,** and **Restful API and** developed backend components in **Spring Boot, Struts, REST Webservices, Spring Batch, Spring Data, Hibernate, JPA, and Unix.**
* Involved in the requirement gathering phase of **SDLC** with Business Analyst, Product Owner and worked in **Agile**-**Scrum** methodology.
* Worked in using **React JS components, Forms, Events, Keys, Routers, Animations,** and **Flux concepts**.
* Working on **React** **JS** **Virtual** **DOM** and React views, rendering using components that contain additional components.
* Developed Single Page Applications (**SPAs**) using **React**-**Router** and enabled navigation from one view to the next as clients perform application tasks.
* Implemented Jasmine framework and karma server to write and test the unit test cases for code.
* Used **AJAX** **calls** and requested **JSON** information through **REST** **APIs** to fetch data and secure client information from the server **asynchronously**.
* Involved in Microservices development of RESTful Web Services, using Spring Boot, Spring Cloud, Spring, Spring Security, and PCF
* Created **RESTful** web services using **JAX-RS**, **JSON**, **Jersey**, and **Spring** **Rest** (**Web** **MVC**) along with unit test cases for **REST** endpoints and **API** Documentation (using **Swagger**)
* Tested the Restful web services calls for **API** using **POST, PUT, DELETE**, and **GET**methods.
* Used **Jenkins** for continuous integration in which all development work is integrated and involved in deploying the application user to the **AWS** **environment**.
* Used version control system tools such as **Git** and **SVN** to manage the code repository.
* Handled **cross-browser** **compatibility** in browsers like safari, Internet explorer, Firefox, and chrome.
* Used **Docker** for automating the provisioning of environments and deployments to achieve Continuous Delivery

**Medtronic, Minneapolis-St. Paul, MN** **Oct 2018– May 2021**

**Frontend Developer**

**Responsibilities:**

* Designed the Front-end applications, as well for user interactive (UI) web pages using web technologies like **HTML5, CSS3, Es6 JavaScript, React.JS and React-Native, Redux Saga**.
* Developed the Front-End and User Interface of the application using **Javascript, ES6, React,** and **Redux.**
* Involved to implement various **screens** for the front end using **React** **JS** and used various predefined components from **NPM** (Node Package Manager) and Redux library
* Development of single-page application using **React**.JS, **Redux**, **HTML**. Leveraged Redux for state management using Actions, Dispatchers, Providers and Store
* Applied **redux** **middleware** using redux-promise, **redux-thunk**, and redux-saga.
* Used **Babel** to compile **JSX, ES6**, and follow **JSHint** standards for high code quality.
* Configured **redux** **store**, **reducers**, and actions. Used sagas for handling **middleware** and **'fetch** **api'** for rest calls.
* Used **React**.js and **redux** library functions for the logical implementation part at client side for all the application.
* Used **Grunt**, **Web**-**packer**, Yarn to perform bundling, magnification, compilation and ESLint for code cleaning.
* Used **React**.js library functions for the logical implementation part at client side for all the application.
* Used web scripting languages like **NodeJS** and **Yarn** for interaction between client side and server-side technologies.
* Used **Git** for version and source control.
* Worked on **Jira** **Tickets** and helped multiple teams in closing out their tickets.

**US Bank, Minneapolis-St. Paul, MN** **Feb 2015 - Sept 2018**

**Software Developer**

**Responsibilities:**

* Developed and maintained the front end using **HTML**, **CSS**, and **Ajax**, supported with Java backend and related frameworks.
* Used **Hibernate**, object/relational-mapping (ORM) solution, the technique of mapping data representation from **MVC** model to **Oracle** Relational data model with a **SQL** schema.
* Used **Spring** **IOC** to inject the services and their dependencies in the dependency injection mechanism. Designed and developed business components using **Spring** **AOP**, **Spring** **IOC**, and **Spring** **Annotations**.
* Used **Spring** **JDBC** Template and Spring DAO’s Support classes for JDBC.
* Used **JavaScript** for client-side validations and **AJAX** to create interactive front-end GUI.
* Used **JSON** for data exchange between application modules along with XML.

**Education**

Bachelor’s in science, Computer Science  
St. Cloud State University

**References Available Upon Request**