Woody Wang

587-968-4908
woody.career@gmail.com
Toronto, Canada

Technical Highlights

- Solid knowledge of complete **Software Development Life** Cycle (SDLF), **unit test,** and **agile** development
- Skilled with (language) **Java**, **Python**, **JavaScript**, **C/C++,** (framework) **React**, **Spring Boot,** (dev tools) **Jira**
- Familiar with **Linux/Unix**, version control (**Git**), **RabbitMQ**, constructing **GraphQL** and **RESTful** APIs
- Demonstrated skills with relational database like **MySQL**, and non-relational database like **MongoDB**
- Familiar with deployment environments like **AWS**, and deployment tools, like **Docker**, **K8S**, and **CI/CD** tools

Education Background

MSc., Computer Science	**Starts Jan 2022**
Georgia Institute of Technology	Atlanta, USA
BSc., Computing Information Systems (GPA 4.0/4.0, in progress)	**Ends Dec 2021**
Athabasca University	Edmonton, Canada
MSc., Chemical Engineering (GPA 3.8/4.0)	**Sep 2012-Nov 2014**
University of Alberta	Edmonton, Canada

Work Experience

Full Stack Developer	**Sep 2020 – Jul 2021**
EyesHigh Edu. & Tech	Toronto, Canada

Worked on developing a web app, which helped jobseekers efficiently compose and revise resumes by intelligently suggesting. The web app allows the user to either import original resumes or make a new resume from scratch, along with the section guidance for collecting resume information. Then the web app provides revision suggestions which users may refer to.

- Confirmed requirements from the front end and developed **REST API** with **Java**
- Worked in an **agile dev** team and assisted in organizing daily stand-up meetings
- Illustrated user stories with clear ticket scopes and definition of done through **Jira** documents
- Identified data relations and developed the job and user database with **AWS RDS** (MySQL)
- Configured **MongoDB clusters** to store the information of 600 K job posts crawled from the internet
- Reviewed peers' code, closed pull requests and followed version control policy with **Git** and **GitHub**
- Documented product development requirements, and user stories with **Jira** and **Confluence**
- Developed UI with **React**, **Redux, HTML, CSS, Ajax,** closely following the design from the UX designer
- Configured, built, and deployed images with **GitHub CI/CD pipeline**, **Docker, K8S,** and **AWS EKS**

Back End Developer	**Jun 2019 – Jan 2020**
EyesHigh Edu. & Tech	Edmonton, Canada

Single-handed and responsible for collecting requirements and developing a plug-in. Internal instructors mainly used the plugin to log student's information, streamline coaching services for students, track the progress, assign coaching sessions to instructors, visualize the incurred cost, and monitor the coaching efficiency.

- Built and decorate tables, modals, and popups with **jQuery** and free **JavaScript** libraries like **DataTables**
- Incorporated the free library of **PHPMailer** and **TCPDF** to email receipts (PDF) and account info to users
- Connected job database (MySQL) with the **MongoDB** via the **PHP extension** to fetch job posts information
- Integrated the **Google Drive library** for users to conveniently upload/download google-files in the portal
- Secured the login and register gateway via inserting **Google reCAPTCHA** and reconfigure plug-in
- Visualized the coaching performance KPI via free the **JS library of Google chart**
- Asynchronized HTTP requests and responses (like form submission) with **jQuery**, **Ajax**, **PHP,** and **SQL**
- Analyzed data relations and created a database with **MySQL (phpMyAdmin**) hosted by Godaddy

Project Experience

Backend development of car website	Udacity, Jun 2019

- Developed **REST APIs** for services: vehicle list, pricing, and location information
- Wrote CRUD operations to store/retrieve vehicle info and implemented an HTTP client to retrieve address
- Integrated vehicle API with pricing API and used **Swagger** to create documentation for the APIs
- Populate HTML templated with **Spring's MVC** data model and conducted the **unit test**, error handling

Reference Available Upon Request