Brian Wan

(415) 816-7279 bcwan8@gmail.com linkedin.com/in/bcwan github.com/bcwan San Francisco / Bay Area

Skills

JavaScript, React.js, Redux.js, Ruby, Ruby on Rails, HTML5, CSS3, Git, PostgreSQL, jQuery, Java, C#, Recharts.js, JSON, AJAX, Node.js, MongoDB, Express.js, Postman, API, SQL Server, DB2, Oracle

Experience

County of Alameda
Software Engineer (Programmer I) Aug 2018 - Aug 2019

- Engineered with a colleague a chatbox utilizing Chatbot API, Microsoft Azure services, and Angular components, allowing users to ask county-related questions
- Implemented efficient SQL queries on 10 million data rows hosted on Oracle and DB2 databases to create Excel spreadsheets for social workers to track caseloads
- Upgraded old Cognos visualization software to modern Microsoft PowerBI reports, hosted by SQL Server, allowing better visualizations
- Monitored ticketing softwares, Ivanti and HEAT, to document and keep records of software troubleshooting issues and solutions
- Mentored a summer intern on database querying, teaching him how to create data visualizations and SQL skills by supervising his project

County of Alameda
Software Engineering Intern Jun 2016 - Aug 2017

- Revamped a web form utilizing Bootstrap, JavaScript, jQuery to create a more dynamic and modern web page for General Service Agency
- Upgraded a web form utilizing JavaScript, jQuery, and Bootstrap to create a modern web page for County Administrator's Office
- Utilized AJAX API calls to grab JSON data from county servers to populate web forms with updated user information
- Populated a PDF form utilizing DocuSign API that transferred information from a web form, allowing users and the county to keep electronic records

Education

App Academy Feb 2020 - Jun 2020
Immersive software development course with a focus on full-stack web development.

University of California, Santa Cruz
*Bachelor's Degree in **Computer Science*** Oct 2014 - Aug 2018

Projects

StackOverflow Jr. (JavaScript, React / Redux, Ruby / Rails, HTML, CSS, PostgreSQL, JSON, Bootstrap) Live Site | Github
Inspired new website based on Stackoverflow where users can post and update questions and answers

- Designed React component questions/answers page utilizing Redux store and JavaScript to update the user interface with live data and reduce expensive backend database calls
- Implemented user authentication using Ruby / Rails and Bcrypt library to allow account creation and give selected users the authority to delete, update, and create input functionality
- Constructed AJAX API calls with JBuilder and PostgreSQL database to allow the backend Rails controller to send JSON user data back to the frontend, updating UI with requested data, ensuring the front-end application has updated data.
- Incorporated a responsive UI website using HTML, CSS, and Bootstrap libraries to create a smooth, comfortable, and modern web design and enhance the user experience

F.L.A.M.E. (JavaScript, React / Redux, NPM, Node, Recharts.js, MongoDB, Express.js, Axios) Live Site | Github
A financial literacy website where users can track, budget, and formulate future retirement contributions

- Engineered backend with MongoDB and Express.js and encrypting with JSON web tokens to pass user data securely to front end UI
- Collaborated in designing a JavaScript data parsing algorithm utilizing hashtable data structure to concurrently populate Recharts.js financial graphs and reduce load time
- Incorporated user form and user transactions table utilizing React modals/components and Redux frontend storage to allow users to input, update, and delete data
- Utilized toast components from NPM libraries coded into Axios backend API functions that notified users of errors and successful data validations to database

2560 (JavaScript, HTML5, CSS) Live Site | Github
A number grid game inspired by the popular 2048 game in which the user utilizes strategies to combine blocks in creating a 2560 block

- Engineered a dynamic 2D game board through JavaScript and HTML DOM that stores block pieces and organize data of each block
- Leveraged debugging process using JavaScript debugger and Google Chrome console to catch bugs and runtime errors
- Implemented board pieces and game logic utilizing JavaScript Object Orientated Programming fundamentals, allowing inheritance of functions across files and smooth data access
- Designed sliding animations feature for blocks with CSS and JavaScript to allow visual transitions and movements across the game board