

Bradon Siegle

587-777-8088 | bradonsiegle@gmail.com | [linkedin.com/in/bradon-siegle](https://www.linkedin.com/in/bradon-siegle) | github.com/bradonsiegle

TECHNICAL SKILLS

Languages: HTML5, CSS3/SASS, JavaScript/TypeScript, Java, C++, SQL
Frameworks: React, Redux, Node.js, Next.js, Express.js, Strapi, MongoDB, TailwindCSS, MaterialUI
Developer Tools: Docker, AWS, Postman, Storybook, Chromatic, Jest, Git, Unix

EDUCATION

Bachelor of Science in Computer Science

2022

Vancouver Island University

Nanaimo, BC

GPA: 3.5/4.0, Dean's List 6 Semesters

PROJECTS

Job Tracking App | *React, Express.js, Node.js, MongoDB*

- Designed and developed a full-stack React app for web-based job application organization and tracking
- Created a REST API with Express and Node to perform CRUD operations and JWT user authentication
- Deployed a MongoDB Atlas cluster as a cloud-based solution to persist and manage user and application data
- Used Postman to develop test cases for the API endpoints, and automated the API test suite using Newman
- Deployed both the client and server as a monorepo using Heroku CLI to host the application on Heroku

Weather Forecast App | *React, JavaScript, TailwindCSS, Vite*

- Developed and designed a weather forecast web application using React and Vanilla JavaScript
- Used TailwindCSS and a mobile-first design paradigm to create a responsive user experience for all device sizes
- Scaffolded the app with Vite JS to take advantage of hot module replacement and improved build speed
- Interfaced with OpenWeather and GeoDB Cities web APIs to send and retrieve real-time weather data
- Implemented a CI/CD pipeline using GitHub actions to host the website on GitHub Pages

Team Messaging App | *React, Express.js, Node.js, MongoDB, SocketIO*

- Created a full-stack web application to support real-time team communication via text and voice messaging
- Developed both client and server-side logic for user login, registration, direct messaging, and contact systems
- Implemented web socket and API endpoints with Node.js for bi-directional client and server interfacing
- Securely designed a data storage and authentication pipeline to store and encrypt sensitive user data
- Developed an automated test suite to test both the front-end and API endpoints with Jest and Postman

Film Statistics Website | *JavaScript, BulmaCSS*

- Developed and designed a website to search and compare statistics of film and box office releases using JavaScript
- Designed a search system to send API requests and interface with OMBD API based on user queries
- Implemented comparison algorithms to display statistics and determine film ranking based on API responses
- Configured continuous deployment of the website on Netlify using Netlify CLI

Culinary Blog Website | *React, Typescript, Next.js, Strapi, mySQL*

- Designed and developed a full-stack website for sharing cooking recipes, using Next.js and Strapi for headless CMS
- Created a library of reusable React components using Storybook and component-driven development methodology
- Utilized Strapi server-side rendering and static-site generation to optimize the front-end performance and SEO
- Refactored the default SQLite database to use mySQL with SQL Server on Amazon RDS Free Tier
- Developed a suite of unit tests to thoroughly test the user functionality using Jest and React Testing Library
- Implemented a CI/CD pipeline with GitHub actions to automate the application build, testing, and deployment

Billing Management System | *Java, Swing, JDBC, MySQL*

- Developed a window-based program to perform small-business billing and inventory management tasks, using Java
- Implemented a user-friendly GUI using Java Swing and reusable components from the AWT API
- Interfaced with JDBC API to connect the system to a MySQL database to execute queries and persist data

FTP Client and Server | *C++*

- Led a team of 4 students to design and develop a command-line file-transferring program using C++
- Developed a client and server to provide remote access using TCP and UNIX socket programming
- Implemented FTP functionality and commands to transfer files in adherence to the standards outlined in RFC 959
- Automated program compilation and testing using GoogleTest with CMake