BRIAN LOCKERBIE

Toronto, ON M4Y 1W4| 647.919.5145
brian.lockerbie@gmail.com | https://www.linkedin.com/in/brian-lockerbie/ | https://github.com/brianlockerbie
https://brianlockerbie.com

CAREER ASPIRATION

Junior Full Stack Developer • Full Stack JavaScript Developer • Junior Software Developer

- ***An organized, meticulous self-starting professional*** with 1+ years of experience with multiple languages while using creativity in solving problems and designing beautiful web applications.
- ***A diversely skilled Junior Full Stack Developer with knowledge across most technologies,*** including React, Express, Node.js, Angular and Git.
- ***A focused and eager to learn problem solver*** that loves to learn and excels in any up and coming new technologies.
- ***A recent Full Stack JavaScript graduate,*** you can expect to see my passion for completing projects and deadlines, my teamwork strengths, as well as my superior written and verbal communication skills. These characteristics make me a stand out candidate for this role.

EDUCATION

Build Apps with ReactJS: The Complete Course
Udemy
September 2020

Full Stack JavaScript Tech Degree
Treehouse
This certifies proficiency in the fundamentals of JavaScript, Node, React, Express, and SQL
June 2019 – May 2020

Google Analytics for Beginners
Google Analytics Academy
Issued June 2020

Bachelor of Business Administration – Accounting
George Brown College, Toronto, ON
Completed 1st year

AREAS OF EXPERTISE

- JavaScript
- ReactJS
- Express
- Node.js
- Angular
- OOP
- Git & GitHub
- jQuery
- MySQL
- REST APIs
- PHP
- Asynchronous JS
- Agile Methodologies
- HTML
- CSS

FULL STACK DEVELOPMENT EXPERIENCE

Treehouse, Toronto, ON **04.2020**

Full-Stack-APP with React and a REST API

I created a full-stack application that provides a way for users to administer a school database containing information about courses. Users can interact with the database by retrieving a list of courses, viewing details of a specific course, as well as creating, updating, and deleting courses from the database. Users must create an account and password in order to log in to make changes to the database.

- ***Technologies & Tools:*** Express, Sequelize ORM, DB Browser for SQLite, Postman, Express-validator, bcryptjs, basic-auth, CORS, React Context API, react-router-dom, create-react-app, react-markdown.

Treehouse, Toronto, ON **02.2020**

SQL-Library-Manager

In this project, I was given HTML designs and an existing SQLite database. I was required to implement a dynamic website using JavaScript, Node.js, Express, Pug, and the SQL ORM Sequelize. The application permits a user to create their own personalized list to add, update, and delete books from an SQLite database.

- ***Technologies & Tools:*** Sequelize, SQLite database, Node.js, Express.js, Pug template engine, npm.

Treehouse, Toronto, ON **01.2020**

React Gallery Application

An image gallery app built using the React library. In this project, JSX is used to write markup-like syntax directly in the JS files and manage state in a container component that passes data down to reusable stateless components.

- ***Technologies & Tools:*** create-react-app, JSX, React Router, Fetch API, Flickr API.

Treehouse, Toronto, ON **09.2019**

OOP Game Show Application

I created a phrase hunter game similar to the popular game 'hangman'. The user guesses what letters are within the phrase by clicking on that key. If the user guesses incorrectly, you lose a life. If the user guesses all the letters correctly you win the game. Uses Object-Oriented Programming & DOM manipulation

- ***Technologies & Tools:*** OOP, JavaScript classes, Array iteration methods & CSS variables.

Treehouse, Toronto, ON **06.2019**

Random Quote Generator

Single-paged app that displays a quotation at random each time the button is clicked. Each time the button is clicked a random color is generated as well. Uses knowledge of basic JavaScript syntax, variables, loops, conditionals and object literals.

- ***Technologies & Tools:*** JavaScript syntax, variables, loops, conditionals and object literals.