

Matthew Francis | 289-772-7465 | Burlington, ON

<https://mattfrancis.vercel.app> | mattfrancis888@gmail.com | <https://github.com/mattfrancis888> | [Linkedin](#)

Technical Skills

Frontend: React, React-Spring, Redux, Typescript, React- Testing-Library, Enzyme, Jest, HTML, SCSS, BEM, CSS, Bootstrap, JS, Figma.

Backend: Authentication, Database Normalization, Docker, CI/CD, Node.js, Express.js, PostgreSQL, MongoDB, GraphQL.

Tools: Git, AWS, Gitlab, Azure DevOps, TravisCI, GitHub Actions, Shopify, Microsoft Teams, Slack, Heroku, Vercel .

Work Experience

Habanero | Frontend Software Developer

Sep 2021 – Jan 2022

- Collaborated with design, operations and engineering team to develop web application for over a dozen of clients with React, React-Spring, Typescript, HTML, SCSS, CSS, JS.
- Created customized Halloween animations for over a dozen of subscription clients.
- Involved in agile methodology that included biweekly scrum, backlog refinement, check-ins, QA, and retrospective.
- Used Git and Azure DevOps to conduct code reviews, raised pull requests, pushed code, and operate a CI/CD pipeline,

Badal | Full-Stack Software Developer

May 2021 – Sep 2021

- Collaborated with design, operations and engineering team to develop a Shopify application that automates sales for Shopify stores with over hundreds of products and users with React, Redux, Typescript, PostgreSQL, GraphQL, Node.js, Express.js, Docker, HTML, SCSS, CSS, JS.
- Involved in agile methodology that included biweekly sprints, backlog refinement and daily standups.
- Used Git and Gitlab to conduct code reviews, raised pull requests, pushed code, and operate a CI/CD pipeline,

Education

Honours Bachelor of Science, Computer Science | Wilfrid Laurier University, Waterloo

Sep 2018 - Present

Extracurricular Experience

Laurier's Computer Science Club | Lead Software Developer

Sep 2020 – May 2021

- Guided 15 team members in creating a web application for the club with React, HTML, SCSS, CSS, JS.
- Held weekly meetings to explain technology stack and version control with Git; used agile methodology that included biweekly sprints and backlog refinement; Used Git and Github actions to conduct code reviews, raised pull requests, pushed code, and operate a CI/CD pipeline,

Projects

Blizzard ([Link](#))

- Replicated Blizzard's website, known for publishing games like Call of Duty and World of Warcraft. Users can view pages for each game. Used React Spring to create physics-based animations like accordion, image parallax, and card flip.
- Built with React, React Spring, Redux, Typescript; tested with React Testing Library.

Steam ([Link](#))

- Replicated Steam's website, the largest video game retailer for PC games. Users can browse for games and discounted games; add games to their cart; create, edit, delete reviews for games; authenticate; upload profile picture and edit their profile.
- Data is normalized to BCNF and is stored in PostgreSQL. Authentication is done with cookies, JWTs and refresh tokens. Used security practices against XSS / CSRF attacks.
- Built with React, Redux, Express.JS, Typescript; tested with React Testing Library.

Kijiji ([Link](#))

- Replicated Kijiji's website. Users can create, edit, and delete their listing; upload a display picture to their listings, search listings based on category and location; view personally created listings; authenticate.
- Data is normalized to BCNF and is stored in PostgreSQL. Authentication is done with cookies, JWTs and refresh tokens. Used security practices against XSS / CSRF attacks.
- Built with React, Redux, Express.JS, Typescript; tested with React Testing Library.