

Cameron St. Amant

Full Stack Software Engineer

MERN stack specialist, continuous learner

✉ cameronstamant@protonmail.com

📍 Orem, UT

in [linkedin.com/in/cameron-stamant](https://www.linkedin.com/in/cameron-stamant)

📞 385-498-4294

🌐 cameronstamant.com

🐙 github.com/CameronStAmant

WORK EXPERIENCE

Level 2 Support Engineer Instructure

05/2018 - 09/2020

Pleasant Grove, Utah

Achievements/Tasks

- Resolve customer and coworker technical issues related to the Canvas LMS platform
- Mentor others in using APIs, troubleshooting browser issues, and identifying solutions using internal and external channels
- Lead education efforts for the Canvas LMS platform
- Call APIs to test and reproduce bugs
- Write Jira support tickets for the engineering team, complete with replication steps and environmental information
- Utilize APIs to test and replicate defects

Level 1 Support Engineer Instructure

11/2016 - 05/2018

Pleasant Grove, Utah

Achievements/Tasks

- Independently search for and select new software for internal training
- Maintain a positive attitude when dealing with unhappy customers
- Be a team player with excellent communication with both local and remote coworkers
- Prioritize tasks based on severity

EDUCATION

Bachelor of Science in Business Management

Brigham Young University – Idaho

07/2011 - 12/2016

SKILLS

JavaScript

React

HTML5

CSS3

Sass

Pug

Tailwind CSS

Node

Express

MongoDB

Mongoose ODM

Firebase

Webpack

Jest

Git

npm

GitHub

REST APIs

Unit Testing

PERSONAL PROJECTS

Blog [🔗](#)

- Protect routes based on role to only allow permitted users to use certain APIs
- Built responsive UI using Tailwind CSS to give the best device-agnostic experience
- Manage user data within MongoDB using Mongoose ODM
- Communicate with two frontends via RESTful APIs

Battleship [🔗](#)

- Implement device-dependent input options such as touch on mobile and mouse on computer
- Leverage source maps to troubleshoot any compilation errors
- Stay bug free by using Test Driven Development

Members Only [🔗](#)

- Hash passwords using bcrypt.js
- Authenticate users and maintain the session using PassportJS
- Develop styling with CSS preprocessor

Hidden States [🔗](#)

- Utilize Firebase to manage leaderboard data
- Add a dropdown list that appears when an item is clicked
- Handle shared state data by lifting it up to the closest common ancestor

Shopping Cart [🔗](#)

- Consume an API that pulls product images
- Implement route handling with React Router
- Use hooks and function components for better legibility and maintainability

Odin Clone [🔗](#)

- Authenticate users using Firebase's Google and anonymous Sign-In
- Update the database asynchronously
- Render content conditionally