JAKE THRASHER

(HE/HIM)

mjakethrasher@gmail.com

(503) 847-8016

/m-jake-thrasher

github.com/jakethrasher

www.jakethrasher.com

[ABOUT]

Software engineer with a degree in Biology. Driven by curiosity and problem solving. Hard-working, and dedicated to learning new tools and techniques to further my understanding of software development. I'm also passionate about self-expression, equality, and community.

[SKILLS]

Languages

JavaScript
TypeScript
HTML
CSS

Libraries & FrameWorks

React
Node
Express
Material-UI

Testing

Jest
Mocha/Chai
QUnit
React Testing Library

Dev Tools

GitHub
Google Cloud Services
Postman
Heroku
Sequelize
pgAdmin
VS Code

Database

PostgreSQL
Firebase
Google Cloud Storage

[EDUCATION]

Alchemy Code Lab

800+hr rigorous full-stack software development training

University Of Alabama at Birmingham

B.S. Biology
Minor in Chemistry

[EXPERIENCE]

Software Engineer PayClearly

I improved app performance by simplifying complex frontend logic and implementing backend search and filtering patterns. I made major contributions to our internal customer service tool, unlocking customer spend and improving quality of life for customer service team. I created and updated many full-stack features, including adding a 1Password integration to enhance our ability to make payments. Gained experience and knowledge working with Google cloud services, such as Firebase realtime database, cloud storage, emulators, and cloud functions.

[PROJECTS]

Socket Jockey [Deployed Site] [GitHub]

Node.js | Express| React | Socket.io | Matter.js | Tone.js | p5 | Heroku

Socket Jockey is a collaborative audiovisual app that uses colliding objects to generate abstract soundscapes. The frontend was built in React and utilizes the Matter.js physics engine. I worked on the custom hook that handles a lot of the apps functionality and used Material-UI to build modals that conditionally render when users enter a room. I set up the Node server and used Socket.io to enable real-time collaboration with multiple users. I also wrote the logic that assigns every three users to a random room for anonymous collaboration and added the ability to create custom rooms if desired. I was also able to incorporate some of my musical background to program synth sounds using the Web Audio framework Tone.js.

AwesomeBot [GitHub]

Node.js | Express | PostgreSQL | Slack-API/SDK | Cheerio.js | Heroku

Slack app created for the Alchemy Code Lab workspace that allows students to access course assignments, virtual classrooms, and schedule meetings with custom slack commands. The entire app was built in Node, using Express to create a REST api. I wrote the endpoints that fetched course assignment data from the Canvas LMS api and used the Node Slack SDK to format custom interactive modals. I also added a security layer to the server by writing middleware that uses HMAC from Nodes Crypto module to verify the authenticity of the requests from slack.

Clear Skies [Deployed Site] | [GitHub]

Node.js | Express | PostgreSQL | Jest | React | React-Bootstrap | Heroku

Built remotely with three other developers, this full-stack app helps users locate celestial objects from their current location. On the backend, Node is used to service a PostgreSQL database and CRUD routes were written using Express. I used node-postgres to interface with PostgreSQL and set up the database from the CLI. Using TDD, I wrote CRUD routes for handling user account information and wrote some endpoints for fetching data from the Nasa API. On the frontend, I worked with multiple component libraries, included React-Bootstrap to build the UI.

NutriZone [Deployed Site] | [GitHub]

JavaScript | CanvasJS| HTML | CSS | Netlify

This highly collaborative project was written entirely in vanilla javascript. It serves as a nutrition tracking app that helps users meet daily caloric goals. We used local storage to track user info and CanvasJs charting library to display the data. I wrote the logic for filtering and sorting the food data based on the users preferred diet type. I also added some CSS animations from the animate.css library.