
TECHNICAL SKILLS

Strong: JavaScript ES6+, React, Redux, React Hooks, Context API, Recoil, Node.js, Express, Functional Programming, OOP, SQL and NoSQL, PostgreSQL, MongoDB, HTML, CSS/SCSS, Git/Github, JIRA

Experienced: AWS, React Router, D3.js, Test-Driven Development (Jest, Supertest, React Testing Library), Websockets, Bcrypt, Kubernetes, Docker, OAuth 2.0, Travis CI, Shell/Zsh

EXPERIENCE

Recoilize | Software Engineer**2021-Present**

- Developed a SPA with component architecture using React to take advantage of its flexibility, modularity and diffing algorithm
- Utilized Redux for a predictable, centralized application state, with store serving as the single source of truth, and leveraged Redux toolkit to minimize certain aspects of Redux store configurations and repetitive boilerplate codes
- Implemented Recoil API to take snapshots of Recoil stateful components and aggregated relevant data to send to DevTool for visual representation and time travel features
- Integrated React Hooks to ensure reliable access to state changes while considering the hierarchical structure and the reusable stateful logic among app components
- Used D3.js to generate svg data visualization and dynamically display graphs of the react components atom family network, and performance metrics graphs and added interactive functional sliders for flexibility in user experience
- Implemented chrome's storage API and persisting reducer to store stateful info per user sessions for developer's convenience when refreshing pages during debugging
- Product developed under tech accelerator OS Labs (opensourcelabs.io)

OPEN SOURCE

Algoverse | Software Engineer**2021-2022**

- Designed and configured a SQL database using ElephantSQL to store relational data, user info and algorithms questions
- Implemented password-hashing using bcrypt for encryption to securely save sensitive user passwords, and JWTs for session authentication
- Engineered intuitive RESTful API endpoints using Node/Express server to take the advantage of Node.js cross-platform runtime environment - streamlining communication between the front-end and a SQL database
- Built application with webpack for future scalability, to deliver a performant web app & superb user experience utilizing techniques such as minification, uglification as well as ES6+ transpilation for developer packages, and fast load times utilizing hot module replacement

unbEATable | Software Engineer**2021-2022**

- Integrated Redux to achieve a centralized state management for application and manage conditional rendering
- Employed test-driven development using jest with React Testing Library for unit-testing and snapshot testing to mock the flow of data and for scalability and improved code design quality
- Designed Node.js/Express server architecture to efficiently handle HTTP requests, using modularized middleware, controllers, and routers to manage backend requests as well as improving code
- Prototyped a NoSQL database for storing unrelated data, fast deployment, ease of horizontal scaling and created organization and security using schemas to enforce data type declarations

Pomodorable | Software Engineer**2021-2022**

- Utilized React to follow a predictable and unidirectional data flow and React Hooks to design an event-driven state management system that minimizes excessive prop-drilling and efficiently renders applications based on local state changes
- Implemented NoSQL database to provide quick and flexible setup as well as modularity in timer and preset components
- Employed User Experience Driven Development (UXDD) for a scalable and dynamic application with an intuitive user interface

TalksSingle Sprout Tech Talk: **Build Tools Alternative** / 2022

PublicationsMedium Article: **Recoil** / 2022

EducationUC San Diego, Interdisciplinary Computing Arts B.A. / 2010
Java, Pure Data, Max/MSP

Interests

Build cool apps, jog around the neighborhood, karaoke my heart out, dream about some sustainable environmental design to save earth, make my daughter a trust fund babe