

Brian Tran

Software Engineer

San Diego, CA (seeking remote/hybrid position) | +1(858)-357-4104

btran946@gmail.com | [linkedin.com/in/btran946](https://www.linkedin.com/in/btran946) | <https://github.com/btran946>

TECHNICAL SKILLS

- **Strong** - JavaScript, TypeScript, React, HTML, SCSS + CSS, Next.js 13, Tailwind CSS (DaisyUI/Framer-motion), Node.js(Express), NoSQL(MongoDB), SQL(PostgreSQL/mysql), GraphQL, Authentication (OAuth, JWTs), Webpack, TDD(Jest), E2E(Cypress)
- **Experienced** - Java, Python, Scrum/Agile methodology

EXPERIENCE

QuiL | OS Labs

2022 – Present

Software Engineer

New York, NY

- Designed an interactive, responsive relational database visualizer for developers to view the relationships between their tables by implementing React Flow's highly customizable, node-based components in hopes of assisting devs to migrate to a GraphQL backend
- Engineered the core algorithms to crawl a database and generate GraphQL schemas and resolvers given a user-inputted URI
- Utilized React Hooks to create composable, functional components which gave us the ability to comprehensively manage/organize the application through related stateful logic rather than by lifecycle methods which prevents unnecessary prop-drilling and potential errors
- Used Next.JS 13 to take advantage of its default react server components which empowers optimized data fetching and reduces the bundle size/time it takes to be delivered to the client resulting in a predictable and performant application
- Constructed the backend with with Apollo Server and GraphQL to enable the frontend to precisely request only necessary data through predefined by custom built schema definitions and resolvers to alleviate under/over fetching
- Integrated TypeScript by defining consistent type definitions and enforcing them throughout the codebase to ensure compile-time type safety, produce reliable, testable, and modular code, as well as, improved developer experience through predictive intellisense
- Designed a captivating UI/UX through DaisyUI's highly customizable and responsive components on top of tailwindCSS's flexible utility classes to enable declarative styling and consistent themes, as well as, utilizing FramerMotion for smooth animations of components
- Implemented Docker to containerize and deploy our backend server through AWS Elastic Beanstalk, as well as, configured security options to only permit authorized communication between both the frontend and backend of our application
- Accessed Github's OAuth 2.0 API to increase security and integrity of the signup and login process so users can seamlessly onboard
- Followed Scrum and Agile methodology, coordinating team efforts in an efficient and organized process through hosting consistent sprint reviews and daily stand-ups to facilitate and adhere to engineering best practices
- QuiL is developed under tech accelerator OS Labs (opensource.labs.io)

OPEN SOURCE PROJECTS

capableHumans | Series of brain games and cognitive tests

- Leveraged React Router's conditional rendering to create declarative, dynamic routes within an SPA, reducing server requests with client-side routing resulting in a seamless user experience of navigating between different game modes and statistics
- Took advantage of Sass's reusable variables and nested syntax capabilities to keep colors consistent throughout the application and to have more precise targeting of elements to create an organized, maintainable, easy-to-read, and performant style sheet
- Incorporated a Node.js/Express backend to take advantage of its MVC architecture by modularizing/organizing routes and controllers based on individual game stats. Also leveraged its middleware for its elegant and precise error handling capabilities
- Architected a SQL database to leverage ACID compliance so the frontend and backend can be in sync and can rely on consistent, accurate, and secure data being shared. Allowed application to confidently handle and display dynamic statistics for each game

Smoove | Tool to compare cities to make your next move, smooth

- Integrated React for optimized performance through its intelligent reconciliation process and its declarative JSX syntax
- Implemented a Node.js/Express server to efficiently handle HTTP requests and API sourced data based on major cities names by engaging the intuitive middleware design pattern to process requests across multiple modularized controllers and routers
- Constructed a BASE compliant noSQL database to harness the advantages of the document model which include, fast queries as a result of denormalized data to quickly retrieve user data, high-availability and reliability, and flexible and fluid manipulation of data
- Curated a custom webpack config file to modularize source code into a single bundled JS file to improve speed by leveraging its uglifying and minifying processes. Also, took advantage of HMR from its dev server to optimize the developer experience

TALKS/PUBLICATIONS

- What is CI/CD? - Jeeny and Bractlet Speaker Series (2022) | QuiL- Schemas/Resolver Generator - Medium (2022)

EDUCATION

Codesmith

New York, NY

- Advanced Residency for Software Engineering

San Diego State University

San Diego, CA

- Principles of Information Systems, Computer Organizations/Assembly language, Intro to Java, Intermediate Java

INTERESTS/ACTIVITIES

- Visiting/hiking national parks like Yosemite and Zion, experiencing different cultures and their cuisines through travel, and playing football, basketball, and badminton