Hang Xu

Software Engineer

909-610-5158 | hxu009@gmail.com | github.com/nplaner | linkedin.com/in/hangxu09

Technical Skills

Javascript (ES6+), React (Hooks, Context, Router), Redux, WebSockets/SSE, Node.js, Express, OOP, Webpack, Authentication (JWT/Auth/OAuth), Relational/Non-relational Databases, Containerization (Docker), AWS (EC2, RDS,EB,IAM), HTML,SASS/SCSS,VersionControl(Git/GitHub),Testing Frameworks(Jest,SuperTest)

Experience

SynapseJS | *Real-time API Library* **2020**

- Designed a protocol agnostic library in TypeScript that dynamically creates routes and controllers, reducing boilerplate setups in Express, while also allowing scalability without complication for a streamlined and manageable workflow across development environments
- Developed a test application in React.js to simulate network protocol requests to the live server, ensuring we receive expected errors or correct responses
- Used Express.js webserver to handle HTTP, SSE, and Websockets requests utilizing custom methods such as subscribe and unsubscribe to assist in the use of Websockets and SSE
- Utilized Jest and SuperTest to create a robust protocol and endpoint testing suite while adhering to TDD principles for continuous integration throughout the development cycle
- Orchestrated and deployed a clustered instance of a demo application on AWS using load balancers and EBS to test our reliability, scalability and functionality in a real time environment
- Leveraged Typescript's static type checking to standardize method inputs, enforce validation checks, and reduce environmental errors for a scalable and maintainable code base
- Designed and implemented a SQL database to store relational data between users, guaranteeing a durable and scalable database that is ACID compliant
- Generated Typedoc documentation for an improved code base foundation while highlighting formalized design decisions to elaborate points of ambiguity for future collaborator
- Employed React Hooks and React Router to create an intuitive single page application that dynamically renders state based upon reusable functional components
- *Accelerated under Open Source Labs (opensourcelabs.io).*

Open-Source Work

CodeRacer | *Online Multiplayer Typing Competitions* **2020**

- Implemented multiplayer game mode via websockets for concurrent state changes from different players, facilitating user to user interaction over a single TCP port without requiring a polling server, reducing HTTP overhead and improving latency for better user experience
- Utilized a SQL database to store relational data used to populate our user facing application, while enabling our CRUD functionality, and adhering to ACID compliance
- Leveraged Express.js for handling HTTP and API requests at specified endpoints while operating under the middleware design pattern, this included defining routers and controllers to modularize backend requests, resulting in clear, concise and reusable code
- Incorporated Github OAuth and api after narrowing down our targeted user base, streamlining the validation process as well as gaining access to usable data for our application

800Hex | *Hexadecimal take on the classic 2048 puzzle game* **2020**

- Leveraged React-Redux with the observer pattern to optimize state management, maintaining state within a single source of truth for the application's dynamic renders
- Incorporated Bcrypt to secure passwords within PostgreSQL database taking advantage of its native one-way hashing algorithm insulating users from dictionary attacks

Education

University of California, Riverside | Economics: Law & Society **2019**

PublicTalks

Understanding Decorators | *SingleSprout Speaker Series* **2020**

Interests

Learning to cook, reading the fictional works of Patrick Rothfuss, owning noobs in Dota 2, playing chess, yearly travel to different continents, water, swimming, working out