LinkedIn- https://www.linkedin.com/in/valdezdev   Portfolio- https://valdezdev.com   Github- https://github.com/valdezdev

Paul Valdez

Richmond, VA
Email: pvaldezdev@gmail.com || **Phone**: 757-345-1792

Every day is a new opportunity to get up bright and early in order to tackle problems that need solving with a constant emphasis on self-improvement and efficiency.

#Skills

Front-End	Back-End	FrameWorks & Libraries	Dev Toolkit
JavaScript ES6 / jQuery	REST APIs	React / Gatsby	Git / Bash / Git Flow
CSS / Sass	MongoDB	Redux / Thunk	mLab
HTML / JSX	NodeJS	Bootstrap / Reactstrap	Postman
Webpack	ExpressJS	Axios	Heroku / Netlify
Babel	GraphQL	Lodash	KanbanFlow
	Python	Jest / Enzyme	WakaTime
	PHP	Django	Redux DevTools

#Project Experience

DevGab

Demo || Source

⤷ Developed this larger scale social media CRUD application on top of a MERN stack.

⤷ Executed React & Redux front-end in ES6 with Node.

⤷ Interfaced Axios to carry out requests to a RESTful API-based backend written in NodeJS, ExpressJS, Mongoose.

⤷ Achieved form validation with specific parameters using backend services provided in the API.

⤷ Hosted a Mongoose Database on mLab that holds onto profiles, posts, and comments obtained from the REST API.

⤷ Utilized JSON Web Tokens and Bcrypt to register, authenticate, provide hashes, and time users out of a session.

⤷ Designed database with Mongoose that utilized models / schema.

Hacker News Search

Demo || React Source || React Redux Source || Redux Blog Post

⤷ Incorporated Axios within this React application to make requests to the Hacker News, Algolia Search API.

⤷ Lifted state throughout the application into its various child components to keep them sleek and lightweight.

⤷ Applied Lodash to allow for the user to sort search results based on the title, author, number of comments, and points.

⤷ Employed the API to cache paginated data to allow the client to make only one API request when searching.

⤷ Practiced snapshot and unit tests by running both Jest and Enzyme.

⤷ Wrote a blog entry explaining how to rebuild the app and fetch data from the API as well as save search queries in the Redux state.

Developer's Blog Template

Demo || Source

⤷ Constructed this application using GatsbyJS and set up an API using GraphQL.

⤷ Optimized to display and paginate through markdown files that are taken from the API.

⤷ Assigned routes with a unique slug based on an individual markdown file's title as well as its unique tags using Node.

⤷ Included a Disqus plugin that allows users to comment on and/or share blog posts to their social media.

Budget Calculator

Demo || Source

⤷ Implemented React Hook syntax to contain a dynamic local state that allows for CRUD functionality.

⤷ Designed user-friendly form validation logic to keep the application free of bugs.

⤷ Stringified JSON and side effect functionality that saves the user's information to Local Storage.

⤷ Ascribed each individual user expense with a Universally Unique Identifier.