

# JAMES SAULSBERRY

(817) 470-6565

[jamesnsaulsberry@gmail.com](mailto:jamesnsaulsberry@gmail.com)

[PORTFOLIO](#)

[GITHUB](#)

[LINKEDIN](#)

## SKILLS

React, Redux, Ruby, Rails, Javascript, HTML, CSS, PostgreSQL, SQL, GraphQL, Git, Flutter, Dart, Node, Express, Mongoose, MongoDB, Apollo, Docker, AWS, Heroku, Swift, Java, Kotlin, Firebase, Firestore, Webpack, Angular, Python, PHP, Passport, React Native

## PROJECTS

**Haply** (Rails, ReactJS)

[live](#) | [github](#)

*Haply is a single page web application based on Eventbrite that allows users to create events, manage events, like events, create tickets, search for events, and get tickets to events.*

- Utilized Node Dropzone npm package to allow users to drag and drop files and set up upload to AWS S3 Buckets.
- Dynamically generated different versions of the navigation header component using React Router based on the hash-router path to optimize user experience.
- Incorporated postGIS extension in PostgreSQL to allow for geoQueries.
- Employed active record callback functions to create a custom method in Rails controller to authenticate model ownership before allowing edits.

**PartyBnB** (Express, MongoDB)

[live](#) | [github](#)

*PartyBnB is a single page web app based on Airbnb that allows users to book and create parties at locations.*

- Implemented Geospatial queries which allow users to search for spots to book by location by utilizing MongoDB's geoNear aggregation method and setting a 2dsphere index on our Mongoose model and utilized the Google Maps API to grab the relative latitude and longitude of the address being queried and used React-Router to send the info to custom REST API endpoints on Express backend.
- Configured Passport with the passport-jwt strategy to handle json web token authentication and verify requests on REST API routes and used bcryptJS to hash and salt the credentials.
- Created booking algorithm for Express API route to ensure that overlapping bookings could not occur, which responded with the appropriate json response depending on if a user attempted to book twice or if another user had already made a booking.

**MusicBox** (GraphQL, ReactJS)

[live](#) | [github](#)

*Musicbox is a single page web app heavily influenced by Amazon that allows users to buy and sell music equipment.*

- Utilized new React Hooks in combination with Apollo GraphQL and Mongo's \$text search method to allow users to search for items in the store by keywords and created GraphQL mutations to allow users to add and remove items from the cart.
- Updated GraphQL cache on mutation changes relating to the cart to allow for dynamic CSS changes to the amount of items displayed in the cart on the Navigation React Component.

## EXPERIENCE

**Full Stack Software Engineer**

*Wedoo Inc.*

Aug 2019 - Present

- Built mobile app event oriented app in Flutter and with native android and IOS that takes advantage of the physical hardware of the device.
- Operated Google Suite of tools using Google Cloud Firestore in the backend with Google Analytics for tracking and Google Cloud Storage for picture and video uploads.
- Implemented group chat that supports video, and text messaging in real time using flutter's StreamBuilder to grab messages that are sent from real time and mapped out the objects received to different widgets based on the message type for a better user experience.

**Equities Swing Trader**

*Self-Employed*

May 2007 - Present

- Interpreted charts of financial markets applying technical analysis and using support, resistance, moving averages, price

patterns, and technical indicators.

- Analyzed company financials, balance sheets, 10-k, and other documents and used fundamental analysis to execute trades..
- Implemented successful futures and options strategies as well swing trades.

## EDUCATION

**Software Engineer** - *App Academy* | *Fall 2019 - Spring 2020*

**BA Architectural Engineering** - *University of Texas at Austin* | *2005 - 2012*