Milton Ivan Martinez Gonzalez
+528127335976, Monterrey, Nuevo Leon, MX
Web and mobile developer

Experience

Uber
Feb 2021 – Present
Role: L3 Software Engineer

- Lead in charge of the development of the Uber eats promotions view per store when the user wants to purchase products.
- Lead in charge of the refactor of the Uber eats cart view to disable multiple store checkout at once, this refactor included a cart, promotions and checkout refactor to disable multiple checkout and the development of testing coverage to avoid regressions.
- Development of the implementation of a new Payment method for Brazilian users (Pix payments) in the frontend for Uber eats which included features for the user settings view, checkout and active orders views.

Technologies used: Javascript, React, Redux, Redux Thunk, Jest, React Testing Library, Sentry, Circle CI

Shopply
Nov 2018 – Present
Role: Full stack developer

Mobile shop cart application available for iOS and Android.
Shopping cart built with React Native for android and iOS currently available on the play store and the app store, this was a hobby project built in two weeks because I needed a shopping manager for myself when doing the shopping cart, consists just on SQLite db internal storage for metrics and jest with enzyme for testing coverage, CI pipeline with github actions in order to lint the code and execute the test suites, and CD pipeline also with github actions in order to publish new beta versions of the application to the stores when a PR is merged to staging and new production versions when a PR is merged to master.
Technologies used: React Native, Redux, Redux Sagas, iOS, Android, SQLite, Github Actions.

Gobierno Santa Catarina
Jan 2020 – Nov 2021

Role: Frontend and backend technical lead

Web and mobile application for polls processing with React, React Native, Node JS and MongoDB.
Technologies used: React, React Native, Node.js, MongoDB, CI/CD, GitHub Actions, Docker, Jest, Cypress.

Tango
Aug 2020 – Set 2021
Role: Senior Frontend Engineer

- Groundspeed: Unit, integration and e2e testing coverage for production application built with React and meteorJS in order to add reliability to a migration process to replace meteorJS with graphql, my role was to add testing coverage to the authentication and authorization flows of a CRM web application for brokers management. Some of the obstacles faced were that the build system was using old versions of Node and the communication between client and server (via MeteorJS) was done directly from UI components without an abstracted service layer so I needed to create that abstraction layer in order to make the testing assertions on those services, that way they dont needed to change any UI code in order to migrate from MeteorJS to graphql client, they just needed to change the service layer implementation instead of refactoring UI server calls and UI interactions.
Technologies used: Jest, React, Redux React Testing Library, Cypress.

Crossbridge Global Partners
Aug 2020 – Aug 2021
Role: Senior React Engineer

- AdvisoryCloud: Frontend tech lead building a React JS web application for advisors with graphql client using Redux and redux-thunk for state

management, Context API, hooks and custom hooks, authentication guards with custom hooks, custom wrappers and react-router-dom v5 for the route management. Also I have built a internal UI components library with React Material UI and typescript, bundled and transpiled with webpack, this library is hosted in the NPM registry so it can be installed via npm or yarn, also have unit and integration testing coverage with jest and testing library, storybook for documentation and Github actions for CI and CD pipelines, the integration pipeline consisted on linting the code and executing the test suites on each commit of a PR, then on PR merge to master, the CD pipeline deployed the new version of the library based on the release PR title which contained the new version that was going to be released and automatically pushed the tags to the remote repository.
Technologies used: React, Redux, redux-thunk, typescript, graphql, webpack, Github Actions, jest, react-testing-library, Cypress.

ArkusNexus
May 2019 – Aug 2020
Role: Senior Front-End Engineer

- FIGO Pet Cloud: Frontend tech lead bulding a React Native application for android and ios of an pet insurance company which had a social forum, AI real time chat, real time map tracking and insurance requests, it was built using Firebase for real time communication and push notifications, Redux with Redux-thunk for the state management and Lottie for animations, jest and react-native-testing-library for unit and integration testing and Appium for e2e testing. Custom promise-based libraries for communication between the mobile native APIs and react-native where built with Kotlin for Android and Swift for iOS.

- Paradise 4 Paws: Frontend tech lead building a simple hotel booking React Native application for pets, built with React Native, Context API, jest and react-native-testing-library, it hasn't a state manager library because the application was kind of simple, it was just an authentication flow, a calendar and an iframe. This was more of a techlead tryout for a team of 4 non-react developers

Technologies used: React Native, Kotlin, Swift, jest, react-native-testing-library, firebase, Appium.

X8BIT

Mar 2018 – May 2019
Role: Mobile and Web Engineer

- Unraitte: Private rides application built with React Native for the Android and iOS application, it included google maps support with real-time geolocation devices tracking for both the driver and client apps in order to draw in the map the positions of the driver and client within a travel, and also to draw in the admin map the real-time positions of all the drivers using the application and show how many are in travel and how many are not, also Conekta and Paypal was used for credit-card payments, Redux and Redux-sagas as the state management system with deepstream-io as the socket communication manager and OneSignal as the push notifications manager; For the backend we were using NodeJS with express as the router manager, Redis for the cache and MySQL as the relational database, we also used MongoDB as the non-relational database in order to store the travel coordinates, the reason why we used a non-relational database instead of the relational database for the positions was because of the speed of response in order to draw the travel history in the admin website; The admin website was built with React, redux and redux-thunk, it also had a map implementation in order to show all the drivers connected in the platform, also had an implementation of ACLs, roles and authentication flows in order to show certain data only to certain roles because the platform had multiple administrator users.

- Sensai: Quality control tablet application for android and iOS built with React Native for a company called Metalsa which is a company that builds car monocoques, the application purpose was to measure certain pieces of the car monocoque and determine if the piece has measurement failures or inconsistencies, my work here was to create a bluetooth bridge between the device application and some bluetooth beacons which were digital measurement tools such as Verniers, micrometers and nanometers, the bluetooth bridge was built with Kotlin for android and Swift for iOS, was a service that in order to retrieve the tool measurements, the service needed to be connected to the tool bluetooth subscriptor in order to listen for measurement changes and then publish the changes to the device bridge and store them into the application, then via business logic metrics determine if the measure is correct or not.

- Wheeling: Hardware and Software development for an electric charge station for electrical vehicles such as scooters, I was in charge of the development of the hardware device that received the instructions to turn on the scooter from a mobile application, this was done with a GSM

module that via MQTT(Mosquitto) was connected to a socket channel in order to receive commands to turn on or turn off the scooter, in order to turn on the scooter, when the socket channel receives a TURN ON command, the ATMega chip sends a 5v signal (Logical 1) to a MOSFET transistor base, then the MOSFET closes the circuit between the scooter

battery and the wheel motors in order to turn on the scooter, then when the socket channel receives a TURN OFF, the ATMega closes the MOSFET base signal in order to close the circuit with the battery and shut down the scooter; For the electrical charge station, it was built with a master/slaves architecture where the master was the device in charge of listening commands from the wheeling platform via WIFI with a Raspberry PI Zero W and MQTT, the slots slaves consisted of surface mount ATMegas mini communicating with the master via i2c communication, the architecture was designed in order to add as many slots as possible without any programming knowledge needed, this was done via dynamic i2c addressing with Python, when a new slott is connected to the charge station, it gets an unique dynamic i2c address so the master can send data to the slott using that i2c address, when a scooter requests a slott to charge, if a slott is available, the charge station sends a bit to the available slott in order to start a charging process which consists on the activation of current flow between the charge station battery and the scooter, the measurement of the electric current spent during the charge process, then the stop of charging which can be via disconnection of the charge cable, or the detection of full charge of the battery, all of this measures were done processing the current flow passing through a 2N2222 transistor; The mobile application that requests charges and the activation of the scooter was built with React Native for android and iOS, the rest API was connected to the MQTT channel and via client requests, sent MQTT commands to the scooter and the charge station.

Education

Industrial Mechatronics Technician - Escuela Industrial y Preparatoria Tecnica Alvaro Obregon - UANL

Systems Administrator Engineer - Facultad de Ingenieria Mecanica y Electrica - UANL

Languages

English - Fluent conversational