

EDUCATION

Atlanta, GA, USA	Georgia Institute of Technology	Aug. 2015 - Dec. 2019
<ul style="list-style-type: none"> Bachelor of Science in Computer Science major GPA: 4.0/4.0, cumulative GPA: 3.97/4.0 Minor in Physics (Graduate-level Coursework: General Relativity) //galmungral.github.io/physics CS Coursework: OOP, Algorithms & Data Structures, Analysis of Algorithms, Computer Organization, Operating Systems, Computer Networking, Relational Database, Information Security, Computer Simulation, Machine Learning (Incomplete) 		

EXPERIENCE

Front-end Developer, Intern	Pegasus CRM, Decatur, GA	Jan. 2018 - May. 2018
<ul style="list-style-type: none"> Provided a proof-of-concept Vue.js reimplementation of existing UI features for the team to evaluate framework adoption. Implemented designer's page designs using Blade PHP templates and SASS in a Laravel application. Integrated resizing and drag-and-drop features to existing data tables using vanilla JavaScript. Expedited bug fixing process by tracing into back-end code and identifying reported bugs that originate in the backend. 		

SOFTWARE PROJECTS

[//github.com/GalMunGrAl](https://github.com/GalMunGrAl)**SitBit** *Accelerometer-based sitting time recorder. (Proof of Concept)*

- Utilized [Core Motion](#) framework on iOS ([Swift](#)) and the [Sensor](#) framework on Android ([Kotlin](#)) to detect acceleration.
- Created a GitHub-style calendar heat map based on (mock) data pulled from backend using [D3.js](#) and [SVG](#).
- Embedded the visualization in mobile app using [WebView](#) on Android and [WKWebView](#) on iOS.
- Implemented the back end in [Golang](#) with a [MySQL](#) database.

ReSpotify *Reimplementation of Spotify music player*

- Recreated Spotify's playback/volume controls by integrating [Spotify Web Playback SDK](#) with a [React.js](#) UI.
- Implemented [OAuth 2.0](#) authorization flows in [Python](#) using [Flask](#) framework to access [Spotify Web API](#).

NoTube *Reimplementation of YouTube Website (Proof of Concept)*

- Partially replicated the UI using [Angular.js](#) and [SASS](#). Implemented video streaming using [Shaka player](#) and [FFmpeg](#).

WebREPL *Interactive shell in browser, as commonly seen on coding websites such as Codecademy*

- Designed a mechanism to evaluate submitted scripts on server and send back output/error to be displayed in browser.
- Implemented using a parent [Node.js](#) process that communicates with a [Python](#) interpreter child process.

TwoFactor *Two-factor authentication with push notifications (Proof of Concept)*

- Implemented two-factor login using a [Node.js](#) backend that communicates with the login page through [Socket.io](#).
- Crafted push notification requests to [Firebase Cloud Messaging](#) and to [Apple Push Notification Service](#) in [HTTP/2](#).

MARTA Passenger Traffic *(Database Course Project)*

- Implemented the UI using [React.js](#) and [Redux.js](#), and the [REST API](#) using [Express.js](#) framework and [MySQL](#) database.

Declarative DOM *UI state dependency/synchronization manager, inspired by frontend Web frameworks (Proof of Concept)***Wikipedia Prerequisite Search** *Breadth-first search of Wikipedia's citation graph (Incomplete)***WolframAlpha CLI** *Command-line utility for solving equations based on [Wolfram|Alpha XML API](#)***WebSocket Server** *[Node.js](#) implementation of [WebSocket](#) protocol***Todo Apps** *Implemented using [React NativeFlutter \(Dart\)](#), [SwiftUI](#), [UIKit \(Objective-C\)](#), etc. (Proof of Concept)***Clean Water Crowdsourcing** *Mobile app based on [Google Maps SDK](#) and [Firebase SDK](#) (Android Course Project)*

TECHNICAL SKILLS

Proficient: Bash, Vim, Git, JavaScript (ES6), HTML, CSS **Familiar:** C, Java, Swift, Python, MATLAB, shell script, React.js, Redux.js, Angular.js, Vue.js, jQuery, Bootstrap, Node.js, Express.js, Flask, SQL, socket.io **Used:** Objective-C, C#, C++, Golang, Kotlin, Dart, TypeScript, SASS/LESS, PHP, Android, iOS, Xcode, UIKit, SwiftUI, React Native, Flutter, Xamarin, Jupyter Notebook, NumPy, SymPy, Matplotlib, LaTeX