

AARRON LEE

240 Baldwin Ave, Apt #3, Jersey City, NJ 07306

CELL 347-559-1724 E-MAIL aclee91@gmail.com

GITHUB <https://github.com/aarron-lee>

LINKEDIN <https://www.linkedin.com/in/aarron-lee>

WEBSITE <http://www.aarronlee.com>

EXPERIENCE

Software Engineer December 2017 – Present

Zinc Platform, NY, NY 10012

- Quickly adapted to Zinc's Rails/React codebase and made contributions to the company's core infrastructure across the stack.
- Led the development of the Quoting Engine, a Rails high performance concurrent microservice that aggregates and optimizes insurance quotes from various third party carriers.
- Helped architect product infrastructure to handle for bottlenecks in performance related to integrating with third party insurance carriers.
- Implemented client-side features using React, including frontend filtering of data, Stripe-related integrations, and creating/improving components used for displaying insurance quotes

Web Developer / Linux Technical Specialist Spring 2015 - Fall 2017

Emory University - Department of Mathematics and Computer Science

- Website Upgrade/Port - Upgraded and rebuilt Main Department website with an updated, responsive CSS template
- Created basic MVC framework with PHP to use alongside the department's custom CMS solution, along with a JSON-based API for to modularize website structure
- Created Internal-use web applications with PHP, including video streaming, inventory management, basic information portals, and other tools
- Complied, configured, and deployed dependencies to Linux research servers

EDUCATION

Emory University Spring 2015

Bachelor of Science - Computer Science (GPA 3.1), Bachelor of Arts - Japanese

- Undergraduate Teaching Assistant for Introductory Computer Science, Summer 2014

SKILLS

Ruby, Rails, JavaScript, React, Redux, PostgreSQL, PHP, Java, Python

PROJECTS

Dischat (Rails, React/Flux) [live](#) [github](#)

A full-stack, single-page live chat application heavily inspired by [Discord](#), frontend written with JavaScript, React and Redux, backend consists of a RESTful API handled by Rails

- Leverages websockets via Pusher for real-time updates on new messages, new users, and newly added channels
- Enables Image uploading and processing through the Rails backend, hosting images on AWS S3 bucket instances
- Protects backend JSON APIs to prevent illegal/unauthorized AJAX calls with Rails validations and session token checks

Maze Life (JavaScript, CSS, HTML) [live](#) [github](#)

A visualization of maze generation algorithms and path traversal algorithms using vanilla JS and CSS/HTML

- Utilized JavaScript timeouts to add CSS classes with corresponding colors to simulate animated path traversal in maze
- Encapsulated and abstracted UI component to ensure consistency between UI and internal data structures