

# Keri Warr

905-999-2131 • [keri@warr.ca](mailto:keri@warr.ca) • [github.com/keriwarr](https://github.com/keriwarr)

## Education

### University of Waterloo

Class of 2018

- Honours Bachelor of Computer Science

## Experience

### Software Engineer • Faire

September 2018 – June 2019

- Developed order processing and fulfillment tool for small-scale manufacturers featuring competitive shipping rate selection, packing slip and shipping label generation and printing, signature confirmation and insurance, and more.
- Created an interactive analytics product enabling clients to monitor revenue, exposure, conversion, etc.
- Implemented a variety of internal tools, such as a real-time global revenue visualizer, and a developer tool platform.

### Software Engineering Intern • ZenReach

September 2017 – December 2017

- Built Go micro-services producing and consuming kafka events, replicating them across several data stores.

### Software Engineering Intern • Sortable

May 2017 – August 2017

- Integrated various Ad Demand Partners into a real-time header-bidding ad solution for publishers.
- Vastly improved codebase quality by integrating **ESLint**, and other linting tools, into our Continuous Integration.
- Prototyped an In-Stream Video header-bidding ad solution from scratch, using **Closure Compiler**.

### Software Engineering Intern • Outschool

September 2016 – December 2016

- Capitalized on highly networked target audience by designing and developing a variety of referral flows.
- Improved user retention and sense of ownership by implementing profile pages.
- Partially re-implemented back-end as **GraphQL** server, used it to develop schedule management feature.
- Delivered report on viable GraphQL implementations and represented company at GraphQL summit '16.

## Projects

### Splitwise SDK • [github.com/keriwarr/splitwise](https://github.com/keriwarr/splitwise)

2017 – Present

- The best JavaScript SDK for building third-party Splitwise applications. Use it to keep track of debts among friends.
- Optimized for ease of use: the most basic use case is only five lines of code!
- Extensively documented, well tested, and up to date.

### Hanamikojinzou • [github.com/keriwarr/hanamikojinzou](https://github.com/keriwarr/hanamikojinzou)

2018

- An engine which implements Hanamikoji, a fantastic two-player card game, built using in **Redux**.
- Enables game strategies to be rapidly tested and iterated upon.
- Works equally well with an in-browser frontend and human players, or as a headless process with bot players.

### Pledge • [github.com/keriwarr/pledge-api](https://github.com/keriwarr/pledge-api)

2016 – 2017

- A project which aims to make it as easy as possible to make casual wagers with friends and co-workers.
- Initially implemented as a **Slack** bot backed by a REST API and an integration with Splitwise.
- Implemented as a universal JS app, with a Slack-OAuth-providing **GraphQL** back-end.
- Built using Travis CI and Coveralls in order to guarantee code quality.

## Proficiencies

- **Languages:** TypeScript · JavaScript · Ruby · C/C++
- **Front-end:** React · MobX · Redux · SASS
- **Back-end:** GraphQL · Node.js · Rails