

Dispatchr

Team Savage

Alok Gupta

Jordan Nguyen

Sal Olivares

Spencer Prescott

Brian Yan

Appfolio Mentors

Daniel Vicory

Lu Jin

Problem/Motivation

- Everyone needs groceries or items from local stores
- Not everybody has cars
- Major Pain Point: **It's inconvenient**
- College students are **busy**
- College students generally live close together
 - i.e. Isla Vista
- Why not help out your neighbors and get paid?



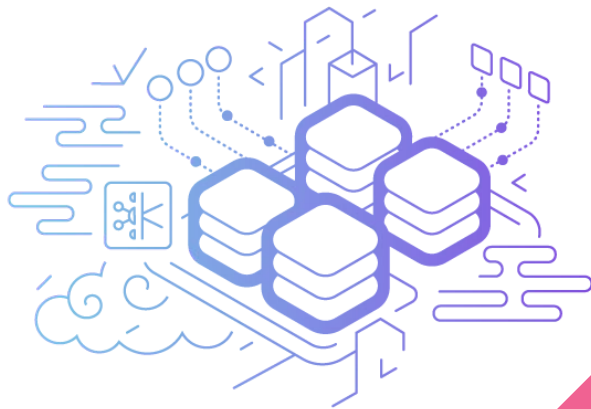
The Solution: Dispatchr

- Community powered mobile platform
- Users can post their item requests to a global feed
 - What items they want, from what stores
- Other users can accept requests and be compensated
 - These “heroes” earn points that help build their reputation
 - Anyone and everyone is encouraged to be a hero
- Items delivered right to your doorstep



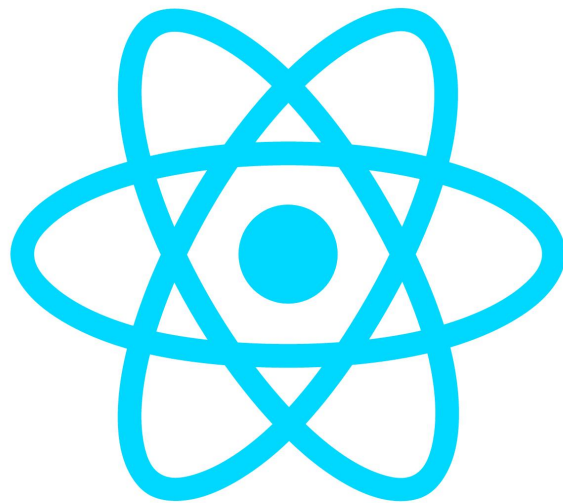
Technologies - Backend

- Ruby on Rails RESTful API
 - Lots of basic functionality out the box to build upon
- RSpec for unit testing
 - Used for test-driven development
- Deployed to cloud via Heroku
 - Cloud Platform-As-A-Service
 - High Availability



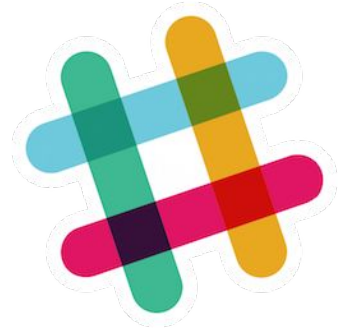
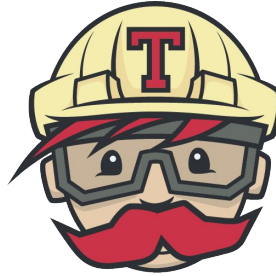
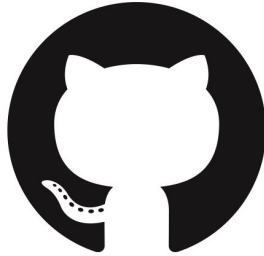
Technologies - Frontend

- **React Native**
 - Write once, run anywhere: mobile, web, desktop.
- **React-Redux**
 - State container for JavaScript apps.
- **Redux-Saga**
 - Allows us to model user flows.
- **React-Native-Router-Flux**
 - Easy routing to screens in the app
- **ES6 JavaScript**



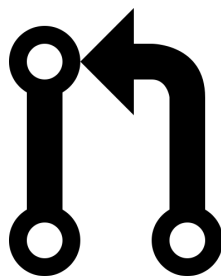
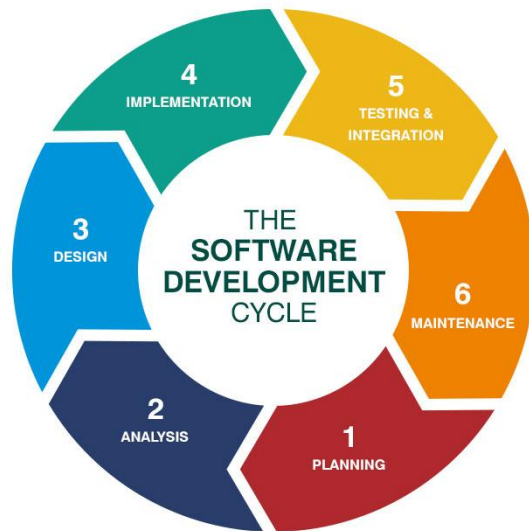
Tools Used

- Github
- Travis CI
- Heroku
- Trello
- Slack
- Hangouts
- Messenger
- Adobe XD

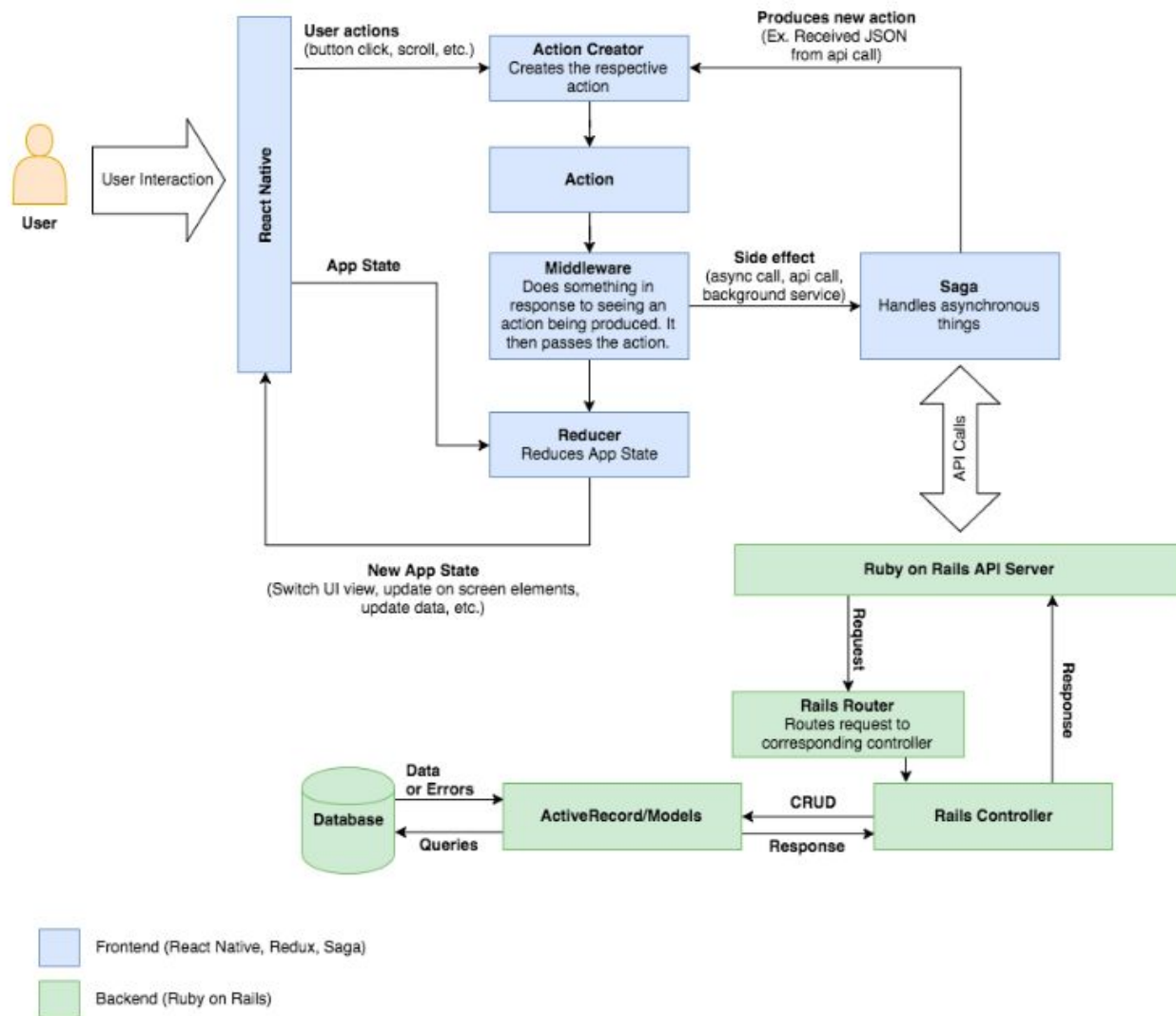


Development Workflow


1. Test-Driven Development
 - Tests must be written prior to any dev
2. Feature Implemented according to requirements
3. Pull request opened on Github
 - Entire test suite must be already passing
 - Travis CI checks must pass
 - 2 peer approvals required
4. Merge



System Design



Work Completed

- Custom API built from ground up
 - MVP endpoints setup: Items, Users, Addresses, Requests, Request Items
 - Multi-Platform Front-End Clients: iOS, Android, and Web
 - Sign-in page
 - Global requests view
 - Personal requests view
 - Request detail view
 - New request submission
 - Landscape support
- 

Demo

Next Steps

- Vision: Wireframe
- Work to be done:
 - Robust, secure login system
 - UI enhancements for the front-end
 - Real-time Notifications
 - Integrate Machine Learning/Artificial Intelligence
 - Automatically recommend items
 - Suggest which stores are best to pick up items



Questions?