

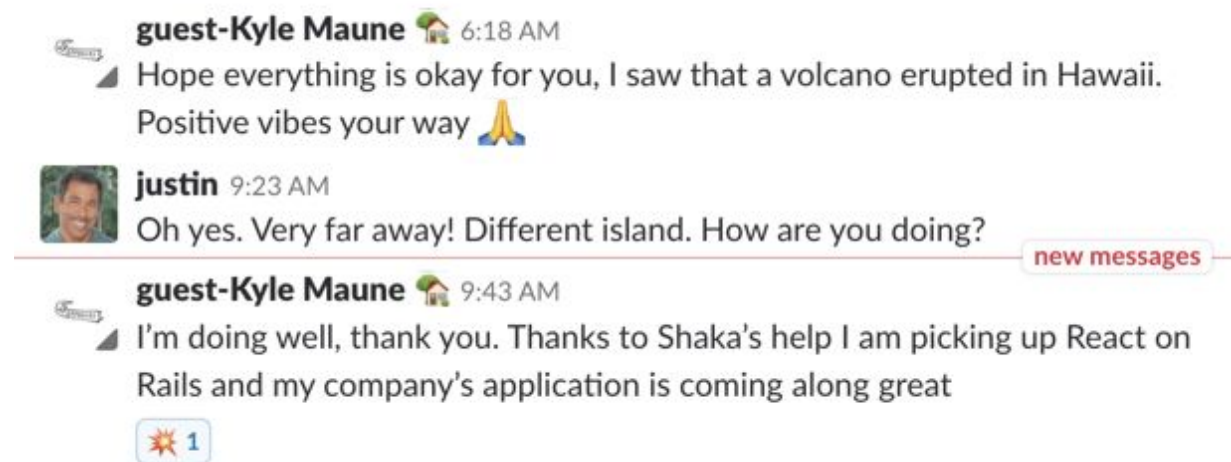
ShakaCode Pro Support

Updated June 6, 2018

For questions: justin@shakacode.com

Complement your development team with support from the technical team behind [React on Rails](#) and [Hawaii Chee](#). We want to share our knowledge with your team via chat conversations (Slack), code reviews, video calls, and screen sharing. Do you need a CTO? Is your CTO overloaded, or can use help with our specialized technologies?

From Kyle Maune of [Cooper Aerial](#), May 4,, 2018



Services Included

- License to React on Rails Pro during the time that Pro Support is active. [Docs](#).
- Providing supplemental help to your team on our team's specialties.
- A private slack channel (or we'll join yours) for quick responses.
- Pull request reviews.
- Pair programming as needed.
- Best practices and performance help. We keep on top of latest advances in React, Webpack, and all aspects of front-end development so you can focus on your code. See the listing of specialties below.
- Maximum 24 hour response time for any developer (yak shaving) question.
- Participation in our React on Rails support private slack channel where we'll do our best to answer your questions promptly.

Billing Details

- Prices regarding hours of support committed per *calendar* month:
 - 4 hours per month: \$600/month, additional hours at \$150/hr.
 - 8 hours per month: \$1100/month, additional hours at \$137.50/hr.
 - 12 hours per month: \$1500/month, additional hours at \$125/hr.
- Additional consulting per hour is available, per the plan dependent rates.
- You can increase your plan during any month or decrease it for the following months.
- Should billable time not be used within the month, it rolls over and becomes available for the following months for which a support contract is in place. You need to keep your subscription active to keep the accrued rollover hours.
- The first billing is for a full calendar month, regardless of start date. Any unused time for the first month will roll over to the following month.
- At the beginning of each calendar month, the client will receive a statement and a billing request that includes any additional hours beyond what's allocated.

Testimonials

- Daniel Ackerman of Apartment Showcase (Washington Post). [Video](#). [Notes](#).
- More testimonials on the forum.shakacode.com and at www.shakacode.com/work.

Signing Up

- Book a time with Justin for a free 30 minute consultation to see if this plan is appropriate for you. [Click here](#).
- Questions: [email Justin](#).

ShakaCode Technical Expertise

React on Rails

- We're working on more advanced examples for React on Rails that incorporate the best of our code in [Hawaii Chee](#).
- Includes a license and support for our React on Rails Node Server rendering product, used by egghead.io in production (as of 4/14/2018) with far better memory usage.

Angular Migration To Rails

- We've helped 2 teams migrate components from Angular to Rails: [Egghead](#) and [Everwise](#). We've written a detailed migration article available upon request. Please email us if you'd like that.

Rails Support

- Stripe Integration
- Devise, with Facebook and Google OAuth in React on Rails App.
- Sidekiq
- 3-phase migrations
- Query optimization
- Unit tests, Integration tests & Feature tests (RSpec & Capybara)
- Code Style: Rubocop and others

Performance

Client-Side Performance

- Identify and eliminate performance bottlenecks using:
 - React performance tools
 - Chrome Dev Tools
 - React Pure Component
 - Proper use of reselect and Redux
- CDN for Assets

Server Side Performance

- N+1 query optimization and use of the Bullet Gem
- Fragment and Russian doll caching
- Performance analysis using tools such as New Relic and Heroku's Scout

JavaScript

- React, React Native, and ReasonML
- Optimization of Webpack usage for the various environments of test, development, and production, including Code splitting, Tree Shaking, CSS, source maps and image assets.
- Static Code Checking
 - ESLint
 - Static typing with FlowType
- Asynchronous Redux with

- redux-thunk
 - redux-saga
 - MobX
- Organization of client-side code
 - Standard Redux
 - redux-ducks
 - redux-tree (our own variation of redux-ducks)
- Forms and client-side validation
 - redux-form
 - shakacode/react-validation-layer
- Other common tooling
 - reselect
 - normalizr
 - Immutable.js
 - numeral
 - date-fns and moment.js
 - react-responsive
 - react-measure
 - react-motion
 - prettier
 - react-google-maps
 - react-day-picker

Integrations

- **Stripe** for payments and subscriptions
- **Cloudinary** for image hosting and reducing page weight due to images
- **Intercom** for customer service
- **Facebook** and **Google** OAuth with Rails and/or React
- **Papertrail** for log management and analytics.
- **Sendgrid** for transactional emails.
- **FullContact**
- **DevOps**
 - Heroku
 - Feature Flags with LaunchDarkly
 - Continuous Integration with Circle CI, Travis, Semaphore, Codeship
 - Trunk Based Development
 - Docker Setup
- Picking an error reporting system (Sentry vs. Honeybadger)
- Picking a CI vendor (CircleCi vs. Codeship)

Testing

- Rails
 - Integration testing with Capybara
 - RSpec
- JavaScript
 - Using Jest, Jest snapshots, and enzyme

Product Development

- React Storybook UI Library
 - Isolated environment Importance
 - Test N props, states
 - UX/UI style guide helper
 - Components refactoring helper
- Trello, InvisionApp, and Sketch

Other Integrations

- Full Contact
- HelloSign
- Sendgrid Email Templates
- Stripe Connect

Hawaii Chee (Formerly called “Friends and Guests”)

Besides the core technologies listed above...

[Hawaii Chee](#) is based on a very similar technology stack to Airbnb with a Rails backend and React on the front end. You can see the results of the ShakaCode technologies in the features, including:

- **Advanced form editing** for the creation of a listing, including auto-saving, image handling, draft mode, and client-side validation.
- **Browsing listings**, including Google maps integration
- **Image handling**, allowing for a variety of image viewing options, such as on the tiles or going into a full-screen view.

