ShakaCode Pro Support

Updated October 24, 2017

Complement your development team with support from the technical team behind [React on Rails](#) and [Friends and Guests](#). 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?

Services Included

- Providing supplemental help to your team on our team's specialities.
- 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 listing of specialities 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 in a timely manner.

Billing Details

- Prices in terms of hours of support committed per 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.

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 [Friends and Guests](#).
- Includes a license and support for our node server rendering product.

Angular Migration To Rails

- 2 large projects: Egghead and Everwise

Rails

- Devise
- 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

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 and React Native

- 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
- Asychronous 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

- **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 styleguide helper
  - Components refactoring helper
- Trello, InvisionApp, and Sketch

Other Integrations

- HelloSign
- Sendgrid Email Templates
- Stripe Connect

Friends and Guests

Besides the core technologies listed above...

Friends and Guests is based on a very similar technology stack to Airbnb with a Rails backend and React on the front end. You can see the end results of the ShakaCode technologies in the features, including:
- **Advanced form editing** for 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.