PATRICK KING

Full-stack web developer in Ruby on Rails + React + Node.js

patrick.f.king@gmail.com
(403) 922-4960

307-1422 Centre A ST NE
Calgary, AB
T2E 2Z9

OBJECTIVE

Further my breadth and depth of skills in server and client development, and expand into cloud and devops.

EXPERIENCE

VizworX Inc. - March 2013 to Present

Unreleased NEB Data Visualization - 2017 to 2018

React.js Immutable.js Redux.js Webpack IIS

Role: Consulting Developer - Challenges: While transitioning out of the company, I bootstrapped this new visualization and assisted with scoping and estimation before handing it off to the team. The main challenge was to offload and document all of the knowledge I had built up of the client and our projects. Keeping up with code review also became interesting, as I was no longer the author of any of code, and had less frequent contact with the team over time.

Pipeline Incidents *(NEB)* ***(Github)*** *2017*

React.js Immutable.js Redux.js Webpack IIS Node.js

Role: Lead Developer - Challenges: I onboarded a new junior developer, while transitioning the team to a new React based stack, while delivering a very complex browser data visualization app under a very tight deadline. My organizational skills, time management, client interfacing and detail orientation were tested on this project! We delivered the app successfully and on time, after resorting to a limited amount of overtime. I lead the adoption of code review as a practice at Vizworx during this project.

Exploring Canada's Energy Future *(NEB)* ***(Github)*** *- 2016 to 2017*

Coffeescript Browserify D3.js Node.js IIS Phantom.js

Role: Lead Developer - Challenges: I stepped into this project after it was already underway. As a pilot project for this client we had wide latitude in our choices of technology and approach, but we had to deliver this browser based visualization under tight deadlines and with limited access to the client's hosting environment. Challenges included building the app with tools not well suited to browser app development (D3 especially), integrating Node.js applications into an IIS hosting environment, and interfacing with an external design team and external client staff. The project teams moved fast but our partners were chiefly concerned with the design of the app, ensuring that essential features mentioned only in passing were discussed, captured, and scoped was also a challenge.

Geoviz - 2015 to 2016

React.js Flux.js Leaflet Node.js Objective-C + iOS C# + WPF ArcGIS Server + SDKs

Role: Developer - Challenges: Geoviz was a sprawling prototype project, with applications all communicating with each other on Windows hosts, iOS devices, servers, and in the browser. The essential challenge was keeping all of these platforms in sync as new features were added and communication protocols changed. Another notable challenge was implementing a React based mapping application at a time when adapters for Leaflet and Mapbox were at an early stage of development, or did not exist at all. Adapting DOM-state heavy and procedural tools like Leaflet to React's less stateful rendering approach was a challenge.

Jobbsite - 2013 to 2015

Ruby on Rails Postgres SQL Rspec Cucumber Phantom.js Selenium Paperclip Prawn Ampersand.js

Role: Developer - Challenges: I began on Jobbsite as a junior developer working under a talented senior to build a custom time and order management system for a client, and ended as a leading developer on a larger team adapting the software to serve as a SAAS product. There were numerous challenges over the years, including onboarding additional junior devs, adding a rich client experience using a browser toolkit called Ampersand.js, the Rails 3 to 4 upgrade, a continuous need to optimize the speed of a sprawling test suite, balancing the needs of the client vs. the needs of the product, and implementing a reliable multi-tenant database system.

Agile Software Engineering Lab, University of Calgary (Dr. Frank Maurer) - May 2011 to February 2013

Multi-Surface Environment API - 2012

C# + WPF | Kinect | Node.js | Objective-C + iOS

Role: Developer - Challenges: As one developer in a team of six, we built a framework for cross device communication incorporating Kinect skeletal recognition and gestures on mobile devices. The main challenges included keeping codebases for different platforms in sync, and coordinating work between a medium size team of students and interns who had many other demands on their time. In and around work on this project, I started a drive within the lab to move our source control from an in-house TFS Version Control server to Github.

EDUCATION

BHSc - *Bioinformatics (Honours) - University of Calgary - 2012*

Honours Thesis: *An Algorithm for Chromatin Immunoprecipitation Sequencing Analysis.* Python | R

SKILLS + TECH

What I'm Best At

React.js | Node.js | Ruby on Rails | RSpec | Cucumber | Javascript (ES 2015+) | ESLint | Coffeescript | Immutable.js | Redux + Mobx + Flux | Git (CLI + Git Flow) | D3.js | Agile practices | jQuery | Promises + Bluebird.js

What I'm Good At

Webpack | Postgres SQL | IIS | Phantom.js | Python | C# + .NET (Web and Windows desktop) | WebGL + Three.js | Team Foundation Version Control (TFS) | Leaflet | Browserify | Heroku | ArcGIS (APIs + SDKs + server configuration) | Prawn | French (Written + Spoken)

What I've Played With

Java + Android | Objective-C + iOS | Docker | Sunspot Solr + Lucene | R | MongoDB | Go | Ember.js | MSBuild | Ampersand.js | GLSL (with WebGL) | Puppet | Meteor.js | LSL (Second Life) | Drupal | SQL Server (TSQL) | C + C++ | Django | Kinect | Blender | Selenium

OTHER WORKS

Accessible Visualizations: A Case Study (slides) - I presented work on making the Energy Futures visualization accessible, as part of the our design partner's Data Empowerment Speaker Series.

"Star Trek: Armada" Gallery - A shrine for an old computer game I enjoyed.

Coffeescript | Node.js | Three.js | WebGL

Neoderelict - A simple prototype browser game.

Coffeescript | Three.js | WebGL | Blender

References available on request