

Jason Rushton

435-680-0190

jason@jasonrushton.com

jasonrushton.com

linkedin.com/in/jrushton

github.com/fooeey

Overview

I'm a full-stack web developer with a deep understanding of every layer of web application architecture, and I'm truly passionate about building applications on the web. I've been at it since 2000 — way back in the days of font tags, web safe colors, and using pixel.gif to make table based layouts work.

As a developer who enjoys front-end work, web development is at an incredible place. With the rise of client side SPA frameworks like React and Angular, alongside leveraging Node on the server and for tooling, I'm very excited for the future of building web based applications.

Please see JasonRushton.com for additional information and project examples.

Work History

Full Stack Web Developer, Owner and Founder; The Local Network, LLC; 01/2011 – Current

I created The Local Network after a hobby project took off and start generating enough income to live on. It's provided me the opportunity to work from home for the last few years, and I've used this time to experiment heavily with new technology and techniques. I've ported some of my old applications from ColdFusion over to Node.js, and I've especially focused on mastering front-end JavaScript applications.

The money maker has been Local-Nursing-Homes.com. And although at its core it's a fairly straight forward database driven site, extra effort has been put into performance and SEO. The backend uses data caching aggressively, and front end resource concatenation and minification was implemented where beneficial. I was an early adopter of microformats and Schema.org, which were a great boosts for SEO. At its peak, the site was getting over 30,000 page views per day, with its lifetime total page views coming in just under 32 million.

My app I use to experiment with is gw2w2w.com (github.com/fooeey/gw2w2w-react), which I've rewritten from scratch at least 5 or 6 times, with many major overhauls and restructurings in between. It's a dashboard for the MMO GuildWars 2. The current version is running on React.js 0.14 using ES6 classes for components, Immutable.js for data, and Page.js for routing. Previous builds include most React releases since 0.11, a couple versions on Angular, partial builds on Backbone, and several using vanilla/home brew frameworks.

The various incarnations of gw2w2w.com has spawned a few spinoffs. Piely.net and guilds.gw2w2w.com generates hot linkable SVGs, while state.gw2w2w.com is middleware for the official GW2 API's to enhance the provided data. The guilds site has proven popular among the community and serves roughly ~100k SVG emblems each week. Source for all these projects is on under my GitHub profile at <https://github.com/fooeey>

Similar to my GW2 Guilds service, nosrc.net is a "placeholder image" service which allows web developers to easily hot link an image at the dimensions they choose. This site usually serves around 25,000 images each week.

Lead Web Developer; Transiac; 01/2011 – 06/2011

Transiac was a small startup created by a handful of ex-coworkers in the wake of our prior employer going out of business. The business model was to create a payment processing solution, based on virtual bank checks (ACH), as an

alternative payment method for merchants to offer besides credit cards. As the sole developer, I architected and deployed a platform to support one off, batch, and recurring charges, all supporting multiple banks, as well as a robust CRM and reporting platform available to the merchants.

I created a JavaScript module using Canvas which allowed customers to “draw” their signature using their mouse or touch interface. The signature, which was stored as a series of scalable coordinates, was then combined with their banking information to generate a federally acceptable image of a check. We successfully processed 10’s of thousands of transactions before funding ran out.

Lead Web Developer; iWorks, Inc; 02/2001 – 12/2010

1 year as designer, 2 years as developer, 6 years as lead developer

I started out on the design side at iWorks, but quickly moving over programming, where I soon became the lead developer. We were ahead of the curve, successfully fielding cutting edge web applications for our marketing, billing, and customer service way back in the early 2000s. I architected, designed, developed, and maintained the front-end UI, back-end applications, API integrations, reporting solutions, and databases, as well as assisted with maintaining and optimizing the servers. As the lead developer for a team of 5 engineers, I worked closely with all levels of management, as well other engineering and design teams, to support and promote the needs and goals of the company.

The iWorks CRM, on which I was the primary architect and engineer, supported hundreds of concurrent users spread around the globe, as well as all the fulfillment and lead management systems you'd expect. The CRM itself was an early single page application, using iframes instead of today's XHR due to the limitations of the technology of the time. This approach allowed us to build and maintain disparate sections of the application in isolation while inheriting state and design themes down the chain; very similar to modern SPA techniques.

The corresponding billing system was integrated deeply with the CRM, and supported with dozens of different payment gateways, and hundreds of merchant accounts with load balancing functionality, handling a combined volume of over 10 million credit transactions per year.

We built out extensive reporting capabilities, and management was able to deeply analyze all aspects of the business without needing constant engineering support to generate custom reports.

The marketing platform consisted of hundreds of landing pages for lead generation and sales, and provided support for split testing and real time reporting. We were able to very rapidly both implement new designs, and to iterate tweaks on existing designs as needed to optimize conversion rates.

The system provided in depth tracking metrics with comprehensive fraud detection, all while handling over 40,000 new acquisitions per week at its peak. Our API allowed affiliates to host marketing materials their own servers and marketing platforms as well.

Education

Dixie State College, 1998 – 2000

I attended Dixie State College, now Dixie State University, on a Presidential Scholarship, which paid full tuition and student fees in order to pursue an Associate’s Degree in Visual Technologies. The Visual Technology program included courses covering HTML, PhotoShop and Illustrator, as well as C++, 3d modeling, ShockWave/Flash and print applications. After completing 3 semesters, I was hired at an advertising agency as a web developer on the personal recommendation of my professor.

Skills

Frontend Development

Frameworks: React.js, Angular.js

Libraries: Immutable.js, Async, lodash / Underscore, jQuery Page.js

Techniques: ES6, Isomorphic, Single Page Applications

Templating: Jade, React JSX, lodash, JST

Tools: NPM, Browserify, Babel, Gulp, Grunt, Bower,

Misc: API, JSON, AJAX, DOM, Object Oriented

CSS: CSS3, Less, PostCSS, Bootstrap, Grid Layouts, Responsive

Misc: DOM, HTML5, SVG

Web Optimization

Optimization: Minification Concatenation, CSS Prefixing, Cache Busting, CDN Usage

SEO: URL Rewriting, Site Structure, Schema.org, Semantic HTML

Database Development

Engines: MS SQL, SQLite

Programming: T-SQL, Stored Procedures

Tuning: Indexing, Query Optimization, (De)Normalization

Backend Development

Applications: Node.js, ColdFusion

Frameworks: Express, Restify

Languages: JavaScript, T-SQL, CFML & CFScript

Source Control: Git & GitHub, SVN, Kiln / HG

Techniques: Isomorphic Javascript, MVC, MVVM

Server Management

Servers: Heroku, Node, Windows, IIS, Coldfusion, MS SQL