PATRICK STEPHAN

FULL-STACK WEB DEVELOPER

314.873.4824
pstephan1187@gmail.com
www.patrickstephan.me

https://linkedin.com/in/pstephan1187
https://twitter.com/pstephan1187
https://github.com/pstephan1187

I am a self-taught web developer who started experimenting with web development in high school. I began by hand-writing each page in FrontPage. Over time, I learned Flash and ActionScript, PHP, and Javascript. I started following other developers on Twitter and discovered the power of frameworks. Today, I develop web-based applications mostly using Laravel and Vue.js.

WORK EXPERIENCE

I have been building websites and web applications both professionally or as a hobby for 15 years. I have been using PHP since 2010 and Javascript since about the same time. Laravel has been my server-side framework of choice since version 4.1 and Vue.js, my client-side choice, since version 0.11. Recently, I have been focusing on building client-side SPAs that access microservices via an API. I have been using Nuxt and Tailwind CSS for the client-side apps and hosting them on Netlify. I typically host server-side apps on Digital Ocean.

ALL STAR
Apr 2018 - Present

All Star is a Chrysler-family (Chrysler, Dodge, Jeep, RAM) car dealership. My responsibilities at All Star included the development and deployment of ancillary websites, automating inventory processing, creating website templates, and developing an in-house dealership website. Most of the applications were Vue.js SPAs built on Vue CLI or Nuxt, styled with Tailwind CSS and powered by Laravel-based microservices.

https://www.gogetapproved.com

Go Get Approved is a website that was designed to attract low-credit consumers to our dealership. It includes a powerful inventory search system that filters results as you type. It also includes a wizard-like system to walk users through a long form.

Inventory System

I built a custom inventory system for the dealership that imports vehicles from a 3rd party inventory provider (Homenet IOL) and processes them into a system that makes it easy for our other applications to access. The inventory system also imports vehicle incentives, applies them to the vehicles, and exports them back to Homenet for aggressive pricing on all sales platforms. A frontend was also added to allow personnel to add, remove, and override vehicle incentives.

Email Templating Software

Inventory is constantly changing at dealerships. Since you don't want to send email templates with incorrect inventory, I built a system that allows me to create templates that automatically pull in inventory listings.

METROGISTICS (RECENTLY REBRANDED TO ACERTUS)
Oct 2016 - Apr 2018

Metrogistics is a vehicle transportation brokerage company. They transport vehicles for manufacturers, dealer groups, and individuals. I worked with a team of developers there to build and maintain the software that the company used to manage and sell transportation services. The sites and applications built here were traditional web apps built on Laravel, Vue.js, Bootstrap and hosted on either Digital Ocean or AWS.

Transportation Management System

This was their flagship system. The TMS allowed their largest customers to order transportation services online. It was a multi-tenet, hyper-customizable ordering system. Each customer had a custom design and multiple custom-built ordering processes. This introduced very interesting challenges. I led efforts to simplify the codebase, add customization features, establish git-flow procedures, upgrade the framework, introduce zero-downtime deployments, implement automated testing, and make other improvements.

Performance Tracking Software

This was a system that I was tasked with building from the ground up to give managers the ability to track the performance of satellite offices. Users could import data from a CSV or add it via a form. Data would be presented to them via a dashboard that could display charts, tables, stats, etc. The dashboard could be customized and saved into multiple configurations to easily switch between different views.

Order Automation

Metrogistics' core software system that every application relied on was a very old, 3rd party system. Unfortunately, this system did not have an API and would randomly crash at various stages of inputting orders. Adding data had to be done by hand and was very time-consuming. I was able to utilize a headless browser to automate the inputting of data. I also built it to run each process in multiple steps and reattempt any failed steps. This automation saved thousands of man-hours and we were able to move employees to other responsibilities because of it.

Other Contributions

There were a number of other systems that I worked on as well including a customer care portal, a phone/SMS based status updating system built on Twilio, and a system that imported orders from NetSuite. I implemented unit and end-to-end testing, zero-downtime deployments, and other reliability improvements on a number of systems.

BLUSOLUTIONS
Mar 2012 - Oct 2016

Blusolutions was a web agency that specialized in car dealership websites. I worked with a team of other developers and designers while here to build websites and custom applications for a number of dealerships and other companies. Apps were built using a number of different technologies including Laravel, CodeIgniter, ModX, Facebook, Vue.js, AngularJS, jQuery, Bootstrap, and hosted on Media Temple, Digital Ocean, or AWS.

Torch

Torch was a multi-tenant CMS. It allowed you to build pages using draggable blocks that could dynamically analyze the SEO power of the page. You could create websites on the fly and host multiple white-labeled site managers on it.

Comparison Pages

Our dealership sites had what we called "comparison pages". They would line up a vehicle that the dealership sold with several competitor vehicles. Stats would be displayed that could be compared along with images and incentives to promote the vehicle. A backend system was built to import and manage the data. Managers could upload the data via CSV or manually. The system also integrated with the Edmunds API for data, incentives, and images.

My Market Now

My Market Now was a custom ordered project. A client needed software to manage and track tent-sales at dealerships. We built a custom solution that allowed the client to add sales events to the system, manage salespeople, input orders, and track and chart sales to determine profitability.

DIVINE DESIGN
Apr 2008 - Jan 2012

Divine Design was a web agency that specialized in church websites. While working here, I learned Flash and ActionScript, Javascript, PHP, and all the basics. I built and upgraded the multi-tenant CMS we used along with all the sites that it hosted. I also built a number of smaller applications that we used in-house. All the systems were built using vanilla PHP, jQuery, and Flash and hosted on various shared hosting services.

MY SKILLS

APPLICATION UI

I started, years ago, making UI more friendly by utilizing jQuery. I moved on to AngularJS, which was a vast improvement with 2-way data binding. Now I use Vue.js to build much more rich, front-end UIs for the applications that I work on. I work

on creating code that is as modular and easy to test as possible. The application might be complex, but the individual parts are simple. I also like SPAs because of how fast they make the application feel.

SIMPLE (BUT WELL ORGANIZED) CODE

I take great pride in how my code is organized and maintained. I want it to be easy for someone else to pick up and work on, but also be able to handle complicated logic. I try to use simple patterns where it makes sense, and use abstractions where needed. I do my best to break things down into self-contained parts. Making pieces that work on their own but can be assembled together to form complex applications can help to ease the load of maintenance and support.

BROWSER AUTOMATION

I have built a number of applications that have needed browser automation. I have used Puppeteer, Laravel Dusk, Selenium, and PhantomJS to provide solutions to fit needs that were otherwise impossible.

MULTI-TENANT APPLICATIONS

Most of the jobs that I have had, I either built or heavily contributed to a multi-tenet application. In these applications I had to take consideration for things like database connections and migrations, queue connections, testing, command line tasks, customization, etc; all of which adds complexity to application development. I was able to extract these in such a way to make the application easier to maintain.

MY PROJECTS

PERSONAL BLOG
https://www.patrickstephan.me

Occasionally, I like to blog about my experiences, and I do that here. I built this site using Tighten's static site generator, Jigsaw, and Bootstrap.

A RESUME GENERATOR

I have found it hard to find a good way to create attractive resumes that were easy to update. Standard word processors make boring resumes, and graphic design tools make them hard to update. So I built a system that allows me to design a resume using HTML and CSS, and then convert it to a PDF using a headless browser. Now I can update my resume without dragging text blocks all over the place. Built using Vue.js, Nuxt, and Puppeteer and hosted locally. Maybe one day I'll turn it into a service.

VUE.JS DATATABLE
https://vuejs-datatable.patrickstephan.me

At the time that I made this component, I could not find any Vue datatable components that I liked so I built my own. I built the documentation site as an SPA using Vue.js and Tailwind CSS.

OLD BLACK BOOK
http://oldblackbook.net

This was one of my first side projects I built years ago. Its a very simple vanilla PHP app for Bible search and online reading.

SNOWMAN SPRINT
https://www.patrickstephan.me/snowman-sprint

This was a fun project to build. I have wanted to build a game since I was a child. I followed a simple tutorial on building a game in Unity 3D. I took that tutorial and expanded it to produce this game which took about 20 coding hours from start to finish. I used Unity 3D, Blender, and C# to build the game.