RYAN TAYLOR

SENIOR WEB DEVELOPER

ryan@ryantaylor.info
(801) 448-6584

Profile

I am a software developer with more than 10 years of experience working with web technologies. I am interested in complex web systems built using Java, Javascript, Python and the great tools and frameworks that surround those technologies. I enjoy learning about the latest trends in software, and rediscovering classic books about best practices and design patterns.

Skills Knowledge and Abilities

Design Oriented

I value well-designed software that emphasizes security, reliability, readability, and testability.

Web Focused

I have a deep understanding of the web technology stack, including: Nginx, Apache, HTTP(S), and SQL.

Experienced

I have developed and maintained large-scale applications based in Java, Javascript, Python and PHP.

Technical Skills

PHP	Python	Java
Javascript	Laravel	Vue.js
React	HTML	CSS
Git	Docker	PostgreSQL

Professional Experience

Software Engineer/Architect

University of Utah **2010-Present**

Built and maintained various web applications. I managed team projects to build new web apps and add features to legacy applications. These projects include: an application for hiring TAs, an application for helping students apply for scholarships, a scheduling system for reserving lab tools, and several more.

Website Maintenance and Development

Ticket Exchange Utah **2006-2010**

Maintained website. Built web applications for internal inventory tracking. Automated business processes to reduce the amount of manual work required.

Software Developer

Verite **2004-2006**

Built custom websites for clients using PHP, HTML, Javascript, CSS and C#. Handled bug fixes and feature requests for legacy software.

Education

B.S. in Computer Science

University of Utah **2009**

Projects

API Design and Implementation

Technologies: Java, Dropwizard, Javascript, React.js, HTML, OpenAPI

In my position as software developer for the Utah Nanofab lab at the University of Utah, we were dependent on using a legacy software system to manage many of our internal lab processes. The legacy software was written in Java and provided many useful features for lab management including supply tracking, user roles, training, equipment reservations, policy enforcement, maintenance, and other features. In order to improve the experience of our lab members, we decided to implement a browser interface for giving them better access to the system on mobile and in the browser. Because we needed to limit our changes to the upstream project source code, I decided on an architecture that exposes the legacy code functionality through a web-based API. The resulting project used a java service as the web api and a client code base that used html, javascript, reactjs for communicating to the api. The api handled authentication and translating all communications to the underlying software.

Billing and Invoicing System

Technologies: Python, PHP, Javascript

I created a system to implement the business rules that were defined by our accounting department for billing internal and external teams for various items including staff time, supplies, and reservables. The final system is used monthly to process all billable items and generate invoices for clients. Each step of the process creates a detailed, auditable log that can be used to compare each run to historical data. TDD practices were used to ensure compliance with the business rules.

Academic Profile Web Application

Technologies: PHP, Javascript, Laravel, Vue.js

In order to make it easier for faculty to keep their academic profiles up to date, I designed and implemented a system that gives them the ability to edit their personal profile and highlight various accomplishments. It gives them the option to automatically pull in publications from various third party sources so they don't have to manually perform updates. The front-end was a Vue.js application and the backend was written in PHP (with Laravel). The automatic publication scanning involved integration with web APIs provided by several different services. Each scan notifies the authors of newly discovered publications.

Purchase Request Management

Technologies: PHP, Javascript, Laravel, Vue.js

I created this system to replace the ad-hoc workflow of tracking purchase requests through emails and excel spreadsheets. After gathering a list of requirements from the administrators who were manually tracking these purchases, I was able to create a structured application that formalized the rules they had in place. The system used Laravel on the backend and Vue.js on the frontend to handle authentication & authorization. There are different levels of user roles with appropriate levels of access and notifications at each step of the process.

Ryan Taylor — ryan@ryantaylor.info — (801) 448-6584