Software Developer Internship Final Report

Dynamic Search Redesign in Salesforce CRM Environment

**Student:** Fred O’Hara

**Company:** Trident Contract Management

**Location:** Fitchburg, WI

**Starting date:** March 28, 2016

**Completion date:** August 9, 2016

**Total hours:** 254+

Contact information:

(608) 512-7356 (cell)

[fohara@madisoncollege.edu](mailto:fohara@madisoncollege.edu)

Trident Contract Management is a provider of Salesforce configuration and application integration solutions for many companies of varying sizes. They have a team of 4 developers and 6 interns working to provide programming and administrative support in a wide range of tasks that vary from point and click configuration to complex manipulation of data delivered in sequenced workflows and presented in both out of the box and completely customized views.

The Salesforce programming environment consists of their own programming, database, and markup languages that closely follow existing languages. These technologies include: Apex programming language (Java-like), SOQL query Language (SQL-like), and Visualforce (HTML/CSS-like).

One of the staple features that Trident provides its clients is a dynamic search feature that allows the client to search through their own data and export particular views based on filter criteria that updates itself on the page as you type. This solution has been well received, but customers were concerned about the deteriorating performance of the feature that had been originally created by a former employee of Trident.

I took up the challenge of redesigning this feature during those times I wasn’t engaged in billable work for an active project at the encouragement of my supervisor, Temple Sutfin and the senior developer Drew Casebeer. The key weaknesses of the current design were that it took too long to load large datasets, had an outdated front-end design, and queried the database with each keypress; further increasing overhead and counting against the database call maximum count.

In speaking with the senior developer, the specific goals they were seeking for the redesigned feature consisted of:

1. More performant
2. Front end only (database calls through front end API instead of Apex controller)
3. Retain the existing functionality

I began by discussing these goals with the senior developer. We also covered possible technologies to accomplish these goals. The result of the initial conversation sent me in the direction of a technology called Angular.JS.

After roughly 30 hours of research and the implementation of a basic proof of concept I had a working demonstration of the dynamic search using an Angular module. One of the advantages of angular is a built in binding between input fields and the view. This allowed the data to live update as information in the input was being typed.

After review of this proof of concept, we determined that while Angular did the job well, we would have to write too much code from scratch to adapt it to the differing requirements of the various implementations of the search. After a little more digging, we discovered a javaScript plugin called Data Tables. The advantages of this plugin were primarily in the pre-built code that we could extend from, allowing flexibility while keeping the amount of custom code needed to a minimum.

In roughly 30 hours, I was able to put together a proof of concept as well as the first redesign implementation in a sandbox environment. This first implementation has received very positive reviews and is currently in a testing phase before it gets implemented.

In Summary, the design of the new dynamic search required research into multiple technologies as a best solution was considered. I was able to learn many new and useful concepts for calling and handling JSON data, making API calls, and processing that data through javaScript frameworks and plugins. This was a very educational project in what has been a great opportunity for me to grow as a developer.