Full Stack Web Development in Vue.js

By: Cole J Calamos

Introduction

The NKC Family Foundation (NKCFF) requested for me to develop an application to replace their current management software. After a couple months of solidifying what exactly they wanted to be done, during this time of J-Term and what is expected from me to continue on after the term, I was able to form a team and create a project proposal to begin development.

Technology Stack

- Vue.js
- Webpack
- Express
- Node.js
- Babel
- Postgres
- AWS
- JWT

I contributed to choosing most of the technology stack as I felt that through my past experiences at the various companies I have worked for and through me looking into the future of what we all will be using at the various jobs that we are all applying for. Although some of the technologies just made sense to use together other larger pieces, such as Vue.js and JWT, was chosen due to their documentation and ease of use for those developing on this project.

Objectives

- Access certain company information from anywhere
- Store Documents, Contact Information, Organization information, and Grant Information
- Ability to change what forms are displayed on their non-secured page
- Redesign their public non-secure website

Front-End

The front end was developed using Vue.js and I learned a lot about single page applications through out this project. As well I was tested on my designer skills with creating UI mockups for the customer of the various pages that went into the application and then developed those images into actual code.

The most challenging part about developing from mockups is getting all of the strange shapes to properly form and keep its ratio throughout the different screen sizes. However through the use of SVGs we were able to accomplish this without too much hassle.

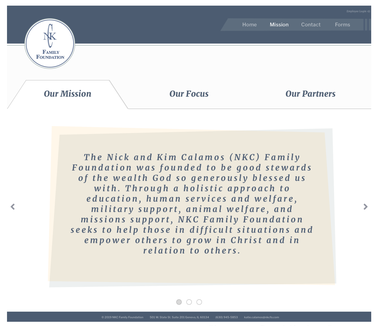

Figure 1: Our Mission Page

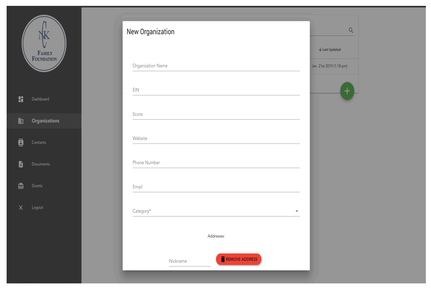

Figure 2: Create Document Page

Once most of the views were done I went ahead and created the models that would interact with my API service to request information properly from the back end server. These models and services would verify every request coming in and would format the data that was coming in and going out of the different views.

Deployment

Deployment was done using Amazon Web Services, due to the great support that amazon provides and the low costs that will play a role in the future for NKCFF. I decided to use an AWS S3 bucket as it is very low cost and has the capabilities of hosting a static web page that is created through webpack. I also chose an RDS (Relation database server) as it made the most sense for hosting a postgres database. Finally I decided on using an elastic beanstalk for the backend deployment as it allows for easy scalability due to the amount of requests it may or may not receive on the various API routes, all while running the node server.

Back-End

The back end work that I accomplished during the interterm was authentication, authorization, controllers, middleware, and routes for all currently existing functions in this application. The authentication and authorization for this application was complicated as I wanted to test my abilities to create a secure way of authenticating that the request and response of every interaction between the back end and front end would be verified. In order to accomplish this I used JSON web token (JWT) to check the unique signature of request and response done on the front end and on the back end.

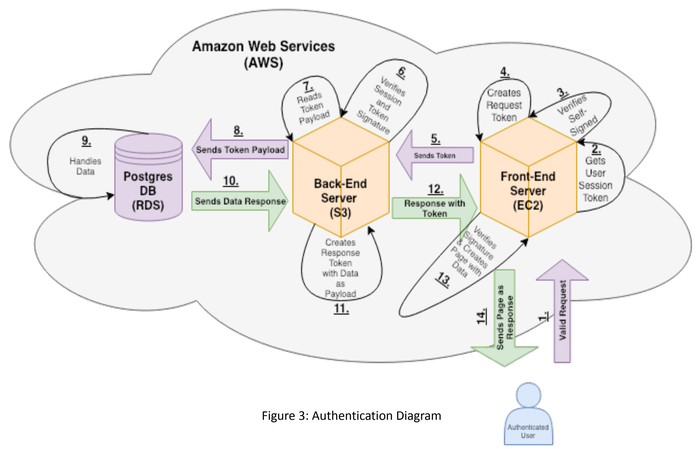

Figure 3: Authentication Diagram

Database

The database consisted of thirteen different tables that all were connected to each other through some sort of connecting table. I was responsible for creating the database and the tables with the unique indexes on each of the unique columns of the table. As well, I create insert statements to create dummy data that was used to test out our application while developing. Finally I created a database function to generate a `UUID` for the session's and I created a trigger to auto update the `updated_at` column of the row that is being updated, with the current timestamp.

Conclusion

In conclusion my team and I were able to deliver a minimal viable product that reached all of the objectives that were in scope. I learned a lot more about full stack development through out this process and was able to better understand how to intertwine different parts of the stack all together to work in harmony and to produce a product that I am willing to put my name on.

Future Work

- Accounting Features
- Guidestar integration
- Break components further into abstract functions
- Ability to comment on documents
- Ability to view documents online, before downloading

Acknowledgements

Special thanks to:

- Dr. Jonathan Geisler for proctoring and providing helpful insight into this project.
- Katie Bifoss for being very willing to meet with us and provide answers and clarifications to all and any requirements.
- NKC Family Foundation for providing us with this opportunity to work on a real-world application.