

WEB701 Assignment 3

Carlos Amos

Contents

Introduction	2
How the technology was implemented	2
Problems Faced.....	2
Impact on the future.....	3

Introduction

The technology that was used in the development of the project is Vue.js. Vue is a progressive framework that is mainly focused on developing the view layer of an application. This framework was chosen for this project as the framework is easy to learn, has reactive components, easy to setup, from installing to the start of development is fast, and Vue allows developers to start simple and add more tools and features as the app develops.

It can be used to build components which can be connected to a html template and the data within the components is reactive so if a value within the data is changed, the component will automatically update, and the framework supports a plug-in system where the developer can drop in anything they want.

How the technology was implemented

Firstly, the app was implemented by installing the necessary packages which included Vue and Vue CLI which is a tool used for Vue development. The structure of the app was then setup so the all the pages would be stored under Views and any components which included the navigation bar, and any forms were stored under Components. All the add-on packages were installed and imported which included Vuetify, Vue Materials, and Bootstrap Vue which are all components which have templates for UI elements and Firebase was also imported which would be used for the communication with the backend to the firebase database.

After the app was setup and addon packages were installed, development on the app started. The Home page was the first page which was implemented. With Vue's file structure which contains the HTML template, script and styling in one file, VS code allowed a fast setup of the template by typing Vue and enter and the whole template would be inserted. For the layout of the page, is used containers which contained rows and columns which allowed me to place content within the page in the place I wanted it easily. The page navigation was the next step which involved creating a router.js file and a navigation component in the Component folder. As the files in the components folder have the same file structure as the page files, I could easily get a template setup and begin designing and implementing the routing and navigation. As Vue allows developers to call components, I was able to call the Navigation component from the main app and was implemented without a problem.

The Login, Register, Store and Profile pages all feature components that are called from another location. Having the components within its own file allowed the size of the page files and code to be smaller which made it easy to read what is within the page.

Firebase was also implemented to the project which allowed data to be called from the Firebase Database through the backend and then shown within the website. Setting up Firebase within the website was relatively simple which involved getting the data from the backend in the form of a JSON file and then putting it into a list and displaying the information in the desired position using the name of the desired data name.

Problems Faced

One of the problems I faced with the implementation was with the login system. The plan for the login system was to send the login information and an authentication token which would be stored as a cookie when the user logged in and then call the backend again using the authentication token to check if the user were logged in. The issue with this was calling the backend to check if the user

was logged in as an unauthorized error would occur. I was not able to find a solution to this problem, so I had to redesign my login system.

Impact on the future

For the future of Vue, I do expect Vue to continue growing as learning Vue is not as difficult as other frameworks like React and Angular. As Vue continues to grow, I think it could become an extremely popular framework. Currently there are more jobs using React and Angular, but it is difficult to say what the future holds for Vue.