May 19, 2022

Frank Schulz

WEB701

NMIT

Web Framework Comparison

Milestone Two

Contents

[Introduction 2](#_Toc103847843)

[Common Web Framework Features 2](#_Toc103847844)

[Features used in this Website 2](#_Toc103847845)

[Framework One (MERN Stack) 3](#_Toc103847846)

[Login 3](#_Toc103847847)

[Register 4](#_Toc103847848)

[Tokens 6](#_Toc103847849)

[Framework Two (MEAN Stack) 7](#_Toc103847850)

[Login 7](#_Toc103847851)

[Register 8](#_Toc103847852)

[Tokens 10](#_Toc103847853)

[Comparison 11](#_Toc103847854)

[Recommendation 12](#_Toc103847855)

[Bibliography 13](#_Toc103847856)

## Introduction

Web frameworks were created in order to simplify the web development process:

“A **web application framework** is a type of framework, or foundation, specifically designed to help developers build web applications. These frameworks typically provide core functionality common to most web applications, such as user session management, data persistence, and templating systems. By using an appropriate framework, a developer can often save a significant amount of time building a web site.” (*Web Application Framework - DocForge*, 2015).

### Common Web Framework Features

* Built-in cross-browser compatibility
* Pre-existing styling
* Components
* Routing
* Extensions

### Features used in this Website

* Built-in cross-browser compatibility
* Components
* Routing
* Extensions

## Framework One (MERN Stack)

### Login

#### UI

Graphical user interface, text, application, email

Description automatically generated

#### Frontend

Text

Description automatically generated

#### Backend

Text

Description automatically generated

### Register

#### UI

Graphical user interface, text, application, email

Description automatically generated

#### Frontend

(Incomplete)

#### Backend

Text

Description automatically generated

### Tokens

#### UI

(Incomplete)

#### Frontend

(Incomplete)

#### Backend

Text

Description automatically generated

## Framework Two (MEAN Stack)

### Login

#### UI

Graphical user interface, application

Description automatically generated

#### Frontend

(Incomplete)

#### Backend

Text

Description automatically generated

### Register

#### UI

Graphical user interface, application

Description automatically generated

#### Frontend

A screen shot of a computer

Description automatically generated with low confidence

Text

Description automatically generated

#### Backend

Text

Description automatically generated

### Tokens

#### UI

(Incomplete)

#### Frontend

(Incomplete)

#### Backend

Text

Description automatically generated

## Comparison

The differences between the two stacks I have chosen to compare are confined to the frontend, as suggested by their names:

**MERN Stack**

Mongo, Express, *React*, Node

**MEAN Stack**

Mongo, Express, *Angular*, Node

Because of this I will be focussing my comparison on the frontend part of the two stacks as they can actually be switched between, leaving the rest of the app the same.

The most noticeable difference is that React uses JavaScript and JSX while Angular uses Typescript to render and control their templates. I found it much easier to manage variables in React as they are much more directly available (with the use of the ‘useState’ React functionality) when compared to Angular.

Angular is an actual MVC framework unlike React which technically a library.

“The learning curve for Angular is relatively high with React possessing a low learning curve that is initially difficult to grasp” (Daniel Fleury, n.d.).

## Recommendation

|  |  |  |  |
| --- | --- | --- | --- |
| **React** | | **Angular** | |
| **Pros** | **Cons** | **Pros** | **Cons** |
| * Easier to learn * Reliable * Intensive * Straightforward programming | * Difficult to use in traditional MVC frameworks * Has a more limited use case | * Runs on all browser environments * Reliable * Bi-directional data binding | * Steep learning curve * No extensive, all-inclusive documentation or clear manual. |

“When there is a lot of dynamic content in your application that is when React would be the right choice. Many popular brands like Instagram and Facebook prefer to use ReactJs to base their mobile apps on because of its dynamic nature” (Technostacks, 2021).

The site I am building will have a fair amount of dynamic content, paired with the above pro/con comparison and that I have used React before; I believe that the best option for building the NFT website is React.

## Bibliography

Daniel Fleury. (n.d.). *Angular vs. React in 2022: Side-By-Side Comparison | Trio Developers*. Retrieved 18 May 2022, from https://www.trio.dev/blog/angular-vs-react

Technostacks. (2021, December 8). React vs Angular—Which is Best For You in 2022? *Technostacks Infotech*. https://technostacks.com/blog/react-vs-angular/

*Web application framework—DocForge*. (2015, July 23). https://web.archive.org/web/20150723163302/http://docforge.com/wiki/Web\_application\_framework