

**ADOPTION OF WEB APPLICATION
FRAMEWORKS TO ENHANCE WEB
PROJECT RESULT**

LAB UNIVERSITY OF APPLIED
SCIENCES LTD
Bachelor of Business Administration
Degree programme in Business
Information Technology
Autumn 2016
Author: Anh Le

Abstract

Author(s) Le, Anh	Type of publication Bachelor's thesis	Published Spring 2020
	Number of pages (40)	
Title of publication Adoption of Web application frameworks to enhance web project result		
Name of Degree Bachelor of Business Administration, Business Information Technology		
Abstract <p>The year is 2020 and we have already passed a fifth of the 21st century. Digital world is emerging and developing every second and web application frameworks, a useful and optimal tool for digital world development are rising fast. Angular, React and Vue are the three industry-leading frameworks.</p> <p>The research provides a detailed view about the current trending web frameworks, how they work, what they bring to the table and their pros and cons. This research's purpose is to become a helpful and effective tool for web and application developer newbies or experienced developers but not familiar with modern frameworks.</p> <p>The theoretical framework of the thesis provides basic information, containing history and basic implementation of Angular, React and Vue. After that, a practical comparison between the three frameworks' technical specs follows. This action gives a detailed view about each framework's technical component. Then, to obtain community's familiarity about these frameworks, a small survey was made to collect both qualitative and quantitative data.</p> <p>By analyzing all above information, we can see the advantages of adopting web application to a web related project.</p>		
Keywords Web application frameworks, Comparison, Pros and Cons, Survey, Framework Adoption		

LIST OF ABBREVIATIONS

HTML – Hypertext Mark-up Language

CSS – Cascading Style Sheets

CLI – Command Line Interface

DOM – Document Object Model

Npm – Node Package Manager

JS – JavaScript

JSON – JavaScript Object Notation

Dev – Developers

WWW – World Wide Web

HTTP – Hypertext Transfer Protocol

URL – Uniform Resource Location

CGI – Common Gateway Interface

W3C – World Wide Web Consortium

C# – C# programming language

PHP – PHP programming language

Java – Java programming language

IE – Internet Explorer

MF – Mozilla Firefox

TABLE OF CONTENTS

LIST OF ABBREVIATIONS.....	3
1 INTRODUCTION	1
1.1 Thesis background	1
1.2 Thesis objectives, research questions and limitations.....	1
1.3 Theoretical framework	2
1.4 Research methodology and data collection	2
1.5 Thesis structure	4
2 FRAMEWORKS OVERVIEW.....	5
2.1 Frameworks introduction.....	5
2.1.1 Front end	5
2.1.2 Back end	6
2.2 History of web frameworks.....	6
2.2.1 CSS and CSS frameworks.....	7
2.2.2 JavaScript and JS frameworks	9
3 WEB FRAMEWORKS.....	10
3.1 Generic web frameworks history.....	10
3.2 Angular	10
3.2.1 Brief history	11
3.2.2 Implementation	11
3.2.3 Generic features	13
3.3 React.....	14
3.3.1 Brief history	14
3.3.2 Implementation	15
3.4 Vue.....	16
3.4.1 Brief history	17
3.4.2 Implementation	17
3.5 Frameworks comparison.....	20
3.5.1 Pros and cons of Angular	20
3.5.2 Pros and cons of React	21
3.5.3 Pros and cons of Vue	21
3.5.4 Conclusion.....	22
4 STATISTICS BREAKDOWN.....	23
4.1 Survey statistics.....	23
4.1.1 Quantitative research.....	23

4.1.2	Qualitative research.....	25
5	CONCLUSION.....	28
5.1	Answers to research questions.....	28
5.2	Reliability and Validation.....	29
5.3	Suggestions for further research.....	29
5.4	Summary.....	29
	REFERENCES.....	31
	APPENDICES.....	34

1 INTRODUCTION

1.1 Thesis background

Internet is a no-brainer concept these days. From 70s elders, to teeny tiny children are familiar with The Net. Considered the communication revolution in the late 20th century and early 21st, Internet has turned our life over a new leaf, making its way from a connection method in military to our everyday communication (Dentzel 2019, Fitzpatrick 2017).

Having been participating in a working environment at a start-up tech company as a junior application and web developer, I believe in order to implement an outright, interactive and functional web project. The adoption of web development frameworks is crucial, especially in nowadays buzzing online world, building projects from scratch like pure JavaScript or HTML, CSS is redundant and not necessary. By collecting data and analyzing the model of many frameworks, the research will provide a deeper view of what web frameworks bring to the table.

1.2 Thesis objectives, research questions and limitations

In this sub chapter, the thesis objectives are introduced, research questions, as well as limitations that will occur during the process.

The project at the end will come as an assistance for the key points of these questions. In order to reach that target, the research will analyze some popular framework models using online articles about user opinions and peer review of the author. These analyses will contain detailed information, instruction, differences and affection of each of these frameworks.

However, since there are numerous distinct web development frameworks available at the moment, only popular and well-known frameworks will be chosen to be analyzed, due to their comprehensiveness.

Also, defining a research question and sub questions are very important to start the thesis. The main purpose of the research is to answer this research question:

How to apply web development framework to increase website quality?

To fully answer the above question, I will divide it into 3 different sub questions which stand for 3 different parts of the thesis:

1. What is a web development framework?

2. How web frameworks enhance website quality?
3. How popular are web frameworks nowadays?

There are three limitations to this thesis. The biggest limitation in this case is also a big advantage for the author: statistics given in this research were highly affected by the world's current trends on web application development. Therefore, these numbers can change in a short time. Since JavaScript frameworks are dominating other languages, developers with JavaScript mastery can adopt a new, better and optimal framework in a short period. The second limitation is: the choices may vary according to regions. We can see that people from different regions choose different web frameworks. If this research were made in another region, the survey result might change and give distinct statistics and answers. And the last limitation is that surveyed people were handpicked, and the survey was only able to spread to a small amount of respondents

1.3 Theoretical framework

Theoretical framework of the research will be presented in this sub chapter.

To answer the research question, Angular, React and Vue's definition, history will be presented. Additional to that, using the three frameworks as one of the components to implement three small projects to demonstrate how they work.

1.4 Research methodology and data collection

For this sub chapter, research methodology and data collection will be presented.

There are two main types of research approach: the deductive reasoning and the inductive reasoning. The thesis will follow the deductive approach, which starts with the theories that will be tested by collecting data.

For research methodology, both quantitative methods and qualitative methods will be implemented. Quantitative methods will be used when conducting multiple-choice questions in a survey. Qualitative data is obtained by an open question in the same survey. By this method, both quantitative and qualitative research methods will be utilized.

Quantitative research	Qualitative Research
Focuses on testing theories and hypotheses	Focuses on exploring ideas and formulating a theory or hypothesis
Is analyzed through math and statistical analysis	Is analyzed by summarizing, categorizing and interpreting
Mainly expressed in numbers, graphs and tables	Mainly expressed in words
Requires many respondents	Requires few respondents
Closed (multiple choice) questions	Open-ended questions
Key terms are: testing, measurement, objectivity, replicability	Key terms are: understanding, context, complexity, subjectivity

Qualitative vs. quantitative research

Figure 1: Qualitative vs Quantitative (Streefkerk 2019)

As we can see in the above table, qualitative and quantitative allow user to answer different kinds of research questions (Streefkerk 2019).

After acquiring methods, data collection process will start. Primary data will be collected via the survey created online. For secondary data, these will be acquired through relevant online resources. To understand more about web application frameworks and the process of integrating a website through that frameworks, the research uses various information sources from articles, posts, online courses and author's personal knowledge, combining with data collected through books and online forums and communities.

The research will continue with gathering information from many developers from many fields (web, program, game, both frontend and backend)

The data for the introduction and for the research were collected by the author through online articles, newspaper and researches in the industry, along with common and basic understanding in the field gathered through years of studying.

Along with the data collection, researches involving many cases where exist the affection of web frameworks will be made (for example: popular sites created using frameworks, comparison how frameworks affect developer's work...). This is the comparative research method. Comparative research analyzes and compares two or more object or ideas to demonstrate the ability to examine, compare and contrast subjects (Bukhari, 2011).

1.5 Thesis structure

Step	Content	Outcome
Gather information	Find ideas and opinions from different sources and different people	Coming up with research questions
Frameworks history	Gather information about the history of web frameworks	To have an insight view and understand the concept of web frameworks
Web frameworks analyzing	List down multiple frameworks and analyze these frameworks	Examination and evaluation of multiple frameworks
Web frameworks comparisons and discussion	Comparison between frameworks, pros and cons and discussions.	Point out most suitable framework
Creating the survey	Survey contain multiple-choice questions and open questions	Qualitative and quantitative data collection
Conclusion	Analyze website features and come up with conclusion.	Conclusion

Table 1: Structure of the thesis

2 FRAMEWORKS OVERVIEW

2.1 Frameworks introduction

What stands behind and powers a great website? The interface? The responsiveness? The algorithm? The answer is simple: all of them, plus more. To create a smart, responsive and attractive website to user, vanilla HTML, CSS and JavaScript might not be the optimal choice, due to the variety and complexity of nowadays web designing trends. That is why developers need convenient web frameworks.

Web frameworks, or web application frameworks, are a common concept for those who works in the web programming and development field. These frameworks transformed the world of programming and become vitally important in web development process, bringing many advantages to the table. As the name, these “frameworks” work as the inner workings to the website, a solid backbone that connect every component. A website is usually split into 2 core parts: the front end and the back end.

2.1.1 Front end

Front end is the design or the interface of the web, also known as “the client side”, “the UI” ... (Pluralsight 2015). The front end is created using tools like HTML, CSS, JavaScript. This is the closest part to the user, where all interaction between user and website happens and all elements are under the browser’s control. As you can see in figure 2,



Figure 2: Front-end vs Back-end (Liverty 2018)

the difference between front end and back end is distinct. Front-end side is brighter and have better visual since this is “face” of the application where user interacts. On the other side is the back end where code and algorithms happen. Therefore, we need two developer with different jobs and tasks in order to build a fully functional web application.

The one in charge of a website’s front end is called *the front-end developer*. A front-end developer is responsible for the code that implement the client side and the architecture of the website’s immersive user experience. A designer creates graphical elements, a writer creates text elements and the front-end developer put together all those pieces, convert

them to lines of code in order to craft up a website. Alongside with basic languages like HTML, CSS, front-end developer needs to be able to utilize frameworks like Bootstrap, Backbone, Angular (Wales 2014). To be considered a front-end developer is quite complicated since this career is not taken to a level to make a degree in any college. Therefore, most of front-end developers are self-taught or is a crossover from designer or programmer (Lindley 2019)

2.1.2 Back end

If front-end developers are in charge of the exterior functions of a house, back-end developer controls the house's wiring and plumbing: server, application or database. The back end, also known as "the server side".

Unlike the front end, back-end developer uses PHP, Python or Ruby languages and different tools like MySQL, Oracle to build and maintain back-end's element (Wales 2014). They are more logical and systemic, in contrast to front-end developer's tasks.

Here in the first part of the research, before we dive into some of the most well-known frameworks nowadays, we will take a quick peek at the history of web development history.

2.2 History of web frameworks

In this subchapter, we will get to know more about the general history of web application frameworks.

At the beginning of the age of information, the existence of HTML, CSS or JavaScript is already rich and thick that could easily fill up a whole book. But the most notable events were occurred in the last 30 years, according to Web Design Museum, with the birth of WWW. In 1991,

Sir Tim Berners-Lee, or TimBL at that time, the genius physicist, internet pioneer and computer scientist after shocking the world with his greatest creation – the World Wide Web – the first browser, made public the world's first HTML specification. This first version of HTML supported only text and consisted of just 18 tags. In 1994, the first HTML validator were created by software engineers Dan Connolly and Mark Gaither to check the validity of documents. Two years later, in 1996, Håkon Wium Lie introduced CSS to become a describing presentation of a document written in a markup language. Two years later, the first version of HTML to support CSS were published, the HTML 4.0. Since then W3C has actively deprecated other approaches to style HTML documents. It is certain that you can



Figure 3: HTML, CSS and JavaScript (Dudkin 2016)

style your page with HTML's `<style>` tag or even use inline styling method but using CSS to style the document is an optimal solution.

Speaking of W3C, this is the acronym for World Wide Web Consortium. The organization was founded by Tim Berners-Lee at Massachusetts Institute of Technology, Laboratory for Computer Science. The organization's task is to oversee development of the web technology by creating standards.

JavaScript was created in an accidental circumstance, or at least the name "JavaScript". In 1990s, a war between Microsoft's Internet Explorer and Netscape's NetScape Navigator broke out to gain authority in the usage share of web browsers (Brown 2018). In 1995, a Netscape employee, Brendan Eich, was given a task to create a small, lightweight scripting language for easier programming. After ten days, Eich created Mocha and its job is to create and control dynamic web components. After changing name a few times with no modifications at all, the language was officially known as *JavaScript*. JavaScript evolved and became a standard language for web development. The adoption of JavaScript was slow at the beginning, but after the Mozilla Foundation acquired Netscape's browser code, JavaScript' popularity grew and later became the world's most popular web coding language, chosen by the majority of developers (Brown 2018). Nowadays, HTML, CSS and JavaScript are running as the cornerstone of all popular web frameworks. And for that, let's find out what, when and why are these so important to the modern web technology.

2.2.1 CSS and CSS frameworks

So, for just 5 years, from 1990 to 1995, the world witnessed the birth of HTML, CSS and JavaScript. The follow up decade is more chaotic and turbulent. Web technologies experience a huge change. Browsers at that time were not compatible with the standardized specifications of CSS (or even HTML and JavaScript). This leads to a big mess if a developer tries to open his website on a different browser. The confusing problem even happens to Internet Explorer, or IE, and carries on for the next few years and does not seem to have any solution. An organization named *Web Standards Project* created a series of tests called ACID. When a webpage implemented correctly with CSS; an iconic image will appear in browser.



Figure 4: ACID test 2's image indicate good CSS.

Thirteen years and countless modifications later, in 2011, CSS 2.1 became the proposed recommendation. Although level 2 CSS was born in 1998 and were recommended by W3C, many features of it caused some delays to the completion of the new revision. After that, in CSS level 3 and 4, the previous level's specification will be analyzed and become a series of independent modules. This process has immensely helped vendors to catch up faster with the latest recommendations at modular level (Wanyoike 2018).

In the mid-2000s, CSS frameworks and libraries emerged, introducing some early grids to help developer lay out their contents. These frameworks supported most browsers, so they are widely used due to their multi-platform convenience. Some of the popular names are *Blueprint*, *YAML YUI*, *960*.

Forms

Simple sample form

Text input (title)

Field with class .title

Another field

Field with class .text

Textarea

This is a <div> with the class .error. [Link](#).

This is a <div> with the class .notice. [Link](#).

This is a <div> with the class .success. [Link](#).

Select, checkboxes, lists

Select field

Ottawa

Select with groups

Favorite pet

Radio buttons

Radio one

Radio two

Radio three

Checkboxes

Check one

Check two

Check three

Figure 5: Blueprint default form styles (Storimer 2008)

2.2.2 JavaScript and JS frameworks

Just like CSS, JavaScript also experiences some browser compatibility problems in its early days. Developers had to modify their programs so they can fit on different browsers. After a few years, in 2006, the first JavaScript library, *jQuery* were released. Browser issues are solved with jQuery, this makes the libraries so popular at that time. Since then, many libraries and frontend frameworks appeared and made some confusion for developers. In 2012, Twitter created *Bower*, a package manager platform, originally released as part of Twitter's open source effort. Bower was used to fetch and download dependencies that were located all over the web. (Wanyoike 2018.) Bower became popular since it helps developer get their needed packages in just one place. Soon later, a problem was found with Bower. Web development needs both front and back-end implemented properly to work functional and Bower only host front-end packages and dependencies. Because of this problem, Bower soon become less trusted. Many other package managers like *Yarn* or *webpack* got more popular. In 2014, a software registry called *npm* was released. *Npm* soon became the world's largest central repositories for packages. Currently, Npm is the host for more than 800,000 code packages, both frontend and backend (w3schools 2019). For JavaScript frameworks, jQuery was the most popular JavaScript libraries at that time or even became a backbone for many other libraries (Guo 2019), though it still lacks efficiency when developers try to handle data, especially when implementing in a shared view. Various replacing JS frameworks were developed to fix this flaw. *Backbone*, *Knockout*, *Ember* are some notable names, they also gained some popularity at that time for their sufficiency over jQuery.

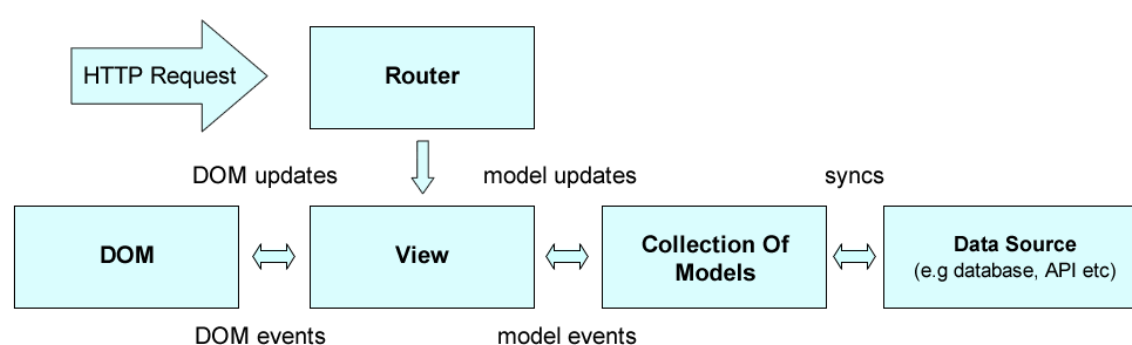


Figure 6: Basic structure of Backbone framework (Osmani 2012 - 2013)

Backbone was known as the first framework with the primary target is to create single page applications. Backbone was an independent framework, functional without requiring jQuery.

3 WEB FRAMEWORKS

3.1 Generic web frameworks history

Web frameworks come in all shapes and sizes, and these frameworks have become an essential part of web development (Goel 2020). Each framework was created with their own language: we have Java within *Vaadin*, *Javalin*, *Grails...*; C# within *ASP.NET*, *Blazor* and PHP language within *Laravel*, *CodeIgniter* or *Symfony*. As we see, web development frameworks are various and merely based on developer's choice for coding language. In this research, I will focus on introducing and analyzing only Angular, React and Vue frameworks since they show their excellency and sufficiency over frameworks created from other languages. Solid JavaScript framework should have its own ecosystem. (Sviatoslav 2020.) These frameworks provide tons of interesting (and essential) things to your website, including components, routing... The upside of using JavaScript frameworks is the comprehensive efficiency and code organization – your code might get a bit messy at first but will be arranged into a clean structure. In addition to that, JS frameworks also provide many built-in solutions for common coding errors. Therefore, JS frameworks are the optimal choices for web development.

In the next part, we will be reviewing three most popular web development JS frameworks these days: Angular, React and Vue. These frameworks share a similarity: they build projects using components. In each subtopic, I will introduce a brief history of the frameworks' origins, followed by the basic and core features of that framework. To install these, developer needs to set up *node.js*. Node.js is a JavaScript runtime built on Chrome's V8 JavaScript engine. In short, Node is a JavaScript interpreter without a browser attached, created to develop JavaScript applications that runs on various operating system like Windows, Linux, Mac (w3school, 2020). Node.js is not mandatory but it makes the process easier by creating a lightweight web server for your application. Npm will be installed alongside with Node.js. Therefore, you will be able to access npm's online database containing over 800,000 code packages.

3.2 Angular

In this subchapter, Angular's brief history and basic installation instruction will be presented.

3.2.1 Brief history

Angular was first released in 2010 and is the oldest of the three. In 2008 and 2009, Miško Hevery, a Google employee was developing a project to help making web application easier for web designer and web developer (Gudelli 2019). Hevery and his friend Adam Abrons named the project AngularJS. The project name was based on angular brackets in HTML. After a few modifications and some application created with AngularJS, the project was released as an open source in 2010. In May 2011, after some delayed, Angular Js v1.0 was officially released. Angular was written with TypeScript language, a superset of JavaScript that will compile to JavaScript (Bright 2012), developed and maintained by Microsoft.

The community was taken by storm and started to build applications with AngularJS. Since AngularJS is open source, anyone can browse and alter it to fit their need, from small-project-grade usage to enterprises level usage. Many big names like Deutsche Bank or Google and GitHub started to utilize AngularJS to develop their own application and web environment. (Gavigan 2018.) But AngularJS created as a small internal project and was not intended to serve these purposes. A few years later, AngularJS was a little left behind due to the emergence of JavaScript new advancements and standards, therefore, new demands.

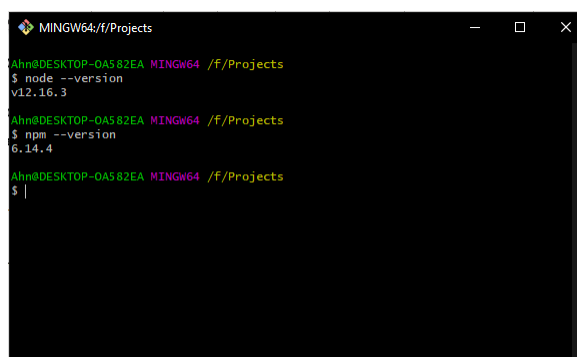
The Google team decided to develop Angular 2 (or Angular 2.0). The idea is to build a new framework that could solve bigger problems occurred when implementing large project or cross-platform applications, which was struggled by its predecessor. A total ground-up re-write was executed, led to the birth of Angular 2. Over the years, Angular keeps on developing to become one of the most optimized, and chosen-by-many, web development frameworks.

Now in 2020, Angular has developed to Angular 9, which was released on February 6, 2020.

3.2.2 Implementation

To start a project using Angular 8, developer need Node 8.9 or later and Npm 5.5.1 or later installed on their device (Boucheфра, 2019). To check your Node version, use these commands:

```
$ node --version
```



```
MINGW64/f/Projects
Ahn@DESKTOP-0A582EA MINGW64 /f/Projects
$ node --version
v12.16.3
Ahn@DESKTOP-0A582EA MINGW64 /f/Projects
$ npm --version
6.14.4
Ahn@DESKTOP-0A582EA MINGW64 /f/Projects
$
```



```
$ npm --version
```

Figure 7: node and npm version checking

Next, we will install Angular CLI. Open the computer's terminal and execute the following command:

```
$ npm install -g @angular/cli
```

In this command, “npm install” is the command to access to npm's storage and install a code package. “-g” makes the installed package accessible from anywhere, not just the folder it was installed. And “angular/cli” is the package's name.

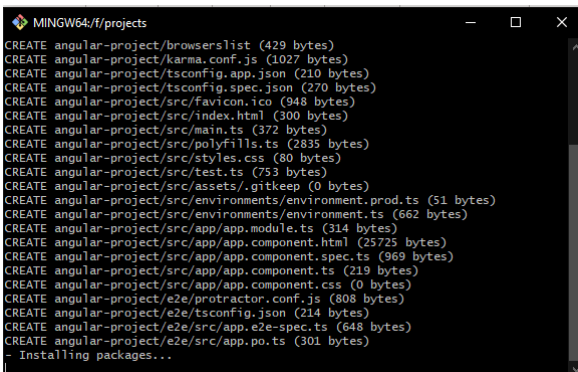
After Angular CLI was installed, we can start a new Angular project. You can open terminal to start the project directly at the folder or use command:

```
$ cd f/Projects
```

“cd” will redirect you to another folder, where you want to save your project. Then execute the command:

```
$ ng new angular-project
```

This is the command to start a new project named “angular-project”. In the next few minutes, packages needed for an Angular project will be installed.



```

MINGW64/f/projects
CREATE angular-project/browserlist (429 bytes)
CREATE angular-project/karma.conf.js (1027 bytes)
CREATE angular-project/tsconfig.app.json (210 bytes)
CREATE angular-project/tsconfig.spec.json (270 bytes)
CREATE angular-project/src/favicon.ico (948 bytes)
CREATE angular-project/src/index.html (300 bytes)
CREATE angular-project/src/main.ts (372 bytes)
CREATE angular-project/src/polyfills.ts (2835 bytes)
CREATE angular-project/src/styles.css (80 bytes)
CREATE angular-project/src/test.ts (753 bytes)
CREATE angular-project/src/assets/.gitkeep (0 bytes)
CREATE angular-project/src/environments/environment.prod.ts (51 bytes)
CREATE angular-project/src/environments/environment.ts (662 bytes)
CREATE angular-project/src/app/app.module.ts (314 bytes)
CREATE angular-project/src/app/app.component.html (25725 bytes)
CREATE angular-project/src/app/app.component.spec.ts (969 bytes)
CREATE angular-project/src/app/app.component.ts (219 bytes)
CREATE angular-project/src/app/app.component.css (0 bytes)
CREATE angular-project/e2e/protractor.conf.js (808 bytes)
CREATE angular-project/e2e/tsconfig.json (214 bytes)
CREATE angular-project/e2e/src/app.e2e-spec.ts (648 bytes)
CREATE angular-project/e2e/src/app.po.ts (301 bytes)
- Installing packages...

```

Figure 8: System creating project

Once the terminal finishes the installation process, you need to redirect to the file's root folder and execute

```
$ cd angular-project
```

```
$ ng serve
```

The “ng serve” command will start compiling the file and the project is good to go. A local server was created to display your project's interface and is accessible by typing “localhost:4200” in your browser search bar. Below is the project's interface (left) and project's files in Visual Studio Code editor (right).

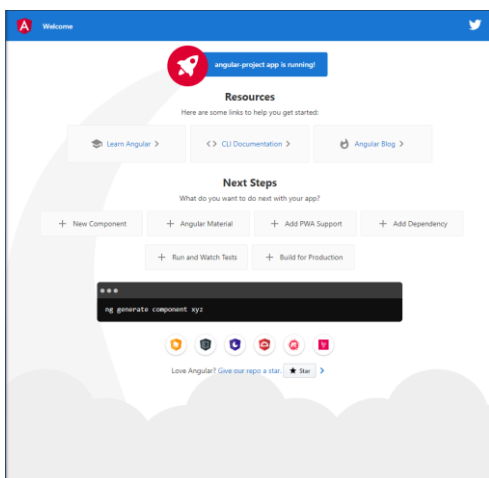


Figure 10: Project UI

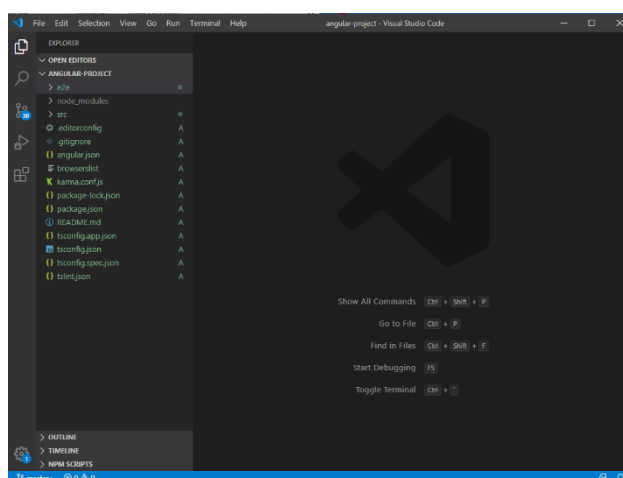


Figure 9: Project in a code editor

For more information and tutorial about Angular, please visit: <https://angular.io/>

3.2.3 Generic features

One of Angular's basic feature is cross-platform development. Developers can build up progressive web applications (PWA), which creates an app-like testing and adjusting environment on a modern web browser. This works for both mobile application and desktop or laptop applications.

Angular CLI itself is considered a useful feature of Angular. With CLI, you can customize your application with just simple commands

Another effective feature is Angular's two-way data binding system: data exchange from the component to view and from view to the component. This helps user establishing communication in a two-way factor. To use two-way data binding, developer needs to use “ngModel” syntax.

```
<input type="text" [(ngModel)] = 'val'>
```

For this example, ngModel was used to bind to the input field with a variable named 'val'. This will change the value of the variable based on the input field's value.

3.3 React

In this subchapter, React's brief history and basic installation instructions will be presented.

3.3.1 Brief history

React, or React.js or ReactJS, is an open source JavaScript library created by Jordan Walke to focus on building user interface on single-page applications and is used for handling the view layer for websites and mobile applications (Pandit 2020). In 2011, John Walke created FaxJS to develop search element on Facebook. FaxJS is considered the prototype of React. On May 29th, 2013, React was initial released, got open sourced and became famous in the community. (Hámori 2018). Facebook is currently in charge of maintaining and keeping React up to date. And since this is an open source library, the community's contribution is taken credit for React's development. Due to its ecosystem and versatility in extensions, React is usually mistaken as a full web development framework. Both framework and library are code written and reusable. The main difference is named as a term: The inversion of control.

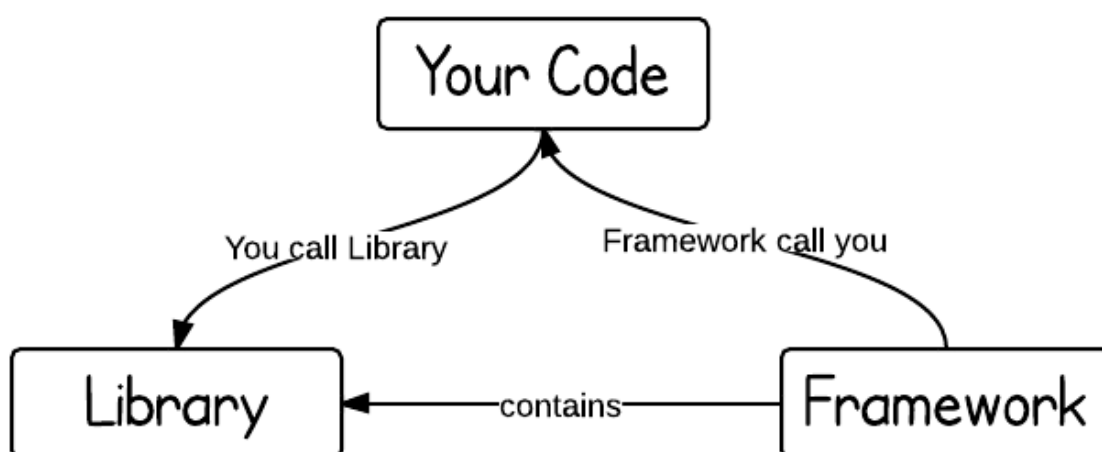


Figure 11: Connection between library and framework (ProgramCreek 2018)

In the above illustration, the framework is in charge of the flow and provide room for the code and will call this code whenever needed. But in a library, developer is responsible for

the flow of the application and will call the library whenever needed. (Wozniwicz 2019.)
React's latest version is 16.13.0.

3.3.2 Implementation

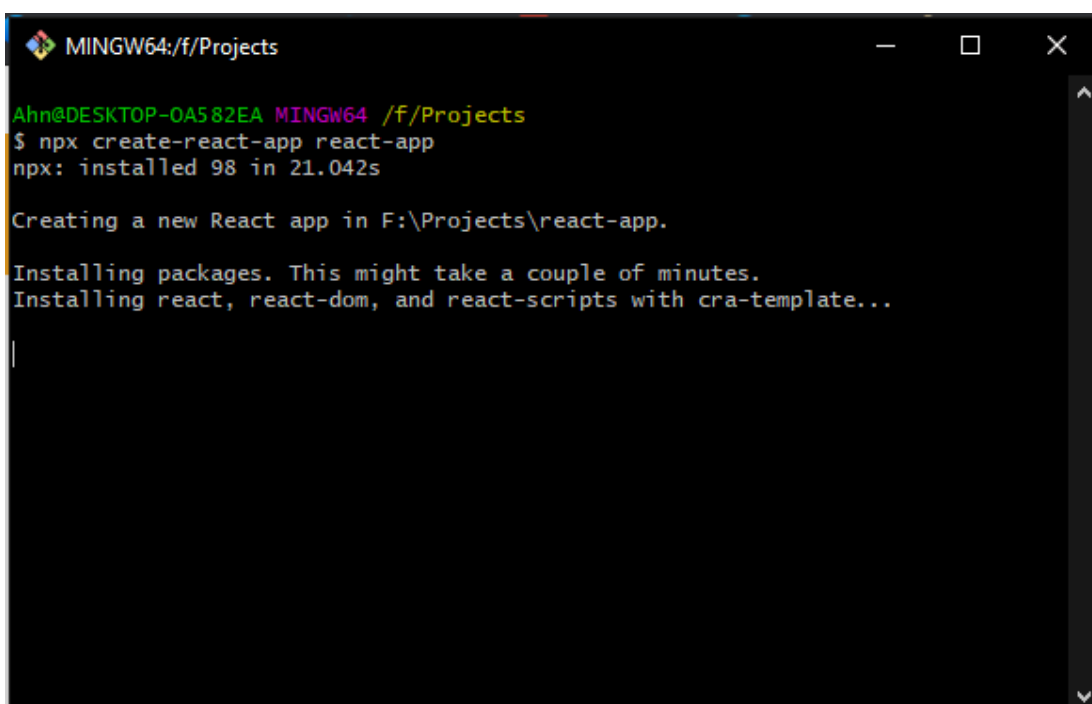
Since React is a versatile JavaScript library, and is not restricted as a framework, there are many ways to integrate React.

First, we can use "Create React App". This is a useful and easy environment for creating a simple React application project. Like Angular, React required installation of Node (8.10 or higher) and Npm (5.6 or higher).

Browse your project folder and run terminal. Execute commands:

```
$ npx create-react-app react-app
```

`npx` is a binary added alongside with `npm`. While `npm` is used to install packages, `npx` is used to execute packages. In this occasion, `npx` will execute `create-react-app` package to create a new project named `react-app`.

A screenshot of a terminal window titled 'MINGW64:/f/Projects'. The prompt is 'Ahn@DESKTOP-0A582EA MINGW64 /f/Projects'. The user enters the command '\$ npx create-react-app react-app'. The terminal shows the following output: 'npx: installed 98 in 21.042s', 'Creating a new React app in F:\Projects\react-app.', and 'Installing packages. This might take a couple of minutes. Installing react, react-dom, and react-scripts with cra-template...'. The terminal has a dark background with light-colored text and a scrollbar on the right side.

```
MINGW64:/f/Projects
Ahn@DESKTOP-0A582EA MINGW64 /f/Projects
$ npx create-react-app react-app
npx: installed 98 in 21.042s
Creating a new React app in F:\Projects\react-app.
Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...
```

Figure 12: System creating React project

Then execute this command to redirect to your project and run development local test server:

```
$ cd react-app
```

```
$ npm start
```

After successfully compiled, the test server is accessible at <http://localhost:3000/> on your browser. Below is the image of the application's default interface (left) and the project's files (right).

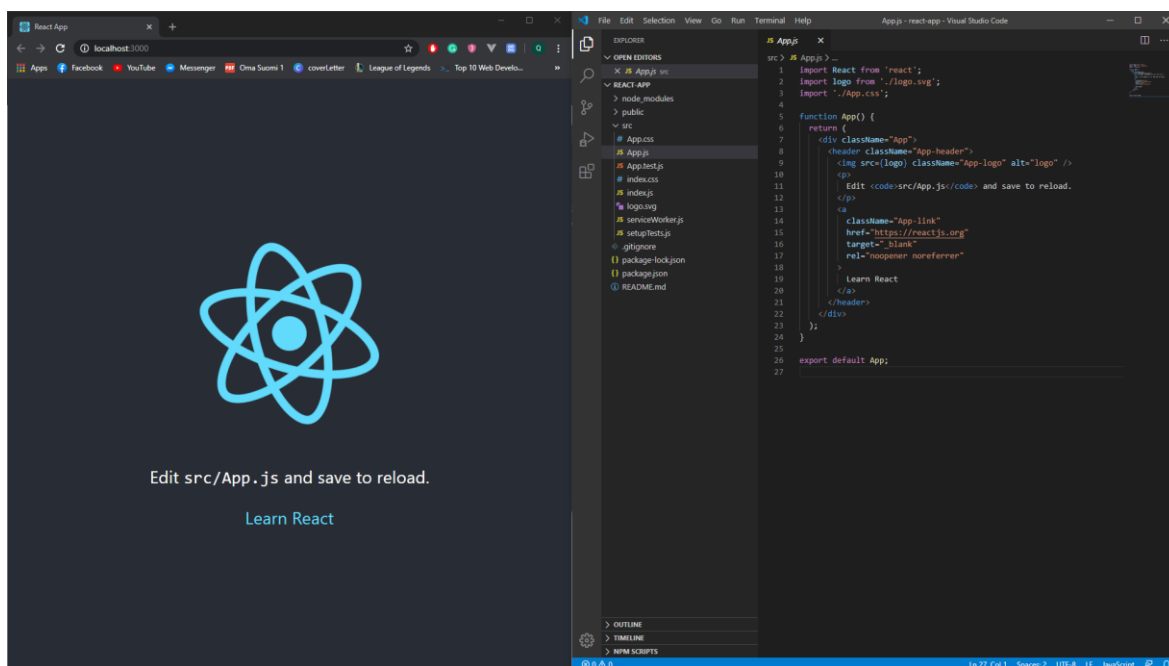


Figure 13: Default React application

For the second method, we can simply add React as a plain `<script>` tag on a normal HTML page. In the HTML file that needs editing, create a new `<div>` tag with an `id` within the `<body>` tag

```
<div id="button_container"></div>
```

This method is called creating a DOM (Document Object Model) containers and you can have as many containers as needed (reactjs.org). Then add 2 more `<script>` tags

```
<script src="https://unpkg.com/react@16/umd/react.development.js" crossorigin></script>
```

```
<script src="https://unpkg.com/react-dom@16/umd/react-dom.development.js" crossorigin></script>
```

The first `<script>` tags will integrate React in your HTML file. Then, in the folder, create a new JavaScript file and name it similar to the id of the DOM container.

For more information and tutorial about React, please visit <https://reactjs.org/>

3.4 Vue

This subchapter presents Vue's brief history and basic installation instructions.

3.4.1 Brief history

Vue.js, or Vue, is a progressive front-end web application framework created for building user interfaces and single page application (Vue.js 2019). Vue's initial release was February 2014. This makes Vue is the youngest of the three frameworks. Unlike Angular and React, Vue started out just an independent project and not created within a big company. Evan You, Vue's founder, developed the initial version before Vue was adopted by the community. Vue's latest version is 2.6.11 and is updated at the end of 2019.

3.4.2 Implementation

Similar to React, there are two ways to integrate a Vue project: start a whole new Vue project or just simply inject Vue into a HTML file with `<script>` tag.

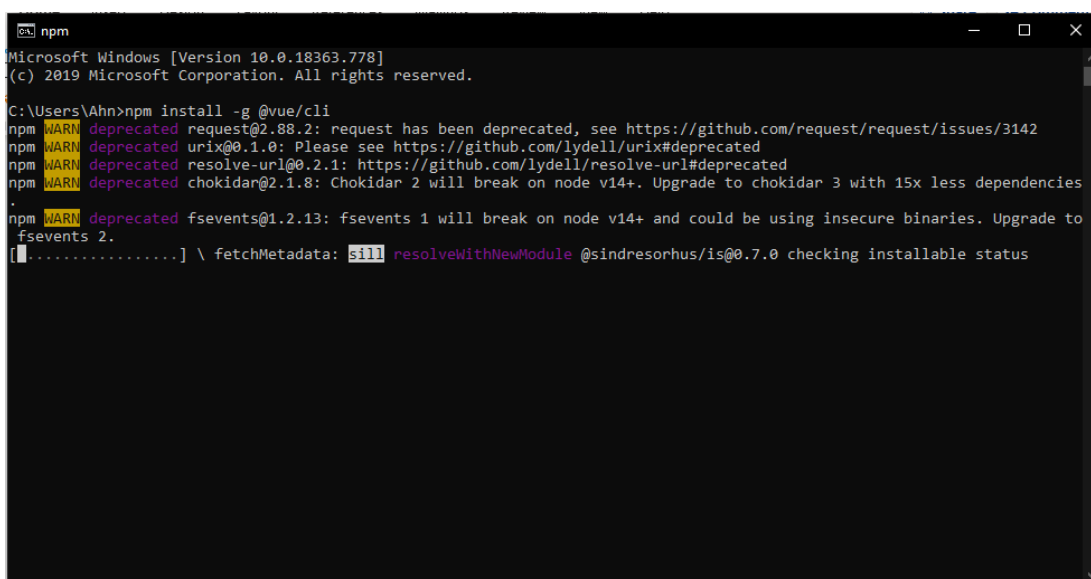
First method is to use Node to install Vue CLI to start a new Vue project.

Open terminal and execute commands:

```
$ npm install -g @vue/cli
```

This command will install Vue CLI exactly the same as Angular CLI since both of them are packages on npm.

Vue will be installed after executing these commands.



```
npm
Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

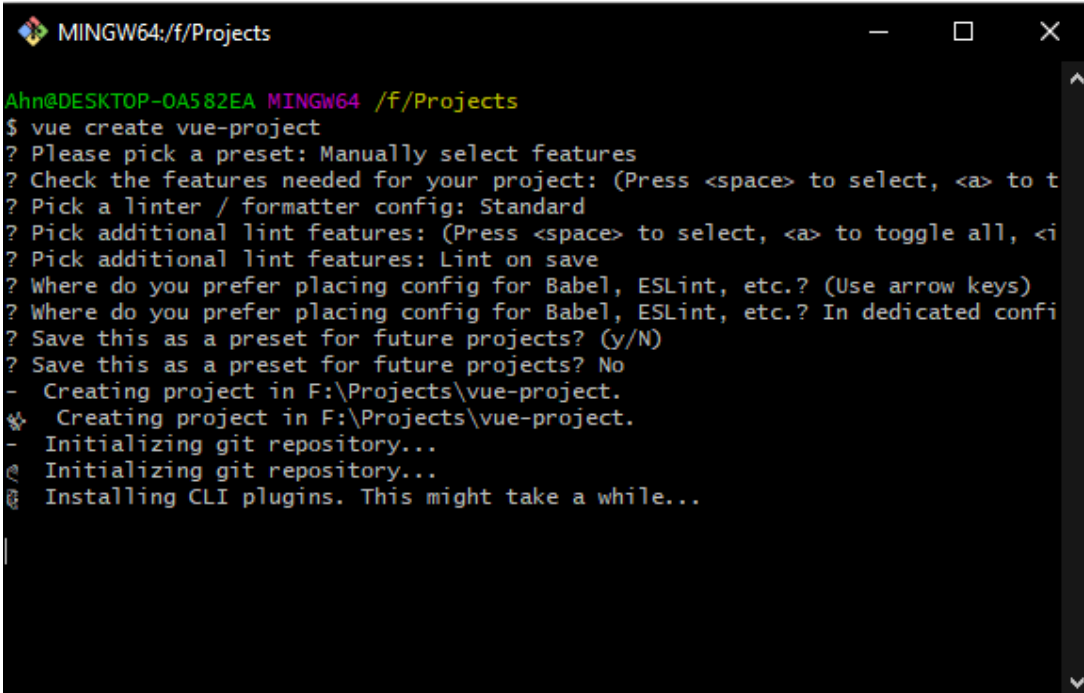
C:\Users\Ahn>npm install -g @vue/cli
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated
npm WARN deprecated chokidar@2.1.8: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies
.
npm WARN deprecated fsevents@1.2.13: fsevents 1 will break on node v14+ and could be using insecure binaries. Upgrade to
fsevents 2.
[.....] \ fetchMetadata: sill resolveWithNewModule @sindresorhus/is@0.7.0 checking installable status
```

Figure 14: Vue CLI installation process

After installing Vue CLI, we can execute commands to start a new Vue application:

```
$ vue create vue-project
```

The terminal will let user choose whether to do default or manual. Since this is a simple and basic illustrating project, we will choose default.



```
MINGW64:/f/Projects
Ahn@DESKTOP-OA582EA MINGW64 /f/Projects
$ vue create vue-project
? Please pick a preset: Manually select features
? Check the features needed for your project: (Press <space> to select, <a> to t
? Pick a linter / formatter config: Standard
? Pick additional lint features: (Press <space> to select, <a> to toggle all, <i
? Pick additional lint features: Lint on save
? Where do you prefer placing config for Babel, ESLint, etc.? (Use arrow keys)
? Where do you prefer placing config for Babel, ESLint, etc.? In dedicated confi
? Save this as a preset for future projects? (y/N)
? Save this as a preset for future projects? No
- Creating project in F:\Projects\vue-project.
* Creating project in F:\Projects\vue-project.
- Initializing git repository...
* Initializing git repository...
* Installing CLI plugins. This might take a while...
```

Figure 15: Creating Vue app using Vue CLI

After the process is finished, execute these commands to redirect to the project's folder and deploy test server (this is recommended in the terminal after project was created):

```
$ cd vue-project
```

```
$ npm run serve
```

After successfully compiled, the application will be running on deployment server <http://localhost:8080/>. Below are the images of the application's default user interface and the folder in Visual Studio Code editor.

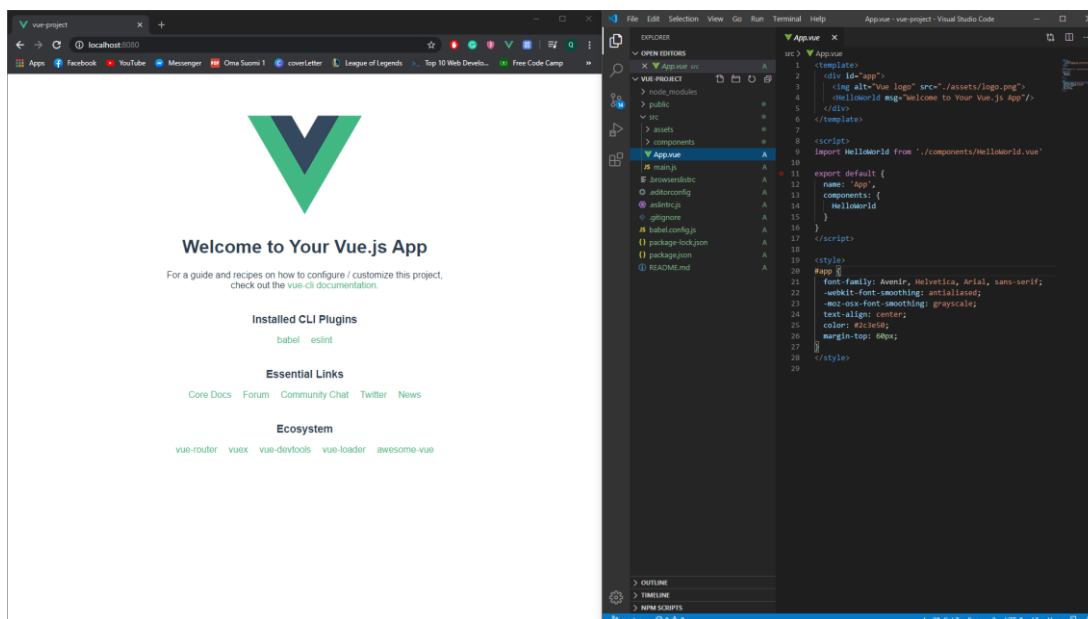


Figure 16: Default application

For the second method, we will inject Vue directly to a HTML file using `<script>` tag.

Similar to React, we will create a `<div>` element with a unique id working as a DOM container.

```
<div id="app"></div>
```

Then create a new `<script>` tag. This is where data will be linked

```
<script>
```

```
  const App = new Vue({
```

```
    el: '#app',
```

```
    data: {
```

```
      text: 'Anh Le',
```

```
    },
```

```
  })
```

```
</script>
```

Now we go back to the DOM container and render `text` using double curly brackets

```
<div id="app">{{text}}</div>
```

For more information and tutorial about Vue, please visit: <https://vuejs.org/>

3.5 Frameworks comparison

In the earlier part, we have reviewed all three frameworks' history and basic implementation. Each framework has a different background as well as implementation methods. Therefore, in this part, we will discuss their pros and cons and come up with a practical comparison.

*Note: The pros and cons below are from the author's perspective after experiencing and implementing project with these frameworks, as well as personal's knowledge and sources.

3.5.1 Pros and cons of Angular

- Advantages and distinguished features:
 - Angular utilizes TypeScript, which is said to be an updated version of JavaScript due to its supplementation with static type options and object-oriented classes, as well as containing ES6 – the latest version of JavaScript (Tran 2018).
 - Enable using dynamic data instead of static HTML to create a web application. (James 2019)
 - Created as an updated version of AngularJS specifically to handle large, enterprise-level projects.
 - Has specific update release schedule (Google 2020).
- Disadvantages:
 - Developer has to get familiar with TypeScript. This might not take too much time but still a time-consuming task.
 - Various different structures that might cause confusion in learning process.
 - Larger size and slower performance (when comparing to Vue and React).
 - Developer have to re-write the whole project if want to adopt another development framework. (Schwarz Müller 2020.)

Angular is now the backbone of many companies' digital infrastructure: Google, Microsoft, AT&T, Apple, Nike, PayPal...

3.5.2 Pros and cons of React

- Advantages and distinguished features:
 - Utilizes JSX, a syntax extension of JavaScript, for templating (optional).
 - Detailed documentations.
 - Lightweight and simple design. Excels in speed
 - Develop their own application generator “create-react-app”. This action omits an extra step in generating a new project or application.
 - Supported by Flow – Facebook’s open source type checker dependency
 - Dominating other frameworks, therefore, offer better working opportunity when mastered. (TechMagic 2018.)
- Disadvantages:
 - React is splitting class-based components. This might become an obstacle for traditional object-oriented programmers.
 - Although JSX is optional, it can cause confusion for the mixture of templating and logics, functions.
 - Developer have to re-write the whole project if want to adopt another development framework. (Schwarz Müller 2020.)

Due to many advantages and features, React is now the world’s dominating web development frameworks. Companies utilizes React: Facebook, Instagram, Netflix, WhatsApp, Airbnb...

3.5.3 Pros and cons of Vue

- Advantages and distinguished features:
 - Complexity level is low (containing only HTML, CSS and JavaScript). This makes Vue the easiest framework to learn (Schwarz Müller 2020).
 - Vue shares many similarities with Angular and React in term of design, architecture and syntax. This creates Vue’s adaptability and provide a quick switch between frameworks (TechMagic).
 - Even lighter than React (about 60 KB total), equal faster downloading time than the two competitors.

- Can easily integrate from small, simple single-page applications to complex website user interface and mobile application.
- Disadvantages:
 - By not being supported by big names, Vue is still lack of resources and share only a small market comparing to React and Angular.
 - Need additional tools to integrate data from the server.
 - Developer have to re-write the whole project if want to adopt another development framework. (Schwarz Müller 2020.)

Companies utilizes Vue for their digital infrastructures: Alibaba, Grammarly, Xiaomi, GitLab, ...

Feature	Angular	React	Vue
UI / DOM Manipulation	✓	✓	✓
State Management	✓	✓	✓
Routing	✓	✗	✓
Form Validation & Handling	✓	✗	✗
Http Client	✓	✗	✗

Figure 17: Technical comparison for three frameworks (Schwarz Müller 2020)

3.5.4 Conclusion

After a brief comparison, we can see each framework possess their own benefits and downsides. Angular and React share a similar benefit of having detailed documentations. This helps new developers get to know these frameworks faster and more convenient, reduce time and effort for learning process. Vue is lighter and less complicated, therefore increase flexibility in implementation. That is why all three frameworks, and library, are very popular for their own advantages, as well as their own disadvantages. Choosing the best and most suitable framework is 100% base on personal preference or other factors at work.

4 STATISTICS BREAKDOWN

4.1 Survey statistics

In this research, a small survey about familiarity of other students about the three mentioned web development frameworks will be presented and analyzed.

The survey was sent to 20 IT students and IT-related developer, from inside and outside of Finland. They also helped the author by sending the survey to some friends of them. Therefore, the number of respondents were increased to 32.

4.1.1 Quantitative research

a. Data collection

For the sake of quantitative data collection, there are 3 simple multiple-choice questions sent to other students and developer whose jobs or study related to IT or web development. These questions were created to serve the purpose of measuring quantitatively how much these interviewees know about Angular, React and Vue frameworks

b. Data analysis

This sub-topic analyzes the data collected through quantitative data collection. The results were harvested through Google's automatically generated charts for the survey's creator. This function helps reduce time and effort since the author have the right to freely manipulate these charts and also helps readers find it easier to track and understand the answers.

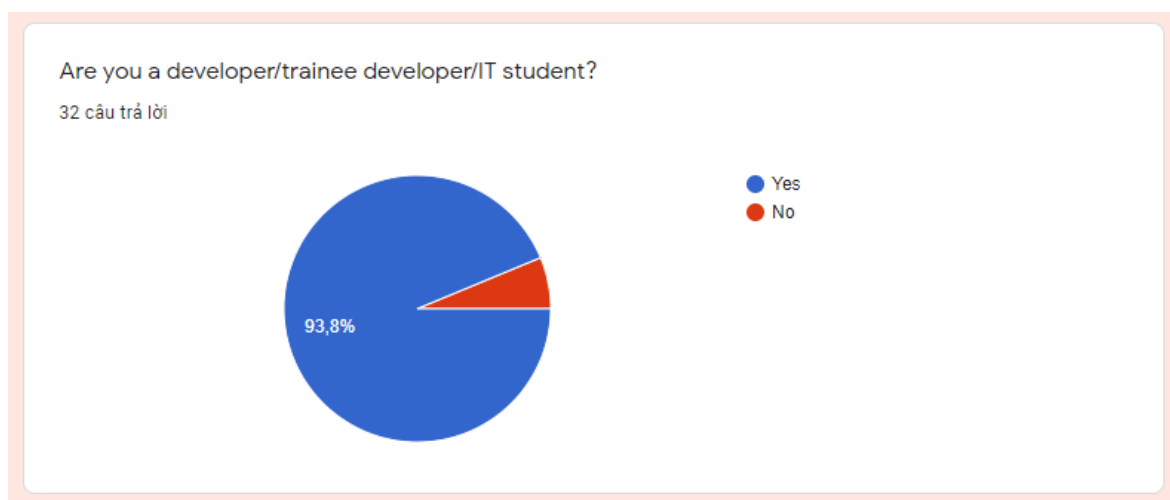


Figure 18: Respondents' background

The first question asks the interviewee if he or she is an IT-related person. Since the survey was sent selectively, 30 people answered “Yes”, occupy 93.8% out of total 32 respondents. That means the ones who return negative answer occupy 6.3%. The question’s result showed that the survey was not sent randomly. That would have caused a major interference to the flow of the data source.

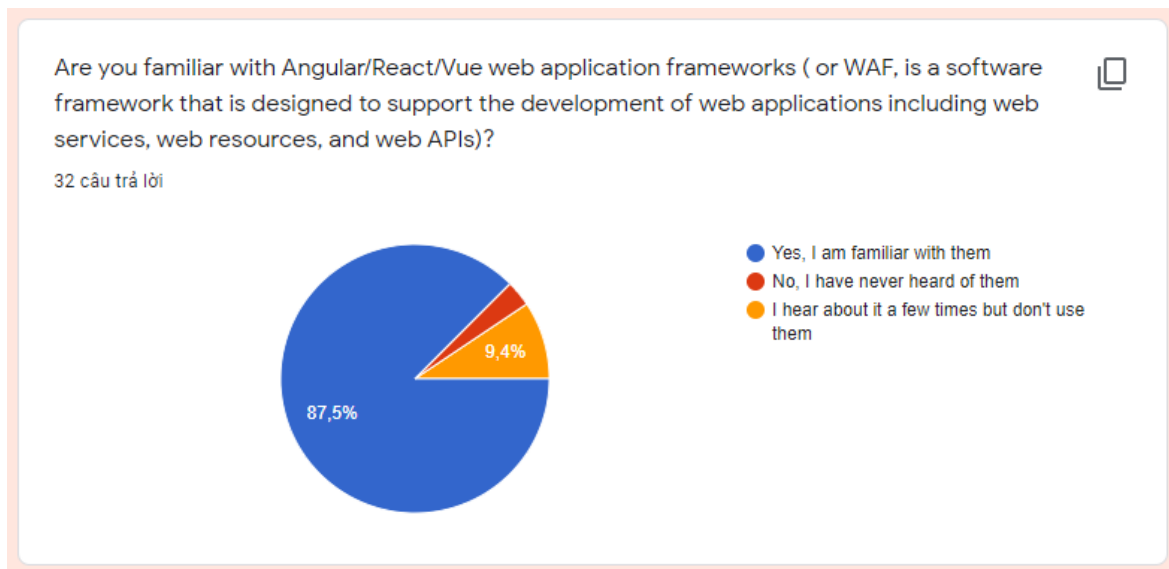


Figure 19: Respondents' familiarity with frameworks

The second question mentioned Angular, React and Vue – the three frameworks mentioned and introduced in the earlier part. In this question, respondent will answer their familiarity, or their knowledge about these frameworks. To make the question more understandable, the writer added a definition of web application frameworks. This clarification action helps reduce misunderstanding rate or prevent attendants from mistaken the terms. The result shows that 87.5%, or 28 attendants, are familiar with these frameworks. The rest of the respondents return a negative answer, being not familiar to these frameworks. One respondent, equals to 3.1%, has never heard of these terms and 3 persons, equals to 9.4%, might have heard of but do not use them.

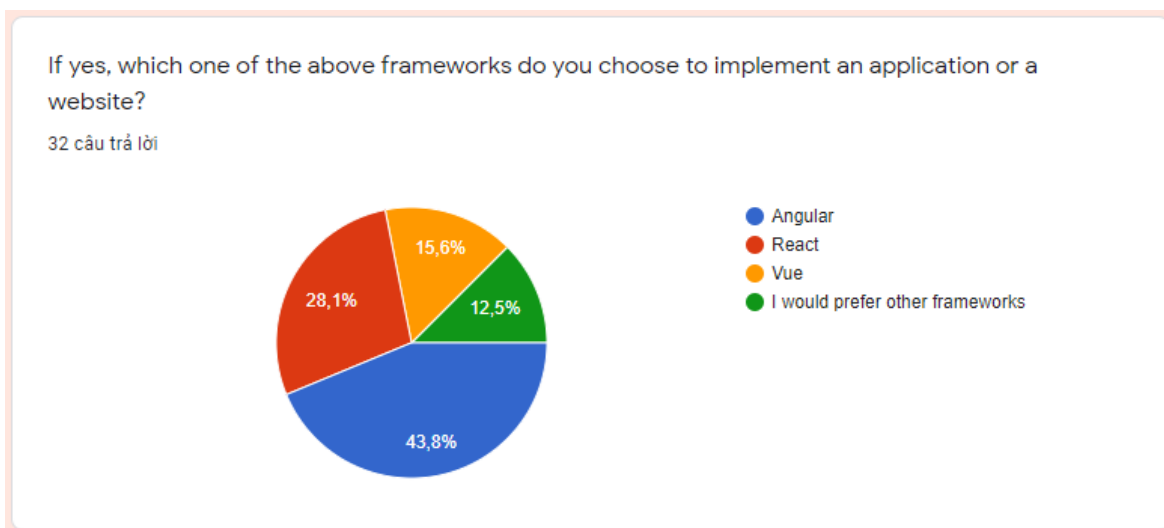


Figure 20: Framework preferences

The last multiple-choice question let attendants choose one of three frameworks given if they want to implement a project. 14 attendants – 43.8% chose Angular, 9 respondents – 28.1% picked React and 5 people decided to go with Vue. The rest – 12.5%, would prefer another framework. The statistics acquired through this question showed personal preferences of attending developers. Through these numbers, we found out that Angular was chosen the most, follow by React and Vue finishes last. This result is 100% personal preferences and is a little against current trends for web development since Angular is no longer dominating other frameworks. For the past few years, the emergence of React and Vue attracted most of the community's attention made Angular got left behind for the race for the most popular framework.

4.1.2 Qualitative research

a) Data collection

The next part of the survey is an open question where attendants can share their opinions on each or all of the chosen frameworks. Most of the answers are short and simple, but they are direct opinions from respondents, so all ideas are taken into consideration. The table below presents 30 answers from 32 respondents, 2 questions missing is the result of a mistake in the process of creating the survey: this question can be left unanswered.

1. <i>I like Vue for its lightweight and design.</i>
2. <i>I am not an IT student or developer.</i>
3. <i>I heard of Angular so I think it would be a good one</i>

4. <i>VueJS style is pretty simple when html, js, and css are used seperately</i>
5. <i>I am most familiar with Angular framework due to its high speed and optimal performance</i>
6. <i>React has many advantages over the other two in the list. It is better in rendering data. React also possesses JSX which stores logic and markup in just one file. Coding with React is also more simpler than that in other frameworks especially Angular</i>
7. <i>I am using Flutter</i>
8. <i>About Angular, it is fast and it optimizes the development time</i>
9. <i>React is the best to create user interface, on my opinion</i>
10. <i>I personally think Angular is the best for frontend development</i>
11. <i>All of them are very smart and effectice frameworks in building apps or web-sites</i>
12. <i>I would choose Angular as it allows us to build highly interactive Single Page apps, which is more impressive feature than the other two frameworks</i>
13. <i>All are good but I am more familiar with AngularJS</i>
14. <i>React seems the best for me. Data rendering is the advantage</i>
15. <i>I choose Angular. It has outstanding technical features like two-way data binding, MVC architecture, less code framework, etc</i>
16. <i>React has virtual DOM</i>
17. <i>React is different because it has one-way data binding</i>
18. <i>Angular seems to be the best for me. It has testing frameworks and users can build accessible apps</i>
19. <i>All are good</i>
20. <i>No comment</i>
21. <i>ReactJS is the best choice when it comes to data rendering</i>
22. <i>It is easier to understand Vue.</i>
23. <i>I find using Angular very helpful as I can myself develop progressive web applications (PWA) and develop apps for desktop</i>
24. <i>Most poplular one</i>
25. <i>I use Angular the most and I think it is the best option. However, it depends on each person how how familiar they are</i>
26. <i>I like Angular as it has many built-in features</i>

<i>27. It is light, its file size is lighter than others</i>
<i>28. Coding with Vue is easier for me and it is getting more popular in the market now</i>
<i>29. All of them are fine</i>
<i>30. I like Vue because of its simplicity and cool design</i>

Table 1: Qualitative data.

b) Data analysis

First, to sort these comments, a coloring method was used: comments support Vue will be colored green, Angular is red, React will be painted blue and answer related to other frameworks will be painted orange.

After analyzing the table, we got:

Five answers were supporting Vue's design and features: comment 1, 4, 22, 28 and 30.

Eight attendants supported React's features: comment 6, 9, 14, 16, 17, 21, 24 and 27.

Eleven respondents believed that Angular is the best frameworks based on its features: comment 5, 8, 12 and 13. Comment 3 and 10 were painted red but they were based on respondents' feelings

Three attendants gave a neutral comment, stated that all of these frameworks are good for web development but do not go deep on the details (number 11, 19 and 29).

One attendant mentioned Flutter, a new web framework developed by Google specifically created to build application UI, instead of Angular, Vue or React (number 7).

Comment 2 and 20 are skipped since these comments do not provide any data for the research.

In conclusion, attendants to the survey give different answers but most of them is to support their choice that were previously made. They must be all aware of these frameworks in order to give precise answers. This part of the survey served its purpose to collect qualitative data and find out developers' choices and their feelings for these frameworks.

5 CONCLUSION

5.1 Answers to research questions

In the last part of the research are the answers to research questions and a quick reminder of reviewed and analyzed data mentioned in earlier parts. After that are differences between expectations and achievement of the thesis, as well as limitations occurred during the thesis execution.

The thesis contains three major parts, which serve as researches to identify answers for the given research questions in the first chapter:

- What is a web development framework and how to optimize?
- How popular are web frameworks nowadays?
- How web frameworks enhance website quality and benefit IT companies?

For the first questions, the third part reviewing information about history of web development methods has given a clear answer.

The survey was created as a method to answer the second question. Through the survey and its statistics, we got an insight view of how popular web development frameworks is nowadays. The answer for the question was also created by gathering information from outside sources

And for the last question, through the fourth part, web frameworks expressed many benefits to the web or application development process. First, developer can build and design a project on a stable and ready-made framework but not being restrained or shackled by any means. Web frameworks use component method – a simple, effective and optimal solution for developing web and application. These components are customizable, therefore decreases bugs and errors since developer can control each part of the application instead of re-write various lines of code to fix one small problem. Web frameworks are also fast and secure. This led to a major cutdown on implementing time and cost. And lastly, the first two frameworks, Angular and React are open-sourced, well documented and supported, by the community and by their mother companies with professional IT departments within each of them, except for Vue. This helps these frameworks are always supported, bug fixing and getting improvement every day.

With many features updating over time, web frameworks serve different developing type and methods or purposes, as well as their extensions or advancement. For a free solution, in my opinion, web development frameworks are optimal and highly recommended

5.2 Reliability and Validity

The main purpose of the thesis was to give an insight answer to the research question. Secondary and primary data source were obtained through online sources, the survey and the author's personal knowledge. The author was exposed to and had experienced these frameworks by creating many projects, as well as had been trained at a local start-up IT company. The author's tasks were to develop personal skills about the company's framework. Assisting tools for the research were popular, highly trusted and recommended.

5.3 Suggestions for further research

This research's purpose was achieved but there are always rooms further study or advancement:

- Adoption of back-end frameworks to increase project's result and quality.
- Developing website and application using Angular framework.
- Developing website and application using React framework.
- Developing website and application using Vue framework.
- Comparison between Flutter – a cutting edge UI framework, with older JavaScript frameworks (Angular, React and Vue).

These topics were recommended because they are the advancement for this research. Researching above topics will help the author acquire more information and knowledge about both old and new web application frameworks. Also, since this research is about front-end frameworks, a research about back-end frameworks, along with development instruction for web and application will create a thorough set of complete guides for developing web application.

5.4 Summary

The last decade witnessed the advancement of web and application development methods, and the digital world has been changing rapidly and non-stop ever since. Many web standards were adopted and change our everyday online experience: faster, smoother, more secure and more colorful. This led to a growth in adoption of platforms and tools that support and enhance web application building. Therefore, choosing a suitable framework to assist project implementation is important for any IT company or individual developer.

With reviews and practical statistics given, it can be concluded that using web application frameworks is important for business since digitalization is growing and exploding fast enough to be considered a trend. Also, the research archived the target and gave a clear answer to the research question mentioned at the beginning. In conclusion, this research can be a reference for, or become an assistant for developers who want to adopt web frameworks to enhance their project's quality.

REFERENCES

Electronic references

- Bright, P. 2012. Microsoft TypeScript: the JavaScript we need, or a solution looking for a problem? [accessed 15 March 2020]. Available at: <https://arstechnica.com/information-technology/2012/10/microsoft-typescript-the-javascript-we-need-or-a-solution-looking-for-a-problem/>
- Brown, K. 2018. JavaScript: How did it get so popular? [accessed 17 February 2020]. Available at: <https://news.codecademy.com/javascript-history-popularity/>
- Bukhari, 2011. What is Comparative Study. [accessed 4 April 2020]. Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1962328
- Copes, F. 2018. The Vue Handbook: a thorough introduction to Vue.js. [accessed 8 March 2020]. Available at: <https://www.freecodecamp.org/news/the-vue-handbook-a-thorough-introduction-to-vue-js-1e86835d8446/>
- Dentzel, Z. 2019. How the Internet Has Changed Everyday Life [accessed 13 July 2019]. Available at: <https://www.bbvaopenmind.com/en/articles/internet-changed-everyday-life/>
- Dudkin, I. 2016. HOW TO MAKE A WEBSITE WITH JAVASCRIPT, HTML AND CSS [accessed 10 May 2020]. Available at: <https://skywell.software/blog/how-to-make-a-website-with-javascript-html-and-css/>
- Fitzpatrick, T. 2017. A brief history of the internet. [accessed 13 July 2019]. Available at: <https://sciencenode.org/feature/a-brief-history-of-the-internet-.php>
- Front-end and back-end. Interaction in plain words. 2018. Livity [accessed 10 May 2020]. Available at: <https://lvivity.com/front-end-back-end-interaction>
- Gavigan, D. 2018. The History of Angular [accessed 15 September 2019]. Available at: <https://medium.com/the-startup-lab-blog/the-history-of-angular-3e36f7e828c7>
- Goel, A. 2020. 10 Best Web Development Frameworks [accessed 17 February 2020]. Available at: <https://hackr.io/blog/top-10-web-development-frameworks-in-2020>
- Gudelli, A. 2019. History of AngularJs [accessed 15 September 2019]. Available at: <https://www.angularjswiki.com/angular/history-of-angularjs/>
- Guo, D. 2019. The history and legacy of jQuery. [accessed 20 February 2020]. Available at: <https://blog.logrocket.com/the-history-and-legacy-of-jquery/>

- Hámori, F. 2018. The history of React.js on a Timeline [accessed 5 March 2020]. Available at: <https://blog.risingstack.com/the-history-of-react-js-on-a-timeline/>
- Hefetz, A. 2017. Bower is dead, long live npm. And Yarn. And webpack. [accessed 20 August 2019]. Available at <https://snyk.io/blog/bower-is-dead/>
- James, R. 2019. Best Web Development Framework Comparison [accessed 12 March 2020]. Available at: <https://www.pixelcrayons.com/blog/best-web-development-frameworks-comparison/>
- Library vs. Framework. 2018. ProgramCreek [accessed 11 May 2020]. Available at: <https://www.programcreek.com/2011/09/what-is-the-difference-between-a-java-library-and-a-framework/>
- Lindley, C. 2019. Front-end Developer Handbook. [accessed 28 December 2019]. Available at: <https://frontendmasters.com/books/front-end-handbook/2019/>
- Morgan, A. 2019. 6 Useful features in Angular CLI [accessed 5 March 2020]. Available at: <https://blog.logrocket.com/6-useful-features-in-angular-cli-eb502bd95874/>
- Morris, S. 2018. Tech 101: What is a JavaScript framework? Here's everything you need to know. [accessed 15 September 2019]. Available at: <https://skillcrush.com/2018/07/23/what-is-a-javascript-framework/>
- Morris, S. 2019. Tech 101: What is ReactJS? [accessed 4 April 2020]. Available at: <https://skillcrush.com/blog/what-is-react-js/>
- Osmani, A. 2012-2013. Developing Backbone.js Applications. [accessed 10 May 2020]. Available at: <https://addyosmani.com/backbone-fundamentals/>
- Pandit, N. 2020. What and Why React.js [accessed 4 March 2020]. Available at: <https://www.c-sharpcorner.com/article/what-and-why-reactjs/>
- Pluralsight, 2015. What's the Difference Between the Front-End and Back-End? [accessed 3 April 2020]. Available at: <https://www.pluralsight.com/blog/film-games/whats-difference-front-end-back-end>
- Rascia, T. 2019. Getting Started with Vue [accessed 8 March 2020]. Available at: <https://www.taniarascia.com/getting-started-with-vue/>
- Ryabtsev, A. 2019. Web frameworks: How to get started [accessed 3 July 2019]. Available at: <https://djangostars.com/blog/what-is-a-web-framework/>

Schwarz Müller, M. 2020. Angular vs React vs Vue [accessed 5 May 2020]. Available at: <https://academind.com/learn/angular/angular-vs-react-vs-vue-my-thoughts/>

Stack Overflow. 2018. Developer survey results [accessed 19 July 2019]. Available at: <https://insights.stackoverflow.com/survey/2018/#top>

Storimer, J. 2008. A Closer Look At the Blueprint CSS Framework [accessed 10 May 2020]. Available at: <https://code.tutsplus.com/tutorials/a-closer-look-at-the-blueprint-css-framework--net-2206>

Streefkerk, 2019. Qualitative vs. quantitative research [accessed 4 April 2020]. Available at: <https://www.scribbr.com/methodology/qualitative-quantitative-research/>

Sviatoslav, A. 2020. The Best JS Frameworks for Front End [accessed 15 February 2020]. Available at: <https://rubygarage.org/blog/best-javascript-frameworks-for-front-end>

TechMagic. 2018. React vs Angular vs Vue.js — What to choose in 2020? [accessed 13 March 2020] Available at: <https://medium.com/techmagic/reactjs-vs-angular5-vs-vue-js-what-to-choose-in-2018-b91e028fa91d>

W3school, 2020. Node.js Introduction [accessed 20 March 2020]. Available at: https://www.w3schools.com/nodejs/nodejs_intro.asp

W3schools. 2019. What is npm? [accessed 21 August 2019]. Available at: https://www.w3schools.com/whatis/whatis_npm.asp

Wales, M. 2014. 3 Web Dev Careers Decoded: Front-end vs Back-end vs Full Stack [accessed 5 April 2020]. Available at: <https://blog.udacity.com/2014/12/front-end-vs-back-end-vs-full-stack-web-developers.html>

Wanyoike, M. 2018. History of front-end frameworks. [accessed 20 August 2019]. Available at: <https://blog.logrocket.com/history-of-frontend-frameworks/>

Wozniewicz, B. 2019. The difference between a Framework and a Library [accessed 5 March 2020]. Available at: <https://www.freecodecamp.org/news/the-difference-between-a-framework-and-a-library-bd133054023f/>

APPENDICES

Web development frameworks

A small survey about your familiarity with web frameworks

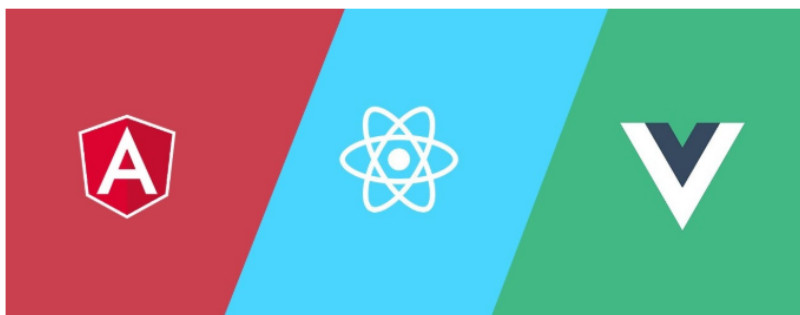
Are you a developer/trainee developer/IT student? *

- Yes
- No

Are you familiar with Angular/React/Vue web development frameworks? *

- Yes, I am familiar with them
- No, I have never heard of them
- I hear about it a few times but don't use them

Angular - React - Vue



⋮

If yes, which one of the above frameworks do you choose to implement an application or a website?

- Angular.js
- Vue.js
- React.js
- I would prefer other frameworks

Can you write a small comment on your feelings about each or only one of these frameworks?

Vấn bản câu trả lời ngắn

.....