

ELDER-CARE

A Progressive Web Application (PWA) to Assist the Elderly

Student Name: Craig Lawlor

Student Number: C00184465

Supervisor: Hisain Elshaafi

Submission Date: 9/11/2018

Research Manual

BSc (Hons) in Software Development



INSTITUTE *of*
TECHNOLOGY

CARLOW

Institiúid Teicneolaíochta Cheatharlach

Table of Contents

1. Abstract	3
2. Introduction	4
3. Background	5
4. Existing Applications	7
4.1 CaringBridge	7
4.2 CareZone	7
4.3 Carely	8
4.4 Summary	8
5. Progressive Web Applications	9
5.1 Why Progressive?	9
5.2 The Benefits	11
5.2.1 Responsive	11
5.2.2 Offline Availability	11
5.2.3 Low Data Usage	11
5.2.4 App-Like Experience	11
5.2.5 Installation Isn't Necessary	11
5.2.6 Secure	11
5.2.7 No Need to Update	12
5.2.8 No App Store	12
5.3 Pitfalls	12
5.3.1 Hardware	12
5.3.2 No App Store Presence	13
5.3.3 Not Fully Supported	13
7. Technologies	14
7.1 Languages	14
7.1.1 Html5	14
7.1.2 CSS3	14
7.1.3 JavaScript	15
7.1.4 JSON	15
7.1.5 SQL	16
7.2 Frameworks	16
7.2.1 Bootstrap	16

7.2.2 Angular	16
7.2.3 Ionic	17
7.3 Additional Technologies	17
7.3.1 Node.js	17
7.3.2 React	17
7.3.3 Knockout JS	17
7.3.4 GitHub	18
7.4 Development Environments/Code Editors	18
7.4.1 Visual Studio Code	18
7.4.2 Atom	19
7.4.3 WebStorm	19
7.5 Technologies Summary	20
8. References	21

1. Abstract

Digital technology can enhance our lives in many ways from keeping in contact to finding the information we want on demand. As technology advances it seems the elderly are being left behind. There are many contributing factors such as the fear of making mistakes or tough learning curves to modern applications. From the research carried out it is believed that the elderly, with the help of family and/or their care assistants, can too enhance their lives through the use of a web application.

The aim of this project is to help elderly people to stay connected and organised with the aid of a web application. Through the research it was found that a Progressive Web Application was an ideal way to deliver a smooth user experience across multiple devices. This will ensure that an elderly person who prefers to use a tablet or phone will have a similar experience to an elderly person who prefers to use a desktop or laptop. Multi device accessibility will enable more family members and/or care assistants to access and aid the elderly person with the application.

2. Introduction

This document will outline the research that was carried out for this project. The aim of this project is to provide a multi-platform application to aid elderly people and their carers.

At first the document will review applications already in existence to compare the features they provide. The document will then give a detailed overview of what a progressive web application is and why it would be the best option for this project.

The final part of this document will look at technologies to be considered for the development of the progressive web application.

3. Background

Through the research carried out for this project it was found that elderly people can gain from digital technology just as much as young people. It is true that technology usage amongst elderly people is consistently lower than that of younger people. It may be that younger people are more open to change and trying new things but there is no definite answer to this. One thing that is sure is that more and more elderly people are taking up smartphones and tablets. “According to Pew Research, almost 60% of American seniors over age 65 are now online, a number that is increasing rapidly each year.” (Breeding, 2016).

More and more elderly people are using the internet for things like keeping in contact with family and friends, playing games and keeping up to date with the news. Though there are elderly people who use desktops and laptops there is a significant rise in the number of elderly people using smartphones and tablets. The benefits to these devices speak for themselves with portability and internet access.

As people get older it is inevitable that things like memory and mobility loss will slow them down. The research for this project has shown that a Progressive Web Application can help elderly people keep up to speed in the digital age. The proposed application will contain features to help keep the elderly person independent by reducing the risk of missing appointments, events and medication. The application will also allow family members and care assistants to assist the elderly person by inputting information.

Some of the features of the application proposed application are:

- Medication reminder
- Scheduling appointments
- A help section with emergency service contact information and a facility to send a message to registered carers
- An activity section with exercises, memory games and assessments for early detection of diseases such as dementia
- A section for what's happening today such as appointments, birthdays and weather

There are multiple potential users for the proposed application. The aim is that the application can be used independently by a more capable elderly person and as their capabilities lessen the family members and care assistants would use the application

increasingly. It will also be possible for family members and care assistants to use the application exclusively in cases where the elderly person is unable to.

4. Existing Applications

During the research period of the project a number of existing applications were reviewed. The main focus of this research was the features the applications provided to the user, the user interface of the application and the platforms the application was available on. The sources used to find these applications were Google, the Google Play Store and Apple's App Store. Though many applications were researched only the most relevant are included in this document.

4.1 CaringBridge

This website enables the users faced with medical conditions and their families to communicate. The site allows the user to create their own personal site where they can input information on their condition to be shared with friends and family. Some of the features include the ability of family and friends to post messages to the person with the medical condition, personalisation options, a journal section, a gallery and personal fundraising. "Your personal CaringBridge website gives you one centralized, private place to share health updates and request help." (CaringBridge, 2018).

CaringBridge have an Android and an Iphone application as well as a web interface, though the Android application is not available in Ireland. The website interface is responsive and can be accessed on an android mobile device with a similar feel to the full desktop experience.

4.2 CareZone

This is a mobile application which enables the user to organise health information and health services. Once downloaded the application allows the user to scan their medication for detailed information which can be shared with their doctor. Some of the other features this application provides are medication reminders and a medication delivery facility.

CareZone have an Android and Iphone application, but again the application is not available in Ireland. With the use of screenshots from the application it was clear to see the user interface was clean and well laid out.

4.3 Carely

Carely is a mobile application which enables family and friends of an elderly person to plan visits and share information. The focus of this application is solely on the family members as users rather than the elderly person. Some of the features include planning visits (also to include a report on the visit), tracking the mood of a loved one, a calendar and a facility to send messages to other family members/carers.

Carely have an Android and Iphone application. The application is available on both platforms in Ireland, unlike the previous two applications.

4.4 Summary

A common problem found while researching was the lack of feature rich applications available in Ireland. Though there are applications available with appealing features they are mainly available in the USA. From the above applications Carely was the only one available in Ireland and while it has some nice features it focuses on the family rather than the patient. The aim of this project is to provide an application that both the family and elderly person can interact with.

5. Progressive Web Applications

5.1 Why Progressive?

An early decision made during the research phase of this project was to develop a progressive web application. There are many benefits to developing a progressive web application, which this document will expand upon, but first an overview on PWA's will be given.

A progressive web application is in some ways a bridge between standard web applications and native mobile phone applications. Firstly, they are responsive. So the PWA will be rendered appropriately for the window or screen size you are accessing from. PWA's use service workers to enable offline use so that even in areas of poor internet signal the application can still be used. Though many standard web applications developed nowadays are responsive to suit the large amount of varying screen sizes available today, they are not available offline. This is the first instance of that bridge between standard web and native applications mentioned earlier.

Progressive web applications offer an app-like experience to the user while running inside a browser. They initially ran on Google's Chrome browser but support has now expanded to many other browsers. As the PWA runs inside a browser they have a URL, but where they stand apart from standard web applications is that they are installable. This means that the user can choose to install the PWA and it can appear like a native application in the users app drawer on their mobile or on their PC.

A progressive web application can be easily updated. The developer can add new features or fix problems and upload the changed files to the server and if the application is implemented correctly these changes can be downloaded automatically. PWA's also include support for push notifications much like native mobile applications.

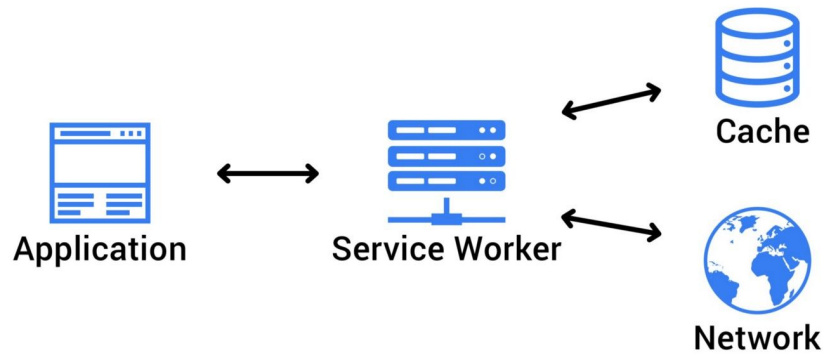


Fig.1: (Zlatkov, 2018)

As mentioned earlier, service workers enable progressive web applications to work offline. “A service worker is a script that your browser runs in the background, separate from a web page, opening the door to features that don’t need a web page or user interaction.” (Gaunt, 2018). This Javascript file sits between the network and the browser and can listen for events. As the service worker runs in the background it does not need the web application to be open. The service worker can respond to push messages sent from the server even when the application is not open.

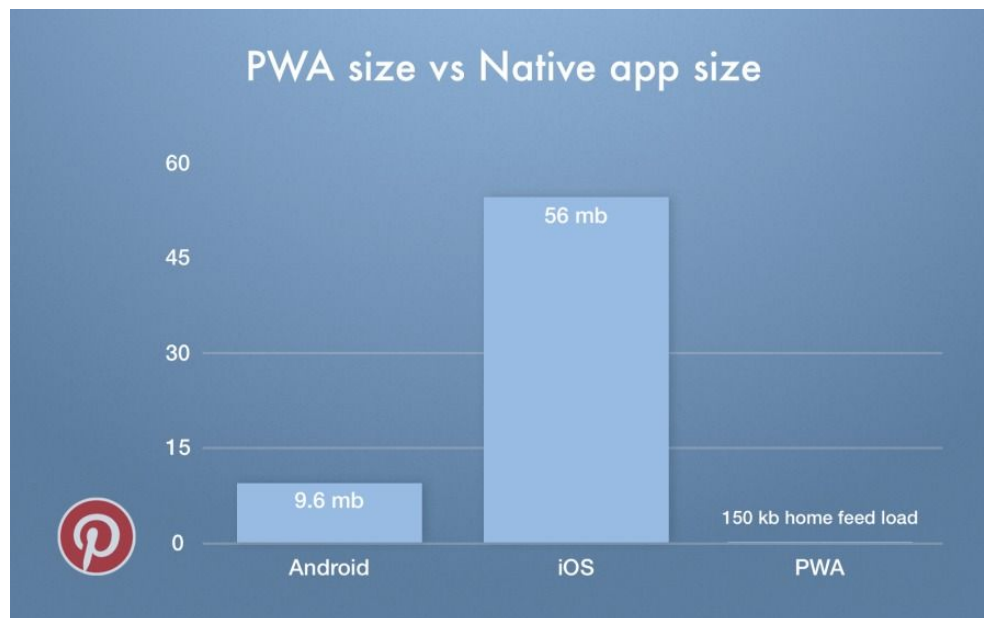


Fig.2: (Cleveroad, 2018)

Though progressive web applications can be installed they won't be found on an app store like Google's Play Store. Instead the PWA can be installed via a link or directly from the PWA's URL. As can be seen Fig.2 above, when compared with Android and iOS native applications Pinterest's PWA is only a fraction of the size.

5.2 The Benefits

Many of the benefits of progressive web applications stand out from the overview above but this section will explore them in more depth in the sub-headings below.

5.2.1 Responsive

Pages load to suit the size of screen that the PWA is being accessed from.

5.2.2 Offline Availability

With the use of service workers PWA's are usable offline. This means the user can use certain features of the application without an internet connection.

5.2.3 Low Data Usage

Firstly PWA's require much less data to install. After installation the use of service workers allow the PWA to keep data usage low when compared to native apps. This is especially advantageous in places with slower internet capabilities.

5.2.4 App-Like Experience

PWA's have the look and feel of native applications while still being accessible from the web browser. There is also the ability to install the PWA and add it to the home screen or app drawer of your mobile device. The PWA can also use push notifications just like a native app.

5.2.5 Installation Isn't Necessary

Though it is possible to install a PWA, all features can also be used via a web browser.

5.2.6 Secure

One of the requirements of progressive web apps are that they are served over HTTPS rather than HTTP.

5.2.7 No Need to Update

PWA's do not need to be updated as changes to files on the server made by the developer will automatically be downloaded.

5.2.8 No App Store

PWA's don't need to be published on the app stores. This benefits developers mostly as they don't have to develop the same applications for multiple devices. The same PWA will work across multiple devices.

	Native App	Responsive Website	Progressive Web App
Functions Offline	✓	✗	✓
Push Notifications	✓	✗	✓
Installable on home screen	✓	✗	✓
Full screen experience	✓	✗	✓
Indexable by search engines	✗	✓	✓
One place to enter content	✗	✓	✓
Works across all devices	✗	✓	✓
No download required	✗	✓	✓
Doesn't require updates	✗	✓	✓

Fig.3: (DeJong, 2018)

As can be seen from Fig.3 above, progressive web applications take many of the benefits from both web applications and native mobile applications.

5.3 Pitfalls

There are many benefits to progressive web applications but the fact that they are relatively new means there are areas in which they fall down.

5.3.1 Hardware

Though PWA's can use the device camera some of the latest hardware such as fingerprint scanners and facial recognition are not supported at present.

5.3.2 No App Store Presence

As mentioned above there are benefits to not needing to have the PWA on an app store, but the disadvantage to this is that a PWA does not get the exposure of being on an app store. For this reason it may be harder to gain popularity.

5.3.3 Not Fully Supported

As it is still early days in the life of progressive web applications, they are not fully supported by all browsers.

Over time some of these pitfalls will improve as progressive web applications evolve. The important thing to note is that they are still quite new in comparison with web technology and mobile applications.

7. Technologies

This section of the document will explore possible technologies and development environments to be used to build the progressive web application. The aim of this research is to find what is already being used to develop progressive web application and see which technologies would be best suited to the proposed application.

7.1 Languages

There are many languages which can be used to develop a progressive web application. This section of the document will give an overview of some of these languages.

7.1.1 Html5

The base of any website is HTML (Hypertext Markup Language). It is the markup language used to structure and present a web page. It will be the bones of any web page developed for the progressive web application in this project. HTML5 is the fifth version of HTML. “HTML5 is an effort is to bring order to web development chaos by organizing common practices, embracing implementations from various browsers.” (Techopedia, 2018). When used in isolation a HTML web page will be extremely basic and unattractive. It is with the use of CSS and Javascript that web pages become more appealing with style and functionality.

7.1.2 CSS3

CSS (Cascading Style Sheets) are used with HTML to make web pages more appealing visually. They describe how a HTML page should be displayed. “While HTML is used to structure a web document (defining things like headlines and paragraphs, and allowing you to embed images, video, and other media), CSS comes through and specifies your document’s style—page layouts, colors, and fonts are all determined with CSS.” (Morris, 2018). CSS3 is the third version of CSS and adds many new features such as flexboxes and grid layouts. Some of these features are needed to develop responsive websites. As a progressive web application must be responsive to appear well on multiple devices, these features will be essential.

7.1.3 JavaScript

Though it is possible to create a visually appealing website with only HTML and CSS, functionality would be very limited. This is where JavaScript comes in. JavaScript is a scripting language which enables interactive web pages. It can be thought of as a programming language which runs inside a web browser. Most desktop and mobile web browsers now support JavaScript. The majority of frameworks this document will look at later rely on JavaScript which will make it a fundamental part of the project.

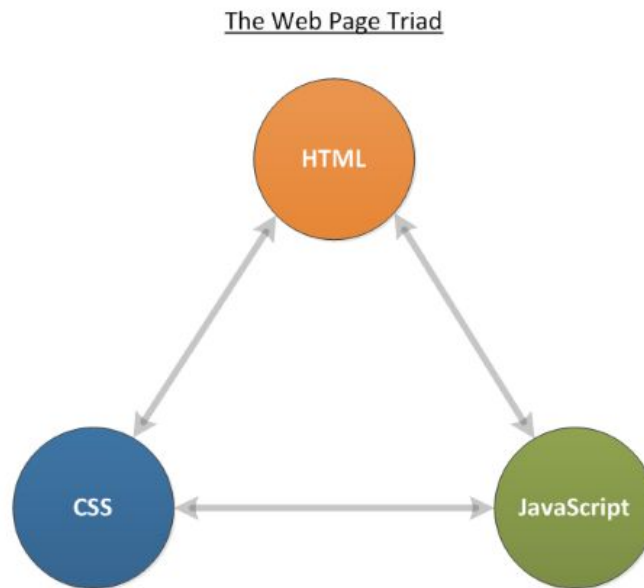


Fig.4: (Sridhar, 2017)

The combination of HTML, CSS and JavaScript is known as “The Web Page Triad” and forms the basis of the majority of web pages found on the internet today.

7.1.4 JSON

JSON (JavaScript Object Notation) is a way to store data in key-value pairs and is easier to read than XML. It is important for websites load data quickly without delaying loading times or rendering. JSON helps with this.

Progressive web applications must have a file known as the web app manifest. This file stores information about the app such as it's name, icons, start_url etc. The manifest file is a JSON file.

7.1.5 SQL

As there will be multiple users using the progressive web application certain information will have to be stored. Some of the information will be required for logging in. This information will have to be stored in a database. SQL will be used in order to communicate with the database.

SQL (Structured Query Language) is used for storing, changing and retrieving data in databases. “SQL commands allow you to create a host of components such as tables, schemas, stored procedures, indexes, domains, character sets, or even new databases altogether.” (BYTESCOUT, 2017).

7.1.6 noSQL

Another type of database which will be considered is noSQL. Unlike SQL databases noSQL databases are non relational which means the information is not stored in tables but rather in key value pair type structures similar to how JSON stores information. Examples of noSQL databases are MongoDB and Google’s Firebase.

7.2 Frameworks

To begin this section of the document it is important to explain what a framework is in terms of computer systems. A framework is a platform for developing applications which may include libraries to help the developer use existing functions rather than reinventing the wheel. Frameworks enable developers to get a project up and running quickly and provide structure.

7.2.1 Bootstrap

Bootstrap is a front-end framework which uses HTML, CSS and JavaScript. Bootstrap provides components that the developer can use to build a responsive web app. It is also possible for the developer to customise the components to suit their needs. Bootstrap also provide a number of templates which can make getting started with developing a web app a lot quicker.

To function, Bootstrap requires a JavaScript Library called jQuery. This library helps with cross browser compatibility.

7.2.2 Angular

Angular is a front-end web application framework based on JavaScript. It is a framework for building Javascript heavy single-page based web application. Single-page applications load all of a sites content within a single page, usually the index.html page. This means that once the page is initially loaded, clicking on a link will not reload the entire page. Angular allows the developer to add if statements, loops and data binding into the HTML template. Angular uses a language called Typescript which is based on Javascript.

7.2.3 Ionic

“Ionic is an open source, front-end SDK for developing Hybrid Mobile Applications using web technologies such as HTML, CSS and JavaScript. It provides mobile optimised web technology based components as well as native APIs using Cordova and Ionic Native.” (Mittal, 2017). Ionic also works well with Angular which could be instrumental when deciding on which technologies will be used for this project. It is also a framework which allows the developer to build mobile applications across multiple devices such as iOS and Android.

7.3 Additional Technologies

7.3.1 Node.js

“Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux.” (Tutorials Point, 2018). Node.js is a platform, it is not a language or framework. There are over 60000 modules currently available via the Node Package Manager (NPM).

7.3.2 React

React is a javascript library for building user interfaces. It's purpose is to be fast, scalable and simple. A major benefit in terms of this project is that React works well with other libraries and frameworks, such as Angular. Where some of the frameworks mentioned above will run through the stack, React is a front-end library. It will work well being served by any web server or backend.

Another benefit to using React is that from the JavaScript libraries out there it has a minimal learning curve. “If you know JavaScript then you can probably start writing React code in a single day.” (Suzdalnitski, 2018).

7.3.3 Knockout JS

“KnockoutJS is a library that connects parts of the UI to a data model using declarative bindings.” (Kirda, 2014). Unlike Angular, Knockout JS does not manage the whole application. This allows the developer to design the application structure themselves. In comparison with the binding syntax of Angular, Knockout is more involved requiring substantially more typing to achieve the same goal. Additionally bindings can only be applied once with Knockout.

7.3.4 GitHub

For any project it is important to adopt some form of version control. Version control allows developers to save their work and collaborate with their team. Managing source code is important and version control allows the developer to view previous versions of their code and even revert to a previous version if a problem arises with the current version. There are many different types of version control. The one of interest for this project is Git.

Git can be used locally on a system for version control but to store a git repository online a web-based host is required. GitHub as the name suggests is a hub or hosting service for git repositories. If your local system crashes it could mean you lose personal files as well as your code. For this reason it is important to use a web based host such as GitHub. For this project GitHub will be used for version control.

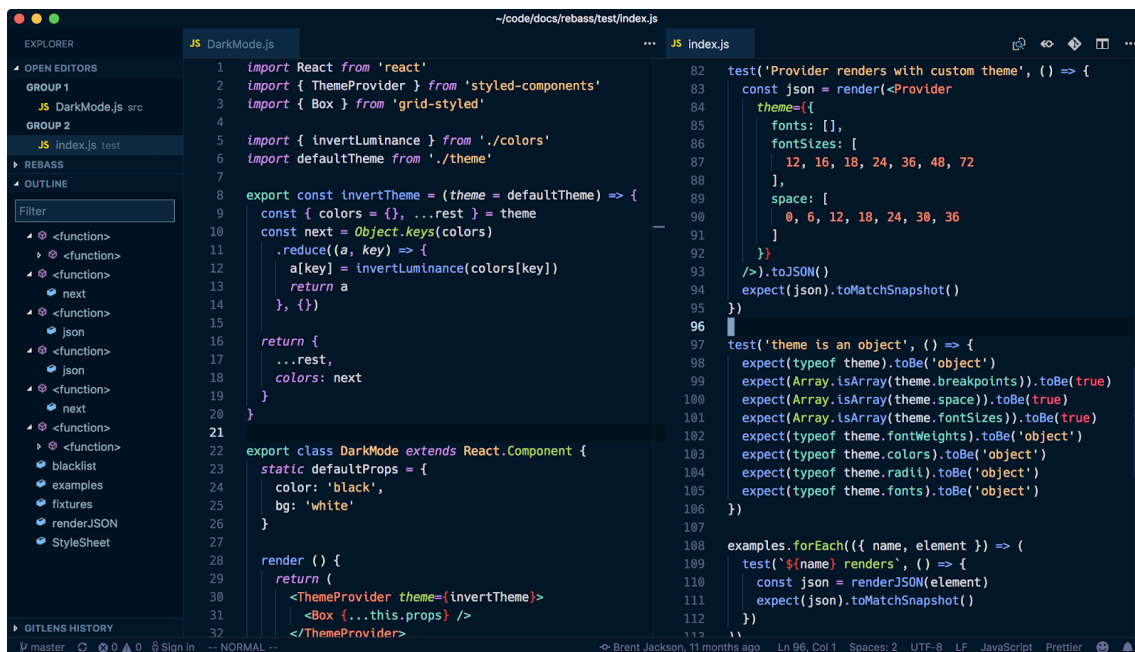
7.4 Development Environments/Code Editors

This section of the document will give an overview of a number of development environments and try to identify the best ones to use for this project. “A development environment is a collection of procedures and tools for developing, testing and debugging an application or program.” (Techopedia, 2018).

7.4.1 Visual Studio Code

“Visual Studio Code (VSCode) is a source code editor developed by Microsoft that can be run on Windows, macOS, and Linux.” (Le, 2018). It is customisable and provides themes which can be helpful to a developer for viewing syntax highlighted in different colours, especially when using multiple languages at the same time. The code editor is free to use and provides the ability to debug and use version control.

Visual Studio Code also has a large amount of extensions which can be downloaded such as linting tools, rich support for languages and many more. There is also a built in terminal. See Fig.5 below for an example of some JavaScript code in Visual Studio Code.



```

1 import React from 'react'
2 import { ThemeProvider } from 'styled-components'
3 import { Box } from 'grid-styled'
4
5 import { invertLuminance } from './colors'
6 import defaultTheme from './theme'
7
8 export const invertTheme = (theme = defaultTheme) => {
9   const { colors = {}, ...rest } = theme
10  const next = Object.keys(colors)
11    .reduce((a, key) => {
12      a[key] = invertLuminance(colors[key])
13      return a
14    }, {})
15
16  return {
17    ...rest,
18    colors: next
19  }
20
21 export class DarkMode extends React.Component {
22   static defaultProps = {
23     color: 'black',
24     bg: 'white'
25   }
26
27   render () {
28     return (
29       <ThemeProvider theme={invertTheme}>
30         <Box {...this.props} />
31       </ThemeProvider>
32     )
33   }
34 }
35
36 test('Provider renders with custom theme', () => {
37   const json = render(<Provider
38     theme={
39       theme:{
40         fonts: [],
41         fontSizes: [
42           12, 16, 18, 24, 36, 48, 72
43         ],
44         space: [
45           0, 6, 12, 18, 24, 30, 36
46         ]
47       }
48     />).toJSON()
49   expect(json).toMatchSnapshot()
50 })
51
52 test('theme is an object', () => {
53   expect(typeof theme).toBe('object')
54   expect(Array.isArray(theme.breakpoints)).toBe(true)
55   expect(Array.isArray(theme.space)).toBe(true)
56   expect(Array.isArray(theme.fontSizes)).toBe(true)
57   expect(typeof theme.fontWeights).toBe('object')
58   expect(typeof theme.colors).toBe('object')
59   expect(typeof theme.radii).toBe('object')
60   expect(typeof theme.fonts).toBe('object')
61 })
62
63 examples.forEach(({ name, element }) => {
64   test(`${name} renders`, () => {
65     const json = renderJSON(element)
66     expect(json).toMatchSnapshot()
67   })
68 })

```

Fig.5: (Uesugi, 2018)

7.4.2 Atom

Atom is a free and open-source code editor developed by GitHub. Similarly to Visual studio code Atom supports a large number of plugins and customisation options. With the use of plugins Atom can also be used as an integrated development environment (IDE).

7.4.3 WebStorm

“Webstorm is a light-weight and powerful javascript IDE developed by JetBrains. It can easily handle complex server and client side development with Node.js.” (Yadav, 2017). It is developed by JetBrains who claim it is the smartest editor for JavaScript. JetBrains can be used for both client-side and server-side development. The server-side development uses Node.js which this document explored earlier.

Unlike Atom and Visual Studio Code, WebStorm is not free which is not ideal for developers. Though it is possible to obtain a free student license. There are also a number of plugins available to install and debugging capabilities for both client and server side applications.

7.5 Technologies Summary

For this project it is intended that almost all of the technologies above will play a part in the final product, with the exception of the development environments. From the research conducted the code editor that will be used for the front end will be Visual Studio Code. The large number of plugins and syntax highlighting make it a very appealing option. For the more heavy lifting JavaScript elements of the project the WebStorm IDE will be used.

From the research carried out all of the languages above will be used for this project. The Web Page Triad mentioned above of HTML, CSS and JavaScript will form the basis of the progressive web application. Bootstrap will be used to aid in making the progressive web application responsive, as one of the most fundamental parts of being progressive is to be responsive.

From the research carried out on JavaScript frameworks and libraries it is intended to form another triangle of technologies. The combination of Ionic, Angular and React could form this triangle. Though many people see Angular and React as rival technologies in this area it is often forgot that Angular is a Model View Controller (MVC) framework and React is a library. From this viewpoint it is possible to use them in combination, with Ionic on top. A problem that may arise is with data binding. Angular uses a two-way binding system where React uses a one-way binding system. See Fig. 6 below for the differences between one and two way binding.

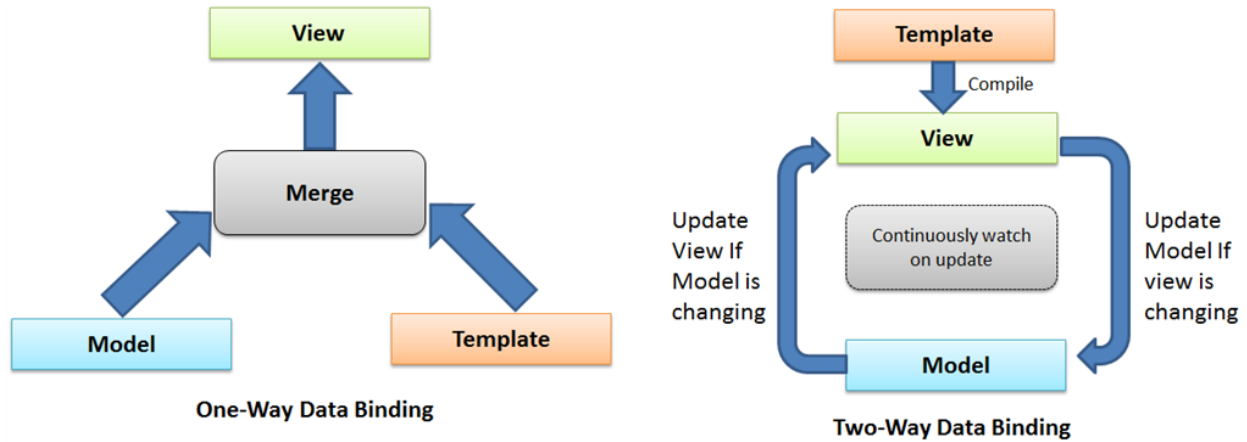


Fig.6 (Trivedi, 2015)

If problems do arise during the development phase of this project the aim would be to include Knockout JS to handle the bindings in combination with React.

8. References

Breeding, B (2016). Seniors and Technology- Embracing the Digital Age [ONLINE].

Available at:

<https://www.mylifesite.net/blog/post/seniors-and-technology-embracing-the-digital-age/>

[Accessed 15 November 2018]

BYTESCOUT (2017). WHAT IS SQL AND WHAT IS IT USED FOR [ONLINE]. Available

at: <https://bytescout.com/blog/what-is-sql-and-what-is-it-used-for.html> [Accessed 8

November 2018]

CaringBridge (2018). How CaringBridge Works [ONLINE]. Available at:

<https://www.caringbridge.org/how-it-works/> [Accessed 28 October 2018]

Cleveroad (2018). How to Develop PWA: All You Need To Know Before Creating a Progressive Web App [ONLINE]. Available at:

<https://www.cleveroad.com/blog/how-to-build-a-progressive-web-app-best-tools-and-examples> [Accessed 1 November 2018]

DeJong, J (2018). What's Next in Mobile: Progressive Web Apps [ONLINE]. Available at: <https://www.onenorth.com/blog/post/whats-next-in-mobile-progressive-web-apps> [Accessed 5 November 2018]

Gaunt, M (2018). Service Workers: an Introduction [ONLINE]. Available at: <https://developers.google.com/web/fundamentals/primers/service-workers/> [Accessed 4 November 2018]

Kirda, T (2014). Angular vs. Knockout: Similarities and Fundamental Differences [ONLINE]. Available at: <https://www.devbridge.com/articles/angular-vs-knockout-similarities-and-fundamental-differences/> [Accessed 11 November 2018]

Le, V (2018). An overview of Visual Studio Code for front-end developers [ONLINE]. Available at: <https://medium.freecodecamp.org/an-overview-of-visual-studio-code-for-front-end-developers-49a4aa0771fb> [Accessed 8 November 2018]

Mittal, A (2017). What is Ionic? [ONLINE]. Available at: <https://hackernoon.com/what-is-ionic-c1da6eab0d8a> [Accessed 9 November 2018]

Morris, S (2018). HTML & CSS, TECH 101: What Is CSS? [ONLINE]. Available at: <https://skillcrush.com/2012/04/03/css/> [Accessed 8 November 2018]

Sridhar, J (2017). What Is JavaScript and How Does It Work? [ONLINE]. Available at: <https://www.makeuseof.com/tag/what-is-javascript/> [Accessed 8 November 2018]

Suzdalnitski, I (2018). React.js: a better introduction to the most powerful UI library ever created. [ONLINE]. Available at: <https://hackernoon.com/react-js-a-better-introduction-to-the-most-powerful-ui-library-ever-created-eed96e8f4621> [Accessed 11 November 2018]

Techopedia (2018). Definition - What does Development Environment mean? [ONLINE]. Available at: <https://www.techopedia.com/definition/16376/development-environment> [Accessed 8 November 2018]

Techopedia (2018). HTML5: Techopedia explains HTML5 [ONLINE]. Available at: <https://www.techopedia.com/definition/1891/html5> [Accessed 4 November 2018]

Trivedi, J (2015). Data Binding in AngularJS [ONLINE]. Available at: <https://www.c-sharpcorner.com/UploadFile/ff2f08/data-binding-in-angularjs/> [Accessed 11 November 2018]

Tutorials Point (2018). What is Node.js? [ONLINE]. Available at: https://www.tutorialspoint.com/nodejs/nodejs_introduction.htm [Accessed 9 November 2018]

Uesugi, S (2018). The Best Visual Studio Code Dark and Light Themes [ONLINE] Available at: <https://medium.com/@chibicode/the-best-visual-studio-code-dark-and-light-themes-july-2018-edition-a9c2cc9548da> [Accessed 8 November 2018]

Yadav, H (2017). Quora - What is WebStorm IDE? [ONLINE]. Available at: <https://www.quora.com/What-is-WebStorm-IDE> [Accessed 9 November 2018]

Zlatkov, A (2018). How JavaScript works: Service Workers, their lifecycle and use cases [ONLINE]. Available at: <https://blog.sessionstack.com/how-javascript-works-service-workers-their-life-cycle-and-use-cases-52b19ad98b58> [Accessed 8 November 2018]