ELDER-CARE

A Progressive Web Application (PWA) to Assist the Elderly

Student Name: Craig Lawlor

Student Number: C00184465

Supervisor: Hisain Elshaafi

Submission Date: 23/11/2018

Functional Specification

BSc (Hons) in Software Development

INSTITUTE of TECHNOLOGY CARLOW

Institiúid Teicneolaíochta Cheatharlach

Table of Contents

1. Introduction

The purpose of this document is to outline the functionality and requirements for the project. The project aims to provide a Progressive Web Application that will help an elderly person stay organised and live independently for as long as possible.
As there are varying levels of technological ability amongst elderly people, care assistants will also be able to use the application to aid the elderly person. Both family members and care assistants will have identical permissions with the application so for the purposes of this document they will both be referred to as care assistants.

The first section is the Project Concept and will expand upon the inspiration behind the project, how elderly people use technology, the prefered platform, how care assistants can aid the elderly with technology and how application may be used to enrich the lives of the elderly.

The System Architecture section will display the proposed architecture for the project. The Use Case Diagram will illustrate how the user will use the system. It will also identify the Use Cases for the following section. Once identified the Use Cases will be expanded upon.

The Metrics sections will aim to measure the level of success the project will gain by outlining the project goals. The Supplementary Specification section will outline the requirements which are not typically covered by the use cases. The Project Plan section will break the entire project down into three iterations and then expand on what will take place during each iteration of the project.

2. Project Concept

Digital technology is all around us today, from smartphones to home heating systems. It is often thought that technology is for younger people. The fact that technology is so present in everyday life now makes it a lot more accessible and also hard to avoid. Many elderly people fear technology due to the learning complexities and possibility of making mistakes. The use of smartphones and tablets make technology much more accessible and research has seen a large increase in the amount of elderly people now using technology and the internet. There are many benefits to having access and being able to use the internet such as keeping in contact with family and gaining access to information.

As people get older it is inevitable that things like memory and mobility will decrease over time. When these things start to deteriorate it is harder to stay organised and live independently. The vision of this project is to enable elderly people to keep their independence by staying organised through the use of technology. As there are different levels of capability amongst the elderly when it comes to technology it would be beneficial to enable family members and care assistants to aid the elderly person with this technology.

To expand on this idea a suitable platform had to be decided upon. As the device preference between elderly people may differ substantially from one user wanting to use a tablet to another wanting to use a laptop it is important that the platform is similar across multiple devices. It is also possible that the elderly person may wish to, knowingly or otherwise, use the application while they are not connected to the internet. This may be to log information or check an appointment.

For the above reasons it was decided to develop a Progressive Web Application. “A progressive web application takes advantage of the latest technologies to combine the best of web and mobile apps. Think of it as a website built using web technologies but that acts and feels like an app.” (Farrugia, 2016). Additional benefits to developing a Progressive Web Application is that they can be installed, receive push notifications and be used offline. The use of service workers enable offline use by caching information until an internet connection is reestablished.

At this stage the aim of the project is to develop a Progressive Web Application that an elderly person can register on and interact with. The proposed interactions are for the elderly person to be able to log diary information, to record appointments, to set

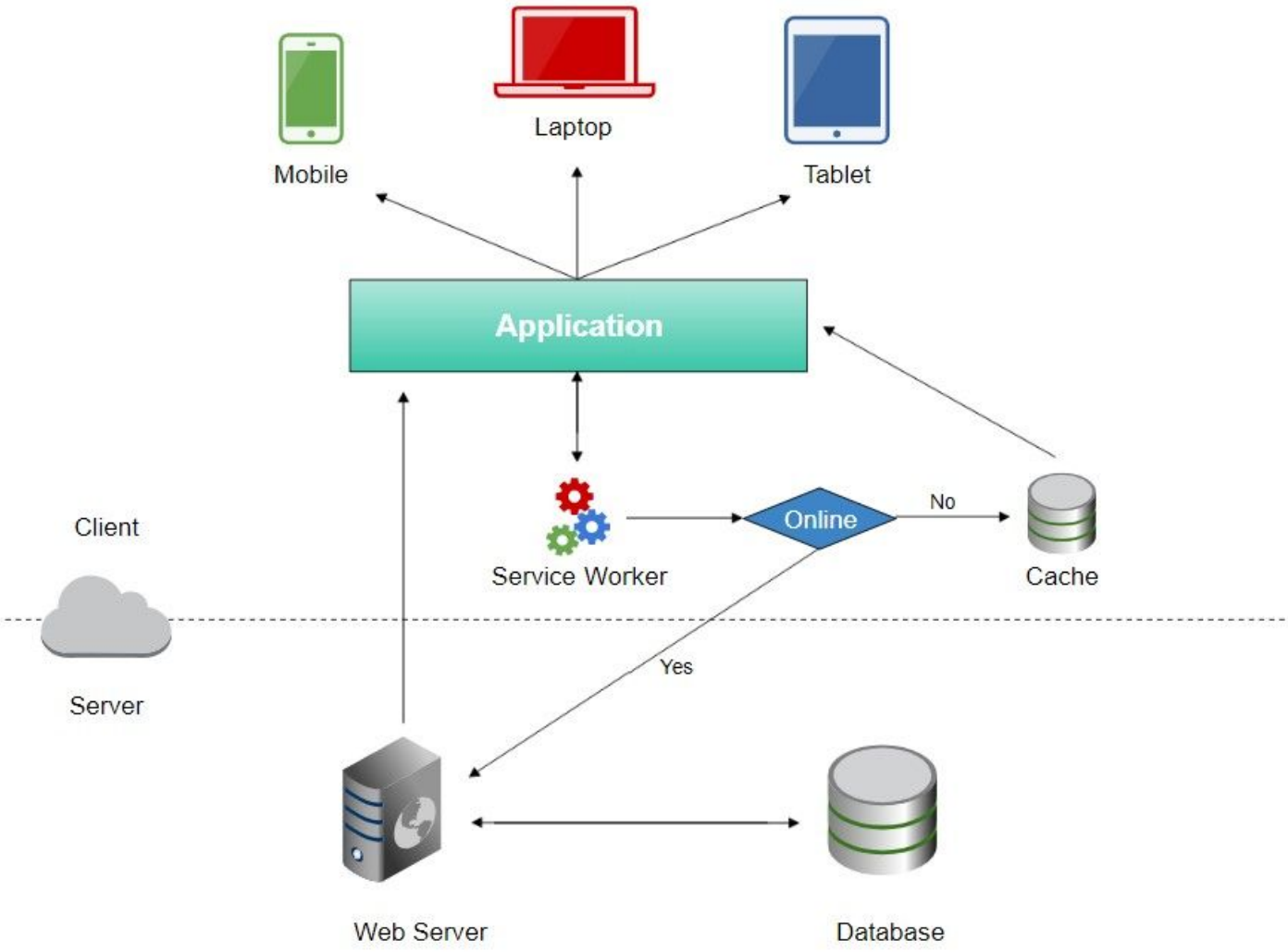
reminders for medication and other events, to play activity games, to take medical assessments, to view emergency contact information and possibly call for aid via the application. To help the elderly person further family members and care assistants would be able to register an account associated with the elderly person via a family code. This would enable them to help the elderly person with some of the interactions above. There would also be a section where caregivers (family members and care assistants) can log information and communicate with each other.

A more capable elderly person would need less help from caregivers with the Progressive Web Application and could even use it exclusively. On the other hand the Progressive Web Application could be used exclusively by caregivers as a collaboration tool when caring for a loved one. The latter scenario would apply if the elderly person was incapable of using the application.

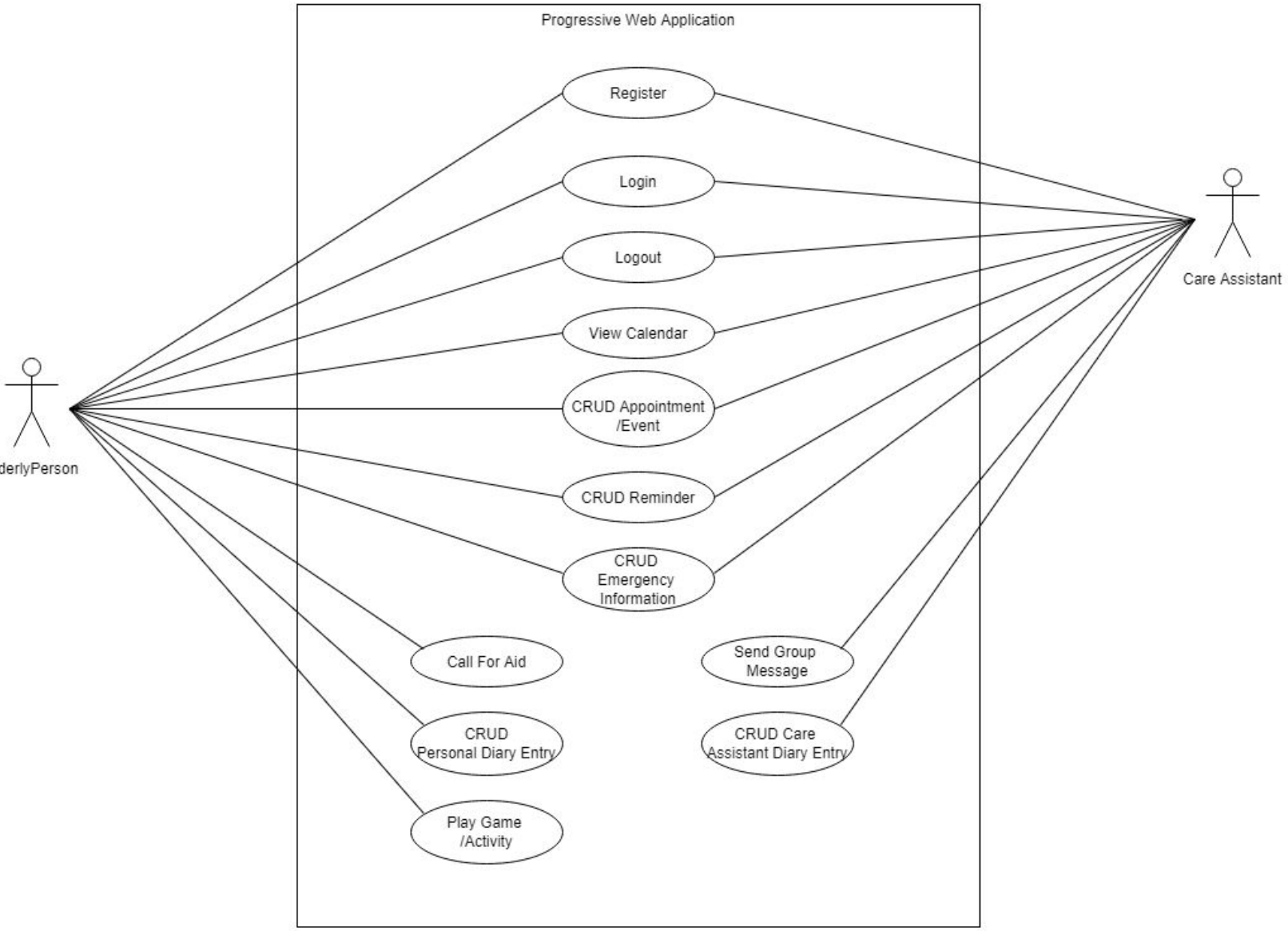
The original inspiration came when searching for a similar application to help a loved one. The research conducted failed to find a feature rich multi-platform application available in Ireland. With the majority of feature rich applications being exclusively available in the USA.

3. System Architecture

The initial concept of the system architecture.



4. Use Case Diagram



5. Brief Use Cases

For the following use cases the actor ‘Care Assistant’ will refer to both family members and care assistants as they both have the same role and permissions for this project. CRUD stands for create, read, update and delete.

5.1 Register Elderly Person

Actor: Elderly person
Description: This use case begins when the actor wishes to register for an account. The actor fills in the required fields and submits the information. The use case ends when the actor is successfully registered with the system.

5.2 Register Care Assistant

Actor: Care Assistant
Description: This use case begins when the actor wishes to register for an account. The actor fills in the “Family Username” assigned to the elderly person they are associated with. The actor is then redirected to the registration form if the “Family Username” exists. The actor then fills in the registration form fields and submits the form. The use case ends when the actor is successfully registered with the system.

5.3 Login

Actors: Elderly person, Care Assistant
Description: This use case begins when the actor wishes to log into the system. The actor enters their username and password into the fields displayed on the login screen. The use case ends when the actor is successfully logged in.

5.4 Logout

Actors: Elderly person, Care Assistant
Description: This use case begins when the actor wishes to logout of the system. The actor selects logout from the menu. The system prompts the actor with a message asking if they are sure they want to log out. The actor selects yes. The use case ends when the actor is successfully logged out.

5.5 View Calendar

Actors: Elderly person, Care Assistant
Description: This use case begins when the actor wishes to view the calendar of the elderly person. The actor selects the calendar from the application dashboard and the system displays the calendar. This use case ends when the actor has viewed the calendar.

5.6 CRUD Appointment/Event

Actors: Elderly person, Care Assistant
Description: This use case begins when the actor wishes to CRUD an appointment or event. When the actor has selected “Appointments and Events” from the application dashboard they will be presented with a number of options. They can create a new appointment or event, view existing appointments and events, amend an existing appointment or event and delete an existing appointment or event. Alternatively the actor can create an appointment or event from the calendar. The actor can also select and RUD existing appointments or events from the calendar view. This use case will end when the actor has CRUD an appointment or event.

5.7 CRUD Reminder

Actors: Elderly person, Care Assistant
Description: This use case begins when the actor wishes to CRUD a reminder. When the actor has selected “Reminders” from the application dashboard they will be presented with a number of options. They can create a new reminder, view existings reminders, amend existing reminders and delete existing reminders. Alternatively the actor can create a reminder from the calendar. The actor can also RUD a reminder from the calendar. This use case ends when the the actor has CRUD a reminder.

5.8 CRUD Emergency Information

Actors: Elderly person, Care Assistant
Description: This use case begins when the actor wishes to CRUD emergency or contact information. When the actor has selected “Emergency Information” from the application dashboard they will be presented with a number of options. They can create

a new emergency contact, view an existing emergency contact, amend an existing emergency contact and delete an existing contact. This use case ends when the actor has CRUD an emergency contact.

5.9 Call For Aid

Actor: Elderly person
Description: This use case begins when the actor wishes to call either a contact or emergency services for aid. When the actor is on the “Emergency Contacts” page they can call a listed contact if they are using a mobile device. Alternatively if they are using a computer or tablet they can send an alert that will be seen by all care assistants. This use case ends when the actor has either called for aid or sent an alert for aid.

5.10 CRUD Personal Diary Entry

Actor: Elderly person
Description: This use case begins when the actor wishes to CRUD a personal diary entry. A diary entry made by the elderly person cannot be seen by care assistants. When the actor has selected “Diary” from the application dashboard they will be presented with a number of options. They can create a new diary entry, view past diary entries, amend past diary entries and delete past diary entries. This use case ends when the actor has CRUD a diary entry.

5.11 Play Game/Activity

Actor: Elderly person
Description: This use case begins when the actor wishes to play a game or activity. When the actor has selected “Games and Activities” from the application dashboard they are presented with options to play memory based games and activities. This used case ends when the actor is playing a game or activity.

5.12 Send Group Message

Actor: Care Assistant
Description: This use case begins when the actor wishes to send a group message. When the actor selects “Group Messages” from the application dashboard they will be presented with a number of options. They can create a new message that will be sent to

all care assistants registered to the elderly person. They can also view previously sent and received messages. The elderly person will not see these messages on their account. This use case ends when the actor has successfully sent a group message.

5.13 CRUD Care Assistant Diary Entry

Actor: Care Assistant
Description: This use case begins when the actor wishes to CRUD a diary entry. A diary entry made by a care assistant can be seen by all care assistants but not the elderly person. When the actor has selected “Diary” from the application dashboard they are presented with a number of options. They can create a new diary entry, view past diary entries, amend past diary entries and delete past diary entries. This use case ends when the actor has CRUD a diary entry.

6. Metrics

This project will be measured against a number of goals or achievables to determine whether or not it is a success. There will be certain goals that must be reached as a minimum with additional goals which will be deemed more ambitious. This project must at a minimum provide a Progressive Web Application which can be used by both elderly people and their care assistants.

The users of the application should be able to:

- CRUD appointments & events
- CRUD reminders
- CRUD diary entries
- Play Games & activities
- CRUD emergency information
- Call for aid
- Send messages to other care assistants

The aim at this stage is to create a fully working signup and login system which will enable the two types of users (elderly person and care assistant) to have different privileges based on the account type they sign up for. This could be challenging as it is important that care assistants can also access and modify certain elements on the elderly persons behalf. So the care assistant must be able to carry out certain actions for themselves and other actions on behalf of the elderly person. The back end of the Progressive Web Application will have to be carefully designed to achieve this goal.

An ambitious goal for this project is to implement a games/assessments section where the elderly user can play memory based games and take early symptom detection assessments. Scores would be stored in the back end and could be reviewed over certain time periods to identify degradation over time via charts.

7. Supplementary Specification

To define the supplementary specifications for this project the FURPS+ model will be used. The FURPS+ acronym stands for Functionality, Usability, Reliability, Performance, Supportability and the plus is used to specify constraints around the design, interface and implementation among other things. This section of the document will first give a brief overview of each part of the FURPS+ model followed by the specifics for this project.

7.1 Functionality

“What the customer wants! Note that this includes security-related needs.” (Ottinger & Langr, 2009).

For the purposes of this project the following functionality will be required:

- ➢ The system will allow the user to CRUD appointments and events, reminders, emergency information and diary entries.
- ➢ The system will cache the users session when there is no internet connection.
- ➢ An internet connection will be required to send group messages.

7.2 Usability

“How effective is the product from the standpoint of the person who must use it? Is it aesthetically acceptable? Is the documentation accurate and complete?” (Ottinger & Langr, 2009).

This application will be primarily be used by elderly people and therefore must take their needs into consideration. The information must be displayed clearly and navigation must be as simple as possible. With this in mind the usability of this project will be defined as follows:

- ➢ The main functionality of the application should be clearly visible to the user.
- ➢ There should be a link to the home page on every screen.
- ➢ The buttons should be large enough so that a user has less chance of hitting the wrong one.

- ➢ The application should give a mobile user a similar experience to a desktop user by being fully responsive.

7.3 Reliability

“What is the maximum acceptable system downtime? Are failures predictable? Can we demonstrate the accuracy of results? How is the system recovered?” (Ottinger & Langr, 2009).

For the purposes of this project the following reliability will be required:

- ➢ The system must cache the user’s session when there is no internet connection as it is a progressive web application.
- ➢ The system must be available almost all of the time.
- ➢ The user should be able to access stored information such as appointments and emergency contact information without an internet connection.

7.4 Performance

“How fast must it be? What's the maximum response time? What's the throughput? What's the memory consumption?” (Ottinger & Langr, 2009).

For the purposes of this project the following performance will be required:

- ➢ The system should be able to handle a large number of active users at one time.
- ➢ The system screens should load relatively quickly so that users are not waiting too long.

7.5 Supportability

“Is it testable, extensible, serviceable, installable, and configurable? Can it be monitored?” (Ottinger & Langr, 2009).

For the purposes of this project the following supportability will be required:

- ➢ The application should be available for almost any platform with the responsive design.
- ➢ The system should be easily expanded upon in the future to add more features.

7.6 +

The plus covers the additional needs not met by the previous headings.

7.6.1 Security

- ➢ The progressive web application must be delivered over HTTPS.
- ➢ The system must use encryption to protect the users data.

8. Project Plan

This section of the document will outline the plan for the different stages of the project which will be broken down into three iterations. Each iteration will contain tasks to be carried out.

8.1 First Iteration

8.1.1 Research

The first iteration began by researching existing application which were similar to the proposed project idea. Further research was carried out on how elderly people use technology and how it could be made easier on them. It was decided at this point that a Progressive web Application would be the best option to suit the varying needs of the elderly users and their care assistants.

8.1.2 Platform

Having decided that progressive was the way to go, it was then time to decide on the platform. The initial plan is to develop the Progressive Web Application using React, Angular and Ionic together on top of HTML and CSS aided by Bootstrap. From the research carried out it was decided that the back end should be designed with SQL rather than a noSQL database.

8.1.3 Functionality

To follow on from the research phase of the project, the functionality of the Progressive Web Application had to be decided. At this point while writing the functional specification the core functionalities of the project were identified.

8.1.4 Design

After the functionalities of the project are decided on the project must be designed. Before any coding can begin the screens and overall navigation must be designed. Though any user interfaces will not be set in stone at this point it is important that all functionality is visible and full application navigation is designed. The database tables will also be designed at this stage.

8.2 Second Iteration

The second iteration is where the majority of development will take place. It is intended that the existing documents will be amended as needed will the coding takes place.

8.2.1 Software

At the beginning of the second iteration the coding will start with an aim of providing a rough prototype by the end. The IDE's and frameworks have to be set up to begin learning the selected languages and to figure out quickly what will work and what won't. This stage will involve some trial and error to identify the technologies best suited for the Progressive Web Application.

8.3 Third Iteration

The final iteration will begin with the preparation of the project presentation. It will also involve writing the remaining two documents as well as amending existing documents for the project.

8.3.1 PWA Implementation

It is also intended that at this stage of the project the Progressive Web Application will be coded into a fully functioning application. Though the number of features/functionality that will be completed at this stage is hard to estimate, the minimum expected outcome is that the application is truly progressive and installable. It is also expected that a fully working signup and login system is in place for the different types of users. Building upon this any features/functionality must work correctly based on this multiple user account system. The privileges the user will have will be dependant on the type of account the user signs up for.

9. References

Farrugia, K (2016). A Beginner's Guide To Progressive Web Apps [ONLINE]. Available at: https://www.smashingmagazine.com/2016/08/a-beginners-guide-to-progressive-web-apps/ [Accessed 20 November 2018]

Ottinger, T., Langr J (2009). Agile in a Flash [ONLINE]. Available at: http://agileinaflash.blogspot.com/2009/04/furps.html [Accessed 20 November 2018]