Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

Cieran Hughes

School of
Applied Computing

Aim and Objectives

Hypothesis

To research the concept of progressive web applications and see whether a progressive web app can be used alongside or instead of a native application.

Aim

- To research and evaluate the functionality, feature usability and requirement specification criteria for a Progressive Web App (PWA).

Objectives

- To research the PWA technology and specification criteria that a web application needs to meet to qualify as an official PWA.
- To design, develop and implement a PWA using all the requirements researched. This will allow all the functionality that a PWA requires to be included. It will be achieved by developing the PWA using web technology languages and a JavaScript library.
- To evaluate the functionality usability of a PWA from the user point of view. Looking at the ease of installation and usability comparing it to the other applications offered.

Research

PWA

What is a PWA?

- An application that can be run in any web browser.
- It uses web technologies that are used on many websites such as HTML, CSS and JS Functionally.
- Takes advantage of APIs to provide dynamic content for the user.

Features

- Performance
- Push Notification
- Accessible and Usability
- On-Board Device Hardware
- Installable
- App Maintenance
- Offline Content

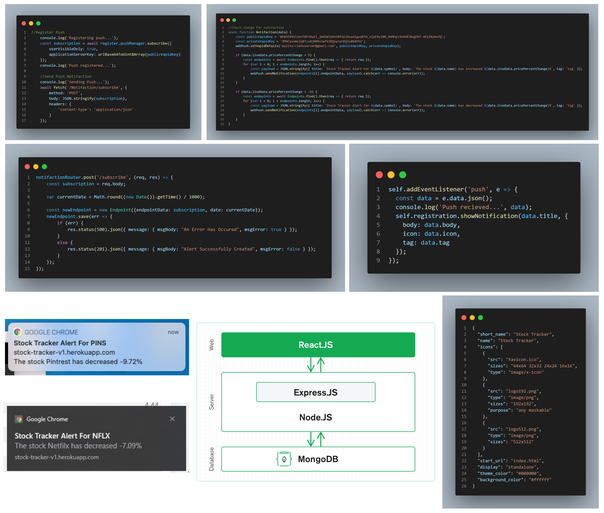
Technical Features

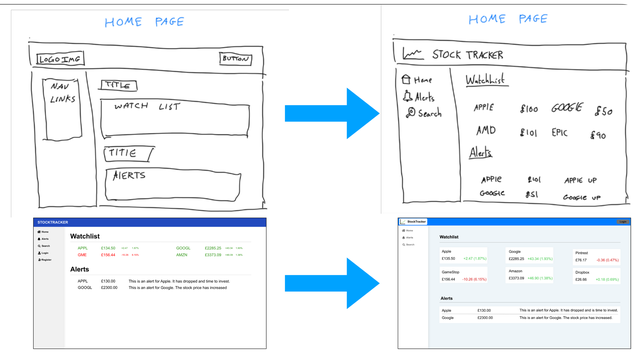
- Service Workers
- Web Manifest

Design & Implementation

Design Techniques

- WireFrame Digram
- Software Prototype 1
- Paper Prototype
- Software Prototype 2





Implementation

Successes

The main successes of the development were implementing push notification to the users when a stock increases or decreases by five percent, as seen in the screenshots.

Another success of the development was the installability. The users are able to install the application as they would a native application for the device, showing in the home screen or dock.

Techniques

For the development of this project the MERN stack was chosen. It consists of MongoDB as the database server, ExpressJS as the web framework, ReactJS as the client-side framework and NodeJS as the web server

Results

Lighthouse

'Lighthouse PWA Analysis Tool' is used to audit a PWAs features, is open source and developed by Google. When it is run on a PWA, it provides the tester with a set of metrics on how the PWA is able to give a full application experience. As seen the PWA section of the analysis tool is all successfully met, this is the most important metric that the tool gives.

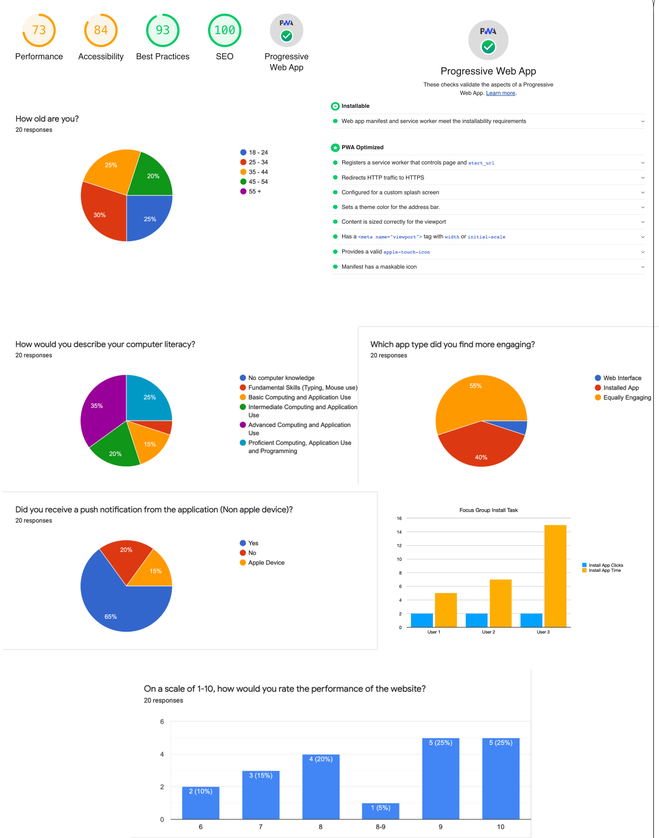
Questionnaire

All the users taking part in the questionnaire had a varying of age, computer knowledge and device used to ensure a fair test. 40% of the user found the installed application most engaging and these users were mainly the laptop device users. The performance score was very high from all the users and every smartphone user gave the score of an eight or over.

Looking at the usability stand point all the users that gave an opinion on the design and colour schemed were all positive. 100% of the user through the application rendered well. Finally only one user thought they had to click too much to navigate to the pages where as the other 95% did not.

Focus Group

Looking at the focus group graph. All the users that were asked to install the application from the browser could and did it in two clicks. The least computer literate user took fifteen seconds to get it done, but did so with no extra input.



Analysis

From the data that was collected on the usability aspects of the application developed, a positive conclusion can be drawn. When looking at the results gathered about the usability standpoint of the PWA, the conclusion can be positive as all the answers on the design and colour scheme were positive. Also, the focus group showed that the usability design was at a high quality. Therefore, the results show that the usability side of the PWA can be used as an application alongside or over a native application for the OS.

The PWA functionality covered the majority of things that a native application can also do. Firstly, the application can be installed with ease over a native phone application. This is backed up by the focus group data collected on the time taken and amount of taps it takes the users to install the PWA. Also, the questionnaire shows that all the users found the install process to be seamless and found the installed PWA more engaging over the web based version.

The second aspect of the PWA is the push notification, this worked as a native application would have with two downfalls. The first downfall being that the notifications are only sent if the PWA is active when installed or if not installed then the browser used to install it is open. The second minor downfall of the push notifications is the compatibility of OS and devices. The notifications will not work if the application is being used on an iPhone, iPad or used in Safari.

Overall, when looking at the results from the data gathered all the users except from four that were expected to receive a notification did. Therefore, when using PWA push notifications they can hold up well against native applications.

Conclusions

The aim for this project was to research and evaluate the functionality, feature usability and requirement specification criteria for a progressive web application. This allowed for the gathering of the specification criteria of what a PWA is defined as. Using this the author was able to implement a successful PWA to be used for data collection.

The objectives gave structure to the study as the second chapter was able to meet the first object by researching the main criteria that was needed. Following this the third chapter was the data collection methods, which ensures an accurate evaluation. The design and implementation section allowed for planning of the main development of the PWA and how it would work.

The key result from this project was the development piece, the developed PWA was a considered a success that allowed the testing on.

Overall, the study was a success as the aim and all the objectives were met with the main conclusion being drawn that a PWA can rival a native phone aplcaiotn as the feature all so, even though they are in their infancy.

Further Work

If this study was repeated, the first change would be focusing more on the process behind the PWA and how they work over the frontend. This change would make the study more in depth to the technologies behind a PWA and therefore may give different results. The problem with this change would be there would be a lesser understanding on the whole process of the PWA and more on the backend and how they work rather than the implementation.