Building Progressive Web Apps Using Polymer Library

Harry Alvin Waidan Kefas

Informatics / Computer Science
School of Electrical Engineering and Computer Science, Bandung Institute of Technology
Bandung, Indonesia
harryalvinwk@gmail.com

Abstract—**The problem in the web applications nowadays is the user experience when the connection is unreliable. This problem is tried to be solved by Progressive Web Apps. With this progressive web apps, the users of the web application can be more engaged and have a new experience to web applications. In this paper, we will see how does Progressive Web Apps create a new user experience to the web application users.**

Keywords—user experience; web application; service worker; web component; native app experience, progressive web apps, web library

I. INTRODUCTION

These days, the web development is having more version of concepts. But, the time spent in native mobile applications is dominating over the time spent in websites. According to the survey by the comScore, in July 2016, the usage of smartphone/native applications (50%) is bigger than desktop web application or websites (32%).
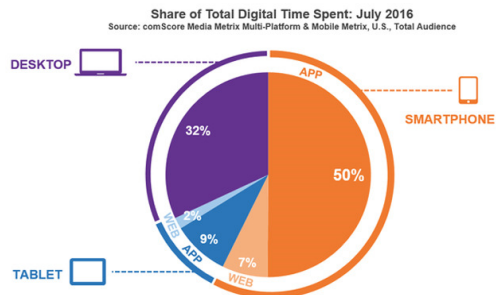

Figure 1. Survey of Total Digital Time Spent on Electronic Devices[2]

Because of that, the websites need to be more variative. One of the new version of technology on websites nowadays is the Progressive Web Apps. Lynch (2016) in reporting a study conducted by Progressive Web Apps highlights that "A Progressive Web App uses modern web capabilities to deliver an app-like user experience". One of some ways to create this Progressive Web Apps is by using a web library. Within this paper, will be explained the usage of Polymer, a web library that is used for building a Progressive Web Apps.

This paper is divided into four sections. The first section is introduction. The second section will explain more about the concepts of progressive web apps. The third section will explain about the implementation of progressive web apps using Polymer Library. Finally, in the fourth section we will conclude the implementation.

II. PROGRESSIVE WEB APPS

A. Service Worker

Unlike a native websites, the progressive web app have something to serve the web so it can be accessed when the client is offline. This concepts is realized by Service Worker. The service worker is an additional layer that bridge the web server and the client. This service worker can be like a proxy when the client request to a web server. Like proxy in general, service worker is also used as a cache for the content that were provided in the web server.

B. The App Shell

The interesting user experience in a progressive web app is the App Shell. By enabling this App Shell to the progressive web apps, the users can feel the experience of a website like a native application. With this app shell, we can optimize a website so it will only request for content only while in the website because the wrapper (navbars, drawer, header-footer) only loaded once.

C. Components

The contents of the progressive web apps is moduled by a concept called Component. With this Component concept, the progressive web apps can be more easy to be manipulated and placed in the page.

D. The JSON Manifest

Like a native application, the progressive web apps need a manifest for storing the datasets for the application. This manifest used so the progressive web app's shortcut can be like the native application's launcher button.

III. IMPLEMENTATION USING POLYMER LIBRARY

By using the Polymer 1.9 Library, we can arrange the contents of the website and for each content, we can make the content's own container.

In order to use the libraries provided in the Polymer, we can use Node.js Package Manager (npm) and the Bower Package Manager. After we install the two package manager we mentioned earlier, we can start downloading the components we needed in order to give our progressive web apps a content.

After we download the components, we can use the components we downloaded to the HTML file containing the content we wanted to make. In order to define a content using components we downloaded earlier, we need to import the component by using :

```
<link rel="import"
href="path/to/component/module/directory">
```

After we import the components to the content, we can then define the style and the structure of the content using HTML and CSS called template. There is a convention to define the template, that is :

```
<dom-module id="content-name">
    <template>
        <style>
        </style>
        <!-- DOM STRUCTURE -->
    </template>
    <script>
    </script>
</dom-module>
```

After we finished the template, at the *script* tag that is in the *dom-module* tag, we need to define the basic Polymer constructor to activate the components we imported, that will be looked like the text below :

```
Polymer({
    is: content-name;
});
```

We can see that the value of "is" in the Polymer constructor must be the same as the value of id attribute in the *dom-module* tag. After we complete the content's construction, we can either use it to another content or use it in the main HTML file of the website. In order to do that, we just need to make a tag that named by the value of id attribute in the *dom-module* tag before.

If we using the previous example, it means to use the *content-name* content, we just need to make a tag like below :

```
<content-name></content-name>
```

In order to make this chapter more understandable, we can make a project that creates a progressive web app like the figures below.
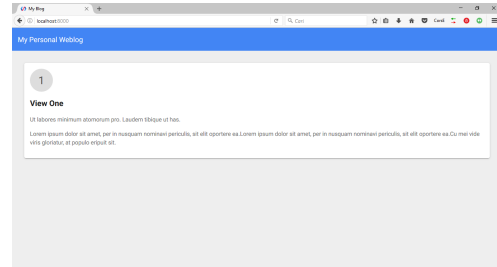

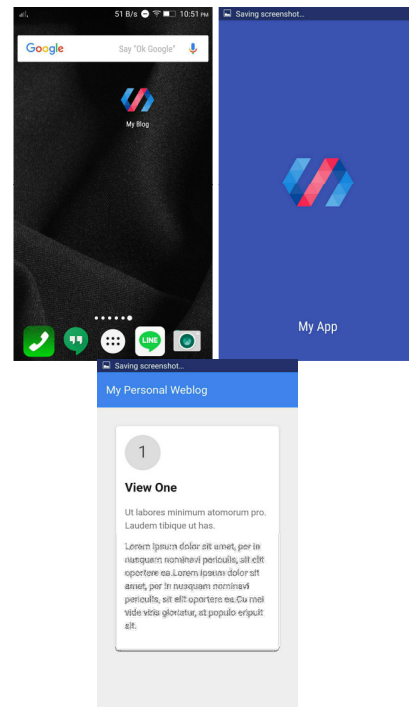Figure 2. Progressive Web App in computer screen

Figure 3. The Mobile view of the Progressive Web Apps

IV. CONCLUSION

After we experimented in the section before, we conclude that the development of progressive web apps is quite understandable and give a new experience of a web

application. This could also engange the users of websites and low storage capacity of mobile phone.

REFERENCES

[1]   https://blog.ionic.io/what-is-a-progressive-web-app/ accessed on May 5th, 2017

[2]   https://cmsphoto.ww-cdn.com/superstatic/41269/art/default/10172541-16584509.jpg?v=1476195507 accessed on May 5th, 2017

[3]   https://speckyboy.com/progressive-web-applications/ accessed on May 5th, 2017

STATEMENT

I hereby declare that this paper is my own work and not a copy, translation, nor plagiarism of somebody else's work.

Bandung, May 5th, 2017

Harry Alvin Waidan Kefas