Progressive Web Apps Development and Analysis with Angular Framework and Service Worker for E-Commerce System

Zulkifli Tahir, Amil Ahmad Ilham, Muhammad Niswar, Adnan, Andi Ahmad Fauzy
Department of Informatics
Hasanuddin University
Makassar, Indonesia
{zulkifli, amil, niswar, adnan}@unhas.ac.id, fauzyaa14d@student.unhas.ac.id

***Abstract*— E-commerce is a small, medium, and large business that sells and buys products online. In Indonesia, e-commerce has become one of the drivers of the nation's economic development. On the other hand, the development of information technology is growing rapidly. Many technologies have claimed to be reliable in developing web systems. In this research, reliable information technology has been implemented by using Progressive Web Apps (PWA). This technology combines the Angular framework as a user interface development, the Angularfire library as a link to the server, and a Service Worker to support the website in various network conditions. The process of performance testing and analysis of the system has been carried out with two types of systems, with Service Worker and without Service Worker, including the response time, throughput, and latency. Base on the results, it shows that this technology is fast and reliable, and it is suitable for the implementation of e-commerce systems.**

Keywords*— *E-Commerce, Progressive Web Apps, Angular, Service Worker

I. INTRODUCTION

E-commerce has become a determinant of success in business activities for small, medium, and large-scale sellers in today's era. In developing countries, the number of e-commerce users continues to grow by many businesses, ranging from big cities, small cities to even village scales [1]. As one of the developing countries, Indonesia has a great opportunity for economic development with e-commerce among Asian countries which is estimated to ranks third after China and India [2].

Even though Indonesia is experiencing an economic recession, after it was reported that its economic growth contracted by 3.49% and 2.07% in the third and fourth quarters of 2020, the information and communication business sector, on the other hand, increased by 10.58% [3]. The use of e-commerce as an alternative for people to meet their needs during the pandemic continues to increase. It is supported by the increasing trend of online shopping, since the pandemic, it has been observed that the number of sellers and transactions in the marketplace has skyrocketed. During the corona pandemic, *Bukalapak* and *Tokopedia* as well-known e-commerce in Indonesia claimed, their partner income had increased by 15 times since joining [4].

Therefore, there will be more brands and manufacturers from various sectors that will spread their wings to e-commerce, considering that currently they increasingly understand the importance of digital channels such as e-commerce for the sustainability of their business. Currently, 41 e-commerce Startups in Indonesia have been recognized in the global world, and that number will continue to increase [5].

In Indonesia, more than 60% of Indonesians use smartphones in e-commerce matters, the rest is through other devices such as computers, tablets, and others. Based on a research survey, after 30 days of someone installing an application on a smartphone, 84% of the respondents said that they use no more than ten applications a day, 55% of the respondents said no more than two applications a day [6]. This reason indicates that smartphone device users prefer to browse rather than have to install applications. With such facts, of course, developing an application for e-commerce is still a challenge nowadays.

Then, accessibility on the website still continues to be an issue for internet users [7]. In recent times, many companies are facing the problem of developing different applications for different platforms. They often need to develop two types of mobile application operating systems, one with iOS and another for Android. In addition, they need a web application that works well on computers and mobile devices.

In this research, a Progressive Web Apps (PWA) technology from Google is used which is expected to be able to answer those challenges. A PWA is a revamped web system with additional features and functionality that gives it an app-like look (like a smartphone app) displayed in a web browser. This technology also uses the 'Service worker' function to ensure smooth operation and increase the load speed of the website, and even it can be accessed without an internet connection at all. With the Service Worker API, it ensures that it caches almost all website data for use when revisited, the data is not reloaded so that data usage over the internet will be reduced [8]. The PWA is like a smartphone application but is a website that can be a novelty in the development of e-commerce applications.

II. E-COMMERCE SYSTEM

A. E-commerce

E-commerce or electronic commerce is a collection of technologies, applications, and businesses that connect companies or individuals as consumers to conduct electronic transactions, exchange materials, and exchange information through the internet or television, www, or other computer networks [9]. This e-commerce activity is the application of e-business related to commercial transactions, for example,

electronic transfer of funds, Supply Chain Management (SCM), Electronic Data Interchange (EDI), online marketing, or e-marketing, online transaction processing, etc.

In this work, the system is built as an e-commerce web system for businesses ranging from small to large. This system will provide the ability for businesses to sell and promote their products online. E-commerce that runs web technology generally has the same problem that it cannot be accessed if the network is not connected to the server. So, this development uses web technology that can still work offline.

B. Progressive Web Apps (PWA) with Service Worker

PWA is the development of several API technologies in one code place. It is similar to application platforms but the run-on top of a browser. The PWA has many advantages over the APIs used, including WebRTC, geolocation, push notifications, and many others including new upcoming APIs. This technology also makes access to the system more reliable and faster through the network. Coupled with the ability to keep working on a slow network or in an offline state which often happens when we want to access the server [10].

This PWA allows e-commerce users to delay the delivery of an action performed on the website if there is no internet connection, and the action will be performed when an internet connection is available. For example, if you buy products on an e-commerce system when you want to buy an item, you add it to the cart, but if the network doesn't exist, then the background sync from the Service Worker will work to delay the action until the internet connection reappears.

Plus, PWA-based web applications no longer need to be updated like native applications. The latest updates will continue to be available for users as long as developers update their apps.

C. Angular

Angular is a platform using the Typescript language to develop front-end web applications and is open source. The platform was developed by the Google team and the Angular open-source community [11]. Angular is designed for all platforms including web, mobile web, native mobile, and native desktop. It is built for speed, scalability when you need big data and it is already widely used by web developers around the world [12]. To connect Angular with Firebase, Angularfire library is used as a backend service. It can provide storage, authentication, and deployment services [13].

III. PWA System Development

In the development of this system, several technological instruments and research specifications are used, including hardware and software. Hardware uses a computer with an Intel Core i5 2.5 GHz processor, 4 GB of DDR3 1600 RAM, and an Inter Graphic 4000 1536 MB GPU. The software and

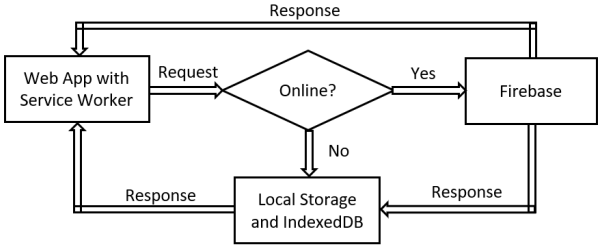

Fig. 2. PWA System Workflow

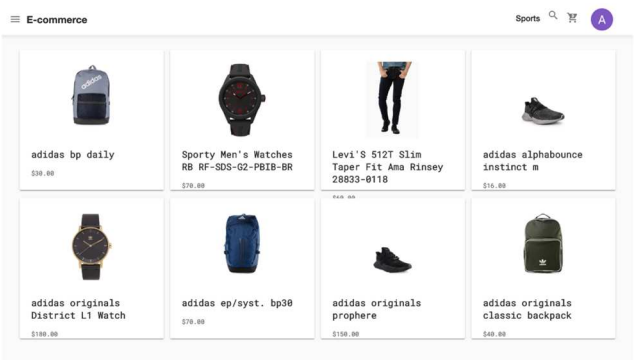

Fig. 1. PWA User Interface

technology used are Angular, Angularfire, NodeJS, Local Storage, and IndexedDB. The system is also analyzed over a network with a maximum bandwidth of 20 Mbps or data transfer of 2.5 MBps.

The system is made according to the flow in Fig. 1. Web App with Service Worker will send data request then check whether the website is online or offline. If offline, the Service Worker will retrieve data on the local system and cache storage and then respond to the website to be displayed in the browser. When online, the Service Worker fetches data in Firebase Storage. Then from there, a response will be sent to the website to be displayed in the browser and synchronize the latest data in Firebase and enter it into the Local System for updating the data in it which works if the website is accessed offline again, then the latest data can be retrieved from the local system.

In the local systems, Local Storage and IndexedDB is used. The data stored in Local Storage is in the form of image data, videos, or other files regarding the products on the e-commerce website. IndexedDB is used to store text data such as user data, tokens, and other important data.

For the user interface, this system applies Single Page Application (SPA) technology. One of user interface of the system is shown in Fig. 2.

IV. PWA System Analysis

After the system has been built, data of system performances are collected in the form of Response Time, Throughput, and Latency. The data are analyzed with two types of systems, namely with Service Worker and without Service Worker.

Response time is calculated from the value of the time interval when the request is made and when the response is received. Response time testing is obtained by providing many requests at once from the client to the server for each thread then calculating the average response time, as in (1).

$$Average = \frac{1}{n}\sum_{i=1}^{n} a_i = \frac{1}{n}\,(a_1 + a_2 + \cdots + a_n) \qquad (1)$$

n : the number of requests from the client to the server.

i : *counter number* (i = 1,2, …., n)

a : time interval between client to server for each request

$$Throughput = \frac{Number\ of\ requests}{Total\ time\ x\ Number\ of\ requests} \quad (2)$$

Throughput is the total number of transactions that can be carried out within a certain time or transactions per second. In throughput testing, threads are given repeatedly and the throughput value of each experiment is taken. Then, the calculation of the average value of the throughput results of each trial for each thread is carried out, as in (2).

Latency is defined as the duration between the end of the request or the start of the request when the response from the server is received. Latency is part of response time. What distinguishes it is the response time of the request process until the response is received by the user.

A. Response Time Analysis

One of the results of recording response time from a web menu is shown in Table I. The graph obtained from each experiment on accessing website URLs with and without Service Worker in Fig. 3. The x-axis represents the number of threads used which can be interpreted as the number of requests for each attempt per unit of time. Then the y-axis shows the average response time in milliseconds. The graph shows that the average response time required for websites with Service Worker is lower than websites without Service Worker of each experiment with a different number of threads.

TABLE I. DATA OF RESPONSE TIME

Test Data	Number of Threads	Response Time	
		With Service Worker	*Without Service Worker*
1	1	185	319,332
2	50	223,68	365,547
3	100	260,21	452,71
4	200	267,91	524,46
5	300	324,08	766,17
6	400	497,885	957,95
7	500	553,662	1543,773
8	600	667,49667	2274,7392
9	700	1022,974	2452,11734

B. Throughput Analysis

Based on one of the test data of the system shown in Table II, the graph obtained from each experiment on website URL access (Fig. 4). The x-axis in the graph also shows the number of threads for each trial, while the y-axis shows the average number of throughputs generated in units of transactions per second (tps). All average throughputs resulting from accessing websites without Service Worker are smaller than websites with Service Workers from all experiments with the different number of threads ranging from 1 to 900. Although the difference is not significant, this shows that a website using Service Worker can produce higher throughput, so it can process more requests in a unit of time.

C. Latency Analysis

Based on data latency from Table III, the graph depicted from each experiment on accessing website URLs in Fig. 5. The x-axis shows the number of threads, and the y-axis shows the average latency in milliseconds. The latency generated from websites without Service Worker is greater than websites with Service Worker. This shows that a website using Service Worker is able to produce smaller latency, so it is able to process requests and receive faster responses.

TABLE II. DATA OF THROUGHPUT

Test Data	Number of Threads	Throughput	
		With Service Worker	*Without Service Worker*
1	1	6,5	4,2
2	50	41	32
3	100	51,5	38,5
4	200	88,6	60,4
5	300	122,1	90,5
6	400	158,3	95,55
7	500	151,1	120,76
8	600	157,7	110,234
9	700	151,2	125,39
10	800	189,6	150,71
11	900	208,7	162,1
12	1000	173,6	180,6

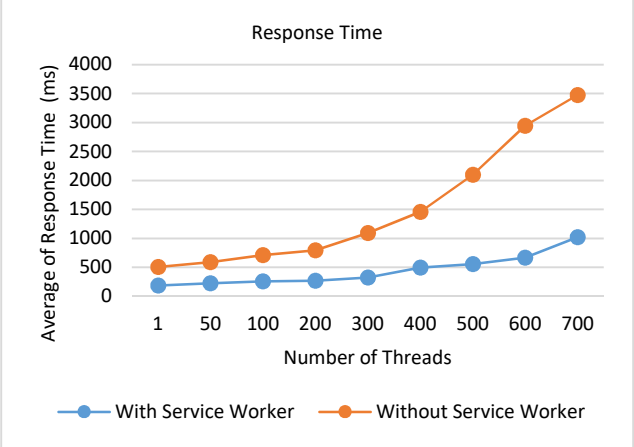

Fig. 3. Response time with and without Service Worker

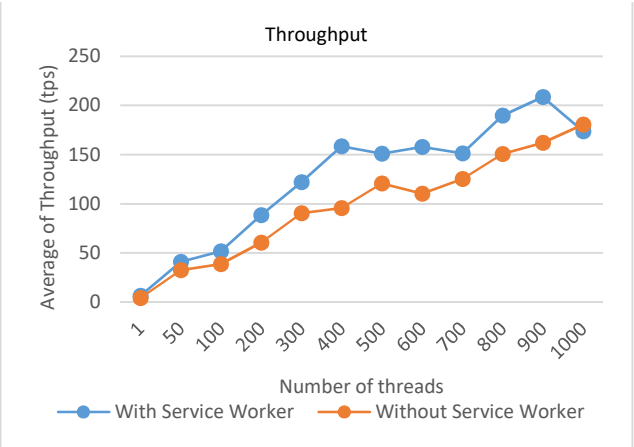

Fig. 4. Throughput with and without Service Worker

TABLE III. DATA OF LATENCY

Test Data	Number of Threads	Latency	
		With Service Worker	*Without Service Worker*
1	1	52	85,3
2	50	75,24	105,1
3	100	86,61	153,9
4	200	80,515	213,8
5	300	99,03	463,7
6	400	118,435	330,2
7	500	118,066	452,8
8	600	110,78667	484,4
9	700	159,40714	473,8
10	800	205,908	414,2
11	900	226,7211	488,9
12	1000	262,423	628,7

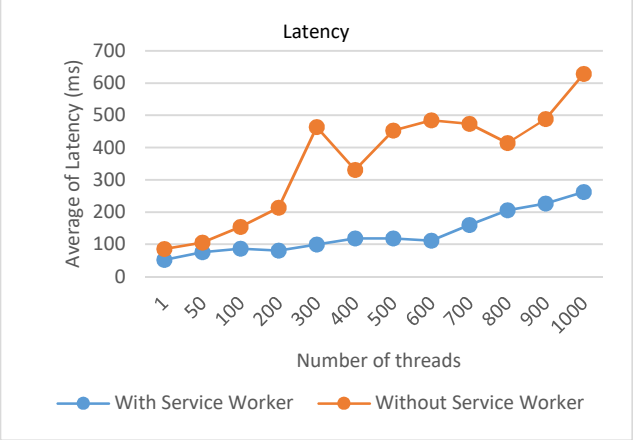

Fig. 5. Latency with and without Service Worker

V. CONCLUSION

This work has successfully implemented web technology for fast and reliable e-commerce using PWA. The web technology integrates the Angular framework, Angularfire library, and Service Worker. Based on web system testing with and without Service Worker, it shows significant differences in response time, throughput and latency. By using a Service Worker, it results in a smaller average response time, greater throughput and lower latency than websites without a Service Worker. This means that the developers that using Service Worker can improve web system performance to be faster and more reliable. Future research will use several different web frameworks such as React, Vue, Flutter or Svelte.

ACKNOWLEDGMENT

All authors would like to thank for Institute for Research and Community Service, Hasanuddin University for assistance with the research and funding process.

REFERENCES

[1] N. Kshetri, Rural e-commerce in developing countries. It Professional, 20(2), pp.91-95, 2018.

[2] H. Harsono, "Indonesia Will Be Asia's Next Biggest e-Commerce Market," TechCrunch, blog, 2016; https://techcrunch.com/2016/07/29/indonesia-will-be-asiasnext-biggest-e-commerce-market.

[3] Badan Pusat Statistik, https://www.bps.go.id/pressrelease/2021/02/05/1811/ekonomi-indonesia-2020-turun-sebesar-2-07-persen--c-to-c-.html. , 2021

[4] Mime Asia, https://www.mime.asia/bukalapak-and-tokopedia-record-msme-partner-transactions/ , 2021

[5] 500 Website, "500 Startups is more than just a name", https://500.co/, 2021

[6] I. Malavolta et. al. "Assessing the Impact of Service WorkerS on the Energy Efficiency of Progressive Web Apps". IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft), 2017.

[7] J. Vanhala, "Implementing an Offline First Web Application". VAASAN Aalto University, Finlandia, 2017.

[8] J. Korva, "Developing a web application with Angular 2: Graphical editor for Happywise's Cove Trainer," 2016.

[9] R. S. Mishra. "Progressive WEBAPP: Review," International Research Journal of Engineering and Technology (IRJET). e-ISSN: 2395 -0056, 2016.

[10] O. Adetunji, C. Ajaegbu, N. Otuneme, O.J. Omotosho. Dawning of Progressive Web Applications (PWA): Edging Out the Pitfalls of Traditional Mobile Development. American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS). 2020 May 22;68(1):85-99.

[11] E. Saks, JavaScript Frameworks: Angular vs React vs Vue, 2019.

[12] Angular Home. URL: https://angular.io/

[13] Angularfire Github https://github.com/angular/angularfire