Online version at https://journal.lenterailmu.com/index.php/josapen

JOSAPEN

E-ISSN: 3031-2272 (Online)

Implementation of a Progressive Web App for Stock Opname Management at Cakra Abhipraya

Rafi Bowo Raharjo [1], Kiki Setiawan [2], Mesra Betty Yel [2]

[1] *Department of Informatics, Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika (STIKOM CKI), Jakarta, Indonesia*

[2] *Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika (STIKOM CKI), Jakarta, Indonesia*

ARTICLE INFO

Article history:
Received July 2026
Revised August 8, 2026
Accepted August 11, 2026

Keywords:
Progressive Web App
Stock Opname
Inventory
Google Apps Script
Agile Development

A B S T R A C T

The advancement of information technology in the digital era encourages organizations to improve inventory management efficiency through modern and flexible web-based systems. Cakra Abhipraya still faces challenges in the stock opname process, including manual data recording, delays in stock information synchronization between divisions, and difficulties accessing the system via mobile devices in the field. These issues cause inventory monitoring to be less effective and prone to errors in stock data. This research aims to implement a Progressive Web App (PWA) on a web-based stock opname system at Cakra Abhipraya to improve efficiency, flexibility, and accessibility of inventory management. The system was developed using the Agile Development methodology with HTML5, CSS3, and plain JavaScript as the frontend, and Google Apps Script integrated with Google Spreadsheet as the backend. The PWA concept is applied through service workers and a web app manifest to support offline capability, responsive interface, and installability. Testing was conducted using the Black-Box Testing method and a Google Lighthouse audit. The results show that the system achieved a perfect PWA score (100) in the Google Lighthouse audit, improved the stock opname process efficiency by 73% compared to the manual method, and achieved an average user satisfaction score of 4.3 out of 5. The implementation of this system has proven to significantly support the digitalization of inventory management at Cakra Abhipraya.

1. Introduction

Advancements in information technology during the digital era have encouraged organizations to adopt web-based information systems to manage data and business processes more efficiently and in an integrated manner [1]. Traditionally, users have had two primary application options, each with its own limitations: native applications, which require installation and consume device storage, and browser-based applications, which are generally lighter but may provide a less reliable user experience under limited network conditions, such as 2G or 3G connectivity [1]. These limitations have contributed to the emergence of Progressive Web Apps (PWAs), an approach that combines the

* *Corresponding author:* Kiki Setiawan
email: ki2djoaz@stikomcki.ac.id

advantages of web and mobile applications while addressing limitations associated with both [1,2]. PWAs provide features such as offline and low-connectivity support, an app-like user experience, up-to-date content, secure access through HTTPS, search engine discoverability, and re-engagement through push notifications [2,3].

Cakra Abhipraya is an organization that manages various social and humanitarian programs requiring structured inventory and stock management. Although the organization has undertaken digitalization efforts, including the implementation of a web-based portal for publication and donation transparency [4], observations indicate that challenges remain in its stock-taking processes. These challenges include manual stock data recording, delays in synchronizing inventory information across divisions, and limited accessibility through mobile devices during field operations. Similar problems have been reported in organizations that rely on manual record-keeping and conventional electronic spreadsheets, including data discrepancies, duplicate item names, and reporting processes that are time-consuming and prone to errors [5–7].

Previous studies have demonstrated the applicability of PWAs in various domains, including online learning media [8], student e-portfolio repositories [9], agricultural field data collection [10], and inventory management using barcode or QR-code scanning [11]. PWAs have also been applied in e-commerce inventory management using the Periodic Review System method [12]. However, existing research on web-based stock-taking systems has generally relied on linear development approaches such as the Waterfall method [7,12] or has not incorporated PWA technology, resulting in systems that are primarily accessed through desktop computers [6,7]. These limitations reduce system flexibility, particularly for staff conducting inventory activities in the field. Therefore, this study aims to implement PWA technology in the web-based stock-taking system at Cakra Abhipraya, using Google Apps Script as the backend and Google Sheets as the data storage platform. The proposed system is intended to improve inventory management efficiency, flexibility, and accessibility, while its functionality and PWA capabilities are evaluated using Black-Box Testing and Google Lighthouse audits [13,14].

2. Method

This study employs a Research and Development (R&D) approach with an Agile-based system development methodology. Agile was selected because of its iterative and incremental nature, which enables the development process to accommodate changing user requirements throughout each iteration [15,16]. The methodological approach is supported by previous system development studies, including the application of the Agile/Scrum model in the development of a stock-taking system for a building supply store [15] and the division of product backlogs into sprint backlogs in the development of a stock-taking application integrated with SAP Business One [16]. Accordingly, the development process in this study consists of five main phases that are performed iteratively: (1) Planning and Requirements Analysis, (2) System Design, (3) Implementation, (4) Testing, and (5) Deployment and Evaluation. The overall development process is illustrated in Figure 1.

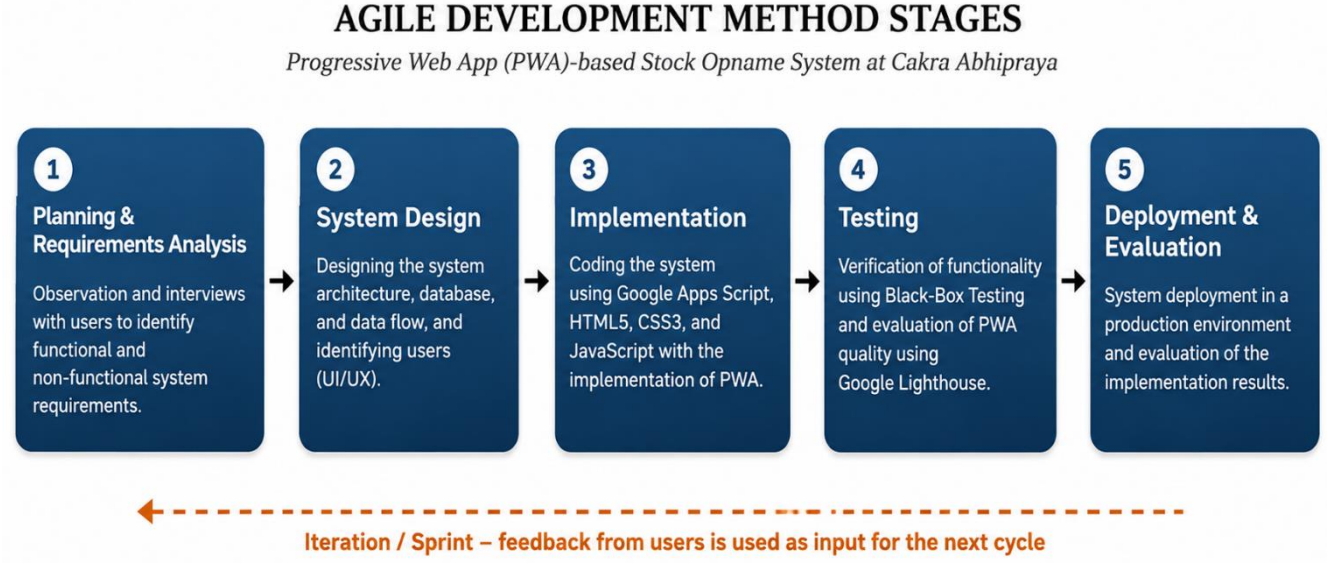

Figure 1 – Agile Development Method Stages

A needs analysis was conducted through direct observation, interviews with Cakra Abhipraya's administrative and operational staff, and a review of documentation related to inventory management procedures. The functional requirements identified through this process are presented in Table 1 and include two-level access authentication, a real-time statistical dashboard, item data management, an item request system with an approval workflow, and program and activity data management. The non-functional requirements specify a minimum Google Lighthouse score of 80 for the Performance and Accessibility categories, as well as an API response time of less than two seconds. These performance thresholds were established based on PWA quality evaluation practices reported in previous studies [9,12,14].

JOURNAL OF COMPUTER SCIENCE APPLICATION AND ENGINEERING VOL. 4, NO. 2, JULY 2026, PP. 31~36

Table 1 – Analysis of system functional requirements

Code	Functional Requirement	Priority	Status
KF-01	Multi-level authentication (Admin & Staff)	High	Fulfilled
KF-02	Real-time inventory statistics dashboard	High	Fulfilled
KF-03	Item data management (CRUD & search)	High	Fulfilled
KF-04	Item request system with an approval workflow	High	Fulfilled
KF-05	Program/activity management	Medium	Fulfilled

The system design employs a web-based client–server architecture using a serverless approach, with Google Apps Script handling server-side logic and Google Sheets serving as the data storage platform. This approach is particularly suitable for organizations with limited technical infrastructure and budgetary resources [17,18]. The database structure consists of four primary entities: Users, Items, Programs, and Requests, with each entity stored in a separate sheet [7].

The PWA implementation consists of three main technical steps. First, a service worker is registered to implement caching strategies, using a Cache First strategy for static assets and a Network First strategy for API requests [2,3]. Second, a manifest.json file is configured with the required application metadata, including 192 × 192 and 512 × 512 pixel icons and the standalone display mode [3,19]. Third, essential application assets, including HTML, CSS, JavaScript, and application icons, are precached during service worker installation to improve application availability and loading performance [2].

System evaluation is conducted using two complementary methods. First, Black-Box Testing is used to verify system functionality from the user's perspective, with test scenarios developed based on the equivalence partitioning technique [13]. Second, Google Lighthouse is used to evaluate PWA quality across the Performance, Accessibility, Best Practices, and SEO categories, with scores reported on a 0–100 scale [14]. Both evaluation methods are performed at the end of each Agile iteration, allowing identified deficiencies to be addressed before the system proceeds to the subsequent iteration or final deployment.

3. Result and Discussion

3.1. System Implementation

The system backend was developed using five primary functional modules within Google Apps Script to support the main inventory management processes. The first module, Auth, is responsible for handling user authentication and session management to control access to the system. The second module, Items, supports CRUD (Create, Read, Update, and Delete) operations as well as inventory data retrieval, enabling users to manage and access item information efficiently. The third module, Requests, manages automated identification generation and approval workflows for item requests. The fourth module, Programs, is used to manage program and activity data associated with inventory activities. Finally, the Dashboard module provides real-time statistical aggregation, allowing users to monitor inventory-related information through summarized statistical data. This modular functional structure is consistent with data management and stock control patterns reported in previous research on web-based inventory management information systems [7].

The user interface was developed using HTML5, CSS3, and vanilla JavaScript to provide a responsive and accessible interaction environment for system users. A mobile-first design approach was adopted to ensure that the interface can be used effectively on mobile devices, particularly by staff who may access the system during field activities. In addition, a consistent dark-theme interface was implemented across different device types to maintain visual consistency throughout the system. The combination of the modular backend structure and mobile-oriented frontend design supports the system's objective of providing an efficient, flexible, and accessible inventory management solution.

3.2. The Result of Black-Box Testing

Black-box testing was conducted using 45 test scenarios designed to evaluate the functional performance of the developed system from the user's perspective. The scenarios covered several core processes, including user login, inventory data management, item submission, approval processes, and status updates. Each scenario was executed by providing the specified input and comparing the resulting system output with the expected outcome. The testing results showed that all 45 scenarios were successfully executed and produced the expected outputs. Furthermore, no critical bugs or functional errors were identified that could interfere with the system's primary operations.

The 100% test success rate indicates that the system was able to fulfill all functional requirements represented in the tested scenarios. This result also demonstrates the effectiveness of the Agile iterative development approach implemented in this study, in which each functional module was tested immediately after its implementation. By identifying and addressing potential errors during individual development iterations, functional problems could be resolved before being carried over to subsequent stages of development. This iterative testing process therefore helped minimize the accumulation of functional errors and contributed to the system's readiness for final testing and deployment [16].

3.3. Google Lighthouse Test Results

The Google Lighthouse audit results demonstrate satisfactory performance across all evaluated categories, with a perfect PWA score of 100. This result indicates that the system has successfully satisfied the assessed technical PWA criteria, including the presence of an active service worker, a valid web app manifest, secure HTTPS access, and support for offline functionality [2,14]. The complete Lighthouse audit results are presented in Table 2.

Table 2 – Google Lighthouse Test Results

Category	Score	Rating	Description
Performance	87	Needs Improvement	Loading time: 1.8 seconds
Accessibility	92	Excellent	Complete ARIA labels
Best Practices	95	Excellent	Modern web standards met
SEO	89	Needs Improvement	Optimized meta tags
PWA	100	Excellent	All PWA criteria met

The Performance score of 87, with a loading time of 1.8 seconds, exceeded the minimum threshold of 80 established during the requirements analysis stage. Although this result is slightly lower than that reported in a comparable study, which improved its Performance score from 62% to 98% [9], the obtained score still indicates satisfactory system performance. The difference can reasonably be attributed to the system's dependence on communication with Google Apps Script and Google Sheets, which may introduce additional latency compared with backend technologies such as Express.js or Next.js [9,20]. Despite this limitation, the achieved loading time demonstrates that the system remains sufficiently responsive for its intended inventory management activities.

The Accessibility score of 92 also exceeded the predefined minimum threshold, indicating that the system adequately addresses important accessibility requirements, including the appropriate use of ARIA labels for interactive elements. This result is consistent with the importance of web accessibility guidelines, such as WCAG, in ensuring that digital systems can be accessed and used by a broad range of users [21]. Meanwhile, the SEO score of 89, although relatively lower than the other categories, remains acceptable considering that the developed system is intended as an internal organizational application rather than a publicly indexed website. This differs from public-facing platforms, such as the Cakra Abhipraya humanitarian portal, where search engine visibility is more relevant [4]. Finally, the Best Practices score of 95 indicates strong compliance with modern web development practices, including the implementation of HTTPS, which represents one of the fundamental security characteristics required for Progressive Web Applications [2,19].

3.4. Impact of Implementation on Efficiency and User Satisfaction

The comparison of the stock-taking process involving 50 items demonstrated a substantial improvement in operational efficiency after the implementation of the proposed system. Using the conventional manual method, the stock-taking process required an average of 45 minutes, whereas the same process using the developed system required only 12 minutes. This represents a reduction of approximately 73% in processing time, indicating that the system can significantly accelerate routine inventory activities. The shorter processing duration allows staff to complete stock-taking activities more efficiently and provides additional time for other operational responsibilities.

The improvement in processing time can be attributed to several features incorporated into the system. First, the elimination of manual recording reduces the time required to write down and subsequently re-enter inventory information. Second, automated data validation helps minimize repetitive verification activities and reduces the possibility of errors during data entry. Third, the PWA-based architecture allows staff to access the system directly through mobile devices while conducting stock-taking activities in the field. Consequently, staff are not required to return to a computer workstation to record or update inventory data, thereby streamlining the overall stock-taking workflow. These advantages are consistent with previous research on PWA-based inventory management and the Periodic Review System, which highlights the potential of digital inventory systems to improve stock control and reduce the risks of stockouts and overstocking [12].

The findings also demonstrate that the developed system addresses several operational problems reported in previous studies of web-based inventory management. In particular, the automated submission and approval workflow can help reduce delays associated with manual inter-departmental document processing and approval [7]. The system also reduces the administrative burden associated with conventional stock-taking practices. This is particularly relevant when compared with MSME case studies in which manual stock-taking required approximately six to eight hours per session and resulted in data discrepancies ranging from 10% to 15% [22]. By integrating data recording, validation, and reporting into a single system, the proposed solution provides a more structured mechanism for managing inventory information and supporting stock-taking activities.

In addition to operational efficiency, the post-implementation evaluation indicates a positive level of user acceptance. The user satisfaction survey produced an average score of 4.3 out of 5, with most respondents reporting that the developed system was considerably easier to use than the previous manual method. This result is consistent with previous studies examining user acceptance and engagement with PWA-based systems [9,20,23]. Similar findings have also been reported in web-mobile stock-taking and monitoring systems, where users generally demonstrated positive acceptance while still identifying several aspects requiring further improvement [5]. Furthermore, research on PWA user engagement reported that approximately 43% of participants in an e-commerce platform preferred PWAs over native applications because they did not require a separate installation process, while PWA adoption was associated with the potential to increase user retention by up to 35% compared with conventional applications [23]. These findings support the suitability of the PWA approach for systems that require convenient and flexible access across different devices.

Overall, the implementation results indicate that the developed system provides improvements across three interconnected dimensions: operational efficiency, data accuracy, and system accessibility. The reduction in stock-taking time demonstrates improved operational efficiency, while automated data validation and centralized data management

contribute to more consistent and reliable inventory information. At the same time, mobile accessibility through the PWA architecture enables staff to perform inventory-related activities more flexibly, including during field operations. These three dimensions reinforce one another because faster processing is more valuable when the resulting data are reliable and readily accessible to users. Therefore, the combination of automated recording, data validation, streamlined workflows, and multi-device accessibility provides an integrated solution for improving inventory management at Cakra Abhipraya.

5. Conclusion

The implementation of a Progressive Web App (PWA) for Cakra Abhipraya's stock-taking system was successfully completed using Google Apps Script for the backend and HTML5, CSS3, and JavaScript for the frontend. The system achieved a perfect PWA score of 100 in Google Lighthouse, indicating that it satisfied the assessed technical PWA criteria. The integration of service workers and a web app manifest supported key PWA capabilities, including offline access and installability, while the system achieved satisfactory scores for Performance (87), Accessibility (92), and Best Practices (95). The adoption of an Agile-based development methodology also enabled iterative refinement of the system based on users' actual requirements and feedback. Furthermore, all 45 Black-Box Testing scenarios were successfully executed without identifying critical functional bugs. The system reduced the average stock-taking time by approximately 73% compared with the manual method, while the post-implementation user satisfaction score reached 4.3 out of 5.0.

These findings demonstrate that the Google Apps Script-based PWA approach can provide an efficient, affordable, and practical solution for non-profit organizations with limited technological infrastructure and resources. The developed system also complements Cakra Abhipraya's ongoing digitalization initiatives, which previously included the development of a social publication and donation transparency portal. Nevertheless, several opportunities remain for further development. Future improvements may include the integration of push notifications, enhanced analytics and reporting features, migration to a more scalable database to accommodate larger data volumes, and barcode or QR-code scanning capabilities to further accelerate physical stock recording during field activities.

REFERENCES

[1] S. S. Tandel and A. Jamadar, "Impact of Progressive Web Apps on Web App Development," Int. J. Innov. Res. Sci. Eng. Technol. (IJIRSET), vol. 7, no. 9, pp. 9439–9444, Sep. 2018.

[2] A. Biørn-Hansen, T. A. Majchrzak, and T.-M. Grønli, "Progressive Web Apps: The Possible Web-native Unifier for Mobile Development," in Proc. 13th Int. Conf. Web Inf. Syst. Technol. (WEBIST), 2017, pp. 344–351.

[3] V. Sharma, R. Verma, V. Pathak, M. Paliwal, and P. Jain, "Progressive Web App (PWA) – One Stop Solution for All Application Development Across All Platforms," Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol. (IJSRCSEIT), vol. 5, no. 2, pp. 1–3, 2019, doi: https://doi.org/10.32628/CSEIT1952290.

[4] R. B. Raharjo, D. I. Yusuf, E. Poerwandono, et al., "Pengembangan Portal Web Kemanusiaan untuk Publikasi Sosial dan Transparansi Donasi Cakra Abhipraya," AJAD: J. Pengabdian kpd. Masy., vol. 6, no. 1, pp. 13–21, Apr. 2026, doi: https://doi.org/10.59431/ajad.v6i1.704.

[5] Syahrudin and Sudewi, "Aplikasi Pemantauan dan Pengelolaan Stock Opname Berbasis Web Mobile di Percetakan Squad Computer," J. Market. Bus. Intell. (JMBI), vol. 3, no. 1, pp. 14–22, Mei 2025.

[6] M. R. Fazillah and J. Devitra, "Sistem Informasi Pemantauan Inventory Stock Opname Berbasis Web Pada SBU Teh Kemasan PTPN VI," Jurnal Manajemen Sistem Informasi, vol. 7, no. 4, pp. 644–656, Dec. 2022.

[7] E. Kurniawati and A. Ikhwan, "Perancangan Sistem Informasi Manajemen Inventaris Kontrol Stok Barang Berbasis Web," J. Teknol. Sist. Inf. dan Apl. (JTSI), vol. 6, no. 3, pp. 408–415, Jul. 2023, doi: https://doi.org/10.32493/jtsi.v6i3.30881.

[8] G. Matiini, R. Setiyadi, A. Setiawan, and M. Ramli, "Pengembangan Aplikasi Progressive Web Application (PWA) untuk Pembelajaran dan Evaluasi Kelas English Grammar Online Course," J. Pendidik. Edutama (JPE), vol. 8, no. 2, pp. 163–176, Jul. 2021.

[9] S. Aripin and Somantri, "Implementasi Progressive Web Apps (PWA) pada Repository E-Portofolio Mahasiswa," J. Eksplora Inform., vol. 10, no. 2, pp. 148–158, 2021.

[10] P. Koysawat, et al., "Progressive Web App for Crop Field Data Collection," IOP Conf. Ser.: Mater. Sci. Eng., vol. 1163, art. 012018, 2021, doi: https://doi.org/10.1088/1757-899X/1163/1/012018.

[11] A. Desai, "Enhancing Inventory Management with Progressive Web Applications (PWAs): A Scalable Solution for Small and Large Enterprises," arXiv preprint arXiv:2506.11011, 2025.

[12] A. F. Siagian and Suendri, "Applying the Periodic Review System Method in Progressive Web Apps for E-Commerce Inventory Management," J. Inf. Syst. Informatics (JournalISI), vol. 7, no. 2, 2025, doi: https://doi.org/10.51519/journalisi.v7i2.1117.

[13] Muhammad Zen, Irwan, Hafni, and M. Dea Putra Ananda, "Implementasi dan Pengujian Menggunakan Metode BlackBox Testing Pada Sistem Informasi Tracer Study," Bulletin of Computer Science Research, vol. 4, no. 4, pp. 327–340, Jun. 2024, doi: https://doi.org/10.47065/bulletincsr.v4i4.359.

[14] R. V. Rochim, A. Rahmatulloh, R. R. El Akbar, and R. Rizal, "Performance Comparison of Response Time Native, Mobile and Progressive Web Application Technology," Innov. Res. Informatics (INNOVATICS), vol. 5, no. 1, pp. 36–43, 2023.

[15] Syech, M. Sandy, and W. Haryono, "Perancangan Sistem Stokopname Berbasis Web pada Toko Bangunan Mutiara," Repeater: Publ. Tek. Inform. dan Jaringan, vol. 3, no. 3, pp. 32–46, Jul. 2025, doi: https://doi.org/10.62951/repeater.v3i3.545.

[16] A. Ramadhan, M. Lubis, W. Puspitasari, and A. R. Lubis, "Development of Web Stock Opname Application With SAP Business One Using Scrum Method," in Proc. 2019 Int. Conf. Comput. Sci. Inf. Technol. (ICoSNIKOM), 2019, pp. 1–5.

[17] A. M. Adzani, C. P. A. Mulya, F. Ahluna, S. Febrianti, Y. Akbar, and M. B. Yel, "Sistem Inventory Barang Gudang Berbasis Web Studi Kasus Yayasan Indonesia Care," AJAD: J. Pengabdian kpd.

Masy., vol. 4, no. 1, pp. 76–83, Apr. 2024, doi: https://doi.org/10.59431/ajad.v4i1.284.

[18] V. Arinal, Y. Akbar, F. M. Sarimole, A. Rosiana, D. S. Arroufu, N. A. Khairudin, and R. Adawiyah, "Implementasi Sistem Informasi Pengelolaan Bantuan Warga Kampung Pulojahe Jakarta Timur Berbasis Web," SELAPARANG: J. Pengabdian Masy. Berkemajuan, vol. 6, no. 1, Mar. 2022.

[19] R. S. Mishra, "Progressive WEBAPP: Review," Int. Res. J. Eng. Technol. (IRJET), vol. 3, no. 6, pp. 1–5, Jun. 2016.

[20] Ronny Jubhari Phie Joarno, Mohammad Fajar, and Arfan Yunus, "Implementasi Progressive Web Apps Pada Website Gethelp Menggunakan NEXT.JS," Jurnal KHARISMA Tech, vol. 17, no. 2, pp. 1–15, Sep. 2022.

[21] A. Muawwal, "Evaluating Progressive Web App Accessibility for People with Disabilities," 2024.

[22] F. Firdaus, M. Komaro, and V. Dwiyanti, "Rancang Bangun Sistem Informasi Stock Opname Berbasis Web dengan Metode Waterfall pada UMKM," JUTIK: Jurnal Teknologi Informasi dan Komputer, vol. 11, no. 1, pp. 45–56, Apr. 2025.

[23] D. K. Ebechue, "Progressive Web Applications and Users' Engagement: Investigating the Impact of Progressive Web Applications on Users' Engagement," B.Eng. thesis, Degree Programme in Information Technology, Oulu University of Applied Sciences, Oulu, Finland, 2021.