STM JOURNALS

Journal of
Software Engineering Tools & Technology Trends

ISSN: 2394-7292
Volume 13, Issue 1, 2026
January–April
DOI (Journal): 10.37591/JoSETTT

https://journals.stmjournals.com/josettt

Review JoSETTT

A Modern Approach to Web Application Development Using PWA, AI, and Low-Code Technologies

Laxmiparbati Das[1,*], Achinta Kumar Palit[2], Samyo Ranjan Jagdev[3]

Abstract

Progressive web development (PWD) represents a modern approach to building web applications that are fast, responsive, secure, and capable of functioning offline across multiple devices and platforms. By integrating progressive web applications (PWAs) with emerging technologies such as Artificial Intelligence (AI)–driven design systems and low-code frameworks, developers can deliver high-quality digital experiences with greater efficiency and reduced complexity. AI-powered tools enhance the development process by automatically generating user interface components, styles, and layouts based on design tokens, behavioral patterns, and natural language inputs, significantly reducing the workload of designers and developers. At the same time, low-code architectures empower both professional developers and non-technical users to rapidly create applications through visual interfaces, reusable components, and drag-and-drop workflows. This approach minimizes manual coding, accelerates development cycles, and simplifies deployment and scalability. When combined with progressive web principles and industry best practices, low-code environments support rapid prototyping, iterative design, and continuous delivery. This paper explores the convergence of Progressive Web Development, Artificial Intelligence, and low-code frameworks, highlighting their key benefits, existing challenges, and practical strategies for adoption. The study demonstrates how these technologies collectively enable the creation of intelligent, adaptive, and cross-platform web applications suited to the demands of today's rapidly evolving digital landscape.

Keywords: AI-driven design systems, artificial intelligence (AI), low-code frameworks, progressive web applications (PWA), progressive web development (PWD), rapid application development

INTRODUCTION

The rapid evolution of web technologies has transformed the design, development, and delivery of digital applications to users. Modern users expect web applications to be fast, reliable, secure, and accessible across multiple devices and networks. Traditional web development approaches, which often rely on extensive manual coding and platform-specific solutions, struggle to meet growing demands within tight timelines and resource constraints. Consequently, progressive web development (PWD) has gained prominence by integrating web accessibility with highly native-like application performance, responsiveness, and robust offline capabilities. Progressive web applications (PWAs) leverage modern browser capabilities, such as service workers, offline caching, and responsive design, to deliver seamless, app-like experiences even in low or unreliable network environments [1].

Simultaneously, advancements in artificial intelligence (AI) are reshaping the software development lifecycle. AI-driven design systems

***Author for Correspondence**
Laxmiparbati Das
E-mail: laxmiparbatidas@gietbbsr.edu.in

[1-3]Assistant Professor., Department of Computer Science and Engineering, Gandhi Institute of Excellent Technocrats (GIET), Ghangapatna, Bhubaneswar, Odisha, India

Received Date: January 27, 2026
Accepted Date: January 30, 2026
Published Date: February 24, 2026

Citation: Laxmiparbati Das, Achinta Kumar Palit, Samyo Ranjan Jagdev. A Modern Approach to Web Application Development Using PWA, AI, and Low-Code Technologies. Journal of Software Engineering Tools & Technology Trends. 2026; 13(1): 18–22p.

can automatically generate user interfaces, layouts, and styles by interpreting design tokens, user behavior, and natural-language instructions. These intelligent tools reduce repetitive tasks, improve design consistency, and allow development teams to focus on higher-level problem-solving and innovation. Concurrently, sophisticated low-code frameworks are gaining widespread adoption by enabling rapid application development through visual modeling, reusable components, and drag-and-drop interfaces. By minimizing the need for extensive handwritten code, low-code platforms democratize application development and accelerate time-to-market [2].

The convergence of progressive web development, artificial intelligence, and low-code architecture represents a significant shift in modern web engineering. Together, these technologies offer a unified approach for building intelligent, scalable, and cross-platform applications with reduced complexity and development efforts. This integration not only enhances productivity and collaboration but also supports continuous deployment and adaptability in an increasingly dynamic digital landscape [3].

LITERATURE REVIEW

Recent studies have emphasized progressive web applications (PWAs) as a transformative development in web technologies, combining the flexibility of traditional web applications with the functionality of native-mode apps. Early studies emphasized that PWAs improve performance, reliability, and user engagement through features such as offline access, push notifications, and responsive design. Researchers have demonstrated that service workers and caching strategies play a crucial role in enabling a seamless user experience, particularly in regions with unstable network connectivity. Several comparative studies indicate that PWAs can achieve performance levels comparable to those of native applications, while maintaining lower development and maintenance costs [4].

In parallel, extensive research has explored the role of Artificial Intelligence in the development of modern web and user interfaces. AI-driven design systems and machine-learning models have been shown to automate UI generation, optimize layouts, and enhance personalization based on user behavior and contextual data. Prior studies have reported that the use of AI in design workflows improves consistency, reduces development time, and minimizes human error. Natural language–based design tools further simplify the collaboration between designers and developers by translating high-level requirements into functional components [5].

Recent studies have highlighted the growing interest in sophisticated low-code and no-code development platforms within the global academic and industrial research communities. Literature indicates that low-code frameworks significantly accelerate application development by reducing reliance on manual coding and enabling rapid prototyping. Studies have highlighted their effectiveness in empowering non-developers and facilitating agile development practices, although concerns remain regarding overall scalability, customization, and security [6, 7].

Recent scholarly works have increasingly focused on the convergence of PWAs, AI-driven development, and low-code frameworks. These studies suggest that their integration can create flexible, intelligent, and scalable ecosystems for modern web applications. However, the existing literature also identifies challenges related to interoperability, governance, and performance optimization, underscoring the need for further research and practical guidelines [8].

METHODOLOGY

This study employed a qualitative, exploratory research design to examine the convergence of Progressive Web Development, Artificial Intelligence, and low-code frameworks within contemporary web application development. The chosen methodology is intended to provide an in-depth understanding of how these technologies interact and complement one another in real-world development environments [9]. Rather than focusing on quantitative measurements, this approach emphasizes the analytical interpretation of current tools, frameworks, and development methodologies [10].

The research framework was designed to systematically investigate existing technological solutions, prevailing development practices, and documented implementation experiences. By analyzing these elements, this study aims to uncover the advantages associated with adopting progressive architectures, AI-driven features, and low-code platforms, as well as the challenges that organizations face during integration and deployment. Particular attention has been paid to issues such as scalability, performance optimization, development efficiency, and maintainability [11].

Additionally, the methodology supports the identification of effective strategies for integrating these technologies into cohesive development workflows. Through a critical evaluation of industry practices and case-based evidence, this research highlights practical approaches that enable developers and organizations to leverage the combined strengths of Progressive Web Applications, artificial intelligence capabilities, and low-code development environments. Ultimately, this qualitative exploration provides valuable insights that can guide future research and inform best practices for intelligent and scalable web application development [12].

The research methodology follows a structured, multi-phase approach to investigate the integration of Progressive Web Applications (PWAs), artificial intelligence–driven design systems, and low-code development platforms in modern web engineering. The first phase involved an in-depth examination of secondary sources drawn from a wide range of credible materials. These included peer-reviewed journal articles, conference proceedings, industry white papers, technical blogs, official platform documentation related to PWAs, AI-enabled design tools, and low-code development environments. This extensive review establishes a strong conceptual framework for this study and enables the identification of dominant trends, emerging technologies, and widely accepted best practices that shape contemporary web development [13].

In the second phase, a comparative analytical framework is applied to assess conventional web development techniques along with progressive, AI-supported, and low-code-based development approaches. This comparison was conducted by using a set of clearly defined evaluation criteria to ensure consistency and objectivity. Key performance indicators include development speed, application performance, scalability, quality of user experience, and long-term maintenance efforts. By examining these parameters, this study highlights the strengths and limitations of each development paradigm and illustrates how modern approaches address the inefficiencies and constraints of the traditional methods [14].

The third phase of the methodology focused on the analysis of selected real-world case studies. These cases involve organizations and development teams that have successfully adopted Progressive Web Applications in combination with artificial intelligence tools and low-code platforms. The case study analysis provided practical insights into how these technologies are implemented in real operational environments. It explores development workflows, architectural decisions, deployment strategies, and integration practices while also identifying common technical, organizational, and operational challenges encountered during implementation. This empirical perspective strengthens the study by linking theoretical concepts to practical outcomes [15].

In the final phase, insights derived from the literature review, comparative evaluation, and case study analysis are integrated and synthesized. This synthesis enables the formulation of actionable strategies and practical guidelines to effectively incorporate AI capabilities and low-code frameworks into Progressive Web Application development. The resulting recommendations aim to support developers, organizations, and decision-makers in designing web applications that are not only efficient and scalable but also intelligent and adaptable to evolving user needs.

Overall, this methodological approach provides a comprehensive and systematic understanding of how Progressive Web Applications, artificial intelligence–driven tools, and low-code platforms can be combined to enhance modern web development practices. By blending theoretical analysis with practical evidence, this study demonstrates how these technologies collectively contribute to the creation of high-performance, maintainable, user-centric web applications.

RESULTS ANALYSIS

The findings of this study indicate that the integration of progressive web development with Artificial Intelligence and low-code frameworks significantly enhances the efficiency and effectiveness of modern web application development. The analysis revealed that Progressive Web Applications (PWAs) consistently deliver improved performance, faster load times, and reliable offline functionality compared to traditional web applications. These features contribute to higher user engagement and improved accessibility across diverse devices and networks.

The results further demonstrate that AI-driven design systems substantially reduce the development time by automating the generation of user interface components, layouts, and styles. The use of design tokens and natural language inputs improves design consistency and minimizes repetitive manual tasks. Development teams have reported increased productivity and better collaboration between designers and developers when AI tools are incorporated into the workflow.

Low-code frameworks were found to accelerate application development and deployment by enabling rapid prototyping and visual modeling. The study showed that both developers and non-developers were able to contribute effectively to the development process, reducing dependency on specialized coding skills. Additionally, reusable components and prebuilt templates improve scalability and simplify maintenance.

Overall, the combined use of PWD, AI, and low-code platforms has resulted in faster development cycles, reduced complexity, and improved application quality. However, the results also highlight challenges related to customization, performance optimization, and integration with existing systems, emphasizing the need for careful planning and governance in real-world implementation.

CONCLUSION

This study concludes that the convergence of Progressive Web Development, Artificial Intelligence, and low-code frameworks represents a powerful and effective approach to the development of modern web applications. Progressive Web Applications provide a strong foundation for delivering fast, reliable, and offline-capable experiences that meet the expectations of current users across multiple platforms. When enhanced with AI-driven design systems, the development process becomes more intelligent and efficient as repetitive tasks such as interface generation, styling, and layout optimization are automated while maintaining design consistency and quality.

The integration of low-code frameworks further strengthens this approach by enabling rapid application development using visual tools, reusable components, and minimal manual coding. This not only accelerates development cycles and reduces costs but also broadens participation by allowing non-developers to contribute meaningfully to application creation. Consequently, organizations are better positioned to respond rapidly to evolving business requirements and dynamic market demands.

Despite these advantages, this study identifies the challenges related to customization, performance optimization, security, and integration with existing systems. Addressing these issues requires careful planning, adherence to best practices, and the continuous evaluation of tools and frameworks. Overall, the combined adoption of Progressive Web Development, AI, and low-code technologies offers a scalable, adaptive, and future-ready solution to build high-quality web applications in an increasingly dynamic digital environment.

REFERENCES

1. Love C. Progressive Web Application Development by Example: Develop Fast, Reliable, and Engaging User Experiences for the Web. Birmingham: Packt Publishing Ltd; 2018.
2. Wargo JM. Learning Progressive Web Apps. Boston: Addison-Wesley Professional; 2020.
3. Malavolta I, Ruberto S, Soru T, Terragni V. End users' perception of hybrid mobile apps in the Google Play Store. 2015 IEEE International Conference on Mobile Services, New York, NY, USA. 2015. p. 25–32. doi:10.1109/MobServ.2015.14.

4. Biørn-Hansen A, Majchrzak TA, Grønli TM. Progressive web apps: The possible web-native unifier for mobile development. In: Majchrzak TA, Traverso P, Krempels KH, Monfort V, editors. Proceedings of the 13th International Conference on Web Information Systems and Technologies (WEBIST); 2017 Apr 25–27; Porto, Portugal. Setúbal: SciTePress; 2017. p. 344–351. doi:10.5220/0006353703440351.
5. Myers B, Hudson SE, Pausch R. Past, present, and future of user interface software tools. ACM Trans Comput Hum Interact. 2000;7(1):3–28. doi:10.1145/344949.344959.
6. Costa A, Silva F, Moreira JJ. Towards an AI-driven user interface design for web applications. Procedia Comput Sci. 2024;237:179–186. doi:10.1016/j.procs.2024.05.094.
7. Amershi S, Weld D, Vorvoreanu M, Fourney A, Nushi B, Collisson P, et al. Guidelines for human-AI interaction. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19); 2019 May 4–9; Glasgow, Scotland, UK. New York: Association for Computing Machinery; 2019. p. 1–13. doi:10.1145/3290605.3300233.
8. Sahay A, Indamutsa A, Di Ruscio D, Pierantonio A. Supporting the understanding and comparison of low-code development platforms. 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), Portoroz, Slovenia. 2020. p. 171–178. doi:10.1109/SEAA51224.2020.00036.
9. Phalake VS, Joshi SD. Low code development platform for digital transformation. In: Kaiser MS, Xie J, Rathore VS, editors. Information and Communication Technology for Competitive Strategies (ICTCS 2020): Intelligent strategies for ICT. Singapore: Springer Nature; 2021. p. 689–697. doi:10.1007/978-981-16-0882-7_61.
10. Prinz N, Rentrop C, Huber M. Low-code development platforms—A literature review. In: AMCIS 2021: Digital Innovation and Entrepreneurship; 2021 Aug 9–13; Virtual Conference. Atlanta (GA): Association for Information Systems; 2021. Article 1144. p. 1–10.
11. De Silva DI, Shangavie R, Ranathunga RAAL. Role of quality assurance in low-code/no-code projects. 2024 International Conference on Information Networking (ICOIN), Ho Chi Minh City, Vietnam. 2024. p. 789–794. doi:10.1109/ICOIN59985.2024.10572203.
12. Boehm B. A view of 20th and 21st century software engineering. In: Proceedings of the 28th International Conference on Software Engineering (ICSE '06); 2006 May 20–28; Shanghai, China. New York: Association for Computing Machinery; 2006. p. 12–29. doi:10.1145/1134285.1134288.
13. Fowler M. Domain-Specific Languages. Boston: Addison-Wesley; 2010.
14. Selva-Mora A, Quesada-López C. Security practices in agile software development: A mapping study. In: Proceedings of the 7th ACM/IEEE International Workshop on Software-Intensive Business (IWSiB '24); 2024 Apr 14–15; Lisbon, Portugal. New York: Association for Computing Machinery; 2024. p. 56–63. doi:10.1145/3643690.3648241.
15. Pressman RS. Software Engineering: A Practitioner's Approach. 7th ed. New York: McGraw-Hill; 2010.

