



Layr: No Code Solution For Building Websites

¹Dr. Reshma J, ²Sarvochcha Sharma, ³Shambhav Kumar, ⁴Sunny Kumar, ⁵Vishesh Kumar

¹ Associate Professor, Department of Information Science & Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka, India,

^{2,3,4,5} Student, Department of Information Science & Engineering, Dayananda Sagar College of Engineering, Bangalore, Karnataka, India

Abstract: Web-Based No-Code Website Builder (NCWB) platforms aim to democratize web development by enabling users with minimal or no programming experience to design and deploy responsive websites. Leveraging intuitive drag-and-drop interfaces, these platforms integrate pre-built modular components—such as text fields, images, navigation bars, and multimedia elements—to streamline the design process and eliminate the need for manual coding. The review paper emphasizes accessibility, usability, and flexibility, providing real-time visual customization and responsive design capabilities. By adopting a no-code approach, the framework significantly reduces development time and cost while enhancing creative freedom for individuals, startups, and small enterprises. The study highlights the potential of no-code platforms in fostering digital inclusivity, empowering non-technical users to establish an effective online presence, and promoting innovation in modern web development practices.

Index Terms - No-Code Development, Drag-and-Drop Interface, Website Builder, Web Design Automation.

I. INTRODUCTION

Traditional website development requires proficiency in programming languages such as HTML, CSS, and JavaScript, which can be a significant barrier for individuals without technical expertise. Designing, coding, and deploying a website manually involves a series of complex steps including layout structuring, style customization, responsiveness optimization, and debugging — all of which demand substantial time, effort, and skill. For many small businesses, startups, educational institutions, and independent creators, hiring professional web developers or agencies to handle these tasks can be both time-consuming and financially burdensome. Consequently, the lack of accessible web development solutions limits opportunities for non-technical users to establish an effective online presence and showcase their ideas, products, or services in the digital landscape.

To address these challenges, **no-code website development platforms** have emerged as a transformative solution. These platforms enable users to design, build, and launch websites without writing a single line of code by utilizing intuitive, visual drag-and-drop interfaces. Through an interactive and user-friendly workspace, users can assemble websites using pre-built components such as text boxes, images, buttons, forms, navigation bars, and multimedia elements. The underlying system automatically generates and manages the necessary code in HTML, CSS, and JavaScript, thereby simplifying the entire web development process.

Modern no-code platforms also integrate advanced features such as **real-time preview, responsive layout optimization, and cloud-based hosting**, ensuring that the created websites adapt seamlessly across various devices including smartphones, tablets, and desktop computers. Furthermore, many of these platforms incorporate libraries of customizable templates, theming options, and content management capabilities to support creative flexibility and brand consistency.

Beyond usability, no-code platforms contribute to **digital inclusivity and innovation** by empowering a diverse range of users—students, entrepreneurs, educators, and small enterprises—to engage directly in web creation. By lowering the entry barrier to web design, these systems promote experimentation, rapid prototyping, and independent digital expression. Ultimately, no-code website builders redefine traditional development workflows by prioritizing accessibility, efficiency, and creativity, making professional-quality web development achievable for everyone regardless of their technical background.

In recent years, the **no-code development paradigm** has gained significant attention as part of a broader movement toward low-code and rapid application development platforms. These technologies are redefining the software creation process by shifting the focus from syntax-heavy programming to **visual modeling and configuration-based design**. In the context of web development, this means users can visually construct website structures, define layouts, and configure functionality using graphical tools, while the platform automatically manages the technical implementation. This approach not only accelerates the development lifecycle but also bridges the gap between technical and non-technical users, enabling greater participation in digital innovation.

Moreover, **no-code web builders** integrate essential web technologies in the background — including front-end frameworks, responsive grid systems, and content management modules — ensuring that the generated websites maintain professional standards in performance and scalability. Many platforms employ **component-based architectures**, where reusable design blocks are dynamically combined to build complex pages. This modularity supports rapid updates and customization without affecting other site components, making maintenance simpler and more efficient.

From a design perspective, no-code platforms emphasize **user experience (UX)** and **user interface (UI)** consistency through pre-built templates, responsive design principles, and accessibility compliance (such as WCAG standards). These features ensure that websites are not only visually appealing but also inclusive and functional for diverse audiences. The automation of repetitive and technical tasks also allows users to concentrate on creativity, branding, and content strategy rather than low-level code optimization.

From a broader perspective, the rise of no-code platforms aligns with the global movement toward **digital empowerment and technological inclusivity**. By reducing the dependency on specialized programming skills, these platforms democratize web creation and support entrepreneurship, education, and digital transformation across industries. Small businesses can establish an online identity quickly; educators and students can create learning portals or portfolios; and non-profit organizations can share their missions with the world—all without needing professional developers.

Furthermore, as the demand for **rapid digital presence and content personalization** continues to grow, no-code tools are expected to play a pivotal role in the future of web design and development. Continuous innovation in this field, including the integration of **AI-assisted design generation, real-time analytics, and modular plugin ecosystems**, promises to make website creation even more accessible and intelligent. Ultimately, no-code website builders are not merely simplifying development; they are transforming the very definition of who can create and innovate in the digital era.

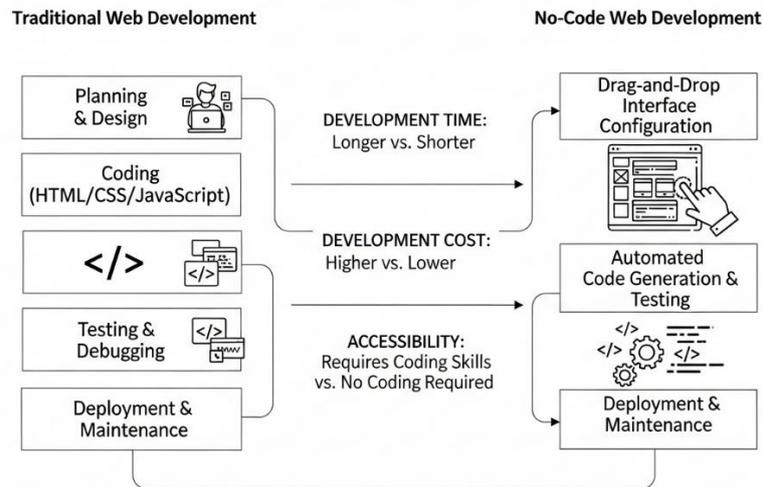


Fig.1 Traditional vs No Code web development

This approach reduces development cost and time, makes web creation accessible to everyone, and encourages creativity and independence. It also helps small businesses and startups to quickly establish their digital presence without depending on developers. With the growing need for fast, user-friendly website creation methods, no-code builders are becoming an important part of modern web development.

II. LITERATURE REVIEW

In this paper, we look at various studies and reviews that focus on no-code and low-code web development, drag-and-drop website builders, and automated website generation systems. These studies collectively highlight how such technologies are transforming web development by removing coding barriers and enabling non-technical users to create professional websites with ease. The following literature provides an overview of existing research, frameworks, and comparative analyses on this evolving field.

[1] Kaluarachchi and Wickramasinghe. A Systematic Literature Review on Automatic Website Generation.

A systematic literature review on automatic website generation by Kaluarachchi and Wickramasinghe. This systematic literature review by Kaluarachchi and Wickramasinghe provides a thorough overview of the push toward automatic website generation, motivated by the desire to use technology like machine learning to cut down on the time, effort, and cost of building websites. The researchers analyzed 47 relevant studies, organizing the current methods into three main categories. First, mock-up-driven generation focuses on turning a picture of a design, like a hand-drawn sketch or digital mock-up, directly into working code, aiming to automate the manual coding step entirely. Second, example-based generation creates new, high-quality websites by learning and adapting the best design elements from existing, professional websites, offering a solution for users who want a custom look without writing code. Finally, the most advanced approach is AI-driven generation, where an intelligent system asks a user for their preferences and then automatically designs and builds the entire customizable site based on those inputs.

By detailing the unique strengths and weaknesses of each strategy, the paper serves as a valuable guide for researchers, pointing out the major challenges and suggesting promising avenues for future work in this rapidly evolving field. For instance, while mock-up-driven approaches are excellent for quickly translating a visual concept into a functional front-end, they often struggle with creating complex back-end logic or ensuring the generated code is clean and maintainable. Similarly, example-based generation can produce aesthetically pleasing sites but may be limited by the quality and variety of the initial examples it learns from. The AI-driven approach, while the most ambitious, faces the significant hurdle of accurately interpreting user intent and balancing creative design with functional requirements.

[2] Nikhil. Low-Code/No-Code Development Platforms: Empowering the Next Generation of Software Creators.

In this research paper, Nikhil explores the rise of low-code/no-code (LCNC) development platforms as a direct response to the growing gap between the high demand for software and the limited supply of expert developers. The author highlights a key industry projection from Gartner, which predicts that over 70% of new enterprise applications will be built using LCNC technologies by 2025. The paper makes a clear distinction between low-code platforms (like Mendix and OutSystems), which are designed for developers to minimize manual coding while retaining customization power, and no-code platforms (like Appgyver and Bubble), which are aimed at business users or "citizen developers" with little to no programming experience.

Nikhil details the core features driving this trend, such as visual drag-and-drop interfaces, reusable components, and automated workflows. The benefits are quantified with findings from a McKinsey report, noting up to a 60% reduction in development time and a 30-40% reduction in costs. However, the paper maintains a balanced view by thoroughly discussing the significant challenges, including security concerns with data access, scalability issues for enterprise-level workloads, the risk of vendor lock-in, and inherent limitations in deep customization. This is particularly relevant for mission-critical systems where highly customized user interfaces or complex business logic may not be feasible without resorting to custom code.

[3] Shreyansh Jagtap et al. A No-Code Website Builder Approach.

This paper by Shreyansh Jagtap and his colleagues directly addresses the technical complexity that has traditionally made website creation inaccessible for non-programmers. Their work proposes a "No Code Website Builder" centered on a highly intuitive, user-friendly design philosophy. The fundamental innovation is a methodology built around pre-configured, modular HTML components, or "blocks," which users can simply drag and drop into a workspace to construct their webpage. The system is architected with a clear separation between the user's live workspace and the editor, which contains modules for managing blocks, styles, and layers. This allows users to manipulate and style elements in real-time without writing a single line of code.

The architectural design of separating the 'Workspace' and the 'Editor' into two distinct HTML documents is a key technical decision highlighted in the paper. This modular approach not only adheres to the principle of separation of concerns—allowing the editor to focus on user-friendly content creation while the workspace handles layout—but also improves performance by reducing rendering overhead. Furthermore, this separation streamlines the process of exporting the final website, as the clean HTML from the workspace can be easily extracted without being entangled with the editor's interface code.

[4] K. Akil Prasath et al. Drag and Drop Visual Website Builder.

This paper by K. Akil Prasath and his team positions the drag-and-drop visual website builder as a vital and transformative tool in modern web development. The authors argue that in an era where an online presence is essential, traditional coding-intensive methods are a significant barrier. Their work focuses on a proposed system that entirely removes the need for manual HTML or CSS editing, allowing users to build attractive, high-quality websites through a simple, step-by-step visual process. The paper outlines the core functionality: users add pre-made blocks, edit images, links, and styles, and can even add custom blocks to extend the site's capabilities.

A key advantage highlighted is the efficiency it offers, saving both time and money. The authors differentiate between online proprietary tools and offline software, noting that their proposed "what you see is what you get" (WYSIWYG) approach is designed to be so straightforward that creating a landing page can be done in just a few clicks. The proposed system is not just for novices; it also includes features designed for more advanced use cases, such as the ability for users to export their sites (including all assets like JavaScript and CSS files), publish directly to a live server via FTP, and even create and manage user accounts for clients to access their own built sites.

[5] Karlis Rokis and Marite Kirikova. Exploring Low-Code Development: A Comprehensive Literature Review.

In this comprehensive literature review, Karlis Rokis and Marite Kirikova synthesize the existing body of knowledge on low-code development, presenting it as a tool-based approach that significantly accelerates software creation. The paper traces the term "low-code" back to a 2014 Forrester Research paper and

amalgamates various definitions into a cohesive one, describing it as an approach that uses visual programming and abstraction to minimize hand-coding. The review is meticulously structured, analyzing 42 unique articles to answer specific research questions about low-code platforms, development lifecycles, and application areas.

The authors identify and categorize a vast list of features supported by Low-Code Development Platforms (LCDPs), including visual designers for UI, drag-and-drop functionality, predefined components and templates for reusability, and graphical workflow editors for business logic specification. They underscore that low-code aligns well with Agile methodologies by promoting iterative delivery and continuous stakeholder feedback. The review also thoroughly summarizes the key benefits—such as accelerated development, decreased costs, and the empowerment of citizen developers—alongside the persistent challenges, which include extensibility limitations, vendor lock-in, and difficulties in testing non-functional requirements.

[6] Prithviraj Sutar et al. No Code Website Builder.

This paper by Prithviraj Sutar and his team details the creation of a no-code, drag-and-drop webpage builder developed using the React.js framework. The project's primary goal is to democratize website creation by providing a purely visual and intuitive tool for users who lack programming skills. The application features a real-time, dynamic interface where users can drag, drop, and customize various UI components like text boxes, images, lists, and videos. A key technical aspect is the system's ability to generate clean, structured HTML and inline CSS code from the visual design, which can then be easily exported for hosting.

The methodology is described as user-centric and modular, beginning with a secure, cookie-based login system. Once authenticated, users are directed to a dashboard where they can start a new project from scratch, use a pre-designed template for things like resumes or blogs, or open an existing project. Projects are saved in a proprietary .wbproj format, which captures all element properties and layout configurations. The core of the application is the editor's workspace, which dynamically resizes to accommodate large projects and includes a dedicated properties section for granular control over attributes like font, color, and alignment for each element.

[7] Arti Ochani et al. AI-Powered Drag-and-Drop Website Builder.

This paper introduces a next-generation website builder that integrates artificial intelligence directly into the design process to further simplify web development for non-technical users. The authors, Arti Ochani and her team, present a platform built on the VvwebJs framework that features a standard drag-and-drop interface but enhances it with powerful AI tools. Specifically, it incorporates Gemini/ChatGPT for AI text generation, allowing users to automatically populate their site with coherent, contextually relevant content. It also uses Stable Diffusion 1.5 for AI image generation, enabling the creation of custom visuals from simple descriptive prompts.

The paper includes a comparative analysis against industry leaders like Wix, WordPress, and Webflow, arguing that its seamless integration of both text and image AI sets it apart. The analysis, presented in both a table and a feature comparison graph, highlights key advantages. For instance, unlike Wix, their builder allows for a full project export to a ZIP file, offering greater flexibility for external hosting. And unlike WordPress, the AI tools are core integrations, not third-party plugins. The platform also includes a built-in code editor, providing a level of customization that is often limited in other user-friendly builders.

[8] MD. SIAM et al. A Review on Web Design & Development.

In this review, MD. SIAM and his colleagues synthesize foundational research on web design and development to establish a core set of principles for creating effective and engaging user experiences. The paper argues that with millions of users online, a well-designed website is crucial for engaging users and conveying information effectively. After reviewing the literature, the authors identify and consolidate seven frequently mentioned elements of effective design: navigation, graphical representation, organization, content utility, purpose, simplicity, and readability.

The paper also provides a primer on the underlying technologies, clearly distinguishing between front-end technologies like HTML, CSS, and JavaScript, which control what the user sees, and back-end technologies, which manage the server, applications, and databases. The authors use the analogy of web development being like "house building," where a structured process is essential. Just as one would consult an architect and get building permits before construction, web development requires careful planning, design, and documentation. This metaphor effectively communicates the idea that a successful website is not just an aesthetic creation but a well-engineered product.

[9] Mrs. Mamatha S. K. et al. No-Code Web Development.

This research paper by Mrs. Mamatha SK and her team investigates the transformative impact of no-code web development, focusing on its ability to empower non-programmers. The authors position no-code platforms as a democratizing force in the digital landscape, enabling individuals and small businesses to establish an online presence without the traditional barriers of technical skill or high cost. Using a mixed-methods approach that combines qualitative interviews with developers and user satisfaction surveys, the study explores both the benefits and the challenges of this movement.

The paper explicitly frames its work as a solution to a critical problem: the fact that a significant number of small businesses still lack an online presence. The authors argue that for entrepreneurs without technical skills, the task of creating a website from scratch is often "overwhelming and discouraging." No-code builders bridge this gap by offering intuitive, drag-and-drop interfaces and pre-designed templates, thereby leveling the playing field. The study's system analysis confirms that these solutions are technically, economically, and operationally feasible, as they require minimal technical resources and have low implementation costs.

[10] Fahim Sufi. Algorithms in Low-Code-No-Code for Research Applications.

This practical review by Fahim Sufi explores the growing adoption of low-code-no-code (LCNC) platforms as powerful tools for scientific and academic research. The author frames the evolution of programming from complex machine code to intuitive visual interfaces, arguing that LCNC is the next logical step in democratizing access to powerful computational methods. The paper systematically reviews 47 relevant studies to identify how researchers are leveraging these platforms, which tools are most popular, and what specific features are driving their adoption. The findings show that Microsoft Power Platform is the dominant choice, used in 61% of the analyzed research projects, which span diverse fields such as social media analysis, disaster management, and logistics.

The paper meticulously details the benefits and limitations of LCNC specifically from a researcher's perspective. Key benefits include the ability to align research goals directly with IT solutions without intermediary developers, address resource scarcity in academic settings, and rapidly prototype and test ideas without large investments. The most critical advantage highlighted is the seamless integration of AI, ML, and Deep Learning algorithms. Researchers can now perform complex tasks like sentiment analysis, named entity recognition (NER), and even apply CNN-based models using simple drag-and-drop interfaces. However, the author also presents significant limitations for the research community, such as vendor lock-in, a lack of on-premises support for sensitive data, unsuitability for mission-critical real-time systems, and ongoing subscription costs that can be prohibitive for academic institutions.

[11] Prajwal Nimje. The Rise of Low-Code/No-Code Development Platforms.

In this paper, Prajwal Nimje explores the rapid emergence of low-code/no-code (LCNC) platforms as a transformative force in the software development landscape. The author frames LCNC as a powerful alternative to traditional development, which has long been characterized by lengthy coding cycles and a reliance on highly skilled programmers. The central argument is that LCNC is democratizing application creation by empowering users with minimal coding experience to build functional software through intuitive, drag-and-drop interfaces and pre-built components. The paper sets out to analyze the key factors driving this trend, including the promise of faster development, greater business agility, and the ability to innovate more rapidly.

The analysis provides a balanced overview of both the significant benefits and the potential challenges associated with LCNC adoption. Drawing from a literature review, the paper emphasizes the democratization of app creation as a primary driver, empowering "citizen developers" to build their own solutions and thereby alleviating the pressure on overburdened IT departments. This aligns with findings that point to the scarcity of skilled developers as a key motivator for companies to adopt these platforms. However, the author also gives weight to the challenges, highlighting critical concerns such as security vulnerabilities arising from pre-built components, the need for robust data governance to maintain data integrity, and the inherent limitations in complexity, which may render LCNC platforms unsuitable for highly complex or mission-critical systems.

Author	LAYR: No Code Solution For Building Websites		
	Year of publication	Focus / Methodology	Key Findings / Results
Kaluarachchi & Wickramasinghe	2023	Systematic Literature Review on Automatic Website Generation	Categorized 47 studies into mock-up-driven, example-based, and AI-driven approaches; highlighted efficiency and adaptability challenges.
Nikhil	2022	Study on Low-Code/No-Code (LCNC) Platforms	Defined "citizen developers" found that LCNC reduces cost and time but introduces vendor lock-in and security risks.
Shreyansh Jagtap et al.	2022	No-Code Website Builder Framework	Real-time drag-and-drop modular interface; increased accessibility and flexibility for non-programmers.
K. Akil Prasath et al.	2021	Drag-and-Drop Visual Website Builder	Compared online vs offline builders; confirmed reduced dependency on skilled coders and shorter build times.
Karlis Rokis & Marite Kirikova	2021	Review on Low-Code Development Platforms	Highlighted visual modeling and workflow automation; noted limitations in customization and maintainability.
Prithviraj Sutar et al.	2022	React.js-based No-Code Website Builder	Generated clean HTML/CSS automatically; supported rapid prototyping and flexible component editing.
Arti Ochani et al.	2023	AI-Powered Drag-and-Drop Website Builder	Integrated Gemini/ChatGPT for content and Stable Diffusion for images; achieved faster and smarter web creation.
MD. SIAM et al.	2020	Review on Web Design & Development	Analyzed key UI/UX principles; distinguished front-end and back-end roles for optimal performance.
Mrs. Mamatha S. K. et al.	2021	Study on No-Code Web Development	Survey-based findings showed empowerment of non-technical creators and rise of small-scale entrepreneurship.

Fahim Sufi	2023	LCNC for Research Applications	Concluded that LCNC platforms are essential tools for non-programming researchers, enabling them to apply complex AI/ML algorithms without manual coding.
Prajwal Nimje	2024	Overview of the Rise of LCNC Platforms	Concluded that LCNC democratizes app creation and improves agility, but faces challenges in security, governance, and suitability for complex systems.

TABLE I: Literature Review on No-Code and Low-Code Website Development Approaches

III. CONCLUSION

No-code website development platforms have emerged as a transformative solution to the limitations of traditional web development, enabling users without programming expertise to design and deploy responsive, functional, and visually appealing websites. This survey highlights how these platforms utilize drag-and-drop interfaces, pre-built components, and automated code generation to reduce complexity, cost, and development time. The integration of advanced technologies such as cloud computing, AI-driven optimization, and modular architecture further enhances usability, scalability, and accessibility.

The study also emphasizes the growing significance of no-code solutions in promoting digital inclusivity and democratizing website creation, thereby empowering entrepreneurs, educators, and small organizations to establish their online presence efficiently. Despite their rapid advancement, challenges remain in areas such as customization limits, performance optimization, and integration flexibility. Addressing these concerns through improved architecture and AI-enhanced adaptability will be crucial for future developments.

Overall, no-code website builders represent a major step toward simplifying web development workflows while maintaining professional standards of design and functionality. As research and innovation in this domain continue to evolve, these platforms are expected to play a vital role in shaping the future of accessible, intelligent, and user-centric web design.

IV. REFERENCES

- [1] Kaluarachchi, S. & Wickramasinghe, W. (2020). A systematic literature review on automatic website generation. Department of Computing, Sri Lanka Institute of Information Technology, Sri Lanka. Proceedings of the 2nd International Conference on Advanced Research in Computing (ICARC), IEEE.
- [2] Nikhil (2021). Low-Code/No-Code Development Platforms: Empowering the Next Generation of Software Creators. International Journal of Research in Engineering and Science (IJRES), Volume 13, Issue 4, pp. 229–232.
- [3] Shreyansh Jagtap, et al. (2023). A No Code Website Builder Approach. International Journal of Scientific Research in Engineering and Management (IJSREM), Volume 7, Issue 3.
- [4] K. Akil Prasath, et al. (2017). Drag and Drop Visual Website Builder. International Journal of Advanced Research in Basic Engineering Sciences and Technology (IJARBEST), Volume 3, Issue 34, pp. 13–17.
- [5] Rokis, K. & Kirikova, M. (2023). Exploring Low-Code Development: A Comprehensive Literature Review. Complex Systems Informatics and Modeling Quarterly (CSIMQ), Issue 36, pp. 55–74.

[6] Prithviraj Sutar, et al. (2025). No Code Website Builder. International Research Journal of Modernization in Engineering Technology and Science (IRJMETS), Volume 7, Issue 3.

[7] Arti Ochani, et al. (2025). AI-Powered Drag-and-Drop Website Builder. International Research Journal of Modernization in Engineering Technology and Science (IRJMETS), Volume 7, Issue 4.

[8] MD. SIAM, et al. (2022). A Review on Web Design & Development. International Journal of Scientific Development and Research (IJS DR), Volume 7, Issue 12.

[9] Mrs. Mamatha S.K., et al. (2023). No-Code Web Development. International Research Journal of Modernization in Engineering Technology and Science (IRJMETS), Volume 6, Issue 6.

[10] SUFI, F. (2023). Algorithms in Low-Code-No-Code for Research Applications: A Practical Review. *Algorithms*, Volume 16, Issue 2.

[11] NIMJE, P. (2024). The Rise of Low-Code/No-Code Development Platforms. *International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)*, Volume 4, Issue 3”

