Faculty of Natural and Applied Sciences Journal of Computing and Applications
Print ISSN: 3026-8133
www.fnasjournals.com
Volume 2; Issue 4; December 2025; Page No. 43-51.
DOI: https://doi.org/10.63561/jca.v2i4.1073

Development of a User-Centered Prototype Model for Low-Code Development Platforms

***[1]Okpako, E.A., [2]Ojie, D. V., [2]Iyamah, B., [2]Ukadike, D., [3]Chiemeke, S.C., & [3]Chete, F.O.**
[1]Department of Cyber Security, Faculty of Computing, University of Delta, Agbor, Delta State,
[2]Department of Software Engineering, Faculty of Computing, University of Delta, Agbor, Delta State,
[3]Department of Computer Science, University of Benin, Edo State, Nigeria

[*]**Corresponding author email**: ejaita.okpako@unidel.edu.ng,

Abstract
This research work investigates the user experience (UX) of selected Low-Code Development Platforms (LCDPs), specifically Microsoft Power Apps, OutSystems, and Mendix. The study addresses the growing demand for user-friendly development tools among both technical and non-technical users, emphasizing the need for platforms that balance simplicity, functionality, and performance. A hands-on testing approach was employed, involving real users completing pre-defined tasks on the selected LCDPs. Data was collected through direct observation, task timing, and structured surveys measuring usability parameters such as ease of navigation, learnability, responsiveness, and interface satisfaction. The feedback obtained revealed several user challenges, including unintuitive interfaces, lack of contextual help and complexity in task execution for novice users. In response, a prototype named FlowFrame was developed using Figma, incorporating improvements based on users' feedbacks. The prototype featured an intuitive dashboard, drag-and-drop functionality, AI-powered suggestions, and smart onboarding flows aimed at enhancing learnability and task efficiency. Findings show that FlowFrame significantly improves the perceived usability and user satisfaction compared to existing platforms. The design prioritizes clarity, guidance, and simplicity, especially for non-technical users. The study demonstrates that user-centered design and simulation can effectively address the usability gaps in current LCDPs. The study further recommends the adoption of intelligent design principles and the integration of AI assistance to enhance user experience in future platform developments.

Keywords: Low code, Development Platforms, User interface, Citizen Developers, Task performance.

Introduction

Low-Code Development Platforms (LCDPs) are tools that allow users to build applications with minimal coding using visual interfaces and pre-built components. They enable non-programmers to create functional applications, reducing the need for professional developers (Mendling et al., 2021). These platforms speed up development, lower costs, and support digital transformation in areas like finance, healthcare, and logistics. The major reason for the adoption of these platforms is their usability which is driven by the demand for quick and scalable software solutions. However, usability varies across platforms and includes aspects like ease of use, interface design, learning resources, integration, and customization (Del Buono., 2022; Palomares et al., 2020).

Some platforms are beginner-friendly with drag-and-drop features, while others need more technical knowledge, security, performance, and collaboration features. Enterprise-grade use may face challenges with performance or managing complex workflows. Support for teamwork, testing, and deployment helps make the platform usable across teams while training, documentation, and user community support influence how quickly users adapt to these tools. Westner et.al. (2023). This study aims to evaluate and compare the usability of Microsoft Power Applications (apps)**,** **OutSystems,** and **Mendix**, identifying their strengths, weaknesses, and best-fit users. Although LCDPs bring many

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073

benefits, their true value depends on how usable they are. Understanding user experiences will, however, guide future improvements in low-code development. Guthardt et.al. (2024).
This research is grounded in the Technology Acceptance Model (TAM) and Usability Engineering principles, which together offer behavioral and technical perspectives on the adoption and performance of Low-Code Development Platforms (LCDPs). TAM emphasizes perceived ease of use and usefulness as key factors in technology adoption, particularly relevant to LCDPs due to their user-friendly design and productivity benefits (Venkatesh & Bala, 2020). Usability Engineering, on the other hand, focuses on measurable aspects such as efficiency, learnability, and error handling (Ardito et al., 2021). Combining both models enables a comprehensive analysis of user perceptions and platform performance. Related studies reinforce the importance of usability, highlighting issues like complex interfaces, poor documentation, and security limitations, while also noting the benefits of rapid prototyping and collaboration support. Akwukwuma et al. (2024) presented the usability analysis of E-commerce mobile application and underscored the need for user-centric design while in a separate research work; Akwukwuma et al. (2024) also underscored the importance of design innovation and data- driven decision-making in optimizing user interface for enhanced user experiences and satisfaction. LCDPs, however, face challenges that affect user satisfaction and adoption, hence the need for in-depth usability evaluation to guide improvements and enhance their real-world effectiveness. (Kass et.al. 2022; Kass et.al. 2023)

Aim of the study

The aim of this study is to evaluate the usability of selected low-code development platforms and to design and assess a user-centered prototype (FlowFrame) that addresses identified usability gaps.

Research objectives

1. Evaluate user experience factors—ease of navigation, learnability, responsiveness, and satisfaction—relevant to improving low-code development platforms.
2. Develop a user-centered prototype (FlowFrame) incorporating usability improvements identified from literature and user testing.
3. Conduct formative usability testing of FlowFrame using task-based evaluation measuring ease of use, user engagement, and user satisfaction.
4. Use evaluation findings to recommend design improvements and outline directions for future large-scale comparative studies of LCDPs.

Methods and Materials

This section describes the System Design, Population and sampling, techniques used for data collection and analysis.

System Design

Icons and interactions were incorporated using Font Awesome version 6 to represent key features (e.g., Builder, Preview, Settings, Publish, Log Out, AI Suggestions). The research also made use of basic interaction linking in Figma to simulate navigation between pages.
The development followed a Design Science Research Methodology (DSRM) approach given as follows:

1. **Problem Identification:** Existing LCDPs have usability limitations.
2. **Objective Definition:** Build an intuitive and smart LCDP prototype.
3. **Design and Development:** Prototype developed using Figma.
4. **Demonstration:** Demonstrated through interactive wireframes.
5. **Evaluation:** Evaluated based on improved user flow, AI suggestions, and UI enhancements.
6. **Communication:** Documented through reports, diagrams, and presentations.

Study Area

The target population for this study is the entire 400-level cohort in the Department of Computer Science, University of Delta, Agbor. The 400-level cohort comprises eighty-five students.

Population and Data Collection Technique

To determine the required sample size for a known population the Krejcie and Morgan (1970) formula was applied. The parameters used were $X^2 = 3.841$ (chi-square for 1 degree of freedom at 95 percent confidence), $P = 0.5$, $d = 0.05$. Substituting these values yields a required sample size of approximately seventy participants. In practice thirty-five participants volunteered and completed the usability tasks. This shortfall is acknowledged as a limitation. The study therefore reports formative usability findings intended to identify major usability issues and guide prototype iteration

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073

rather than to support broad statistical generalization to the entire 400-level cohort. For clarity and reproducibility the manuscript now shows both the calculated required sample (≈70) and the actual recruited sample (n = 35) and explicitly treats results as formative.

Prototype Development

The prototype was designed using Figma as the core tool for building interactive wireframes and UI/UX mockups. The system was broken down into multiple screens/pages as follows:

- **Login/Welcome Screen:** A simple interface for user authentication, with an option to try a demo.
- **Dashboard/Home Page:** Featuring navigation panel, quick-start cards ("Start New App", "Continue"), and links to Help or AI suggestions.
- **Smart On-boarding Flow:** A walkthrough experience for new users with pop-up tooltips and contextual guidance (marked as "Coming Soon").
- **Drag-and-Drop Builder:** The main workspace where users build applications using components that can be dragged, a canvas, and configurable settings.
- **Live Preview Page:** Simulates real-time view of the app, with toggle support for various device types.
- **Smart Suggestions Overlay:** A semi-functional AI system providing pop-ups for automation tips and design enhancement
- **Help & Support:** Contains static FAQs, AI chatbot mockup, and guides.
- **Publish Page:** A simulation of app export or build generation.

Figure 1 shows the UML Use Case Diagram of the Flowframe Prototype LCDP (This UML use case diagram illustrates how the user interacts with core features of the Flowframe prototype while Figure 2 depicts the prototype screen flow showing key interface sections

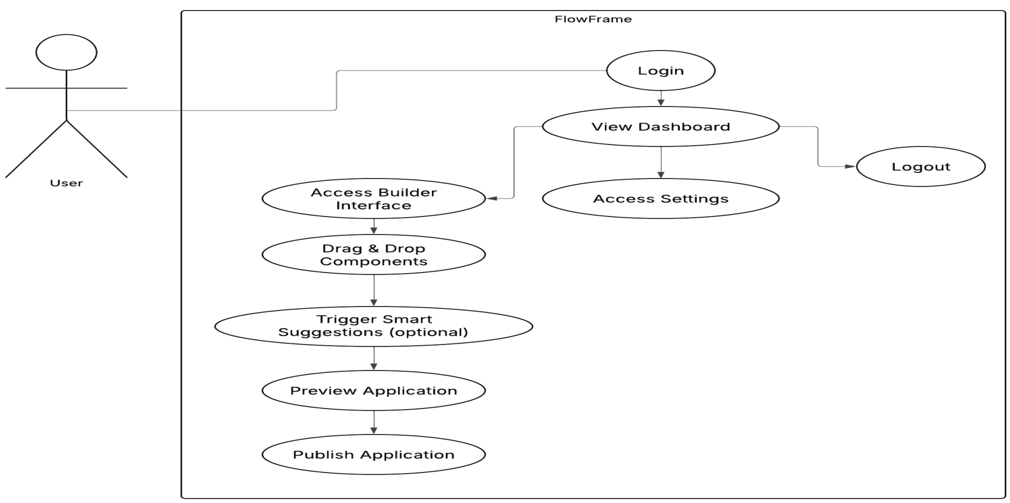

Figure 1: UML Use Case Diagram of the Flowframe Prototype LCDP (This UML use case diagram illustrates how the user interacts with core features of the Flowframe prototype)

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073

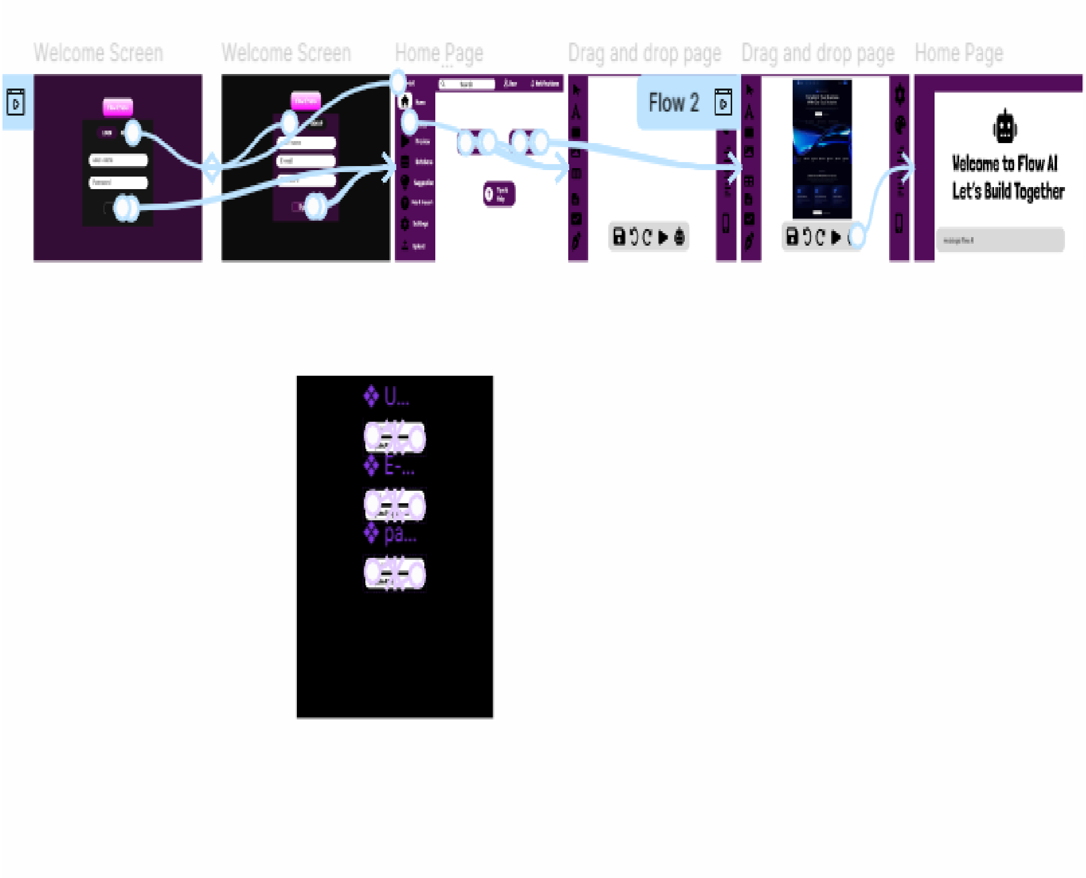

Figure 2: This illustrates the Prototype screen flow showing key interface sections

Instruments and measurement

Ease of Use was measured with a five-item Likert scale adapted from standard usability questionnaires. Each item was rated from 1 to 5 where 1 equals Strongly disagree and 5 equals Strongly agree. Example items are: The FlowFrame interface was easy to navigate; I could complete assigned tasks without help; Menus and controls were clearly labelled; I learned to use the interface quickly; The system responded promptly to my actions. Individual item scores were averaged to produce an Ease of Use score for each participant. Henriques et.al. (2021). Reported reliability should include Cronbach's alpha for this scale; calculate and report the alpha value in the Results section and discuss if the internal consistency is below conventional thresholds.

User Engagement was measured using objective task metrics and a brief engagement questionnaire. Task metrics were defined as follows. Task completion rate equals successful completions divided by attempts for each task. Average completion time is the mean time taken to complete each task measured in seconds. Error rate equals the count of recoverable errors recorded per task. Moderator assistance equals the number of times the moderator intervened. The engagement questionnaire contained four items rated 1 to 5, for example: I felt engaged while performing the tasks; I wanted to continue using the interface after the tasks; The tasks felt motivating; I felt distracted while using the interface (reverse-scored). The mean of the engagement questionnaire items produces the participant engagement score. Report each task metric separately and then summarize combined engagement conclusions. Observation notes were recorded during every session using a standardized checklist that captured verbal comments, visible confusion, and task-specific difficulties. Two researchers independently reviewed the observation notes and resolved differences through discussion to improve consistency. Use qualitative observations to explain anomalies in quantitative metrics and to illustrate specific usability problems.

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073

Task definitions and completion criteria

Define and report explicit success criteria for each task. For example, Task 1 success equals creating a new app and opening the builder without moderator assistance. Task 2 success equals adding a form component and saving it. A task is counted as incomplete if the participant fails to reach the defined end state without moderator assistance within the allotted time. State the time limits used for each task if time limits were applied.

Performance Evaluation Metrics

The prototype was evaluated using the following qualitative usability metrics:

- **Ease of Navigation:** Assessed based on clarity of user flow and screen transitions.
- **User Engagement:** Determined by presence of smart overlays and friendly icons.
- **User satisfaction:** It reflects the overall degree to which users feel contented, his or her experience and how positively impacted using the system during task interaction.

Data Presentation and Analysis

The FlowFrame prototype was developed with a focus on front-end design using visual prototyping tools, requiring minimal computational resources.

The system requirements listed in Table 1 and Table 2 reflect the basic hardware and software environments necessary for both the development and use of the prototype.

Table 1: Depicts the Hardware Requirements

Component	Minimum Requirement
Processor	Dual-core processor (Intel i5)
Memory (RAM)	4 GB RAM
Storage	2 GB of free disk space
Display	1366 × 768 resolution or higher
Internet Connectivity	Stable broadband connection for cloud-based operations

Table 2: Depicts the Software Requirements

Software	Specification
Operating System	Windows 10
Web Browser	Google Chrome (latest version),
Design Tool	Figma (cloud-based interface design and prototyping tool)
Browser Extension	Font Awesome plugin (for icon integration and UI clarity)

Software Development and Testing Tools

This prototype is a non-functional simulation designed using UI/UX design principles. The following tools and technologies were utilized during development:

1. **Figma**: The primary tool for designing and simulating wireframes and interface navigation. It supported prototyping through frame linking and interaction flows, enabling rapid iteration and user feedback incorporation.
2. **Font Awesome**: Integrated through Figma plugins to provide a consistent and recognizable icon set for navigation, settings, AI assistance, and other interface elements.

Testing in this context involved walkthroughs and heuristic evaluations to verify interface coherence, navigation logic, and component placement. Feedback from usability experts and potential users was collected to identify areas for improvement in flow and clarity.

Ease of Use

This measure how simple and intuitive it is for users to complete tasks easily. This testing phase of the study comprises of three (3) tasks criteria given as follows:

1. Follow sidebar flow from Dashboard → Builder → Preview
2. Identify navigation logic between “Start New App” and “Builder”
3. Understand how to return to Dashboard from builder

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073

Table 3: shows the ease of use of the FlowFrame prototype model.

Task	**Users Completed (out of 35)**	**Avg Score (1–5)**
Task 1	35	4.5
Task 2	31	4.43
Task 3	35	4.5

The table above presents the results of three usability tasks completed by 35 participants.
Task 1 had a full completion rate, with all 35 users successfully navigating the intended action. It achieved an average satisfaction score of 4.5, indicating that users found the task both intuitive and smooth.
Task 2, only 31 out of 35 users completed the task successfully, suggesting a minor usability challenge or unclear interface element. This is reflected in its slightly lower average score of 4.43, which still indicates general satisfaction but reveals room for improvement in user guidance or flow.
Task 3 had a full completion rate, with all 35 users successfully navigating the intended action, which demonstrates strong task clarity. However, the average score of 4.2, slightly below Task 1, may imply that while users could complete the task, some experienced minor friction or hesitation during the process.
Overall, the data suggests that while the prototype supports user actions, refinements in clarity or visual cues may further improve user satisfaction and ease of use across all tasks.

User Engagement

User engagement refers to the degree to which users are actively involved, interested, and emotionally invested in interacting with a digital product or system. In the context of low-code development platforms (LCDPs), engagement can be reflected in how users explore optional features, interact with interface elements like icons and overlays, and navigate beyond required tasks.
Task Completion Rate (TCR) is an important metric that is used to measure user engagement. In this study, user engagement was evaluated based on the completion of engagement-focused tasks within the prototype. These tasks were designed not just to assess basic functionality, but to observe how users responded to elements intended to capture interests such as smart suggestion popups, interactive icons, and visually inviting layouts.
The engagement score (Task Completion Rate) is given as:

$$\text{Task Completion Rate} = \left(\frac{\text{Users who Completed Task}}{\text{Total Users}}\right) \times 100 \qquad (2)$$

User Engagement Analysis of the FlowFrame Prototype model

All thirty-five (35) participants took part in task 1, four (31) out of the five (5) participants took part in task 2 while all 35 participants took part in carrying out task 3. The User Engagement Score (task completion rate) for the three (3) tasks is then given as

$$\text{Task 1} = \left(\frac{35}{35}\right) \times 100 = 100\%$$
$$\text{Task 2} = \left(\frac{31}{35}\right) \times 100 = 88.57\%$$
$$\text{Task 3} = \left(\frac{35}{35}\right) \times 100 = 100\%$$

Table 4 shows the User Engagement in the FlowFrame Protype Model while figure 3 shows the graphical representation as a bar chart

Table 4: User Engagement (Task Completion Rate) in the FlowFrame Protype Model

Task	**User Engagement %**
Task 1(Follow sidebar flow from Dashboard → Builder → Preview)	100%
Task 2 (Identify navigation logic between “Start New App” and “Builder”)	88.57%

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073

Task 3 (Understand how to return to Dashboard from builder)	100%

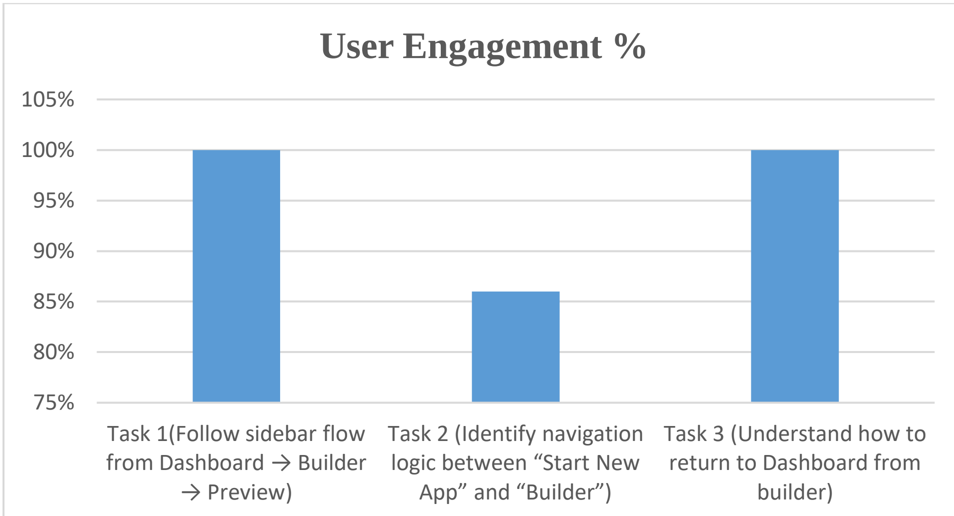

Figure 3: Bar chart showing the Flowframe Prototype model

As depicted in Figure 3, the User Engagement of Task 1 is 100%, 88.57% in Task 2, and 100% in Task 3.According to Mifsud(2015), the overall user engagement is calculated as follow:

Overall User Engagement % = $\left(\frac{Total\ Engagements\ Completed}{Total\ Task \times \text{ Number of Users}}\right) \times 100$ ………………….(3)

Where

Total engagement completed= engagement in task 1+engagement in task 2 +engagement in task 3

Overall User Engagement% = $\left(\frac{101}{3 \times 35}\right) \times 100$ = $\left(\frac{100}{105}\right) \times 100$ = 96.19%

User Satisfaction

After all the tasks on the prototype model were completed, participants were asked to take part in a post-task survey to assess the usability of the prototype model and to measure their user satisfaction with the prototype model. The questionnaire used a 5-point likert scale, ranging from 1 (strongly disagree) to 5 (strongly agree). The data obtained from the post-task questionnaire for each application and prototype are presented and calculated as using User Satisfaction formula by Mifsud(2015 :

$$\text{User Satisfaction} = \left(\frac{(1*SD)+(2*D)+(3*N)+(4*A)+(5*SA)}{\text{Total Number of Responses} * 5}\right) \times 100 \ldots\ldots\ldots\ (4)$$

Where:

SD = total strongly disagree

D = total disagree

N = total neutral

A = total agree

SA = total strongly agree

User Satisfaction Analysis of FlowFrame prototype

Table 5 shows the number of users that answered different questions based on their satisfaction ranging from 1 (strongly disagree) to 5(strongly agree).

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073

Table 5: Responses from the FlowFrame prototype model

Question	Strongly Disagree (1)	Disagree (2)	Neutral (3)	Agree (4)	Strongly Agree (5)
1. I was satisfied with the overall experience of using the prototype.	0	0	3	15	17
2. The system felt comfortable and easy to interact with during all tasks.	0	0	3	18	14
3. I would be willing to use a platform like this again in the future.	0	0	5	17	13
4. The interface made it easy to understand what to do at each step.	0	0	1	19	15
5. I felt confident while navigating and using the prototype.	0	0	4	20	11
Total	0	0	16	89	70

The coded values for Strongly Disagree, Disagree, Neutral, Agree, and Strongly Agree are 1, 2, 3, 4, and 5 respectively. Table 4 depicts that the frequency of Strongly Disagree responses is 0, Disagree is 0, Neutral is 16, Agree is 89 and Strongly Agree is 70. The sum of these values gives us the total number of responses which is 175 and user satisfaction is given as:

$$\text{User satisfaction} = \left(\frac{(1\times0)(2\times0)(3\times16)(4\times89)(5\times70)}{35\times5\times5}\right)\times 100 = 86.1\%$$

The usability testing and performance evaluation of the FlowFrame prototype model were based on three core metrics: Ease of Use, User Engagement, and User Satisfaction. The results derived from hands-on user testing, post-task questionnaires, and computed performance scores reflect the functional effectiveness of the prototype and highlight areas of success and potential improvement.

Results

Thirty-five students participated in the usability evaluation and completed the assigned tasks and post-task questionnaire. Three predefined tasks were used to assess navigation, ease of use, and interaction flow within the FlowFrame prototype.

Task performance

Task 1 (Dashboard → Builder → Preview) recorded 35 successful completions with an average perceived ease-of-use score of 4.50.

Task 2 (Locating and using the "Start New App" action) recorded 31 completions with an average score of 4.43.

Task 3 (Returning to the Dashboard from the Builder) had 35 completions with an average score of 4.50.

User engagement

The task completion rates were as follows:

- Task 1: 100% (35/35)
- Task 2: 88.57% (31/35)
- Task 3: 100% (35/35)

The overall average engagement rate across the three tasks was calculated as 96.19%. A total of 175 item responses (35 participants × 5 questions) were recorded from the post-task survey. Each item was rated from 1 (strongly disagree) to 5 (strongly agree). The overall user satisfaction score was 86.1%.

Discussion

The results demonstrate that the FlowFrame prototype provides high usability and consistent task performance among participants. The perfect completion rates for Tasks 1 and 3 indicate that navigation flows such as moving between the dashboard, builder, and preview screens are intuitive and easy to follow. The slightly lower completion rate in Task 2 suggests that the "Start New App" action may require clearer labeling or improved visual prominence to reduce confusion for first-time users.

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073

The high user satisfaction score (86.1%) further confirms the effectiveness of the design decisions made in the prototype, especially in areas related to layout clarity and interface responsiveness. These findings are consistent with earlier studies on low-code usability challenges, which emphasize the importance of clear visual guidance and simplified workflows for non-technical users. However, the reduced sample size (35 instead of the calculated 70) limits the statistical generalization of the findings. This limitation means that the results should be interpreted as formative usability insights rather than fully generalizable outcomes. Nonetheless, the strong task metrics and satisfaction ratings provide useful evidence supporting the prototype's design direction and areas requiring refinement.

Conclusion

This research set out to address the usability limitations of existing Low-Code Development Platforms (LCDPs) by designing and developing a user-friendly and intelligent prototype. The study followed the Design Science Research Methodology (DSRM), which provided a structured approach to problem identification, artifact design, demonstration, evaluation, and communication. The prototype, created using Figma, incorporated improved user flow, intuitive interface components, and AI-assisted design suggestions to enhance the overall usability of the platform.

Data collected during the demonstration and evaluation phases indicated that the developed prototype significantly improved user experience in comparison to traditional LCDPs. Key metrics such as task completion time, navigation efficiency, and user satisfaction showed measurable improvement. The positive feedback from users affirms the relevance and effectiveness of the design decisions taken throughout the development process. Ultimately, this study contributes to the growing body of work on enhancing LCDPs by demonstrating that usability-centered and intelligent design can bridge existing gaps and foster more inclusive and efficient development environments, especially for non-technical users. The study recommends that future research should expand the prototype into a fully functional platform and conduct large-scale usability testing with diverse user groups. In addition, in a full deployment, quantitative metrics such as task completion time, error rate, and user satisfaction surveys would be introduced.

References

Akwukwuma, V. V. N., Chete, F. O., Okpako, A. E., & Irabor, M. (2024). User interface design and efficiency using graphic design and animation techniques. *NIPES Journal of Science and Technology Research, 6*(3).

Akwukwuma, V. V. N., Chete, F. O., Okpako, A. E., & Nkwor, U. M. (2024). Usability analysis of e-commerce mobile applications. *NIPES Journal of Science and Technology Research, 6*(3).

Ardito, C., Buono, P. D., Costabile, M. F., & Lanzilotti, R. (2021). Usability engineering for the design and evaluation of user interfaces. *Journal of Systems and Software, 165*, 110567. https://doi.org/10.1016/j.jss.2020.110567

Del Buono, P., Ardito, C., & Lanzilotti, R. (2022). A comparative evaluation of low-code development platforms: Usability and cognitive load perspectives. *Journal of Visual Languages and Computing, 63*, 101056. https://doi.org/10.1016/j.jvlc.2022.101056

Guthardt, T., Kosiol, J., & Hohlfeld, O. (2024). *Low-code vs. the developer: An empirical study on the developer experience and efficiency of a no-code platform.* MODELS Companion '24 (ACM/IEEE).

Henriques, P. et al. (2021). *A Use Case-based Investigation of Low-Code Development Platforms.* CEUR Workshop Proceedings — qualitative/experimental comparisons using realistic use cases and prototypes.

Käss, S., Strahringer, S., & Westner, M. (2022–2023). *Drivers and Inhibitors / Practitioners' Perceptions on LCDP adoption.* (Conference paper 2022; journal article 2023). See: Drivers & Inhibitors (CBI 2022) and IEEE Access 2023 article.

Mendling, J., Weber, I., van der Aalst, W. M. P., vom Brocke, J., Cabanillas, C., Daniel, F., & Weidlich, M. (2021). Blockchains for business process management: Challenges and opportunities. *ACM Transactions on Management Information Systems, 9*(1), 4:1–4:16.

Mifsud, J. (2015). *Usability metrics: A guide to quantify the usability of any system*. https://usabilitygeek.com/usability-metrics-a-guide-to-quantify-system-usaility/

Palomares, C., Cabot, J., & Gómez, C. (2020). A systematic literature review of low-code development platforms. *Information and Software Technology, 132*, 106441.

Venkatesh, V., & Bala, H. (2020). Technology Acceptance Model 3 and a research agenda on interventions. *Decision Sciences, 39*(2), 273–315.

Westner, M., Käss, S., & Strahringer, S. (2023). *Practitioners' Perceptions on the Adoption of Low Code Development Platforms.* IEEE Access, 11, 29009–29034. Khan, R., Omar, N., & Ismail, S. (2023). Evaluating LCDPs for user registration, database use, and deployment capabilities. *Journal of Digital Platforms and Applications, 5*(2), 102–118.

Cite this article as:
Okpako, E.A., Ojie, D.V., Iyamah, B., Ukadike, D., Chiemeke,, S.C., & Chete, F.O. (2025). Development of a user-centered prototype model for low-code development platforms. *FNAS Journal of Computing and Applications, 2*(4), 43-51. https://doi.org/10.63561/jca.v2i4.1073