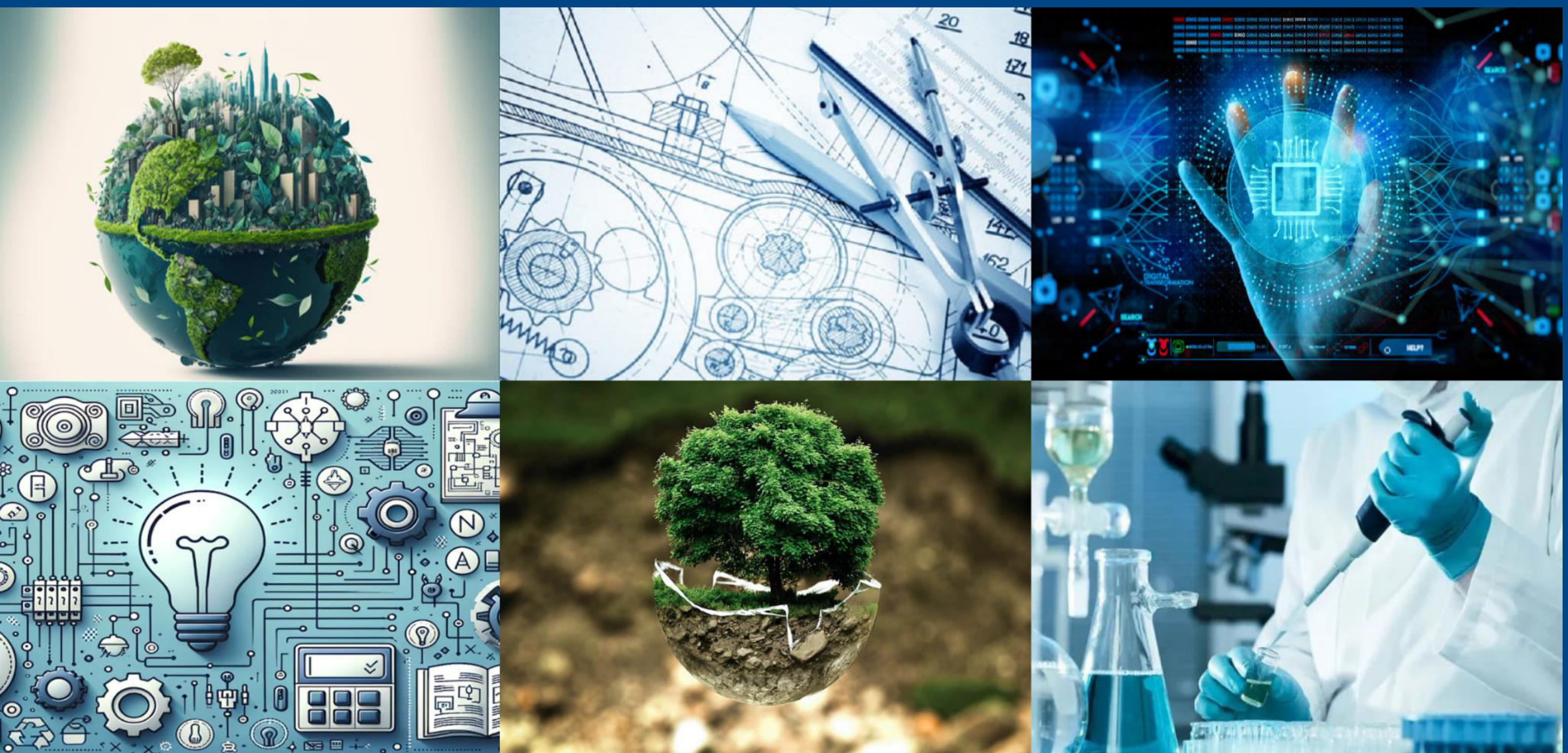
ISSN: 2582-7219

International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.206

Volume 8, Issue 6, June 2025

www.ijmrset.com ijmrset@gmail.com +91 6381 907 438 +91 63819 07438

© 2025 IJMRSET | Volume 8, Issue 6, June 2025| DOI:10.15680/IJMRSET.2025.0806048

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Design and Development of a No-Code Canvas Application for Customizable Dashboards

Shravan S S, Brill Brenhill

PG Student, St Joseph Engineering College, Mangaluru, Karnataka, India Assistance Professor, St Joseph Engineering College, Mangaluru, Karnataka, India

ABSTRACT: The difficulty of changing production lines presents a barrier for Original Equipment Manufacturers (OEMs) trying to shorten time to market. Using no-code techniques to tackle these problems is investigated in this study. Here, I describe the conception and implementation of a canvas-based program for building Internet of Things (IoT)-specific dashboards that are customizable. Without writing any code, users may customize and link pre-built widgets to several data sources by simply dragging and dropping them onto a canvas. Wireframes, user stories, and input from BIBA (Bremer Institut für Produktion und Logistik GmbH) are all included in the document that describes the users-centric design approach. Creating a dashboard doesn't have to be complicated for people who don't know much about technology due to my approach.

I.INTRODUCTION

Original Equipment Manufacturers (OEMs) frequently experience extended time-to-market due to the complexity of creating and modifying production lines. The traditional dashboard creation methods, which are sometimes labor-intensive and need a high level of technical competence, make this problem worse.

No-code solutions are becoming a very effective tool for dealing with these issues. By enabling users to build dashboards through simple, visual interfaces without requiring a deep understanding of coding, these platforms streamline the development process. This method is especially useful in the Internet of Things (IoT) space, where interactive and adaptable dashboards are necessary for controlling and monitoring a wide range of sensors and devices.

The concept and development of a no-code canvas-based application that aims to make creating configurable dashboards easier is shown in this article. The goal of the application is to give users who need to visualize and interact with data from multiple sources an easy-to-use and effective solution. The application cuts development time and does away with difficult code by using a canvas where users can drag and drop pre-built widgets.

According to the research, BIBA - Bremer Institut für Produktion und Logistik GmbH will first gather requirements for a robotic use case. User stories, wireframes, and iterative design changes based on user feedback will then be created. Through this procedure, the application's compliance with user experience standards and user needs was guaranteed.

II.LITERATURE REVIEW

The rapid evolution of low-code and no-code platforms has significantly transformed the landscape of software development, enabling even non-technical users to create complex applications with minimal coding. This section reviews the existing literature on low-code and no-code platforms, particularly their applications across various domains such as smart agriculture, IoT, manufacturing, and artificial intelligence.

Oteyo et al. (2021) explored the application of low-code tools in building smart agriculture applications, demonstrating the potential of these platforms to integrate with existing agricultural technologies effectively. Their study highlighted the advantages of using low-code platforms to enhance efficiency and data management within farming practices, offering a promising solution to the challenges faced in modern agriculture [1].

Ihirwe et al. (2020) provided a comprehensive review of the state of low-code engineering for the Internet of Things (IoT), underscoring the importance of model-driven engineering in the development of IoT systems. Their research

© 2025 IJMRSET | Volume 8, Issue 6, June 2025| DOI:10.15680/IJMRSET.2025.0806048

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

identified the key challenges and opportunities associated with using low-code platforms for IoT, particularly in terms of scalability and flexibility [2].

The application of low-code platforms in automating business processes within manufacturing environments was extensively studied by Waszkowski (2019). His research focused on how these platforms could streamline complex workflows and improve operational efficiency in industrial settings, contributing to the broader goals of Industry 4.0 [3].

Arora et al. (2020) examined the Sagitec Software Studio (S3), a low-code application development platform designed for Industry 4.0. Their study illustrated how S3 facilitates the rapid development and deployment of industrial applications, thus enabling digital transformation in manufacturing processes [4].

Sufi (2023) reviewed the practical applications of algorithms in low-code/no-code platforms, particularly in research-oriented contexts. This study emphasized the potential of these platforms to incorporate advanced algorithms, allowing researchers and developers to build sophisticated applications without extensive programming knowledge [5].

In the domain of smart environments, Desolda, Ardito, and Matera (2017) explored how model-driven development paradigms empower end-users to customize their environments using domain-specific tools. Their research highlighted the flexibility and user-centric approach of low-code platforms, which allow for significant customization without requiring deep technical expertise [6].

III.METHODOLOGY

An organized approach was used during the no-code canvas-based application development process to guarantee the production of an efficient and user-friendly dashboard customization tool. The main steps in the research and development process are described in this part, starting with requirement collection and ending with design and evaluation.
The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

Requirement gathering:
Identifying user demands and obstacles in the context of dashboard development was the first step in the process. Particular requirements were acquired for a robotic use case at Bremer Institut für Produktion und Logistik GmbH, often known as BIBA. This included:

- Gathering information from stakeholders through workshops and interviews.
- Examining current solutions to find weaknesses and potential areas for development.
- Defining the essential functions and features that the program needs.

User Stories
In order to clarify and convey the application's functionalities from the viewpoint of the user, user stories were developed. Every user story adheres to this structure:
As a user, [function], so that [outcome].
Example User Stories:

- "As a user, I want to drag and drop widgets onto the canvas so that I can position them precisely where needed."
- "As a user, I want to enter configuration details like IP address and protocol so that I can establish a connection to the data source."

These stories guided the design and development process by outlining specific user needs and expected interactions.

Design and Wireframing:
The design process was iterative, involving the following steps:

© 2025 IJMRSET | Volume 8, Issue 6, June 2025| DOI:10.15680/IJMRSET.2025.0806048

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- Wireframing: To illustrate the application's functioning and layout, preliminary wireframes were made. The positioning and details of the user interface elements werespecified using mid-fidelity wireframes.
- Design Iterations: Several wireframe iterations were created based on input from stakeholders, including BIBA. Applying fundamental design concepts like Jakob's Lawguaranteed usability and compliance with well- known design patterns.

The initial wireframe design is shown in Fig. 1. This illustrates the placement of the widget window, canvas, and properties window.

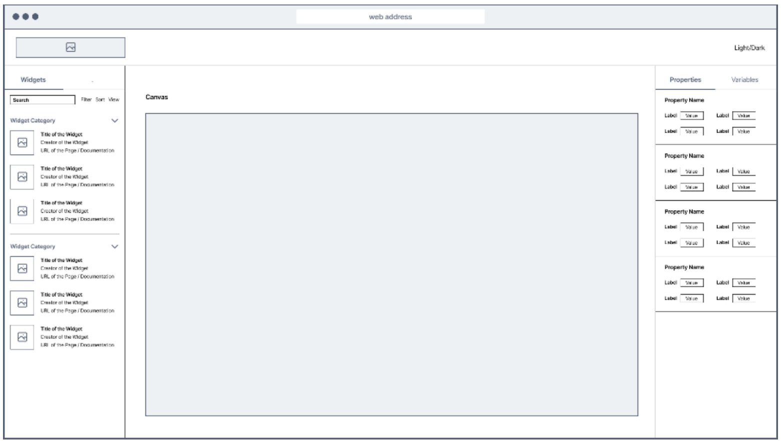


Figure 1: Initial wireframe design for the dashboard interface User Interface and Experience (UI/UX) Design:

The application's user interface was designed to be intuitive and easy to use. Key aspects included:

- A canvas area where users can drag and drop widgets.
- A widget window providing pre-built widgets for easy selection.
- A properties window for configuring widget properties and data connections.

Design tools such as Figma were used for creating and refining the UI/UX. Feedback was continuously integrated to enhance user experience.

User Interaction Flow:

The user interaction flow is depicted in the Fig. 2 below, which illustrates the steps a user takes to create and configure a dashboard. This visual representation helps in understanding the sequence of actions within the application.

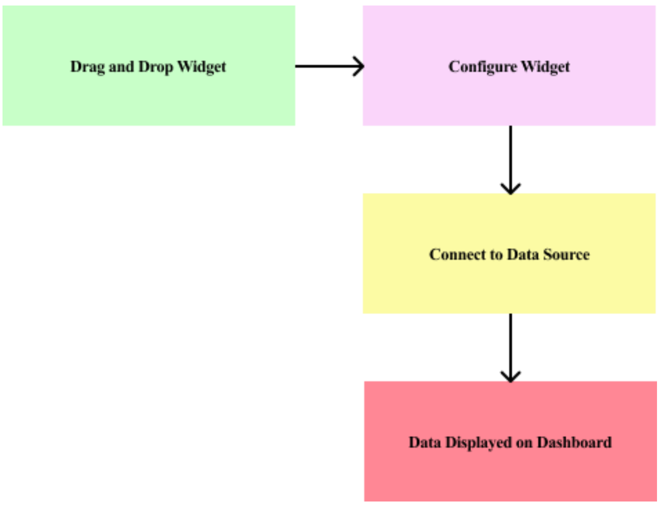


Figure 2: User Interaction Flow. This shows the steps a user takes to create a dashboard.

ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 8.206| ESTD Year: 2018|

International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Evaluation and Feedback:
The design was evaluated through:

- User Testing: Conducting usability tests with real users to gather feedback on functionality and ease of use.
- Iterative Refinement: Incorporating feedback into design iterations to address issues and improve the application.

Future Work:
The next phase involves transforming the current design into a fully functional application. This will include:

- Developing the application based on the refined design.
- Conducting further testing and validation to ensure the application meets user needs and performs effectively.

IV. DISCUSSION

The development and implementation of the no-code canvas-based dashboard application have shown promising results, demonstrating the potential to significantly enhance the process of creating customizable dashboards. This section discusses the implications of these results, the challenges faced, and the future directions for this project.

Implications of Findings:
The findings from this project indicate that a no-code approach can simplify the dashboard creation process for users with varying levels of technical expertise. By providing an intuitive drag-and-drop interface, the application enables users to design and configure dashboards without needing extensive programming knowledge. This democratization of dashboard creation can lead to faster deployment and more widespread use of IoT monitoring solutions.

Addressing OEM Challenges:
Original Equipment Manufacturers (OEMs) often face challenges in reducing time to market due to the complexities involved in setting up and adapting manufacturing lines. The no-code dashboard application addresses this issue by streamlining the design process and reducing dependency on specialized developers. By allowing users to quickly prototype and deploy dashboards, OEMs can respond more swiftly to market demands and operational changes.

User-Centric Design:
The user-centric design process, which involved gathering user requirements and iterative feedback, was crucial in creating a product that meets the needs of its users. The positive feedback regarding the interface's ease of use and flexibility underscores the importance of involving users throughout the development process. This approach ensures that the final product is not only functional but also aligned with user expectations and workflows.

Challenges and Limitations:
Despite the successes, several challenges were encountered:
Protocol Integration: Supporting multiple communication protocols is a complex task. While initial efforts focused on a few key protocols, expanding this support will require substantial effort.
Widget Customization: Developing a comprehensive library of pre-built widgets that cater to diverse use cases is challenging. Ensuring that these widgets are both versatile and easy to configure is essential for user satisfaction.
Performance Optimization: Ensuring that the application performs efficiently under various conditions, including different data loads and network environments, remains a continuous effort.

V. FUTURE WORK

While the development of the no-code canvas-based dashboard application has yielded significant achievements, there are several areas where further enhancements can be pursued to increase its capabilities and user-friendliness. This section outlines the potential directions for future work.

Expansion of Widget Library:
One of the key areas for future development is the expansion of the pre-built widget library. Including a broader range of widgets will cater to more diverse use cases and industries. Advanced widgets, such as those for real-time analytics,

predictive maintenance, and advanced data visualization, can provide users with more powerful tools for monitoring and controlling their IoT systems.

Enhanced Customization and Flexibility:
Providing advanced customization options for widgets and dashboards is another critical area. This includes enabling users to script widget behaviors and interactions, allowing for more complex and tailored applications. By incorporating a scripting interface, users with programming knowledge can extend the functionality of the pre-built widgets and create highly specialized dashboards.

Improved Protocol Support:
Expanding the range of supported communication protocols is essential for ensuring compatibility with a wider array of IoT devices and systems. Future work should focus on integrating additional protocols commonly used in the industry, making the application more versatile and accommodating to various user needs.

Performance and Scalability Enhancements:
As the application is adopted by more users, ensuring its performance and scalability becomes paramount. Future work should include optimizing the application's architecture to handle larger datasets and more concurrent users efficiently. Implementing performance monitoring and automated scaling solutions can help maintain a responsive and reliable user experience.

User Training and Documentation:
To maximize the utility of the application, comprehensive user training and documentation are crucial. Developing detailed guides, tutorials, and best practices can help users get the most out of the application. Additionally, creating a robust support system, including forums and customer service channels, can assist users in troubleshooting and learning advanced features.

Integration with External Systems:
Future work could also explore the integration of the dashboard application with other enterprise systems, such as ERP, CRM, and MES. This integration can provide users with a more holistic view of their operations and enable seamless data flow between different systems. APIs and connectors can be developed to facilitate these integrations.

User Feedback and Continuous Improvement:
Lastly, establishing a continuous feedback loop with users will be essential for ongoing improvement. Regularly soliciting user feedback, conducting usability studies, and analyzing usage data can help identify areas for enhancement and ensure that the application evolves in line with user needs and technological advancements.

VI. CONCLUSION

In this paper, I have presented the development and implementation of a no-code, canvas-based dashboard application designed to streamline the creation and customization of IoT dashboards. The primary aim was to simplify the process for Original Equipment Manufacturers (OEMs) and other users by enabling them to quickly and easily develop dashboards without needing extensive coding knowledge.

The methodology involved gathering detailed user requirements, creating comprehensive user stories, and iterating through wireframe designs to refine the user interface and user experience. The application was developed with a user-centric approach, ensuring that it is intuitive and accessible to a wide range of users, including those without technical backgrounds.

Through collaboration with BIBA – Bremer Institut für Produktion und Logistik GmbH at the University of Bremen, the application was continuously evaluated and improved based on user feedback. The results demonstrated that the application significantly reduces the time and complexity involved in creating IoT dashboards, making it a valuable tool for monitoring and controlling various devices and sensors.

Future work will focus on expanding the widget library, enhancing customization options, supporting additional communication protocols, and improving the overall performance and scalability of the application. Additionally, providing comprehensive user training and documentation, integrating with external systems, and establishing a continuous feedback loop will be crucial for the ongoing success and evolution of the application. In conclusion, the no-code canvas-based dashboard application represents a significant advancement in the field of IoT, offering a powerful and flexible tool for users to efficiently develop and deploy dashboards. This work lays the groundwork for future enhancements that will further optimize the user experience and broaden the application's capabilities, ensuring it remains a relevant and indispensable tool in the rapidly evolving landscape of IoT.

REFERENCES

1. Oteyo, I.N., et al. "Building smart agriculture applications using low-code tools: the case for discopar." In: IEEE AFRICON Conference. IEEE, 2021.
2. Ihirwe, F., et al. "Low-code engineering for internet of things: a state of research." In: Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, MODELS-C 2020.
3. Waszkowski, R. "Low-code platform for automating business processes in manufacturing." In: IFAC-PapersOnLine, 2019.
4. Arora, R., Ghosh, N., Mondal, T. "Sagitec software studio (S3) – a low code application development platform." In: 2020 International Conference on Industry 4.0 Technology, I4Tech 2020.
5. Sufi, F. "Algorithms in Low-Code-No-Code for Research Applications: A Practical Review." Algorithms, 2023.
6. Desolda, G., Ardito, C., & Matera, M. "Empowering End Users to Customize their Smart Environments: Model, Composition Paradigms, and Domain-Specific Tools." ACM Transactions on Computer-Human Interaction (TOCHI), 24(2), 2017.

INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com