

SoftBoard: A Multi-Agent Tool for the Creation and Evaluation of Low-Fidelity Prototypes

Gabriel R. S. Scapim
State University of Maringá
Maringá, Brazil
scapimgabriel@gmail.com

Gislaine C. L. Leal
State University of Maringá
Maringá, Brazil
gcilleal@uem.br

Guilherme C. Guerino
State University of Paraná
Apucarana, Brazil
guilherme.guerino@ies.unespar.edu.br

Abstract

User Experience (UX) is recognized as a critical factor for the success of digital products, particularly in software startups, environments marked by time constraints, limited resources, and low maturity in design practices. Building Minimum Viable Products (MVPs) through low-fidelity prototyping represents a well-established strategy for rapid validation cycles at reduced cost. A systematic literature mapping, however, revealed gaps in the ecosystem of available tools: a predominance of general-purpose solutions adapted for prototyping, the absence of integrated methodological guidance, and the incipient use of Artificial Intelligence in the design process. This paper presents SoftBoard, a web-based tool for the creation and evaluation of low-fidelity prototypes in the context of MVP development. The tool integrates a prototype editor, team-based project organization, and a multi-agent system based on large language models that supports requirements elicitation and refinement, automates prototype generation, and evaluates interface quality based on usability heuristics. This integration aims to reduce reliance on prior UX expertise, standardize the prototyping process, and support teams in building MVPs aligned with user needs. As future work, a feasibility study with software professionals is currently underway.

Demo video: <https://doi.org/10.5281/zenodo.20369601>

Keywords

User Experience, Software Startups, Minimum Viable Product, Low-Fidelity Prototyping, Wireflows, Artificial Intelligence.

1 Introduction

User Experience (UX) plays a central role in the development of digital products, encompassing users' perceptions and responses resulting from the use or anticipated use of a system [14]. Ensuring an adequate UX from the early stages of development is particularly critical in Software Startups, environments characterized by strong time pressure, limited resources, small teams, and high market uncertainty [8]. In such contexts, design decisions are often made empirically and with limited methodological support, which may lead to products that are misaligned with user needs, increased rework, and low user adoption [12, 18].

One strategy adopted to mitigate these risks is the development of a Minimum Viable Product (MVP), which enables short and iterative cycles of validated learning [23]. To fulfill this role, an MVP must balance simplicity with the ability to generate meaningful evidence about user behavior. In this context, low-fidelity prototypes, such as wireframes and digital sketches, are particularly suitable for the early stages of development, as they enable rapid iterations, low modification costs, and a focus on interface structure and

navigation flow rather than visual aesthetics [2, 4]. Nevertheless, many Software Startups still face difficulties in structuring MVPs that achieve this balance, whether due to a lack of methodological knowledge or the absence of tools and processes tailored to their context [18, 20].

To systematize the development of UX-oriented MVPs in Software Startups, the StartFlow method was proposed [11]. The method adopts wireflows as its central artifact, visual representations that combine wireframes and user flow maps, enabling the simultaneous visualization of screen structures and navigation transitions. The process is organized into three main stages: identifying and structuring MVP functionalities, transforming these functionalities into wireflows, and reviewing the developed prototype. Feasibility studies with software professionals indicate that StartFlow contributes to prioritization, reduces inconsistencies, facilitates alignment among product, technology, and UX teams, and promotes a more strategic prototyping process [11].

Despite these contributions, the application of StartFlow relies exclusively on textual documentation, leaving the choice of prototyping tools to practitioners. The absence of a dedicated tool may hinder the adoption of the method, increase variability in outcomes, and make its application highly dependent on the experience and preferences of individual team members. To investigate this scenario, a Systematic Mapping Study was conducted on low-fidelity prototyping tools. The results revealed gaps, including functional limitations in existing tools, limited incorporation of structured design methods and heuristics, and the still incipient adoption of Artificial Intelligence (AI) to support the prototyping process [7].

In response to these challenges, this paper presents SoftBoard, a web-based tool designed to operationalize the StartFlow method for the creation and evaluation of low-fidelity prototypes. SoftBoard integrates a wireframe and wireflow editor, team-based project management, and a guided workflow (wizard) structured around the three stages of StartFlow, supported by a multi-agent system based on Large Language Models (LLMs). Within this system, specialized agents collaborate throughout the process by supporting requirements elicitation and validation, automating wireflow generation, and evaluating interfaces based on StartFlow-specific heuristics as well as Nielsen's Usability Heuristics [21]. In this way, SoftBoard aims to lower the adoption barrier of the StartFlow method, standardize the prototyping process, and support teams with limited UX expertise in developing MVPs that are aligned with user needs.

The remainder of this paper is organized as follows. Section 2 presents the theoretical background and related work. Section 3 describes SoftBoard, its architecture, and its main functionalities. Section 4 presents a usage example. Finally, Section 5 concludes the paper.

2 Background

This section presents the theoretical background that underpins this work. First, concepts related to low-fidelity prototyping and MVP development are discussed. Next, the StartFlow method is described. Finally, related tools and studies on low-fidelity prototyping are presented.

2.1 MVPs and Low-Fidelity Prototyping

The Minimum Viable Product (MVP) is a central concept in digital product development and is defined as an initial version of a product designed to maximize validated learning about customers with the least possible effort [18, 23]. Rooted in the Lean Startup movement, the MVP emphasizes the delivery of a minimal functional solution that generates real value while enabling the rapid collection of user feedback [3]. Literature reviews indicate that concepts such as a minimum set of features, an initial prototype, reduced effort, and customer feedback are recurrent elements in MVP definitions [16].

The development of MVPs in startups takes place under constraints related to time, team size, and financial resources, often accompanied by communication difficulties and product–market misalignment [18]. Recent studies reinforce that the MVP should be understood as an experimental mechanism for value validation rather than merely a reduced technical artifact [15]. In this context, the MVP also plays a guiding role by supporting the prioritization of essential and testable functionalities, particularly in teams with lower levels of technical maturity [10]. The adoption of more structured practices and processes contributes to improving both the speed and quality of decision-making in environments characterized by uncertainty [20].

Interface prototyping plays a central role in this process, particularly within the User-Centered Design (UCD) approach, which prioritizes users' needs and limitations throughout the development of digital solutions [1, 9]. The iterative creation and evaluation of prototypes enable teams to validate ideas, anticipate problems, and establish early visual representations before implementation [19, 24]. Low-fidelity prototypes are characterized by their simplicity, low cost, and ease of modification, making them suitable for exploring initial concepts and structuring interaction flows without requiring advanced visual or technical details [6, 17]. Examples include paper sketches, wireframes, and rapidly created digital representations [4, 17], which are considered particularly appropriate for MVP development due to their support for rapid iteration and idea validation [2].

2.2 The StartFlow Method

StartFlow was designed to support startups during the early stages of digital product conception, enabling teams to represent the essential functionality of an MVP in a clear and cost-effective manner before implementation through the use of wireflows [11]. These artifacts combine screen sketches and navigation flows, allowing teams to visualize the expected user behavior during interaction with the system.

The method organizes the work into three main stages: identifying and structuring functionalities, transforming these functionalities into wireflows, and reviewing the interaction flows. This process guides teams in understanding how each operation will be

performed by users, which interface elements are required at each stage, and how transitions between screens should occur. As a result, the approach promotes alignment among technology, product, and UX teams while making the rationale behind design decisions more transparent and collaborative [11].

The wireflows produced during the method allow the simultaneous visualization of the minimal interface structure and user navigation paths. Elements such as buttons, input fields, and menus are represented in a simplified manner, while connectors indicate each step of the interaction flow. This representation enables teams to examine alternative scenarios, identify dependencies among functionalities, and anticipate error situations or unlikely navigation paths [11].

Feasibility studies conducted with software professionals indicate that the method's guiding questions serve as a reasoning framework, helping participants organize priorities, refine requirements, and reduce common omissions during the early stages of product conception. Participants also highlighted that the visual representation of flows improves the collective understanding of product behavior, facilitates communication among teams, and can be applied at a low cost, even by professionals without formal UX training [11].

2.3 Related Tools

To identify the state of the art in low-fidelity prototyping tools, a Systematic Mapping Study was conducted [7]. The initial search returned 1,493 publications, and after applying the inclusion and exclusion criteria, 37 primary studies were selected, from which 22 tools were identified. Among the most frequently cited tools were Figma¹, Balsamiq², and Microsoft PowerPoint³, followed by Adobe XD⁴, Pencil Project⁵, and Sketch⁶.

One of the main findings concerns the predominance of general-purpose tools adapted for prototyping. Solutions such as Figma, PowerPoint, Adobe XD, and Sketch were not originally designed specifically for low-fidelity prototyping, yet they account for most of the citations in the analyzed studies. This pattern suggests that practitioners rely on these tools due to familiarity, integration with established workflows, or the lack of sufficiently comprehensive dedicated alternatives. Although these tools provide a wider range of features, their use may reduce methodological guidance and focus during the early stages of design.

The study also revealed that only a small portion of the analyzed tools incorporate AI-based functionalities, such as automatic prototype generation or suggestions for layout and usability improvements. Uizard⁷ stands out by converting hand-drawn wireframes into digital prototypes and providing automated suggestions. Nevertheless, this approach remains relatively uncommon among the available tools.

Another finding was the limited integration of structured methods, heuristics, and design guidelines within prototyping tools.

¹<https://www.figma.com>

²<https://balsamiq.com>

³<https://www.microsoft.com/pt-br/microsoft-365/powerpoint>

⁴<https://adobexdplatform.com>

⁵<https://pencil.evolus.vn>

⁶<https://www.sketch.com>

⁷<https://uizard.io>

Only 5 of the 22 identified tools incorporate some form of method or guideline, and only 2 are dedicated exclusively to low-fidelity prototyping. Among the existing examples, Figma and Adobe XD allow the integration of guidelines through plugins, while WOZ Pro incorporates the Wizard of Oz method for evaluation [13]. However, such integration is indirect and depends on the initiative of practitioners. In most tools, there is no built-in guidance on how to structure interface decisions, leaving users responsible for ensuring the quality and consistency of the resulting solutions.

Recent studies point to a growing trend in the use of Artificial Intelligence to support activities related to design and requirements engineering. A conceptual framework for generating user stories from real-time conversations with users was proposed to reduce dependence on pre-existing requirements documentation [22]. Similarly, a ChatGPT-based tool was developed to support the automated identification and classification of usability defects according to Nielsen’s heuristics, assisting teams that lack dedicated usability specialists [5].

3 SoftBoard

This section presents SoftBoard, describing its overview, target users, requirements, system architecture, and main implemented functionalities. For improved readability and visualization, high-resolution versions of both architectural diagrams are available in the artifacts section of the replication package, accessible at <https://doi.org/10.5281/zenodo.20369601>.

3.1 Overview and Target Users

SoftBoard is a web-based tool designed to support the creation and evaluation of low-fidelity prototypes. Its primary goal is to lower the adoption barrier by providing a structured environment that guides practitioners from MVP functionality definition to wireflow generation and evaluation, without requiring prior experience in UX or design tools.

The tool is primarily intended for professionals working in software startups, who need to build MVPs quickly and systematically but often lack dedicated UX specialists within their teams.

SoftBoard is proprietary software, all rights reserved. The source code is not publicly available at this time, as a feasibility study with software professionals is currently underway.

3.2 Requirements

The following requirements were defined for SoftBoard based on the findings of the Systematic Mapping Study [7] and the operational needs of the StartFlow method [11]. The requirements were organized into three modules: Platform, Editor, and StartFlow.

3.2.1 Platform.

- **FR1. User registration:** The tool shall allow users to register and edit their personal information.
- **FR2. Authentication:** The tool shall provide secure authentication, ensuring that each user can access only their own teams and information.
- **FR3. Password recovery:** The tool shall allow users to recover access through password reset functionality.

- **FR4. Team management:** The tool shall allow users to create, edit, and delete teams, which serve as collaborative workspaces.
- **FR5. Member management:** The tool shall allow users to add and remove team members, with role-based permission control.
- **FR6. Board management:** The tool shall allow users to create, edit, list, and delete boards within a team. Each board represents a prototyping project and centralizes the editor, requirements, interaction history, and the current StartFlow stage.
- **FR7. Board sharing:** The tool shall allow users to generate a public link for the interactive visualization of completed boards, facilitating communication with stakeholders.

3.2.2 Editor.

- **FR8. Component gallery:** The tool shall provide a gallery of essential components for mobile interface prototyping, including buttons, input fields, icons, text elements, shapes, and toggles, all configurable through visual properties.
- **FR9. Drag-and-drop support:** The tool shall allow users to add and position components on the canvas through drag-and-drop interactions, with support for hierarchical relationships between components and screens.
- **FR10. Component editing:** The tool shall allow users to resize, reposition, and reorder components across layers, providing automatic alignment guides during manipulation.
- **FR11. Wireflow creation:** The tool shall allow users to create multiple screens and establish navigation connections between them through linked components, representing the MVP interaction flows.
- **FR12. Interactive mode:** The tool shall provide an interactive mode that enables navigation across connected screens, simulating application behavior directly within the editor.

3.2.3 StartFlow.

- **FR13. Guided workflow:** The tool shall structure each board into three sequential stages aligned with StartFlow: requirements organization, wireflow construction, and review, with controlled progression between stages.
- **FR14. Requirements management:** The tool shall allow users to create, edit, and delete requirements in the form of user stories.
- **FR15. Conversational agent:** The tool shall provide a conversational agent that assists practitioners throughout the wizard stages, adapting its behavior, tools, and instructions according to the current stage.
- **FR16. Automatic wireflow generation:** The tool shall allow users to trigger automatic wireflow generation from the defined requirements through a multi-agent system executed asynchronously, with notification upon completion.
- **FR17. Automatic UX evaluation:** The tool shall automatically evaluate generated wireflows based on UX heuristics, assign Likert-scale scores ranging from 1 to 5, and provide improvement suggestions for each criterion.

3.3 Architecture

SoftBoard is a web application organized into three layers: frontend, backend, and infrastructure, as illustrated in Figure 1. The entire application was developed using the TypeScript programming language⁸.

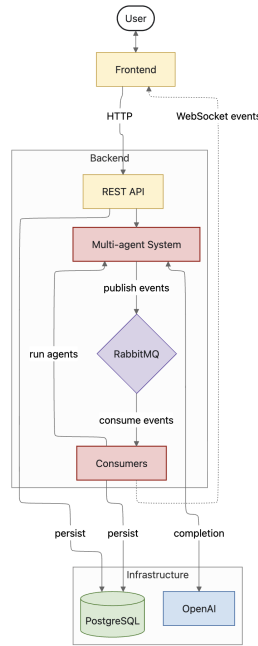


Figure 1: SoftBoard architecture.

3.3.1 Frontend. The frontend was implemented using React⁹. The interface includes the platform screens (authentication, profile, teams, and members) as well as the workspace environment, referred to as *boards*. A board serves as the central integration point of the user experience, bringing together the wireframe and wireflow editor, chat interface, requirements management, and the wizard that guides practitioners through the stages of the method.

Communication with the backend occurs through HTTP requests for synchronous operations and WebSockets for real-time notifications, such as wireflow generation completion and UX evaluation delivery. The interface is available in both Portuguese and English.

3.3.2 Backend. The backend was developed using Node.js¹⁰ and Express¹¹. The REST API handles authentication, password recovery, team and permission management, as well as board and requirement administration. During the wizard execution, the conversational agent operates synchronously through API requests.

When more computationally intensive tasks are required, an event is published to RabbitMQ¹² and processed asynchronously

by consumers responsible for executing specialized agents. Once processing is completed, the frontend is notified via WebSocket.

3.3.3 Infrastructure. PostgreSQL¹³ stores all application data, including users, teams, boards, requirements, wireflows, chat messages, and UX evaluations. RabbitMQ decouples computationally intensive processing from the HTTP request cycle. The entire infrastructure is containerized using Docker Compose¹⁴.

The language models used by the multi-agent system are accessed through the OpenAI API¹⁵.

3.4 Multi-Agent System

The integration with the StartFlow method is achieved through four specialized agents organized into three stages that mirror the wizard workflow (Figure 2). All agents share the same underlying abstraction, differing primarily in the LLM model employed, the available tools, and the instructions provided, which are grounded in the StartFlow guiding questions. The conversational agent uses GPT-4o, whereas the synthesis, generation, and evaluation agents use GPT-5.5.

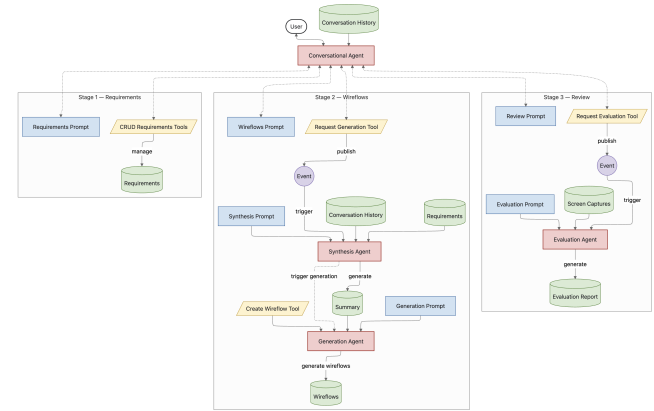


Figure 2: SoftBoard multi-agent system.

During Stage 1 (Requirements), the conversational agent supports the identification and structuring of MVP functionalities as user stories, encouraging reflection on essential user needs and behaviors. Throughout the interaction, the agent incrementally creates, edits, and organizes requirements.

During Stage 2 (Wireflows), when generation is requested, the conversation history and consolidated requirements are submitted for asynchronous processing. First, the synthesis agent verifies the sufficiency of the available information and produces a structured MVP summary. Next, the generation agent transforms this summary into wireflows composed of interconnected screens, which are then persisted in the database. If the information is deemed insufficient, the system returns an error message to the user.

During Stage 3 (Review), images of the generated wireflows are analyzed by an evaluation agent with computer vision capabilities. The system examines eight criteria: four derived from StartFlow

⁸<https://www.typescriptlang.org>

⁹<https://react.dev>

¹⁰<https://nodejs.org>

¹¹<https://expressjs.com>

¹²<https://www.rabbitmq.com>

¹³<https://www.postgresql.org>

¹⁴<https://docs.docker.com/compose>

¹⁵<https://openai.com>

and four based on Nielsen’s Usability Heuristics [21]. Each criterion receives a score ranging from 1 to 5, accompanied by suggestions for improvement.

4 Usage example

This section presents an illustrative usage scenario of SoftBoard to demonstrate the tool’s operation in practice. The example simulates the conception of an MVP for a food delivery application.

The workflow begins on the platform’s board management screen, where projects are organized by teams. Figure 3 shows the main dashboard, in which the board for the “Food Delivery App” project can be viewed.

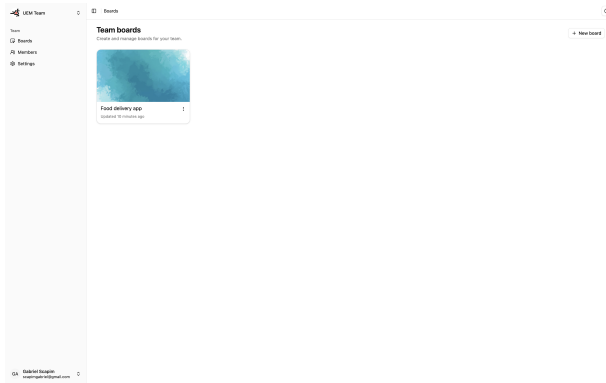


Figure 3: SoftBoard board management screen.

Upon accessing the board, the user is directed to the wizard responsible for identifying and organizing MVP functionalities. At this stage, the user interacts with a conversational agent that employs StartFlow’s guiding questions to support requirements elicitation and refinement. Figure 4 presents the result of this interaction: the conversation history is displayed on the left, while the consolidated list of requirements is shown on the right.

Progression through the workflow is controlled by the tool: users cannot proceed to the next stage unless at least one requirement has been defined, ensuring that subsequent activities are grounded on a consistent minimum scope. This activity illustrates how SoftBoard transforms a typically informal task into a guided and structured process, reducing dependence on prior UX expertise and promoting greater consistency in MVP scope definition.

Once the requirements have been defined, the user proceeds to the wireflow construction stage. At this point, the automatic generation agent can be triggered to create the initial screen structure and navigation connections based on the previously identified functionalities. Figure 5 shows a request for prototype generation and the beginning of the asynchronous processing.

After generation, the wireflows remain fully editable within the integrated editor. Figure 6 presents the editing environment in use, highlighting the canvas containing the complete screen flow, the low-fidelity component gallery, and the properties panel of the selected element.

As in the previous stage, navigation is constrained: the tool requires the existence of at least one screen containing at least

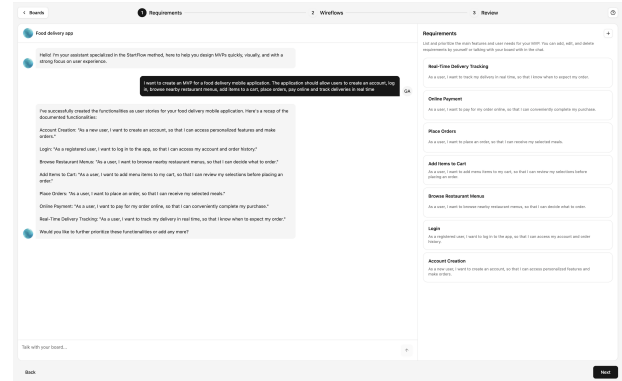


Figure 4: Interaction with the conversational agent and requirements structuring.

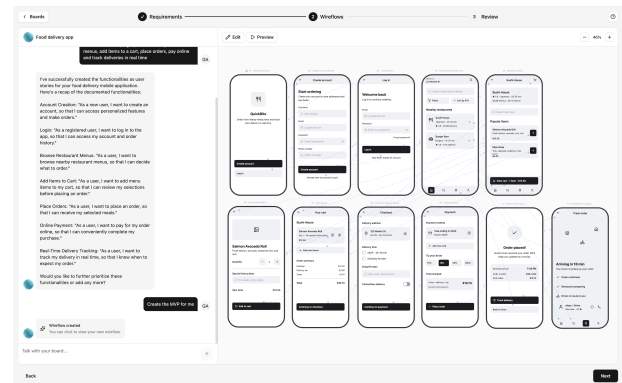


Figure 5: Request for automatic wireflow generation.

one component before allowing progression to the review stage, ensuring that the evaluation is performed on a minimally functional artifact.

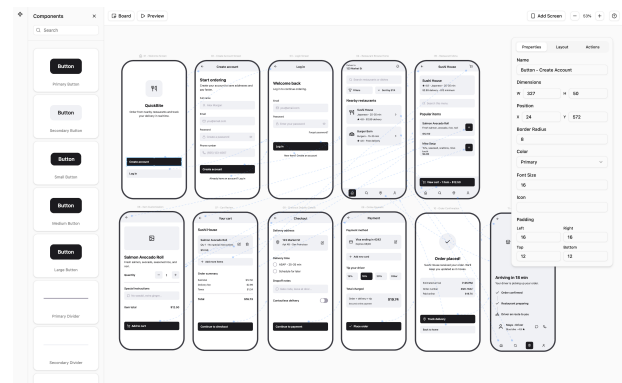


Figure 6: Wireflow editor with screen flow and properties panel.

In addition to static editing, the editor provides an interactive mode that enables navigation between connected screens, simulating application behavior directly within the editing environment. This functionality allows users to traverse the defined navigation flows, validate the consistency of screen transitions, and identify gaps in the user journey. Figure 7 presents an example of this navigation mode.

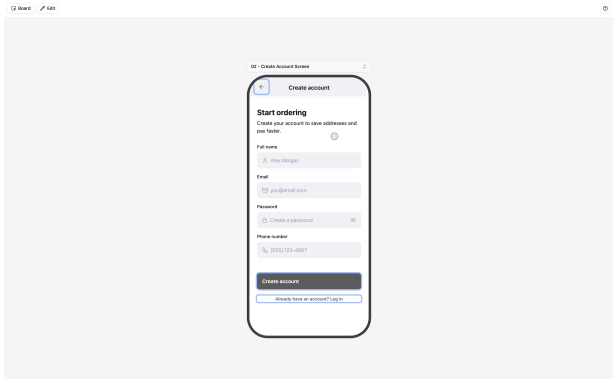


Figure 7: Interactive mode for navigation between connected screens.

After editing is completed, the user advances to the review stage, where the evaluation agent analyzes the wireflows according to StartFlow criteria and Nielsen’s Usability Heuristics. Figure 8 presents the *Board Review* panel, displaying the overall UX score and a detailed breakdown of the evaluated criteria, accompanied by suggestions for improvement.

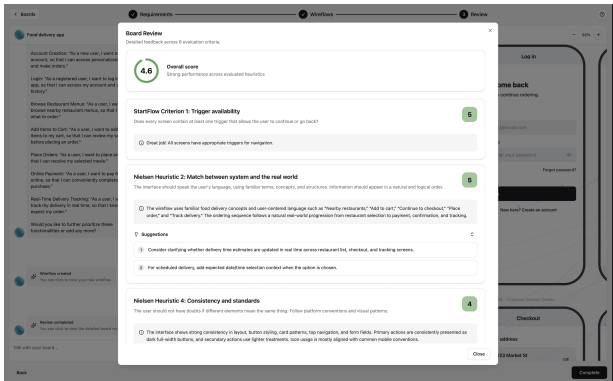


Figure 8: Automatic UX review panel with scores and improvement suggestions.

The described scenario demonstrates how SoftBoard integrates the three stages of the StartFlow method into a continuous workflow: guided requirements structuring, wireflow generation and refinement, and automatic UX evaluation. This workflow highlights the tool’s potential to reduce the cognitive burden of prototyping, standardize the application of the method, and support teams with limited UX expertise in developing MVPs that are better aligned with user needs.

Additionally, the tool allows users to generate a public link for viewing completed boards, facilitating stakeholder communication and the sharing of prototyping results.

5 Conclusion and Future Work

This paper presented SoftBoard, a web-based tool designed to operationalize the StartFlow method and support the creation and evaluation of low-fidelity prototypes in software startups. The motivation for this work stems from the observation that, although User Experience is widely recognized as a critical factor for the success of digital products, startup teams often face constraints related to time, resources, and expertise that hinder the systematic adoption of structured UX practices during the early stages of development.

The tool was conceived based on findings from a previously conducted Systematic Mapping Study, which revealed important gaps in the ecosystem of low-fidelity prototyping tools. The main findings included the predominance of general-purpose tools adapted for prototyping, the limited incorporation of explicit methodological guidance, and the still incipient use of Artificial Intelligence as continuous support throughout the design process.

SoftBoard was developed to address these gaps by integrating, within a single environment, the StartFlow method, a dedicated wireflow editor, and an LLM-based multi-agent system that supports the entire prototyping process. This integration transforms a method originally supported only by textual documentation into an interactive and guided workflow, reducing application variability and dependence on practitioners’ individual experience.

The results presented indicate that the tool contributes in three main ways. First, it promotes the explicit incorporation of methodological guidance by structuring activities into sequential stages aligned with StartFlow. Second, it introduces Artificial Intelligence as continuous support throughout the process through agents that assist with requirements elicitation, automatic wireflow generation, and heuristic-based UX evaluation. Third, it seeks to lower the entry barrier for teams with limited UX expertise by providing a guided workflow that supports structured decision-making from the earliest stages of product development.

As future work, a feasibility study of the tool is currently underway, involving usability testing and interviews with software professionals in real startup contexts. The results of this study are expected to provide insights into practitioners’ perceptions, identify opportunities for improvement, and offer initial evidence regarding the adoption potential of SoftBoard in the MVP development process.

Artifact Availability

All artifacts, including the demonstration video and diagrams, are available at: <https://doi.org/10.5281/zenodo.20369601>.

References

- [1] Chadia Abras, Diane Maloney-Krichmar, Jenny Preece, et al. 2004. User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction*. Thousand Oaks: Sage Publications 37, 4 (2004), 445–456.
- [2] Silvio Alonso, Marcos Kalinowski, Marx Viana, Bruna Ferreira, and Simone D.J. Barbosa. 2021. A Systematic Mapping Study on the Use of Software Engineering Practices to Develop MVPs. In *2021 47th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, Palermo, Italy, 62–69. doi:10.1109/SEAA53835.2021.00017

- [3] Carmelo Ardito, Paolo Buono, Danilo Caivano, Maria Francesca Costabile, and Rosa Lanzilotti. 2014. Investigating and promoting UX practice in industry: An experimental study. *International Journal of Human-Computer Studies* 72, 6 (2014), 542–551. doi:10.1016/j.ijhcs.2013.10.004 Interplay between User Experience Evaluation and System Development.
- [4] Jas Brooks and Pedro Lopes. 2023. Smell & Paste: Low-Fidelity Prototyping for Olfactory Experiences. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 1–16.
- [5] Lennon Chaves, Márcia Lima, and Tayana Conte. 2025. Solaria-GPT: A Tailored ChatGPT Tool for Usability Inspection. In *Anais do XXXIX Simpósio Brasileiro de Engenharia de Software (Recife/PE)*. SBC, Porto Alegre, RS, Brasil, 956–962. doi:10.5753/sbes.2025.11455
- [6] Adrien Coyette, Suzanne Kieffer, and Jean Vanderdonck. 2007. Multi-fidelity Prototyping of User Interfaces. In *Human-Computer Interaction – INTERACT 2007*, Cécilia Baranauskas, Philippe Palanque, Julio Abascal, and Simone Diniz Junqueira Barbosa (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 150–164.
- [7] Gabriel Resende da Silva Scapim, Gislaine Camila Lapasini Leal, and Guilherme Corredato Guerino. 2025. New Perspectives on Low-Fidelity Prototyping: A Systematic Mapping Study on Tools and Functionalities. In *Anais do XXIV Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais (Belo Horizonte/MG)*. SBC, Porto Alegre, RS, Brasil, 1565–1588. doi:10.5753/ihc.2025.10912
- [8] Carmine Giardino, Michael Unterkalmsteiner, Nicolò Paternoster, Tony Gorschek, and Pekka Abrahamsson. 2014. What Do We Know about Software Development in Startups? *IEEE Software* 31, 5 (2014), 28–32. doi:10.1109/MS.2014.129
- [9] Natã Lael Gomes Raulino, Rossana Maria de Castro Andrade, and Ismayle de Sousa Santos. 2024. Integration of User-Centered Design in the Development of Big Data and Machine Learning-Based Applications: A Systematic Mapping Study. In *Proceedings of the XXIII Brazilian Symposium on Human Factors in Computing Systems (IHC '24)*. Association for Computing Machinery, New York, NY, USA, Article 59, 12 pages. doi:10.1145/3702038.3702097
- [10] Guilherme Guerino, Suellen Martinelli, Joelma Choma, Gislaine Camila Leal, Renato Balancieri, and Luciana Zaina. 2024. Investigating UX work in Software Startups: A Survey about Attitudes, Methods, and Key Challenges. *Journal of the Brazilian Computer Society* 30, 1 (Dec. 2024), 621–638. doi:10.5753/jbcs.2024.4260
- [11] Guilherme Corredato Guerino, João Pedro de Souza Olivo Tardivo, Renato Balancieri, and Gislaine Camila Lapasini Leal. 2026. StartFlow: From method conception to multi-perspective evaluation in UX prototyping for software startups. *Information and Software Technology* (2026), 108193.
- [12] Laura Hokkanen, Yueqiang Xu, and Kaisa Väänänen. 2016. Focusing on User Experience and Business Models in Startups: Investigation of Two-Dimensional Value Creation. In *Proceedings of the 20th International Academic Mindtrek Conference (Tampere, Finland) (AcademicMindtrek '16)*. Association for Computing Machinery, New York, NY, USA, 59–67. doi:10.1145/2994310.2994371
- [13] Christopher D. Hundhausen, Anzor Balkar, Mohamed Nuur, and Stephen Trent. 2007. WOZ pro: a pen-based low fidelity prototyping environment to support wizard of oz studies. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems (San Jose, CA, USA) (CHI EA '07)*. Association for Computing Machinery, New York, NY, USA, 2453–2458. doi:10.1145/1240866.1241023
- [14] International Organization for Standardization. 2019. *Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems*. ISO, Geneva, Switzerland. Edition 2.
- [15] Taghi Javdani Gandomani, Hazura Zulzalil, and Rami Bahsoon. 2024. Empowering software startups with agile methods and practices: A design science research. *Software: Practice and Experience* 55 (09 2024), 220–242. doi:10.1002/spe.3371
- [16] Valentina Lenarduzzi and Davide Taibi. 2016. MVP Explained: A Systematic Mapping Study on the Definitions of Minimal Viable Product. In *2016 42th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, Limassol, Cyprus, 112–119. doi:10.1109/SEAA.2016.56
- [17] Junchen Li, Garreth W. Tigwell, and Kristen Shinohara. 2021. Accessibility of High-Fidelity Prototyping Tools. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (Yokohama, Japan) (CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 493, 17 pages. doi:10.1145/3411764.3445520
- [18] Gerardo Matturro, Gonzalo Nieto, Alfonso González, and Martín Solari. 2021. Minimum Viable Product Creation and Validation in Software Startups. In *2021 XLVII Latin American Computing Conference (CLEI)*. Institute of Electrical and Electronics Engineers (IEEE), Cartago, Costa Rica, 1–8. doi:10.1109/CLEI53233.2021.9639942
- [19] Michael McCurdy, Christopher Connors, Guy Pyrzak, Bob Kanefsky, and Alonso Vera. 2006. Breaking the fidelity barrier: an examination of our current characterization of prototypes and an example of a mixed-fidelity success. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Montréal, Québec, Canada) (CHI '06)*. Association for Computing Machinery, New York, NY, USA, 1233–1242. doi:10.1145/1124772.1124959
- [20] Anh Nguyen-Duc, Xiaofeng Wang, and Pekka Abrahamsson. 2017. What Influences the Speed of Prototyping? An Empirical Investigation of Twenty Software Startups. In *Agile Processes in Software Engineering and Extreme Programming*, Hubert Baumeister, Horst Lichter, and Matthias Riebisch (Eds.). Springer International Publishing, Cham, 20–36.
- [21] Jakob Nielsen. 1994. Heuristic evaluation. In *Usability Inspection Methods*. John Wiley & Sons, New York, NY, 25–62.
- [22] Ramon Pereira, Andreia Malucelli, and Sheila Reinehr. 2025. From Real-Time Conversation to User Story: Leveraging Agile Requirements through LLM. In *Anais do XXXIX Simpósio Brasileiro de Engenharia de Software (Recife/PE)*. SBC, Porto Alegre, RS, Brasil, 657–663. doi:10.5753/sbes.2025.10988
- [23] Eric Ries. 2011. *The lean startup : how constant innovation creates radically successful businesses*. Portfolio Penguin, London; New York. http://www.amazon.de/The-Lean-Startup-Innovation-Successful/dp/0670921602/ref=sr_1_2?ie=UTF8&qid=1396199893&sr=8-2&keywords=eric+ries
- [24] Karoline Romero, Lucas Beluomini, Juliana Scanavacca, Isabella Pazinato, Renato Balancieri, Gislaine Leal, and Guilherme Guerino. 2024. Use of Design Thinking for the Design and Evaluation of a Mobile Nutrition Information Application. In *Proceedings of the XXIII Brazilian Symposium on Human Factors in Computing Systems (IHC '24)*. Association for Computing Machinery, New York, NY, USA, Article 15, 14 pages. doi:10.1145/3702038.3702053