

Evaluating Interface Consistency and User Control in Fog and Edge Computing Dashboards: A Heuristic Analysis of Open-Source Platforms

Seung Hwan Kang^{1*} and Jira Yammeesri²

¹seung_h@payap.ac.th and ²jira_y@payap.ac.th

^{1, 2}The International College, Payap University, Chiang Mai

Received: June 24, 2025; Revised: July 1, 2025; Accepted: August 5, 2025; Published: December 27, 2025

Abstract – Fog and edge computing platforms increasingly become necessary for the management of distributed Internet of Things services, but their management dashboards are not typically very friendly to users. This study discusses a heuristic analysis of two most widely used open-source management dashboards, EdgeX Foundry and FogFlow, in terms of five usability guidelines: consistency and standards, user control and freedom, visibility of system status, error prevention, and help and documentation. Nineteen usability issues were identified and rated according to Nielsen's heuristic model through task-based, expert-led walkthroughs. Major issues included non-consistent terminology, a lack of undo or confirm features for critical actions, inadequate real-time feedback, and a lack of embedded help facilities. These issues reduce operation effectiveness and increase the likelihood of user error, most notably in time-sensitive or high-risk scenarios. The study proposes certain recommendations such as standardizing the components of the interface, improving the options provided to the users for control, improving system feedback, and incorporating contextual help. The incorporation of human-centered design principles during the creation of fog and edge dashboards will significantly enhance usability, enable better decision-making, and enable easier use in real-world implementations.

KEY WORDS: Usability Evaluation, Dashboard Interface Design, Heuristic Analysis, Fog Computing, Edge Computing, Distributed Computing

1. Background and Significance

Fog and edge computing have shown themselves to be vital distributed computing paradigms of late, especially in Internet of Things (IoT) settings, because they enable low latency processing and more efficient resource utilization close to data sources. Unlike traditional cloud computing models that see storage and data processing take place largely in main data centers, fog and edge computing bring computational capacities nearer to where data is produced, either at the network's edge or regions nearest to it. This approach builds upon some of the underlying challenges of latency reduction, bandwidth optimization, and privacy enhancement of data networks. These are critical in autonomous vehicles, smart grids, real-time industrial automation, and smart city infrastructure.

Fog computing, envisioned by Bonomi et al. [1], provides a hierarchical architecture that adds local processing capability through

intermediate nodes in addition to cloud capabilities. Edge computing builds on this by providing the ability to perform computation on devices or on immediate gateways, reducing reliance on distant data centers. Together, these paradigms provide more scalable and responsive IoT solutions.

Effective control and monitoring are required to operate such distributed and dynamic systems. Administrative dashboards are the chief user interfaces through which system operators and administrators can control device status, install services, respond to alarms, and maintain the system healthy. How easily accessible these dashboards are will directly impact on the safety, accuracy, and efficiency of system administration tasks. In manufacturing facilities, for example, a well-designed dashboard may allow for prompt response to failures to minimize downtime and prevent cascading failure.

Despite the crucial contribution of fog and edge computing, research was limited to a

theoretical basis for judging usability for associated dashboards. The limited usability studies conducted have largely focused on application domains such as healthcare [14], industrial Supervisory Control and Data Acquisition (SCADA) systems [11], and data analytics environments [13]. These investigations pin down common usability issues, including inconsistent user interface design, insufficient prevention of errors, and inadequate feedback mechanisms. These factors can sharply degrade operator performance and system reliability.

The growing adoption of fog and edge computing platforms in mission-critical environments underlines the need for making use of rigorous usability evaluation methods for such systems. Poor usability in such interfaces is not likely to cause operator error but also undermines the large-scale deployment and feasibility of IoT integration. This study aims to fill this research gap by conducting a heuristic evaluation of two of the most popular open-source fog and edge computing dashboards with focus on significant usability factors that affect system management.

2. Objectives

This study evaluates administrative dashboard usability within fog and edge computing systems in terms of heuristic design principles previously applied. The evaluation is aimed at five factors: consistency in the interface, user control, visibility of system status, prevention of errors, and provision of help and documentation.

The aim is to uncover common usability challenges in these areas and turn the findings into practical design recommendations. Making dashboards easier to use can lighten the load on system operators, reduce the chances of mistakes, and improve the overall efficiency of managing the system.

Additionally, this research fills an important literature gap by applying a robust heuristic evaluation framework to an area where usability is frequently overlooked. Results are intended to support developers, researchers, and system integrators in promoting more user-centered ways of designing distributed computing interfaces.

3. Hypothesis

The main hypothesis for this study is that administrative dashboards of top fog and edge

computing systems suffer from serious usability issues that complicate system management. In generally, it is predicted that these user interfaces will exhibit infractions of core usability rules, including inconsistencies in design language, ineffective mechanisms for guiding users, insufficient system status visibility, inadequate measures to prevent errors, and lack of accessible help tools.

This hypothesis is supported by previous findings in related fields, such as SCADA systems [1] and medical dashboards [15], where similar interface shortcomings have been linked with high cognitive load and procedural errors. Considering the relative novelty and technological complexity of fog and edge computing infrastructures, and alongside the frequently open-source development environment, it is reasonable to expect accompanying or even elevated levels of usability problems.

Moreover, such platforms are commonly found in mission-critical and time-sensitive environments where usability issues can lead to serious operational risks. Therefore, testing this hypothesis will highlight some design flaws as well as the need to include human-centered design principles at an early stage in the development of fog and edge computing interfaces.

4. Methodology

The usability issues with fog and edge computing dashboards were evaluated in this study using an organized methodology. To identify recurrent design issues, the assessment focused on a qualitative expert review using a commonly accepted usability inspection approach. The techniques employed are described in depth in the following subsections.

4.1 Heuristic Evaluation Approach

This study utilizes heuristic evaluation as its primary methodological approach. Introduced by Nielsen and Molich [6], heuristic evaluation is a well-established usability inspection method in which a small group of experts systematically examines an interface based on a set of predefined usability principles, or heuristics. The approach has been recognized for its ability to identify usability issues quickly and cost-effectively, making it particularly useful for evaluating complex technical systems prior to more extensive user testing.

4.2 Selection of Heuristics

Using Nielsen's usability heuristics as a guide [5], this study focused on five key areas that are particularly relevant to fog and edge computing dashboards. These included keeping the interface consistent to reduce confusion, giving users enough control to undo actions or move around freely, making system status clear and easy to follow, designing in ways that help prevent mistakes before they happen, and offering built-in help or guidance when needed. These principles were chosen because they play a major role in how effectively system administrators can interact with complex platforms and manage their cognitive workload.

4.3 Platforms Under Review

This research compares two widely used open-source systems: FogFlow and EdgeX Foundry [9][10]. They were selected because they are used in the fog and edge computing and IoT communities. They are also representative design patterns one finds when dealing with administrative dashboards used for distributed system administration. EdgeX Foundry claims modular design with interoperability and integration ease that allows administrators to monitor device state and streams of data effectively. In contrast to this, FogFlow is designed for orchestration in edge environments and provides real-time monitoring and control features. The differences in system design and user interface design of these two platforms make them excellent candidates for comparative usability testing.

4.4 Evaluation Procedure

To systematically uncover usability issues, the study employed a heuristic evaluation carried out by two experts in human-computer interaction who had prior experience assessing complex user interfaces. The evaluation began with a familiarization stage, during which the experts explored key features of each dashboard and typical workflows followed by system administrators. The experts then walked through based on common administrative operations like creating devices, deploying services, and handling alerts.

In addition to these general workflows, the evaluators performed a set of predefined tasks: device registration, service deployment, configuration of data pipelines, reviewing alerts and notifications, and log inspection. These are

tasks considered important administrator activities that are common in operational fog and edge computing environments. For each of these tasks, the evaluators recorded issues found on the interface, the time taken to perform the activity, clarity and immediacy of system feedback as the test subjects went through each step, and any navigation difficulties or ambiguities in terminology.

Any usability issues that were generated from performing such actions were detailed, rated for severity using Nielsen's zero to four scale.

Table 1. Nielsen's Usability Severity Rating Scale.

Rating	Label	Description
0	Not a usability problem	No negative impact on usability; no corrective action needed.
1	Cosmetic problem only	Minor issue; fix is optional and typically addressed only if time allows
2	Minor usability problem	Fix is useful but not urgent; may be considered if it affects many users
3	Major usability problem	Fix is important; significantly disrupts the user experience
4	Usability catastrophe	Must be resolved immediately; severely impacts system usability

To help assess the level of agreement between the two evaluators, a simplified inter-rater reliability check was performed. The ratio of agreement, based on a comparison of the severity judgments for all identified issues, is roughly 85 percent. The quick estimate of Cohen's Kappa, via the formula, is as follows:

$$\kappa = (P_o - P_e) / (1 - P_e)$$

where P_o represents the observed agreement (approximately 0.85) and P_e represents the expected agreement. Using these values yields an estimated $\kappa \approx 0.69$, which is generally interpreted a high degree of agreement. Although not a complete statistical assessment of reliability, a degree of agreement is useful correlating to enhanced overall evaluation consistency.

Although the evaluation was carried out by two experts, it is rigorous with respect to documentation, following guidelines on conducting a heuristic evaluation. In fact, in the future, having multiple people conduct evaluation tasks can increase reliability through triangulation.

4.5 Limitations

While heuristic evaluation is a well-established approach for identifying usability issues during the early phases of system assessment, it cannot fully replace testing with actual users. Because this approach relied on expert analysis rather than direct observation or experimental data from system operators, certain usability problems tied to real-world usage, such as those influenced by operational pressure, complex workflows, or varying levels of user experience, may not have been fully captured.

In addition, the scope of the evaluation was limited to two representative fog and edge computing platforms. These systems were chosen for their relevance and for reflecting common architectural features, yet the findings may not extend to every platform in this category. Differences in how mature the software is, who the users are, and the specific environments in which the systems are used can all influence the kinds of usability issues that may arise.

In addition, the scope of assessment has been limited to two representative fog and edge computing platforms. The reason is that such systems are representative but may not necessarily generalize to all platforms categorized under fog and edge computing platforms. This is attributed to software development stages, user base, as well as the environment in which such platforms are used.

Despite these limitations, the study provides key findings regarding ubiquitous interface design issues in fog and edge environments. The resulting issues and proposed solutions are a starting point for follow-up research and design improvement, ideally based on empirical user testing, larger sampling of platforms, and longitudinal studies for identifying evolving trends in use.

5. Findings

This study conducted a heuristic evaluation of two fog and edge computing dashboards, namely EdgeX Foundry and FogFlow. The evaluation process identified

nineteen unique usability issues categorized under five established heuristic principles. These results indicate multiple interface deficiencies that may adversely affect system administration efficiency and operational safety.

Table II. Identified Usability Issues Based on Nielsen's Heuristics and Severity Ratings.

Heuristic	Usability Issue	Severity (0-4)
Consistency and Standards	1. Inconsistent use of terminology across modules	3
	2. Varied iconography for similar functions	2
	3. Inconsistent layout between dashboard sections	2
	4. Unclear or inconsistent labeling of navigation and action items	3
User Control and Freedom	5. No undo function for critical actions	4
	6. Missing confirmation dialogs for irreversible changes	4
	7. Lack of versioning or rollback functionality	3
Visibility of System Status	8. Delayed or missing feedback during key operations	3
	9. Absence of a centralized real-time system health overview	2
	10. Vague error messages lacking actionable information	3
	11. Insufficient feedback on background processes	2

Error Prevention	12. Lack of input validation for critical fields	4
	13. Absence of guided workflows during complex configurations	3
	14. No warnings for potentially harmful parameter settings	4
	15. Advanced options presented without sufficient context or explanations	2
Help and Documentation	16. No contextual help	2
	17. No onboarding guidance for first-time users	3
	18. Help materials located externally and not integrated into the interface	2
	19. Ineffective or absent search functionality in documentation	2

5.1 Consistency and Standards

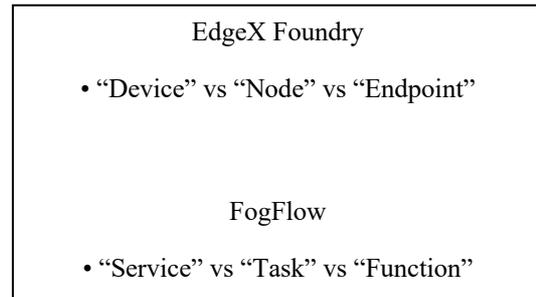
One of the frequent usability issues on both systems was the inconsistency of the visual organization and terminology. In EdgeX Foundry, for example, words like "device," "node," and "endpoint" were used interchangeably but without any background information or definition. Such inconsistency is liable to mislead users and makes it difficult for users to know how the system is structured or how its components relate to one another.

Therefore, the FogFlow dashboard was also inconsistent in its menu organization and button placement between modules. Inconsistency in navigation behaviors resulted in the fact that users would need to acquire new control patterns every time they shifted views,

potentially adding more cognitive effort as well as possibility of error in operation.

Figure 1. Inconsistent Terminology

5.2 User Control and Freedom



There were no adequate controls on either dashboard to reverse a motion or verify possibly harmful actions. For example, EdgeX Foundry increased the risk of inadvertent data loss by removing a confirmation popup when users erased a device or service. FogFlow also lacked an undo mechanism for config changes, making it hard for the user to simply recover from errors.

Moreover, navigation paths were occasionally rigid with little provision for backtracking or crossing parallel paths without the need to restart operations. This constraint made error recovery difficult and decreased confidence from the user during complex maneuvers.

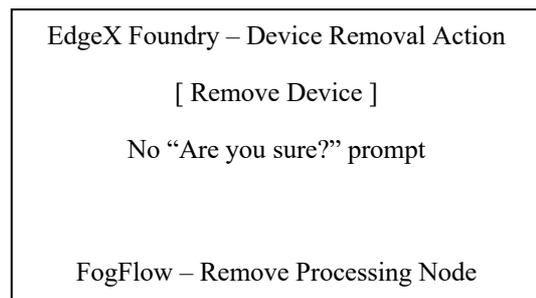


Figure 2. Missing Confirmation Dialogs for Critical Operations.

5.3 Visibility of System Status

Immediate and direct feedback on system status is critical in distributed computing resource management. Both dashboards were lacking it. EdgeX Foundry had very little real-time feedback on services available and devices online, and multiple manual refreshes or visiting specific logs had to be performed to establish system well-being. Such a delay in feedback can

introduce uncertainty and delayed response in critical scenarios.

FogFlow's alerting was minimalist and sometimes infrequent, failing to adequately notify users of important events such as node failure or job completion. Actors had incomplete system state awareness, which compromises situational awareness.

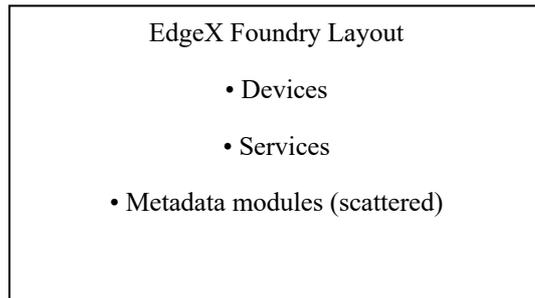


Figure 3. Lack of Real-Time System Status Visibility.

5.4 Error Prevention

Preventing errors before they occur is a basic design goal that was not discussed adequately. Neither platform integrated form validation consistently. For example, EdgeX Foundry allowed for filling in or submitting partially null or inaccurately filled-out device registration data, which instigated backend faults difficult to interpret or correct by users.

FogFlow also lacked guided workflows or inline help that could steer users away from errors. Failure to have proactive error prevention can result in increased levels of operational faults and maintenance overhead.

5.5 Help and Documentation

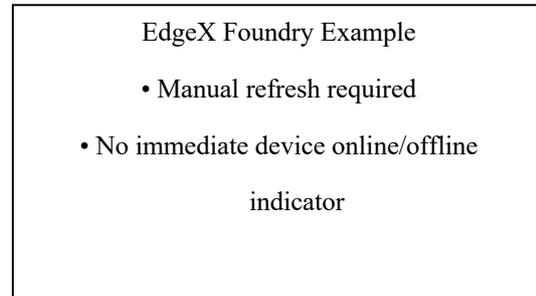
Both dashboards lacked ready documentation and support. Internal help was light, and most support consisted of separate, static reference websites. This required customers to leave the dashboard just to switch settings or debug problems, disrupting their workflow and making routine processes even more frustrating than necessary.

This disconnects between the dashboard and help resources disrupts workflow and can lead to frustration, particularly for new or less experienced administrators.

In summary, the findings disclose key usability shortcomings that need to be addressed to make fog and edge computing dashboards more

user-friendly and dependable for everyday operational use.

Figure 4. Comparison of Dashboard Structures.



6. Discussion

The heuristic evaluation results identify significant usability flaws in fog and edge computing dashboards to confirm the primary hypothesis that such interfaces lack elementary human-centered design principles. These shortcomings have long-term implications on the efficiency, accuracy, and overall user experience of administrators overseeing distributed IoT systems.

6.1 Terminology and Inconsistencies Issues

Terminological and graphical design inconsistencies undermine users' ability to form stable mental models, essential for effective decision-making and error avoidance in complex systems [7]. As dashboards utilize interchangeable or vague vocabulary, users put cognitive effort into only interpreting interface language instead of task goals. This slows down processes and potentially increases chances for misinterpretation, especially where time pressures typical of real-world deployment apply.

These findings are in alignment with previous research conducted in industrial SCADA and clinical dashboards, when it was identified that standardization of vocabulary was essential to avoid confusion and improve learnability [11] [15]. For fog and edge platforms, having a standardized vocabulary and visual language becomes even more important given the technical heterogeneity of IoT devices and services.

6.2 The Critical Role of User Control

Lack of undo function and confirm dialogs is a serious usability risk. In dynamic systems where configurational changes have

extensive effects, operators should be allowed to experiment and correct errors freely without endangering irreversible harm. The absence of such facilities not only introduces stress but can lead to operational errors with costly repercussions.

This reflects usability principles in general terms across HCI literature [5][12], but appears under-discussed in fog and edge computing platforms, feasibly because the technical focus of these tools deflects from usability concerns. More robust user control mechanisms need to be emphasized to empower operators and reduce errors.

6.3 Visibility of System Status and Real-Time Feedback

The two systems did not support evident and prompt system status indicators, and this undermines situational awareness. This awareness is a strong predictor of effective monitoring and swift decision-making. [3]. Without the instant feedback, the users are never certain if an operation was performed effectively or devices were functioning properly. Such ambiguity can delay important actions and lead to misconceptions about the status of the system.

Real-time feedback has been documented as a key component of usable system design, particularly in cases where continuous attention is demanded [4]. The study indicates that fog and edge dashboard developers to improve notice mechanisms and to incorporate automated status updates more effectively.

6.4 Error Prevention in Practice

The evaluation showed a tendency to focus on error detection rather than proactive error prevention. Modern usability models focus on the development of interfaces that prevent users from making errors in the first place [8]. These include form validations, guided workflows, and contextual suggestions.

Due to the technical complexity and mission-critical nature of fog and edge computing systems, interfaces should proactively assist users by validating inputs and warning users about potential conflicts or misconfigurations before changes are submitted. This preventative design reduces operational risks and eases training for new users.

6.5 Help and Documentation Integration

The disperse and poorly integrated help materials found in both systems disrupt workflow and reduce usability, particularly among new users to both sites. The embedding of contextual, accessible help within dashboards can significantly increase user confidence and productivity, as documented in usability best practices [2].

Use of static documentation instead of embedded support may be a side effect of the open-source nature of these projects with limited resources for designing user experience. However, as usage grows, it will be important to prioritize using embedded help to allow for larger user bases.

In summary, this study applies heuristic assessment to fog and edge computing dashboards, a crucial component of contemporary IoT systems, occupying a largely unexplored area in usability research. The findings suggest that most usability issues encountered in industrial and clinical dashboards are applicable to these systems as well, with their own set of challenges in the environment of fog and edge computing.

Future research should build upon these results by including actual operators in user testing, testing a more diverse range of platforms, and exploring interface designs that adapt to the experience level of the user.

These findings also indicate that the fog and edge computing community has a need overall for increased support for usability. The improvement of dashboard design cannot be limited to merely meeting the performance requirements but should continue to emphasize human-centered development practices. The open-source community, designers, and researchers should increase collaboration to optimize design standards, enhance the development integration for usability, as well as validate the design with end-users. This is essential for making such platforms reliable, user-friendly, and more usable, especially given the increased importance of fog computing platforms within IoT infrastructure.

7. Conclusion

Even though this assessment has identified a number of usability problems within the two platforms used, the assessment results need to be considered as preliminary, as they are not a conclusive determination concerning all fog and edge computing dashboard interfaces. As

edge and fog computing infrastructures become more central to distributed IoT management, the usability of their control panels becomes critical. This work found extensive usability shortcomings through a rigorous heuristic evaluation, identifying significant issues ranging from lack of consistent terminology, minimal user control features, lack of immediate feedback, and lack of integrated help facilities. In sophisticated IoT environments, such vulnerabilities erode the efficiency, security, and well-being of the administrators.

To address these gaps, there must be a shared vocabulary for the interface and a shared look and feel to support such systems. This approach enhances clarity and reduces confusion for users. An extension of control features such as undo and redo actions, confirmation prompts, and audit trails can lead users to make aggressive actions without fear of making irreversible mistakes. In addition, greater visibility of system state through clearly visible alerts and readily accessible notifications enhances situational awareness and timely decision-making. As similarly useful, contextual assistance features in dashboards, such as tooltips and guided tours, can also support user independence and allow new users to get settled in earlier. Finally, employing preventive usability practices like input validation and context-sensitive tips can avoid errors from happening at all, and therefore the system is stable and manageable.

By incorporating these human-centered design principles, fog and edge dashboards may better support the demands of distributed IoT infrastructures by improving security, productivity, and usability. Looking ahead, future research should build on these findings by conducting empirical user studies, developing adaptive interface prototypes for both novice and experienced users, and evaluating a broader range of open-source and commercial platforms.

References

- [1] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proc. First Edition MCC Workshop on Mobile Cloud Computing*, Helsinki, Finland, 2012, pp. 13-16.
- [2] J. Carroll, Human-computer interaction: Psychology as a science of design, *Annual Review of Psychology*, vol. 48, no. 1, pp. 61-83, 1997.
- [3] M. R. Endsley, "Toward a theory of situation awareness in dynamic systems," *Human Factors*, vol. 37, no. 1, pp. 32-64, 1995.
- [4] T. Bach, L. Weber, and M. Li, "Patterns and principles of dashboard design: A survey of 144 systems," *IEEE Trans. Vis. Comput. Graphics*, vol. 28, no. 7, pp. 2835-2848, July 2022.
- [5] J. Nielsen, *Usability Engineering*. San Francisco, CA, USA: Morgan Kaufmann, 1994.
- [6] J. Nielsen and R. Molich, "Heuristic evaluation of user interfaces," in *Proc. ACM CHI Conf. Human Factors Comput. Syst.*, Seattle, WA, USA, 1990, pp. 249-256.
- [7] D. A. Norman, *The Psychology of Everyday Things*. New York, NY, USA: Basic Books, 1988.
- [8] D. A. Norman, *The Design of Everyday Things: Revised and Expanded Edition*. New York, NY, USA: Basic Books, 2013.
- [9] OpenFog Consortium, "FogFlow User Guide," 2024. [Online]. Available: <https://www.openfog.org/fogflow>. [Accessed: Jan. 18, 2025].
- [10] EdgeX Foundry, "EdgeX Foundry Documentation," EdgeX Foundry, 2024. [Online]. Available: <https://www.edgexfoundry.org/docs>. [Accessed: Jan. 18, 2025].
- [11] J. Pham, T. Nguyen, and M. Tran, "Usability issues in SCADA interfaces: Heuristic and salience model evaluation," *Int. J. Hum.-Comput. Stud.*, vol. 164, May 2023, Art. no. 102876.
- [12] B. Shneiderman, C. Plaisant, and M. Cohen, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, 6th ed., Boston, MA, USA: Addison-Wesley, 2016.
- [13] S. Setlur, A. Karapanos, and A. Kapoor, "Design heuristics for interactive dashboards: A cooperative analytics perspective," *ACM Trans. Comput.-Hum. Interact.*, vol. 30, no. 2, pp. 1-37, April 2023.
- [14] J. Cram, R. H. Wilson, and P. G. Thompson, "User control and visual clarity in clinical dashboards: A heuristic assessment," *J. Healthcare Informatics Res.*, vol. 3, no. 3, pp. 261-274, 2019.
- [15] M. Vilarinho, C. Silva, and A. Oliveira, "Heuristic evaluation of patient-reported outcome dashboards: Insights for healthcare UX design," *J. Med. Syst.*, vol. 46, no. 4, pp. 1-15, April 2022.