Human Factors in Design, Engineering, and Computing, Vol. 159, 2024, 935–947
https://doi.org/10.54941/ahfe1005660

An Approach to System Architecture Design Through Usability Heuristics

Preston Myers and Duha Ali

California Polytechnic State University – San Luis Obispo, San Luis Obispo CA 93407, USA

ABSTRACT

This research explores applying usability heuristics in system architecture design to address the lack of clear guidelines in systems engineering. It emphasizes integrating human factors and usability principles to create robust, user-friendly system architectures. The study employs two primary usability testing approaches: timed performance assessments and A/B testing. Timed performance assessments involved 16 participants with STEM backgrounds evaluating their ability to locate and interpret information from two different autonomous vehicle system architectures. Results showed that participants identified elements faster in the redesigned architecture with fewer line intersections and clearer node organization. A/B testing compared a patented aircraft avionics system architecture with a new design. Participants provided feedback on understanding, satisfaction, and perceived complexity. The newly designed architecture received higher satisfaction ratings and was easier to understand, emphasizing organized node grouping and clear information flow. The study developed a set of usability heuristics, including minimizing arrow intersections, using straight lines, clear labeling, ensuring information flow, reducing noise, promoting flexibility, and providing keys for abbreviations and symbols. These guidelines aim to help system architects create intuitive and efficient designs.

Keywords: Human systems integration, Systems engineering, Usability, User experience

INTRODUCTION

Systems engineering has evolved into a distinct scientific field (Keating et al., 2003). Unlike other engineering disciplines, it is not governed by fundamental laws but focuses on managing complex engineering problems (Niamat Ullah Ibne Hossain, 2021). Learning systems engineering offers a significant return on investment, as it provides valuable tools applicable across industries (Schatz, 2023). Studies have shown that projects with higher systems engineering efforts have higher success rates, up to 80% (INCOSE, 2023). The field has expanded across various industries, with Model-Based Systems Engineering (MBSE) gaining prominence due to technological advancements (Nataliya Shevchenko, 2020). NASA emphasized MBSE's importance for tracking system complexity (NASA, 2016). MBSE supports requirement traceability, system architecture, performance analysis, verification, and validation (Hart, 2015). Effective system architecture is crucial for business success, requiring clarity, detail, and consistency (Amanda McGrath &

Alexandra Jonker, 2023). Characteristics of a good architecture include robustness, feasibility, usability, durability, traceability, and elegance (Sparx Systems, 2023). However, guidelines for creating strong system architectures are lacking, leaving architects with a challenging task (Peter Brook, 2024). Maier and Rechtin's "The Art of Systems Architecting" highlights the balance between the scientific and artistic approaches to systems architecture (Maier et al., 2009). Heuristics, or problem-solving shortcuts, play a vital role in developing complex systems (Kendra Cherry, 2022). Despite their abstract nature, heuristics offer strategies and guidelines for decision-making (Menshenin et al., 2022). A focus on human factors in system design could enhance architecture development (Sanders & McCormick, 1993). Usability heuristics, such as Jakob Nielsen's ten principles, measure user interface experience and can improve system architecture by making it more user-friendly and efficient (Felipe Guimaraes, 2022). This paper proposes creating a detailed list of heuristics based on usability principles to guide system architects in designing effective system architectures. Addressing human factors and usability can lead to foundational success, efficiency, and performance for businesses in a rapidly evolving world.

METHODOLOGY

To develop systems architecture usability heuristics, usability testing was conducted on two systems with differently designed architectures. Contrasting architectures of each system were created by the investigator to facilitate the testing methodology and were evaluated based on selected heuristics. Sixteen participants, eight for each test, were evaluated. The eligibility criteria were: 1) a bachelor's degree or higher, 2) an academic background in a STEM field, 3) basic knowledge of design concepts, and 4) inexperience in systems engineering design. The testing methods used were a Timed Performance Assessment and A/B Testing.

Timed Performance Testing

For the timed assessment, a system for autonomous vehicle active safety features was used. This architecture is represented in Figure 2.

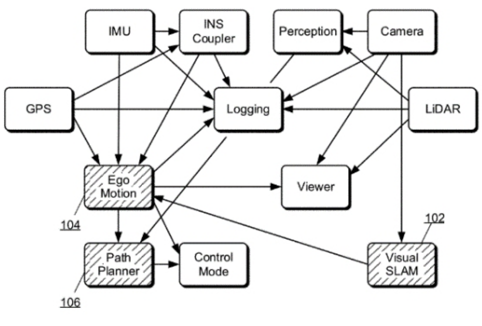

Figure 2: Systems architecture. (Vay Technology Patent US10397019B2).

To gather conclusive evidence and data for usability, a second architecture was designed to emulate this architecture in a different manner. This architecture is represented below in Figure 3.

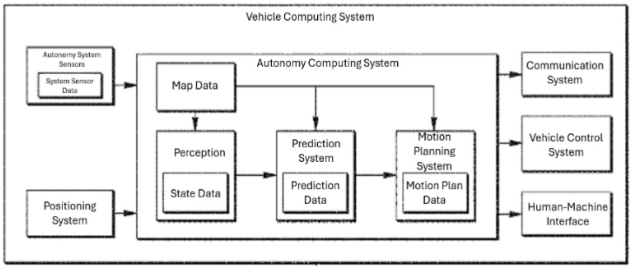

Figure 3: Designed system for autonomous vehicle active safety features.

For the timed performance assessment, the participants were asked to complete a timed assessment, to measure time and errors, and a follow up survey to assess their ideas and thoughts about the observed system architectures. The quantitative questions pertaining to the timed assessment can be seen in appendices A and B and the list of qualitative questions seen in appendices C and D were created to gain insight from users on the usability of contrasting architectures for A/B Testing. Comprehensive quantitative and qualitative data from both tests should allow for a complete and more detailed list of heuristics to be curated.

A/B Testing

A/B Testing is a controlled experiment used to compare two or more variations of a product with the goal of determining which product performs better (Gallo, 2017). For this test, the participants were asked to complete a qualitative survey based on the systems architectures of an aircraft avionics system. The first architecture, Figure 4, is a patent architecture from Pilatus Aircraft.

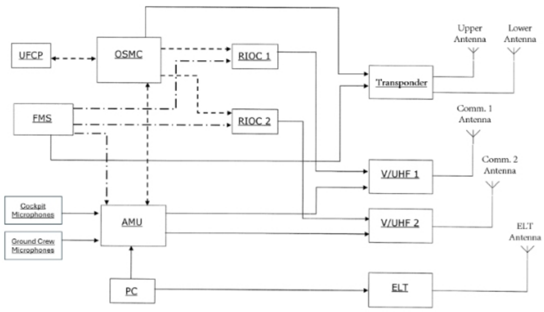

Figure 4: System architecture. (Pilatus Aircraft Patent US8538603B2).

The designed architecture can be seen below, in Figure 5.

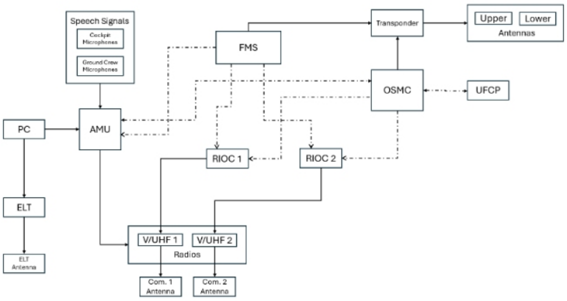

Figure 5: Designed system for aircraft avionics system.

For the purposes of A/B testing, these system architectures (Figures 4 and 5) were designed with the exact same components. The design considerations intended to draw conclusions include system element organization, line intersections, grouped elements, abbreviations, differing lines (solid vs dashed), and differently labeled symbols, such as the antennas. In the common modeling languages, there is no discrete symbol for an antenna, but an antenna icon has the potential to be more intuitive to inexperienced users. In UML and SysML a dashed line represents a dependency whereas a solid line represents an association. A dependency relationship is weaker than an association and shows how a change in one element might alter other elements. In simple terms, it shows the use of another element (IBM 2023).

The list of qualitative questions presented in appendices E and F was created to gain insight from users on the usability of contrasting architectures for A/B Testing. With the answers and results from the tests, conclusions on specific design heuristics for systems architectures can be created, as shown in the sections to follow.

RESULTS AND ANALYSIS

By the design of experiment based on specific system architecture heuristics, the usability tests of system architectures were performed. In understanding the results and analysis, the results from the Timed Performance Assessment will be discussed first, followed by the results from A/B Testing.

Timed Performance Survey Results

The design of the system architectures (Figure 2 and Figure 3) differs in organization, the number of lines and intersections, and element labeling. Timing participants on their ability to extract information from these architectures helped develop heuristics related to architecture design. The timed performance assessment focused on the system architectures of autonomous vehicle active safety features. Figure 2, a US patent, has many intersecting lines and nodes in a crowded model, while Figure 3 was designed with fewer lines, intersections, and a different node organization. Quantitative questions for the timed performance assessment are in appendices A and B. Data comparison was done using statistical tests to evaluate the significance between sample means. If samples were normal, a two-sample t-Test was used; if not, a nonparametric Mann-Whitney test was applied. The questions tested users on model identifications: data inflows and outflows, node identification, and central component connections. The data is shown in Table 1.

Table 1. Timed performance assessment data results ($\alpha = 0.05$).

Model Identification	Figure 2 Mean (SD) in Seconds	Figure 3 Mean (SD) in Seconds	Test Performed	Result Test Value	P Value
Outflows	25.92 (9.92)	9.77 (3.44)	*t*-test	$t(6) = 3.08$	0.022*
Inflows					
Motion	32.87 (11.55)	21.47 (20.11)	Mann-Whitney	U = 22.00	0.312
Perception	7.51 (2.18)	13.46 (11.51)	Mann-Whitney	U = 15.00	0.470
Node Identification	11.51 (1.60)	9.42 (1.87)	*t*-test	$t(6) = 1.70$	0.141
Central Component Identification	12.77 (2.30)	8.67 (1.27)	Mann-Whitney	U = 25.00	0.061

The timed assessments revealed notable differences between the two system architectures. For outflows, participants identified elements faster in Figure 3 (mean 9.77 seconds, SD 3.44) than in Figure 2 (mean 25.92 seconds, SD 9.92), with a significant difference confirmed by a two-sample *t*-test, $t(6) = 3.08$, $p = 0.022$. For inflows, identification times showed no significant difference between the two architectures. The mean time for Figure 2 was 32.87 seconds (SD 11.55) and for Figure 3, it was 21.47 seconds (SD 20.11), with the Mann-Whitney U test indicating no significant difference, $U = 22.00$, $p = 0.312$. Similarly, for perception identification, Figure 2 had a mean time of 7.51 seconds (SD 2.18) and Figure 3 had 13.46 seconds (SD 11.51), with no significant difference, $U = 15.00$, $p = 0.470$. For node identification, the mean time was 11.51 seconds (SD 1.60) for Figure 2 and 9.42 seconds (SD 1.87) for Figure 3, with no significant difference found, $t(6) = 1.70$, $p = 0.141$. For identifying central components, Figure 2 had a mean time of 12.77 seconds (SD 2.30) compared to 8.67 seconds (SD 1.27) for Figure 3, with the Mann-Whitney U test suggesting a marginally non-significant difference, $U = 25.00$, $p = 0.061$.

In summary, these results highlight that the design and organization of system elements significantly impact usability. Participants were able to identify outflows more quickly in the architecture of Figure 3, and there was a trend towards faster identification of central components in this design as well. Although no significant differences were observed in inflows and node identification, the overall findings suggest that Figure 3's design might be more efficient in certain usability aspects.

In addition to the timed assessments, other design elements such as data flow and nested nodes were evaluated separately within each figure, as

Figure 2 did not include nested nodes. For Figure 2, data flows were analyzed by comparing single path data flow to multiple potential paths. For Figure 3, nested nodes were examined at the system level, encompassing the entire autonomous vehicle safety features, and at the data level, where individual elements were more distinct. According to UML standards, nested nodes contain other nodes (IBM, 2023). The goal was to determine which nested node structure was easier for participants to recognize. The results are presented in Table 2 below.

Table 2. Timed performance assessment separate figure data results ($\alpha = 0.05$).

Figure	Model Identification	Data	Performed- Test	Result Test Value	P Value	
2		1 Path Mean (SD)	2 Paths Mean (SD)			
	Data Flow Identification	24.44 (9.29)	34.45 (17.14)	t-Test	$t(6) = -1.03$	0.344
3		Nested Node Mean (SD)	Nested System Mean (SD)			
	Nested Node Identification	10.36 (2.15)	16.96 (8.67)	t-Test	$t(6) = -1.48$	0.190

For Figure 2, the analysis showed that the mean identification time for the single path was 24.44 seconds (SD = 9.29), while for multiple paths it was 34.45 seconds (SD = 17.14). A two sample t-test indicated no significant difference between these two conditions ($t(6) = -1.03$, p = 0.344), suggesting that the complexity of data flow paths did not significantly impact participant performance in this scenario. For Figure 3, the results indicated that the mean identification time for data level nested nodes was 10.36 seconds (SD = 2.15), while for system level nested nodes it was 16.96 seconds (SD = 8.67). Although a two sample t-test showed no significant difference between these conditions ($t(6) = -1.48$, p = 0.190), the lower mean identification time for data level nested nodes suggests a trend towards easier recognition at this level. These findings suggest that while the complexity of data flows and the use of nested nodes did not significantly impact usability in the given scenarios, there is a trend towards better performance with simpler data flows and more accessible nested node structures.

In addition to the quantitative assessments, qualitative data was collected to gain deeper insights into the participants' experiences and perceptions of the different system architectures. The questions for the qualitative portion of the timed performance can be seen in appendices C and D. The satisfaction of users with a product is important for any usability test. Satisfaction refers to users' comfort and positive attitudes towards the use of a system (Frøkjær et al., 2000). In this experiment, participants were asked how satisfied with the architecture they were. The satisfaction results can be seen below in Figure 6.

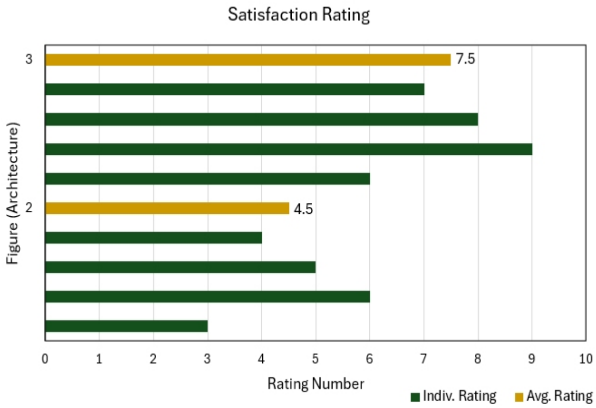

Figure 6: Timed performance architecture satisfaction rating.

The data provided in Figure 6 shows that Figure 2 received an average satisfaction rating of 4.5, while Figure 3 achieved a higher average satisfaction rating of 7.5. These ratings suggest that Figure 3's architecture was perceived more positively and met user needs or expectations more effectively than Figure 2, demonstrating better usability, and possibly offering improved performance, scalability, and user efficiency.

A superior system architecture enables both inexperienced and experienced users to grasp the system efficiently, enhancing business performance. Post-timed assessment, participants were asked to rate their understanding of the system architecture. Results varied from satisfaction ratings: Figure 2 was marginally simpler to understand compared to Figure 3, which had fewer lines. Participants for Figure 3 found nested nodes challenging to grasp, whereas Figure 2 participants noted difficulty due to excessive line crossings hindering information flow. However, both groups found arrows intuitive and user-friendly.

Furthermore, one of the critical design elements evaluated during the timed performance assessment was the number of lines and how it affected participants' understanding of the system architecture. From the survey results, the difference in average ratings between Figure 2 (5) and Figure 3 (2.5) highlights the critical role of line density in system architecture design. Higher ratings indicate more confusion, implying that Figure 2's denser arrangement of lines and intersections were challenging for users in following the flow of information. On the other hand, the lower rating for Figure 3 implies that its simplified design allowed for better navigation of the system architecture.

A/B Testing Survey Results

The survey questions for the A/B Testing portion of the experiment can be seen in appendices E and F. The satisfaction results can be seen below in Figure 7.

Figure 7: A/B testing architecture satisfaction rating.

Participants' satisfaction ratings indicated that the experimental architecture, Figure 5, received higher satisfaction than the patented system architecture, Figure 4. Figure 4 achieved an average satisfaction rating of 6, meeting basic requirements but leaving room for improvement in enhancing user satisfaction or addressing specific usability concerns. In contrast, Figure 5 received a significantly higher average satisfaction rating of 7.75, suggesting that its design or features were more positively received.

A well-designed system architecture enhances understanding for both inexperienced and experienced users, leading to improved efficiency and performance. During A/B Testing, participants rated their understanding of the system architecture. Figure 4 received a higher average rating of 7 for understandability, suggesting it was more challenging for participants to grasp. Issues with line crossings in Figure 4 complicated information flow and structure comprehension. Conversely, Figure 5 received a lower average rating of 4.75 for understandability. Participants found its organized system layout easier to understand, mitigating challenges posed by line crossings and facilitating navigation and comprehension of system component relationships.

These results are expanded upon based on the survey results for the participants' line intersection confusion rating. Figure 4 received a higher average line intersection confusion rating of 7.25, indicating greater complexity and navigational difficulty due to dense intersecting lines. In contrast, Figure 5 had a much lower rating of 2.5, suggesting it was clearer and easier to understand. However, the distinction between solid and dashed lines remained confusing for participants.

DISCUSSION AND RECOMMENDATIONS

The purpose of conducting surveys was to gather the thoughts and opinions of inexperienced or experienced users on systems architectures. The survey questions were tailored to target and elicit specific design elements of system architectures: line intersections, node organization, clearly defined node labels, symbols, arrows, flows.

Timed Performance Assessment Discussion

The timed performance assessment was conducted on the autonomous vehicle active safety features systems. From the results of the timed assessment, some design elements had a difference in means that were statistically significant that could help develop specific heuristics. For most of the timed performance survey questions, participants responded correctly; however, there were a few questions in which participants responded incorrectly. This is interesting because it shows that a better aesthetic design does not always lead to the correct display of needed information for an architecture. System architecture design should lead to a clean and clear display of all necessary information (Stefanuk, 2020).

Statistical significance helps develop concrete solutions. However, much of the experiment led to results that were not statistically significant. There is still some room for discussion. This pilot experiment involved only sixteen participants, with not much diversity. Allowing for a greater pool of participants with different backgrounds could prove significance for some design elements that were marginally close. From the results, node outflows were statistically significant whereas node inflows, node identification, central component identification data flow identification, and nested node identification were not statistically significant. Although these characteristics were not statistically significant in this pilot experiment, all characteristics of system architecture act in harmony. A great system architecture needs all elements to balance each other, which is created by an architecture that is organized effectively (The MITRE Corporation, 2014). Further qualitative survey results from the timed performance assessment are addressed in the system architecture heuristic discussion that follows.

A/B Testing Discussion and Recommendations

A/B testing was performed on the aircraft avionics system architecture in order to design a system architecture that encompassed the feedback from the surveys and followed the heuristics outlined in Table 3 below. The result of A/B testing resulted in a system architecture shown in Figure 10.

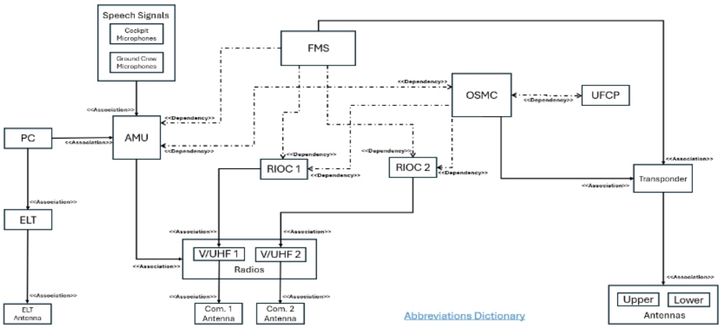

Figure 8: Optimal aircraft avionics architecture.

The system architecture has been updated to reflect the optimal solution for the aircraft avionics system. Figure 12 was designed to have less line intersections, grouped nodes, a general flow, clearly labeled elements, and a link to an abbreviation dictionary. Less line intersections were created by organizing nodes in a different manner, as compared to Figures 4 and 5. From the participants surveys, it was important to keep grouped nodes together and create a general flow. This was accomplished by keeping the antennas and radios grouped as well as making the general flow move from top to bottom. This way, system engineers could recognize different flows easier and more effectively. As stated, in UML and SysML a dashed line represents a dependency whereas a solid line represents an association. One of the most important feedback points from the participants was the clear distinction of lines. By labeling each arrow flow as a dependency or association helps clarify line design (Holt & Perry, 2018). To improve efficiency and enhance aesthetic design, the entire architecture was spread out to reduce noise. By increasing space, it allows the existing system elements to be recognized easier and for arrow flows to be followed better. Increasing space also allows for greater flexibility in the future as the system endures updates (Paradkar, 2024). Better organization leads to a better and coherent model (NASA, 2016). Through A/B Testing better design configuration can be curated that can lead to better performance, which shows the importance of usability tests.

System Architecture Heuristic Discussion and Recommendations

Many of the design elements targeted across both experiments were talked about in both a negative and positive manner. First, discussing the negative survey results, many participants found that a large number of lines and intersections made it hard to follow data paths. Participants agreed that nodes that were too close and models that looked crowded, such as Figure 2, were hard to follow. The nested nodes, as seen in Figure 3, received mixed reviews. Some users experienced difficulty understanding them, but the majority claimed it made the system architecture easier to understand. Nested nodes can help in reducing the number of lines in an architecture which was an important model element as part of this experiment. Participants also stated that diagonal lines were harder to track than lines that only moved in a horizontal or vertical path. Having clearly defined nodes, lines, symbols was a critical feedback issue. Users did not like abbreviated nodes, different patterned lines, such as dashed or solid as, or undeclared objects, such as the antennas in Figure 4. Most, if not all, participants recommended the use of a legend or key to resolve this problem, which was addressed in the solution design for A/B testing.

Despite the negative feedback, there were some positive results regarding the design of the four system architectures shown in the experiments. The lines and arrows were intuitive for every user, clearly communicating the flow of information. Node organization that resulted in an overall model flow, such as left to right or top to bottom, received positive feedback. More organization that resulted with a positive response was the grouping of similar objects as seen in Figure 5, such as speech signals, radios, and antennas. The antennas that were more clearly defined and labeled in Figure 5 received better feedback from participants than the use of an antenna symbol.

The results from the timed assessment as well as the surveys provide enlightenment into the user's perspective. Design is subjective, but some design decisions should be concrete. By understanding the thoughts and gaining feedback from users who have seen an architecture for the first time, a more intuitive design can potentially be made. Heuristics can provide clarity and act as a guide for system architects trying to solve complex problems. Right now, heuristics for systems engineers are abstract and vague. However, through this pilot experiment, the following list of heuristics was created.

Table 3. Usability heuristics for system architectures.

1. Minimize the intersection of arrows.
2. Create straight lines. No diagonal lines.
3. Label and identify all elements in a system architecture.
4. Create a clear information flow.
5. Design with enough space to promote flexibility and reduce noise.
6. Attach a link or key for all abbreviations and symbols.
7. Apply the consistency and standards set by UML and SysML or by company.

The heuristics outlined above should provide system architectures a concise and clear guide to the design of system architectures based on the usability testing performed in this experiment. Each heuristic is explained as follows. The consistency and standards of system architectures are set forth by UML and SysML. If a company uses proprietary or legacy information to design system architectures, then follow the standards set forth by that company. In order to achieve flexibility and efficiency of use, design a system architecture to not be crowded. By giving enough space for all elements, users can better understand the system architecture. With space, flexibility can be increased by accommodating for future changes and additions with minimal impact on existing elements. This foundation that space and organization are crucial for system architecture design aligns directly with the NASA Systems Engineering Handbook (NASA, 2016). By attaching a link or key to all abbreviations and symbols, an inexperienced user can better efficiently use and understand the model. A model that is efficient to use is also designed to be aesthetic and minimalist (Abulfaraj & Steele, 2020). By reducing intersections, using straight lines, clearly labeling elements, and creating a general flow, a system architecture can achieve an aesthetic and minimalist design that enhances user efficiency and performance. In following these heuristics, architects should be able to create system architectures that are organized well, convey clear information, and set the foundation for future designs. Exceptional systems engineering is fundamental to any successful business, which starts with a well-designed system architecture (INCOSE, 2023).

CONCLUSION

The purpose of this research is to act as a pilot for further human factor research related to systems engineering. Further research would include the usability testing of more people, eventually leading to testing with system architects. The application of systems engineering spreads to multiple industries. By further testing with a variety of system architectures and diagrams, a more complete list of heuristics could be developed.

Systems engineering is ubiquitous and system architectures are a key element in the system engineering process. By further defining the characteristics of what constitutes a good architecture, businesses can have better performance and efficiency for all future products and increase the connection and understanding for all lines of business.

APPENDICES

Appendices will be provided upon request.

REFERENCES

Abulfaraj, A., & Steele, A. (2020). Detailed Usability Heuristics: A Breakdown of Usability Heuristics to Enhance Comprehension for Novice Evaluators. In HCI International 2020 - Late Breaking Papers: User Experience Design and Case Studies (pp. 3–18). essay, Springer International Publishing.

Brook, P. (2024). Systems engineering heuristics. Guide to the System Enginnering Body of Knowledge. https://sebokwiki.org/wiki/Systems_Engineering_Heuristics

Characteristics of Good Architecture. Sparx Systems Pty Ltd. (2023). https://sparxsystems.com/enterprise_architect_user_guide/14.0/guidebooks/ea_characteristics_of_good_architecture.html

Cherry, K. (2022, November 8). How Heuristics Help You Make Quick Decisions. Verywell Mind. https://www.verywellmind.com/what-is-a-heuristic-2795235

Frøkjær, E., Hertzum, M., & Hornbæk, K. (2000). Measuring usability: Are effectiveness, efficiency, and satisfaction really correlated? Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 345–352. https://doi.org/10.1145/332040.332455

Gallo, A. (2017, November 27). A refresher on a/B testing. Harvard Business Review. https://hbr.org/2017/06/a-refresher-on-ab-testing

Guimaraes, Felipe. (2022, September 2). Nielsen's Heuristics: 10 Usability Principles to Improve UI Design. Aela School - Where Ambitious Designers Thrive. https://aelaschool.com/en/interactiondesign/10-usability-heuristics-ui-design/

Hart, L. E. (2015, July 30). Introduction to model-based system engineering (MBSE) and SysML [PowerPoint Slides]. https://www.incose.org/docs/default-source/delaware-valley/mbse-overview-incose-30-july-2015.pdf

Holt, J., & Perry, S. (2018). Chapter 5: The SysML Notation. In SysML for Systems Engineering: A Model-Based Approach (3rd ed., pp. 121–222). essay, Institution of Engineering and Technology.

INCOSE. (2015). Systems engineering Jr. Handbook. https://www.incose.org/docs/default-source/los-angeles/libraryresources/se-jr-handdbook-2015.pdf?sfvrsn=2

INCOSE. (2023). Incose Systems Engineering Handbook (5th ed.). John Wiley & Sons, Incorporated.

Keating, C., Ralph, R., Unal, R., Dryer, D., Sousa-Poza, A., Safford, R., Peterson, W. and Rabadi, G. (2003) ‘System of systems engineering’, Engineering Management Journal, Vol. 15, No. 3, pp. 35–44.

Maier, M. W., & Rechtin, E. (2009). Art of Systems Architecting (3rd ed.). CRC Press.

McCormick, E. J., & Sanders, M. S. (1993). Human Factors in Engineering and Design (7th ed.). McGraw-Hill.

McGrath, A., & Jonker, A. (2023, December 2). What is model-based systems engineering (MBSE)?. IBM. https://www.ibm.com/topics/model-based-systems-engineering

Menshenin, Y., Mordecai, Y., Crawley, E., & Cameron, B. (2022, March). (rep.). Model-Based System Architecting and Decision-Making (pp. 1–50). Cambridge: Massachusetts Institute of Technology.

Nodes in UML models. IBM. (2023, July 18). https://www.ibm.com/docs/en/dma?topic=diagrams-nodes

NASA Systems Engineering Handbook. (2016). National Aeronautics and Space Administration.

Paradkar, S. (2024, February 22). Crafting highly available systems: Practices, patterns, and principles-part 1. Medium. https://medium.com/oolooroo/crafting-highly-available-systems-practices-patterns-and-principles-part-1-8a1d4d301781

Schatz, J. (2023, June 5). Why Every Industry Needs Systems Engineering. SPEC Innovations - The Systems Engineering Experts. https://specinnovations.com/blog/why-every-industry-needs-systems-engineering

Shevchenko, N. (2020, December 21). An introduction to model-based systems engineering (MBSE). SEI Blog. https://insights.sei.cmu.edu/blog/introduction-model-based-systems-engineering-mbse/

Stefanuk, A. (2020, May 4). The Importance of Information Architecture to UX Design. UXmatters. https://www.uxmatters.com/mt/archives/2020/05/the-importance-of-information-architecture-to-ux-design.php

Sullivan, McCool, Perlman (2016). System and Method for Managing Audio and Video Channels for Video Game Players and Spectators. U. S Patent No. 9339728B2. U. S. Patent and Trademark Office.

Systems Engineering Guide. (2014). The MITRE Corporation.

Vay Technologies. (2024). https://technologies.vay.io/

Weilkiens, T., Lamm, J. G., Roth, S., & Walker, M. (2022). Model-Based System Architecture (2nd ed.). Wiley.