Available online at
https://jcrinn.com/
https://crinn.conferencehunter.com/

Journal of Computing Research and Innovation 9(2) 2024

Journal of Computing Research and Innovation

Evaluation of SPARK C++ Usability: Modular Programming for Mechanical and Civil Engineering Students

Azrina Suhaimi[1], Mohamad Faizal Ab Jabal[2*], Harshida Hasmy[3], Azyan Yusra Kapi[4]

[1,2,3,4] *Department of Computing Sciences Study, College of Computing, Informatics & Mathematics, Universiti Teknologi MARA Johor Branch, Pasir Gudang Campus, 81750 Johor, Malaysia*
[2] *Applied Mathematics & System Development (AMSys)-Special Interest Group (SIG), Universiti Teknologi MARA Johor Branch, Pasir Gudang Campus, 81750 Johor, Malaysia*

ARTICLE INFO

Article history:
Received 1 July 2024
Revised 24 July 2024
Accepted 24 July 2024
Online first
Published 1 September 2024

Keywords:
Educational Technology
Modular Programming
Nielsen Usability Metrics
Programming Education
SPARK C++
Web-based Learning Tools

DOI:
10.24191/jcrinn.v9i2.481

ABSTRACT

Technological innovations have transformed education, offering learning opportunities outside traditional classrooms through online resources. However, self-learning programming remains challenging for beginners. The SPARK C++ tool was developed to assist Mechanical and Civil Engineering students at Universiti Teknologi MARA in mastering C++ modular programming. This study evaluates the usability of SPARK C++ using Nielsen's metrics: learnability, efficiency, memorability, errors, and satisfaction. Sixty-three students from these engineering programs participated, providing data through Nielsen's Usability Test, which included Likert scale questions and open-ended feedback. The results revealed significant differences in usability perceptions by gender and program, with female Mechanical Engineering students reporting higher satisfaction. Key strengths were user-friendly design, engaging content, interactive quizzes, and multimedia features. Areas for improvement included better error messages, more comprehensive notes, and downloadable results. Enhancing these aspects can improve learning experiences and retention rates. This study highlights the need for continuous improvement of educational technologies based on user feedback and suggests future research should broaden usability testing to ensure inclusivity and adaptability.

1. INTRODUCTION

Technological advancements have altered the educational landscape, allowing students to learn beyond the classroom. Nowadays, higher education institutions commonly feature ubiquitous learning environments (Aljawarneh, 2020). Web-based learning tools have made a significant contribution to this shift by providing learning opportunities ranging from beginner to advanced. However, learning programming

[2*] Corresponding author. *E-mail address*: m.faizal@uitm.edu.my
https://dx.doi.org/10.24191/jcrinn.v9i2.481

without the assistance of a lecturer can be difficult, particularly for beginners. Despite the availability of numerous programming learning tools and websites, a lack of fundamental programming knowledge can result in difficulties and frustration (Kapi et al., 2023).

Furthermore, understanding programming necessitates additional skills such as problem-solving, logical thinking, metacognitive abilities, and creativity. Specifically, the 'why' and 'when' of metacognitive skills are critical in the early stages of learning programming (Cheah, 2020). Learning and understanding programming requires a variety of instructional methods, such as foundational classes, practical or laboratory sessions, and tutorial sessions. There are several effective approaches for teaching programming. One approach focuses on specific programming tools to ensure that students fully understand computational concepts (Flórez et al., 2017). Another approach emphasises pair or group programming over individual work (Umapathy & Ritzhaupt, 2017). Furthermore, incorporating blended and project-based learning into programming classes has been beneficial (Hsu et al., 2018).

Despite these approaches, programming courses continue to have high failure rates, even at the introductory level (Cheah, 2020). The most difficult aspect of teaching programming is developing students' problem-solving abilities and logical reasoning. Helping students understand abstract programming concepts, such as control structures and developing algorithms to solve concrete problems, is also difficult. In response to these issues, a web-based learning tool called SPARK C++: Modular (C++ Simplified Practices with Analogies and Resources for Knowing Modular) was created (Kapi et al., 2023). However, preliminary feedback indicates that users' perceptions of usability vary, which may have an impact on learning outcomes, user satisfaction, and overall engagement with the tool. SPARK C++'s effectiveness in improving educational outcomes is determined by its ability to deliver a consistently positive user experience across diverse student demographics and learning environments.

This study will conduct a comprehensive usability metrics analysis across user segments to evaluate SPARK C++, identify factors contributing to user experience disparities, and propose enhancements to make the tool more effective and accessible to all learners. Usability metrics such as learnability, efficiency, memorability, errors, and overall satisfaction are highly valued by researchers. Learnability, or how easily users can complete tasks initially, is critical for implementing educational technology (Tsai & Tsai, 2020; Didik-Hariyanto & Bruri-Triyono, 2020). Tools that provide real-time feedback improve efficiency, or the speed with which tasks can be completed after learning. Memorability aids users in recalling how to use a tool after a break and is enhanced by engaging and interactive designs (Crandall et al., 2015; Chu et al., 2019). Error reduction and clear recovery methods enhance the user experience and learning outcomes (Demitriadou et al., 2019; Tsai & Tsai, 2020). Overall satisfaction, including ease of use and engagement, results in improved learning and motivation (Hotle, 2020; Didik-Hariyanto & Bruri-Triyono, 2020).

The significance of this study stems from its potential to improve educational outcomes through improved usability of web-based learning tools. By systematically analysing and addressing usability issues in SPARK C++, this research hopes to contribute to the development of more effective and accessible educational technologies. This can result in better learning experiences, increased engagement, and higher retention rates for programming students, ultimately preparing them to succeed in the digital age.

2. LITERATURE REVIEW

The usability of digital educational resources has been assessed across various disciplines, highlighting the importance of comprehensive usability guidelines to enhance the learning experience. Research investigating the use of information technology and interactive activities in computer science education found significant improvements in learner engagement, with participants reporting enhanced learning experiences due to interactive and collaborative activities. The primary goal of these tools is to enhance student engagement and learning outcomes through the integration of interactive and multimedia elements.

https://dx.doi.org/10.24191/jcrinn.v9i2.481

Research has consistently highlighted the importance of designing educational technology that incorporates animations, interactive sounds, and narratives to make complex programming concepts more accessible (Demitriadou et al., 2019; Didik-Hariyanto & Bruri-Triyono, 2020). This approach is aligned with the broader objective of creating an engaging and effective learning environment that caters to the diverse needs of contemporary students.

The methodologies employed in these studies typically encompass a blend of quantitative and qualitative approaches to evaluate the usability and effectiveness of educational tools. For instance, Didik-Hariyanto and Bruri-Triyono (2020) utilised the USE questionnaire to collect data on user satisfaction and the overall usability of a personalised adaptive e-learning system. In a similar vein, Demitriadou et al. (2019) conducted comparative evaluations of virtual and augmented reality tools to teach mathematics, providing valuable insights into the efficacy of these technologies in enhancing learning outcomes. These studies often include pre- and post-intervention assessments, user feedback, and comprehensive usability testing to ensure robust evaluation.

The effectiveness of multimedia-rich educational tools in teaching programming has been substantiated by numerous studies. Research has demonstrated that incorporating interactive and multimedia elements significantly enhances student engagement and knowledge retention (Hotle, 2020; Hoyt, 1999). For example, a systematic review by Crandall et al. (2015) on augmented reality game-based learning in STEM education revealed that these tools are effective in making abstract concepts more comprehensible and engaging for students. Furthermore, studies have shown that students who utilise these tools exhibit higher levels of motivation and engagement, which are crucial factors for successful learning in complex subjects such as programming (Sanchez et al., 2020).

The integration of browser-based platforms provides greater accessibility and flexibility in learning. These tools enable students to learn at their own pace and from any location, thus making education more inclusive and accommodating (Chu et al., 2019). The use of real-time feedback and practice exercises, as highlighted in several studies, is particularly beneficial in reinforcing learning and helping students master programming skills more effectively (Tsai & Tsai, 2020). The ability to provide personalised learning experiences through adaptive technologies further enhances the effectiveness of these tools in meeting individual learning needs.

Furthermore, recent literature points out the significant potential of browser-based, multimedia-rich platforms in revolutionising the learning experience for novice programming students. These tools not only improve engagement and learning outcomes but also offer flexible and personalised learning experiences. As educational technology continues to evolve, ongoing research and development in this field are essential to optimise the design and implementation of such innovative teaching tools (Becker et al., 2018). The insights gained from recent studies provide a solid foundation for future advancements in educational technology, ensuring that it remains responsive to the needs of learners in the digital age.

The usability of innovative teaching tools for programming languages has been a focal point of research over the years, emphasising the importance of usability metrics such as learnability, efficiency, memorability, errors, and overall satisfaction. Learnability, which pertains to how easily users can accomplish basic tasks on their first use, has been highlighted as a crucial factor in the adoption of educational technology. Research by Tsai and Tsai (2020) indicates that well-designed digital educational games enhance the learnability of programming concepts. Similarly, Didik-Hariyanto and Bruri-Triyono (2020) found that personalised adaptive e-learning systems with intuitive designs significantly improve initial learning experiences for users.

Efficiency, which measures how quickly users can perform tasks after learning the system, is another key metric. Studies have shown that tools which streamline the coding process and provide real-time feedback greatly enhance efficiency. Sanchez and Wiley (2020) highlight that digital technologies incorporating immediate feedback mechanisms help students quickly identify and correct errors, thereby

https://dx.doi.org/10.24191/jcrinn.v9i2.481

improving their coding efficiency and maintaining their motivation. This efficiency is crucial for keeping students engaged and progressing in their programming education.

Memorability, referring to how well users can remember how to use a tool after not using it for some time, is essential for effective educational tools. Crandall et al. (2015) demonstrated that augmented reality games for teaching abstract concepts in STEM education have high memorability, as students easily recall how to use the tool due to its engaging and interactive nature. Chu et al. (2019) also found that e-learning tools with consistent and straightforward navigation aid in skill retention, enabling students to effectively use the system even after a break.

The metric of errors, which includes the number and severity of mistakes users make and how easily they can recover from them, is critical in usability studies. Research by Demitriadou et al. (2019) revealed that educational tools with clear error messages and easy-to-follow troubleshooting guides significantly reduce user frustration and improve learning outcomes. Similarly, Tsai and Tsai (2020) noted that digital games provide contextual hints and support minimise errors and facilitate a better understanding of programming concepts.

Overall satisfaction, encompassing the user's overall experience with the tool, including ease of use, engagement, and perceived value, is vital for the success of educational technologies. High levels of satisfaction are linked to increased motivation and better learning outcomes. Hotle (2020) indicated that students using multimedia-rich platforms reported higher satisfaction due to engaging content and interactive features. Likewise, Didik-Hariyanto and Bruri-Triyono (2020) found that adaptive e-learning systems that offer personalised learning paths greatly enhance student satisfaction by addressing individual learning needs.

Although many usability tests have been conducted in previous studies to evaluate educational tools, there exists a gap in the literature specifically regarding how different user groups perceive these tools. Thus, despite only capturing data across several usability metrics (learnability, efficiency, memorability, errors, and satisfaction), this study also provides a detailed analysis of how these metrics vary across different user groups, such as faculty (Mechanical or Civil Engineering) and gender. Additionally, this study examines the impact of these tools for further enhancement. This suggests a potential gap in understanding how different segments of the user base perceive the tool's usability and its impact on their learning experiences.

In summary, the usability metrics of learnability, efficiency, memorability, errors, and overall satisfaction are pivotal in assessing the effectiveness of innovative teaching tools for programming languages. These metrics not only influence the adoption and success of educational technologies but also significantly impact student engagement and learning outcomes. As educational technology continues to evolve, focusing on these usability aspects will be crucial in developing tools that effectively support programming education.

3. METHODOLOGY

This section outlines the methodology employed to assess the usability and content analysis of the SPARK C++: Modular tool among undergraduate students at Universiti Teknologi MARA Pasir Gudang Campus. The study was structured into four distinct phases, each designed to systematically gather and analyse data on participants' experiences with the educational tool. The methodology was structured into four essential phases, illustrated in Fig. 1.

https://dx.doi.org/10.24191/jcrinn.v9i2.481

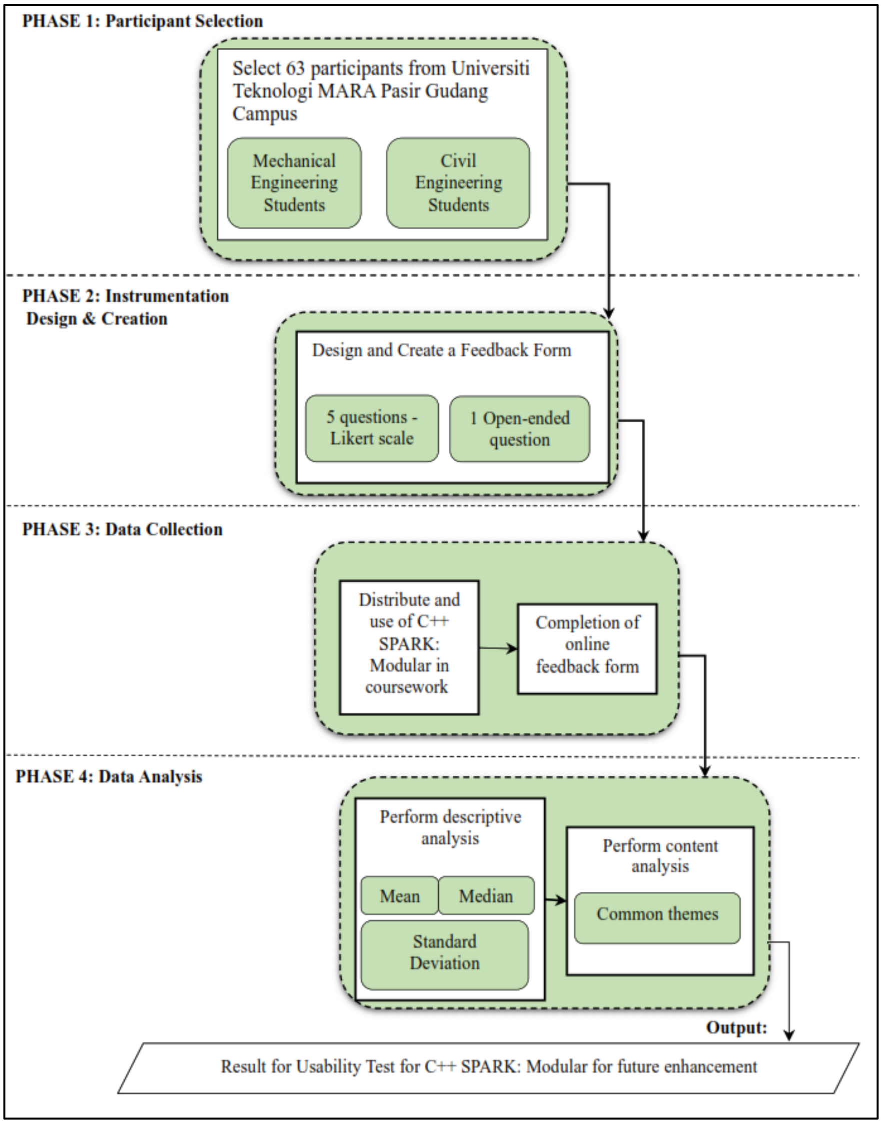

Fig. 1. Methodology Used for Evaluating the Usability of the SPARK C++: Modular

Based on Fig. 1, the first phase includes the participant selection. Convenience sampling was utilised to select a representative sample of 63 participants. Table 1 shows the demographic distribution for the participants.

Table 1. Demographic distribution of participants by engineering discipline and gender

Engineering Discipline	Total	Male	Female
Civil Engineering	34	12	22
Mechanical Engineering	29	24	5

https://dx.doi.org/10.24191/jcrinn.v9i2.481

Among the Civil Engineering participants, 34 students took part in the evaluation, with a noticeable majority being female (22 participants) compared to male students (12 participants). This distribution features a notable representation of female students within the Civil Engineering program at the institution. In contrast, the Mechanical Engineering cohort comprised 29 participants, predominantly male with 24 participants, while female students constituted a smaller subset of the group (5 participants). This demographic profile reflects a higher proportion of male students enrolled in Mechanical Engineering, indicative of the discipline's gender composition at the university. Participants entered into a mutual agreement to participate in the study, acknowledging the objectives and procedures involved.

Subsequently, the Nielsen's Usability Test (NAU), introduced by Nielsen in 1994, was utilized to gather both quantitative and qualitative feedback in the second phase of the methodology. The quantitative section featured Likert scale questions assessing key usability factors: learnability, efficiency, memorability, errors, and satisfaction on a scale of 1 to 7, where 7 indicated the highest level of satisfaction or effectiveness. The five questions include:

(i) Learnability: How easy is it for you to accomplish basic tasks the first time you encounter the application?

(ii) Efficiency: Once you have learned the application design, how quickly can you perform tasks?

(iii) Memorability: When you return to the application after a period of not using it, how easily can you reestablish proficiency?

(iv) Errors: How many errors do you make, how severe are these errors, and how easily can you recover from the errors? If you don't encounter any errors, please rate 7.

(v) Satisfaction: How pleasant is it to use the application?

Complementary to the Likert scale questions, the qualitative section allowed participants to provide open-ended feedback on their experiences. It aimed at eliciting detailed insights into user experiences, perceived strengths and weaknesses of the tool, and suggestions for improvement. The question is, what aspects of this application were most useful or valuable? Please input any comments for future recommendations.

In Phase 3, participants were asked to use the tool as depicted in Fig. 2 as part of their regular coursework over a period of 10 days. There was no minimum number of hours required, and participants could complete their use of the tool within a single day if they chose. After using the tool, they were required to complete an online feedback form. To ensure unbiased feedback, participants were instructed to complete the form immediately after their use of the tool, without discussing their experiences with others. This timeframe and procedure were established to maintain the integrity of the usability study and to prevent participants from influencing each other's feedback.

The collected data was subjected to descriptive statistical analysis to derive meaningful insights from the quantitative responses in Phase 4. The analysis included calculating the mean, median, and standard deviation for each usability factor. Additionally, this study also categorised how usability metrics vary across different user groups defined by faculty (Mechanical or Civil Engineering) and gender. This additional analysis aimed to identify significant differences in usability perceptions among different demographic segments, thereby addressing the gap in understanding how various user groups interact with the SPARK tool. After that, an open-ended question was analysed using content analysis techniques to identify recurring themes and patterns related to usability, user satisfaction, and recommendations for tool improvement.

https://dx.doi.org/10.24191/jcrinn.v9i2.481

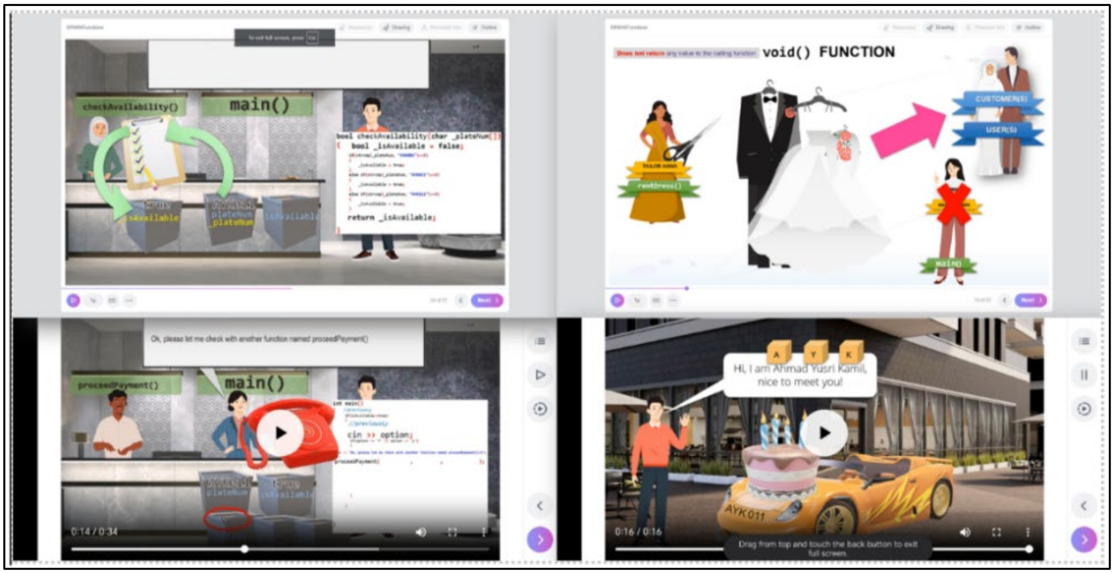

Fig. 2. User Interface of SPARK (top) full-screen view on a laptop, showcasing the lesson overview through analogies, (bottom) full-screen view on a smartphone, demonstrating mobile accessibility

This methodological framework ensured a comprehensive evaluation of the SPARK C++: Modular tool's usability, leveraging both quantitative and qualitative data collection and analysis methods. The structured approach facilitated a detailed understanding of user experiences and informed recommendations for enhancing the tool's effectiveness in educational settings.

4. RESULT AND DISCUSSION

In this section, descriptive statistics including mean, standard deviation, minimum, and maximum scores for each usability metric were discussed and explained in detail. These metrics are evaluated across different user groups defined by program (Civil and Mechanical Engineering) and gender to assess the usability of the SPARK C++ application. The results and discussion that follow are based on the data in Table 2.

Table 2. Detailed descriptive statistics for each usability metric across different user groups defined by program and gender

Metric	Program	Gender	Mean	Std Dev	Min	Max
Learnability	Civil Engineering	Female	5.68	0.99	3	7
		Male	5.42	1.31	3	7
	Mechanical Engineering	Female	6.20	0.45	6	7
		Male	5.67	1.17	3	7
Efficiency	Civil Engineering	Female	5.64	0.9	4	7
		Male	5.83	1.27	3	7
	Mechanical Engineering	Female	5.40	0.89	4	7
		Male	6.00	0.83	4	7
Memorability	Civil Engineering	Female	5.59	0.96	4	7
		Male	5.58	1.00	3	7
	Mechanical Engineering	Female	6.00	0.71	5	7
		Male	6.00	0.78	4	7
Errors	Civil Engineering	Female	5.36	1.05	4	7

https://dx.doi.org/10.24191/jcrinn.v9i2.481

		Male	5.50	1.17	3	7
	Mechanical Engineering	Female	5.60	0.89	5	7
		Male	5.71	1.12	4	7
Satisfaction	Civil Engineering	Female	6.14	0.89	4	7
		Male	5.83	1.19	3	7
	Mechanical Engineering	Female	6.60	0.55	6	7
		Male	6.50	0.66	5	7

The detailed analysis of the usability metrics for the SPARK C++ application reveals significant differences in how various user groups perceive the tool's usability. The following discussion delves into these differences based on the five Nielsen usability metrics: Learnability, Efficiency, Memorability, Errors, and Satisfaction.

(i) Learnability

Learnability is crucial for any educational tool, as it directly impacts the user's initial interaction and willingness to continue using the tool. Female students in both programs reported higher learnability scores than their male counterparts, with female Mechanical Engineering students showing the highest mean score (6.20) and lowest standard deviation (0.45). This suggests that the application is particularly easy to learn for female students in the Mechanical Engineering program. The range of scores (6-7) further supports this, indicating a consistent perception of high learnability among these users. Male students, especially in Civil Engineering, reported more variability in their learnability scores, suggesting a need for potential adjustments to improve the learning experience for this group.

(ii) Efficiency

Efficiency measures how quickly and smoothly users can complete tasks using the tool. Efficiency scores were relatively high and consistent across both genders in the Mechanical Engineering program, with male students reporting a slightly higher mean score (6.00). Female students in the Civil Engineering program had a lower mean efficiency score (5.64) compared to their male counterparts (5.83), but the standard deviation was also lower (0.90), indicating a more consistent experience. These results suggest that while the application is generally efficient, there might be room for improvement to enhance the efficiency for female Civil Engineering students.

(iii) Memorability

Memorability is important for educational tools used intermittently. High memorability reduces the learning burden during repeated interactions. Memorability scores were high and consistent across both programs and genders, with Mechanical Engineering students reporting a mean score of 6.00 for both genders. The relatively low standard deviations indicate that most users found the application easy to remember how to use after periods of non-use. This consistency suggests that the design of the application effectively supports users in retaining their knowledge of its use.

(iv) Errors

Errors in this context reflect how frequently users encounter problems and how severely this impact their experience. Error rates were slightly higher for male students in both programs. Female students in the Civil Engineering program reported a mean error score of 5.36, whereas males reported 5.50. In the Mechanical Engineering program, females reported a mean of 5.60 and males 5.71. Although these differences are small, they suggest that female students might find the application slightly more user-friendly, resulting in fewer mistakes.

https://dx.doi.org/10.24191/jcrinn.v9i2.481

This indicates that while the application is generally reliable, enhancing error handling or providing clearer user guidance could further improve its usability.

(v) Satisfaction

User satisfaction is a comprehensive indicator of the application's success in meeting its intended goals. The high average score is encouraging and suggests that most users find the tool beneficial and enjoyable. Satisfaction levels were higher among female students, especially in the Mechanical Engineering program, where female students reported a mean satisfaction score of 6.60 compared to 6.50 for males. In the Civil Engineering program, female students reported a mean score of 6.14, compared to 5.83 for males. These higher satisfaction levels among female students could be attributed to their higher scores in learnability and lower error rates, contributing to an overall better user experience.

Overall, the SPARK C++ application demonstrates promising usability metrics, with higher scores among female students. Additionally, students in the Mechanical Engineering program tend to report slightly higher usability scores compared to those in the Civil Engineering program. The higher satisfaction reported by female students can be linked to research indicating that females often benefit from e-learning tools that are highly interactive, include multimedia content, and provide immediate feedback, which are key features of the SPARK C++ tool (Yu & Deng, 2022; Soub et al., 2021). In the context of Mechanical Engineering students, the alignment of SPARK C++ with the practical and application-oriented focus of their curriculum may explain the slightly higher usability scores. Mechanical Engineering programs typically emphasize problem-solving and hands-on learning (Leung et al., 2021), which the SPARK C++ tool effectively supports through its interactive quizzes and practical examples. This indicates that while the application is well-received across different user groups, there are opportunities to further tailor it to meet the specific needs of each program and gender.

Further insights were also gained from the content analysis of the feedback provided by users. The open-ended responses revealed several common themes that highlight the strengths and areas for improvement of the SPARK C++ application. Table 3 summarises the common themes identified from the open-ended question, "What aspects of this application were most useful or valuable? Please input any comments for future recommendations.". The feedback from users highlights several key aspects of the SPARK: Modular Programming in C++ tool that were found to be particularly valuable. The key aspects are categorised into the following common themes:

Table 3. Summarisation of comments based on common themes

Common Themes	Comments
User-Friendliness and Accessibility	• Improved for user-friendly • User-friendly options • The arrangement of the button
Educational Content	• The slide is particularly useful and interesting, easy to understand. • The way of teaching is remarkably interesting • The way of the explanation • Absolute best way to study because of animation and explanation that easy to understand and can relate in our life • The examples and explanations are quite easy to understand • The sound of the presenter is clear. Genuinely nice • Easily to use and friendly help to improve my knowledge with the way or method given when we answer wrong • Function learning
Quizzes and Practice Tests	• The quiz taken in the slide can be attempted repeatedly. Other than that, answers and explanations provided. • The quiz section is most useful because when we answer the questions with wrong answers, it will give the answers with explanations.

https://dx.doi.org/10.24191/jcrinn.v9i2.481

	• Practice quiz as it can help you to understand the content easily. • The quiz after the information • By answering quiz, make me more confident and understanding • The application teaches good skills to answer all the functions questions by explaining what function is and then asking the user to try some questions after the teaching session. • The test and the quiz which can be attempted repeatedly. • The best part is the question where it requires the user to drag the answers to the blank spot. Hoping to see more of it. • The voice-over and video aspect creates a unique and engaging experience.
Multimedia and Interactive Features	• The learning tool has a separate video that pauses on itself so it is easy for you to replay the video rather than finding which of the videos you want to rewatch. • Video explanation is using a clear sound • The video representation is nice.
Suggestions and Recommendations	• Not auto play slide, because I can take my time adapting the information • Maybe more notes • I think some of the videos should be compiled together because it is easier to watch and a bit of a hassle to click a button to see the next video. • Downloadable and printable results would be useful.

(i) User-Friendliness and Accessibility

Users frequently mentioned the tool's user-friendliness and accessible design. Comments about the arrangement of buttons and overall user-friendly options suggest that the interface is intuitive and easy to navigate. This likely contributes to the overall positive usability metrics, as an intuitive design can significantly enhance user satisfaction and reduce errors.

(ii) Educational Content

The educational content provided by the SPARK C++ tool was highly praised. Users appreciated the clarity and usefulness of the slides, animations, and explanations. The ability to relate the content to real-life scenarios and the clear sound of the presenter were also highlighted. These features are crucial in helping students understand and retain complex programming concepts.

(iii) Quizzes and Practice Tests

Quizzes and practice tests were another highly valued feature. Users noted that the ability to repeatedly attempt quizzes and receive explanations for incorrect answers greatly enhanced their understanding and confidence. This interactive aspect of the tool not only aids in learning but also engages students more effectively by providing immediate feedback and reinforcing knowledge through practice.

(iv) Multimedia and Interactive Features

The inclusion of multimedia elements, such as videos and voice-overs, received positive feedback. These features make the learning experience more engaging and help to break down complex information into more manageable and understandable segments. The ability to replay videos and pause them as needed allows students to learn at their own pace, further enhancing the tool's usability.

(v) Suggestions and Recommendations

Users also provided valuable suggestions for improvement. Some recommended more notes and the option to compile videos for easier viewing. Others suggested making downloadable and printable results available, which could facilitate thorough discussions in class. These

https://dx.doi.org/10.24191/jcrinn.v9i2.481

recommendations indicate areas where the tool can be further refined to meet user needs and preferences better.

In conclusion, the descriptive analysis of the usability metrics for the SPARK C++ application shows that the tool is generally well-received by users, particularly among female students and those in the Mechanical Engineering program. While the application is effective across different user groups, there is potential to better meet the needs of male students and those in the Civil Engineering program. The metrics for learnability, efficiency, memorability, errors, and satisfaction all show promising results, emphasising the application's strengths in teaching C++ modular programming. Additionally, the content analysis of user feedback highlights the most valued aspects of the SPARK C++ application. Key themes include user-friendliness, accessibility, educational content, quizzes and practice tests, multimedia and interactive features, and suggestions for improvement. Users particularly appreciated the intuitive design, clear and engaging educational content, interactive quizzes, and multimedia elements. Constructive feedback offers valuable recommendations for enhancing the tool's usability and effectiveness, such as providing more notes, compiling videos, and offering downloadable and printable results. Overall, the combination of quantitative and qualitative analyses demonstrates that the SPARK C++ application is a robust educational tool with significant strengths. By addressing the identified areas for improvement and incorporating user suggestions, the application can be further refined to offer an even more effective and satisfying learning experience for all users.

5. CONCLUSION AND FUTURE WORK

This study evaluated the usability of SPARK C++: Modular Programming tool among Mechanical and Civil Engineering students at Universiti Teknologi MARA Pasir Gudang Campus. By leveraging Nielsen's Attributes of Usability (NAU), the study assessed five key factors: learnability, efficiency, memorability, errors, and satisfaction. The findings indicate that SPARK C++ is a highly effective educational tool, with significant positive feedback from users. Female students, particularly in the Mechanical Engineering program, reported higher satisfaction and better usability metrics compared to their male counterparts. The educational content, user-friendly interface, interactive quizzes, and multimedia features were highlighted as the most valuable aspects of the tool. The quantitative analysis demonstrated high scores across all usability metrics, with female students consistently reporting better experiences. The qualitative feedback provided deeper insights into specific strengths and areas for improvement. Users appreciated the clarity and engagement provided by the educational content and interactive features, but also suggested enhancements such as more comprehensive notes, video compilations, and downloadable results.

However, there are several areas for future work that can further enhance the SPARK C++: Modular Programming tool. Key areas for improvement include developing more detailed notes and comprehensive explanations to support varied learning styles, as well as creating consolidated video modules to facilitate easier navigation and uninterrupted learning experiences. Additionally, introducing features for downloadable and printable quiz results to aid in offline review and classroom discussions will be beneficial. Implementing adaptive learning paths to cater to individual student needs, thereby improving engagement and learning outcomes, is also crucial. Furthermore, improving error messages and providing clearer troubleshooting guides to reduce user frustration and enhance learning efficiency will make the tool more user-friendly. Expanding the usability testing to include a broader demographic will ensure that the tool meets the needs of a diverse student population.

By incorporating these enhancements, the SPARK C++ tool can become an even more effective and satisfying educational resource, ultimately contributing to better learning outcomes and higher retention rates for programming students. Future research should continue to explore innovative approaches to improve the usability and effectiveness of educational technologies in higher education. This ongoing

https://dx.doi.org/10.24191/jcrinn.v9i2.481

development will help in creating more engaging, accessible, and efficient learning tools that can adapt to the evolving needs of students in the digital age.

6. ACKNOWLEDGEMENTS/FUNDING

The authors declare that this research was conducted independently, without any self-benefits, commercial, or financial conflicts of interest. This work did not receive any external funding.

7. CONFLICT OF INTEREST STATEMENT

The authors agree that this research was conducted in the absence of any self-benefits, commercial or financial conflicts and declare the absence of conflicting interests with the funders.

8. AUTHORS' CONTRIBUTIONS

Azrina Suhaimi: Conceptualisation, methodology, formal analysis, investigation, and writing-original draft; **Mohamad Faizal Ab Jabal:** Conceptualisation, writing – review & editing, validation, and data curation; **Harshida Hasmy:** Resources, formal analysis, and investigation; **Azyan Yusra Kapi:** Conceptualisation, methodology, and software.

9. REFERENCES

Aljawarneh, S. A. (2020). Reviewing and exploring innovative ubiquitous learning tools in higher education. *Journal of Computing in Higher Education*, *32*(1), 57-73. https://doi.org/10.1007/s12528-019-09207-0

Becker, S. A., Brown, M., Dahlstrom, E., Davis, A., DePaul, K., Diaz, V., & Pomerantz, J. (2018). *NMC Horizon Report: 2018 Higher Education Edition*. Louisville, Co: Educause, 2018. https://library.educause.edu/~/media/files/library/2018/8/2018horizonreport.pdf

Cheah, C. S. (2020). Factors contributing to the difficulties in teaching and learning of computer programming: A literature review. *Contemporary Educational Technology*, *12*(2), ep272. https://doi.org/10.30935/cedtech/8247

Chu, A., Biancarelli, D., Drainoni, M. L., Liu, J. H., Schneider, J. I., Sullivan, R., & Sheng, A. Y. (2019). Usability of learning moment: Features of an e-learning tool that maximize adoption by students. *The Western Journal of Emergency Medicine*, *21*(1), 78–84. https://doi.org/10.5811/westjem.2019.6.42657

Crandall, P. G., Engler, R. K., Beck, D. E., Killian, S. A., O'bryan, C. A., Jarvis, N., & Clausen, E. (2015). Development of an augmented reality game to teach abstract concepts in food chemistry. *Journal of Food Science Education*, *14*(1), 18–23. https://doi.org/10.1111/1541-4329.12048

Demitriadou, E., Stavroulia, K., & Lanitis, A. (2019). Comparative evaluation of virtual and augmented reality for teaching mathematics in primary education. *Education and Information Technologies*, *25*(1), 381–401. https://doi.org/10.1007/s10639-019-09973-5

Didik-Hariyanto, M., & Bruri-Triyono, T. K. (2020). Usability evaluation of personalized adaptive e-learning system using USE questionnaire. *Knowledge Management & E-Learning: An International Journal*, *12*, 85–105. https://doi.org/10.34105/j.kmel.2020.12.005

Flórez, F. B., Casallas, R., Hernández, M., Reyes, A., Restrepo, S., & Danies, G. (2017). Changing a generation's way of thinking: Teaching computational thinking through programming. *Review of Educational Research*, *87*(4), 834–860. https://doi.org/10.3102/0034654317710096

Hotle, F. (2020). Technology and the science class: Going beyond the walls of the disk drive. In T. R. Cannings & L. Finkel (Eds.), *The technology age classroom*. Wilsonville: Franklin, Breddle and Associates.

Hoyt, J. E. (1999). *Does the delivery method matter? Comparing technologically delivered distance education with on-campus instruction*. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=0dd75ed2b093b5097b6809311bad09a0e680d718

Hsu, T.-C., Chang, S.-C., & Hung, Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education*, *126*, 296–310. https://doi.org/10.1016/j.compedu.2018.07.004

Kapi@Kahbi, A. Y., Suhaimi, A., Hasmy, H., & Jabal, M. F. A. (2023). Innovation of digital multimedia: Recognizing the advantages. *e-Academia Journal of UiTM Cawangan Terengganu*, *12*(2), 224-239. https://ir.uitm.edu.my/id/eprint/88005/

Leung, S. L., Marsh, E., Lynch, S., Sommer, H. J., Brennan, S. N., Wong, T. S., ... & Thole, K. A. (2021, July). A New Approach to Equip Students to Solve 21st-Century Global Challenges: Integrated Problem-Based Mechanical Engineering Laboratory. In *2021 ASEE Virtual Annual Conference Content Access*. American Society for Engineering Education.

Nielsen, J. (1994). *Usability Engineering*. Morgan Kaufmann Publishers Inc.

Scherer, R., Siddiq, F., & Viveros, B. S. (2020). A meta-analysis of teaching and learning computer programming: Effective instructional approaches and conditions. *Computers in Human Behavior*, *109*, 106349. https://doi.org/10.1016/j.chb.2020.106349

Soub, T. F. A., Alsarayreh, R. S., & Amarin, N. Z. (2021). Students' satisfaction with using e-learning to learn Chemistry in light of the Covid-19 pandemic in Jordanian universities. *International Journal of Instruction*, *14*(3), 1011-1024. https://files.eric.ed.gov/fulltext/EJ1304581.pdf

Tsai, Y. L., & Tsai, C. C. (2020). A meta-analysis of research on digital game-based science learning. *Journal of Computer Assisted Learning*, *36*(3), 280-294. https://doi.org/10.1111/jcal.12430

Umapathy, K., & Ritzhaupt, A. D. (2017). A meta-analysis of pair-programming in computer programming courses: Implications for educational practice. *ACM Transactions on Computing Education*, *17*(4), 1–13. https://doi.org/10.1145/2996201

Yu, Z., & Deng, X. (2022). A meta-analysis of gender differences in e-learners' self-efficacy, satisfaction, motivation, attitude, and performance across the world. *Frontiers in Psychology*, *13*, 897327. https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2022.897327/full

https://dx.doi.org/10.24191/jcrinn.v9i2.481