2023 IEEE 15th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM) | 979-8-3503-8117-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/HNICEM60674.2023.10589162

Exploring Student Preference Between AI-Powered ChatGPT and Human-Curated Stack Overflow in Resolving Programming Problems and Queries

Manuel B. Garcia
FEU Institute of Technology
Manila, Philippines
mbgarcia@feutech.edu.ph

Teodoro F. Revano, Jr.
FEU Institute of Technology
Manila, Philippines
tfrevanojr@feutech.edu.ph

Renato R. Maaliw III
Southern Luzon State University
Quezon, Philippines
rmaaliw@slsu.edu.ph

Pitz Gerald G. Lagrazon
Southern Luzon State University
Quezon, Philippines
plagrazon@slsu.edu.ph

Arlene Mae C. Valderama
Jose Rizal University
Mandaluyong City, Philippines
arlene.valderama@jru.edu

Ari Happonen
LUT University
Lappeenranta, Finland
ari.happonen@lut.fi

Basit Qureshi
Prince Sultan University
Riyadh, Saudi Arabia
qureshi@psu.edu.sa

Ramazan Yilmaz
Bartin University
Bartın, Turkey
ryilmaz@bartin.edu.tr

Abstract—**Among computer programmers and developers, the user-oriented question-and-answer website of Stack Overflow is a useful platform for sourcing solutions to programming problems, exchanging insights, and accessing a wealth of shared knowledge. However, the timeliness of responses on this platform is frequently a limiting factor that ChatGPT could potentially address. The goal of this study was to explore the preferences of novice programmers between these platforms for finding answers to their programming questions. Anchored in the Technology Acceptance Model (TAM) and the Information Foraging Theory (IFT), the study investigates users' perceptions of usefulness, ease of use, information scent, cognitive effort, as well as overall preferences. Our findings show discernible variations in preferences within the group of students (i.e., application and website developers). In line with these results, we discussed theoretical and practical implications and suggested a dual-pronged approach to leverage both environments as coding assistants in computer programming education.**

Keywords—ChatGPT, Stack Overflow, Computer Programming, Large Language Models, Artificial Intelligence, Problem Solving

I. Introduction

Problem-solving is a cornerstone of computer programming education. This proficiency plays an essential role in developing students' analytical and logical thinking skills [1]. Programmers utilize this capability to deconstruct complicated challenges into manageable parts, create solutions, and implement them through coding. This process not only cultivates technical skills but also nurtures a constituted problem-solving mindset that can also be applied in various fields [2-4]. For programming students, these assets for adeptly addressing programming problems can grant them the essential tools to excel in a dynamic digital landscape. Additionally, it cultivates resilience as students learn to refine strategies, debug errors, and learn from mistakes. These skills extend beyond coding to real-world problem-solving scenarios. Amidst the shifting digital terrain, the relevance of programming problem-solving skills resonates across educational, vocational, and innovation-driven contexts. Therefore, it is a key pursuit in computer science and information technology education [5-7].

Developing strong problem-solving skills is indeed crucial for students, yet many new programmers struggle in this aspect. In such cases, they often rely on peers [8], online resources [9], or other external references when confronted with programming problems or queries. Given their limited experience in the field, they often resort to these informational reservoirs as a deliberate strategy for improving their understanding of coding concepts, troubleshooting errors, and devising solutions. From a student's perspective, these supplemental resources serve as a conduit to bridge the gap that exists between their present understanding and the multifaceted demands posed by the task at hand [10]. It becomes even more pronounced when students find themselves confronted with an array of challenges, which span beyond mere syntax-related issues to encompass broader conceptual hurdles. In this situation, computing students prefer to use programming websites [11], YouTube tutorials [12], TikTok videos [13], and even online communities like Stack Overflow [14] as their aids for resolving programming problems. Of these online platforms, Stack Overflow is one of the most popular among developers.

Despite its popularity in recent years, the emergence of large language models like ChatGPT has sparked debates within the programming community [15]. These discussions center on the potential role of these AI-powered models in verifying solutions or locating pertinent code snippets. Nevertheless, the preference of novice programmers between AI technologies like ChatGPT and human-curated resources like Stack Overflow is one of the latest ongoing investigations and discourses within the realms of software development and educational research. In this study, our goal is to contribute to the programming education literature by discerning the preferences exhibited by student programmers as they navigate the procedure of choosing online resources for programming assistance. This topic holds significance for both theoretical and practical contexts, as it imparts new insights into educational approaches and contributes to the ongoing dialogue surrounding the potential integration of AI technologies into the educational and computer programming spheres [16, 17].

979-8-3503-8117-7/23/$31.00 ©2023 IEEE

II. Background of the Study

A. Stack Overflow

Stack Overflow stands as a pivotal platform within the realm of computer programming and software development [18]. This online platform serves as a hub where programmers congregate to share knowledge, troubleshoot software bugs, and collaborate on coding challenges. Established in 2008, Stack Overflow has since emerged as a prominent destination for both novices and seasoned programmers in pursuit of knowledge, solutions, and engagement within a community of shared interests. At its core, this website operates as a question-and-answer forum specially curated for programming-related queries. The primary strength of this platform resides in its user-generated content [19]. Users frequently pose questions concerning coding challenges, and in turn, receive responses that not only tackle these challenges but also foster an environment of collaborative knowledge sharing. This collaborative ecosystem results in a compendium of best practices, valuable insights, and creative workarounds that cater to various programming languages, frameworks, and domains.

B. ChatGPT

One of the common barriers when using Stack Overflow is the potential delay in receiving responses [20]. This obstacle is a hindrance for users who are seeking immediate assistance to overcome programming problems. Conversely, students stated that a primary advantage of incorporating ChatGPT into their programming learning experience is its ability to swiftly deliver responses that are predominantly accurate in addressing their inquiries [21]. Developed by OpenAI, ChatGPT is a language model designed to engage in human-like conversations [22] and provide contextual responses across a wide range of topics [15]. Recent studies highlighted that ChatGPT can assist even people without programming knowledge in solving problems [16, 23]. Some of its most crucial uses include creating instances of code snippets, identifying flaws within provided code, suggesting the necessary enhancements for current code structures, and even autonomously correcting syntax and logical errors. Therefore, it can potentially serve as a significant auxiliary tool in coding.

C. Theoretical Frameworks

The conceptual underpinning of this study draws from both the Technology Acceptance Model (TAM) and the Information Foraging Theory (IFT). First, the core constructs of TAM, such as perceived usefulness and perceived ease of use, can provide significant insights into why students might prefer ChatGPT or Stack Overflow for programming assistance. These constructs will be leveraged to probe how students' perceptions of efficacy and ease of interaction distort their resource preferences [24]. Complementing TAM, the IFT provides a lens to examine how students navigate and evaluate information within each website platform. This theory highlights the balance between cognitive cost and information scent to guide an analysis of how students' decision-making aligns with the principles of swift information seeking [25]. Together, these theoretical frameworks illuminate the cognitive, psychological, and usability aspects of modeling student preferences, enhancing the study's understanding of technology adoption and information-seeking behavior.

III. Methods

A. Research Design

We employed a descriptive quantitative approach to identify student preference between ChatGPT and Stack Overflow in the context of resolving programming problems and queries. Using TAM and IFT as our theoretical bases, we specifically seek to understand their technology adoption and information-seeking behavior. We structured the study using a switching-replications design [26], which involved exposing students to both ChatGPT and Stack Overflow in a controlled manner. This research design facilitated a direct comparison between the two online platforms within the same group of students, which allowed us to capture nuanced preferences and variations in their experiences.

B. Participant Recruitment and Sampling

To establish a comprehensive representation of preferences, computing students were recruited from diverse programming backgrounds and academic levels. Specifically, the participants encompassed three distinct categories of novice programmers, including application developers using Java, game developers using C#, and web developers using PHP. Notably, all students were either in their second or third year of academic study. This target sample already possessed a foundational understanding of programming concepts, which rendered them well-equipped candidates for the task of assessing and juxtaposing the efficacy of the online platforms in question. This strategic participant recruitment underpins the robustness of our study's findings and contributes to the broader relevance of the research in the field of programming education and technology integration.

C. Procedures and Data Collection

Conducting two problem-solving sessions formed a crucial part of our methodology. To ensure comprehensive evaluations, students were presented with three levels of machine problems (easy, moderate, and difficult) per session [5]. In the first round, the instruction was to find solutions from Stack Overflow. For the next round, students transitioned to utilizing ChatGPT. This switching-replications approach permitted them to interact with both resources, thereby mitigating any order effects that might influence their preferences. They were also explicitly directed not to attempt problem-solving without leveraging the specified resources. Each problem-solving session spanned one hour and thirty minutes, offering ample time for participants to interact with the resources and navigate through the given problems. Following the problem-solving sessions, students were directed to complete a survey designed based on TAM and IFT.

D. Instrument and Data Analysis

The survey instrument was created based on TAM and IFT. Piloted within a programming class, the instrument underwent exploratory factor analysis for factorial validity and Cronbach's alpha analysis for reliability, with thresholds of 0.50 and 0.70, respectively. These analyses certified its validity and reliability in quantifying participants' perceptions of usefulness, ease of use, information scent, cognitive effort, and overall preferences. We used the Wilcoxon Signed-Rank Test and Kruskal-Wallis H tests to examine if there were statistically significant differences in evaluations between resources and across the groups.

TABLE I. COMPARATIVE EVALUATION OF CHATGPT AND STACK OVERFLOW ACROSS SPECIALIZATIONS

Constructs	**Within-Group Comparison**						**Between Group Comparison**	
	Application Developer		**Game Developer**		**Website Developer**			
	Mean ± SD	***p*-value**	**Mean ± SD**	***p*-value**	**Mean ± SD**	***p*-value**	χ^2	***p*-value**
Perceived Usefulness		.214		.213		.110		
Stack Overflow	5.67 ± 1.02		5.21 ± 0.97		5.56 ± 1.18		7.875	.862
ChatGPT	5.48 ± 1.13		5.38 ± 0.89		5.72 ± 1.02		7.188	.757
Perceived Ease of Use		.531		.272		.096		
Stack Overflow	5.83 ± 0.78		5.38 ± 0.89		6.41 ± 0.22		8.939	.771
ChatGPT	5.95 ± 0.63		5.64 ± 0.76		6.39 ± 0.30		9.331	.068
Information Scent		**.035**		.065		**.012**		
Stack Overflow	**5.46 ± 1.05**		5.12 ± 1.24		**5.94 ± 0.82**		7.892	.152
ChatGPT	3.89 ± 1.68		4.94 ± 1.72		5.01 ± 1.47		11.245	**.002**
Cognitive Effort		**.022**		**.031**		**.004**		
Stack Overflow	4.75 ± 1.43		5.25 ± 0.89		5.19 ± 1.38		7.051	.080
ChatGPT	**5.78 ± 0.98**		**6.12 ± 0.56**		**6.22 ± 0.43**		9.535	**.026**
Overall Preferences		**.002**		.082		**.001**		
Stack Overflow	5.28 ± 1.56		5.44 ± 1.18		**6.12 ± 0.82**		10.234	**.029**
ChatGPT	**6.10 ± 0.89**		5.92 ± 0.85		5.62 ± 1.16		9.918	**.043**

IV. RESULTS AND DISCUSSION

The primary aim of this study was to explore and analyze the preferences of students when it comes to resolving programming problems and queries. We specifically focused on comparing the Stack Overflow and ChatGPT platforms using constructs under the theoretical frameworks of TAM and IFT. By utilizing these models, we seek to uncover the underlying factors influencing student choices between these two resources. Our study seeks to provide insights into students' perceptions of usefulness, ease of use, cognitive effort, and information scent, while also assessing the overall preference for one resource over the other. Through the lenses of TAM and IFT, our research aims to contribute to a thorough understanding of students' decision-making processes and experiences when using ChatGPT and Stack Overflow for programming problem-solving. The results of our study are also intended to spark discussions on whether ChatGPT can play the role of a "knowledgeable other" in programming courses, which is in line with Vygotsky's cognitive development theory.

A. Participant Demographics

A total of 120 information technology students participated in this two-session programming problem-solving study. These students were enrolled in different specializations and randomly recruited according to their previous programming courses. The participants were consequently categorized into distinct groups. The first group consisted of application developers who passed the *Object-Oriented Programming* course and are familiar with Java. The second group consisted of game developers who knew C# programming and passed the *Game Design* course. The third group consisted of web developers who passed the *Web System Technologies* and were familiar with PHP. Each of these groups comprised 40 students with a mean age of 19.53 years. In terms of gender, most participants identified as male (n = 53, 60.55%), female (n = 53, 60.55%), and others (n = 53, 60.55%). All of the participants have prior knowledge of computer programming.

B. Between-Group Analyses

The between-group analyses showed statistically significant differences in specific constructs across the three student groups. Particularly noteworthy were the findings involving their overall preferences, which proved substantial differences for both Stack Overflow ($\chi^2 = 10.234$; $p = .029$) and ChatGPT ($\chi^2 = 9.918$; $p = .043$). This significant divergence in preference signals that the decision to pick one platform over the other varied considerably among computing students. This outcome has profound effects, suggesting that their preferences are influenced not only by the unique features and qualities of each platform but also by their specializations and programming backgrounds [27]. It not only highlights the intricate nature of platform selection within the diverse spectrum of developer roles but also paves the way for future research into exploring the distinctions among these roles.

On the other hand, the construct of cognitive effort displayed a significant difference ($\chi^2 = 9.535$; $p = .026$) when students used ChatGPT. This discrepancy indicates that they faced contrasting degrees of cognitive effort while operating ChatGPT compared to Stack Overflow. Importantly, it is important to acknowledge that ChatGPT's responses may not consistently align with users' expectations, and at times, may be incorrect [23, 28]. According to a recent assessment [29], it has been observed that ChatGPT can sometimes generate inaccurate codes, semantically incorrect outputs, or syntactically invalid responses. This factor could be a contributing factor to the observed fluctuations in cognitive effort among students. From a learning standpoint, this issue is particularly problematic to introductory programming students who do not possess the necessary self-regulation skills [1]. This implication stresses the importance of understanding cognitive load as an influential factor in the overall user experience. It also opens avenues for further investigation into the role of cognitive processes underlying platform interaction among students.

Furthermore, the analysis unveiled a substantial variance in the construct of information scent ($\chi^2 = 11.245$; $p = .002$). This finding emphasizes ChatGPT's superiority over Stack Overflow in facilitating students' discovery of relevant information. This outcome is intriguing as it indicates that they found ChatGPT to be more effective in directing them to the codes they are looking for compared to the broadly used Stack Overflow. A reasonable explanation for this discovery is the natural language processing capabilities of ChatGPT, which may have contributed to a more intuitive and user-friendly information retrieval process. For the practical implications of this result, it underlines the pivotal role of information accessibility in platform design. Particularly, the accessibility of responses is more crucial within domains where rapid and precise information retrieval is a condition. Students' ability to locate appropriate resources significantly impacts their productivity and problem-solving efficiency. ChatGPT's ability to excel in this aspect features the potential of leveraging natural language processing to further information-seeking experiences.

C. *Within-Group Analysis*

Examining the ratings of application developers, the analysis unveiled a noteworthy discrepancy in terms of information scent ($p = .035$) between ChatGPT (3.89 ± 1.68) and Stack Overflow (5.46 ± 1.05). This finding suggests that application developers experienced contrasting degrees of effectiveness when pursuing relevant information on these two platforms. The origin of this distinction could be rooted in the differing ways in which these platforms provide information in response to queries. Moreover, the significant difference in cognitive effort ($p = .022$) highlights divergent cognitive demands connected to using ChatGPT (5.78 ± 0.98) and ChatGPT (4.75 ± 1.43). This distinction in cognitive effort could be influenced by the nature of the platforms. While Stack Overflow relies on traditional search, ChatGPT employs artificial intelligence and natural language interactions. Overall preferences among this group are also significantly different ($p = .002$), which highlights the need for tailored platform design to accommodate the specific cognitive and preferential aspects of application development using Java programming language.

Among game developers, only the cognitive effort construct demonstrated a significant difference ($p = .031$) between Stack Overflow (5.25 ± 0.89) and ChatGPT (6.12 ± 0.56). This result suggests that game developers encountered cognitive challenges when utilizing these platforms. One possible explanation for this divergence is rooted in the multifaceted characteristics of game development tasks, which often extend beyond coding and entail a range of intricate and task-extensive activities. Unlike some programming tasks, game development may require abstracting and implementing complex gameplay mechanics [30], character behaviors [31, 32], and immersive environments [33, 34]. These multifaceted tasks demand holistic thinking, problem-solving, and creativity which influences the evident cognitive processing demands when seeking information. Given the diverse nature of game development, the cognitive effort required to engage with support platforms could differ considerably. Game developers might need to explore a wide array of resources to address these challenges spanning from coding issues to narrative elements. It also encourages future research to delve into this topic.

Lastly, the results revealed a series of significant differences among the group of web developers. First, the information scent construct yielded a significant difference ($p = .012$), indicating variability in their ability to locate relevant information on Stack Overflow (5.94 ± 0.82) and ChatGPT (5.01 ± 1.47). Similar to the other groups, this observation suggests that web developers experienced varying degrees of efficiency when searching for pertinent information across these platforms. On the other hand, the significant difference in cognitive effort ($p = .004$) indicated differing cognitive demands associated with the use of the two platforms. This result is consistent with the ratings given by the other groups. Collectively, these findings illuminate the diverse techniques in which website developers interact with platform features, each carrying its own cognitive implications.

D. *Implications and Limitations*

In summary, the comprehensive analysis that encompasses both between-group and within-group comparisons collectively unveils a multifaceted picture of student developer interactions with platforms in the domain of information technology. These insights hold substantial implications for platform design, user experiences, and the broader understanding of the relationships between developer roles and platform attributes. Theoretical and practical implications from the educational perspective are also apparent. Specifically, these implications are salient in terms of curriculum development, instructional strategies, and the overall academic journey for aspiring IT professionals.

First, the observed variations in platform preferences as well as cognitive processes and task-specific demands across cohorts of novice programmers highlight the critical role of customizing educational curricula. This awareness highlights the imperative of tailored educational paths that cater to the nuanced needs of individual learners in preparation for real-world challenges [35]. By mirroring the cognitive aspects and preferences relevant to various developer roles, educators equip students with skills and knowledge that not only align with the realities of the industry but also resonate with their individual aptitudes and inclinations. Moreover, this tailored approach to curriculum design promotes a deeper engagement level among students. When they perceive that their learning experiences are accustomed to their cognitive strengths and preferences, they are more likely to be motivated, attentive, and proactive in their learning journey. Consequently, it enhances the effectiveness of educational tools by capitalizing on the intrinsic motivations of students [36, 37].

In the context of information technology education, one of the key takeaways from our study is the significance of cognitive skill development tailored to platform interaction. Recognizing the cognitive processes necessary for effective engagement with different platforms, such as ChatGPT and Stack Overflow, may provide a foundation for educators to nurture these skills from the early stages of a student programmer's learning journey. For instance, incorporating hands-on exercises that require students to interact with both platforms can provide firsthand experience in navigating their distinct cognitive landscapes. Students might practice formulating queries in natural language for ChatGPT or refining search queries for Stack Overflow. By immersing them in scenarios that replicate real-world challenges, educators can bridge the gap between theoretical knowledge and practical application necessary for a successful career in this field.

Platform integration within the educational framework holds significant potential as well for preparing students in the field of information technology. By strategically introducing a diverse array of tools, particularly those that mirror industry practices, educators can enrich the learning experience in multifaceted ways. This approach not only bridges the gap between academia and real-world application but also nurtures the development of adaptable skills that are essential for success in the technology field. In the context of the two platforms under investigation, it may be beneficial if students would use them in ways similar to how they might be utilized by programming professionals. The motivational theory of role modeling has been recognized to be effective among novices [1]. This exposure empowers students to discern which coding support platform aligns better with their cognitive strengths and preferences. As they encounter different platforms with exceptional cognitive demands, they develop the capacity to acquire new skills, adapt to evolving technologies, and remain effective problem solvers throughout their careers.

Another fundamental implication of our findings within the field of education is the need to foster holistic skill development among computing students. As we delve into the intricacies of platform usage, the substance of cultivating a well-rounded skill set becomes gradually evident. This aspect extends beyond mere technical competencies and encapsulates a broader spectrum of abilities that are indispensable for thriving in the ever-evolving landscape of information technology. In the context of the two platforms, this holistic skill development pertains to more than just mastering the mechanics of these tools. Additionally, what we advocate extends to the decision-making aspect. As students interact with platforms to seek programming information, solve coding problems, or simply enhance their understanding of code snippets, making informed choices based on individual likings come to the fore. Developing this discernment equips students with a valuable skill that transcends the immediate task at hand and extends to their overall growth as professionals [38, 39].

It should be taken into consideration that this research has some limitations. Firstly, the research was conducted as a two-session programming problem-solving study, and then student preferences were determined. In future studies, research can be planned to determine student preferences after a longer period of exposure. Another limitation of the research is related to the participants. The research participants are students with a certain amount of experience in computer programming education. In future studies, research can be done to determine the preferences of inexperienced and expert programmers. Another limitation of the study is that the participants are university students. Future research can also be conducted to determine the preferences of programmers among different ages and education levels.

V. Conclusion

This study aimed to elucidate student preferences between ChatGPT and Stack Overflow. Our findings discovered distinct tendencies among students, revealing that ChatGPT is favored by application developers while web developers preferred Stack Overflow. In addition, our evaluation proposes that ChatGPT's strengths lie in its capacity to swiftly provide answers, facilitate idea exchange, and generate individualized learning experiences when addressing programming problems. These characteristics appear to significantly influence students' general preferences. However, it is notable that ChatGPT's responses may sometimes veer towards being inaccurate or misleading, which could prove challenging for novice programmers seeking precise and reliable solutions. In contrast, Stack Overflow's structured question-and-answer format fosters clarity and comprehension, enabling users to articulate and understand problems effectively. The platform's extensive community of expert programmers offers invaluable support, particularly for those new to computer programming. Their repository of questions and answers serves as a reservoir of knowledge, aiding future users facing similar challenges. The inclusion of code examples in Stack Overflow answers further facilitates understanding and learning, providing novices with tangible illustrations of problem-solving approaches.

Given the nuanced advantages of both platforms, the study recommends a dual-pronged approach. In individual learning processes, students can capitalize on the strengths of ChatGPT for solving programming problems. Nevertheless, recognizing its limitations, students are likewise encouraged to maximize the collaborative environment of Stack Overflow for more complex challenges. Engaging with peers in Stack Overflow's community permits novices to address intricate issues collectively, thereby nurturing the cultivation of programming knowledge and skill development. In essence, the research underscores the value of a hybrid strategy by leveraging the unique attributes of ChatGPT and Stack Overflow for more effective learning in programming.

References

[1] M. B. Garcia, "Facilitating Group Learning Using an Apprenticeship Model: Which Master is More Effective in Programming Instruction?," *Journal of Educational Computing Research,* 2023, doi: 10.1177/07356331231170382.

[2] J. Kay *et al.*, "Problem-Based Learning for Foundation Computer Science Courses," *Computer Science Education,* vol. 10, no. 2, pp. 109-128, 2000, doi: 10.1076/0899-3408(200008)10:2;1-C;FT109.

[3] S. B. Fee and A. M. Holland-Minkley, "Teaching Computer Science Through Problems, Not Solutions," *Computer Science Education,* vol. 20, no. 2, pp. 129-144, 2010, doi: 10.1080/08993408.2010.486271.

[4] J. Kiesewetter, R. Ebersbach, A. Görlitz, M. Holzer, M. R. Fischer, and R. Schmidmaier, "Cognitive Problem Solving Patterns of Medical Students Correlate with Success in Diagnostic Case Solutions," *PLOS ONE,* vol. 8, no. 8, pp. 1-8, 2013, doi: 10.1371/journal.pone.0071486.

[5] M. B. Garcia, J. B. Enriquez, R. Adao, and A. Happonen, "'Hey IDE, Display Hello World': Integrating a Voice Coding Approach in Hands-on Computer Programming Activities," in *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2022, doi: 10.1109/HNICEM57413.2022.10109412.

[6] Z. Liu, R. Zhi, A. Hicks, and T. Barnes, "Understanding Problem Solving Behavior of 6–8 Graders in a Debugging Game," *Computer Science Education,* vol. 27, no. 1, 2017, doi: 10.1080/08993408.2017.1308651.

[7] I. Kožuh, R. Krajnc, L. J. Hadjileontiadis, and M. Debevc, "Assessment of Problem Solving Ability in Novice Programmers," *PLOS ONE,* vol. 13, no. 9, pp. 1-21, 2018, doi: 10.1371/journal.pone.0201919.

[8] M. B. Garcia, "Cooperative Learning in Computer Programming: A Quasi-Experimental Evaluation of Jigsaw Teaching Strategy with Novice Programmers," *Education and Information Technologies,* vol. 26, no. 4, pp. 4839-4856, 2021, doi: 10.1007/s10639-021-10502-6.

[9] R. P. Bringula, "Influence of Usage of E-Books, Online Educational Materials, and Other Programming Books and Students' Profiles on Adoption of Printed Programming Textbooks," *Program,* vol. 51, no. 4, pp. 441-457, 2017, doi: 10.1108/PROG-06-2015-0046.

[10] M. B. Garcia and T. F. Revano, "Assessing the Role of Python Programming Gamified Course on Students' Knowledge, Skills Performance, Attitude, and Self-Efficacy," in *2021 IEEE 13th*

International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management, 2021, doi: 10.1109/HNICEM54116.2021.9731935.

[11] S. Lausa, R. Bringula, A. E. Catacutan-Bangit, and C. Santiago, "Information-Seeking Behavior of Computing Students while Programming: Educational Learning Materials Usage, Satisfaction of Use, and Inconveniences," in *2021 IEEE Global Engineering Education Conference*, 2021, doi: 10.1109/EDUCON46332.2021.9453905.

[12] A. Kadriu, L. Abazi-Bexheti, H. Abazi-Alili, and V. Ramadani, "Investigating Trends in Learning Programming Using YouTube Tutorials," *International Journal of Learning and Change,* vol. 12, no. 2, pp. 190-208, 2020, doi: 10.1504/IJLC.2020.106721.

[13] M. B. Garcia, I. C. Juanatas, and R. A. Juanatas, "TikTok as a Knowledge Source for Programming Learners: a New Form of Nanolearning?," in *2022 10th International Conference on Information and Education Technology (ICIET)*, 2022, doi: 10.1109/ICIET55102.2022.9779004.

[14] P. Dondio and S. Shaheen, "Is StackOverflow an Effective Complement to Gaining Practical Knowledge Compared to Traditional Computer Science Learning?," in *11th International Conference on Education Technology and Computers*, Amsterdam, Netherlands, 2020: Association for Computing Machinery, pp. 132–138, doi: 10.1145/3369255.3369258.

[15] S. Kabir, D. N. Udo-Imeh, B. Kou, and T. Zhang, "Who Answers It Better? An In-Depth Analysis of ChatGPT and Stack Overflow Answers to Software Engineering Questions," *arXiv,* 2023, doi: 10.48550/arXiv.2308.02312.

[16] R. Yilmaz and F. G. Karaoglan Yilmaz, "The Effect of Generative Artificial Intelligence (AI)-Based Tool Use on Students' Computational Thinking Skills, Programming Self-Efficacy and Motivation," *Computers and Education: Artificial Intelligence,* vol. 4, pp. 1-14, 2023, doi: 10.1016/j.caeai.2023.100147.

[17] C. A. Philbin, "Exploring the Potential of Artificial Intelligence Program Generators in Computer Programming Education for Students," *ACM Inroads,* vol. 14, no. 3, pp. 30-38, 2023, doi: 10.1145/3610406.

[18] P. Chatterjee, M. Kong, and L. Pollock, "Finding Help with Programming Errors: An Exploratory Study of Novice Software Engineers' Focus in Stack Overflow Posts," *Journal of Systems and Software,* vol. 159, pp. 1-13, 2020, doi: 10.1016/j.jss.2019.110454.

[19] M. Tavakoli, M. Izadi, and A. Heydarnoori, "Improving Quality of a Post's Set of Answers in Stack Overflow," in *2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2020, pp. 504-512, doi: 10.1109/SEAA51224.2020.00084.

[20] T. Bhasin, A. Murray, and M. A. Storey, "Student Experiences with GitHub and Stack Overflow: An Exploratory Study," in *2021 IEEE/ACM 13th International Workshop on Cooperative and Human Aspects of Software Engineering*, 2021, doi: 10.1109/CHASE52884.2021.00017.

[21] R. Yilmaz and F. G. Karaoglan Yilmaz, "Augmented Intelligence in Programming Learning: Examining Student Views on the Use of ChatGPT for Programming Learning," *Computers in Human Behavior: Artificial Humans,* vol. 1, no. 2, 2023, doi: 10.1016/j.chbah.2023.100005.

[22] M. B. Garcia, "Can ChatGPT Substitute Human Companionship for Coping with Loss and Trauma?," *Journal of Loss and Trauma,* pp. 1-3, 2023, doi: 10.1080/15325024.2023.2240697.

[23] N. M. S. Surameery and M. Y. Shakor, "Use ChatGPT to Solve Programming Bugs," *International Journal of Information technology and Computer Engineering,* vol. 3, no. 1, pp. 17-22, 2023, doi: 10.55529/ijitc.31.17.22.

[24] A. S. Mustafa and M. B. Garcia, "Theories Integrated With Technology Acceptance Model (TAM) in Online Learning Acceptance and Continuance Intention: A Systematic Review," in *2021 1st Conference on Online Teaching for Mobile Education (OT4ME)*, 2021, pp. 68-72, doi: 10.1109/OT4ME53559.2021.9638934.

[25] X. Shi, X. Zheng, and F. Yang, "Exploring Payment Behavior for Live Courses in Social Q&A Communities: An Information Foraging Perspective," *Information Processing & Management,* vol. 57, no. 4, pp. 1-16, 2020, doi: 10.1016/j.ipm.2020.102241.

[26] M. B. Garcia, L. S. Nadelson, and A. Yeh, "'We're going on a virtual trip!': A Switching-Replications Experiment of 360-Degree Videos as a Physical Field Trip Alternative in Primary Education," *International Journal of Child Care and Education Policy,* vol. 17, no. 4, pp. 1-16, 2023/01/25 2023, doi: 10.1186/s40723-023-00110-x.

[27] X. Zhu, Y. Yue, and S. Chen, "Impact of Major Backgrounds on Student Learning Achievement: A Case Study for Java Programming Course," *Education Sciences*, vol. 13, no. 2, pp. 1-11, 2023, doi: 10.3390/educsci13020127.

[28] M. B. Garcia, "Using AI Tools in Writing Peer Review Reports: Should Academic Journals Embrace the Use of ChatGPT?," *Annals of Biomedical Engineering*, 2023, doi: 10.1007/s10439-023-03299-7.

[29] S. E. Arefin, T. A. Heya, H. Al-Qudah, Y. Ineza, and A. Serwadda, "Unmasking the Giant: A Comprehensive Evaluation of ChatGPT's Proficiency in Coding Algorithms and Data Structures," *arXiv*, 2023, doi: 10.48550/arXiv.2307.05360.

[30] M. B. Garcia et al., "Promoting Social Relationships Using a Couch Cooperative Video Game: An Empirical Experiment With Unacquainted Players," *International Journal of Gaming and Computer-Mediated Simulations (IJGCMS)*, vol. 14, no. 1, pp. 1-18, 2022, doi: 10.4018/IJGCMS.303106.

[31] T. Luluquisin et al., "Beast Chasers: A 3D PC-based Third Person Action RPG Game used to Spread Societal Issue Awareness," in *2021 the 11th International Workshop on Computer Science and Engineering (WCSE)*, 2021, doi: 10.18178/wcse.2021.06.030.

[32] P. A. Arayata, J. Banzon, B. M. Franco, D. Lubrin, M. R. Perez, and M. B. Garcia, "Chyilax: An Innovative 3D Game Approach for Mental Breakdown Awareness Campaign," in *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2022, doi: 10.1109/HNICEM57413.2022.10109535.

[33] L. P. Cortez et al., "Corona Larona: A 2.5D Mobile Game Advocating COVID-19 Safety Protocols and Mitigation Strategies," in *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management*, 2022, doi: 10.1109/HNICEM57413.2022.10109445.

[34] D. Parel et al., "Escape from Oblivion: A 3D Hack and Slash Survival Horror Video Game for Promoting Awareness of Persistent Depressive Disorder," in *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2022, doi: 10.1109/HNICEM57413.2022.10109542.

[35] T. F. Revano Jr and M. B. Garcia, "Manufacturing Design Thinkers in Higher Education Institutions: The Use of Design Thinking Curriculum in the Education Landscape," in *2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, 2020, pp. 1-5, doi: 10.1109/HNICEM51456.2020.9400034..

[36] M. B. Garcia, "Hackathons as Extracurricular Activities: Unraveling the Motivational Orientation Behind Student Participation," *Computer Applications in Engineering Education*, vol. 30, no. 6, pp. 1903-1918, 2022/11/01 2022, doi: 10.1002/cae.22564.

[37] S. Mohanarajah, "Increasing Intrinsic Motivation of Programming Students: Towards Fix and Play Educational Games," *Issues in Informing Science and Information Technology*, vol. 15, pp. 69-77, 2018, doi: 10.28945/4027.

[38] A. M. Valderama, J. B. Tuazon, and M. B. Garcia, "Promoting Student Thinking and Engagement Through Question-Based and Gamified Learning," in *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, 2022, doi: 10.1109/HNICEM57413.2022.10109470.

[39] A. S. Mustafa, G. A. Alkawsi, K. Ofosu-Ampong, V. Z. Vanduhe, M. B. Garcia, and Y. Baashar, "Gamification of E-Learning in African Universities: Identifying Adoption Factors Through Task-Technology Fit and Technology Acceptance Model," in *Next-Generation Applications and Implementations of Gamification Systems*, F. Portela and R. Queirós Eds. USA: IGI Global, 2022, doi: 10.4018/978-1-7998-8089-9.ch005.