

Seasharp: Gamifying C# Programming Education for Engaging Learning

Jeazabel B. Cabarlo, Carlo Genster P. Camposagrado, Alvin A. Casipit, Lyka P. Casipit, John Jay S. Suniga

Information Technology, Pangasinan State University, Alaminos City Campus

DOI : <https://doi.org/10.51583/IJLTEMAS.2025.1401033>

Received: 31 January 2025; Revised: 07 February 2025; Accepted: 18 February 2025; Published: 18 February 2025

Abstract– This study introduces SeaSharp, a gamified web-based platform designed to enhance C# programming education for Information Technology students at Pangasinan State University, Alaminos City Campus. By incorporating immersive narratives, interactive coding challenges, and game mechanics such as levels, rewards, and leaderboards, SeaSharp aims to improve student engagement, motivation, and comprehension. The platform is designed using a User-Centered Design approach and integrates the Mechanics, Dynamics, and Aesthetics framework to ensure a structured and engaging learning experience. This study presents main objectives include developing a compelling storyline, implementing innovative game mechanics, and integrating gamification principles to create an interactive and personalized learning environment. Additionally, this paper presents the acceptance level of the platform among students. Surveys and usability testing conducted with both students and instructors revealed significant improvements in motivation, participation, and problem-solving skills. Respondents highlighted the platform's accessibility, interactivity, and ability to cater to diverse educational needs. By making complex programming concepts more engaging and intuitive, SeaSharp provides an effective alternative to traditional teaching methods. The integration of immediate feedback, structured progression, and interactive challenges fosters a deeper understanding of C# programming. The platform's success underscores the transformative potential of gamification in programming education. Future recommendations include expanding SeaSharp to cover advanced programming topics and integrating real-time analytics for adaptive feedback, further optimizing learning outcomes.

Keywords – gamification, C# programming, education technology, student engagement, user-centered design

I. Introduction

Gamification is a controversial topic in the field of education, with both proponents and opponents (Kittichai Nilubol, et.al, 2023). In online learning environments, gamification serves multiple purposes, such as providing timely feedback, fostering a fun and engaging atmosphere, and increasing motivation and student participation. By incorporating game-like elements, gamification has been shown to attract students' attention to course content, enhancing their learning experience and promoting a deeper connection to the material. Research suggests that gamification increases the time students spend in the online environment, boosts course completion rates, and lowers dropout rates (Castro et al., 2018; Looyestyn et al., 2017). These findings indicate that gamification holds promise for improving student outcomes in flipped learning environments, where active participation and engagement are key.

The combination of gamification and flipped learning is increasingly seen as a powerful tool for enhancing the learning and teaching process (Pozo Sánchez et al., 2020). As both of these approaches focus on promoting active learning and engaging students in a more personalized way, they align well to foster motivation, deeper learning, and improved academic achievement. Personalized gamification designs, which take into account individual learning preferences and cognitive abilities, have become a trend in recent years (Santos et al., 2021). By tailoring gamified educational experiences to the unique needs of each student, educators can create more effective and engaging learning environments that help learners achieve better outcomes.

One of the most significant challenges in education, particularly in programming, is keeping students motivated and engaged with complex, abstract concepts. This challenge is especially prominent in programming languages like C#, which require a deep understanding of technical skills and logical thinking. Gamification has proven to be an effective strategy in overcoming these challenges, particularly in the realm of teaching programming (Marín et al., 2018). By transforming the learning process into an interactive and rewarding experience, gamification has the potential to turn what might be a daunting subject into an engaging and enjoyable activity.

Studies highlight numerous benefits of gamification in education, including increased access to learning materials, enhanced flexibility, and the promotion of critical thinking skills, all of which result in better student performance (Vanduhe et al., 2019). Gamification fosters motivation, collaboration, and increased participation, creating an environment that encourages continuous learning and achievement. Elements like rewards, levels, and challenges not only engage students but also enhance their focus, problem-solving abilities, and ability to work within a team. The competitive and collaborative nature of gamified learning encourages students to strive for improvement and embrace challenges.

The SeaSharp project, which aims to gamify C# programming education, seeks to revolutionize traditional teaching methods by

integrating game elements into the learning process. Compared to conventional education models, which often rely on lectures, textbooks, and practical exercises, SeaSharp offers an interactive and immersive learning experience that keeps students engaged and motivated (Smith & Jones, 2018). By blending structured learning with game elements such as points, rewards, and character progression, SeaSharp enhances student engagement and motivation (Gonzalez et al., 2020). This gamified approach not only makes programming more accessible but also encourages learners to actively participate and take ownership of their learning journey.

Additionally, SeaSharp integrates a narrative-driven experience, where learners embark on a thematic journey, solving coding challenges that build on each other and contribute to an overarching story. This approach transforms learning into an adventure, increasing students' investment in the material and making the learning process more memorable (Jayasinghe, 2022). Through progressive levels and challenges, SeaSharp ensures that learners are exposed to coding concepts in a structured and gradual manner. This method encourages students to think critically and develop their programming skills while maintaining their interest and enthusiasm for the subject.

This study focuses on developing SeaSharp, a gamified web-based platform to enhance C# programming education at Pangasinan State University, Alaminos City Campus. Traditional programming education can be challenging, affecting student engagement and comprehension. SeaSharp integrates gamification to create an interactive and enjoyable learning experience, making C# more accessible.

Pangasinan State University Alaminos City Campus offers quality IT education, but other Bachelor of Science in Information Technology students struggle with complex programming concepts. SeaSharp supports first- and second-year students by making learning engaging and effective. Through collaboration, this project aims to revolutionize programming education and set a new standard for interactive learning.

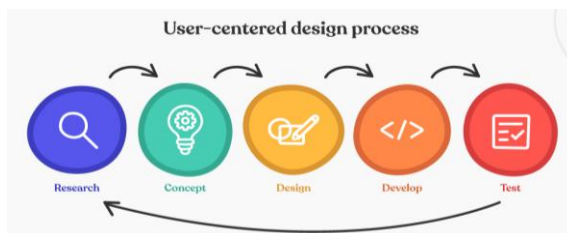
The authors aim to address these issues through the design of a gamified web-based platform called "SeaSharp." This platform is specifically developed to enhance C# programming education for students at Pangasinan State University, Alaminos City Campus. It will incorporate gamification principles to make learning more engaging and interactive, helping students improve their programming skills in a more enjoyable and effective way.

II. Methodology

The research design for SeaSharp incorporates a combination of User-Centered Design (UCD) and descriptive approaches, both of which were selected to ensure that the platform meets the needs of its users while being flexible enough to adapt to their feedback throughout the development process. The UCD methodology focuses on understanding user needs, preferences, and behaviors through user research methods such as interviews, surveys, and usability testing. This approach emphasizes iterative prototyping and refinement to align the game design with user expectations, ensuring an engaging and effective learning experience. The descriptive approach complements UC by utilizing qualitative and quantitative methods to gather in-depth insights into the platform's development and user interaction, providing a more holistic understanding of the system's impact. While UCD focuses on continuous, iterative user feedback to inform design improvements, the descriptive approach allows for deeper analysis of user behavior and satisfaction, providing a comprehensive understanding of the platform's effectiveness and the learning process.

The development of SeaSharp follows the UCD principles, which include phases such as research, concept, design, development, and testing. This approach allows for a flexible and responsive development cycle, adapting to user feedback and evolving project goals. The iterative nature of UCD ensures continuous improvement, with prototypes of game mechanics, interfaces, and interactions being tested and refined based on real user feedback. Collaboration with stakeholders and cross-functional teams plays a key role, ensuring that diverse perspectives contribute to the game's design and development. This iterative cycle is visually represented in Figure 1, which outlines the stages of the User-Centered Design process. Figure 1 shows the User centered Design process.

Figure 1. User-centered design process



The research was conducted at Pangasinan State University, Alaminos City Campus, during the academic year 2024-2025. The study employed a purposive sampling method to select participants aligned with the research objectives. The primary participants included first- and second-year student from the Bachelor of Science in Information Technology (BSIT) program As shown in Table I, these students were chosen to evaluate the SeaSharp platform's effectiveness and usability due to their foundational knowledge of programming and relevance as the intended user base. Additionally, IT instructors were involved to provide expert

insights into the platform's impact on teaching and learning outcomes, further enriching the study.

Table I. Respondents of the Study

Respondents	Number of Respondents
1st year students in IT	45
2nd year students in IT	106
IT Instructor	2
Total	153

Data collection involved a combination of observation, questionnaires, interviews, and internet research. The observation provided real-time insights into user interactions with the platform, while questionnaires gathered quantitative data on user satisfaction, perceived impact, and suggestions for improvements. Interviews with instructors and students offered qualitative insights into their experiences and expectations. Internet research supplemented these findings with additional information on best practices in gamified learning platforms and C# programming education.

To assess the platform's acceptability, a survey questionnaire was adapted from the Technology Acceptance Model (TAM) by Davis (1989). As shown in Table II, the survey used a 5-point Likert Scale to measure responses, where 5 indicated 'Strongly Agree' and 1 indicated 'Strongly Disagree.' The collected data were analyzed using weighted means to prioritize areas for development based on user feedback.

Table II. The Scale of Measurement for Acceptance Test

Scale	Statistical Range	Equivalent	Interpretation
1	1.00 – 1.80	Poor	Not
2	1.81 – 2.60	Fair	Not
3	2.61 – 3.40	Good	Accepted
4	3.41 – 4.20	Very Good	Accepted
5	4.21 – 5.00	Excellent	Accepted

Statistical treatment was applied to process the collected data, which included drawing conclusions, identifying statistical significance, and ensuring generalizability. Additionally, internet research provided related information on gamified learning platforms and C# programming education, while document analysis of course syllabi and curriculum guidelines ensured that SeaSharp was aligned with academic standards and student needs.

The SeaSharp platform was developed using a combination of tools to enhance functionality and user engagement. The Unity Game Engine was employed to create an immersive and interactive environment, while Microsoft Visual Studio Code supported coding and debugging. Visual assets were designed using Adobe Photoshop, and Blender was used to create 3D models and animations, improving the visual appeal of the platform. These tools collectively contributed to building a robust and engaging gamified learning experience.

III. Results and Discussion

Gamified learning makes programming more engaging and effective. SeaSharp is a web-based platform designed to enhance C# programming education at Pangasinan State University, Alaminos City Campus. With interactive challenges and immersive storytelling, students can develop coding skills while enjoying a game-like experience. The following discussions cover the proposed system framework, flowchart and game mechanics.

To guide the development and design process effectively, the study incorporated the Mechanics, Dynamics, and Aesthetics (MDA) framework into the project framework. The MDA framework bridges game design, criticism, and research, allowing developers to decompose and analyze the game's elements iteratively. By integrating MDA, the study ensured a comprehensive alignment between the game mechanics (rules and systems), dynamics (user interactions), and aesthetics (emotional responses). This combination provided a structured approach to understanding how the platform's features and design contribute to an engaging and educational experience for its users.

The MDA framework consists of three components: Mechanics, Dynamics, and Aesthetics. Mechanics are the rules and basic components of the game, Dynamics represent the run-time behavior of the mechanics acting on player inputs and outputs, and Aesthetics refer to the emotional responses evoked in players. Integrating these elements into the project management framework ensures a comprehensive understanding of the game creation process, leading to more refined and successful outcomes. Figure 2 illustrates the MDA framework model used in this integration.

The MDA framework directly influenced key design decisions in SeaSharp, such as the implementation of level gating and customization options, which were chosen to create a balanced and engaging learning flow while maintaining a sense of achievement and player agency.

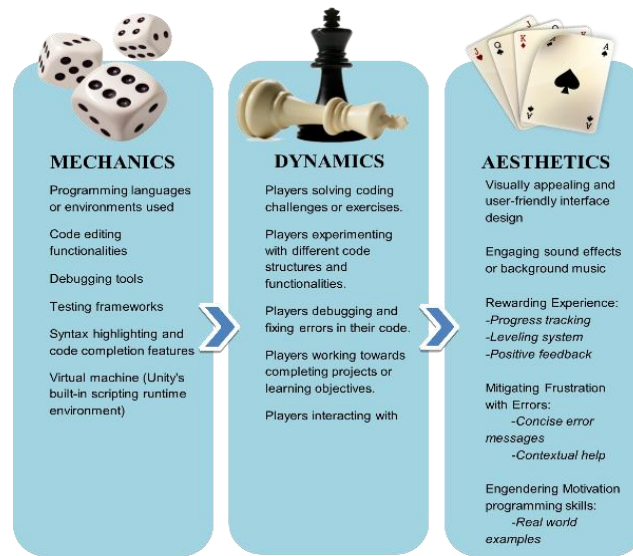
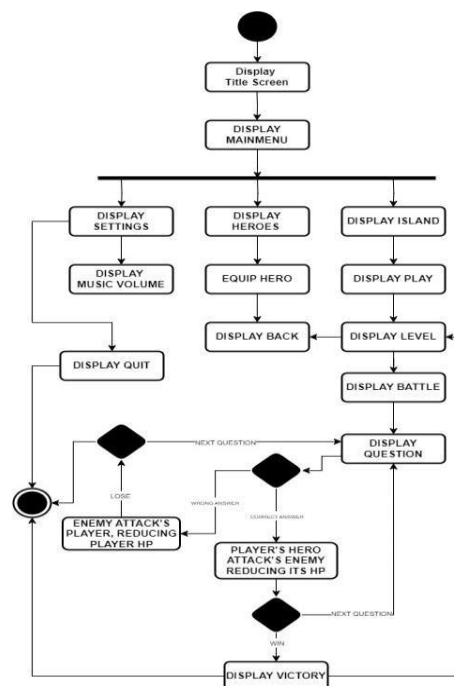


Figure 2. Mechanics-Dynamics-Aesthetics Framework Model

In the proposed integration of MDA into the project management framework, developers begin by defining the game mechanics, which are the foundational rules and systems of the game. These mechanics are then tested and refined during development to ensure they function as intended. The dynamics are observed through playtesting, which allows developers to see how players interact with the mechanics and how the game environment evolves. This iterative process helps identify any issues or opportunities for improvement.

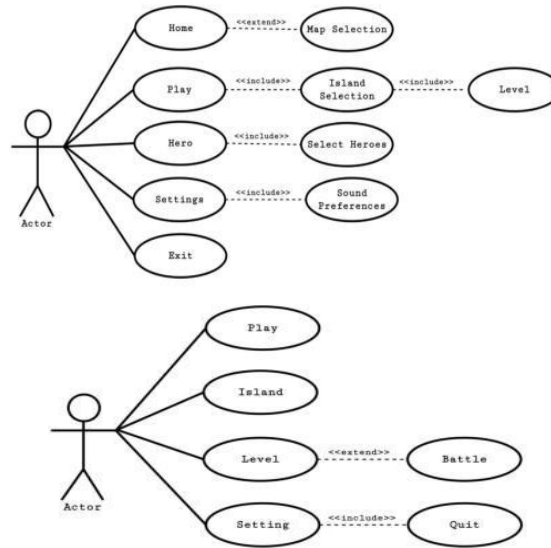
To visualize system interactions and user progression, flowcharts were used to map out key processes. These flowcharts depicted step-by-step procedures that guide users through the game's levels, challenges, and feedback mechanisms as shown in Figure 3. By providing a clear representation of user progression and decision-making pathways, the flowcharts ensured that the platform was intuitive and easy to navigate.

Figure. 3 Flowchart



Additionally, a use case diagram mapped out the major functionalities and services of SeaSharp, illustrating the relationship between actors (users) and the system as shown in Figure 4. This model helped clarify system operations and served as a guide for future design and implementation.

Fig. 4 Use Case Diagram



The game mechanics module manages the interactive features of SeaSharp, ensuring an engaging learning experience for students. This module includes key mechanics such as the battle coding quiz and a leveling system. It also handles preparatory lesson modules that introduce programming concepts before challenges. As shown in Figure 5, and 6 SeaSharp integrates gamified elements like an damage mechanics, and battle coding quiz to make C# programming education more dynamic and enjoyable.

Figure 5. Damage Mechanics



SeaSharp's storyline uses coding as a core combat mechanism. As the Hero encounters various creatures, each correct answer to a C# coding challenge results in an attack on the enemy. Wrong answers allow the enemy to retaliate, draining the Hero's health. Clearing these battles and coding tasks brings the Hero closer to the final confrontation with the Sea Demon King, blending combat with educational coding challenges.

Figure 6. Battle Coding Quiz



IV. Conclusions

The SeaSharp project effectively enhances the learning experience for students by integrating gamification into C# programming education. By incorporating game elements like points, levels, and rewards, the system increases student engagement and motivation, making complex programming concepts more accessible and enjoyable. The SeaSharp platform, designed specifically for students at Pangasinan State University, ensures personalized and interactive learning environments that resonate with today's tech-savvy students. The user-centered design approach, involving extensive surveys, interviews, and usability testing, ensures that the platform meets the diverse needs of learners. With features like immediate feedback and a structured yet dynamic framework, SeaSharp empowers students to navigate the complexities of C# programming confidently and enthusiastically. The successful deployment and positive feedback from students and instructors highlight the platform's potential to revolutionize programming education, setting a benchmark for similar educational initiatives.

Acknowledgement

The author would like to express sincere gratitude to their parents for their unwavering support, encouragement, and belief in this work. Their guidance has been a constant source of strength throughout this journey.

Appreciation is also extended to friends whose motivation and insightful discussions have provided inspiration and encouragement along the way.

The author is deeply grateful to the editors and all those involved in the production of this work. Their dedication and expertise have played a crucial role in shaping this paper. Special thanks also go to the publishers for their support in bringing this research to a wider audience.

A heartfelt acknowledgment is given to the contributors, advisers, and scholars whose research has informed and enriched this study, including Albion & Ertmer (2019), Alha et al. (2020), Alhammad & James (2021), Anderson & Shneiderman (2019), and many others whose work has provided valuable insights.

The author would also like to extend gratitude to teachers, mentors, and professional guides for their wisdom and support. Their encouragement has been instrumental in fostering both academic and professional growth.

The author acknowledges all the inspirations—past and present—that have contributed to the development of this research. Every challenge, idea, and discovery has played a role in shaping this work.

References

1. Albion, P. R., & Ertmer, P. A. (2019), The development of an instrument to measure preservice teachers' technology self-efficacy. *Journal of Research on Technology in Education*
2. Alha, K., et al. (2020), Well-rounded character development in storytelling games. *Journal of Gaming Studies*
3. Alhammad, M., & James, A. (2021), Enhancing Learning through Gamification: A Systematic Review. *International Journal of Information and Education Technology*
4. Anderson, J. R., & Shneiderman, B. (2019), The craft of computer-based teaching. *Educational Technology*
5. Baylor, A. L., & Ritchie, D. (2021), What factors facilitate or inhibit the use of technology in education? *Educational Technology Research and Development*
6. Brown, J. S., Collins, A., & Duguid, P. (2022), Situated cognition and the culture of learning. *Educational Researcher*
7. Chen, C. M., Wang, Y. C., & Chen, M. H. (2020), Personalized learning and assessment: Exploring teachers' perceptions and practices with technology-enhanced formative assessment. *Educational Technology Research and Development*
8. Chen, S., Wang, L., & Zhang, Q. (2021), Enhancing Learning Through Simulation in Computer Aided Instruction. *Computers & Education*
9. Cheung, A. C., Slavin, R. E., & Kim, E. S. (2022), Effects of educational technology applications on reading outcomes for struggling readers: A best-evidence synthesis. *Review of Educational Research*
10. Chiang, T. H. C., Chen, Y. C., & Lin, C. Y. (2023), Game Based Learning and Game Mechanics: A Systematic Review of the Effects on Student Motivation and Learning Performance. *Journal of Educational Technology & Society*
11. Collins, K. (2020), Effective storytelling in game design. *Journal of Game Development*
12. Deterding, S., & Khaled, R. (2023). Enhancing player immersion through meaningful choices. *Game Research Journal*
13. Dichev, C., & Dicheva, D. (2021), Gamifying Education: What Is Known, What Is Believed and What Remains Uncertain: A Critical Review. *International Journal of Educational Technology in Higher Education*
14. Dunn, J., Loch, B., & McGovern, E. (2023), Insights into the relationship between the learning environment and self-directed learning readiness in first-year higher education students. *Higher Education Research & Development*
15. Ermi, L., & Mäyrä, F. (2020), Meaningful character arcs in games. *Journal of Interactive Narratives*
16. Forsyth, K., & McMillan, J. H. (2021), The effects of computer-assisted instruction on reading achievement: A synthesis of research. *Review of Educational Research*