THE INTERNATIONAL JOURNAL OF SCIENCE, TECHNOLOGY AND ENGINEERING
Volume 1, Issue 1, January - June 2025

CODE QUEST: A GAMIFICATION PLATFORM FOR ASPIRING JAVA PROGRAMMERS

Authors:

Jendel R. Juguilon
Pangasinan State University, Dagupan City, Pangasinan, Philippines
mymail.jrjuguilon@gmail.com

Shoun A. Ramos
Pangasinan State University, Lingayen, Pangasinan, Philippines
shounramos09@gmail.com

Nathaniel Akebono T. Saringan
Pangasinan State University, Lingayen, Pangasinan, Philippines
nathanielsaringan19@gmail.com

Joseph C. Oris
Pangasinan State University, Lingayen, Pangasinan, Philippines
oriscurry05@gmail.com

Jayson E. Tamayo
Pangasinan State University, Lingayen, Pangasinan, Philippines
jtamayo.lingayen@psu.edu.ph

Corresponding Author:

Jendel R. Juguilon
Pangasinan State University, Dagupan City, Pangasinan, Philippines
mymail.jrjuguilon@gmail.com

Date of Submission:	Date of Acceptance:	Date of Publication:
April 2, 2025	May 10, 2025	June 30, 2025

ABSTRACT

This study aimed to identify the algorithms used in the game creation of the Cross-platform game Code Quest. Cross-platform games work on multiple devices, making them more accessible and fostering larger player communities. They provide excellent accessibility and can serve educational purposes, such as learning through games. The purpose of educational games is to help users learn in a fun and engaging way. A great example is a quiz battle game, where players must select the correct answers to progress. Code Quest, a cross-platform Java programming quiz battle game, was developed to enhance users' Java knowledge. To achieve their goal, the proponents used Agile methodology (SCRUM), incorporating phases like product backlog, sprint planning, sprint backlog, and iterative sprints to build and refine the application efficiently. The proponents conducted user acceptance testing in terms of gameplay and user interface, and the participants gave excellent feedback on the application’s acceptability. In general, the cross-platform quiz battle game created in this study provides an enjoyable and effective tool for students facing difficulties in learning the Java programming language. It helps improve their knowledge through gameplay.

Keywords: algorithm, agile methodology, cross-platform, education game, quiz battle

INTRODUCTION

In today's fast-paced tech industry, programming has become a crucial skill, with web developers, software engineers, and programmers shaping the digital landscape. Among these, Java remains

a foundational language due to its platform independence, rich libraries, and strong community support. However, teaching and learning Java can be challenging, especially for aspiring programmers who struggle with problem-solving, modular thinking, and engagement in traditional settings. To address these challenges, this study aims to implement an engaging approach to help BSIT students at Pangasinan State University-Lingayen Campus (PSU-LC) enhance their Java skills through gamification.

The project introduces CodeQuest, a Java-based mobile game developed using Flutter and Dart, designed to create an immersive learning experience. Specifically, the study aims to: (1) determine the Java topics where BSIT students require additional support; (2) identify the algorithms used in game development, focusing on randomizing questions without repetition, adjusting question difficulty based on player performance, implementing a monitoring and leaderboard system, and establishing game logic; and (3) determine the acceptability level of the developed system. By transforming Java learning into an interactive experience, CodeQuest seeks to improve students' critical thinking, problem-solving abilities, and overall understanding of Java programming in a fun and accessible way.

METHODS

This study employed a quantitative approach with a descriptive design to explore how a gamified learning approach can enhance the Java programming skills of second-year Bachelor of Science in Information Technology (BSIT) students at Pangasinan State University, Lingayen Campus. The quantitative method was chosen because it allows for the objective measurement of the impact of gamification on student learning outcomes by generating numerical data that can be analyzed to identify trends and correlations. The descriptive design was used to assess how the gamified approach affects student engagement, motivation, and performance without manipulating any variables, providing a clear and accurate representation of the learning dynamics within the target population.

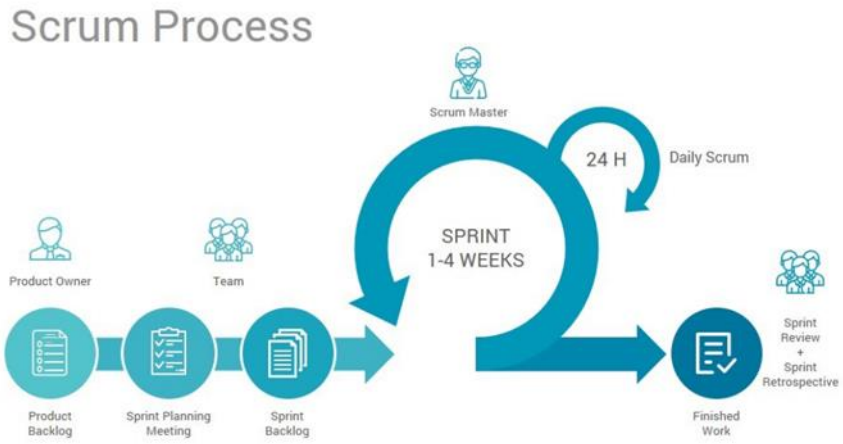
The study was conducted at Pangasinan State University, Lingayen Campus, and involved second-year BSIT students across three sections: Section A (45 students), Section B (42 students), and Section C (44 students), totaling 131 students. To determine the sample size, the researchers applied Slovin’s formula with a confidence level of 90% and a margin of error of 10% (0.1). The calculation resulted in a sample size of 57 respondents, ensuring a representative sample of the population. A purposive sampling technique was used due to time constraints, allowing the researchers to focus on the target group that best represented the needs of the study.

To achieve the study’s objectives, the researchers used multiple data collection instruments. For Objective 1, a needs assessment survey was conducted to identify the Java topics that BSIT students found most challenging and required additional support. This step was crucial in ensuring that the game content aligned with the specific learning needs of the students. For

Objective 2, the researchers adopted the Scrum framework to guide the development of the game. Scrum, an agile project management methodology, was used to organize the development phases and ensure a collaborative, flexible, and iterative approach.

Figure 1

Scrum Process



Note: Adapted from https://optimmus.io/what-is-scrum-the-ultimate-guide-to-the-scrum-software-development-process/

Flow of Randomizing and Generating Questions without Repetition

The algorithm ensures that each question in the game is unique and not repeated. It starts by creating a list of questions to be used during gameplay. A random number generator is then used to pick a question from this list. Once a question is selected, it is removed from the list to prevent it from being chosen again in the same round. This process continues until all the questions have been used, ensuring a varied and engaging experience for the player while maintaining fairness in the gameplay.

Algorithm in Adjusting the Difficulties of Questions Based on the Player's Performance

The algorithm starts by displaying the player with a default question. Based on the player's response, it adjusts the difficulty of the next question dynamically. If the player answers correctly, the next question is chosen from a more complex set of questions. On the other hand, if the player answers incorrectly, the following question comes from an easier set. This pattern continues, with the difficulty level adapting to the player's performance. If the player answers several questions correctly in a row, the questions remain challenging, while multiple incorrect answers keep the difficulty in the easier set. This approach ensures a balanced gameplay experience, keeping it engaging and tailored to the player's skill level.

Algorithm in Monitoring and Leaderboard System

The teacher creates a folder to organize and store a list of students, including their information and scores. Students play the game by entering a class code provided by the teacher. Using the same class code, all students are directed to the corresponding folder, where their progress is tracked. The students are then ranked within the folder based on the number of stars they earn in the game, ensuring a clear and motivating leaderboard that reflects their performance.

Flow of the Game Logic

The game starts by displaying a map where the players start with module one, chapter one, and round one, as intended for new users. The game generates questions, and the player's responses determine the flow of the battle. A correct answer allows the player's character to attack the enemy, while an incorrect answer results in the enemy attacking the player's character. If the player fails to answer within 60 seconds, the enemy automatically attacks. If the player successfully defeats the enemy, the player advances to the next round. To unlock the next chapter, the player must unlock the first three rounds of the current chapter. Similarly, to move to the next module, the player must complete all three chapters of the current module.

For Objective 3, an acceptability level survey questionnaire was used to evaluate user acceptance of the developed game. The questionnaire assessed various aspects such as gameplay, user interface, and overall user experience. Data collected from the surveys were analyzed using descriptive statistics. The needs assessment survey results were used to identify the most critical Java topics for inclusion in the game. To evaluate the acceptability of the gamified approach, the researchers used a 5-point Likert scale, where responses ranged from "Poor" (1) to "Excellent" (5), with the corresponding descriptive interpretations:

Table 1
Likert Scale for System Acceptability

Point Score	Statistical Range	Descriptive Equivalent	Descriptive Interpretation
5	4.51 – 5.00	Excellent (E)	Acceptable
4	3.51 – 4.50	Very Good (VG)	Acceptable
3	2.51 – 3.50	Good (G)	Acceptable
2	1.51 – 2.50	Fair (F)	Not Acceptable
1	1.00 – 1.50	Poor (P)	Not Acceptable

The weighted mean was computed to measure and analyze the data, providing an overall assessment of system acceptability. The formula used was:

$$\bar{\mathbf{x}} = \frac{\sum(\boldsymbol{wx})}{\mathrm{n}}$$

Where:

$\bar{x}$ = weighted mean

w = weight of each question

x = frequency of each respondent

n = total number of respondents.

Ethical standards were upheld throughout the study. Informed consent was obtained from all participants, ensuring that they were aware of the study's purpose, procedures, and their right to withdraw at any time. Confidentiality and anonymity were maintained by safeguarding participants' identities and ensuring that collected data was used solely for research purposes. Institutional approval was obtained prior to conducting the study, ensuring compliance with ethical guidelines.

By using a well-defined methodology, this research ensured that the gamified learning approach was developed and evaluated effectively, providing meaningful insights into its impact on student engagement, motivation, and learning outcomes.

RESULTS

The developed game incorporated effective algorithms for managing game logic, randomizing questions without repetition, adjusting difficulty dynamically, and tracking player progress through a monitoring and leaderboard system.

Table 2

Needs Assessment Survey

Topics	Tally	Percentage
Java Programming Fundamentals	36	63.15%
Abstract Windowing Toolkit and Swing	23	40.35%
Decision Control Structures	35	61.40%
Repetition Control Structures	34	59.65%
GUI Event Handling	23	40.35%
Objects and Classes	23	40.35%
User-Defined Classes	22	38.60%
Inheritance, Polymorphism and Interfaces	21	36.84%
Getting Input from the Keyboard	35	61.40%
Java Arrays	33	57.89%
Total	285	

The results of the needs assessment survey highlighted the five most challenging Java topics for second-year BSIT students, which were integrated into the game to address the identified learning gaps.

The table 3 shows the results of acceptability level in terms of Gameplay.

Table 3
Acceptability level in terms of Gameplay

Indicators	Weighted Mean	Descriptive Equivalent	Descriptive Interpretation
Learnability – effort in learning the game is reduced.	4.51	Excellent	Acceptable
Enjoyment – the game is fun and entertaining.	4.60	Excellent	Acceptable
Originality – the game has unique qualities.	4.53	Excellent	Acceptable
Animation – the game characteristics display smooth movement.	4.54	Excellent	Acceptable
Educational – the game is informative.	4.54	Excellent	Acceptable
Average Weighted Mean	4.54	Excellent	Acceptable

The table 4 shows the result of acceptability level in terms of User Interface.

Table 4
Acceptability level in terms of User Interface

Indicators	Weighted Mean	Descriptive Equivalent	Descriptive Interpretation
Buttons and Game Text – buttons and game text are readable	4.53	Excellent	Acceptable
User Friendliness – the game is easy to learn and use	4.54	Excellent	Acceptable
Background – used background has attractiveness.	4.68	Excellent	Acceptable
Music – background and sound effects make the game lively	4.72	Excellent	Acceptable
Average Weighted Mean	4.62	Excellent	Acceptable

The acceptability evaluation revealed that CodeQuest achieved high levels of acceptability in terms of both gameplay and user interface, demonstrating that the game is enjoyable, educational, and easy to navigate. The findings affirm that the developed game meets the requirements of aspiring Java programmers, providing an engaging and effective learning platform.

CONCLUSIONS AND RECOMMENDATIONS

Code Quest effectively covers key Java topics where students require additional support, including Java Programming Fundamentals, Decision Control Structures, Repetition Control

Structures, Getting Input from the Keyboard, and Java Arrays. The inclusion of essential algorithms for randomizing questions, adjusting difficulty, tracking progress, and enhancing game logic ensures that the game provides an engaging and adaptive learning experience.

The needs assessment survey validated that these Java topics were the areas where students needed further reinforcement, and their integration into the game addressed these gaps effectively. The high ratings in terms of gameplay and user interface indicate that Code Quest was enjoyable, well-designed, and highly acceptable to the target audience. The game was successful in providing an interactive, fun, and educational platform that supports student learning while maintaining a high level of engagement.

The study contributes to the gamification of education, demonstrating that incorporating game-based learning can effectively address learning gaps and motivate students to improve their programming skills.

To further improve Code Quest based on the study's findings, several recommendations are suggested. Future developers should expand the game's content by incorporating more advanced Java topics such as Object-Oriented Programming, Inheritance, Polymorphism, and Interfaces to enrich the educational experience. Enhancing gameplay through the addition of new maps and game modes can help sustain player interest. Visual appeal and engagement can also be improved by refining game assets, animations, and graphics. Additionally, improving the user interface will ensure smoother navigation and usability. For classroom integration, the monitoring system should support CSV or Excel file imports for easier student data management, and it must be thoroughly tested with teachers to ensure its effectiveness in tracking student progress.

ACKNOWLEDGMENT

The researchers would like to thank all the following people for their assistance and lending their hand in completing this study. This project will not be possible and successful without them. To Mrs. Carla Carmela P. Ramos, our subject adviser for checking on us and monitoring on our work, giving us advice and some ideas. To Mr. Robert V. Ocampo Jr, our defense chairman, who provided his researching insights and suggestions to improve the study. To our panel members, Engr. Jerry C. Diaz and Mr. Virgilio A. Aquino, for their constructive feedback and support. To Mr. Jayson E. Tamayo, who taught us the programming language we used to develop this project, for sharing his knowledge of game development, and for his unwavering encouragement. To Mr. Julio Cervantes, for providing essential information for the game creation. We want to express our sincerest gratitude to our family, relatives, friends and our Almighty God for strength, financial support, love, and understanding.

DECLARATION ON THE USE OF AI TOOLS

In this study, the use of artificial intelligence (AI) tools was employed to enhance language refinement and improve the overall coherence of the manuscript. Specifically, AI-assisted tools were utilized to ensure clarity and conciseness in presenting research findings, while maintaining the originality and integrity of the content. No AI tools were used in data analysis or image generation, ensuring that the interpretation and conclusions drawn remain solely the responsibility of the researchers. This acknowledgment highlights the transparent use of AI technologies while adhering to ethical research practices and safeguarding the accuracy and validity of the study.

REFERENCES

Ateneo, A., Mercedes, M., Rodrigo, T., Marion, J., Torres, R., & Lim, T., et al. (2021). For people and planet: Teachers' evaluation of an educational mobile game and resource pack. Pacific Society for Computers in Education.

Francisco-Catle, A. (2018). Effects of computer-based educational games on students' achievement in science. [Unpublished master's thesis or dissertation if applicable].

Funcion, D. G. D. (2019). Development of random tree-based student competency model in Java programming. Indian Journal of Science and Technology, 12(6), 1–8. https://doi.org/10.17485/ijst/2019/v12i6/137465

Garcia, M. B., & Oducado, R. M. F. (2021). Intention to utilize mobile game-based learning in nursing education from teachers' perspective: A theory of planned behavior approach. 2021 1st Conference on Online Teaching for Mobile Education (OT4ME), 103–107. https://doi.org/10.1109/OT4ME53559.2021.00024

Ibrahim, R., Rahim, N. A. Z., Wong Ten, D. H., Yusoff, R. C., Maarop, N., Yaacob, S., & Yahaya Petra, J. S. (2018). Students' opinions on online educational games for learning programming introductory. International Journal of Advanced Computer Science and Applications (IJACSA), 9(6). https://www.ijacsa.thesai.org

Jondya, A. G., Anwar, H. I., Hermawan, M. R., & Deru, T. A. (2022). Developing "EMBER": A fantasy puzzle adventure game with an approach to Agile by applying game-Scrum methodology. Engineering, Mathematics and Computer Science (EMACS) Journal, 4(2), 39–43.

Jones, L., & Smith, T. (2020). Introductory computer programming courses in mathematics curriculum. Teaching Mathematics and Computer Science, 18(1), 19–30. https://doi.org/10.5485/TMCS.2020.0492

Krajcsi, A., Csapodi, C., & Stettner, E. (2021). Algotaurus: An educational computer programming game for beginners. Interactive Learning Environments, 29(4), 634–647. https://doi.org/10.1080/10494820.2019.1587463

Lomugdang, T. A., Montibon, R., & Omagad, M. P. (2021). A study on the development of e-learning system for Java programming. Journal of Innovative Technology Convergence, 3(2), 1–10.

Majalin, T., et al. (2020). A preliminary study of difficulties in learning Java programming for secondary school. International Journal of Advanced Trends in Computer Science and Engineering.

Manuel, R., et al. (2019). Simplifying the creation of adventure serious games with educational-oriented features. Educational Technology and Society.

Marcial, D. E., Dy, J. M., & Montemayor, J. O. (2022). Gamifying “whole-person education”: The development of a mobile application with augmented reality. Information Technologies and Learning Tools, 91(5), 130–144. https://doi.org/10.33407/itlt.v91i5.4857

Mathew, R., Malik, S. I., & Tawafak, R. M. (2019). Teaching problem-solving skills using an educational game in a computer programming course. Informatics in Education, 18(2), 359–373. https://doi.org/10.15388/infedu.2019.17

Mehmood, E., Abid, A., Farooq, M. S., & Nawaz, N. A. (2020). Curriculum, teaching and learning, and assessments for introductory programming course. IEEE Access, 8, 125961–125981. https://doi.org/10.1109/ACCESS.2020.3008096

Francisco, K. B., & Montes, S. B. (2019). Philippines first: An edutainment 3D game for Android mobile platform using separating axis theorem (SAT) algorithm.

Resuelo, M., et al. (2023). Motivation and self-efficacy in Java programming among B.S. Information Technology and B.S. Information Systems students. Journal of Namibian Studies.

Ruiz, M. F. L., Esclada, R. I., Balderama, J. H., Aguirre, A. A. I., Cutin, M. N. I., & Lozada, R. M. (2021). WISE game: Wireless interactive software educational game. Indonesian Journal of Electrical Engineering and Computer Science, 23(3), 1357. https://doi.org/10.11591/ijeecs.v23.i3.pp1357-1365

Seralidou, E., & Douligeris, C. (2021). Learning programming by creating games through the use of structured activities in secondary education in Greece. Education and Information Technologies, 26(1), 859–898. https://doi.org/10.1007/s10639-020-10280-1

Souza, M. R. de A., Veado, L., Moreira, R. T., Figueiredo, E., & Costa, H. (2018). A systematic mapping study on game-related methods for software engineering education. Information and Software Technology, 95, 201–218. https://doi.org/10.1016/j.infsof.2017.10.002

Bigueras, R. T. (2020). Mobile game-based learning to enhance the reading performance of dyslexic children. International Journal of Advanced Trends in Computer Science and Engineering, 9(1.3), 332–337. https://doi.org/10.30534/ijatcse/2020/4791.32020

Tsiotras, D., & Xinogalos, S. (2021). Investigating the perceived player experience and short-term learning of the text-based Java programming serious game “Rise of the Java Emperor.” Informatics in Education, 20(1), 153–170. https://doi.org/10.15388/infedu.2021.08

Xinogalos, S., & Satratzemi, M. (2022). The use of educational games in programming assignments: SQL Island as a case study. Applied Sciences, 12(13), 6563. https://doi.org/10.3390/app12136563

Yabut, E. R., Jamis, M. N., Manuel, R. E., & Fabito, B. S. (2019). Empowering elementary schools on learning math: A development of gamified educational mobile application for grade 3 students. 2019 IEEE 11th International Conference on Humanoid,

Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), 1–5. https://doi.org/10.1109/HNICEM48295.2019.9072844

Zhu, J., Alderfer, K., Furqan, A., Nebolsky, J., Char, B., Smith, B., Villareale, J., & Ontañón, S. (2019). Programming in game space. Proceedings of the 14th International Conference on the Foundations of Digital Games, 1–10. https://doi.org/10.1145/3337722.3341876