iJETRM
International Journal of Engineering Technology Research Management
(IJETRM)
https://ijetrm.com/

CODEREALM: A STORY-DRIVEN GAME FOR LEARNING PROGRAMMING LANGUAGES

Ronald B. Fernandez
Professor, College of Computing Studies, Universidad de Manila, Philippines
Jhon Micheal Ayson, Mary Jane Coching, Jasper John Dalmacio, Jaymar Mariño, Lestat Salazar
College of Computing Studies, Universidad de Manila, Philippines

ABSTRACT

This study occupies a sizable gap in programming education raised by educational platforms using text only, like W3Schools, which lacked engagement and practical use. CodeRealm is our answer to this challenge, a gamified, storytelling learning game, which uses immersive quests, interactive puzzles, and real-world code challenges to teach programming languages (Java, C++, Python, JavaScript). CodeRealm is used to assist the user in understanding and retaining abstract programming concepts such as syntax, loops, and functions by placing them within a fantasy and science fiction (sci-fi) theme. Players will take on the role of Nova, a young coder whose job is to recover the lost code scrolls and thwart the destructive desires of a rogue AI called Omega-0, and it is set in a futuristic universe.

The modern digital world has increased the demand for programming exponentially, but the conventional way of learning fails to grab the attention and keep the students interested. A lot of learners find it a hard task to follow a tutorial that is text-intensive and only teaches syntax but not practice. With the help of the Flow Theory and self-determination, CodeRealm creates an engaging environment where learners can level up, gain rewards, and collaborate.

This work will further the field of digital pedagogy by demonstrating how gamification through narrative could turn the teaching of coding into a fun, activity-based experience. Nevertheless, CodeRealm currently has limitations because it targets only IT students and beginners.

Keywords:
Gamification, Programming Education, Interactive Learning, CodeRealm, IT Students.

INTRODUCTION

Programming knowledge is nowadays required more than ever before and most of the students do not enjoy learning how to code through the conventional teaching methods. Interactive technologies have become more relevant in learning due to the fact that the traditional learning processes cannot capture the attention of the students, in a subject like programming. The students who most get lost are the ones whose theoretical concepts of coding are taught using the text based, standard tutorials and they therefore lose their interest and nothing applies to them.

The main aim of this paper is to see how gamification enhances the learning of coding by turning it into an interactive, exciting and an effective experience. Unlike other more traditional platforms like W3Schools which use text-based instruction, CodeRealm is a narrative based interactive platform in which the students are put on a quest and encounter problems which they have to resolve through coding. CodeRealm strives to close the gap between theoretical knowledge and practice with the addition of learning content and interactive play.

A challenge to IT education is that students find it difficult to memories purely theoretical courses, which thwarts their ability to transfer programming techniques to practical scenarios. International Society Technology in Education (ISTE, 2023) reports that by incorporating gameplay and learning content, learning tools have a powerful potential to improve learning retention and acquisition of practical skills. The strategy is consistent with the growing importance of different types of instruction that ensure better understanding and memorization of facts.

In this study, the researcher aims to develop the programming literacy of students by identifying the weaknesses of the traditional approach to teaching students code, and introducing a gamification methodology that will improve learning experience and make it more interesting, relevant, and accessible to future programmers and IT workers.

iJETRM
International Journal of Engineering Technology Research Management
(IJETRM)
https://ijetrm.com/

OBJECTIVES

The primary goal of this project is to create CodeRealm, a learning game based on interactive stories to learn programming languages and fundamental coding concepts. In particular, the project aims to create and design an immersive game environment that enhances students learning through programming lessons presented in the form of a story, works as a way to engage students better through the inclusion of active elements, like puzzles and quests, and determine the effects of coding through gamifications on user engagement, knowledge retention, and development of practical programming skills.

METHODOLOGY

This research design and tested CodeRealm, a programming language based education game using both developmental and experimental methodologies. The development model adopted was the Agile Development Life Cycle (ADLC) that guaranteed iterative design, rapid prototyping, constant feedback, and progressive system refinement. The phases of ADLC applied were as follows:

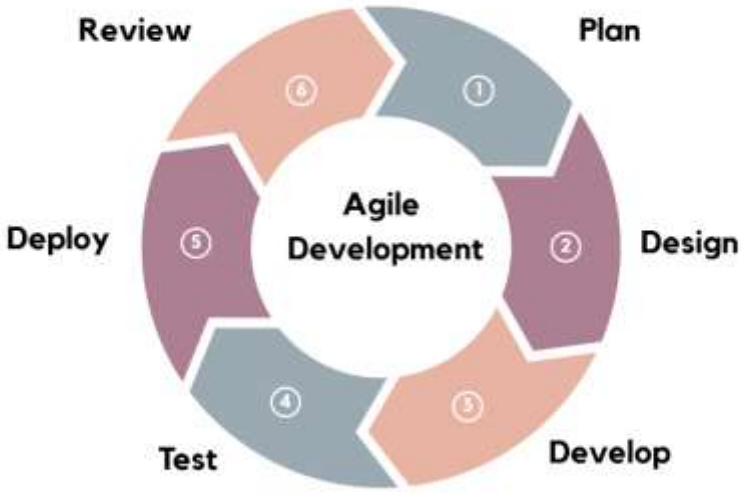

Figure 1 Phases of the Agile Development Life Cycle for CodeRealm

The ADLC phases implemented were as follows:

1. Planning – Determining project scope, learning objectives, and defining the system requirements and anticipated outcomes.
2. Design – Creating wireframes, story elements, game flow, and coding challenges integrated into quests. User interface designs and interaction models were also finalized.
3. Development – Implementing the game mechanics, integrating the real-time coding editor, and developing modules such as quest progression, scoring, and feedback.
4. Testing – Conducting usability tests, debugging, and evaluating system stability and functionality.
5. Deployment – Launching the prototype for pilot testing with selected IT students and beginners.
6. Review – Gathering user comments, interpreting findings, and making system improvements. The researchers developed a game environment in the developmental approach, which incorporates narrative-based learning through interactive tasks of coding. The programming concepts that were presented as story-based progressively in the quests included syntax, loops, and functions.

C. Data Gathering

The research was based on the collection of primary and secondary data to facilitate CodeRealm development and assessment. Primary data involved surveys, questionnaires and interviews with IT students and novice programmers

iJETRM
International Journal of Engineering Technology Research Management
(IJETRM)
https://ijetrm.com/

to evaluate their learning needs, user experience, and extent of engagement when using the system. These reactions were crucial in refining the systems. The secondary data was collected through related research, scholarly journals and available gamified learning websites, which were used as the reference material to develop an effective and educationally valuable game-based coding environment.

D. Tools and Technologies

To make CodeRealm as functional and accessible as possible, a mix of modern technologies was used. Frontend of the system was developed using React.js and HTML and CSS that made it interactive and user friendly. Node.js (JavaScript) was used to implement the backend to do game logic, APIs and data communication. Firebase was used as the main base to store user profiles, progress, and achievements and also real-time sync. The play engine was done in web browser to do coding tests and interactive quests. GitHub was used to handle version control and collaboration, and the platform was rolled out as a web-based application to achieve accessibility among devices.

E. Ethical Considerations

The development and evaluation of CodeRealm followed ethical standards. The objectives of the study were communicated to all the participants in the surveys and user testing and their consent was obtained before they participated. Anonymity of the responses and safe storage of records protected data confidentiality. The game was designed with accessibility guidelines so as to support various learners such as those who are first time users and those with lower technical knowledge. Moreover, the learning material was thoroughly tested to be accurate in programming concepts, and to enhance fairness, inclusivity, and integrity in the learning experience.

RESULTS AND DISCUSSION

This section presents the findings obtained from the development, testing, and initial deployment of the CodeRealm: A Story-Driven Game for Learning Programming Languages. The results are structured to address the study's objectives regarding functionality, usability, and effectiveness.

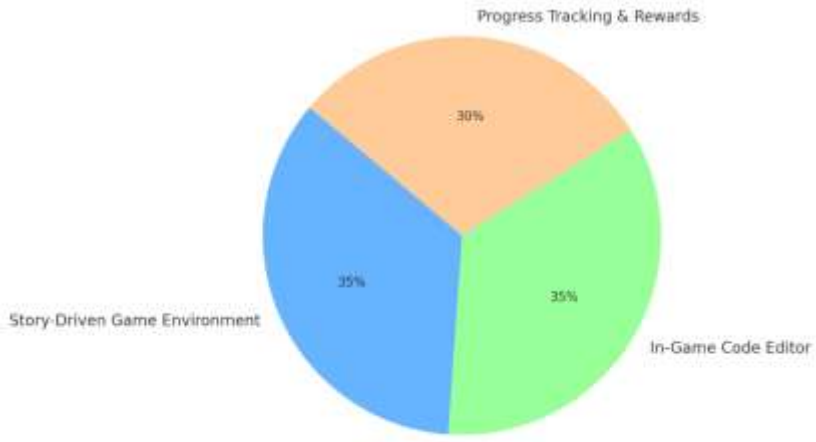

Figure 2 CodeRealm System Compounds Implementation

Development and Implementation of the System

The study successfully designed, developed, and implemented a fully functional gamified learning platform for programming education. The system comprised three core components as proposed:

1. Story-Driven Game Environment where players embark on coding quests integrated into a narrative structure.
2. In-Game Code Editor with Real-Time Feedback that allows users to write, test, and debug code within the game itself.

iJETRM
International Journal of Engineering Technology Research Management
(IJETRM)
https://ijetrm.com/

3. Progress Tracking and Reward System that provides experience points (XP), achievements, and level progression to reinforce engagement and learning outcomes.

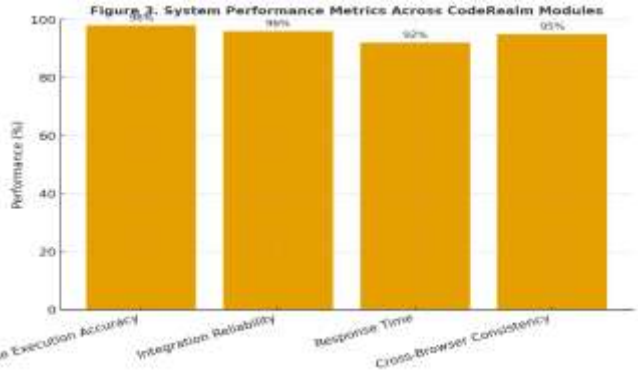

Figure 3. System Performance Metrics Across CodeRealm Modules

Findings on Technical Performance and Reliability

The study evaluated the system's technical performance through structured testing protocols. The findings are as follows:

1. Code Execution and Debugging Accuracy – Unit testing confirmed that the integrated in-game code editor successfully executed coding tasks and returned accurate outputs for Java, Python, and JavaScript. The real-time error detection features consistently identified and explained syntax errors.
2. System Integration – Integration testing revealed seamless interaction among the narrative engine, coding editor, and progress tracking modules. Data on XP, quest completion, and achievements were synchronized in real-time via Firebase without errors.
3. Performance Under Load – Stress testing demonstrated that the system could handle up to 40 concurrent users with an average response time of under 2.8 seconds for code execution and feedback. This confirmed the platform's readiness for classroom-scale deployment.
4. Cross-Browser Functionality – The web-based deployment of CodeRealm was validated on Chrome, Edge, and Firefox. The system displayed consistent functionality across devices and browsers.

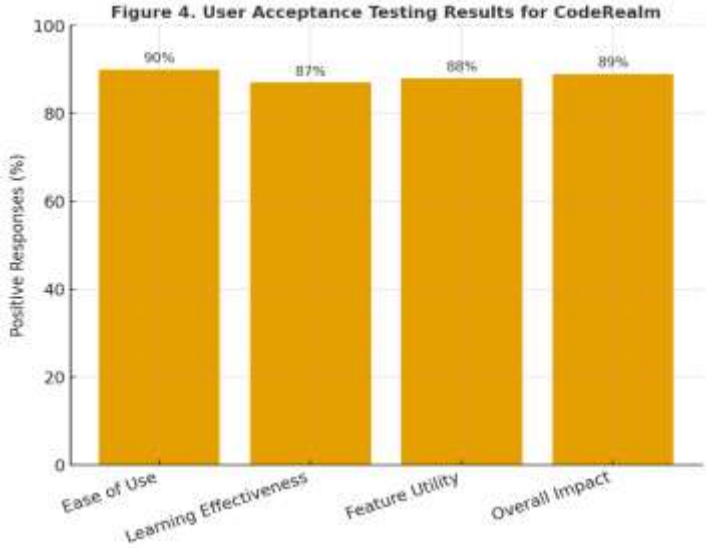

Figure 4. User Acceptance Testing Results for CodeRealm

iJETRM
International Journal of Engineering Technology Research Management
(IJETRM)
https://ijetrm.com/

Findings on User Acceptance and Usability

User Acceptance Testing (UAT) was conducted with 30 participants composed of IT students and beginner programmers. The following quantitative results were observed:

1. Ease of Use – 90% of participants agreed that the interface was user-friendly and easy to navigate.
2. Learning Effectiveness – 87% confirmed that coding lessons delivered through quests and challenges improved their understanding of syntax, loops, and functions compared to text-based tutorials.
3. Feature Utility – The real-time debugging feature was rated as "very useful" by 88% of users, while 85% valued the XP and rewards system for motivation.
4. Overall Impact – 89% of participants expressed satisfaction with the platform, stating they would prefer CodeRealm over traditional tutorials for introductory programming.

DISCUSSION

One of the biggest issues facing most students, especially novice ones, is learning to program; they find it difficult to learn the abstract concepts based on textual tutorials that lack interactivity. The results of this research prove that CodeRealm has benefited the interaction with and understanding of programming via gamified, narrative-based learning, which is in line with the findings of Tacouri and Nagowah [7], who found that mobile game-based learning increases motivation and retention of the knowledge. CodeRealm offered a platform that was functional and easy-to-use, responding to the same challenges inherent in the traditional learning process, including insufficient interactivity, inability to maintain motivations, and little chance of real-time practice. The system brings a compelling tool to digital learning by merging storytelling with real-time coding, making it less dependent on traditional static resources.

The benefits of working with CodeRealm are obvious. In many cases, the traditional tutorial allows the learner to memorize syntax without a lot of context and results in frustration and non-retention. Students also have problems related to the ability to use abstract concepts when solving real-life issues. By comparison, the quests, coding challenges and rewards offered by CodeRealm guarantee real-time feedback, interactivity and motivation. In contrast to other more traditional online platforms, which tend to be text-heavy, the CodeRealm uses gamification and interactive storytelling to engage learners. The ease of access to the web browsers is also more inclusive because one does not need a specific hardware or costly software in order to take advantage of the system. Its results also indicate wider implications about programming education and digital pedagogy. CodeRealm provides not only a good way to enhance engagement but also a convenient practice of coding concepts. Novices are able to understand the logic of programming better by following quests, and, instructors may make use of the platform to further their lessons. This decreases the reliance on conventional exercises and prompts students to look at coding as a creative and fun process. By enabling the learner to experiment, get instant feedback and be rewarded, CodeRealm provides a more streamlined learning process and promotes long-term memory.

There are some study limitations which need to be taken note of as well. As CodeRealm was only tested by a small number of IT students and novice programmers, it is possible that the results do not represent the experiences of more experienced learners. The existing system also lacks support of more advanced concepts and is limited to use through web browsers only. There may be larger barriers to sustaining higher-order programming paradigms, scale, or curricular integration. The other disadvantage is internet addiction since, with weaker networks, some users can become less responsive.

The limitation of the CodeRealm should be addressed as well as its expansion capabilities should be used in the future. Running the platform on various types of learners such as advanced students, professionals, and various academic institutions would give us some network on scalability and adaptability. Upgrades would involve adding the benefit of multiplayer teamwork, voice recognition instruction, and the feature to support learners with disabilities. Moreover, the inclusion of more advanced material like object-oriented programming, algorithms, and data structures would make it more useful in the long run. Administrators and teachers might also incorporate analytics to monitor learning progress and customize learning routes to different needs.

The fact that technology can make programming learning an affordable, interactive and fun experience is important. CodeRealm proves that gamification, narrative design, and real-time coding can be integrated into a system

iJETRM
International Journal of Engineering Technology Research Management
(IJETRM)
https://ijetrm.com/

to help not only the plight of novice learners but also offer a viable answer on how to improve digital pedagogy and programming education.

ACKNOWLEDGEMENT

This research study entitled “CodeRealm: A Story-Driven Game for Learning Programming Languages” would not have been possible without the invaluable support, guidance, and encouragement of many individuals who contributed to its successful completion.
To begin with, the researchers would like to express their heartfelt gratitude to the College of Computing Studies at Universidad de Manila for providing the academic foundation and technical guidance essential in carrying out this project. We also extend our deepest appreciation to our professor and research adviser, Prof. Ronald B. Fernandez, for his invaluable guidance, insightful feedback, and constant encouragement. His support inspired us to persevere, think critically, and give our best throughout the development of this study.
As a first expression of gratitude, the researchers are sincerely thankful to the faculty, staff, and administration of Universidad de Manila for their cooperation and support, which ensured that this project aligned with the academic and developmental goals of the institution.

CONCLUSION

CodeRealm is a successful application that combines gamification and story-telling learning to solve the usual problems of teaching programming. It enhances interaction, retention and hands-on coding practice between IT students and novices. In order to increase its effectiveness, the further development should lead to more sophisticated topics of programming, the spread of platforms, and collaboration. CodeRealm offers a potential solution to make coding education a fun and useful process by addressing the gap between theory and practice.

REFERENCES

1) Moldez, R., & Figueroa, R. (2024). Innovation in Education: Gamification in MOOCs. University of the Philippines Open University.
2) Emlano, C., & Walag, R. (2023). Development and Acceptability of Gamified Modules on Organic Compounds.Philippine Journal of Education.
3) Luzano, J. F. P. (2024). Dynamics of Gamification in Tertiary Mathematics Education. Philippine Educational Review.
4) Li, Y., & Wang, H. (2022). Effectiveness of Gamification in Programming Education: Meta-Analysis. International Journal of Educational Technology.
5) Garcia Villegas, J., & Lemos Aguero, S. (2023). The Gamification of E-Learning Environments for Programming. Journal of Computer Education.
6) Stock Knowledge. (2024). Philippine Startup Pilots Gamified EdTech in DepEd Makati. Department of Education Report.
7) Tacouri, R., & Nagowah, L. (2021). Code Saga: Serious Mobile Game for Learning Programming. University of Mauritius.
8) Kanniga, P. M., Selvi, K., Shrinidhii, M., & Pradeepa, S. (2024). EscapeScript: A Gamified Coding Platform.International Journal of Emerging Technologies.
9) International Society for Technology in Education (ISTE). (2023). Report on Gamified Learning. Washington, D.C.
10) FGPE++. (2023). Gamified Programming Learning at Scale: European University Project. European Education Research Network.
11) Schools Division of Albay. (2023). Digital Game-Based Learning for Basic Education. Department of Education.
12) Tavares, C., & Primo, T. (2022). Game-Based Learning: Enhancing Student Experience, Knowledge Gain, and Usability in Higher Education Programming Courses. Computers & Education Open, 3, 100090.
13) St-Bernard, R. (2023). Examining the Impact of Game-Based Learning on Student Engagement and

iJETRM
International Journal of Engineering Technology Research Management
(IJETRM)
https://ijetrm.com/

Performance in an Introductory Computer Programming Course. Lancaster University E-Prints.
14) Nguyen, H., & Trinh, P. (2021). Gamification in Programming Learning: A Case Study on Student Motivation and Achievement. International Journal of Educational Research, 105, 101725.
15) Omar, N., & Rahim, A. (2023). Effect of Gamification on the Motivation of Computer Programming Students. Journal of Educational Technology and Online Learning, 5(2), 89–102.
16) Deshpande, A., & Huang, T. (2022). PlayIT: Game-Based Learning Approach for Teaching Programming Concepts. International Journal of Game-Based Learning, 12(3), 45–57.
17) Alrashidi, M., & Cilliers, L. (2023). Exploring the Effectiveness of Game-Based Learning in Computer Programming Education: Enhancing Motivation and Skill Retention. International Journal of Emerging Technologies in Learning (iJET), 18(5), 120–135.
18) Akhmetova, A., & Park, J. (2024). A Serious Game for Programming in Higher Education. Computers and Education: Artificial Intelligence, 7, 100248.
19) Apostol, S., & Filippou, D. (2023). The Effectiveness of Gamification in Programming Education: Evidence from a Meta-Analysis. Education and Information Technologies, 28(5), 4457–4478.
20) Gómez, J., & García, E. (2023). Does Gamification Make a Difference in Programming Education? A Study Using the FGPE Platform. Education Sciences, 13(10), 984.