IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 11, Issue 10, October 2023

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

9940 572 462 | 6381 907 438 | ijircce@gmail.com | www.ijircce.com

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110027 |

"KNOCK KNOCK", HERE'S THE ONLINE GAME FOR LEARNING BASIC CONCEPTS OF PROGRAMMING LANGUAGES

Shaharukh Shaikh, Prof. Swaroopa Kamble

Dept. of Computer Engineering, Ajeenkya DY Patil School of Engineering Lohegaon, Maharashtra, India

ABSTRACT: This research investigates the integration of online serious games as a novel pedagogical approach to teaching basic programming concepts to high school students while fostering engaging and gameful learning experiences. As technology continues to shape our world, proficiency in computer programming has become an essential skill. However, traditional programming education methods often fail to capture the interest and motivation of high school students.

To address this challenge, we designed and implemented an online serious game that combines elements of gaming with structured programming tasks. This game was deployed to a group of high school students, and their experiences were thoroughly assessed using a mixed-methods approach.

The results of the study revealed that the use of an online serious game not only enhanced students' understanding of basic programming concepts but also significantly increased their motivation and engagement with the learning material. By incorporating game elements such as points, levels, and immediate feedback, the students found the learning process to be more enjoyable and immersive.

This research contributes to the growing body of knowledge on game-based learning and offers valuable insights into the potential of online serious games as an effective educational tool for high school students. Moreover, it highlights the significance of creating gameful experiences in educational settings, making programming education more accessible and engaging for the next generation of digital learners.

KEYWORDS: Serious games, programming basics, gameful experience, data flow diagrams, C language.

I. INTRODUCTION

In an era characterized by rapid technological advancement, the ability to code and program has become an increasingly essential skill, transcending the boundaries of niche expertise and infiltrating nearly every facet of modern life. Whether it's in creating software applications, understanding data analytics, or solving complex problems, programming knowledge is a powerful tool that empowers individuals to navigate and shape our digital world. Consequently, it is crucial that programming education begins at an early stage, equipping high school students with the fundamental skills they need to thrive in the 21st century.

However, traditional methods of teaching programming, often characterized by dry lectures and rote memorization, have struggled to captivate the imaginations of high school students. As educators grapple with the challenge of fostering interest and motivation in this critical field, innovative pedagogical approaches are required. One such approach is the integration of online serious games into the curriculum, offering a promising avenue for reimagining programming education.

Online serious games represent a unique fusion of gaming and education, designed to provide an immersive and engaging learning experience. By incorporating the principles of gamification, these games introduce elements such as points, levels, competition, and immediate feedback into the learning process. Such approaches can not only facilitate a more interactive and enjoyable educational experience but also stimulate higher levels of engagement and motivation among students.

This research explores the implementation of an online serious game aimed at teaching high school students basic

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110027 |

programming concepts while fostering a gameful learning environment. By investigating the impact of this innovative approach, we aim to shed light on the potential of online serious games in shaping the future of programming education, making it more accessible, engaging, and relevant to the digital generation. Through a comprehensive examination of the learning outcomes and experiences of high school students engaged with the game, this study contributes valuable insights to the growing field of game-based learning, with the potential to reshape how programming is taught in schools.

II.METHODOLOGY

1. Game Development and Content Creation:

- Game Design: Develop the online serious game designed to teach basic programming concepts and principles. The game should cover programming languages such as C, C++, Java, HTML, and CSS, while focusing on their fundamental concepts.
- Curriculum Integration: Align the game content with the high school programming curriculum to ensure that it covers essential topics.

2. Participant Selection:

- High School Students: Recruit a diverse group of high school students who have varying levels of programming knowledge. This may involve partnering with local schools or educational institutions.

3. Pre-assessment:

- Administer a pre-assessment to gauge the students' baseline knowledge of programming concepts and languages, helping to establish a starting point for the study.

4. Game Deployment:

- Provide access to the online serious game to the selected participants. Ensure that they have the necessary technical resources, such as computers and internet access, to engage with the game.

5. Data Collection:

- Quantitative Data:

Monitor in-game progress and performance metrics, such as completed tasks, points earned, levels achieved, and time spent on different sections.

Use quizzes and assessments within the game to track the students' understanding of programming concepts.

- Qualitative Data:

Conduct interviews, focus group discussions, and surveys to gather qualitative insights into students' experiences, motivations, and challenges.

6. Post-assessment:

- Administer a post-assessment to measure the knowledge and skills acquired by students after engaging with the game.

7. Data Analysis:

- Quantitative Analysis: Analyze the quantitative data to determine improvements in programming knowledge and skills. Use statistical tools to compare pre- and post-assessment scores.
- Qualitative Analysis: Thematic analysis of qualitative data from interviews, focus groups, and surveys to identify patterns in student experiences, challenges, and motivations.

8. Feedback and Iteration:

- Collect feedback from students about their experiences with the game and their suggestions for improvement.
- Use student feedback to make necessary adjustments to the game, potentially adding or modifying content to enhance the learning experience.

9. Comparison Group:

- If possible, create a control group of high school students who receive traditional programming education to compare learning outcomes and experiences.

10. Ethical Considerations:

- Ensure that all data collection and research activities comply with ethical standards and obtain necessary consent from students and their guardians.

11. Data Interpretation:

- Interpret the results in the context of the game's impact on the students' learning, motivation, and engagement with programming concepts.

12. Conclusion and Implications:

- Draw conclusions regarding the effectiveness of the online serious game in teaching programming concepts.
- Discuss the implications of the findings for the future of programming education for high school students.

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110027 |

13. Recommendations and Future Work:

- Offer recommendations for further game development, curriculum integration, and research in the field of game-based programming education.
- Suggest areas for future investigation, such as exploring the long-term retention of knowledge and assessing the transferability of skills learned through the game to real-world programming tasks.

III. MODELING AND ANALYSIS

Basically Our Concept is to teach the student via Game mode, So Below Mention are some of the games of it.
We also provide the score Dashboard for improvement levels of the students, so as if we are gaming and leveling up .

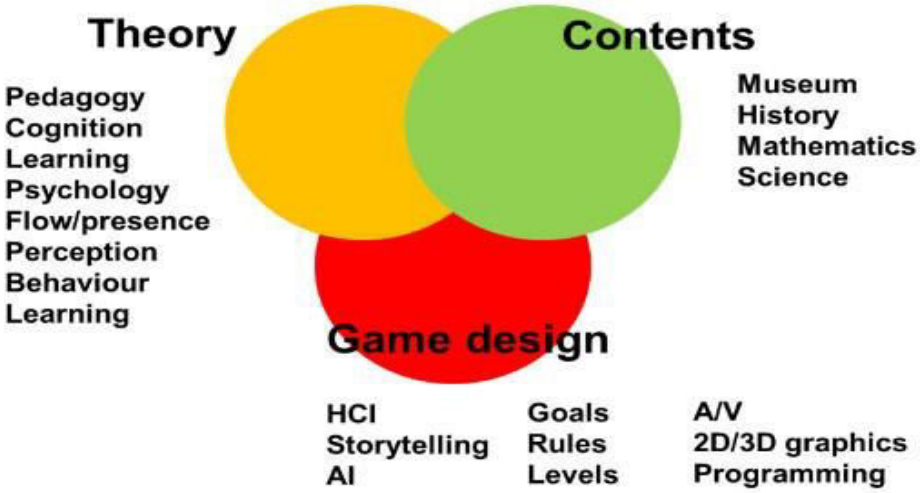

FIGURE 1. The three kernels of SG design

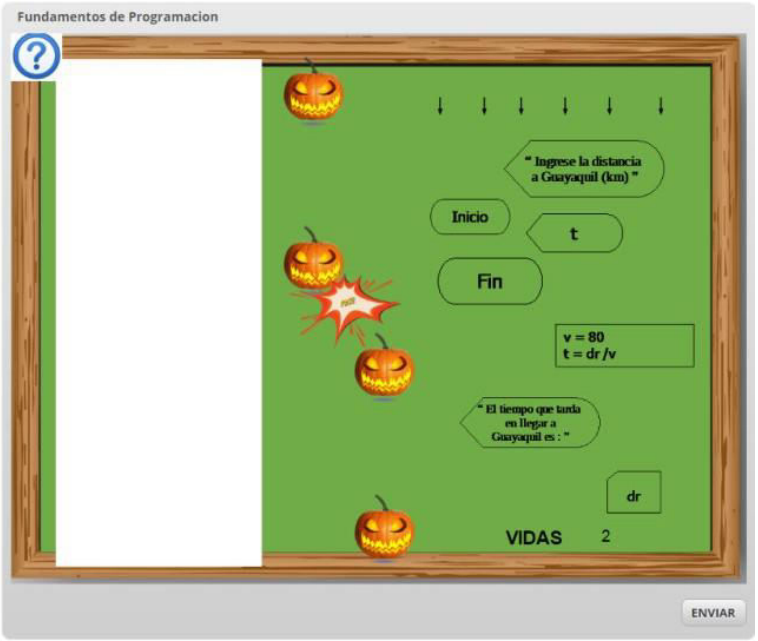

FIGURE 2. Game ''Avoid the Pumpkins'' the player must arm the DFD for the sequential algorithm. By dragging the DFD piece (right) and dropping onto the board (left), a collision with the pumpkin occurs.

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110027 |

FIGURE 3. Narrative for an activity on the arm the DFD game for aconditional algorithm.

TABLE 1. Games activities associated with programming skills.

Task	Programming skills	Game activity
Problem identification and	Problem solving (algorithmic thinking)	Helps
effi repeatable	Building algorithms	
	Simulation	thatstudent creates.
Sequence	Planning involving arranged actions in the order that produces accurate effects	Perform DFD, desktop testing and coding by dragging and dropping commands into the solution and check for correctness. The DFD-C provides support for this activity
Conditionals	The ability to make decisions based on certain conditions, rting the expression of multiple outcomes	Perform DFD, desktop testing and coding by dragging and dropping commands into the solution and check for correctness. The DFD-C provides support for this activity

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110027 |

Score Board

Part	Question/Concept/Problem	Type	Score (pts.)
Conceptual	Definition of Computer	Yes-or-no	0.5
	Problem analysis, coding, documentation, maintenance, testing & debugging	Matching definitions	0.5
	Methodology for computer-based problem solving	Ordering	0.5
	Priority of the operators	Multiple choice	0.5
Practical	Given a number of percentages for student final grade on a course, define his/her final grade asking for the required inputs, then show the outputs	Design a sequential structure flowchart (DFD)	3
	Given a flowchart on the calculation of total amounts depending on given values	Perform the desktop test	3
	Given a code written in C with inputs, data operations and outputs	Correct syntax errors	2

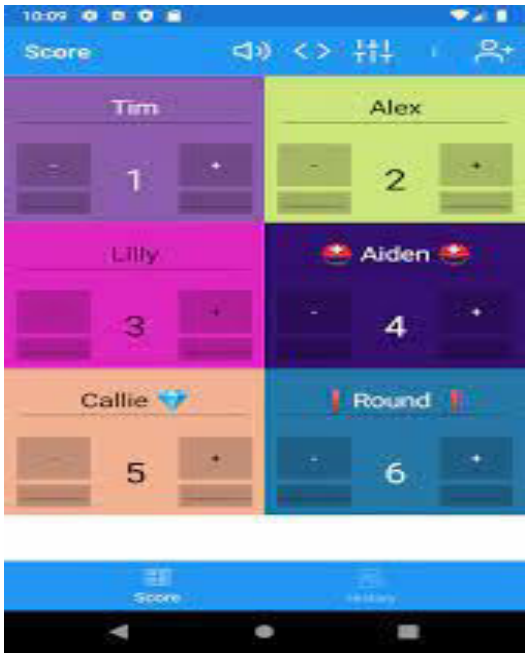

Figure 4. **Score Dashboard**

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 || A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 10, October 2023 ||

| DOI: 10.15680/IJIRCCE.2023.1110027 |

IV. ANALYSIS

As in all experimental research, this study also presents somelimitations that should be taken into account. Firstly, the number of participants is not as high as would be desired as it was not possible to gather more students. However, it is a typical sample size for most papers in this field and, given the normality of the data, the statistical results are robust, although the number of cases is low, because the necessary conditions to carry out this statistical inference with all the necessary guarantees are fulfilled.

V. RESULTS

The As one of the primary objectives of this study, the efficiencyof the proposed system was investigated considering its rolein improving students' learning achievement. The analyses of results were oriented to perform a descriptive analysis of thedata presented and then to deduce whether or not there had been a significant improvement between the control group and the test group of the variable evaluated in the first year of high school. The analysis was performed using the programRKWard.
Interpret the findings, highlighting improvements in student understanding of specific programming concepts and any statistically significant changes.

VI.CONCLUSION AND FUTURE WORK

Using Serious Games (SGs) to teach programming to High School students significantly increases their learning scores. This improvement is similar for male and female studentsas gender does not appear to be a discriminating factor. The DFD-C serious game was created using the LG-MG model taking into account the three kernels to design effective SGs for education: theory, contents and game design.
A future version of DFD-C will include topics such as repetitive cycles and functions, it will improve the immersionon the game, and it will also improve of the contest part, for instance, challenging the player to work harder and get rewards. A new experiment with a larger number of students from different countries (i.e. Spain and Ecuador) to alsotest social and individual differences in the learning of pro- gramming fundamentals and their perception of the gameful experience using the new version of DFD-C is intended to beconducted at a future date.

REFERENCES

1. Computer Science Teachers Association. (2012). Computer Science K-8: Building a Strong Foundation. [Online]. Available: http://csta.acm.org/Curriculum/sub/CurrFiles/CS_K8_Building_a_Foundation.pdf
2. H. Jacobsen. (2014). Five-Years-Olds Learn Coding in Schools to Prepare for Future Labour Market. EU News & Policy Debates, Across Languages
. [Online]. Available: EurActiv.com
3. M. Adam, M. Daoud, and P. Frison, ''Direct manipulation versus text-based programming: An experiment report,'' in Proc. ACM Conf. Innov. Technol. Comput. Sci. Edu., Jul. 2019, pp. 353–359, doi: 10.1145/ 3304221.3319738.
4. W. Feurzeig and S. Papert, ''The LOGO programming language,'' in ODP-Open Directory Project, 1967.
5. S. Papert, Mindstorms: Children, Computers, and Powerful Ideas. New York, NY, USA: Basic Books, 1980.
6. R. Muñoz, T. Barcelos, R. Villarroel, and I. F. Silveira, ''Using scratch to support programming fundamentals,'' Int. J. Comput. Thinking, vol. 1, no. 1, p. 68, Oct. 2017.
7. I. Ouahbi, F. Kaddari, H. Darhmaoui, A. Elachqar, and S. Lahmine, ''Learning basic programming concepts by creating games with scratch programming environment,'' Procedia Social Behav. Sci., vol. 191, pp. 1479–1482, Jun. 2015.
8. B. N. Serna Agudelo, E. Recalde España, G. A. Beltrán, and C. A. Cañón Recalde, ''El scratch como estrategia didáctica para desarrollar la explo- ración del medio en la educación inicial–Fase i y II –,'' Revista Inclusión Desarrollo, vol. 5, no. 2, pp. 21–38, Dec. 2018.

INNO SPACE
SJIF Scientific Journal Impact Factor

Impact Factor: 8.379

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details