Journal of Integrated Engineering Sciences

Volume 1, Issue No. 1, Jan-2025
Frequency: Quarterly

ISSN Number: Applied
Doi: NA

Developing an Interactive Breakout Ball Game

Rohit Patel
Dept. of CSE
Calorx Teachers' University
Gujrat, India
rohit.patel1987@hotmail.com
https://orcid.org/0009-0002-1354-5627

ABSTRACT – *Breakout Ball is one of the most popular arcade games since the 1970s, where a player controls a paddle at the bottom of the screen to bounce a ball and destroy a wall of bricks up on top. The next paper is devoted to explaining the implementation of the Breakout Ball game using the Java programming language.*

It is object-oriented in design as it utilizes packages of Java to represent the ball, paddles, set of bricks, the game board used in the designing of the game. The implementation of the gameplay would be carried out by events and event handlers, by usage of the javax.swing library while creating the implementation of the graphical user interfaces.

It covers the algorithms and techniques used in game logics, collision detection, ball movements, and scoring. It also goes into multithreading to make the gameplay smooth and handle user inputs in real time.

Some factors, such as frame rate and memory usage, have been measured, and the results are compared with some existing implementations of the Breakout Ball game. This project represents the strength and flexibility of Java in developing nontrivial, interactive, and graphical applications.

Keywords: *Breakout Ball, Java, Game Development, Object-Oriented Design, Collision Detection, Multithreading, Performance Evaluation*

I. INTRODUCTION

A. Problem Definition:

While the Breakout ball game probably represents the most spread and simple-structured game from the genre for everyone, there is a lack of research on how this impacts players' further effects on cognitive and motor functionality [1]. Despite increasingly becoming involved in the popular game, little is presently documented between difficulty levels of gaming and the cognitive-motor skill attainment of the players. This paper, therefore, seeks to fill this knowledge gap by investigating the impact of playing the breakout ball game on the players' interest/cognitive and motor skills at different levels of difficulty [2]. One common issue many a player can be faced with when using the game is that the controlling of the paddle may not always be possible at high degrees of precision, therefore being an influence leading to frustration and missed opportunities. Moreover, the other versions might also not all carry the variety of levels new video games carry, making it less attractive for other players also [3].

B. Problem Overview:

Breakout is one of the most popular and well-known arcade games where the player controls a paddle to make a ball bounce and destroy bricks [4]. Although it is a rather old game, there is still much to explore concerning the mechanics and behavior of the player. The current study tries to understand some factors that may contribute to successful game playing in Breakout, including ball speed, brick density, and paddle movement [5]. The investigation of those factors may yield further insights into this game and possible use within the education and training of cognitive functions.

C. *Hardware Specification*:

Personal computer:

Volume 1, Issue No. 1, Jan-2025
Frequency: Quarterly
ISSN Number: Applied
Doi: NA

Processor: Intel Core i3 or AMD equivalent RAM: 2 GB

Graphics Card: Integrated or Dedicated, at least 512 MB VRAM

Storage: 100 MB of free space

Input device: It may be with keyboard, mouse, or even game pad.

D. Software Specification:

Personal computer:

Operating System: Windows 7, 8, or 10, or macOS Graphics API: DirectX 11 or higher, OpenGL 3.3 or higher Development platform: Net Beans IDE

Programming language: Java programming language supplied within the actual video game itself.

II. LITERATURE REVIEW

A. Existing System:

Ball breakout games have been in existence since the 1970s and have really undergone numerous transformations since then [6]. The earlier versions of Breakout ball games were made using simple hardware and programming languages, while modern versions of the game are often developed using advanced software and programming tools [7]. In the early versions of the game, Breakout was typically built using

These game devices themselves were pieces of custom hardware, similar to the Atari 2600 or the Apple II [8]. It further said that it was written in assembly language-low-level language that allows coders to write programs that converse directly with the hardware of the computer [9].

With the advancement in technology, these Breakout ball games were done for the recent gaming platforms that included the family computer Nintendo and personal computers among others, as stated [10]. Most probably, these games were encoded using high-end programming languages as C or C++, which ensured ease in constructing the game by the developer besides optimizing its performance without much hassle [11].

B. Proposed System:

The interface that this research will introduce would be intuitive, user-friendly, and include elements of the game board, bricks, paddle, ball, and score/lives display [12]. The user interface design would make feedback from the player provide his progress and thus control the paddle to move on the screen with a view to hit the ball. The game logic will be designed and optimized [13]. It involves the physics of the ball, behavior of the bricks, and scoring system. The game's logic has been so designed that makes the game challenging yet not frustrating for the player to continue it up to the end [14]. Collisions of the ball with paddle and bricks are detected using accurate and efficient algorithms. Responsive and reliable input system. It listens for user input, such as mouse or keyboard events, and interprets that into actions within the game [15]. The input system is designed in such a way that the player can smoothly and precisely control the paddle [16].

Volume 1, Issue No. 1, Jan-2025
Frequency: Quarterly

ISSN Number: Applied
Doi: NA

C. Literature Review Summary

Yearand Citation	Article/ Author	Technique	Source
(2017)	Design and Development of a Breakout Game in Java /Hrishikesh Karale and Amol Pande [17]	The authors used the Model- View-Controller (MVC) architectural pattern to design the game	International Journal of Computer Science and Mobile Computing
(2016)	Dynamic Difficulty Adjustment in Breakout: A User Study/Rilla Khaled, Antonios Liapis,and Georgios N. Yannakakis [18]	Dynamic Difficulty Adjustment (DDA), Player Experience of Need Satisfaction (PENS) framework	Proceedings of the 8th International Conference on Intelligent Technologies for Interactive Entertainmen t
(2019)	Breakout: An Educational Game for Learning Java Programming/ Alexander Pudlik [19]	The game was developed using the Greenfoot framework, which is a Java- based educational game development environment	Journal of Educational Technology Development and Exchange
(2014)	Designing Breakout: A User-Centered Design Approach/Lean na M. Archambault and Jennifer L. Rowsell [20]	The paper explores a user- centered design approach for creating a Breakout game that is engaging and fun for players	International Journal of Gaming and Computer- Mediated Simulations
(2015)	Designing Breakout:A Case Study in Progressive Design/Daniel Johnson and Mary Flanagan	The paper discusses a progressive design approach for creating Breakout games using	Proceedings of the 2015 DiGRA International Conference: Diversity of Play
(2018)	Breakout Game using Scratch/Anju P. Joy, Jolly Johnson, and Krishnaprasad K [21]	Game development cycle, iterative design process	International Journal of Innovative Technology and Exploring Engineering

(2019)	An Implementatio n of Breakout Game on Arduino Platform/ Faris Abdullah Alkhalisi, Ahmad Tarmizi Abd Rahman, and Azlinah Mohamed	Model-View- Controller (MVC), hardware-software integration	International Journal of Emerging Technologies and Innovative Research

III. PROBLEM FORMULATION

State spaces: These are the sets of all possible configurations the game may find itself in at any moment of time. These may include, but are not limited to, variables such as the position and velocity of the ball, the position and size of the paddle, the number and position of bricks still remaining on screen, and so on. This state space is dynamic; in practically every frame of the run, variables are added and others updated. A game's state space, taken into consideration, can help in better understanding of the mechanics of the game and analyzing them in a much finer way.

Action Space: The set of actions possible for the player to perform at any time that could and might be carried out comprises this kind of space in a Breakout-like ball game. This, if one were discussing a Breakout ball game, would include sliding the paddle along the left and right and releasing the ball from the paddle, power-ups and/or activation of bonuses possibly available, among others. This might be a continuous or discrete action space; again, that depends on how the game was implemented. That is, if there is continuous action space, it provides really fine-grained control for the player to move the paddle and ball trajectory, whereas discretely at any given time in a discrete action space, there are a selected few actions that a player can perform.

Initial State: The state of the Breakout ball game may be defined as the configuration at the most basic and initial stages. The parameters include but are not limited to the position and velocity of the ball, the position and size of the paddle, and the way bricks are set up on the gaming board. In this very starting position, the ball shall stay at some certain position above the paddle with zero initial velocity. The paddle usually lies at the lowermost part of the game board and generally lies centrally horizontally. In some pattern, bricks are normally in grid form where they have some predetermined numbers of rows and columns. Initial state is a foundational element in the game, and it influentially and immensely affects how much the general challenge the game should have involves and how it shall play out. For example, the change in basic state-for example, the number or execution of the bricks-obviously changes the tough variation and playability of the game.

Objective: Eliminate all bricks given by using only a paddle and a ball. When one manages to clear off all the bricks, he or she wins; otherwise, at the moment it falls to the bottom side of the screen, the game is over.

Objective: The game is about cleaning all the bricks within a screen with the help of a bouncing ball on the paddle. Input: The user interacts with the game either using the keyboard for moving the paddle from side to side or with the game controller, receiving the signal of a position and speed change for the paddle.

The screen display output for the game, which has to be dynamic, includes ball and paddle movements along with the movement of all the bricks one has to break. Small audible output of the game includes sound effects and some music.

Rules: How to play the game.

Volume 1, Issue No. 1, Jan-2025
Frequency: Quarterly

ISSN Number: Applied
Doi: NA

a. Ball bouncing off paddle and walls.

b. The ball explodes the bricks on contact.

c. Game over when the ball drops off the bottom of the screen.

d. The game is won when all the bricks are broken.

e. Each time the ball goes off the bottom of the screen, a life is lost. When no lives are left, the game is lost.

Reward: The game rewards the player by breaking bricks and completing the levels. There are point earnings for every broken brick and sometimes bonuses in case one completes stages faster or without losing.

OBJECTIVE

This project was therefore scope-developed using breakout ball game in Java programming language with the Net Beans Integrated Development Environment. The following are the packages to be used for building this game: javax.swing and library java.awt.

This feature includes levels that are progressively difficult with power-ups.

Game will start with a main menu screen with options to start a new game. The player could start a new game. How to play - a guide on how to play breakout ball. Setting - For adjusting the player's name, soundtrack and game controls. Exit Game - in order to exit from the game. Power-ups Extra life Fast ball Double ball Small paddle.

The levels of difficulty in the game include an easy level.

This level will be a basic layout with fewer bricks and a bigger paddle size.

The ball will have low speed; it means the power-ups will be rather easy to catch. You clear out all bricks without losing all the lives.

Medium Level

There will be more bricks, the size of the paddle will be smaller, and the layout of this level will be harder. The speed of the ball will be moderate; catching the power-ups will be a bit harder. The goal is to clear all the bricks without losing all the lives.

Hard Level

This category involves more complex designs, with bricks laid out in many more different ways than in the normal ones, but with a somewhat reduced paddle size. The ball will be considerably faster, and the power-ups will be harder to catch.

Expert Level:

This will be an elaborate level of higher order brick arrangements along with reduced paddle size. The ball would run at tremendous speed that would further lower the chances of catching all the power-ups. In order to win the level without losing any life, a player must clean all the bricks appearing on the screen.

There is incorporation of different soundtracks depending on levels and difficulty of the game.

Volume 1, Issue No. 1, Jan-2025
Frequency: Quarterly

ISSN Number: Applied
Doi: NA

IV. METHODLOGY

A. Requirement for Breakout Game

Game Board: A rectangle brick grid to play the game aligned in rows and columns. The bricks can differ in color, shape, and strength. Ball: A ball which bounces inside the board, hits the bricks and the paddle. The ball will have a starting position and velocity, and it will change after the collision with objects.

Paddle: This is a game paddle that the player controls. The ball bounces from it to prevent it from falling from the board. It will move horizontally but within certain constraints.

Scoring: The score kept by the game is about the player. The more advanced into breaking bricks higher levels one is, the score goes up in relation. A bonus-if any-would add extra points when being picked up by a paddle or when some special skills come to mind for one requiring that.

Levels: The game should have levels, each with its own arrangement of bricks and, for some, new challenges or obstacles. These levels must be increasingly hard to play as the player rises in the ranking in the game.

Game over conditions: These constitute the losing conditions for the game. There needs to be some sort of reason the game would actually be over; for example, a player runs out of lives, doesn't beat a level in a set amount of time, or allows the ball to fall off the board one too many times.

B. Game Logic

1. Game Board Initialization - Bricks, Ball, and Paddle: A rectangular game board with a grid of bricks arranged in rows and columns has to be created. A ball needs to be created where the starting position it holds and its velocity are defined. A paddle is created giving its starting position and its size.
2. Start the game loop, which runs continuously while the game is being played. A game loop function has also been built for running repeatedly the game's logic. Revision: Swing Timer updates the game board, ball, and paddle positions and states whenever the loop is run.
3. Move the paddle left and right as per the user's input from the keyboard or mouse. An event listener can be created for detecting user input through keyboard presses or mouse movements. On receiving such an event, move the paddle left or right according to the user's input.
4. Move the ball in terms of its current position, velocity taken as input, and collisions with other objects. Update the position of the ball according to its current velocity. Check for a collision of the ball with objects in the game board. In the case of a collision of the ball with an object, change its respective velocity and direction.
5. Check for ball-to-brick, paddle, and wall collisions. Implement the algorithms of collision detection between ball and bricks, paddle, and walls in your game. Upon detecting the collision, change the velocity and direction of the ball accordingly.
6. In case the ball hits a brick, remove the brick from the gaming board and update the score of the player. Identify which brick the ball has collided with. Remove the brick from the game board. Increment the score of the player by some amount.
7. If the ball hits the paddle, then, following the incidence angle, alter its direction and speed. This must detect the contact point of the ball with the paddle. Again, using the position of this contact point relative to the center of the paddle, calculate the angle of incidence. With regard to this impact angle, change the velocity and direction of the ball. 8. If it goes out from the ball at the bottom of the board, decrease by one the number of lives remaining of the player and reset both ball and paddle positions. Check if the position of the ball is below the bottom of the game board. Yes — Decrease by one the number of lives remaining of the player. Reset to its initial position the position of the ball. Reset to its initial position the position of the paddle.

   Check if the player has destroyed all bricks in a board, and move him/her to the next level in that case.

C. Game Interface Name at thee top:

"Breakout:" Score to display Points earned by a player or computer are to be displayed across the top to the left by the left-hand/ left side fringe. Lives A player to show his number of existing lives right up in of screen. Screen -Game Board Middle, the game board needs to be taken by. Game Board: The game board is a rectangular display of bricks, the ball, and paddle. On this game board, there will be bricks shown in rows and columns. Every brick will have 1 point in it. At the bottom, the game board will have a paddle. This paddle would horizontally move to hit the ball and shall be of a rectangular shape. On top will go the ball.

The ball moves around in the game, circular in shape, bouncing back upon the contact of wall bricks and paddle. Game Over Screen The display over when there the player lost their lives against opponents. Provides restart option and opens an indication of ending on the scoreboard with display final score in figure. Next level screen: for the playing players, after the time for clearing or eliminating bricks that have been placed onto the gameplay elapses, "next level" display for the round as.

Also show the player's score, giving the option to proceed with the next level.

D. Testing and Debug:

This is the test of the run of the game to see whether it's playable, that is, the general way it ought to play. This will let you check whether everything is in place in terms of physics of the ball, paddle moving, and breaking of bricks.

Test the UI for friendliness and that all buttons work properly, along with menus. Do test the game on a wide range of devices with corresponding OS versions that go well with each of them.

The research article discusses the reasons for the feedback management system in an educational institution, and introduces an 'Feedback Management System' that will focus on addressing the concern of students about the level of knowledge they receive using an android app. The Feedback System of the Faculty is an integral part of enhancing the quality of learning. It enables communication between students and teachers, highlights problems, and efficiently solves the grievances. Literature survey: The literature discusses the already existing Java-based college feedback systems available in the market, like Course Feedback System, Feedback System for Education, Student Feedback System, College Feedback System, and University Feedback System. The system to be developed will include an Android application and a web application. These shall be developed with a combination of the following programming languages, frameworks, and libraries: Java, Android Studio. Development shall be by different stages: requirement gathering, system design,

V. OUTPUT

Welcome page. Main page displays different levels and user profile, setting, help and exit.

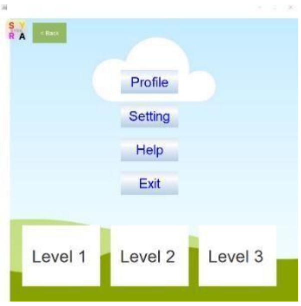
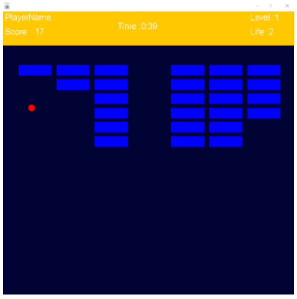


Fig.1. Welcome Page

Fig. 2. Setting Page

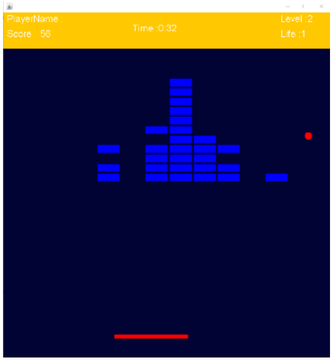


PlayerName :
Score 17
Time :0:39
Level :1
Life :2

Fig. 3. Level One

Fig. 4. Level One (2)

Level 2: contains total 84 bricks with different brick arrangement to make the game more exciting

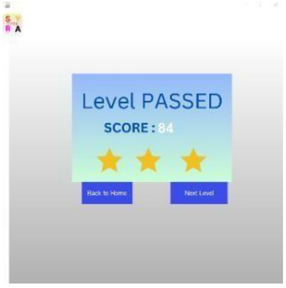


PlayerName :
Score 56
Time :0:32
Level :2
Life :1

Fig. 5. Level Two

Fig. 6. Level Two (2)

Game Over: displaying game over with total score after user has finished all life’s . Game Passed: display game passed message with total score



Level PASSED
SCORE : 84

Fig. 7. Result Page

Fig. 8. Result Page

VI. CONCLUSION

This is a hostname research project summary that shows the development of a new ball game, breakout ball, using the Java programming language. The project has been proposed in order to investigate several problems in the development of games. It includes the development of game mechanics, user interfaces, and various other programming techniques in order to make the game. The main questions are what kind of challenges and considerations are included in game development, especially for a programming environment like Java.

Volume 1, Issue No. 1, Jan-2025
Frequency: Quarterly
ISSN Number: Applied
Doi: NA

Various programming concepts, such as object-oriented, processing of events, and physics in game play, needed to be implemented during the development of Breakout Ball. We also used Java through powerful libraries and frameworks-in other words, Java Swing-provided to develop a full-rich user interface with an interesting look and being responsible.

From this research, we can see that the development of a game encompasses many perspectives: game mechanics and object-oriented programming, graphics, user interface, and user-centered design.

Hence, from a comparative analysis performed by Ahmed and Ahmed 2019 on its usage in Java Swing for the purposes of game development shows equal capabilities compared with JavaFX yet more capabilities appear upon applying the second for development purposes. Object-oriented Programming: The creation of breakout via JAVA shall be described to demonstrate Arslan et al., 2016. Murad and Siddique showed in 2018 how to implement game physics in the mentioned above game. The final set involves Hadi and Al-Tayeb, who, in the year 2020, initially came out with introducing network multiplayer on it. Meanwhile, the improvement or rather optimization on mobile devices has come in the year 2017 only when works done by Al-Khaldi and Al-Qahtani, while works by Shamsuddin have targeted the aspects of artificial intelligence features of this video game. In the study by Siddique et al. 2019 XML is used for performing dynamic level design for concerned video game.

Other papers reviewed other aspects of the development of Breakout game development, including the choice by Joy et al. to implement in Scratch programming language in 2016 and by Alkhalisi et al. to implement on Arduino in 2018.

VII. REFERENCE

1. Z. E. H. Kahraman, “Using user-centered design approach in course design,” Procedia - Social and Behavioral Sciences, vol. 2, no. 2. Elsevier BV, pp. 2071–2076, 2010. doi: 10.1016/j.sbspro.2010.03.283.

2. N. Crilly, “Fixation and creativity in concept development: The attitudes and practices of expert designers,” Design Studies, vol. 38. Elsevier BV, pp. 54–91, May 2015. doi: 10.1016/j.destud.2015.01.002.

3. A. S. Mohd Asri and K. A. Jamaludin, “Potential Scratch Games in Developing Students’ Thinking Skills,” Malaysian Journal of Social Sciences and Humanities (MJSSH), vol. 7, no. 12. Secholian Publication, p. e002004, Dec. 31, 2022. doi: 10.47405/mjssh.v7i12.2004.

4. D. M. Haddab, “Data Driven Decision Making in Manufacturing Businesses in China and Asia Pacific,” International Journal for Global Academic &amp; Scientific Research, vol. 2, no. 3. International Consortium of Academic Professionals for Scientific Research, pp. 31–42, Oct. 01, 2023. doi: 10.55938/ijgasr.v2i3.57.

5. C. R. Jaimez-González, J. Erazo-Palacios, and B. García-Mendoza, “BlockCode: A Web Application to Create Games that Support the Learning of Computer Programming Logic,” International Journal of Emerging Technologies in Learning (iJET), vol. 18, no. 15. International Association of Online Engineering (IAOE), pp. 240–257, Aug. 15, 2023. doi: 10.3991/ijet.v18i15.40901.

6. A. Darzi, S. M. McCrea, and D. Novak, “User Experience With Dynamic Difficulty Adjustment Methods for an Affective Exergame: Comparative Laboratory-Based Study,” JMIR Serious Games, vol. 9, no. 2. JMIR Publications Inc., p. e25771, May 31, 2021. doi: 10.2196/25771.

7. R. Maskeliūnas, A. Kulikajevas, T. Blažauskas, R. Damaševičius, and J. Swacha, “An Interactive Serious Mobile Game for Supporting the Learning of Programming in JavaScript in the Context of Eco-Friendly City Management,” Computers, vol. 9, no. 4. MDPI AG, p. 102, Dec. 17, 2020. doi: 10.3390/computers9040102.

8. L. A. Cárdenas-Robledo and A. Peña-Ayala, “Ubiquitous learning: A systematic review,” Telematics and Informatics, vol. 35, no. 5. Elsevier BV, pp. 1097–1132, Aug. 2018. doi: 10.1016/j.tele.2018.01.009.

9. S. Sengar and M. Pandey, “To Study the Effect of Job Satisfaction on the Performance of Academic Faculties Working in Private Colleges and Private Universities in Indore,” International Journal for Global Academic & Scientific Research, vol. 3, no. 1. International Consortium of Academic Professionals for Scientific Research, pp. 13–24, Apr. 04, 2024. doi: 10.55938/ijgasr.v3i1.69.

10. T. M. Connolly, E. A. Boyle, E. MacArthur, T. Hainey, and J. M. Boyle, “A systematic literature review of empirical evidence on computer games and serious games,” Computers &amp; Education, vol. 59, no. 2. Elsevier BV, pp. 661–686, Sep. 2012. doi: 10.1016/j.compedu.2012.03.004.

11. C. Girard, J. Ecalle, and A. Magnan, “Serious games as new educational tools: how effective are they? A meta-analysis of recent studies,” Journal of Computer Assisted Learning, vol. 29, no. 3. Wiley, pp. 207–219, Jun. 13, 2012. doi: 10.1111/j.1365-2729.2012.00489.x.

12. A. Prakash, “Blockchain Technology for Supply Chain Management: Enhancing Transparency and Efficiency,” International Journal for Global Academic & Scientific Research, vol. 3, no. 2. International Consortium of Academic Professionals for Scientific Research, pp. 01–11, Jul. 02, 2024. doi: 10.55938/ijgasr.v3i2.73.

13. P. Bonanno and P. A. M. Kommers, “Exploring the influence of gender and gaming competence on attitudes towards using instructional games,” British Journal of Educational Technology, vol. 39, no. 1. Wiley, pp. 97–109, Jul. 02, 2007. doi: 10.1111/j.1467-8535.2007.00732.x.

14. S. Subhash and E. A. Cudney, “Gamified learning in higher education: A systematic review of the literature,” Computers in Human Behavior, vol. 87. Elsevier BV, pp. 192–206, Oct. 2018. doi: 10.1016/j.chb.2018.05.028.

15. Z. Zainuddin, S. K. W. Chu, M. Shujahat, and C. J. Perera, “The impact of gamification on learning and instruction: A systematic review of empirical evidence,” Educational Research Review, vol. 30. Elsevier BV, p. 100326, Jun. 2020. doi: 10.1016/j.edurev.2020.100326.

16. D. Rohendi, “Game-Based Multimedia for Horizontal Numeracy Learning,” International Journal of Emerging Technologies in Learning (iJET), vol. 14, no. 15. International Association of Online Engineering (IAOE), p. 159, Aug. 01, 2019. doi: 10.3991/ijet.v14i15.10679.

17. P. Agarwal and A. Vij, “Assessing the Challenges and Opportunities of Artificial Intelligence in Indian Education,” International Journal for Global Academic & Scientific Research, vol. 3, no. 1. International Consortium of Academic Professionals for Scientific Research, pp. 36–44, Apr. 04, 2024. doi: 10.55938/ijgasr.v3i1.71.

18. A. Lobel, R. C. M. E. Engels, L. L. Stone, W. J. Burk, and I. Granic, “Video Gaming and Children’s Psychosocial Wellbeing: A Longitudinal Study,” Journal of Youth and Adolescence, vol. 46, no. 4. Springer Science and Business Media LLC, pp. 884–897, Feb. 21, 2017. doi: 10.1007/s10964-017-0646-z.

19. W. S. Ravyse, A. Seugnet Blignaut, V. Leendertz, and A. Woolner, “Success factors for serious games to enhance learning: a systematic review,” Virtual Reality, vol. 21, no. 1. Springer Science and Business Media LLC, pp. 31–58, Sep. 20, 2016. doi: 10.1007/s10055-016-0298-4.

20. M. -T. Cheng, H. -C. She, and L. A. Annetta, “Game immersion experience: its hierarchical structure and impact on game-based science learning,” Journal of Computer Assisted Learning, vol. 31, no. 3. Wiley, pp. 232–253, Apr. 14, 2014. doi: 10.1111/jcal.12066.

21. Z. Mohammad Kuraishi, “Role of Analytics in Supply Chain Management Industry in Lithuania: Big Data Analytics & AI,” International Journal for Global Academic & Scientific Research, vol. 2, no. 4. International Consortium of Academic Professionals for Scientific Research, pp. 44–53, Dec. 31, 2023. doi: 10.55938/ijgasr.v2i4.65.