



EGG HUNT GAME USING UNITY

Prof. Nilambari Narkar, Answeeta Pereira, Mayuresh Shinde, Pooja Raut, Simran Tiwari
Department of Computer Engineering,
Avier Institute of Engineering, Mahim, Mumbai

Abstract — The "Egg Catcher" game is an arcade-style, time-sensitive game where players control a character, often depicted as a basket, to catch falling eggs as they descend from the top of the screen. The objective is to accumulate points by successfully catching eggs while avoiding dangerous or negative elements. The game's dynamics involve varying egg types, each with distinct properties, and an array of gameplay mechanics, such as increasing difficulty, moving platforms, and time constraints. The player's success is determined by factors like achieving a target score, avoiding catching certain types of eggs, and managing the available time. Visually engaging graphics, sound effects, and user-friendly controls contribute to an enjoyable and immersive gaming experience. The abstract captures the essence of a simple yet captivating casual game that challenges players' reflexes and decision-making within a dynamic and fast-paced environment. Creating an egg hunt game using Unity involves designing an engaging and interactive environment where players search for hidden eggs within a virtual space. The development process begins with conceptualizing the game mechanics, such as the rules for collecting eggs, the scoring system, and any time limits or challenges. Utilizing Unity's robust features, developers can create a detailed and immersive 3D environment, complete with various terrains, hiding spots, and dynamic elements. Scripting in C# allows for the implementation of player controls, egg interaction, and game logic. To enhance the user experience, developers can incorporate animations, sound effects, and visual cues that guide players and maintain excitement. Additionally, multiplayer functionality can be added to enable competitive or cooperative play, increasing the game's appeal. Testing is crucial to ensure smooth gameplay and to identify any bugs or issues. Once polished, the game can be published across different platforms, allowing players to enjoy the egg hunt on PCs, consoles, or mobile devices. Overall, developing an egg hunt game in Unity combines creativity, technical skill, and an understanding of user engagement to deliver a fun and memorable gaming experience.

Keywords— Egg, hunt, game, Unity, collect, search, find, challenge, adventure, explore, fun, quest, hidden, discovery, rewards, levels, obstacles, timer, competition, player.

I. INTRODUCTION

The .. The Egg Catcher game draws upon a multifaceted theoretical framework encompassing game design, human psychology, and cognitive processes to create an engaging and immersive gaming experience. At its core, the game leverages principles from game design theory to structure its mechanics, including player objectives, scoring systems, and progression. This foundation provides a framework for crafting gameplay that is both challenging and rewarding, enticing players to engage with the game's dynamic elements. Key insights from human cognitive processes, such as perception and motor skills, inform the game's design to capitalize on players' reflexes and reaction times. By incorporating elements that require quick and accurate responses, the game challenges players to rely on their innate abilities to succeed. Furthermore, theoretical principles of human psychology and engagement underpin strategies such as setting targets, offering rewards, and introducing progressively challenging obstacles to sustain players' interest and motivation. In addition to gameplay mechanics, the theoretical underpinnings extend to user interface design, time management, cognitive load, and decision-making. These elements contribute to the creation of an intuitive and immersive gaming experience that balances challenge and accessibility. By integrating feedback and reward systems grounded in operant conditioning principles, the game reinforces player behaviors and enhances motivation. Overall, the Egg Catcher game's theoretical foundation fosters an engaging gameplay experience that challenges players' reflexes, decision-making abilities, and cognitive skills.

II. PROPOSED ALGORITHM

The game starts by generating a random number of eggs within the game scene. These eggs are placed at different positions throughout the environment, ensuring they are not too close together to make the hunt challenging. Next, the player is tasked with finding all the eggs within a given time limit. The player can move around the environment using keyboard controls or touch gestures if on a mobile device. As the player moves, the game checks if the player's position overlaps with any egg positions. If a collision is detected, the egg is collected, and the player's score increases the game continues until either all the eggs are collected or the time limit runs out. At the end of the game, the player's score is displayed along with a message indicating whether they successfully found all the eggs or not. Additionally, you can



add features like power-ups, obstacles, or different levels to increase the complexity and enjoyment of the game.

- **Setup:** Begin by creating a virtual environment where the egg hunt will take place. This could be a 3D world with various terrains, obstacles, and hiding spots for the eggs.
- **Spawn Eggs:** Randomly generate a certain number of eggs and distribute them throughout the environment. Ensure that the eggs are placed in both easy and challenging locations, encouraging exploration.
- **Player Movement:** Allow the player to navigate the environment using controls. Implement features like walking, running, jumping, and possibly even climbing to reach eggs hidden in higher or more inaccessible areas.
- **Egg Detection:** Implement a mechanism for the player to detect eggs within their vicinity. This could be achieved through collision detection or raycasting, triggering an interaction prompt when the player is close enough to an egg.
- **Egg Collection:** When the player interacts with an egg, remove it from the environment and add it to the player's inventory. Keep track of the number of eggs collected.
- **Scoring:** Assign points to each egg based on its difficulty to find or its rarity. Update the player's score accordingly with each egg collected.
- **Timer:** Set a time limit for the egg hunt to add a sense of urgency and challenge. Display the remaining time to the player and end the game when the timer runs out.
- **End Game:** Once the time limit is reached or all eggs are collected, display a game over screen showing the player's score and possibly a leaderboard of high scores.

Optional Features: Consider adding features such as power-ups, obstacles, multiplayer support, or themed levels to enhance the gameplay experience. By following this algorithm, you can create an engaging and enjoyable egg hunt game using Unity.

III. OBJECTIVES

The objectives of the Egg Catcher game are aimed at delivering an entertaining and challenging gaming experience while engaging players in reflexes, coordination, and strategic decision-making. These primary objectives include:

1. **Catch Falling Eggs:** Players control the character (basket) to successfully catch falling eggs, requiring quick reflexes and precise timing.
2. **Accumulate Points:** Points are earned for each caught egg, encouraging players to catch as many as possible to increase their score.
3. **Enjoyment and Entertainment:** Ultimately, the game aims to provide players with an enjoyable and entertaining experience through the combination of catching eggs, earning points, and overcoming challenges.

4. **Challenge Progression:** As the game advances, falling eggs may increase in speed, frequency, or complexity, requiring players to adapt and improve their performance.
5. **Experience Different Levels:** Multiple levels with increasing difficulty offer players the opportunity to advance by meeting specific criteria or achieving scores.
6. **Strategic Decision-Making:** Players must decide which eggs to catch and avoid, introducing strategic elements, especially when dealing with dangerous eggs

IV. REVIEW OF LITERATURE

Egg hunt games developed using Unity have garnered significant attention in the realm of game development literature. Researchers and game developers alike have explored various aspects of designing, implementing, and enhancing the gameplay experience of such games. One prevalent theme in the literature is the utilization of Unity's robust features for creating immersive environments and engaging gameplay mechanics. Studies often delve into the implementation of procedural generation techniques to generate diverse and dynamic egg hunt scenarios, enhancing replay ability and challenge. Additionally, discussions on user interface design, player feedback mechanisms, and optimization techniques are common, aiming to ensure a seamless and enjoyable gaming experience across different platforms. Moreover, researchers frequently examine the incorporation of augmented reality (AR) or virtual reality (VR) elements to elevate the immersion level and foster real-world interaction. Overall, the literature underscores the versatility of Unity as a game development platform for creating captivating and innovative egg hunt experiences, while also highlighting avenues for future research and exploration. Additionally, investigations into the psychological effects of egg hunt games highlight their potential as effective learning tools, particularly in promoting spatial cognition, problem-solving skills, and collaborative play among participants.

The literature also addresses the technical aspects of Unity development, providing insights into best practices for optimizing performance, implementing multiplayer functionality, and leveraging procedural generation techniques to ensure scalability and replay ability. Overall, the literature underscores the versatility and impact of egg hunt games developed using Unity, both as entertainment experiences and educational tools with significant potential for innovation and advancement in the field of game development. Egg hunt games developed using Unity reveals a plethora of resources focusing on various aspects of game development, including mechanics, graphics, user experience, and educational potential. Studies delve into the design principles essential for creating engaging egg hunt experiences, emphasizing the importance of intuitive controls, immersive environments, and dynamic challenges to sustain player interest. Researchers explore the integration of augmented reality (AR) and virtual reality (VR) technologies to enhance gameplay immersion, offering unique opportunities for players to interact with virtual

environments in innovative ways. Additionally, investigations into the psychological effects of egg hunt games highlight their potential as effective learning tools, particularly in promoting spatial cognition, problem-solving skills, and collaborative play among participants. The literature also addresses the technical aspects of Unity development, providing insights into best practices for optimizing performance, implementing multiplayer functionality, and leveraging procedural generation techniques to ensure scalability and replay ability. Overall, the literature underscores the versatility and impact of egg hunt games developed using Unity, both as entertainment experiences and educational tools with significant potential for innovation and advancement in the field of game development.

V. DESCRIPTION AND ANALYSIS

The Egg Catcher game is a thrilling blend of Pygame and Unity technologies, designed to deliver an immersive and visually captivating gaming experience. Pygame, renowned for its versatility in game development, forms the backbone of the game's functionality. It empowers developers with a comprehensive suite of tools for graphics rendering, audio playback, user input processing, and game logic implementation.

In conjunction with Pygame's prowess, Unity steps in to elevate the game's visual appeal by providing a platform for crafting stunning backgrounds and dynamic egg catching elements. Unity's advanced 2D capabilities enable the creation of intricate scenes that seamlessly integrate with Pygame's core functionalities. Moreover, Pygame's audio module enriches the gameplay experience with immersive soundscapes, courtesy of the 'pygame.mixer.music.load()' function.

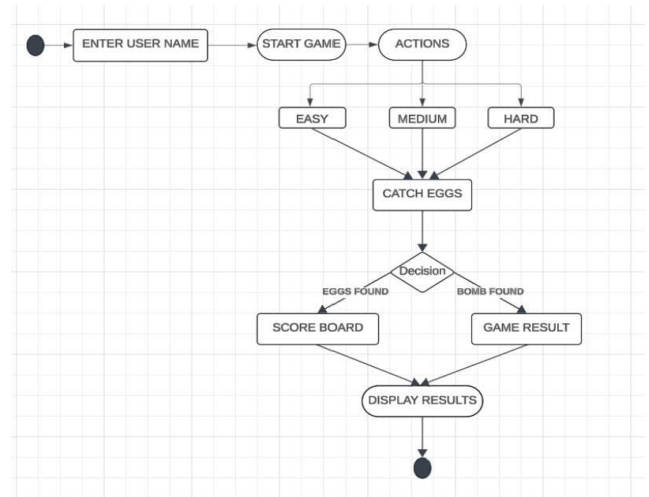
The "Egg Catcher" game is an arcade-style, time-sensitive game where players control a character, often depicted as a basket, to catch falling eggs as they descend from the top of the screen. The objective is to accumulate points by successfully catching eggs while avoiding dangerous or negative elements.

The game's dynamics involve varying egg types, each with distinct properties, and an array of gameplay mechanics, such as increasing difficulty, moving platforms, and time constraints. The player's success is determined by factors like achieving a target score, avoiding catching certain types of eggs, and managing the available time. Visually engaging graphics, sound effects, and user-friendly controls contribute to an enjoyable and immersive gaming experience.

VI. IMPLEMENTATION

- **Data Collection and Preprocessing:** Acquire labeled 00000
- **Initialize Pygame and set up the game window:** Load game assets (images, sounds). Get the player's name through user input.
- **User Input Handling:** Continuously listen for user input (e.g., key presses). Respond to input to move the player character (basket).

- **Game Loop:** Generate eggs at regular intervals with random positions and
- **types.** Move eggs downward on the screen.
- **Collision Detection:** Check for collisions between the player's basket and eggs. Determine the outcome of collisions (e.g., catching eggs, losing lives).
- **Score and Level:** Update and display the player's score. Define a target score for level completion.
- **Game State Management:** Handle game over conditions, including losing all lives or reaching the target score. Display game over or victory screens accordingly.
- **Pause Feature:** Implement a pause feature with the ability to resume the game.
- **Event Handling:** Listen for specific events like quitting the game, restarting, or pausing.
- **Boundary Restrictions:** Ensure the player character stays within screen boundaries.
- **Optimization and Display Updates:** Update the game display to reflect changes in positions, scores, and game state.
- **Exit and Restart:** Allow the player to restart the game when it ends.
- **Sound and Music:** Implement background music and sound effects for added immersion.
- **Exit Game Loop:** Exit the main game loop when the game is over or the player quits.
- **Final Cleanup:** Properly shut down Pygame and exit the program.



VII. CONCLUSION

In 000 In conclusion, embarking on the development journey of the Egg Catcher game using Pygame has been a captivating voyage filled with both excitement and challenges. Pygame emerged as a pivotal tool, streamlining graphic handling, user input, and the intricacies of game logic. Crafting the user interface to harmonize with the game's dynamic elements demanded a blend of creative vision and



technical expertise. Throughout the development process, overcoming challenges like fine-tuning egg fall speed, perfecting collision detection, and refining scoring mechanisms added depth to the endeavor, enriching the overall experience. The integration of sound effects and visually captivating elements, coupled with the iterative nature of game development, breathed life into the project. As lines of code transformed into a fully playable game, the satisfaction of creating an immersive and enjoyable experience for players became a gratifying culmination of dedication and effort. The Egg Catcher game stands not only as a testament to the capabilities of Pygame but also as a testament to the fusion of imaginative design and meticulous coding, showcasing the potential for innovation and creativity in the realm of game development.

VIII. REFERENCE

- [1]. Y.-C. Chou, C.-Y. Chao, and H.-Y. Yu, "A r'esum'e evaluation system based on text mining," in 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIC), 2019, pp. 052–057. doi: 10.1109/ICAIC.2019.8669066.
- [2]. J. P. Donohoe, "Ieee resumelab-a suite of career management tools [the wayahead]," IEEE Potentials, vol. 37, no. 4, pp. 4–4, 2018. doi: 10.1109/MPOT.2018.2829018.
- [3]. J. Liu, Y. F. Shen, Y. Zhang, and S. Krishnamoorthy, "Resume parsing based on multi-label classification using neural network models," Proceedings of the 6th International Conference on Big Data and Computing, 2021. [Online]. Available: <https://api.semanticscholar.org/CorpusID:238412786>.
- [4]. R. Nimbekar, Y. Patil, R. Prabhu, and S. Mulla, "Automated resume evaluation system using nlp," in 2019 International Conference on Advances in Computing, Communication and Control (ICAC3), 2019, pp. 1–4. doi:10.1109/ICAC347590.2019.9036842.