Mini Project 2 Report

Title of the Project: Bounce Game
Name of the Institution: New Uzbekistan University
Name of Department: Software Engineering
Name of Course: Game Development
Team Members: Zarshedjon Nasimov, Rustam Kodirov, Sukhrobjon Jalolov, Sarvarbek Azimjonov
Date of submission: 03.12.2024

New
Uzbekistan
University

Table of Contents

Section 1: Introduction

The *Bounce Game* is an ambitious game development project that reimagines the nostalgic essence of the classic Bounce Tales game, blending it with modern gaming elements to deliver a fresh, dynamic, and engaging experience. Developed as part of the Game Development curriculum for Fall Semester 2024, this project serves as a hands-on opportunity for students to apply their theoretical knowledge in Unity and C# programming. The primary objective is to design a fully functional 2D arcade game that features multiple levels, each with unique designs, challenges, and increasing difficulty, offering players a progressive and immersive experience. The game incorporates obstacles, enemies, interactive puzzles, and power-ups to enhance gameplay, showcasing a balance between creativity and technical mastery.

Structured to promote teamwork and collaboration, the project divides responsibilities into key roles, including project management, environment design, gameplay mechanics development, and user interface creation. Each role contributes significantly to the game's overall quality and cohesion. Team members work together to create visually appealing environments, develop responsive player controls, design intricate gameplay mechanics, and implement intuitive and engaging UI elements. This structured approach reflects real-world game development practices, fostering effective communication and teamwork skills.

The *Bounce Game* stands out with its integration of diverse game levels and power-ups. Each level introduces distinct themes, environments, and challenges, ensuring players remain engaged and entertained. Players must navigate through complex terrains, avoid obstacles, defeat enemies, and solve puzzles to progress. Power-ups play a pivotal role in enhancing gameplay by granting temporary advantages, such as increased speed, invincibility, or bonus points, adding layers of strategy and excitement to the experience.

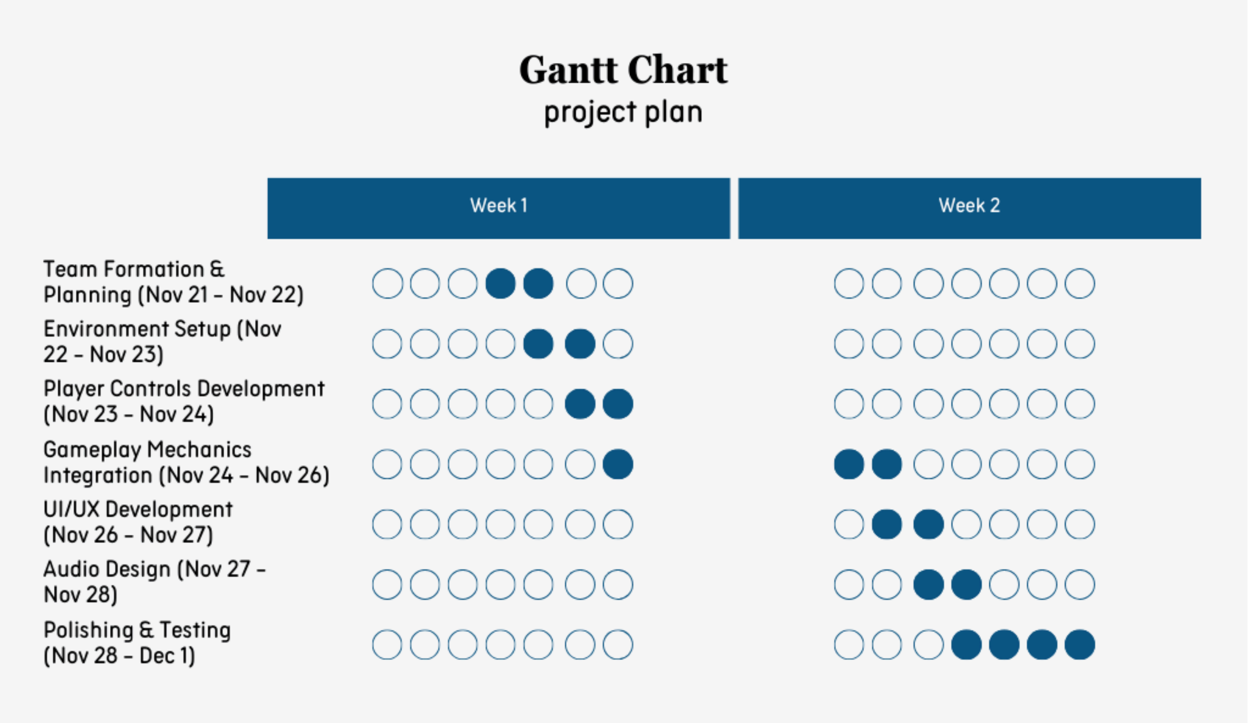
In addition to its engaging mechanics, the project integrates physics-based gameplay, immersive graphics, dynamic sound effects, and smooth user interaction. By prioritizing rigorous testing, debugging, and optimization, the final game delivers a polished and enjoyable experience. Beyond its technical achievements, the project encourages students to embrace creative problem-solving, innovate solutions, and deepen their understanding of game development principles. Ultimately, the *Bounce Game* is more than an academic exercise—it's a testament to the team's ability to merge creativity and technical expertise into a modernized, highly engaging version of a beloved classic.

Section 2: Project Management

Team Members:

Zarshedjon Nasimov - Project Manager & Audio Designer
Sarvarbek Azimjonov - Environment Designer
Sukhrobjon Jalolov - Player Controls Developer & Game Play Mechanics Developer
Rustam Kodirov - UI/UX Designer

Project Timeline:



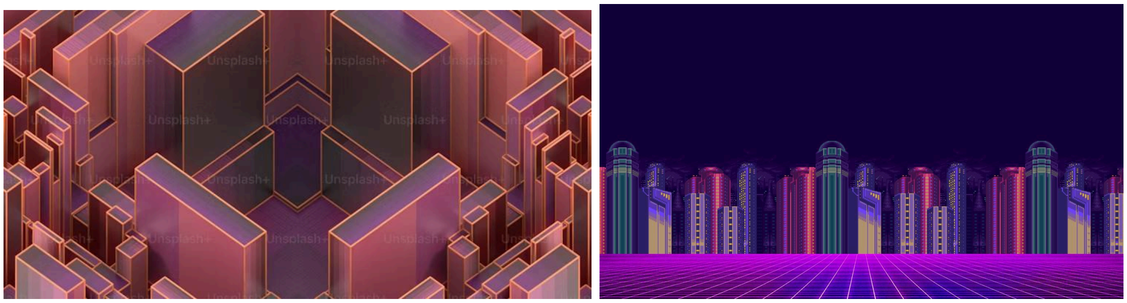
Gantt Chart
project plan

	Week 1	Week 2
Team Formation & Planning (Nov 21 - Nov 22)	○○○●●○○	○○○○○○○
Environment Setup (Nov 22 - Nov 23)	○○○○●●○	○○○○○○○
Player Controls Development (Nov 23 - Nov 24)	○○○○○●●	○○○○○○○
Gameplay Mechanics Integration (Nov 24 - Nov 26)	○○○○○○●	●●○○○○○
UI/UX Development (Nov 26 - Nov 27)	○○○○○○○	○●●○○○○
Audio Design (Nov 27 - Nov 28)	○○○○○○○	○○●●○○○
Polishing & Testing (Nov 28 - Dec 1)	○○○○○○○	○○○●●●●

Section 3: Environment Design

Background and Visual Aesthetics

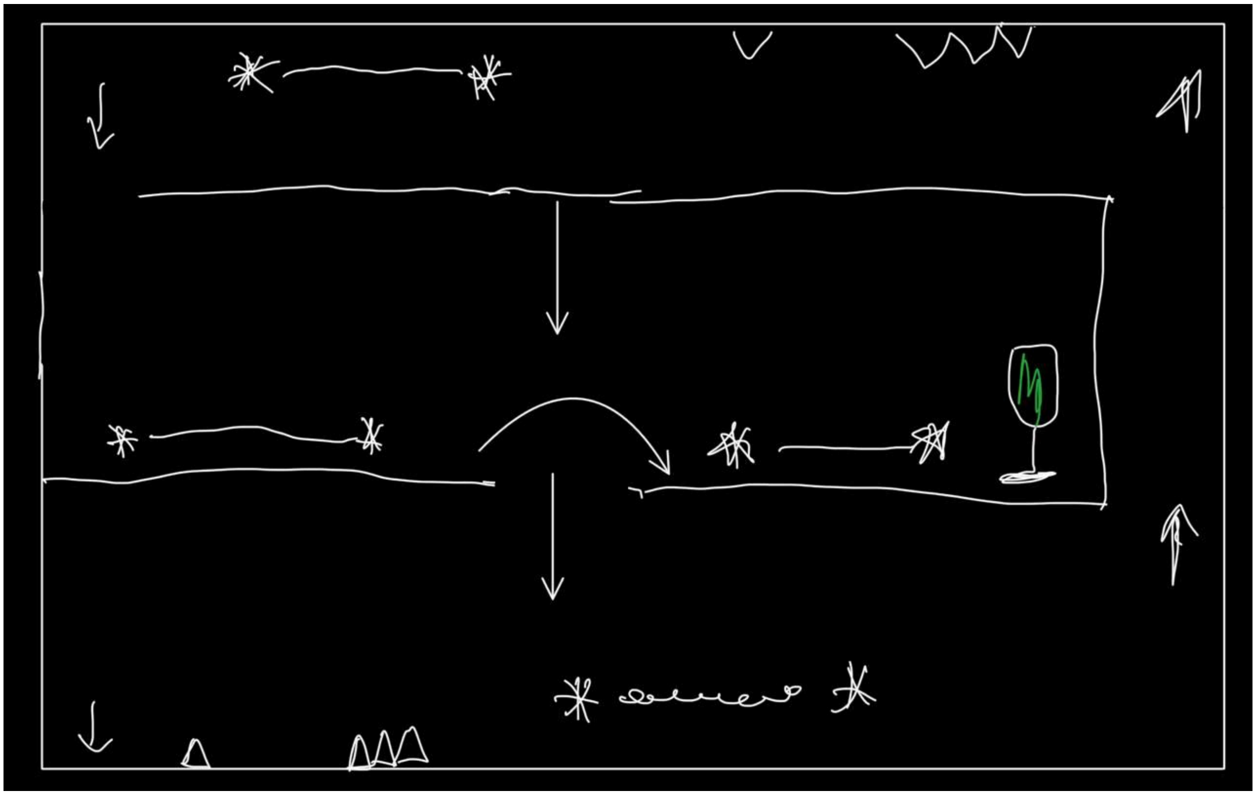
The visual aesthetics of our 2D Bounce Game are designed to evoke a nostalgic retro vibe, combining geometric patterns and pixel art styles to create a visually engaging experience. The backgrounds feature a harmonious blend of vibrant, futuristic designs and classic pixel art elements, enhancing the game's immersive appeal. These artistic choices not only align with the retro theme but also provide a visually distinct environment that captures the attention of players and complements the gameplay mechanics. The carefully selected backgrounds serve to elevate the overall gaming experience by striking a balance between simplicity and creative detail. Here are some samples we used:



Background Samples

Scene Level Maps

The design process for scene-level maps in our 2D Bounce Game involves creating draft layouts for each level, which are then translated into Unity using sprites. These sprites, which form the building blocks of the levels, are carefully selected and positioned to match the drafted blueprints. Each sprite is equipped with collider properties to define the physical boundaries and interactions within the game environment, ensuring seamless integration with the gameplay mechanics. This approach allows us to create visually appealing and functionally robust maps while maintaining precise control over player interactions and level progression.



Level Map Design Draft

Section 4: Player Controls

Movement Controls

The player controls in *Bounce Game* were designed to be intuitive and responsive. The core mechanic involves the player controlling the movement of a bouncing ball that moves across various levels. The player can move the ball left or right and adjust the ball's bounce height using simple keyboard or touch screen inputs, depending on the platform. The control system was designed to be consistent and easy to master while providing enough challenge as the player progresses through different levels.

The ball's physics and bounce behavior were programmed using Unity’s built-in physics engine, which helped create realistic interactions with the environment. The ball’s movement was further enhanced by adjusting the gravity and bounce mechanics, ensuring that the gameplay felt dynamic and responsive.

Boundary Restrictions:

Boundary restrictions were implemented to prevent the player from going off-screen or getting stuck in inaccessible areas. Invisible barriers were placed along the edges of the screen, and when the player's ball collided with these barriers, it would bounce back into the playable area. This was essential for maintaining the flow of gameplay and ensuring players were always engaged with the level's challenges. fully engage with the action without any interruptions. Ultimately, the boundary restrictions contribute to a more polished and enjoyable gaming experience.

Section 5: Game Play

Game Mechanics:

The core gameplay revolves around navigating a bouncing ball through various levels filled with obstacles, enemies, and challenges. The ball's movement is controlled by the player, who must avoid obstacles, jump across platforms, and interact with power-ups to reach the end of each level. Each level presents new obstacles and enemies, ensuring that the gameplay remains fresh and exciting as the player progresses.

The game also incorporates elements of puzzle-solving, where players must figure out how to reach certain platforms or areas by using their ball's bounce and other interactive elements. These mechanics encourage the player to experiment with different strategies and approaches, adding depth to the gameplay experience.

Obstacles:

Obstacles are a critical element in maintaining the game's challenge. Moving platforms, spikes, and walls block the player's progress, requiring quick thinking and precise timing. As players advance through the game, new obstacles are introduced, including enemies with specific movement patterns, environmental hazards, and more complex traps that require careful planning to navigate.

Section 6: Interaction Logic

Player Interactions:

Interaction logic is an essential component of the game's design. When the player's ball collides with specific elements, such as power-ups or enemies, the appropriate interaction is triggered. For example, when the ball touches a power-up, it grants a temporary ability such as reverse gravity or scaling up/down. Similarly, when the ball collides with an enemy or obstacle, the player may lose a life or restart the level, depending on the game's rules.

These interactions were implemented using **Unity's collision detection system**, where each object in the game is assigned a collider that determines how it interacts with other objects. These interactions provide immediate feedback to the player, ensuring that the game feels responsive and rewarding.

Sound Effects:

Sound design was an important part of creating an immersive experience. Various sound effects were implemented for different events in the game, such as the ball bouncing, collecting power-ups, defeating enemies, and transitioning between levels. The sound effects help reinforce the player's actions, providing auditory feedback that enhances the overall gameplay experience.

Background music was composed to fit the themes of each level, creating a cohesive atmosphere that matched the game's pace and tone. The sound design ensured that players felt fully immersed in the game world, with both visual and audio elements working together to create a rich and engaging experience.

Section 7: User Interface

UI Design:

The user interface (UI) was designed to be clean, simple, and user-friendly. It needed to provide players with essential information without detracting from the gameplay experience. The UI includes elements such as the player's score, available lives, power-up indicators, and level progress. These elements were strategically placed to ensure they were easily visible but did not interfere with the gameplay.

Main Menu/Pause Menu/Levels:

The game includes a main menu where players can start a new game, load an existing game, or adjust settings. A pause menu is accessible during gameplay, allowing players to pause the game, restart the level, or return to the main menu. Each level is unlocked as the player progresses, and the level transitions are smooth, with clear indicators for the player's current progress.

Section 8: Scoring and Game Over

Scoring System:

The scoring system is designed to reward players for completing levels, avoiding obstacles, and collecting power-ups. Points are awarded based on various actions, such as defeating enemies, collecting bonuses, and completing the level within a certain time frame. This scoring system encourages players to replay levels and aim for higher scores, adding to the game's replayability.

Game Over Conditions:

A game over occurs when the player loses all their lives or fails to reach the end of the level within the allotted time. The game-over screen shows the player's score and offers options to restart the level or return to the main menu. This system provides a clear sense of progression and encourages players to improve their performance.

Section 9: Polishing and Testing

Bug Testing:

Bug testing was crucial to ensure that the game functioned properly and free of errors. Issues such as collision detection errors, unresponsive controls, and level progression bugs were identified and addressed during the testing phase. Playtesting also helped identify areas where the difficulty needed adjustment or where new features could be added to improve gameplay.

Performance Optimization:

Performance optimization was another critical area, as the game needed to run smoothly across different devices. This was achieved by reducing the size of assets, optimizing physics calculations, and streamlining animations to reduce resource usage. Extensive testing on various platforms helped ensure that the game would run seamlessly on a wide range of devices.

Section 10: Challenges and Solutions

Throughout the development process, several challenges arose, such as balancing difficulty, implementing realistic physics, and optimizing the game for various devices. These challenges were addressed through testing, iteration, and collaboration among the team members. The result is a polished game that successfully balances fun gameplay, challenging mechanics, and smooth performance.

Section 11: Conclusion

The *Bounce Game* project was an invaluable learning experience, providing hands-on practice in game development, programming, and team collaboration. By combining creative design with technical skills, the team was able to create an engaging game that is both nostalgic and innovative. The project not only taught us how to create a functioning game but also how to work as a team, troubleshoot issues, and optimize our code for performance.

Section 12: References

SFX from web: Pixabay, 101soundboards,mixkit.
Gantt Chart Tool: Canva.