Administrator's Manual

22BitArcade

Juan Luis Guerra Gennich

Introduction

I am Juan Luis Guerra Gennich, I'm 22 years old, and I'm presenting my Integrated Project for the Application Development Cycle. It consists of a simulator for an arcade store with various mini-games.

This project is uploaded on my GitHub and my Itch.io page.

GitHub: https://github.com/juanlugg/22BitArcade_Game

Itch.io: https://juanlugg.itch.io/

Contents

1. Main Menu

The Main Menu is the first screen players will see when starting the game. From here, players can access different parts of the game.
Main Menu Options

- Start Game: Begins a new gameplay session.
- Options: Adjust player preferences such as volume and resolution.
- Exit: Closes the game.

2. Loading Screen

The loading screen is the first thing seen when starting the game or transitioning between screens. It helps to load the next scene in the background using a coroutine to prevent issues.

3. Pause Menu

The Pause Menu allows players to pause the game and access various options without exiting the game.
Pause Menu Options:

- Resume: Resumes gameplay.
- Restart: Restarts the current scene.
- Options: Adjust player preferences. Same as the Main Menu.
- Return to Main Menu: Returns to the Main Menu. (Not yet implemented)
- Quit Game: Exits the application.

4. 3D Lobby

You start in a room-like space with several interactive machines, a mirror reflecting the room, and 3D spatial music from four speakers that operate simultaneously. Both this room and the character remain undestroyed constantly until exiting the game or returning to the Menu, in order to save the character's state and even the background music minute.

The machines without any sign and in a color other than blue are fully functional, each with different games, except one that will have a small humoristic easter egg.

You can move freely, dance (by pressing 1 or 2), jump, run, and zoom.

5. Sounds and Images Created by AI

All the songs featured are generated by AI using Suno.

Some images, like the main menu or the icon, are created by AI as copilots. The remaining images are either from assets or sprites or created manually by me. Other sounds are sourced from internet pages. Further details on this will be discussed in the last section.

6. Data Saving with PlayerPrefs

Values that persist even after the game is closed are saved using Unity's PlayerPrefs function. Game settings such as volume and brightness are saved thanks to this function. Additionally, records and other adjustments will be saved using this feature.

7. Integrated Games

Four games have been integrated, manually created by me using references and ideas from the internet for some. Three of these games are clear references to classic retro games (Flappy Bird, Space Invaders, and Arkanoid). Additionally, two projects for class that I created were used to develop two of the games (PinkGuyJump and Arkanoid).

Flappy Bird

A classic game where the objective is to pass through pipes to earn points and achieve your highest score. You can change the bird's color, and this setting will be saved permanently, even when you exit the game.

PinkGuyJump

An original platform game where the objective is to collect the 22 coins scattered across the map in the shortest possible time. This game is derived from a class project.

Space Invaders

A classic game where the objective is to destroy enemy ships without being defeated to achieve your highest score.

Arkanoid

A classic game where the objective is to bounce a ball against bricks without losing lives, with a fixed time limit. You win when you destroy all the bricks. This game is also derived from a class project.

8. Assets, Sprites, Third-Party Code, and more

The assets and sprites used that were sourced from the Unity Store:

https://assetstore.unity.com/packages/3d/props/arcade-machines-polypack-207908
https://assetstore.unity.com/packages/3d/arcade-machine-free-92191
https://assetstore.unity.com/packages/2d/textures-materials/50-free-pbr-materials-242760
https://assetstore.unity.com/packages/3d/props/electronics/hq-acoustic-system-41886
https://assetstore.unity.com/packages/3d/environments/3d-free-modular-kit-85732
https://assetstore.unity.com/packages/2d/environments/free-platform-game-assets-85838

Some scripts, assets, or sprites were sourced from repositories or other pages:

Lobby (Movement script that was used and modified):
https://assetstore.unity.com/packages/3d/characters/modular-first-person-controller-189884

Space Invaders:
https://github.com/Jocyf/Space-Invaders-Clon

Flappy Bird:
https://www.spriters-resource.com/mobile/flappybird/sheet/59894/
https://www.dafont.com/04b-19.font

PinkGuyJump:
Assets from the Unity Store and created by me.

Arkanoid:
Sprites created by me and sprites found on Google or recycled from other games.

9. Controls

Lobby:

- Move: WASD / Arrow Keys
- Camera: Mouse
- Jump: Space
- Run: Shift
- Exit: Esc
- Dance 1: 1
- Dance 2: 2
- Zoom: Right Click
- nteract: E

FlappyBird:

- Jump/Move: Space / Left Click

PinkGuyJump:

- Move: WASD / Arrow Keys
- Jump: Space

SpaceInvaders:

- Move: WASD / Arrow Keys / Drag the Ship
- Shoot: Space / Left Click

Arkanoid:

- Move: WASD / Arrow Keys / Drag the Paddle