|  |  |
| --- | --- |
| A picture of a winding road and trees  PintMan2d  Developer Diary | Abstract  A comical 2D scrolling platformer developed using Unity.  KEVIN BARRY - Student  Mobile Application Development 3 |

PintMAn2D

Contents

[Introduction 2](#_Toc531907750)

[Build Options 2](#_Toc531907751)

[Developer Log 2](#_Toc531907752)

[Bug Log 5](#_Toc531907753)

[The rate of increasing difficulty 6](#_Toc531907754)

[the speed of gameplay 6](#_Toc531907755)

[Gameplay conditions 6](#_Toc531907756)

[Gameplay Systems 6](#_Toc531907757)

[Gain conditions: 6](#_Toc531907758)

[Lose conditions: 6](#_Toc531907759)

[Controls 7](#_Toc531907760)

[Windows 7](#_Toc531907761)

[Touch 7](#_Toc531907762)

[the appropriateness of the control mechanics 8](#_Toc531907763)

[the user of colours and text elements on the user interface 8](#_Toc531907764)

# Introduction

Pintman2d is a comical 2D platformer where the user, Paddy Losty, must navigate through the levels collecting Pints while avoiding bottles of water.

Developer: Kevin Barry

Customer: Thomas Duffy

# Build Options

The game has been developed to deploy on Android, UWP, PC, MAC LINUX and Standalone.

The controls for each differ and require a small change in configuration when changing from a touchscreen to a non-touchscreen device. Each platform has its own advantages and disadvantages which I will discuss further on.



# Developer Log

|  |  |  |
| --- | --- | --- |
| Date | Task | Comment |
| 04- 10-2018 | Initial Set up of project | * Created blank project in Unity. * Set up basic folder structure. |
| 05-10-18 | Setting up player | * Created initial pixel art sprite sheet. * Added movement to player. * Added Jump to player. |
| 13-10-18 | Checkpoints and respawn | * Added CheckpointController.cs script that activates when a player passes checkpoint area. * Now respawns to the last activated checkpoint when player dies. |
|  | Enemy Object | * Created enemy sprite and object as a prefab. |
|  | Collectables | * Created a pint prefab as a collectable object |
| 22-10-18 | Particle System | * Particle system now in place when player dies and respawns. * Particle object destroys after use. |
| 23-10-18 | Enemy behaviour | * Enemies behaviour implemented and animation applied. |
|  | Point System | * Basic point system put in place . * 50 points for killing an enemy. * 100 points for collecting collectable pint object. |
|  | Camera | * Camera is now set to follow the player. |
|  | Game boundaries | * A bounder collider set up under game map so when a player falls of the level a respawn is activated to latest checkpoint. |
|  |  |  |
| 09/11/18 | Firing weapon | * Added shooting action to player. * Control = V (will be changed). * Fires a star object. * +50 points kill enemy. * Particle effect added when contact is made with another object. * Star destroyed on contact to save memory space. * Star ignores checkpoints. |
|  | Player Animation | * Added animations to play state. * Walking * Idle * Jumping * Flip (flips player horizontally based on movement direction) |
|  | Level Management | * Set a door object to act as end of level. (Activates a menu) * Level menu implemented. * User control enters door (exact buttons yet to be confirmed) * Started design on level 2 |
|  | Pints (Collectables) | * Rotation animation applied using pixel art sprite sheet. |
|  | Menu | * Created Main menu as a level. * Has 3 buttons to select: ‘New Game’, ‘Level Select’, ‘Quit Game’ |
|  |  |  |
| 10/11/18 | Enemy | * Updated Sprite with new pixel art. * Added Animation to enemy state and script for enemy patrol. |
|  | Firing weapon (Star) | * Refined star object. * Destroys star object on contact or after 3 seconds to increase performance. |
|  |  |  |
| 12/11/18 |  |  |
|  | Health System Enemy | * Initial health system put in place * Takes 2 firing shots the kill enemy (will increase as game progresses) |
|  | Health System Player | * Added Player health system. * Player loses health when collides with enemy. |
|  | Player | * Player is knocked back on contact with enemy. |
| 02/12/18 |  |  |
|  | Life System | * Started player life system. * Carrying lives through level. * Reset lives when new game started |
|  | Score | * Carry score through levels. * Reset score when new game started |
|  | UI | * Set up Heads up Display UI to display player health, number of lives, score and countdown timer. |
|  | Time system | * Set a countdown, counts down from one minute per level. * Timer is stopped when game paused. * When time < 0 a life is decremented. |
|  |  |  |
|  |  |  |
|  |  |  |
|  | Sound System | * Sound system implemented. * Noise on Player contact with enemy. * When player shoots. * When shooting star hits any other object. * When pickup (pint or extra health) activated. |
| 3-12-18 | Health | * Health pickup (+3 health) |
|  | Enemy | * Destroy object on impact * Player bouncing off enemies corrected. |
|  | Touch System | * Touch buttons UI created. * Mapped touch to the methods. * Scripted to perform different code if touch or pc input. |
|  | Firing weapon | * Added rotation animation to shooting object. * Set delay time on shootinrg. |
|  | Android | * Build tested on android device (need tunings) |
|  | Level Select map | * Added level select sprites. * Added movement to player to navigate to selected level. |
|  | Level Locking | * Levels are now locked until the previous level is completed. |
| 6/12/18 | Enemy objects | * Changed Water objects to polygon colliders . |
|  | Menu | * Anchored menu to support different screen resolutions. |
|  |  |  |
|  |  |  |
|  |  |  |

# 

# Bug Log

Below is a brief log of some of the bugs encountered during the development process which I fixed at a later date. As this was a new framework to me most bugs where fixed as soon as they occurred.

|  |  |  |
| --- | --- | --- |
| Date | Bug | Solution |
| 22/10/18 | Player Stick to walls | Increased gravity scale |
| 23/10/18 | Player falling of screen | Added a collider under game map, respawns player to last activated checkpoint. |
| 26/11/18 | Player scaling | Player size was scaling due to incorrectly configured sprites. Redesigned sprite sheet and learned better import methods. |
| 3-12-18 | Player firing multiple stars | Set counter to next shot (0.2 sec delay) |
|  |  |  |
|  | Android controls   * All Input Methods acting uncontrollable | Fixed moved kickback code. |
|  |  |  |
| 4-12-18 | UI sizes out of scale | Added scale to screen option on canvas and anchored elements. |
|  | Can’t exit door on touch | Added function to call level loader. Now uses jump touch button |
|  | Level Selector Not saving unlocked levels | Used local storage to save which levels are unlocked. |
|  |  |  |
|  | Level selector: On pause player not in right position | Use player pref to save position. |
| 6-12-18 | Pause menu not display correct | Anchored canvas and set to scale |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# The rate of increasing difficulty

The game has been designed to challenge player competence and confidence as they progress through the game. Each level is more difficult to complete than the previous level. In the 1st level it takes 2 shots to kill an enemy, 3shots in the 2nd level and 5 shots in the 3rd level. As the user progresses each levels difficulty is also challenged by increasing the number of enemy and enemy objects. Navigation the levels becomes more difficult as the levels increase.

# the speed of gameplay

# Gameplay conditions

The Aim of the game is to progress through each level by avoiding and killing enemies whilst collect as much pints as possible.

## Gameplay Systems

Time Counter: (Initially one minute)

Player Health Counter: (UI scroller, represents from 1-5) Game starts with 5 health.

Score Counter: initially set to 0

Lives Counter: initially set to 3

Enemy Health

## Gain conditions:

Collect Pint object increases points by 100

Killing an enemy increases point by 50.

Shooting an enemy or jumping on enemy head deducts 1 health from enemy.

Finishing a level unlocks the next level.

Collecting a heart increases life by 1.

Collecting a small paddy item increase health by 3 points.

## Lose conditions:

A collision with a water object decreases player health by 3/4/5 depending on level and decreases score by 50 .

The timer reaching 0 decreases one life from player.

When health is less than 0 player loses a life.

When player loses last life, the game is ended.

When player collides with enemy 1 health point is lost

# Controls

## Windows

All UI Menu and pause option buttons are controlled by mouse click.

|  |  |  |
| --- | --- | --- |
| **Function** | **Keyboard** | **Xbox Game Pad** |
| Move Left or right | Left / right arrow | Joystick axis X |
| Shoot | V | joystick button 2 |
| Jump | Space bar | joystick button 0 |
| Enter pause menu | Escape | start |
| Enter door (end of level) | U arrow | Joystick axis Y |
| Enter Level | Space bar | joystick button 2 |

## Touch

All UI Menu and pause option buttons are controlled by touch.

|  |  |
| --- | --- |
| **Button** | **Action** |
|  | Move Player Left |
|  | Move Player Right |
|  | Jump  Enter end of level door  Enter level select door |
|  | Shoot |
|  | Pause / Un pause game |

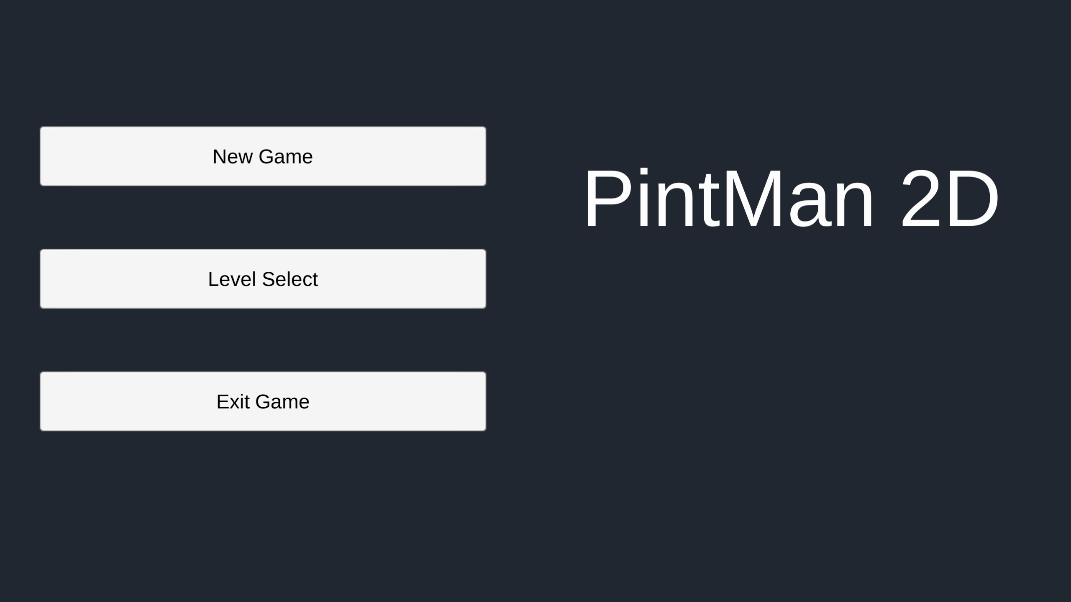
# Storing Data

In order to increase the game functionality some data is stored locally. This was achieved by using the class UnityEngine.PlayerPrefs. Using PlayerPrefs enabled data to be carried through levels while also allowing the game to be exited and re-opened in the same state.

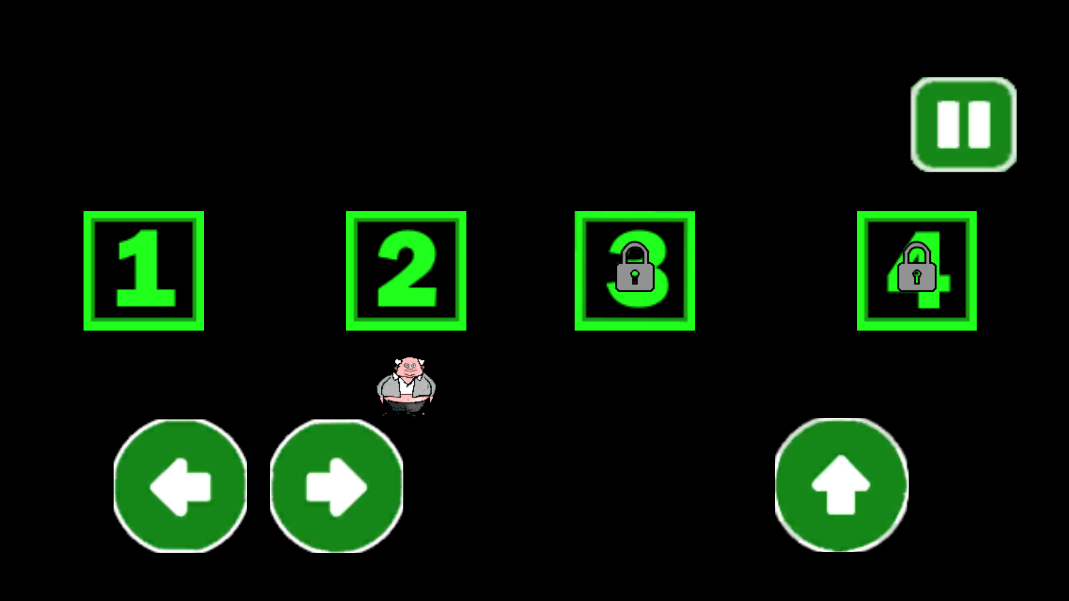
* CurrentPlayerHealth: Saves current play health so game can be reloaded with same health and health level can be carried through levels.
* unlockedLevel: This saves the level unlocked from finishing a level. unlockedLevel is the name of the lock which can be either level1locked, level2locked or level3locked and is set to a value of 1 if the level is unlocked.
* PlayerLevelSelectPosition: Is used to store the x value of the players position in the level select menu. This meaning when a player enters the level select menu the character will be located under the current level.
* CurrentPlayerLives: Stores the current value of player lives.
* CurrentPlayerScore: Stores the players current score and is reset to 0 on new game.

# Screenshots of finished Android build

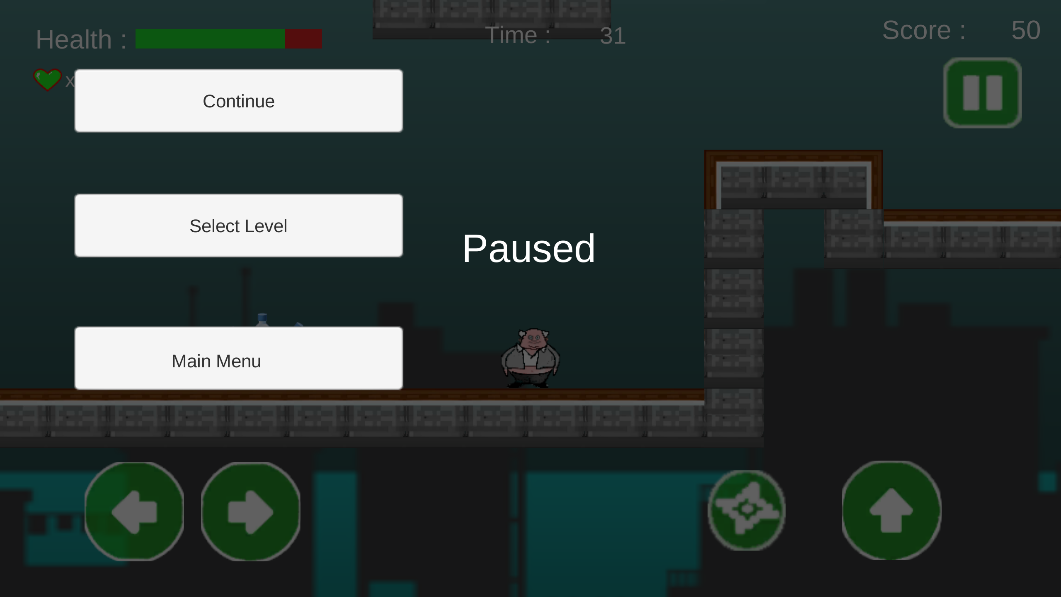
## Main Menu



## Level Select Menu



## Pause Menu



## In Game



# the appropriateness of the control mechanics

# the user of colours and text elements on the user interface