DESIGN DOCUMENTATION

Michael Dodd

CONTENTS

GAME BASIC INFORMATION

NAME – Pipe Hextreme

STRAPLINE – Taking the Hex to the treme

DESCRIPTION – A ball that travels down a hexagon shaped pipe overcoming obstacles and completing levels earning points to compete on the leaderboard. With each level increasing in difficulty and challenges. A perfect game to pass free time with this casual action game..

UNIQUE SELLING POINT – Dedicated online game, bringing back the retro gaming feeling to modern day

TERRITORIES – Globally, the game is planned to be released as a web game therefor it can be played wherever they have internet access, the main targets of the game will be English speaking countries UK and America is where the game advertisement will be mainly focused to get the largest audience.

TEXT DESCRIPTION OF THE GAME

OVERVIEW

The game is set in space in a world where everything is hexagon shaped. Two space stations have collided and now it is perfect for riding.

OBJECTIVE

There are a couple of objects that the player when playing the game, the primary objective of the game is to complete the levels that slowly increase in difficulty with new obstacles and different layouts of levels, also there is a high score for each level and the overall game.

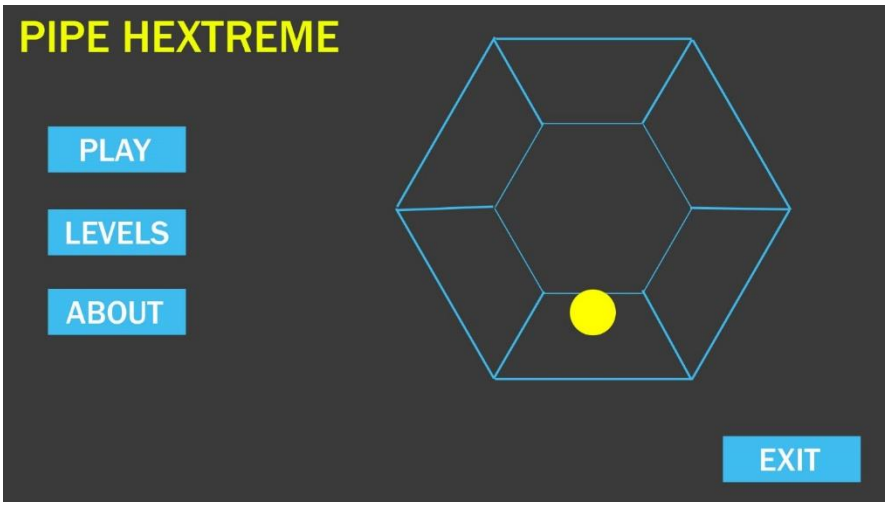
THEME

The theme of the game has a retro tone to it. With higher resolution graphics, the main pipe is made out of blue colored plane.

PLAYER PERSPECTIVE

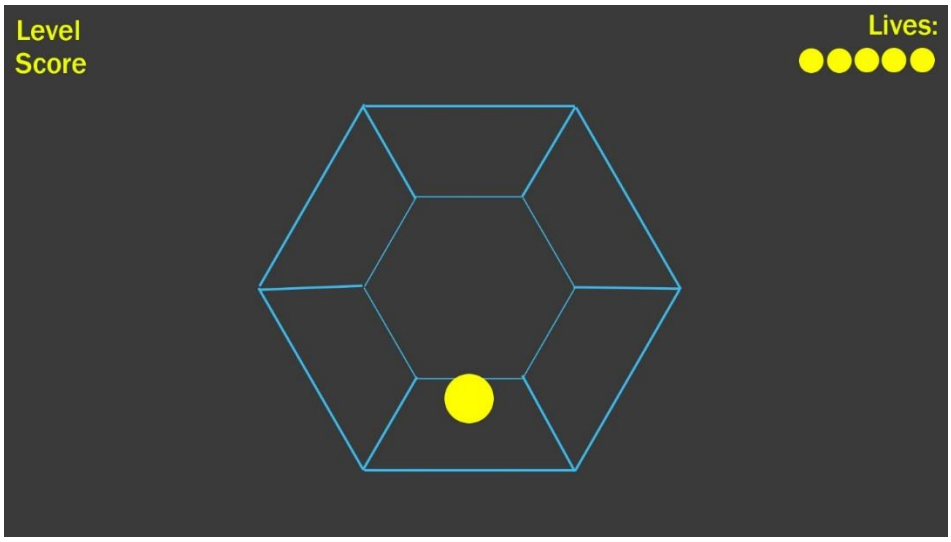
The camera will stay stationary in the center of the pipe following just behind the ball. This perspective allows you to see the ball when it goes through the entire pipe as well as oncoming objects, obstacles and points.

GAME LAYOUT

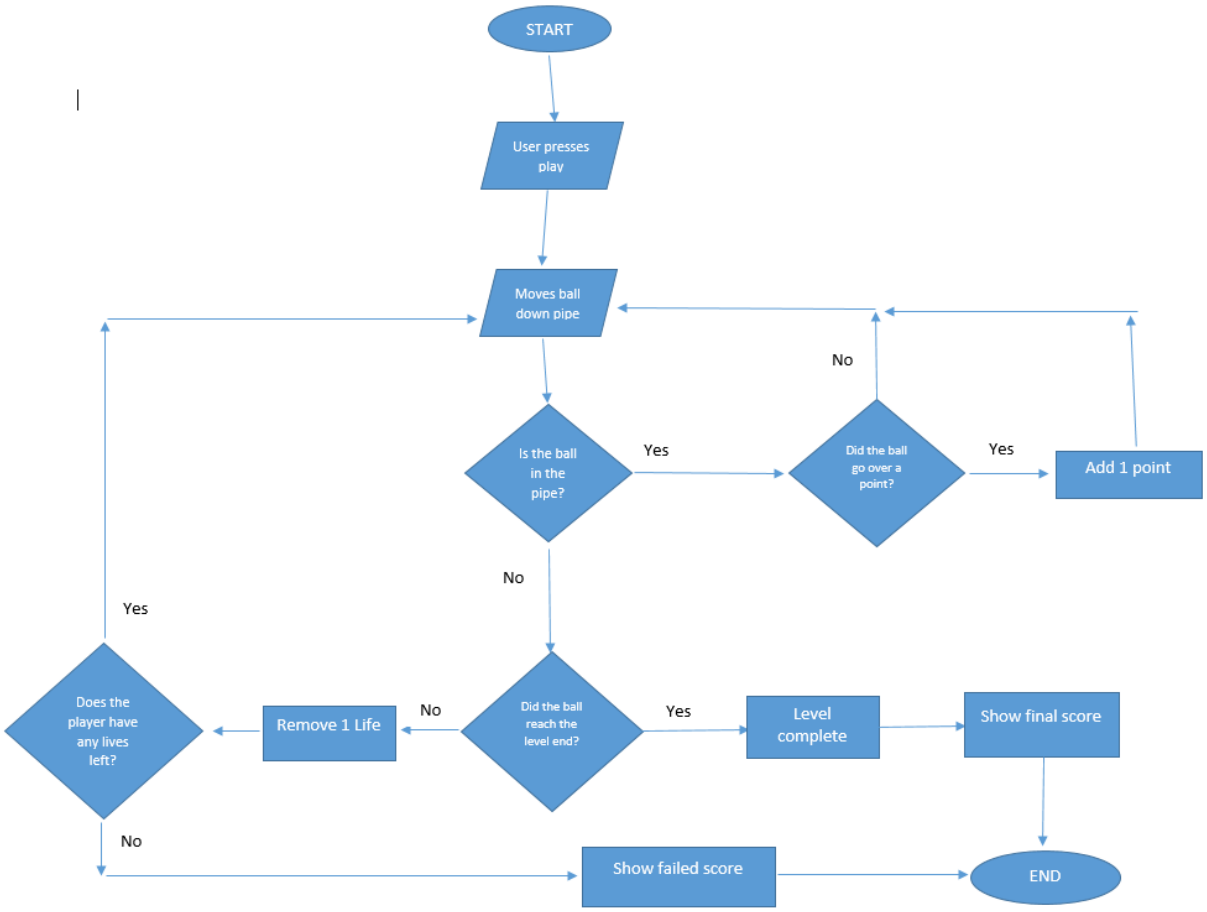


This is the main menu it has buttons that takes the user input on what they want to go, when you hover over the button it shows that it has been hovered over by changing colour, the click on the buttons will give a sound when clicked on.

When playing the game this is what the player will see. It has the main pipe in the middle where the ball will travel and have obstacles. The lives are displayed in the top right, each circle represents a ball that is a life, similar to how it is displayed in pinball. Overall the screen designs are very simple and will help the user know exactly what is happening at any time, giving plenty of visual feedback and there will be lots of audio feedback included for when clicking a button and also when collecting points.



GAME FLOWCHART



GAME INPUTS, OUTPUTS AND PROCESSES

Inputs	Outputs	Processes
Left/ right/ up/ down arrows or W/ A/ S/ D keys for movement	Sound and music while playing	Calculate score
Spacebar for jump	Display lives	Update and change lives
Click on menu buttons	Display score	Check ball is in pipe and remove lives if it is out of pipe
	Display highscore	Check if ball is at end of level
		Collide with point tile
		Collide with speed time and increase speed

VARIABLES

Variable	Data Type	Description	Justification
Score	int	Displays the current score	I have chosen to use an int since it will be added to, with whole numbers only and displayed for the user. Also I used an int not a string since I will need to add to it as the game goes on
HighScore	int	Displays the players highest score achieved	I have chosen an int since it will only handle whole numbers and will need to display a score.
BallSpeed	decimal	How much forward momentum the ball has	This is a decimal since the ball speed will need to be accurately changed with the amount of forward force given to it.
PlayerName	string	This is the name of the current player	I have used a sting since no calculations will be needed and this

			will include letters so that I can display the players name
BallLives	int	The players lives that start at 5 then gets lower when a life is lost	The number will refer to the amount of lives and they will need to be subtracted and added so an int is needed.

DESCRIBING AND JUSTIFYING STRUCTURES

When making the game I will use structures such as arrays, If else statements, collisions and loops. To make sure the game runs correctly and smoothly for the player.

I will use an array in my game called “Highscore” to store the top 10 high scores and player names. At the end of the game the game will check the score variable to the scores in the array starting at the lowest and working up until it is lower and if it is a high score then it will be put in the array and the lowest will be removed. The array will then be used to display on the leaderboard since I can save the player name with the score and then use it to display the top 10 scores.

I will have many loops in my game to check different things, I will have a loop that checks using an IF statement to check the y coordinates of the ball to check if it has fallen out of the pipe and if it has then to remove a life. From this I will also have another IF, else statement that checks if the lives are at 0 to end the game or whether to restart with 1 less life.

In the game I will also be detecting collisions for collecting points that are in the pipe. When the ball touches a point tile it will add value to the score variable. These are just the main structures in my game there will be other loops and IF else statements in my game that make it work and function correctly.