**HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY AND EDUCATION**

**FACULTY FOR HIGH QUALITY TRAINING**

**Logo, company name

Auto-create description**

**REPORT**

**PROJECT IT**

**Lecturer: Le Van Vinh, Ph.D**

**Students: Nguyễn Xuân Lộc - 20110002**

**Hoàng Lê Tiến Đạt - 20110011**

**Class : 221PROJ215879E\_03CLC**

Ho Chi Minh City, 1st October 2022

CHAPTER 1 – INTRODUCTION

This report describes the process of making a 2D Platform game: Fire Dragon Warrior. This game was made using Unity and is only available on PC. We will introduce the basic information about the game in this first chapter.

* 1. Game Overview

The main objective of this game is to reach the end; and to reach it, players will have to control the main character to move forward and get over traps, and enemies that are waiting to reduce the player’s health bar. If players lose all of HP points, they will have to start again from the nearest checkpoint

1.1.1 Plot

The main character of the game is a dragon warrior, Ifrit, who was left behind in a war between humans and dragons. Being alone, what can he do to survive and escape the human territory?

1.1.2 Target Audience

2D Platform games have appeared for a very long time and they are still famous now. One of the most remarkable games in this category is Mario. So, everyone, of every age can easily play and finish this game.

1.1.3 Game Description

Fire Dragon Warrior will bring a world of fantasies to players through the main character: Ifrit. He has only two jobs: to run and to defeat all humans that he sees. So, this single-player game will have the dragon, some humans as enemies, and traps. These enemies and traps have the purpose are to decreasing Ifrit’s HP bar until it reaches 0, players have to reach the checkpoint so that they can be respawned if their character dies. There are totally … stages that players need to pass before reaching this game’s end. Overall, it only requires from 5 to 10 minutes to finish this game.

* 1. Game Objectives:

To make a simple Platform game that has the most basic functions so that everyone can play it.

To implement a fireball attack instead of a commonly melee attack.

To add animation to the player, enemies, and traps.

* 1. Reason to choose this project

We need something that is not so hard to do, but also enough to prove that we are IT students, so a simple Platform game is more than enough. Because it requires us must have certain programming skills to do scripts, and also helps us to get use to developing games.

* 1. Technical Feasibility

In order to make this game, we are using:

Game Engine: Unity 2021.3.9f1

Code Editor: Visual Studio 2019

Programming Language: C#

Because this project is not too complicated, so we only use two programs and all of them are free.

CHAPTER 2 – SYSTEM DESIGN

2.1 List of Classes and their Responsibilities

|  |  |  |
| --- | --- | --- |
| No | Class Name | Role |
| 1 | CameraController.cs | Make the camera follow player. |
| 2 | Door.cs | Make the camera change from this room to another room. |
| 3 | Enemy\_Sideway.cs | Sets up Saw Trap movement, hitbox. |
| 4 | Health.cs | Relates to main character health, taking damage, and also a cause to his death. |
| 5 | HeathCollectible.cs | Modifies number of HP point generates when players pick up Hearth. |
| 6 | PlayerAttack.cs | Trigger attack, cooldown time. |
| 7 | PlayerMovement.cs | Takes care most of the player animations: moving, jumping, climbing. |
| 8 | Projectile.cs | Adjust the fire ball’s speed, duration. |

2.2 Class Diagram

These class diagram show classes that are used in game design.

|  |
| --- |
| *PlayerMovement* |
| *+speed: float*  *+jumpPower: float*  *+wallJumpCooldown: float*  *+horizontalInput: float* |
| *#Awake()*  *#Update()*  *-jump()*  *-isGrounded()*  *-onWall()*  *-canAttack* |

|  |
| --- |
| *PlayerAttack* |
| *+attackCooldown: float* |
| *#Awake()*  *#Update()*  *-Attack()*  *-FindFireball()* |

|  |
| --- |
| *CameraController* |
| *+speed: float*  *+currentPosX: float*  *+aheadDistance: float*  *+cameraSpeed: float* |
| *#Update()*  *-MoveToNewRoom(Transform \_newRoom)* |

|  |
| --- |
| *Projectile* |
| *+speed: float*  *+direction: float*  *+hit: bool*  *+lifetime: float* |
| *#Awake()*  *#Update()*  *-OnTriggerEnter2D(Collider2D collision)*  *-SetDirection(float \_direction)*  *-Deactivate()* |

|  |
| --- |
| *Door* |
|  |
| *-OnTriggerEnter2D(Collider2D collision)* |

CHAPTER 3: CHARACTERS DESIGN

3.1 Main character

We use the dragon warrior asset which is free on unity store



Image of Health bar/ HP point is taken from a content creator, Pandemonium.

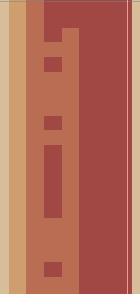
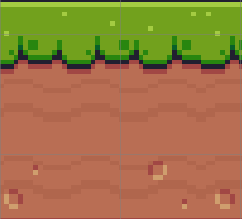


3.2 Enemies

CHAPTER 4: LEVEL DESIGN

4.1 Room design

We also use a free asset on unity store, Pixel Adventures to make level design.

Wall Background Ground



Start Point

CHAPTER 5: GAME DESIGN

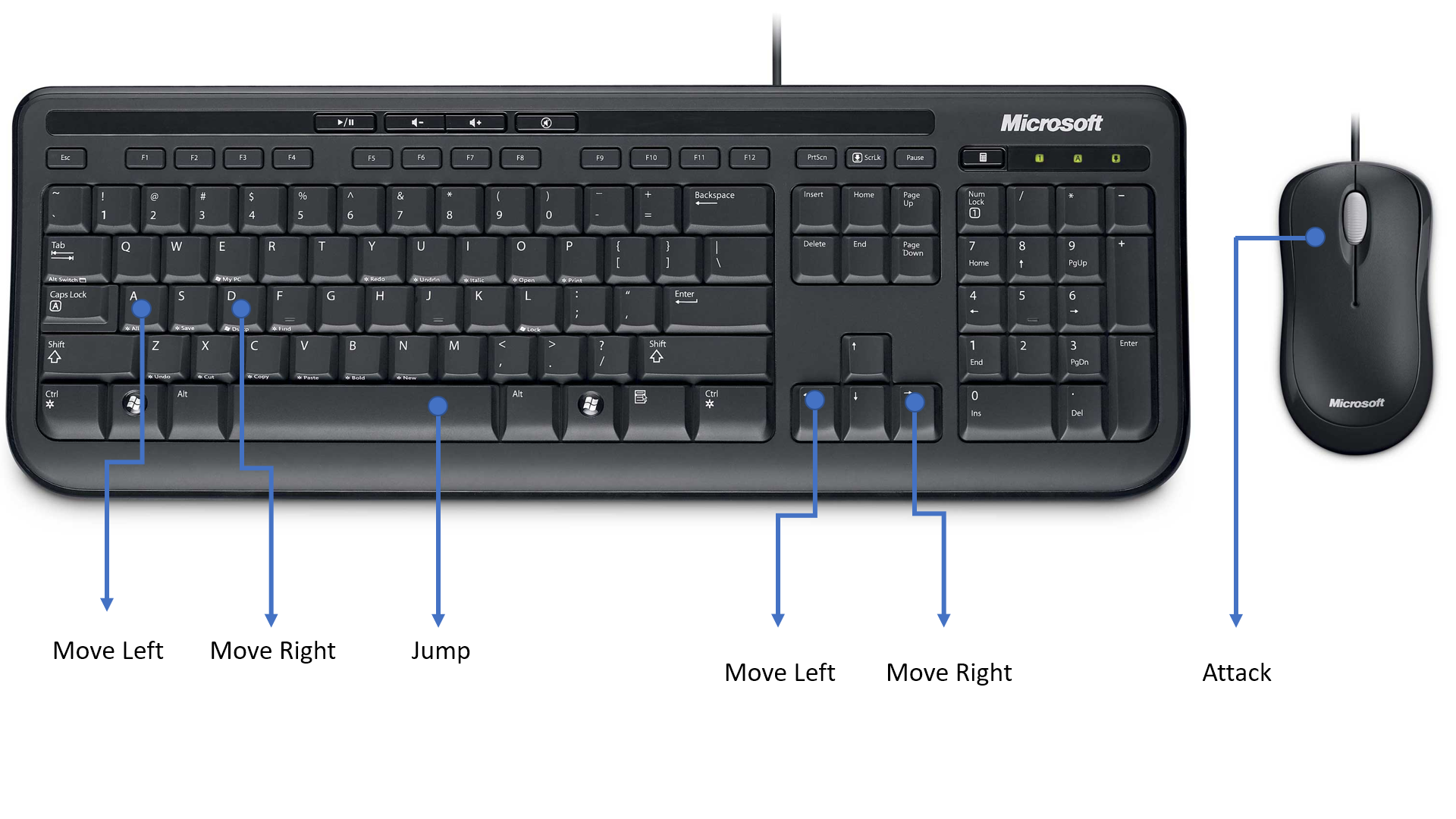
5.1 Game rules

The game starts with player and environment spawning scene, player can move freely from left to right, jump and also climb wall. Main character has a HP bar that can be decreased by multiple enemies that spawn in the following rooms. Players can use attack to knock down these enemies and continue their journey but also players can choice to attack those mob or pass them. Players can also decrease their HP bar by activating traps. When HP bar decreases to 0, player will be spawn back at checkpoint.

5.2 Game structure

5.3 Game play

Because our game are designed for only PC, so we use keyboard as our only game controller. Players can use either A/D or Left/Right Arrow key to move left and right, space to jump and left mouse to attack.



5.4 Camera

At menu there are two types of Camera movement can be set, that is, the camera will move along the main character or the camera will move after players enter a new room.

5.5 HUD

5.6 Animation States

5.6.1 Player

Main character has two main movement that is move left and right, which is depends on how player press button, or horizontalInput. Moreover, there are jumping- Jump(), climbing – onWall() and attacking – Attack() animation.



Idle Attack Climb Move

Hurt Die

5.6.2 Enemies

5.6.5 UI

5.7 Audio