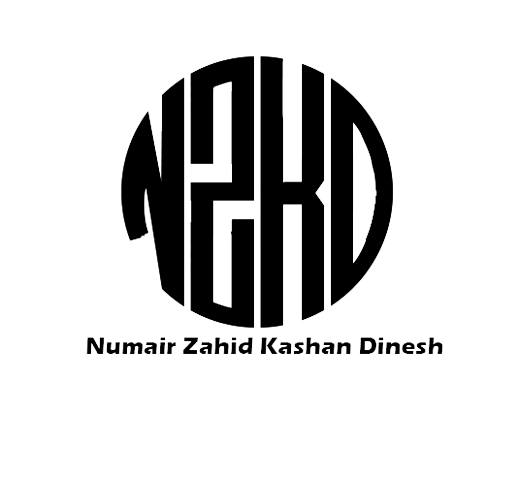
Computer Graphics

& Multimedia

Lab Project

|  |  |  |
| --- | --- | --- |
| **MEMBERS NAME’S** | **ROLL NO’S** | **REMARKS** |
| Kashan Khan Ghor | BSCS-2020-070 |  |
| Dinesh Kumar | BSCS-2020-095 |  |
| Muhammad Zahid Hashim | BSCS-2020-075 |  |
| Numair Nasir | BSCS-2020-72 |  |

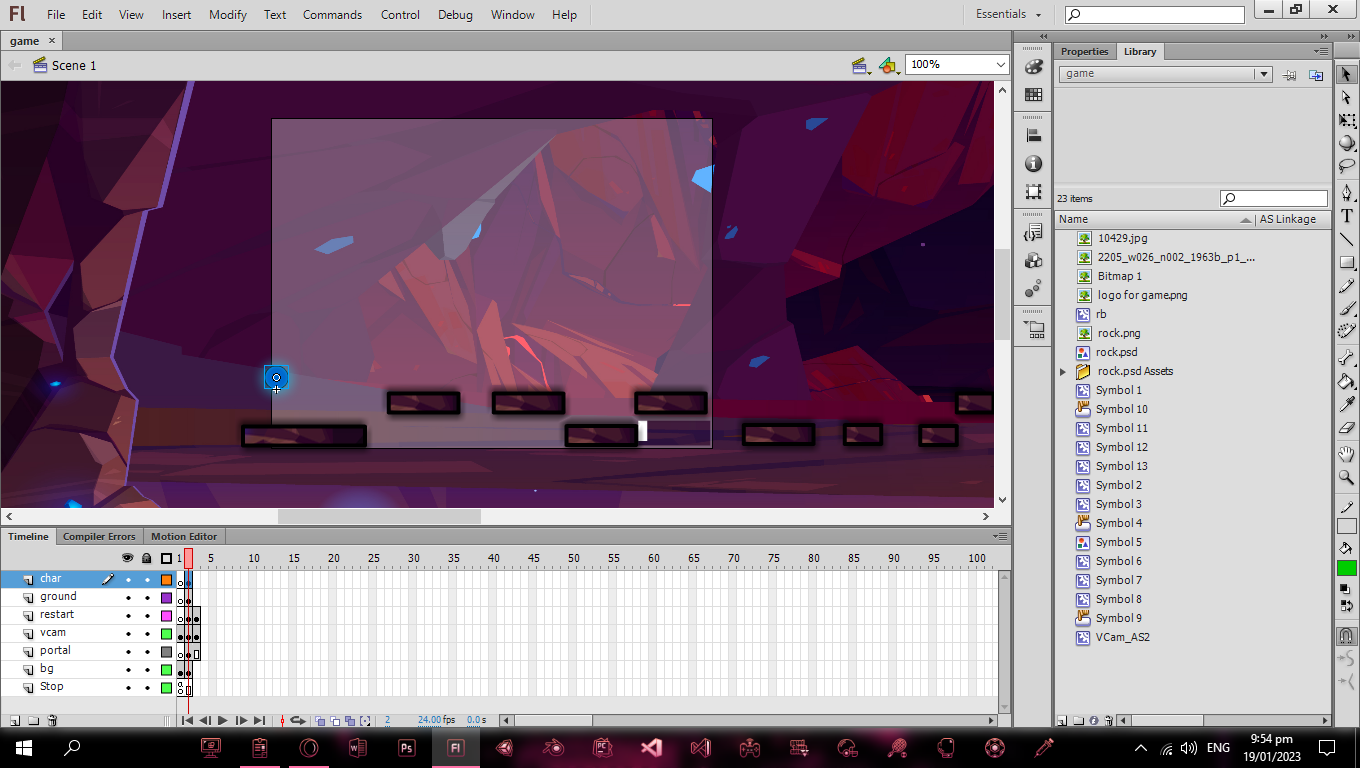


Project Report

Project Title: Bouncy Trip

Description: We have designed a game on Adobe Flash named “bouncy trip.” In this game, the player has to make the ball reach the end portal to win the game. The user will have 3 controls to make the ball reach the end portal right, left and jump. The ball needs to go through a path of small, floating bricks. So, if the ball falls out of the brick, the game will restart, and the ball will be placed at its initial point.

Layers: There are total 7 layers that we created in adobe flash to make this game.

1. Char (Ball)
2. Ground
3. Restart
4. Vcam
5. Portal
6. bg (Background)
7. Stop

Codings On Layers:

1. Char (Ball)

onClipEvent (load) {

var ground:MovieClip = \_root.ground;

var grav:Number = 0;

var gravity:Number = 2;

var speed:Number = 7;

var maxJump:Number = -12;

var touchingGround:Boolean = false;

}

onClipEvent (enterFrame) {

\_y += grav; 'making characters y position = grav'

grav += gravity;

while (ground.hitTest(\_x, \_y, true)) {

\_y -= gravity;

grav = 0;

/\*while char is hitting ground, grav = 0, so char is

always grounded\*/

}

if (ground.hitTest(\_x, \_y+5, true)) {

touchingGround = true;

} else {

touchingGround = false;

}

if (Key.isDown(Key.RIGHT) || Key.isDown(68)) {

\_x += speed;

/\*'when right key is pressed move character to right'\*/

}

if (Key.isDown(Key.LEFT) || Key.isDown(65)) {

\_x -= speed;

/\*'when left key is pressed move character to left'\*/

}

if ((Key.isDown(Key.UP) || Key.isDown(Key.SPACE) || Key.isDown(87)) && touchingGround) {

grav = maxJump;

/\*when up key is pressed, grav = jump, so

the char can jump\*/

}

if (ground.hitTest(\_x+(\_width/2), \_y-(\_height/2), true)) {

\_x -= speed;

/\*'for left and jump at the same time'\*/

}

if (ground.hitTest(\_x-(\_width/2), \_y-(\_height/2), true)) {

\_x += speed;

/\*'for right and jump at the same time'\*/

}

if (ground.hitTest(\_x, \_y-(height), true)) {

grav = 3;

}

}

1. Restart

onClipEvent (enterFrame) {

if (\_root.char.hitTest(this)) {

\_root.char.\_x = charX = 54.45

\_root.char.\_y = charY = 294.7

}

}

/\*if char hits the restart, it will return to starting position \*/

1. Vcam

onClipEvent (enterFrame) {

\_y += (\_root.char.\_y-100-\_y)/4;

\_x += (\_root.char.\_x-\_x)/4;

}

/\*camera to follow characters movement\*/

1. Portal

onClipEvent (enterFrame) {

if (this.hitTest(\_root.char))

{

\_root.gotoAndStop(3);

}

}

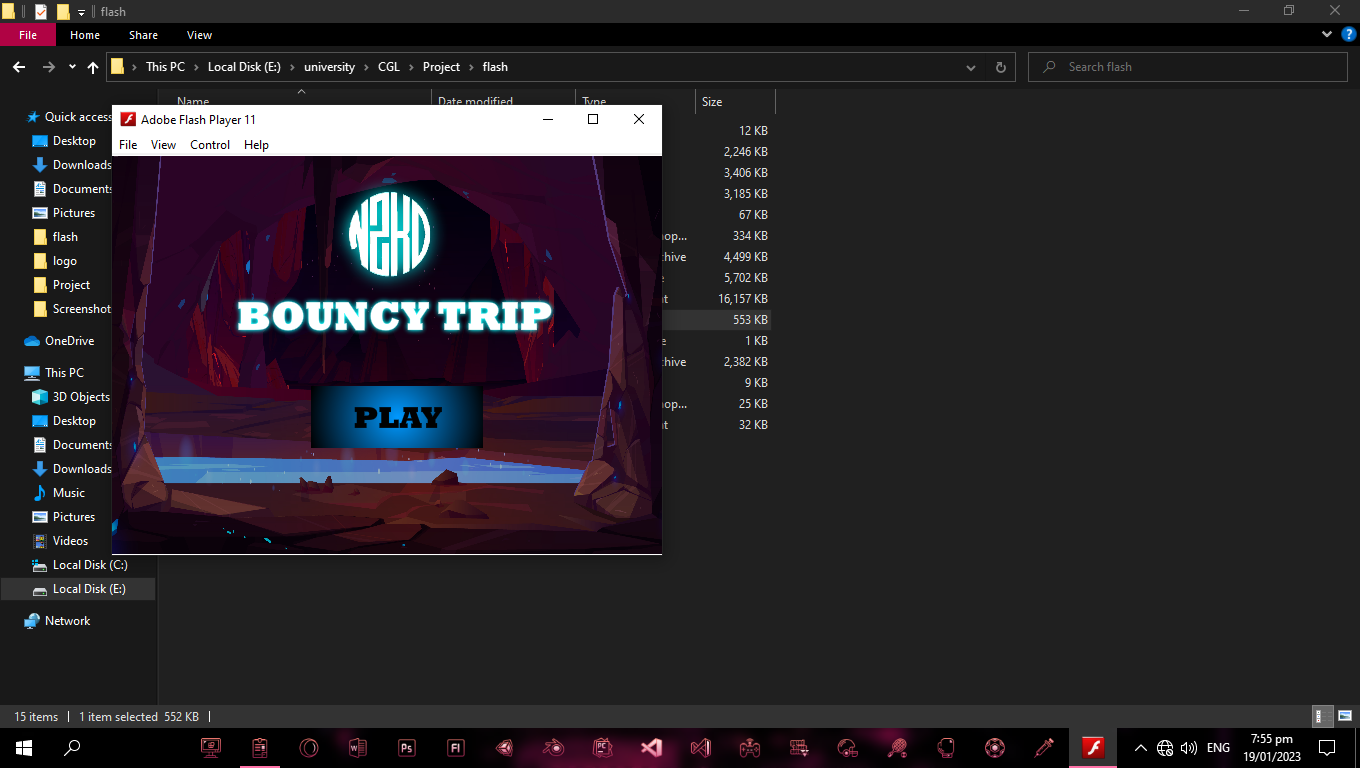
/\*if char hits portal, frame is changed to 3\*/

1. Stop

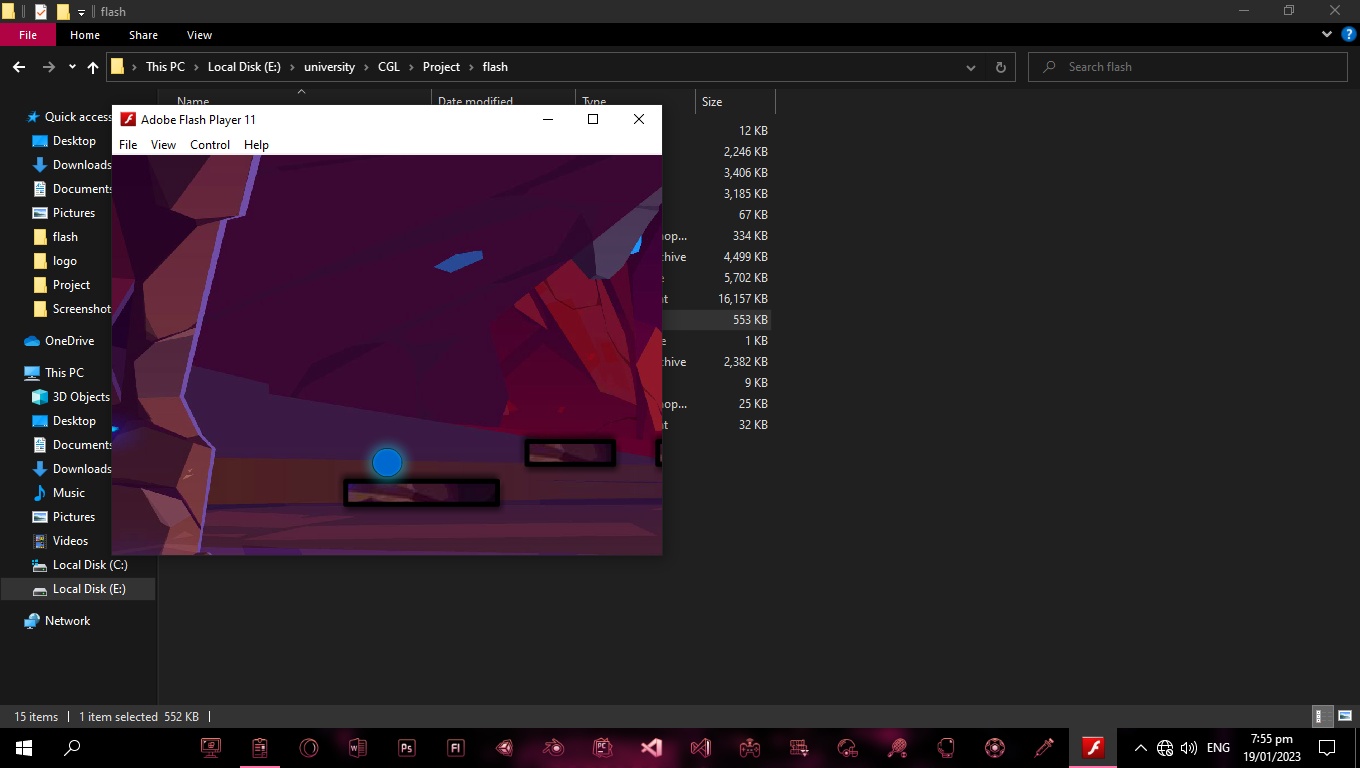
stop();

Game images:

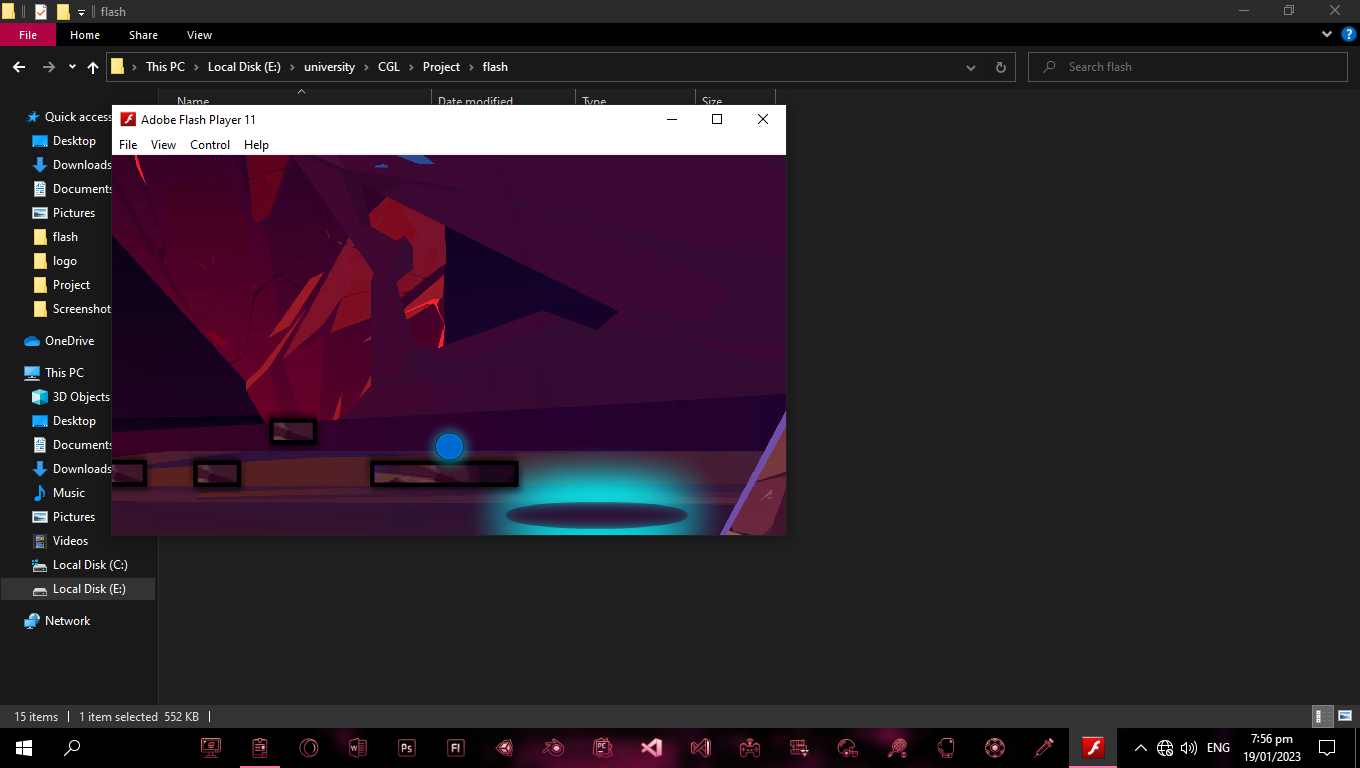
Start Screen:



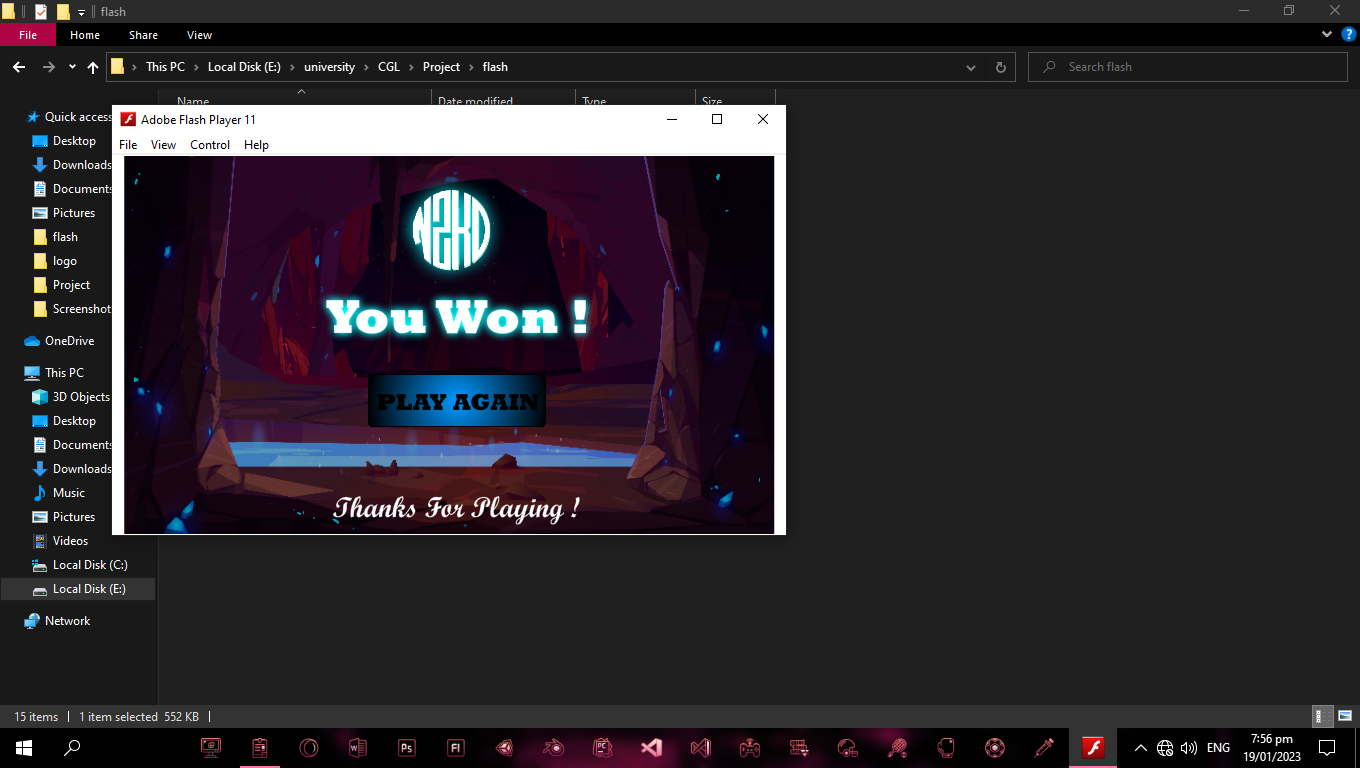
Game Screen:

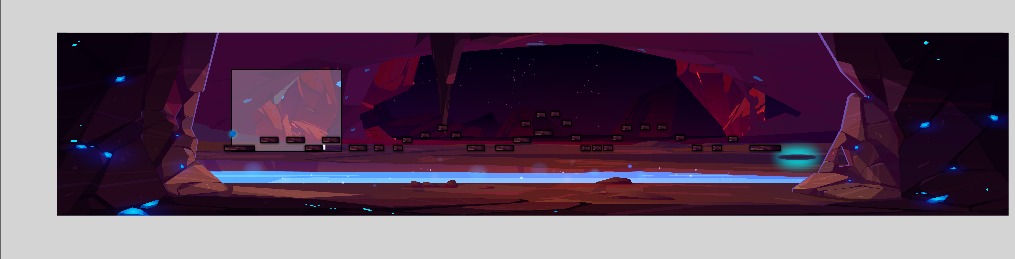


End Portal:



End Screen:



Complete Game View: