**ИГРА «ЛАБИРИНТ»**

**РУКОВОДСТВО ПРОГРАММИСТА**

Киров

2022

# **1.ОГЛАВЛЕНИЕ**

[**1.ОГЛАВЛЕНИЕ** 2](#_Toc119770593)

[**2.ВВЕДЕНИЕ** 2](#_Toc119770594)

[**3. ОПИСАНИЕ ОБЪЕКТОВ ИГРЫ** 3](#_Toc119770595)

[**4. ЛИСТИНГ ПРОГРАММЫ** 9](#_Toc119770596)

# **2.ВВЕДЕНИЕ**

Руководство предназначено для программистов, работающих с изделием “Игра Лабиринт” В руководстве содержатся сведения, необходимые для интеграции игры.

# **3. ОПИСАНИЕ ОБЪЕКТОВ ИГРЫ**

Игровой уровень включает в себя следующие объекты:

1. Объект Person состоит из следующих компонентов:

* root
* footik.L
* footroll.ik.L
* footroll.ik.L\_end
* MCH-foot.L.rocker.01
* MCH-foot.L.rocker.02
* MCH-foot.L.roll.01
* MCH-foot.L.roll.02
* MCH-foot.L.001
* MCH-foot.L.001\_end
* MCH-toe.L.socket
* MCH-toe.L.parent
* toe.L
* toe.L\_end
* foot.ik.R
* foot roll.ik.R
* footsoll.ik.R\_end
* MCH-foot.R.rocker.01
* MCH-foot R.rocker.02
* MCH-foot.R.roll.01
* MCH-foot.R.roll.02
* MCH-foot.R.001
* MCH-foot.R.001\_end
* MCH-toe.R.socket!
* MCH-toe.R.parent
* toe.R
* toe.R\_end
* hand.ik.L
* hand.ik.R
* MCH-elbow\_target.ik\_parent.L
* elbow\_target.ik.L
* MCH-elbow\_target.ik\_parent.R
* shin\_hose\_end.L\_end
* MCH-shin\_hose\_p.L
* shin\_hose.L
* shin\_hose.L\_end
* MCH-thigh\_hose\_end\_p.L
* thigh-hose\_end.L
* thigh\_hose\_end.L\_end
* MCH-thigh\_hose\_p.L
* thigh\_hose.L
* thigh\_hose.L\_end
* MCH-hips\_sh\_ns\_ch.001
* MCH-knee\_hose\_p.R
* knee\_hose.R
* knee\_hose.R\_end
* MCH-shin\_hose\_end\_p.R
* shin\_hose\_end.R
* shin\_hose\_end.R\_end
* MCH-shin\_hose\_p.R
* shin\_hose.R
* shin\_hose.Rend
* MCH-thigh\_hose\_end\_p.R
* thigh\_hose\_end.R
* thigh\_hose\_end.R\_end
* MCH-thigh\_hose\_p.R
* thigh\_hose.R
* thigh\_hose.R\_end
* MCH-knee\_target.ik\_parent.L
* knee \_target.ik.L
* knee\_target.ik.L\_end
* MCH-knee\_target.ik\_parent.R
* knee \_target.ik.R
* knee\_target.ik.R\_end
* MCH-shoulder\_fk\_ns\_ch.L
* MCH-upper\_arm.fk.L.socket!
* MCH-upper\_arm.fk.L.socket1\_end
* MCH- shoulder ik ns\_ch.R
* MCH-forearm.stretch.ik.R
* MCH-forearm.stretch.ik.R\_end
* MCH-upper\_arm.ik.R
* MCH-forearm.ik.R
* MCH-forearm.ik.R\_end
* MCH-upper\_arm.nostr.ik.R
* MCH-forearm.nostr.ik.R
* MCH-forearm.nostr.ik.Rend
* MCH-upper\_arm.stretch.ik.R
* MCH- upper\_arm.stretch.ik.R\_end
* MCH-shoulder\_rh\_ns\_ch.L
* MCH-elbow\_hose\_p.L
* elbow\_hose.L
* elbow hose.L\_end
* MCH-forearm\_hose\_end\_p.L
* forearm\_hose\_end.L
* forearm \_hose\_end.L\_end
* MCH-forearm\_hose\_p.L
* forearm hose.L
* forearm\_hose.L\_end
* MCH-upper\_arm\_hose\_end\_p.L
* upper\_arm\_hose\_end.L
* upper\_arm\_hose\_end.L\_end
* MCH-upper\_arm\_hose\_p.L
* upper\_arm\_hose.L
* upper\_arm\_hose.L\_end
* MCH-shoulder\_rh\_ns\_ch.R
* MCH-elbow\_hose\_p.R
* elbow\_hose.R
* elbow\_hose.R\_end
* MCH-forearm\_hose\_end\_p.R
* forearm\_hose\_end.R
* forearm \_hose\_end.R\_end
* MCH-forearm\_hose\_p.R

Для данных объектов используется материал Mat\_Blocky\_Person\_Blue\_2

1. Объект BigMaze состоит из следующих объектов:

* StageBorder Mesh\_02
* StageBorder Mesh\_03
* StageBorder\_Mesh\_04
* StageBorder\_Mesh\_05
* StageBorder Mesh\_06
* StageBorder Mesh\_07
* StageBorder Mesh\_08
* StageBorder Mesh\_09
* StageBorder Mesh\_010
* StageBorder Mesh\_07 (1)
* Stage Meshs
* Stage Mesh\_01
* Stage Mesh\_02
* Stage Mesh\_03
* Stage Mesh\_o4
* Stage Mesh\_05
* Stage Mesh\_06
* Stage Mesh\_07
* Stage Mesh\_08
* Stage Mesh\_09
* Stage Mesh\_02
* Stage Mesh\_03
* Stage Mesh\_o4
* Stage Mesh\_05
* Stage Mesh\_06
* Stage Mesh\_07
* Stage Mesh\_08
* Stage Mesh\_09
* Stage Mesh\_010
* Stage Mesh\_07 (1)
* Stage Mesh\_07 (2)
* Stage Mesh\_05

Для данных объектов используется материал Stage\_Gray.

3. 30 объектов cubes.

4.Объект Canvas включает в себя объекты:

* CountText
* WinText

5.Объект Directional Light

# **4. ЛИСТИНГ ПРОГРАММЫ**

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

public class control : MonoBehaviour

{

public CharacterController controller; //переменная типа компонента CharacterController

public float speedMove = 15f; //переменная отвечает за скорость движения вперёд-назад

public float speedRotation = 100f; //переменная отвечает за скорость поворота по сторонам, когдабудем двигать мышь вправо-влево

public Text countText, winText;//переменная для хранения компонента текст

private int count, count1;//переменная в которую записывает число собранных кубиков

void Start()

{

count = 0;

count1 = 30;

winText.text = " ";

setCount();

Cursor.visible = false;//отключаем видимость курсора

Cursor.lockState = CursorLockMode.Locked;//ставим курсор по центру

}

// Update is called once per frame

void Update()

{

if (controller.isGrounded)//метод проверки находится ли объект на земле

{

float vertical = Input.GetAxis("Vertical");//создаём переменную для которой записываем метод проверки нажатия клавиш движения вперёд-назад

float mouse = Input.GetAxis("Mouse X"); //переменная в которую записывается вращение мыши вправо-влево

if (vertical != 0)//проверка движения по вертикали

{

controller.Move(transform.forward \* vertical\*speedMove\*Time.deltaTime);//Move-это метод для движения персонажа встроенный в компоненте CharacterController

}

if (mouse!=0)//проверка поворота мыши

{

transform.Rotate(transform.up \* mouse \* speedRotation \* Time.deltaTime);

}

}

controller.Move(Physics.gravity \* Time.deltaTime);

}

void OnTriggerEnter(Collider other)//функция которая срабатывает при касании объектов с включенным триггером у коллайдера

{

if (other.tag == "Cube")

{

Destroy(other.gameObject);

count++;

count1--;

setCount();

}

}

void setCount()

{

countText.text = "Собрано блоков: " + count.ToString() +

"\nОсталось: " + count1.ToString();

if (count >= 30)

winText.text = "Вы собрали все блоки!";

}

}

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class Rotate : MonoBehaviour

{

void Update()

{

transform.Rotate(new Vector3(30, 35, 40) \* Time.deltaTime);

}

}