

The Escaping Balls - Multiplayer

Opis gry:

Gra zręcznościowa, w której wcielamy się w małą kulkę. Naszym zadaniem jest uciekanie przed „złymi” kulkami.

Za każdą zdobytą monetę otrzymujemy punkt.

W pełni działa „Multiplayer”, ale TYLKO w sieci LAN - serwer dla dwóch osób.

Wskazówka:

„Złe” piłki mają większą prędkość ale większy poślizg!

Jak grać „Multiplayer” - tylko LAN?

- Serwer:

- * Port domyślny 7777 (musi być odblokowany),
- * Klikamy „START”,
- * Wymagane jest konto z prawami „Administratora” na PC.

- Klient:

- * Analogicznie port domyślny 7777 (musi być odblokowany),
- * Klikamy „JOIN”.

- Jeśli nie działa należy wyłączyć „Zaporę Windows” lub dodać wyjątek,

- Na serwer przypada 2 osoby (klient/serwer).

Zgodnie z wpisem na forum Unity3D:

“Can I use Unity Multiplayer without Unity Relay Server and Unity Matchmaking?
Sure! You can continue to use the Low Level and High Level API and stand up your own servers or try NAT punchthrough yourself. At that point, you’re just paying for your Unity License (if you are paying for a Unity License).”

https://forum.unity.com/threads/unity-multiplayer-launch-faq-march-2016.392474/?_ga=2.130552591.1654061704.1524296861-344596289.1523904138

W tym przypadku nie płacę za licencje Unity (Unity Personal - FREE) i nie płacę za „Multiplayer” bo samodzielnie próbuje NAT lub własny serwer

The Escaping Balls 2 - Multiplayer

Utworzyłem kolejną wersję gry z całkowicie nową mapą w kolorze niebieskim. Podana wersja gry również wspiera „Multiplayer”, a także „AI”.

Przykład „Sterowania” w grze:

```
1  using UnityEngine;
2  using System.Collections;
3  using UnityEngine.Networking;
4
5  public class Sterowanie : NetworkBehaviour {
6
7      private Rigidbody grawitacja;
8
9      public float speed = 20;
10
11     void Start () {
12
13         this.gameObject.transform.position = new Vector3(this.gameObject.transform.position.x,-2, this.gameObject.transform.position.z);
14         grawitacja = GetComponent <Rigidbody>();
15     }
16
17     void OnEnable()
18     {
19         gameObject.AddComponent(typeof(Niewrazliwosc));
20     }
21
22     void Update () {
23
24         if(!isLocalPlayer)
25             return;
26
27         float x = Input.GetAxis("Horizontal");
28         float y = Input.GetAxis("Vertical");
29
30         grawitacja.velocity = new Vector3(x * speed * Time.deltaTime, 0, y * speed * Time.deltaTime);
31     }
32 }
33
34
35
```