

VR játék fejlesztése memória javítására

Készítette: Vörös Bálint Miklós (Mérnökinformatikus BSc)

Témavezető: Szabó Patrícia

Bevezetés

Alkalmazás célja

Alkalmazás létjogosultsága

Témaválasztás indoka

Versenyképessége

Feladat bemutatása

Az alkalmazás
tematikája/felépítése

Funkciók bemutatása

Az alkalmazás
különböző játécai

Simon játék

Bevásárló listás játék

Ki számára fontos?

Idősebb
korosztály

Memória
zavaros
betegeknek

Feledékeny
embereknek

Gyermekeknek

Oktatóknak



Bevásárló listás játék



Simon játék

Játéktípusok



Versenytárs elemzés

Water Bears VR

- Trópusi témájú puzzle játék
- VR térben megvalósított
- Gyermek számára készített



Enchance VR

- 5 féle különböző játék
- Korcsoporttól független
- Futurisztikus témát vesz alapul



Unity



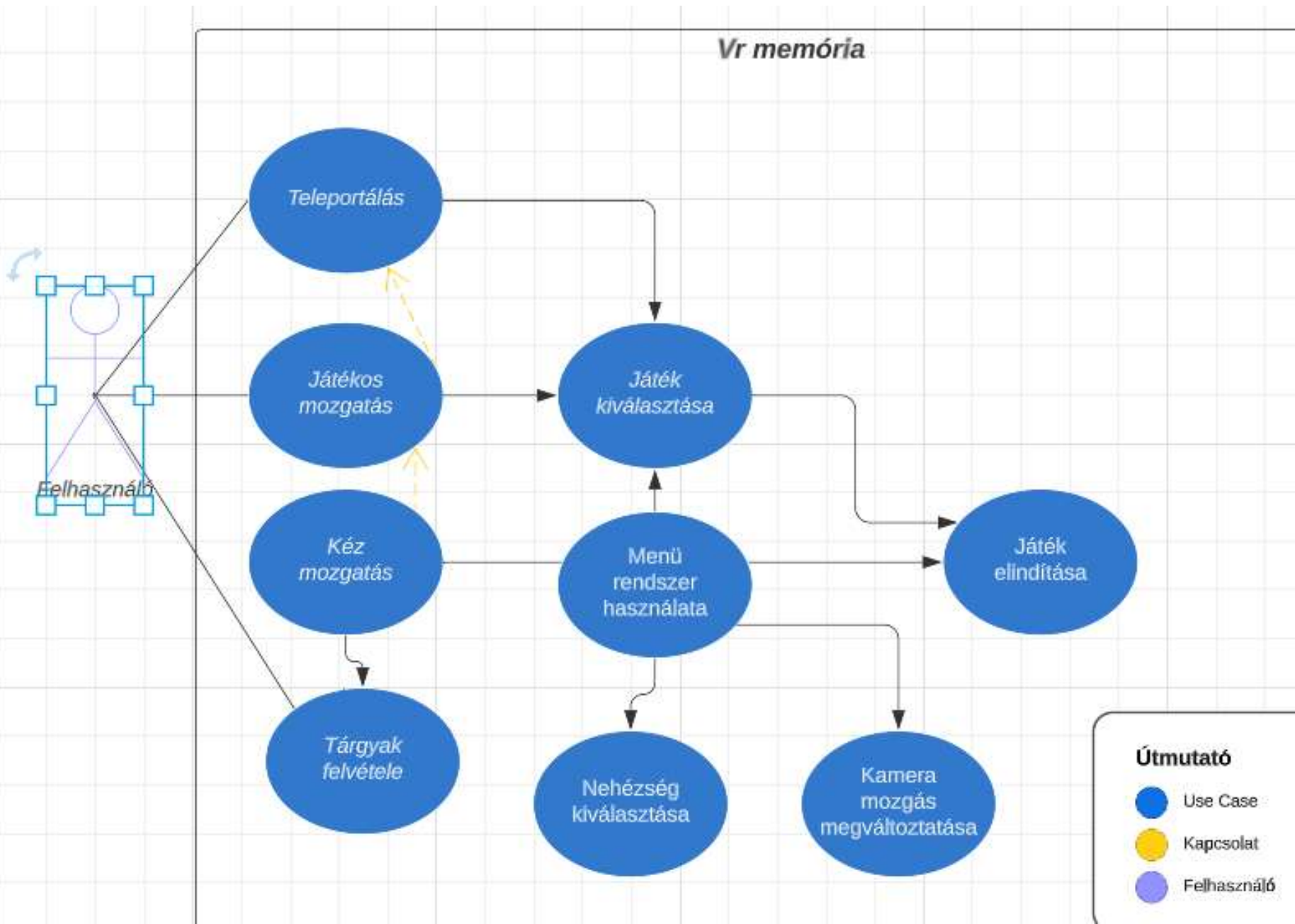
Felhasznált technológiák



- Blender
- Unity
- C#
- VS Code
- MySQL
- Github



Alkalmazás bemutatása



- Játékok
- Nehézség állítás
- Kamera funkciók átállítása
- Teleportálás
- Menürendszer

Games	
GameID	int
GameName	varchar
description	text

InventoryItems	
ItemID	int
ItemName	varchar
ItemDescription	text

PlayerInventory	
playerID	int
ItemID	int
CollectedQty	int
collectedDate	datetime

player	
playerID	int
Username	varchar
Email	varchar
Date joined	datetime

Scores	
ScoreID	int
playerID	int
GameID	int

Adatbázis felépítése

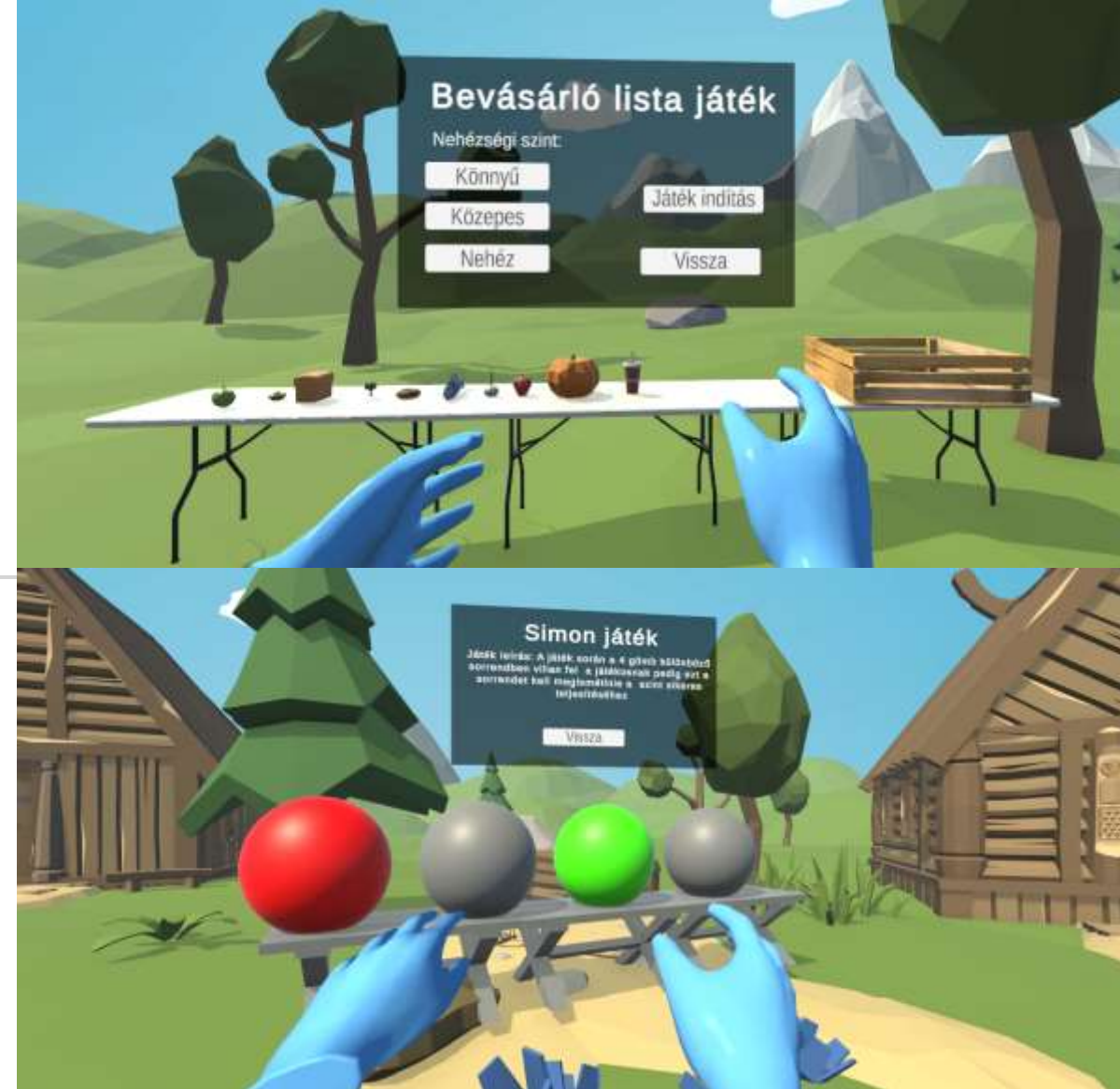


Elkészült munka

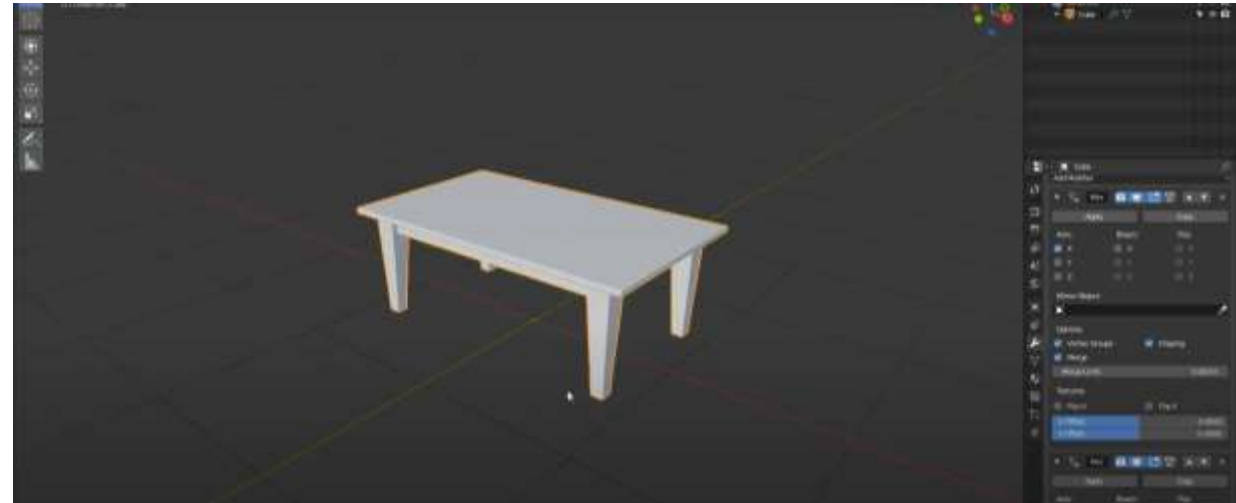
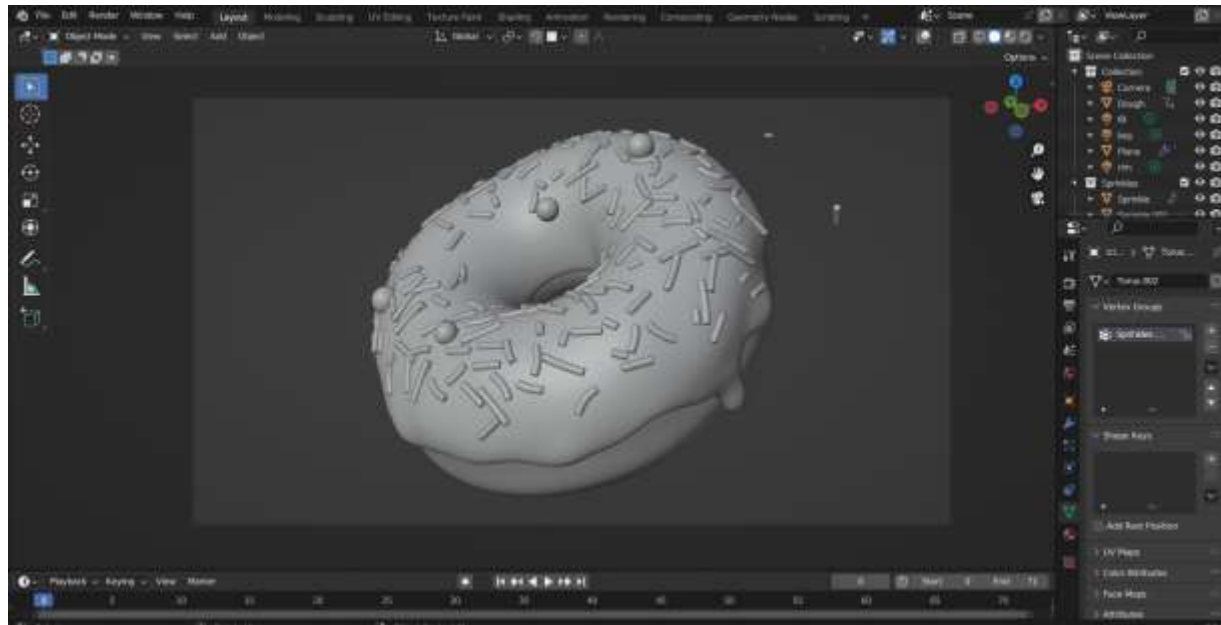
Menürendszer

Elkészült munka

- Bevásárló listás játék
- Simon játék



Modellezés



Funkciók megvalósítása

```
public AudioClip successSound;

public List<Transform> spawnPoints;

public int currentScore = 0;
public List<ItemCode> trueSequence = new List<ItemCode>();

public ScoreScreen scoreScreen;

private AudioSource audioSource;

void Start()
{
    currentState = GameStates.Idle;
    Instance = this;
    audioSource = GetComponent<AudioSource>();
}

public void OnButtonPressed()
{
    if (currentState == GameStates.Idle)
    {
        scoreScreen.UpdateScore(currentScore);
        currentState = GameStates.PlayingSequence;
        StartCoroutine(PlaySequence());
    }
}

public void OnEmergencyButtonPressed() {
    if (currentState == GameStates.ArrangeItem)
    {
```

```
public enum ItemCode : int
{
    Red = 0,
    Green = 1,
    Blue = 2,
    Yellow = 3,
    Black = 4,
}

public class Cuboltum : MonoBehaviour
{
    public ItemCode code;
    public AudioClip sound;
    public Material color;

    private AudioSource audioSource;

    private void Start()
    {
        audioSource = GetComponent<AudioSource>();
        audioSource.clip = sound;

        GetComponent<Renderer>().material = color;
    }

    private void OnTriggerEnter(Collider other)
    {
        if (other.gameObject.tag == "Table")
        {
            other.gameObject.GetComponent<Table>().OnItemAttached(this);
            audioSource.Play();
        }
    }

    private void OnTriggerExit(Collider other)
    {
        if (other.gameObject.tag == "Table")
        {
            other.gameObject.GetComponent<Table>().OnItemRemoved(this);
        }
    }
}
```

Simon játék kódjai:

- Színválasztás
- Játékindítás
- Eredményjelzés

Funkciók megvalósítása

Bevásárló lista játék kódjai:

- Tárgy hozzáadás
- Kosár ellenőrzés
- Azonos tárgyak kiszűrése

```
using System.Collections.Generic;
using UnityEngine;

public class ShoppingCart : MonoBehaviour
{
    public static ShoppingCart Instance { get; private set; }
    private List<Item> itemsCollected;

    public Transform cartTransform; // Assign this via the Unity Inspector

    private void Awake()
    {
        if (Instance == null)
        {
            Instance = this;
            itemsCollected = new List<Item>();
            DontDestroyOnLoad(gameObject);
        }
        else
        {
            Destroy(gameObject);
        }
    }
}
```

```
public void AddItem(Item item)
{
    itemsCollected.Add(item);
    Debug.Log("Item added: " + item.itemName);
}

public Vector3 GetCartPosition()
{
    return cartTransform.position;
}

public void ListItems()
{
    Debug.Log("Items in Cart:");
    foreach (Item item in itemsCollected)
    {
        Debug.Log(item.itemName);
    }
}
```

A torn piece of a yellow calendar with black numbers. The numbers visible are 11, 12, 13, 14, 18, 19, 20, 21, 26, 27, and 28. The calendar is torn along a diagonal line.

Tesztelés

- Eddig Fejlesztői Tesztelések
- Későbbiekben Felhasználói tesztelés

Megvalósult feladatok

VR Memória játékok

- -Simon játék
- - Bevásárlólista játék

Adatbázis

- -MYSQL

Modellezés

- -Környezet modellek
- -Étel modellek
- -Bútor modellek