**Міністерствоосвітиі наукиУкраїни**

**Національнийуніверситет «Львівська політехніка»**

**Інститут телекомунікацій, радіоелектроніки та електронноїтехніки**



Звіт з лабораторної роботи

з дисципліни «Розробка мобільних додатків»

На тему: «Розробка мобільного додатку»

Виконав:

ст. гр. ТР-45

Пилип С.С.

Прийняв:

Андрущак В.С.

Львів 2022

**Візуальна Складова:**

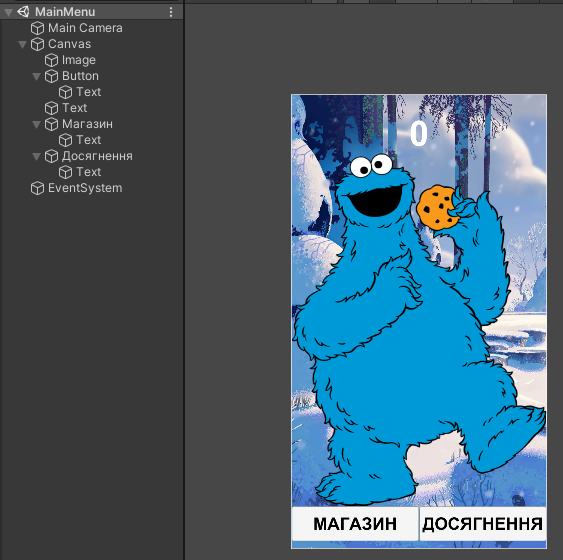
Візуальна складова, з якою буде взаємодіяти користувач, налаштована в середовищі Unity.

При першому запуску середовища створюємо проект нашої гри та налаштовуємо наступні параметри:

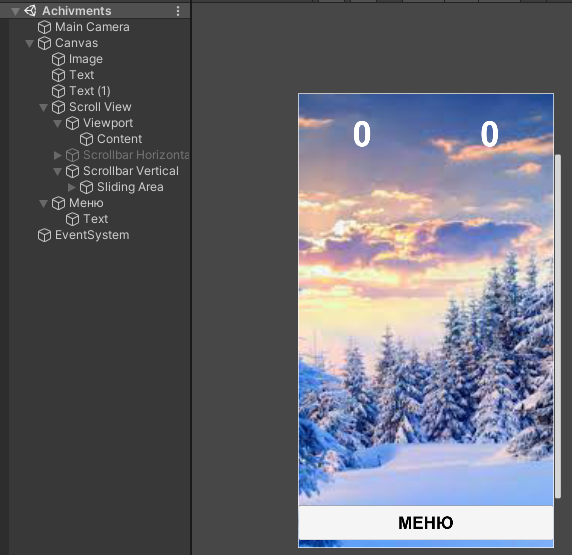
* Тип ОС: Android
* Тип підтримуваної архітектури: ARM64
* Мінімальна версія ОС: Android 4.4 ‘KitKat’
* Назва компанії: SergiosProdaction
* Назва гри: Clicker Game
* Версія: 1.4(Остання встановлена мною версія)

Дана гра має три сцени які виглядають так:

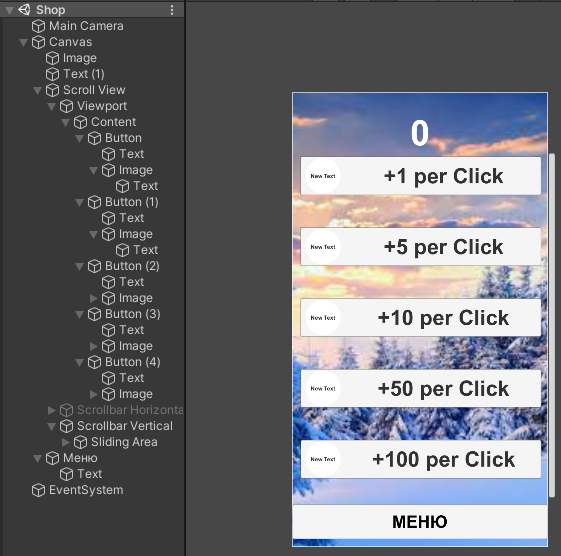
1) Сцена MainMenu:



2) Сцена Achivments:



3) Сцена MainMenu:



Вся логіка додатку написана в трьох файлах .cs, які написані мовою C#.

Файл MainMenu.cs є головним файлом у якому створені основні змінні для підрахунку зароблених коштів та кількості натисків на кнопу.

/\*Підключння необхідних бібліотек для роботи з елементами Unity\*/

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

using UnityEngine.SceneManagement;

/\*Основний клас для роботи сцени MainMenu\*/

public class MainMenu : MonoBehaviour

{

/\*Основні змінні даного класу\*/

[SerializeField] int money;

public int total\_money;

public Text moneyText;

public int K;

[SerializeField] int C;

Також створені функція задання початкових параметрів Start(). Функції для підрахунку кількості кліків та суми ігрової валюти ButtonClick(). Продубльована функція для авто-кліків IdleFarm(), яка починає працювати пісяля виконання першого досягнення. Функція переходу до меню Досягнень ToAchievements(). Функція переходу до Меню магазину ToShop(). Функція яка оновлюється 20 разів за секунду і відображає зміну клількості зароблених коштів Update().

/\*Функція задання параметрів при кожному запуску сцени\*/

private void Start()

{

money = PlayerPrefs.GetInt("money");

total\_money = PlayerPrefs.GetInt("total\_money");

K = PlayerPrefs.GetInt("K");

C = PlayerPrefs.GetInt("C");

StartCoroutine(IdleFarm());//ферма

}

/\*Функція роботи Кнопки на яку ми натискаємо\*/

public void ButtonClick()

{

money+=1+C;

total\_money++;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("total\_money", total\_money);

}

/\*Функція авто-кліків\*/

IEnumerator IdleFarm()

{

K = PlayerPrefs.GetInt("K");

yield return new WaitForSeconds(1);

money+=K;

PlayerPrefs.SetInt("money", money);

StartCoroutine(IdleFarm());

}

/\*Функція переходу до меню Досягнень\*/

public void ToAchievements()

{

SceneManager.LoadScene(1);

}

/\*Функція переходу до меню Магазину\*/

public void ToShop()

{

SceneManager.LoadScene(2);

}

/\*Функція оновлення виводу кількості заробленої ігрової валюти\*/

void Update()

{

moneyText.text = money.ToString()+" $";

}

}

Файл AchMenu.cs є файлом сцени Achivments у якому створені основні змінні для підтвердження виконання досягнень та обєкти за допомогою яких можна створити конопки по шаблону і помістити їх в слайдер та призначити дію цим кнопакам.

/\*Підключння необхідних бібліотек для роботи з елементами Unity\*/

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

using UnityEngine.SceneManagement;

/\*Основний клас для роботи сцени Achivments\*/

public class AchMenu : MonoBehaviour

{

/\*Основні змінні даного класу\*/

public Text moneyText, total\_moneyText;

public int money;

public int total\_money;

[SerializeField] int K;//змінна для роботи авто-кліків

private bool SET0, SET1, SET2, SET3, SET4, SET5;//змінні виконання досягень

public string[] arrayTitles;// назва досягнення

public Sprite arraySprites;// масив картинок для досягнень

public GameObject button;// шаблон кнопки

public GameObject content;

private List<GameObject> list = new List<GameObject>();

private VerticalLayoutGroup \_group;

Також створені функція задання початкових параметрів Start(). Допоміжна функція для очистки елементів RemovedList(). Функція для створення кнопок SetAchievs(), які поміщаються в Scroll View -> Content. Функція для опрацювання виконання досягнень GetAchievment(), після виконання яких гравцеві надається доступ до автоматичного заробляння коштів. Функція для авто-кліків IdleFarm(), яка починає працювати після виконання першого досягнення. Функція переходу до Головного Меню ToMenu(). Функція яка оновлюється 20 разів за секунду і відображає зміну клількості зароблених коштів та зроблених кліків Update().

/\*Функція задання параметрів при кожному запуску сцени\*/

void Start()

{

money = PlayerPrefs.GetInt("money");

total\_money = PlayerPrefs.GetInt("total\_money");

K = PlayerPrefs.GetInt("K");

SET0 = PlayerPrefs.GetInt("SET0") == 1 ? true : false;

SET1 = PlayerPrefs.GetInt("SET1") == 1 ? true : false;

SET2 = PlayerPrefs.GetInt("SET2") == 1 ? true : false;

SET3 = PlayerPrefs.GetInt("SET3") == 1 ? true : false;

SET4 = PlayerPrefs.GetInt("SET4") == 1 ? true : false;

SET5 = PlayerPrefs.GetInt("SET5") == 1 ? true : false;

RectTransform rectT = content.GetComponent<RectTransform>();

rectT.transform.localPosition = new Vector3(0.0f, 0.0f, 0.0f);// нульові кординати для кнопок

\_group = GetComponent<VerticalLayoutGroup>();

setAchievs();

/\*Перевірка виконнання досягнень та вимикання їх повторного отримання\*/

if (SET0 == true)

{

list[0].GetComponent<Button>().interactable = false;

}

if (SET1 == true)

{

list[1].GetComponent<Button>().interactable = false;

}

if (SET2 == true)

{

list[2].GetComponent<Button>().interactable = false;

}

if (SET3 == true)

{

list[3].GetComponent<Button>().interactable = false;

}

if (SET4 == true)

{

list[4].GetComponent<Button>().interactable = false;

}

if (SET5 == true)

{

list[5].GetComponent<Button>().interactable = false;

}

StartCoroutine(IdleFarm());

}

/\*Допоміжна функція очистки елементів\*/

public void RemovedList()

{

foreach (var elem in list)

{

Destroy(elem);

}

list.Clear();

}

/\*Функція створення кнопок для отримання Досягнень\*/

void setAchievs()

{

RectTransform rectT = content.GetComponent<RectTransform>();//зчитування кнопки шаблону

rectT.transform.localPosition = new Vector3(-360.0f, 10.0f, 0.0f);//встановлення початкових координат кнопки

RemovedList();

if (arrayTitles.Length > 0)

{

var pr1 = Instantiate(button, transform);// створення образца

var h = pr1.GetComponent<RectTransform>().rect.height;// запис висоти

var tr = GetComponent<RectTransform>();// створення змінної

tr.sizeDelta = new Vector2(tr.rect.width, h \* arrayTitles.Length);

Destroy(pr1);

for(var i = 0; i < arrayTitles.Length; i++)//цикл досягнень

{

var pr = Instantiate(button, transform);

pr.GetComponentInChildren<Text>().text = arrayTitles[i];//запис тексту

pr.GetComponentInChildren<Image>().sprite = arraySprites;//запис іконки

var i1 = i;

pr.GetComponent<Button>().onClick.AddListener(() => GetAchievement(i1));// опрацювання нажиму на кнопки

list.Add(pr);// добавити досягнення до списку

}

}

}

/\*Функція опрацювання виконання Досягень\*/

void GetAchievement(int id)

{

switch(id)

{

case 0:

if (!SET0 && total\_money >= 10)

{

money += 10;

K += 1;

SET0 = true;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("K", K);

PlayerPrefs.SetInt("SET0", 1);

list[0].GetComponent<Button>().interactable = false;

}

Debug.Log(id);

break;

case 1:

if (!SET1 && total\_money >= 50)

{

money += 50;

K += 5;

SET1 = true;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("K", K);

PlayerPrefs.SetInt("SET1", 1);

list[1].GetComponent<Button>().interactable = false;

}

Debug.Log(id);

break;

case 2:

if (!SET2 && total\_money >= 100)

{

money += 100;

K += 10;

SET2 = true;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("K", K);

PlayerPrefs.SetInt("SET2", 1);

list[2].GetComponent<Button>().interactable = false;

}

Debug.Log(id);

break;

case 3:

if (!SET3 && total\_money >= 500)

{

money += 500;

K += 50;

SET3 = true;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("K", K);

PlayerPrefs.SetInt("SET3", 1);

list[3].GetComponent<Button>().interactable = false;

}

Debug.Log(id);

break;

case 4:

if (!SET4 && total\_money>= 1000)

{

money += 1000;

K += 100;

SET4 = true;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("K", K);

PlayerPrefs.SetInt("SET4", 1);

list[4].GetComponent<Button>().interactable = false;

}

Debug.Log(id);

break;

case 5:

if (!SET5 && total\_money >= 10000)

{

money += 10000;

K += 1000;

SET5 = true;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("K", K);

PlayerPrefs.SetInt("SET5", 1);

list[5].GetComponent<Button>().interactable = false;

}

Debug.Log(id);

break;

}

}

/\*Функція авто-кліків\*/

IEnumerator IdleFarm()

{

K = PlayerPrefs.GetInt("K");

yield return new WaitForSeconds(1);

money+=K;

PlayerPrefs.SetInt("money", money);

Debug.Log(K);

StartCoroutine(IdleFarm());

}

/\*Функція повернення до сцени MainMenu\*/

public void ToMenu()

{

SceneManager.LoadScene(0);

}

/\*Функція оновлення виводу кількості заробленої ігрової валюти та кількості накисків\*/

void Update()

{

total\_moneyText.text = total\_money.ToString() + " Clics";

moneyText.text = money.ToString() + " $";

}

}

Файл ShopMenu.cs є файлом сцени Shop у якому створені основні змінні для можливості здійснювати покупки натискаючи на кнопки, змінні для показу ціни та змінні які збільшують вартість кожної наступної покупки, а також змінна яка збільшує кількість ігрової валюти яку гравець може отримати при одному натиску на кнопку.

/\*Підключння необхідних бібліотек для роботи з елементами Unity\*/

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

using UnityEngine.SceneManagement;

/\*Основний клас для роботи сцени Shop\*/

public class ShopMenu : MonoBehaviour

{

/\*Основні змінні даного класу\*/

public Text moneyText;

public Text TSET0;

public Text TSET1;

public Text TSET2;

public Text TSET3;

public Text TSET4;

public int money;

public int K;

private int FSET0, FSET1, FSET2, FSET3, FSET4;

public int l = 1;

public int C;

private int sum0, sum1, sum2, sum3, sum4;

Також створені функція задання початкових параметрів Start(). Функція для опрацювання роботи першої кнопки магазину GetShop0(), в якій проходить операція покупки додаткової суми, яку може отримати гравець та перерахунок вартості наступної покупки. Функція для опрацювання роботи другої кнопки магазину GetShop1(). Функція для опрацювання роботи третьої кнопки магазину GetShop2(). Функція для опрацювання роботи четвертої кнопки магазину GetShop3(). Функція для опрацювання роботи п’ятої кнопки магазину GetShop4(). Дублювання функції для авто-кліків IdleFarm. Функція переходу до Головного Меню ToMenu(). Функція яка оновлюється 20 разів за секунду і відображає зміну клількості зароблених коштів та суми кожної наступної покупки в магазині Update().

/\*Функція задання параметрів при кожному запуску сцени\*/

void Start()

{

money = PlayerPrefs.GetInt("money");

C = PlayerPrefs.GetInt("C");

//total\_money = PlayerPrefs.GetInt("total\_money");

K = PlayerPrefs.GetInt("K");

FSET0 = PlayerPrefs.GetInt("FSET0");

FSET1 = PlayerPrefs.GetInt("FSET1");

FSET2 = PlayerPrefs.GetInt("FSET2");

FSET3 = PlayerPrefs.GetInt("FSET3");

FSET4 = PlayerPrefs.GetInt("FSET4");

sum0 = (int)(Mathf.Pow(1.2f, FSET0) \* 10);

sum1 = (int)(Mathf.Pow(1.2f, FSET1) \* 50);

sum2 = (int)(Mathf.Pow(1.2f, FSET2) \* 100);

sum3 = (int)(Mathf.Pow(1.2f, FSET3) \* 500);

sum4 = (int)(Mathf.Pow(1.2f, FSET4) \* 1000);

StartCoroutine(IdleFarm());

}

/\*Функція роботи кнопки 1\*/

public void GetShop0()

{

float d;

Debug.Log(sum0);

if (money >= sum0)

{

money -= sum0;

FSET0 ++;

C += 1;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("C", C);

PlayerPrefs.SetInt("FSET0", FSET0);

}

d = 10 \* Mathf.Pow(1.2f,FSET0);

sum0 = (int)d;

Update();

}

/\*Функція роботи кнопки 2\*/

public void GetShop1()

{

float d;

if (money >= sum1)

{

money -= sum1;

FSET1 ++;

C += 5;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("C", C);

PlayerPrefs.SetInt("FSET1", FSET1);

}

d = 50 \* Mathf.Pow(1.2f, FSET1);

sum1 = (int)d;

Update();

}

/\*Функція роботи кнопки 3\*/

public void GetShop2()

{

float d;

if (money >= sum2)

{

money -= sum2;

C += 10;

FSET2 ++;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("C", C);

PlayerPrefs.SetInt("FSET2", FSET2);

}

d = 100 \* Mathf.Pow(1.2f, FSET2);

sum2 = (int)d;

Update();

}

/\*Функція роботи кнопки 4\*/

public void GetShop3()

{

float d;

if (money >= sum3)

{

money -= sum3;

C += 50;

FSET3 ++;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("C", C);

PlayerPrefs.SetInt("FSET3", FSET3);

}

d = 500 \* Mathf.Pow(1.2f, FSET3);

sum3 = (int)d;

Update();

}

/\*Функція роботи кнопки 5\*/

public void GetShop4()

{

float d;

if (money >= sum4)

{

money -= sum4;

C += 100;

FSET4 ++;

PlayerPrefs.SetInt("money", money);

PlayerPrefs.SetInt("C", C);

PlayerPrefs.SetInt("FSET4", FSET4);

}

d = 1000 \* Mathf.Pow(1.2f, FSET4);

sum4 = (int)d;

Update();

}

/\*Функція авто-кліків\*/

IEnumerator IdleFarm()

{

K = PlayerPrefs.GetInt("K");

yield return new WaitForSeconds(1);

money += K;

PlayerPrefs.SetInt("money", money);

Debug.Log(C);

Update();

StartCoroutine(IdleFarm());

}

/\*Функція повернення до сцени MainMenu\*/

public void ToMenu()

{

PlayerPrefs.SetInt("FSET0", FSET0);

PlayerPrefs.SetInt("FSET1", FSET1);

PlayerPrefs.SetInt("FSET2", FSET2);

PlayerPrefs.SetInt("FSET3", FSET3);

PlayerPrefs.SetInt("FSET4", FSET4);

PlayerPrefs.Save();

SceneManager.LoadScene(0);

}

/\*Функція оновлення виводу кількості заробленої ігрової валюти та вартості покупок\*/

void Update()

{

moneyText.text = money.ToString() + " $";

TSET0.text = sum0.ToString();

TSET1.text = sum1.ToString();

TSET2.text = sum2.ToString();

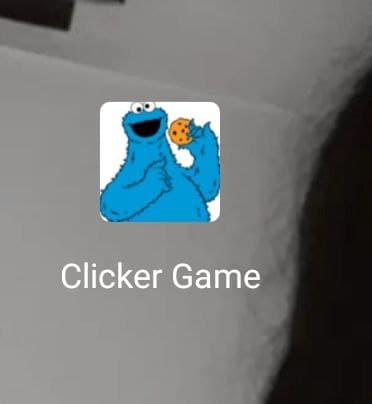
TSET3.text = sum3.ToString();

TSET4.text = sum4.ToString();

}

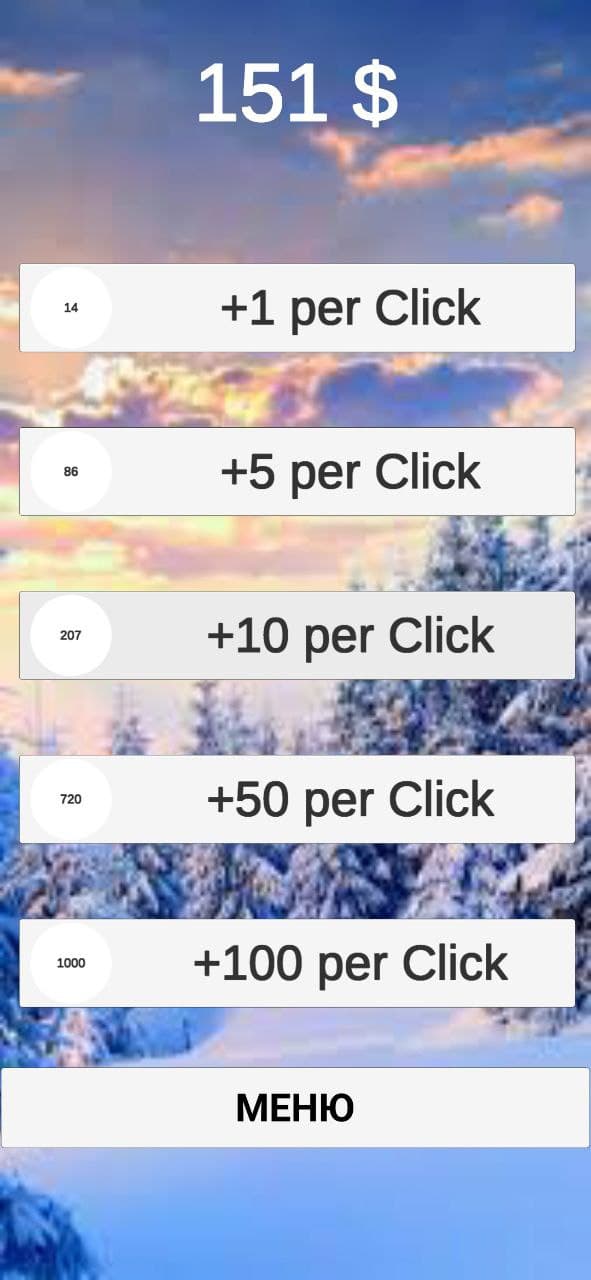
}

Далі встановлюємо додаток та заходимо до нього.



Виконання роботи програми продемонстровано на наступних скріншотах:





Також ознайомитись з грою можна за наступними посиланнями:

[**https://youtu.be/3D9QIOjQKpA**](https://youtu.be/3D9QIOjQKpA)