

FACULTATEA CALCULATOARE, INFORMATICA SI MICROELECTRONICA

UNIVERSITATEA TEHNICA A MOLDOVEI

MEDII INTERACTIVE DE DEZVOLTARE A PRODUSELOR SOFT

LUCRAREA DE LABORATOR#4

Dezvoltarea unei aplicații mobile

Autor:

Eugen MIROVSCHI

lector asistent:

Irina COJANU

lector superior:

Svetlana COJOCARU

Laboratory work #2

1 Scopul lucrării de laborator

Studierea tehnologiilor de creare a unui joc folosind platforma Unity. Crearea unui joc și lansarea acestuia pe un dispozitiv Android.

2 Obiective

- a) Cunoștințe de bază privind arhitectura unei aplicații mobile
- b) Cunoștințe de bază ale platformei SDK

3 Laboratory work implementation

3.1 Tasks and Points

- a) Realizarea unui joc
- b) Folosirea unei librării cross platform pentru a realiza o aplicație platform (aplicația poate fi compilată atât pe Android cât și iOS)

3.2 Analiza lucrării de laborator

- a) Primul pas a fost inițializarea unui nou repository pe GitHub și clonarea acestuia pe calculatorul personal: <https://github.com/emirovschi/MIDPS\4>.
- b) Projeny a fost creat utilizând Projeny[1]. Acesta oferă posibilitatea de a organiza codul în module care pot fi configurate individual pentru diferite platforme.
- c) Projeny generează proiect de tip Unity3D[2]. Avantajele acestui instrument este posibilitatea de a crea ușor jocuri pentru mai multe platforme, este customizabil, extensibil și ușor de utilizat.

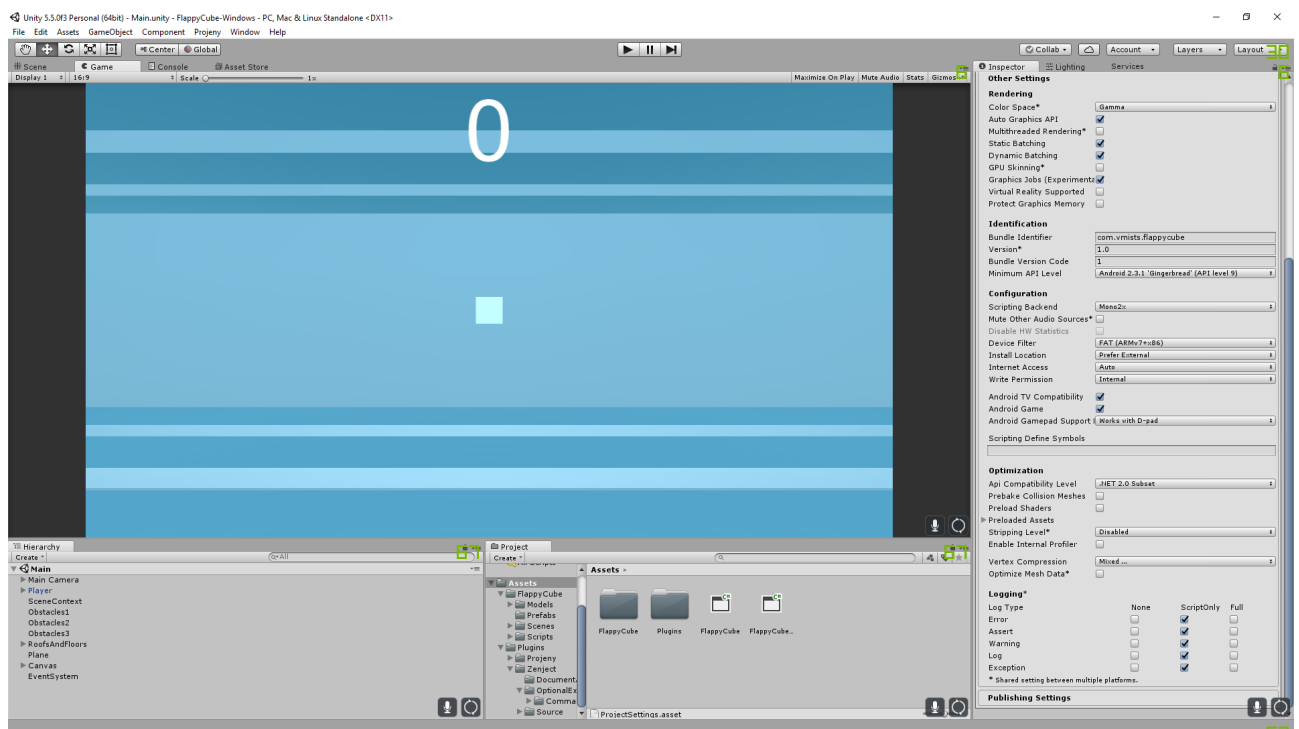


Figure 3.1 – Interfața Unity3D

- d) Unity3D utilizează platforma .NET respectiv limbajul de programare este C#[3]. Astfel pentru redactarea codului am folosit Visual Studio[4].
- e) La elaborarea acestui proiect am folosit biblioteca Zenject[5]. Ea conține un sistem de Inversion Of Control similar cu cel din Spring pentru Java. Aici regulile de injectare a dependențelor sunt definite cu ajutorul unor comenzi specifice. Un alt avantaj al acestei biblioteci este compatibilitatea sa cu Unity3D oferind posibilitatea de a injecta dependențe în componentele controlate de engine.

25 references

```
public override void InstallBindings()
{
    if (Application.isMobilePlatform)
    {
        Container.Bind<IControls>().To<TouchControls>().AsSingle();
    }
    else
    {
        Container.Bind<IControls>().To<KeyboardControls>().AsSingle();
    }

    Container.BindAllInterfaces<Game>().To<Game>().AsSingle();
    Container.Bind<ITickable>().To<GameStarter>().AsSingle();
    Container.BindAllInterfaces<GameEnder>().To<GameEnder>().AsSingle();
    Container.BindSignal<GameStartSignal>();
    Container.BindSignal<AddScoreSignal>();

    Container.BindAllInterfacesAndSelf<Player>().FromInstance(Player);
    Container.BindAllInterfacesAndSelf<PlayerController>().To<PlayerController>().AsSingle();
}
```

Figure 3.2– Exemplu de configurare a dependențelor în Zenject

- f) Pentru simplitatea am folosit modele 3D primitive existente deja în Unity3D, partea personalizată fiind reprezentată de componentele implementate în C#.
- g) Un element important al jocului dat este generarea procedurală a lumii. Aceasta are loc prin generarea unei liste de obstacole și repoziționarea lor când sunt

3.3 Imagini

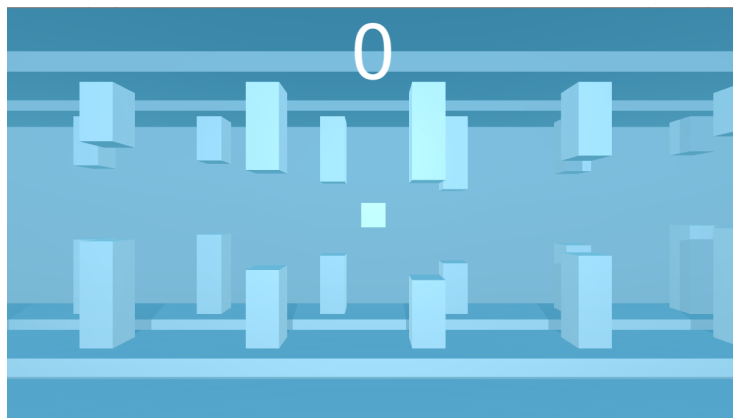


Figure 3.3– Scene inițială

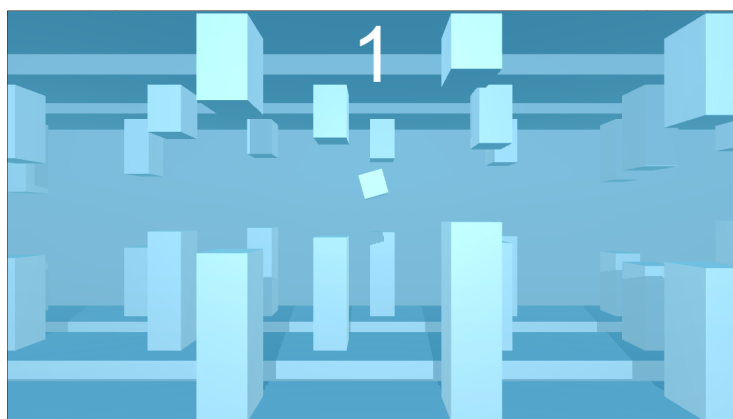


Figure 3.4– Exemplu de lume generată procedural

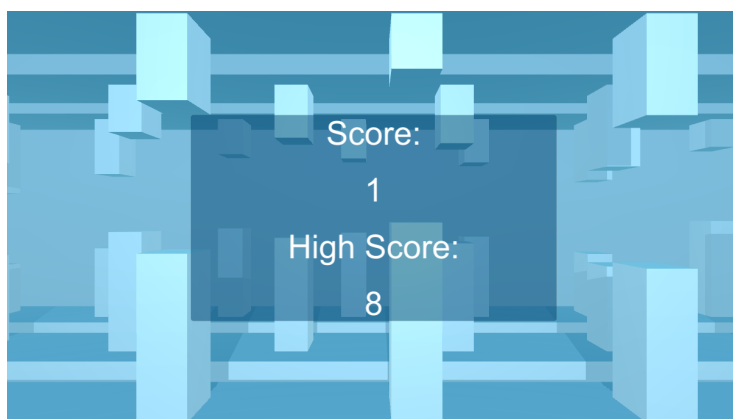


Figure 3.5– Scene de finisare și afișarea a scorului final

Concluzie

Pentru efectuarea acestei lucrări am utilizat un set de instrumente care m-au ajutat la implementarea unui joc. Am folosit git ca sistem de control al versiunilor. Acesta oferă posibilitatea de a înregistra fiecare modificare ca o versiune separată precum și crearea mai multor branch-uri. Ca mediu de dezvoltare am folosit Unity3D Editor care permite crearea lumelor virtuale. Codul însă l-am scris utilizând Visual Studio care este mediaul implicit de dezvoltare pentru limbajul de programare C#. Pentru a controla structura și modul de construire a proiectului, precum și includerea eficientă și rapidă a dependențelor externe am folosit managerul Projeny. Una din dependențele principale utilizate la acest proiect este Zenject care ușurează crearea componentelor injectând automat toate dependențele inclusiv și în componentele care aparțin platformei Unity. Toate aceste instrumente m-au ajutat să creez într-un mod eficient un joc care poate fi lansat pe mai multe platforme.

References

- 1 Projeny, *repository URL*, <https://github.com/modesttree/Projeny/>
- 2 Unity3D, *official page*, <https://unity3d.com/>
- 3 Microsoft C#, *documentation*, <https://msdn.microsoft.com/en-us/library/kx37x362.aspx>
- 4 Visual Studio, *official page*, <https://www.visualstudio.com/>
- 5 Zenject, *repository URL*, <https://github.com/modesttree/Zenject>