

COURSE OUTLINE

GAME PROGRAMMING

Oleh: Abas Setiawan

Contents (DAFTAR ISI)

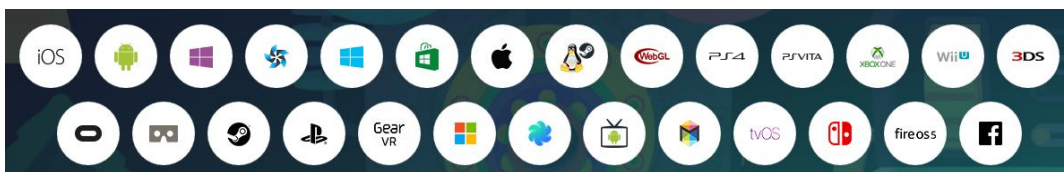
Pendahuluan	1
Android Deployment.....	1
GUI Main Menu.....	4
Fixing Bug Flappycopter	7
Bug Kendali	7
Bug Text HUD Score	7
Bug Restart.....	8
Singleton Class	8
Menambahkan Musik	9
Latihan (untuk pertemuan 6).....	10
REFERENSI	11
Mekanisme Pengumpulan Praktikum Mingguan.....	11

Pendahuluan

Pada praktikum ke-5 ini, kita akan mencoba **melanjutkan project flappycopter**. Kita juga akan membuat flappycopter agar bisa dimainkan didalam smartphone android. Disamping itu, kita juga akan membuat menu GUI game dan memberi background music untuk game flappycopter.

Android Deployment

Aplikasi yang dibuat dengan Unity memungkinkan untuk dapat di built ke banyak platform.

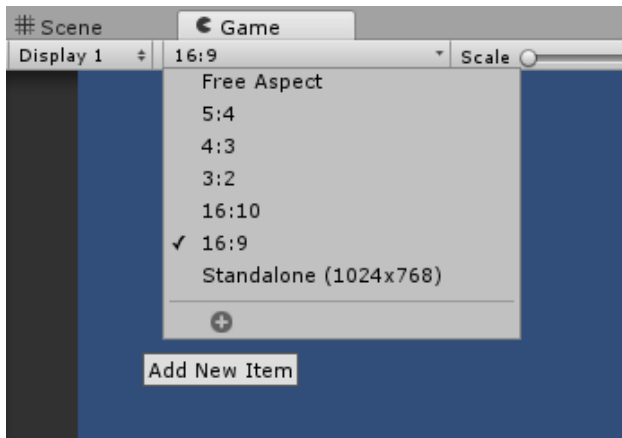


Pada saat deployment android ada beberapa hal penting yang perlu diperhatikan diawal ketika membuat project Unity. Beberapa persiapan yang harus sudah diatur adalah:

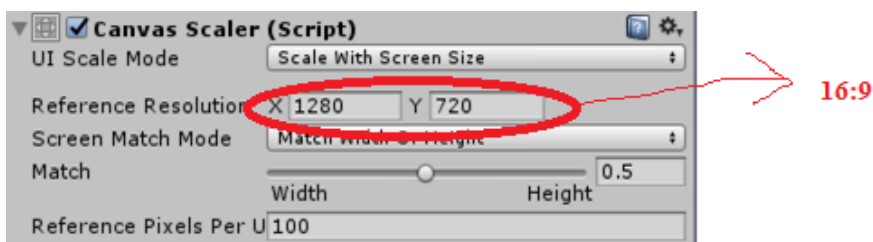
PRAKTIKUM:

1. Buka project flappycopter.

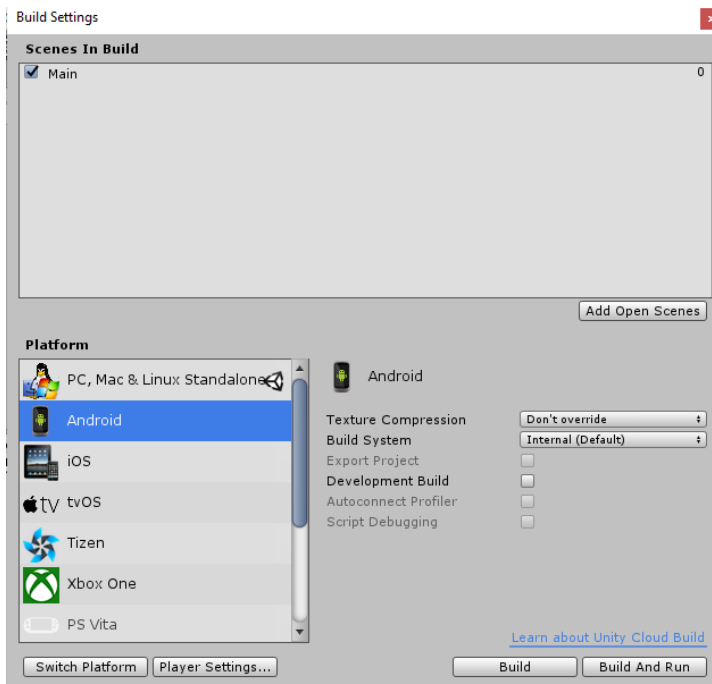
2. Pada Game tab “Game” atur aspek ratio. Aspek ratio merupakan perbandingan antara panjang dan lebar layar game yang dibuat. Dalam kasus ini aspek ratio diatur 16:9 (jika belum ada bisa “Add new item”)



3. **(SKIP LANGKAH INI, TAPI PERHATIKAN BAIK-BAIK KETIKA SUDAH ADA GUI NANTINYA!)** Ketika menggunakan GUI pada Unity. Pastikan “Canvas” harus cocok dengan aspek ratio yang sebelumnya telah diatur seperti gambar berikut **(LANGKAH INI PENTING UNTUK DILAKUKAN SAAT APLIKASI GAME YANG DIBUAT MENGGUNAKAN GUI)**:



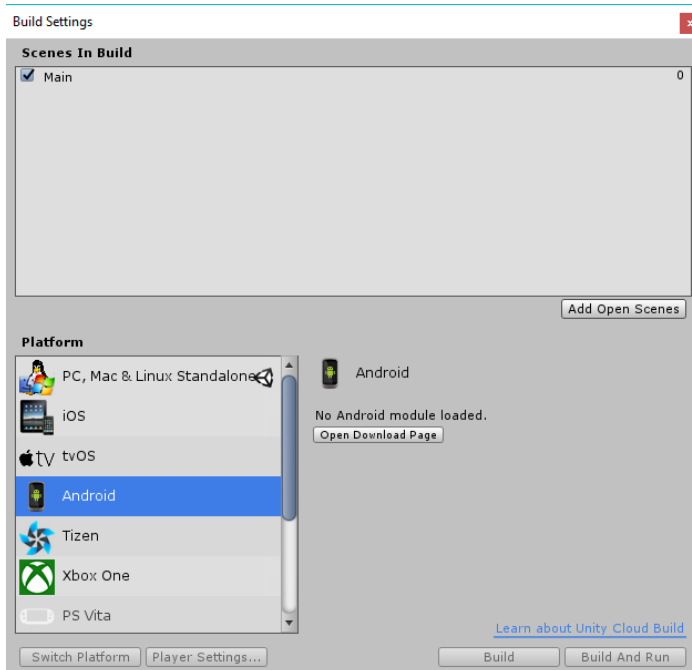
4. Pada saat akan membuat file “apk”, masuk pada File->Build Settings dan pastikan bahwa Android SDK dan scene unity yang ingin dibangun telah diatur dengan benar:



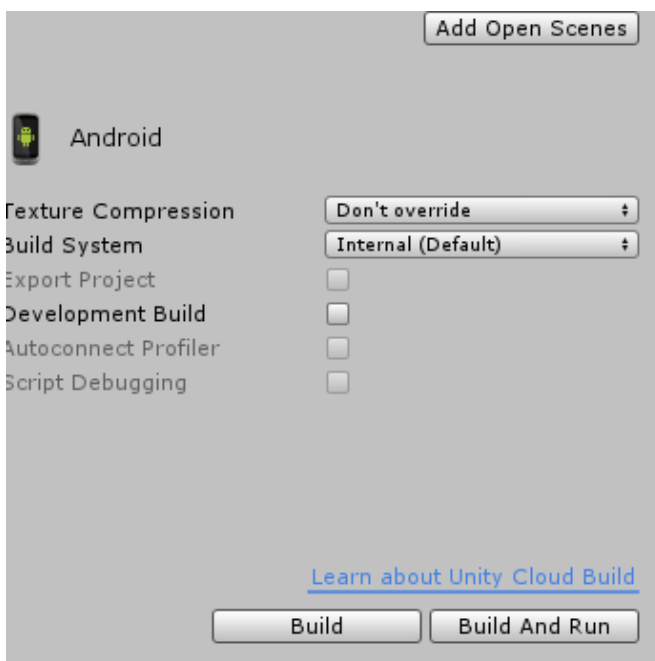
Nb. Jika belum muncul keterangan platform ketika memilih platform android, maka sebelumnya harus melakukan download dan install Android SDK

(http://filehippo.com/download_android_sdk/). Buka Android SDK Manager dan install Android API yang diinginkan(dalam kasus ini Android 6.0).

Jika belum terinstal maka di build setting akan nampak seperti gambar dibawah ini:



Klik “Open Download Page” untuk mendownload UnitySetup-Android-Support-for-Editor-5.6.0f3.exe. Close aplikasi unity editor, Instal Android Support. Setelah itu akan muncul dialog box seperti pada gambar Build Setting yang pertama.

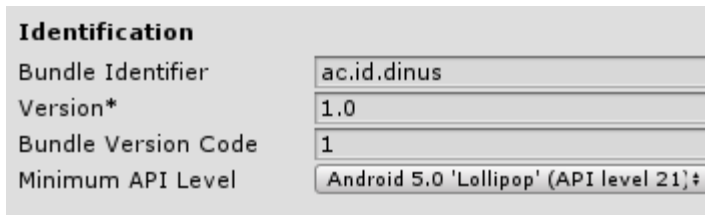


Klik Build, kemudian akan ditanya tentang nama file .apk, isikan flappycopter.apk. **Lakukan langkah 5 dan 6 terlebih dahulu!**

Tunggu beberapa saat dan kamu akan ditanya tentang Android SDK, masukkan folder Android SDK dan lanjutkan proses deployment dan tunggu beberapa saat sampai file apk jadi.

5. Klik “Switch Platform” pada “Build Settings” dan tunggu beberapa saat.

6. Klik “Player Settings” untuk mengatur manifestasi yang harus ada sebagai syarat untuk membuat apk pada Android.



Identification	
Bundle Identifier	ac.id.dinus
Version*	1.0
Bundle Version Code	1
Minimum API Level	Android 5.0 'Lollipop' (API level 21) ↑

Pada publishing setting, atur “certificate” sebelum aplikasi dibuild pada Android.



Publishing Settings	
Keystore	
<input checked="" type="checkbox"/> Use Existing Keystore	<input type="checkbox"/> Create New Keystore
<input type="button" value="Browse Keystore"/>	Browse to select keystore name
Keystore password	*****
Confirm password	
Key	
Alias	Unsigned (debug) ↑
Password	
Split Application Binary	<input type="checkbox"/>

7. Setelah semua persiapan selesai, selanjutnya adalah klik “Build” dan masukan nama file flappycopter.apk.

GUI Main Menu

GUI atau Graphical User Interface merupakan komponen penting yang harus ada dalam game. GUI dapat dimanfaatkan untuk membuat Main Menu, Menu Credits, Menu help, dan juga HUD (Head Up Display). GUI pada Unity diperlihatkan pada GameObject->UI. Berikut adalah penjelasan dari beberapa UI yang sering digunakan di Unity.

Canvas : merupakan area tempat dimana semua elemen UI bisa masuk didalamnya. Area canvas dapat diatur dengan property Render Mode.

Text: Merupakan elemen UI Unity untuk menambahkan UI text atau label.

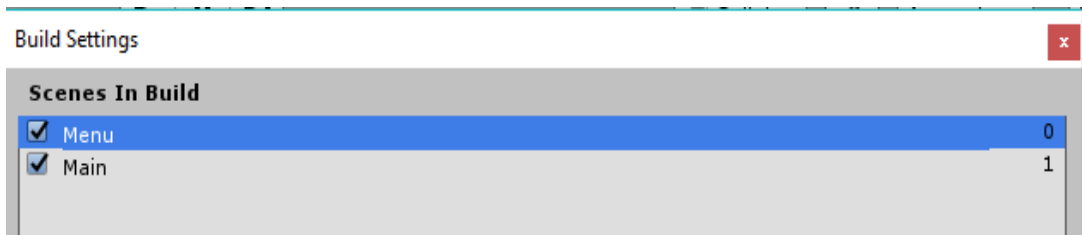
Button: Merupakan elemen UI Unity untuk menambahkan UI Button.

Image: Merupakan elemen UI Unity untuk menambahkan UI Gambar.

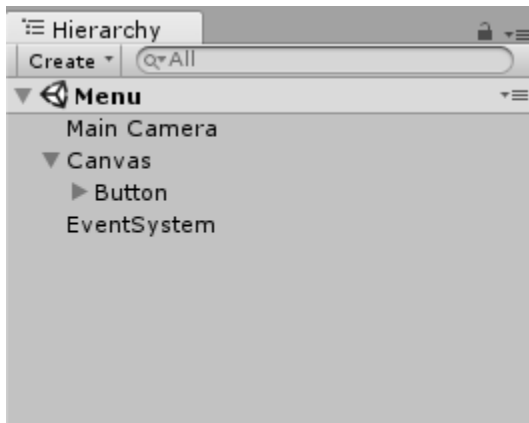
Input Field: Merupakan elemen UI Unity untuk menambahkan UI input text.

PRAKTIKUM:

1. Buat scene baru dengan cara File-> New Scene atau CTRL+N. Simpan scene baru tersebut dengan nama Menu.
2. Tarik dan letakkan pada build setting di dengan urutan index scene ke 0. Hal ini penting karena ketika game pertama kali dibuka, scene yang akan terbuka lebih dulu adalah scene Menu ini.

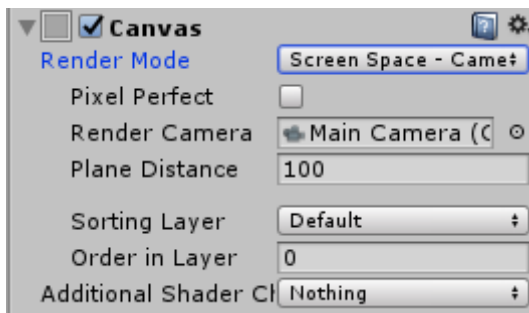


3. Masuk pada pengaturan Game Tab, pastikan aspect rasionya 16:9.
4. Pada Menu tab, GameObject->UI->Button. Maka pada Hierarchy window akan tampak seperti gambar berikut:

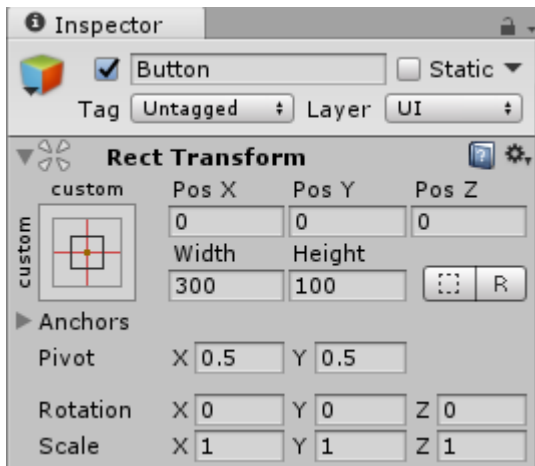


Jangan HAPUS EventSystem dan Canvas!

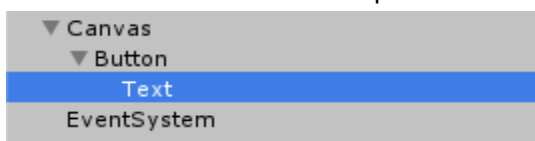
5. Klik object Canvas pada Hierarchy window dan lihat Inspector.
6. Pada Komponen Property "Canvas" -> "Render Mode", rubah nilainya menjadi "Screen Space - Camera"
7. Tarik Main Camera kedalam nilai dari Render Camera, agar terlihat seperti gambar di bawah ini:



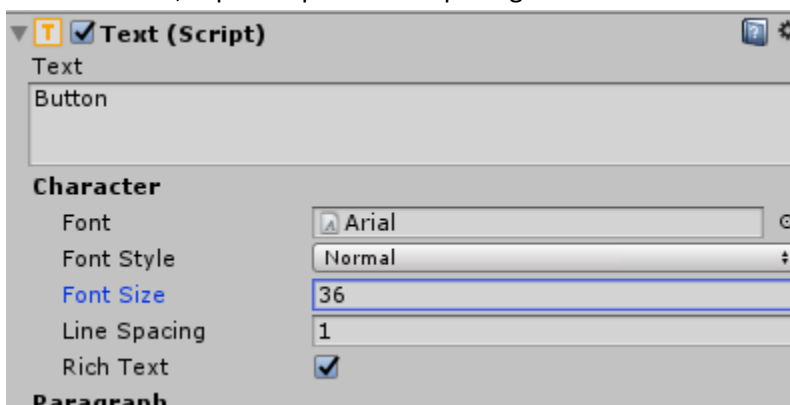
8. Rubah Canvas Scaler seperti pada langkah ke-3 dari bab Android Development.
9. Klik object Button pada hierarchy window dan lihat pada Inspector. Atur posisi dan skala dari button seperti gambar dibawah:



10. Untuk memudahkan Rename Button pada Hierarchy window menjadi "ButtonPlay".
11. Klik Text dibawah child Button pada hierarchy window:



12. Lihat Inspector dari object Text. Isikan Font Size bernilai 36 dan rubah property Text dengan tulisan "PLAY", seperti diperlihatkan pada gambar berikut:



13. Buat Game object kosong dan rename dengan nama "codeGUI".
14. Buat script baru didalam code GUI dengan nama "GUIManager", tambahkan library scene management:

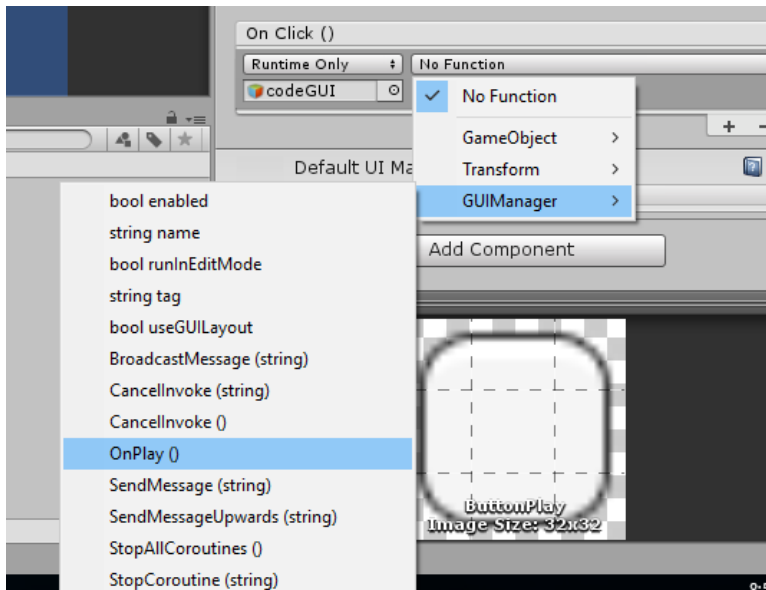
```
using UnityEngine.SceneManagement;
```

Buat method baru dengan nama OnPlay() dengan akses modifier **public**:

```
public void OnPlay()
{
    SceneManager.LoadScene ("Main");
}
```

15. Klik ButtonPlay pada hierarchy window dan lihat pada Inspector. Scrol ke bawah sampai melihat property On Click() dengan nilai List is Empty.
16. Klik symbol "+" di property On Click().
17. Tarik codeGUI dari hierarchy window ke dalam "Nobe (Object)" dibawah button Runtime Only.

18. Klik button “No Function” disebelahnya, pilih GUIManager->OnPlay. (Jika OnPlay tidak muncul, lihat script GUIManager dan **pastikan Akses modifier dari method OnPlay adalah Public**)



19. Jalankan unity dan klik button play pada game tab.
20. Lihat apa yang terjadi.

KUMPULKAN:

Screenshoot Game tab saat di menu scene dan simpan dengan nama sc_056_01.jpg.

Screenshoot Game tab saat di main scene dan simpan dengan nama sc_056_02.jpg.

Fixing Bug Flappycopter

Terdapat beberapa bug atau kesalahan fungsional program pada Flappycopter.

Bug Kendali

Terdapat bug kendali player pada saat game dimainkan di smartphone android. Bug tersebut disebabkan oleh penggunaan keyboard “spasi”. Rubah isi kode ObstacleInAct:

```
if (Input.GetKeyDown ("space"))  
menjadi
```

```
float touchclicked = Input.GetAxis ("Fire1");  
if (touchclicked == 1f)
```

Sekarang gunakan mouse click ketika game flappycopter dimainkan di computer. Jika dimainkan di Android gunakan touch.

Bug Text HUD Score

Jika dimainkan di android tulisan score pada saat game flappycopter dimainkan masih terlalu kecil. Maka dari itu ubah kode GenerateObs.

1. Tambahkan variabel baru dibawah int score = 0;

```
GUIStyle guiStyle = new GUIStyle();
```

2. Ganti kode:

```
GUILayout.Label ("Score: " + score.ToString());
```

Menjadi:

```
guiStyle.fontSize = 40;  
GUI.Label(new Rect(0, 0, 300, 50), "Score: " +  
score.ToString(), guiStyle);
```

Bug Restart

Jika kalah akan lebih baik jika kembali ke menu dari pada harus restart game. Maka dari itu rubah kode PlayerControl, pada method Die() dari:

```
SceneManager.LoadScene ("Main");
```

Menjadi:

```
SceneManager.LoadScene ("Menu");
```

Singleton Class

Seringkali, sistem hanya perlu membuat satu instance dari kelas, dan instance yang akan diakses di seluruh program. Contohnya, objek yang dibutuhkan untuk logging, komunikasi, akses database, dll. Jadi, jika sistem hanya perlu satu instance dari kelas, dan instance yang perlu dapat diakses di banyak bagian yang berbeda dari sistem.

Sebuah Singleton adalah kombinasi dari dua sifat penting:

- Pastikan kelas hanya memiliki satu instance.
- Memberikan titik global akses melalui instance tersebut.

Ada banyak cara untuk membuat kelas singleton, tetapi berikut adalah contoh kelas singleton yang paling sederhana. Lakukan praktikum ini hanya untuk uji coba class singleton.

PRAKTIKUM:

1. Buat Kelas baru dengan nama SingleTonSample pada scene Menu. Pastikan seluruh isi default seperti :MonoBehavior, Init(), Update() sudah dihapus atau kelas baru tersebut **kosong** kemudian definisikan konstruktor method secara private.

```
private SingleTonSample()  
{}
```

2. Buat variable private yang digunakan sebagai objek Instance kelas itu sendiri.

```
private volatile static SingleTonSample singleTonObject;
```


3. Buat fungsi/method statis yang berfungsi untuk men-instance suatu class itu sendiri dengan mengembalikan nilai instance itu sendiri.

```
public static SingletonSample InstanceCreation()
{
    private static object lockingObject = new object();
    if(singleTonObject == null)
    {
        lock (lockingObject)
        {
            if(singleTonObject == null)
            {
                singleTonObject = new SingletonSample();
            }
        }
    }
    return singleTonObject;
}
```

4. Buat fungsi public untuk mencoba menampilkan tulisan “Kelas Singelton!”

```
public void DisplayMessage()
{
    Debug.Log("Kelas Singelton!");
}
```

5. Buat Kelas baru dengan nama TestSample dan Game object kosong, kemudian masukkan TestSample kedalam komponen dari Game object kosong yang dibuat tadi.
6. Panggil SingletonSample didalam fungsi Start() kelas TestSample.

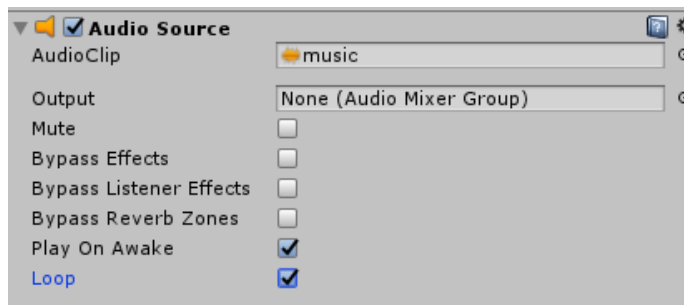
```
void Start()
{
    SingletonSample singleton = SingletonSample.InstanceCreation();
    singleton.DisplayMessage();
}
```

Menambahkan Musik

Penambahan music didalam game dengan unity cukuplah rumit. Pada dasarnya penambahan sound music dapat dilakukan di Main Camera dengan menambahkan komponen “Audio Source”. Kemudian menambahkan “AudioClip” didalam property komponen “Audio Source”. Tetapi, jika berpindah scene, maka suaranya akan hilang. Untuk itu digunakan singleton class.

PRAKTIKUM:

1. Buat folder Sounds pada project window dan masukkan music.mp3 didalam folder tersebut.
2. Pada scene Menu, buat game object kosong dan rename dengan nama codeMusic.
3. Lihat Inspector codeMusic dan tambahkan komponen Audio Source.
4. Tarik music dari project window kedalam Audio Clip dan centang Loop, seperti gambar berikut:



5. Buat script baru pada codeMusic dengan nama MusicManager, dengan isi seperti berikut: (perhatikan bahwa MusicManager merupakan SingletonClass, dari pada menggunakan method konstruktor untuk instance object atau method Start(), digunakan method Awake() karena akan dipanggil lebih awal daripada method Start())

```

1 using UnityEngine;
2
3 public class MusicManager : MonoBehaviour
4 {
5
6     private static MusicManager instance = null;
7
8     public MusicManager Instance()
9     {
10         return instance;
11     }
12
13     public void Awake()
14     {
15         if (instance != null && instance != this)
16         {
17             Destroy(this.gameObject);
18             return;
19         }
20
21         else
22         {
23             instance = this;
24         }
25
26         DontDestroyOnLoad(this.gameObject);
27     }
28
29 }
30

```

DontDestroyOnLoad digunakan untuk mencegah object codeMusic menjadi hilang saat berpindah scene.

6. Jalankan unity dan lihat apa yang terjadi.
7. Build aplikasi di Android. Dan Akhirnya full Flappycopter game sudah jadi.

Latihan (untuk pertemuan 6)

PRAKTIKUM:

1. Tambahkan 2 button baru dengan nama ButtonCredits dan ButtonHelp di dalam scene menu.
2. Buat dua scene baru dengan nama credits dan help.

3. Scene credits berisi elemen UI Text siapa saja yang membuat game dan satu UI Button untuk kembali ke menu.
4. Scene help berisi elemen UI Text cara bermain game dan satu UI Button untuk kembali ke menu.
5. Dengan mekanisme yang sama dengan bab GUI main menu. Tambahkan method di GameManager,
 OnCredits() → untuk membuka menu credits
 OnHelp() → untuk membuka menu help
 OnBack() → untuk kembali ke main menu.
6. Jalankan Unity, kemudian build pada Android.

KUMPULKAN:

Screenshoot Game tab saat di menu scene dan simpan dengan nama sc_056_03.jpg.

Screenshoot Game tab saat di credits scene dan simpan dengan nama sc_056_04.jpg.

Screenshoot Game tab saat di help scene dan simpan dengan nama sc_056_05.jpg.

REFERENSI

<http://answers.unity3d.com/questions/137143/how-to-set-the-font-size-in-guilabel.html>

<https://docs.unity3d.com/Manual/UISystem.html>

<https://www.codeproject.com/Tips/219559/Simple-Singleton-Pattern-in-Csharp>

Mekanisme Pengumpulan Praktikum Mingguan

Untuk setiap mahasiswa: Buat 1 folder beri nama NIM lengkap. Masukkan file screenshot sc_056_01.jpg, sc_056_02.jpg, sc_056_03.jpg, sc_056_04.jpg, dan sc_056_05.jpg. Kemudian rar atau zip folder NIM kamu tadi dan kirimkan ke ketua kelas.

Contoh: A11200904997.rar

Untuk Ketua Kelas: Slapkan folder dengan nama Kode Kelas “_056” yang berisi kumpulan praktikum setiap mahasiswa. Lalu buat rar atau zip dan kirimkan ke email dosen.

Contoh: A114401_056.rar