

# UNITY GÜNLÜĞÜ

## UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON

GÜN 1

# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

by yselim

## Unity-Vuforia ile Augmented Reality Virtual Button Uygulaması

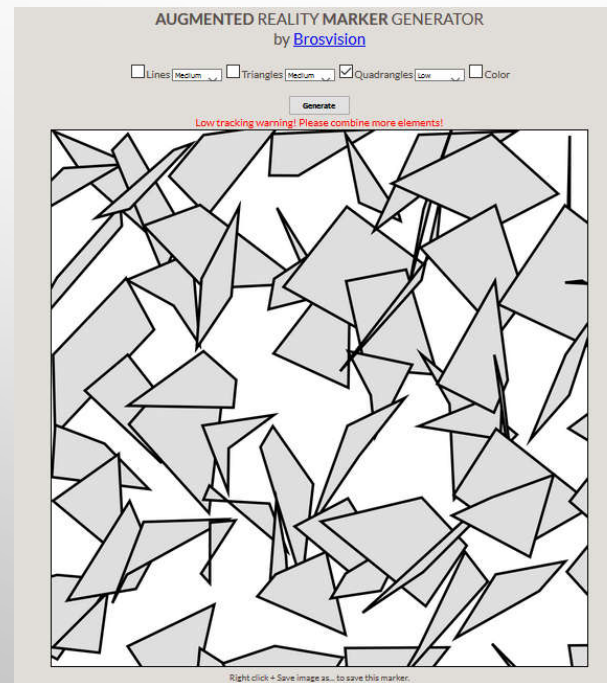
- 1.Unity-Vuforia AR için bir marker oluşturmak gereklidir. Bu marker'ı kendinizde tasarlayabilirsiniz yada online generator kullanabilirsiniz.
- 2.Bu yollardan biri ile marker'ı oluşturup resim dosyası ve çıktı olarak hazırlayınız.
- 3.Online marker oluşturmak için;

<http://www.brosvision.com/ar-marker-generator/>

# UNITY-VUFORIA-AUGMENTED REALITY

Unity-Vuforia ile Augmented Reality Uygulaması  
Online marker oluřturmak için;

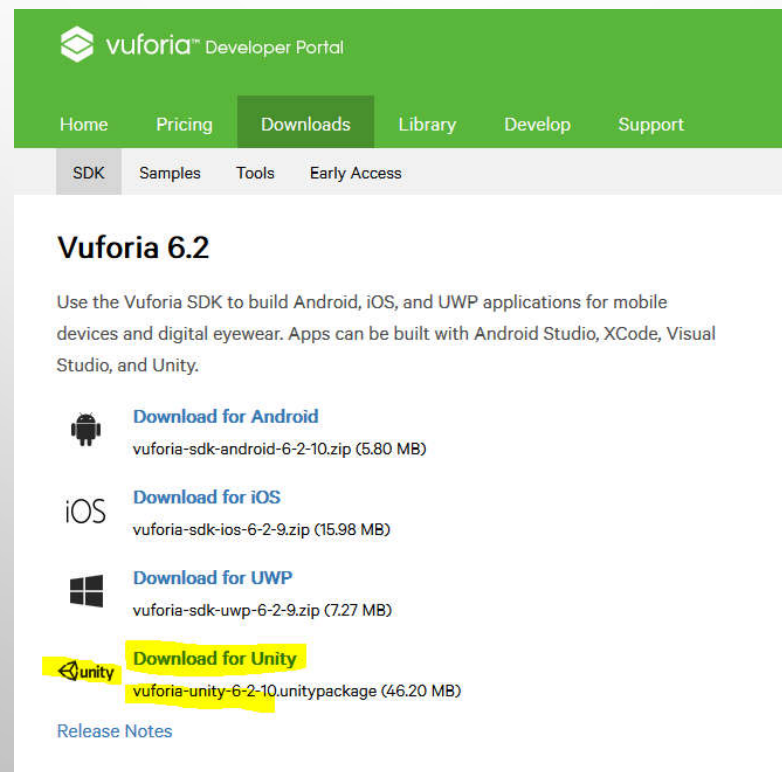
<http://www.brosvision.com/ar-marker-generator/>



# UNITY-VUFORIA-AUGMENTED REALITY

4. Unity3D kurulur, Unity3D için gerekli Vuforia unity paketi aşağıdaki adresten 'Download for Unity' linkinden indirilir.(Artık INTEGRATED!)

<https://developer.vuforia.com/downloads>







**vuforia™ Developer Portal**

Home Pricing Downloads Library Develop Support

SDK Samples Tools Early Access

## Vuforia 6.2

Use the Vuforia SDK to build Android, iOS, and UWP applications for mobile devices and digital eyewear. Apps can be built with Android Studio, XCode, Visual Studio, and Unity.

-  **Download for Android**  
vuforia-sdk-android-6-2-10.zip (5.80 MB)
-  **Download for iOS**  
vuforia-sdk-ios-6-2-9.zip (15.98 MB)
-  **Download for UWP**  
vuforia-sdk-uwp-6-2-9.zip (7.27 MB)
-  **Download for Unity**  
vuforia-unity-6-2-10.unzippackage (46.20 MB)

[Release Notes](#)

# UNITY-VUFORIA-AUGMENTED REALITY

5. Vuforia Develop sayfasında 'Develop' başlığı seçilerek uygulama için yeni bir lisans anahtarı(license key) oluşturulur.
6. 'Add License' key ile App için bir lisans anahtarı oluşturulur.
7. Proje türü olarak Developer/Consumer/Enterprise biri seçilip projeye isim verilir. Key oluşturulur. Bu anahtar Unity içinde uygulama çalıştırılırken kullanılacak key olacaktır.

License Manager

Target Manager

## License Manager

Create a license key for your application.

Add License Key

Name

arDeneme

zombiAP



zombiAP

Edit Name

Delete License Key

License Key

Usage

Please copy the license key below into your app

```
AezB3rb/////AAAAG73AHlHNkaSsITzdEspfVMW9Rub7y5yx6pV  
eSEr4sspe3JU8JIiFRSvCE6tjXwWPZVHNC51K0RtHtFeloanYb2m  
IcEppwcv/pMs173n51pOGej+2g5gaVEjeMjL4vL87oJtSiJdXWfe  
A0DLbmeF9C/ksU/ma5uFuX0u4qGs6YLM5W1A2/kXYG5GHbNONjiT  
73bFW0vUp2p9Yadr3tYQDeNwr/oJ24MD98souXzv3XFFjKEk4Gjd  
+NFhuCT9mohd4uV+n6v8Alc/p9BKvk6Wx6j2jfgbtVs+eddr1Qjm  
Tv5FtXDYnWdgk4ouTB2nOY+4YisqSVgwNquLj9x6BWz3CvSdM21L  
Rcq8Hh8A+ipG2Kgz
```

Type: Develop

Status: Active

Created: Feb 17, 2017 15:06

# UNITY-VUFORIA-AUGMENTED REALITY

8. Aynı ekran 'target manager' tıklanarak yeni bir Database eklenir. Burada kamera tarafından görülecek olan imajın kendisi belirlenecektir.

9. Add Target ile daha önceden belirlenmiş olan Marker'ı burada sisteme yüklüyoruz.

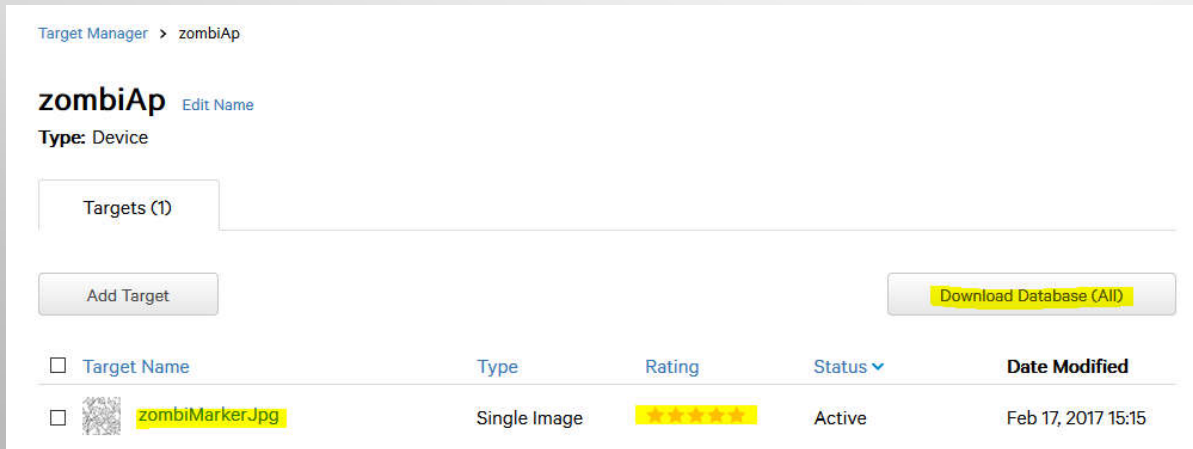
Target Manager			
Use the Target Manager to create and manage databases and targets.			
<button>Add Database</button>			
Database	Type	Targets	Date Modified
arDeneme	Device	1	Sep 11, 2015 01:21
zombiAp	Device	1	Feb 17, 2017 15:15

# UNITY-VUFORIA-AUGMENTED REALITY

10. Single Image ile dosya Browse edilir ve seçilip eklenir. Eğer uygun bir resim dosyası değilse hata verecektir. 8bit gray yada 24bit RGB türünde JPG yada PNG dosya kullanılabilir.

11. Eğer target uygun ise 'Rating' kolonundaki yıldızlar ne kadar uygun olduğunu işaret eder.

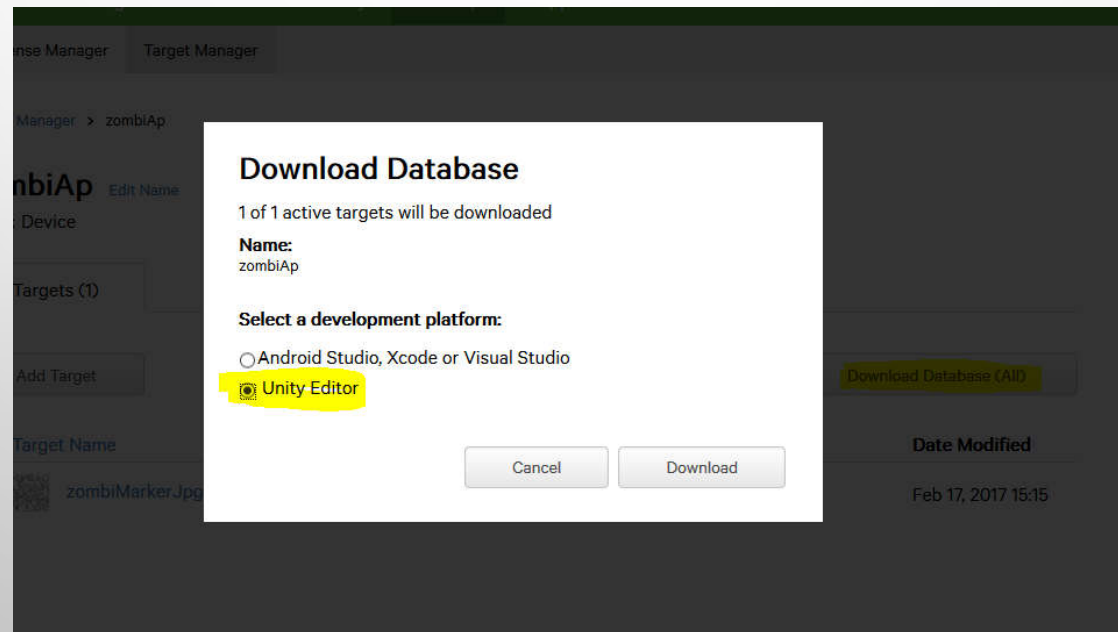
12. Target yüklendikten sonra marker resmini tıklayıp resim görülebilir. Burada Hide/Show Features diyerek kontorlar görülebilir.





# UNITY-VUFORIA-AUGMENTED REALITY

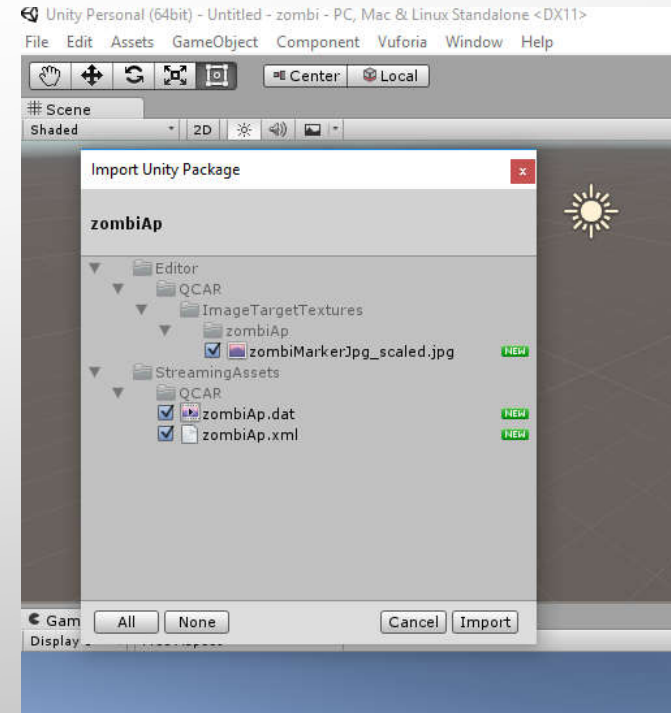
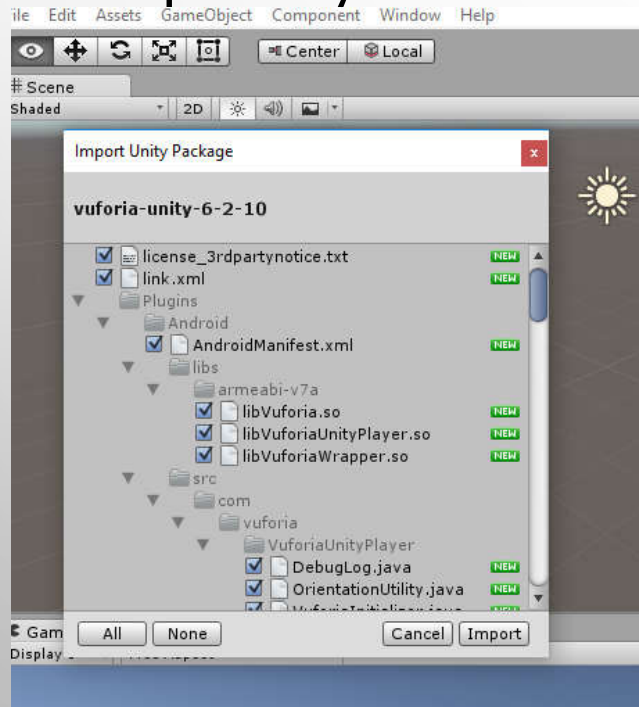
13- Download Database All ile Unity editörü içine paket çekilir. Elimizde vuforia eklentisi ve oluşturulan bu marker için database mevcut. Bundan sonra unity içine bunları import edeceğiz





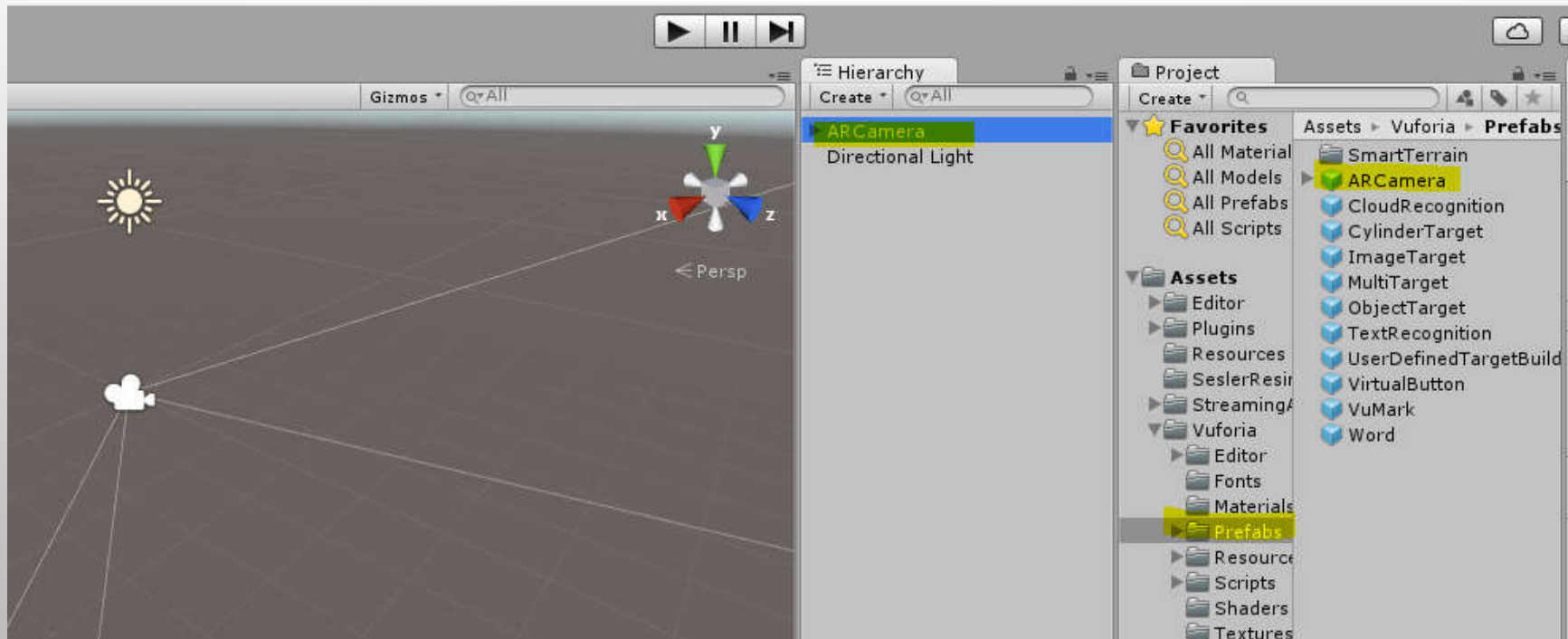
# UNITY-VUFORIA-AUGMENTED REALITY

- Unity editörü içinde Assets / Import Package / Custom Package **yada**
- Project paneli içindeki Assets klasörüne sürükleyip bırakarak bu iki paket yüklenir.



# UNITY-VUFORIA-AUGMENTED REALITY

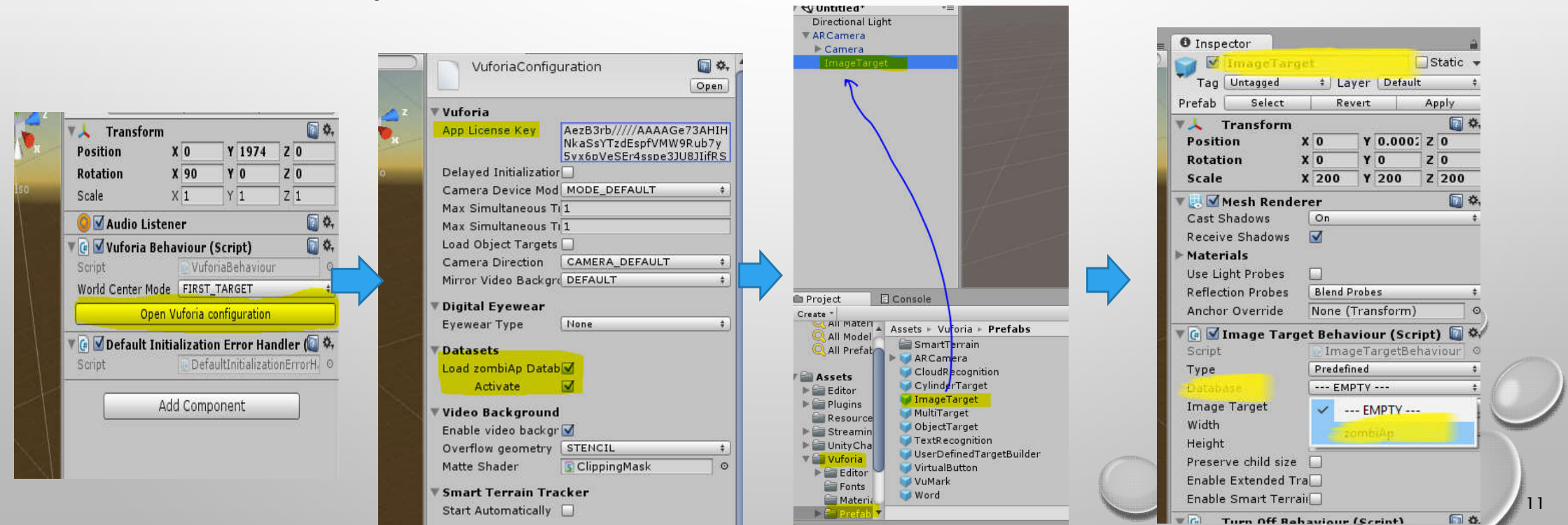
Yeni sahnede main camera silinerek yerine AR camera eklenir. AR camera nesnesi vuforia eklentisi içindeki **Prefabs** içinden ARCamera hiyeraşik listeye eklenir.



# UNITY-VUFORIA-AUGMENTED REALITY

ArCamera seçilerek 'open Vuforia Configuration' butonu tıklanarak daha önceden hazırlanan Target Manager'deki dataset yüklenir ve aktive edilir.

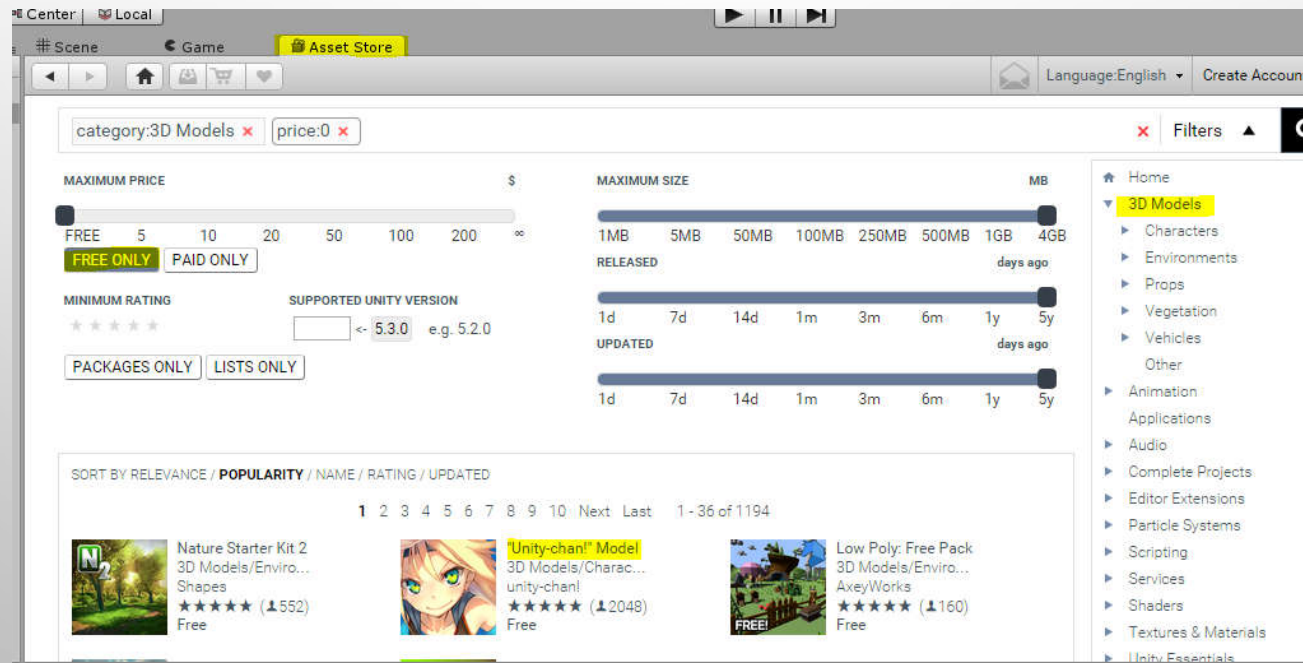
Ayrıca yine License Manager'daki Key editörde «App Licence Key» kısmına yapıştırılır. Ardından ImageTarget seçilerek Database kısmından ilgili 'Dataset' yüklenir. Dataset yüklendiğinde seçilen marker resmi düzlemde gözükür.



# UNITY-VUFORIA-AUGMENTED REALITY

Artık ArCamera ve ImageTarget hazır. Assets store'dan indirilecek 3D model sahneye eklenecek.

Window/Asset store yada Asset Store kulakçısından 3D models kısmından ücretsiz olan modellerden biri Unity'ye import edilir.



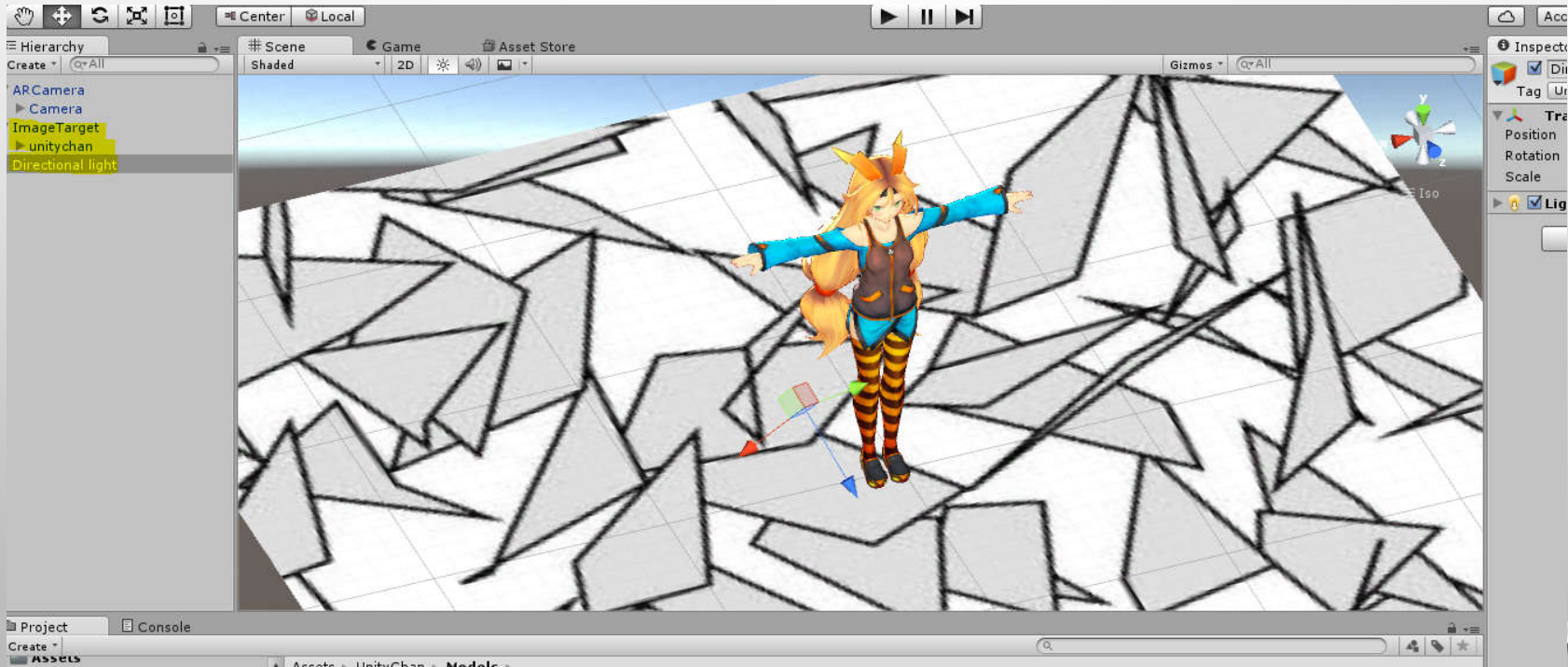


# UNITY-VUFORIA-AUGMENTED REALITY

Eklenen 3D modelin klasörlerinden 'models' klasöründeki 3D model hiyerarşi listesine eklenir ve *ImageTarget*'ın *child*'ı olarak atanır.

Sahneyi ışıklandırmak için hiyerarşi panelinde Sağ buton/Light/Directional Light eklenir.

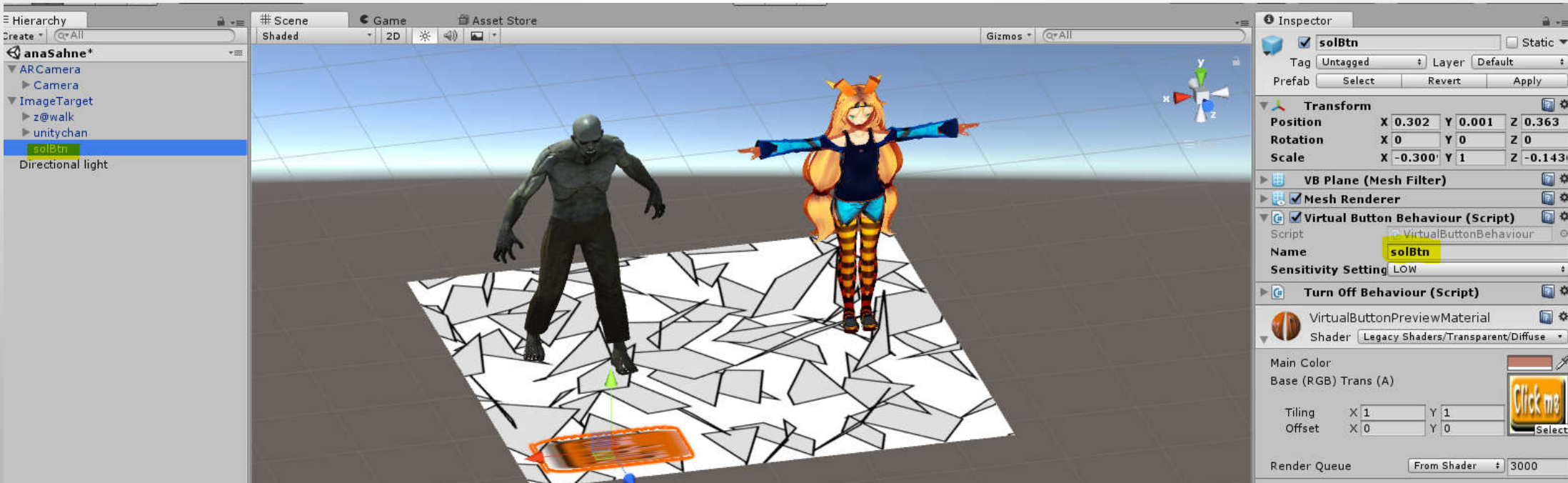
Sahne çalıştırılır, kamera'ya marker tutulduğunda 3d karakter görülür.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

by yselim

Sahneye yeni bir buton eklemek için Vuforia paketindeki Prefabs içindeki VirtualButton nesnesi eklenir. Bu butona görev atamak için Script kullanılır. Virtual buton eklenir ve ImageTarget'ın child'ı olarak atanır. ImageTarget seçilip yeni bir Script (C# yada javascript) eklenir. Virtual buton çalışma esnasında ekranda gözükmez.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

Virtual button için aşağıdaki C# scripti yazılır. Burada Vuforia alanadı yüklendikten sonra *IVirtualButtonEventHandler* sınıfı kullanılır. Start() fonksiyonu ile sol ve sağ butonlar tespit edilir, ilgili sanal buton üzerine gelindiğinde *OnButtonPressed()* ve *OnButtonReleased()* metodları ile görevler yazılır.

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

public class virtualButton : MonoBehaviour,
Vuforia.IVirtualButtonEventHandler {
    private GameObject vbObject;
    // Use this for initialization
    void Start () {
        vbObject = GameObject.Find("solBtn");
        //Event handler ile register etmek için

        vbObject.GetComponent<VirtualButtonBehaviour>
().RegisterEventHandler(this);
    }
}
```

```
public void OnButtonPressed(VirtualButtonAbstractBehaviour vb)
{
    Debug.Log("buton basıldı...!!");
    //throw new NotImplementedException();
}

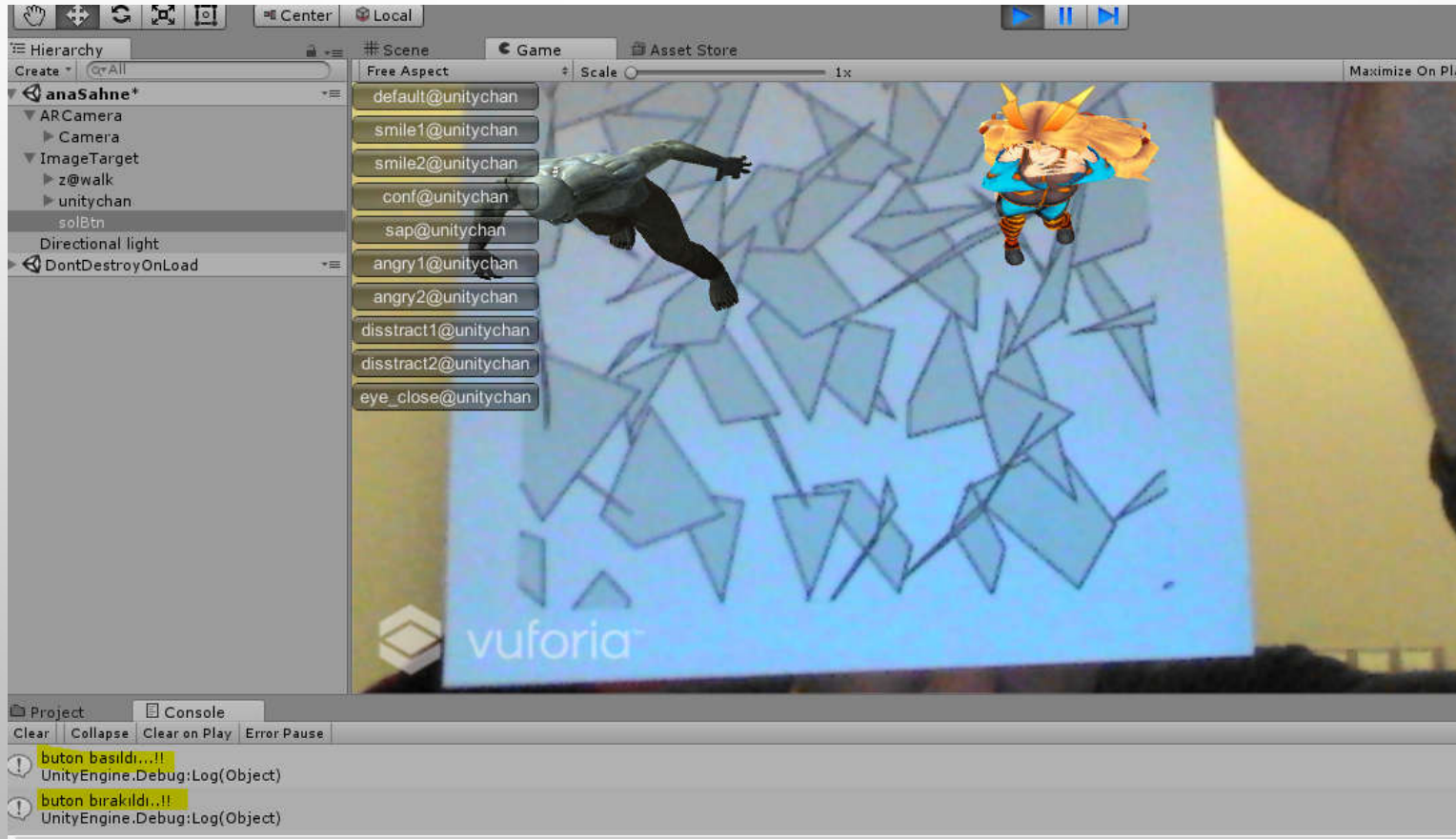
public void OnButtonReleased(VirtualButtonAbstractBehaviour vb)
{
    Debug.Log("buton bırakıldı...!!");
    //throw new NotImplementedException();
}
}
```



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

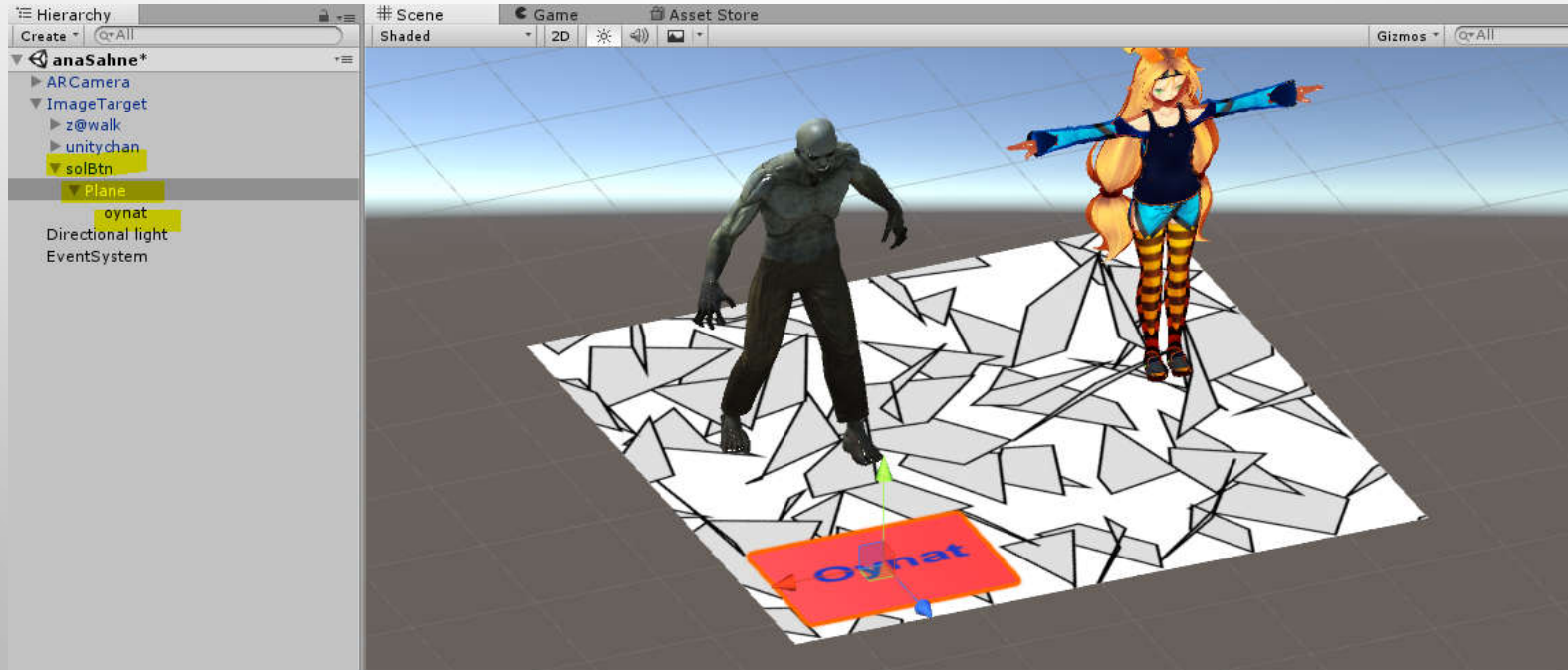
by yselim

Sanal buton üzerine gelindiğinde(Pressed) yada buton üzerinden gidildiğine (Released) Console ekranında ilgili mesajlar yazdırılır.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

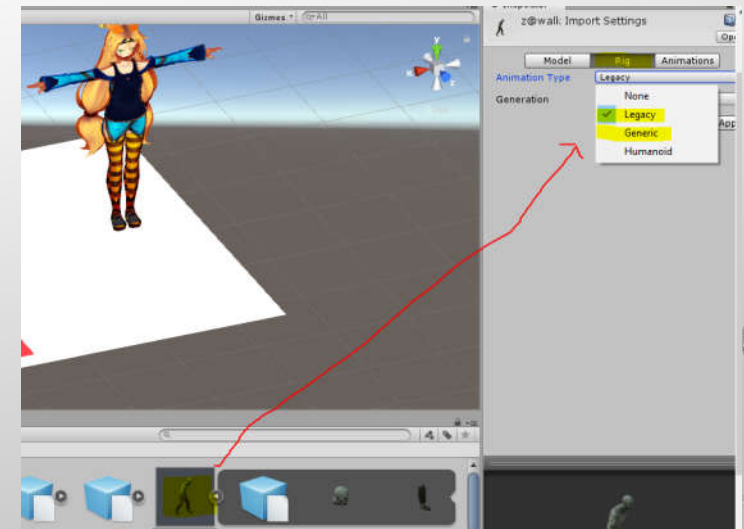
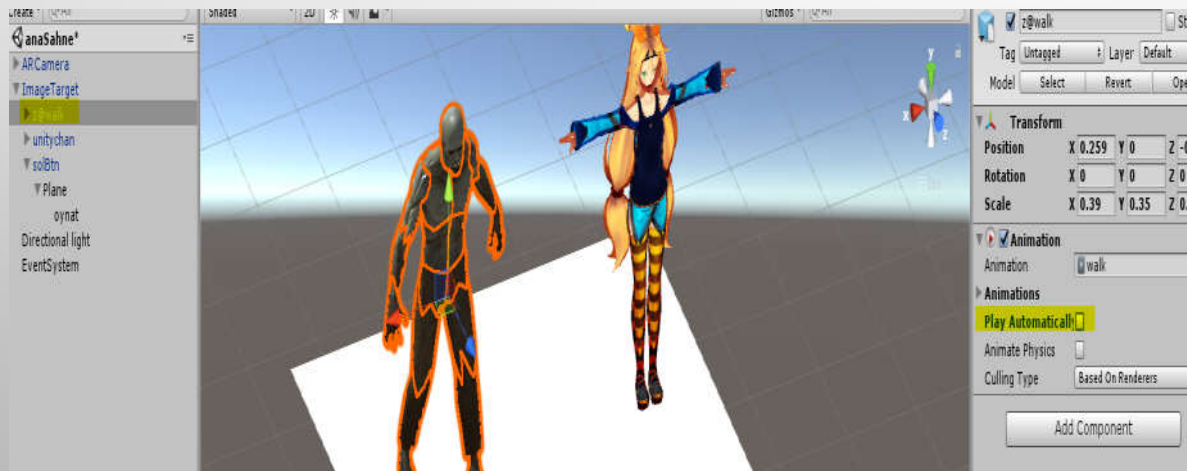
Virtual button çalışma esnasında gözükmediği için Virtual butonun olduğu lokasyona bir küp eklenebilir. Böylece butonun bulunduğu yer kullanıcı için belirlenmiş olur. Sanal butonun child'ı olarak bir plane eklenir, plane üzerine(child'ı) de bir 3D text eklenir ve boyutları sanal buton büyüklüğüne getirilir.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

Modeli oynatmak için buton tıklandığında çalıştırılacak C# kodunu yazıp çalıştıralım. Öncelikle animasyon otomatik oynanması için 'Play Automatically' iptal edilmeli. 'Oynat' butonu ile hareket ettirilecektir.

Kodu yazıp çalıştırdığınızda eğer animasyon çalışmıyorsa modelin animasyon ayarlarından 'Rig' başlığında 'Generic' seçeneğini 'Legacy' olarak değiştirilebilir.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

by yselim

Modeli oynatmak *ImageTarget* için buton tıklandığında çalıştırılacak C# kodunu yazıp çalıştıralım. (Modelin Wrap modu : loop olmalı)

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

public class virtualButton : MonoBehaviour,
Vuforia.IVirtualButtonEventHandler {
    private GameObject vbObject;
    private GameObject zombie;
    // Use this for initialization
    void Start () {
        vbObject = GameObject.Find("solBtn");
        zombie = GameObject.Find("z@walk"); //walk
        animation'ı harekete geçirmek için Play metodu
        kullanılır.
        //Event handler ile register etmek için
        vbObject.GetComponent<VirtualButtonBehaviour>
        ().RegisterEventHandler(this);
    }
}
```

```
public void OnButtonPressed(VirtualButtonAbstractBehaviour vb)
{
    Debug.Log("buton basıldı...!!");
    //throw new NotImplementedException();
    zombie.GetComponent<Animation>().Play();
}

public void OnButtonReleased(VirtualButtonAbstractBehaviour
vb)
{
    Debug.Log("buton bırakıldı...!!");
    //throw new NotImplementedException();
    zombie.GetComponent<Animation>().Stop();
}
}
```



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

Çift virtual button kullanımı;

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

public class virtualButton : MonoBehaviour, IVirtualButtonEventHandler {
    private GameObject vbSolObject;
    private GameObject zombie;
    private GameObject unitychan;
    private GameObject vbSagObject;
    // Use this for initialization
    void Start () {
        vbSolObject = GameObject.Find("solBtn");
        vbSagObject = GameObject.Find("sagBtn");
        zombie = GameObject.Find("z@walk"); //walk animation'ı harekete
        geçirmek için Play metodu kullanılır.
        unitychan = GameObject.Find("Barbarian");
        //Event handler ile register etmek için
        vbSolObject.GetComponent<VirtualButtonBehaviour>
        ().RegisterEventHandler(this);

        vbSagObject.GetComponent<VirtualButtonBehaviour>().RegisterEventHandler(t
        his);
    }
}
```

```
public void OnButtonPressed(VirtualButtonAbstractBehaviour vb)
{
    if (vb.VirtualButtonName == "solBtn")
        zombie.GetComponent<Animation>().Play();
    if(vb.VirtualButtonName=="sagBtn")
        unitychan.GetComponent<Animation>().Play();
}

public void OnButtonReleased(VirtualButtonAbstractBehaviour vb)
{
    Debug.Log("buton bırakıldı..!!");
    if (vb.VirtualButtonName == "solBtn")
        zombie.GetComponent<Animation>().Stop();
    if (vb.VirtualButtonName == "sagBtn")
        unitychan.GetComponent<Animation>().Stop();
}
}
```

