

# UNITY GÜNLÜĞÜ

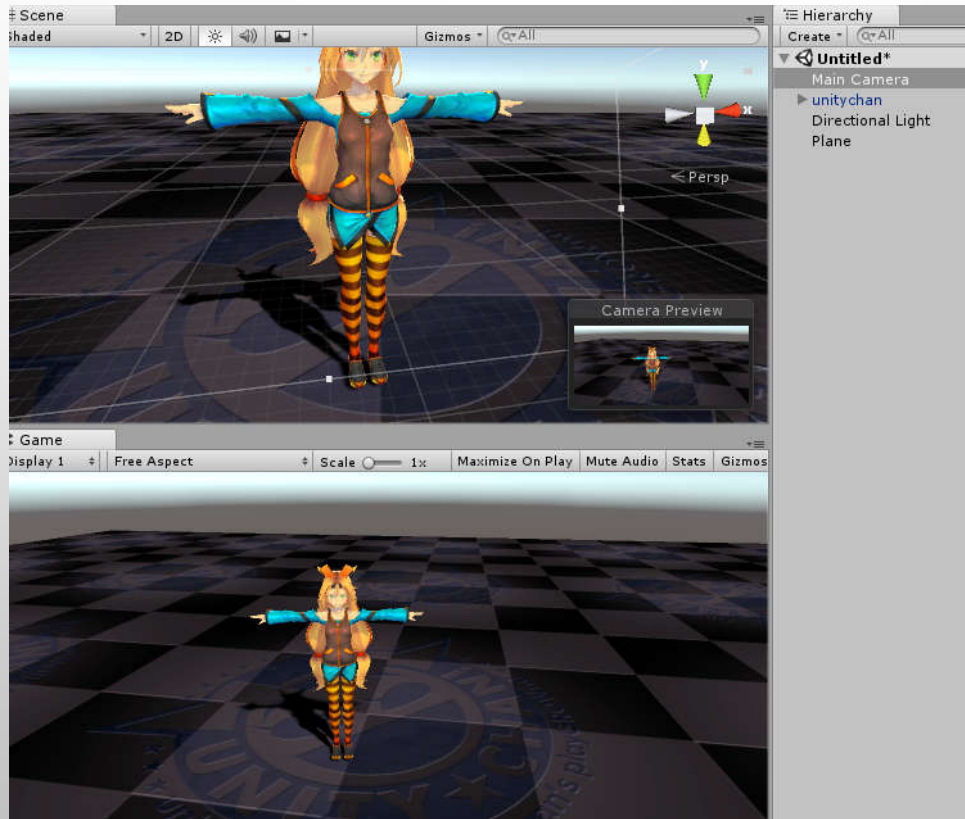
## UNITY-VUFORIA-AUGMENTED REALITY CHAN KONTROL- PART I

([HTTP://UNITY-CHAN.COM/DOWNLOAD/?LANG=EN](http://unity-chan.com/download/?lang=en))

GÜN 1

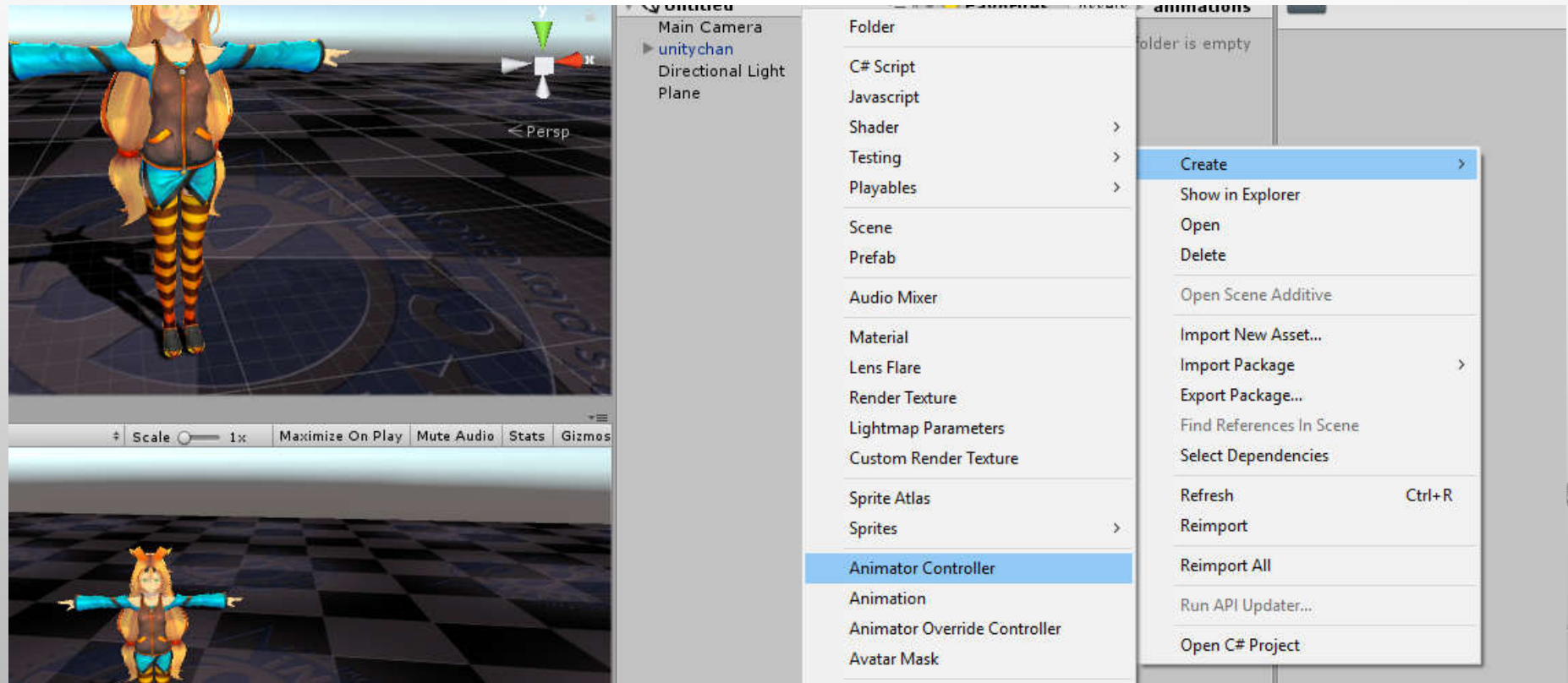
# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

- Chan kontrolü için , Chan paketindeki animasyonları script ile kontrol edeceğiz.
- Chan paketini projeye dahil edip UnityChan /Models /UnityChan sahneye eklenir.
- Yeni bir 3D plane eklenir ve *UnityChan/Stages/Textures* altından biri ile bezenir.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

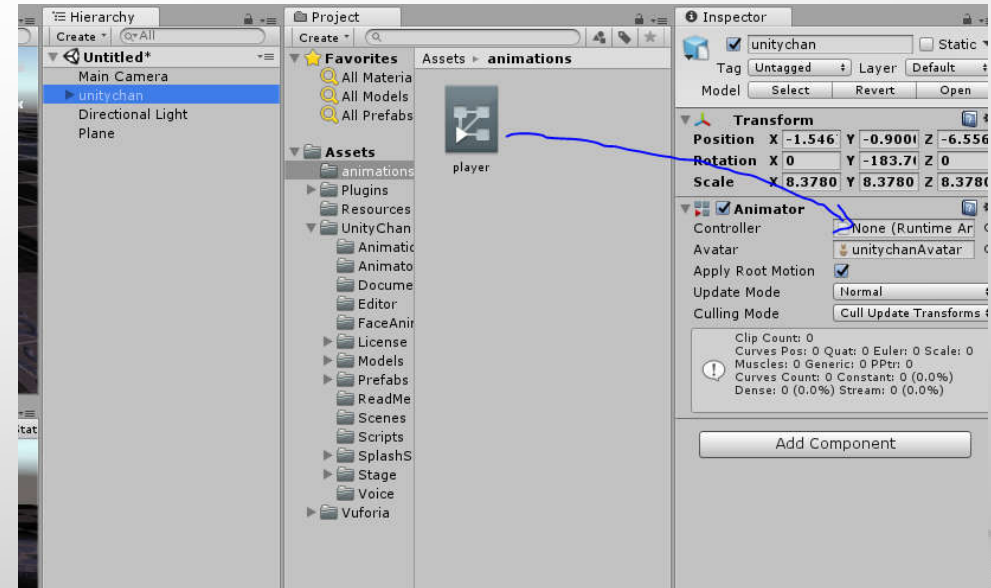
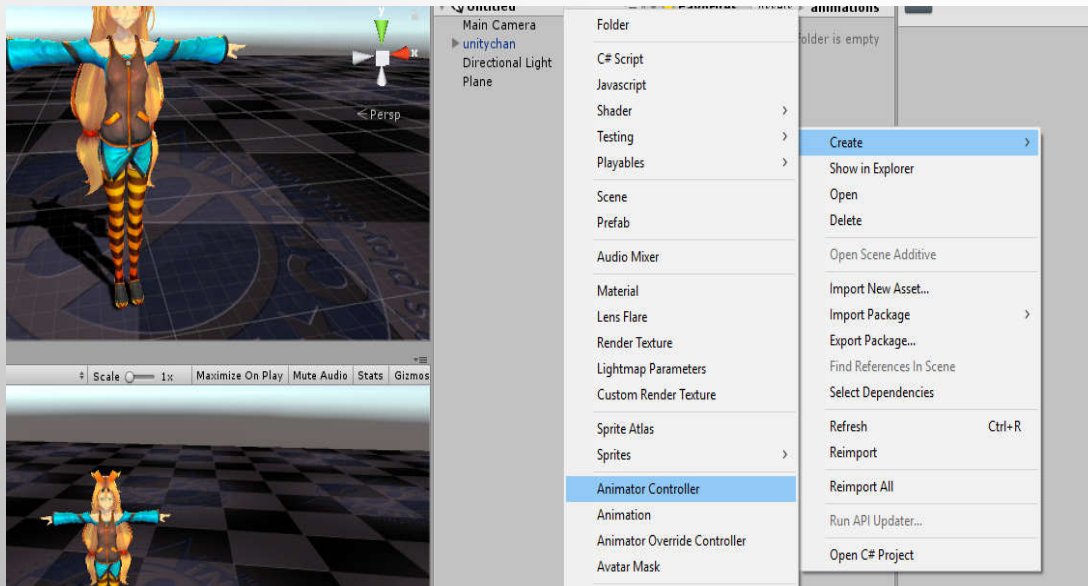
Chan kontrolü için , Chan paketindeki animasyonları script ile kontrol edeceğiz. Kontrol için önce Animator isimli bir folder oluşturuyoruz.Bunun içine *Create/Animator Controller* ekliyoruz.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

by yselim

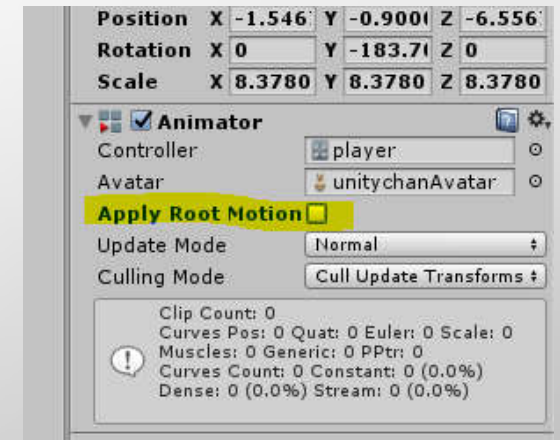
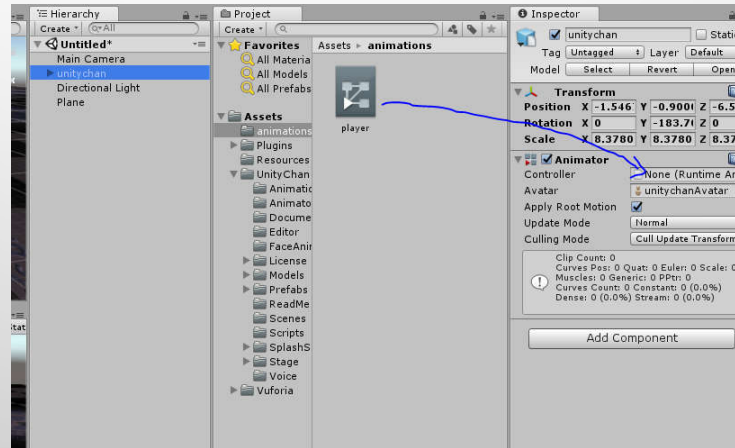
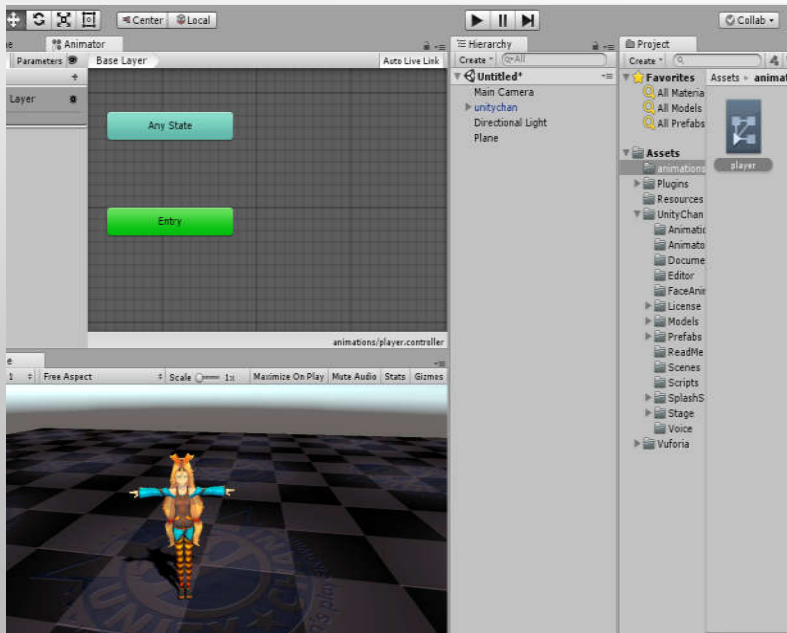
Kontrolleri Chan modelindeki *Animator/Controller*'a taşıyoruz.





# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

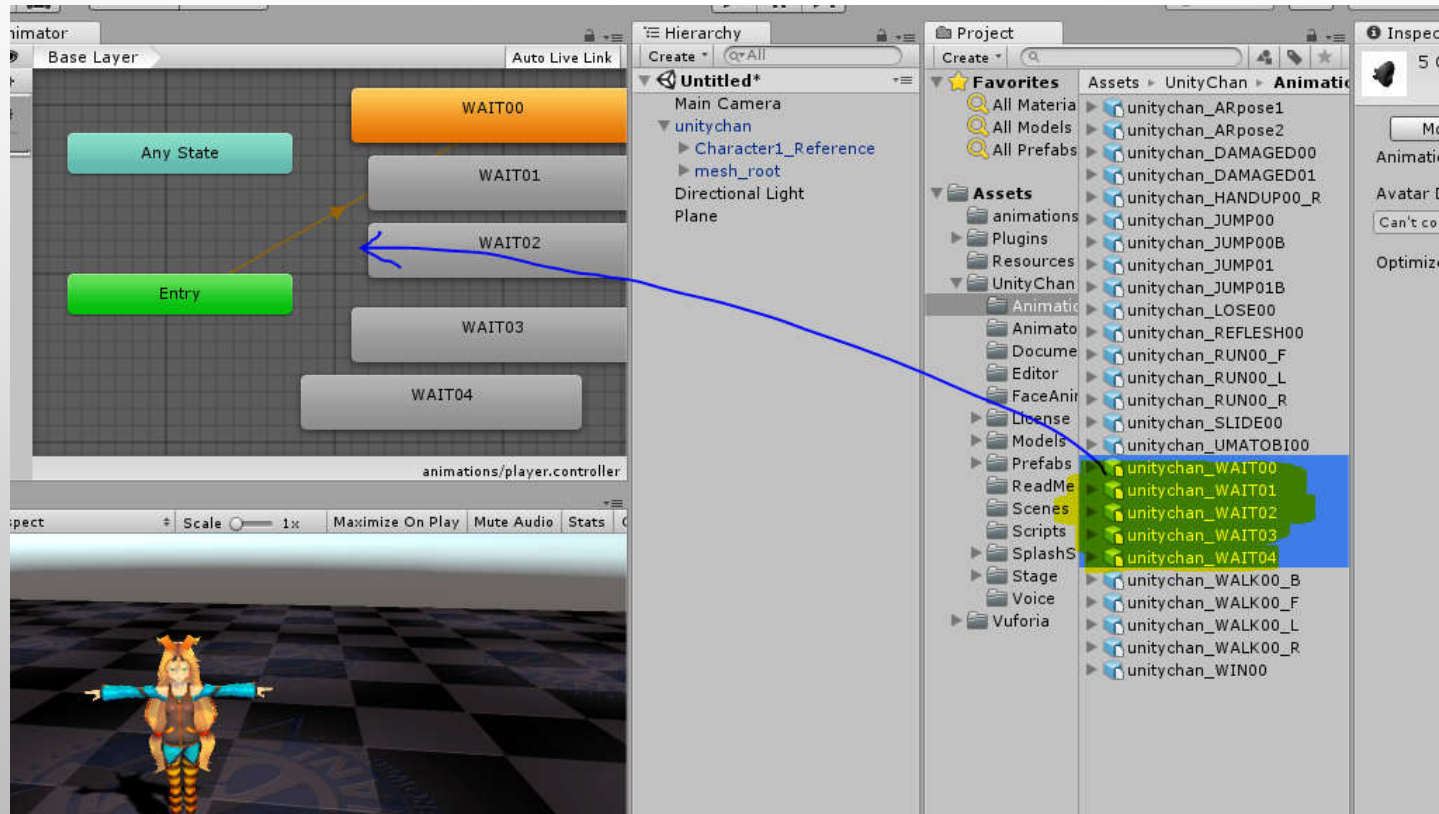
*Window /Animator* penceresi açılır. Buraya Chan içinde tanımlı olan animasyonları ekleyip yazacağımız script ile bunların kontrolünü sağlayacağız. Apply Root Motion iptal edilecek.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

by yselim

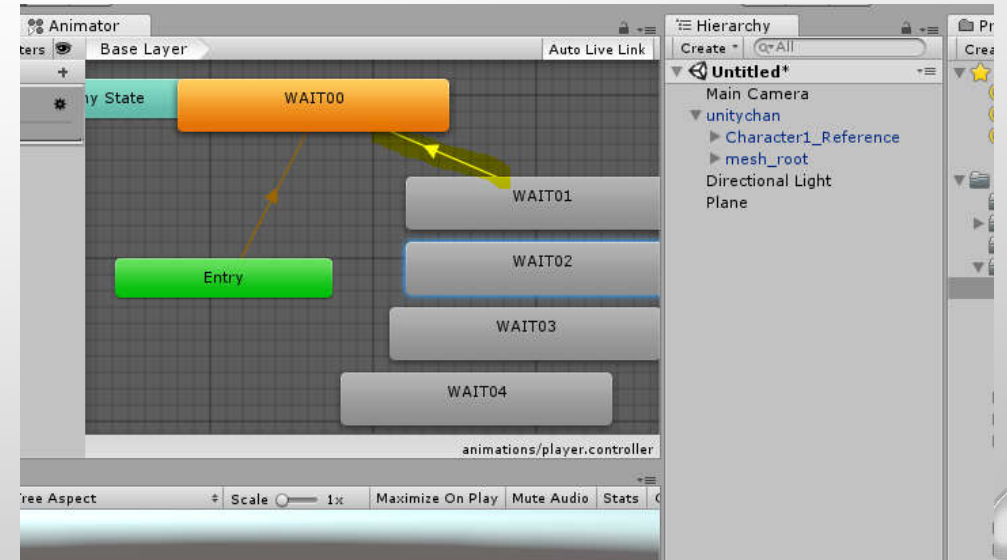
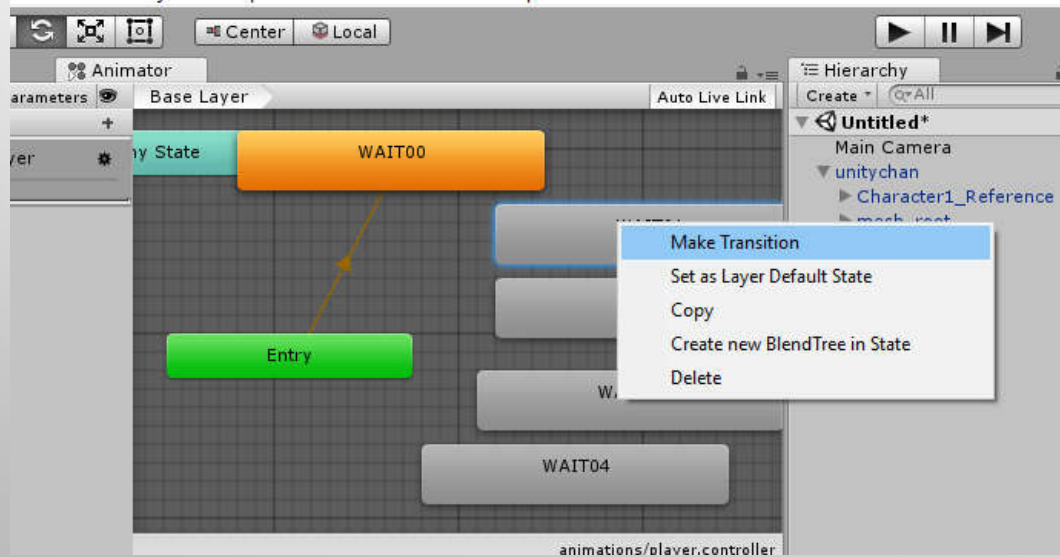
*UnityChan* paketi altındaki *Animations* başlığı altındaki animasyonlardan **WAIT** ile başlayanları *Animator* penceresine taşıyoruz. Turuncu olan default layer state.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

by yselim

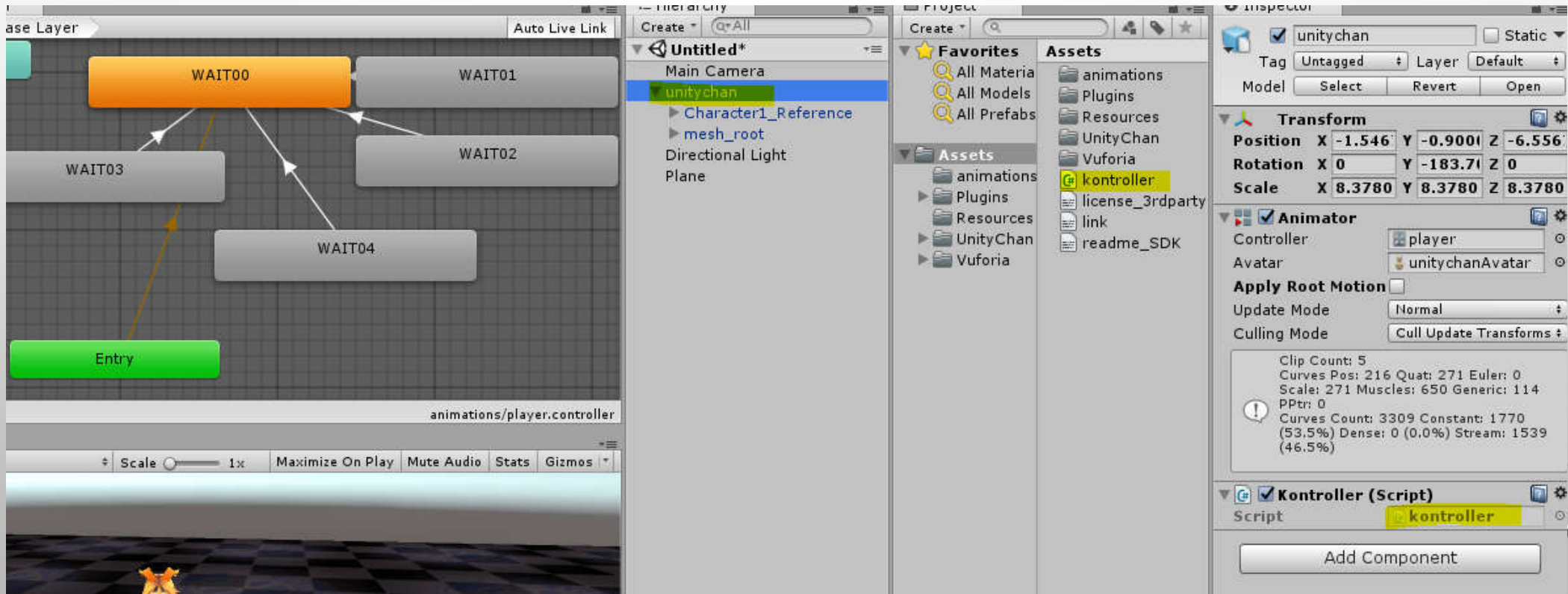
Animasyonlar arası transition için *Make Transition* kullanılır. Silmek için *Delete* kullanılır.



# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

by yselim

Tüm transitionları ayarladıktan sonra yeni bir Script ekliyoruz kontrol için. Scriptimizi *UnityChan*'e ekliyoruz.





# UNITY-VUFORIA-AUGMENTED REALITY VIRTUAL BUTTON UYGULAMASI

by yselim

Public Animator bileşeni ile sahnedeki Chan'ın Animator kontroller'i eşleştirilir. Basılan her tuş için bir animasyon oynatılır. *Play* komutundaki *-1 layer değeri* Base Layer değeri yani *Animator* penceresindeki katmandır. Son parametre ise animasyonun başlangıç zaman değeridir.

```
public class kontroller : MonoBehaviour {  
  
    public Animator anim;  
    // Use this for initialization  
    void Start () {  
  
        anim = GetComponent<Animator>();  
    }  
}
```

```
// Update is called once per frame  
void Update () {
```

```
    if(Input.GetKeyDown("1"))  
    {  
        anim.Play("WAIT01", -1, 0f);  
    }  
    if(Input.GetMouseButtonDown(0))  
    {  
        anim.Play("DAMAGED00", -1, 0f);  
    }  
    if (Input.GetMouseButtonDown(1))  
    {  
        anim.Play("DAMAGED01", -1, 0f);  
    }  
}
```