



**Ad:** Hilal

**Soyad:** Kaya

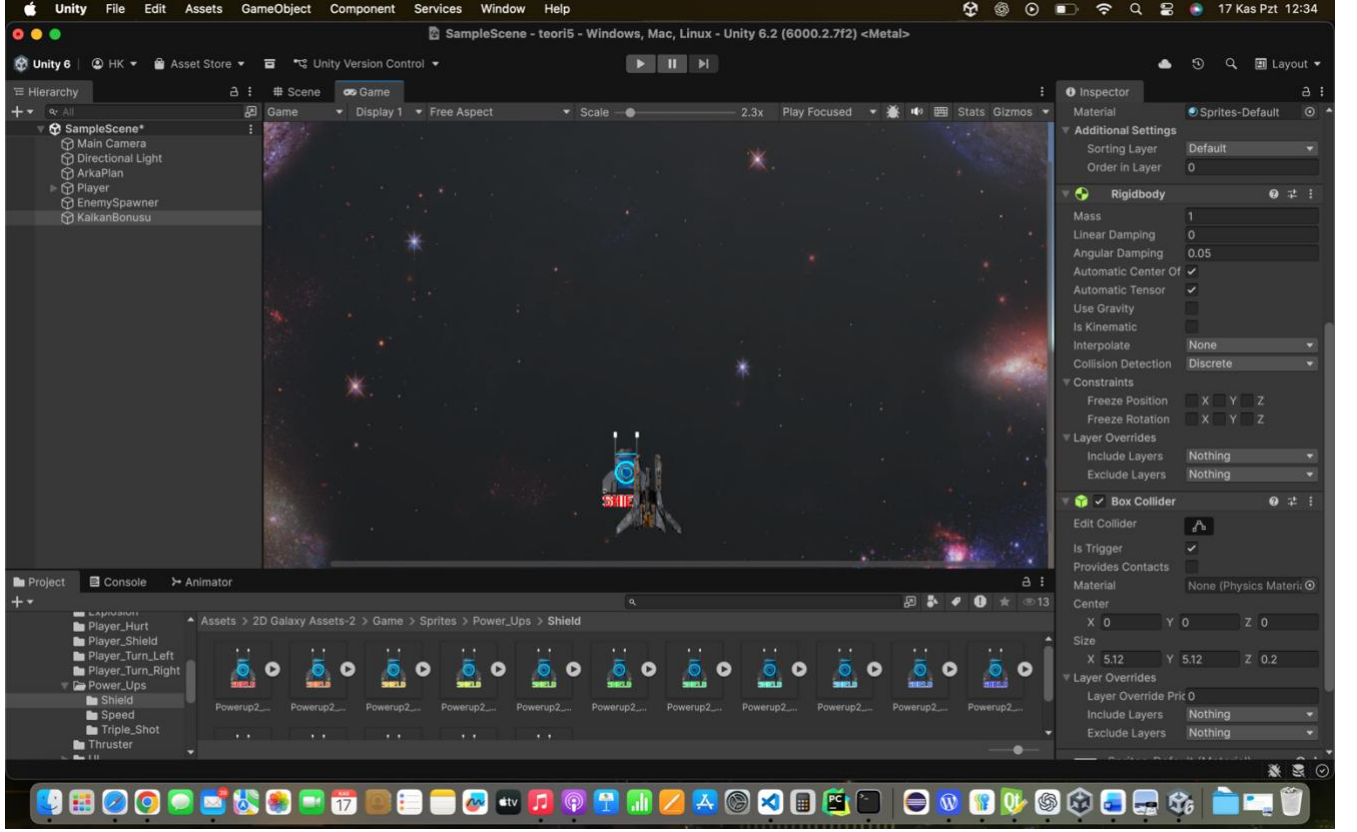
**Numara:** 22360859046

**Ders:** Oyun Programlama

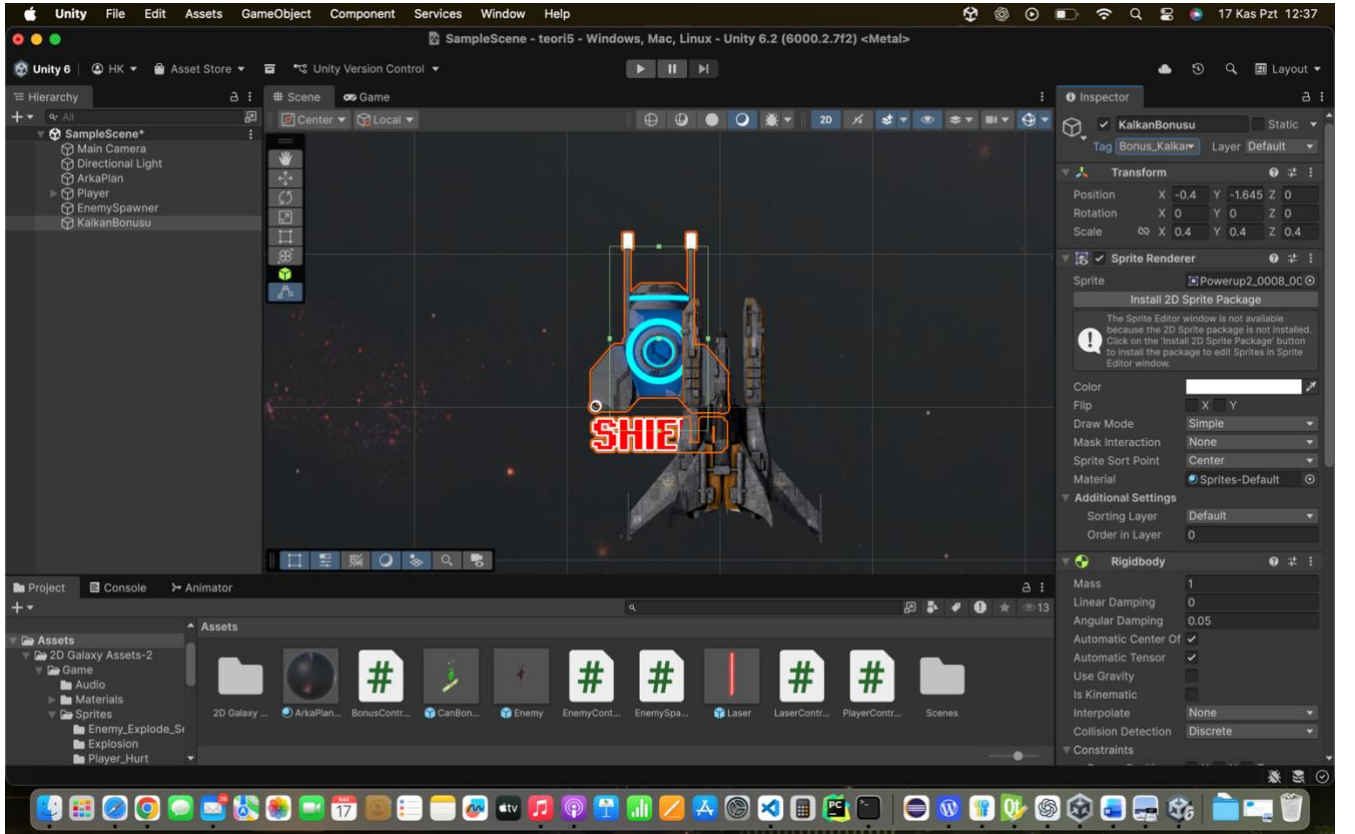
**Ödev adı:** Hafta 6

Geçen hafta bir uzay oyunu yapmıştık. Şimdi bu oyunu biraz daha geliştirip bazı ek özellikler ekleyeceğiz.

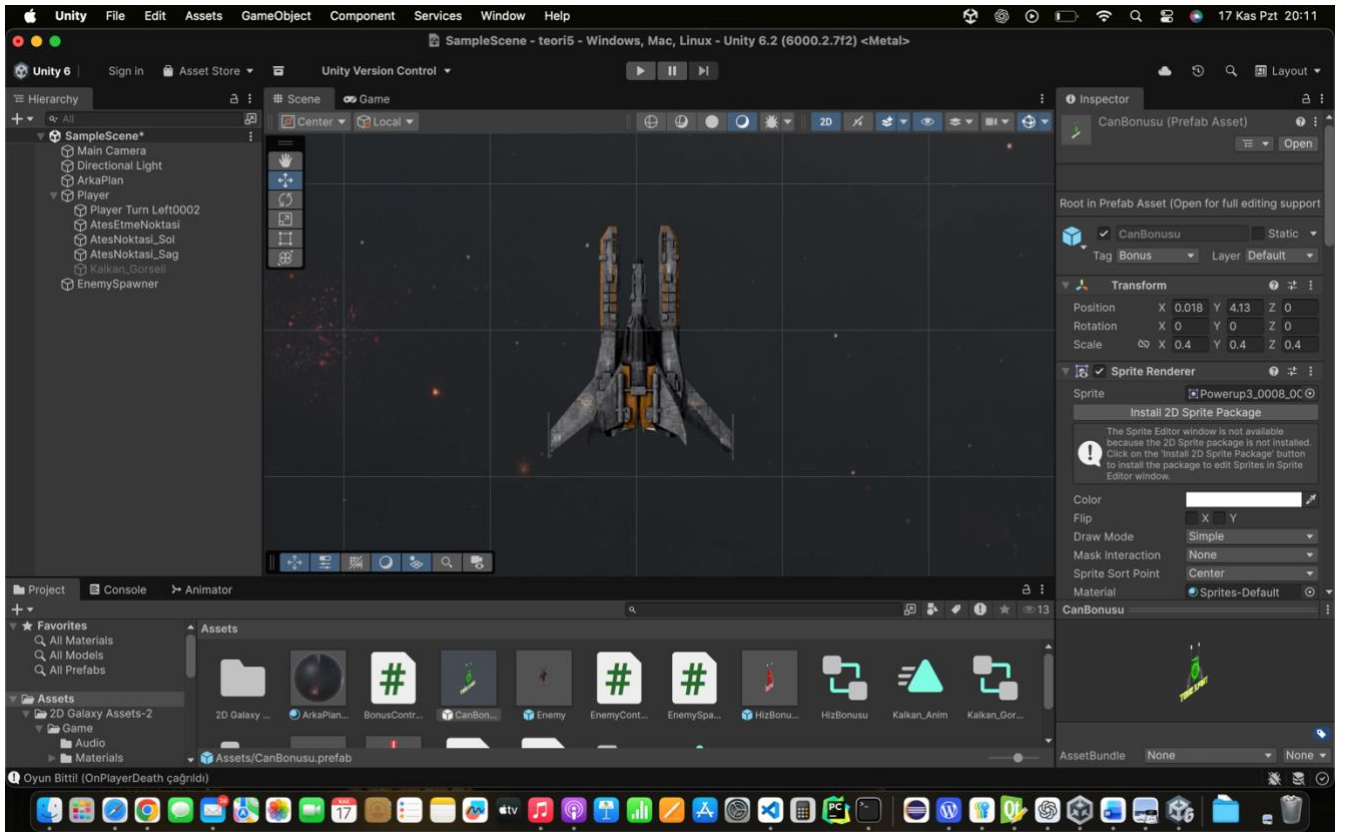
Bonus can nesneleri gibi yakalanabilen bonus kalkan nesneleri ve bonus hız nesnelerini ekleyeceğiz.



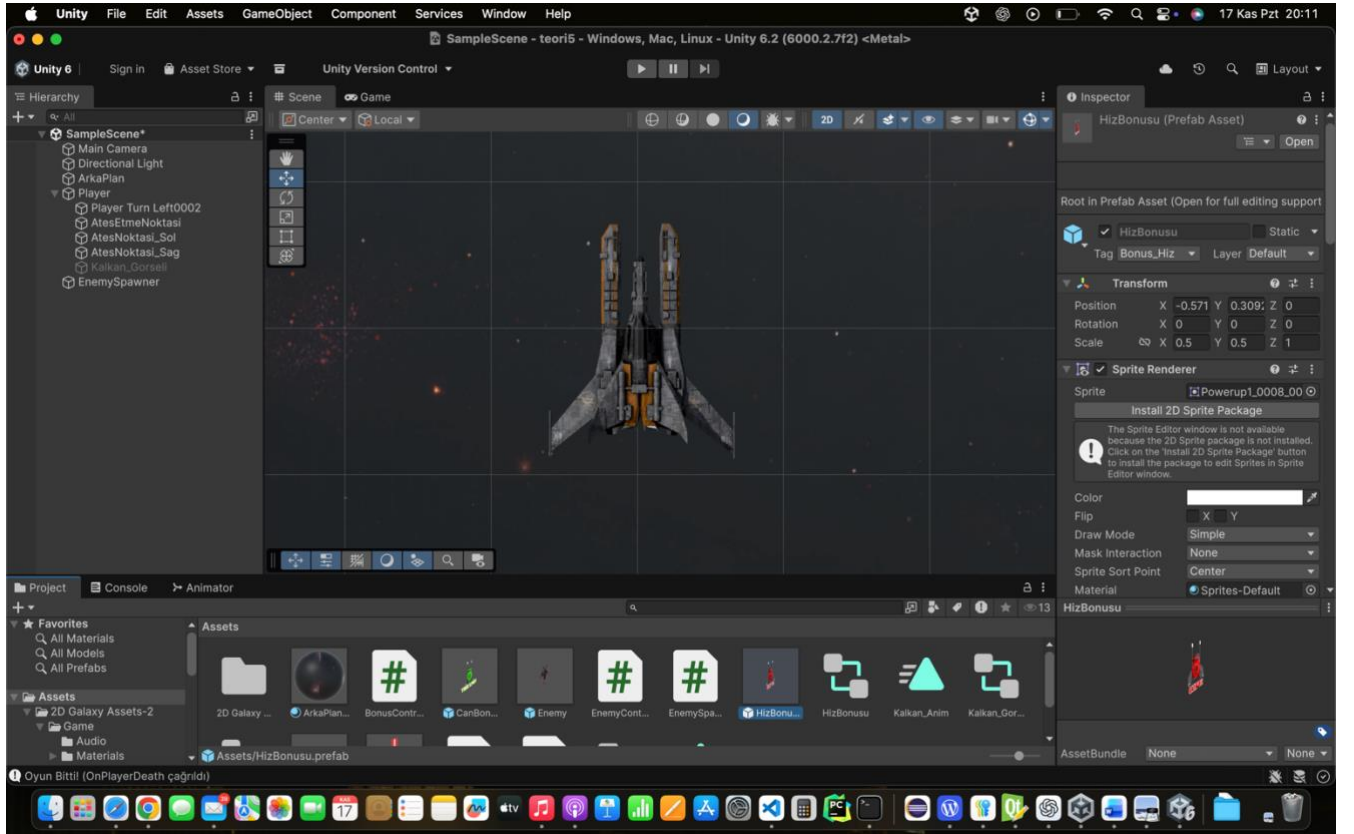
Kalkan bonusu nesnesi için istediğimiz görselleri assets klasöründen animasyona ekleyerek zaman kısa zaman aralıklarıyla farklı renklerle görünmesini sağlarız.



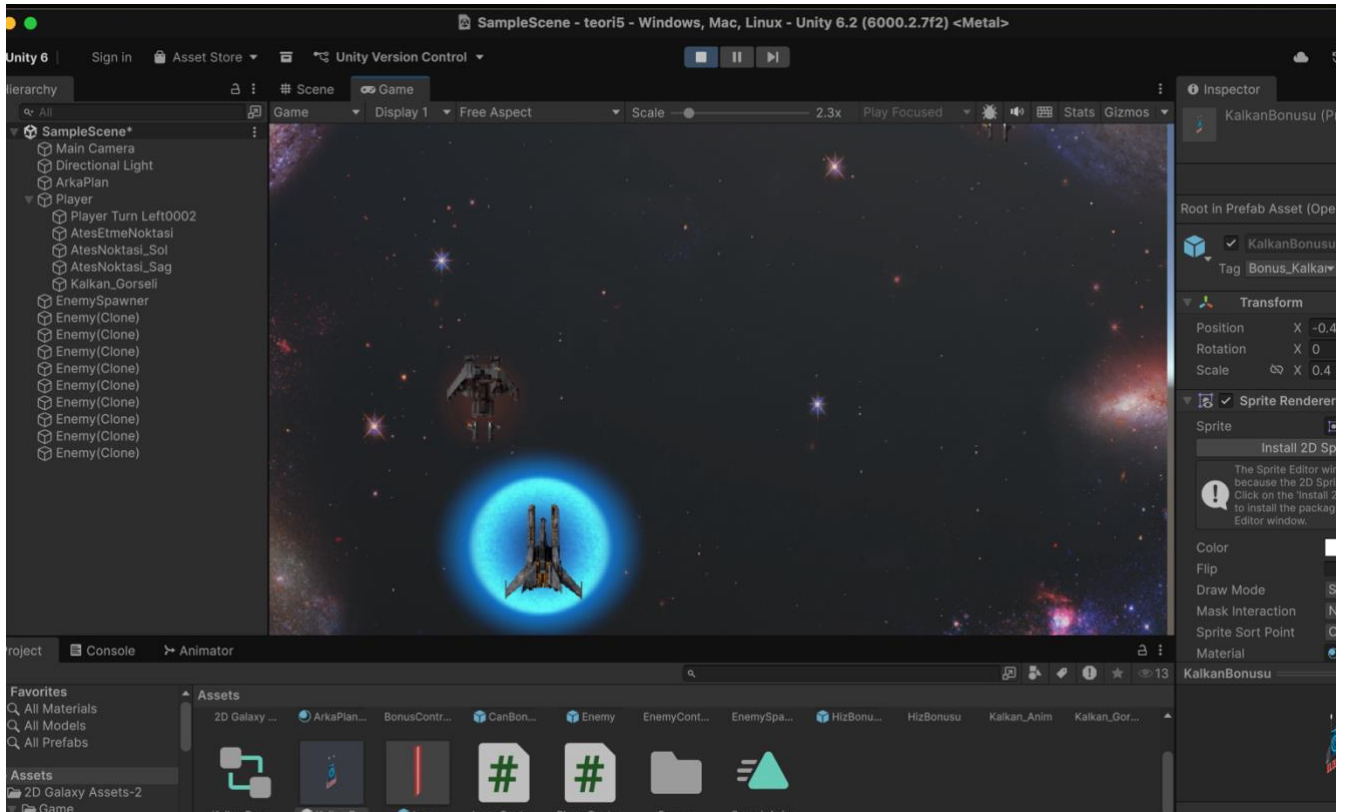
Sornasında diğer bonus nesnesine yaptığımız ayarlamaları yaparız. Rigidbody, Sprite Renderer ve Box Collider gibi komponentleri ekleriz. Sorna bonusu prefab haline getiririz.

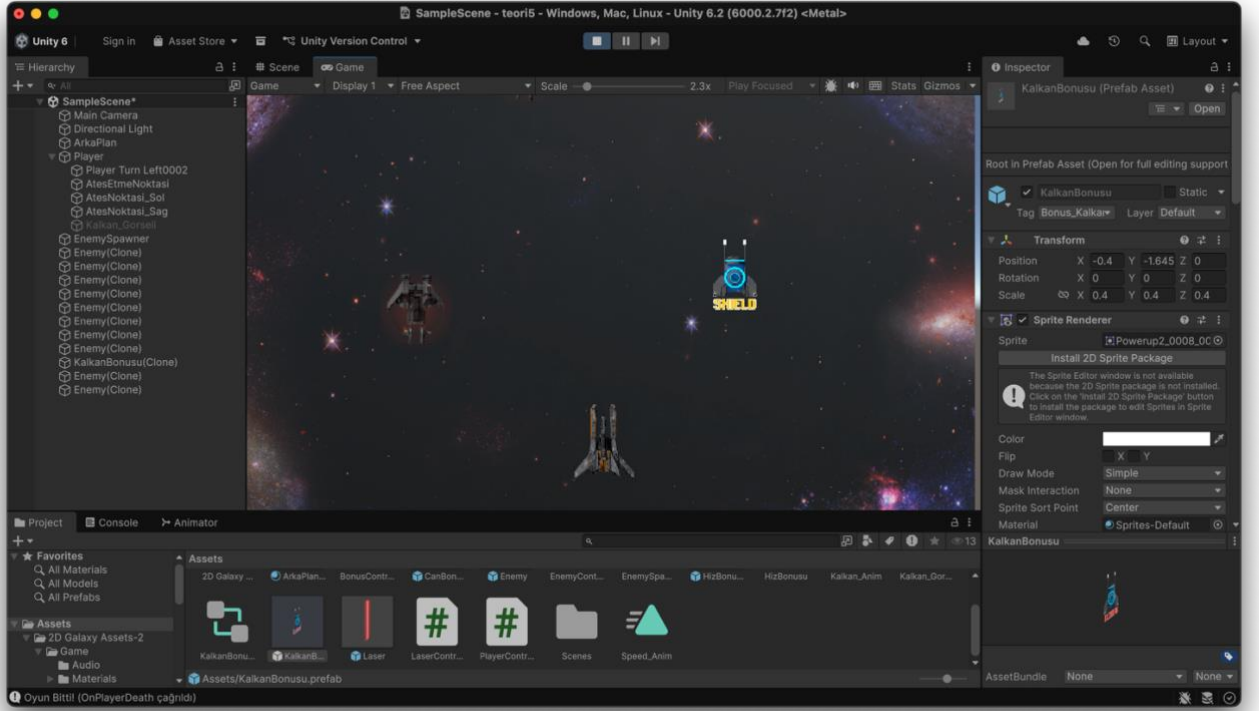


Daha sorna, kalkanı aldığıında player nesnesinin etrafını sarsın diye koruma kalkanı ekleriz.



Diğer bonuslara yaptığımız tüm işlemleri hız bonusu adında oluşturduğumuz boş bir nesne olan gameObject'e de yaparız. Ve onu da diğerleri gibi prefab haline getiririz.





Oyunu başlattığımızda bunlar gibi anlarla karşılaşırız. Şimdi sırada bunun arka planında çalışan kodları inceleyeceğiz.

Bir önceki hafta kullandığımız kodların çoğunu değiştirmedik için onları kısaca anlatacağım ama playerController scriptini daha yakından inceleyeceğiz.

**LaserController :** Lazerin yukarı yönlü hareketini ve yaşam süresini belirledik.

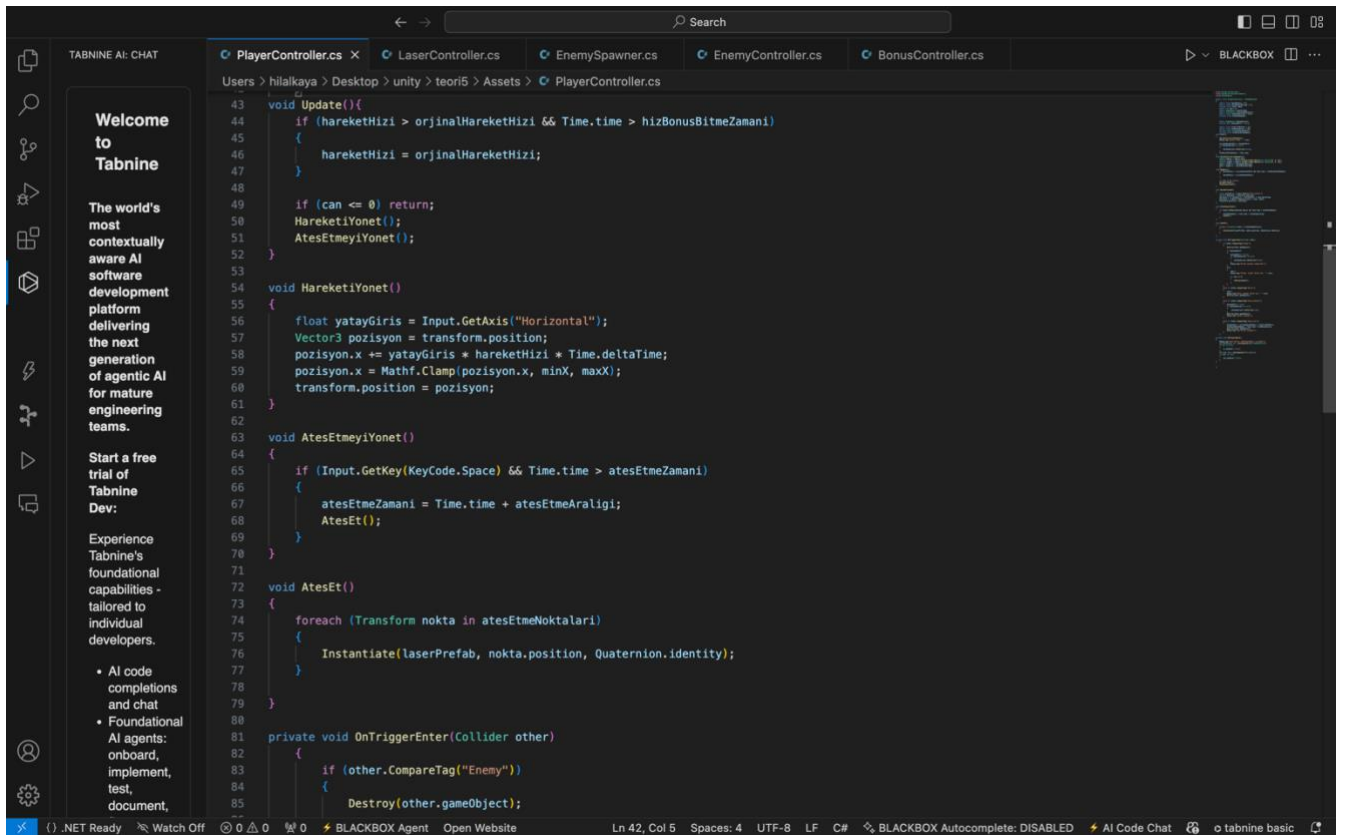
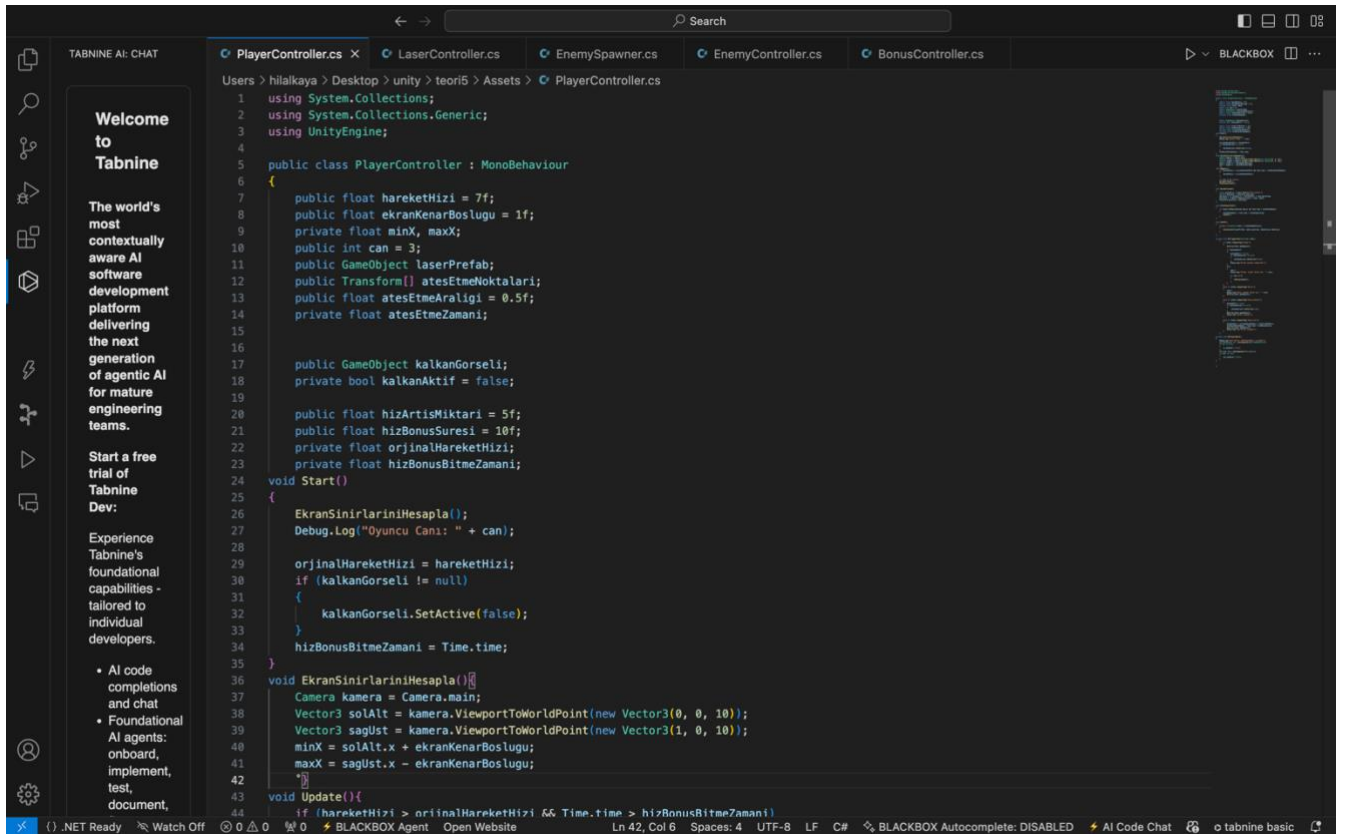
**EnemyController :** Düşman nesnelerinin aşağı yönlü hareketini sağlamak için oluşturulmuştur. Lazerle temasını da algılar.

**BonusController :** Bonusları da aynı düşmanlar gibi aşağı yönlü hareketi için yazılmıştır.

**EnemySpawner :** Oyunu sonsuz bir döngüde tutar. Ve rastgele konumlardan nesnelerin oluşmasını sağlar.

**PlayerController :**





```
92 can--;  
93 Debug.Log("Düşman çarptı! Kalan Can: " + can);  
94 if (can <= 0)  
95 {  
96     OnPlayerDeath();  
97 }  
98  
99  
100  
101 else if (other.CompareTag("Bonus"))  
102 {  
103     can++;  
104     Debug.Log("Bonus alındı! Kalan Can: " + can);  
105     Destroy(other.gameObject);  
106 }  
107 else if (other.CompareTag("Bonus_Kalkan"))  
108 {  
109     kalkanAktif = true;  
110     if (kalkanGorseli != null)  
111     {  
112         kalkanGorseli.SetActive(true);  
113     }  
114     Destroy(other.gameObject);  
115     Debug.Log("Kalkan alındı!");  
116 }  
117 else if (other.CompareTag("Bonus_Hiz"))  
118 {  
119     hareketHizi = orjinalHareketHizi + hizArtisMiktari;  
120     hizBonusBitmeZamani = Time.time + hizBonusSuresi;  
121     Destroy(other.gameObject);  
122     Debug.Log("Hız bonusu alındı!");  
123 }  
124 }  
125  
126 private void OnPlayerDeath()  
127 {  
128     Debug.Log("Oyun Bitti! (OnPlayerDeath çağrıldı)");  
129     SpriteRenderer sr = GetComponent<SpriteRenderer>();  
130     if (sr != null)  
131     {  
132         sr.enabled = false;  
133     }  
134     Collider col = GetComponent<Collider>();  
135     if (col != null)  
136     {  
137         col.enabled = false;  
138     }  
139 }
```

Oyuncunun yatay hareketini (Input.GetAxis("Horizontal")), ekranda kalma sınırlarını (Mathf.Clamp), ateş etmesini (Input.GetKey(KeyCode.Space)), canını (OnPlayerDeath() Fonksiyonu), bonuslarla etkileşimini (other.CompareTag(...) ile etiketini kontrol ederiz) yönetir.

Github Linki : <https://github.com/hilal-kaya/OyunProgramlama/blob/main/Teo6.pdf>