

# Kocaeli Üniversitesi

## Bilgisayar Mühendisliği

### Programlama Laboratuvarı I Sporcu Kart Oyunu

*Ertuğrul Dutçu - Mehmet Fatih Alıcı*  
190201046@kocaeli.edu.tr - 190201056@kocaeli.edu.tr

#### ÖZET

Programlama Laboratuvarı II Projesi olarak bizden C++, C# veya Java nesneye yönelik programlama dillerinden birini kullanarak Sporcuların özelliklerinin karşılaştırılmasını temel alarak oynanan bir oyun yapmamız beklenmektedir. Biz bu proje için Java programlama dilini ve Eclipse geliştirme ortamını, arayüz tasarımı için Java dilinin swing kütüphanesini kullandık.

#### I. GİRİŞ

Oyunda 2 farklı Sporcu ve 2 farklı Oyuncu türü bulunmaktadır. Sporcu türleri Futbolcu ve Basketbolcu'dur. Her sporcunun adı, takımı ve 3 farklı pozisyon için değerleri vardır. Futbolcuların penaltı, serbest atış ve kaleciyle karşı karşıya; Basketbolcuların ikilik, üçlük ve serbest atış pozisyonları için değerleri bulunmaktadır. Oyuncu türleri ise Kullanıcı ve Bilgisayar'dır. Her oyuncunun ID bilgisi, adı, skoru ve kart listesi bulunmaktadır. Oyun başlamadan önce kullanıcıya kullanmak istediği isim sorulur. Oyun, toplamda 16 kartın bilgisayar ve kullanıcı arasında rastgele bir şekilde dağıtılmasıyla başlamaktadır. Daha sonra kullanıcının bir Basketbolcu kartı seçmesiyle oynama evresi başlamaktadır. Kullanıcının seçtiği karta karşılık olarak bilgisayar da rastgele bir basketbolcu kartı seçer. Daha sonra her iki kartın rastgele seçilen pozisyon için değerleri karşılaştırılır. Seçilmesi gereken sporcu türü diğerine geçerek oyun devam eder. Her tur sonunda seçilmesi gereken sporcu türü diğerine geçmektedir. Eldeki kartlar bitene kadar oyun bu şekilde devam etmektedir.

#### II. YÖNTEM

Öncelikle arayüzün gösterileceği bir JFrame oluşturduk. Kullanılan yazıların arkasına gölgelendirme eklemek için JLabel sınıfından kalıtım yapan kendi oluşturduğumuz shadowedJLabel sınıfını kullandık. Böylece tüm yazılarımıza tek bir yerde

gölgelendirme eklemiş olduk. Oyunu yüklemeyen önce kullanıcıdan isim aldık. Daha sonra başta belirlenmiş olan 16 Sporcu kartını Oyuncular arasında Basketbolcu ve Futbolcu kartları 4'er tane olacak şekilde rastgele dağıttık. Tüm kartları JFrame'e ekledikten sonra bilgisayarın kartlarının bilgilerini kullanıcının görememesi için Sporcu sınıfında bulunan setShowable metodu ile gizledik. Sonrasında kullanıcının bir Basketbolcu kartı seçerek oyunu başlatmasını bekledik. Oyuncu ilk kartı seçtikten sonra kartı JFrame'in orta sağ tarafına hareket ettirdik. Sonrasında bilgisayara rastgele bir Basketbolcu kartı seçtirdik ve bu kartı da JFrame'in orta sol tarafına hareket ettirdik. Daha sonra rastgele bir pozisyon seçerek kartların bu pozisyon için değerlerini karşılaştırdık. Seçilen pozisyon için değeri daha fazla olan kartı atan oyuncuya 10 puan ekledik ve kartları JFrame'in dışına hareket ettirdik. Eğer kartların seçilen pozisyon için değerleri eşitse kartların kartKullanildiMi değerini false yaptık ve eski yerine hareket ettirdik. Kartları arayüzde hareket ettirme işlemlerinin hepsinde Sporcu sınıfında bulunan moveTo metodunu kullandık. Ardından seçilmesi gereken sporcu türünü diğerine yani basketbolcuysa futbolcuysa, futbolcuysa basketbolcuysa değiştirerek diğer tura geçtik. Eğer oyuncunun seçmesi gereken sporcudan elinde hiç bulunmuyorsa seçilmesi gereken sporcu türü diğerine otomatik olarak geçirildi. Oyuncunun herhangi bir kartı seçip seçemeyeceği isSelectable değişkeni üzerinden ayarlandı ve yine mouse efektlerinin uygulanıp uygulanmayacağı da bu değişken üzerinden kontrol edildi. Ayrıca kartlarda görüntü bozukluklarının önüne geçmek için hasBrightened değişkeni kullanılarak efektlerin üst üste uygulanılamaması sağlandı. Oyuncuların ellerindeki tüm kartlar bitene kadar oyun böyle devam etmektedir ve kartlar bittiğinde oyun bitmektedir. Eğer oyuncuların son kartlarının 3 pozisyon değerleri de eşitse bu kartlar elde kalarak oyun yine bitmektedir.

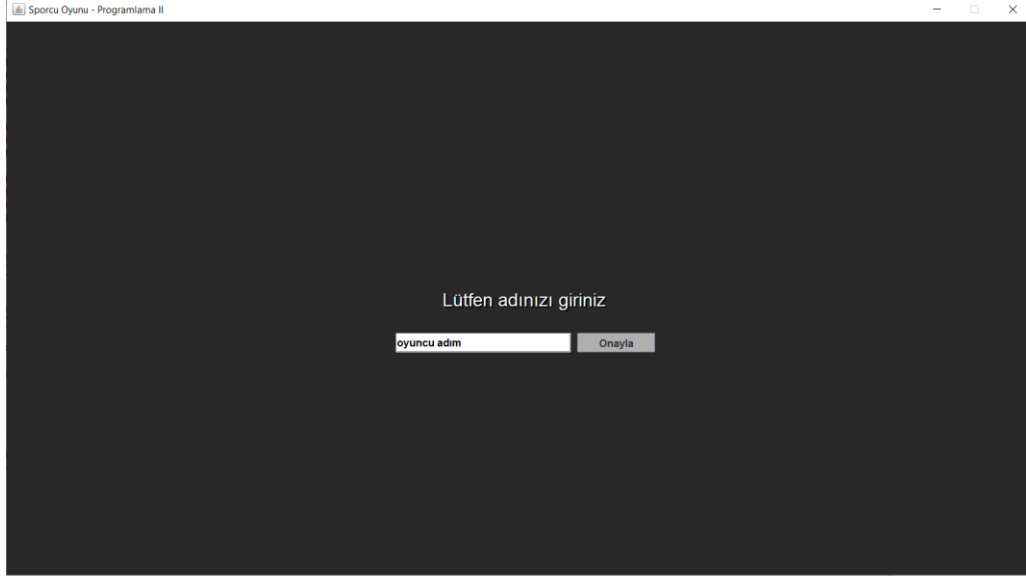
### III. SONUÇ

Bu proje sayesinde nesneye yönelik programlama dilinin özelliklerini kullanarak arayüze sahip bir kart oyunu tasarladık. Zamana

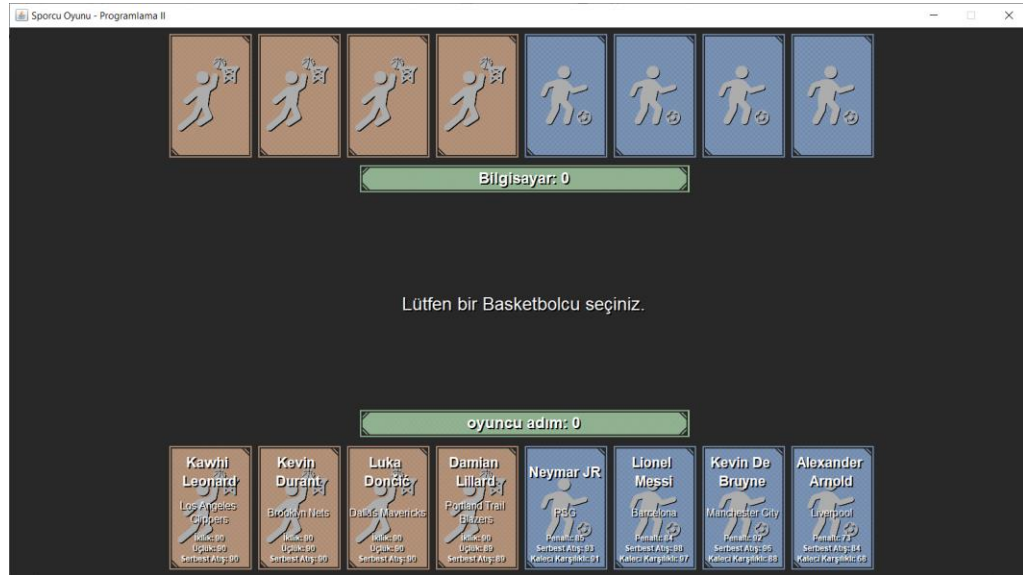
bağlı işlemler yapmayı yani Timer kullanımını, Encapsulation, Inheritance, Polymorphism, Abstraction gibi terimlerin anlamlarını ve hangi durumda kullanılmaları gerektiğini öğrendik.

### IV. ÇIKTILAR

Oyuncudan isim alma ekranı:



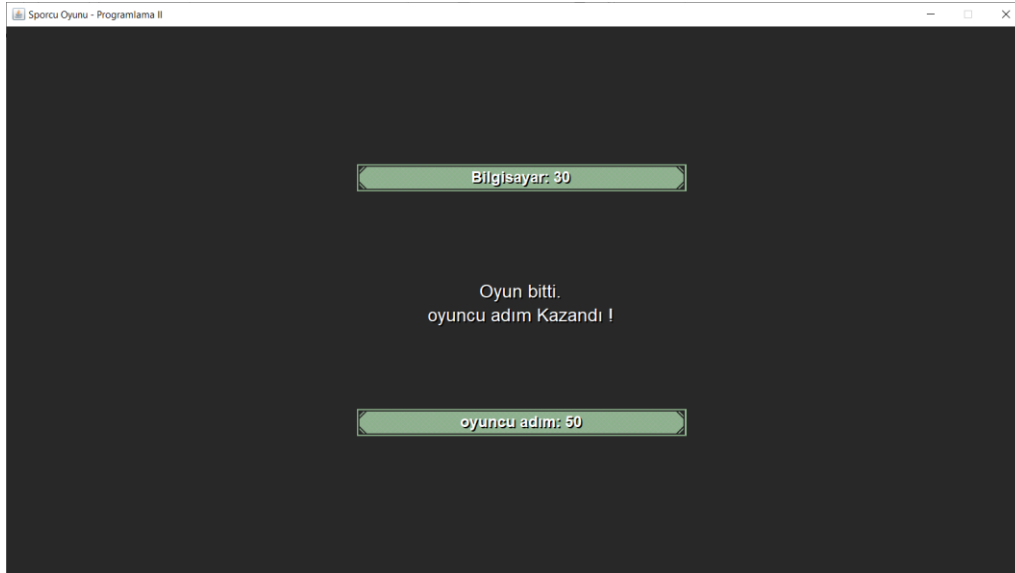
Oyuncudan isim alma ekranı:



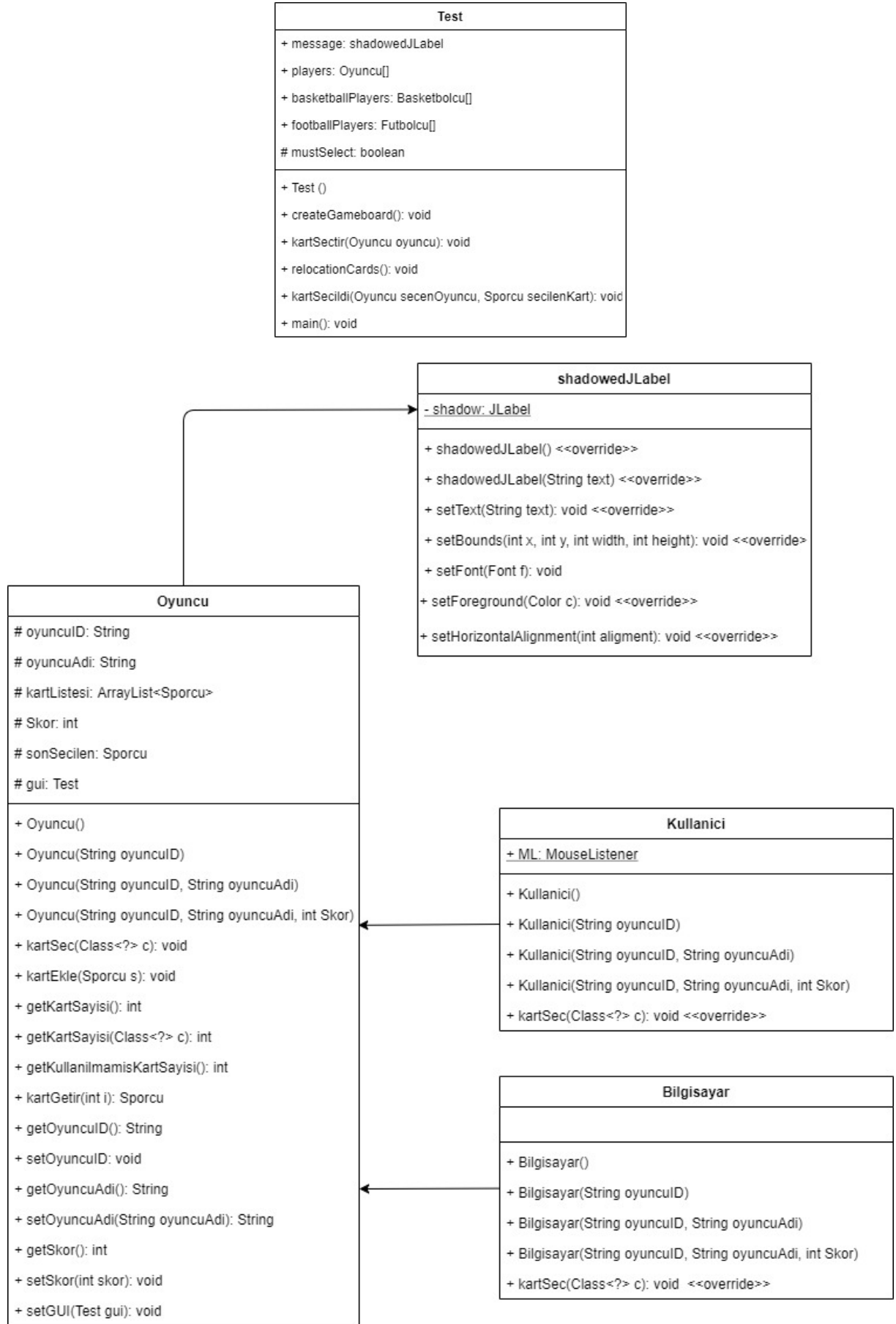
## Sporcuların karşılaştırılma ekranı:

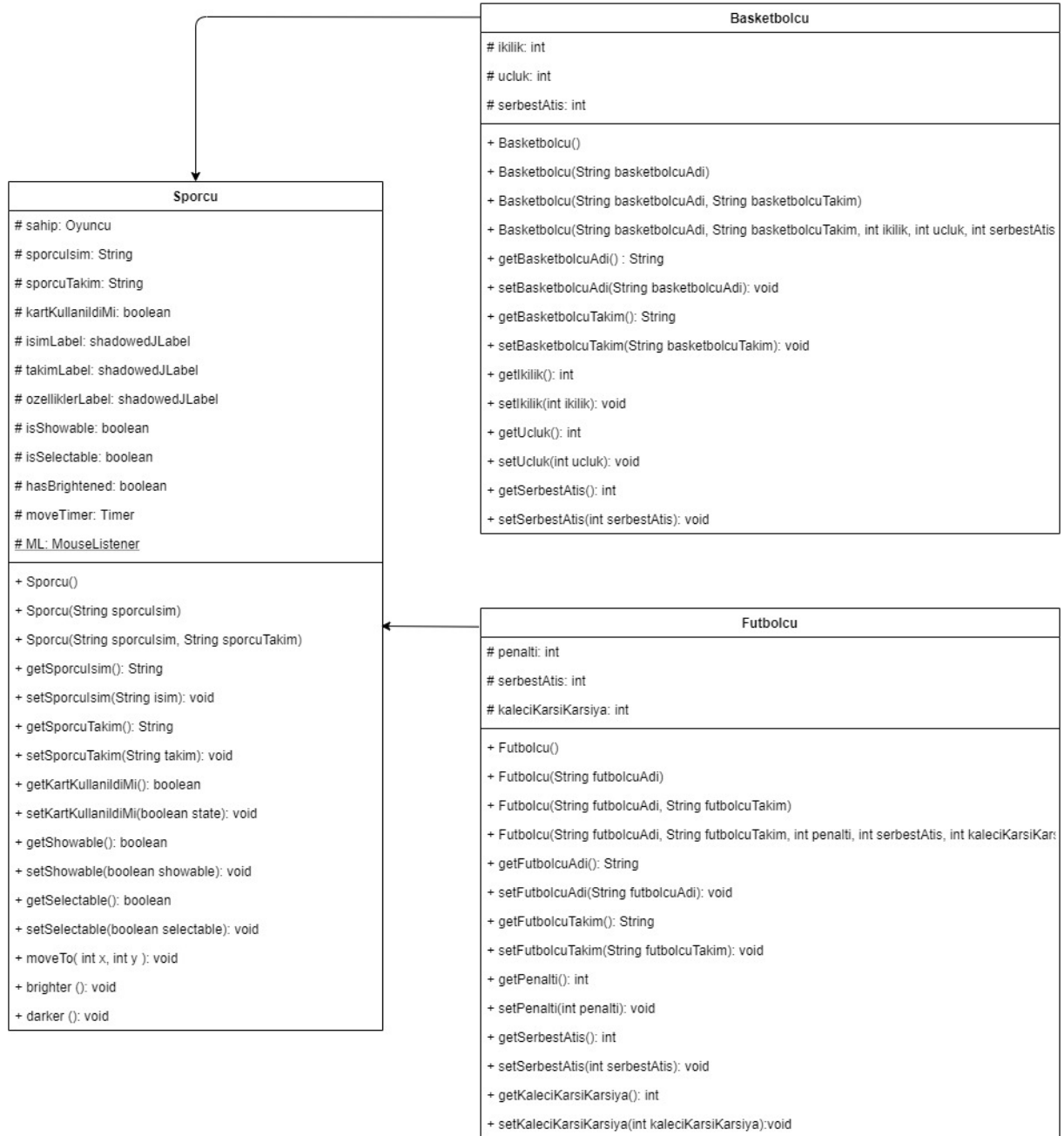


## Oyun sonu ekranı:

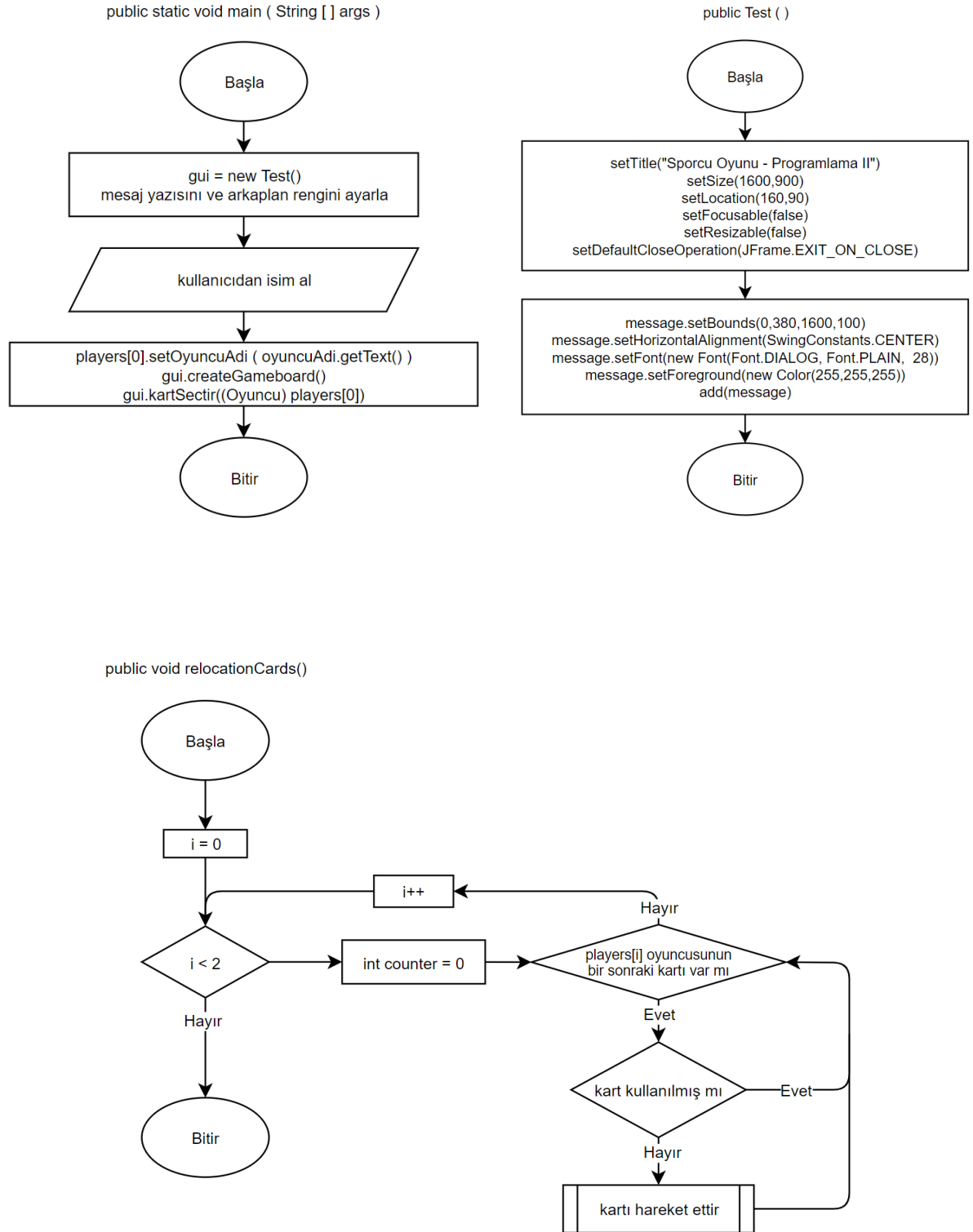


# UML SINIF DİYAGRAMLARI

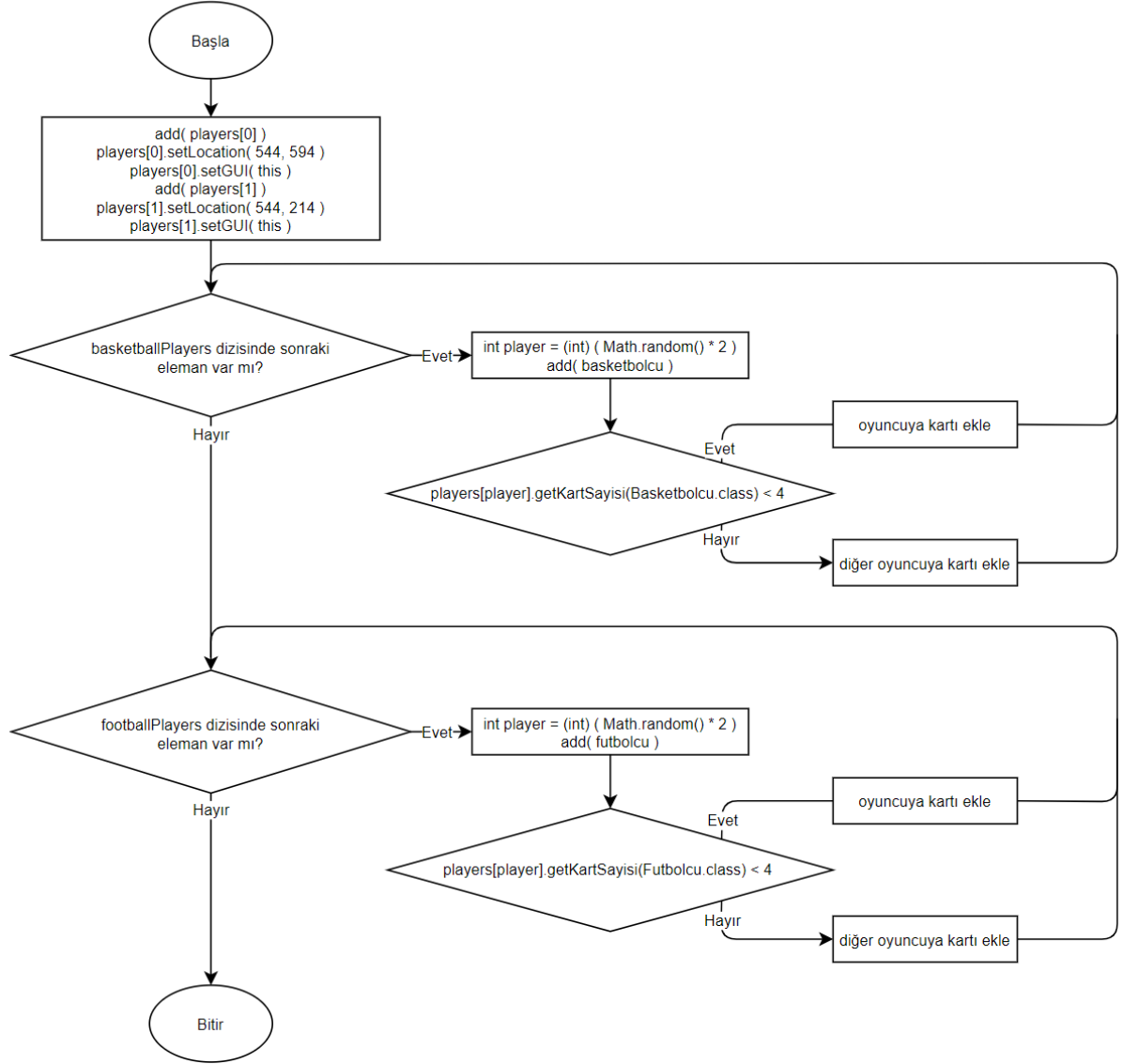




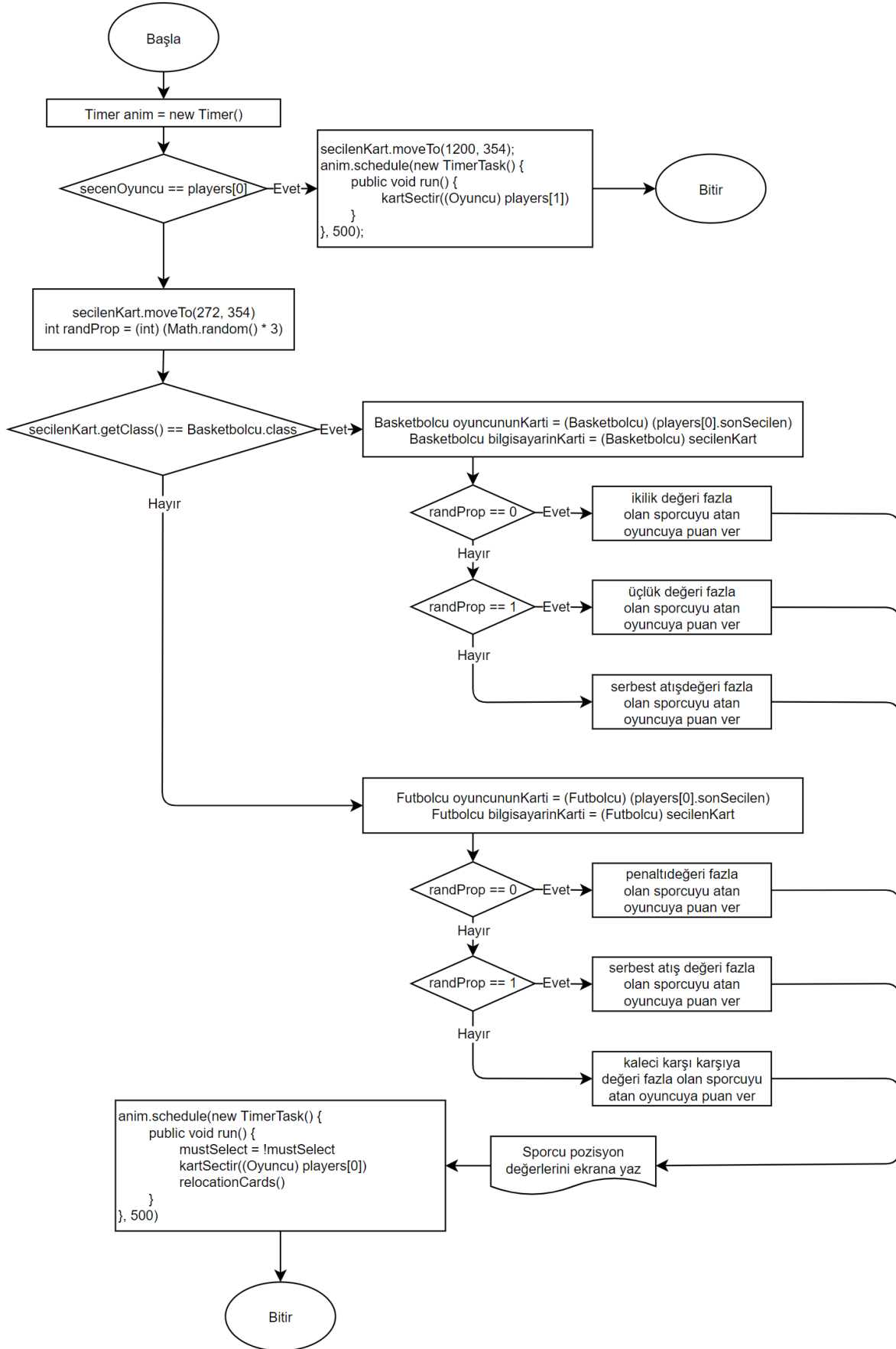
## AKIŞ ŞEMALARI



public void createGameboard ( )

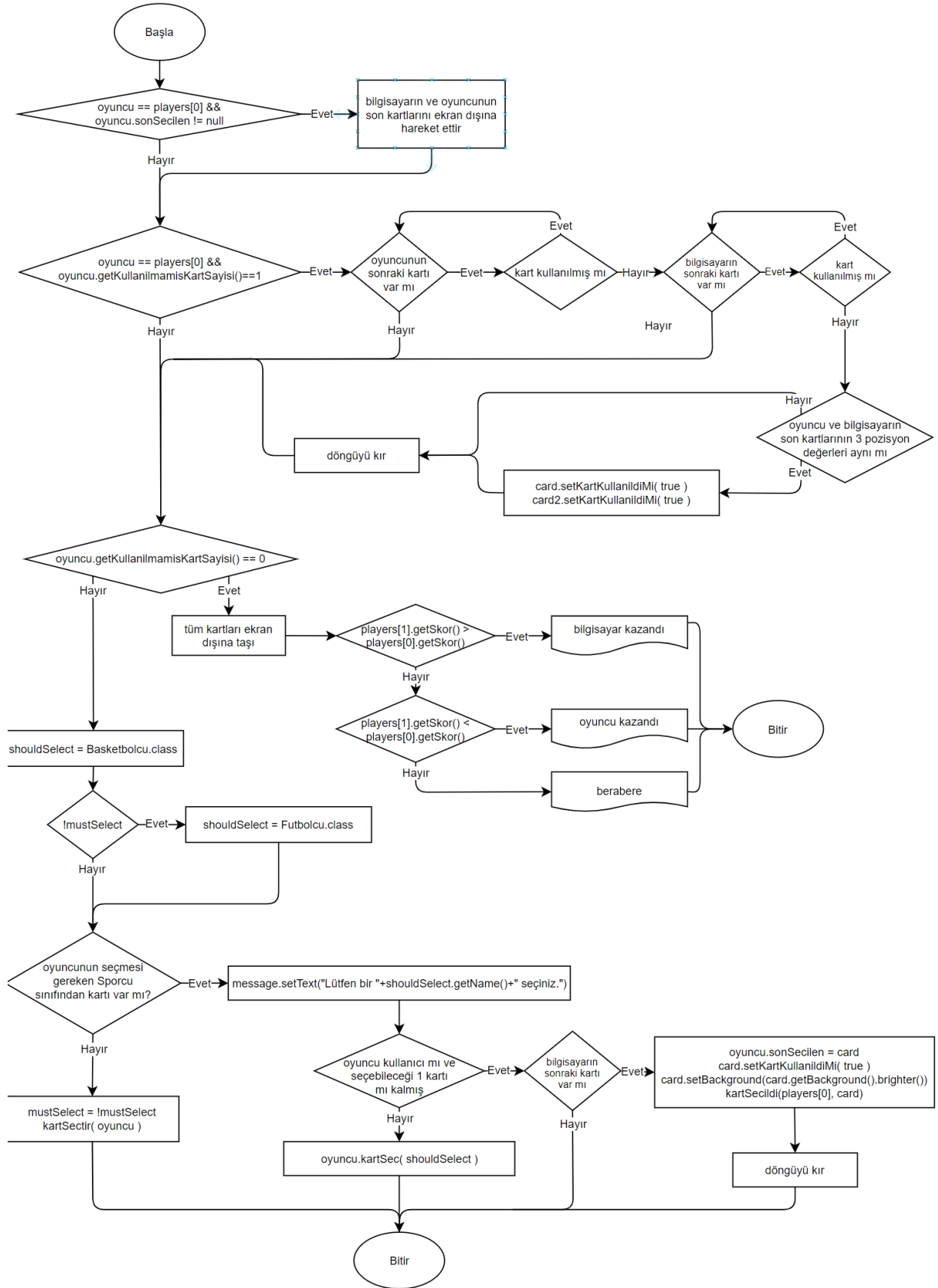


public void kartSecildi( Oyuncu secenOyuncu, Sporcu secilenKart )





public void kartSectir ( Oyuncu oyuncu )



## KAYNAKÇA

- [1] [https://www.tutorialspoint.com/java/java\\_inheritance.htm](https://www.tutorialspoint.com/java/java_inheritance.htm)
- [2] [https://www.tutorialspoint.com/java/java\\_polymorphism.htm](https://www.tutorialspoint.com/java/java_polymorphism.htm)
- [3] <https://docs.oracle.com/javase/tutorial/uiswing/index.html>
- [4] <https://www.tutorialspoint.com/swing/index.htm>
- [5] <https://www.baeldung.com/java-timer-and-timertask>
- [6] <https://docs.oracle.com/javase/7/docs/api/java/util/Timer.html>