

PROGRAMLAMA LABORATUVARI 2. PROJE

SPORCU KART OYUNU

Kocaeli Üniversitesi
Bilgisayar Mühendisliği

Feyzanur yeşildal
190201085

Hilal Fatma Gül Nişancı
190201041

hilalfnisanci@gmail.com , feyzanuryesildal@gmail.com

GİRİŞ

Bu projenin amacı kullanıcının bilgisayara karşı rekabet edebileceği basit bir kart oyunu tasarlamaktır. Bu oyunda 16 adet kart bulunmaktadır ve bu kartların 8'i basketbol 8'i futbol kartlarından oluşmaktadır. Oyun başladığında rastgele olarak 4 futbol ve 4 basketbol olmak üzere kullanıcılara kartlar dağıtılacaktır. Oyunda Futbol kartına karşı futbol kartı ve basketbol kartına karşı da basketbol kartının özellikleri karşılaştırılacaktır. Futbolcuların penaltı, asist ve serbest vuruş özellikleri bulunmaktadır. Basketbolcuların üçlük, ikilik ve serbest atış özellikleri bulunmaktadır. Bu proje java programlama dili kullanarak geliştirilmiştir ve ide olarak Netbeans kullanılmıştır.

ÖZET

Bilgisayara ve kullanıcıya rastgele olarak 4'ü futbol ve 4'ü basketbol kartı dağıtılır. Kullanıcı başlangıçta bir futbol kartı atar ve kullanıcı kart attıktan sonra bilgisayar da random bir şekilde futbol kartı atar. Bu kartlarda bulunan üç özellikten biri random bir şekilde karşılaştırılır. Özelliği büyük olan oyuncu 10 puan alır. Eğer karşılaştırılan özellik eşitse kartlar geri alınır ve kart atma sırası diğer takıma geçer. oyun bu şekilde bütün kartlar bitene kadar devam eder. Tüm kartlar bittikten sonra skoru yüksek olan oyuncu kazanır.

YÖNTEM

Projeye, bizden istenen classları ve özellikleri oluşturarak başladık. Oluşturduğumuz classlarda javanın nesneye yönelik programlama yapısına uygun olarak encapsulation, inheritance, polymorphism, abstraction gibi özelliklerine dikkat ederek projeyi oluşturduk.

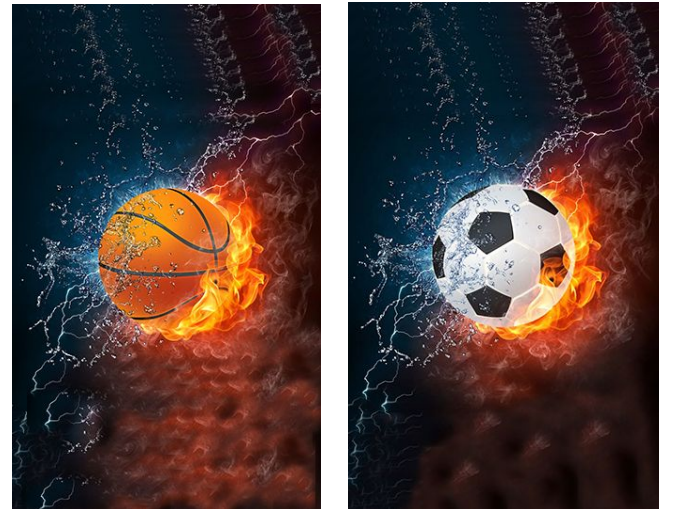
JFrame arayüzünü kullandığımız bu projede, kullanıcıdan aksiyon alabilmek için buton ekledik ve yine oyun takibini arayüzde görselleştirmek için photoshop programını kullanarak kart ve arka plan tasarımını yaptık.

Oyunda, seçtiğimiz futbolculara ve basketbolculara özellikleri için belli puanlar verdik. ve verdiğimiz bu özellikleri de oluşturduğumuz kart görsellerine ekledik. (bkz: şekil 1)



Şekil 1. Basketbolcu ve Futbolcu Kartları

Oyunda bilgisayarın kartlarını göremeyeceğimiz için futbol ve basketbol kartları arka planı için de farklı tasarımlar yaptık. (bkz: şekil 2)



Şekil 2. Kartların Arka Yüzü

Oyun başladığında oyuncu 4'ü futbol ve 4'ü basketbol olmak üzere 8 kart görmektedir. Bilgisayara da aynı şekilde kartlar rastgele bir şekilde dağıtılmıştır. Oyun anında bilgisayarda bulunan kartları göremiyoruz. Kartlar oyuncu seçimine göre ve bilgisayar için random olarak ortada rastgele belirlenen özelliklerine göre karşılaştırılıyor.

Oyun, oyuncunun bir futbol kartı seçmesiyle başlıyor. Seçilen futbol kartı ortada bulunan karşılaştırma alanına gidiyor ve aynı anda da bilgisayar bir futbol kartını rastgele seçiyor. Seçilen kartlar ortadaki karşılaştırma alanında rastgele 3 özellikten biri seçilerek karşılaştırılıyor. Karşılaştırılan özelliğe göre yüksek olan oyuncunun skoru 10 artıyor. Eğer karşılaştırılan özellik puanı her iki kartta da eşitse kartlar oyunculara geri veriliyor ve başka kart oynayarak oyun devam ediyor.

Kartlar geri alındığında sıra basketbol kartlarına geliyor ve oyuncunun bir basketbol oyuncu kartı seçmesiyle bilgisayarın attığı kart karşılaştırılarak aynı işlemler devam ediyor.

Oyuncunun ve bilgisayarın elindeki tüm kartlar bittiğinde skoru yüksek olan oyuncu oyunu kazanır veya skorların eşitlik durumunda oyun berabere biter.

OLUŞTURDUĞUMUZ SINIFLAR

❖ Sporcu Sınıfı:

Bu sınıfın parametreleri `sporculsim` ve `sporcuTakim` olarak iki tanedir. Sporcuların puanını göstermek için `sporcuPuanGoster` metodunu içerir. Kartların

❖ Futbolcu Sınıfı:

Bu sınıf `Sporcu` sınıfından kalıtım alır. `Constructor` metodları içerir. Bu sınıfın `penalti`, `asist` ve `serbestVurus` özellikleri bulunmaktadır. `sporcuPuanGoster` metodu `override` edilir. `Boolean` veri tipinde `kartKullanildiMi` bilgisi tutulmaktadır.

❖ Basketbolcu Sınıfı:

Bu sınıf `Sporcu` sınıfından kalıtım alır. `Constructor` metodları içerir. Bu sınıfın `ikilik`, `ucluk` ve `serbestAtis` özellikleri bulunmaktadır. `sporcuPuanGoster` metodu `override` edilir. `Boolean` veri tipinde `kartKullanildiMi` bilgisi tutulmaktadır.

❖ Oyuncu Sınıfı:

Bu sınıfın parametreleri `oyuncuID`, `oyuncuAdi` ve `Skor` olmak üzere üç tanedir. `kartListesi` özelliği ile oyuncuların elinde bulunan kartlar listede tutulur. `SkorGoster()` fonksiyonu ile oyuncuların skorları gösterilir. `kartSec()` fonksiyonu yazılmıştır.

❖ Bilgisayar Sınıfı:

Bu sınıf `Oyuncu` sınıfından kalıtım alır. `Constructor` metodları içerir. `Oyuncu` sınıfında bulunan `kartSec()` metodu `override` edilir ve özelleştirilerek kullanılır.

❖ Kullanıcı Sınıfı:

Bu sınıf `Oyuncu` sınıfından kalıtım alır. `Constructor` metodları içerir. `Oyuncu` sınıfında bulunan `kartSec()` metodu `override` edilir ve özelleştirilerek kullanılır.

❖ CreatorClass Sınıfı:

Futbolcu ve basketbolcuların isimleri ve özellikleri burada tanımlanmaktadır. Kartlar oyunculara bu sınıfta `kartlariDagit()` metodu ile rastgele dağıtılır.

❖ Test Sınıfı:

Bu sınıf `javax.swing.JFrame` den kalıtım almaktadır. Oyunun ilerleyişi bu sınıfta takip edilmektedir. Oyunun ilerleyiş algoritması bu sınıfta oluşturulmuştur. Kartların hangi özelliğinin karşılaştırılacağı bu sınıfta random olarak belirlenmektedir. Kartların bitmesi ile `oyunSonu()` metodu kullanılarak, mesaj kutusunda kazananın puanı ve ad bilgisi gösterilmektedir.

ALGORİTMA ANALİZİ

Oyun giriş ekranında kullanıcıdan isim alınmasıyla başlar. Kartlar rastgele bilgisayar ve kullanıcı listelerine dağıtılır. Oyun akışında `sıra=0` olduğunda kullanıcı ortaya bir kart atar. `sıra=1` olduğunda bilgisayar rastgele bir kart atar. Kartlar atıldıktan sonra iki kartın üç özelliğinden biri rastgele karşılaştırılır. Puanı yüksek olan kartın sahibine 10 puan yazılır. Eğer beraberlik durumu varsa geri al butonu ile kartlar geri alınır. Tüm kartlar bitene kadar işlem devam eder.

SÖZDE KOD

- 1- Başla.
- 2- Kart tanımlamalarını yap.
- 3- Tanımlanan kartları `ArrayListe` ata.
- 4- Random bir şekilde `ArrayListteki` kartları labelde göster.
- 5- Kullanıcının bir futbol kartı seçmesini bekle.
- 6- Karta tıklandığında o kartı ortaya al.
- 7- Bilgisayardan random bir şekilde kart ortaya al.
- 8- Random bir sayı at.
- 9- Alınan random sayıya göre özellikleri karşılaştır.
- 10- Eğer kullanıcının puanı yüksekse skorunu 5 arttır.
- 11- Bilgisayarın puanı yüksekse bilgisayarın puanını 5 arttır.
- 12- Özellik puanları eşitse skor arttırma ve kartları geri al butonu çıkar.
- 13- Butona basıldığında kartları geri al.
- 14- Kullanıcının bir basketbol kartı seçmesini bekle.
- 15- Karta tıklandığında o kartı ortaya al.
- 16- Bilgisayardan random bir şekilde kart ortaya al.

17- Random bir sayı at.

18- Alınan random sayıya göre özellikleri karşılaştır.

19- Eğer kullanıcının puanı yüksekse skorunu 5 arttır.

20- Bilgisayarın puanı yüksekse bilgisayarın puanını 5 arttır.

21- Özellik puanları eşitse skoru arttırma ve kartları geri al butonu çıkar.

22- Butona basıldığında kartları geri al.

23- Elde kart kalmayınca kadar döngüyü sürdür.

KAYNAKÇA

<https://www.w3schools.com/java/>

[JFrame in Java | Learn Methods And How to Construct JFrame In Java \(educba.com\)](#)

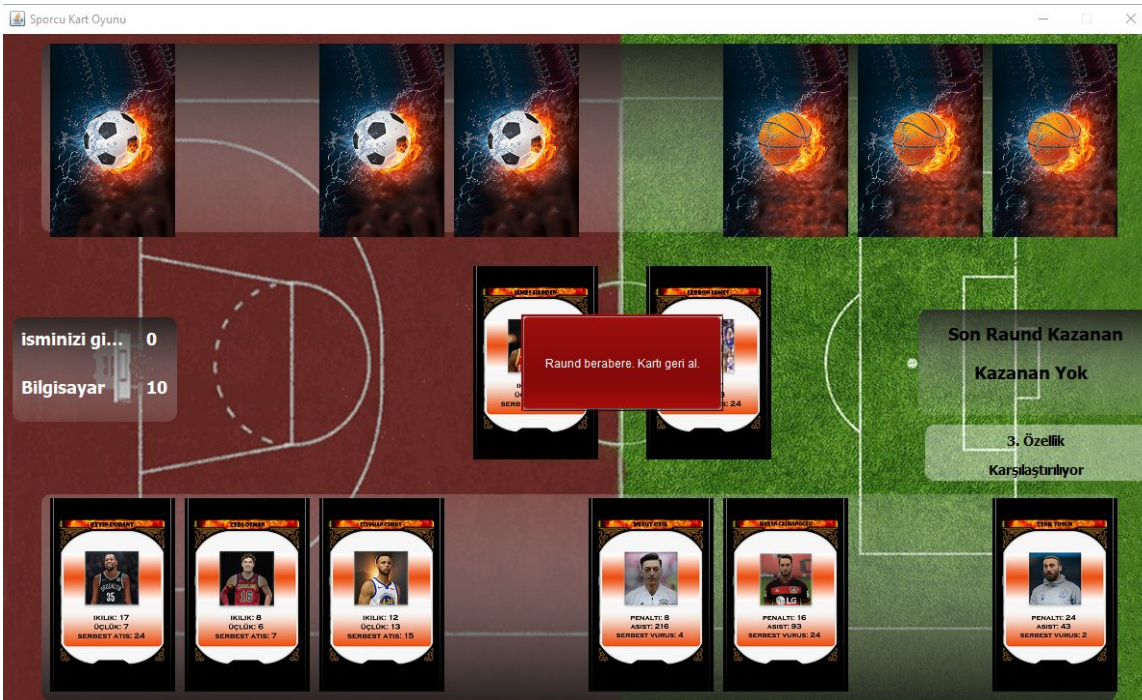
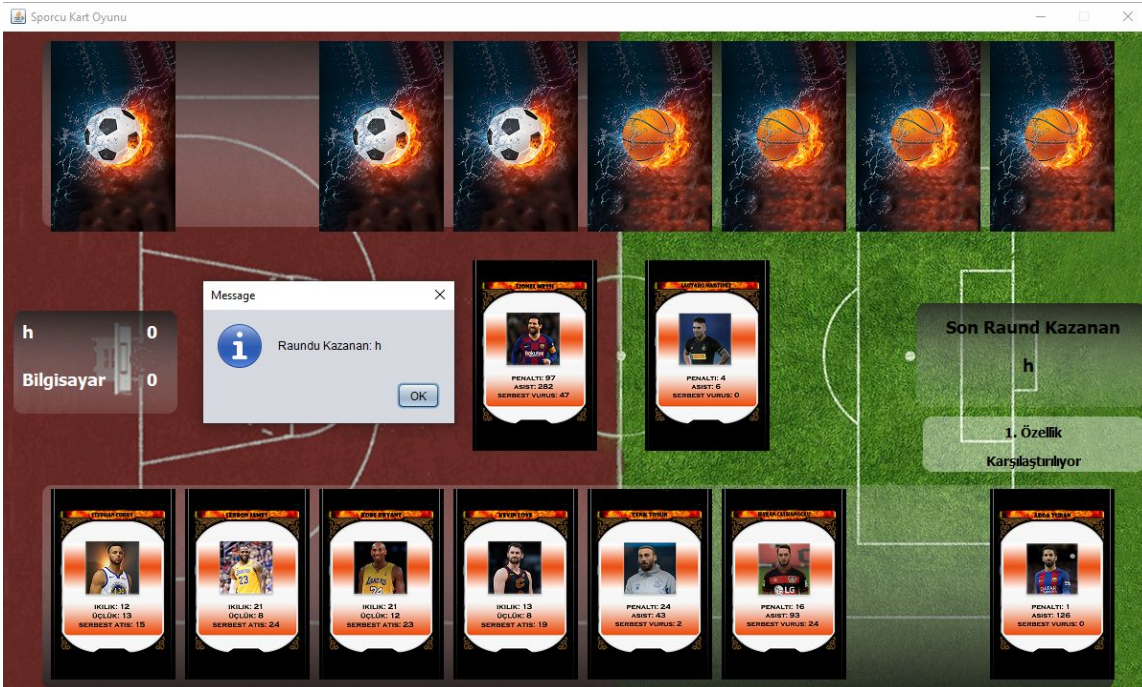
[Java Swing - Tutorial And Example](#)

[List get\(\) method in Java with Examples - GeeksforGeeks](#)

[ArrayList removeAll\(\) method in Java with Examples - GeeksforGeeks](#)

Karşılaştığımız hatalar için: <http://stackoverflow.com/>

DENEYSEL SONUÇLAR



UML DİYAGRAMI

