

## I.Özet

Bu proje java dilinde yazılmıştır. NYP mantığına uygun olarak yazıldı. Bilgisayara karşı rekabet edilen bir kart oyunudur. Bilgisayarın rastgele seçmiş olduğu kartlara karşı kullanıcının ara yüzde seçtiği kartlar savaş yapar. Kartların savaş sonucuna göre kullanıcı ve bilgisayar skor puanı artar. Belli hamle sayısı, kartların bitmesi ya da 1 kart kalma durumuna göre oyun biter ve kazanan taraf ilan edilir.

## II.Giriş

Oyunun başında bilgisayar ve oyuncunun elinde hava, kara ve deniz olmak üzere 3 farklı kategorideki savaş araçlarını içeren rastgele oluşturulan kartlardan 6'şar adet kart vardır. Önce oyuncu, destesindeki kartlar arasından üç tane kart seçer daha sonra bilgisayar, destesindeki kartlar arasından rastgele üç kart seçimi yapar. En son kartlar karşılaştırılarak skorlar hesaplanır ve hamle tamamlanır. Her yeni adımda taraflara rastgele birer kart eklenir. Eğer taraflardan birinin kartı 1 tane kalmış ise ona 2 kart verilir yani 3'e tamamlanarak diğer adım son hamle olur. Kartlar ya da adım sayısı bitince de oyun biter.

Destedeki bütün kartlar seçilmeden daha önce seçilmiş bir kart 2. defa seçilemez. Destedeki tüm kartlar birer kez seçildiğinde, daha önce seçilmiş kartlar tekrar seçilebilir.

3 farklı kategori: Hava Araçları kategorisinde Uçak ve Siha; kara araçları kategorisinde Obüs ve KFS (kara füze sistemi); deniz araçları kategorisinde ise Fırkateyn ve Sida savaş araçlarıdır.

Kartların belli güçte vuruş ve dayanıklılık özellikleri vardır. Vuruş gücüne göre dayanıklılık azalır ve 0 ya da altına düşerse kart ölür. Her kartın başka kategorideki karta karşı ekstra vuruş avantajı vardır. Eğer savaştığı kartın kategorisi vuruş avantajı olan bir kart ise o karta özgü daha fazla

vuruş yapar. Siha, KFS, Sida özel kartlardır. Bu 3 kart taraflar belli bir skora ulaştıktan sonra destelerine eklenebilir. Özel olmasının nedeni 2 tane kategoriden vuruş avantajı olmasıdır.

Kartların seviye puanı vardır. Rütbe gibi düşünülebilir. Savaşta kazanan kartın seviye puanı, ölen kartın seviye puanı kadar attırılır. Eğer düşman kartın seviye puanı 0 ise kazanan karta 10 puan eklenir. Bir kartın her kazandığı savaş adımıyla kartın puanı artarak devam eder. Oyuncuya da skor olarak getirisi yüksektir.

Bilgisayar ve kullanıcı skor özelliği vardır. Seçtiği kartı kazanır ise kartın kazandığı seviye puanı kadar skorlarına eklenmiş olur. Kazanan tarafı belirlemek için kritiktir.

Taraflardan birinin elindeki tüm nesneler tükendiğinde rakip oyuncu oyunu kazanacaktır. Adım sayısı ya da 1 kart kalma durumuna göre oyun biterse, iki tarafın da elinde kart varsa, bu durumda tarafların skor değeri yüksek olan oyunu kazanacaktır. Eğer skorlar eşit olursa bu durumda tarafların elindeki elenmemiş mevcut kartların toplam dayanıklılık miktarı en fazla olan oyunu kazanacak ve toplam dayanıklılık değeri arasındaki fark kazananın skoruna eklenecektir.

## III.Yöntem

Nesne sınıflarını oluşturarak başlandı. Savaş Araçları abstract sınıf yapısı ile tanımlandı. Kart nesnelerinin ortak özellikleri burada tanımlandı. Kartın ilgili kategorisine özgü olan özelliklerini Hava, Kara, Deniz abstract sınıfı şeklinde sınıflar oluşturuldu. Savaş Araçları sınıfından kalıtım alırlar. Constuctor metodu, Savaş Araçları sınıfından gelen özelliğe atama yapmak için super() kullanıldı. Kart nesnelerinin sınıfları, ilgili kategorideki sınıftan kalıtım aldı. Üst sınıflardan gelen metotları kullanmak için override kullanıldı. Hem bilgisayar hem de kullanıcı için nesneler

Oyuncular sınıfı üzerinden türetildi. Oyuncuların ortak bilgileri bu sınıfta oluşturuldu. Kullanıcı ve Bilgisayar sınıfı Oyuncular sınıfında kalıtım alır. kartSec metodu burada override edilerek kullanıldı. Oyun sınıfı oluşturuldu. Bu sınıfta savaş yapma, deste atama, deste ekleme gibi metotlar yazıldı.

GUI sınıfı ara yüz ve arka plan için aracı görevi gören sınıftır. Oyun ve Controller arasında iletişimi sağlar. Oyunun akışını sağlayan yerdir. Oyun sınıfında yazılan metotları burada kullanarak elde edilen verileri Controller sınıfına göndererek ara yüzde gösteriyoruz. Start metodu görüntü için Application sınıfından override edilerek ilk görüntü işlemi, yüklenmesi vb. burada başlar. savaşYapButton metodu Savaş Yap butonu tıklanınca kadar bekler ve sonsuza kadar döner. Kart destesi gösterme, kullanıcı ve bilgisayar kart seçme savaş yapma gibi oyunun akışı burada sağlanır. Metodun içinde adımın bitmesi için koşullara göre tıklama durur. Oyun ve Controller sınıfı arasında ara yüz için köprüdür.

Controller sınıfı ara yüzde yapmak istenilen durumların metotları yazılıdır. GUI'dan gönderilen parametrelere göre nesnelerin özellikleri, görselleri gösterilir. Ara yüzde kullanıcı görsele tıklayarak seçmek istediği kart seçilebilir. Controller sınıfına ilk olarak gosterKartlar metodu ile gelinir. Burada savaş araçlarının görselleri ara yüzde gösterilir ve görsellere tıklayarak kullanıcı kart seçer. secileni\_Ekle metodu seçilen kartın geçmişte seçilip seçilmediğini kontrol eder. Bilgisayar\_Secilen\_Karti\_Goster ve Kullanici\_Secilen\_Karti\_Goster metotlarıyla tarafların savaşması için seçtikleri üçer kartın görseli ara yüzde gösterilir. efekt\_ekle metodu ile destedeki seçilen kartlara efekt uygulanarak o kartın seçildiği belli edilir ve kartın tekrardan seçilmesi engellenir. efekt\_kaldir metodu ile destedeki tüm kartlar seçildiğinde, geçmişte seçilen kartlar tekrardan seçilebilir hale getirilir ve efektleri kaldırılır. temizleSecilenKartlar ile yeni savaş adımına geçildiğinde ara yüzde gösterilen tarafların seçtiği kartlar ara yüzden kaldırılır. initialize metodu tarafların seçtiği kartlara çerçeve ekler ve program başından sonuna kadar çerçeveler silinmez. Sadece çerçeve içine gelen kartlar değişir. kazananigoster metodu ile savaş sonlanınca ara yüze kimin galip geldiği yazdırılır. Oyun sınıfında; oyuncular, kartlar, dosya yazmak

için değişken, geçmiş seçilen kartlar için Liste Constuctor oluşturuldu. Main metodu bu sınıf içerisinde GUI sınıfını başlatıyor. DesteAta metodu oyun başında 6 kartı rastgele tarafların destesine eklenir. desteEkle metodu ile her adımda rastgele destelere birer kart eklenir. Oyunculardan birinin skoru 20 den fazla ise özel kartlar da aktif olur ve sonraki adımlarda destelere eklenebilir. savaşYap metodu oyuncuların seçtiği kartları karşılıklı vuruş gücünü (avantajı varsa onu da ekleyerek) hesaplayıp dayanıklılığından çıkarır. Kartların kazanma durumuna göre seviyePuanlarını ve oyuncuların da skorlarını günceller. Ölen kart listeden silinir. aracOluştur metodu yeni kart oluşturulacağı zaman birbirinden bağımsız yeni nesneler üretir.

Ara yüz için FXML kullanıldı. Ara yüzün temelini oluşturan fxml adlı dosyamız var. Buton, yazı, resim gibi nesnelerin düzenli bir şekilde konumu, boyutu, hangi nesne ile ilişkili olduğu vb. burada yazılmıştır.

#### IV.Katkılar

Bu proje, nesne yönelimli programlama (OOP) temel kavramlarını anlamayı ve uygulamayı sağladı. Aşağıdaki OOP prensiplerini ve yapılarını öğrenme ve geliştirme fırsatı sundu. Nesne oluşturarak aralarındaki ilişkiyi sağlamayı, sınıflar arası bilgi aktarımını, metotları kullanmayı, Listler üzerinde işlemler yapmayı öğretti. Bu yapı, daha büyük ve karmaşık projelerin temellerini atmaya olanak tanıdı.

#### V.Sonuç

Oyun tam ekranda açılır. Ekranın alt kısmında kullanıcının kartları, en üstte bilgisayarın kartları üstü kapalı bir şekilde altışar tane görünür. Savaş kartlarının kullanıcı için görseli ve üst tarafında özellikleri yazılıdır. Ekranın ortasında bilgisayarın ve kullanıcının savaş için seçtiği kartların resmi ve özellikleri görünür. Kullanıcı, görsele tıklayarak savaş yaptırmak istediği karta tıklar. Sol ortada oyuncuların skor bilgileri vardır. Sağ ortada savaş yap butonu ile seçilen kartlar savaş yaptırılarak diğer adıma geçilir. Hamleler bittiğinde kazanan taraf ekranda kupa ile ilan edilir. Diğer adıma geçememesi için savaş yap butonu aktifliği kalker

ve rengi soluklaşır. ESC tuşu ile oyundan çıkılabilir.

<https://muhammedtalhacevik.medium.com/javada-nesne-y%C3%B6nelimli-programlama-oop-by-muhammed-talha-%C3%A7evik-e718f9e7436e>

[https://docs.oracle.com/javafx/2/get\\_started/fxml\\_tutorial.htm](https://docs.oracle.com/javafx/2/get_started/fxml_tutorial.htm)

## VI.Kaynakça

[https://www.youtube.com/watch?v=WDaXpDtYk3E&list=PLrzWQu7Ajpi26jZvP8JhEJgFPFEj\\_fojO](https://www.youtube.com/watch?v=WDaXpDtYk3E&list=PLrzWQu7Ajpi26jZvP8JhEJgFPFEj_fojO)

<https://examples.javacodegeeks.com/java-development/desktop-java/javafx/fxml/javafx-fxml-controller-example/>

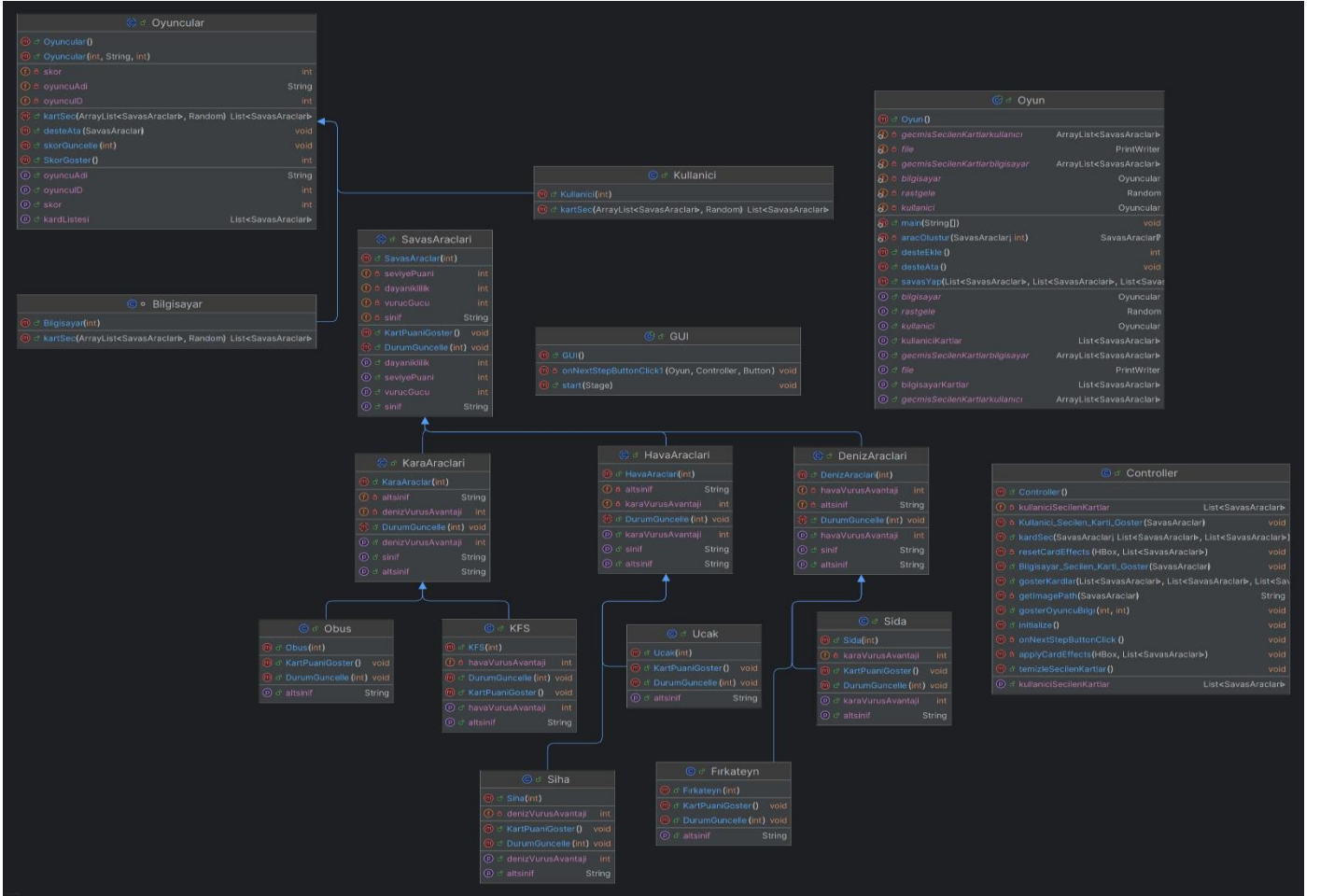
## VII.Tablo

Hava Araçları	Dayanıklılık	Vuruş Gücü	Avantajı / Değeri
Uçak	20	10	Kara / 10
Siha	15	10	Kara / 10, Deniz / 10

Kara Araçları	Dayanıklılık	Vuruş Gücü	Avantajı / Değeri
Obüs	20	10	Deniz / 5
KFS	10	10	Hava / 20, Deniz / 10

Deniz Araçları	Dayanıklılık	Vuruş Gücü	Avantajı / Değeri
Fırkateyn	25	10	Hava / 5
Sida	15	10	Kara / 10, Hava / 10

## VIII. UML Diyagramı



## IX. Oyundan Örnek Görsel

