**POKEMON KART OYUNU PROJESİ**

*Yağız Türer, Hüseyin Yılmaz*

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

[yagizturer@gmail.com](mailto:yagizturer@gmail.com), [ylmzhsyn98@gmail.com](mailto:ylmzhsyn98@gmail.com)

***Özet***

Bu projede encapsulation, inheritance, polymorphism ve abstraction yapılarından faydalanarak bir pokemon kart oyunu yapmamız istenmiştir. Projede arayüz oluşturulması ve bilgisayara karşı bilgisayar oynayabileceği gibi bilgisayara karşı insan da oynayabilmesi istenmiştir.

**Giriş**

Bu çalışmada amaç nesneye yönelik programlama mantığını, sınıf yapılarını kullanmayı, encapsulation, polymorphism, inheritance ve abstraction gibi nesneye programlama dillerinin önemli özelliklerini aktif bir şekilde kullanmayı öğrenmek, arayüz tasarlama hakkında deneyim edinmektir. Bizden her pokemon kartı için Pokemon sınıfından kalıtım alan ayrı bir sınıf tanımlamamız, Oyuncu sınıfından kalıtım alan Bilgisayar ve İnsan oyuncusu için ayrı birer sınıf oluşturmamız, tüm sınıflarda en az iki adet yapıcı fonksiyon tanımlamamız ve tüm sınıflarda gerekli nesneye yönelik programlama özelliklerinden faydalanmamız istendi. Programlama dili olarak C++ veya Java arasında tercih yapmamız belirtildi. Biz bu projede Java dilini kullanmayı tercih ettik ve arayüz için Swing kütüphanesini kullandık.Oyunu, insan müdahalesi olmadan bilgisayar bilgisayara karşı oynayabildiği gibi, insan bilgisayara karşı da oynayabilmektedir. Ayrıca istenildiği taktirde tüm kartlar açık şekilde oynanabilmektedir.

**Yöntem**

Öncelikle bir frame oluşturularak programın frame kapatılana kadar çalışması sağlandı. Bu frame’e bir label nesnesi kullanılarak arkaplan resmi eklendi ve bu frame objesi oluşturulan oyun objesine aktarıldı. Main classında tüm butonların ve yazıları ekranda gösterecek labelların objeleri baştan tanımlandı. Hepsinin koordinat ve boyut bilgileri önceden elle girildi. Program başladığında yalnızca iki buton gösterilmekte ve oyuncu2’nin insan mı yoksa bilgisayar mı olacağı seçilmektedir. Oyuncu2 bilgisayar olacağı takdirde kartlariGor değişkeni false’a eşitlenerek oyuncu2’nin elindeki kartların arkasının dönük olması gerektiği bildirilmektedir. Benzer şekilde, oyuncu2 insan olacaksa kartlariGor değişkeninin değeri true yapılarak elindeki kartların görülmesi gerektiği bildirilmektedir.

Daha sonra 10 adet pokemon kartı tanımladık. Bunların hepsini bir arrayliste aktardık. Böylece ortadaki kartların sürekli takibi yapılabildi. Ayrıca iki oyuncu için de birer adet arraylist oluşlturularak ellerindeki kartlar yine listede saklandı. Kart dağıtımı yapılırken üç defa iki oyuncuya da ortadaki kart listesinden birer adet rastgele kart verilmektedir. En son bu verilen üçer kart ekranda oyuncuların önlerinde sergilenmekte, geriye kalan dört kart da ortada gösterilmektedir. Her el başında ve sonunda ekrandaki kartların düzgün gösterimini sağlamak için bir kartlariSifirla metodu tanımladık.Bu metot çalıştırıldığında oyuncuların elindeki kartlar duruma göre arkası veya dönük olacak şekilde önlerine yerleştirilmekte, ortadaki kartlar da sayısına göre belirmektedir.

Sırası gelen oyuncunun kart seçilmesi istendiğinde eğer oyuncu bilgisayarsa random bir indis atanarak hangikartı oynayacağı belirlenmektedir. Eğer oyuncu insansa oyuncu2’nin önündeki kartlara actionListener eklenerek seçilen kartın indisi döndürülmektedir. Bu döndürülen indislerdeki kartlar arkası dönük halde öne çıkartılarak açılmakta, seçtiği kartın puanı yüksek olan oyuncuya 5 puan eklenmekte, eğer hasar puanları eşitse iki oyuncuya da bir puan eklemesi yapılmamaktadır. Daha sonra iki oyuncudan da ortadan birer kart seçmesi istenmektedir. Oyuncu bilgisayarsa yine rastgele bir indis döndürülmekte, eğer insana ortadaki kartlara actionListener eklenerek seçtiği kartın indisi döndürülmektedir.

Seçilen kartlar ortadaki kartların listesinden silinerek oyuncuların kart listesine eklenmekte ve kartlariSifirla metodu çağırılara gösterilmektedir. Ayrıca oyuncuların puanları her el başında ve sonunda güncellenerek gösterilmektedir. Oyunun sonunda puanı yüksek olan oyuncunun adı ekrana yazdırılmaktadır. İki oyuncunun puanı eşit olduğu takdirde oyun berabere bitmektedir. Oyun bir döngü içinde oynandığı için Main classının Thread classından kalıtım alması sağlanmıştır ve oyunun oynandığı metot run metodu olarak tanımlanmıştır. Ayrıca her hamlede ortadaki yazı güncellenerek hangi oyuncunun kart seçme sırası geldiği ekranda gösterilmektedir.

Butonlara resim eklemek için File objeleri ImageIcon nesnelerine aktarıldı ve setIcon metodu kullanılarak buton şekillendirildi, daha sonra kartların önü veya arkası çevrilirken yine aynı yöntem kullanıldı. Butonların veya yazıların ekranda görünüp görünmemesi setVisible metoduyla kontrol edildi. Bütün swing objeleri setBounds metoduyla istenen yere yerleştirildi. Programa gerekli yerlerde bekleme süresi eklemek için Thread classına bağlı sleep metodundan faydalandık.

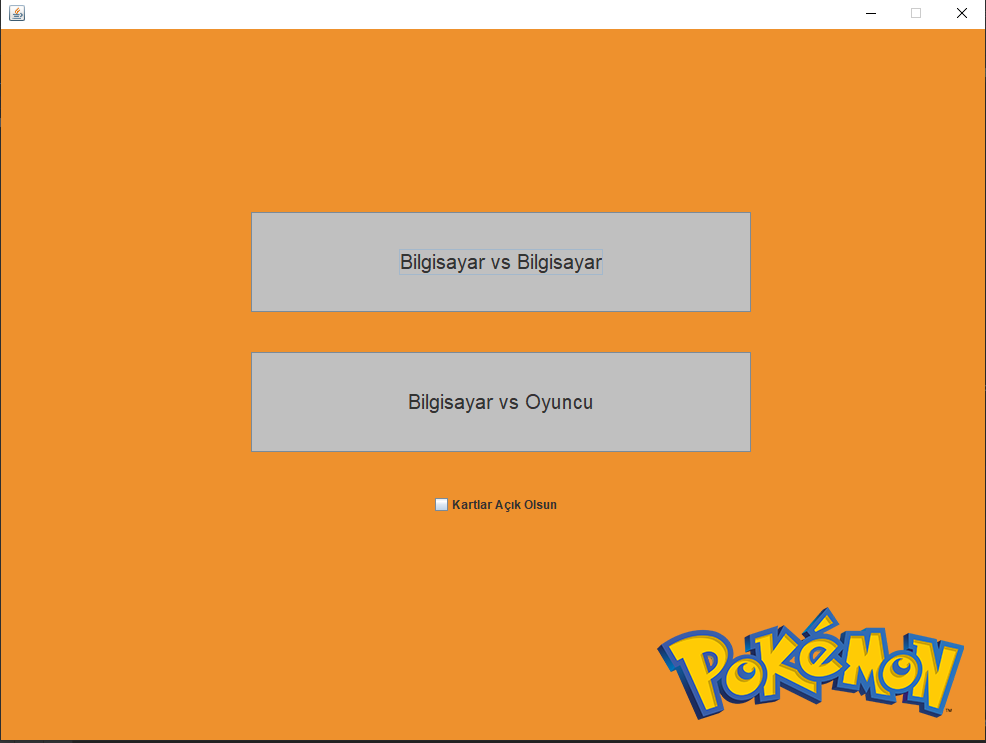
Tüm classlar için ikişer adet yapıcı fonksiyon tanımlandı, hepsinde birisi parametre alırken diğeri default değerleri atamaktadır. Gerekli tüm değişkenler için set get metodları tanımlanarak değerlerinin okunması ve yeni değer ataması sağlanmıştır.Oyuncu classı abstract olarak tanımlandı, böylece kart seçme metodları rahatlıkla override edildi.

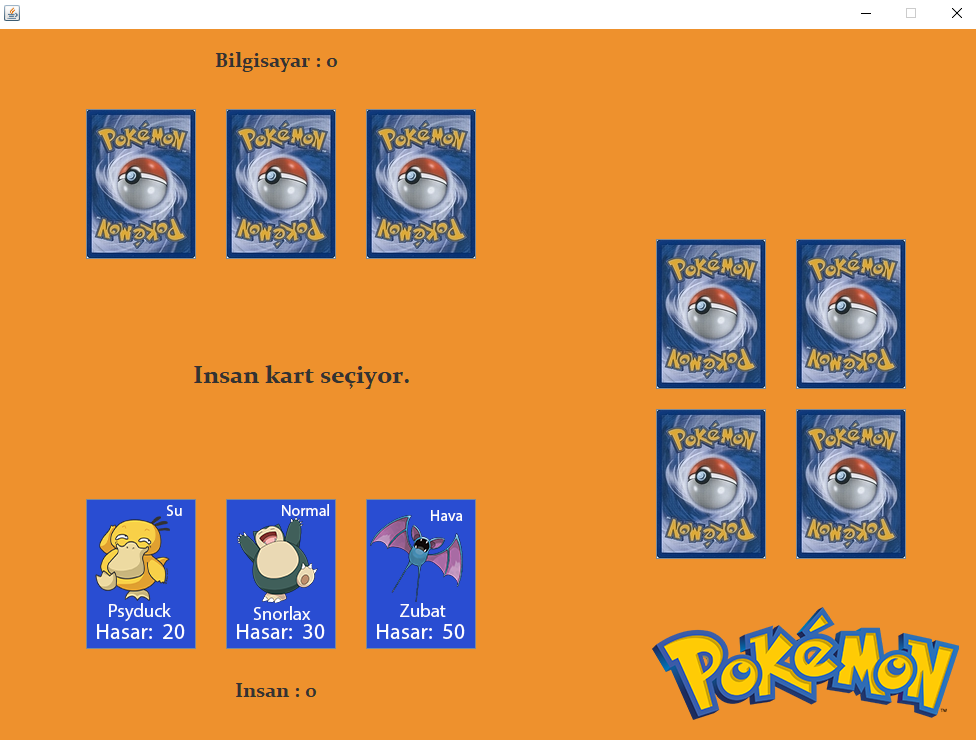
**Sonuçlar**

Tüm bu işlemler sonucunda nesneye yönelik programlama özellikleri kullanılarak arayüze sahip bir kart oyunu tasarlandı. Çeşitli testler yapılarak oyunun runtime hataları vermediği tespit edildi. Gerekli yerlere muhtemel hatalar için exception tanımlamaları eklendi. Program Eclipse ve IntelliJ’de ayrı ayrı compile edilerek hatasız çalıştığı doğrulandı.

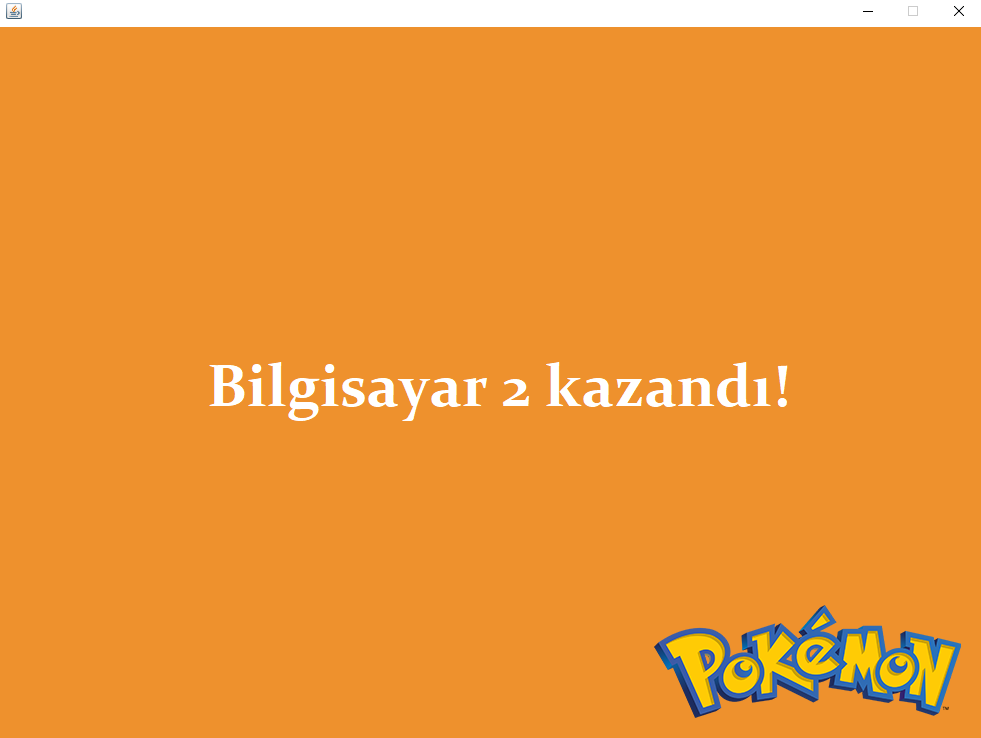
İki oyuncu da oyuna 0 puan ile başlamaktadır, her el sonunda seçtiği kartın hasar puanı yüksek olan oyuncuya 5 puan eklenmektedir, oyunu daha yüksek puana sahip oyuncu kazanmaktadır. İki oyuncunun puanı eşit olduğu takdirde oyun berabere bitmektedir.

**Çıktılar**

****

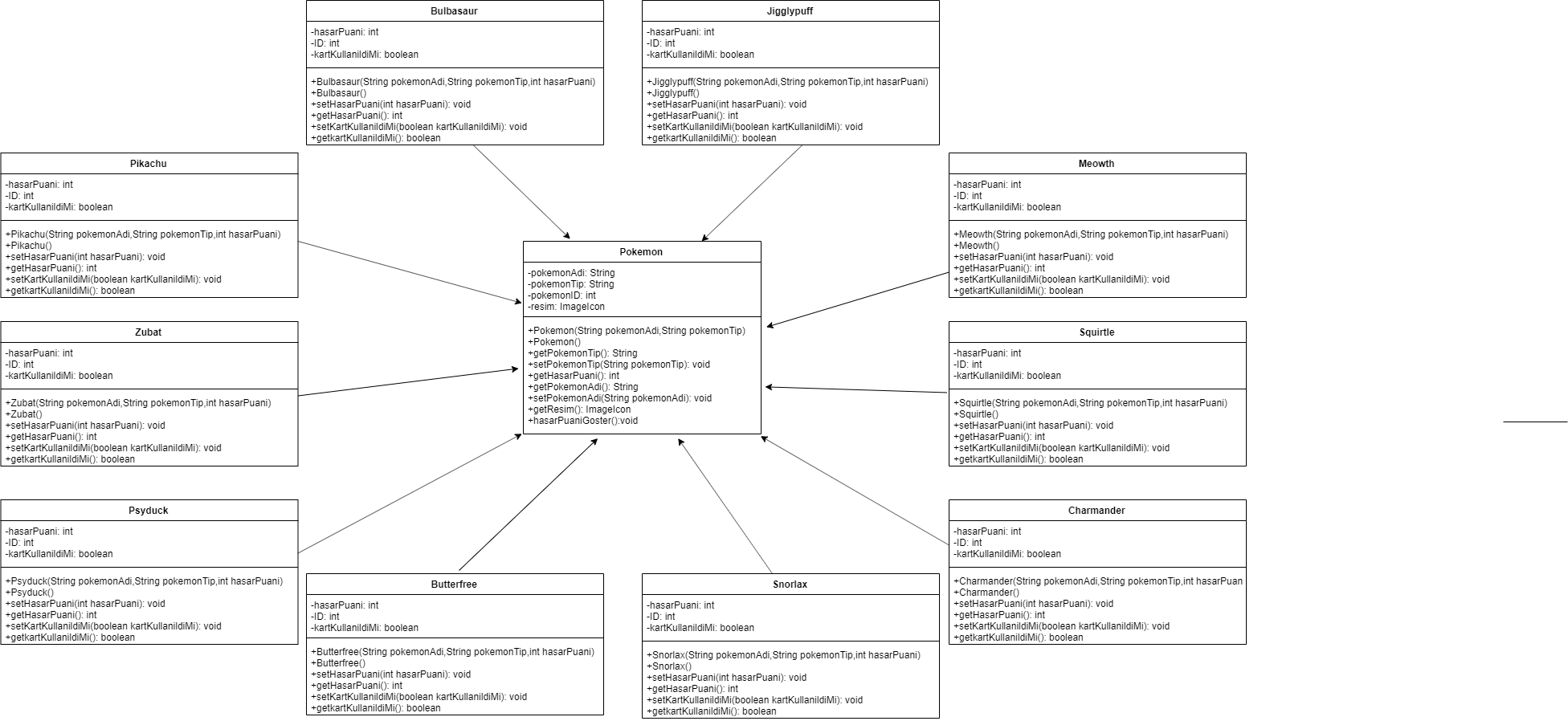
****

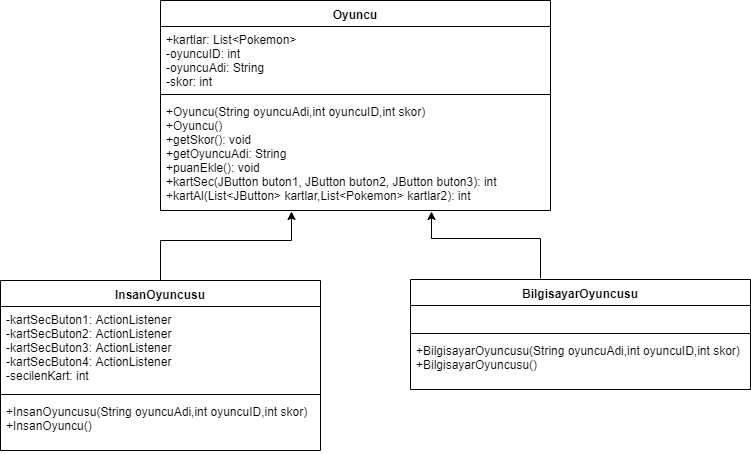
****

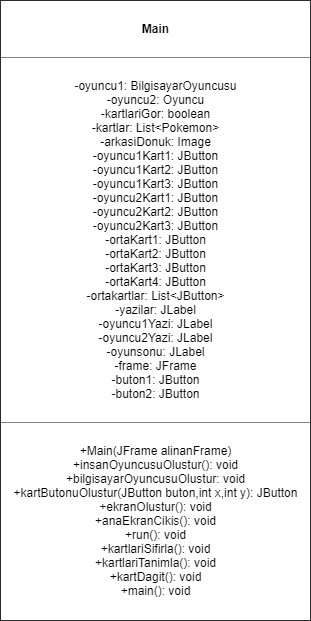
****

****

**Diyagramlar**

****

****

****

**Kaynakça**

Ahmet Sayar Nesneye Yönelik Programlama Slaytları

<https://stackoverflow.com/questions/18031704/jframe-how-to-disable-window-resizing>

<https://www.javatpoint.com/java-jbutton>

<https://www.javatpoint.com/java-jlabel>

<https://www.javatpoint.com/java-jcheckbox>

<https://www.javatpoint.com/java-jframe>

<https://www.geeksforgeeks.org/polymorphism-in-java/>

<https://www.geeksforgeeks.org/encapsulation-in-java/>

<https://www.geeksforgeeks.org/inheritance-in-java/>

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

<https://www.tutorialspoint.com/swing/swing_imageicon.htm>

<https://www.javatpoint.com/java-actionlistener>

<http://www.java2s.com/Code/JavaAPI/java.awt/newFontStringnameintstyleintsize.htm>

<https://www.javatpoint.com/sleep()-method>

<https://www.geeksforgeeks.org/multithreading-in-java/>

<https://www.javatpoint.com/creating-thread>

<https://www.javatpoint.com/java-jlabel>

<https://www.geeksforgeeks.org/throw-throws-java/>

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

<https://www.geeksforgeeks.org/file-class-in-java/>

<https://chortle.ccsu.edu/Java5/Notes/chap56/ch56_9.html>