**ARAÇ YÖNETİM SİSTEMİ**

*Erhan ÖZDOĞAN, Fırat KAPLAN*

Bilgisayar Mühendisliği Bölümü

Kocaeli Üniversitesi

ozerhan9@gmail.com , kpln977@gmail.com

Özet

Araç yönetim sistemi projesi on dört adet sınıftan meydana gelmektedir.

Projede bir araç sınıfı ve bu sınıfı miras alan deniz taşıtları kara taşıtları ve hava taşıtları sınıfları vardır ve yine bu sınıfları miras alan araba, bisiklet, gemi, uçak sınıfları bulunmaktadır aynı zamanda deniz taşıtları sınıfını miras alan ve bir arayüzü implements eden bir uçan gemi sınıfı bulunmaktadır. Bu sınıflar ile kullanıcıya beş adet taşıt üzerinde işlem yapabilmesini sağlamaktadır. Bütün taşıtların markası, rengi ,hızı, yolcu sayısı, teker sayısı, fiyatı, üretim yılı, yakıt türü özellikleri vardır bunlardan tekerlek sayısı ve yakıt türü hariç araç sınıfında veriler değişkenlere atanır.Diğer özellikler ise alt sınıflarda set metodları yardımıyla değişkenlere atanır. Program başlatıldığında kullanıcıya bir arayüz sunulur bu arayüzde önce yapmak istediği işlemi seçer daha sonra veri ekleyecekse veya veri listeleyecekse hangi araç türünde veri ekleyip veya listeleyeceğini seçer. Veri ekleme seçildiğinde istenen araç türüne göre veriler girilir daha sonra aracın hızının değiştirilip değiştirilmeyeceği sorulur. Veri güncelle kısmı seçilirse kullanıcıdan aracın ıd si girilmesi istenir ve id bulunursa o id deki veriler kullanıcının girdikleri ile değiştirilir. Veri Girişinden sonra her kayıt Kayıt.txt dosyasında tutulur ve program her çalıştığında buradaki veriler çekilerek işlem yapılır.

**1.Giriş**

Araç yönetim sistemi genel olarak kullanıcıya sunulan bir arayüzde veri girişi veri listele ve veri güncelle fonksiyonlarından oluşmaktadır girilen veriler bir txt dosyasında tutulur diğer fonksiyonlar çalıştırılmak istendiğinde bu txt dosyasından faydalanılır. Projede gerçekleştirilen kısımlar:

* Veri Girişi
* Txt den verileri okuyup listeleme
* Txt deki verileri güncelleme

Bu projede amaç bir araç şirketi için araçlarının kayıtlarını tutabilmesini ve istediği zaman güncellenmesini sağlamaktır. Aynı zamanda nesneye yönelik programlama tekniklerinin uygulamada kullanılarak kalıcılığının sağlanmasıdır.

2.Temel Bilgiler

Gezgin zeplin projesinde programlama dili olarak C dili ve geliştirme ortamı (IDE) olarak NetBeans ve Eclipse kullanılmıştır.

**2.1.JAVA Programlama Dili:** Sun Microsystems mühendislerinden James Gosling tarafından 1996 yılında geliştirilmeye başlanmış açık kodlu, nesneye yönelik, platformdan bağımsız, yüksek performanslı, çok işlevli, yüksek seviye, adım adım işletilen (interpreted) bir dildir.

**2.2.Geliştirme Ortamı (IDE):** IDE bilgisayar programcılarının hızlı ve rahat bir şekilde program geliştirebilmesini amaçlayan, geliştirme sürecini organize edebilen birçok araç ile birlikte geliştirme sürecinin verimli kullanılmasına katkıda bulunan araçların tamamını içerisinde barındıran bir yazılım türüdür

**3.Diğer Bölümler**

**3.1.Proje Bölümleri**

*3.1.1.Main Sınıfı*

Main sınıfında araçları oluşturmayı sağlayan otomobil bisiklet uçak gemi ve uçan gemi dizileri mevcuttur. Ve main sınıfında sadece programın çalışmasını sağlayan “public static void main” fonksiyonu bulunur ve bu fonksiyon programın çalşıtıran arayüz sınıfının yapıcısını çağırı ve program arayüz sınıfından başlar.

*3.1.2.Araç Sınıfı*

Bu sınıf diğer bütün sınıfların türediği süper sınıftır. Araçların marka, hız, yolcu sayısı, fiyat , üretim yılı, aracID, renk, teker sayısı, yakıt türü özelliklerini tutar. İki yapıcısı vardır birisi parametresiz yapıcıdır. Diğeri parametre olarak gelen özellikleri atayarak aracı oluşturan yapıcıdır. Aynı zamanda bütün özelliklerin get ve set metodlarını bulundurur. Bu sınıf soyut sınıftır bütün araçlarda olan hızlan yavaşla dur fonksiyonları soyut olarak tanımlanmıştır. Ayrıca soyut bir yazdır metodu bulunmaktadır bu metod aracı dosyaya yazıdırır.

*3.1.3.Kara Taşıtları Sınıfı*

Kara taşıtları sınıfı araç süper sınıfını extend etmektedi ve yazdır metodunu override etmediği için soyut olarak tanımlanmıştır bu sınıfta hızlan yavaşla ve dur metodları override edilmiştir.Bu metodlar kendine gelen parametreleri mevcut hıza ekler veya çıkarır dur metoduda hızı direkt sıfır yapar. Bu sınıfın parametreli ve parametresiz olarak iki yapıcısı bulunur. Parametreleri yapıcı bu sınıfı extend eden sınıflarda ortak olan özellikleri alır ve bu özelliklerle birlikte üst sınıfın yapıcısnı çağırır sadece ayrı olarak teker sayısını set metodunu çağırarak atar.Çünkü teker sayısı Araç sınıfından türeyen bütün sınıflarda ortak değildir.

*3.1.4.Otomobil Sınıfı*

Otomobil sınıfı kara taşıtları sınıfnı miras alır böylece araç sınfınıda miras almış olur. Bu sınıfta parametreli ve parametresiz iki yapıcı bulunur parametreli yapıcı otomobilin marka, hız, yolcu sayısı, fiyat, üretim yılı, tekerlek sayısı, yakıt türü, renk, ve otosayısı özelliklerini parametre olarak alır otosayı değişkeni o zamana kadarki oluşturulan arabaların sayısını tutmaktadır araç ıd si için gereklidir. Yapıcı daha sonra bu özellikler ile kara taşıtının yapıcısını çağırır ve ayrı olarak yakıt türünü set metodunu kullanarak atama yapar çünkü bisiklette yakıt türü yoktur. Ayrıca bu sınıfta yazdır metodu override edilir bu metod aracın bilgilerini dosyaya yazıdırır.

*3.1.5.Bisiklet Sınıfı*

Bisiklet sınıfı kara taşıtları sınıfnı miras alır böylece araç sınfınıda miras almış olur. Bu sınıfta parametreli ve parametresiz iki yapıcı bulunur parametreli yapıcı Bisikletin marka, hız, yolcu sayısı, fiyat, üretim yılı, tekerlek sayısı, renk, ve araç sayısı özelliklerini parametre olarak alır araç sayısı değişkeni o zamana kadarki oluşturulan bisikletlerin sayısını tutmaktadır araç ıd si için gereklidir. Yapıcı daha sonra bu özellikler ile kara taşıtının yapıcısını çağırır ve ayrı olarak yakıt türünü set metodunu kullanarak “Yok” ataması yapar çünkü bisiklette yakıt türü yoktur. Ayrıca bu sınıfta yazdır metodu override edilir bu metod aracın bilgilerini dosyaya yazıdırır.

*3.1.6.Hava Taşıtları Sınıfı*

Hava taşıtları sınıfı araç süper sınıfını extend etmektedir ve yazdır metodunu override etmediği için soyut olarak tanımlanmıştır bu sınıfta hızlan yavaşla ve dur metodları override edilmiştir.Bu metodlar kendine gelen parametreleri mevcut hıza ekler veya çıkarır dur metoduda hızı direkt sıfır yapar. Bu sınıfın parametreli ve parametresiz olarak iki yapıcısı bulunur. Parametreleri yapıcı bu sınıfı extend eden sınıflarda ortak olan özellikleri alır ve bu özelliklerle birlikte üst sınıfın yapıcısnı çağırır sadece ayrı olarak teker sayısını set metodunu çağırarak atar.Çünkü teker sayısı Araç sınıfından türeyen bütün sınıflarda ortak değildir.Ayrıca bu sınıfta aracın havada olup olmadığını tutan bir özellik ve bu özelliğin get ve set metodları bulunur bu özellik aracın hızı 100 den büyükse true değilse false değeri alır.

*3.1.7.Ucak Sınıfı*

Uçak sınıfı hava taşıtları sınıfnı miras alır böylece araç sınfınıda miras almış olur. Bu sınıfta parametreli ve parametresiz iki yapıcı bulunur parametreli yapıcı uçağın marka, hız, yolcu sayısı, fiyat, üretim yılı, tekerlek sayısı, yakıt türü, renk, ve araç sayısı özelliklerini parametre olarak alır araç sayısı değişkeni o zamana kadarki oluşturulan uçakların sayısını tutmaktadır araç ıd si için gereklidir. Yapıcı daha sonra bu özellikler ile hava taşıtının yapıcısını çağırır ve ayrı olarak teker sayısını set metodunu kullanarak atama yapar çünkü teker sayısı bütün sınıflarda ortak değildir Ayrıca bu sınıfta yazdır metodu override edilir bu metot aracın bilgilerini dosyaya yazdırır.

*3.1.8.Deniz Taşıtları Sınıfı*

Deniz taşıtları sınıfı araç süper sınıfını extend etmektedir ve yazdır metodunu override etmediği için soyut olarak tanımlanmıştır bu sınıfta hızlan yavaşla ve dur metodları override edilmiştir. Bu metodlar kendine gelen parametreleri mevcut hıza ekler veya çıkarır dur metoduda hızı direkt sıfır yapar. Bu sınıfın parametreli ve parametresiz olarak iki yapıcısı bulunur. Parametreleri yapıcı bu sınıfı extend eden sınıflarda ortak olan özellikleri alır ve bu özelliklerle birlikte üst sınıfın yapıcısnı çağırır sadece ayrı olarak teker sayısını ve yakıt türünü set metodunu çağırarak atar. Çünkü teker sayısı ve yakıt türü Araç sınıfından türeyen bütün sınıflarda ortak değildir.

*3.1.9.Gemi Sınıfı*

Gemi sınıfı deniz taşıtları sınıfnı miras alır böylece araç sınfınıda miras almış olur. Bu sınıfta parametreli ve parametresiz iki yapıcı bulunur parametreli yapıcı geminin marka, hız, yolcu sayısı, fiyat, üretim yılı,, yakıt türü, renk, ve araç sayısı özelliklerini parametre olarak alır araç sayısı değişkeni o zamana kadarki oluşturulan gemilerin sayısını tutmaktadır araç ıd si için gereklidir. Yapıcı daha sonra bu özellikler ile gemi taşıtının yapıcısını çağırır, Araç Id si için süper sınıfa gönderilen string “Gemi”+araç sayısı şeklindedir. Ayrıca bu sınıfta yazdır metodu override edilir bu metod aracın bilgilerini dosyaya yazdırır.

*3.1.10.Uçan Gemi Sınıfı*

Uçan gemi sınıfı deniz taşıtları sınıfını miras alır. Javada birden fazla sınıf miras alınamadığı için arayüz kullanılarak çoklu kalıtım sağlanmıştır bunun içinde hava arayüz sınıfı arayüz olarak tanımlanmış ve uçan gemi sınıfndan implement edimiştir. Deniz taşıtları sınıfını miras aldığı için gemi yapıcısı gerekli özelliklerle birlik deniz taşıtları sınıfının yapıcısını çağırır.Hava arayüzünü implement ettiği için hızlan yavaşla ve dur metodları bulunmaktadır. Ayrıca dosyaya yazdırılması için yazdır metodu override edilir.

*3.1.11.Arayüz Sınıfı*

Arayüz sınıfı kullanıcıya bir arayüz sunan aynı zamanda programın diğer sınıfları kullanarak çalıştığı kısımdır. Bu kısımda JFrame,JPanel,JTextField,JLabel lar kullanılarak kullanıcıya bir arayüz oluşturulmuş ve program başladığın kullanıcıya ilk önce seçim yaptırılmıştır kullanıcı veri girişi seçtiyse önce dosya okunmuş Aracın seri numarası için o ana kadarki araçların sayısı bulunmuş ve kullanıcıdan JTextField kullanılarak gerekli bilgiler alınmış ve bu bilgiler ile gerekli sınıfların yapıcıları kullanılarak araçlar oluşturulmuş dosyaya yazmdanda aracın hzının arttrılıp azaltılmasını olanak veren bir kısım kullanıcıya sunulmuştur ve tekrar dosyaya yazılmıştır kullanıcı veri girişi seçtikten sonra oluşturmak istediği araç türüde seçtirilmiştir. Eğer kullanıcı araç listele seçerse daha sonra hangi araç türünü listelemek istediğide sorulmuş ve ona göre txt dosyası okunup bir diziye atılmış ve seçilen araç türü ile başlayan stringler ekrana yazılmıştır.

3.2.UML Diyagramı

Bkz.:Şekil 1 (Sayfa 5)

4.Sonuçlar

Projede

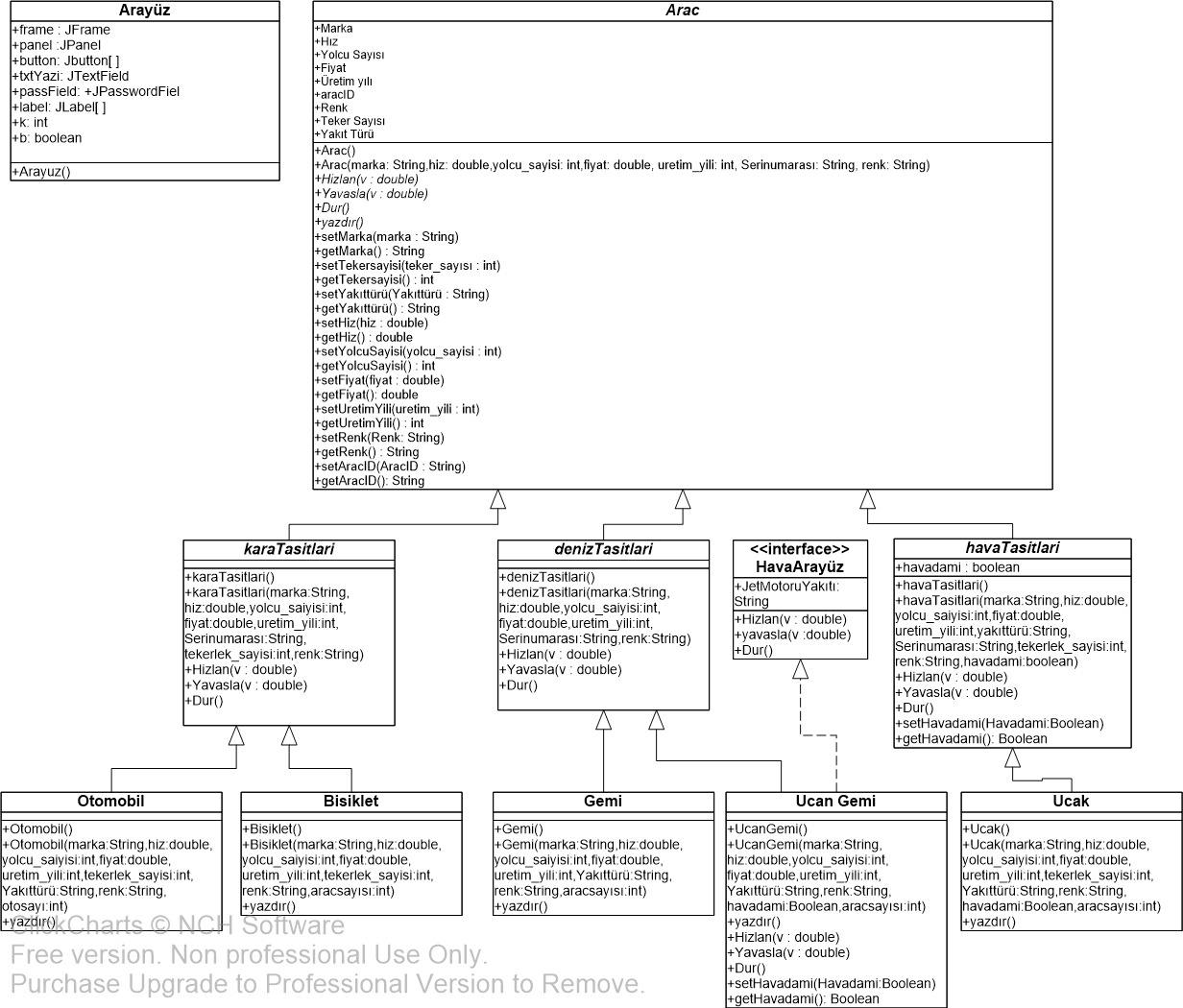
* Kullanıcıya bir arayüz sunulmuştur.
* Arayüzde kullanıcının seçimine göre veri ekleme, listeleme ve güncelleme fonksiyonları başarıyla çalıştırılmaktadır.
* Veri ekledikten sonra hızlanma , yavaşlama , durma fonksiyonlarını kullanmak için kullanıcıya imkan verilmiştir.
* Kullanıcının yapmak istediği işlem sonrası txt dosyası başarıyla güncellenebilmektedir.
* Arayüzde işlem yaptıktan sonra ana menüye dönme fonksiyonu sadece veri listele işleminden sonra kullanılabilmektedir.
* Nesneye yönelik programlama teknikleri kullanılmıştır.
* Kullanıcı bir işlemi yaptıktan sonra diğer bir işlem için programı kapatmalıdı.

***5.Kaynakça***

[1] Kirazlı, M. ve Tanrıverdioğlu, S. “JAVA 8”, 2107

[2] “Java Programming Tutorial GUI”, https://www.ntu.edu.sg/home/ehchua/programming/java/J4a\_GUI.html

[3]“SıfırdanJava”, https://www.youtube.com/watch?v=uAJrC1wpSPk&list=PLHfYetw\_BGF-Gm\_MsqKApw5nHPuHsytr3&index=43

****

Şekil 1: UML diyagramı