**ARAÇ ALIŞ-SATIŞ PROGRAMI**

Ali Recep KARACA, Mehmet Fırat KÖMÜRCÜ

Bilgisayar Mühendisliği Bölümü

[recepkaraca@yandex.com](mailto:recepkaraca@yandex.com) , [mehmetfiratkomurcu@hotmail.com](mailto:mehmetfiratkomurcu@hotmail.com)

*Bu proje; kullanıcının bir arayüz üzerinden ilan, araba, vites türü, yakıt türü, renk, şehir tabloları içerisine veri girmesine, güncellemesine ve silmesine olanak sağlamaktadır. Ayrıca bunların yanında bu ilamlar arasında ilanın adına, araç markasına, modeline, rengine, vites türüne, şehre, ilan tarihine, araç kilometre ve fiyatına göre filtreleme de yapılabilmektedir. Ek olarak filtreleme ekranı içerisinde; marka, modele göre sıralama ve kilometre ve fiyata göre de artan-azalan şeklinde bir sıralama da yapılabilmektedir. Bu özellikler neticesinde; bu projede bir araç alış-satış programı gerçeklenmesi amaçlanmıştır.*

**1. Problemin Tanımlanması**

Bu projede çözümlenmesi gereken sorun; “tbl\_ilan”, “tbl\_araba”, “tbl\_yakitturu”, “tbl\_vitesturu”, “tbl\_renk”, “tbl\_sehir” tabloları içerisine bir arayüz yardımıyla veri girişi sağlamak, verileri güncellemek ve verileri silebilmektir. Bunların yanında, proje içerisinde yine, bir arayüz içerisinde tüm kayıtlar ilan adı, marka, model, renk vites türü, yakıt türü, şehir, tarih, kilometre ve fiyat gibi ölçütlere göre sıralanabilmeli ve, marka veya modele göre sıralamak ve, kilometre veya fiyata göre göre artan-azalan sıralaması yapılabilmesi gerekmektedir.

**2. Giriş**

Verilen projenin konusu; bir arayüz içerisinden, projede bizlerden istenen tablolar içerisine veri girişi, veri güncellemesi ve veri silme işlemleri yapılması ve bunların yanı sıra bir filtreleme modülü geliştirilerek girilen ilanlar arasından kullanıcının istediği ölçütlere göre ilan görüntülenmesini sağlamaktır.

Verilen projenin amacı; proje içerisinde bizlerden mysql veri tabanı kullanılması istendiğinden dolayı, öğrencilerin mysql veri tabanı üzerinde deneyim kazanmalarını ve öğrenilen teorik derslerin pratik olarak gerçekleştirilmesini sağlamaktır. Ayrıca proje içerisinde bizlerden bir arayüz kullanılması da istenildiğinden dolayı, öğrencilerin, bir masaüstü programı ile bir veri tabanı arasında bağlantı kurma noktasında da deneyim kazanmaları amaçlanmıştır.

**3. Temel Bilgiler**

Projeyi gerçekleştirme aşamasında Java programlama dili kullanılmıştır. Ayrıca; yapmış olduğumuz proje; Java programlama dili kullanılarak nesneye yönelik programlama mantığında oluşturulmuştur.

Yazılan bu program Windows 10 üzerindeki NetBeans 8.2 derleyicisi üzerinde derlenmiş olup, derleme aşamasında herhangi bir hatayla karşılaşılmamıştır. Ayrıca derlenen bu program yine Windows 10 üzerinde çalıştırılmış ve programın çalışması sırasında yine herhangi bir hatayla karşılaşılmamıştır.

**4. Genel Yapı**

Projede ilk olarak; bir combo box içerisine veri tabanındaki tüm tabloların isimleri eklenmiştir. Bu tablolar; ilan, araba, yakıt türü, vites türü, renk ve şehir olmak üzere altı adettir. Kullanıcı veri girişi, veri güncellemesi veya silme işlemi yapması için ilk olarak bu combo box içerisinden işlem yapacağı tabloyu seçmelidir.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Ekle” butonuna basarsa ve seçilen tablo yakıt türü, vites türü, renk veya şehir tablolarından herhangi biri ise; seçilen tablolar haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından, kullanıcının önüne gelen panel içerisinde bir text box yardımıyla kullanıcıdan tablo türüne göre yeni veri girişi istenir ve “Kaydet” butonu yardımıyla girilen verinin veri tabanına girilmesi sağlanmaktadır. “Kaydet” butonu içerisinde ilk olarak; işlem yapılan tablonun adı alınır. Ardından, ekleme işlemine uygun olan sql sorgusu belirlenen tablo adına göre düzenlenir, daha sonra veri tabanı ile bağlantı kurulur ve belirlenen sql sorgusu çalıştırılarak yakıt türü, vites türü, renk veya şehir tabloları içerisine veri eklenmesi sağlanmaktadır.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Ekle” butonuna basarsa ve seçilen tablo araba tablosu ise; araba tablosu haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından, kullanıcının önüne gelen panel içerisinden marka, model, vites türü, yakıt türü, ve renk bilgileri kullanıcı tarafından istenir. Burada; vites türü, yakıt türü ve renk bilgisi bir combo box içerinden seçilmektedir ve bu combo box içerisinde veriler, veri tabanındaki güncel veriler çekilerek elde edilmektedir. Ardından “Kaydet” butonu yardımıyla kullanıcı tarafından girilen verilerin veri tabanına girilmesi sağlanmaktadır. “Kaydet” butonu içerisinde, kullanıcının belirlemiş olduğu girdilere göre bir sql sorgusu düzenlenmektedir. Ardından bu sql komutu çalıştırılarak kullanıcının girmiş olduğu veriler veri tabanına kayıt edilmektedir.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Ekle” butonuna basarsa ve seçilen ilan tablosu ise; ilan tablosu haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından, kullanıcının önüne gelen panel içerisinden ilan adı, ilan fiyatı, araç kilometre, ilan tarihi ilan verilecek olan araç ve şehir bilgileri kullanıcı tarafından istenir. Burada; araç ve şehir bilgisi bir combo box içerinden seçilmektedir ve bu combo box içerisinde veriler, veri tabanındaki güncel veriler çekilerek elde edilmektedir. Araç combo box’ı içerisinde “Araç” tablosundaki tüm veriler ve bu tabloya bağlı olan diğer tablolardan veriler çekilerek her araç için toplu bir şekilde kullanıcıya gösterilir. Ardından “Kaydet” butonu yardımıyla kullanıcı tarafından girilen verilerin veri tabanına girilmesi sağlanmaktadır. “Kaydet” butonu içerisinde, kullanıcının belirlemiş olduğu girdilere göre bir sql sorgusu düzenlenmektedir. Ardından bu sql komutu çalıştırılarak kullanıcının girmiş olduğu veriler veri tabanına kayıt edilmektedir.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Sil” butonuna basarsa ve seçilen tablo yakıt türü, vites türü, renk veya şehir tablolarından herhangi biri ise; seçilen tablolar haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından pencere içerisindeki tabloya, seçilen tablonun bilgisine göre id ve veriler eklenir. Ayrıca seçilen tablonun isim bilgisi de bir değişkene aktarılır. Daha sonra, kullanıcı tablo üzerinden silmek istediği veriye tıklayıp “Sil” butonuna tıklaması üzerine; seçilen tablo türüne göre, kullanıcının seçmiş olduğu veri, veri tabanından silinmiş olmaktadır.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Sil” butonuna basarsa ve seçilen tablo araba tablosu ise; araba tablosu haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından pencere içerisindeki tabloya, araba tablosunda bulunan marka, model, vites türü, yakıt türü ve renk verileri eklenir. Daha sonra, kullanıcı tablo üzerinden silmek istediği veriye tıklayıp “Sil” butonuna tıklaması üzerine; kullanıcının seçmiş olduğu veri, veri tabanından silinmiş olmaktadır.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Sil” butonuna basarsa ve seçilen tablo ilan tablosu ise; ilan tablosu haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından pencere içerisindeki tabloya, ilan tablosunda bulunan ad, fiyat, kilometre, tarih, araba ve şehir verileri eklenir. Daha sonra, kullanıcı tablo üzerinden silmek istediği veriye tıklayıp “Sil” butonuna tıklaması üzerine; kullanıcının seçmiş olduğu veri, veri tabanından silinmiş olmaktadır.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Güncelle” butonuna basarsa ve seçilen tablo yakıt türü, vites türü, renk veya şehir tablolarından herhangi biri ise; seçilen tablolar haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından pencere içerisindeki tabloya, seçilen tablonun bilgisine göre id ve veriler eklenir. Ayrıca seçilen tablonun isim bilgisi de bir değişkene aktarılır. Daha sonra, kullanıcı tablo üzerinden güncellemek istediği veriye tıklayıp içerik text box’ı içerisine yeni bir değer girerek “Güncelle” butonuna tıklaması üzerine; seçilen tablo türüne göre, kullanıcının seçmiş olduğu veri, veri tabanı üzerinde güncellenmiş olmaktadır.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Güncelle” butonuna basarsa ve seçilen tablo araba tablosu ise; araba tablosu haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından pencere içerisindeki tabloya, araba tablosunda bulunan marka, model, vites türü, yakıt türü ve renk verileri eklenir. Daha sonra, kullanıcı tablo üzerinden güncellemek istediği veriye tıklayıp “Verileri Getir” butonuna tıklaması ardından seçilen veri düzenlenebilir hale gelmektedir. Kullanıcı belirtmiş olduğu veriye ait bilgileri değiştirdikten sonra “Güncelle” butonuna basarak veriyi, veri tabanı üzerinden değiştirebilmektedir.

Tablo seçimi yapıldıktan sonra, eğer kullanıcı “Kayıt Güncelle” butonuna basarsa ve seçilen tablo ilan tablosu ise; ilan tablosu haricindeki tüm tabloların pencereleri görünmez hale getirilir. Ardından pencere içerisindeki tabloya, ilan tablosunda bulunan ad, fiyat, kilometre, tarih, araba ve şehir verileri eklenir. Daha sonra, kullanıcı tablo üzerinden güncellemek istediği veriye tıklayıp “Verileri Getir” butonuna tıklaması ardından seçilen veri düzenlenebilir hale gelmektedir. Kullanıcı belirtmiş olduğu veriye ait bilgileri değiştirdikten sonra “Güncelle” butonuna basarak veriyi, veri tabanı üzerinden değiştirebilmektedir.

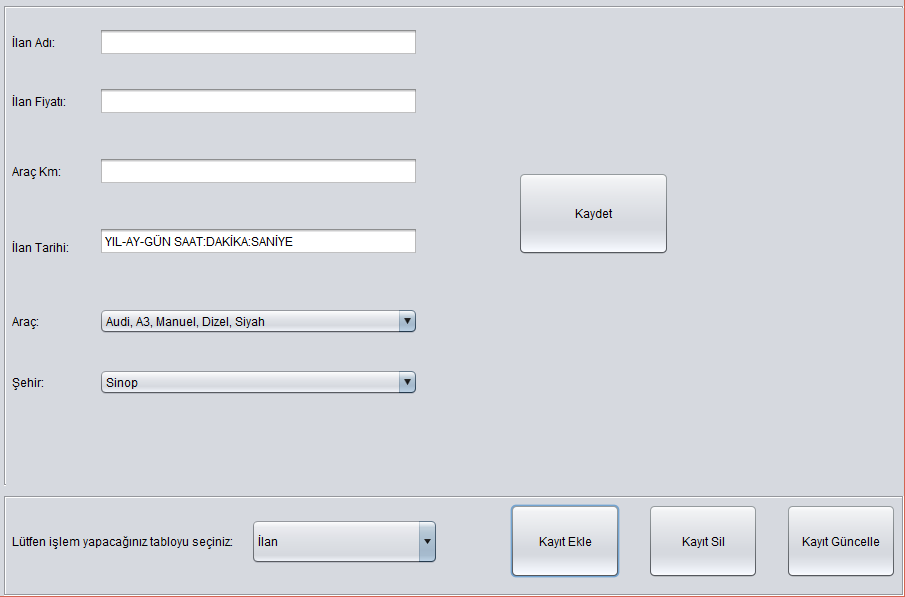
Eğer kullanıcı bu veri ekle, güncelle veya sil işlemlerini yapmak yerine mevcut ilanlar üzerinde filtreleme işlemleri yapmak isterse, arayüz içerisinde sol tarafta bulunan ilan adı, model, renk, vites türü, yakıt türü, şehir, tarih, kilometre, fiyat alanlarından seçimler yaparak filtreleme işlemini gerçekleştirebilmektedir. Bu işlemler sırasında kullanıcı filtrelemek istediği özelliklerden sadece birini seçebileceği gibi, isterse filtrelemek istediği tüm verileri farklı kombinasyonlar halinde seçerek te ilanlar arasında filtreleme işlemi de gerçekleştirebilmektedir. Ayrıca filtreleme sırasında, marka veya modele göre a’dan z’ye sıralama veya z’den a’ya göre sıralama yapılabileceği gibi, kilometre veya fiyata göre de artan veya azalan bir şekilde sıralama da yapılabilmektedir. Kullanıcı, filtrelemek istediği tüm özellikleri seçtikten sonra “Ara” butonuna basması halinde, sol tarafta bir tablo içerisinde tüm veriler seçilen kriterler ölçüsünde kullanıcıya gösterilmektedir.

Tüm bu işlemler esnasında, veri tabanını sürekli meşgul tutmak yerine, her işlem sırasında veri tabanına bağlantı kurulması sağlanmış olup, her işlem bitişinde de veri tabanı bağlantısı kesilerek veri tabanı üzerindeki meşguliyetin kaldırılması amaçlanmıştır. Bu sayede veri tabanından sadece çok kısa süreler için bağlantı kurularak ilgili işlemlerin yapılması sağlanmıştır.

**5. Proje Hakkında Diğer Veriler**

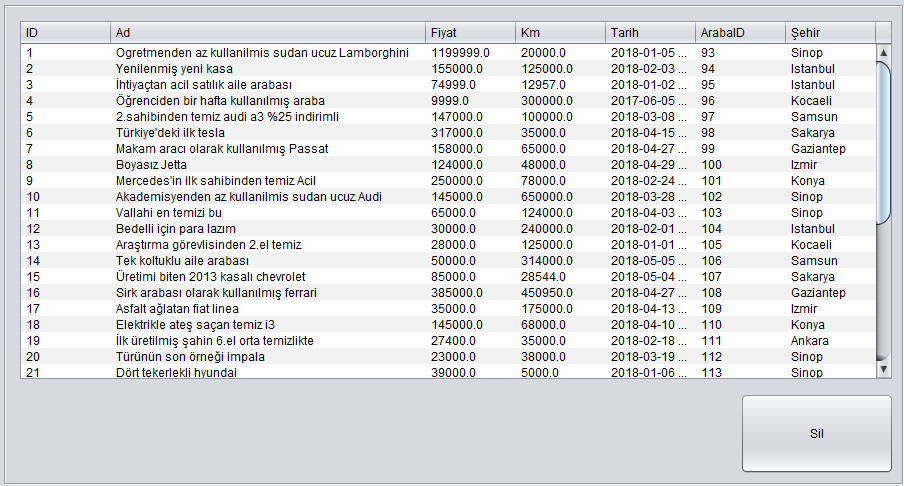
**5.1 Programın Çalışması Sırasından Bazı Görseller**

Kayıt eklenme sırasından bir görüntü:



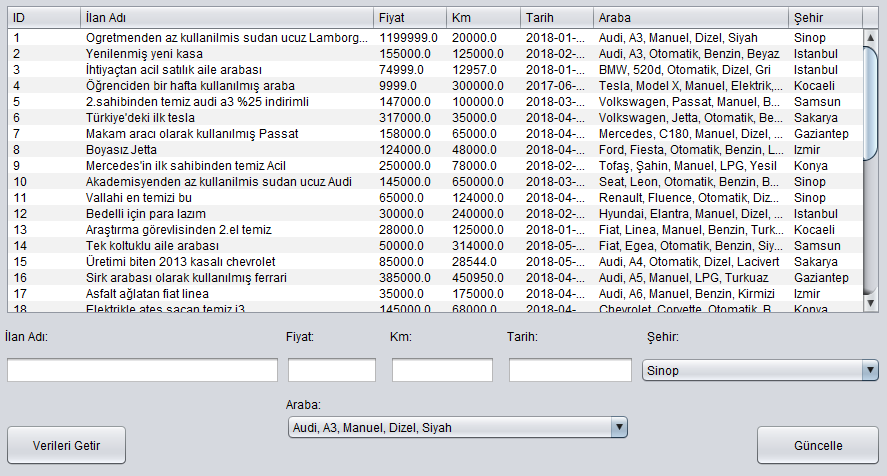
*Şekil 1:* Kayıt Ekleme

Verilerin silinmesi anından bir görüntü:



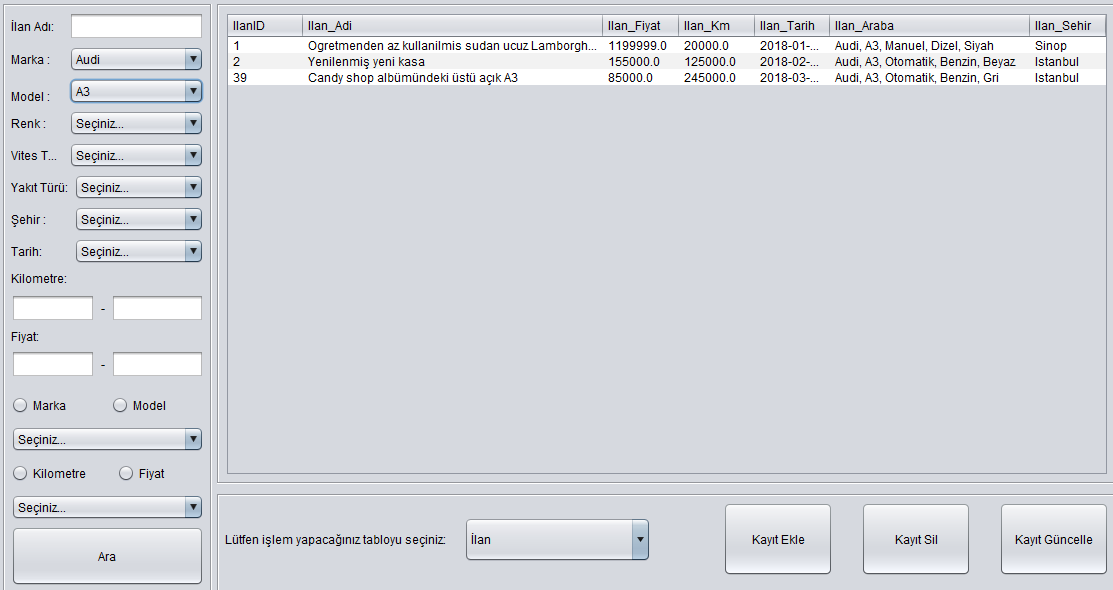
*Şekil 2:* Veri Silinmesi

Verilerin güncellenmesi sırasında bir görüntü:



*Şekil 3*: Verilerin Güncellenmesi

Filtreleme işlemi sırasından bir görüntü:



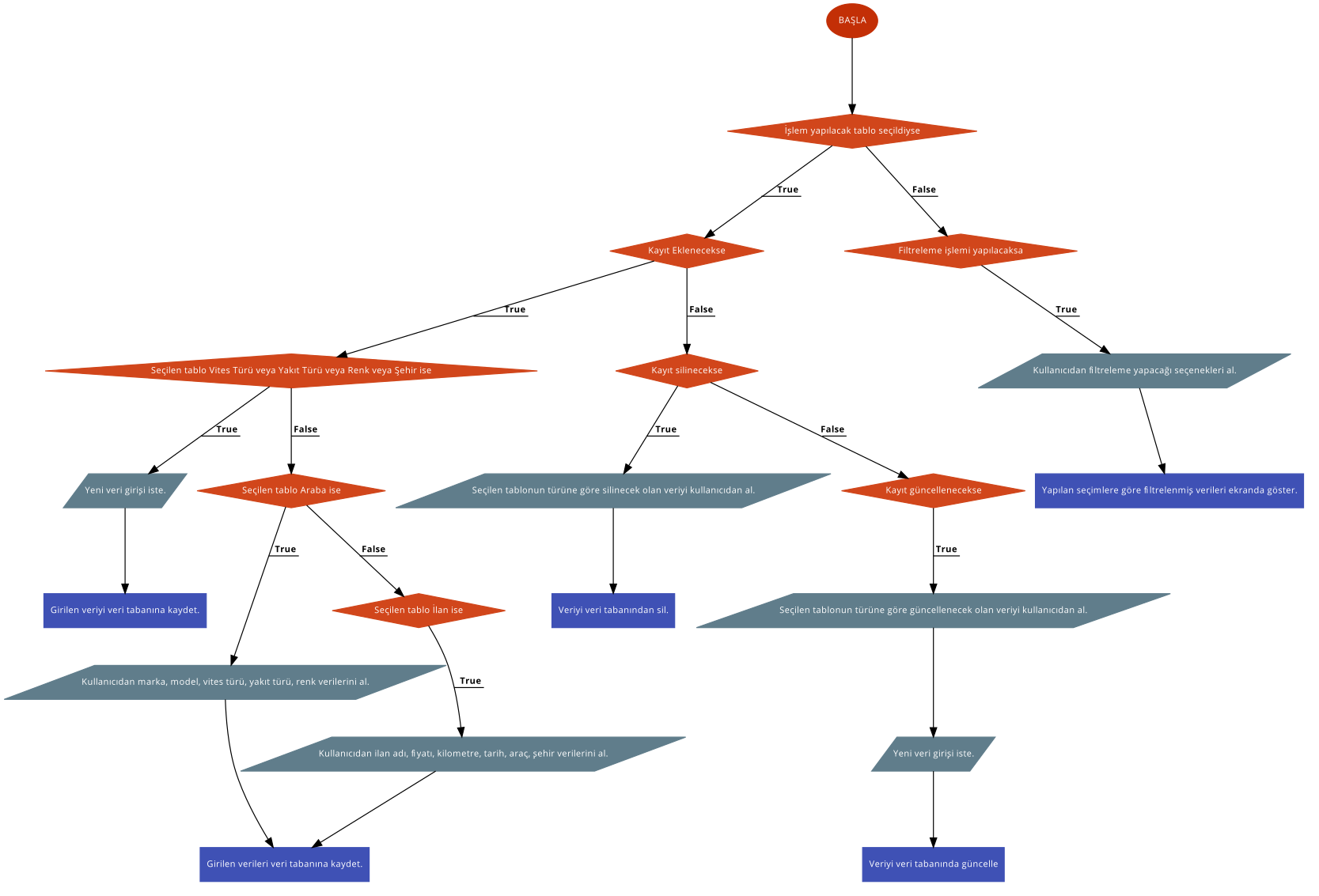
*Şekil 4*: Veri Filtreleme

**6. Sonuçlar**

* Projenin gerçekleştirilmesi sırasında veri tabanı kullanılmıştır. Veri tabanı olarak MYSQL veri tabanı seçilmiştir.
* Veri tabanının yapısı şu şekildedir:
* Tbl\_ilan tablosu içerisinde şu sütunlar yer almaktadır; IlanID, Ilan\_Adi, Ilan\_Fiyat, Ilan\_Km, Ilan\_Tarih, Ilan\_ArabaID, Ilan\_SehirID. Burada; Ilan\_ArabaID ve Ilan\_SehirID sütunları tablo içerisinde “Foreign Key” olarak tanımlanmışlardır ayrıca “IlanID” sütunu da tablo içerisinde “Primary Key” olarak tanımlanmıştır..
* Tbl\_Araba tablosu içerisinde şu sütunlar yer almaktadır; ArabaID, Araba\_Marka, Araba\_Model, Araba\_VitesTuruID, Araba\_YakitTuruID, Araba\_RenkID. Burada; Araba\_VitesTuruID, Araba\_YakitTuruID ve Araba\_RenkID sütunları tablo içerisinde “Foreign Key” olarak tanımlanmışlardır ayrıca “ArabaID” sütunu da tablo içerisinde “Primary Key” olarak tanımlanmıştır...
* Tbl\_YakitTuru tablosu içerisinde şu sütunlar yer almaktadır; YakitTuruID ve Yakit\_Turu. Burada YakitTuruID sütunu “Primary Key” olarak tanımlanmıştır.
* Tbl\_VitesTuru tablosu içerisinde şu sütunlar yer almaktadır; VitesTuruID ve Vites\_Turu. Burada VitesTuruID sütunu “Primary Key” olarak tanımlanmıştır.
* Tbl\_Renk tablosu içerisinde şu sütunlar yer almaktadır; RenkID ve Renk. Burada RenkID sütunu “Primary Key” olarak tanımlanmıştır.
* Tbl\_Sehir tablosu içerisinde şu sütunlar yer almaktadır; SehirID ve Renk. Burada Sehir sütunu “Primary Key” olarak tanımlanmıştır.
* Proje bir masaüstü uygulaması olarak geliştirilmiştir ve programlama dili olarak Java kullanılmıştır.
* Proje iki modülden oluşmaktadır. Birinci kısım tüm tablolar için kayıt ekleme, silme ve güncelleme, ikinci kısım ise filtrelemeden oluşmaktadır.
* Filtreleme kısmında ilan adı, marka, model, renk, vites türü, yakıt türü, şehir, renk, tarih, kilometre ve fiyat seçenekleri arasında kombinasyonel seçim yapılabilmektedir.
* Veriler sıralanırken filtreleme maddelerine göre a-z, z-a, azalan/artan şeklinde sıralanabilmektedir.

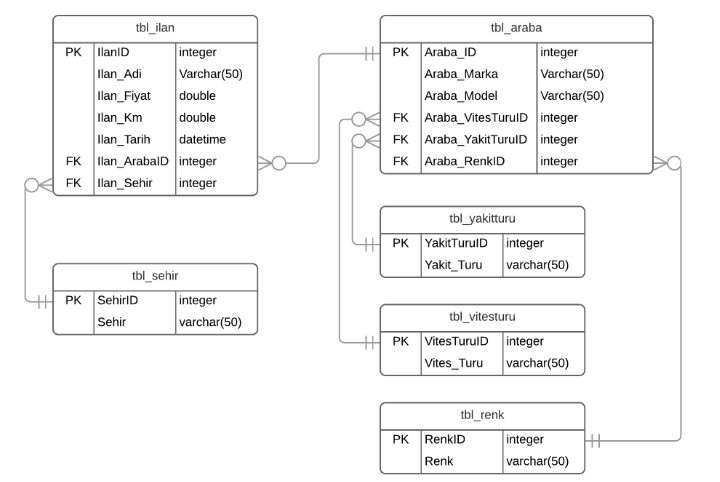
Yukarıdaki değerlendirmeler ışığında, yapmış olduğumuz program, proje sırasında bizlerden istenen tüm adımları yerine getirmekte olup, hiçbir eksik fonksiyonu bulunmamaktadır.

**7. Akış Şeması**



*Şekil 5*: Akış Diyagramı

**8. ER Diyagramı**



*Şekil 6*: ER Diyagramı

**9.Kazanımlar**

Bu proje bize, daha önce derste teorik olarak görmüş olduğumuz veri tabanı konusu üzerinde bir uygulama geliştirerek pratik yapmamızı ve bu konu üzerinde kendimizi geliştirmemizi sağladı. Ayrıca bu proje içerisinde, veri tabanı ile bir masaüstü uygulamasının nasıl birbirine bağlanacağını da öğrenmemizi sağladı.

**10. Referanslar**

<http://www.java2s.com/Tutorial/Java/0240__Swing/AddItemstoJComboBox.htm> (Access Date: 11.05.2018)

<https://stackoverflow.com/questions/3549206/how-to-add-row-in-jtable> (Access Date: 11.05.2018)

<https://stackoverflow.com/questions/29345792/java-jtable-getting-the-data-of-the-selected-row> (Access Date: 11.05.2018)

<https://www.tutorialspoint.com/jdbc/jdbc-insert-records.htm> (Access Date: 12.05.2018)

<https://www.tutorialspoint.com/jdbc/jdbc-select-records.htm> (Access Date: 12.05.2018)

<https://www.tutorialspoint.com/jdbc/jdbc-update-records.htm> (Access Date: 12.05.2018)

<https://www.tutorialspoint.com/jdbc/jdbc-delete-records.htm> (Access Date: 12.05.2018)