



**MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**MOBİL PROGRAMLAMA DERSİ FİNAL PROJESİ**

**Proje Adı**  
**KARABAŞ OTOMOTİV**

**BEGÜM KARABAŞ**  
**22120205002**

**Ders Sorumlusu**  
Dr. Öğr. Üyesi Muhammet Sinan BAŞARSLAN

Ocak, 2025  
İstanbul Medeniyet Üniversitesi, İstanbul

# İÇİNDEKİLER

## Sayfa No

İÇİNDEKİLER.....	1
ÖZET.....	2
1. GİRİŞ .....	3
2. LİTERATÜR TARAMA/İLGİLİ ÇALIŞMALAR .....	4
3. MATERYAL METOD .....	5
3.1 Önyüz Tasarım Araçları .....	5
3.2 Veri Tasarımı.....	5
3.3 Geliştirme Süreci.....	6
4. UYGULAMA.....	7
4.1 Database(Veri Tabanı) .....	7
4.1.1. Firebase Authentication.....	7
4.1.2. Firestore Database .....	8
4.2 Uygulama Arayüzü .....	9
5. SONUÇ VE TARTIŞMA.....	12
6. EKLER .....	12
7. KAYNAKÇA .....	13

## ÖZET

Bu çalışma, bir araba alım-satım uygulaması geliştirilmesini amaçlamaktadır. "Karabaş Otomotiv" adı verilen uygulama, kullanıcıların arabaları incelemesine ve şirket sahibinin araç ilanlarını yönetmesine olanak tanır. Firebase Authentication ve Firestore teknolojileri kullanılarak, müşteri ve şirket sahibi rollerine özel giriş yetkileri ve veritabanı yönetimi sağlanmıştır. Şirket sahipleri, yeni araç ekleme, mevcut ilanları düzenleme ve satılan arabaların ilanlarını kaldırma gibi işlemleri gerçekleştirebilirken, müşteriler mevcut araçları detaylıca inceleyebilir. Uygulama, kullanıcı dostu bir tasarım ve güvenli bir giriş sistemi sunmayı hedeflemiştir.

**Anahtar Kelimeler:** Flutter, Firebase, Firestore, Araç Yönetimi

# 1. GİRİŞ

Mobil teknolojilerin hızla gelişmesi, günlük hayatın pek çok alanında dijitalleşme süreçlerini hızlandırmıştır. Araba alım-satım sektörü de dijitalleşmenin etkisiyle önemli bir dönüşüm yaşamış ve bu alanda kullanıcı dostu, güvenli ve işlevsel mobil uygulamalara olan ihtiyaç giderek artmıştır.

Bu bağlamda geliştirilen **Karabaş Otomotiv** mobil uygulaması, kullanıcıların araba alım-satım işlemlerini dijital bir platformda kolaylıkla gerçekleştirmelerini sağlamayı amaçlamaktadır. Uygulama, hem bireysel müşterilere hem de şirket sahibi rolündeki kullanıcılara hizmet vermek üzere tasarlanmıştır. Bireysel kullanıcılar uygulama üzerinden mevcut araba ilanlarını inceleyebilir ve araç detaylarına ulaşabilirken, şirket sahipleri yeni araç ekleme, ilan düzenleme ve satılan araçların ilanlarını kaldırma gibi işlemleri gerçekleştirebilmektedir.

Bu projede Firebase teknolojileri kullanılarak güvenli bir kimlik doğrulama sistemi ve rol tabanlı erişim modeli uygulanmıştır. **Firestore Authentication**, kullanıcıların güvenli bir şekilde giriş yapmalarını sağlarken, **Firestore** veritabanı, ilan ve kullanıcı verilerinin güvenli ve ölçeklenebilir bir şekilde saklanmasına olanak tanımaktadır. Uygulamada iki ana kullanıcı rolü tanımlanmıştır:

1. **Müşteri:** Mevcut araçları görüntüleyebilir, arabaların detaylı bilgilerine erişebilir.
2. **Şirket Sahibi:** Yeni araç ekleyebilir, ilanları düzenleyebilir ve satılan araçların ilanlarını kaldırabilir.

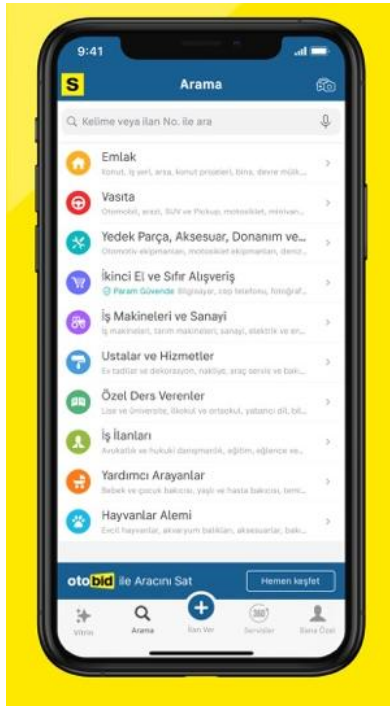
Proje geliştirme sürecinde, **Flutter** framework'ü kullanılarak platformdan bağımsız bir mobil uygulama oluşturulmuştur. Flutter, kullanıcı dostu bir arayüz ve hızlı geliştirme imkânı sunarak projeye önemli katkılar sağlamıştır. Uygulamanın tasarımında, sadelik ve işlevsellik ön planda tutulmuş, kullanıcıların ihtiyaçlarına yönelik kolay bir navigasyon ve anlaşılır bir arayüz sunulmuştur.

Bu projeyle, geleneksel araç alım-satım süreçlerinde karşılaşılan problemleri dijital bir çözümle aşmak hedeflenmiştir. Geleneksel yöntemlerde yaşanan zaman kaybı, iletişim eksikliği ve erişim kısıtlamaları, mobil bir platform üzerinden çözülmüş ve kullanıcıların hem zamandan hem de maliyetten tasarruf etmesi sağlanmıştır. Ayrıca, şirket sahipleri için araç yönetim süreci kolaylaştırılmış, müşteriler için ise güvenilir bir platformda araç arama ve tercih yapma imkânı sunulmuştur.

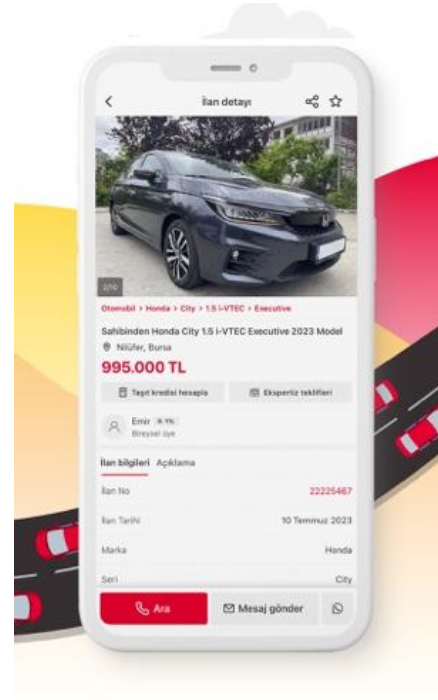
Raporun ilerleyen bölümlerinde, bu uygulamanın geliştirilmesinde izlenen adımlar, kullanılan teknolojiler ve elde edilen sonuçlar detaylı bir şekilde ele alınacaktır. **Karabaş Otomotiv** uygulaması, araba alım-satım sektöründe yenilikçi bir çözüm sunarak kullanıcı deneyimini iyileştirme ve sektöre dijital bir dönüşüm kazandırma yolunda önemli bir adım atmaktadır.

## 2. LİTERATÜR TARAMA/İLGİLİ ÇALIŞMALAR

Literatür taraması kapsamında, araba alım-satım sektöründe popüler olan **Sahibinden** ve **Arabam** uygulamaları incelenmiştir. Bu uygulamaların kullanıcı arayüzleri, işlevsellikleri ve sundukları özellikler detaylı bir şekilde analiz edilmiştir. Özellikle ilan oluşturma, araç detaylarını listeleme, kullanıcı etkileşimleri ve güvenli bir alışveriş ortamı sağlama konularında uygulamaların başarılı yaklaşımları göz önünde bulundurulmuştur. Yapılan bu incelemeler, Karabaş Otomotiv uygulamasının tasarımı ve işlevsel özelliklerinin belirlenmesinde yol gösterici olmuştur. Bu süreçte elde edilen fikirler, kullanıcı deneyimini iyileştirmek ve sektöre uygun, kullanıcı dostu bir çözüm sunmak için projeye entegre edilmiştir.



Şekil 2.1 Sahibinden ana ekran [1]



Şekil 2.2 Arabam uygulamasının araç inceleme ekranı [2]

### 3. MATERYAL METOD

Bu çalışmada, **Karabaş Otomotiv** uygulamasının geliştirilmesi için kullanılan yazılım araçları, yöntemler ve süreçler detaylı bir şekilde ele alınmıştır.

#### 3.1 Önyüz Tasarım Araçları

Uygulamanın ön yüz tasarımında, kullanıcı dostu ve estetik bir görünüm sağlamak amacıyla **Flutter** framework'ü kullanılmıştır. Flutter, zengin widget kütüphanesi ve platformlar arası uyumluluğu ile modern ve işlevsel bir arayüz oluşturulmasına olanak tanımaktadır.

Uygulamanın kullanıcı arayüzü şu bileşenlerden oluşmaktadır:

- **Login ve Kayıt Ekranları:** Kullanıcıların giriş yapmasını ve kayıt olmasını sağlayan formlar.
- **Araç Listeleme ve Detay Ekranları:** Kullanıcıların araçları görüntüleyip detaylarına ulaşmasını sağlayan dinamik arayüzler.
- **Şirket Sahibi Paneli:** Şirket sahiplerinin araç ekleme, düzenleme ve kaldırma işlemlerini gerçekleştirdiği özel bir panel.



Şekil 3.1.1 Flutter logosu [3]

#### 3.2 Veri Tasarımı

Uygulamanın veri yönetimi için **Firestore** teknolojilerinden faydalanılmıştır:

- **3.2.1 Firestore Authentication:** Kullanıcıların güvenli bir şekilde kimlik doğrulaması yapılmıştır. Müşteri kullanıcı rolü tanımlanmıştır ve bu rol üzerinden erişim kontrolleri uygulanmıştır.
- **3.2.2 Firestore Veritabanı:** Uygulamanın veri saklama ihtiyaçları için Firestore kullanılmıştır. Firestore, NoSQL tabanlı bir veritabanıdır ve verilerin hızlı bir şekilde okunup yazılmasını sağlar. Veritabanında şu yapılar oluşturulmuştur:
  - **Users Koleksiyonu:** Kullanıcı bilgilerini saklamak için kullanılır.
  - **Cars Koleksiyonu:** Araçlara ait başlık, görsel, fiyat, marka, model, yıl gibi detayların saklandığı yapı.



Şekil 3.2.1 Firebase logosu [4]

### 3.3 Geliştirme Süreci

Uygulama, aşağıdaki adımlarla geliştirilmiştir:

1. **Planlama ve Gereksinim Belirleme:** Kullanıcıların ihtiyaçları ve uygulama özellikleri belirlendi.
2. **Tasarım:** Kullanıcı dostu bir arayüz oluşturmak için Flutter widget'ları ve tasarım prensipleri uygulandı.
3. **Veritabanı Yapılandırması:** Firebase Authentication ve Firestore yapılandırılarak güvenli bir veri yönetimi sağlandı.
4. **Kodlama:** Uygulama, Google tarafından geliştirilen açık kaynaklı bir framework olan Flutter kullanılarak geliştirilmiştir.

### 3.4 Kullanılan Araçlar

- **Flutter SDK:** Uygulamanın geliştirilmesi için.
- **Firebase Console:** Authentication ve Firestore yapılandırmaları için.
- **Visual Studio Code:** Kodlama ve proje yönetimi için.

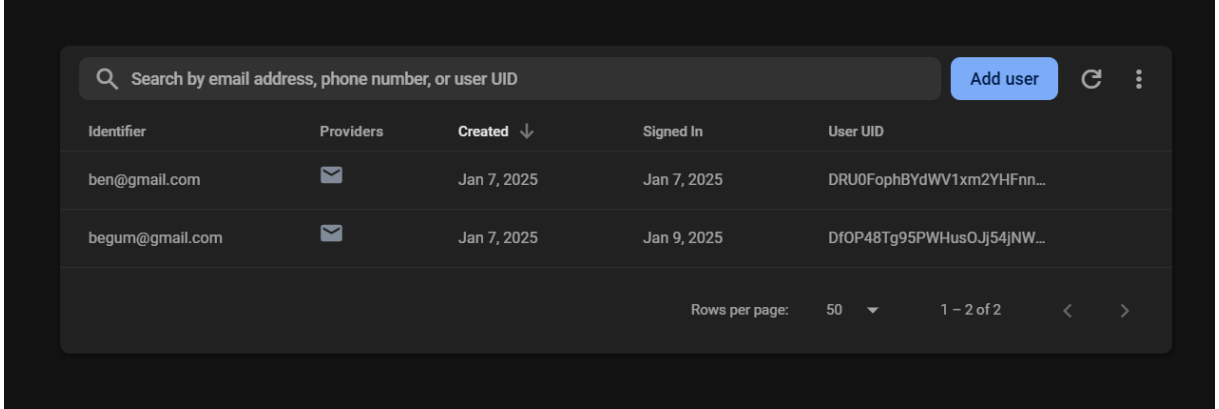
Bu materyal ve metotlar, uygulamanın kullanıcı dostu, güvenilir ve yüksek performanslı bir çözüm sunmasını sağlamıştır.

## 4. UYGULAMA

### 4.1 Database(Veri Tabanı)

#### 4.1.1. Firebase Authentication

Müşterilerin güvenli bir şekilde giriş ve kayıt yapabilmesi için Firebase Authentication'dan yararlanılmıştır.



Şekil 4.1.1.1 Programın Firebase Authentication ekran görüntüsü

```
try {  
    UserCredential? userCredential = await FirebaseAuth.instance  
        .createUserWithEmailAndPassword(  
            email: emailController.text, password: passwordController.text);  
    Navigator.pop(context);  
  
    final User? user = userCredential.user;
```

Şekil 4.1.1.2 FirebaseAuth kullanarak yeni bir kullanıcı kaydı oluşturmaya yaran kod bloğu

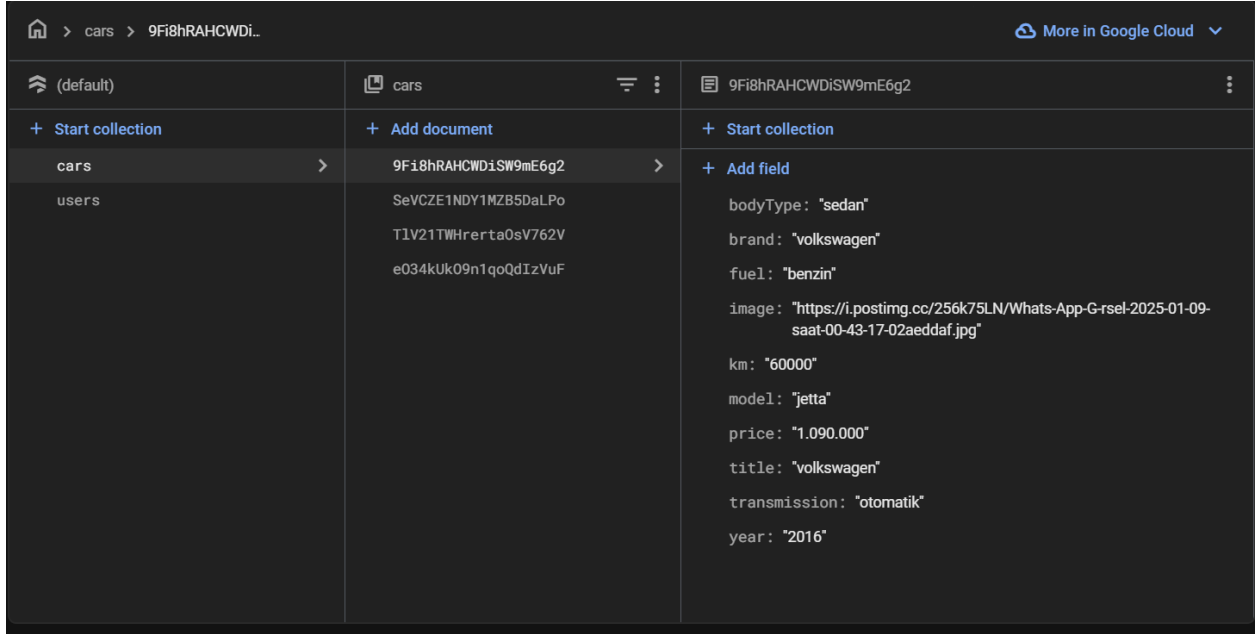
```
try {  
    await FirebaseAuth.instance.signInWithEmailAndPassword(  
        email: _emailController.text.trim(),  
        password: _passwordController.text.trim(),  
    );
```

Şekil 4.1.1.3 FirebaseAuth kullanarak kullanıcının girişini kontrol etmeye yaran kod bloğu



### 4.1.2. Firestore Database

Firebase Firestore veritabanı "**cars**" ve "**users**" adlı iki ana koleksiyon içeriyor.



Şekil 4.1.2.1 Firestore Database görüntüsü

Cars, koleksiyonunda bodyType, brand, fuel, image, km, model, price, title, transmission, year bilgileri depolanır.

```
try {  
  await FirebaseFirestore.instance.collection('cars').add({  
    'title': _titleController.text,  
    'image': _imageController.text,  
    'price': _priceController.text,  
    'bodyType': _bodyTypeController.text,  
    'km': _kmController.text,  
    'brand': _brandController.text,  
    'model': _modelController.text,  
    'transmission': _transmissionController.text,  
    'fuel': _fuelController.text,  
    'year': _yearController.text,  
  });  
}
```

```
final carRef = FirebaseFirestore.instance.collection('cars').doc(carId);
```

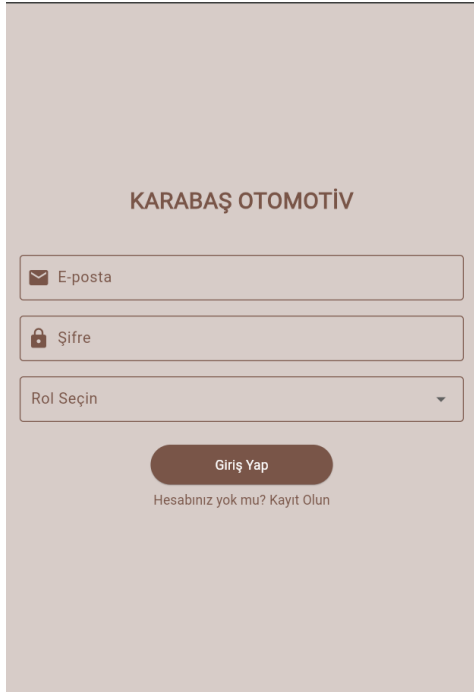
Şekil 4.1.2.2 Araba bilgilerinin database e eklenmesi ve araba bilgilerinin databaseden okunmasını sağlayan kod blokları

Users, koleksiyonunda kullanıcının maili, ne zaman kayıt yaptığı ve id bilgileri depolanır.

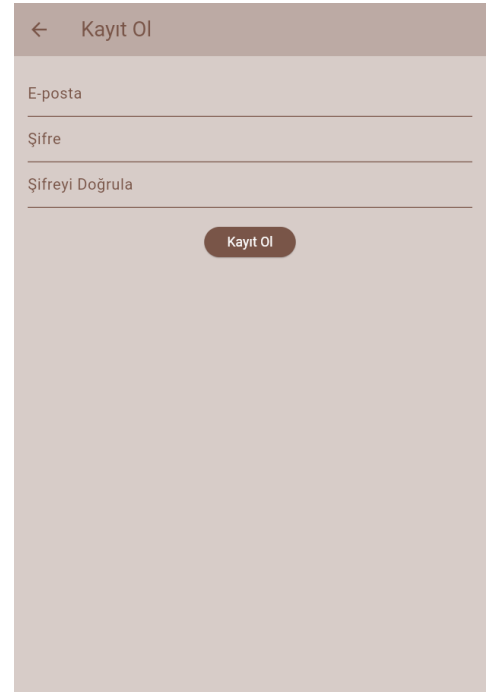
```
await FirebaseFirestore.instance
  .collection('users')
  .doc(user.uid)
  .set({
    'email': user.email,
    'uid': user.uid,
    'createdAt': Timestamp.now(),
  });
```

Şekil 4.1.2.3 kullanıcı bilgilerinin Firebase kaydedilmesini sağlayan kod bloğu

## 4.2 Uygulama Arayüzü



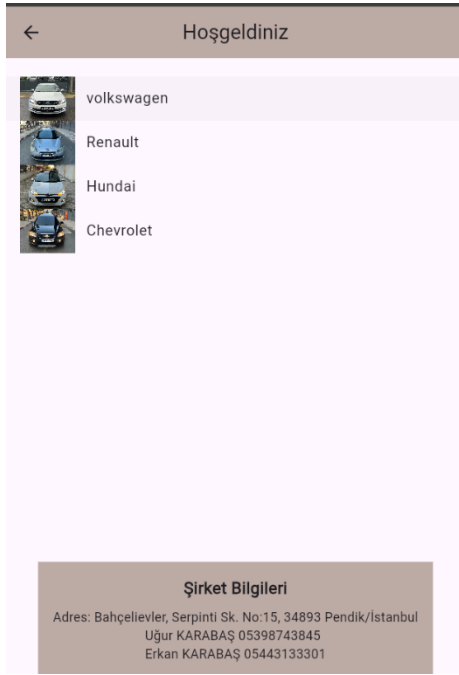
Şekil 4.2.1 Giriş ekranı



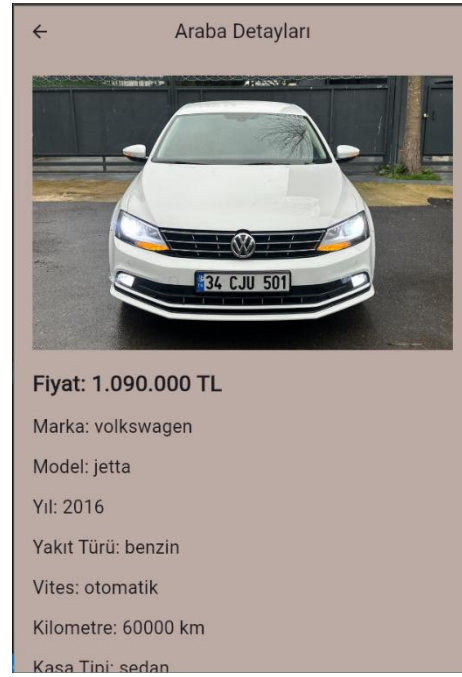
Şekil 4.2.2 Kayıt ekranı

Giriş ekranında, kullanıcının eposta ve şifre bilgilerini girmesi için alan bulunmaktadır. Ayrıca kullanıcının şirket sahibi veya müşteri rollerinden birini seçmesi istenmektedir. Alt tarafta giriş yap butonu bulunmaktadır. Onun altında kayıt ekranına yönlendiren bir buton bulunmaktadır.

Kayıt ekranında eposta ve şifre girişi için alanlar bulunmaktadır. Ayrıca kayıt ol butonu bulunmaktadır.

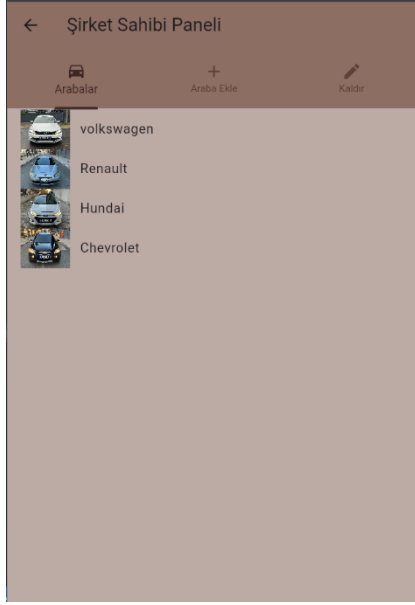


Şekil 4.2.3 Müşteri ekranı

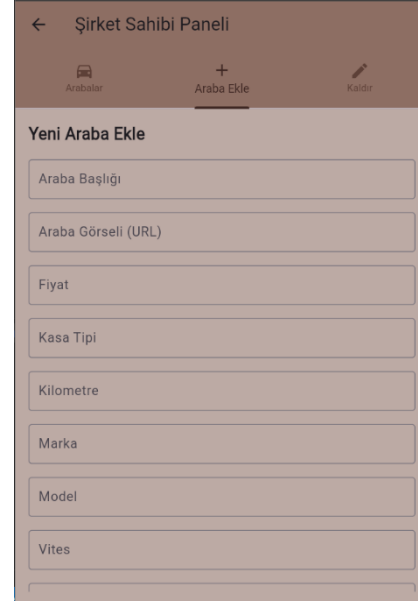


Şekil 4.2.4 Araba özelliklerini gösteren ekran

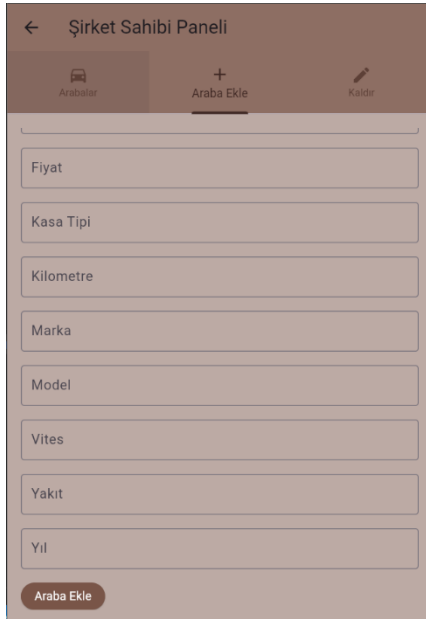
Müşteri ekranında arabaların listesi bulunmaktadır. En alt kısımda ise adres ve iletişim bilgileri bulunmaktadır. Eğer müşteri arabaya tıklarsa (şekil 4.2.4) arabanın detaylı özelliklerini gösteren ekran açılmaktadır.



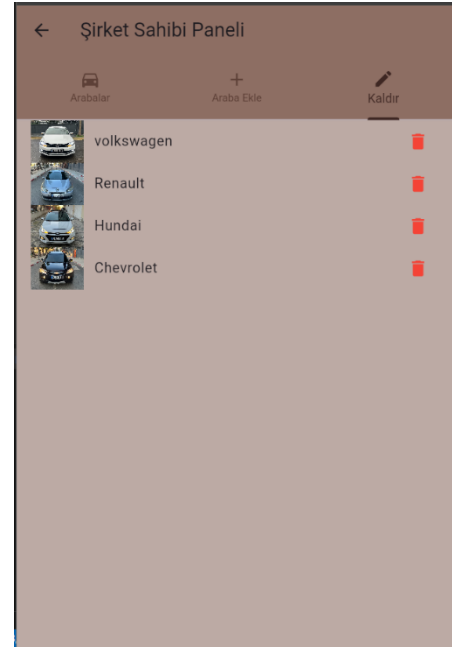
Şekil 4.2.5 Şirket sahibi ekranı



Şekil 4.2.6 Araba ekleme ekranı



Şekil 4.2.7 Araba ekleme ekranının alt kısmı



Şekil 4.2.8 Araba silme ekranı

Şirket sahibi ekranında 3 adet tabbar bulunmaktadır. Birincisi ilanda olan arabaların listelendiği ekrandır. Bu ekranda da müşteri ekranında olduğu gibi arabalara tıklanırsa (şekil 4.2.4) arabanın detaylı özelliklerini gösteren ekran açılmaktadır. İkincisi şirket sahibinin araba ekleme ekranıdır. Bu ekranda arabaların özelliklerinin girilebileceği alanlar ve araba ekle butonu bulunur. Üçüncü ekran ise arabaları ilandan kaldırmak için kullanılan ekrandır. Arabaların yanlarında bulunan çöp kutusu simgesine tıklayınca araba databaseden silinir ve ilandan kalkar.

## 5. SONUÇ VE TARTIŞMA

Bu proje kapsamında, Flutter framework'ü kullanılarak araba alım-satım işlemleri için kullanıcı dostu bir mobil uygulama geliştirilmiştir. Firebase Authentication ve Firestore entegre edilerek kullanıcı kimlik doğrulama, veri saklama ve yönetim süreçleri güvenilir ve etkili bir şekilde gerçekleştirilmiştir. Kullanıcılar, sisteme e-posta ve şifre ile kayıt olabilir, giriş yapabilir ve araçların detaylarını görüntüleyebilirler. Araçların özellikleri (marka, model, fiyat, kilometre, yakıt türü gibi) Firestore veritabanında yapılandırılmıştır ve dinamik olarak kullanıcı arayüzüne yansıtılmıştır.

Sonuç olarak, proje hedeflerine uygun şekilde işlevsel ve ölçeklenebilir bir mobil uygulama ortaya konulmuştur. Kullanıcılar, uygulama aracılığıyla araç ilanlarını inceleme ve favorilere ekleme gibi temel işlemleri rahatlıkla gerçekleştirebilmektedir.

## 6. EKLER

Projeye ait kodlar GitHub üzerinde public olarak tutulmaktadır. Proje linki kaynaklar bölümünde verilmiştir [5].

## 7. KAYNAKÇA

- [1] <https://apps.apple.com/tr/app/sahibinden-alsatkiralakesfet/id418535251>
- [2] <https://play.google.com/store/apps/details?id=com.dogan.arabam>
- [3] [https://api.reliasoftware.com/uploads/what\\_is\\_flutter\\_fcb6c7a4b8.png](https://api.reliasoftware.com/uploads/what_is_flutter_fcb6c7a4b8.png)
- [4] <https://www.kozmoslisesi.com/wp-content/uploads/2022/11/Firebase-1.jpg>
- [5] <https://github.com/begumkarabas/FinalProjeKarabasOtomotiv>