

ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



YEMEK TARİFİ PROJESİ

BLM4537

Batuhan Arda Kibrit

20290266

Github Linki:

<https://github.com/BatuhanArdaKibrit/BLM4537>

ÖZET

Proje flutter sayesinde hem android hem ios cihazlarda çalışan multi platform, herkesin kendi tariflerini paylaştığı, düzenlediği ve isterse silebildiği kullanımı basit ama işlevsel bir mobil uygulamadır. Herkesin kendine özgü favori ve tariflerim sekmesi bulunmaktadır. Ana sayfada ise uygulamaya kayıt olmuş kişilerin oluşturduğu tüm tarifler mevcuttur. Bu şekilde herkesin etkileşimde olduğu bir platform oluşturulmaya çalışılmıştır. Uygulamada kayıt olan kişiler sonradan isterlerse Ad-Soyad, Email ve Parola gibi bilgilerini değiştirebilmektedir. Projenin backend tarafında dotnet kullanılmıştır. Her istek yaratılan json web token (jwt) ile atılmaktadır.

İÇİNDEKİLER

ÖZET.....	1
İÇİNDEKİLER.....	2
1. Projenin Amacı.....	3
2. Başlangıç Ekranları.....	3
2.1. <u>Slider Ekranı</u>	4
2.2. <u>Giriş Ekranı</u>	5
3. Tarif Ekranları.....	6
3.1. <u>Ana sayfa</u>	6
3.2. <u>Favoriler sayfası</u>	7
3.3. <u>Kendi tariflerim</u>	7
3.4. <u>Tarif sekmesi</u>	9
4. Kullanıcı Ekranları.....	10
5. Kaynak kod.....	12
KAYNAKLAR.....	14
EKLER.....	15
Ek-1 - <u>Kullanıcı bulunmamaktadır</u>	15
Ek-2 - <u>Hatalı parola</u>	15
Ek-3 - <u>Kayıtlı hesap bulunmaktadır</u>	16
Ek-4 - <u>Hesabı sil</u>	16
Ek-5 - <u>Eski parola uyarı</u>	16

1. Projenin Amacı

Uygulamayı kullanan kullanıcılar hem kendi kaydetmek istedikleri tariflerini kayıt edebilmektedir hem de uygulamayı kullanan diğer kullanıcıların paylaştığı tarifleri görebilmektedir bu sayede hem herkesin etkileşimde olduğu hem de not defteri gibi bir kullanım amaçlanmıştır.

2. Başlangıç Ekranları

Uygulamanın logosu ve giriş ekranları ekran görüntüsü şeklinde aşağıda belirtilmiştir. Kullanıcıya tatlı bir tasarımla uygulamaya giriş yapması amaçlanmıştır.

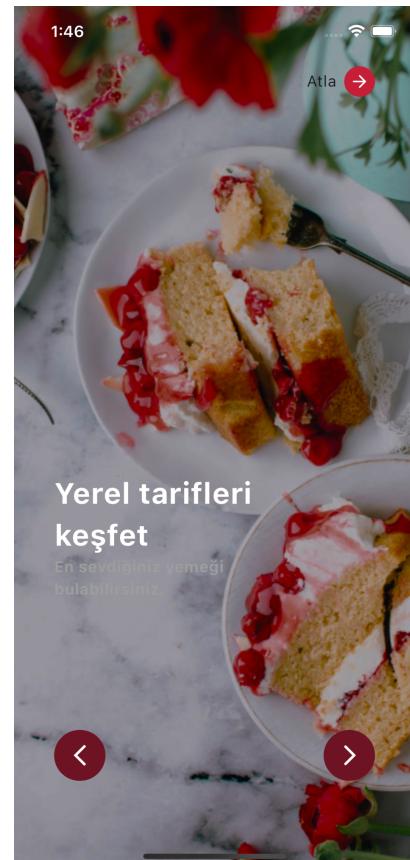


Uygulamanın ikonu sol tarafta gözüktüğü gibidir.



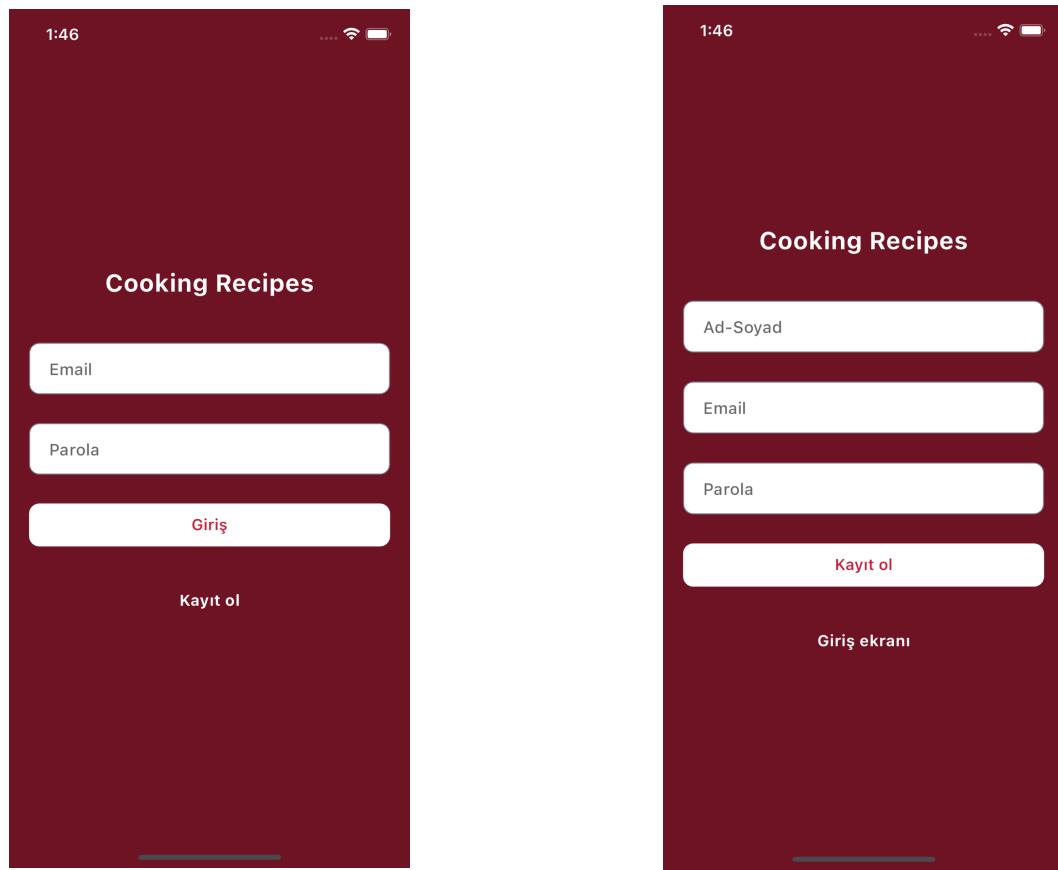
Uygulama ilk açıldığı zaman yüklenen ekran sol tarafta gözükmektedir.

2.1. Slider ekranı



Giriş ekranında kullanıcının karşısına çıkan ekranlar sayfa içerisinde verilmiştir.

2.2. Giriş Ekranı



Kullanıcı girişi ve kayıt olma ekranları yukarıda gözüktüğü gibidir. Bu ekranlarda bazı uyarı mesajları bulunmaktadır. Uyarı mesajları şunlardır;

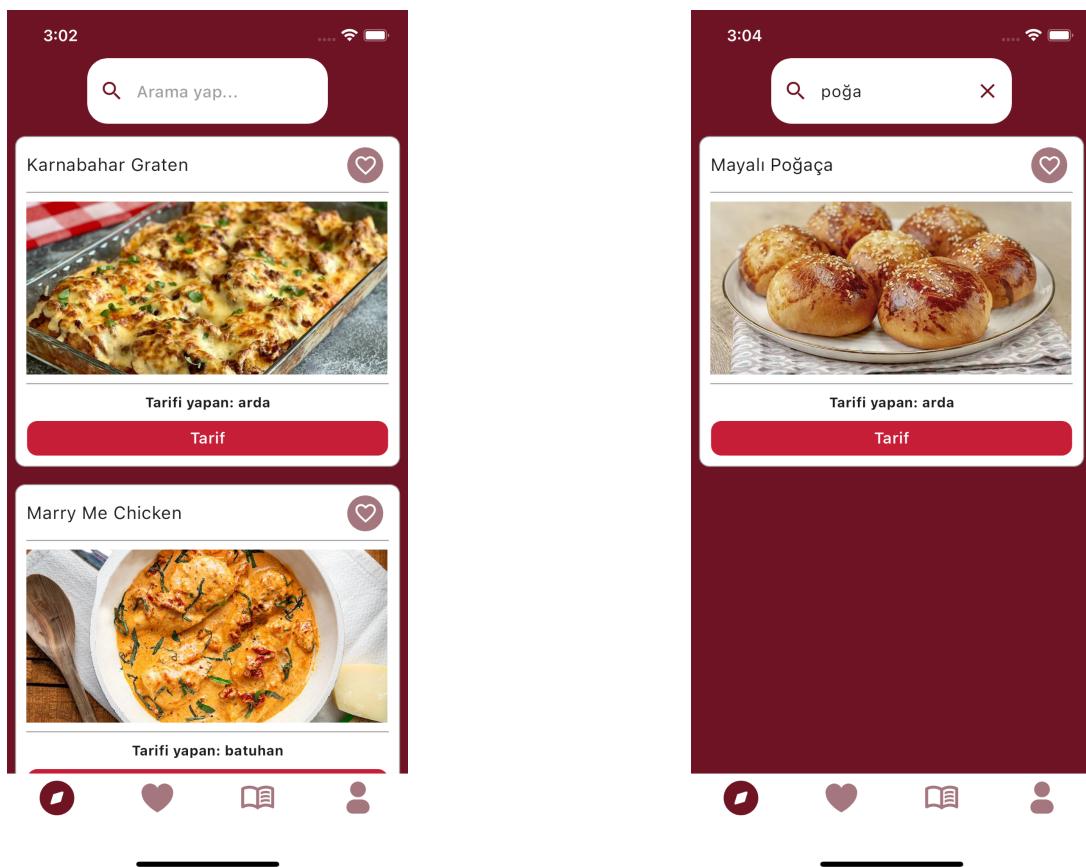
- Girilen e-mail adresiyle kayıt olmuş kullanıcı bulunmamaktadır. (Ek-1)
- Girilen parola doğru değil. (Ek-2)
- Girilen e-mail adresi zaten kayıtlıdır. (Ek-3)

Parola yazı alanında girilen yazılar “*” harfi ile belirtilmektedir.

3. Tarif Ekranları

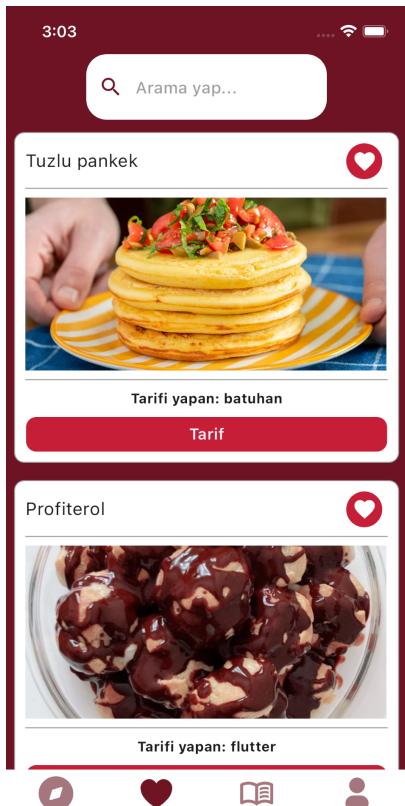
Tariflerin gösterildiği ekranlar üç bölümden oluşmaktadır bunlar sırasıyla Ana sayfa, Favorilerim ve Kendi tariflerim ekranlarıdır.

3.1. Ana sayfa



Ana sayfada uygulamaya kayıtlı olan kullanıcıların oluşturduğu tüm tarifler gözükmektedir. Yukarıdaki ekran görüntülerinde de görüldüğü üzere ekranın en üst kısmında bir arama kutusu bulunmaktadır bu arama kutusu her tarif sayfasında bulunmaktadır.

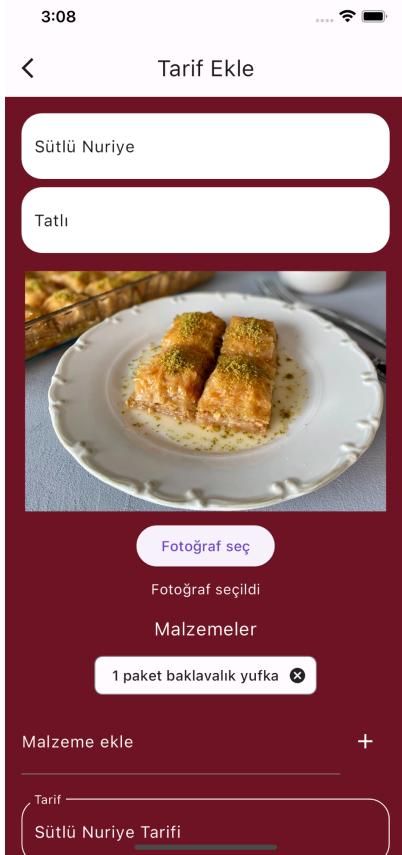
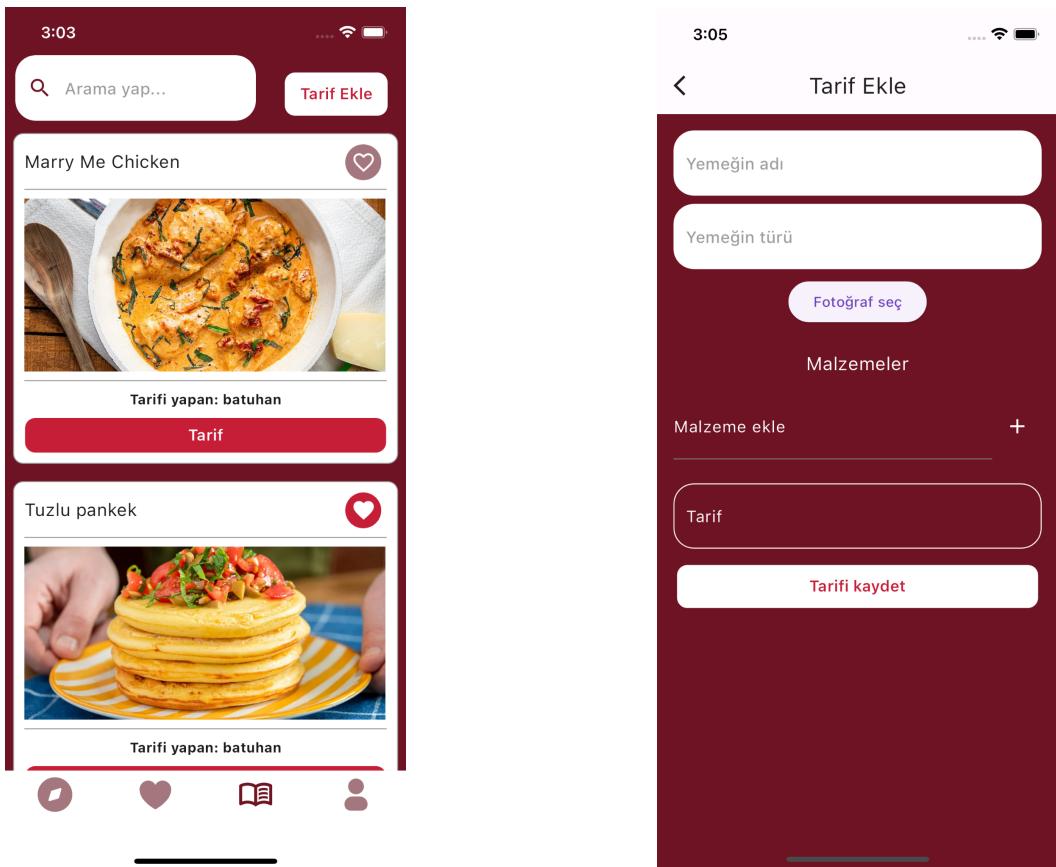
3.2. Favoriler sayfası



Favoriler sayfasında sadece kalp içine alınmış tarifler gözükmektedir. Her kullanıcının favoriler sayfası farklıdır ve favoriler sayfasındayken de bir tarif favorilerden kaldırılabilmektedir. Basıldığı zaman otomatik olarak tarif yok olur.

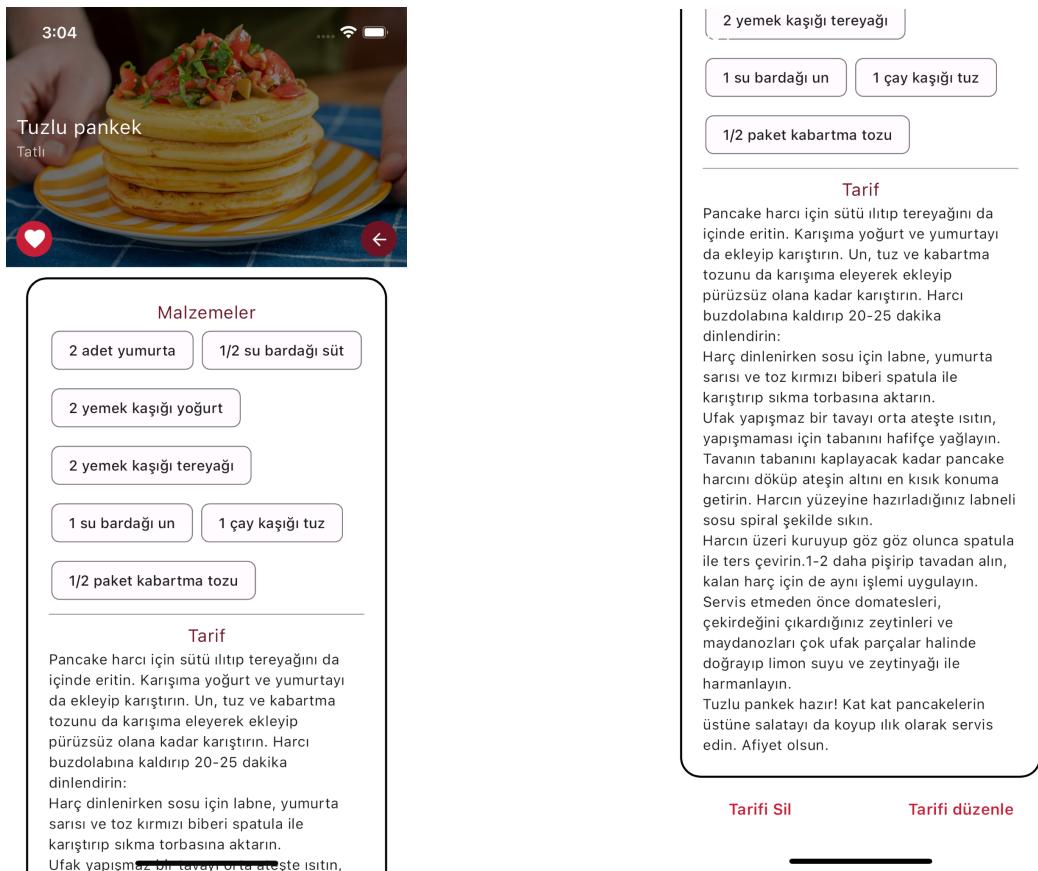
3.3. Kendi tariflerim

Kendi tariflerim sayfasında diğer sayfalardan farklı olarak tarif ekleme kısmı da bulunmaktadır. Tarif ekleme sayfasında tarifin kayıt edilebilmesi için istenilen bilgiler şu şekildedir Yemeğin ismi, türü, fotoğrafı, yemeği yaparken kullanılan malzemeler ve tarifi. Bu bilgiler doldurulduktan sonra bir hatayla karşılaşılmadığı takdirde (bir alanın boş bırakılması gibi) tarif kaydedilecektir ve kullanıcı kendi tariflerim sayfasına atılacaktır.

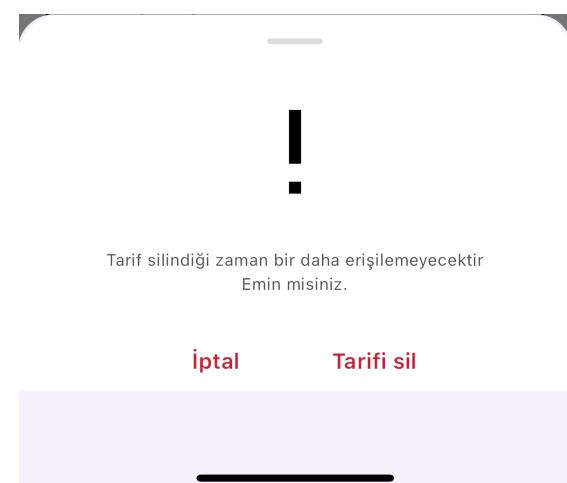


Tarifi eklendikten sonra bir hata ile karşılaşmadıysa kendi tariflerim sayfasına eğer bir hata çıktıysa da hata mesajı bastırılmaktadır.

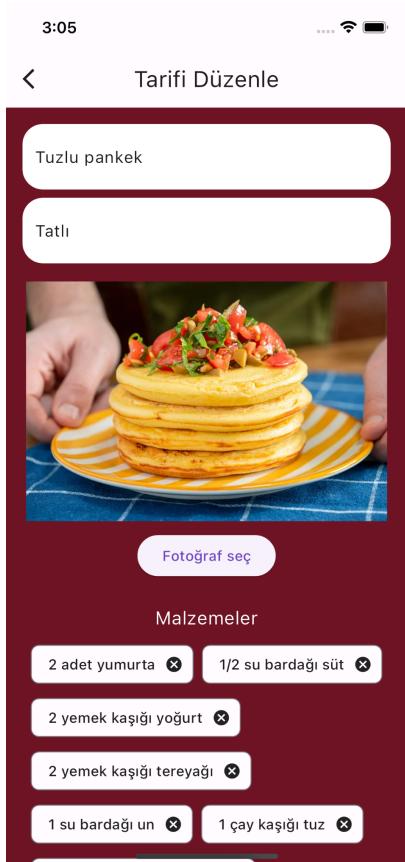
3.4. Tarif sekmesi



Herhangi bir sayfada tarif tuşuna basıldığı zaman Tarif sekmesi açılmaktadır. Eğer tarifi yapan kişi bu tuşa bastıysa kendi tarifini silebilmesi veya düzenleyebilmesi için tuşlar çıkmaktadır ama eğer kullanıcı kendisinin olmayan bir tarife tıkladıysa bu tuşlar gözükmeyecektir. Tarif silme tuşuna basıldığı zaman aşağıdaki ekran görüntüsünde görülen uyarı çıkmaktadır.

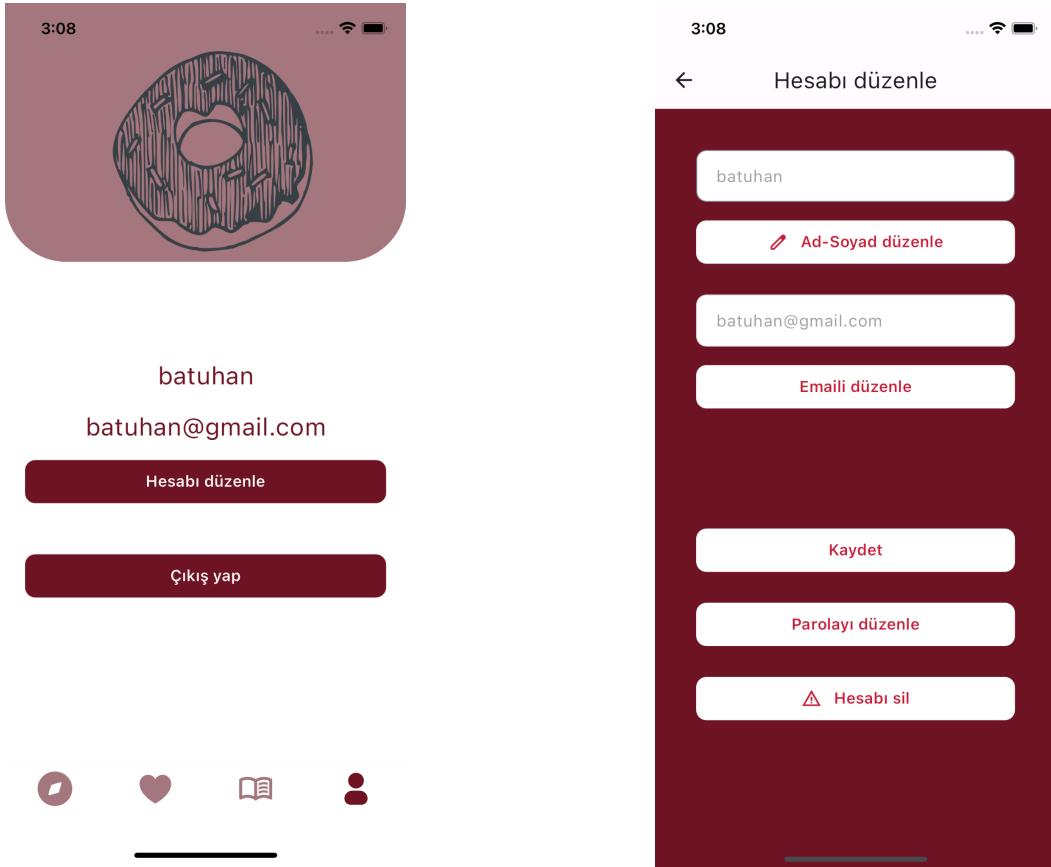


Tarifi düzenleyen tuşuna basıldığında ise tarifte var olan bilgiler ile birlikte tarif eklemek ekranına benzer bir ekran gelmektedir yemeğin ismi veya fotoğrafı fark etmez ne kadar bilgi değiştirilmek istenilirse değiştirilebilir.

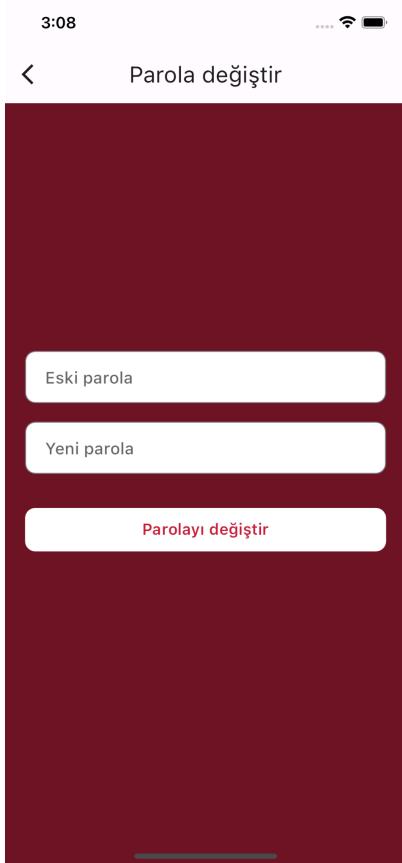


4. Kullanıcı Ekraneları

Kullanıcının bilgilerinin göründüğü ekranda Ad-Soyad ve Email bilgileri yer almaktadır bilgileri düzenleyen butonu ile Ad-Soyad, Email ve Parola bilgileri değişim mümkündür. Çıkış yap butonu sayesinde de girilmiş olan hesaptan çıkış yapmaktadır.



Hesabı sil seçeneğine tıklandığı zaman çıkan ekran ekler kısmında gösterilmiştir (Ek-4)



Ad-Soyad ve Email düzenleme işlemleri aynı anda yapılabilir ama parola düzenleme işlemi farklı bir sayfadadır bunun sebebi Eski parola kontrolu olmasıdır (Ek-5)

5. Kaynak Kod

```
Future<List<RecipeModel>> getAllFood(String email) async {
  ResponseModel response =
    await ApiManager().get(ApiPath.getAllFood, queryParameters: {
      "email": email,
    });

  if (response.status == ResponseEnum.success) {
    List<RecipeModel> recipeModel = dynamicToListRecipeModel(response.data);
    print("data get it Successfull");
    return recipeModel;
  } else {
    print("Some ERROR !");
    return null;
  }
}
```

Ana sayfadaki tüm yemek listesini çektiğim istek buradan statefull widget olan home view a bu listeyi döndürüyorum. Database de image ı byte array olarak malzemeler listesini de virgülle birleştirilmiş string olarak tutuyorum.

```
Future<ResponseModel> postSignIn(String email, String password) async {
  ResponseModel response = await ApiManager().post(ApiPath.signIn,
    body: {"email": email, "password": password}, isHaveHeader: false);

  if (response.status == ResponseEnum.success) {
    // context.push("/home");
    print("Login Successfull");
    Globals.token = response.data["token"];

    // Credentiallari al
    Map<String, dynamic> decodedToken = JwtDecoder.decode(Globals.token);
    Globals.name = decodedToken[
      "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name"];
    Globals.email = decodedToken[
      "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress"];
    return response;
  } else {
    print("Login ERROR !");

    return response;
  }
}
```

Kullanıcı ilk giriş yapınca Json Web Token (jwt) decode edip email bilgisini alıyorum çoğu sorguda mesela yemeği kullanıcı beğenmiş mi bilgisini alabilmek için hangi kullanıcının giriş yaptığıni anlamama yarıyor.

```

class ApiPath {
  static const String prodURL = 'http://localhost:5066/api';

  static const String signIn = '$prodURL/auth/login';
  static const String signUp = '$prodURL/auth/register';
  static const String getAllFood = '$prodURL/food/all';
  static const String likeFood = '$prodURL/food/like';
  static const String userFood = '$prodURL/food/user';
  static const String food = '$prodURL/food';
  static const String dislikeFood = '$prodURL/food/dislike';
  static const String getLikedFood = '$prodURL/food/like';
  static const String account = '$prodURL/account';
}

```

İstek attığım router ların birkaçı yukarıdaki gibidir.

```

class _FoodDetailState extends State<FoodDetail> {
  RecipeModel? recipe;
  late int foodId;
  late Uint8List image;
  late List<String> ingredients;
  late bool isOwner;

  @override
  void initState() {
    setState(() {
      foodId = widget.foodId;
    });
    ApiCallViewModel().checkOwner(foodId).then((value) {
      if(value != null){
        setState(() {
          isOwner = value == Globals.email ? true : false;
        });
      }else{
        setState(() {
          isOwner = false;
        });
      }
    });
    ApiCallViewModel().getFoodDetail(Globals.email, foodId).then((value) {

```

Burada da herhangi bir yemeğin tarif tuşuna basıldığı zaman gelen sayfa (tarif sekmesi) eğer kullanıcı tarifin sahibi ise Tarifi düzenle ve Tarifi sil butonları eklemek için kontrol ediyorum ve Yemeğin bilgilerini çekiyorum.

KAYNAKLAR

<https://docs.flutter.dev>

<https://pub.dev>

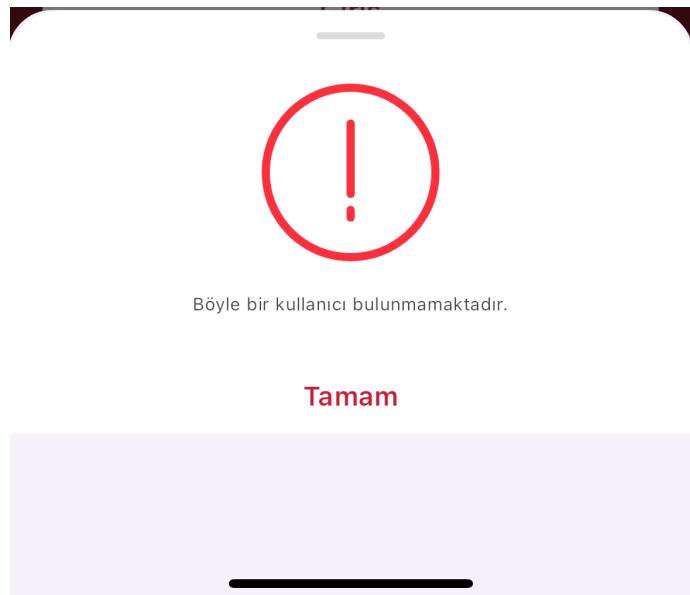
<https://learn.microsoft.com/en-us/dotnet/>

<https://fonts.google.com/icons>

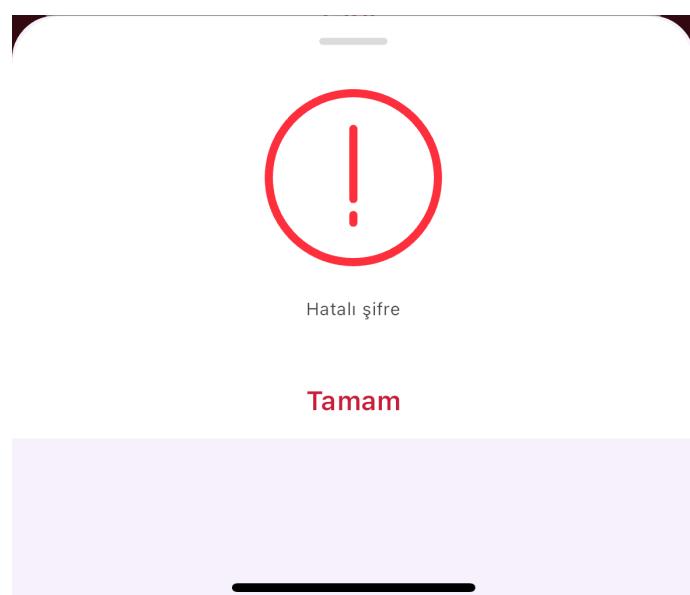
<https://api.flutter.dev/flutter/material/Icons-class.html>

EKLER

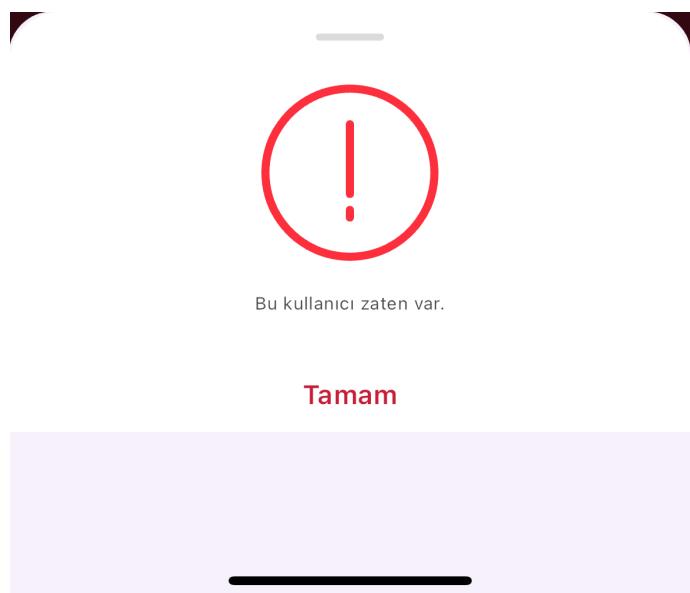
Ek-1 - Kullanıcı bulunmamaktadır



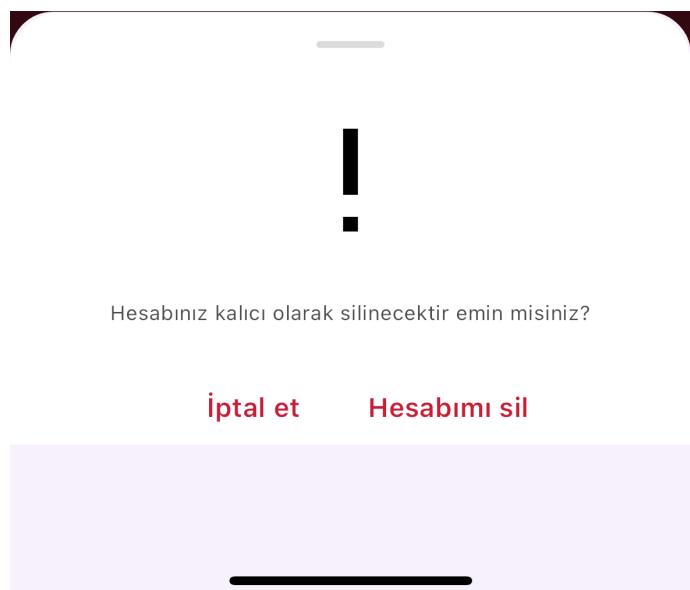
Ek-2 - Hatalı parola



Ek-3 - Kayıtlı hesap bulunmaktadır



Ek-4 - Hesabı sil



Ek-5 - Eski parola uyarı

