



# BS438 UYGULAMA ÖDEVİ/KONU

Anlatım Raporu

Açıklama:  
Erciyes Üniversitesi/Mühendislik Fakültesi/ Bilgisayar  
Mühendisliği/MOBILE APPLICATION DEVELOPMENT/YRD.DOÇ.  
DR. FEHİM KÖYLÜ

Rüveyda Özcan  
1030521038

## İçindekiler Tablosu

<b>Formlar ve Kullanıcı Girişi.....</b>	
<b>Uygulamanın Amacı ve Genel Tanıtım .....</b>	
<b>Uygulamanın Bölümleri.....</b>	
<b>Onboarding Ekranı .....</b>	
<b>OnboardingScreen Özellikleri .....</b>	
<b>Login Ekranı .....</b>	
<b>LoginScreen Özellikleri .....</b>	
<b>Kullanılan Flutter Widget'ları .....</b>	

## Formlar ve Kullanıcı Girişi

Formlar, kullanıcıdan veri almanın ve kullanıcı etkileşimini yönetmenin temel yollarından biridir. Bu formlar, doğru verilerin toplanması için genellikle doğrulama işlemleri gerektirir. Flutter, form doğrulama için basit ve etkili araçlar sunmaktadır. Bu araçlardan biri de TextFormField bileşeni ile entegre edilen validator fonksiyonudur. Flutter, bu sayede form verilerini doğrularken kullanıcı dostu geri bildirimler sağlayabilir.

### Temel Kavramlar ve İşleyiş:

Flutter'da form doğrulama süreci genel olarak aşağıdaki adımları içerir:

1. **Form Oluşturma:** Kullanıcı giriş alanlarını (örneğin, TextFormField) içeren bir Form widget'ı oluşturulur. Form widget'ı, birden fazla form alanını gruplamak ve toplu olarak doğrulamak için bir konteyner görevi görür.
2. **GlobalKey Tanımlama:** Formun durumuna (state) erişmek ve doğrulama işlemlerini tetiklemek için bir GlobalKey<FormState> nesnesi tanımlanır ve bu anahtar Form widget'ının key parametresine atanır. Bu anahtar, formun mevcut durumuna erişim sağlar.
3. **Doğrulama Mantığı Ekleme:** Her bir kullanıcı giriş alanı için (TextFormField) bir validator fonksiyonu tanımlanır. Bu fonksiyon, kullanıcının girdiği metni parametre olarak alır ve belirli koşulları (örneğin, boş olmama, e-posta formatına uygunluk, minimum uzunluk vb.) kontrol eder. Eğer giriş geçersizse, kullanıcıya gösterilecek bir hata mesajı döndürülür. Giriş geçerliyse, null değeri döndürülür.
4. **Doğrulama ve Gönderme İşlemi:** Formu doğrulamak ve göndermek için genellikle bir düğme (ElevatedButton, TextButton vb.) kullanılır. Bu düğmenin onPressed olayında, GlobalKey aracılığıyla erişilen FormState nesnesinin validate() metodu çağrılır.

validate() metodu, formdaki tüm TextFormField widget'larının validator fonksiyonlarını çalıştırır.

Eğer tüm validator fonksiyonları null döndürürse (yani tüm girişler geçerliyse), validate() metodu true değerini döndürür ve form gönderme işlemi gerçekleştirilebilir.

Eğer herhangi bir validator fonksiyonu bir hata mesajı döndürürse, validate() metodu false değerini döndürür ve ilgili TextFormField widget'larının altında hata mesajları otomatik olarak görüntülenir. Form gönderme işlemi engellenir.

```

// ignore
Form(
  key: _formKey,
  child: Column(
    children: [
      // E-posta Alanı
      TextFormField(
        controller: emailController,
        decoration: InputDecoration(
          labelText: "E-posta",
          border: OutlineInputBorder(),
        ),
        // InputDecoration
        validator: (value) => value!.contains('@') ? null : "Geçerli bir e-posta girin", //belirtilen karakterin mail kısmında olup olmadığını kontrol ettim.
      ), // TextFormField
      SizedBox(height: 20),
      // Şifre Alanı
      TextFormField(
        controller: passwordController,
        decoration: InputDecoration(
          labelText: "Şifre",
          border: OutlineInputBorder(),
        ),
        // InputDecoration
        obscureText: true,
        validator: (value) => value!.length >= 6 ? null : "En az 6 karakter olmalı", //Şifreye karakter sınırı getirdim.
      ), // TextFormField
    ],
  ),
)

```

Şekil 1. Uygulamada kullanılan Form kısmı

## Uygulamanın Amacı ve Genel Tanıtım:

Bu uygulama, bir onboarding (hoş geldiniz) ekranı ile başlayarak, kullanıcıya temel bir giriş ekranı sunan basit bir Flutter uygulamasıdır. Kullanıcılar, uygulamayı açtıklarında sırasıyla üç aşamalı onboarding ekranı ile karşılaşırlar. Son aşamada, kullanıcıdan e-posta adresi ve şifresi ile giriş yapması istenmektedir.



Şekil 2. Uygulama Onboarding başlangıç sayfası

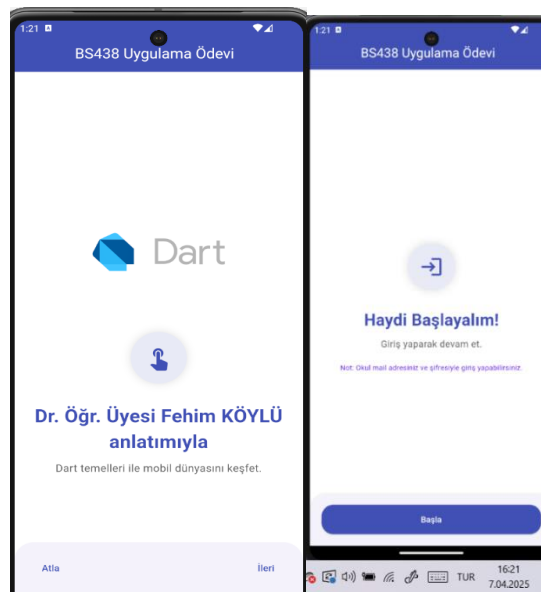
## Uygulamanın Bölümleri:

### 1. Onboarding Ekranı:

Uygulamanın ilk kısmı OnboardingScreen widget'ında gerçekleşmektedir. Kullanıcılar üç aşamadan oluşan bir onboarding sürecinden geçer. Bu ekran, kullanıcılara uygulama hakkında bilgi vererek onlara yönlendirme sağlar.

### OnboardingScreen Özellikleri:

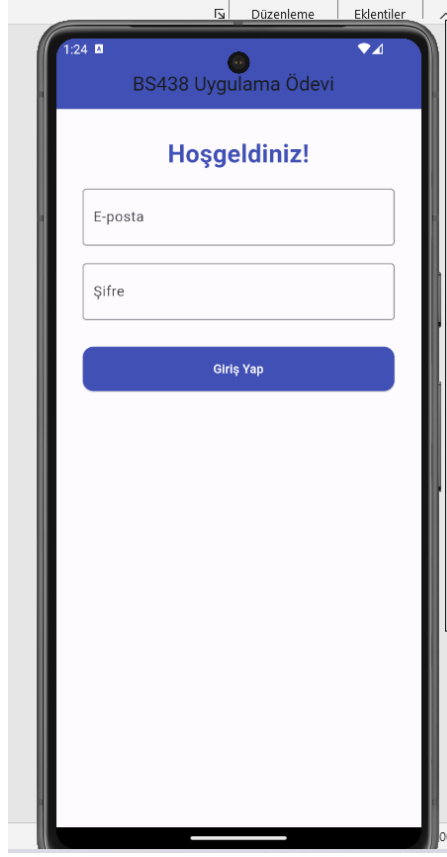
- **PageView Kullanımı:** Onboarding ekranındaki her bir sayfa, PageController ile kontrol edilen bir PageView widget'ı tarafından yönetilir. Bu widget sayesinde ekranlar arasında geçiş yapılabilir.
- **Dinamizm:** Son sayfada, kullanıcıya "Başla" butonu sunulur. Butona tıklandığında kullanıcı, LoginScreen ekranına yönlendirilir.
- **İçerik Özelleştirme:** Her sayfada başlık, açıklama, resim (veya ikon) ve son sayfada ek bir not yer alır. Bu içeriklerin sayfalara dinamik olarak geçirilmesi sağlanır.
- **Geçiş Düğmeleri:** Kullanıcılar ekranlar arasında "Atla" ve "İleri" butonları ile geçiş yapabilirler.



Şekil 3. Onboarding ikinci ve üçüncü sayfaları

## 2. Login Ekranı:

LoginScreen widget'ı, kullanıcıdan okul e-posta adresi ve şifresiyle (default verildi.) giriş yapmasını isteyen bir ekran içerir. Kullanıcı doğru bilgileri girerse başarılı bir giriş mesajı gösterilir, aksi halde hata mesajı gösterilir.



Şekil 4. Login ekranı

### LoginScreen Özellikleri:

- **Form Doğrulama:** Form ve TextFormField widget'ları kullanılarak kullanıcıdan alınan e-posta ve şifre bilgileri doğrulanır. E-posta için geçerli bir e-posta formatı, şifre için ise en az 6 karakter kuralı belirlenmiştir.
- **Giriş İşlemi:** Kullanıcı doğru bilgileri girerse, ekranda başarı mesajı görüntülenir; yanlış girildiğinde ise hata mesajı çıkar.
- **UI Geliştirmeleri:** Başlık, form alanları, butonlar, ve giriş metinleri tasarıma uygun şekilde düzenlenmiştir.

## Hoşgeldiniz!

E-posta

Geçerli bir e-posta girin

Şifre

En az 6 karakter olmalı

Giriş Yap

## Hoşgeldiniz!

E-posta

1030521038@erciyes.edu.tr

Şifre

.....

Giriş Yap

Giriş başarılı!

## Hoşgeldiniz!

E-posta

1030521078@erciyes.edu.tr

Şifre

.....

Giriş Yap

E-posta veya şifre yanlış

Şekil 5. Login kontrolleri

### **Kullanılan Flutter Widget'ları:**

- **MaterialApp:** Uygulamanın temel yapısını belirleyen widget. home özelliği ile uygulama başlatıldığında hangi ekranın gösterileceği belirtilir.
- **Scaffold:** Sayfa yapısını oluşturan temel widget. İçinde AppBar, body, ve bottomSheet gibi alt widget'ları barındırır.
- **PageView:** Kullanıcıya birden fazla sayfadan oluşan bir onboarding deneyimi sunmak için kullanılır.
- **TextFormField:** Kullanıcıdan e-posta ve şifre bilgilerini almak için kullanılır.
- **ElevatedButton:** Giriş yapma butonunu oluşturmak için kullanılır. Kullanıcı giriş işlemini bu butona tıklayarak başlatır.
- **Form & GlobalKey:** Form elemanlarının doğrulanmasını sağlamak ve kullanıcı hatalı bilgi girdiğinde uyarı vermek için kullanılır.