

FLUTTER

Alt Menü ve Sekmeler

16. hafta • Düzen ve Gezinme

36 hafta, 72 bağımsız çalışan Flutter projesi, gerçek uygulama ekranları, uygulamalı dersler ve öğrenci geri bildiriminden sonra incelenebilen çözüm kaynakları.

Bu kitap güncel ders kaynaklarından otomatik üretilmiştir; özet ya da jenerik şablon değildir.

Öğrenme Haritası

Hafta	Ders	Projeler
16	Alt Menü ve Sekmeler	Okul Menüsü • Kişisel Ajanda

16. Hafta — Alt Menü ve Sekmeler

Bu hafta ne yapacağız?

Ana Sayfa, Dersler ve Profil bölümleri arasında `NavigationBar` ile geçilen bir okul portalı geliştireceğiz. Dersler bölümünde Program ve Ödevler içeriklerini `TabBar` ile ayıracağız. Ardından sekmeler değişirken görev durumunu kaybetmeyen özgün kişisel ajanda yapacağız.

Kazanımlar

- Birincil uygulama bölümlerini `NavigationBar` ile düzenleme
- Aynı bölüm içindeki ilişkili içerikleri `TabBar` ile ayırma
- Seçili alt menü indeksini durum olarak yönetme
- `IndexedStack` ile sayfaların durumunu koruma
- Alt gezinme ile sayfa yığını gezinmesinin farkını açıklama

Uygulama 16 — Okul Menüsü

1. Seçili bölümü tutma

```
int secili = 0;
final sayfalar = const [AnaSayfa(), DerslerSayfasi(), ProfilSayfasi()];
```

2. Alt menüyü oluşturma

```
NavigationBar(
  selectedIndex: secili,
  onDestinationSelected: (index) => setState(() => secili = index),
  destinations: const [
    NavigationDestination(icon: Icon(Icons.home_outlined), label: 'Ana Sayfa'),
    NavigationDestination(icon: Icon(Icons.school_outlined), label: 'Dersler'),
    NavigationDestination(icon: Icon(Icons.person_outline), label: 'Profil'),
  ],
)
```

Alt menü, eş düzeydeki ana bölümler içindir. Ayrıntı sayfası açmak için önceki haftadaki `Navigator.push` kullanılmalıdır.

3. Sayfaları durumlarıyla koruma

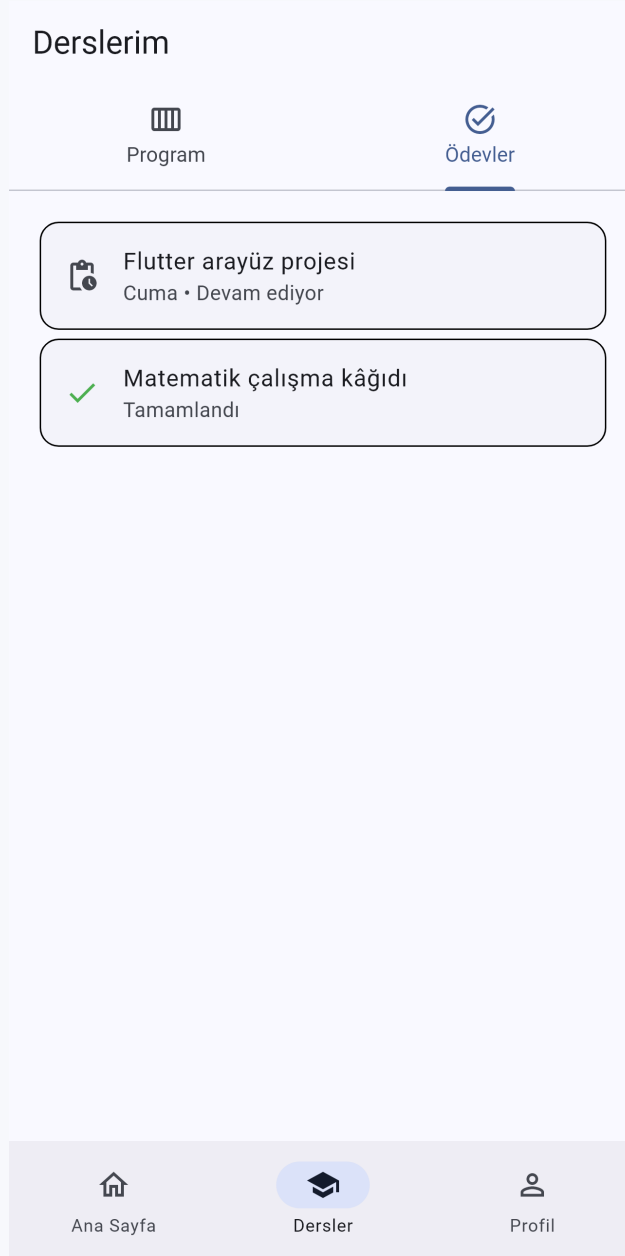
```
IndexedStack(
  index: secili,
  children: sayfalar,
)
```

`IndexedStack` yalnız seçili çocuğu gösterir fakat diğerlerini ağaçtan kaldırmaz.

4. Dersler içindeki sekmeler

```
DefaultTabController(  
  length: 2,  
  child: Column(children: [  
    const TabBar(tabs: [  
      Tab(text: 'Program'),  
      Tab(text: 'Ödevler'),  
    ]),  
    Expanded(child: TabBarView(children: [program, odevler])),  
  ]),  
)
```

Testte alt menüden Dersler'e, içeride Ödevler sekmesine, ardından Profil'e geçildi. Her hedef içeriğin doğru görüldüğü doğrulandı.



Dersler bölümünün Ödevler sekmesini gösteren Okul Menüsü

Özgün proje — Kişisel Ajanda

Ajanda Bugün, Takvim ve Ayarlar bölümlerinden oluşur. Görev listesi kabuk widget'ın durumunda tutulur:

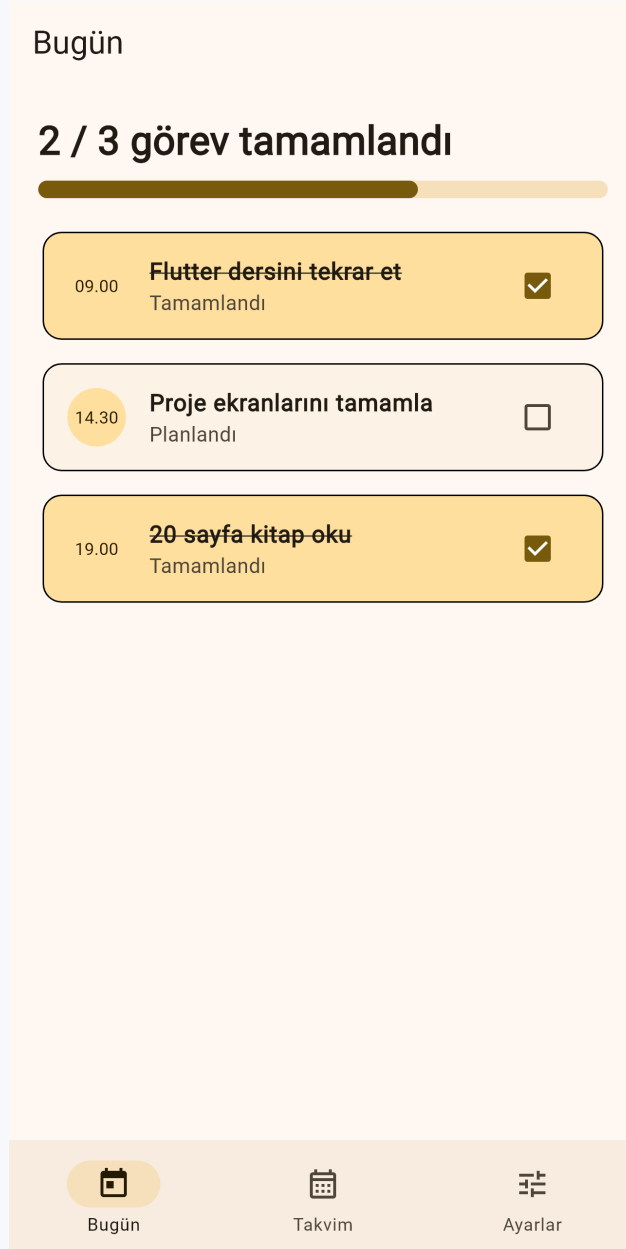
```
final gorevler = [
  Gorev('09.00', 'Flutter dersini tekrar et'),
  Gorev('14.30', 'Proje ekranlar■n■ tamamla'),
  Gorev('19.00', '20 sayfa kitap oku', tamam: true),
];
```

Görev işaretlendiğinde alt sayfa, kabuğa callback gönderir:

```
BugunSayfasi(
  gorevler: gorevler,
  degisti: () => setState(() {}),
)
```

Bu sayede alt menü değişse bile veri yeni sayfa içinde sıfırlanmaz. Saatler dar alanda `FittedBox` ile tek satırda tutulur.

Testte Flutter tekrar görevi tamamlandı; özet 2/3 oldu. Takvim'e geçilip tekrar Bugün'e döndüğünde 2/3 durumu korunuyordu.



Sekme geçişinden sonra görev durumunu koruyan Kişisel Ajanda

Sıra sende

1. Kolay: Okul menüsüne Bildirimler bölümü ekle.
2. Orta: Ajandada tamamlanan görevleri ayrı bir sekmede göster.
3. İleri: Telefon ekranında NavigationBar, geniş ekranda NavigationRail kullanan ve aynı seçili indeksi paylaşan uyarlanabilir kabuk geliştir.

Çözümünü gönderdikten ve geri bildirim aldıktan sonra Kişisel Ajanda'nın gerçek Dart kodu ile çalışan proje paketi açılacaktır. İpucu durumun hangi üst widget'ta tutulacağını belirlemene yardım edecek; çözüm düğmesi eksiksiz `lib/main.dart` kodunu gösterecektir.

Kontrol listesi

- Ana bölümler alt menüde, ilişkili alt içerikler sekmelerde gösterildi.
- Seçili indeks tek bir durum değişkeniyle yönetiliyor.
- Sekme değişiminde görev durumu kaybolmuyor.
- Uzun/dar içeriklerde taşma oluşmuyor.
- İki proje Flutter widget testinde menü ve sekme geçişleriyle doğrulandı; gerçek ekran çıktıları görsel olarak incelendi.