

2209/A

ÜNİVERSİTE ÖĞRENCİLERİ ARAŞTIRMA PROJELERİ DESTEK
PROGRAMI

SONUÇ RAPORU

PROJE BAŞLIĞI: **Bilgisayar Öğreniyoruz**

GENEL BİLGİLER

PROJENİN KONUSU	Bilgisayar Öğreniyoruz
PROJE YÜRÜTÜCÜSÜNÜN ADI	Emir Koç
DANIŞMANIN ADI	Ahmet Berk Üstün
PROJE BAŞLANGIÇ VE BİTİŞ TARİHLERİ	Nisan 2023 – 22 Haziran 2025

1. Giriş

Proje konusu hakkında kısa bilgi

Günümüzde teknolojinin hızla gelişmesiyle mobil uygulamalar, günlük yaşamın ve özellikle eğitim süreçlerinin vazgeçilmez bir parçası haline gelmiştir. Türkiye'de genç ve teknolojiye uyum sağlayabilen nüfus sayesinde mobil uygulama kullanımı oldukça yaygındır. Bu bağlamda, "Bilgisayar Öğreniyoruz" projesi, 8 ila 12 yaş arası çocuklara bilgisayar donanımı ve popüler yazılımlar hakkında temel bilgileri basit, anlaşılır ve etkileşimli bir şekilde sunmayı amaçlayan bir mobil uygulama geliştirme çalışmasıdır. Uygulama, yazılı, görsel ve videolu içerikler aracılığıyla çocukların bilgisayar teknolojisini anlamalarına ve kullanmaya başlamalarına olanak tanımaktadır. Literatürdeki çocuklara yönelik özelleştirilmiş ve yaşa uygun eğitim materyallerinin sınırlılığı göz önüne alındığında, bu proje özgün bir değer taşımaktadır. Proje, donanım ve yazılım bilgilerinin bir arada, çocukların seviyesine uygun, interaktif bir mobil platformda sunulmasıyla bu alandaki bir ihtiyaca cevap vermektedir.

2. Rapor dönemlerinde yapılan çalışmalar

Proje süresince, TÜBİTAK 2209-A programı kapsamında planlanan iş paketleri başarıyla tamamlanmıştır. Yapılan çalışmalar aşağıda özetlenmiştir:

- Gereksinim Analizi (0-2. Aylar):** Projenin ilk aşamasında, hedeflenen yaş grubu olan 8-12 yaş arası çocukların bilgisayar donanımı ve popüler bilgisayar uygulamaları hakkındaki bilgi düzeyleri ve öğrenme ihtiyaçları analiz edildi. Bu

analiz, literatür taraması, benzer uygulamaların incelenmesi ve potansiyel kullanıcı grubuyla (çocuklar ve ebeveynler/eğitimciler) yapılan ön görüşmelerle desteklendi.

- **İçerik Hazırlığı (2-5. Aylar):** Belirlenen gereksinimler doğrultusunda, bilgisayar donanım bileşenleri (işlemci, bellek, anakart vb.) ve popüler yazılımlar (işletim sistemleri, ofis programları, grafik tasarım araçları vb.) hakkında çocukların anlayabileceği bir dilde yazılı metinler, ilgi çekici görseller ve eğitici videolar hazırlandı. İçeriklerin pedagojik uygunluğuna ve yaş seviyesine dikkat edildi.
- **Uygulama Geliştirme (5-9. Aylar):** Mobil uygulamanın geliştirilmesi aşamasında, başlangıçta planlanan Android Studio ve Java/Kotlin yerine, daha modern ve çapraz platform yetenekleri sunan **Dart programlama dili ve Flutter framework'ü** tercih edildi. Geliştirme ortamı olarak Android Studio kullanılmaya devam edildi. Kullanıcı arayüzü (UI) ve kullanıcı deneyimi (UX) tasarımı, çocukların ilgisini çekecek renkler, grafikler ve animasyonlar kullanılarak kullanıcı dostu bir şekilde oluşturuldu. Hazırlanan yazılı, görsel ve videolu içerikler uygulamaya entegre edildi. Uygulama içerisinde etkileşimli öğrenme materyalleri ve öğrenmeyi pekiştirecek testler de geliştirildi.
- **Test ve Geri Bildirim Toplama (9-10. Aylar):** Geliştirilen mobil uygulamanın alfa ve beta sürümleri, hedef kitledeki çocuklar ve danışman gözetiminde test edildi. Kullanılabilirlik, performans, içerik anlaşılabilirliği ve genel kullanıcı deneyimi hakkında geri bildirimler toplandı.
- **İyileştirme ve Sonuçlandırma (10-11. Aylar):** Toplanan geri bildirimler ışığında uygulamada gerekli hata düzeltmeleri ve iyileştirmeler yapıldı. Uygulamanın son hali Google Play Store'da yayınlanmak üzere hazırlandı. Proje süreci, elde edilen sonuçlar ve deneyimler bu sonuç raporunda derlendi.

3. Sonuç:

"Bilgisayar Öğreniyoruz" projesi kapsamında, 8-12 yaş arası çocuklara bilgisayar donanımı ve popüler yazılımları öğretmeyi amaçlayan bir mobil uygulama başarıyla geliştirilmiş ve hedeflerine ulaşılmıştır. Dart programlama dili ve Flutter framework'ü kullanılarak geliştirilen uygulama, çocukların yaş seviyelerine uygun, anlaşılır, görsel ve işitsel materyallerle zenginleştirilmiş bir öğrenme deneyimi sunmaktadır.

Projenin temel hedefleri olan;

- En az 10 farklı bilgisayar donanımı ve 10 popüler bilgisayar yazılımı hakkında anlatım içeren içeriklerin oluşturulması,
- Kullanıcı dostu bir arayüz (UI) ve etkili bir kullanıcı deneyimi (UX) sunulması,
- Etkileşimli öğrenme deneyimi ile öğrenme motivasyonunun artırılması,
- Kapsamlı test süreçleri ve geri bildirimlerle uygulamanın iyileştirilmesi hedefleri gerçekleştirilmiştir.
- Uygulamanın Play Store'da yayınlanması hakkındaki hedefi hayata geçirilememiştir.

Geliştirilen uygulama, çocukların bilgisayar donanımının temel bileşenlerini anlamalarını, popüler bilgisayar yazılımlarını tanımalarını ve temel düzeyde kullanabilmelerini sağlamaktadır. Bu sayede çocukların teknoloji dünyasına olan ilgilerinin artırılması ve bilinçli teknoloji kullanıcıları olmaları teşvik edilmiştir. Uygulama, eğlenceli ve etkileşimli yapısıyla öğrenme sürecini pekiştirmiş ve çocukların motivasyonunu artırmıştır. Proje, çocukların erken yaşta dijital okuryazarlık becerileri kazanmalarına ve gelecekteki teknoloji adaptasyonlarına olumlu katkılar sunmaktadır.

4. Çıktılar (Yayınlar,sunumlar v.b.)

Akademik Yayın ve Sunum Hedefleri: Proje sonuçlarının ve geliştirilen uygulamanın üniversitelerde düzenlenen proje pazarlarında sunulması ve özet bildiri olarak yayınlanması, bulguların ulusal veya uluslararası hakemli bir dergide makale olarak yayınlanması, projenin geliştirilmesi ve ardından yayınlanması halinde gerçekleştirilebilir bir hedef olarak düşünülmektedir.

Sosyal Etki:

Ulaşılması kolay kaynaklar ve modern teknolojiler (Dart/Flutter) kullanılarak geliştirilen yazılımın, erken yaşlardan itibaren bilgisayar becerilerinin geliştirilmesine katkıda bulunması ve toplumda dijital yetkinliği artırması beklenmektedir.

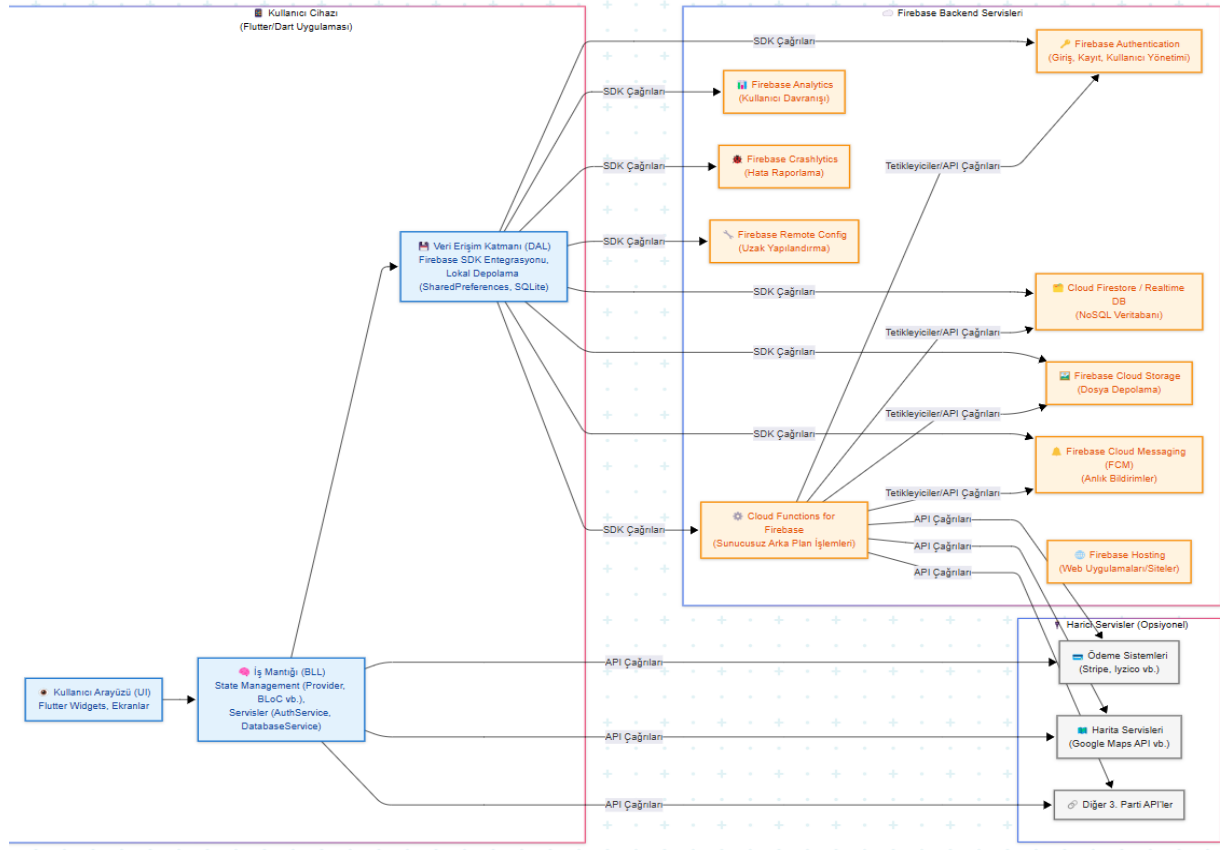
Uygulamanın, ilerleyen süreçte yeni içerikler ve fonksiyonlar eklenerek geliştirilmeye uygun bir yapıda olması sağlanmıştır.

Proje, çocukların bilgisayar öğrenimi alanındaki bilgi ve deneyimlerin paylaşılmasına ve genç nesillerin teknolojiye olan ilgisinin artırılmasına katkı sağlamıştır.

Proje, araştırmacı yetiştirilmesi ve bu alanda yeni projelerin oluşturulmasına zemin hazırlayabilecek bir potansiyele sahiptir.

Projeye ilişkin görseller aşağıda belirtilmektedir.

Bu kısımda uygulamanın genel işleyiş yapısı şematize edilmiştir.



Giriş Yap

Email

Şifre

Giriş Yap

[Hesabınız yok mu? Kayıt Olun](#)

veya sosyal medya ile giriş yap

 Google

 Facebook

 Instagram



Oyunlar



Donanım Eşleştirme



Klavye Tuş Bulmaca



İşletim Sistemi Yarışı



Yazılım mı Donanım
mı?



Dersler



Quiz



Başarı



Dükkan



Oyunlar



Hakkımızda



Ayarlar



Quiz



Dersler



Quiz



Başarı



Dükkan



Oyunlar



Hakkımızda



Ayarlar



İşletim Sistemi



Bir işletim sistemi örneği nedir?

Windows

Linux

MacOS

Tüm yukarıdakiler



Hakkımızda



Bilgisayar Öğreniyoruz

Eğitim Platformu

Vizyonumuz

Teknoloji okuryazarlığını küçük yaşlarda kazandırarak çocuklara dijital dünyada özgüvenli bireyler yetiştirmek.

Misyonumuz

8-12 yaş arası çocuklara temel bilgisayar becerilerini eğlenceli ve öğretici yollarla sunmak.

500+

Öğrenci

5 Yıl

Deneyim

4.8

Puan

Hakkımızda

Bilgisayar Öğreniyoruz, çocukların erken yaşta bilgisayarla tanışmasını sağlar. Uzman eğitim kadromuz ile eğlenceli ve güvenli içerikler sunuyoruz.

Lokasyon

Harita sadece mobil cihazlarda görüntülenebilir.

Bizi Takip Edin



İletişim

info@bilgisayarogreniyoruz.com



Dersler



Quiz



Başarı



Dükkan



Oyunlar



Hakkımızda



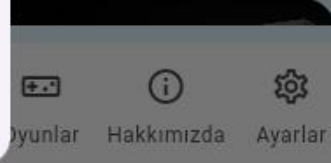
Ayarlar



eslem@gmail.com

-  Ana Sayfa
-  Quiz Sayfası
-  Başarılarım
-  Dükkan
-  Oyunlar
-  Hakkımızda
-  Ayarlar

 Çıkış Yap



```

lib > pages > lessons_page.dart > LessonsPage > lessons
1 import 'package:flutter/material.dart';
2 import 'lesson_detail_page.dart';
3
4
5
6
7 class LessonsPage extends StatelessWidget {
8   final List<Map<String, dynamic>> lessons = [
9     {
10       'title': 'Bilgisayar Nedir?',
11       'image': 'https://cdn-icons-png.flaticon.com/512/843/843286.png',
12       'background': 'https://i.ytimg.com/vi/X24h9MydL4/maxresdefault.jpg',
13       'content': '',
14     },
15     //Bilgisayar Nedir?
16     Bilgisayar, Bilgiyi işleyebilen, saklayabilen ve gerektiğinde tekrar kullanıma sunabilen **elektronik bir cihazdır**. Günlük yaşamımızda hemen her alanında yer almaktadır: okulda, evde, işte hatta ceplerinizde bile! Peki bu cihazlar nasıl çalışır?
17
18     //Bilgisayarın Temel Yapı Bileşenleri
19
20     Bilgisayarlar iki ana bölüme ayrılır:
21
22     1. **Donanım (Hardware)**: Gözle görebildiğimiz fiziksel parçalar. Örneğin: klavye, fare, ekran, işlemci, anakart, RAM, hard disk.
23     2. **Yazılım (Software)**: Bilgisayarı yöneten programlar ve uygulamalardır. Örneğin: işletim sistemleri (Windows, Linux), internet tarayıcıları, oyunlar, ofis programları.
24
25     Bu iki yapı birlikte çalışarak bilgisayarın işlev görmesini sağlar.
26
27     //Bilgisayar Nasıl Çalışır?
28
29     Bilgisayarlar **girdi** alır (örneğin klavyeden yazılan), bu girdiyi **işler** (işlemci ve RAM kullanarak), ve sonra bir **çıkış** üretir (örneğin ekrana yazılanı yazmak). Bu süreç **Girdi-İşleme-Çıkış Döngüsü** denir.
30
31     //Kısa Tarihçe
32
33     İlk bilgisayarlar 1940'lerde ortaya çıktı. O dönemde oda büyüklüğündeydiler ve yalnızca basit matematiksel işlemleri yapabiliyorlardı. Bugün ise cebinizde taşımaya kadar küçüldüler ve neredeyse her şeyi yapabiliyorlar: oyun oynamak, film izlemek, alışveriş yapmak...
34
35     //Günümüzde Bilgisayarların Kullanım Alanları
36
37     - Eğitim (e-öğrenim, dersler, araştırmalar)
38     - Sağlık (hasta kayıtları, tıbbi sistemler)
39     - Eğlence (oyunlar, müzik, film)
40     - Ulaştırma (uçaklar, tren sistemleri)
41     - İletişim (e-posta, görüntülü konuşma)
42     - Bilim (veri analizi, simülasyonlar)
43
44     //Destekleyici Video İçeriği:
45
46
47     //Unutma!
48     Bilgisayarlar sadece bir araçtır. Onlara ne yapmaları gerektiğini insanlar söyler. Aslında önemli olan, bu araçları **doğru, güvenli ve yaratıcı** şekilde kullanabilmektir.
49
50     'videoUrl': 'https://youtu.be/m4Bq98gEdV',
51     'question': 'Bilgisayarlarla ilgili aşağıdakilerden hangisi DOĞRUDUR?',
52     'options': [
53       'Bilgisayarlar sadece yazı yazmak için kullanılır.',
54       'Bilgisayarlar sadece internet bağlantısı olan yerlerde çalışır.',
55       'Bilgisayar veri işleyebilen elektronik bir cihazdır.',
56       'Bilgisayar yalnızca oyun oynamaya yarar.'
57     ],
58     'correct': 2,
59   },
60

```

```

lib > services > auth_service.dart > AuthService > getUserPointsByEmail
1 // ignore_for_file: avoid_print
2
3 import 'package:firebase_auth/firebase_auth.dart';
4 import 'package:cloud_firestore/cloud_firestore.dart';
5 import 'package:google_sign_in/google_sign_in.dart'; // Google Sign-In Eklendi
6
7 class AuthService {
8   final FirebaseAuth _auth = FirebaseAuth.instance;
9   final FirebaseFirestore _firestore = FirebaseFirestore.instance;
10
11   // Giriş Yap
12   Future<User?> signIn(String email, String password) async {
13     try {
14       UserCredential result = await _auth.signInWithEmailAndPassword(
15         email: email,
16         password: password,
17       );
18       return result.user;
19     } catch (e) {
20       print("Giriş Hatası: $e");
21       return null;
22     }
23   }
24
25   // Google ile Giriş Yap
26   Future<User?> signInWithGoogle() async { // Google Sign-In Eklendi
27     try {
28       final GoogleSignInAccount? googleUser = await GoogleSignIn().signIn();
29       if (googleUser == null) return null;
30
31       final GoogleSignInAuthentication googleAuth = await googleUser.authentication;
32
33       final credential = GoogleAuthProvider.credential(
34         accessToken: googleAuth.accessToken,
35         idToken: googleAuth.idToken,
36       );
37
38       final result = await _auth.signInWithCredential(credential);
39
40       // Yeni kullanıcı ise Firestore'a kayıtları ekle
41       final uid = result.user?.uid;
42       if (uid != null) {
43         final userDoc = await _firestore.collection("users").doc(uid).get();
44         if (userDoc.exists) {
45           await _firestore.collection("points").doc(uid).set({'point': 0});
46           await _firestore.collection("purchased_avatars").doc(uid).set({'avatars': []});
47           await _firestore.collection("users").doc(uid).set({'email': result.user?.email});
48         }
49       }
50
51       return result.user;
52     } catch (e) {
53       print("Google Sign-In Error: $e");
54       return null;
55     }
56   }
57
58   // Kayıt Ol
59   Future<User?> registerWithEmailPassword(String email, String password) async {
60     try {
61       UserCredential result = await _auth.createUserWithEmailAndPassword(
62         email: email,
63         password: password,
64       );
65       return result.user;
66     } catch (e) {
67       print("Kayıt Hatası: $e");
68       return null;
69     }
70   }
71

```

lib > services > models > avatar_model.dart > ...

```
1 class AvatarModel {
2   final String id;
3   final String name;
4   final String imageUrl;
5   final int price;
6
7   AvatarModel({
8     required this.id,
9     required this.name,
10    required this.imageUrl,
11    required this.price,
12  });
13 }
14
```

```
lib > pages > software_or_hardware_game.dart > _SoftwareOrHardwareGameState > build
1 import 'package:flutter/material.dart';
2
3 class SoftwareOrHardwareGame extends StatefulWidget {
4   const SoftwareOrHardwareGame({super.key});
5
6   @override
7   State<SoftwareOrHardwareGame> createState() => _SoftwareOrHardwareGameState();
8 }
9
10 class _SoftwareOrHardwareGameState extends State<SoftwareOrHardwareGame> {
11   final List<Map<String, String>> items = [
12     {'name': 'Klayve', 'type': 'donarım'},
13     {'name': 'Photoshop', 'type': 'yazılım'},
14     {'name': 'Fare', 'type': 'donarım'},
15     {'name': 'Windows', 'type': 'yazılım'},
16     {'name': 'Monitor', 'type': 'donarım'},
17     {'name': 'Excel', 'type': 'yazılım'},
18     {'name': 'Hoparlör', 'type': 'donarım'},
19     {'name': 'Word', 'type': 'yazılım'},
20     {'name': 'Anakart', 'type': 'donarım'},
21     {'name': 'Linux', 'type': 'yazılım'},
22     {'name': 'İşlemci', 'type': 'donarım'},
23     {'name': 'Antivirüs', 'type': 'yazılım'},
24     {'name': 'Tarayıcı', 'type': 'yazılım'},
25     {'name': 'RAM', 'type': 'donarım'},
26     {'name': 'macOS', 'type': 'yazılım'},
27   ];
28
29   int currentIndex = 0;
30   int score = 0;
31
32   void handleAnswer(String guess) {
33     String correct = items[currentIndex]['type'];
34     if (guess == correct) {
35       score++;
36       ScaffoldMessenger.of(context).showSnackBar(SnackBar(
37         content: Text("Doğru! 🟢"),
38         backgroundColor: Colors.green,
39         duration: Duration(milliseconds: 500),
40       )); // SnackBar
41     } else {
42       ScaffoldMessenger.of(context).showSnackBar(SnackBar(
43         content: Text("Yanlış! 🟡 Bu bir $correct"),
44         backgroundColor: Colors.red,
45         duration: Duration(milliseconds: 500),
46       )); // SnackBar
47     }
48
49     setState(() {
50       currentIndex++;
51     });
52
53     if (currentIndex >= items.length) {
54       Future.delayed(Duration(milliseconds: 600), _showResultDialog);
55     }
56   }
57 }
```

```

1 import 'package:flutter/material.dart';
2 import 'dart:math';
3
4 class KeyboardGame extends StatefulWidget {
5   const KeyboardGame({super.key});
6
7   @override
8   State<KeyboardGame> createState() => _KeyboardGameState();
9 }
10
11 class _KeyboardGameState extends State<KeyboardGame> {
12   final List<String> keys = [
13     'Q', 'W', 'E', 'R', 'T', 'Y', 'U', 'I', 'O', 'P',
14     'A', 'S', 'D', 'F', 'G', 'H', 'J', 'K', 'L',
15     'Z', 'X', 'C', 'V', 'B', 'N', 'M',
16   ];
17
18   String targetKey = '';
19   int score = 0;
20
21   @override
22   void initState() {
23     super.initState();
24     _setNewKey();
25   }
26
27   void _setNewKey() {
28     final random = Random();
29     setState(() {
30       targetKey = keys[random.nextInt(keys.length)];
31     });
32   }
33
34   void _checkKey(String selectedKey) {
35     if (selectedKey == targetKey) {
36       score++;
37       ScaffoldMessenger.of(context).showSnackBar(SnackBar(
38         content: Text("Doğru! $selectedKey tuşuna buldun 🎉"),
39         backgroundColor: Colors.green,
40         duration: Duration(milliseconds: 800),
41       )); // SnackBar
42     } else {
43       ScaffoldMessenger.of(context).showSnackBar(SnackBar(
44         content: Text("Yanlış! Bu $selectedKey tuşu ❌"),
45         backgroundColor: Colors.red,
46         duration: Duration(milliseconds: 800),
47       )); // SnackBar
48     }
49     _setNewKey();
50   }
51
52   @override
53   Widget build(BuildContext context) {
54     return Scaffold(
55       appBar: AppBar(title: Text(" Klavye Tuş Bulmaca")),
56       body: Padding(
57         padding: const EdgeInsets.all(16),
58         child: Column(
59           children: [
60             SizedBox(height: 20),
61             Text("AsaBuldukları tuşu bul!", style: TextStyle(fontSize: 20)),
62

```

```

lib > pages > os_race_game.dart > _OsRaceGameState > options
1 import 'package:flutter/material.dart';
2
3 class OsRaceGame extends StatefulWidget {
4   const OsRaceGame({super.key});
5
6   @override
7   State<OsRaceGame> createState() => _OsRaceGameState();
8 }
9
10 class _OsRaceGameState extends State<OsRaceGame> {
11   final List<Map<String, dynamic>> options = [
12     {'name': 'Windows', 'isOs': true, 'emoji': '🪟'},
13     {'name': 'Excel', 'isOs': false, 'emoji': '📊'},
14     {'name': 'Linux', 'isOs': true, 'emoji': '🐧'},
15     {'name': 'Photoshop', 'isOs': false, 'emoji': '🖼️'},
16     {'name': 'Android', 'isOs': true, 'emoji': '🤖'},
17     {'name': 'Word', 'isOs': false, 'emoji': '📝'},
18     {'name': 'macOS', 'isOs': true, 'emoji': '🍏'},
19     {'name': 'Chrome', 'isOs': false, 'emoji': '🌐'},
20   ];
21
22   int score = 0;
23   int round = 1;
24   List<Map<String, dynamic>> currentChoices = [];
25
26   @override
27   void initState() {
28     super.initState();
29     _loadNextRound();
30   }
31
32   void _loadNextRound() {
33     options.shuffle();
34     currentChoices = options.take(4).toList();
35   }
36
37   void _handleChoice(Map<String, dynamic> choice) {
38     setState(() {
39       if (choice['isOs']) {
40         score++;
41         ScaffoldMessenger.of(context).showSnackBar(SnackBar(
42           content: Text("Doğru! ${choice['name']} bir işletim sistemidir 🎉"),
43           backgroundColor: Colors.green,
44         )); // SnackBar
45       } else {
46         ScaffoldMessenger.of(context).showSnackBar(SnackBar(
47           content: Text("Yanlış! ${choice['name']} işletim sistemi değildir ❌"),
48           backgroundColor: Colors.red,
49         )); // SnackBar
50       }
51       round++;
52
53       if (round > 5) {
54         _showFinalDialog();
55       } else {
56         _loadNextRound();
57       }
58     });
59   }
60 }
61

```

```

lib > pages > settings_page.dart > SettingsPage > build
1 import 'package:flutter/material.dart';
2 import 'package:firebase_auth/firebase_auth.dart';
3
4 class SettingsPage extends StatelessWidget {
5   const SettingsPage({super.key});
6
7   @override
8   Widget build(BuildContext context) {
9     final user = FirebaseAuth.instance.currentUser;
10
11     return Scaffold(
12       body: ListView(
13         children: [
14           // Kullanıcının mail adresi gösteriliyor
15           ListTile(
16             leading: const CircleAvatar(
17               backgroundImage: NetworkImage('assets/logo.png'),
18               radius: 30,
19             ), // CircleAvatar
20             title: Text(
21               user?.email ?? "Bilinmeyen kullanıcı",
22               style: const TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
23             ), // Text
24             subtitle: const Text("Kayıt e-posta adresinizi"),
25           ), // ListTile
26           const Divider(),
27           // Hesap ayarları
28           _buildSettingsGroup("Hesap", [
29             _buildFile(Icons.privacy_tip, "Gizlilik", "Veri ve gizlilik seçenekleri"),
30             _buildFile(Icons.security, "Güvenlik", "Şifre koruma, doğrulama"),
31             _buildFile(Icons.lock, "Şifreyi Değiştir", "Yeni şifre belirle"),
32           ]),
33           // Uygulama ayarları
34           _buildSettingsGroup("Uygulama", [
35             _buildFile(Icons.brightness_6, "Tema", "Açık veya Koyu"),
36             _buildFile(Icons.language, "Dil", "Şu an: Türkçe"),
37           ]),
38           // Bildirimler
39           _buildSettingsGroup("Bildirimler", [
40             _buildFile(Icons.notifications, "Bildirim Ayarları", "Sesler, titreşim ve daha fazlası"),
41           ]),
42           // Yardım
43           _buildSettingsGroup("Yardım", [
44             _buildFile(Icons.help_outline, "SSS", "Sıkça sorulan sorular"),
45             _buildFile(Icons.feedback_outlined, "Geri Bildirim Gönder", "Fikrinizi paylaşın"),
46           ]),
47         ],
48       ), // ListView
49     ); // Scaffold
50   }
51 }

```

```

lib > pages > register_page.dart > RegisterPage
1 class RegisterPage extends StatelessWidget {
2
3   @override
4   Widget build(BuildContext context) {
5     return Scaffold(
6       backgroundColor: const Color(0xFFE3F2FD),
7       body: Center(
8         child: SingleChildScrollView(
9           child: Container(
10             margin: const EdgeInsets.all(24),
11             padding: const EdgeInsets.symmetric(horizontal: 24, vertical: 32),
12             decoration: BoxDecoration(
13               color: Colors.white,
14               borderRadius: BorderRadius.circular(20),
15               boxShadow: const [
16                 BoxShadow(
17                   color: Colors.black12,
18                   blurRadius: 20,
19                   offset: Offset(0, 10),
20                 ), // BoxShadow
21               ],
22             ), // BoxDecoration
23             child: Form( // Form eklendi
24               key: _formKey,
25               child: Column(
26                 mainAxisAlignment: MainAxisAlignment.min,
27                 children: [
28                   const Text(
29                     "Kayıt Ol",
30                     style: TextStyle(
31                       fontSize: 26,
32                       fontWeight: FontWeight.bold,
33                       color: Colors.deepPurple,
34                     ), // TextStyle
35                   ), // Text
36                   const SizedBox(height: 24),
37                   TextFormField(
38                     controller: emailController,
39                     decoration: InputDecoration(
40                       hintText: "Email",
41                       prefixIcon: const Icon(Icons.email),
42                       filled: true,
43                       fillColor: Colors.grey[100],
44                       border: OutlineInputBorder(
45                         borderRadius: BorderRadius.circular(16),
46                         borderSide: BorderSide.none,
47                       ), // OutlineInputBorder
48                     ), // InputDecoration
49                     validators: (value) {
50                       if (value == null || value.isEmpty) {
51                         return 'Email boş olamaz';
52                       }
53                       if (!value.contains('@') || !value.contains('.')) {
54                         return 'Geçerli bir email girin';
55                       }
56                       return null;
57                     },
58                   ), // TextFormField
59                   const SizedBox(height: 16),
60                   TextFormField(
61                     controller: passwordController,
62                     obscureText: true,

```

```

lib > pages > achievements_page.dart > _AchievementsPageState > maxPoint
1 // Başta
2 import 'package:flutter/material.dart';
3 import 'package:bilgisayar_ogreniyoruz/services/auth_service.dart';
4
5 class AchievementsPage extends StatefulWidget {
6   const AchievementsPage({super.key});
7
8   @override
9   State<AchievementsPage> createState() => _AchievementsPageState();
10 }
11
12 class _AchievementsPageState extends State<AchievementsPage> {
13   final AuthService auth = AuthService();
14   int point = 0;
15   bool isLoading = true;
16   final int maxPoint = 10000;
17   String currentEmail = '';
18   String searchQuery = '';
19
20   List<String> purchasedAvatarIds = [];
21   List<Map<String, dynamic>> allUsers = [];
22
23   final List<AvatarModel> avatars = [
24     AvatarModel(
25       id: '1',
26       name: 'Robot',
27       imageUrl:
28         'https://t4.ftcdn.net/jpg/05/43/48/93/360_F_943489384_zq3u5kkeffJPV311f6t8IKx0M31vx5z.jpg',
29       price: 30,
30     ),
31     AvatarModel(
32       id: '2',
33       name: 'Kedi',
34       imageUrl:
35         'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9Gc5Kfz357n7iy1-E9EZGu0G13f8a3ag0436wks',
36       price: 50,
37     ),
38     AvatarModel(
39       id: '3',
40       name: 'Ninja',
41       imageUrl:
42         'https://png.pngtree.com/png-clipart/20240504/original/pngtree-ninja-mascot-cartoon-avatar-character-design-png-image_15085817.png',
43       price: 70,
44     ),
45     AvatarModel(
46       id: '4',
47       name: 'Uzay',
48       imageUrl:
49         'https://img.pixbest.com/origin/09/28/36/45QpIkbEsTrqE.pnglw700wp',
50       price: 40,
51     ),
52     AvatarModel(
53       id: '5',
54       name: 'Super Kahraman',
55       imageUrl:
56         'https://st1.depositphotos.com/5934848/12048/v/450/depositphotos_120489392-stock-illustration-superhero-avatar-superman-comic-design.jpg',
57       price: 60,
58     ),
59     AvatarModel(
60       id: '6',
61       name: 'Dinozor',
62       imageUrl:
63         'https://png.pngtree.com/png-clipart/20231021/original/pngtree-clip-art-illustration-of-cute-dinosaur-cartoon-character-for-kids-png-image_13395063.png',
64       price: 35,
65     ),
66     AvatarModel(
67       id: '7',
68       name: 'Bilinmeyen',
69       imageUrl:
70         'https://png.pngtree.com/png-clipart/20250428/original/pngtree-a-high-quality-3d-render-of-young-female-scientist-avatar-png-image_28860475.png',
71       price: 45,
72     ),
73   ];

```

```

lib > pages > login_page.dart > _LoginPageState > _login
1 import 'package:flutter/material.dart';
2 import 'package:bilgisayar_ogreniyoruz/pages/register_page.dart';
3 import 'package:bilgisayar_ogreniyoruz/pages/home_page.dart';
4 import 'package:bilgisayar_ogreniyoruz/services/auth_service.dart';
5
6 class LoginPage extends StatefulWidget {
7   const LoginPage({super.key});
8
9   @override
10  State<LoginPage> createState() => _LoginPageState();
11 }
12
13 class _LoginPageState extends State<LoginPage> {
14   final TextEditingController emailController = TextEditingController();
15   final TextEditingController passwordController = TextEditingController();
16   final GlobalKey<FormState> _formKey = GlobalKey<FormState>();
17   final AuthService auth = AuthService();
18
19   bool isValidEmail(String email) {
20     final emailRegex = RegExp(r"^[w-\\.]+@[w-\\.]+(\\.[w-\\.]{2,4})$");
21     return emailRegex.hasMatch(email);
22   }
23
24   Future<void> _login() async {
25     if (_formKey.currentState!.validate()) {
26       String email = emailController.text.trim();
27       String password = passwordController.text.trim();
28
29       final user = await auth.signIn(email, password);
30
31       if (user != null) {
32         Navigator.pushReplacement(
33           // ignore: use_build_context_synchronously
34           context,
35           MaterialPageRoute(builder: (context) => const HomePage()),
36         );
37       } else {
38         // ignore: use_build_context_synchronously
39         ScaffoldMessenger.of(context).showSnackBar(
40           const SnackBar(
41             content: Text('Giriş başarısız. Lütfen bilgileri kontrol edin.'),
42             // SnackBar
43           ),
44         );
45       }
46     }
47   }

```



```

60
61 static const FirebaseOptions ios = FirebaseOptions(
62     apiKey: 'AIzaSyCYVc4KnrmPodhq76q8KomGbhWzdPSWTTU',
63     appId: '1:529007804807:ios:be9bebe62befa1955a1322',
64     messagingSenderId: '529007804807',
65     projectId: 'bilgisayarogreniyorum-5a0dc',
66     storageBucket: 'bilgisayarogreniyorum-5a0dc.firebaseio.app',
67     iosBundleId: 'com.example.bilgisayarOgreniyoruz',
68 );
69
70 static const FirebaseOptions macos = FirebaseOptions(
71     apiKey: 'AIzaSyCYVc4KnrmPodhq76q8KomGbhWzdPSWTTU',
72     appId: '1:529007804807:ios:be9bebe62befa1955a1322',
73     messagingSenderId: '529007804807',
74     projectId: 'bilgisayarogreniyorum-5a0dc',
75     storageBucket: 'bilgisayarogreniyorum-5a0dc.firebaseio.app',
76     iosBundleId: 'com.example.bilgisayarOgreniyoruz',
77 );
78
79 static const FirebaseOptions windows = FirebaseOptions(
80     apiKey: 'AIzaSyAYaO_mb1sGwi0DX7x3do5lwPn95B64PaA',
81     appId: '1:529007804807:web:95bb9f0cf8593fe05a1322',
82     messagingSenderId: '529007804807',
83     projectId: 'bilgisayarogreniyorum-5a0dc',
84     authDomain: 'bilgisayarogreniyorum-5a0dc.firebaseio.com',
85     storageBucket: 'bilgisayarogreniyorum-5a0dc.firebaseio.app',
86     measurementId: 'G-8ZFVNTZ2N6',
87 );
88 }
89

```

```

lib > firebase_options.dart > DefaultFirebaseOptions > currentPlatform
1 // File generated by FlutterFire CLI.
2 // ignore_for_file: type=lint
3 import 'package:firebase_core/firebase_core.dart' show FirebaseOptions;
4 import 'package:flutter/foundation.dart'
5     show defaultTargetPlatform, kIsWeb, TargetPlatform;
6
7 /// Default [FirebaseOptions] for use with your Firebase apps.
8 ///
9 /// Example:
10 /// ```dart
11 /// import 'firebase_options.dart';
12 ///
13 /// await Firebase.initializeApp(
14 ///   options: DefaultFirebaseOptions.currentPlatform,
15 /// );
16 /// ```
17 class DefaultFirebaseOptions {
18   static FirebaseOptions get currentPlatform {
19     if (kIsWeb) {
20       return web;
21     }
22     switch (defaultTargetPlatform) {
23       case TargetPlatform.android:
24         return android;
25       case TargetPlatform.iOS:
26         return ios;
27       case TargetPlatform.macOS:
28         return macos;
29       case TargetPlatform.windows:
30         return windows;
31       case TargetPlatform.linux:
32         throw UnsupportedError(
33           'DefaultFirebaseOptions have not been configured for linux - '
34           'you can reconfigure this by running the FlutterFire CLI again.',
35         );
36       default:
37         throw UnsupportedError(
38           'DefaultFirebaseOptions are not supported for this platform.',
39         );
40     }
41   }
42
43   static const FirebaseOptions web = FirebaseOptions(
44     apiKey: 'AIzaSyAYaO_mb1sGwi0DX7x3do5lwPn95B64PaA',
45     appId: '1:529007804807:web:da8bfc7ba883cc8d5a1322',
46     messagingSenderId: '529007804807',
47     projectId: 'bilgisayarogreniyorum-5a0dc',
48     authDomain: 'bilgisayarogreniyorum-5a0dc.firebaseio.com',
49     storageBucket: 'bilgisayarogreniyorum-5a0dc.firebaseio.app',
50     measurementId: 'G-15G4DWXVN',
51   );
52
53   static const FirebaseOptions android = FirebaseOptions(
54     apiKey: 'AIzaSyDrKLPSSPh1Sh81VaRxc3S1e4IR_PESs8',
55     appId: '1:529007804807:android:2bc7929b6e04584c5a1322',
56     messagingSenderId: '529007804807',
57     projectId: 'bilgisayarogreniyorum-5a0dc',
58     storageBucket: 'bilgisayarogreniyorum-5a0dc.firebaseio.app',
59   );

```

```

1b > main.dart > %MyApp > @ build
1 import 'package:flutter/material.dart';
2 import 'package:firebase_core/firebase_core.dart';
3 import 'package:firebase_auth/firebase_auth.dart';
4
5 import 'firebase_options.dart';
6 import 'pages/login_page.dart';
7 import 'pages/register_page.dart';
8 import 'pages/home_page.dart';
9
10 Run [Debug] Profile
11 void main() async {
12   WidgetsFlutterBinding.ensureInitialized();
13   await Firebase.initializeApp(
14     | options: DefaultFirebaseOptions.currentPlatform,
15   );
16   runApp(const MyApp());
17 }
18
19 class MyApp extends StatelessWidget {
20   const MyApp({super.key});
21
22   @override
23   Widget build(BuildContext context) {
24     return MaterialApp(
25       debugShowCheckedModeBanner: false,
26       title: 'Bilgisayar Öğretiyorum',
27       theme: ThemeData(
28         primarySwatch: Colors.teal,
29         scaffoldBackgroundColor: Colors.lightBlue[50],
30       ),
31       textTheme: const TextTheme(
32         bodyLarge: TextStyle(
33           fontSize: 18,
34           color: Colors.black87,
35           fontWeight: FontWeight.w600,
36         ), // TextStyle
37         bodyMedium: TextStyle(
38           fontSize: 16,
39           color: Colors.black54,
40         ), // TextStyle
41       ), // TextTheme
42       appBarTheme: const AppBarTheme(
43         backgroundColor: Colors.teal,
44         elevation: 8,
45         centerTitle: true,
46         titleTextStyle: TextStyle(
47           fontSize: 24,
48           fontWeight: FontWeight.bold,
49           color: Colors.white,
50         ), // TextStyle
51       ), // IconThemeData(color: Colors.white),
52       iconTheme: const IconThemeData(color: Colors.white),
53     ), // AppBarTheme
54     elevatedButtonTheme: ElevatedButtonThemeData(
55       style: ElevatedButton.styleFrom(
56         backgroundColor: Colors.orangeAccent,
57         shape: RoundedRectangleBorder(
58           borderRadius: BorderRadius.circular(20),
59         ), // RoundedRectangleBorder
60         padding: const EdgeInsets.symmetric(horizontal: 30, vertical: 15),
61         elevation: 5,
62         shadowColor: Colors.orangeAccent.withOpacity(0.5),
63       ), // ElevatedButtonThemeData
64       textStyle: const TextStyle(
65         fontSize: 18,
66         fontWeight: FontWeight.bold,
67         color: Colors.white,
68         foregroundColor: Colors.white,
69       ), // ElevatedButtonThemeData
70     ), // ThemeData
71   ), // ThemeData
72 }

```

```

1b > pages > quiz_detail_page.dart > %QuizDetailPageState > @ build
22 class _QuizDetailPageState extends State<QuizDetailPage> {
23   Widget build(BuildContext context) {
24     return Scaffold(
25       backgroundColor: Colors.deepPurple,
26       appBar: AppBar(
27         title: Text(widget.title),
28         backgroundColor: Colors.deepPurple,
29       ), // AppBar
30       body: SafeArea(
31         child: Padding(
32           padding: const EdgeInsets.all(16),
33           child: Column(
34             crossAxisAlignment: CrossAxisAlignment.stretch,
35             children: [
36               Container(
37                 height: 180,
38                 decoration: BoxDecoration(
39                   borderRadius: BorderRadius.circular(16),
40                   image: DecorationImage(
41                     image: NetworkImage(q['image']),
42                     fit: BoxFit.cover,
43                   ), // DecorationImage
44                 ), // BoxDecoration
45               ), // Container
46               SizedBox(height: 20),
47               Card(
48                 shape: RoundedRectangleBorder(
49                   borderRadius: BorderRadius.circular(12)), // RoundedRectangleBorder
50                 color: Colors.white,
51                 child: Padding(
52                   padding: const EdgeInsets.all(16.0),
53                   child: Text(
54                     q['question'],
55                     style:
56                       TextStyle(fontSize: 20, fontWeight: FontWeight.w600),
57                   ), // Text
58                 ), // Padding
59               ), // Card
60               SizedBox(height: 20),
61               ...q['options'].map<Widget>((opt) {
62                 return Padding(
63                   padding: const EdgeInsets.symmetric(vertical: 6.0),
64                   child: ElevatedButton(
65                     onPressed: answered ? null : () => checkAnswer(opt),
66                     style: ElevatedButton.styleFrom(
67                       backgroundColor: getOptionColor(opt, correctAnswer),
68                       foregroundColor: Colors.black,
69                       minimumSize: Size(double.infinity, 50),
70                       shape: RoundedRectangleBorder(
71                         borderRadius: BorderRadius.circular(10),
72                       ), // RoundedRectangleBorder
73                       side: BorderSide(color: Colors.black12),
74                     ),
75                     child: Text(
76                       opt,
77                       style: TextStyle(fontSize: 16),
78                     ), // Text
79                   ), // ElevatedButton
80                 ); // Padding
81               }).toList(),
82             ],
83           ),
84         ),
85       ),
86     );
87   }
88 }

```

```

lib > pages > quiz_detail_page.dart > QuizDetailPageState > build
1 import 'package:bilgisayar_ogreniyoruz/services/auth_service.dart';
2 import 'package:flutter/material.dart';
3
4 class QuizDetailPage extends StatefulWidget {
5   final String title;
6   final String imageUrl;
7   final String content;
8   final List<Map<String, dynamic>> questions;
9
10  const QuizDetailPage({
11    super.key,
12    required this.title,
13    required this.imageUrl,
14    required this.content,
15    required this.questions,
16  });
17
18  @override
19  State<QuizDetailPage> createState() => _QuizDetailPageState();
20 }
21
22 class _QuizDetailPageState extends State<QuizDetailPage> {
23   final AuthService auth = AuthService();
24   int currentQuestion = 0;
25   int score = 0;
26   String? selectedOption;
27   bool answered = false;
28
29   void checkAnswer(String selected) {
30     if (answered) return;
31
32     setState(() {
33       selectedOption = selected;
34       answered = true;
35
36       if (selected == widget.questions[currentQuestion]['answer']) {
37         score += 10;
38       }
39
40       Future.delayed(Duration(seconds: 1), () {
41         if (currentQuestion < widget.questions.length - 1) {
42           setState(() {
43             currentQuestion++;
44             selectedOption = null;
45             answered = false;
46           });
47         }
48       }); // Future.delayed
49     });
50   }
51 }

```

```

lib > pages > lesson_detail_page.dart > LessonDetailPageState > build
28 class LessonDetailPageState extends State<LessonDetailPage> {
44   Widget build(BuildContext context) {
45
46     // İkerik metni
47     Container(
48       padding: const EdgeInsets.all(16),
49       decoration: BoxDecoration(
50         color: Colors.blue[50],
51         borderRadius: BorderRadius.circular(20),
52       ), // BoxDecoration
53       child: Text(
54         "■ ${widget.content}",
55         style: const TextStyle(fontSize: 18,
56           fontWeight: FontWeight.bold,
57           color: Colors.black87, // TextStyle
58           textAlign: TextAlign.justify,
59         ), // Text
60       ), // Container
61
62       const SizedBox(height: 20),
63
64       // YouTube Video Bağlantısı
65       if (widget.videoUrl.isNotEmpty) ...[
66         const Divider(thickness: 1),
67         Text(
68           "📺 Destekleyici Video",
69           style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold, color: Colors.deepOrange),
70         ), // Text
71         const SizedBox(height: 8),
72         InkWell(
73           onTap: () => _launchURL(widget.videoUrl),
74           child: Container(
75             padding: const EdgeInsets.symmetric(vertical: 12, horizontal: 16),
76             decoration: BoxDecoration(
77               color: Colors.red[50],
78               borderRadius: BorderRadius.circular(12),
79               border: Border.all(color: Colors.redAccent),
80             ), // BoxDecoration
81             child: Row(
82               mainAxisAlignment: MainAxisAlignment.min,
83               children: const [
84                 Icon(Icons.play_circle_fill, color: Colors.red,
85                   SizedBox(width: 10),
86                 Text("Videoyu İzle", style: TextStyle(fontSize: 16, color: Colors.redAccent)),
87               ], // Row
88             ), // Container
89           ), // InkWell
90         const SizedBox(height: 30),
91       ],
92     );

```

```

lib > pages > lesson_detail_page.dart > _LessonDetailPageState > build
1 import 'package:flutter/material.dart';
2 import 'package:url_launcher/url_launcher.dart';
3
4 class LessonDetailPage extends StatefulWidget {
5   final String title;
6   final String imageUrl;
7   final String content;
8   final String question;
9   final List<String> options;
10  final int correctAnswerIndex;
11  final String videoUrl;
12
13  const LessonDetailPage({
14    super.key,
15    required this.title,
16    required this.imageUrl,
17    required this.content,
18    required this.question,
19    required this.options,
20    required this.correctAnswerIndex,
21    required this.videoUrl,
22  });
23
24  @override
25  State<LessonDetailPage> createState() => _LessonDetailPageState();
26 }
27
28 class _LessonDetailPageState extends State<LessonDetailPage> {
29   int? selectedAnswer;
30
31   Future<void> _launchURL(String url) async {
32     final uri = Uri.parse(url);
33     if (await canLaunchUrl(uri)) {
34       await launchUrl(uri, mode: LaunchMode.externalApplication);
35     } else {
36       // ignore: use_build_context_synchronously
37       ScaffoldMessenger.of(context).showSnackBar(
38         SnackBar(content: Text('Video bağlantısı açılmadı')),
39       );
40     }
41   }
42
43   @override
44   Widget build(BuildContext context) {
45     return Scaffold(
46       appBar: AppBar(
47         title: Text(widget.title),
48         backgroundColor: Colors.deepPurple,
49       ), // AppBar
50       body: SingleChildScrollView(
51         padding: const EdgeInsets.all(16),

```

```

lib > pages > home_page.dart > _HomePageState > build
22 class _HomePageState extends State<HomePage> {
23   Widget build(BuildContext context) {
24     // NavigationBar
25     bottomNavigationBar: NavigationBar(
26       selectedIndex: _selectedIndex,
27       onDestinationSelected: enum NavigationDestinationLabelBehavior
28       height: 70,
29       backgroundColor: package:flutter/src/material/navigation_bar.dart
30       indicatorColor: Specifies when each [NavigationDestination]'s label should appear.
31       surfaceTintColor: This is used to determine the behavior of [NavigationBar]'s destinations.
32       elevation: 5,
33       labelBehavior: NavigationDestinationLabelBehavior.alwaysShow,
34       destinations: const [
35         NavigationDestination(
36           icon: Icon(Icons.book_outlined),
37           selectedIcon: Icon(Icons.book, color: Colors.teal),
38           label: 'Dersler',
39         ), // NavigationDestination
40         NavigationDestination(
41           icon: Icon(Icons.quiz_outlined),
42           selectedIcon: Icon(Icons.quiz, color: Colors.deepPurple),
43           label: 'Quiz',
44         ), // NavigationDestination
45         NavigationDestination(
46           icon: Icon(Icons.emoji_events_outlined),
47           selectedIcon: Icon(Icons.emoji_events, color: Colors.orange),
48           label: 'Bazılar',
49         ), // NavigationDestination
50         NavigationDestination(
51           icon: Icon(Icons.store_outlined),
52           selectedIcon: Icon(Icons.store, color: Colors.green),
53           label: 'Dükkan',
54         ), // NavigationDestination
55         NavigationDestination(
56           icon: Icon(Icons.videogame_asset_outlined),
57           selectedIcon: Icon(Icons.videogame_asset, color: Colors.redAccent),
58           label: 'Oyunlar',
59         ), // NavigationDestination
60         NavigationDestination(
61           icon: Icon(Icons.info_outline),
62           selectedIcon: Icon(Icons.info, color: Colors.indigo),
63           label: 'Hakkımızda',
64         ), // NavigationDestination
65         NavigationDestination(
66           icon: Icon(Icons.settings_outlined),
67           selectedIcon: Icon(Icons.settings, color: Colors.grey),
68           label: 'Ayarlar',
69         ), // NavigationDestination
70       ],
71     ), // NavigationBar

```

```

lib > pages > home_page.dart > _HomePageState > build
22 class _HomePageState extends State<HomePage> {
51
52 void _playWelcomeSound() async {
53   await assetsAudioPlayer.open(
54     Audio('assets/sounds/ttsMP3.com_VoiceText_2025-4-26_19-21-54.mp3'),
55     autoStart: true,
56     showNotification: false,
57   );
58 }
59
60 @override
61 void initState() {
62   super.initState();
63   _playWelcomeSound();
64 }
65
66 @override
67 void dispose() {
68   assetsAudioPlayer.dispose();
69   super.dispose();
70 }
71
72 @override
73 Widget build(BuildContext context) {
74   final User? user = FirebaseAuth.instance.currentUser; // ● Kullanıcıyı al
75
76   return Scaffold(
77     appBar: AppBar(
78       backgroundColor: Colors.deepPurple,
79       title: Text(_pageTitles[_selectedIndex]),
80     ), // AppBar
81     drawer: Drawer(
82       child: ListView(
83         padding: EdgeInsets.zero,
84         children: <Widget>[
85           UserAccountsDrawerHeader(
86             accountName: null, // Ad gösterilmiyor
87             accountEmail: Text(user?.email ?? 'E-posta bulunamadı'), // ● Giriş yapan kullanıcının e-postası
88             currentAccountPicture: CircleAvatar(
89               backgroundImage: NetworkImage('assets/logo.png'),
90             ), // CircleAvatar
91           ), // UserAccountsDrawerHeader
92           ListTile(
93             leading: Icon(Icons.home),
94             title: Text('Ana Sayfa'),
95             onTap: () {
96               setState(() {
97                 _selectedIndex = 0;
98               });
99             },
100             Navigator.pop(context);
101

```

```

lib > pages > home_page.dart > _HomePageState > build
1 import 'package:bilgisayar_ogreniyoruz/pages/about.dart';
2 import 'package:bilgisayar_ogreniyoruz/pages/achievements_page.dart';
3 import 'package:bilgisayar_ogreniyoruz/pages/lessons_page.dart';
4 import 'package:flutter/material.dart';
5 import 'quiz_page.dart';
6 import 'games_page.dart';
7 import 'package:bilgisayar_ogreniyoruz/pages/settings_page.dart';
8 import 'package:bilgisayar_ogreniyoruz/pages/shop_page.dart';
9 import 'package:assets_audio_player/assets_audio_player.dart';
10 import 'package:firebase_auth/firebase_auth.dart'; // Firebase çıkış ve kullanıcı bilgisi için eklendi
11
12
13
14 class HomePage extends StatefulWidget {
15   const HomePage({super.key});
16
17   @override
18   // ignore: library_private_types_in_public_api
19   _HomePageState createState() => _HomePageState();
20 }
21
22 class _HomePageState extends State<HomePage> {
23   final assetsAudioPlayer = AssetsAudioPlayer();
24   int _selectedIndex = 0;
25
26   final List<Widget> _pages = <Widget>[
27     LessonsPage(), // 0
28     QuizPage(), // 1
29     AchievementsPage(),
30     ShopPage(studentPoints: 100), // 3
31     GamesPage(), // 4
32     HakkimizdaPage(), // 5
33     SettingsPage(), // 6
34   ]; // <Widget>[]
35
36   final List<String> _pageTitles = [
37     'Dersler',
38     'Quiz',
39     'Başarıları',
40     'Dükkan',
41     'Oyunlar',
42     'Hakkımızda',
43     'Ayarlar',
44   ];
45
46   void _onItemTapped(int index) {
47     setState(() {
48       _selectedIndex = index;
49     });
50   }
51

```

```

lib > pages > quiz_page.dart > QuizPage > quizzes
1 import 'package:flutter/material.dart';
2 import 'quiz_detail_page.dart';
3
4 // ignore: use_key_in_widget_constructors
5 class QuizPage extends StatelessWidget {
6   final List<Map<String, dynamic>> quizzes = [
7     {
8       "title": "Bilgisayarın Beyni",
9       "image": "https://cdn-icons-png.flaticon.com/512/2721/2721286.png",
10      "background": "https://i.ytimg.com/vi/ub10bZGJlKQ/hq728.jpg?sqp=-oaymwEhCK4FEIIDSFrYq4qpAxpMIAUAAAAAAGE1AADIQj0AgCjD&rs=AOn4CLBn-ZI2iHojs8LEuagB2HTsFQdX1A",
11      "content": "Bilgisayarın beyni olan CPU hakkında bilgi edin.",
12      "questions": [
13        {
14          "image": "https://cdn-icons-png.flaticon.com/512/2721/2721286.png",
15          "question": "Bilgisayarın beyni olarak bilinen parça hangisidir?",
16          "options": ["RAM", "CPU", "Harddisk", "Mouse"],
17          "answer": "CPU"
18        },
19        {
20          "image": "https://cdn-icons-png.flaticon.com/512/727/727399.png",
21          "question": "Hangi cihaz bir ses birimidir?",
22          "options": ["Klavye", "Hoparlör", "Fare", "Mikrofon"],
23          "answer": "Hoparlör"
24        },
25        {
26          "image": "https://cdn-icons-png.flaticon.com/512/888/88879.png",
27          "question": "CPU ne işe yarar?",
28          "options": [
29            "Bilgisayarı tasıma",
30            "Bilgisayara güç sağlama",
31            "Bilgileri işleme",
32            "Ekranı gösterme"
33          ],
34          "answer": "Bilgileri işleme"
35        },
36        {
37          "image": "https://cdn-icons-png.flaticon.com/512/1063/1063393.png",
38          "question": "CPU olmadan bilgisayar çalışabilir mi?",
39          "options": ["Evet", "Hayır", "Bazen", "Sadece internetle"],
40          "answer": "Hayır"
41        },
42        {
43          "image": "https://cdn-icons-png.flaticon.com/512/2306/2306154.png",
44          "question": "Aşağıdakilerden hangisi bir giriş birimidir?",
45          "options": ["Ekran", "Yazıcı", "Klavye", "Hoparlör"],
46          "answer": "Klavye"
47        },
48        {
49          "image": "https://cdn-icons-png.flaticon.com/512/3845/3845731.png",
50          "question": "CPU'nun diğer adı nedir?",
51          "options": ["Ekran kartı", "İşlemci", "Depolama", "Kasa"],
52          "answer": "İşlemci"
53        },
54        {
55          "image": "https://cdn-icons-png.flaticon.com/512/2920/2920277.png",
56          "question": "Bilgisayarda verileri geçici olarak saklayan parça hangisidir?",
57          "options": ["CPU", "RAM", "SSD", "Fare"],
58          "answer": "RAM"
59        },
60      ],
61    },
62  ];
63 }

```

5. Proje ile ilgili harcama kalemleri hakkında ayrıntılı bilgi

Ekranda görülen ekran resmi projeyi hayata geçirebilmek için gerekli olan malzeme listesinde belirtmiş olduğumuz “Yazıcı” adlı malzemenin satın alım faturasıdır.

amazon.com.tr

Amazon Turkey Perakende Hizmetleri Limited Şirketi

Esentepe MahallesiBahar Sk. No: 13/52

Şişli-İstanbul,34394,Turkey

Web Sitesi -Web site:http://www.amazon.com.tr/

Vergi Dairesi -Tax Office:Zincirlikuyu

Vergi Kimlik No -Tax Number:0680972288

Ticaret Sicil No -Trade Registration Number:91626-5

Mersis No:068097228800001

Teslimat Adresi -Delivery Address:

Bartın Üniversitesi, Fen fakültesi giriş kat ofis no z33

Bartın,Kutlubey(Tuzcularköyü),74100

TR

Fatura Adresi -Invoice Address:

emir köç

Kemalpaşa Mahallesi Ali Koç Sokak Serdivan Sakarya, 8 daire 2

Sakarya,Sakarya,54050

TR

TC Kimlik No -TC Identity Number:44068314976



e-Arşiv Fatura
(e-Archive Invoice)

[Handwritten signature]



Sipariş No -Order No	407-3236279-5880153
Özelleştirme No -Customization Id	TR1.2
Fatura Senaryo -Invoice Scenario	EARSIVFATURA
Fatura Tipi -Invoice Type	SATIS
Fatura No -Invoice No	3002024016672256
Fatura Tarihi -Invoice Date	04 / 06 / 2024
Düzenleme Zamanı -Issue Time	16:26:04

ETTN: 877e0ac1-1195-38ae-82a9-4757fe9b10a5

*Bu tutarlara K.D.V dahil değildir.(These amounts are exclusive of VAT.)

No	MKT (QTY)	Ürün Kodu (Product ID)	Mal Hizmet Cinsi (Type of Goods/Services)	Birim Fiyat (Unit Price)	Birim Fiyat* (Unit Price)	Genel Tutar* (Gross Amount)	İskonto* (Discount)	KDV Oranı (VAT Rate)	KDV Tutarı (VAT Amount)	Toplam Tutar (Total Amount)	MKT. İade (QTY. Returned)	İade Tutarı (Return Amount)
1	1	B09SKZY1WZ	HP Smart Tank 516 Wireless All-in-One (3YW70A) IMEI:3YW70A	5,700,00TL	4,750,00TL	4,750,00TL	83,33TL	20%	933,33TL	5,600,00TL		

Toplam Tutar -Total Amount 5,600,00TL

Mal Hizmet Tutarı* -Total Goods/Services Amount 4,666,67TL

Toplam Vergi Tutarı -Total Tax Amount 933,33TL

Ödenecek Tutar -Total Payable Amount 5,600,00TL

KDV Matrahı* (VAT Base)	KDV Oranı (VAT Rate)	KDV Tutarı (VAT Amount)
4,666,67TL	20%	933,33TL

Ödeme Şekli - Payment Type:KREDİKARTI/BANKAKARTI

e-Arşiv izni kapsamında elektronik ortamda iletilmiştir.

(e-Archive invoice was transmitted electronically.)

Bu satış internet üzerinden yapılmıştır.

(This sale was made over the internet.)

İADE BÖLÜMÜ (Return Section)		
İade, web sitesindeki hesabınız üzerinden başlatılmalıdır. Lütfen hesabınızı ziyaret ediniz: www.amazon.com.tr (Your return account must be initiated online. Please visit Your Account at www.amazon.com.tr)		
İade Eden Adı (Sender's Name)	Adres (Address)	İmza (Signature)

Amazon Turkey Perakende Hizmetleri Limited Şirketi
Esentepe MahallesiBahar Sk. No: 13/52
Şişli-İstanbul,34394,Turkey
Web Sitesi -Web site:http://www.amazon.com.tr/
Vergi Dairesi -Tax Office:Zincirlikuyu
Vergi Kimlik No -Tax Number:0680972288
Ticaret Sicil No -Trade Registration Number:91626-5
Mersis No:68097228800001



Teslimat Adresi -Delivery Address:
Bartın Üniversitesi, Fen fakültesi giriş kat ofis no z33
Bartın,Kutlubey(Tuzcularköyü),74110
TR

Fatura Adresi -Invoice Address:
emir koç
Kemalpaşa Mahallesi Ali Koç Sokak Serdivan Sakarya, 8 daire 2
Sakarya,Sakarya,54050
TR
TC Kimlik No -TC Identity Number:44068314976

e-Arşiv Fatura
(e-Archive Invoice)

Sipariş No -Order No	407-3236279-5889155
Özelleştirme No -Customization Id	TR1.2
Fatura Senaryo -Invoice Scenario	EARSIVFATURA
Fatura Tipi -Invoice Type	SATIS
Fatura No -Invoice No	3002024016672256
Fatura Tarihi -Invoice Date	04 / 06 / 2024
Düzenleme Zamanı -Issue Time	16:26:04

ETTN: 877e0ac1-1195-38ae-82a9-4757fe9b10a5
*Bu tutarlara K.D.V dahil değildir.(These amounts are exclusive of VAT.)

No	MKT (QTY)	Ürün Kodu (Product ID)	Mal/Hizmet Cinsi (Type of Goods/Services)	Birim Fiyat (Unit Price)	Birim Fiyat* (Unit Price)	Genel Tutar* (Gross Amount)	İskonto* (Discount)	KDV Oranı (VAT Rate)	KDV Tutarı (VAT Amount)	Toplam Tutar (Total Amount)	MKT. İade (QTY. Returned)	İade Tutarı (Return Amount)
1	1	B09SKZV1WZ	HP Smart Tank 516 Wireless All-in-One (3YW70A) IMEI:3YW70A	5,700.00TL	4,750.00TL	4,750.00TL	83.33TL	20%	933.33TL	5,600.00TL		
Toplam Tutar -Total Amount										5,600.00TL		

amazon.com.tr

Amazon Turkey Perakende Hizmetleri Limited Şirketi
Esentepe MahallesiBahar Sk. No: 13/52
Şişli-İstanbul,34394,Turkey
Web Sitesi -Web site:http://www.amazon.com.tr/
Vergi Dairesi -Tax Office:Zincirlikuyu
Vergi Kimlik No -Tax Number:0680972288
Ticaret Sicil No -Trade Registration Number:91626-5
Mersis No:68097228800001



Teslimat Adresi -Delivery Address:
Bartın Üniversitesi, Fen fakültesi giriş kat ofis no z33
Bartın,Kutlubey(Tuzcularköyü),74110
TR

Fatura Adresi -Invoice Address:
emir koç
Kemalpaşa Mahallesi Ali Koç Sokak Serdivan Sakarya, 8 daire 2
Sakarya,Sakarya,54050
TR
TC Kimlik No -TC Identity Number:44068314976

e-Arşiv Fatura
(e-Archive Invoice)

Sipariş No -Order No	407-3236279-5889155
Özelleştirme No -Customization Id	TR1.2
Fatura Senaryo -Invoice Scenario	EARSIVFATURA
Fatura Tipi -Invoice Type	SATIS
Fatura No -Invoice No	3002024016672256
Fatura Tarihi -Invoice Date	04 / 06 / 2024
Düzenleme Zamanı -Issue Time	16:26:04

ETTN: 877e0ac1-1195-38ae-82a9-4757fe9b10a5

Ekranın bu kısmında görmüş olduğunuz ekran görüntüsü ise talep edilen ücretin malzeme ve ihtiyaç alımında kullanılmayan ve artan kısmının iade dekontundandır.



Bilgi Dekontu

FAST GÖNDERİMİ

MÜŞTERİ NO : 74662559	İŞLEM REF : 249844814440
VKN/TCKN/YKN : *****	SIRA NO/ID : - / 2498427102401
IBAN NO : TR63 0006 7010 0000 0058 3994 63	PERSONEL SİCİL : INT001
İŞLEM TARİHİ : 28.05.2025 14:42:44	BELGE NUMARASI : SYE2025300347467
VALÖR : 28/05/2025	ETTN : -

GİDEN FAST TUTARI :-3120	VERGİ : -
DÖVİZ CİNSİ : TL	KOMİSYON : -
GÖNDEREN HESAP NO : 58399463/IBAN:TR63 0006 7010 0000 0058 3994 63	TOPLAM TAHSİLAT TUTARI :-3120
GÖNDEREN ADI : EMİR KOÇ	ÖDEMENİN KAYNAĞI : İnternet Bankacılığı
ALICI BANKA : Türkiye Vakıflar Bankası T.A.O.	MESAJ TÜRÜ : Bireysel Ödeme
ALICI ŞUBE : -	SORGU NO : 1817633141
ALICI HESAP : TR490001500158007290109235	KOLAY ADRES : -
ALICI ADI : Tübitak Strateji Geliştirme Daire Başkanlığı	
ALICI TCKN/VD/VKN : -	

AÇIKLAMA

YUKARIDAKİ TUTAR HESABINIZA BORÇ/ALACAK KAYDEDİLMİŞTİR.

AÇIKLAMA:FAST/2209-A 2023 2. Dönem 1919B012322831 Başvuru numaralı projenin artan bütçe tutarı, Emir Koç olarak tarafınıza iletıyorum.

EMİR KOÇ

e-Dekont yerine geçmez bilgi amaçlıdır.
e-Dekontunuzu şubeden veya internet/mobil şubeden temin edebilirsiniz.

Ticari Unvan: Yapı ve Kredi Bankası A.Ş.
Büyük Mükellefler VD: 9370020892
Web Adresi: www.yapikredi.com.tr
Ticaret Sicil Numarası: 32736
İşletmenin Merkezi: Yapı Kredi Plaza D Blok 34330
Levent - İstanbul
Mersis No: 0937002089200741
SİSTEM



Bilgi Dekontu

FAST GÖNDERİMİ

MÜŞTERİ NO : 74662559	İŞLEM REF : 249844814440
VKN/TCKN/YKN : *****	SIRA NO/ID : - / 2498427102401
IBAN NO : TR63 0006 7010 0000 0058 3994 63	PERSONEL SİCİL : INT001
İŞLEM TARİHİ : 28.05.2025 14:42:44	BELGE NUMARASI : SYE2025300347467
VALÖR : 28/05/2025	ETTN : -

GİDEN FAST TUTARI :-3120	VERGİ : -
DÖVİZ CİNSİ : TL	KOMİSYON : -
GÖNDEREN HESAP NO : 58399463/IBAN:TR63 0006 7010 0000 0058 3994 63	TOPLAM TAHSİLAT TUTARI :-3120
GÖNDEREN ADI : EMİR KOÇ	ÖDEMENİN KAYNAĞI : İnternet Bankacılığı
ALICI BANKA : Türkiye Vakıflar Bankası T.A.O.	MESAJ TÜRÜ : Bireysel Ödeme
ALICI ŞUBE : -	SORGU NO : 1817633141
ALICI HESAP : TR490001500158007290109235	KOLAY ADRES : -
ALICI ADI : Tübitak Strateji Geliştirme Daire Başkanlığı	
ALICI TCKN/VD/VKN : -	

AÇIKLAMA



YUKARIDAKİ TUTAR HESABINIZA BORÇ/ALACAK KAYDEDİLMİŞTİR.

AÇIKLAMA:FAST/2209-A 2023 2. Dönem 1919B012322831 Başvuru numaralı projenin artan bütçe tutarı, Emir Koç olarak tarafınıza iletıyorum.

EMİR KOÇ

e-Dekont yerine geçmez bilgi amaçlıdır.
e-Dekontunuzu şubeden veya internet/mobil şubeden temin edebilirsiniz.

Ticari Unvan: Yapı ve Kredi Bankası A.Ş.
Büyük Mükellefler VD: 9370020892
Web Adresi: www.yapikredi.com.tr
Ticaret Sicil Numarası: 32736
İşletmenin Merkezi: Yapı Kredi Plaza D Blok 34330
Levent - İstanbul
Mersis No: 0937002089200741
SİSTEM

PROJE YÜRÜTÜCÜSÜNÜN ADI – SOYADI - İMZA	DANIŞMANIN ADI – SOYADI - İMZA
Emir Koç	Ahmet Berk Üstün
	

Tarih :16.06.2025