ANKARA ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BLM 4537 PROJE RAPORU

Security for Everyone Blog-App

Ömer Faruk Eren

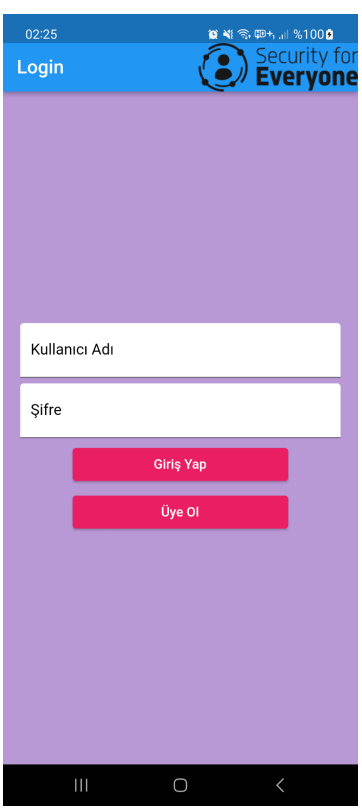
19290242

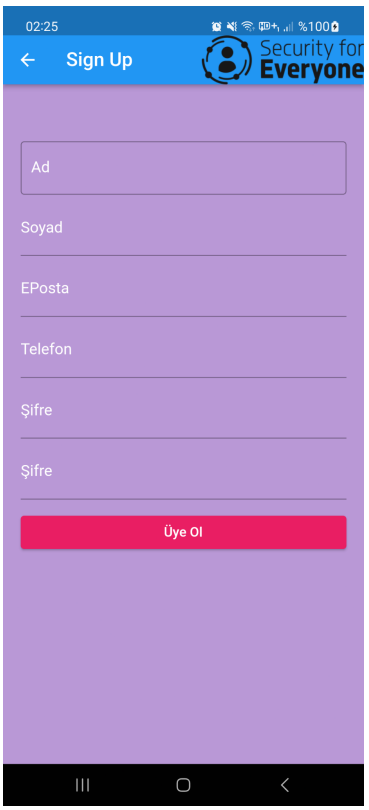
Enver Bağcı

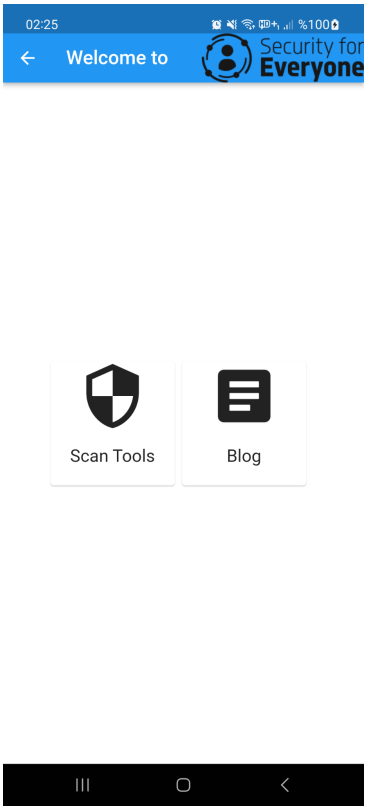
“Security for Everyone” bloglarının yayınlanması ve sunulan scan türlerinin tanıtılması için geliştirdiğim taslak uygulamadır. Firebase ile üye olunup giriş yapılabilmektedir. Scan ve Blog sayfaları, verilen json dosyalarından çekilip dosya devam ettiği sürece akışına devam ederek ilerlemektedir. Ben 6 blogluk bir json ve 100 scanlik bir json dosyasını siteden indirdim. Siteye atılacak bir fetch sonucu alınan bir json entegre edilirse sitedeki her veriyi alacak şekilde de değiştirilebilir ama bunu yapmak yerine local dosya ile çalıştırdım. Json dosyalarından çekilen html kodunu anlamlandıracak uygun kütüphaneleri ve kodları da kullanarak blog ve scan ekranlarını şekillendirdim. Bu sayede daha gerçekçi görüntüye ulaştım. Json dosyaları ve blogların resimleri images klasöründedir. Uygulama 5 sayfadan oluşmaktadır. Login, Sign in sayfaları dışında admin, scan ve blog sayfaları vardır. Kullanıcı admin sayfasından blog veya scanleri açmayı tercih edebilir.

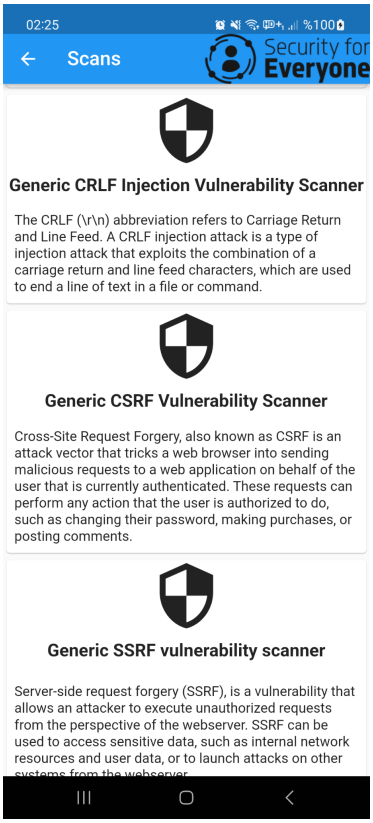
Ekranlar ve onlara ait kodlardan kısımlar:

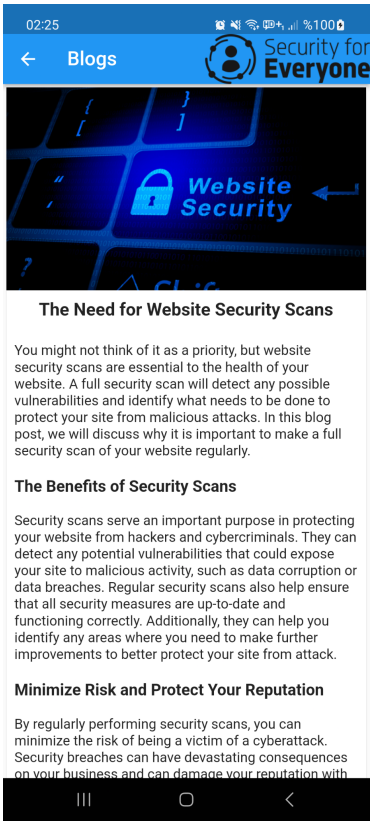
Tüm Ekranlar:



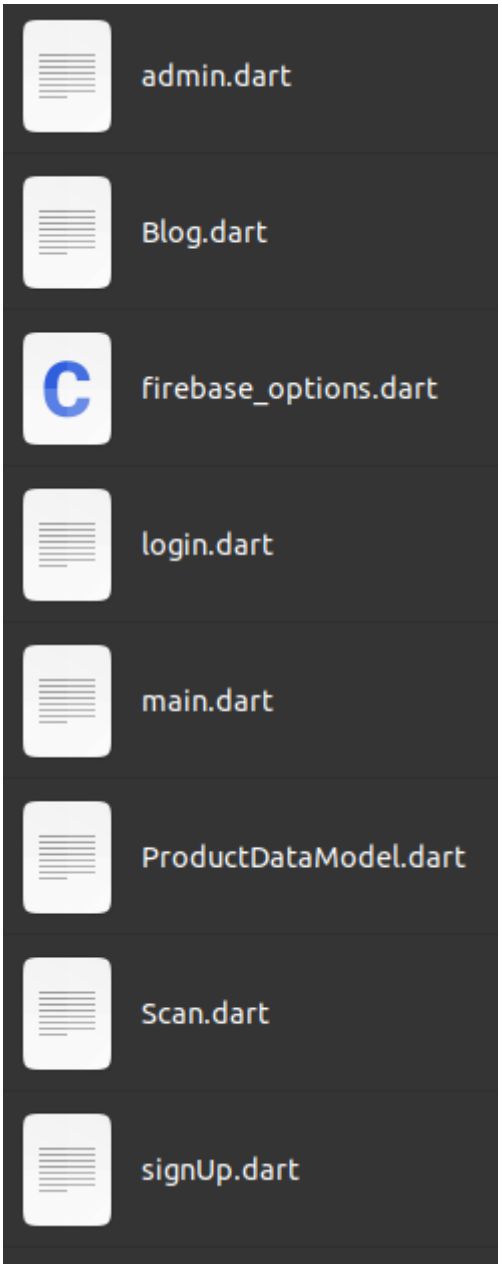








Tüm dosyaların listesi:



Model dosyası:

```
class ProductDataModel{    The file name 'ProductDataMo

  String? mini_desc;    The variable name 'mini_desc' i
  String? meta_title;    The variable name 'meta_title'
  String? name;
  String? article;
  String? img_banner;    The variable name 'img_banner'

  ProductDataModel(
      {

      this.mini_desc,    The variable name 'mini_desc'
      this.meta_title,    The variable name 'meta_title
      this.name,
      this.article,
      this.img_banner,    The variable name 'img_banner

      });

  ProductDataModel.fromJson(Map<String,dynamic> json)
  {

    mini_desc =json['mini_desc'];
    meta_title = json['meta_title'];
    name = json['name'];
    article = json['article'];
    img_banner = json['img_banner'];

  }
}
```

Json çekme:

```
Future<List<ProductDataModel>> ReadJsonData() async {    The variable name 'Re
  final jsondata = await rootBundle.rootBundle.loadString('images/scan.json');
  final list = json.decode(jsondata) as List<dynamic>;
  return list.map((e) => ProductDataModel.fromJson(e)).toList();
}
}
```

Logo içeren appBar

```
      appBar: AppBar(
    title: Text("Scans "),    Use 'const' with
    actions: <Widget>[
      Row(
        children: <Widget>[
          Image.asset("images/banner.png"),
        ], // <Widget>[]
      ), // Row
    ], // <Widget>[]
  ), // AppBar
```

sign up page kod parçacığı

```
import 'dart:math';    Unused import: 'dart:math'. Try removing the import dir

import 'package:flutter/src/widgets/container.dart';    Unused import: 'packag
import 'package:flutter/src/widgets/framework.dart';    The import of 'package

import 'package:flutter/material.dart';
import 'package:flutter_application_1/login.dart';
import 'package:cloud_firestore/cloud_firestore.dart';

class SignUp extends StatefulWidget {
  const SignUp({super.key});

  @override
  State<SignUp> createState() => _SignUpState();
}

class _SignUpState extends State<SignUp> {
  final _sigInFormKey = GlobalKey<FormState>();    The value of the field '_si
  final _firestore = FirebaseFirestore.instance;
  TextEditingController ad = TextEditingController();
  TextEditingController soyad = TextEditingController();
  TextEditingController eposta = TextEditingController();
  TextEditingController telefon = TextEditingController();
  TextEditingController sifre1 = TextEditingController();
  TextEditingController sifre2 = TextEditingController();

  @override
  Widget build(BuildContext context) {
    CollectionReference makeupRef = _firestore.collection('makeup');
    return Scaffold(
     appBar: AppBar(
          title: Text("Sign Up "),    Use 'const' with the constructor to impr
          actions: <Widget>[
            Row(
              children: <Widget>[
                Image.asset("images/banner.png"),
              ], // <Widget>[]
            ), // Row
          ], // <Widget>[]
        ), // AppBar
      backgroundColor: Color.fromARGB(221, 214, 175, 247),    Use 'const' wi
      body: Form(
        child: Padding(
          padding: EdgeInsets.all(20),    Use 'const' with the constructor to
```

Json okuma örneği

```
children: <Widget>[
  Image.asset('images/${(items[index].img_banner.toString())}'),
```

Blog sayfasının kodlarının tamamı:

```
import 'dart:convert';    The file name 'Scan.dart' isn't a snake_case identifi
import 'package:flutter/material.dart';
import 'package:flutter_application_1/ProductDataModel.dart';
import 'package:flutter/services.dart' as rootBundle;    The prefix 'rootBundle
import 'package:flutter_html/flutter_html.dart';

class ScanPage extends StatefulWidget {
  const ScanPage({super.key});

  @override
  State<ScanPage> createState() => _ScanPageState();
}

class _ScanPageState extends State<ScanPage> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
                appBar: AppBar(
          title: Text("Scans "),    Use 'const' with the constructor to improve
          actions: <Widget>[
            Row(
              children: <Widget>[
                Image.asset("images/banner.png"),
              ], // <Widget>[]
            ), // Row
          ], // <Widget>[]
        ), // AppBar
        body: FutureBuilder(
      future: ReadJsonData(),
      builder: (context, data) {
        if (data.hasError) {
          return Center(child: Text("${data.error}"));
        } else if (data.hasData) {
          var items = data.data as List<ProductDataModel>;
          return ListView.builder(
              itemCount: items == null ? 0 : items.length,    The operand can't
              itemBuilder: (context, index) {
                return Container(    Use a 'SizedBox' to add whitespace to a lay
                  width: MediaQuery.of(context).size.width,
                  child: Card(
                    child: Column(
                      children: <Widget>[
                        Icon(
                            Icons.security,
```

```
                                size: 80,
                              ), // Icon    Use 'const' with the constructor to im
                            SizedBox(
                              height: 10,
                            ), // SizedBox    Use 'const' with the constructor to im
                            Text(items[index].name.toString(),
                                style: TextStyle(
                                    fontWeight: FontWeight.bold, fontSize: 18)), //
                            SizedBox(
                              height: 10,
                            ), // SizedBox    Use 'const' with the constructor to im
                            Html(data: items[index].mini_desc),
                          ], // <Widget>[]
                        ), // Column
                      ), // Card
                    ); // Container
                  }); // ListView.builder
            } else {
              return Center(
                child: CircularProgressIndicator(),    Use 'const' with the construct
              ); // Center    Use 'const' with the constructor to improve performanc
            }
          },
        )); // FutureBuilder // Scaffold
      }

      Future<List<ProductDataModel>> ReadJsonData() async {    The variable name 'Rea
        final jsondata = await rootBundle.rootBundle.loadString('images/scan.json');
        final list = json.decode(jsondata) as List<dynamic>;
        return list.map((e) => ProductDataModel.fromJson(e)).toList();
      }
    }
```