

ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



BLM4061 PROJE RAPORU

Kızılay Kan Bağışı Mobil Uygulaması

Tuğba Oğuz

20290278

Şahin Emrah

Ocak 2024

ÖZET

Bitirme projesi; basit bir arayüz sahip, herkesin kolaylıkla kullanabileceği ve farkındalık oluşturmaya yararı olacak bir kan bağış mobil uygulamasıdır. Bu uygulamanın amacı, kan bağışının toplumda ne kadar önemli olduğunu vurgulayarak insanları bu hayati eyleme daha fazla katılmaya teşvik etmektir. Ayrıca insanların kan verme aralıklarının düzenli bir şekilde takibinin sağlanmasında kolaylık sağlar.

Uygulamada, bilgilendirme sayfaları kullanıcılara kan bağış süreci hakkında detaylı bilgi sunarak, yanlış anlaşılmalara önlemek ve sürece ilişkin bilinci artırmaktadır.

Projemde, hem öğrenme aşaması kolay olması açısından hem de performans boyutunda iyi olmasından dolayı flutter framework'ünü tercih ettim. Flutter ve firebase bağlantısı yaparak kimlik doğrulama işlemleri ve kullanıcı bilgilerinin depolama işlemlerini gerçekleştirmesini sağladım. Bu da daha derli toplu bir veritabanı yönetimi imkanı sunmuş oldu.

Uygulamanın mobil platformda olması, kullanıcıların her yerden ve her zaman erişim sağlamasına imkan verir, bu da katılımı ve etkinliği artırır. Uygulamam IOS, ANDROID, MACOS mobil işletim sistemlerini desteklemektedir.

Geri bildirim sistemi sayesinde, uygulamanın sürekli olarak iyileştirilmesine ve kullanıcı ihtiyaçlarına daha iyi hizmet etmesine olanak tanır.

İÇİNDEKİLER

ÖZET	ii
İÇİNDEKİLER	iii
ŞEKİLLER DİZİNİ.....	iii
1. GİRİŞ	1
1.1. Kan Bağışı Neden Önemlidir?	1
1.2. Kan Bağışının Faydaları Nelerdir?	1
1.1. Kan Vermeden Önce Yapılan İşlemler Nelerdir?	2
2. KULLANILAN TEKNOLOJİLER VE MİMARİ	3
2.1. Flutter Nedir?	3
2.1.1. Flutter ve React Native Arasındaki Farklar.....	3
2.2. Firebase Nedir?.....	4
2.1.1. Firebase Authentication Nedir?.....	5
2.1.1. Firebase Storage Nedir?.....	5
2.1.1. Firestore Database Nedir?.....	6
2.3. Figma Nedir?.....	7
3. PROJE YAPISI	8
3.1. Kan Bağışı Uygulamasının Kullanım Senaryosu	8
3.2. Visual Studio Code Kurulumu	12
3.3. Flutter Projesi Oluşturma.....	14
4. VERİTABANI BAĞLANTISI.....	15
4.1. Firebase Projesi Oluşturma	15
4.2. Firebase Flutter Bağlantısı	18
5. SONUÇ	21
KAYNAKÇA	22

ŞEKİLLER DİZİNİ

Şekil 1.1. Kan bağışına ait bir gösterim	2
Şekil 2.1. Firebase yapısı	4
Şekil 2.2. Firebase authentication seçenekleri.....	5
Şekil 2.3. Firebase storage logosu	6
Şekil 2.4. Cloud firestore yapısı	6
Şekil 2.5. Uygulamamın figma tasarımı	7
Şekil 3.1. Kayıt ve giriş sayfası	8
Şekil 3.2. Ön bilgi sayfası ve anasayfa	9
Şekil 3.3. Kullanıcı profili ve randevu sistemi sayfası	10
Şekil 3.4. Harita ve geri bildirim sayfaları	11
Şekil 3.5. Visual studio code indirme sayfası	12
Şekil 3.6. Visual studio code kurulumu.....	12
Şekil 3.7. Flutter eklentisi indirme.....	13
Şekil 3.8. Dart eklentisi indirme	13
Şekil 3.9. Flutter indirme sayfası	14
Şekil 3.10.Flutter dosya dizininin ortam değişkenlerine eklenmesi	14
Şekil 3.11.Flutter projesi oluşturma ve VS Code üzerinde açılması	15
Şekil 4.1. Firebase’de proje oluşturma	16
Şekil 4.2. Firebase özellikleri ekleme.....	16
Şekil 4.3. Firestore database görünümü	17
Şekil 4.4. Firebase proje ayarları	17
Şekil 4.5. Firebase uygulama yapılandırması	18
Şekil 4.6. Firebase paketlerinin VS Code’a eklenmesi	18
Şekil 4.7. Flutter firebase bağlantısı komutları	19
Şekil 4.8. Projenin genel görüntüsü	19
Şekil 4.8. Projenin genel görüntüsü	19
Şekil 4.9. Services ve user_auth dosyaları	20
Şekil 4.10.Diyalog dosyaları	20
Şekil 4.11.Diyalog ile ilgili bir görünüm	21

1.GİRİŞ

Kan, hayat kurtaran bir kaynaktır ve düzenli kan bağıışı, hastanelerde yeterli kan stoğunun sürekli olarak bulunmasını sağlar. Bu uygulama, kan bağıışının önemine dair toplumsal farkındalığı artırarak daha fazla insanın kan bağıışında bulunmasını teşvik edebilir. Basit ve sezgisel arayüz, kullanıcıların uygulamayı kolayca kullanmalarını sağlar, böylece daha geniş bir kitleye ulaşılabilir.

1.1 Kan Bağıışı Neden Önemlidir?

Kan bağıışı öncelikle gerekli kan bulunamadığından yaşamını kaybeden kişiler açısından çok önemlidir. Tüm dünyada hastalıklar ya da kazalar sonucunda, kan bulunamadığı için çok sayıda insan yaşamını kaybetmektedir. Toplumda yapılan kan bağıışı nüfusun sadece % 1 kadarını oluşturur. Bu oldukça düşük bir orandır. Kan bağıışı hem kan ihtiyacı olan kişiler açısından, hem de bağıışı yapan kişi açısından sağlıklı bir yapıya sahip olmak açısından önemlidir.

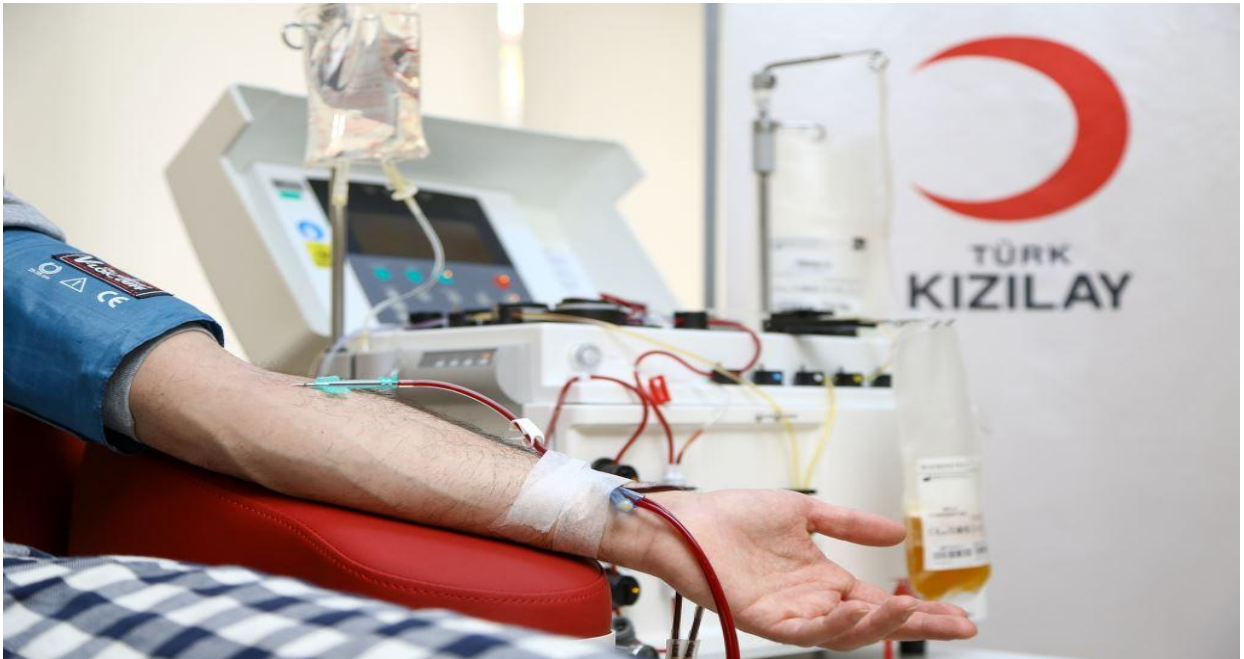
1.2 Kan Bağıışının Faydaları Nelerdir?

Yapılan araştırmalar düzenli kan bağıışının hem duygusal hem de fiziksel anlamda insan sağlığı üstündeki çok ciddi faydaları üzerinde durur.

Kişinin ruhsal durumundan kalp sağlığına kadar birçok yarar sunan kan bağıışı, hücreleri yenileyerek daha güçlü vücut yapısının oluşmasını destekler. Kalp krizi riskini büyük oranda azalttığı bilinen kan bağıışı, daha sağlıklı karaciğere sahip olmanın yanı sıra baş ağrısı ve yorgunluğa iyi gelmesiyle de bilinir. İnsanı psikolojik olarak rahatlatan kan verme işlemi stresi azaltmasıyla da dikkat çeker. Ayrıca kandaki yüksek yağ oranını da azaltır.

1.3 Kan Vermeden Önce Yapılan İşlemler Nelerdir?

1. Öncelikle donöre form doldurturularak bazı sorulara cevap vermesi istenir.
2. Kan vermeye engel bir durum olup olmadığının kontrolü için fiziki muayene yapılır.
3. Bir sorun çıkmazsa tüpe kan alınır ve gerekli kan sayımları yapılır.
4. Kan testlerinde de problem tespit edilmezse bağışçı donör yatağına alınır ve bir ünite kan alma işlemi gerçekleştirilir. Bu işlem 15 dakikadan daha az sürmektedir.
5. Kan verme işleminin ardından bağışcını 10-15 dakika dinlenmesi istenir.



Görsel 1.1 Kan bağışına ait bir gösterim

2. KULLANILAN TEKNOLOJİLER VE MİMARİ

Projemi geliştirirken kullandığım platformlar,yazılım dilleri ve araçlar hakkındaki bilgilere bu bölümde yer vereceğim.

2.1 Flutter Nedir?

Uygulamamı geliştirirken flutter framework¹'ünü kullandım. Flutter, uygulama geliştirmek isteyenler için Google tarafından sunulan SDK²'dir ve mobil UI framework'üdür. Bu sayede özellikle mobil için ihtiyaç duyulan hemen her uygulama geliştirilebilir.

Platformda mobil uygulamaların yanında web uygulamaları hazırlamak da mümkündür. Cross yani çapraz platform olması sayesinde de hem iOS hem de Android tabanlı uygulamalar geliştirilebilir. Çapraz platformlu mobil uygulamalar sayesinde hem maliyet hem de zaman açısından tasarruf sağlanır. Aynı zamanda Flutter'ın Hot Reload³ özelliği sayesinde kodlama yaparken yapılan değişiklikler anında görülür.

2.1.1 Flutter- React Native Arasındaki Farklar

Flutter kadar yaygın kullanılan bir diğer uygulama geliştirme framework'ü ise React Native'dir. Facebook tarafından geliştirilen açık kaynaklı bir mobil uygulama geliştirme çerçevesidir(framework). Flutter ile birçok ortak özellikleri bulunmakla beraber farklılıkları da mevcuttur.

- Flutter, Google tarafından geliştirilen Dart dili üzerine geliştirilmiştir. Nesneye yönelik programlama konusunda herhangi bir deneyimi olan geliştiricinin, Dart'ı öğrenmesi oldukça kolay olacaktır.
- React Native ise temel dil yapısı olarak JavaScript'i kullanır. JavaScript kolay öğrenme eğrisi ve yaygın kullanımı nedeniyle oldukça popülerdir.
- React Native'e göre Flutter biraz daha iyi performans sağlar. Performansı yavaşlatan herhangi bir köprüsü olmaması, doğrudan kod derleme yapılabilmesi özellikleriyle Flutter uygulamalarının hızı yüksektir.

¹ Yazılım geliştiricilere belirli bir dil veya teknoloji için yapılandırılmış bir temel ve araç seti sağlayan bir yazılım çerçevesidir.

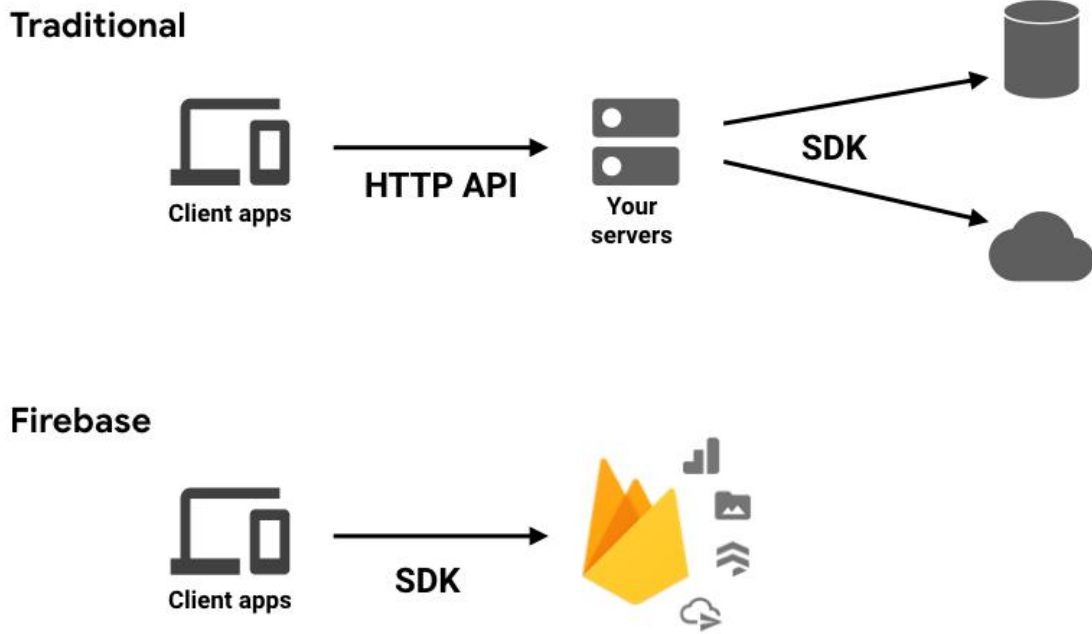
² Yazılım geliştiricilerin belirli bir platform veya teknoloji için uygulamalar oluşturabilmeleri için gerekli araçlar, kütüphaneler ve belgeleri içeren bir yazılım paketidir.

³ Geliştiricilerin uygulamalarındaki değişiklikleri hızlı bir şekilde görmelerini sağlayan bir özelliktir.

Geliştirilen projenin ihtiyaçlarına göre flutter ya da react native arasında seçim yapmak geliştiriciye aittir.

2.2 Firestore Nedir?

Firestore, Google tarafından sağlanan bir BaaS (Backend as a Service) hizmetidir. Firestore, uygulama geliştiricilerine sunucu tarafı altyapısını yönetme zorunluluğundan kurtararak, uygulamalarının hızlı bir şekilde geliştirilmesine ve dağıtılmasına olanak tanır.

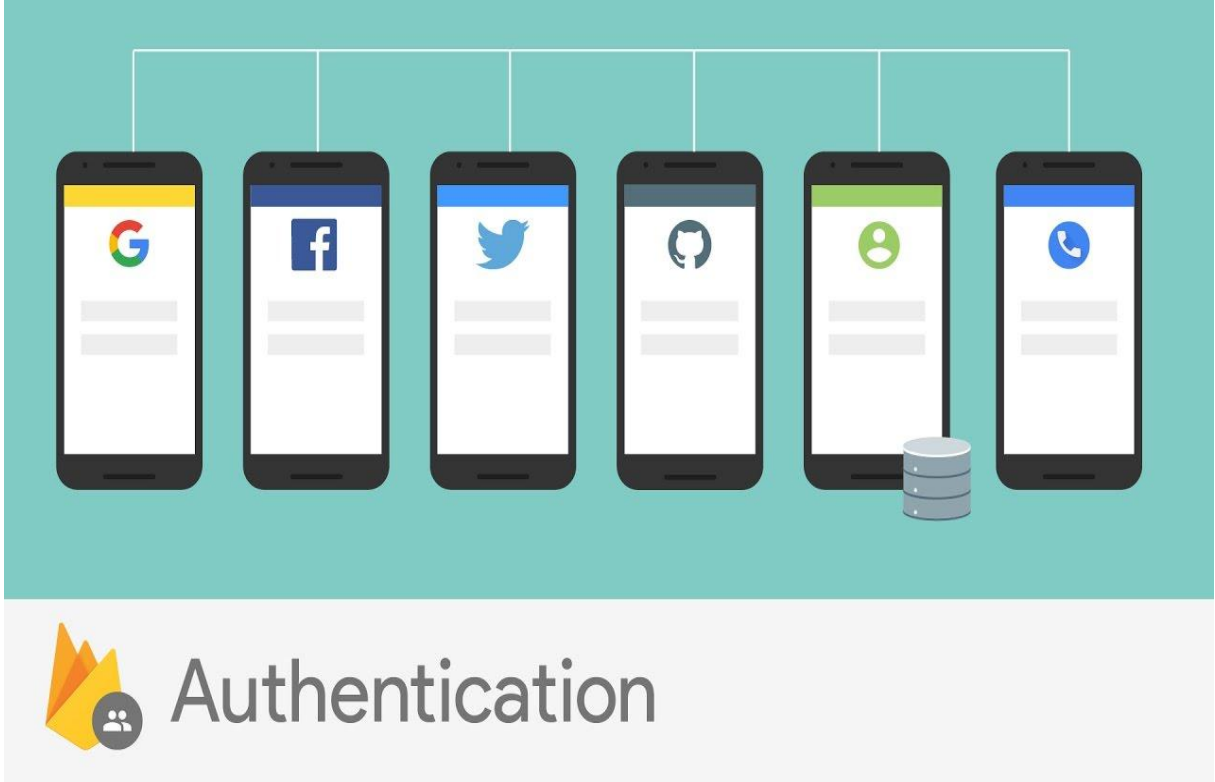


Görsel 2.1 Firestore yapısı

Kullanıcı kimlik doğrulama, veritabanı yönetimi, dosya depolama, analitik, mesajlaşma ve hizmet yönetimi gibi birçok özellik sunar. Ayrıca, Android, iOS ve web uygulamaları için SDK'lar sağlar. Bu sayede, uygulama geliştiricileri Firestore'ı kullanarak uygulamalarının backend'ini kolayca yönetebilirler.

2.2.1 Firebase Authentication Nedir?

Firebase Authentication(kimlik doğrulama), sağladığı SDK'larıyla uygulamanız da basit bir şekilde kullanıcı kimliğini doğrulamada yardımcı olur. Google, Facebook, Twitter, Github, Email/Password ve Anonim olarak kimlik doğrulamayı destekler.



Görsel 2.2 Firebase authentication seçenekleri

Uygulamamda kullanıcı giriş ve kayıt yönetimi için firebase kullanıcı kimlik doğrulama(authentication) özelliğini kullandım. Bu özellik sayesinde kullanıcının, kayıt altında olan bilgileri saklanılabilecek. Ayrıca kaydı olmayan kullanıcılar için denetleme sağlanıp yönlendirmeler yapılabilecek. Projemde giriş yaparken kullanılan kimlik doğrulama seçeneği olarak email-password authentication kullanılmaktadır.

2.2.2 Firebase Storage Nedir?

Firebase, kullanıcıların sürekli olarak ürettiği verileri depolama imkanı veren Storage hizmeti de sunar. Resim, ses, video depolanabilir.

Projemde görsel ve video gösterimi için firebase storage özelliğini kullandım. Kan verme ile ilgili bilgilendirme videosunu ve bilgilendirme resmini depolayarak saklanabilirliği kolaylaştı.



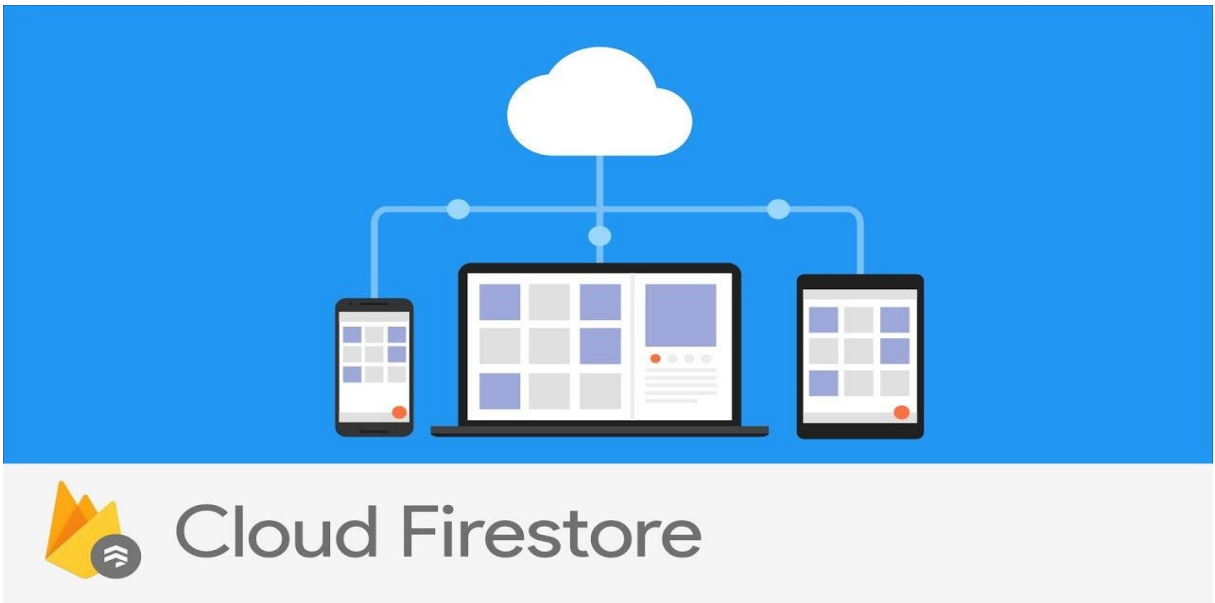
Görsel 2.3 Firebase storage logosu

2.2.3 Firestore Database Nedir?

Cloud Firestore, Google'nin büyük ve karmaşık yapıdaki verileri saklamak ve hızlıca sorgulamak amacıyla geliştirdiği dokuman tabanlı Nosql veritabanıdır.

Sorgulama yapısı olarak daha çok seçenek bulunmaktadır ayrıca verileriniz belgeler ve koleksiyonlardan oluşmaktadır. Belgelere bağlı olarak başka belgeler oluşturulabilir. Böylece iç içe (Nested) verileri saklarken ve sorgularken Cloud Firestore daha performanslı ve esnek bir yapı sağlar.

Projem için cloud firestore'da 'users' adlı bir koleksiyon oluşturarak kullanıcı profilinde yer alan bilgilerin depolanması sağlanmaktadır.



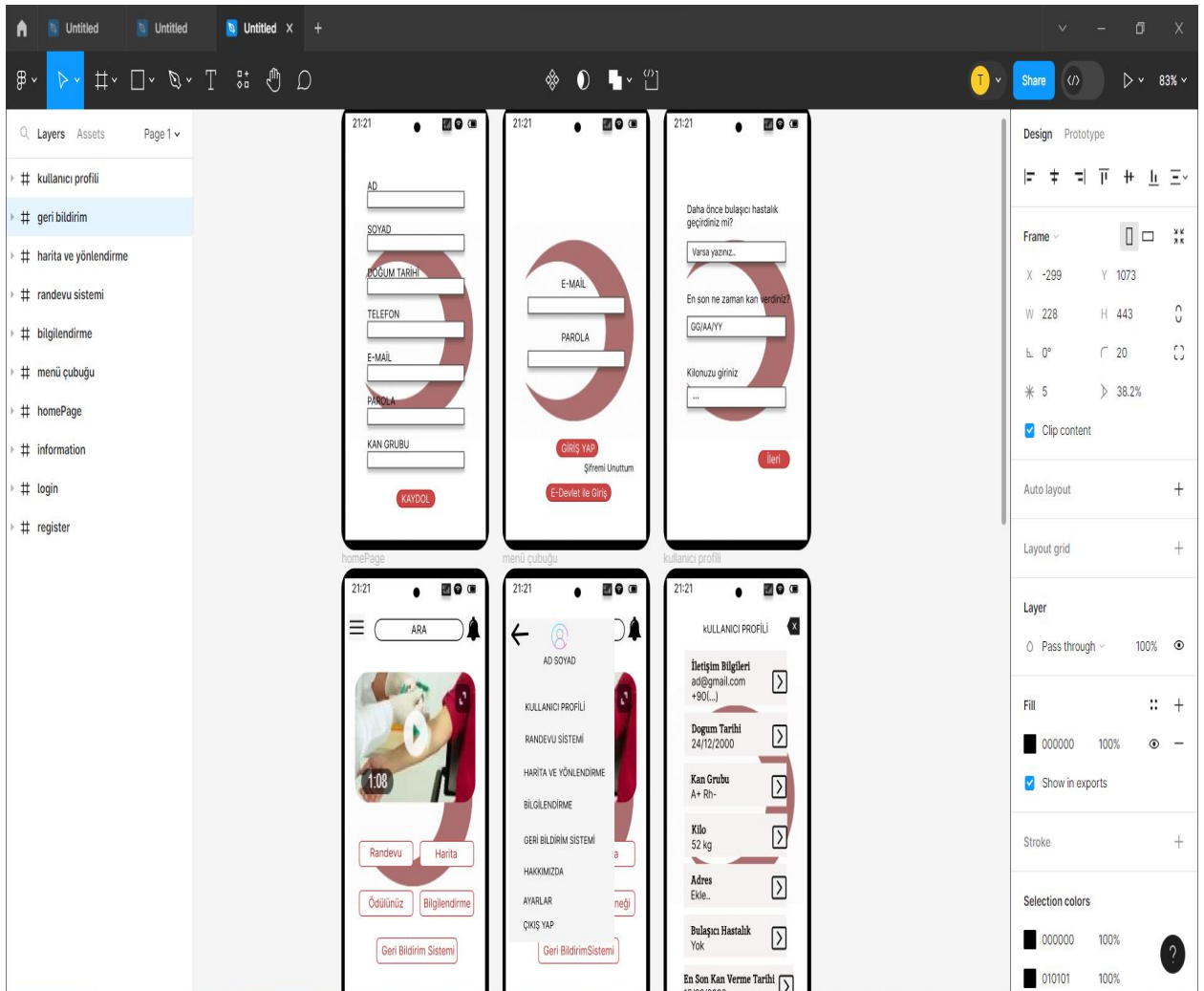
Görsel 2.4 Cloud firestore yapısı

2.3 Figma Nedir?

Figma, insanların bilgisayarları üzerinden internet tarayıcıları aracılığıyla kullanabilecekleri bir tasarım programıdır. Bu program, kullanıcıların web siteleri, mobil uygulamalar ve diğer dijital ürünler için görsel tasarımlar yapmalarına olanak tanır.

Figmanın resmi sitesinden ya da uygulaması üzerinden kayıt olup giriş yaparak kolayca tasarım yapmaya başlayabilirsiniz. Ayrıca kullanımını kolaylaştırıcı videolar internette mevcuttur.

Figmada tasarladığım projemin belli bir kısmının arayüzünü aşağıdaki görselde görebilirsiniz.



Görsel 2.5 Uygulamamın figma tasarımı

3.PROJE YAPISI

3.1 Kan Bağışı Uygulamasının Kullanım Senaryosu

Projem, insanları kan vermeye teşvik etmek ve bu süreci kolaylaştırmak için tasarlanan bir mobil uygulamadır. Basit arayüzüyle herkesin kolaylıkla kullanabilmesini amaçladım. Ayrıca ANDROİD ve IOS destekli olması ile de akıllı telefon kullanan herkes uygulamama erişebilecek.

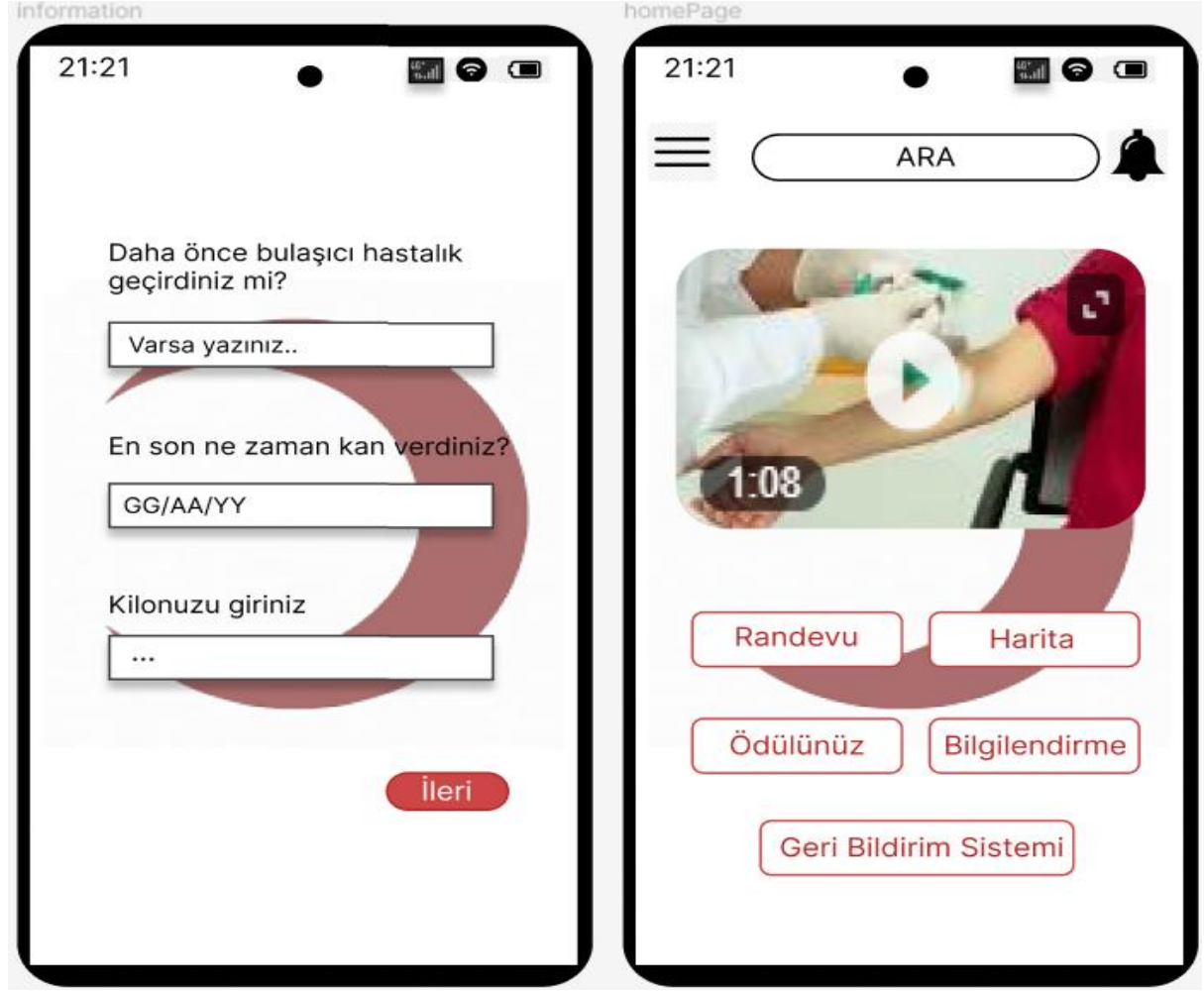
Kayıt ve Giriş Sayfası

The image displays two mobile application screens side-by-side. The left screen, titled 'register', features a white background with a large red crescent moon and star graphic. It contains seven input fields: 'AD', 'SOYAD', 'DOĞUM TARİHİ', 'TELEFON', 'E-MAİL', 'PAROLA', and 'KAN GRUBU'. A red 'KAYDOL' button is at the bottom. The right screen, titled 'login', also has a white background with the same red crescent moon and star graphic. It contains two input fields: 'E-MAİL' and 'PAROLA'. Below these are four buttons: a red 'GİRİŞ YAP' button, a red button with the text 'Şifremi Unuttum', a red button with the text 'E-Devlet ile Giriş', and a red 'KAYDOL' button. Both screens show a status bar at the top with the time '21:21' and various icons.

Görsel 3.1 Uygulamamın kayıt ve giriş sayfası

Uygulama açıldığı zaman ilk olarak kullanıcıyı giriş sayfası karşılayacak. Kaydı olmayan kullanıcılar kayıt sayfasına yönlendirilmektedir. Ayrıca E-devlet ile giriş yapma imkanı da vardır.

Ön Bilgi Sayfası ve Anasayfa



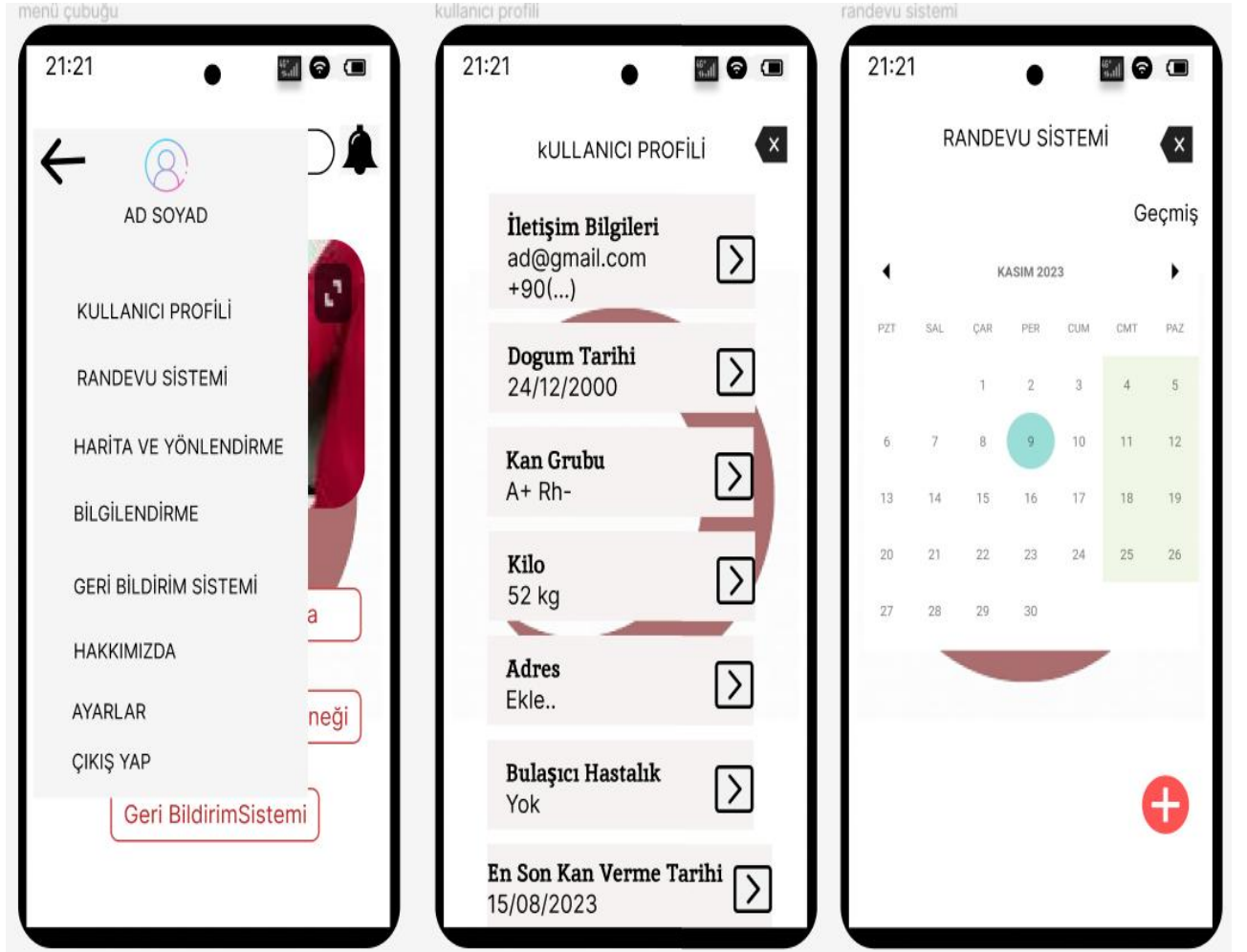
Görsel 3.2 Uygulamamın ön bilgi sayfası ve anasayfa

Kullanıcı giriş yaptıktan sonra kullanıcıyı ön bilgi sayfası karşılamaktadır. Bu sayfada alınan bilgiler cloud firestore'da 'users' adlı koleksiyonda saklanmaktadır. Aynı kullanıcı tekrar giriş yaptığıında bu sayfa görülmemektedir. Bilgileri tekrar güncellemek istendiğinde ise menü çubuğunda erişim sağlanmaktadır. Sonrasında ise anasayfa gelmektedir.

Anasayfada bilgilendirme amaçlı bir video vardır. Ayrıca bildirim sistemi ile donörlere bildirim gitmektedir. Uygulamamın diğer sayfalarına da anasayfadan ulaşmak mümkündür.

Anasayfadan ulaşılan ödül kısmında ise donörlerin ödül seviyesine ulaşınca kazanılan ödüllerin bilgilendirmesi ve hangi seviyede olduğu bulunmaktadır.

Kullanıcı Profili ve Randevu Sistemi

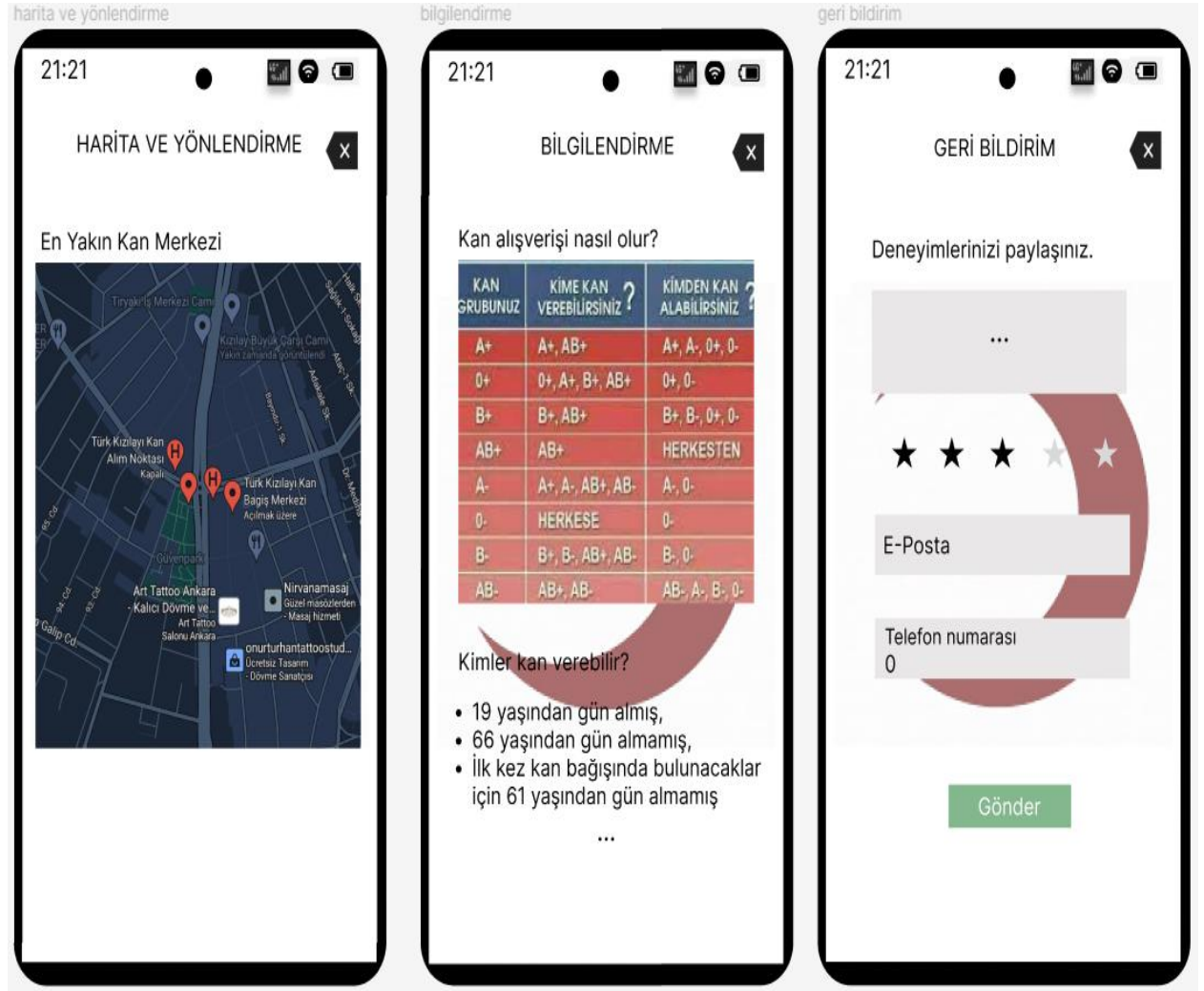


Görsel 3.3 Kullanıcı profili ve randevu sistemi sayfası

Kullanıcı profilinde donörün kişisel bilgileri yer almaktadır. Ayrıca istenildiği zaman tekrar güncelleme imkanı sunulmaktadır. Bu bilgiler cloud firestore'da depolanmaktadır.

Donör, randevu sisteminden uygun olan zamanlar için bağış gününü seçebilmektedir. Menü çubuğundan daha kolay bir şekilde uygulamada yapılmak istenilen işleme ulaşmak mümkündür.

Harita Yönlendirme ve Geri Bildirim Sistemi



Görsel 3.4 Uygulamamın harita ve geri bildirim sistemi

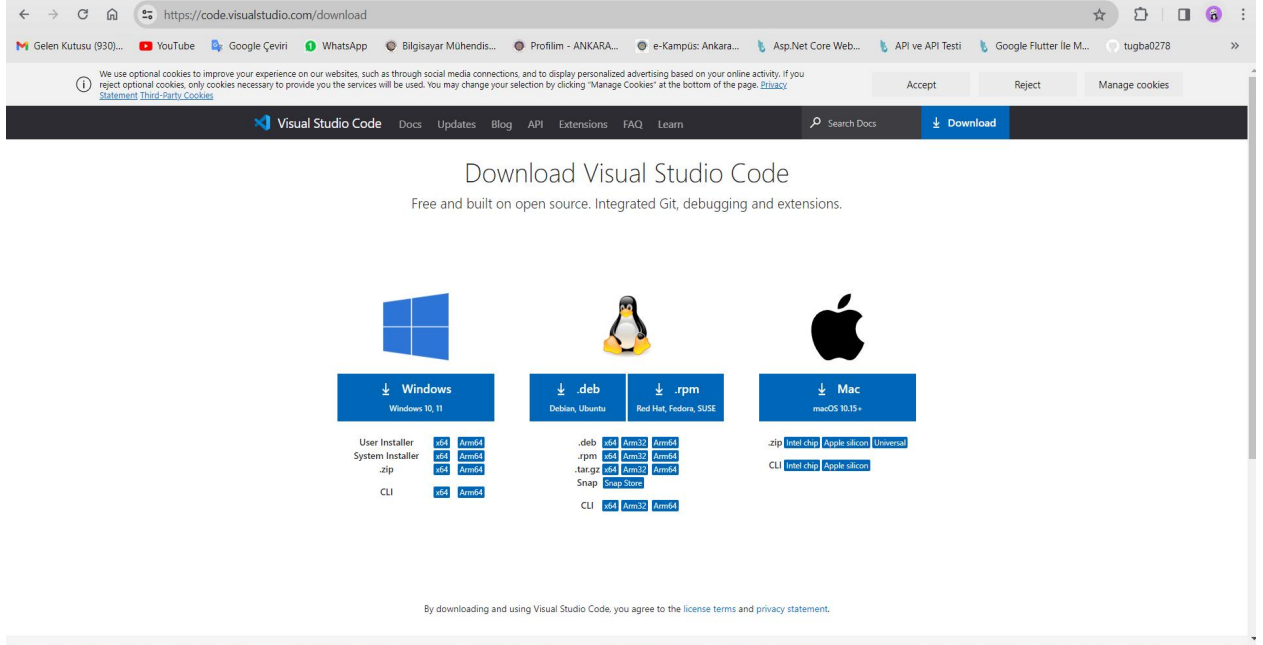
Kullanıcı, konumunu etkinleştirerek harita sisteminde en yakın kan bağış merkezini görüntüleyebilmektedir.

Bilgilendirme sayfasına gidildiğinde kan alışverişinin hangi kan grupları arasında gerçekleştiği ve kimlerin donör olabileceği hakkında bilgiler karşılamaktadır.

En son olarak geri bildirim sisteminde kullanıcıların uygulama ile ilgili deneyimleri, önerileri ve şikayetlerini bildirebilecekleri bir alan sunulmaktadır. Ayrıca uygulamayı puanlayabileceklerdir. Böylece geri dönüşler hem uygulama hakkında bir fikir sunacak hem de geliştirmeye olanak sağlayacaktır.

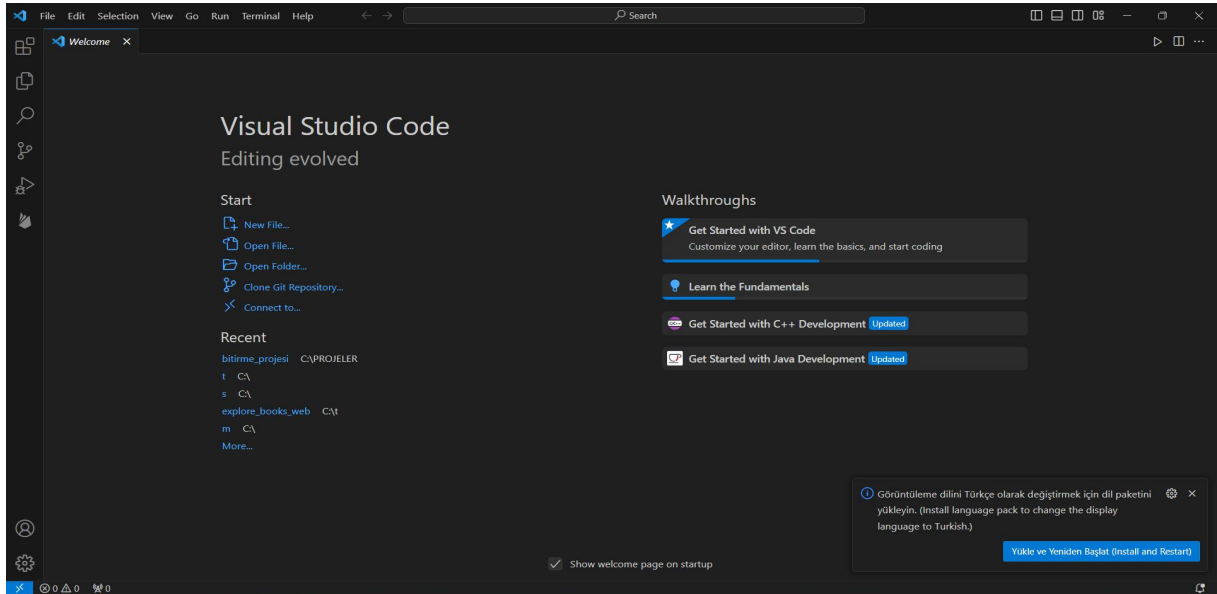
3.2 Visual Studio Code Kurulumu

Projemi geliştirme ortamı olarak Visual Studio Code kullandım. Resmi sitesine girip bilgisayarınızın işletim sistemine uygun editörün indirme işlemini yapabilirsiniz.



Görsel 3.5 Visual Studio Code indirme sayfası

İndirme bittikten sonra kurulumu gerçekleştirince şöyle boş bir proje önünüze gelecek.

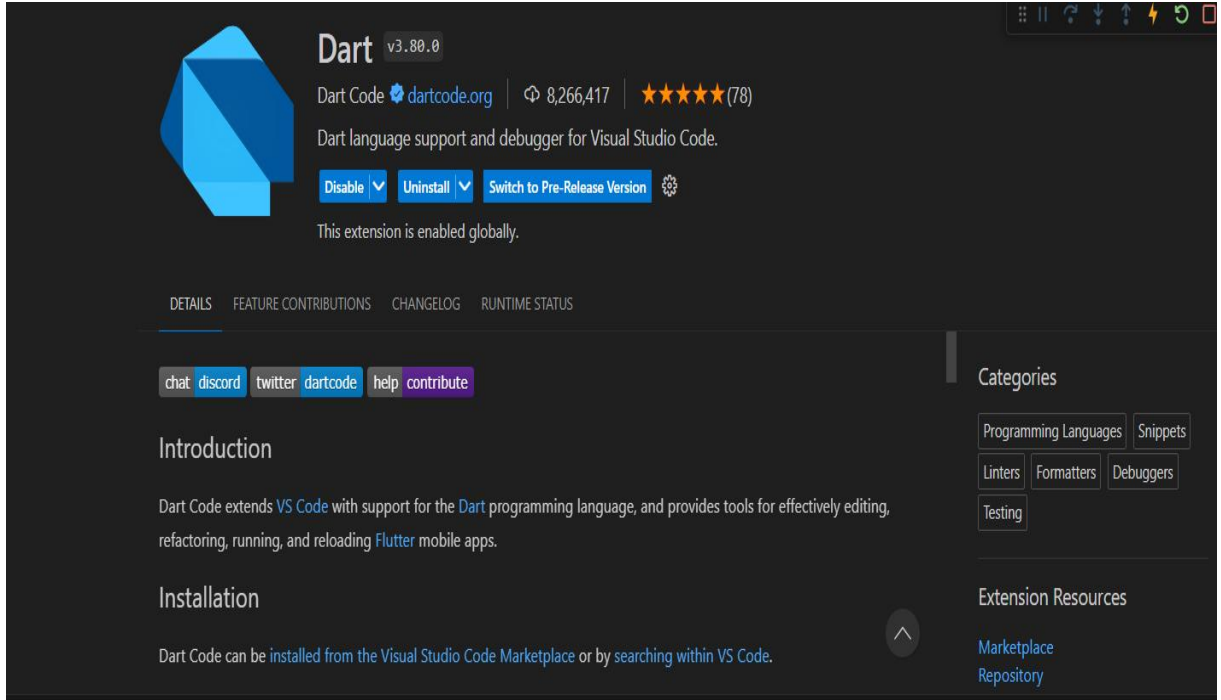


Görsel 3.6 Visual Studio Code kurulumu

Sonrasında ise Flutter çalıştırabilmek için aşağıda gösterilen görseldeki eklentileri kurmamız gerekiyor.



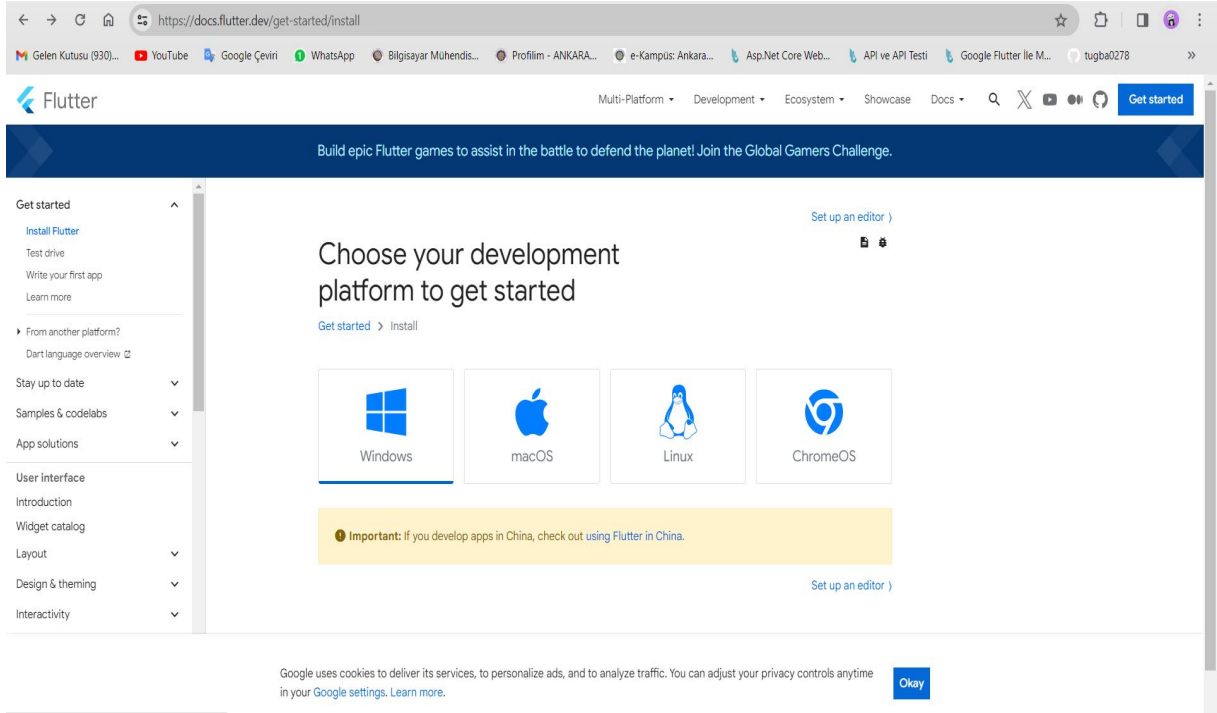
Görsel 3.7 Flutter eklentisini indirme



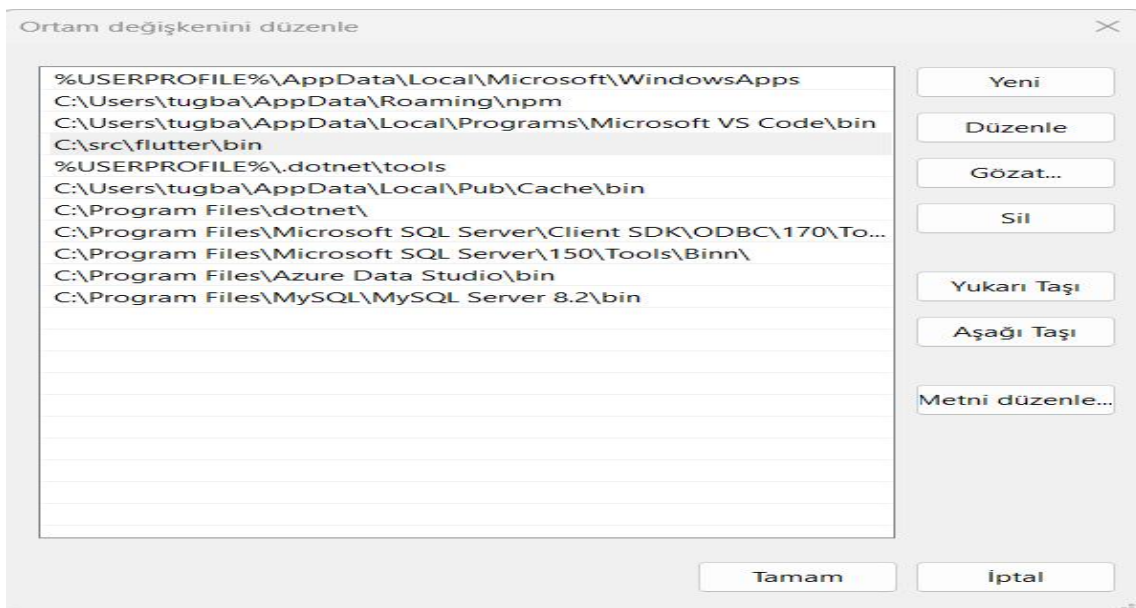
Görsel 3.8 Dart eklentisini indirme

3.3 Flutter Projesi Oluřturma

İlk olarak flutter'ın resmi sitesinden bilgisayar'ımın işletim sistemine uygun versiyonu indirdim. İnen zip dosyasından flutter klasörünü çıkararak yerel diskte src klasörünün içine aktardım. Sonrasında ise dosyanın dizinini ortam değışkenlerinde sistem path'ine ekledim.



Görsel 3.9 Flutter indirme sayfası



Görsel 3.10 Flutter dosya dizininin ortam değışkenlerine eklenmesi

```
C:\PROJELER>flutter create myapp
Creating project myapp...
Resolving dependencies in myapp... (1.5s)
Got dependencies in myapp.
Wrote 129 files.

All done!
You can find general documentation for Flutter at: https://docs.flutter.dev/
Detailed API documentation is available at: https://api.flutter.dev/
If you prefer video documentation, consider: https://www.youtube.com/c/flutterdev

In order to run your application, type:

$ cd myapp
$ flutter run

Your application code is in myapp\lib\main.dart.

C:\PROJELER>code .

C:\PROJELER>cd myapp

C:\PROJELER\myapp>code .
```

Görsel 3.11 Flutter projesi oluşturma ve vs code üzerinde açılması

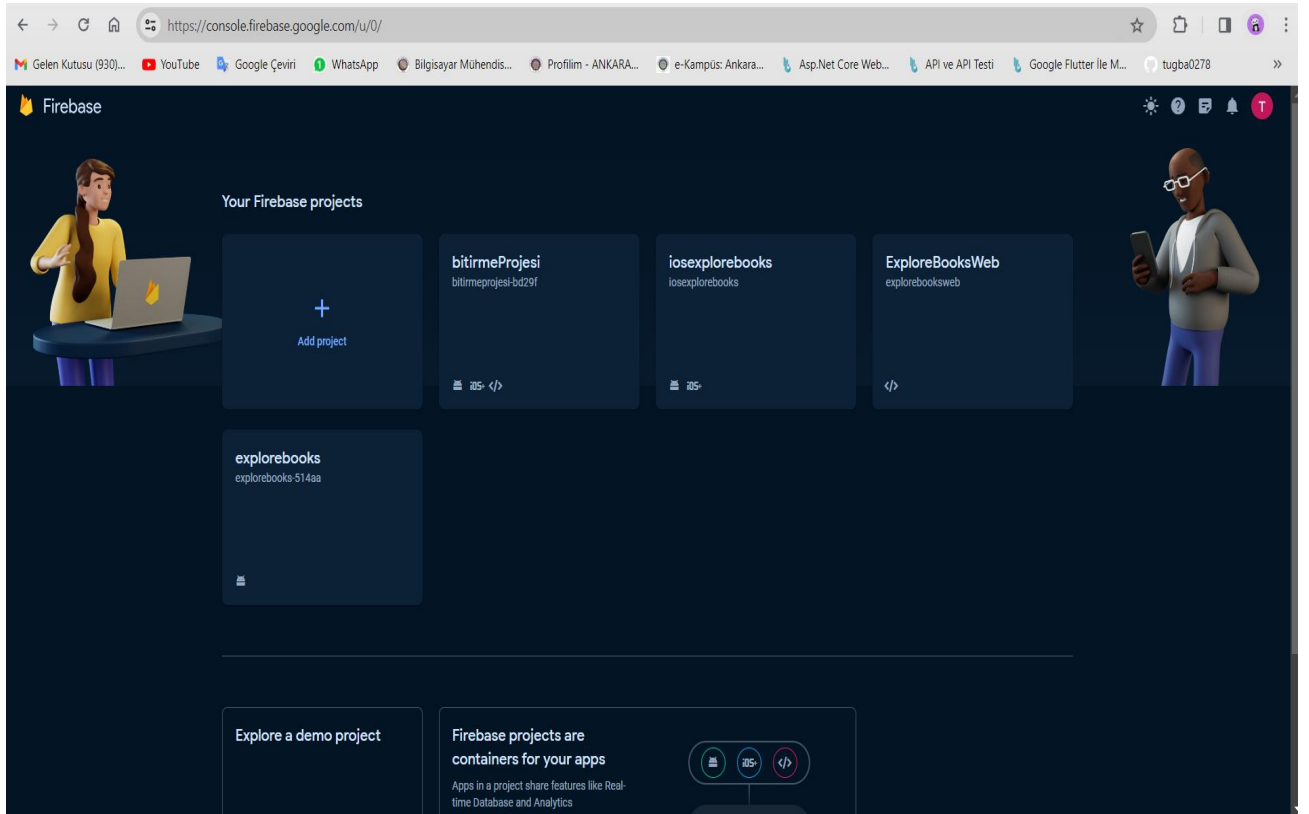
Yukarıdaki görselde görüleceği üzere terminalde projemi depolamak istediğim dizine giderek bir proje oluşturdum ve sonrasında proje dizinine girerek visual studio code'da açtım.

4. VERİTABANI BAĞLANTISI

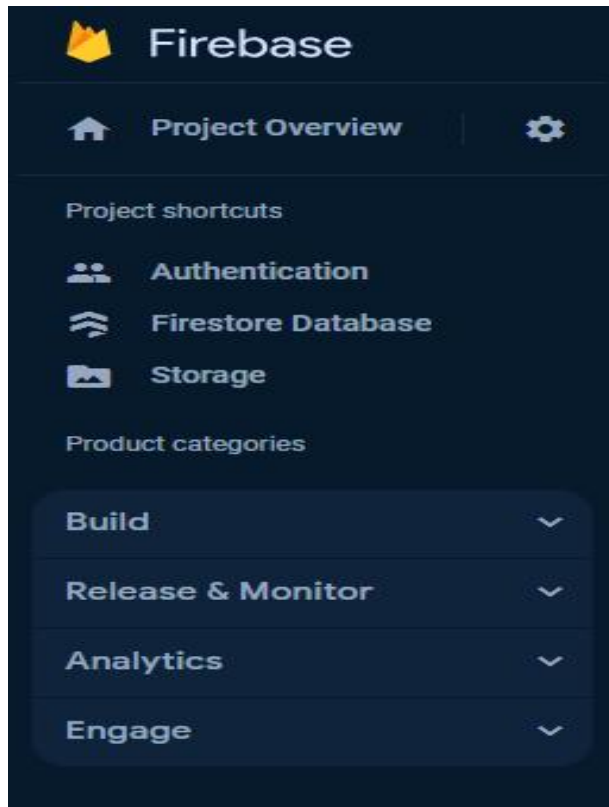
4.1 Firestore Projesi Oluşturma

Firestore'in resmi sitesinde kayıt olduktan sonra giriş yaparak proje ekleyebilirsiniz.

Proje eklendikten sonra kullanacağınız özellikleri ekleyebilirsiniz. Ben authentication, firestore database ve storage ekledim. Örnek bir görünüm aşağıdaki görsellerde verilmektedir.

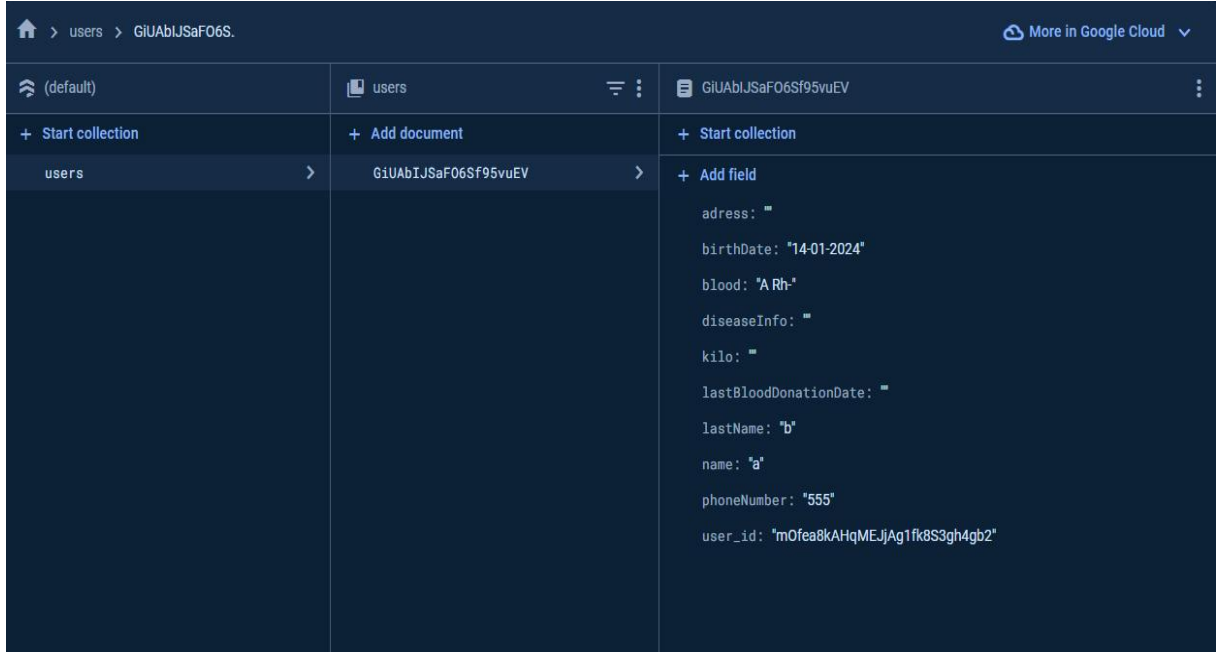


Görsel 4.1 Firebase'de proje oluşturma



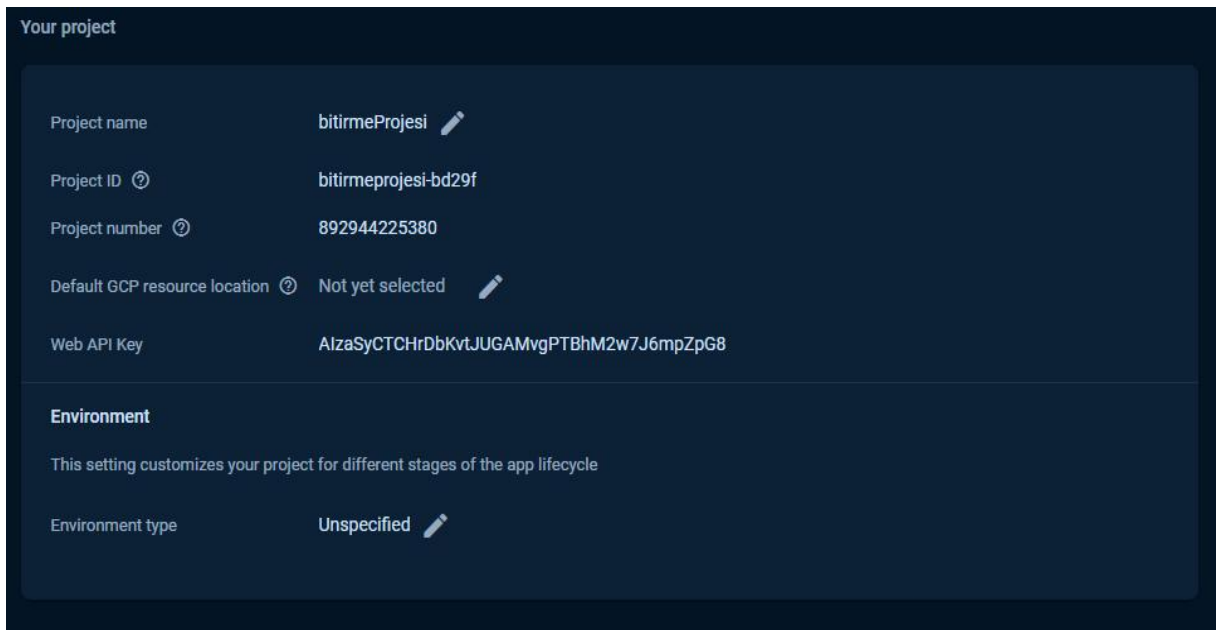
Görsel 4.2 Firebase özellikleri ekleme

Projenize eklemek istediğiniz özellikleri 'build'e tıklayarak ekleyebilirsiniz. Authentication kısmında ise giriş yapma seçeneğimi email-password olarak etkinleştirdim.



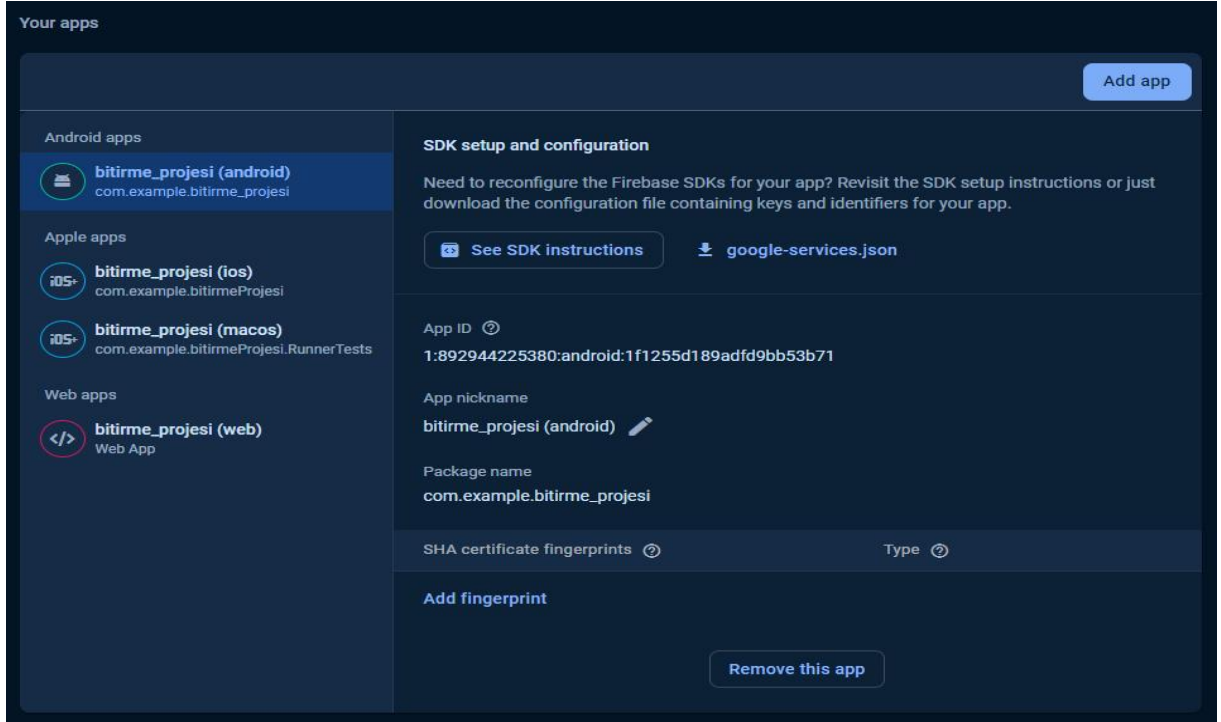
Görsel 4.3 Firestore database görünümü

Kullanıcıdan alınan bilgiler 'users' adlı koleksiyonda depolanmaktadır. Kullanıcı, bilgilerini güncellemek istediğinde kullanıcı profiline girerek güncellemeyi gerçekleştirebilmektedir.



Görsel 4.4 Firebase proje ayarları

Flutter ve firebase projemizin bağlantısını yaparken proje ayarlarına girerek API key'i öğrenebiliriz. Ayrıca hemen altında yer alan uygulama yapılandırma kısmından da projemizin hangi platformlarda çalışmasını istediğimizi seçebiliriz.



Görsel 4.5 Firebase uygulama yapılandırması

4.2 Firestore Flutter Bağlantısı

İlk olarak firebase paketlerini pubspec.yaml dosyasına ekledim. Bunun için firebase'in pub dev sayfasından yararlandım.

```
dependencies:
  flutter:
    sdk: flutter
  url_launcher: ^6.1.13
  intl: ^0.17.0

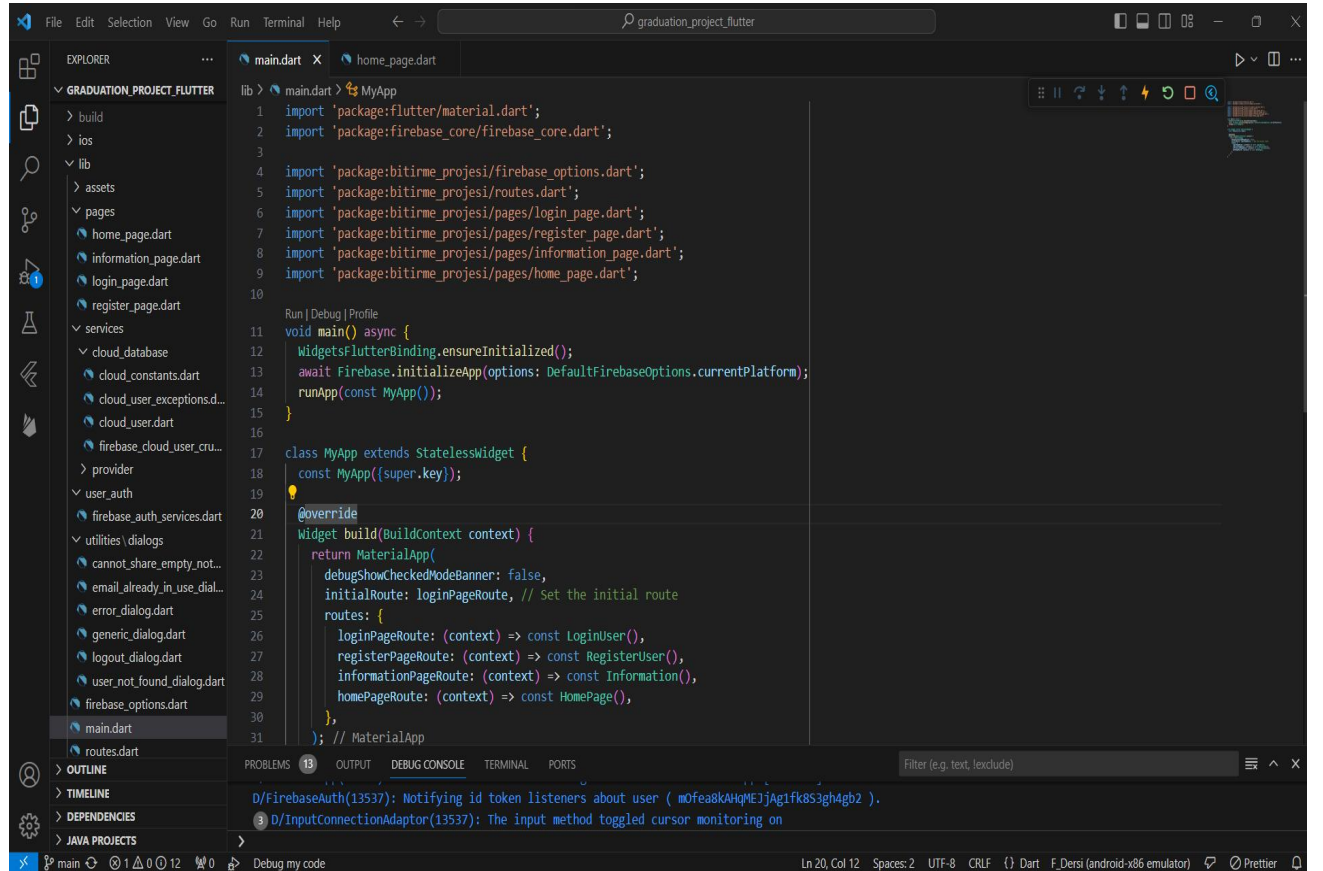
# The following adds the Cupertino
# Use with the CupertinoIcons class
cupertino_icons: ^1.0.2
datetime_picker_formfield: ^2.0.1
firebase_core: any
cloud_firestore: ^4.14.0
firebase_auth: ^4.15.1
firebase_storage: ^11.5.4
```

Görsel 4.6 Firestore paketlerinin vs code'a eklenmesi

Proje yapılandırmasını yaparken kurulması gereken CLI'ı kurdum ve projemin kök dizininde terminali açıp aşağıdaki komutları girerek flutter projem ve veritabanı bağlantısını kurmuş oldum.

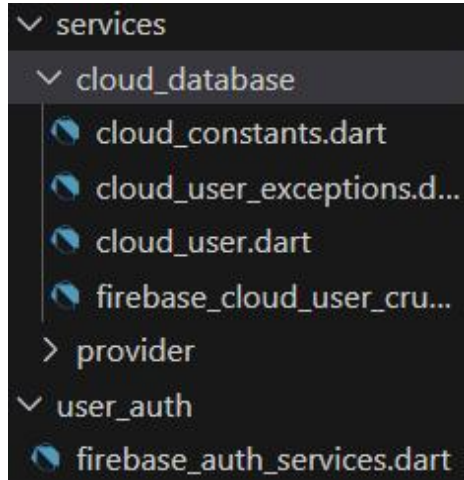


Görsel 4.7 Flutter firebase bağlantısı komutları



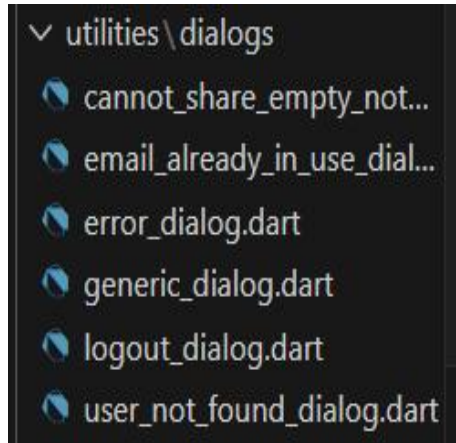
Görsel 4.8 Projenin genel görüntüsü

Uygulamaya giriş yaparken veya kaydolurken user_auth klasörünün altındaki dart dosyası ile authentication işlemleri sağlanmaktadır. Ayrıca kullanıcı güncelleme,silme vb işlemler için services klasörünün altındaki dart dosyaları kullanılmaktadır.



Görsel 4.9 Services ve user_auth dosyaları

Giriş ve kayıt olurken bazı diyalog gösterimleri için kullandığım fonksiyon dosyaları görselde gösterilmektedir.



Görsel 4.10 Diyalog dosyaları



Görsel 4.11 Diyalog ile ilgili örnek bir görünüm

5. SONUÇ

Sonuç olarak, bu proje, kullanıcı dostu bir arayüz ve kolay erişilebilirlik sayesinde, kan bağışının hayati önemini geniş bir kitleye ulaştırmayı başarıyor. Toplumsal farkındalığı artırarak kan bağışı sürecine daha fazla insanın katılımını teşvik ederken, aynı zamanda teknolojiyi kullanarak sağlık sektöründe önemli bir sosyal sorumluluğu yerine getiriyor. Bu uygulama, hem bireysel sağlık katkılarına hem de toplum sağlığına önemli bir katkı sağlayarak, kan bağışının toplum üzerindeki etkisini güçlendirmekte ve bu hayati sürecin daha geniş kitleler tarafından benimsenmesini sağlamaktadır.

KAYNAKÇA

<https://www.memorial.com.tr/saglik-rehberi/kan-vermenin-faydalari>

<https://www.tamamlayicisaglik.com/blog/saglik/bir-damla-kan-bin-damla-umut>

<https://www.paynet.com.tr/blog/flutter-nedir-ne-ise-yarar#:~:text=Flutter%20ise%20uygulama%20geli%C5%9Ftirmek%20isteyenler,web%20uygulamalar%C4%B1%20haz%C4%B1rlamak%20da%20m%C3%BCmk%C3%BCnd%C3%BCr.>

<https://www.datamarket.com.tr/sozluk/react-native/#:~:text=React%20Native%2C%20Facebook%20taraf%C4%B1ndan%20geli%C5%9Ftirilen,kullan%C4%B1lan%20React.js%20k%C3%BCt%C3%BCphanesine%20dayan%C4%B1r.>

<https://coderspace.io/sozluk/flutter>

<https://sinanyilmaz.medium.com/android-firebase-authentication-email-parola-ve-google-cc154d1bf5fe>

<https://www.mobilhanem.com/android-firebase-cloud-firestore-kullanimi/>

<https://www.userspots.com/rehber/figma-nedir-nasil-kullanilir>

<https://code.visualstudio.com/download>

<https://docs.flutter.dev/get-started/install>

<https://pub.dev/>