

Emine DEMİRCAN B171200008

Gülşah SEVİNEL B171200025

Projemizi yaparken kullanacağımız teknolojiler

-Flutter

-Firebase

-SQLite

-Dart

Neden Flutter kullanıyoruz?

Zaman Tasarrufu

Genelde kullanılan zamanın bir bölümünde ve mobil uygulama oluşturma'nın kolaylığı nedeniyle Flutter, mobil uygulamanın daha hızlı tasarlanmasına ve geliştirilmesine katkıda bulunduğundan zaman tasarrufu sağlar.

Flutter, widget kütüphanesindeki kodun %95'inin tekrar kullanılmasını sağlar. Ana ekranda uygulama değişikliklerinin güncellenmesi bir dakikadan daha az bir zaman sürer.

Uygulama ne kadar hızlı pazara ulaşırsa bizde o daha çabuk geri dönüş alır ve eksiklerimizi daha hızlı bir şekilde gideririz.

Açık Kaynak

Ücretsiz erişilebilen yüzlerce üçüncü taraf paket, reklam, video, bulut vb. birçok yardımcı eklentiler içerir. Açık kaynak olması sebebi ile de bir sorun ile karşılaşıldığında internet üzerindeki topluluklardan kolay bir şekilde soruna çözüm bulunabilmektedir.

Daha iyi kaynak kullanımı

Flutter çapraz bir platformdur, aynı altyapıyı kullanarak hem IOS hem de Android tabanlı uygulamalar geliştirebilirsiniz. Bu her açıdan bize tasarruf sağlayacaktır.

Performans

Artık mobil uygulamalarda animasyon dediğimiz şeyi sürekli görmekteyiz. En sıradan gördüğünüz uygulamada bile listeler, switchler, checkbox'lar, sayfa geçişleri, diyaloglar, gesture efektleri vs. derken aslında her yer animasyonla doluyor. Böylece yüksek ve stabil FPS bir mobil uygulama için kritik hale geliyor.

Özelleştirme Kolaylığı

Flutter'da arayüz katmanı ne native SDK'lere bağımlı ne de onların üzerine kurulan birtakım soyutlamalara, arayüzlere, köprülere vs. bağımlı. Tek bir dille, donanım seviyesinde native, tek bir codebase'den aklınıza gelebilecek her türlü kullanıcı arayüzü mümkün. Üstelik Flutter çok zengin bir hazır arayüz bileşenleri kataloğu sunmaktadır.

Tutarlılık

Android ve IOS SDK'leri elbette birbirinden çok farklı. Cross-platform çözümlerde kodu olabildiğince paylaşmak isteriz ve bu gibi platform farklılıklarında zorlanabiliriz. Kod ayrımı yapmak, ne zorluk varsa ondan en az 2 tane, hatta 3 tane demek. Flutter'da platformlar arasında ne kadar ayrım yapacağınız sizin tercihinize bağlı. Çünkü hazır arayüz bileşenleri native SDK'leri bire bir taklit edecek şekilde ayrılmış evet, ama cross-platform da çalışabiliyorlar ve daha da önemlisi kendi çözümlerinizi de üretebilirsiniz.

Dart

Flutter da uygulamalar geliştirebilmek için Dart dili kullanılır.

Neden Firebase kullanıyoruz?

Herhangi bir platformdan Firebase kullanarak kullanıcı ekleyebilir, silebilir kullanıcının bilgileri anlık olarak değiştirebiliriz (Yani ios uygulamasında eklediğimiz kullanıcıyı hem webden hem Android'den hızlıca takip edebiliriz). Bunun dışında yazdığınız bir uygulamanın analizlerini ve raporlarını rahatlıkla takip edebilirsiniz. Örnek olarak uygulama ne zaman nasıl hata almış gibi. Hangi şehirde daha çok uygulamanızı kullanıyor gibi birçok analize ulaşabilirsiniz. Bence en büyük özelliklerinden birisi bir platformda değiştirdiğiniz herhangi bir bilgiyi (DB içindeki) diğer platformdan kolayca kullanıcıya yansıtabilirsiniz.

Firebase'in Bize Sağladığı Özellikler

- Eşzamanlı veritabanı
- Kullanıcı girişlerinde yetkilendirme
- Depolama
- Makine öğrenme kiti
- Performans test ortamı
- Platform arası ortak biçimde uygulama analizleri
- Ortak fonksiyon
- Reklam materyalleri
- Bildirim gönderimi
- Genel uygulama yönetimi

Bizim uygulamamızı geliştirirken Firebase'in ürün ve hizmetini bir arada kullanabileceğiz. Bu hizmet ve ürünler şu şekilde:

Realtime Database: Veri tutmaya yarayan büyük bir veritabanı uygulaması diyebiliriz. Temel amacı anlık olarak verileri senkronize etmektir.

Test Lab: Tasarım aşamasında olduğumuz mobil uygulama ile ilgili bize ön bilgiler sunar. Bu sayede uygulamada denediğimiz her şeyi hızlıca test edebiliyoruz.

Crashlytic: Anlık olarak loglama yapmamızı sağlar. Ayrıca hatalarımızı anında deşifre eder ve bu sayede güncellemeler yapmamıza olanak sağlıyor.

Firestore: Anlık senkronizasyonun daha ölçeklenebilir halidir.

Neden SQLite kullanıyoruz?

Herhangi bir yazılım veya sunucu kurulumu gerektirmez. Bu sayede, bu modülü kullanabilmek için öncelikle bir sunucu yapılandırmanıza da gerek yoktur. Bazı veritabanlarını kullanabilmek için arka planda bir veritabanı sunucusu çalıştırıyor olmanız gerekir. SQLite'ta ise böyle bir şey yapmazsınız.

Aynı zamanda diğer veritabanlarına göre boyut olarak daha küçük ve pek çok veritabanı alternatifine göre daha basittir.

SQLite kullanımı için bir yöneticinin yeni bir veritabanı örneği oluşturmasına veya kullanıcılara erişim izni vermesine gerek yoktur.

SQLite açık kaynak bir yazılımdır. Dolayısıyla SQLite kodlarının her zerresini istediğiniz gibi kullanabilir, değişikliğe uğratabilir, satabilir ve ticari olan/olmayan bütün uygulamalarınızda gönül rahatlığıyla kullanabiliriz.