

**Öğr.Gör.Nilgün İncereis**

Ömer Faruk ERTÜRK

18MY93003

**Online bilgi yarışması**

**Bitirme Projesi Raporu**

**ONLİNE**

**BİLGİ YARIŞMA UYGULAMASI**

**BİTİRME PROJESİ**

**ÖMER FARUK ERTÜRK**

**tarafından**

**OKAN ÜNİVERSİTESİ**

**Mobil Teknolojileri Bölümü**

**Önlisans**

**derecesi şartını sağlamak için sunulmuştur.**

**Onaylayan:**

**Danışman**

**Öğr.Gör.Nilgün İNCEREİS**

**Üye Üye**

**Üye Üye**

**Mayıs 2020**

**Program: Mobil Teknolojileri**

**ÖNSÖZ**

“Online Bilgi Yarışma Uygulaması” konulu bu çalışma, Okan Üniversitesi Mobil Teknolojileri Bölümü’nde “Bitirme Projesi” olarak hazırlanmıştır.

Bitrime tez danışmanlığımı yaparak, bu çalışmamda bana her türlü yardımı ve desteği esirgemeyen saygıdeğer hocam Öğr.Gör.Nilgün İNCEREİS’e ve çalışma süresi boyunca bana desteğini esirgemeyen arkadaşlarıma teşekkür ederim.

**ÖZET**

**ONLİNE**

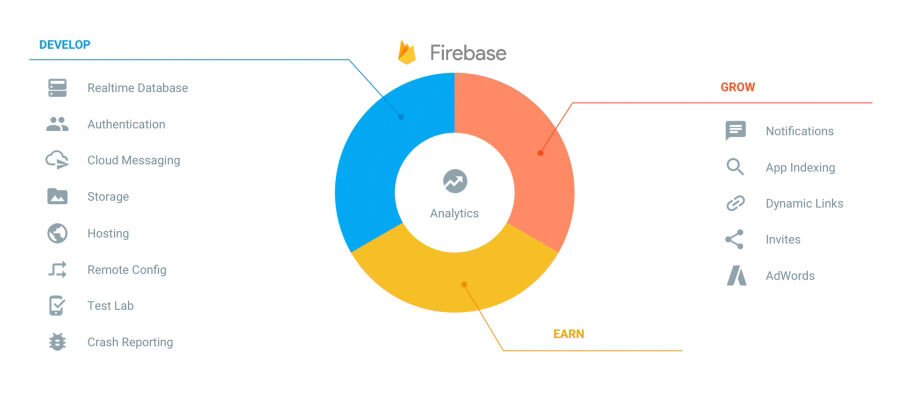
**BİLGİ YARIŞMA UYGULAMASI**

**Proje Danışmanı:** **Öğr.Gör.Nilgün İNCEREİS**

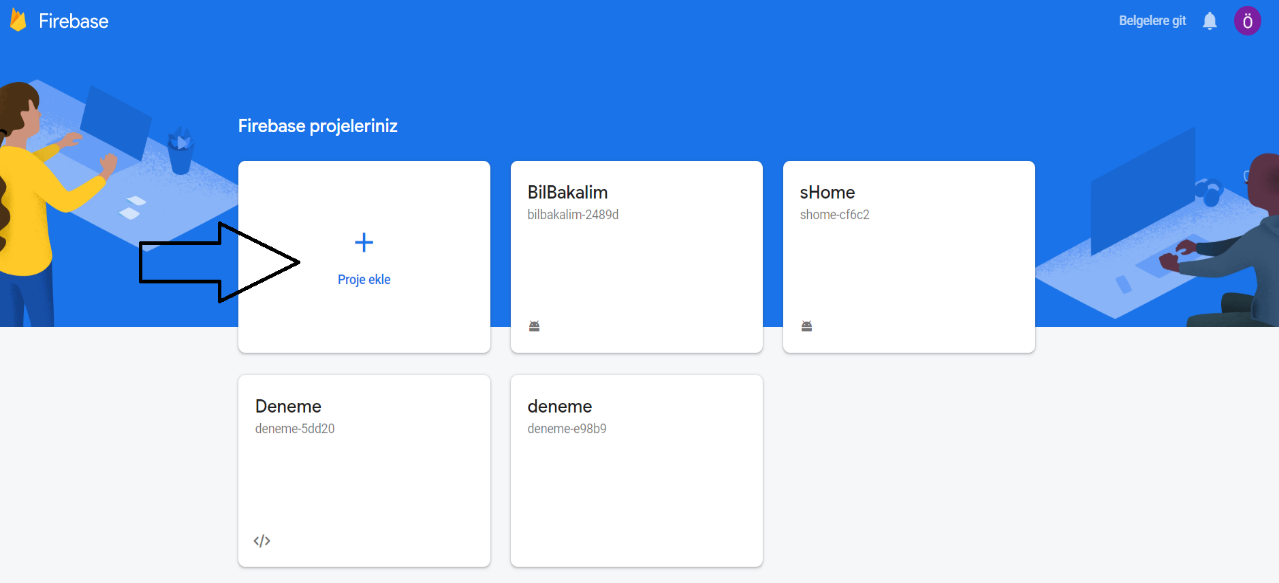
**Bu tez, Android Studio platformu üzerinden yapılan online bilgi yarışması ile hedef kitlenin boş vakitlerinde akıllı telefonları üzerinden eğlenerek öğrenecekleri bir projeyi hedeflemektedir.**

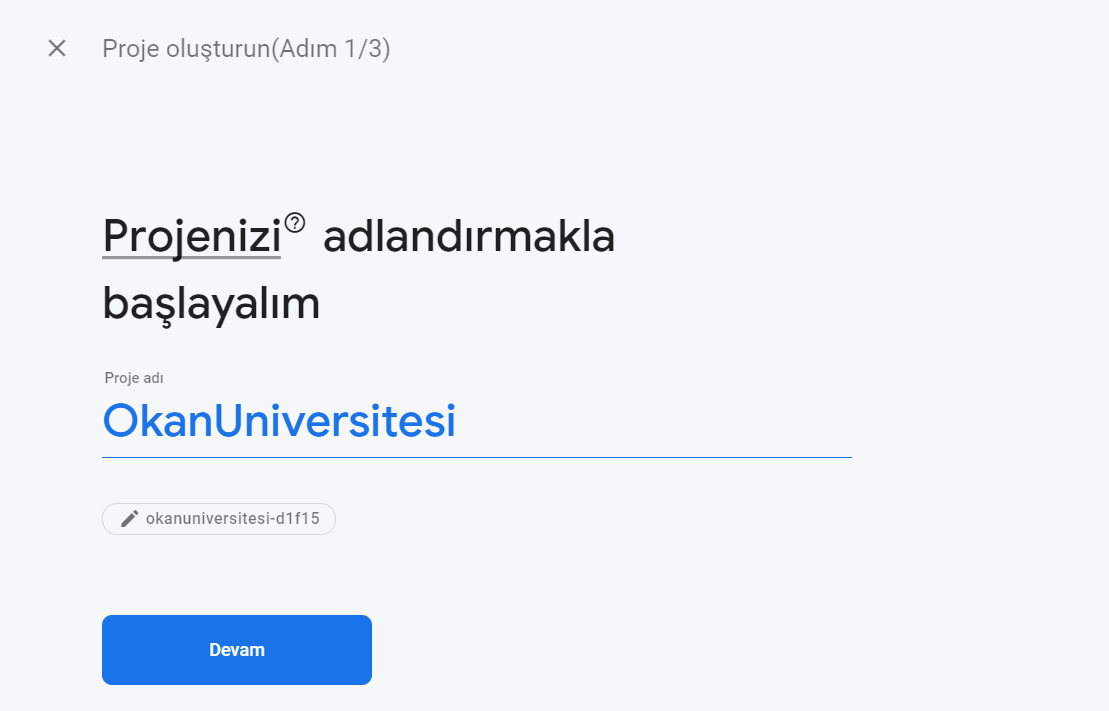
**Anahtar Kelimeler: Firebase, JSON, Android Studio, Adobe Photoshop, Database, Java, Picasso, Fragment, viewHolder, Interface.**

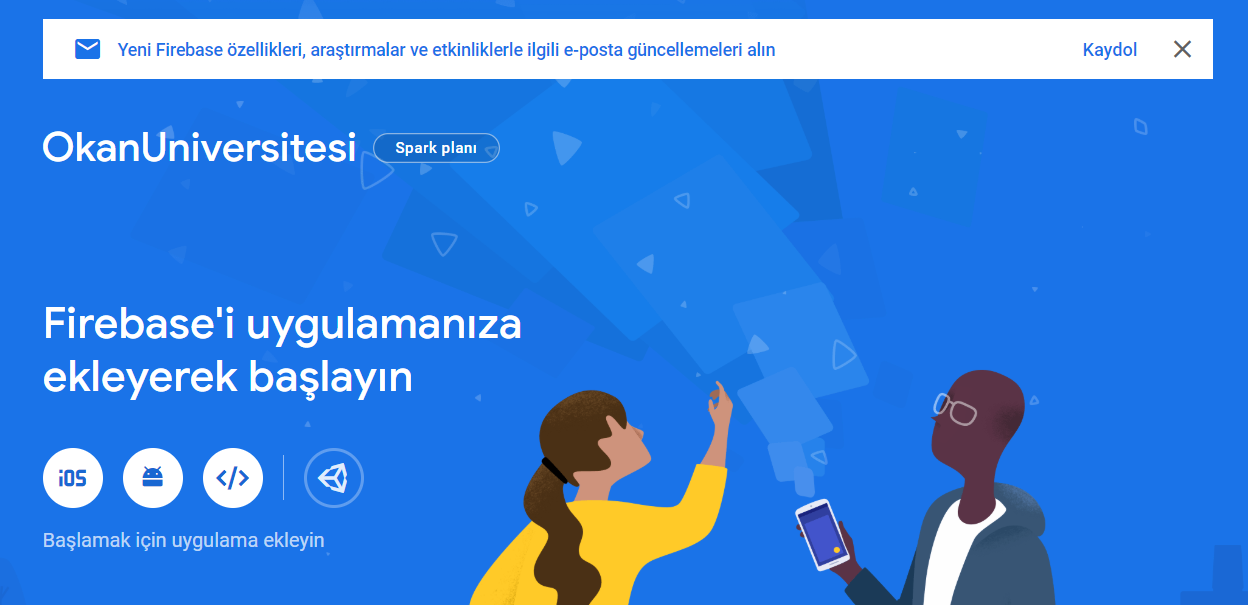
**İÇİNDEKİLER**

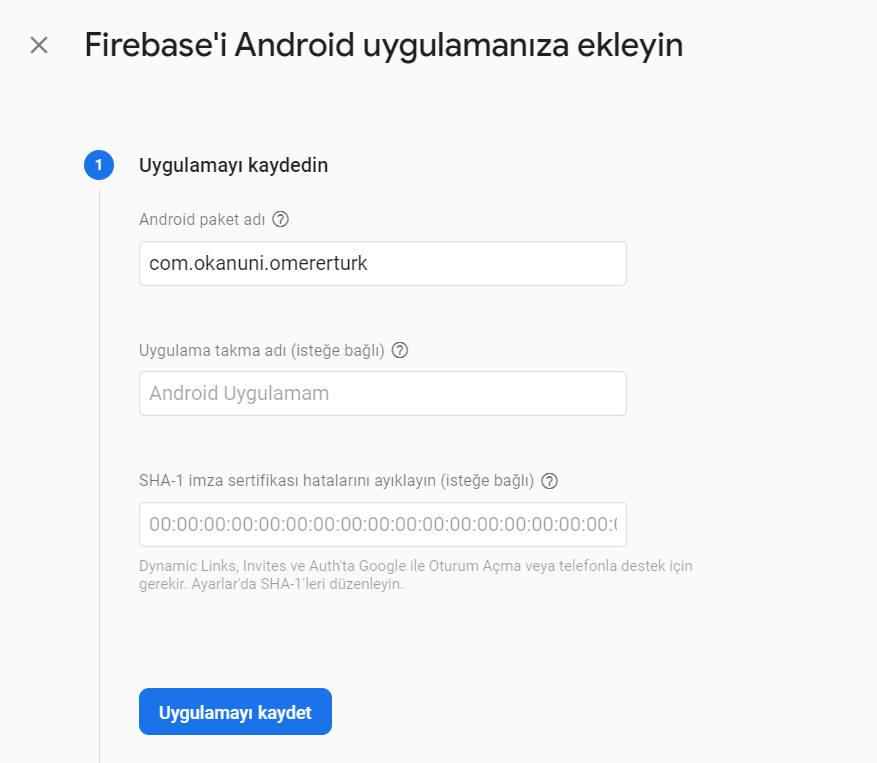
1. **GİRİŞ**5
2. **GENEL BİLGİLER5**
   1. Firebase Nedir?5
      1. Firebase, Android Studio Projesine Nasıl Eklenir?6
         1. Firebase Realtime Database9
            1. Firebase Veri Tabanı Yapısı9
   2. Android Studio ve Java10
      1. Picasso Kütüphanesi10
      2. Fragment Nedir?10
      3. Interface Nedir?10
      4. Viewholder Nedir?10
      5. Splash Screen Yapım11
3. **PROJE BİLGİLERİ12**
   1. Neden Bu Proje?12
   2. Kullanılan Teknolojiler12
4. **PROJE YAPILARI13**
   1. Veritabanı Yapısı13
      1. Veritabanı İçeriği13
         1. Kategori 13
         2. Kullanıcılar14
         3. Sıralama 14
         4. Sorular14
   2. Proje Sayfaları ve Tasarımı 15
      1. Giriş Sayfası15
      2. Kayıt Sayfası16
      3. Kategoriler Sayfası16
      4. Test Başlangıç Sayfası17
      5. Test Sayfası17
      6. Sonuç Sayfası 18
      7. Sıralama Sayfası18
5. **EKLER**19
   1. Projenin Kaynak Kodları19
   2. XML Sayfaları19
   3. Görsel Çalışmalar24
6. **KAYNKAÇA25**
7. **GİRİŞ  
     
   Proje Adı:** Bil Bakalım  
     
   **Proje Konusu:** Birbirinden farklı mantık sorularını kullanıcılar, belirlenen süre içerisinde mobil uygulama üzerinden çözmeye çalışması.  
     
   **Hedef Kitle:** 10-60 yaş arası herkese hitap etmektedir.  
     
   **Platform:** Android Studio  
     
   **Veritabanı:** Firebase Realtime Database  
     
   **Kullanılan Dil:** Java  
     
   **Çizim Aracı:** Adobe Photoshop
8. **GENEL BİLGİLER**
   1. **Firebase Nedir?**Google Firebase, mobil ve web uygulamaların server üzerinden geliştiricinin uğraşmasına gerek bıraktırmadan kulanıcı giriş yetkisi ve aynı zamanda verileri anlık olarak senkron içinde tutulmasını sağlayan platformdur. Günümüz zamanında çıkan uygulamalar tüm marketlere hitap ettiğinden dolayı iOS, Android ve aynı zamanda web platformlarında geliştirilir. Firebase tüm platformlar için geliştirilmiş olan ortak çözümdür.  
      1. **Firebase, Android Studio Projesine Nasıl Eklenir?**

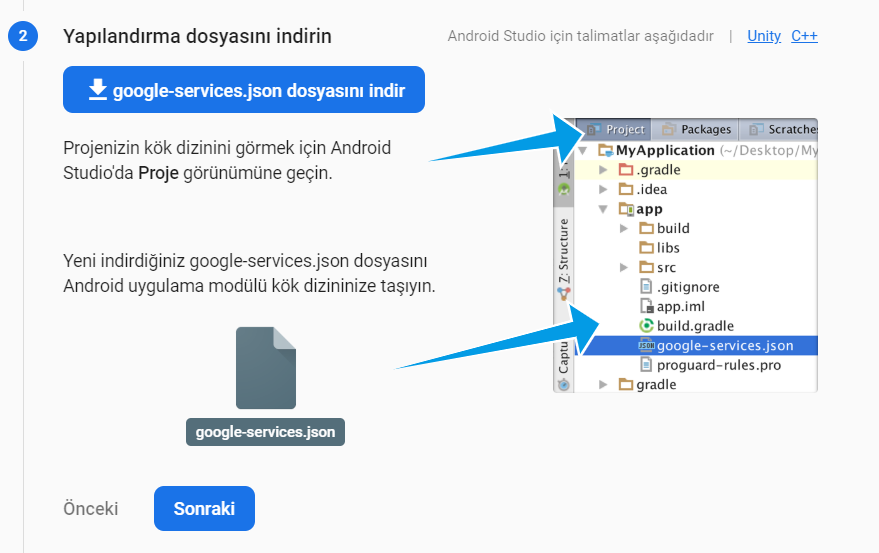
* Android Studio üzerinden projemizi oluşturduktan sonra firebase.google.com adresine giderek Google hesabımız ile giriş yapıp “Proje Ekle” kısmına tıklıyoruz



* 3 adımlı proje oluşturma kısımlarını tamamlıyoruz
* Yönlendirilen sayfada uygulama platformunu seçerek devam ediniz.



* Android paket adını, uygulama takma adını ve imza serfitakasını girdikten sonra uygulamayı kaydede basıyoruz.  
    
  
* Yapılandırma dosyasını (Google-services.json) indirdikten sonra Android Studio üzerinden kök dizine bu dosyayı aktarıyoruz.



* Proje düzeyinde build.gradle dosyasına

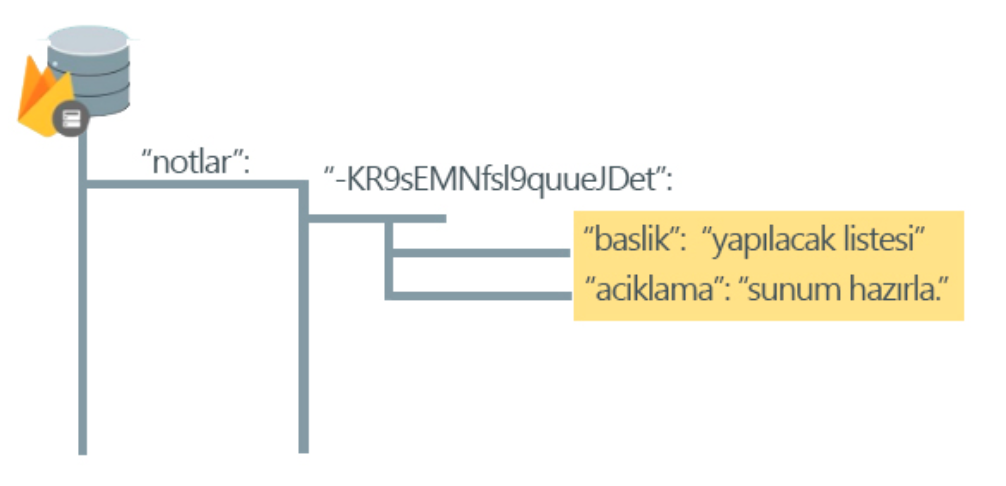
|  |
| --- |
| classpath 'com.google.gms:google-services:4.3.3' |

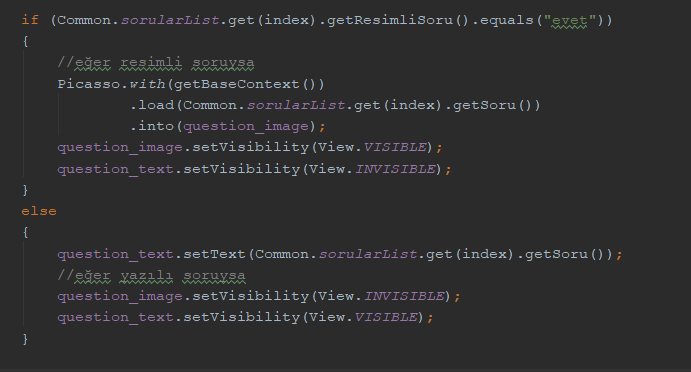
Uygulama düzeyinde build.gradle dosyasına

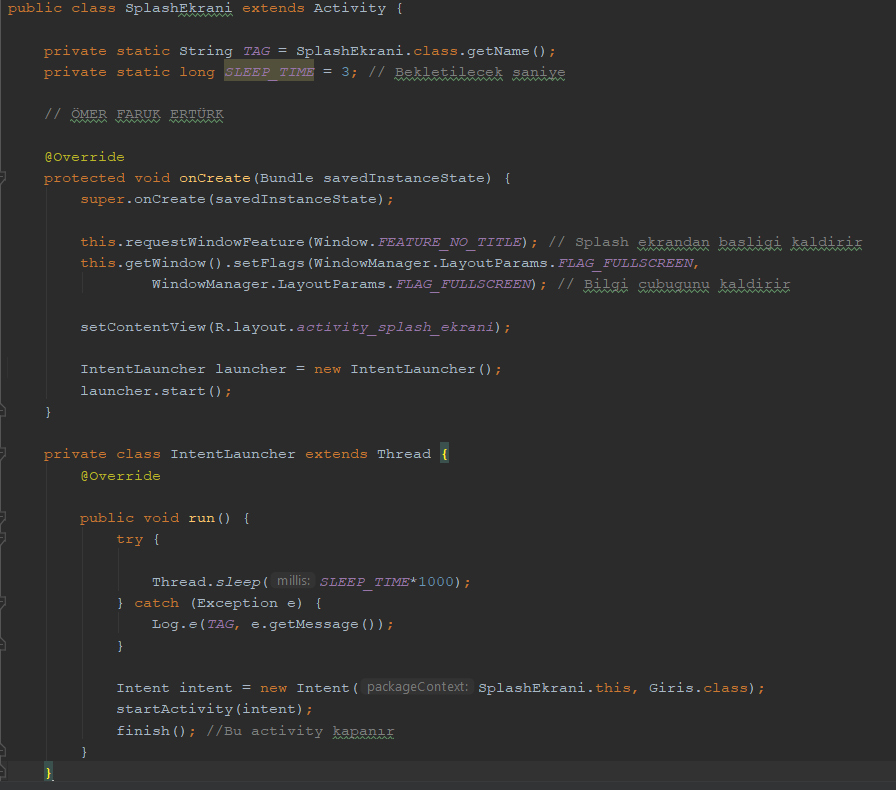
|  |
| --- |
| apply plugin: 'com.android.application' |
| apply plugin: 'com.google.gms.google-services' |

Satırlarını ekledikten sonra uygulamamızı senkronize edip firebase bağlantısını başarıyla sağlamış oluyoruz

* + 1. **Firebase Relatime Database** Firebase Realtime Database, bulut tabanlı olmak üzere NoSql veritabanı türüdür. Asla sql sorugsuna gerek kalmadan Json parametleri sayesinde yönetilmektedir. Veri tutmanın yanı sıra asenkron çalışması sayesinde veri değişimlerini anlık olarak uygulma üzerinde gösterme imkanı sunuyor.  
         
       1. **Firebase Veri Tabanı Yapısı** Firebase Realtime Database, tüm verileri JSON yani Javascript Object Notation objesi olarak hafızasında tutmaktadır. Bu veritabanı yapısını bulut tabanlı olmak üzere JSON ağaç yapısı olarakta düşünebiliriz. Json ağacına herhangi bir veri eklemesi yaptığımız taktirde JSON formatı olarak bir düğüm haline gelir.



* 1. **Android Studio ve Java**
     1. **Picasso Kütüphanesi** Picasso kütüphanesi, Android Studio tabanlı yazılmış uygulamalarda görsellerin yüklenmesi ve işlenmesi için oluşturulmuş bir kütüphanedir. Kullanımı oldukça kolay olduğu gibi görsellerin çerezlenmesini sağlamaktadır.
     2. **Fragment Nedir?** Bu projede skor listelemesi için ve kategori listelemesi için Fragment kullanılmıştır. Fragmentler, Activtiylerin içersinde çağrılır. Fragment’in ne olduğunu bir örnekle açıklamak gerekirse; bir kitap okuduğumuzu varsayalım ve bu kitabın her sayfasını ACTİVİTY olarak düşünelim. Geçtiğimiz sayfaları Activity olarak düşünürsek, sayfa bittiğinde bir Activityi kapatıp diğerine geçmiş oluyoruz. Fragmentler ise bu kitap üzerine yapıştırdığımız post-it olarak düşünebiliriz. Birisi ile işimiz bittiğinde kaldırıp, yeni post-it ekleyebiliriz.
     3. **Interface Nedir?** Interface, java dilinde sınıflar nasıl tanımlanıyorsa o şekilde tanımlanır, tek farkı class yerine Interface yazılmasıdır. Fakat somut olarak değil de soyut olarak sınıf sayabiliriz. Sınıflardan farklı olarak statik değişkenleri mevcuttur. Activityler arası alışverişi Intentler ile yapabiliyorken, Fragmentler arası geçişi ise Interface ile yapmaktayız.
     4. **Viewholder Nedir?** Viewholder, ListView kullanılan uygulamalarda iyleştirme amaçlı geliştirilmiş bir yöntemdir. List için her bir nesne yaratıldığı zaman findViewByID() methoduyla nesneleri çağrıp ekrana basıyoruz. ViewHolder sayesinde bu işlemi her seferinde yapmaya gerek kalmadan veri şişmesi durumunun önüne geçiyoruz.
     5. **Splash Screen Yapımı** Uygulamalara estetiklik kazandırmak için giriş ekranı yani splash screen kullanıyoruz. Uygulamamıza splash screen eklemek için;
* İlk olarak Splash Screen tasarımı yapılmalı ve bu görseli Drawable dosyasına aktarılmalı
* Yeni bir activity çağrılarak gerekli kodlar yazılır

****

* Activity tasarımını yaptıktan sonra AndroidManifest.xml dosyasına gelerek uygulamamızın açılış ekranını classını SplashEkrani olarak değiştiriyoruz

|  |
| --- |
| <activity android:name=".SplashEkrani"> |
| <intent-filter> |
| <action android:name="android.intent.action.MAIN" /> |
| <category android:name="android.intent.category.LAUNCHER" /> |
| </intent-filter> |
| </activity> |

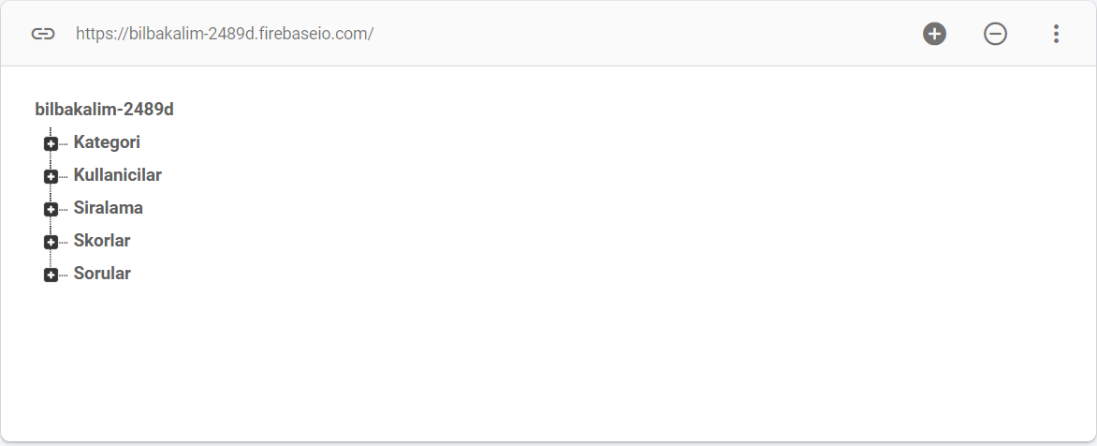
1. **PROJE BİLGİLERİ**
   1. **Neden Bu Proje?** Mantık sorularını çözmeyi aslında merak etmeyi insanların çoğu sevmektedir. Ayrıca doğru cevaplar sonucu insanlarda oluşan haz sayesinde uygulamayı tekrar kullanma yüzdeliği artması ve aynı zamanda boş vakitlerini beyin egzersizi yaparak geçirmeleri hedeflenmektedir.
   2. **Kullanılan Teknolojiler**Kullanılan Platform: Android Studio  
      Kullanılan Tasarım Programıı: Adobe Phtoshop   
      Kullanılan Veritabanı Platformu: Firebase

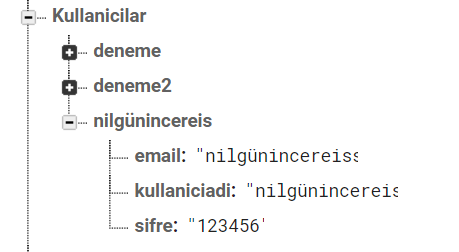
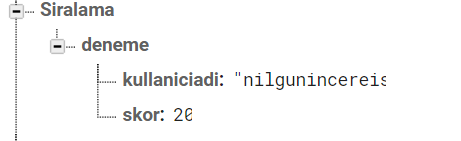


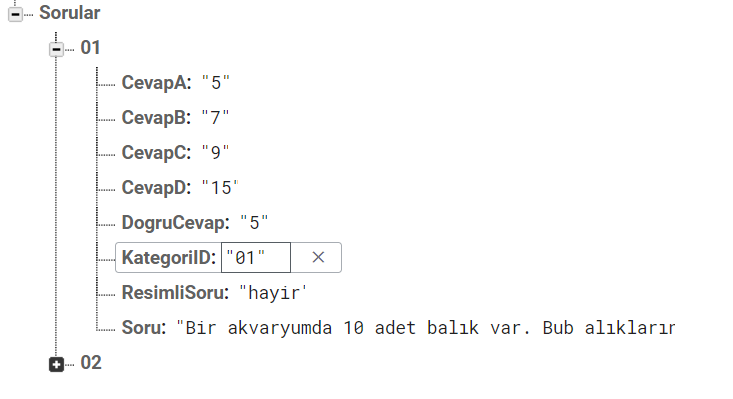


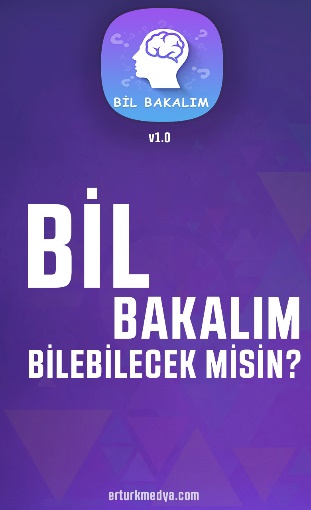
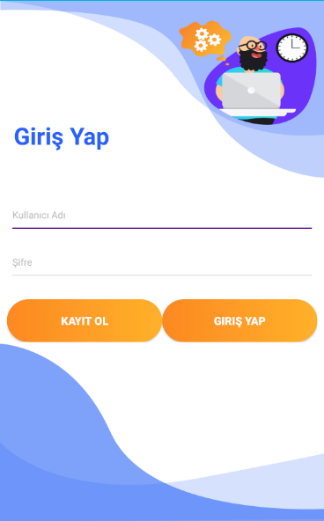
1. **PROJE YAPILARI**
   1. **Veritabanı Yapısı** Bil Bakalım projesinde veritabanı olarak Firebase Realtime Database tercih edildi. Bu tercihin en büyük sebepleri; kullanışlı olması ve anlık olarak kullanıcılara güncelleme fırsatı sunmaktır.

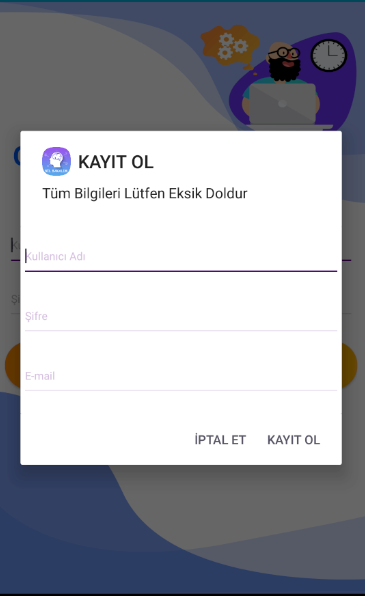


* + 1. **Veritabanı İçeriği** Veritabanın başlıca tabloları; kategori, kullanıcılar, sıralama, skorlar ve sorulardır.  
       1. **Kategori** Kategori tablosunda, kategoriler ID’lere ayrılmıştır. Her kategoriye özel Kategori ismi ve kategori resmi eklenebilmektedir.

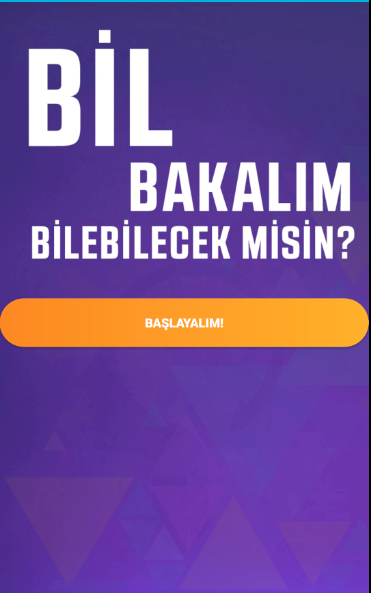
* + - 1. **Kullanıcılar** Kullanıcılar tablosunda, kullanıcı adları, e-mail adresleri ve şifreleri bulunmaktadır.
      2. **Sıralama** Sıralama tablosunda, kategorinin en iyi skoruna ulaşan kullanıcının adını ve skorunu göstermektedir.  
           
           
         
      3. **Sorular** Sorular tablosundan soru ekleme işlemi yapılmaktadır. Sorular ID’lere ayrılmakla beraber; CevapA, CevapB, CevapC, CevapD, DogruCevap, KategoriID, ResimliSoru, Soru olmak üzere ayrılmıştır

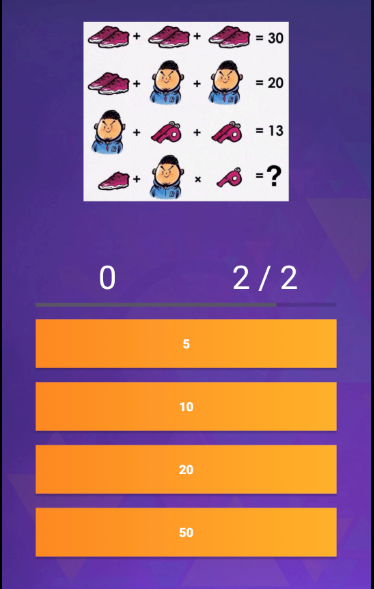


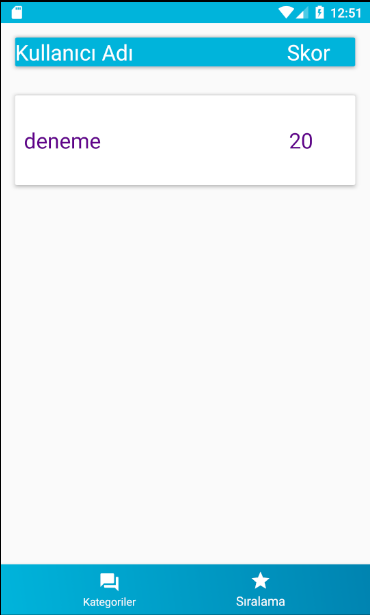
* 1. **Proje Sayfaları ve Tasarımı**
     1. **Splash Screen** Splash Screen oluşturmak yeni bir activity oluşturuldu. Bu activityde full screen olması için uygulama başlığı ve bildirim çubuğu kaldırıldı. 1 adet imageView’dan oluşmaktadır.
     2. **Giriş Sayfası** Giriş sayfasında kullanıcılar uygulamaya mevcut üyeliklerine giriş yapabilir veya yeni bir üyelik oluşturabilmektedir. sayfasında, 2 adet editText, 2 adet button, 1 adet textView, background resmi, RelativeLayout ve LinerLayout kullanılmıştır.
     3. **Kayıt Sayfası** Giriş Sayfasından “Kayıt Ol” butonuna tıklayan kullanıcılar alertDialog ile karşılarına kayıt ol sayfası çıkmaktadır. Bu sayfada kullanıcıdan, kullanıcı adı, şifre ve e-mail istenmektedir. Yanlışlıkla tıklayan kullanıcılar için “İptal Et” ve kayıt işlemini tamamlamak isteyen kullanıcılar için “Kayıt Ol” eklenmiştir.

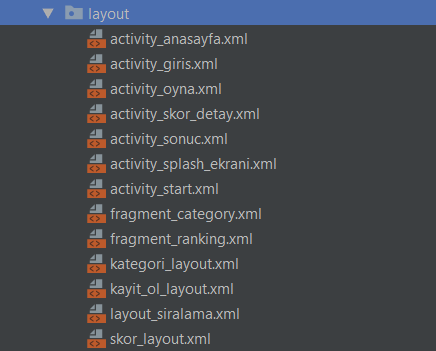
****

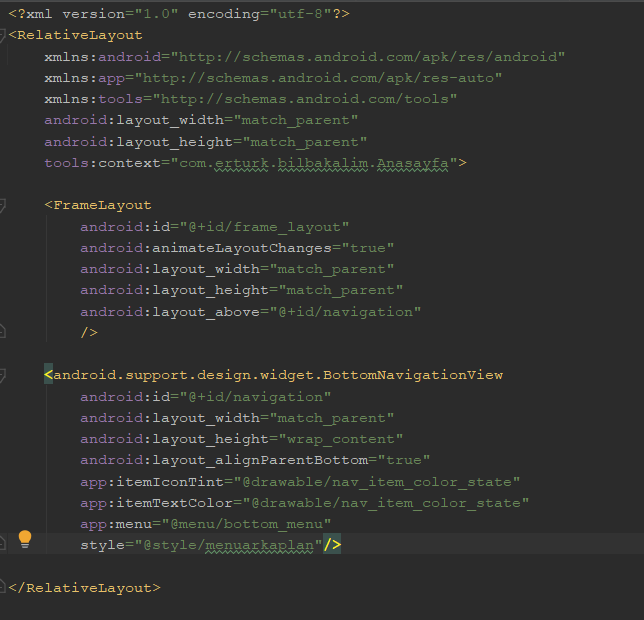
* + 1. **Kategoriler Sayfası** Giriş yaptıktan sonra “Kategoriler” sayfasına yönlendirilen kullanıcılar bu sayfa üzerinden çözmek istedikleri testin kategorisini seçmektedirler. Bu sayfada kategori görseli için 1 adet imageView, kategori adı için 1 adet textView ve RelativeLayout kullanılmıştır.

 **4.2.5 Test Başlangıç Sayfası** Kategori seçiminden sonra teste başlamadan önce kullanıcılara “Test Başlangıç Sayfası” çıkmaktadır. Bunun amacı, yanlışıkla tıklanma olayını önlemektir.Bu sayfada 1 adet background ve button kullanılmıştır.

* + 1. **Test Sayfası** Test sayfasında, kullanıcıların karşısına resimli veya yazılı sorular çıkmaktadır. Bu sayfada resimli sorular için ımageView, yazılı sorular için textView, skoru gösteren textView, kaçıncı soruda olduğunu gösteren textView ve 4 adet şıklar için button kullanılmıştır.
    2. **Sonuç Sayfası** Testin sonuna ulaşan kullanıcılara, kaç doğru cevap verdikleri ve kaç skora ulaştıklarını gösteren bir sayfa çıkmaktadır. Bu sayfada, 2 textView ve 1 adet buton kullanılmıştır.

* + 1. **Sıralama Sayfası** En iyi skorlara ulaşan kullanıcıların listelendi sayfa. Bu sayfada 2 textview ve RecyclerView kullanılmıştır.

1. **Ekler**
   1. **Projenin Kaynak Kodları**Projenin kaynak kodları için github adresi:  
      <https://github.com/OmerErturk>
   2. **Xml Dosyaları**Bu projede 13 adet layout kullanılmıştır.



activtiy\_anasayfa.xml

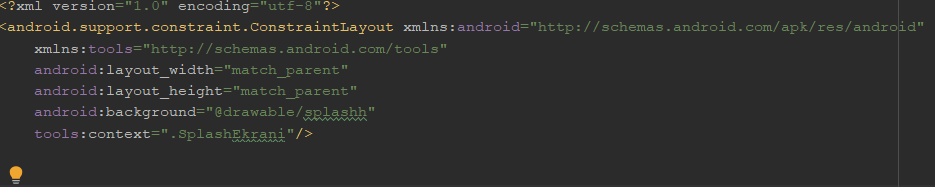

activity\_giris.xml

activity\_oyna.xml

activity\_skort\_detay.xml

****

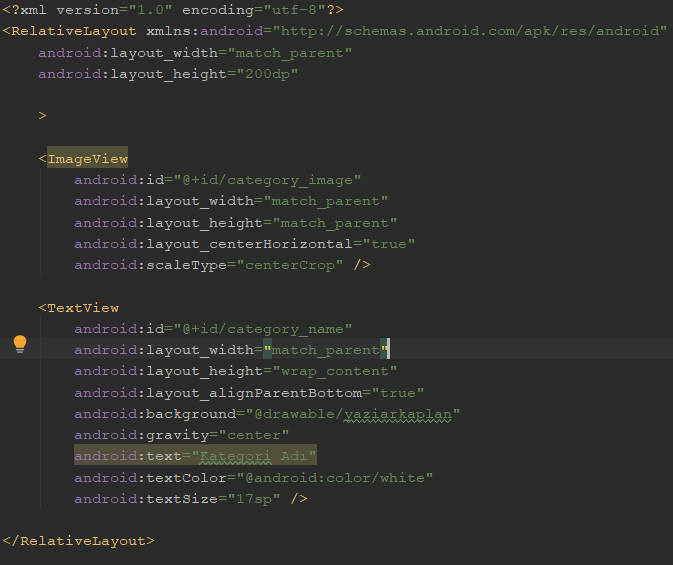
activity\_sonuc.xml



activity\_splash\_ekrani.xml



avrivity\_start.xml



Kategori\_layout.xml

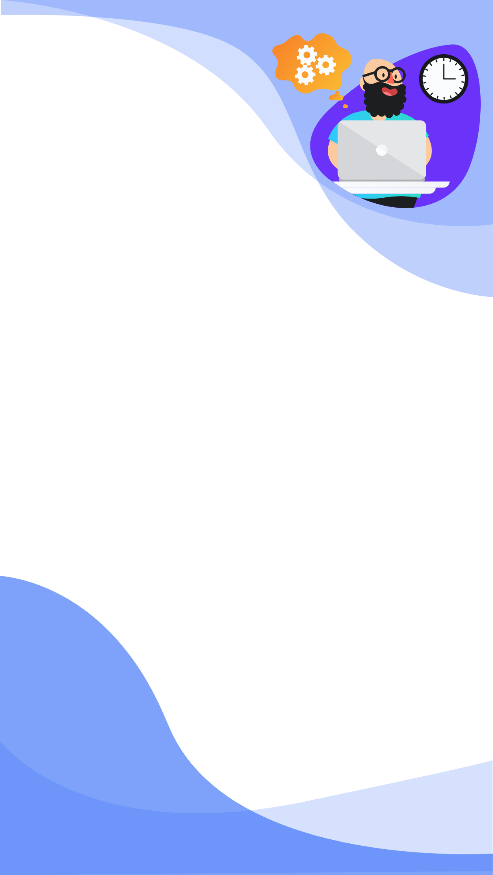


Kayit\_ol\_layout.xml



Layout\_siralama.xml

* 1. **Görsel Çalışmalar**

****

Logo ve Icon

Arkaplan Görseli

Splash Screen

Giriş sayfası arkaplan

1. **KAYNAKÇA**

<https://firebase.google.com/docs/database>

https://serifgungor.com/blogs/246-recyclerview-multiple-viewholder-kullanimi

<https://www.mediaclick.com.tr/blog/firebase-nedir>

<https://gelecegiyazanlar.turkcell.com.tr/konu/android/egitim/android-201/fragment-olusturmak>

<https://selimkaratas.com.tr/wp/javada-interface.html>

<http://erturkmedya.com/>