

**Android Uyguluma Geliştirme Dersi**

**Final Projesi**

**NCS APP**

**19MY93012**

**BATUHAN ÜNVER**

# Proje Açıklaması

Kotlin ile geliştirilmiş tamamen ücretsiz ve reklamsız,sadece **Firebase Authentication ile** **doğrulanmış** **kullanıcılar** tarafından kayıt olunup kullanılabilen bir mobil müzik uygulamasıdır.

İçindekiler

[Proje Açıklaması 1](#__RefHeading___Toc1218_1853827928)

[1-)Graddle Scripts 3](#__RefHeading___Toc1220_1853827928)

[1.a)Build.gradle (Module:app) 3](#__RefHeading___Toc55_614076770)

[2-)Splash Screen 4](#__RefHeading___Toc57_614076770)

[2.a)Styles.xml tema eklentisi 4](#__RefHeading___Toc59_614076770)

[2.b)AndroidManifet.xml 4](#__RefHeading___Toc61_614076770)

[2.c)SplashScreenActivity 4](#__RefHeading___Toc63_614076770)

[3-)Uygulama Ana Ekranı 5](#__RefHeading___Toc65_614076770)

[3.a)Ana Ekran Tasarımı 5](#__RefHeading___Toc67_614076770)

[3.b)Ana Ekran Kaynak Kodları 6](#__RefHeading___Toc69_614076770)

[4-)Uygulamaya Kayıt Olma 7](#__RefHeading___Toc71_614076770)

[4.a)Signup Sınıfı Kaynak Kodları 7](#__RefHeading___Toc73_614076770)

[4.b)SignUp Kullanıcı Kayıt Fonksiyonu Ve Kayıt Button onClick 8](#__RefHeading___Toc75_614076770)

[4.c)Firebase Console Kayıtlı Kullanıcılar 9](#__RefHeading___Toc77_614076770)

[5-)Uygulamaya Giriş 10](#__RefHeading___Toc83_614076770)

[5.a)SingIn Sınıfı Kaynak Kodları 10](#__RefHeading___Toc85_614076770)

[5.b)SingIn Kullanıcı Giriş Fonksiyonu Ve Giriş Butonu Onclick 11](#__RefHeading___Toc87_614076770)

[5-)Parolanızı Mı Unuttunuz? 12](#__RefHeading___Toc89_614076770)

[5.a)Parola Sıfırlama Kaynak Kodları 13](#__RefHeading___Toc91_614076770)

[6-)Kullanıcı LogOut 15](#__RefHeading___Toc93_614076770)

[7-)MediaPlayer Ve Seekbar 16](#__RefHeading___Toc95_614076770)

[7.a)MediaPlayer Ve SeekBar Kaynak Kodları 16](#__RefHeading___Toc97_614076770)

[8-)Push Notification(FCM) 20](#__RefHeading___Toc99_614076770)

[8.a)Manifest Dosyasına Servisi Eklemek 20](#__RefHeading___Toc101_614076770)

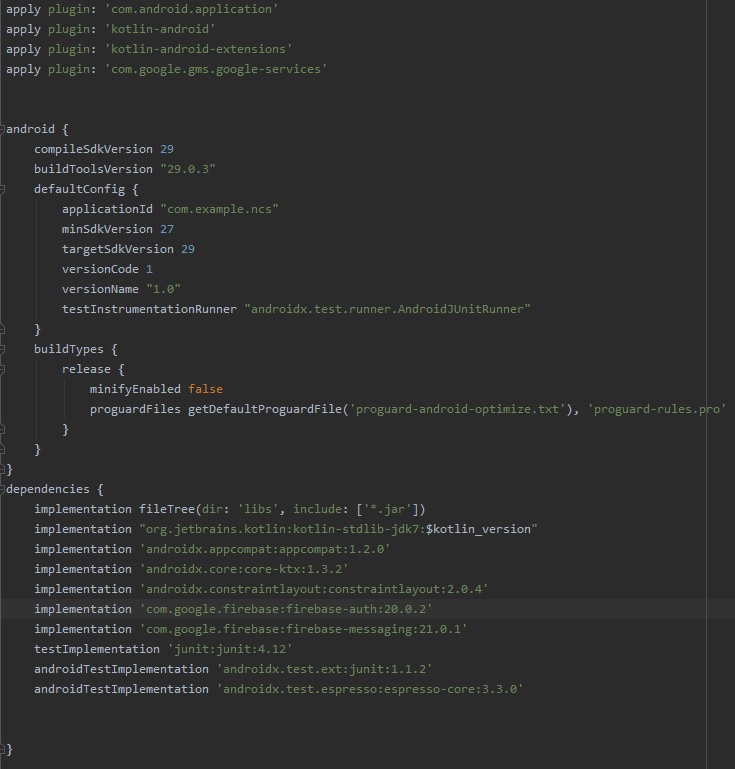
[8.b)PushNotification Class 21](#__RefHeading___Toc103_614076770)

[8.c)Push Notification Test 22](#__RefHeading___Toc105_614076770)

# 1-)Graddle Scripts

## 1.a)Build.gradle (Module:app)

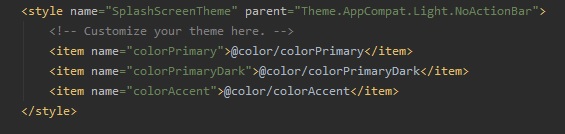
Geliştirmiş olduğumuz uygulamamız da kullanıcıları kayıt edebilmek için ve giriş yapabilmeleri için graddle’ımızın içerisine **Firebase User Authentication ve Push Notification(Cloud Messaging)** için gerekli servisleri ve paketleri ekledik.



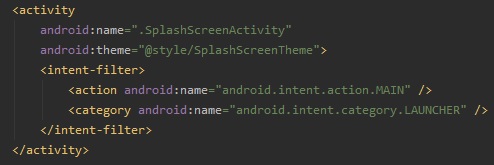
# 2-)Splash Screen

Uygulamamız açılırken ekrana gelicek olan Splash Ekranı için gerekli ayarlamalar.

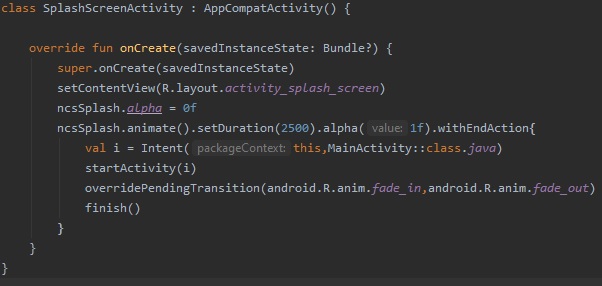
## 2.a)Styles.xml tema eklentisi



## 2.b)AndroidManifet.xml



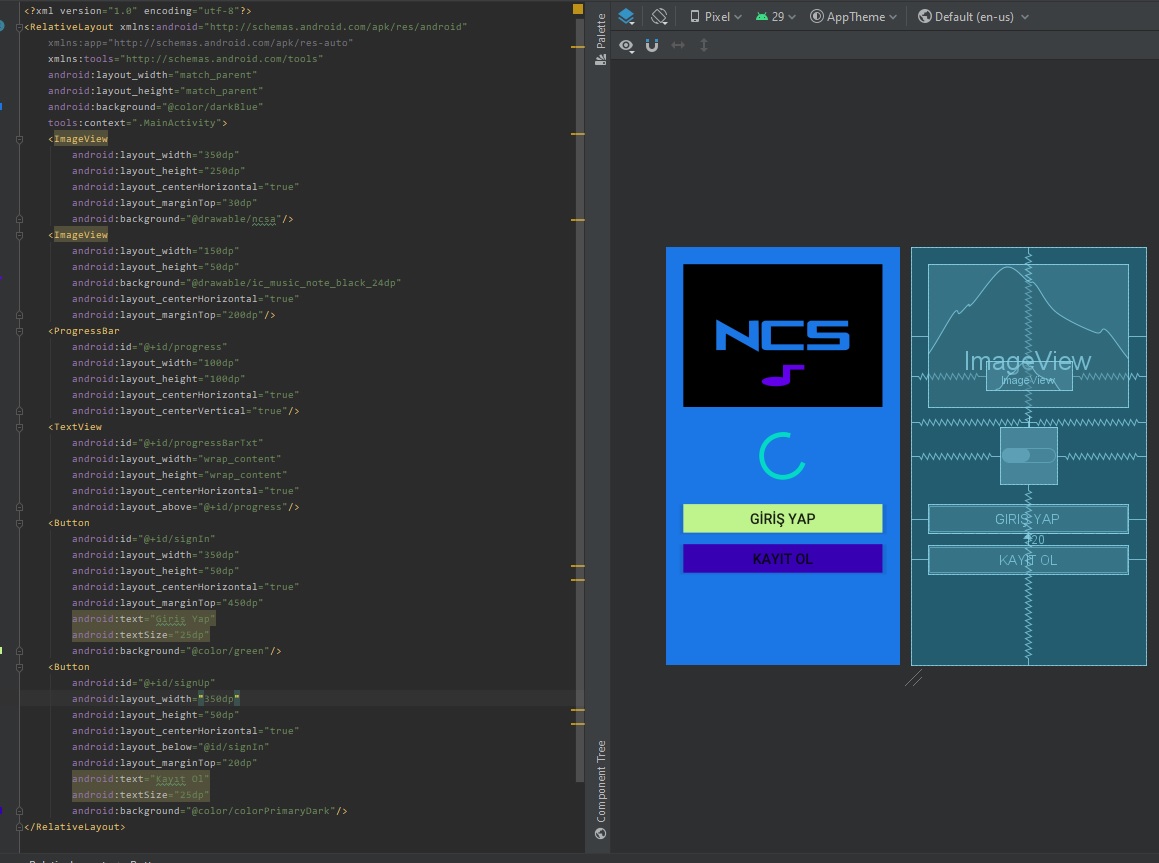
## 2.c)SplashScreenActivity



Yeni bir kotlin class’ı oluşturulup Splash’ın animasyonun dan sonra Ana ekrana otomatik gitmesi için Intent nesnesi oluşlurulmuştur.

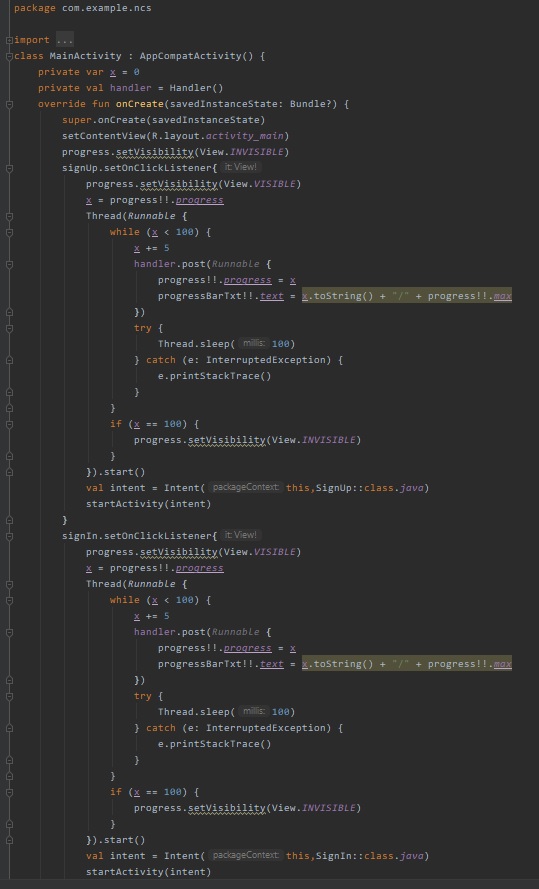
# 3-)Uygulama Ana Ekranı

## 3.a)Ana Ekran Tasarımı



Ana ekran tasarımın da kullanıcı kayıt ve giriş sayfalarına yönlendirme butonları ve progressbar mevcuttur.

## 3.b)Ana Ekran Kaynak Kodları



Ana ekran da bulunan 2 adet Button için de hem **kayıt sayfası** hem **giriş sayfası** için yönlendirme yapılması için **Intent** nesneleri oluşturulmuştur

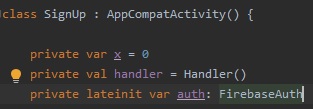
Sayfa yüklenir yüklenmez **progressbar invisible** olarakgörünmeyecek şekilde ayarlanmıştır,butonlara basıldığı takdirde aktif olacaklardır.

# 4-)Uygulamaya Kayıt Olma

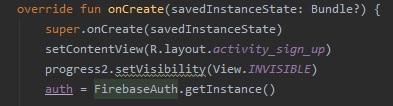
## 4.a)Signup Sınıfı Kaynak Kodları



Firebase kimlik doğrulamasını kullanabilmek için projemize **import** ediyoruz.

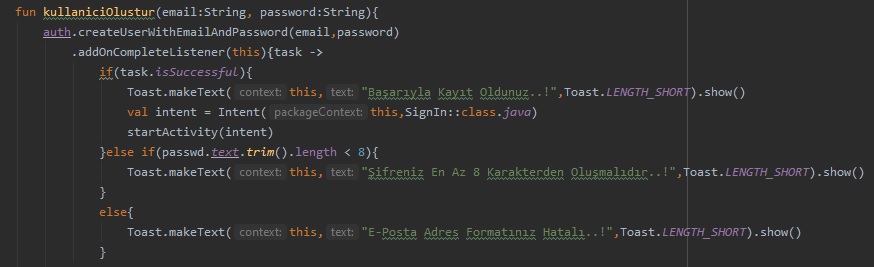


SignUp sınıfımızın içerisine değişkenlerimizi yazıyoruz, **auth** değikenine FirebaseAuth’ı atıyoruz.



**OnCreate** metodumuzun içerisine **FirebaseAuth** sınıfının nesnelerini kullanabilmek için **getInsatnce()** kullanıyoruz.

## 4.b)SignUp Kullanıcı Kayıt Fonksiyonu Ve Kayıt Button onClick

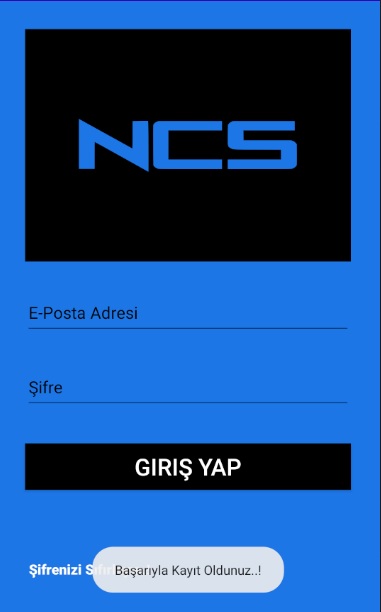
****

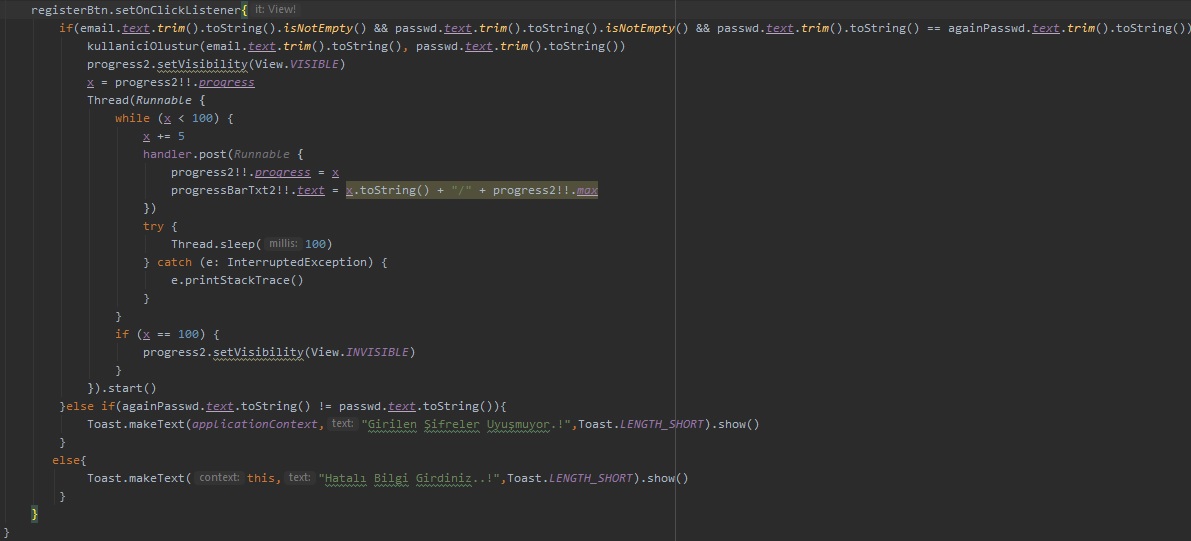
**KullaniciOlustur** fonksiyonumuzun parametreleri, email ve şifre ile kayıt olacağımız için. **(email:String, password:String)** olarak tanımlanmıştır.

**Auth**. İle kullanıcı adı ve **parola ile kullanıcı oluştur** firebase nesnemizi çağırıyoruz.**task’ın** koşullarının sağlanıp sağlanmadığını if else blokları ile kontrol ediliyor.

**Fireabase**  kullanıcılarından en az 8 karakterli şifre istediğinden, Şifre 8 karakterden az olur ise koşul ile kontrol edildiğinden kullanıcılar **Toast** mesajı ile hata alacaklardır.

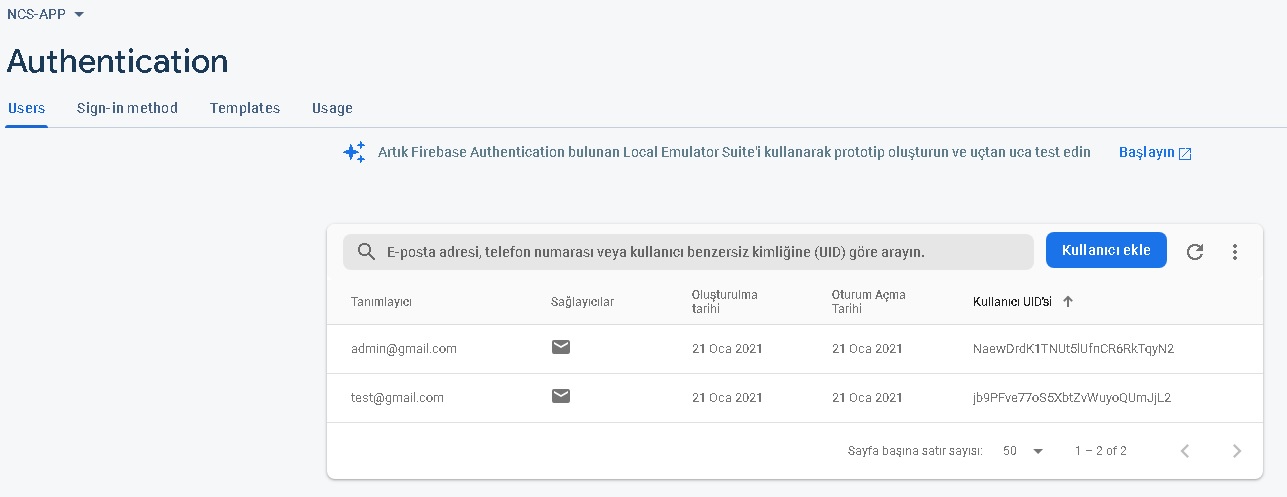
Eğer kayıt başarılı olur ise **Toast** mesajı ile beraber SignIn(Giriş Ekranı)’na yönlendiriliyor.

****



**registerBtn** yani kayıt butonumuzun **onClick**(Butona Bastığımız da)içerisin de **progressbar** görünümü ve **EditText’lerin** kayıt olmaya uygun olup olmadığını ve **kullaniciOlustur()** fonksiyonumuzu çağırıp koşullar sağlandığı takdirde kayıt olup olunamayacağı kontrol ediliyor.

## 4.c)Firebase Console Kayıtlı Kullanıcılar

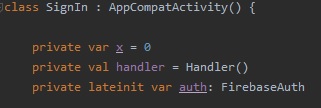
****

# 5-)Uygulamaya Giriş

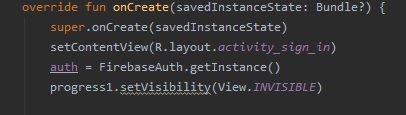
## 5.a)SingIn Sınıfı Kaynak Kodları

****

Firebase kimlik doğrulamasını kullanabilmek için projemize **import** ediyoruz.

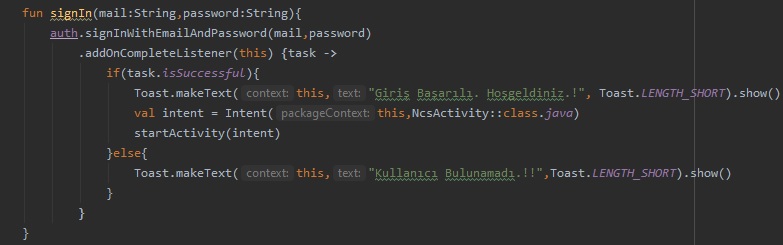
****

**SignIn** sınıfımızın içerisine değişkenlerimizi yazıyoruz, **auth** değikenine FirebaseAuth’ı atıyoruz.

****

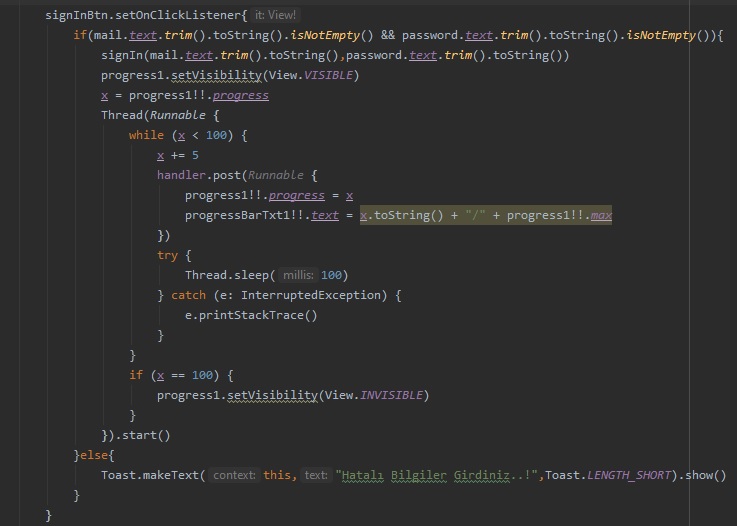
**OnCreate** metodumuzun içerisine **FirebaseAuth** sınıfının nesnelerini kullanabilmek için **getInsatnce**() kullanıyoruz.

## 5.b)SingIn Kullanıcı Giriş Fonksiyonu Ve Giriş Butonu Onclick

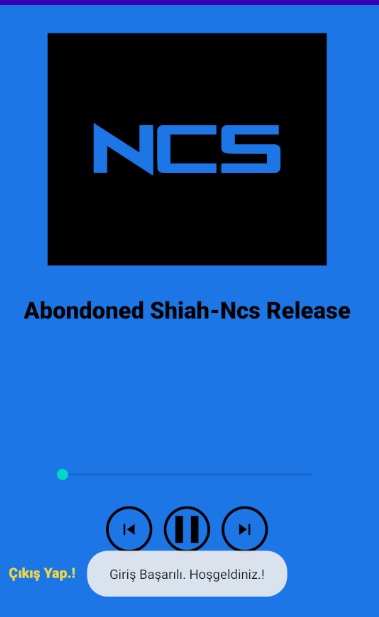
****

Giriş fonksiyonumuza parametlerimizi verdikten sonra firebase **signInWithEmailAndPassword()**’ parametreleri giriyoruz.

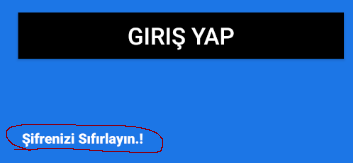
**.addOnCompleteListener** task’ımızın içinde ki koşullar sağlandığı takdir de **Intent** ile Müzik aktivitemize geçiş yapabiliyoruz.**Toast** mesajı ile beraber.

****

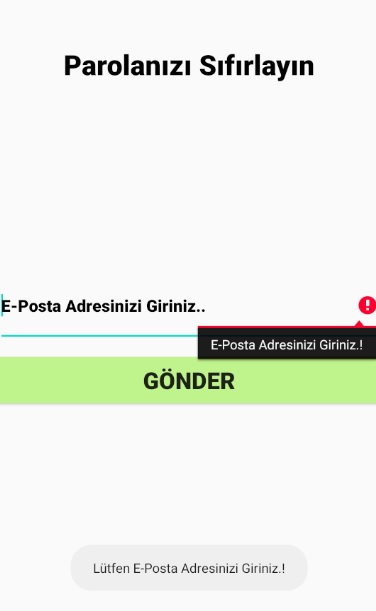
**signInBtn** yani **giriş** butonumuzun onClick(Butona Bastığımız da) içerisin de **progressbar** görünümü ve **EditText’lerin** giriş yapılıp yapılamayacağına ve **singIn() fonksiyonumuzu** çağırıp koşullar sağlandığı takdirde giriş yapılıp yapılamayacağı kontrol ediliyor.

****

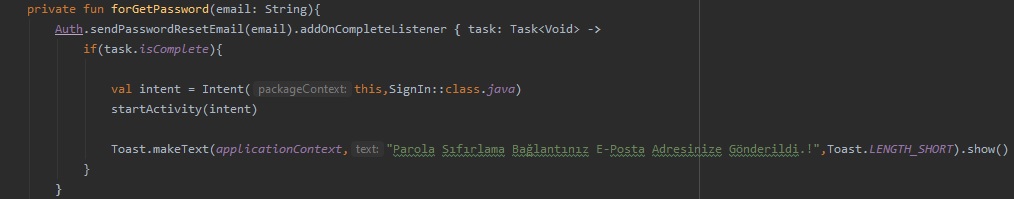
# 5-)Parolanızı Mı Unuttunuz?



Uygulamaya giriş esnasın da parolanızını unuttuğunuz an da tıklayıp sizi **Parola** **Sıfırlama** **sayfasına** yönlendiren buton.

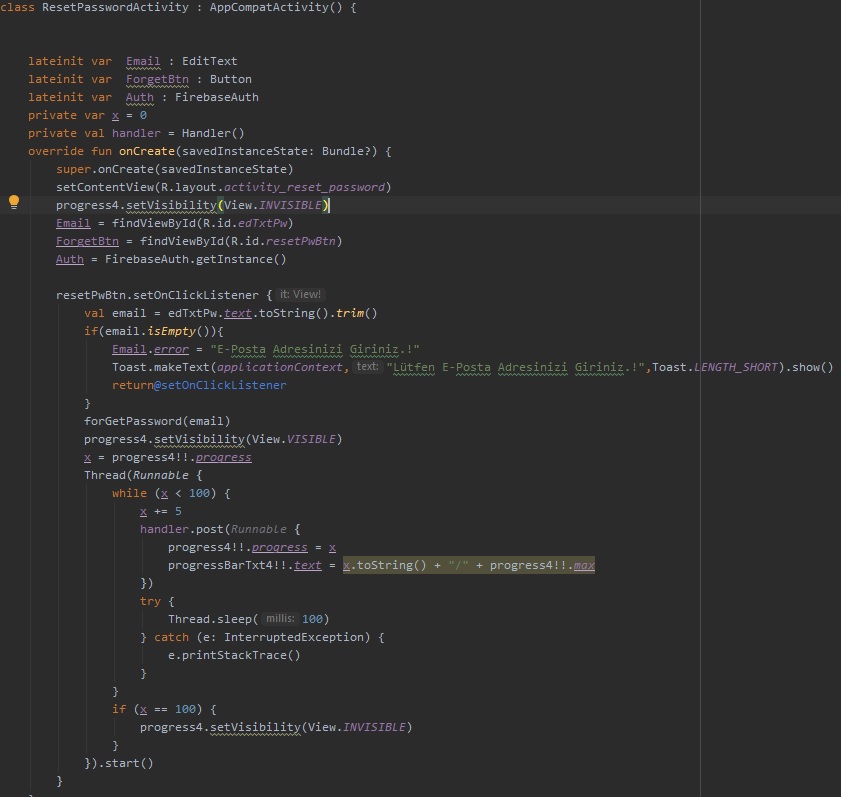


## 5.a)Parola Sıfırlama Kaynak Kodları



Unutulan şifre isimli **fonksiyonumuzun** **parametresi** ‘email’ **FireBase’in** .**sendPasswordResetEmail** Completelistener task oluşturup, eğer tamamlanırsa **Intent** ile SignIn’ login sayfamıza **Toast** mesajı ile yönlendirileceksiniz.





**OnCreate metodumuz** içerisin de

**email**  değişkenimize EditText atanmıştır.

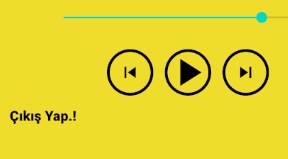
**ForgetBtn** değişkenimize Parola sıfırlama butonumuz atanmıştır.

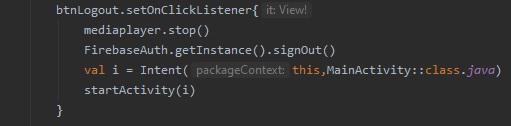
Auth ‘ FirebaseAuth.getInstance() ile paralo sıfırlama kullanabilmek için atanmıştır= **sendPasswordResetEmail()**

**resetPwBtn onClick** metodumuz da koşullar kontrol edilip,

**forGetPassword(email) fonksiyonumuz** çağırılmıştır.

# 6-)Kullanıcı LogOut





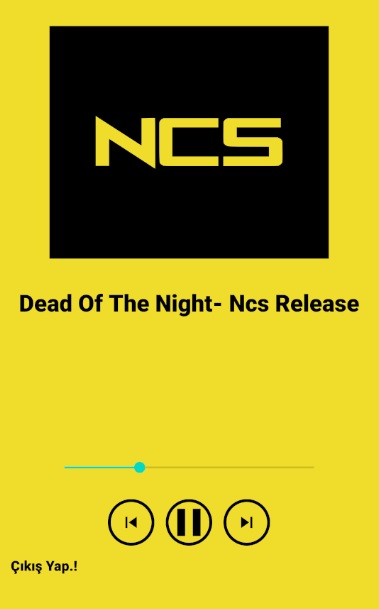
Kullanıcı **Çıkış Yap.!** Butonuna tıkladığı takdir de **OnClick metodu** içerisin de,

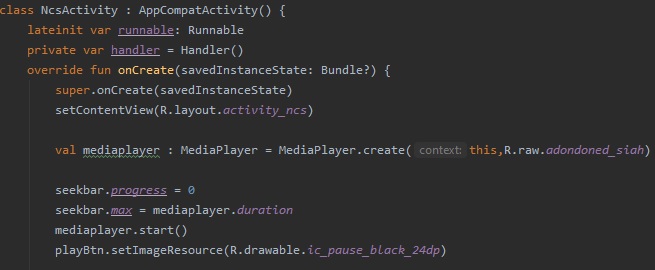
**FirebaseAuth.getInstance().signOut()** metodu ile çıkış yapar ve Ana ekrana(**MainActivity’e**) **Intent** ile yönlendirilir.

# 7-)MediaPlayer Ve Seekbar

Müzik çalar uygulamamız da **MediaPlayer** kütüphanesi kullanılıp **SeekBar** eklenmiştir.

## 7.a)MediaPlayer Ve SeekBar Kaynak Kodları

****

****

**NcsActivity** class’ımızın içerisine runnable, handler değişkenleri eklenmiştir.

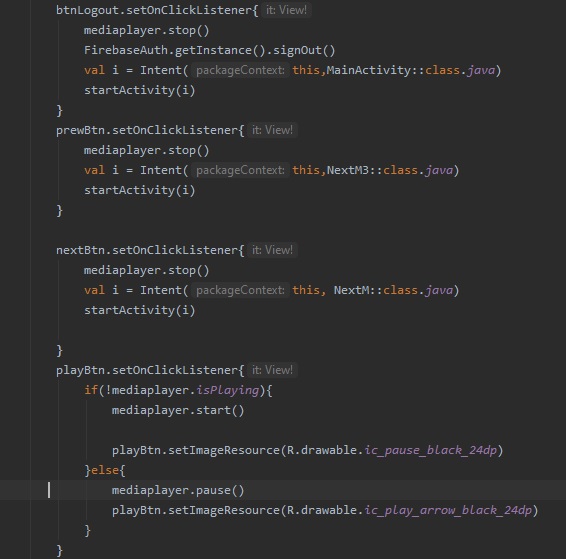
**OnCreate** metodumuzun içerisin de ise **MediaPlayer’ı** tanımlıyoruz, Önceden **“raw”** isimli **Directory’**imizin içerisine atmış olduğumuz mp3 lerden **create** mediaplayer(this,R.raw.mp3name) metodu ile o müziği **mediaplayer**  değişkeni içerisine atıyoruz.

Seekbar.progress = 0 ‘dan başlayıp

seekbar.max = mediaplayer.duration yani Müziğin uzunluğu kadar.

Mediaplayer.start() metodu ile müzik başlar.

Müzik çalmaya başlar başlamaz **playBtn** setImageResource ile Pause simgesi olarak değişir.

****

Oynat,next,prew butonları Sonraki müzik önceki müzik durdur oynat görevi görüyorlar.

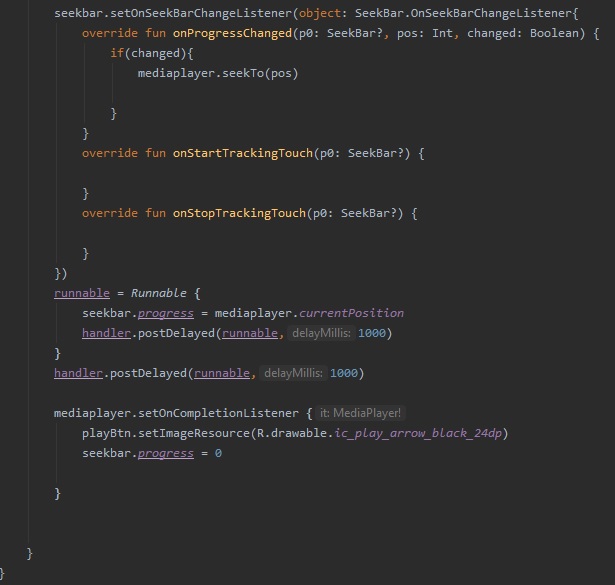
**PrewBtn** basildigin da müzik durur bir önceki müziğe geçer.

**NextBtn** basildiğin da müzik durur bir sonrakine geçer.

**PlayBtn** koşullar ile eğer müzik aktif değilse müziği çalmaya başlar, play iconunu pause olarak set eder.

Sonraki koşul ise müziği durdurup pause icon’u play icon olarak set eder.

Yani aynı tuşta durdurma ve oynatma özelliği kazanır.

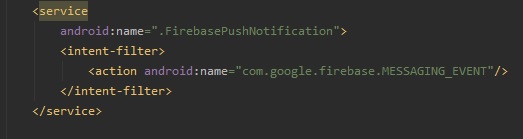
****

# 8-)Push Notification(FCM)

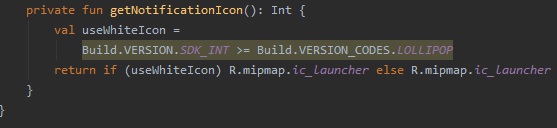
## 8.a)Manifest Dosyasına Servisi Eklemek

**Firebase’in** Push Notification özelliğini Kullanabilmek için **AndroidManifest.xml** dosyasına eklenmesi gereken **permissions** ve **servisler.**

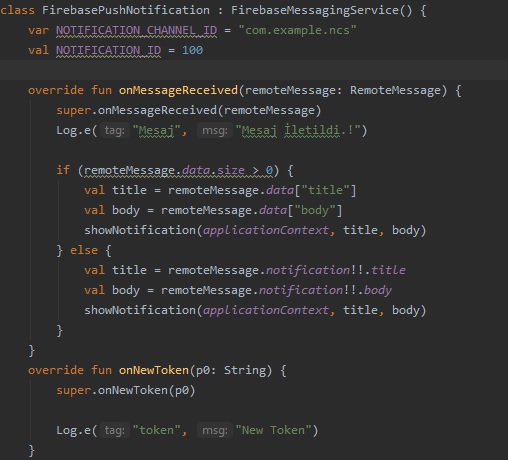
****

****

## 8.b)PushNotification Class



Uzaktan bildirim aldığımız da bildirim ekranın da çıkacak olan Icon Fonksiyonumuz.



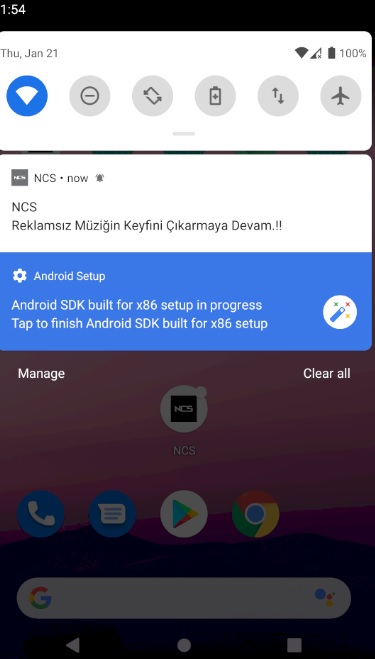
Class’ın içerisine Channeld ıd ye **package name(**paket adımızı) yazıp

id değeri 100 olarak atanmıştır.

Koşullar ile notification(bildirimin) ekranda gözükmesi sağlanmıştır.

## 8.c)Push Notification Test

Firebase Console’dan Telefonumuza bir **Notification** gönderelim.



Eğer bildirime tıklarsak uygulamamız açılacaktır.