metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

metin, ekran görüntüsü, ekran, dizüstü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Giriş ekranımızda;

Karşılama cümlemiz için Yazı Özellikli Text,

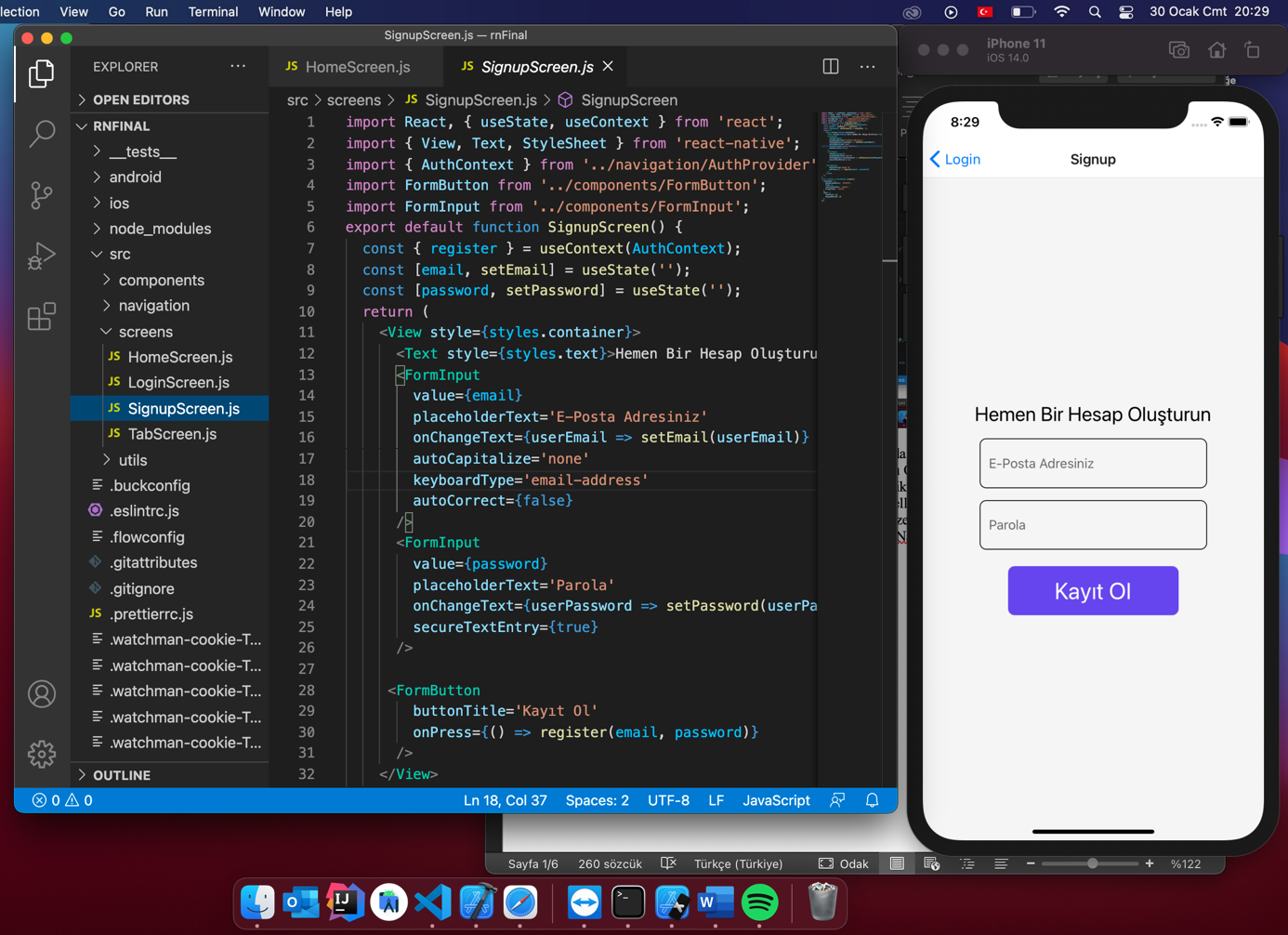
E-Mail için Mail Özellikli Text,

Password için Parola Özellikli Text,

Buton için Action Method Özellikli Button,

Kayıt Olmak için, Text Özellikli NavButtonText’ten

yararlanıldı.



Kayıt Ekranımızda;

E-Mail için Mail Özellikli Text,

Password için Parola Özellikli Text,

Kayıt Ol Butonu için Action hareketli Button

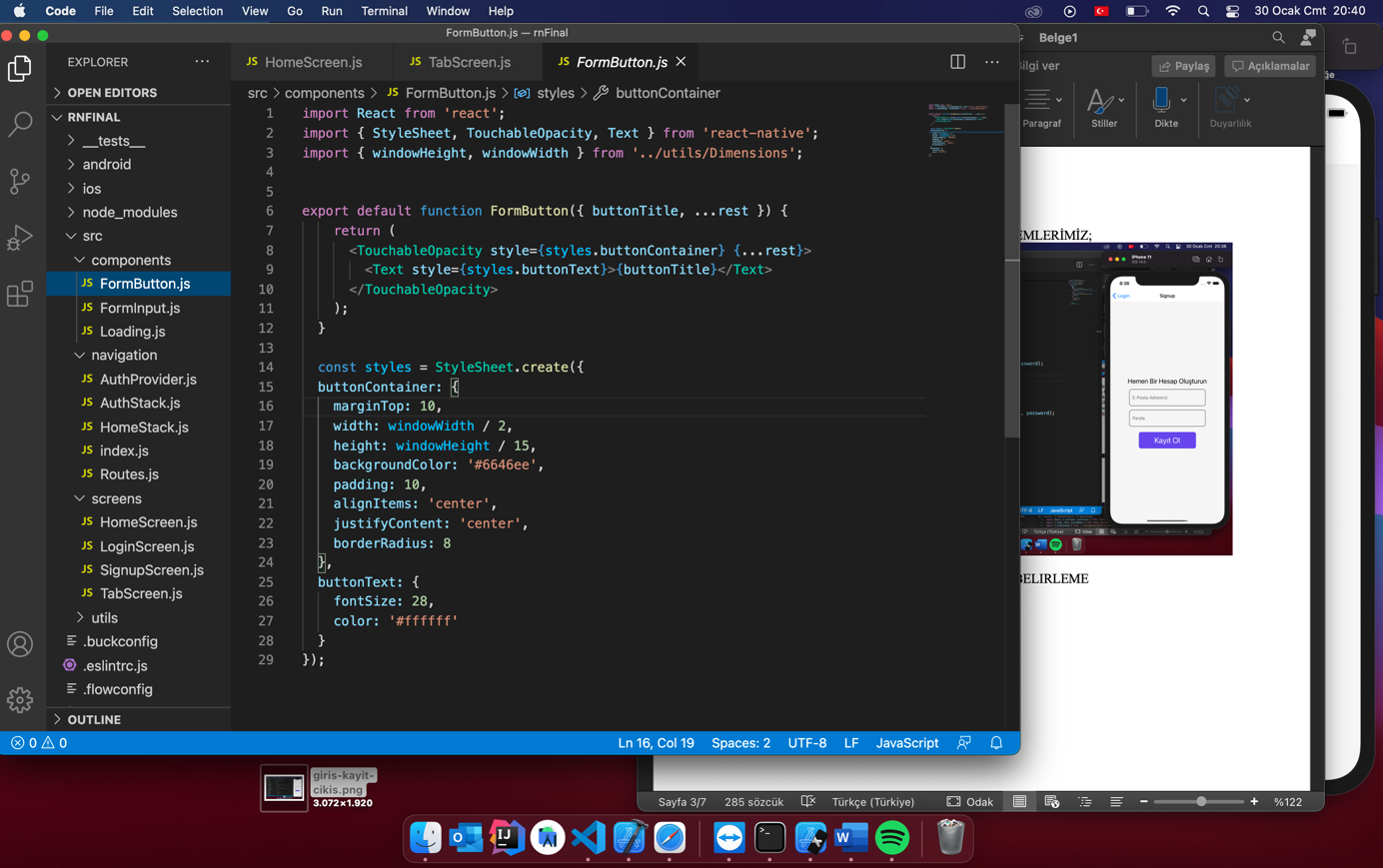
Kullandım.

GİRİŞ-KAYIT-ÇIKIŞ İŞLEMLERİMİZ;

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

BUTTON TASARIM





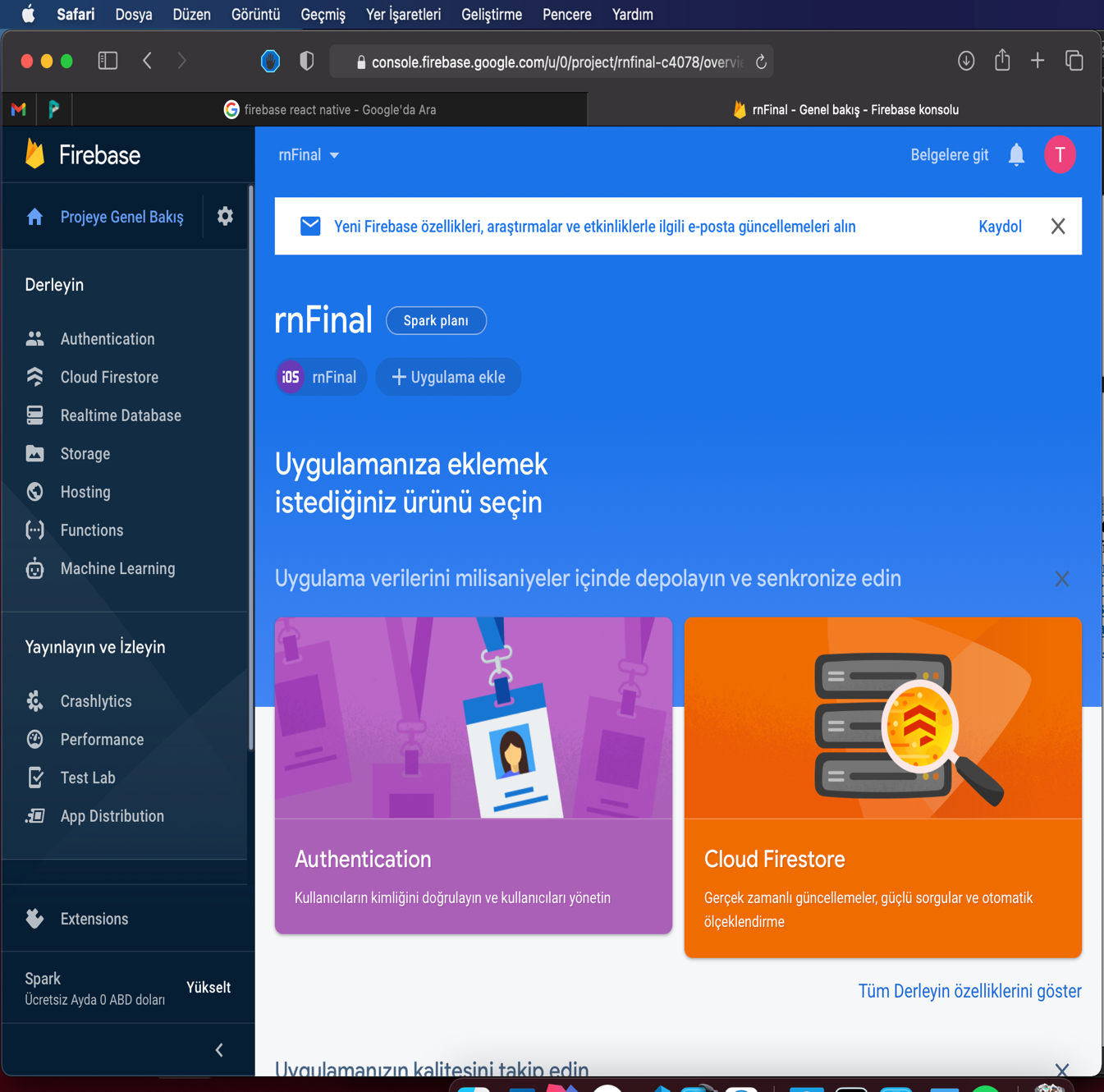
FIREBASE KULLANIMI

İlk önce hesap oluşturup, projemizi ekliyoruz.

https://console.firebase.google.com/u/0/

metin içeren bir resim

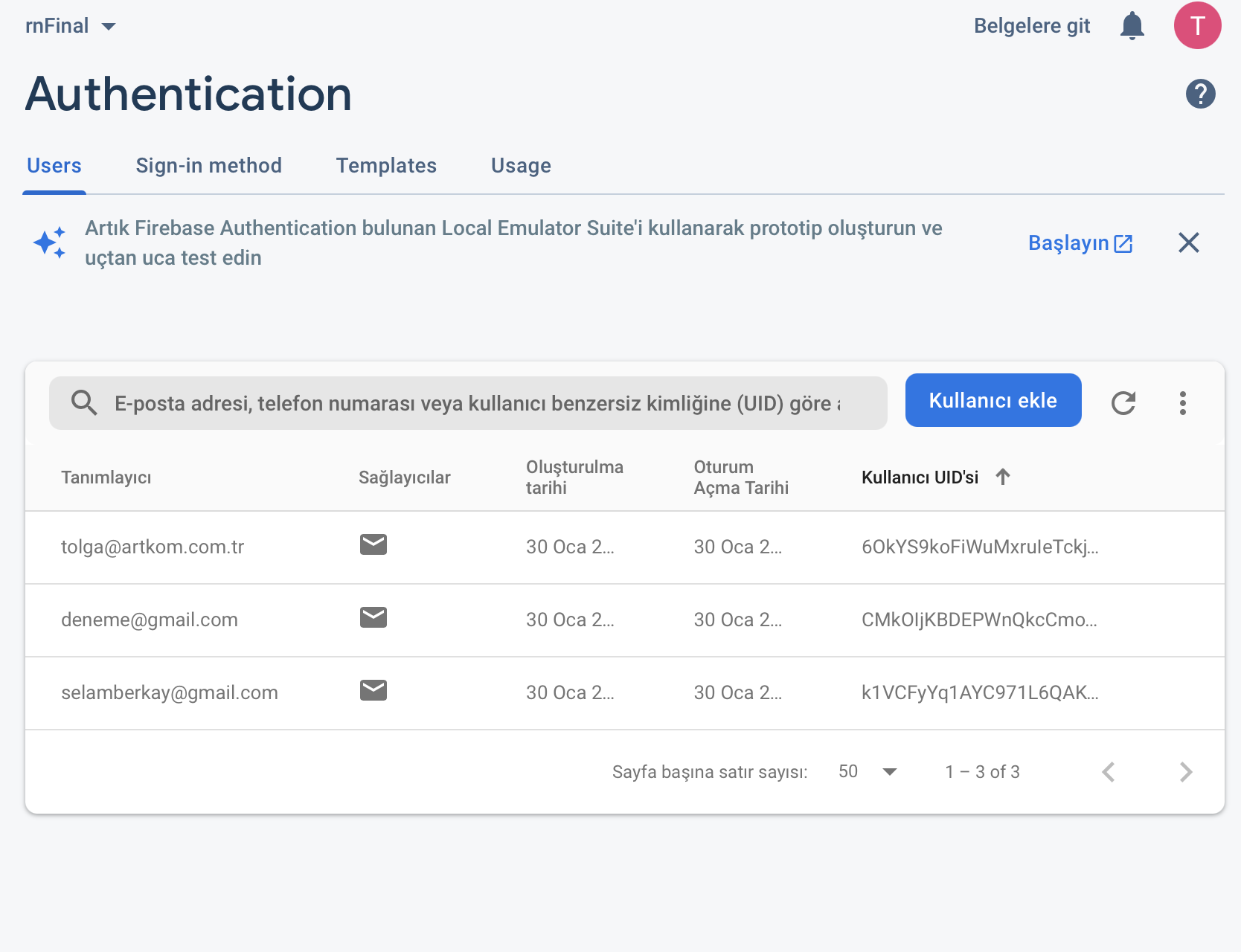
Açıklama otomatik olarak oluşturuldu

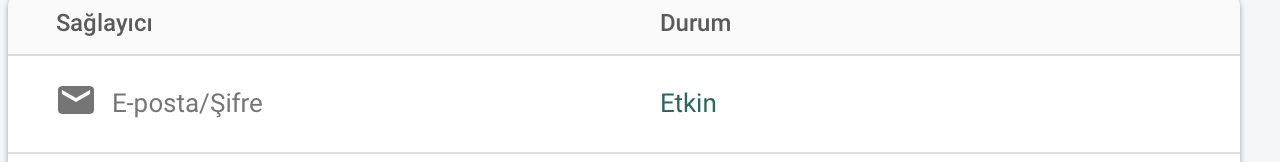


metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

E-Mail entegreasyonunu açmak için proje ayarlarına girelim. Ardından İzinler Menüsü altından giriş methodları ve mail-password aktifleştirelim.





NPM ILE KURULUM YAPTIĞIMIZ İÇİN;

npm install --save @react-native-firebase/app

İndirdiğimiz plist dosyasımızı rnFinal(proje adı) klasörümüze atıyoruz. Ardından xCODE IDE mizi açıp proje adına sağ tıklayıp add flies to “.plist dosyasını” import ediyoruz.

AppDelegate.m Dosyamıza import ediyoruz.

#import <Firebase.h>

Bool applicationumuza ekliyoruz;

if ([FIRApp defaultApp] == nil) {

[FIRApp configure];

}

Building işlemi için;

pod install --repo-update

cd ..

npx react-native run-ios

PUSH-NOTIFICATION İÇİN

Projemizi en başından “NPM” ile oluşturduğumuz için kütüphanemizi “NPM” kullanarak Install ediyoruz.

**Push Natification Kütüphanesi için;**

Npm add @react-native-community/push-notification-ios

**React Native v0.60+ için;**

npx pod-install

iOS’a entegre etmek için;

Ben kendi proje ismim olan rnFinal ile anlatacağım, xCode IDE’mize projemizin içerisinde iOS klasörünü import ediyoruz. Sonrasında ise rnFinal.scworkspace workspace dosyasını açıyoruz. Projenin üzerinden “Signing & Capabilities” e tıklayıp sağındaki + butonundan Backgraound Mode ve Remote Notifications modüllerini ekliyoruz.



Ardından Push Notification’u projemize import etmemiz gerekiyor.

Projemizi iOS sekmesi altındaki AppDelegate.h dosyamızın içerisine girerek;

#import <UserNotifications/UNUserNotificationCenter.h>

Sayfamızın ortasına( UNUserNotificationCenterDelegate e gelerek) protokolümüzü ekliyoruz

@interface AppDelegate : UIResponder <UIApplicationDelegate, RCTBridgeDelegate, UNUserNotificationCenterDelegate>

Diğer .m dosyamız olan AppDelegate.m dosyamıza girerek PushNotificationu import ediyoruz.

#import <UserNotifications/UserNotifications.h>

#import <RNCPushNotificationIOS.h>

Ardından methodlarımızı ekliyoruz

// Kayıt için gerekli methodumuz

- (void)application:(UIApplication \*)application didRegisterForRemoteNotificationsWithDeviceToken:(NSData \*)deviceToken

{

[RNCPushNotificationIOS didRegisterForRemoteNotificationsWithDeviceToken:deviceToken];

}

// Uzaktan bildirim için sağlanacak method/fonksiyonumuzu ekliyoruz

- (void)application:(UIApplication \*)application didReceiveRemoteNotification:(NSDictionary \*)userInfo

fetchCompletionHandler:(void (^)(UIBackgroundFetchResult))completionHandler

{

[RNCPushNotificationIOS didReceiveRemoteNotification:userInfo fetchCompletionHandler:completionHandler];

}

// Hata aldığında uyarıları verecek Methodumuz.

- (void)application:(UIApplication \*)application didFailToRegisterForRemoteNotificationsWithError:(NSError \*)error

{

[RNCPushNotificationIOS didFailToRegisterForRemoteNotificationsWithError:error];

}

// Giriş yapıldığındaki bildirim methodumuz

- (void)userNotificationCenter:(UNUserNotificationCenter \*)center

didReceiveNotificationResponse:(UNNotificationResponse \*)response

withCompletionHandler:(void (^)(void))completionHandler

{

[RNCPushNotificationIOS didReceiveNotificationResponse:response];

}

Ardından AppDelegate impelementasyonumuzu takip ederek kodlarımızı ekliyoruz.

- (BOOL)application:(UIApplication \*)application didFinishLaunchingWithOptions:(NSDictionary \*)launchOptions

{

...

// Çağralım, UNUserNotificationCenter

UNUserNotificationCenter \*center = [UNUserNotificationCenter currentNotificationCenter];

center.delegate = self;

return YES;

}

//Bildirim methodumuzu buraya tekrardan çağırıyoruz.

-(void)userNotificationCenter:(UNUserNotificationCenter \*)center willPresentNotification:(UNNotification \*)notification withCompletionHandler:(void (^)(UNNotificationPresentationOptions options))completionHandler

{

completionHandler(UNNotificationPresentationOptionSound | UNNotificationPresentationOptionAlert | UNNotificationPresentationOptionBadge);

}

metin, elektronik eşyalar, siyah içeren bir resim

Açıklama otomatik olarak oluşturuldu

<https://github.com/react-native-push-notification-ios/push-notification-ios>

https://www.pradipdebnath.com/2020/10/04/how-to-implement-email-authentication-in-react-native-using-firebase/

<http://www.ahmetbaltaci.com/react-native/react-native-uygulama-isim-icon-splash-screen-islemleri/>

<https://devnot.com/2019/react-native-ve-hooks-ile-haber-uygulamasi-yapimi-bolum-2/>

<https://newsapi.org>

<https://console.firebase.google.com/u/0/>