**ISPARTA UYGULAMALI BİLİMLER ÜNİVERSİTESİ**

**TEKNİK BİLİMLER MYO**

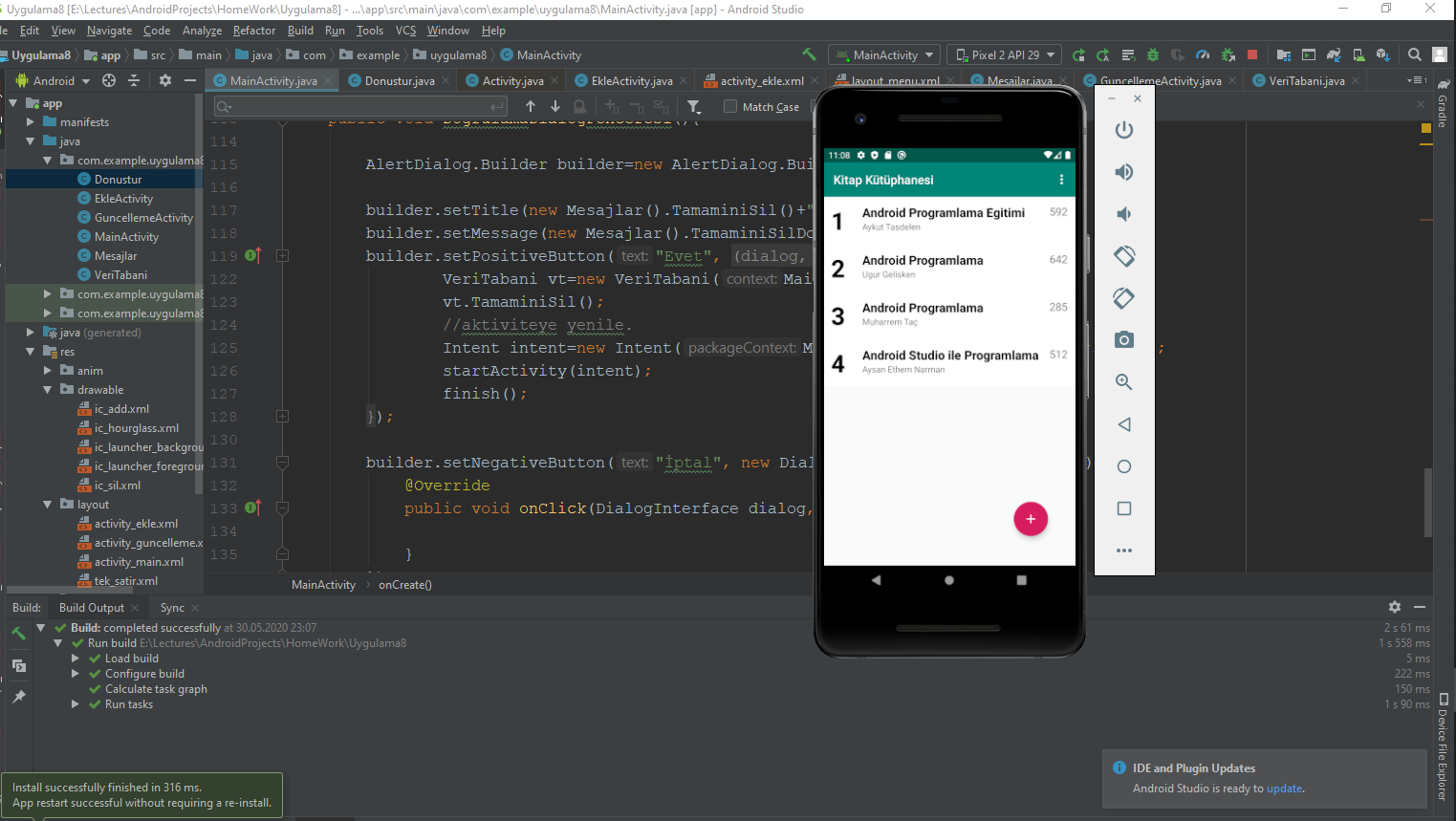
**MOBİL PROGRAMLAMA VİZE SINAV SORULARI**

**Not: Aşağıdaki notlara dikkat etmeniz beklenmektedir. Beklenenler;**

* **Uygulamayı ayrı bir android projesi olarak açıp kodlama ve tasarım yapmanız beklenmektedir.**
* **Her uygulamanın xml, java ve tasarım çıktılarını word dosyalasına yapıştırınız.**
* **Başkasına ya da farklı yollara giderek ödev yaptırmayınız.**
* **Çalışan Ekran Görüntülerini, print screen yaparak tüm ekranın Word dosyasına eklenmesi istenmektedir. Aksi takdirde kopya çekildiği düşünülecektir.**
* **1 farklı uygulama klasörü ve 1 farklı Word dosyasını son tarihten önce adınızı ve soyadınızı ve öğrenci numaranızı içeren bir isimle isimlendirdiğiniz bir cevap klasöründe toplayıp winrar ile sıkıştıktan sonra obs sistemine yükleyiniz.**

**UYARI: Çalışan Ekran Görüntüsünü aşağıdaki şekilde olduğu gibi yukarıda dosya yolu, kırmızı çalışma butonu, alt tarafta build işleminde hata olmaması, ekranın tam görünmesi istenmektedir. Bu görüntü Word dosyası içerisinde verilmediği takdirde kopya şüphesi ile hareket edilecektir. Aşağıdaki görüntüyü uygulamanızın her çalışan adımını Word dosyası içerisinde paylaşmanız sınavda iyi bir not alabilmeniz için önemlidir. Bu görüntülerin yukarıdan yolu kesmeleri durumunda görüntüyü paylaşmanızın hiçbir önemi yoktur. Puan olarak 50 puana kadar etkiye sahip olacaktır.**

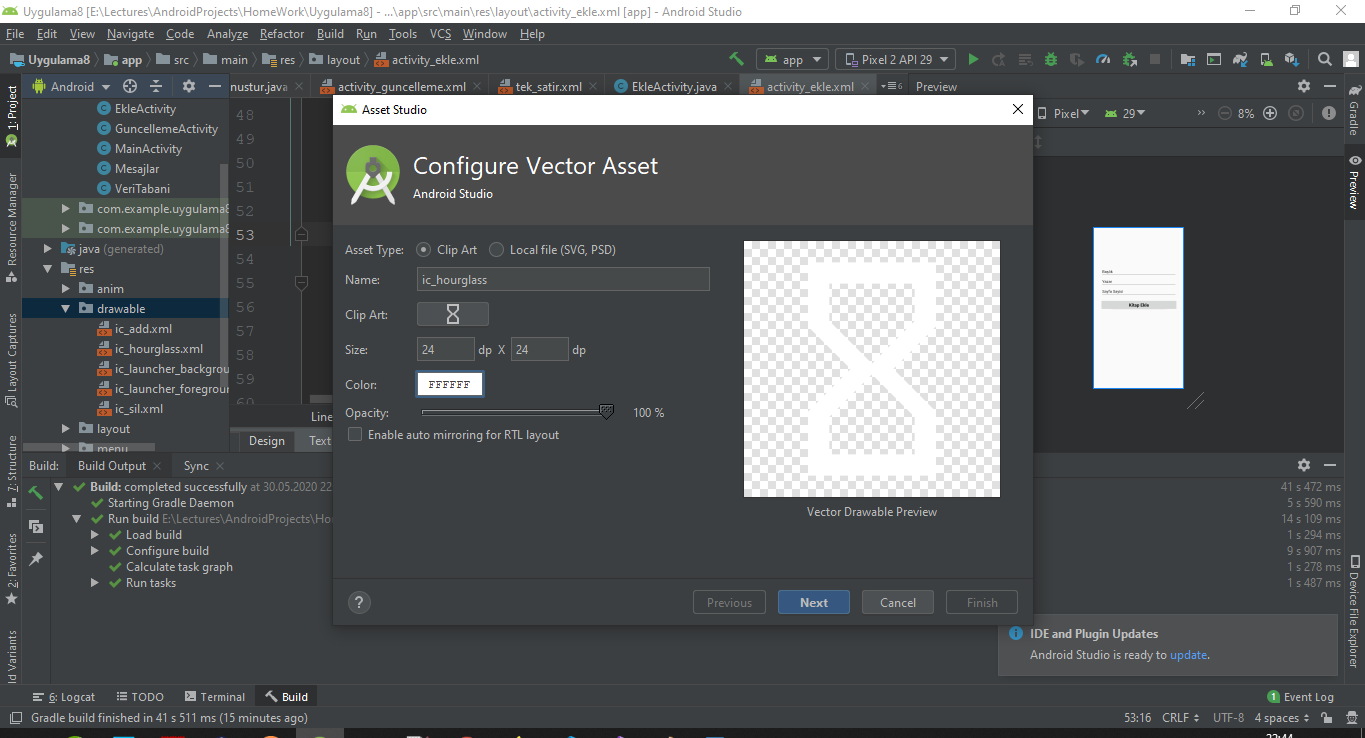
**Şekil 1.’de verildiği gibi ne kadar fazla UYARI’da belirttiğim şekilde 20 puan verilecektir. En az belirtilen uyarılara göre 6 ekran görüntüsü verilmesi beklenmektedir. UYARI’DA BELİRTİLEN KURALA UYMAYAN GÖRÜNTÜLERİN HİÇBİR ÖNEMİ YOKTUR. ÖDEVLERİ GRUP HALİNDE DEĞİL, HER ÖĞRENCİNİN KENDİ YAPMASI BEKLENMEKTEDİR. GÖRÜNTÜLERİ PAYLAŞMAYAN KOD BENZERLİĞİ YÜKSEK ÖDEVLERDE 50 PUANA KADAR DÜŞÜRÜLEBİLİR.**



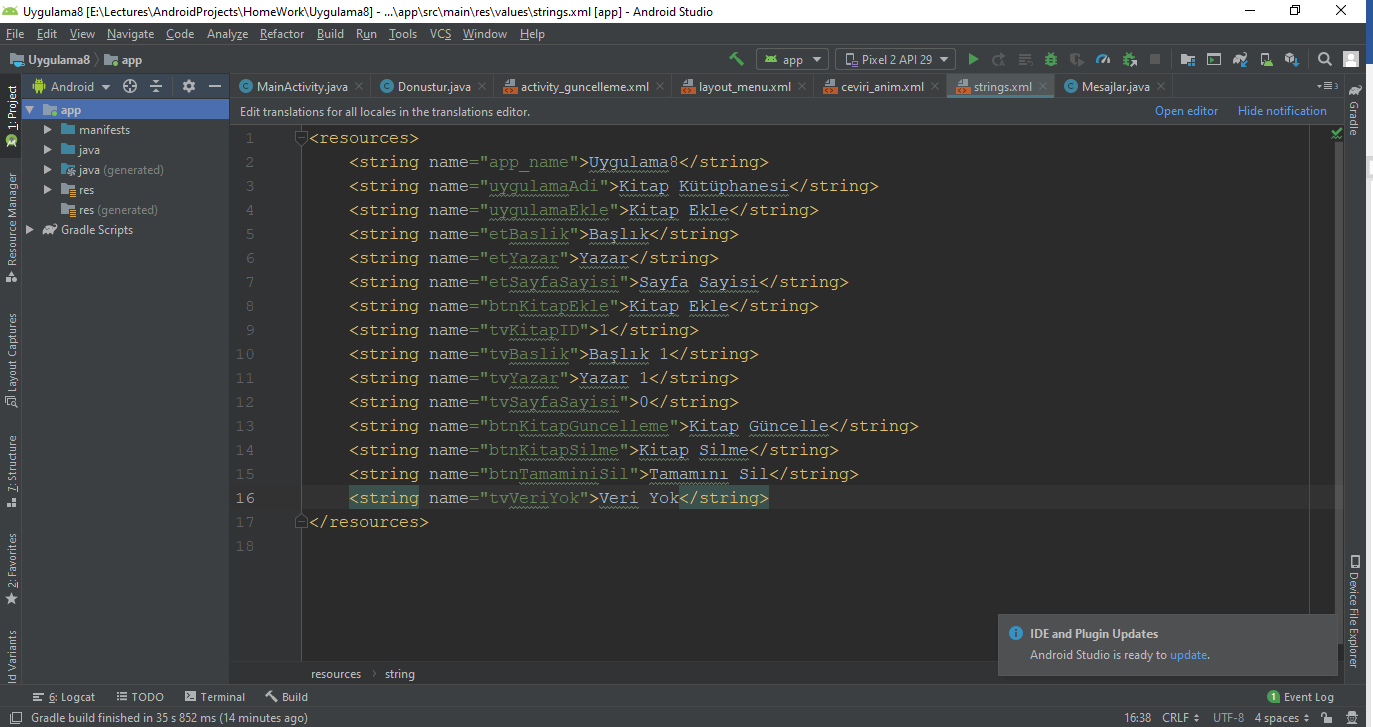
**Şekil 1. Çalışan Ekran Görüntüsü Formatı**

**Uygulama (100 puan): Uygulamada 3 farklı Activity tanımlıdır. Her aktivite de farklı bir görev vardır. Ana Aktivitede Ekli olan kitapları listeleyecek, Diğer 2 farklı aktivitelerin birisinde Kitapları Veritabanına Ekleme işlemi yapmaktadır. Bir diğerinde ise ana aktivitede seçili olan kitabı güncellemektedir. Uygulamada kullanılan resim dosyaları Şekil 2’de verilmiştir.**

Şekil 2’de görüldüğü gibi uygulamada kullanılan resimler, iconlar Vector Asset kullanılarak Android Studio içerisindeki hazır resimler kullanılarak elde edilmiştir.



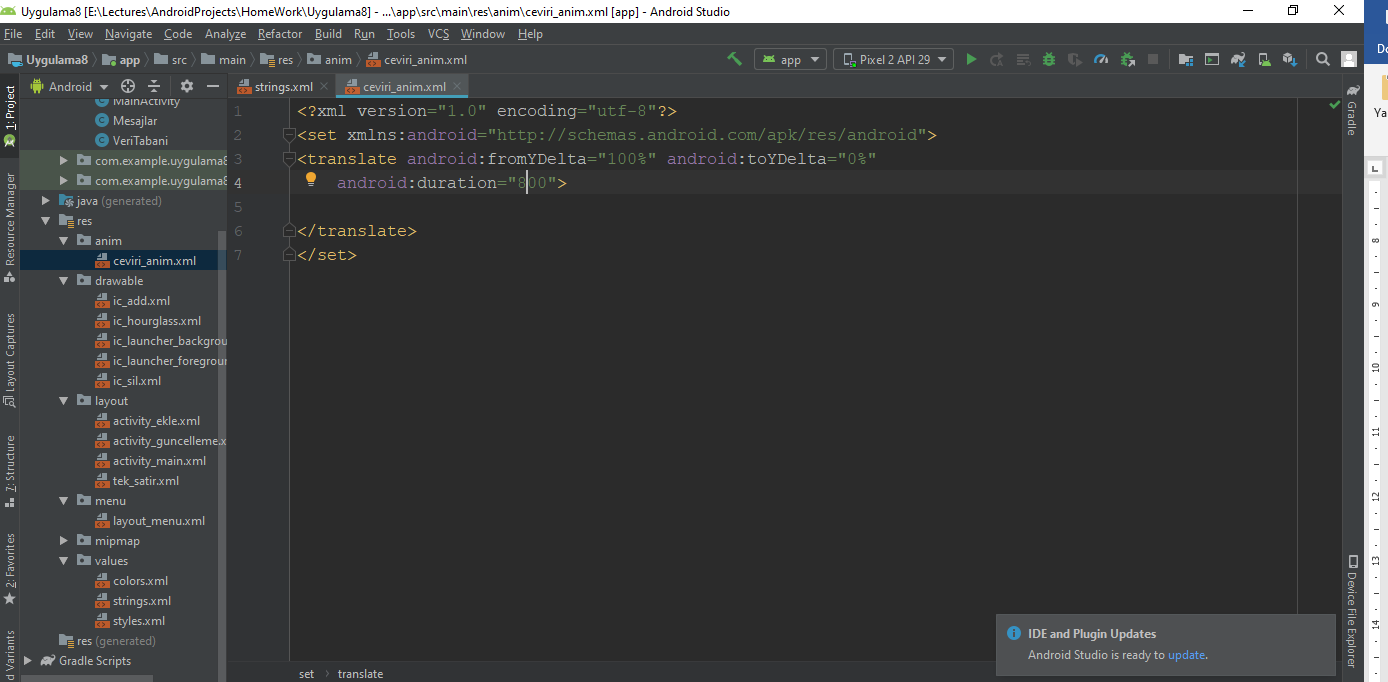
**Şekil 2.Uygulamada kullanılan resimler, Vector Asset içerisinden eklenmiştir.**



**Şekil 3.String.xml**

**ceviri\_anim.xml (Animasyon Dosyası) (10 puan)**

Ana Aktiviteye Gelindiğinde sayfanın animasyonel hareketini sağlayacaktır.



**Şekil 4.ceviri\_anim.xml**

Şekil 4.’de görülen icc\_add, ic\_sil, ic\_hourglass.xml dosyaları Şekil 2.’de görülen yoldan eklenmiştir.

**Activity\_main.xml (30 puan)**

En dışarıda clAna adında bir constaint layout tanımlıdır. **MainActivity’e** bağlıdır.

İçerisinde btnEkle adında bir FloatingActionButton, ivVeriYok adında ekranın ortasında bir image view, hemen altında tvVeriYok adında bir TextView, rvGoster adında genişlik ve yüksekliği match\_parent olan bir RecyclerView tanımlıdır.

btnEkle adındaki FloatingActionButton;

android:layout\_width="wrap\_content"  
android:layout\_height="wrap\_content"  
android:layout\_marginEnd="44dp"  
android:layout\_marginRight="44dp"  
android:layout\_marginBottom="36dp"  
android:clickable="true"  
android:focusable="true"  
app:layout\_constraintBottom\_toBottomOf="parent"  
app:layout\_constraintEnd\_toEndOf="parent"  
android:onClick="Goster"  
app:srcCompat="@drawable/ic\_add"

ivVeriYok adındaki image view,

android:layout\_width="wrap\_content"  
android:layout\_height="wrap\_content"  
android:layout\_marginTop="150dp"  
app:layout\_constraintEnd\_toEndOf="parent"  
app:layout\_constraintStart\_toStartOf="parent"  
app:layout\_constraintTop\_toTopOf="parent"  
app:srcCompat="@drawable/ic\_hourglass"  
android:alpha="0.1"  
android:visibility="gone">

tvVeriYok adındaki TextView,

android:layout\_width="wrap\_content"  
android:layout\_height="wrap\_content"  
android:layout\_marginTop="16dp"  
android:text="@string/tvVeriYok"  
android:textSize="20sp"  
android:alpha="0.4"  
android:visibility="gone"  
app:layout\_constraintEnd\_toEndOf="parent"  
app:layout\_constraintStart\_toStartOf="parent"  
app:layout\_constraintTop\_toBottomOf="@+id/ivVeriYok">

rvGoster adındaki RecyclerView,

<androidx.recyclerview.widget.RecyclerView  
android:layout\_width="match\_parent"  
android:layout\_height="match\_parent"

Özelliklerine sahiptir.

llSatir

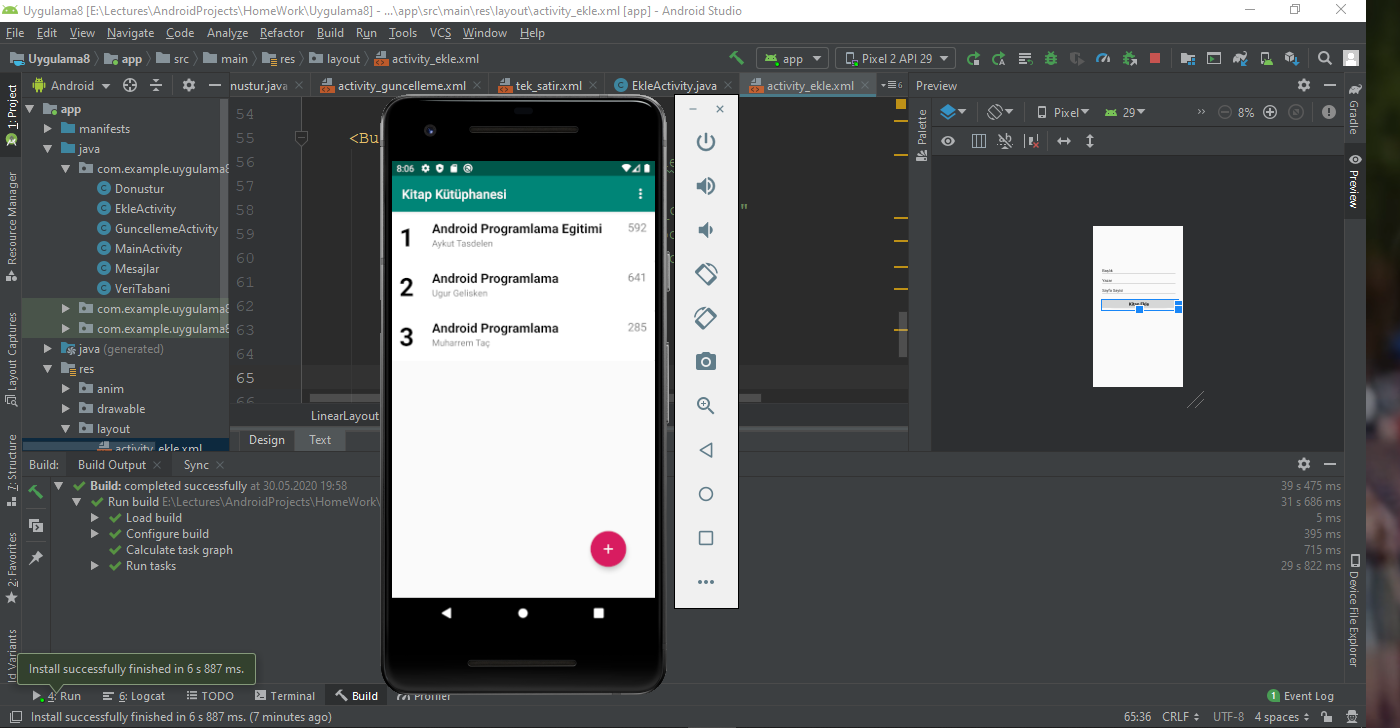
adındaki linearlayout’a tıklandığı zaman GuncellemeActivity’e tıklanan satırdaki verileri aktarması ve o ekranın açılması istenmektedir. Ayrıntılar Şekil 5-6-7’da verilmiştir.

**activity\_guncelleme.xml (30 puan)**

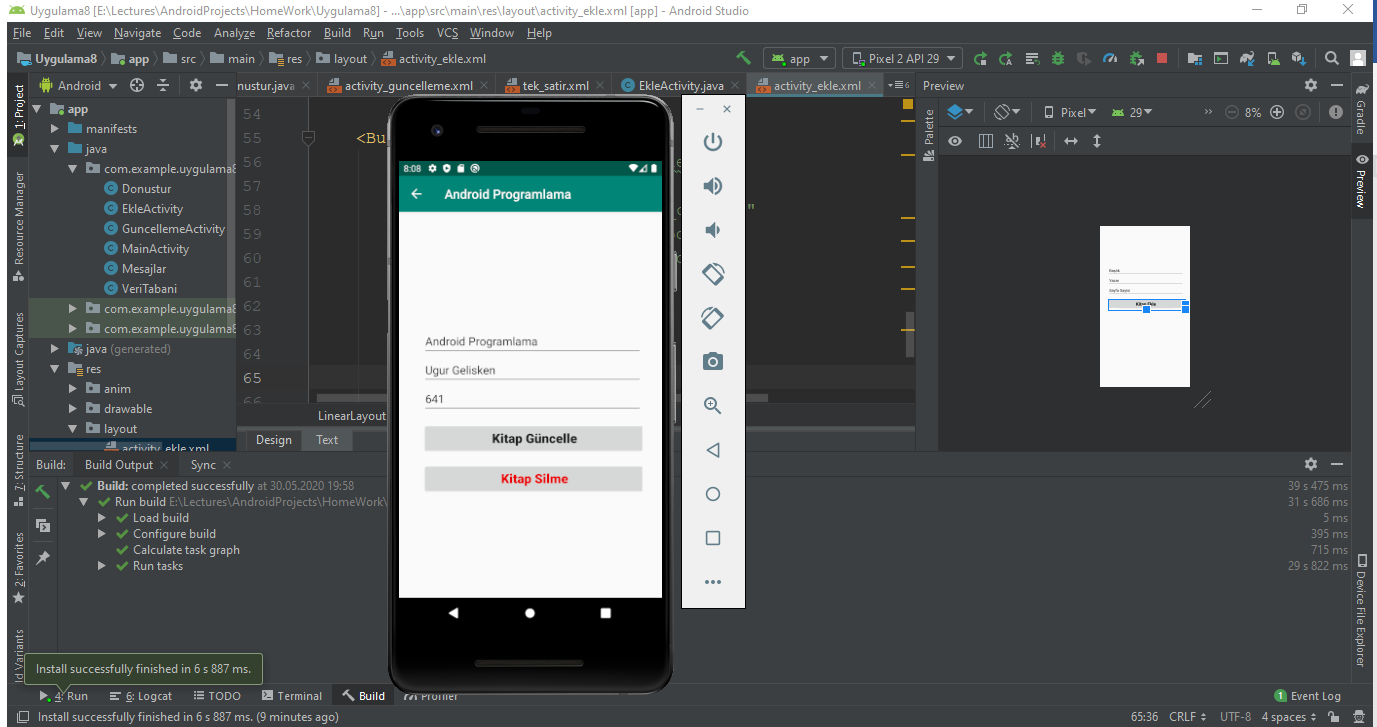
En dışarıda, llGuncelleme adında bir Linear Layout tanımlıdır. İçerisinde etBaslik, etYazar, etSayfaSayisi, btnKitapGuncelle, btnKitapSil adında bileşenler tanımlıdır.

**GuncellemeActivity java sınıfına bağlıdır.**

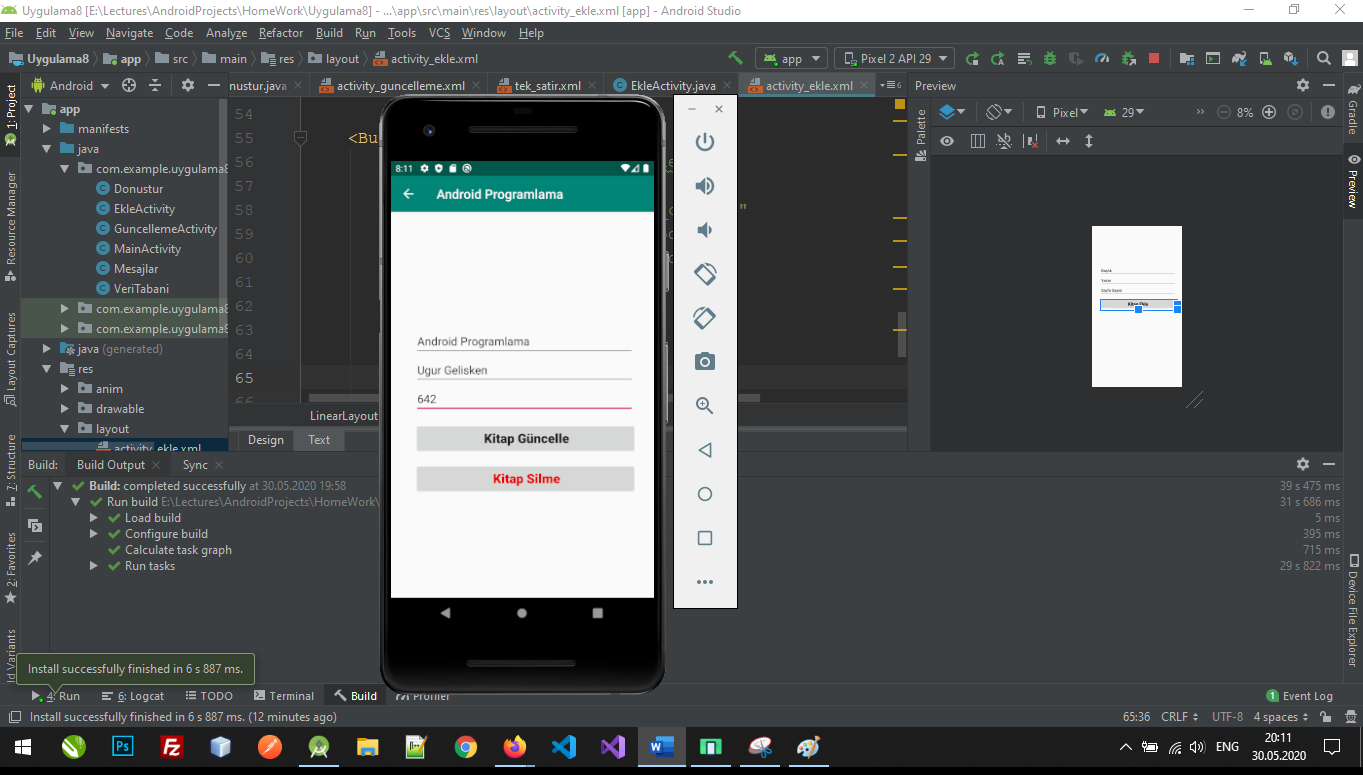
* **btnKitapGuncelleme butonuna tıklandığında KitapGuncelleme adında bir fonksiyon çağırınız. Bu fonksiyon içerisinde ana aktiviteden seçili satırı aktarıp, aktarılan satırdaki değerleri activity\_guncelleme içerisinde bileşenlere aktarınız. Aktardıktan sonra kullanıcı istediği değişiklik yaptıktan sonra KitapGuncelleme fonksiyon içerisinde kullanıcının belirtilen butonuna tıklanması sonrasında güncelleme işlemini yapınız. Ve sonrasında İşlem Başarılı ise Mesajlar adlı bir sınıf içerisinde tanımlı İşlem Başarılı adlı statik bir mesaj dondurup kullanıcıya bilgi veriniz. (15 puan)**
* **btnKitapSil adındaki butonuna tıklandığında KitapSilme adında bir fonksiyon çağırınız. Bu fonksiyon içerisinde ana aktiviteden seçili satırı aktarıp, aktarılan satırdaki değerleri activity\_guncelleme içerisinde bileşenlere aktarınız.Butona tıkladıktan sonra silme işlemini yapınız. Ve sonrasında İşlem Başarılı ise Mesajlar adlı bir sınıf içerisinde tanımlı İşlem Başarılı adlı statik bir mesaj dondurup kullanıcıya bilgi veriniz. Silme sonrasında activity\_main.xml dosyasının bağlı olduğu aktiviteye dönünüz (10 puan). Silmeden önce uyarı ekranı çıkması kullanıcı Evet derse silinsin, değilse dialog penceresi kapatılsın (Şekil 9, 5 puan).**



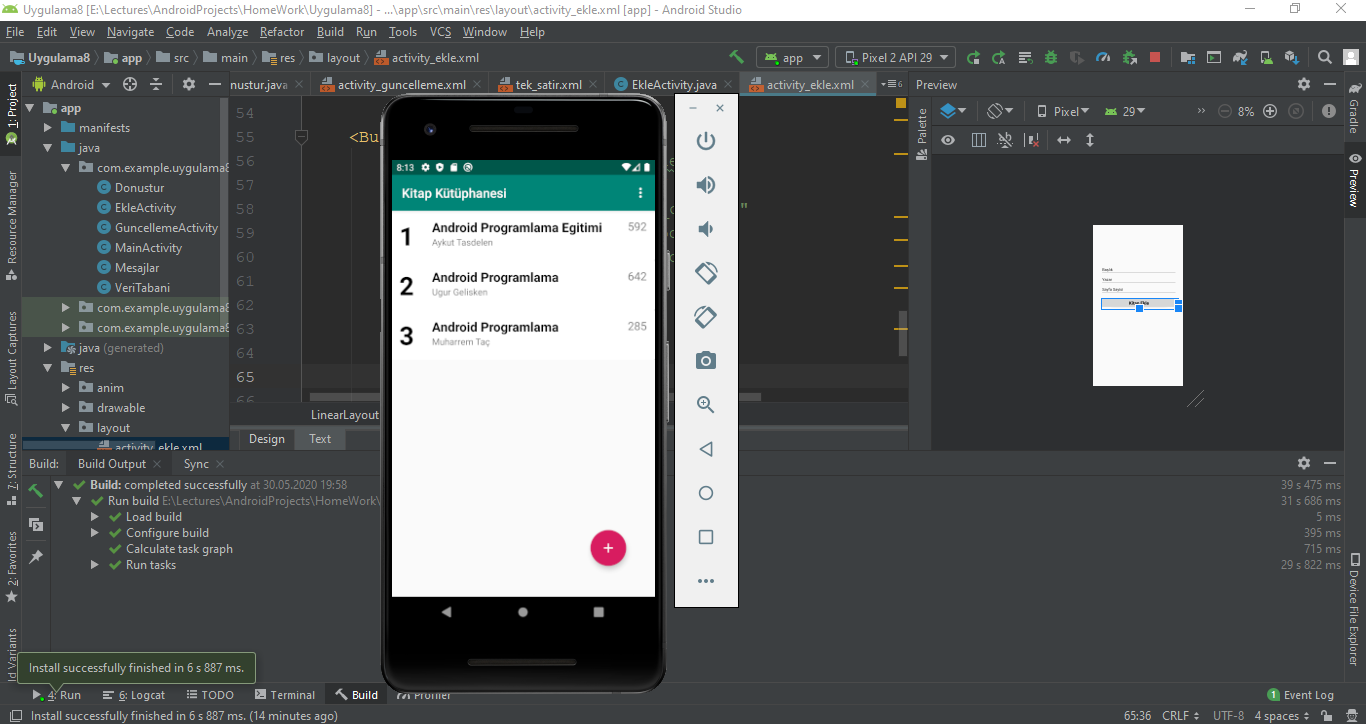
**Şekil 5.Güncelleme Öncesi, ana ekran satır seçimi ve güncelleme ekranına geçiş**

****

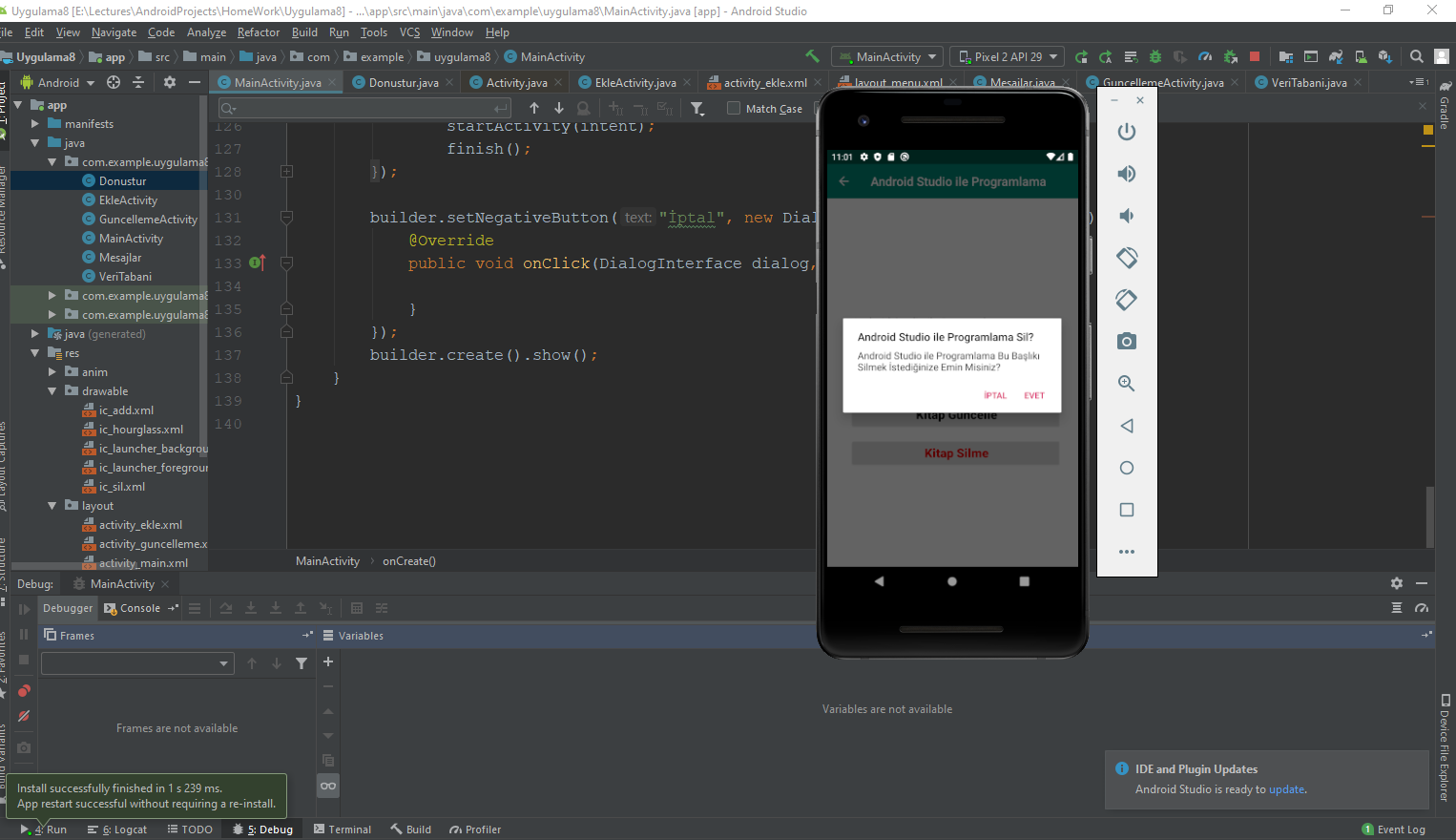
**Şekil 6.Güncelleme Veri Giriş Ekranı**

****

**Şekil 7.Güncelleme Veri Giriş Ekranı**



**Şekil 8.Güncelleme Sonrası, Ana Ekran**



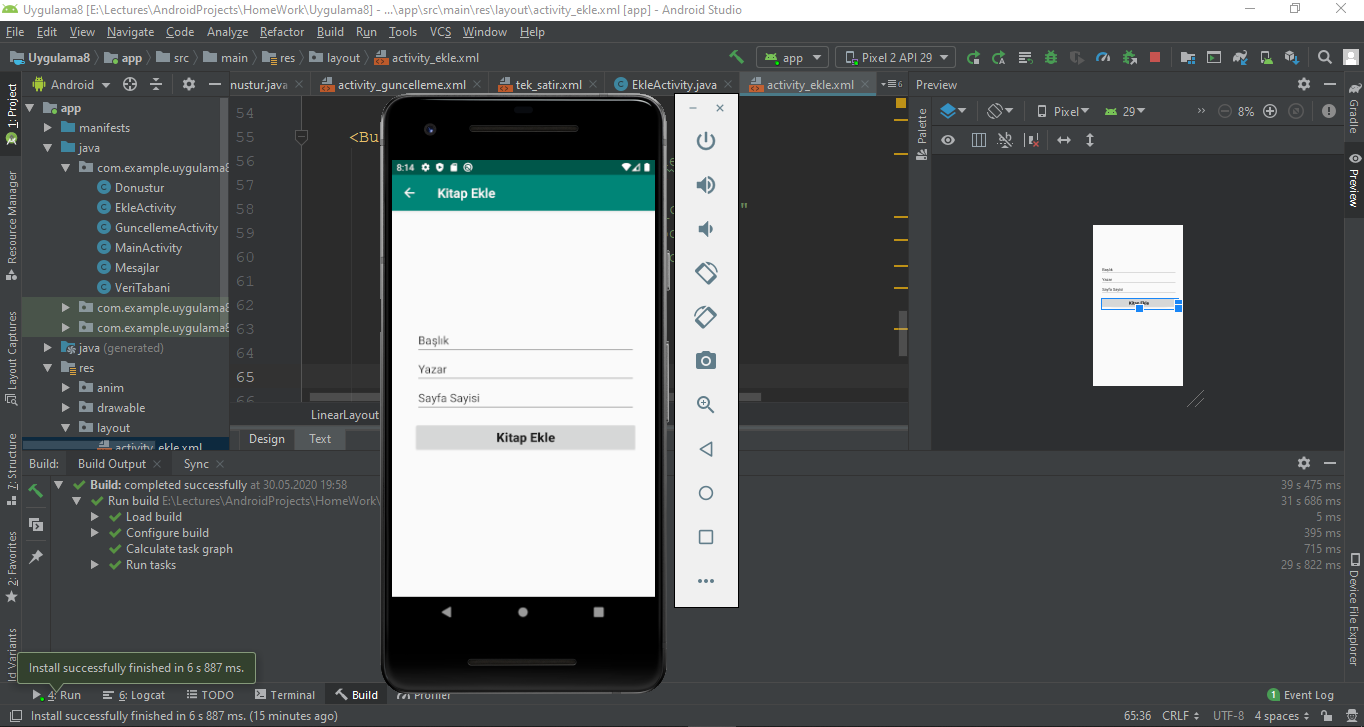
**Şekil 9.Silme sonrasında uyarı ekranı**

**activity\_ekle.xml (10 puan)**

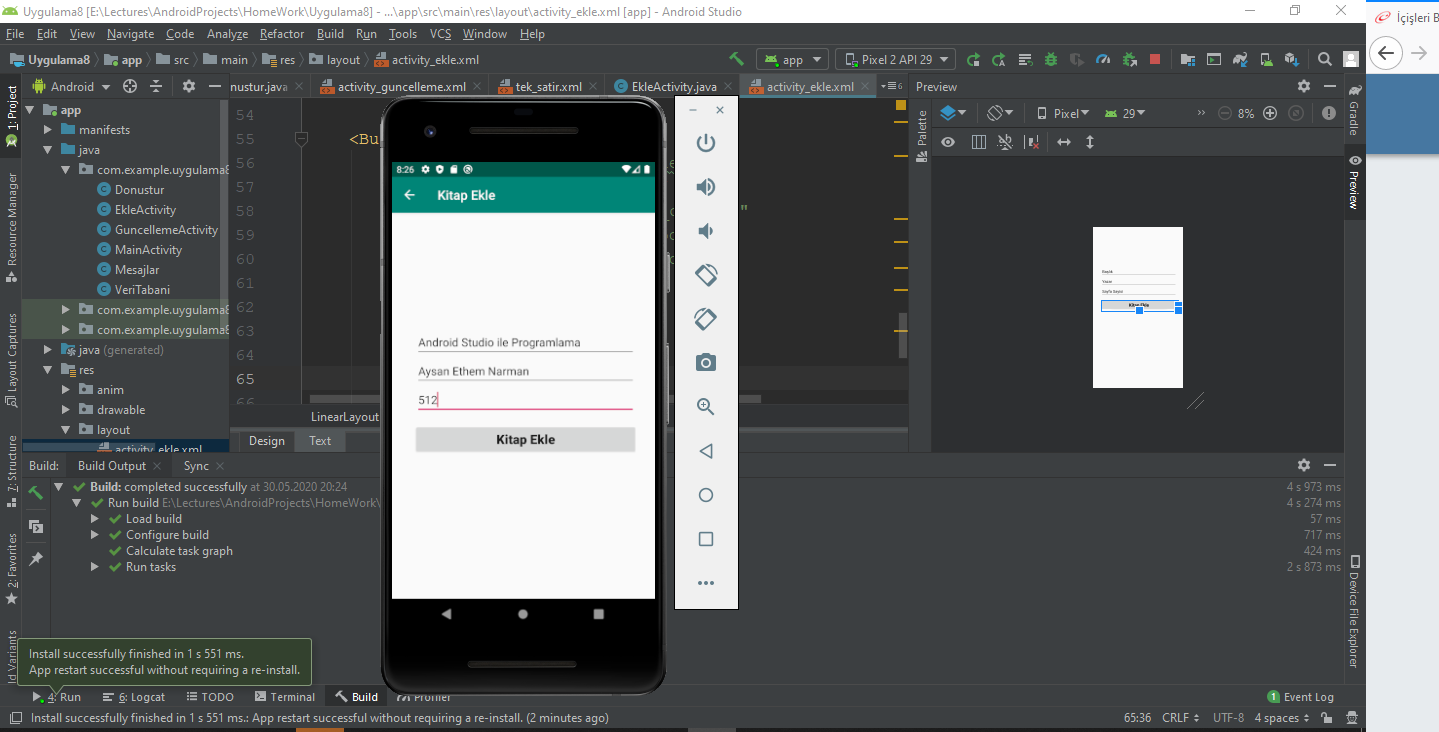
En dışarıda, llEkle adında bir Linear Layout tanımlıdır. İçerisinde etBaslik, etYazar, etSayfaSayisi, btnKitapEkle adında bileşenler tanımlıdır.

**EkleActivity java sınıfına bağlıdır.**

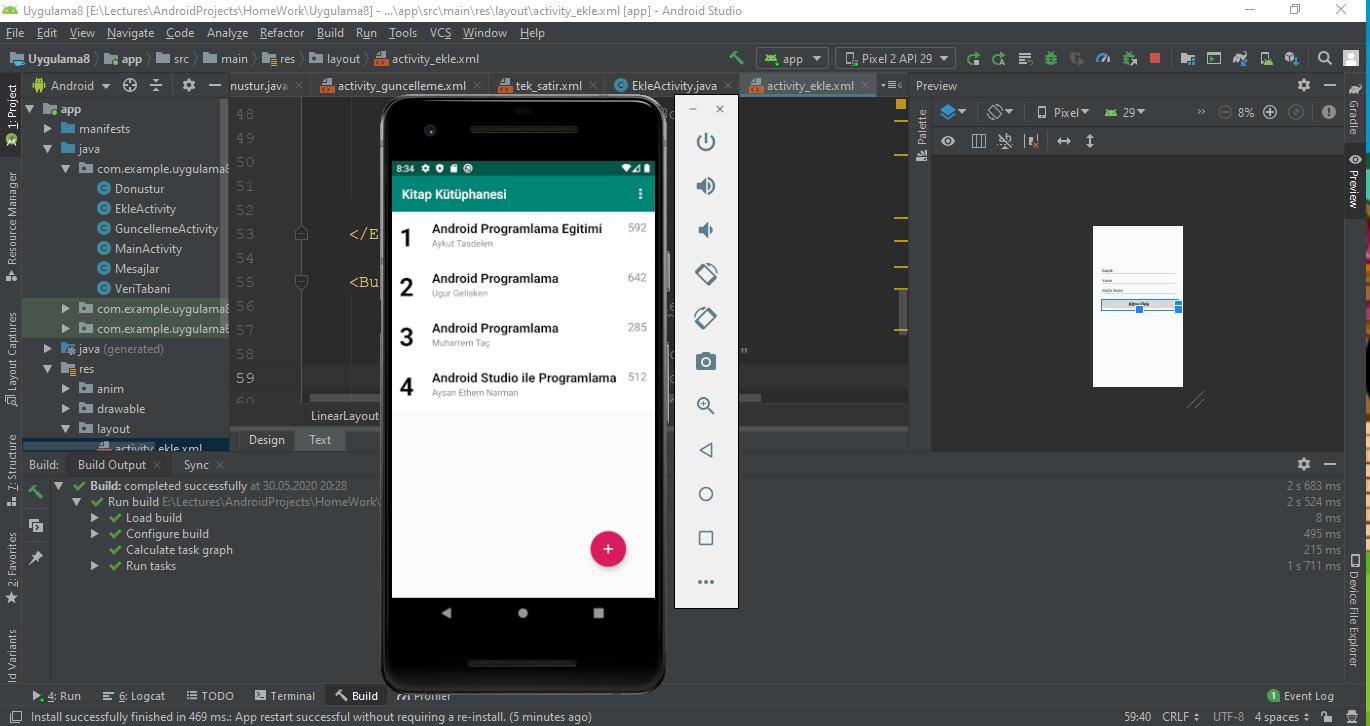
* **btnKitapEkle butonuna tıklandığında KitapEkle adında bir fonksiyon çağırınız. Bu fonksiyon içerisinde belirtilen bileşenlerden girilen değerleri sqlite veritabanına ekleyiniz. Ve sonrasında İşlem Başarılı ise Mesajlar adlı bir sınıf içerisinde tanımlı İşlem Başarılı adlı statik bir mesaj dondurup kullanıcıya bilgi veriniz.**



**Şekil 10. Ekleme Ekranı**

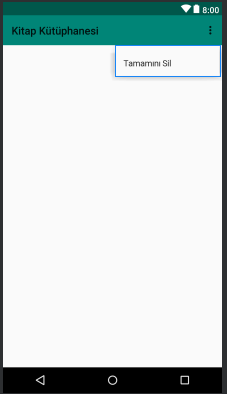


**Şekil 11.Ekleme Veri Giriş Ekranı**

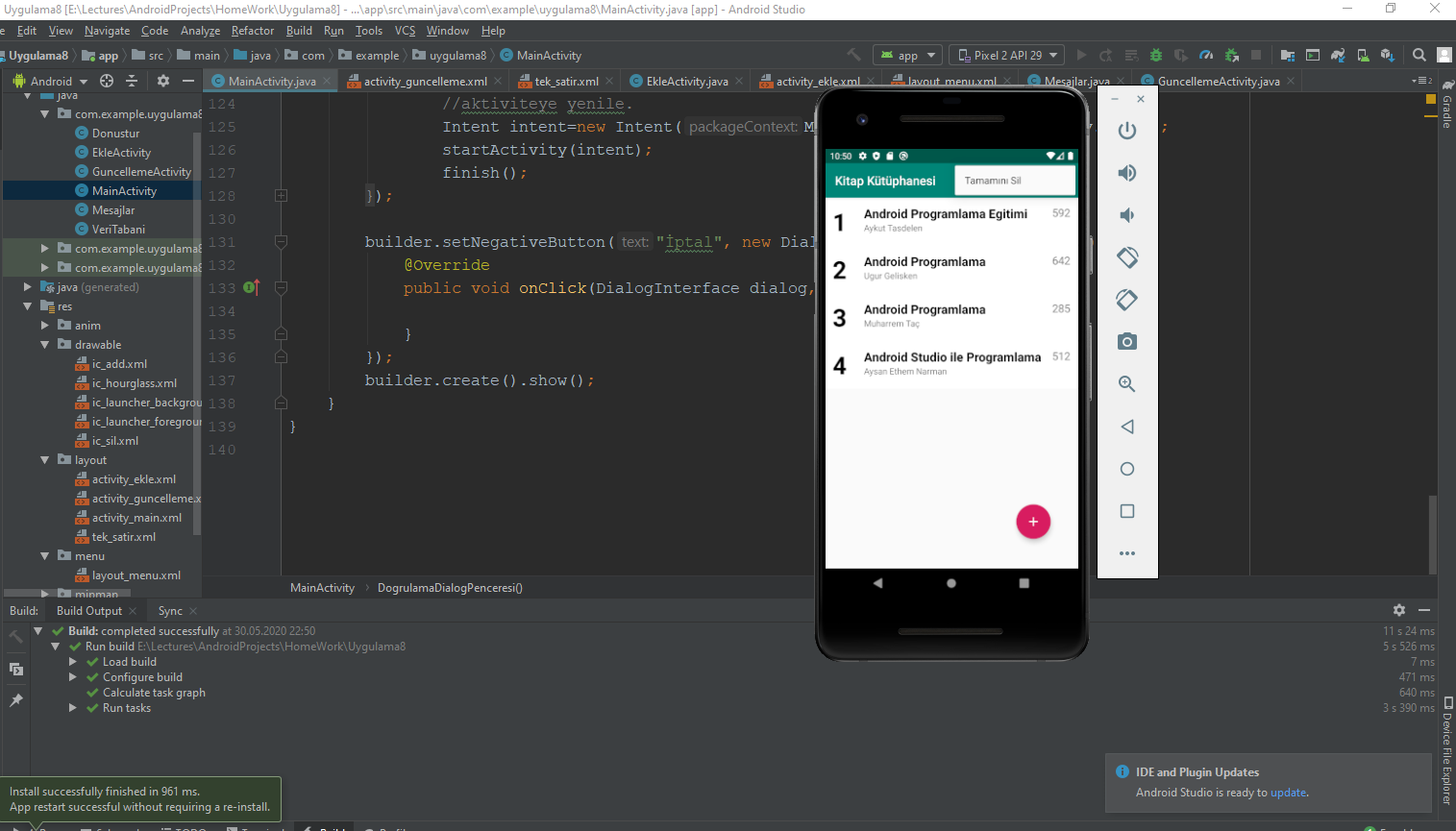


**Şekil 12.Ekleme Sonrası**

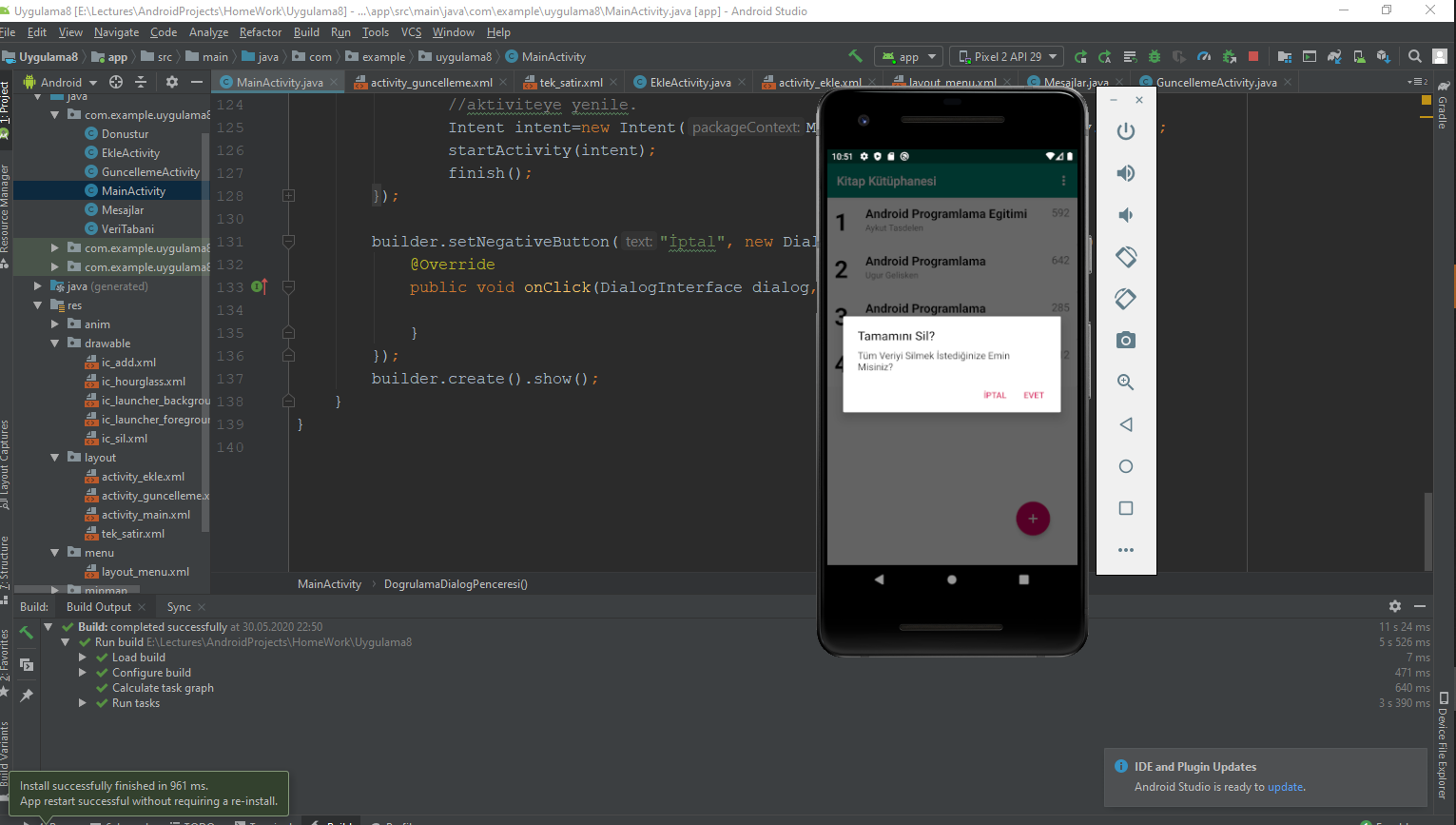
**Layout\_menu.xml (10 puan)**



**Şekil 13. Ana ekran Menu Ekranı**



**Şekil 14.Ana Ekran Menu Silme Görüntüsü**



**Şekil 15.Silme Öncesi Uyarı Dialog Penceresi**

**btnTamaminiSil adlı butona tıklandığı zaman, bir dialog penceresi gelecektir ekrana. Kullanıcı evet derse tüm kitapları veritabanında silmesi istenmektedir. İptal derse dialog penceresini kapatıp ekrana dönecektir.**