



**T.C**  
**BEYKENT ÜNİVERSİTESİ**  
**MÜHENDİSLİK – MİMARLIK FAKÜLTESİ**  
**YAZILIM MÜHENDİSLİĞİ BÖLÜMÜ**

## **KELİME BİLMECE OYUNU**

**Mobil Uygulama Programlama**  
**Dersi Dönem Projesi**

**Hazırlayan**  
**TUĞBA NUR BAYRAM**  
**1803013068**

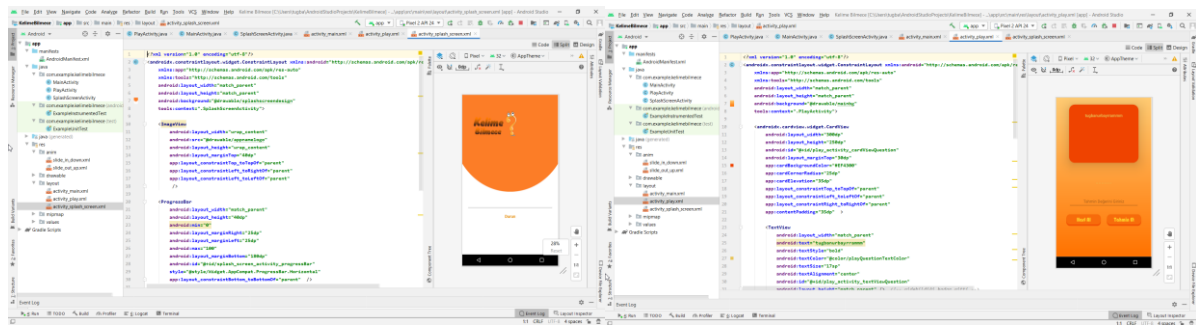
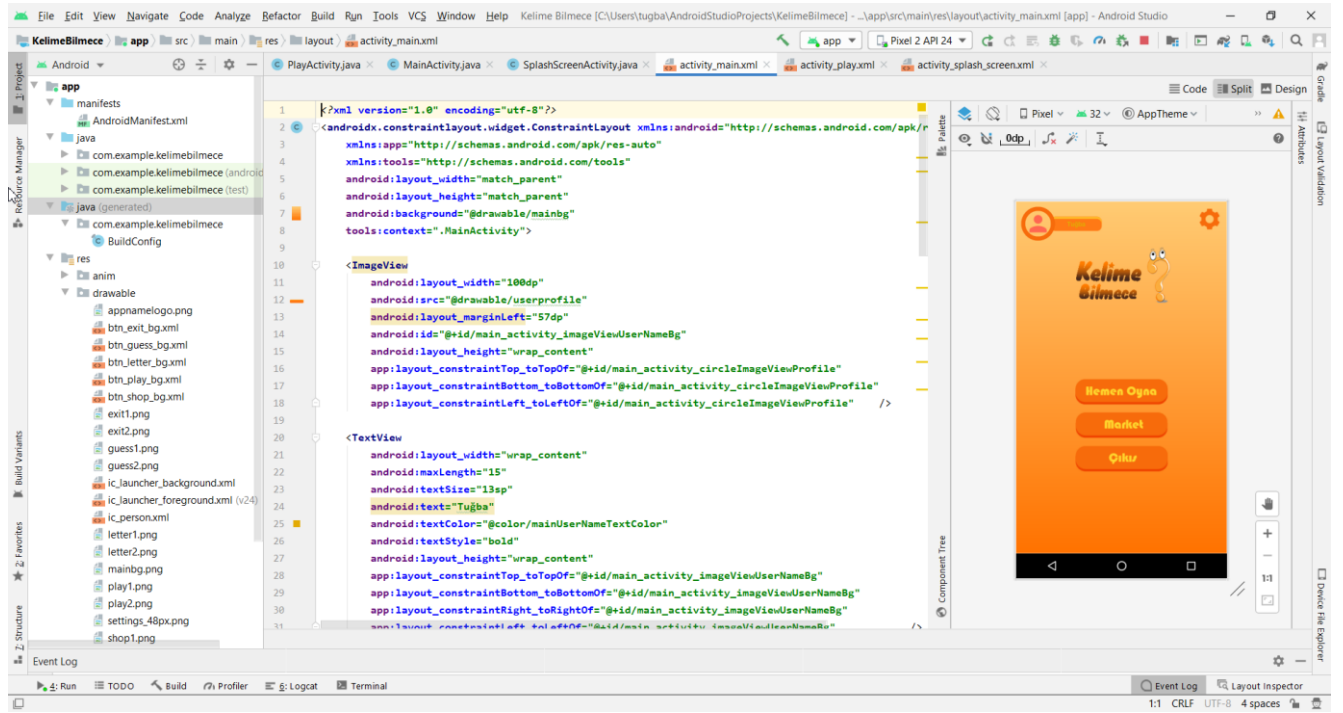
**Danışman**  
**Dr. Öğr. Üyesi Atınç YILMAZ**

# 1. Özet

Kelime Bilmecesi uygulamasının amacı bize doğrultulan soruların cevaplarını bilmeye çalışarak hem beynimizin kıvrımlarını çalıştırmak hem de boş vakitlerimizde eğlenceli bir oyun ile ufkumuzu genişletmektir. Kelime Bilmecesi Android Studio ile geliştirilmiş bir mobil uygulamadır. Ekranı düşen soruların doğru cevaplarını tahmin etmeye çalışıp, eğer kelime bilinmezse harf alınıp nihayetinde doğru kelimeyi yazmamız beklenmiştir. Uygulamanın bir sonraki aşamasında puan sistemi ile kullanıcıların birbiri ile yarışması sağlanacaktır. Zamanını keyifli şekilde değerlendirmek için herkesin yöneldiği oyunlar kategorisinde, oynamaktan mutluluk duyulacak bir mobil uygulama tasarlanmıştır.

## 2. Projenin Açıklaması

### Xml Dosyaları



Ekran görüntülerinde Drawable klosörüne png resimleri yüklenmiştir. Logo, ImageView ile şekillendirilmiştir. 3 tane butonlar Linear Layoutın içinde Button bölümünde tanımlanmıştır. Kullanıcın butonlara basmadığı ve bastığı durumlara özel farklı resimler state\_pressed ile

atanmıştır. Bu sayede kullanıcıya estetik bir tasarım sunulması amaçlanmıştır. Circle image view kütüphanesi eklenerek profil tasarımı yapılmıştır. CardView tasarımı ile ekrana düşecek sorular için bir bölme oluşturulmuştur. TextView ve EditText tasarımları ile kelimelerin yazılacağı bölüm tasarlanmıştır. SplashScreen dosyasında ProgressBar eklenmiştir.

## Java Dosyaları

```
activity.java x MainActivity.java x SplashScreenActivity.java x activity_main.xml x activity_pli
package com.example.kelimebilmece;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void mainBtnClick(View v){
        switch (v.getId()){
            case R.id.main_activity_btnPlay:
                Intent playIntent =new Intent( packageContext: this, PlayActivity.class);
                finish(); //aktiviteyi bitirmesi için
                startActivity(playIntent);
                overridePendingTransition(R.anim.slide_out_up, R.anim.slide_in_down);
                break;

            case R.id.main_activity_btnShop:
                break;

            case R.id.main_activity_btnExit:
                break;
        }
    }
}
```

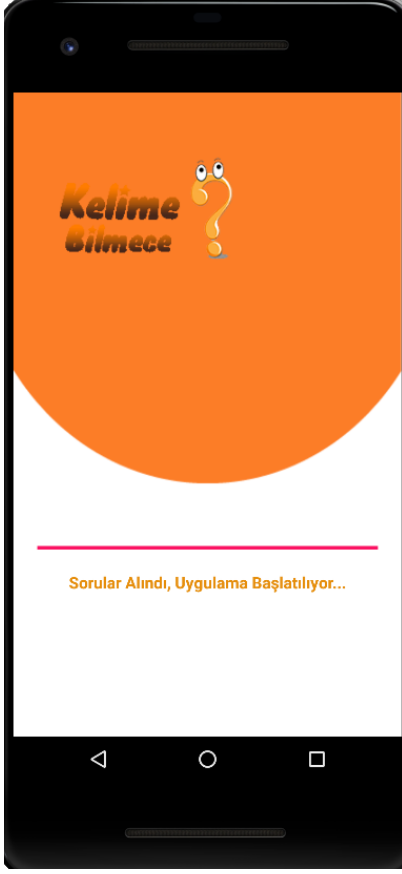
MainActivity dosyasında butonlar View aracılığı ile çağırılmıştır. Aktivitenin başlaması sırasında R.anim kodları ile ekran yukarı ve aşağı kaydırılacaktır.

```
MainActivity.java x PlayActivity.java x SplashScreenActivity.java x activity_main.xml x activity_play.xml x activity_splash_screen.xml
126 public void btnHarfAl(View v) { setRastgeleHarfAl(); }
129 public void btnTahminEt(View v){
130     textTahminDegeri = editTextTahminDegeri.getText().toString();
131
132     if (!TextUtils.isEmpty(textTahminDegeri)){
133         if(textTahminDegeri.matches(rastgeleKelime))
134             Toast.makeText(getApplicationContext(), text: "Doğru Tahmin.", Toast.LENGTH_SHORT).show();
135         else
136             Toast.makeText(getApplicationContext(), text: "Yanlış Tahmin.", Toast.LENGTH_SHORT).show();
137     }else
138         Toast.makeText(getApplicationContext(), text: "Tahmin Değeri Boş Olamaz", Toast.LENGTH_SHORT).show();
139
140 }
141 private void setRastgeleHarfAl() {
142     if (kelimeHarfleri.size() > 0){
143         rndHarfNumber = rndHarf.nextInt(kelimeHarfleri.size());
144         String[] txtHarfler = textViewQuest.getText().toString().split( regex: " "); // string döndüğü için string dizi
145         //System.out.println("Gelen Harf = " + kelimeHarfleri.get(rndHarfNumber));
146         char[] gelenKelimeHarfler = rastgeleKelime.toCharArray();
147
148         for (int i = 0; i< rastgeleKelime.length(); i++){ // uzunlugunu alıp her bir harfin içerisinde gezebileceğimiz
149             if (txtHarfler[i].equals("_") && gelenKelimeHarfler[i] == kelimeHarfleri.get(rndHarfNumber)){
150                 txtHarfler[i] = String.valueOf(kelimeHarfleri.get(rndHarfNumber));
151                 kelimeBilgisi = "";
152
153                 for (int j= 0; j< txtHarfler.length; j++){
154                     if (j< txtHarfler.length - 1 )
155                         kelimeBilgisi += txtHarfler[j] + " ";
156                     else
157                         kelimeBilgisi += txtHarfler[j];
158                 }
159                 break;
160             }
161         }
162     }
163 }
```

PlayActivity dosyasında soruları alınıp listlere atanmıştır ve random olarak sorular alınmıştır. Random gelen sorulara göre kelimeleri aldıktan sonra kelimeler üzerinde de random işlem uygulanmıştır ve diziye atanmıştır. Dizinin içerisinde rastgele kelime seçilip ekrana yazdırılmıştır. Kelimeler random alındıktan sonra diziden silinmiştir. Kelimelerdeki harf sayısına göre “\_” sayısı belirlenmiştir.

```
47 try {
48     database = this.openOrCreateDatabase( name: "KelimeBilmece", MODE_PRIVATE, factory: null);
49
50     database.execSQL("CREATE TABLE IF NOT EXISTS Sorular (id INTEGER PRIMARY KEY, sKod VARCHAR UNIQUE, soru VARCHAR)"); //benzersiz id atadık. unique b
51     database.execSQL("DELETE FROM Sorular"); //sürekli sürekli eklenmemesi için sildik
52     sqlSorulariEkle();
53
54     database.execSQL("CREATE TABLE IF NOT EXISTS Kelimeler(kKod VARCHAR, kelime VARCHAR, FOREIGN KEY (kKod) REFERENCES Sorular (sKod))"); //kKod u , sor
55     database.execSQL("DELETE FROM Kelimeler");
56     sqlKelimeriEkle();
57
58     cursor = database.rawQuery( sql: "SELECT * FROM Sorular", selectionArgs: null); // tüm kayıtları aldık
59     artacakProgress = maksimumProgres / cursor.getCount(); //içersindeki eleman sayısını getirdi //artacak progres miktaro databasedeki soru sayısına g
60
61     int sKodIndex = cursor.getColumnIndex( columnName: "sKod"); //columnindexleri belirledik
62     int soruIndex = cursor.getColumnIndex( columnName: "soru");
63
64     mTextView.setText("Sorular Yükleniyor...");
65
66     while (cursor.moveToNext()) {
67         sorularHashMap.put(cursor.getString(sKodIndex), cursor.getString(soruIndex)); //hasmap'te de benzersiz olacağı için hasmap haline getirdik
68         progresMiktari += artacakProgress;
69         mProgress.setProgress((int)progresMiktari); //döngü içinde verileri alırken progress miktarını arttırdık
70     }
71
72     mTextView.setText("Sorular Alındı, Uygulama Başlatılıyor...");
73     cursor.close();
74
75     new CountDownTimer( millisInFuture: 1100, countDownInterval: 1000) {
76
77
78     @Override
79     SplashScreenActivity -> onCreate()
```

SplashScreen java dosyasında gelen sorulara göre kelimeleri alacağımız bir veri tabanı oluşturulmuştur. Her soru için kendine ait bir kod olacaktır. Veri tabanına sorular ve kelimelerin verileri eklenmiştir. Veri miktarını aldıktan sonra maksimumProgress miktarına bölerek artacakProgress miktarı belirlenmiştir.



Ekran görüntüsünde uygulamanın ilk açılış ekranı görülmektedir. Uygulamaya tıklanıldığında üst tarafta uygulamanın logosu bizi karşılamaktadır. Alt bölmede ise progress bar ile yapılan işlemlerin arka plandaki ilerleyişini kullanıcıya göstermemize yarayan bir arayüz bileşeni yani widget ile tasarım şekillendirilmiştir. Progress bar; veri tabanına eklenen soru miktarına göre dolacaktır. Diğer ekrana geçişi progress bar dolduğunda gerçekleşmektedir. Progress bar'ın altındaki "sorular alındı, uygulama başlatılıyor" textview 'i ile kullanıcıya bilgi vermek amaçlanmıştır.

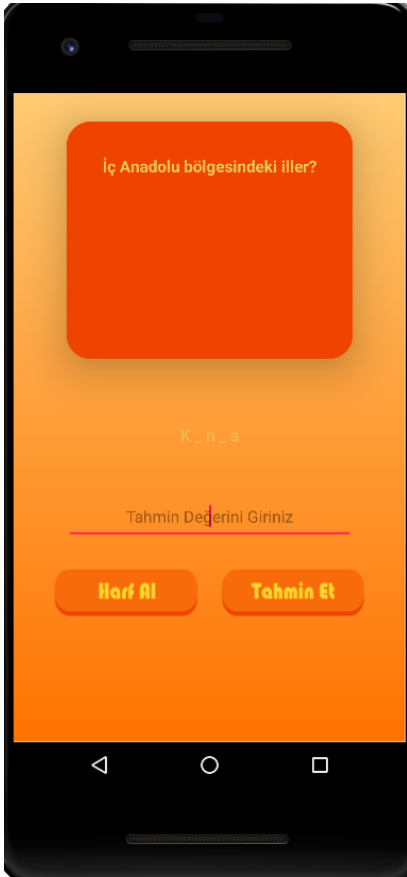


Ekran görüntüsünde kullanıcıyı ana menü tasarımı karşılamaktadır. Sol üst köşede kullanıcının ismi görülmektedir. Sağ kısımda ayarlar butonu orta kısımda logo, alt kısımda ise Hemen Oyna, Market ve Çıkış işlemlerinin yapılabildiği butonlar görülmektedir. Hemen Oyna butonuna tıklanıldığında kullanıcın karşısına oyun arayüzü çıkacaktır.

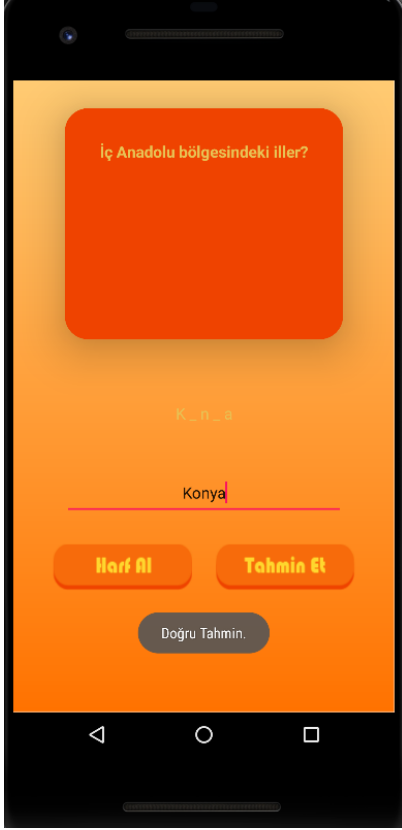


Ekran görüntüsünde, kullanıcıya doğrultulan soru ve cevabın 5 harfli olduğu (5 tane “\_”) görülmektedir. Bu kısımda kelime uzunluğu 7’ye eşit veya küçükse otomatik olarak sistem 1 tane harf verecektir. Kelime uzunluğu 8 ve 10 arasında veya 8 ve 10 ‘a eşit ise sistem 2 tane rastgele harf atayacaktır. Eğer kelime uzunluğu 11 ve 14 harf arasında ise 3 harf, 15 den büyük ise 4 harf rastgele sistem tarafından atanacaktır. Görüldüğü üzere kelime 5 harflidir ve sistem son harfini otomatik olarak kullanıcıya göstermiştir.

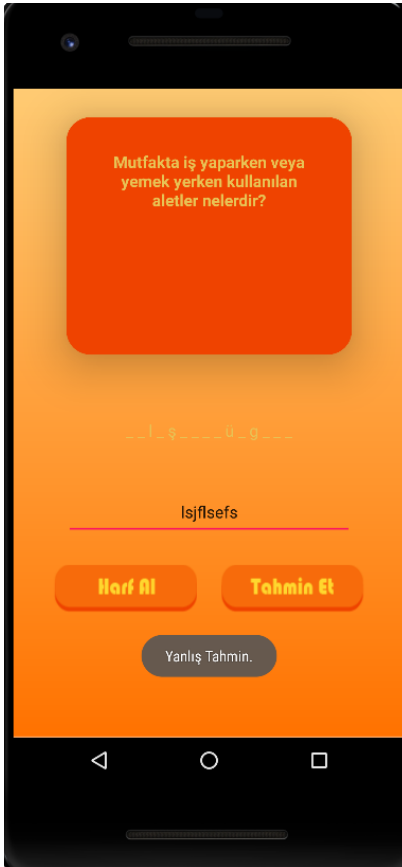
Kullanıcı dilerse tahmin ettiği kelimeyi klavye ile tuşlayabilir. Dilerse Harf Al butonuna tıklayarak harf alabilir.



Ekran görüntüsünde, Harf Al butonuna tıklandığında sistem “k” harfini ve sonra tekrar tıklanıldığında da “n” harfini kullanıcıya göstermiştir. Harf Al butonu başarılı şekilde çalışmaktadır.

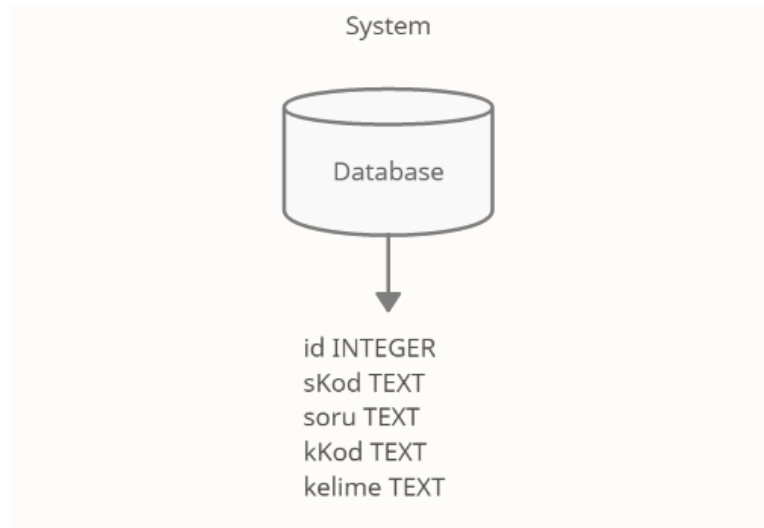
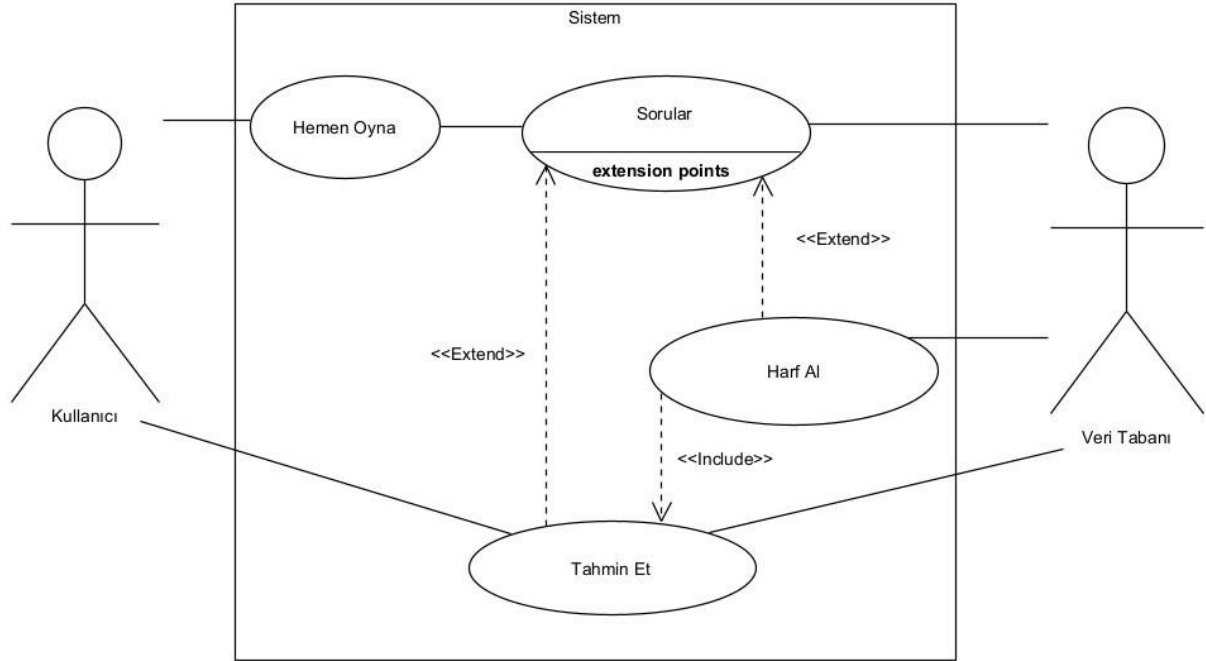


Ekran görüntüsünde, kullanıcın doğru kelimeyi tahmin ettikten sonra Tahmin Et butonu ile kontrolünü gerçekleştirdiğini görmekteyiz. Sistem tarafından ekrana “Doğru Tahmin” yazdırılmıştır.



Ekran Görüntüsünde, yanlış tahmin yapıldığı üzere sistem tarafından ekranda kullanıcıya “Yanlış Tahmin” bildirisi yapılmıştır.

### 3. Use Case Diyagramları





## 4. Sonuç

Android Studio işleyiş mantığını kavradım, bu süreçte bol bol hata alarak hata çözme becerimi geliştirdim ve telefonlarımızdaki tüm uygulamaların karşımıza sunulana kadar ne tür aşamalardan geçtiğini yakından öğrendim. Uygulama tamamen bitmiş olmamasına rağmen büyük bir kısmı tasarlanmıştır ve doğru bir şekilde çalışmaktadır. Kelime Bilmece oyununda kullanıcılar gelen sorulara doğru yanıtlar vermeye çalışarak keyifli vakit geçireceklerdir. İlerleyen zamanda uygulamayı geliştirerek github'a eklemeyi amaçlamaktayım.

## 5. Kaynakça

Circle Image View Kütüphanesi

<https://github.com/hdodenhof/CircleImageView>

Android Emülatör Kurulumu

<https://www.youtube.com/watch?v=2pl1wSwwZdw&list=PL20Zn-5nPIPHvLPq5xJTtImOd0qeNd9rW&index=3>

Android SQLite Kullanımı

<https://ayselaydin.medium.com/android-sqlite-kullan%C4%B1m%C4%B1-1477a89fc0ad>

Visual Paradigm İle Use Case Çizimi

<https://online.visual-paradigm.com/>