

Yazılım Laboratuvarı-2 Dersinin İkinci Projesi

Kelime Oyunu

Zeynep Duvarcı (210201113), Mehtap Özbunar (200201095)

*Bilgisayar Mühendisliği
Kocaeli Üniversitesi*

Özet

Bu rapor Yazılım Laboratuvarı 2 dersinin 2. Projesini açıklamak ve sunumunu gerçekleştirmek için oluşturulmuştur. Raporda projenin içeriği, yöntemi, deneysel, sonuç kısımları bulunmaktadır. Ek olarak kaynakçayı da eklemiş bulunmaktayız. Projenin gereklilikleri kullanıcının metinleri ve sonuçları görebileceği arayüz oluşturmak sonucu veritabanına kaydetmek ve listelemek. Metinleri birleştirirken ise kendi oluşturduğumuz algoritmayı kullanmaktır.

Giriş

Java dilinde Android için bir kelime oyunu yaptık. Oyunda kullandığımız kelime havuzunu ise veri tabanında sakladık. Veri tabanı için ise MongoDB Atlas kullandık. Oyunumuzu zorluk seviyesine göre iki mod şeklinde oluşturduk. Oyunun içeriğine gelecek olursak; oyunumuz harflerin düşmesiyle başlıyor ve bir sütunun yukarıya doğru harfler ile tamamen dolmasıyla sonlanıyor. Amaç, bu süreçte var olan harflerden kelimeler oluşturmak. Seçilen kelimelerin doğruluğuna göre kelimede var olan harfler ekrandan siliniyor ve puan kazanılmış oluyor. Harfler belli aralıklarla düşmeye devam ediyor, oyun sonlandığında ise puan veri tabanına kaydediliyor.

Yöntem

Uygulamayı oluştururken ortam olarak Android Studio kullandık. Veri tabanı olarak ise MongoDB Atlas kullandık.

Öncelikle harfler için gerekli verileri tutabilmek amacıyla LinkedHashMap tipinde 10x8'lik bir matris yapısı oluşturduk.

LinkedHashMap yapısı:

key: "index_in_alphabet" – harfin, Alphabet class'ından dönen alphabet_unordered ArrayList'indeki indexini tutuyor.

key: "index_in_word" – Harf seçimi kaldırıldığında harfi kelimeden silmek için harfin kelimedeki indexi tutuluyor.

key: "type" – Harfin tipi tutuluyor. (normal, orijinal buz, yapay buz)

key: "count_of_clicks" – Harfin tıklanma sayısı tutuluyor.

Harflerin yer alacağı arayüz için harflerin hareketini oluşturmak adına, 10x8'lik imageView'lerden oluşan bir activity tasarladık.

Harflerin satır satır düşmesi için CountdownTimer nesnelerini projemize dahil ederek, onTick metodu içinde imageView nesnelerinin resimlerini düzenleyerek (harf resmi ve siyah resim) harf hareketini sağladık.

Harflerin hangi satırda duracağını ise alt satırındaki matris elemanının null olup olmaması kontrolüyle sağladık. Alt satırdaki eleman null ise harf harekete devam ediyor, null değilse duruyor şeklinde bir yapı oluşturduk.

Harflerin düşmesi esnasında bazı harflerin (random olarak belirleniyor) buz harf olarak düşmesini sağladık. Düştüğü anda bu harfe temas eden diğer harfler de buz harf özelliği kazandığından bu durumun kontrolünü yaptık ve buz harfe dönüşen harfler için gerekli güncellemeleri (resim, type...) gerçekleştirdik.

Bunun dışında düşen harf buz harf olmasa bile düştüğü anda etrafında buz harf varsa buz harf

özelliği kazanması gerektiği için bu durumu da kontrol ettik ve buza dönüşen harf için gerekli güncellemeleri (resim, type...) yaptık.

Hafiflerin düşme aralıklarını oyun sırasındaki puana göre düzenledik. (Puan arttıkça harflerin düşme sıklığı artacak şekilde)

Sonrasında onaylama ve seçilen kelimeyi silme butonlarını ekleyerek gerekli işlevleri tanımladık.

Tanımladığımız işlevler şu şekilde; Onaylama butonuna tıklandığında kelimenin veri tabanındaki varlığı kontrol ediliyor, eğer varsa, puan güncelleniyor, seçili harfler (buz harfler hariç) ekrandan siliniyor ve boş kalan yerler üst satırdaki harflerle dolduruluyor. Buz harflerin ekrandan silinmesi ise 2. defa veri tabanında yer alan bir kelime kullanılmasıyla oluyor.

Veri tabanında yok ise yanlış girilen kelime sayısı kaydediliyor, bu sayı 3 olduğunda bir satır harf düşüyor.

Silme butonuna tıklandığında ise seçilen kelime textView' dan siliniyor ve seçili harflerin resmi ve harflerin özellikleri (resim, tıklanma sayısı...) değiştiriliyor.

Harflerin düşmesi sırasında eğer bir harf 1. satırda kalırsa ve ilerlemezse, o sütunlara ait tüm satırların dolduğu ve oyunun sonlandığı anlamına gelir, bu durumda puan veri tabanına kaydedilerek işlemler tamamlanıyor.

En yüksek 10 puana ait puan listesine de ekranın aşağısına eklediğimiz buton ile ulaşılıyor.

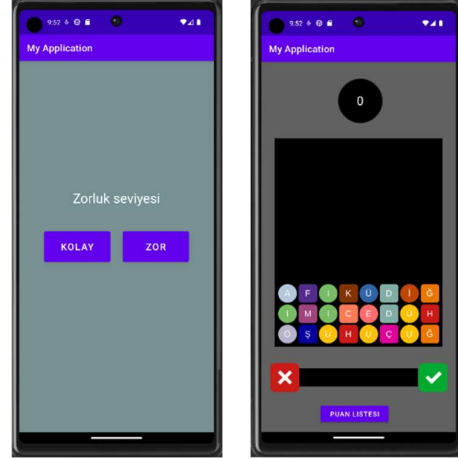
Deneyisel

Algoritma üzerinde birden fazla kelime ve kelime olmayan harf grubu denedik. Çoğu zaman beklediğimiz sonuçları aldık ve algoritma üzerinde düzenlemeler yaptık. Bu düzenlemelerde konuya uygun olmasına dikkat ettik.

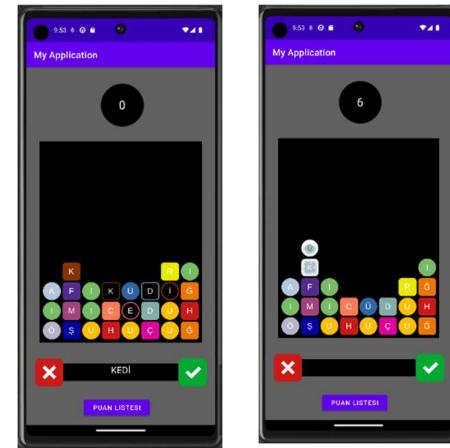
MongoDB'de oluşturduğumuz kelime veri setiyle kullanıcı tarafından girilen kelimelerin kontrolünü yaptık.

Sonuçlar

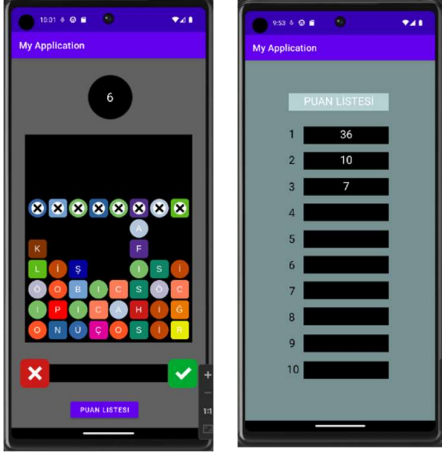
Öncelikle oyunda belirlediğimiz iki seviyeden birini kullanıcının seçebileceği bir arayüz ekranı tasarladık. Kullanıcının seçiminden sonra üç satır harfin aşağı düştükten sonra kullanıcının oyuna başlamasından emin olduk.



Kullanıcı oyuna başladığında en az üç harflik girdiği kelimeleri veri tabanında kontrol ederek sildik ve üzerindeki harflerin aşağı kaydığından emin olduk. Aynı zamanda düşen buz harfin de değdiği harflere etki ettiğini de deneyimledik. Girilen kelimenin harf bazlı puanlamasını toplam puana ekledik.



Daha sonra kullanıcı olarak üç adet verisetinde karşılığı olmayan kelime girdik ve ceza olarak ekstra bir satır harfin düşmesini sağladık. Aynı zamanda kullanıcının dilediğinde puan sıralamasını göreceği arayüzde puanların büyükten küçüğe sıralandığından emin olduk.



Sonuç

Java ve MogoDB Atlas kullanarak bir mobil oyun yapmış olduk. Raporun devamında kaynakça ve sözde kod ya da akış şemamızı ekledik. Algoritmayı daha yakından inceleyebilirsiniz.

Kaynakça

MongoDB:

<https://www.mongodb.com/docs/atlas/>
<https://www.mongodb.com/docs/realm/>
<https://www.youtube.com/playlist?list=PLBqHLq3IFiRLzpPgWwP-eUfazUBOObm-F>

Android Studio:

<https://developer.android.com/docs>
<https://www.youtube.com/playlist?list=PL20Zn-5nPIPHvLPq5xJTTImOd0qeNd9rW>

Çeşitli sorular için:

<https://stackoverflow.com>

Sözde Kod

Oyunun başlangıcı için

create_start_count_down_timer

fonksiyonunu tanımla.

CountDownTimer nesnesi tanımla

onTick metodu içinde

0. satırda değilse önceki satırın resmini
"black" yap.

Matris elemanına hangi harf geleceğini
belirle (harfin alfabedeki sırası)

Eğer k=0 değilse

Random sayı belirle

temporary_index_array[m]'e ata

imageViews[k][m]' in resmini ayarla(

alphabet_unordered'in

temporary_index_array[m] indexindeki
resmi)

k'yı arttır

onFinish metodu içinde

k=(finish_time-500)/500;

Harfin alfabedeki sırası matrix arraylist ine[k]
[m] kaydediliyor.

Harfin türü (buz veya normal...) matrix
arraylist ine[k][m] kaydediliyor.

Tıklanma sayısı ilk etapta 0 olarak matrix
arraylist ine[k][m] kaydediliyor.

Eğer finish_time-500<4000 ise

Harfleri tıklanır yap

"Oyun başladı" yaz

create_game_count_down_timer(5000,
1000)'ı çağır

Değilse

create_start_count_down_timer(finish_time
- 500, tick_time)'ı çağır

Harflerin satır satır düşmesi için

create_move_count_down_timer

fonksiyonunu tanımla.

CountDownTimer nesnesi tanımla

if sesli harf sayısı azsa {

Düşecek harf sesli harf olsun

değilse

Sessiz harf olsun

Düşecek harfin buz olup olmayacağını
random olarak belirle
Düşen harf buz ise
Ona temas eden harfler yapay buz harf olsun
Resimler ve değişkenleri güncelle
Değilse
Düştüğünde etrafında buz harf varsa bu harf
yapay buz harf olsun
Resimler ve değişkenleri güncelle

Harf, alt satırdaki matris elemanı null
olmayana kadar düşmeye devam etsin
Null ise
Dursun
Matris elemanına değerleri ata
create_game_count_down_timer'ı çağır

Eğer harf 1. satırda kaldıysa
"Oyun Bitti" yaz
Harfleri tıklanamaz yap
Puanı veri tabanına kaydet

Oyunun devamı için
create_game_count_down_timer
fonksiyonunu tanımla.
CountDownTimer nesnesi tanımla
onFinish metodu içinde:
create_move_count_down_timer'ı çağır.

3 kez yanlış kelime girildiğinde 1 satır harf
düşmesi için
create_wrong_count_down_timer
fonksiyonunu tanımla.
CountDownTimer nesnesi tanımla
onTick:
create_start_count_down_timer
fonksiyonuna benzer şekilde işlemler
gerçekleştirilecek. Farklı olarak 1 kez
gerçekleştirilecek.

Harfler yok olduğunda boşlukların dolması
için **fill_in_the_empty_space** adında
recursive fonksiyon tanımla. (Bu fonksiyon
kolay mod için kullanılıyor)
Fonksiyona gönderilen harfin bulunduğu
matris elemanına null ata

Resmini "black" yap.
Üst satırı kontrol et.
Önce aynı sütunda üst satırdaki elemanı, bu
eleman null ise, sol çapraz, sağ çapraz
şeklinde tüm üst satırı kontrol et.
Null olmayan ilk elemanı boş kalan matris
elemanına ata.
Daha sonra bu matris elemanını
fill_in_the_empty_space fonksiyonuna
gönder.

Harfler yok olduğunda boşlukların dolması
için **fill_in_the_empty_space2** adında
recursive fonksiyon tanımla. (Bu fonksiyon
zor mod için kullanılıyor)
Fonksiyona gönderilen harf buz harf ise
etrafındaki yapay buzları normal harfe
dönüştür.

Bu harfin bulunduğu matris elemanına null
ata.

Resmini "black" yap.
Üst satırda aynı sütundaki elemanı kontrol
et.

Null değilse o elemanı boş kalan matris
elemanına ata.
Eğer bu harf orijinal buzsa ona temas eden
harfleri yapay buz yap.
Yapay buzsa etrafında orijinal buz yoksa
tipini normal yap.

Veri tabanı bağlantısı için gerekli değişkenleri
tanımla.

```
Void onCreate{  
Veri tabanı için kullanıcı değişkenlerini  
tanımla.  
For(10 kere)  
    For(8 kere)  
        Harflerin olacağı 10x8  
matrisi başlatma  
Girilen kelime, puan ve buton atama
```

Alfabe.java için nesne oluşturma ve atama
create_start_count_down_timer(5000,5000
)

```

For(10 kere)
    For(8 kere)
        İki kere tıklandı mı kontrol et
        Tıklandıysa
            İf normal harfse
                Harfin seçilmemiş versiyonunu koy
            Else if buz harfse
                Harfin seçilmemiş versiyonunu koy
            Else
                Yarı buz harfin seçilmemiş
                versiyonunu koy

        Bir kere tıklandıysa ve üç kere
        tıklanmış normal harf ise
            Harfi al
            txt_chosen_word'e ekle

```

```

btn_confirm-> kelime onaylama{
    kelimeyi al ve küçük harfe çevir
    kelimenin üç harften büyük olduğunu
    kontrol et
        veri tabanına bağlan
        kelimenin ilk harfi olan koleksiyonu
    al

    tabloda kelime için query yap
    varsa: puanı harf puanına göre
    hesapla arayüzde puanı güncelle ve harfleri
    yok et ->kolay,zor
        yoksa: yanlış kelime sayısını artır.
    Kelimeyi temizle
    For(10 kere)
        For(8 kere)
            Normal harfse-> tıklanmamış normal
            harfe çevir
            Buz harfse-> tıklanmamış buz harfe
            çevir
            Yarı buz harfse-> tıklanmamış yarı
            buz harfe çevir

```

```

    İf yanlış harf sayısı 3'e çıktıysa
        // bir satır harf düşüyor
        İf(puan 100 olduğunda) 4 saniye
        Else if (puan 200 olduğunda) 3 saniye
        Else if (puan 300 olduğunda) 2 saniye
        Else (puan 400 olduğunda) 1 saniye

```

```

}

btn_remove-> kelimenin hepsini silme{
    (Tıklandığında)
        Kelime input kısmını sil
        For(10 kere)
            For(8 kere)
                Harf type'ını al
                İf normal harfse
                    Harfin seçilmemiş versiyonunu koy
                Else if buz harfse
                    Harfin seçilmemiş versiyonunu koy
                Else
                    Yarı buz harfin seçilmemiş
                    versiyonunu koy
            }
        }
}

```

```

btn_show_activity_point ->puan listesi
görüntüle{
    (Tıklandığında)
        Puan listesi arayüzüne geç
    }
}

```

```

onBackPressed->{
    (CountDownTimer'lar)
    start_ct null değilse->kapat
    move_ct null değilse->kapat
    game_ct null değilse->kapat
    ct null değilse->kapat
    ct2 null değilse->kapat
    ct3 null değilse->kapat
    ct4 null değilse->kapat
    }
}

```