

# YAZILIM LABORATUVARI 2

## 2. PROJE

Kocaeli Üniversitesi Bilgisayar Mühendisliği Bölümü  
Yazılım Laboratuvarı 2 - 2.Proje

Hakan AKGÜN  
180202103

### 1.Giriş

Bu doküman Yazılım Laboratuvarı 2 dersi 2.

Projesi için çözümü açıklamaya yönelik oluşturulmuştur. Dökümanda projenin tanımı, çözüme yönelik yapılan araştırmalar, kullanılan yöntemler, proje hazırlanırken kullanılan geliştirme ortamı ve kod bilgisi gibi programın oluşumunu açıklayan başlıklara yer verilmiştir.

Doküman sonunda projeyi hazırlarken kullanılan kaynaklar bulunmaktadır

### 2.Temel Bilgiler

Program XCode üzerinde Swift program dili ile yazılmıştır.

Mobil uygulama olarak oluşturulmuştur.

### 3.Proje Tanımı

Oyun 8x10'luk( yani 8 sütun 10 satır olacak) bir alan üzerinde kelime seçilerek anlamlı kelimeler oluşturulacaktır. Mobil programlama kodları Android veya İOS için geliştirilecektir. Oyun içeriği kelime bağlı bir oyundur. Bundan dolayı bir kelime havuzu oluşturulmalıdır.

#### 3.1.Proje İsterler

•Kelime sayısı 50.000 den fazla kelime içermesi gereklidir.

- Oyunun başlangıç anında harflerin düşüp oyunun başlama işlemi
- Puan hesaplanması işleminin doğru yapılması ve hesaplanarak puan tablosuna eklenmesi
- Bulunan kelimenin arayüz üstünde seçilmesi ve hızlı bir şekilde kontrolünün yapılıp hatalı veya doğru olduğunun tespit edilmesi
- Arayüz tasarımı ve harf geçişleri, oyunun son bulması

Arayüzler hakkında birkaç görsel;





- **ViewController.swift:** View Controller yaptığımız iOS uygulamalarının temel bileşenidir. Her uygulamada en az bir tane bulunur ancak bir çok uygulama birden fazla view controller'a sahiptir. View controller kullanıcı arayüzünü (UI) yönetir; bu arayüzler arasındaki geçişleri ve altındaki model adı verilen data'nın birbirleri arasındaki geçişini sağlar.

- **loadView():** Eğer view'ı storyboard kullanarak oluşturduysanız bu metodu kullanmanıza gerek yok. Ancak view'ı kod yazarak oluşturduysanız viewDidLoad() yerine bu metodu kullanabilirsiniz.

- **words.txt:** Kelime havuzudur. TDK yer alan kelimeler kullanılmıştır. Kelime sayısı 50.000 kelimeyi geçmektedir. Havuzdaki kelimeler en az 3 harflidir. Kelimeler anlamlıdır.

### 3.2 Yapılan Örnek Fonksiyonlar ve Bazı Source Code Files

**setupMatrix():** Ekranı matrix olarak bölüm harfleri alan method. ekranı 8X10 luk matrislere bölüp ekranın ayarlarını yapan, harf hücrelerini ayar yapan oluşturan methottur.

**letterTapped():** Harfleri ekranda yerleştirdiği random renkli hücrelere basma işlemini dokunma işlemini veren metottur. Basma işlemine göre nelerin yapılacağını belirtir.

**setupWordInput():** Labelleri, butonları, title itemlerini belirten canlılık veren methottur.

### Örnek Fonksiyon

```
func setupWordInput() {
    let screenSize = UIScreen.main.bounds.size
    let inputFrame = CGRect(x: padding, y: screenSize.height - 160.0, width:
        screenSize.width - padding * 2.0, height: 50.0)

    // Word input field
    wordTextField = UITextField(frame: inputFrame)
    wordTextField.placeholder = "..."
    wordTextField.textAlignment = .center
    //wordTextField.borderStyle = .roundedRect
    wordTextField.font = UIFont.systemFont(ofSize: 22.0, weight: .regular)
    wordTextField.textColor = UIColor.white
    wordTextField.backgroundColor = UIColor.darkGray
    //wordTextField.clearButtonMode = .whileEditing
    wordTextField.autocorrectionType = .no

    self.addSubview(wordTextField)

    let labelFrame = CGRect(x: padding, y: screenSize.height - 700.0, width:
        screenSize.width - padding * 2.0, height: 50.0)
    titleLabel = UILabel(frame: labelFrame)
    titleLabel.text = "KELİME OYUNU"
    titleLabel.font = UIFont(name: "GillSans-Bold", size: 40.0)
    titleLabel.textAlignment = .center
    titleLabel.textColor = UIColor.white
}
```

```
func kelimeBul(_ enteredWord: String) -> Bool {
    if let path = Bundle.main.path(forResource: "words", ofType: "txt") {
        do {
            let data = try String(contentsOfFile: path, encoding: .utf8)
            let kelimeListesi = data.components(separatedBy: .newlines)
            return kelimeListesi.contains(enteredWord)
        } catch {
            print(error.localizedDescription)
            return false
        }
    }
    return false
}
```

## 4. Kısaca Program Çalışma Yöntemi

Ekranı 8X10 matrisli olacak şekilde bölüyor sonrasında random harfleri alarak random renkli ayrılan hücrelere yerleştiriyor. harflerin altında kontrol butonu ve silme butonu da mevcut. Sonrasında harfleri seçip harflerin yok olması gerçekleştiriliyor seçilen harflerle oluşan kelime yazılıyor ve kelime havuzunda var mı kontrolü olur.

## 5. Akış Şeması





