

swiftkodlamadersleri.com



Xcode nesneleri

Text Field Özellikleri ve Kullanımı

İçindekiler

- ❖ Bölüm-1 : Text Field'ın tanıtılması,
- ❖ Bölüm-2 : Text Field ekleme ve utilities ile düzenleme,
- ❖ Bölüm-3 : Text Field ekleme ve kod ile düzenleme,
- ❖ Bölüm-4 : Klavye gizleme yöntemleri,
- ❖ Bölüm-5 : Text giriş sınırlamaları yöntemleri,
- ❖ Bölüm-6 : Text Field olayları (events) kullanımı,

Carrier

5:03 PM



Kullanıcı Adı

Password



Kullanıcı Adı Giriniz

0 - 10

1

2

ABC

3

DEF

4

GHI

5

JKL

6

MNO

7

PQRS

8

TUV

9

WXYZ

0



Bölüm 1

Text Field'ın tanıtılması

Bölüm 1'in Amacı

Amaç : Text Field nesnesini tanımak, kullanım amacını anlamak, arayüze ekleme yöntemleri, Referans dosyalarına ulaşma ve özelliklerine ulaşma yolları

Örnek Text Field çeşitleri

Text



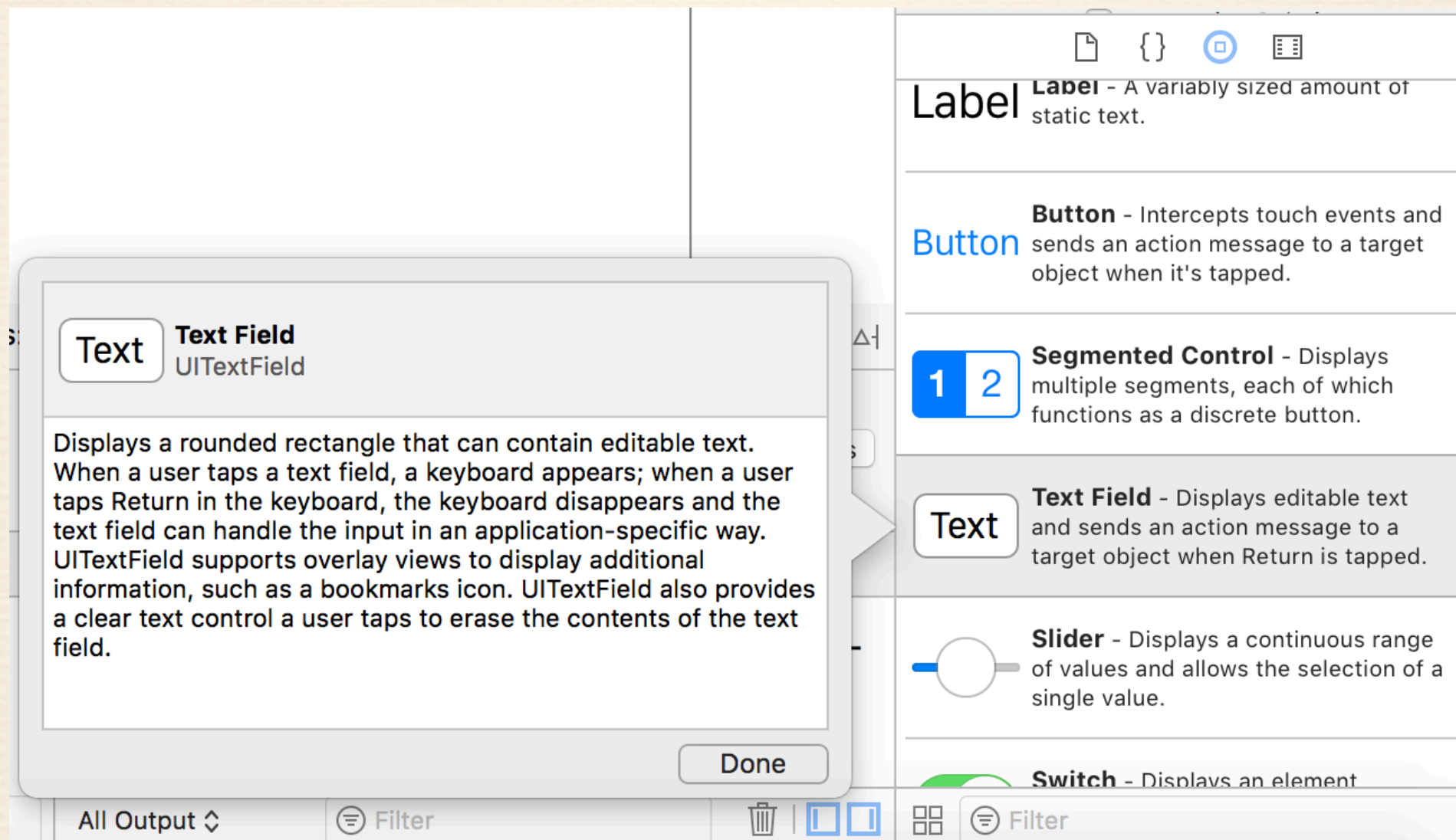
Kullanıcı Adı Giriniz

Text Field Nedir?

- ❖ Arayüzünüzde düzenlenebilir bir metin alanı gösteren bir nesne.
- ❖ Hem **Outlet** hemde **Action** özelliği vardır.
- ❖ Outlet ile; özelliklerine ulaşılır. (Renk, Font vb.)
- ❖ Action ile; Text Field'ın seçilmesi yada kontrolü kaybetmesine bağlı olarak eylemler gerçekleştirir.
- ❖ Kullanıcıya, ekran klavyesini kullanarak metin (string) tabanlı girdiler yapmasını sağlarsınız,
- ❖ Klavye'yi düz metin, e-posta, numara vb. gibi birçok farklı giriş türü için yapılandırabilirsiniz.

Text Field Arayüze Nasıl Eklenir?

- ❖ Arayüzünüze programsal olarak veya Arayüz Oluşturucu kullanarak düğmeler ekleyebilir ve yapılandırabilirsiniz.

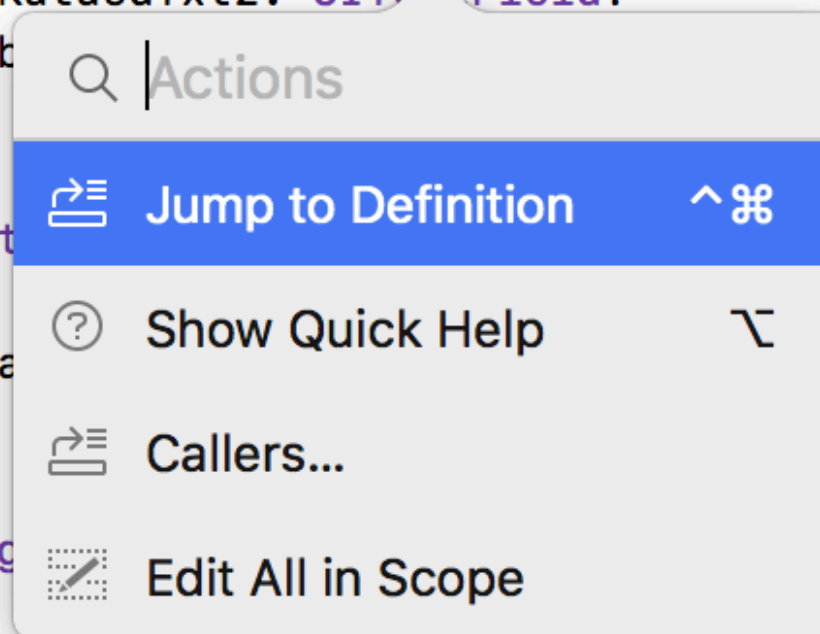


En önemli başvuru kaynağımız

❖ <https://developer.apple.com/documentation/uikit/uitextfield>

ve

```
9  import UIKit
10
11  class ViewController: UIViewController, UITextFieldDelegate {
12
13      @IBOutlet weak var textKutusuTxt: UITextField!
14      @IBOutlet weak var textKutusuTxt2: UITextField!
15      @IBOutlet weak var sayiLabel: UILabel!
16
17      let ilkSayı : Int = 0
18      let istenenUzunluk : Int
19
20      override func viewDidLoad() {
21          super.viewDidLoad()
22
23          textKutusuTxt2.delegate = self
24
```



Bölüm 2

Text Field ekleme ve utilities ile düzenleme,

Bölüm 2'nin Amacı

Amaç : Arayüze Text Field nesnesini eklemek, Connection etmek ve Utilities üzerinden değişiklikler yapmak

- ❖ Yapacağımız değişiklikler sabit atama sınıfına girecek,
- ❖ Unutma! Kod satırında yapılacak olan atama Utilities bölümünden önce gelir,
- ❖ Düzenlemeler “**kullanıcıAdıTxt**” ve “**şifreTxt**” isimli Text kutuları üzerinde izlenecektir.

Carrier 5:03 PM

Kullanıcı Adı

Password

 Kullanıcı Adı Giriniz

0 - 10

1

2
ABC

3
DEF

4
GHI

5
JKL

6
MNO

7
PQRS

8
TUV

9
WXYZ

0





Bölüm 3

TextField ekleme ve kod ile düzenleme,

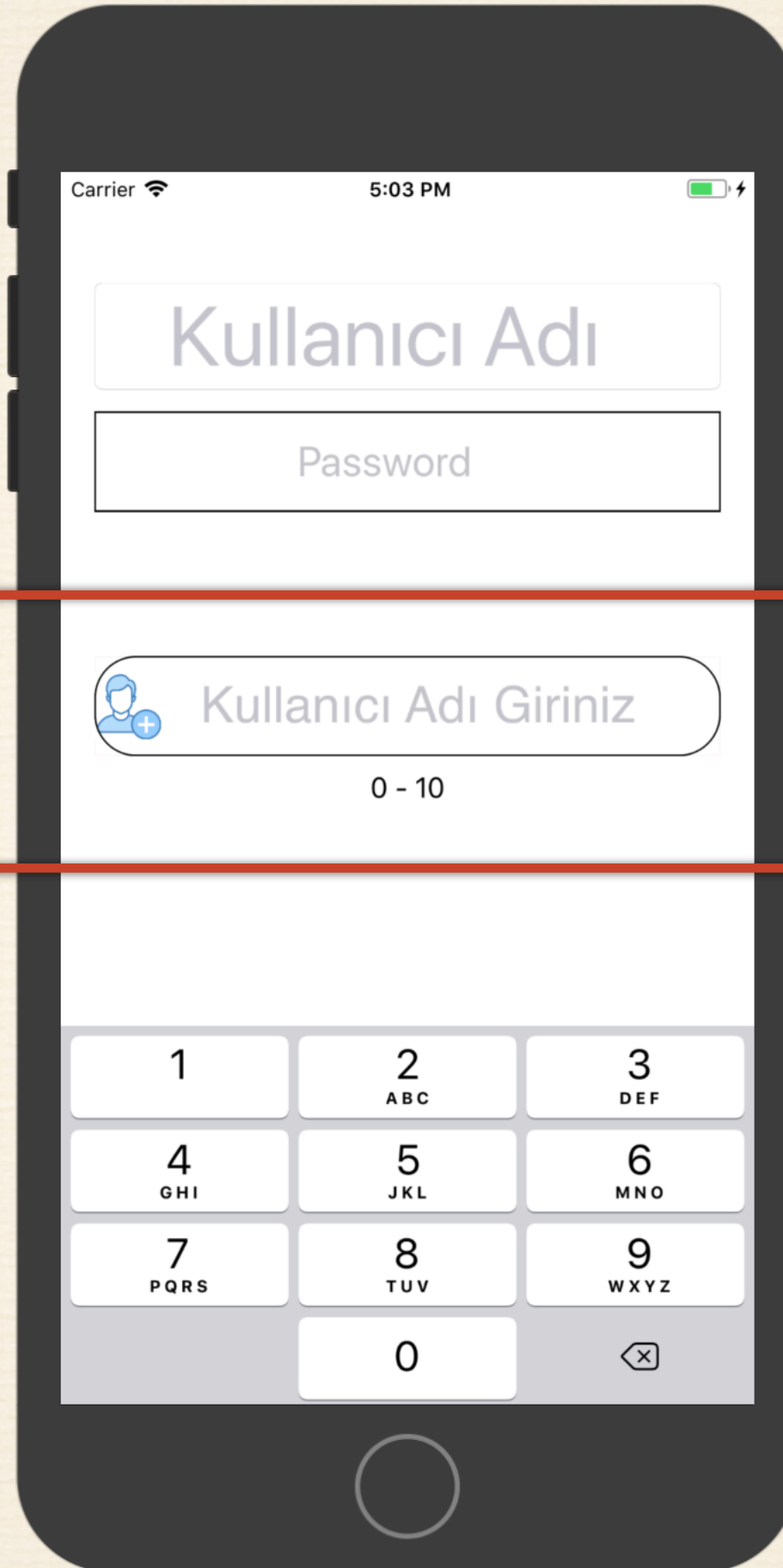
Bölüm 3'ün Amacı

Amaç : Arayüze Text Field nesnesini eklemek, Connection etmek ve kod ile özelliklerine ulaşmak ve atamalar yapmak

❖ Kod ile Text Field'ın;

- Text'i ve placeholder'ı,
- Çerçeve rengi ve kalınlığı,
- Alpha'sı,
- Sil butonu durumu,
- Klavye tipi,
- Yazının durumu (Hepsi Büyük, hepsi Küçük, Kelimenin ilk harfı, Cümlelerin ilk Harfı vb.)
- Köşeleri,
- Text konumu, rengi, Fontu,
- Kutu içine resim eklenmesi ve konumunun düzenlenmesi,

❖ Düzenlemeler “**resimliKullanıcıAdıTxt**” isimli Text Kutusu üzerinde izlenecektir.



Bölüm 4

Klavye gizleme yöntemleri,

Bölüm 4'ün Amacı

Amaç : Text Field nesnesi kontrolü kazandığında klavyenin gösterilmesi ve kontrol kaybedildiğinde klavyenin otomatik olarak gizlenmesini sağlamak

- ❖ Klavyeyi göstermek;
- ❖ Kullanıcı, Text alanına dokunduğunda, text alanı otomatik olarak “becomeFirstResponder()” durumuna geçer ve klavye görünür duruma gelir.
- ❖ Klavyeyi gizlemek;
- ❖ İlgili Text Field’a “Editing Did End” ile bir action oluşturarak, “resignFirstResponder ()” yöntemini çağırarak, klavyeden kapatabilirsiniz,
- ❖ “touchesBegan” fonksiyonunu kullanarak düzenlemenin bittiğini “endEditing” ile uygulamaya bildirebilirsiniz. Bu fonksiyon ile herhangi bir Text Field’ın kontrolü kaybetmesi durumunda klavye otomatik olarak gizlenir.

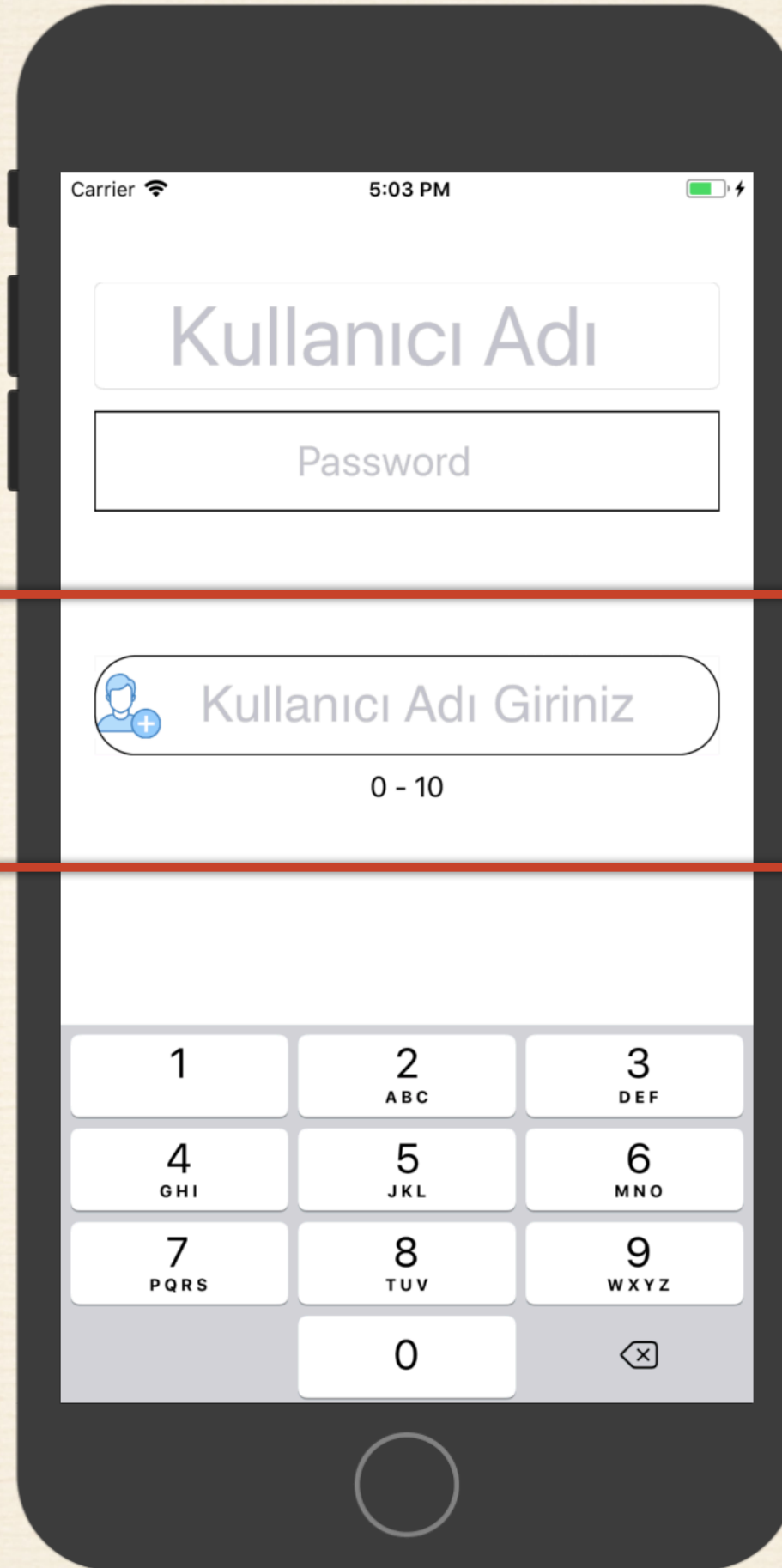


Bölüm 5

Text giriş sınırlamaları yöntemleri,

- ❖ Text Field'a ait protokol fonksiyonlara ulaşmak için Delegate ve UITextFieldDelegate tanımlaması yapılmalıdır.
- ❖ Kısıtlama için “Class ViewController :” satırına “UITextFieldDelegate” protokolü eklenmelidir.
- ❖ Ve ayrıca ilgili Text Field için “Delegate” tanımlanmalıdır.
- ❖ Düzenlemeler “resimliKullanıcıAdıTxt” ve “kullanıcıAdıSayLbl” isimli Text Field'ı ve Label'ı üzerinde izlenecektir.





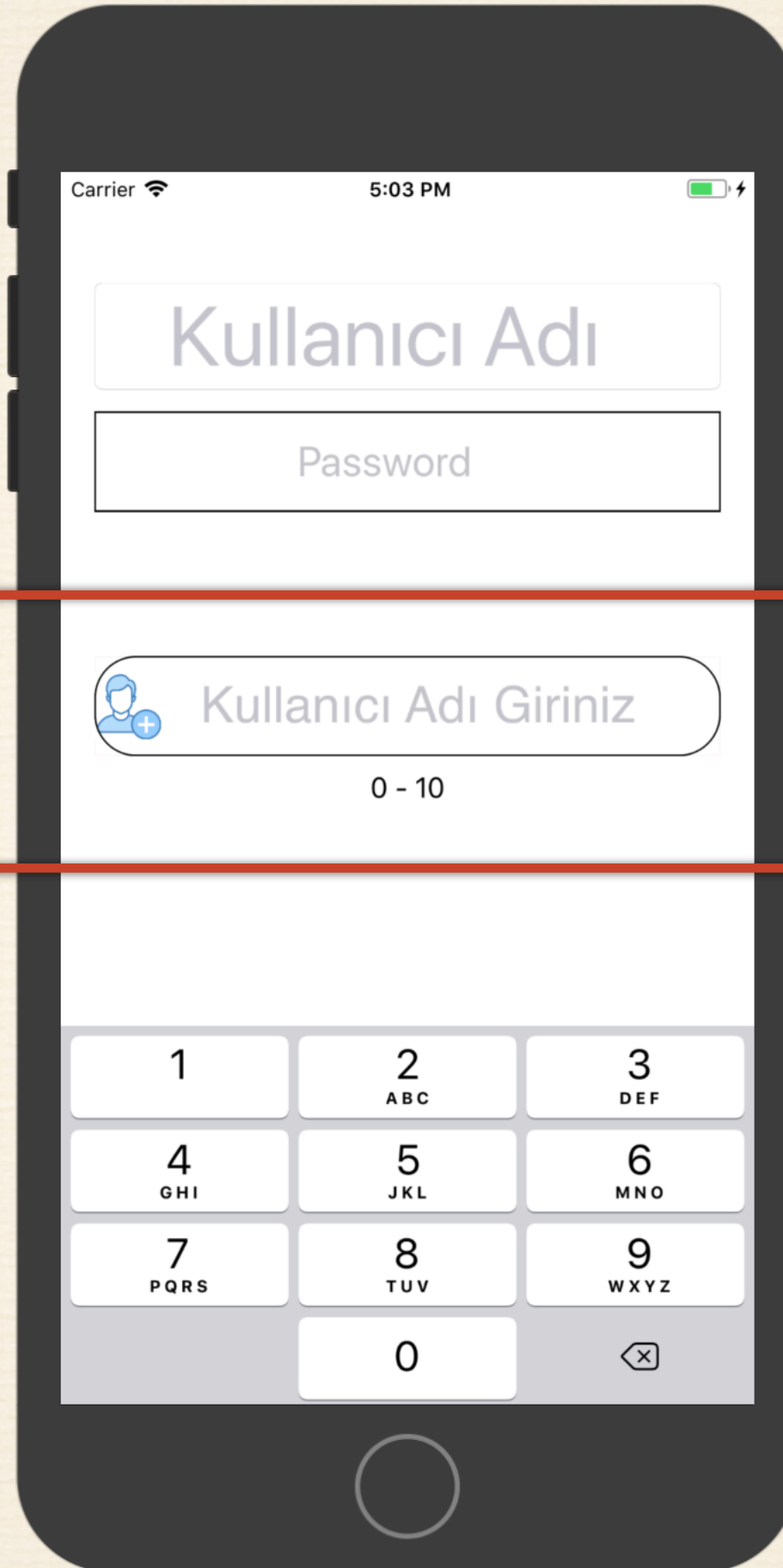
Bölüm 6

TextField olayları (events) kullanımı,

❖ Text Field event'lerini tanıma;

- DidEndOnExit,
- editingChanged,
- editingDidBegin,
- editingDidEnd,
- touchDown,
- touchUpInside,

❖ Düzenlemeler “**resimliKullanıcıAdıTxt**” isimli Text Kutusu üzerinde izlenecektir.



Events

❖ **DidEndOnExit**

UITextField nesnesi kontrolü kaybettiğinde gerçekleşen olaydır.

❖ **editingChanged**

UITextField nesnesinde bir değişiklik gerçekleştiğinde gerçekleşen olaydır.

❖ **editingDidBegin**

UITextField nesnesinde kendi sınırlarını girerek düzenleme oturumunu başlatan bir dokunuş. Text Field nesnesine dokunulduğunda gerçekleşir.

Events

- ❖ **editingDidEnd**

UITextField nesnesindeki bir düzenlemenin sonlandığını ve kontrolü kaybettiğinde gerçekleşen olaydır.

- ❖ **touchDown**

UITextField nesnesine bastığınız anda tetiklenir. Basışı bırakmanıza gerek yok.

- ❖ **touchUpInside**

UITextField nesnesine basıldığında tetiklenmez. Text Field'a basıldıktan sonra parmağın kaldırılması ile gerçekleşir.

