

Desarrollo de aplicaciones para dispositivos iOS – Un cliente Twitter

Pablo Romeu – CEU-UCH

Moncada, Enero 2015

Ciclo de Vida

- Lo que debes saber para empezar:
 - Introducción al iPhone SDK
 - Xcode
 - Interface Builder
 - Simulador
 - Introducción a Objective-C
 - Desarrollar un cliente Twitter

Ciclo de Vida

- Introducción al iPhone SDK
 - Xcode
 - Interface Builder
 - Simulador
- Introducción a Objective-C

Ciclo de Vida

- *Introducción al iPhone SDK*
 - Xcode
 - Interface Builder
 - Simulador
- Introducción a Objective-C

Introducción al iPhone SDK

- Se divide en varias partes, las más interesantes son:
 - Xcode
 - Interface Builder
 - Simulador
 - Instruments

Introducción al iPhone SDK

- **Xcode**
 - Interfaz gráfico de programación
 - Se utiliza tanto para Mac OS X como para iOS
 - Tiene algunas facilidades muy interesates como autocompletado de información, ayuda, etc.

Introducción al iPhone SDK

- ***Interface Builder***

- Interfaz gráfico de diseño
- Nos permite diseñar la vista de la aplicación
- Además, permite asignar gráficamente qué métodos (funciones) se ejecutarán dado un determinado evento sobre un objeto de la vista
- Ej: ¿Qué función ejecutar al pulsar un botón? ¿Y al mover un slider?

Introducción al iPhone SDK

- ***Simulador***
 - Permite probar programas
 - Hay algunas cosas que NO puede probar (ej. Cámara)
- ***Instruments:***
 - Permite hacer profiling de programas.
 - Comprobar rendimiento, llamadas a base de datos, etc.

Ciclo de Vida

- Introducción al esquema de Licencias, AppStore, etc.
- Introducción al iPhone SDK
 - Xcode
 - Interface Builder
 - Simulador
- ***Introducción a Objective-C***

Introducción a Objective-C

- Lenguaje Orientado a Objetos
- Superconjunto de C
 - Cualquier cosa hecha en C funciona!!!
- Sintaxis –supuestamente- clara y simple.
- Como todo, tiene ventajas:
 - Cosas que nos hacen más fácil la vida (@property)
 - Gestión de memoria automatizada (nada de alloc,free)
- Apple acaba de sacar un lenguaje nuevo llamado SWIFT. De momento, *Beta*.

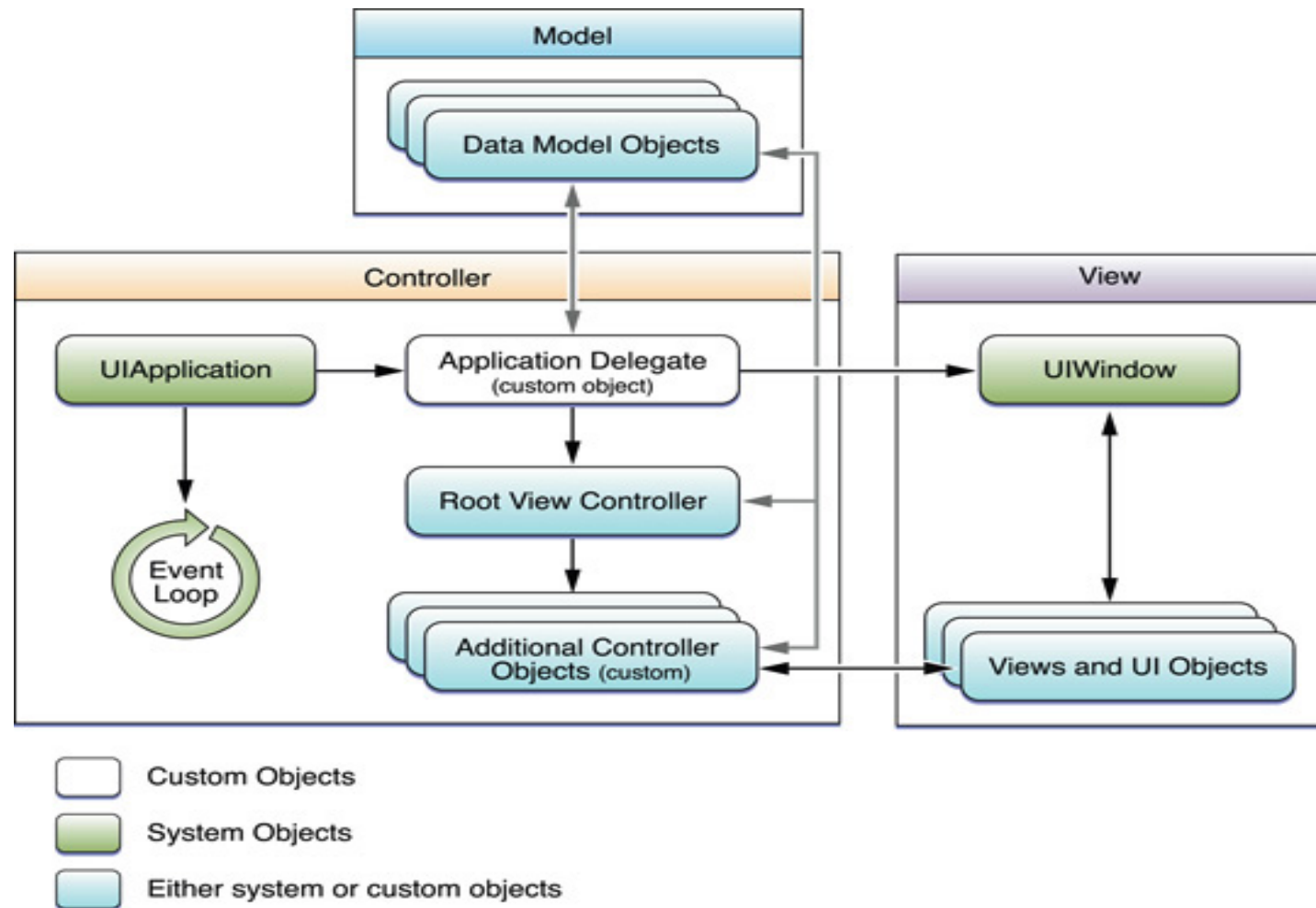
Introducción a Objective-C

- **MAXIMA SUPREMA:** En Objective-C \rightarrow “[]” = “.”
- Invocar un método:
 - *miObjeto.miMétodo;*
 - Es lo mismo que: *[miObjeto miMétodo];*
- **RECOMENDACIÓN:** Utilizar sintaxis propia de Objective-C.
O sea “[....];”
- Hay alguna excepción

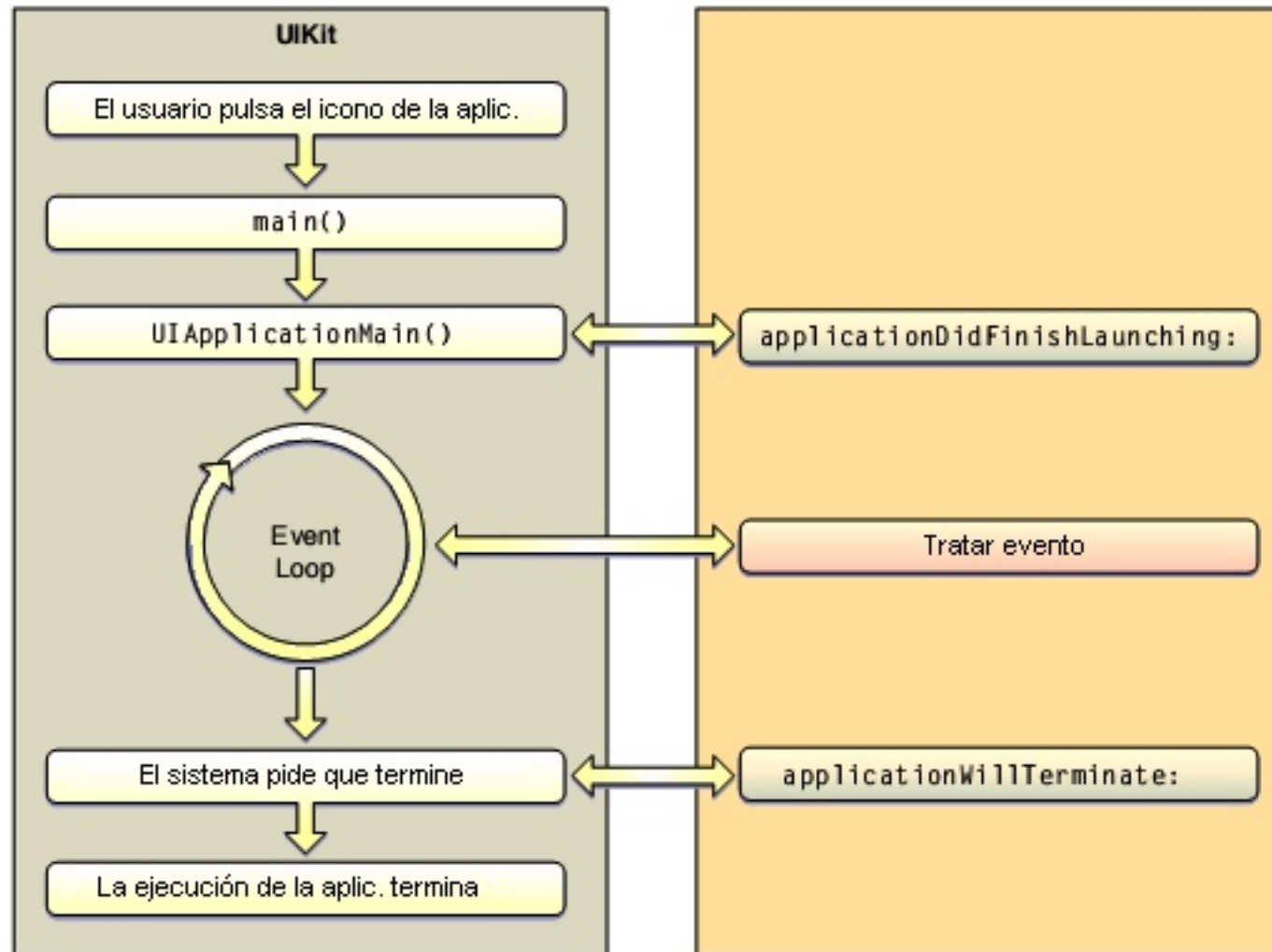
¿Qué hay que saber para empezar?

- Objective-C
- Foundation Framework (clases base)
- Patrones de diseño
 - Memory Management
 - Delegación
 - Target-Action
 - MVC (Modelo-Vista-Controlador)
- UIKit Framework (clases interfaz gráfica)
 - Ficheros NIB (Interface Builder)
 - Tipos de vistas (navegación, tablas,)

Modelo-Vista-Controlador



Ciclo de vida de una aplicación



Desarrollar un cliente Twitter

EMPECEMOS

Desarrollar un cliente Twitter

Nuestra App funcionará con nuestro cliente:

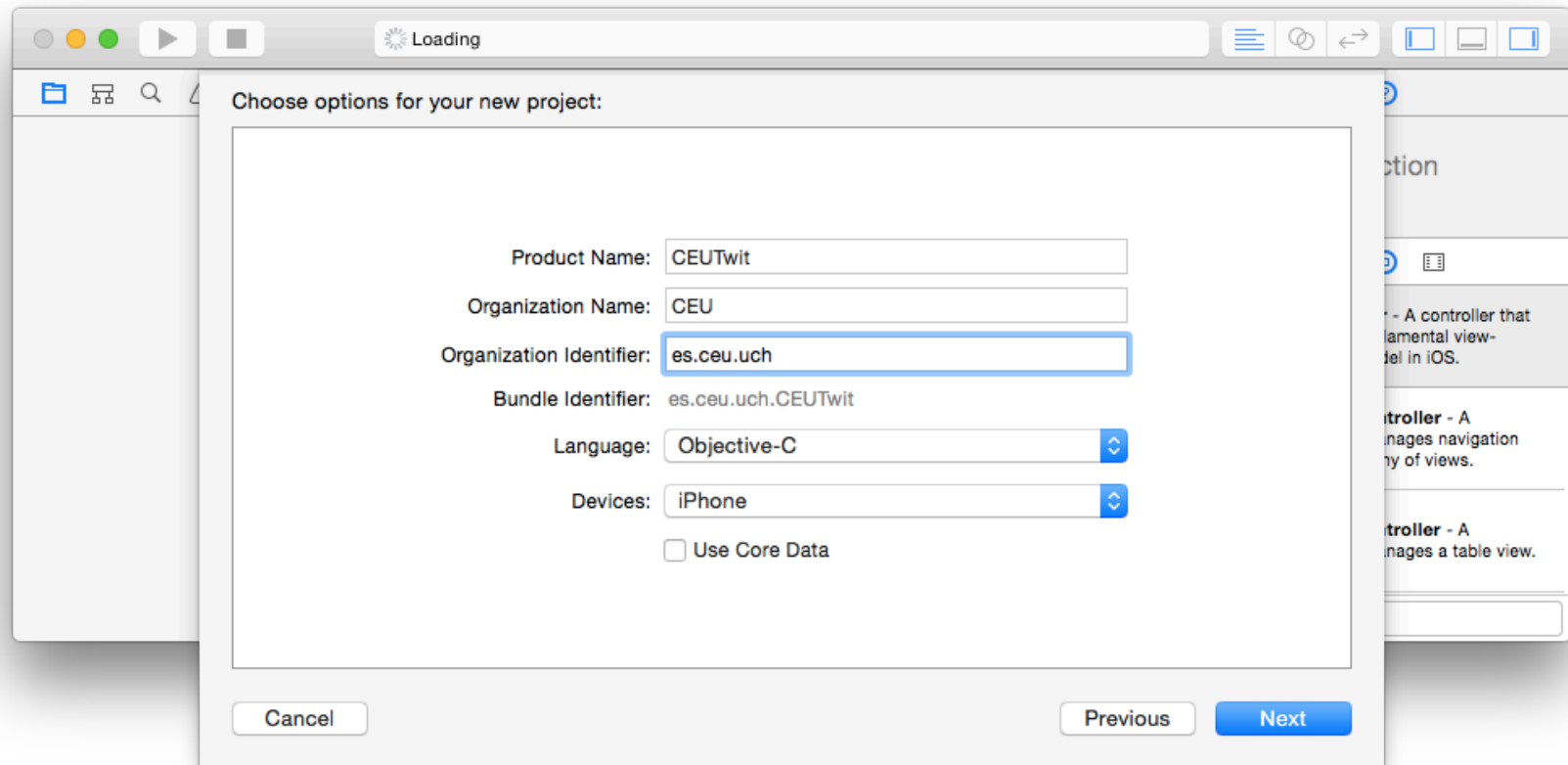
- Necesitamos tener configurado un usuario en el iPhone/iPad
- Podremos publicar y podremos ver nuestro timeline.

Desarrollar un cliente Twitter

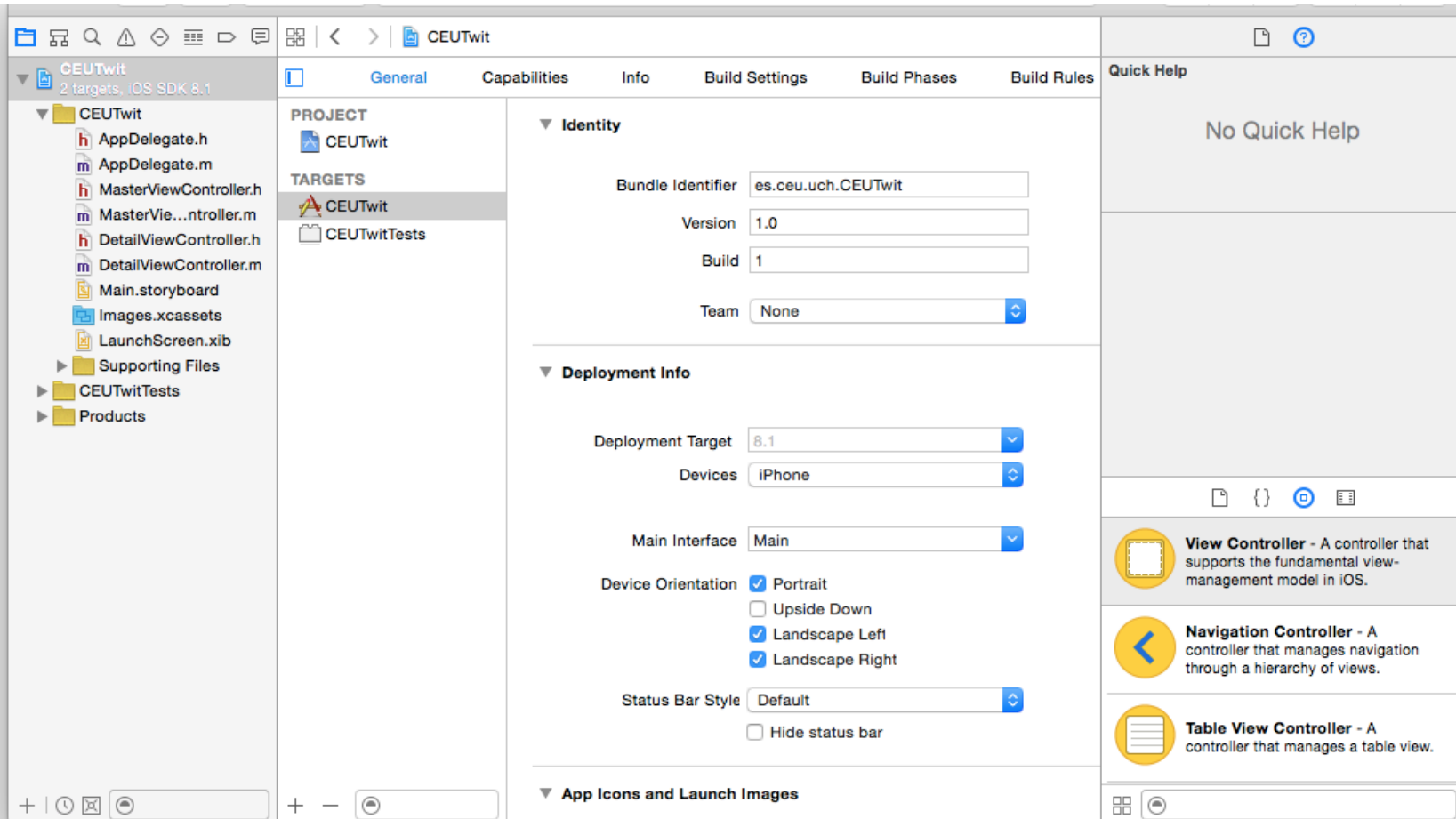
Vamos a abrir un proyecto nuevo:

- Se llamará CEUTwit
- Será de tipo “master-detail” app
- Sin CoreData

Desarrollar un cliente Twitter



Desarrollar un cliente Twitter



Desarrollar un cliente Twitter

- Ahora vamos a añadir el Social.Framework y Accounts
- Luego crearemos un fichero para trabajar con el modelo (twitter).
- Crearemos el código necesario y lo probaremos
- Diseñaremos la interfaz gráfica
- Enlazaremos la interfaz gráfica (vista) con el modelo mediante controladores.

Desarrollar un cliente Twitter

- Añadir social framework y Accounts framework
- Crear una clase CEUTwit
- Y añadir el código.

Desarrollar un cliente Twitter

- En CEUTwit.h:
 - (BOOL)userHasAccessToTwitter;
 - (void)fetchTimelineForUser:(NSString *)username withBlock:
(void(^)(NSError*,id))completionBlock;

Desarrollar un cliente Twitter

- En CEUTwit.m:

```
#import <Social/Social.h>  
#import <Accounts/Accounts.h>
```

Desarrollar un cliente Twitter

- En CEUTwit.m, antes de @implementation:

```
@interface CEUTwit ()  
@property (nonatomic, strong) ACAccountStore*accountStore;  
@end
```


Desarrollar un cliente Twitter

- En CEUTwit.m, dentro de @implementation:
 - ```
(instancetype)init{
 if (self = [super init]){
 _accountStore=[[ACAccountStore alloc] init];
 }
 return self;
}
```

## *Desarrollar un cliente Twitter*

- Seguimos:

```
- (BOOL)userHasAccessToTwitter
{
 return [SLComposeViewController
 isAvailableForServiceType:SLServiceTypeTwitter];
}
```

## Desarrollar un cliente Twitter

Seguimos (ver listado):

```
- (void)fetchTimelineForUser:(NSString *)username withBlock:(void (^)(
 NSError*,id))completionBlock
{
 // Step 0: Check that the user has local Twitter accounts
 if ([self userHasAccessToTwitter])
 {

 // Step 1: Obtain access to the user's Twitter accounts
 ACAccountType *twitterAccountType =
 [self.accountStore accountTypeWithAccountTypeIdentifier:
 ACAccountTypeIdentifierTwitter];

 ...
 }
}
```

## *Desarrollar un cliente Twitter*

Vamos a probarlo. En MasterViewController.m

```
#import "CEUTwit.h"
```

```
@interface MasterViewController ()
```

```
@property NSMutableArray *objects;
```

```
@property CEUTwit*ceuTwit;
```

```
@end
```

## *Desarrollar un cliente Twitter*

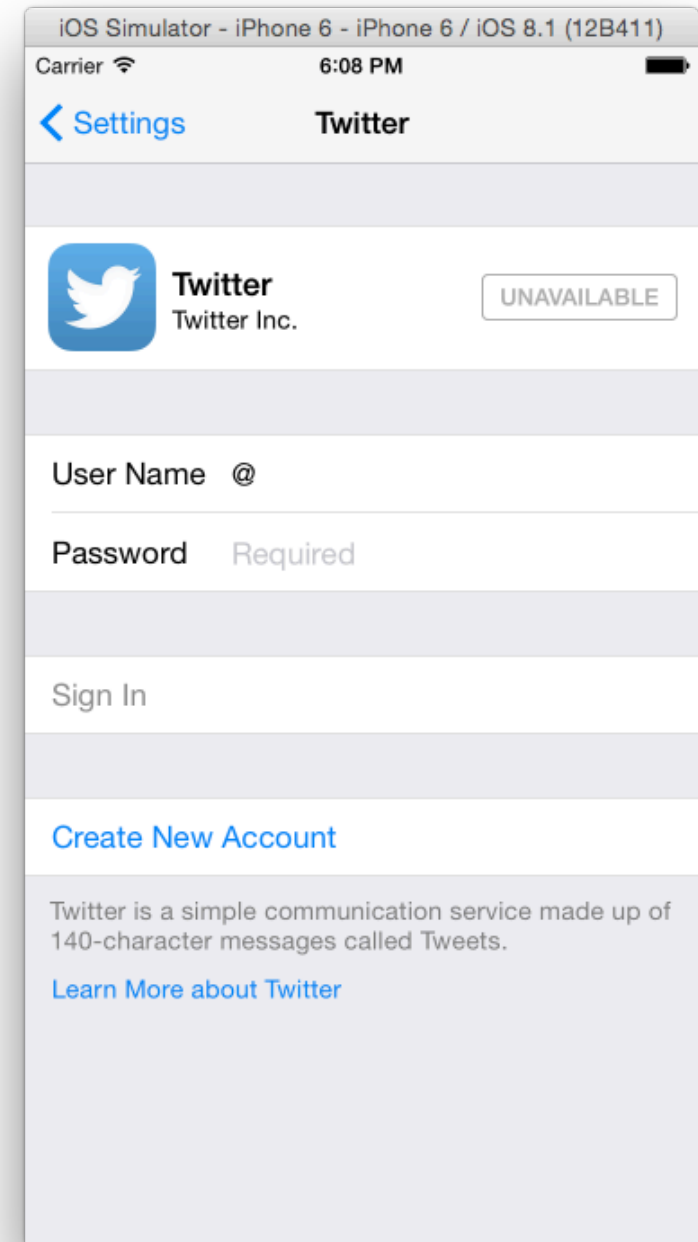
- Dentro del método – `(void)viewDidLoad`

```
self.ceuTwit=[[CEUTwit alloc] init];
[self.ceuTwit fetchTimelineForUser:@"pabloromeu" withBlock:^(NSError
*error, id twits) {
 NSLog(@"twits received");
}];
```

Poned el nombre del usuario del que queráis mostrar el timeline

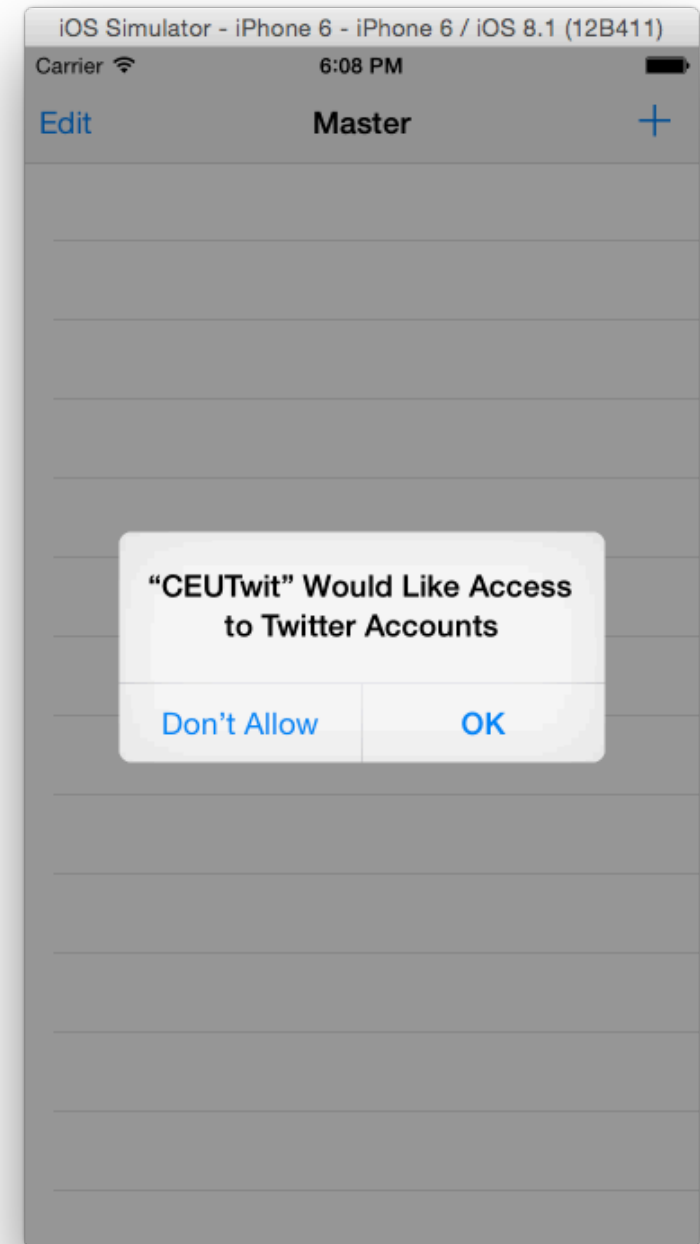
## *Desarrollar un cliente Twitter*

- Y ejecutamos (Comando+R). Es necesario poner nuestro twitter en la configuración



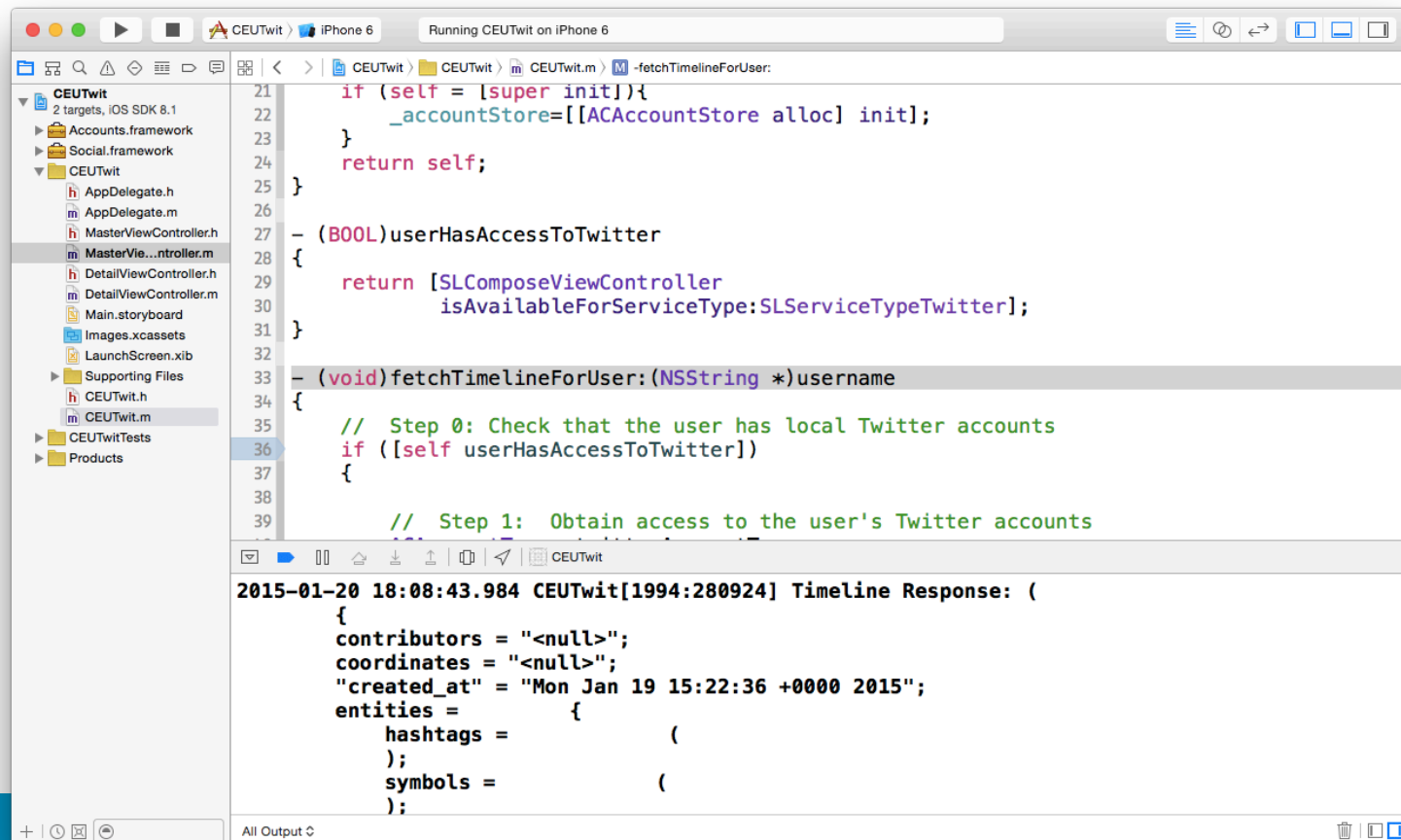
## *Desarrollar un cliente Twitter*

- Una vez tenemos el cliente aceptado, volvemos a ejecutar y damos a permitir acceder a las cuentas (Comando+R)



## Desarrollar un cliente Twitter

- En el Xcode se abrirá la consola y nos mostrará los twits:



```
21 if (self = [super init]){
22 _accountStore=[[ACAccountStore alloc] init];
23 }
24 return self;
25 }
26
27 - (BOOL)userHasAccessToTwitter
28 {
29 return [SLComposeViewController
30 isAvailableForServiceType:SLServiceTypeTwitter];
31 }
32
33 - (void)fetchTimelineForUser:(NSString *)username
34 {
35 // Step 0: Check that the user has local Twitter accounts
36 if ([self userHasAccessToTwitter])
37 {
38
39 // Step 1: Obtain access to the user's Twitter accounts
40
41 // ... (code continues) ...
42
43 NSLog(@"Timeline Response: (
44 {
45 contributors = "<null>";
46 coordinates = "<null>";
47 "created_at" = "Mon Jan 19 15:22:36 +0000 2015";
48 entities =
49 {
48 hashtags =
49 (
50);
51 symbols =
52 (
53);
54 };
55 };
56 }
57);
58 }
59 }
```

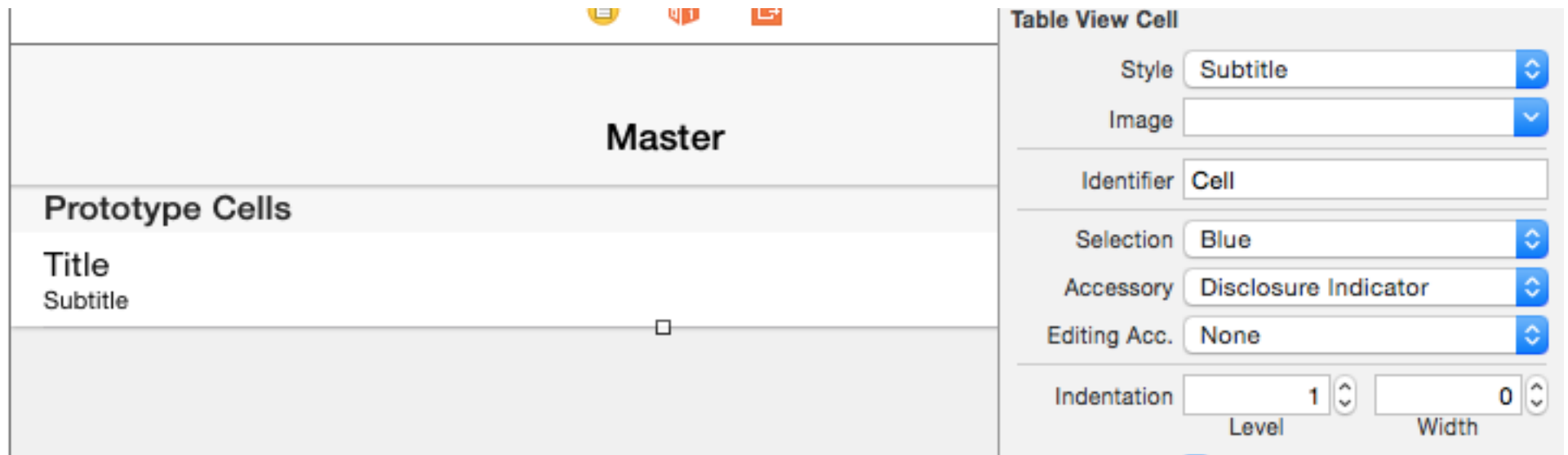
2015-01-20 18:08:43.984 CEUTwit[1994:280924] Timeline Response: (

```
{
 contributors = "<null>";
 coordinates = "<null>";
 "created_at" = "Mon Jan 19 15:22:36 +0000 2015";
 entities =
 {
 hashtags =
 (
);
 symbols =
 (
);
 };
}
```



## *Desarrollar un cliente Twitter*

- Vamos a diseñar la celda. En este caso sencilla



## *Desarrollar un cliente Twitter*

- Ahora enlazaremos los datos con la interfaz gráfica

```
#import <Social/Social.h>
```

```
@interface MasterViewController ()
```

```
@property NSMutableArray *objects;
```

```
@property NSMutableDictionary*images;
```

```
@property CEUTwit*ceuTwit;
```

```
@end
```

## *Desarrollar un cliente Twitter*

```
- (void)viewDidLoad {
 [super viewDidLoad];
 UIBarButtonItem *addButton = [[UIBarButtonItem
alloc]
initWithBarButtonSystemItem:UIBarButtonSystemItemComp
ose target:self action:@selector(insertNewObject:)];
 self.navigationItem.rightBarButtonItem =
addButton;

 self.ceuTwit=[[CEUTwit alloc] init];
 [self reloadTweets];
}
```

## Desarrollar un cliente Twitter

```
- (void)reloadTweets{
 [self.ceuTwit
fetchTimelineForUserWithBlock:^(NSError *error, id
twits) {
 NSLog(@"twits received");
 self.objects=[twits mutableCopy];
 self.images=[NSMutableDictionary new];
 dispatch_async(dispatch_get_main_queue(), ^{
 // Foreground sync code
 [self.tableView reloadData];
 });
}];
}
```

## Desarrollar un cliente Twitter

```
- (void)insertNewObject:(id)sender {
 if ([SLComposeViewController
 isAvailableForServiceType:SLServiceTypeTwitter]) {
 SLComposeViewController *mySLComposerSheet =
 [SLComposeViewController
 composeViewControllerForServiceType:SLServiceTypeTwitter];
 [mySLComposerSheet setInitialText:@"Twit de prueba desde
 @uchceu!"];
 [mySLComposerSheet addURL:[NSURL URLWithString:@"http://
 www.uchceu.es"]];
 [mySLComposerSheet
 completionHandler:^(SLComposeViewControllerResult result) {
 [self reloadTweets];
 }];
 [self presentViewController:mySLComposerSheet animated:YES
 completion:nil];
 }
}
```

## Desarrollar un cliente Twitter

```
- (void)insertNewObject:(id)sender {
 if ([SLComposeViewController
 isAvailableForServiceType:SLServiceTypeTwitter]) {
 SLComposeViewController *mySLComposerSheet =
 [SLComposeViewController
 composeViewControllerForServiceType:SLServiceTypeTwitter];
 [mySLComposerSheet setInitialText:@"Twit de prueba desde
 @uchceu!"];
 [mySLComposerSheet addURL:[NSURL URLWithString:@"http://
 www.uchceu.es"]];
 [mySLComposerSheet
 completionHandler:^(SLComposeViewControllerResult result) {
 [self reloadTweets];
 }];
 [self presentViewController:mySLComposerSheet animated:YES
 completion:nil];
 }
}
```

## *Desarrollar un cliente Twitter*

¡Y ESTO ES TODO!

¡Gracias!