



POLITÉCNICA

ETSIT
UPM

dit
UPM

Desarrollo de Apps para iOS

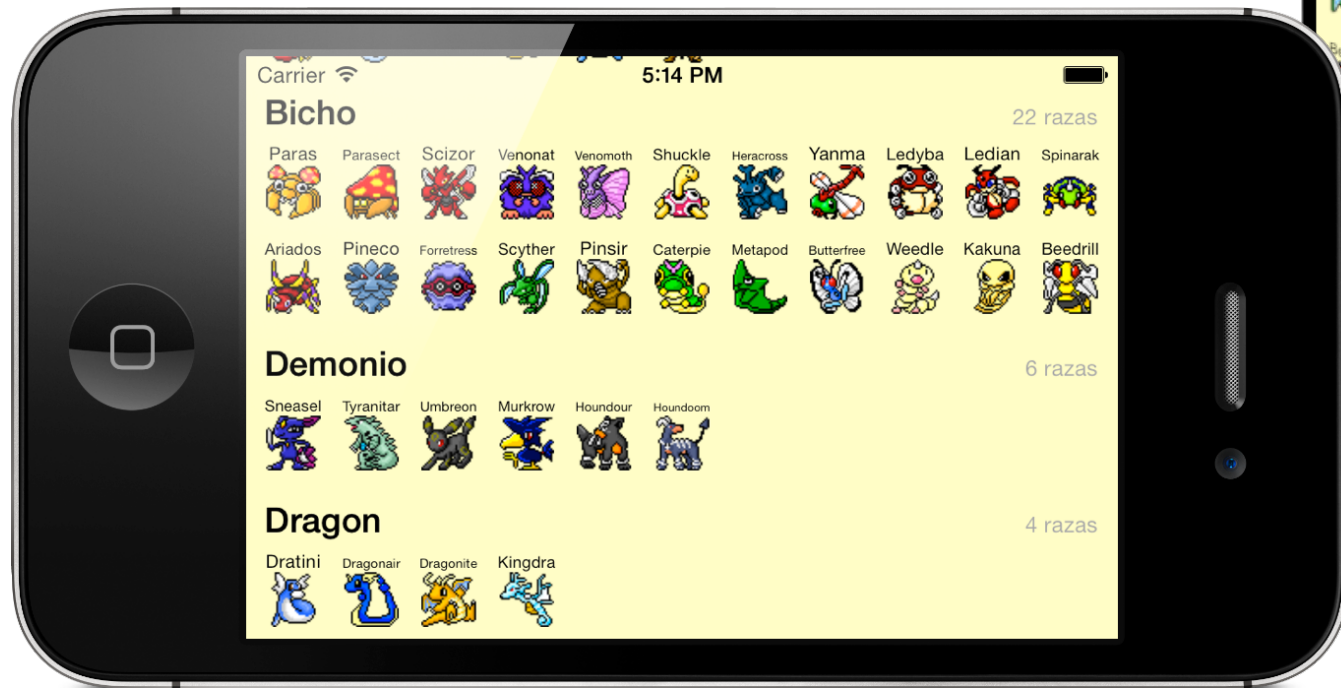
Collection Views

IWEB,LSWC 2013-2014
Santiago Pavón

ver: 2014.03.29

Características de las Collection Views

- Muestran los datos usando celdas dispuestas con diferentes layouts.
 - Actualmente solo se soporta Flow Layout.
 - Coloca las celdas en una parrilla completando líneas horizontal o verticalmente.
- Las celdas se puede agrupar en secciones.
 - Las secciones pueden tener un vistas complementarias
 - haciendo el papel de cabecera y pie de las secciones.
- El contenido de las celdas es siempre personalizado.
- Realiza un uso eficiente de celdas y vistas complementarias.
 - Reutilizando las no visibles.



Clases y Protocolos

- Algunas clases:
 - **UICollectionView**: la vista que contiene las celdas.
 - **UICollectionViewCell**: la celda con los datos.
 - **UICollectionViewController**: un UIViewController conteniendo una UICollectionView ocupando todo el espacio.
 - **UICollectionViewFlowLayout**: clase del gestor de layout Flow.
 - **UICollectionViewLayoutAttributes**: clase con los atributos que aplican a las celdas o vistas complementarias.
- Algunos protocolos:
 - **UICollectionViewDataSource**: define el protocolo de obtención de datos del objeto dataSource.
 - **UICollectionViewDelegate**: define el protocolo para comunicarse con el delegado informando de sucesos.

UICollectionView

- Propiedades:

dataSource
delegate

collectionView

collectionViewLayout

allowsSelection
allowsMultipleSelection

- Algunos métodos:

- `initWithFrame:collectionViewLayout:`
- `registerClass:forCellWithReuseIdentifier:`
- `registerNib:forCellWithReuseIdentifier:`
- `registerClass:forSupplementaryViewOfKind:withReuseIdentifier:`
- `registerNib:forSupplementaryViewOfKind:withReuseIdentifier:`
- `dequeueReusableCellWithReuseIdentifier:forIndexPath:`
- `dequeueReusableSupplementaryViewOfKind:withReuseIdentifier:forIndexPath:`
- `setCollectionViewLayout:animated:completion:`
- `reloadData`
- `reloadSections:`
- `reloadItemsAtIndexPaths:`
- `numberOfSections`
- `numberOfItemsInSection:`
- `visibleCells`
- `insertItemsAtIndexPaths:`
- `moveItemAtIndexPath:toIndexPath:`
- `deleteItemsAtIndexPaths:`
- `indexPathsForSelectedItems`
- `selectItemAtIndexPath:animated:scrollPosition:`
- `deselectItemAtIndexPath:animated:`
- `indexPathForItemAtPoint:`
- `indexPathForCell:`
- `cellForItemAtIndexPath:`
- `scrollToItemAtIndexPath:atScrollPosition:animated:`
- etc . . .

UICollectionViewCell

- Propiedades:

collectionView

collectionView

collectionView

collectionView

collectionView

UICollectionViewDataSource

- (NSInteger) **collectionView:** (UICollectionView*)
 numberOfItemsInSection: (NSInteger)
- (NSInteger) **numberOfSectionsInCollectionView:** (UICollectionView*)
- (UICollectionViewCell*) **collectionView:** (UICollectionView*)
 cellForItemAtIndexPath: (NSIndexPath*)
- (UICollectionViewReusableView*) **collectionView:** (UICollectionView*)
 viewForSupplementaryElementOfKind: (NSString*)
 atIndexPath: (NSIndexPath*)

UICollectionViewDelegate

- Selección de celdas:
 - `collectionView:shouldSelectItemAtIndexPath:`
 - `collectionView:didSelectItemAtIndexPath:`
 - `collectionView:shouldDeselectItemAtIndexPath:`
 - `collectionView:didDeselectItemAtIndexPath:`
- Destacar celdas:
 - `collectionView:shouldHighlightItemAtIndexPath:`
 - `collectionView:didHighlightItemAtIndexPath:`
 - `collectionView:didUnhighlightItemAtIndexPath:`
- Eliminar vistas:
 - `collectionView:didEndDisplayingCell:forItemAtIndexPath:`
 - `collectionView:didEndDisplayingSupplementaryView:forElementOfKind:atIndexPath:`
- Transición entre layouts:
 - `collectionView:transitionLayoutForOldLayout:newLayout:`
- Gestión de acciones
 - `collectionView:shouldShowMenuForItemAtIndexPath:`
 - `collectionView:canPerformAction:forItemAtIndexPath:withSender:`
 - `collectionView:performAction:forItemAtIndexPath:withSender:`

UICollectionViewController

- Inicializador:

 - **initWithCollectionViewLayout:**

- Propiedades:

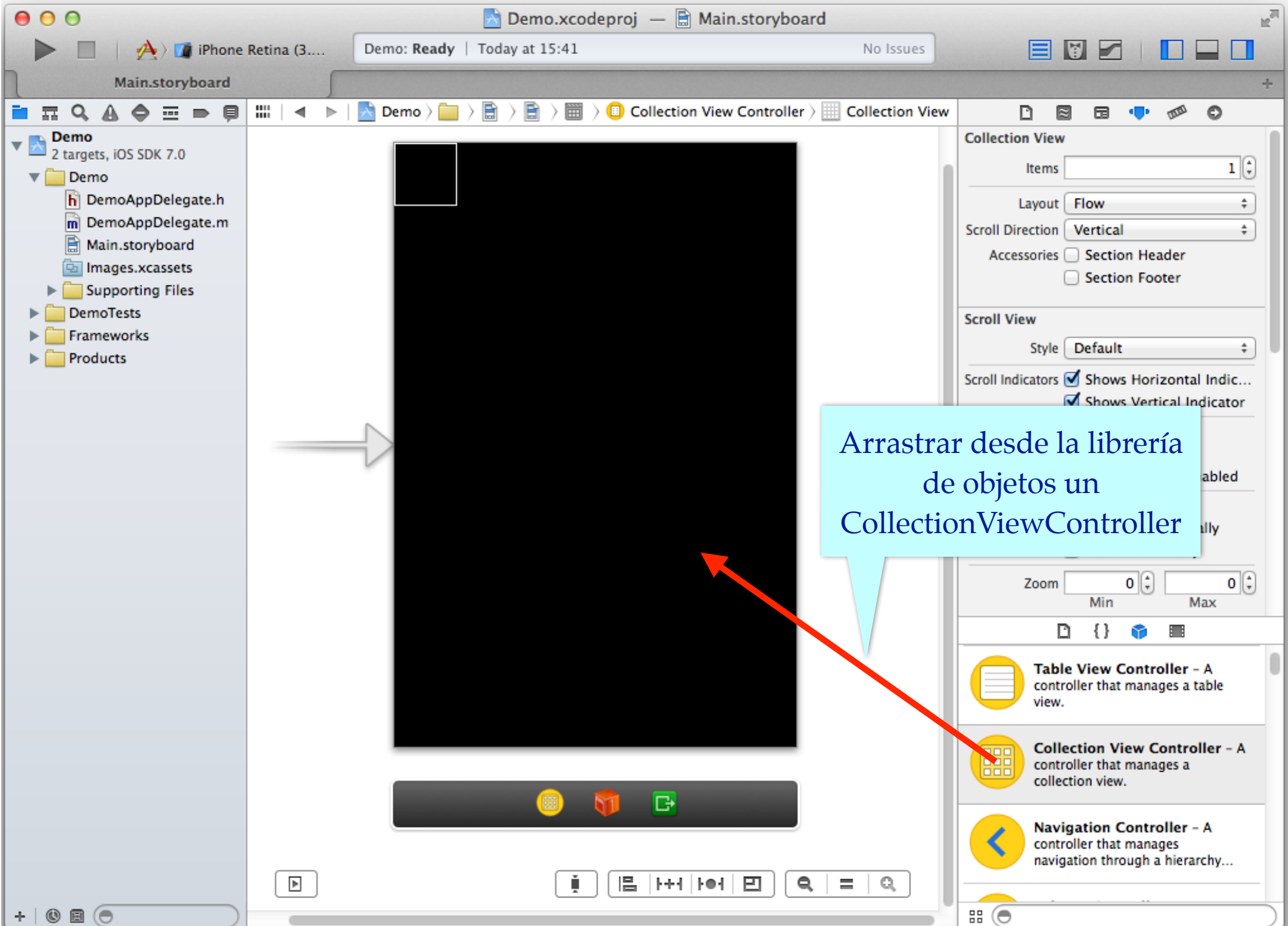
 - collectionView**

 - collectionViewLayout**

 - clearsSelectionOnViewWillAppear**

 - useLayoutToLayoutNavigationTransitions**

Demo



Arrastrar desde la librería de objetos un CollectionViewController

Choose options for your new file:

Crear una clase nueva para la nueva pantalla.
Derivada de UINavigationController.

Class

Subclass of

- Targeted for iPad
- With XIB for user interface

Cancel

Previous

Next

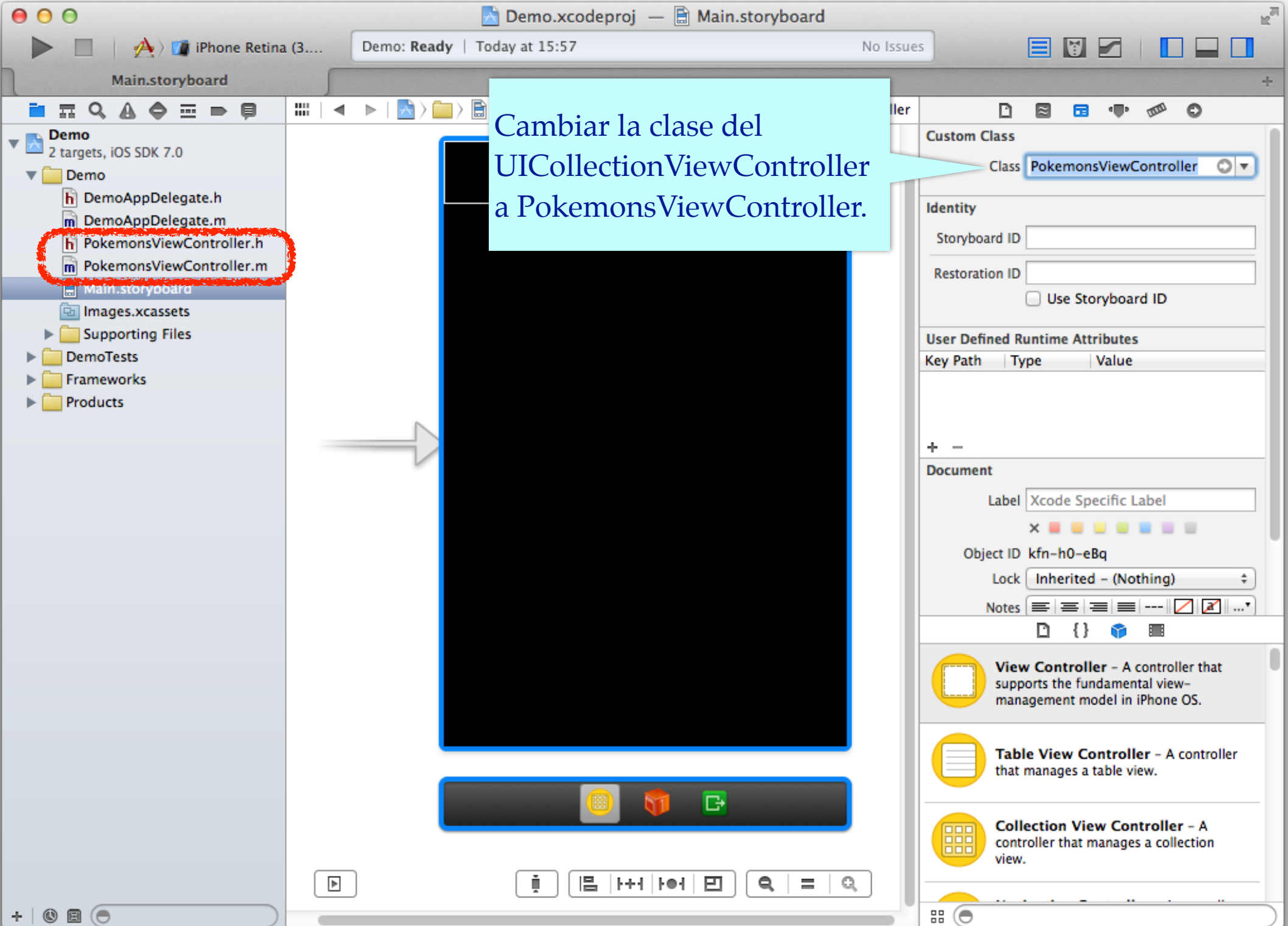


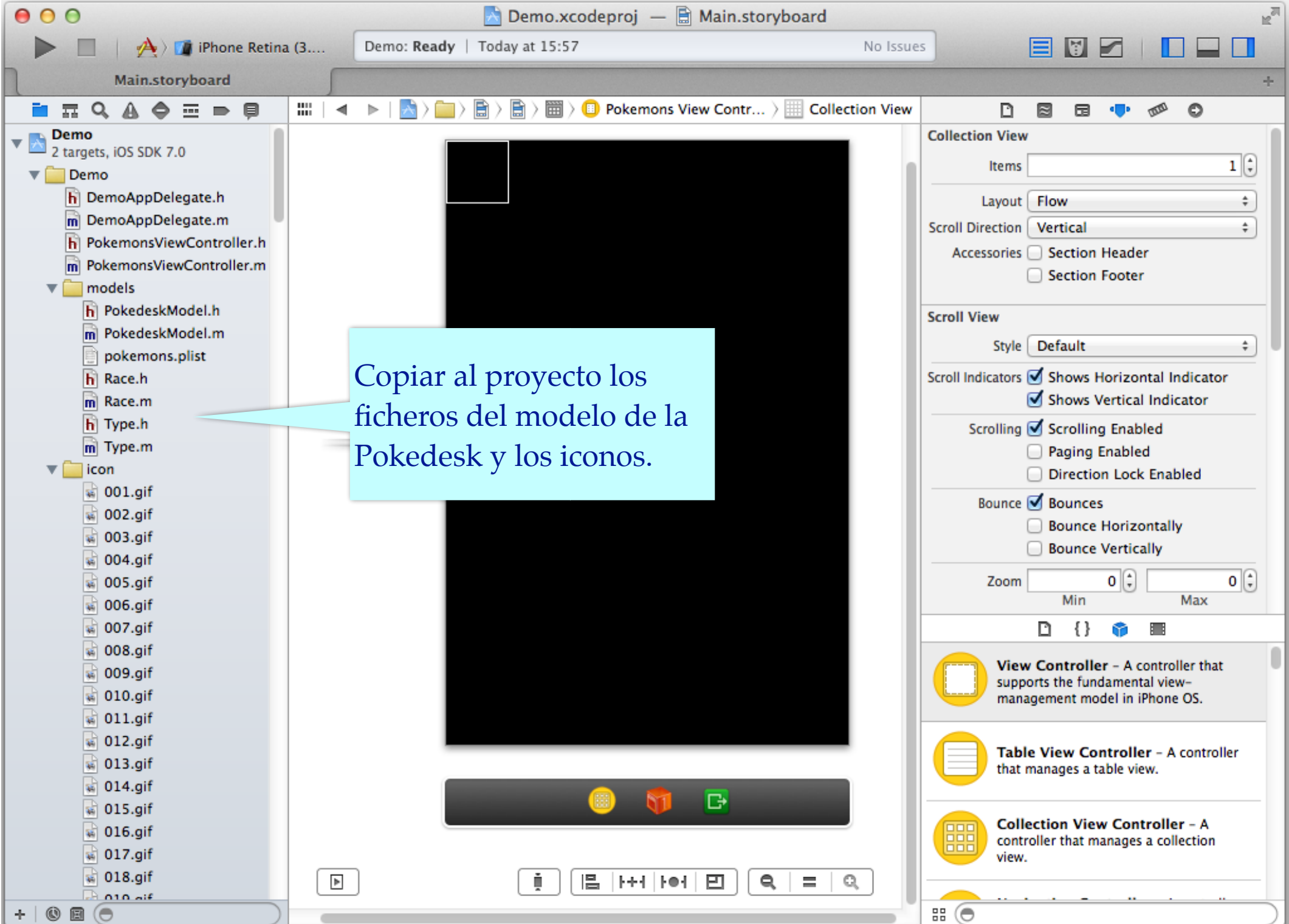
Collection View Controller - A controller that manages a collection view.



Navigation Controller - A controller that manages navigation through a hierarchy...

Cambiar la clase del
UICollectionViewController
a PokemonsViewController.





Copiar al proyecto los ficheros del modelo de la Pokedesk y los iconos.

```

Demo.xcodeproj - PokemonsViewController.m
Demo: Ready | Today at 15:57 | No Issues
PokemonsViewController.m
Demo
  2 targets, iOS SDK 7.0
  Demo
    DemoAppDelegate.h
    DemoAppDelegate.m
    PokemonsViewController.h
    PokemonsViewController.m
    models
      PokedeskModel.h
      PokedeskModel.m
      pokemons.plist
      Race.h
      Race.m
      Type.h
      Type.m
    icon
      001.gif
      002.gif
      003.gif
      004.gif
      005.gif
      006.gif
      007.gif
      008.gif
      009.gif
      010.gif
      011.gif
      012.gif
      013.gif
      014.gif
      015.gif
      016.gif
      017.gif
      018.gif
      019.gif

// PokemonsViewController.m
// Demo
//
// Created by Santiago Pavón on 28/10/13.
// Copyright (c) 2013 UPM. All rights reserved.
//

#import "PokemonsViewController.h"

#import "PokedeskModel.h"
#import "Race.h"
#import "Type.h"

@interface PokemonsViewController ()

@property (nonatomic, strong) PokedeskModel* pokedeskModel;

@end

@implementation PokemonsViewController

- (id)initWithNibName:(NSString *)nibNameOrNil bundle:(NSBundle *)nibBundleOrNil
{
    ...
}

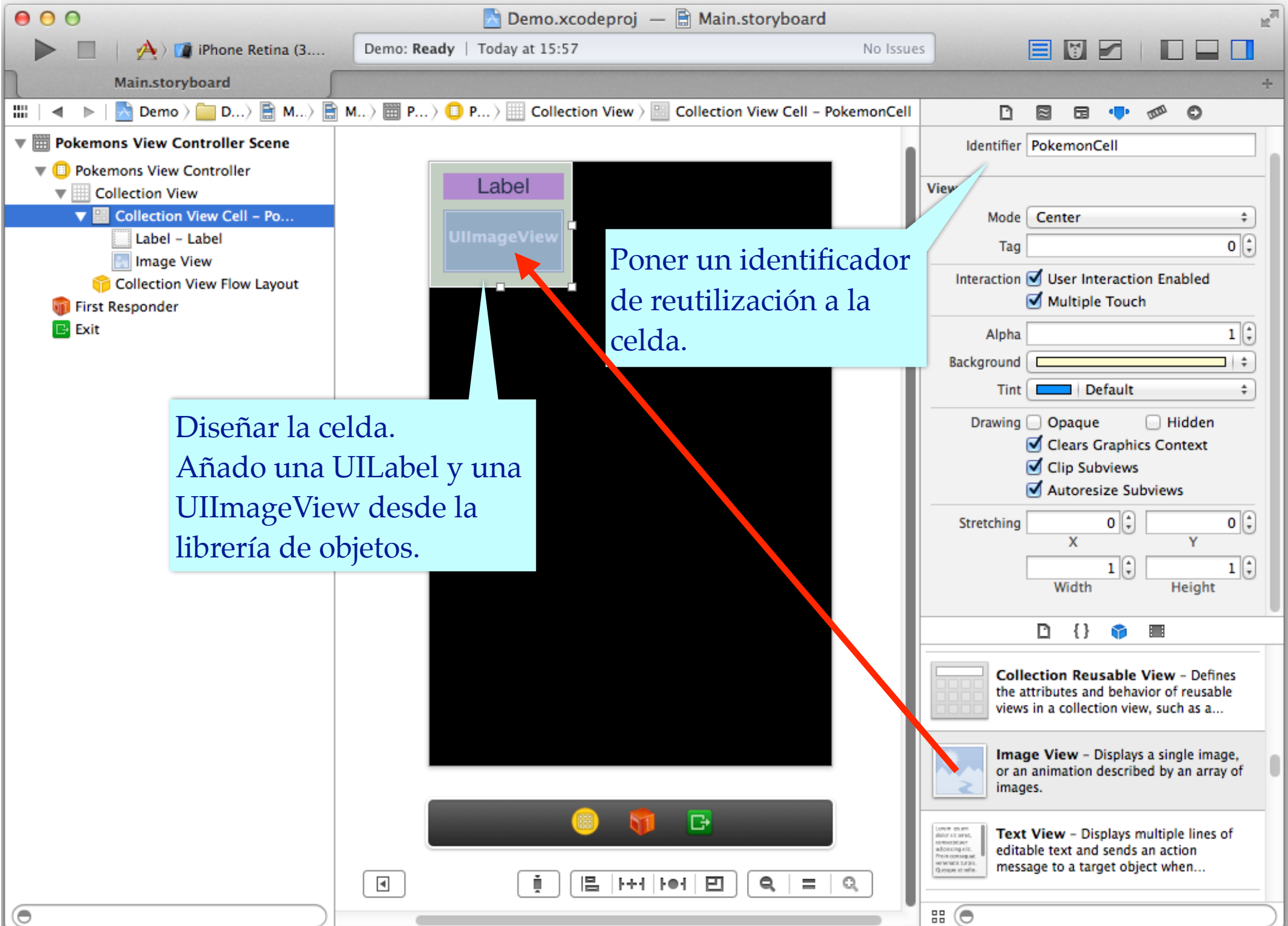
- (void)viewDidLoad
{
    [super viewDidLoad];
    // Do any additional setup after loading the view.

    self.pokedeskModel = [[PokedeskModel alloc] init];
}

- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
    // Dispose of any resources that can be recreated.
}

@end
```

Esta propiedad es el modelo.



Diseñar la celda.
Añado una UILabel y una UIImageView desde la librería de objetos.

Poner un identificador de reutilización a la celda.

Choose options for your new file:

Crear una clase nueva para las celdas.
Derivada de UICollectionViewCell.

Class

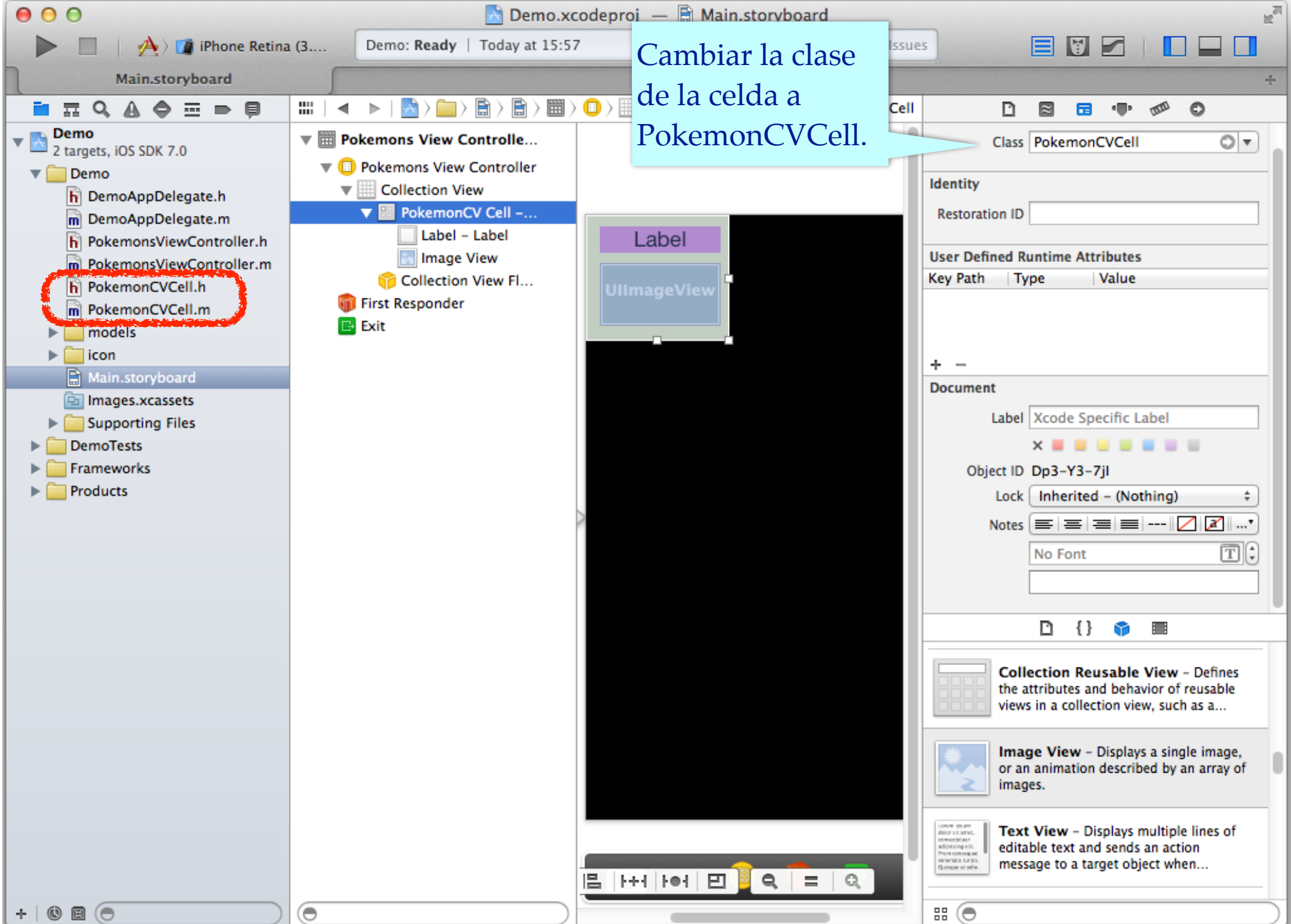
Subclass of

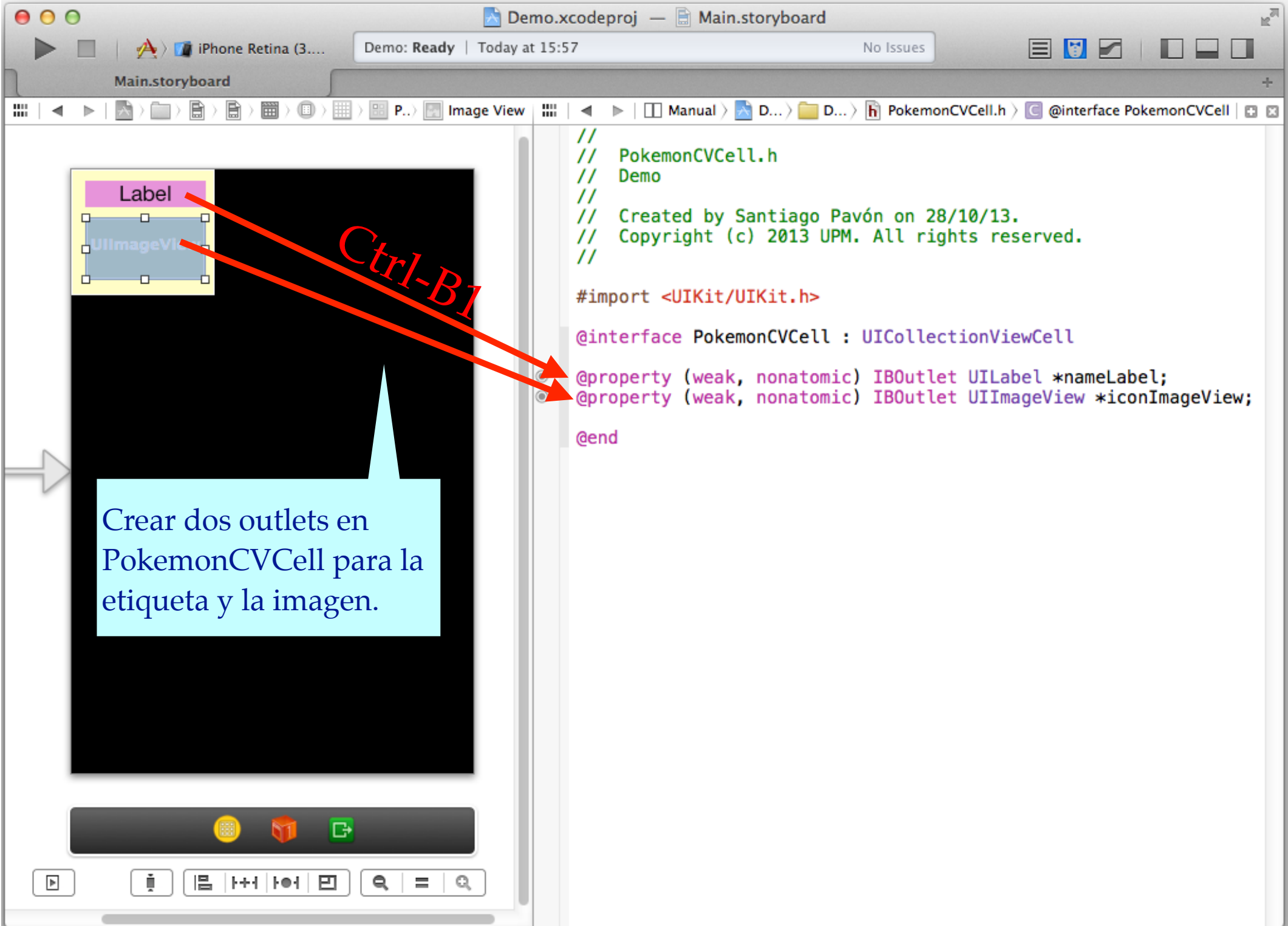
- Targeted for iPad
- With XIB for user interface

Cancel

Previous

Next





Ctrl-B1

Crear dos outlets en PokemonCVCCell para la etiqueta y la imagen.

```
//  
// PokemonCVCCell.h  
// Demo  
//  
// Created by Santiago Pavón on 28/10/13.  
// Copyright (c) 2013 UPM. All rights reserved.  
//  
#import <UIKit/UIKit.h>  
  
@interface PokemonCVCCell : UICollectionViewCell  
  
@property (weak, nonatomic) IBOutlet UILabel *nameLabel;  
@property (weak, nonatomic) IBOutlet UIImageView *iconImageView;  
  
@end
```

```

}

#pragma mark - Collection View Data Source

- (NSInteger) numberOfSectionsInCollectionView:(UICollectionView *)collectionView
{
    return [self.pokedeskModel.types count];
}

- (NSInteger) collectionView:(UICollectionView *)collectionView
    numberOfItemsInSection:(NSInteger)section
{
    Type * type = self.pokedeskModel.types[section];
    return [type.races count];
}

- (UICollectionViewCell *)collectionView:(UICollectionView *)collectionView
    cellForItemAtIndexPath:(NSIndexPath *)indexPath
{
    static NSString * cellId = @"PokemonCell";

    PokemonCVCell * cell = [collectionView dequeueReusableCellWithIdentifier:cellId
                                                                    forIndexPath:indexPath];

    NSInteger section = indexPath.section;
    NSInteger item = indexPath.item;

    Type * type = self.pokedeskModel.types[section];
    Race * race = type.races[item];

    cell.nameLabel.text = race.name;
    cell.imageView.image = [UIImage imageNamed:race.icon];

    return cell;
}

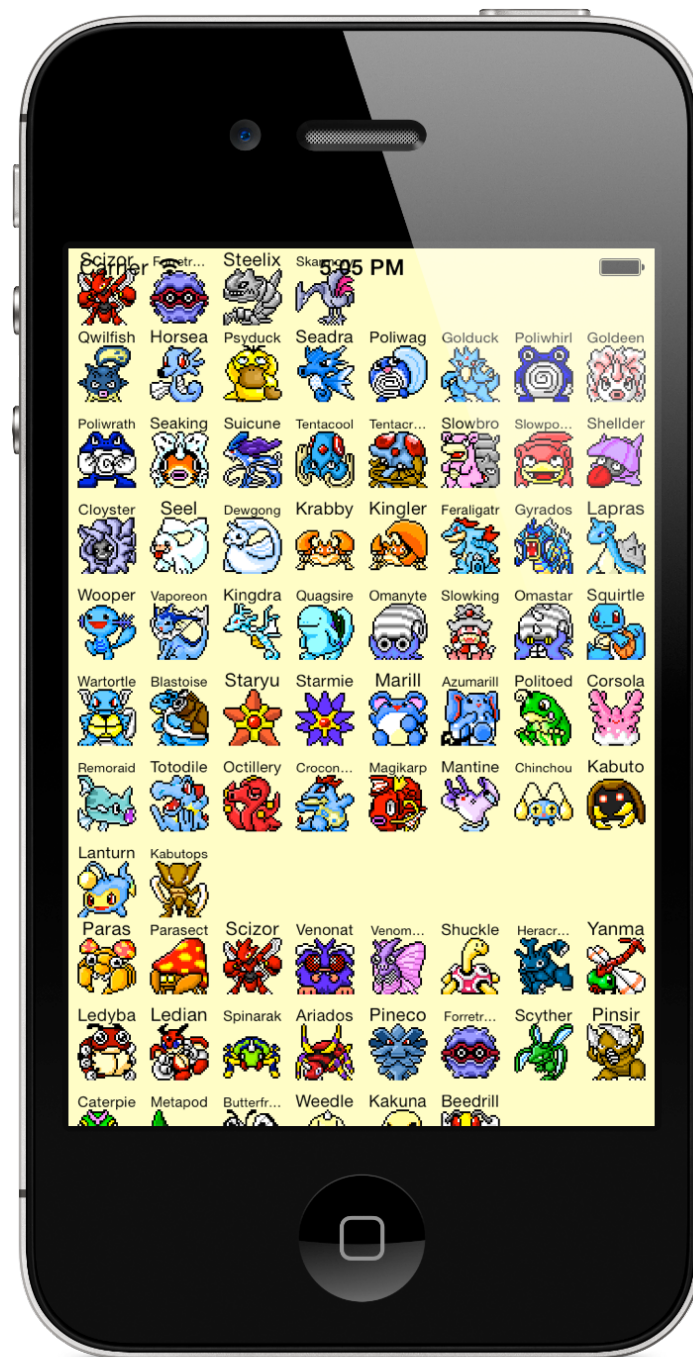
```

Métodos del Data Source

Añadir arriba:
#import "PokemonCVCell.h"



Retocar tamaños, colores, fonts en el inspector.

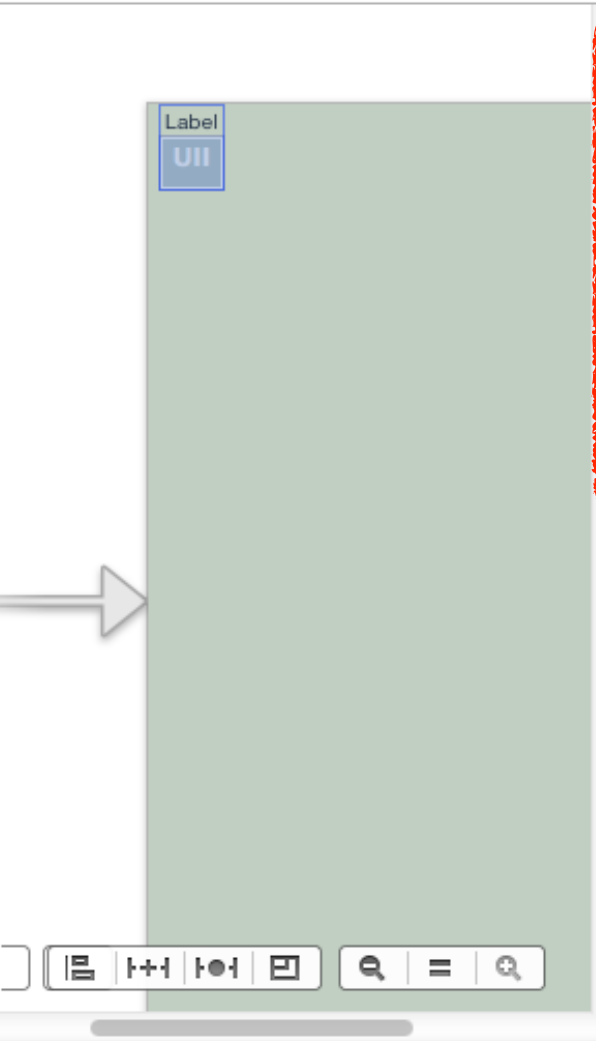


Demo
2 targets, iOS SDK 7.0

- Demo
 - DemoAppDelegate.h
 - DemoAppDelegate.m
 - PokemonsViewController.h
 - PokemonsViewController.m
 - PokemonCVCCell.h
 - PokemonCVCCell.m
 - models
 - icon
 - Main.storyboard**
 - Images.xcassets
 - Supporting Files
 - DemoTests
 - Frameworks
 - Products

Pokemons View Cont...

- Pokemons View Co...
 - Collection View
 - PokemonCV...
 - Label - Label
 - Image View
 - Constraints
 - Collection Vi...
 - First Responder
 - Exit



Collection View Size

Cell Size	Width: 32	Height: 42
Header Size	Width: 0,0	Height: 0,0
Footer Size	Width: 0,0	Height: 0,0
Min Spacing	For Cells: 5	For Lines: 5
Section Insets	Top: 0	Bottom: 0
	Left: 5	Right: 5

Scroll View Size

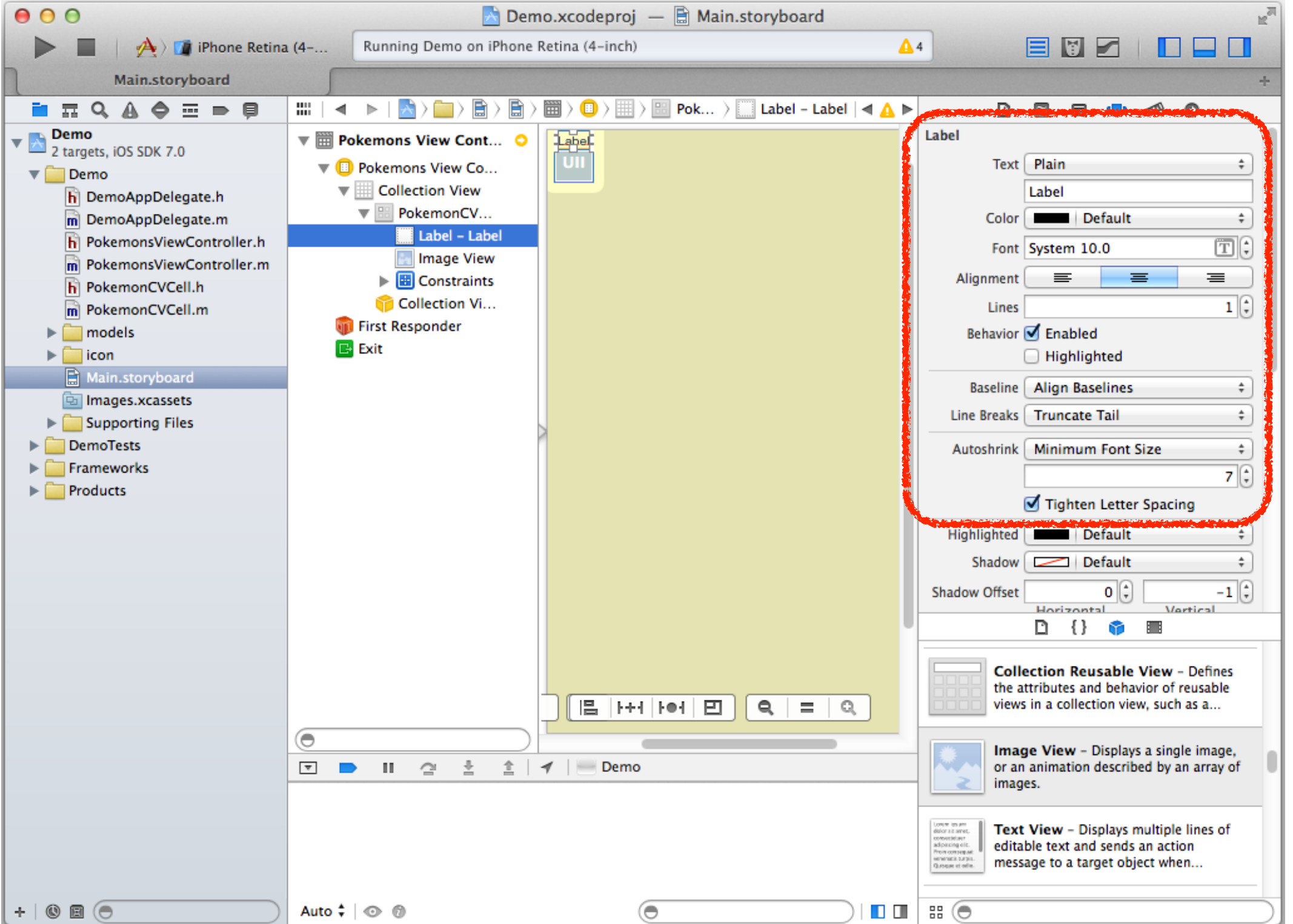
Scroll Indicators

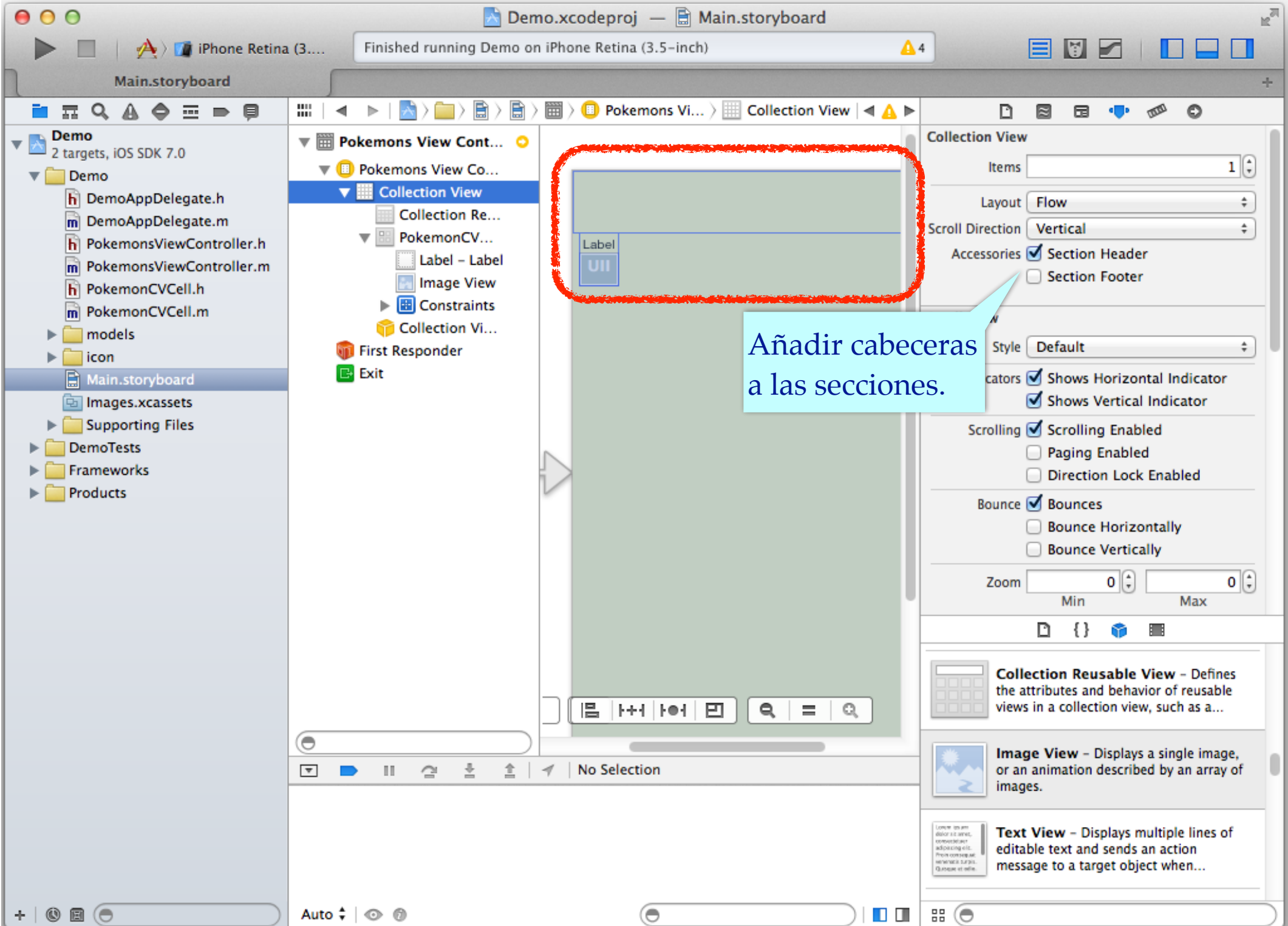
Insets	Top: 0	Bottom: 0
	Left: 0	Right: 0

Collection Reusable View - Defines the attributes and behavior of reusable views in a collection view, such as a...

Image View - Displays a single image, or an animation described by an array of images.

Text View - Displays multiple lines of editable text and sends an action message to a target object when...





Añadir cabeceras a las secciones.

Choose options for your new file:

Crear una clase nueva para las cabeceras.
Derivada de UICollectionViewReusableView.

Class Subclass of

- Targeted for iPad
 With XIB for user interface

Cancel

Previous

Next

Delegate.h

C Header

Group

Delegate.h

tiago/Documents/
encia/iOS/
cias/2013-14 IWEB/
ection Views/
mo/Demo/
Delegate.h

Unicode (UTF-8)

OS X / Unix (LF)

able View - Defines
behavior of reusable
on view, such as a...



Image View - Displays a single image, or an animation described by an array of images.



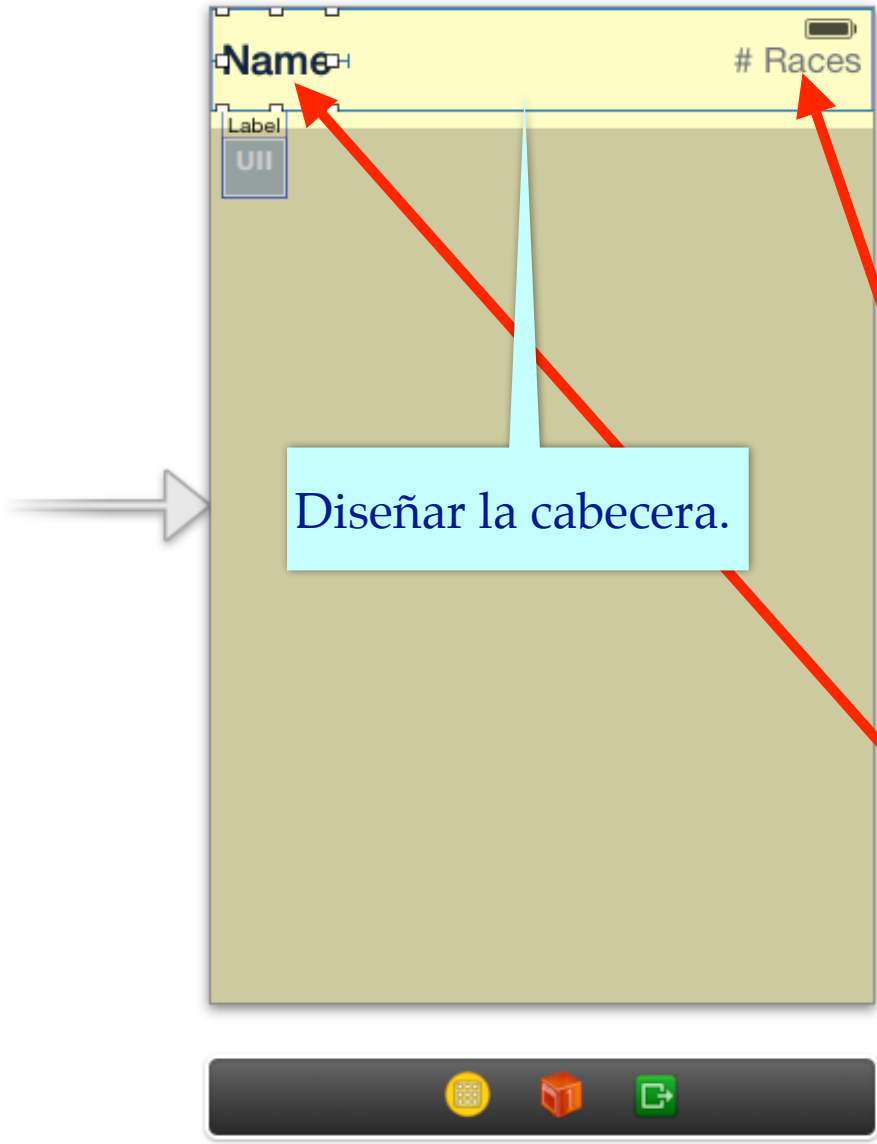
Text View - Displays multiple lines of editable text and sends an action message to a target object when...

Cambiar la clase de la cabecera a PokemonCVHeader.

The screenshot displays the Xcode IDE with the following components:

- Left Panel (Project Navigator):** Shows the project structure for 'Demo'. The files 'PokemonCVHeader.h' and 'PokemonCVHeader.m' are circled in red.
- Middle Panel (Storyboard):** Shows a 'Collection View' with a 'PokemonCV Header' cell selected. A 'Label' widget is visible on the header.
- Right Panel (Inspector):** Shows the 'Custom Class' section with the class name 'PokemonCVHeader' selected in the dropdown menu.
- Bottom Panel (Library):** Shows various UI components like 'Collection Reusable View', 'Image View', and 'Text View'.

- ▼ Pokemons View Controller...
 - ▼ Pokemons View Controller
 - ▼ Collection View
 - ▼ PokemonCV Header
 - Label - Name
 - Label - # Races
 - ▼ Constraints
 - Vertical Spac...
 - Vertical Spac...
 - Horizontal S...
 - Vertical Spac...
 - Vertical Spac...
 - Horizontal S...
 - Horizontal S...
 - PokemonCV Cell - ...
 - Collection View Flo...
 - First Responder
 - Exit



Diseñar la cabecera.

Label

Text: Plain

Name

Color: Black Color

Font: System Bold 20.0

Alignment: [Left | Center | Right]

Lines: 1

Behavior: Enabled

Highlighted

Baseline: Align Baselines

Line Breaks: Truncate Tail

Autoshrink: Fixed Font Size

Tighten Letter Spacing

Highlighted: Default

Shadow: Default

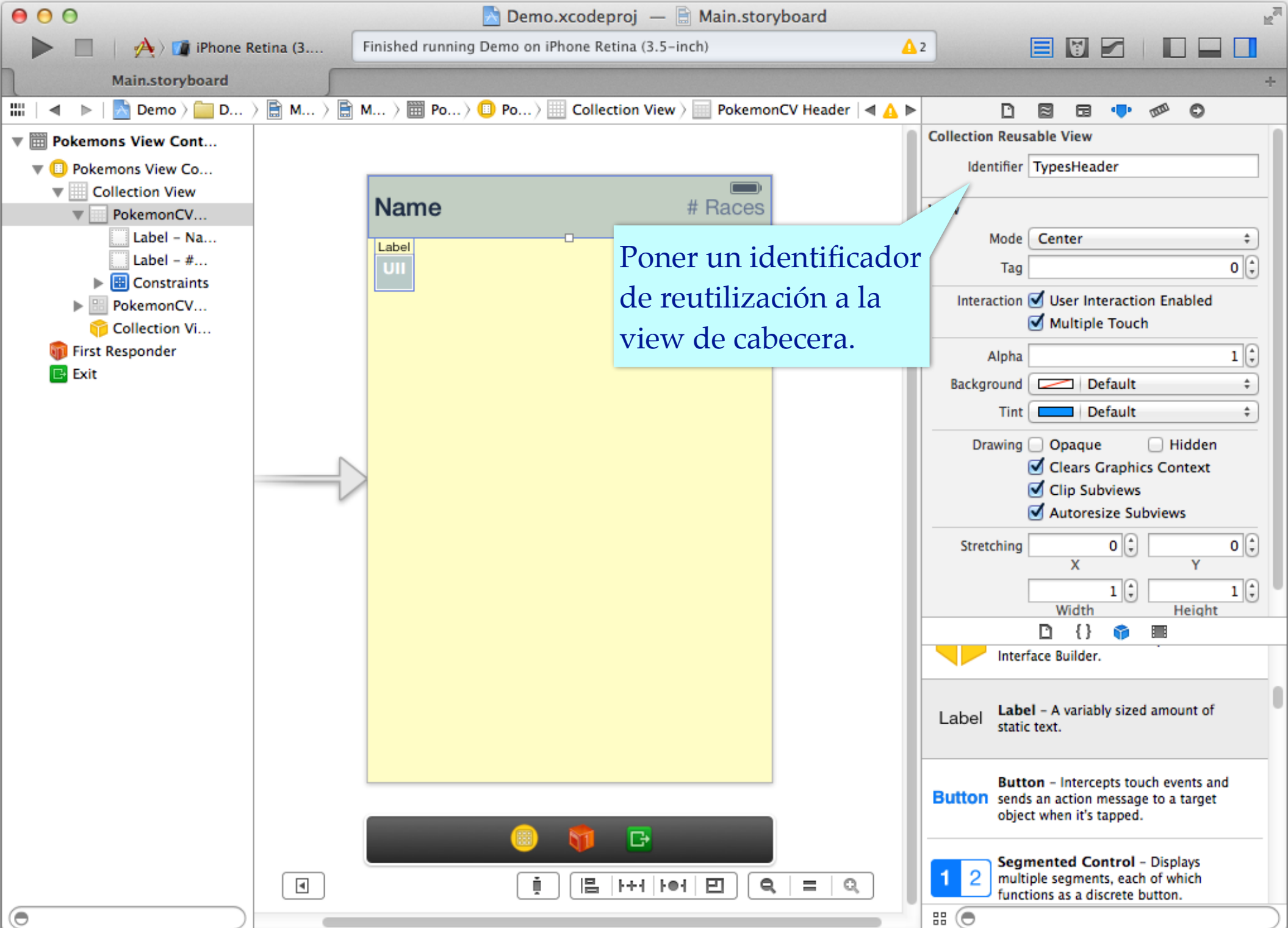
Shadow Offset: 0 Horizontal, -1 Vertical

Interface Builder.

Label - A variably sized amount of static text.

Button - Intercepts touch events and sends an action message to a target object when it's tapped.

1 2 Segmented Control - Displays multiple segments, each of which functions as a discrete button.



Poner un identificador de reutilización a la view de cabecera.

Collection Reusable View

Identifier

Mode

Tag

Interaction User Interaction Enabled
 Multiple Touch

Alpha

Background

Tint

Drawing Opaque Hidden
 Clears Graphics Context
 Clip Subviews
 Autosize Subviews

Stretching
X Y

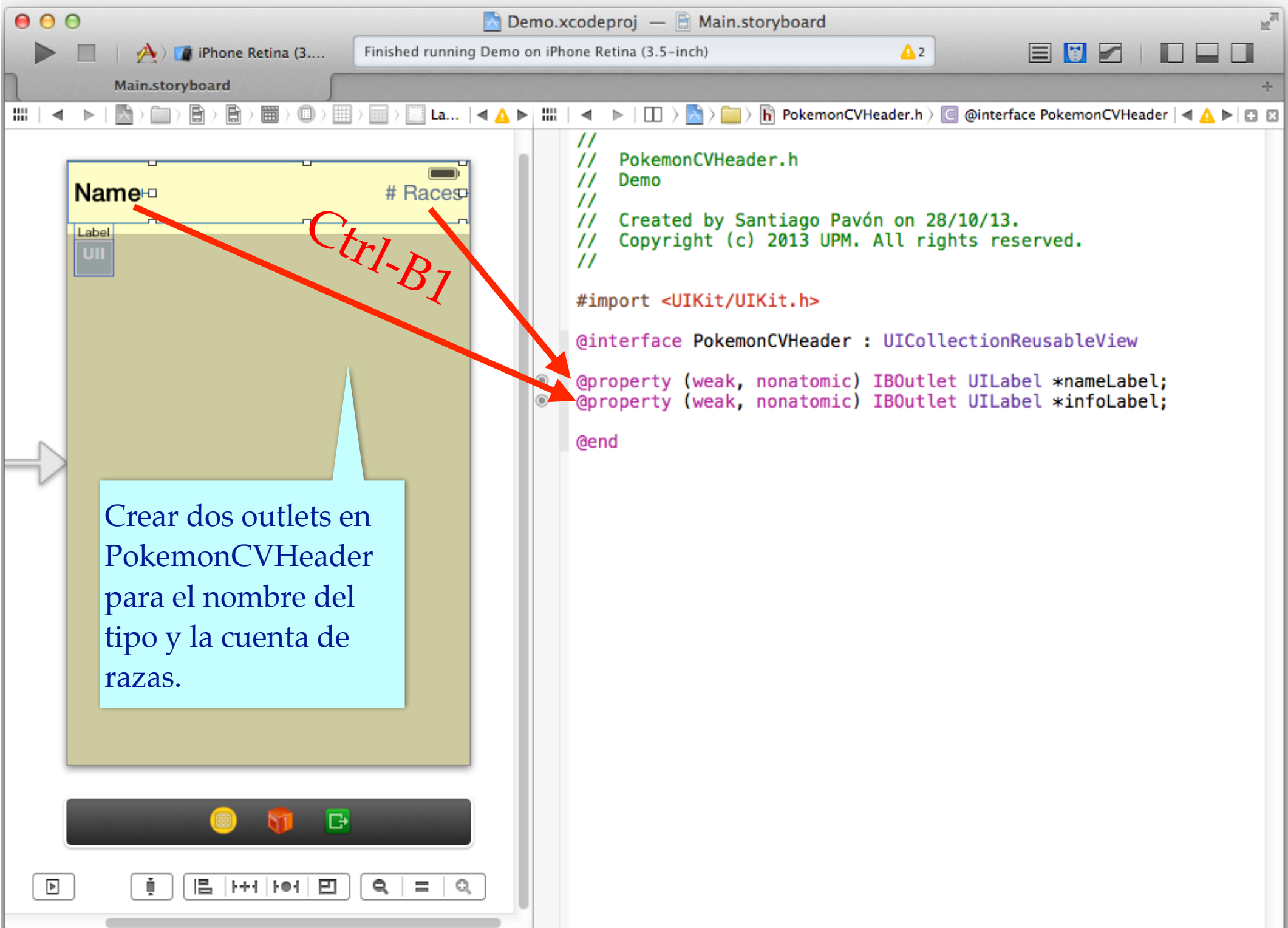
Width Height

Interface Builder.

Label Label - A variably sized amount of static text.

Button Button - Intercepts touch events and sends an action message to a target object when it's tapped.

1 2 Segmented Control - Displays multiple segments, each of which functions as a discrete button.



```

Demo.xcodeproj — PokemonsViewController.m
Finished running Demo on iPhone Retina (3.5-inch) No Issues
PokemonsViewController.m
Demo > Demo > PokemonsViewController.m > No Selection

#pragma mark - Collection View Data Source
- (NSInteger) numberOfSectionsInCollectionView:(UICollectionView *)collectionView
{ ... }
- (NSInteger) collectionView:(UICollectionView *)collectionView
numberOfItemsInSection:(NSInteger)section
{ ... }
- (UICollectionViewCell *)collectionView:(UICollectionView *)collectionView
cellForItemAtIndexPath:(NSIndexPath *)indexPath
{ ... }
-(UICollectionViewReusableView *)collectionView:(UICollectionView *)collectionView
viewForSupplementaryElementOfKind:(NSString *)kind
atIndexPath:(NSIndexPath *)indexPath
{
    if ([kind isEqualToString:UICollectionViewElementKindSectionHeader]) {
        PokemonCVHeader *header = [collectionView dequeueReusableSupplementaryViewOfKind:kind
                                     withReuseIdentifier:@"TypesHeader"
                                     forIndexPath:indexPath];

        Type * type = self.pokedeskModel.types[indexPath.section];

        header.nameLabel.text = [type.name capitalizedString];
        header.infoLabel.text = [NSString stringWithFormat:@"%d razas",[type.races count]];

        return header;
    }

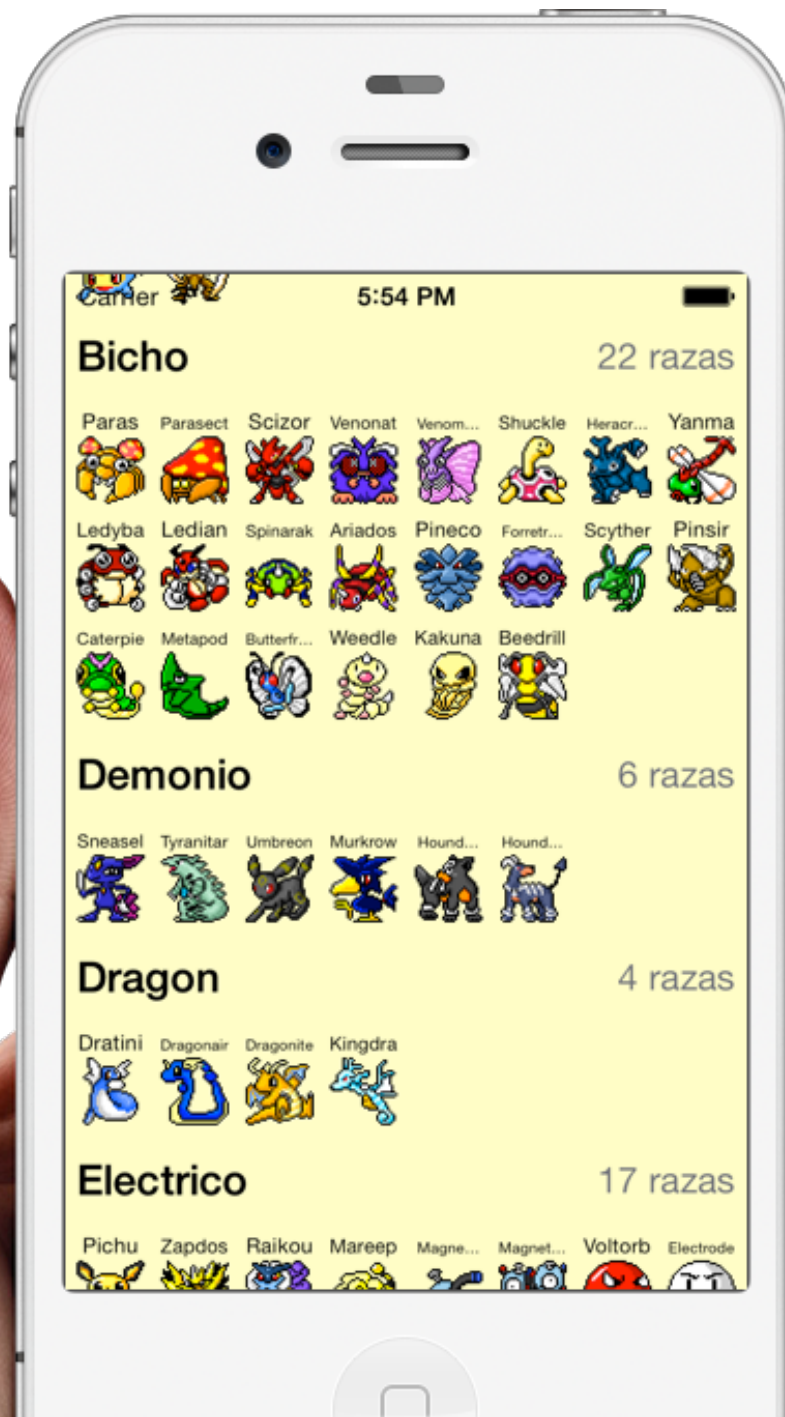
    return nil;
}

@end
|

```

Método del Data Source

Añadir arriba:
#import "PokemonCVHeader.h"



Bicho								22 razas
Paras	Parasect	Scizor	Venonat	Venom...	Shuckle	Heracr...	Yanma	
Ledyba	Ledian	Spinarak	Ariados	Pineco	Forretr...	Scyther	Pinsir	
Caterpie	Metapod	Butterfr...	Weedle	Kakuna	Beedrill			
Demonio								6 razas
Sneasel	Tyranitar	Umbreon	Murkrow	Hound...	Hound...			
Dragon								4 razas
Dratini	Dragonair	Dragonite	Kingdra					
Electrico								17 razas
Pichu	Zapdos	Raikou	Mareep	Magne...	Magnet...	Voltorb	Electrode	

Usar un Navigation Controller y otra escena con un UIWebView para mostrar información descargada de internet.
Similar al ejemplo 065-Pokedesk.

