

Camera

Brent Nycum
brentnycum@gmail.com

Choose a template for your new project:

iOS

Application

Framework & Library

Other

Mac OS X

Application

Framework & Library

Application Plug-in

System Plug-in

Other

Navigation-based
Application

OpenGL ES
Application

Split View-based
Application

Tab Bar Application

Utility Application

View-based
Application

Window-based
Application

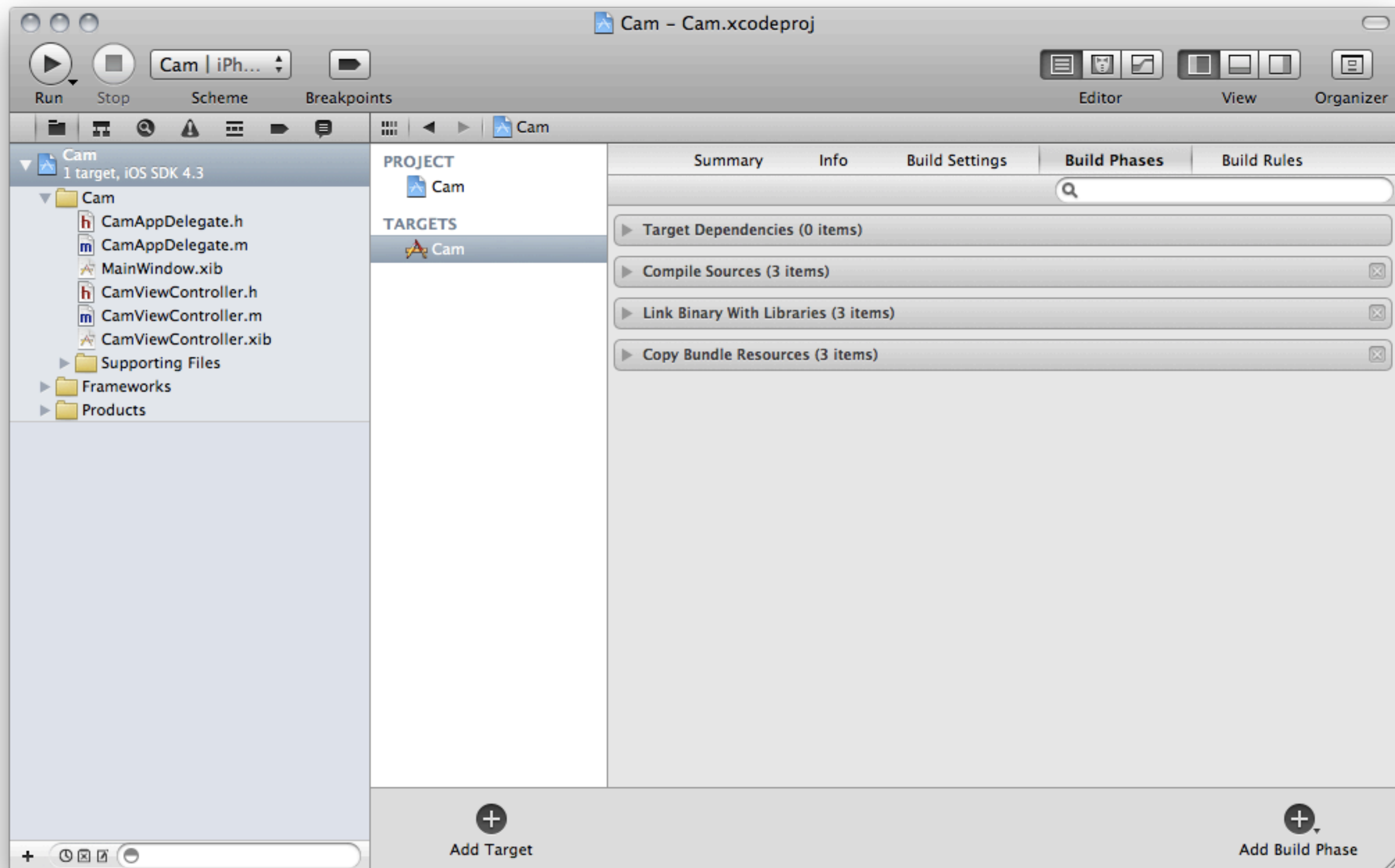
 **View-based Application**

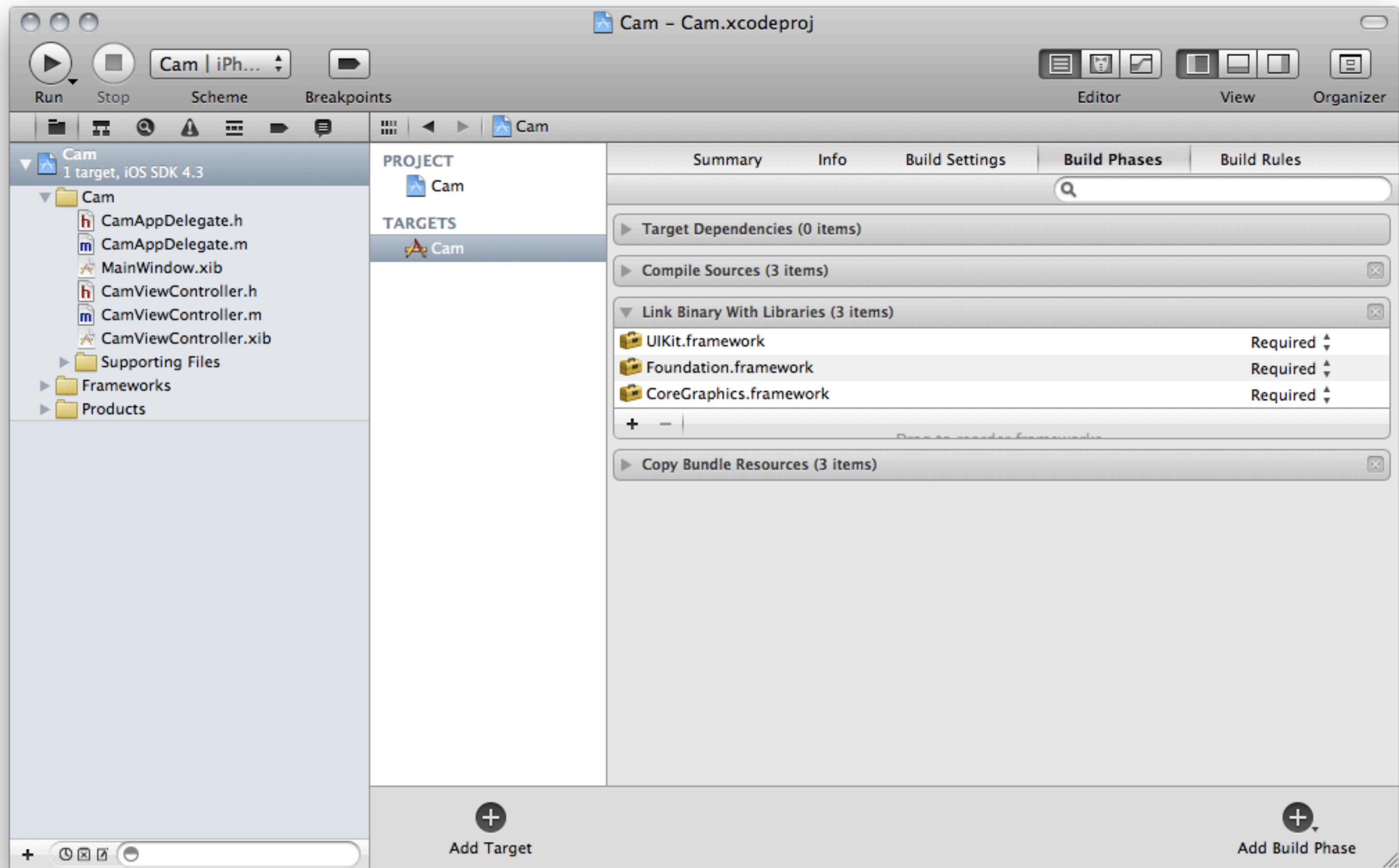
This template provides a starting point for an application that uses a single view. It provides a view controller to manage the view, and a nib file that contains the view.

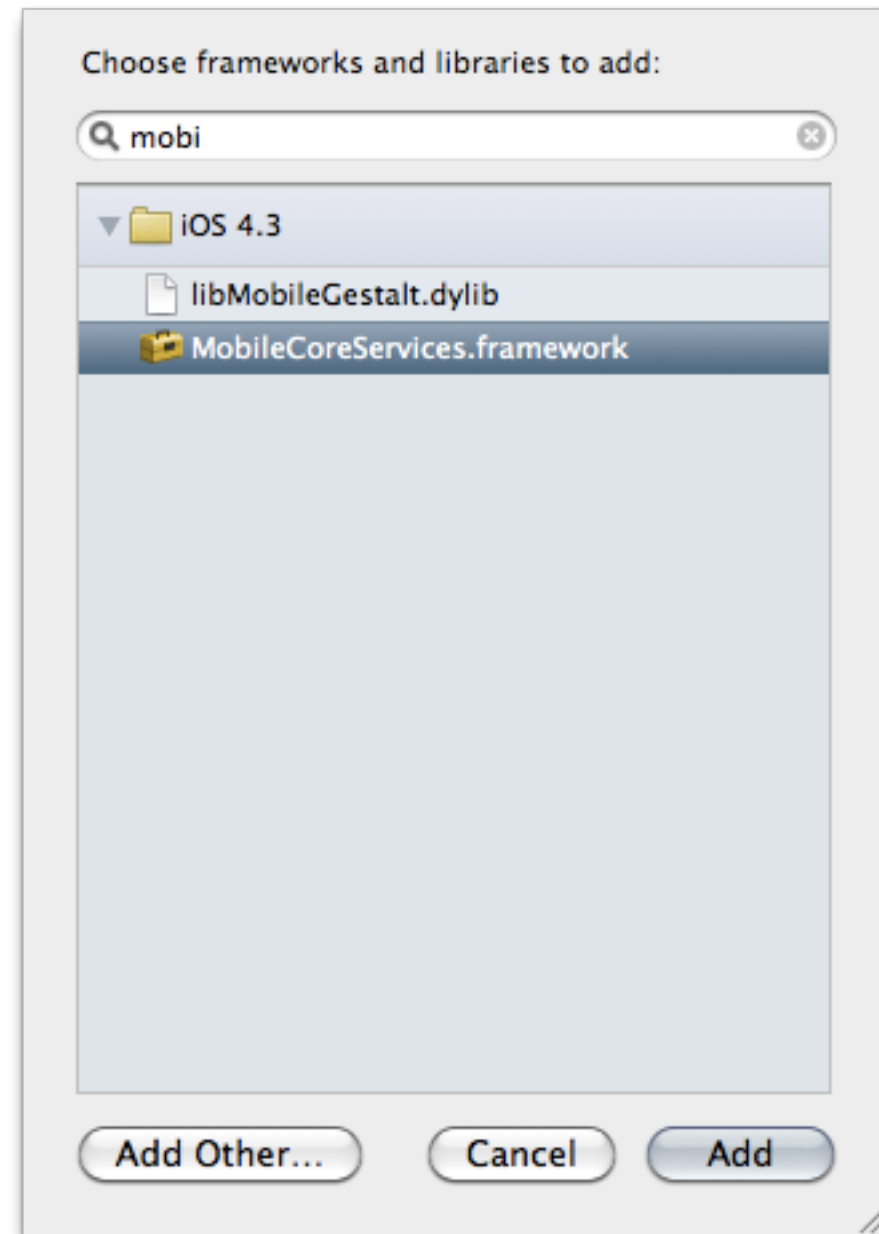
Cancel

Previous

Next

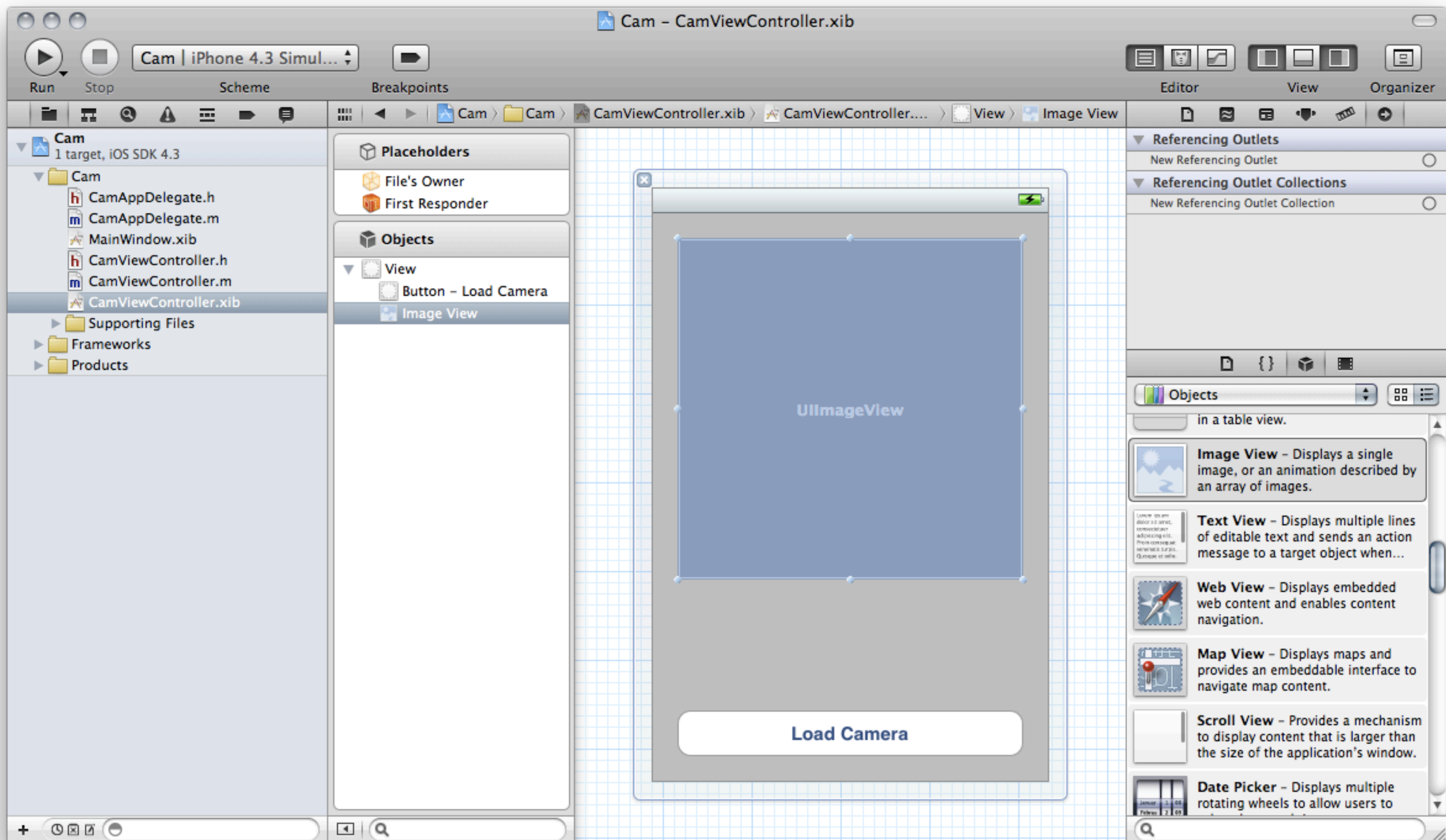


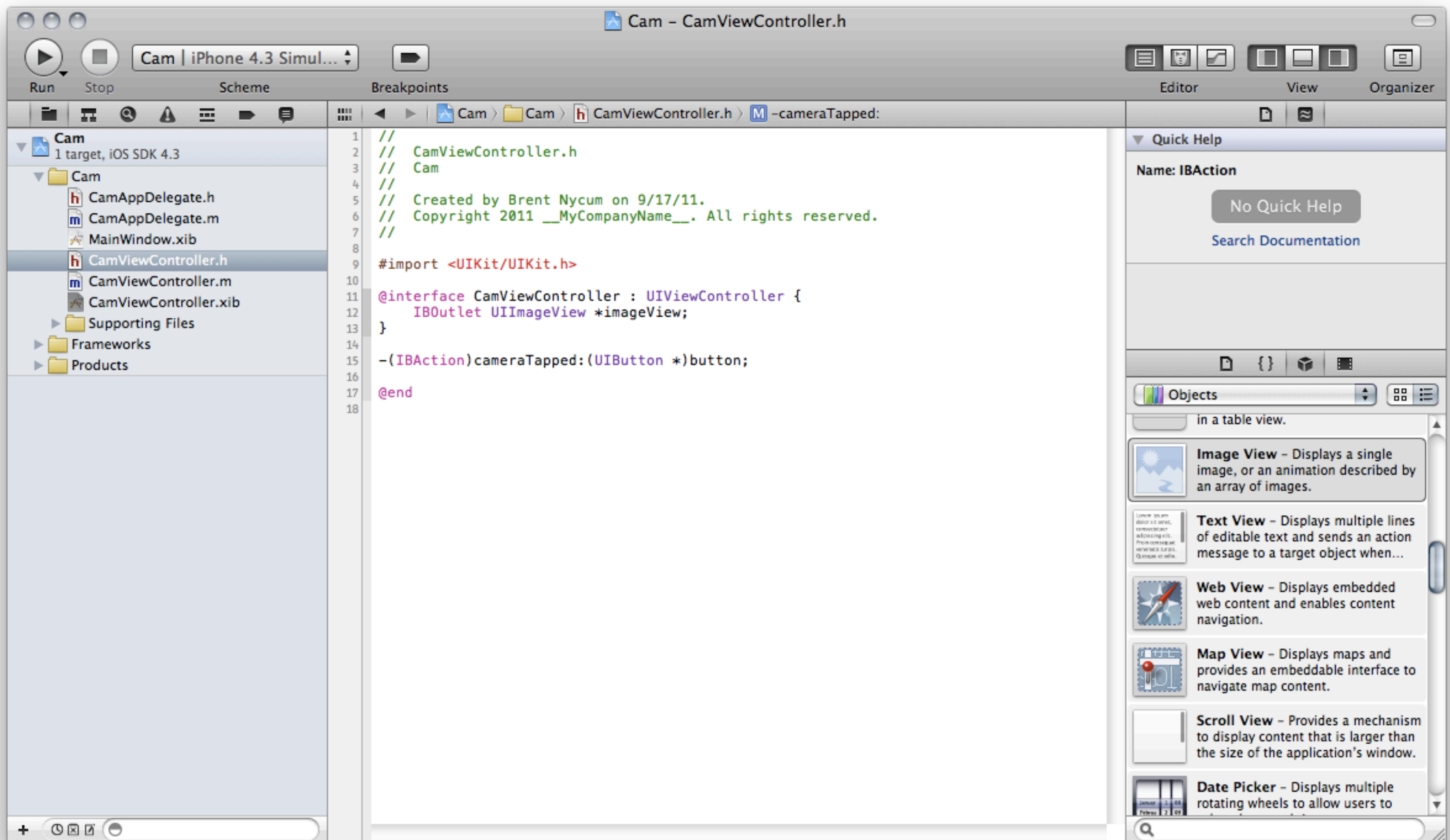


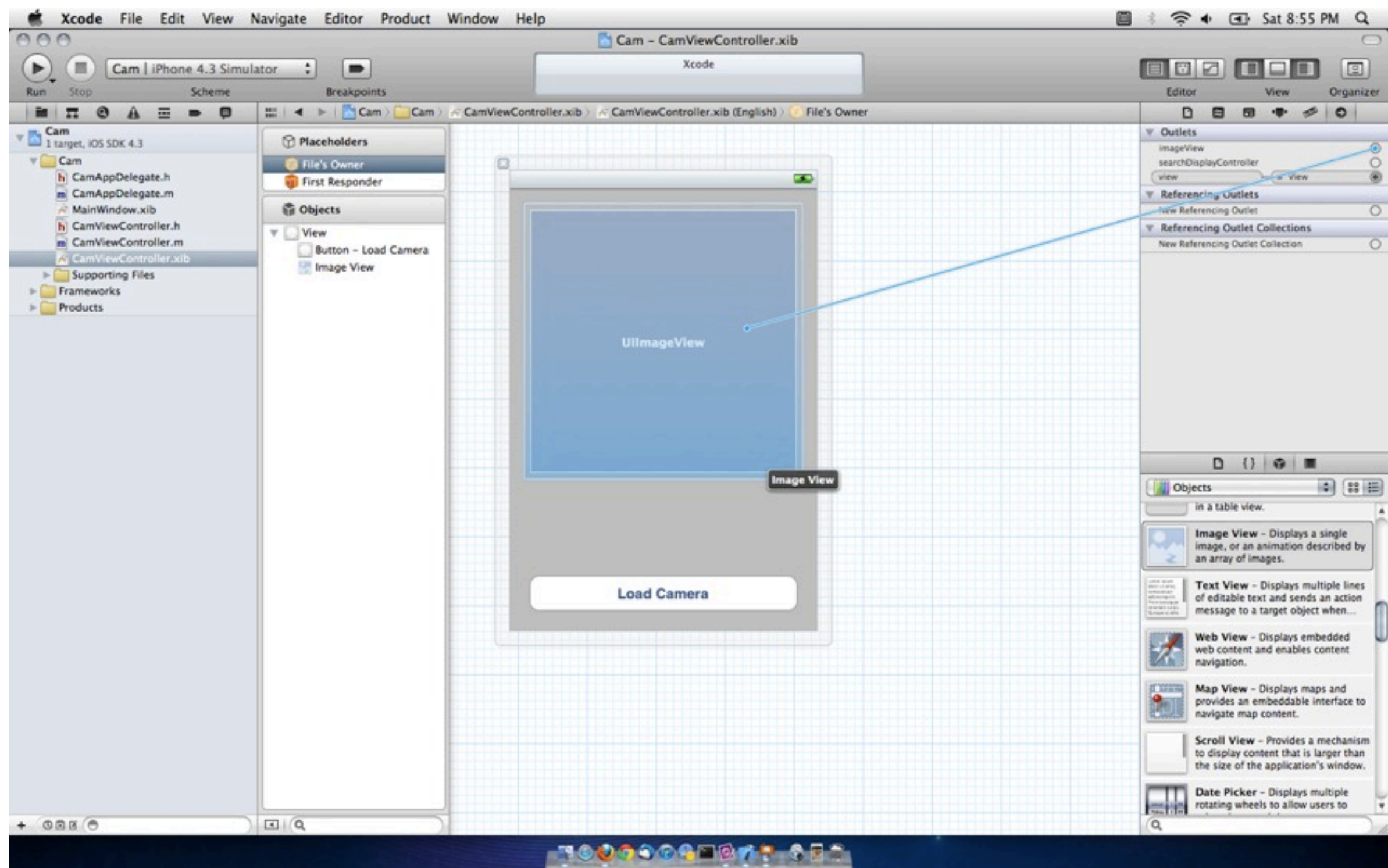


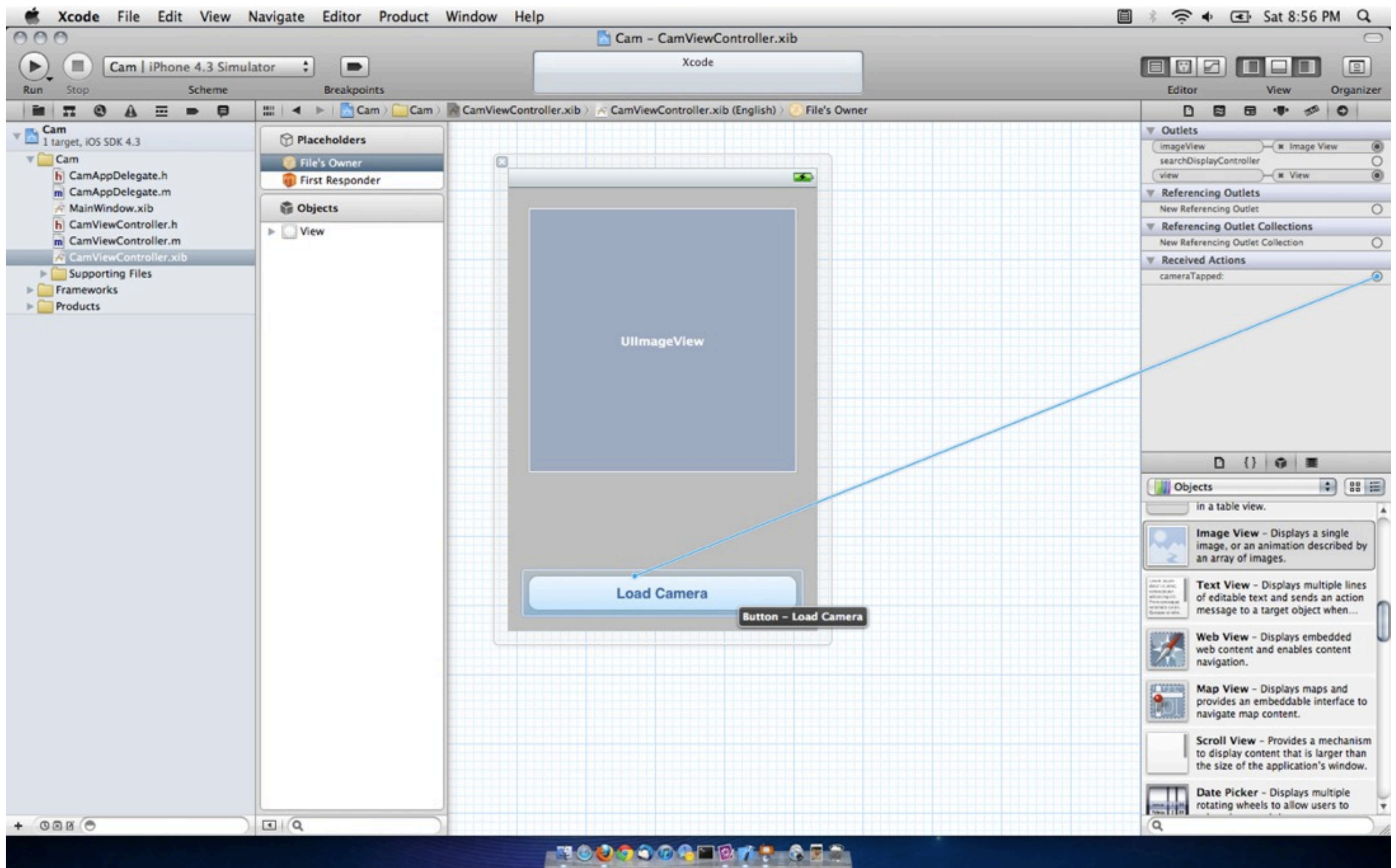
Interface

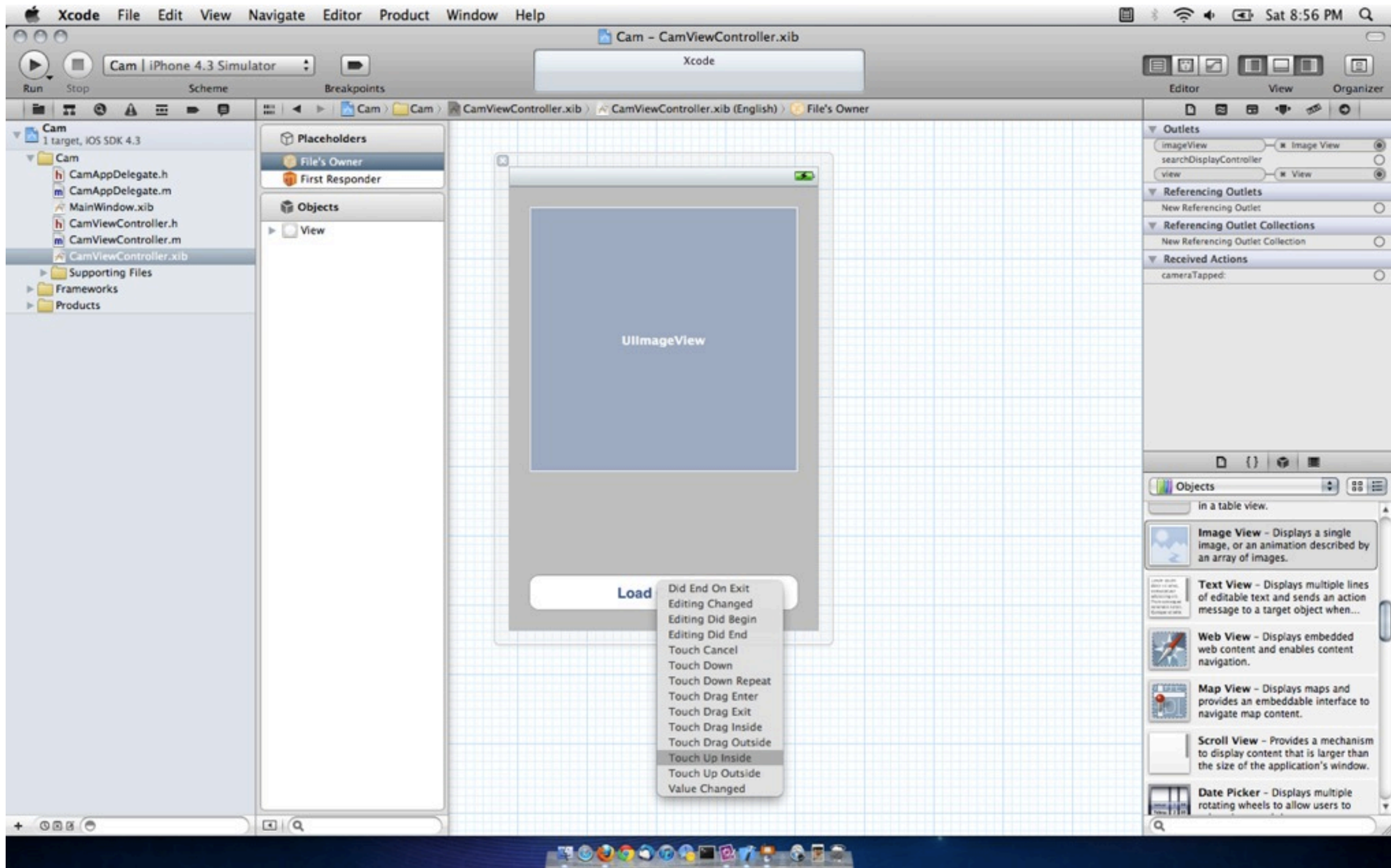
- | Button to load camera
- | Image View to show image from camera

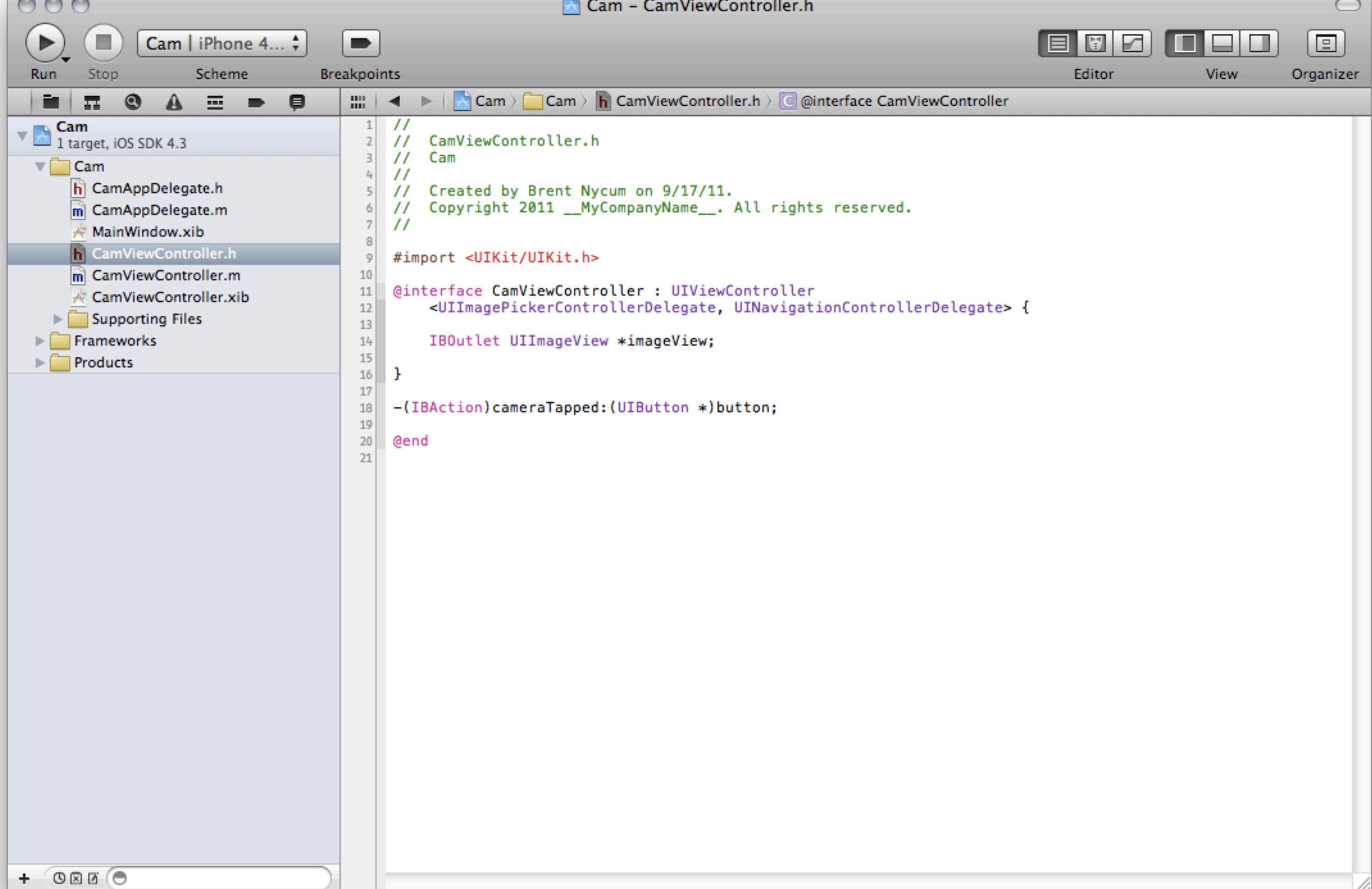


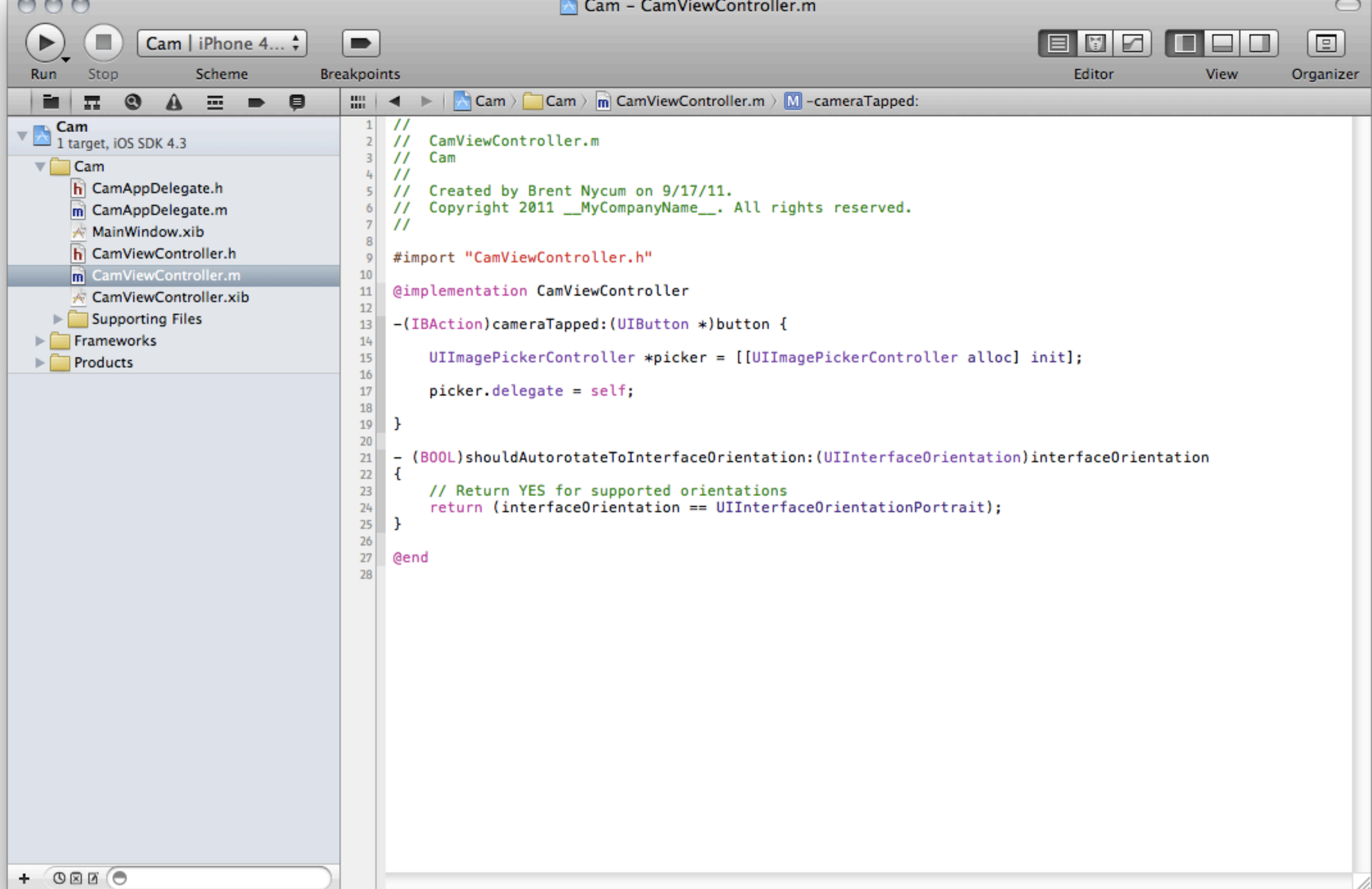


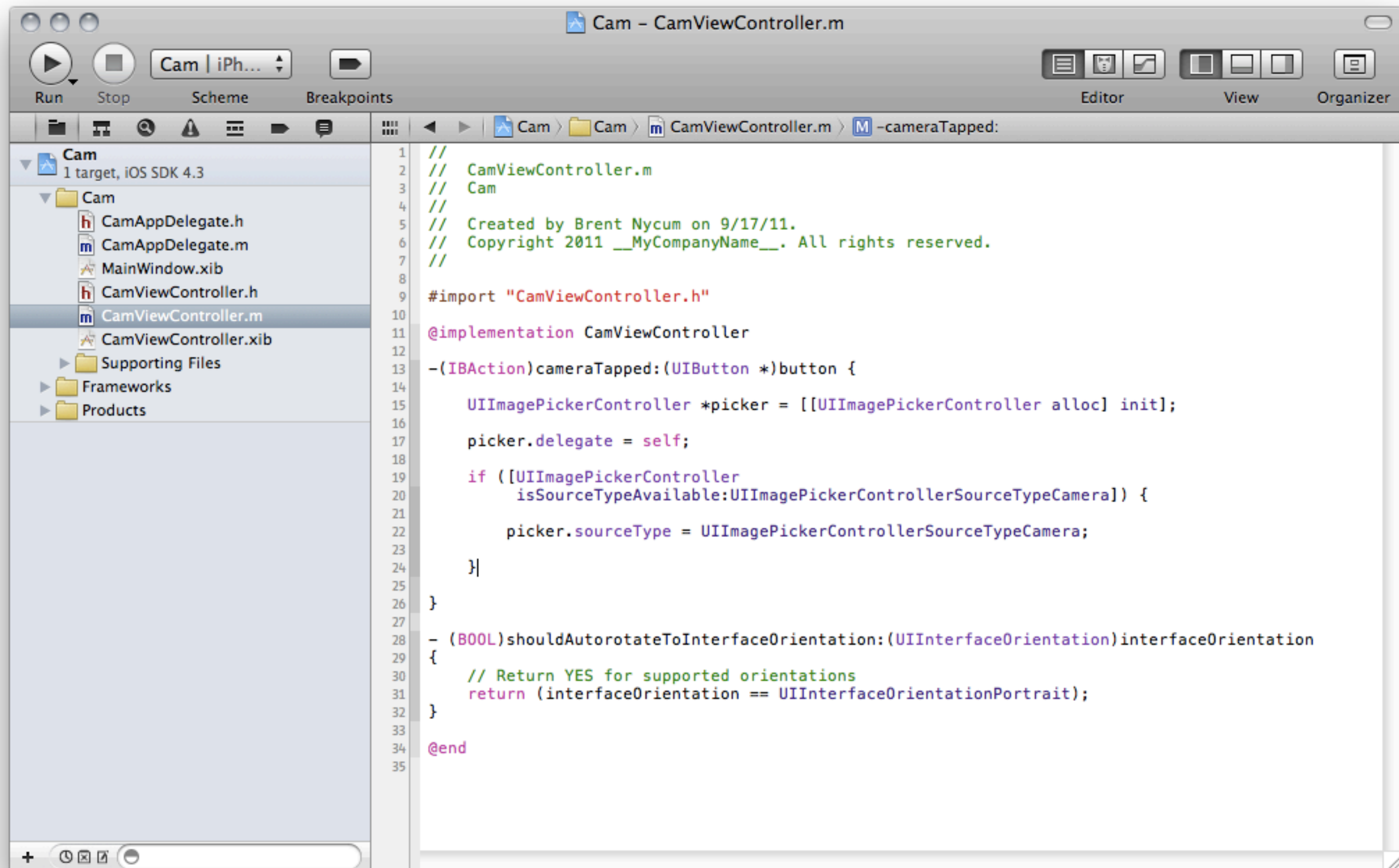


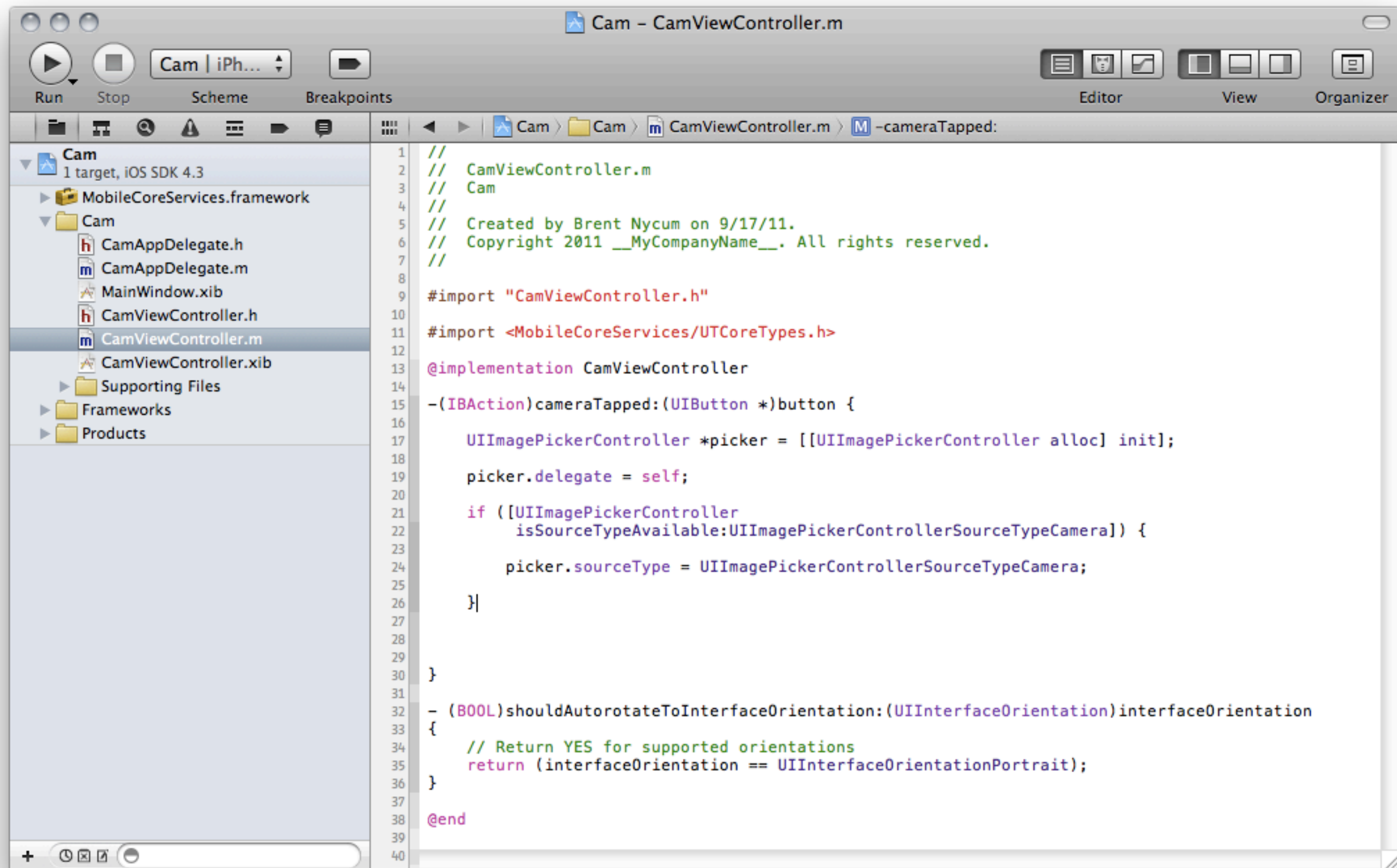


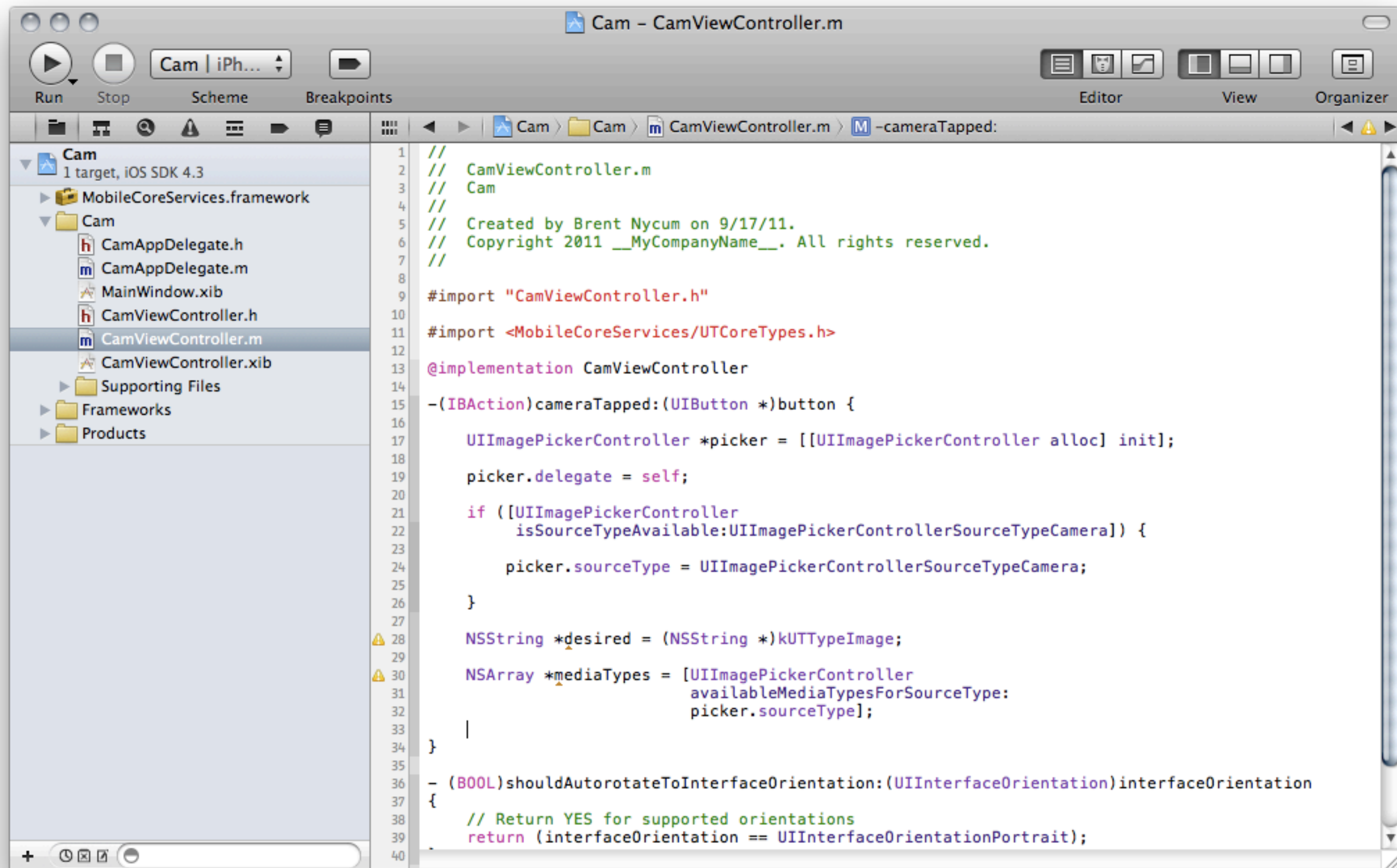


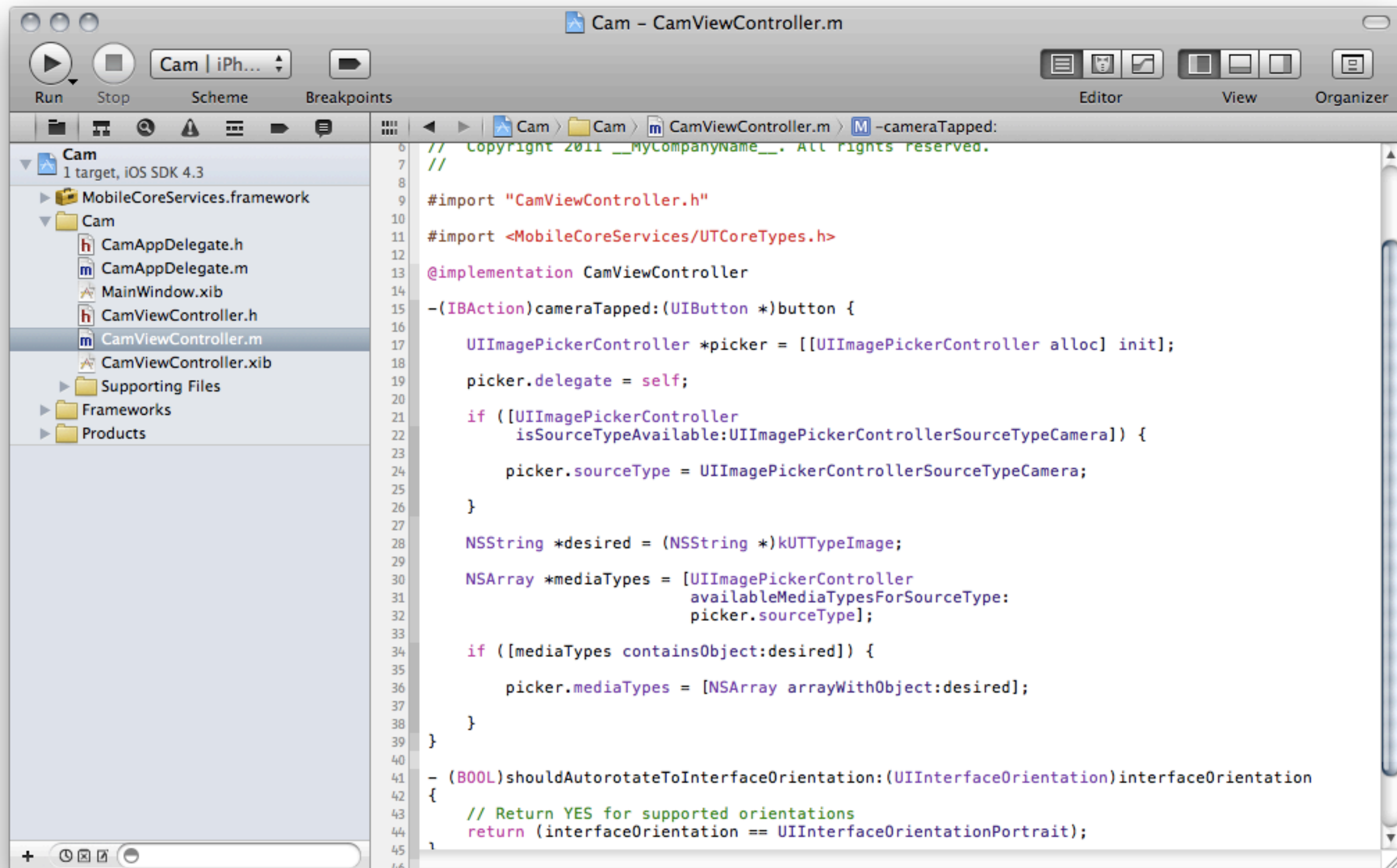


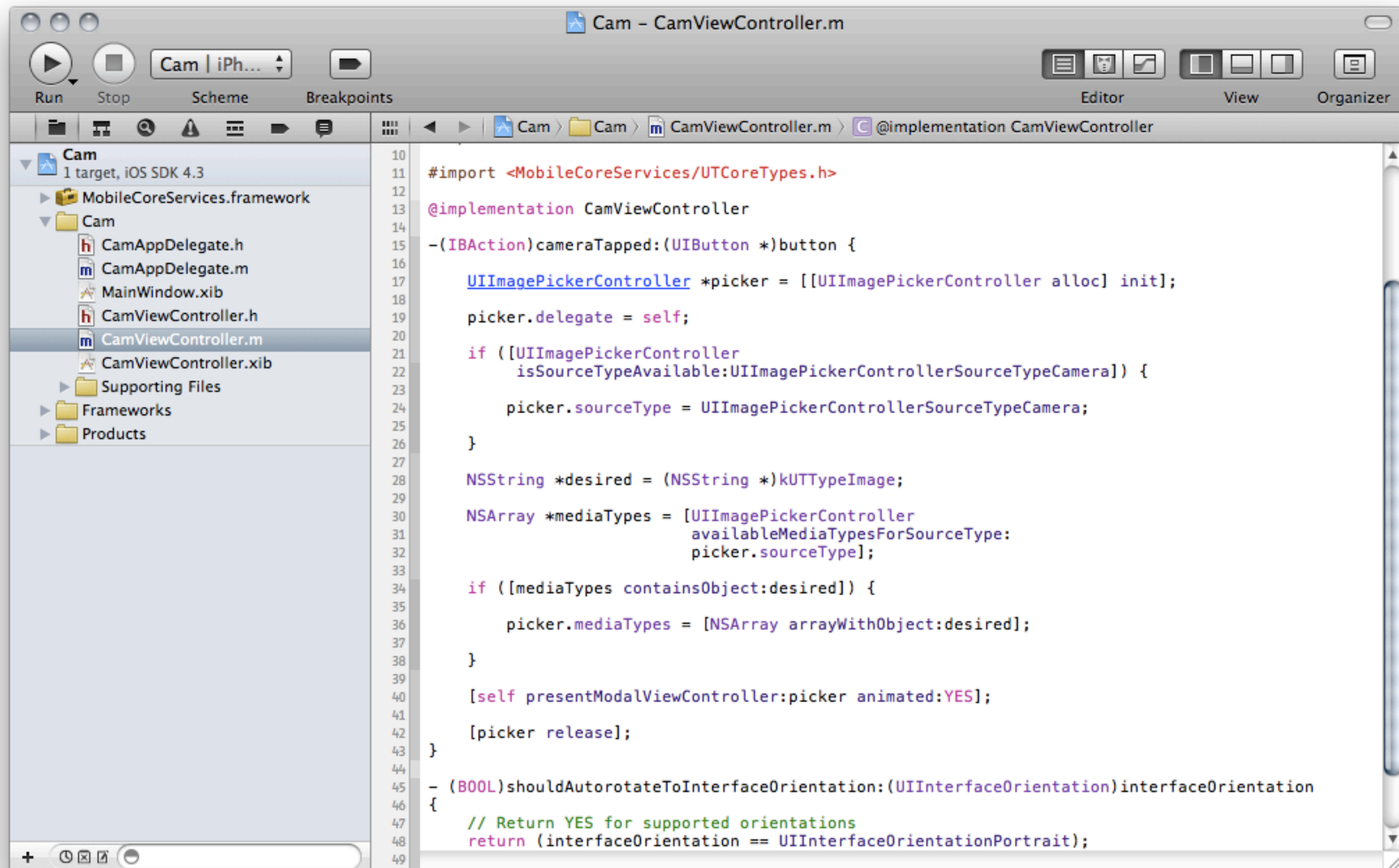












```
UIImagePickerController *picker = [[UIImagePickerController alloc] init];

picker.delegate = self;

if ([UIImagePickerController
    isSourceTypeAvailable:UIImagePickerControllerSourceTypeCamera]) {

    picker.sourceType = UIImagePickerControllerSourceTypeCamera;
}

NSString *desired = (NSString *)kUTTypeImage;

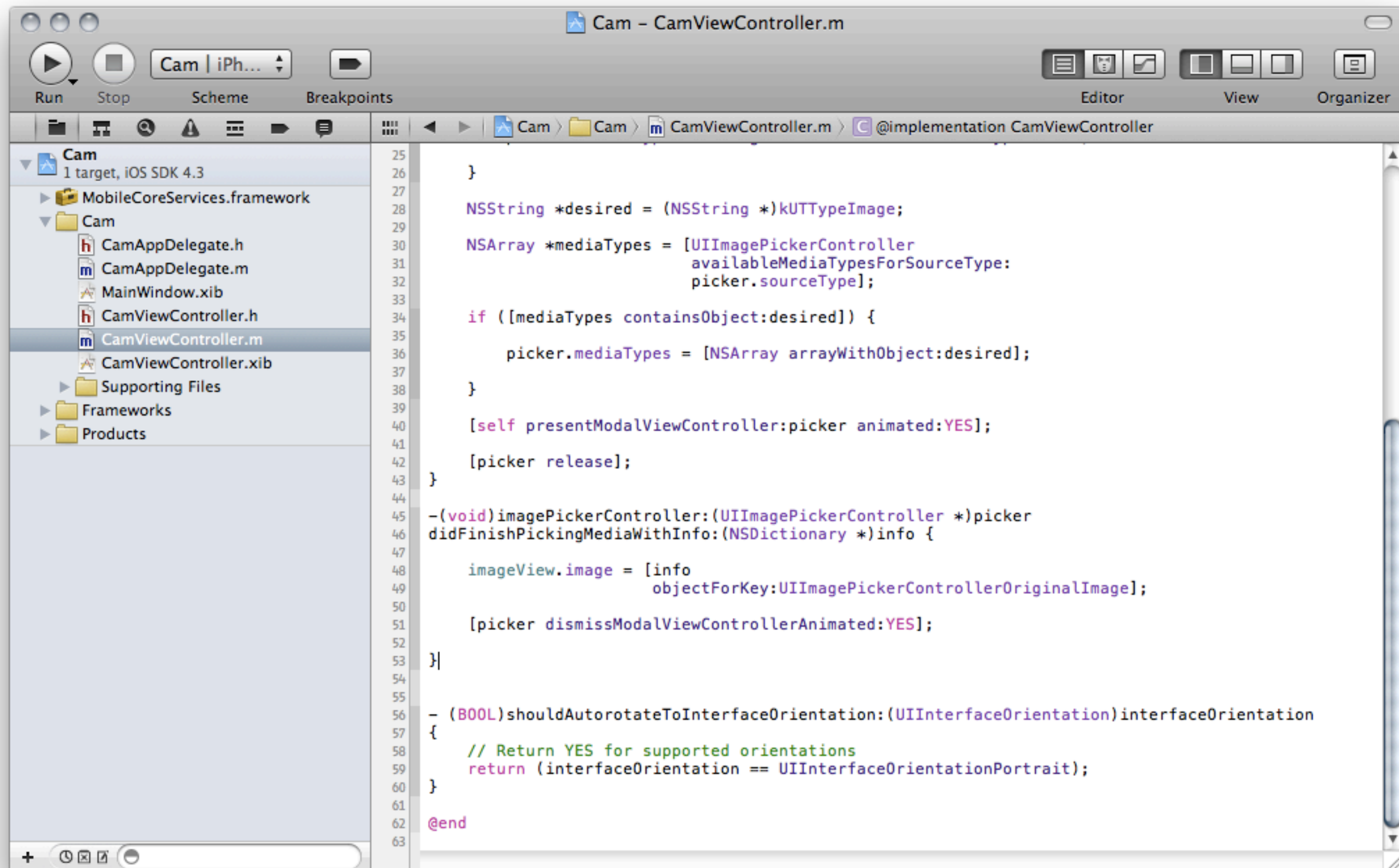
NSArray *mediaTypes = [UIImagePickerController
    availableMediaTypesForSourceType:
    picker.sourceType];

if ([mediaTypes containsObject:desired]) {

    picker.mediaTypes = [NSArray arrayWithObject:desired];
}

[self presentViewController:picker animated:YES];

[picker release];
```



```
-(void)imagePickerController:(UIImagePickerController *)picker  
didFinishPickingMediaWithInfo:(NSDictionary *)info {  
  
    imageView.image = [info  
                        objectForKey:UIImagePickerControllerOriginalImage];  
  
    [picker dismissModalViewControllerAnimated:YES];  
  
}
```