

Stanford CS193p

Developing Applications for iOS
Winter 2015



CS193p
Winter 2015

This Week

• Modal (& more about Popover) Segues

Trax Demo - Add a waypoint

• Camera

Trax Demo - Add an image to a waypoint

• Persistence

Archiving

SQLite

File System

Core Data

Trax Demo - Store a waypoint image added by the user in the file system

• Embed Segue

Putting an MVC's View as a subview of another MVC's View

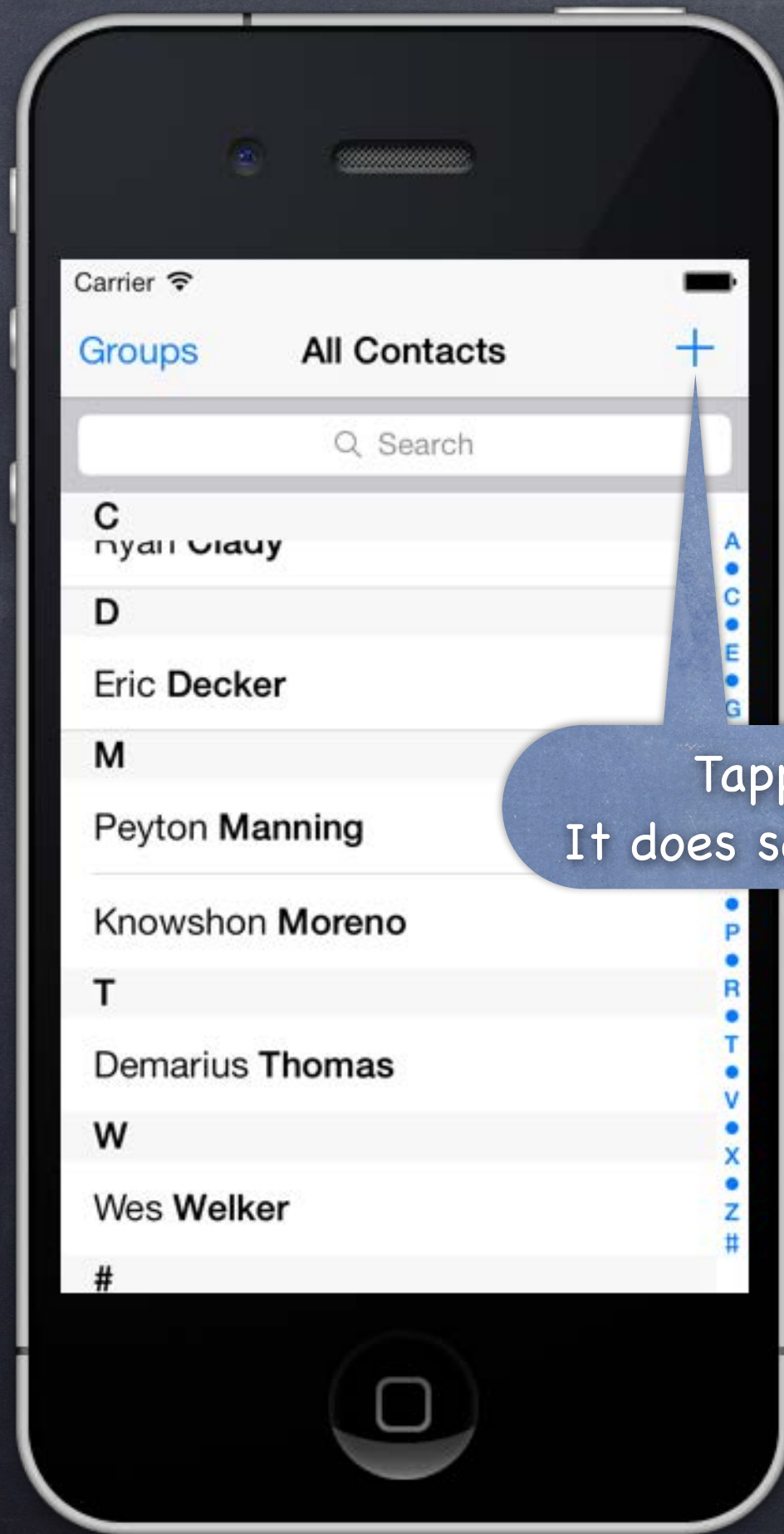
Trax Demo - Show a "mini-map" of the waypoint when viewing its image



Modal View Controllers

- A way of segueing that takes over the screen
Should be used with care.
- Example
Contacts application.

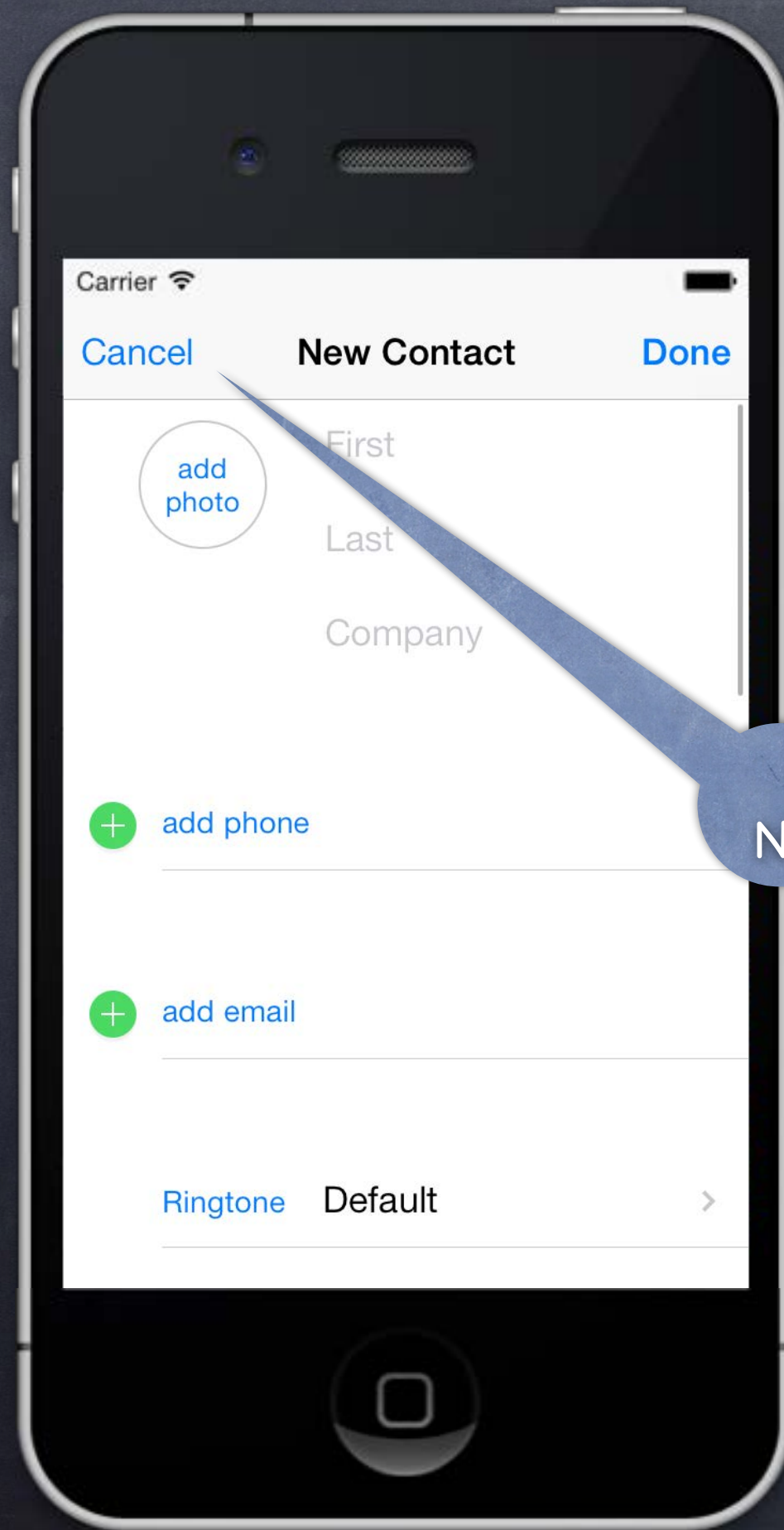
Tapping here adds a new contact.
It does so by taking over the entire screen.



Modal View Controllers

- A way of segueing that takes over the screen
Should be used with care.
- Example
Contacts application.

This is not a push.
Notice, no back button (only Cancel).



Modal View Controllers

- A way of segueing that takes over the screen
Should be used with care.
- Example
Contacts application.



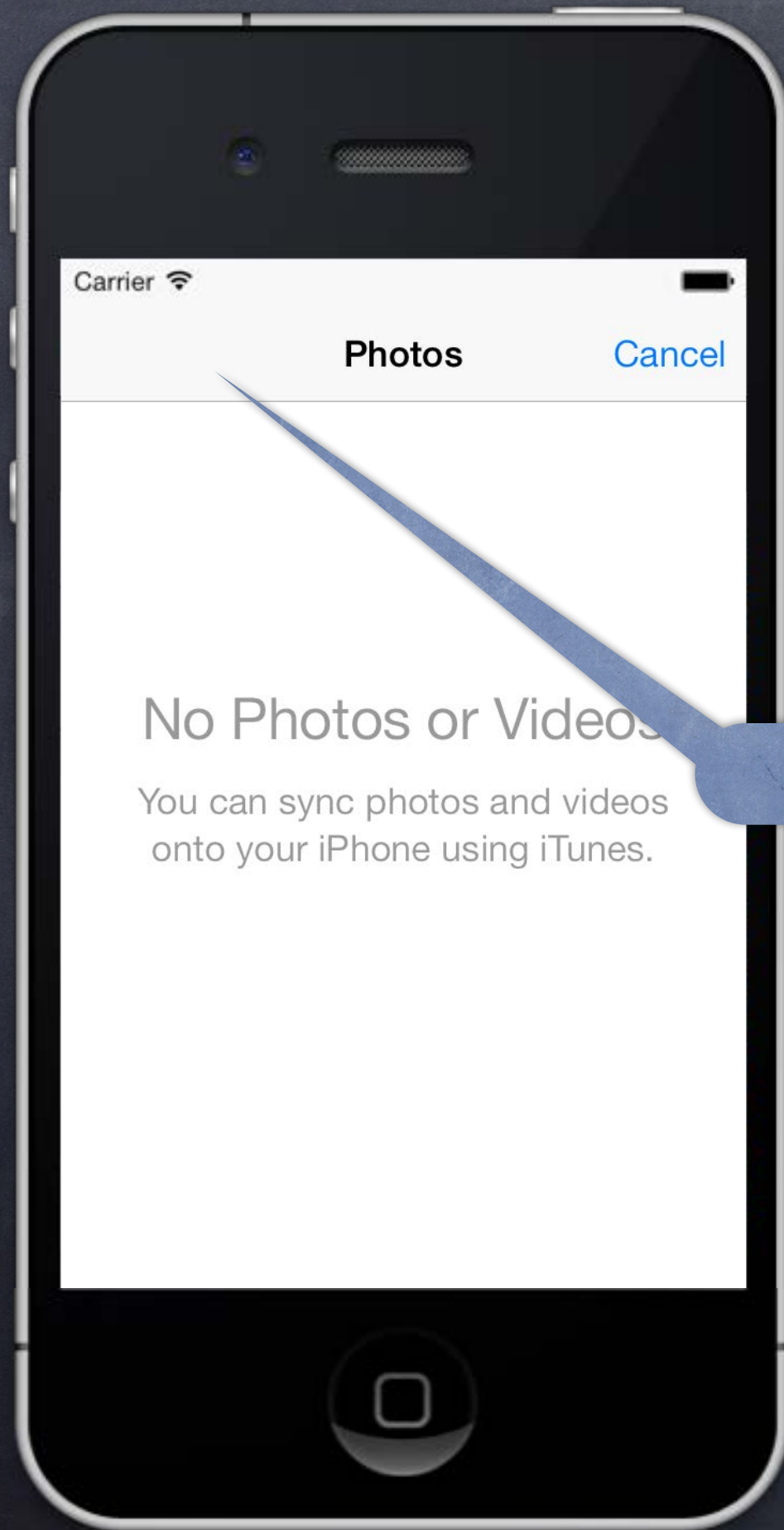
Tapping here adds a photo to this contact.
It also does so by taking over the entire screen.



Modal View Controllers

- A way of segueing that takes over the screen
Should be used with care.
- Example
Contacts application.

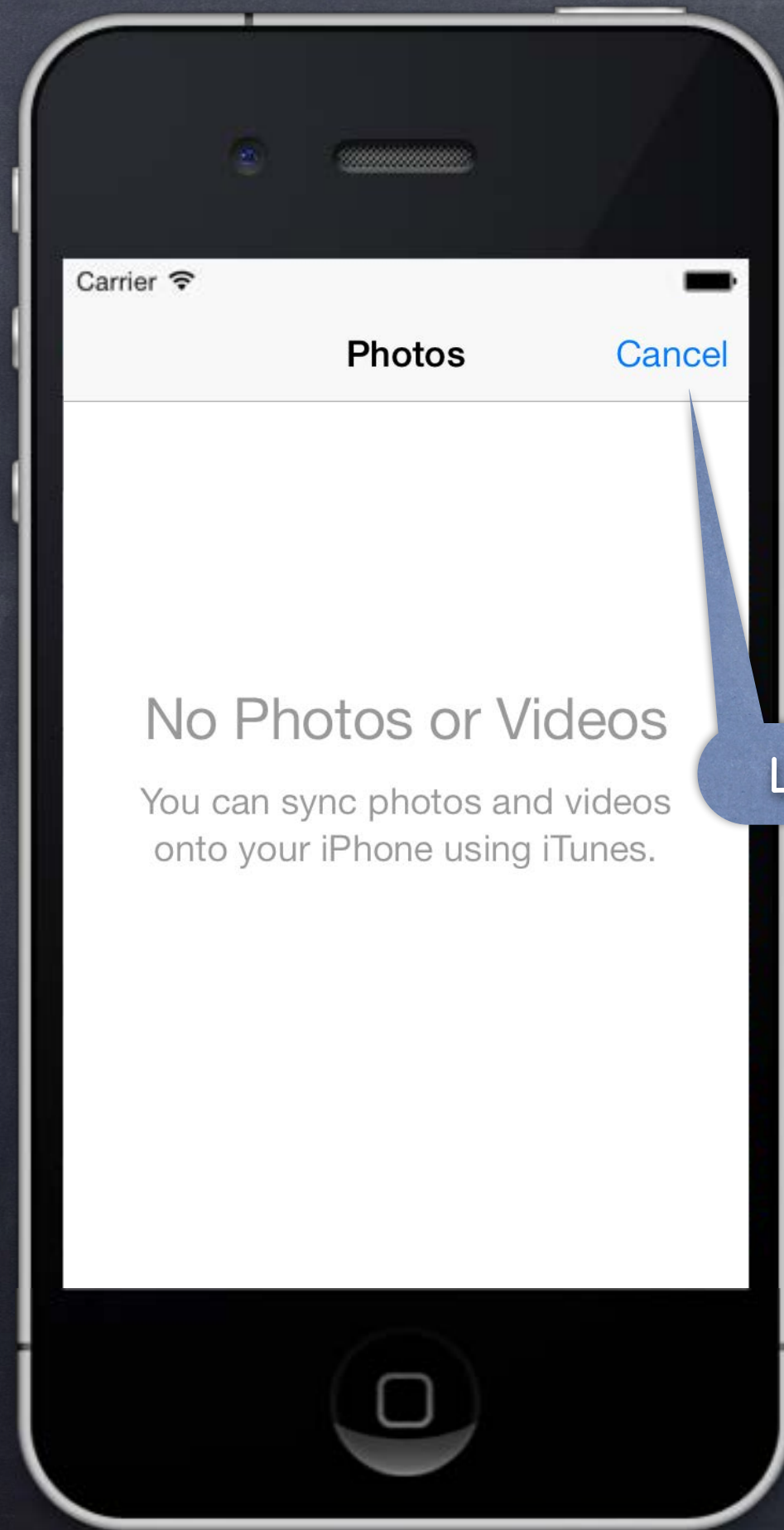
Again, no back button.



Modal View Controllers

- A way of segueing that takes over the screen
Should be used with care.
- Example
Contacts application.

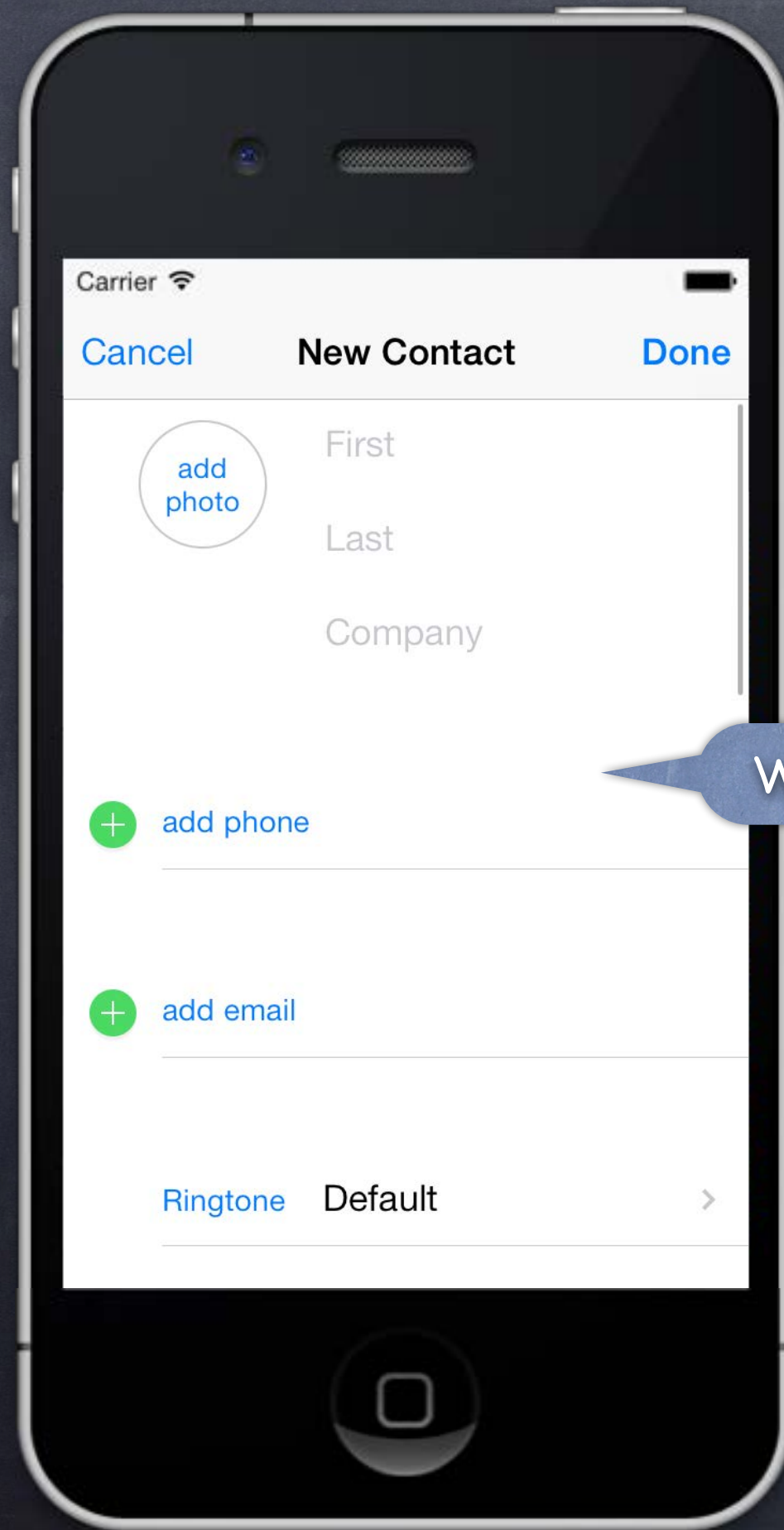
Let's Cancel and see what happens.



Modal View Controllers

- A way of segueing that takes over the screen
Should be used with care.
- Example
Contacts application.

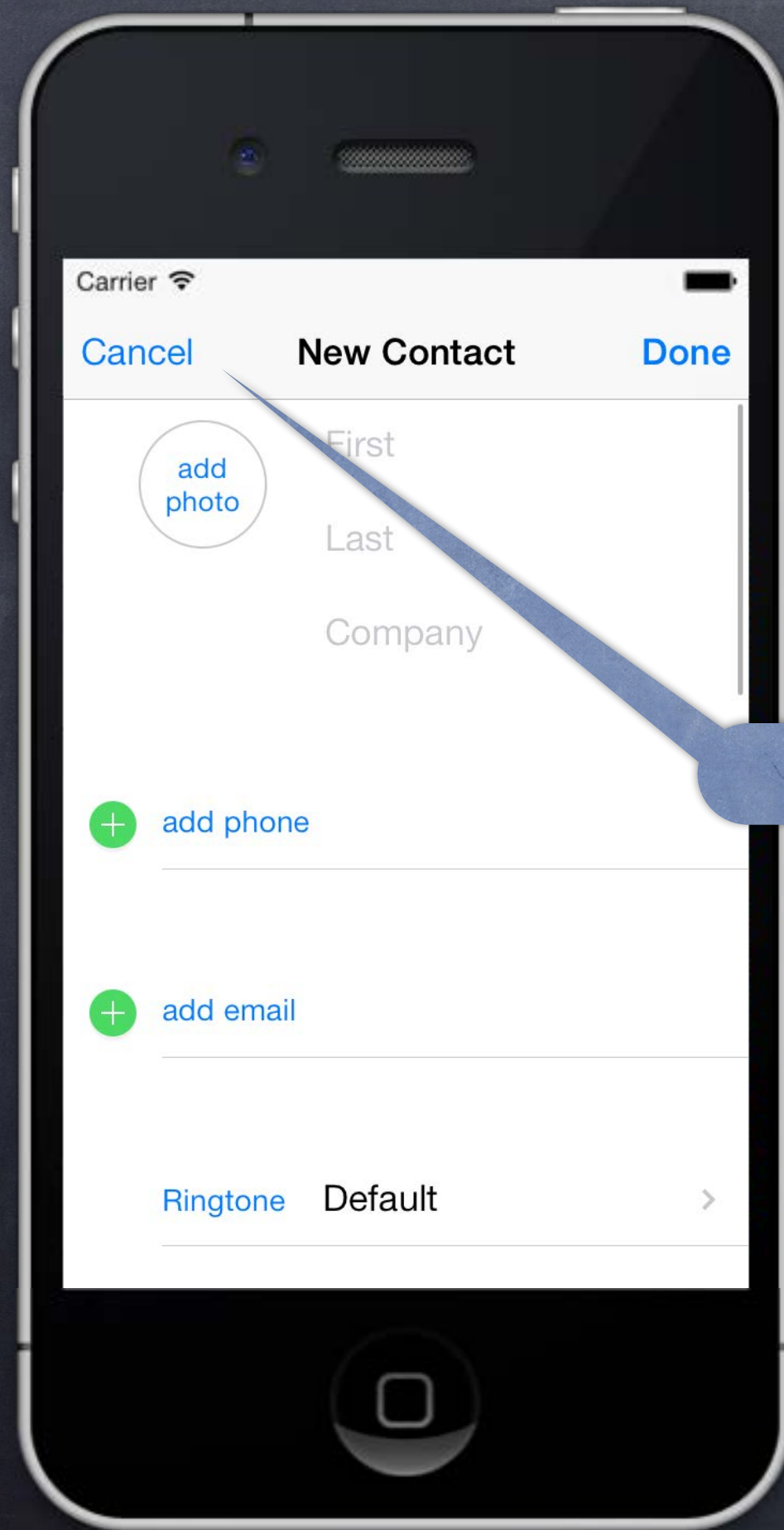
We're back to the last Modal View Controller.



Modal View Controllers

- A way of segueing that takes over the screen
Should be used with care.
- Example
Contacts application.

And Cancel again ...

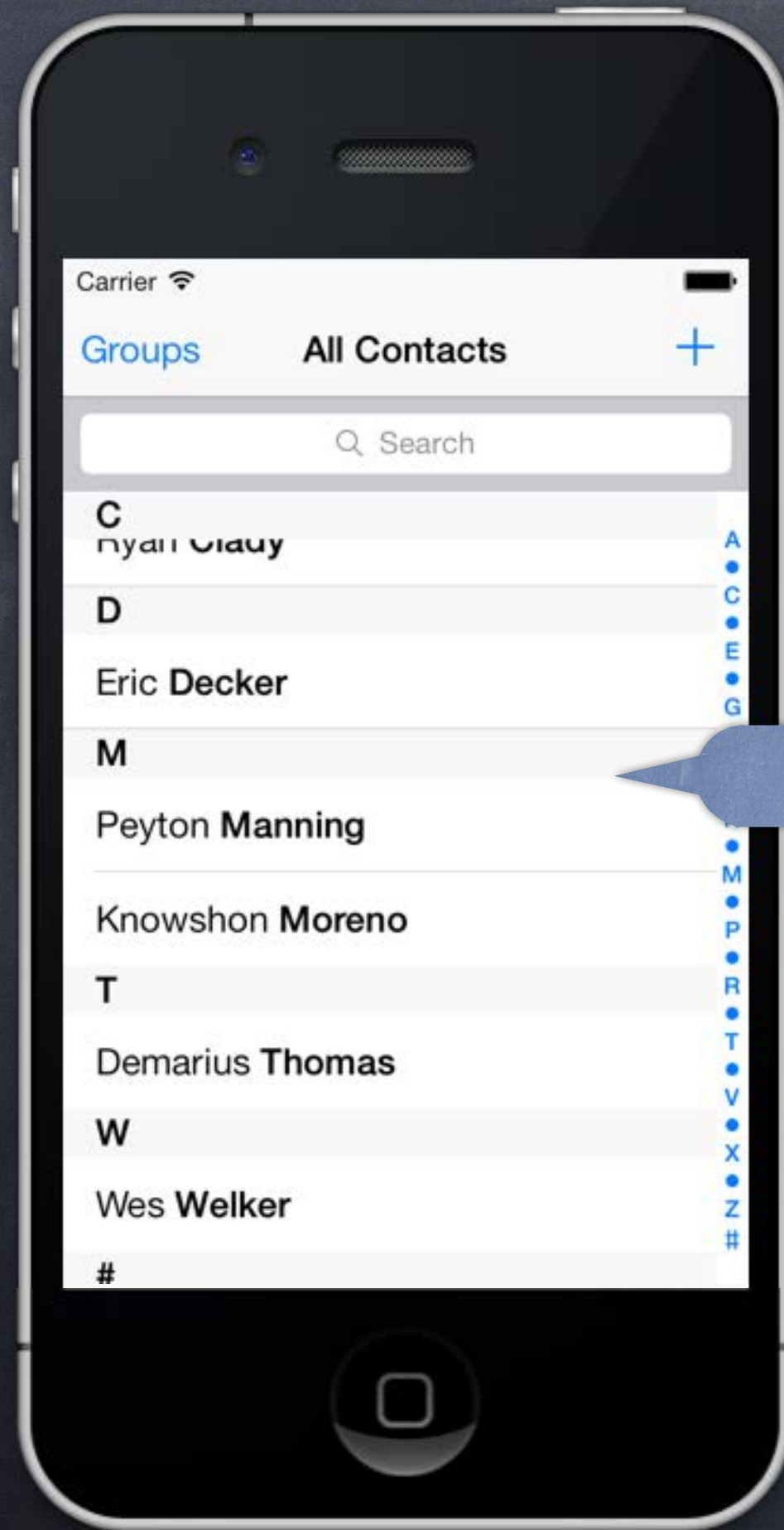


Modal View Controllers

- A way of segueing that takes over the screen
Should be used with care.

- Example
Contacts application.

Back to where we started.



Modal View Controllers

👁 Considerations

The view controller we segue to using a Modal segue will take over the entire screen
This can be rather disconcerting to the user, so use this carefully

👁 How do we set a Modal segue up?

Just ctrl-drag from a button to another View Controller & pick segue type "Modal"
Inspect the segue to set the style of presentation

If you need to present a Modal VC not from a button, use a manual segue ...

```
func performSegue(UISStoryboardSegue, withIdentifier: String)
```

... or, if you have the view controller itself (e.g. Alerts or from instantiateViewController...) ...

```
func presentViewController(UIViewController, animated: Bool, completion: () -> Void)
```

In horizontally compact environments, this will adapt to be full screen!

In horizontally regular environments, `modalPresentationStyle` will determine how it appears ...

`.FullScreen`, `.OverFullScreen` (presenter left underneath), `.Popover`, `.FormSheet`, etc.



Modal View Controllers

• Preparing for a Modal segue

You prepare for a Modal segue just like any other segue ...

```
func prepareForSegue(segue: UIStoryboardSegue, sender: AnyObject!) {  
    if segue.identifier == "GoToMyModalVC" {  
        let vc = segue.destinationViewController as MyModalVC  
        // set up the vc to run here  
    }  
}
```

• Hearing back from a Modally segue-to View Controller

When the Modal View Controller is "done", how does it communicate results back to presenter?

If there's nothing to be said, just dismiss the segued-to MVC (next slide).

To communicate results, generally you would Unwind (though delegation possible too).



Modal View Controllers

👁 How to dismiss a view controller

The presenting view controller is responsible for dismissing (not the presented)

If you send this message to the presented view controller, it will forward to its presenter

(unless it has presented another MVC, in which case it is the presenter for that other MVC)

```
func dismissViewControllerAnimated(Bool, completion: () -> Void)
```

Remember that unwind automatically dismisses



Modal View Controllers

👁 How is the modal view controller animated onto the screen?

Depends on this property in the view controller that is being put up modally ...

```
var modalTransitionStyle: UIModalTransitionStyle
```

```
.CoverVertical // slides up and down from bottom of screen (the default)
```

```
.FlipHorizontal // flips the current view controller view over to modal
```

```
.CrossDissolve // old fades out as new fades in
```

```
.PartialCurl // only if presenter is full screen (and no more modal presentations coming)
```

