

What's New in Cocoa Touch

Session 202

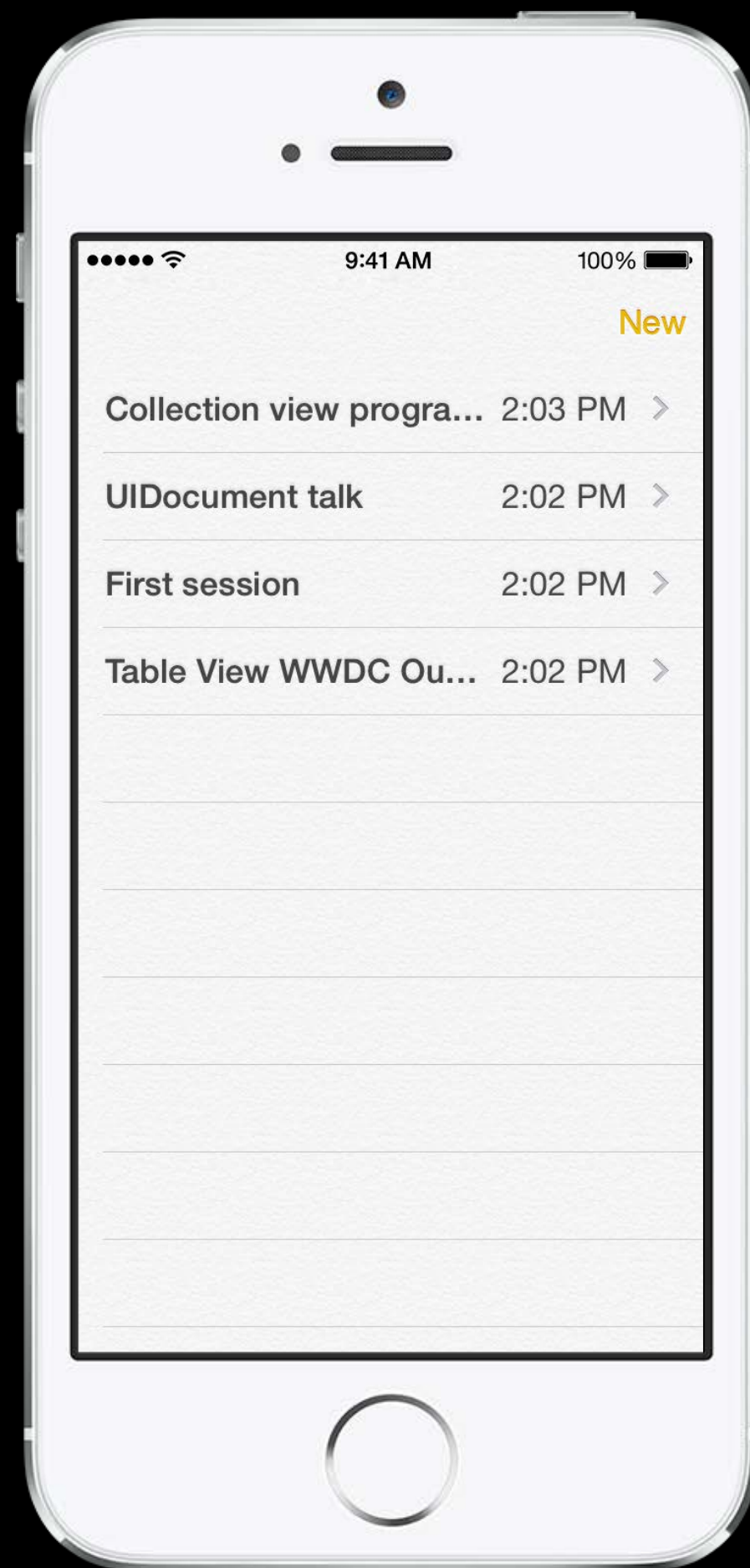
Luke Hiesterman

UIKit Engineer

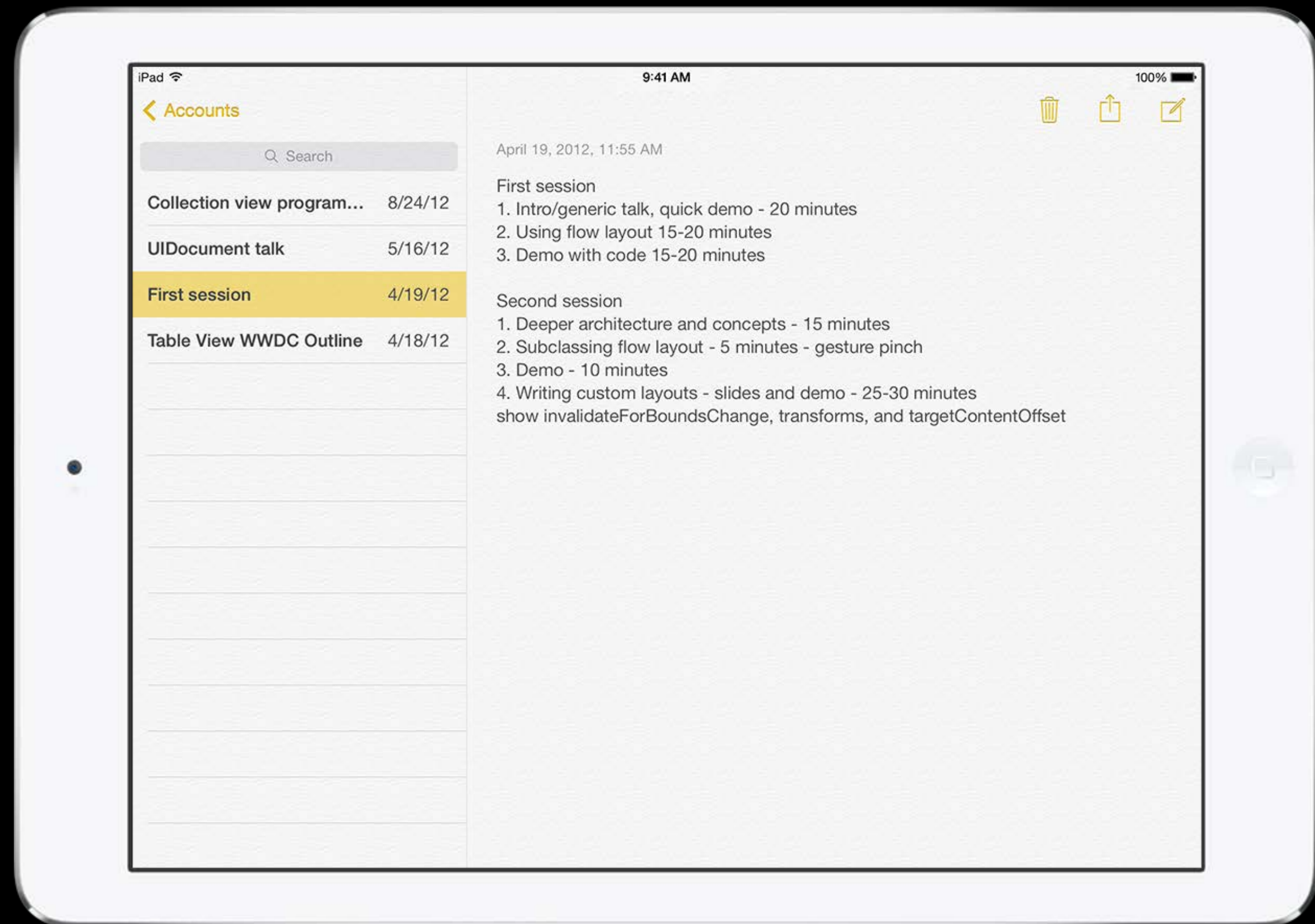
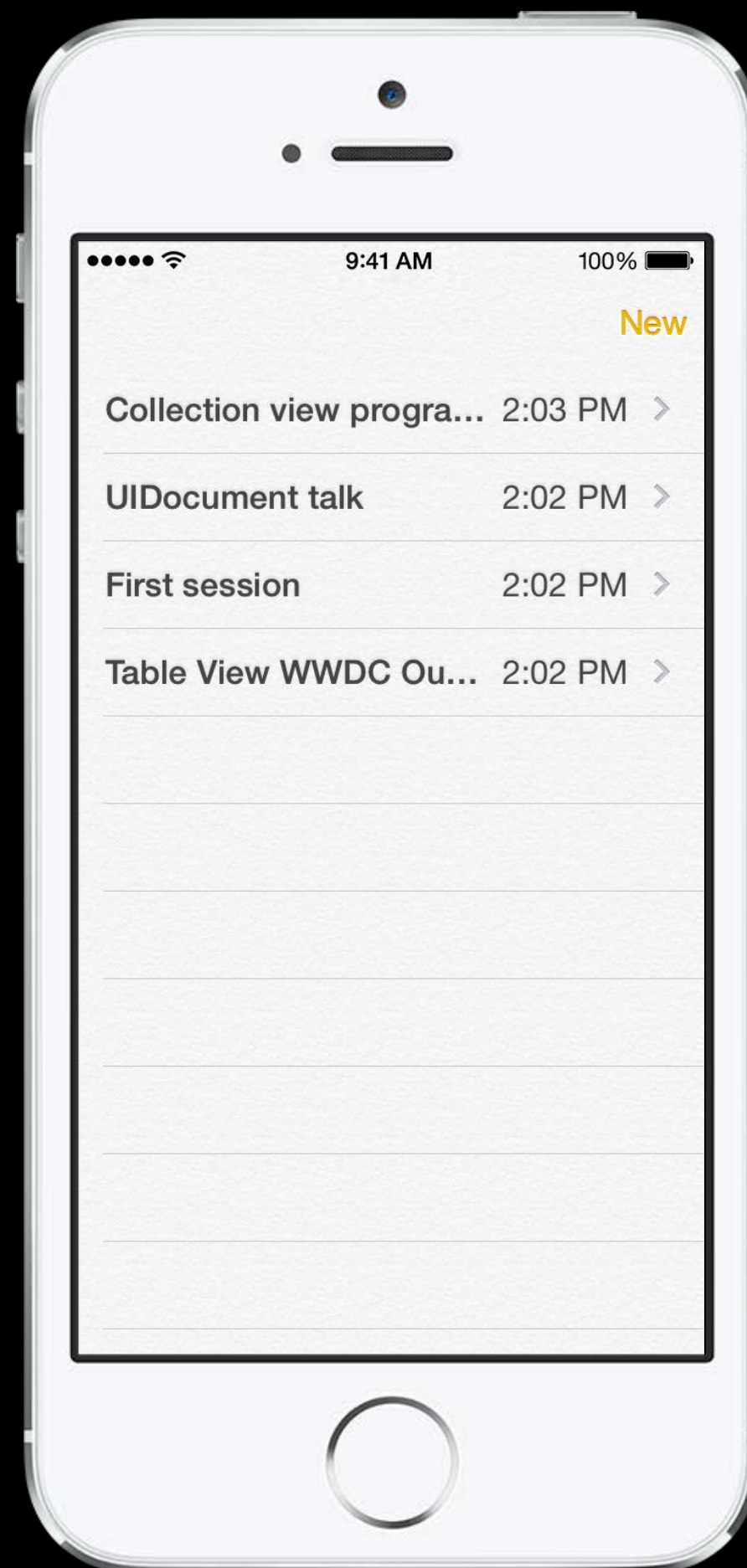
iOS 8 → Adaptivity

Adaptive View Controllers

Adaptive View Controllers

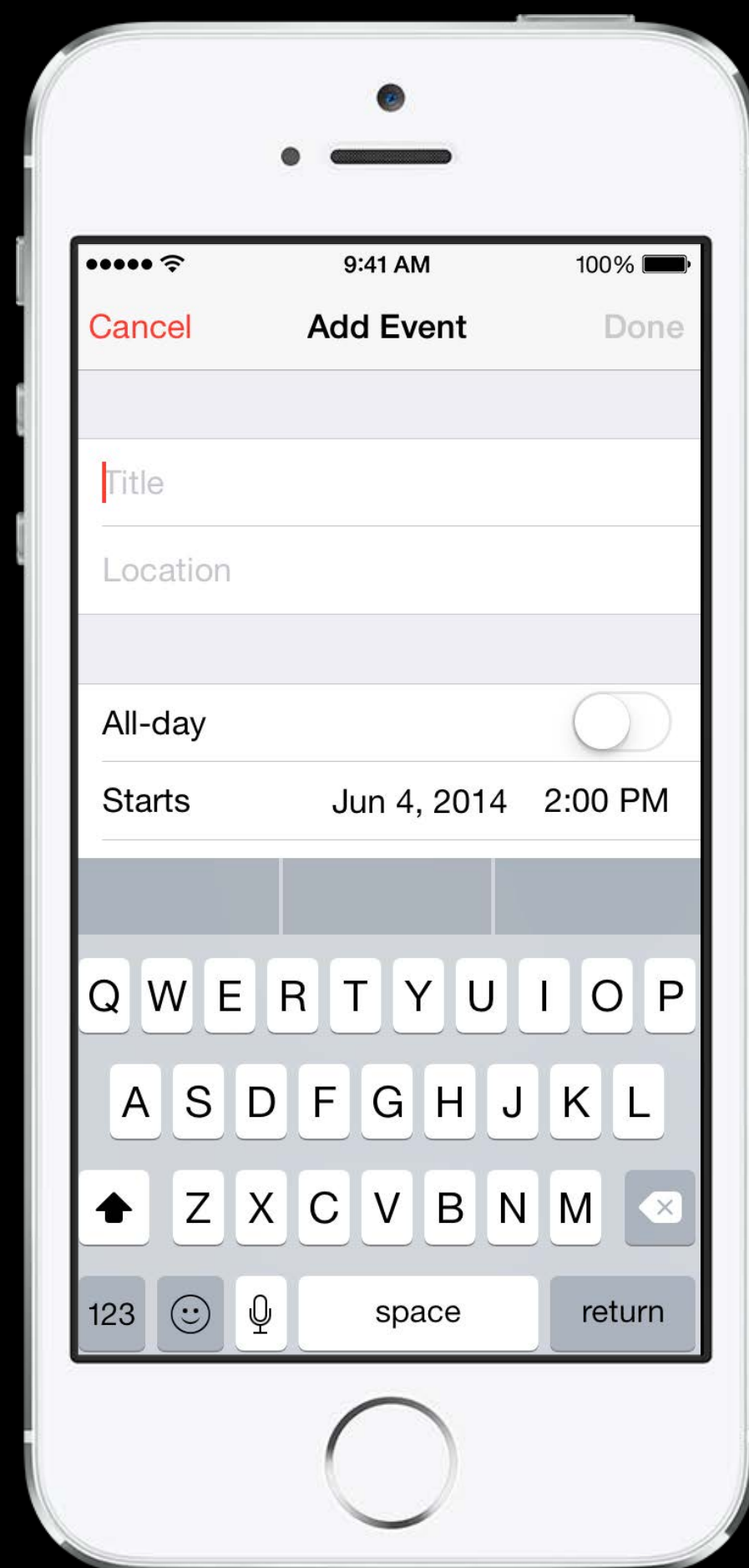


Adaptive View Controllers

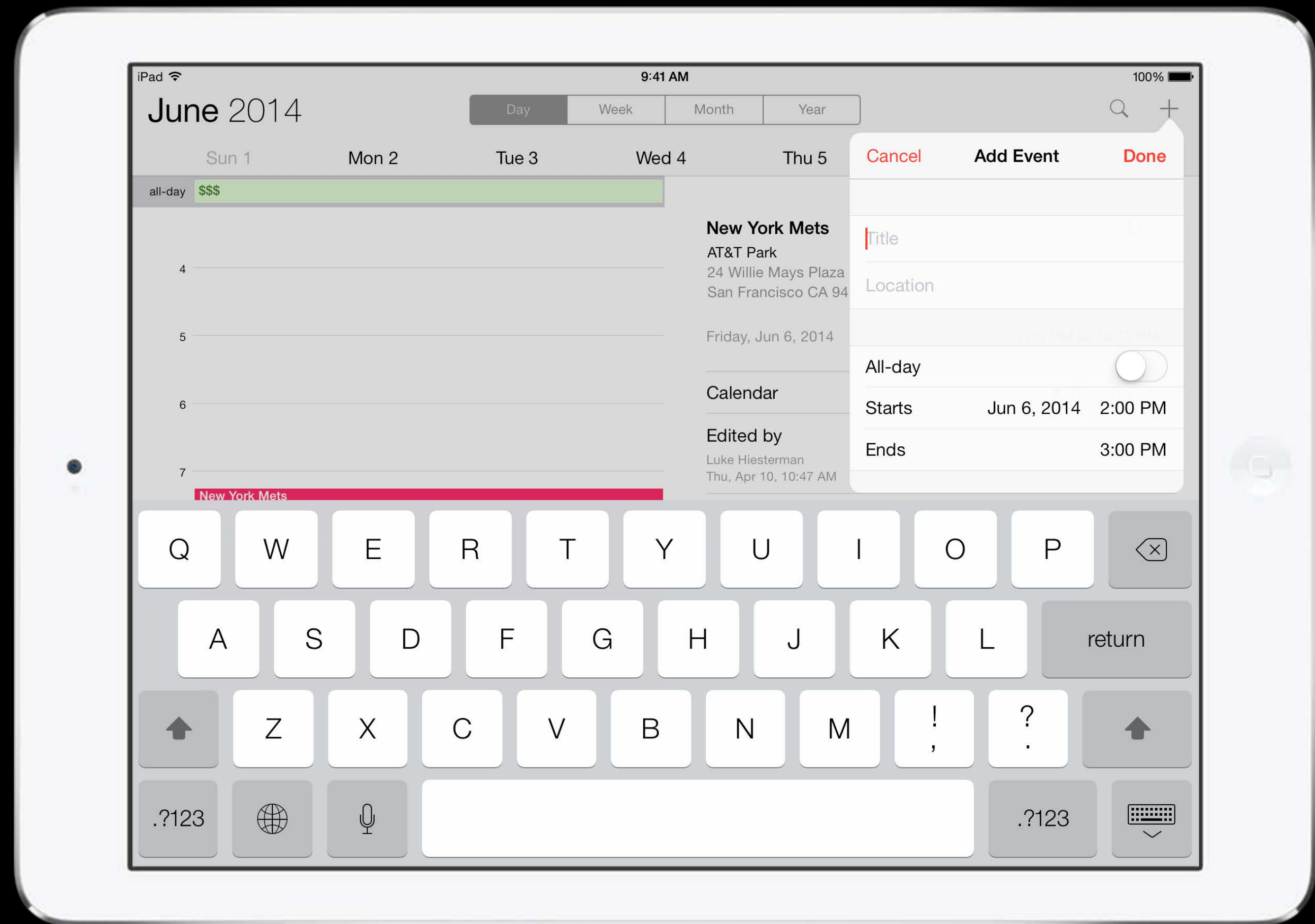
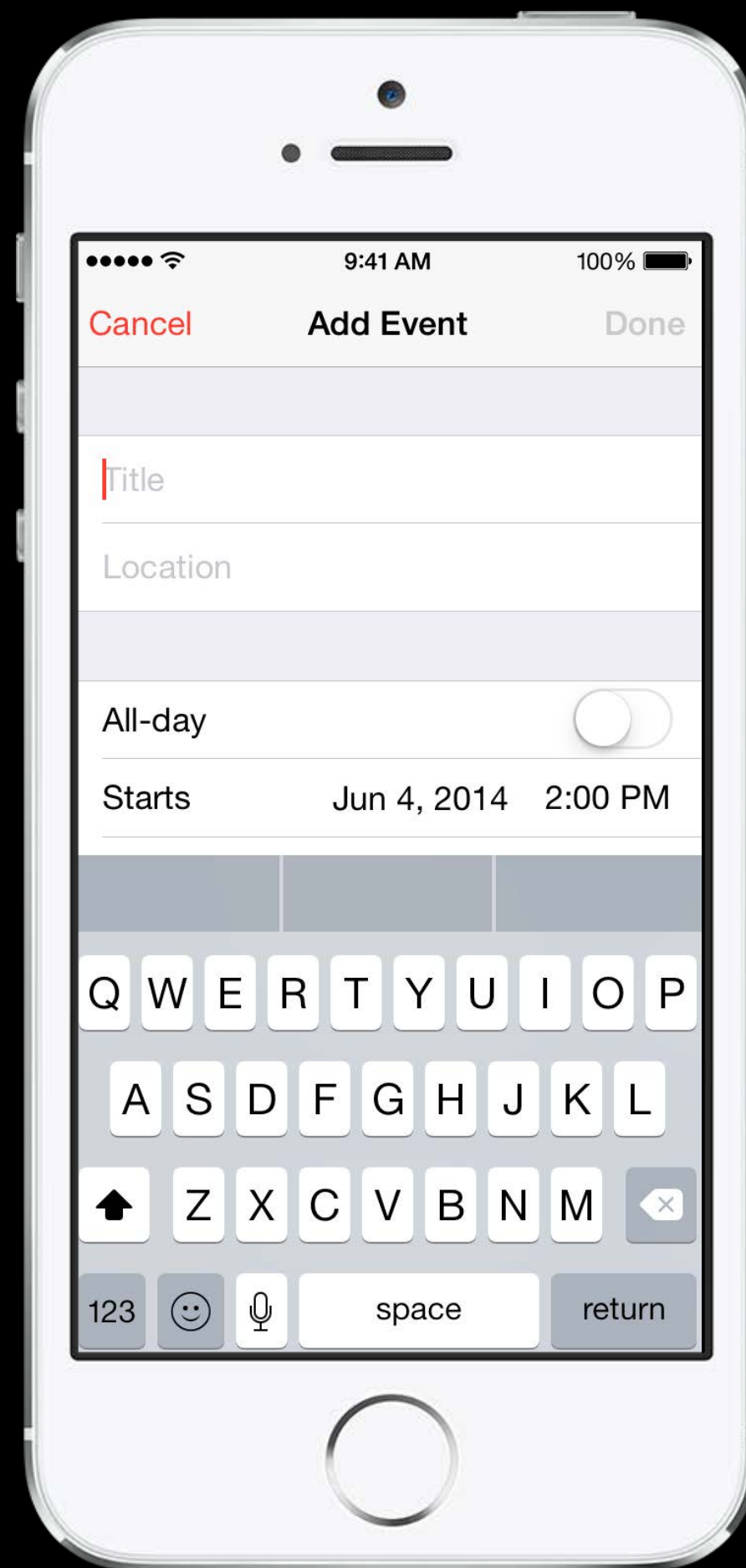


Adaptive Presentations

Adaptive Presentations

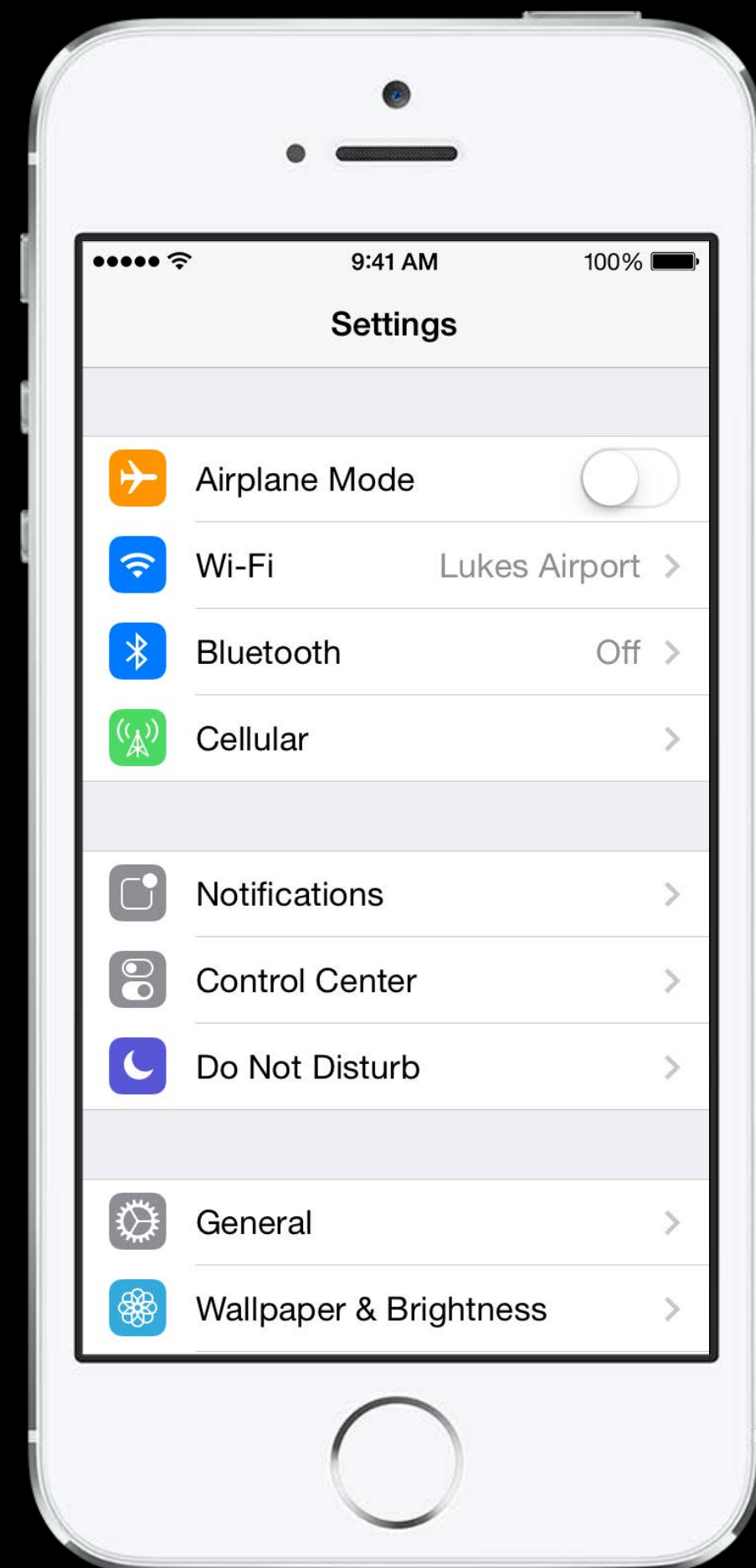


Adaptive Presentations



Adaptive Text and Tables

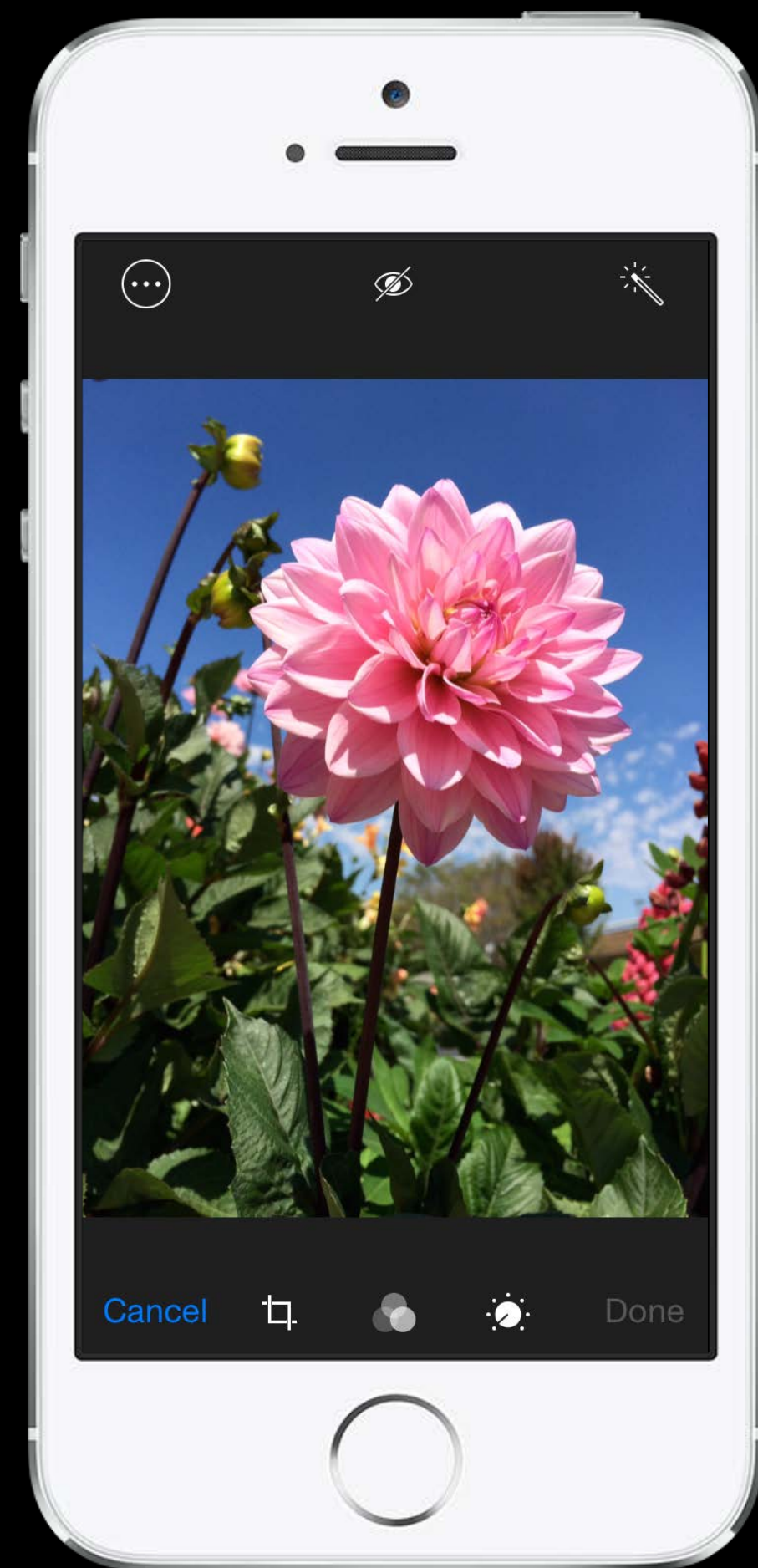
Adaptive Text and Tables



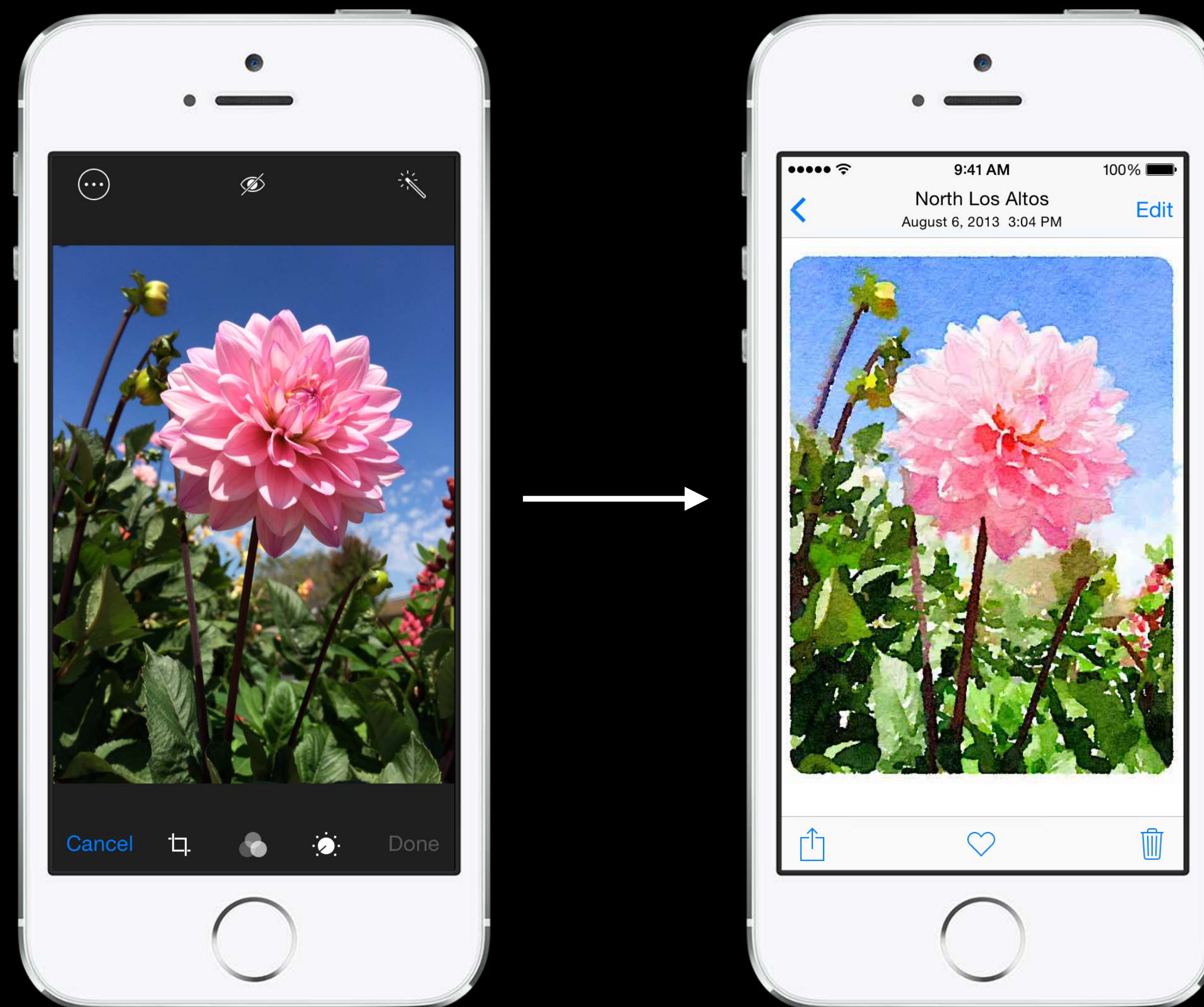
Adaptive Text and Tables



App Extensions



App Extensions



Adaptive Layout

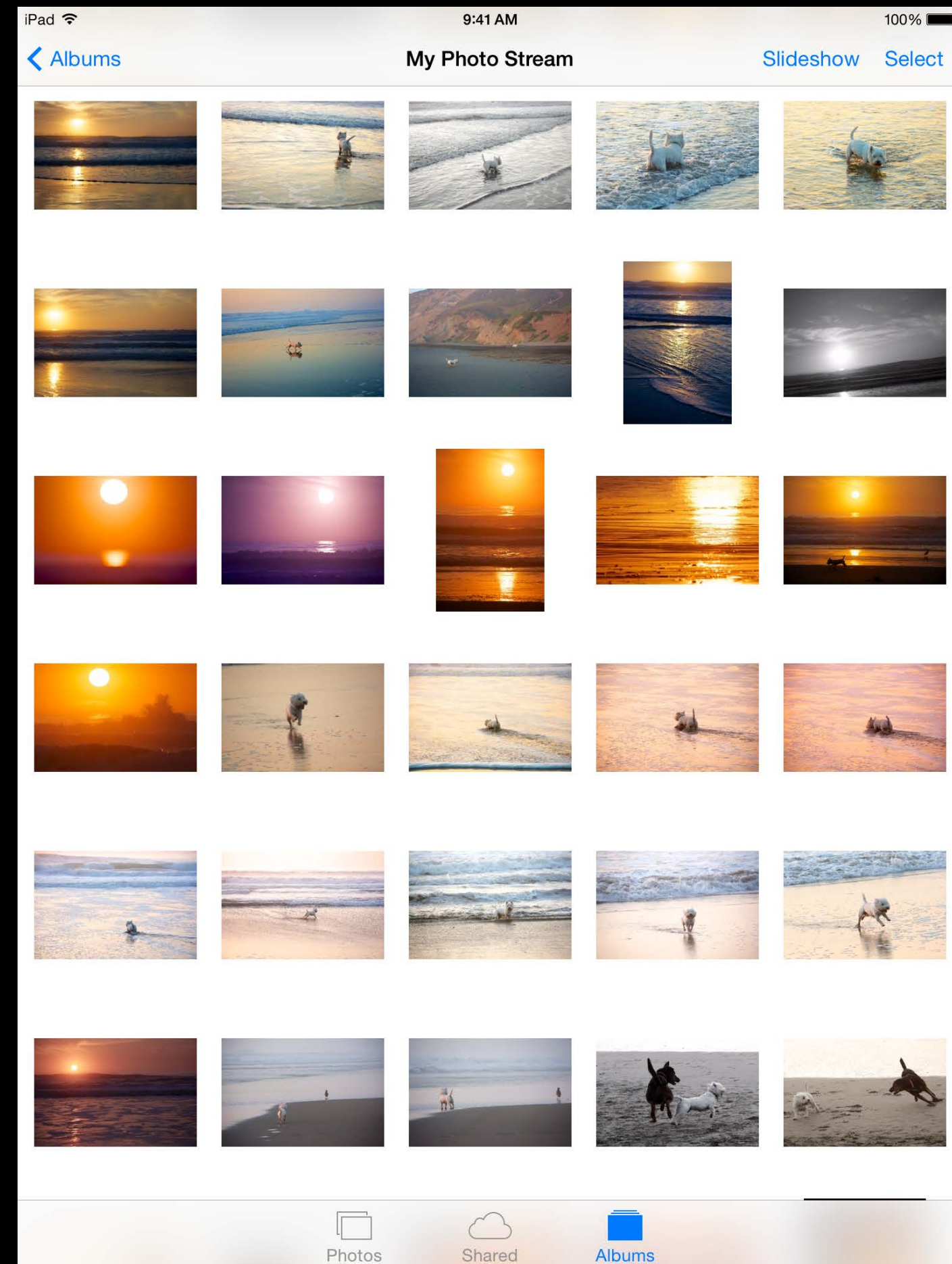
Orientations, sizes, and margins

Adaptive Layouts

Interface orientation

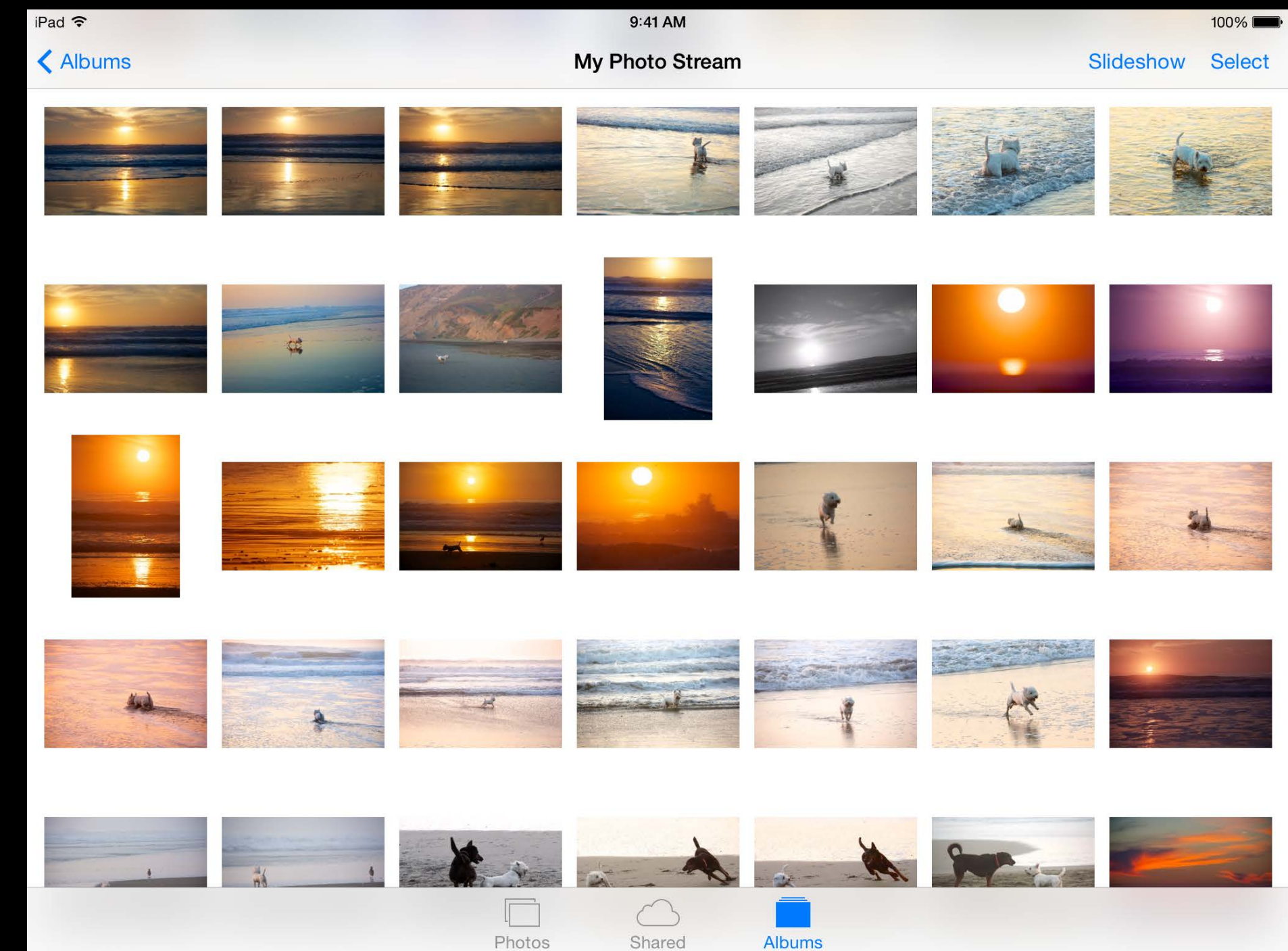
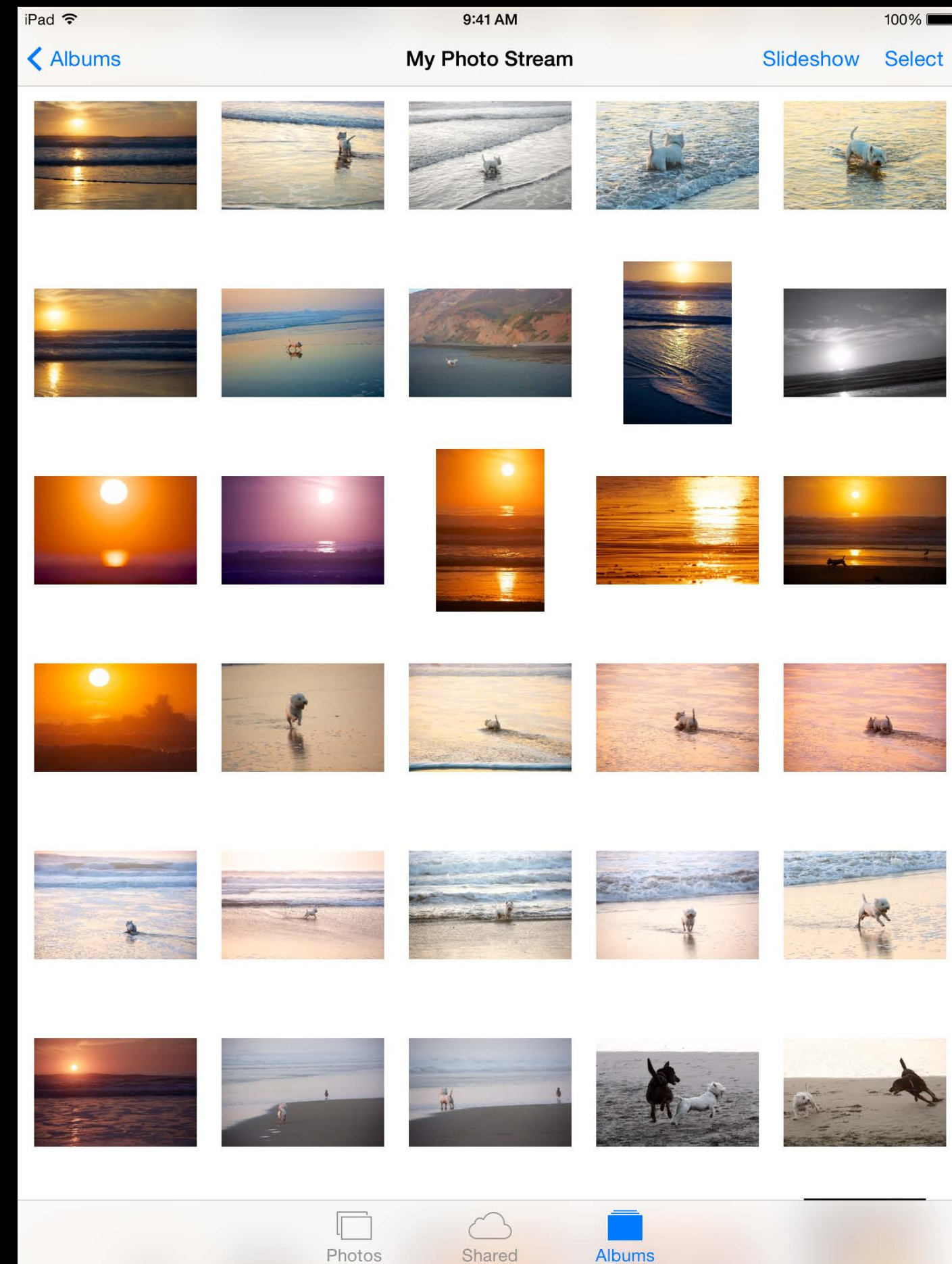
Adaptive Layouts

Interface orientation



Adaptive Layouts

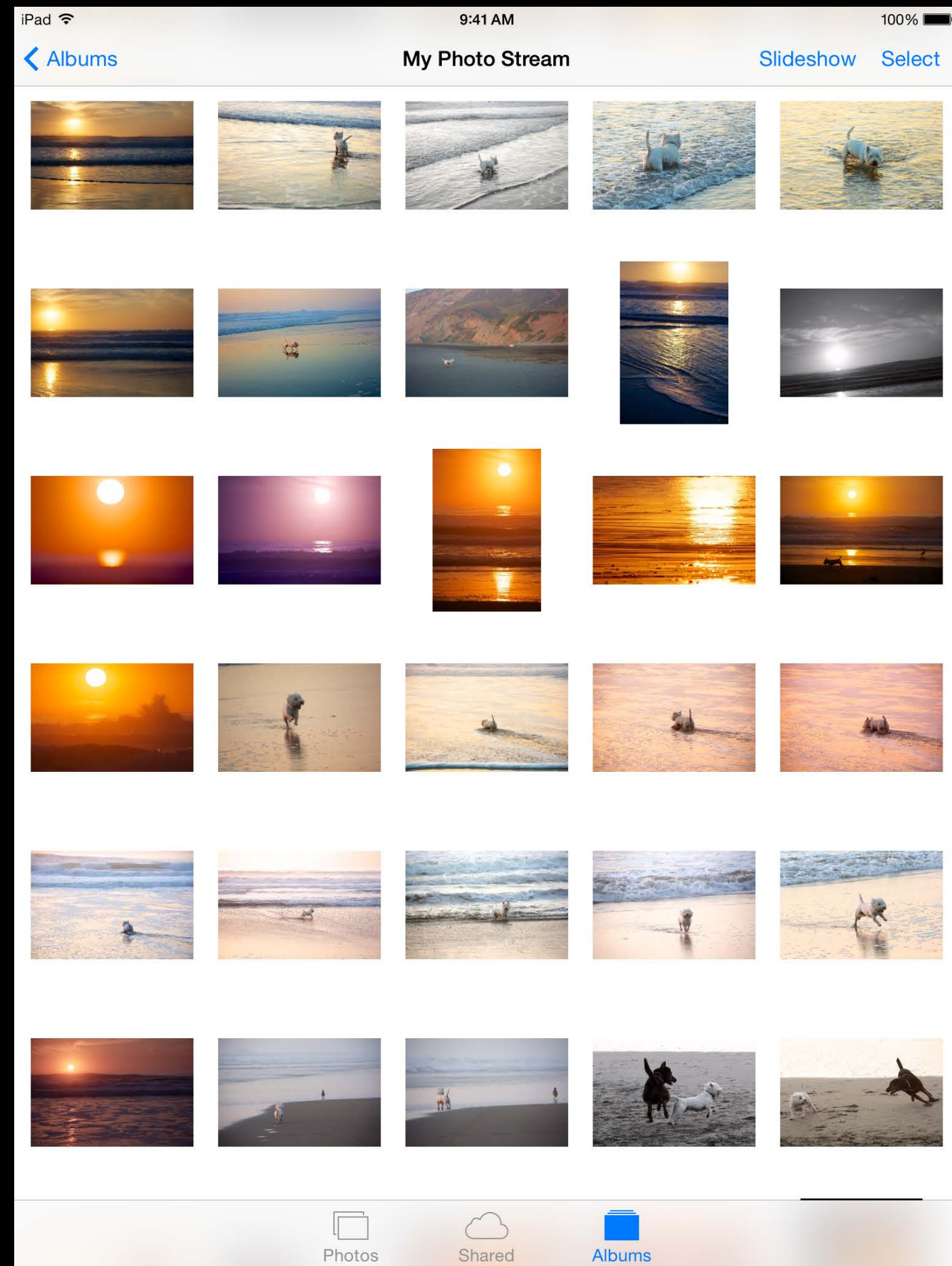
Interface orientation



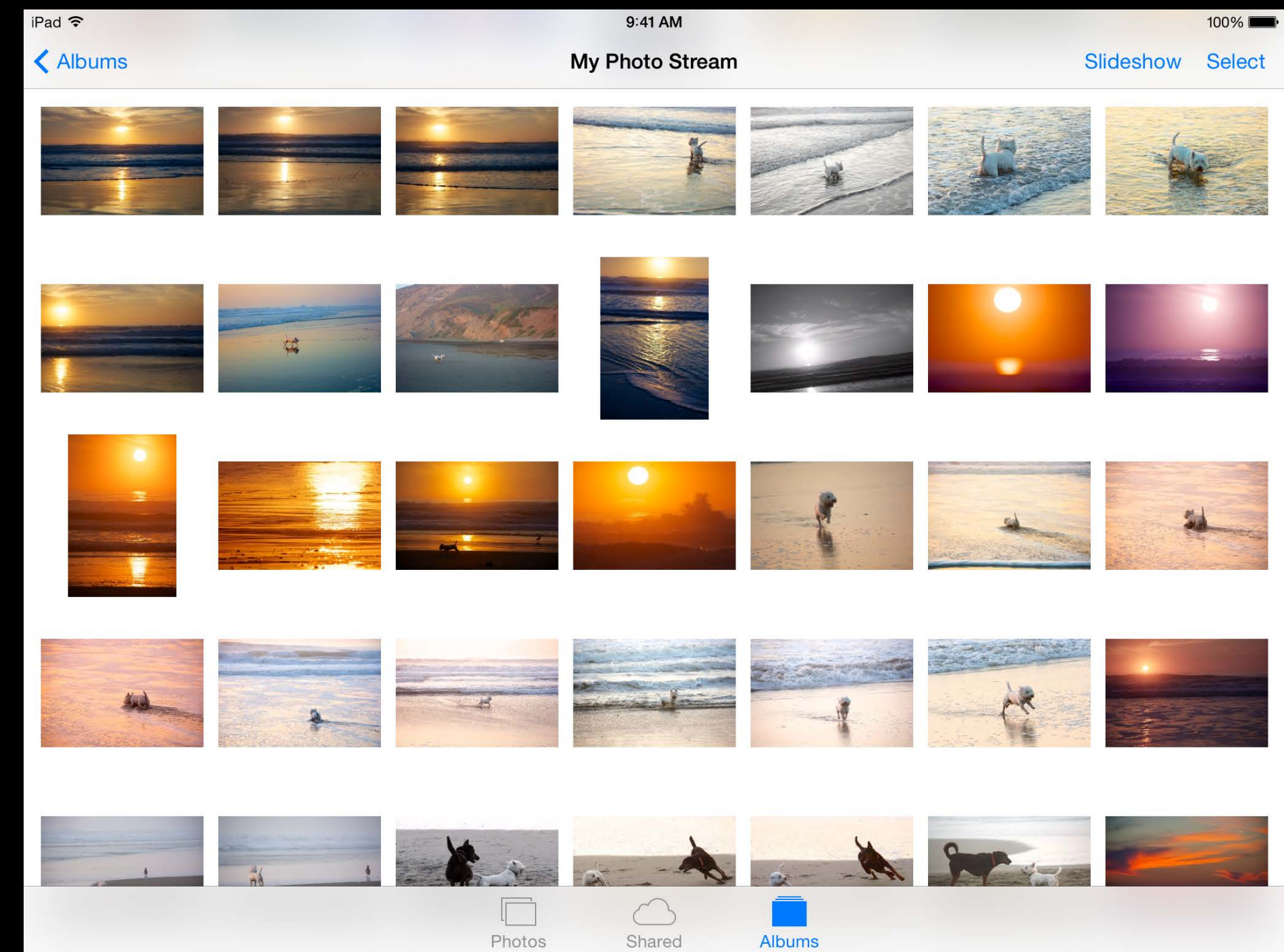
Adaptive Layouts

Interface orientation

1024

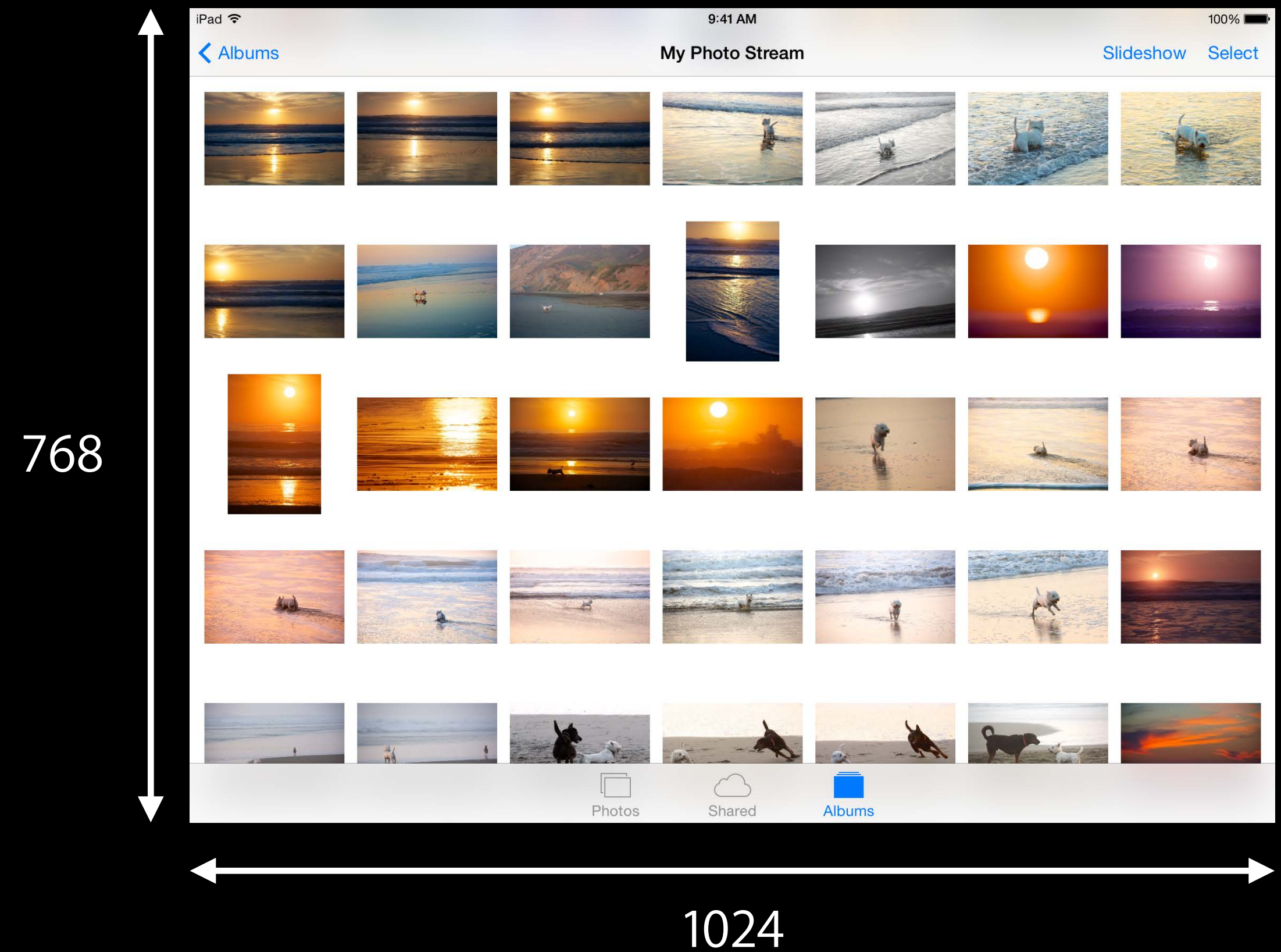
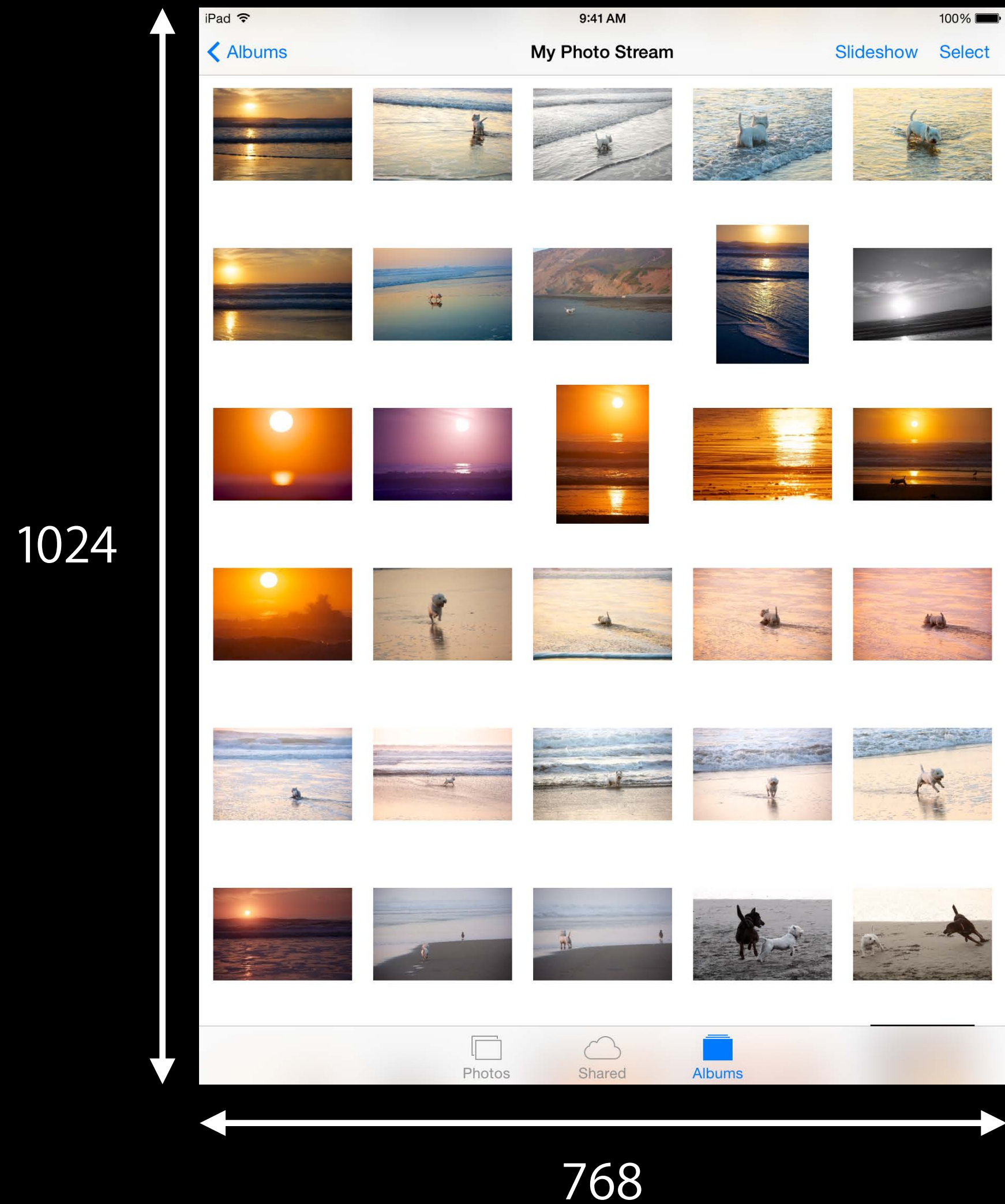


768



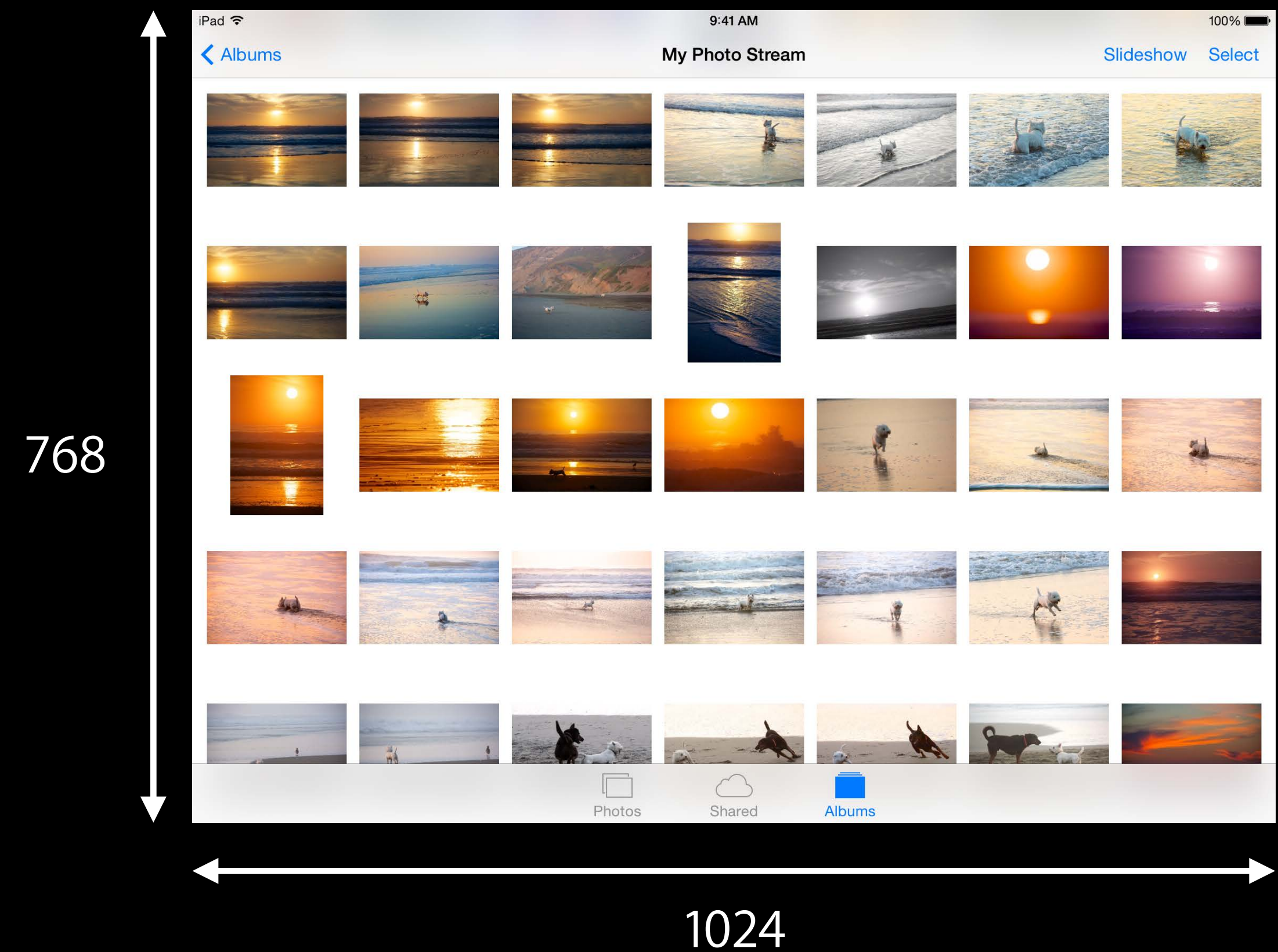
Adaptive Layouts

Interface orientation



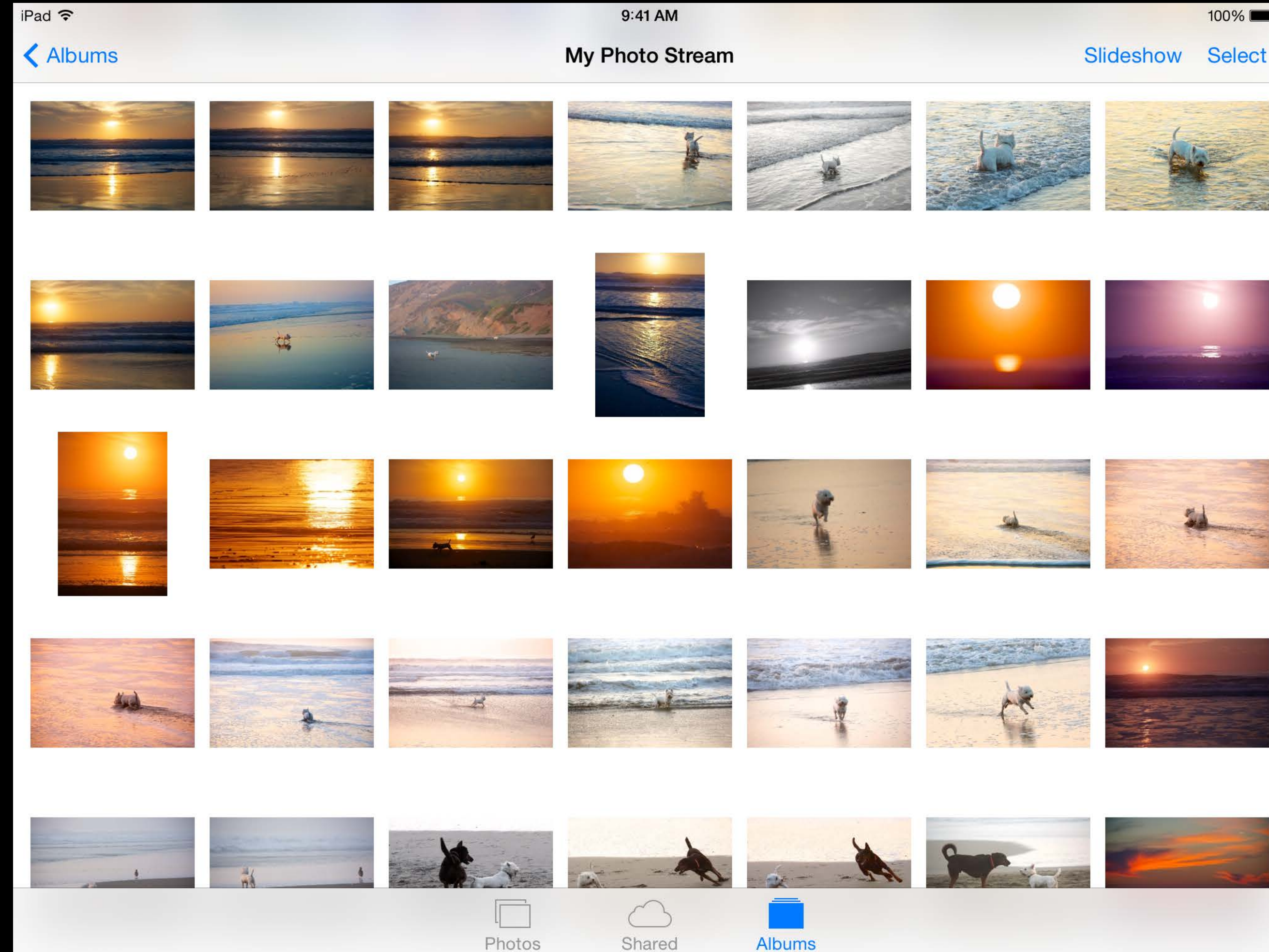
Adaptive Layouts

Interface orientation



Adaptive Layouts

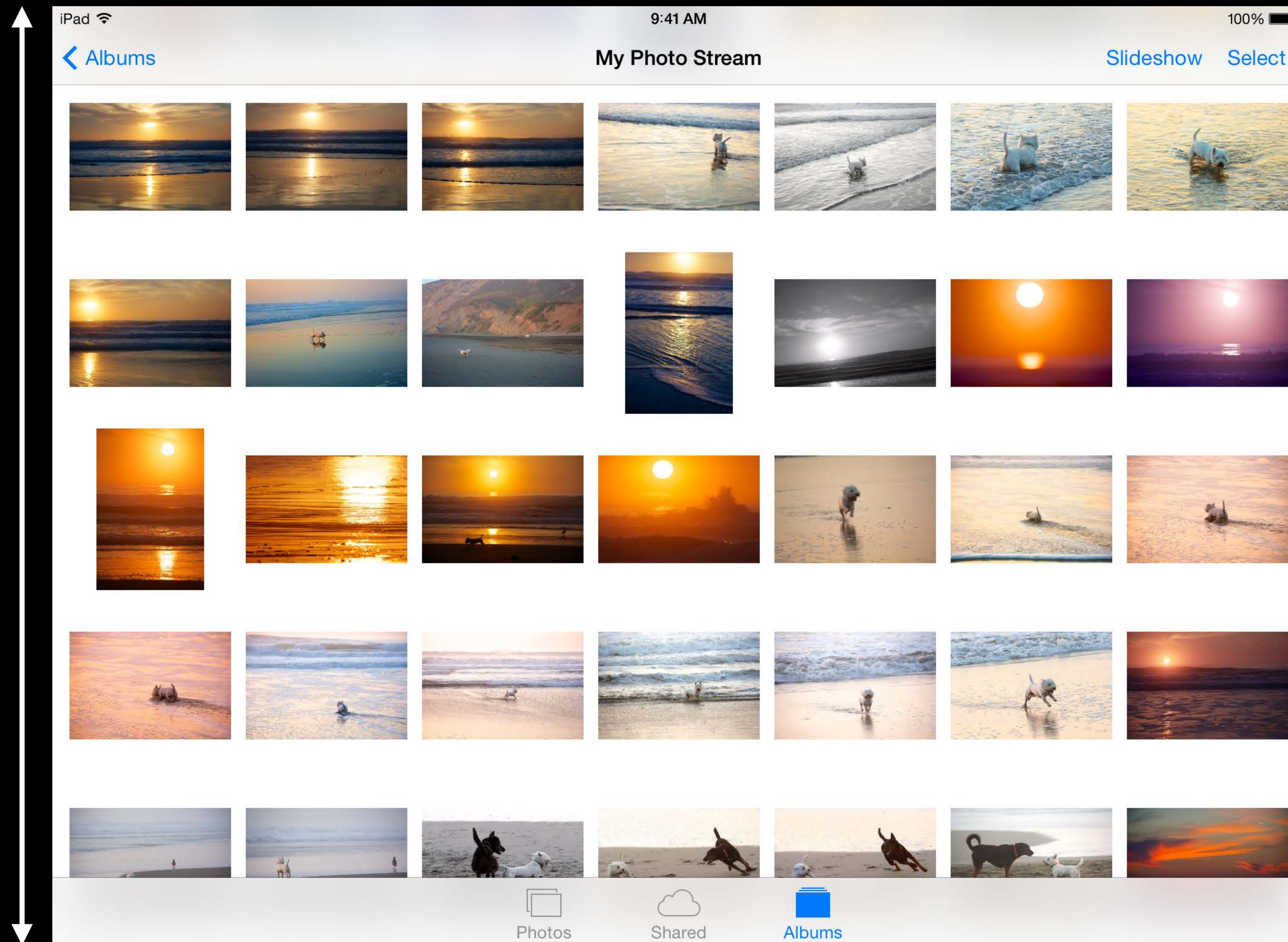
Sizing



Adaptive Layouts

Sizing

Regular Height



Regular Width

Size Classes

The new interface orientation

Size Classes

The new interface orientation

Canvas size rather than interface orientation

Size Classes

The new interface orientation

Canvas size rather than interface orientation

Independent horizontal and vertical-size classes

Size Classes

The new interface orientation

Canvas size rather than interface orientation

Independent horizontal and vertical-size classes

- Compact

Size Classes

The new interface orientation

Canvas size rather than interface orientation

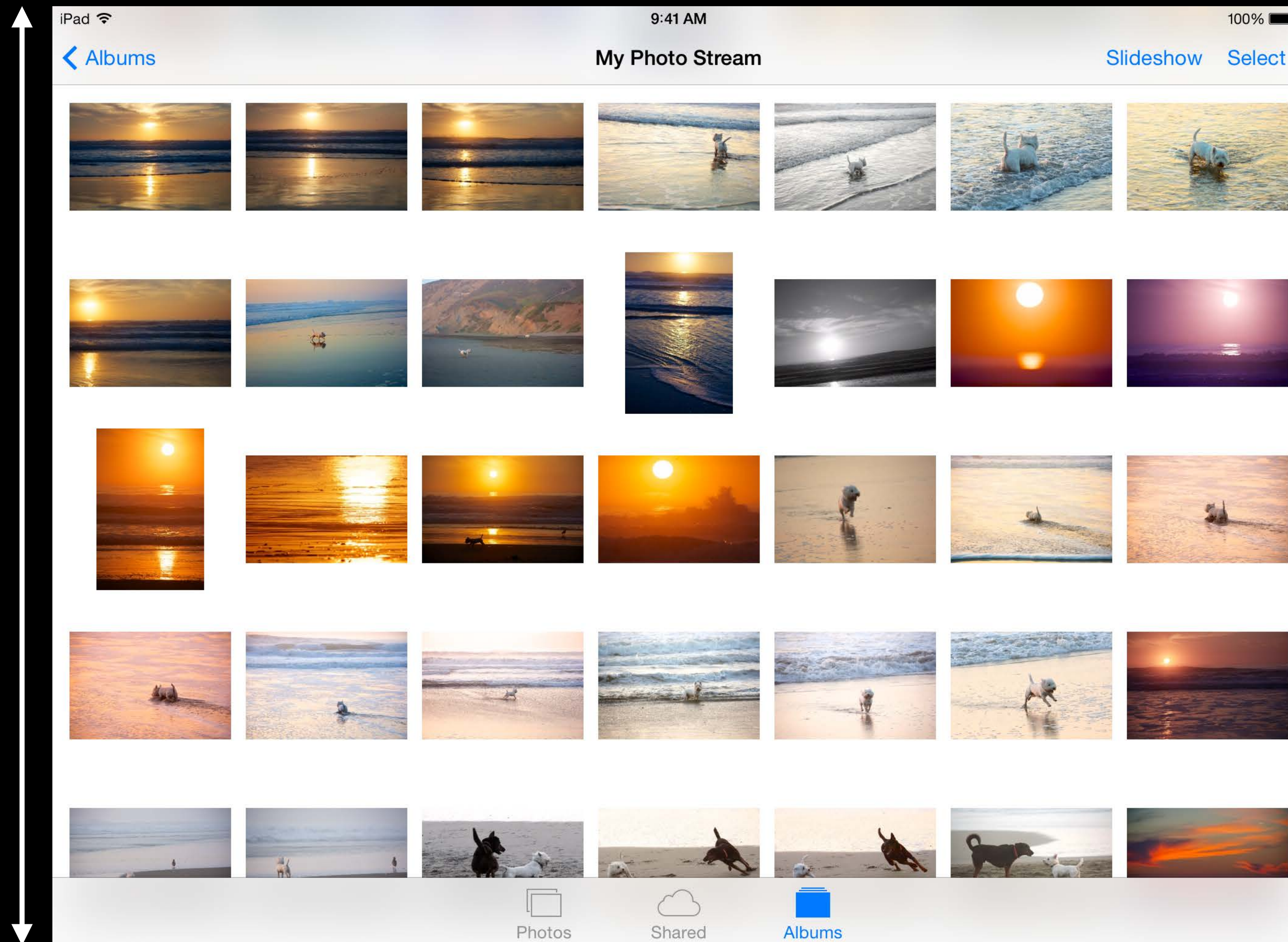
Independent horizontal and vertical-size classes

- Compact
- Regular

Adaptive Layouts

Sizing

Regular Height

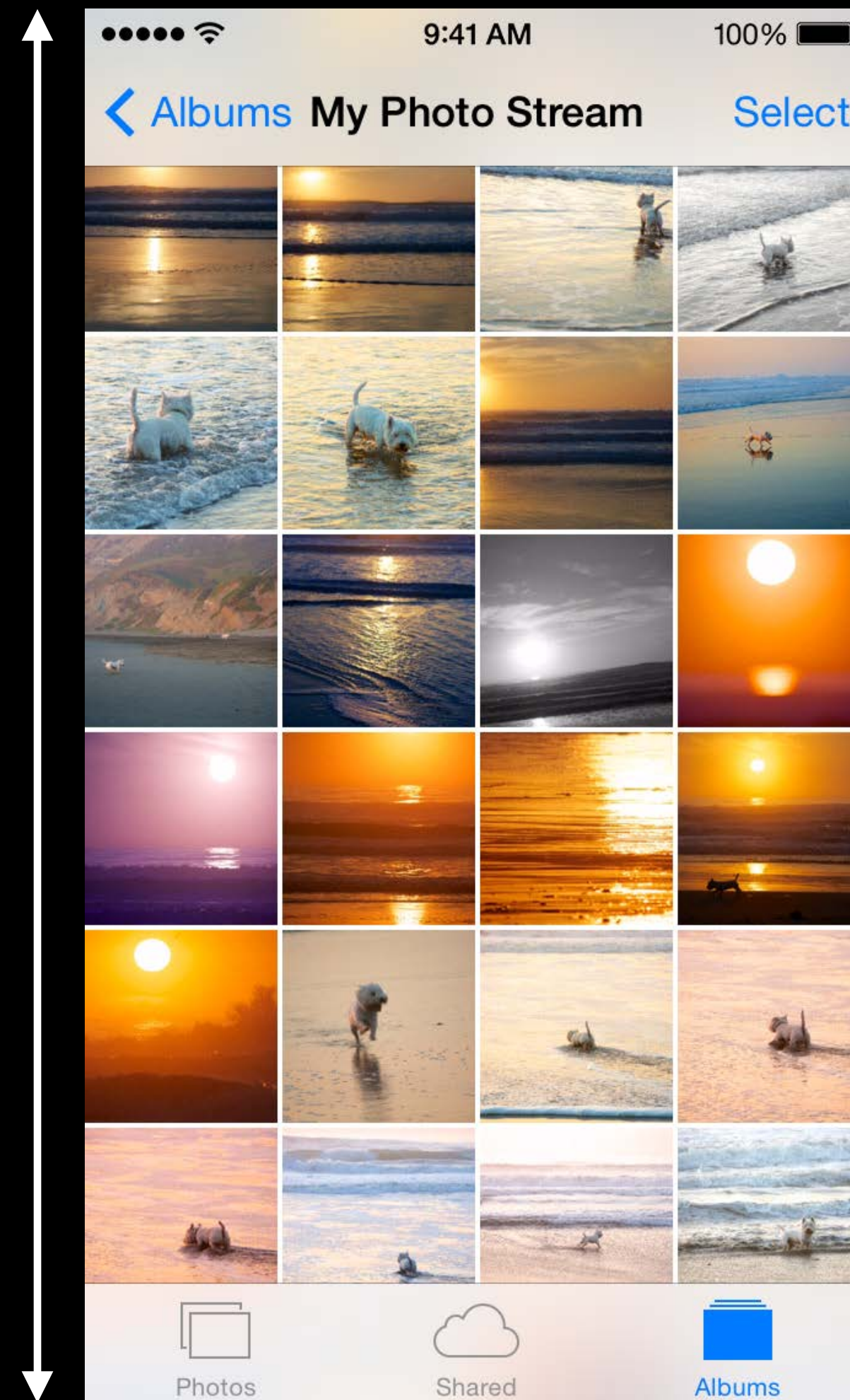


Regular Width

Adaptive Layouts

Sizing

Normal Height

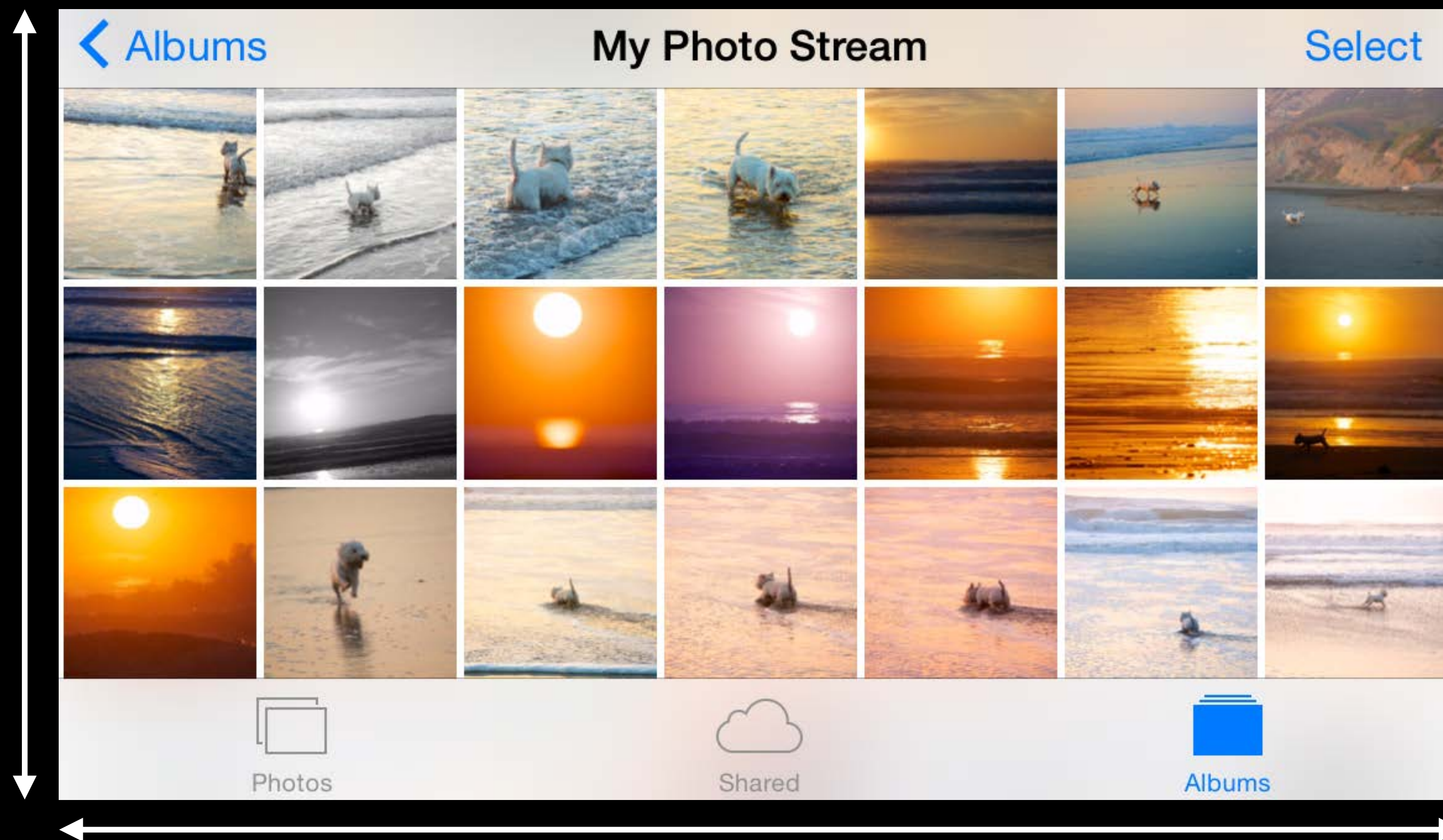


Compact Width

Adaptive Layouts

Sizing

Compact Height



Compact Width

Trait Collections

Everything you need to know to layout

Trait Collections

Everything you need to know to layout

horizontalSizeClass

Trait Collections

Everything you need to know to layout

horizontalSizeClass

verticalSizeClass

Trait Collections

Everything you need to know to layout

horizontalSizeClass

verticalSizeClass

displayScale

Trait Collections

Everything you need to know to layout

horizontalSizeClass

verticalSizeClass

displayScale

userInterfaceIdiom

Trait Collections

Everything you need to know to layout

```
override func traitCollectionDidChange(previousTraitCollection: UITraitCollection!) {
```

```
}
```

Trait Collections

Everything you need to know to layout

```
override func traitCollectionDidChange(previousTraitCollection: UITraitCollection!) {
```

```
}
```

Trait Collections

Everything you need to know to layout

```
override func traitCollectionDidChange(previousTraitCollection: UITraitCollection!) {  
    let currentTraits = self.traitCollection  
  
}
```

Trait Collections

Everything you need to know to layout

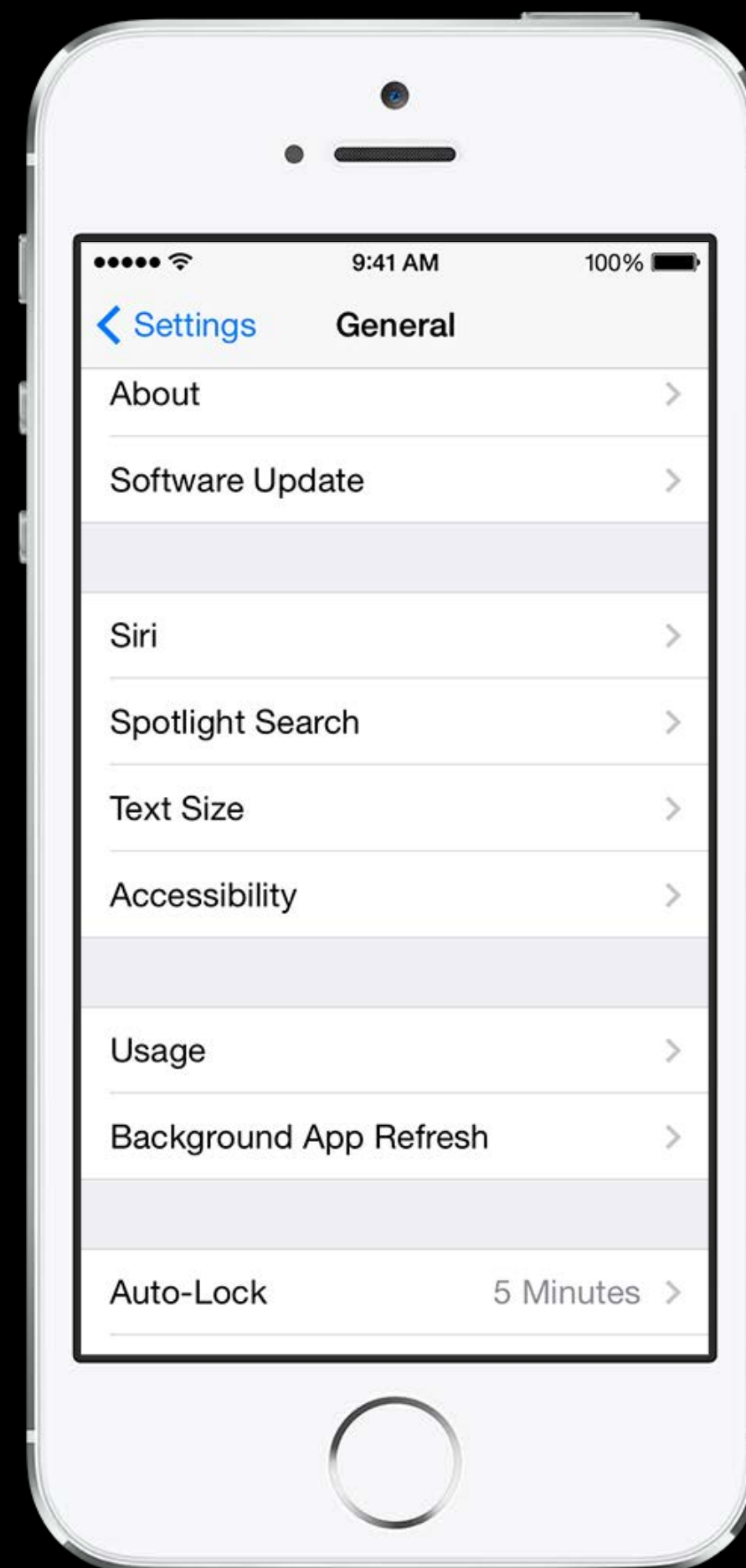
```
override func traitCollectionDidChange(previousTraitCollection: UITraitCollection!) {  
    let currentTraits = self.traitCollection  
    let newLayout = currentTraits.horizontalSizeClass == UIUserInterfaceSizeClass.Compact  
        ? self.squaresLayout  
        : self.rectanglesLayout  
}
```

Trait Collections

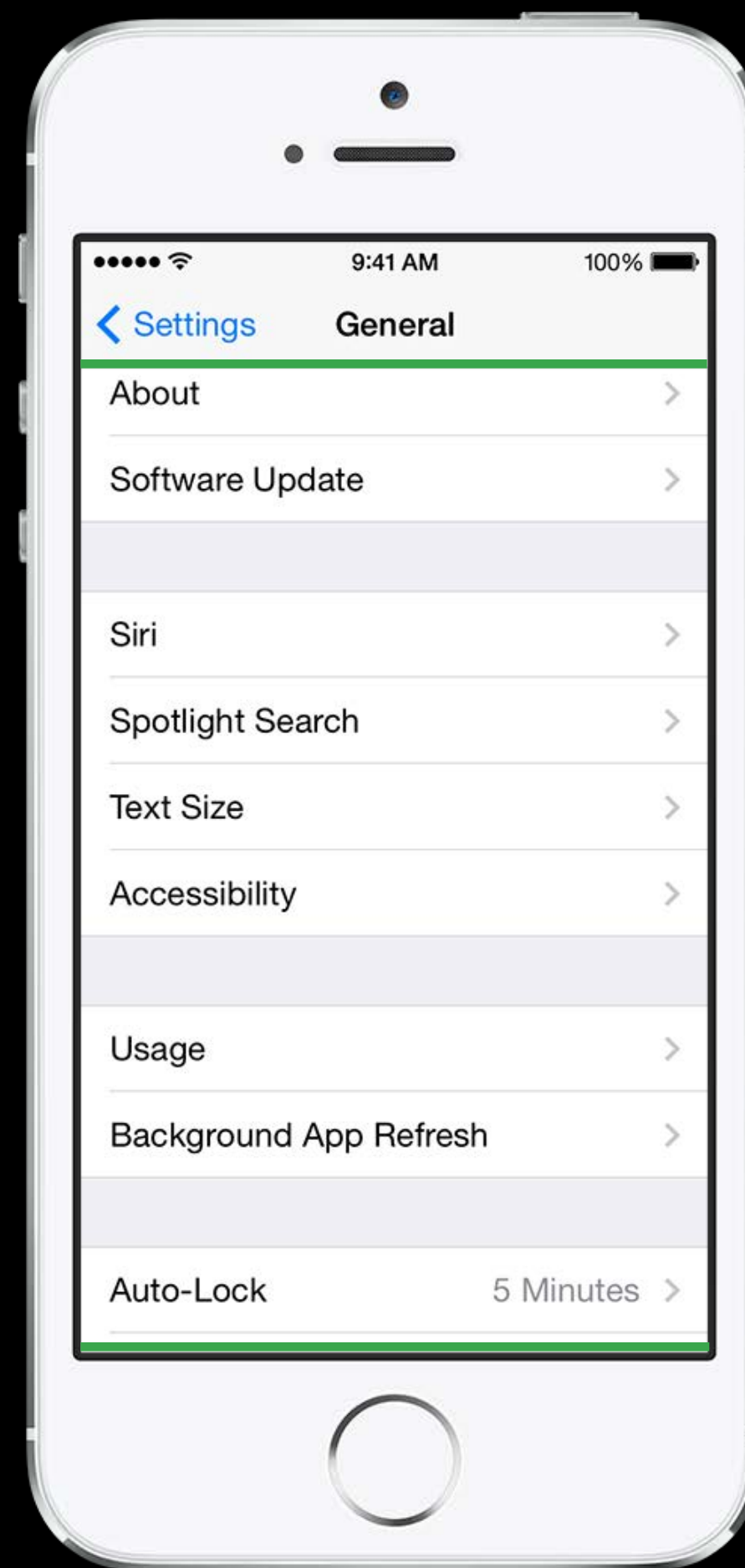
Everything you need to know to layout

```
override func traitCollectionDidChange(previousTraitCollection: UITraitCollection!) {  
    let currentTraits = self.traitCollection  
    let newLayout = currentTraits.horizontalSizeClass == UIUserInterfaceSizeClass.Compact  
        ? self.squaresLayout  
        : self.rectanglesLayout  
    self.collectionView.setCollectionViewLayout(newLayout, animated: true)  
}
```


Adaptive Margins



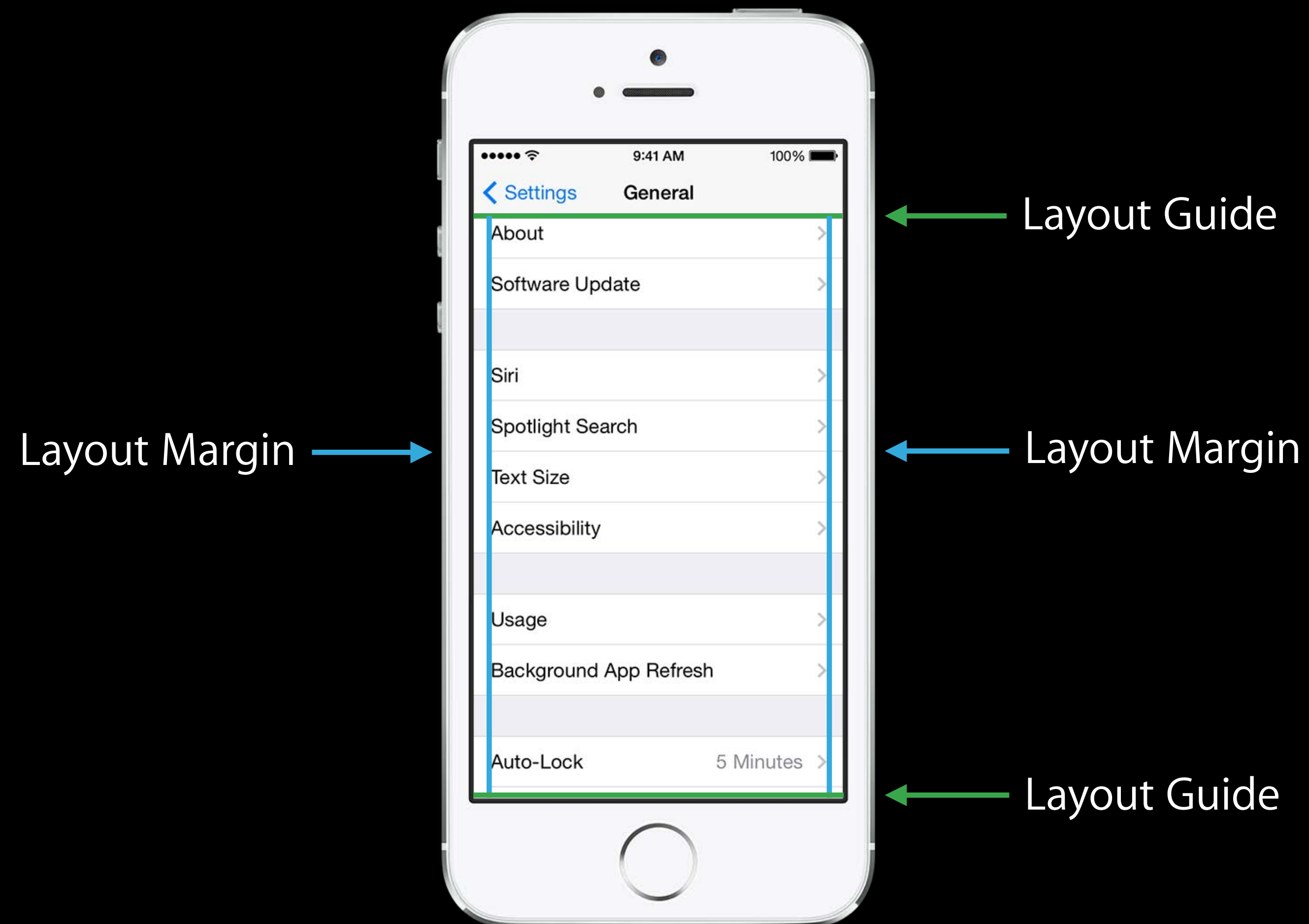
Adaptive Margins



Layout Guide

Layout Guide

Adaptive Margins



Adaptive Margins

Auto Layout support

Adaptive Margins

Auto Layout support

New margin NSLayoutAttributes

Adaptive Margins

Auto Layout support

New margin `NSLayoutAttributes`

- `LeftMargin`

Adaptive Margins

Auto Layout support

New margin NSLayoutAttributes

- LeftMargin
- RightMargin

Adaptive Margins

Auto Layout support

New margin NSLayoutAttributes

- LeftMargin
- RightMargin
- LeadingMargin

Adaptive Margins

Auto Layout support

New margin NSLayoutAttributes

- LeftMargin
- RightMargin
- LeadingMargin
- ...etc

More Adaptive Layout Information

-
- Building Adaptive Apps with UIKit

Mission

Wednesday 10:15AM

Adaptive View Controllers

Rotation

What is rotation, really?

Rotation

What is rotation, really?

Rotation is an animated bounds change

Rotation

What is rotation, really?

Rotation is an animated bounds change



Rotation

What is rotation, really?

Rotation is an animated bounds change



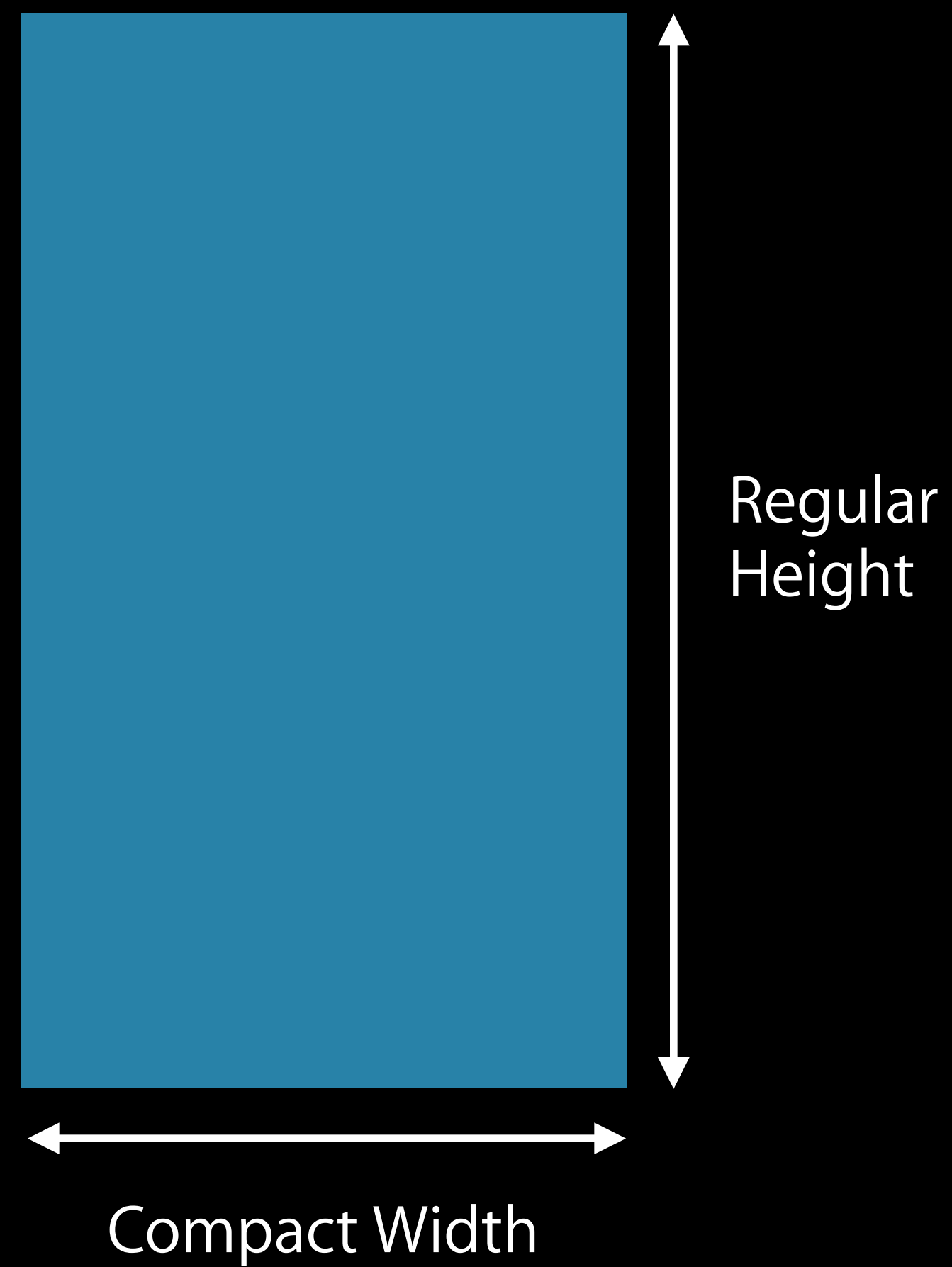
Rotation

What is rotation, really?



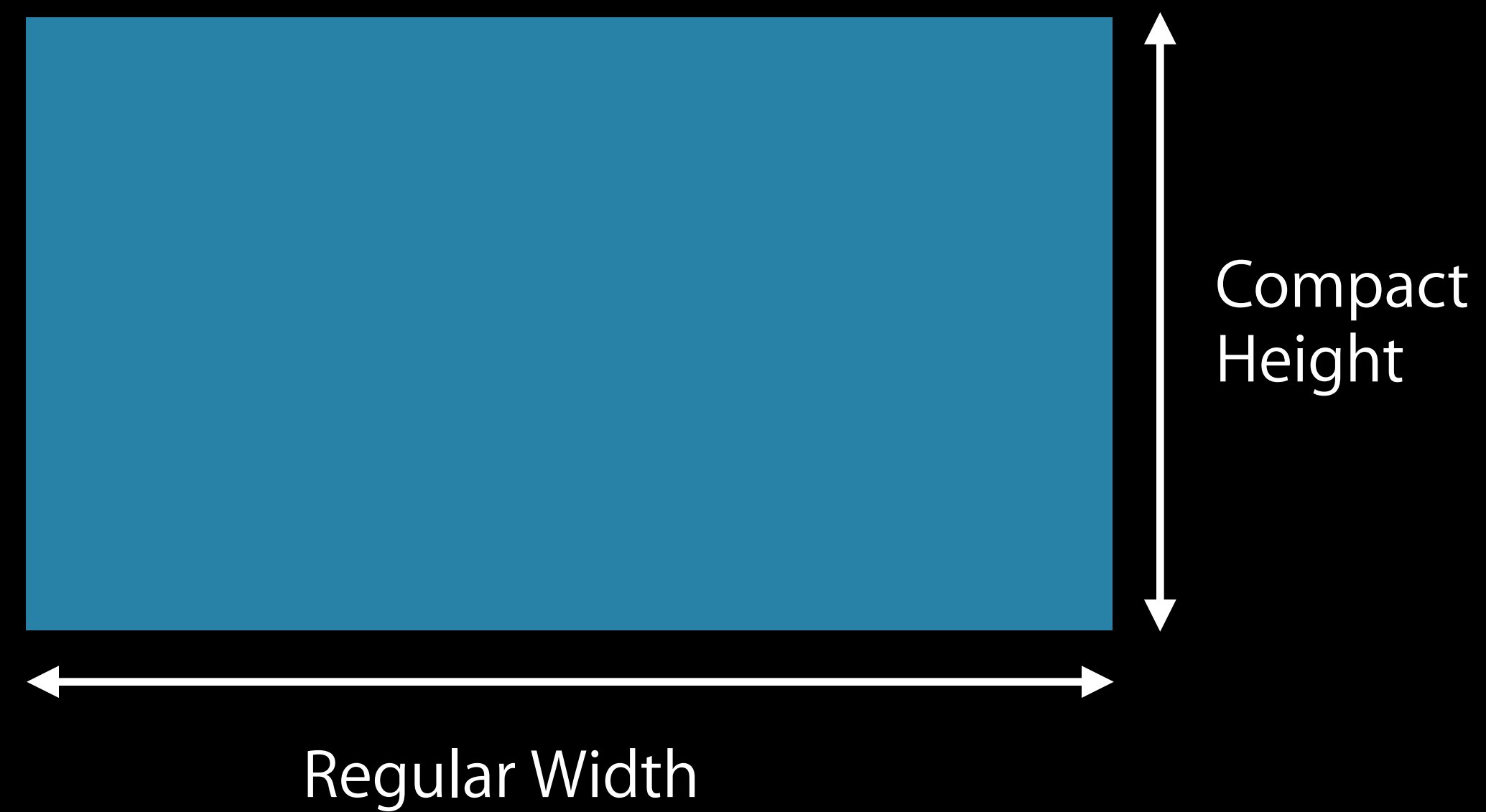
Rotation

What is rotation, really?



Rotation

What is rotation, really?



Rotation

Deprecations

Rotation

Deprecations

```
-(void)willRotateToInterfaceOrientation:  
(UIInterfaceOrientation)toInterfaceOrientation duration:  
(NSTimeInterval)duration;  
-(void)willAnimateRotationToInterfaceOrientation:  
(UIInterfaceOrientation)toInterfaceOrientation duration:  
(NSTimeInterval)duration;  
-(void)didRotateFromInterfaceOrientation:  
(UIInterfaceOrientation)fromInterfaceOrientation;  
-(BOOL)shouldAutomaticallyForwardRotationMethods;  
-(UIInterfaceOrientation)interfaceOrientation;  
-(void)rotatingHeaderView;  
-(void)rotatingFooterView;
```

Rotation

New hotness

```
- (void)viewWillTransitionToSize:(CGSize)size
    withTransitionCoordinator:(id<UIViewControllerTransitionCoordinator>)coordinator
{

}

}
```

Rotation

New hotness

```
- (void)viewWillTransitionToSize:(CGSize)size
    withTransitionCoordinator:(id<UIViewControllerTransitionCoordinator>)coordinator
{
    [coordinator animateAlongsideTransition:^(id<UIViewControllerTransitionCoordinatorContext>
                                             context) {
        [self.collectionView.collectionViewLayout.numberofColumns = [self columnsForSize:size];
    } completion:nil];
}
```

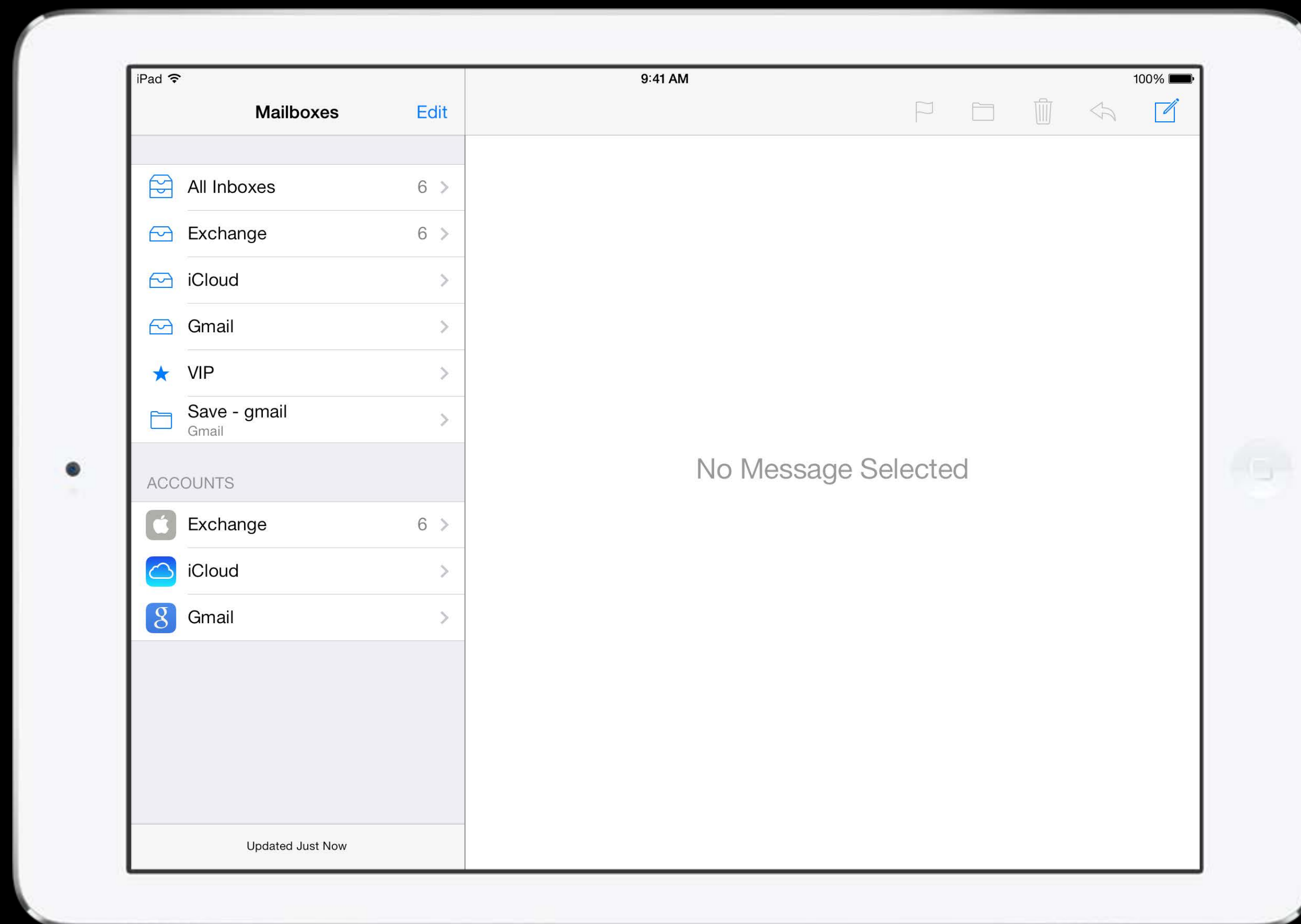
Rotation

New hotness

```
- (void)viewWillTransitionToSize:(CGSize)size
    withTransitionCoordinator:(id<UIViewControllerTransitionCoordinator>)coordinator
{
    [coordinator animateAlongsideTransition:^(id<UIViewControllerTransitionCoordinatorContext>
                                             context) {
        [self.collectionView.collectionViewLayout.numberOfColumns = [self columnsForSize:size];
    } completion:nil];
}
```

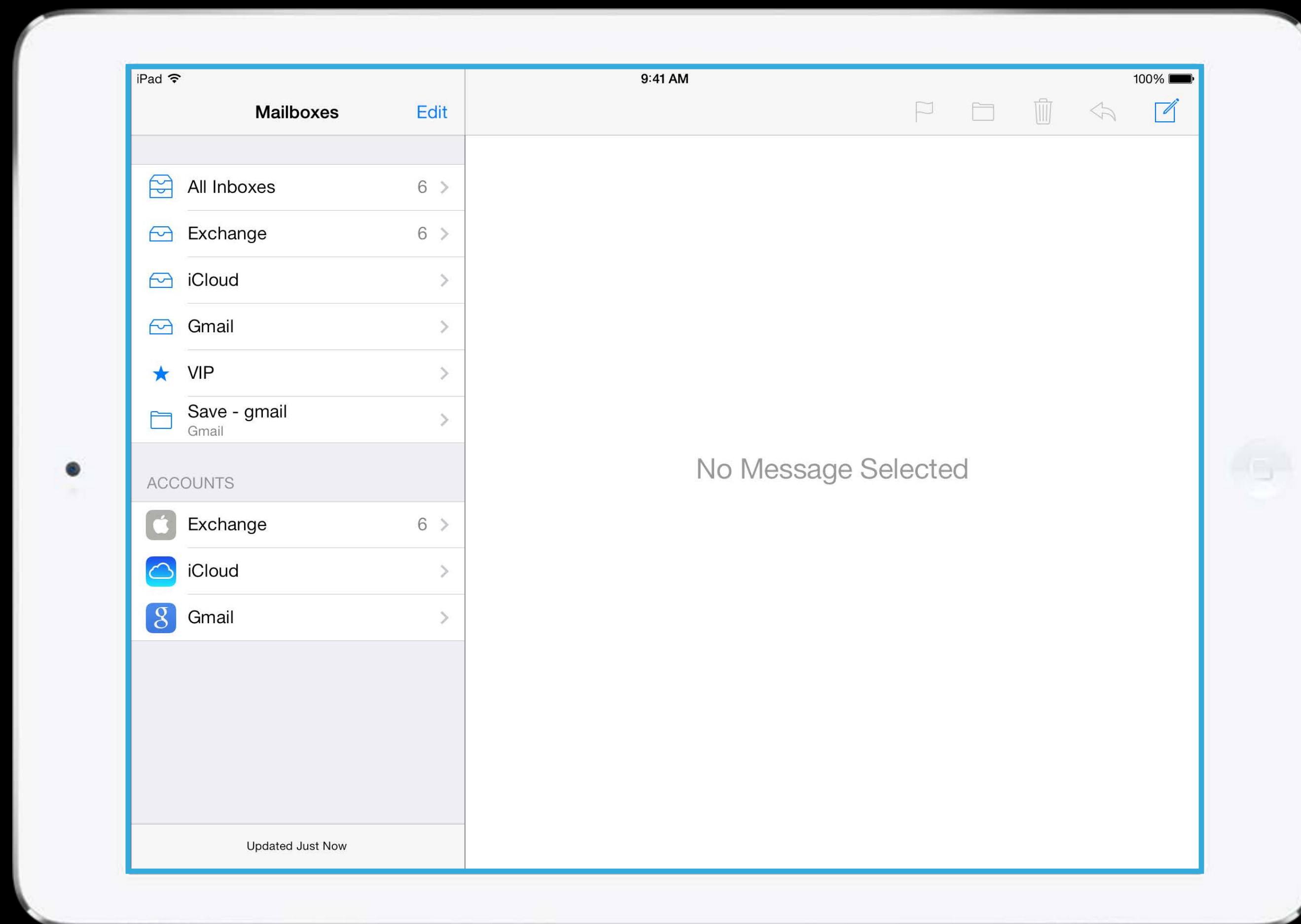
Adaptive View Controller Hierarchies

Simplifying and unifying code



Adaptive View Controller Hierarchies

Simplifying and unifying code

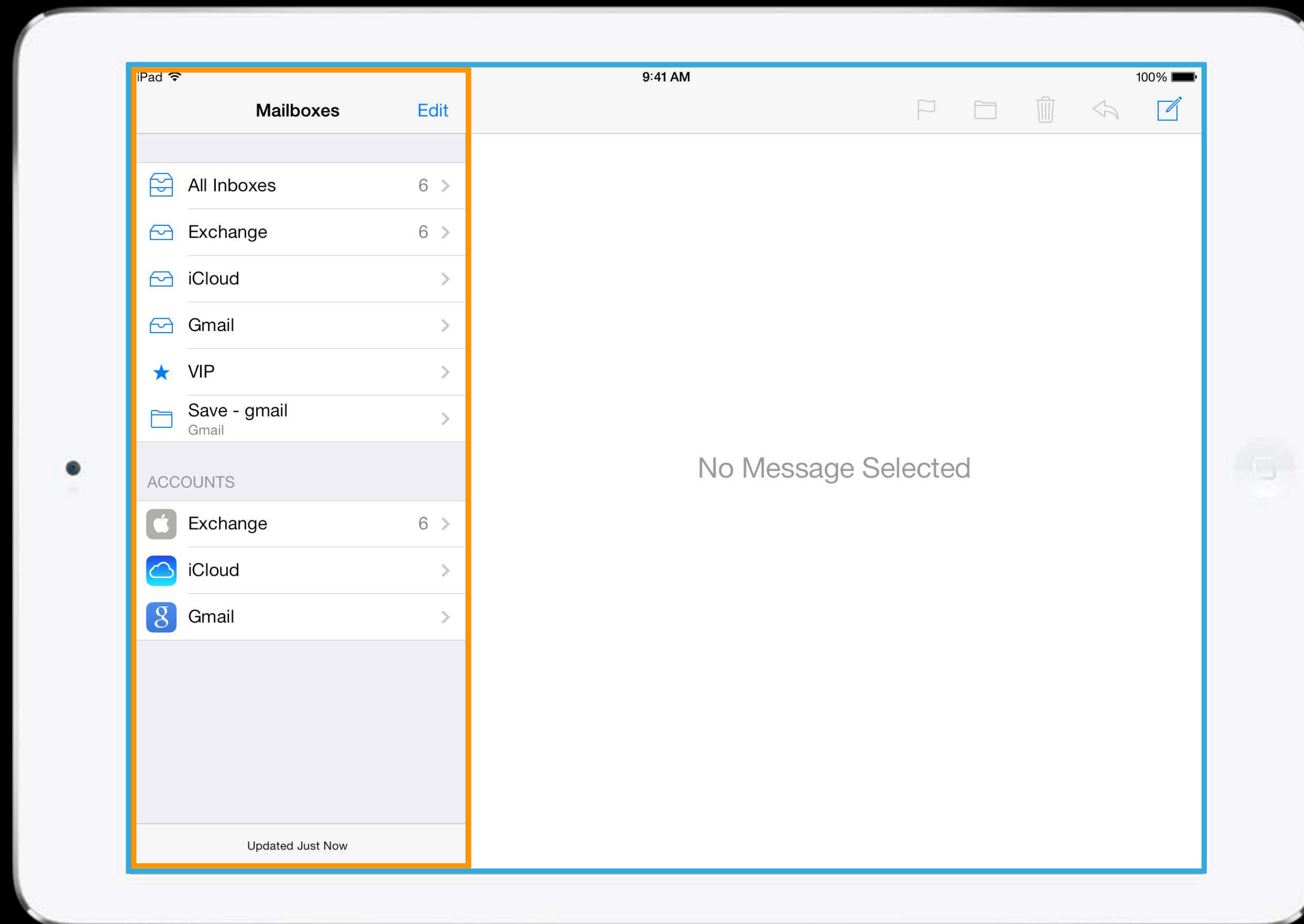


Split View
Controller

Adaptive View Controller Hierarchies

Simplifying and unifying code

Navigation
Controller

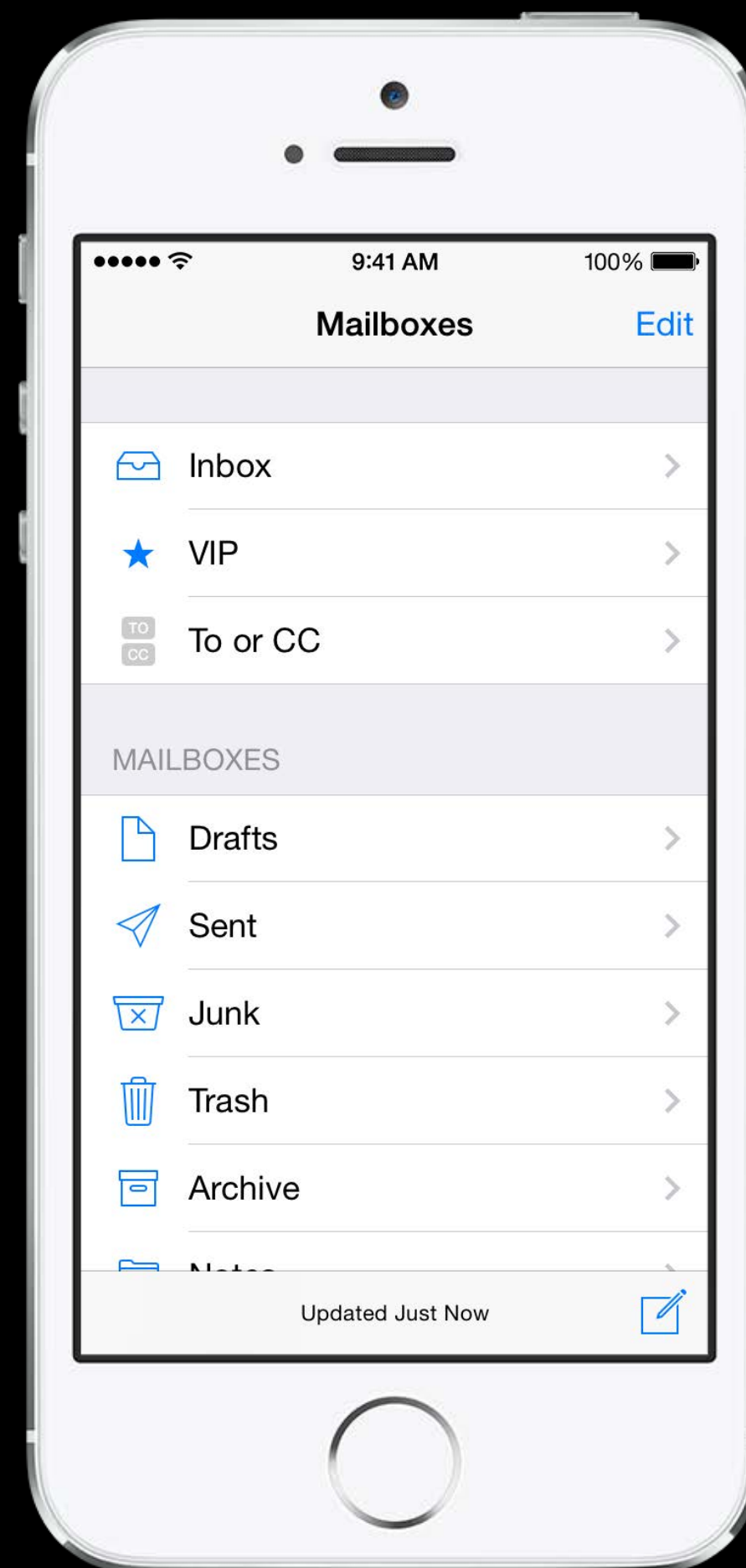


Split View
Controller



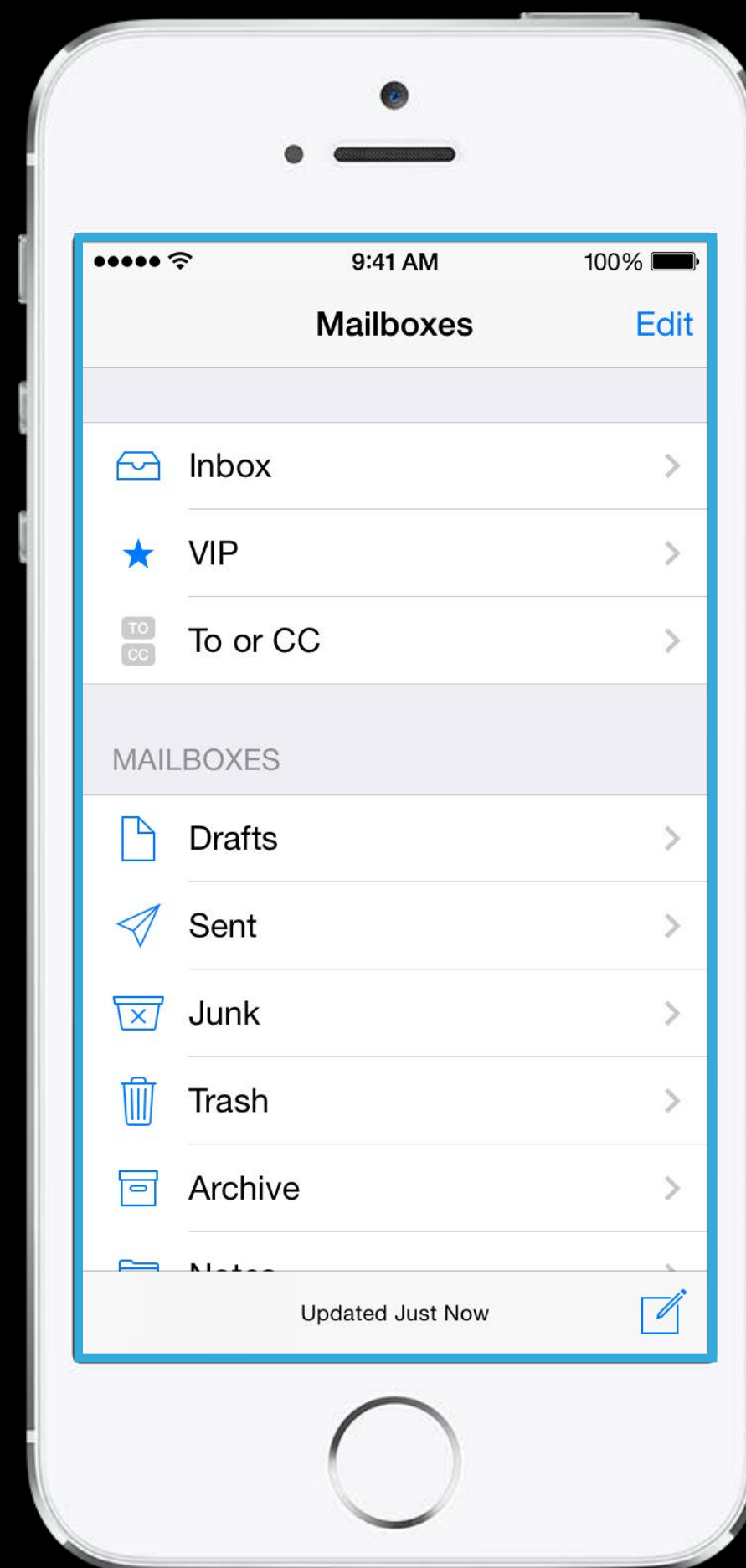
Adaptive View Controller Hierarchies

Simplifying and unifying code



Adaptive View Controller Hierarchies

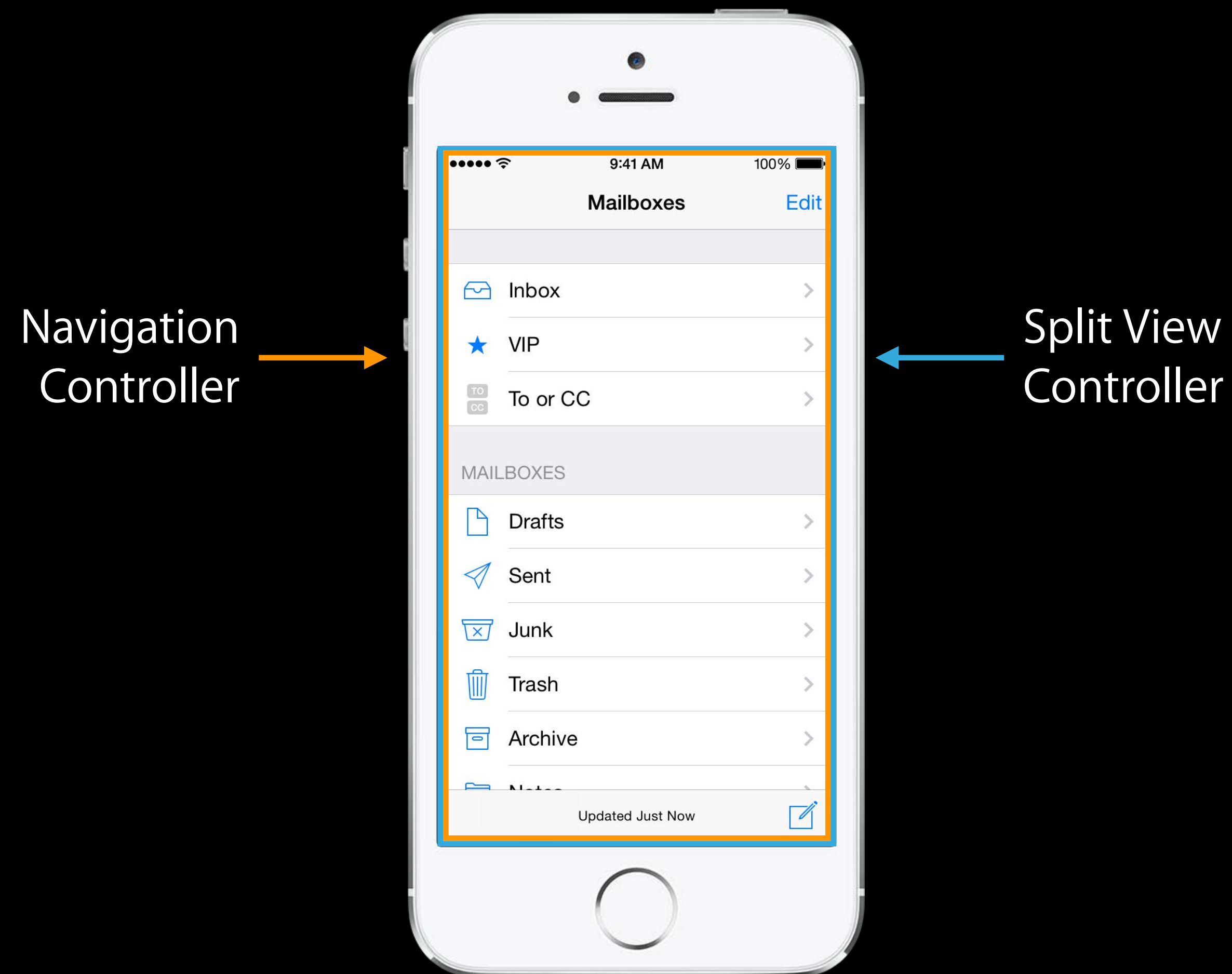
Simplifying and unifying code



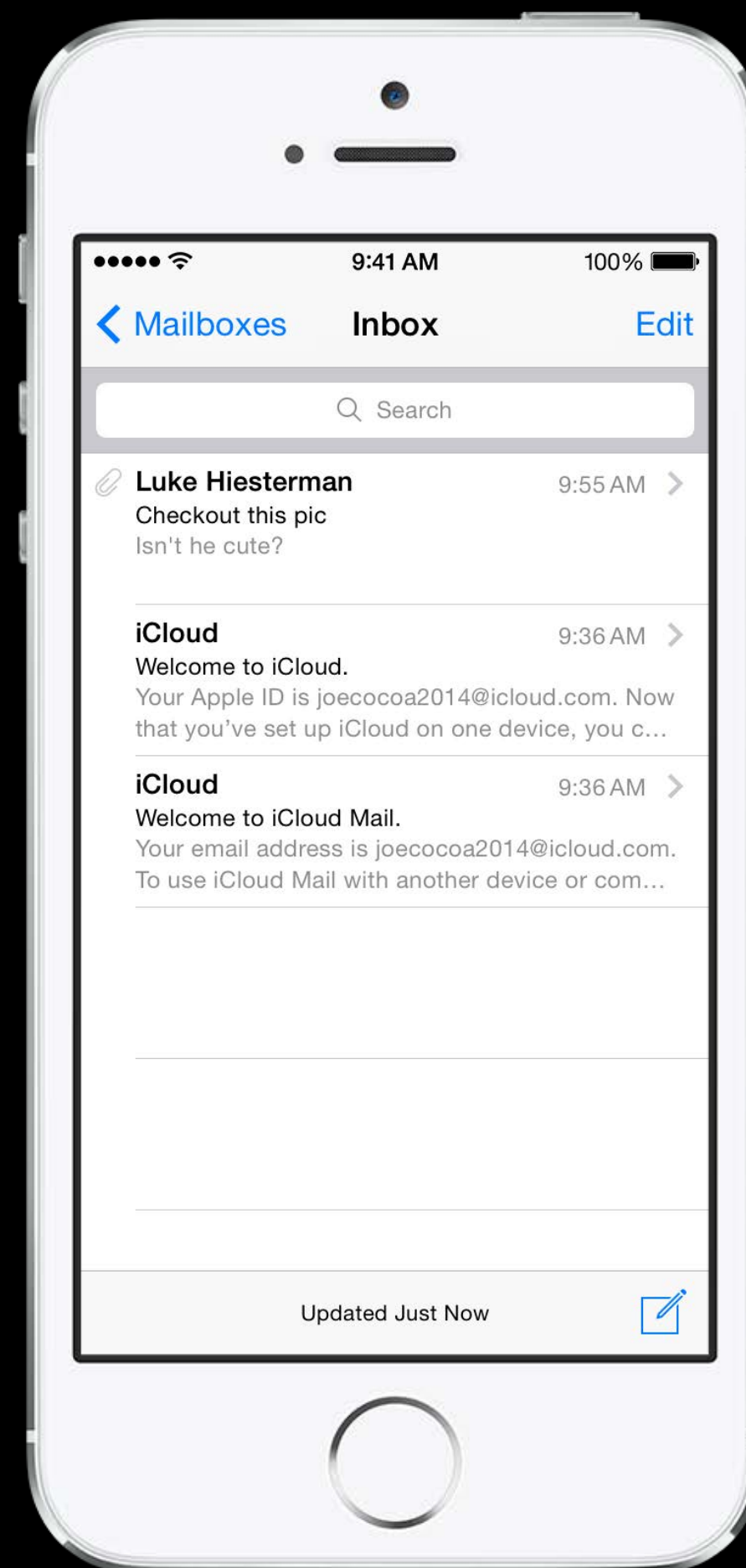
Split View
Controller

Adaptive View Controller Hierarchies

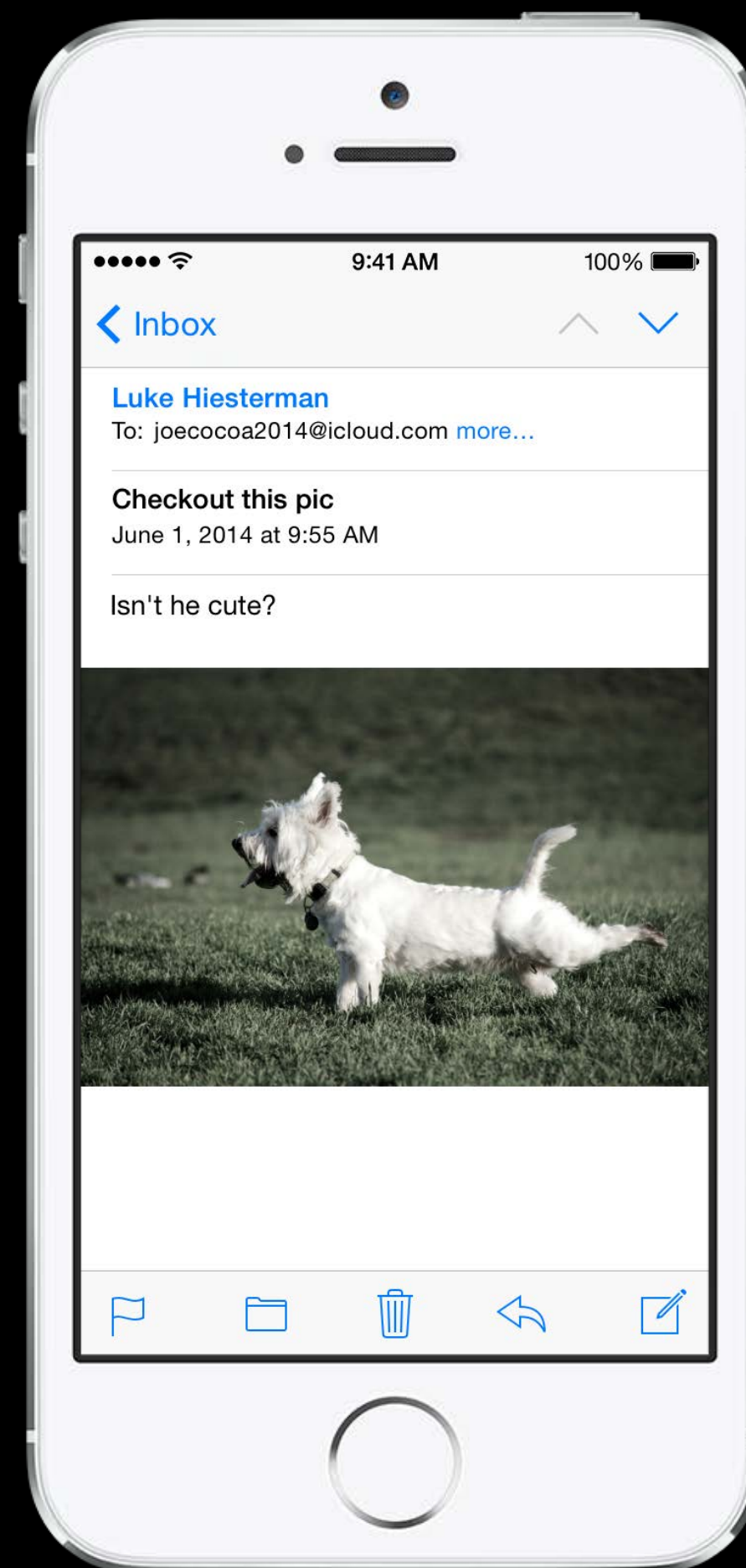
Simplifying and unifying code



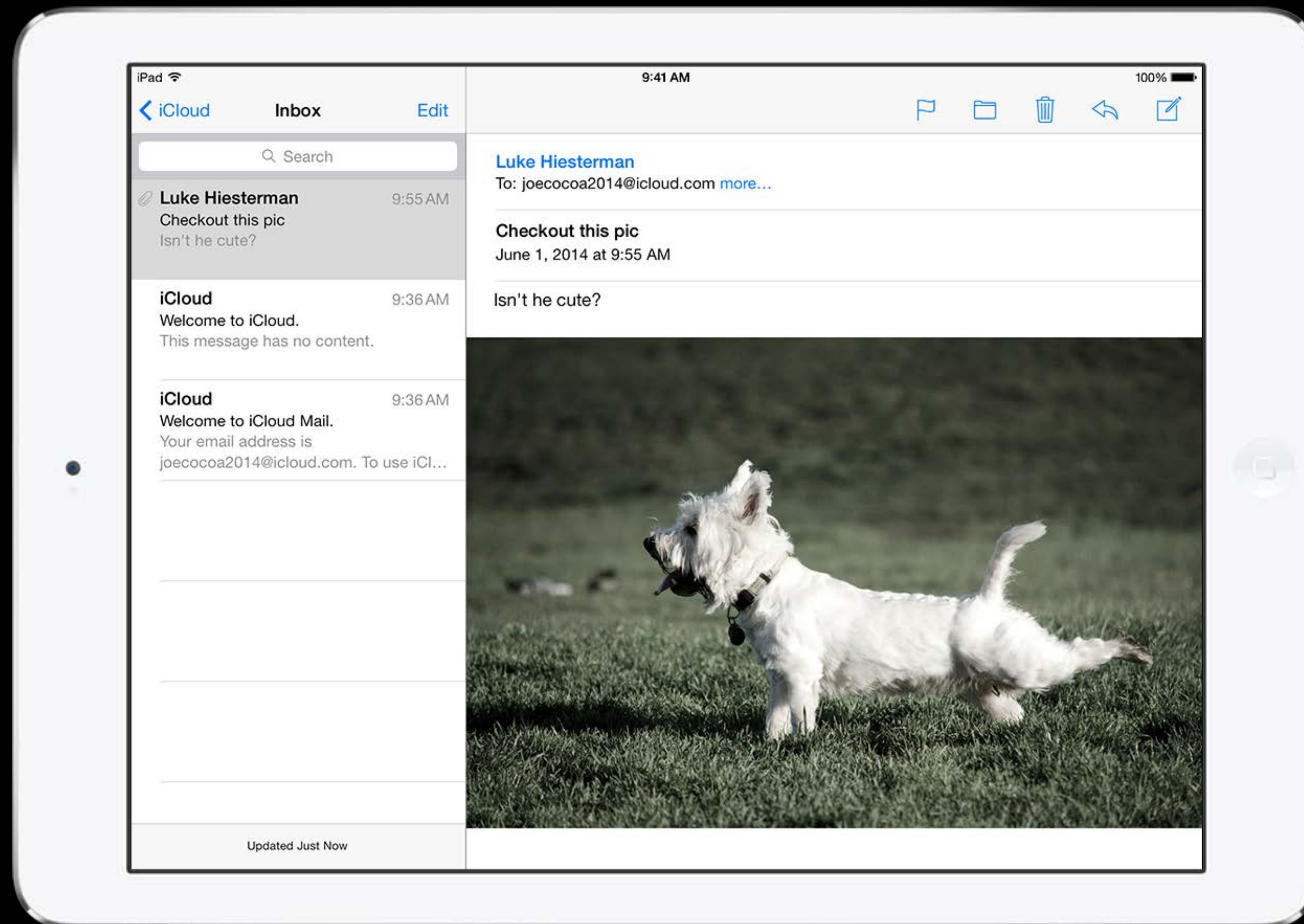
Adaptive View Controller Hierarchies



Adaptive View Controller Hierarchies



Adaptive View Controller Hierarchies



UISplitViewController

iOS 8 workhorse

UISplitViewController

iOS 8 workhorse

Now available on all devices

UISplitViewController

iOS 8 workhorse

Now available on all devices

Handles primary-secondary pattern transparently

UISplitViewController

iOS 8 workhorse

Now available on all devices

Handles primary-secondary pattern transparently

Enhanced customizability

More View Controller Information

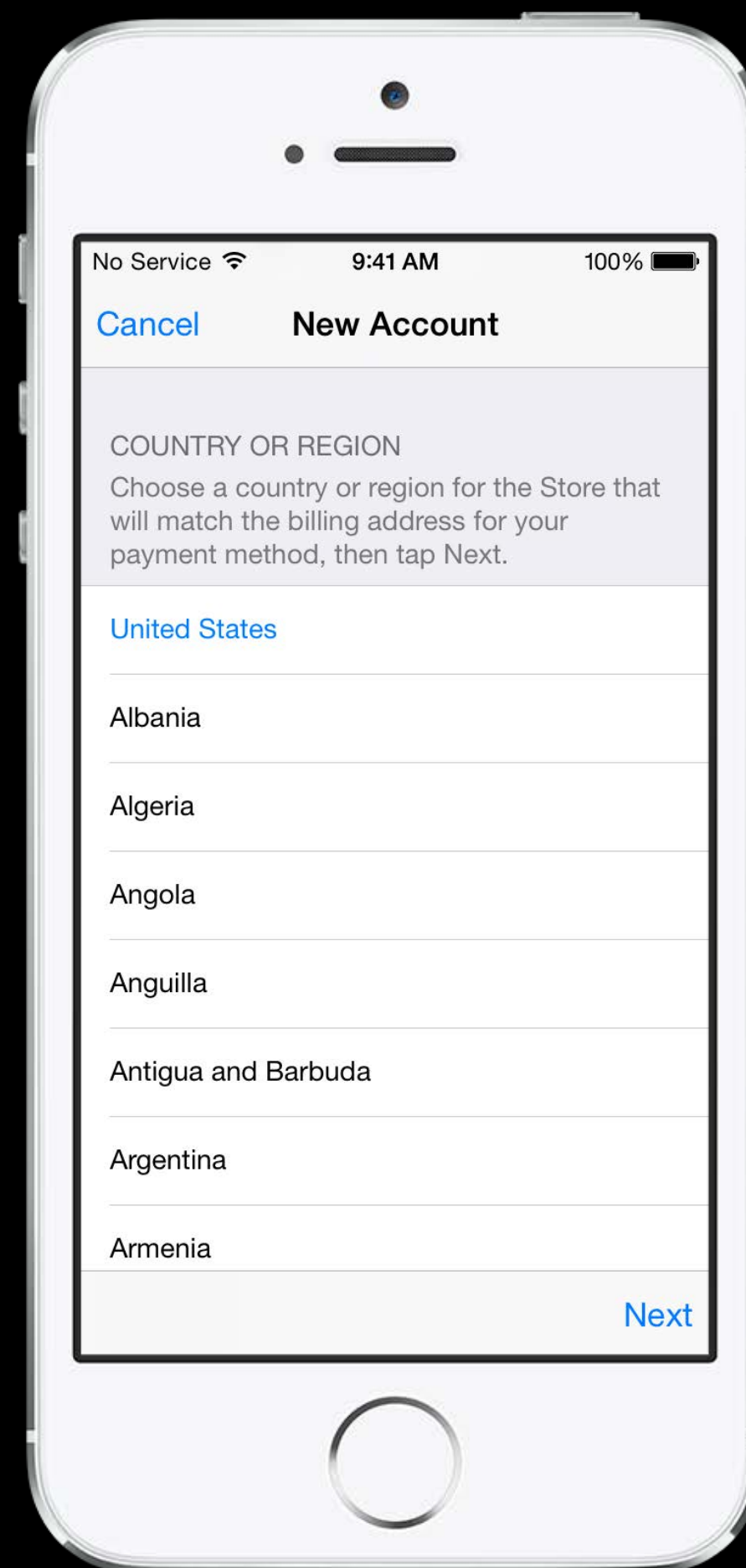
-
- View Controller Advancements in iOS 8 Mission Wednesday 9:00AM
-

Adaptive Presentations

Popovers, search results, and alerts

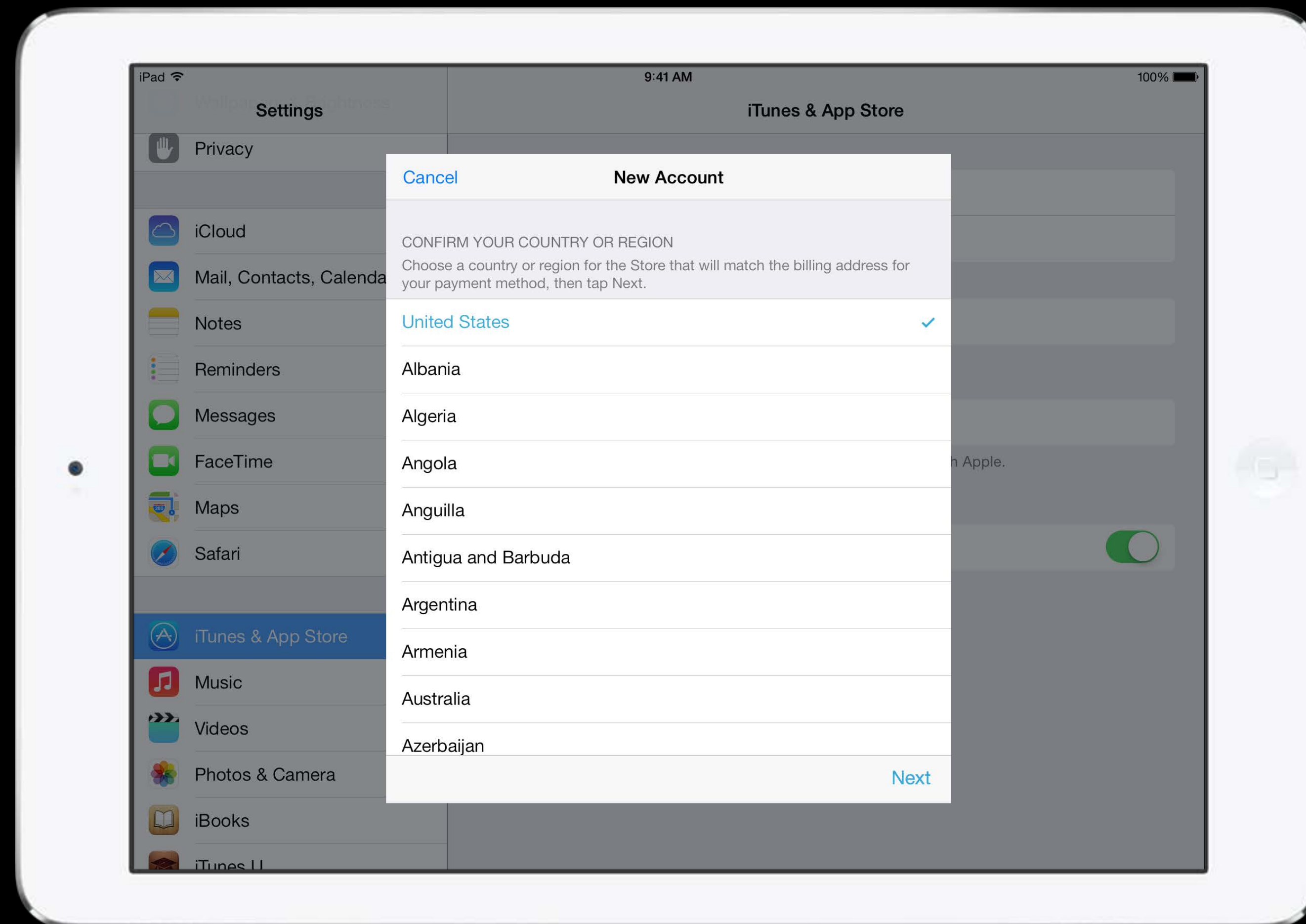
View Controller Presentations

Present how you want, where you want, when you want



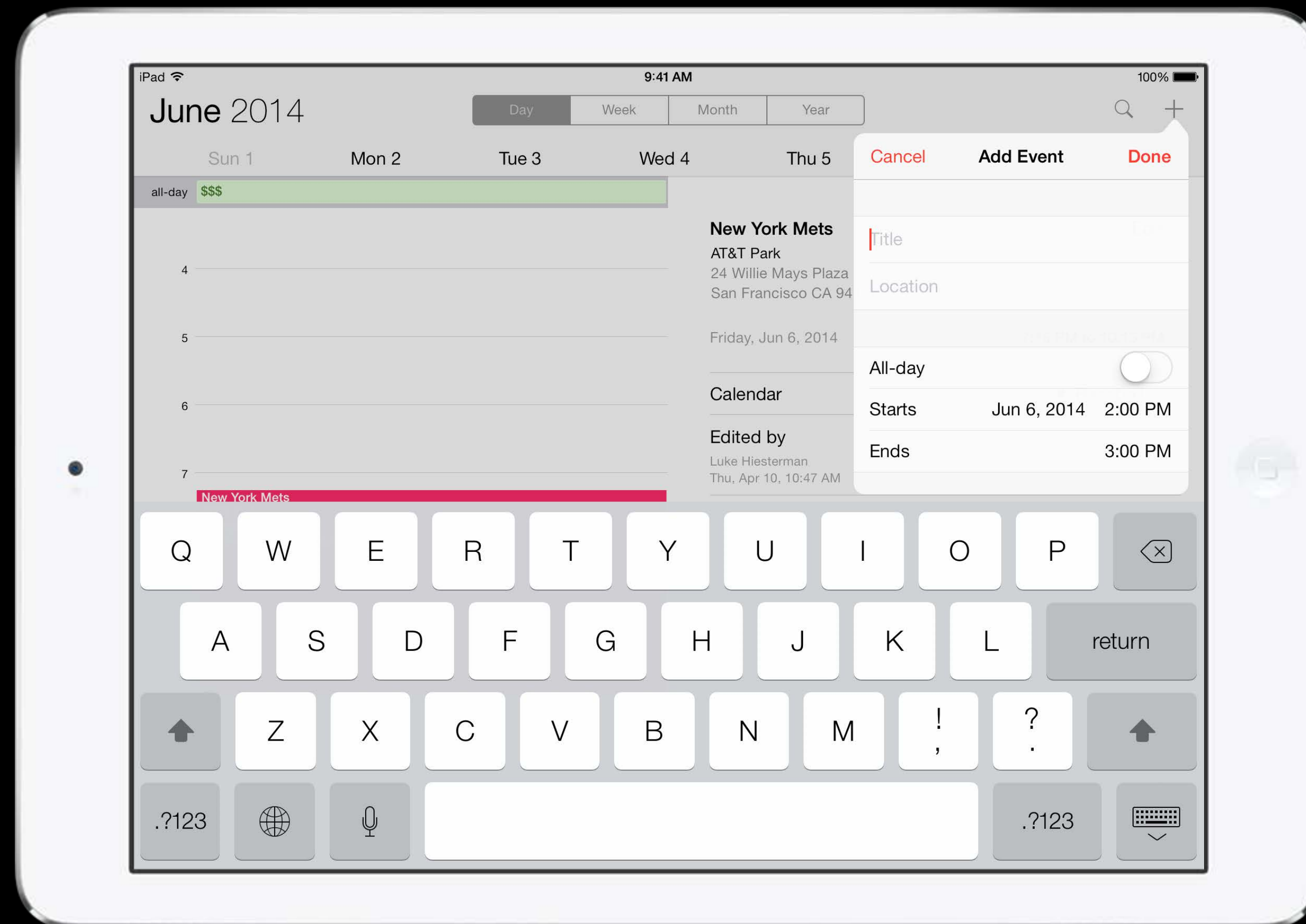
View Controller Presentations

Present how you want, where you want, when you want



View Controller Presentations

Present how you want, where you want, when you want



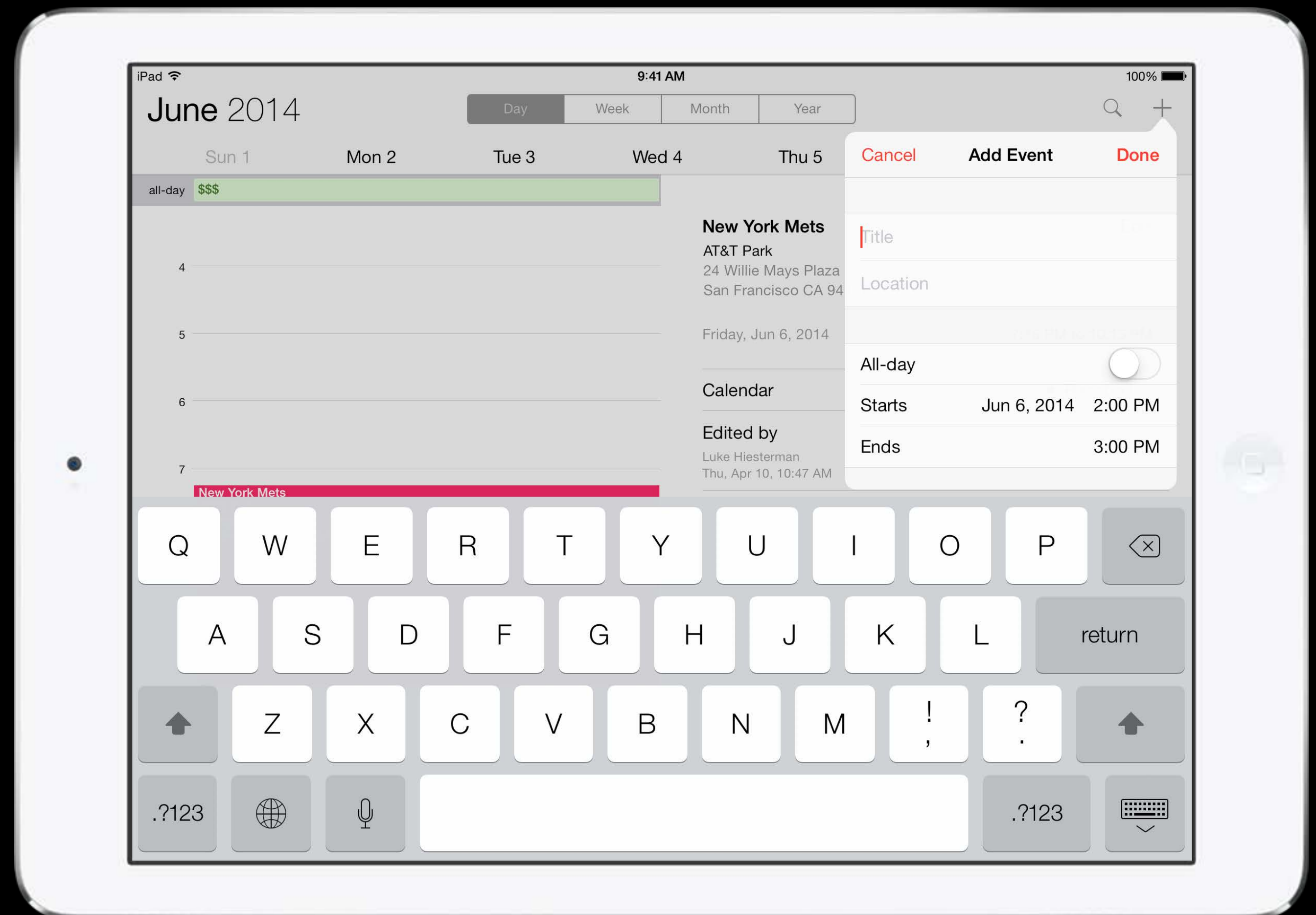
Adaptive Presentations

Popovers

Adaptive Presentations

Popovers

Natural presentation
in regular width

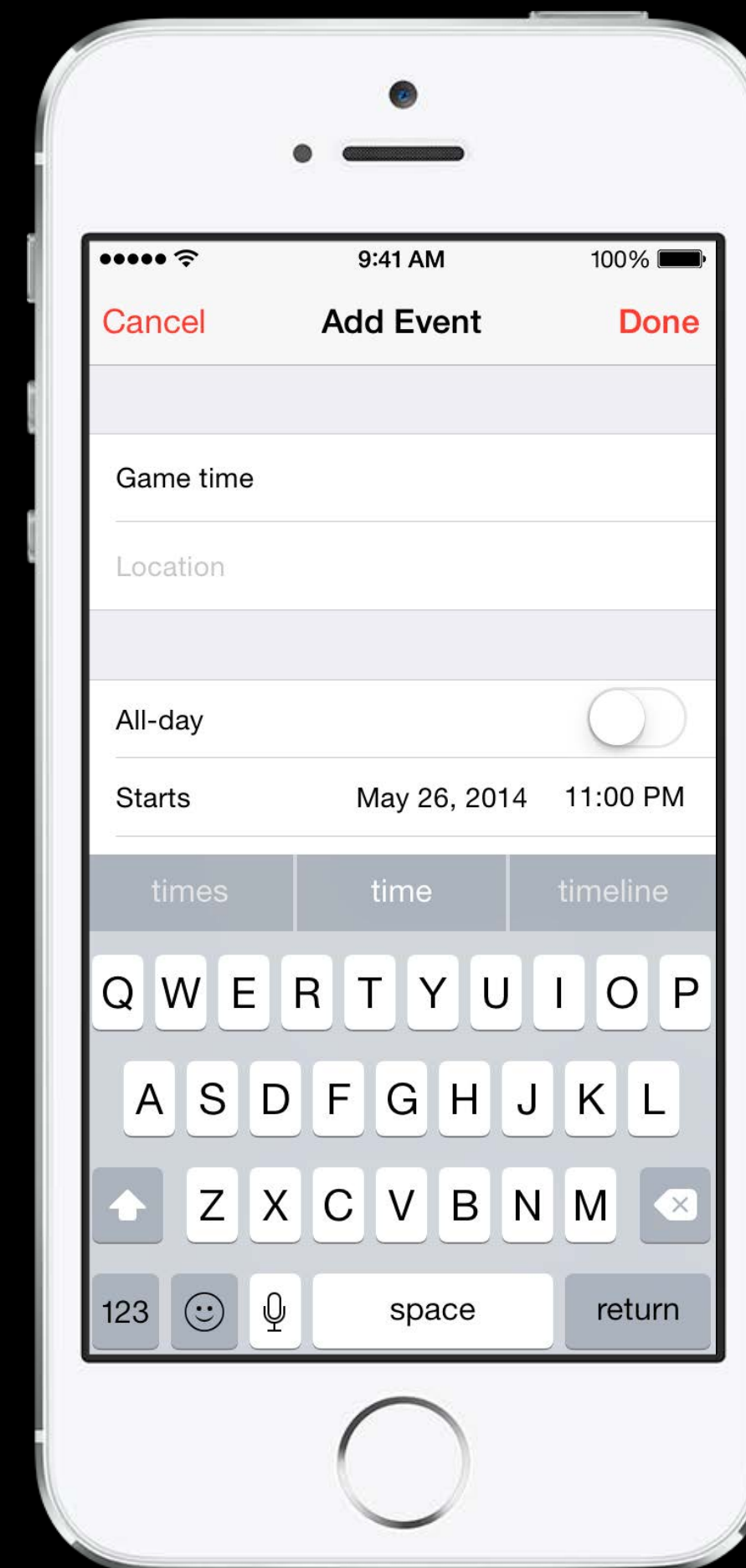


Adaptive Presentations

Popovers

Natural presentation
in regular width

FullScreen or
OverFullScreen in
compact width



Adaptive Search Results

Adaptive Search Results

UISearchDisplayController replaced by
UISearchController

Adaptive Search Results

UISearchDisplayController replaced by

UISearchController

UIViewController subclass

Adaptive Search Results

UISearchDisplayController replaced by
UISearchController

UIViewController subclass

All the power of adaptive presentations
and hierarchies is included

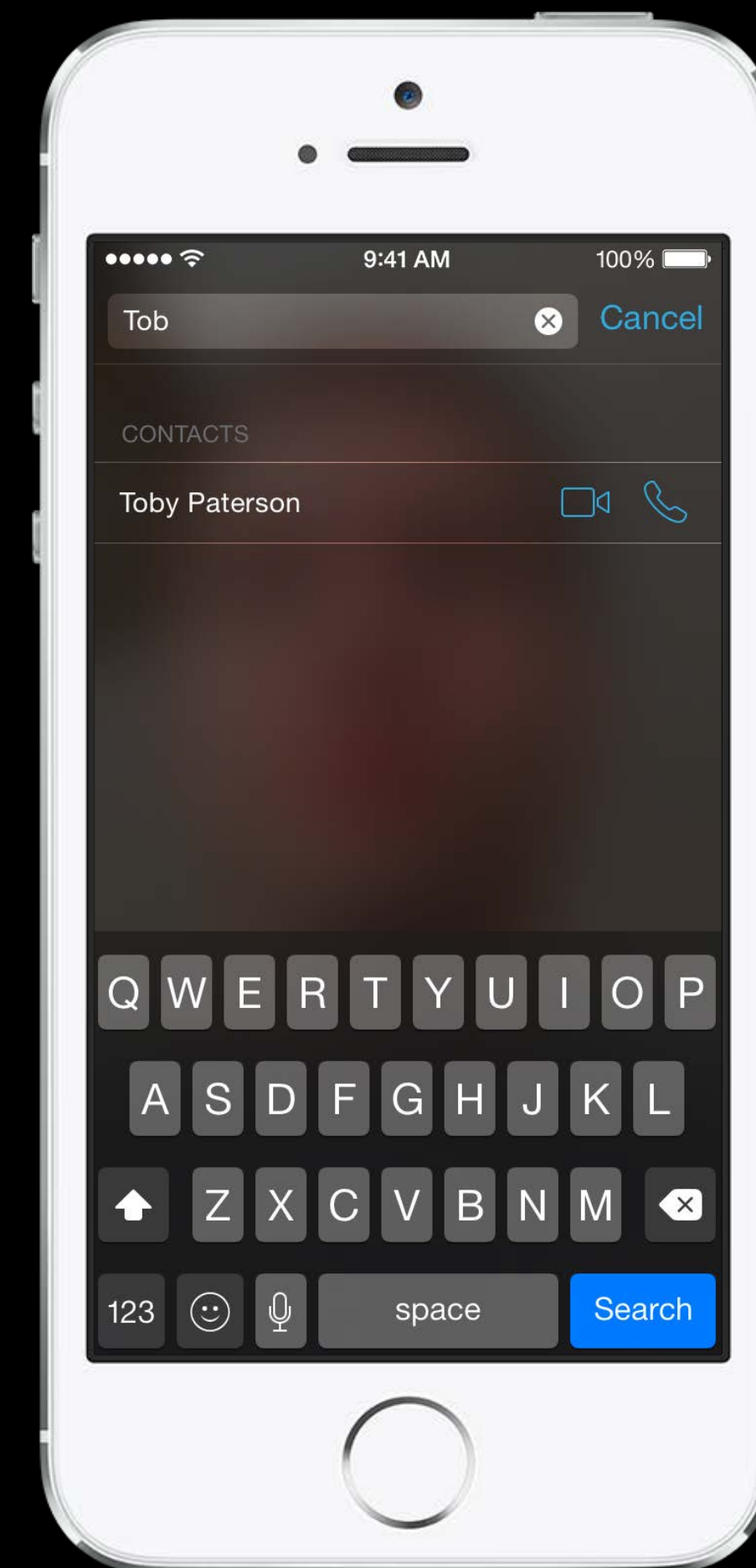
Adaptive Search Results

UISearchDisplayController replaced by
UISearchController

UIViewController subclass

All the power of adaptive presentations
and hierarchies is included

Customizable UI



Adaptive Alerts

Adaptive Alerts

UIAlertController replaces UIAlertView and UIActionSheet

Adaptive Alerts

UIAlertController replaces UIAlertView and UIActionSheet

UIAlertController is a view controller subclass

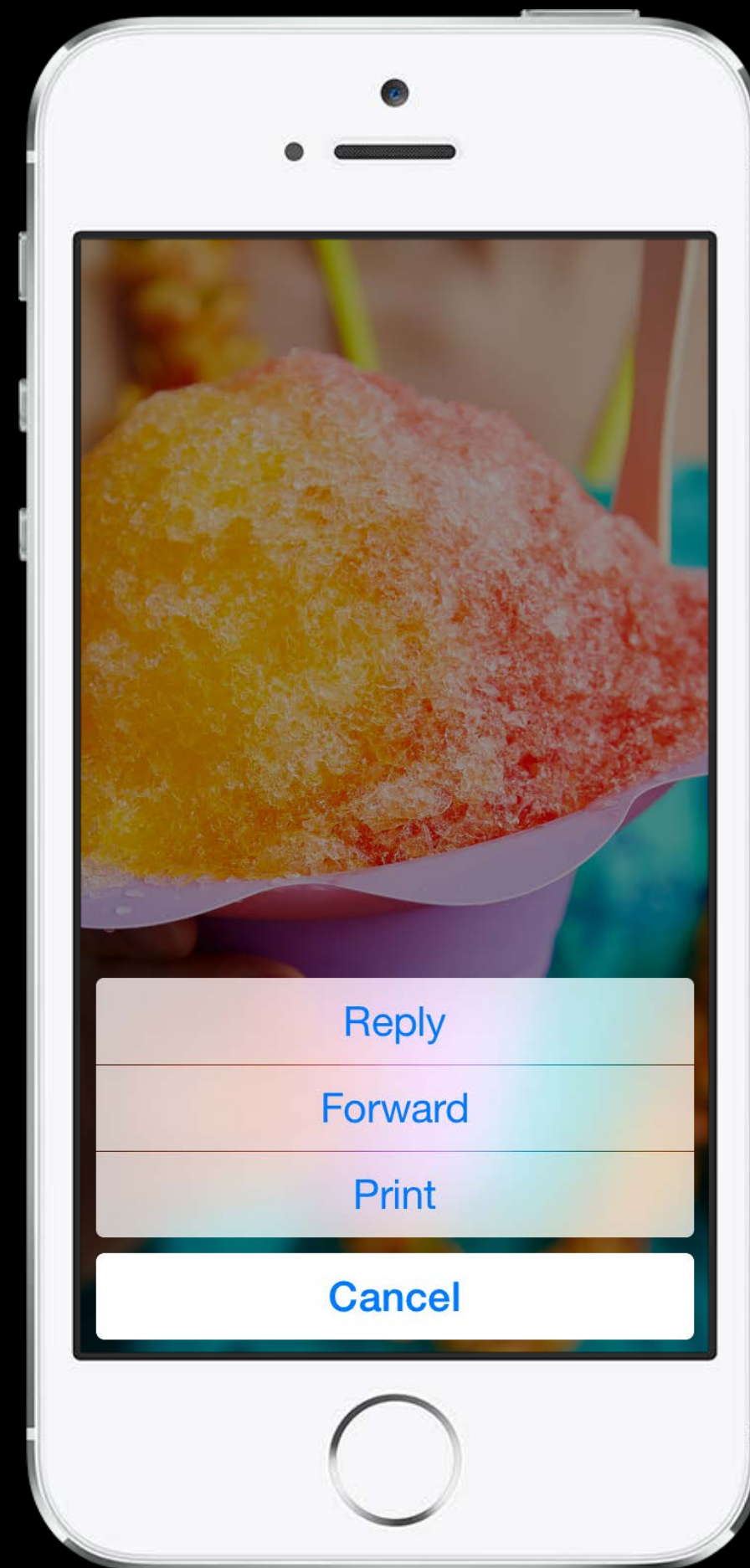
Adaptive Alerts

UIAlertController replaces UIAlertView and UIActionSheet

UIAlertController is a view controller subclass

Adapts styling to context

Adaptive Alerts



Adaptive Alerts



More Adaptive Presentations

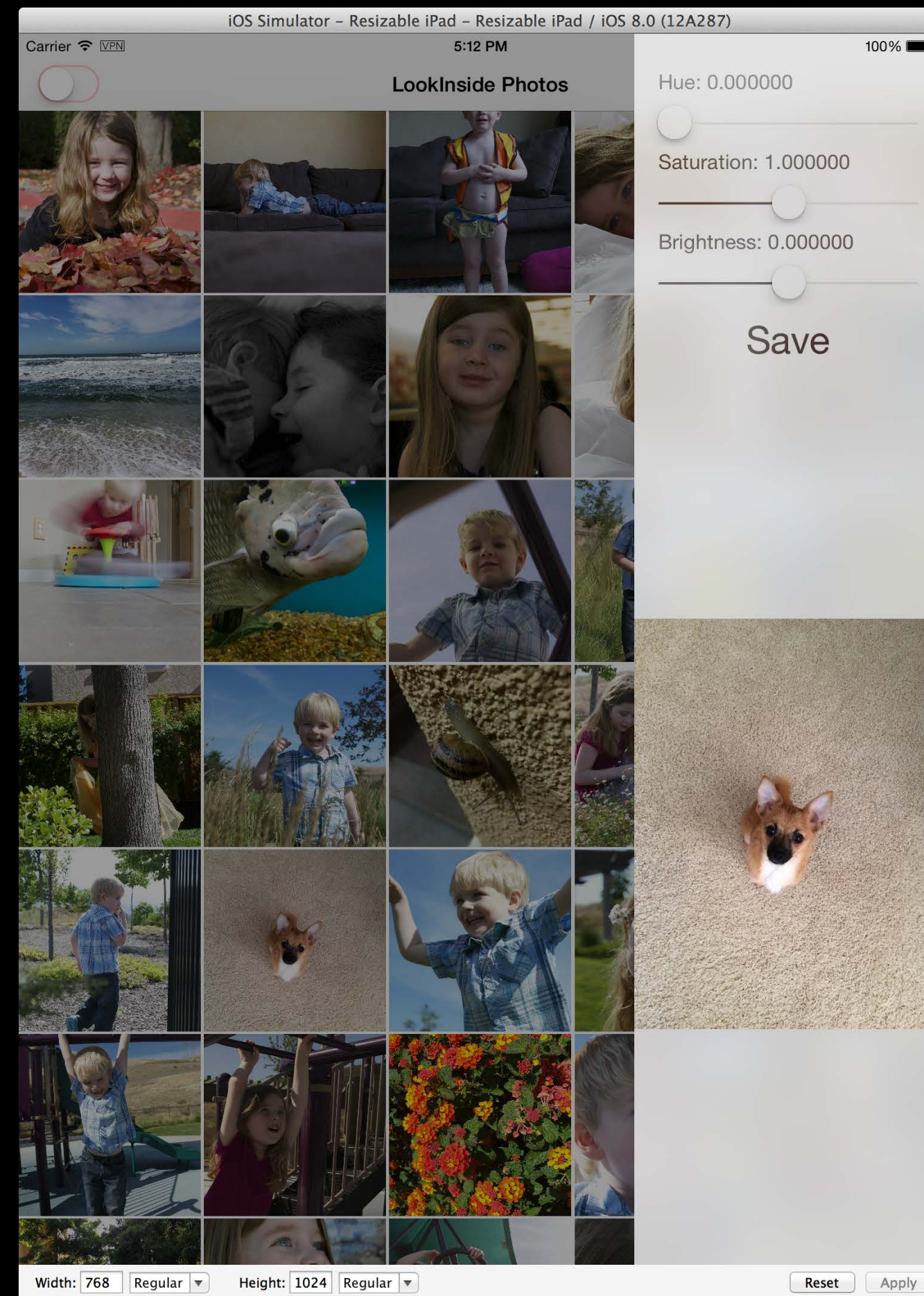
-
- A Look Inside Presentation Controllers

Mission

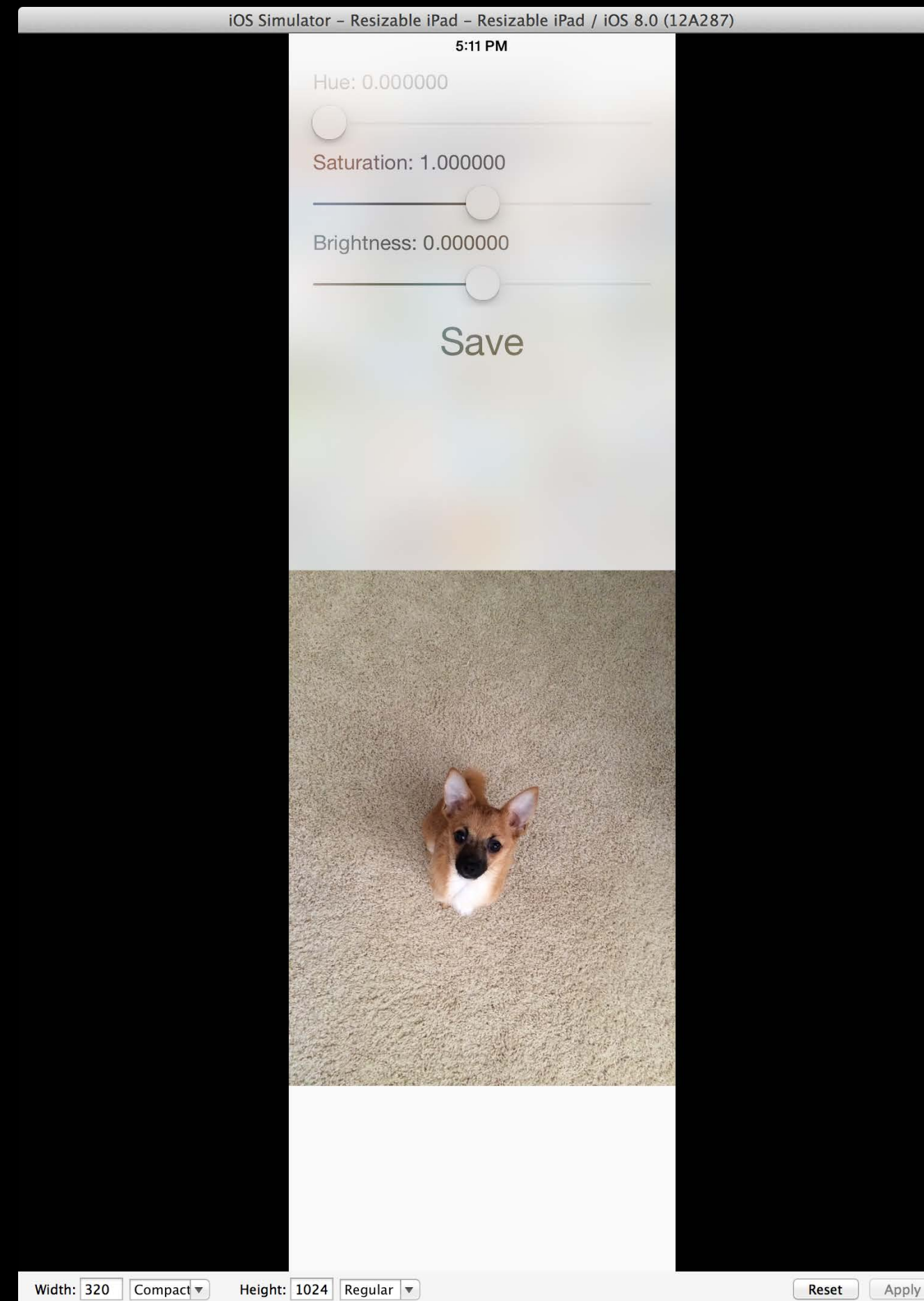
Thursday 11:30AM

Testing with the iOS Simulator

Testing with the iOS Simulator



Testing with the iOS Simulator



Customizing User Interface

Visual Effects

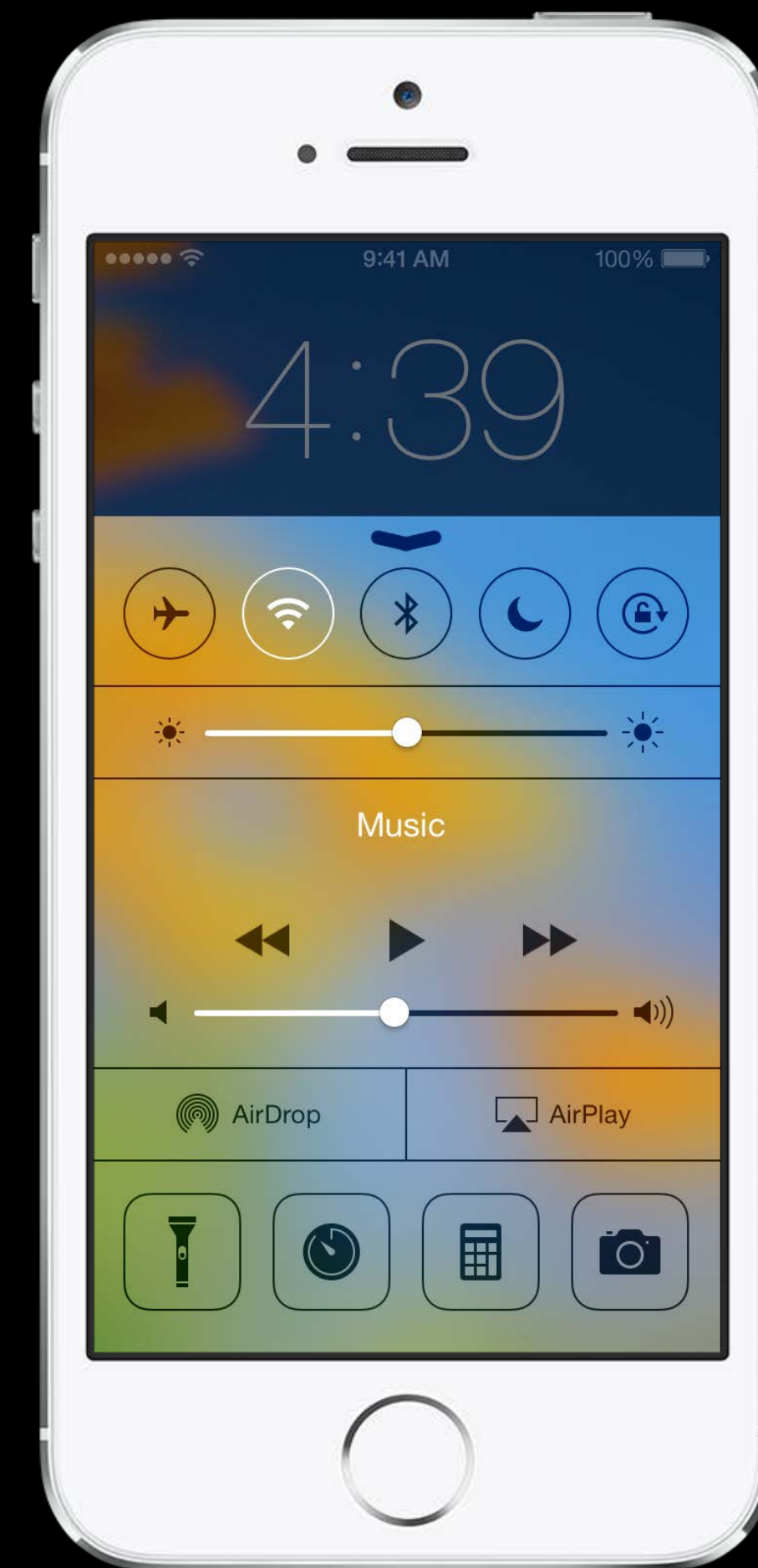
Visual Effects

UIVisualEffectView

Visual Effects

UIVisualEffectView

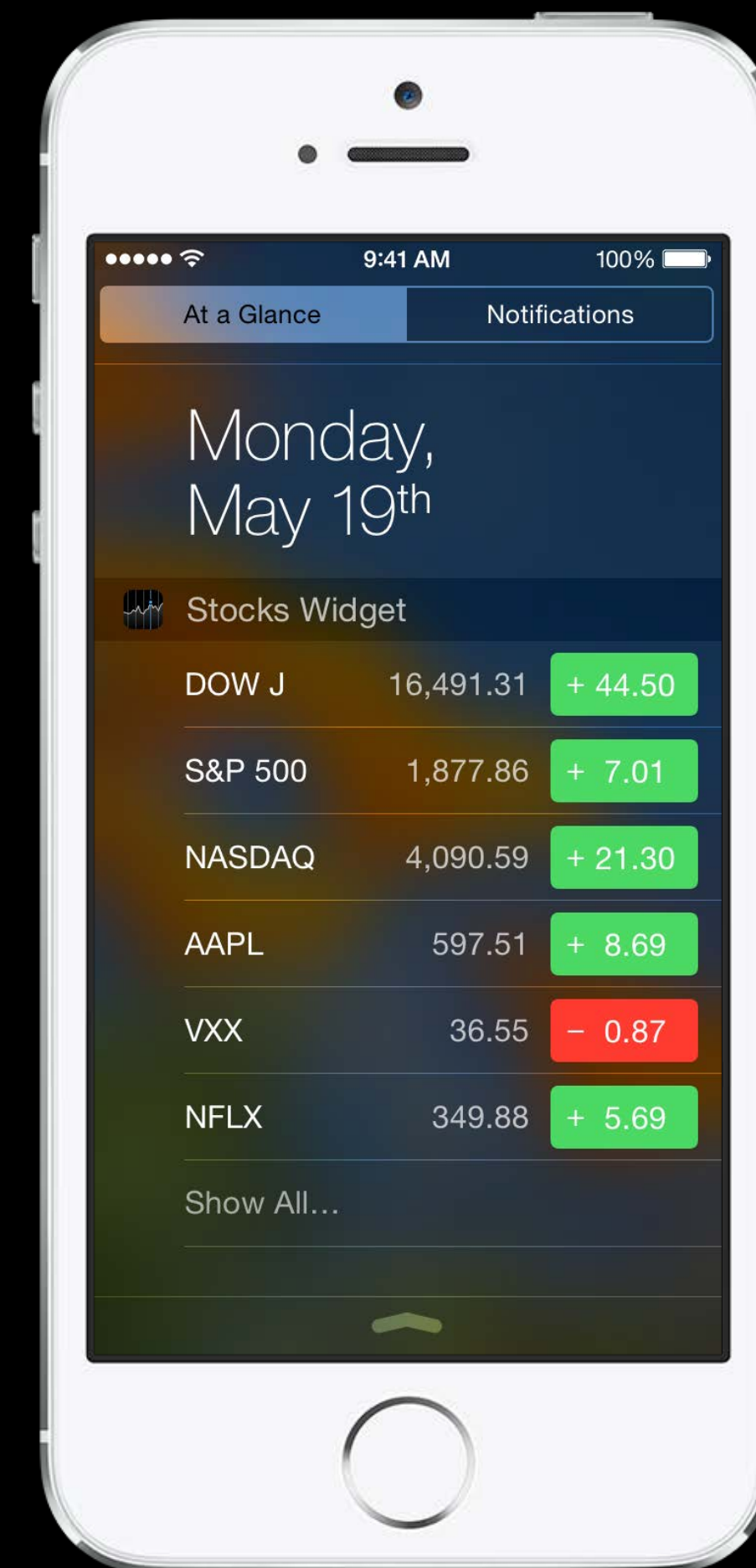
- UIBlurEffect



Visual Effects

UIVisualEffectView

- UIBlurEffect
- UIVibrancyEffect



Visual Effects

UIVisualEffectView

- UIBlurEffect
- UIVibrancyEffect

Vibrant
Separators

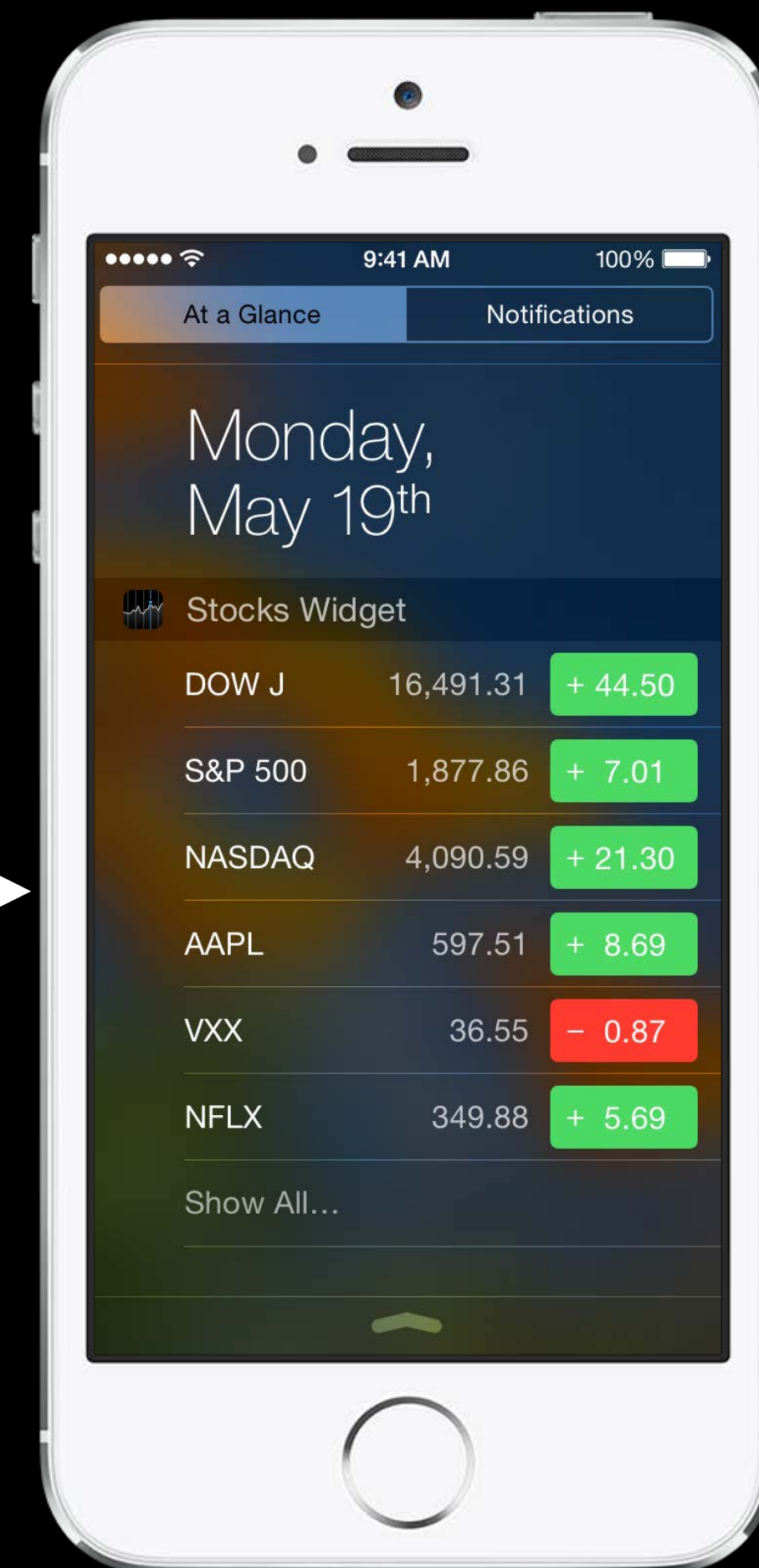


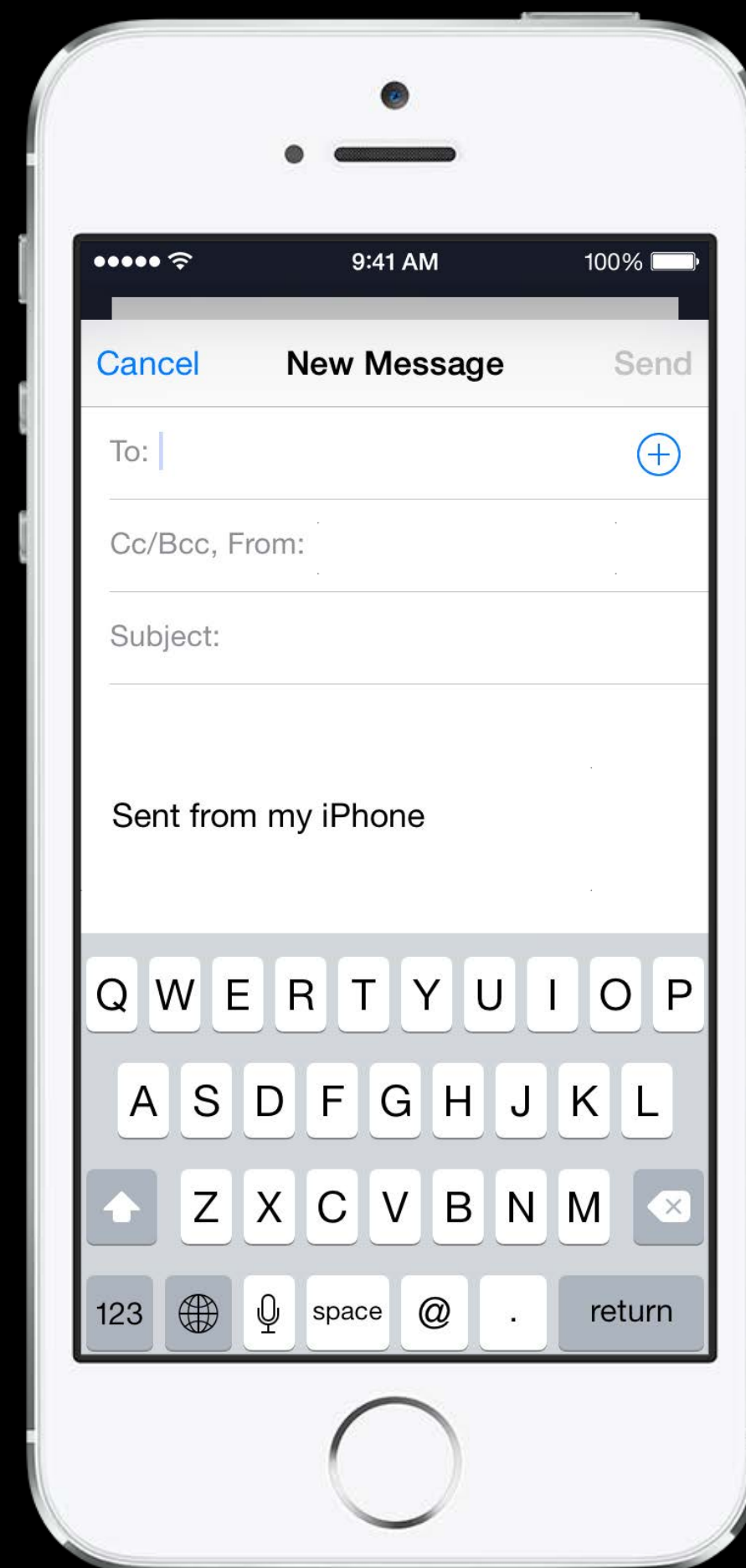
Image Assets

```
override func viewDidLoad(animated: Bool) {  
    self.imageView.image = UIImage(named: "My-Dog",  
                                     inBundle: nil,  
                                     compatibleWithTraitCollection: self.traitCollection)  
}
```

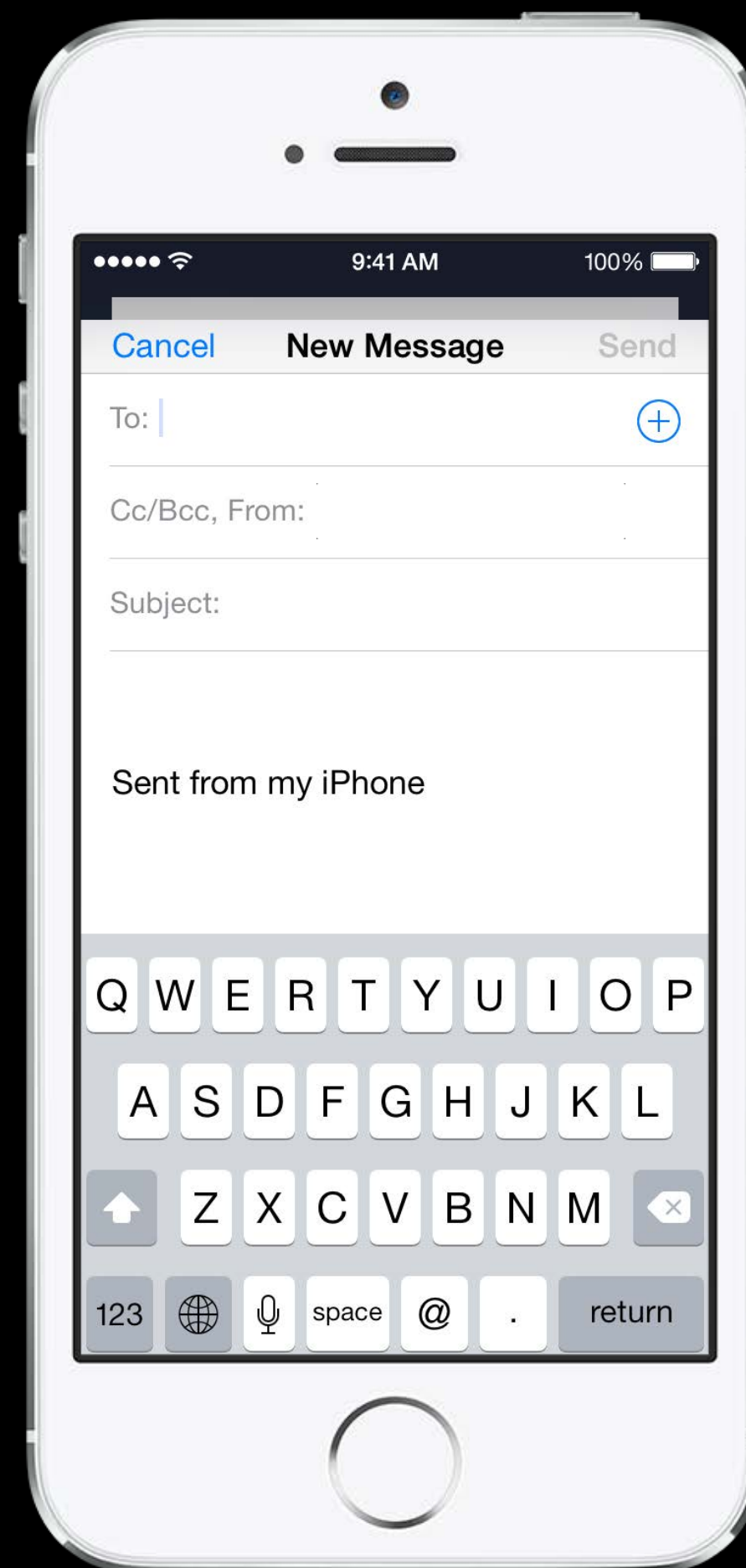
Image Assets

```
override func viewDidLoad(animated: Bool) {  
    self.imageView.image = UIImage(named: "My-Dog",  
                                     inBundle: nil,  
                                     compatibleWithTraitCollection: self.traitCollection)  
}
```

Condensing Bars



Condensing Bars



Condensing Bars



Condensing Bars



More Interface Customizations

-
- Creating Custom iOS User Interfaces

Marina

Wednesday 3:15PM

Self-sizing Table Cells

Self-sizing Table Cells

Variable row heights without `tableView:heightForRowAtIndexPath:`

Self-sizing Table Cells

Variable row heights without `tableView:heightForRowAtIndexPath:`

Table cells can encapsulate sizing logic

Self-sizing Table Cells

Variable row heights without `tableView:heightForRowAtIndexPath:`

Table cells can encapsulate sizing logic

Supports Auto Layout

Self-sizing Table Cells

```
- (instancetype)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString*)identifier
{
    if ((self = [super initWithStyle:style reuseIdentifier:reuseIdentifier])) {

    }

    return self;
}
```

Self-sizing Table Cells

```
- (instancetype)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString*)identifier
{
    if ((self = [super initWithStyle:style reuseIdentifier:reuseIdentifier])) {
        [self.contentView
         addConstraints:[NSLayoutConstraint constraintsWithVisualFormat:@"V: |-myTextView-|"
                              options:0
                              metrics:nil
                              views:NSDictionaryOfVariableBindings(myTextView)]];
    }

    return self;
}
```


Self-sizing Table Cells

```
- (instancetype)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString*)identifier
{
    if ((self = [super initWithStyle:style reuseIdentifier:reuseIdentifier])) {
        [self.contentView
            addConstraints:[NSLayoutConstraint constraintsWithVisualFormat:@"V: |-myTextView-|"
                options:0
                metrics:nil
                views:NSDictionaryOfVariableBindings(myTextView)]];

        [self.contentView
            addConstraints:[NSLayoutConstraint constraintsWithVisualFormat:@"H: |-myTextView-|"
                options:0
                metrics:nil
                views:NSDictionaryOfVariableBindings(myTextView)]];
    }

    return self;
}
```

Self-sizing Table Cells

```
- (instancetype)initWithStyle:(UITableViewCellStyle)style reuseIdentifier:(NSString*)identifier
{
    if ((self = [super initWithStyle:style reuseIdentifier:reuseIdentifier])) {
        [self.contentView
            addConstraints:[NSLayoutConstraint constraintsWithVisualFormat:@"V: |-myTextView-|"
                options:0
                metrics:nil
                views:NSDictionaryOfVariableBindings(myTextView)]];

        [self.contentView
            addConstraints:[NSLayoutConstraint constraintsWithVisualFormat:@"H: |-myTextView-|"
                options:0
                metrics:nil
                views:NSDictionaryOfVariableBindings(myTextView)]];
    }

    return self;
}
```

More Table Views

-
- What's New in Table and Collection Views

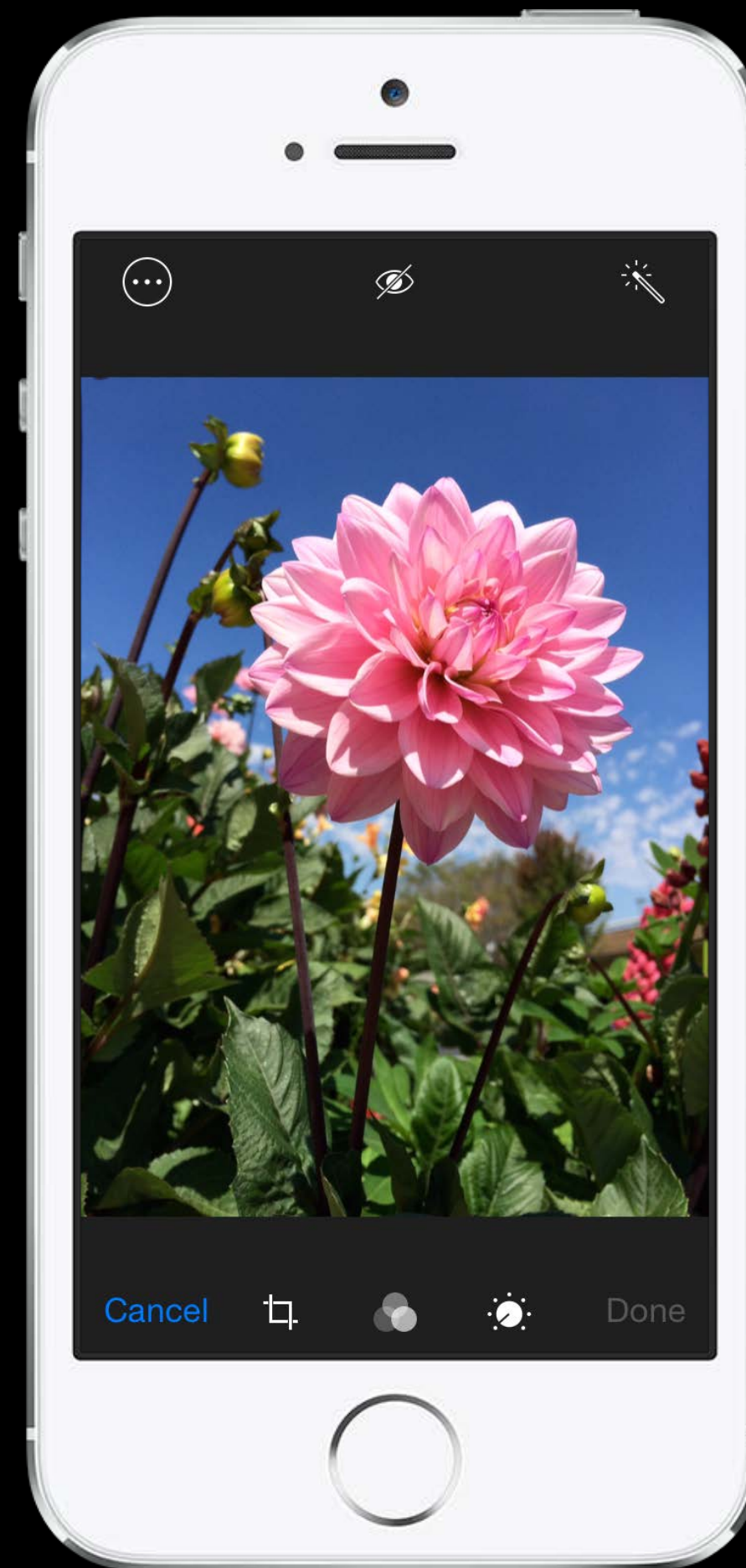
Presidio

Thursday 10:15AM

App Extensions

App Extensions

Extending the system



App Extensions

Extending the system



App Extensions

Extending the system



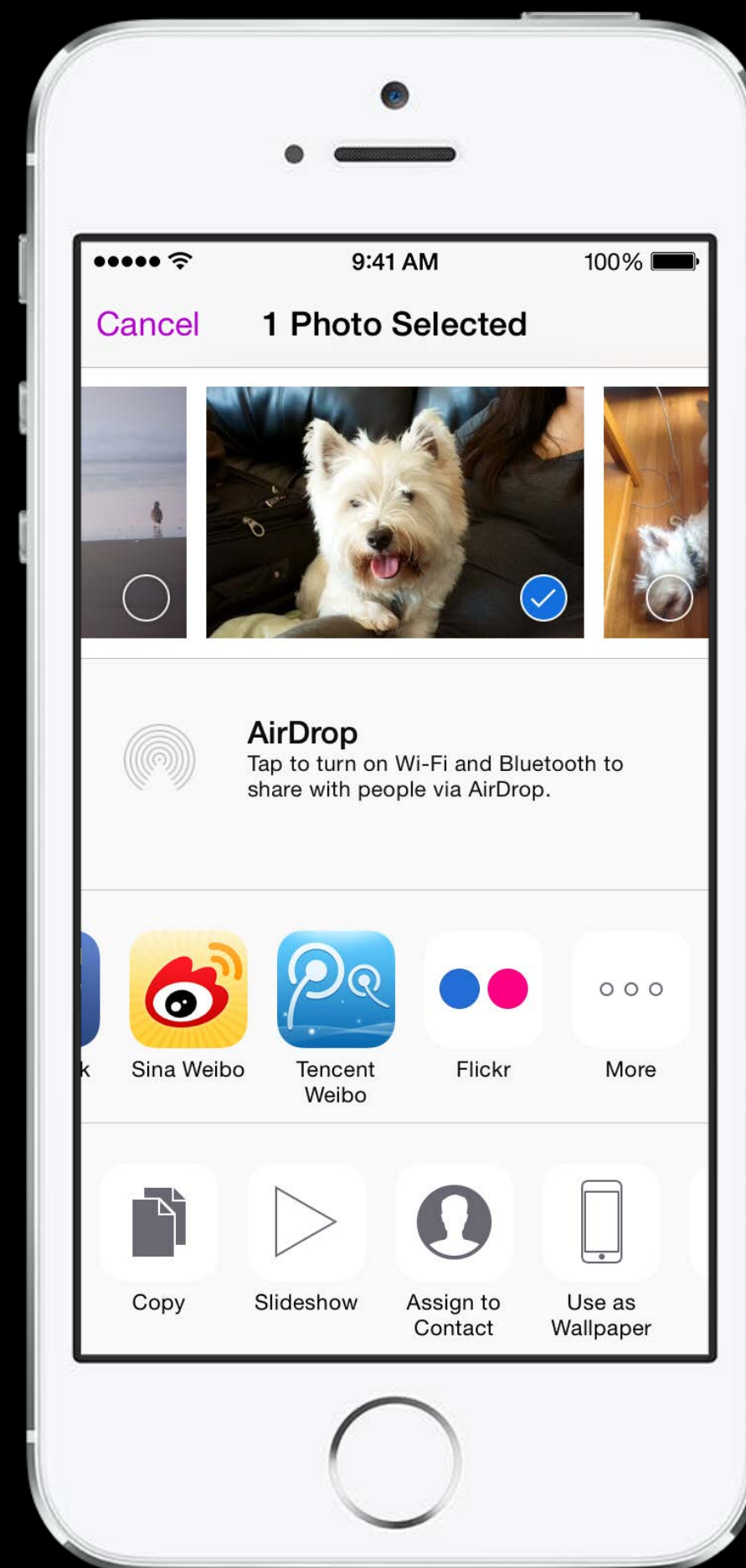
App Extensions

Extending the system



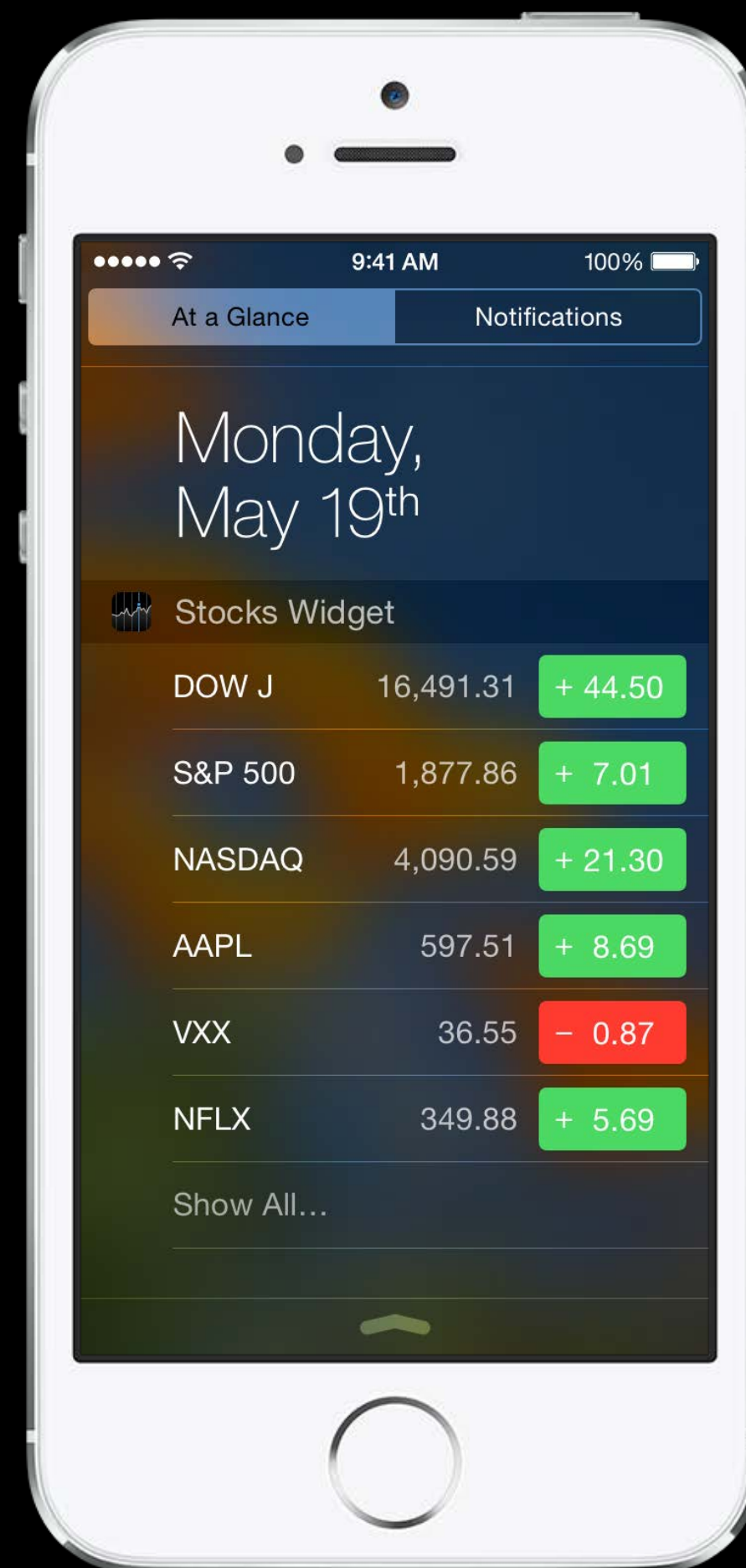
App Extensions

Extending the system



App Extensions

Extending the system



App Extensions

Extending the system

App Extensions

Extending the system

Photos

App Extensions

Extending the system

Photos

Sharing

App Extensions

Extending the system

Photos

Sharing

Widgets

App Extensions

Extending the system

Photos

Sharing

Widgets

Actions without UI

App Extensions

Extending the system

Photos

Sharing

Widgets

Actions without UI

Document providers

App Extensions

Extending the system

Photos

Sharing

Widgets

Actions without UI

Document providers

Custom keyboards

More App Extensions

-
- Creating Extensions for iOS and OS X, Part 1 Mission Tuesday 2:00PM
 - Creating Extensions for iOS and OS X, Part 2 Mission Wednesday 11:30AM
-

Notification Updates

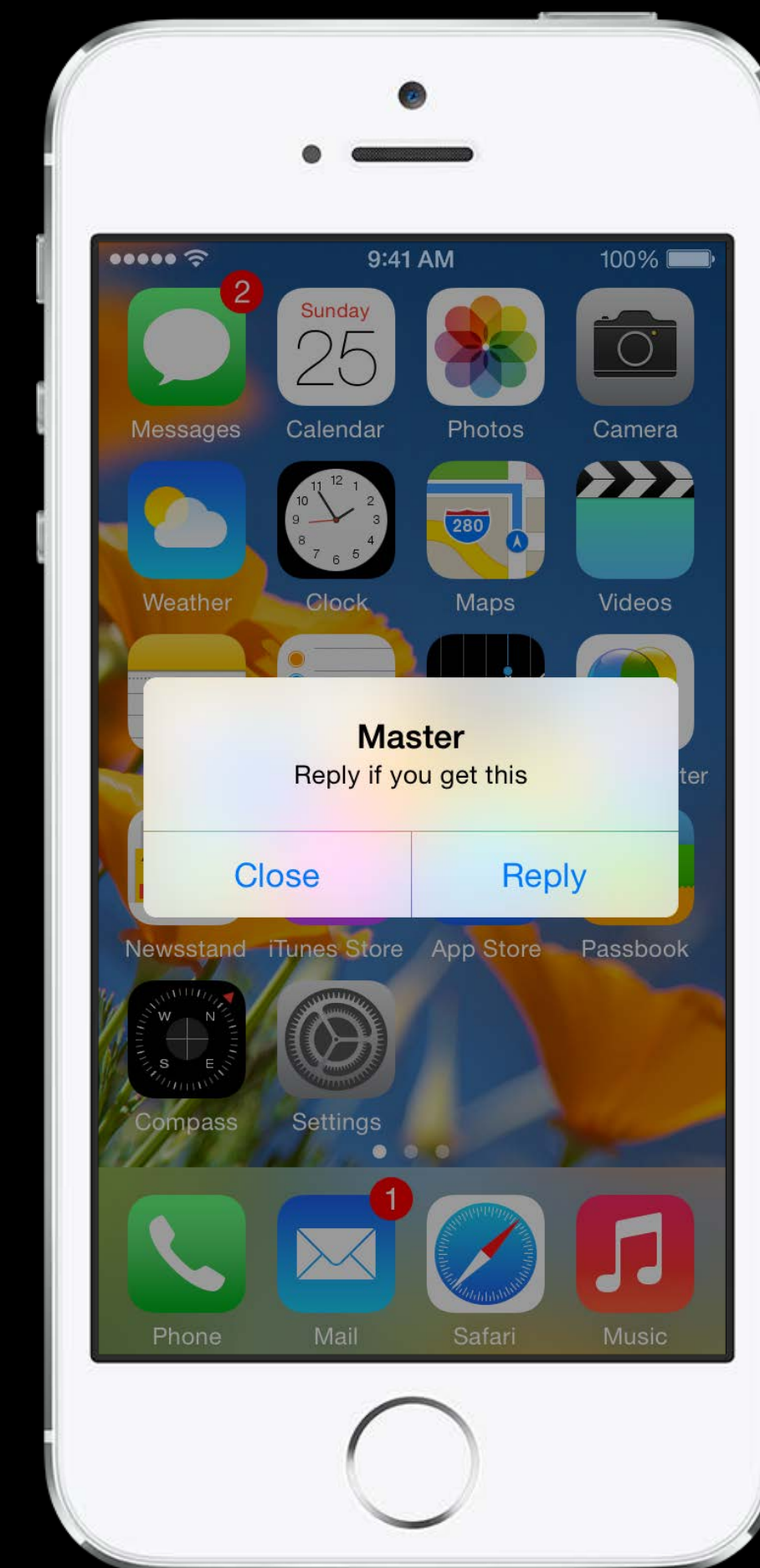
Notification Updates

Notification Updates

User approval required for UI only

Notification Updates

User approval required for UI only
Notifications can have user actions



Notification Updates

User approval required for UI only

Notifications can have user actions

Location-based notifications

Notification Updates

User approval required for UI only

Notifications can have user actions

Location-based notifications

Increased push payload size (256 bytes → 1k)

More Notifications

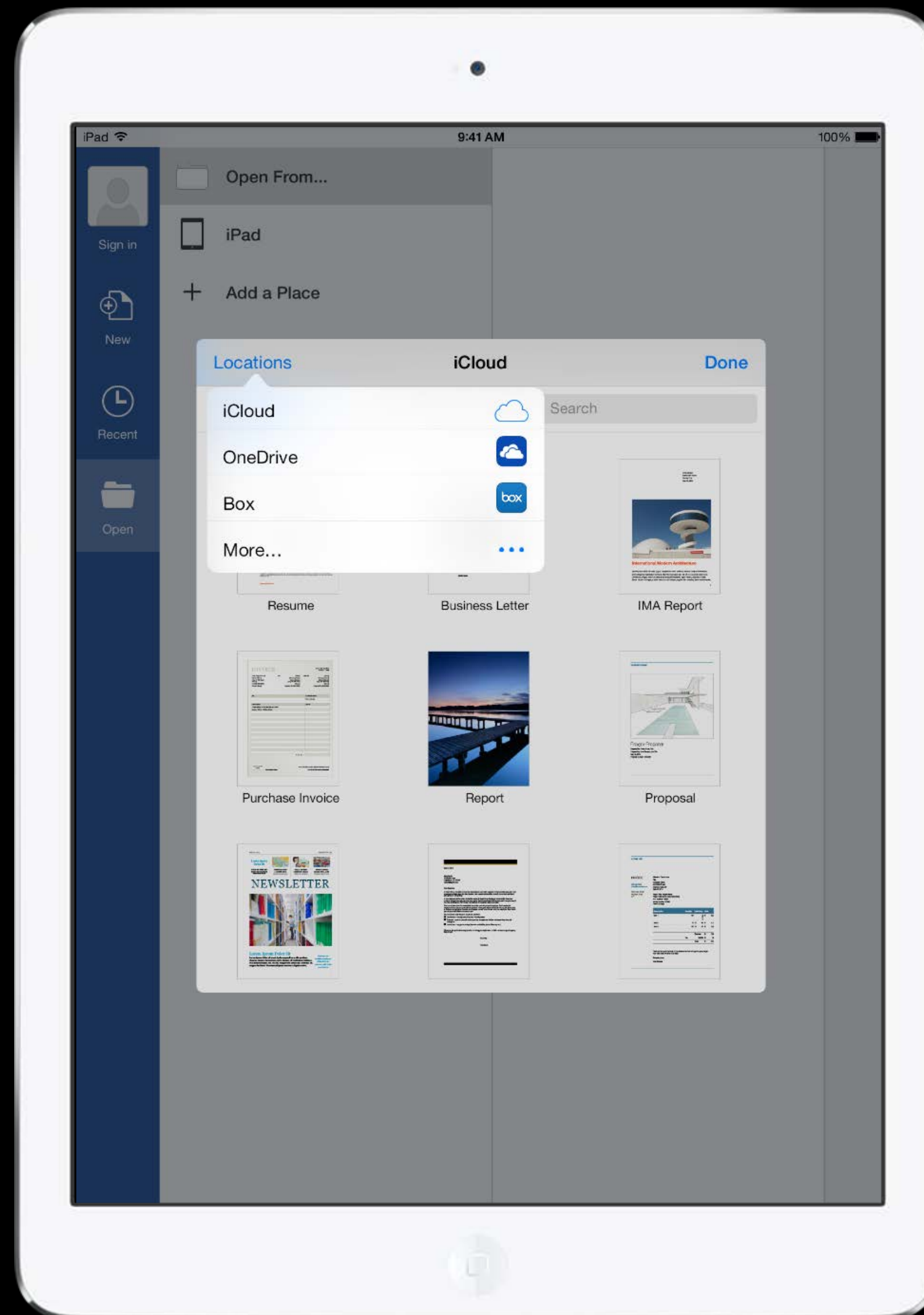
-
- What's New in iOS Notifications

Nob Hill

Wednesday 2:00PM

Document Picker

UIDocumentPickerViewController



UIDocumentPickerViewController

UIDocumentPickerViewController

System UI for selecting documents

UIDocumentPickerViewController

System UI for selecting documents

- Local documents

UIDocumentPickerViewController

System UI for selecting documents

- Local documents
- iCloud documents

UIDocumentPickerViewController

System UI for selecting documents

- Local documents
- iCloud documents
- Third-party document providers

More Document Support

-
- Building a Document-based App

Marina

Thursday 11:30AM

SDK Modernization

SDK Modernization

SDK Modernization

NS_DESIGNATED_INITIALIZER

SDK Modernization

NS_DESIGNATED_INITIALIZER

id → instancetype

SDK Modernization

NS_DESIGNATED_INITIALIZER

id → instancetype

Additional @properties

Handoff

Handoff

Handoff

User activities shared between multiple devices

Handoff

User activities shared between multiple devices

Built-in support in UIKit and AppKit

Handoff

-
- Adopting Handoff on iOS and OS X

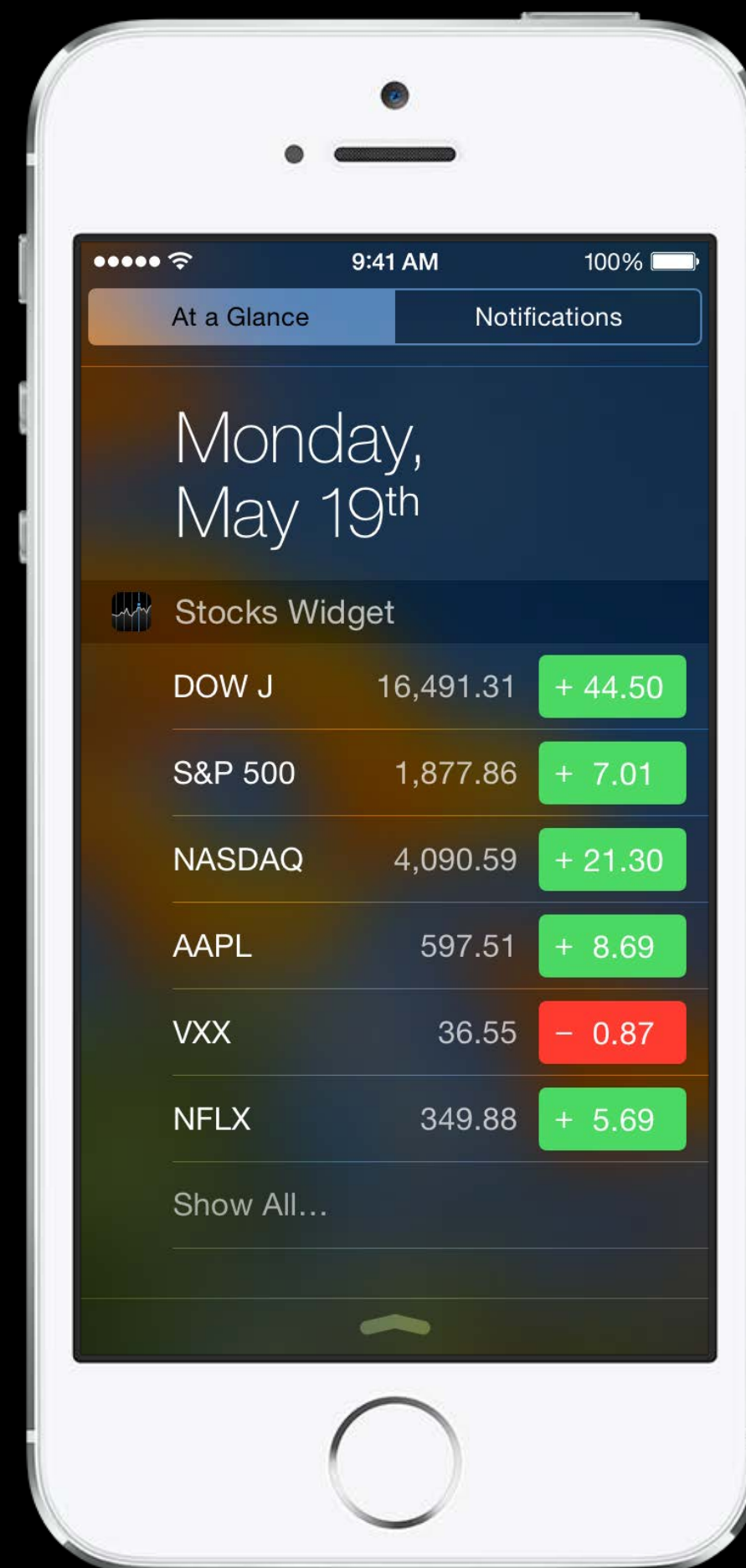
Mission

Wednesday 2:00PM

More Goodies in iOS

Notification Center

Custom widgets



Photos

-
- Introducing the Photos Framework

Nob Hill

Thursday 10:15AM

Photos

Read/write access to photos library

-
- Introducing the Photos Framework

Nob Hill

Thursday 10:15AM

Photos

Read/write access to photos library

Custom CoreImage filters

- Introducing the Photos Framework

Nob Hill

Thursday 10:15AM

CloudKit

- Introducing CloudKit

Mission

Tuesday 3:15PM

- Advanced CloudKit

Mission

Thursday 3:15PM

CloudKit

More control over data you put in the cloud

- Introducing CloudKit

Mission

Tuesday 3:15PM

- Advanced CloudKit

Mission

Thursday 3:15PM

CloudKit

More control over data you put in the cloud

Build client-server apps without building a server

- [Introducing CloudKit](#)

Mission

Tuesday 3:15PM

- [Advanced CloudKit](#)

Mission

Thursday 3:15PM

HealthKit

-
- Introducing HealthKit

Mission

Tuesday 10:15AM

HealthKit

Access information from biometric accessories in one framework

-
- Introducing HealthKit

Mission

Tuesday 10:15AM

HomeKit

-
- Introducing HomeKit

Mission

Tuesday 4:30PM

HomeKit

Access information from connected home accessories in one framework

-
- Introducing HomeKit

Mission

Tuesday 4:30PM

Local Authentication

-
- Keychain and Authentication with Touch ID

Nob Hill

Wednesday 10:15AM

Local Authentication

Leverage biometric authentication in your app

-
- Keychain and Authentication with Touch ID

Nob Hill

Wednesday 10:15AM

Local Authentication

Leverage biometric authentication in your app

- TouchID

- Keychain and Authentication with Touch ID

Nob Hill

Wednesday 10:15AM

SceneKit

-
- What's New in SceneKit Pacific Heights Thursday 10:15AM
 - Building a Game with SceneKit Pacific Heights Thursday 11:30AM
-

SceneKit

Cross-platform 3D rendering for iOS

-
- What's New in SceneKit Pacific Heights Thursday 10:15AM
 - Building a Game with SceneKit Pacific Heights Thursday 11:30AM
-

Core Location

-
- What's New in Core Location Marina Tuesday 2:00PM
 - Taking Core Location Indoors Marina Tuesday 3:15PM
-

Core Location

Indoor location

-
- What's New in Core Location Marina Tuesday 2:00PM
 - Taking Core Location Indoors Marina Tuesday 3:15PM
-

Core Location

Indoor location

Power-saving location updates

- What's New in Core Location

Marina

Tuesday 2:00PM

- Taking Core Location Indoors

Marina

Tuesday 3:15PM

Core Location

Indoor location

Power-saving location updates

Always vs. when-in-use user approval

- What's New in Core Location

Marina

Tuesday 2:00PM

- Taking Core Location Indoors

Marina

Tuesday 3:15PM

Summary

Summary

Simplify layout with size classes

Summary

Simplify layout with size classes

Unify iPhone and iPad code with adaptive view controllers

Summary

Simplify layout with size classes

Unify iPhone and iPad code with adaptive view controllers

Integrate into the system experience with app extensions

Summary

Simplify layout with size classes

Unify iPhone and iPad code with adaptive view controllers

Integrate into the system experience with app extensions

Expand your tool set with new frameworks in iOS 8

More Information

Jake Behrens
Frameworks Evangelist
behrens@apple.com

Documentation
<http://developer.apple.com>

Apple Developer Forums
<http://devforums.apple.com>

Related Sessions

-
- [Introducing HealthKit](#) Mission Tuesday 10:15AM

 - [What's New in Core Location](#) Marina Tuesday 2:00PM

 - [Introducing CloudKit](#) Mission Tuesday 3:15PM

 - [Taking Core Location Indoors](#) Marina Tuesday 3:15PM

 - [Introducing HomeKit](#) Mission Tuesday 4:30PM

 - [Keychain and Authentication with Touch ID](#) Nob Hill Wednesday 10:15AM

 - [Advanced CloudKit](#) Mission Thursday 3:15PM

Related Sessions

-
- | | | |
|------------------------------------|-----------------|------------------|
| ● Introducing the Photos Framework | Nob Hill | Thursday 10:15AM |
| ● What's New in SceneKit | Pacific Heights | Thursday 10:15AM |
| ● Building a Game with SceneKit | Pacific Heights | Thursday 11:30AM |
-

Labs

-
- Cocoa Touch Lab Frameworks Lab A Tuesday 12:30PM
 - Cocoa Touch Lab Frameworks Lab A Wednesday 11:30AM
 - Cocoa Touch Lab Frameworks Lab A Thursday 2:00PM
-

 WWDC14