Taking iPad Apps for Mac to the Next Level

Jamie Montgomerie, UIKit
Glen Low, UIKit
Nils Beck, AppKit
Chris D'Angelo, Xcode
Taking your app further
Application lifecycle
Distributing your application
Making Your UIKit App a Better Mac App
Better iPad apps are better Mac apps
Support Varying Screen Sizes
Implement Great Keyboard Support

Use `UIKeyCommand`

Adopt standard responder actions

Look after your responder chain
Make Great Use of Drag and Drop

Use the UIKit Drag and Drop APIs

- UIDragInteraction
- UIDropInteraction
Make Great Use of Drag and Drop

Use the UIKit Drag and Drop APIs

- UIDragInteraction
- UIDropInteraction
Use the Latest APIs

**WKWebView**, not **UIWebView**

Metal, not OpenGL ES, etc.

Most deprecated iOS APIs are unavailable on Mac
Support New iOS 13 features

Multi Window
Dark Mode
Symbol Images
Etc.
Better Mac apps contain refinements for Mac
The Menu Bar

New UIKit APIs

- UIKeyCommand
- UICommand
- UIMenu
- UIMenuBuilder
Context Menus

New UIKit APIs

- UIContextMenuInteraction
- UIAction
- UICommand

Expressed as context menu on Mac
Context Menus

New UIKit APIs
• `UIContextMenuInteraction`
• `UIAction`
• `UICommand`

Expressed as context menu on Mac
Context Menus

New UIKit APIs

- `UIContextMenuInteraction`
- `UIAction`
- `UICommand`

Expressed as context menu on Mac
Sidebars

Mostly Old UIKit APIs

- **UISplitViewController** Properties
  - minimumPrimaryColumnWidth
  - maximumPrimaryColumnWidth
  - primaryBackgroundStyle

- **UITableView** Changes Style
  - UITableViewStyleGrouped
  - UITableViewStyleInsetGrouped
Hover

New UIKit API

- **UIHoverGestureRecognizer**

- Works just like other gesture recognizers.
Hover

New UIKit API

- UIHoverGestureRecognizer
- Works just like other gesture recognizers.
Mac Toolbars

Dip your toes into AppKit API!

- NSToolbar
- UIWindowScene.titlebar.toolbar
Continue diving into AppKit

- `UIResponder.touchbar`
- `NSTouchBar`
- `UIViewController.childViewControllerForTouchBar`
- `UIViewController.setNeedsTouchBarUpdate`
Other Mac Features

- Take control of window sizing
- Implement support for printing
- Author a Help Book
- Customize Assets and Strings
Design Considerations
Navigation

Use a sidebar

Reconsider use of tab bars

Make use of the Mac toolbar
Layout

Be flexible in layout

Use the space in the big window

Reflow, redesign, or reword
Typography

Baseline font sizes are not the same
• Mac: 13 points
• iPad: 17 points

UIKit apps are scaled to 77% on Mac
Typography

Baseline font sizes are not the same

• Mac: 13 points
• iPad: 17 points

UIKit apps are scaled to 77% on Mac
The Menu Bar

What should go in the menu bar?

Think about your app’s complete functionality
iPad is optimized for multitouch

Macs always have keyboards

Macs always have trackpads or mice

Consider accessibility too
More Design Resources

Design for iPad Apps on Mac

WWDC 2019
Application Lifecycle

Nils Beck, AppKit
iOS App Lifecycle ≠ macOS App Lifecycle
App Lifecycle States

iOS Review

Foreground + Active
• Normal interaction
App Lifecycle States

iOS Review

Foreground + Inactive
- Not receiving events
- Temporary interruption

Diagram:
- Not running
- Foreground
  - Active
  - Inactive
- Background
  - Background
  - Suspended
App Lifecycle States

iOS Review

Background
• About to be suspended
• Background tasks allowed
App Lifecycle States

iOS Review

Suspended
- No code is executed
- May be killed without warning
App Lifecycle States

iOS Review

Not running
- Not in memory

[Diagram showing the app lifecycle states: Not running, Background, Suspended, Inactive, Active, Foreground]
// UIApplication Lifecycle Notifications
    .didFinishLaunchingNotification
    .didBecomeActiveNotification
    .willResignActiveNotification
    .willEnterForegroundNotification
    .didEnterBackgroundNotification
    .willTerminateNotification
App Lifecycle States

State Changes

// UIApplication Lifecycle Notifications
.didFinishLaunchingNotification
.didBecomeActiveNotification
.willResignActiveNotification
.willEnterForegroundNotification
.didEnterBackgroundNotification
.willTerminateNotification

Not running

Foreground

Active

Inactive

Background

Background

Suspended
App Lifecycle States

State Changes

// UIApplication Lifecycle Notifications
    .didFinishLaunchingNotification
    .didBecomeActiveNotification
    .willResignActiveNotification
    .willEnterForegroundNotification
    .didEnterBackgroundNotification
    .willTerminateNotification

Not running

Foreground

Active -> Inactive

Background

Suspended
App Lifecycle States
State Changes

// UIApplication Lifecycle Notifications
- didFinishLaunchingNotification
- didBecomeActiveNotification
- willResignActiveNotification
- willEnterForegroundNotification
- didEnterBackgroundNotification
- willTerminateNotification
App Lifecycle States
State Changes

// UIApplication Lifecycle Notifications
- didFinishLaunchingNotification
- didBecomeActiveNotification
- willResignActiveNotification
- willEnterForegroundNotification
- didEnterBackgroundNotification
- willTerminateNotification

Not running

Foreground
- Active
- Inactive

Background
- Background

Suspended
App Lifecycle States
State Changes

// UIApplication Lifecycle Notifications
.didFinishLaunchingNotification
.didBecomeActiveNotification
.willResignActiveNotification
.willEnterForegroundNotification
.didEnterBackgroundNotification
.willTerminateNotification

[Diagram showing state transitions from Not running to Foreground (Active), Inactive, Background, and Suspended]
App Lifecycle States

State Changes

// UIApplication Lifecycle Notifications
    .didFinishLaunchingNotification
    .becomeActiveNotification
    .willResignActiveNotification
    .willEnterForegroundNotification
    .didEnterBackgroundNotification
    .willTerminateNotification

Not running

Foreground
    - Active
    - Inactive

Background
    - Background

Suspended

Inactive

Active

Not running
App Lifecycle States
State Changes

// UIApplication Lifecycle Notifications
.didFinishLaunchingNotification
.didBecomeActiveNotification
.willResignActiveNotification
.willEnterForegroundNotification
.didEnterBackgroundNotification
.willTerminateNotification

Not running
Active ➔ Inactive

Foreground
Inactive ➔ Active

Background
Active ➔ Inactive

Suspended
Active ➔ Background

Inactive ➔ Suspended
App Lifecycle States

State Changes

// UIApplication Lifecycle Notifications
- didFinishLaunchingNotification
- didBecomeActiveNotification
- willResignActiveNotification
- willEnterForegroundNotification
- didEnterBackgroundNotification
- willTerminateNotification

[Diagram showing the lifecycle states: Not running, Foreground (Active, Inactive), Background, Suspended]
Responding to State Changes

iOS Review

Deactivation

- Reduce frame rate
- Reduce amount of work
- Pause game
Responding to State Changes
iOS Review

Backgrounding
• Stop rendering
• Minimize CPU usage
• Free up memory
Your iPad app is now a Mac app
macOS Considerations

Frontmost app
• Receives key events
• Controls menu bar
macOS Considerations

Window visibility
• Occluded
• Minimized
• In another space
macOS Considerations

App visibility
• App hidden
• No windows
• Different login session
State changes affect Mac apps less
Foreground + Active
AppNap applies to iPad apps

- macOS observes apps for lack of use
- Automatic throttling when appropriate
App Termination
Comparison

Switching between apps on iOS
• Multiple macOS mappings

Can be similar to
• Changing frontmost Mac app
• Terminating Mac app
App Termination
iPad Apps for Mac

⌘ Q goes to background state
Immediately looks "not running"
Background tasks may complete

No background audio — Will pause

```swift
let task = UIApplication.shared.beginBackgroundTask {
    // handle task expiration
}

UIApplication.shared.endBackgroundTask(task)
```
App Termination
iPad Apps for Mac

User can "relaunch" during background tasks

Keep willTerminate short
App Termination
iPad Apps for Mac

Process exits after last task
Background App Launch

Several iOS APIs launch to background
Background App Launch
iPad Apps for Mac

Several iOS APIs launch to background

Subset supported
- Background URL sessions
- Silent push notifications
- Notification actions
- BackgroundTasks (BGProcessingTask, BGAppRefreshTask)
Background App Launch
iPad Apps for Mac

Usual restrictions apply
User can "launch" app fully
Standard state transitions
Suspension
iPad Apps for Mac

Apps rarely suspended
Suspension = immediate termination
No memory limits
Apps live longer: Fix leaks
Suspension
iPad Apps for Mac

Apps rarely suspended
Suspension = immediate termination
No memory limits
Apps live longer: Fix leaks
Suspension
iPad Apps for Mac

Apps rarely suspended
Suspension = immediate termination
No memory limits
Apps live longer: Fix leaks
Summary

Usually Foreground + Active

Background only during
• Termination
• Background launches

No background audio

Handle state changes as usual
Distribution

Chris D’Angelo, Xcode
**identity**

- App Category: Lifestyle
- Display Name: ChocolateChip
- Bundle Identifier: com.cookies.ChocolateChip
- Version: 1.0
- Build: 5

**Deployment Info**

<table>
<thead>
<tr>
<th>Target</th>
<th>Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS 13.0</td>
<td>iPhone</td>
</tr>
<tr>
<td></td>
<td>iPad</td>
</tr>
<tr>
<td></td>
<td>Mac (requires macOS 10.15)</td>
</tr>
</tbody>
</table>

- Main Interface: Main
- Device Orientation: Portrait, Landscape Left, Landscape Right
- Status Bar Style: Default, Hide status bar
Signing for Mac
Signing for Mac

Use Automatic Signing
Signing for Mac

Use Automatic Signing

One Profile for Mac, one for iOS
Signing for Mac

Use Automatic Signing

One Profile for Mac, one for iOS
Signing for Mac

Use Automatic Signing

One Profile for Mac, one for iOS
Signing for Mac

Use Automatic Signing

One Profile for Mac, one for iOS

Apple Development Certificates
Signing for Mac

Use Automatic Signing

One Profile for Mac, one for iOS

Apple Development Certificates
Signing for Mac

Use Automatic Signing

One Profile for Mac, one for iOS

Apple Development Certificates

Unique Bundle Identifier
Signing for Mac

Use Automatic Signing

One Profile for Mac, one for iOS

Apple Development Certificates

Unique Bundle Identifier
Signing for Mac

Capabilities

Share Entitlements
Signing for Mac

Capabilities

Share Entitlements
Signing for Mac
Capabilities

Share Entitlements

Mac Capabilities added
Signing for Mac
Capabilities

Share Entitlements

Mac Capabilities added
Signing for Mac
Capabilities

Share Entitlements

Mac Capabilities added

Explicit iCloud Container
Signing for Mac

Capabilities

Share Entitlements

Mac Capabilities added

Explicit iCloud Container
Capabilities
Manual migration checklist
Capabilities
Manual migration checklist

Add Keychain Sharing Capability
Capabilities
Manual migration checklist

- Add Keychain Sharing Capability
- Send push notifications to your iOS app
<table>
<thead>
<tr>
<th>Name</th>
<th>Creation Date</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>ChocolateChip</td>
<td>May 29, 2019 at 8:36 PM</td>
<td>1.0 (1)</td>
</tr>
</tbody>
</table>

**Archive Information**

- **ChocolateChip**
  - May 29, 2019 at 8:36 PM

**Details**

- Version 1.0 (1)
- Identifier: uikitformac.com.cookies.Ch...
- Type: macOS App Archive

**Description**

No Description
$ xcodebuild archive
   -scheme ChocolateChip\n   -archivePath ChocolateChip.xcarchive

...
** ARCHIVE SUCCEEDED **

$ xcodebuild archive
   -scheme ChocolateChip\
   -archivePath ChocolateChip.xcarchive

...  

** ARCHIVE SUCCEEDED **

$ xcodebuild -exportArchive\
   -archivePath ChocolateChip.xcarchive\
   -exportOptionsPlist ExportOptions.plist

...  

** EXPORT SUCCEEDED **
$ xcodebuild archive
    -scheme ChocolateChip\n    -archivePath ChocolateChip.xcarchive

... ** ARCHIVE SUCCEEDED **

$ xcodebuild -exportArchive\n    -archivePath ChocolateChip.xcarchive\n    -exportOptionsPlist ExportOptions.plist

... ** EXPORT SUCCEEDED **
Mac App Store

Getting started
Mac App Store
Getting started

✔ Create an App Record on App Store Connect
Mac App Store
Getting started

- Create an App Record on App Store Connect
- Associate your Mac synthesized App Identifier
Mac App Store
Getting started

- Create an App Record on App Store Connect
- Associate your Mac synthesized App Identifier
- Always increment your build number
Mac App Store
Getting started

- Create an App Record on App Store Connect
- Associate your Mac synthesized App Identifier
- Always increment your build number
- Upload from Xcode
Mac App Store
Getting started

- Create an App Record on App Store Connect
- Associate your Mac synthesized App Identifier
- Always increment your build number
- Upload from Xcode
- Release for sale
Drag and Drop Installation
Drag and Drop Installation

Mac App Store migration checklist
Drag and Drop Installation
Mac App Store migration checklist

✅ App Thinning is unsupported
Drag and Drop Installation
Mac App Store migration checklist

- App Thinning is unsupported
- Add Receipt Validation for DRM
Drag and Drop Installation
Mac App Store migration checklist

- App Thinning is unsupported
- Add Receipt Validation for DRM
Chocolate Chip Cookies

INGREDIENTS

- Baking Soda
  0.5 teaspoons
- Brown Sugar
  9.25 ounces
- Semisweet Chocolate Chips
  1.25 cups
- Unsalted Butter
  14.0 tablespoons
- Vanilla Extract
  2.0 teaspoons

STEPS

Heat oven to 375 degrees.
Capabilities
Mac App Store migration checklist
Capabilities
Mac App Store migration checklist

- Recreate In-App Purchases and subscriptions
Capabilities
Mac App Store migration checklist

- ✔ Recreate In-App Purchases and subscriptions
- ✔ Track purchase history server-side
Capabilities
Mac App Store migration checklist

- ✔️ Recreate In-App Purchases and subscriptions
- ✔️ Track purchase history server-side
- ✔️ Use Game Center Groups
Capabilities
Mac App Store migration checklist

- ✓ Recreate In-App Purchases and subscriptions
- ✓ Track purchase history server-side
- ✓ Use Game Center Groups
- ✓ Update Game Center Multiplayer Compatibility
Mac App Store
Submission timeline
Mac App Store
Submission timeline

Create new App Record

WWDC 2019

Upload to App Store Connect

Summer 2019
Mac App Store
Submission timeline

- WWDC 2019: Create new App Record
- Summer 2019: Upload to App Store Connect
- Xcode 11 GM: Submit for App Review
Notarized Apps

Highlights

Apple verified signature
Notarized Apps
Highlights

Apple verified signature

Hardened Runtime required
Notarized Apps

Highlights

Apple verified signature

Hardened Runtime required

`altool`, `stapler`, `xcodebuild` command line support
Notarized Apps
Highlights

Apple verified signature

Hardened Runtime required

`altool`, `stapler`, `xcodebuild` command line support

Alternative to TestFlight
Notarized Apps

Highlights

Apple verified signature

Hardened Runtime required

`altool`, `stapler`, `xcodebuild` command line support

Alternative to TestFlight

GameKit and StoreKit are unavailable
Notarized Apps

Getting started
Notarized Apps
Getting started

✓ Upload app with Xcode
Notarized Apps
Getting started

- Upload app with Xcode
- Wait for Xcode notification
Notarized Apps

Getting started

- Upload app with Xcode
- Wait for Xcode notification
- Export Notarized app
Notarized Apps
Getting started

- ✔ Upload app with Xcode
- ✔ Wait for Xcode notification
- ✔ Export Notarized app
- ✔ Host your app
Notarized Apps
Getting started

- Upload app with Xcode
- Wait for Xcode notification
- Export Notarized app
- Host your app

All About Notarization  
WWDC 2019
Development Signing

Highlights

Same as iOS
Development Signing

Highlights

Same as iOS

Alternative to TestFlight
Development Signing

Highlights

Same as iOS

Alternative to TestFlight

StoreKit and GameKit are available
# Chocolate Chip Cookies

## Ingredients

<table>
<thead>
<tr>
<th>Item</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baking Soda</td>
<td>0.5 teaspoon</td>
</tr>
<tr>
<td>Brown Sugar</td>
<td>3.5 tablespoons</td>
</tr>
<tr>
<td>Eggs</td>
<td>1 egg</td>
</tr>
<tr>
<td>Egg Yolk</td>
<td>1/2 egg</td>
</tr>
<tr>
<td>Flour</td>
<td>8 1/4 cups</td>
</tr>
<tr>
<td>Salt</td>
<td>1 1/2 teaspoons</td>
</tr>
<tr>
<td>Semisweet Chocolate Chips</td>
<td>1 cup</td>
</tr>
<tr>
<td>Unsalted Butter</td>
<td>1/4 cup</td>
</tr>
<tr>
<td>Vanilla Extract</td>
<td>2 teaspoons</td>
</tr>
</tbody>
</table>

## Steps

1. Heat oven to 375 degrees.
3. Mix sugars, salt, eggs, butter, and vanilla.
4. Mix in flour, baking soda, and chocolate.
5. Bake for 12 minutes.
## More Information

[developer.apple.com/wwdc19/235](developer.apple.com/wwdc19/235)

<table>
<thead>
<tr>
<th>Event</th>
<th>Time/Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>iPad Apps for Mac Lab</td>
<td>Friday, 10:00</td>
</tr>
<tr>
<td>Introducing iPad Apps for Mac</td>
<td>WWDC 2019</td>
</tr>
</tbody>
</table>