

Advanced Notifications

Session 708

Michele Campeotto

iOS Notifications

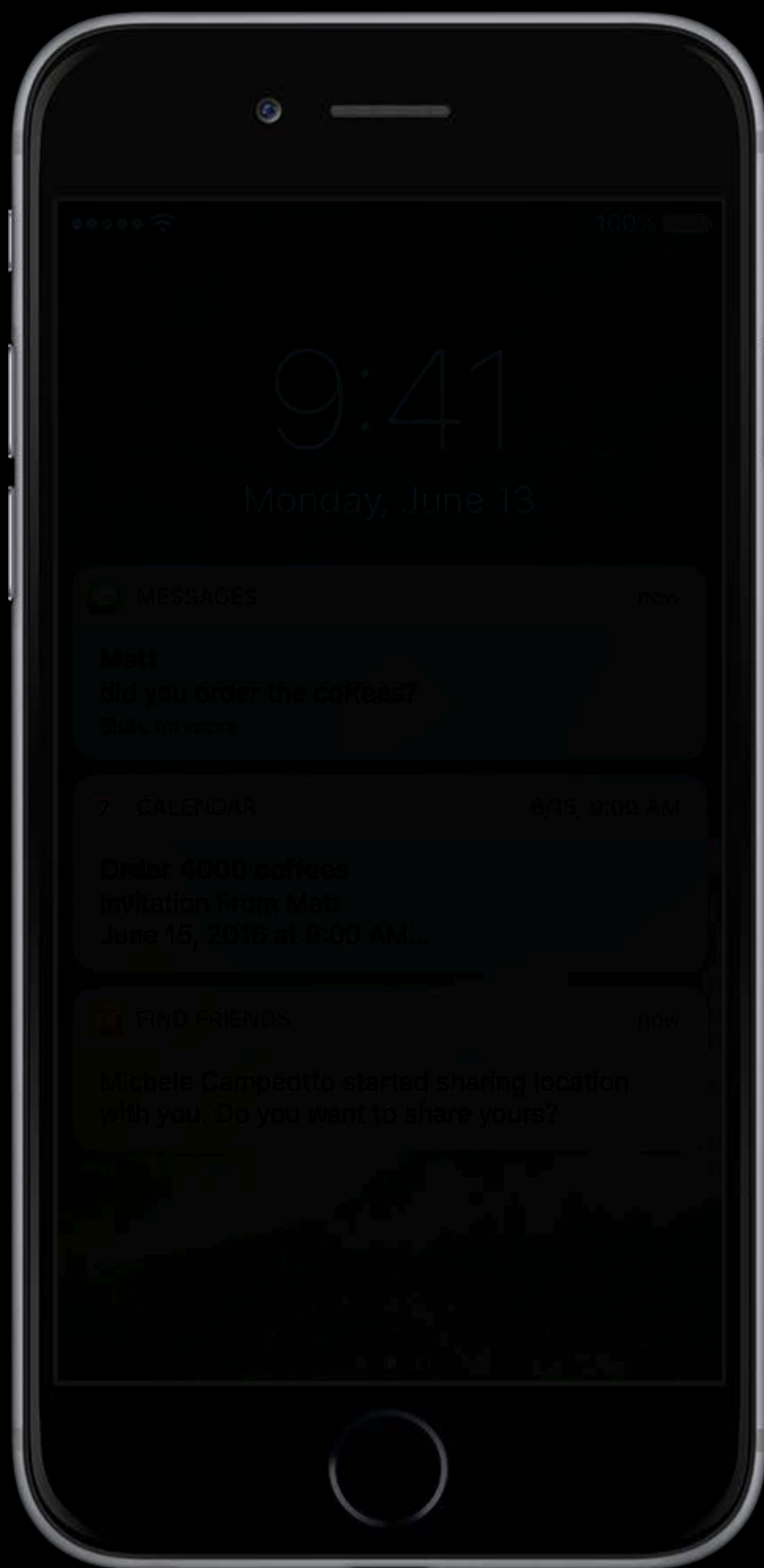
Agenda

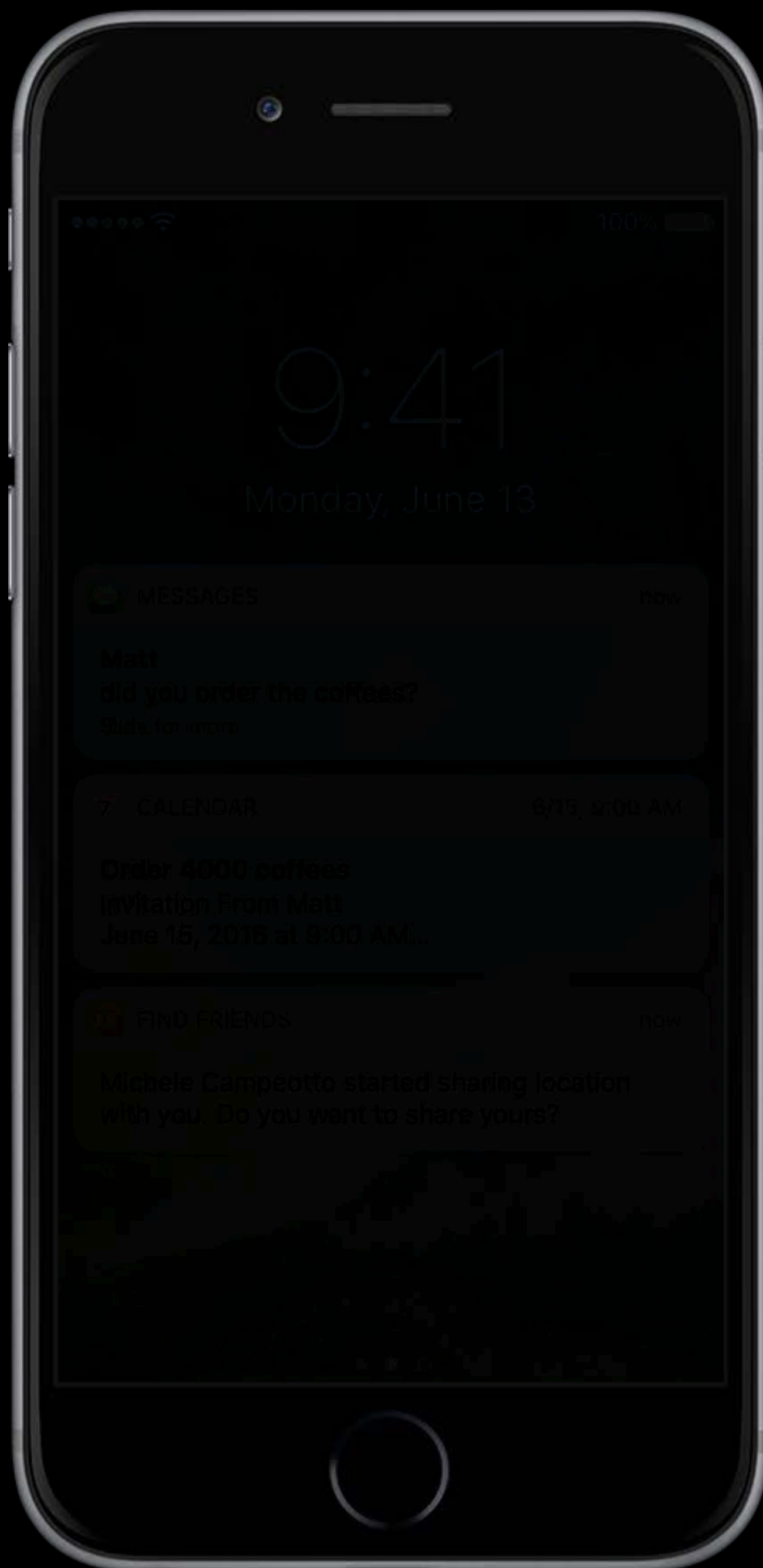
Notifications User Interface

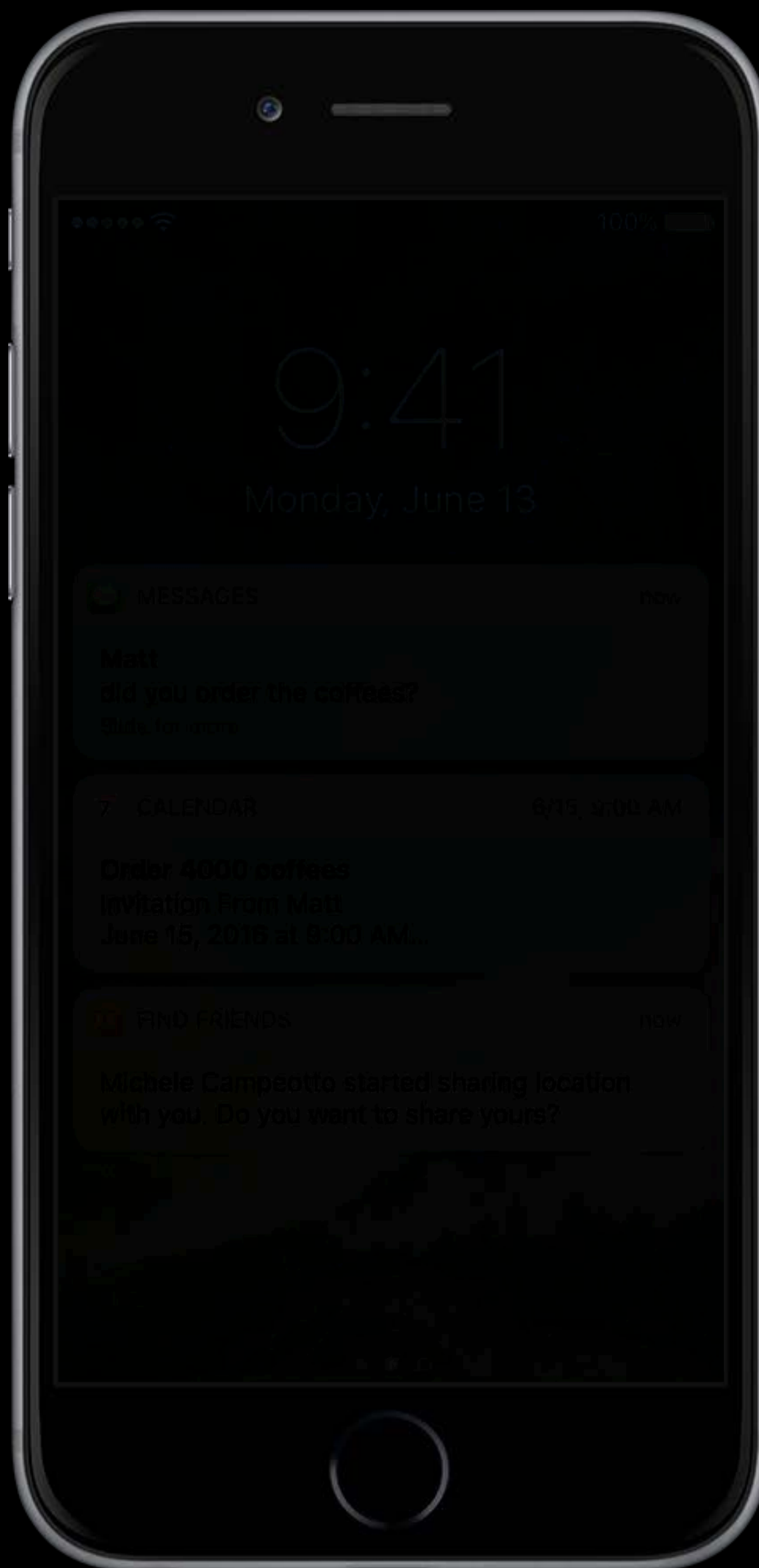
Media Attachments

Customization

Notifications User Interface









100%

9:41

Monday, June 13

15

CALENDAR

6/15, 9:00 AM

Order 4000 coffees

Invitation From Matt

June 15, 2016 at 9:00 AM...

Slide for more

Press home to open





100%

9:41

Monday, June 13



MESSAGES

now

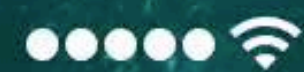
Matt

did you order the coffees?

Slide for more

Press home to open





100%

9:41

Monday, June 13

Tuesday
7

CALENDAR

6/15, 9:00 AM

Order 4000 coffees

Invitation From Matt

June 15, 2016 at 9:00 AM...

Press for more

Press home to open



Tuesday
7

CALENDAR



8 AM

9 AM

• Order 4000 coffees

Moscone Center

10 AM 747 Howard St, San Francisco, CA 94103, United States

11 AM Feed the lemurs

Noon

Order 4000 coffees

Invitation From Matt

June 15, 2016 at 9:00 AM

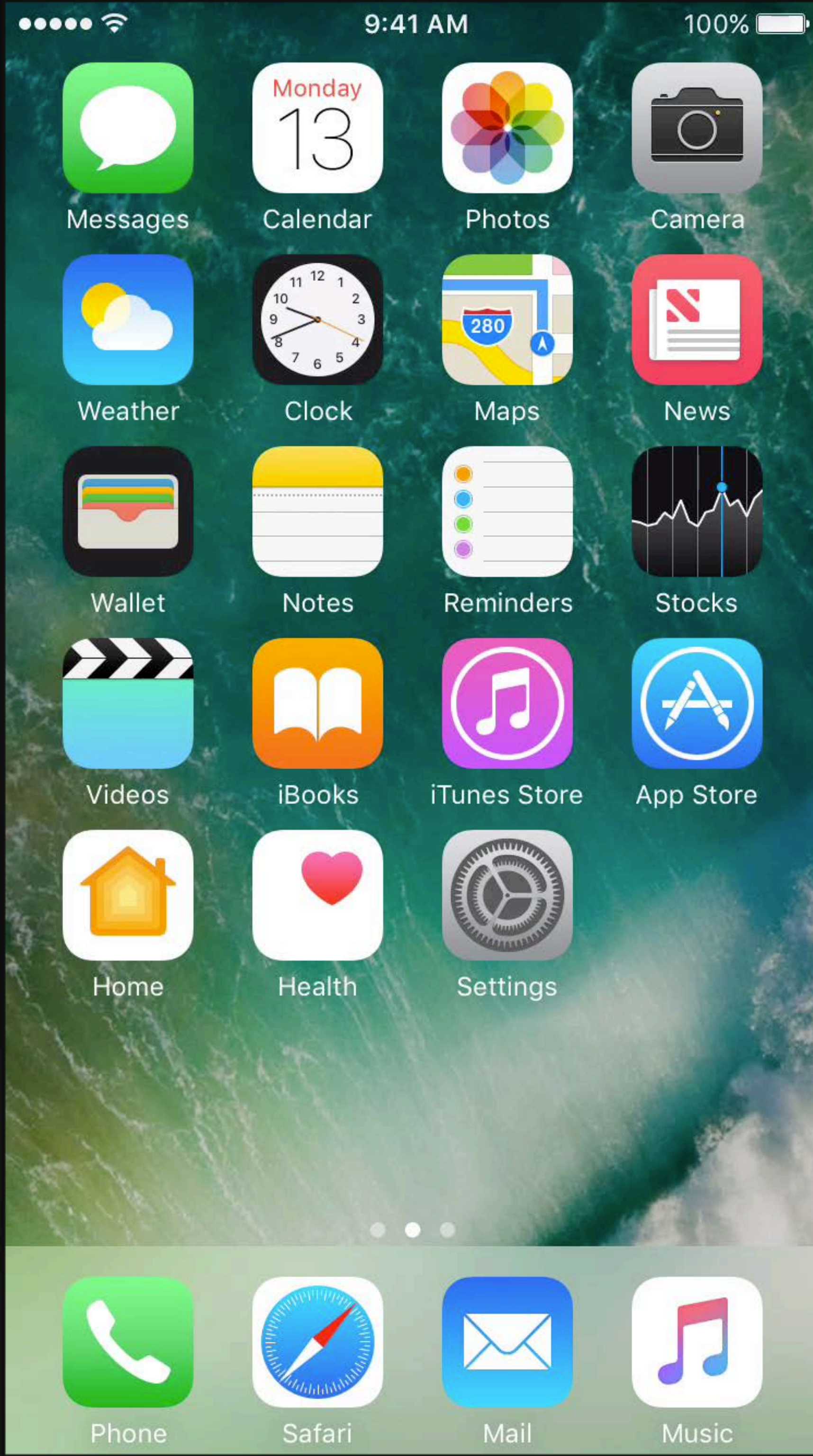
Moscone Center

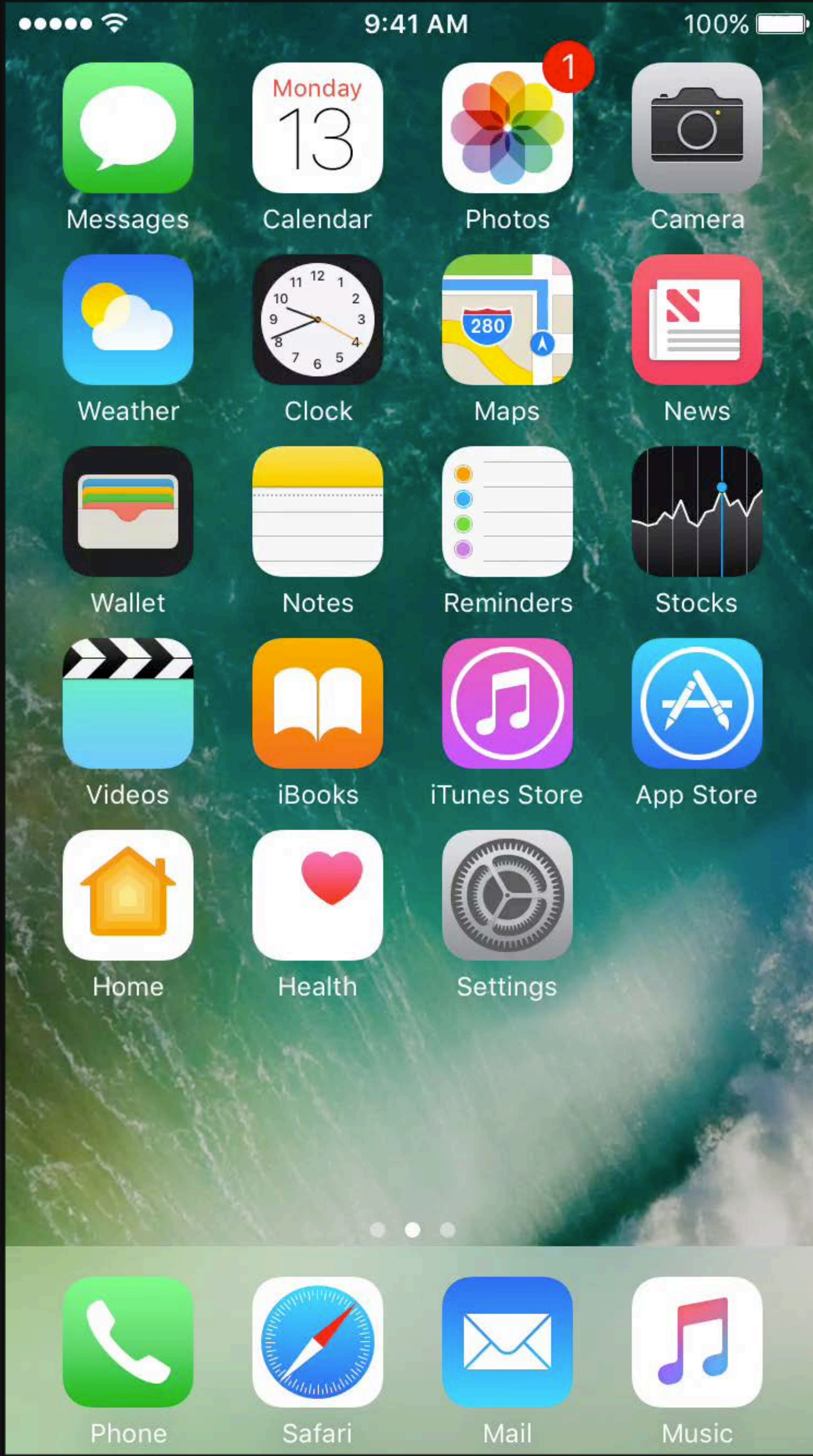
747 Howard St, San Francisco, CA 94103,
United States

Accept

Maybe

Decline







MESSAGES

now

Matt

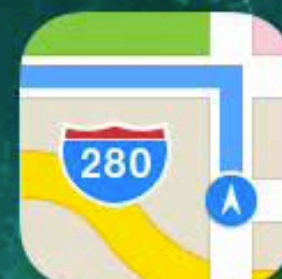
did you order the coffees?



Weather



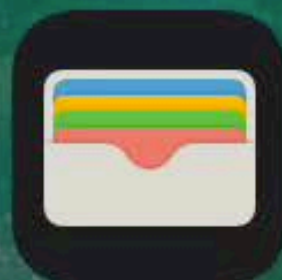
Clock



Maps



News



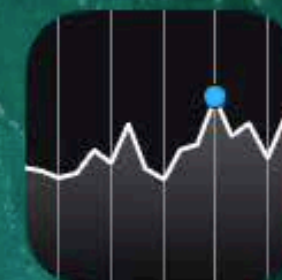
Wallet



Notes



Reminders



Stocks



Videos



iBooks



iTunes Store



App Store



Home



Health



Settings



Phone



Safari



Mail



Music

Media Attachments

Media Attachments



Media Attachments



```
{
  aps: {
    alert: { ... },
    mutable-content: 1
  }
  my-attachment: "https://example.com/photo.jpg"
}
```


Media Attachments



```
{
  aps: {
    alert: { ... },
    mutable-content: 1
  }
  my-attachment: "https://example.com/photo.jpg"
}
```

Media Attachments

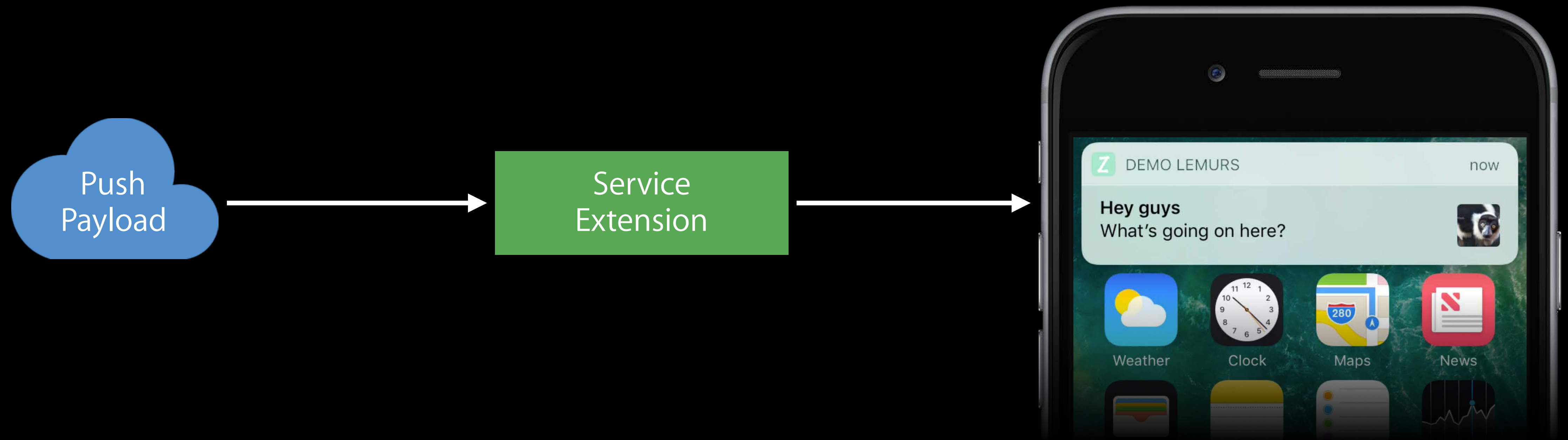


```
{
  aps: {
    alert: { ... },
    mutable-content: 1
  }
  my-attachment: "https://example.com/photo.jpg"
}
```


Media Attachments



Media Attachments




```
// Adding an attachment to a user notification
```

```
public class NotificationService: UNNotificationServiceExtension {  
    override public func didReceive(_ request: UNNotificationRequest,  
        withContentHandler contentHandler: (UNNotificationContent) -> Void) {  
  
        let fileURL = // ...  
  
        let attachment = UNNotificationAttachment(identifier: "image",  
                                                    url: fileURL,  
                                                    options: nil)  
  
        let content = request.content.mutableCopy as! UNMutableNotificationContent  
        content.attachments = [ attachment ]  
  
        contentHandler(content)  
    }  
}
```

```
// Adding an attachment to a user notification
```

```
public class NotificationService: UNNotificationServiceExtension {  
    override public func didReceive(_ request: UNNotificationRequest,  
        withContentHandler contentHandler: (UNNotificationContent) -> Void) {  
  
        let fileURL = // ...  
  
        let attachment = UNNotificationAttachment(identifier: "image",  
                                                    url: fileURL,  
                                                    options: nil)  
  
        let content = request.content.mutableCopy as! UNMutableNotificationContent  
        content.attachments = [ attachment ]  
  
        contentHandler(content)  
    }  
}
```



```
// Adding an attachment to a user notification
```

```
public class NotificationService: UNNotificationServiceExtension {  
    override public func didReceive(_ request: UNNotificationRequest,  
        withContentHandler contentHandler: (UNNotificationContent) -> Void) {  
  
        let fileURL = // ...  
  
        let attachment = UNNotificationAttachment(identifier: "image",  
                                                    url: fileURL,  
                                                    options: nil)  
  
        let content = request.content.mutableCopy as! UNMutableNotificationContent  
        content.attachments = [ attachment ]  
  
        contentHandler(content)  
    }  
}
```

```
// Adding an attachment to a user notification
```

```
public class NotificationService: UNNotificationServiceExtension {  
    override public func didReceive(_ request: UNNotificationRequest,  
        withContentHandler contentHandler: (UNNotificationContent) -> Void) {  
  
        let fileURL = // ...  
  
        let attachment = UNNotificationAttachment(identifier: "image",  
                                                    url: fileURL,  
                                                    options: nil)  
  
        let content = request.content.mutableCopy as! UNMutableNotificationContent  
        content.attachments = [ attachment ]  
  
        contentHandler(content)  
    }  
}
```

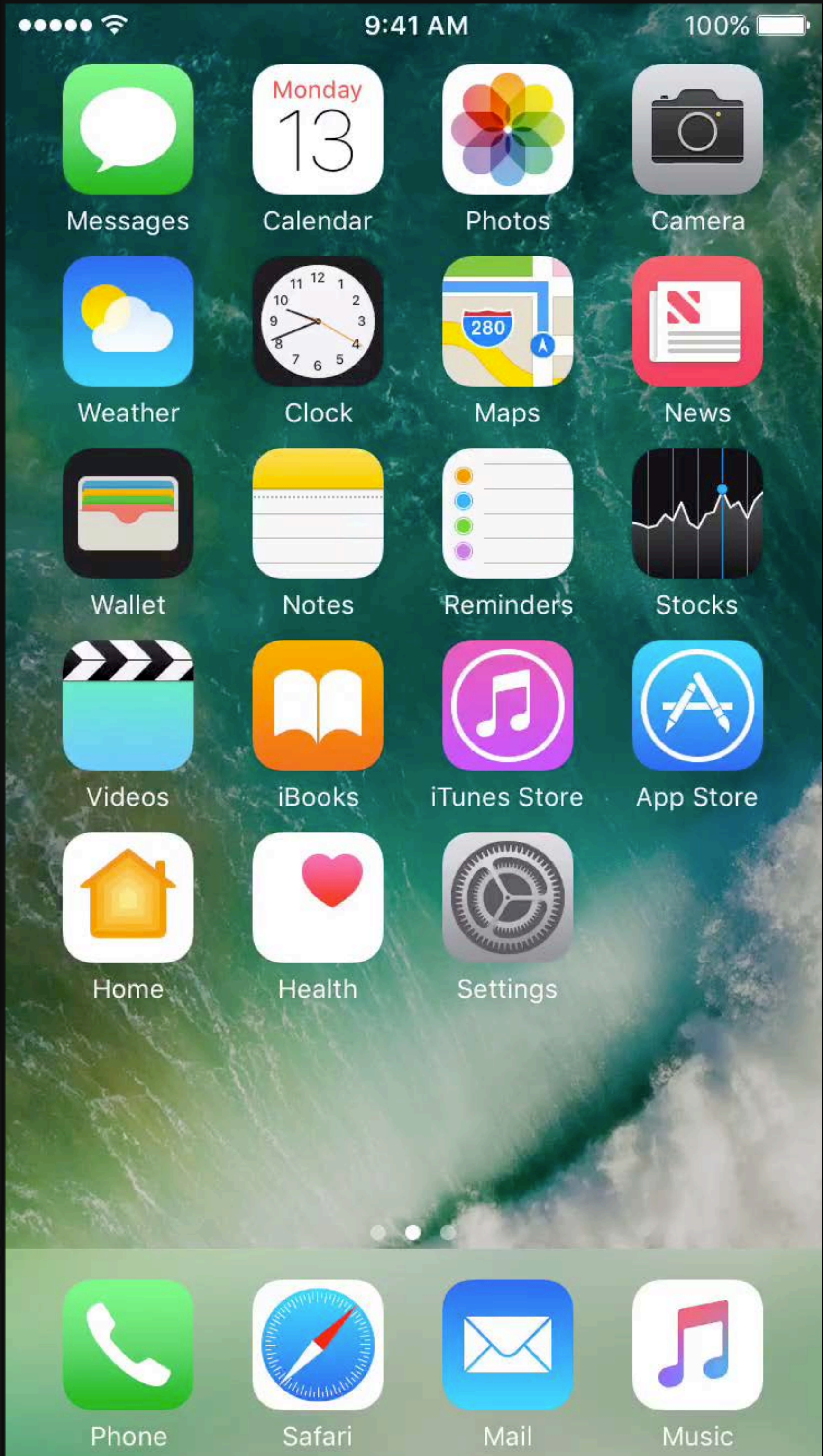


```
// Adding an attachment to a user notification
```

```
public class NotificationService: UNNotificationServiceExtension {  
    override public func didReceive(_ request: UNNotificationRequest,  
        withContentHandler contentHandler: (UNNotificationContent) -> Void) {  
  
        let fileURL = // ...  
  
        let attachment = UNNotificationAttachment(identifier: "image",  
                                                    url: fileURL,  
                                                    options: nil)  
  
        let content = request.content.mutableCopy as! UNMutableNotificationContent  
        content.attachments = [ attachment ]  
  
        contentHandler(content)  
    }  
}
```

```
// Adding an attachment to a user notification
```

```
public class NotificationService: UNNotificationServiceExtension {  
    override public func didReceive(_ request: UNNotificationRequest,  
        withContentHandler contentHandler: (UNNotificationContent) -> Void) {  
  
        let fileURL = // ...  
  
        let attachment = UNNotificationAttachment(identifier: "image",  
                                                    url: fileURL,  
                                                    options: nil)  
  
        let content = request.content.mutableCopy as! UNMutableNotificationContent  
        content.attachments = [ attachment ]  
  
        contentHandler(content)  
    }  
}
```

Media Attachments

Media Attachments

Local and remote notifications

Media Attachments

Local and remote notifications

Image, audio, video

Media Attachments

Local and remote notifications

Image, audio, video

Download in the service extension

Media Attachments

Local and remote notifications

Image, audio, video

Download in the service extension

- Limited processing time and size

Media Attachments

Local and remote notifications

Image, audio, video

Download in the service extension

- Limited processing time and size

Add attachment to notification

Media Attachments

Local and remote notifications

Image, audio, video

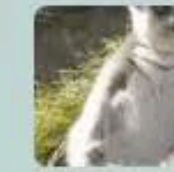
Download in the service extension

- Limited processing time and size

Add attachment to notification

- File is moved and managed by the system

GIFs work
Did U say GIF?



Weather



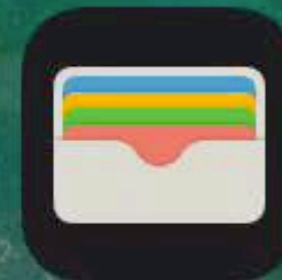
Clock



Maps



News



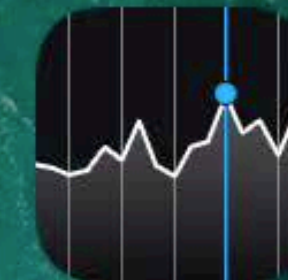
Wallet



Notes



Reminders



Stocks



Videos



iBooks



iTunes Store



App Store



Home



Health



Settings



Phone



Safari



Mail



Music

Custom User Interface

Custom User Interface

Custom User Interface

Notification content extension

Custom User Interface

Notification content extension

Custom views

Custom User Interface

Notification content extension

Custom views

No interaction

Custom User Interface

Notification content extension

Custom views

No interaction

Respond to notification actions



100%

9:41

Monday, June 13

15

CALENDAR

6/15, 9:00 AM

Order 4000 coffees

Invitation From Michele Campeotto

June 15, 2016 at 9:00 AM...

Press for more

Press home to open





100%

9:41

Monday, June 13

15

CALENDAR

6/15, 9:00 AM

Order 4000 coffees

Invitation From Michele Campeotto
June 15, 2016 at 9:00 AM...

Press for more

Press home to open



15

CALENDAR



8 AM

9 AM

10 AM

11 AM

Noon

• Order 4000 coffees
Moscone Center
747 Howard St, San Francisco, CA 94103, United States

Order 4000 coffees

Invitation From Michele Campeotto

June 15, 2016 at 9:00 AM

Moscone Center

747 Howard St, San Francisco, CA 94103,
United States

Accept

Maybe

Decline

15

CALENDAR



8 AM

9 AM

• Order 4000 coffees

Moscone Center

10 AM 747 Howard St, San Francisco, CA 94103, United States

11 AM

Noon

Order 4000 coffees

Invitation From Michele Campeotto

June 15, 2016 at 9:00 AM

Moscone Center

747 Howard St, San Francisco, CA 94103,
United States

Accept

Maybe

Decline

15

CALENDAR



8 AM

9 AM

• Order 4000 coffees

Moscone Center

10 AM 747 Howard St, San Francisco, CA 94103, United States

11 AM

Noon

Order 4000 coffees

Invitation From Michele Campeotto

June 15, 2016 at 9:00 AM

Moscone Center

747 Howard St, San Francisco, CA 94103,
United States

Accept

Maybe

Decline

15

CALENDAR



8 AM

9 AM

• Order 4000 coffees

Moscone Center

10 AM 747 Howard St, San Francisco, CA 94103, United States

11 AM

Noon

Order 4000 coffees

Invitation From Michele Campeotto

June 15, 2016 at 9:00 AM

Moscone Center

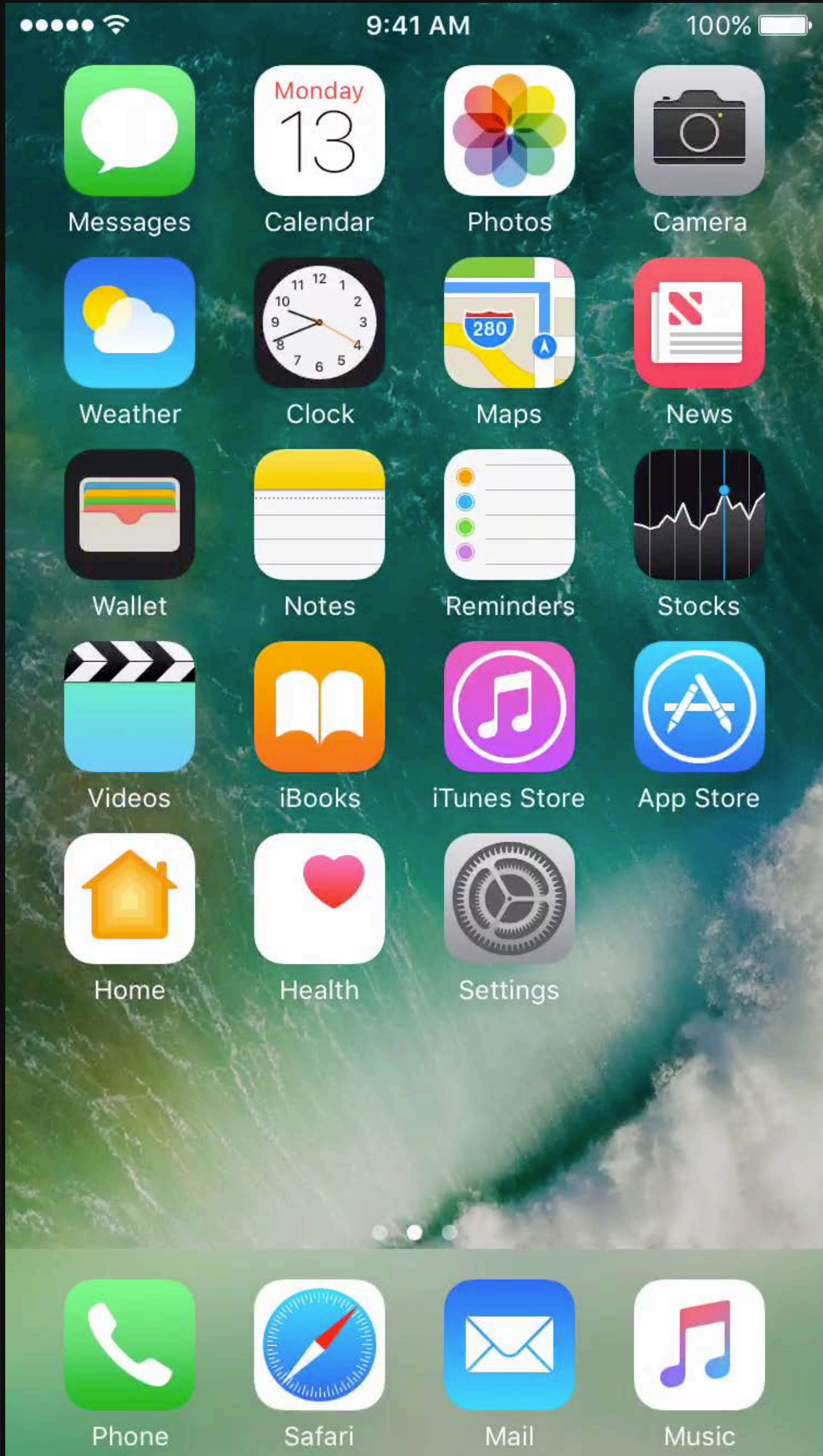
747 Howard St, San Francisco, CA 94103,
United States

Accept

Maybe

Decline

Notification Content Extension



The image shows the Xcode IDE interface. On the left is the Project Navigator showing a project named 'Demo' with files: AppDelegate.swift, ViewController.swift, Main.storyboard, Assets.xcassets, LaunchScreen.storyboard, Info.plist, and folders Frameworks and Products. In the center is the 'Choose a template for your new target' dialog. The dialog has a sidebar on the left listing platforms and target types: iOS (Application, Framework & Library, Application Extension, Test), watchOS (Application, Framework & Library), tvOS (Application, Framework & Library, Application Extension, Test), and OS X (Application, Framework & Library, Application Extension, Test). The 'Application Extension' target for iOS is selected. The main area of the dialog displays a grid of extension templates: Action Extension, Audio Unit Extension, Broadcast UI Extension, Broadcast Upload, Call Directory Extension, Content Blocker Extension, Custom Keyboard, Document Provider, Intents Extension, Intents UI Extension, Messages Extension, and Notification Content. The 'Notification Content' template is highlighted with a blue border and a blue button. Below the grid, a description reads: 'Notification Content Extension' and 'This template builds a Notification Content application extension.' At the bottom of the dialog are 'Cancel', 'Previous', and 'Next' buttons. The 'Next' button is highlighted in blue. On the right side of the image, there are several horizontal lines representing a list or table.

Choose a template for your new target:

iOS

Application

Framework & Library

Application Extension

Test

watchOS

Application

Framework & Library

tvOS

Application

Framework & Library

Application Extension

Test

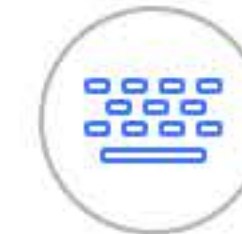
OS X

Application

Framework & Library

Application Extension

Test

Action
ExtensionAudio Unit
ExtensionBroadcast UI
ExtensionBroadcast
UploadCall Directory
ExtensionContent Blocker
ExtensionCustom
KeyboardDocument
ProviderIntents
ExtensionIntents UI
ExtensionMessages
ExtensionNotification
Content

Notification Content Extension

This template builds a Notification Content application extension.

Cancel

Previous

Next



Demo > EventNotification > NotificationViewController.swift > No Selection

- ▼ Demo M
 - ▼ Demo
 - AppDelegate.swift
 - ViewController.swift
 - Main.storyboard
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - ▼ EventNotification
 - NotificationViewController.swift**
 - MainInterface.storyboard
 - Info.plist
 - ▶ Frameworks
 - ▶ Products

```
1  //
2  // NotificationViewController.swift
3  // EventNotification
4  //
5  // Created by Michele Campeotto on 6/1/16.
6  // Copyright © 2016 Apple. All rights reserved.
7  //
8
9  import UIKit
10 import UserNotifications
11 import UserNotificationsUI
12
13 class NotificationViewController: UIViewController, UNNotificationContentExtension {
14
15     @IBOutlet var label: UILabel?
16
17     override func viewDidLoad() {
18         super.viewDidLoad()
19         // Do any required interface initialization here.
20     }
21
22     func didReceive(_ notification: UNNotification) {
23         self.label?.text = notification.request.content.body
24     }
25
26 }
27
```



```
// Minimal Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
  
    @IBOutlet var label: UILabel?  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
        // Do any required interface initialization here.  
    }  
  
    func didReceive(_ notification: UNNotification) {  
        label?.text = notification.request.content.body  
    }  
  
}
```

```
// Minimal Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    @IBOutlet var label: UILabel?
```

```
    override func viewDidLoad() {
```

```
        super.viewDidLoad()
```

```
        // Do any required interface initialization here.
```

```
    }
```

```
    func didReceive(_ notification: UNNotification) {
```

```
        label?.text = notification.request.content.body
```

```
    }
```

```
}
```

```
// Minimal Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    @IBOutlet var label: UILabel?
```

```
    override func viewDidLoad() {
```

```
        super.viewDidLoad()
```

```
        // Do any required interface initialization here.
```

```
    }
```

```
    func didReceive(_ notification: UNNotification) {
```

```
        label?.text = notification.request.content.body
```

```
    }
```

```
}
```


Key	Type	Value
▼ Information Property List	Dictionary	(11 items)
Localization native development re...	String	en
Bundle display name	String	EventNotification
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	XPC!
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1
▼ NSExtension	Dictionary	(3 items)
▼ NSExtensionAttributes	Dictionary	(1 item)
UNNotificationExtensionCategory	String	event-invite
NSExtensionMainStoryboard	String	MainInterface
NSExtensionPointIdentifier	String	com.apple.usernotifications.content-extension

Key	Type	Value
▼ Information Property List	Dictionary	(11 items)
Localization native development re...	String	en
Bundle display name	String	EventNotification
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	XPC!
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1
▼ NSExtension	Dictionary	(3 items)
▼ NSExtensionAttributes	Dictionary	(1 item)
UNNotificationExtensionCategory	String	event-invite
NSExtensionMainStoryboard	String	MainInterface
NSExtensionPointIdentifier	String	com.apple.usernotifications.content-extension

Key	Type	Value
▼ Information Property List	Dictionary	(11 items)
Localization native development re...	String	en
Bundle display name	String	EventNotification
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	XPC!
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1
▼ NSExtension	Dictionary	(3 items)
▼ NSExtensionAttributes	Dictionary	(1 item)
▼ UNNotificationExtensionCategory	Array	(2 items)
Item 0	String	event-invite
Item 1	String	event-update
NSExtensionMainStoryboard	String	MainInterface
NSExtensionPointIdentifier	String	com.apple.usernotifications.content-extension

Custom Views

```
// Notification Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    @IBOutlet var eventTitle: UILabel!
```

```
    @IBOutlet var eventDate: UILabel!
```

```
    @IBOutlet var eventLocation: UILabel!
```

```
    @IBOutlet var eventMessage: UILabel!
```

```
    func didReceive(_ notification: UNNotification) {
```

```
        let content = notification.request.content
```

```
        eventTitle.text = content.title
```

```
        eventDate.text = content.subtitle
```

```
        eventMessage.text = content.body
```

```
        if let location = content.userInfo["location"] as? String {
```

```
            eventLocation.text = location
```

```
        }
```

```
    }
```

```
}
```

```
// Notification Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    @IBOutlet var eventTitle: UILabel!  
    @IBOutlet var eventDate: UILabel!  
    @IBOutlet var eventLocation: UILabel!  
    @IBOutlet var eventMessage: UILabel!
```

```
    func didReceive(_ notification: UNNotification) {  
        let content = notification.request.content
```

```
        eventTitle.text = content.title  
        eventDate.text = content.subtitle  
        eventMessage.text = content.body
```

```
        if let location = content.userInfo["location"] as? String {  
            eventLocation.text = location  
        }
```

```
    }
```

```
}
```



```
// Notification Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    @IBOutlet var eventTitle: UILabel!
```

```
    @IBOutlet var eventDate: UILabel!
```

```
    @IBOutlet var eventLocation: UILabel!
```

```
    @IBOutlet var eventMessage: UILabel!
```

```
    func didReceive(_ notification: UNNotification) {
```

```
        let content = notification.request.content
```

```
        eventTitle.text = content.title
```

```
        eventDate.text = content.subtitle
```

```
        eventMessage.text = content.body
```

```
        if let location = content.userInfo["location"] as? String {
```

```
            eventLocation.text = location
```

```
        }
```

```
    }
```

```
}
```

```
// Notification Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    @IBOutlet var eventTitle: UILabel!
```

```
    @IBOutlet var eventDate: UILabel!
```

```
    @IBOutlet var eventLocation: UILabel!
```

```
    @IBOutlet var eventMessage: UILabel!
```

```
    func didReceive(_ notification: UNNotification) {
```

```
        let content = notification.request.content
```

```
        eventTitle.text = content.title
```

```
        eventDate.text = content.subtitle
```

```
        eventMessage.text = content.body
```

```
        if let location = content.userInfo["location"] as? String {
```

```
            eventLocation.text = location
```

```
        }
```

```
    }
```

```
}
```

```
// Notification Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    @IBOutlet var eventTitle: UILabel!
```

```
    @IBOutlet var eventDate: UILabel!
```

```
    @IBOutlet var eventLocation: UILabel!
```

```
    @IBOutlet var eventMessage: UILabel!
```

```
    func didReceive(_ notification: UNNotification) {
```

```
        let content = notification.request.content
```

```
        eventTitle.text = content.title
```

```
        eventDate.text = content.subtitle
```

```
        eventMessage.text = content.body
```

```
        if let location = content.userInfo["location"] as? String {
```

```
            eventLocation.text = location
```

```
        }
```

```
    }
```

```
}
```






DEMO LEMURS



The Bash

Thursday, June 16, 7pm

Bill Graham Civic

Have you heard the rumors? I can't believe who's playing this year!

The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!

Accept

Decline



DEMO LEMURS



The Bash

Thursday, June 16, 7pm

Bill Graham Civic

Have you heard the rumors? I can't believe who's playing this year!


The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!

Accept

Decline

 DEMO LEMURS



The Bash

Thursday, June 16, 7pm

Bill Graham Civic

Have you heard the rumors? I can't believe who's playing this year!

The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!

Accept

Decline

Key	Type	Value
▼ Information Property List	Dictionary	(11 items)
Localization native development region	String	en
Bundle display name	String	EventNotification
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	XPC!
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1
▼ NSExtension	Dictionary	(3 items)
▼ NSExtensionAttributes	Dictionary	(2 items)
UNNotificationExtensionDefaultContentHidden	Boolean	YES
UNNotificationExtensionCategory	String	event-invite
NSExtensionMainStoryboard	String	MainInterface
NSExtensionPointIdentifier	String	com.apple.usernotifications.content-extension

Key	Type	Value
▼ Information Property List	Dictionary	(11 items)
Localization native development region	String	en
Bundle display name	String	EventNotification
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	XPC!
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1
▼ NSExtension	Dictionary	(3 items)
▼ NSExtensionAttributes	Dictionary	(2 items)
UNNotificationExtensionDefaultContentHidden	Boolean	YES
UNNotificationExtensionCategory	String	event-invite
NSExtensionMainStoryboard	String	MainInterface
NSExtensionPointIdentifier	String	com.apple.usernotifications.content-extension


```
// Notification Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
  
    override func viewDidLoad() {  
        super.viewDidLoad()  
  
        let size = view.bounds.size  
        preferredContentSize = CGSize(width: size.width, height: size.width / 2)  
    }  
  
    func didReceive(_ notification: UNNotification) {  
        // ...  
    }  
  
}
```

```
// Notification Content Extension
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    override func viewDidLoad() {  
        super.viewDidLoad()
```

```
        let size = view.bounds.size  
        preferredContentSize = CGSize(width: size.width, height: size.width / 2)
```

```
    }
```

```
    func didReceive(_ notification: UNNotification) {
```

```
        // ...
```

```
    }
```

```
}
```

Z DEMO LEMURS

now

The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!



Weather



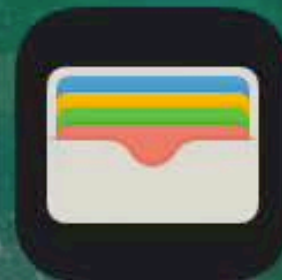
Clock



Maps



News



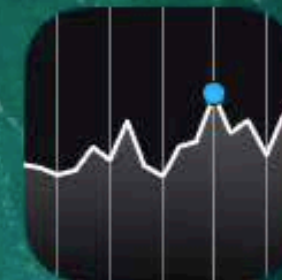
Wallet



Notes



Reminders



Stocks



Videos



iBooks



iTunes Store



App Store



Home



Health



Settings



Phone



Safari



Mail



Music

Key	Type	Value
▼ Information Property List	Dictionary	(11 items)
Localization native development region	String	en
Bundle display name	String	EventNotification
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	XPC!
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1
▼ NSExtension	Dictionary	(3 items)
▼ NSExtensionAttributes	Dictionary	(3 items)
UNNotificationExtensionInitialContentSizeRatio	Number	0.5
UNNotificationExtensionDefaultContentHidden	Boolean	YES
UNNotificationExtensionCategory	String	event-invite
NSExtensionMainStoryboard	String	MainInterface
NSExtensionPointIdentifier	String	com.apple.usernotifications.content-extension

Key	Type	Value
▼ Information Property List	Dictionary	(11 items)
Localization native development region	String	en
Bundle display name	String	EventNotification
Executable file	String	\$(EXECUTABLE_NAME)
Bundle identifier	String	\$(PRODUCT_BUNDLE_IDENTIFIER)
InfoDictionary version	String	6.0
Bundle name	String	\$(PRODUCT_NAME)
Bundle OS Type code	String	XPC!
Bundle versions string, short	String	1.0
Bundle creator OS Type code	String	????
Bundle version	String	1
▼ NSExtension	Dictionary	(3 items)
▼ NSExtensionAttributes	Dictionary	(3 items)
UNNotificationExtensionInitialContentSizeRatio	Number	0.5
UNNotificationExtensionDefaultContentHidden	Boolean	YES
UNNotificationExtensionCategory	String	event-invite
NSExtensionMainStoryboard	String	MainInterface
NSExtensionPointIdentifier	String	com.apple.usernotifications.content-extension

Z DEMO LEMURS

now

The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!



Weather



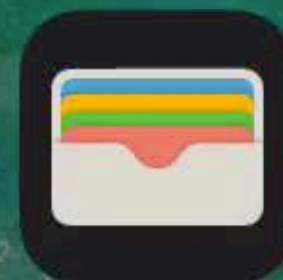
Clock



Maps



News



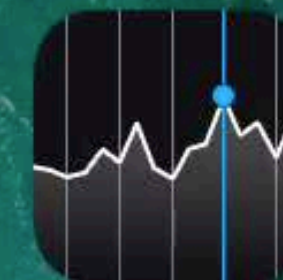
Wallet



Notes



Reminders



Stocks



Videos



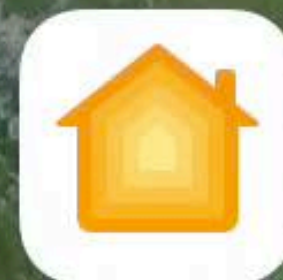
iBooks



iTunes Store



App Store



Home



Health



Settings



Phone



Safari

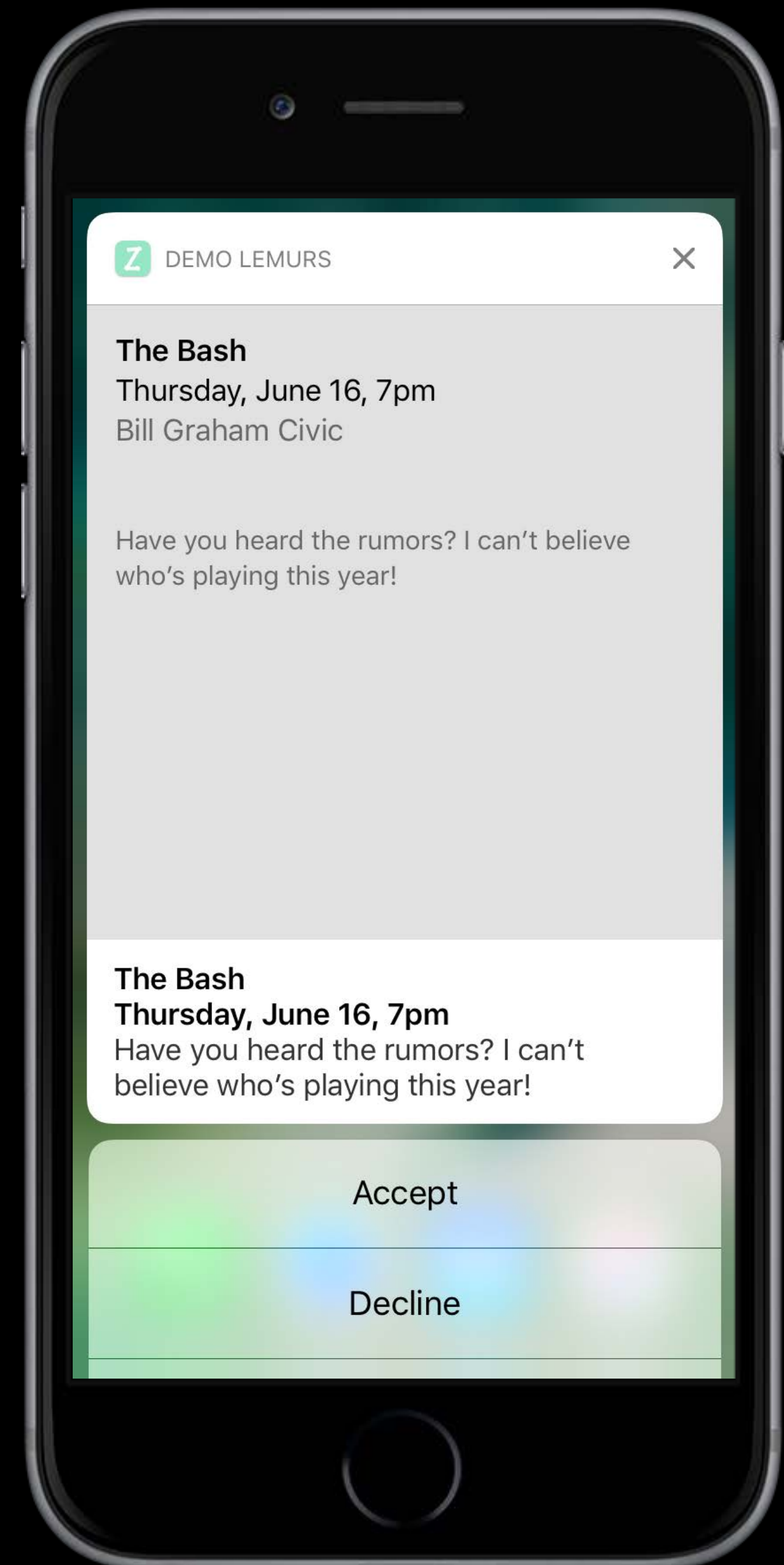


Mail



Music

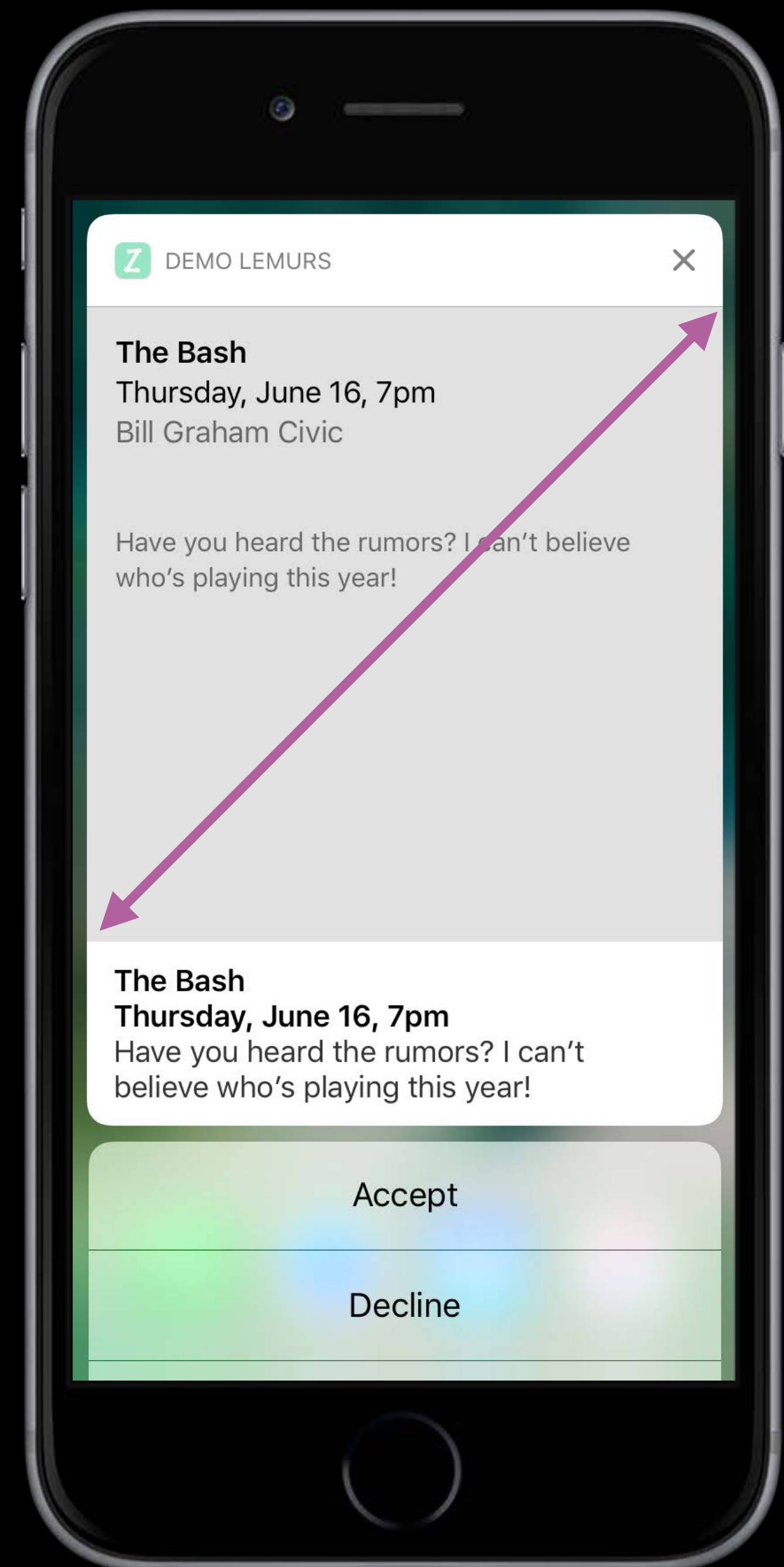
Custom Notification UI



Custom Notification UI

Presentation size

`UNNotificationExtensionInitialContentSizeRatio`



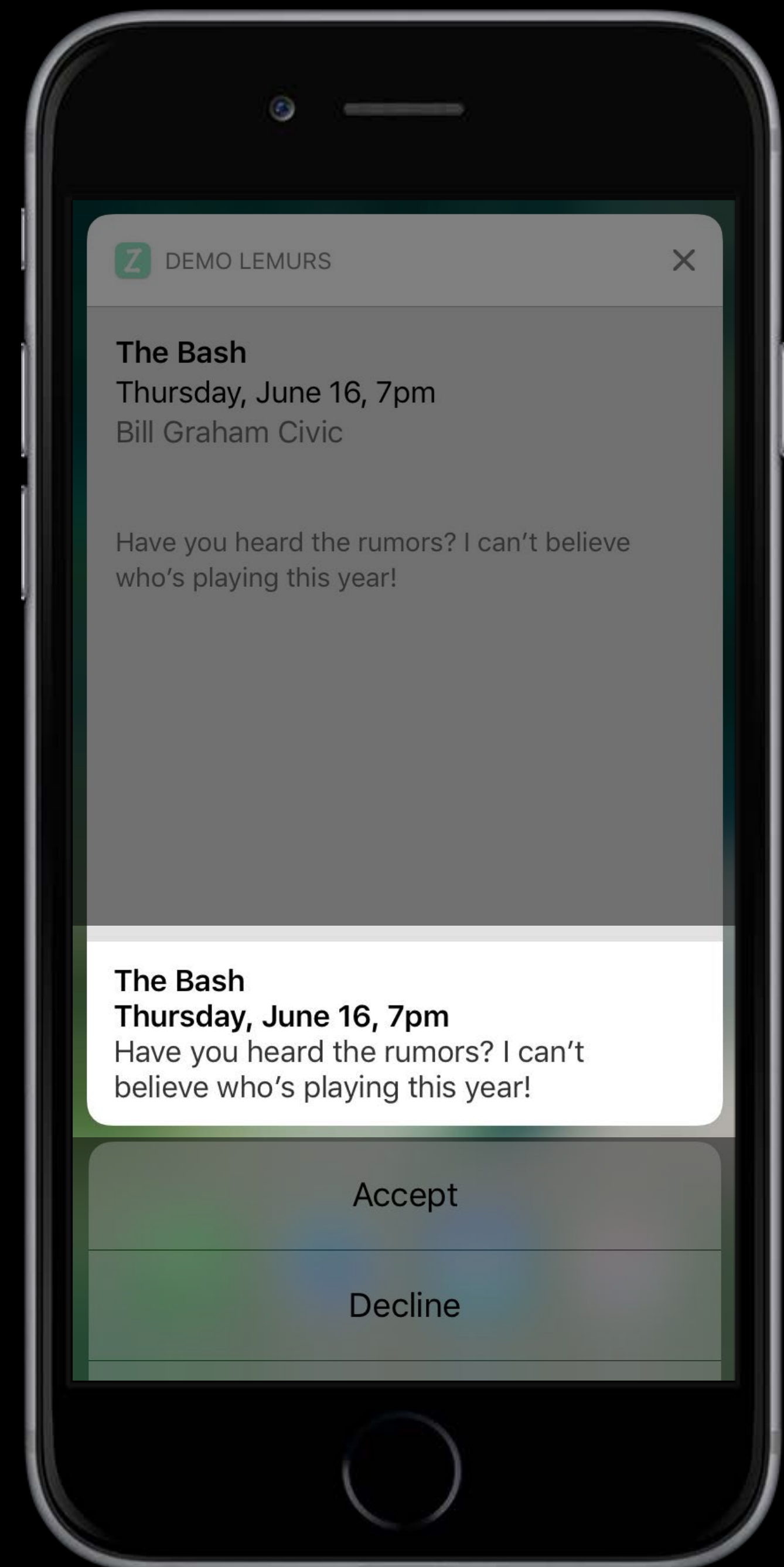
Custom Notification UI

Presentation size

`UNNotificationExtensionInitialContentSizeRatio`

Default content

`UNNotificationExtensionDefaultContentHidden`



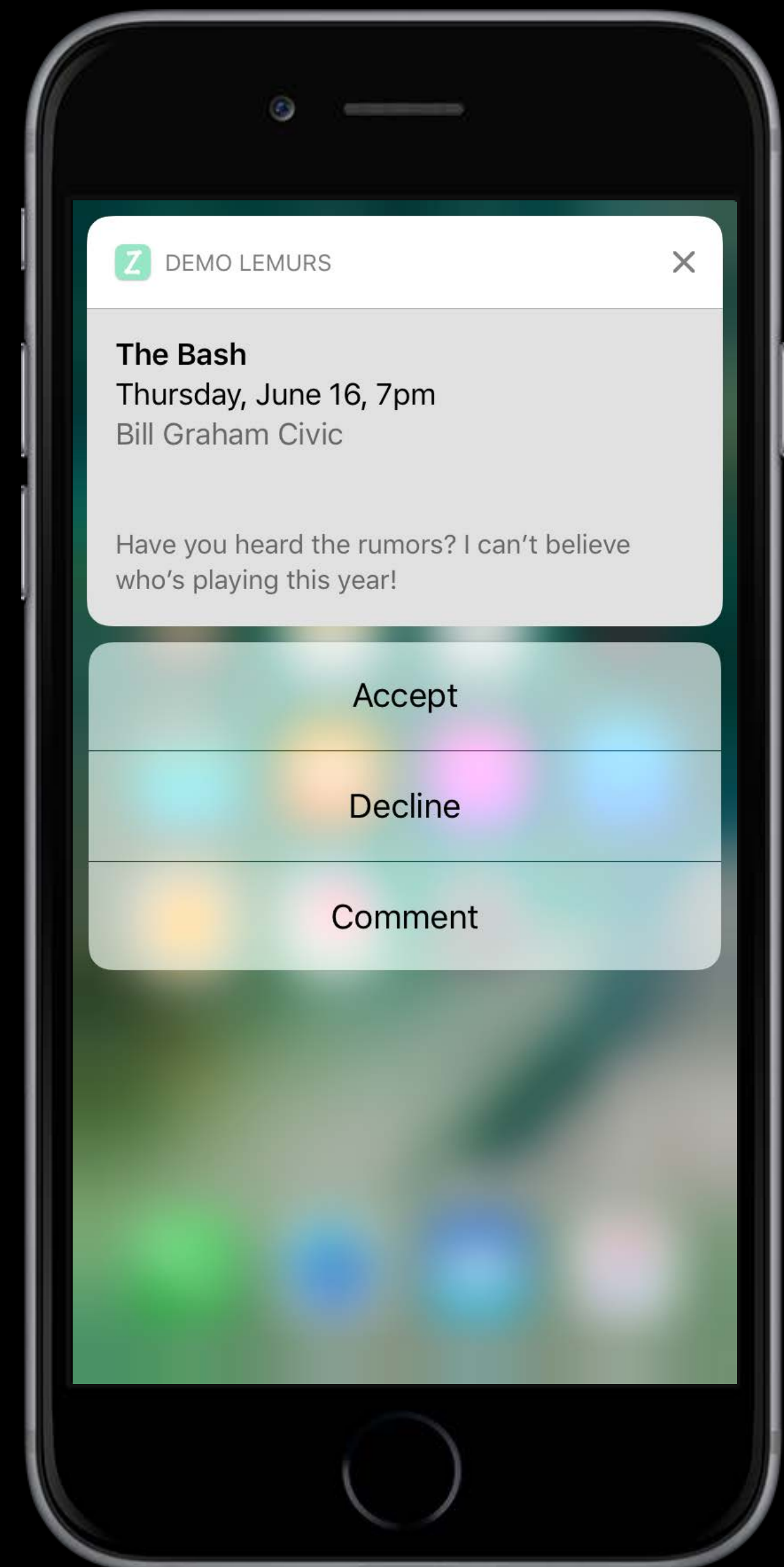
Custom Notification UI

Presentation size

`UNNotificationExtensionInitialContentSizeRatio`

Default content

`UNNotificationExtensionDefaultContentHidden`



Media Attachments

```
// Notification Content Extension Attachments
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
  
    @IBOutlet var eventImage: UIImageView!  
  
    func didReceive(_ notification: UNNotification) {  
        let content = notification.request.content  
  
        if let attachment = content.attachments.first {  
            if attachment.url.startAccessingSecurityScopedResource() {  
                eventImage.image = UIImage(contentsOfFile: attachment.url.path!)  
                attachment.url.stopAccessingSecurityScopedResource()  
            }  
        }  
    }  
}
```



```
// Notification Content Extension Attachments
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {

    @IBOutlet var eventImage: UIImageView!

    func didReceive(_ notification: UNNotification) {
        let content = notification.request.content

        if let attachment = content.attachments.first {
            if attachment.url.startAccessingSecurityScopedResource() {
                eventImage.image = UIImage(contentsOfFile: attachment.url.path!)
                attachment.url.stopAccessingSecurityScopedResource()
            }
        }
    }
}
```

```
// Notification Content Extension Attachments
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {

    @IBOutlet var eventImage: UIImageView!

    func didReceive(_ notification: UNNotification) {
        let content = notification.request.content

        if let attachment = content.attachments.first {
            if attachment.url.startAccessingSecurityScopedResource() {
                eventImage.image = UIImage(contentsOfFile: attachment.url.path!)
                attachment.url.stopAccessingSecurityScopedResource()
            }
        }
    }
}
```

```
// Notification Content Extension Attachments
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {

    @IBOutlet var eventImage: UIImageView!

    func didReceive(_ notification: UNNotification) {
        let content = notification.request.content

        if let attachment = content.attachments.first {
            if attachment.url.startAccessingSecurityScopedResource() {
                eventImage.image = UIImage(contentsOfFile: attachment.url.path!)
                attachment.url.stopAccessingSecurityScopedResource()
            }
        }
    }
}
```


Z DEMO LEMURS

now

The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!



Weather



Clock



Maps



News



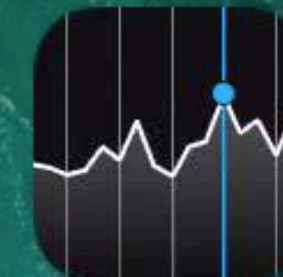
Wallet



Notes



Reminders



Stocks



Videos



iBooks



iTunes Store



App Store



Home



Health



Settings



Phone



Safari



Mail



Music

Z DEMO LEMURS

now

The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!



Weather



Clock



Maps



News



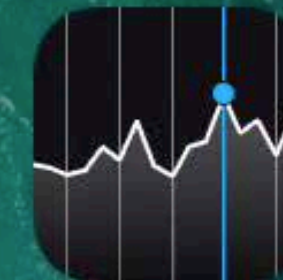
Wallet



Notes



Reminders



Stocks



Videos



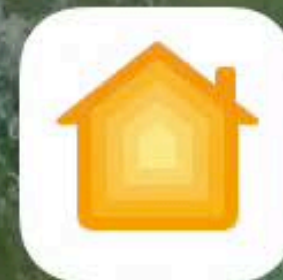
iBooks



iTunes Store



App Store



Home



Health



Settings



Phone



Safari



Mail



Music

Actions

Default Action Handling

Default Action Handling

Delivered to the app

Notification gets dismissed immediately

Intercepting Action Response

Intercepting Action Response

Delivered to the extension

Can delay dismissal

```
// Intercepting notification action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {

    func didReceive(_ response: UNNotificationResponse,
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {

        server.postEventResponse(response.actionIdentifier) {
            if response.actionIdentifier == "accept" {
                eventResponse.text = "Going!"
                eventResponse.textColor = UIColor.green()
            } else if response.actionIdentifier == "decline" {
                eventResponse.text = "Not going :("
                eventResponse.textColor = UIColor.red()
            }
        }

        done(.dismiss)
    }
}
```

```
// Intercepting notification action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    func didReceive(_ response: UNNotificationResponse,
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {
```

```
        server.postEventResponse(response.actionIdentifier) {
            if response.actionIdentifier == "accept" {
                eventResponse.text = "Going!"
                eventResponse.textColor = UIColor.green()
            } else if response.actionIdentifier == "decline" {
                eventResponse.text = "Not going :("
                eventResponse.textColor = UIColor.red()
            }
        }
```

```
        done(.dismiss)
```

```
    }
```

```
}
```

```
}
```



```
// Intercepting notification action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    func didReceive(_ response: UNNotificationResponse,
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {
```

```
        server.postEventResponse(response.actionIdentifier) {
```

```
            if response.actionIdentifier == "accept" {
```

```
                eventResponse.text = "Going!"
```

```
                eventResponse.textColor = UIColor.green()
```

```
            } else if response.actionIdentifier == "decline" {
```

```
                eventResponse.text = "Not going :("
```

```
                eventResponse.textColor = UIColor.red()
```

```
            }
```

```
            done(.dismiss)
```

```
        }
```

```
    }
```

```
}
```

```
// Intercepting notification action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {

    func didReceive(_ response: UNNotificationResponse,
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {

        server.postEventResponse(response.actionIdentifier) {

            if response.actionIdentifier == "accept" {
                eventResponse.text = "Going!"
                eventResponse.textColor = UIColor.green()
            } else if response.actionIdentifier == "decline" {
                eventResponse.text = "Not going :("
                eventResponse.textColor = UIColor.red()
            }

            done(.dismiss)
        }
    }
}
```

```
// Intercepting notification action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {

    func didReceive(_ response: UNNotificationResponse,
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {

        server.postEventResponse(response.actionIdentifier) {
            if response.actionIdentifier == "accept" {
                eventResponse.text = "Going!"
                eventResponse.textColor = UIColor.green()
            } else if response.actionIdentifier == "decline" {
                eventResponse.text = "Not going :("
                eventResponse.textColor = UIColor.red()
            }
        }

        done(.dismiss)
    }
}
```



```
// Intercepting notification action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {

    func didReceive(_ response: UNNotificationResponse,
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {

        server.postEventResponse(response.actionIdentifier) {
            if response.actionIdentifier == "accept" {
                eventResponse.text = "Going!"
                eventResponse.textColor = UIColor.green()
            } else if response.actionIdentifier == "decline" {
                eventResponse.text = "Not going :("
                eventResponse.textColor = UIColor.red()
            }
        }

        done(.dismissAndForwardAction)
    }
}
```



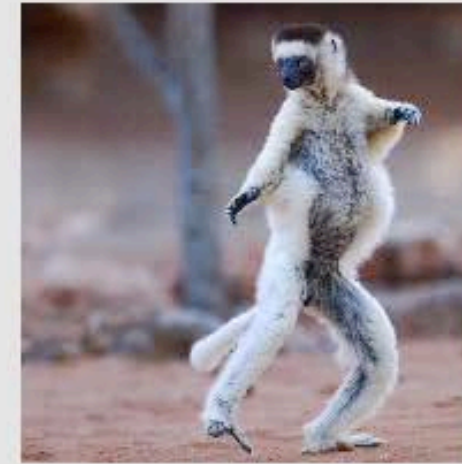
DEMO LEMURS



The Bash

Thursday, June 16, 7pm

Bill Graham Civic



Have you heard the rumors? I can't believe who's playing this year!

Accept

Decline

Comment

Text Input Action


```
// Text Input Action
```

```
private fun makeEventExtensionCategory() -> UNNotificationCategory {  
  
    let commentAction = UNTextInputNotificationAction(  
        identifier: "comment",  
        title: "Comment",  
        options: [],  
        textInputButtonTitle: "Send",  
        textInputPlaceholder: "Type here...")  
  
    return UNNotificationCategory(identifier: "event-invite",  
                                   actions: [ acceptAction, declineAction, commentAction ],  
                                   minimalActions: [ acceptAction, declineAction ],  
                                   intentIdentifiers: [],  
                                   options: [])  
}
```

```
// Text Input Action
```

```
private fun makeEventExtensionCategory() -> UNNotificationCategory {
```

```
    let commentAction = UNTextInputNotificationAction(
```

```
        identifier: "comment",
```

```
        title: "Comment",
```

```
        options: [],
```

```
        textInputButtonTitle: "Send",
```

```
        textInputPlaceholder: "Type here...")
```

```
    return UNNotificationCategory(identifier: "event-invite",
```

```
        actions: [ acceptAction, declineAction, commentAction ],
```

```
        minimalActions: [ acceptAction, declineAction ],
```

```
        intentIdentifiers: [],
```

```
        options: [])
```

```
}
```

```
// Text Input Action
```

```
private fun makeEventExtensionCategory() -> UNNotificationCategory {
```

```
    let commentAction = UNTextInputNotificationAction(
```

```
        identifier: "comment",
```

```
        title: "Comment",
```

```
        options: [],
```

```
        textInputButtonTitle: "Send",
```

```
        textInputPlaceholder: "Type here...")
```

```
    return UNNotificationCategory(identifier: "event-invite",
```

```
        actions: [ acceptAction, declineAction, commentAction ],
```

```
        minimalActions: [ acceptAction, declineAction ],
```

```
        intentIdentifiers: [],
```

```
        options: [])
```

```
}
```



```
// Text Input Action
```

```
private fun makeEventExtensionCategory() -> UNNotificationCategory {
```

```
    let commentAction = UNTextInputNotificationAction(
        identifier: "comment",
        title: "Comment",
        options: [],
        textInputButtonTitle: "Send",
        textInputPlaceholder: "Type here...")
```

```
    return UNNotificationCategory(identifier: "event-invite",
                                   actions: [ acceptAction, declineAction, commentAction ],
                                   minimalActions: [ acceptAction, declineAction ],
                                   intentIdentifiers: [],
                                   options: [])
```

```
}
```

Z DEMO LEMURS

now

The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!



Weather



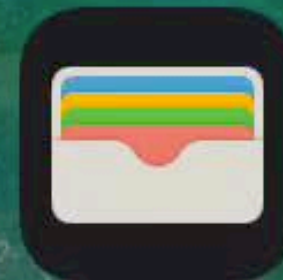
Clock



Maps



News



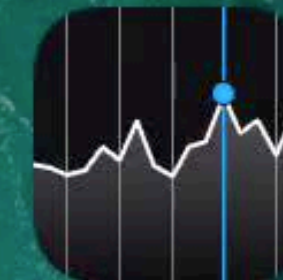
Wallet



Notes



Reminders



Stocks



Videos



iBooks



iTunes Store



App Store



Home



Health



Settings



Phone



Safari



Mail



Music


```
// Text input action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
  
    func didReceive(_ response: UNNotificationResponse,  
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {  
  
        if let textResponse = response as? UNTextInputNotificationResponse {  
            server.send(textResponse.userText) {  
                done(.dismiss)  
            }  
        }  
    }  
}
```



```
// Text input action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {
```

```
    func didReceive(_ response: UNNotificationResponse,  
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {
```

```
        if let textResponse = response as? UNTextInputNotificationResponse {  
            server.send(textResponse.userText) {  
                done(.dismiss)  
            }  
        }  
    }
```

```
}
```

```
}
```

```
// Text input action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
  
    func didReceive(_ response: UNNotificationResponse,  
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {  
  
        if let textResponse = response as? UNTextInputNotificationResponse {  
            server.send(textResponse.userText) {  
                done(.dismiss)  
            }  
        }  
    }  
}
```

```
// Text input action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
  
    func didReceive(_ response: UNNotificationResponse,  
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {  
  
        if let textResponse = response as? UNTextInputNotificationResponse {  
            server.send(textResponse.userText) {  
                done(.dismiss)  
            }  
        }  
    }  
}
```



```
// Text input action response
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
  
    func didReceive(_ response: UNNotificationResponse,  
                    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {  
  
        if let textResponse = response as? UNTextInputNotificationResponse {  
            server.send(textResponse.userText) {  
                done(.dismiss)  
            }  
        }  
    }  
}
```



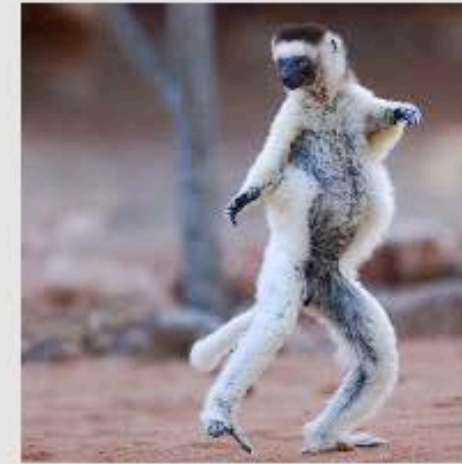
DEMO LEMURS



The Bash

Thursday, June 16, 7pm

Bill Graham Civic



Have you heard the rumors? I can't believe who's playing this year!

Accept

Decline

Comment

The Bash

Thursday, June 16, 7pm

Bill Graham Civic



Have you heard the rumors? I can't believe who's playing this year!

Yay!



The

I

Oops

Q

W

E

R

T

Y

U

I

O

P

A

S

D

F

G

H

J

K

L



Z

X

C

V

B

N

M



123



space

Done

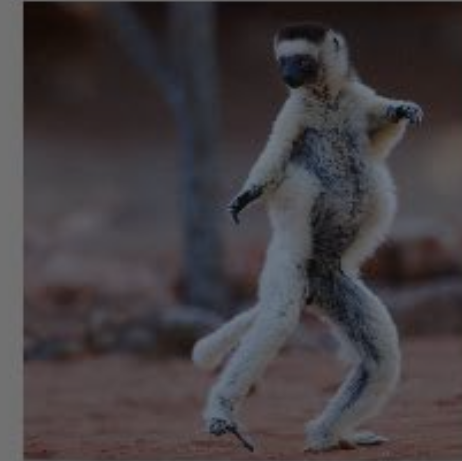
Z DEMO LEMURS



The Bash

Thursday, June 16, 7pm

Bill Graham Civic



Have you heard the rumors? I can't believe who's playing this year!

Yay!



The

I

Oops

Q

W

E

R

T

Y

U

I

O

P

A

S

D

F

G

H

J

K

L



Z

X

C

V

B

N

M



123



space

Done

```
// Custom input accessory view
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
    override func canBecomeFirstResponder() -> Bool {  
        return true  
    }  
  
    override var inputAccessoryView: UIView { get {  
        return inputView  
    }  
}  
  
    func didReceive(_ response: UNNotificationResponse,  
        completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {  
        if response.actionIdentifier == "comment" {  
            becomeFirstResponder()  
            textField.becomeFirstResponder()  
        }  
    }  
}
```

```
// Custom input accessory view
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
    override func canBecomeFirstResponder() -> Bool {  
        return true  
    }  
}
```

```
    override var inputAccessoryView: UIView { get {  
        return inputView  
    }  
}
```

```
func didReceive(_ response: UNNotificationResponse,  
    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {  
    if response.actionIdentifier == "comment" {  
        becomeFirstResponder()  
        textField.becomeFirstResponder()  
    }  
}
```



```
// Custom input accessory view
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
    override func canBecomeFirstResponder() -> Bool {  
        return true  
    }  
}
```

```
    override var inputAccessoryView: UIView { get {  
        return inputView  
    }  
}
```

```
func didReceive(_ response: UNNotificationResponse,  
    completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {  
    if response.actionIdentifier == "comment" {  
        becomeFirstResponder()  
        textField.becomeFirstResponder()  
    }  
}  
}
```

```
// Custom input accessory view
```

```
class NotificationViewController: UIViewController, UNNotificationContentExtension {  
    override func canBecomeFirstResponder() -> Bool {  
        return true  
    }  
}
```

```
    override var inputAccessoryView: UIView { get {  
        return inputView  
    }  
}
```

```
    func didReceive(_ response: UNNotificationResponse,  
        completionHandler done: (UNNotificationContentExtensionResponseOption) -> Void) {  
        if response.actionIdentifier == "comment" {  
            becomeFirstResponder()  
            textField.becomeFirstResponder()  
        }  
    }  
}
```

```
}
```


Z DEMO LEMURS

now

The Bash

Thursday, June 16, 7pm

Have you heard the rumors? I can't believe who's playing this year!



Weather



Clock



Maps



News



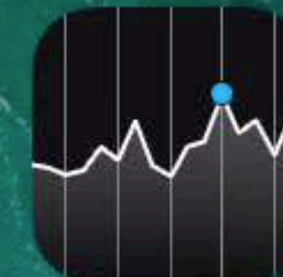
Wallet



Notes



Reminders



Stocks



Videos



iBooks



iTunes Store



App Store



Home



Health



Settings



Phone



Safari



Mail



Music

Summary

Attachments and custom UI

Attachments with service extension

Custom UI with content extension

- Media attachments
- User interaction

More Information

<https://developer.apple.com/wwdc16/708>

Related Sessions

Introduction to Notifications	Pacific Heights	Wednesday 9:00AM
-------------------------------	-----------------	------------------

Labs

Notifications Lab	Frameworks Lab C	Wednesday 11:00AM
Notifications Lab	Graphics, Games, and Media Lab B	Friday 9:00AM

