

#WWDC19

Visual Design Accessibility

Sommer Panage, Accessibility Engineering Manager

Dynamic Type

Reduce Motion

Differentiate Without Color

Dynamic Type

Reduce Motion

Differentiate Without Color

9:41



< Library



LOVE + FEAR

MARINA

Pop • 2019



▶ Play

↻ Shuffle

★ 1 Handmade Heaven



★ 2 Superstar

★ 3 Orange Trees



4 Baby (feat. MARINA & Luis Fonsi)

Clean Bandit



★ 5 Enjoy Your Life



6 True



★ To Be Human



To Be Human



Library



For You



Browse



Radio



Search

9:41



< Library



LOVE + FEAR
MARINA
Pop • 2019



▶ Play

↻ Shuffle

★ 1 Handmade Heaven



★ 2 Superstar

★ 3 Orange Trees



4 Baby (feat. MARINA & Luis Fonsi)
Clean Bandit



★ 5 Enjoy Your Life



6 True



★ ... To Be Human



8 End Of The Earth



To Be Human



Library



For You



Browse



Radio



Search

9:41



< Library



LOVE + FEAR

MARINA

Pop • 2019



▶ Play

↻ Shuffle

★1 Handmade Heaven



To Be Human



Library



For You



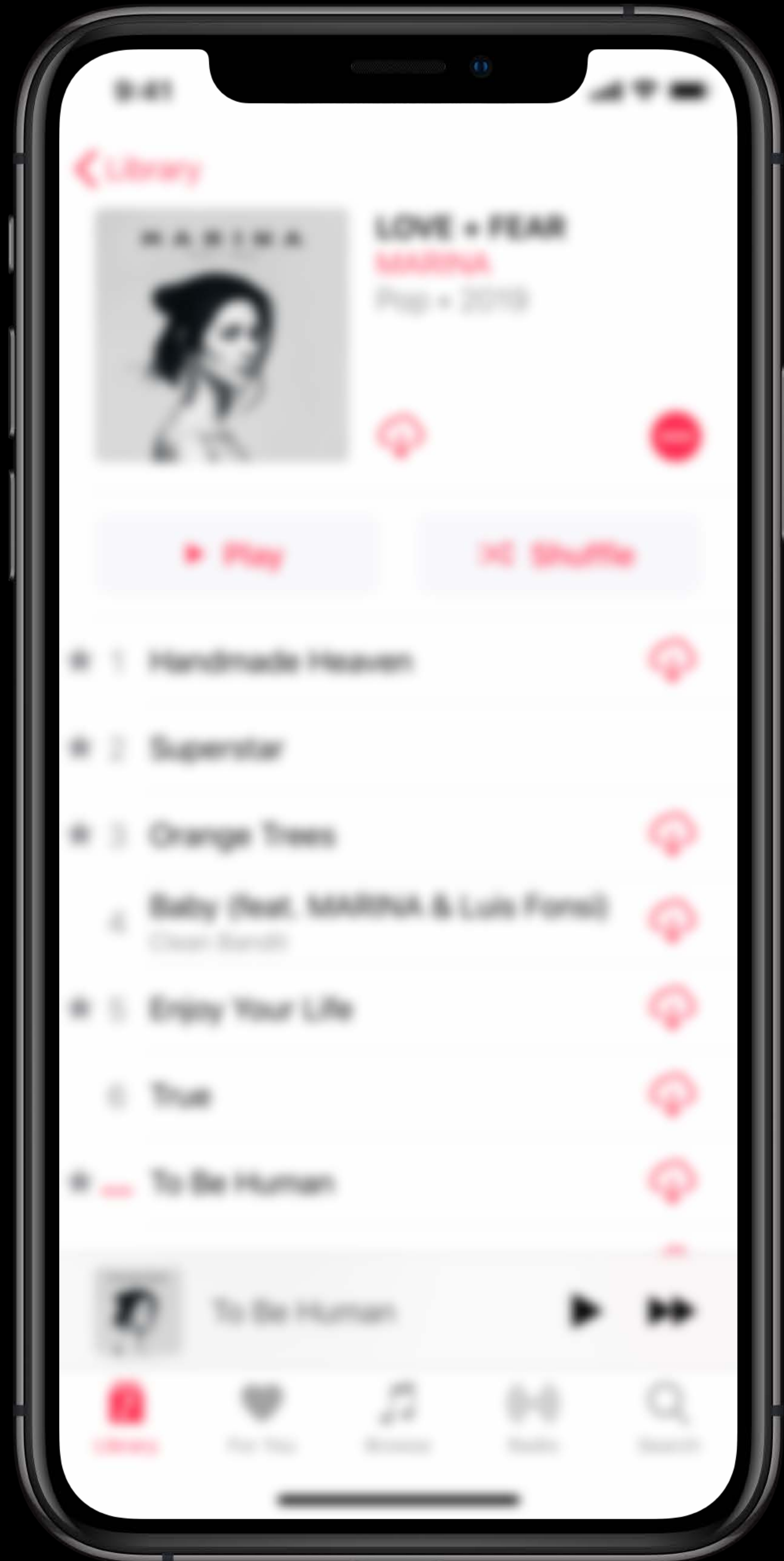
Browse



Radio



Search



← Library



LOVE & FEAR

Marina

Pop • 2019



▶ Play

🔀 Shuffle

- 1 Handmade Heaven 
- 2 Superstar 
- 3 Orange Trees 
- 4 Baby (feat. Khalid & Luke Ford) 
feat. Khalid & Luke Ford
- 5 Enjoy Your Life 
- 6 The 
- 7 To Be Human 

 To Be Human ▶ 

< Library



LOVE + FEAR
MARINA
Pop • 2019



▶ Play

⌵ Shuffle

🎵 Handmade Heaven



To Be Human

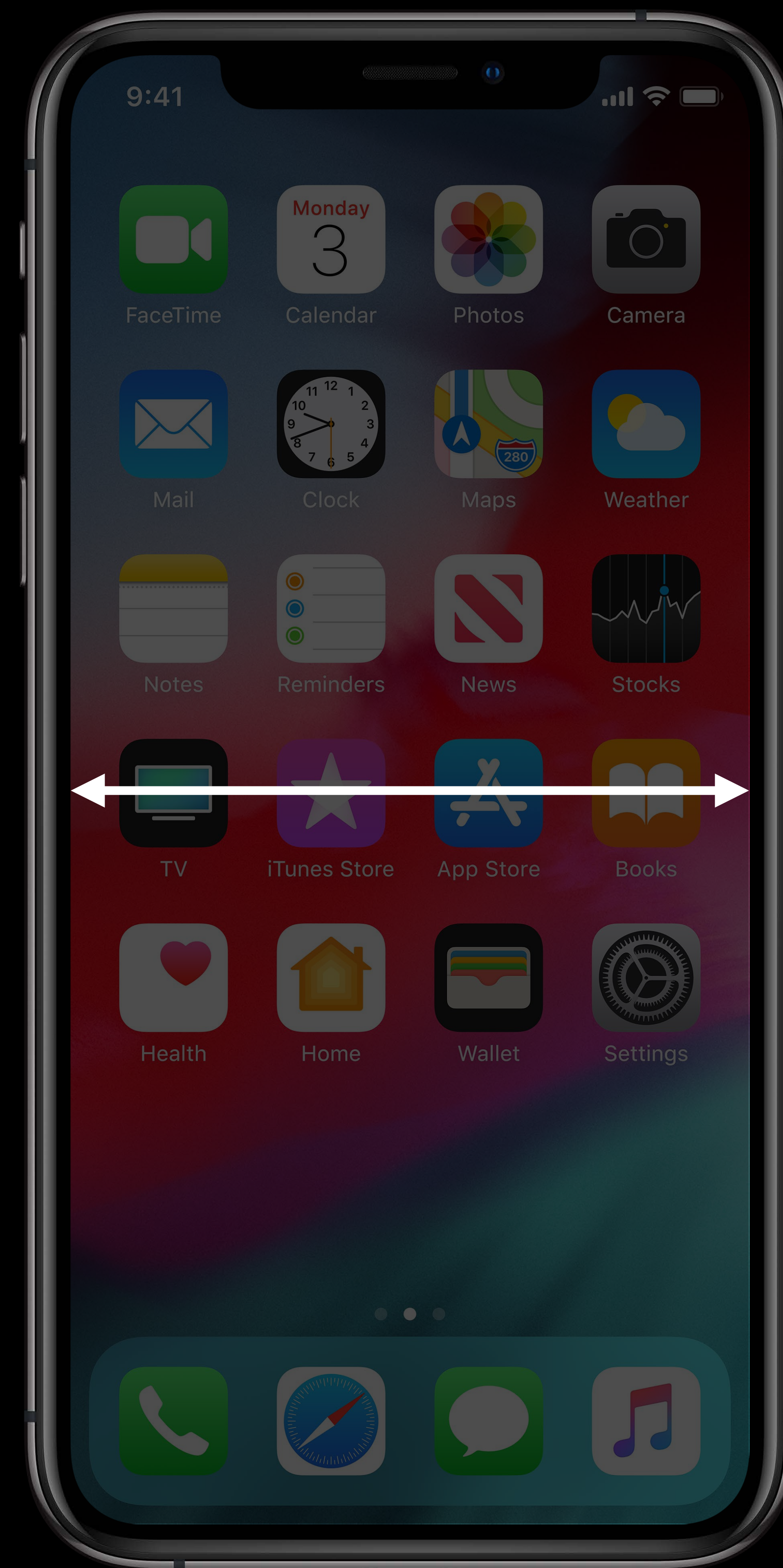


Dynamic Type Principles

If text can be dynamic,
it should be dynamic.

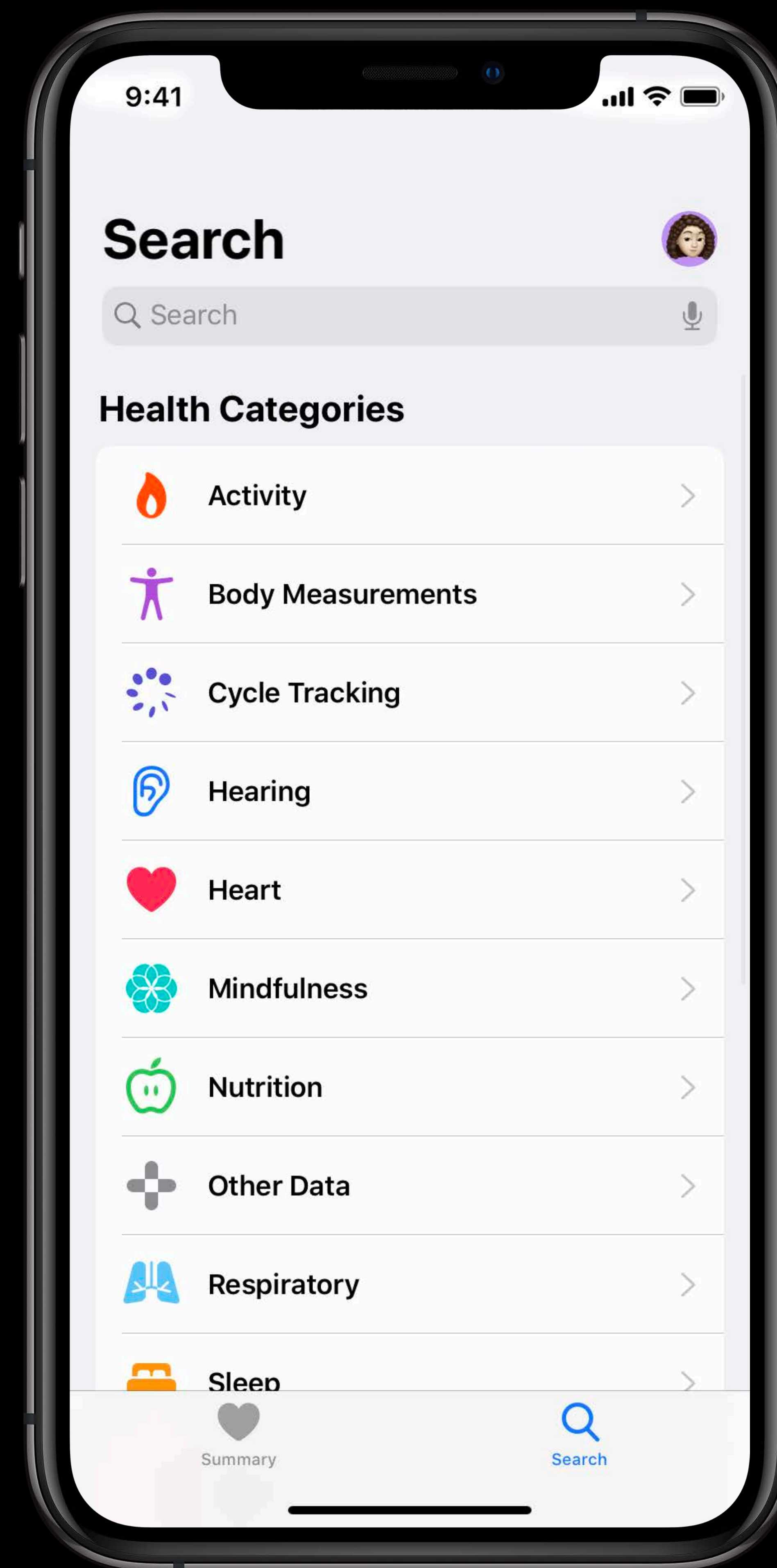


Use all screen width

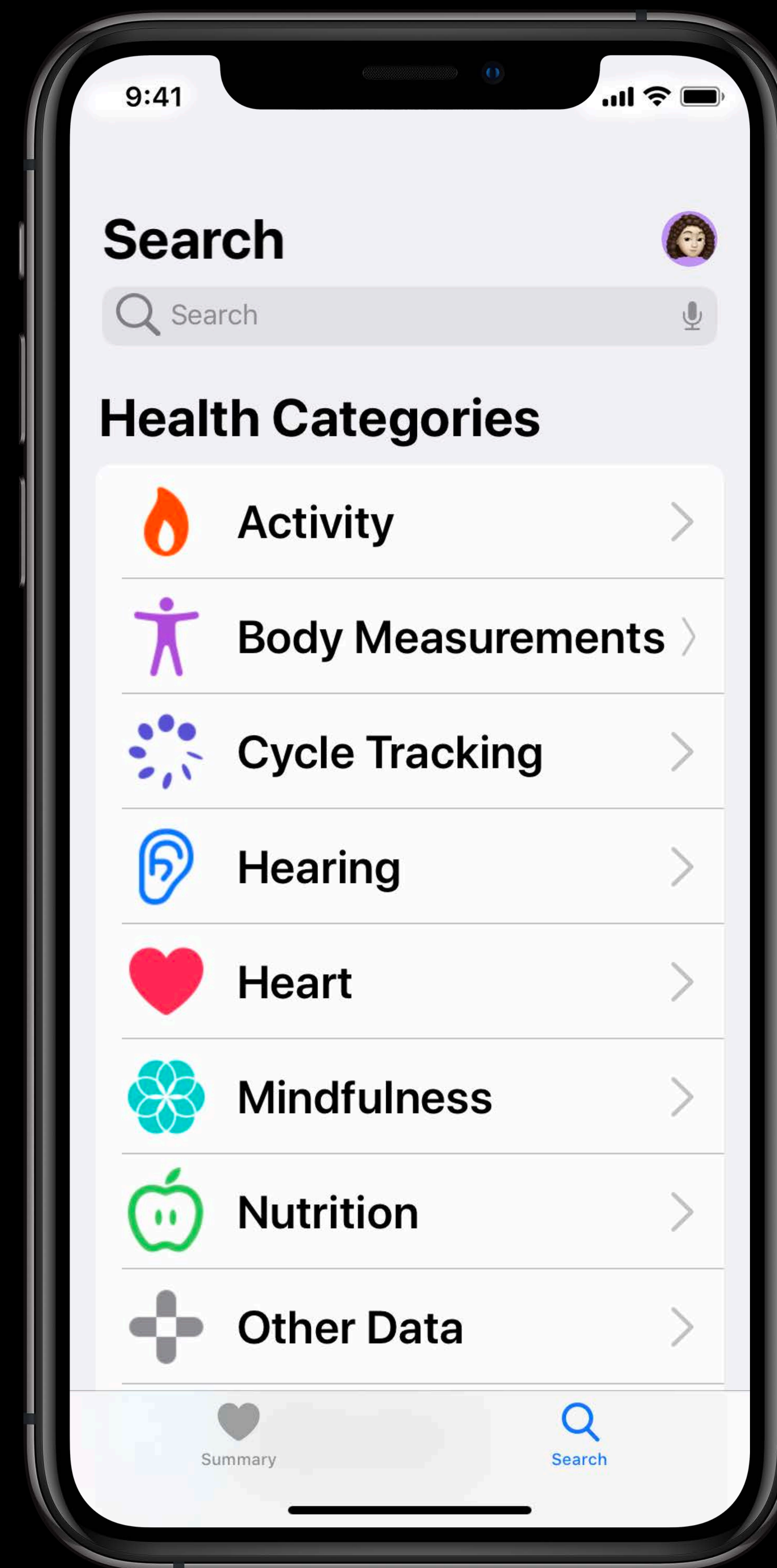


Use all screen width

Don't truncate text



Scale content glyphs



Scale content glyphs

Text Styles

Large Title

Title 1

Title 2

Title 3

Headline

Subheadline

Body

Caption 1

Caption 2

Callout

Footnote

Text Styles

Large Title

Title 1

Title 2

Title 3

Headline

Subheadline

Body

Caption 1

Caption 2

Callout

Footnote



Large Title

Body

Title 1

Caption 1

Title 2

Caption 2

Title 3

Footnote

Headline

Callout

Subheadline

Text Style

```
label.font = UIFont.preferredFont(forTextStyle: .body)  
label.adjustsFontForContentSizeCategory = true
```

Custom Fonts

```
var fontDict: [UIFont.TextStyle : UIFont] = myCustomFontDict()
if let font = fontDict[style] {
    label.font = UIFontMetrics(forTextStyle: style).scaledFont(for: font)
} else {
    label.font = UIFont.preferredFont(forTextStyle: style)
}
```

Custom Fonts

```
var fontDict: [UIFont.TextStyle : UIFont] = myCustomFontDict()
if let font = fontDict[style] {
    label.font = UIFontMetrics(forTextStyle: style).scaledFont(for: font)
} else {
    label.font = UIFont.preferredFont(forTextStyle: style)
}
```

Custom Fonts

```
var fontDict: [UIFont.TextStyle : UIFont] = myCustomFontDict()
if let font = fontDict[style] {
    label.font = UIFontMetrics(forTextStyle: style).scaledFont(for: font)
} else {
    label.font = UIFont.preferredFont(forTextStyle: style)
}
```

Custom Fonts

```
var fontDict: [UIFont.TextStyle : UIFont] = myCustomFontDict()
if let font = fontDict[style] {
    label.font = UIFontMetrics(forTextStyle: style).scaledFont(for: font)
} else {
    label.font = UIFont.preferredFont(forTextStyle: style)
}
```

Custom Fonts

```
var fontDict: [UIFont.TextStyle : UIFont] = myCustomFontDict()
if let font = fontDict[style] {
    label.font = UIFontMetrics(forTextStyle: style).scaledFont(for: font)
} else {
    label.font = UIFont.preferredFont(forTextStyle: style)
}
```

peanutbutter

- peanutbutter
 - AppDelegate.swift
 - PBListViewController.swift
 - PBCell.swift
 - PBDetailViewController.swift
 - PBDetailView.swift
 - PBImagePicker.swift
 - StatusView.swift
 - PBCell.xib
 - PBDetailView.xib
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - Model
 - Status.swift
 - PeanutButter.swift
 - PBViewModel.swift
 - peanutbutter-Bridging-Header.h
- Products

Environment Overrides

Text

Dynamic Type A A

Accessibility

- Increase Contrast
- Reduce Transparency
- Bold Text
- Reduce Motion
- On/Off Labels
- Button Shapes
- Grayscale
- Smart Invert
- Differentiate Without Color

Nutter Buddy

- Creamy • no-stir: **Patty's**
- Chunky • sugar-free: **Anna's**
- Creamy • no-stir: **GIFF**
- Creamy: **Better Butter**
- Chunky • sugar-free • no-stir: **Nutty Nathan's**
- Chunky • sugar-free: **Chunky Monkey**
- Creamy • no-stir: **All Natural**

+ FilterAutoiPhone XR — 13.0 (17A494)Filter

peanutbutter

- peanutbutter
 - AppDelegate.swift
 - PBListViewController.swift
 - PBCell.swift
 - PBDetailViewController.swift
 - PBDetailView.swift
 - PBImagePicker.swift
 - StatusView.swift
 - PBCell.xib
 - PBDetailView.xib
 - Assets.xcassets
 - LaunchScreen.storyboard
 - Info.plist
 - Model
 - Status.swift
 - PeanutButter.swift
 - PBViewModel.swift
 - peanutbutter-Bridging-Header.h
- Products

```
1 //
2 // AppDelegate
3 // AppDelegate
4 // AppDelegate
5 // AppDelegate
6 // AppDelegate
7 // AppDelegate
8
9 import UIKit
10
11 @UIApplicationMain
12 class AppDelegate: UIResponder, UIApplicationDelegate {
13
14     var window: UIWindow?
15
16     func application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]?) throws -> Bool {
17         // Override point for customization after application launch.
18
19         return true
20     }
21
22     func applicationWillResignActive(_ application: UIApplication) {
23         // Sent when the application is about to move from active to inactive state. This can occur due to memory pressure, suspending the application, or pressing the home key. Use this opportunity to eventually save your application data, release extra resources, etc.
24     }
25
26     func applicationDidEnterBackground(_ application: UIApplication) {
27         // Use this method to release shared resources, save user data, invalidate timers, and perform one-time cleanup.
28     }
29
30     func applicationWillEnterForeground(_ application: UIApplication) {
31         // Called as part of the transition from the background to the active state; here you can undo many of the changes made on entering the background.
32     }
33
34     func applicationDidBecomeActive(_ application: UIApplication) {
35         // Restart any tasks that were paused (or not yet started) while the application was inactive. If the application was previously in the background, you may have to refresh the UI.
36     }
37
38     func applicationWillTerminate(_ application: UIApplication) {
39         // Called when the application is about to terminate. Save data if appropriate. See also applicationDidEnterBackground:.
40     }
41
42     func application(_ application: UIApplication, open url: URL, options: [UIApplication.OpenURLOptionsKey: Any]?) throws -> Bool {
43         // Respond to app opening with URL that was shared with application.
44     }
45
46     func application(_ application: UIApplication, continue userActivity: NSUserActivity, restorationHandler: () -> Void?) throws -> Bool {
47         // Respond to app opening with user activity that was shared with application.
48     }
49
50     func application(_ application: UIApplication, supportedActivities: Set<NSUserActivityTypeKey>, restorationRestrictions: UIApplication.RestorationRestrictions?, userActivity: NSUserActivity?) -> Bool {
51         // Add supported user activities to your app.
52     }
53
54     func application(_ application: UIApplication, supportedInterfaceOrientationsFor window: UIWindow?) -> UIInterfaceOrientationMask {
55         // Add supported interface orientations to your app.
56     }
57
58     func application(_ application: UIApplication, shouldSaveApplicationState userStateKey: String, completionHandler: () -> Void?) -> Bool {
59         // Respond to app state saving request.
60     }
61
62     func application(_ application: UIApplication, shouldRestoreApplicationState userStateKey: String, completionHandler: () -> Void?) -> Bool {
63         // Respond to app state restoration request.
64     }
65
66     func application(_ application: UIApplication, shouldPerformActionForShortcutItem: UIApplicationShortcutItem, completionHandler: () -> Void?) -> Bool {
67         // Respond to app shortcuts.
68     }
69
70     func application(_ application: UIApplication, didReceiveRemoteNotification userInfo: [AnyHashable: Any]?, fetchCompletionHandler completionHandler: @escaping (UIBackgroundFetchResult) -> Void) -> Void {
71         // If your app supports background execution, this method is called when the user didn't open the app because it was in the background. You can use the completionHandler to start a short-running task and return one of the background fetch completion results.
72     }
73
74     func application(_ application: UIApplication, didRegisterForRemoteNotificationsWithDeviceToken deviceToken: Data) {
75         // Your app has successfully registered for remote notifications. Here you can do more in-depth processing, such as handling a subscription to new notifications with the APNs Authentication Key.
76     }
77
78     func application(_ application: UIApplication, didFailToRegisterForRemoteNotificationsWithError error: (NSError)?) {
79         // Your app failed to register for remote notifications. Here you can do more in-depth processing, such as handling an Apple Push Notification service (APNs) authentication error.
80     }
81
82     func application(_ application: UIApplication, didReceiveRemoteNotification userInfo: [AnyHashable: Any]?, fetchCompletionHandler completionHandler: @escaping (UIBackgroundFetchResult) -> Void) -> Void {
83         // If your app supports background execution, this method is called when the user didn't open the app because it was in the background. You can use the completionHandler to start a short-running task and return one of the background fetch completion results.
84     }
85
86     func application(_ application: UIApplication, didReceiveRemoteNotification userInfo: [AnyHashable: Any]?, fetchCompletionHandler completionHandler: @escaping (UIBackgroundFetchResult) -> Void) -> Void {
87         // If your app supports background execution, this method is called when the user didn't open the app because it was in the background. You can use the completionHandler to start a short-running task and return one of the background fetch completion results.
88     }
89
90     func application(_ application: UIApplication, didReceiveRemoteNotification userInfo: [AnyHashable: Any]?, fetchCompletionHandler completionHandler: @escaping (UIBackgroundFetchResult) -> Void) -> Void {
91         // If your app supports background execution, this method is called when the user didn't open the app because it was in the background. You can use the completionHandler to start a short-running task and return one of the background fetch completion results.
92     }
93
94     func application(_ application: UIApplication, didReceiveRemoteNotification userInfo: [AnyHashable: Any]?, fetchCompletionHandler completionHandler: @escaping (UIBackgroundFetchResult) -> Void) -> Void {
95         // If your app supports background execution, this method is called when the user didn't open the app because it was in the background. You can use the completionHandler to start a short-running task and return one of the background fetch completion results.
96     }
97
98     func application(_ application: UIApplication, didReceiveRemoteNotification userInfo: [AnyHashable: Any]?, fetchCompletionHandler completionHandler: @escaping (UIBackgroundFetchResult) -> Void) -> Void {
99         // If your app supports background execution, this method is called when the user didn't open the app because it was in the background. You can use the completionHandler to start a short-running task and return one of the background fetch completion results.
100    }
```

Environment Overrides

Text

Dynamic Type A A

Accessibility

- Increase Contrast
- Reduce Transparency
- Bold Text
- Reduce Motion
- On/Off Labels
- Button Shapes
- Grayscale
- Smart Invert
- Differentiate Without Color

Nutter Buddy

9:41

Creamy • no-stir

♥ **Patty's**

👍👍👍👍👍👍👍

Chunky • sugar-free

♥ **Anna's**

👍👍👍👍👍👍👍

Creamy • no-stir

♥ **GIFF**

👍👍👍👍👍👍👍

Creamy

♥ **Better Butter**

👍👍👍👍👍👍👍

Chunky • sugar-free • no-stir

♥ **Nutty Nathan's**

👍👍👍👍👍👍👍

Chunky • sugar-free

♥ **Chunky Monkey**

👍👍👍👍👍👍👍

Creamy • no-stir

♥ **All Natural**

Filter

Auto

iPhone XR — 13.0 (17A494)

Filter

peanutbutter \ peanutbutter \ AppDelegate.swift

```

1 //
2 // AppDelegate
3 //
4 //
5 //
6 //
7 //
8
9 import
10
11 @UIApp
12 class
13
14 va
15
16
17 fu
18
19
20
21
22

```

peanutbutter

- AppDelegate.swift
- PBListViewController.swift
- PBCell.swift
- PBDetailViewController.swift
- PBDetailView.swift
- PBImagePicker.swift
- StatusView.swift
- PBCell.xib
- PBDetailView.xib
- Assets.xcassets
- LaunchScreen.storyboard
- Info.plist
- Model
 - Status.swift
 - PeanutButter.swift
 - PBViewModel.swift
- peanutbutter-Bridging-Header.h
- Products

Environment Overrides

Text


Dynamic Type A A
Accessibility Medium

Accessibility

- Increase Contrast
- Reduce Transparency
- Bold Text
- Reduce Motion
- On/Off Labels
- Button Shapes
- Grayscale
- Smart Invert
- Differentiate Without Color


9:41 Nutter Buddy

Creamy • no-stir

♥ Patty's 


👉👉👉👉👉👉👉

Chunky • sugar-free

♥ Anna's 


👉👉👉👉👉👉👉

Creamy • no-stir

♥ GIFF 


👉👉👉👉👉👉👉

Creamy

♥ Better Butter 


👉👉👉👉👉👉👉

Chunky • sugar-free • no-stir

♥ Nutty Nathan's 

👉👉👉👉👉👉👉

Chunky • sugar-free

♥ Chunky Monkey 

👉👉👉👉👉👉👉

Dynamic Type

Reduce Motion

Differentiate Without Color

9:41



Madison

Rain

55°

Friday TODAY

69 64

Now	12PM	1PM	2PM	3PM	4PM	5PM
	90%	50%			40%	50%
55°	55°	58°	60°	63°	66°	67°

Saturday		83	55
Sunday		75	54
Monday		70	59
Tuesday		72	57
Wednesday		73	52
Thursday		68	50
Friday		70	55
Saturday		70	53
Sunday		70	52

The Weather Channel



9:41



Madison

Rain

55°

Friday TODAY

69 64

Now	12PM 90%	1PM 50%	2PM	3PM	4PM 40%	5PM 50%
55°	55°	58°	60°	63°	66°	67°

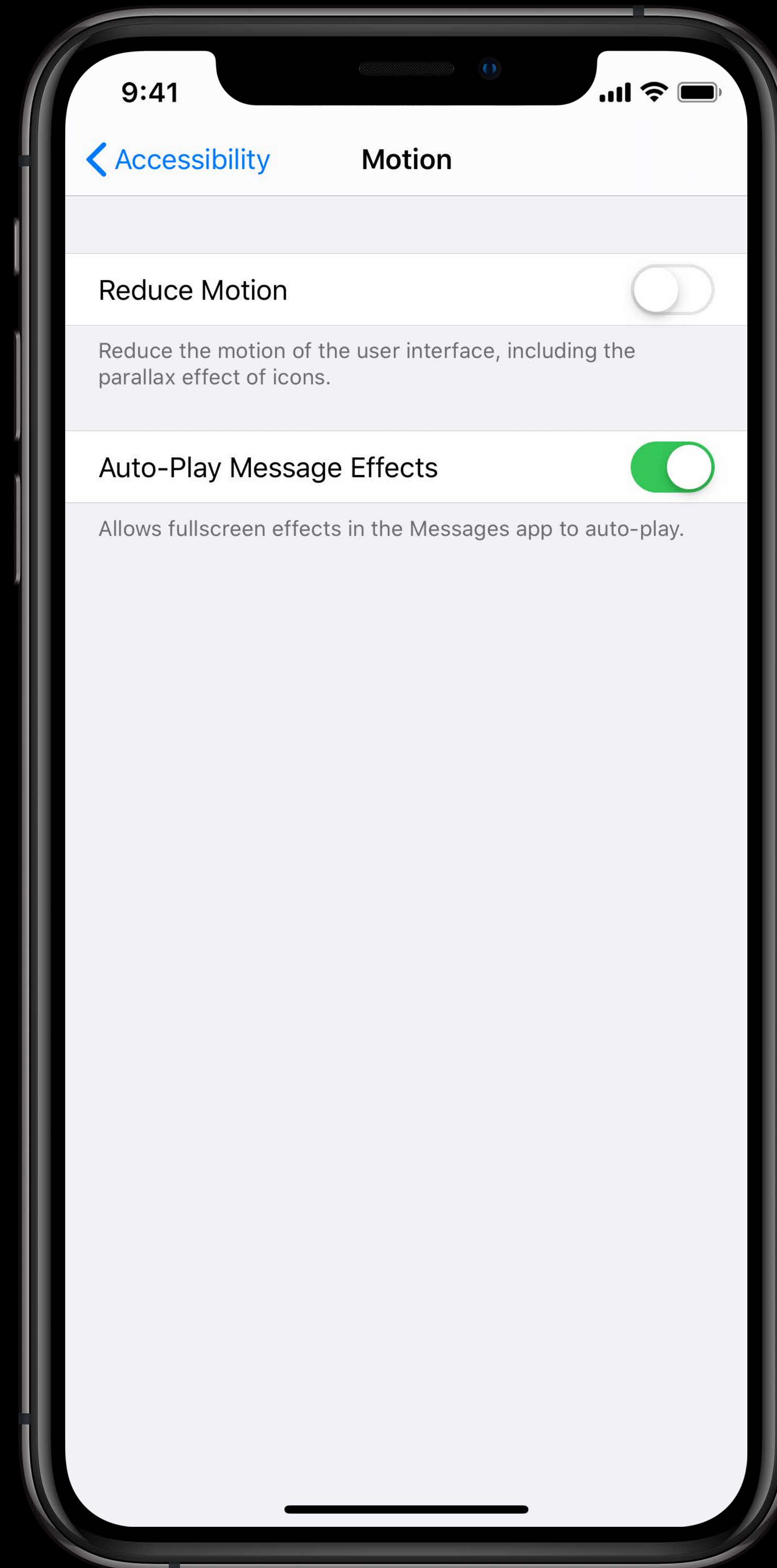
Saturday		83 55
Sunday		75 54
Monday		70 59
Tuesday		72 57
Wednesday		73 52
Thursday		68 50
Friday		70 55
Saturday		70 53
Sunday		70 52



1 in 3

Experience some form of motion sensitivity

Reduce Motion

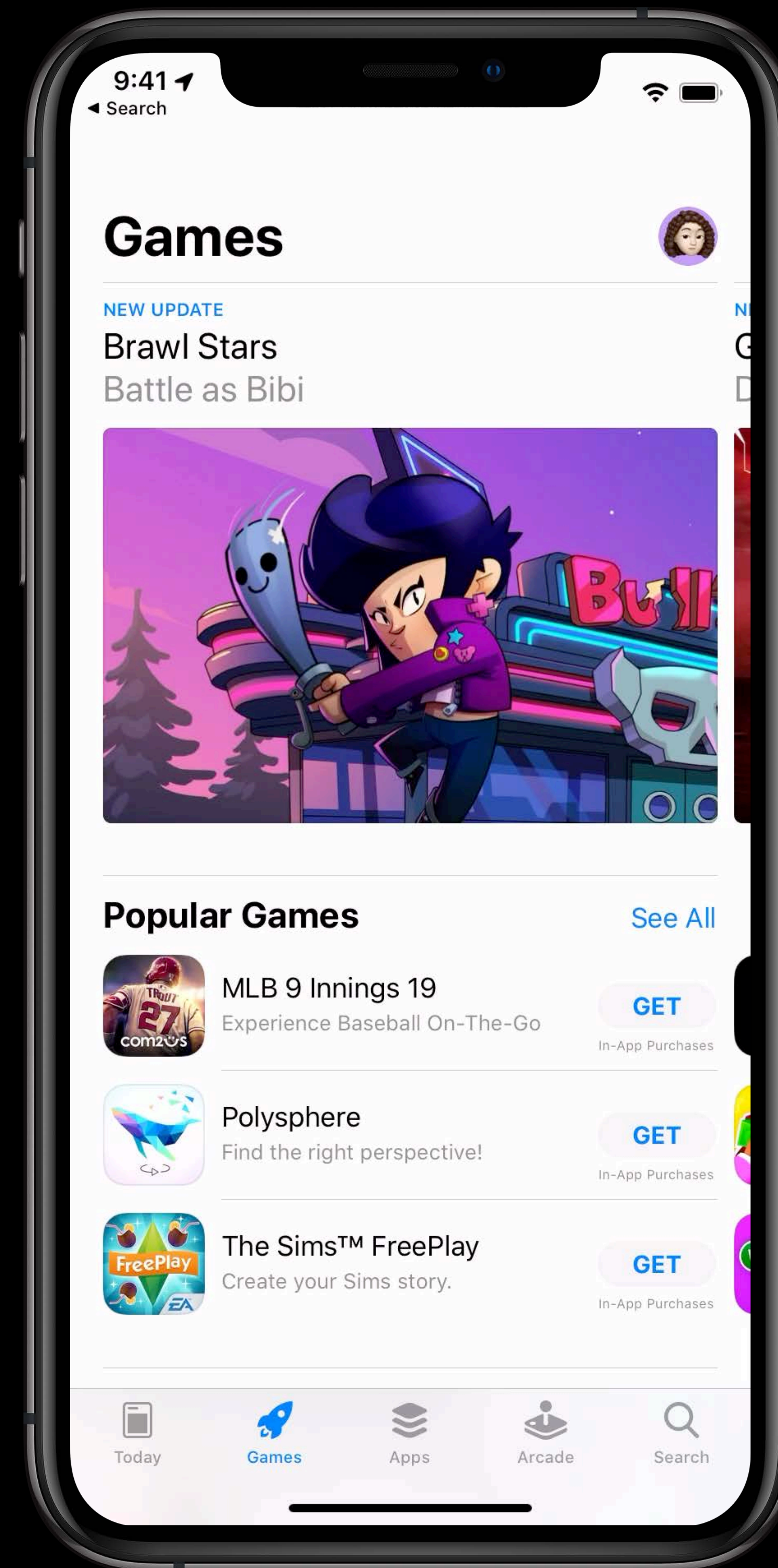


Reduce Motion

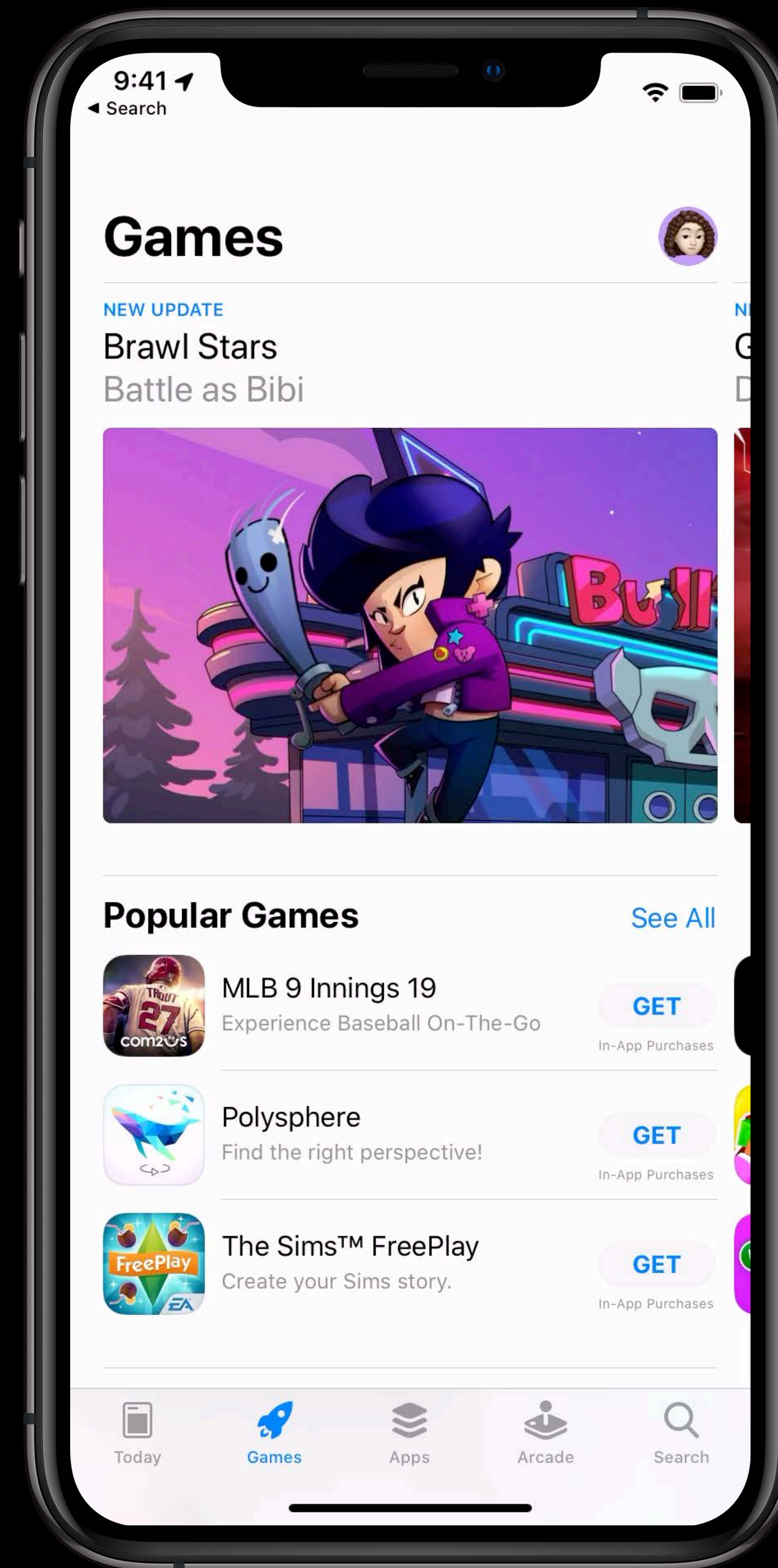



```
static var UIAccessibility.isReduceMotionEnabled: Bool { get }
```

```
static let UIAccessibility.reduceMotionStatusDidChangeNotification: NSNotification.Name
```

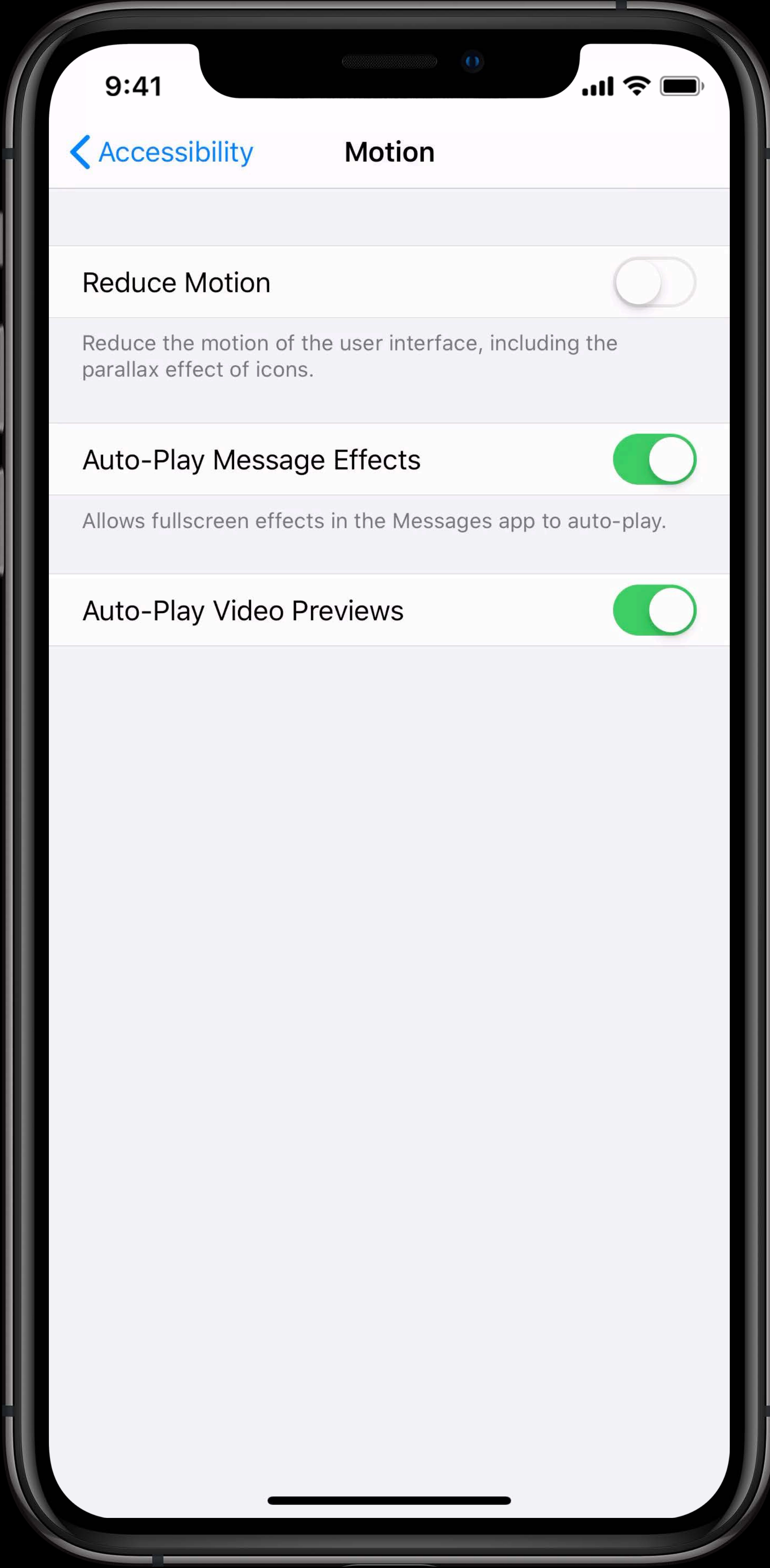


Autoplaying video



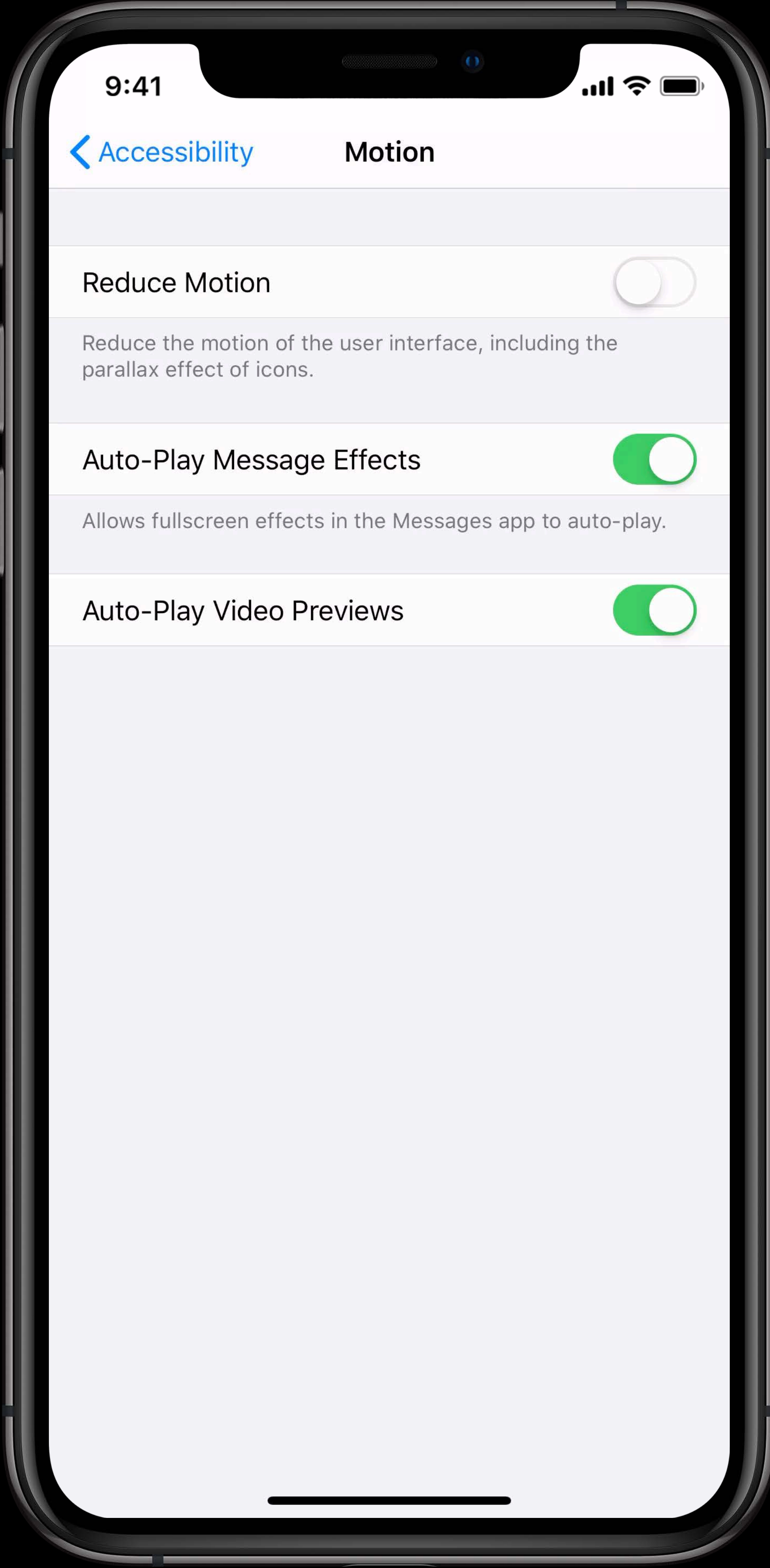
Autoplaying video

NEW



Auto-Play Video Previews

NEW



Auto-Play Video Previews

```
static var UIAccessibility.isVideoAutoplayEnabled: Bool { get }
```

```
static let UIAccessibility.videoAutoplayStatusDidChangeNotification: NSNotification.Name
```

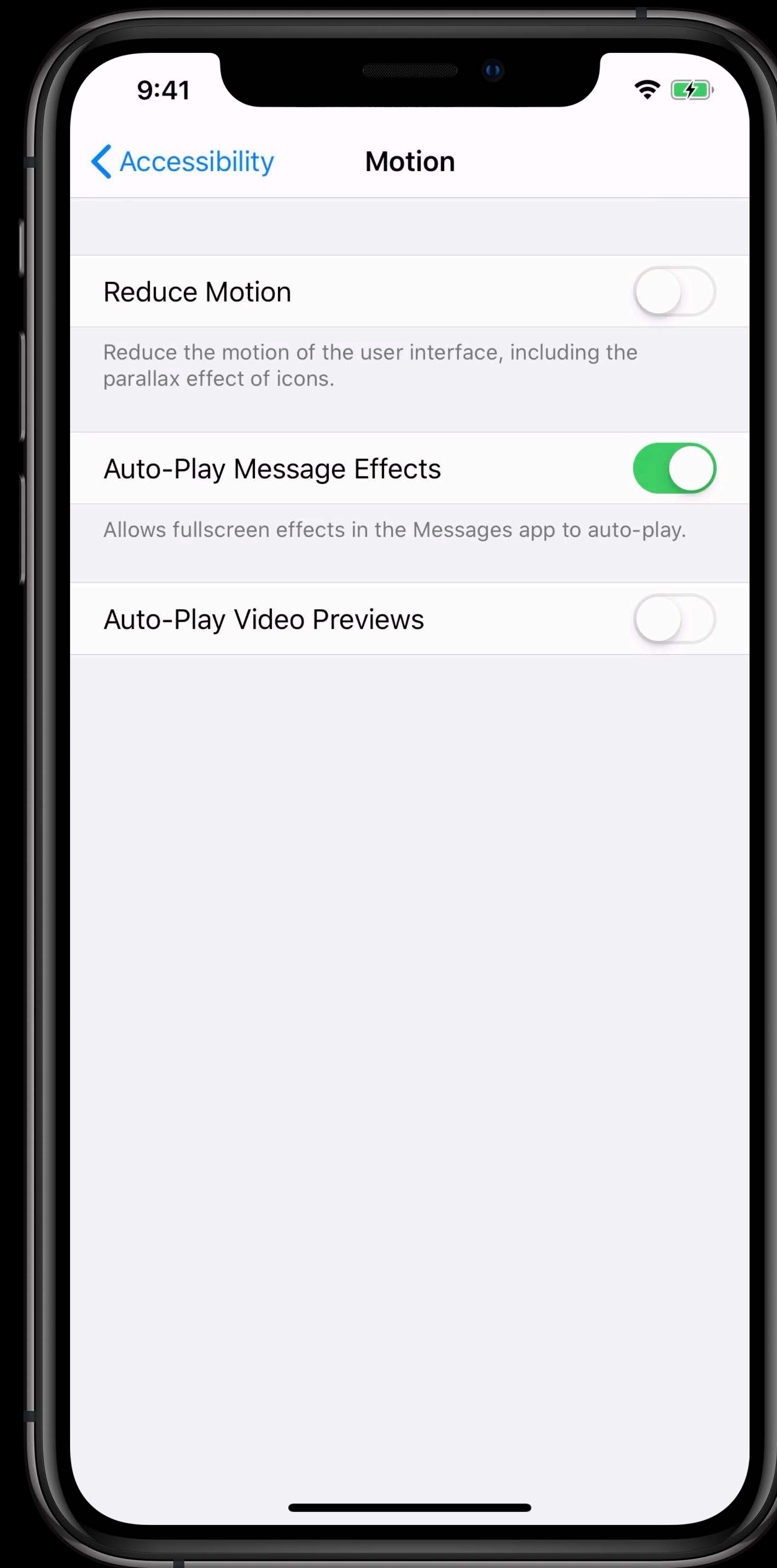


NEW

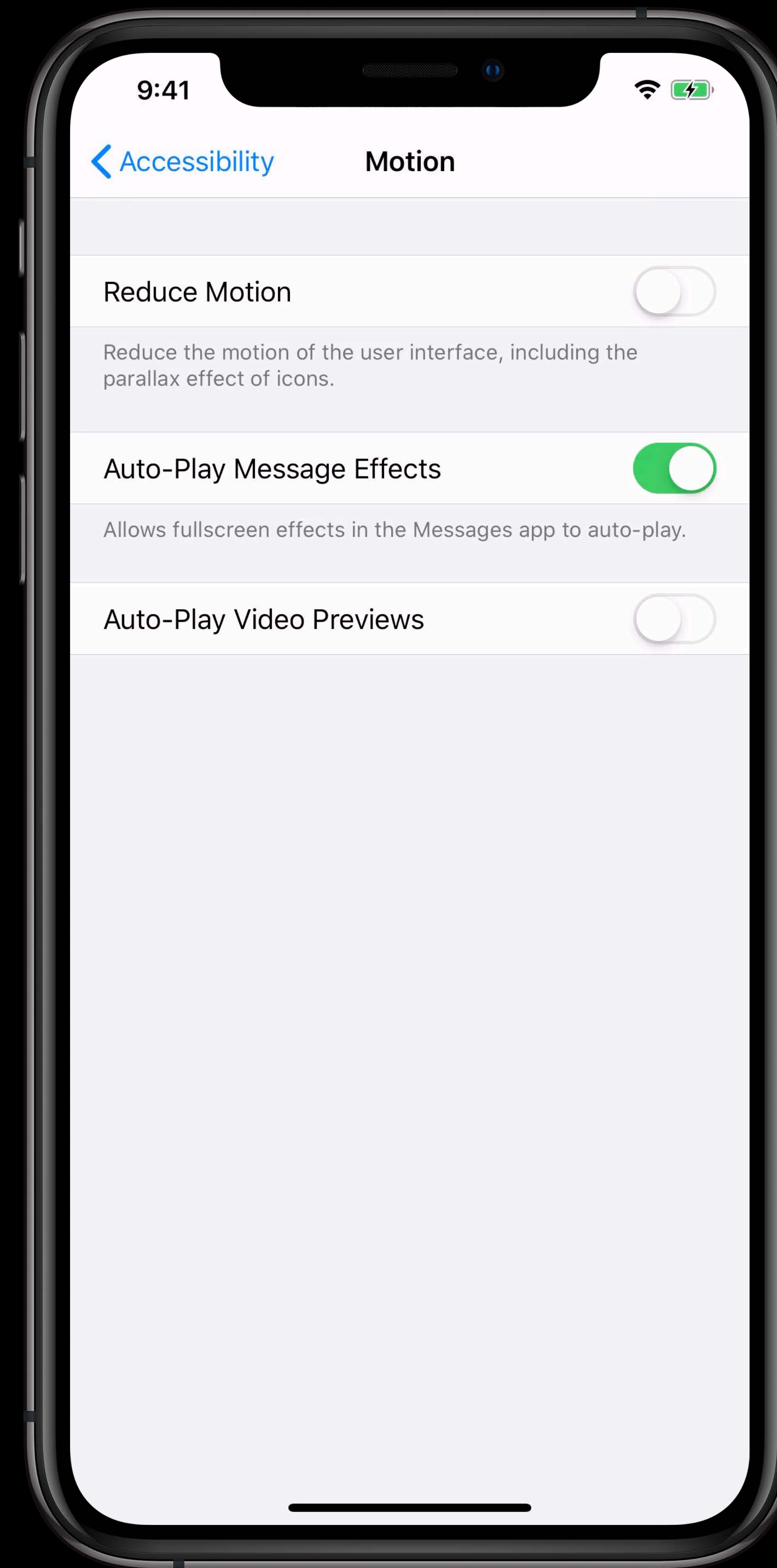
Mirror by Default

Mirror the video-autoplay setting by default

Users can override

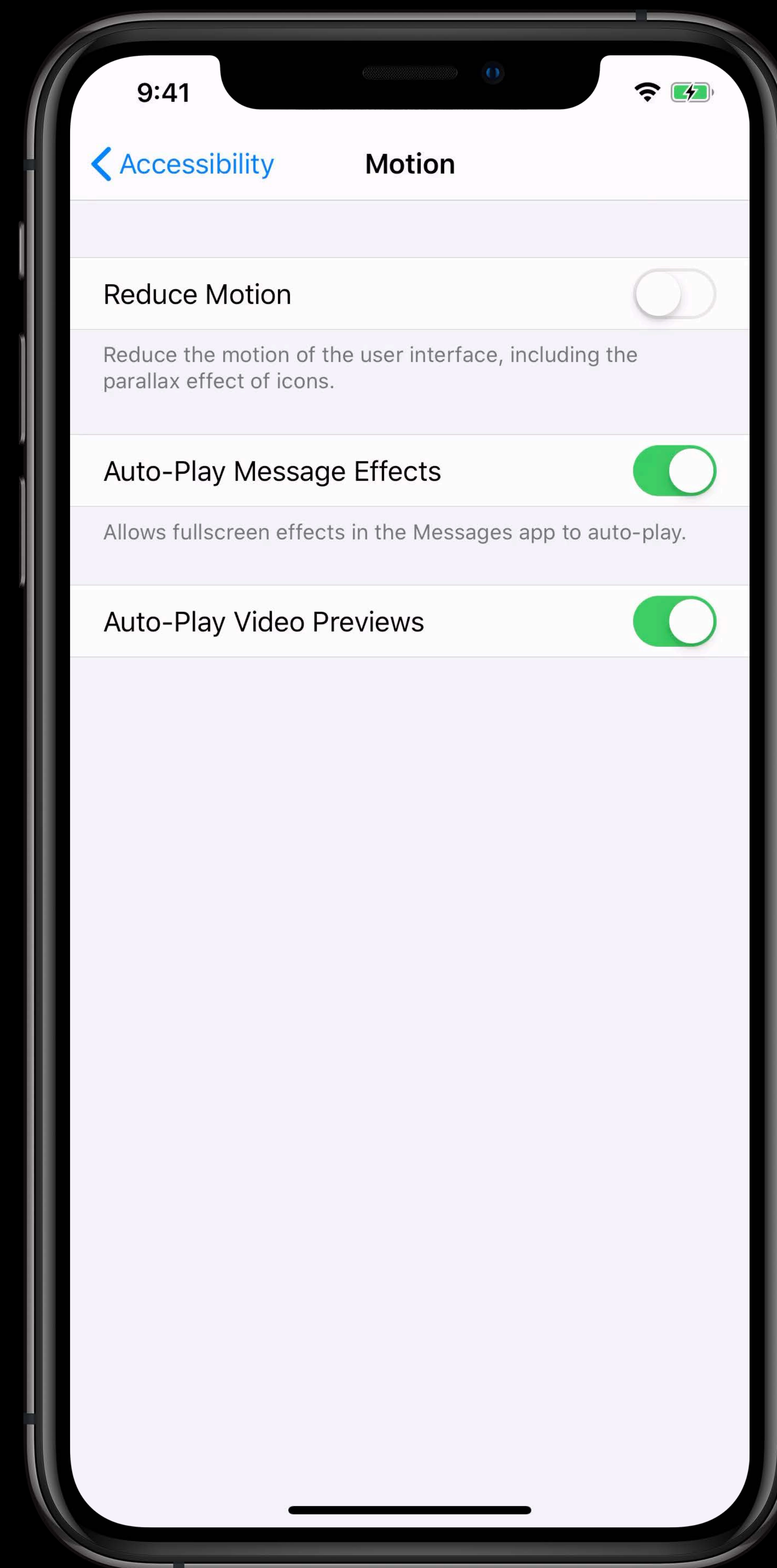


Screen Transitions



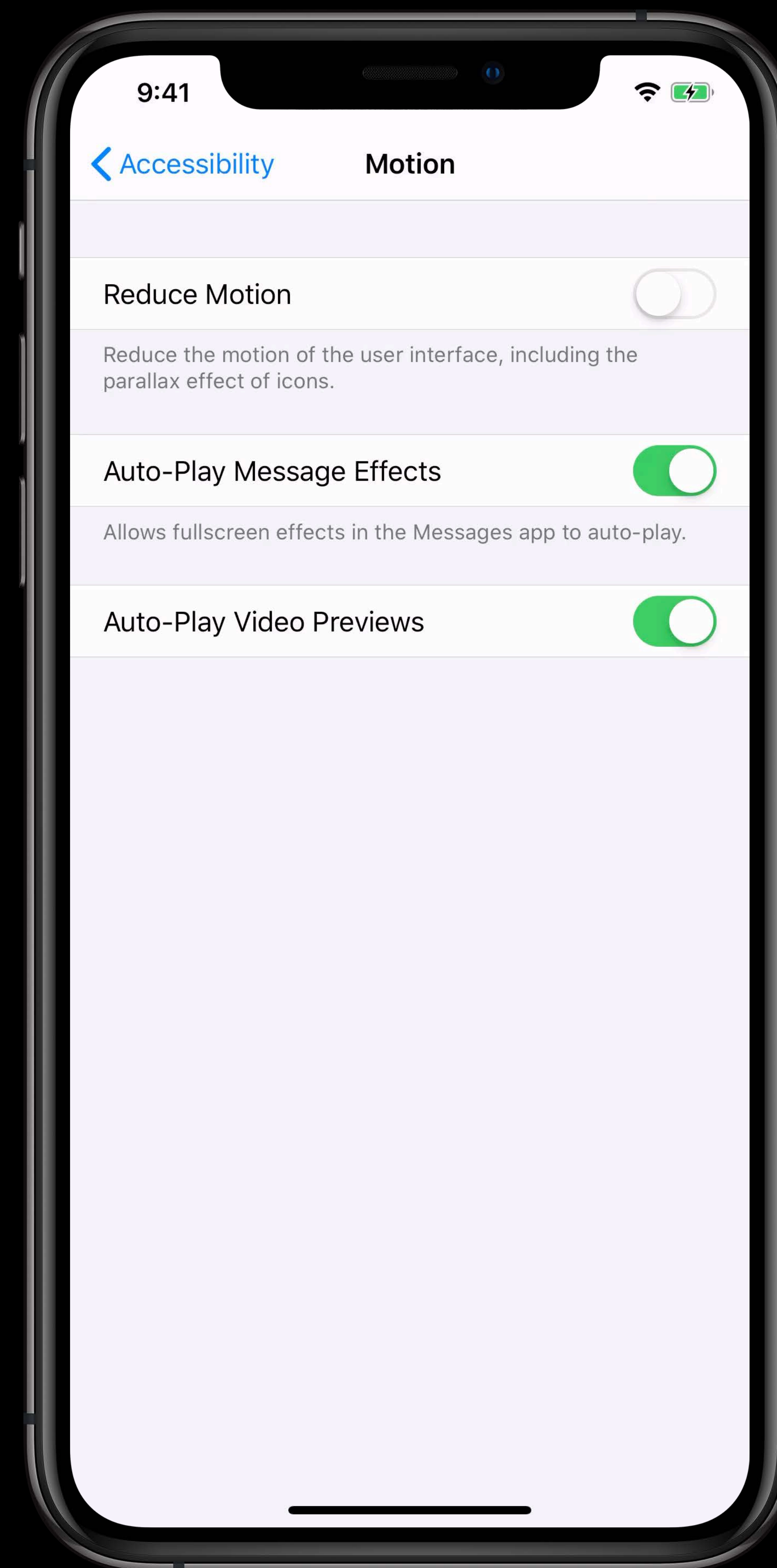
Screen Transitions

NEW



Cross-fade Transitions

NEW



Cross-fade Transitions

Dynamic Type

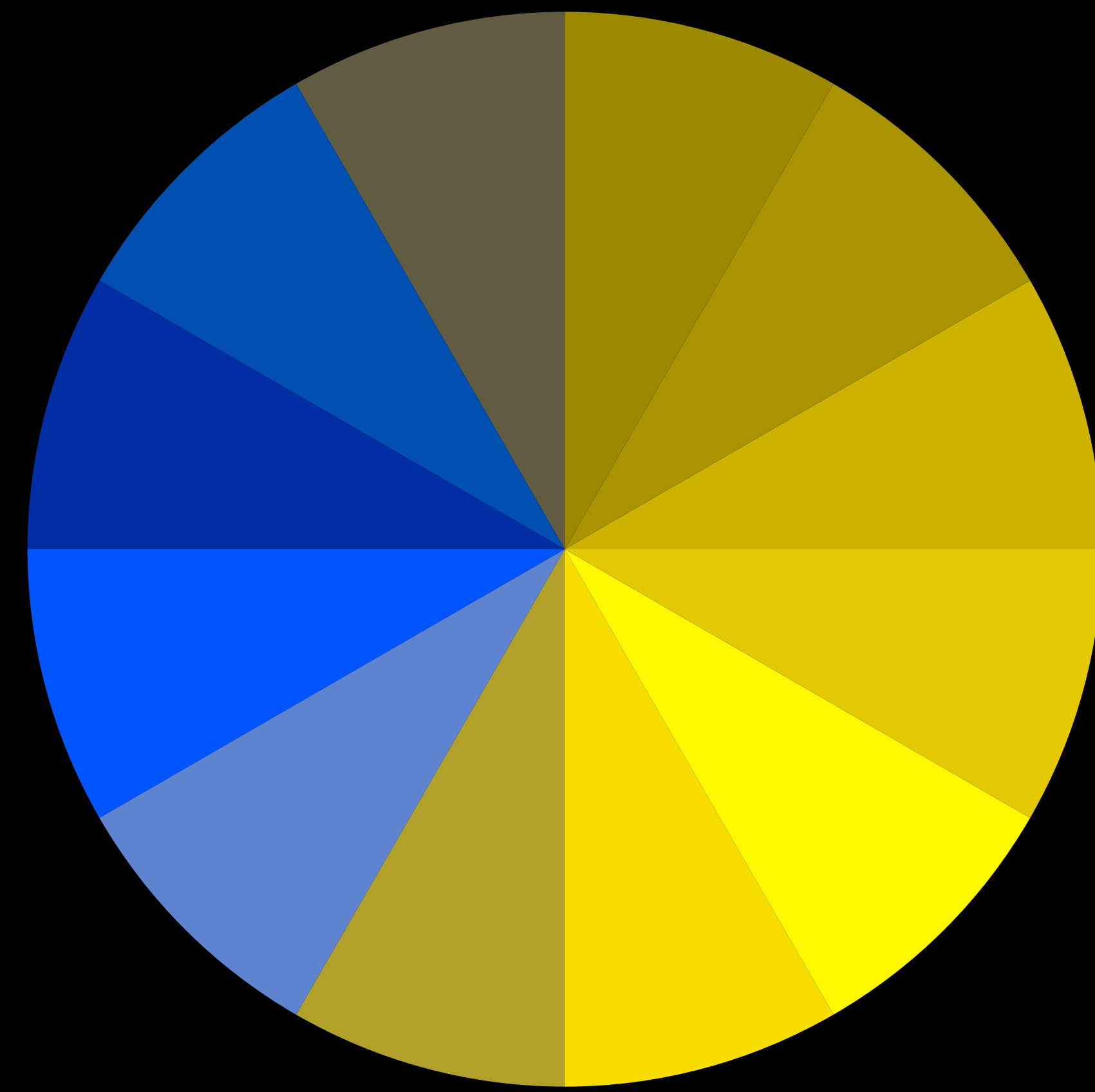
Reduce Motion

Differentiate Without Color

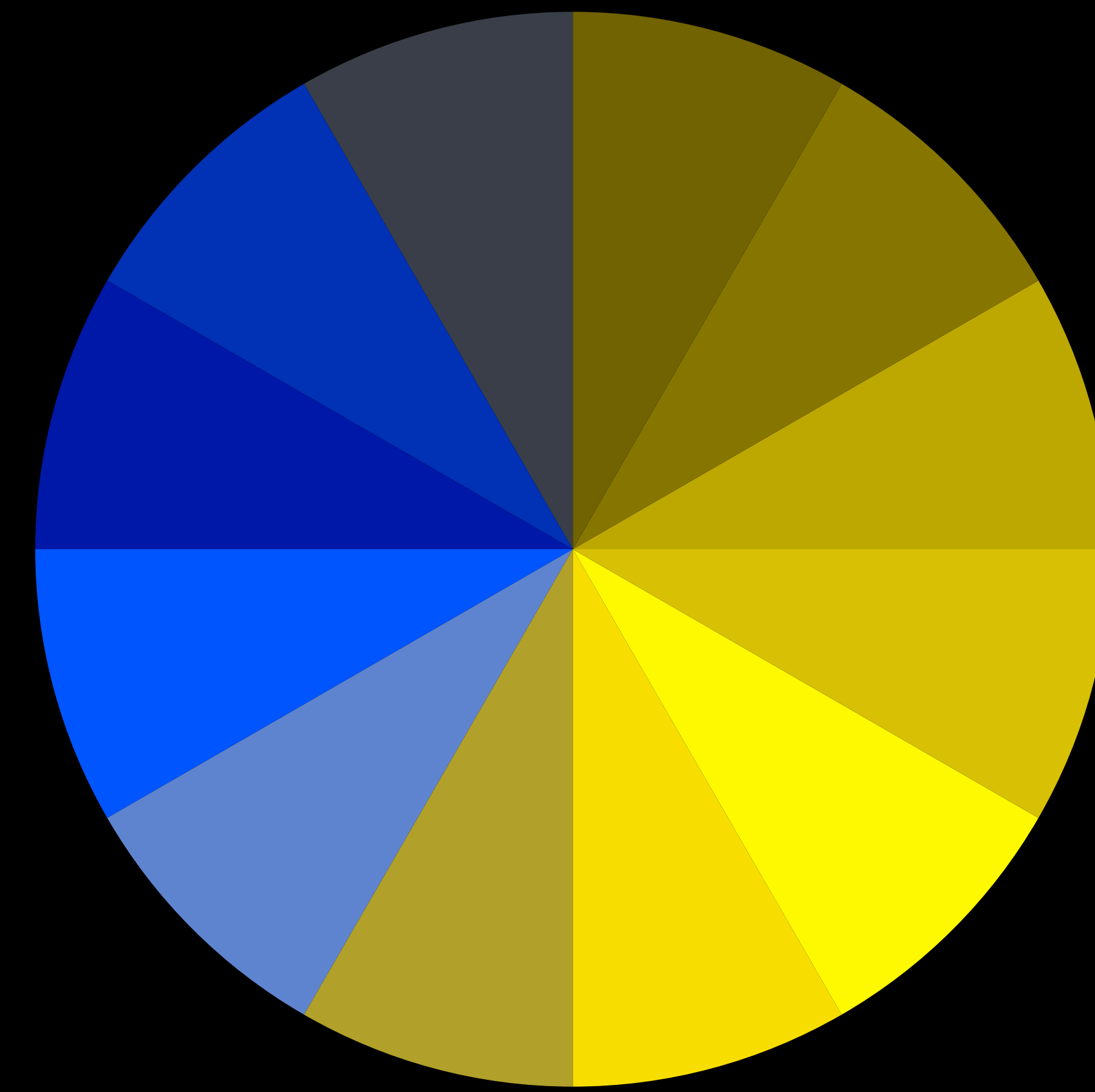
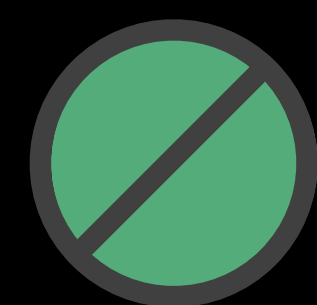
Color-Blindness



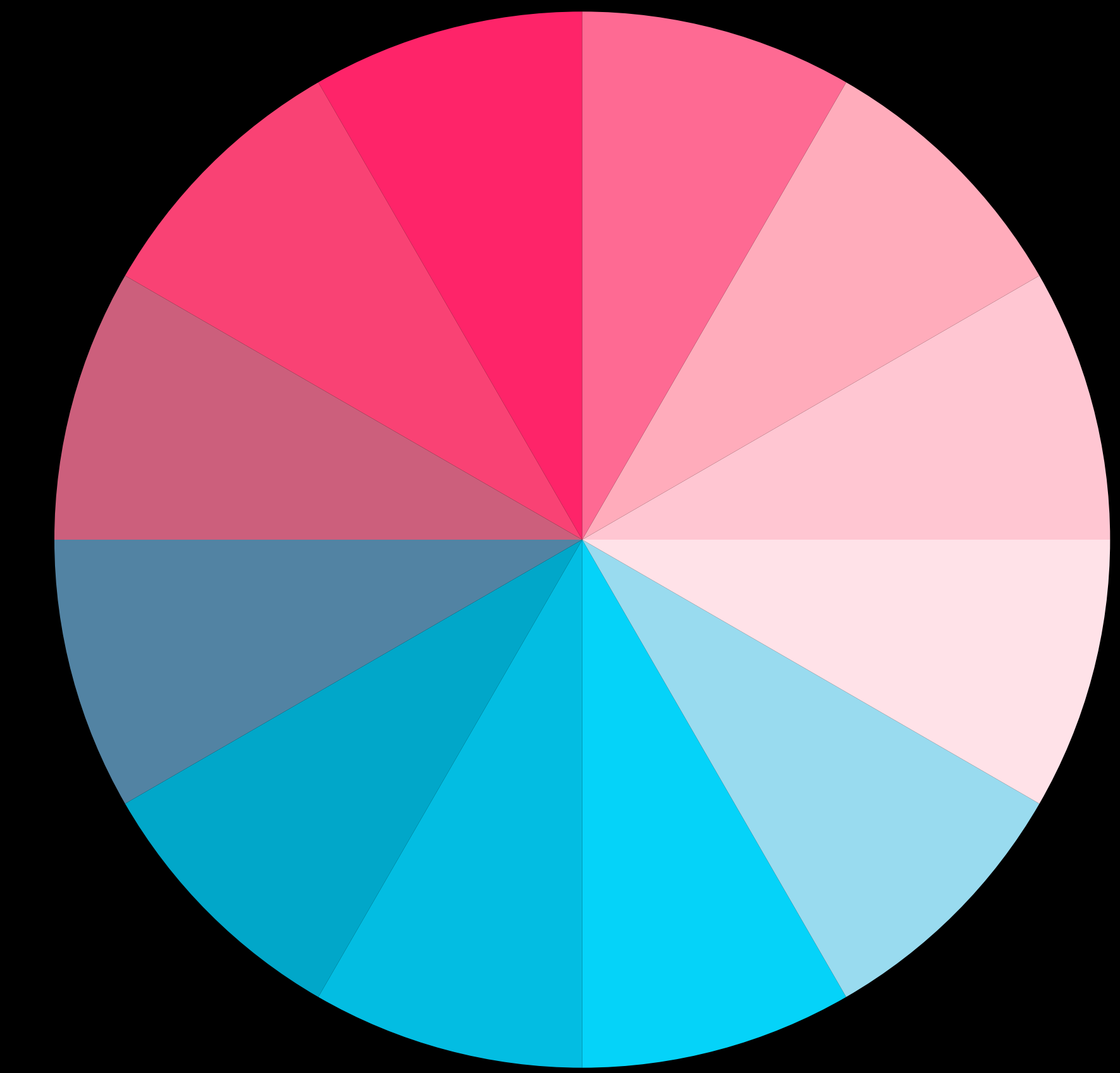
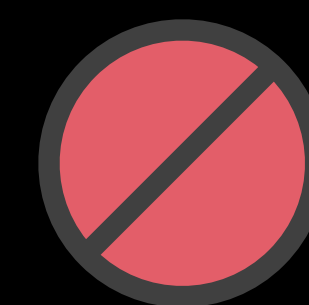
Normal Vision



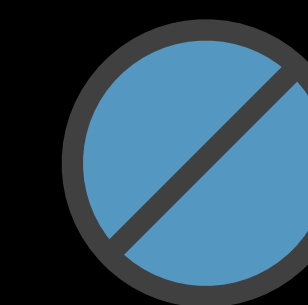
Deuteranopia



Protanopia



Tritanopia



9:41



Nutter Buddy



Creamy • no-stir

Patty's



Chunky • sugar-free

Anna's



Creamy • no-stir

GIFF



Creamy

Better Butter



Chunky • sugar-free • no-stir

Nutty Nathan's



Chunky • sugar-free

Chunky Monkey



Creamy • no-stir

All Natural



9:41



Nutter Buddy



Creamy • no-stir

Patty's



Chunky • sugar-free

Anna's



Creamy • no-stir

GIFF



Creamy

Better Butter



Chunky • sugar-free • no-stir

Nutty Nathan's



Chunky • sugar-free

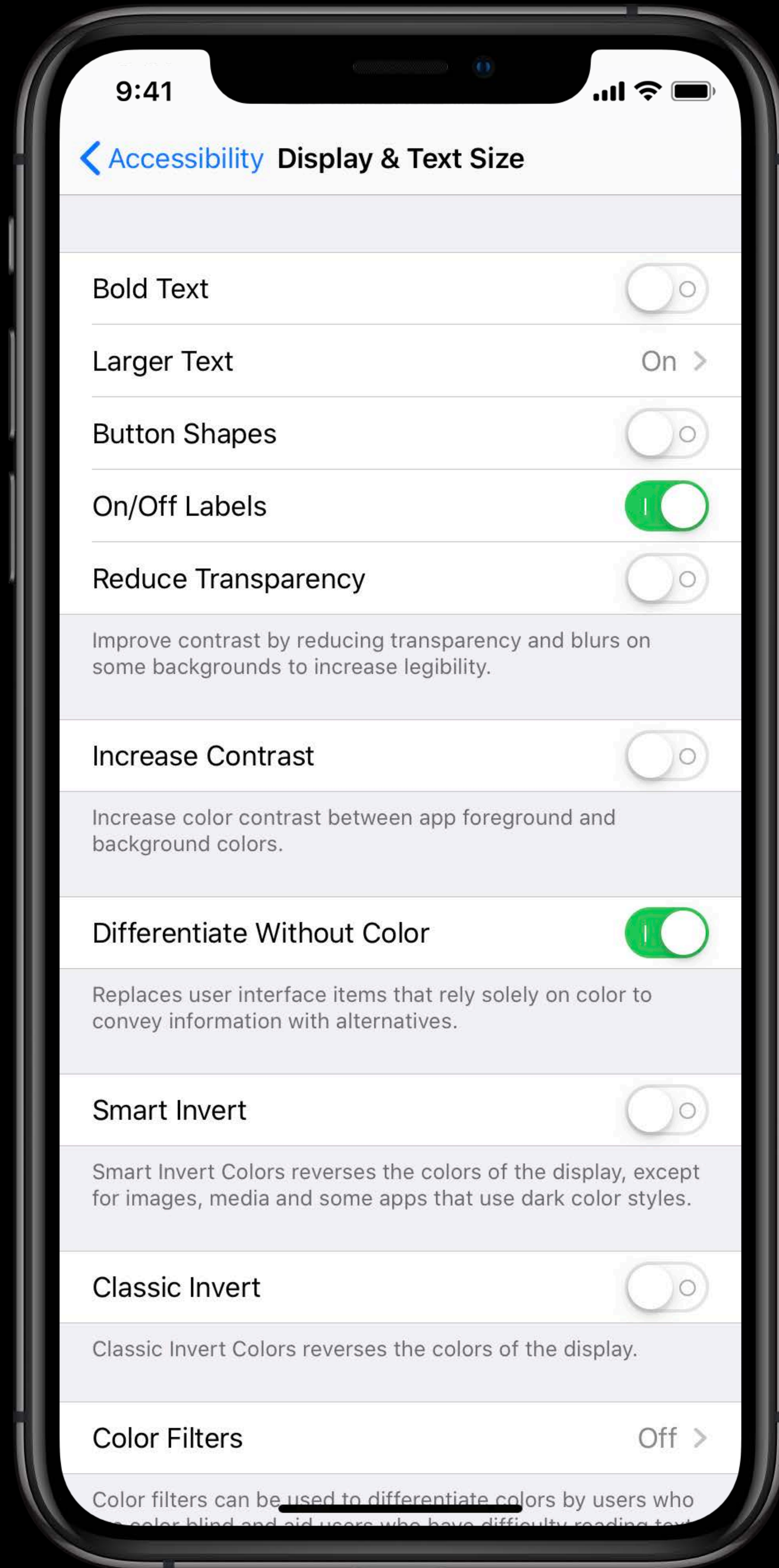
Chunky Monkey



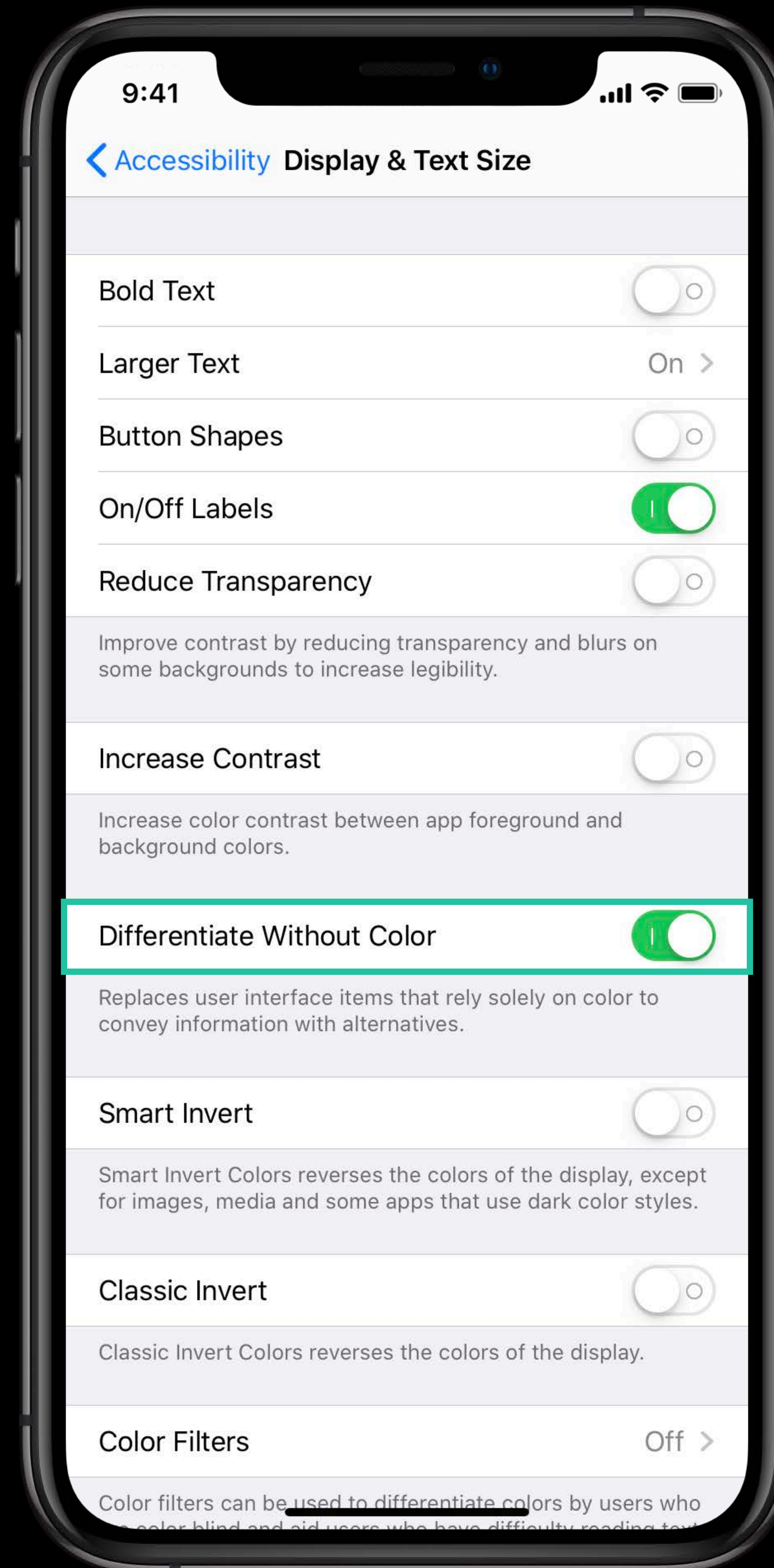
Creamy • no-stir

All Natural





NEW



NEW

```
public func UIAccessibility.shouldDifferentiateWithoutColor() -> Bool
```

NEW

```
public static let UIAccessibility.differentiateWithoutColorDidChangeNotification:  
NSNotification.Name
```

9:41



Nutter Buddy



Creamy • no-stir



Patty's



Chunky • sugar-free



Anna's



Creamy • no-stir



GIFF



Creamy



Better Butter



Chunky • sugar-free • no-stir



Nutty Nathan's



Chunky • sugar-free



Chunky Monkey



Creamy • no-stir



All Natural



9:41



Nutter Buddy



Creamy • no-stir



Patty's



Chunky • sugar-free



Anna's



Creamy • no-stir



GIFF



Creamy



Better Butter



Chunky • sugar-free • no-stir



Nutty Nathan's



Chunky • sugar-free



Chunky Monkey



Creamy • no-stir



All Natural



9:41



Nutter Buddy



Creamy • no-stir



Patty's



Chunky • sugar-free



Anna's



Creamy • no-stir



GIFF



Creamy



Better Butter



Chunky • sugar-free • no-stir



Nutty Nathan's



Chunky • sugar-free



Chunky Monkey



Creamy • no-stir



All Natural



9:41



Nutter Buddy



Creamy • no-stir



Patty's



Chunky • sugar-free



Anna's



Creamy • no-stir



GIFF



Creamy



Better Butter



Chunky • sugar-free • no-stir



Nutty Nathan's



Chunky • sugar-free



Chunky Monkey



Creamy • no-stir



All Natural



9:41



Nutter Buddy



Creamy • no-stir



Patty's



Chunky • sugar-free



Anna's



Creamy • no-stir



GIFF



Creamy



Better Butter



Chunky • sugar-free • no-stir



Nutty Nathan's



Chunky • sugar-free



Chunky Monkey



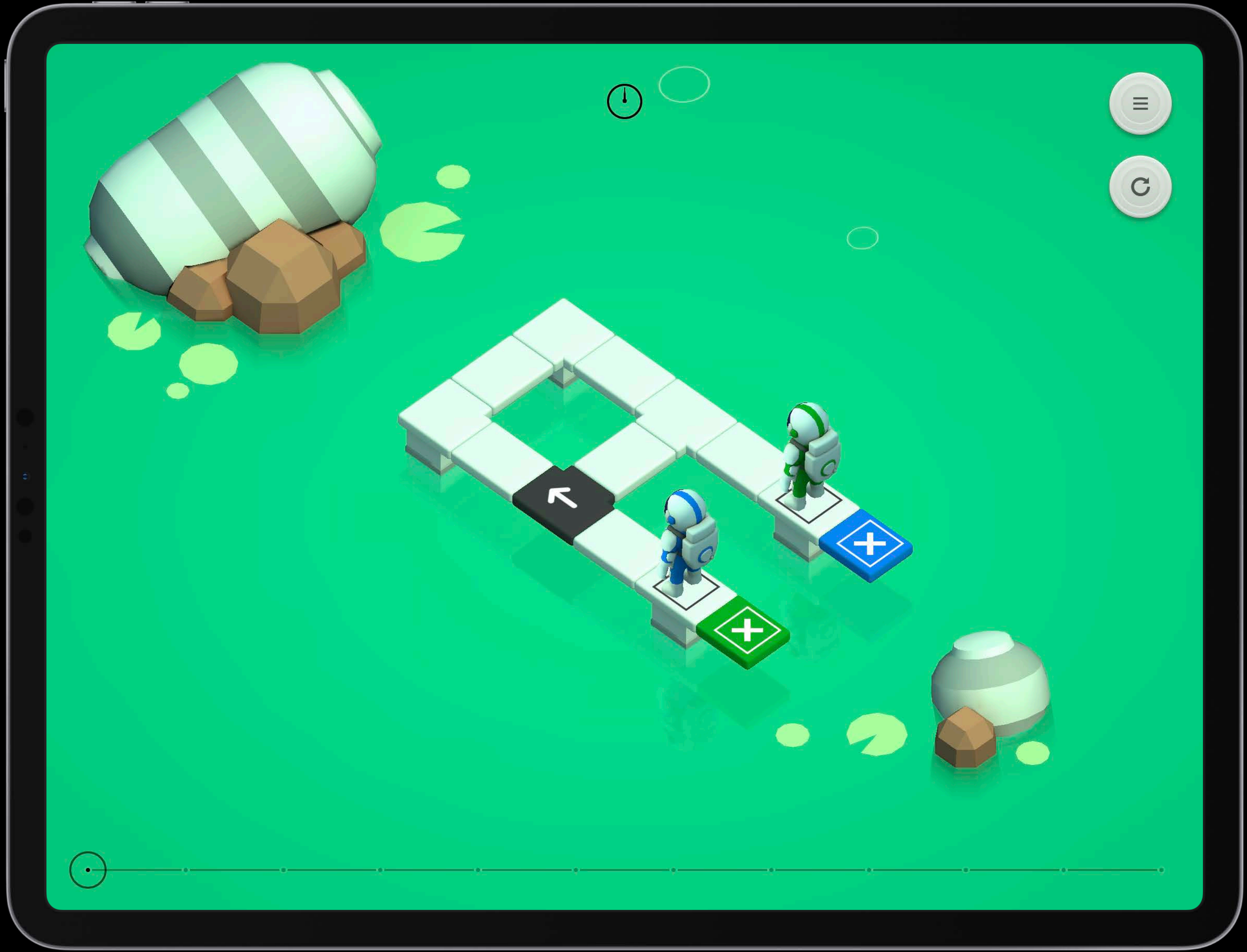
Creamy • no-stir



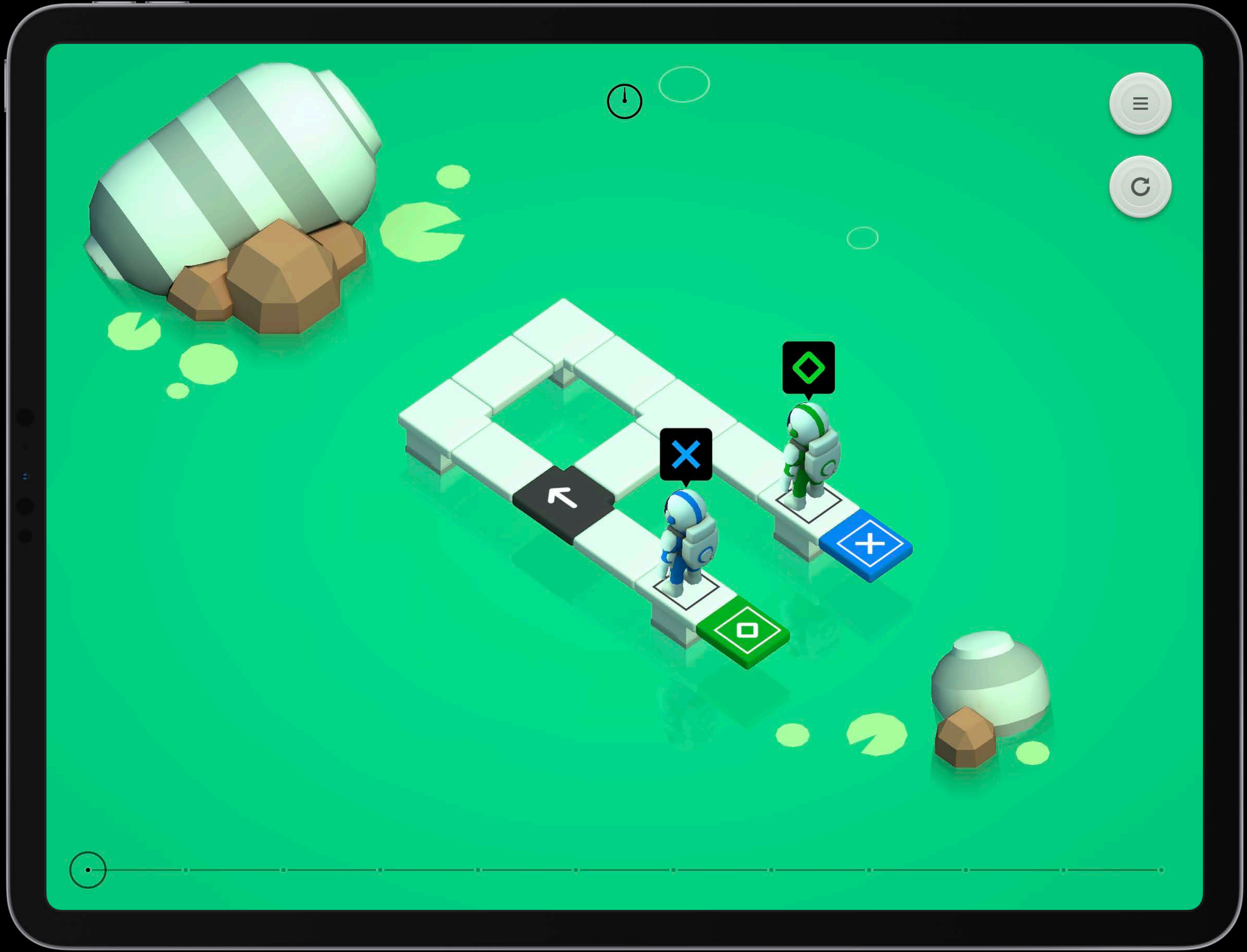
All Natural



When possible,
do not rely on color alone.







Summary

Summary

Dynamic Type and custom fonts

Summary

Dynamic Type and custom fonts

New motion settings

Summary

Dynamic Type and custom fonts

New motion settings

Differentiate without color

#WWDC19

Accessibility Inspector

Gary Chen, Accessibility Engineer



More Information

developer.apple.com/wwdc19/244

Accessibility Engineering Lab

Tuesday, 10:00
Friday, 11:00

Accessibility Design Lab

Daily, 9:00

Accessibility in SwiftUI

Friday, 10:00

