

#WWDC19

Debugging in Xcode 11

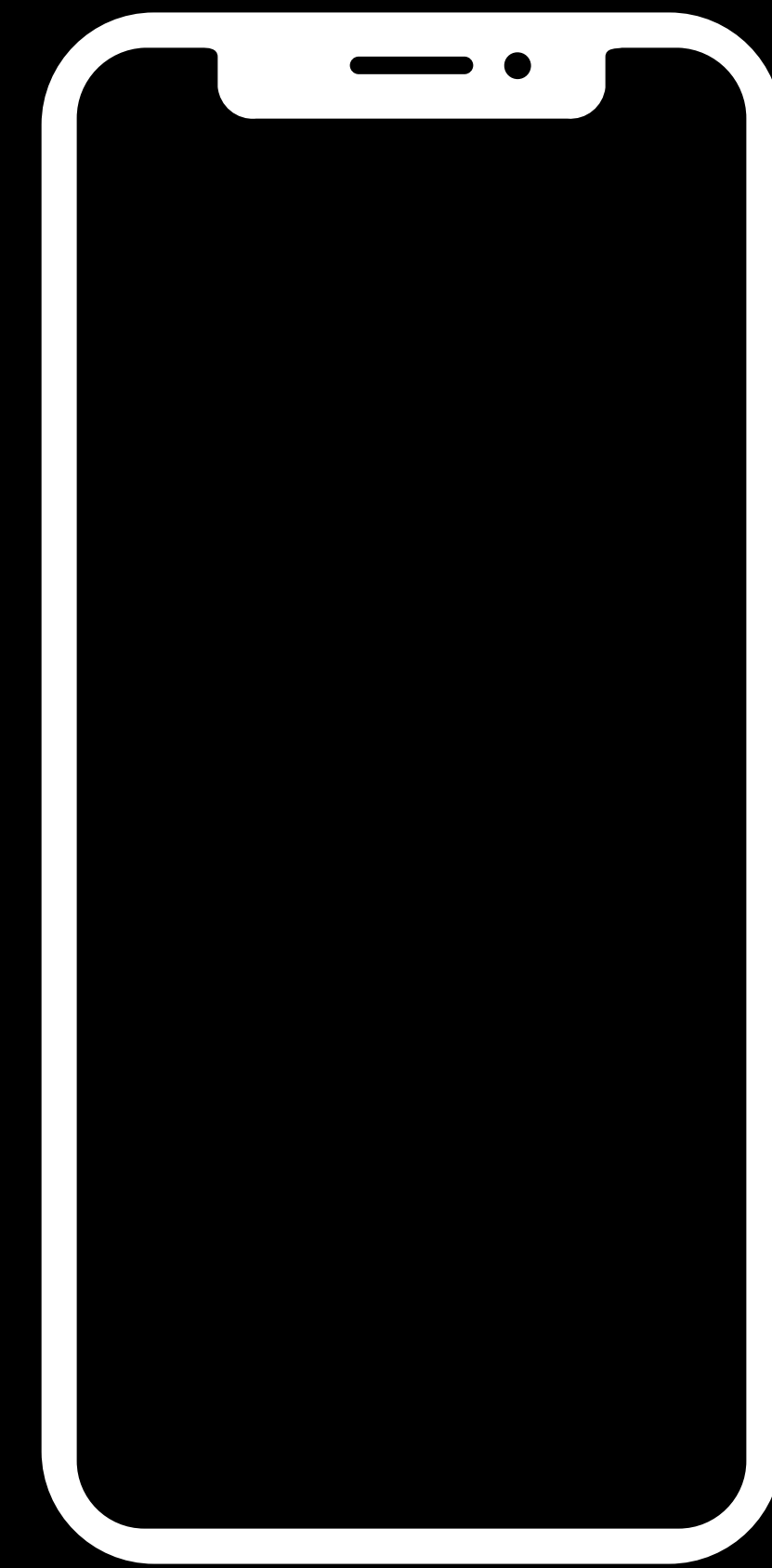
Chris Miles, Xcode Engineering Manager

Han Ming Ong, Xcode UI Engineer

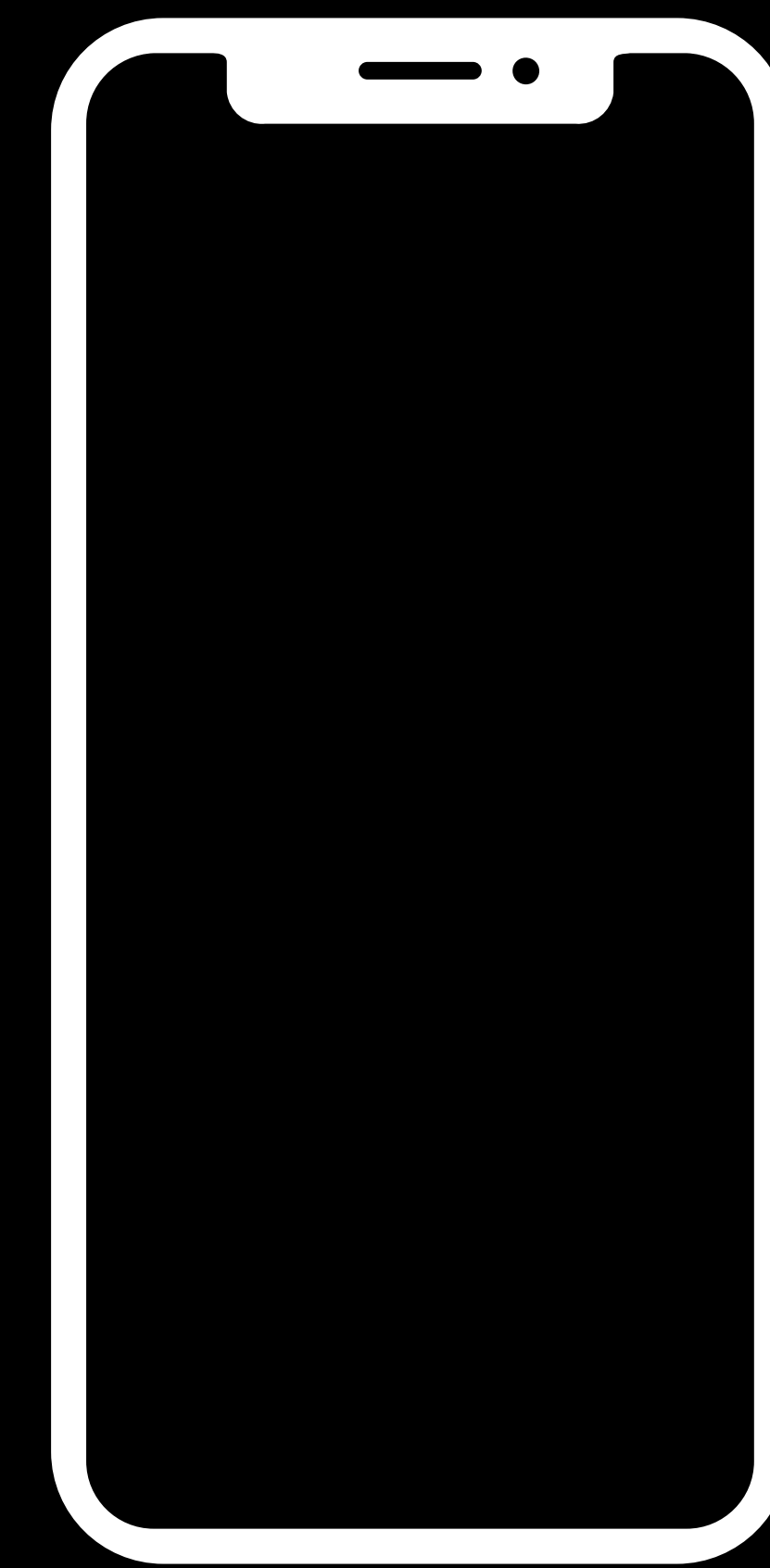
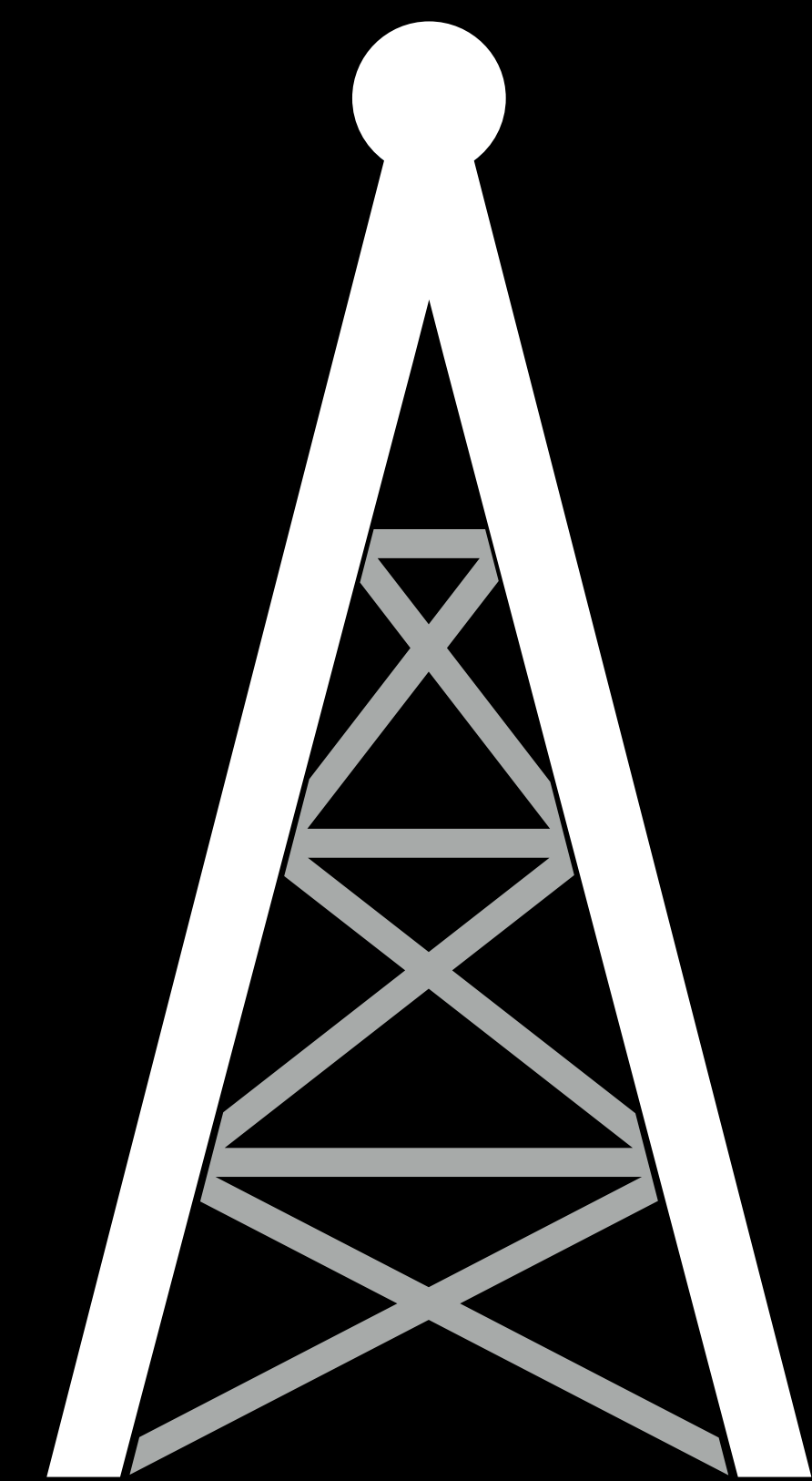
Sebastian Fischer, Xcode UI Engineer

Device Conditions and Environment Overrides

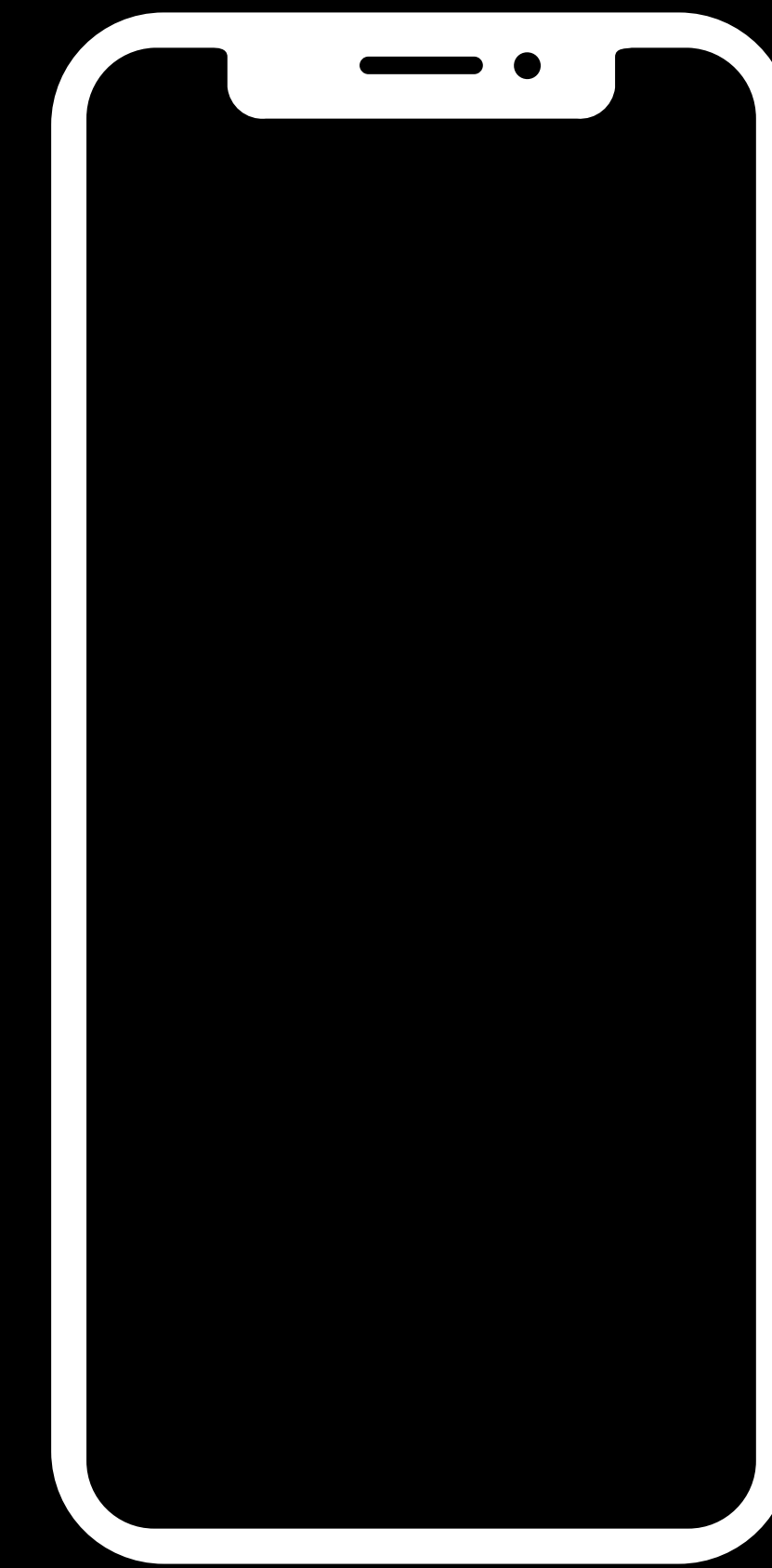
Device Conditions and Environment Overrides



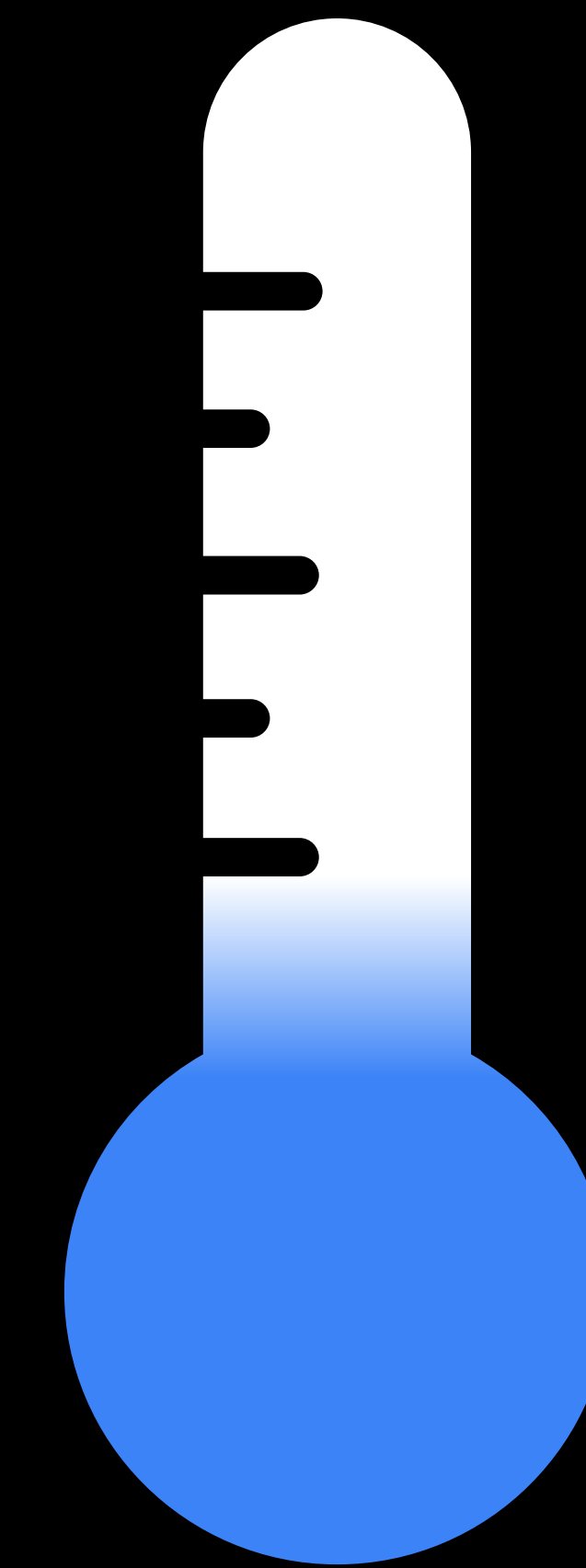
Device Conditions and Environment Overrides



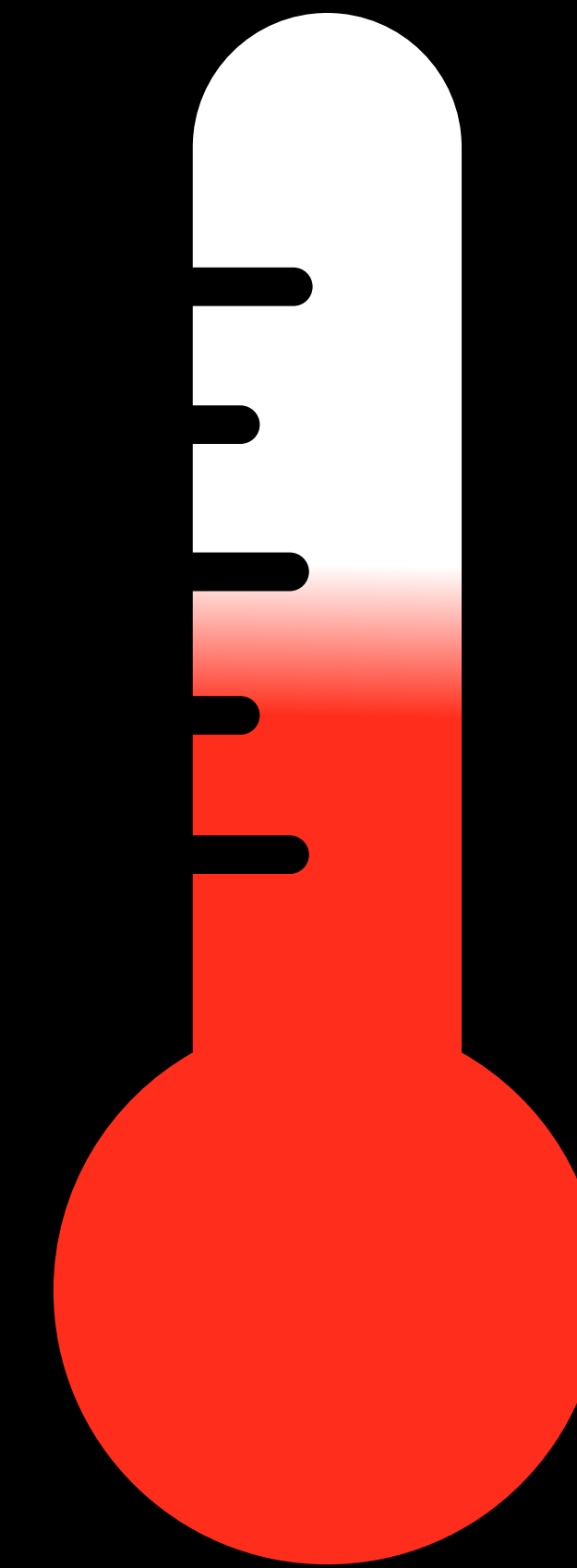
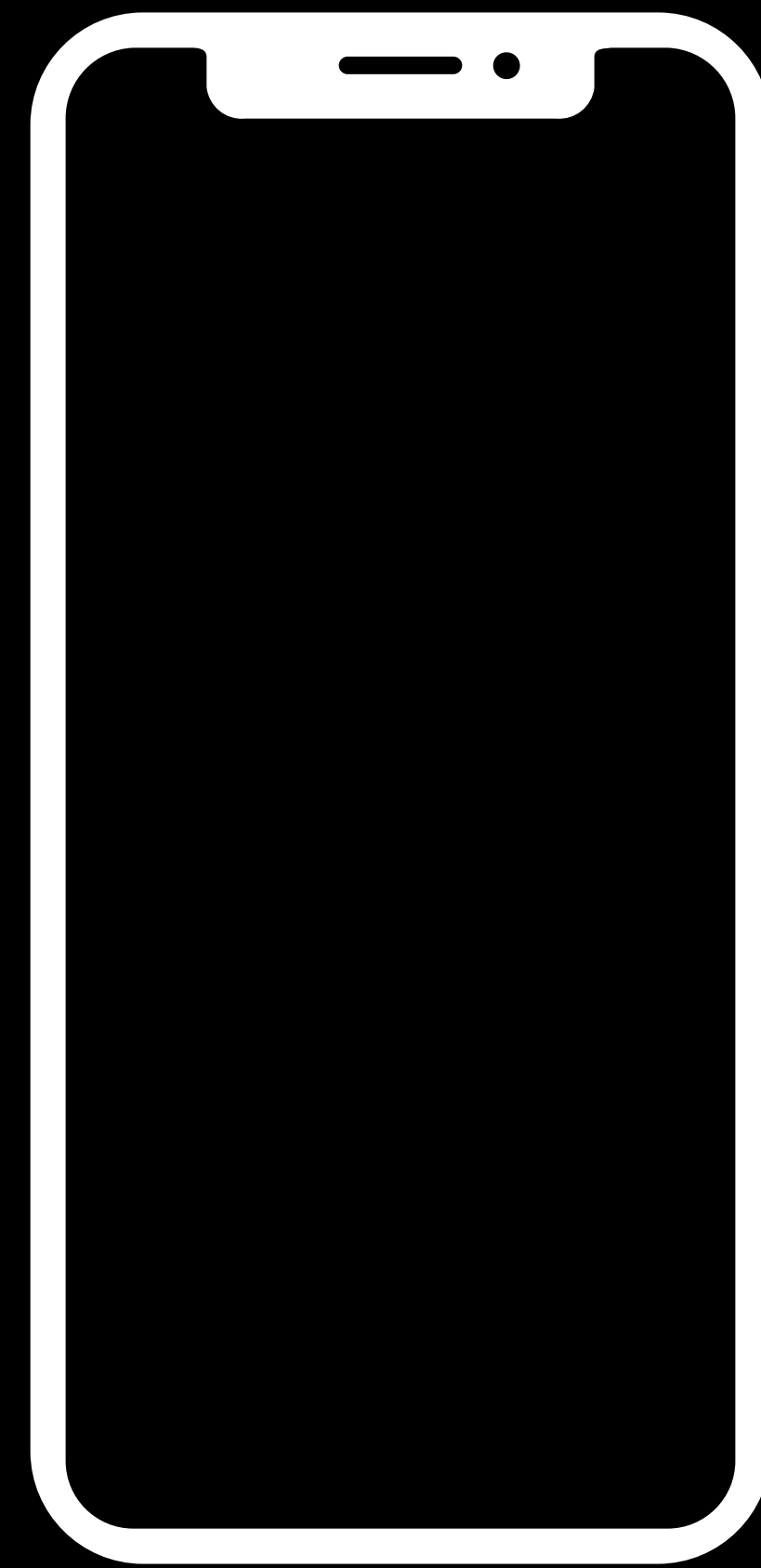
Device Conditions and Environment Overrides



Device Conditions and Environment Overrides



Device Conditions and Environment Overrides



Device Conditions

Thermal state condition

Raises thermal state to — fair, serious, or critical

So you can reliably

- Understand app behavior under these elevated states
- Debug and verify your handling of thermal state changes

The device does not actually get warmer

Device Conditions

Network link condition

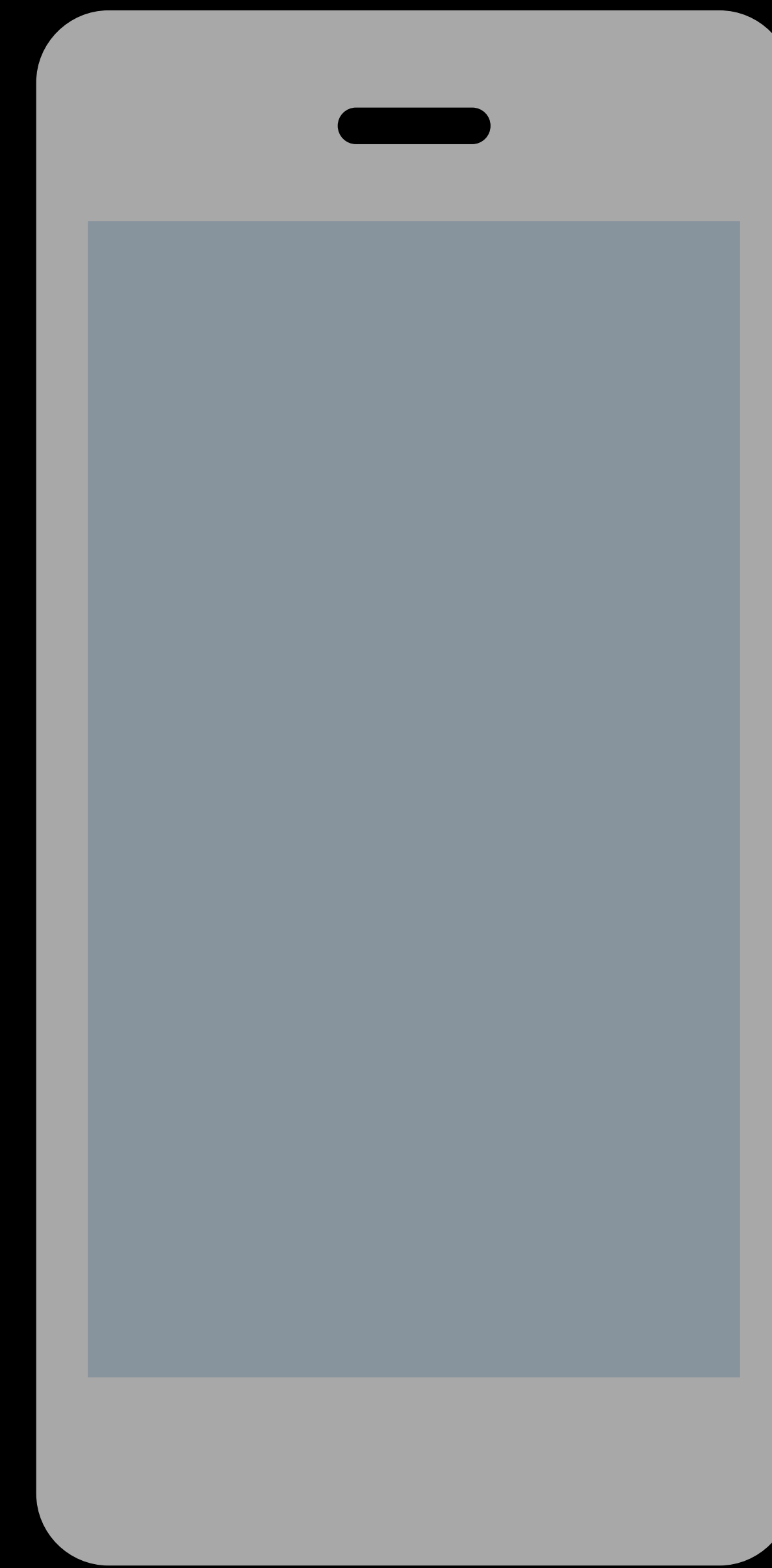
Real world profiles simulating network conditions

- High latency
- Low bandwidth
- High packet loss
- Slow DNS response

So you can reliably

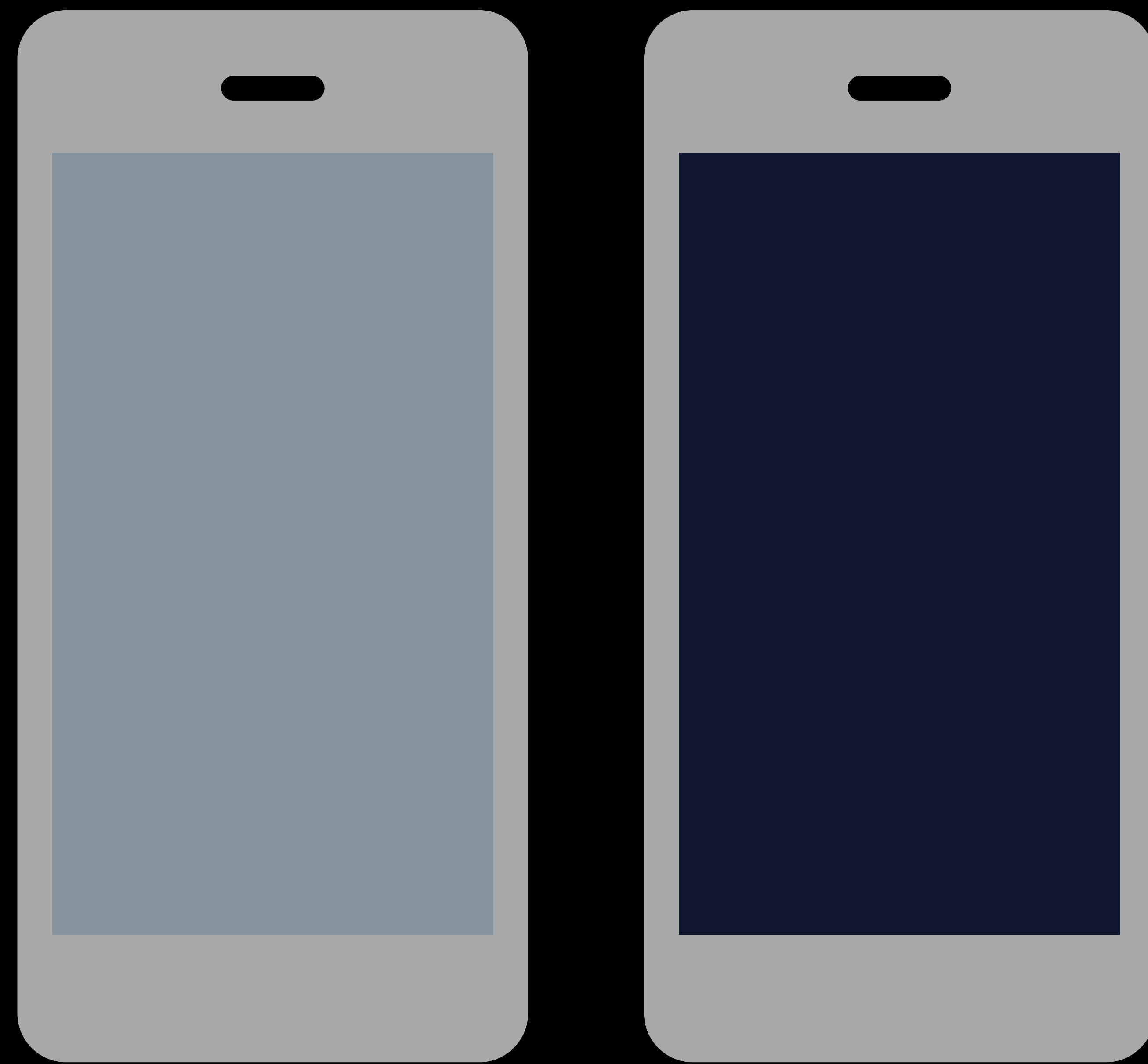
- Understand and debug app behavior under various network conditions

Environment Overrides



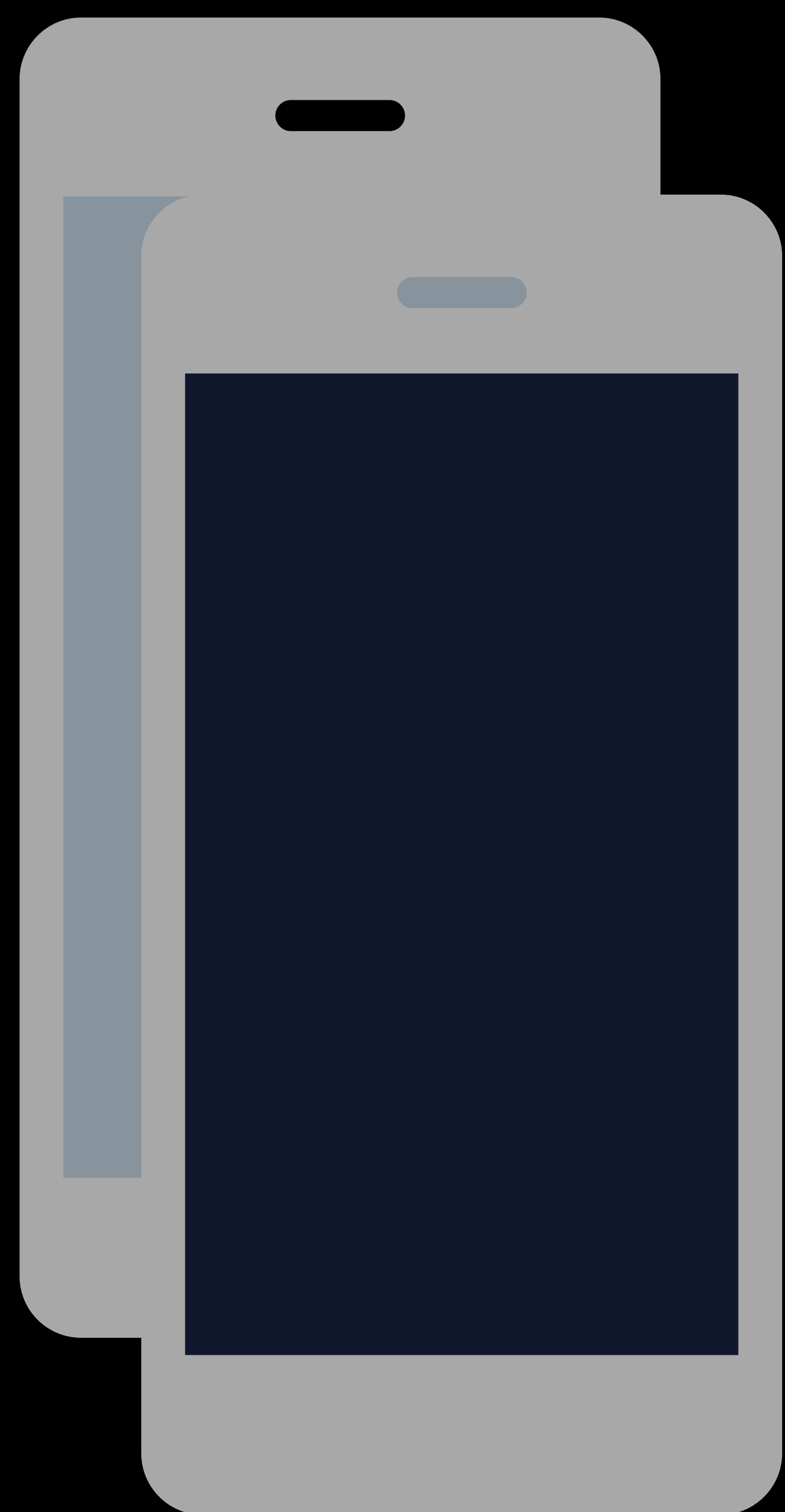
Interface Style

Environment Overrides

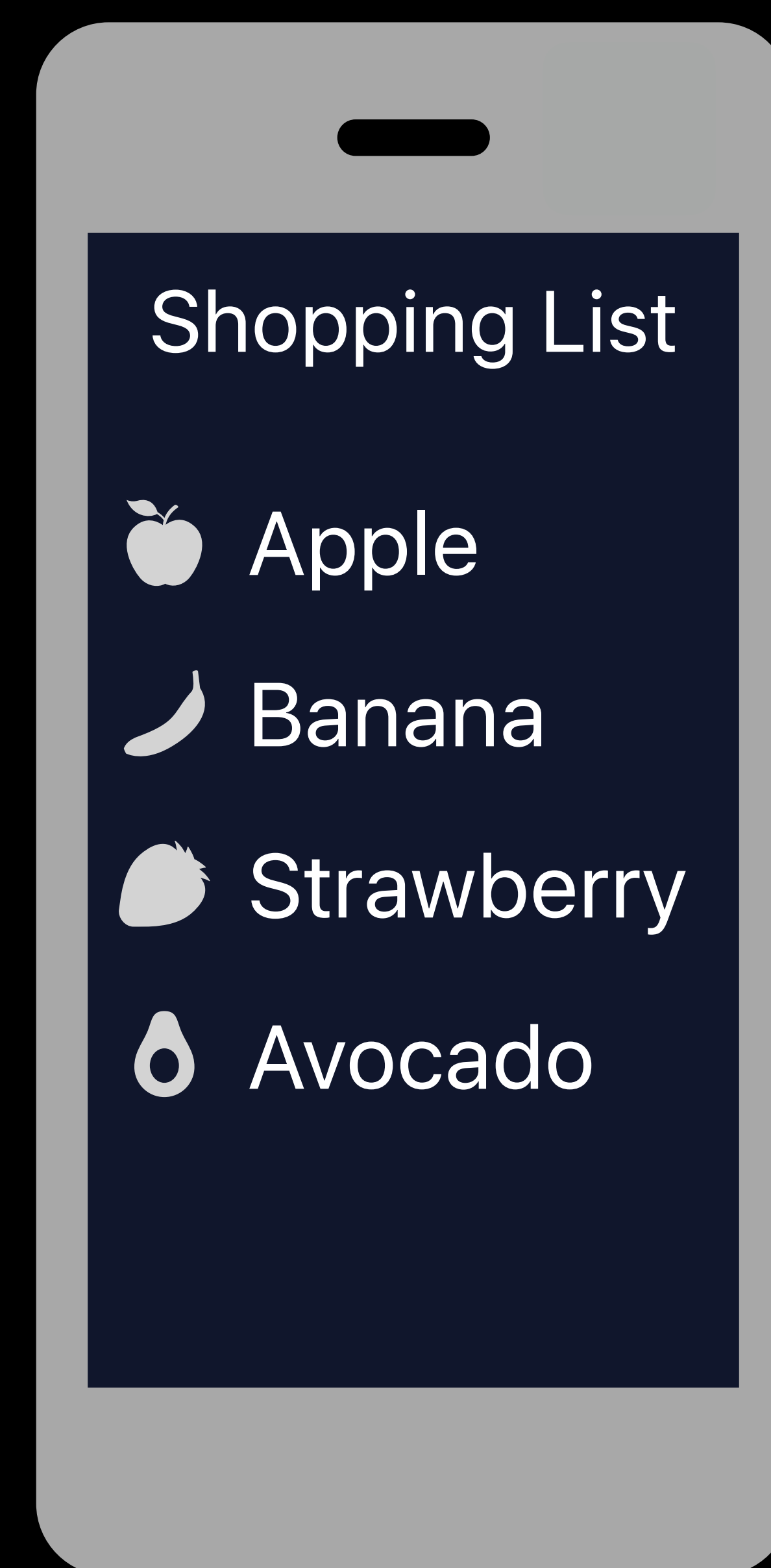


Interface Style

Environment Overrides

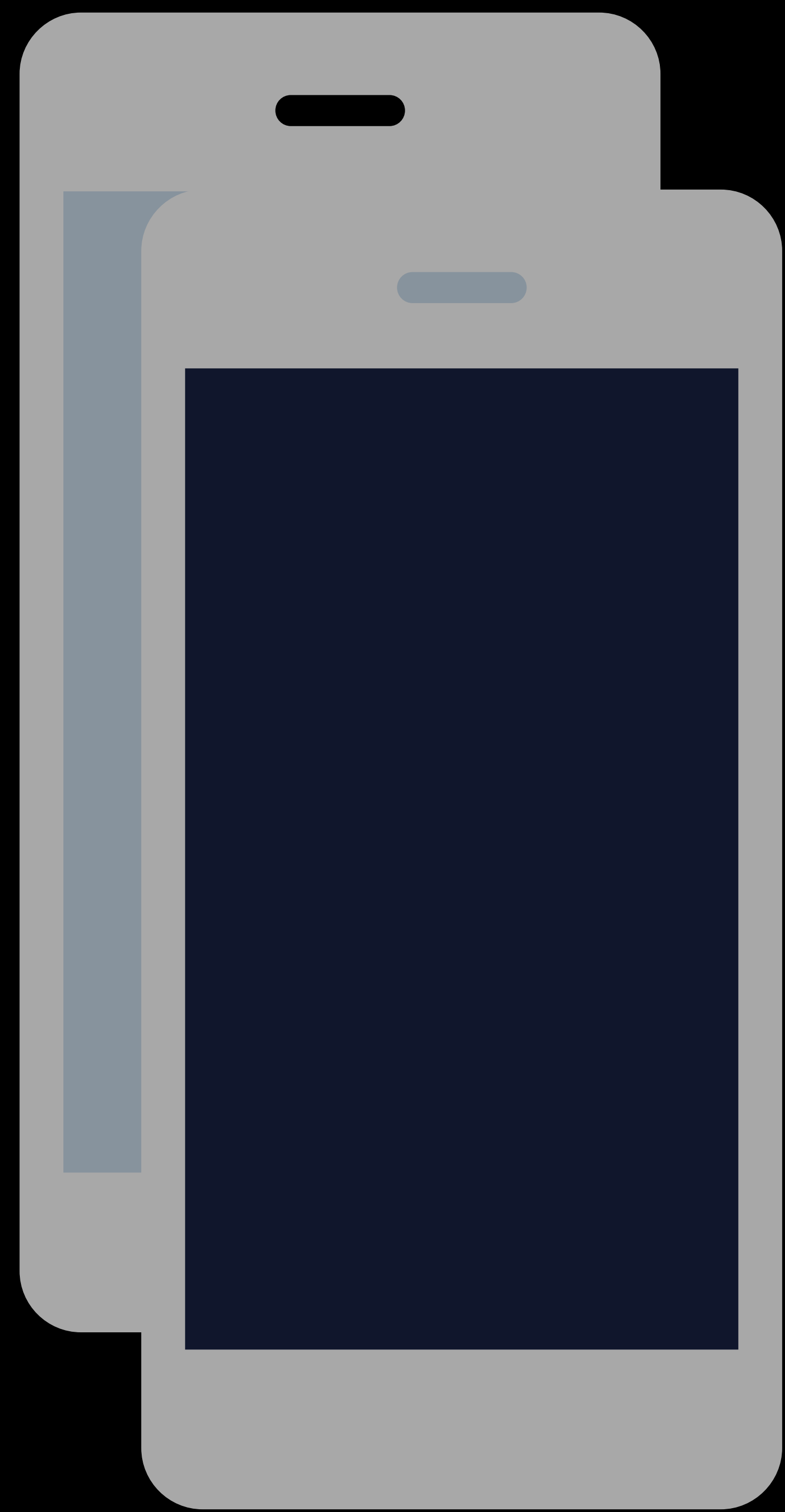


Interface Style



Dynamic Type

Environment Overrides

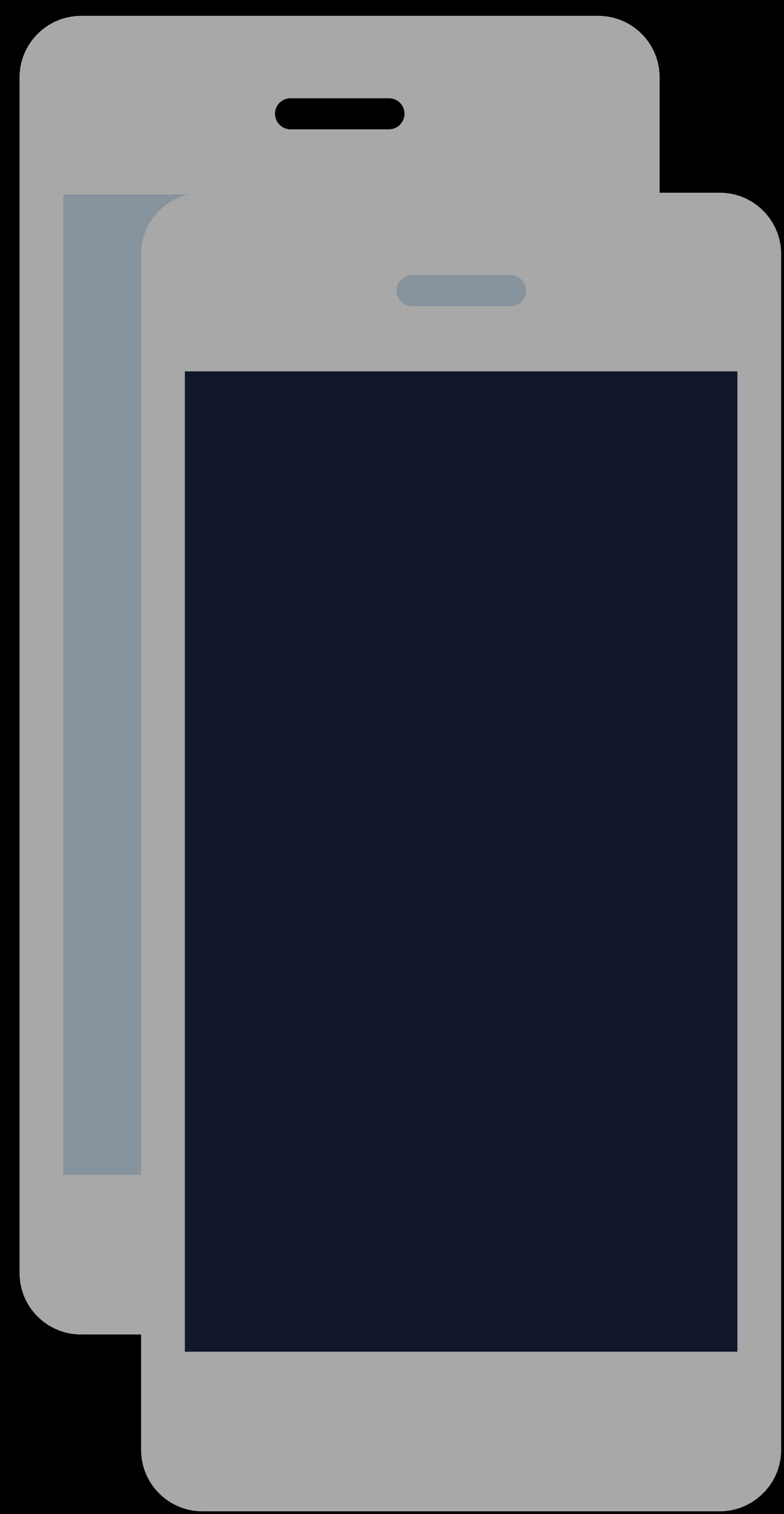


Interface Style

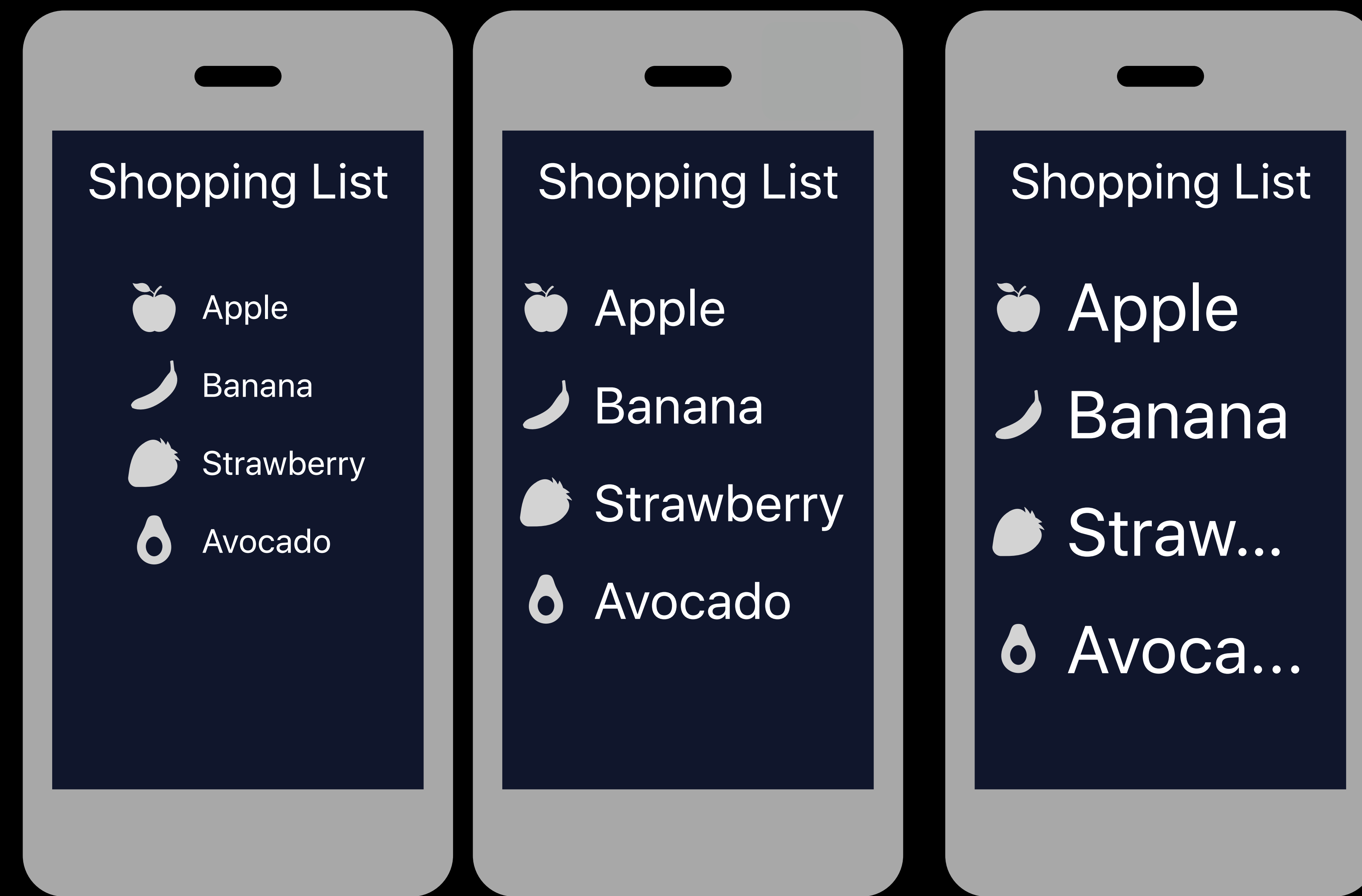


Dynamic Type

Environment Overrides

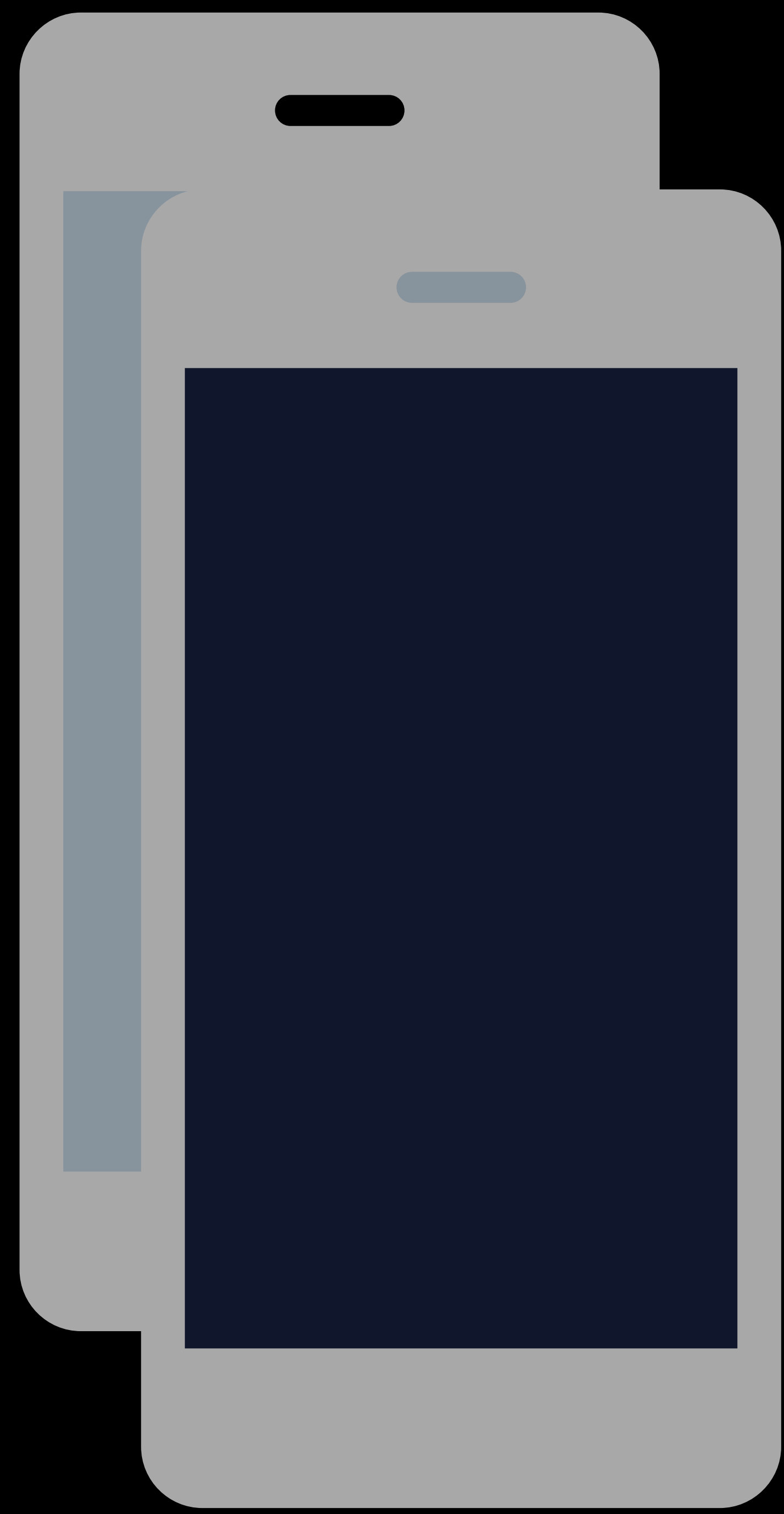


Interface Style

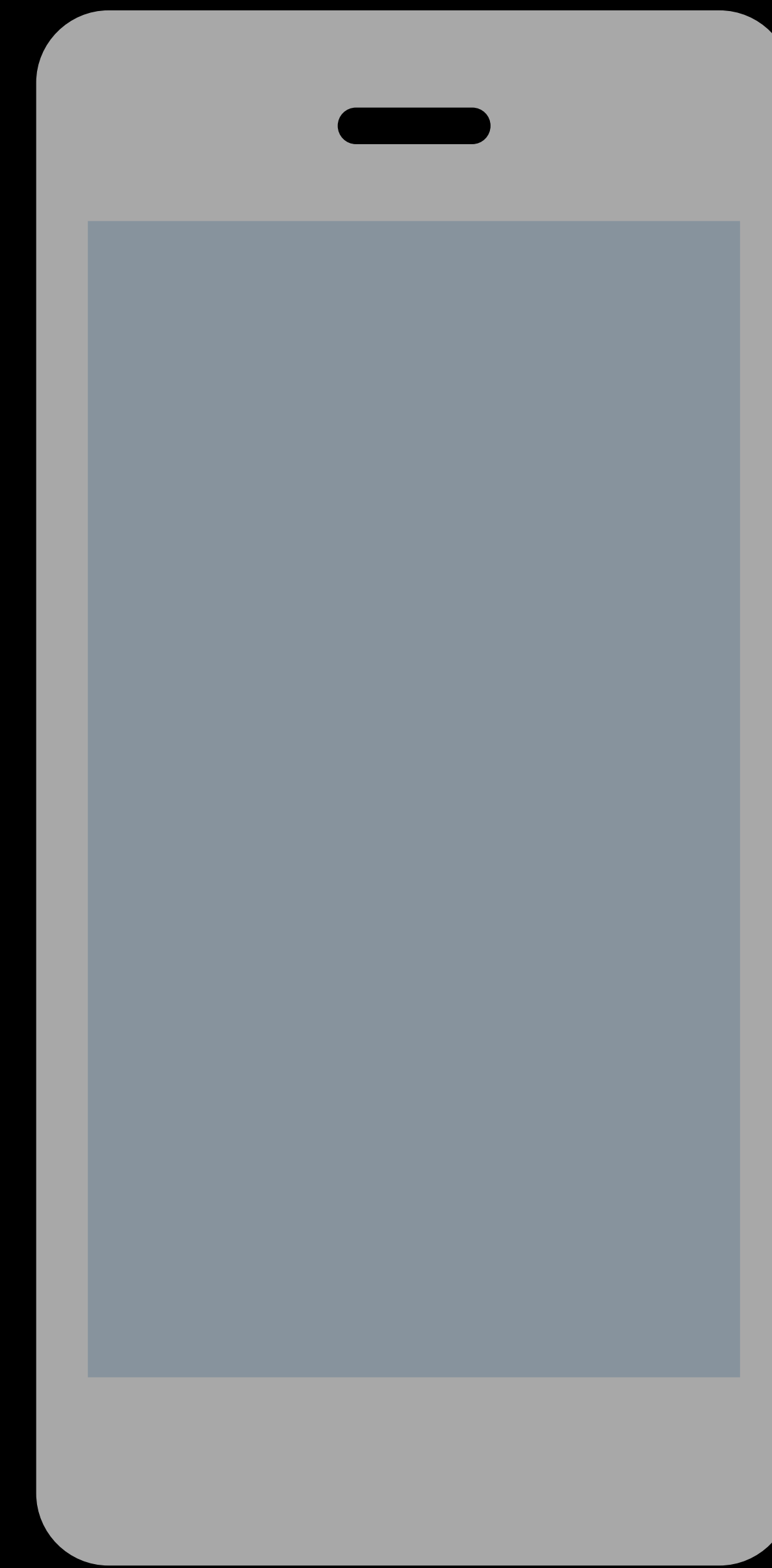


Dynamic Type

Environment Overrides



Interface Style

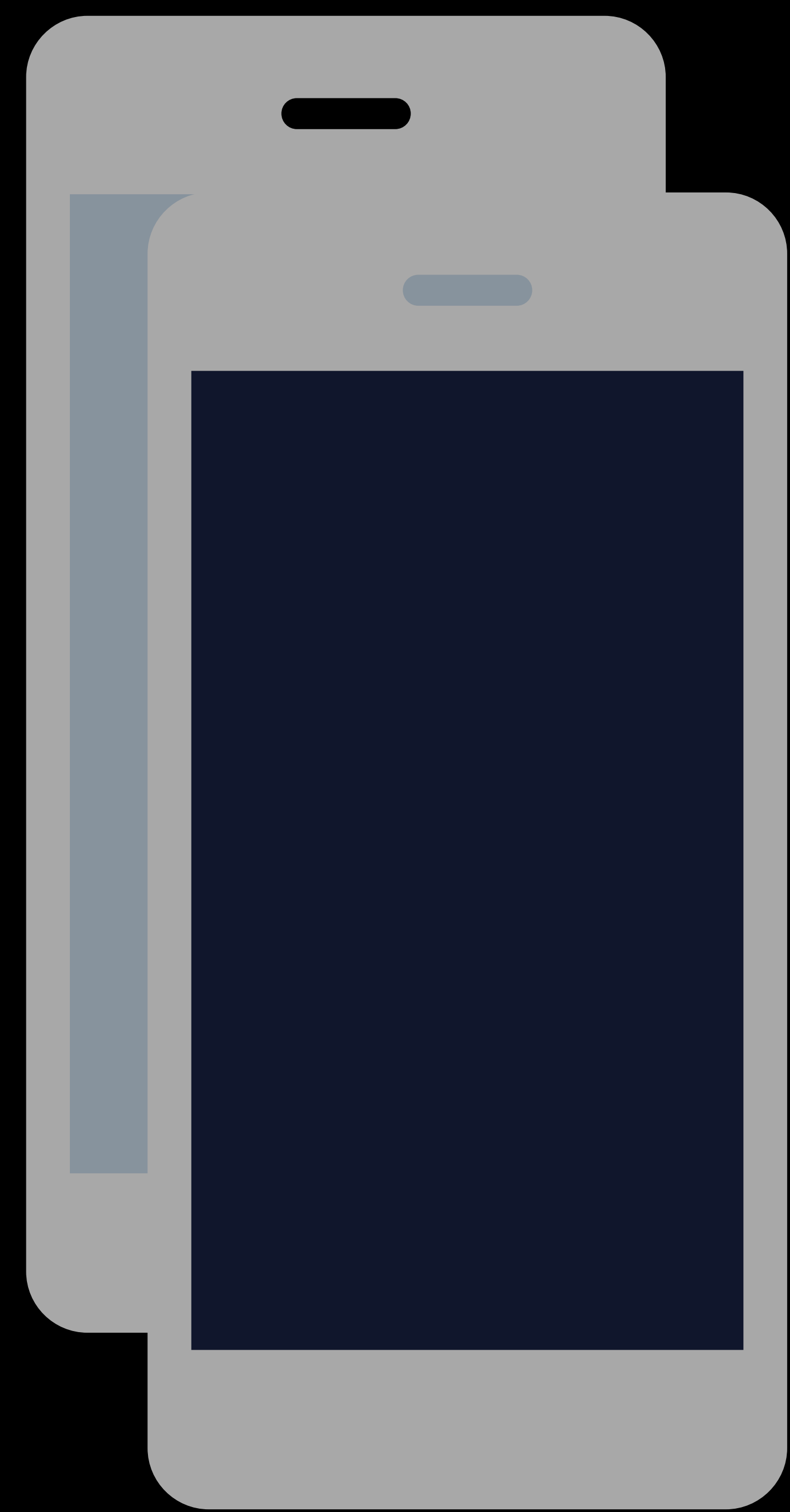


Accessibility Options

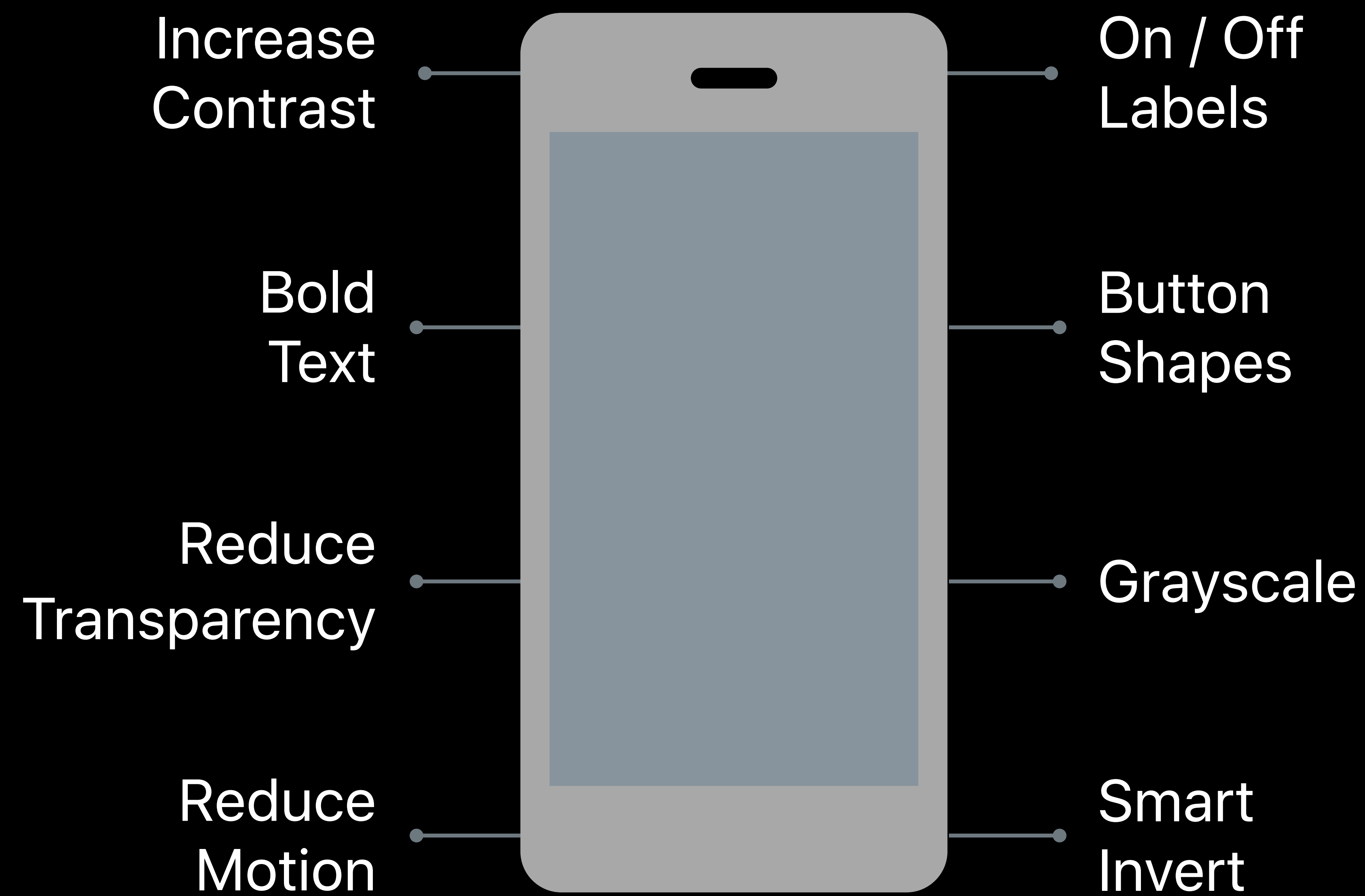


Dynamic Type

Environment Overrides



Interface Style



Accessibility Options



Dynamic Type

Demo

Device conditions and environment overrides

Chris Miles, Xcode Engineering Manager

Device Conditions

The screenshot shows the Xcode interface with the 'Devices' tab selected. On the left, a sidebar shows a connected 'iPhone' device. The main area is divided into two sections: 'INSTALLED APPS' and 'DEVICE CONDITIONS'. The 'INSTALLED APPS' section is currently empty, displaying 'No apps installed'. The 'DEVICE CONDITIONS' section is active, showing a 'Condition' dropdown set to 'Network Link' and a 'Profile' dropdown set to '100% packet loss'. Below these, the specific network parameters are listed: Name: 100% Loss Scenario, Downlink Bandwidth: 0 Mbps, Downlink Latency: 0 ms, Downlink Packet Loss Ratio: 100%, Uplink Bandwidth: 0 Mbps, Uplink Latency: 0 ms, and Uplink Packet Loss Ratio: 100%. A 'Start' button is located at the bottom of this section. At the bottom left of the main area, there is a '+ Filter' button.

Connected

iPhone

Devices Simulators

INSTALLED APPS

Name	Version	Identifier
No apps installed		

+ - ⚙️

DEVICE CONDITIONS

Condition Network Link

Profile 100% packet loss

Name: 100% Loss Scenario
Downlink Bandwidth: 0 Mbps
Downlink Latency: 0 ms
Downlink Packet Loss Ratio: 100%
Uplink Bandwidth: 0 Mbps
Uplink Latency: 0 ms
Uplink Packet Loss Ratio: 100%

Start

+ Filter

Device Conditions

The screenshot shows the Xcode interface with the 'Devices' tab selected. On the left, a sidebar shows 'Connected' devices, with an iPhone selected. The main area is titled 'INSTALLED APPS' and shows a table with columns for Name, Version, and Identifier. Below this, there are controls for adding (+) and removing (-) apps, and a settings gear icon. The 'DEVICE CONDITIONS' section is visible, showing a dropdown menu for 'Condition' with options: 'None', 'Network Link' (checked), and 'Thermal State'. Below the dropdown, the details for the '100% Loss Scenario' are listed: Name: 100% Loss Scenario, Downlink Bandwidth: 0 Mbps, Downlink Latency: 0 ms, Downlink Packet Loss Ratio: 100%, Uplink Bandwidth: 0 Mbps, Uplink Latency: 0 ms, and Uplink Packet Loss Ratio: 100%. A 'Start' button is located at the bottom of this section. At the bottom left of the main area, there is a '+ Filter' button.

Connected

iPhone

Devices Simulators

INSTALLED APPS

Name	Version	Identifier
No apps installed		

+ - ⚙️

DEVICE CONDITIONS

Condition: **Network Link** Thermal State

Profile: 100% Loss Scenario

Name: 100% Loss Scenario
Downlink Bandwidth: 0 Mbps
Downlink Latency: 0 ms
Downlink Packet Loss Ratio: 100%
Uplink Bandwidth: 0 Mbps
Uplink Latency: 0 ms
Uplink Packet Loss Ratio: 100%

Start

+ Filter

Device Conditions

Device-wide induced condition

Managed by Xcode

Tied to Xcode device connection

iOS status bar indicates when a condition is active

Can be stopped on device



Device Conditions

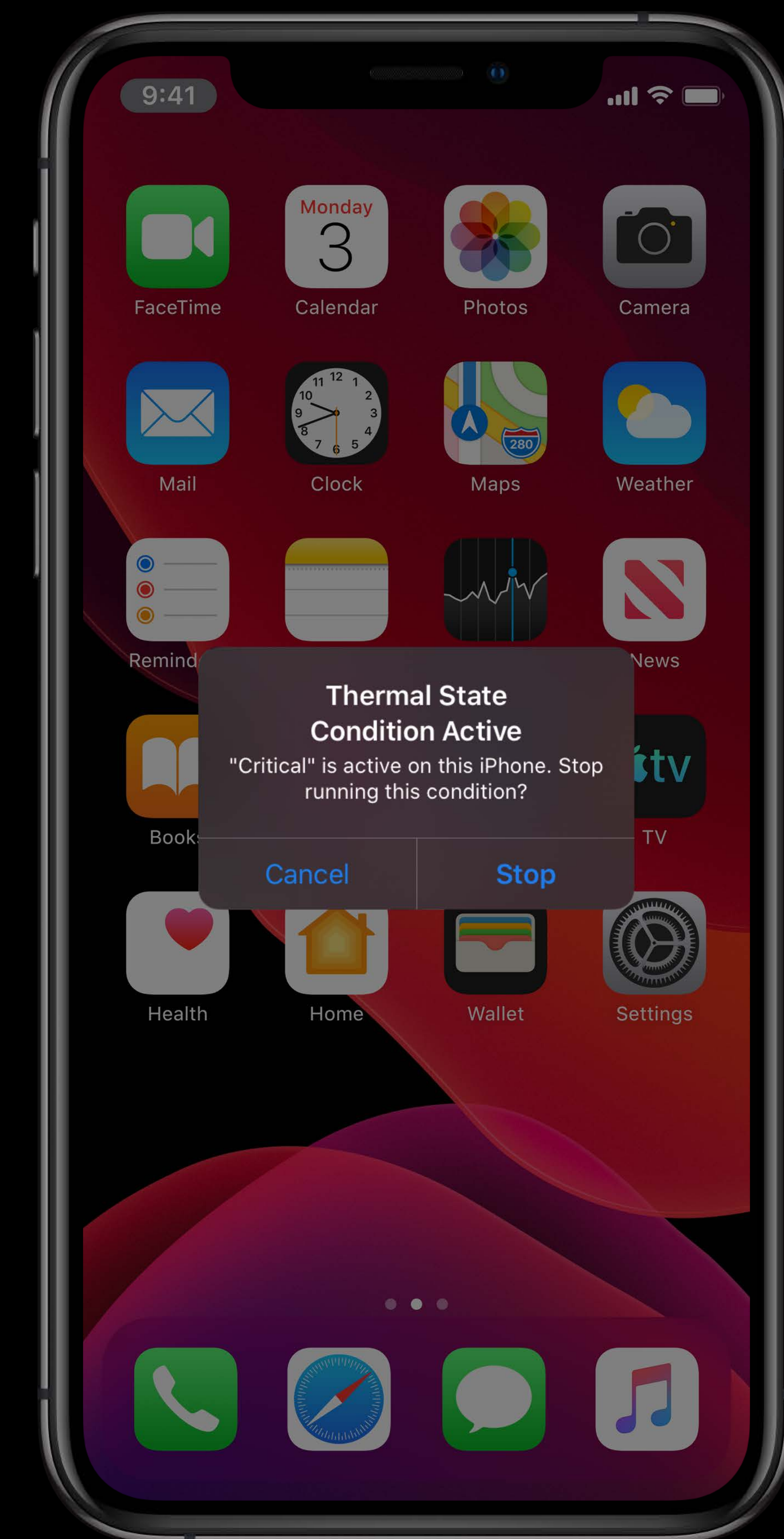
Device-wide induced condition

Managed by Xcode

Tied to Xcode device connection

iOS status bar indicates when a condition is active

Can be stopped on device



Environment Overrides

Override runtime environment of debugged app

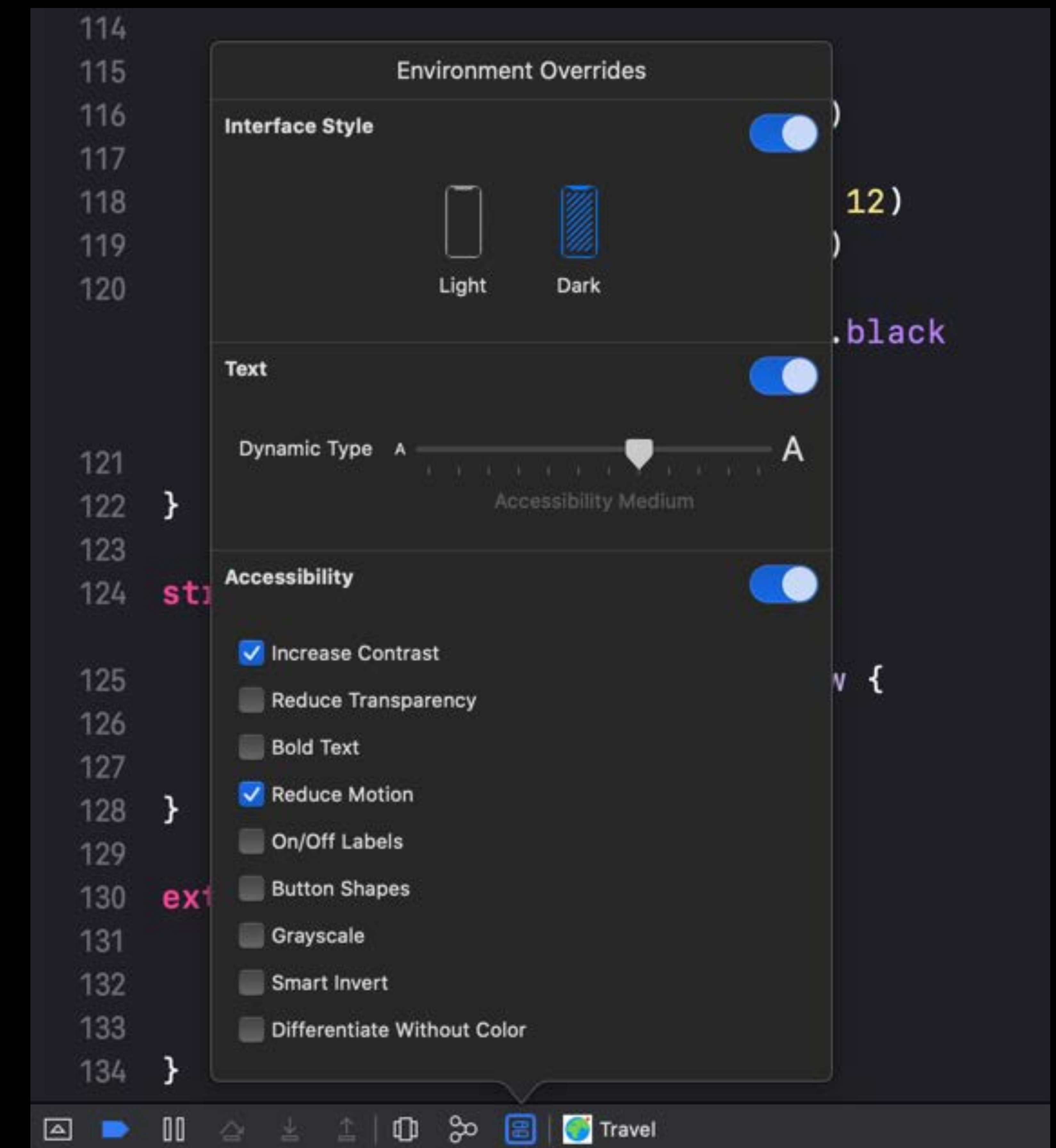
Overrides instantly

Only affects debugged app

System settings remain unchanged

Supports devices, simulators, and previews

Available for all platforms



Debugging Live Previews

Han Ming Ong, Xcode UI Engineer

Travel | Preview Travel: Succeeded | Today at 12:07 PM

Travel > Travel > Plan > PlanDetailView.swift > displayables(for:)

Preview

- CPU 0%
- Memory 6.7 MB
- Disk Zero KB/s
- Network Zero KB/s
- Thread 1 Queue: com.a...thread (serial)
 - 0 PlanDetailView.__preview_displ...
 - 1 PlanDetailView.displayables(for:)
 - 2 PlanDetailView.init(_:sceneCont...
 - 4 static PlanDetailView.Previews...
 - 5 static PlanDetailView.Previews...
 - 25 UIApplicationMain
 - 26 main
 - 27 start
- Thread 2
- Thread 3
- Thread 4
- Thread 5
- com.apple.uikit.eventfetch-thread (6)

```

137     }
138 }
139
140 func displayables(for plan: Plan) -> (displayables: [Displayable],
141     headerDisplayable: Displayable) {
142     let headerDisplayable = HeaderDisplayable(plan: plan)
143     var displayables: [Displayable] = []
144     let events = plan.events
145
146     for i in 0..

```

Preview > Thread 1 > 0 PlanDetailView.__preview_displayables(for:)

- plan = (Travel.Plan) 0x0000600003642d80
- self (Travel.PlanDetailView)
- displayables = ([Travel.Displayable]) 0 values
- headerDisplayable = (Travel.PlanDetailView.HeaderDisplayable) 0x0000600002351440
- events = ([Travel.Event]) 8 values
- i = (Int) 0
- event (Travel.Event)
- dayChanged = (Bool) false
- newSunAltitude = (Travel.SolarAltitude?) sunrise

(lldb) po event

```

Event
- dateString : "14/06/2019 9:30"
- poiID : "sf-poi-1"
- userEvent : nil
- userPhotos : 6 elements
  0 : Photo
    - imageName : "new-york"
    - title : "some title"
  1 : Photo

```


Travel | Preview Travel: Succeeded | Today at 12:07 PM

Travel > Travel > Plan > PlanDetailView.swift > displayables(for:)

Preview

- CPU 0%
- Memory 6.7 MB
- Disk Zero KB/s
- Network Zero KB/s

Thread 1 Queue: com.a...thread (serial)

- 0 PlanDetailView.__preview_displ...
- 1 PlanDetailView.displayables(for:)
- 2 PlanDetailView.init(_:sceneCont...
- 4 static PlanDetailView_Previews....
- 5 static PlanDetailView_Previews....
- 25 UIApplicationMain
- 26 main
- 27 start
- Thread 2
- Thread 3
- Thread 4
- Thread 5
- com.apple.uikit.eventfetch-thread (6)

```

}
}
func displayables(for plan: Plan) -> (displayables: [Displayable],
headerDisplayable: Displayable) {
    let headerDisplayable = HeaderDisplayable(plan: plan)
    var displayables: [Displayable] = []
    let events = plan.events

    ..<events.count {
        event
        ent = events[i]

        always "changed" for the first item
        yChanged = true
        wSunAltitude: SolarAltitude? = nil

        previous event if applicable
        previousEvent = (i == 0) ? nil : events[i-1] {
            yChanged = !Calendar.current.isDate(event.date,
            inSameDayAs:previousEvent.date)

            !dayChanged {
                newSunAltitude =
                SolarAltitude.betweenDates(previousEvent.date,
                event.date)

            wSunAltitude = .sunrise

            displayables
            yChanged) {
                i != 0 {

```

19 days to go!

June 14

Sunrise

- Golden Gate Bridge 9:30 AM
- Golden Gate Park 11:30 AM

Noon

- China Town 1:00 PM

plan = (Travel.Plan) 0x0000600003642d80

self (Travel.PlanDetailView)

displayables = ([Travel.Displayable]) 0 values

headerDisplayable = (Travel.PlanDetailView.HeaderDisplayable) 0x0000600002351440

events = ([Travel.Event]) 8 values

i = (Int) 0

event (Travel.Event)

dayChanged = (Bool) false

newSunAltitude = (Travel.SolarAltitude?) sunrise

(lldb) po event

```

Event
- dateString : "14/06/2019 9:30"
- poiID : "sf-poi-1"
- userEvent : nil
- userPhotos : 6 elements
  0 : Photo
    - imageName : "new-york"
    - title : "some title"
  1 : Photo

```

Travel | Preview Travel: Succeeded | Today at 12:07 PM

Travel > Travel > Plan > PlanDetailView.swift > displayables(for:)

Preview

- CPU 0%
- Memory 6.7 MB
- Disk Zero KB/s
- Network Zero KB/s
- Thread 1 Queue: com.a...thread (serial)
- 0 PlanDetailView.__preview_displ...

```

137     }
138 }
139
140 func displayables(for plan: Plan) -> (displayables: [Displayable],
141     headerDisplayable: Displayable) {
142     let headerDisplayable = HeaderDisplayable(plan: plan)
143     var displayables: [Displayable] = []
144     let events = plan.events
145
146     for i in 0..

Thread 1: breakpoint 1.2



19 days to go!



Golden Gate Bridge 9:30 AM



Golden Gate Park 11:30 AM



Noon



China Town 1:00 PM



100%



Preview > Thread 1 > 0 PlanDetailView.__preview_displayables(for:)



- plan = (Travel.Plan) 0x0000600003642d80
- self (Travel.PlanDetailView)
- displayables = ([Travel.Displayable]) 0 values
- headerDisplayable = (Travel.PlanDetailView.HeaderDisplayable) 0x0000600002351440
- events = ([Travel.Event]) 8 values
- i = (Int) 0
- event (Travel.Event)
- dayChanged = (Bool) false
- newSunAltitude = (Travel.SolarAltitude?) sunrise



(lldb) po event



```

Event
- dateString : "14/06/2019 9:30"
- poiID : "sf-poi-1"
- userEvent : nil
- userPhotos : 6 elements
 0 : Photo
 - imageName : "new-york"
 - title : "some title"
 1 : Photo

```


```

Travel | Preview Travel: Succeeded | Today at 12:07 PM

Travel > Travel > Plan > PlanDetailView.swift > displayables(for:)

Preview

- CPU 0%
- Memory 6.7 MB
- Disk Zero KB/s
- Network Zero KB/s
- Thread 1 Queue: com.a...thread (serial)
 - 0 PlanDetailView.__preview_displ...
 - 1 PlanDetailView.displayables(for:)
 - 2 PlanDetailView.init(_:sceneCont...
 - 4 static PlanDetailView_Previews_...
 - 5 static PlanDetailView_Previews_...
 - 25 UIApplicationMain
 - 26 main
 - 27 start
- Thread 2
- Thread 3
- Thread 4
- Thread 5
- com.apple.uitk.eventfetch-thread (6)

```

137     }
138 }
139
140 func displayables(for plan: Plan) -> (displayables: [Displayable],
141     headerDisplayable: Displayable) {
142     let headerDisplayable = HeaderDisplayable(plan: plan)
143     var displayables: [Displayable] = []
144     let events = plan.events
145
146     for i in 0..

```

Thread 1 > 0 PlanDetailView.__preview_displayables(for:)

plan = (Travel.Plan) 0x0000600003042080

self (Travel.PlanDetailView)

displayables = ([Travel.Displayable]) 0 values

headerDisplayable = (Travel.PlanDetailView.HeaderDisplayable) 0x0000600002351440

events = ([Travel.Event]) 8 values

i = (Int) 0

event (Travel.Event)

dayChanged = (Bool) false

newSunAltitude = (Travel.SolarAltitude?) sunrise

Event

- dateString : "14/06/2019 9:30"
- poiID : "sf-poi-1"
- userEvent : nil
- userPhotos : 6 elements
 - 0 : Photo
 - imageName : "new-york"
 - title : "some title"
 - 1 : Photo

Filter | Auto | Filter | All Output | Filter

Travel | Preview Travel: Succeeded | Today at 12:07 PM

Travel > Travel > Plan > PlanDetailView.swift > displayables(for:)

Preview

- CPU 0%
- Memory 6.7 MB
- Disk Zero KB/s
- Network Zero KB/s
- Thread 1 Queue: com.a...thread (serial)
 - 0 PlanDetailView.__preview_displ...
 - 1 PlanDetailView.displayables(for:)
 - 2 PlanDetailView.init(_:sceneCont...
 - 4 static PlanDetailView_Previews_...
 - 5 static PlanDetailView_Previews_...
 - 25 UIApplicationMain
 - 26 main
 - 27 start
- Thread 2
- Thread 3
- Thread 4
- Thread 5
- com.apple.uikit.eventfetch-thread (6)

```
137     }
138   }
139
140   func displayables(for plan: Plan) -> (displayables: [Displayable],
141     headerDisplayable: Displayable) {
142     let headerDisplayable = HeaderDisplayable(plan: plan)
143     var displayables: [Displayable] = []
144     let events = plan.events
145
146     for i in 0..

Thread 1: breakpoint 1.2



19 days to go!



June 14



Sunrise



Golden Gate Bridge 9:30 AM



Golden Gate Park 11:30 AM



Noon 1:00 PM



100%



plan = (Travel.Plan) 0x0000600003642d80



self (Travel.PlanDetailView)



displayables = ([Travel.Displayable]) 0 values



headerDisplayable = (Travel.PlanDetailView.HeaderDisplayable) 0x0000600002351440



events = ([Travel.Event]) 8 values



i = (Int) 0



event (Travel.Event)



dayChanged = (Bool) false



newSunAltitude = (Travel.SolarAltitude?) sunrise




Filter


```


SwiftUI Runtime Issues

SwiftUI Runtime Issues


▼  Modifying state during view update, this will cause undefined behavior.


▼  Thread 1

 0 StoredLocation.set(_:transa...

 1 StateLink.value.setter

 2 DiscoverView.__preview_bo...

 3 DiscoverView.body()










 38 UIApplicationMain

 39 main

 40 start

SwiftUI Runtime Issues










Found when process is running

- ▼  Modifying state during view update, this will cause undefined behavior.
- ▼  Thread 1
 -  0 StoredLocation.set(_:transa...
 -  1 StateLink.value.setter
 -  2 DiscoverView.__preview__bo...
 -  3 DiscoverView.body()
 -
 -  38 UIApplicationMain
 -  39 main
 -  40 start

SwiftUI Runtime Issues

Found when process is running

Process continues executing

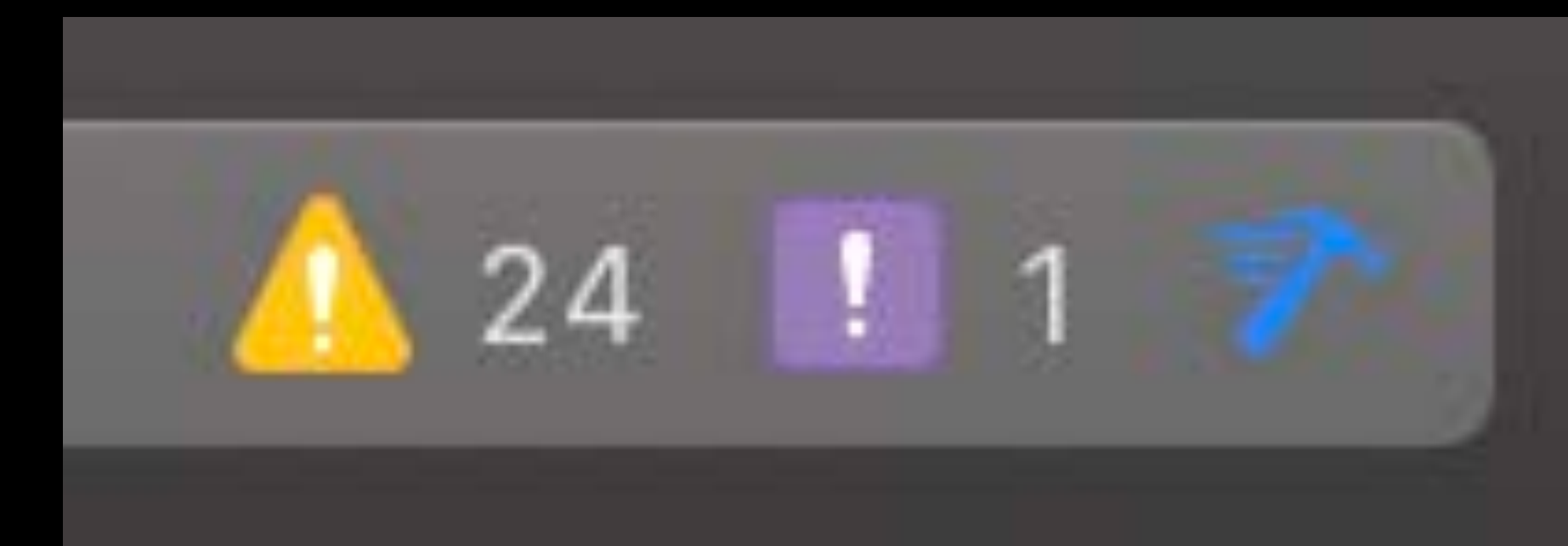
- ▼  **Modifying state during view update, this will cause undefined behavior.**
- ▼  **Thread 1**
 -  0 StoredLocation.set(_:transa...
 -  1 StateLink.value.setter
 -  2 DiscoverView.__preview__bo...
 -  3 DiscoverView.body()
 -
 -  38 UIApplicationMain
 -  **39 main**
 -  40 start

SwiftUI Runtime Issues

Found when process is running

Process continues executing

Details in Issues Navigator

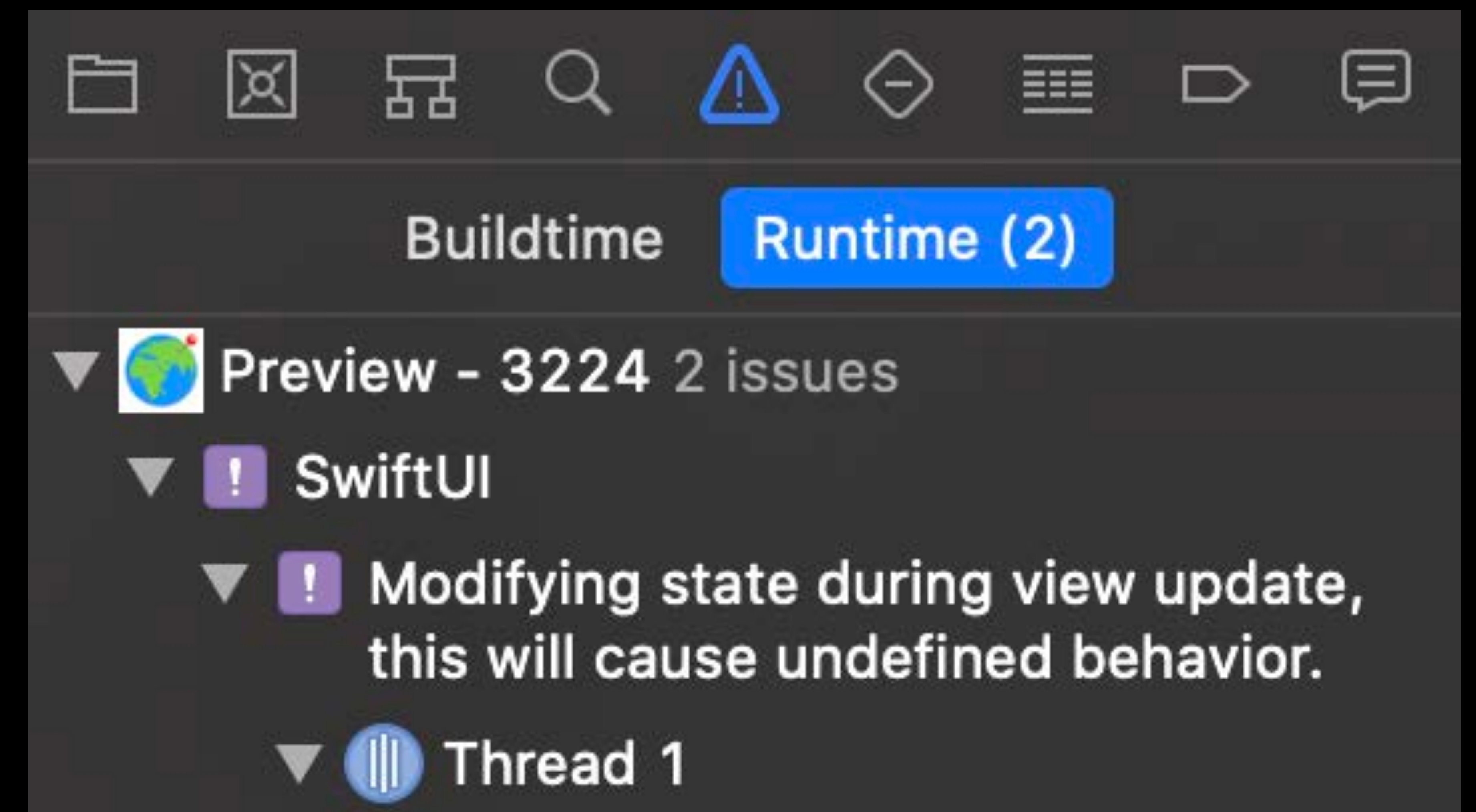
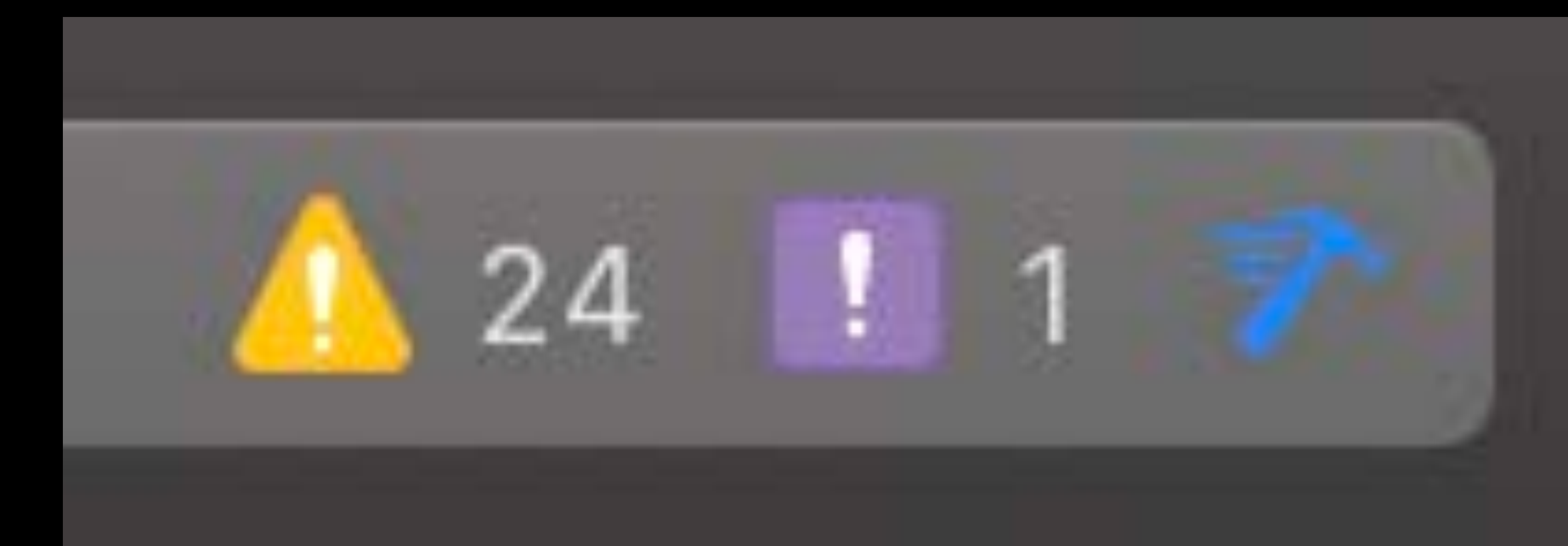


SwiftUI Runtime Issues

Found when process is running

Process continues executing

Details in Issues Navigator



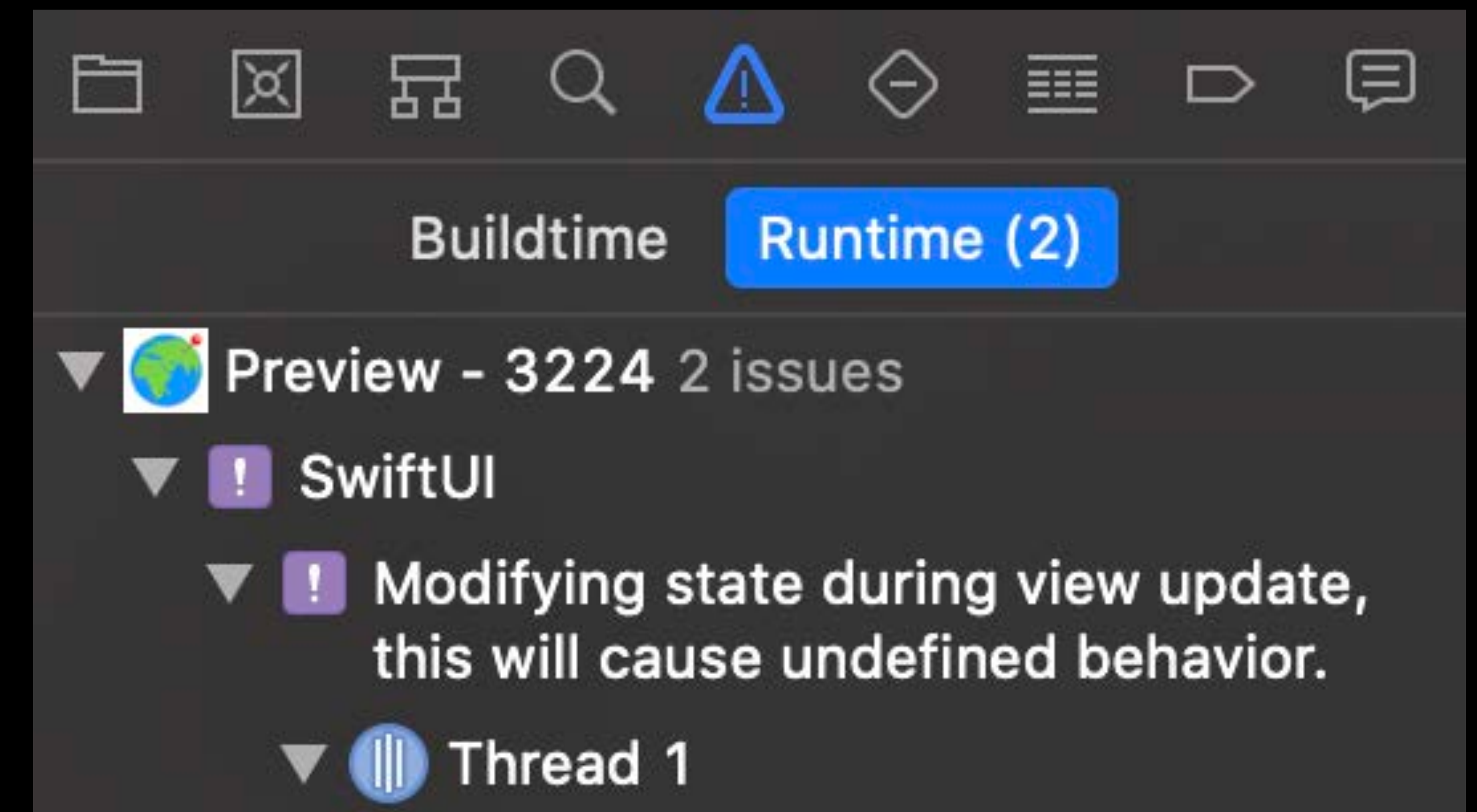
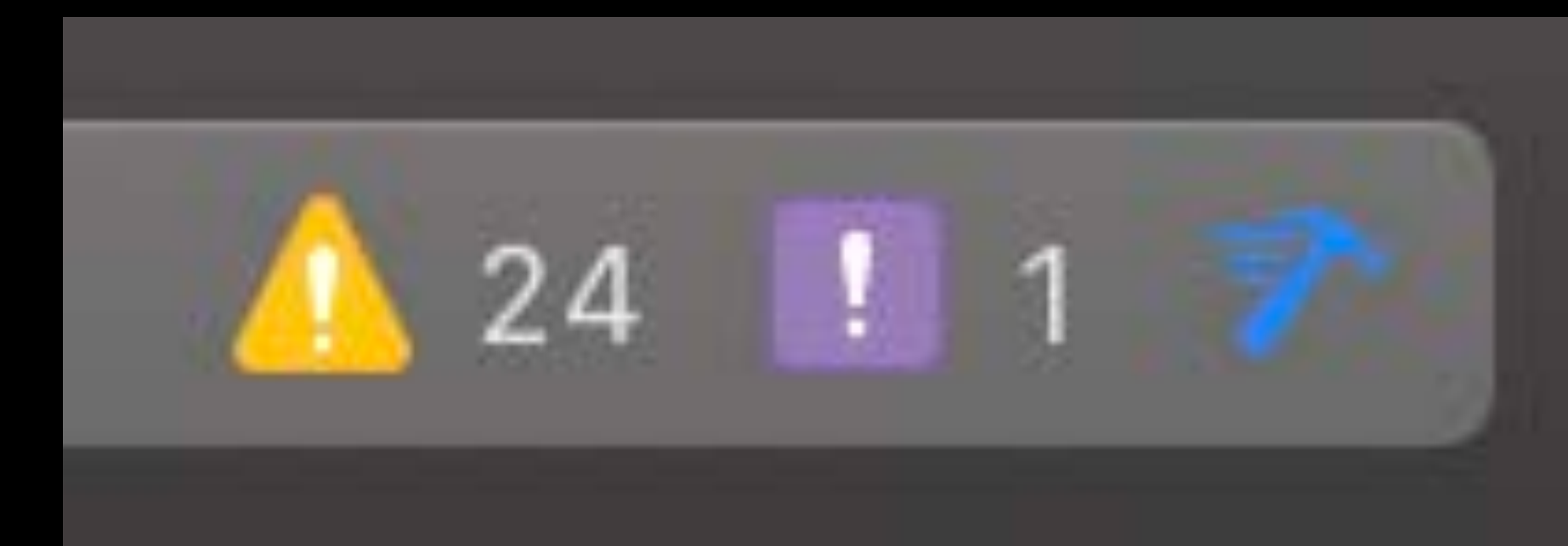
SwiftUI Runtime Issues

Found when process is running

Process continues executing

Details in Issues Navigator

All platforms and run destinations



Demo

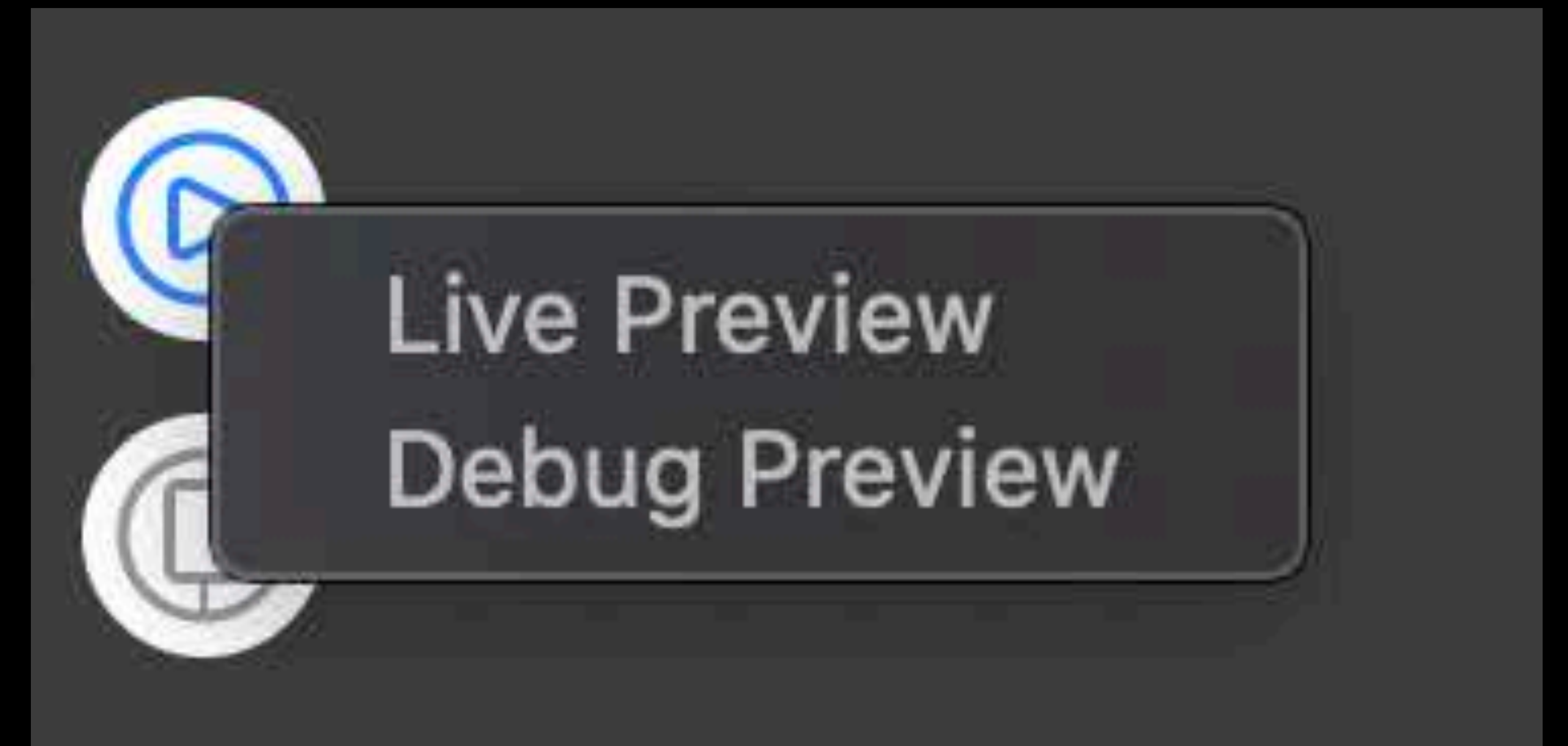
Debugging live previews

Han Ming Ong, Debugger UI Engineer

Debugging Workflow

Debugging Workflow

Use contextual menu to start debugging



Debugging Workflow

Use contextual menu to start debugging

New debug session for source changes

Debugging Workflow

Use contextual menu to start debugging

New debug session for source changes

Keep the same file in main editor

Debugging Live Previews

Debugging Live Previews

Live previews are debuggable

Debugging Live Previews

Live previews are debuggable

Workflows for previews

Debugging Live Previews

Live previews are debuggable

Workflows for previews

SwiftUI runtime issues

Debugging SwiftUI View Hierarchies

Sebastian Fischer, Xcode UI Engineer

Debugging SwiftUI View Hierarchies

Declarative API

Composition and lightweight modifiers over complex views

Value type semantics

Platform specific views at runtime

Interoperability with UIKit and AppKit

Demo

Debugging SwiftUI view hierarchies

Sebastian Fischer, Xcode UI Engineer

Debugging SwiftUI View Hierarchies

View hierarchy in navigator and canvas

Properties and modifiers in inspector

Automatic inspector properties through Swift reflection

Custom inspector properties by adopting `CustomReflectable`

Support for mixed view hierarchies

View Hierarchy Debugging Improvements

UIWindowScene support

Inspectors

- Trait collections
- Named images
- Symbol images
- Named colors
- Improved constraint details

More Information

developer.apple.com/wwdc19/412

Designing for Adverse Network and Temperature Conditions

Friday, 4:20

LLDB: Beyond "po"

WWDC 2019

