

# iOS Status Bar in Depth

Experiments with Private API

Alexander Zimin · iOSCon 2018 · 23th of March





# Alexander Zimin

Software Engineer at Badoo



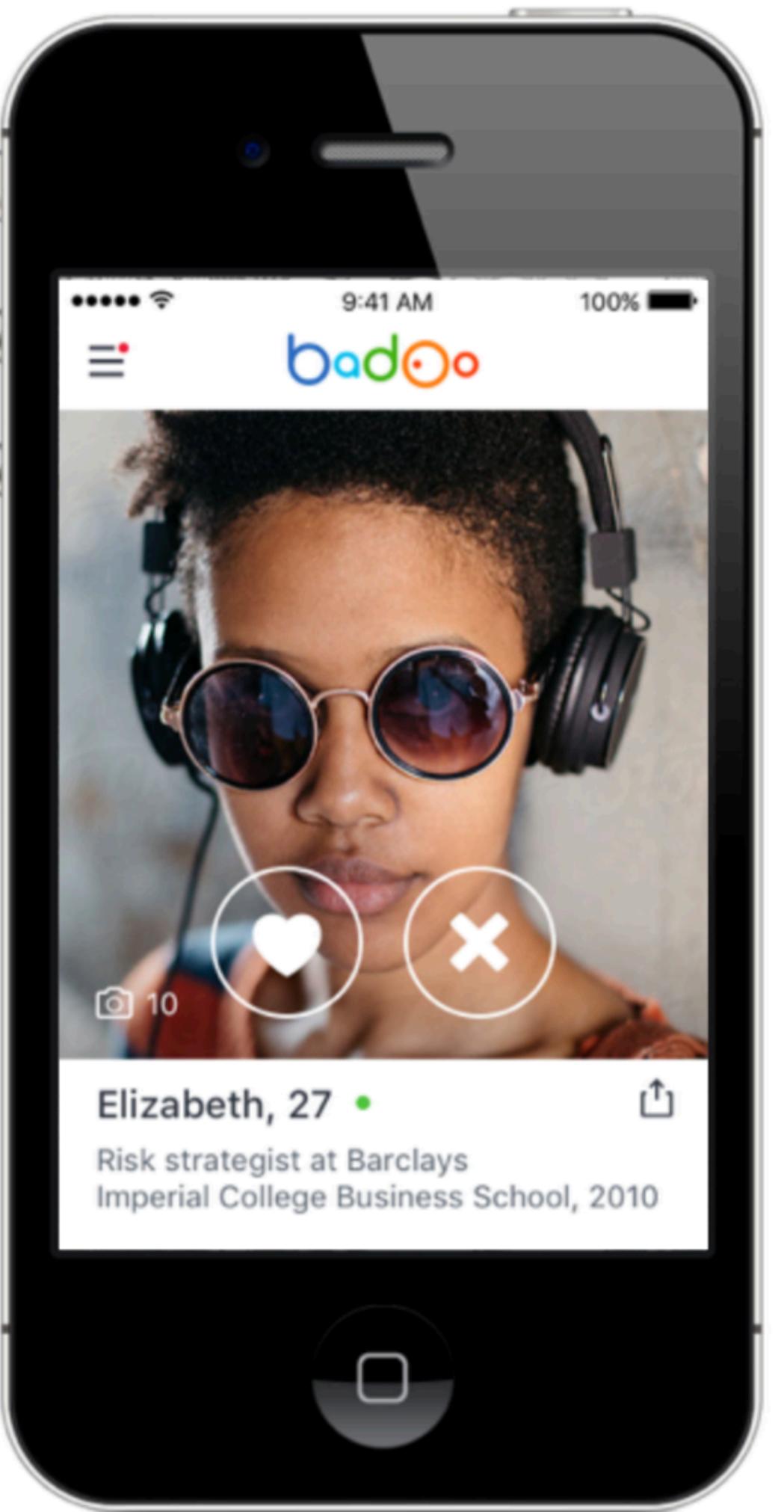
@ziminalex



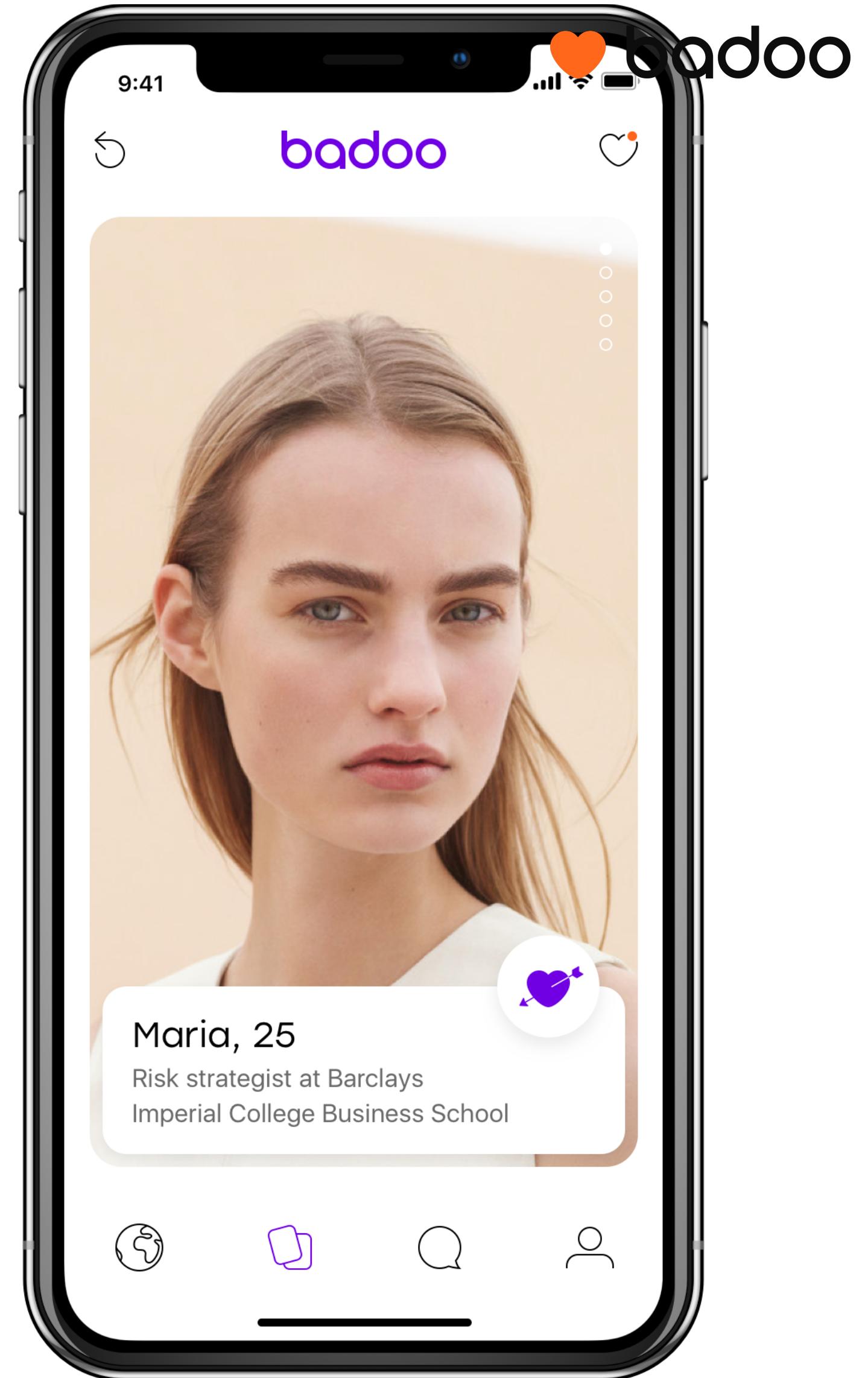
@azimin



2009

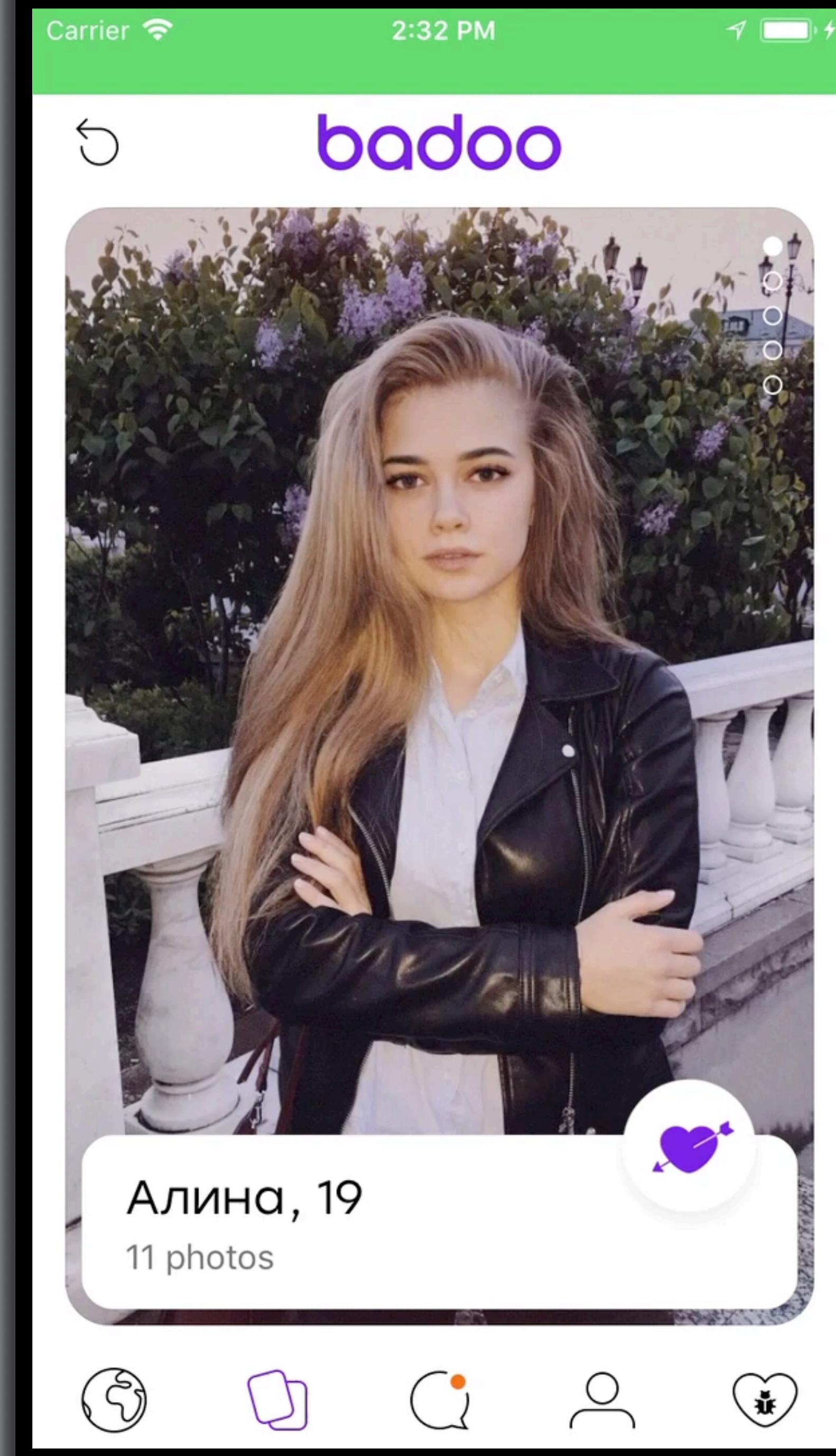


2014



2017

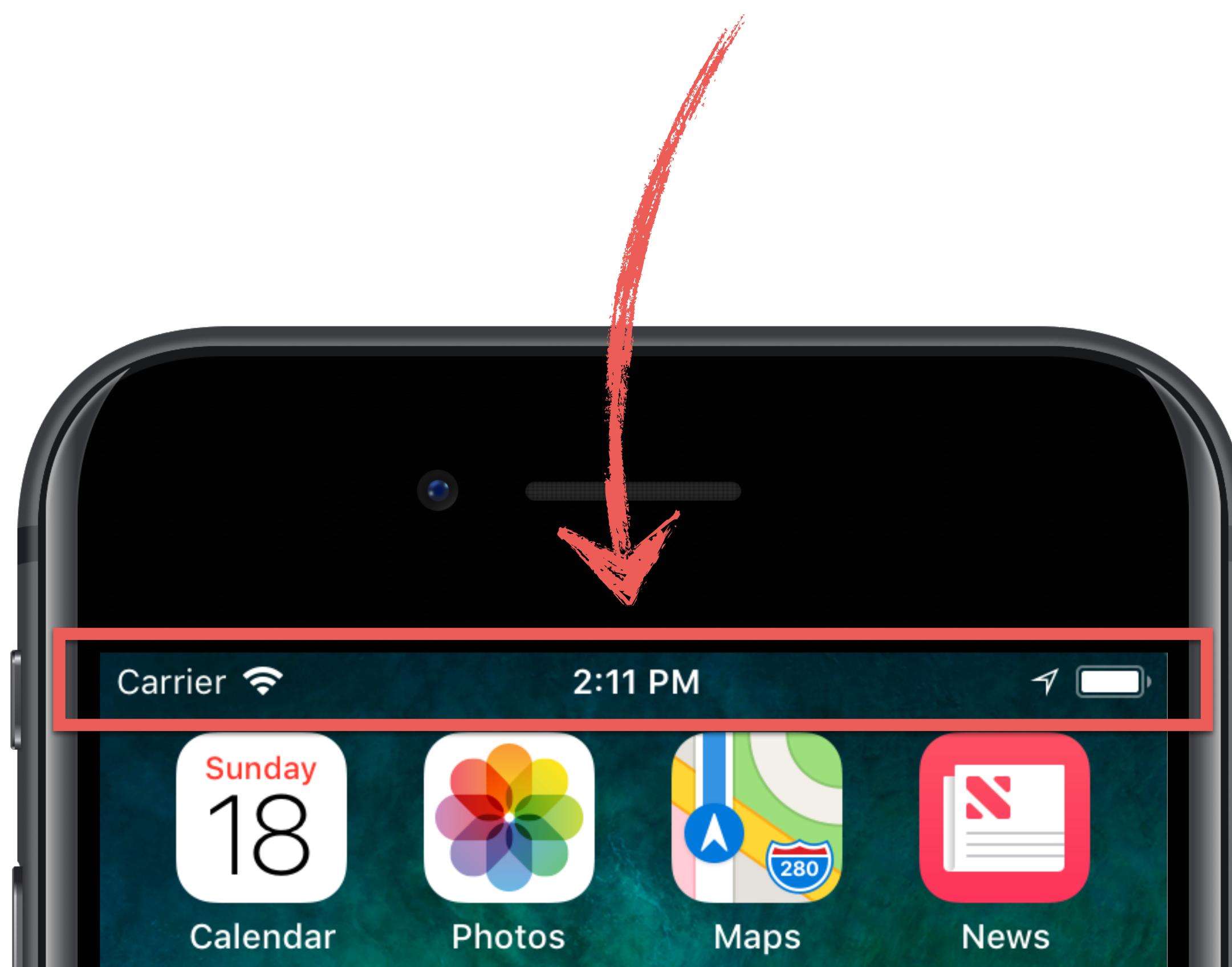
# Agenda



# What is it?

This is a “modern”  
status bar

This is a status bar





•• UIStatusBarWindow

▀ □ UIStatusBar

□ UIStatusBarBackgroundView

▀ □ UIStatusBarForegroundView

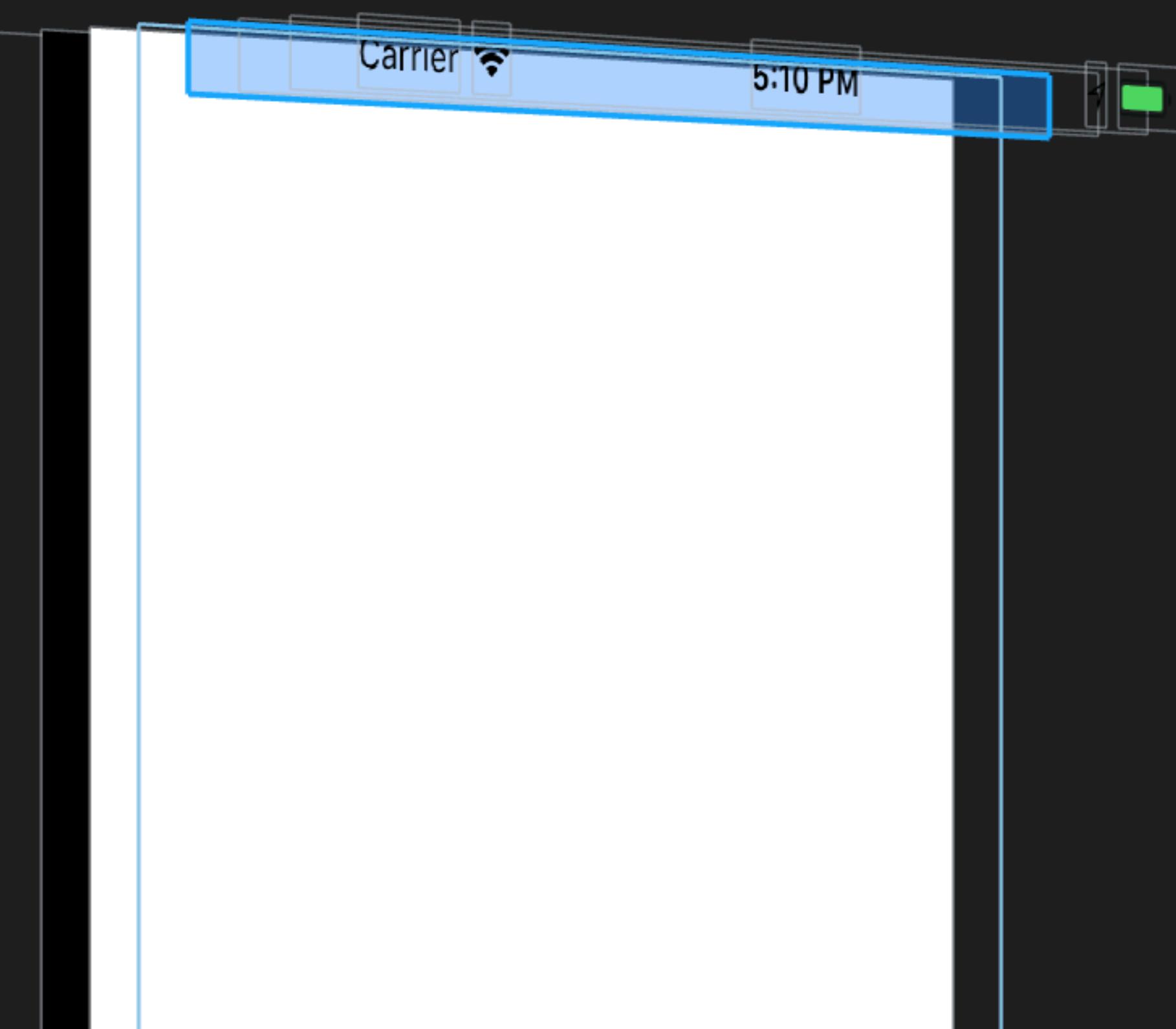
□ UIStatusBarServiceItemView

□ UIStatusBarDataNetworkItemView

► □ UIStatusBarBatteryItemView

□ UIStatusBarLocationItemView

□ UIStatusBarTimeItemView



[...]

▼  **UIStatusBarWindow**

▼  **UIStatusBar**

 **UIStatusBarBackgroundView**

▼  **UIStatusBarForegroundView**

 **UIStatusBarServiceItemView**

 **UIStatusBarDataNetworkItemView**

►  **UIStatusBarBatteryItemView**

 **UIStatusBarLocationItemView**

 **UIStatusBarTimeItemView**

► Constraints

▼ UIStatusBarWindow

▼ UIStatusBar\_Modern

▼ \_UIStatusBar

► Constraints

▼ UIView

► Constraints

▼ \_UIStatusBarForegroundView

► Constraints

► Layout Guides

▼ UIView

► Constraints

► Layout Guides

► \_UIStatusBarStringView: "2:34"

► \_UIStatusBarImageView

▼ UIView

► Constraints

▼ UIView

► Constraints

► Layout Guides

► \_UIStatusBarCellularSignalView

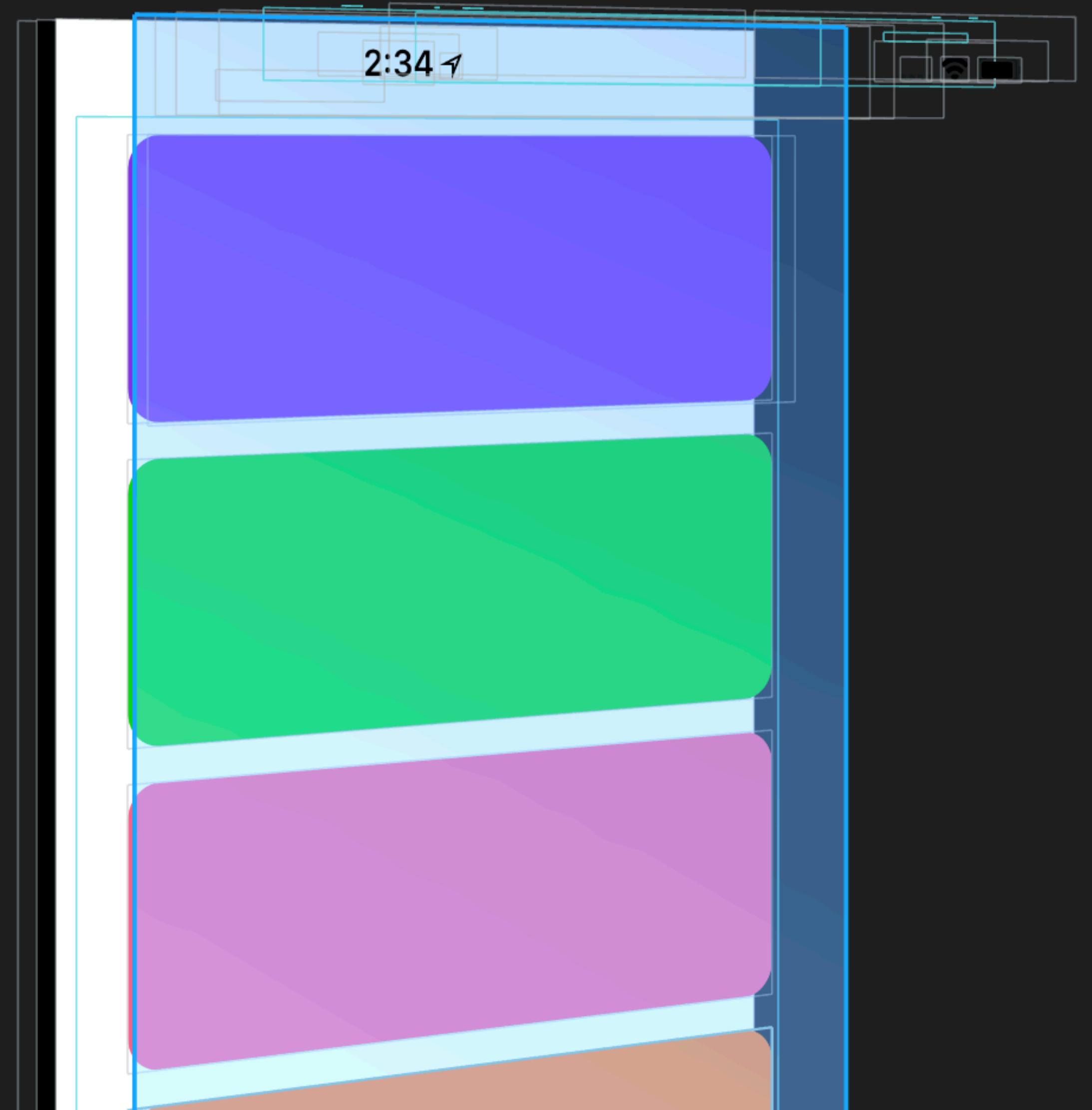
► \_UIStatusBarWifiSignalView

► \_UIBatteryView

UIView

UIView

UIView



▶  Constraints

▼  UIStatusBarWindow

▼  UIStatusBar\_Modern

▼  \_UIStatusBar

▶  Constraints

▼  UIView

▶  Constraints

▼  \_UIStatusBarForegroundView

▶  Constraints

▶  Layout Guides

▼  UIView

▶  Constraints

▶  Layout Guides

▶  \_UIStatusBarStringView: "2:34"

▶  \_UIStatusBarImageView

▼  UIView

▶  Constraints

# **Cons**

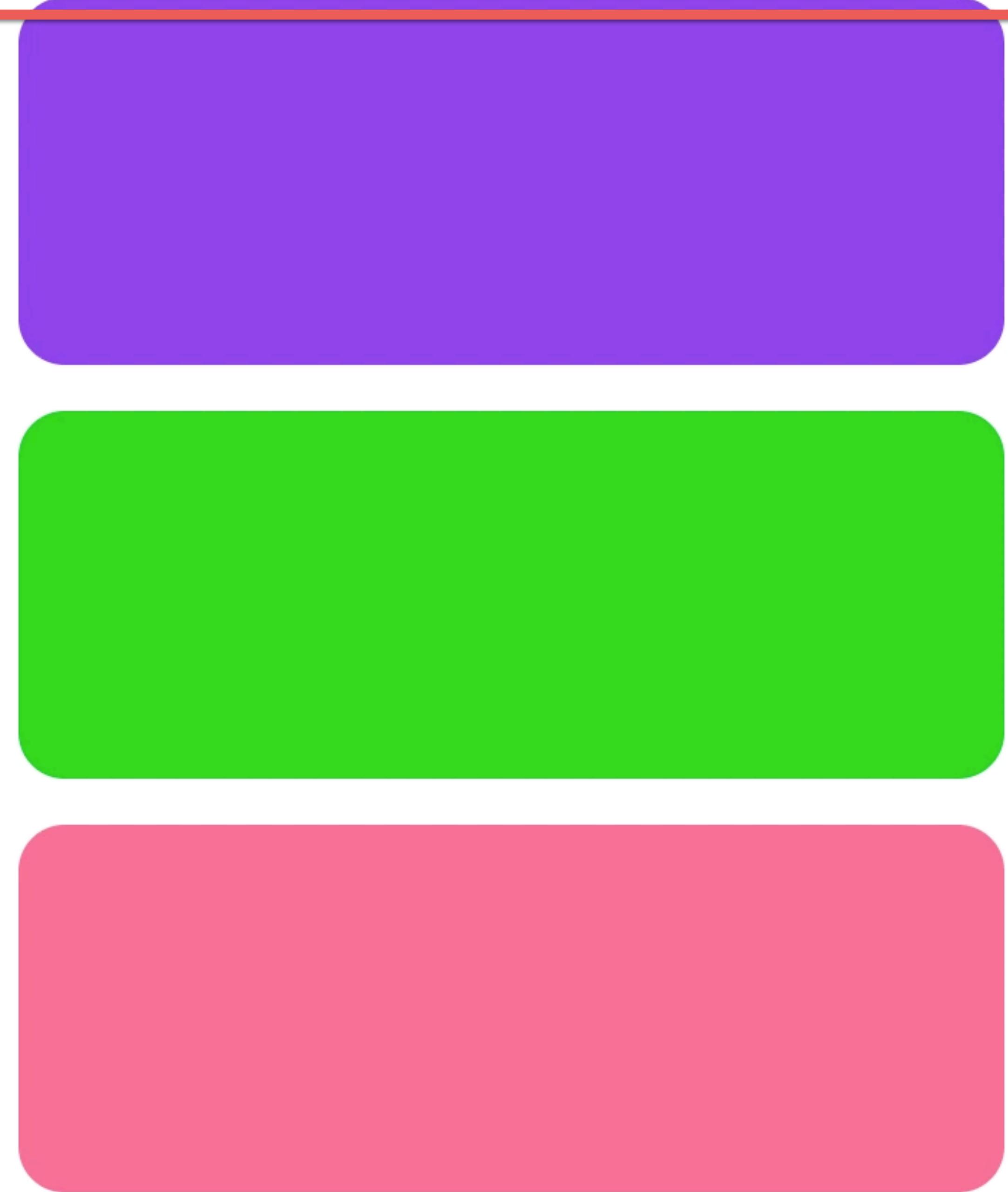
- 1. No customization**
- 2. No extension of behaviour**
- 3. Sometimes not ideal**

2015

```
override var prefersStatusBarHidden: Bool {  
    return true  
}
```

Carrier

6:31 PM



```
var statusBarShouldBeHidden = true
```

```
statusBarShouldBeHidden = true
UIView.animate(withDuration: 0.1) {
    self.setNeedsStatusBarAppearanceUpdate()
}
```

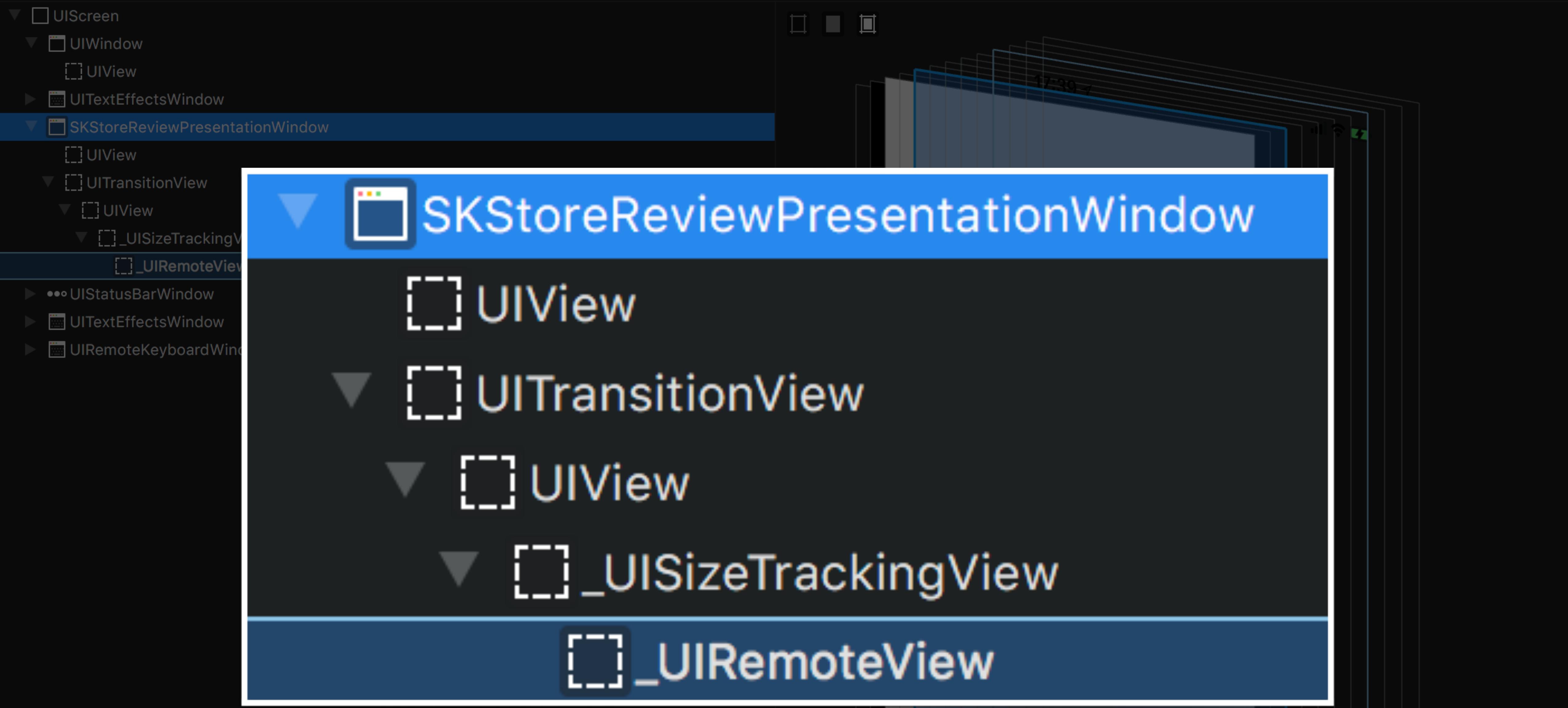
```
override var prefersStatusBarHidden: Bool {
    return statusBarShouldBeHidden
}
```

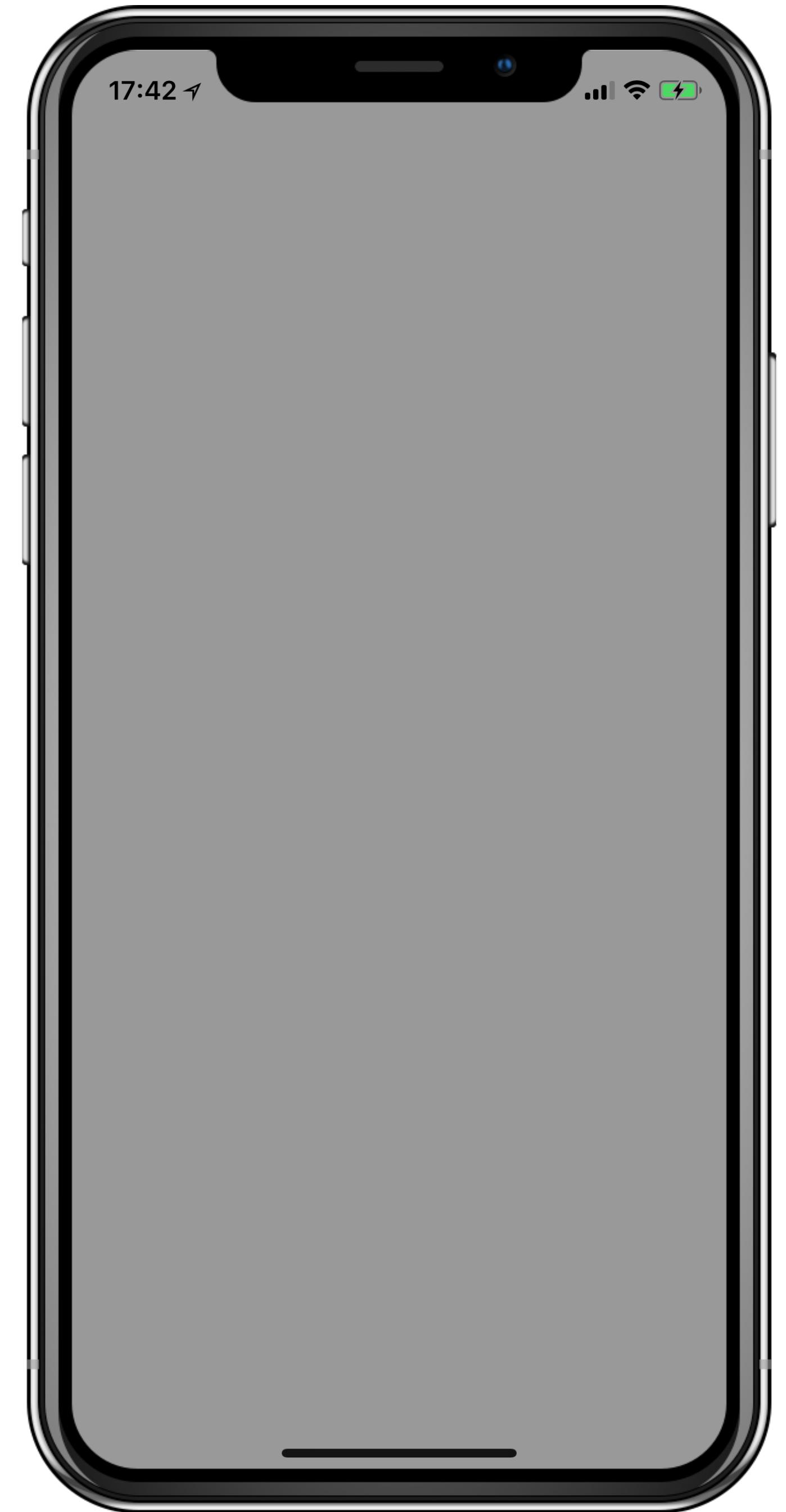
# Idea

- 1. Change style of status bar by ourselves**
- 2. Doesn't change top layout guide**

```
public var statusBar: UIView? {  
    if UIApplication.shared.responds(to: NSSelectorFromString("statusBar")) {  
        if let view = UIApplication.shared.value(forKey: "statusBar") as? UIView {  
            return view  
        }  
    }  
    return nil  
}
```

statusBar/statusBarWindow







**Enjoying this awesome App?**

Tap a star to rate it on the  
App Store.



[Not Now](#)

# Safety rules

- Read private headers

[Code](#)[Issues 48](#)[Pull requests 11](#)[Projects 0](#)[Wiki](#)[Insights](#)

iOS Objective-C headers as derived from runtime introspection <https://github.com/nst/RuntimeBrowser>

79 commits

1 branch

41 releases

1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

 Nicolas Seriot committed on Apr 20, 2017 iOS 10.3

Latest commit 9c31ca2 on Apr 20, 2017

 Frameworks

iOS 10.3

11 months ago

 PrivateFrameworks

iOS 10.3

11 months ago

 lib

iOS 10.3

11 months ago

 protocols

iOS 10.3

11 months ago

 README.md

Update README.md

3 years ago

 ios\_frameworks.png

updated history PNG

2 years ago

Branch: master ▾

iOS-Runtime-Headers / Frameworks / UIKit.framework / UIBarStatus.h

Find file

Copy path

 Nicolas Seriot iOS 10.3

9c31ca2 on Apr 20, 2017

1 contributor

269 lines (263 sloc) | 20 KB

Raw Blame History   

```
1  /* Generated by RuntimeBrowser
2   Image: /System/Library/Frameworks/UIKit.framework/UIKit
3   */
4
5  @interface UIBarStatus : _UIScrollsToTopInitiatorView <UIStatusBarServerClient, UIStatusBarStateObserver> {
6      <UIViewControllerTransitionCoordinator> * __transitionCoordinator;
7      UIBarBackgroundView * _backgroundView;
8      NSString * _currentDoubleHeightText;
9      struct {
10          BOOL itemEnabled[35];
11          BOOL timeString[64];
12          int gsmSignalStrengthRaw;
13          int gsmSignalStrengthBars;
14          BOOL serviceString[100];
15          BOOL serviceCrossfadeString[100];
16          BOOL serviceImages[2][100];
17          BOOL operatorDirectory[1024];
18          unsigned int serviceContentType;
19          int wifiSignalStrengthRaw;
20          int wifiSignalStrengthBars;
21          unsigned int dataNetworkType;
22          int batteryCapacity;
23          unsigned int batteryState;
24          BOOL batteryDetailString[150];
25          int bluetoothBatteryCapacity;
26          int thermalColor;
27          unsigned int thermalSunlightMode : 1;
28          unsigned int slowActivity : 1;
29          unsigned int syncActivity : 1;
30          BOOL activityDisplayId[256];
31          unsigned int bluetoothConnected : 1;
```

- **(BOOL)isDoubleHeight;**
- **(BOOL)isHidden;**
- **(BOOL)isTimeHidden;**
- **(BOOL)isTranslucent;**

# Safety rules

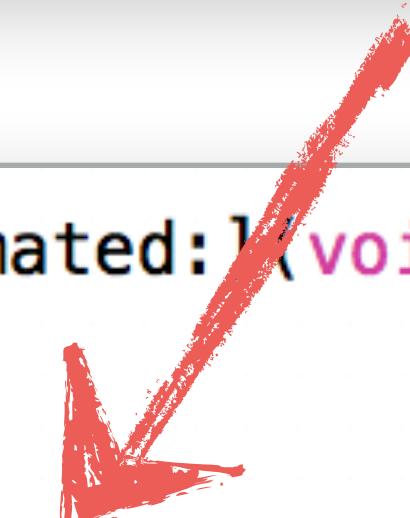
- **Read private headers**
- **Use Hopper to check the behaviour of the methods**

```
void -[UIStatusBar_Base setHidden:](void * self, void * _cmd, bool arg2) {  
    [self setHidden:arg2 & 0xff animated:0x0];  
    return;  
}
```

```
void -[UIStatusBar_Base setHidden:animated:](void * self, void * _cmd, bool arg2, bool arg3) {  
    r14 = arg2;  
    r15 = self;  
    if (arg3 != 0x0) {  
        rbx = [[UIStatusBarHideAnimationParameters alloc] initWithDefaultParameters];  
    }  
    else {  
        rbx = 0x0;  
    }  
    [r15 setHidden:r14 & 0xff animationParameters:rbx];  
    rdi = rbx;  
    [rdi release];  
    return;  
}
```

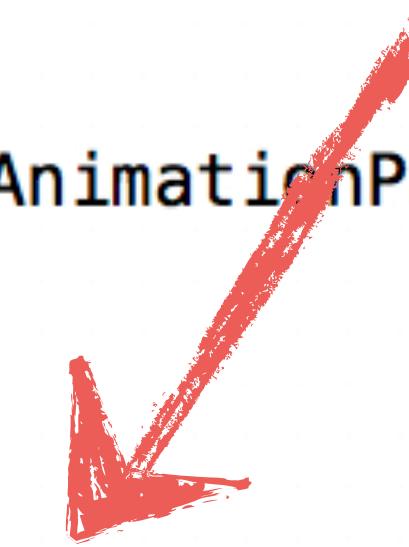
```
void -[UIStatusBar_Base setHidden:](void * self, void * _cmd, bool arg2) {  
    [self setHidden:arg2 & 0xff animated:0x0];  
    return;  
}
```

```
void -[UIStatusBar_Base setHidden:animated:](void * self, void * _cmd, bool arg2, bool arg3) {  
    r14 = arg2;  
    r15 = self;  
    if (arg3 != 0x0) {  
        rbx = [[UIStatusBarHideAnimationParameters alloc] initWithDefaultParameters];  
    }  
    else {  
        rbx = 0x0;  
    }  
    [r15 setHidden:r14 & 0xff animationParameters:rbx];  
    rdi = rbx;  
    [rdi release];  
    return;  
}
```



```
void -[UIStatusBar_Base setHidden:](void * self, void * _cmd, bool arg2) {  
    [self setHidden:arg2 & 0xff animated:0x0];  
    return;  
}
```

```
void -[UIStatusBar_Base setHidden:animated:] (void * self, void * _cmd, bool arg2, bool arg3) {  
    r14 = arg2;  
    r15 = self;  
    if (arg3 != 0x0) {  
        rbx = [[UIStatusBarHideAnimationParameters alloc] initWithDefaultParameters];  
    }  
    else {  
        rbx = 0x0;  
    }  
    [r15 setHidden:r14 & 0xff animationParameters:rbx];  
    rdi = rbx;  
    [rdi release];  
    return;  
}
```

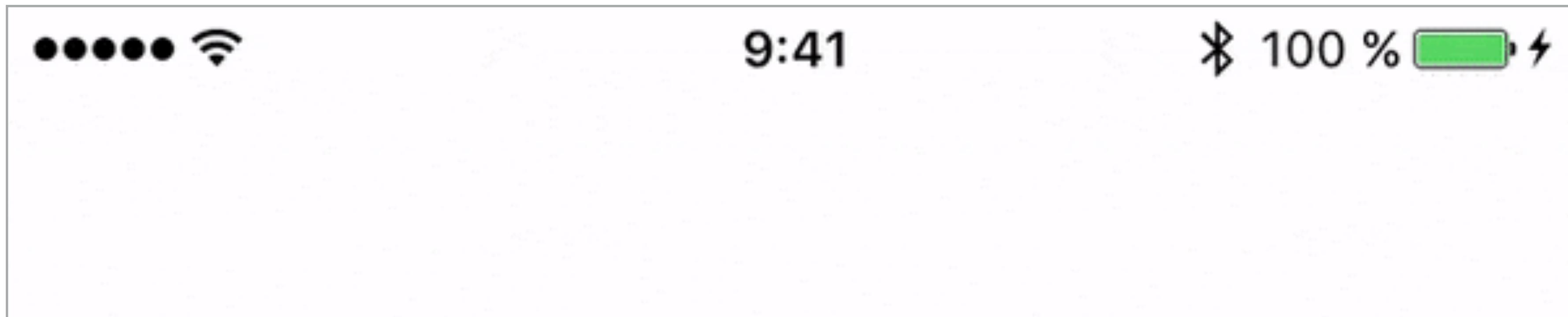


# Safety rules

- **Read private headers**
- **Use Hopper to check the behaviour of the methods**
- **Cover with error handling blocks**

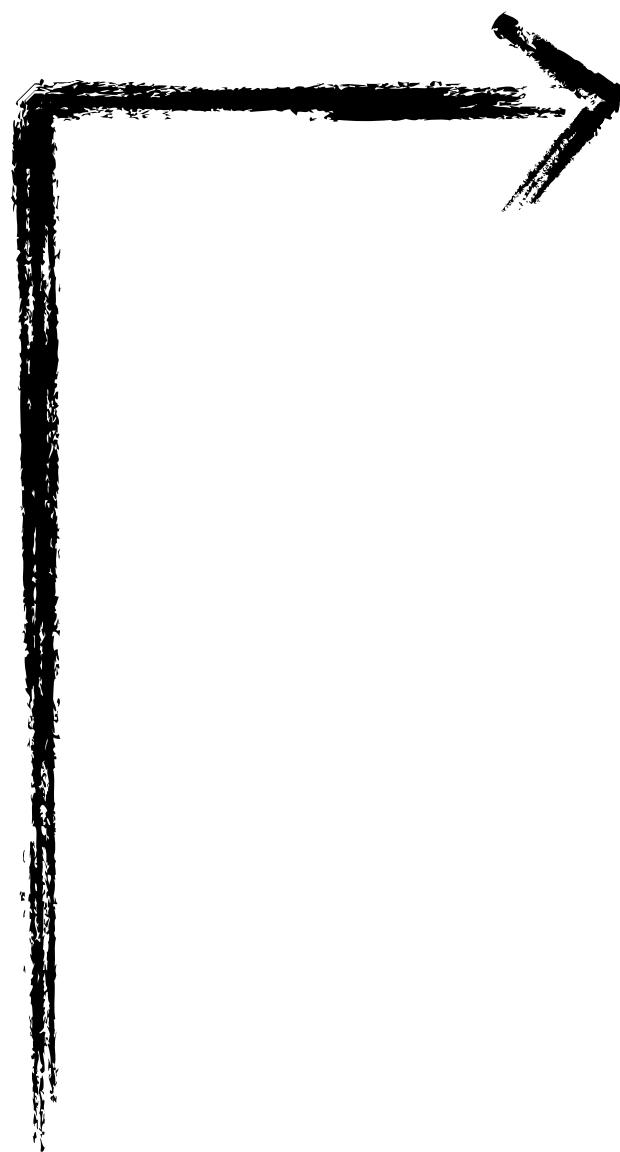
# Safety rules

- **Read private headers**
- **Use Hopper to check the behaviour of the methods**
- **Cover with error handling blocks**
- **Create to-test list and check before major releases**



<https://medium.com/ml-works-engineering/overlap-f60f60df09e6>

# Array

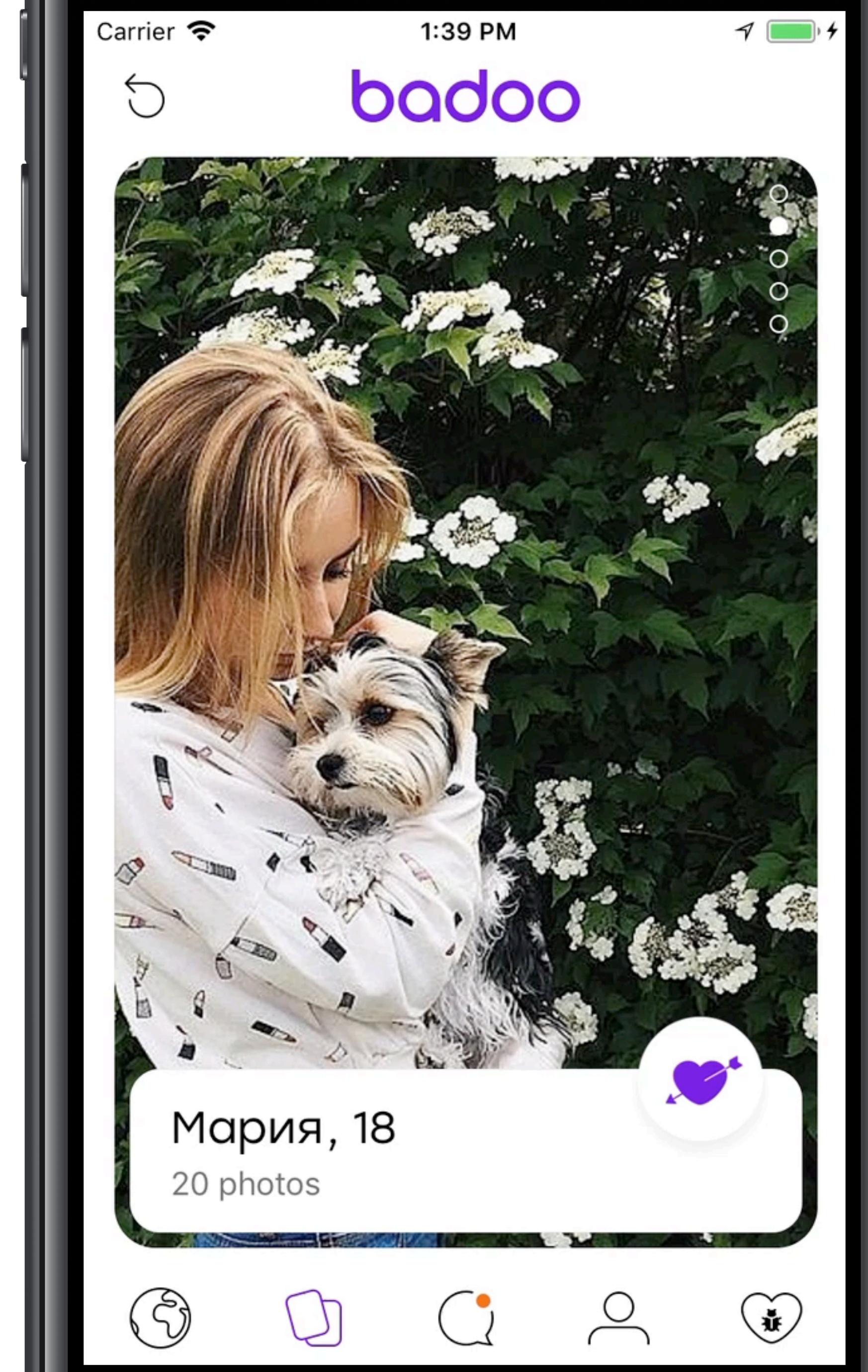


## Reference

- Contains appearance parameters

## Manager

- Find an appropriate reference
- Apply reference logic

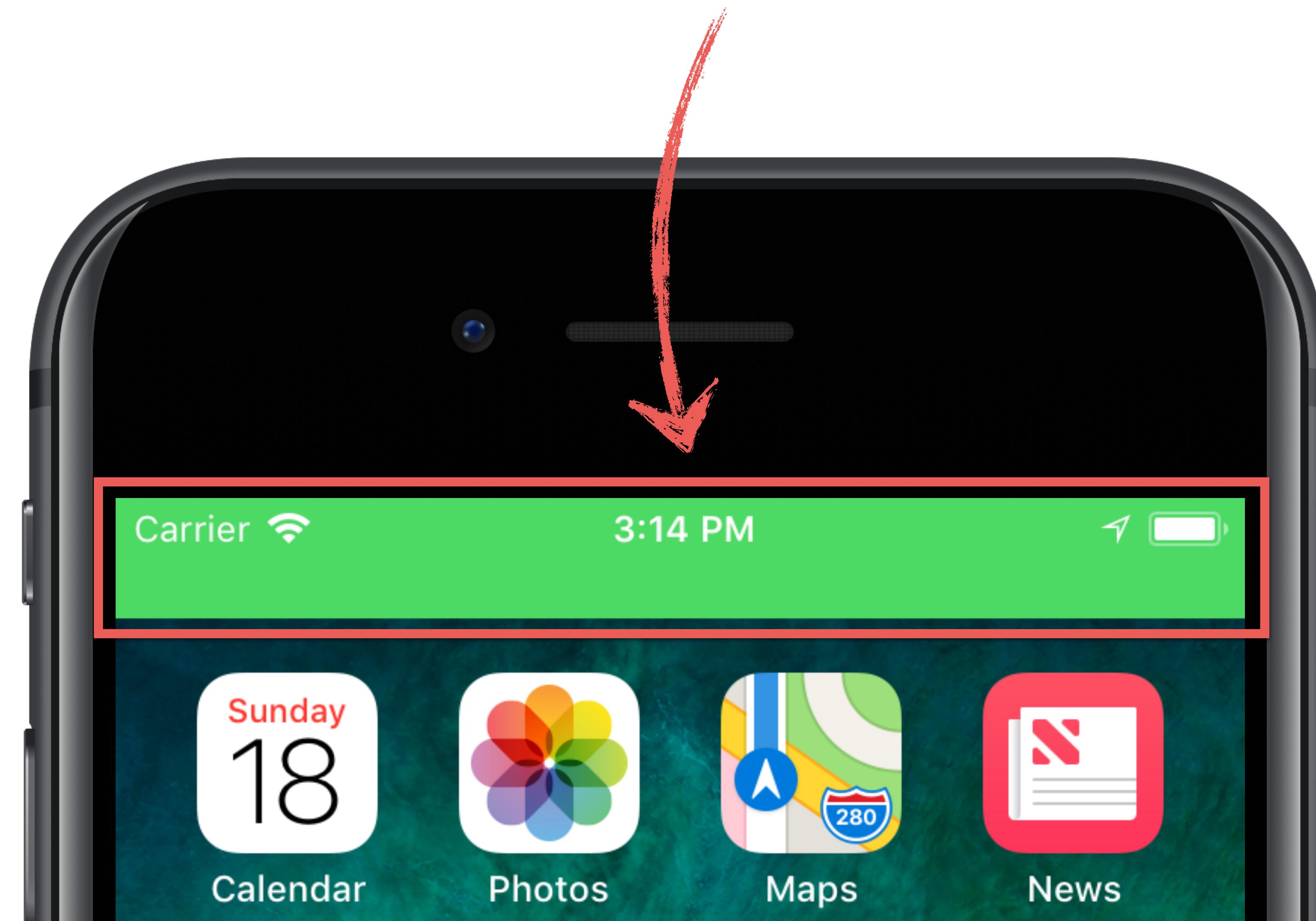


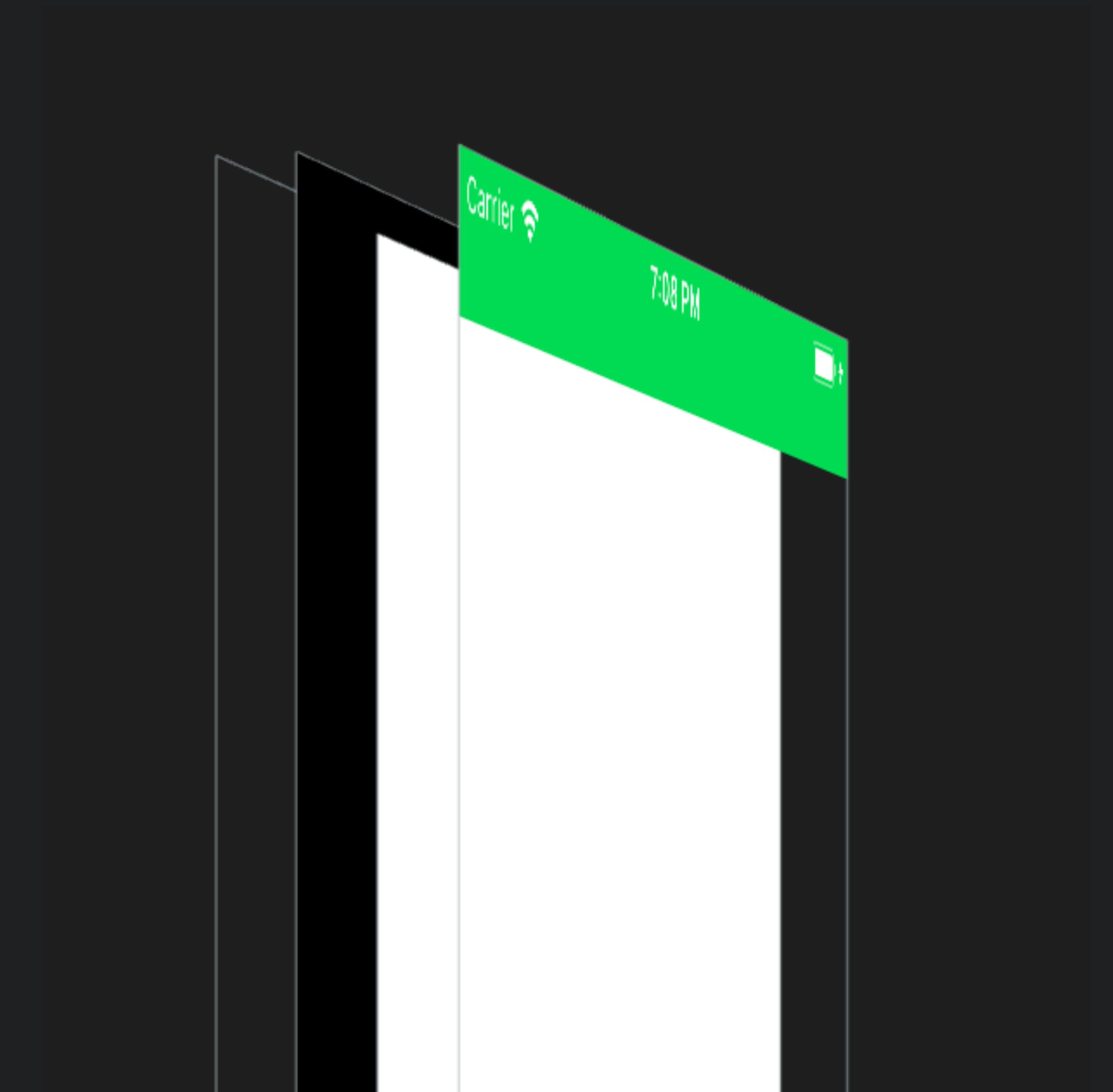
2017

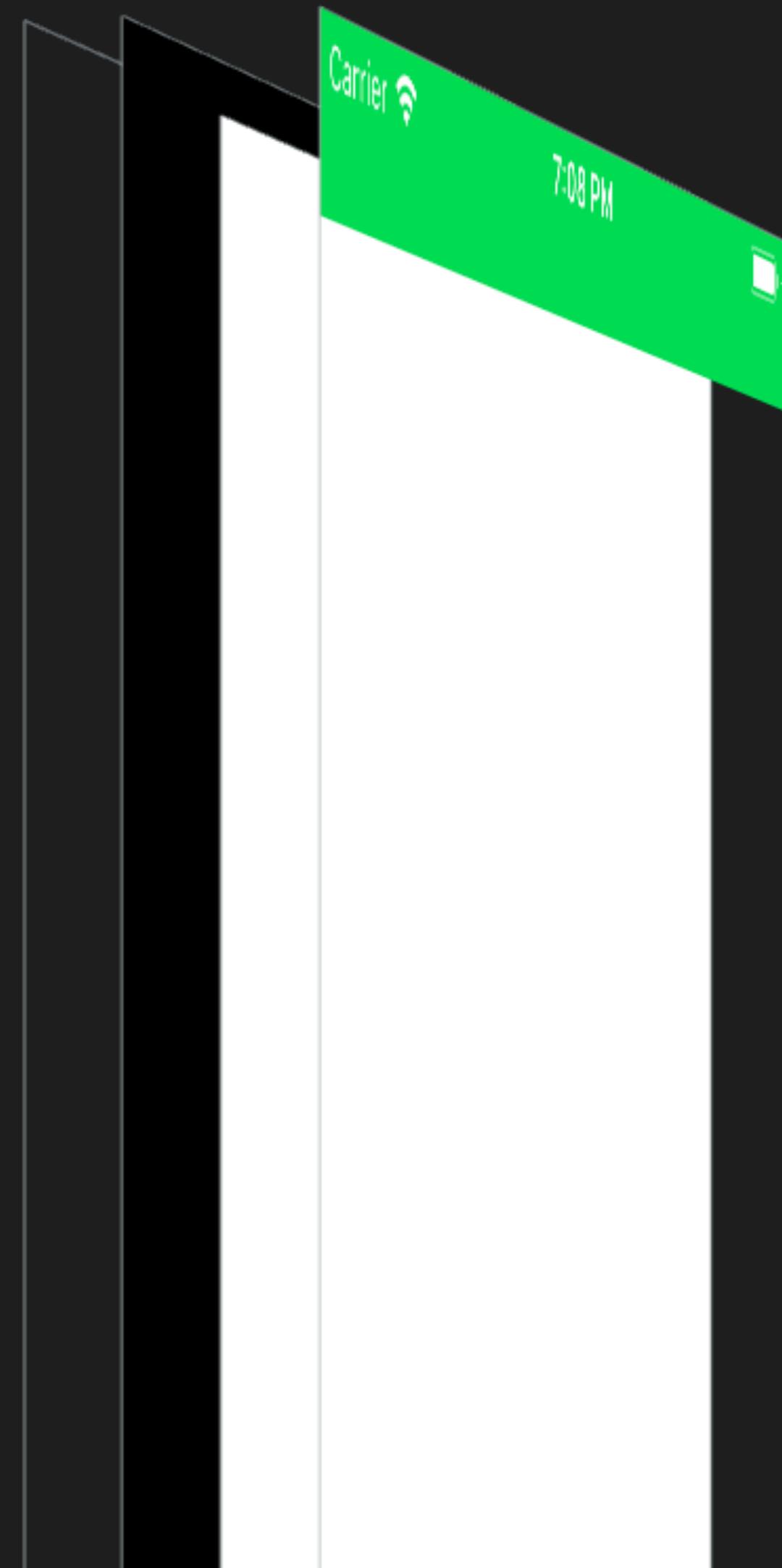
In-Call Status Bar



# This is an In-Call Status Bar



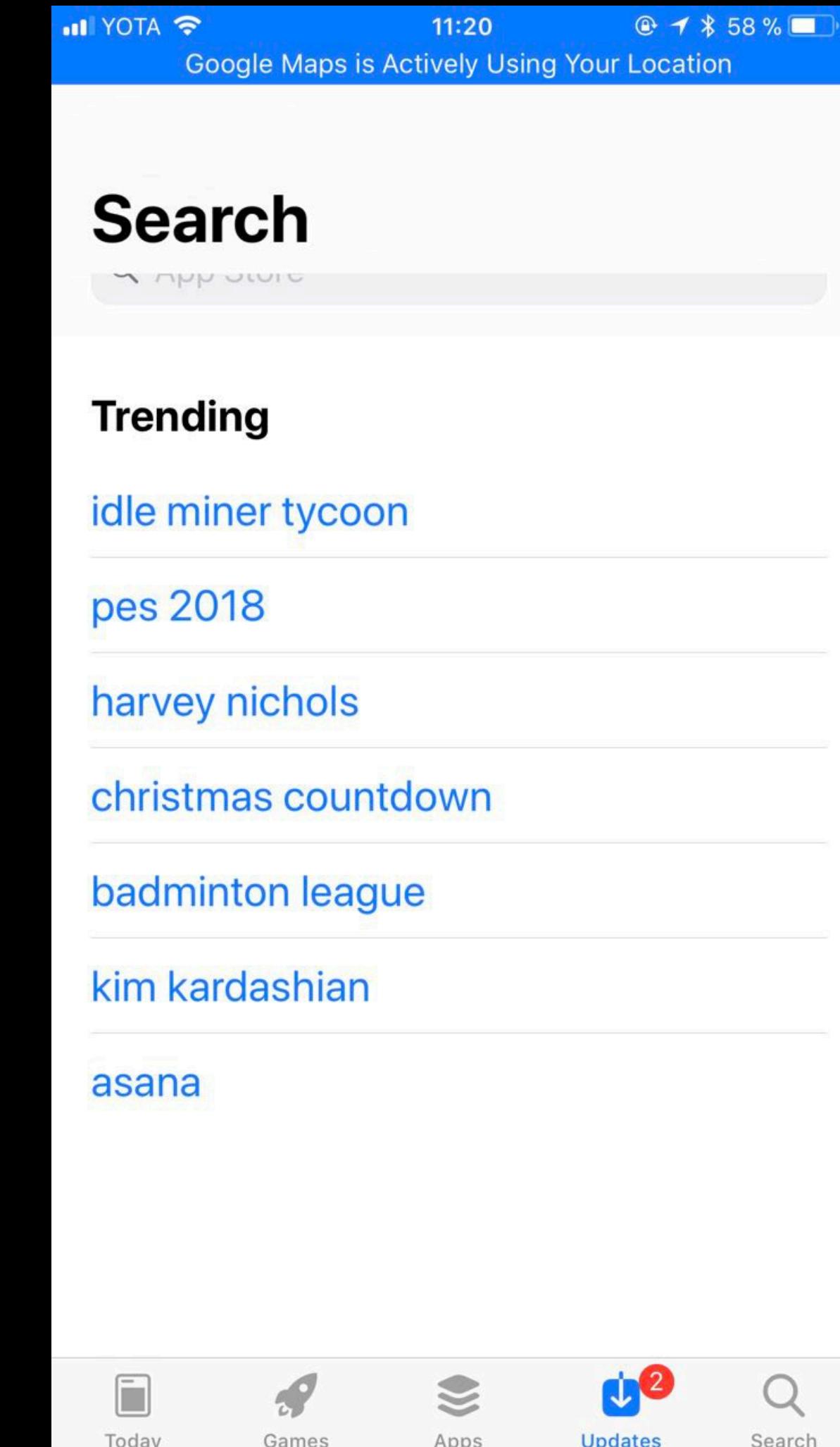
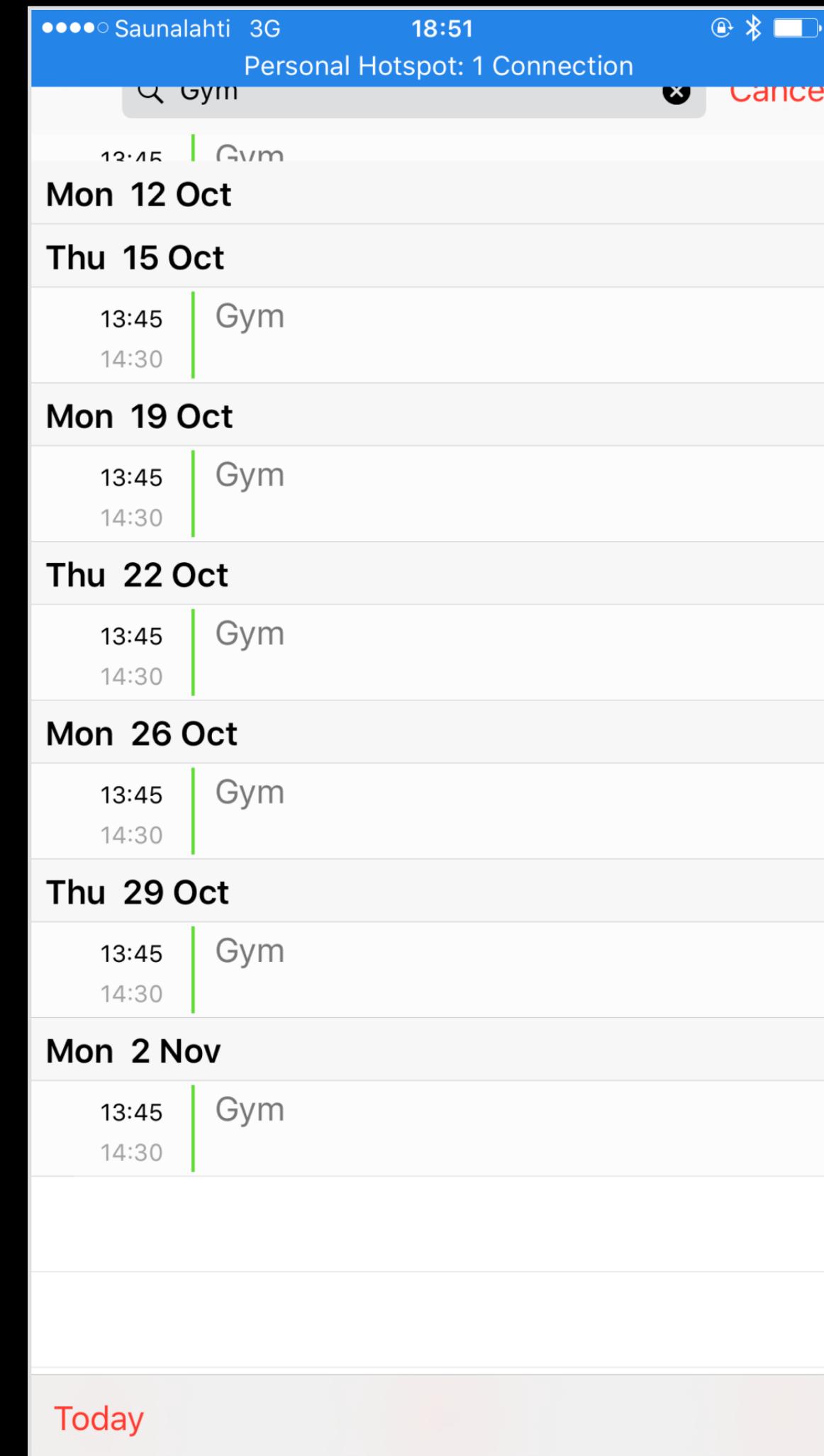
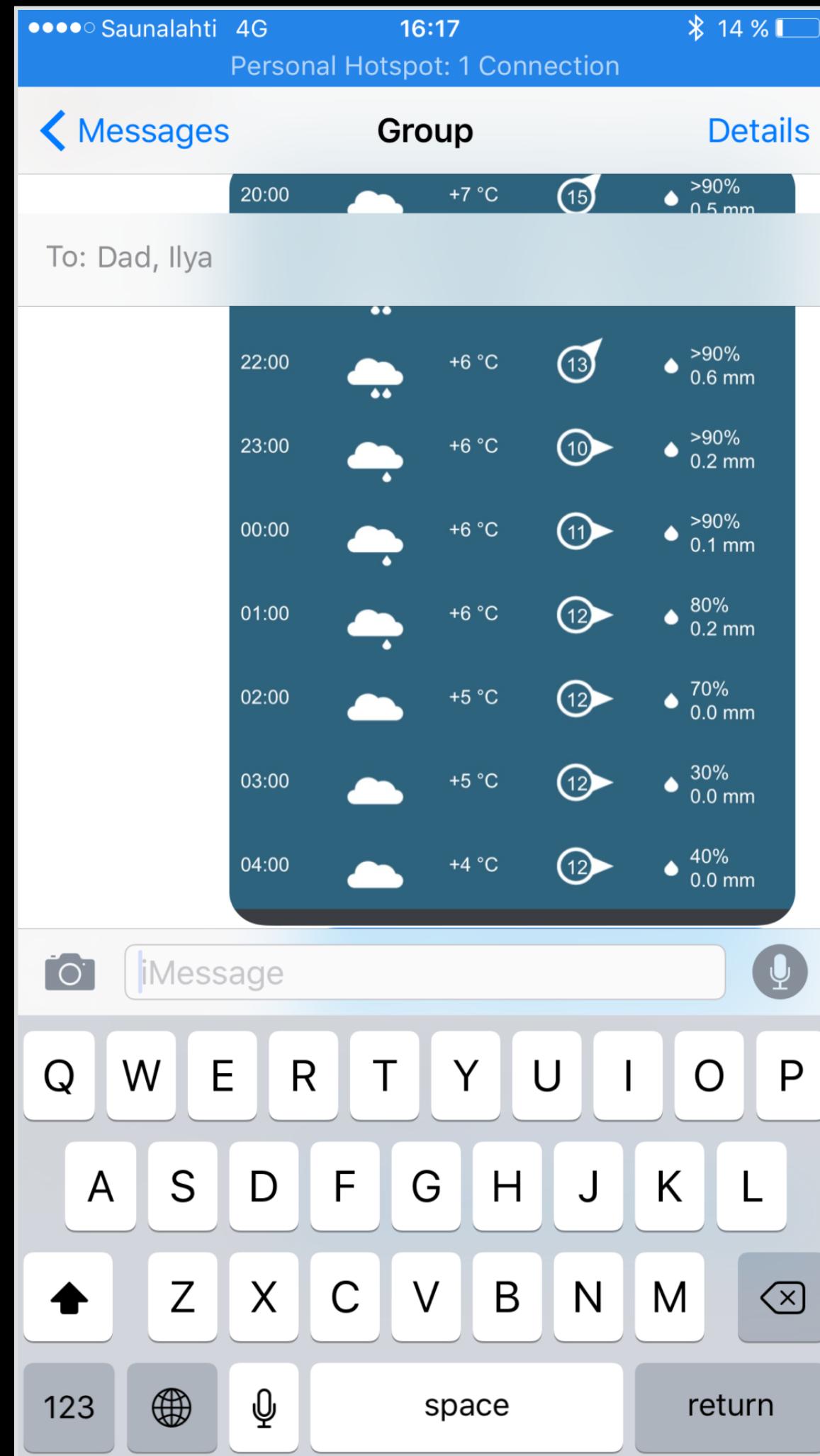




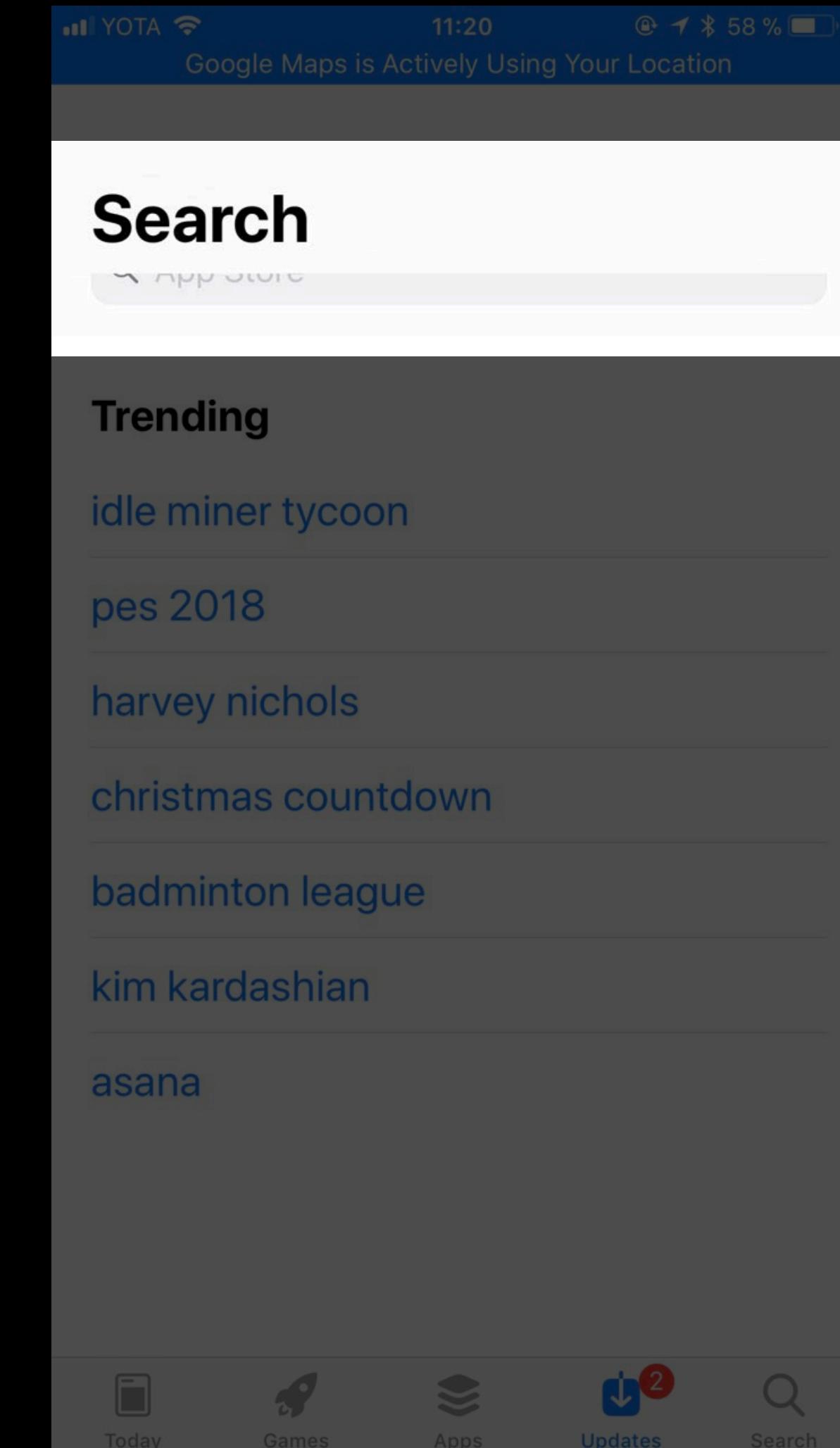
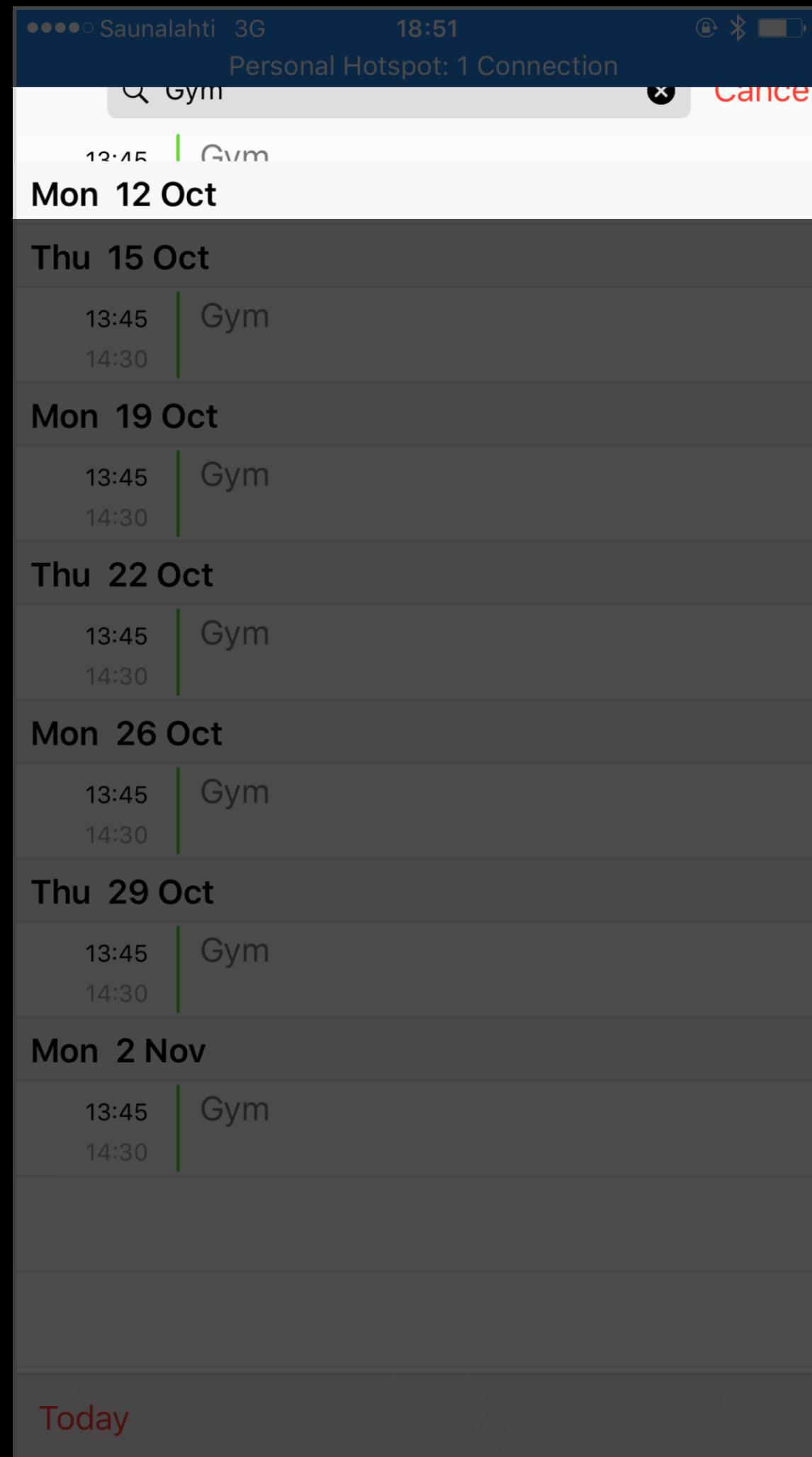
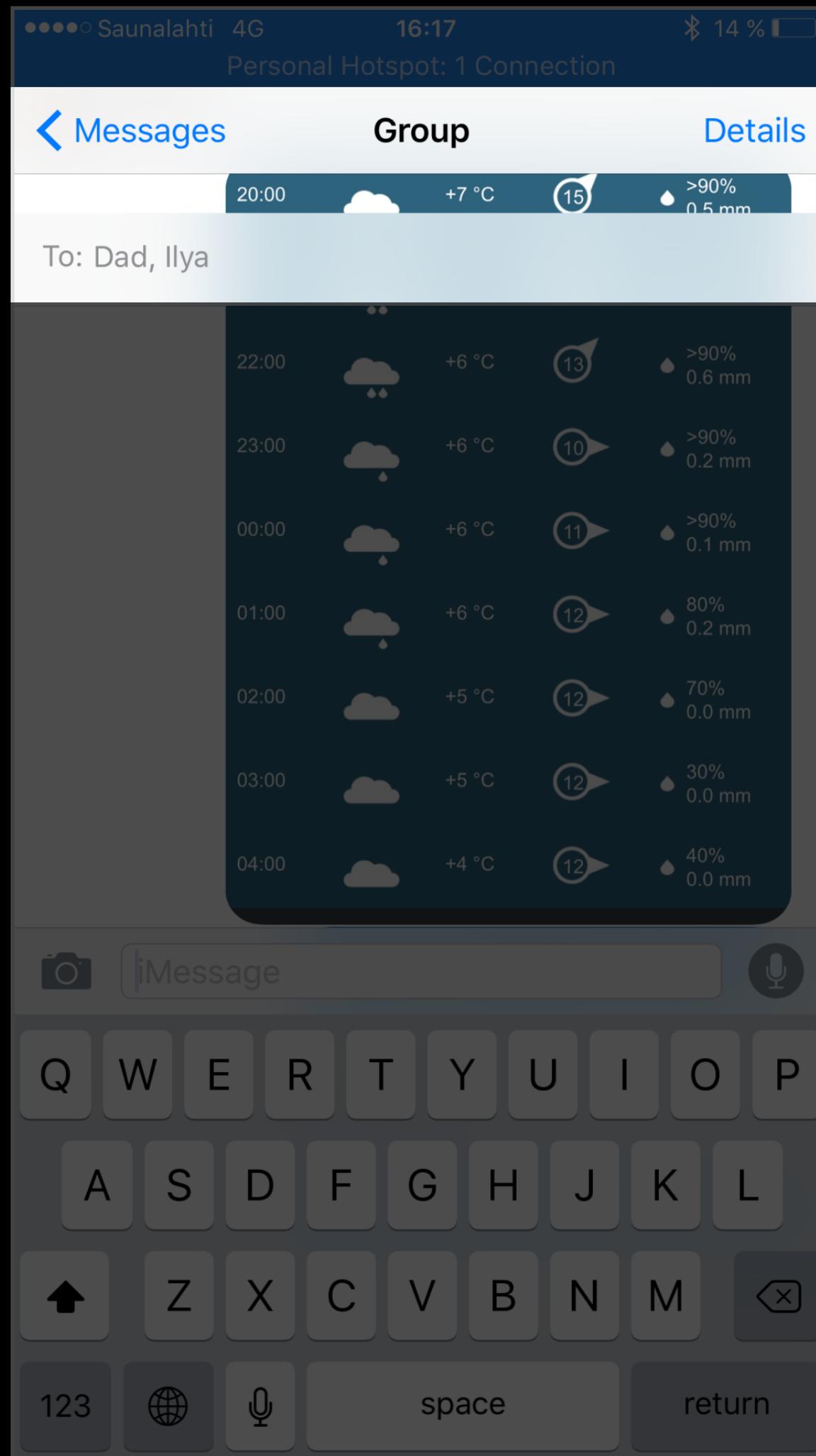
## Layout

Frame

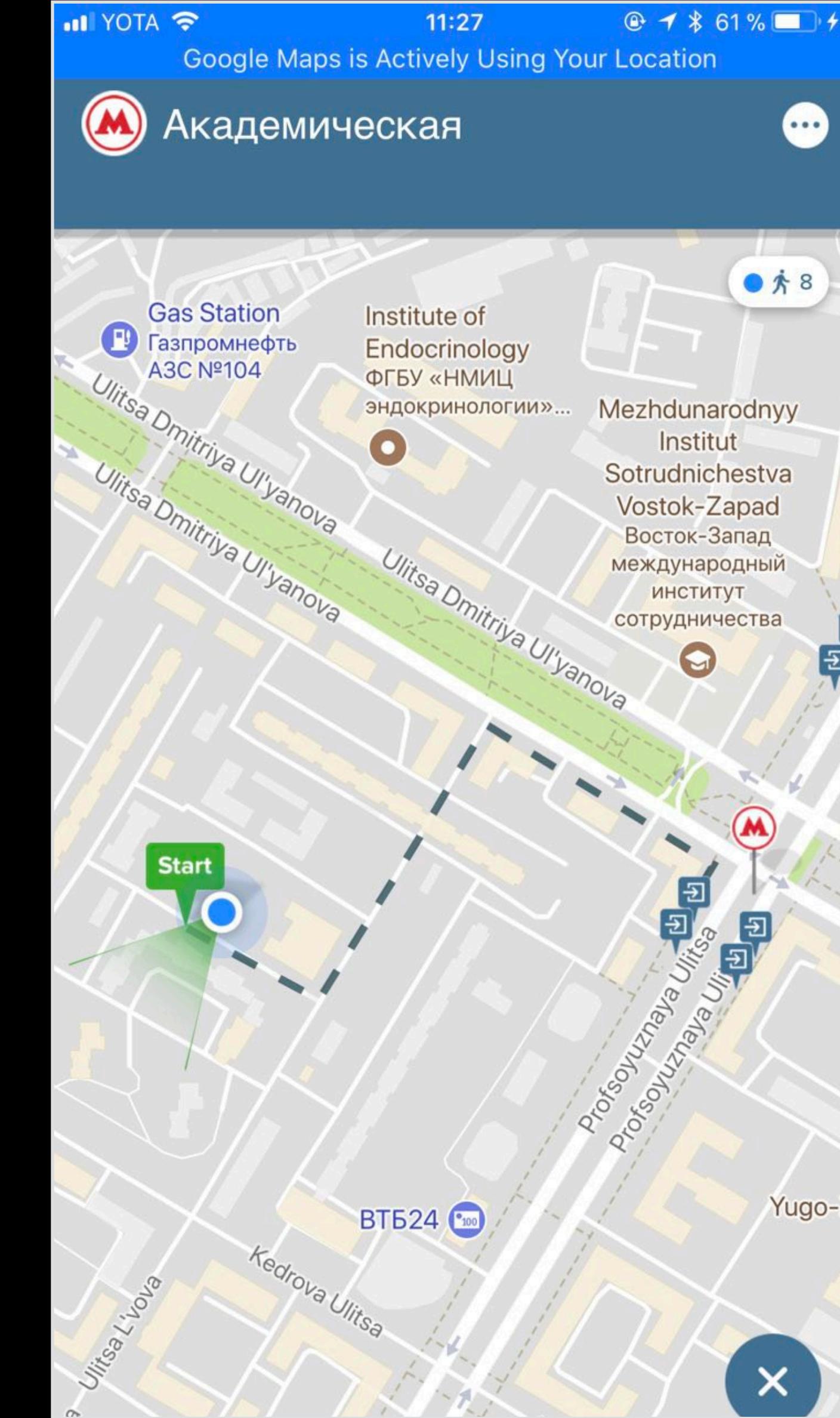
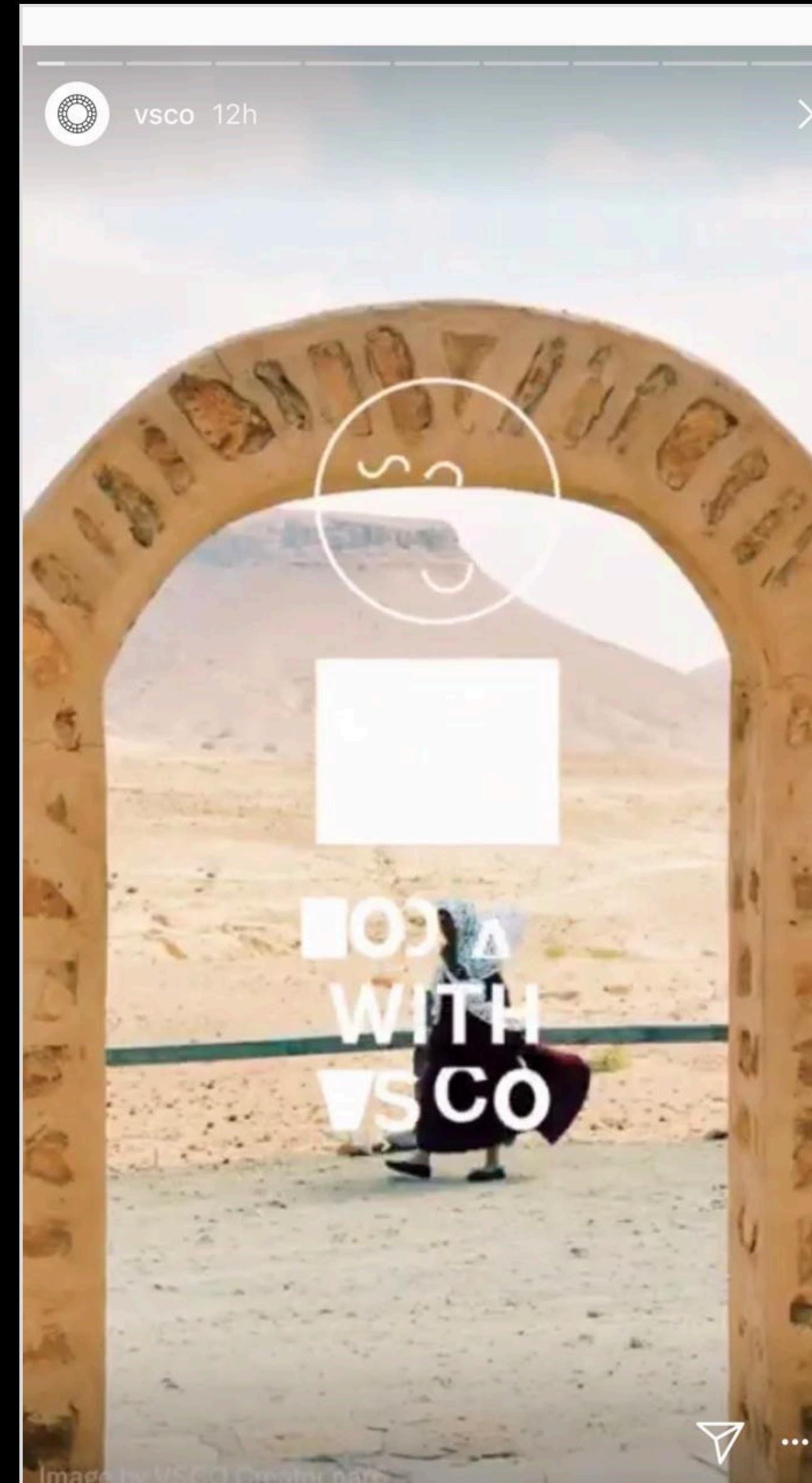
X	0	▲	Y	20	▲
W	375	▼	H	647	▼

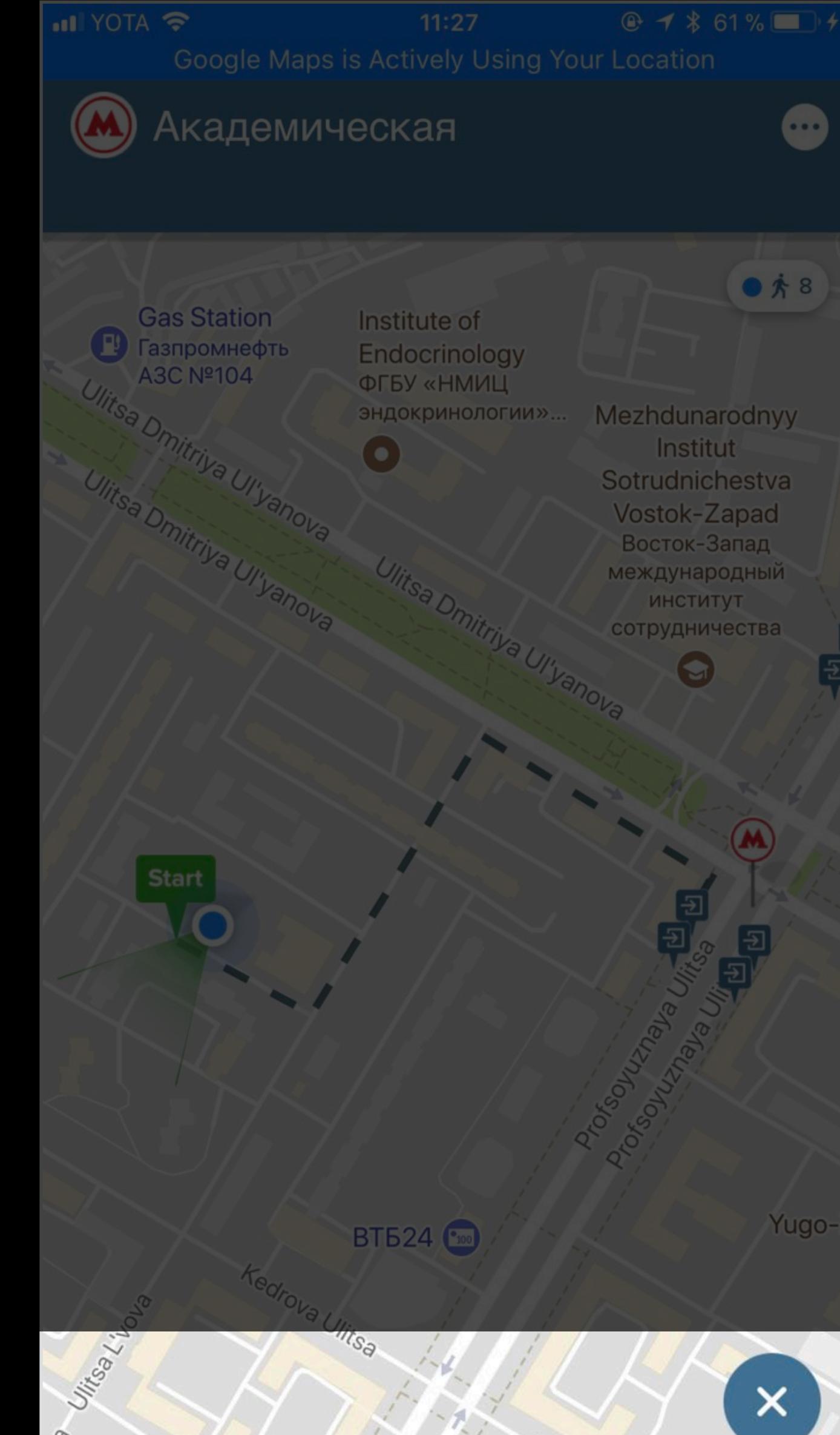
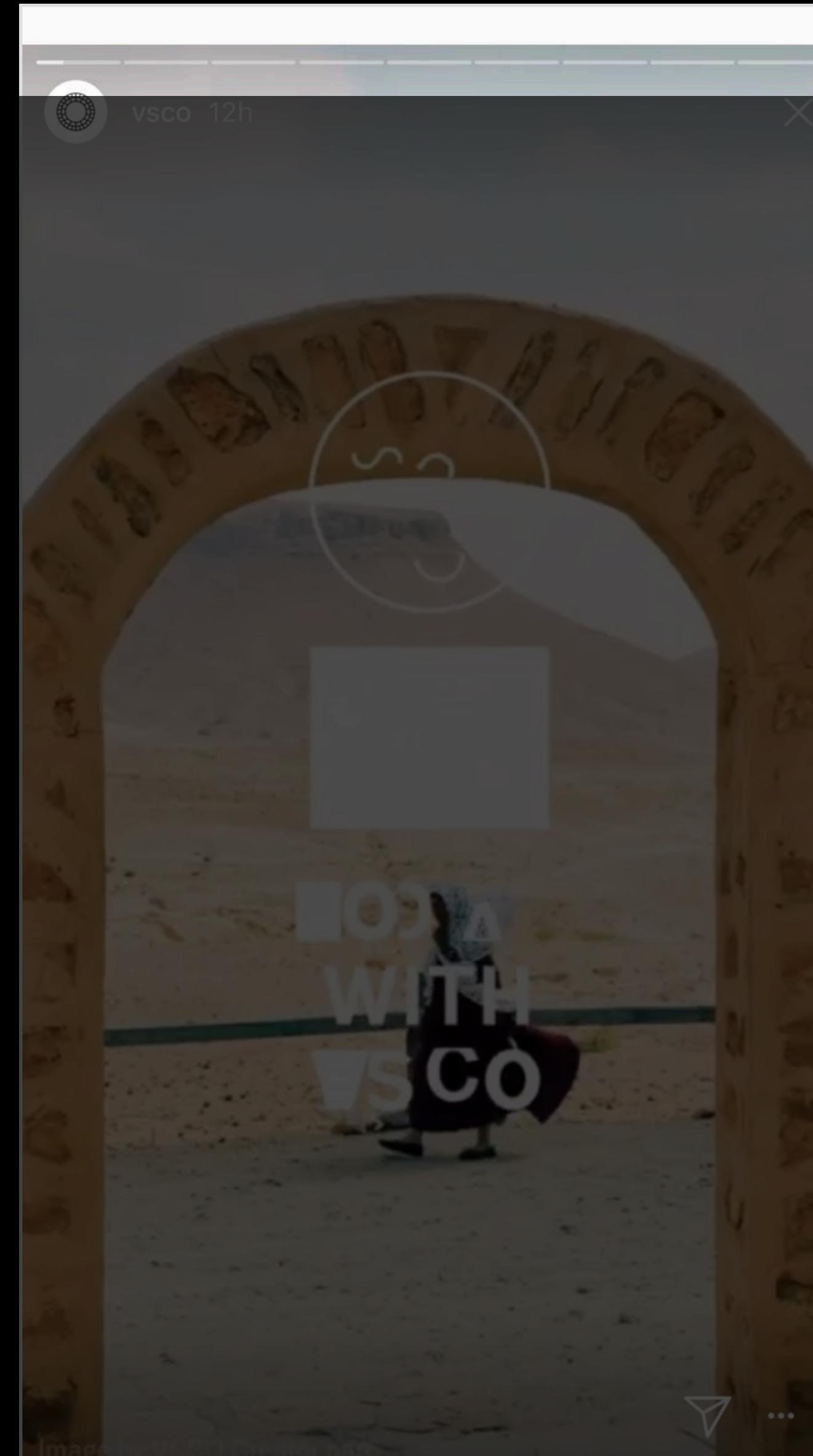


<https://double-high-status-bar.tumblr.com>



<https://double-high-status-bar.tumblr.com>

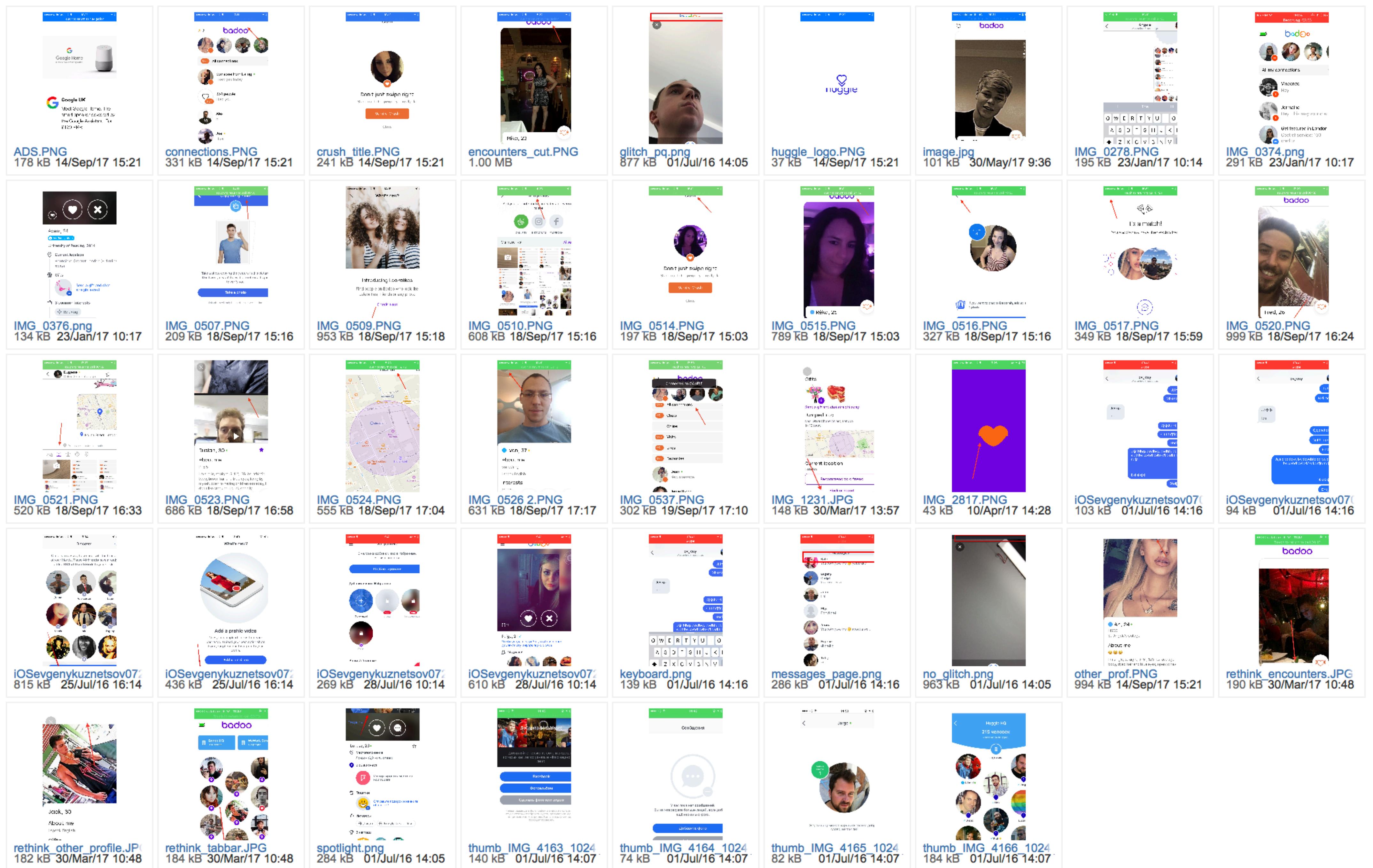


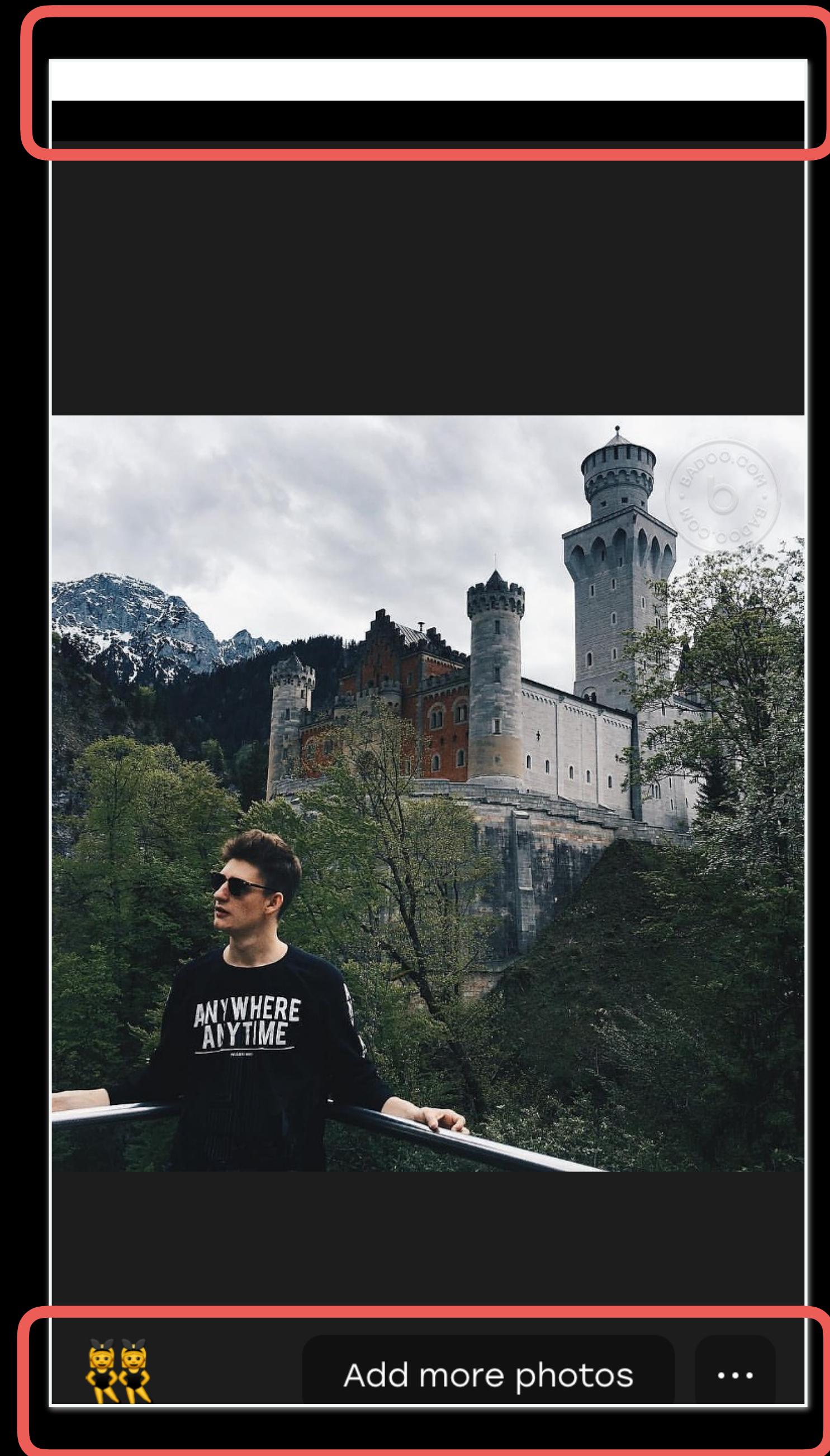
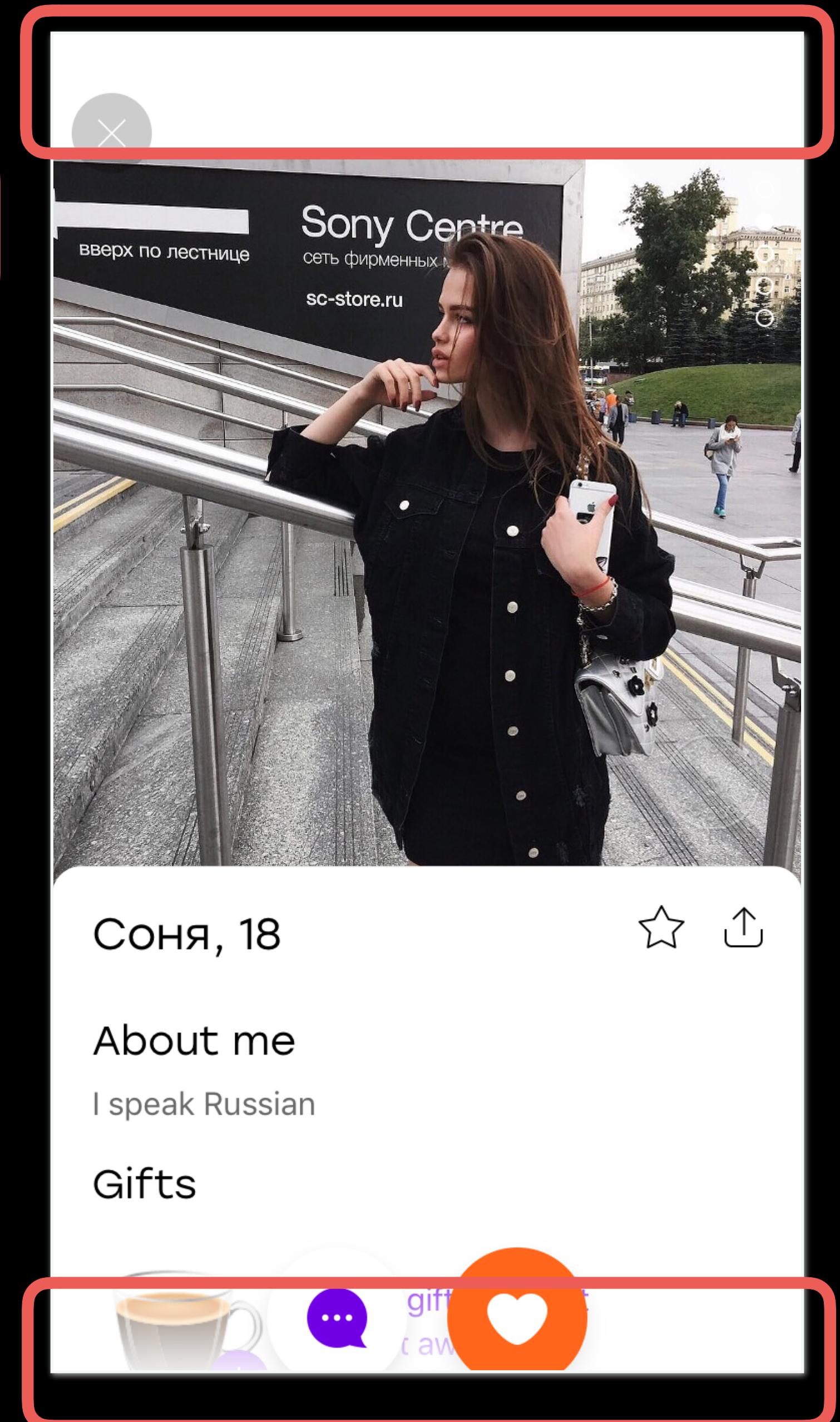
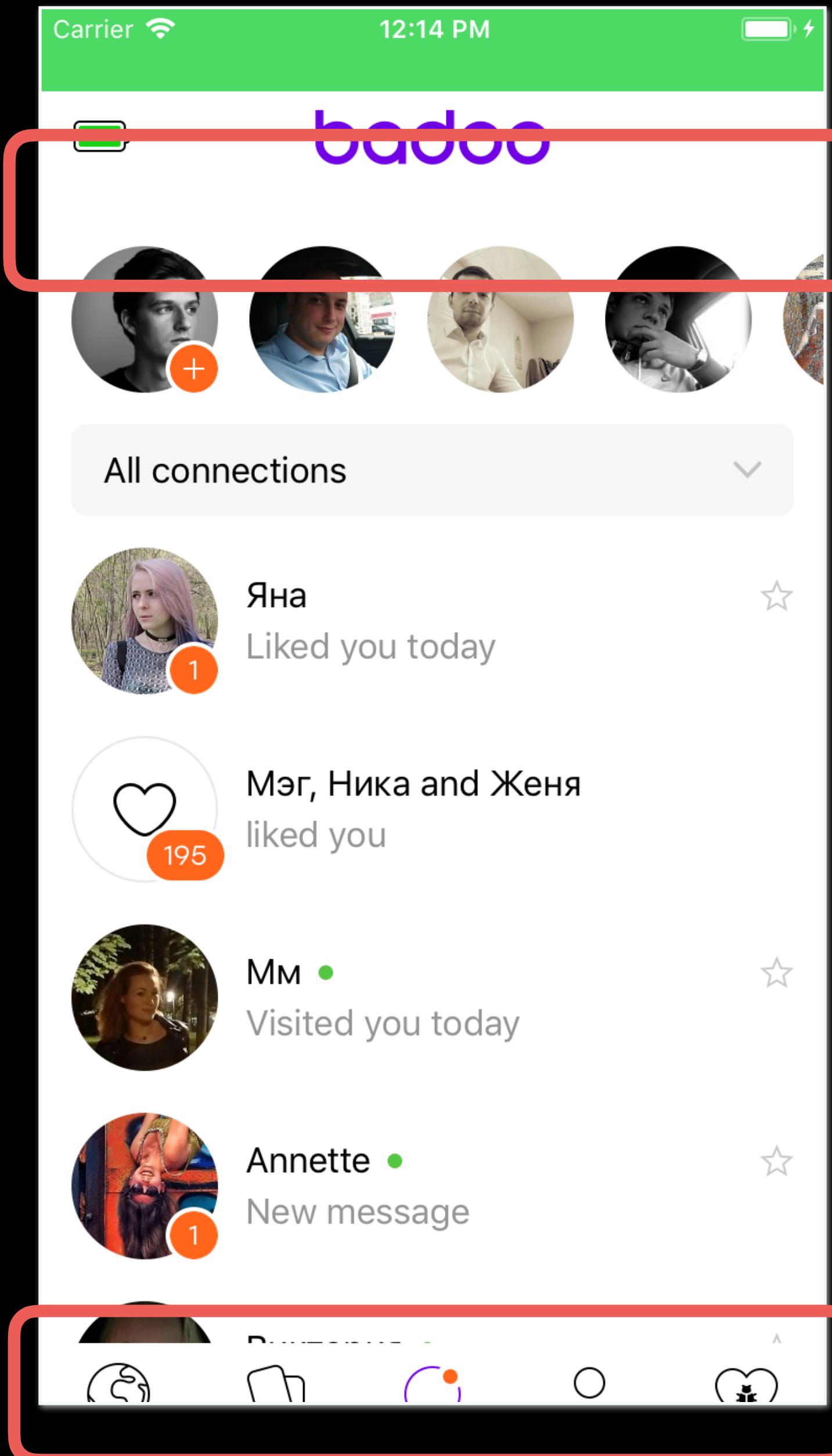


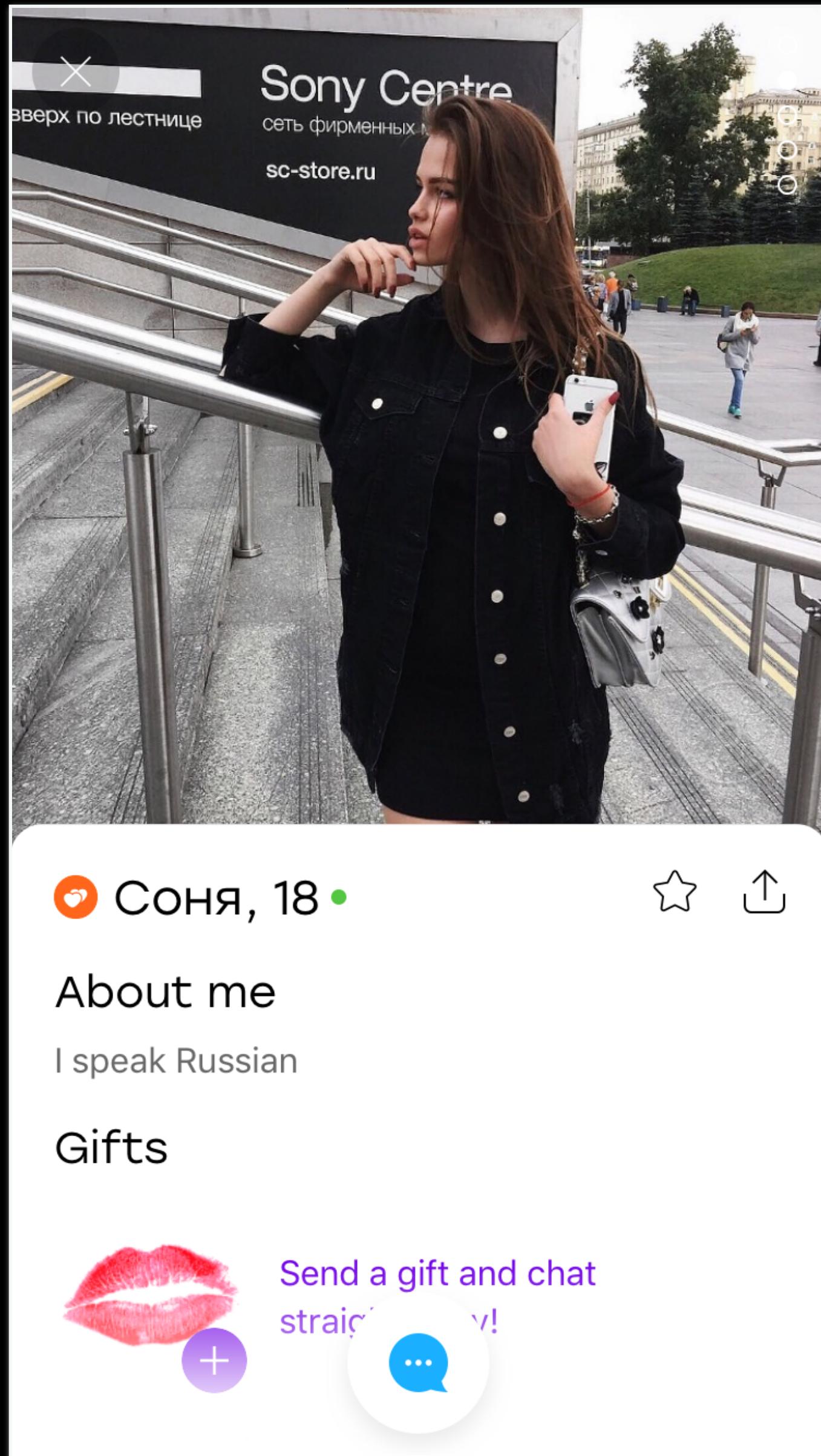
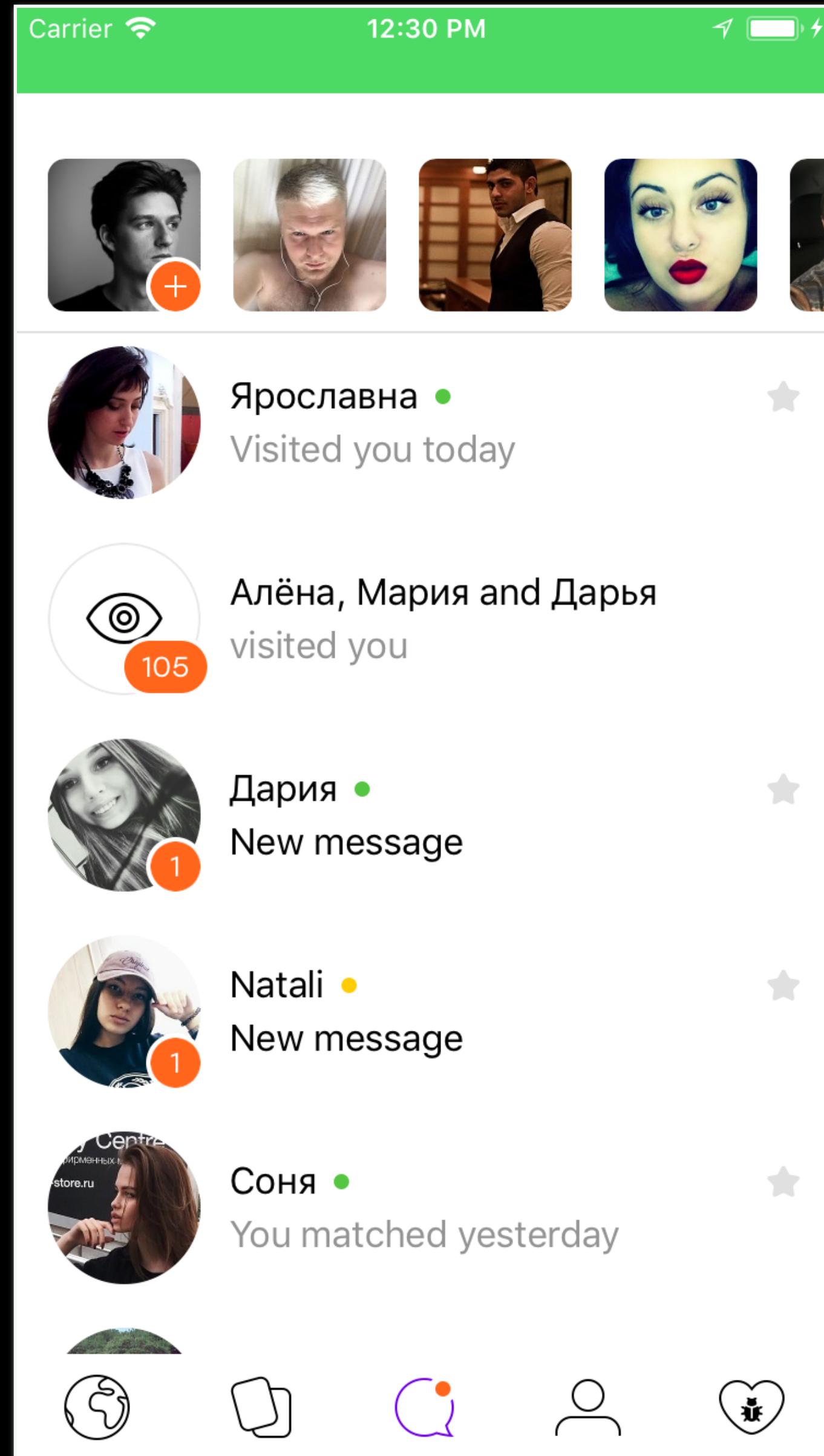
# How could we fix it?

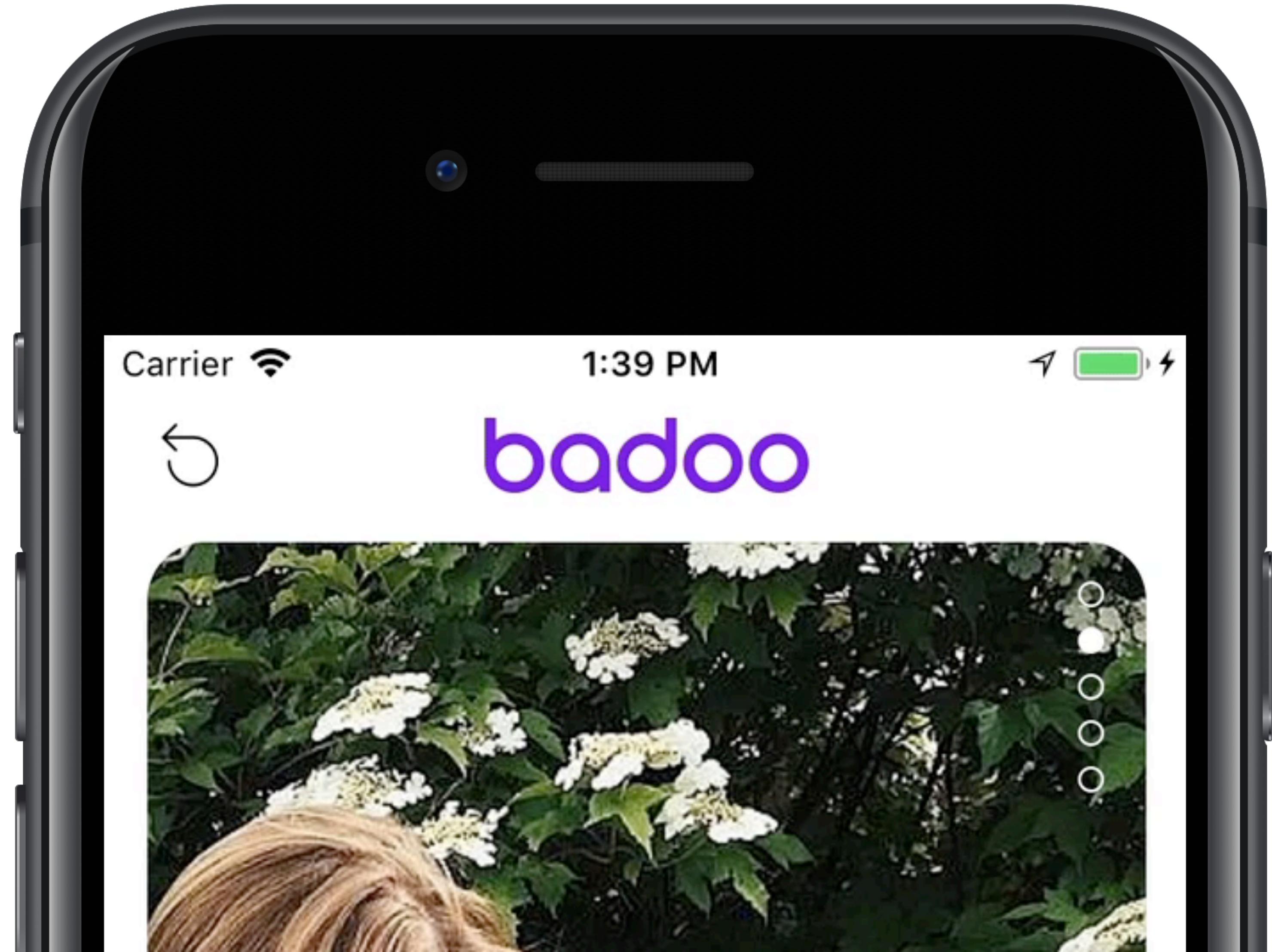


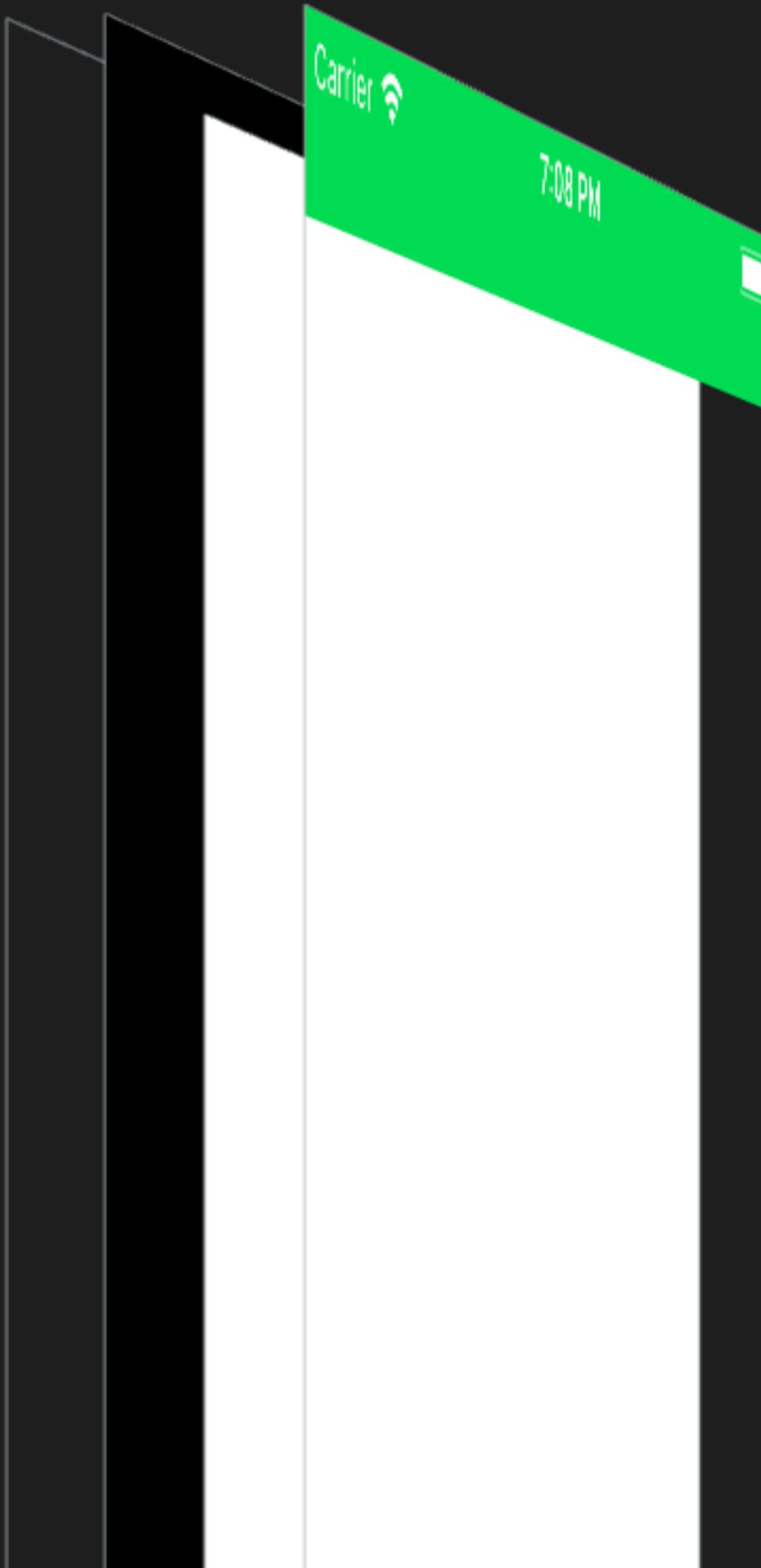
## Attachments

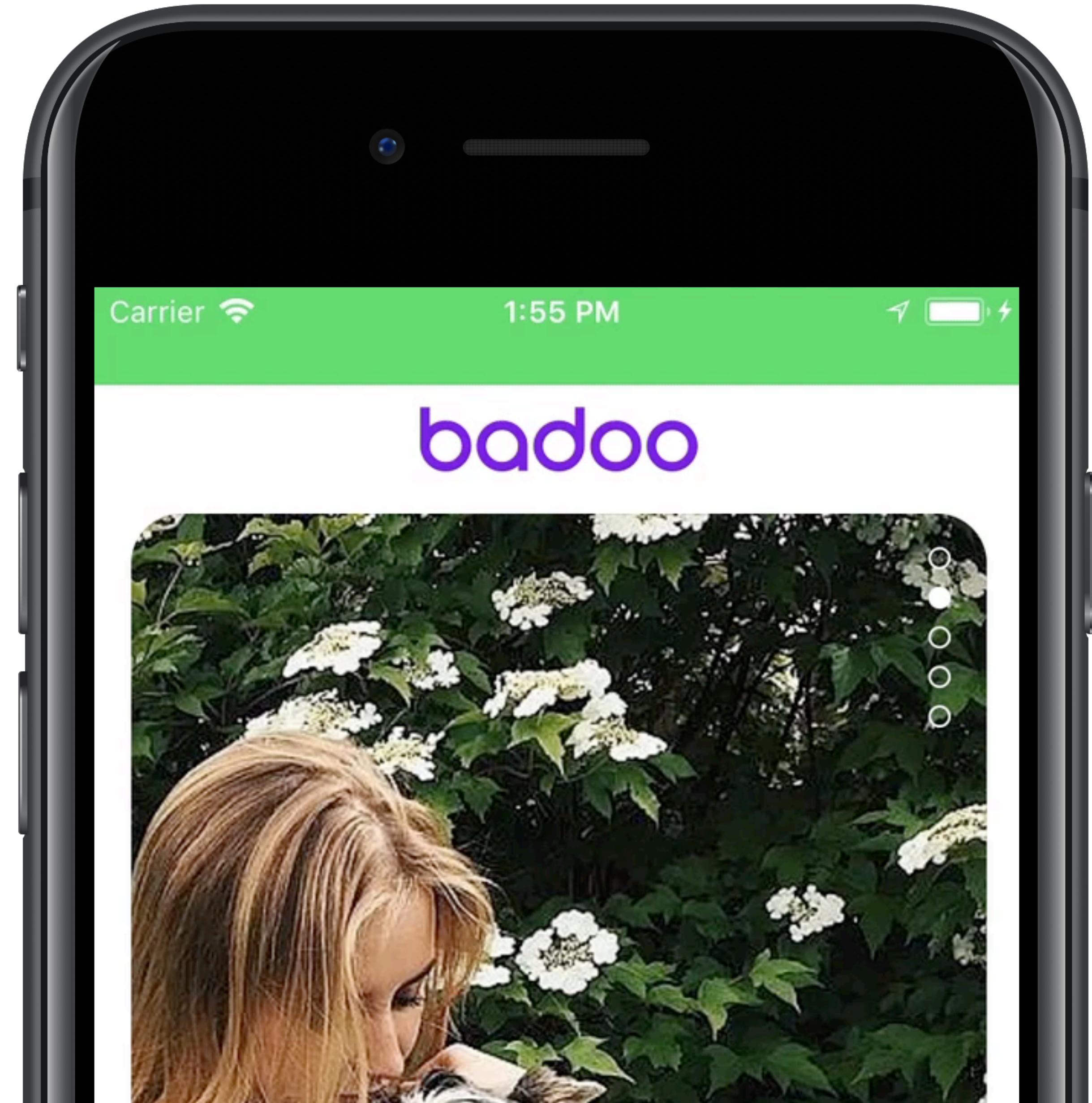














## Layout

Frame	X	0	Y	20
	W	375	H	598
Bounds	X	0	Y	0
	W	375	H	598
Center	X	187.5	Y	319
Transform	[1, 0, 0, 1, 0, 0]			
Layout Margins	T	20	B	0
	L	16	R	16
<input type="checkbox"/> Preserves Superview Layou...				

Semantic Cont. Attr. Unspecified

## Autoresizing

Mask	Width, Height
<input checked="" type="checkbox"/> Autoresizes Subviews	
<input checked="" type="checkbox"/> Translates Autoresizing Ma...	

## Auto Layout

Content Hugging	H	250	V	250
Comp. Resistance	H	750	V	750
Intrinsic Size	W	-1	H	-1
Alignment Rect	X	0	Y	20
	W	375	H	598

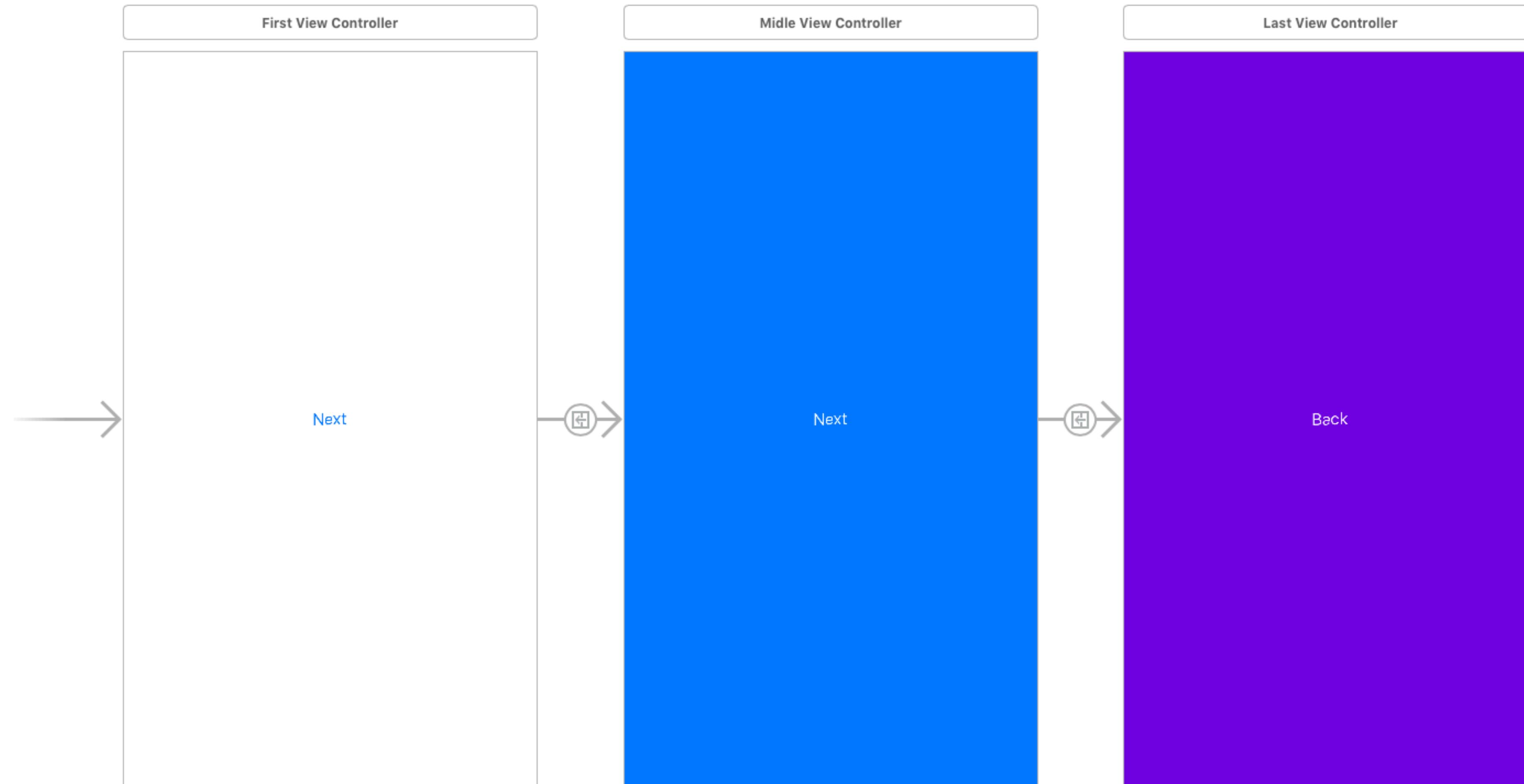
## Alignment Rect Insets

T	0	B	0
L	0	R	0

## Participating Constraints

►  **NSAutoresizingMaskLayoutConstraint**

**OK, so we should avoid usage of  
private API?**



```
window?.backgroundColor = UIColor.white

class FirstViewController: UIViewController { }

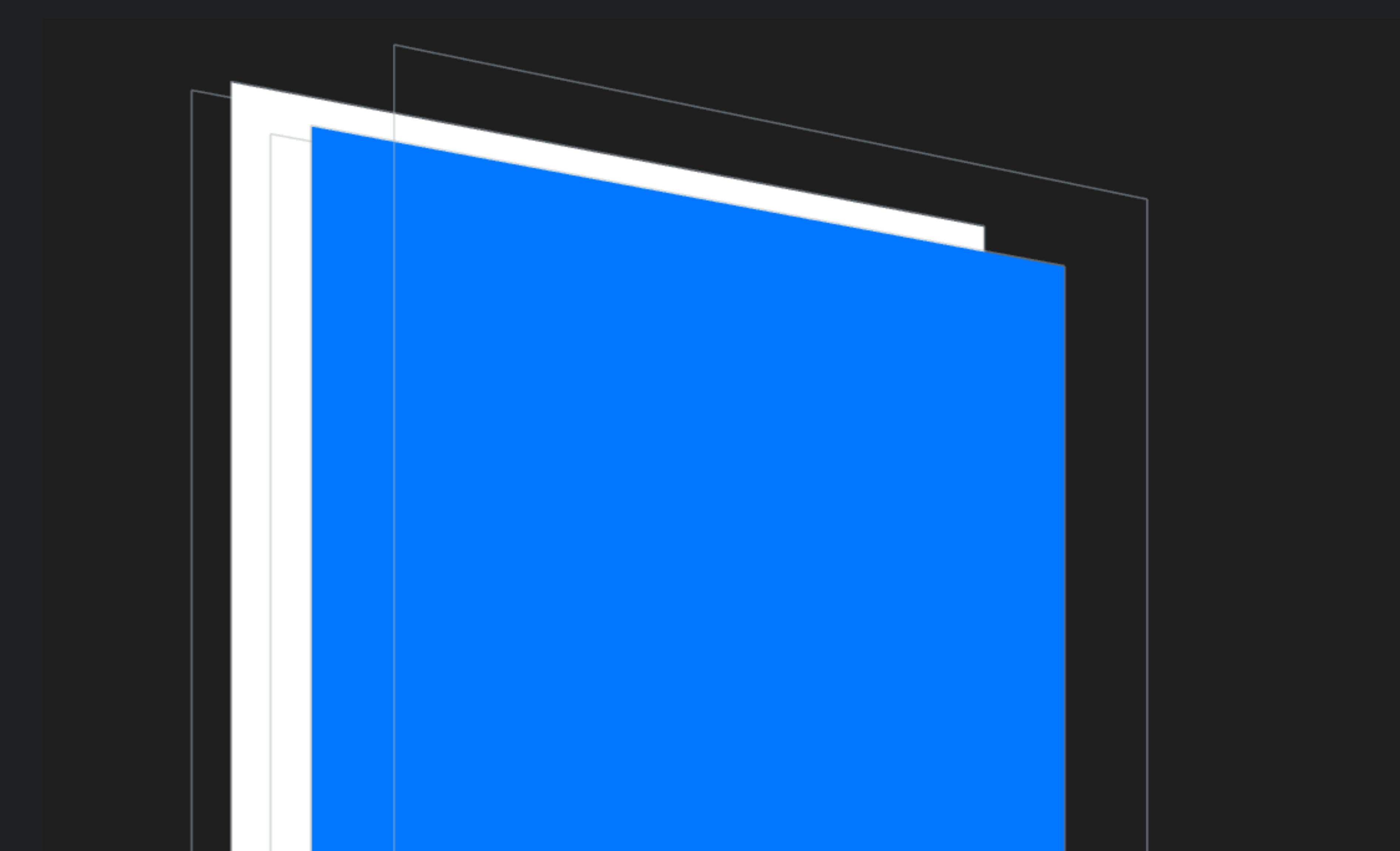
class MidleViewController: UIViewController {
    override var prefersStatusBarHidden: Bool {
        return true
    }
}

class LastViewController: UIViewController {
    override var prefersStatusBarHidden: Bool {
        return false
    }
}
```



Carrier 2:05 PM

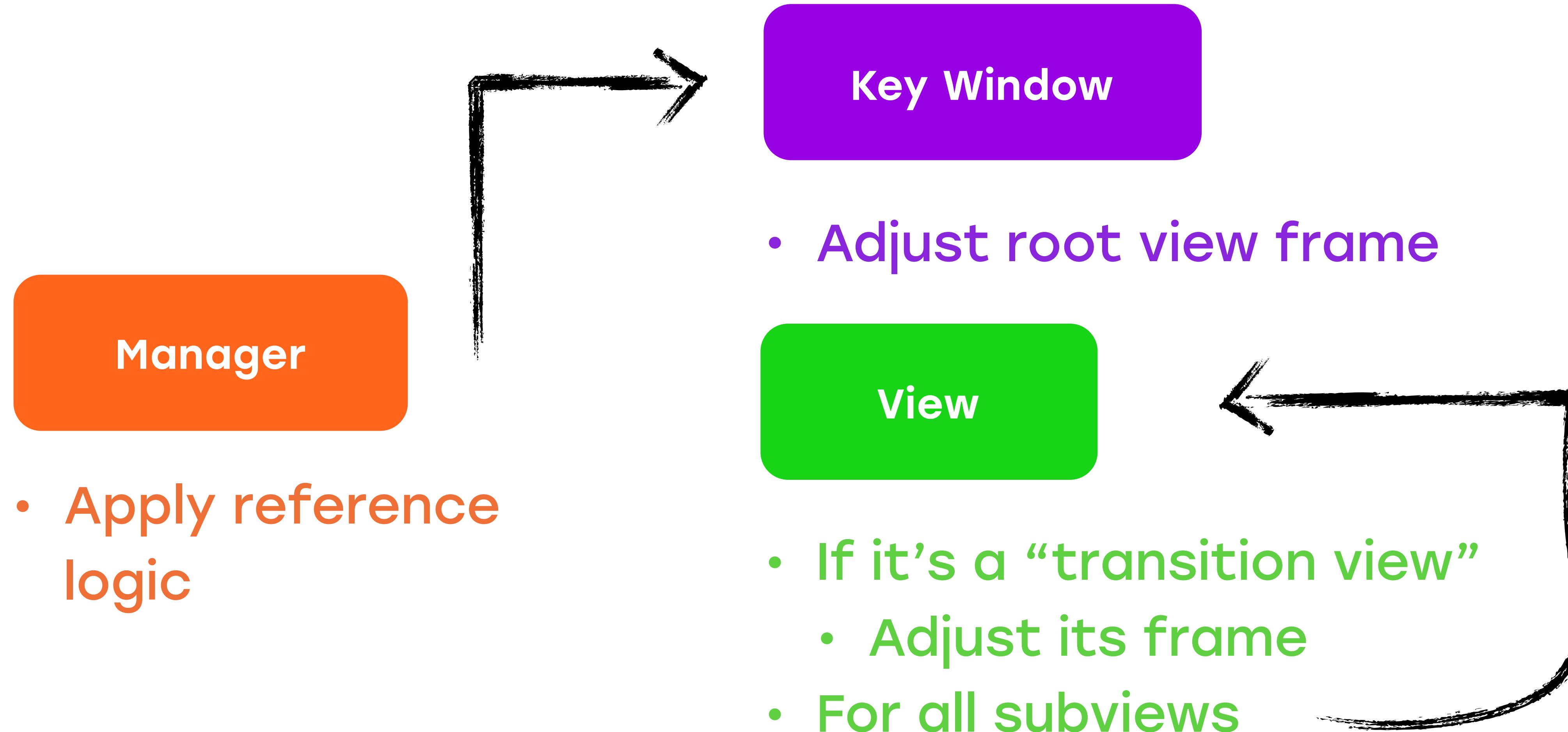
Next



**What should we do?**



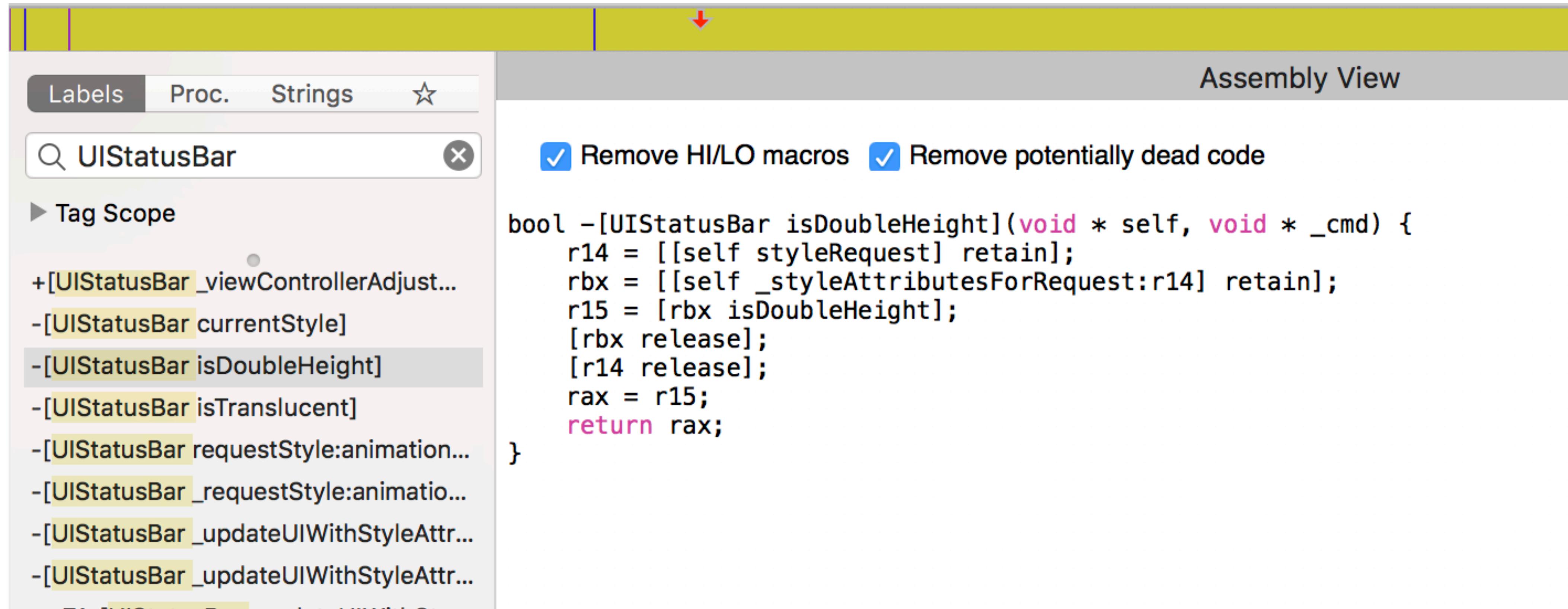
DIY



```
func adjustFrame(of view: UIView,  
                 statusBarHidden: Bool,  
                 baseView: UIView) {  
  
    if statusBarHidden && view.isFullScreen {  
        view.frame = baseView.bounds  
    } else if view.isFullScreen {  
        let inCallStatusBarOffset = StatusBarManager.shared.inCallStatusBarOffset  
        view.frame = CGRect(x: 0,  
                            y: inCallStatusBarOffset,  
                            width: baseView.bounds.width,  
                            height: baseView.bounds.height - inCallStatusBarOffset)  
    }  
}
```

```
func forceAdjustmentsOfViews(isStatusBarHidden: Bool) {  
    guard StatusBarManager.shared.isInCallStatusBar else {  
        return  
    }  
  
    for view in self.subviews {  
        self.adjustFrame(of: view, isStatusBarHidden:  
isStatusBarHidden, baseView: self)  
    }  
  
    self.forceAdjustmentsOfViewsRecursive(isStatusBarHidden:  
isStatusBarHidden, baseView: self)  
}
```

```
func forceAdjustmentsOfViews(isStatusBarHidden: Bool) {  
    guard StatusBarManager.shared.isInCallStatusBar else {  
        return  
    }  
  
    for view in self.subviews {  
        self.adjustFrame(of: view, isStatusBarHidden:  
isStatusBarHidden, baseView: self)  
    }  
  
    self.forceAdjustmentsOfViewsRecursive(isStatusBarHidden:  
isStatusBarHidden, baseView: self)  
}
```



The screenshot shows the Assembly View of Valgrind. The search bar at the top left contains the text "UIStatusBar". The tabs at the top right are "Labels", "Proc.", "Strings", and a star icon. The "Assembly View" tab is selected.

Two checkboxes are checked: "Remove HI/LO macros" and "Remove potentially dead code".

The assembly code shown is:

```
bool -[UIStatusBar isDoubleHeight](void * self, void * _cmd) {  
    r14 = [[self styleRequest] retain];  
    rbx = [[self _styleAttributesForRequest:r14] retain];  
    r15 = [rbx isDoubleHeight];  
    [rbx release];  
    [r14 release];  
    rax = r15;  
    return rax;  
}
```

```
func forceAdjustmentsOfViews(isStatusBarHidden: Bool) {  
    guard StatusBarManager.shared.isInCallStatusBar else {  
        return  
    }  
  
    for view in self.subviews {  
        self.adjustFrame(of: view, isStatusBarHidden:  
isStatusBarHidden, baseView: self)  
    }  
  
    self.forceAdjustmentsOfViewsRecursive(isStatusBarHidden:  
isStatusBarHidden, baseView: self)  
}
```

```
func forceAdjustmentsOfViews(isStatusBarHidden: Bool) {  
    guard StatusBarManager.shared.isInCallStatusBar else {  
        return  
    }  
  
    for view in self.subviews {  
        self.adjustFrame(of: view, isStatusBarHidden:  
isStatusBarHidden, baseView: self)  
    }  
  
    self.forceAdjustmentsOfViewsRecursive(isStatusBarHidden:  
isStatusBarHidden, baseView: self)  
}
```



```
func forceAdjustmentsOfViewsRecursive(isStatusBarHidden: Bool,  
                                      baseView: UIView) {  
    for view in self.subviews {  
        if view.isTransitionView {  
            self.adjustFrame(of: view,  
                             isStatusBarHidden: isStatusBarHidden,  
                             baseView: baseView)  
        }  
        view.forceAdjustmentsOfViewsRecursive(isStatusBarHidden: isStatusBarHidden,  
                                              baseView: baseView)  
    }  
  
    private var isTransitionView: Bool {  
        let viewTypeNome = String(describing: type(of: self))  
        return viewTypeNome == "UITransitionView"  
    }  
}
```



Carrier 2:32 PM

badoo



Алина, 19

11 photos

# Thank you!

Let's build awesome apps together



badoo



badoo  
tech



@ziminalex