

# **IN4MATX 133: User Interface Software**

**Lecture 18:**  
**Device Resources & Sensors**

Professor Daniel A. Epstein  
TA Lucas de Melo Silva  
TA Jong Ho Lee

# Question



**What phone, if any, do you own?**

- A I own a recent-ish phone (2016 or newer) running iOS
- B I own a recent-ish phone (2016 or newer) running Android
- C I own an older phone running iOS
- D I own an older phone running Android
- E I don't own a smartphone or I own a different device

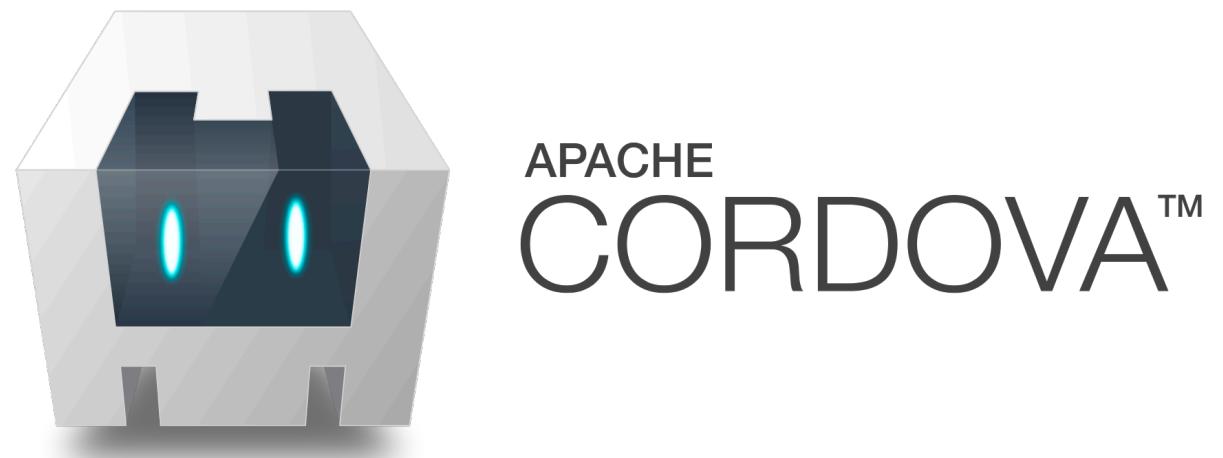
# Today's goals

By the end of today, you should be able to...

- Use Ionic's devApp to test an app on a mobile device
- Access device resources using an Ionic Native Plugin
- Describe some of the sensors on modern smartphones
- Describe some ways in which sensors can be used

# Ionic Native

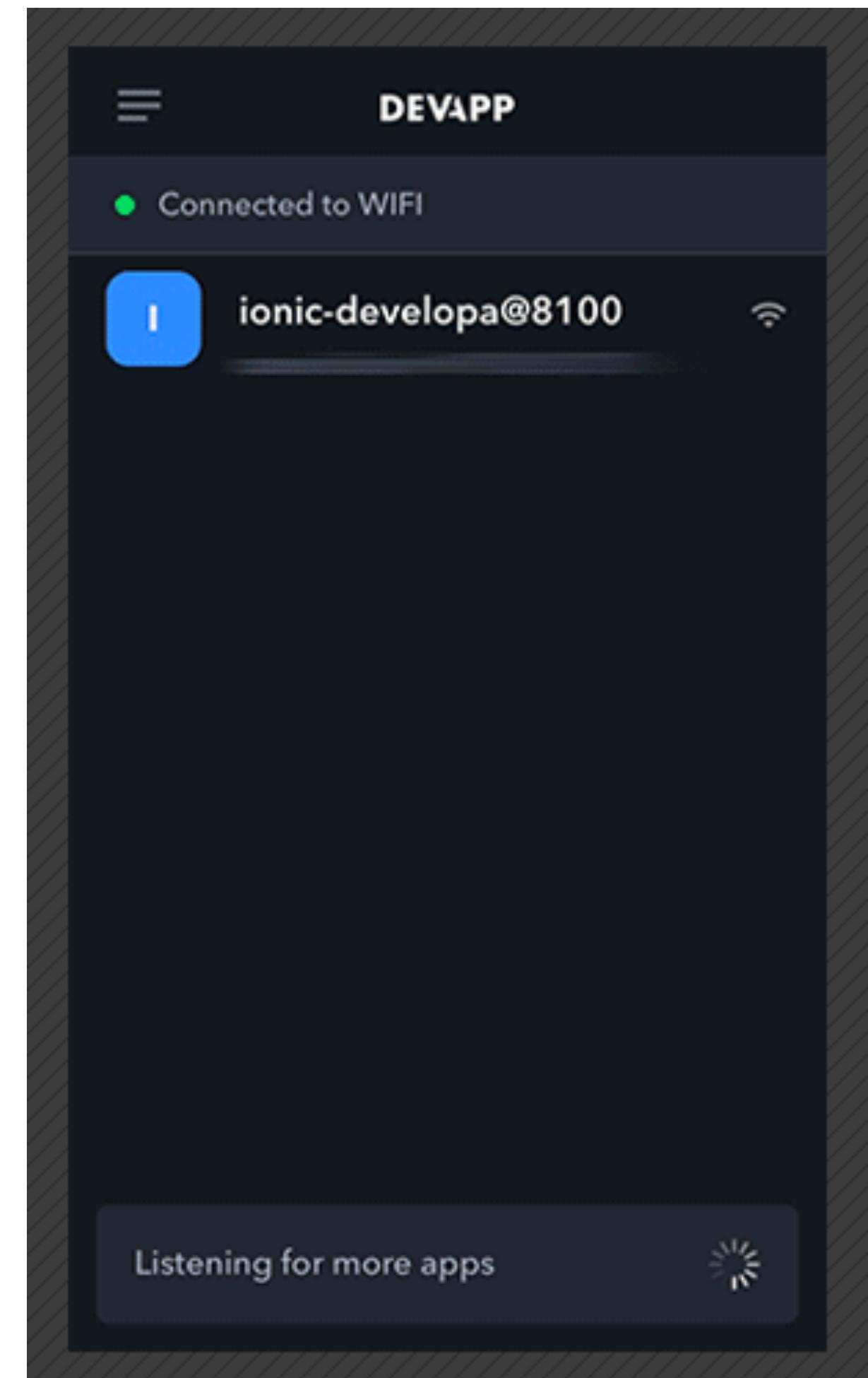
- Apache Cordova also provides libraries for connecting to device resources
- Ionic Native brings these libraries to Ionic as plugins
  - Ionic Native plugins are imported as services
  - Hundreds of plugins



<https://ionicframework.com/docs/native/>

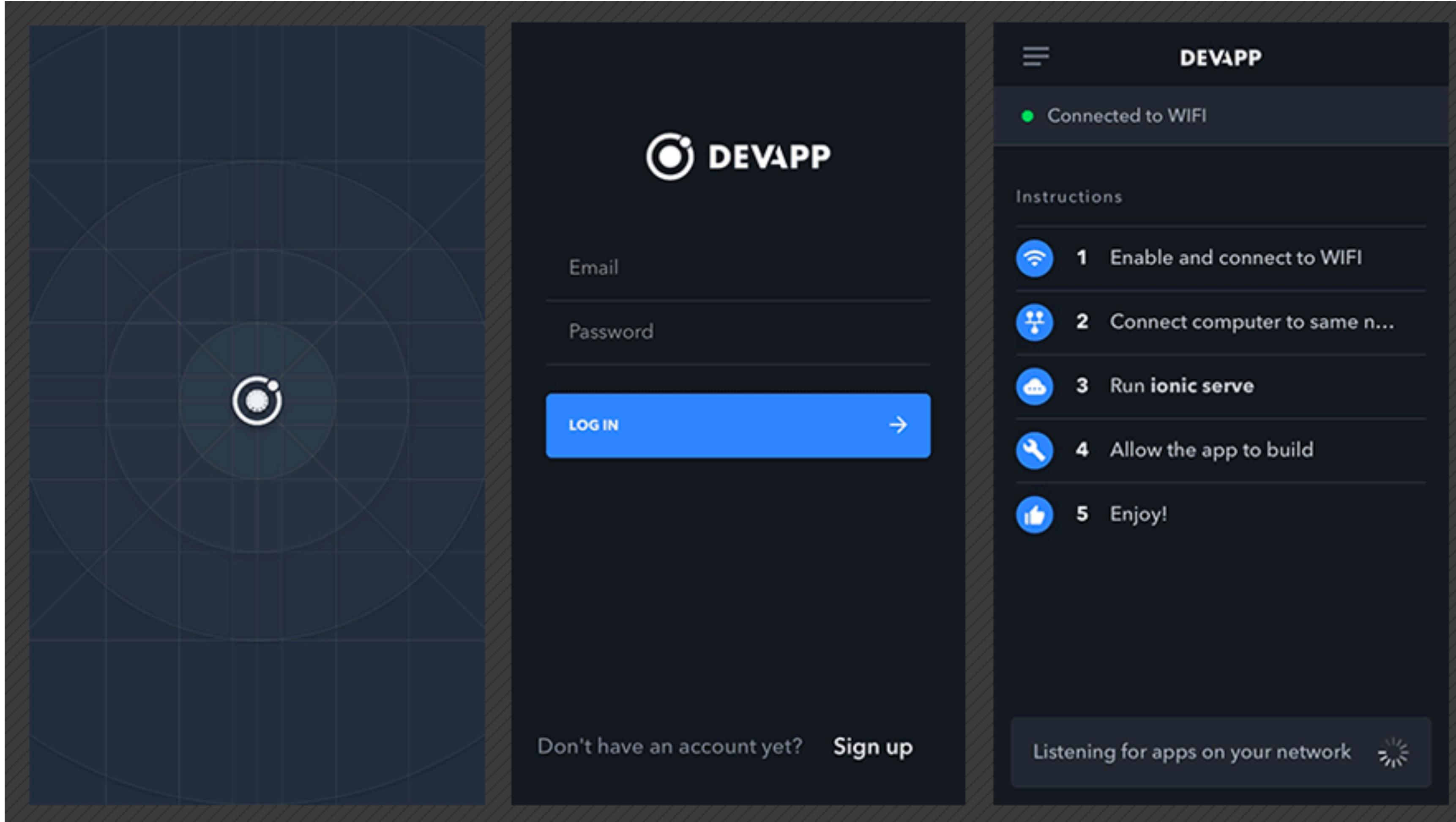
# Ionic DevApp

- An app on Android and iOS made by Ionic
- Provides a WebView to open up Ionic apps
  - Lets you test your Ionic app on an actual device
  - Has some Ionic Native plugins, but not all



<https://ionicframework.com/docs/pro/devapp/>

# Ionic DevApp



Or run ionic lab

<https://ionicframework.com/docs/pro/devapp/>

# Ionic DevApp

## Native Cordova Plugin Support

Ionic DevApp currently supports the following plugins:

```
card.io.cordova.mobilesdk 2.1.0 "CardIO"
com-intel-security-cordova-plugin 2.0.3 "APP Security API"
com.darktalker.cordova.screenshot 0.1.5 "Screenshot"
com.paypal.cordova.mobilesdk 3.5.0 "PayPalMobile"
cordova-admob-sdk 0.8.0 "AdMob SDK"
cordova-base64-to-gallery 4.1.2 "base64ToGallery"
cordova-instagram-plugin 0.5.5 "Instagram"
cordova-launch-review 2.0.0 "Launch Review"
cordova-plugin-3dtouch 1.3.5 "3D Touch"
cordova-plugin-actionsheet 2.3.3 "ActionSheet"
cordova-plugin-add-swift-support 1.6.2 "AddSwiftSupport"
cordova-plugin-admob-free 0.10.0 "Cordova AdMob Plugin"
cordova-plugin-advanced-http 1.8.1 "Advanced HTTP plugin"
cordova-plugin-app-event 1.2.0 "Application Events"
cordova-plugin-apprate 1.3.0 "AppRate"
cordova-plugin-battery-status 1.2.4 "Battery"
cordova-plugin-ble-central 1.1.4 "BLE"
cordova-plugin-bluetooth-serial 0.4.7 "Bluetooth Serial"
cordova-plugin-brightness 0.1.5 "Brightness"
cordova-plugin-calendar 4.6.0 "Calendar"
cordova-plugin-camera 2.4.1 "Camera"
cordova-plugin-compat 1.1.0 "Compat"
cordova-plugin-contacts 2.3.1 "Contacts"
cordova-plugin-datepicker 0.9.3 "DatePicker"
cordova-plugin-device 1.1.6 "Device"
cordova-plugin-device-motion 1.2.5 "Device Motion"
cordova-plugin-device-orientation 1.0.7 "Device Orientation"
cordova-plugin-dialogs 1.3.3 "Notification"
cordova-plugin-email-composer 0.8.7 "EmailComposer"
cordova-plugin-geolocation 2.4.3 "Geolocation"
cordova-plugin-globalization 1.0.7 "Globalization"
```

```
cordova-plugin-media 3.0.1 "Media"
cordova-plugin-mixpanel 3.1.0 "Mixpanel"
cordova-plugin-music-controls 2.0.0 "MusicControls"
cordova-plugin-nativeaudio 3.0.9 "Cordova Native Audio"
cordova-plugin-nativestorage 2.2.2 "NativeStorage"
cordova-plugin-network-information 1.3.3 "Network Information"
cordova-plugin-request-location-accuracy 2.2.1 "Request Location Accuracy"
cordova-plugin-safariviewcontroller 1.4.7 "SafariViewController"
cordova-plugin-screen-orientation 2.0.1 "Screen Orientation"
cordova-plugin-secure-storage 2.6.8 "SecureStorage"
cordova-plugin-shake 0.6.0 "Shake Gesture Detection"
cordova-plugin-sim 1.3.3 "SIM"
cordova-plugin-splashscreen 4.0.3 "Splashscreen"
cordova-plugin-statusbar 2.2.4-dev "StatusBar"
cordova-plugin-stripe 1.5.3 "cordova-plugin-stripe"
cordova-plugin-taptic-engine 2.1.0 "Taptic Engine"
cordova-plugin-themeablebrowser 0.2.17 "ThemeableBrowser"
cordova-plugin-touch-id 3.2.0 "Touch ID"
cordova-plugin-tts 0.2.3 "TTS"
cordova-plugin-vibration 2.1.5 "Vibration"
cordova-plugin-whitelist 1.3.2 "Whitelist"
cordova-plugin-x-socialsharing 5.1.8 "SocialSharing"
cordova-plugin-x-toast 2.6.0 "Toast"
cordova-plugin-zip 3.1.0 "cordova-plugin-zip"
```

• • •

<https://ionicframework.com/docs/pro/devapp/>

# Cordova Plugins

- Some (few) are maintained by Apache
- Others (many) are maintained by the community
- As a result, quality varies immensely
- Features may not work as expected

# Three Cordova Plugins

- Device Motion (Apache-maintained)
- Local Notification (Community-maintained)
- Social Sharing (Community-maintained)

# Device Motion

- Goal: get data from the phone's accelerometer every second
- Could be used to estimate how restful someone's sleep was, for example

# Device Motion

- ionic cordova plugin add cordova-plugin-device-motion
- npm install @ionic-native/device-motion
- Add as a provider to the app's module (app.module.ts)

```
import { DeviceMotion } from '@ionic-native/device-motion/ngx';

@NgModule({
  providers: [
    ...
    DeviceMotion,
    ...
  ],
})
```

<https://ionicframework.com/docs/native/device-motion>

# Device Motion

## Instance Members

### getCurrentAcceleration

Get the current acceleration along the x, y, and z axes.

### watchAcceleration

Watch the device acceleration. Clear the watch by unsubscribing from the observable.

## Parameters

### options

See DeviceMotionAccelerometerOptions table below

Type: `DeviceMotionAccelerometerOptions`

optional

## DeviceMotionAccelerationData

### timestamp

Creation timestamp in milliseconds.

Type: `intrinsic`

### x

Amount of acceleration on the x-axis. (in m/s<sup>2</sup>)

Type: `intrinsic`

### y

Amount of acceleration on the y-axis. (in m/s<sup>2</sup>)

Type: `intrinsic`

### z

Amount of acceleration on the z-axis. (in m/s<sup>2</sup>)

Type: `intrinsic`

## DeviceMotionAccelerometerOptions

### frequency

Requested period of calls to accelerometerSuccess with acceleration data in

Type: `intrinsic`

Milliseconds. Default: 10000 optional

# Device Motion

- DeviceMotion is a Service and can be injected like any other service

```
import { Platform } from '@ionic/angular';
import { DeviceMotion, DeviceMotionAccelerationData } from '@ionic-native/device-motion/ngx';

@Component({
  selector: 'app-home',
  templateUrl: 'home.page.html',
  styleUrls: ['home.page.scss'],
})
export class HomePage {
  acceleration:string = 'No acceleration data';

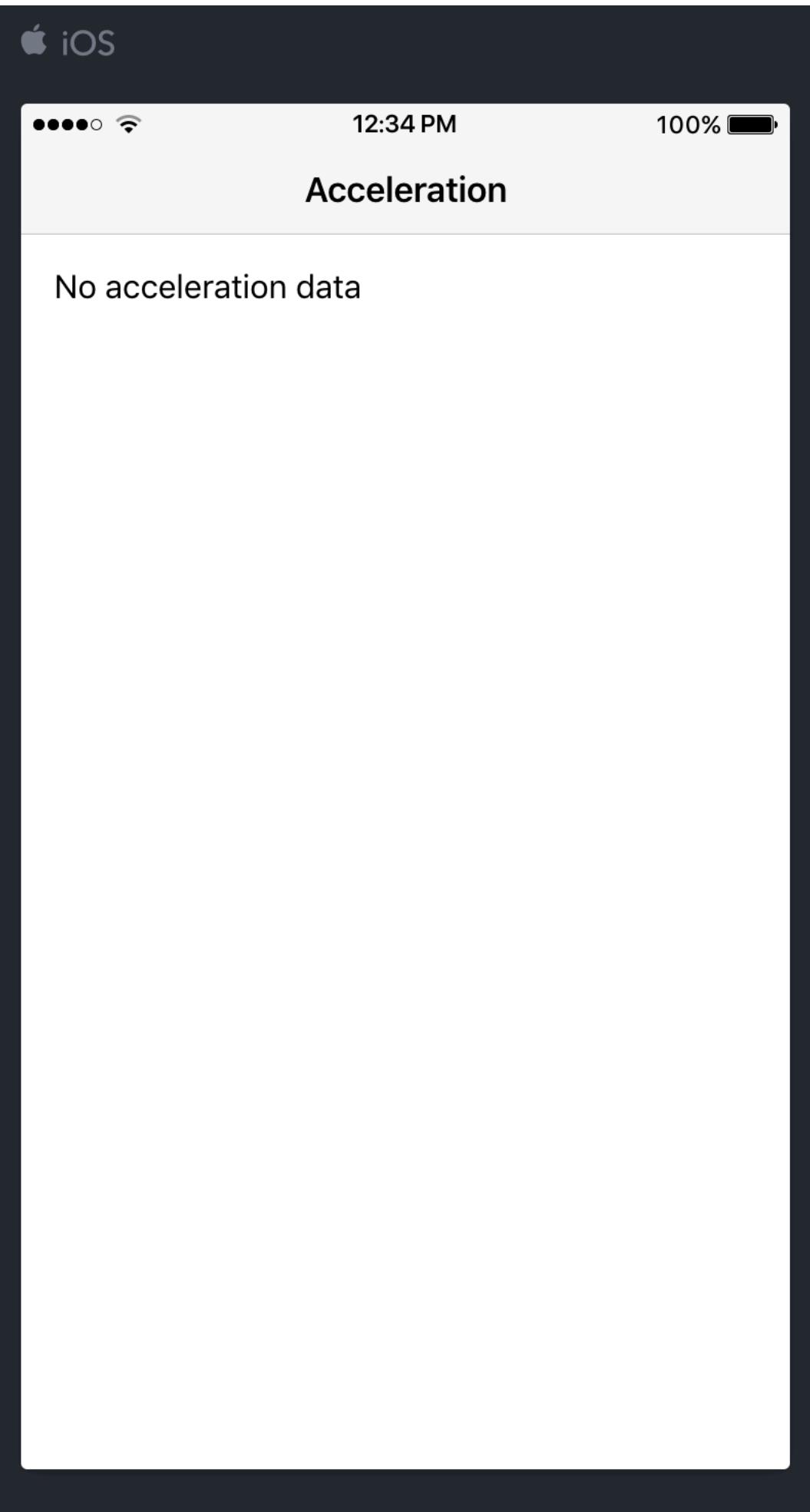
  constructor(private platform:Platform, private deviceMotion:DeviceMotion) {
    this.platform.ready().then(() => {
      this.deviceMotion.watchAcceleration({frequency: 1000}).subscribe((acc) => {
        this.acceleration = acc.x + ' ' + acc.y + ' ' + acc.z;
      });
    });
  }
}
```

<https://ionicframework.com/docs/native/device-motion>

# Device Motion

```
<ion-header>
  <ion-toolbar>
    <ion-title>
      Acceleration
    </ion-title>
  </ion-toolbar>
</ion-header>

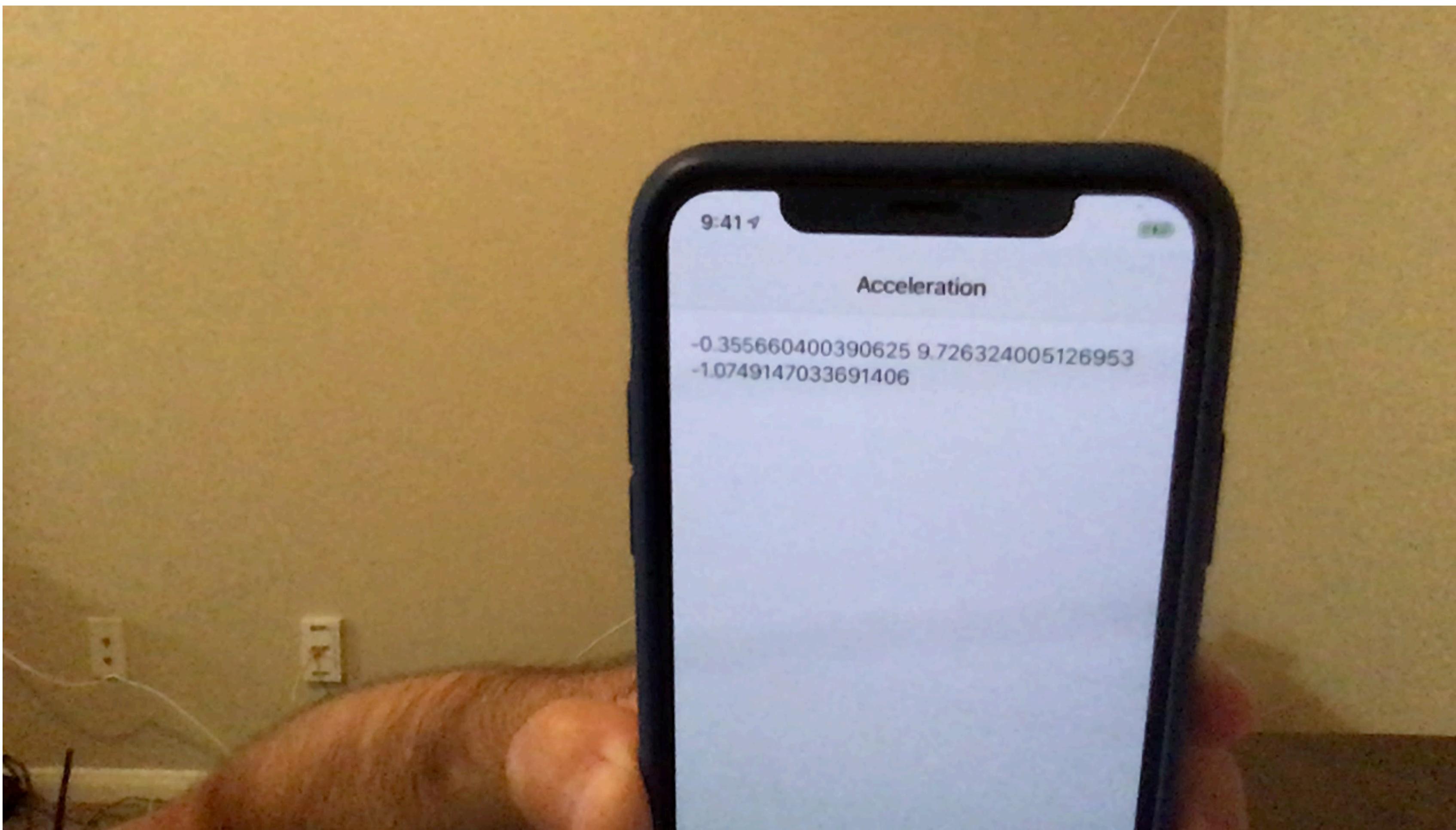
<ion-content padding>
  {{acceleration}}
</ion-content>
```



No data on a computer!

<https://ionicframework.com/docs/native/device-motion>

# Device Motion



# Local Notification

- Goal: send a notification to the phone
- Could be used to remind someone to journal their sleepiness, for example

# Local Notification

## LocalNotifications

### CONTENTS

[Installation](#)  
[Instance Members](#)  
[ILocalNotification](#)  
[ILocalNotificationAction](#)  
[ILocalNotificationProgressBar](#)  
[ILocalNotificationTrigger](#)

This plugin allows you to display local notifications on the device

Repo: <https://github.com/katzer/cordova-plugin-local-notifications>

## Installation

1. Install the Cordova and Ionic Native plugins:

```
$ ionic cordova plugin add cordova-plugin-local-notification
$ npm install --save @ionic-native/local-notifications@beta
```

2. Add this plugin to your app's module

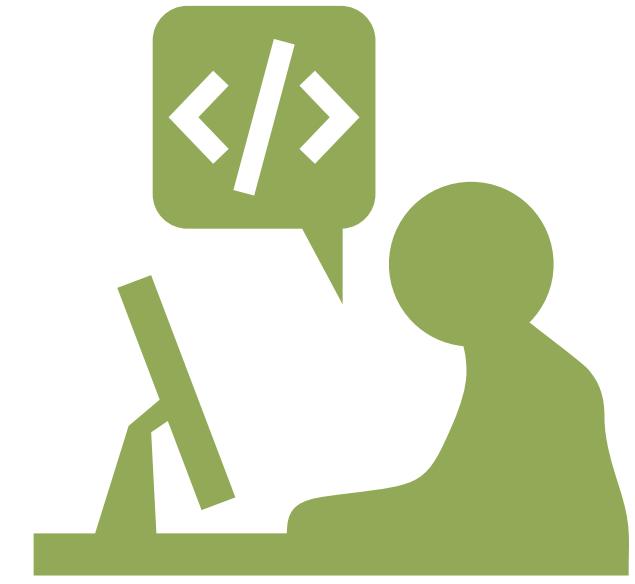
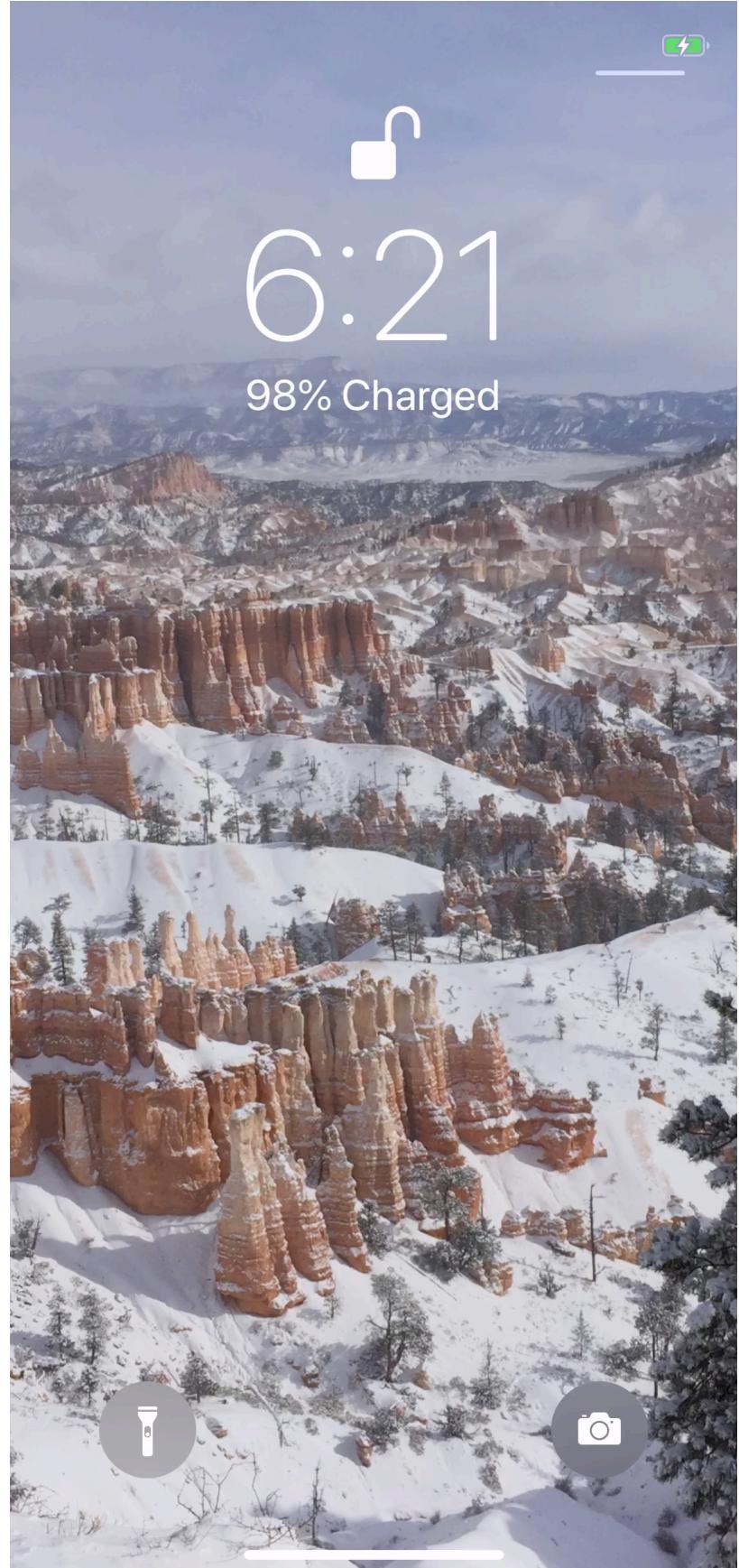
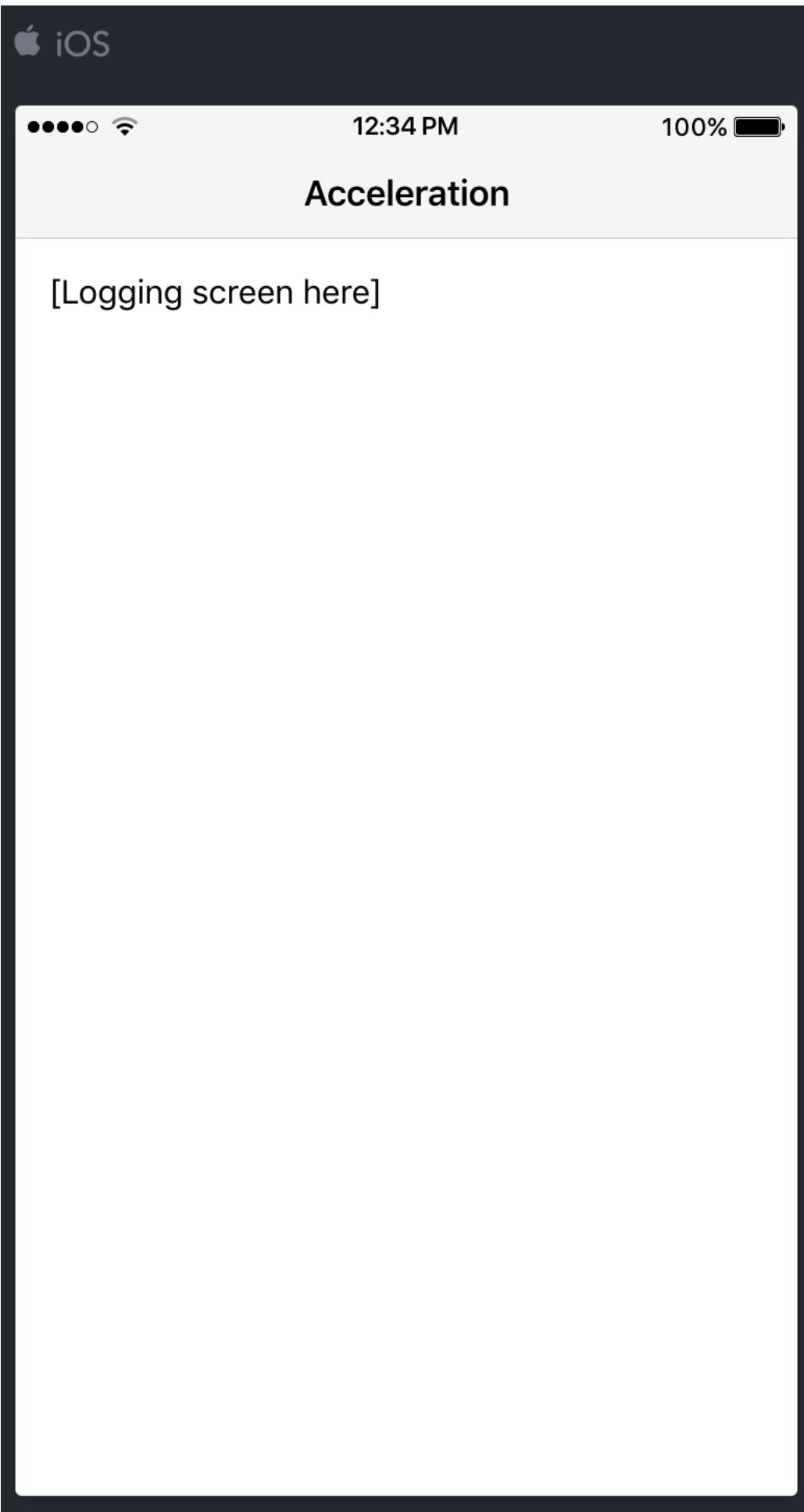
<https://ionicframework.com/docs/native/local-notifications>

# Local Notification

```
this.localNotifications.schedule({
  text: 'Time to log your sleepiness!',
  trigger: {at: new Date(new Date().getTime() + 3600)},
  led: 'FF0000',
  sound: null
});
```

<https://ionicframework.com/docs/native/local-notifications>

# Local Notification



# Local Notification

- There are 225 open issues, many of which are critical to the functionality of the plugin
- For example, I was unsuccessful at getting the “at” functionality to work and had to schedule notifications myself
- Other plugins may be just as unreliable as this one
  - It’s the reality of relying on open-source tools

<https://github.com/katzer/cordova-plugin-local-notifications/issues>

# Social Sharing

- Goal: export data from your app to a social app on the device
- Could be used to share photos to Facebook
- Could be used to share text in a text message

<https://ionicframework.com/docs/native/social-sharing>

# Social Sharing

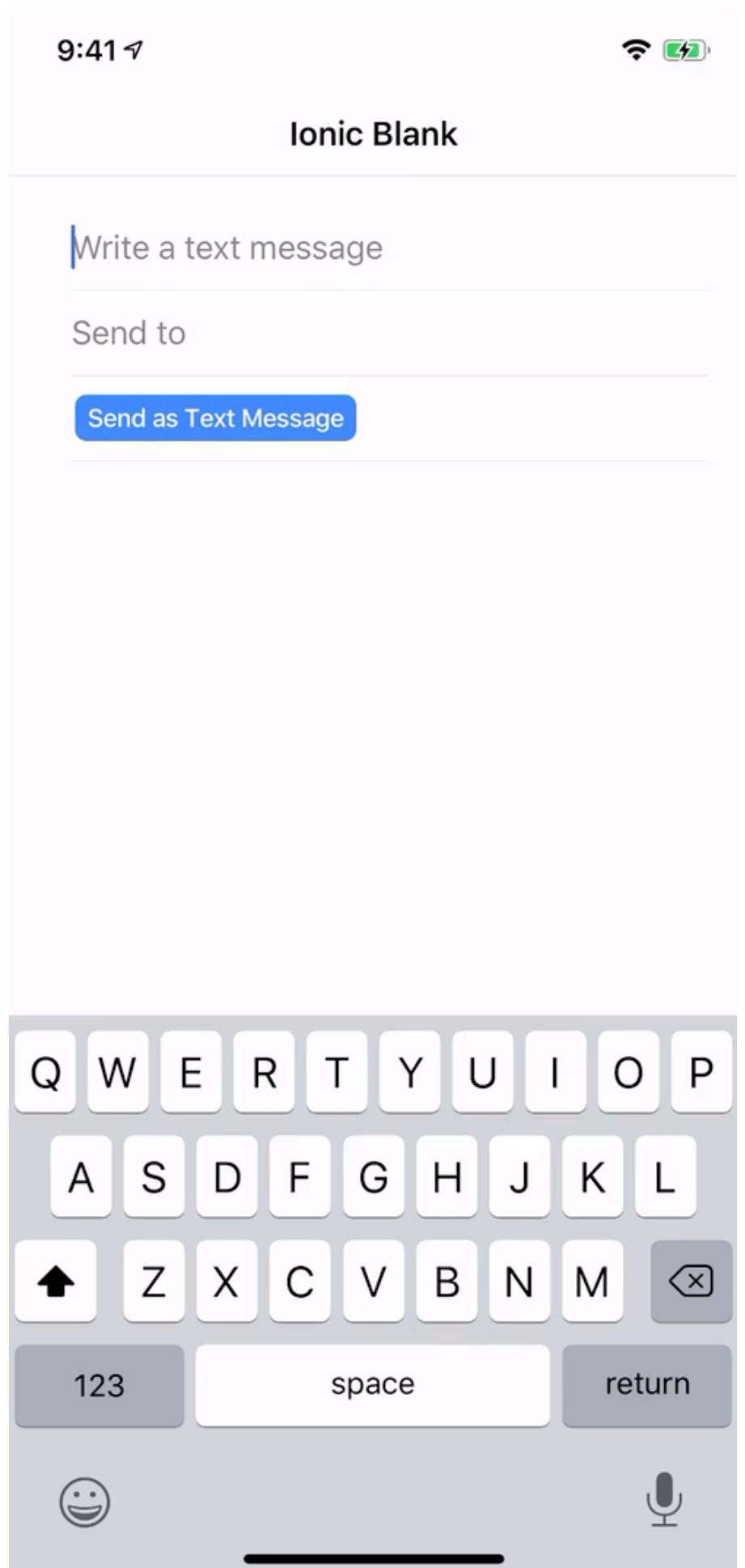
```
this.social.shareViaSMS(this.textMessage, this.phoneNumber);
```

- There are other helper functions for sharing to Facebook, uploading images, etc.

<https://github.com/EddyVerbruggen/SocialSharing-PhoneGap-Plugin>

<https://ionicframework.com/docs/native/social-sharing>

# Social Sharing



<https://www.youtube.com/watch?v=6WTdTwmxyo>  
<https://ionicframework.com/docs/native/social-sharing>

# Premier Plugins

- The company behind Ionic maintains a set of plugins
- A lot of similar functionality
  - Accessing camera
  - Device motion
  - Making notifications
- They are presumably more reliable, but this comes at a cost

Features	Community Plugins \$0/mo	Premier Plugins Contact Us
Maintainer	OSS Community	Ionic
Regular Release Cycles & Updates	No	✓
Support SLA & Ticketing System	No	✓
Advisory & Support	No	✓
Security & Bug fixes	OSS Community	✓
Implementation Guidance	No	✓
Guaranteed SLA	No	✓

Learn More

<https://ionicframework.com/docs/enterprise>

# **Comparing to React Native's Plugins**

# React Native Libraries

- React Native includes a few libraries for accessing device resources
- Examples:
  - CameraRoll
  - AsyncStorage (device storage)
  - Geolocation (GPS)
- The rest are installed through plugins which look similar to Ionic's

<https://facebook.github.io/react-native/docs/native-modules-ios>

# React Native Plugins

- Used and installed in roughly the same way: install and link
  - `npm install react-native-sensors`
  - `react-native link react-native-sensors`
  - `npm install react-native-notification`
- (notifications require manual linking in Xcode or Android build)
- Notifications plugin only has 18 open issues,  
better maintained than Cordova's

<https://github.com/react-native-sensors/react-native-sensors>

<https://github.com/wix/react-native-notifications>

# Question



**When might developing a hybrid app be preferable to a native app?**

- A When you need to access a lot of native device resources
- B When you don't need any native device resources
- C When you can use well-maintained libraries to access native resources
- D When you need high performance
- E I don't know, this lecture has scared me away from making hybrid apps

# Thoughts on native resources

- The state of native support is pretty poor
- You could fork (copy) a broken or incomplete plugin and patch it yourself
- Is this better or worse than having to write the plugin yourself?
- This is the clear downside to building hybrid apps rather than native
  - Device libraries can't be used directly
  - Either need to rely on community libraries or fill in the missing pieces

# Strengths of hybrid apps

- Can share a codebase between web and mobile
- Can save time and effort (sometimes)
- Easily design for various form factors
- Access to some device capabilities

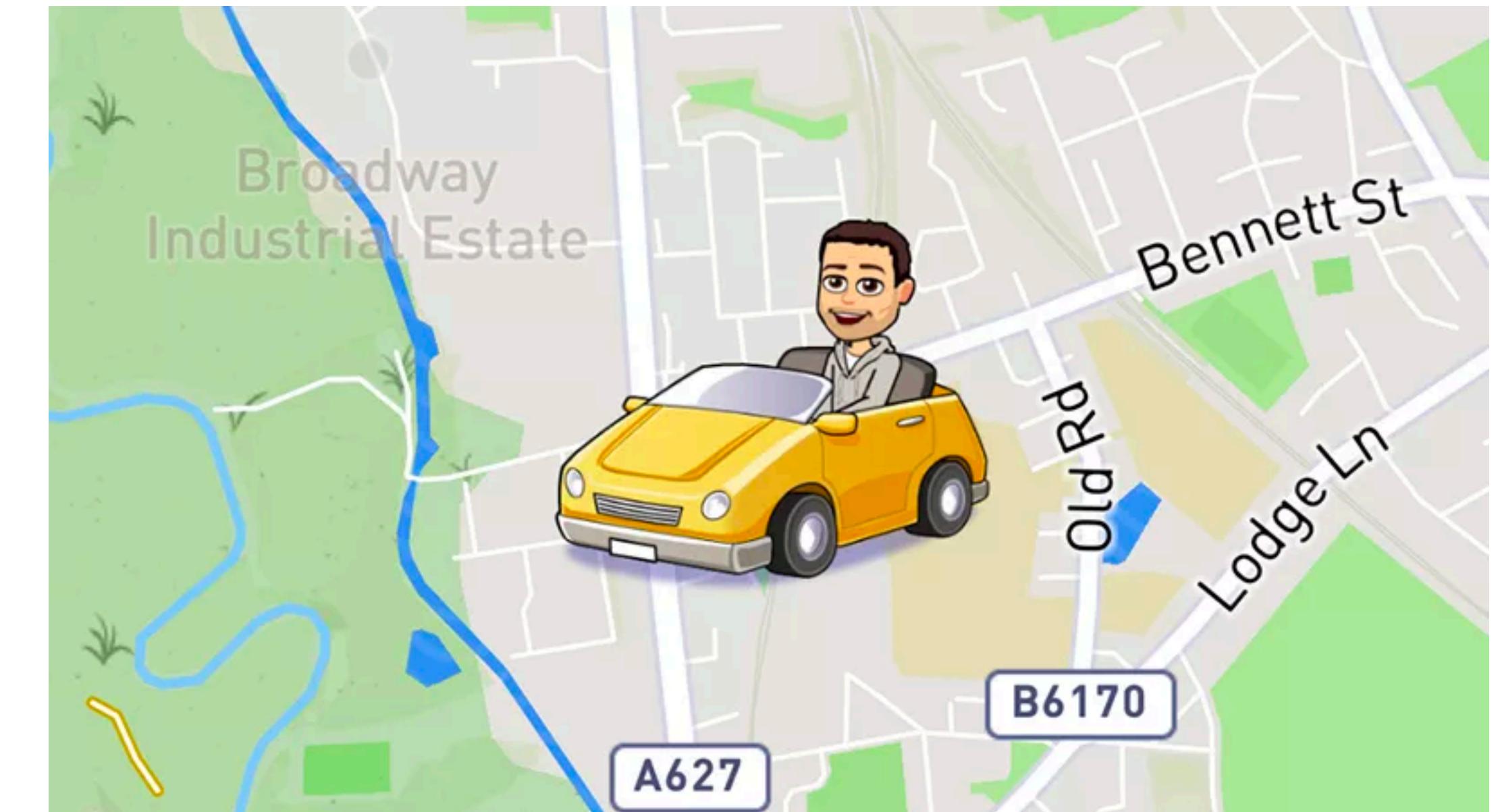
# Weaknesses of hybrid apps

- Performance issues
- Inconsistency with platform
- Limited access to device capabilities

**Modern phones include a lot of sensors**

# Sensors

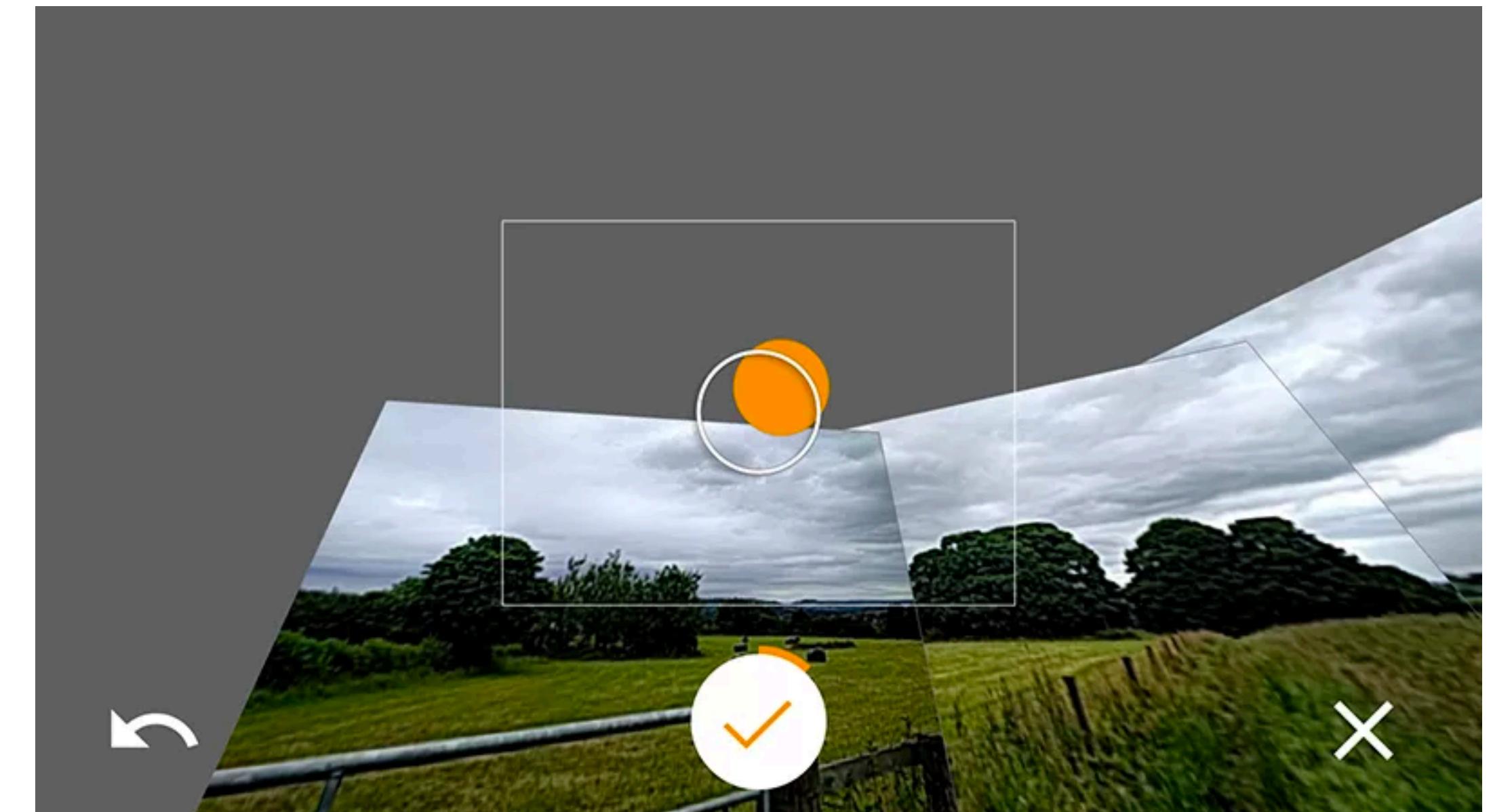
- Accelerometers
  - Axis-based motion sensing
  - Measures acceleration in a particular direction



<https://gizmodo.com/all-the-sensors-in-your-smartphone-and-how-they-work-1797121002>

# Sensors

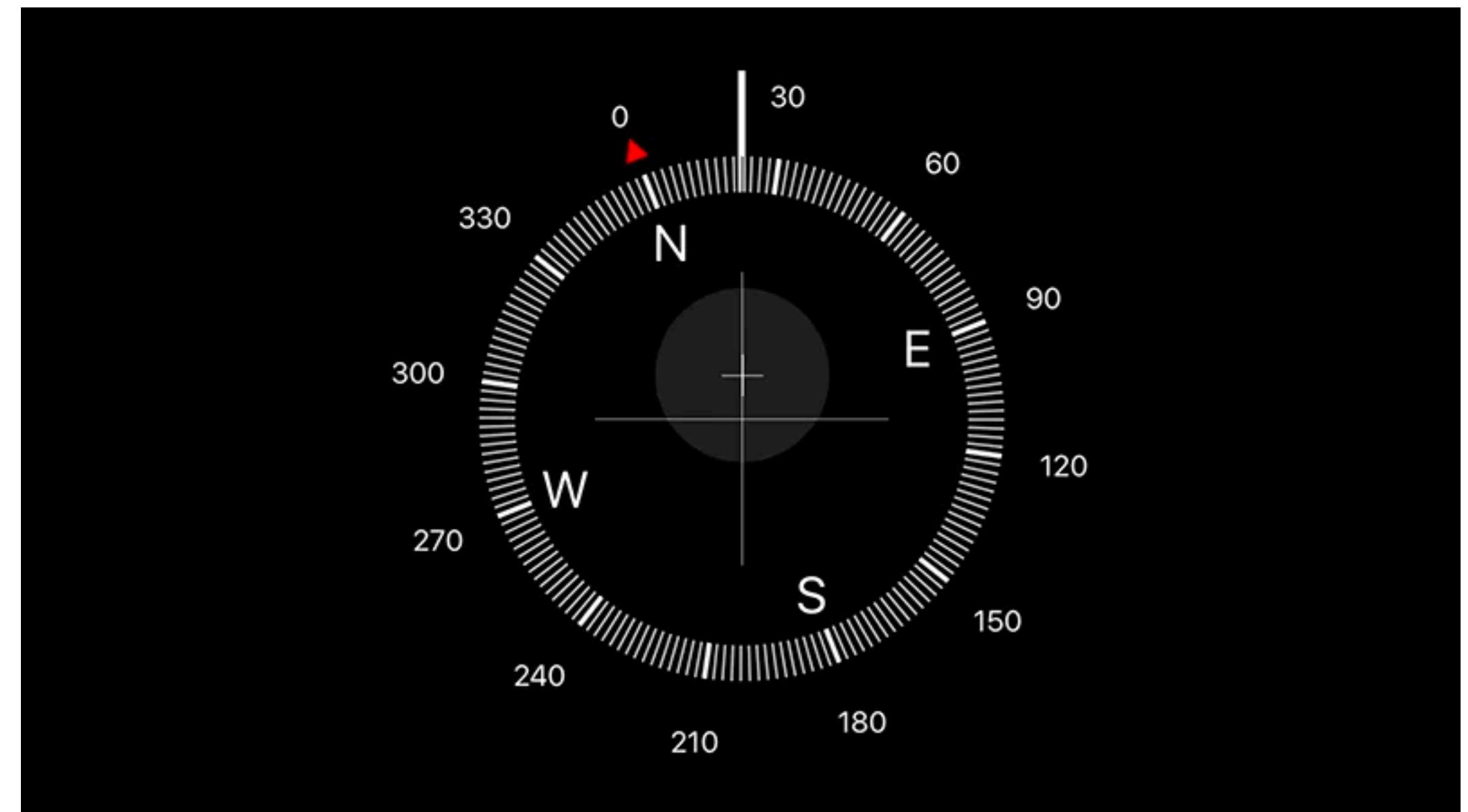
- Gyroscope
  - Measures device orientation
  - Can measure device rotation, where accelerometer cannot



<https://gizmodo.com/all-the-sensors-in-your-smartphone-and-how-they-work-1797121002>

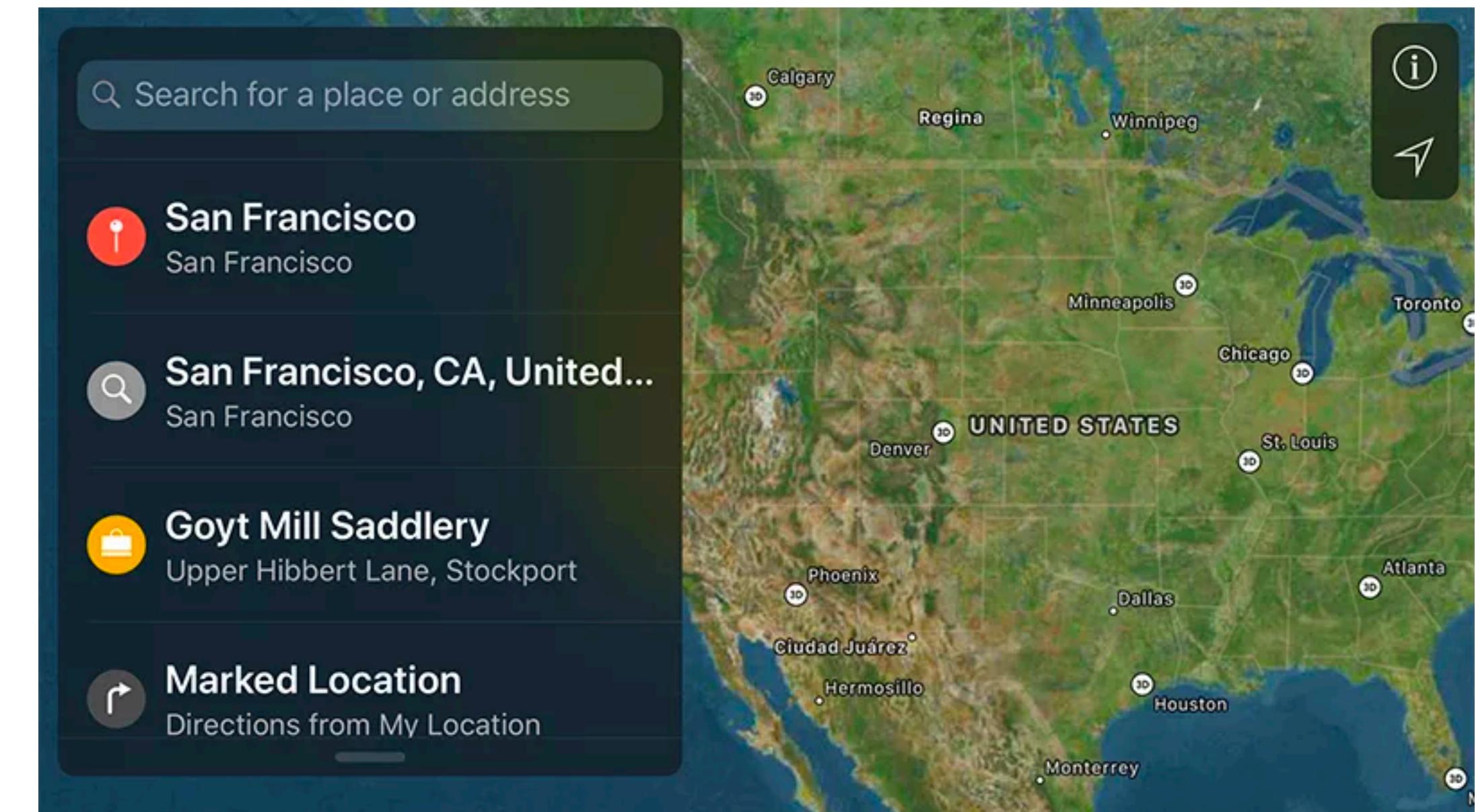
# Sensors

- Magnometer
  - Identifies cardinal direction
  - Can be used together with Gyroscope to create a compass



# Sensors

- Global Positioning System (GPS)
  - Identify where on the planet you are
  - Navigation in Apple Maps, Google Maps, etc. combines all four sensors



# Sensors

- Proximity sensor: how close/far an object is
  - Switches off your screen when it's in your pocket/backpack/purse
- Ambient light: measures how bright a room is
  - Changes screen brightness to accommodate
- Near field communication (NFC): allows nearby objects to communicate
  - Powers Apple Pay, etc.

# Sensors

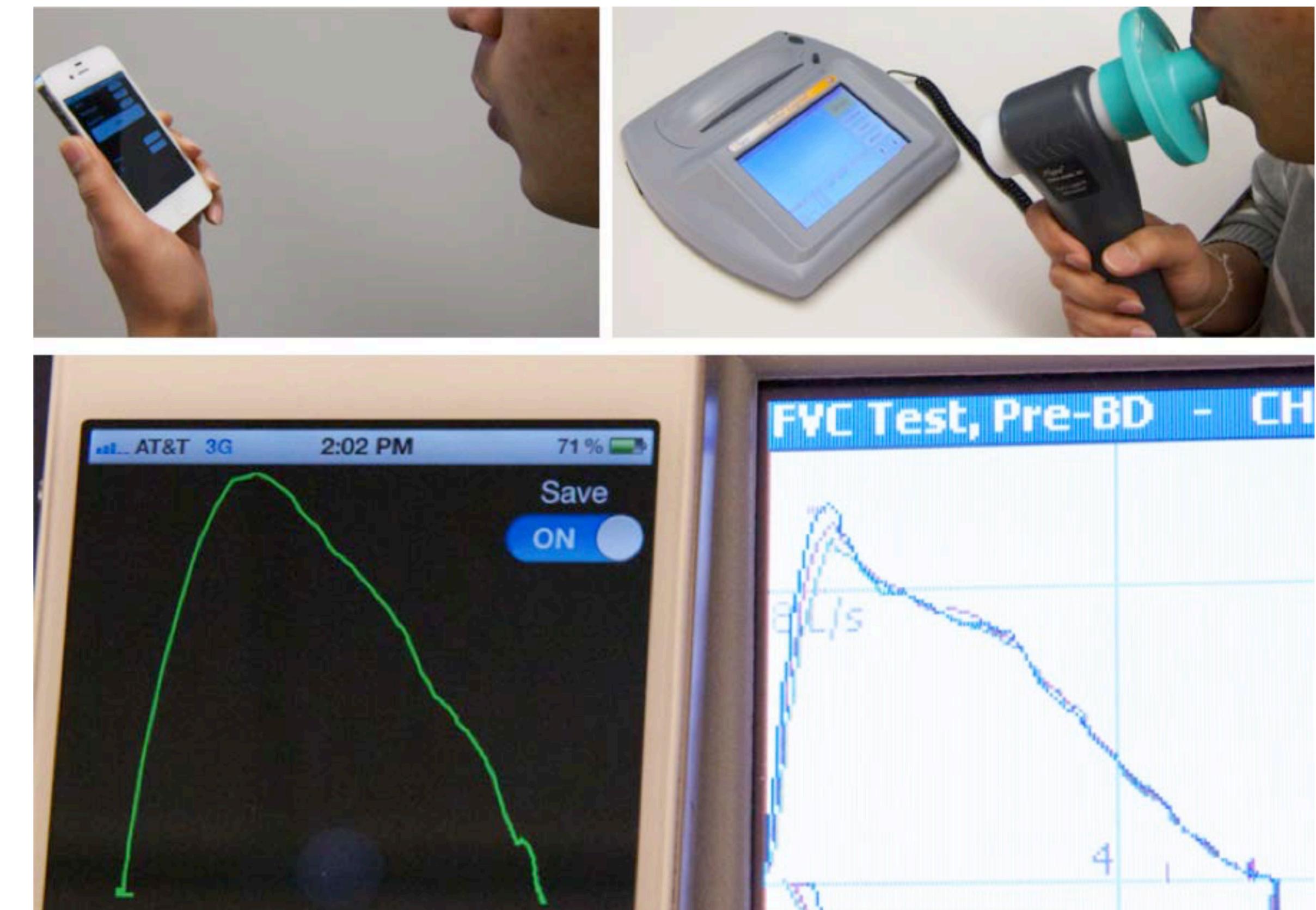
- Sensors can also be re-appropriated
  - Microphone: measure noise, such as for sleep quality
  - Camera: barcode or QR scanner
  - Accelerometer: pedometer
  - Touchscreen: pressure

# **Appropriating Sensors in Research**

# Appropriating Sensors in Research

## SpiroSmart

- Lung function (asthma/blockage) via a microphone



<https://dl.acm.org/citation.cfm?id=2370261>

Eric C. Larson, Mayank Goel, Gaetano Borriello, Sonya Heltshe, Margaret Rosenfeld, Shwetak N. Patel.  
SpiroSmart: Using a Microphone to Measure Lung Function on a Mobile Phone. UbiComp 2012

# Appropriating Sensors in Research

## BiliCam

- Jaundice in newborns via camera and a calibration card



<https://dl.acm.org/citation.cfm?id=2632076>

Lilian de Greef, Mayank Goel, Min Joon Seo, Eric C. Larson, James W. Stout, James A. Taylor, Shwetak N. Patel.  
BiliCam: Using Mobile Phones to Monitor Newborn Jaundice. UbiComp 2014

# Appropriating Sensors in Research

## Why?

- Medical devices are expensive and inaccessible
- Phones are widely available
  - ~40% of the world owns a smartphone today
  - Can enable these tests in lower-resource countries or counties
  - Can enable at-home tests and continuous monitoring
- Regulation is a separate and important issue

# Today's goals

By the end of today, you should be able to...

- Use Ionic's devApp to test an app on a mobile device
- Access device resources using an Ionic Native Plugin
- Describe some of the sensors on modern smartphones
- Describe some ways in which sensors can be used

# **IN4MATX 133: User Interface Software**

**Lecture 18:**  
**Device Resources & Sensors**

Professor Daniel A. Epstein  
TA Lucas de Melo Silva  
TA Jong Ho Lee