CS47SI: Cross-Platform Mobile Development

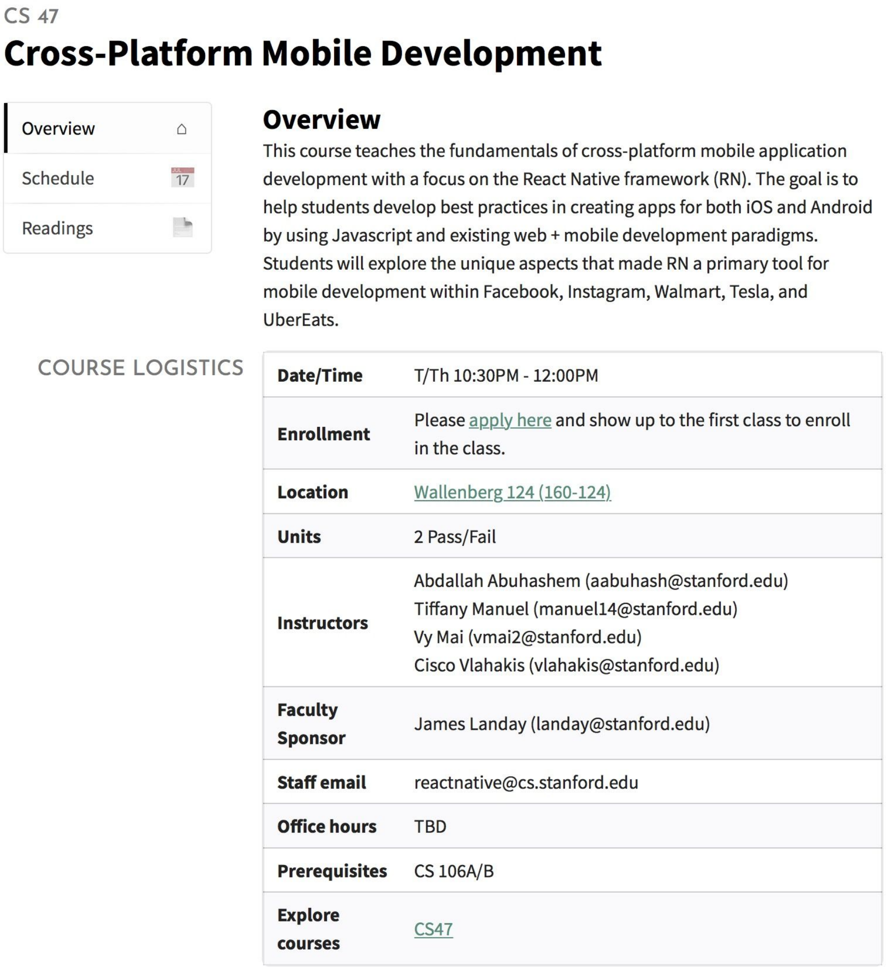
Lecture 4A: Third-Party Components & Expo Libraries

James Landay
Abdallah AbuHashem
Tiffany Manuel
Cisco Vlahakis
Vy Mai

https://cs47.stanford.edu

cs47-fall19.slack.com

Fall 2019



CS 47

Cross-Platform Mobile Development

- Overview ⌂
- Schedule
- Readings

Overview

This course teaches the fundamentals of cross-platform mobile application development with a focus on the React Native framework (RN). The goal is to help students develop best practices in creating apps for both iOS and Android by using Javascript and existing web + mobile development paradigms. Students will explore the unique aspects that made RN a primary tool for mobile development within Facebook, Instagram, Walmart, Tesla, and UberEats.

COURSE LOGISTICS

Date/Time	T/Th 10:30PM - 12:00PM
Enrollment	Please apply here and show up to the first class to enroll in the class.
Location	Wallenberg 124 (160-124)
Units	2 Pass/Fail
Instructors	Abdallah Abuhashem (aabuhash@stanford.edu) Tiffany Manuel (manuel14@stanford.edu) Vy Mai (vmai2@stanford.edu) Cisco Vlahakis (vlahakis@stanford.edu)
Faculty Sponsor	James Landay (landay@stanford.edu)
Staff email	reactnative@cs.stanford.edu
Office hours	TBD
Prerequisites	CS 106A/B
Explore courses	CS47

https://cs47.stanford.edu

To access lectures use Stanford email

Live Exercise

To-Do List

STARTER CODE

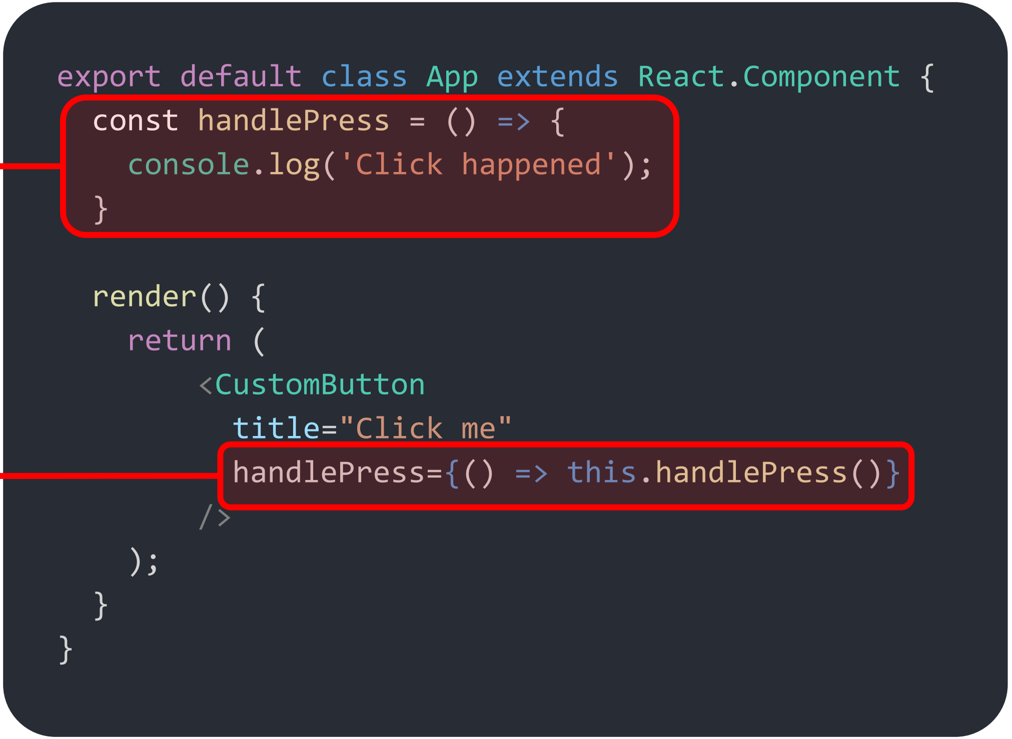
1) Run `npm install`
2) Open with Expo

Passing Function As Props

Fat Arrows Only

1 Write the function you want to pass in as a prop.

2 Pass the function as a prop using a fat arrow function.



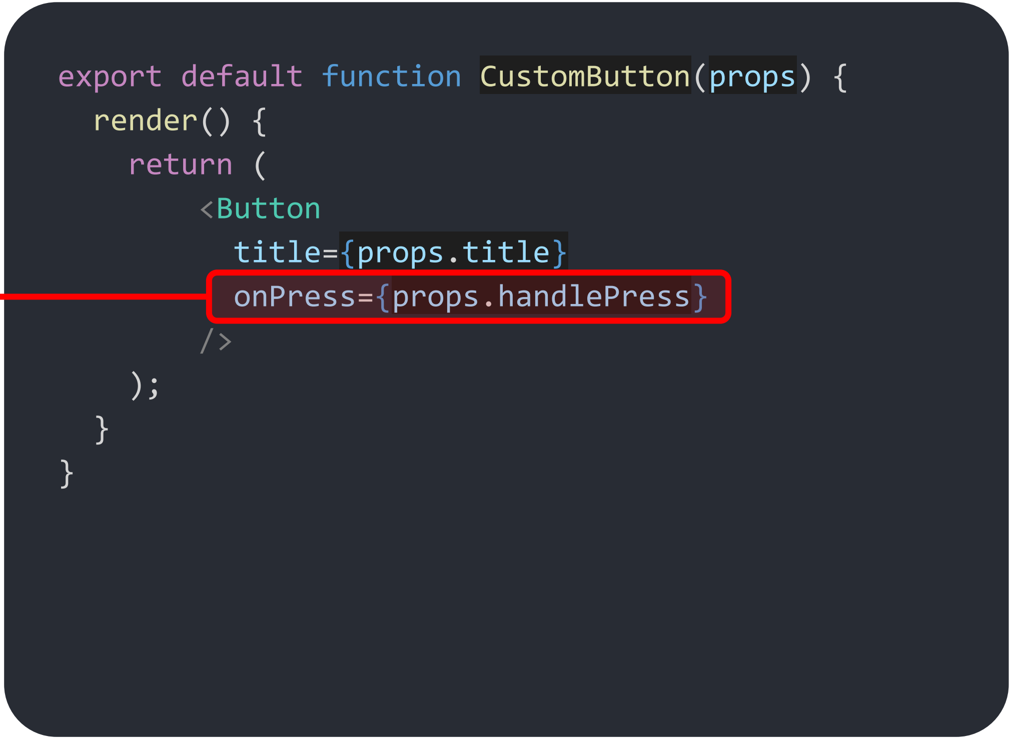
```
export default class App extends React.Component {
  const handlePress = () => {
    console.log('Click happened');
  }

  render() {
    return (
        <CustomButton
          title="Click me"
          handlePress={() => this.handlePress()}
        />
    );
  }
}
```

Passing Function As Props

Fat Arrows Only

Call the function in the child component



```
export default function CustomButton(props) {
  render() {
    return (
        <Button
          title={props.title}
          onPress={props.handlePress}
        />
    );
  }
}
```

MORE IN-DEPTH
Snack Example

Follow along code

https://reflect.sh/sole-sister

Live Exercise

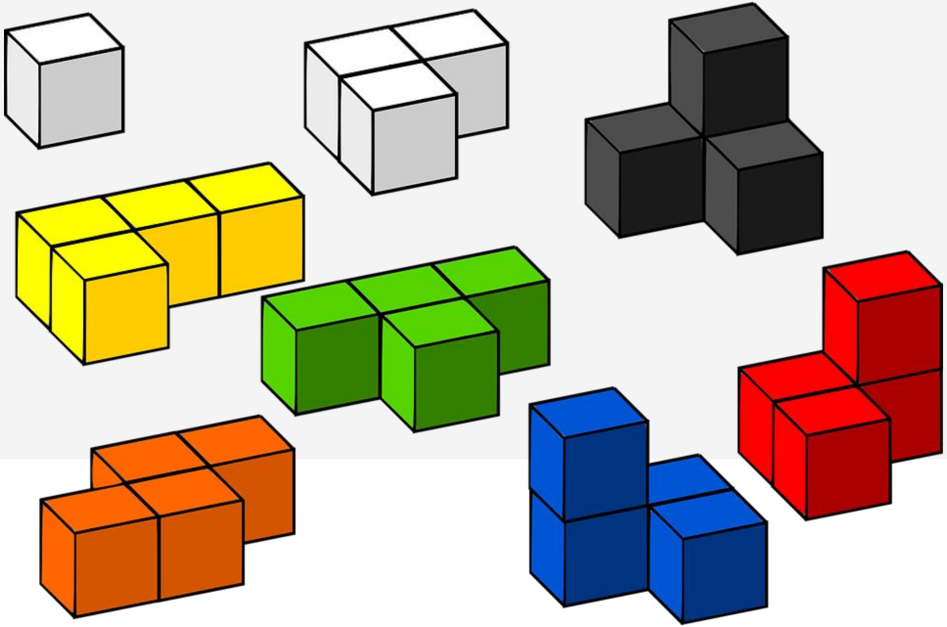
To-Do List

Complete part 1

Components

Building blocks

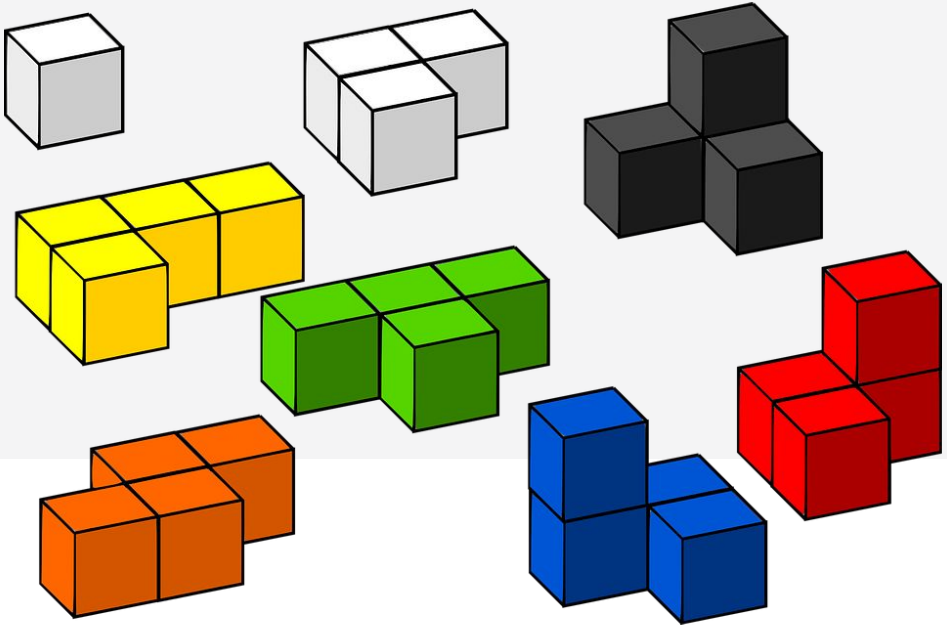
- React Native Components
- Custom Components
- Third-Party Components



Components

Building blocks

- React Native Components
- Custom Components
- Third-Party Components



Third-Party Components

WHY?

- Don’t reinvent the wheel
- Same time and energy

Third-Party Components

HOW?

1. `cd` into project directory

2. Run `npm install <third-party library> --save`
   NOTE: `--save` *is super important because it saves the library in package.json.*

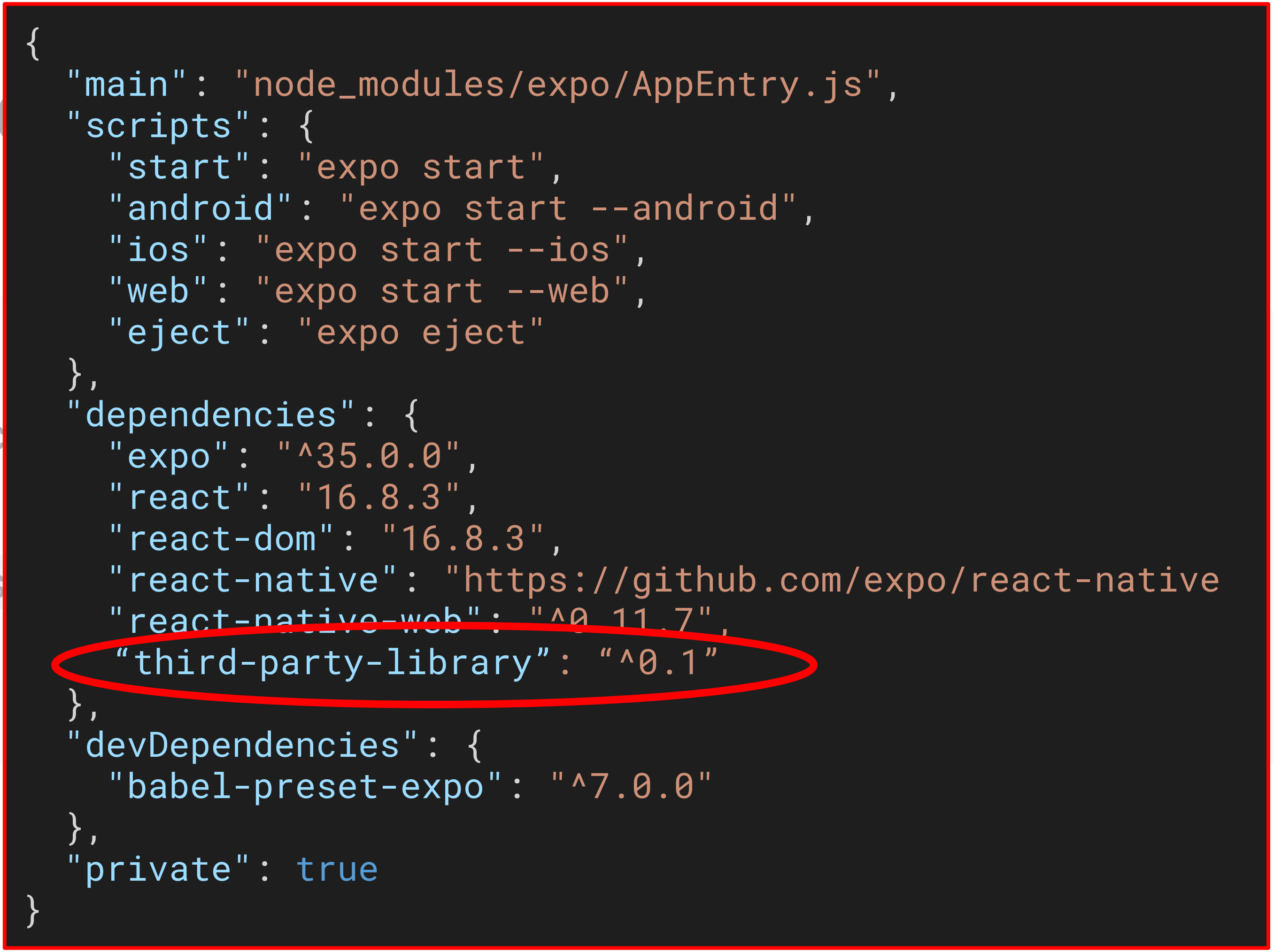
Third-Party (

HOW?

1 cd into projec

2 Run npm ins

NOTE: --save *is*



```
{
  "main": "node_modules/expo/AppEntry.js",
  "scripts": {
    "start": "expo start",
    "android": "expo start --android",
    "ios": "expo start --ios",
    "web": "expo start --web",
    "eject": "expo eject"
  },
  "dependencies": {
    "expo": "^35.0.0",
    "react": "16.8.3",
    "react-dom": "16.8.3",
    "react-native": "https://github.com/expo/react-native
    "react-native-web": "^0.11.7",
     “third-party-library”: “^0.1”
  },
  "devDependencies": {
    "babel-preset-expo": "^7.0.0"
  },
  "private": true
}
```

package.json

Third-Party Libraries

- React Native Elements
- NativeBase
- Expo Components

*See **API Reference** on sidebar for list of components.*

Follow new code

https://reflect.sh/sole-sister

Live Exercise

To-Do List

Complete part 2

Native Base CheckBox

Expo Libraries

- LinearGradient
- Font
- Camera
- ImagePicker
- WebView
- Constants
- etc

NOTE: *Some components do not need to be* `expo install`*-ed and come with Expo. Some do need to be* `expo install`*-ed. Read the documentation for each specific component to understand which cases require extra installation.*

Expo Vector Icons

- Expo Documentation
- Searchable Directory of Vector Icons

Follow along code

https://reflect.sh/sole-sister

Live Exercise

To-Do List

Complete part 3

Vector Icons

Third-Party Components

You can also download singular components from the npm registry:

- react-native-swipe-gestures
- react-native-modal
- react-native-animatable

NOTE: *Some components require linking, which then requires ejecting from Expo. For now, it is much simpler to avoid using such components.*

Follow along code

https://reflect.sh/sole-sister

Live Exercise

To-Do List

Complete part 4

react-native-swipe-gesture

Live Exercise

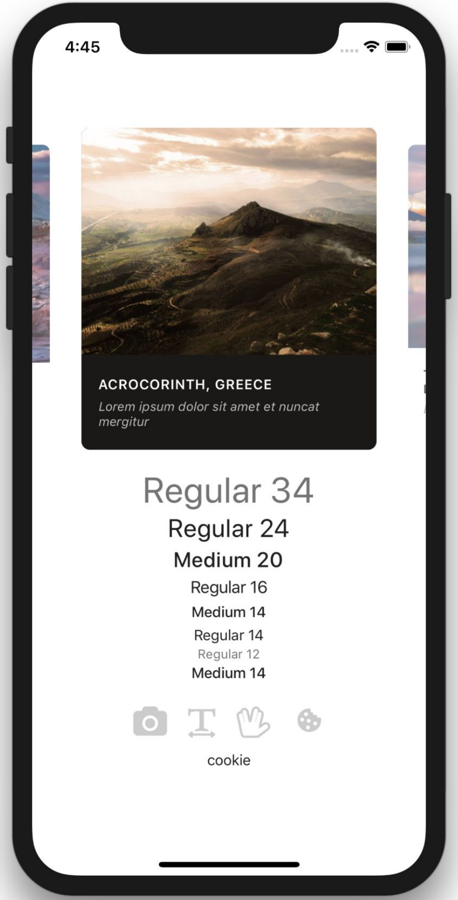
To-Do List

Final Code

https://github.com/stanford-cs47/Lecture4a-Final

Follow along code

https://reflect.sh/sole-sister



4:45

ACROCORINTH, GREECE

Lorem ipsum dolor sit amet et nuncat mergitur

Regular 34

Regular 24

Medium 20

Regular 16

Medium 14

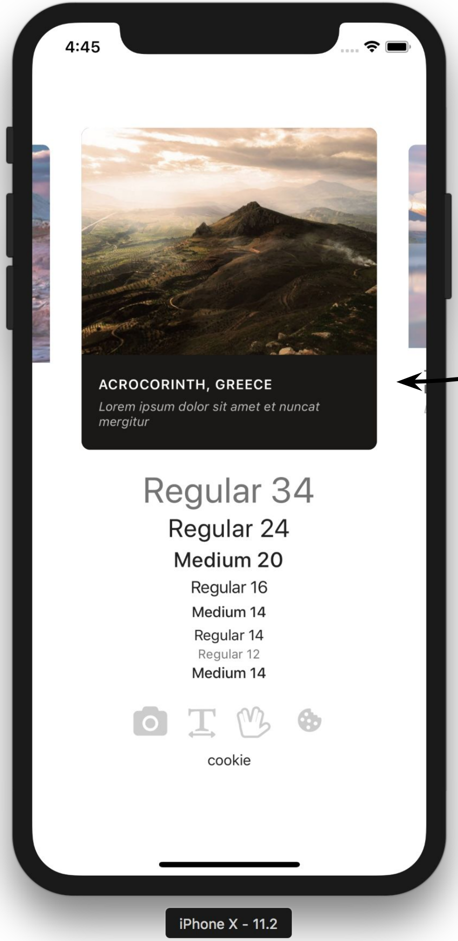
Regular 14

Regular 12

Medium 14

cookie

iPhone X - 11.2



ACROCORINTH, GREECE

Lorem ipsum dolor sit amet et nuncat mergitur

Regular 34

Regular 24

Medium 20

Regular 16

Medium 14

Regular 14

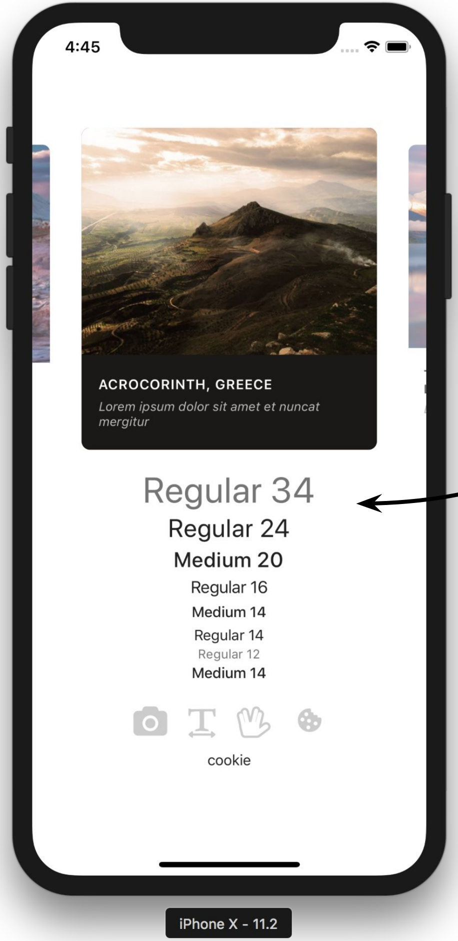
Regular 12

Medium 14

cookie

Snap Carousel

https://github.com/archriss/react-native-snap-carousel



4:45

ACROCORINTH, GREECE

Lorem ipsum dolor sit amet et nuncat mergitur

Regular 34

Regular 24

Medium 20

Regular 16

Medium 14

Regular 14

Regular 12

Medium 14

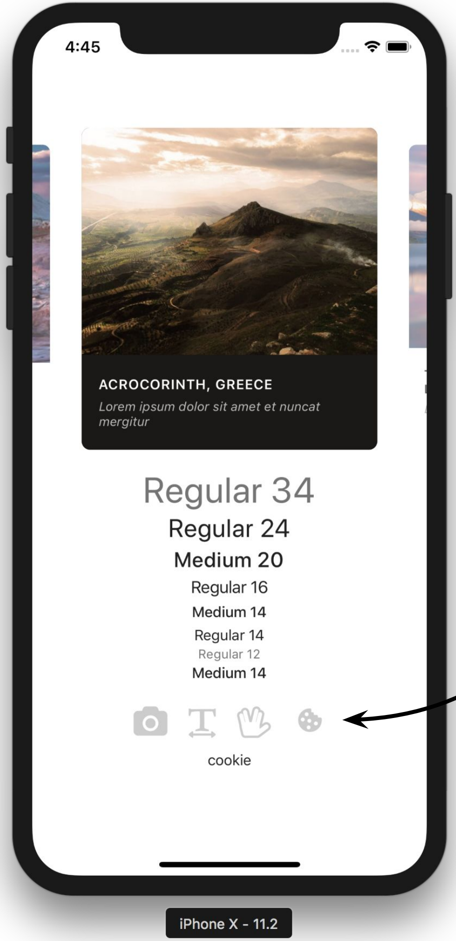
cookie

iPhone X - 11.2

Snap Carousel

Typography

https://github.com/hectahertz/react-native-typography



4:45

ACROCORINTH, GREECE

Lorem ipsum dolor sit amet et nuncat mergitur

Regular 34

Regular 24

Medium 20

Regular 16

Medium 14

Regular 14

Regular 12

Medium 14

cookie

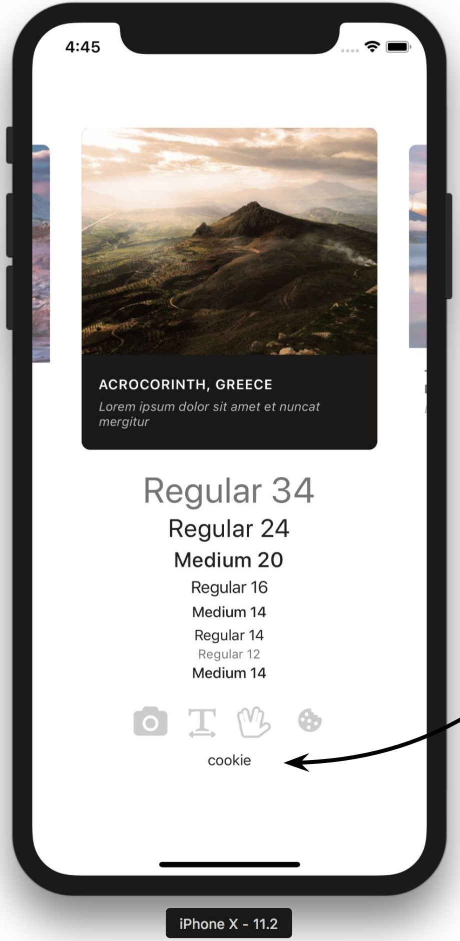
iPhone X - 11.2

Snap Carousel

Typography

Vector Icons

https://github.com/oblador/react-native-vector-icons
https://expo.github.io/vector-icons/



ACROCORINTH, GREECE

Lorem ipsum dolor sit amet et nuncat mergitur

Regular 34

Regular 24

Medium 20

Regular 16

Medium 14

Regular 14

Regular 12

Medium 14

cookie

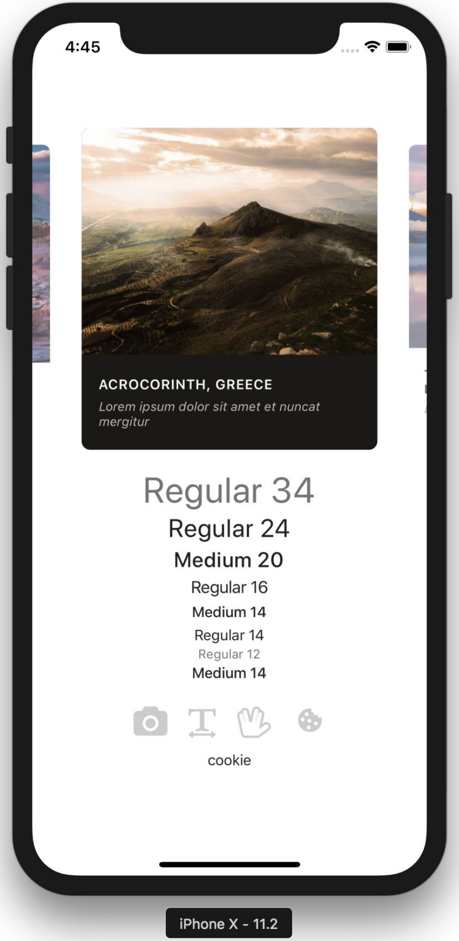
Snap Carousel

Typography

Vector Icons

Animatable

https://github.com/oblador/react-native-animatable



ACROCORINTH, GREECE

Lorem ipsum dolor sit amet et nuncat mergitur

Regular 34

Regular 24

Medium 20

Regular 16

Medium 14

Regular 14

Regular 12

Medium 14

cookie

Snap Carousel

Typography

Vector Icons

Animatable

Exercise

Extend your ToDo app with any third-party component.

SOME IDEAS:

- Add a clickable 'X' vector icon to make each ToDo deleteable (again)
- Use a third-party button element to clear the entire list of items (Native Base or React Native Elements)
- Make a toast appear when you add and/or delete an item (Native Base)
- Google a functionality/UI you want to have and try to use a third-party component that pops up

Office Hours

Email us directly if you’re not available at these times

Abdallah AbuHashem
Monday (12-1 PM) @ Huang Basement
Or by appointment

Vy Mai
Tuesday (3-4 PM) @ Old Union
Or by appointment

Cisco Vlahakis
Wednesday (8-9 PM) @ Huang Basement
Or by appointment

Tiffany Manuel
Thursday (2-3 PM) @ Huang Basement
Or by appointment

For today’s attendance, please see #general channel in our Slack.

Invitation:
https://tinyurl.com/cs47slack2019

CS47SI: Cross-Platform Mobile Development

Lecture 4A: Third-Party Components & Expo Libraries

James Landay
Abdallah AbuHashem
Tiffany Manuel
Cisco Vlahakis
Vy Mai

https://cs47.stanford.edu

cs47-fall19.slack.com

Fall 2019