

CS47: Cross-Platform Mobile Development

Lecture 7A: Nested Navigation (+ navigation exercise)

James Landay
Abdallah AbuHashem
Tiffany Manuel
Cisco Vlahakis
Vy Mai

<https://cs47.stanford.edu>

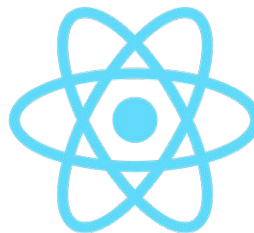


cs47-fall19.slack.com

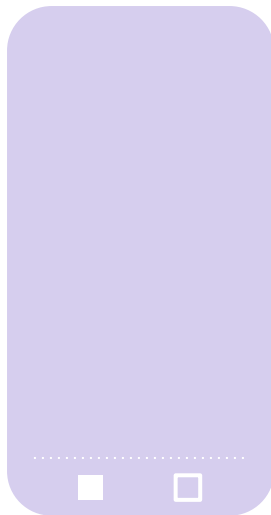
Fall 2019

Overview for today

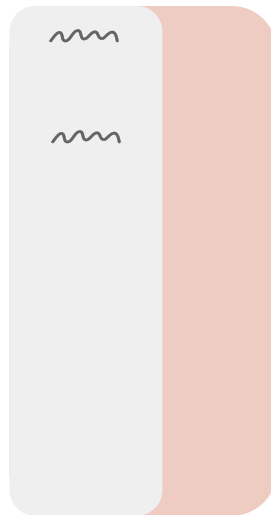
- [Authentication Flow Overview](#)
- Nested Navigation
- We will be working on an in-class project!
 - Groups of 2, some of 3



Navigation

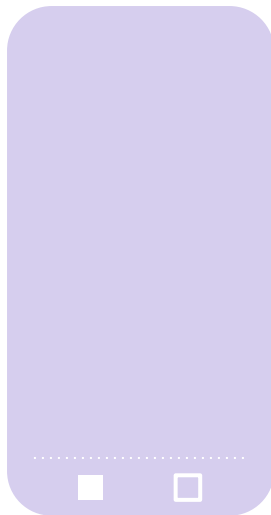


Tab
Navigator



Drawer
Navigator

Navigation

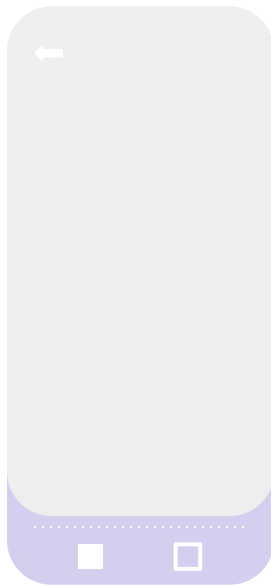


Tab
Navigator



Drawer
Navigator

Navigation

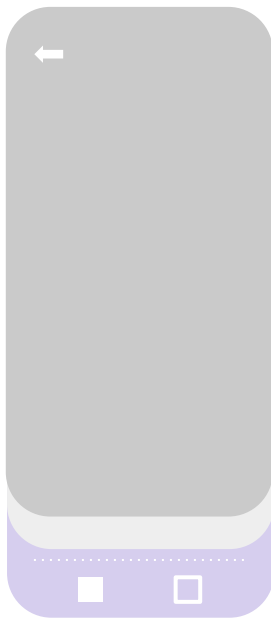


Tab
Navigator
+
Stack



Drawer
Navigator

Navigation

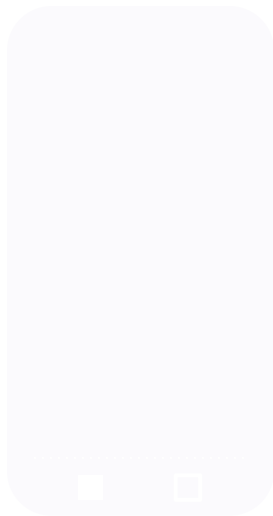


Tab
Navigator
+
Stack

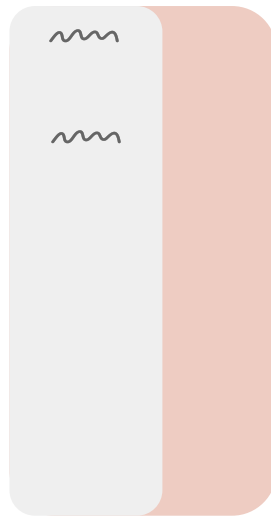


Drawer
Navigator

Navigation

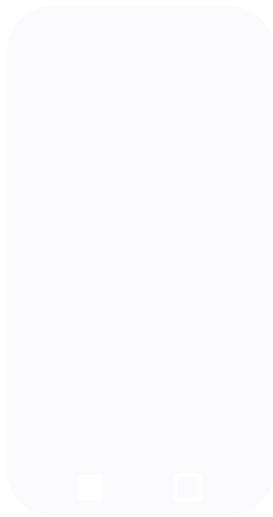


Tab
Navigator

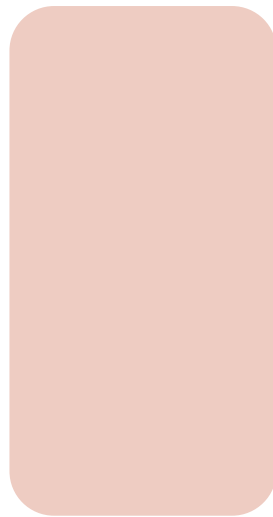


Drawer
Navigator

Navigation

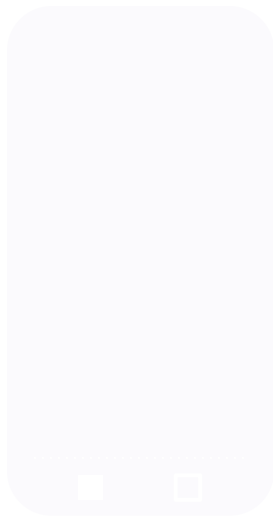


Tab
Navigator

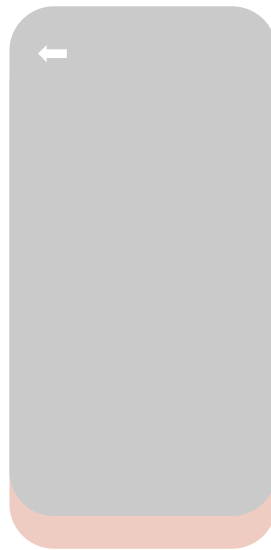


Drawer
Navigator

Navigation

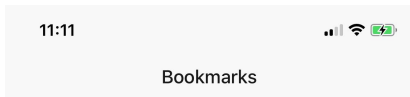


Tab
Navigator



Drawer
Navigator
+
Stack

What does this look like
programmatically?



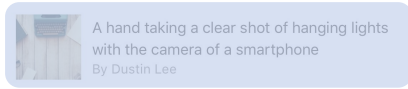
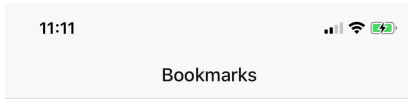
A hand taking a clear shot of hanging lights
with the camera of a smartphone
By Dustin Lee



No Description
By NASA

```
import HomeScreen from '../Screens/HomeScreen'
import BookmarkScreen from '../Screens/BookmarkScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkScreen },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})
```





No Description
By NASA



ಽ_(ツ)_/ಽ

```
import HomeScreen from '../Screens/HomeScreen'
import BookmarkScreen from '../Screens/BookmarkScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkScreen },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})
```



11:11



Bookmarks



A hand taking a clear shot of hanging lights
with the camera of a smartphone
By Dustin Lee



No Description
By NASA



A vintage analogue typewriter with a white
plain paper
By Florian Klauer

~_(\ツ)_/~



```
import HomeScreen from '../Screens/HomeScreen'
import BookmarkScreen from '../Screens/BookmarkScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkScreen },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})
```

```
import HomeScreen from '../Screens/HomeScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkNavigator },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})

import BookmarkScreen from '../Screens/BookmarkScreen'
const BookmarkNavigator = createStackNavigator({
  BookmarkScreen : {screen: BookmarkScreen},
  BookmarkViewerScreen: {screen: BookmarkViewerScreen},
}, {
  // Default config for all screens
  headerMode: 'float',
  initialRouteName: 'BookmarkScreen'
})

const AppContainer = createAppContainer(TabNav);
export default AppContainer
```

```
import HomeScreen from '../Screens/HomeScreen'
import BookmarkScreen from '../Screens/BookmarkScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkScreen },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})
```

```
import HomeScreen from '../Screens/HomeScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkNavigator },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})

import BookmarkScreen from '../Screens/BookmarkScreen'
const BookmarkNavigator = createStackNavigator({
  BookmarkScreen : {screen: BookmarkScreen},
}, {
  // Default config for all screens
  headerMode: 'float',
  initialRouteName: 'BookmarkScreen'
})
```

```
import HomeScreen from '../Screens/HomeScreen'
import BookmarkScreen from '../Screens/BookmarkScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkScreen },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})
```

```
import HomeScreen from '../Screens/HomeScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkNavigator },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})

import BookmarkScreen from '../Screens/BookmarkScreen'
const BookmarkNavigator = createStackNavigator({
  BookmarkScreen : {screen: BookmarkScreen},
  BookmarkViewerScreen: {screen: BookmarkViewerScreen},
}, {
  // Default config for all screens
  headerMode: 'float',
  initialRouteName: 'BookmarkScreen'
})
```



```
import HomeScreen from '../Screens/HomeScreen'
import BookmarkScreen from '../Screens/BookmarkScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkScreen },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})
```

```
import HomeScreen from '../Screens/HomeScreen'
//The main nav of my app
const TabNav = createBottomTabNavigator({
  FeedScreen: { screen: HomeScreen },
  BookmarkScreen: { screen: BookmarkNavigator },
}, {
  // Default config for all screens
  initialRouteName: 'FeedScreen',
  tabBarOptions: {
    activeTintColor: Colors.black,
  },
})

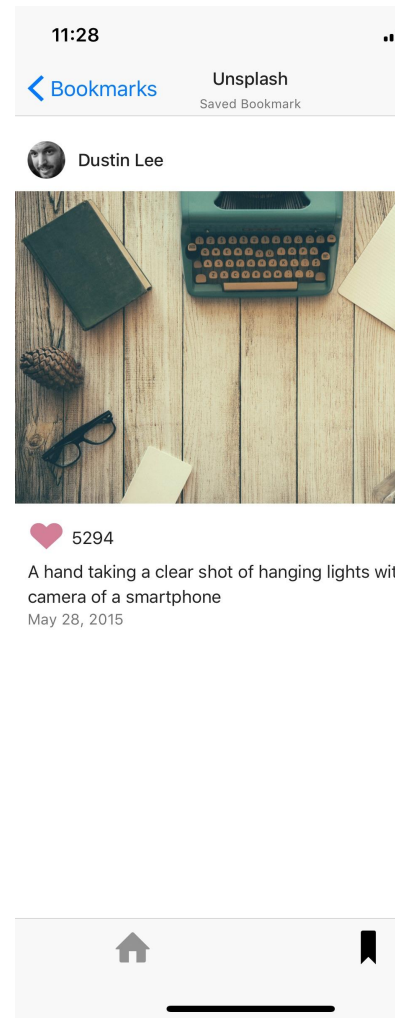
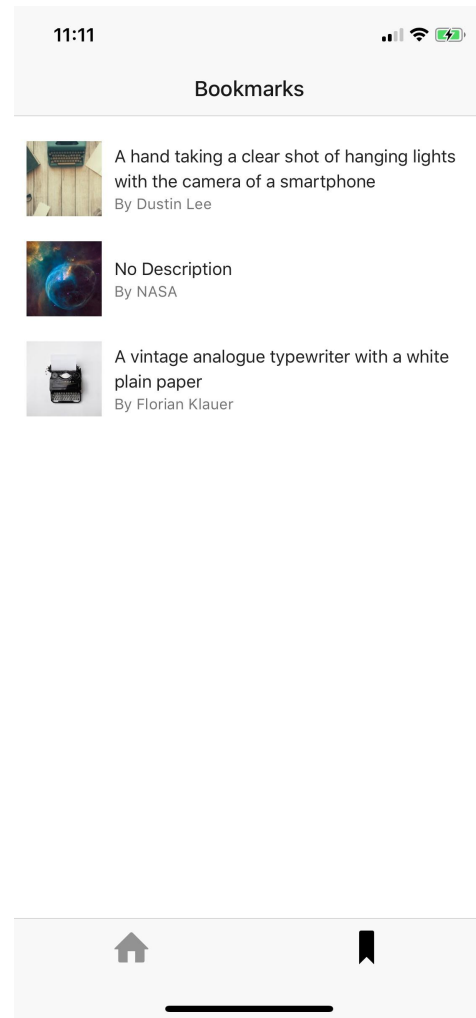
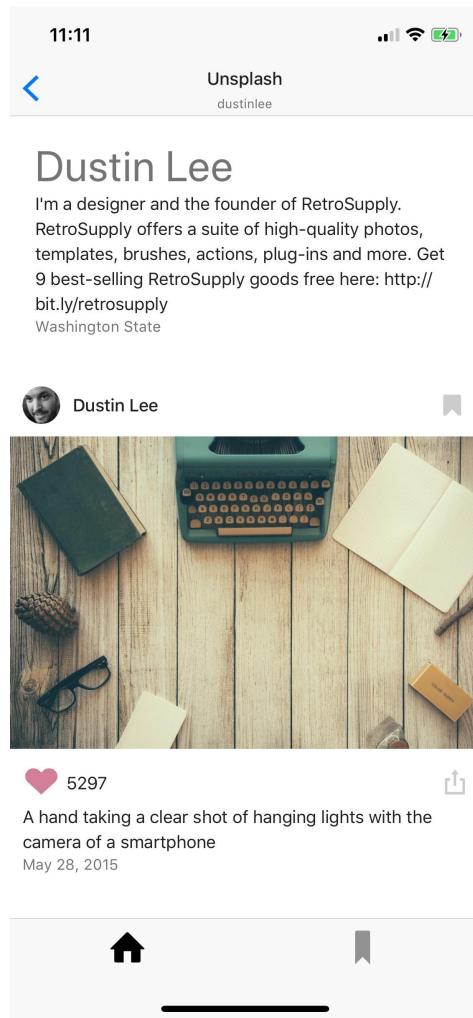
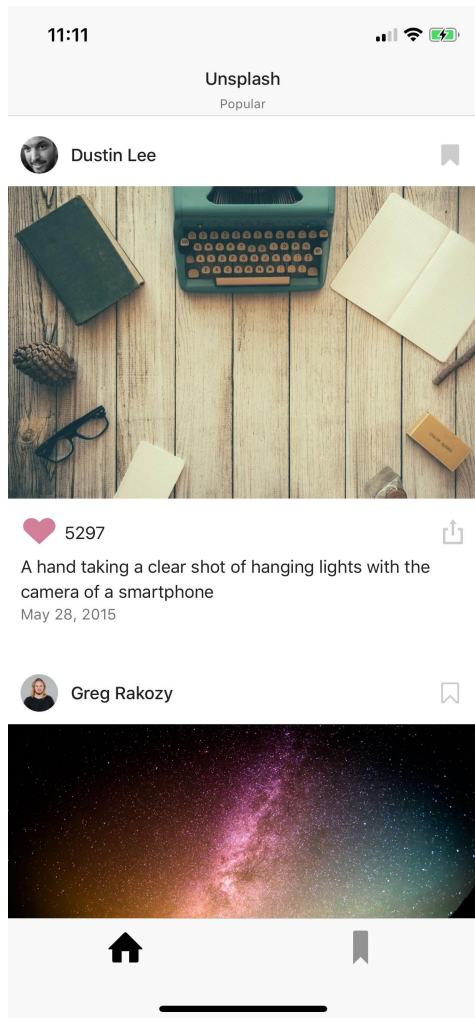
import BookmarkScreen from '../Screens/BookmarkScreen'
const BookmarkNavigator = createStackNavigator({
  BookmarkScreen : {screen: BookmarkScreen},
  BookmarkViewerScreen: {screen: BookmarkViewerScreen},
}, {
  // Default config for all screens
  headerMode: 'float',
  initialRouteName: 'BookmarkScreen'
})

const AppContainer = createAppContainer(TabNav);
export default AppContainer
```

Demo

<https://reflect.sh/thriving-hook>

Exercise



Roadmap

1. Implement a FlatList using server data
 - a. We want to be able to tap on a FlatList element and navigate somewhere else...
2. Implement Nested Navigation.
3. Implement the navigate() function.
4. If time allows, we'll then make things look prettier.
5. If time allows, we will implement bookmark functionality!

Groups of 2, or 3.

Follow these slides as guidance.

Don't move on to next step until everyone in the group
has finished the step.

Starter Code:

<https://github.com/stanford-cs47/Lecture7A-Starter>

Step 0.5: note that the App.js renders a TabNavigator. The main tab shown is HomeScreen.js

Step 1: start with the Feed.js file, which is what HomeScreen.js renders

- Comment in the “this.getFeedData()” method in componentDidMount
- Go to the getTabContent() function and follow the steps in the comments.
- Go to onProfilePressed and follow the comments.

Before moving on, make sure that everyone in your group has gotten the previous step.

Step 2: go to the AppNavigation.js file

- Follow all three steps
- Make sure that your code runs

Live code

<https://reflect.sh/overjoyed-basket>

Before moving on, check-in with your group.

Step 3: go back to the HomeScreen.js file

- Follow the steps on onProfileRequested()
 - This relies on your Feed.js implementation. Note how we are passing data back to the screen which is defined in the Navigator — only this screen has access to the navigate() function.
- You have now opened a new screen inside of your nested Stack :')

Starter Code: <https://github.com/stanford-cs47/Lecture7A-Starter>

If time allows

Step 4: go back to your Feed.js file

- Implement part X in `getTabContent()`. Use the `ListHeaderComponent` prop of a `FlatList`. Pass on to it something to render. The `Feed.js` has a prop with a view that you should render here.

The `BookmarkScreen.js` and `BookmarkViewerScreen.js` is already coded. Let's hookup some bookmarks!

Before moving on, check-in with your group.

Step 5: go to the `FeedItem.js` file

- Implement the `_storeBookmark()` and `_removeBookmark()` functions using `AsyncStorage`. Your `BookmarkScreen.js` file will pick up these changes once you refresh.

CS47: Cross-Platform Mobile Development

Lecture 7A: Nested Navigation (+ navigation exercise)

James Landay
Abdallah AbuHashem
Tiffany Manuel
Cisco Vlahakis
Vy Mai

<https://cs47.stanford.edu>



cs47-fall19.slack.com

Fall 2019