ECS 165A Discussion

React-Native Setup

Setting up React-Native

- You'll need Xcode, node and npm on your system.
- To install react-native tool for CLI use the following:

  "npm install -g react-native-cli"

- After react-native is installed successfully you can set up a project by using the following command:

  "react-native init <appname>"
  Where you replace appname with the name of your app. This will set up a basic app in a folder with app name.
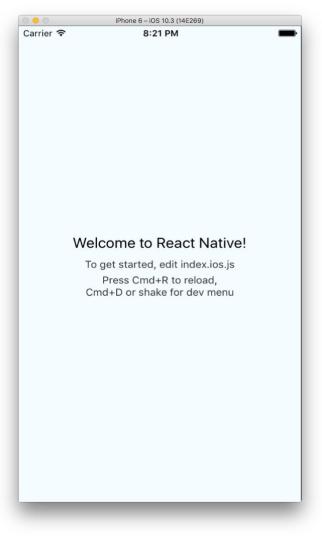
Running your App

- You can simulate your initial app using the following commands.

  "cd <appname>"
  "react-native run-ios" for ios or
  "react-native run-android" for android.

- The above command will take some time for the first time. It should open a terminal with Metro budler running.
- Did you run into some issues?
  Try clearing the cache and running the app using the following command:
  "react-native start --reset-cache"

What should you see?

- Something like the picture on the right.
- Click Hardware tab of simulator and click Shake Gesture and make sure you enable hot reloading.
- This helps to reflect the changes you make in app.js immediately in simulator.

Welcome to React Native!

To get started, edit index.ios.js

Press Cmd+R to reload,
Cmd+D or shake for dev menu

React Buttons

- Part of react-native.
- It is of the form <Button style="{color: red}" onPress="function()" title="name"/>
- onPress as name suggests executes the function when the button is pressed.
- You need pass a function to onPress
- Buttons play a crucial part of your App.

React Navigation

- We have different parts in your react application. How to seamlessly transition between them?
- Do it using react-navigation package.

How to Install it to your Project?

- run "npm install --save react-navigation"
- Then run "npm install --save react-native-gesture-handler" (This is a dependency)
- Finally "react-native link"

Using React Navigation

- Define different screens each for one part of your app.
- You will import 'createStackNavigator' from react-navigation and use as following:

```
"const RootStack = createStackNavigator({
           Home: HomeScreen,
           Login: LoginScreen,
     },{
           initialRouteName: 'Home',
     });
"
```

- InitialRouteName sets the default screen.

Using React Navigation (cont.)

- Wrap your StackNavigator in Appcontainer using createAppContainer method. This method should be imported from react-navigation.

  "const AppContainer = createAppContainer(RootStack);"

- Return this constant value(AppContainer) from your App class.

React Forms

- Forms are commonly used for login and registration.
- We can use react-native-form-builder package to make things easy for us.

How to Install?

- run "npm i react-native-clean-form --save"
- Finally "react-native link"

React Forms Usage

```
<Form>
 <FieldsContainer>
        <Fieldset label="User Registration">
           <FormGroup>
              <Label>Email</Label>
              <Input placeholder="esbenspetersen@gmail.com" />
           </FormGroup>
```

- The usage of Form is similar to the form element in HTML.

Snack Demo Link

- You can check the online version of Demo [here](https://snack.expo.io/@giritheja/ZnJvd2)(https://snack.expo.io/@giritheja/ZnJvd2).

Setting up SQLite

How to install SQLite to react-native?

- run "npm i react-native-clean-form --save"
- Finally "react-native link"

Usage:

- You have to import opendatabase from react-native-clean-form
- Execute the queries using:

```
db.transaction(function(txn) {
 txn.executeSql(
 query,                  //Query to execute as prepared statement
 argsToBePassed[],       //Argument to pass for the prepared statement
 function(tx, res) {}   //Callback function to handle the result response
 );
 });
```