Exercise (Instructions): Network Info

Objectives and Outcomes

In this exercise you will use the NetInfo API from React Native to access network information. At the end of this exercise you will be able to:

- Use the NetInfo API to obtain network information
- Respond to changes in the network information

Network Info

- Open MainComponent.js and update it as follows:

```
. . .

import { View, Platform, Text, ScrollView, Image, StyleSheet, NetInfo,
  ToastAndroid } from 'react-native';


. . .

class Main extends Component {

  componentDidMount() {
    this.props.fetchDishes();
    this.props.fetchComments();
    this.props.fetchPromos();
    this.props.fetchLeaders();

    NetInfo.getConnectionInfo()
        .then((connectionInfo) => {
            ToastAndroid.show('Initial Network Connectivity Type: '
                + connectionInfo.type + ', effectiveType: ' + connectionInfo
                  .effectiveType,
  }

  handleConnectivityChange = (connectionInfo) => {
    switch (connectionInfo.type) {
        ToastAndroid.show('You are now offline!', ToastAndroid.LONG);
        break;
      case 'wifi':
        ToastAndroid.show('You are now connected to WiFi!', ToastAndroid.LONG);
        break;
      case 'cellular':
        ToastAndroid.show('You are now connected to Cellular!', ToastAndroid
          .LONG);
        break;
      case 'unknown':
        ToastAndroid.show('You now have unknown connection!', ToastAndroid.LONG
          );
        break;
      default:
        break;
    }
  }

. . .
```

- Save the changes and do a Git commit with the message "Network Info".

Conclusions

In this exercise you use the NetInfo API to obtain network information and then respond to changes in network connectivity

Mark as completed