

```
1: import React, {Component} from 'react';
2: import {
3:   AppRegistry, StyleSheet, Text, View, ImagePickerIOS, Image, TouchableOpacity
4: } from 'react-native';
5:
6: export default class L10_CameraRollPicker extends Component {
7:   constructor(props) {
8:     super(props);
9:     this.state = {
10:       image: null
11:     };
12:     this.chooseImageFromGallery = this.chooseImageFromGallery.bind(this);
13:     this.chooseImageFromCamera = this.chooseImageFromCamera.bind(this);
14:   }
15:
16:   chooseImageFromGallery() {
17:     ImagePickerIOS.openSelectDialog({}, imageUri => {
18:       this.setState({image: imageUri});
19:     }, error => console.error(error));
20:   }
21:
22:   chooseImageFromCamera() {
23:     ImagePickerIOS.openCameraDialog({}, imageUri => {
24:       this.setState({image: imageUri});
25:     }, error => console.error(error));
26:   }
27:
28:   render() {
29:     return (
30:       <View style={{flex: 1}}>
31:         <View style={{flex: 1}}>
32:           {this.state.image?<Image style={{flex: 1}} source={{uri: this.state.image}}></Image>:null}
33:         </View>
34:         <View style={styles.container}>
35:           <TouchableOpacity style={styles.button} onPress={this.chooseImageFromGallery}>
36:             <Text style={styles.buttonText}>Open from Gallery</Text>
37:           </TouchableOpacity>
38:           <TouchableOpacity style={styles.button} onPress={this.chooseImageFromCamera}>
39:             <Text style={styles.buttonText}>Open from Camera</Text>
40:           </TouchableOpacity>
41:         </View>
42:       </View>
43:     );
44:   }
45: }
46:
47: const styles = StyleSheet.create({
48:   container:{
49:     flex: 1,
50:     flexDirection: 'row',
51:     alignItems: 'center',
52:     justifyContent: 'center'
53:   },
54:   button: {
55:     backgroundColor: 'gray',
56:     width: 150,
57:     height: 50,
58:     borderRadius: 10,
59:     alignItems: 'center',
60:     justifyContent: 'center',
61:     margin: 10
62:   },
63:   buttonText:{
64:     color: 'white'
65:   }
66: });
67:
68: AppRegistry.registerComponent('L10_CameraRollPicker', () => L10_CameraRollPicker);
```

```
1: var Platform = require('react-native').Platform;
2: var ImagePicker = require('react-native-image-picker');
3: import React, {Component} from 'react';
4: import {
5:   AppRegistry, StyleSheet, Text, View, Image, TouchableOpacity
6: } from 'react-native';
7:
8: export default class L10_CameraRollPicker extends Component {
9:   constructor(props) {
10:     super(props);
11:     this.state = {
12:       image: null
13:     };
14:     this.chooseImage = this.chooseImage.bind(this);
15:   }
16:
17:   chooseImage(){
18:     ImagePicker.showImagePicker({noData: true }, (response) => {
19:       console.log('Response = ', response);
20:
21:       if (response.didCancel) {
22:         console.log('User cancelled image picker');
23:       }
24:       else if (response.error) {
25:         console.log('ImagePicker Error: ', response.error);
26:       }
27:       else if (response.customButton) {
28:         console.log('User tapped custom button: ', response.customButton);
29:       } else {
30:         //If it is iOS, remove 'file://' prefix
31:         let source = {uri: response.uri.replace('file://', ''), isStatic: true};
32:
33:         //If android, don't need to remove the 'file://' prefix
34:         if (Platform.OS === 'android') {
35:           source = {uri: response.uri, isStatic: true};
36:         }
37:
38:         this.setState({image: source});
39:       }
40:     });
41:   }
42:
43:   render() {
44:     return (
45:       <View style={{flex: 1}}>
46:         <View style={{flex: 1}}>
47:           {this.state.image?<Image style={{flex: 1}} source={this.state.image}></Image>:null
48:         </View>
49:         <View style={styles.container}>
50:           <TouchableOpacity style={styles.button} onPress={this.chooseImage}>
51:             <Text style={styles.buttonText}>Choose Images...</Text>
52:           </TouchableOpacity>
53:         </View>
54:       </View>
55:     );
56:   }
57: }
58:
59: const styles = StyleSheet.create({
60:   container:{
61:     flex: 1,
62:     flexDirection: 'row',
63:     alignItems: 'center',
64:     justifyContent: 'center'
65:   },
66:   button: {
67:     backgroundColor: 'gray',
68:     width: 150,
69:     height: 50,
70:     borderRadius: 10,
71:     alignItems: 'center',
72:     justifyContent: 'center',
73:     margin: 10
74:   },
75:   buttonText:{
76:     color: 'white'
```

`77: }``78: });``79:``80: AppRegistry.registerComponent('L10_CameraRollPicker', () => L10_CameraRollPicker);`

```
1: var Platform = require('react-native').Platform;
2: var ImagePicker = require('react-native-image-picker');
3: import React, {Component} from 'react';
4: import {
5:   AppRegistry, StyleSheet, Text, View, Image, TouchableOpacity
6: } from 'react-native';
7:
8: export default class L10_CameraRollPicker extends Component {
9:   constructor(props) {
10:     super(props);
11:     this.state = {
12:       image: null
13:     };
14:     this.takePhoto = this.takePhoto.bind(this);
15:     this.chooseImage = this.chooseImage.bind(this);
16:     this.setImage = this.setImage.bind(this);
17:   }
18:
19:   takePhoto(){
20:     ImagePicker.launchCamera({noData: true }, this.setImage);
21:   }
22:
23:   chooseImage(){
24:     ImagePicker.launchImageLibrary({noData: true }, this.setImage);
25:   }
26:
27:   setImage(response){
28:     console.log('Response = ', response);
29:
30:     if (response.didCancel) {
31:       console.log('User cancelled image picker');
32:     }
33:     else if (response.error) {
34:       console.log('ImagePicker Error: ', response.error);
35:     }
36:     else if (response.customButton) {
37:       console.log('User tapped custom button: ', response.customButton);
38:     } else {
39:       //If it is iOS, remove 'file://' prefix
40:       let source = {uri: response.uri.replace('file://', ''), isStatic: true};
41:
42:       //If android, don't need to remove the 'file://' prefix
43:       if (Platform.OS === 'android') {
44:         source = {uri: response.uri, isStatic: true};
45:       }
46:
47:       this.setState({image: source});
48:     }
49:   }
50:
51:   render() {
52:     return (
53:       <View style={{flex: 1}}>
54:         <View style={{flex: 1}}>
55:           {this.state.image?<Image style={{flex: 1}} source={this.state.image}></Image>:null
56:         }
57:       </View>
58:       <View style={styles.container}>
59:         <TouchableOpacity style={styles.button} onPress={this.takePhoto}>
60:           <Text style={styles.buttonText}>Camera</Text>
61:         </TouchableOpacity>
62:         <TouchableOpacity style={styles.button} onPress={this.chooseImage}>
63:           <Text style={styles.buttonText}>Gallery</Text>
64:         </TouchableOpacity>
65:       </View>
66:     );
67:   }
68: }
69:
70: const styles = StyleSheet.create({
71:   container:{
72:     flex: 1,
73:     flexDirection: 'row',
74:     alignItems: 'center',
75:     justifyContent: 'center'
76:   },
```

```
77:   button: {
78:     backgroundColor: 'gray',
79:     width: 150,
80:     height: 50,
81:     borderRadius: 10,
82:     alignItems: 'center',
83:     justifyContent: 'center',
84:     margin: 10
85:   },
86:   buttonText:{
87:     color: 'white'
88:   }
89: });
90:
91: AppRegistry.registerComponent('L10_CameraRollPicker', () => L10_CameraRollPicker);
```

```
1: var Platform = require('react-native').Platform;
2: var ImagePicker = require('react-native-image-picker');
3: import React, {Component} from 'react';
4: import {
5:   AppRegistry, StyleSheet, Text, View, Image, TouchableOpacity, Modal, Slider
6: } from 'react-native';
7:
8: export default class L10_CameraRollPicker extends Component {
9:   constructor(props) {
10:     super(props);
11:     this.state = {
12:       star3:[],
13:       star2:[],
14:       star1:[],
15:       image: null,
16:       modalVisible:false,
17:       value:3
18:     };
19:     this.takePhoto = this.takePhoto.bind(this);
20:     this.chooseImage = this.chooseImage.bind(this);
21:     this.setImage = this.setImage.bind(this);
22:     this.putImageIntoList = this.putImageIntoList.bind(this);
23:   }
24:
25:   putImageIntoList(){
26:     this.setState({
27:       ['star'+this.state.value]:
28:       [...this.state['star'+this.state.value], this.state.image],
29:       modalVisible:false,
30:       value:3
31:     });
32:   }
33:
34:
35:   takePhoto(){
36:     ImagePicker.launchCamera({noData: true }, this.setImage);
37:   }
38:
39:   chooseImage(){
40:     ImagePicker.launchImageLibrary({noData: true }, this.setImage);
41:   }
42:
43:   setImage(response){
44:     console.log('Response = ', response);
45:
46:     if (response.didCancel) {
47:       console.log('User cancelled image picker');
48:     }
49:     else if (response.error) {
50:       console.log('ImagePicker Error: ', response.error);
51:     }
52:     else if (response.customButton) {
53:       console.log('User tapped custom button: ', response.customButton);
54:     } else {
55:       //If it is iOS, remove 'file://' prefix
56:       let source = {uri: response.uri.replace('file://', ''), isStatic: true};
57:
58:       //If android, don't need to remove the 'file://' prefix
59:       if (Platform.OS === 'android') {
60:         source = {uri: response.uri, isStatic: true};
61:       }
62:       this.setState({image: source, modalVisible: true});
63:     }
64:   }
65:
66:   render() {
67:     return (
68:       <View style={{flex: 1}}>
69:         <Modal
70:           animationType="slide"
71:           transparent={true}
72:           visible={this.state.modalVisible}
73:           onRequestClose={() => {alert("Modal has been closed.")}}
74:         >
75:         <View style={styles.modal}>
76:           <View>
77:             <TouchableOpacity style={styles.closeButton}
```

```
78:         onPress={() => this.setState({modalVisible: false})}}>
79:         <Text>X</Text>
80:     </TouchableOpacity>
81:     <Text>Please select coolness of this picture.</Text>
82:     <Text style={styles.headerText}>{this.state.value}</Text>
83:     <Slider maximumValue={3} minimumValue={1} step={1} value={3}
84:       onChange={(value) => this.setState({value: value})}></Slider>
85:     <TouchableOpacity style={styles.submitButton} onPress={() => {
86:       this.putImageIntoList();
87:     }}>
88:       <Text style={styles.buttonText}>OK</Text>
89:     </TouchableOpacity>
90:   </View>
91: </View>
92: </Modal>
93:
94:   <View style={{flex: 1, paddingTop:22}}>
95:     <View style={styles.title}><Text style={styles.titleText}>3 Stars</Text></View>
96:     <View style={styles.row}>
97:       {this.state.star3.map((source, i)=> <Image key={"star3-"+i}
98:         style={styles.image} source={source}></Image>)}
99:     </View>
100:    <View style={styles.title}><Text style={styles.titleText}>2 Stars</Text></View>
101:    <View style={styles.row}>
102:      {this.state.star2.map((source, i)=> <Image key={"star2-"+i}
103:        style={styles.image} source={source}></Image>)}
104:    </View>
105:    <View style={styles.title}><Text style={styles.titleText}>1 Star</Text></View>
106:    <View style={styles.row}>
107:      {this.state.star1.map((source, i)=> <Image key={"star1-"+i}
108:        style={styles.image} source={source}></Image>)}
109:    </View>
110:  </View>
111:  <View style={styles.rowCenter}>
112:    <TouchableOpacity style={styles.button} onPress={this.takePhoto}>
113:      <Text style={styles.buttonText}>Camera</Text>
114:    </TouchableOpacity>
115:    <TouchableOpacity style={styles.button} onPress={this.chooseImage}>
116:      <Text style={styles.buttonText}>Gallery</Text>
117:    </TouchableOpacity>
118:  </View>
119: </View>
120:   );
121: }
122: }
123:
124: const styles = StyleSheet.create({
125:   row:{
126:     flexDirection:'row',
127:     flexWrap: 'wrap'
128:   },
129:   rowCenter:{
130:     flexDirection: 'row',
131:     alignItems: 'center',
132:     justifyContent: 'center'
133:   },
134:   image:{
135:     width:100,
136:     height:100
137:   },
138:   button: {
139:     backgroundColor: 'gray',
140:     width: 150,
141:     height: 50,
142:     borderRadius: 10,
143:     alignItems: 'center',
144:     justifyContent: 'center',
145:     margin: 10
146:   },
147:   buttonText:{
148:     color: 'white'
149:   },
150:   modal:{
151:     height: 200,
152:     width: 300,
153:     marginTop: 200,
154:     padding: 10,
```

```
155:     alignSelf: 'center',
156:     backgroundColor: 'lightblue',
157:     margin: 10,
158:     borderRadius: 10,
159:     justifyContent: 'center',
160:     alignItems: 'center'
161:   },
162:   closeButton: {
163:     alignSelf: 'flex-end'
164:   },
165:   submitButton: {
166:     alignSelf: 'center',
167:     backgroundColor: 'darkblue',
168:     width: 100,
169:     height: 44,
170:     borderRadius: 10,
171:     alignItems: 'center',
172:     justifyContent: 'center',
173:     margin: 10
174:   },
175:   headerText: {
176:     fontSize: 20,
177:     alignSelf: 'center'
178:   },
179:   title: {
180:     backgroundColor: 'gray',
181:     padding: 5
182:   },
183:   titleText: {
184:     color: 'white'
185:   }
186: });
187:
188: AppRegistry.registerComponent('L10_CameraRollPicker', () => L10_CameraRollPicker);
```



```
1: var Platform = require('react-native').Platform;
2: var ImagePicker = require('react-native-image-picker');
3: import React, {Component} from 'react';
4: import {
5:   AppRegistry, StyleSheet, Text, View, Image, TouchableOpacity, Modal, Slider
6: } from 'react-native';
7:
8: export default class L10_CameraRollPicker extends Component {
9:   constructor(props) {
10:     super(props);
11:     this.state = {
12:       star3:[],
13:       star2:[],
14:       star1:[],
15:       image: null,
16:       modalVisible:false,
17:       value:3,
18:       fullScreenImage: null,
19:       fullScreenModalVisible: false
20:     };
21:     this.takePhoto = this.takePhoto.bind(this);
22:     this.chooseImage = this.chooseImage.bind(this);
23:     this.setImage = this.setImage.bind(this);
24:     this.putImageIntoList = this.putImageIntoList.bind(this);
25:     this.openFullScreenModal = this.openFullScreenModal.bind(this);
26:   }
27:
28:   putImageIntoList(){
29:     this.setState({
30:       ['star'+this.state.value]:
31:       [...this.state['star'+this.state.value], this.state.image],
32:       modalVisible:false,
33:       value:3
34:     });
35:   }
36:
37:
38:   takePhoto(){
39:     ImagePicker.launchCamera({noData: true }, this.setImage);
40:   }
41:
42:   chooseImage(){
43:     ImagePicker.launchImageLibrary({noData: true }, this.setImage);
44:   }
45:
46:   openFullScreenModal(source){
47:     this.setState({
48:       fullScreenImage: source,
49:       fullScreenModalVisible: true
50:     });
51:   }
52:
53:   setImage(response){
54:     console.log('Response = ', response);
55:
56:     if (response.didCancel) {
57:       console.log('User cancelled image picker');
58:     }
59:     else if (response.error) {
60:       console.log('ImagePicker Error: ', response.error);
61:     }
62:     else if (response.customButton) {
63:       console.log('User tapped custom button: ', response.customButton);
64:     } else {
65:       //If it is iOS, remove 'file://' prefix
66:       let source = {uri: response.uri.replace('file://', ''), isStatic: true};
67:
68:       //If android, don't need to remove the 'file://' prefix
69:       if (Platform.OS === 'android') {
70:         source = {uri: response.uri, isStatic: true};
71:       }
72:       this.setState({image: source, modalVisible: true});
73:     }
74:   }
75:
76:   render() {
77:     return (
```

```
78: <View style={{flex: 1}}>
79: <Modal
80:   animationType={"slide"}
81:   transparent={false}
82:   visible={this.state.fullScreenModalVisible}
83:   onRequestClose={() => {alert("Modal has been closed.")}}
84: >
85:   <TouchableOpacity style={{flex:1}}
86:     onPress={() => this.setState({fullScreenModalVisible: false})}>
87:     <Image
88:       style={{flex:1}} source={this.state.fullScreenImage}></Image>
89:     </TouchableOpacity>
90:   </Modal>
91:   <Modal
92:     animationType={"slide"}
93:     transparent={true}
94:     visible={this.state.modalVisible}
95:     onRequestClose={() => {alert("Modal has been closed.")}}
96:   >
97:     <View style={styles.modal}>
98:       <View>
99:         <TouchableOpacity style={styles.closeButton}
100:           onPress={() => this.setState({modalVisible: false})}>
101:           <Text>X</Text>
102:         </TouchableOpacity>
103:         <Text>Please select coolness of this picture.</Text>
104:         <Text style={styles.headerText}>{this.state.value}</Text>
105:         <Slider maximumValue={3} minimumValue={1} step={1} value={3}
106:           onChange={(value) => this.setState({value: value})}></Slider>
107:         <TouchableOpacity style={styles.submitButton} onPress={() => {
108:           this.putImageIntoList();
109:         }}>
110:         <Text style={styles.buttonText}>OK</Text>
111:       </TouchableOpacity>
112:     </View>
113:   </View>
114: </Modal>
115:
116: <View style={{flex: 1, paddingTop:22}}>
117: <View style={styles.title}><Text style={styles.titleText}>3 Stars</Text></View>
118: <View style={styles.row}>
119: {this.state.star3.map((source, i)=>
120:   <TouchableOpacity onPress={()=> this.openFullScreenModal(source)}>
121:     <Image key={"star3-"+i}
122:       style={styles.image} source={source}></Image>
123:   </TouchableOpacity>)}
124: </View>
125: <View style={styles.title}><Text style={styles.titleText}>2 Stars</Text></View>
126: <View style={styles.row}>
127: {this.state.star2.map((source, i)=>
128:   <TouchableOpacity onPress={()=> this.openFullScreenModal(source)}>
129:     <Image key={"star2-"+i}
130:       style={styles.image} source={source}></Image>
131:   </TouchableOpacity>)}
132: </View>
133: <View style={styles.title}><Text style={styles.titleText}>1 Star</Text></View>
134: <View style={styles.row}>
135: {this.state.star1.map((source, i)=>
136:   <TouchableOpacity onPress={()=> this.openFullScreenModal(source)}>
137:     <Image key={"star1-"+i}
138:       style={styles.image} source={source}></Image>
139:   </TouchableOpacity>)}
140: </View>
141: </View>
142: <View style={styles.rowCenter}>
143:   <TouchableOpacity style={styles.button} onPress={this.takePhoto}>
144:     <Text style={styles.buttonText}>Camera</Text>
145:   </TouchableOpacity>
146:   <TouchableOpacity style={styles.button} onPress={this.chooseImage}>
147:     <Text style={styles.buttonText}>Gallery</Text>
148:   </TouchableOpacity>
149: </View>
150: </View>
151: );
152: }
153: }
154:
```

```
155: const styles = StyleSheet.create({
156:   row:{
157:     flexDirection:'row',
158:     flexWrap: 'wrap'
159:   },
160:   rowCenter:{
161:     flexDirection: 'row',
162:     alignItems: 'center',
163:     justifyContent: 'center'
164:   },
165:   image:{
166:     width:100,
167:     height:100
168:   },
169:   button: {
170:     backgroundColor: 'gray',
171:     width: 150,
172:     height: 50,
173:     borderRadius: 10,
174:     alignItems: 'center',
175:     justifyContent: 'center',
176:     margin: 10
177:   },
178:   buttonText:{
179:     color: 'white'
180:   },
181:   modal:{
182:     height: 200,
183:     width: 300,
184:     marginTop: 200,
185:     padding: 10,
186:     alignSelf: 'center',
187:     backgroundColor:'lightblue',
188:     margin:10,
189:     borderRadius:10,
190:     justifyContent: 'center',
191:     alignItems: 'center'
192:   },
193:   closeButton:{
194:     alignSelf: 'flex-end'
195:   },
196:   submitButton:{
197:     alignSelf: 'center',
198:     backgroundColor: 'darkblue',
199:     width: 100,
200:     height: 44,
201:     borderRadius: 10,
202:     alignItems: 'center',
203:     justifyContent: 'center',
204:     margin: 10
205:   },
206:   headerText:{
207:     fontSize: 20,
208:     alignSelf: 'center'
209:   },
210:   title:{
211:     backgroundColor:'gray',
212:     padding:5
213:   },
214:   titleText:{
215:     color: 'white'
216:   }
217: });
218:
219: AppRegistry.registerComponent('L10_CameraRollPicker', () => L10_CameraRollPicker);
```

```
1: var Platform = require('react-native').Platform;
2: var ImagePicker = require('react-native-image-picker');
3: import React, {Component} from 'react';
4: import {
5:   AppRegistry,
6:   StyleSheet,
7:   Text,
8:   View,
9:   Image,
10:  TouchableOpacity,
11:  Modal,
12:  Slider,
13:  AsyncStorage
14: } from 'react-native';
15:
16: export default class L10_CameraRollPicker extends Component {
17:   constructor(props) {
18:     super(props);
19:     this.state = {
20:       star3: [],
21:       star2: [],
22:       star1: [],
23:       image: null,
24:       modalVisible: false,
25:       value: 3,
26:       fullScreenImage: null,
27:       fullScreenModalVisible: false
28:     };
29:     AsyncStorage.getItem('state').then((myState) => {
30:       console.log(myState);
31:       this.setState(JSON.parse(myState || '{}'));
32:     });
33:     this.clearState = this.clearState.bind(this);
34:     this.takePhoto = this.takePhoto.bind(this);
35:     this.chooseImage = this.chooseImage.bind(this);
36:     this.setImage = this.setImage.bind(this);
37:     this.putImageIntoList = this.putImageIntoList.bind(this);
38:     this.openFullScreenModal = this.openFullScreenModal.bind(this);
39:   }
40:
41:   saveState(){
42:     AsyncStorage.setItem('state', JSON.stringify(this.state));
43:   }
44:
45:   clearState(){
46:     AsyncStorage.removeItem('state');
47:     this.setState({star3:[], star2:[], star1:[]});
48:   }
49:
50:   putImageIntoList() {
51:     this.setState({
52:       ['star' + this.state.value]: [
53:         ...this.state['star' + this.state.value],
54:         this.state.image
55:       ],
56:       modalVisible: false,
57:       value: 3
58:     });
59:     setTimeout(() => this.saveState(), 0);
60:   }
61:
62:   takePhoto() {
63:     ImagePicker.launchCamera({
64:       noData: true
65:     }, this.setImage);
66:   }
67:
68:   chooseImage() {
69:     ImagePicker.launchImageLibrary({
70:       noData: true
71:     }, this.setImage);
72:   }
73:
74:   openFullScreenModal(source) {
75:     this.setState({fullScreenImage: source, fullScreenModalVisible: true});
76:   }
77: }
```

```

78:   setImage(response) {
79:     console.log('Response = ', response);
80:
81:     if (response.didCancel) {
82:       console.log('User cancelled image picker');
83:     } else if (response.error) {
84:       console.log('ImagePicker Error: ', response.error);
85:     } else if (response.customButton) {
86:       console.log('User tapped custom button: ', response.customButton);
87:     } else {
88:       //If it is iOS, remove 'file://' prefix
89:       let source = {
90:         uri: response.uri.replace('file://', ''),
91:         isStatic: true
92:       };
93:
94:       //If android, don't need to remove the 'file://' prefix
95:       if (Platform.OS === 'android') {
96:         source = {
97:           uri: response.uri,
98:           isStatic: true
99:         };
100:      }
101:      this.setState({image: source, modalVisible: true});
102:    }
103:  }
104:
105:  render() {
106:    return (
107:      <View style={{
108:        flex: 1
109:      }}>
110:        <Modal animationType="slide" transparent={false} visible={this.state.fullScreenMod
alVisible} onRequestClose={() => {
111:          alert("Modal has been closed.")
112:        }}>
113:          <TouchableOpacity style={{
114:            flex: 1
115:          }} onPress={() => this.setState({fullScreenModalVisible: false})}>
116:            <Image style={{
117:              flex: 1
118:            }} source={this.state.fullScreenImage}></Image>
119:          </TouchableOpacity>
120:        </Modal>
121:        <Modal animationType="slide" transparent={true} visible={this.state.modalVisible}
onRequestClose={() => {
122:          alert("Modal has been closed.")
123:        }}>
124:          <View style={styles.modal}>
125:            <View>
126:              <TouchableOpacity style={styles.closeButton} onPress={() => this.setState({mod
alVisible: false})}>
127:                <Text>X</Text>
128:              </TouchableOpacity>
129:              <Text>Please select coolness of this picture.</Text>
130:              <Text style={styles.headerText}>{this.state.value}</Text>
131:              <Slider maximumValue={3} minimumValue={1} step={1} value={3} onChange={(v
alue) => this.setState({value: value})}></Slider>
132:              <TouchableOpacity style={styles.submitButton} onPress={() => {
133:                this.putImageIntoList();
134:              }}>
135:                <Text style={styles.buttonText}>OK</Text>
136:              </TouchableOpacity>
137:            </View>
138:          </View>
139:        </Modal>
140:
141:        <View style={{
142:          flex: 1,
143:          paddingTop: 22
144:        }}>
145:          <View style={styles.title}>
146:            <Text style={styles.titleText}>3 Stars</Text>
147:          </View>
148:          <View style={styles.row}>
149:            {this.state.star3.map((source, i) => <TouchableOpacity onPress={() => this.openF
ullScreenModal(source)}>

```

```
150:         <Image key={"star3-" + i} style={styles.image} source={source}></Image>
151:         </TouchableOpacity>))}
152:     </View>
153:     <View style={styles.title}>
154:       <Text style={styles.titleText}>2 Stars</Text>
155:     </View>
156:     <View style={styles.row}>
157:       {this.state.star2.map((source, i) => <TouchableOpacity onPress={() => this.openF
ullScreenModal(source)}>
158:         <Image key={"star2-" + i} style={styles.image} source={source}></Image>
159:         </TouchableOpacity>))}
160:     </View>
161:     <View style={styles.title}>
162:       <Text style={styles.titleText}>1 Star</Text>
163:     </View>
164:     <View style={styles.row}>
165:       {this.state.star1.map((source, i) => <TouchableOpacity onPress={() => this.openF
ullScreenModal(source)}>
166:         <Image key={"star1-" + i} style={styles.image} source={source}></Image>
167:         </TouchableOpacity>))}
168:     </View>
169:   </View>
170:   <View style={styles.rowCenter}>
171:     <TouchableOpacity style={styles.button} onPress={this.takePhoto}>
172:       <Text style={styles.buttonText}>Camera</Text>
173:     </TouchableOpacity>
174:     <TouchableOpacity style={styles.button} onPress={this.chooseImage}>
175:       <Text style={styles.buttonText}>Gallery</Text>
176:     </TouchableOpacity>
177:   </View>
178:   <View style={styles.rowCenter}>
179:     <TouchableOpacity style={styles.button} onPress={this.clearState}>
180:       <Text style={styles.buttonText}>Clear</Text>
181:     </TouchableOpacity>
182:   </View>
183: </View>
184:   );
185: }
186: }
187:
188: const styles = StyleSheet.create({
189:   row: {
190:     flexDirection: 'row',
191:     flexWrap: 'wrap'
192:   },
193:   rowCenter: {
194:     flexDirection: 'row',
195:     alignItems: 'center',
196:     justifyContent: 'center'
197:   },
198:   image: {
199:     width: 100,
200:     height: 100
201:   },
202:   button: {
203:     backgroundColor: 'gray',
204:     width: 150,
205:     height: 50,
206:     borderRadius: 10,
207:     alignItems: 'center',
208:     justifyContent: 'center',
209:     margin: 10
210:   },
211:   buttonText: {
212:     color: 'white'
213:   },
214:   modal: {
215:     height: 200,
216:     width: 300,
217:     marginTop: 200,
218:     padding: 10,
219:     alignSelf: 'center',
220:     backgroundColor: 'lightblue',
221:     margin: 10,
222:     borderRadius: 10,
223:     justifyContent: 'center',
224:     alignItems: 'center'
```

```
225:   },
226:   closeButton: {
227:     alignSelf: 'flex-end'
228:   },
229:   submitButton: {
230:     alignSelf: 'center',
231:     backgroundColor: 'darkblue',
232:     width: 100,
233:     height: 44,
234:     borderRadius: 10,
235:     alignItems: 'center',
236:     justifyContent: 'center',
237:     margin: 10
238:   },
239:   headerText: {
240:     fontSize: 20,
241:     alignSelf: 'center'
242:   },
243:   title: {
244:     backgroundColor: 'gray',
245:     padding: 5
246:   },
247:   titleText: {
248:     color: 'white'
249:   }
250: });
251:
252: AppRegistry.registerComponent('L10_CameraRollPicker', () => L10_CameraRollPicker);
```