

Exercise (Instructions): Activity Indicator

Objectives and Outcomes

In this exercise you will create a new Loading Component that will make use of the Activity Indicator component to display a loading message. At the end of this exercise you will be able to:

- Set up a Loading Component using the Activity Indicator component
- Trigger the showing of the loading message when data is being fetched from the server

Loading Component

- Add a new file named LoadingComponent.js to the components folder and update it as follows:

```
1 import React from 'react';
2 import {
3   ActivityIndicator,
4   StyleSheet,
5   Text,
6   View,
7 } from 'react-native'
8
9 const styles = StyleSheet.create({
10   loadingView: {
11     alignItems: 'center',
12     justifyContent: 'center',
13     flex: 1
14   },
15   loadingText: {
16     color: '#512DA8',
17     fontSize: 14,
18     fontWeight: 'bold'
19   },
20 })
```

- Then, open AboutComponent.js and update it as follows:

```
1 . . .
2
3 import { Loading } from './LoadingComponent';
4
5 . . .
6
7 if (this.props.leaders.isLoading) {
8   return(
9     <ScrollView>
10       <History />
11       <Card>
12         title='Corporate Leadership'>
13         <Loading />
14       </Card>
15     </ScrollView>
16   );
17 }
18 else if (this.props.leaders.errMess) {
19   return(
20     <ScrollView>
21       <History />
22       <Card>
23         title='Corporate Leadership'>
24         <Text>{this.props.leaders.errMess}</Text>
25       </Card>
26     </ScrollView>
27   );
28 }
29 else {
30   return(
31     <ScrollView>
32       <History />
33     </ScrollView>
34   );
35 }
36 . . .
```

- Then, open HomeComponent.js and update it as follows:

```
1 . . .
2
3 import { Loading } from './LoadingComponent';
4
5 . . .
6
7 if (props.isLoading) {
8   return(
9     <Loading />
10   );
11 }
12 else if (props.errMess) {
13   return(
14     <View>
15       <Text>{props.errMess}</Text>
16     </View>
17   );
18 }
19 else {
20   if (item != null) {
21     return(
22       <Card>
23         featuredTitle={item.name}
24         featuredSubtitle={item.designation}
25         image={{uri: baseUrl + item.image}}>
26         <Text>
27           style={{margin: 10}}>
28           <RenderItem item={this.props.dishes.dishes.filter((dish) => dish
29             .featured[0])} />
30         </Text>
31       </Card>
32     );
33   }
34 }
```

```

40         isLoading={this.props.dishes.isLoading}
41         erreMess={this.props.dishes.erreMess}
42       />
43     <RenderItem item={this.props.promotions.promotions.filter((promo
44       ) => promo.featured)[0]}
45       isLoading={this.props.promotions.isLoading}
46       erreMess={this.props.promotions.erreMess}
47     />
48     <RenderItem item={this.props.leaders.leaders.filter((leader) =>
49       leader.featured)[0]}
50       isLoading={this.props.leaders.isLoading}
51       erreMess={this.props.leaders.erreMess}
52     />

```

- Then, open MenuComponent.js and update it as follows:

```

1   . . .
2
3   import { Loading } from './LoadingComponent';
4
5   . . .
6   if (this.props.dishes.isLoading) {
7     return(
8       <Loading />
9     );
10  }
11  else if (this.props.dishes.erreMess) {
12    return(
13      ...
23      keyExtractor={item => item.id.toString()}
24    />
25  );
26  }
27
28  . . .

```

- Save all the changes and do a Git commit with the message "Activity Indicator".

Conclusions

In this exercise you have seen the use of the Activity Indicator to show a loading message in your app

Mark as completed