

2019

Coursework Report

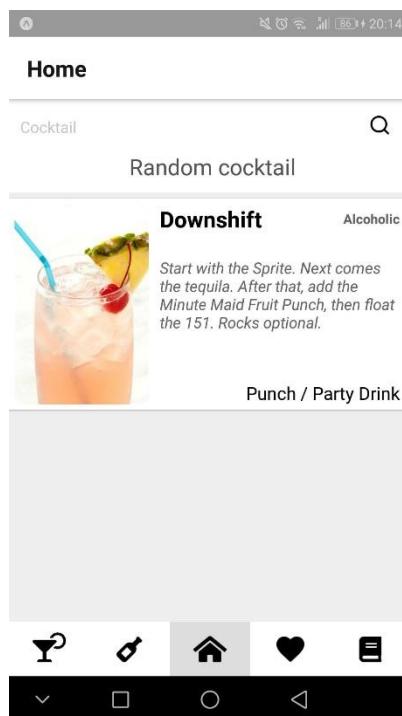
MOBILE APPLICATION DEVELOPMENT – SET08114
ELOI FILAUDEAU

Abstract

In this document, you will find the results of my work on the Mobile Application Development coursework. In this coursework, I have to develop my own Android mobile application. This report is structured in 2 parts. In the first part, you can read the explanation of the work done about the development of my mobile application. And in the second part you can see my evaluation of the work I have done.

Introduction

The mobile application I developed aims to search cocktail recipes by cocktail name, ingredients, etc. I think it can be interesting for a good appetizer to have quickly and simply many drinks recipes. I often look for cocktail recipes at the last minute because I don't know what to do. So, I would like to facilitate this with *CocktailKing* app. In the screenshot below you can see in the navigation bar the possibility to search a cocktail by name for example. The core features of my application are, firstly the search of cocktail and secondly the possibility to save a cocktail as favorite and to share it.

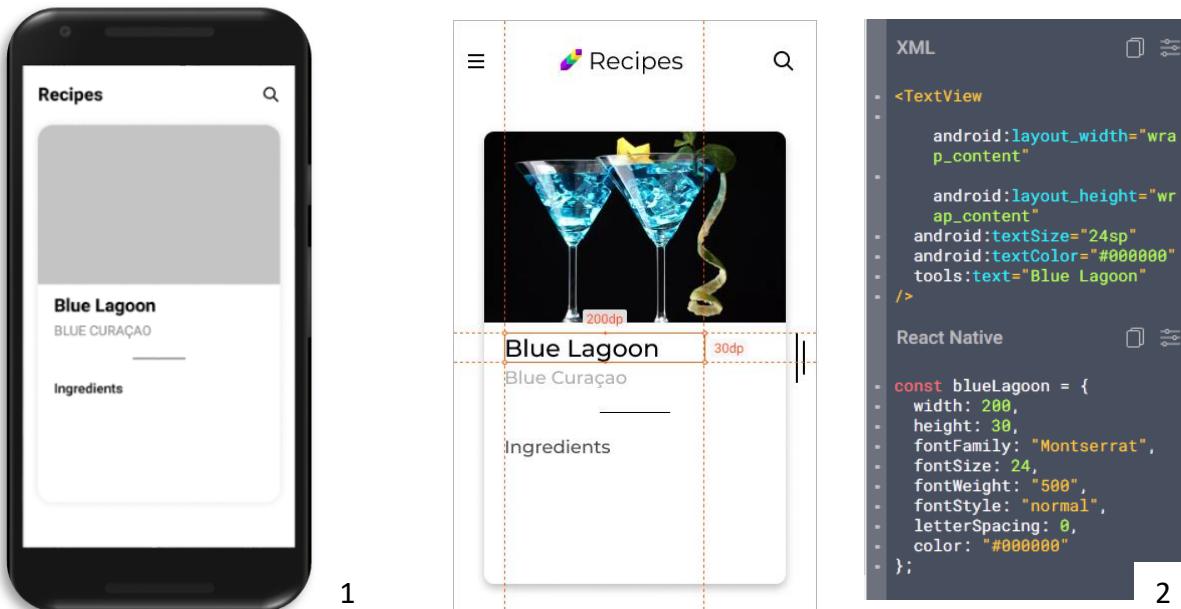


There are a multitude of solutions to develop a mobile application, I decided to use **React Native**. It is an open source and free solution developed by *Facebook* in 2015. There are two main reasons for my choice. First, I already know **Android Studio** because I have done several projects with it (e.g. Hearthstone deck management application) and secondly, **React Native** offers the possibility to make a *cross-platform* application. Indeed, the problem with a native application (developed for a single platform: Android or IOS) is that you have to learn a particular language for each platform (Android: *Java* or *Kotlin*, IOS: *Swift* or *Objective-C*) and develop two different applications adapted to each platform. Whereas in the case of **React Native** we develop only one application, only in *Javascript*, which will be compatible with IOS and Android. So, we save time and implementation learning.

Software design

Concerning the research of the design of my application, I made a first mockup draft on paper (see Appendix 1 - Mockup rough). I have drawn the main views, the search, the display of a cocktail, the menus...

Then I reproduced this basic design on **Figma** (it is an equivalent of Sketch, online and for all OS). During the design process I also researched mockup in design banks like **Dribbble** to find ideas in the current trend. Finally, I used **Zeplin** to retrieve the design created on **Figma** and make it into **React Native** code. That is really useful for the design part and style sheets.

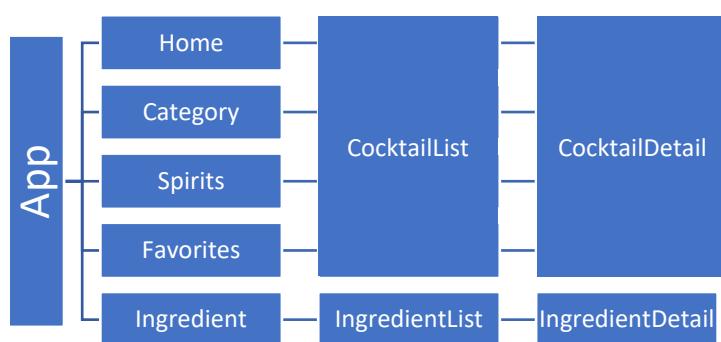


1 – Figma mockup view CocktailDetail

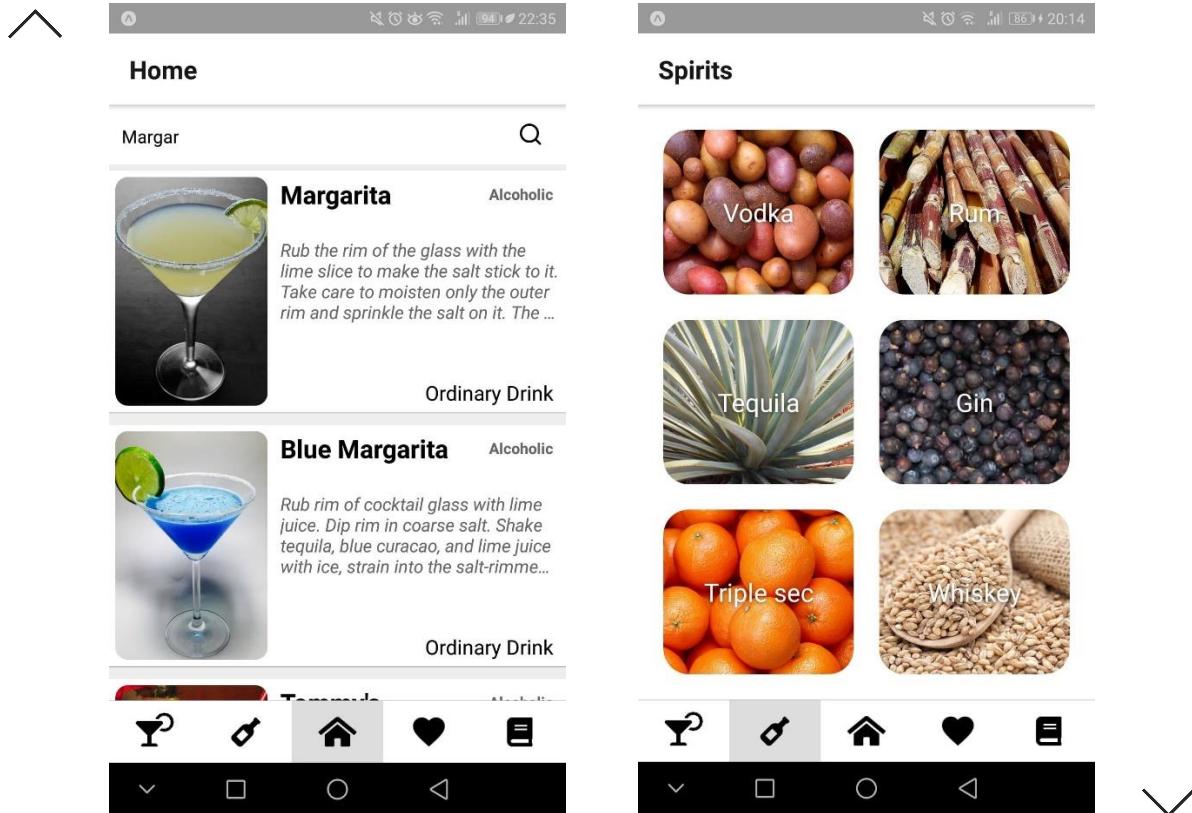
2 – Zeplin work space with the React Native code in the right panel

Application functionalities

My application is built around a tabNavigator that allows you to access all the necessary tabs. The Home, Category, Spirits, Favorites and Ingredients pages are directly accessible from this navigation bar. Here is the skeleton of my application.



Home and Search page: the home page allows you to search for a cocktail or cocktails by name, to start the search you can press the loupe icon or the keyboard entry key on the smartphone (which makes the keyboard disappear at the same time). Moreover, when you arrive on the page without having done a search (at the launch of the application), a random cocktail is displayed to give ideas if you don't have any, it disappears when you launch a search.



Spirits page: this page allows you to search for cocktails by base spirits, there is a choice between Vodka, Rum, Tequila, Gin, Triple sec and Whiskey. To display the search cocktails, I make an API call (see below) with the method to get cocktails according to an ingredient (which is here the base spirit) and I pass the result to the *CocktailList* view to display it.

```
export function getCocktailByIngredientFromApi (ingredient) {
  return fetch('https://www.thecocktaildb.com/api/json/v1/' + API_TOKEN + '/filter.php?i=' + ingredient)
    .then((response) => response.json())
    .catch((error) => console.error(error));
}
```

Category page: like the previous page, it allows a specific search by categories. The operation is exactly the same but with a different API call. To be able to put an image in the background of an area, I had to use the *ImageBackground* **React Native** component in addition to the *TouchableOpacity* which allows to create a touchable area (see below).

```

<TouchableOpacity style={styles.backgroundContainer} onPress={() =>
  ...
  <ImageBackground ...
    source={require('../assets/ordinary_drink.jpg')}
    resizeMode='cover'
    imageStyle={{borderRadius: 20}}
    style={styles.backgroundImage}
  >
    ...
    <Text style={styles.default_text}>Ordinary Drink</Text>
  </ImageBackground>
}
</TouchableOpacity>

```

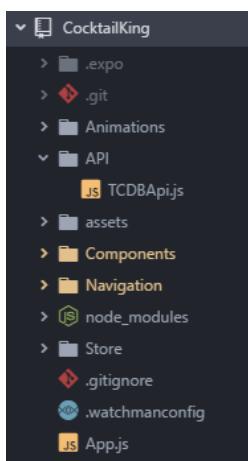


Favorites page: this page retrieves in the global state of the redux store the table of favorite cocktails to display them in a *CocktailList*. Favorite cocktails remain in memory even when the application is closed. To add a cocktail as a favorite, simply touch the heart icon that is linked to a global state in the application, which allows you to add the cocktail to the Favorites tab.

Ingredient page: the *Ingredient* page just lists the ingredients, in alphabetical order, that may be required and their description. When you look at the recipe of a cocktail you can click on an ingredient and be redirected directly to its detail.

```
ingredient_sort = data.drinks.sort(function(a, b) {
  var nameA=a.strIngredient1.toLowerCase(), nameB=b.strIngredient1.toLowerCase()
  if (nameA < nameB) //sort string ascending
    return -1
  if (nameA > nameB)
    return 1
  return 0 //default return value (no sorting)
```

This function sort the ingredient objects in alphabetical order thanks to a comparison between the name of the ingredient (strIngredient1 property of the object ingredient). That is better for research.



On the software side, the code treeview is constructed as described below:

- Animations contains the animations
- API contains API call methods
- assets contains icons and images (png, jpg, svg)
- Components contains all the re-usable components that make up the application
- Navigation contains the creation and configuration of the navigation through the tabNavigator
- Store contains redux stores that allow to manage the flow of information between views thanks to a global state

In addition to being able to search for a recipe and save a cocktail, you can share it with your friends via SMS, Messenger, Instagram, Slack... When sharing, the message sent contains the name of the cocktail and the link of its picture, which allows applications like Messenger to display the picture directly (See Appendix 2 – CocktailDetail, Ingredient and Share View).

Evaluation

Original

At first, I wanted to make a cocktail recipe application to make cocktails and keep our favourites in mind. So I have largely implemented the application I was expecting, which is why I am quite satisfied with my work.

My only regret is the lack of time to do more and not being able to justify the description text on android because **React Native** doesn't implement it yet and you have to use dirty tricks to do it in a less than optimal way, so I didn't do it.

Other apps

To realize my application, I initially did a "market study" to see what already existed in the sector. So, I downloaded and tested several cocktail applications and brainstormed ideas for features and navigation. However, most of the applications on the market are not very

advanced in terms of design and functionality, so I am easily in the "top" part. But I found a very well-designed application, with a lot of design work (all the cocktails redesigned) and a very good user experience. This application probably uses its own database because there is much more information about each cocktail.

About my own application, I think that's the negative point, there are few free and open access cocktail databases. The one I have chosen is the best but remains lacking in information and rather badly structured. For example, ingredients and their measurements are not in tables but in numbered variables, which prevents iteration on a table for reading and creates many null or empty variables.

```

strIngredient1:      "Tequila"
strIngredient2:      "Triple sec"
strIngredient3:      "Lime juice"
strIngredient4:      "Salt"
strIngredients:      ""
strIngredient6:      ""
strIngredient7:      ""
strIngredient8:      ""
strIngredient9:      ""
strIngredient10:     ""
strIngredient11:     ""

```

Extract from the Json returned by a cocktail

User experience

UX is really important because it is what keeps the user from uninstalling the application quickly and enjoys using it. To have a better approach to the subject I have done two complete online courses, the first on the User Interface and the second on User Xperience. This has allowed me to improve my knowledge and skills.

Concretely, not to mention the design, I added loading animations when the API call to get cocktails takes longer than expected, as well as a cocktail display animation that performs a slide from the right side of the screen.

Improvements

For the rest of the development, being very interested in the field of music in general, I plan to add the possibility of playing a playlist corresponding to each cocktail during the aperitif. So, I started looking at the **Spotify** and **Soundcloud** APIs to find out how to get music and playlists and display them.

I would like to add the possibility to create a sort of bar shelf with the ingredients we have at home. And that would allow us to see the cocktails we can do without buy anything.

With the next version of **React Native**, *textAlign* justify should be implemented for Android which would allow me to improve the design.

Sources

Here is the non-exhaustive sources I used for this coursework

Mobile Application Development courses – Sally Smith and Andreas Steyven

Dribbble : <https://dribbble.com>

Zeplin : <https://app.zeplin.io>

Figma : <http://figma.com/>

Thecocktaildb : <https://www.thecocktaildb.com>

Expo : <https://expo.io/>

APK creation : <https://stackoverflow.com/questions/42967465/created-an-app-with-create-react-native-app-how-to-publish-it-to-the-google-play-store>

React Native documentation :

<https://facebook.github.io/react-native/docs>

<https://reactnavigation.org>

React Native property : <https://github.com/vhpoet/react-native-styling-cheat-sheet#text>

Conditional render : <https://kylewbanks.com/blog/how-to-conditionally-render-a-component-in-react-native>

Text shadow property : <https://stackoverflow.com/questions/46387355/text-shadow-in-react-native>

Icon : <https://fontawesome.com>

Design color scheme : <https://coolors.co>

Sorting an array of object : <http://www.javascriptkit.com/javatutors/arraysort2.shtml>

Cocktail application studied :

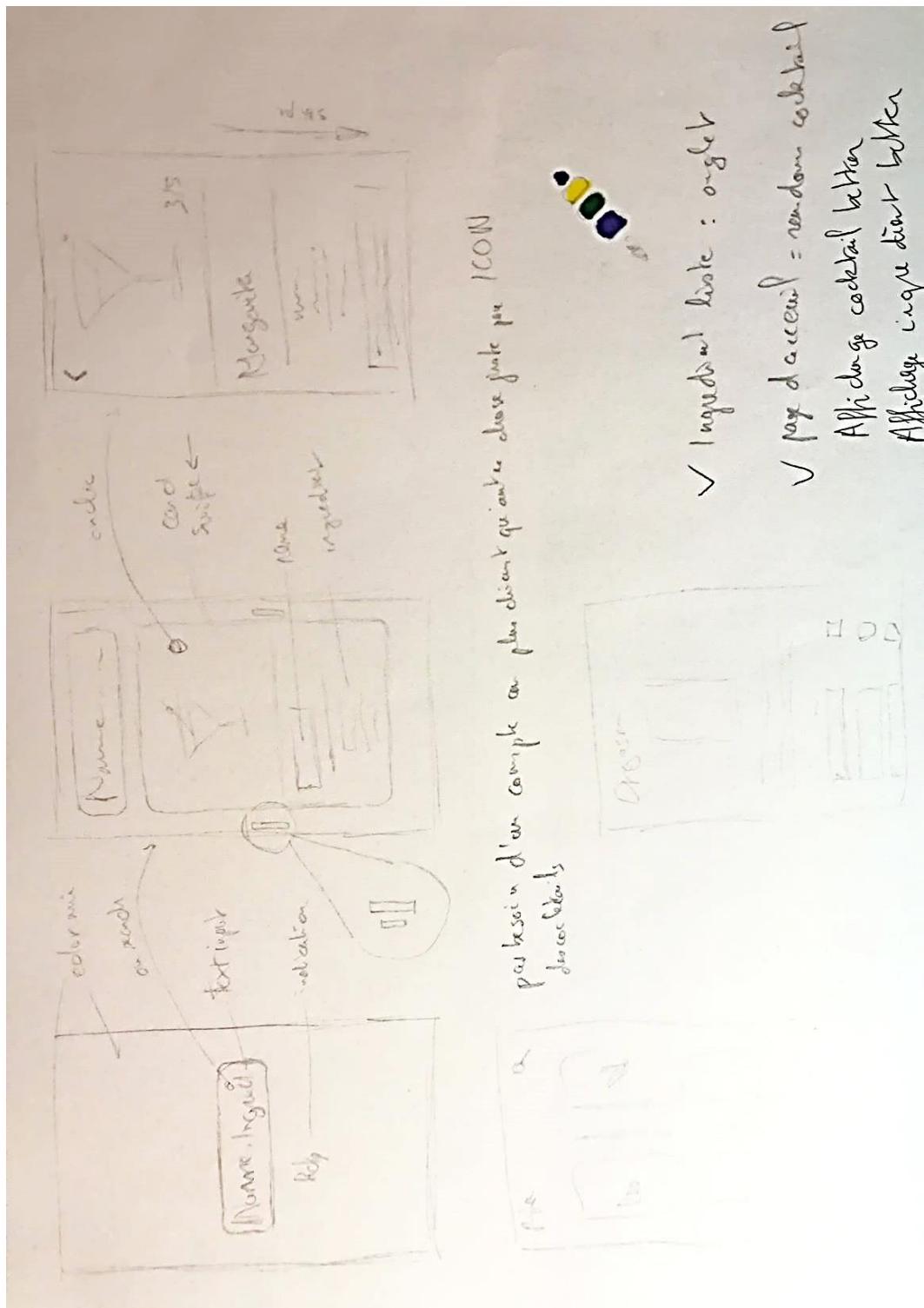
<https://play.google.com/store/apps/details?id=com.florianisme.cocktailer>

<https://play.google.com/store/apps/details?id=com.mybarapp.free>

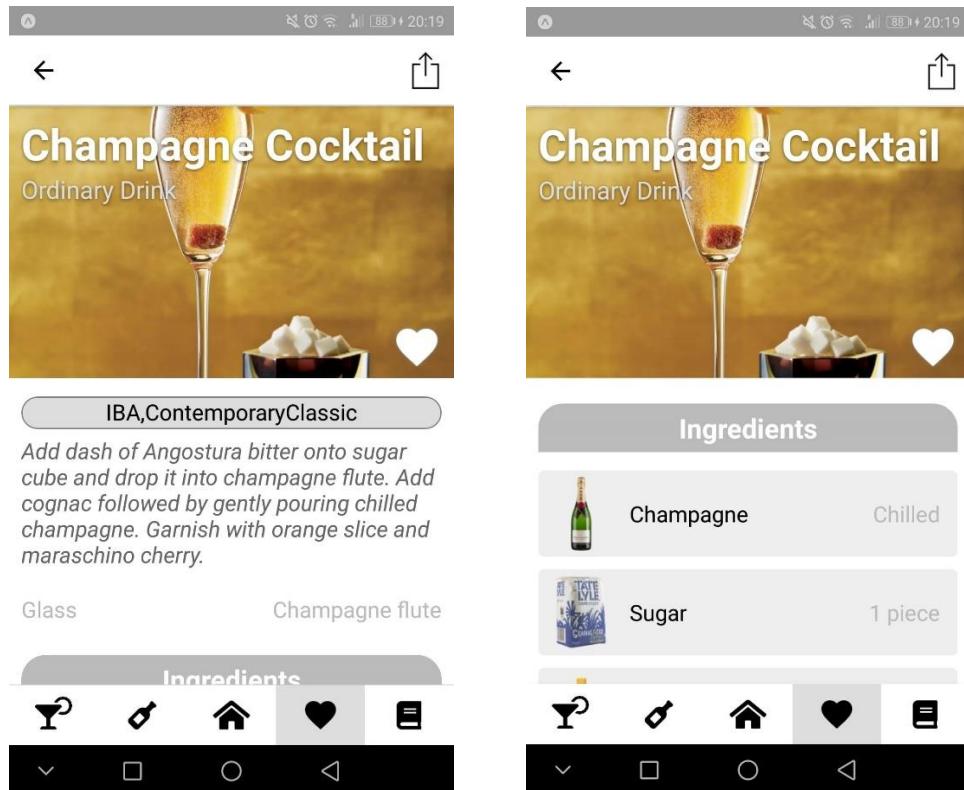
<https://play.google.com/store/apps/details?id=hu.distinction.cflow>

APPENDIX

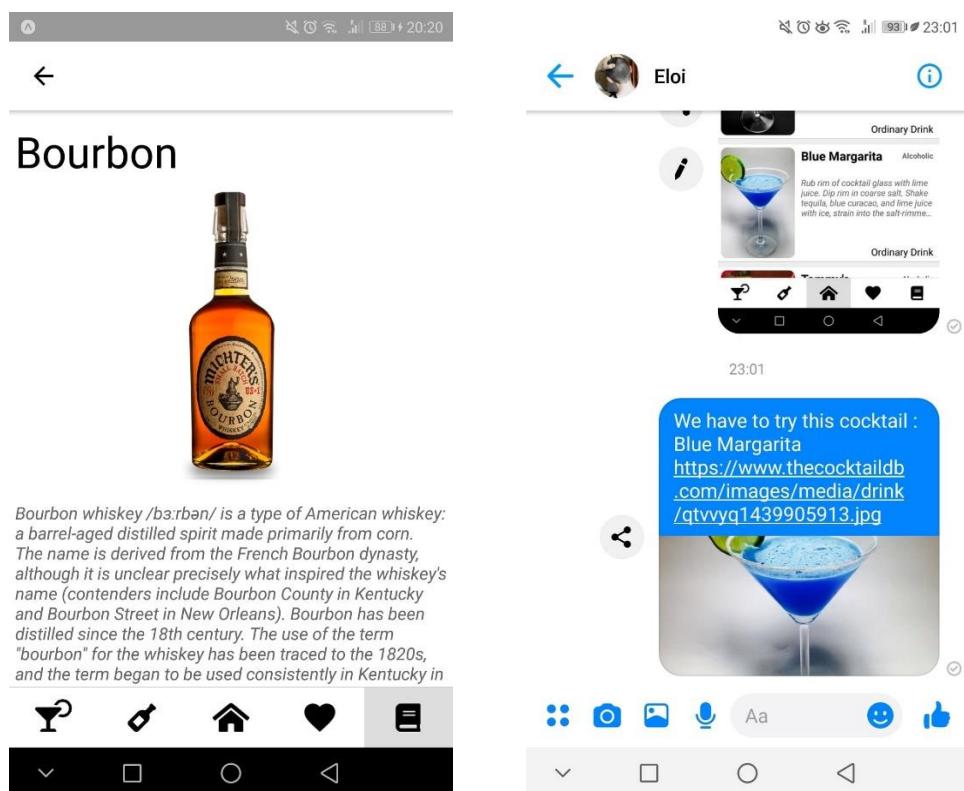
Appendix 1 - Mockup rough



Appendix 2 – CocktailDetail, Ingredient and Share View



CocktailDetail view with tags, recipe, ingredients and glass



Ingredient view with description

Cocktail share on Messenger

```
1 import React from 'react'
2 import { } from 'react-native'
3 import CocktailList from './CocktailList'
4 import { getRandomCocktailFromApi } from '../API/TCDBApi'
5
6 class News extends React.Component {
7     constructor(props) {
8         super(props)
9         this.state = {
10             cocktails: [],
11             isLoading: false
12         }
13         this._loadCocktails = this._loadCocktails.bind(
14             this)
15     }
16     componentDidMount() {
17         this._loadCocktails()
18     }
19
20     _loadCocktails() {
21         this.setState({ isLoading: true })
22         getRandomCocktailFromApi().then(data => {
23             this.setState({
24                 cocktails: [ ...data.drinks ],
25                 isLoading: false
26             })
27         })
28     }
29
30     render() {
31         return (
32             <CocktailList
33                 cocktails={this.state.cocktails}
34                 navigation={this.props.navigation}
35                 loadCocktails={this._loadCocktails}
36                 favoriteList={false}
37             />
38         )
39     }
40 }
41
42 export default News
43
```

```
1 import React from 'react'
2 import { StyleSheet, View, Text, ActivityIndicator } from
  'react-native'
3 import { getCocktailByIngredientFromApi } from '../API/
  TCDBApi'
4 import CocktailItem from './CocktailItem'
5 import CocktailList from './CocktailList'
6
7 class Alcool extends React.Component {
8
9   constructor(props) {
10     super(props)
11     this.state = {
12       cocktails: [],
13       isLoading: false
14     }
15   }
16   _displayLoading() {
17     if (this.state.isLoading) {
18       return (
19         <View style={styles.loading_container}>
20           <ActivityIndicator size='large' />
21         </View>
22       )
23     }
24   }
25
26   componentDidMount() {
27     this._loadCocktails(this.props.navigation.state.
  params.alcool)
28   }
29   _loadCocktails(ingredient) {
30     this.setState({ isLoading: true })
31     getCocktailByIngredientFromApi(ingredient).then(
32       data => {
33         this.setState({ cocktails: data.drinks,
  isLoading: false })
34       }
35     )
36   }
37   render() {
38     return (
39       <View style={styles.main_container}>
40         <View style={styles.result_container}>
41           <CocktailList
```

```
41                               cocktails={this.state.cocktails}
42                               navigation={this.props.navigation}
43                               loadCocktails={this._loadCocktails
44                               }
45                               />
46                               {this._displayLoading() }
47                           </View>
48                       </View>
49                   )
50   }
51
52 const styles = StyleSheet.create({
53   main_container: {
54     flex: 1
55   },
56   result_container: {
57     flex: 9
58   },
59   default_text: {
60     marginLeft: 5,
61     marginRight: 5,
62     paddingLeft: 5,
63     paddingBottom: 10,
64     fontSize: 20,
65     textAlign: 'center',
66     color: '#555'
67   },
68   loading_container: {
69     position: 'absolute',
70     left: 0,
71     right: 0,
72     top: 0,
73     bottom: 0,
74     alignItems: 'center',
75     justifyContent: 'center'
76   }
77 })
78
79 export default Alcool
80
```

```
1 import React from 'react'
2 import { StyleSheet, View, TextInput, Button, FlatList,
3   Text, ActivityIndicator } from 'react-native'
4 import { getCocktailFromApiWithSearchedText } from '../API
/TCDBApi'
5 import Icon from 'react-native-vector-icons/Feather'
6 import CocktailItem from './CocktailItem'
7 import CocktailList from './CocktailList'
8 import News from './News'
9
10 class Search extends React.Component {
11   constructor(props) {
12     super(props)
13     this.searchedText = ""
14     this.state = {
15       cocktails: [],
16       isLoading: false
17     }
18   }
19   _loadCocktail() {
20     if (this.searchedText.length > 0) {
21       this.setState({ isLoading: true })
22       getCocktailFromApiWithSearchedText(this.
23         searchedText).then(data => {
24         this.setState({ cocktails: data.drinks,
25           isLoading: false })
26       })
27     }
28     _searchTextInputChanged(text) {
29       this.searchedText = text
30     }
31     _displayLoading() {
32       if (this.state.isLoading) {
33         return (
34           <View style={styles.loading_container}>
35             <ActivityIndicator size='large' />
36           </View>
37         )
38       }
39     }
40   }
41   render() {
```

```
42         if (this.searchedText == "") {
43             result = <View style={styles.result_container}>
44                 <Text style={styles.default_text}>Random
45                 cocktail</Text>
46                 <News
47                     navigation={this.props.navigation}
48                 />
49             </View>
50         } else if (this.state.cocktails == null) {
51             result = <Text style={styles.noresult}>No
52             results</Text>
53         } else {
54             result = <CocktailList
55                 cocktails={this.state.cocktails}
56                 navigation={this.props.navigation}
57                 loadCocktails={this._loadCocktails}
58             />
59         }
60
61         return (
62             <View style={styles.main_container}>
63                 <View style={styles.search_container}>
64                     <TextInput
65                         style={styles.textinput}
66                         placeholder='Cocktail'
67                         onChangeText={(text) => this.
68                         _searchTextInputChanged(text)}
69                         onSubmitEditing={() => this.
70                         _loadCocktail()}
71                     />
72                     <Icon.Button
73                         name="search"
74                         color="#000000"
75                         borderRadius={0}
76                         style={styles.button}
77                         onPress={() => this._loadCocktail(
78                         )} }
79                         ></Icon.Button>
80                 </View>
81             <View style={styles.result_container}>
82                 {result}
83                 {this._displayLoading()} }
84             </View>
85         </View>
```

```
81          )
82      }
83  }
84
85  const styles = StyleSheet.create({
86      main_container: {
87          flex: 1
88      },
89      search_container: {
90          flex: 1,
91          flexDirection: 'row'
92      },
93      result_container: {
94          flex: 9
95      },
96      textinput: {
97          marginLeft: 5,
98          marginRight: 5,
99          height: 50,
100         flex: 9,
101         paddingLeft: 5
102     },
103     default_text: {
104         marginLeft: 5,
105         marginRight: 5,
106         paddingLeft: 5,
107         paddingBottom: 10,
108         fontSize: 20,
109         textAlign: 'center',
110         color: '#555'
111     },
112     button: {
113         flex: 1,
114         backgroundColor: 'white'
115     },
116     noresult: {
117         paddingLeft: 5,
118         marginLeft: 5,
119         marginRight: 5,
120         marginTop: 20,
121         textAlign: 'center',
122         height: 50,
123         color: '#555'
124     },
125     loading_container: {
```

```
126         position: 'absolute',
127         left: 0,
128         right: 0,
129         top: 0,
130         bottom: 0,
131         alignItems: 'center',
132         justifyContent: 'center'
133     }
134 )
135
136 export default Search
137
```

```
1 import React from 'react'
2 import { StyleSheet, View, Text, ActivityIndicator } from
  'react-native'
3 import { getCocktailByCategoryFromApi } from '../API/
  TCDBApi'
4 import CocktailItem from './CocktailItem'
5 import CocktailList from './CocktailList'
6
7 class Category extends React.Component {
8
9   constructor(props) {
10     super(props)
11     this.state = {
12       cocktails: [],
13       isLoading: false
14     }
15   }
16   _displayLoading() {
17     if (this.state.isLoading) {
18       return (
19         <View style={styles.loading_container}>
20           <ActivityIndicator size='large' />
21         </View>
22       )
23     }
24   }
25
26   componentDidMount() {
27     this._loadCocktails(this.props.navigation.state.
  params.category)
28   }
29   _loadCocktails(category) {
30     this.setState({ isLoading: true })
31     getcocktailByCategoryFromApi(category).then(data =
  > {
32       this.setState({ cocktails: data.drinks,
  isLoading: false })
33     })
34   }
35
36   render() {
37     return (
38       <View style={styles.main_container}>
39         <View style={styles.result_container}>
40           <CocktailList
```

```
41                               cocktails={this.state.cocktails}
42                               navigation={this.props.navigation}
43                               loadCocktails={this._loadCocktails
44                               }
45                               />
46                               {this._displayLoading() }
47                           </View>
48                       </View>
49                   )
50   }
51
52 const styles = StyleSheet.create({
53   main_container: {
54     flex: 1
55   },
56   result_container: {
57     flex: 9
58   },
59   default_text: {
60     marginLeft: 5,
61     marginRight: 5,
62     paddingLeft: 5,
63     paddingBottom: 10,
64     fontSize: 20,
65     textAlign: 'center',
66     color: '#555'
67   },
68   loading_container: {
69     position: 'absolute',
70     left: 0,
71     right: 0,
72     top: 0,
73     bottom: 0,
74     alignItems: 'center',
75     justifyContent: 'center'
76   }
77 })
78
79 export default Category
80
```

```
1 import React from 'react'
2 import { StyleSheet, Text } from 'react-native'
3 import CocktailList from './CocktailList'
4 import { connect } from 'react-redux'
5
6 class Favorites extends React.Component {
7
8     render() {
9         return (
10             <CocktailList
11                 cocktails={this.props.favoritesCocktail}
12                 navigation={this.props.navigation}
13                 favoriteList={true}
14             />
15         )
16     }
17 }
18
19 const styles = StyleSheet.create({})
20
21 const mapStateToProps = state => {
22     return {
23         favoritesCocktail: state.favoritesCocktail
24     }
25 }
26
27 export default connect(mapStateToProps)(Favorites)
28
```

```
1 import React from 'react'
2 import { } from 'react-native'
3 import IngredientList from './IngredientList'
4 import { getIngredientListFromApi } from '../API/TCDBApi'
5
6 class Ingredient extends React.Component {
7     constructor(props) {
8         super(props)
9         this.state = {
10             ingredients: [],
11             isLoading: false
12         }
13         this._loadIngredients = this._loadIngredients.bind
14     (this)
15     }
16     componentDidMount() {
17         this._loadIngredients()
18     }
19
20     _loadIngredients() {
21         this.setState({ isLoading: true })
22         getIngredientListFromApi().then(data => {
23             ingredient_sort = data.drinks.sort(function(a,
24             b) {
25                 var nameA=a.strIngredient1.toLowerCase(),
26                 nameB=b.strIngredient1.toLowerCase()
27                 if (nameA < nameB) //sort string ascending
28                     return -1
29                 if (nameA > nameB)
30                     return 1
31                 return 0 //default return value (no
32                 sorting)
33             })
34             this.setState({
35                 ingredients: [...ingredient_sort ],
36                 isLoading: false
37             })
38         })
39     }
40     render() {
41         return (
42             <IngredientList
43                 ingredients={this.state.ingredients}
```

```
42           navigation={this.props.navigation}
43           loadIngredients={this._loadIngredients}
44         />
45       )
46     }
47   }
48
49 export default Ingredient
50
```

```
1 import React from 'react'
2 import { StyleSheet, View, Text, Image, TouchableOpacity } from 'react-native'
3 import FadeIn from '../Animations/FadeIn'
4
5 class CocktailItem extends React.Component {
6
7     _displayFavoriteImage() {
8         // console.log(this.props.isCocktailFavorite);
9         if (this.props.isCocktailFavorite) {
10             return (
11                 <Image
12                     style={styles.favorite_image}
13                     source={require('../assets/ic_favorite.png')} )
14             />
15         )
16     }
17 }
18
19 render() {
20     const { cocktail, displayDetailForCocktail } =
21     this.props
22     return (
23         <FadeIn>
24             <TouchableOpacity style={styles.
25                 main_container} onPress={() => displayDetailForCocktail(
26                     cocktail.idDrink)}>
27                 <Image
28                     style={styles.image}
29                     source={{uri: cocktail.
30                         strDrinkThumb}} )
31                 />
32                 <View style={styles.content_container}>
33                     <View style={styles.
34                         header_container}>
35                         {this._displayFavoriteImage()} <Text style={styles.title_text}
36                         >{cocktail.strDrink}</Text>
37                         <Text style={styles.
38                             alcoholic_text}>{cocktail.strAlcoholic}</Text>
39                     </View>
40                     <View style={styles.
41                         description_container}>
```

```
35                                     <Text style={styles.  
36                                         description_text} numberOfLines={4}>{cocktail.  
37                                         strInstructions}</Text>  
38                                     </View>  
39                                     <View style={styles.  
40                                         category_container}>  
41                                         <Text style={styles.  
42                                         category_text}>{cocktail.strCategory}</Text>  
43                                         </View>  
44                                     </View>  
45                                     </TouchableOpacity>  
46                                     </FadeIn>  
47                                     )  
48                                     }  
49                                     }  
50                                     }  
51                                     }  
52                                     }  
53                                     }  
54                                     }  
55                                     }  
56                                     }  
57                                     }  
58                                     }  
59                                     }  
60                                     }  
61                                     }  
62                                     }  
63                                     }  
64                                     }  
65                                     }  
66                                     }  
67                                     }  
68                                     }  
69                                     }  
70                                     }  
71                                     }  
72                                     }  
73                                     }  
74                                     }  
75                                     }
```

```
76     title_text: {
77         fontWeight: 'bold',
78         fontSize: 20,
79         flex: 1,
80         flexWrap: 'wrap',
81         paddingRight: 5
82     },
83     alcoholic_text: {
84         fontWeight: 'bold',
85         fontSize: 12,
86         paddingTop: 5,
87         paddingRight: 5,
88         color: '#666666'
89     },
90     description_container: {
91         flex: 7
92     },
93     description_text: {
94         fontStyle: 'italic',
95         color: '#666666'
96     },
97     category_container: {
98         flex: 1,
99         paddingBottom: 5
100    },
101    category_text: {
102        textAlign: 'right',
103        paddingRight: 5,
104        fontSize: 16
105    },
106    favorite_image: {
107        width: 25,
108        height: 25,
109        marginRight: 5
110    }
111 })
112
113 export default CocktailItem
114
```

```
1 import React from 'react'
2 import { StyleSheet, FlatList } from 'react-native'
3 import CocktailItem from './CocktailItem'
4 import { connect } from 'react-redux'
5
6 class CocktailList extends React.Component {
7
8     constructor(props) {
9         super(props)
10        this.state = {
11            cocktails: []
12        }
13    }
14
15    _displayDetailForCocktail = (idCocktail) => {
16        this.props.navigation.navigate("CocktailDetail", {
17            idCocktail: idCocktail })
18    }
19
20    render() {
21        return (
22            <FlatList
23                style={styles.list}
24                data={this.props.cocktails}
25                extraData={this.props.favoritesCocktail}
26                keyExtractor={(item) => item.idDrink}
27                renderItem={({item}) => (
28                    <CocktailItem
29                        cocktail={item}
30                        isCocktailFavorite={(this.props.
31                            favoritesCocktail.findIndex(cocktail => cocktail.idDrink =
32                            item.idDrink) !== -1) ? true : false}
33                        displayDetailForCocktail={this.
34                            _displayDetailForCocktail}
35                        />
36                ) }
37            )
38    }
39
40    const styles = StyleSheet.create({
41        list: {
42            flex: 1,
43            backgroundColor: '#eee'
```

```
42      }
43  })
44
45 const mapStateToProps = state => {
46   return {
47     favoritesCocktail: state.favoritesCocktail
48   }
49 }
50
51 export default connect(mapStateToProps)(CocktailList)
52
```

```
1 import React from 'react'
2 import { StyleSheet, View, Text, Image, ActivityIndicator,
3   ScrollView, FlatList, TouchableOpacity, Share } from 'react-native'
4 import { getCocktailDetailFromApi } from '../API/TCDBApi'
5 import { connect } from 'react-redux'
6 import Ingredient from './IngredientMeasure'
7
8 class CocktailDetail extends React.Component {
9   static navigationOptions = ({ navigation }) => {
10     const { params } = navigation.state
11     // access to sharefilm function via navigation
12     params
13     if (params.cocktail != undefined) {
14       return {
15         headerRight: <TouchableOpacity
16           style={styles.
17             share_touchable_headerrightbutton}
18           onPress={() => params.shareCocktail()}>
19             <Image
20               style={styles.share_image}
21               source={require('../assets/
22               ic_share.png')}>
23             />
24           </TouchableOpacity>
25         }
26       }
27     }
28     constructor(props) {
29       super(props)
30       this.state = {
31         cocktail: undefined,
32         isLoading: true
33       }
34       this._shareCocktail = this._shareCocktail.bind(
35         this)
36     }
37     _updateNavigationParams() {
38       this.props.navigation.setParams({
39         shareCocktail: this._shareCocktail,
40         cocktail: this.state.cocktail
41       })
42     }
43   }
44 }
```

```
39     componentDidMount() {
40         const favoriteCocktailIndex = this.props.
41         favoritesCocktail.findIndex(item => item.id === this.props.
42         .navigation.state.params.idCocktail)
43         if (favoriteCocktailIndex !== -1) { // cocktail
44             already in favorite list
45                 // get cocktail infos from global state
46                 this.setState({
47                     cocktail: this.props.favoritesCocktail[
48                     favoriteCocktailIndex]
49                     }, () => { this._updateNavigationParams() })
50             return
51         }
52         // cocktail not in favoritelist
53         // API call to get cocktail infos
54         this.setState({ isLoading: true })
55         getcocktailDetailFromApi(this.props.navigation.
56         state.params.idCocktail).then(data => {
57             this.setState({
58                 cocktail: data.drinks[0],
59                 isLoading: false
60             }, () => { this._updateNavigationParams() })
61         })
62     }
63
64     _displayLoading() {
65         if (this.state.isLoading) {
66             return (
67                 <View style={styles.loading_container}>
68                     <ActivityIndicator size='large' />
69                 </View>
70             )
71         }
72     }
73
74     _toggleFavorite() {
75         const action = { type: "TOGGLE_FAVORITE", value:
76             this.state.cocktail }
77         this.props.dispatch(action)
78     }
79
80     _displayFavoriteImage() {
81         var sourceImage = require('../assets/
82         ic_favorite_border.png')
83         if (this.props.favoritesCocktail.findIndex(item =>
```

```

76     item.idDrink === this.state.cocktail.idDrink) !== -1) {
77         sourceImage = require('../assets/ic_favorite.png')
78     }
79     return (
80         <Image
81             style={styles.favorite_image}
82             source={sourceImage}
83         />
84     )
85 }
86
87 // share cocktail name and picture
88 _shareCocktail() {
89     const { cocktail } = this.state
90     Share.share({ title: 'CocktailKing app', message:
91         'We have to try this cocktail : ' + cocktail.strDrink +
92         ' ' + cocktail.strDrinkThumb })
93 }
94
95 _displaycocktail() {
96     const { cocktail } = this.state
97
98     if (cocktail != undefined) {
99         var ingredient = []
100
101         if (cocktail.strIngredient1 !== "" &&
102             cocktail.strIngredient1 !== null) {ingredient.push(
103             <Ingredient
104                 key='1'
105                 ingredient={cocktail.strIngredient1}
106                 measure={cocktail.strMeasure1}
107                 navigation={this.props.navigation}
108             />
109         ) }
110         if (cocktail.strIngredient2 !== "" &&
111             cocktail.strIngredient2 !== null) {ingredient.push(<
112             Ingredient key='2' ingredient={cocktail.strIngredient2}
113             measure={cocktail.strMeasure2} navigation={this.props.
114             navigation}/>) }
115         if (cocktail.strIngredient3 !== "" &&
116             cocktail.strIngredient3 !== null) {ingredient.push(<
117             Ingredient key='3' ingredient={cocktail.strIngredient3}
118             measure={cocktail.strMeasure3} navigation={this.props.
119             navigation}/>) }

```

```

109          if (cocktail.strIngredient4 !== "" &&
cocktail.strIngredient4 !== null) {ingredient.push(<
Ingredient key='4' ingredient={cocktail.strIngredient4}
measure={cocktail.strMeasure4} navigation={this.props.
navigation}/>) }
110          if (cocktail.strIngredient5 !== "" &&
cocktail.strIngredient5 !== null) {ingredient.push(<
Ingredient key='5' ingredient={cocktail.strIngredient5}
measure={cocktail.strMeasure5} navigation={this.props.
navigation}/>) }
111          if (cocktail.strIngredient6 !== "" &&
cocktail.strIngredient6 !== null) {ingredient.push(<
Ingredient key='6' ingredient={cocktail.strIngredient6}
measure={cocktail.strMeasure6} navigation={this.props.
navigation}/>) }
112          if (cocktail.strIngredient7 !== "" &&
cocktail.strIngredient7 !== null) {ingredient.push(<
Ingredient key='7' ingredient={cocktail.strIngredient7}
measure={cocktail.strMeasure7} navigation={this.props.
navigation}/>) }
113          if (cocktail.strIngredient8 !== "" &&
cocktail.strIngredient8 !== null) {ingredient.push(<
Ingredient key='8' ingredient={cocktail.strIngredient8}
measure={cocktail.strMeasure8} navigation={this.props.
navigation}/>) }
114          if (cocktail.strIngredient9 !== "" &&
cocktail.strIngredient9 !== null) {ingredient.push(<
Ingredient key='9' ingredient={cocktail.strIngredient9}
measure={cocktail.strMeasure9} navigation={this.props.
navigation}/>) }
115          if (cocktail.strIngredient10 !== "" &&
cocktail.strIngredient10 !== null) {ingredient.push(<
Ingredient key='10' ingredient={cocktail.strIngredient10}
measure={cocktail.strMeasure10} navigation={this.props.
navigation}/>) }
116          if (cocktail.strIngredient11 !== "" &&
cocktail.strIngredient11 !== null) {ingredient.push(<
Ingredient key='11' ingredient={cocktail.strIngredient11}
measure={cocktail.strMeasure11} navigation={this.props.
navigation}/>) }
117          if (cocktail.strIngredient12 !== "" &&
cocktail.strIngredient12 !== null) {ingredient.push(<
Ingredient key='12' ingredient={cocktail.strIngredient12}
measure={cocktail.strMeasure12} navigation={this.props.
navigation}/>) }

```

```

118             if (cocktail.strIngredient13 !== "" &&
cocktail.strIngredient13 !== null) {ingredient.push(<
Ingredient key='13' ingredient={cocktail.strIngredient13}
measure={cocktail.strMeasure13} navigation={this.props.
navigation}/>) }
119             if (cocktail.strIngredient14 !== "" &&
cocktail.strIngredient14 !== null) {ingredient.push(<
Ingredient key='14' ingredient={cocktail.strIngredient14}
measure={cocktail.strMeasure14} navigation={this.props.
navigation}/>) }
120             if (cocktail.strIngredient15 !== "" &&
cocktail.strIngredient15 !== null) {ingredient.push(<
Ingredient key='15' ingredient={cocktail.strIngredient15}
measure={cocktail.strMeasure15} navigation={this.props.
navigation}/>) }
121             var tag = null
122             if (cocktail.strTags !== "" && cocktail.
strTags !== null) {
123                 tag = <Text style={styles.tag_text}>{
cocktail.strTags}</Text>
124             }
125
126             return (
127                 <View style={styles.card_container}>
128                     <View style={styles.card}>
129                         <View style={styles.
image_container}>
130                             <View style={styles.
title_container}>
131                                 <Text style={styles.
title_text}>{cocktail.strDrink}</Text>
132                                 <Text style={styles.
subtitle_text}>{cocktail.strCategory}</Text>
133                             </View>
134                             <TouchableOpacity
style={styles.
favorite_container}
136                                     onPress={() => this.
_toggleFavorite()}>
137                                         <this.
_displayFavoriteImage() >
138                                     </TouchableOpacity>
139                                     <Image
style={styles.image}
source={{uri: cocktail.

```

```
141 strDrinkThumb} }  
142                                     />  
143                                 </View>  
144                                 <ScrollView>  
145                                         <View style={styles.  
    description_container}>  
146                                         {tag}  
147                                         <Text style={styles.  
    description_text}>{cocktail.strInstructions}</Text>  
148                                         <View style={styles.  
    glass_container}>  
149                                         <Text style={styles.  
    glass_text_left}>Glass</Text>  
150                                         <Text style={styles.  
    glass_text_right}>{cocktail.strGlass}</Text>  
151                                         </View>  
152                                         <Text style={styles.  
    title_ingredient}>Ingredients</Text>  
153                                         </View>  
154                                         <FlatList  
155                                         data={ingredient}  
156                                         keyExtractor={(item) =>  
    item.key}  
157                                         renderItem={({item}) =>  
    item}  
158                                         />  
159                                         </ScrollView>  
160                                         </View>  
161                                         </View>  
162                                         )  
163                                         }  
164                                         }  
165  
166     render() {  
167         return (  
168             <View style={styles.main_container}>  
169                 {this._displayLoading()}  
170                 {this._displaycocktail()}  
171             </View>  
172         )  
173     }  
174  
175 }  
176  
177 const styles = StyleSheet.create({
```

```
178     main_container: {
179         flex: 1
180     },
181     share_touchable_headerrightbutton: {
182         marginRight: 8
183     },
184     share_image: {
185         width: 30,
186         height: 30
187     },
188     favorite_container: {
189         margin: 10,
190         position: 'absolute',
191         bottom: 0,
192         right: 0,
193         zIndex: 5
194     },
195     favorite_image: {
196         width: 40,
197         height: 40,
198     },
199     image_container: {
200         // flex:3
201     },
202     loading_container: {
203         position: 'absolute',
204         left: 0,
205         right: 0,
206         top: 0,
207         bottom: 0,
208         alignItems: 'center',
209         justifyContent: 'center'
210     },
211     card_container: {
212         flex: 1,
213         marginTop: -4
214     },
215     card: {
216         flex: 1,
217     },
218     title_ingredient: {
219         textAlign: 'center',
220         fontSize: 22,
221         fontWeight: 'bold',
222         color: 'white',
```

```
223     backgroundColor: '#bbb',
224     paddingTop: 5,
225     paddingBottom: 5,
226     borderTopLeftRadius: 20,
227     borderTopRightRadius: 20,
228     margin: 5
229   },
230   image: {
231     height: 220,
232     marginTop: 5,
233     marginBottom: 5
234   },
235   title_container: {
236     position: 'absolute',
237     top: 2,
238     // backgroundColor: 'rgba(200, 200, 200, 0.2)',
239     width: '100%',
240     zIndex: 5
241   },
242   title_text: {
243     textShadowColor: 'rgba(0, 0, 0, 0.5)',
244     textShadowOffset: {width: 0, height: 1},
245     textShadowRadius: 10,
246     fontSize: 35,
247     fontWeight: 'bold',
248     flexWrap: 'wrap',
249     marginLeft: 10,
250     marginRight: 10,
251     marginTop: 10,
252     marginBottom: 0,
253     color: '#fff',
254     textAlign: 'left'
255   },
256   subtitle_text: {
257     textShadowColor: 'rgba(0, 0, 0, 0.5)',
258     textShadowOffset: {width: 0, height: 1},
259     textShadowRadius: 10,
260     fontSize: 20,
261     flexWrap: 'wrap',
262     marginLeft: 10,
263     marginRight: 10,
264     marginBottom: 0,
265     color: '#ddd',
266     textAlign: 'left'
267   },
```

```
268     description_container: {
269         padding: 5
270     },
271     description_text: {
272         fontStyle: 'italic',
273         color: '#666666',
274         margin: 5,
275         marginBottom: 15,
276         fontSize: 18
277     },
278     glass_container: {
279         flexDirection: 'row',
280         flex: 1,
281         marginBottom: 20
282     },
283     glass_text_left: {
284         marginLeft: 5,
285         marginRight: 5,
286         marginTop: 5,
287         fontSize: 18,
288         textAlign: 'left',
289         flex: 1,
290         color: '#bbb'
291     },
292     glass_text_right: {
293         marginLeft: 5,
294         marginRight: 5,
295         marginTop: 5,
296         fontSize: 18,
297         textAlign: 'right',
298         flex: 1,
299         color: '#bbb'
300     },
301     tag_text: {
302         marginLeft: 5,
303         marginRight: 5,
304         marginTop: 5,
305         textAlign: 'center',
306         fontSize: 18,
307         borderRadius: 20,
308         borderStyle: 'solid',
309         borderColor: 'gray',
310         borderWidth: 1,
311         backgroundColor: '#ddd'
312 }
```

```
313 })
314
315 const mapStateToProps = (state) => {
316     return {
317         favoritesCocktail: state.favoritesCocktail
318     }
319 }
320
321 export default connect(mapStateToProps)(CocktailDetail)
322
```

```
1 import React from 'react'
2 import { StyleSheet, View, Text, Image, TouchableOpacity } from 'react-native'
3
4 class IngredientItem extends React.Component {
5
6     render() {
7         const { ingredient, displayDetailForIngredient } =
8             this.props
9
10        return (
11            <TouchableOpacity style={styles.main_container}
12                onPress={() => displayDetailForIngredient(ingredient.
13 strIngredient1)}>
14                <Image
15                    style={styles.image}
16                    source={{uri: 'https://www.
17 thecocktaildb.com/images/ingredients/' + ingredient.
18 strIngredient1 + '-Medium.png'}}}
19                />
20                <View style={styles.content_container}>
21                    <Text style={styles.title_text}>{
22 strIngredient1}</Text>
23                </View>
24            </TouchableOpacity>
25        )
26    }
27
28    const styles = StyleSheet.create({
29        main_container: {
30            height: 110,
31            flexDirection: 'row',
32            borderRadius: 0,
33            backgroundColor: 'white',
34            marginTop: 5,
35            marginBottom: 5,
36            shadowColor: 'black',
37            shadowOffset: {width: 30, height: 10},
38            shadowOpacity: 0.5,
39            shadowRadius: 20,
40            elevation: 1
41        },
42        image: {
43            width: 120,
44            height: 100,
```

```
39         resizeMode: 'contain',
40         margin: 5,
41     },
42     content_container: {
43         flex: 1,
44         margin: 5
45     },
46     title_text: {
47         textAlignVertical: 'center',
48         textAlign: 'right',
49         fontWeight: 'bold',
50         fontSize: 20,
51         flex: 1,
52         color: '#bbb',
53         flexWrap: 'wrap',
54         paddingRight: 22
55     }
56 })
57
58 export default IngredientItem
59
```

```
1 import React from 'react'
2 import { StyleSheet, FlatList } from 'react-native'
3 import IngredientItem from './IngredientItem'
4 import { connect } from 'react-redux'
5
6 class IngredientList extends React.Component {
7
8     constructor(props) {
9         super(props)
10        this.state = {
11            ingredients: []
12        }
13    }
14
15    _displayDetailForIngredient = (name) => {
16        this.props.navigation.navigate("IngredientDetail",
17        { name: name })
18    }
19
20    render() {
21        return (
22            <FlatList
23                style={styles.list}
24                data={this.props.ingredients}
25                keyExtractor={(item) => item.
26                    strIngredient1}
27                renderItem={({item}) => (
28                    <IngredientItem
29                        ingredient={item}
30                        displayDetailForIngredient={this.
31                            _displayDetailForIngredient}
32                    />
33                ) }
34        )
35    }
36
37    const styles = StyleSheet.create({
38        list: {
39            flex: 1,
40            backgroundColor: '#eee'
41        }
42    })

```

```
43 const mapStateToProps = state => {
44     return {
45         favoritesCocktail: state.favoritesCocktail
46     }
47 }
48
49 export default connect(mapStateToProps)(IngredientList)
50
```

```
1 import React from 'react'
2 import { StyleSheet, View, TouchableOpacity, Text,
3 ImageBackground } from 'react-native'
4 import Category from './Category'
5
6 class SortedCategory extends React.Component {
7
8     constructor(props) {
9         super(props)
10        this.state = {
11            cocktails: []
12        }
13
14        _navigate(category) {
15            this.props.navigation.navigate("Category", {
16                cocktails:this.state.cocktails, navigation:this.props.
17                navigation, category:category})
18
19        render() {
20            return (
21                <View style={styles.main_container}>
22                    <View style={styles.area_container_top}>
23                        <TouchableOpacity style={styles.
24                            backgroundContainer} onPress={() => this._navigate("25
26                                Ordinary Drink")}>
27                                    <ImageBackground
28                                        source={require('../assets/
29                                ordinary_drink.jpg')}
30                                    resizeMode='cover'
31                                    imageStyle={{borderRadius: 20}}
32
33                                    style={styles.backgroundImage}
34
35                                    >
36
37                                <Text style={styles.
38                                    default_text}>Ordinary Drink</Text>
39
40                                    </ImageBackground>
41
42                                    </TouchableOpacity>
43
44                                    <TouchableOpacity style={styles.
45                            backgroundContainer} onPress={() => this._navigate("46
47                                Cocktail")}>
48
49                                    <ImageBackground
50                                        source={require('../assets/
```

```
35 cocktail.jpg' ) }  
36                     resizeMode='cover'  
37                     imageStyle={ {borderRadius: 20 }  
38 }  
39                     style={ styles.backgroundImage }  
40                 >  
41                     <Text style={ styles.  
42             default_text }>Cocktail</Text>  
43                     </ImageBackground>  
44                     </TouchableOpacity>  
45                 </View>  
46                 <View style={ styles.area_container_middle }  
47             >  
48                     <TouchableOpacity style={ styles.  
49             backgroundContainer } onPress={ () => this._navigate("Milk  
50             / Float / Shake") }>  
51                     <ImageBackground  
52                         source={ require('../assets/  
53             milkshake.jpg') }  
54                         resizeMode='cover'  
55                         imageStyle={ {borderRadius: 20 }  
56 }  
57                         style={ styles.backgroundImage }  
58                     >  
59                     <Text style={ styles.  
60             default_text }>Milk Shake</Text>  
61                     </ImageBackground>  
62                     </TouchableOpacity>  
63                     <TouchableOpacity style={ styles.  
64             backgroundContainer } onPress={ () => this._navigate("Cocoa"  
65             ) }>  
66                     <ImageBackground  
67                         source={ require('../assets/  
68             cocoa.jpg') }  
69                         resizeMode='cover'  
70                         imageStyle={ {borderRadius: 20 }  
71 }  
72                         style={ styles.backgroundImage }  
73                     >  
74                     <Text style={ styles.  
75             default_text }>Cocoa</Text>  
76                     </ImageBackground>  
77                     </TouchableOpacity>  
78                 </View>  
79                 <View style={ styles.area_container_middle }  
80             >
```

```
66 >
67             <TouchableOpacity style={styles.
68     backgroundContainer} onPress={() => this._navigate("Shot"
69 )}>
70             <ImageBackground
71                 source={require('../assets/
72     shot.jpg')}
73             resizeMode='cover'
74             imageStyle={{borderRadius: 20
75 }}>
76             <Text style={styles.
77     default_text}>Shot</Text>
78         </ImageBackground>
79         </TouchableOpacity>
80         <TouchableOpacity style={styles.
81     backgroundContainer} onPress={() => this._navigate("
82     Coffee / Tea")}>
83             <ImageBackground
84                 source={require('../assets/
85     coffee_tea.jpg')}
86             resizeMode='cover'
87             imageStyle={{borderRadius: 20
88 }}>
89             <Text style={styles.
90     default_text}>Coffee / Tea</Text>
91         </ImageBackground>
92         </TouchableOpacity>
93     </View>
94     <View style={styles.area_container_middle
95 }>
96         <TouchableOpacity style={styles.
97     backgroundContainer} onPress={() => this._navigate("Beer"
98 )}>
99             <ImageBackground
100                source={require('../assets/
101     beer.jpg')}
102                resizeMode='cover'
103                imageStyle={{borderRadius: 20
104 }}>
105         </View>
106     </View>
107 </View>
```

```
94                      style={styles.backgroundImage}
95                  }
96              >
97              <Text style={styles.
98      default_text}>Beer</Text>
99              </ImageBackground>
100             </TouchableOpacity>
101             <TouchableOpacity style={styles.
102     backgroundContainer} onPress={() => this._navigate("Soft
103     Drink / Soda")}>
104                 <ImageBackground
105                     source={require('../assets/
106     soft_drink.jpg')} }
107                     resizeMode='cover'
108                     imageStyle={{borderRadius: 20
109     }}}
110                 style={styles.backgroundImage
111     }
112             >
113             <Text style={styles.
114     default_text}>Soft Drink / Soda</Text>
115             </ImageBackground>
116             </TouchableOpacity>
117         </View>
118         <View style={styles.area_container_bottom
119     }>
120             <TouchableOpacity style={styles.
121     backgroundContainer} onPress={() => this._navigate("Homemade
122     Liqueur")}>
123                 <ImageBackground
124                     source={require('../assets/
125     homemade.jpg')} }
126                     resizeMode='cover'
127                     imageStyle={{borderRadius: 20
128     }}}
129                 style={styles.backgroundImage
130     }
131             >
132             <Text style={styles.
133     default_text}>Homemade Liqueur</Text>
134             </ImageBackground>
135             </TouchableOpacity>
136             <TouchableOpacity style={styles.
137     backgroundContainer} onPress={() => this._navigate("Punch
138     / Party Drink")}>
```

```
122                     <ImageBackground
123                         source={require('../assets/
124                           punch.jpg')} }
125
126                         resizeMode='cover'
127                         imageStyle={{borderRadius: 20
128                           }
129                         style={styles.backgroundImage
130                           }
131                         >
132                             <Text style={styles.
133                               default_text}>Punch / Party Drink</Text>
134                             </ImageBackground>
135                             </TouchableOpacity>
136                         </View>
137                     </View>
138                 )
139             }
140         )
141     }
142
143     const styles = StyleSheet.create({
144       main_container: {
145         flex: 1
146       },
147       area_container_top: {
148         flex: 1,
149         marginTop: 10,
150         marginLeft: 10,
151         marginRight: 10,
152         flexDirection: 'row'
153       },
154       area_container_middle: {
155         flex: 1,
156         marginLeft: 10,
157         marginRight: 10,
158         flexDirection: 'row'
159       },
160       area_container_bottom: {
161         flex: 1,
162         marginBottom: 10,
163         marginLeft: 10,
164         marginRight: 10,
165         flexDirection: 'row'
166       },
167       default_text: {
168         textShadowColor: 'rgba(0, 0, 0, 0.5)',
```

```
163         textShadowOffset: {width: 0, height: 1},  
164         textShadowRadius: 10,  
165         fontSize: 20,  
166         textAlign: 'center',  
167         color: 'white'  
168     },  
169     backgroundContainer: {  
170         margin: 10,  
171         flex:1,  
172         justifyContent: 'center',  
173         borderRadius: 20,  
174     },  
175     backgroundImage: {  
176         margin: 0,  
177         flex:1,  
178         justifyContent: 'center',  
179         borderRadius: 20,  
180     }  
181 })  
182  
183 export default SortedCategory  
184
```

```
1 import React from 'react'
2 import { StyleSheet, View, TouchableOpacity, Text,
3         ImageBackground } from 'react-native'
4
5 class SortedCocktail extends React.Component {
6
7     constructor(props) {
8         super(props)
9         this.state = {
10             cocktails: []
11         }
12     }
13
14     _navigate(alcool) {
15         this.props.navigation.navigate("Alcool", {
16             cocktails:this.state.cocktails, navigation:this.props.
17             navigation, alcool:alcool})
18     }
19
20     render() {
21         return (
22             <View style={styles.main_container}>
23                 <View style={styles.area_container_top}>
24                     <TouchableOpacity style={styles.
25                         backgroundContainer} onPress={() => this._navigate("vodka"
26                         )}>
27                         <ImageBackground
28                             source={require('../assets/vodka.
29                             jpg')}
30                             resizeMode='cover'
31                             imageStyle={{borderRadius: 20}}
32                             style={styles.backgroundImage}
33                         >
34                             <Text style={styles.default_text}>
35                             Vodka</Text>
36                         </ImageBackground>
37                     </TouchableOpacity>
38                     <TouchableOpacity style={styles.
39                         backgroundContainer} onPress={() => this._navigate("rum") }
40                         >
41                             <ImageBackground
42                                 source={require('../assets/rum.jpg
43                                 ')}
```

```
36                     resizeMode='cover'
37                     imageStyle={{borderRadius: 20}}
38                     style={styles.backgroundImage}
39                 >
40                     <Text style={styles.
41             default_text}>Rum</Text>
42                     </ImageBackground>
43                     </TouchableOpacity>
44                 </View>
45                 <View style={styles.area_container_middle}>
46                     <TouchableOpacity style={styles.
47             backgroundContainer} onPress={() => this._navigate("tequila")}>
48                         <ImageBackground
49                             source={require('../assets/tequila.
50                             .jpg')} >
51                             resizeMode='cover'
52                             imageStyle={{borderRadius: 20}}
53                             style={styles.backgroundImage}
54                         >
55                         <Text style={styles.default_text}>
56                             Tequila</Text>
57                         </ImageBackground>
58                         </TouchableOpacity>
59                     <TouchableOpacity style={styles.
60             backgroundContainer} onPress={() => this._navigate("gin")}>
61                         <ImageBackground
62                             source={require('../assets/gin.jpg
63                             ') >
64                             resizeMode='cover'
65                             imageStyle={{borderRadius: 20}}
66                             style={styles.backgroundImage}
67                         >
68                         <Text style={styles.default_text}>
69                             Gin</Text>
70                         </ImageBackground>
71                         </TouchableOpacity>
72                     </View>
73                     <View style={styles.area_container_bottom}>
74                         <TouchableOpacity style={styles.
75             backgroundContainer} onPress={() => this._navigate("triple
76             sec")}>
77                             <ImageBackground
78                             source={require('../assets/
79                             .jpg')} >
```

```
69 triple_sec.jpg' ) }  
70                     resizeMode='cover'  
71                     imageStyle={{borderRadius: 20}}  
72                     style={styles.backgroundImage}  
73                 >  
74             <Text style={styles.default_text}  
75             >Triple sec</Text>  
76         </ImageBackground>  
77     </TouchableOpacity>  
78     <TouchableOpacity style={styles.  
79     backgroundContainer} onPress={() => this._navigate("whiskey")}>  
80         <ImageBackground  
81             source={require('../assets/  
82             whiskey.jpg')}  
83             resizeMode='cover'  
84             imageStyle={{borderRadius: 20}}  
85             style={styles.backgroundImage}  
86         >  
87         <Text style={styles.default_text}  
88         >Whiskey</Text>  
89     </ImageBackground>  
90     </TouchableOpacity>  
91     </View>  
92     </View>  
93     )  
94   }  
95 }  
96  
97 const styles = StyleSheet.create({  
98     main_container: {  
99         flex: 1  
100    },  
101    backgroundContainer: {  
102        margin: 10,  
103        flex:1,  
104        justifyContent: 'center',  
105        borderRadius: 20,  
106    },  
107    backgroundImage: {  
108        margin: 0,  
109        flex:1,  
110        justifyContent: 'center',  
111        borderRadius: 20,  
112    },
```

```
109     area_container_top: {
110         flex: 1,
111         marginTop: 10,
112         marginLeft: 10,
113         marginRight: 10,
114         flexDirection: 'row'
115     },
116     area_container_middle: {
117         flex: 1,
118         marginLeft: 10,
119         marginRight: 10,
120         flexDirection: 'row'
121     },
122     area_container_bottom: {
123         flex: 1,
124         marginBottom: 10,
125         marginLeft: 10,
126         marginRight: 10,
127         flexDirection: 'row'
128     },
129     default_text: {
130         textShadowColor: 'rgba(0, 0, 0, 0.5)',
131         textShadowOffset: {width: 0, height: 1},
132         textShadowRadius: 10,
133         fontSize: 20,
134         textAlign: 'center',
135         color: 'white'
136     }
137 })
138
139 export default SortedCocktail
140
```

```
1 import React from 'react'
2 import { StyleSheet, View, Text, Image, ActivityIndicator,
3         ScrollView } from 'react-native'
4 import { getIngredientDetailFromApi } from '../API/TCDBApi'
5
6 class IngredientDetail extends React.Component {
7     constructor(props) {
8         super(props)
9         this.state = {
10             ingredient: [],
11             isLoading: true
12         }
13     _displayLoading() {
14         if (this.state.isLoading) {
15             return (
16                 <View style={styles.loading_container}>
17                     <ActivityIndicator size='large' />
18                 </View>
19             )
20         }
21     }
22     _updateNavigationParams() {
23         this.props.navigation.setParams({
24             ingredient: this.state.ingredient
25         })
26     }
27     componentDidMount() {
28         this.setState({ isLoading: true })
29         getIngredientDetailFromApi(this.props.navigation.
state.params.name).then(data => {
30             this.setState({
31                 ingredient: data.ingredients[0],
32                 isLoading: false
33             }, () => { this._updateNavigationParams() })
34         })
35     }
36
37     render() {
38         const { ingredient } = this.state
39         return (
40             <View style={styles.main_container}>
41                 {this._displayLoading()}
42                 <View style={styles.header}>
```

```
43             <Text style={styles.title_text}>{
44                 ingredient.strIngredient}</Text>
45             <Image
46                 style={styles.image}
47                 source={{uri: 'https://www.
48                 thecocktaildb.com/images/ingredients/' + ingredient.
49                 strIngredient + '.png'}}}
50             />
51         </View>
52         <ScrollView style={styles.scrollview}>
53             <Text style={styles.description_text}>
54                 {ingredient.strDescription}</Text>
55             </ScrollView>
56         </View>
57     )
58 }
59
60 const styles = StyleSheet.create({
61     main_container: {
62         flex: 1
63     },
64     loading_container: {
65         position: 'absolute',
66         left: 0,
67         right: 0,
68         top: 0,
69         bottom: 0,
70         alignItems: 'center',
71         justifyContent: 'center'
72     },
73     image: {
74         marginTop: 5,
75         marginBottom: 5,
76         flex: 1,
77         resizeMode: 'contain'
78     },
79     title_text: {
80         fontSize: 35,
81         flexWrap: 'wrap',
82         marginLeft: 5,
83         marginRight: 5,
84         marginTop: 10,
85         marginBottom: 0,
```

```
84         color: '#000000',
85         textAlign: 'left'
86     },
87     description_text: {
88         fontStyle: 'italic',
89         color: '#666666',
90         margin: 5,
91         marginBottom: 15,
92         flex: 2
93     },
94     scrollview: {
95         flex: 1
96     },
97     header: {
98         flex: 2
99     }
100 })
101
102
103 export default IngredientDetail
104
```

```
1 import React from 'react'
2 import { StyleSheet, View, Text, Image, ActivityIndicator,
3   ScrollView, FlatList, TouchableOpacity, Share } from 'react-native'
4 import { getCocktailDetailFromApi } from '../API/TCDBApi'
5 import { connect } from 'react-redux'
6
7 class IngredientMeasure extends React.Component {
8
9   render() {
10     const { ingredient, measure, navigation } = this.props
11     return (
12       <TouchableOpacity
13         style={styles.ingredient_container}
14         onPress={() => navigation.navigate("IngredientDetail", { name: ingredient })}>
15         <Image
16           style={styles.image_ingredient}
17           source={{uri: 'https://www.
18             thecocktaildb.com/images/ingredients/' + ingredient + '-Small.
19             png'}}>
20         />
21         <Text style={styles.ingredient_text}>{ingredient}</Text>
22         <Text style={styles.measure_text}>{measure}</Text>
23       </TouchableOpacity>
24     )
25   }
26
27   const styles = StyleSheet.create({
28     ingredient_container: {
29       backgroundColor: '#eee',
30       flexDirection: 'row',
31       marginTop: 5,
32       marginBottom: 5,
33       marginLeft: 10,
34       marginRight: 10,
35       borderRadius: 5
36     },
37     ingredient_text: {
38       fontSize: 18,
```

```
38         textAlign: 'left',
39         textAlignVertical: 'center',
40         paddingTop: 5,
41         paddingBottom: 5,
42         paddingLeft: 5,
43         flex: 2,
44     },
45     measure_text: {
46         fontSize: 18,
47         color: '#bbb',
48         textAlign: 'right',
49         paddingBottom: 5,
50         paddingTop: 5,
51         textAlignVertical: 'center',
52         paddingRight: 10,
53         flex: 2
54     },
55     image_ingredient: {
56         height: 60,
57         paddingLeft: 5,
58         marginBottom: 5,
59         marginTop: 5,
60         resizeMode: 'contain',
61         flex: 1
62     }
63 })
64 export default IngredientMeasure
65
```

```
1 import React from 'react';
2 import Navigation from './Navigation/Navigation'
3 import Store from './Store/configureStore'
4 import { Provider } from 'react-redux'
5 import { persistStore } from 'redux-persist'
6 import { PersistGate } from 'redux-persist/es/integration/react'
7
8 export default class App extends React.Component {
9     render() {
10         let persistor = persistStore(Store)
11         return (
12             <Provider store={Store}>
13                 <PersistGate persistor={persistor}>
14                     <Navigation />
15                 </PersistGate>
16             </Provider>
17         )
18     }
19 }
20
```

```
1 import React from 'react'
2 import { StyleSheet, Image } from 'react-native'
3 import { createStackNavigator, createAppContainer,
4         createBottomTabNavigator } from 'react-navigation'
5 import Search from '../Components/Search'
6 import CocktailDetail from '../Components/CocktailDetail'
7 import Favorites from '../Components/Favorites'
8 import IngredientDetail from '../Components/
9           IngredientDetail'
10 import News from '../Components/News'
11 import Ingredient from '../Components/Ingredient'
12 import Alcool from '../Components/Alcool'
13 import SortedCocktail from '../Components/SortedCocktail'
14 import SortedCategory from '../Components/SortedCategory'
15 import Category from '../Components/Category'
16
17 const SearchStackNavigator = createStackNavigator({
18     Search: {
19         screen: Search,
20         navigationOptions: {
21             title: 'Home'// /!\attention
22         }
23     },
24     CocktailDetail: {
25         screen: CocktailDetail
26     },
27     IngredientDetail: {
28         screen: IngredientDetail,
29     }
30 })
31
32 const FavoritesStackNavigator = createStackNavigator({
33     Favorites: {
34         screen: Favorites,
35         navigationOptions: {
36             title: 'Favorites'
37         }
38     },
39     CocktailDetail: {
40         screen: CocktailDetail
41     },
42     IngredientDetail: {
43         screen: IngredientDetail,
44     }
45 })
```

```
44
45 const AlcoolStackNavigator = createStackNavigator({
46     SortedCocktail: {
47         screen: SortedCocktail,
48         navigationOptions: {
49             title: 'Spirits',
50         },
51     },
52     Alcool: {
53         screen: Alcool,
54     },
55     CocktailDetail: {
56         screen: CocktailDetail
57     },
58     IngredientDetail: {
59         screen: IngredientDetail,
60     }
61 })
62 const SortedCategoryStackNavigator = createStackNavigator(
63 {
64     SortedCategory: {
65         screen: SortedCategory,
66         navigationOptions: {
67             title: 'Category',
68         },
69     },
70     Category: {
71         screen: Category,
72     },
73     CocktailDetail: {
74         screen: CocktailDetail
75     },
76     IngredientDetail: {
77         screen: IngredientDetail,
78     }
79 })
80 const IngredientStackNavigator = createStackNavigator({
81     Ingredient: {
82         screen: Ingredient,
83         navigationOptions: {
84             title: 'Ingredients',
85         },
86     },
87     IngredientDetail: {
```

```
88         screen: IngredientDetail,
89     }
90   })
91
92 const CocktailTabNavigator = createBottomTabNavigator({
93   SortedCategory: {
94     screen: SortedCategoryStackNavigator,
95     navigationOptions: {
96       tabBarIcon: () => {
97         return <Image
98           source={require('../assets/
99             ic_cocktail.png')} }
100          style={styles.icon}/>
101        }
102      },
103      SortedCocktail: {
104        screen: AlcoolStackNavigator,
105        navigationOptions: {
106          tabBarIcon: () => {
107            return <Image
108              source={require('../assets/
109                ic_spirits.png')} }
110              style={styles.icon}/>
111            }
112          },
113          Search: {
114            screen: SearchStackNavigator,
115            navigationOptions: {
116              tabBarIcon: () => { // Icon render
117                return <Image
118                  source={require('../assets/
119                    ic_home.png')} }
120                  style={styles.icon} // Icon
121                  style
122                />
123              }
124            },
125            Favorites: {
126              screen: FavoritesStackNavigator,
127              navigationOptions: {
128                tabBarIcon: () => {
129                  return <Image
```

```
129                     source={require('../assets/
  ic_favorite_dark.png') }
130                     style={styles.icon}
131                 />
132             }
133         }
134     },
135     Ingredient: {
136         screen: IngredientStackNavigator,
137         navigationOptions: {
138             tabBarIcon: () => {
139                 return <Image
140                     source={require('../assets/
  ic_book.png') }
141                     style={styles.icon}
142                 />
143             }
144         }
145     },
146 },
147 {
148     initialRouteName: 'Search',
149     tabBarOptions: {
150         activeBackgroundColor: '#DDDDDD', // // active tab background color
151         inactiveBackgroundColor: '#FFFFFF',
152         showLabel: false, // hide titles
153         showIcon: true // show icons
154     }
155 }
156 )
157
158 const styles = StyleSheet.create({
159     icon: {
160         width: 30,
161         height: 30
162     }
163 })
164
165 export default createAppContainer(
  CocktailTabNavigator)
166
```

```
1 import React from 'react'
2 import { Animated, Dimensions } from 'react-native'
3
4 class FadeIn extends React.Component {
5
6     constructor(props) {
7         super(props)
8         this.state = {
9             positionLeft: new Animated.Value(Dimensions.
get('window').width)
10        }
11    }
12
13    componentDidMount() {
14        Animated.spring(
15            this.state.positionLeft,
16            {
17                toValue: 0
18            }
19        ).start()
20    }
21
22    render() {
23        return (
24            <Animated.View
25                style={{ left: this.state.positionLeft }}>
26                {this.props.children}
27            </Animated.View>
28        )
29    }
30 }
31
32 export default FadeIn
33
```

```
1 import { createStore } from 'redux';
2 import toggleFavorite from './Reducers/favoriteReducer'
3 import { persistReducer } from 'redux-persist'
4 import storage from 'redux-persist/lib/storage'
5
6 const rootPersistConfig = {
7   key: 'root',
8   storage: storage
9 }
10
11 export default createStore(persistReducer(
  rootPersistConfig, toggleFavorite))
12
```

```
1 const initialState = { favoritesCocktail: [] }
2
3 function toggleFavorite(state = initialState, action) {
4     let nextState
5     switch (action.type) {
6         case 'TOGGLE_FAVORITE':
7             const favoritesCocktailIndex = state.
8                 favoritesCocktail.findIndex(item => item.idDrink ===
9                     action.value.idDrink)
10            if (favoritesCocktailIndex !== -1) {
11                // cocktail already in favorite, remove it
12                nextState = {
13                    ...state,
14                    favoritesCocktail: state.favoritesCocktail
15                    .filter( (item, index) => index !== favoritesCocktailIndex
16                )
17            }
18            else {
19                // cocktail not in favorite, add it
20                nextState = {
21                    ...state,
22                    favoritesCocktail: [...state.
23                        favoritesCocktail, action.value]
24                }
25            }
26        }
27
28 export default toggleFavorite
29
```

```
1 const API_TOKEN = '1'
2
3 // Cocktail methods
4 export function getCocktailFromApiWithSearchedText(text) {
5     const url = 'https://www.thecocktailedb.com/api/json/v1
6         /' + API_TOKEN + '/search.php?s=' + text
7     return fetch(url)
8     .then((response) => response.json())
9     .catch((error) => console.error(error))
10}
11
12 export function getCocktailDetailFromApi (id) {
13     return fetch('https://www.thecocktailedb.com/api/json/
14         v1/' + API_TOKEN + '/lookup.php?i=' + id)
15     .then((response) => response.json())
16     .catch((error) => console.error(error));
17}
18
19 export function getRandomCocktailFromApi () {
20     return fetch('https://www.thecocktailedb.com/api/json/
21         v1/' + API_TOKEN + '/random.php')
22     .then((response) => response.json())
23     .catch((error) => console.error(error));
24}
25
26 export function getCocktailByIngredientFromApi (ingredient
27 ) {
28     return fetch('https://www.thecocktailedb.com/api/json/
29         v1/' + API_TOKEN + '/filter.php?i=' + ingredient)
30     .then((response) => response.json())
31     .catch((error) => console.error(error));
32}
33
34 export function getCocktailByCategoryFromApi (category) {
35     return fetch('https://www.thecocktailedb.com/api/json/
36         v1/' + API_TOKEN + '/filter.php?c=' + category)
37     .then((response) => response.json())
38     .catch((error) => console.error(error));
39}
40
41 // Ingredient methods
42
43 export function getIngredientListFromApi () {
44     return fetch('https://www.thecocktailedb.com/api/json/
45         v1/' + API_TOKEN + '/list.php?i=list')
46     .then((response) => response.json())
47     .catch((error) => console.error(error));
48}
```

```
39 export function getIngredientDetailFromApi (name) {  
40     return fetch('https://www.thecocktaildb.com/api/json/  
    v1/' + API_TOKEN + '/search.php?i=' + name)  
41     .then((response) => response.json())  
42     .catch((error) => console.error(error));  
43 }  
44
```