Coursework Report

Mobile application development – set08114

eloi filaudeau

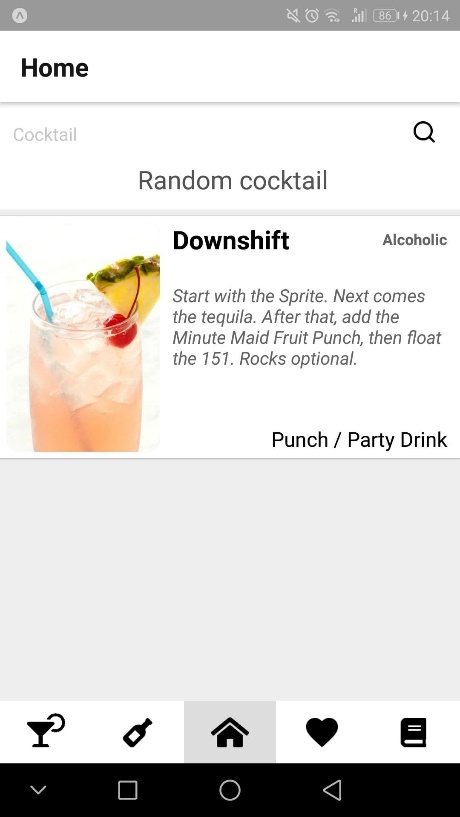
2019

# 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.

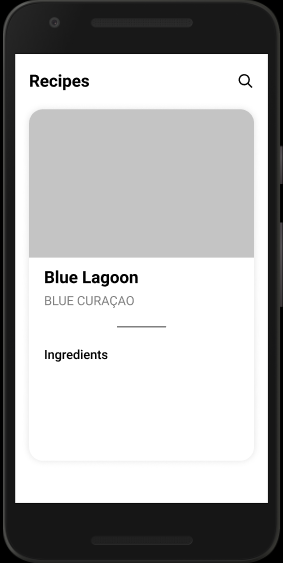
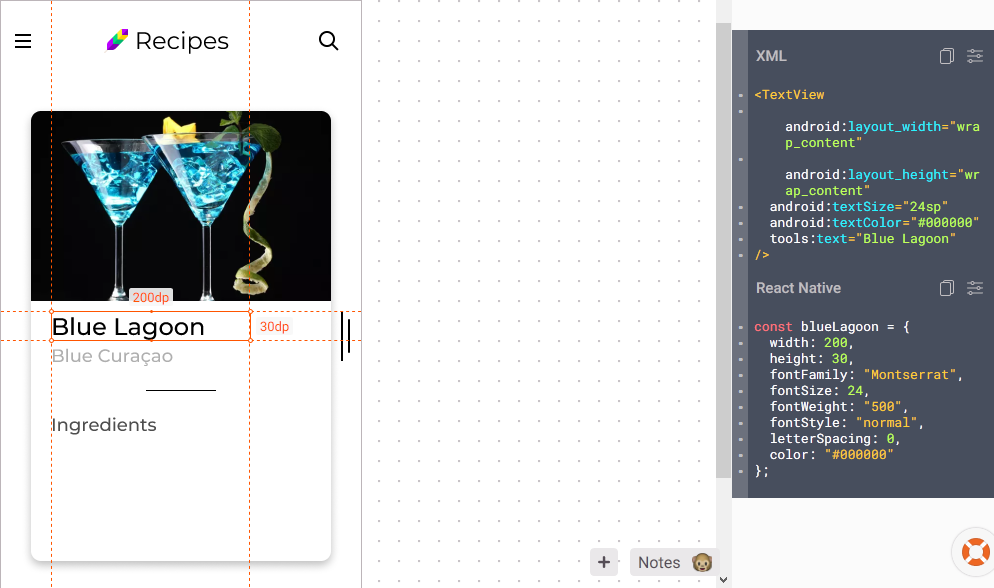
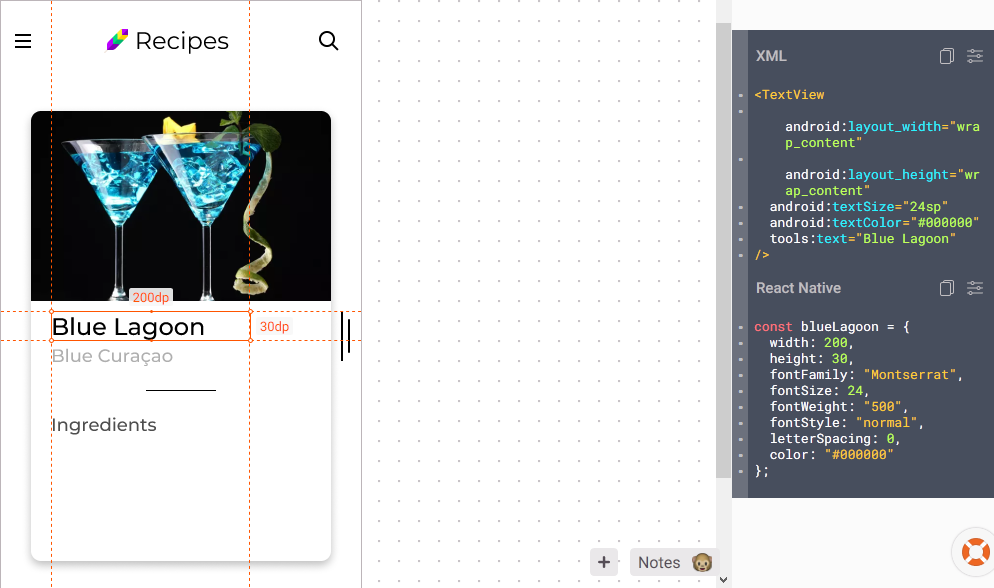


There are a multitude of solutions to develop a mobile application, I decided to use **React Native**. It is an open source and free solution developed by *Facebook* in 2015. There are two main reasons for my choice. First, I already know **Android Studio** because I have done several projects with it (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.



2

1

*1 – Figma mockup view CocktailDetail*

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

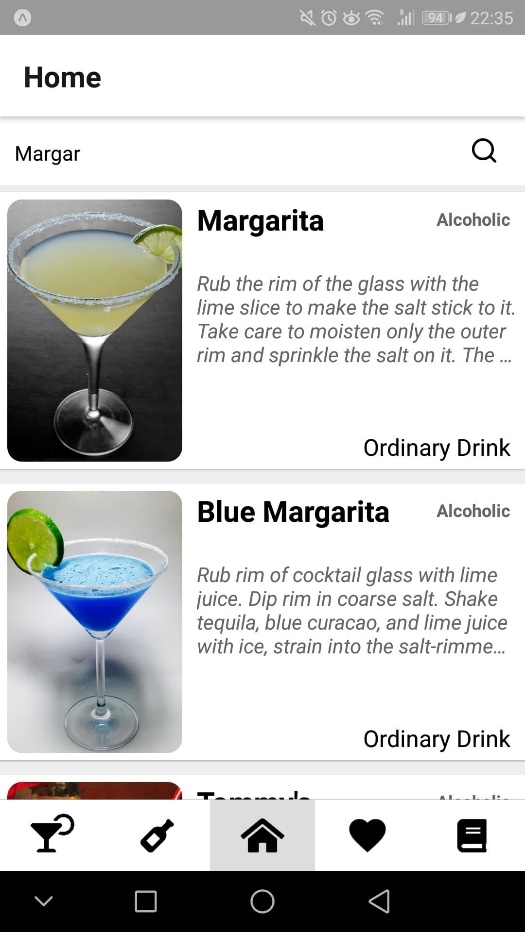
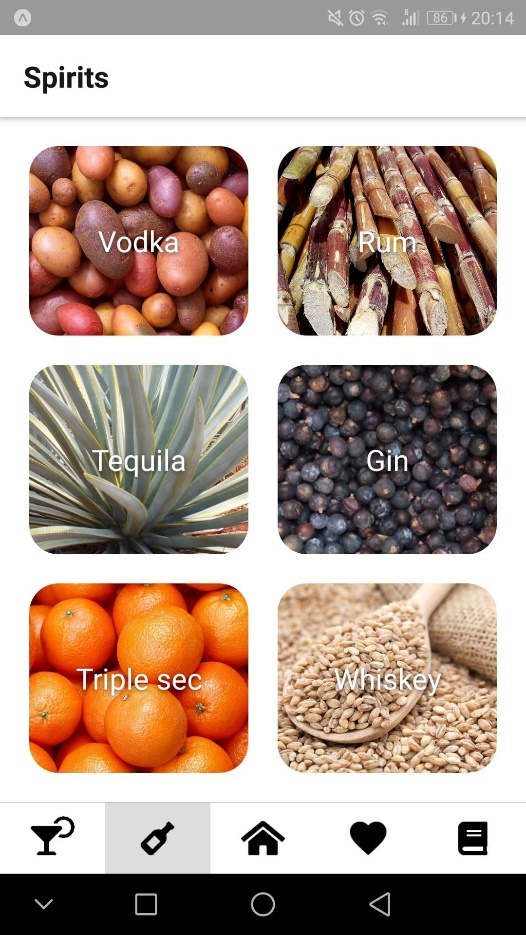
# Application functionalities

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.

CocktailList

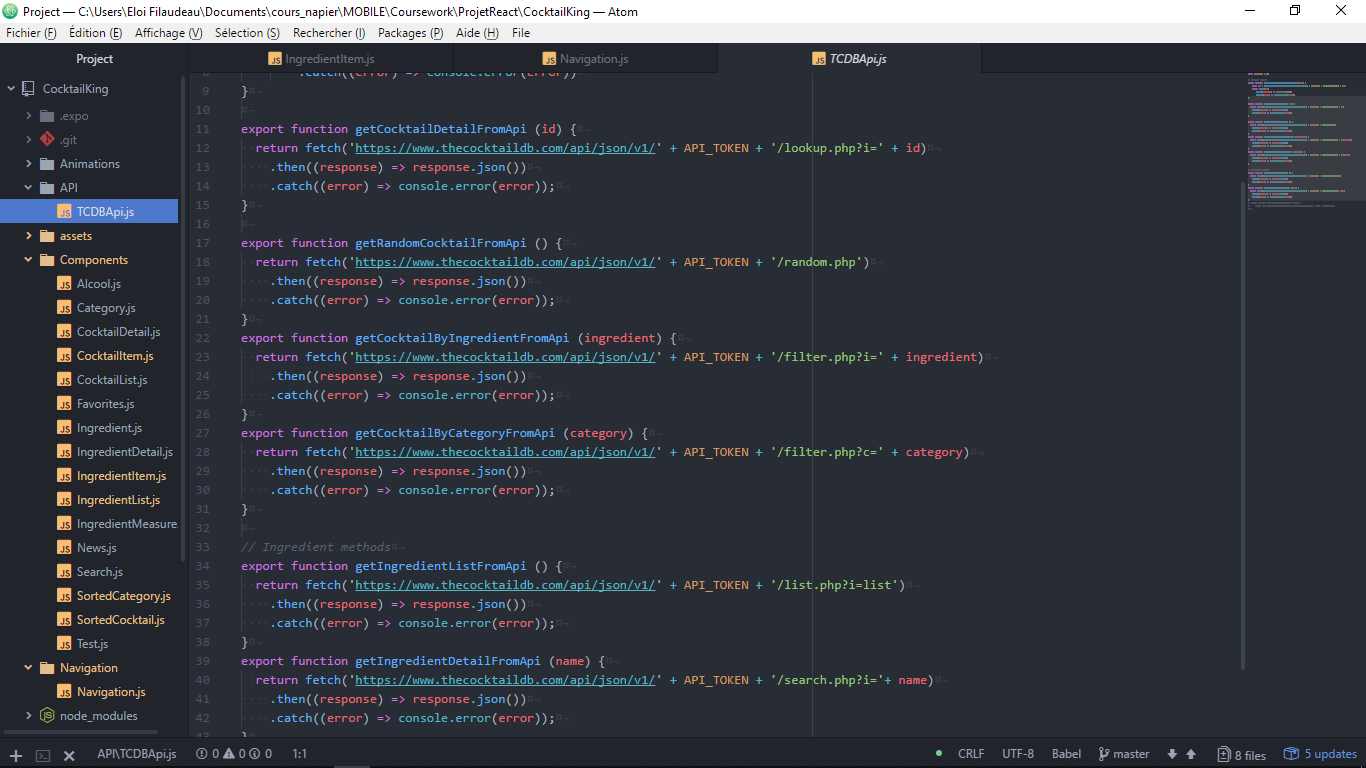
CocktailDetail

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.

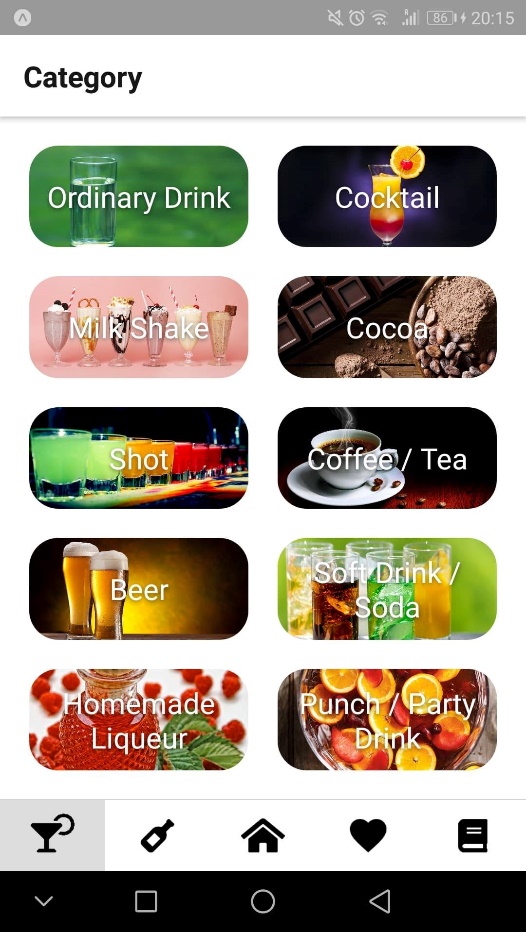
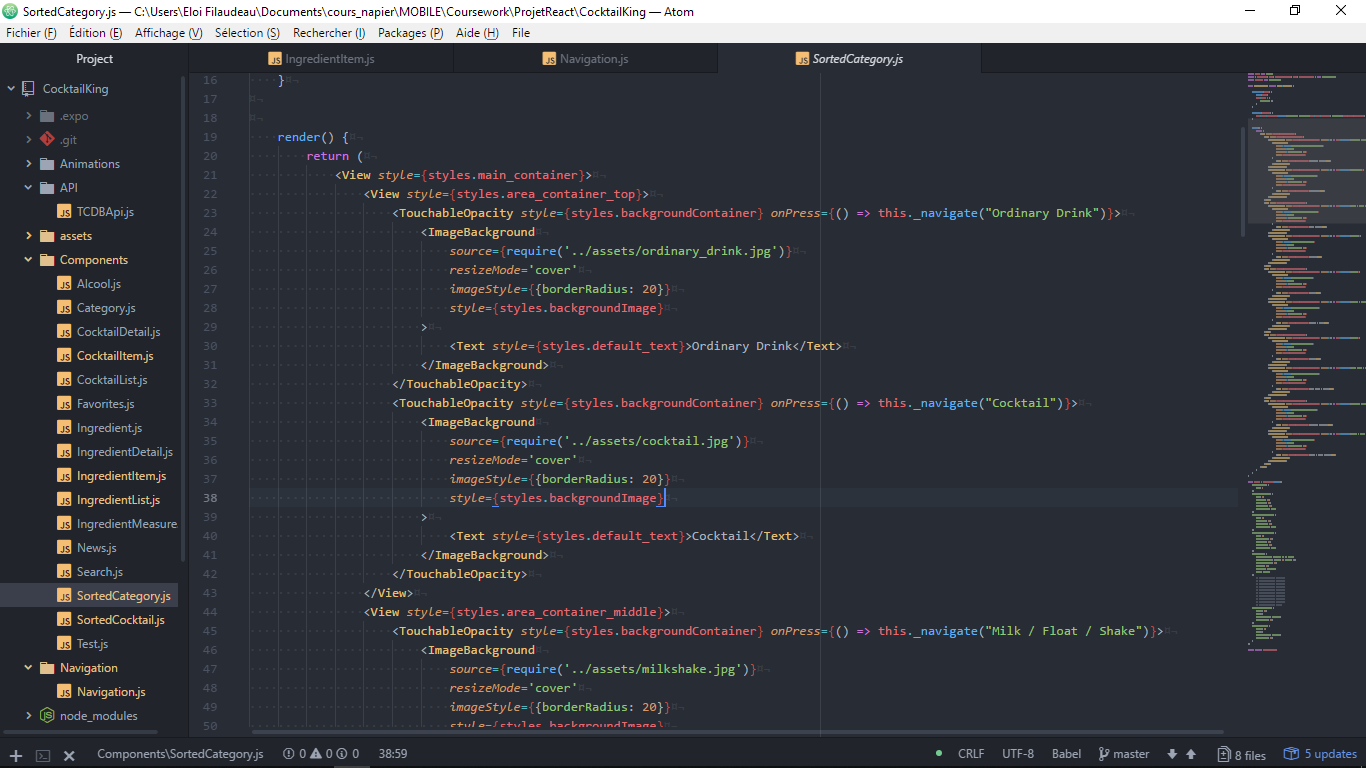
 

v

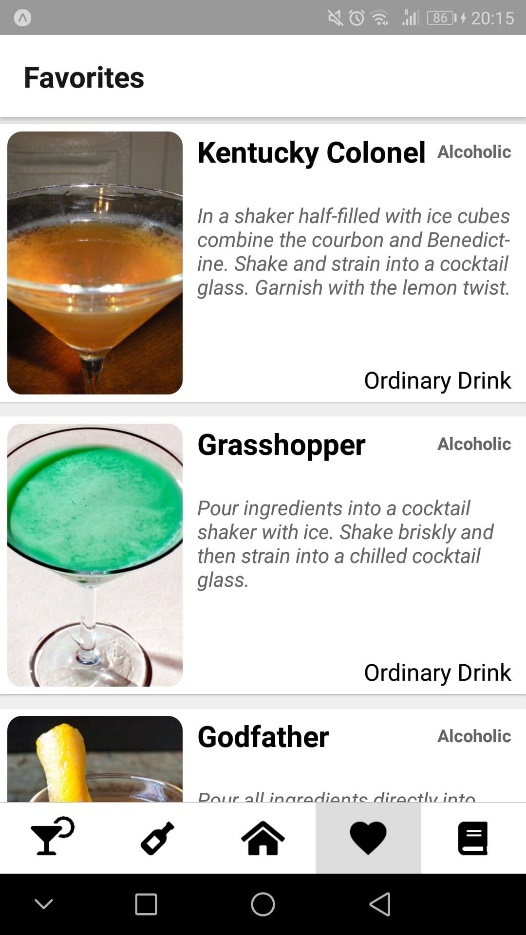
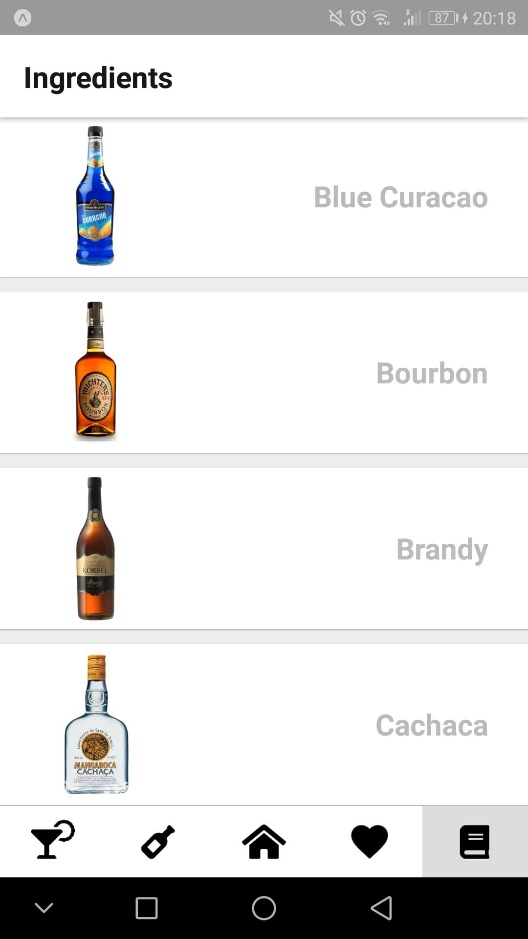
Spirits page: this page allows you to search for cocktails by base spirits, there is a choice between Vodka, Rum, Tequila, Gin, Triple sec and Whiskey. To display the search cocktails, I make an API call (see below) with the method to get cocktails according to an ingredient (which is here the base spirit)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



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).



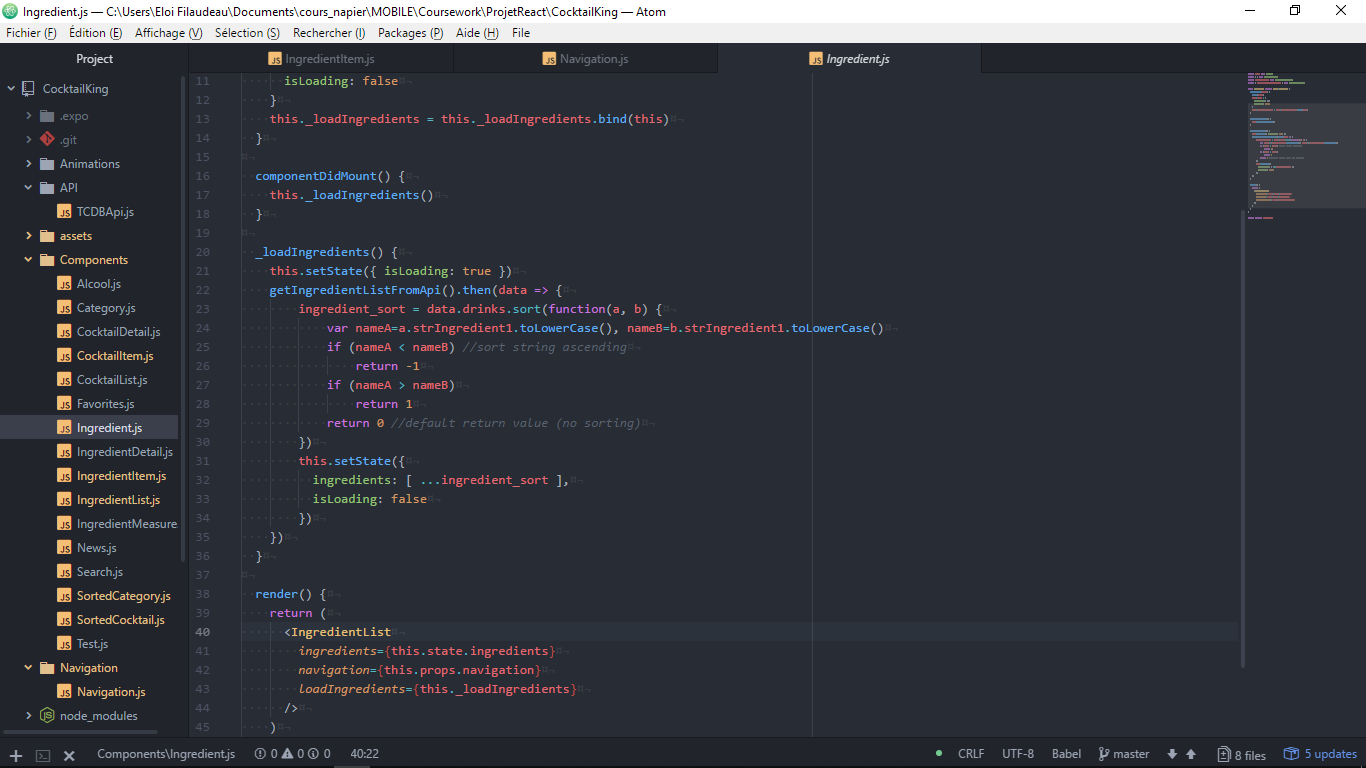
Favorites page: this page retrieves in the global state of the redux store the table of favorite cocktails to display them in a *CocktailList*. Favorite cocktails remain in memory even when the application is closed. To add a cocktail as a favorite, simply touch the heart icon that is linked to a global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

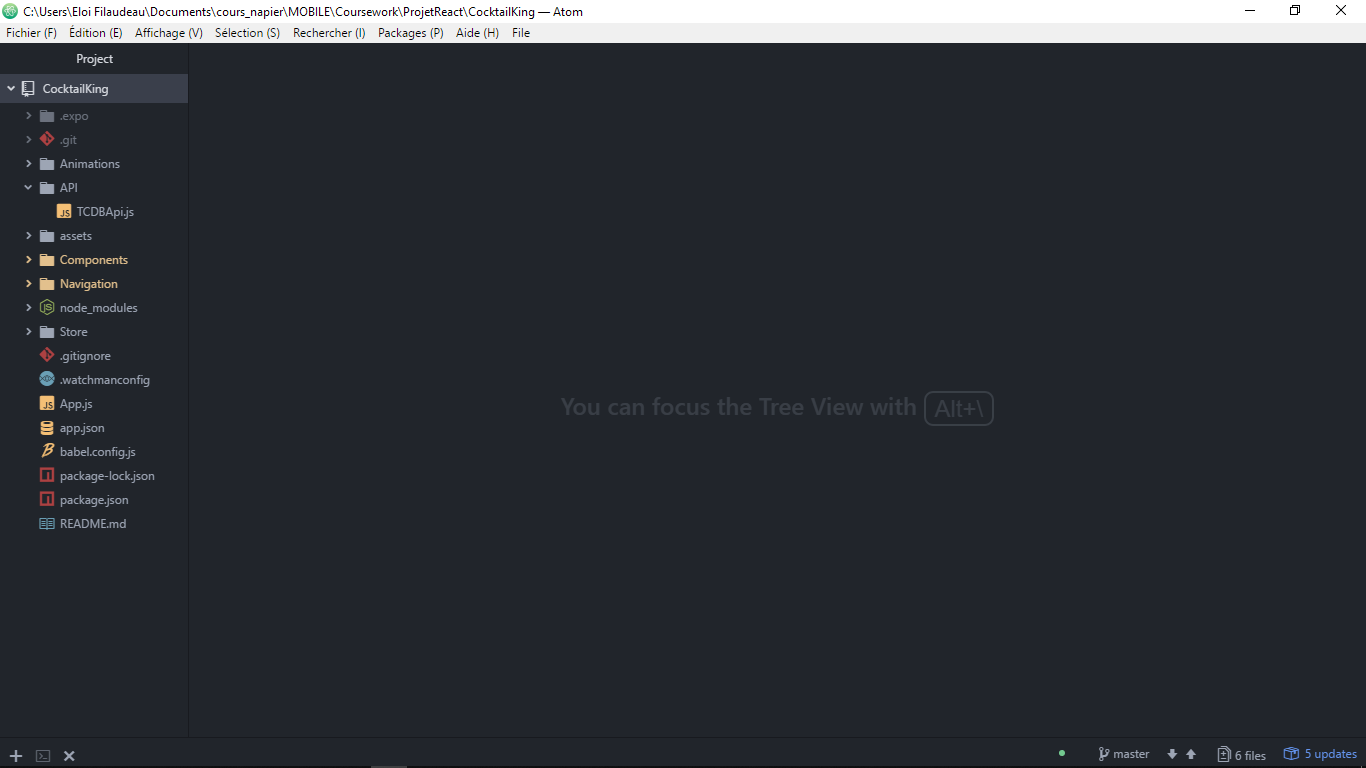
v

v

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.



*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

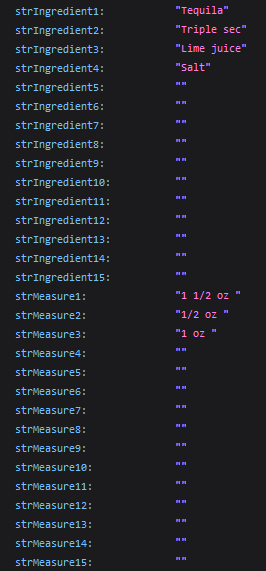
At first, I wanted to make a cocktail recipe application to make cocktails and keep our favourites in mind. So I have largely implemented the application I was expecting, which is why I am quite satisfied with my work.

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.



*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-pla>

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>

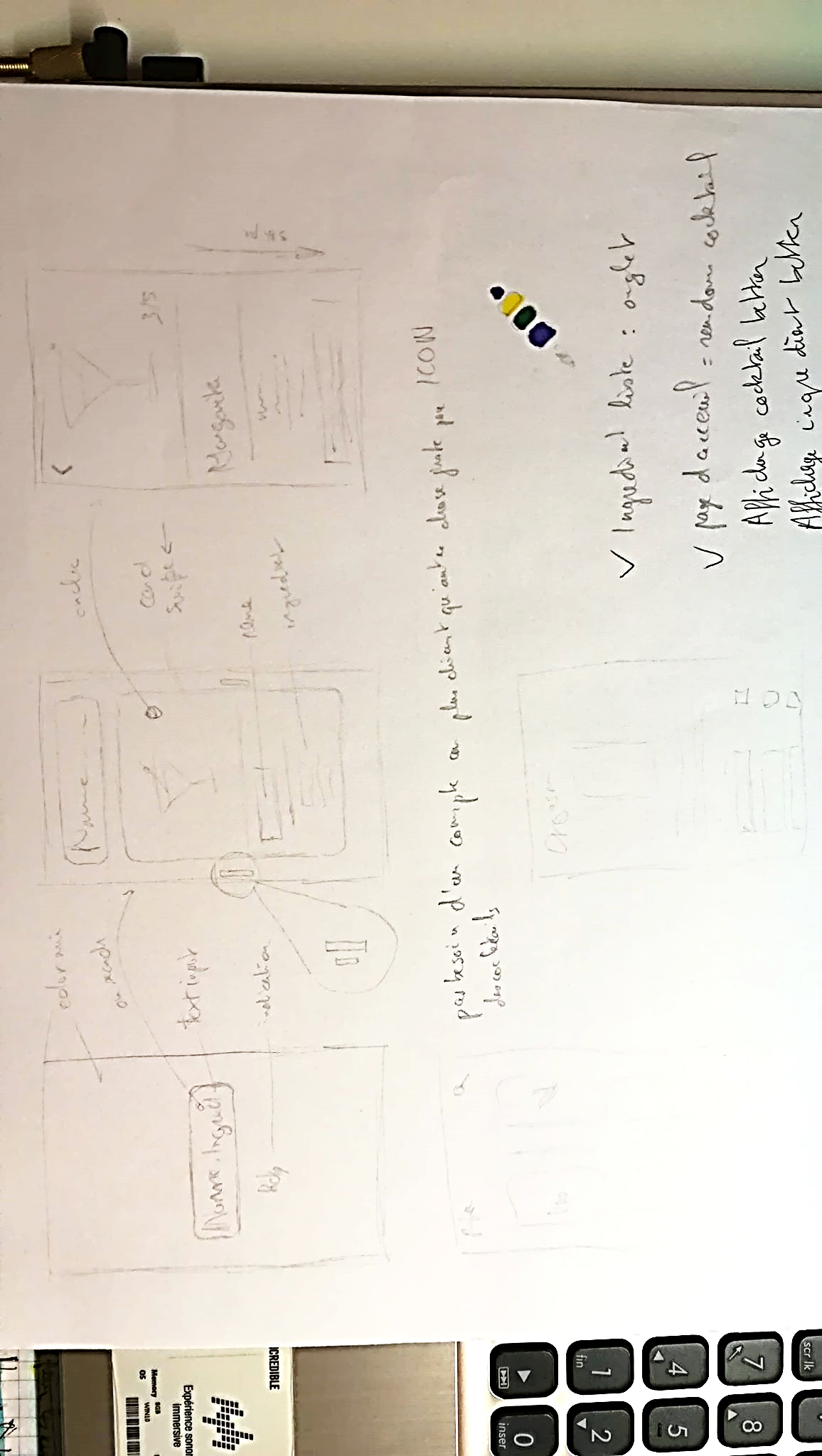
Cocktail application studied :

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

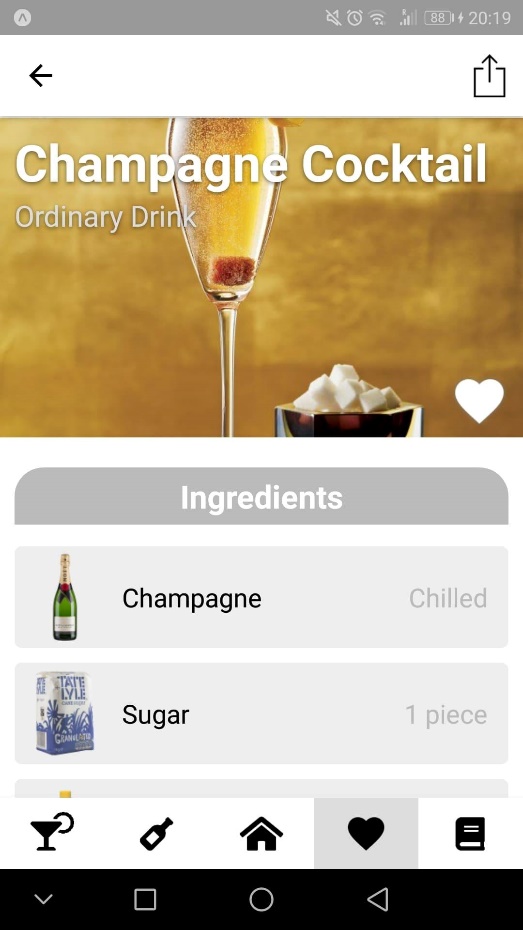
<https://play.google.com/store/apps/details?id=hu.distinction.cflow>

APPENDIX

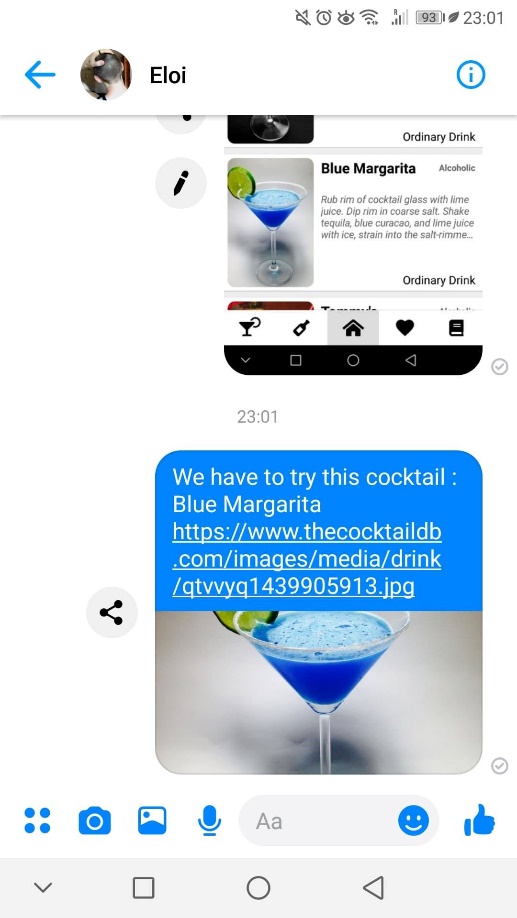
# Appendix 1 - Mockup rough



# Appendix 2 – CocktailDetail, Ingredient and Share View

*CocktailDetail view with tags, recipe, ingredients and glass*

*Ingredient view with description Cocktail share on Messenger*