



Institut Supérieur D'Électronique de Paris

# News App Report

**Prepared for**

[II.3510 Mobile Application  
Development for Android]

**Prepared by**

[Mario Aroca Páez - 62518]  
[mario.aroca-paez@eleve.isep.fr](mailto:mario.aroca-paez@eleve.isep.fr)

[Benjamin Guiloff - 62528]  
[benjamin.guiloff@eleve.isep.fr](mailto:benjamin.guiloff@eleve.isep.fr)

## **Index**

<b>Introduction</b>	<b>3</b>
<b>Architecture</b>	<b>4</b>
<b>Problems</b>	<b>8</b>
<b>References</b>	<b>9</b>
<b>Source code</b>	<b>10</b>

## Introduction

News App is an android app for reading and sharing news from different sources and countries. The news articles are retrieved through the [News API](#).

To create an account and/or log in, it is necessary to have a Google account. The user's information is then stored in a Firebase database.

A preview of the news articles is available in the app and if interested in more details the app is able to redirect the user to the source site to see the whole article. It is also possible to share the different articles with a link provided by the API.

When searching for News, the app is able to filter between different categories being *Business*, *Entertainment*, *General*, *Health*, *Science*, *Sports and Technology*. Also, the user can search for specific keywords with the available text search bar at the top of the news feed.

News articles are available from a large list of different countries. The app allows the user to search for news from all the countries at the same time, or from one specific country. Once a country is selected, the app will remember that choice even after the user has logged out.

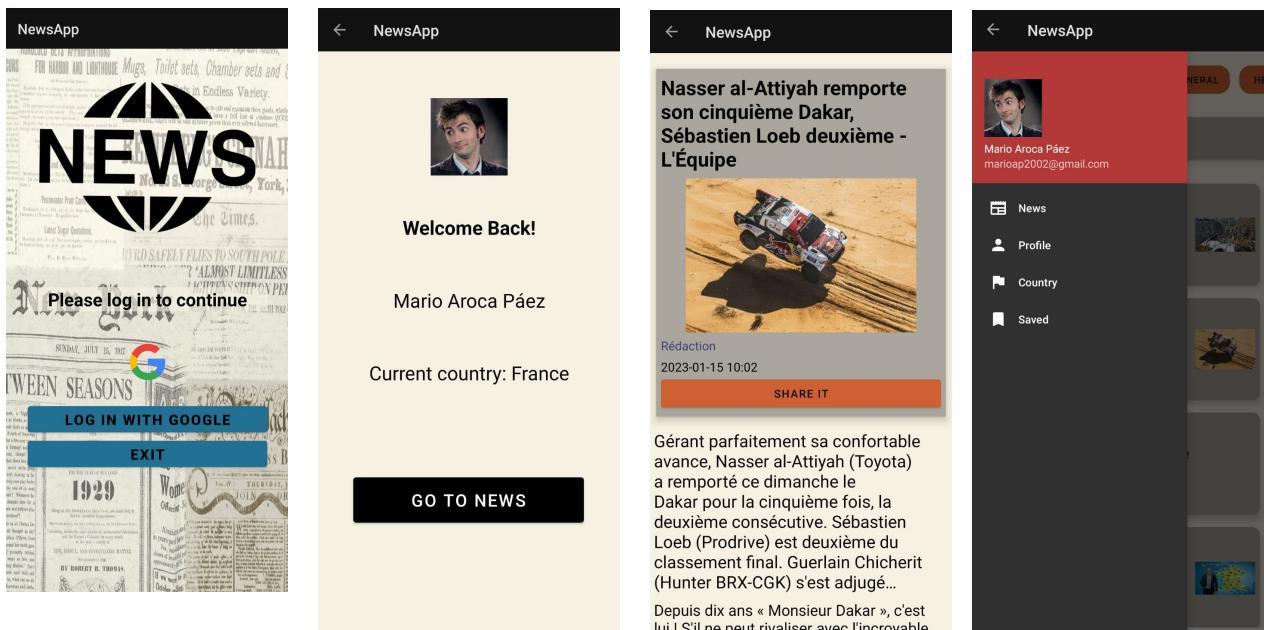
Finally, users can save news articles of their interest, which will then be displayed in the "Saved news" page. This feature is designed for the cases in which the user wants to read or reread the article later.



## Architecture

When opening the app the first screen the user will find is a simple Login screen where he will be able to access the app by logging with his google account or else exit the app.

Then if it is the first time the user accesses the app he will see the main news screen where he will have news from all around the world. If it is not his first time accessing the app with that google account he will see a welcome screen that reminds the country he has selected. From this news screen it is possible both to enter inside one article to see more details, or to swipe right/click the menu logo in the top left in order to display the side toolbar with more functionalities.



If the news article is clicked, a new screen will be displayed with a headline, an image of the news (if available if not there will be a question mark displayed) the author, publication date and a button with the option to share the news article. The link for the original news article is what will be shared, not the link to the app. Then a brief description of the article and a button at the bottom in order to see more details, it will directly redirect you to the source page.

Depuis dix ans « Monsieur Dakar », c'est lui ! S'il ne peut rivaliser avec l'incroyable palmarès de Stéphane Peterhansel sur l'épreuve (six victoires en motos, huit en autos), Nasser al-Attiyah est b... [+3207 chars]

[SEE MORE DETAILS](#)

If the toolbar is selected the user will find the following options: *News*, *Profile*, *Country* and *Saved*. Also in the header it is displayed the profile picture of the user, his name and gmail.

When selecting one of the four different available options if *News* is chosen the app will simply return to the Main News Screen. When the *Profile* option is chosen the user will have a short summary with his profile picture, gmail and name and the option to Log out of the app.

For the *country* option the user will see a large list with different countries (including all) and flags from which he can choose where he wants to get his news from. At the top of this list the app will remind the user which country is currently selected. This information is stored in the Firebase database for each different user.

Finally for the *Saved* option the app will display a list of the different news the user has saved to read later. If the user has not saved any news the app will just tell him that there is no news to be displayed because he has not saved any. Saving news is done using the bookmark we can see at the left of each article preview, blank meaning that it has not been saved and dark meaning that it has been saved.



In order to save or delete news from the saved list the user can add and delete any he wants at the News Screen. Inside the Saved screen he will only be able to delete saved news, however the news will not disappear until the next time the user access that screen, so if he regrets deleting the article from the saved list he can still re save it before leaving the screen.

Regarding the storage of the user's information, we make use of the Firebase Realtime Database. Each user is a node in the database, represented by their user id.

Each user node has 3 children: "country\_code" which stores a 2 letter code of the country selected, that is used to make the call to the API; "country\_label" which stores the name of the country selected; and "saved", which stores the news articles the user has saved.



For the saved news, each article is stored by setting the publication time as the key (for example "2023-01-14T21:07:58Z"), and the information of the article as the value.



When retrieved, the data of each article is converted into the custom java object Articles.

For the API call we are just doing normal GET calls to News Api with our own Api key that can be found in the [documentation](#). However there are 3 parameters that are changed in the different calls depending on our actions. First we have the country from which the news articles are retrieved. There are a total of 51 different countries and also there is the possibility of having all the countries at the same time.

In order to choose one country its equivalent code must be entered in the API call, if no code is entered the API call will retrieve news from all the countries.

**country** The 2-letter ISO 3166-1 code of the country you want to get headlines for. Possible options: ae ar at au be bg br ca ch cn co cu cz de eg fr gb gr hk hu id ie il in it jp kr lt lv ma mx my ng nl no nz ph pl pt ro rs ru sa se sg si sk th tr tw ua us ve za . Note: you can't mix this param with the **sources** param.

Then there are also different categories which are provided by the own API, these are business, entertainment, general, health, science, sports and technology and can be selected within one API call. Finally there is the keyword parameter, An specific term or series of terms can be entered in the API call and it will return news articles that have those terms, in the app it is used for the search box.

## Problems

The main issues we faced while developing the app were related to the uploading and retrieving of information from the Firebase database. According to the documentation we read, the only way to retrieve data of a Firebase database from a java application is by attaching an asynchronous listener to the database reference. This listener is triggered every time there is a change in the data, and also once for the initial state of the data.

Since the callback method of the listener to retrieve the data is relatively slow (take 1-2 seconds), our problem was that the data was not available before displaying the layout of a certain activity. For example, if we started the Saved News activity (by going to the “saved news” tab in the sidebar) and immediately retrieved the data for the saved news, it was impossible to load the recycler view with the necessary information, because said data arrived at the end of the thread.

The way we managed to solve this was by making the call to the database on the News activity instead of the Saved News activity. This way, once the user goes to the “saved news” tab, an Array List with the saved articles is passed to the Saved News activity through the Intent object, using the .putExtra() method. By the time the layout for the Saved News activity displays, the information necessary is already available to use.

Another problem we encountered, caused by the same reasons as stated before, was the feature to maintain the country selected by the user between log out and log in. As this information is also stored in the Firebase database, the asynchronous nature of the retrieval of the data didn’t allow us to get that information before displaying the layout of the news feed, if the call was made in the same activity.

Similar to the first problem, we managed to overcome this problem by displaying an activity before going to the news feed, in which we made the call to the database to access the information of the country selected. In this case, we did it by starting the Profile activity immediately after the login, giving a welcome message to the user and reminding them of the selected country. Once the user presses the button to “Go to news”, the information of the country is already available and the API call can be correctly executed.

## References

- <https://guides.codepath.com/android/Displaying-Images-with-the-Picasso-Library>
- <https://square.github.io/picasso/>
- <https://newsapi.org/docs>
- <https://firebase.google.com/docs/auth/android/google-signin>
- **How to make a Share Button in Android App Android Studio 2.2**
- **How to Change App Icon in Android Studio | Beginner Tutorial**
- **Android Navigation View With Navigation Component | Material Design Co...**
- <https://stackoverflow.com/questions/30659569/wait-until-firebase-retrieves-data>
- <https://firebase.google.com/docs/database/admin/retrieve-data>

## **Source code**

Source code of the project is available on the following link of our github repository:

- <https://github.com/MarioArocaPaez/NewsAppAssignment2ISEPAD>