

Goodness Groceries - Eco-Friendly Mobile Application for iOS Summary

Tuesday 15th December, 2020 - 15:29

Flavio De Jesus Matias
University of Luxembourg
Email: flavio.dejesus.001@student.uni.lu

Benoît Ries
University of Luxembourg
Email: benoit.ries@uni.lu

Abstract—This document presents the bachelor semester project of Flavio De Jesus Matias under the tutoring of Benoît Ries. The project consists of developing an eco-friendly mobile application for iOS by partnering up with *Pall Center*, a Luxembourgish supermarket, and the Sociology research department of the University of Luxembourg. The scientific part of the project focuses on the use of evaluation techniques for mobile applications to evaluate the produced technical deliverable. The technical section of the project is the iOS application.

1. Introduction & Project description

Ethical shopping has been a relatively new trend for most consumers around the world. A lot of people don't pay attention to the provenance of the products they buy, as well as the company that hides behind it. Most of these huge companies do not respect peoples values and in fact, use their revenue to actively work on the opposite.

This project comes in handy for people starting to pay attention to what they buy, along with people that already practice this kind of shopping. The goal of this application is to promote ethical and eco-responsible shopping to the consumers by giving them an easy approach to consult the different products they are willing to buy. It provides them with a transparent platform to increase their knowledge of the companies that hide behind certain products.

The objective of the project is to develop the mobile application *Goodness Groceries* and the use of evaluation techniques to evaluate the User Experience (UX) on the app.

2. Prerequisites

The scientific prerequisite of this project is the knowledge of a simple sophomore Computer Science student (3rd-semester student). All the necessary scientific tasks for this project, such as User Experience evaluation techniques, will be presented and illustrated in the scientific deliverable section.

The technical prerequisite of this project is the knowledge in the programming language Swift alongside with some basics of the framework SwiftUI. Although the report introduces some SwiftUI elements, it is recommended to already have some background knowledge because otherwise, the reader might get lost.

3. Requirements

There are **two** functional requirements for the scientific deliverable:

- FR1: The deliverable shall present a process for the UX evaluation of the application.
- FR2: The deliverable shall present the results of the presented evaluation process applied to the technical deliverable.

For the technical deliverable, there are **three** functional requirements:

- FR1: Each product needs to be displayed in a well-structured view including the corresponding attributes.
- FR2: The app shall provide the customers with a quick way to find products by allowing them to scan QR-codes with the camera of their smartphone.
- FR3: The app should be capable of establishing a stable connection to the distant web server to allow customers to send an authentication request and fetch their status.

4. Design & Production

4.1. Scientific Deliverable

The scientific part of the project focuses on the use of evaluation techniques for mobile applications to evaluate the produced technical deliverable. We can divide the scientific deliverable into two parts: the **evaluation process** and the **UX result analysis**.

4.1.1. Evaluation process. The evaluation process is composed of multiple steps.

- 1) The evaluation process starts by explaining the concept and objective of the project to the participants not yet familiar with it.

- 2) After the participant feels comfortable and understands the project, they get to use the app for the first time and are given the sheet with useful barcodes, i.e., the *carte de fidélité* and QR-codes of the products. Each participant will then be given some minutes to navigate through the app by itself depending on the knowledge they have in this field and how comfortable they feel about using an unknown app.
- 3) The participants are then given the sheet with the missions to accomplish while using the app.
- 4) Once the participant finishes the missions, they are given the questions sheet which evaluates each targeted attribute of the evaluation.

4.1.2. UX result analysis. The questions were evaluated using a 1-5 ranking, which were represented by words such as, for instance, *very bad*, *bad*, *normal*, *good* and *very good*. As shown in Table 1, the average of each attribute is outstanding, as except for one, all are above 4.0. The overall average (average of the averages) of all the attributes is a remarkable **4.55**, which is a very good result. The only attribute which is below 4.0 is **user errors** due to some bugs detected during the evaluation process.

In the end, the results are very good.

Attribute	P-1	P-2	P-3	P-4	P-5	Average
Learnability	4	5	4	5	5	4.6
Efficiency	5	4	5	5	4	4.6
Memorability	5	5	4	5	5	4.8
User errors	5	5	3	4	2	3.8
User satisfaction	5	4	4	4	5	4.4
Effectiveness	5	4	5	5	5	4.8
Simplicity	5	5	4	5	4	4.6
Comprehensibility	5	5	3	5	5	4.6
Learning performance	5	5	5	4	5	4.8

Table 1: Evaluation results for each attribute with 5 participants

4.2. Technical Deliverable

The technical section of the project is the implementation of the design prototype as an iOS application. The deliverable was produced following the **Model-View-ViewModel** (MVVM) architecture. Using this design pattern, we could separate the backend logic from the views.

The Models define the structure of the data which is manipulated and used in the application. The goal of the ViewModels is to load data from the static JSON files into the Models. These files contain all the products, indicators and categories necessary for the app. The ViewModels pass the data to the Views and act as a middleman between the Models and the Views.

The Views can be divided into the **welcome pages** (see Figure 2), the **categories and products** (see Figure 3) and the **scanning tool** (see Figure 1). Each view was created using SwiftUI elements such as VStacks and HStacks, which allow creating vertical and horizontal stacks. On the categories list view, each icon follows the colour palette given

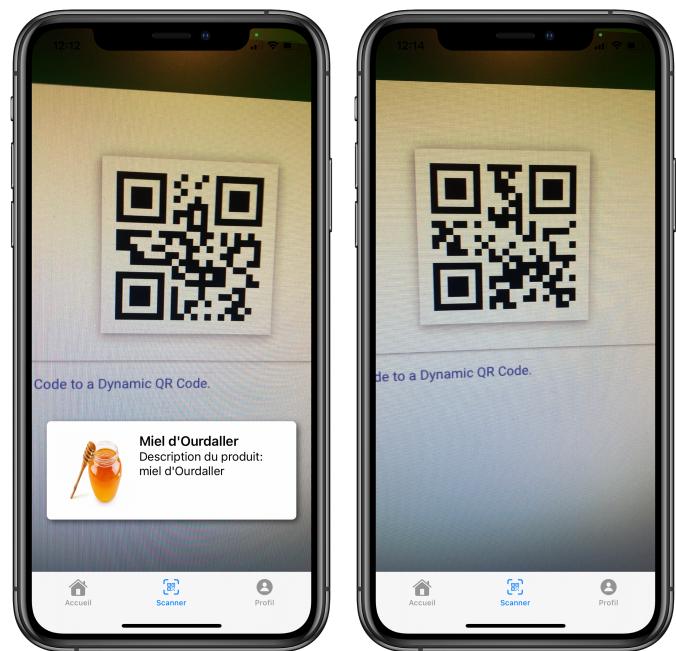


Figure 1: Goodness Groceries scanning tool design (left: valid QR-code; right: invalid QR-code)

by the Pall Center and sociologists team. The colours are as following:

- **Environnement** → #F5BA8E
- **Bien-être économique** → #FCEDA3
- **Bien-être social** → #CCEB97
- **Bonne gouvernance** → #AAE4FB

Overall, the Views were created in a very simple way (stacking elements inside each other). Apart from some third-party libraries, for instance, the QR-code scanning tool, the entire application was built using only basic SwiftUI elements (VStacks, HStacks, ScrollView, NavigationLinks, etc.).

5. Assessment & Conclusion

This paper presents a summary for the bachelor semester project whose goal is the development of an eco-friendly mobile application for iOS and the execution of a selected UX evaluation technique on it. The scientific part focusses on the UX evaluation of the application, while the technical part is the development of the application itself. In the end, the app achieved every functional requirement set and provides all the expected functionalities. Thanks to the evaluation results, the weaknesses of the application were detected and this allows us to address them in more depth in the future.

After all, this project allowed me to learn a lot about iOS mobile app development while participating in such an interesting project, as well as about the different evaluation techniques used for mobile applications and what is important for the end-users.

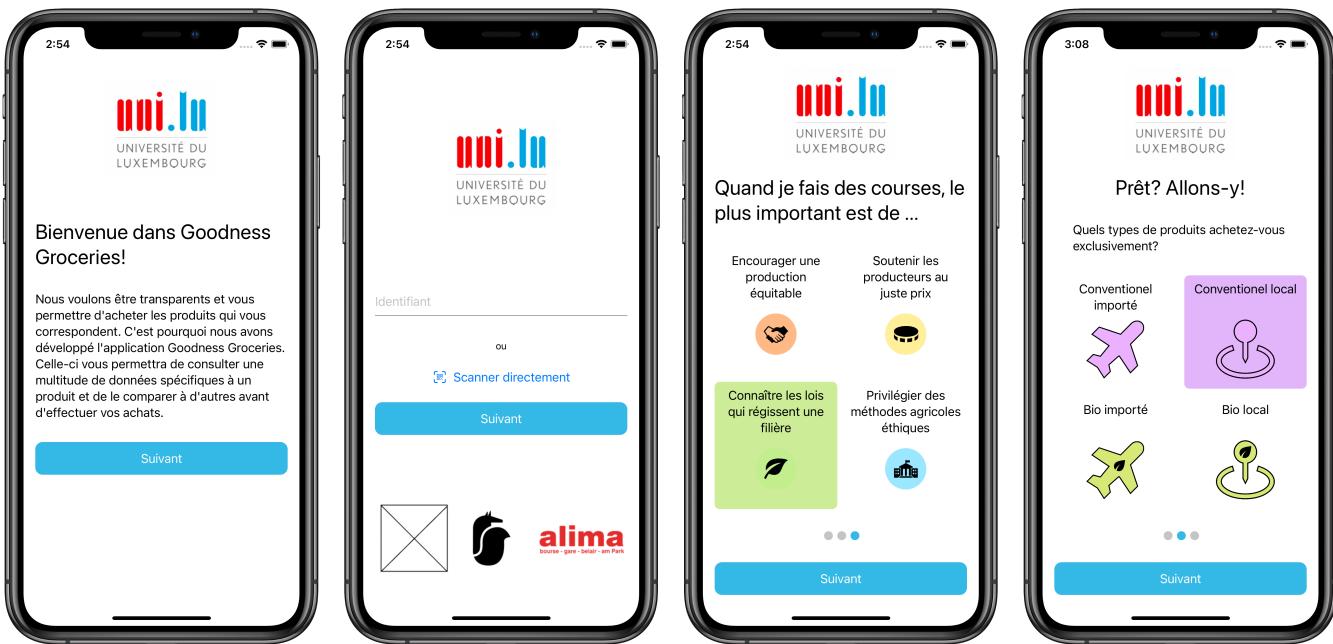


Figure 2: Goodness Groceries user interface design (welcome pages; not complete)

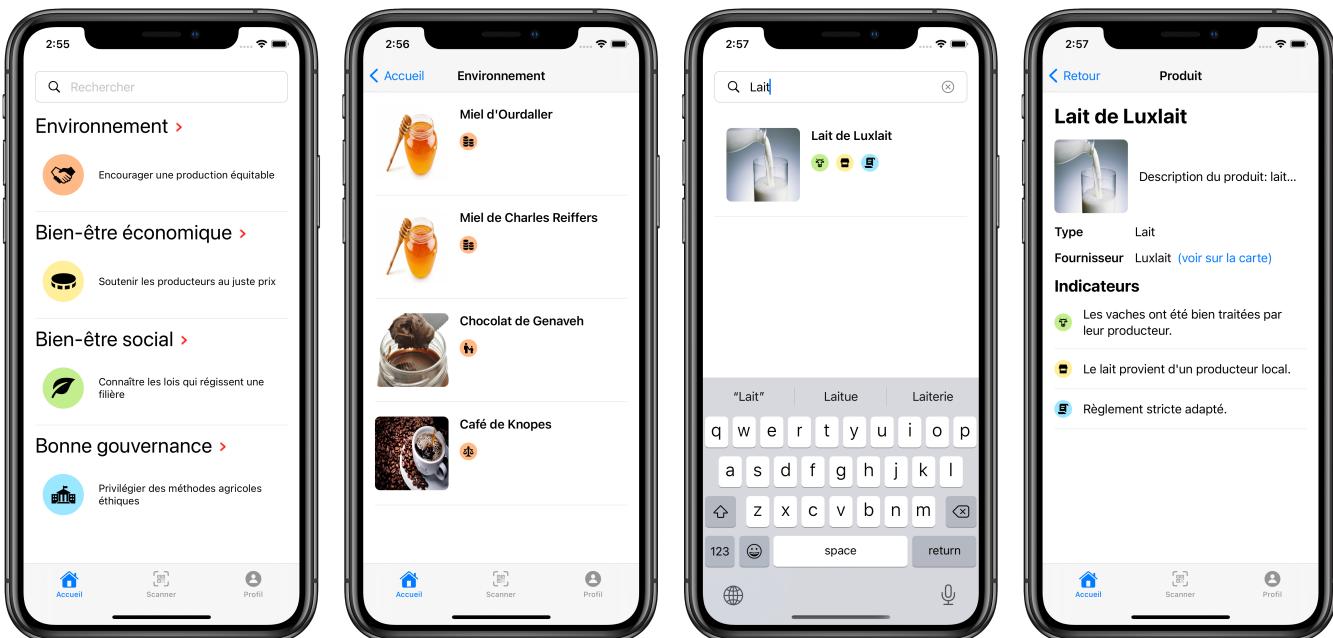


Figure 3: Goodness Groceries user interface design (categories and products)