

# **Goodness Groceries - Umweltfreundliche mobile Anwendung für iOS**

## **Zusammenfassung**

**Wednesday 16<sup>th</sup> December, 2020 - 15:37**

Flavio De Jesus Matias  
*Universität Luxemburg*  
Email: [flavio.dejesus.001@student.uni.lu](mailto:flavio.dejesus.001@student.uni.lu)

Benoît Ries  
*Universität Luxemburg*  
Email: [benoit.ries@uni.lu](mailto:benoit.ries@uni.lu)

**Abstract**—Dieses Dokument ist die Zusammenfassung des Bachelor-Semesterprojekts von Flavio De Jesus Matias unter der Leitung von Benoît Ries.

## **1. Einführung & Projektbeschreibung**

Ethisches Einkaufen ist für die meisten Verbraucher weltweit ein relativ neuer Trend. Dieses Projekt ist besonders interessant und praktisch für Leute, die anfangen, auf das zu achten, was sie kaufen, sowie für die Leute, die diese Art des Einkaufens bereits praktizieren. Ziel dieser Anwendung ist es, den Verbrauchern ein ethisches und umweltbewusstes Einkaufen zu ermöglichen, indem sie ihr Wissen über die verschiedene Produkte, die sie interessieren, erweitern können.

Ziel des Projekts ist die Entwicklung der mobilen Anwendung *Goodness Groceries* und die Verwendung von User Experience (UX) Bewertungstechniken zur Bewertung der App.

## **2. Voraussetzungen**

Für das wissenschaftliche Teil des Projekts gibt es **zwei** verschiedene funktionale Anforderungen: einen UX Bewertungsprozess präsentieren und die Bewertungsergebnisse analysieren.

Für das technische Teil des Projekts gibt es **drei** funktionale Anforderungen: jedes Produkt mit den dazugehörigen Attributen anzeigen, Produkte finden indem man QR-Codes mit der Kamera des Smartphones scannen kann und eine stabile Verbindung zum Webserver aufbauen für den Authentifizierungsprozess.

## **3. Design & Produktion**

### **3.1. Scientific Deliverable**

Der wissenschaftliche Teil des Projekts kann in zwei Teile unterteilt werden: den **Bewertungsprozess** und die **UX-Ergebnisanalyse**.

**3.1.1. Bewertungsprozess.** Der Bewertungsprozess besteht aus mehreren Schritten.

- 1) Der Bewertungsprozess beginnt mit der Erläuterung des Konzepts und des Ziels des Projekts gegenüber den Teilnehmern, die noch nicht damit vertraut sind.
- 2) Die Teilnehmer benutzen die App zum ersten Mal und erhalten einen Blatt mit verschiedene QR-codes, d.h. die *carte de fidélité* und die QR-Codes der Produkte. Jeder Teilnehmer erhält dann einige Minuten Zeit, um selbst durch die App zu navigieren.
- 3) Die Teilnehmer erhalten dann das Blatt mit den Missionen, die sie während der Nutzung der App ausführen müssen.
- 4) Sobald der Teilnehmer die Missionen beendet hat, erhält er das Fragenblatt, in dem jedes Zielattribut der Bewertung bewertet wird.

**3.1.2. UX-Ergebnisanalyse.** Wie wir in Tabelle 1 sehen können, sind die Ergebnisse für jedes Attribut sehr gut (über **4.0**). Das einzige Attribut unter 4.0 ist **User errors** und das liegt an verschiedene Fehler die während dem Bewertungsprozess aufgetaucht sind. Alle Fehler wurden notiert.

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

*Tabelle 1: Bewertungsergebnisse für jedes Attribut mit 5 Teilnehmern*

### **3.2. Technical Deliverable**

Der technische Teil des Projekts ist die programierung der iOS-Anwendung. Die Anwendung wurde nach

der MVVM-Architektur (**Model-View-ViewModel**) erstellt. Mit diesem Entwurfsmuster könnten wir die Backend-Logik von den Ansichten trennen.

Die Modelle definieren die Struktur der Daten, die in der Anwendung bearbeitet und verwendet werden. Das Ziel der ViewModels ist das Laden von Daten aus den statischen JSON-Dateien in die Modelle. Diese Dateien enthalten alle für die App erforderlichen Produkte, Indikatoren und Kategorien.

Die Ansichten können in die **Begrüßungsseiten** (siehe Abbildung 2), die **Kategorien und Produkte** (siehe Abbildung 3) und das **QR-Code-Scan-Tool** (siehe Abbildung 1) unterteilt werden. Jede Ansicht wurde mit SwiftUI-Elementen wie VStacks und HStacks erstellt, mit denen vertikale und horizontale Stapel erstellt werden können.

Insgesamt wurden die Ansichten auf sehr einfache Weise erstellt (Elemente ineinander stapeln). Abgesehen von einigen Bibliotheken von Drittanbietern, beispielsweise dem QR-Code-Scan-Tool, wurde die gesamte Anwendung nur mit grundlegenden SwiftUI-Elementen (VStacks, HStacks, ScrollView, NavigationLinks usw.) erstellt.

die verschiedenen Bewertungstechniken, die für mobile Anwendungen verwendet werden.

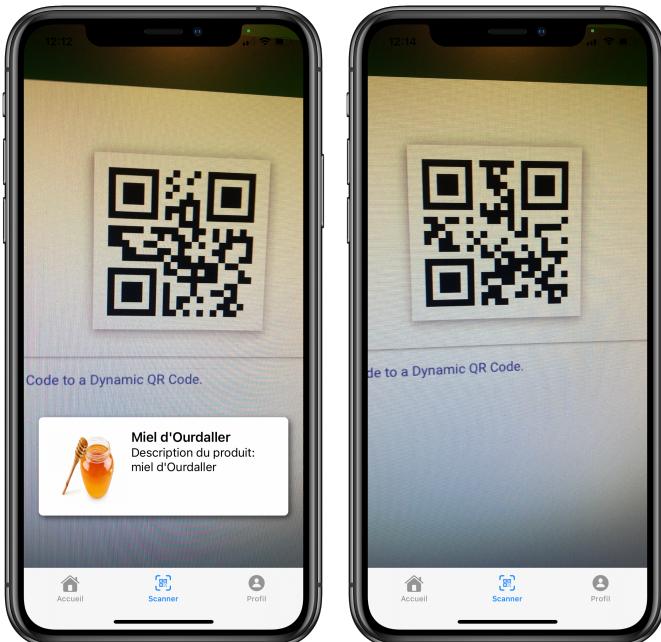


Figure 1: Goodness Groceries - QR-Code-Scan-Tool (links: gültiges QR-code; rechts: ungültiges QR-code)

#### 4. Bewertung & Abschluss

Die zu Beginn gestellten Anforderungen wurden erfolgreich erfüllt da am Ende die App alle erwarteten Funktionen bietet. Dank der Bewertungsergebnisse wurden die Schwachstellen der Anwendung auch erkannt, sodass wir sie in Zukunft genauer untersuchen können.

Schließlich konnte ich in diesem Projekt viel über die Entwicklung von iOS-Apps für Mobilgeräte lernen, während ich an einem so interessanten Projekt teilnahm, sowie über

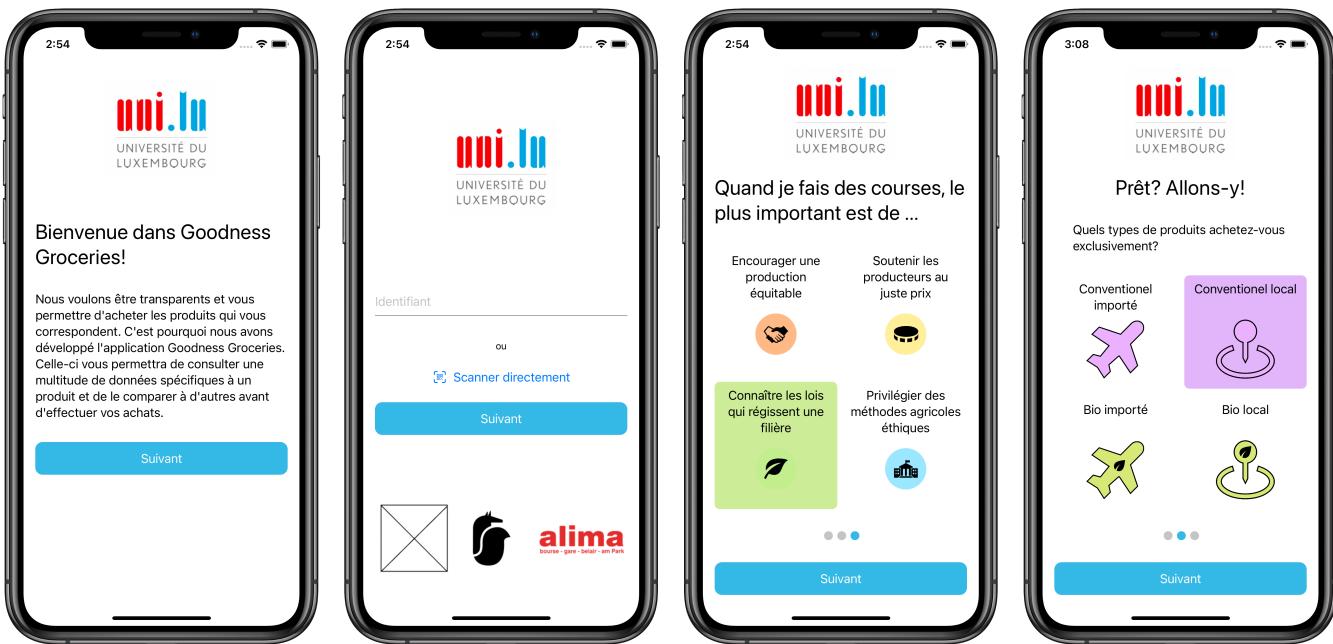


Figure 2: Goodness Groceries - Begrüßungsseiten (nicht vollständig)

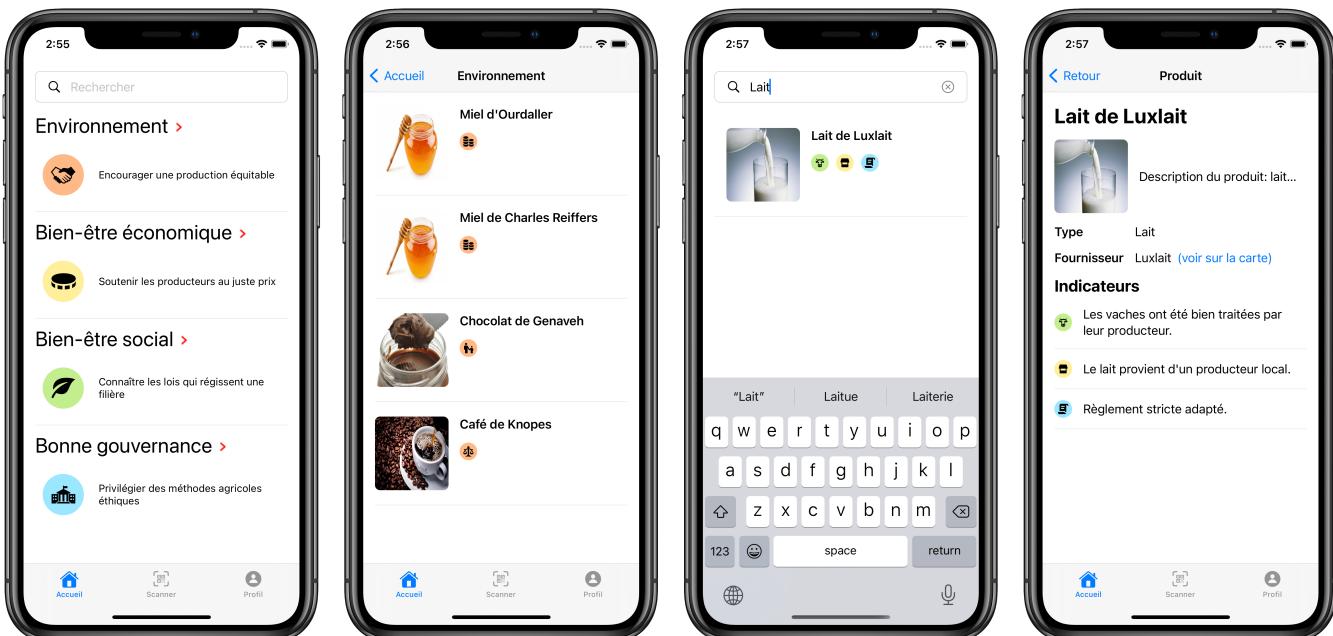


Figure 3: Goodness Groceries - Kategorien und Produkte