

APPLICATION MOBILE POUR LA DIGITALISATION D'UN EPICIER

Samira ETTAQY

Group : G1

2025/2026

1	Presentation du projet.....	2
1.1	Introduction	2
1.2	Problématique	2
1.3	Objectifs Général du Projet.....	2
1.4	Solution.....	3
2	Spécification des besoins	4
2.1	Besoins fonctionnels	4
2.1.1	Besoins Fonctionnels - Administrateur	4
2.1.2	Besoins Fonctionnels - Épicier	4
2.1.3	Besoins non fonctionnels	4
2.2	Traduction des besoins en diagramme de cas d'utilisation	5
2.2.1	Description des acteurs :	5
2.3	Diagramme de cas d'utilisation global	5
3	Architecture et Conception.....	6
4	Conception et Modélisation	7
4.1	Diagramme de Classe	7
4.2	Diagramme de Séquence.....	9
5	Réalisation et Test	10
5.1	Outils et Technologies Utilisés	10
5.1.1	Environnement de Développement.....	10
5.1.2	Technologies Frontend.....	10
5.1.3	Technologies Backend.....	10
5.1.4	Outils de Développement et Tests.....	11
6	Réalisation	11
6.1	Les interface de l'application.....	11
6.1.1	Espace epicier	11
6.1.2	Espace Administrateur	15

1 Presentation du projet

1.1 Introduction

Dans un contexte économique où la digitalisation devient incontournable, les petits commerces de proximité, notamment les épiceries, font face à des défis majeurs de gestion. La gestion manuelle des stocks, le suivi des ventes et la prévention des pertes liées aux produits périmés représentent des problématiques quotidiennes pour les épiciers marocains.

Hanooty est une application mobile conçue pour moderniser et simplifier la gestion des épiceries. Développée avec React Native et Firebase, cette solution cross-platform (Android/iOS) offre aux épiciers un outil complet, intuitif et accessible pour gérer leur commerce au quotidien.

Le projet s'inscrit dans une démarche de transformation digitale des commerces de proximité, en proposant une solution adaptée au contexte marocain, notamment par l'intégration de l'API Open Food Facts version marocaine pour une meilleure pertinence des données produits.

1.2 Problématique

Les épiceries de quartier au Maroc représentent un secteur économique vital, mais elles fonctionnent encore majoritairement avec des méthodes de gestion traditionnelles (cahiers, calculs manuels, inventaires approximatifs). Cette approche génère plusieurs problèmes critiques tels que :

- Temps considérable perdu à compter manuellement le stock
- Produits périmés non détectés à temps
- Ruptures de stock non anticipées
- Pas d'historique des ventes
- Saisie manuelle des prix lors de chaque vente
- Absence de système de caisse moderne
- Recherche laborieuse des informations nutritionnelles

Donc :

Comment digitaliser efficacement la gestion d'une épicerie tout en restant accessible aux commerçants avec des compétences techniques limitées ?

1.3 Objectifs Général du Projet

L'objectif principal de ce projet est de développer une application mobile cross-platform permettant aux épiciers de gérer intégralement leur commerce de manière digitale, simple et efficace.

Cette solution vise à moderniser les pratiques de gestion traditionnelles des épiceries en proposant un outil complet qui couvre l'ensemble des besoins quotidiens : de la gestion du

stock à l'analyse des performances commerciales, en passant par le point de vente et le suivi des transactions.

Objectifs Spécifiques :

- Gestion du Stock : Scanner de code-barres, récupération automatique des données avec API Open Food Facts et notification sur les stocks faibles et expirés bientôt ou expirés.
- Point de Vente : Vente par scanner, calcul automatique et mise à jour automatique du stock.
- Statistiques : Visualisation du chiffre d'affaires, produits vendus, répartition par catégorie et historique des ventes.
- Sécurité : Authentification Firebase, isolation des données par utilisateur, et gestion des rôles.

1.4 Solution

Ce projet consiste à concevoir et développer une application mobile s'adresse particulièrement aux commerces de proximité marocains et se veut accessible même pour les utilisateurs ayant des compétences techniques limitées. En tirant des technologies modernes telles que React Native, Firebase et l' API externe Open Food Facts, , le projet ambitionne de transformer radicalement la façon dont les épiciers gèrent leur activité, en leur offrant une visibilité en temps réel sur leur commerce, en réduisant les pertes liées aux produits périmés et aux ruptures de stock, et en leur permettant de prendre des décisions éclairées basées sur des données concrètes et des statistiques détaillées.

2 Spécification des besoins

2.1 Besoins fonctionnels

Le modèle gestion des tâches doit mettre à la disposition des acteurs les fonctionnalités suivantes :

2.1.1 Besoins Fonctionnels - Administrateur

L'administrateur peut :

- Se connecter et se déconnecter de l'espace administrateur.
- Créer des comptes épiciers avec email et mot de passe.
- Modifier les informations des comptes épiciers existants.
- Supprimer des comptes épiciers.
- Consulter la liste de tous les épiciers enregistrés.
- Visualiser les statistiques globales de tous les épiciers
- Accéder à un tableau de bord administrateur avec vue d'ensemble du système.

2.1.2 Besoins Fonctionnels - Épicier

L'épicier peut :

- Se connecter avec les identifiants fournis par l'administrateur.
- Se déconnecter de l'application.
- Consulter et modifier son profil personnel.
- Scanner un code-barres pour identifier un produit.
- Ajouter un nouveau produit avec récupération automatique des informations à partir Open Food Facts .
- Ajouter manuellement un produit avec toutes ses informations (nom, catégorie, prix d'achat, prix de vente, quantité, date d'expiration).
- Modifier les informations d'un produit existant.
- Supprimer un produit de son stock.
- Rechercher un produit par nom ou code-barres.
- Consulter la liste complète de ses produits en stock.
- Scanner des produits pour les ajouter au panier de vente.
- Visualiser le total de la vente calculé automatiquement.

2.1.3 Besoins non fonctionnels

- L'application soit être facile à utiliser

- Le temps de réponse de l'application doit être le plus possible
- L'application doit être portable
- Possibilité d'optimiser l'application plus tard
- L'application doit sécuriser les informations de ses utilisateurs

2.2 Traduction des besoins en diagramme de cas d'utilisation

2.2.1 Description des acteurs :

Dans cette application , nous identifions deux types d'acteurs : l'épicier et l'administrateur

Critère	Administrateur	Epicier
Nombre	Unique	Multiple
Privilèges	Elevés	Standards
Fréquences d'utilisation	Occasionnelle	Quotidienne
Plateforme	Mobile	Mobile
Compétences techniques	Avancées	Limitées
Accès	Espace Admin	Espace épicer
Objectif	Supervision et gestion	Gestion commerciale

2.3 Diagramme de cas d'utilisation global

Nous avons choisi d'étudier le cas d'utilisation qui définit le rôle de l'épicier à savoir la gestion de l'utilisateurs

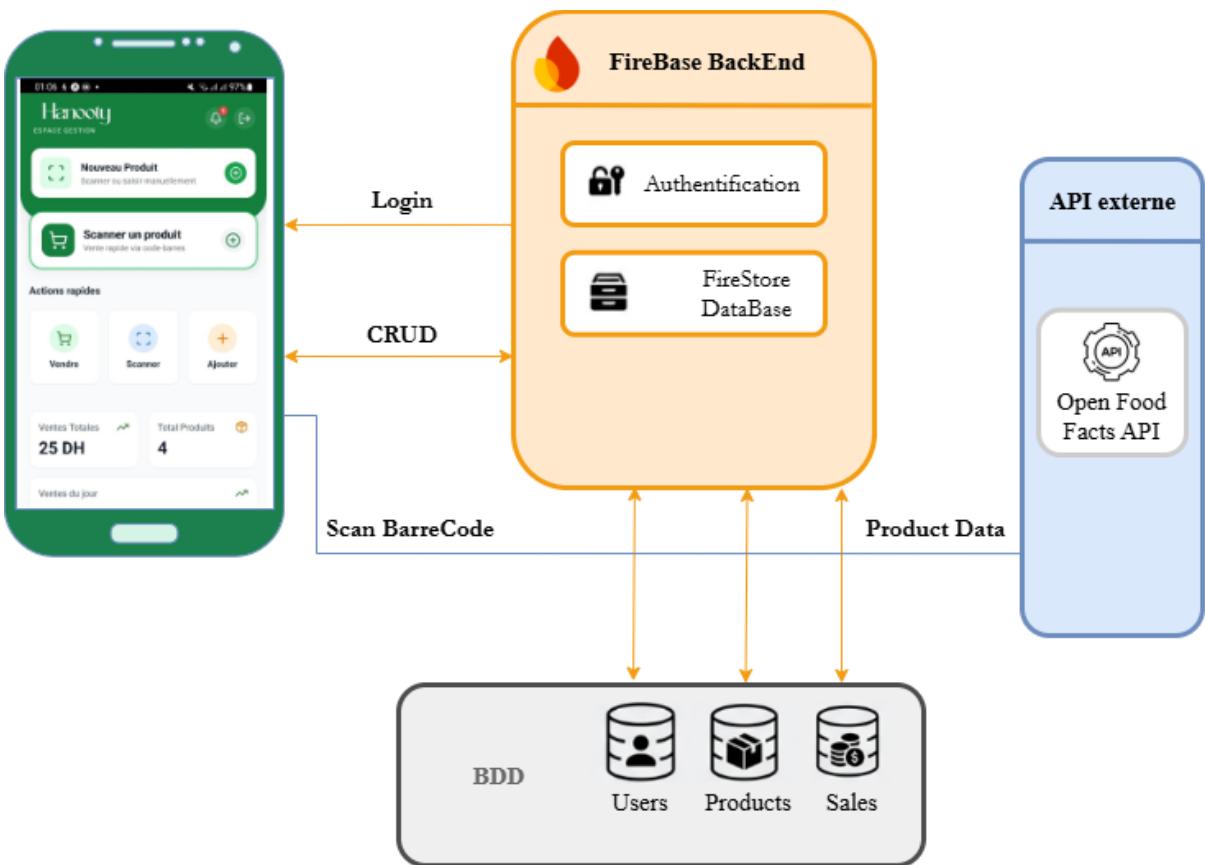
Le diagramme de la figure en décrit les détails :



3 Architecture et Conception

L'application adopte une architecture combinée une application mobile React Native avec un backend Firebase et API externe de Open Food Facts pour l'enrichissement automatique des données produits. Cette architecture en trois couches assure une scalabilité optimale, une synchronisation en temps réel, et une expérience utilisateur fluide sur iOS et Android.

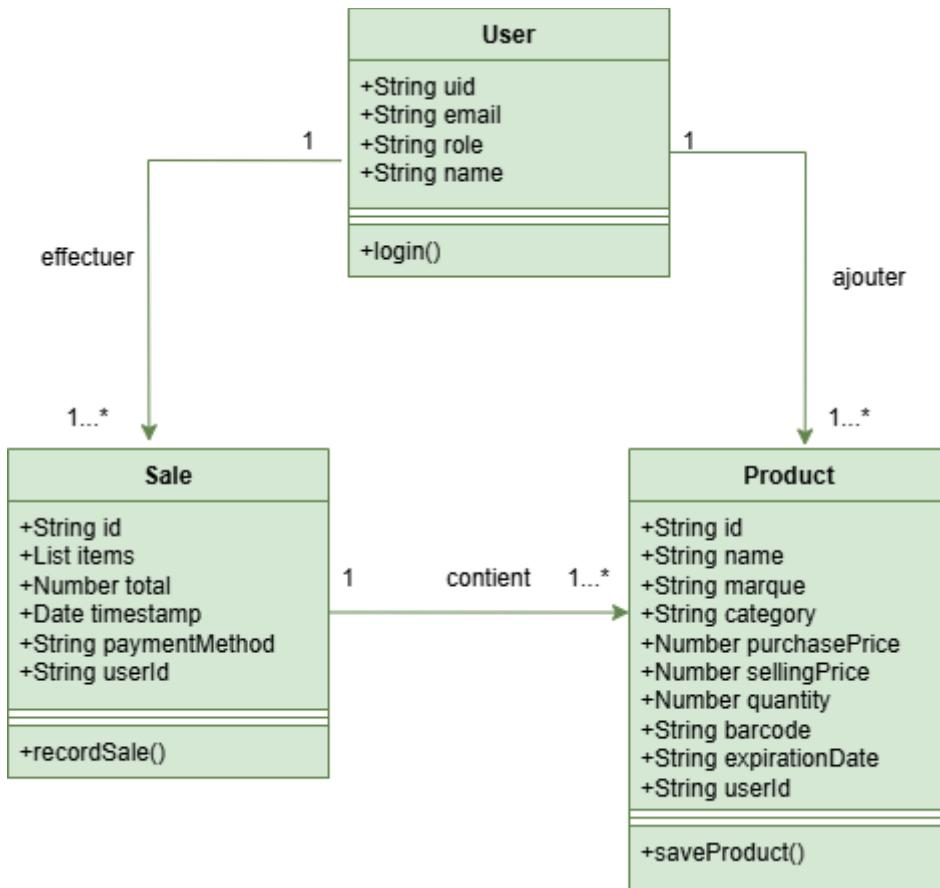
Le backend repose sur Firebase avec deux composants essentiels : Firebase Authentication pour la gestion sécurisée des utilisateurs et de leurs rôles (épicier /administrateur), et Firestore Database, une base de données NoSQL qui organise les données en quatre collections principales : users , products et sales



4 Conception et Modélisation

4.1 Diagramme de Classe

Ce diagramme montre la représentation des entités principales et de leurs relations.



-La classe **Product** modélise les produits en stock avec leurs informations détaillées

-La classe **User** représente les utilisateurs de l'application avec leurs attributs (id, email, Name, role) et leurs méthodes

-la classe **Sale** représente les transactions de vente, contenant une liste d'articles, le montant total

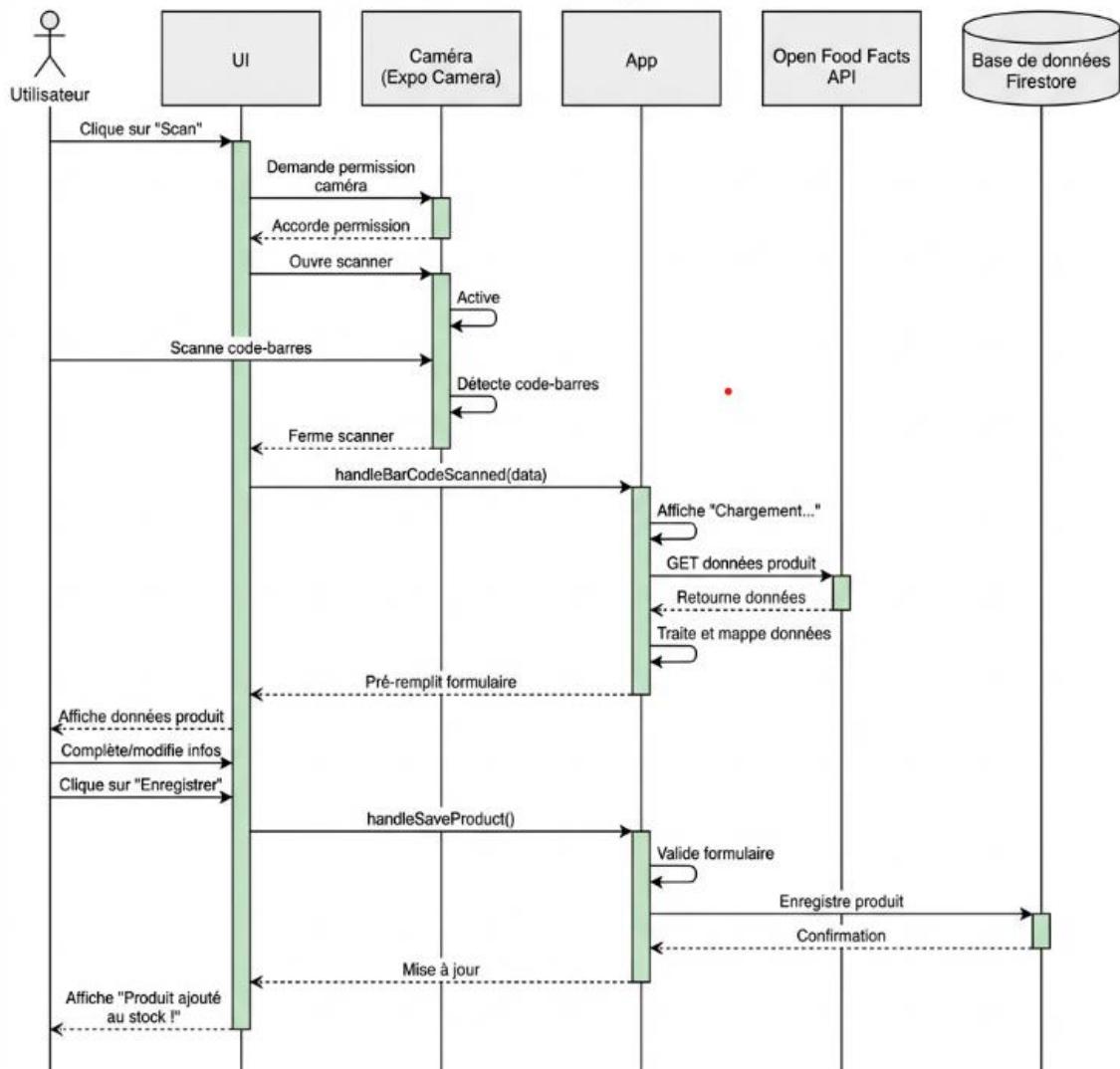
The screenshot shows the Firebase Cloud Firestore interface with the following details:

- Project:** HanootyApp
- Collection:** products
- Document ID:** 0zR2faJXHJlzOJvKg3lf
- Fields:**
 - barcode: "6111028001130"
 - brand: "bel for all for good"
 - category: "Fromages blancs"
 - createdAt: 6 janvier 2026 à 03:36:38 UTC-1
 - description: ""
 - expirationDate: "02/01/2026"
 - image: "https://images.openfoodfacts.org/images/products/611/102/800/
 - ingredients: ""
 - name: "la vache qui rit - bel for all for good"
 - nutriments
 - nutriscore: "unknown"
 - productSize: "96 pièces "

Les relations entre ces classes suivent un modèle hiérarchique : un User peut créer plusieurs Products , et effectuer plusieurs Sales. chaque produit est lié à son créateur, chaque vente est associée à l'utilisateur qui l'a effectuée, et les quantités en stock sont automatiquement mises à jour lors du checkout.

4.2 Diagramme de Séquence

Le diagramme de séquence suivant décrit le processus d'ajout d'un produit à travers le code barre :



Le diagramme de séquence suivant décrit le processus complet de scan de code-barres et d'ajout de produit. Il illustre comment l'application récupère automatiquement les informations du produit depuis l'API Open Food Facts, puis pré-remplit le formulaire d'ajout.

L'utilisateur peut ensuite compléter ou modifier les informations (prix d'achat, prix de vente, quantité, date d'expiration) avant d'enregistrer le produit dans la base de données.

5 Réalisation et Test

5.1 Outils et Technologies Utilisés

5.1.1 Environnement de Développement

Dans ce paragraphe nous décrivons les différents outils et logiciels utilisés pour la mise en œuvre de l'application

L'application Hanooty a été développée avec React Native en utilisant le framework Expo pour faciliter le développement cross-platform (iOS et Android). Expo offre un ensemble d'outils et de bibliothèques qui simplifient le développement mobile.

5.1.2 Technologies Frontend

- React Native : Framework principal pour le développement mobile
- Expo Vector Icons : Bibliothèque d'icônes
- React Native Chart Kit : Pour la visualisation des données, graphiques circulaires et courbes
- Expo Camera : Pour la fonctionnalité de scan de codes-barres

5.1.3 Technologies Backend

Firebase Services

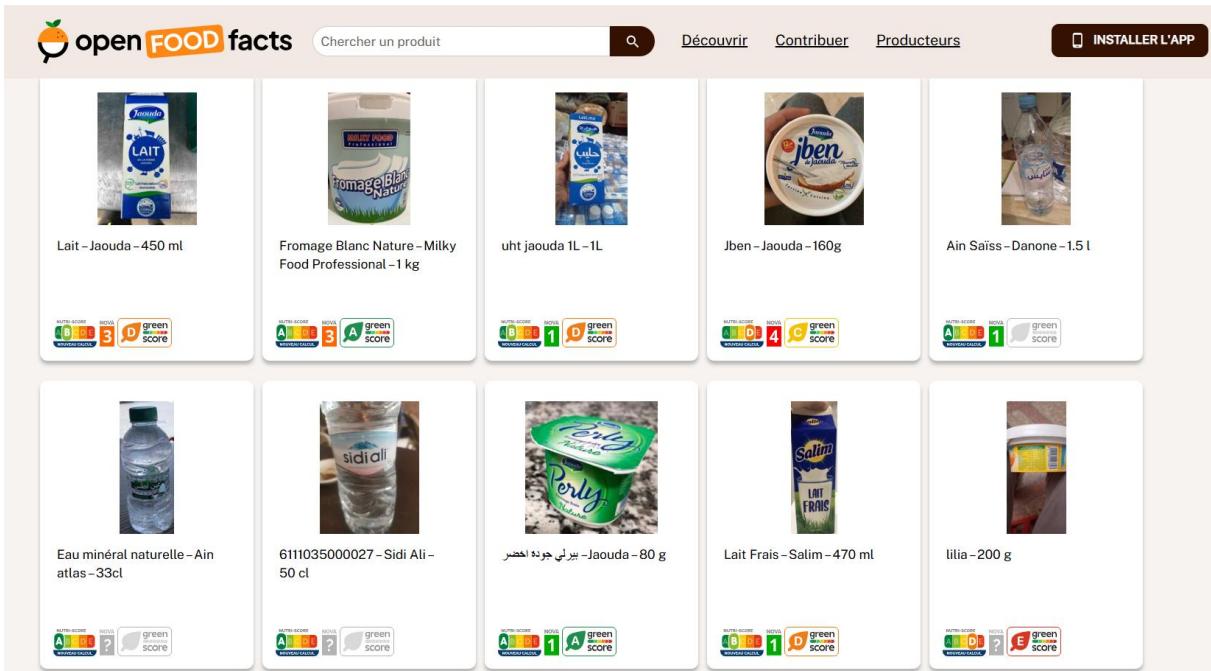
Firebase est utilisé comme backend principal pour l'application :

- Firebase Authentication : Gestion de l'authentification des utilisateurs (épiciers et administrateurs)
- Cloud Firestore : Base de données NoSQL en temps réel pour stocker :
 - Informations des utilisateurs
 - Produits
 - Ventes
 - Historique des transactions
- Firebase Storage : Stockage des images de produits

API Externes

Open Food Facts API : API publique pour récupérer les informations des produits via code-barres.

Données récupérées : nom, catégorie, image, informations nutritionnelles...



5.1.4 Outils de Développement et Tests

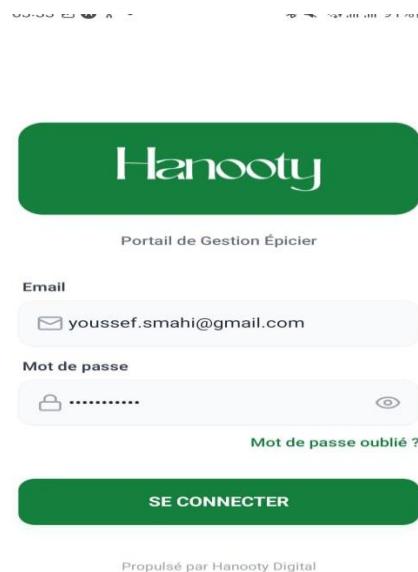
- Expo Go : Application mobile pour tester l'application en développement
- Android Studio: Pour les tests sur émulateurs
- Git : Contrôle de version
- Taiga : Plateforme de gestion de projet agile pour la planification des sprints, gestion des user stories et tâches, suivi de l'avancement du projet et le tableau Kanban pour visualiser le workflow

6 Réalisation

6.1 Les interface de l'application

6.1.1 Espace épicerie

Authentification et Gestion des Rôles :



Ajout de Produits :

Two side-by-side screenshots of a mobile application for adding products. The left screenshot shows a scanning interface with a red camera overlay and instructions: 'Placez le code-barres au centre du carré pour scanner automatiquement.' The right screenshot shows the 'Ajouter un produit' (Add Product) screen. It includes fields for 'Photo du produit' (with an image of a red cow), 'Nom du produit *' (labeled 'la vache qui rit - bel for all for good'), 'Marque' (labeled 'bel for all for good'), 'Format / Poids (ex: 450g, 1L)' (labeled '96 pièces'), 'Code-barres' (labeled '611028001130'), 'Catégorie *' (labeled 'Fromages blancs'), 'Prix d'achat *' (labeled 'DH 95'), 'Prix de vente *' (labeled 'DH 96'), 'Quantité *' (labeled '# 20'), 'Expiration' (labeled '02/01/2026'), and a 'Description' section. A large green button at the bottom right says 'Enregistrer le produit' (Register Product). The bottom navigation bar includes icons for Accueil, Stock, Ajouter, Vente, Stats, and Profil.

Modification et Suppression d'un Produit :

Détails du produit



Kiri 72p - Kiri

MARQUE Kiri

Nutrition et Ingrédients

Nutri-Score

Tableau nutritionnel

	POUR 100G
Énergie	962 kJ (230 kcal)
Matières grasses	20 g
Acides gras saturés	13 g
Glucides	3.5 g
Sucres	2 g
Protéines	9 g
Sel	1.6 g

Détails du produit

Nutri-Score Qualité nutritionnelle médiocre

Tableau nutritionnel

	POUR 100G
Énergie	962 kJ (230 kcal)
Matières grasses	20 g

Ingrédients

Liste d'ingrédients non disponible.

Supprimer ce produit

Processus de Vente :

Caisse / Vente

Chercher par code...

Commande (1)

Vider

	la vache qui rit - bel fromage	-	2	+	
96.00 DH					

Paiement en Espèces

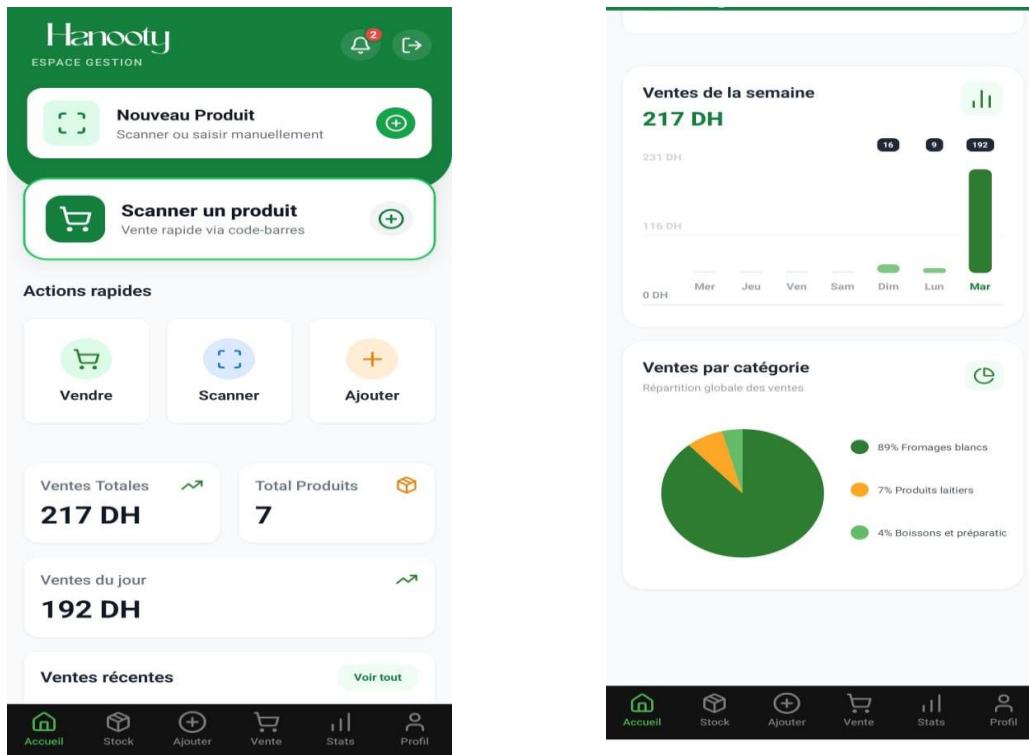
Sous-total 153.60 DH
TVA (20%) 38.40 DH

Total TTC
Taxes incluses **192.00 DH**

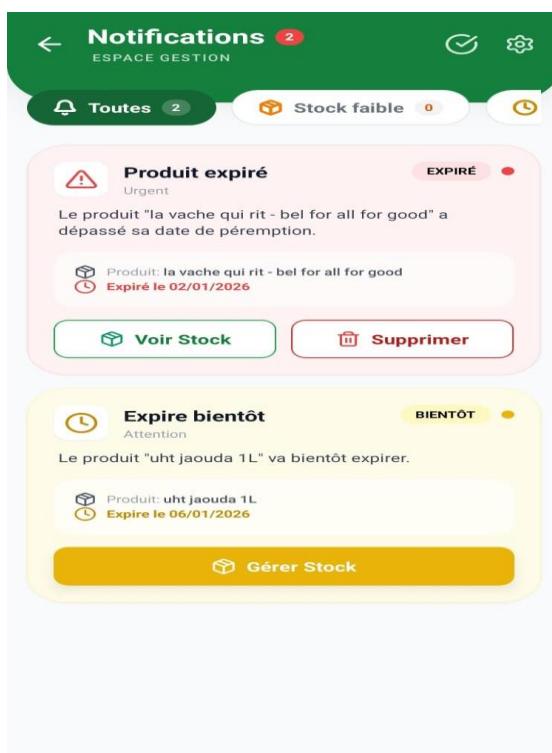
Valider la vente →

Accueil Stock Ajouter Vente Stats Profil

Tableau de Bord et Statistiques

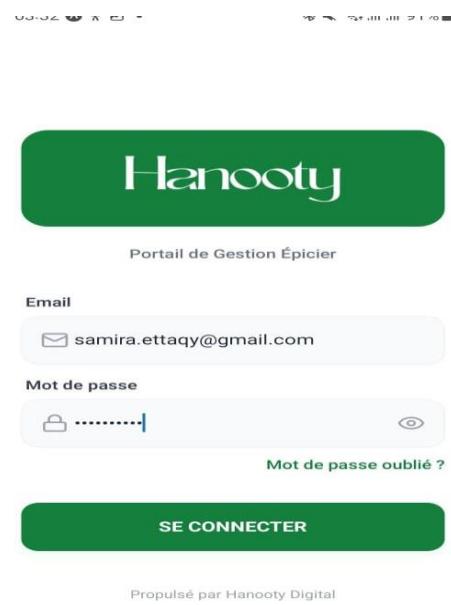


Notifications et Alertes



6.1.2 Espace Administrateur

Authentification et Gestion des Rôles



Gestion des épiciers

This screenshot displays the "Gestion des Épiciers" (Management of Traders) section. At the top, there's a header with a back arrow, the title "Gestion des Épiciers", a "QUITTER" button, and a search bar containing the placeholder "Rechercher un épicer...". Below this, a table lists two users:

Avatar	Nom	Email	Action
	Hanooty	youssef.smahi@gmail.com	
	Hanooty	saraeihilloui12@gmail.com	

At the bottom of the list is a large green button with a white plus sign (+), used for adding new users.

Conclusion

Hanooty répond de manière complète et innovante aux défis de gestion des épiceries modernes. En combinant une technologie de pointe (React Native, Firebase) avec une approche centrée utilisateur, le projet démontre qu'il est possible de digitaliser efficacement les commerces de proximité tout en restant accessible et abordable.

La solution proposée ne se contente pas de numériser les processus existants, elle les réinvente en tirant parti des APIs externes, du temps réel et de l'intelligence des notifications pour offrir une expérience véritablement transformatrice.