UNIVERSITÉ MOHAMMED V DE RABAT
Faculté des Sciences

Université Mohammed V
Faculté des Sciences
Rabat

Département d'Informatique

Filière Licence Fondamentale
en Sciences Mathématiques et Informatique

PROJET DE FIN D'ÉTUDES

Intitulé :

Conception et réalisation d'une application web pour la gestion d'approvisionnement domestique

Présenté par :
ANASS CHAMMAMI
MARWANE EL HAIL

Soutenu le 24 Juin 2024 devant le Jury

M. Moulay Driss RAHMANI	Professeur à la Faculté des Sciences - Rabat	*Président*
Mme. Salma MOULINE	Professeur à la Faculté des Sciences - Rabat	*Examinatrice*
Mme. Safae CHERDAL	Professeur à la Faculté des Sciences - Rabat	*Encadrante*

Année Universitaire 2023-2024

Remerciements

Nos remerciements vont d'abord au Dieu Tout-Puissant, qui nous a donné la force et la volonté de vivre ce moment et de réaliser ce projet.

Au terme de ce travail, nous tenons à exprimer notre profonde gratitude et nos sincères Remerciements à notre encadrante Professeur **Safae Cherdal** qui a proposé le sujet, pour tout le temps qu'il nous a consacré, ses conseils précieux, et pour la qualité de son suivi durant toute la période de notre projet.

Nous tenons également à remercier et à exprimer notre profond respect à nos chers professeurs :
Monsieur **Moulay Driss Rahmani** et Madame **Salma Mouline**, qui ont accepté de juger ce travail.

Nos plus vifs remerciements s'adressent aussi nos professeurs qui nous devons toutes les connaissances acquises au cours de notre formation à la Faculté des Sciences de Rabat et pour les connaissances qu'ils nous ont transmises.

Nos remerciements à nos chers parents, pour leur patience et leur soutien tout au long de nos parcours pédagogiques.

Nos remerciements vont enfin à toute personne qui a contribué de près ou de loin à l'élaboration de ce travail.

Résumé

L'approvisionnement domestique est un terme critique dans la vie de l'être humain, car au cours de notre vie quotidienne, nous consommons une multitude d'éléments dans diverses catégories : alimentaire, santé, ménage, etc. Lors de ce processus, il y a des produits qui sont achetés de nouveau, des produits qui expirent, etc. Ainsi, une personne souhaite gérer ses courses pour savoir quels produits sont consommés, expirés, et aussi savoir où elle peut acheter des produits à moindre coût, afin de gérer de manière efficace sa santé et son économie.

L'objectif principal de ce projet de fin d'études est de développer une application web pour la gestion de l'approvisionnement domestique alors que cette application permettra aux utilisateurs de gérer leur stock de manière simple et efficace. Elle offrira une interface conviviale pour enregistrer les produits et appliquer différentes fonctionnalités comme : la visualisation du stock, des analyses graphiques des courses, et des notifications d'alerte.

Concernant la réalisation, cette application utilise le langage PHP côté serveur, ainsi que les langages HTML5, CSS et JavaScript pour le développement frontend. La bibliothèque jQuery et la bibliothèque Chart.js ont été utilisées pour la création des graphiques. En termes de frameworks, nous avons utilisé Bootstrap pour le design et la mise en page. Par ailleurs, MySQL a été choisi comme système de gestion de base de données (SGBD) et le langage SQL pour la création et l'interrogation de la base de données.

Mots clés : PHP, MySql, Gestion d'approvisionnement domestique, JavaScript, Diagrammes UML

Abstract

Domestic supply management is a critical term in human life, as in our daily lives, we consume a multitude of items in various categories : food, health, household, etc. During this process, some products are repurchased, some expire, and so on. Thus, a person wishes to manage their shopping to know which products are consumed, which have expired, and also to find out where to buy products at the lowest cost, in order to efficiently manage their health and economy.

The main objective of this final year project is to develop a web application for managing domestic supply. This application will allow users to manage their stock in a simple and efficient manner. It will offer a user-friendly interface for recording products and applying different functionalities, such as stock visualization, graphical analyses of purchases, and alert notifications.

Regarding the implementation, this application uses the PHP language on the server side, as well as HTML5, CSS, and JavaScript for frontend development. The jQuery library and the Chart.js library were used for creating the graphs. In terms of frameworks, we used Bootstrap for design and layout. Additionally, MySQL was chosen as the database management system (DBMS) and SQL language for database creation and querying.

Keywords : PHP, MySql, Domestic Supply Management, JavaScript, UML Diagrams

Table des matières

Remerciements **1**

Résumé **2**

Abstract **3**

Introduction **7**

1 Presentation générale du projet **1**
1.1 Contexte du projet 1
1.2 Problématique 1
1.3 Solution proposée : **Provision Rapide** 2
1.4 Analyse des besoins 3
1.4.1 Les besoins fonctionnels 3
1.4.2 Les besoins non fonctionnels 4

2 Analyse conceptuelle et modélisation **6**
2.1 UML 6
2.1.1 Présentation d'UML 6
2.1.2 Définition des diagrammes utilisés 6
2.2 Diagramme de cas d'utilisation de l'application **Provision Rapide** . . . 7
2.2.1 Identification des Acteurs 8
2.2.2 Diagramme de cas d'utilisation 9
2.2.3 Description des cas d'utilisation 10
2.3 Diagramme de classes de l'application **Provision Rapide** 12
2.4 Diagrammes de séquence de l'application **Provision Rapide** 13
2.4.1 Diagramme de séquence « authentification » 13
2.4.2 Diagramme de séquence « ajouter client » 14

3 Environnement technique du projet **15**
3.1 Outils utilisés 15
3.1.1 Editeur VS Code 15
3.1.2 PhpMyAdmin 15
3.1.3 MySql 16
3.1.4 Apache 16
3.1.5 XAMPP SERVER 16
3.1.6 Modelio 16
3.2 Langages utilisés 17

3.2.1 HTML5 17
3.2.2 CSS 17
3.2.3 PHP 18
3.2.4 SQL 18
3.2.5 JAVASCRIPT 18
3.3 Bibliothèques 19
3.3.1 Chart JS 19
3.3.2 jQuery 19
3.4 Framework : Bootstrap 19

4 Développement et Réalisation de l'Application PROVISION RAPIDE 20
4.1 Inroduction 20
4.2 Interface visiteur 20
4.2.1 Page home 20
4.2.2 Page connection 21
4.2.3 Page inscription 22
4.3 Interface utilisateur 22
4.3.1 Tableau de bord 22
4.3.2 Enregistrer un achat 23
4.3.3 Gérer les categories 24
4.3.4 Gérer le stock 25
4.3.5 Analyses graphiques 27
4.3.6 Notifications 28
4.4 Interface administrateur 29
4.4.1 Tableau de bord 29
4.4.2 Ajouter un utilisateur 29
4.4.3 Gérer les utilisateurs 30

Conclusion 31

Table des figures

1.2.1 Statistiques sur le gaspillage alimentaire au monde [1] 2

2.2.1 Acteurs du système 8
2.2.2 Diagramme de cas d'utilisation de l'application 9
2.3.1 Diagramme de classe de l'application 12
2.4.1 Diagramme de séquence de l'application - «authentification» 13
2.4.2 Diagramme de séquence de l'application- «ajouter client» 14

4.2.1 Page home 21
4.2.2 Formulaire de connection 21
4.2.3 Formulaire d'inscription 22
4.3.1 Tableau de bord utilisateur 23
4.3.2 Formulaire du dépôt d'achat 23
4.3.3 Détails des catégories 24
4.3.4 Ajouter une catégorie 25
4.3.5 La gestion du stock 25
4.3.6 Modification du quantité 26
4.3.7 Suppression d'un produit 26
4.3.8 Analyses graphiques 1 27
4.3.9 Analyses graphiques 2 27
4.3.10 Exporter les analyses graphiques 28
4.3.11 Les alerts 28
4.4.1 Tableau de bord administrateur 29
4.4.2 Ajouter un utilisateur 30
4.4.3 Gestion d'utilisateur 30

Introduction

L'approvisionnement domestique est un aspect fondamental de la vie quotidienne , avec la diversité des produits que nous consommons , qu'ils soient : alimentaires, sanitaires, etc... . Il devient crucial de pouvoir gérer efficacement ses achats et son stock. La gestion manuelle de ces produits peut s'avérer fastidieuse et sujette à des erreurs, rendant nécessaire le développement d'outils automatisés pour faciliter cette tâche.

Actuellement, beaucoup de personnes rencontrent des difficultés à suivre l'état de leur stock domestique. Les produits peuvent expirer, être oubliés, ou achetés en double par manque de suivi adéquat. De plus, il est souvent difficile de comparer les prix entre différents magasins pour réaliser des économies. Ces problématiques soulignent l'importance de disposer d'une solution intégrée permettant une gestion optimale de l'approvisionnement domestique.

L'objectif principal de ce projet de fin d'études est de développer une application web dédiée à la gestion de l'approvisionnement domestique que nous avons nommé **Provision Rapide**. Cette application vise à :

- Permettre aux utilisateurs de gérer leur stock de manière simple et efficace.
- Offrir une interface conviviale pour enregistrer les produits.
- Proposer des fonctionnalités de visualisation du stock, des analyses graphiques des achats et des notifications d'alerte.

Ce rapport se divise en quatre parties principales :

Dans le premier chapitre, nous allons d'abord présenter une introduction sur l'aspect de la gestion d'approvisionnement domestique. Ensuite, nous aborderons le contexte du projet ainsi que ses objectifs généraux et l'importance de ce dernier. Enfin, nous dévoilerons le cahier des charges en énumérant les besoins concernés.

Au second chapitre, nous donnerons une récapitulation des diagrammes utilisés dans la conception de ce projet. Ensuite, nous identifierons les acteurs impliqués. Enfin, nous réaliserons la modélisation des diagrammes.

Concernant le troisième chapitre, nous fournissons un résumé des langages et des outils utilisés pour la réalisation du projet, en veillant à ce qu'ils soient compatibles avec nos connaissances.

Enfin, nous présentons les résultats que nous avons obtenus, en les illustrant à l'aide de captures d'écran des interfaces présentes dans l'application.

Chapitre 1

Presentation générale du projet

1.1 Contexte du projet

La gestion d'approvisionnement domestique n'est pas uniquement une préoccupation pour les entreprises elle est également essentielle dans le cadre domestique, La gestion des biens domestiques permet d'optimiser l'utilisation des ressources, permet de réduire le gaspillage et d'assurer que les articles nécessaires sont toujours disponibles, Donc Une application de gestion d'approvisionnement domestique peut simplifier ce processus rendant la vie quotidienne plus organisée et efficace.

1.2 Problématique

Dans un monde de plus en plus axé sur l'efficacité et la durabilité alors que la gestion domestique n'échappe pas à cette tendance bien que Les ménages modernes sont souvent confrontés à des défis liés à la gestion des biens de consommation courante : nourriture, produits ménagers, vêtements, etc. Le gaspillage alimentaire, le désordre et aussi la gestion inefficace des ressources sont des problèmes courants qui affectent à la fois le budget des ménages et l'environnement.

Les statistiques montrent que le gaspillage alimentaire est un problème majeur avec environ un tiers de la nourriture produite dans le monde étant perdue ou gaspillée donc cela représente non seulement un problème éthique et environnemental, mais aussi une perte économique significative pour les ménages et de plus, la mauvaise gestion des biens ménagers peut entraîner des dépenses inutiles et un désordre qui impacte la qualité de vie [1] .

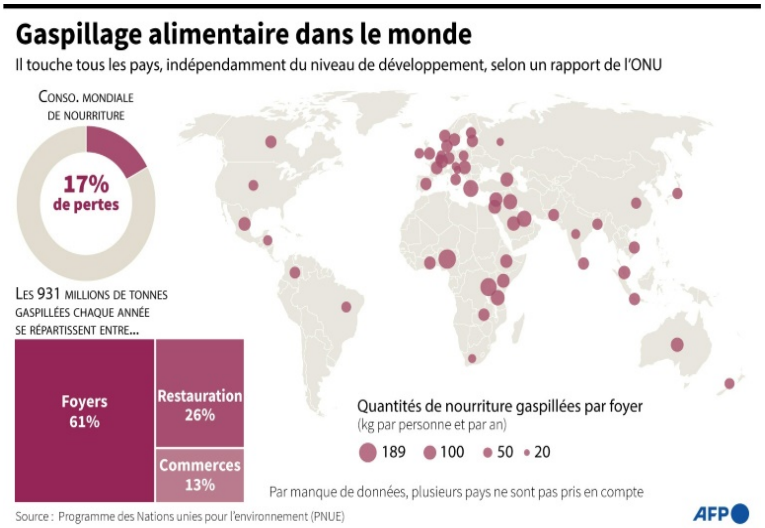

Figure 1.2.1 – Statistiques sur le gaspillage alimentaire au monde [1]

1.3 Solution proposée : Provision Rapide

Le projet de développement d'une application de gestion d'approvisionnement domestique vise à résoudre les défis auxquels les gens sont confrontés dans leur vie quotidienne. Il propose une solution technologique simple et efficace pour les foyers, visant à réduire le gaspillage alimentaire et à améliorer leurs économies financières.

Réduction du gaspillage alimentaire :

Le gaspillage alimentaire est un problème global aux répercussions environnementales, économiques et sociales significatives donc À l'échelle mondiale, environ un tiers de la nourriture produite est perdue ou gaspillée chaque année par conséquent pour les ménages, cela représente non seulement une perte financière mais aussi une inefficacité dans la gestion des ressources, il est donc crucial de mettre en œuvre des stratégies efficaces pour réduire le gaspillage alimentaire.

Économies financières :

Le gaspillage alimentaire représente une perte financière considérable pour les ménages. Selon des études, une famille moyenne peut gaspiller des centaines de dollars par an en nourriture non consommée alors que réduire ce gaspillage peut donc entraîner des économies substantielles et bien que adopter des pratiques efficaces de gestion alimentaire permet de maximiser l'utilisation des ressources, d'optimiser les dépenses et de soutenir la durabilité économique du foyer.

Les objectifs de cette application sont les suivants :

- Permettre à l'utilisateur de suivre ses achats d'articles domestiques.
- Mémoriser les prix les plus élevés et les moins élevés des articles achetés.
- Enregistrer la date et le magasin d'achat pour chaque article.
- Gérer le stock des articles domestiques.
- Fournir des analyses graphiques sur les articles les plus achetés, les marques les plus achetées, et les quantités par période.
- Permettre à l'utilisateur d'exporter les données pour une utilisation externe.

1.4 Analyse des besoins

1.4.1 Les besoins fonctionnels

Pour développer une application de gestion de stock domestique efficace, il est essentiel de définir clairement les besoins fonctionnels. Ces besoins décrivent les fonctionnalités spécifiques que l'application doit offrir pour répondre aux attentes des utilisateurs et aux objectifs du projet.

1. Gestion des Inventaires :

- **Enregistrement des Produits** : Les utilisateurs doivent pouvoir ajouter de nouveaux produits à l'inventaire avec des détails tels que le nom, la catégorie, la date d'achat, la date de péremption, et la quantité.
- **Mise à Jour des Stocks** : Les utilisateurs doivent pouvoir modifier les informations des produits existants (quantité, date de péremption, etc.) et supprimer des produits de l'inventaire.
- **Catégorisation des Produits** : Les produits doivent pouvoir être classés par catégories (par exemple, alimentation, produits ménagers, articles de toilette) pour une organisation facile.

2. Alertes et Notifications :

- **Notifications de Péremption** : L'application doit envoyer des notifications lorsque des produits approchent de leur date de péremption pour éviter le gaspillage.
- **Rappels de Réapprovisionnement** : Les utilisateurs doivent recevoir des rappels pour réapprovisionner certains produits lorsque les stocks sont bas, basés sur des seuils définis par l'utilisateur.

3. Génération de Listes de Courses :

- **Listes de Courses Automatiques** : L'application doit générer automatiquement des listes de courses basées sur les besoins identifiés par l'inventaire et la planification des repas.
- **Partage de Listes** : Les utilisateurs doivent pouvoir partager leurs listes de courses avec d'autres membres de la famille ou les imprimer.

4. Suivi et Analyses :

— **Rapports sur le Gaspillage** : L'application doit offrir des rapports détaillés sur les produits jetés, incluant les raisons (périmé, inutilisé) pour analyser les tendances et améliorer la gestion future.
— **Statistiques de Consommation** : L'application doit fournir des statistiques sur les habitudes de consommation des utilisateurs, aidant à identifier les produits les plus et les moins utilisés.

5. Interface Utilisateur :
— **Tableau de Bord** : Un tableau de bord convivial doit présenter un aperçu rapide de l'état des stocks, des alertes en attente, et des tâches à accomplir.
— **Recherche et Filtrage** : Les utilisateurs doivent pouvoir rechercher des produits et filtrer les résultats par catégorie, date de péremption, etc.
— **Interface Multi-utilisateur** : L'application doit permettre à plusieurs utilisateurs d'accéder à un même compte pour faciliter la gestion collaborative des stocks domestiques.

1.4.2 Les besoins non fonctionnels

Les besoins non fonctionnels décrivent les critères de performance, de sécurité, de convivialité, et autres caractéristiques essentielles pour assurer le bon fonctionnement et l'adoption de l'application.

Ci-après, nous citons les principaux besoins non fonctionnels pour une application de gestion de stock domestique :

1. Performance :
— **Temps de Réponse** : L'application doit répondre aux actions des utilisateurs en moins de 2 secondes pour la majorité des opérations (ajout de produits, mise à jour des stocks, génération de listes).
— **Scalabilité** : L'application doit pouvoir gérer une augmentation du nombre d'utilisateurs et de la quantité de données sans dégradation significative des performances.

2. Sécurité :
— **Authentification et Autorisation** : Les utilisateurs doivent s'authentifier pour accéder à l'application et les permissions doivent être gérées pour restreindre l'accès aux fonctionnalités sensibles.
— **Protection des Données** : Les données des utilisateurs, y compris les informations personnelles et de stock, doivent être protégées par des mesures de cryptage, tant au repos qu'en transit.

3. Fiabilité :

— **Disponibilité** : L'application doit être disponible 99.9% du temps pour garantir un accès continu aux utilisateurs.

— **Tolérance aux Pannes** : L'application doit être capable de fonctionner correctement même en cas de pannes partielles du système.

4. Convivialité :

— **Interface Utilisateur** : L'application doit avoir une interface intuitive, facile à utiliser pour tous les niveaux de compétence technique des utilisateurs.

Chapitre 2

Analyse conceptuelle et modélisation

2.1 UML

2.1.1 Présentation d'UML

UML (Unified Modeling Language) est un langage de modélisation objet qui est adopté par les grandes sociétés. Il est très utile pour la conception dans un environnement de programmation et la mise en œuvre de systèmes logiciels. UML se compose d'une part des éléments de modélisation qui représentent toutes les propriétés des langages et d'autre part des diagrammes tels que : cas d'utilisation, classes, états-transitions, séquence, etc. Ainsi, ce dernier est conçu pour modéliser les différents types de systèmes pour tous les domaines (temps réel, gestion, ...) [2].

2.1.2 Définition des diagrammes utilisés

Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation est un diagramme UML qui représente le comportement fonctionnel d'un système. Ce comportement est modélisé par une notation très simple et compréhensible. Elle réalise l'ensemble des interactions entre les utilisateurs et le système à travers des cas d'utilisation, dont les utilisateurs sont appelés des acteurs et le système contient des cas d'utilisation. Ainsi, ce dernier a comme objectif une modélisation de la vue des utilisateurs. Il permet également la structuration des besoins ou bien du cahier de charge [3].

Diagramme de classes

Le diagramme de classes est un diagramme UML qui est une représentation abstraite des objets du système qui vont interagir ensemble pour la réalisation des cas d'utilisation. En effet, il est considéré comme une vue statique qui ne prend pas en compte le facteur temporel dans le comportement du système. Ainsi, il permet de modéliser des classes et leurs relations indépendamment d'un langage de programmation précis [4].

— Généralisation, association et dépendances sont les principales relations des classes.
— Une classe est un concept abstrait qui représente une variété d'éléments du système, dont une classe peut être associée à d'autres classes.

Diagramme de séquence

Un diagramme de séquence est un diagramme UML qui représente les interactions entre les acteurs et les objets, dont le comportement est capturé dans un seul cas d'utilisation. Un diagramme de séquence représente un certain nombre d'objets et les messages transmis entre ces objets dans la réalisation du cas d'utilisation. Ainsi, ce dernier prend en considération le facteur temporel [5].

— Chaque objet est symbolisé par une barre verticale appelée **ligne de vie**.
— Le temps s'écoule de haut en bas, de sorte que la numérotation des messages est optionnelle.

2.2 Diagramme de cas d'utilisation de l'application Provision Rapide

Le diagramme de cas d'utilisation représente ce qu'un système est supposé faire. Un cas d'utilisation représente une interaction entre un utilisateur et un système, les utilisateurs sont appelés acteurs, ils interagissent avec les cas d'utilisation.

2.2.1 Identification des Acteurs

Un acteur est une entité abstraite qui incarne un rôle joué par des utilisateurs, des dispositifs matériels ou d'autres systèmes externes. Ces acteurs interagissent directement avec le système étudié en consultant ou en modifiant son état. Ils communiquent en émettant ou en recevant des messages, parfois accompagnés de données.

- **User {abstract} :** c'est un acteur qui n'existe pas réellement, mais il est utilisé dans la conception pour englober les cas d'utilisation faites par plusieurs acteurs.
- **Client :** il peut effectuer plusieurs fonctionnalités dans l'application après avoir été authentifié, par exemple il peut ajouter des achats, visualiser des analyses graphiques, etc.
- **Administrateur :** c'est l'administrateur de l'application, il a le droit de gérer les utilisateurs, mais il n'a pas le droit d'accéder au stock des utilisateurs.

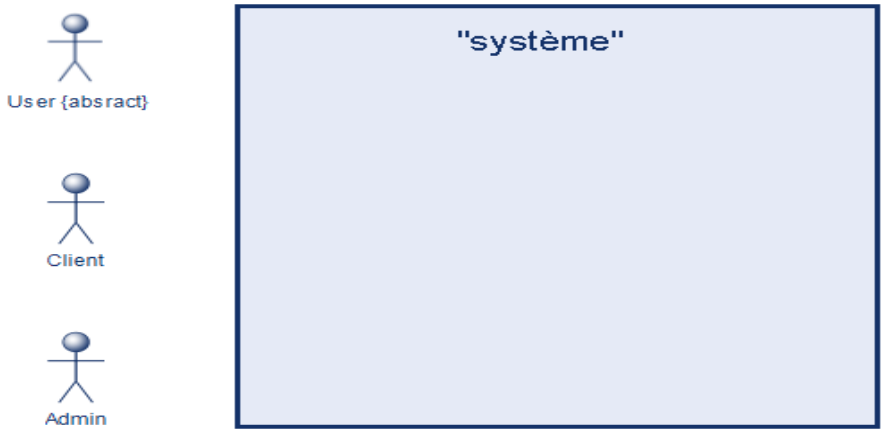

FIGURE 2.2.1 – Acteurs du système

2.2.2 Diagramme de cas d'utilisation

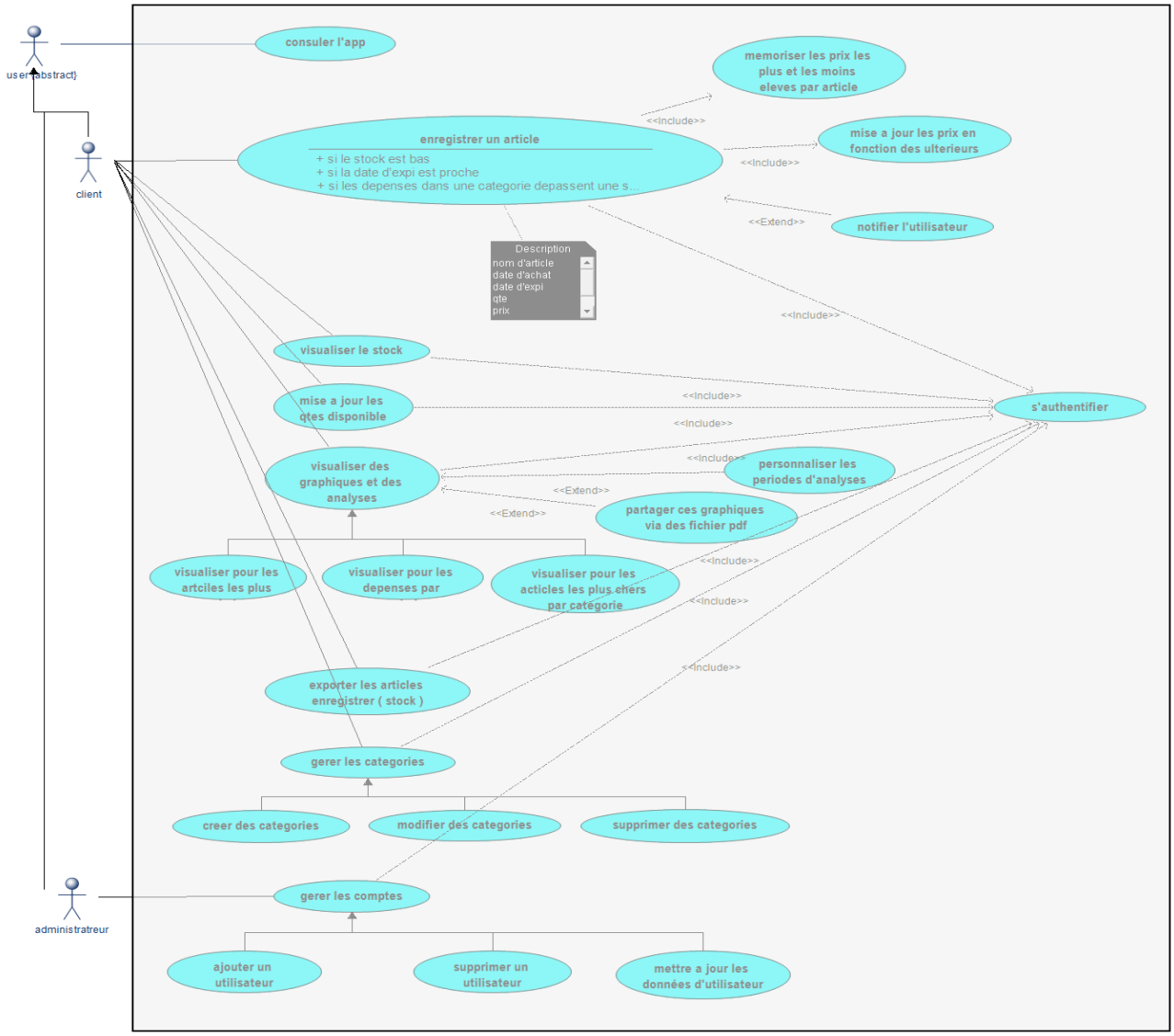

FIGURE 2.2.2 – Diagramme de cas d'utilisation de l'application

2.2.3 Description des cas d'utilisation

Cas d'utilisation « enregistrer un achat »

Nom du cas d'utilisation	"Enregistrer un achat."
Brève description.	L'utilisateur permet d'enregistrer un achat à l'aide d'un formulaire affiché à son interface.
Pré-condition.	L'utilisateur doit être authentifié.
Post-condition.	Enregistrement d'un achat dans la base de données avec des mises à jour correspondantes.
Situation d'erreur.	L'un des champs du formulaire n'est pas rempli, ou l'achat existe déjà.
État du système lors de l'apparition d'une erreur.	Afficher à l'utilisateur un message d'erreur.
Acteurs.	Client.
Déclencheur.	La clique sur le bouton d'ajout.
Scénario.	Le client se connecte, remplit le formulaire puis clique sur le bouton d'ajout d'un achat.

Cas d'utilisation « mettre à jour les quantités »

Nom du cas d'utilisation	"Mettre à jour les quantités."
Brève description.	L'utilisateur permet de modifier les quantités disponibles d'achat à l'aide d'une case affichée dans le tableau du stock.
Pré-condition.	L'utilisateur doit être authentifié.
Post-condition.	Mise à jour des quantités au niveau de la base de données.
Situation d'erreur.	La quantité est inférieure à 1.
État du système lors de l'apparition d'une erreur.	Afficher à l'utilisateur un message d'erreur.
Acteurs.	Client.
Déclencheur.	La clique sur le bouton de modification.
Scénario.	Le client se connecte, modifie le champ de quantité, puis clique sur le bouton de modification.

Cas d'utilisation « supprimer une catégorie »

Nom du cas d'utilisation	"Supprimer une catégorie."
Brève description.	L'utilisateur permet de supprimer une catégorie disponible dans le stock.
Pré-condition.	L'utilisateur doit être authentifié.
Post-condition.	Supprimer la catégorie au niveau de la base de données.
Situation d'erreur.	Le nombre de produits dans la catégorie est supérieur à 1.
État du système lors de l'apparition d'une erreur.	Afficher à l'utilisateur un message d'erreur.
Acteurs.	Client.
Déclencheur.	La clique sur le bouton de suppression.
Scénario.	Le client se connecte et clique sur le bouton de suppression.

Cas d'utilisation « ajouter un utilisateur »

Nom du cas d'utilisation	"Ajouter un utilisateur."
Brève description	L'administrateur permet d'ajouter un utilisateur à l'aide d'un formulaire affiché à son interface.
Pré-condition	L'administrateur doit être authentifié.
Post-condition	L'ajout d'un utilisateur dans la base de données.
Situation d'erreur	L'un des champs du formulaire n'est pas rempli, ou l'utilisateur existe déjà.
État du système lors de l'apparition d'une erreur	Afficher à l'administrateur un message d'erreur.
Acteurs	Administrateur.
Déclencheur	La clique sur le bouton d'ajout.
Scénario	L'administrateur se connecte, remplit le formulaire, puis clique sur le bouton d'ajout.

2.3 Diagramme de classes de l'application Provision Rapide

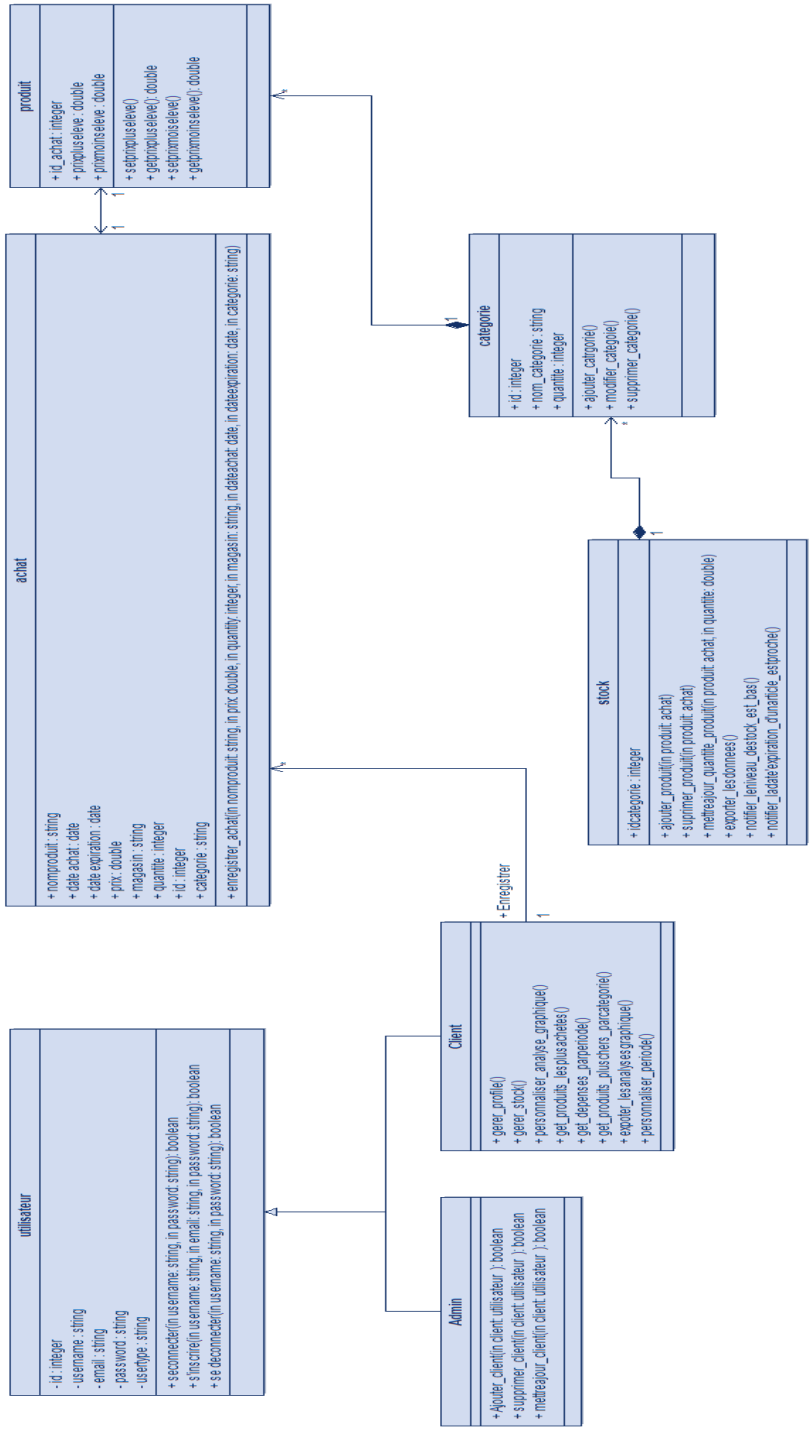

FIGURE 2.3.1 – Diagramme de classe de l'application

2.4 Diagrammes de séquence de l'application Provision Rapide

Le diagramme de séquence permet de montrer les interactions des objets dans le cadre d'un scénario d'un diagramme de cas d'utilisation.

2.4.1 Diagramme de séquence « authentification »

L'utilisateur va s'authentifier d'abord par le remplissage du formulaire d'authentification. Le système va tester l'existence de cet utilisateur en effectuant une vérification au niveau de l'entité des utilisateurs. Si c'est le cas, la page de l'utilisateur va être affichée ; sinon, un message d'erreur va être affiché à l'utilisateur.

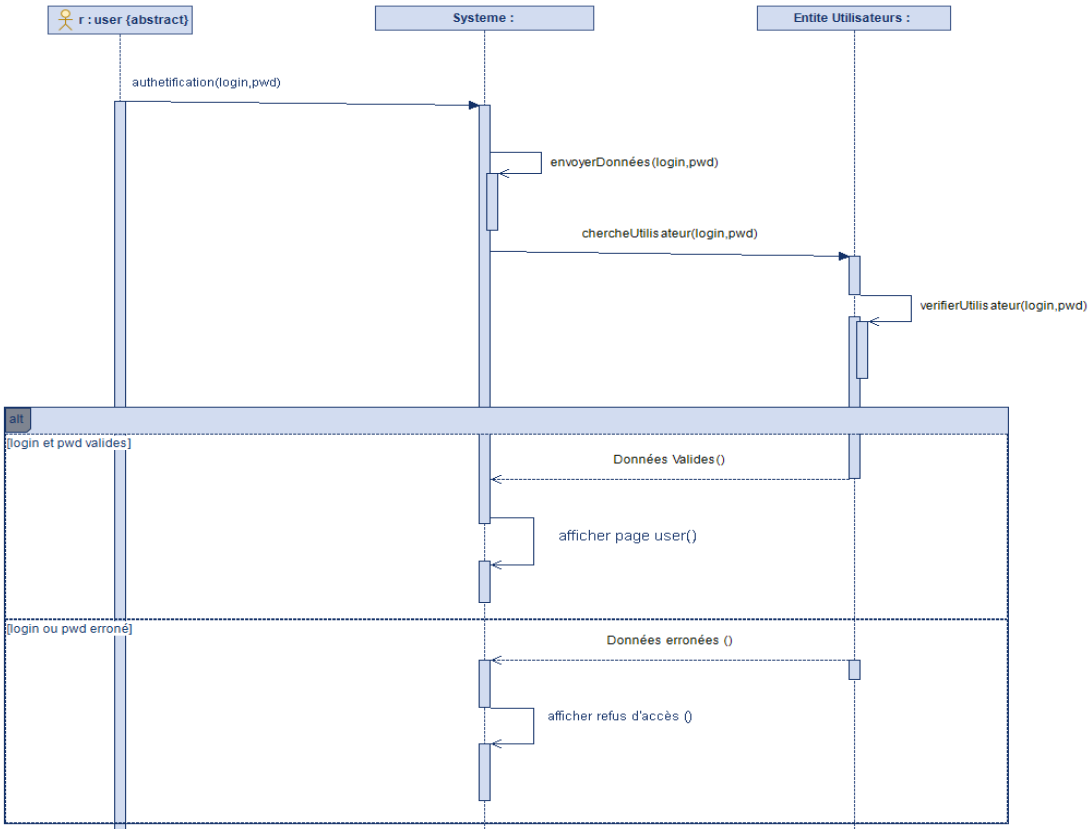

FIGURE 2.4.1 – Diagramme de séquence de l'application - «authentification»

2.4.2 Diagramme de séquence « ajouter client »

L'administrateur va saisir les informations du client à ajouter. Ensuite, ces informations vont être insérées dans l'entité client. Si tous les champs ont été remplis, un message de succès va être affiché. Si ce n'est pas le cas, un message d'erreur va être affiché.

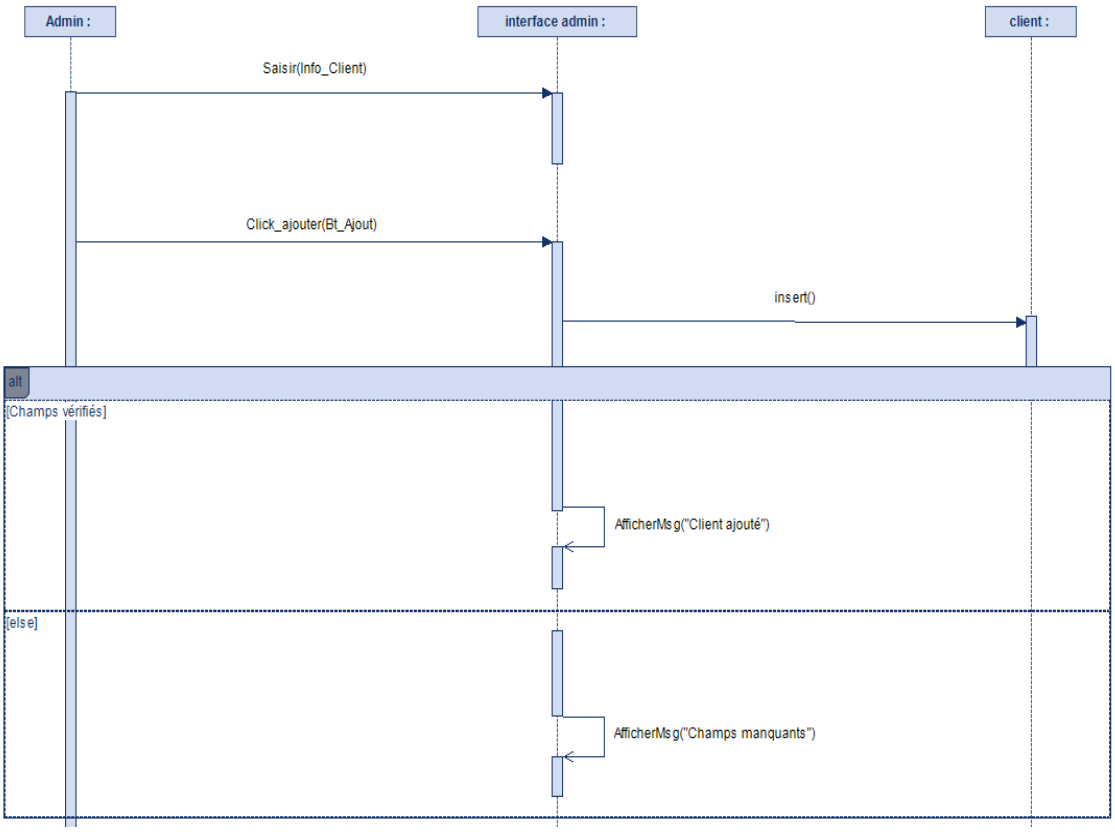

FIGURE 2.4.2 – Diagramme de séquence de l'application- «ajouter client»

Chapitre 3

Environnement technique du projet

Dans ce chapitre sont présenté les différents langages de programmation ainsi que d'autres outils de développement utilisés pour la réalisation du notre application.

3.1 Outils utilisés

3.1.1 Editeur VS Code

Visual Studio Code (VSCode) est un éditeur de code source et un environnement de développement intégré (IDE) créé par Microsoft C'est un logiciel open-source et compatible avec différentes plateformes comme Windows, Linux et macOS donc initialement pensé pour les développeurs web il supporte également de nombreux autres langages de programmation tels que C++, C#, Python, Java, entre autres et parmi ses fonctionnalités principales on peut citer la coloration syntaxique, l'auto-complétion, la détection d'erreurs, la navigation dans le code, le débogage, la gestion de versions et l'intégration avec Git et de plus, VSCode est extensible grâce à une large gamme d'extensions développées par la communauté, offrant ainsi aux développeurs la possibilité de personnaliser l'éditeur selon leurs besoins spécifiques [4].

3.1.2 PhpMyAdmin

PHPMyAdmin est un outil extrêmement efficace conçu pour simplifier **l'administration des bases de données MySQL** son installation et sa configuration sont simples il offre aux développeurs et aux administrateurs système une solution conviviale et puissante pour gérer efficacement les bases de données. C'est l'un des outils les plus appréciés qui permet aux utilisateurs de créer, modifier et interroger des données au sein de bases de données MySQL [7].

3.1.3 MySql

MySQL est un système de gestion de base de données relationnelle **(SGBDR)** open source, reposant sur un modèle client-serveur et utilisant le langage de requête structuré SQL.

MySQL est largement réputé comme étant le système de gestion de bases de données relationnelles le plus utilisé au monde. Il est adopté par des géants tels que Facebook, Google, Adobe, ainsi que par de nombreux autres sites web et applications en ligne.

Développé, distribué et soutenu par Oracle Corporation, MySQL est une solution multiplateforme, robuste, flexible et extensible, fondée sur le langage SQL standardisé. Ce dernier est employé pour la création et la manipulation de bases de données relationnelles [8].

3.1.4 Apache

Apache HTTP Server, couramment désigné sous le nom d'Apache, est un serveur HTTP open source compatible avec les systèmes d'exploitation Unix, Windows, MacOS et Linux. Il figure parmi les serveurs HTTP les plus largement utilisés sur internet, servant de nombreuses plateformes web. Apache offre une grande flexibilité de configuration et de personnalisation grâce à ses fichiers de configuration et à ses modules logiciels additionnels. Il prend en charge divers langages de programmation web tels que PHP, Perl et Python, et se distingue par sa capacité à gérer efficacement des sites web à forte charge grâce à son architecture modulaire. [9].

3.1.5 XAMPP SERVER

XAMPP est un ensemble de logiciels qui permet de configurer facilement un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Cette distribution de logiciels libres (**X** cross, **Apache**, **MariaDB** ou **MySQL**, **Perl**, **PHP**) est appréciée pour son installation rapide et simple. Accessible à un large public, XAMPP ne requiert pas de compétences techniques avancées et fonctionne sur les principaux systèmes d'exploitation[10].

3.1.6 Modelio

Nous avons choisi pour la modélisation UML de l'application Modelio qui se caractérise par sa simplicité d'installation et de prise en main . C'est un outil de création de modèles [11].

Modelio est un éditeur graphique avancé, conçu comme une plateforme extrêmement personnalisable. Il permet de créer et de définir facilement vos propres langages de modélisation et notations, tout en développant des fonctionnalités spécifiques :

— Personnalisation approfondie et extension de **UML**, BPMN, méta-modèles d'exigences et profils standard tels que UPDM, TOGAF, SysML, etc.

— Création de nouvelles propriétés, types de diagrammes, matrices, options de menu, modèles de documents et automatisation de motifs.

— Intégration de code d'extension (Java, Jython) pour une flexibilité maximale.

— Développement et déploiement de modules d'extension sur mesure pour répondre aux besoins spécifiques.

3.2 Langages utilisés

3.2.1 HTML5

Le HTML (Langage de Marquage Hypertexte) est un langage de programmation web essentiel pour ajouter du contenu à une page web et constitue la base de la création de sites internet chaque page web est fondamentalement un fichier HTML en plus Le code HTML permet de définir le contenu qui sera affiché dans un navigateur alors ce contenu est interprété par les navigateurs et il est également analysé par les robots des moteurs de recherche pour extraire le contenu d'une page web d'où l'appellation « page HTML », il se distingue par l'utilisation de balises qui délimitent et caractérisent le contenu de la page avec ces balises HTML et leurs attributs permettent de structurer le contenu (en blocs, niveaux de titres, liens internes et externes, paragraphes, éléments multimédias (comme les images)). Ainsi, le HTML est indispensable pour créer une page web cependant il se limite à la définition du contenu en effet la mise en forme et les animations du contenu sont prises en charge par d'autres langages complémentaires tels que CSS pour le style et JavaScript pour l'interactivité [12].

3.2.2 CSS

Le CSS (Cascading Style Sheets, ou feuilles de style en cascade) est un langage de programmation essentiel pour la mise en forme des pages web codées en HTML fonctionnant de manière complémentaire aux langages de balisage tels que HTML et XML alors le CSS permet de définir des règles de style applicables simultanément à plusieurs éléments ou documents il améliore ainsi l'esthétique et la présentation visuelle d'un site web

.

Avec le CSS vous pouvez appliquer des styles (titrage, alignement, polices, couleurs, bordures, etc.) de manière cohérente à différents

documents cela permet une gestion centralisée des styles aussi facilitant les modifications et l'uniformité visuelle, en matière de conception de pages web le CSS sépare la présentation visuelle de la structure du contenu HTML permettant une plus grande flexibilité et maintenabilité plus les standards du CSS sont définis par le World Wide Web Consortium (W3C), garantissant une utilisation cohérente et compatible à travers différents navigateurs et plateformes [13].

3.2.3 PHP

Le PHP est un langage de scripts donc il est interprété et qui s'exécute sur un serveur alors cela signifie que le code PHP est traité par le serveur avant de générer un document HTML envoyé au navigateur alors que Cette caractéristique le distingue fondamentalement des langages de script comme le JavaScript qui est exécuté directement par le navigateur.

Un des nombreux avantages du PHP est sa capacité à interagir avec divers SGBD (Systèmes de Gestion de Base de Données), notamment le très populaire **MySQL** [14].

3.2.4 SQL

Le langage SQL (**Structured Query Language**) est un outil informatique destiné à la gestion et à la manipulation des bases de données dont il permet de définir, manipuler et sécuriser les données.

En pratique, SQL est utilisé pour créer des tables, pour insérer des enregistrements aussi d'interroger des bases de données, effectuer des mises à jour et gérer les autorisations des utilisateurs, Il est supporté par la majorité des systèmes de gestion de bases de données (SGBD) alors que SQL est aujourd'hui une norme internationale largement reconnue [15].

3.2.5 JAVASCRIPT

JavaScript est un langage de programmation orienté objet largement utilisé pour dynamiser les fonctionnalités interactives des pages web et il est créé par Brendan Eich en 1995, JavaScript se distingue des langages côté serveur en étant exécuté directement par le navigateur de l'utilisateur et plutôt que sur le serveur web. Ainsi, JavaScript est principalement actif côté client, permettant de créer des animations et d'ajouter des effets dynamiques de manière efficace sur les pages HTML [16].

3.3 Bibliothèques

3.3.1 Chart JS

Dans le contexte des analyses graphiques nous avons opté pour la bibliothèque Chart.js en JavaScript alors que cette bibliothèque nous permet de générer des graphiques à partir de données extraites de notre base de données.

Chart.js est une bibliothèque open source et gratuite conçue pour la visualisation de données elle est créée par Nick Downie qui est un développeur web en 2013 et elle est désormais maintenue par la communauté, cette dernier est la deuxième bibliothèque de graphiques JavaScript la plus populaire sur GitHub en termes de nombre d'étoiles juste après D3.js. Les graphiques créés avec Chart.js sont rendus dans un élément canvas HTML5 et en plus La bibliothèque est distribuée sous licence MIT [17].

3.3.2 jQuery

jQuery est une bibliothèque JavaScript gratuite, libre et multiplateforme et créée pour simplifier l'écriture de scripts il est Apparue en 2006, elle est largement reconnue pour sa compatibilité avec tous les principaux navigateurs web (Internet Explorer, Safari, Chrome, Firefox, etc.).

En plus de faciliter la manipulation du code HTML, CSS, JavaScript et AJAX côté client, jQuery agit comme un langage de requête qui permettant une sélection et une manipulation efficaces des éléments du DOM donc c'est un outil essentiel pour améliorer l'interaction et la réactivité des applications web.

- Manipulation avancée du Modèle d'Objet de Document (DOM)
- Gestion dynamique des événements (comme les mouvements de souris, les clics, etc.) et de l'AJAX (Asynchronous JavaScript and XML)
- Création fluide d'effets d'animation interactifs
- Contrôle précis des feuilles de style en cascade

3.4 Framework : Bootstrap

Bootstrap est un framework développé par l'équipe derrière le réseau social Twitter il offert en open source sous licence MIT en utilisant les langages HTML, CSS et JavaScript ce framework fournit aux développeurs une suite d'outils puissants pour simplifier la création de sites web il est spécialement conçu pour assurer un design responsive et il s'adapte parfaitement à tous les types d'écrans en particulier les smartphones. Bootstrap comprend des composants prédéfinis tels que des typographies élégantes, des boutons stylisés, des barres de navigation intuitives, et bien d'autres éléments essentiels à la construction d'interfaces web modernes [19].

Chapitre 4

Développement et Réalisation de l'Application PROVISION RAPIDE

4.1 Inroduction

Après la mise en oeuvre de l'étude et la conception de notre application,nous passons à la phase d'implémentation. ce chapitre présente le résultat du travail effectué durant ce projet de fin d'études. Nous allons présenter, aussi, environnement matériels et les outils de développement utilisés.Nous clôturons ce chapitre par quelques captures d'écran démontrant les fonctionnalités de notre application.

4.2 Interface visiteur

4.2.1 Page home

Afin de répondre aux besoins et attentes de nos clients, nous avons conçu une interface simple à manipulier et à la portée de tous les utilisateurs. Celui-ci a une visibilité sur la page d'accueil et les différentes rubriques comprenant : Home, About, Get Sratred

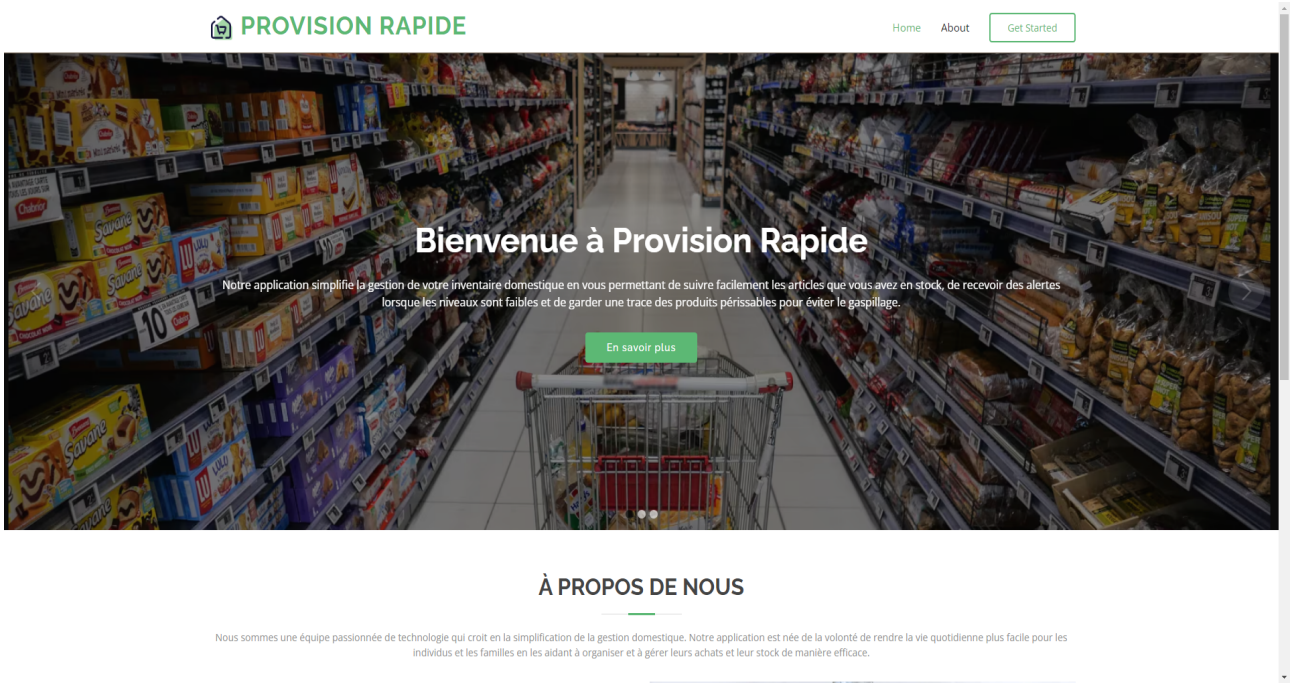

FIGURE 4.2.1 – Page home

4.2.2 Page connection

Une fois l'utilisateur inscrit, il peut se connecter à son compte en insérant son gmail et mot de passe comme le montre la figure qui suit :

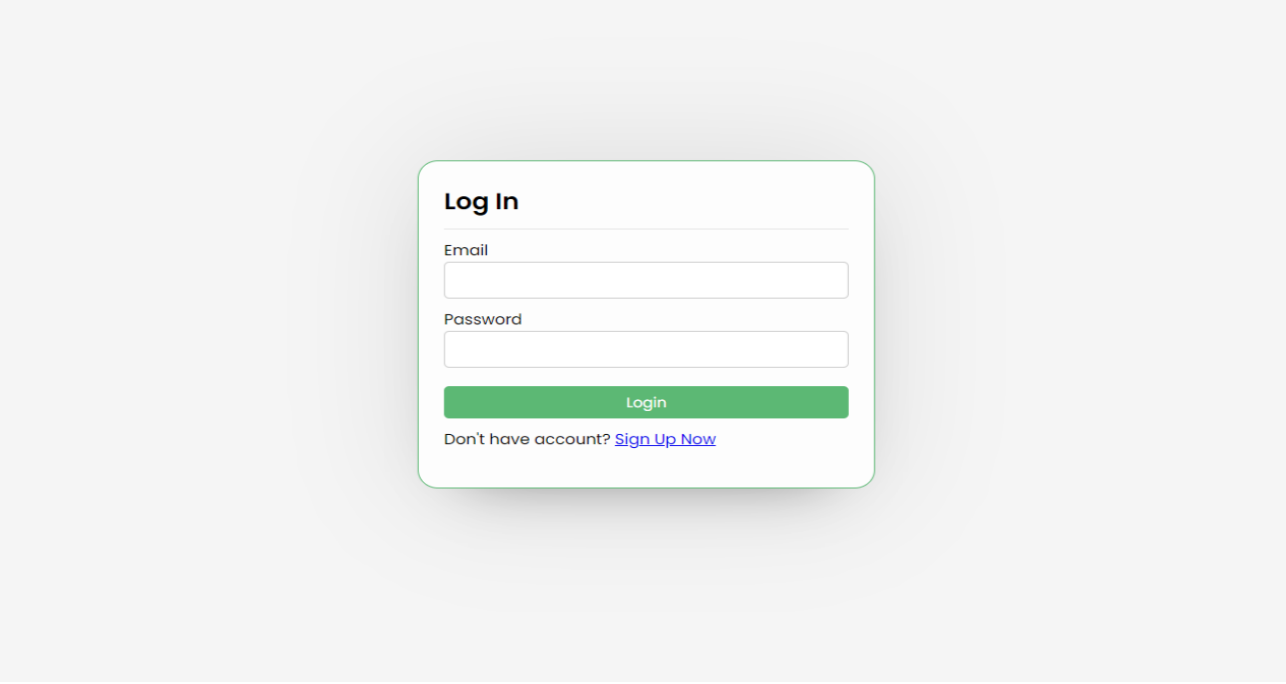

FIGURE 4.2.2 – Formulaire de connection

4.2.3 Page inscription

A partir de la page d'accueil, l'utilisateur peut éventuellement s'inscrire sur notre site afin de pouvoir exploiter les fonctionnalités de l'application en tant que client. La figure ci-dessous montre l'interface d'inscription de l'utilisateur

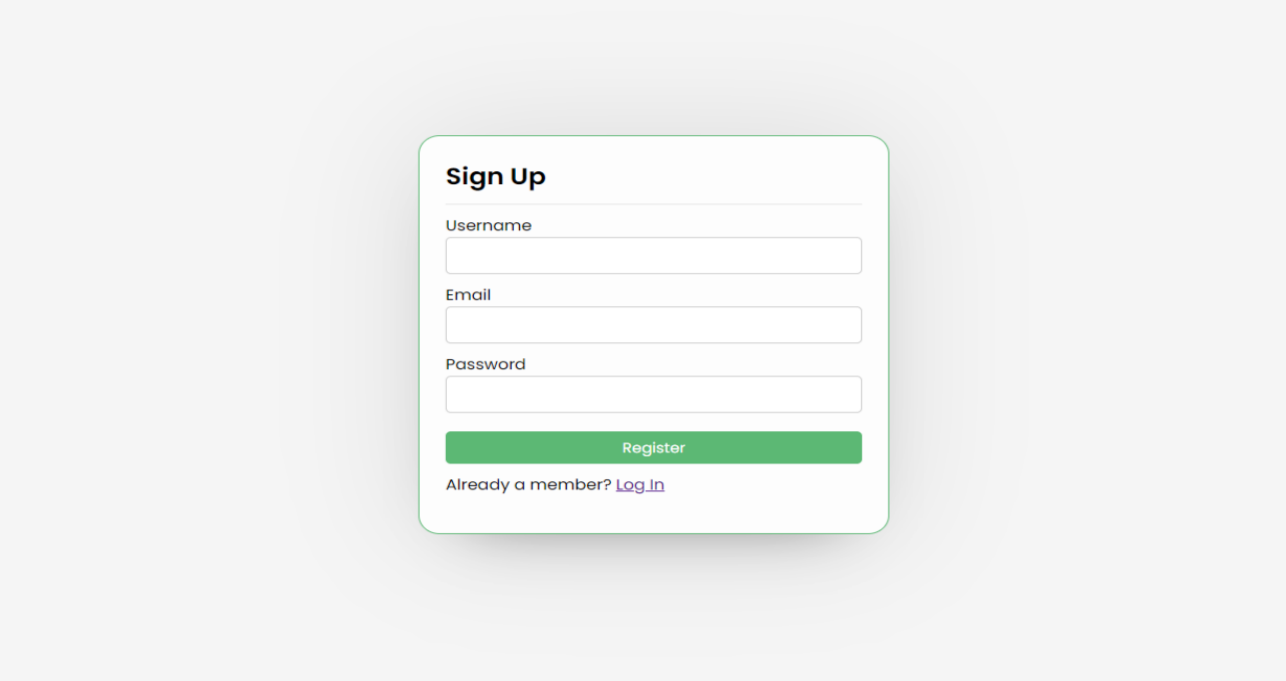

FIGURE 4.2.3 – Formulaire d'inscription

4.3 Interface utilisateur

4.3.1 Tableau de bord

Maintenant que la connexion est établie, le client conviviales et adaptables pour découverte les différents services de la plateforme

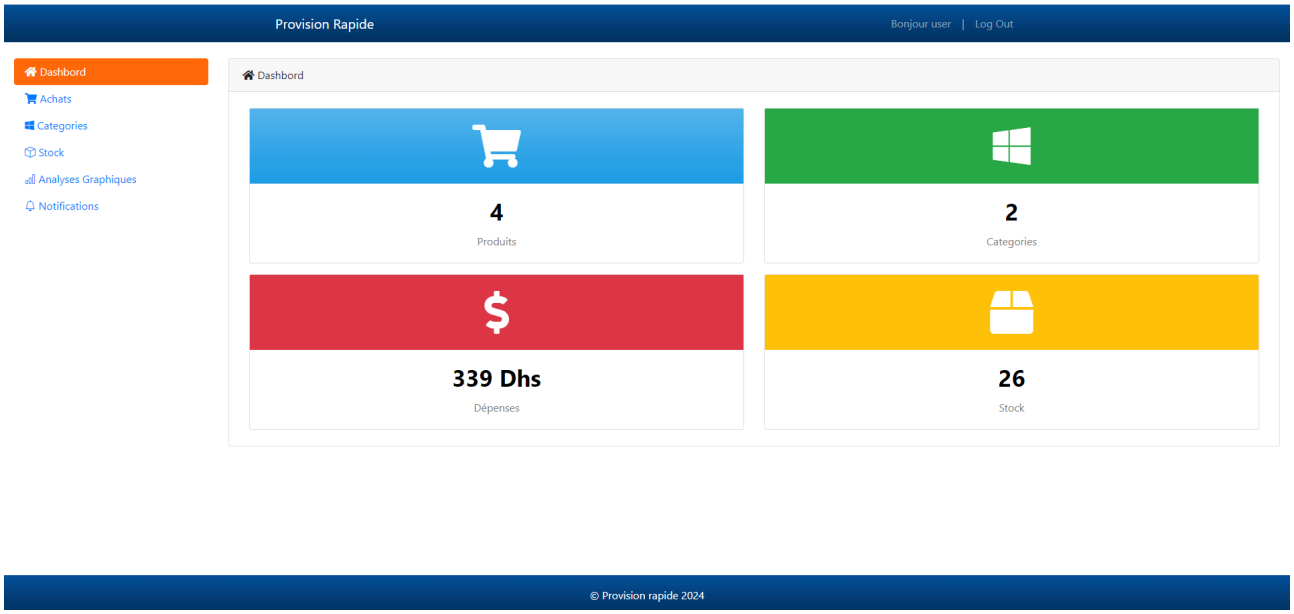

FIGURE 4.3.1 – Tableau de bord utilisateur

4.3.2 Enregistrer un achat

le client peut déposer son achat en remplissant les champs indiqués sur le formulaire représenté sur la figure suivant :

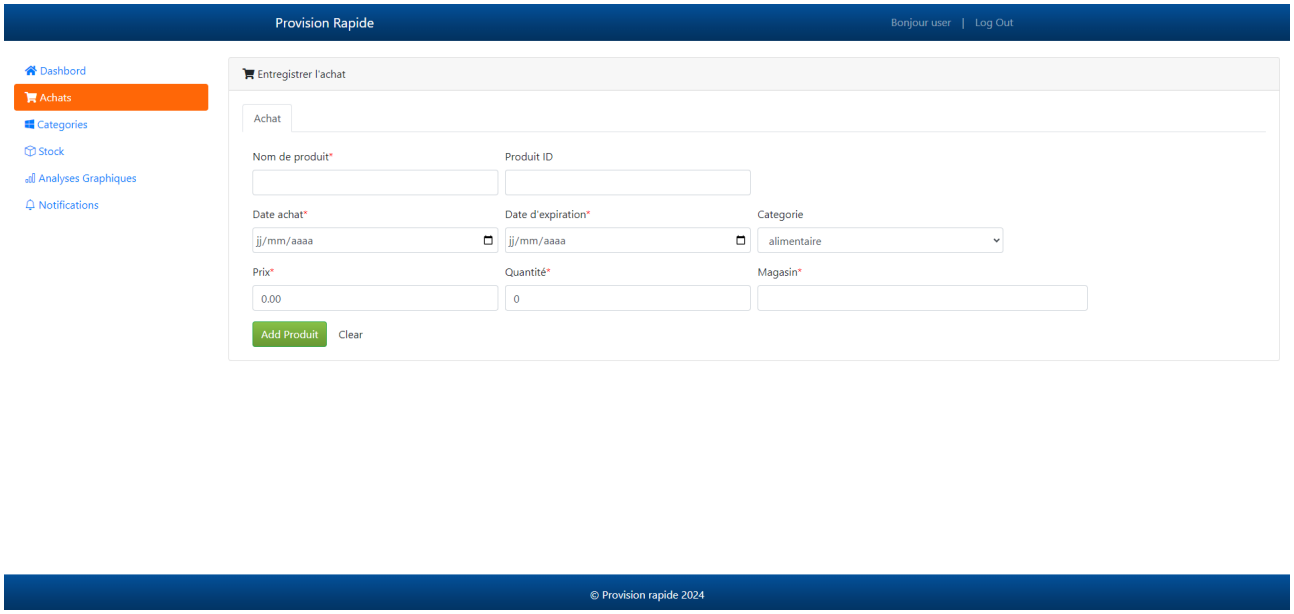

FIGURE 4.3.2 – Formulaire du dépôt d'achat

4.3.3 Gérer les categories

Visualiser les categories

L'utilisateur peut modifier ou supprimer ses catégories selon ses besoins.

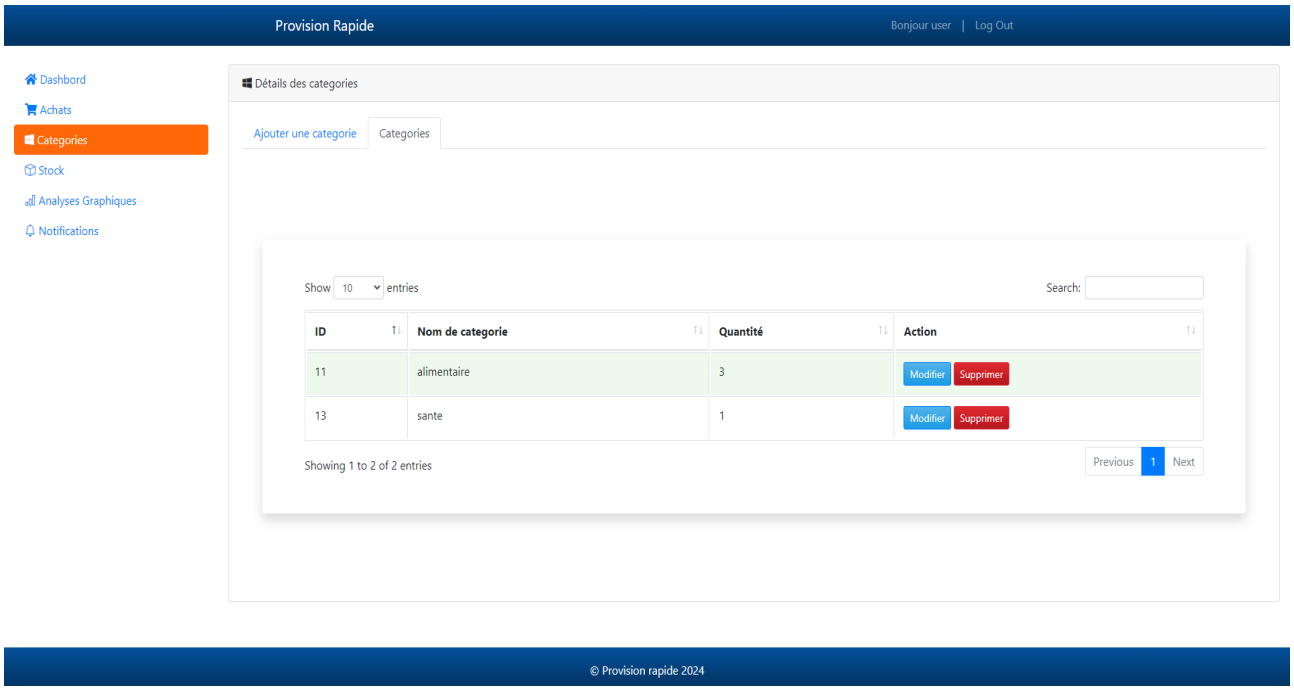

FIGURE 4.3.3 – Détails des catégories

Ajouter une categorie

L'utilisateur a le droit d'ajouter la catégorie dont il a besoin.

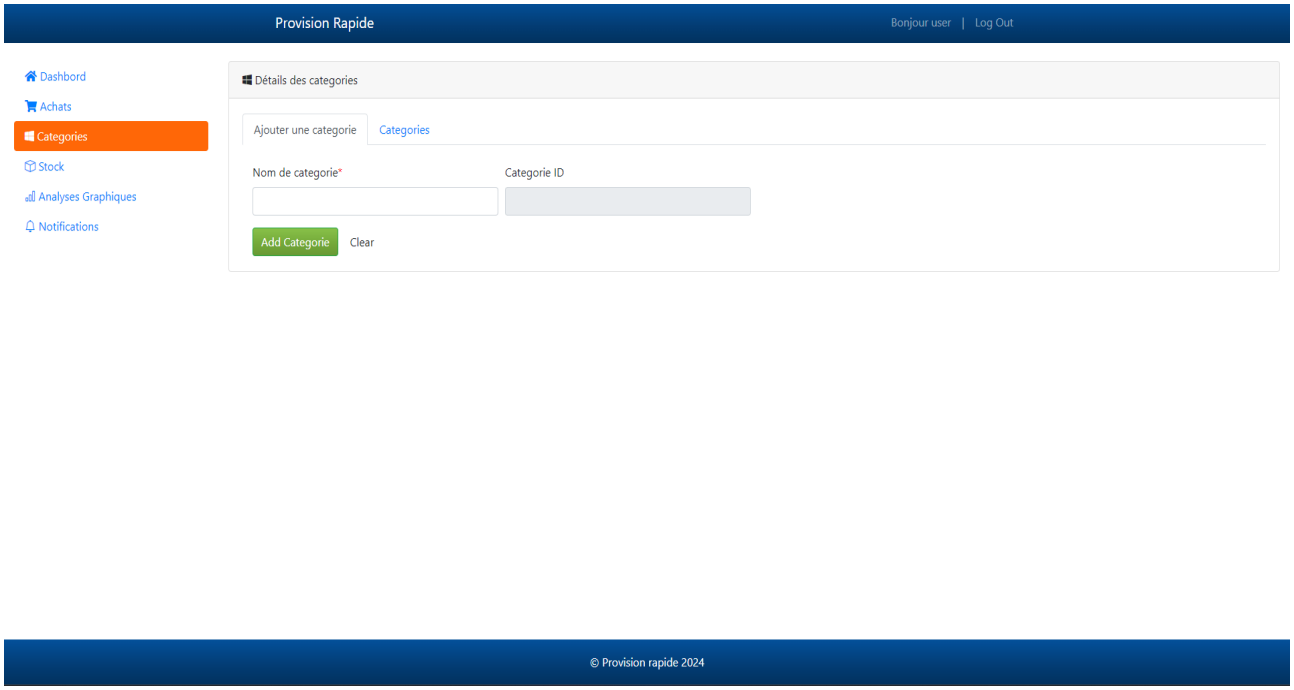

FIGURE 4.3.4 – Ajouter une catégorie

4.3.4 Gérer le stock

Maintenir une liste des articles disponibles dans le stock domestique de l'utilisateur

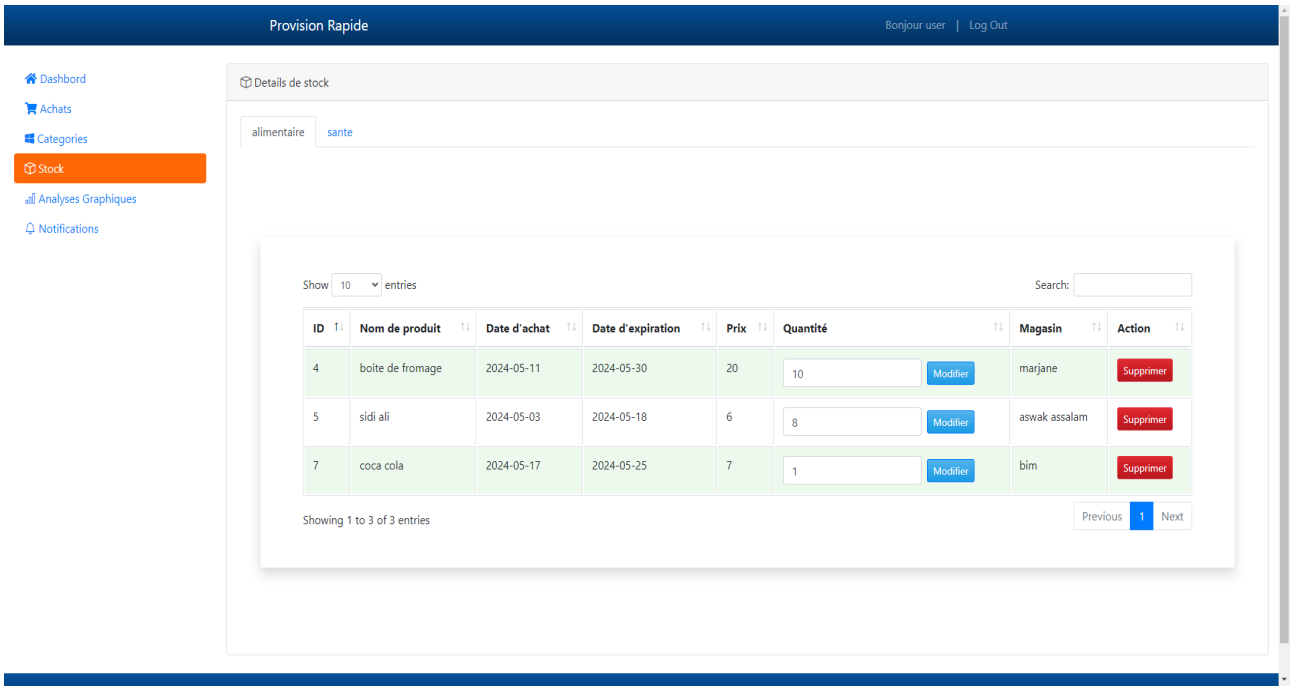

FIGURE 4.3.5 – La gestion du stock

Modifier les quantités

en mettant à jour les quantités disponibles après chaque achat , l'utilisateur doit modifier manuellement les quantités en cas de besoin.

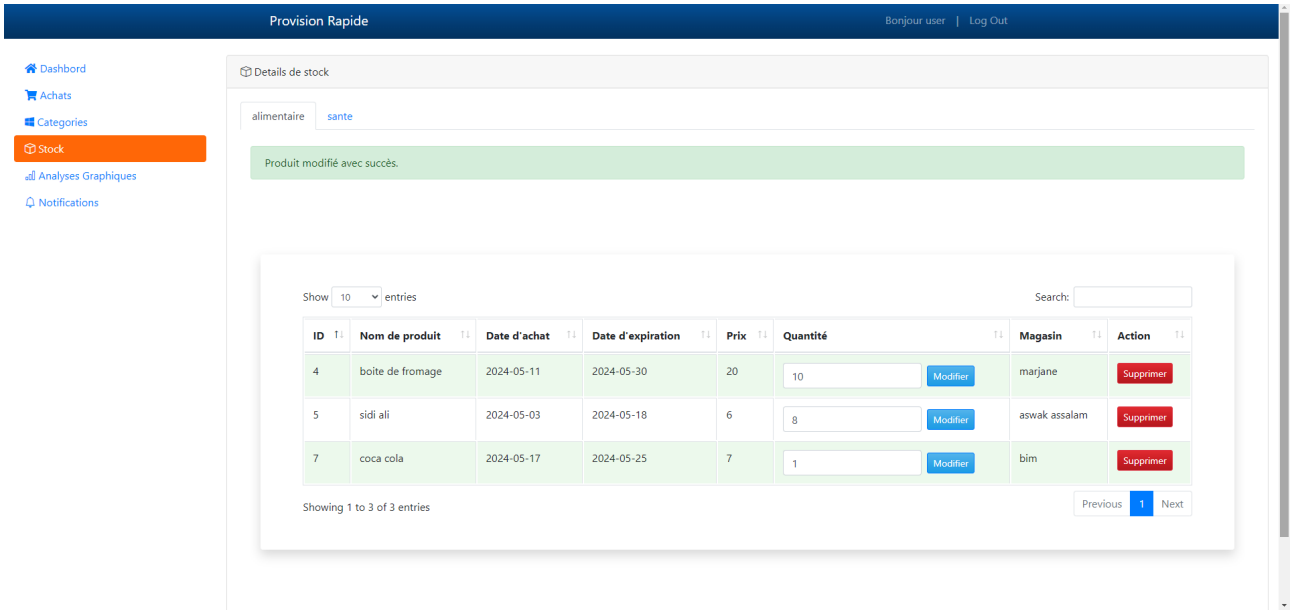

FIGURE 4.3.6 – Modification du quantité

Supprimer un produit

l'utilisateur peut supprimer un produit en cas de besoin

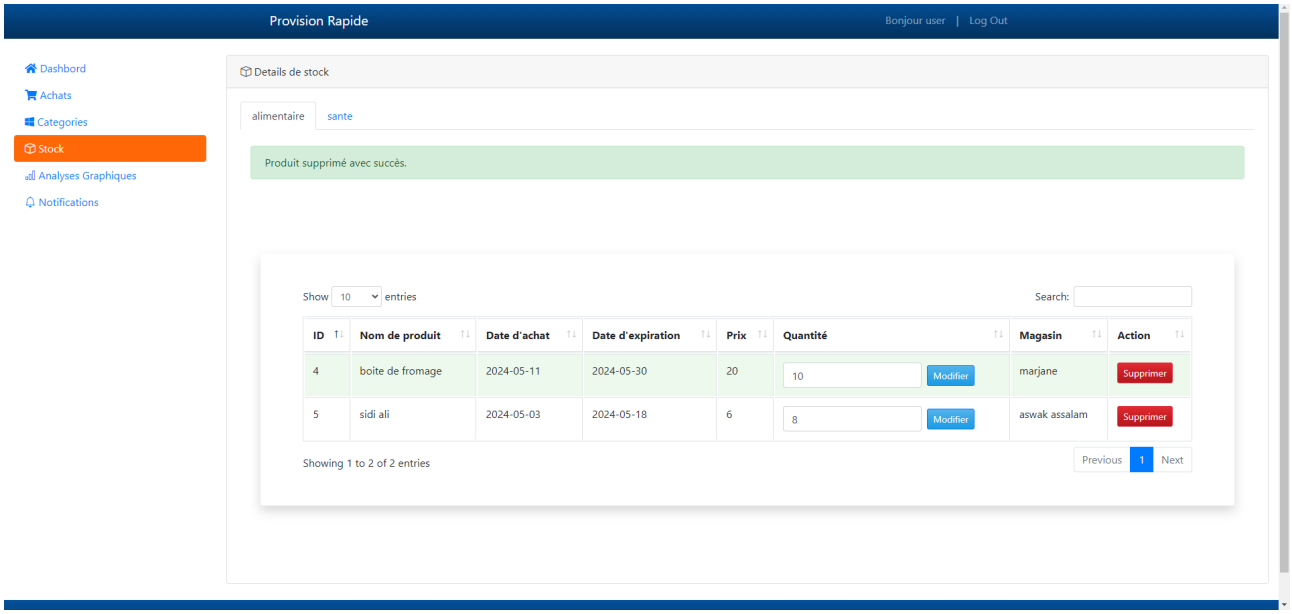

FIGURE 4.3.7 – Suppression d'un produit

4.3.5 Analyses graphiques

l'utilisateur peut visualiser des analyses graphiques concernant les articles les plus achetés, les marques les plus achetées et les quantités achetées par période (par exemple, par semaine, par mois).

FIGURE 4.3.8 – Analyses graphiques 1

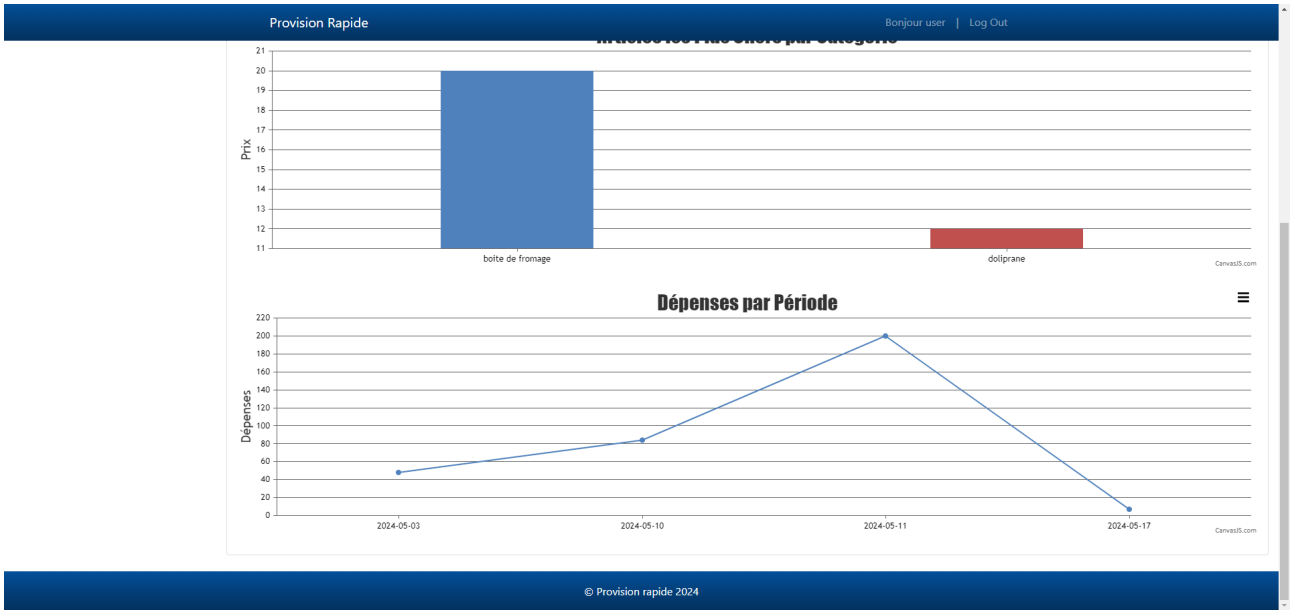

FIGURE 4.3.9 – Analyses graphiques 2

Exporter les graphes

L'utilisateur peut exporter les données enregistrées sous forme de fichiers compatibles avec d'autres applications ou formats (PNG, JPEG ,PDF) ou doit imprimer.

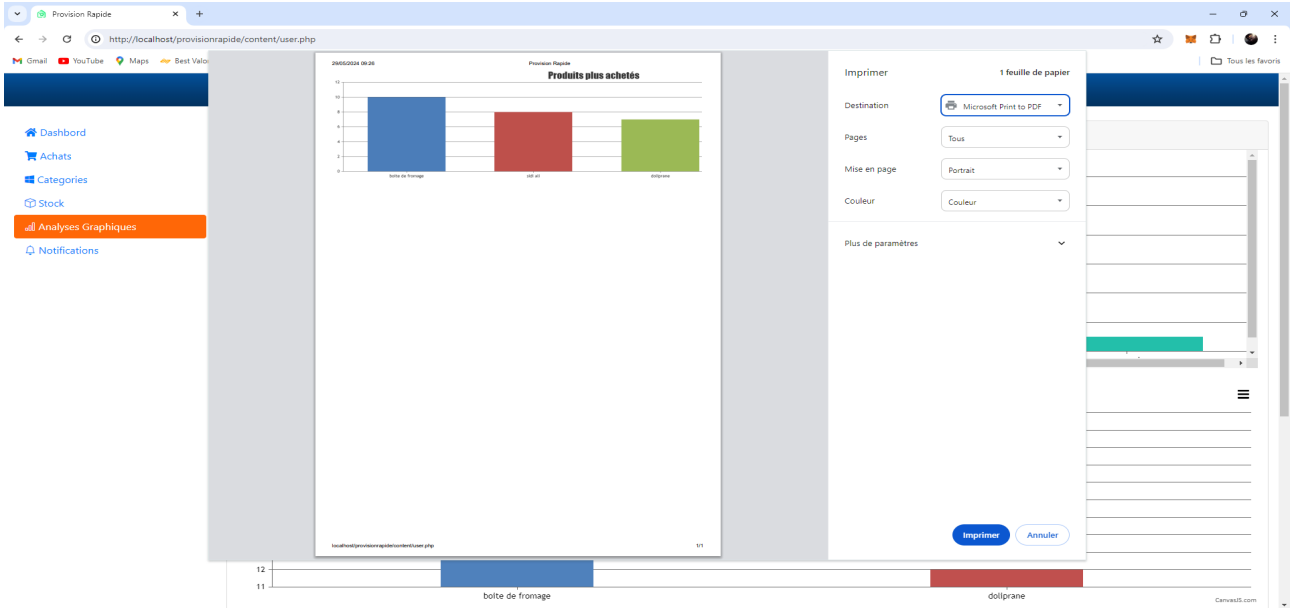

FIGURE 4.3.10 – Exporter les analyses graphiques

4.3.6 Notifications

Quand les dépenses dans une catégorie dépassent un seuil prédéfini ,et les niveaux de stock sont inférieurs a un seuil défini, et l'expiration de la date pris defini .Les couleurs des alertes change en fonction du type d'alertes

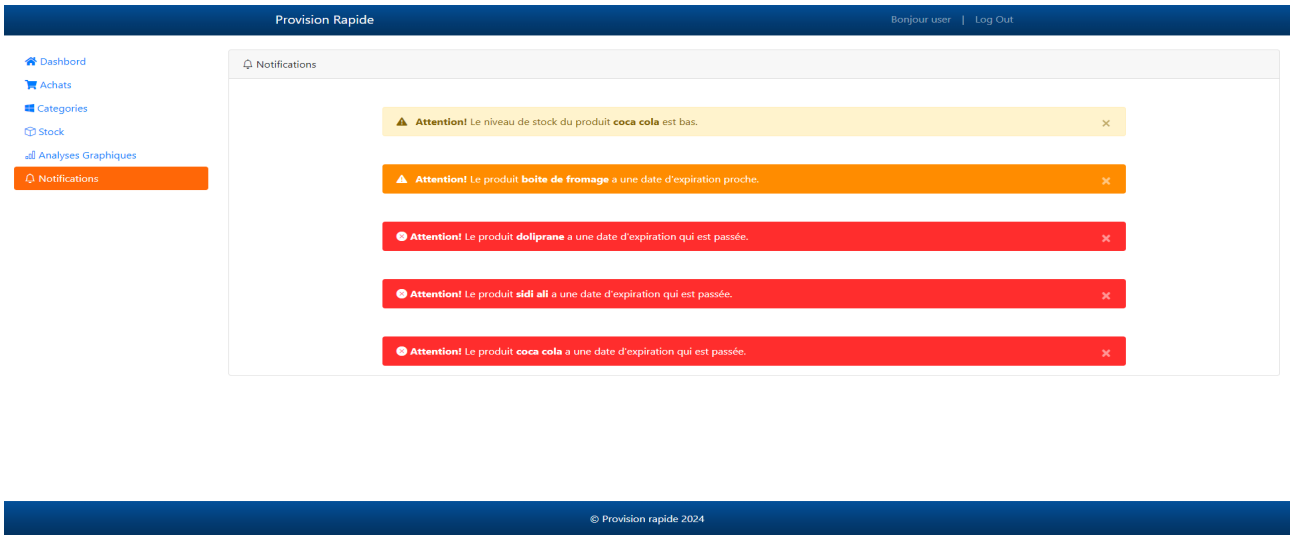

FIGURE 4.3.11 – Les alerts

4.4 Interface administrateur

4.4.1 Tableau de bord

C'est un tableau de bord où l'administrateur peut visualiser le nombre total des utilisateurs de l'application, qu'ils soient utilisateurs ordinaires ou administrateurs.

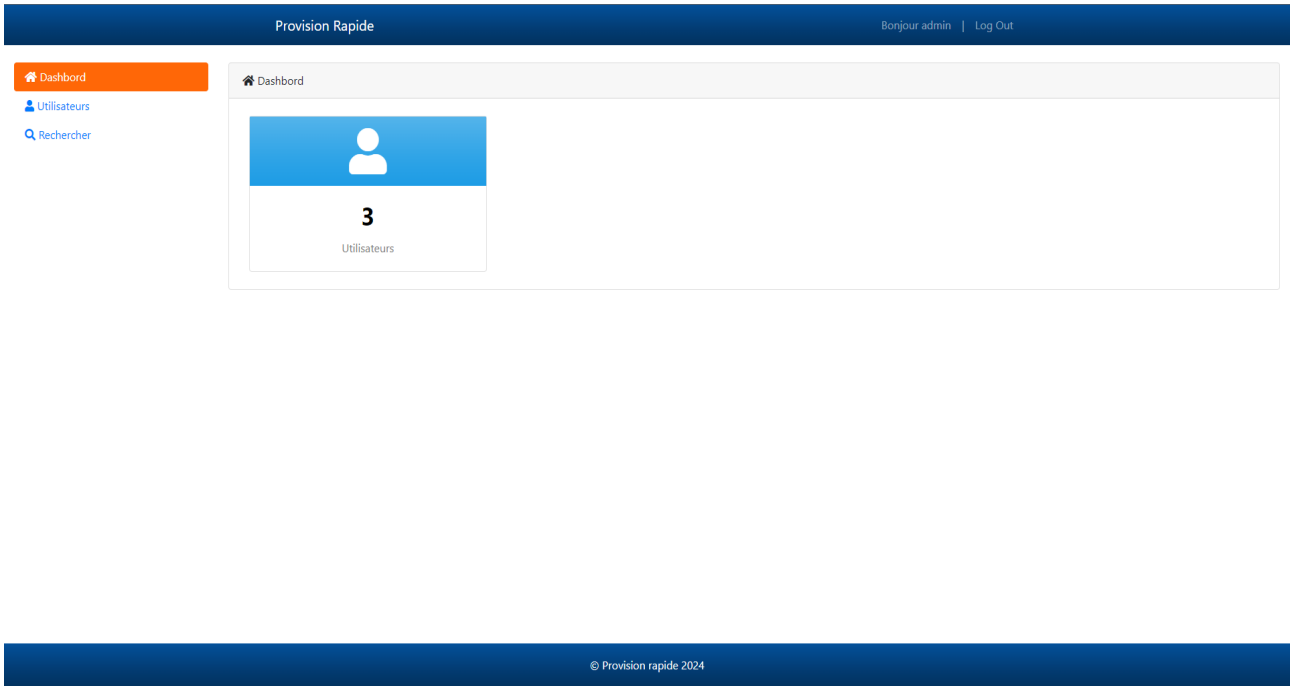

FIGURE 4.4.1 – Tableau de bord administrateur

4.4.2 Ajouter un utilisateur

C'est un formulaire comprenant une liste déroulante permettant de spécifier si l'utilisateur à ajouter est un utilisateur ordinaire ou un administrateur. Ce formulaire doit être rempli par l'administrateur afin d'ajouter un utilisateur à la base de données de l'application.

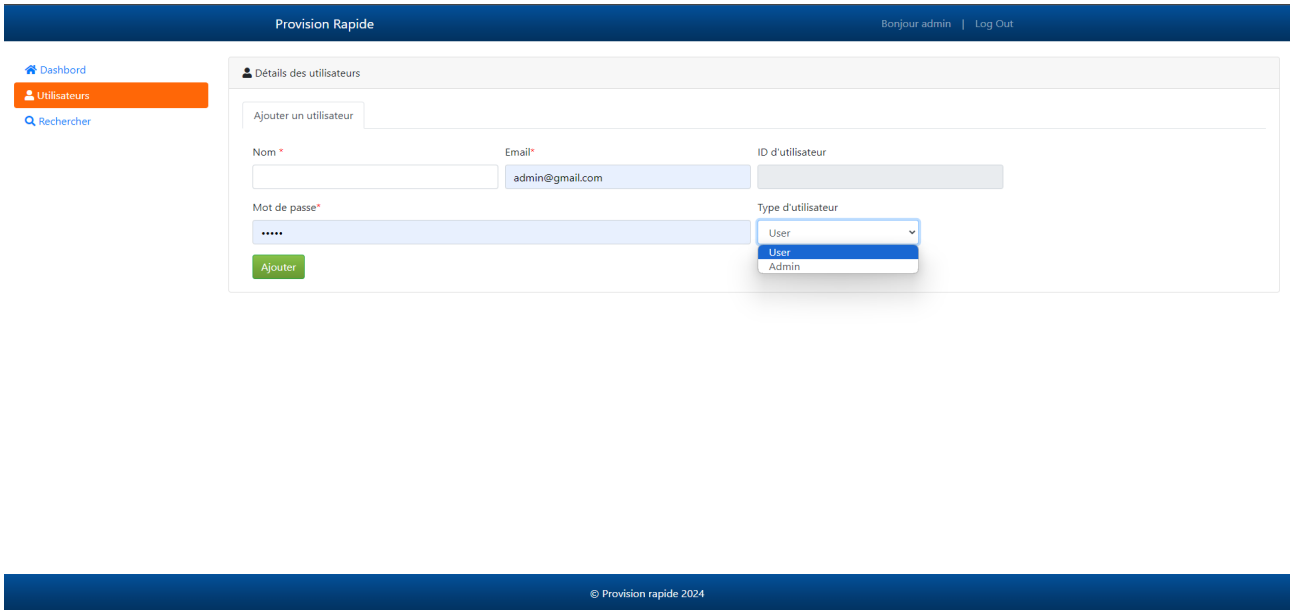

FIGURE 4.4.2 – Ajouter un utilisateur

4.4.3 Gérer les utilisateurs

L'administrateur peut visualiser les informations de tous les utilisateurs de l'application, effectuer des recherches sur un utilisateur spécifique, supprimer un utilisateur et également apporter des modifications aux informations des utilisateurs.

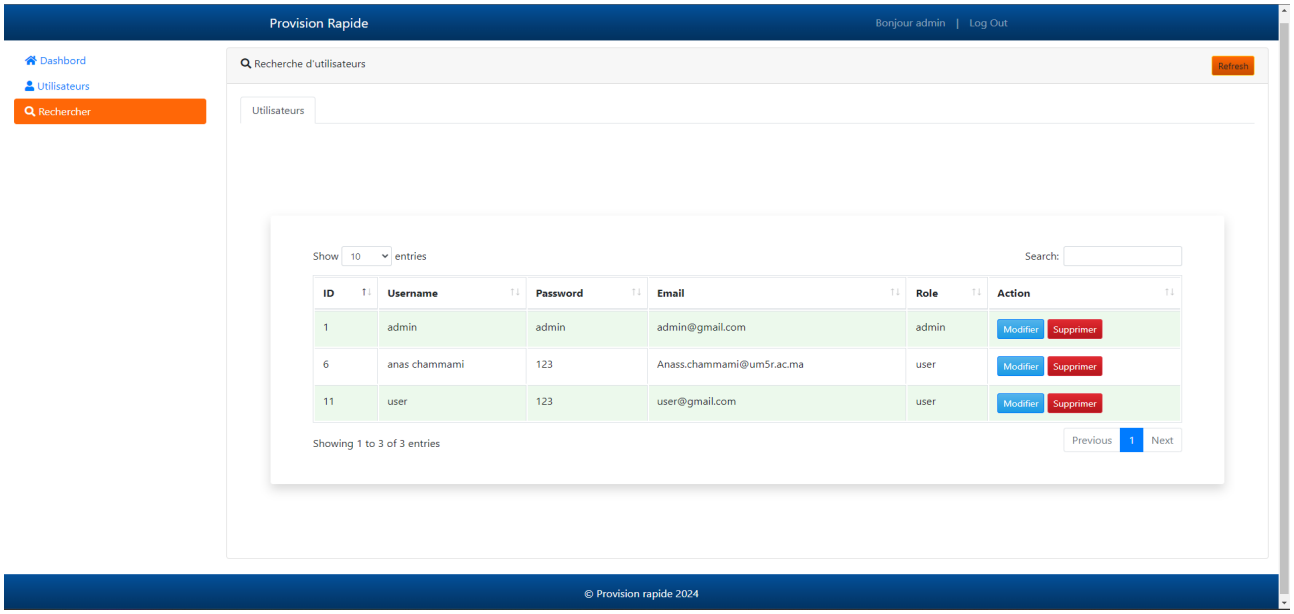

FIGURE 4.4.3 – Gestion d'utilisateur

Conclusions et Perspectives

Le projet de fin d'études est une étape cruciale dans la carrière d'un étudiant, car il offre un aperçu significatif de ses compétences et de son potentiel. Durant ce mémoire, l'étudiant met en pratique les connaissances et les compétences qu'il a accumulées au cours de ses trois années d'études. Il s'intègre ainsi dans le monde professionnel et développe également son sens de la collaboration.

Durant la période de conception et de réalisation de ce projet, nous avons pu appliquer et exploiter tout ce que nous avons appris au cours de notre cursus à la Faculté des Sciences. Nous avons considérablement amélioré notre sens collaboratif et avons également acquis de nouvelles notions qui n'étaient pas explicitement couvertes dans notre formation initiale.

Ce projet visait à développer une application web pour la gestion d'approvisionnement domestique nommée **Provision Rapide**, offrant aux utilisateurs une solution efficace pour gérer leur stock de manière organisée et simplifiée . À travers ce travail, nous avons atteint plusieurs objectifs importants : nous avons développé une interface conviviale permettant l'enregistrement et la gestion des produits ; aussi l'application intègre des fonctionnalités avancées telles que la visualisation graphique du stock et les notifications d'alerte ; une interface dédiée aux administrateurs pour la gestion des utilisateurs inscrits dans l'application.

Nous espérons que ce travail servira de base solide pour de futures initiatives visant à améliorer l'efficacité et la qualité de vie dans le domaine de la gestion domestique des ressources.

En conclusion, nos résultats ont démontré que l'application répond efficacement aux besoins énoncés dans le cahier des charges, permettant aux utilisateurs de mieux gérer leurs ressources et de réaliser des économies significatives. Cependant, nous avons identifié certaines limitations, telles que la nécessité d'améliorer la performance de certaines fonctionnalités, de renforcer la sécurité de l'application et d'ajouter des options de personnalisation pour une meilleure adaptation aux besoins spécifiques des utilisateurs.

Dans un avenir proche, ces limitations seront résolues car nous prévoyons d'ajouter des fonctionnalités qui enrichiront notre application, la rendant plus complète, répondant encore mieux aux besoins des utilisateurs. De plus, nous envisageons de développer des versions mobiles de l'application pour iOS et Android.

Bibliographie et Webographie

[1] Challenges, https://www.challenges.fr/societe/pres-de-20-de-la-nourriture-gaspillee-dans-le-monde_754039 (25/06/2024), dernière visite le 28 juin 2024.

[2] Lucidchart, https://www.lucidchart.com/pages/fr/langage-uml (16/04/2024), dernière visite le 28 juin 2024.

[3] Wikipedia (diagramme de cas d'utilisation), https://fr.wikipedia.org/wiki/Diagramme_de_cas_d'utilisation (27/03/2023), dernière visite le 28 juin 2024.

[4] Wikipedia (diagramme de classes), https://fr.wikipedia.org/wiki/Diagramme_de_classes (25/12/2023), dernière visite le 28 juin 2024.

[5] IBM, https://www.ibm.com/docs/fr/rsm/7.5.0?topic=uml-sequence-diagrams (22/01/2023), dernière visite le 28 juin 2024.

[6] Bility (Visual Studio Code), https://bility.fr/definition-visual-studio-code/ (28/11/2023), dernière visite le 28 juin 2024.

[7] WP Assistance, https://www.wp-assistance.fr/definition-wordpress/phpmyadmin/ (19/05/2023), dernière visite le 28 juin 2024.

[8] Boryl, https://www.boryl.fr/glossaire/mysql/ (01/12/2023), dernière visite le 28 juin 2024.

[9] Bility (Apache HTTP Server), https://bility.fr/definition-apache-http-web-server/ (28/11/2023), dernière visite le 28 juin 2024.

[10] Wikipedia (XAMPP), https://fr.wikipedia.org/wiki/XAMPP (03/03/2024), dernière visite le 28 juin 2024.

[11] Modeliosoft, https://www.modeliosoft.com/fr/fonctionnalites/modelio-studio.html (14/08/2020), dernière visite le 28 juin 2024.

[12] Linkweb, https://linkweb.fr/creation-site-internet-toulouse/html-css/ (21/09/2023), dernière visite le 28 juin 2024.

[13] Journal du Net (CSS), https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/

1203277-css-cascading-style-sheets-definition-traduction/ (16/04/2024), dernière visite le 28 juin 2024.

[14] Easy Micro, https://www.easy-micro.org/ (19/06/2024), dernière visite le 28 juin 2024.

[15] Journal du Net (SQL), https://www.journaldunet.fr/web-tech/guide-du-big-data/1203603-sql-structured-query-language-definition-traduction-et-acteurs/ (03/05/2024), dernière visite le 28 juin 2024.

[16] Journal du Net (JavaScript), https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203585-javascript/ (25/04/2024), dernière visite le 28 juin 2024.

[17] Wikipedia (Chart.js), https://fr.wikipedia.org/wiki/Chart.js (25/06/2024), dernière visite le 28 juin 2024.

[18] Journal du Net (jQuery), https://www.journaldunet.fr/web-tech/dictionnaire-du-webmastering/1203587-jquery-definition/ (25/04/2024), dernière visite le 28 juin 2024.

[19] Journal du Net (Bootstrap), https://www.journaldunet.com/developpeur/1159810-bootstrap-definition-tutoriels-astuces-pratiques/ (10/03/2024), dernière visite le 28 juin 2024.

[20] The Joy of PHP Programming : A Beginner's Guide to Programming Interactive Web Applications with PHP and MySQL by Alan Forbes, (2012).

[21] HTML5 For Web Designers by Jeremy Keith, (2010).