



Filière : Génie logiciel  
Module : web dynamique

## **RAPPORT DE PROJET WEB DYNAMIQUE**

**Réaliser par :**

AGOURRAM Sihame

ALLABOUCH Anass

**Encadrer par :**

IDRAIS

ANNEJAR

Année universitaire : 2024/2025

# Table des matières

Table de figure .....	3
Introduction :.....	4
CHAPITRE I :Contexte générale et cahier de charge .....	5
Introduction : .....	6
1.Problématique : .....	6
2.solution : .....	6
3.Outils utilisés:.....	7
3.1.language de programmation: .....	7
3.2.Logiciels : .....	8
CHAPITRE II : Conception.....	10
Introduction : .....	11
1.Diagramme de cas d'utilisation : .....	11
2.Explication de diagramme de cas d'utilisation : .....	12
3.Diagramme de classe : .....	13
CHAPITRE III : Réalisation .....	14
Introduction : .....	15
2.Architecture technique de l'application : .....	15
2.1.Architecture technique : .....	15
2.2. Patron de conception utilisé : .....	15
2.2.1.L'architecture MVC : .....	16
3.Interphases de l'application web: .....	17
3.1.Registre page: .....	17
3.2.Login page : .....	18
3.3.Dashboard : .....	19
3.4.Catégories pages: .....	20
3.5.Ajouter une catégorie : .....	20
3.5. Clients page : .....	21
3.6.Commandes page : .....	22
3.7.Bon de commande : .....	22

3.8.produits page : .....	23
3.9.Fournisseurs page : .....	24
3.10.utilisateur page:.....	25
Conclusion général : .....	26

## Table de figure

Figure 1:logo de code igniter4.....	7
Figure 2:logo de mysql.....	7
Figure 3:logo de tailwind css.....	8
Figure 4:logo de github.....	8
Figure 5:logo de git .....	8
Figure 6:logo de visual studio code.....	9
Figure 7:logo de xampp .....	9
Figure 8:Diagramme de cas d'utilisation .....	11
Figure 9:Diagramme de classe .....	13
Figure 10:la structure MVC.....	16
Figure 11:register page.....	17
Figure 12:verification email .....	17
Figure 13:login page .....	18
Figure 14:dashboard.....	19
Figure 15:catégorie page .....	20
Figure 16:ajouter une catégorie.....	21
Figure 17:client page.....	21
Figure 18:commandes page.....	22
Figure 19:bon de commande.....	23
Figure 20:produit page .....	23
Figure 21:fournisseur page .....	24
Figure 22:utilisateurs page .....	25

# Introduction :

Dans un monde où la digitalisation des processus est devenue essentielle, la gestion efficace des activités administratives d'une entreprise joue un rôle crucial dans sa performance et sa compétitivité. Ce constat est particulièrement vrai pour des secteurs tels que la bijouterie, où la précision, le suivi rigoureux des stocks et la relation client sont au cœur de l'activité. Dans cette optique, nous avons entrepris le développement d'un site web destiné à répondre aux besoins spécifiques de gestion d'une bijouterie, en nous appuyant sur le framework **CodeIgniter4**.

Le choix de CodeIgniter 4 s'explique par sa simplicité d'utilisation, sa flexibilité et sa capacité à fournir un cadre robuste pour le développement d'applications web. Ce projet a été conçu pour centraliser et faciliter la gestion des différentes composantes de l'activité de la bijouterie. L'objectif est de fournir une solution intuitive et performante qui simplifie les tâches répétitives et optimise les processus internes.

Réalisé en binôme, ce projet s'appuie sur une approche collaborative favorisant une répartition efficace des responsabilités et un partage continu des compétences. Ce travail a permis de combiner nos forces pour concevoir une application répondant aux exigences modernes de la gestion en ligne. Le site, axé sur la partie administrative, vise à devenir un outil indispensable pour le suivi et l'organisation des commandes, des stocks, des clients et des fournisseurs.

Ce rapport retrace les objectifs du projet, les étapes de sa réalisation et les enseignements tirés de cette expérience enrichissante.

# CHAPITRE I :Contexte générale et cahier de charge

## Introduction :

Dans ce chapitre, nous aborderons la section du cahier des charges dédiée à la contextualisation du projet, englobant l'analyse du contexte de travail, comprenant la problématique, la solution envisagée, les objectifs à atteindre, les besoins nécessaires à leur réalisation, ainsi que les outils et ressources prévus pour mener à bien ce projet.

## 1.Problématique :

Dans un secteur aussi exigeant que celui de la bijouterie, où la gestion précise des stocks, des commandes et des relations avec les clients et fournisseurs est essentielle, de nombreuses entreprises continuent de recourir à des méthodes manuelles ou à des outils peu adaptés. Cela peut entraîner des erreurs, une perte de temps et un manque de visibilité sur les activités. Comment concevoir et mettre en place un système numérique performant, simple d'utilisation et adapté aux besoins spécifiques d'une bijouterie pour optimiser sa gestion administrative tout en garantissant la sécurité et la fiabilité des données ?

## 2.solution :

La solution consiste à développer un site web dédié à la gestion administrative de la bijouterie en utilisant le framework **CodeIgniter 4**. Ce site permet de centraliser et d'automatiser les principales tâches administratives, telles que la gestion des commandes, des stocks, des clients, des utilisateurs et des fournisseurs.

Le choix de CodeIgniter 4 garantit une structure robuste, sécurisée et évolutive, tout en offrant une interface intuitive pour les administrateurs. Le système d'authentification intégré assure un accès sécurisé aux fonctionnalités, tandis que les outils de gestion simplifient et optimisent les processus internes. Cette solution numérique vise à réduire les erreurs, à améliorer l'efficacité opérationnelle et à offrir une meilleure visibilité sur l'ensemble des activités de la bijouterie .

## 3.Outils utilisés:

### 3.1.language de programmation:

#### 3.1.1. Code igniter4:



Figure 1:logo de code igniter4

CodeIgniter 4 est un framework PHP léger et rapide, conçu pour le développement d'applications web modernes. Basé sur une architecture MVC (Modèle-Vue-Contrôleur), il permet de structurer le code de manière claire et organisée. Avec ses fonctionnalités intégrées, comme la gestion des routes, la sécurité renforcée et les outils de migration de base de données, CodeIgniter 4 offre une solution efficace et performante pour créer des applications sécurisées et faciles à maintenir.

#### 3.1.2.MySQL :



Figure 2:logo de mysql

MySQL est un système de gestion de base de données relationnelle (SGBDR) open-source, réputé pour sa performance, sa fiabilité et sa simplicité d'utilisation. Il permet de stocker, organiser et manipuler des données en utilisant le langage SQL (Structured Query Language).

### 3.1.3.Tailwind css :



Figure 3:logo de tailwind css

Tailwind CSS est une bibliothèque CSS utilitaire qui permet de créer des interfaces utilisateur rapidement en combinant des classes prédéfinies pour styliser les éléments HTML.

## 3.2.Logiciels :

### 3.2.1.Github :

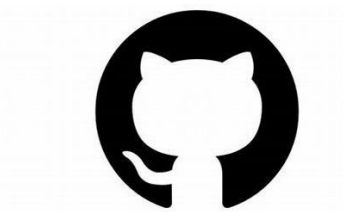


Figure 4:logo de github

GitHub est une plateforme de développement collaboratif de logiciels qui permet aux développeurs de stocker, gérer et partager leur code source, ainsi que de collaborer avec d'autres développeurs à travers le contrôle de version Git.

### 3.2.2.Git :



Figure 5:logo de git

Git est un système de contrôle de version distribué, utilisé principalement pour gérer les modifications apportées aux projets logiciels. Il permet de suivre l'évolution du code source, de gérer les branches de développement et de faciliter la collaboration entre les membres d'une équipe. Avec Git, les développeurs peuvent travailler de manière efficace et sécurisée, en garantissant l'intégrité et la traçabilité des modifications.



### 3.2.3. Visual studio code :



Figure 6: logo de visual studio code

VisualStudio Code est un éditeur de code source léger et polyvalent, développé par Microsoft, offrant une gamme étendue de fonctionnalités intégrées et d'extensions pour répondre aux besoins des développeurs dans divers langages de programmation.

### 3.2.4. Xampp:



Figure 7: logo de xampp

XAMPP est un package logiciel open-source qui facilite l'installation et l'utilisation d'un serveur web local. Il inclut les composants essentiels tels que **Apache** (serveur web), **MySQL/MariaDB** (gestion des bases de données) et **PHP** (langage de programmation). XAMPP est simple à configurer et est largement utilisé par les développeurs pour tester et développer des applications web dans un environnement local avant leur déploiement en ligne.

## CHAPITRE II : Conception

## Introduction :

La phase de conception est une étape clé dans le développement de notre site web. Elle consiste à définir l'architecture du système, les différentes fonctionnalités, ainsi que l'organisation des interfaces. Cette étape permet de traduire les besoins identifiés en une structure claire et fonctionnelle, garantissant ainsi une base solide pour la mise en œuvre technique du projet. Dans ce cadre, nous avons utilisé des outils et des méthodologies adaptés pour concevoir un site performant, intuitif et répondant aux exigences du domaine de la bijouterie.

## 1.Diagramme de cas d'utilisation :

Les diagrammes de cas d'utilisation sont essentiels dans la conception des sites web car ils permettent de comprendre les besoins et les objectifs des utilisateurs finaux. Ils identifient les actions possibles des utilisateurs sur le site et montrent comment ces actions interagissent avec le système. Ces diagrammes définissent les étapes nécessaires pour accomplir une tâche, offrant une vue d'ensemble du flux d'interaction. Ils servent de base pour concevoir des sites web conviviaux, centrés sur les besoins des utilisateurs, favorisant ainsi une expérience utilisateur positive.

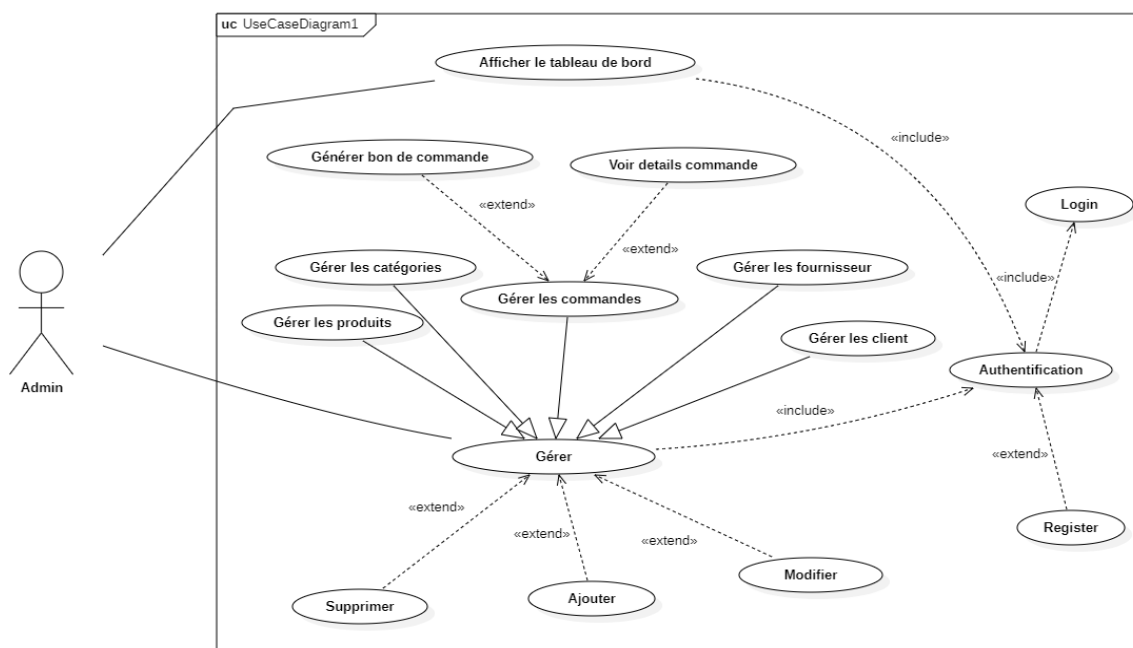


Figure 8:Diagramme de cas d'utilisation

## 2.Explication de diagramme de cas d'utilisation :

Le diagramme de cas d'utilisation présenté ci-dessus décrit les fonctionnalités principales du système de gestion de la bijouterie du point de vue de l'administrateur. Il met en lumière les cas d'utilisation suivants :

1. **Authentification** : L'accès au système est sécurisé par un processus d'authentification, comprenant les fonctionnalités de connexion (*Login*) et d'inscription (*Register*).
2. **Gestion des entités principales** :
  - **Gérer les commandes** : Inclut des actions spécifiques comme la génération d'un bon de commande et la consultation des détails d'une commande.
  - **Gérer les produits, les catégories, les clients et les fournisseurs** : Ces fonctionnalités permettent d'ajouter, modifier ou supprimer les différentes entités nécessaires au bon fonctionnement de la bijouterie.
3. **Actions génériques** : Le cas d'utilisation central *Gérer* regroupe des actions communes telles que *Ajouter*, *Modifier* et *Supprimer*, qui sont étendues aux différentes entités gérées.
4. **Tableau de bord** : L'administrateur peut afficher un tableau de bord offrant une vue d'ensemble des données et statistiques pertinentes.

Les relations « **include** » indiquent des dépendances entre certains cas d'utilisation, comme l'authentification nécessaire pour accéder aux autres fonctionnalités. Les relations « **extend** » illustrent des fonctionnalités optionnelles ou supplémentaires, comme la génération de bons de commande dans la gestion des commandes.

Ce diagramme met en évidence une architecture claire et modulaire du système, assurant une meilleure compréhension et une maintenance simplifiée.

### 3.Diagramme de classe :

Le diagramme de classe est un type de diagramme de modélisation UML utilisé pour représenter la structure statique d'un système logiciel. Il décrit les classes du système, leurs attributs, leurs méthodes, ainsi que les relations entre ces classes, telles que l'héritage, l'agrégation, l'association et la composition. En résumé, il offre une vue statique des entités du système et de leurs interactions, aidant ainsi à la compréhension de la structure globale du logiciel.

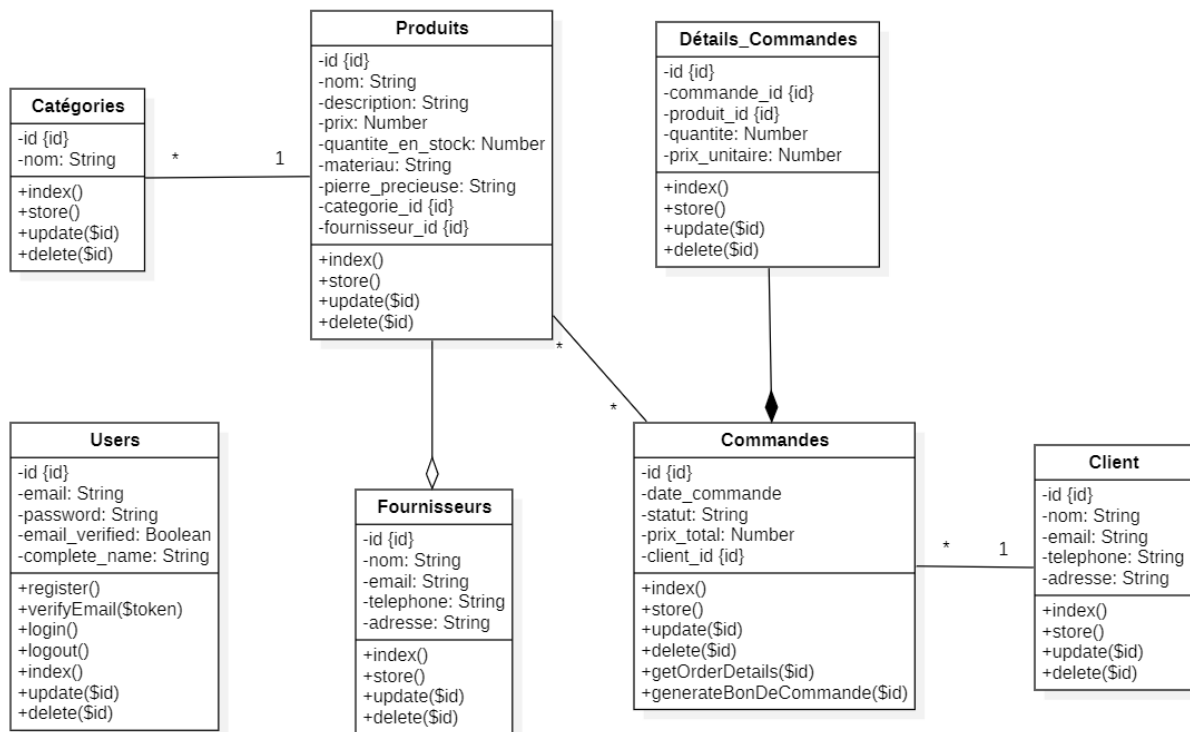


Figure 9:Diagramme de classe

## CHAPITRE III : Réalisation

## Introduction :

La phase de réalisation transforme les concepts de conception en fonctionnalités concrètes, développant à la fois les composants côté serveur et client. En respectant les bonnes pratiques de programmation et en implémentant des tests rigoureux, nous assurons la modularité, la réutilisabilité et la fiabilité du code. Cette étape cruciale aboutit à un produit fonctionnel prêt à être testé et déployé.

## 2. Architecture technique de l'application :

### 2.1. Architecture technique :

L'architecture technique est essentielle pour assurer une bonne compréhension, gestion et optimisation de tout système. Pour structurer notre système, j'ai choisi un découpage en couches, justifiant ainsi le choix de l'architecture MVC. Une application mobile doit accomplir des opérations complexes liées au métier de son concepteur tout en offrant à l'utilisateur des interfaces graphiques riches et conviviales. Il est crucial de séparer autant que possible la logique métier de l'interface graphique afin de permettre un développement parallèle et harmonieux du projet. Cette séparation facilite également les mises à jour inévitables de l'application. Le modèle MVC répond parfaitement à cette exigence de séparation, permettant une organisation claire et efficace des différentes parties du développement.

### 2.2. Patron de conception utilisé :

Une application web doit accomplir des opérations complexes liées au domaine d'activité de son concepteur tout en offrant à l'utilisateur des interfaces graphiques riches et conviviales. Il est crucial de séparer autant que possible la logique métier de l'interface graphique afin de permettre un développement parallèle et efficace du projet. Cette séparation facilite également les mises à jour inévitables de l'application.

Le modèle MVC répond parfaitement à cette exigence de séparation, en organisant les fonctions nécessaires en trois catégories distinctes : le Modèle pour la gestion des données, la Vue pour la présentation de l'interface utilisateur, et le Contrôleur pour la gestion des interactions entre les deux.

### 2.2.1.L'architecture MVC :

L'architecture MVC est un concept puissant utilisé dans le développement d'applications interactives. Son principal avantage réside dans la séparation des données (modèle), de l'affichage (vue) et des actions (contrôleur). Cette division assure une architecture claire et facilite la tâche des développeurs lors de la maintenance et de l'amélioration du projet. Elle permet une gestion efficace des différentes problématiques liées aux composants, en les isolant dans leurs propres architectures respectives. Les différentes interactions entre le modèle, la vue et le contrôleur sont illustrées par le schéma de la figure ci-dessous :

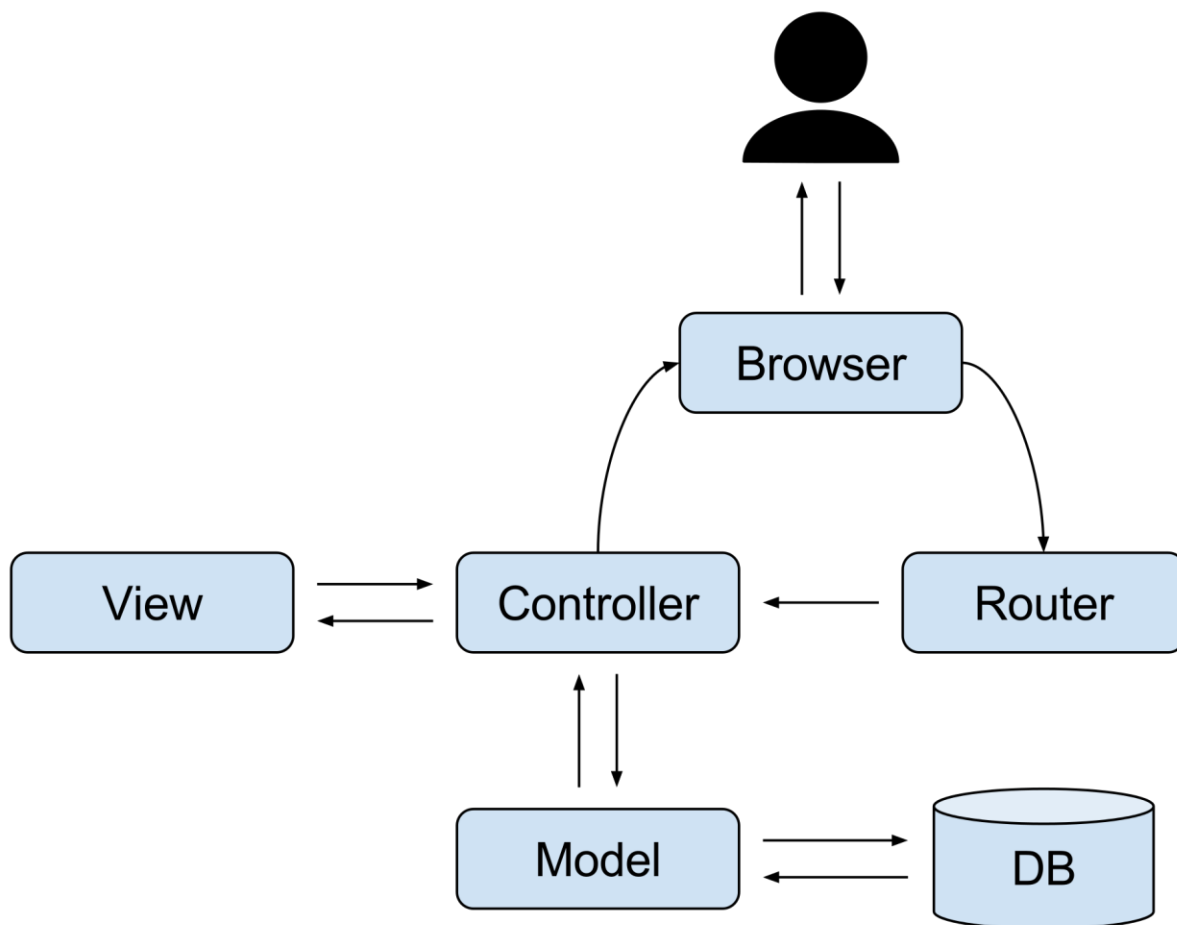


Figure 10:la structure MVC



### 3. Interphases de l'application web:

#### 3.1. Registre page:

Sur la page d'inscription, l'administrateur doit renseigner son nom complet, une adresse e-mail valide, un mot de passe, et confirmer ce dernier.

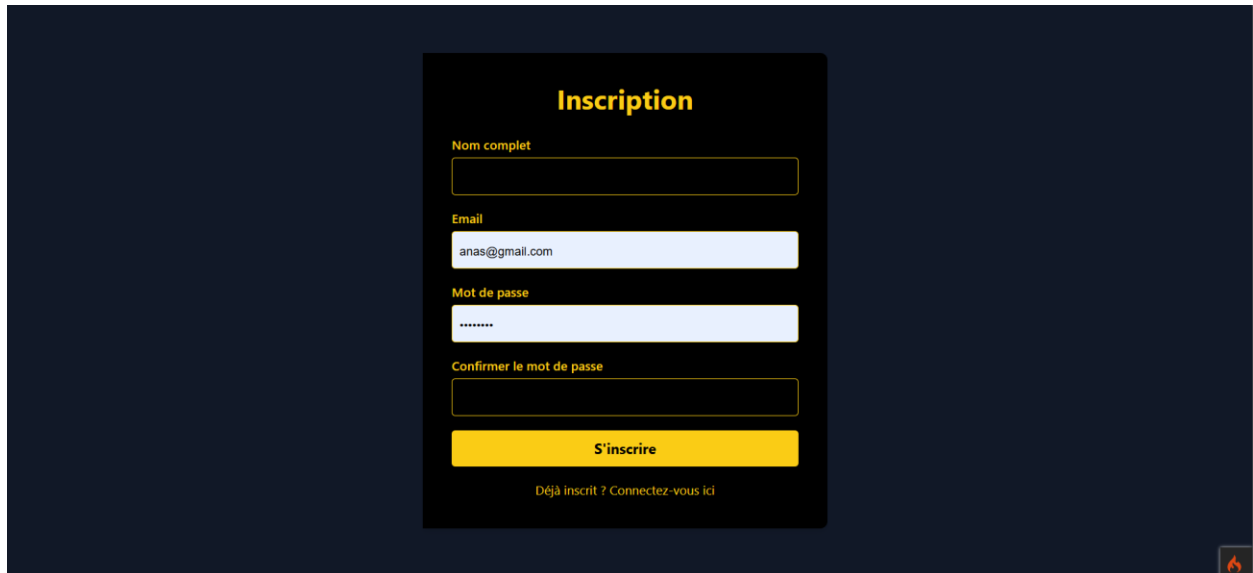
The image shows a registration form titled "Inscription" on a dark background. The form contains four input fields: "Nom complet", "Email" (with the example "anas@gmail.com"), "Mot de passe" (with masked characters "\*\*\*\*\*"), and "Confirmer le mot de passe". Below these fields is a yellow button labeled "S'inscrire". At the bottom of the form, there is a link that says "Déjà inscrit ? Connectez-vous ici".

Figure 11:register page

Après avoir cliqué sur le bouton "S'inscrire", l'administrateur reçoit un e-mail à l'adresse qu'il a renseignée, contenant un lien de confirmation.



Figure 12:verification email

### 3.2.Login page :

Après avoir cliqué sur le lien de vérification, l'administrateur est redirigé vers la page de connexion. Il doit y saisir son adresse e-mail et son mot de passe. Si les informations sont correctes, il pourra accéder au site.

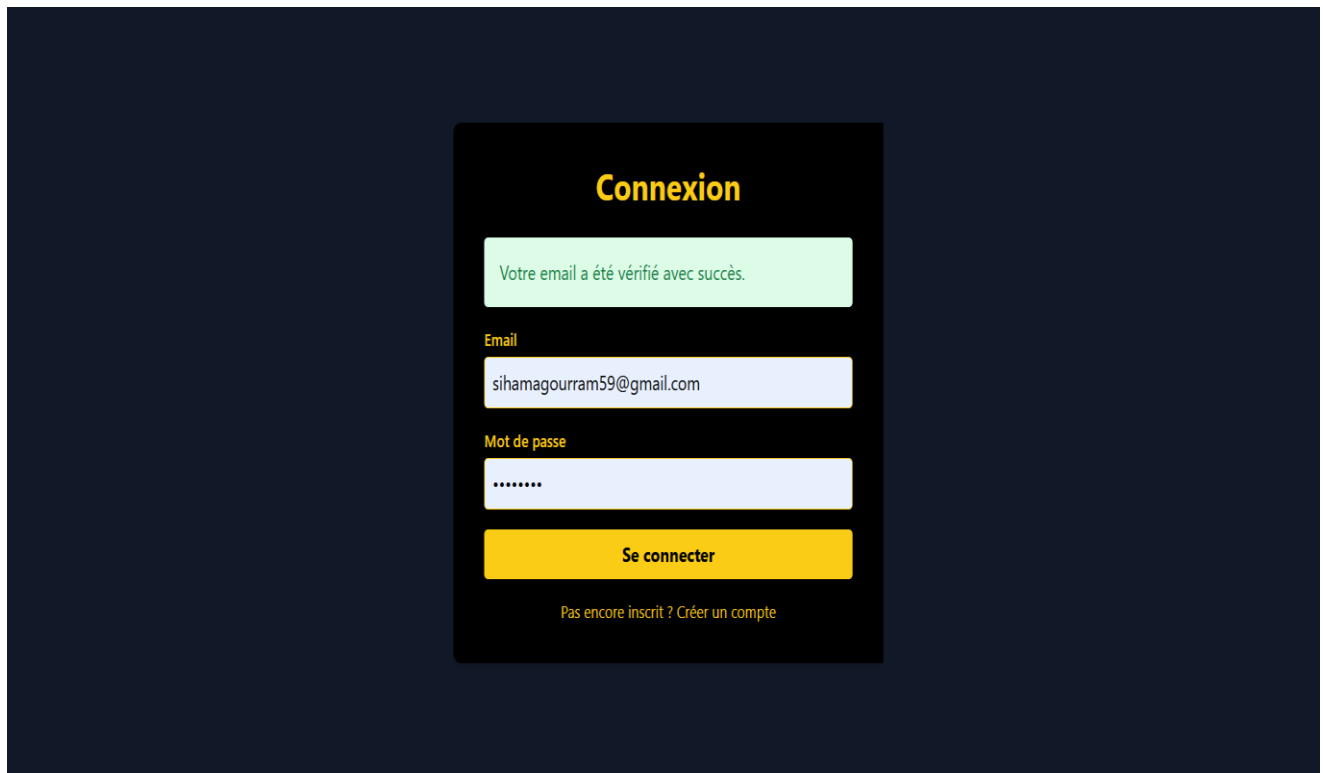
The image shows a login form titled "Connexion" in orange text. At the top, a green message box states "Votre email a été vérifié avec succès." Below this, there are two input fields: "Email" with the value "sihamagourram59@gmail.com" and "Mot de passe" with masked characters ".....". A yellow "Se connecter" button is positioned below the password field. At the bottom, a link reads "Pas encore inscrit ? Créer un compte". The entire form is centered on a dark blue background.

Figure 13:login page

### 3.3.Dashboard :

Cette page fournit un aperçu général et des métriques clés pour gérer efficacement l'inventaire de bijoux. Elle inclut également des rapports de vente détaillés et une répartition de l'inventaire. De plus, une barre latérale (dashside) permet un accès rapide aux sections principales : catégories, clients, commandes, produits et fournisseurs.



Figure 14:dashboard

### 3.4.Catégories pages:

Cette page présente l'ensemble des catégories disponibles dans la bijouterie, offrant la possibilité de les modifier ou de les supprimer selon les besoins aussi que la bouton ajouter catégorie

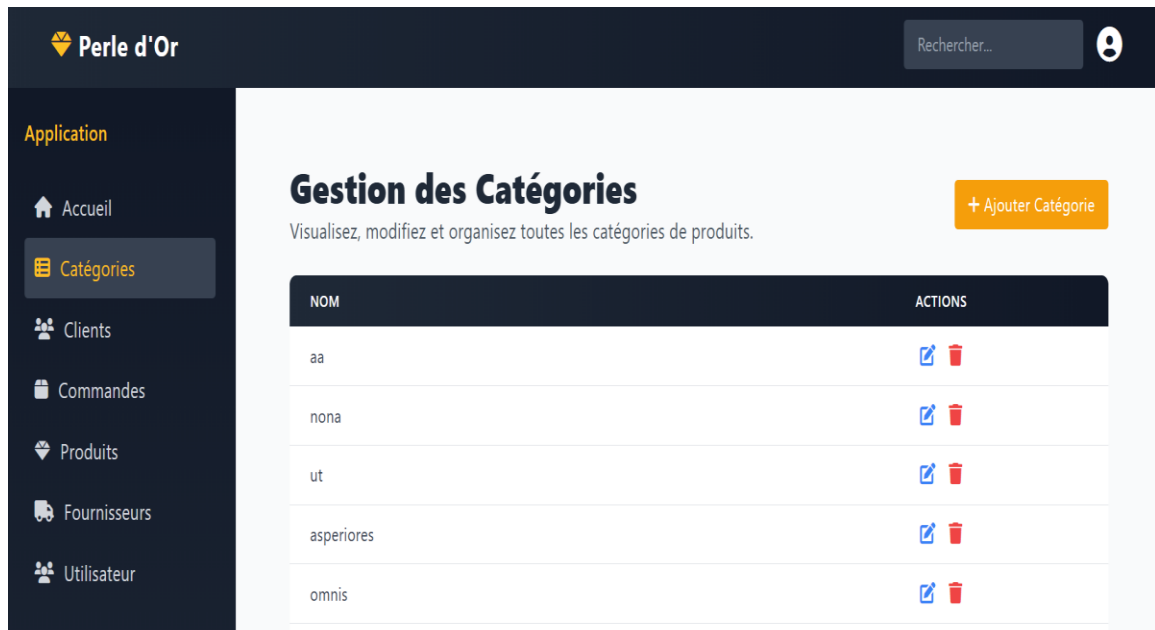


Figure 15:catégorie page

### 3.5.Ajouter une catégorie :

Lorsque l'utilisateur clique sur le bouton "Ajouter une catégorie", un formulaire s'affiche permettant de saisir le nom de la nouvelle catégorie afin de l'ajouter à la liste existante.



Figure 16:ajouter une catégorie

### 3.5. Clients page :

Cette page est dédiée à la gestion des clients, où sont affichés leurs noms, e-mails, numéros de téléphone et adresses. Elle offre également des options pour ajouter, mettre à jour ou supprimer des clients. L'accès à cette page est exclusivement réservé à l'administrateur.

The screenshot displays the 'Gestion des Clients' (Client Management) page. On the left is a dark sidebar with the 'Perle d'Or' logo and a search bar. The sidebar menu includes 'Accueil', 'Catégories', 'Clients' (highlighted), 'Commandes', 'Produits', 'Fournisseurs', and 'Utilisateur'. The main content area has the title 'Gestion des Clients' and a subtitle 'Visualisez, modifiez et organisez toutes les informations de vos clients.' An orange '+ Ajouter Client' button is in the top right. Below is a table with four columns: NOM, EMAIL, TÉLÉPHONE, ADRESSE, and ACTIONS. The table lists four clients: Anahi Robel, Ayden Anderson, Bessie Okuneva, and Philip Ritchie. Each row has edit and delete icons in the ACTIONS column.

NOM	EMAIL	TÉLÉPHONE	ADRESSE	ACTIONS
Anahi Robel	eschneider@thompson.info	(346) 489-0823	8263 Jacobs Mountain New Marcosside, PA 72388-3179	
Ayden Anderson	ieffertz@yahoo.com	+1-248-945-6766	6109 Keebler Brook Port Eleanore, IL 33086-9085	
Bessie Okuneva	waelchi.hollie@yahoo.com	+1.442.348.5261	151 Shanel Mills Apt. 430 New Alberthaport, VT 18671-1932	
Philip Ritchie	rita.larkin@gutkowski.com	+15024627065	915 Randi Trail Spinkaport, RI 89324	

Figure 17:client page

### 3.6.Commandes page :

Cette page est dédiée à la gestion des commandes. Elle affiche les informations essentielles telles que la date, le statut, le client associé, et offre des actions permettant de mettre à jour ou de supprimer une commande. Il est également possible d'ajouter une nouvelle commande ou d'imprimer un PDF récapitulatif.



**Figure 18:commandes page**

### 3.7. Bon de commande :

Ce bon de commande inclut les informations suivantes : le nom du client, son numéro de téléphone, son adresse e-mail, ainsi que le nombre de produits commandés et leur prix.

## Bon de Commande #25

Date: 1983-09-30

**Client:** Alicia Bradtke  
**Email:** huels.briana@adams.com  
**Téléphone:** +1 (224) 468-8978

Produit	Quantité	Prix Unitaire	Total
placeat	3	84.86 €	254.58 €
dignissimos	10	88.71 €	887.10 €
repellendus	8	68.15 €	545.20 €

**Total à payer: 1 686,88 €**

Merci pour votre commande !

Figure 19:bon de commande

### 3.8.produits page :

La gestion des produits permet de gérer les informations suivantes : le nom, la description, le prix, la quantité en stock, le matériau, la pierre précieuse, la catégorie et le fournisseur. Des actions sont également disponibles pour ajouter, mettre à jour (update) ou supprimer (delete) un produit.

## Gestion des Produits

Visualisez, modifiez et gérez tous les produits disponibles.

[+ Ajouter Produit](#)

ID	NOM	DESCRIPTION	PRIX	QUANTITÉ EN STOCK	MATÉRIAU	PIERRE PRÉCIEUSE	CATÉGORIE	FOURNISSEUR	ACTIONS
1	ut	Nobis...	12.5	43	gold	ruby	id	Oberbrunner Ltd	<a href="#">✎</a> <a href="#">🗑</a>
2	placeat	Quia...	12.06	6	gold	ruby	necessitatibus	Oberbrunner Ltd	<a href="#">✎</a> <a href="#">🗑</a>
3	blanditiis	In...	17.39	29	silver	sapphire	ut	Yundt, Huel and Heller	<a href="#">✎</a> <a href="#">🗑</a>
4	et	Fugiat...	73.08	6	silver		asperiores	Strosin Group	<a href="#">✎</a> <a href="#">🗑</a>
5	aut	Vitae...	56.22	31	platinum	sapphire	ut	Purdy-Rice	<a href="#">✎</a> <a href="#">🗑</a>
6	molestiae	Voluptatum...	43.5	22	platinum	ruby	numquam	Oberbrunner Ltd	<a href="#">✎</a> <a href="#">🗑</a>

Figure 20:produit page

### 3.9.Fournisseurs page :

Dans la page des fournisseurs, sont affichés le nom, l'email et le numéro de téléphone de chaque fournisseur. Il est également possible d'effectuer les actions suivantes : ajouter, mettre à jour (update) ou supprimer (delete) un fournisseur.

Gestion des Fournisseurs				
Visualisez, modifiez et gérez tous les fournisseurs disponibles.				
<a href="#">+ Ajouter Fournisseur</a>				
ID	NOM	EMAIL	TÉLÉPHONE	ACTIONS
1	Kilback PLC	chuels@barrows.com	984-421-6343	<a href="#">✎</a> <a href="#">🗑</a>
2	Hoeger Inc	vheathcote@prohaska.com	(910) 391-8908	<a href="#">✎</a> <a href="#">🗑</a>
3	Funk-Ratke	eula81@walter.info	1-734-965-8033	<a href="#">✎</a> <a href="#">🗑</a>
4	Strosin Group	kaden94@gottlieb.com	210-397-6249	<a href="#">✎</a> <a href="#">🗑</a>
5	Marquardt, Glover and Breitenberg	verdie80@smith.com	+1.651.266.2720	<a href="#">✎</a> <a href="#">🗑</a>
6	Yundt, Huel and Heller	demetrius36@labadie.net	580.255.3826	<a href="#">✎</a> <a href="#">🗑</a>
7	Padberg Inc	emard.alberta@ullrich.biz	+1-858-629-8075	<a href="#">✎</a> <a href="#">🗑</a>
8	Purdy-Rice	heathcote.norbert@gislason.com	458-655-8529	<a href="#">✎</a> <a href="#">🗑</a>
9	Oberbrunner Ltd	lexus.morar@schulist.com	+1-689-328-3865	<a href="#">✎</a> <a href="#">🗑</a>
10	Johnson, Roberts and Rosenbaum	lula41@schumm.com	1-314-318-2229	<a href="#">✎</a> <a href="#">🗑</a>

Figure 21:fournisseur page



### 3.10.utilisateur page:

La page des utilisateurs contient la liste des administrateurs authentifiés. Elle permet également d'effectuer les actions de mise à jour (update) et de suppression (delete) des utilisateurs.



The screenshot displays the 'Gestion des utilisateur' page within the 'Perle d'Or' application. The sidebar on the left contains the following navigation items: Accueil, Catégories, Clients, Commandes, Produits, Fournisseurs, and Utilisateur (highlighted). The main content area features a search bar labeled 'Rechercher...' and a user profile icon. Below this, the title 'Gestion des utilisateur' is followed by the subtitle 'Visualisez, modifiez et organisez toutes les informations de vos utilisaeur .'. A table lists the users with columns for NOM, EMAIL, and ACTIONS.

NOM	EMAIL	ACTIONS
Anas allabouch	anas@gmail.com	 
Sihame Agourram	sihame@gmail.com	 

Figure 22:utilisateurs page

## Conclusion général :

En conclusion, le développement de ce système de gestion pour la bijouterie a permis de créer une plateforme robuste et fonctionnelle, facilitant la gestion des stocks, des commandes, des clients, des fournisseurs et des utilisateurs. Grâce à l'utilisation du framework CodeIgniter 4, nous avons pu mettre en place une structure sécurisée et performante, répondant aux besoins spécifiques du domaine de la bijouterie. Ce projet a également permis d'intégrer des fonctionnalités essentielles telles que l'authentification, l'ajout, la mise à jour et la suppression des différentes entités, tout en offrant une interface intuitive pour l'administrateur. L'objectif est d'optimiser les processus internes de la bijouterie, de garantir une gestion efficace de l'inventaire et de contribuer à une meilleure organisation et prise de décision. Ce système représente une base solide pour de futures évolutions et améliorations, tout en répondant aux exigences actuelles du marché.