



InnovShop

Site e-commerce Symfony

Projet réalisé dans le cadre de la présentation au
Titre professionnel Développeur Web et Web Mobile

BAYLE Julien 03 Novembre 2025
Global Digital University

Sommaire

1. Introduction - Page 4
2. Contexte du projet - Page 5
3. Objectifs et enjeux - Page 6
4. Fonctionnalités (Front-Office) - Page 7
5. Fonctionnalités (Back-Office) - Page 10
6. Architecture technique (MVC) - Page 14
7. Logique métier côté serveur - Page 16
8. Outils et organisation du projet - Page 19
9. Déploiement et hébergement - Page 21
10. Bilan et perspectives - Page 23
11. Annexes (diagramme UML, captures d'écran et liens) - Page 24

1-Introduction

Depuis aussi loin que je me souvienne, j'ai toujours été attiré par l'univers de l'informatique et des nouvelles technologies.

Je n'avais pourtant jamais eu l'occasion d'aller voir « l'envers du décor » et de toucher réellement au code.

Tout cela m'a toujours fasciné : certaines personnes parviennent à créer, avec des lignes de code qui n'avaient pour moi aucun sens, des applications et des sites web complets.

Le déclic a eu lieu lorsque j'ai voulu créer mes propres sites. C'est à ce moment-là que j'ai commencé à me renseigner sur les différentes façons de concevoir une application.

J'ai d'abord essayé le no-code avec FlutterFlow : je ne comprenais pas encore tout, mais j'aimais déjà cette sensation de partir d'une page blanche et de construire visuellement un projet.

Ensuite, je me suis lancé en autodidacte dans l'apprentissage de Flutter, mais sans accompagnement ni encadrement, l'aventure s'est vite révélée compliquée.

Sur les conseils d'un ami développeur, j'ai ensuite créé un premier site avec WordPress. J'ai apprécié la simplicité d'utilisation et la rapidité avec laquelle on peut mettre un site en ligne.

Toutes ces expériences m'ont conforté dans l'idée que j'aimais profondément créer des sites web et que je voulais en apprendre davantage — comprendre les rouages et maîtriser la technique.

C'est ainsi que j'ai décidé d'intégrer une formation professionnelle en développement web, afin d'acquérir des bases solides.

Mon apprentissage s'est fait principalement par la pratique, en commençant par HTML, CSS et JavaScript pour les sites statiques, avant de poursuivre avec PHP et Symfony pour concevoir des sites dynamiques et complets.

Aujourd'hui, après six mois de formation, je suis fier de vous présenter InnovShop, mon projet e-commerce développé avec Symfony.

2-Contexte du projet



J'ai eu l'opportunité de concevoir et développer un projet complet intitulé InnovShop, une plateforme e-commerce réalisée avec le framework Symfony.

Le projet m'a été confié par Madame Leroy, directrice de la start-up InnovShop, spécialisée dans la vente de produits technologiques innovants.

Face à une demande croissante et à la nécessité de digitaliser son offre, l'entreprise souhaitait mettre en place une solution e-commerce moderne et performante, capable de faciliter la vente en ligne de ses produits et d'offrir à ses clients une expérience fluide et intuitive.

Mon rôle était celui d'un développeur web freelance chargé de concevoir, développer et déployer l'application complète.

J'ai ainsi travaillé sur la modélisation de la base de données, la création du front-office et du back-office, la gestion des utilisateurs et des commandes, ainsi que l'intégration d'un module de paiement Stripe.

3-Objectifs et enjeux

Le principal objectif du projet est de concevoir une boutique en ligne moderne, intuitive et responsive, afin d'offrir aux utilisateurs une expérience d'achat agréable et fluide.

L'application doit être simple à comprendre, accessible à tous, et adaptée à tout type d'écran, qu'il s'agisse d'un ordinateur, d'une tablette ou d'un smartphone.

D'un point de vue fonctionnel, le site doit permettre à un utilisateur de :

- naviguer facilement dans le catalogue de produits,
- consulter les détails d'un article (description, prix, image),
- ajouter ou retirer des produits de son panier,
- créer un compte et se connecter de manière sécurisée,
- passer une commande complète et recevoir un récapitulatif par email.

Sur le plan des enjeux, le projet a pour but de :

- améliorer la satisfaction des clients grâce à une interface claire et agréable,
- augmenter les ventes de la start-up InnovShop en ouvrant un canal de vente en ligne,
- et moderniser l'image de l'entreprise, en mettant en avant son identité technologique et son dynamisme.

4-Fonctionnalités (Front-Office)

Le front-office correspond à la partie visible par les utilisateurs, c'est-à-dire l'interface avec laquelle les clients interagissent pour consulter les produits et effectuer leurs achats.

Le développement du front-office a été réalisé à l'aide de Symfony, associé à Twig et à Bootstrap pour la partie graphique et responsive design.

4.1 Page d'accueil

La page d'accueil met en avant :

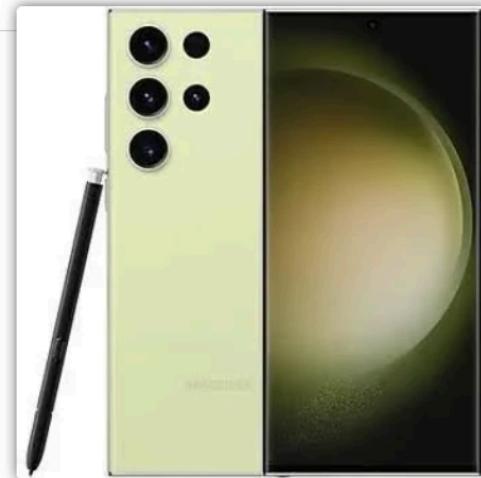
- L'ensemble des produits proposés sur le site avec un carousel à défilement automatique ;
- Les quatre derniers produits ajoutés ;
- Une mise en page visuelle avec images, titres et boutons d'action clairs.

Cette page a été pensée pour attirer immédiatement l'attention de l'utilisateur et lui donner envie de découvrir le catalogue.



4.2 Dernières infos

- Un espace où le client pourra retrouver toutes les diverses actualités du site.



Profitez du black Friday

Ce vendredi 28 novembre, ne ratez pas la mise en vente exceptionnelle du Galaxy S23 à prix cassé

[EN SAVOIR PLUS](#)



Nouvelles lunettes stock limité

📍 Édition ultra-limitée : les Ray-Ban Connect arrivent !

Alliez style iconique et technologie de pointe avec les nouvelles **Ray-Ban Connect**. Capturez vos moments, écoutez votre musique, passez vos appels — tout ça sans sortir votre téléphone. Disponible en **quantité très limitée**, cette collection ne reviendra **jamais**.

👉 Ne laissez pas filer l'occasion d'être **parmi les premiers à voir le monde autrement**.

4.3 Catalogue de produits

Le catalogue affiche l'ensemble des produits disponibles à la vente.
Les fonctionnalités principales sont :

- Affichage des produits sous forme de cartes visuelles (image, nom, prix) ;
- Deux boutons incitant à l'action, soit pour en savoir plus soit pour ajouter le produit au panier ;
- Possibilité de faire une recherche ou d'imposer des contraintes de prix.
- Filtrage par catégorie (Ordinateurs Portables, Smartphone ...) ;

Le catalogue permet de parcourir facilement les articles et de préparer son panier avant l'achat.

4.4 Gestion du panier

Le panier est accessible à tout moment depuis le menu principal.

Il permet :

- D'ajouter et de supprimer des produits ;
- De visualiser le montant total ;
- D'accéder directement à l'étape de commande.

La mise à jour du panier est gérée en temps réel, garantissant une expérience fluide pour l'utilisateur.

4.5 Espace client

Chaque utilisateur authentifié dispose d'un espace personnel lui permettant de :

- Consulter son historique de commandes ;
- Modifier ses informations personnelles et ses adresses de livraison ;
- Suivre l'évolution de ses commandes en cours.

Cet espace renforce la relation de confiance avec le client et facilite le suivi post-achat

The screenshot shows a user profile interface for 'Julien Bayle' from 'Client InnovShop'. At the top left is a circular logo with 'IS'. To the right, the name 'Julien Bayle' and 'Client InnovShop' are displayed. Below this, there are three tabs: 'Mes infos' (selected), 'Commandes à venir', and 'Commandes terminées'. On the left, a sidebar titled 'Raccourcis' lists: 'Mon compte', 'Changer mon mot de passe', 'Mes adresses', and 'Ajouter une adresse'. The main content area displays personal details: 'Nom complet' (Julien Bayle), 'Email' (baylejulien@hotmail.fr), and 'Téléphone' (0627365170). At the bottom right of this area are two buttons: 'Gérer mes adresses' and 'Modifier mes infos'.

5. Fonctionnalités – Back-Office

Le back-office correspond à la partie réservée aux administrateurs du site.

Il permet à l'entreprise InnovShop de gérer son catalogue de produits, ses utilisateurs et les commandes directement depuis une interface simple et sécurisée.

Cette interface d'administration a été mise en place à l'aide du bundle EasyAdmin de Symfony, permettant de générer rapidement un espace de gestion administrateur.

5.1 Connexion et accès sécurisé

L'accès au back-office est réservé aux utilisateurs disposant du rôle administrateur (ROLE_ADMIN).

Lorsqu'un administrateur se connecte :

- il est redirigé vers le tableau de bord d'administration,
- l'accès est protégé par authentification et vérification des droits d'accès.

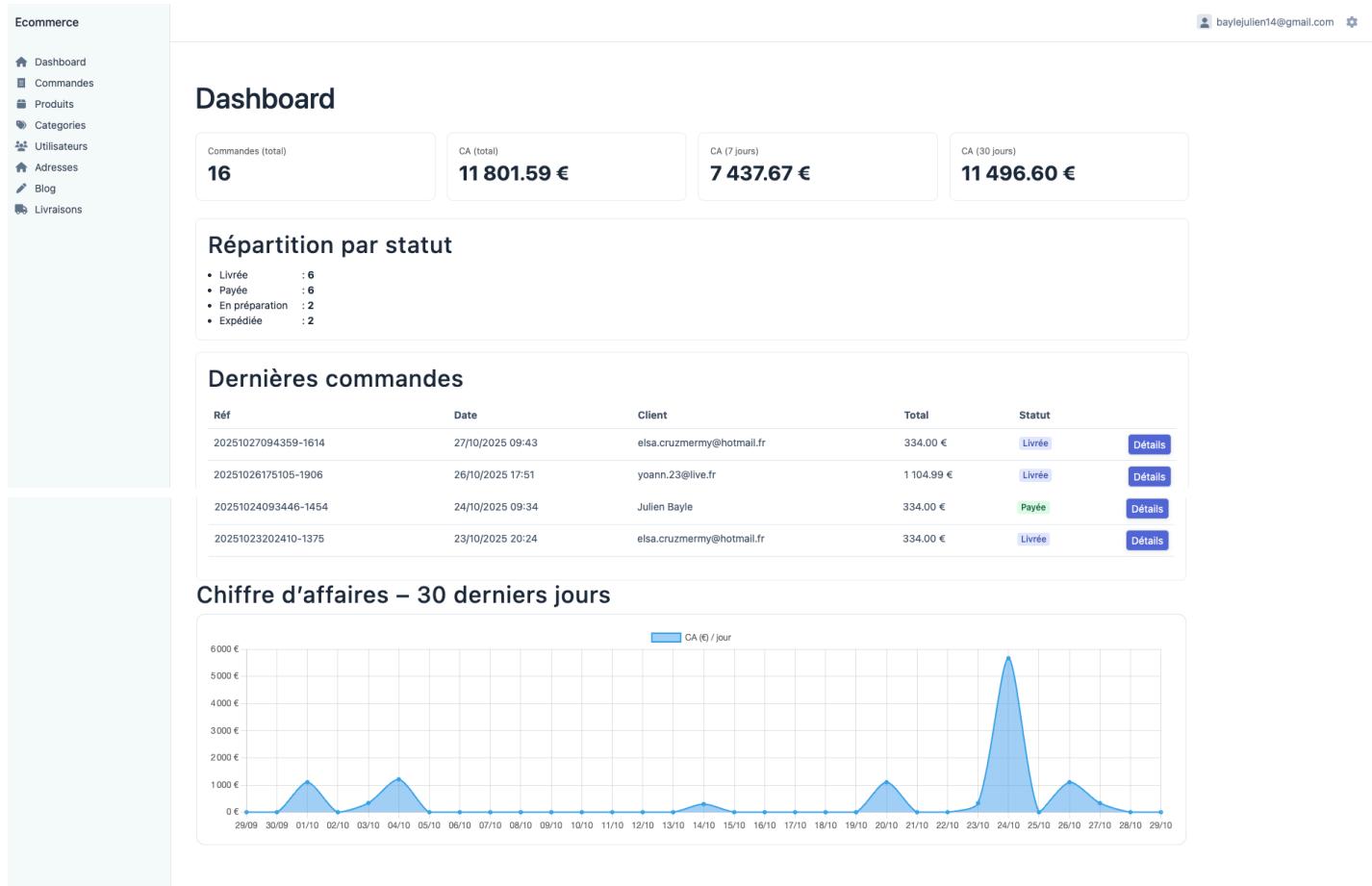
Cette séparation claire entre les rôles garantit la sécurité du site et évite toute modification non autorisée des données.

5.2 Tableau de bord (Dashboard)

La page d'accueil du back-office présente un tableau de bord global, offrant une vue d'ensemble sur :

- le nombre total de commandes enregistrées et sur le chiffre d'affaire généré,
- le nombre de commandes et leur différents statuts (Payées ,en préparations ,expédiées ...)

Ce tableau de bord permet à l'administrateur d'accéder rapidement aux principales sections de gestion.



Capture d'écran du Dashboard

5.3 Commandes

Cet onglet permet à l'administrateur de :

- visualiser la liste complète des commandes,
- consulter le détail de chaque commande (client, adresse, produits achetés, montant total),
- mettre à jour le statut (en attente, expédiée, livrée, annulée).

Cette section facilite le suivi logistique et permet de gérer efficacement les ventes.

5.4 Produits

L'administrateur peut :

- ajouter, modifier ou supprimer un produit,
- renseigner son nom, prix, description, image
- Indiquer le nombre de produit en stock

Nom de la catégorie ↑	Description	...
Accessoires (coques, câbles, chargeurs, écouteurs)	Aucun(e)	...
Lunettes Connectées	Aucun(e)	...
Ordinateurs portables & PC	Aucun(e)	...
Smartphones	Aucun(e)	...
Tablettes & liseuses	Aucun(e)	...

- l'associer à une catégorie.

5.4 Catégories

Cette section permet de :

- créer, renommer ou supprimer une catégorie de produits,
- organiser le catalogue pour faciliter la navigation côté utilisateur.

Chaque produit est lié à une catégorie, assurant une hiérarchie claire dans la boutique.

5.5 Utilisateurs

Cet onglet regroupe l'ensemble des comptes clients.

L'administrateur peut :

- consulter les informations d'un utilisateur,
- et modifier son rôle (passage en administrateur).

Cette fonctionnalité permet un suivi précis des clients et une gestion centralisée des accès.

5.6 Adresses

L'administrateur a la possibilité de consulter les adresses créées par les utilisateurs.

Ces informations permettent :

- Pouvoir les modifier sur demande du client

5.7 Blog

Un onglet "Blog" a été intégré pour permettre à InnovShop de communiquer directement avec ses utilisateurs. On peut envisager la publication des arrivages. Des nouveautés à venir. Parler d'une promotion

...

5.8 Livraisons

Cette section permet de créer et de proposer différents types de livraison. Activable ou désactivable.

Titre du transport	Tarif	Délai	Actif
Colissimo à domicile en France	10,00 €	3-6 j	On
Colissimo Point Relais en France	5,00 €	3-6 j	On
Envoi International	20,00 €	2-5 j	On
Retrait en main propre	0,00 €	0 j	Off

6. Architecture technique (MVC)

Ce projet a été conçu en utilisant le framework Symfony. La structure de ce projet est sur la base du modèle MVC (Model View Controller) qui offre plusieurs avantage. En séparent clairement les taches il rend le code plus facile à comprendre et donc à maintenir dans le temps. Chaque tache est attribué à un élément. La partie « Model » va s'occuper de tout ce qui est communication avec la base de données. La « view » sera tout ce qui est visuel, tout ce que voit le client. Pour finir le « controller » sera un peu le chef d'orchestre, il déclenchera les comportements en fonctions des actions de l'utilisateur.

Tout a été développé en PHP 8.

Pour la gestion des données, j'ai utilisé MySQL.

Les entités de Symfony (par exemple : *Produit*, *Commande*, *Utilisateur*, *Adresse*) sont directement liées aux tables de la base de données grâce à l'ORM Doctrine. Chaque entité est synchronisée avec sa table lors de « migrations »

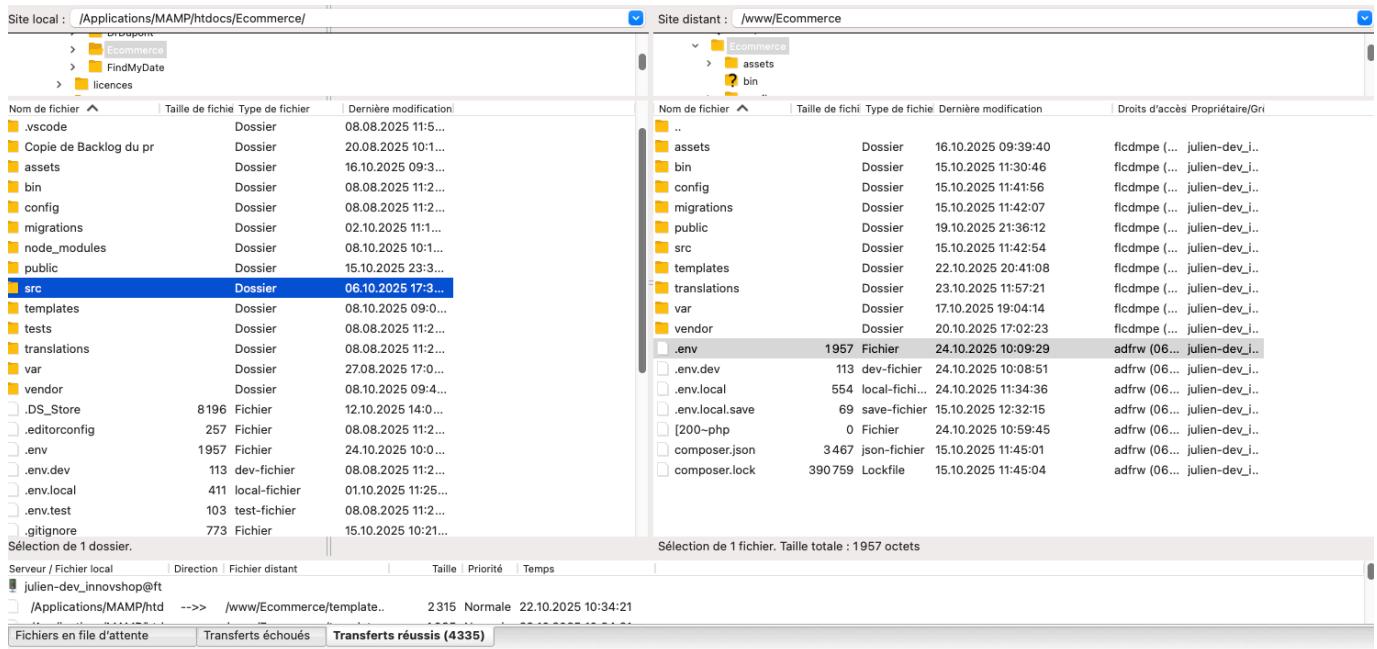
The screenshot shows the MySQL Workbench interface. On the left, the database structure is displayed with various tables and their relationships. On the right, a detailed table statistics window is open, showing the following data:

Table	Action	Rows	Type	Collation	Size	Overhead
adresse	Browse Structure Search Insert Empty Drop	7	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
article	Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
article_commande	Browse Structure Search Insert Empty Drop	31	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
article_panier	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
blog	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
category	Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
commande	Browse Structure Search Insert Empty Drop	16	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
doctrine_migration_versions	Browse Structure Search Insert Empty Drop	19	InnoDB	utf8mb3_unicode_ci	16.0 KiB	-
panier	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	22.0 KiB	-
produit	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
produit_category	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
shipping_method	Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
user	Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
Sum		99	InnoDB	utf8mb4_general_ci	384.0 KiB	0 B

Pour la partie visible du site, j'ai utilisé Twig, le moteur de templates intégré à Symfony.

Le design a été construit avec Bootstrap, afin d'obtenir un rendu responsive, adapté à tous types d'écrans (ordinateur, tablette, mobile).

Le site est hébergé sur AlwaysData. Les fichiers ont été envoyé en FTP via FileZilla.



La sécurité du site repose sur le Security Bundle.

Ce module gère les rôles et les accès selon le type d'utilisateur :

- un client (ROLE_USER) peut naviguer, passer commande et consulter son espace personnel ;
- un administrateur (ROLE_ADMIN) peut accéder au back-office et gérer les produits, commandes et utilisateurs.

Des contrôles d'accès sont appliqués sur chaque route sensible, et les formulaires sont protégés contre les attaques CSRF.

Les mots de passe utilisateurs sont hachés avant d'être enregistrés en base, garantissant une protection des données personnelles.

7. Logique métier côté serveur

Le cœur du projet repose sur plusieurs composants métier essentiels gérés côté serveur.

Ces éléments garantissent la cohérence des données et le bon déroulement du processus d'achat, depuis le panier jusqu'à la commande finale.

7.1 Calcul automatique du total du panier

À chaque modification du panier (ajout, suppression, ou mise à jour d'une quantité), le total du panier est recalculé automatiquement côté serveur.

Cela permet d'assurer que le montant affiché à l'utilisateur correspond toujours à la réalité des prix en base de données.

Cette approche évite toute manipulation possible des montants depuis le navigateur et garantit une facturation exacte au moment de la commande.

```
#Route(path: '/panier', name: 'app_cart', methods: ['GET'])
8 references
public function index(SessionInterface $session): Response
{
    $cart = $session->get(name: 'panier', default: [
        'id'          => [],
        'nom_produit' => [],
        'prix'         => [],
        'quantite'     => [],
        'image'        => []
    ]);

    return $this->render(view: 'cart/show.html.twig', parameters: [
        'items' => $cart,
    ]);
}
```

7.2 Gestion du stock

Chaque produit dispose d'une quantité disponible (quantiteRestante) enregistrée en base de données. Celle ci est créée au moment de l'ajout d'un produit par l'admin.

Le stock est contrôlé à chaque ajout au panier, lors de la validation de commande et aussi lors d'une éventuelle annulation.

- Lorsqu'un utilisateur ajoute un produit : le système vérifie le stock restant.
Si la quantité demandée dépasse le stock, elle est automatiquement ajustée à la limite disponible.
- Lorsqu'une commande est validée : le stock du produit est décrémenté.
- En cas d'annulation ou de remboursement : la quantité est réintégrée dans le stock.

Ces vérifications côté serveur garantissent que le site ne vend jamais plus d'articles qu'il n'en reste.

7.3 Adresse de livraison figée dans la commande

Au moment du passage de commande, l'adresse de livraison choisie par le client est enregistrée et figée dans l'entité Commande.

Cela permet de conserver une trace exacte de l'adresse utilisée lors de l'achat, même si l'utilisateur modifie plus tard son profil.

Cette méthode garantit la traçabilité des commandes et une meilleure fiabilité logistique.

7.4 Envoi d'un email de confirmation

Une fois la commande validée et payée (paiement Stripe), un email de confirmation est automatiquement envoyé au client grâce au Mailer Bundle de Symfony.

Cet email contient :

- Un numéro unique de confirmation ,
- le montant total,
- l'adresse de livraison.

Cela clôture le processus d'achat et apporte une preuve d'achat immédiate au client.

L administrateur reçoit également un email l'avertissant qu'une commande a été passée.

 **Boutique InnovShop** Boîte d...Exchange 24 octobre 2025 à 22:14
Confirmation de votre commande #20251024221434-5586
À : julien bayle

✓ Confirmation de votre commande #20251024221434-5586

Bonjour Julien,

Merci pour votre achat ! Votre commande a bien été enregistrée le 24/10/2025 22:14.

Total : 334.00 €

Statut : Confirmée

Adresse de livraison :
Julien Bayle
36 Allée des tilleuls
30400 Villeneuve les avignons (FR)

Détails

Vous pouvez retrouver vos commandes dans votre espace compte.

Email de confirmation client après commande

7.5 Rôles et autorisations

Le système de rôles gère les autorisations selon le type d'utilisateur :

- Les clients (ROLE_USER) peuvent naviguer, gérer leur panier, passer une commande et accéder à leur espace personnel.
- Les administrateurs (ROLE_ADMIN) disposent d'un accès au back-office pour gérer les produits, les commandes et les utilisateurs.

Cette distinction garantit un contrôle strict des accès et la sécurisation du back-office.

Toutes les routes sensibles sont protégées via le Security Bundle de Symfony.



Un client connecté mais ne possédant pas les autorisations d'administrateur

8. Outils et organisation du projet

8.1 GitHub

Le code du projet a été hébergé sur GitHub afin d'en assurer la sauvegarde et la mise à disposition en ligne.

8.2 Choix de la plateforme – AlwaysData

J'ai choisi AlwaysData pour héberger mon site e-commerce.

Cet hébergeur français offre la possibilité d'héberger facilement la partie Back et Front de mon application.

L'hébergement m'a permis de :

- déployer mon application Symfony sur un serveur distant,
- créer une base de données MySQL pour les produits, utilisateurs et commandes,
- gérer mes fichiers via FTP ou SSH,

Grâce à cette solution, le site est accessible depuis <https://julien-dev.alwaysdata.net/innovshop/>

Cette séparation garantit la sécurité des données sensibles, notamment les clés d'API et les identifiants de base de données.

Seul le fichier .env de développement est visible sur GitHub, tandis que les autres restent privés et protégés.

The screenshot shows the AlwaysData web interface. At the top, there's a navigation bar with the logo and the text "alwaysdata". Below it, a secondary navigation bar shows "Home / Bases de données MySQL". On the left, a sidebar menu is open under the heading "julien-dev". The menu items include:

- Espace disque
- Web
 - Sites
 - Analytics
 - Configuration
- Domaines
- Emails
 - Adresses
 - Listes de diffusion
 - Historique
 - Configuration
- Bases de données
 - MySQL »
 - PostgreSQL
 - RabbitMQ

In the main content area, there's a yellow box containing information about the MySQL host: "Hôte MySQL : mysql-julien-c" and "Version : 10.11 (mariadb)". Below this, a "phpMyAdmin" link is visible. To the right, a "BASES DE DONNÉES" section is shown with a table:

Nom
julien-dev_dupont
julien-dev_innovshop

Interface AlwaysData

9. Déploiement et mise en ligne

La mise en ligne du projet InnovShop a été une étape essentielle pour tester l'application dans des conditions réelles d'utilisation et la rendre accessible depuis n'importe quel appareil connecté à Internet.

9.1 Configuration du serveur et de la base de données

Une base de données MySQL a été importée directement depuis l'interface d'administration d'AlwaysData.

Cette étape a permis de lier l'application à la base distante et de préparer l'hébergement à recevoir les données produits, commandes et utilisateurs.

9.2 Transfert des fichiers via FileZilla

Une fois le projet prêt en local, j'ai transféré tous les fichiers du projet Symfony sur le serveur en utilisant FileZilla.

Le répertoire `public/` a été configuré comme racine du site afin que les fichiers soient accessibles depuis l'URL en ligne.

9.3 Configuration du fichier .env

Le fichier `.env` a été adapté pour le déploiement :

- et les clés sensibles (notamment Stripe et App Secret) ont été masquées et déplacées dans un fichier `.env.local`, non transmis sur GitHub.

Cette séparation garantit la sécurité des informations sensibles tout en permettant à Symfony de charger la bonne configuration selon l'environnement (local ou production).

9.4 Tests et validation

Avant la présentation, j'ai testé le site sur plusieurs appareils (ordinateur, tablette, smartphone) pour m'assurer que l'interface soit bien responsive et que les fonctionnalités principales (connexion, panier, commande, paiement test) fonctionnent correctement.

9.5 URL du site en ligne

Le site est accessible via une URL publique.

Cette mise en ligne permet de le tester depuis n'importe quel appareil connecté.

10. Bilan et perspectives

Ce projet m'a permis de mettre en pratique l'ensemble des compétences acquises au cours de ma formation.

J'ai pu voir toutes les étapes de la mise en ligne d'un site Ecommerce développé avec un framework. L'utilité des bundles (mailer, security, easy admin, l'utilisation de snippets bootstrap.

Ce projet m'a aussi permis de comprendre l'importance d'une bonne organisation du code.

Les prochaines évolutions que j'aimerais apporter au site InnovShop sont :

- La mise en place d'un système d'avis clients.
- La mise en place d'un système de réduction par code.

12. Annexes

(diagramme UML, captures d'écran et liens)

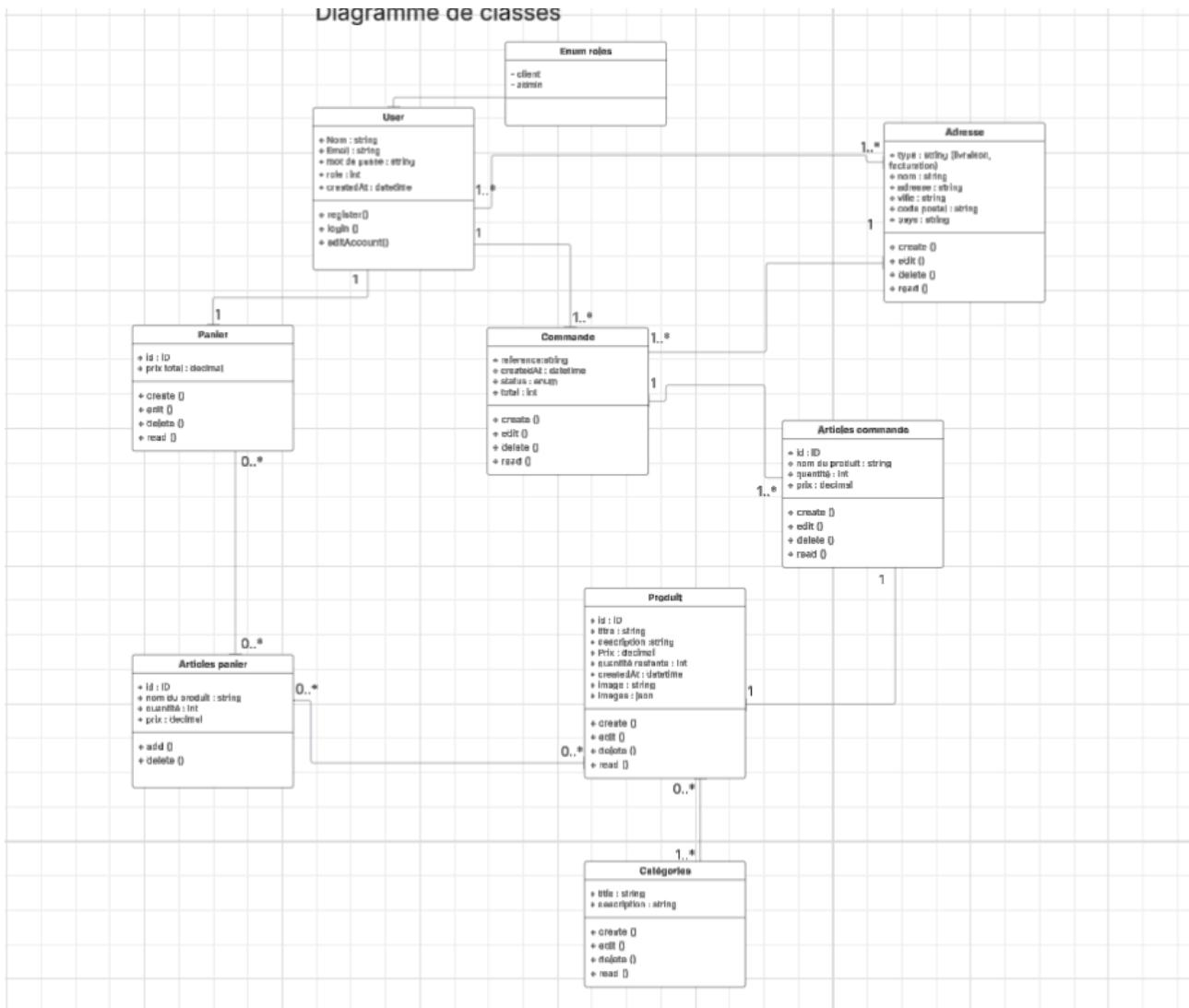


Diagramme UML Initial

Les repos du projet sont accessibles à cette adresse: <https://github.com/Ju-lien-dev>

Le projet est accessible à l'adresse suivante: <https://julien-dev.alwaysdata.net/innovshop/>