



2023

RAPPORT PROJET WEB

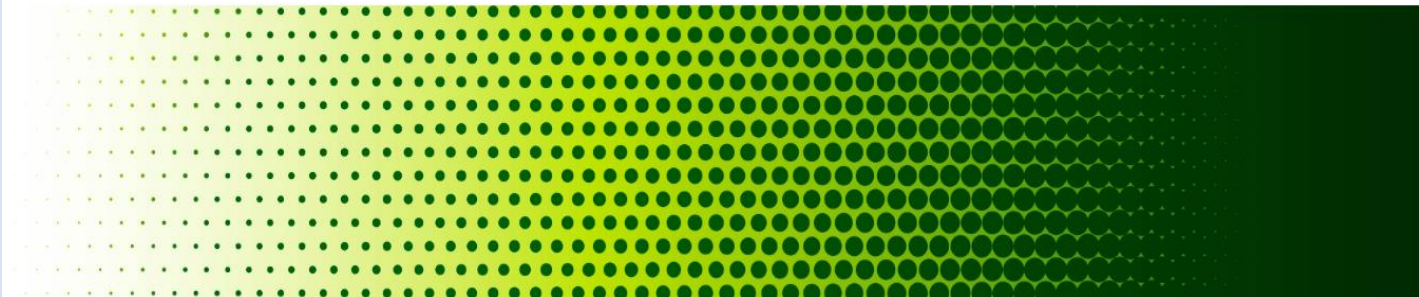
Accueil [M'inscrire](#) [Me connecter](#)

You did it! You updated the system! Amazing!

Site de e-commerce

Bienvenue sur le site de e-commerce

Vous êtes connecté en tant qu'anonyme



Loris Benaitier

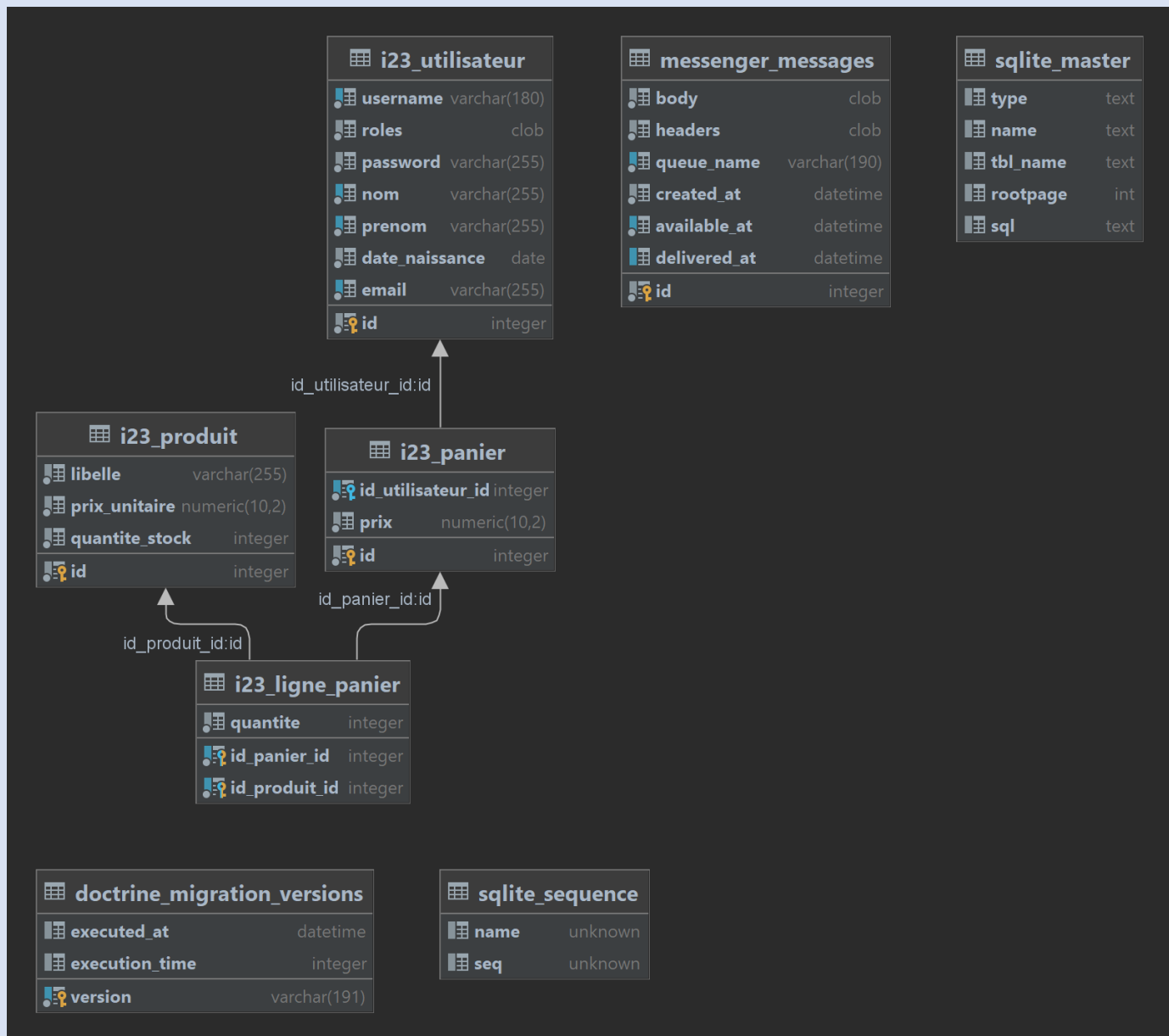
TD1, TP1

10/04/2023

Table des matières

1.	Diagramme de base de données	2
2.	Organisation du code	3
3.	Tutoriel pour la création d'un Service sur Symfony	4
4.	Remarques/Conclusion	6

1. Diagramme de base de données



Voici la conception de ma base de données, elle est assez simple et permet d'accéder aux données assez facilement. La particularité est que j'ai choisi de faire une table de plus que j'appelle i23_ligne_panier qui permet de stocker le contenu d'un panier. Cet ajout me permet notamment de pouvoir retrouver pour chaque panier, tous les produits qui y sont contenus avec leur quantité.

2. Organisation du code

- Dossier « dbvente »
Ce répertoire contient la base de données que j'ai nommé mybase.db
- Dossier « public/assets »
Ce répertoire contient 3 sous-répertoires (CSS, js, image) qui me permettent de stocker le code bootstrap qui est lui utilisé pour le design global du site internet. Enfin, le répertoire image contient 2 sous-répertoires (header, footer) qui stocke les images utilisées par le site internet.
- Dossier src/Controlleur
 - FormControlleurControlleur.php : cette classe permet la gestion des différents formulaires utilisés sur le site de e-commerce.
 - GestionUserController.php : cette classe permet de gérer la partie du site internet qui permet de voir et supprimer un utilisateur.
 - MainControlleur.php : cette classe permet de gérer l'accueil du site internet.
 - PanierControlleur.php : cette classe permet de gérer toutes les fonctionnalités liées au panier (affichage des produits dans son panier, suppression des produits, commandes et vider son panier, affichage du prix total et du nombre d'articles)
 - ProductsControlleur.php : cette classe permet d'afficher tous les produits disponibles sur le site internet. Elle permet aussi de gérer toutes les fonctionnalités il y a cette page notamment le fait de pouvoir choisir un nombre de produits spécifiques grâce à la liste déroulante.
 - RegistrationControlleur.php : cette classe permet de gérer l'inscription d'un utilisateur.
 - SecurityControlleur.php : cette classe permet de gérer le fait de pouvoir se déconnecter et se connecter du site internet.
- Dossier src/entity :
 - I23LignePanier.php : cette classe représente la table ligne panier afin de stocker les produits contenus dans un panier pour un utilisateur.
 - I23Panier.php : cette classe représente la table panier afin de stocker tous les paniers appartenant chacun à un utilisateur spécifique.
 - I23Produit.php : cette classe représente la table produit qui permet de stocker tous les produits existant sur le site internet.
 - I23Utilisateur.php : cette classe représente la table l'utilisateur il permet de stocker tous les utilisateurs qui ont créé un compte sur le site internet où qui ont un rôle sur le site internet.
- Dossier src/Form :

- I23ProduitType.php : cette classe représente un modèle de formulaire hydraté sur l'objet produit qui est la table produit. Elle permet d'ajouter un nouveau produit dans la base de données.
- I23UtilisateurType.php : cette classe représente un formulaire hydraté sur l'objet utilisateur c'est en fait la table utilisateur.
- I23RegistrationFormType.php : cette classe représente un modèle de formulaire hydraté sur l'objet utilisateur qui est en fait la table utilisateur. Elle permet notamment de pouvoir créer un formulaire pour créer un compte utilisateur ou modifier un compte utilisateur.
- src/Repository :
 - Il y a autant de classe dans ce répertoire qu'il y a dans le répertoire entity. Ce sont ces classes qui nous permettent de manipuler les différentes tables contenues dans la base de données dans Symfony en tant qu'objet. On peut également utiliser EntityManagerInterface qui est une alternative.
- src/Security :
 - La classe contenue à l'intérieur de ce répertoire permet de gérer la sécurité lors de l'authentification d'un utilisateur sur le site internet (hachage mot de passe...).
- src/Service :
 - Ce répertoire contient une classe qui m'a permis de créer un service au sens propre sur Symfony. Ce service permet entre autres de générer un message de bienvenue de façon aléatoire à chaque fois qu'on est sur la partie accueil du site.
- Template :
 - Tous les fichiers et dossiers contenus dans ce répertoire représente en fait tout l'affichage qu'on peut voir sur le site internet.

3. Tutoriel pour la création d'un Service sur Symfony

Voici un tutoriel rapide pour créer un service sur Symfony :

1. Créer un dossier pour votre service dans le répertoire src/ :

```
...
mkdir src/Service
...
```

2. Créer un fichier pour votre service :

```
```\ntouch src/Service/MonService.php\n```\n
```

## 3. Dans ce fichier, créer une classe pour votre service :

```
```\n<?php\n\nnamespace App\Service;\n\nclass MonService\n{\n    public function maMethode()\n    {\n        // Votre code ici\n    }\n}\n```\n
```

4. **Optionnelle** -> Ajouter votre service au conteneur de services de Symfony en créant un fichier dans config/services.yaml :

```
```\nyaml\nservices:\n    App\\Service\\MonService:\n        autowire: true\n```\n
```

## 5. Injecter votre service dans un contrôleur ou une autre classe :

```
```\n<?php\n\nnamespace App\\Controller;\n\nuse App\\Service\\MonService;\nuse Symfony\\Bundle\\FrameworkBundle\\Controller\\AbstractController;\nuse Symfony\\Component\\HttpFoundation\\Response;\n\nclass MonController extends AbstractController\n{\n
```

```

public function maRoute(MonService $monService): Response
{
    $resultat = $monService->maMethode();

    // Votre code ici
}
}
...

```

Et voilà, votre service est prêt à être utilisé ! Vous pouvez maintenant l'appeler depuis n'importe quelle classe en l'injectant comme dépendance.

4. Remarques/Conclusion

En inspectant mon site de e-commerce, vous remarquerez sûrement que j'ai designé mon site avec l'aide notamment de bootstrap. Toutes les fonctionnalités demandées sont présentes sur le site avec les différentes exigences spécifiées dans le cahier des charges. Cependant, il y a quelques points qui ont posé problèmes ou qui peuvent être améliorés :

- Dans les consignes, il était bien demandé que le nom des tables devez commencer par I23 mais dans symfony, on ne devait pas avoir ce préfixe. Je l'avais vu mais ne sachant pas comment faire au début du projet, je me suis dit que j'allais le changer plus tard et effectivement, j'ai fait l'erreur d'attendre trop puisque le moment où j'ai compris comment il fallait faire, il y avait beaucoup trop de choses à changer dans le code. J'ai bien sûr essayé mais ça m'a créé des erreurs que je n'ai pas réussi à résoudre donc j'ai préféré laisser comme ça et plutôt l'évoquer dans le rapport.
- Le nommage de certains champs dans les classes entity qui représentent donc les tables de la base de données sont parfois mal choisis avec notamment les clés étrangères qui sont en fait des objets mais que j'ai nommé par exemple « id_utilisateur ». On comprend assez vite que ce nommage peut être assez problématique surtout que ça m'a parfois causé des erreurs car très souvent, je pensais utiliser des id mais ce n'était pas des id mais plutôt des objets. J'ai bien pensé à le changer mais encore une fois le code était trop avancé pour que je me permette de tout changer.
- Lorsqu'on veut changer les informations de notre profil sur le site web, tout fonctionne sauf le changement de mot de passe qui lui, ne fonctionne pas. Actuellement je n'ai pas encore trouvé pourquoi ça ne fonctionnait pas puisque le changement de toutes les autres informations fonctionne sauf celui-là ce qui est assez bizarre.
- La version de Symfony que j'ai utilisée est la version 6.0.2. J'avais vu avec vous mais j'ai eu un problème avec la version de PHP qui est sur mon ordinateur

donc j'ai été obligé de rester sur cette version de symfony. J'ai également du rajouter un fichier nommé .php-version pour forcer symfony a utilisé la version de PHP 8.1 car sinon j'avais des bugs lors de la migration de ma base de données.