

Symfony Projet

La Maison (boutique de vêtements)

Vous allez réaliser sur deux semaines des pages pour présenter des produits d'une boutique de vêtements.

Une partie « administrateur » sera demandée afin de permettre d'ajouter un produit associé à une catégorie « homme » ou « femme ». Une option «code produit» pour chaque produit pourra être associé à un vêtement en particulier, les valeurs seront **solde** ou **normale**.

La table client dans le projet contiendra des données d'exemple, vous ne devez pas développer un système d'authentification pour commander un produit, la partie commande de produit n'est pas demandée dans ce projet. Par contre vous mettrez des données d'exemple pour simuler des achats de produits.

La partie authentification est optionnelle, vous pouvez la réaliser en vous appuyant sur la documentation officielle, voir les liens qui suivent, si vous avez assez de temps :

<https://symfony.com/doc/current/security.html>

https://symfony.com/doc/current/security/form_login_setup.html#generating-the-login-form

Nous vous conseillons de lire le document entièrement avant de commencer à coder. Faites un plan des « features » à développer et essayez de les planifier à l'aide de Trello (todo, done & doing).

Documents à remettre et à présenter (livrable) :

*Le schéma des relations des données de la base de données (png), le schéma de votre organisation (png) et le code de l'application **sans le dossier vendor** ou un lien vers un dépôt GitHub par exemple à l'adresse suivante : antoine.lucsko@gmail.com*

Calendrier

Lancement du projet : lundi et fin de la deuxième semaine vendredi.

Contexte technique

Vous utiliserez le framework PHP **Symfony** ainsi que **MySQL** pour la partie persistance des données et **Doctrine** pour l'ORM, vous créez des données d'exemple en utilisant les fixtures de Doctrine et le composant **Faker** pour les patterns d'exemple de données. Vous utiliserez **Twig** pour le moteur de templating et vous utiliserez le framework HTML/CSS **Bootstrap** pour l'intégration visuelle des pages.

Les données

Pour les données voici des schéma simplifiés des données. Vous êtes libre de les modifier si nécessaire. Vous devez implémenter les relations entre les entités à l'aide de Doctrine, lisez bien le projet avant de commencer. Vous devez dessiner les relations que vous proposer entre les entités, inspirez vous du schéma d'une base de données en UML vu en cours.

Table product

Données principales pour la table produit voyez ci-dessous dans le projet si vous devez ajouter des champs (voir la page d'ajout d'un produit dans la partie backoffice).

product					
id	name	price	description	publishedAt	status
Integer	String	Decimal	Text	Datetime	Enum

Table categorie

Table categories, chaque produit sera dans une catégorie au plus et une catégorie aura de 0 à N produit.

category		
id	name	description
Integer	String	text

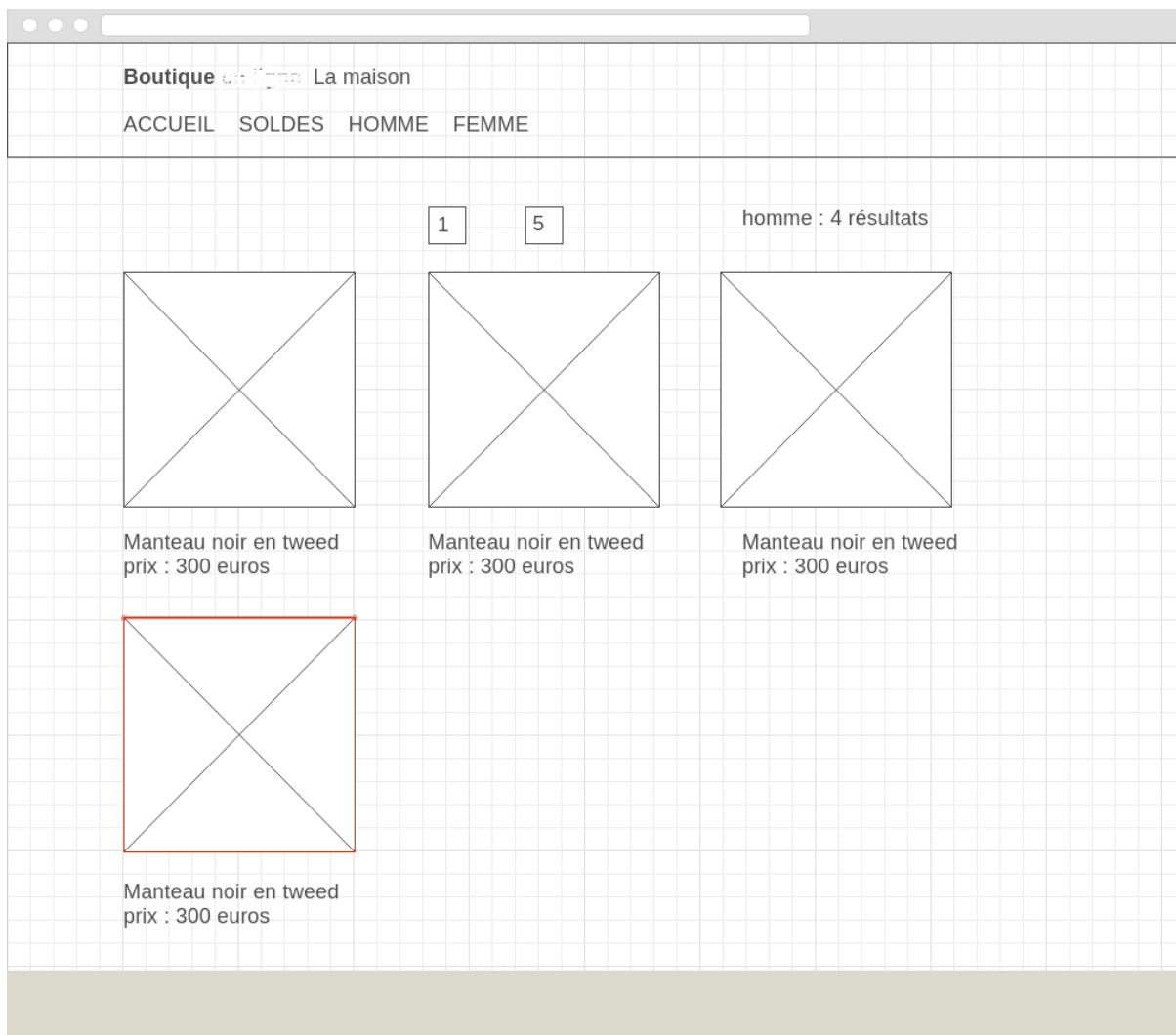
Table client

Cette table permettra d'enregistrer les utilisateurs du site ou les clients et d'enregistrer les commandes passées.

Un client pourra faire plusieurs commandes et un produit pourra être commandé par plusieurs clients. Dans ce but vous pouvez créer si vous le souhaitez une table Commande qui implémentera ces relations entre le client et les commandes.

client			
Id	Email	Address	telephone
Integer	String	Text	string

Page d'accueil



La page d'accueil affichera les derniers produits maximum 6 et si plus vous mettrez en place de la pagination (6 produits par page).

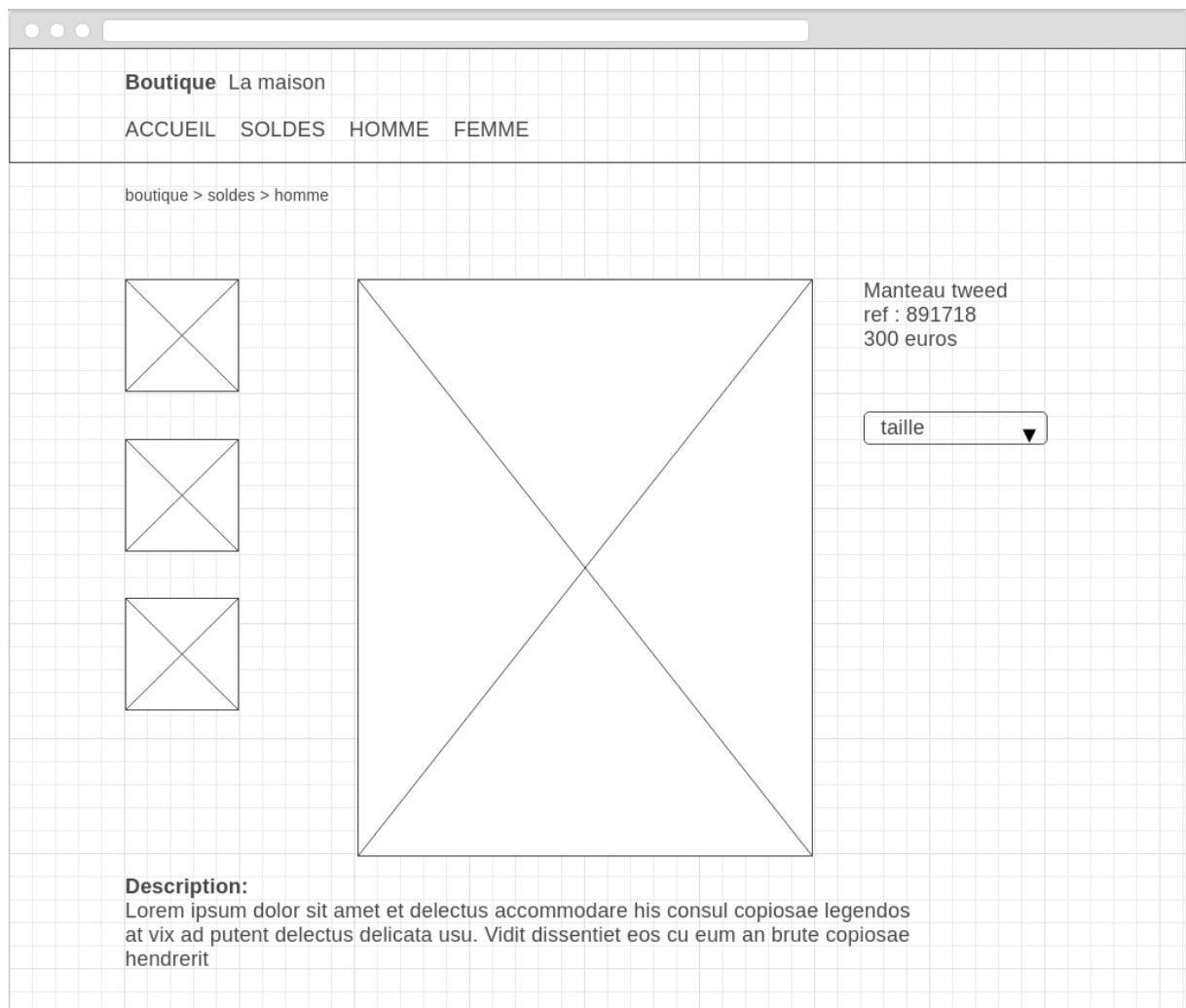
Chaque titre et produit seront cliquables et mèneront sur la page du produit en question.

On indiquera le nombre de produit(s) en haut à droite de cette page, ce nombre correspondra à la totalité des produits publiés.

Menu principal

Les items de ce menu sont respectivement : un lien vers la page d'accueil, un lien vers les catégories « homme » et « femme ». Un item « solde » correspond aux produits, toute catégorie confondue, mais dont le code produit est précisément « solde », la page des produits soldés reprendra la modèle de la page d'accueil avec la même règle pour la pagination.

Page d'un produit



La page produit affichera une grande image centrée (les images sur la partie de gauche ne sont pas à mettre en place, partie facultative), une description, un titre, une référence, un prix et une taille que l'on pourra consulter, à l'aide d'un « select » HTML, pour se faire une idée des tailles existantes pour cet article (voir dans la partie « Données » les tailles proposées).

Le fil d'Ariane est optionnel vous pouvez si vous avez le temps le développer.

Rappelons qu'il n'y a pas de panier, seule la consultation d'un catalogue est demandée.

Backoffice

Si vous avez le temps vous pouvez faire une partie login/password (partie facultative). Dans ce cas utiliser le code et les templates de Laravel classiques.

On accédera au backoffice à l'aide du préfixe d'uri suivant : **/admin**

Dashboard

Boutique La maison					
retour à l'accueil DASHBOARD AJOUTER UN PRODUIT					
<div></div>					
Nom	Catégorie	Prix	Statut	Mettre à jour	Supprimer
Manteau tweed	Homme	300	publié		
Manteau tweed	Homme	300	publié		
Manteau tweed	Homme	300	publié		
Manteau tweed	Homme	300	publié		
Manteau tweed	Homme	300	publié		
<div></div>					

Vous afficherez l'ensemble des produits sur cette page en précisant le titre du produit la catégorie, le prix, le statut et vous ajouterez deux boutons respectivement un pour mettre à jour et un autre pour supprimer un produit.

La page affichera 10 produits et si plus vous mettez en place la pagination en gardant maximum 10 produits par page.

Menu du dashboard

Celui sera spécifique à cette page et aura les items suivants : un lien vers la page d'accueil, un lien vers la page dashboard ci-dessus et un lien pour ajouter un produit.

Page d'ajout d'un produit

Respectez les champs décrit dans le wireframe ci-dessous, ils correspondent aux données à administrer pour la ressource products en base de données, on vous rappelle que l'image (partie url) sera placée dans un champ de la table products.

Précisons que pour les champs ci-dessous :

Le champ « Taille » est un select, il reprendra les tailles existantes (voir le chapitre ci-dessus « Données »).

Le champ « Code produit » est un select il reprendra les deux codes produits que l'on a défini dans les données : **soldes** et **new** (pour nouvelle collection).

Pour le champ image vous devez mettre en place le téléchargement d'une image unique par produit (la partie images multiples par produit est optionnelle), rappelons qu'il ne faut pas créer de table spécifique dans la base de données pour gérer l'url de l'image, enregistrez le lien de l'image dans le champ « url_image » de la table products.

The wireframe shows a web browser window with the following elements:

- Header:** "Boutique La maison" and navigation links: "retour à l'accueil", "DASHBOARD", "AJOUTER UN PRODUIT".
- Form Fields:**
 - Titre:** A text input field.
 - Description:** A large text area.
 - Prix:** A text input field.
 - Catégorie:** A dropdown menu with "Homme" selected.
 - Taille:** A dropdown menu.
 - Image:** A dropdown menu.
 - Statut:** Radio buttons for "Publié" (selected) and "Brouillon".
 - Code produit:** A dropdown menu.
 - référence produit:** A text input field.
- Buttons:** An "AJOUTER" button at the bottom.

Page d'ajout d'un produit (mise à jour)

La page de mise à jour sera identique à cette page, mais avec les contenus de la ressource sélectionnée dans la partie dashboard comme vu en cours.

Page statistique des achats

Proposez dans le backoffice une page présentant les achats des clients de la boutique, pensez à faire de la pagination si nécessaire (fonction du nombre de vos clients &/ou données à présenter).

BON COURAGE A TOUS !

N'hésitez pas à nous solliciter pour tout complément d'info, conseil ou précision... le plus tôt c'est le mieux !

Le formateur,