

# E-Star-Wars

Sujet PHP

Formation alternance

L'École Multimédia - décembre 2015

**Auteur : Antoine Lucsko**

## Contexte

Vous devez réaliser une e-boutique, sans la partie sécurisé avec une banque . Cette boutique vendra des objets de la saga **Star Wars**.

Ce projet est une version simplifié d'un site d'e-commerce.

Le site de vente en ligne proposera une page d'accueil présentant les 10 derniers articles de la boutique ordonnés par ordre de date décroissante, une page listant les produits d'une catégorie (même principe que la page d'accueil), une page présentant un produit avec la possibilité de le commander, une page panier, une page de confirmation de la commande, une page de contact, une administration sécurisée pour ajouter des produits (CRUD products uniquement) et consulter l'historique des commandes.

Chaque produit sera placé dans une unique catégorie et possédera des mots clefs permettant une navigation horizontale dans l'ensemble des produits.

Lorsqu'un internaute aura commandé un/des produit(s), un icône panier apparaîtra à droite dans la bannière en haut sur le site public.

La page panier permettra de retirer un produit commandé et/ou de terminer la commande (redirection vers la page de confirmation de la commande en cours).

Une fois la commande terminée l'internaute arrivera sur une page qui lui permettra de finaliser sa commande. Pour valider la commande, il devra s'authentifier (sur cette page voir ce qui suit).

## Fichiers transmis

- Ce brief.

## Contraintes techniques

- Vous devez réaliser le projet de 1 à 2 élève(s) au plus.
- Vous devez utiliser Trello en créant les listes suivantes et en les remplissant avec les tâches à réaliser sur l'application, la colonne ideas permettra éventuellement d'ajouter des options à votre projet, dans ce cas vous devez valider avec votre formateur l'option ajoutée à l'application).
  - ideas, todo, doing, done
- Vous utiliserez Git et Github pour versionner votre projet (faites de temps en temps une sauvegarde dans un dossier à part de la totalité de votre projet en dehors du versionning git).
- Vous utiliserez MySQL et vous ferez un diagramme UML, vous pouvez utiliser MySQLWorkbench ou DIA, vous devez mettre dans Trello ce diagramme.
- Vous utiliserez un framework Laravel ou Symfony2
- Vous réaliserez une page type login password pour accéder à l'administration du

site, l'administration des produits sera sécurisée.

- Vous mettrez en place un CRUD ( **Create, Read, Update, Delete**) pour les produits du site, les tags ne seront pas administrables, dans l'administration des produits on pourra cependant associer une liste de tags prédéfinis et une catégorie (cette partie n'est pas administrable).
- Vous mettrez en place l'upload d'une image au maximum par produit.
- Une page permettant d'afficher l'historique des produits commandés par les internautes. L'historique ne sera pas administrable.
- Toutes les actions du CRUD des produits seront accompagnés d'un message de succès ou d'échec, les actions de suppression seront accompagnés d'un message de confirmation avant suppression.
- Vous mettrez en place toutes les bonnes pratiques de sécurité dans l'application.
- Vous devez rendre un code qui respect les standards PSR.
- Vous pouvez utiliser des dépendances à la condition qu'elles soient disponibles sur « **packgists** » et qu'elles soient installées avec Composer dans le projet.
- Toutes les dates affichées sur le site public ou dans l'espace d'administration seront standardisées comme suit :
  - **JJ MM AAAA**
- Vous utiliserez comme base HTML/CSS getskeleton (boilerplate).
- Vous mettrez en place un fichier gulp.js pour minifier les CSS et JS et utiliserez SASS/LESS pour personnaliser le HTML/CSS.
- Pour les tests d'envoi d'email vous utiliserez mailDev, un package NodeJS.

## Les données

- En ce qui concerne les mots clés du site, vous créez à l'avance une liste de mots clés, que vous enregistrerez en base de données dans la table tags. Pour cette partie il n'est pas demandée de faire un CRUD pour les tags. Seuls les produits pourront être associés aux tags à partir du CRUD des produits.
- Pour les catégories vous créez les catégories Lasers et Casques en base de données dans la table categories. Cette partie n'est pas administrable. Seule une catégorie pourra être associée à la création d'un produit dans le CRUD des produits.
- Vous créez également une liste de clients ayant un username unique et email unique avec leurs numéros de carte bleue (numéro unique à 16 chiffres). Lorsqu'un client commandera un produit, vous vérifierez que celui-ci est bien dans la base de données clients (customers) afin de valider une commande.

## Livrable

- Vous devez rendre l'application sans les vendors et facilement déployable sur un serveur de production ; en ce sens un script shell type install.sh pourra être mis en place par exemple.

## Schéma indicatif des données

### users

id
username
password

### products

id
title
abstract
content
Datetime
status

### images

id
name
uri
Datetime
status

### tags

id
name

### categories

id
title
description

### customers

id
username
email

## histories

id
total
commanded_at
...

### Indications sur les relation entre les tables de la base de données.

Un produit peut avoir plusieurs tags et un même tag peut se trouver associé à plusieurs produits.

Un produit se trouvera dans une catégorie et une catégorie possédera plusieurs produits.

Un produit possède une image au maximum.

Un client (table customers) peut effectuer plusieurs commandes, un produit peut être commandé plusieurs fois.

### Objectif du site public

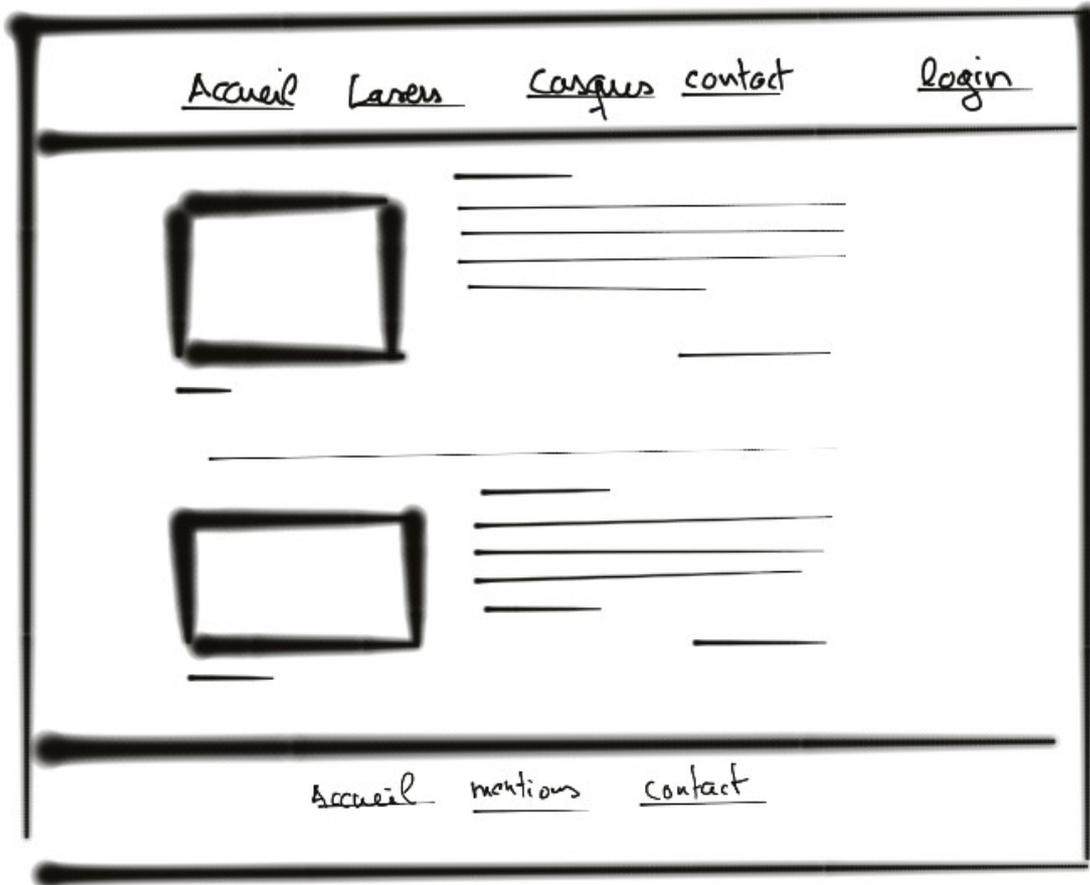
Vous devez réaliser les pages suivantes :

- Page d'accueil.
- Page d'un produit, détails du produit.
- Page de contact.
- Mentions légales.
- Une page panier présentant l'ensemble des produits commandés.
- Page de login à l'espace privé.

### Directives

- **Page d'accueil (wireframe ci-dessous)**
  - Menu bannière haut
    - Accueil (retour à la page d'accueil).
    - Lasers (catégorie).
    - Casques (catégorie).
    - Contact.
    - Login.
  - Menu pied de page
    - Accueil.
    - Contact.
    - Mentions.
  - Contenus (liste des produits)
    - Titre du produit cliquable (lien menant vers la page du produit).
    - Miniature du produit.
    - Nom du produit.
    - Texte court de présentation du produit.
    - Mots clés du produit.
    - Nom de la catégorie dans laquelle le produit se trouve.

- Pied de page
  - Mentions légales.
  - Contact.
  - Accueil.



- **Page d'un produit**

Elle présentera un produit avec :

- Une image plus grande (image unique par produit)
- Un texte plus long
- Le prix
- Un formulaire pour commander ce produit :
  - Dans ce formulaire vous mettrez en place la possibilité de commander une quantité donnée de ce même produit (dans un select avec les quantités 1,2,3,4 fixées à l'avance par exemple).

**Wireframe produit voir ci-dessous :**

Accueil

Carers

Carques

contact

login



price

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

tags

quantity

Accueil

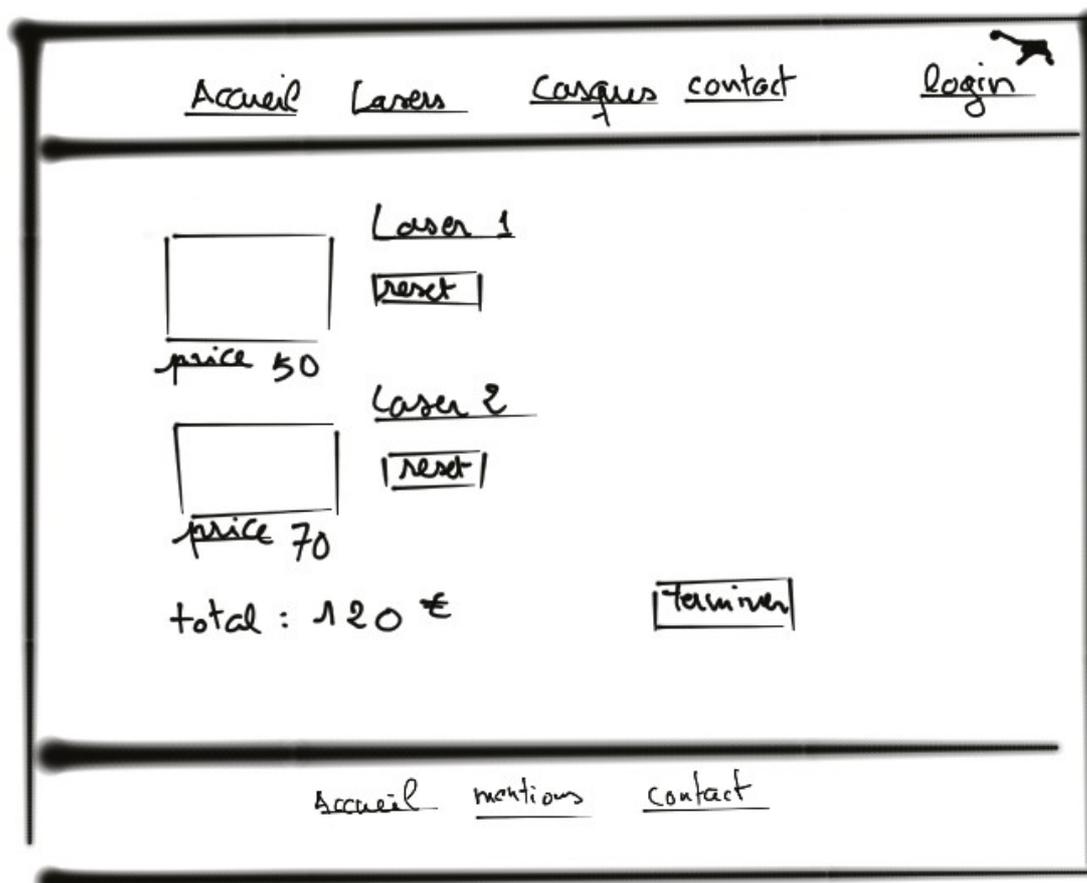
mentions

contact

- **Page panier**

Cette page présentera l'ensemble des produits du panier (commande en cours) avec la possibilité de retirer un produit commandé du panier. Et un bouton pour valider la commande.

**Wireframe panier voir ci-dessous :**



Lorsque l'internaute clique sur le bouton « terminé », il est redirigé vers la page de validation de la commande.

**Wirframe de validation de la commande ci-après :**

[Accueil](#)
[Laser](#)
[Casques](#)
[contact](#)
[login](#)

---

<input type="checkbox"/> price 50	<u>Laser 1</u> <input type="button" value="reset"/>	username <input type="text"/> email <input type="text"/>
<input type="checkbox"/> price 70	<u>Laser 2</u> <input type="button" value="reset"/>	
total : 120 €		<input type="button" value="command"/>

---

[Accueil](#)
[mentions](#)
[contact](#)

L'internaute devra s'authentifier (username et email). Lorsqu'il cliquera sur « commander », le couple username et email sera vérifié et si cette action est correcte, la commande sera enregistrée en base de données, le panier sera vidé et l'internaute sera redirigé vers la page d'accueil avec un message de succès.

- **Page mentions légales**

Page mentions légales classique non administrable.

- **Page de contact**

Ce formulaire présentera les champs suivants :

- Un email.
- Un champ texte.
- Un bouton pour envoyer le message.

The image shows a hand-drawn wireframe of a contact page. It is enclosed in a rectangular border. At the top, there is a horizontal line with five links: Accueil, Langues, Langues, contact, and login. Below this line, the main content area contains three elements: a label 'email' above a rectangular input field, a label 'contenu' above a larger rectangular text area, and a button labeled 'envoyer' to the right of the text area. At the bottom of the page, there is another horizontal line with three links: Accueil, mentions, and contact.

- **Page de connexion**

Vous mettez en place un système de connexion au dashboard du site sécurisé. Vous pouvez mettre en place un système « se souvenir de moi », cependant cette partie est facultative.

The diagram shows a login form with the following elements:

- A label "identifiant" above a rectangular input field.
- A label "Mot de passe" above a rectangular input field.
- A checkbox followed by the text "Se souvenir de moi".
- A rectangular button labeled "connexion".

Mot de passe oublié ?

<- retour sur le site

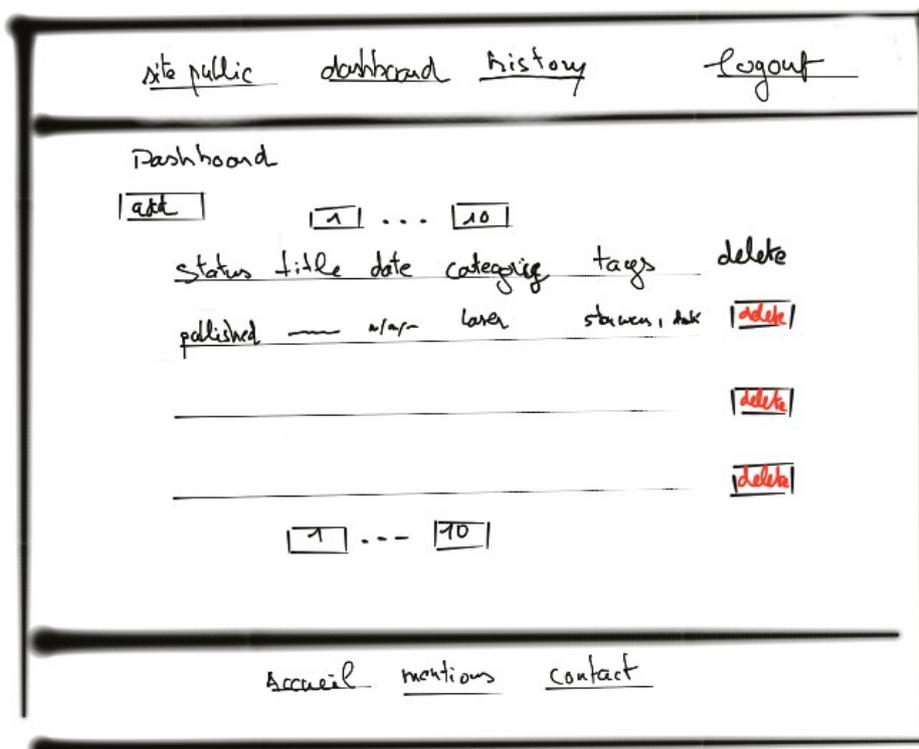
## Dashboard administration des produits

- **Page dashboard**
  - **Menu haut**
    - Un lien vers le site public « Retour au site public ».
    - Un lien vers le dashboard.
    - Un lien pour se déconnecter (logout).
    - Un lien vers la page de l'historique des commandes (présentation des commandes non administrable).

**Remarque : sur la partie publique du site vous pouvez faire un lien vers le dashboard pour revenir sur ce dernier.**

- **Colonne centrale**
  - Un bouton pour ajouter un produit
  - La liste des produits administrables
    - Un statut published ou unpublished
    - Un titre cliquable pour la mise à jour
    - La date de création du produit.
    - La catégorie dans laquelle se trouve ce produit.
    - La liste des tags en ligne.
    - Un bouton pour supprimer le produit (penser à mettre au message de confirmation).

**Wireframe dashboard enseignants ci-dessous :**



- **Fiche Create product**

Cette page permettra d'ajouter un produit en base de données, elle sera constituée de :

- Un titre.
- Un contenu.
- Une image.
- Des tags (prédéfinis max 4).

Elle permettra aussi :

- d'associer une catégorie à un produit (select).
- de définir le prix du produit.

Vous devez mettre en place un système de gestion d'erreurs dans le formulaire de création d'un produit. Ce système de gestion d'erreurs devra fonctionner côté client et serveur.

Gestion des erreurs dans le formulaire :

site public dashboard history logout

Dashboard

title

Content *titre obligatoire*

price *price float*

category

files *file format*

tags

star  space

black  galaxy

create

accueil mentions contact

- **Page Update**

La page de mise à jour d'un produit est identique à la page de création, vous devez cependant mettre le contenu modifiable des champs de la table product dans ce formulaire de mise à jour.

- **Page historique des commandes**

La page de présentation de l'historique des commandes présentera la liste des produits commandés par date décroissante de commande effectuée.

Détails de l'historique :

- date de la commande
- nom du client
- nom du produit commandé
- total
- statut de la commande (finalisé ou non finalisé)

Option facultative, voir toutes les commandes non payé(s) d'un client donné.