

Damjan Jovanovic

I.DA-P4A

2018

Table des matières

[1 Introduction 4](#_Toc514591602)

[2 Rappel du cahier des charges 4](#_Toc514591603)

[2.1 But 4](#_Toc514591604)

[2.2 Spécifications 4](#_Toc514591605)

[2.3 Restrictions 4](#_Toc514591606)

[2.4 Environnement 5](#_Toc514591607)

[2.5 Livrables 5](#_Toc514591608)

[3 Analyse fonctionnelle 5](#_Toc514591609)

[3.1 Fonctionnalités 5](#_Toc514591610)

[3.1.1 Inscription 5](#_Toc514591611)

[3.1.2 Connexion 5](#_Toc514591612)

[3.1.3 Validations utilisateurs 5](#_Toc514591613)

[3.1.4 Ajout de produits 6](#_Toc514591614)

[3.1.5 Modification de produits 6](#_Toc514591615)

[3.1.6 Suppression de produits 6](#_Toc514591616)

[3.1.7 Commande 6](#_Toc514591617)

[3.2 Interfaces 6](#_Toc514591618)

[3.2.1 Utilisateurs non-authentifiés 6](#_Toc514591619)

[3.2.2 Utilisateurs authentifiés 12](#_Toc514591620)

[3.2.3 Administrateur 15](#_Toc514591621)

[4 Analyse organique 21](#_Toc514591622)

[4.1 Diagramme de la base de données (MLD) 21](#_Toc514591623)

[4.2 Plan du site 21](#_Toc514591624)

[4.3 Requêtes principales 22](#_Toc514591625)

[4.4 Fonctions php 25](#_Toc514591626)

[4.4.1 checkNameLastName($var) 25](#_Toc514591627)

[4.4.2 LettersNumbers($var) 25](#_Toc514591628)

[4.4.3 OnlyNumbers($var) 25](#_Toc514591629)

[4.4.4 MinPwd($var) 25](#_Toc514591630)

[4.4.5 Menu() 25](#_Toc514591631)

[4.4.6 getSelectCategories() 25](#_Toc514591632)

[5 Tests 25](#_Toc514591633)

[6 Conclusion 26](#_Toc514591634)

# Introduction

Cette documentation a pour but de détailler le fonctionnement de l’application web développée dans le cadre du TPI (Travail Pratique Individuel). Elle est destinée aux experts évaluant le travail ainsi qu’aux personnes susceptible de reprendre le projet.

Il y a moins de 4 mois, j’ai moi-même créé ma marque de vêtements avec un ami, dans le cadre de ce travail, j’ai donc décidé de développer une application web permettant de vendre des habits.

# Rappel du cahier des charges

## But

Le but du projet est de pouvoir vendre des vêtements sportwears depuis l’application web.

## Spécifications

* La base de données contient les clients, les vêtements, les factures, les prix et le stock.
* L'administrateur gère les produits (CRUD complet depuis le site).
* Le système permet au visiteur authentifié la gestion d'un panier d'achat (sélection de plusieurs articles).
* Le visiteur authentifié possède un porte-monnaie virtuel alimenté par l’administrateur.
* Le visiteur ajoute des produits dans un panier.
* Le panier peut être commandé, le stock est recalculé, le montant est débité.
* Les changements de prix sont historisés.
* Le site gère l'inscription et l'authentification selon les normes du WEB selon la séquence demande inscription -> validation par l’administrateur -> confirmation par email (simulé) ->confirmation d’inscription. Un mécanisme anti robot est à mettre en place.
* Les mots de passe sont encryptés.
* La base de données des produits sera réalisée et pré-remplie de quelques articles avant le début du travail, de même que l’utilisateur admin sera déjà dans la table utilisateurs.

## Restrictions

* Aucune

## Environnement

Matériels nécessaires avant le développement du site :

* Ordinateur de type PC
* Système d’exploitation : Windows 10 Entreprise
* Outil de développement : JetBrains PhpStorm, WAMP

## Livrables

En 1 exemplaire :

* Carnet de bord
* Documentation technique
* Mode d’emploi
* Code source

Par messageries aux experts et au maître d’apprentissage :

* PDF de la documentation technique
* PDF du mode d’emploi
* PDF des sources imprimés du logiciel

Par messagerie au maître d’apprentissage :

* Les sources (projet logiciel) et la version exécutable

# Analyse fonctionnelle

## Fonctionnalités

### Inscription

Cette fonctionnalité permet à un client de créer un compte. Par la suite le compte sera validé ou non par l’administrateur permettant de se connecter sur le site et pouvoir ajouter des articles dans son panier.

### Connexion

Cette fonctionnalité permet à l’utilisateur de se connecter sur le site pour pouvoir ensuite ajouter des articles dans son panier.

### Validations utilisateurs

Cette fonctionnalité permet à l’administrateur de valider des utilisateurs qui se sont inscrits précédemment depuis le site.

### Ajout de produits

Cette fonctionnalité permet à l’administrateur d’ajouter des articles depuis l’administration du site web.

### Modification de produits

Cette fonctionnalité permet à l’administrateur de modifier un article.

### Suppression de produits

Cette fonctionnalité permet à l’administrateur de supprimer un article.

### Commande

Cette fonctionnalité permet à l’utilisateur de passer la commande des articles qui se trouvent dans son panier.

## Interfaces

### Utilisateurs non-authentifiés

#### Accueil

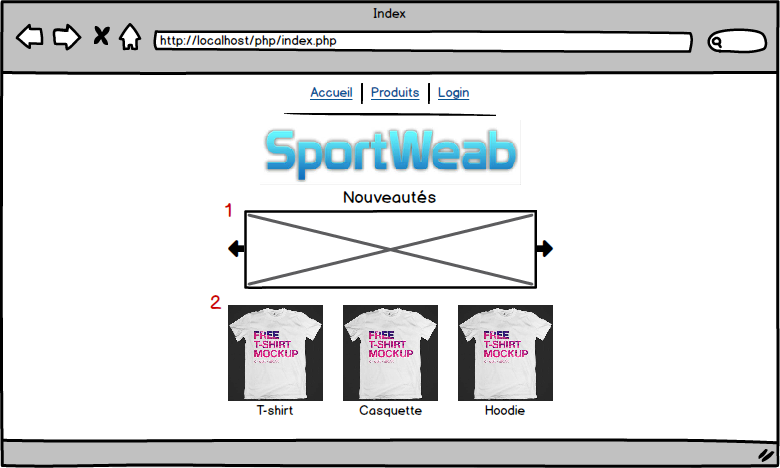


Figure 1 - page d’accueil du site web

1. Un slider qui affiche les tous les produits.
2. On peut apercevoir les trois catégories d’articles différents.

#### Connexion

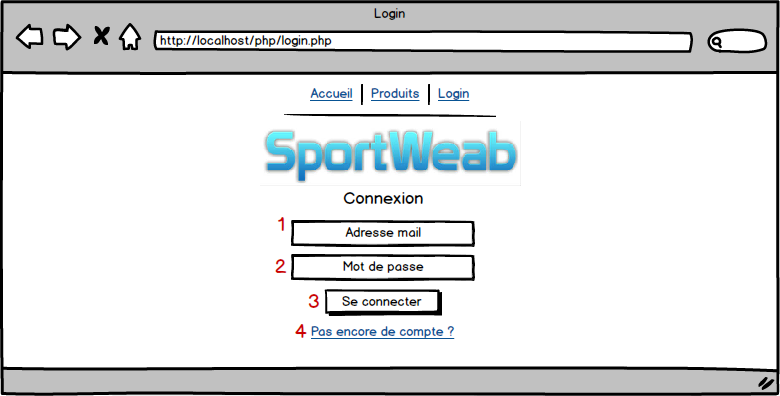


Figure 2 - Page de connexion

1. Champ pour l’email de connexion.
2. Champ pour le mot de passe lié au compte.
3. Bouton de connexion.
4. Lien qui redirige sur la page d’inscription. Voir **figure 3**.

#### Inscription

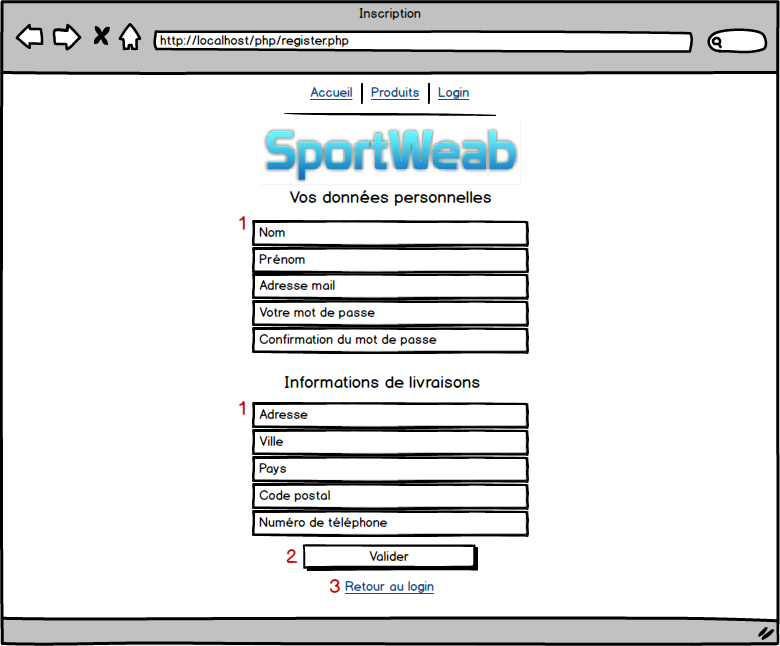


Figure 3 - Page d’inscription

1. Champs pour l’inscription.
2. Bouton d’inscription.
3. Lien qui redirige sur la page de connexion. Voir **figure 2**.

#### Produits



Figure 4 - PAGE DES PRODUITS

1. Lien permettant de trier les articles par prix ou par catégorie.
2. Les articles affichés triés par prix ou par catégorie. Si on clique sur un des articles, on est redirigé sur la page article qui affiche les informations du produit. Voir **figure 5**.

#### Page article

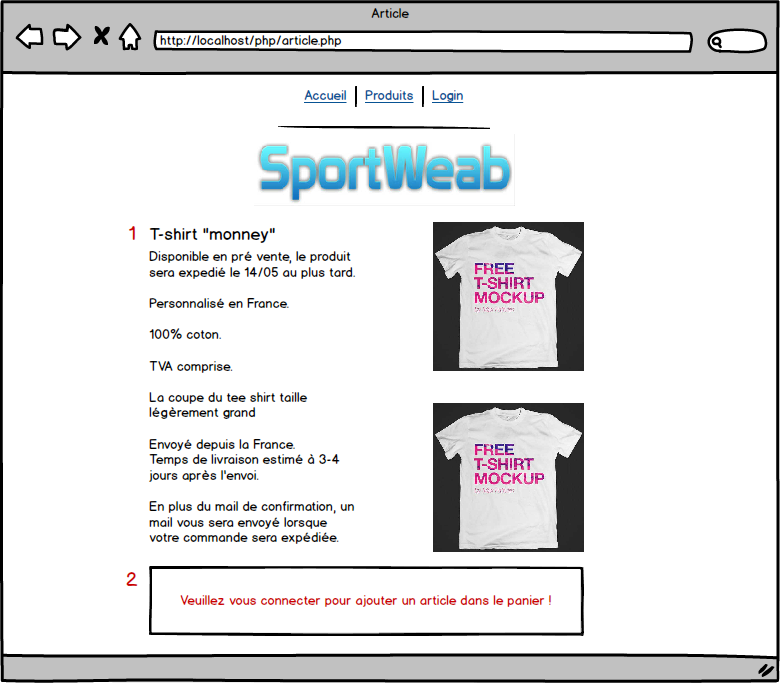


Figure 5 - Page article

1. Informations du produit.
2. Message indiquant qu’il faut être connecté pour pouvoir ajouter des articles dans son panier.

### Utilisateurs authentifiés

#### Article

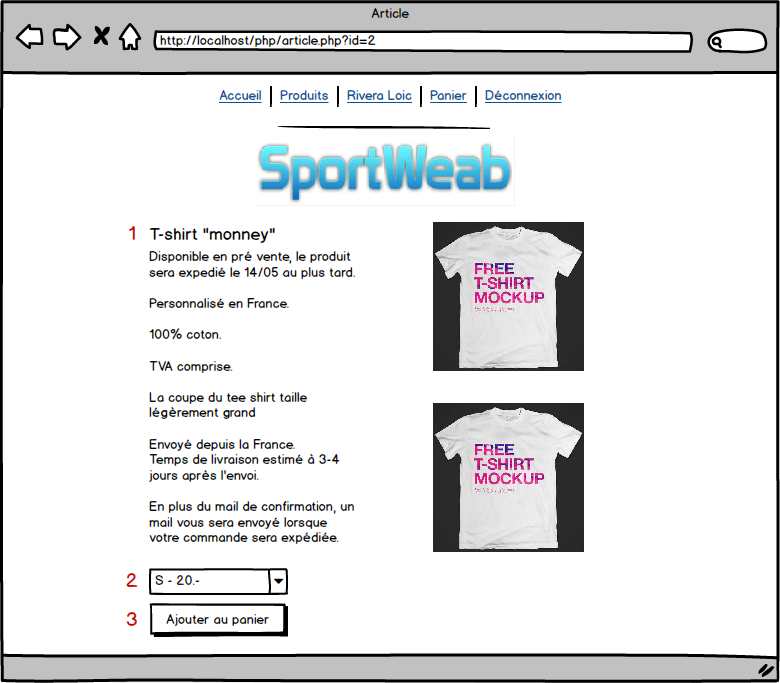


Figure 6 - Page de l’article

Quand on clique sur un article qui se trouve sur la page produit, la page article affiche les informations de l’article grâce à l’identifiant de l’article passé en paramètre.

1. Informations de l’article.
2. Choix de la taille de l’article.
3. Bouton qui permet d’ajouter l’article dans le panier.

#### Panier

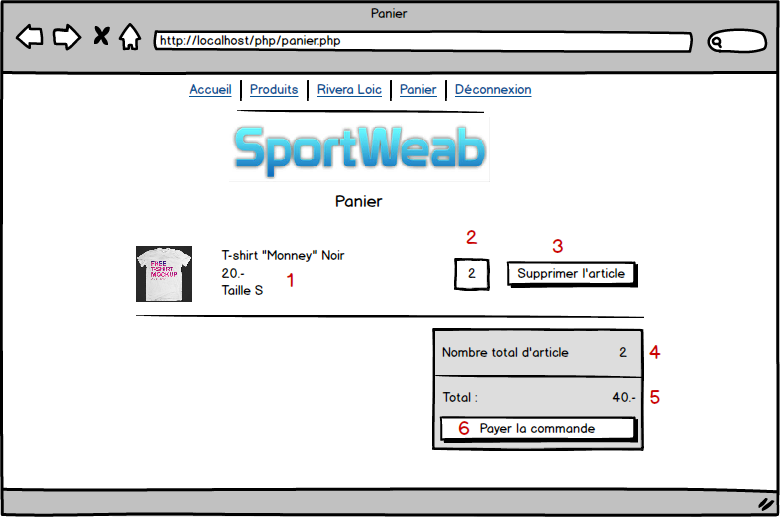


Figure 7 – panier

1. Information du produit.
2. Quantité du produit dans le panier.
3. Bouton permettant de supprimer l’article du panier.
4. Information concernant le nombre d’articles présent dans le panier.
5. Total du prix des articles présents dans le panier.
6. Bouton permettant de procéder à la page de paiement. Voir **figure 7**.

#### Page de paiement

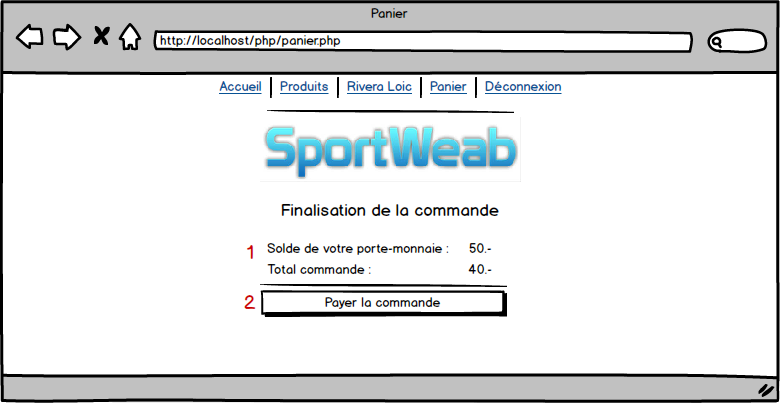
****

Figure 8 - Page de paiement

1. Information du solde du porte-monnaie de l’utilisateur et le prix total de la commande.
2. Bouton pour valider le paiement.

### Administrateur

#### Administration



Figure 9 - page d'administration

#### Validations d’utilisateurs

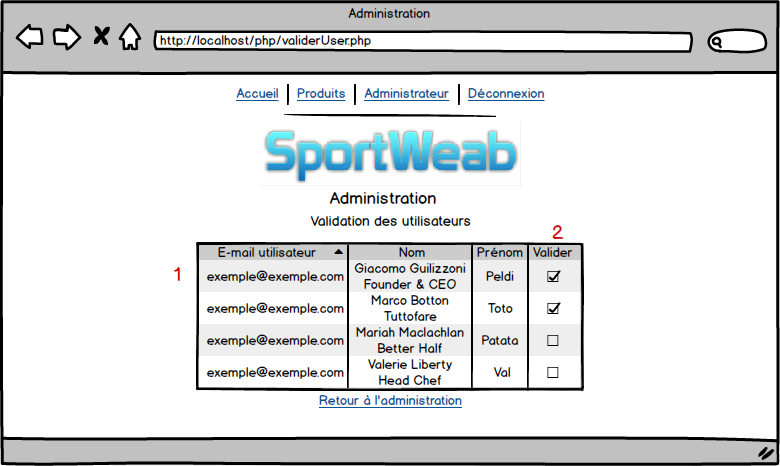


Figure 10 - page de validation des utilisateurs

1. Affichage des utilisateurs pas encore validé et leurs informations.
2. Bouton qui permet de valider les utilisateurs.

#### Page produits

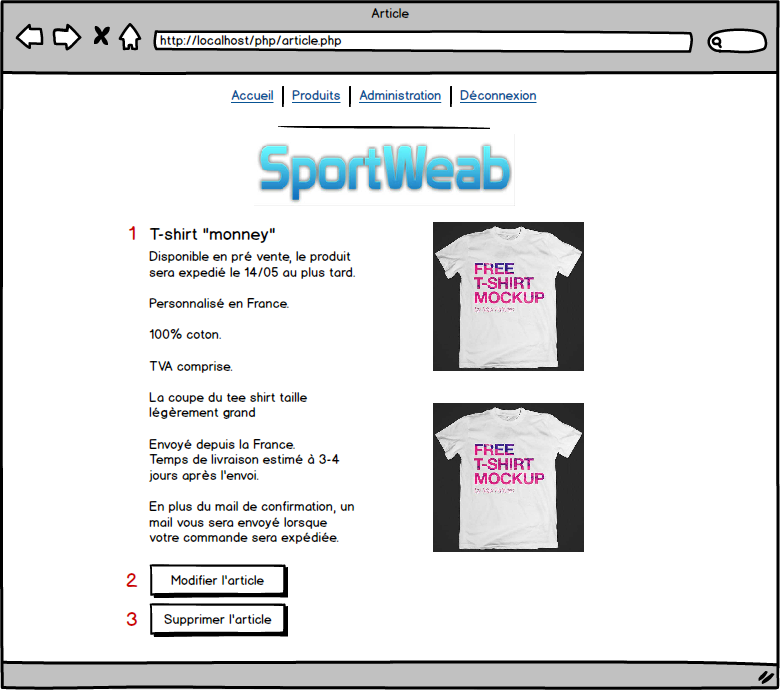


Figure 11 - Page produits administrateur

1. Description de l’article.
2. Bouton permettant de modifier un article et qui redirige sur la page « **modificationProduit.php** » voir figure 10.
3. Bouton permettant de supprimer un article.

#### Ajout de produits

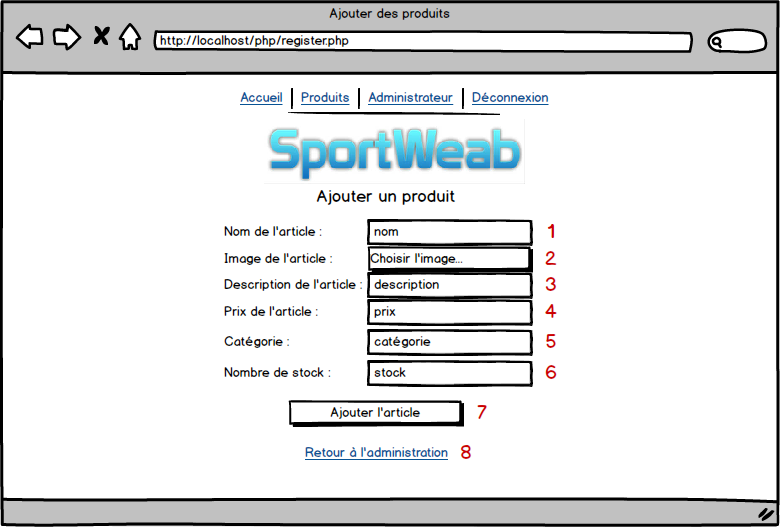


Figure 12 - PAGE D'AJOUT D'ARTICLE

1. Champ pour le nom de l’article.
2. Bouton permettant de sélectionner une image.
3. Champ pour la description de l’article.
4. Champ pour le prix de l’article.
5. Champ pour la catégorie de l’article.
6. Champ pour le nombre de stock.
7. Bouton permettant d’ajouter un produit.
8. Lien permettant de retourner sur la page de l’administration.

#### Modification d’un produit

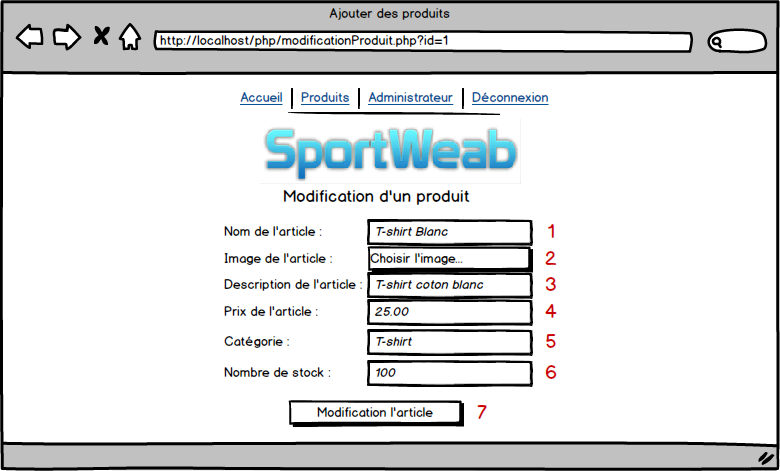


Figure 13 - Modification d'un produit

Les informations sont déjà pré-remplis par les informations de l’article grâce à l’identifiant passé en paramètre.

1. Champ pour le nom de l’article.
2. Bouton permettant de sélectionner une image.
3. Champ pour la description de l’article.
4. Champ pour le prix de l’article.
5. Champs pour la catégorie de l’article.
6. Champ pour le nombre de stock.
7. Bouton permettant de modifier le produit.

#### Suppression d’un produit

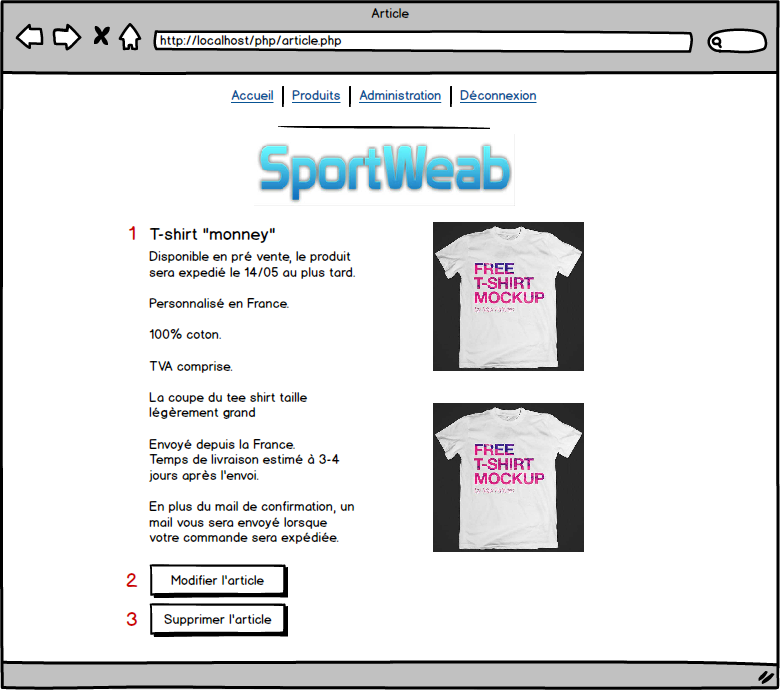
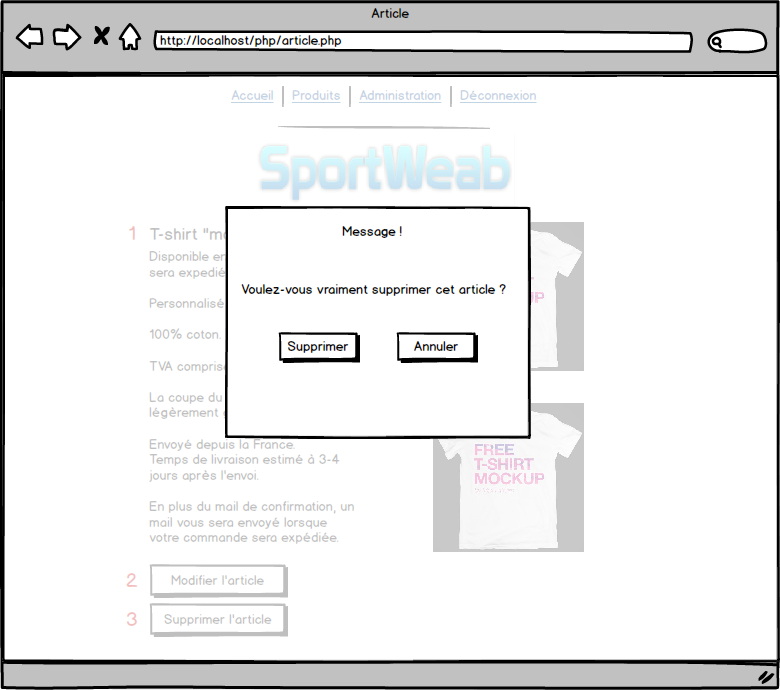


Figure 14 - page article administrateur



# Analyse organique

## Diagramme de la base de données (MLD)

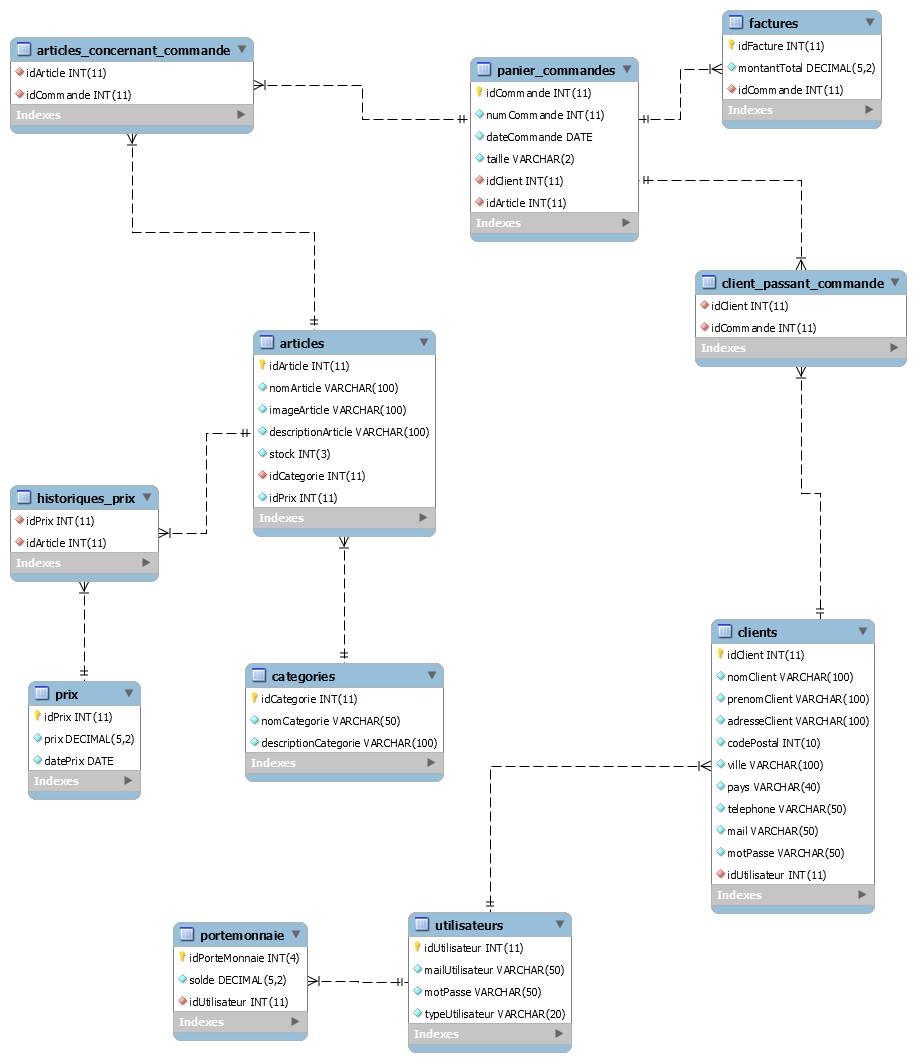


Figure 15 - Modele LOGIQUE de la base de donnees

## Plan du site

## Requêtes principales

|  |
| --- |
| **SELECT DISTINCT articles.idArticle, nomArticle, imageArticle, descriptionArticle, stock, nomCategorie, prixarticles.dateDebut FROM articles, categories, prixarticles WHERE articles.idArticle = prixarticles.idArticle AND articles.idCategorie = categories.idCategorie GROUP BY idArticle ORDER BY idArticle DESC;** |

Cette requête permet de chercher tous les articles.

|  |
| --- |
| **SELECT DISTINCT articles.idArticle, articles.nomArticle, articles.imageArticle, articles.descriptionArticle, articles.stock, categories.nomCategorie, prixarticles.idPrixArticle, prixarticles.prix FROM articles, categories, prixarticles WHERE articles.idCategorie = categories.idCategorie AND articles.idArticle = prixarticles.idArticle AND articles.idArticle = :idArticle ORDER BY prixarticles.dateDebut DESC LIMIT 1;** |

Cette requête permet de chercher les informations d’un article en fonction de l’identifiant de l’article.

|  |
| --- |
| **SELECT** *\** **FROM utilisateurs WHERE mailUtilisateur = :nom AND motPasse = :mdp;** |

Cette requête permet de chercher un utilisateur avec son et son mot de passe passés en paramètres.

|  |
| --- |
| **INSERT INTO `clients` (`nom`, `prenom`, `adresse`, `codePostal`, `ville`, `telephone`, `mail`, `motPasse`) VALUES (:nom, :prenom, :adresse, :codePostal, :ville, :telephone, :mail, :motPasse)** |

Cette requête permet de créer un nouveau client.

|  |
| --- |
| **INSERT INTO `utilisateurs` (`mailUtilisateur`, `motPasse`) VALUES (:mail, :motPasse)** |

Cette requête permet de créer un nouvel utilisateur.

|  |
| --- |
| **INSERT INTO `portemonnaie` (`idUtilisateur`) VALUES (:idUtilisateur)** |

Cette requête permet de créer un nouveau porte-monnaie pour un utilisateur.

|  |
| --- |
| **SELECT** *\** **FROM clients, utilisateurs WHERE clients.mail = :mail AND utilisateurs.mailUtilisateur = :mail** |

Cette requête permet de chercher les données d’un utilisateur.

|  |
| --- |
| **SELECT** *\** **FROM utilisateurs WHERE typeUtilisateur = 'en attente';** |

Cette requête permet de chercher les utilisateurs qui sont en attente de validations.

|  |
| --- |
| **UPDATE utilisateurs SET typeUtilisateur = 'Utilisateur' WHERE idUtilisateur = :id ;** |

Cette requête permet de valider un utilisateur.

|  |
| --- |
| **INSERT INTO `prixarticles` (`prix`, `dateDebut`, `dateFin`, `idArticle`) VALUES (:prix, :dateDebut, :dateFin, :idArticle)** |

Cette requête permet d’ajouter un nouveau prix pour un article.

|  |
| --- |
| **UPDATE prixarticles SET idArticle = :idArticle WHERE idPrixArticle = :idPrixArticle;** |

Cette requête permet de modifier le prix d’un article

|  |
| --- |
| **INSERT INTO `articles` (`nomArticle`, `imageArticle`, `descriptionArticle`, `stock`, `idCategorie`, `idPrixArticle`) VALUES (:nomArticle, :imageArticle, :descriptionArticle, :stock, :idCategorie, :idPrixArticle)** |

Cette requête permet d’ajouter un nouvel article.

|  |
| --- |
| **UPDATE articles SET nomArticle = :nomArticle, imageArticle = :imageArticle, descriptionArticle = :descriptionArticle, stock = :stock, idCategorie = :idCategorie, idPrixArticle = :idPrixArticle WHERE idArticle = :idArticle ;** |

Cette requête permet de modifier les données d’un article.

|  |
| --- |
| **DELETE FROM articles WHERE articles.idArticle = :idArticle;** |

Cette requête permet de supprimer un article en fonction de l’identifiant passé en paramètre.

|  |
| --- |
| **DELETE FROM panier\_commandes WHERE panier\_commandes.idArticle = :idArticle AND panier\_commandes.idClient = :idClient AND panier\_commandes.idCommande = :idCommande;** |

Cette requête permet de supprimer un article de son panier.

|  |
| --- |
| **SELECT distinct panier\_commandes.idCommande, articles.idArticle, articles.nomArticle, articles.imageArticle , prixarticles.prix, panier\_commandes.taille, prixarticles.dateDebut, prixarticles.dateFin, panier\_commandes.dateCommande,panier\_commandes.idClient FROM panier\_commandes, articles, client\_passant\_commande, clients, prixarticles, articles\_concernant\_commande WHERE panier\_commandes.idCommande = articles\_concernant\_commande.idCommande AND articles\_concernant\_commande.idArticle = articles.idArticle AND panier\_commandes.idCommande = client\_passant\_commande.idCommande AND client\_passant\_commande.idClient = clients.idClient AND articles.idArticle = prixarticles.idArticle AND prixarticles.dateDebut <= panier\_commandes.dateCommande  AND (prixarticles.dateFin >= panier\_commandes.dateCommande or prixarticles.dateFin IS NULL) AND client\_passant\_commande.idClient = :idClient ORDER BY prixarticles.dateDebut DESC;** |

Cette requête permet d’afficher les articles dans le panier d’un utilisateur.

## Fonctions php

### checkNameLastName($var)

Cette fonction permet de vérifier que la variable passée en paramètre contient que des lettres et seulement le tiret pour les prénoms et nom composé.

### LettersNumbers($var)

Cette fonction permet de vérifier que la variable passée en paramètre contient que des lettres.

### OnlyNumbers($var)

Cette fonction permet de vérifier que la variable passée en paramètre contient que des chiffres.

### MinPwd($var)

Cette fonction permet de vérifier que la variable contient plus que 4 charactères-

### Menu()

Cette fonction permet au menu d’être dynamique. On affiche l’onglet « Accueil », « Produits » et « Connexion » quand l’utilisateur n’est pas connecté.

Si l’utilisateur est connecté et que l’utilisateur est un administrateur, l’onglet « Accueil », « Produits », « Administration » et « Déconnexion » est affiché.

Si l’utilisateur est connecté et qu’il n’est pas administrateur, l’onglet « Accueil », « Produits », « Panier » et « Déconnexion » est affiché.

### getSelectCategories()

Cette fonction permet d’afficher les catégories dans un select.

# Tests

|  |  |  |  |
| --- | --- | --- | --- |
| **A tester** | **Comment ?** | **Résultat** | **Bug résolu ?** |
| Login avec un compte inexistant | C |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Conclusion