



Cours : Web Dynamique 2 (WD2)
Professeur : M. J.P. Delcroix

Compte rendu de TP n°2

Première approche de la base de données en PHP

Pierre Jouet Club

Magasin de Vente par Correspondance

Auteur : Pierre Galerneau
Année Universitaire : 2008 / 2009

Résumé

L'objectif de ce second TP faisant l'objet d'un compte rendu en Web Dynamique 2 était de réaliser un site de vente par correspondance permettant notamment de gérer les stocks des produits en vente.

J'ai adapté ce sujet afin de pouvoir réaliser un site de VPC le plus complet possible. C'est-à-dire que j'ai pris modèle sur un site réel sur internet et que j'ai recréé ses fonctions principales sur mon propre site. De plus, je me suis fixé l'objectif supplémentaire de créer un script d'upload de fichier afin de permettre à des administrateurs de télécharger à partir du site, des images sur le serveur. Ceci ne faisait pas partie du sujet cependant, j'ai trouvé que cela pouvait être intéressant et rendre le site plus réaliste.

Table des matières

Introduction	1
1 Présentation du projet	2
2 La base de données	2
2.1 Esquisse	2
2.2 La table vpc_Users	3
2.3 La table vpc_produits7	4
2.4 La table vpc_panier	6
2.5 La base de données en général	7
3 Mes choix de programmation	8
3.1 Choix d'architecture	8
3.1.1 Architecture des pages	8
3.1.2 Le menu	10
3.1.3 Les différentes pages du site	10
3.2 Les différents types d'utilisateurs	27
3.2.1 Mise en place des sessions de la VPC	28
3.2.2 Intérêt de l'objet session	28
3.3 Quelle présentation adopter ?	29
4 Explication du code	29
4.1 Arbres programmatiques et explication hors programmation	29
4.1.1 Algorithme de connexion au site et affichage des produits en vente	29
4.1.2 Algorithme d'ajout d'un enregistrement	32
4.1.3 Upload d'un fichier sur le serveur	33
4.1.4 Algorithme de commande d'un produit	34
4.1.5 Modification de données	35
4.2 Détail des fonctionnalités de chaque page	36
4.2.1 L'arrivée sur le site : la page accueil.php	36
4.2.2 Créer un compte : créer_compte.php / créer.php	38
4.2.3 Connexion d'un utilisateur : la page connect.php	39
4.2.4 La page de commande : commande.php	40
4.2.5 Affichage du panier : la page panier.php	40
4.2.6 La page de retour d'articles : retirer.php	41
4.2.7 Payer sa commande, la page payer.php / paie.php	42
4.2.8 Le compte utilisateur : la page mon_compte.php	42
4.2.9 La page admin.php	44
4.2.10 Ajout_produit.php	45
4.2.11 La vérification des utilisateurs	46
4.3 Les erreurs et problèmes rencontrés	46
4.3.1 Prise en main de MySQL	46
4.3.2 Upload d'un fichier sur un serveur	47
Conclusion	48
Annexes	49
Annexe 1 : page d'accueil	50

Annexe 2 : formulaire de création de compte	53
Annexe 3 : création d'un utilisateur : page créer.php.....	54
Annexe 4 : page de connexion : connect.php.....	56
Annexe 5 : commander un produit, la page commande.php.....	58
Annexe 6 : Affichage du panier, la page panier.php.....	60
Annexe 7 : retirer un produit du panier, page retirer.php.....	63
Annexe 8 : règlement de la commande, pages payer.php et paie.php.....	65
Annexe 9 : la page mon_compte.php	70
Annexe 10 : Panneau d'administration, la page admin.php.....	74
Annexe 11 : ajouter un produit, le fichier ajout_produit.php.....	81
Annexe 12 : upload d'un fichier sur le serveur, le fichier upload.php.....	83

Introduction

Dans ce second rapport de Web Dynamique 2; je vais présenter les différents aspects de la réalisation d'un site de vente par correspondance en ligne complet. Ce rapport va me permettre de mettre sur papier toute la réflexion mise en œuvre pour réaliser ce site, que ce soit pour les pages simples ou les plus compliquées.

Dans ce second rapport, l'un des grands objectifs était d'utiliser le PHP couplé aux bases de données MySQL. En effet, jusqu'à maintenant, nous n'avions utilisé que des fichiers séquentiels pour simuler les bases de données. Ce rapport va donc également me permettre d'expliquer les différents aspects de la programmation PHP utilisant ces bases de données et surtout les différences que l'on peut trouver entre des vraies bases de données et une simulation de ces bases !

Comme ce projet ne s'est pas réalisé en une seule étape, il est intéressant de se poser la question suivante pour comprendre comment j'ai pu en arriver au résultat final :

Quelles étapes et quels choix ont été nécessaires à la réalisation en bonne et due forme d'un magasin de vente par correspondance en PHP utilisant bases de données MySQL comme sauvegarde de données

Afin de mettre plus en œuvre mes compétences et surtout de les améliorer, je me suis fixé un objectif supplémentaire qui était de réaliser ce site en y incluant un script permettant l'upload de fichiers sur le serveur hébergeant le site

1 Présentation du projet

Ce projet avait pour objectif de nous faire développer un magasin de vente par correspondance en PHP. Celui-ci devait être développé en utilisant les bases de données MySQL et non plus les fichiers séquentiels comme au cours des TP précédents (notamment la DVD Tech'). Nous allons donc pour la première fois étudier les bases de données en cours de Web Dynamique.

L'objectif premier de ce TP est donc de se familiariser avec l'utilisation des bases de données MySQL en PHP.

Le site réalisé par cette méthode doit permettre, comme je l'ai précisé en introduction, à différents types d'utilisateurs de se connecter sur le site et d'avoir accès à différents traitements selon la catégorie à laquelle ils appartiennent. Il y aura donc deux grandes catégories d'utilisateurs sur mon site :

- Les administrateurs (qui ont l'autorisation de modifier les tables, les supprimer,...),
- Les « simples » utilisateurs (qui ne peuvent que acheter des produits, les rendre et qui n'ont pour seule action sur les bases de données, que l'autorisation de modifier leur propre compte.

Maintenant que les bases du projet sont posées et que l'on comprend qui pourra faire quoi sur notre site, il est nécessaire d'expliquer comment sont gérées les données et comment elles ont organisées entre elles au sein de nos tables.

2 La base de données

Lors de ce second projet, nous avons eu à utiliser pour la première fois les bases de données en temps que telles. En effet, pour la première fois dans le cours de Web Dynamique (que ce soit au premier ou au second semestre), nous avons jusqu'à maintenant utilisé exclusivement les fichiers séquentiels pour simuler des bases de données. Pour la première fois donc, nous allons utiliser réellement les bases de données MySQL et les requêtes SQL pour faire tous les traitements utiles à la réalisation du site.

2.1 Esquisse

Dans ce projet, l'objectif était de réaliser un magasin de vente par correspondance en ligne. Pour ce faire, il nous a fallu mettre en place un certain nombre de tables indispensables au bon fonctionnement du site. En effet, il nous fallait pouvoir enregistrer des informations sur :

- Les produits,
- Les utilisateurs,
- Les achats en attente de chaque utilisateur (panier).

Dans le sujet, rien n'est précisé quand à l'architecture des données, c'est donc à nous de la mettre en place.

De fait, il est apparu après analyse que nous avons besoin de 3 tables pour mener à bien notre projet. Je vais les détailler une par une ci-dessous.

Remarque : pour réaliser ce projet - et ceux qui vont suivre - dans lesquels nous devons utiliser les bases de données MySQL, nos tables seront toujours précédées d'un préfixe correspondant au sujet de TP. En effet, chez Free, nous ne disposons que d'une seule base de données. Il nous est donc indispensable de pouvoir discerner au premier coup d'œil les bases qui se rapportent à l'un ou l'autre des projets. Cette précision nous permettant également de les différencier sur le serveur Free.

2.2 La table vpc_Users

La table vpc_Users est destinée à contenir toutes les données concernant les utilisateurs de notre site. En effet, ces données doivent pouvoir être enregistrées quelque part afin d'être retrouvées et consultées régulièrement.

Devant réaliser un magasin de vente par correspondance, j'ai cherché à rendre le contenu de la base vpc_Users le plus réaliste possible par rapport aux informations que l'on peut nous demander à l'inscription sur un site de Vente par Correspondance (VPC) réel, c'est pourquoi j'ai décidé de mettre en place le schéma suivant :

vpc_users
<u>idUser</u>
Nom
Prenom
Adresse
Code Postal
Ville
Login
Mot de Passe

Je vais maintenant décrire le type de chaque propriété de cette table :

- id User : entier séquentiel non nul,
- nom : caractère variable (chaîne de caractères),
- prénom : caractère variable (chaîne de caractères),
- adresse : caractère variable (chaîne de caractères),
- Code postal : entier compris entre 0 et 99999,
- Ville : caractère variable (chaîne de caractères),
- Login : caractère variable (chaîne de caractères),
- Mot de passe : caractère variable (chaîne de caractères).

Comme nous utilisons maintenant le SQL, il est intéressant de montrer les scripts de créations de tables générés pour chaque table du modèle. En effet, le schéma est destiné à montrer comment la table va se présenter graphiquement, cependant ce n'est qu'une représentation

abstraite. Il est donc important de donner également le script SQL de création puisque c'est lui qui va permettre réellement de créer et d'accéder aux tables par la suite.

Script de création de la base vpc_users :

Code

```
-- =====  
-- Table : VPC_USERS  
-- =====  
create table VPC_USERS  
(  
  IDUSER          DECIMAL(6)      not null,  
  NOM             VARCHAR(50)     not null,  
  PRENOM         VARCHAR(50)     not null,  
  ADRESSE        VARCHAR(50)     not null,  
  CODE_POSTAL    INTEGER         not null,  
  VILLE          VARCHAR(50)     not null,  
  LOGIN          VARCHAR(50)     not null,  
  MOT_DE_PASSE   VARCHAR(32)     not null  
);  
  
-- =====  
-- Index : VPC_USERS_PK  
-- =====  
create unique index VPC_USERS_PK on VPC_USERS (IDUSER asc);
```

Ce script n'est pas un script de création MySQL mais un script SQL classique. Je pense que ça ne change pas grand-chose de présenter celui-ci plutôt qu'un autre puisque le principe est d'expliquer quel type de script est utilisé pour créer une table en SQL et non de faire un cours de MySQL...

2.3 La table vpc_produits

Maintenant que les utilisateurs peuvent être enregistrés sur notre site, il est nécessaire de créer une table destinée à contenir les produits disponibles sur le site. En effet, tout comme les utilisateurs, il faut pouvoir accéder aux produits pour :

- Les afficher,
- Les modifier,
- Les acheter,
- ...

Pour cela, j'ai mis en place une table SQL dont le schéma est :

vpc_produits
<u>idProduit</u>
Nom
nb Dispo
Image
Prix

Pour décrire un produit dans un magasin de vente par correspondance, il est essentiel de pouvoir stocker certaines informations le concernant, notamment :

- Son libellé,
- Sa quantité en stock;
- L'image à afficher pour le représenter,
- Son prix.

D'où les propriétés présentes dans ma table vpc_produits. Je vais maintenant en expliquer les types :

- id Produit : entier séquentiel non nul,
- Nom : caractère variable (chaîne de caractères),
- NbDispo : entier compris entre 0 et plus l'infini,
- Image : caractère variable (chaîne de caractère contenant le lien vers cette image sur le serveur),
- Prix : entier compris entre 0.01 et plus l'infini.

Une fois ce schéma établi, il ne reste plus qu'à générer le code SQL correspondant pour créer la table :

Code

```
-- =====  
-- Table : VPC_PRODUIITS  
-- =====  
create table VPC_PRODUIITS  
(  
  IDPRODUIT      DECIMAL(6)      not null,  
  NOM            VARCHAR(50)      ,  
  NB_DISPO      INTEGER          not null,  
  IMAGE         VARCHAR(100)     not null,  
  PRIX          INTEGER          not null  
);  
  
-- =====  
-- Index : VPC_PRODUIITS_PK  
-- =====  
create unique index VPC_PRODUIITS_PK on VPC_PRODUIITS (IDPRODUIT asc);
```

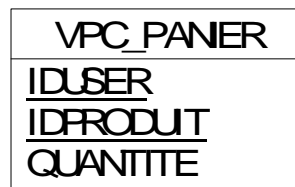
Maintenant que nous avons détaillé la création des deux tables principales de notre site, il est nécessaire d'expliquer pourquoi nous avons besoin d'en utiliser une troisième : la table vpc_tapnier.

Lorsque un utilisateur va commander un certain nombre d'un produit (nombre inférieur ou égal à la quantité en stock), il est nécessaire de pouvoir stocker dans la base de données le fait que l'utilisateur X a commandé le produit Y en quantité Z. C'est pour cela que la table vpc_panier est indispensable; car ces données ne peuvent ni être sauvegardées dans la table vpc_users ni dans la table vpc_produits.

2.4 La table vpc_panier

Cette table vpc_panier est donc en relation avec les deux tables présentées précédemment ce qui veut dire qu'on va retrouver dans cette table les clés des deux autres tables. En effet, ce sont les seules informations permettant de différencier un enregistrement de l'un ou l'autre des tables donc, pour différencier un achat d'un autre, il faut non seulement spécifier de quel produit il s'agit mais également de quel utilisateur. En effet, un utilisateur peut avoir commandé plusieurs produits donc son identifiant ne suffit pas à identifier une seule ligne de commande, cependant, un produit peut également être commandé par plusieurs clients donc l'identifiant du produit n'est pas non plus suffisant. Il faut donc utiliser les deux identifiants. En effet, si l'on précise que l'on cherche la commande dont le client est untel et que cette commande concerne le produit Y, alors le résultat sera unique.

Voici donc le schéma de la table vpc_panier :



Les propriétés IDUSER et IDPRODUIT sont de même type que dans les deux tables présentées ci-dessus, il ne reste donc plus qu'à présenter la propriété « quantité ». Celle-ci est destinée à enregistrer le nombre de produits commandés par utilisateur, c'est donc un entier positif supérieur à 0 et inférieur ou égal à la quantité totale disponible dans la table vpc_produits.

Comme cette table vpc_panier dépend des deux autres, son script de création sera un peu différent. En effet, sa clé primaire (destinée à identifier un enregistrement de manière unique) ne contient plus une seule valeur mais bien deux.

Voici donc le code SQL correspondant :

Code

```
-- =====  
-- Table : VPC_PANIER  
-- =====  
create table VPC_PANIER  
(  
  IDUSER          DECIMAL(6)      not null,  
  IDPRODUIT       DECIMAL(6)      not null,  
  QUANTITE        INTEGER          not null  
);  
  
-- =====  
-- Index : VPC_PANIER_PK  
-- =====  
create unique index VPC_PANIER_PK on VPC_PANIER (IDUSER asc, IDPRODUIT asc);
```

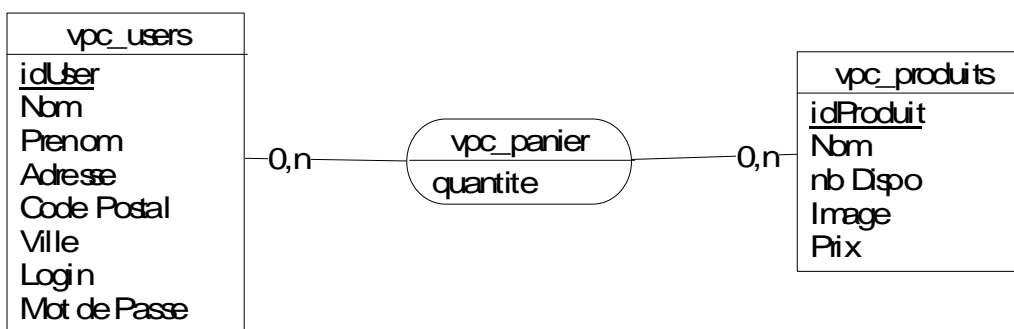
2.5 La base de données en général

Dans ce projet, la base de données se compose donc de trois tables dont deux sont indépendantes et une est en relation avec les deux précédentes. Je vais présenter ici les modèles qui m'ont permis d'arriver à cette modélisation :

- Le Modèle Conceptuel des Données (MCD),
- Le Modèle Physique des Données (MPD).

Ces deux modèles permettent de définir les données conceptuellement puis en terme de tables SQL, ce qui permet une analyse du problème propre et claire.

MCD de la VPC :



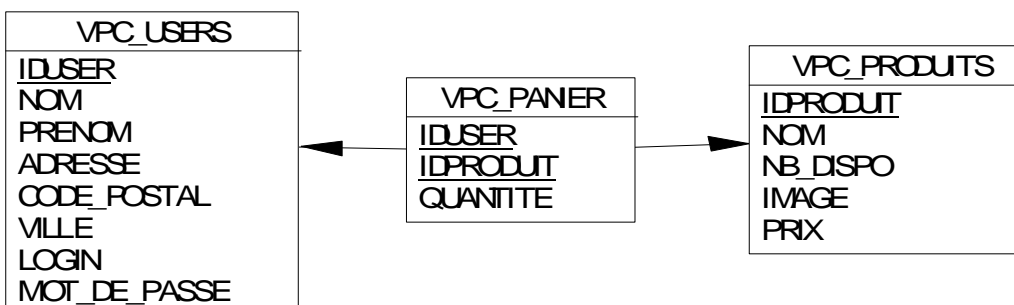
Ce modèle permet de retranscrire le besoin en termes de données. On voit dessus que le site nécessite deux tables totalement indépendantes qui entrent tout de même en relation lors de la création d'un panier, c'est-à-dire lorsqu'un utilisateur commande un produit.

Les cardinalités (0,n sur le schéma) correspondent au fait que :

- Un utilisateur peut commander 0 ou n produits,
- Un produit peut être commandé 0 ou n fois.

Des cardinalités de ce type veulent dire qu'une nouvelle base doit être créée pour contenir les informations dépendant des deux autres.

MPD de la VPC :



Le MPD correspondant, quant à lui, à la représentation réelle des données sous formes de tables SQL. Chaque tableau du modèle correspond à une table SQL. Les flèches signifient

quant à elles que la table `vpc_panier` reçoit des informations venant des deux tables `vpc_users` et `vpc_produits`.

Maintenant que la structure de la base de données nécessaire au bon fonctionnement du site et celle des tables la composant a été expliquée, je vais décrire les choix que j'ai eu à faire lors de la programmation de ce projet.

3 Mes choix de programmation

Nous n'avons pas eu le choix du langage dans lequel effectuer notre projet. Cependant, de nombreux choix se sont présentés à nous au cours de la réalisation de la VPC. Par exemple ; il a fallu faire des choix quand à :

- L'organisation des données,
- Le moyen de connexion au site,
- Etc.

Enfin, il a fallu choisir quelle interface offrir à l'utilisateur. En effet, l'utilisateur doit se sentir, sur notre site de Vente par Correspondance, comme sur un site professionnel, l'affichage doit être le plus réaliste possible et les fonctions proposées également. L'utilisateur doit avoir le choix de modifier son compte, de faire afficher ce qu'il a emprunté, etc., c'est à dire que toutes les informations importantes doivent être accessibles en un minimum de clics afin que l'utilisateur puisse trouver tout ce qu'il recherche sans perdre de temps. Tous ces choix sont orientés de manière à ce que l'utilisateur ait le plus de plaisir possible à naviguer sur notre site.

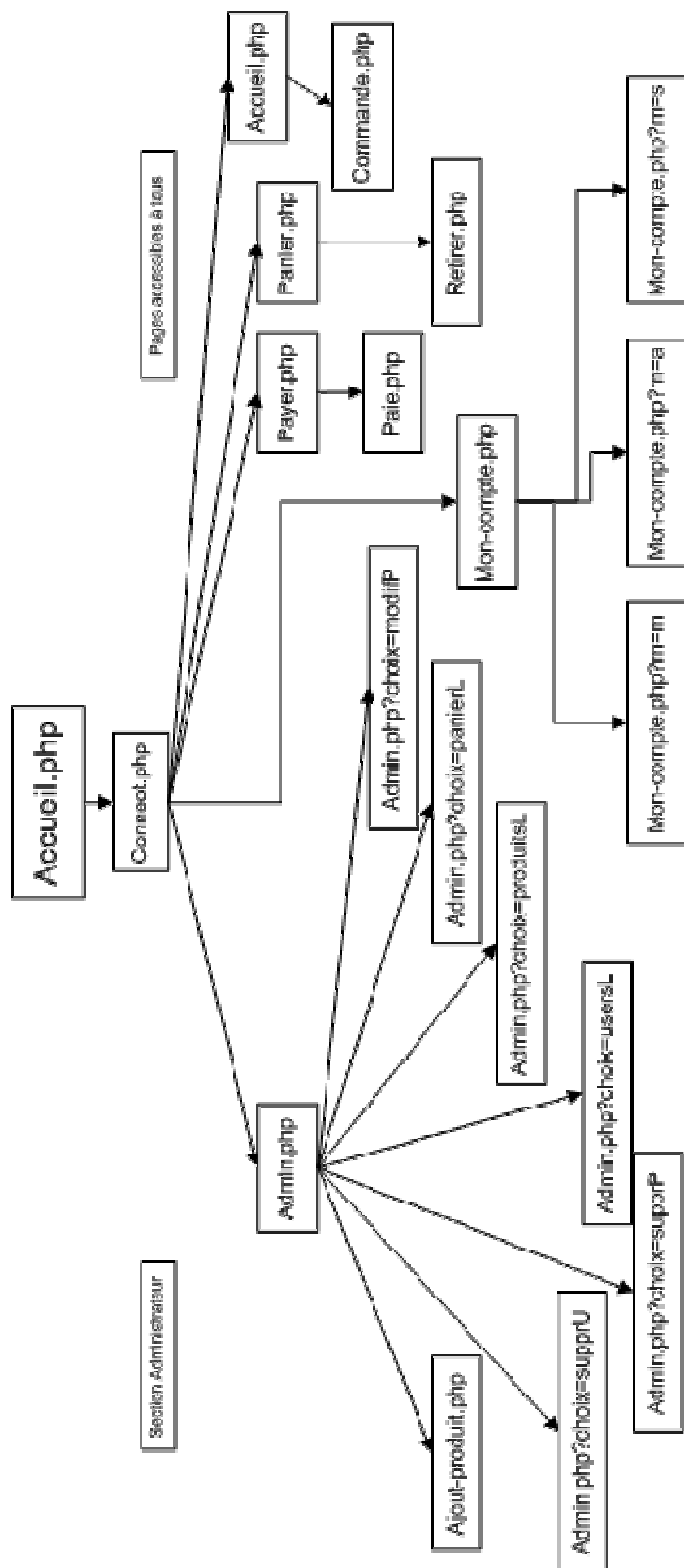
3.1 Choix d'architecture

Pour la réalisation de ce site de vente par correspondance, j'ai décidé de reprendre une architecture classique par rapport au projet précédent (DVD Tech). En effet, réaliser ce site en une seule page PHP contenant des tests et tous les codes de toutes les pages aurait été trop compliqué. En effet, la vente par correspondance est une application plus importante que la DVD Tech et surtout, nous ne l'avons pas réalisé au cours du premier semestre, c'est donc moins évident à réaliser puisque tous les concepts et toute la logique sont à comprendre et à réaliser au fur et à mesure.

Puisque, encore une fois, je dois donner la possibilité à des utilisateurs de différents types de se connecter (administrateurs ou simples utilisateurs), il est nécessaire de mettre en place une certaine architecture entre les pages, architecture que je vais présenter maintenant :

3.1.1 Architecture des pages

Toutes les pages de mon site s'enchaînent à partir du moment où un utilisateur d'un type ou de l'autre s'est connecté sur le site. Selon le type d'utilisateur connecté ; les pages accessibles ne sont pas les mêmes. Je vais donc expliquer cet enchaînement par un diagramme montrant quelles pages sont accessibles à qui au sein de mon site :



Même si l'arborescence des pages est présentée ci-dessus, il est important que j'explique certains autres choix que j'ai faits concernant l'architecture du site, notamment au niveau du menu.

3.1.2 *Le menu*

J'ai décidé de faire un « beau » menu pour ce site, c'est un dire un menu contenant des images a la place des simples boutons HTML utilisés jusque maintenant. Pour cela et pour éviter surtout d'avoir à modifier le menu sur chaque page de mon site à chaque petite modification de son contenu, j'ai décidé de le créer dans une page à part et d'inclure cette page dans chaque page du site nécessitant le menu (toutes les pages à par la page d'accueil quand les utilisateurs ne sont pas connectés) grâce à la fonction PHP include dont l'utilisation est :

Code

```
include(« fichier.php »);
```

Cette ligne de code se retrouve donc dans chacun de mes fichiers, ce qui me permet de n'avoir qu'à modifier un seul fichier quand je veux modifier mon menu. Cela évite que je doive chercher dans tous les fichiers du site l'endroit où j'inclus le menu afin de le modifier. En plus d'un gain de temps, c'est un gain en clarté mais également, cela évite que je n'oublie certaines pages ou bien que je fasse des erreurs dans l'une ou l'autre de ces pages. Par cette méthode, le menu est exactement le même dans toutes les pages du site.

3.1.3 *Les différentes pages du site*

J'ai déjà présenté en gros les différentes pages accessibles dans la section précédente, cependant, je pense qu'il est bon que je fasse une section dans laquelle je présente plus en détail chaque page de manière à présenter les choix graphiques que j'ai faits et les possibilités qu'un utilisateur a sur chaque page du site, qu'il soit administrateur ou non. Cette section n'est pas destinée à présenter la manière dont les pages sont générées par le serveur, mais seulement les choses qu'elles proposent aux utilisateurs, les différentes fonctions et possibilités proposées.

La page d'accueil

La page d'accueil permet deux choses principales :

- La connexion d'un utilisateur,
- L'affichage des produits en vente actuellement.

Lorsqu'un utilisateur arrive sur le site mais qu'il n'est pas encore connecté, cette page affiche tous les produits en vente sur le site ainsi qu'un formulaire permettant la connexion ou la création de compte. Si l'utilisateur souhaite acheter un produit, il devra se connecter pour pouvoir le faire. En effet, sur la page d'accueil, si l'utilisateur n'est pas connecté, les informations concernant le produit ne sont pas affichées complètement. Notamment, le prix n'est pas affiché et l'utilisateur n'a pas la possibilité de choisir le nombre d'articles qu'il désire commander.

Voici l'allure de cette page avant la connexion :

The screenshot shows a web page with a black header. On the left is a blue elephant logo with 'php' written on its side. In the center, white text reads: 'Site de travail de Pierre Galerneau', 'Etudiant en L3 Informatique', and 'Faculté Libre des Sciences et Technologies de Lille'. On the right is a blue 'e' logo with a globe-like effect.

The main content area has a dark background. On the left is a vertical sidebar with a red 'Accueil' link and a home icon, and a red 'Menu' link with two items: 'WD1' and 'WD2'. The main area is titled 'Pierre Jouet Club' and 'Articles disponibles'. Below this is a login section with the text 'Pour commander ou voir les prix de nos articles, connectez vous ici :'. It contains two input fields for 'Login :' and 'Mot de passe :', a 'Connexion' button, and a 'Créer un compte' link.

Below the login section are three product cards, each with a red border. The first card is for 'Ballon' with an image of a ball and 'Quantité en stock : 43'. The second is for 'Ferrari (Enzo) télécommandée' with an image of a red toy car and 'Quantité en stock : 8'. The third is for 'Game Boy' with an image of a Game Boy console and 'Quantité en stock : 9'.

The footer is black with white text: 'Contact - Hébergement - Mes autres sites - Sites partenaires' and 'Site réalisé par Pierre Galerneau pour le cours de Web Dynamique.'

Lorsqu'un utilisateur connecté accède à cette page, il voit apparaître un affichage différent : il a accès à la commande et donc il voit le prix des articles ainsi que le formulaire permettant la commande d'un ou plusieurs articles. La page d'accueil se présente alors de la manière suivante :

Site de travail de Pierre Galerneau
Etudiant en L3 Informatique
Faculté Libre des Sciences et Technologies de Lille

Accueil

Menu

WD1

WD2

Pierre Jouet Club

Articles disponibles

Deconnexion

Vous êtes connecté sous le pseudonyme **sirice**.

<p style="color: white; font-weight: bold;">Ballon</p> <p style="color: red; font-weight: bold;">Quantité en stock : 43</p> <p style="color: red; font-weight: bold;">Prix : 6€</p> <p style="color: red; font-weight: bold;">Quantite désirée : <input style="width: 50px;" type="text" value="0"/></p> <p style="background-color: white; color: black; padding: 2px;">Ajouter au panier</p>	<p style="color: white; font-weight: bold;">Ferrari (Enzo) téléccommandée</p> <p style="color: red; font-weight: bold;">Quantité en stock : 8</p> <p style="color: red; font-weight: bold;">Prix : 59,99€</p> <p style="color: red; font-weight: bold;">Quantite désirée : <input style="width: 50px;" type="text" value="0"/></p> <p style="background-color: white; color: black; padding: 2px;">Ajouter au panier</p>	<p style="color: white; font-weight: bold;">Game Boy</p> <p style="color: red; font-weight: bold;">Quantité en stock : 9</p> <p style="color: red; font-weight: bold;">Prix : 45,99€</p> <p style="color: red; font-weight: bold;">Quantite désirée : <input style="width: 50px;" type="text" value="0"/></p> <p style="background-color: white; color: black; padding: 2px;">Ajouter au panier</p>
--	--	---

Contact - Hébergement - Mes autres sites - Sites partenaires

Site réalisé par **Pierre Galerneau** pour le cours de **Web Dynamique**.

Un message spécifiant à l'utilisateur sous quel pseudonyme il est connecté apparait ainsi que le menu qui lui permet d'accéder simplement à toutes les pages du site auxquelles il a accès. Enfin, il peut commander les articles disponibles puisque le formulaire de commande est apparu et qu'il connaît maintenant le prix de chaque article proposé.

Création d'un compte

Lorsque l'utilisateur arrive sur notre site et qu'il ne le connaît pas encore, il ne possède pas de compte. Il est donc nécessaire qu'il puisse s'en créer un rapidement. C'est à cela que sert le lien « Créer un compte » sur la page de connexion.

Lorsqu'un utilisateur clique sur ce lien, il est redirigé vers un formulaire lui permettant d'inscrire toutes les données utiles à la création de son compte. Ce formulaire est le suivant :

The screenshot shows a web page with a black header. On the left is a blue elephant logo with 'php' written on its side. In the center, white text reads: 'Site de travail de Pierre Galerneau', 'Etudiant en L3 Informatique', and 'Faculté Libre des Sciences et Technologies de Lille'. On the right is a blue 'e' logo. Below the header is a sidebar with 'Accueil' and 'Menu' in red, and 'WD1' and 'WD2' in white. The main content area is titled 'Pierre Jouet Club' and 'Créer un compte'. It contains a registration form with the following fields: 'Nom :', 'Prenom :', 'Adresse :', 'Code Postal :', 'Ville :', 'Login :', 'Mot de passe :', and 'Confirmation :'. Each field has a white input box and an asterisk to its right. Below the form is a button labeled 'Enregistrer'. At the bottom of the form area, a note reads: 'NB : (*) signifie que le champ correspondant est obligatoire !'. The footer contains the text: 'Contact - Hébergement - Mes autres sites - Sites partenaires' and 'Site réalisé par Pierre Galerneau pour le cours de Web Dynamique.'

Lorsqu'il clique sur le bouton « Enregistrer », l'utilisateur est redirigé vers la page qui crée réellement son compte. Sur celle-ci, il ne voit apparaître qu'un message lui signifiant que son compte a été créé et qu'il peut donc se connecter à l'aide du formulaire qui s'affiche sous le message signifiant que le compte a bien été créé.

The screenshot shows a web application interface. At the top, there is a header with a logo of an elephant with 'php' written on its side, the text 'Site de travail de Pierre Galerneau Etudiant en L3 Informatique Faculté Libre des Sciences et Technologies de Lille', and a blue 'e' logo. Below the header, there is a sidebar on the left with a red 'Accueil' button, a blue house icon, and a red 'Menu' button. Under the menu, there are two links: 'WD1' and 'WD2'. The main content area is titled 'Pierre Jouet Club' and 'Créer un compte'. It contains a message: 'Votre compte a été créé, connectez vous :'. Below this message, there is a login form with two input fields: 'Login :' and 'Mot de passe :'. To the right of these fields is a 'Connexion' button. Below the input fields, there is a link 'Créer un compte'. At the bottom of the page, there is a footer with the text 'Contact - Hébergement - Mes autres sites - Sites partenaires' and 'Site réalisé par Pierre Galerneau pour le cours de Web Dynamique.'

Lorsque l'utilisateur se connecte sur le site, il est redirigé vers la page de connexion qui lui affiche la même chose que la page d'accueil lorsqu'un utilisateur est connecté.

La page de connexion

L'affichage est exactement identique à celui de la page d'accueil donc je ne vais pas le remonter. Cependant, je vais montrer ce en quoi il consiste lorsqu'un utilisateur de type administrateur se connecte au site. En effet, l'administrateur n'ayant pas les mêmes droits que le simple utilisateur, il doit avoir accès à toutes les fonctions du site tout en ayant accès à des fonctions supplémentaires : celles du panneau d'administration. Lors de la connexion d'un administrateur, on affiche donc un lien supplémentaire sur la page de connexion et la page d'accueil qui lui permet d'accéder au moment où il le souhaite aux fonctions d'administration du site.

The screenshot shows a web application interface for 'Pierre Jouet Club'. At the top, there is a header with a 'php' logo (an elephant), the text 'Site de travail de Pierre Galerneau Etudiant en L3 Informatique Faculté Libre des Sciences et Technologies de Lille', and a blue 'e' logo. Below the header is a navigation menu on the left with 'Accueil', 'Menu', 'WD1', and 'WD2'. The main content area is titled 'Pierre Jouet Club' and 'Articles disponibles'. It features a navigation bar with icons for home, shopping cart, Euro symbol, and a user profile, with the text 'Deconnexion' next to the profile icon. Below this is a 'Panneau d'administration' section stating 'Vous êtes connecté sous le pseudonyme **pgalerneau**.' The main area displays three product cards: 'Ballon' (quantity in stock: 43, price: 6€), 'Ferrari (Enzo) télécommandée' (quantity in stock: 8, price: 59.99€), and 'Game Boy' (quantity in stock: 9, price: 45.99€). Each card includes a 'Quantité désirée' input field and an 'Ajouter au panier' button. At the bottom, there is a footer with 'Contact - Hébergement - Mes autres sites - Sites partenaires' and 'Site réalisé par Pierre Galerneau pour le cours de Web Dynamique.'

Le lien ajouté pointe vers la page suivante :

La page d'administration du site

Lorsqu'un administrateur veut réaliser les tâches qui incombent à sa fonction, il doit se rendre sur la page d'administration pointée par le lien présent sur la page d'accueil. Cette page lui présente une liste de toutes les tâches qu'il peut accomplir en temps qu'administrateur à partir du site. Ces différentes tâches sont classées par catégories afin de donner un affichage plus clair à la page d'administration.

En effet, il est important de trier les différentes choses qu'un administrateur a l'autorisation de faire car elles sont nombreuses et divisées en catégories. L'utilisateur peut en effet réaliser 4 grandes familles de tâches :

- Supprimer une base de données,
- Afficher le contenu d'une base ;
- Modifier les bases de données,
- Ajouter des données dans les bases.

Dans ces différentes catégories, il y a différentes actions possibles également, ce qui fait que, si nous n'avons pas trié par catégories, je pense que l'affichage aurait été peu clair et certainement peu agréable pour l'administrateur qui utilisera notre site à plein temps.

Voici cette page d'administration telle qu'elle s'affiche lors de son appel :

Site de travail de Pierre Galerneau
Etudiant en L3 Informatique
Faculté Libre des Sciences et Technologies de Lille

Pierre Jouet Club
Panneau d'administration

Deconnexion

Panneau d'administration

Suppression des BDD
Suppression de la table users (la vider)
Suppression de la table produits (la vider)

Affichage des BDD (contenu)
Contenu de la base users
Contenu de la base produits
Contenu de la base Panier (commandes en attente par utilisateur)

Modification des BDD
Modification d'un produit

Création
Créer un nouveau produit

Contact - Hébergement - Mes autres sites - Sites partenaires

Site réalisé par Pierre Galerneau pour le cours de Web Dynamique.

Détail des pages accessibles à partir du panneau d'administration

Lorsqu'un utilisateur veut supprimer une base de données, un petit message s'affiche simplement pour lui signifier que cette suppression a été effectuée correctement. Je ne vais donc pas montrer les pages générées par ces liens puisqu'elles n'ont rien d'intéressant. Par contre, lorsque l'on veut afficher le contenu de l'une ou l'autre des bases de données, c'est beaucoup plus intéressant. En effet, le résultat de la demande de l'utilisateur est affiché (contenu de la base des produits par exemple), cependant, il est affiché en dessous du panneau d'administration. C'est-à-dire que, si un administrateur clique sur le mauvais lien, il n'aura pas besoin de revenir en arrière pour pouvoir retourner au panneau d'administration. Il n'aura qu'à remonter un petit peu la page et cliquer à nouveau sur le lien qui l'intéresse.

De plus, dans un souci de propreté et de clarté, lorsque la base de données dont l'affichage est demandé contient plus de 10 enregistrements, l'affichage de son contenu est paginé. C'est-à-dire que les 10 premiers enregistrements sont affichés alors que tous les autres ne le sont pas. A la place des autres enregistrements, un petit numéro est affiché en lien. Celui-ci permet, lorsque l'utilisateur clique dessus, d'afficher les utilisateurs suivants.

Si l'administrateur clique sur le lien « 2 », il verra afficher les 10 utilisateurs venant après les 10 premiers dans la base de données, etc.

J'ai réalisé ce petit morceau de code afin de permettre un affichage plus clair et plus convivial. En effet, lorsque la base à afficher ne contient que 10 ou 15 enregistrements, il n'est pas gênant que tout s'affiche sur une seule page. Cependant, lorsqu'elle en contient 10 000, cela deviens moins clair !

Voici l'affichage correspondant à l'affichage de la base des utilisateurs : (base ne contenant que 3 utilisateurs pour l'instant)

The screenshot shows a web application interface for 'Pierre Jouet Club'. The header includes a PHP logo, the site name 'Site de travail de Pierre Galerneau', the user's status 'Etudiant en L3 Informatique', the faculty 'Faculté Libre des Sciences et Technologies de Lille', and a blue 'e' logo. A left sidebar contains 'Accueil', 'Menu', and links 'WD1' and 'WD2'. The main content area is titled 'Pierre Jouet Club' and 'Panneau d'administration'. It features navigation icons for home, shopping cart, Euro, and user profile, with a 'Deconnexion' link. Below are two panels: 'Suppression des BDD' (with options to delete 'users' and 'produits' tables) and 'Modification des BDD' (with a 'Création' option to create a new product). A 'Liste des utilisateurs' table is displayed below.

Login	Nom	Prenom	Adresse	Code Postal	Ville
pgalerneau	Galerneau	Pierre	12 rue Pierre Legrand	59000	Lille
sirice	Odile	Leslie	4 impasse de la corne	39190	Vercia
cgalerneau	Galerneau	Camille	16 rue Jean Jaures	59177	Sains du Nord

At the bottom of the page, there is a footer with links for 'Contact - Hébergement - Mes autres sites - Sites partenaires' and a note: 'Site réalisé par Pierre Galerneau pour le cours de Web Dynamique.'

L'affichage de cette liste lorsque la base contient plus de 10 utilisateur se présente de cette manière :

Page n°1 :

Liste des utilisateurs					
Pages : 12					
Login	Nom	Prenom	Adresse	Code Postal	Ville
pgalerneau	Galerneau	Pierre	12 rue Pierre Legrand	59000	Lille
sirice	Odile	Leslie	4 impasse de la corne	39190	Vercia
cgalerneau	Galerneau	Camille	16 rue Jean Jaures	59177	Sains du Nord
clgalerneau	Galerneau	Claire	16 rue Jean Jaures	59177	Sains du Nord
agalerneau	Galerneau	Armelle	16 rue Jean Jaures	59177	Sains du Nord
mgalerneau	Galerneau	Marc	16 rue Jean Jaures	59177	Sains du Nord
Toto	Toto	Toto	Toto	0	Toto
osimpson	Simpson	Omer	430 spalding way	0	Springfield
msimpson	Simpson	Marge	430 spalding way	0	Springfield

Page n°2 :

Liste des utilisateurs					
Pages : 12					
Login	Nom	Prenom	Adresse	Code Postal	Ville
bsimpson	Simpson	Bart	430 spalding way	0	Springfield
masimpson	Simpson	Maggie	430 spalding way	0	Springfield

Les affichages sont identiques pour les autres bases du site donc je ne vais pas mettre d'imprimé écran pour celles là.

Par contre, je vais maintenant détailler le choix administrateur « modifier un produit ».

Modifier un produit

Lorsqu'un administrateur choisit de modifier un produit, il doit d'abord commencer par choisir celui qu'il veut modifier. La liste des produits existants dans la base de données est alors affichée afin qu'il puisse choisir celui qu'il désire modifier. L'affichage de ce choix est le suivant :


 Site de travail de Pierre Galerneau
 Etudiant en L3 Informatique
 Faculté Libre des Sciences et Technologies de Lille
 

Accueil



Menu

WD1

WD2

Pierre Jouet Club

Panneau d'administration





Deconnexion

Panneau d'administration

Suppression des BDD

Suppression de la table users (la vider)
Suppression de la table produits (la vider)

Affichage des BDD (contenu)

Contenu de la base users
Contenu de la base produits
Contenu de la base Panier (commandes en attente par utilisateur)

Modification des BDD

Modification d'un produit

Création

Créer un nouveau produit

Modifier un produit

Liste des produits

Libelle	Modifier
Ballon	<input type="radio"/>
Ferrari (Enzo) télécommandée	<input type="radio"/>
Game Boy	<input type="radio"/>
	<input type="button" value="Modifier"/>

Contact - Hébergement - Mes autres sites - Sites partenaires

L'administrateur peut alors sélectionner le produit qu'il désire modifier et il ne lui reste plus alors qu'à cliquer sur le bouton « Modifier ». Lorsqu'il a cliqué sur ce bouton, un formulaire s'affiche. Formulaire qui lui permet de modifier les données concernant ce produit enregistrées dans la base :

Modifier un produit

Libelle	Ballon	
Nombre de produits disponibles	43	
Photo		<input style="font-size: small; padding: 2px 5px; border: none; border-bottom: 1px solid black;" type="button" value="Parcourir..."/>
Prix	6	
	<input style="font-size: small; padding: 2px 5px; border: none; border-bottom: 1px solid black;" type="button" value="Modifier"/>	

Le formulaire généré comporte toutes les informations enregistrées concernant le produit. Il suffit à l'utilisateur de les modifier et de valider ses modifications par un clic sur le bouton « Modifier ». Lorsque l'utilisateur valide son choix, il voit un message s'afficher lui spécifiant que sa modification a bien été effectuée et enregistrée sur le serveur.

Enfin, l'administrateur peut effectuer une dernière action sur ce panneau. C'est la création d'un nouveau produit et son ajout à la base de données. Pour cela, il dispose d'un lien lui permettant d'accéder à un formulaire de création dont l'affichage est le suivante :

The screenshot shows a web application interface with a dark theme. At the top, there is a header with a PHP logo (an elephant), the text 'Site de travail de Pierre Galerneau Etudiant en L3 Informatique Faculté Libre des Sciences et Technologies de Lille', and a blue 'e' logo. Below the header is a sidebar on the left with 'Accueil' and 'Menu' sections. The 'Menu' section lists 'WD1' and 'WD2'. The main content area is titled 'Pierre Jouet Club' and contains a form for 'Ajouter un produit'. The form includes fields for 'Nom du produit', 'Nombre de produits disponibles', 'Photo', and 'Prix'. There are also icons for home, shopping cart, Euro symbol, and a user profile icon labeled 'Deconnexion'. Below the form is a 'Panneau d'administration' section with an 'Enregistrer' button. At the bottom of the page, there is a footer with links for 'Contact', 'Hébergement', 'Mes autres sites', and 'Sites partenaires', and a note that the site was realized by Pierre Galerneau for a 'Web Dynamique' course.

Lorsqu'il a rempli tous les champs, l'utilisateur peut cliquer sur le bouton « Enregistrer ». Ce clic lui permet de valider sa transaction et par là même d'enregistrer mes nouvelles données entrées dans la base de données. Un message lui sera alors affiché pour lui signifier que son produit a bien été ajouté à la table `vpc_produit`.

Toutes les possibilités et fonctions qu'un administrateur a en plus d'un simple utilisateur ont maintenant été présentées. Il ne reste donc plus qu'à présenter les fonctions accessibles à tous au moyen du menu et des pages générées.

Commande d'un produit

Lorsqu'un utilisateur décide de commander un produit et qu'il entre une quantité positive inférieure à la quantité en stock dans le champ correspondant sur la page d'accueil (ou de connexion), il commande un article. En cliquant sur le bouton « ajouter au panier », la commande est effective et un message est affiché afin de lui rappeler combien de produits et surtout quel produit il vient de commander. Ce message s'affiche de la manière suivante :

Si un utilisateur commande 5 ballons, on aura :

L'utilisateur peut ensuite vérifier que sa commande est bien prise en compte en se rendant dans l'onglet « Panier ».

Le panier

Lorsqu'un utilisateur a commandé des articles et ne les a pas encore payés, il peut se rendre dans l'onglet « panier » pour visualiser ce qu'il a commandé. Dans cette section du site, il pourra également « rendre » la quantité qu'il souhaite de tel ou tel article qu'il a commandé. Ceci permet à un utilisateur de corriger les erreurs qu'il a pu faire au cours de son parcours sur le site et de ce fait, de ne pas être obligé d'annuler complètement sa commande pour avoir ce qu'il veut. Il peut ne supprimer qu'un seul article de son panier s'il en a envie ou bien les supprimer tous (dans la limite de la quantité que son panier contient évidemment).

De plus, cette page contient un aperçu de la facture de l'utilisateur au moment où il visualise son panier. C'est à dire que la facture affichée contient tous les articles contenus dans le panier. Ceci permet à l'utilisateur d'avoir un aperçu de ce qu'il devra payer au moment où il décidera de le faire.

Voici la page Panier :

Pierre Jouet Club
Contenu du panier

Deconnexion

Panneau d'administration

Ballon

Quantité commandée : 5

Quantité à retirer :

Retirer du panier

Facture :

Article :	Prix Unitaire :	Quantité commandée :	Prix à payer :
Ballon	6€	5	30 €
Total :			30 €

Lorsqu'un utilisateur choisi de supprimer un certain nombre de produit de son panier, il lui suffit d'indiquer le nombre de produits qu'il souhaite supprimer de son panier dans le champ « Quantité à supprimer ». Un fois qu'il aura cliqué sur le bouton « Retirer du panier », un message lui sera affiché, lui signifiant que la quantité X dont il vient de demander la suppression a bien été retirée de son panier. Si cette quantité est négative ou bien qu'elle est supérieure à la quantité commandée, un message d'erreur sera affiché afin que les erreurs de ce type là ne puissent pas être faites.

Une fois que l'utilisateur est satisfait du contenu de son panier, il peut décider de payer. Si c'est le cas, il se rend sur :

La page de paiement

Cette page permet à l'utilisateur de visualiser la facture contenant tous les éléments qu'il a commandés. Celle-ci résume le prix qu'il devra payer au moment du paiement.

Site de travail de Pierre Galerneau
Etudiant en L3 Informatique
Faculté Libre des Sciences et Technologies de Lille

Pierre Jouet Club
Régler ma commande

Deconnexion

Panneau d'administration

Facture :

Article :	Prix Unitaire :	Quantité commandée :	Prix à payer :
Ballon	6€	4	24 €
Total :			24 €

Payer

Contact - Hébergement - Mes autres sites - Sites partenaires

Site réalisé par Pierre Galerneau pour le cours de Web Dynamique.

Lorsque l'utilisateur choisi de payer en cliquant sur le bouton associé dans cette page, ses coordonnées enregistrées dans la base de données sont affichées afin qu'il puisse les vérifier. Il a alors le choix entre :

- Conserver les données affichées pour la livraison,
- Changer les données.

Facture :

Article :	Prix Unitaire :	Quantité commandée :	Prix à payer :
Ballon	6€	4	24 €
Total :			24 €

Voulez vous que nous envoyions votre commande à l'adresse contenue dans votre compte ?

Rappel de vos données enregistrées :

Votre nom :	Galerieau
Votre prenom :	Pierre
Votre Adresse :	12 rue Pierre Legrand
Votre code postal :	59000
Votre ville :	Lille

Si ces données sont correctes, et que vous souhaitez vous faire envoyer la commande en cours de paiement à cette adresse, choisissez "Non" et validez.

Si vous souhaitez vous faire livrer à une autre adresse, choisissez "Oui et validez."

Voulez vous modifier votre adresse de livraison ?

Oui

Non

Valider

Lorsqu'il clique sur le choix « Oui », un formulaire de modification des données va apparaître afin de lui permettre de modifier ce qu'il souhaite :

Changement d'adresse :

Nom :

Preom :

Adresse :

Code Postal :

Ville :



Changer votre adresse de livraison pour cette commande ne modifie pas votre compte ! Si vous voulez modifier l'adresse contenue dans votre compte, allez dans le menu "compte" !

Par contre, si il choisi « Non », c'est-à-dire de ne pas modifier ses coordonnées, il verra s'afficher un lien lui permettant de choisir son moyen de paiement :

Paiement :

Choisissez votre mode de paiement :



Lorsqu'il cliquera sur l'image PayPal, la commande sera payée effectivement. Il verra alors s'afficher un message lui signifiant que sa commande est payée. Il pourra alors se rendre dans le panier et il verra bien que plus aucun article ne s'y trouve. En effet, ils auront été supprimer du fait que la commande a été terminée.

Vous avez choisi le règlement par Paypal.

Votre commande a été payée avec succès, votre panier est maintenant vide !

Merci pour votre achat ! Nous espérons que vous reviendrez !

Enfin ; les seules dernière actions possibles pour un utilisateur quelconque sont celles concernant son compte sur le site.

Le compte utilisateur

Tout comme pour le panneau d'administration, un utilisateur peut effectuer des actions « spéciales », c'est-à-dire qu'il peut modifier ou afficher certaines informations contenues dans la base de données.

Un utilisateur doit pouvoir avoir le contrôle des informations contenues dans son compte. En effet, en cas de déménagement par exemple ; rien n'empêche un utilisateur inscrit de changer d'adresse. C'est pour cela que j'ai mis en place cette section permettant à un utilisateur de :

- Modifier son compte,
- Afficher les données de son compte,
- Supprimer son compte.

Voici l'affichage correspondant :

The screenshot shows a web interface for 'Pierre Jouet Club'. At the top, there is a header with a 'php' logo (an elephant), the text 'Site de travail de Pierre Galerneau Etudiant en L3 Informatique Faculté Libre des Sciences et Technologies de Lille', and a blue 'e' logo. The main content area is divided into a left sidebar and a main panel. The sidebar contains 'Accueil' (with a house icon), 'Menu' (with a menu icon), and two links: 'WD1' and 'WD2'. The main panel is titled 'Pierre Jouet Club' and 'Mon compte'. It features a row of icons: a red house, a green shopping cart, a blue Euro symbol, and a yellow person icon labeled 'Deconnexion'. Below this is a 'Panneau d'administration' section with two columns of buttons. The left column has 'Modification :', 'Modifier mon compte', and 'Affichage :', 'Afficher mon compte'. The right column has 'Suppression :', 'Supprimer mon compte'. At the bottom, there is a footer with 'Contact - Hébergement - Mes autres sites - Sites partenaires' and 'Site réalisé par Pierre Galerneau pour le cours de Web Dynamique'.

J'ai utilisé la même charte d'affichage que pour le panneau d'administration car je considère que c'est plus clair de cette manière.

Modification de compte

Lorsqu'un utilisateur souhaite modifier son compte, il clique sur le lien correspondant dans la page précédente et est redirigé vers la page suivante :

Pierre Jouet Club

Mon compte

Deconnexion

Panneau d'administration

Modification :
Modifier mon compte

Affichage :
Afficher mon compte

Suppression :
Supprimer mon compte

Modifier mon compte :

Données du compte :
Login (pseudo) : pgalemeau
Mot de passe :
Confirmation (mot de passe) :

Données personnelles :
Nom : Galemeau
Preom : Pierre
Adresse : 12 rue Pierre Legrand
Code postal : 59000
Ville : Lille

Modifier

Cette page lui présente toutes les données de son compte et il peut les modifier simplement. Lorsque les modifications lui conviennent, il lui suffit de cliquer sur le bouton « Modifier » et les données relatives à son compte seront modifiées selon ses désirs dans la base de données.

Lorsque cette modification sera effective, un message sera affiché afin de prévenir notre utilisateur que ce qu'il a demandé a été effectué.

Votre compte a été modifié avec succès !

A par modifier son compte, l'utilisateur peut également le supprimer ou l'afficher.

Affichage des données utilisateur

En cliquant sur le lien « Afficher mon compte », l'utilisateur peut afficher les données le concernant dans la base de données. Ceci se fait de la manière suivante :

The screenshot shows a web application interface with a dark theme. At the top, there is a header with a PHP logo (an elephant), the text 'Site de travail de Pierre Galerneau Etudiant en L3 Informatique Faculté Libre des Sciences et Technologies de Lille', and a blue 'e' logo. Below the header is a sidebar on the left with 'Accueil' and 'Menu' sections. The main content area is titled 'Pierre Jouet Club' and 'Mon compte'. It features several icons: a red house, a green gear, a blue Euro symbol, and a yellow person icon. Below these icons are buttons for 'Modification : Modifier mon compte', 'Affichage : Afficher mon compte', and 'Suppression : Supprimer mon compte'. A 'Deconnexion' link is also present. Underneath, there is a section 'Vos données :' listing user information: Login (pgalerneau), Nom (Galerneau), Prenom (Pierre), Adresse (12 rue Pierre Legrand), Code postal (59000), and Ville (Lille). At the bottom, there is a footer with 'Contact - Hébergement - Mes autres sites - Sites partenaires' and 'Site réalisé par Pierre Galerneau pour le cours de Web Dynamique.'

Enfin, s'il souhaite supprimer son compte, un message lui sera affiché signifiant que son compte a bien été supprimé selon sa demande !

3.2 Les différents types d'utilisateurs

Lorsque j'ai du penser à la modélisation de mon projet et à la manière dont j'allais permettre aux deux types d'utilisateurs de se différencier, j'ai choisi d'utiliser l'objet PHP session qui permet de conserver dans des variables les informations importantes concernant un utilisateur. Ceci permet donc de savoir si un utilisateur est connecté lorsqu'une page est demandée mais également de savoir si cet utilisateur connecté est bien du type autorisé. C'est-à-dire qu'un simple utilisateur ne doit pas pouvoir accéder à des pages administrateur comme le menu ou la suppression de base de données. Grâce à l'objet session et à la transmission de la session de page en page, il est possible d'afficher des messages lorsque l'utilisateur tentant de se connecter à une page n'est pas du type autorisé par exemple. Notamment, lorsqu'un simple utilisateur tente de se connecter sur des pages accessibles seulement aux administrateurs, un message s'affiche lui spécifiant qu'il n'a pas accès à cette partie du site. Évidemment, les liens vers ces pages n'apparaissent que sur les sessions administrateur connectées, cependant, un simple utilisateur pourrait y accéder s'il modifiait l'url de la page par exemple. Il pourrait trouver, par hasard la bonne url et tomber sur la page administrateur.

C'est pour cela qu'il est important de vérifier à chaque début de page que le type d'utilisateurs connectant est bien celui qui est autorisé.

3.2.1 Mise en place des sessions de la VPC

Lorsque j'ai du choisir quelles informations j'allais stocker dans mes sessions pour la réalisation du magasin de vente par correspondance en ligne, j'ai choisi, après réflexion, d'enregistrer deux données principales :

- Le login de l'utilisateur,
- L'identifiant de l'utilisateur.

En effet, il est utile de conserver le login de page en page car cela permet par exemple d'afficher des messages personnalisés à chaque utilisateur qui se connecte sur le site. De même, la sauvegarde de l'identifiant de l'utilisateur connecté permet de ne pas avoir à le rechercher à chaque fois que l'on en a besoin dans un script ou un autre. En effet, lorsque seul le login est enregistré, dès que l'on doit afficher quelque chose de propre à l'utilisateur comme une de ses commandes par exemple, il est nécessaire de rechercher son identifiant. Si on le conserve toujours dans la session, cela nous évite d'avoir à le rechercher à chaque fois et cela fait gagner un temps certain et évite l'exécution excessive de requêtes inutiles.

Comme lors du TP précédent (la DVD Tech), il est nécessaire de conserver la session créée de page en page. C'est pourquoi la ligne :

Code

```
Session_start();
```

Est insérée à chaque début de nouvelle page afin de récupérer la valeur de la session existante dans la page précédente.

En effet, cette ligne permet la création d'une nouvelle session cependant, elle permet également d'autres choses. En effet, cette ligne cherche à récupérer une session existante dans la page précédente. Si elle n'en trouve pas, alors elle en créera une nouvelle. C'est cette propriété que nous utilisons dans notre page de connexion pour la création de la session. Cependant, dans les autres pages du site, comme une session existera déjà dans la page précédente, cette ligne de code la récupérera et donc les variables enregistrées dans celle-ci seront accessibles de n'importe quelle page du site à partir du moment où cette ligne se trouve bien à la tête du fichier correspondant.

3.2.2 Intérêt de l'objet session

Cet objet permet de simplifier la gestion des données utilisateurs. En effet, il serait techniquement possible de faire passer les informations importantes de page en page à l'aide de formulaires HTML contenant des champs cachés. C'est en partie ce que nous avons fait en VBScript au 1^{er} semestre. Cependant, cette technique est très contraignante et assez peu pratique. L'objet session est donc beaucoup plus simple et pratique à utiliser puisqu'il permet la gestion de toutes ces informations à l'aide d'une simple ligne dans chaque fichier. Lorsque la ligne de démarrage de session est incluse dans un fichier, toutes les informations sur la session courante sont accessibles au moyen d'un simple tableau associatif. Ceci permet donc un gain de temps et de clarté non négligeable puisque la transmission de toutes les informations et leur récupération se fait en une seule ligne par fichier !

3.3 Quelle présentation adopter ?

L'un des objectifs principaux de notre travail en cours de Web Dynamique est de réaliser des sites les plus professionnels possibles. Il faut donc, pour cela, réaliser les interfaces les plus conviviales possibles afin que l'utilisateur ai du plaisir à naviguer sur notre site et n'hésite pas à y revenir régulièrement.

Pour commencer, j'ai donc décidé de mettre un titre. Comme dans le TP précédent, je n'ai pas fais de bannière afin de ne pas surcharger le design du site. Cependant, si ce site devait être réalisé dans un autre contexte, il serait certainement agrémenté d'une bannière.

Le magasin de vente par correspondance que j'ai créé étant un magasin de vente de jouet, le titre des pages est :

Pierre Jouet Club

De plus, concernant le design, j'ai décidé de faire un menu dont les liens étaient matérialisés par des images à la place des simples boutons HTML que j'utilisais jusque maintenant ce qui, je trouve, donne un coté beaucoup plus « fun » et « beau » à mon site.



Enfin, le choix du design général a été fait en surfant sur des vrais sites de vente en ligne. En effet, j'ai copié en partie leur design parce que je trouvais que les affichages de produits cote à cote à l'aide de DIV ou de tableaux HTML rendait bien et donnait une image claire et précise de ce que je site vendait.

Maintenant que toutes mes motivations et mes choix sont expliqués, il me faut présenter et expliquer la programmation réalisée.

4 Explication du code

4.1 Arbres programmatiques et explication hors programmation

Dans ce paragraphe, je vais expliquer les algorithmes principaux hors langage de programmation à l'aide d'arbres programmatiques afin de rendre accessible à tous la compréhension des algorithmes.

4.1.1 Algorithme de connexion au site et affichage des produits en vente

Dans la page connexion, la première du site dans laquelle un algorithme intervient, on retrouve deux algorithmes principaux :

- Celui permettant la connexion,
- Celui permettant l'affichage formaté de tous les produits disponibles à l'achat.

En plus de ces deux algorithmes principaux, on en retrouve un troisième dont la structure se retrouvera sur toutes les pages du site (sauf celle de connexion et de déconnexion), qui est la partie vérifiant que l'utilisateur est bien connecté et a bien le droit d'être là.

L'objectif de ce morceau de code est de vérifier qu'un utilisateur ne se trouve pas par hasard sur une page à laquelle il n'est pas censé accéder. Qu'elle soit appelée dans le but d'afficher une page où une autre, ce sera toujours le même code qui sera exécuté.

Je vais commencer par présenter l'algorithme de connexion :

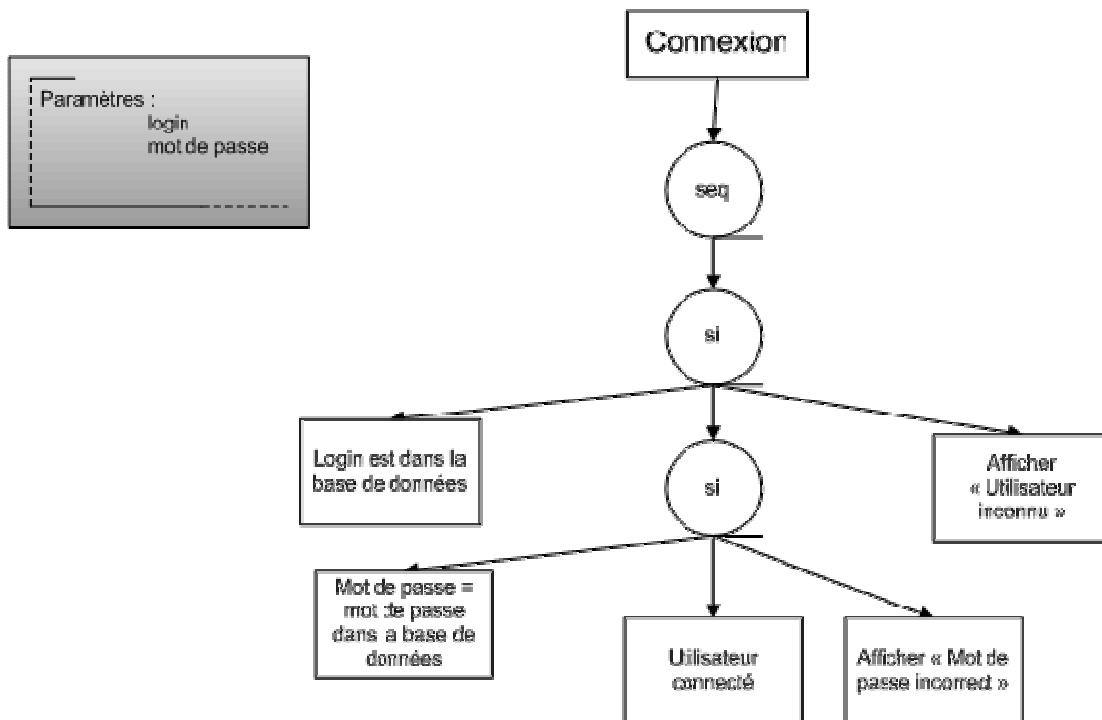
Cet algorithme est simple. En effet, la vérification des informations saisies par l'utilisateur est plus simple à réaliser lorsque l'on utilise des bases de données que lorsque l'on les simule à l'aide de fichiers texte.

L'algorithme commence donc par récupérer les valeurs que l'utilisateur a saisies dans le formulaire de connexion. Il récupère donc deux valeurs :

- Un login ;
- Un mot de passe.

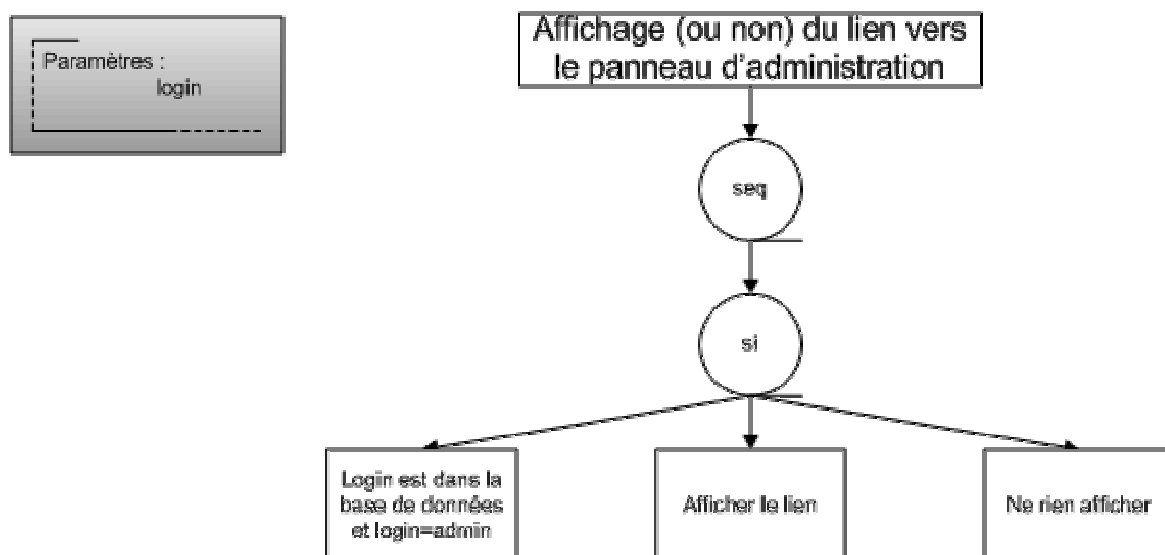
Ces valeurs doivent maintenant être vérifiées. En effet, il faut que le login existe dans la base de données pour que l'on puisse connecter l'utilisateur. De plus, le mot de passe qu'il a saisi doit être identique à celui qui est enregistré dans la base.

Il suffit donc de « chercher » dans la base `vpc_users` une ligne dont le login est égal à celui qui a été saisi par l'utilisateur. S'il n'y en a pas, alors on affiche un message spécifiant que le login n'existe pas. Par contre, si une ligne est trouvée, alors on vérifie que le mot de passe enregistré est bien identique à celui que l'utilisateur a saisi dans le formulaire de connexion. Si c'est le cas, alors l'utilisateur est connecté, sinon, un message lui est affiché, signifiant que son mot de passe est incorrect.

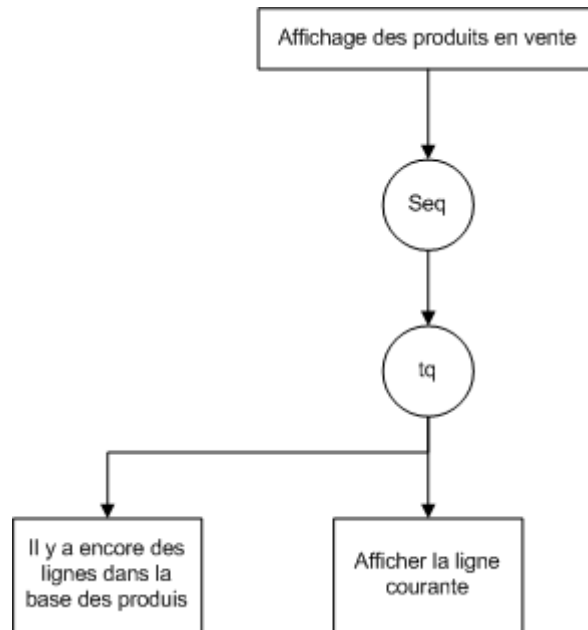


L'objectif de cet algorithme est, comme nous l'avons dit plus haut, de connecter un utilisateur au site ou, du au moins, de vérifier que les informations qu'il a entré comme informations de connexion sont valides.

Une fois connecté, il faut sauvegarder les informations qui permettront de reconnaître cet utilisateur de page en page dans les variables de session. De plus, lorsqu'un utilisateur se connecte, il est nécessaire de savoir si c'est un administrateur ou pas afin d'afficher ou non le lien lui donnant accès à la page d'administration du site. Pour cela, j'ai mis en place un petit algorithme :



Enfin, pour chaque utilisateur, qu'il soit administrateur ou non, il faut afficher la liste des produits en vente actuellement avec leur prix et leur quantité disponible. Pour cela, on met en place l'algorithme suivant :



Cet algorithme se contente en fait d'exécuter une requête sur la base des produits (tout sélectionner) et d'afficher ensuite tous les résultats obtenus. C'est donc extrêmement plus simple d'afficher tout le contenu d'une table lorsque les données sont enregistrées dans des bases de données réelles que lorsqu'elles sont enregistrées dans des fichiers textes. En effet, l'algorithme ici se contente de récupérer toutes les données et de les afficher alors que dans un fichier, il faut tout parcourir ligne par ligne pour récupérer les données. Ici, tout est récupéré au sein d'un tableau. Il ne reste plus alors qu'à afficher le contenu du tableau en le formatant comme on le souhaite !

L'algorithme permettant d'afficher tous les produits enregistrés sur le site est exactement identique à celui qui permet, sur le panneau administrateur, d'afficher le contenu de la base des utilisateurs ou à celui qui permet l'affichage de la base vpc_panier. Je ne vais donc pas détailler ces deux algorithmes puisque ce serait une redite inutile.

Une fois que l'on a présenté les algorithmes permettant de lister les bases de données, il est intéressant d'expliquer ceux qui, au contraire, permettent d'ajouter des données dans ces bases comme :

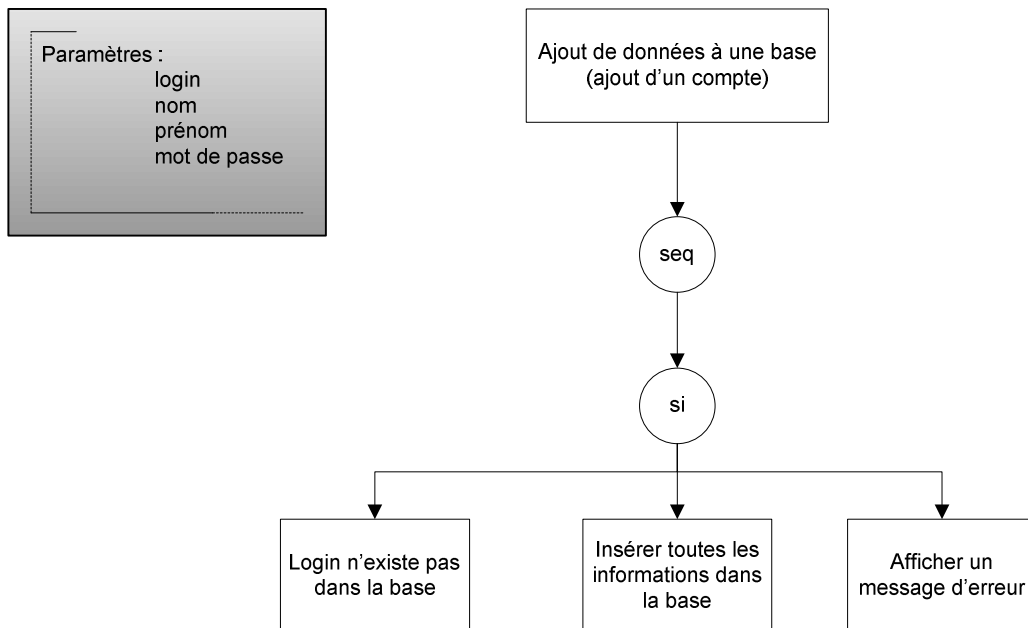
- Ajout de produit,
- Ajout d'utilisateur,
- ...

4.1.2 Algorithme d'ajout d'un enregistrement

Lorsqu'un utilisateur quelconque a besoin d'ajouter un enregistrement dans l'une des tables de notre base de données, il peut s'agir de différentes situations. En effet, cet ajout peut, être fait consciemment ou inconsciemment puisque l'utilisateur peut :

- Créer un compte (ajout conscient),
- Ajouter un produit (s'il est administrateur),
- Ajouter une commande a son panier (ajout inconscient).

Ces mécanismes reposent sur un algorithme principal : celui qui permet d'ajouter des données à une table, dont voici l'arbre programmatique :

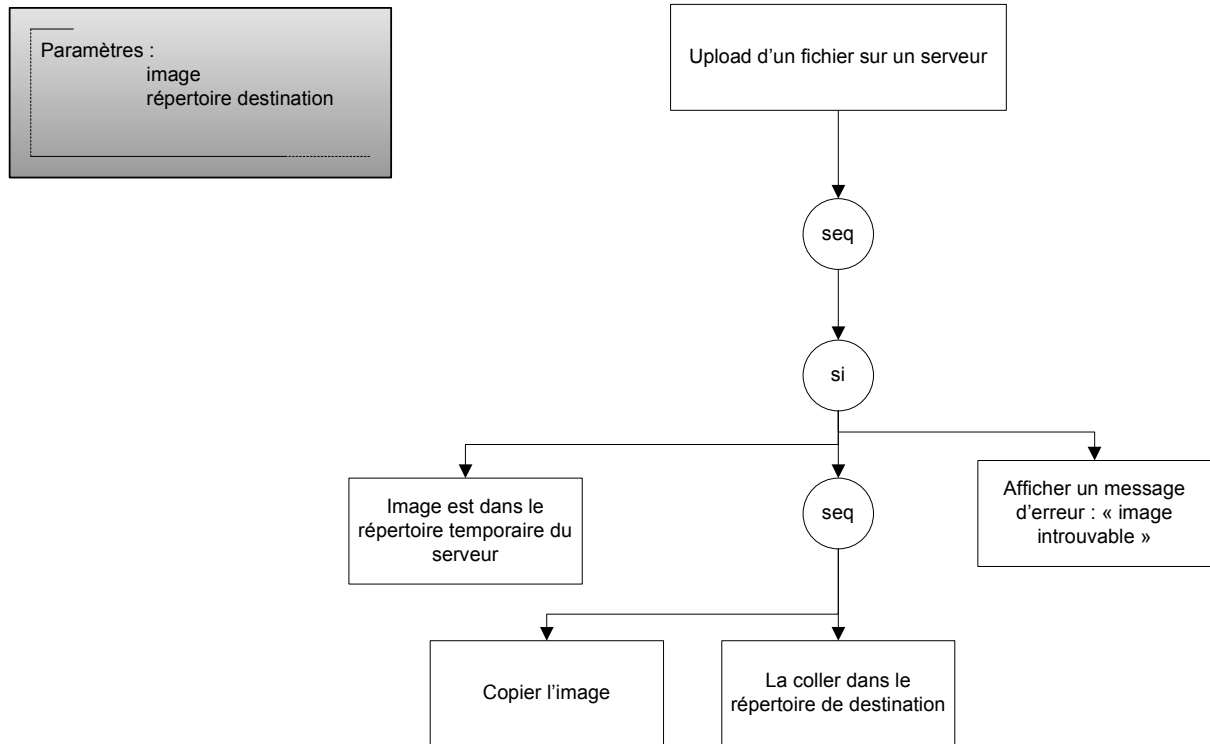


Cet algorithme est également valable pour insérer un nouveau produit dans la base ou pour insérer une nouvelle commande dans la table vpc_panier. Le principe est le même car, à chaque fois, il ne s'agit que de l'insertion d'une ligne dans la table donc l'exécution d'un simple requête SQL.

4.1.3 Upload d'un fichier sur le serveur

Comme je l'ai dit dans le TP précédent, un de mes objectifs a été de réaliser un script permettant d'uploader des fichiers sur le serveur hébergeant mon site afin que, lorsqu'un utilisateur ajoute un nouveau produit à la base, il puisse choisir l'image qu'il souhaite. En effet, sans ce script, les images n'étaient pas téléchargées sur le serveur, il fallait donc qu'un super utilisateur se connecte directement au serveur pour uploader les images sur celui-ci grâce à un logiciel de transfert FTP. Ceci n'était pas très pratique car cela empêchait l'administrateur d'avoir tout contrôle sur le contenu de son site.

J'ai donc décidé de mettre en place un script d'upload. Après quelques recherches, j'en suis arrivé à un algorithme dont le schéma est :



Maintenant que l'on sait comment on peut :

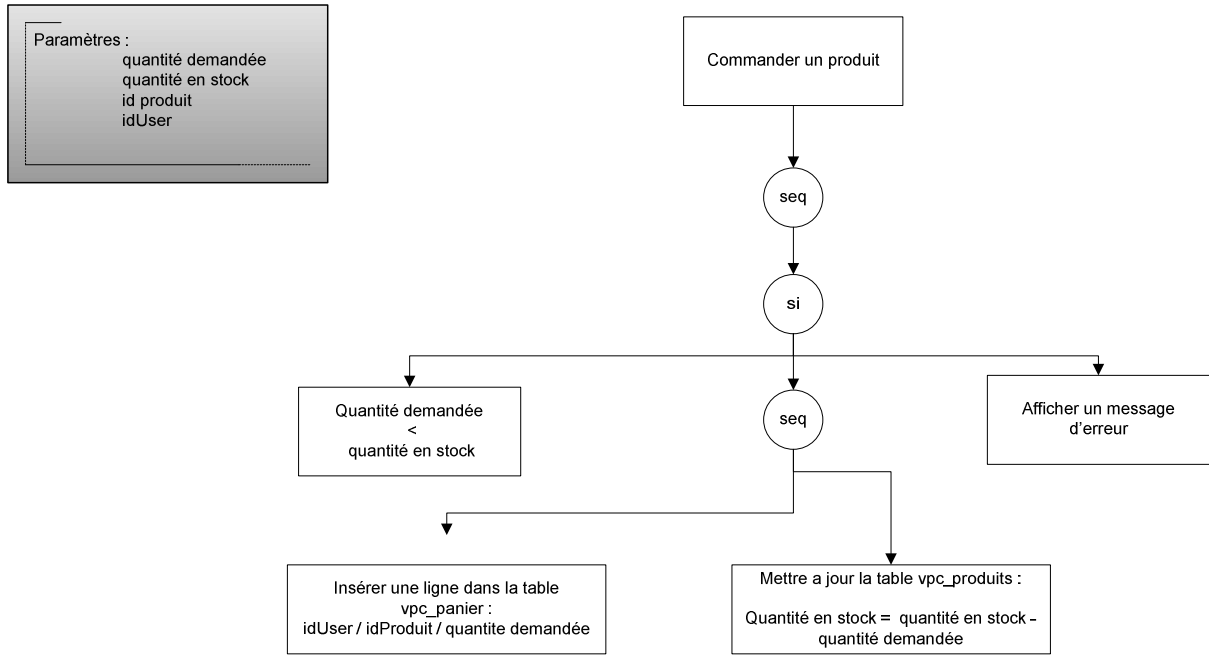
- Afficher le contenu d'une table,
- Ajouter un enregistrement à une table,
- Enregistrer un fichier sur le serveur,

Il faut s'intéresser à l'algorithme permettant de commander un produit dans la quantité demandée.

4.1.4 Algorithme de commande d'un produit

Lorsqu'un utilisateur se trouve sur la page d'accueil du site et qu'il voit la liste des produits s'afficher, il doit pouvoir commander une certaine quantité du produit qu'il souhaite. Pour cela, j'ai mis au point un algorithme permettant de vérifier que la quantité demandée par l'utilisateur est bien inférieure à la quantité en stock. Si c'est le cas, alors la quantité demandée de ce produit est ajoutée à la base vpc_panier et retirée de la base vpc_products. Ceci permet donc de gérer la commande (non payée) tout en gérant les stocks de produits.

Voici l'algorithme permettant de réaliser ceci :



4.1.5 Modification de données

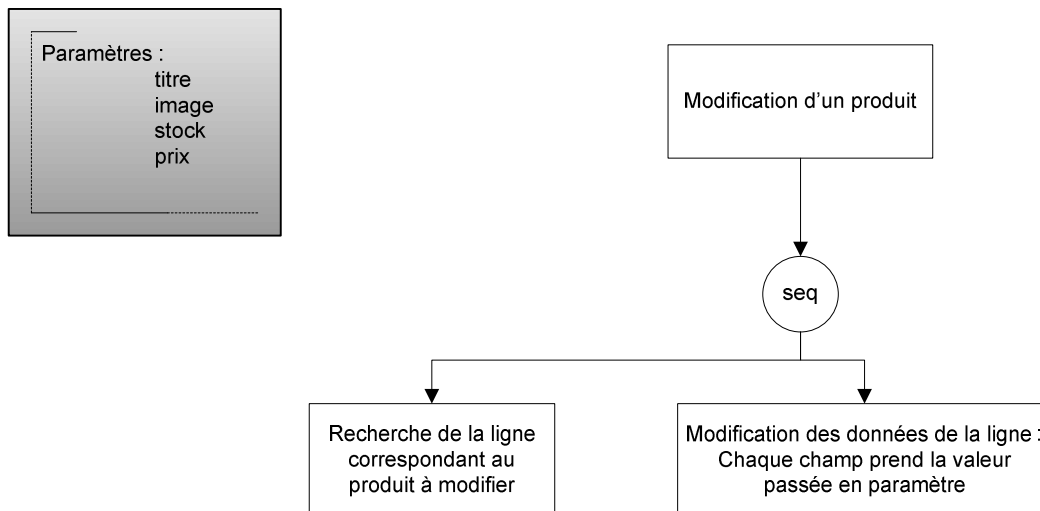
Un des derniers algorithmes principaux de mon site est l’algorithme de modification de données existantes. Celui-ci permet à un utilisateur de modifier différentes choses :

- Son compte (tous les utilisateurs),
- Son panier (tous les utilisateurs),
- Un produit enregistré (administrateur),
- ...

Il est donc important de le présenter sous forme d’arbre programmatique car il revient assez régulièrement dans ce site.

Dans l’arbre suivant, je vais prendre l’exemple de la modification d’un produit, cependant, l’algorithme est sensiblement le même lorsque s’agit de modifier un compte utilisateur ou un panier.

En voici la structure :



Cet algorithme est, encore une fois, très simple puisque l'utilisation de base de données permet, en l'exécution d'une seule requête, de récupérer toutes les données correspondant à un enregistrement. Pour les modifier, il suffit donc de sélectionner la bonne ligne et de la modifier champ par champ !

Pour conclure cette partie, il est important de remarquer que l'utilisation de bases de données et de SQL simplifie grandement le travail. En tout cas, cela permet de réaliser les choses plus simplement qu'en utilisant des fichiers séquentiels en général.

4.2 Détail des fonctionnalités de chaque page

Maintenant que j'ai expliqué les principaux algorithmes et leur forme ci-dessus, je vais parcourir tous les fichiers du projet pour les expliquer un à un afin de m'arrêter plus particulièrement sur chaque particularité de programmation dans ce que j'ai développé. Je vais donc tout expliquer en détail afin que tout le côté « code » soit expliqué et clair.

4.2.1 L'arrivée sur le site : la page *accueil.php*

Lorsqu'un utilisateur arrive sur mon site, il arrive sur la page « *accueil.php* »¹. Il verra alors apparaître différentes choses :

- Un formulaire de connexion,
- La liste des produits disponibles ainsi que leur quantité en stock.

Cette page commence en réalité, lorsqu'elle est chargée sur le serveur, par vérifier d'où vient l'utilisateur et si une session est déjà ouverte sur son ordinateur. S'il vient de la page de déconnexion, alors un message lui signifiant qu'il a été déconnecté avec succès sera affiché. Ceci est fait grâce au code suivant :

Code

```

if($_SERVER["HTTP_REFERER"] == "http://localhost/WD1/fiche%2017/deco.php"){
    echo "<div class=erreur>Vous avez été déconnecté de votre compte.</div><br>";
}
  
```

¹ Le code correspondant à la page *accueil.php* est disponible en annexe 1.

Cette condition vérifie en fait quelle est la page précédente visitée par l'utilisateur et, si c'est la page de déconnexion, alors un message est affiché. Si ce n'est pas celle là (ou que l'utilisateur n'en a pas encore visité), alors rien n'est affiché.

Ensuite, une seconde vérification est effectuée : celle permettant de savoir si l'utilisateur est déjà connecté ou non (si une session – non « morte »- existe déjà sur le navigateur de l'utilisateur). Ceci est fait de la manière suivante :

Code

```
if(isset($_SESSION['login'])){
```

Si la session n'est pas initialisée, alors on affiche le formulaire de connexion, sinon on ne l'affiche pas. Dans tous les cas, on affiche ensuite la liste des produits disponibles dans la base de données (table vpc_produits). A l'affichage, il faut tout de même faire attention à ce que l'on n'affiche que les produits dont la quantité disponible est supérieure à 0. En effet, il serait inutile d'afficher les produits dont la quantité disponible vaut 0 puisque l'utilisateur ne pourrait alors pas les commander !

Concernant l'affichage de la liste des produits, j'ai décidé de les afficher sous forme de DIV HTML. C'est-à-dire que chaque produit (son nom, son image et sa quantité disponible) sont contenus dans un bloc HTML DIV dont la classe est définie dans la feuille de style CSS incluse au document.

Pour chaque produit disponible dans la base, on aura alors un code ressemblant à :

Code

```
<div style="{float:left; border-width:1px; border-color:red; border-style:outset; padding:5px; margin:5px;}">
<?php echo $donnees['nom'];?>
<hr style="{color:red;}">

<hr style="{color:red;}">
<?php echo "<font color=red><b>Quantité en stock : </b></font>". $donnees['nbDispo'];?>
</div>
```

Ce qui permet que tous les produits soient affichés de la même manière en prenant à peu près la même place sur la page.

Remarque : Dans le code présenté ci-dessus, les variables « \$donnees[...] » correspondent aux données récupérées de la base de données. C'est-à-dire que toutes les données de chaque produit sont contenues dans ces variables. Il suffit donc de les récupérer au moyen de leur nom.

Si la ligne présenté précédemment renvoie le fait qu'il existe déjà une session utilisateur « en vie » sur cet ordinateur, alors l'algorithme affiche un message de bienvenue signifiant au visiteur qu'il est connecté sous le pseudonyme X et la liste des produits est affichée tout comme lorsque l'utilisateur n'est pas connecté à ceci prêt qu'un formulaire permettant la commande est ajouté aux DIV contenant les produits.

Lorsqu'un utilisateur n'est pas connecté et qu'il souhaite le faire, alors il a deux choix :

- S'il possède déjà un compte, il se connecte directement grâce au formulaire prévu à cet effet,

- S'il ne possède pas de compte, il peut en créer un grâce à la page :

4.2.2 Créer un compte : *créer_compte.php / créer.php*

Lorsqu'un utilisateur décide de créer un nouveau compte, il clique sur le lien « Créer un compte » de la page d'accueil. Celui-ci le renvoie alors vers la page *créer_compte.php*² qui n'est en réalité qu'un simple formulaire de saisie. Celle-ci va permettre à l'utilisateur de saisir toutes les informations nécessaires à la création de son compte dans la base de données.

Lorsque ce formulaire est validé par l'utilisateur (et que donc il a rempli tous les champs), la page *créer.php*³ est appelée. C'est en réalité celle-ci qui va réellement créer le compte du nouvel utilisateur.

Cette page commence bien sur par récupérer les données que l'utilisateur vient de saisir dans le formulaire. Pour cela, la ligne :

Code

```
$variable=$_POST["nom"];
```

Est utilisée autant de fois qu'il y a de variables (valeurs) à récupérer. Une fois celles-ci stockées dans des variables, il faut vérifier la validité des informations saisies par le nouvel arrivant. En effet, les données doivent avoir les propriétés suivantes :

- Nom : chaîne de caractère non vide,
- Prénom : chaîne de caractère non vide,
- Adresse : chaîne de caractères non vide,
- Code Postal : nombre dont le nombre de chiffre est égal à 5,
- Ville : chaîne de caractères non vide,
- Login : chaîne de caractère non vide,
- Mot de passe 1 : chaîne de caractères non vide et supérieure ou égale à 6 caractères,
- Mot de passe 2 : chaîne de caractères non vide égale à la chaîne mot de passe 1.

Si les données saisies par l'utilisateur ne sont pas de ce format, alors un message d'erreur sera affiché afin de le prévenir des erreurs qu'il a pu commettre. Il pourra alors les corriger et retenter de créer son compte.

Par contre, lorsque les données seront valides, le compte sera créé au moyen d'une requête SQL de type insert dont la syntaxe sera la suivante :

Code

```
Insert into vpc_users value(s', '$nom', '$prenom', '$vile', $codepostal, '$login', '$motdepasse');
```

Une fois cette requête exécutée au moyen des lignes suivantes :

Code

```
$requete="insert into vpc_users values(", '$nom', '$prenom', '$adresse', '$cp', '$ville', '$login', '$pwd1');";  
$base=mysql_connect("localhost", "root", "");  
if(mysql_select_db("galerieau_pierre", $base)){  
    if($reponse = mysql_query($requete)){  
        echo "Votre compte a été créé, connectez vous :";
```

² Le code correspondant à la page *créer_compte.php* est disponible en annexe 2.

³ Le code correspondant à la page *créer.php* est disponible en annexe 3.

```
}  
}  
Mysql_close();
```

L'utilisateur sera enregistré et pourra donc se connecter au site. Un formulaire de connexion lui sera donc affiché sur la page créer.php.

Maintenant que l'on sait de quelle manière un utilisateur peut se créer un compte, il faut voir comment le programme va le connecter lorsqu'il demandera la connexion au moyen du formulaire de connexion présent sur la page d'accueil du site.

4.2.3 Connexion d'un utilisateur : la page connect.php

Lorsqu'un utilisateur décide de se connecter, il remplit les deux champs que contient le formulaire de connexion présent :

- Soit sur la page d'accueil,
- Soit sur la page de création de compte.

Ce formulaire lui demande simplement de s'identifier en précisant son login et son mot de passe, préalablement enregistrés dans la base de données.

Lorsque ce formulaire est validé, la page connect.php⁴ est appelée. Elle commence donc par récupérer les valeurs saisies par l'utilisateur dans le formulaire de connexion prévu à cet effet. Ces données étant récupérées de la même manière que celle présentée ci-dessus, je ne vais pas le réexpliquer.

Lorsque les données ont été récupérées et stockées dans des variables, il est nécessaire de commencer par vérifier si le login saisi correspond bien à un login existant dans la base. Pour cela, une requête SQL de type select est exécutée. En voici la forme :

Code

```
$requete="select * from vpc_users where Login='$login';"
```

Cette requête permet de récupérer toutes les lignes de la table vpc_users dont le champ login est égal au login saisi par l'utilisateur dans le formulaire de connexion. Il est ensuite nécessaire de connaître le nombre de résultats de cette requête. En effet, si elle n'a aucun résultat, il faut afficher un message signifiant que le login n'existe pas. Par contre, si un résultat est trouvé, il faut alors vérifier que le mot de passe associé est bien égal à celui que l'utilisateur a saisi. Si c'est le cas, alors l'utilisateur est connecté.

Lorsqu'il est connecté, l'utilisateur voit apparaître un message lui signifiant sous quel pseudonyme il est connecté ainsi que la liste des produits disponibles affichés de la même manière que dans la page d'accueil (comme je l'ai expliqué plus haut).

De plus, si l'utilisateur connecté est administrateur, alors un lien vers le panneau d'administration lui sera affiché.

Enfin, le menu sera affiché pour tous les types d'utilisateurs.

⁴ Le code correspondant à la page connect.php est disponible en annexe 4.

Du fait que le contenu de la base des produits est affiché, l'utilisateur peut commander ce qu'il souhaite à partir de cette page. Je vais donc présenter maintenant la page commande.php qui permet de commander une certaine quantité d'un article sélectionné.

4.2.4 *La page de commande : commande.php*

Lorsqu'un utilisateur souhaite commander un produit, il peut le faire soit sur la page de connexion soit sur la page d'accueil. Pur cela, il lui suffit de choisir une quantité à commander dans le formulaire correspondant à l'article qu'il souhaite ajouter à son panier. Lorsque cliquera sur le bouton de validation de commande, la page commande.php sera alors lancée.

Dans cette page, il n'y aura quasiment aucun affichage. En effet, elle se contente de réaliser les tests et traitements associés à l'enregistrement d'une commande.

Lorsqu'elle est appelée, la page commande.php⁵ commence par récupérer l'identifiant du produit que l'utilisateur a voulu commander. Une fois ceci fait, il ne reste alors plus qu'à récupérer la quantité commandée par l'utilisateur. Cependant, avant d'enregistrer la commande, il est nécessaire de vérifier que la quantité commandée par l'utilisateur n'est pas supérieure à la quantité disponible dans la base de données ! L'algorithme récupère donc au moyen d'une requête SQL la valeur de la quantité disponible du produit sélectionné et compare cette valeur avec celle que l'utilisateur a entré comme quantité souhaitée. Si la quantité souhaitée est supérieure à la quantité en stock, alors un message d'erreur est affiché signifiant à l'utilisateur qui doit commander une quantité comprise entre 0 et la quantité en stock.

Par contre, si la quantité commandée est inférieure ou égale à la quantité en stock, alors une requête SQL de type insert est exécutée sur la table vpc_panier. En effet, il est nécessaire d'enregistrer, pour chaque commande, l'identifiant de l'utilisateur, l'identifiant du produit commandé, ainsi que la quantité commandé.

Une fois cette requête exécutée et la commande enregistrée, il ne reste donc plus qu'à modifier la table vpc_produits afin que la quantité en stock du produit commandé par l'utilisateur soit mise à jour. En effet, si un produit est en stock pour 20 unités et qu'un utilisateur en commande 5, alors il faut que le champ nbDispo de la base vpc_produits soit mis à jour afin qu'il ne contienne plus que 15 (20 – 5) produits disponibles.

Lorsqu'un utilisateur a effectué une commande, il peut ressentir le besoin de la vérifier. Pour cela, il peut se rendre dans l'onglet panier du menu. Celui-ci va lui permettre d'afficher en détail tous les objets qu'il a commandés. Il pourra également voir quelle quantité il en a commandé et surtout, supprimer certains objets. Ceci est fait sur la page :

4.2.5 *Affichage du panier : la page panier.php⁶*

Lorsqu'un utilisateur souhaite vérifier le contenu de son panier, il lui suffit de se rendre sur la page panier.php. Celle-ci va parcourir la table vpc_panier à la recherche de commandes dont l'identifiant client correspond à l'identifiant du client connecté. Lorsqu'elle en aura trouvé,

⁵ Le code correspondant à la page commande.php est disponible en annexe 5.

⁶ Le code correspondant à la page panier.php est disponible en annexe 6.

elle les affichera, associées à un formulaire permettant de retirer de la commande un certain nombre d'articles.

L'affichage du contenu du panier d'un utilisateur sera réalisé de la même manière que l'affichage du contenu d'une des bases de données. En effet, il suffit de parcourir la base `vpc_panier` pour trouver les enregistrements correspondants à un utilisateur donnée. Cependant la différence est que, pour chaque enregistrement trouvé, une seconde requête sera effectuée, sur la base `vpc_produits` cette fois, afin de récupérer toutes les données associées au produit commande (son libellé, son image,...) afin que l'affichage soit plus pratique et « beau » pour l'utilisateur.

De plus, l'affichage du formulaire de retour d'articles permettra à l'utilisateur de rendre la quantité qu'il souhaite de l'un des articles qu'il possède (dans la limite de la quantité réservée bien sur).

Pour conclure sur cette page, il faut signaler qu'elle affiche également un aperçu de la facture de l'utilisateur. En effet, tous les produits commandés par l'utilisateur sont réunis dans cette facture et le prix que l'utilisateur aura à payer est affiché (produit par produit et total). Ceci permet un rendu plus réaliste et plus professionnel je trouve.

Pour rendre un de ses articles, l'utilisateur devra remplir le formulaire concerné et se rendre sur :

4.2.6 La page de retour d'articles : retirer.php

Lorsqu'un utilisateur se rend compte qu'un article dans son panier ne lui plait plus ou qu'il en a commandé une quantité trop importante, il peut choisir de le supprimer.

Dans la page `panier.php`, un formulaire est ajouté à chaque produit, permettant de spécifier le nombre de produits à supprimer de la commande. Lorsque ce formulaire est validé, la page `retirer.php`⁷ est appelée et c'est elle qui s'occupe de réaliser les traitements adéquats.

Elle commence donc par récupérer l'identifiant du produit duquel il faudra modifier la quantité commandé, puis elle récupère la valeur que l'utilisateur souhaite retirer de sa commande. Des tests sont alors réalisés afin de savoir si la quantité que l'utilisateur souhaite retirer de son panier est bien inférieure ou égale à la quantité que contient le panier. Si c'est le cas, alors la ligne correspondant à la commande de ce produit est mise à jour dans la base `vpc_panier`, c'est-à-dire que la quantité commandée est décrétementée de la valeur indiquée par l'utilisateur.

Lorsque cette quantité à retirer est égale à la quantité commandée, le produit est simplement retiré de la table `vpc_panier`. En effet, il serait illogique que le panier d'un utilisateur contienne des produits dont la quantité commandée est égale à 0 !

Il faut tout de même constater que ceci n'est pas le seul traitement à réaliser pour qu'une commande soit belle et bien annulée. En effet, si on se contente de supprimer une certaine quantité de la table `vpc_panier`, les stocks ne sont pas gérés. En effet, lorsqu'un utilisateur supprime des produits de son panier, ils ne se volatilisent pas, ils doivent retourner en stock.

⁷ Le code correspondant à la page `retirer.php` est disponible en annexe 7.

Pour cela, il faut donc exécuter une requête de modification de type update sur la table vpc_produits afin de rajouter à la quantité en stock du produit concerné, la quantité que l'utilisateur a choisi de retirer de son panier.

Un utilisateur, en plus de commander des articles, les rendre ou les afficher, doit également pouvoir payer sa commande. Pour cela, il lui suffit de se rendre sur la page :

4.2.7 *Payer sa commande, la page payer.php / paie.php*

Lorsqu'un utilisateur souhaite régler sa commande et se la faire livrer, il lui suffit d'aller sur l'onglet payer du menu. Celui-ci appelle alors la page « payer.php »⁸ qui va permettre à l'utilisateur de régler sa commande.

Lorsque l'utilisateur arrive sur cette page, il voit s'afficher sa facture détaillée ainsi qu'un bouton lui permettant de la valider. Lorsqu'il clique sur ce bouton, ses coordonnées sont affichées (c'est la page paie.php qui s'en charge).

Pour ceci, il s'agit simplement d'une requête de type select sur la table vpc_users en sélectionnant seulement l'utilisateur dont l'identifiant est identique à celui de l'utilisateur connecté. Les informations sont donc affichées sous forme d'un tableau afin de rappeler à l'utilisateur quelles données il a enregistré lors de son inscription. Comme il n'est pas évident que l'utilisateur voudra forcément se faire livrer sa commande à l'adresse enregistrée dans la base, il est important de lui donner le choix de modifier son adresse de livraison. Pour cela, j'ai mis en place un formulaire contenant un choix « oui / non » sur la question « voulez vous modifier votre adresse de livraison ». Si l'utilisateur choisi de modifier son adresse en choisissant « oui », alors un formulaire lui permettant de modifier toutes ses informations lui sera affiché. Sinon, un lien image sera affiché, lui permettant de choisir son type de paiement.

Dans ce site, je n'ai réalisé qu'une simulation de paiement donc il n'y a pas d'envoi de mail de confirmation ni de paiement sécurisé réel. Le simple clic sur le lien image permettra de valider la commande en supprimant toutes les lignes correspondantes de la table vpc_panier. La commande sera alors considérée comme terminée et un message sera affiché pour le signifier à l'utilisateur.

Maintenant que toutes les fonctions de base du site sont présentées, il fait présenter les fonctions supplémentaires telles que les fonctions administrateur ou celles permettant à un utilisateur de mettre à jour son compte.

4.2.8 *Le compte utilisateur : la page mon_compte.php*

Lorsqu'un utilisateur se connecte sur son compte, il voit apparaitre un onglet mon compte⁹ dans le menu. Celui-ci lui permet :

- de modifier les informations enregistrées de son compte,
- D'afficher les informations enregistrées relatives à son compte,
- De supprimer son compte.

⁸ Le code correspondant aux pages payer.php et paie.php est disponible en annexe 8.

⁹ Le code correspondant à la page mon_compte.php est disponible en annexe 9.

Modification de compte

Lorsqu'un utilisateur choisi de modifier son compte, il clique sur le lien correspondant dans la page `mon_compte.php` qui le redirige vers une autre section de cette même page. Cette section affiche alors un formulaire contenant toutes les informations relatives au compte de l'utilisateur connecté. Ces informations sont modifiables et il peut donc les mettre à jour à loisir en les modifiant simplement à l'intérieur du formulaire. Lorsque ce formulaire est validé, une requête de type `update` est exécutée sur la table `vpc_users` et les données sont donc modifiées. Bien sur, avant d'être modifiées, les données sont revérifiées afin d'éviter qu'un utilisateur ne puisse entrer des données erronées telles qu'un login nul par exemple.

Lorsque la requête est exécutée avec succès, un message s'affiche pour signaler à l'utilisateur que son compte a bien été modifié comme il l'avait demandé.

S'il veut vérifier les modifications apportées à son compte, l'utilisateur peut se rendre dans la section affichage du compte qui permet d'afficher les données contenues dans la base à propos du compte courant.

Affichage des données d'un compte

Ne requête `select` est exécutée sur la table `vpc_users` et renvoie la ligne correspondant au compte ayant demandé l'affichage de ses informations. Ces informations sont alors formatées sous forme de tableau puis affichées.

Cette page ne fait en réalité rien d'autre que de l'affichage, il n'y a donc pas grand-chose à ajouter dessus.

Enfin, la dernière fonctionnalité que l'utilisateur peut trouver dans la page `mon_compte.php` est la suppression de son compte.

Suppression de compte

Cette page est elle aussi très simple puisqu'elle se contente d'exécuter une requête de type `delete` sur la table `vpc_users`. Cette requête est de type :

Code

```
$requete="select * from vpc_users where idUser='".$$_SESSION['id']."'";
```

Elle permet de supprimer de la table `vpc_users` la ligne correspondant à l'utilisateur courant. C'est-à-dire que son login est supprimé ainsi que toutes les informations inhérentes à son compte. C'est-à-dire que toutes les commandes de son compte son supprimées de la table `vpc_panier` et que les quantités commandées de ces lignes sont rajoutées aux différents produits de la table `vpc_produits`.

Les fonctions accessibles à un simple utilisateur ont maintenant toutes été passées en revue et expliquées, il ne reste donc plus que celles correspondants aux administrateurs à voir.

4.2.9 La page admin.php¹⁰

Cette page est une page qui regroupe des liens vers les différentes fonctions accessibles à un administrateur, tant au niveau affichage, ajout ou suppression de données. Certaines fonctions étant simples, j'ai choisis de ne pas créer de page spécifique pour elles et donc de les développer directement au sein de la page admin.php. Je vais maintenant les expliquer.

La suppression d'une base

La première section accessible aux administrateurs du site est celle permettant de vider les bases de données du site. L'administrateur peut donc vider la base des utilisateurs, la base des produits ou la base des commandes. Pour cela, il suffit simplement d'exécuter une requête de type drop table qui permet de supprimer la table. Une fois cette table supprimée, il ne reste qu'à la recréer afin que les différentes fonctions du site ne soient pas affectées par ces modifications et qu'aucune erreur n'apparaisse.

Exemple :

Si je choisis de vider la table vpc_users, il me faut exécuter le code suivant :

Code

```
//Suppression de la base
$requete="drop table vpc_users";
$base=mysql_connect("localhost", "root", "");
mysql_select_db("galerie_pierre", $base);
mysql_query($requete) or die("Erreur : ".mysql_error());
mysql_close();
//Creation de la base de données
$requete="Create Table vpc_users (idUser int not null auto_increment, nom varchar(50) not null, prenom
varchar(50) not null, adresse varchar(150) not null, codePostal int not null, ville varchar(50) not null, login
varchar(50) not null, motDePasse varchar(50) not null, primary key (idUser));";
$base=mysql_connect("localhost", "root", "");
if(mysql_select_db("galerie_pierre", $base)){
    if($reponse = mysql_query($requete)){
        //Insertion de l'admin
        $requete="insert into vpc_users values('Galerie', 'Pierre', '12 rue Pierre Legrand',
'59000', 'Lille', 'pgalerie', 'tifi');";
        $base=mysql_connect("localhost", "root", "");
        if(mysql_select_db("galerie_pierre", $base)){
            if($requete = mysql_query($requete)){
                echo "Base vidée avec succès !";
            }
        }
    }
}
mysql_close();
```

Qui permet la suppression puis la création de la table vpc_users. Ceci fonctionne exactement de la même manière pour les deux autres tables. Il faut cependant préciser que je n'évoque ici que le côté suppression pure et simple. Il faut bien sûr mettre à jour les tables associées.

A part cela, l'administrateur peut également afficher le contenu des différentes tables du site en utilisant la section « Affichage » du panneau d'administration.

¹⁰ Le code correspondant à la page admin.php est disponible en annexe 10.

Affichage du contenu des bases

Cette section se contente de sélectionner toutes les données d'une table et de les afficher de manière formatée à l'intérieur d'un tableau. La seule particularité est que, lorsque les enregistrements sont en nombre supérieur à 10, l'affichage se pagine. C'est-à-dire que les 10 premiers enregistrements sont affichés et un nombre de page correspondant au nombre d'enregistrement divisé par 10 est créé. Chaque page contiendra alors les 10 enregistrements correspondants à son numéro :

- La page 1 contiendra les 10 premiers,
- La page 2 les 10 suivants,
- Etc.

Un administrateur peut également intervenir sur le contenu des bases sans pour autant les vider. Par exemple il peut :

Modifier un produit

Pour cela, la liste des produits est affichée et chaque produit est sélectionnable au moyen d'une case à cocher de type radio bouton. Lorsque l'utilisateur sélectionne un produit, il voit alors s'afficher un formulaire contenant toutes les données correspondant à ce produit dans la table `vpc_produit`. Il peut alors effectuer les modifications qu'il souhaite sur le produit, elles seront enregistrées dans la base au moyen d'une requête `update`, exactement comme lors de la modification d'un compte utilisateur.

Enfin, un utilisateur peut créer un nouveau produit pour l'ajouter dans la base de données. Pour cela, il lui faut se rendre sur la page :

4.2.10 Ajout_produit.php¹¹

Lorsque l'utilisateur se trouve sur cette page, un formulaire vide s'affiche et il peut y entrer les informations qu'il souhaite concernant le produit. Lorsqu'il validera ce formulaire, le fichier `upload.php`¹² sera appelé. C'est lui qui permettra l'enregistrement du produit dans la base `vpc_produit` et également le téléchargement de l'image associée au produit sur le serveur.

Voici les étapes permettant de charger une image sur un serveur en PHP :

- Lorsque l'image est choisie grâce au champ `input type=file`, elle est enregistrée dans un dossier temporaire sur le serveur
- Ensuite on la copie de ce répertoire vers un répertoire destination non temporaire,
- A ce moment là, elle est uploadée !

Les fonctions principales du site ont maintenant toutes été passées en revue. Cependant, il est bon de préciser une dernière chose : la manière dont les utilisateurs sont vérifiés sur chaque page.

¹¹ Le code correspondant au fichier `ajout_produit.php` est disponible en annexe 11.

¹² Le code correspondant au fichier `upload.php` est disponible en annexe 12.

4.2.11 La vérification des utilisateurs

Lorsqu'un utilisateur arrive sur une des pages de ce site, la première chose qui est faite est de vérifier s'il est connecté et s'il a bien le droit d'être là. En effet, un utilisateur non connecté ne doit avoir accès qu'à la page d'accueil, c'est pour cela que j'ai inclus le morceau de code suivant en début de chaque page :

Code

```
if(!isset($_SESSION['login'])){
    echo "<div class=erreur>Vous devez vous connecter pour accéder à cette page !</div><br><br><a
href=accueil.php>Se connecter</a>";
}
```

Ceci permet d'afficher un message d'erreur si l'utilisateur qui tente d'accéder à la page n'est pas connecté et cela évite que les utilisateurs ne puissent accéder au site sans être connecté.

De plus, je vérifie également le statut de l'utilisateur. C'est-à-dire que, si un utilisateur tente de se connecter à une page administrateur, je vérifie son statut afin de déterminer s'il est administrateur ou non. S'il l'est, alors la page sera affichée correctement, sinon, un message d'erreur sera affiché à la place !

Voici la vérification correspondante :

Code

```
if($_SESSION['login']!="pgalorneau"){
    echo "<div class=erreur>Vous n'avez pas accès à cette section du site</div>";
}
```

4.3 Les erreurs et problèmes rencontrés

Durant toute la durée de mon projet, j'ai eu des choix à faire, des erreurs à corriger et des problèmes à régler. Ces problèmes et erreurs ont pu être de toutes natures, de la simple faute de frappe à l'erreur de logique. Il m'a donc été nécessaire de réfléchir, de chercher comment résoudre mes problèmes en tous genres. Dans cette partie, je vais présenter les principales choses qui m'ont demandé réflexion (en dehors des choix que j'ai présentés plus haut).

4.3.1 Prise en main de MySQL

Pour réaliser ce projet, nous devons utiliser les bases de données MySQL et non plus les fichiers texte. Ceci aura été un challenge intéressant dans le sens où tout était nouveau pour moi là-dedans. En effet, utiliser des fichiers séquentiels pour simuler une base de données n'était pas évident et compliqué à mettre en œuvre, cependant, il n'a pas été beaucoup plus facile de mettre en place ce site au moyen de bases de données puisque nous avons du tout apprendre pour ce site. Ce n'était que le premier TP que nous avons eu à réaliser avec MySQL et il faut donc avouer que j'ai perdu un certain temps en recherches sur internet ou dans les cours fournis. Cependant cela aura été très enrichissant et m'aura bien fait comprendre qu'il n'y a pas photo entre l'utilisation de fichiers séquentiels ou de bases de données. C'est bien plus pratique d'utiliser des vraies bases de données et aussi beaucoup plus propre !

4.3.2 Upload d'un fichier sur un serveur

Depuis déjà quelques temps, je voulais réaliser un script permettant l'upload de fichiers sur un serveur. En effet, déjà au cours du premier semestre j'avais cherché à le faire en VBScript pour le compte du TP sur la DVD Tech. Je n'y étais pas parvenu et m'étais donc dit que je réessaierais plus tard.

A l'abord de ce TP, je me suis donc dit qu'il serait vraiment intéressant que ce script soit réalisé pour celui-ci puisqu'il pourrait permettre de faire un site beaucoup plus réaliste du fait que chaque administrateur pourrait insérer des images sur le site sans avoir besoin de se connecter directement sur le serveur. L'objectif était de permettre aux administrateurs de réaliser ceci à partir de leur poste de travail, de chez eux ou de n'importe où !

Après une journée de recherche et de tests, j'ai enfin compris précisément le fonctionnement d'un tel script et j'ai donc pu le mettre en place au sein de mon site ce qui, comme je le pensais, le rend vraiment plus propre et pratique à utiliser pour tout un chacun.

Conclusion

Au cours de la réalisation de ce projet, j'ai eu à confirmer ce que les cours de Web Dynamique 2 m'avais permis de prendre en main : c'est-à-dire le langage PHP. Grâce à ces connaissances, j'ai pu réaliser un site de vente par correspondance complet. Il m'a également fallu mettre en application les compétences acquises en cours de Bases de Données au moyen de la base de données MySQL.

J'ai donc pu investir les connaissances acquises dans deux matières différentes afin de réaliser un site qui soit le plus professionnel et le mieux réalisé possible. Après ce projet, je commence à apercevoir ce à quoi peut ressembler la programmation d'un site commercial (destiné à être vendu) en PHP. En effet, il m'a fallu ici mettre en place un système de résonnement et une logique proche de celle que j'aurais eu si je devais réaliser ce projet dans le cadre d'un stage ou d'une mission d'entreprise puisque le site à réaliser devait être le plus propre et le plus complet possible.

Après la fin de mon projet, il reste cependant quelques petites améliorations à faire pour que ce site soit totalement apte à être vendu. Notamment, il pourrait être bon de revoir le côté graphique et d'améliorer certains algorithmes. En effet, je ne suis pas très bon en graphisme et je fais donc des graphiques « simples » et carrés qui nuisent peut être un peu au design du site. De plus, certains de mes algorithmes ne sont peut être pas tout à fait optimisés du fait que je n'avais pas vraiment le temps de les revoir plusieurs fois afin de ne garder que le meilleur.

Pour conclure, j'ai réalisé mon grand objectif de ce projet qui était la réalisation d'un site professionnel permettant la gestion d'un magasin de vente en ligne en y incluant un module d'upload de fichiers sur un serveur. En effet, cet aspect de la programmation en PHP n'a pas été abordé en cours et je trouvais important d'être capable de le faire et de comprendre le fonctionnement de tels scripts. C'est pourquoi je m'étais lancé ce défis.

Annexes

Annexe 1 : page d'accueil

Code

```
<?php
session_start();
?>
<html>
<head>
<title>Vente par correspondance</title>
<link rel="stylesheet" type="text/css" href="../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="../";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps">
<center>
<h2>Pierre Jouet Club</h2>
<h3>Articles disponibles</h3>
<?php
//Affichage des produits en vente

if($_SERVER["HTTP_REFERER"] == "http://localhost/WD1/fiche%2017/deco.php"){
    echo "<div class=erreur>Vous avez été déconnecté de votre compte.</div><br>";
}

if(!isset($_SESSION['login'])){
?>
Pour commander ou voir les prix de nos articles, connectez vous ici :
<div style="{border-width:1px; border-color:red; border-style:outset;}">
<table class="table">
<form method=post action=connect.php>
<tr><td>Login :</td><td><input type=text name=login size=10></td></tr>
<tr><td>Mot de passe :</td><td><input type=password name=mdp size=10></td></tr>
<tr><td></td><td><input type=submit value="Connexion"></td></tr>
</form>
</table>
<a href=creer_compte.php>Creer un compte</a></div>
<?php
$requete="select * from vpc_produits;";
$base=mysql_connect("localhost", "root", "");
if(mysql_select_db("galerie_pierre", $base)){
    if($reponse = mysql_query($requete)){
        while($donnees = mysql_fetch_array($reponse)){
            if($donnees['nbDispo']>0){
                ?>
                <div style="{float:left; border-width:1px; border-color:red; border-style:outset;
padding:5px; margin:5px;}">
                <?php echo $donnees['nom'];?>
            }
        }
    }
}
```

```

                <hr style="{color:red;}">
                
                <hr style="{color:red;}">
                <?php echo "<font color=red><b>Quantité en stock :
</b></font>".$donnees['nbDispo'];?>
            </div>
        <?php
        }
        }
    }else{
        echo "Il n'y a aucun produit en vente actuellement.";
    }
}else{
    echo "Echec de la connexion.";
mysql_close();
}
}else{
    include "menu.php";
    echo "<br>Vous êtes connecté sous le pseudonyme <b>".$_SESSION['login']."</b>.<br><br>";
    $requete="select * from vpc_produits;";
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requete)){
            while($donnees = mysql_fetch_array($reponse)){
                if($donnees['nbDispo']>0){
                    ?>
                    <div style="{float:left; border-width:1px; border-color:red; border-
style:outset; padding:5px; margin:5px;}">
                        <?php echo $donnees['nom'];?>
                        <hr style="{color:red;}">
                        <a href="commande.php"></a>
                        <hr style="{color:red;}">
                        <?php echo "<font color=red><b>Quantité en stock :
</b></font>".$donnees['nbDispo'];?><br>
                        <font color=red><b>Prix : </b></font><?php echo $donnees['prix']."€";
?>
                        <hr style="{color:red;}">
                        <form method=post action=commande.php>
                        <font color=red><b>Quantite désirée :</b></font><input type=text
name=qte value="0" size=5">
                        <input type=hidden value=<?php echo $donnees['idProduit']; ?>
name=commande>
                        <br><input type=submit value="Ajouter au panier">
                        </form>
                    </div>
                <?php
                }
                }
            }else{
                echo "Il n'y a aucun produit en vente actuellement.";
            }
        }else{
            echo "Echec de la connexion.";
        }
    }
    mysql_close();
}
?>

```



```
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini !-->
</body>
</html>
```

Annexe 2 : formulaire de création de compte

Code

```
<html>
<head>
<title>Vente par correspondance - Créer un compte</title>
<link rel="stylesheet" type="text/css" href="../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="../";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps" float=center>
<center align=center>
<h2>Pierre Jouet Club</h2>
<h3>Créer un compte</h3>
<?php
if($_SERVER["HTTP_REFERER"] == "http://localhost/WD1/fiche%2017/creer_compte.php"){
    echo "<div class=erreur>Ce login existe déjà !</div><br>";
}
?>

<form method=post action=creer.php>
<table class=table>
<tr><td>Nom : </td><td><input typ=text name=nom>*</td></tr>
<tr><td>Preom : </td><td><input typ=text name=prenom>*</td></tr>
<tr><td>Adresse : </td><td><input typ=text name=adresse>*</td></tr>
<tr><td>Code Postal : </td><td><input typ=text name=cp>*</td></tr>
<tr><td>Ville : </td><td><input typ=text name=Ville>*</td></tr>
<tr><td>Login : </td><td><input typ=text name=login>*</td></tr>
<tr><td>Mot de passe : </td><td><input type=password name=pwd1>*</td></tr>
<tr><td>Confirmation : </td><td><input type=password name=pwd2>*</td></tr>
<tr><td></td><td><input type=submit Value=Enregistrer></td></tr>
</table>
</form>
NB : (*) signifie que le champ correspondant est obligatoire !
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->
</body>
</html>
```

Annexe 3 : création d'un utilisateur : page créer.php

Code

```
<html>
<head>
<title>Vente par correspondance - Créer un compte</title>
<link rel="stylesheet" type="text/css" href="../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="../";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps" float=center>
<center align=center>
<h2>Pierre Jouet Club</h2>
<h3>Créer un compte</h3>
<?php
$nom=$_POST["nom"];
$prenom=$_POST["prenom"];
$adresse=$_POST["adresse"];
$cp=$_POST["cp"];
$ville=$_POST["Ville"];
$login=$_POST["login"];
$pwd1=$_POST["pwd1"];
$pwd2=$_POST["pwd2"];

//Verification de la validité des données saisient
if($nom=="" || !is_string($nom) || $prenom=="" || !is_string($nom) || $adresse=="" || $cp=="" ||
!is_numeric($cp) || $cp<0 || $ville=="" || !is_string($ville) || $login=="" || $pwd1!=$pwd2 ||
strlen($pwd1)<6){
    echo "<div class=erreur align=left>Les données ne sont pas valides : elles doivent etre du type
suivant : <br>
    <ul>
    <li>Nom : chaine de caractere</li>
    <li>Prenom : chaine de caractere</li>
    <li>Adresse : chaine de caractere mixte (chiffres et lettres)</li>
    <li>Code Postal : entier (de 5 chiffres en général)</li>
    <li>Ville : chaine de caractère</li>
    <li>Login : chaine de caractères mixte (chiffres et lettres)</li>
    <li>Mot de passe : chaine de caracteres mixte (chiffres et lettres) de 6 caractères minimum</li>
    </ul>
    </div>";
}else{
    //Insertion dans la base de données users
    $requeteV="select login from vpc_users where login='$login'";
    $requete="insert into vpc_users values('', '$nom', '$prenom', '$adresse', '$cp', '$ville', '$login',
'$pwd1')";
```

```

    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requeteV)){
            if(mysql_num_rows($reponse)>0){
                echo "<div class=erreur>Ce login existe deja !</div><br><br><a
href=creer_compte.php>Retour</a>";
            }else{
                if($reponse = mysql_query($requete)){
                    echo "Votre compte a été créé, connectez vous :";
                    ?>
                    <div style="{border-width:1px; border-color:red; border-
style:outset;}"><form method=post action=connect.php>
                    <table class=table>
                    <tr><td>Login :</td><td><input type=text name=login
size=10></td></tr>
                    <tr><td>Mot de passe :</td><td><input type=password
name=mdp size=10></td></tr>
                    <tr><td></td><td><input
value="Connexion"></td></tr>
                    </table>
                    <a href=creer_compte.php>Créer un compte</a>
                    </form>
                    </div>
                    <?php
                }else{
                    echo "Il y a eu une erreur lors de la création,
réessayez.".mysql_error();
                }
            }
        }else{
            echo "Erreur de connexion au serveur, réessayez plus tard.";
        }
    }
    mysql_close();
}
?>

</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->
</body>
</html>

```

Annexe 4 : page de connexion : connect.php

Code

```
<?php
session_start();
?>
<html>
<head>
<title>Vente par correspondance - Connexion</title>
<link rel="stylesheet" type="text/css" href=" ../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine=" ../";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps" float=center>
<center align=center>
<h2>Pierre Jouet Club</h2>
<h3>Bienvenue</h3>
<?php
$login=$_POST["login"];
$pwd=$_POST["mdp"];

$requete="select * from vpc_users where Login='$login'";
$base=mysql_connect("localhost", "root", "");
if(mysql_select_db("galerie_pierre", $base)){
    if($reponse = mysql_query($requete)){
        if(mysql_num_rows($reponse)==0){
            echo "Ce login n'existe pas !";
        }else{
            $reponse = mysql_fetch_array($reponse);
            if($reponse['motDePasse']==$pwd){
                //echo "Connexion réussie !";
                //Ouverture de session
                $_SESSION['id']=$reponse['idUser'];
                $_SESSION['login']=$login;
                $_SESSION['mail']=$reponse['mail'];
                echo "Vous êtes connecté sous le pseudonyme
<b>".$_SESSION['login'].</b>.<br><br>";
                $requete="select * from vpc_produits";
                $base=mysql_connect("localhost", "root", "");
                if(mysql_select_db("galerie_pierre", $base)){
                    if($reponse = mysql_query($requete)){
                        include("menu.php");
                        if($_SESSION['login']=="pgalerie"){
                            ?>
                            <br><br><a href=admin.php><b>Panneau
d'administration</b></a><br>
                            <?php
                        }
                    }
                }
            }
        }
    }
}
?>
```

```

while($donnees = mysql_fetch_array($reponse)){
    if($donnees['nbDispo']!=0){
        ?>
        <div style="{float:left; border-width:1px; border-
color:red; border-style:outset; padding:5px; margin:5px;}">
        <?php echo $donnees['nom'];?>
        <hr style="{color:red;}">
        <a
            href="commande.php"></a>
        <hr style="{color:red;}">
        <?php echo "<font color=red><b>Quantité en
stock : </b></font>".$donnees['nbDispo'];?><br>
echo $donnees['prix']."€"; ?>
        <font color=red><b>Prix : </b></font><?php
        <hr style="{color:red;}">
        <form method=post action=commande.php>
        <font color=red><b>Quantite désirée
:</b></font><input type=text name=qte value="0" size=5">
        <input type=hidden value=<?php echo
$donnees['idProduit']; ?> name=commande>
        <br><input type=submit value="Ajouter au
panier">
        </form>
        </div>
        <?php
            }
        }else{
            echo "Il n'y a aucun produit en vente actuellement.";
        }
        }else{
            echo "Echec de la connexion.";
        }
        }mysql_close();
    }else{
        echo "Mot de passe erroné !";
    }
}
?>
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->
</body>
</html>

```

Annexe 5 : commander un produit, la page commande.php

Code

```
<?php
session_start();
?>
<html>
<head>
<title>Ajout au panier</title>
<link rel="stylesheet" type="text/css" href=" ../style.css">
</head>
<body bgcolor="#cdcacd">
<center>
<?php
$racine="..";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<center>
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps">
<center>
<?php
if(!isset($_SESSION['login'])){
    echo "<div class=erreur>Vous devez vous connecter pour accéder à cette page !</div><br><br><a
href=accueil.php>Se connecter</a>";
}else{
?>
<h2>Pierre Jouet Club</h2>
<h3>Ajout au panier</h3>
<?php
include("menu.php");
echo "<br><br>";
}
$nbCommande = $_POST["qte"];
$idProduit=$_POST["commande"];
$requete = "select * from vpc_produits where idProduit=$idProduit;";
$base=mysql_connect("localhost", "root", "");
if($nbCommande<=0){
    echo "<div class=erreur>Le nombre d'articles demandé est invalide !</div>";
}else if(mysql_select_db("galerie_pierre", $base)){
    if($reponse = mysql_query($requete)){
        while($donnees = mysql_fetch_array($reponse)){
            if(mysql_num_rows>1){
                echo "<div class=erreur>Erreur de saisie.</div>";
            }else{
                if($donnees['nbDispo']-$nbCommande>-1){
                    //Insertion du produit dans la base panier
                    $requete = "insert into vpc_panier values(',
'$idProduit','.$_SESSION['id'].', '$nbCommande)";
                    $base=mysql_connect("localhost", "root", "");
```


Annexe 6 : Affichage du panier, la page panier.php

Code

```
<?php
session_start();
?>
<html>
<head>
<title>Ajout au panier</title>
<link rel="stylesheet" type="text/css" href=" ../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="..";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<center>
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps">
<center>
<?php
if(!isset($_SESSION['login'])){
    echo "<div class=erreur>Vous devez vous connecter pour accéder à cette page !</div><br><br><a
href=accueil.php>Se connecter</a>";
}else{
    $total=0;
    $soustota=0;
    ?>
    <h2>Pierre Jouet Club</h2>
    <h3>Contenu du panier</h3>
    <?php
    include("menu.php");
    echo "<br><br>";

    $requete = "select * from vpc_panier where idUser='".$_SESSION['id']."'";
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requete)){
            if(mysql_num_rows($reponse)==0){
                echo "Votre panier est vide pour l'instant !";
            }else {
                $facture = "<table border=1 class=table><tr><td><b>Article
:</b></td><td>Prix Unitaire :</td><td>Quantité commandée :</td><td>Prix à payer :</td></tr>";
                while($donnees = mysql_fetch_array($reponse)){
                    $requete1="select * from vpc_produits where
idProduit='".$_SESSION['idProduit']."'";
                    $base=mysql_connect("localhost", "root", "");
                    if(mysql_select_db("galerie_pierre", $base)){
                        if($reponse1 = mysql_query($requete1)){
```

```

while($donnees1=mysql_fetch_array($reponse1)){

    $soustotal=$donnees['quantite']*$donnees1['prix'];

    $facture.="<tr><td>".$donnees1['nom']. "</td><td>".$donnees1['prix']. "€</td><td>".$donnees['qu
antite']. "</td><td>$soustotal €</td></tr>";

    ?>
    <div style="{float:left; border-width:1px;
border-color:red; border-style:outset; padding:5px; margin:5px;}">
    <?php echo $donnees1['nom'];?>
    <hr style="{color:red;}">
    
    <hr style="{color:red;}">
    <?php echo " </font
color=red><b>Quantité commandée : </b></font>".$donnees['quantite'];?>
    <form method=post action=retirer.php>
    <input type=hidden name=idP
value=<?php echo $donnees1['idProduit']; ?>>
    <input type=hidden name=qte
value=<?php echo $donnees['quantite']; ?>>
    <input type=hidden name=idE
value=<?php echo $donnees['idPan']; ?>>
    <hr style="{color:red;}">
    <font color=red><b>Quantité à retirer
:</b></font><input type=text value=0 name=qteR size=5><br>
panier">
    </form>
</div>
<?php

    $total+=$donnees['quantite']*$donnees1['prix'];
    }
}
}??>

<br><br><br><br><br><br><br><br><br><br><br><br>
<br><br>
<br>
<h3 align=center>Facture :</h3>
<?php
$facture.="<tr><td><b>Total :</b><td></td><td></td><td>$total
€</td></tr></table>";
echo $facture;
}
}
mysql_close();
}
?>

</center>
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php

```

```
include("$racine/bas.php");  
?>  
</td></center>  
</table>  
<!--C'est fini !-->  
</body>  
</html>
```

Annexe 7 : retirer un produit du panier, page retirer.php

Code

```
<?php
session_start();
?>
<html>
<head>
<title>Retirer du panier</title>
<link rel="stylesheet" type="text/css" href="../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="..";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<center>
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps">
<center>
<?php
if(!isset($_SESSION['login'])){
    echo "<div class=erreur>Vous devez vous connecter pour accéder à cette page !</div><br><br><a
href=accueil.php>Se connecter</a>";
}else{
?>
<h2>Pierre Jouet Club</h2>
<h3>Ajout au panier</h3>
<?php
include("menu.php");
echo "<br><br>";
}
$idPan = $_POST['idE'];
$idP=$_POST['idP'];
$qteR=$_POST['qteR'];
$qte=$_POST['qte'];
//Rajouter nb emprunté a la base des produits disponibles
$requete1 = "update vpc_produits set nbDispo=nbDispo+$qteR where idProduit=$idP;";
$base=mysql_connect("localhost", "root", "");
if(mysql_select_db("galerie_pierre", $base)){
    if($qteR>0 && $qteR<=$qte){
        if($reponse=mysql_query($requete1)) {
        }else{
            echo mysql_error();
        }
    }
    if($qteR-$qte==0){
        //Suppression de la ligne concernée dans la base panier
        $requete="delete from vpc_panier where idPan='$idPan';";
```


Annexe 8 : règlement de la commande, pages payer.php et paie.php

Payer.php :

Code

```
<?php
session_start();
?>
<html>
<head>
<title>Paie</title>
<link rel="stylesheet" type="text/css" href="../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="..";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<center>
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps">
<center>
<?php
if(!isset($_SESSION['login'])){
    echo "<div class=erreur>Vous devez vous connecter pour accéder à cette page !</div><br><br><a
href=accueil.php>Se connecter</a>";
}else{

    $total=0;
    $facture = "<table border=1 class=table><tr><td><b>Article :</b></td><td>Prix Unitaire
:</td><td>Quantité commandée :</td><td>Prix à payer :</td></tr>";
    ?>
    <h2>Pierre Jouet Club</h2>
    <h3>Régler ma commande</h3>
    <?php
    include("menu.php");
    echo "<br><br>";

    $requete = "select * from vpc_panier where idUser='".$_SESSION['id'].'" ";
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requete)){
            if(mysql_num_rows($reponse)==0){
                echo "Votre panier est vide pour l'instant !";
            }else {
                while($donnees = mysql_fetch_array($reponse)){
                    $requete1="select * from vpc_produits where
idProduit='".$_SESSION['idProduit'].'" ";
                    $base=mysql_connect("localhost", "root", "");
```

```

        if(mysql_select_db("galerieau_pierre", $base)){
            if($reponse1 = mysql_query($requete1)){
                while($donnees1=mysql_fetch_array($reponse1)){

                    $soustotal=$donnees['quantite']*$donnees1['prix'];

                    $facture.="<tr><td>".$donnees1['nom']. "</td><td>".$donnees1['prix']. "€</td><td>".$donnees['qu
antite']. "</td><td>$soustotal €</td></tr>";

                    $total+=$donnees['quantite']*$donnees1['prix'];
                }
            }
        }
    }
}
mysql_close();
}
$facture.="<tr><td><b>Total : </b><td></td><td></td><td>$total €</td></tr></table> ";
?>
<h3 align=center>Facture : </h3>
<?php
echo $facture;
?>

<form method=post action=paie.php>
<input type=submit value=Payer>
</form>

</center>
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->
</body>
</html>

```

Paie.php :

Code

```

<?php
session_start();
?>
<html>
<head>
<title>Paie</title>
<link rel="stylesheet" type="text/css" href=" ../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php

```

```

$racine="..";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<center>
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps">
<center>
<?php
if(!isset($_SESSION['login'])){
    echo "<div class=erreur>Vous devez vous connecter pour accéder à cette page !</div><br><br><a
href=accueil.php>Se connecter</a>";
}else{

    $total=0;
    $facture = "<table border=1 class=table><tr><td><b>Article :</b></td><td>Prix Unitaire
:</td><td>Quantité commandée :</td><td>Prix à payer :</td></tr>";
    ?>
    <h2>Pierre Jouet Club</h2>
    <h3>Régler ma commande</h3>
    <?php
    include("menu.php");
    echo "<br><br>";

    $requete = "select * from vpc_panier where idUser='".$_SESSION['id']."'";
    $base = mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerieau_pierre", $base)){
        if($reponse = mysql_query($requete)){
            if(mysql_num_rows($reponse)==0){
                echo "Votre panier est vide pour l'instant !";
            }else {
                while($donnees = mysql_fetch_array($reponse)){
                    $requete1="select * from vpc_produits where
idProduit='".$_donnees['idProduit']."'";
                    $base = mysql_connect("localhost", "root", "");
                    if(mysql_select_db("galerieau_pierre", $base)){
                        if($reponse1 = mysql_query($requete1)){
                            while($donnees1=mysql_fetch_array($reponse1)){

                                $soustotal=$donnees['quantite']*$donnees1['prix'];

                                $facture.="<tr><td>".$_donnees1['nom'].</td><td>".$_donnees1['prix'].<"€</td><td>".$_donnees['qu
antite'].</td><td>$soustotal €</td></tr>";

                                $total+= $donnees['quantite']*$donnees1['prix'];
                            }
                        }
                    }
                }
            }
        }
    }
    mysql_close();
}
$facture.="<tr><td><b>Total :</b><td></td><td></td><td>$total €</td></tr></table>";
?>

```



```

<h3 align=center>Facture :</h3>
<?php
echo $facture;
?>
<br>

<?php
//Confirmation des informations utilisateur
$requete="select * from vpc_users where idUser='".$$_SESSION['id']. "'";
$base = mysql_connect("localhost", "root", "");
if(mysql_select_db("galerie_pierre", $base)){
    if($reponse = mysql_query($requete)){
        while($donnees = mysql_fetch_array($reponse)){
            echo "Voulez vous que nous envoyions votre commande à l'adresse contenue dans
votre compte ?";

            echo "<br><br><b>Rappel de vos données enregistrées :</b>";
            echo "<table border=1 class=table>";
            ?>
            <tr><td>Votre nom :</td><td><?php echo $donnees['nom']; ?></td></tr>
            <tr><td>Votre prenom :</td><td><?php echo $donnees['prenom']; ?></td></tr>
            <tr><td>Votre Adresse :</td><td><?php echo $donnees['adresse']; ?></td></tr>
            <tr><td>Votre code postal :</td><td><?php echo $donnees['codePostal'];
?></td></tr>

            <tr><td>Votre ville :</td><td><?php echo $donnees['ville']; ?></td></tr>
            </table>
            <?php
        }
    }
}

echo "<br>Si ces données sont correctes, et que vous souhaitez vous faire envoyer la commande en cours de
paiement à cette adresse, choisissez \"Non\" et validez.<br><br>Si vous souhaitez vous faire livrer à une autre
adresse, choisissez \"Oui et validez.\"";

?><br><br>Voulez vous modifier votre adresse de livraison ?
<form method=post action="paie.php?f=1">
Oui <input type=radio name=adr value=oui><br>
Non <input type=radio name=adr value=non><br>
<input type=submit value=Valider>
</form>
<?php
if($_GET["f"]==1){
    $adr=$_POST['adr'];
    if($adr=="oui"){
        echo "<h3 align=center><b>Changement d'adresse :</b></h3>";
        ?>
        <form method=post action="paie.php?f=1">
        <table class=table>
        <tr><td>Nom :</td><td><input type=text></td></tr>
        <tr><td>Preom :</td><td><input type=text></td></tr>
        <tr><td>Adresse :</td><td><input type=text></td></tr>
        <tr><td>Code Postal :</td><td><input type=text></td></tr>
        <tr><td>Ville :</td><td><input type=text></td></tr>
        </table>
        <input type=hidden name=adr value=non>
        <input type=submit value="Envoyer la commande !">
        </form><br>
        <br><br>Changer votre adresse
de livraison pour cette commande ne modifie pas votre compte ! Si vous voulez modifier l'adresse contenue dans
votre compte, allez dans le menu "compte" !

```

```

        <?php
        }else if($adr=="non"){
            ?>
            <h3 align=center> Paiement : </h3>
            Choisissez votre mode de paiement :
            <br><br>
            <a href="paie.php?f=11"></a>
            <?php
            }
        }else if($_GET["f"]==11){
            echo "Vous avez choisi le règlement par Paypal.";
            //Suppression de la base de données
            $requete="delete from vpc_panier where idUser=".$_SESSION['id'].";";
            $base = mysql_connect("localhost", "root", "");
            if(mysql_select_db("galerie_pierre", $base)){
                if($reponse = mysql_query($requete)){
                    echo "<br><br>Votre commande a été payée avec succès, votre panier est
maintenant vide !";
                    echo "<h3>Merci pour votre achat ! Nous espérons que vous reviendrez !</h3>";
                }
            }
        }
        ?>
    </center>
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->
</body>
</html>

```

Annexe 9 : la page mon_compte.php

Code

```
<?php
session_start();
?>
<html>
<head>
<title>Modifier mon compte</title>
<link rel="stylesheet" type="text/css" href="../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="..";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<center>
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps">
<center>
<?php
if(!isset($_SESSION['login'])){
    echo "<div class=erreur>Vous devez vous connecter pour accéder à cette page !</div><br><br><a
href=accueil.php>Se connecter</a>";
}else{

    $total=0;
    $facture = "<table border=1 class=table><tr><td><b>Article :</b></td><td>Prix Unitaire
:</td><td>Quantité commandée :</td><td>Prix à payer :</td></tr>";
    ?>
    <h2>Pierre Jouet Club</h2>
    <h3>Mon compte</h3>
    <?php
    include("menu.php");
    echo "<br><br>";
?>
<div>
<span>
<b>Modification :</b><br>
<hr>
<a href="mon_compte.php?m=m">Modifier mon compte</a>
</span>
<span>
<b>Suppression :</b><br><hr>
<a href="mon_compte.php?m=s">Supprimer mon compte</a>
</span>
<span>
<b>Affichage :</b>
<br><hr>
<a href="mon_compte.php?m=a">Afficher mon compte</a>
</span>
</div>
</span>
```

```

</div>
<?php
$modif=$_GET["m"];
if($modif=="m"){
    //Modification du compte
    $requete="select * from vpc_users where idUser='".$_SESSION['id']."'";
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requete)){
            while($donnees = mysql_fetch_array($reponse)){
                ?><br><br><br><br><br><br>
                <h3 align=center>Modifier mon compte :</h3>
                <table class=table>
                <form method=post action="mon_compte.php?m=mo">
                <tr><td><b>Données du compte :</b></td><td></td></tr>
                <tr><td>Login (pseudo) :</td><td><input type=text name=login
value="<?php echo $donnees['login']; ?>"</td></tr>
                <tr><td>Mot de passe :</td><td><input type=password name=mdp
value="<?php echo $donnees['motDePasse']; ?>"</td></tr>
                <tr><td>Confirmation (mot de passe) :</td><td><input type=password
name=mdp1 value="<?php echo $donnees['motDePasse']; ?>"</td></tr>
                <tr><td><b>Données personnelles :</b></td><td></td></tr>
                <tr><td>Nom :</td><td><input type=text name=nom value="<?php
echo $donnees['nom']; ?>"</td></tr>
                <tr><td>Preom :</td><td><input type=text name=prenom
value="<?php echo $donnees['prenom']; ?>"</td></tr>
                <tr><td>Adresse :</td><td><input type=text name=adresse
value="<?php echo $donnees['adresse']; ?>"</td></tr>
                <tr><td>Code postal :</td><td><input type=text name=cp
value="<?php echo $donnees['codePostal']; ?>"</td></tr>
                <tr><td>Ville :</td><td><input type=text name=ville value="<?php
echo $donnees['ville']; ?>"</td></tr>
                <tr><td></td><td><input type=submit value="Modifier"></td></tr>
                </form>
                </table>

                <?php
                    }
            }
        }
    }else if($modif=="mo"){
        $nom=$_POST["nom"];
        $prenom=$_POST["prenom"];
        $adresse=$_POST["adresse"];
        $cp=$_POST["cp"];
        $ville=$_POST["ville"];
        $login=$_POST["login"];
        $pwd1=$_POST["mdp"];
        $pwd2=$_POST["mdp1"];

        //Verification de la validité des données saisient
        if($nom=="" || !is_string($nom) || $prenom=="" || !is_string($nom) || $adresse=="" || $cp=="" ||
!is_numeric($cp) || $cp<0 || $ville=="" || !is_string($ville) || $login=="" || $pwd1!=$pwd2 ||
strlen($pwd1)<6){
            echo "<br><br><br><br><br><br><div class=erreur align=left>Les données ne sont
pas valides : elles doivent etre du type suivant : <br>
            <ul>
            <li>Nom : chaine de caractere</li>
            <li>Prenom : chaine de caractere</li>
            <li>Adresse : chaine de caractere mixte (chiffres et lettres)</li>

```



```

        while($donnees = mysql_fetch_array($reponse1)){
            $qteR=$donnees['quantite'];
            //Rajouter nb emprunté a la base des produits disponibles
            $requete2 = "update vpc_produits set nbDispo=nbDispo+$qteR
where idProduit='".$donnees['idProduit']."'";
            $base=mysql_connect("localhost", "root", "");
            if(mysql_select_db("galerie_pierre", $base)){
                if($reponse2=mysql_query($requete2)) {
                    $bool=true;
                }else{
                    echo mysql_error();
                }
            }
        }
    }
}
if($bool){
    $requete3 = "delete from vpc_panier where idUser=".$_SESSION['id']."";
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse3=mysql_query($requete3)) {
            if($reponse = mysql_query($requete)){
                echo "Votre compte a bien été supprimé !";
                session_destroy();
            }
        }else{
            echo mysql_error();
        }
    }else{
        echo mysql_error();
    }
}
}
echo "<div class=erreur>Une erreur est survenue lors de la tentative de suppression
de votre compte, réessayez plus tard !</div>";
}
mysql_close();
}
}
?>
</center>
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->

</body>
</html>

```

Annexe 10 : Panneau d'administration, la page admin.php

Code

```
<?php
session_start();
?>
<html>
<head>
<title>Panneau d'administration</title>
<link rel="stylesheet" type="text/css" href="../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="..";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<center>
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps">
<center>
<?php
if($_SESSION['login']!="pgalerneau"){
    echo "<div class=erreur>Vous n'avez pas accès à cette section du site</div>";
}
}else{?>
<h2>Pierre Jouet Club</h2>
<h3>Panneau d'administration</h3>

<?php
include "menu.php";
$choix=$_GET["choix"];
$page=substr($_GET["choix"],9,1);
?>

<div class=conteneur align=left>
<span align=left>
<center><b>Suppression des BDD</b></center>
<br>
<a href="admin.php?choix=supprU">Suppression de la table users (la vider)</a>
<br>
<a href="admin.php?choix=supprP">Suppression de la table produits (la vider)</a>
<br>
</span>
<span>
<center><b>Modification des BDD</b></center>
<br>
<a href="admin.php?choix=modifP">Modification d'un produit</a>
<br><br>
</span>
```

```

<span>
<center><b>Affichage des BDD (contenu)</b></center>
<br>
<a href="admin.php?choix=usersL">Contenu de la base users</a>
<br>
<a href="admin.php?choix=produitL">Contenu de la base produits</a>
<br>
<a href="admin.php?choix=panierL">Contenu de la base Panier (commandes en attente par utilisateur)</a>
<br>
</span>
<span>
<center><b>Création</b></center>
<br>
<a href="ajout-produit.php">Créer un nouveau produit</a>
</span>
</div>
<br><br>
<br><br>
<br><br><br><br>

<?php
if(substr($choix,0, 6)=="usersL"){
    echo "<h3>Liste des utilisateurs</h3>";
    $requete="select * from vpc_users;";
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requete)){
            $bool=true;
            if(mysql_num_rows($reponse)==0){
                echo "<div width=550px class=erreur>Il n'y a aucun utilisateur
enregistré.</div>";

                $bool=false;
            }else if(mysql_num_rows($reponse)>10){
                $nb = mysql_num_rows($reponse)/10;
                echo "Pages : ";
                for($i=1; $i<$nb+1; $i++){
                    ?><a href="admin.php?choix=usersL?p=<?php echo $i;
?>"><?php echo $i; ?></a><?php
                }
            }
            if($bool){
                echo "<table border=1 class=table>";
                echo
                "<tr><td><b>Login</b></td><td><b>Nom</b></td><td><b>Prenom</b></td><td><b>Adresse</b><
</td><td><b>Code Postal</b></td><td><b>Ville</b></td></tr>";
                $cpt=0;
                while($donnees = mysql_fetch_array($reponse)){
                    if(($page==" " || $page==1) && $donnees['idUser']<10){
                        echo
                        "<tr><td>".$donnees['login']. "</td><td>".$donnees['nom']. "</td><td>".$donnees['prenom']. "</td><td>".$
donnees['adresse']. "</td><td>".$donnees['codePostal']. "</td><td>".$donnees['ville']. "</td></tr>";
                        $cpt++;
                    }else if($page>1 && $donnees['idUser']>($page-1)*10-1 &&
$donnees['idUser']<$page*10+1){
                        echo
                        "<tr><td>".$donnees['login']. "</td><td>".$donnees['nom']. "</td><td>".$donnees['prenom']. "</td><td>".$
donnees['adresse']. "</td><td>".$donnees['codePostal']. "</td><td>".$donnees['ville']. "</td></tr>";
                        $cpt++;
                    }
                }
            }
        }
    }
}

```



```

mysql_close();

//Creation de la base de données
$requete="Create Table vpc_users (idUser int not null auto_increment, nom varchar(50) not
null, prenom varchar(50) not null, adresse varchar(150) not null, codePostal int not null, ville varchar(50) not
null, login varchar(50) not null, motDePasse varchar(50) not null, primary key (idUser));";
$base=mysql_connect("localhost", "root", "");
if(mysql_select_db("galerie_pierre", $base)){
    if($reponse = mysql_query($requete)){
        //Insertion de l'admin
        $requete="insert into vpc_users values('Galerie', 'Pierre', '12 rue Pierre
Legrand', '59000', 'Lille', 'pgalerie', 'tifer');";
        $base=mysql_connect("localhost", "root", "");
        if(mysql_select_db("galerie_pierre", $base)){
            if($requete = mysql_query($requete)){
                echo "Base vidée avec succès !";
            }
        }
    }
}
mysql_close();
}else if(substr($choix,0, 8)=="panierL"){
    echo "<h3>Produits commandés par chaque utilisateur</h3>";
    //Affichage des panier des utilisateurs
    $requete = "SELECT vpc_users.login as nomU , vpc_produits.nom as nomP, vpc_panier.quantite as
qte, count( * ) as num
FROM vpc_users, vpc_produits, vpc_panier
WHERE vpc_produits.idProduit = vpc_panier.idProduit
AND vpc_users.idUser = vpc_panier.idUser
GROUP BY vpc_users.nom, vpc_produits.nom;";
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requete)){
            echo
            "
            <table
            border=1
            class=table><tr><td><b>Utilisateur</b></td><td><b>Produit</b></td></tr>";
            while($donnees = mysql_fetch_array($reponse)){
                echo
                "
                <tr><td>".$donnees['nomU']. "</td><td>".$donnees['nomP']. "</td></tr>";
            }
            echo "</table>";
        }
    }
}
}else if(substr($choix,0, 6)=="modifP"){
    echo "<h3>Modifier un produit</h3><h3>Liste des produits</h3>";
    $requete="select * from vpc_produits;";
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)) {
        if($reponse = mysql_query($requete)){
            echo
            "
            <table
            border=1
            class=table><tr><td><b>Libelle</b></td><td><b>Modifier</b></td></tr>";
            echo "<form method=post action=\"admin.php?choix=modifTP\"";
            while($donnees=mysql_fetch_array($reponse)){
                echo "
                <tr><td>".$donnees['nom']. "</td><td><input type=radio name=ch
value=\".$donnees['idProduit']. "</td></tr>";
            }
            echo
            "
            <tr><td></td><td><input
            type=submit
            value=Modifier></td></tr></form></table>";
        }
    }
}
}else if(substr($choix,0, 7)=="modifTP"){

```

```

echo "<h3>Modifier un produit</h3>";
$id=$_POST["ch"];
$requete="select * from vpc_produits where idProduit='$id'";
$base=mysql_connect("localhost", "root", "");
if(mysql_select_db("galerie_pierre", $base)) {
    if($reponse = mysql_query($requete)){
        echo "
        <form method=post enctype=\"multipart/form-data\"
action=\"admin.php?choix=modifSP\">";
        echo "
        <table class=table>";
        while($donnees=mysql_fetch_array($reponse)){
            ?><tr><td>Libelle</td><td><input type=text name=nom value=?php
echo $donnees['nom'];?> </td></tr>
            <tr><td>Nombre de produits disponibles</td><td><input type=text
name=nbDispo value=?php echo $donnees['nbDispo'];?></td></tr>
            <tr><td>Photo</td><td><input type=file name="fichier"></td></tr>
            <tr><td>Prix</td><td><input type=text name=prix value=?php echo
$donnees['prix'];?></td></tr>
            <?php
            }
            echo "
            <input type=hidden value=$id name=id><tr><td></td><td><input
type=submit name=\"upload\" value=Modifier</td></tr></table></form>";
        }
    }
}
}
else if(substr($choix, 0, 7)=="modifSP"){
    echo "<h3>Modification d'un produit</h3>";
    $id=$_POST["id"];
    $produit=$_POST["nom"];
    $nbDispo=$_POST["nbDispo"];
    $image=$_FILES["fichier"]["name"];
    $prix=$_POST["prix"];
    $bool=false;

    if($produit==""){
        ?>
        <div class=erreur>Le nom du produit est invalide.
        <?php
        $bool=true;
    }

    if(!is_numeric($nbDispo) || $nbDispo<0){
        ?>
        <br>Le nombre de produit disponibles doit etre un chiffre supérieur ou égal à 0.
        <?php
        $bool=true;
    }

    if($prix==" || $prix<0){
        ?>
        <br>Le prix doit etre un chiffre supérieur 0.
        <?php
        $bool=true;
    }

    if(!$bool){
        $base=mysql_connect("localhost", "root", "");
        if(mysql_select_db("galerie_pierre", $base)){
            //Enregistrement du nouveau produit dans la table vpc_produit
            $requete="update vpc_produits set nom='$produit', nbDispo='$nbDispo',
image='$image', prix='$prix' where idProduit='$id'";
            //mysql_connect("localhost", "root", "");
            $base=mysql_connect("localhost", "root", "");
            if(mysql_select_db("galerie_pierre", $base)){

```

```

        if($reponse = mysql_query($requete)){
            if( isset($_POST['upload']) ){
                // si formulaire soumis
                $bool=true;
                $content_dir = './img/'; // dossier où sera déplacé le fichier
                $tmp_file = $_FILES['fichier']['tmp_name'];

                if( !is_uploaded_file($tmp_file) ){
                    echo "Le fichier est introuvable";
                    $bool=false;
                    exit("1");
                }
                // on vérifie maintenant l'extension
                $type_file = $_FILES['fichier']['type'];
                if( $bool && !strstr($type_file, 'jpg') && !strstr($type_file,
'jpeg') && !strstr($type_file, 'bmp') && !strstr($type_file, 'gif') ){
                    echo "Le fichier n'est pas une image";
                    $bool=false;
                    exit("2");
                }
                // on copie le fichier dans le dossier de destination
                $name_file = $_FILES['fichier']['name'];
                if($bool && !copy($tmp_file, $content_dir . $name_file) ){
                    echo "Impossible de copier le fichier dans
$content_dir";

                    $bool=false;
                    exit("3");
                }
                if($bool){
                    echo "Le fichier a bien été uploadé";
                    echo "<br>Modification effectuée avec succès !";
                }
            }else{
                echo "Le formulaire n'a pas été soumis !";
            }
        }else{
            echo "<div class=erreur>Erreur lors de la modification des
données : ".mysql_error()."</div>";
        }
    }else{
        echo "Echec de la connexion au serveur.";
    }
    mysql_close();
}else{
    echo mysql_error();
}
}
}

}else if(substr($choix,0, 6)=="supprP"){
    echo "<h3>Suppression de la base produits</h3>";
    //Suppression de la base
    $requete="drop table vpc_produits";
    //$base=mysql_connect("localhost", "root", "");
    $base=mysql_connect("localhost", "root", "");
    mysql_select_db("galerie_pierre", $base);
    mysql_query($requete) or die("Erreur : ".mysql_error());
    mysql_close();

    //Creation de la base de données
    $requete="Create Table vpc_produits (idProduit int not null auto_increment, nom varchar(50) not
null,nbDispo int, image varchar(50), prix float not null, primary key (idProduit));";
}

```

```

    $base=mysql_connect("localhost", "root", "");
    //$base=mysql_connect("localhost", "root", "");
    //Connexion
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requete)){
            echo "Base vidée avec succès !";
        }
    }
mysql_close();
}
}
?>

</center>
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->
</body>
</html>

```

Annexe 11 : ajouter un produit, le fichier ajout_produit.php

Code

```
<?php
session_start();
?>

<html>
<head>
<title>Vente par correspondance - Ajout Produit</title>
<link rel="stylesheet" type="text/css" href=" ../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine=" ../";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps" float=center>
<center align=center>
<h2>Pierre Jouet Club</h2>
<h3>Ajouter un produit</h3>
<?php
include("../menu.php");
?>
<table class=table>
<form method="post" enctype="multipart/form-data" action="upload.php">
<tr><td>Nom du produit :</td><td>
<input enctype="multipart/form-data" type=text name=nom></td></tr>
<tr><td>Nombre de produits disponibles :</td><td>
<input type=text name=nbDisp></td></tr>
<tr><td>Photo :</td><td>
<input type=file name="fichier"></td></tr>
<tr><td>Prix :</td><td><input type=text name=prix></td></tr>
<tr><td></td><td><input type=submit name="upload" value=Enregistrer></td></tr>
</form>
</table>
</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->
</body>
```

</html>

Annexe 12 : upload d'un fichier sur le serveur, le fichier upload.php

Code

```
<?php
session_start();
?>

<html>
<head>
<title>Vente par correspondance - Ajout Produit</title>
<link rel="stylesheet" type="text/css" href=" ../style.css">
</head>
<body bgcolor="#cdcdcd">
<center>
<?php
$racine="../";
include("$racine/bandeau.php");
?>
<table height=75%>
<td class="menu">
<?php
include("$racine/menu.php");
?>
</td>
<td class="corps" float=center>
<center align=center>
<h2>Pierre Jouet Club</h2>
<h3>Ajouter un produit</h3>
<?php
include("../menu.php");
//Verification des données envoyées
$produit=$_POST['nom'];
$nbDisp=$_POST['nbDisp'];
$image=$_POST['fichier'];
$prix=$_POST['prix'];
$bool=false;
if($produit==""){
?>
<div class=erreur>Le nom du produit est invalide.
<?php
$bool=true;
}
if(!is_numeric($nbDisp) || $nbDisp<0){
?>
<br>Le nombre de produit disponibles doit etre un chiffre supérieur ou égal à 0.
<?php
$bool=true;
}
if($prix==" || $prix<0){
?>
<br>Le prix doit etre un chiffre supérieur 0.
<?php
$bool=true;
}
?>
```



```

</div>
<?php
if(!$bool){
    //Verification qu'un produit identique n'existe pas déjà
    $requeteVN = "select count(*) as nb from vpc_produits where nom='$produit';";
    $requeteVI = "select count(*) as nb1 from vpc_produits where
image=".$_FILES['fichier']['name']."";";
    //mysql_connect("localhost", "root", "");
    $base=mysql_connect("localhost", "root", "");
    if(mysql_select_db("galerie_pierre", $base)){
        if($reponse = mysql_query($requeteVN)){
            $nb = mysql_fetch_array($reponse);
            if($nb['nb']==0){
                if($reponse = mysql_query($requeteVI)){
                    $nb1 = mysql_fetch_array($reponse);
                    if($nb1['nb1']==0){
                        //Enregistrement du nouveau produit dans la
table vpc_produit
                        $requete="insert into vpc_produits values(",
'$produit', '$nbDisp',"', $_FILES['fichier']['name'].", '$prix');";
                        //mysql_connect("localhost", "root", "");
                        $base=mysql_connect("localhost", "root", "");
                        if(mysql_select_db("galerie_pierre", $base)){
                            if($reponse = mysql_query($requete)){
                                if( isset($_POST['upload']) ){
                                    // si formulaire soumis
                                }
                                $content_dir = './img/';
                                // dossier où sera déplacé le fichier
                                $tmp_file =
                                if(
                                exit("Le
                                fichier est introuvable");
                                }
                                // on vérifie maintenant
                                l'extension
                                $type_file =
                                if( !strstr($type_file,
'jpg') && !strstr($type_file, 'jpeg') && !strstr($type_file, 'bmp') && !strstr($type_file, 'gif') ){
                                    exit("Le
                                fichier n'est pas une image");
                                }
                                // on copie le fichier
                                dans le dossier de destination
                                $name_file =
                                if( !copy($tmp_file,
$content_dir . $name_file) ){
                                    exit("Impossible de copier le fichier dans $content_dir");
                                }
                                echo "Le fichier a bien
                                été uploadé";
                                }
                                }else{
                                echo "<br>Ajout effectué !";
                                echo "Echec de l'ajout

```

```

! ".mysql_error();
                                        }
server.";                               }else{
                                        echo "Echec de la connexion au
                                        }
                                        mysql_close();
                                        }else{
sur le serveur !";                       echo "Une image portant le meme nom existe deja
                                        }
                                        }
}else{
}
}
}
}
}
}
}

?>

</td>
</table>
<table height="10%"><center>
<td class="signe">
<?php
include("$racine/bas.php");
?>
</td></center>
</table>
<!--C'est fini!-->
</body>
</html>

```