

Duyck Florentin

Six Tristan

Rapport de Projet

Site E-commerce P.M.U

Table des matières

Introduction.....	3	Modifier.....	54
Description du projet.....	3	Supprimer.....	55
Exigence non fonctionnelle.....	4	Produit.....	56
Conception technique.....	4	Article.....	58
Chapitre BDD.....	5	Ajouter.....	58
Chapitre page d'accueil.....	7	Modifier.....	61
Liste des catégories de produits.....	9	Supprimer.....	63
Affiche des meilleurs ventes.....	10	Promotion.....	64
Affichage des quatre dernières promotions....	11	Ajouter.....	64
Chapitre page des produits par catégories.....	12	Modifier.....	66
Chapitre Page des informations d'un produit.....	15	Supprimer.....	67
Chapitre Conseil.....	18	Conseil.....	68
Chapitre Barre de recherche produit.....	22	Ajouter.....	70
Chapitre Portail de Connexion.....	24	Modifier.....	71
Chapitre utilisateur.....	30	Supprimer.....	72
Chapitre Panier d'achat.....	36	Utilisateur.....	73
Panel Administrateur :.....	41	Augmentation de grade.....	74
Accueil.....	49	Commande.....	76
Affichage de la dernière promotion.....	49	Réalisation technique.....	77
Affichage des meilleurs ventes.....	50	Organisation.....	78
Affichage des 5 dernières ventes.....	51	Résultats.....	78
Catégorie.....	52	Améliorations envisagées.....	78
Ajouter.....	53		

Introduction

Il faut moderniser son site web, c'est une problématique pour chaque entreprise qui possède un site web qui a fait son temps. De nouvelles normes et techniques apparaissent rendant les sites plus fluides et esthétiques.

Les sites E-Commerce sont fleurissants de nos jours et banals. Il est donc important de réussir à se démarquer des autres par des fonctionnalités et une esthétique propre. Ces sites sont une fenêtre qui mettent en contact une entreprise avec ses clients afin de partager les nouveaux produits, promotions et nouvelles modalités.

Nous avons donc eu pour projet, de développer un site E-Commerce pour la société P.M.U qui en possédait déjà un mais vieillissant. Ainsi l'objectif est de développer un site web moderne avec une charte graphique et toutes les fonctionnalités que l'on attend d'un site E-Commerce en 2020.

Nous avons suivi les caractéristiques des sites de commerces en ligne telle qu'Amazon, comme une barre de recherche, un espace client, un espace de gestion pour les produits, conseils et commande du site etc ...

Pour réussir ce projet, nous avons tout d'abord listé les tâches à faire et nous nous les sommes répartis, afin d'être efficace compte tenu du temps qui nous était imparti, ce qui nous a permis de développer plus efficacement les différentes fonctionnalités du site. Enfin, nous avons procédé à un rapport sur notre projet.

Description du projet

P.M.U souhaite donc créer un site web afin de pouvoir vendre ses produits de construction.

Ce site permettra à l'entreprise de prendre des commandes, gérer ses clients et ses produits. Il possédera donc un portail de connexion, un affichage de promotion, un espace utilisateur et administrateur, un panier d'achat, un espace de vente par catégorie, une barre de recherche produit etc...

Un client pourra donc prendre des articles dans son panier, commander, et accéder à ses informations dans son espace réservé.

Un administrateur pourra gérer l'ensemble du site, que ce soit les produits, les conseils, les autorisations etc.... depuis un interface spécial accessible depuis le site.

Exigence non fonctionnelle

Exigences de qualité :

Il est important de rendre le site dynamique afin que les clients aient envie de l'explorer et de l'utiliser.

Ergonomie efficace :

Le site doit être simple d'utilisation afin que le client l'utilise de manière naturel sans avoir à réfléchir à comment l'utiliser.

Interface graphique :

Les différentes couleurs et choix typographiques doivent permettre aux clients de repérer les différentes fonctionnalités qui s'offrent à lui, que ce soit dans le panier, dans les catégories de produit et dans son espace réservé.

Conception technique

Le site web est basé sur une architecture MVC. Le pattern MVC permet de bien organiser son code source. Il aide à savoir quels fichiers créer, mais surtout à définir leur rôle. Le but de MVC est justement de séparer la logique du code en trois parties que l'on retrouve dans des fichiers distincts.

Une application conforme au modèle MVC comporte trois types de modules : les modèles, les vues et les contrôleurs :

Modèle : « Élément qui contient les données ainsi que de la logique en rapport avec les données : validation, lecture et enregistrement. Il peut, dans sa forme la plus simple, contenir uniquement une simple valeur, ou une structure de données plus complexe.

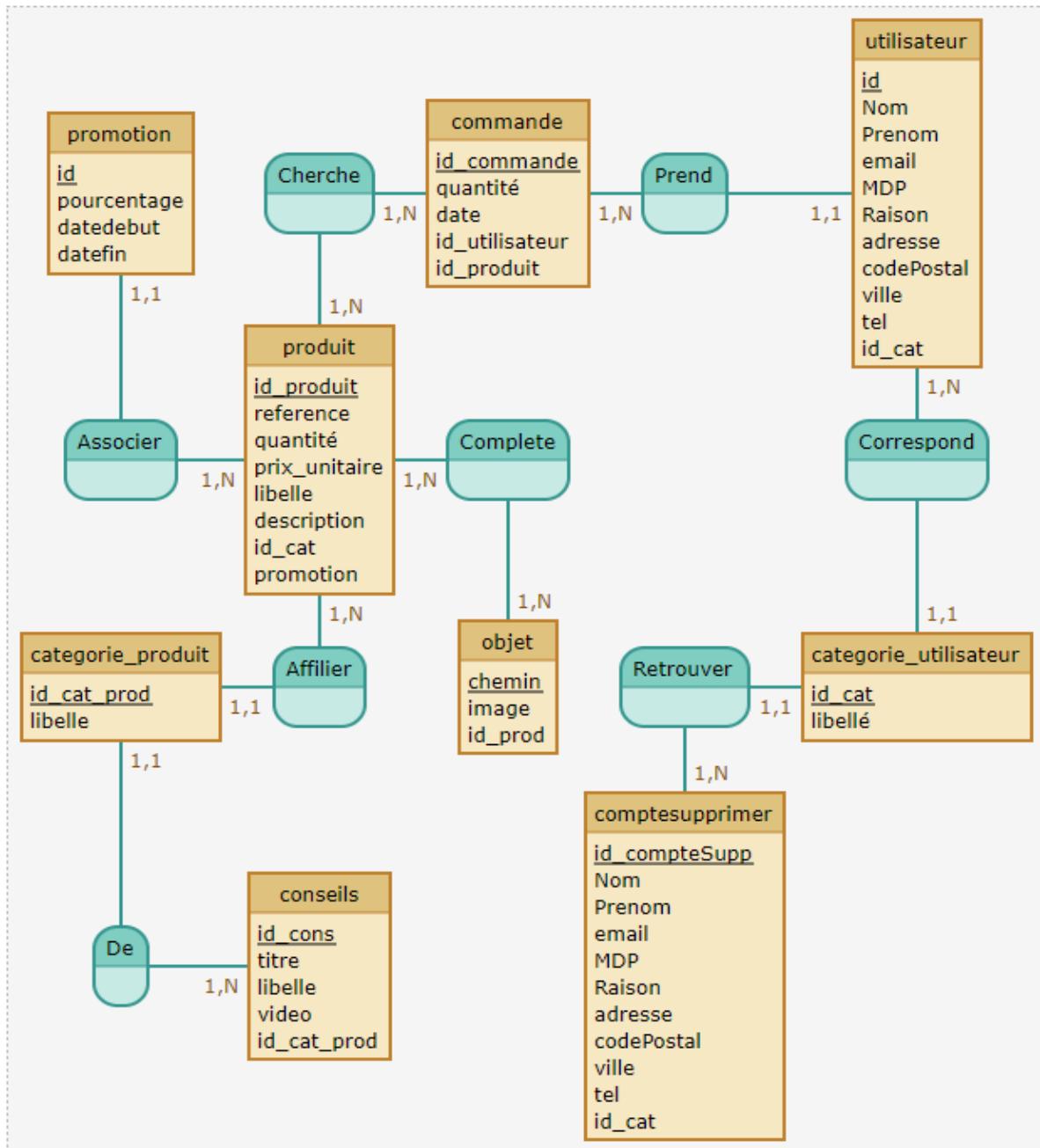
Le modèle représente l'univers dans lequel s'inscrit l'application. Par exemple pour une application de banque, le modèle représente des comptes, des clients, ainsi que les opérations telles que dépôt et retraits, et vérifie que les retraits ne dépassent pas la limite de crédit. »

Vue : « Partie visible d'une interface graphique. La vue se sert du modèle, et peut être un diagramme, un formulaire, des boutons, etc. Une vue contient des éléments visuels ainsi que la logique nécessaire pour afficher les données provenant du modèle. Dans une application de bureau classique, la vue obtient les données nécessaires à la présentation du modèle en posant des questions. Elle peut également mettre à jour le modèle en envoyant des messages appropriés. Dans une application web une vue contient des balises HTML. »

Contrôleur : « Module qui traite les actions de l'utilisateur, modifie les données du modèle et de la vue. »

(Source:Wikipedia)

Chapitre BDD



PROMOTION (`id`, pourcentage, datedebut, datefin, `id_produit`)

CHERCHE (`id_produit`, `id_commande`)

COMMANDE (`id_commande`, quantité, date, `id_utilisateur`, `id_produit`)

UTILISATEUR (`id`, Nom, Prenom, email, MDP, Raison, adresse, codePostal, ville, tel, `id_cat`, `id_commande`)

PRODUIT (`id_produit`, reference, quantité, prix_unitaire, libelle, description, `id_cat`, promotion)

COMPLETE (`id_produit`, `chemin`)

CATEGORIE_PRODUIT (`id_cat_prod`, libelle, `id_produit`, `id_cons`)

OBJET (`chemin`, image, `id_prod`)

CATEGORIE_UTILISATEUR (`id_cat`, libellé, `id`, `id_compteSupp`)

CONSEILS (`id_cons`, titre, libelle, video, `id_cat_prod`)

COMPTESUPPRIMER (`id_compteSupp`, Nom, Prenom, email, MDP, Raison, adresse, codePostal, ville, tel, `id_cat`)

Nous faisons appel à la base de données de deux manières sur le site web :

```
include_once('../public/template/session.php');
include_once('../models/ConnexionBdd.php');
include_once('../models/conseil.php');
include_once('../models/conseilManager.php');
include_once('../models/panier.php');
include_once('../models/ArticleManager.php');

$conseil = [];
$DBase = new Connect();
$db = $DBase->connexion();
```

- L'objet Connect :

```
<?php
class Connect
{
    //connexion a la base de donnees
    protected $db = '';
    private $connec_host = '';
    private $connec_dbname = '';
    private $connec_pseudo = '';
    private $connec_mdp = '';

    public function __construct($connec_host = 'localhost', $connec_dbname = 'projetlicence', $connec_pseudo = 'root', $connec_mdp = ''){
        try {
            $this->db = new PDO('mysql:host='.$connec_host.';dbname='.$connec_dbname, $connec_pseudo, $connec_mdp);
            $this->db->exec("SET CHARACTER SET utf8");
            $this->db->exec("SET NAMES utf8");
            $this->db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_WARNING);
        }
        catch(PDOException $e) {
            die('<h3>Erreur impossible de se connecter à la base de donnée !</h3> <br/> <a href="../index.php">Retour</a>');
        }
    }

    public function connexion(){
        return $this->db;
    }
}
```

- Deuxième méthode :

```
<?php
define("DBHOST", "localhost");
define("DBNAME", "projetlicence");
define("DBUSER", "root");
define("DBPASSWORD", "");
```

```
<?php
require('util.php');
init_php_session();

require_once("../models/config.php");

if(isset($_POST['connexion'])){
    if(isset($_POST['mail']) && !empty(isset($_POST['mail'])) &&
    isset($_POST['mdp']) && !empty(isset($_POST['mdp'])) ){
        $mail = htmlspecialchars($_POST['mail']);
        $mdp = htmlspecialchars($_POST['mdp']);

        try {
            $bddConnexion = new mysqli(DBHOST, DBUSER, DBPASSWORD, DBNAME);
        } catch (Exception $e) {
            $loginErr = "Le service est actuellement indisponible : BDD connexion error";
        }

        $userConnexion = $bddConnexion->prepare("SELECT id, email, Nom, Prenom, Raison, adresse, codepostal, ville, tel, MDP,categorie_utilisateur.libelle FROM utilisateur
JOIN categorie_utilisateur ON utilisateur.id_cat = categorie_utilisateur.id_cat WHERE email = ? AND MDP = ?");
        $userConnexion->bind_param("ss", $mail, $mdp);
        $userConnexion->execute();
        $user = $userConnexion->get_result();
        $userInfo = $user->fetch_assoc();
```

Chapitre page d'accueil

The screenshot shows a web application interface for a catalog. At the top, there's a navigation bar with 'P.M.U' and links for 'Accueil', 'Produit', and 'Conseils'. A search bar contains the placeholder 'Que cherchez vous ?' with a 'Recherche' button. On the right, a 'Panier' icon shows 0 items, and an email 'misterfloflo59@gmail.com' with a power button icon. Below the navigation, there's a sidebar for 'Catégorie:' with options like 'Vis', 'Poulie', 'Roue', and 'Support'. A 'Conseil' section features a video thumbnail for 'GIRAFFE - Mega Bloks - ...'. The main content area displays a 'Promotion' for a 'Pneu Poulie' (Wheel Tire) at a price of 2€ 0.80€ (-30%) with 'Information' and 'Acheter' buttons. To the right, a 'Meilleur vente:' section highlights 'Vis longue'. At the bottom, a footer bar includes links for 'A propos', 'Vie privée', 'Condition d'utilisation', and copyright information: '© 2020 *****, Inc. All rights reserved.'

Les classes d'objet utilisés pour la page d'accueil sont :

- La classe catégorie qui va contenir les catégories de produit :

```
class Categorie
{
    private $_id;
    private $_libelle;

    public function __construct($donnees)
    {
        $this->_id = $donnees['id_cat_prod'];
        $this->_libelle = $donnees['libelle'];
    }

    public function setId($id){
        $this->_id = $id;
    }

    public function setLibelle($libelle){
        $this->_libelle = $libelle;
    }

    public function getId()
    {
        return $this->_id;
    }

    public function getLibelle()
    {
        return $this->_libelle;
    }
}
```

- La classe Produit qui va contenir les produits et leurs promotions :

```
class Article
{
    private $_id ;
    private $_reference = "";
    private $_quantité = 0;
    private $_prix_unitaire = 0;
    private $_libelle = "";
    private $_description = "";
    private $_categorie = "";
    private $_chemin = "";
    private $_nomImage = "";
    private $_id_promotion = "";
    private $_prixPromo;
    private $_pourcentage;
    private $_datefin;
```

```
public function getId()
{
    return $this->_id;
}

public function getReference()
{
    return $this->_reference;
}

public function getQuantite()
{
    return $this->_quantité;
}

public function getPrix()
{
    return $this->_prix_unitaire;
}

public function getLibelle()
{
    return $this->_libelle;
}

public function getDescription()
{
    return $this->_description;
}

public function getCategorie()
{
    return $this->_categorie;
}

public function getImage()
{
    return $this->_chemin;
}

public function getNomImage()
{
    return $this->_nomImage;
}
```

```
public function __construct($donnees)
{
    if (isset($donnees['id_produit'])) {
        $this->_id = $donnees['id_produit'];
    }
    if (isset($donnees['reference'])) {
        $this->_reference = $donnees['reference'];
    }
    if (isset($donnees['quantité'])) {
        $this->_quantité = $donnees['quantité'];
    }
    if (isset($donnees['prix_unitaire'])) {
        $this->_prix_unitaire = $donnees['prix_unitaire'];
    }
    if (isset($donnees['libelle'])) {
        $this->_libelle = $donnees['libelle'];
    }
    if (isset($donnees['description'])) {
        $this->_description = $donnees['description'];
    }
    if (isset($donnees['categorie'])) {
        $this->_categorie = $donnees['categorie'];
    }
    if (isset($donnees['chemin'])) {
        $this->_chemin = $donnees['chemin'];
    }
    if (isset($donnees['image'])) {
        $this->_nomImage = $donnees['image'];
    }
    if (isset($donnees['promotion'])) {
        $this->_id_promotion = $donnees['promotion'];
    }
    if (isset($donnees['prix'])) {
        $this->_prixPromo = $donnees['prix'];
    }
    if (isset($donnees['pourcentage'])) {
        $this->_pourcentage = $donnees['pourcentage'];
    }
    if (isset($donnees['datefin'])) {
        $this->_datefin = $donnees['datefin'];
    }
}
```

```
public function getPromotion()
{
    return $this->_id_promotion;
}

public function getPrixPromo()
{
    return $this->_prixPromo;
}

public function getPourcentage()
{
    return $this->_pourcentage;
}

public function getDateFin()
{
    return $this->_datefin;
}
```

Liste des catégories de produits

On affiche les catégories dans une liste. Quand on clique sur l'une des catégories ça nous amène vers une autre page qui affiche tous les produits de cette catégorie.

Catégorie :

Vis
Poulie
Roue
Support

Le code d'affichage de la liste de catégories.

```
<div class="col">
    <a>Catégorie :</a>
    <div class="list-group">
        <?php
            for ($i=0; $i < $tailleCategories; $i++) {
                echo '<a class="list-group-item list-group-item-action" href="cat_prod.php?categorie='
                    . $categories[$i]->getId().'">' . $categories[$i]->getLibelle() . '</a> ';
            }
        <?>
    </div>
</div>
```

La fonction qui permet de récupérer la liste des catégories.

```
public function listeCategorie()
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $categories = [];

    $q = $db->prepare('SELECT id_cat_prod,libelle from categorie_produit');
    $q->execute();
    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    [
        $categories[] = new Categorie($donnees);
    ]

    return $categories;
}
```

Affiche des meilleures ventes

Ici on va juste afficher les meilleures ventes actuelles.

Meilleur vente :

Vis longue

Le code d'affichage des meilleures ventes.

```
<div class="col">
    <a>Meilleur vente :</a>
    <ul class="list-group">
        <?php
            for ($i=0; $i < $tailleVente; $i++) {
                echo '<li class="list-group-item"> '. $ventes[$i]->getLibelle() . '</li> ';
            }
        ?>
    </ul>
</div>
```

La fonction qui permet de récupérer les 5 premiers produit par ordre décroissant produits ayant fais le plus de commande.

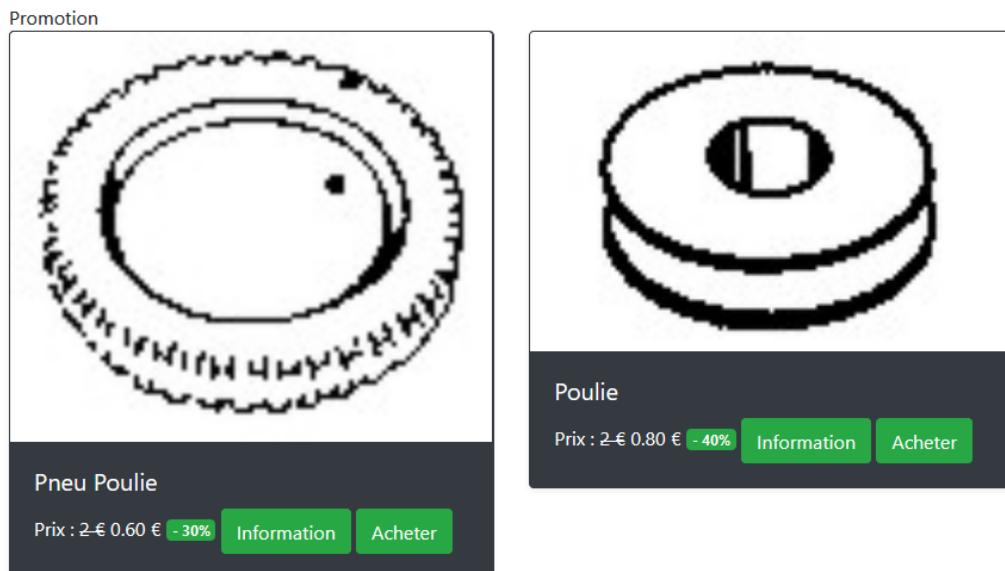
```
public function getMeilleureVente()
{
    $ventes = [];

    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('SELECT `libelle` FROM `produit` JOIN commande ON produit.id_produit=commande.id_produit
GROUP by commande.id_produit ORDER BY COUNT(commande.id_produit) LIMIT 5 ');
    $q->execute();

    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $ventes[] = new Article($donnees);
    }
    return $ventes;
}
```

Affichage des quatre dernières promotions.

On va récupérer les quatre dernières promotions ajoutées pour les afficher sur le site puis on va afficher ces données sous forme de carte avec une image, le nom du produit son ancien prix et le nouveau, le pourcentage de la promotion et deux boutons, un pour avoir plus d'informations et un autre pour acheter le produit.



Le code d'affichage des quatre dernières promotions.

```
<div class="">
    <a>Promotion</a>
    <div class = "row">
        <?php
            $y = 0;
            for ($i=0; $i < $tailleArticlePromo; $i++) {
        ?>
        <div class="col-6">
            <div class="card border border-dark shadow-sm bg-dark text-white rounded">
                <img src=<?php echo "".$articlePromo[$i]->getImage().""> alt="<?php echo $articlePromo[$i]->getNomImage()?>">
                <div class="card-body">
                    <div class="row">
                        <div class="col">
                            <h5 class="card-title"><?php echo $articlePromo[$i]->getLibelle() ?></h5>
                            <span class="card-link">Prix : <del><?php echo $articlePromo[$i]->getPrix() ?></del> <?php echo $articlePromo[$i]->getPrixPromo() ?></span>
                            <span class="badge badge-success">- <?php echo $articlePromo[$i]->getPourcentage() ?>%</span>
                            <a href="pageProduit.php?id=<?php echo $articlePromo[$i]->getId()?>"><button class="card-link btn btn-success" name="info" id="info">Information</button></a>
                            <a class="card-link btn btn-success ajouterPanier" href="/projetphp/controllers/ajouteAuPanier.php?id=<?php echo $articlePromo[$i]->getId()?>" name="acheter" id="acheter">Acheter</a>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    <?php
        }
    ?>
</div>
```

La fonction qui permet de récupérer les quatre dernières promotions encore actives de la plus récente à la plus vieille.

```
public function lastPromotion()
{
    $promotions = [];
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('SELECT id_produit,libelle,chemin,image,description,prix_unitaire,quantité,ROUND(prix_unitaire*(pourcentage/100),2)as prix,promotion,pourcentage,datefin
    from produit join objet on objet.id_prod = id_produit
    join promotion on promotion.id = produit.promotion
    where CURDATE() between datedebut and datefin ORDER by promotion.datedebut DESC LIMIT 4 ');
    $q->execute();
    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $promotions[] = new Article($donnees);
    }
    return $promotions;
}
```

Chapitre page des produits par catégories

On va afficher les produit par catégorie si on a sélectionner une catégorie dans le liste mais si on a cliqué sur produit dans la barre de navigation c'est tous les produits qui s'affiche ajouter on va l'afficher sous forme de carte avec une image, le nom du produit son ancien prix , le nouveaux et le pourcentage de la promotion si le produit est en promotion et deux bouton un pour avoir plus d'information et un pour acheter le produit.

The screenshot shows a web application interface. At the top, there's a header with the logo "P.M.U", navigation links "Accueil", "Produit", and "Conseils", a search bar "Que cherchez vous?", a "Recherche" button, a "Panier" link with a cart icon and "0" items, and a user email "misterfloflo59@gmail.com". Below the header is a sidebar titled "Catégorie:" with four options: "Vis", "Poulie", "Roue", and "Support". To the right, there's a grid of three product cards. The first card shows an image of a pulley wheel, its name "Pneu Poulie", its price "Prix : 2€ 0.60 € -30%", and two buttons "Information" and "Acheter". The second card shows an image of a long screw, its name "Vis longue", its price "Prix : 9 €", and two buttons "Information" and "Acheter". The third card shows an image of another pulley wheel, its name "Poulie", its price "Prix : 2€ 0.80 € -40%", and two buttons "Information" and "Acheter". At the bottom of the page, there's a footer with links "A propos", "Vie privée", "Condition d'utilisation", and a copyright notice "© 2020 ***** Inc. All rights reserved."

La classe d'objet utilisée pour la page d'accueil est : La classe Produit qui va contenir les produits et leurs promotions :

```
class Article
{
    private $_id ;
    private $_reference = "";
    private $_quantité = 0;
    private $_prix_unitaire = 0;
    private $_libelle = "";
    private $_description = "";
    private $_categorie = "";
    private $_chemin = "";
    private $_nomImage = "";
    private $_id_promotion = "";
    private $_prixPromo;
    private $_pourcentage;
    private $_datefin;
```

```

public function __construct($donnees)
{
    if (isset($donnees['id_produit'])) {
        $this->_id = $donnees['id_produit'];
    }
    if (isset($donnees['reference'])) {
        $this->_reference = $donnees['reference'];
    }
    if (isset($donnees['quantité'])) {
        $this->_quantité = $donnees['quantité'];
    }
    if (isset($donnees['prix_unitaire'])) {
        $this->_prix_unitaire = $donnees['prix_unitaire'];
    }
    if (isset($donnees['libelle'])) {
        $this->_libelle = $donnees['libelle'];
    }
    if (isset($donnees['description'])) {
        $this->_description = $donnees['description'];
    }
    if (isset($donnees['categorie'])) {
        $this->_categorie = $donnees['categorie'];
    }
    if (isset($donnees['chemin'])) {
        $this->_chemin = $donnees['chemin'];
    }
    if (isset($donnees['image'])) {
        $this->_nomImage = $donnees['image'];
    }
    if (isset($donnees['promotion'])) {
        $this->_id_promotion = $donnees['promotion'];
    }
    if (isset($donnees['prix'])) {
        $this->_prixPromo = $donnees['prix'];
    }
    if (isset($donnees['pourcentage'])) {
        $this->_pourcentage = $donnees['pourcentage'];
    }
    if (isset($donnees['datefin'])) {
        $this->_datefin = $donnees['datefin'];
    }
}

```

```

public function getId()
{
    return $this->_id;
}

public function getReference()
{
    return $this->_reference;
}

public function getQuantite()
{
    return $this->_quantité;
}

public function getPrix()
{
    return $this->_prix_unitaire;
}

public function getLibelle()
{
    return $this->_libelle;
}

public function getDescription()
{
    return $this->_description;
}

public function getCategorie()
{
    return $this->_categorie;
}

public function getImage()
{
    return $this->_chemin;
}

public function getNomImage()
{
    return $this->_nomImage;
}

```

```

public function getPromotion()
{
    return $this->_id_promotion;
}

public function getPrixPromo()
{
    return $this->_prixPromo;
}

public function getPourcentage()
{
    return $this->_pourcentage;
}

public function getDateFin()
{
    return $this->_datefin;
}

```

Le code d'affichage des produits est :

```
<div class="col">
    <div class="row">
        <?php
            $y = 0;
            for ($i=0; $i < $tailleArticle; $i++) {
        ?>

        <div class="col-4">
            <div class="card border border-dark shadow-sm bg-dark text-white rounded">
                <img src=<?php echo '$article[$i]->getImage()' .> alt=<?php echo $article[$i]->getNomImage() .>>
                <div class="card-body">
                    <div class="row">
                        <div class="col">
                            <h5 class="card-title"><?php echo $article[$i]->getLibelle() .></h5>
                            <?php
                                if($article[$i]->getPromotion() != NULL){
                                    if($article[$i]->getId() == $articlePromo[$y]->getId()){
                                        <span class="card-link">Prix : <del><?php echo $articlePromo[$y]->getPrix() .></del> <?php echo $articlePromo[$y]->getPrixPromo() .></span>
                                        <span class="badge badge-success">- <?php echo $articlePromo[$y]->getPourcentage() .>%</span>
                                    <?php
                                        $y++;
                                    }
                                    else{
                                        <span class="card-link">Prix : <?php echo $article[$i]->getPrix() .></span>
                                    <?php
                                }
                                else{
                                    <span class="card-link">Prix : <?php echo $article[$i]->getPrix() .></span>
                                <?php
                            }
                            <a href="pageProduit.php?id=<?php echo $article[$i]->getId() .>"><button class="card-link btn btn-success" name="info" id="info">Information</button></a>
                            <a class="card-link btn btn-success ajouterPanier" href="#"><?php echo $article[$i]->getId() .><name="acheter" id="acheter">Acheter</a>
                        </div>
                    </div>
                </div>
            </div>
        <?php
            }
        ?>
    </div>

```

Utilisation de deux tableaux, un pour les articles normaux et un autre avec les articles en promotions.

On va vérifier si le produit à un code promotion. Si il y en a un on va chercher le produit dans la table des promotions.

La fonction permet de récupérer la liste des produits selon la catégorie choisie.

```
//recupere la liste des categories
public function getList($categorie)
{
    $articles = [];
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('SELECT id_produit,produit.libelle,quantité,description,prix_unitaire,chemin,image,promotion,(categorie_produit.libelle) as categorie
from objet
join (produit join categorie_produit on categorie_produit.id_cat_prod = id_cat) on produit.id_produit = id_prod where id_cat LIKE "' . $categorie . '"');
    $q->execute();
    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $articles[] = new Article($donnees);
    }

    return $articles;
}
```

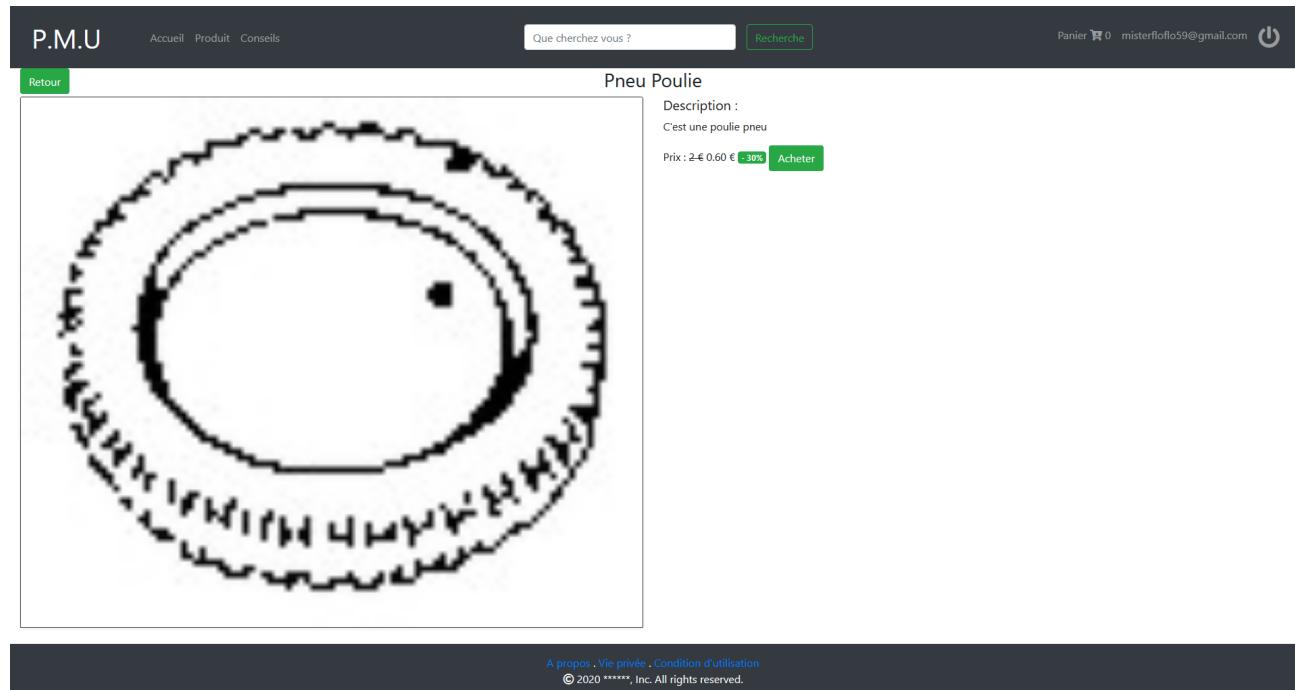
La fonction permet de récupérer la liste des produits en promotion qui sont actifs selon la catégorie choisie.

```
//Recupere la liste des informations de la promotions pour les articles de la même catégorie
public function getPromotionCategorie($categorie)
{
    $promotions = [];
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('SELECT id_produit,produit.libelle,chemin,image,prix_unitaire,ROUND(prix_unitaire*(pourcentage/100),2)as prix,promotion,pourcentage,datefin from produit
join categorie_produit on categorie_produit.id_cat_prod = id_cat
join objet on objet.id_prod = id_produit
join promotion on promotion.id = produit.promotion where CURDATE() between datedebut and datefin and categorie_produit.id_cat_prod LIKE "' . $categorie . '"');
    $q->execute();

    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $promotions[] = new Article($donnees);
    }
    return $promotions;
}
```

Chapitre Page des informations d'un produit

Ici on va afficher les informations d'un produit qui a été sélectionné par l'utilisateur. On va afficher le nom du produit, une image du produit, sa description, son ancien et nouveau prix et pourcentage si le produit est en promotion et un bouton pour pouvoir acheter le produit.



La classe d'objet utilisée pour la page d'accueil est : La classe Produit qui va contenir les produits et leurs promotions :

```
class Article
{
    private $_id ;
    private $_reference = "";
    private $_quantité = 0;
    private $_prix_unitaire = 0;
    private $_libelle = "";
    private $_description = "";
    private $_categorie = "";
    private $_chemin = "";
    private $_nomImage = "";
    private $_id_promotion = "";
    private $_prixPromo;
    private $_pourcentage;
    private $_datefin;
```

```

public function __construct($donnees)
{
    if (isset($donnees['id_produit'])) {
        $this->_id = $donnees['id_produit'];
    }
    if (isset($donnees['reference'])) {
        $this->_reference = $donnees['reference'];
    }
    if (isset($donnees['quantité'])) {
        $this->_quantité = $donnees['quantité'];
    }
    if (isset($donnees['prix_unitaire'])) {
        $this->_prix_unitaire = $donnees['prix_unitaire'];
    }
    if (isset($donnees['libelle'])) {
        $this->_libelle = $donnees['libelle'];
    }
    if (isset($donnees['description'])) {
        $this->_description = $donnees['description'];
    }
    if (isset($donnees['categorie'])) {
        $this->_categorie = $donnees['categorie'];
    }
    if (isset($donnees['chemin'])) {
        $this->_chemin = $donnees['chemin'];
    }
    if (isset($donnees['image'])) {
        $this->_nomImage = $donnees['image'];
    }
    if (isset($donnees['promotion'])) {
        $this->_id_promotion = $donnees['promotion'];
    }
    if (isset($donnees['prix'])) {
        $this->_prixPromo = $donnees['prix'];
    }
    if (isset($donnees['pourcentage'])) {
        $this->_pourcentage = $donnees['pourcentage'];
    }
    if (isset($donnees['datefin'])) {
        $this->_datefin = $donnees['datefin'];
    }
}

```

```

public function getId()
{
    return $this->_id;
}

public function getReference()
{
    return $this->_reference;
}

public function getQuantite()
{
    return $this->_quantité;
}

public function getPrix()
{
    return $this->_prix_unitaire;
}

public function getLibelle()
{
    return $this->_libelle;
}

public function getDescription()
{
    return $this->_description;
}

public function getCategorie()
{
    return $this->_categorie;
}

public function getImage()
{
    return $this->_chemin;
}

public function getNomImage()
{
    return $this->_nomImage;
}

```

```

public function getPromotion()
{
    return $this->_id_promotion;
}

public function getPrixPromo()
{
    return $this->_prixPromo;
}

public function getPourcentage()
{
    return $this->_pourcentage;
}

public function getDateFin()
{
    return $this->_datefin;
}
}

```

Le code d'affichage du produits :

On crée deux tableaux un pour le produit et un autre pour le produit qui est en promotion puis on vérifie si le tableau de promotion est vide. Si il n'est pas vide on utilisera ce tableau.

```
<main>
  <div class="container-fluid">
    <div class="row">
      <div class="col-2">
        <a href="javascript:history.back()" class="btn btn-success">Retour</a>
      </div>
      <div class="col-8 text-center">
        <h3><?php echo $article[0]->getLibelle() ?></h3>
      </div>
    </div>
    <div class="row">
      <div class="col-6">
        <img src=<?php echo $article[0]->getImage() ?> class="border border-dark shadow-sm card-img-top" alt="<?php echo $article[0]->getNomImage() ?>">
      </div>
      <div class="col-5">
        <h5>Description :</h5>
        <p><?php echo $article[0]->getDescription() ?> </p>
        <?php if($article[0]->getPromotion() == NULL){ ?>
          <span class="card-link">Prix : <?php echo $article[0]->getPrix() ?> €</span>
        <?php } ?>
        <?php } ?>
        <span class="card-link">Prix : <del><?php echo $articlePromo[0]->getPrix() ?> €</del> <?php echo $articlePromo[0]->getPrixPromo() ?> €</span>
        <span class="badge badge-success">- <?php echo $articlePromo[0]->getPourcentage() ?>%</span>
      </div>
      <a class="card-link btn btn-success ajouterPanier" href="/projetphp/controllers/ajouteAuPanier.php?id=<?php echo $article[0]->getId() ?>" name="acheter" id="acheter">Acheter</a>
    </div>
  </div>
</main>
```

Cette fonction permet de récupérer le produit a partir de l'id.

```
//recupere les infos d'un article
public function getInfo($id)
{
  $informations = [];
  $DBase = new Connect();
  $db = $DBase->connexion();
  $q = $db->prepare('SELECT id_produit,libelle,chemin,image,description,prix_unitaire,quantité,promotion FROM produit JOIN objet ON objet.id_prod = id_produit WHERE id_produit = "'.$id.'" ');
  $q->execute();
  while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
  {
    $informations[] = new Article($donnees);
  }
  return $informations;
}
```

Cette fonction permet de récupérer le produit en promotion à partir de l'id.

```
//recupere les informations de la promotion en cours sur un article
public function getPromotionArticle($article)
{
  $promotions = [];
  $DBase = new Connect();
  $db = $DBase->connexion();
  $q = $db->prepare("SELECT id_produit,libelle,chemin,image,description,prix_unitaire,quantité,ROUND(prix_unitaire*(pourcentage/100),2)AS prix,promotion,pourcentage,datefin
  FROM produit
  JOIN objet ON objet.id_prod = id_produit
  JOIN promotion ON promotion.id = produit.promotion
  WHERE CURDATE() BETWEEN datedebut AND datefin AND id_produit = '".$article."' ");
  $q->execute();

  while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
  {
    $promotions[] = new Article($donnees);
  }
  return $promotions;
}
```

Chapitre Conseil

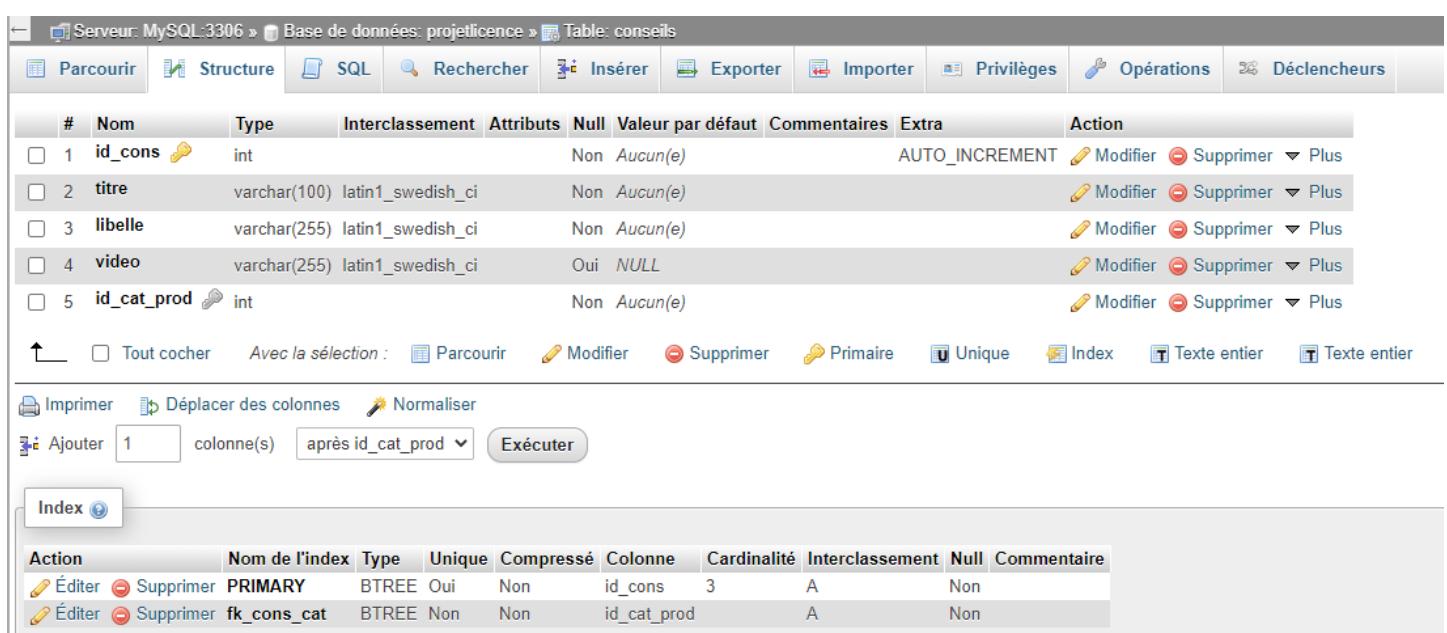
Voici ce qui apparaît lorsque l'on clique sur la rubrique conseil.



The screenshot shows a website interface for 'Bricolage' (DIY). At the top, there's a navigation bar with 'P.M.U', 'Accueil', 'Produit', 'Conseils', a search bar, and a 'Recherche' button. Below the header, a large green banner features the word 'Bricolage' and a video thumbnail showing a person using a power drill on wood. A play button is overlaid on the video image. To the right of the video are links for 'À regarder ...' and 'Partager'. Below the banner, the video title 'Comment visser une vis' is displayed, followed by a brief description: 'Avant de procéder au vissage, appliquez de la paraffine ou de la bougie sur le filet de la vis pour faciliter sa pénétration. Vissez progressivement : adaptez la vitesse de vissage au fut et à mesure. Regardez ce tuto pour visser comme un pro'. There's also a small logo for 'minute facile.com'.

Une vidéo explicative tirée d'un site de vidéos, ainsi qu'un titre et un paragraphe contenant des conseils si les images ne suffisent pas.

Les données d'une vidéo de conseil sont stockées dans la base de données, dans la table conseils et possède la structure suivante : (Pour des infos complémentaires consulter le MCD ou la bibliothèque de base de données fournies en annexe)



The screenshot shows the MySQL Workbench interface for the 'conseils' table. The table structure is as follows:

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id_cons	int			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	titre	varchar(100)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
3	libelle	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	video	varchar(255)	latin1_swedish_ci		Oui	NULL			Modifier Supprimer Plus
5	id_cat_prod	int			Non	Aucun(e)			Modifier Supprimer Plus

Below the table, there are buttons for 'Tout cocher' (Select all), 'Avec la sélection' (With selection), and various actions like 'Parcourir', 'SQL', 'Rechercher', 'Insérer', 'Exporter', 'Importer', 'Priviléges', 'Opérations', and 'Déclencheurs'. There are also links for 'Imprimer', 'Déplacer des colonnes', and 'Normaliser'. At the bottom, there's a section for adding a new index with fields 'Ajouter', '1', 'colonne(s)', 'après id_cat_prod', and 'Exécuter'. The 'Index' tab is selected, showing the current indexes:

Action	Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
Éditer Supprimer	PRIMARY	BTREE	Oui	Non	id_cons	3	A	Non	
Éditer Supprimer	fk_cons_cat	BTREE	Non	Non	id_cat_prod		A	Non	

La page est codée de la manière suivante :

```
<?php
include_once('../public/template/session.php');
include_once('../models/ConnexionBdd.php');
include_once('../models/conseil.php');
include_once('../models/conseilManager.php');
include_once('../models/panier.php');
include_once('../models/ArticleManager.php');

$conseil = [];
$DBase = new Connect();
$db = $DBase->connexion();

$Con_Man = new conseilManager();
$conseils = $Con_Man->listeConseils();
$tailleConseils = sizeof($conseils);

$panier = new Panier();
$article = [];
$Art_man = new articleManager();

?>
```

On importe d'abord les conseils dans un tableau et on les affiche grâce au code PHP ci-dessous :

```
<html lang="fr">
<?php include_once('../public/template/head.php'); ?>

<body>
    <?php include_once('../public/template/header.php'); ?>

    <main>
        <div class="container">
            <?php
                for ($i=0; $i < $tailleConseils; $i++) {
            ?>
            <div class="row mt-3">
                <div class="col">
                    <div class="media">
                        <div class="embed-responsive embed-responsive-16by9">
                            <iframe class="embed-responsive-item" src="<?php echo $conseils[$i]->getVideo()?>" frameborder="0"
                                allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture" allowfullscreen></iframe>
                        </div>
                        <h5 class="mt-0"><?php echo $conseils[$i]->getTitre() ?></h5>
                        <p><?php echo $conseils[$i]->getLibelle() ?></p>
                    </div>
                </div>
            </div>
        </?php
            }
        ?>
    </main>

    <?php include_once('../public/template/footer.php'); ?>
</body>
</html>
```

Voyons comment est construit notre classe conseils :

```
<?php

class Conseil
{
    private $_id;
    private $_titre;
    private $_libelle;
    private $_video;

    public function __construct($donnees)
    {
        $this->_id = $donnees['id_cons'];
        $this->_titre = $donnees['titre'];
        $this->_libelle = $donnees['libelle'];
        $this->_video = $donnees['video'];
    }

    public function setId($id){
        $this->_id = $id;
    }

    public function setTitre($titre){
        $this->_titre = $titre;
    }

    public function setLibelle($libelle){
        $this->_libelle = $libelle;
    }

    public function setVideo($video){
        $this->_video = $video;
    }

    public function getId()
    {
        return $this->_id;
    }

    public function getLibelle()
    {
        return $this->_libelle;
    }
}
```

```
public function getTitre()
{
    return $this->_titre;
}

public function getVideo()
{
    return $this->_video;
}
```

Pour gérer mes objets conseils, nous avons créé le fichier Conseils Manager :

```
<?php
include_once('conseil.php');

class conseilManager
{
    //recupere la liste des conseils XX
    public function listeConseils() {
        $DBase = new Connect();
        $db = $DBase->connexion();
        $listConseils = [];

        $q = $db->prepare('SELECT * from conseils');
        $q->execute();
        while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
        {
            $listConseils[] = new Conseil($donnees);
        }

        return $listConseils;
    }

    public function addConseil($titre,$libelle,$video){
        $DBase = new Connect();
        $db = $DBase->connexion();
        $q = $db->prepare("INSERT INTO conseils(titre, libelle, video) VALUES ('".$titre."','".$libelle."','".$video."')");
        $q->execute();
    }

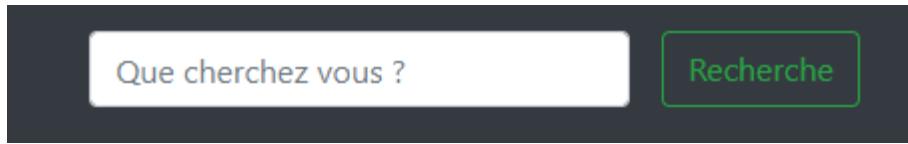
    public function deleteConseil($id){
        $DBase = new Connect();
        $db = $DBase->connexion();
        $q = $db->prepare("DELETE FROM conseils WHERE id_cons = '".$id."'");
        $q->execute();
    }

    public function modifConseil($id,$titre,$libelle,$video){
        $DBase = new Connect();
        $db = $DBase->connexion();
        $q = $db->prepare("UPDATE conseils SET titre='".$titre."',libelle='".$libelle."',video ='".$video."' WHERE id_cons = '".$id."'");
        $q->execute();
    }
}

?>
```

Chapitre Barre de recherche produit

La barre de recherche produit se présente de la manière suivante :



Avec le code suivant :

```
<ul class="navbar-nav mr-auto">
  <li class="nav-item">
    <div class="pt-2 pb-2 bg-dark text-white">
      <form action="../controllers/rechercheProduit.php" method="POST" class="form-inline my-2 my-lg-0">
        <input class="form-control mr-sm-3" id="searchBar" name="searchBar" size="30" type="search" placeholder="Que cherchez vous ?" aria-label="Search">
        <button class="btn btn-outline-success my-2 my-sm-0" type="submit" name="recherche" id="recherche">Recherche</button>
      </form>
    </div>
  </li>
</ul>
```

Il propose une suggestion de produit grâce au Framework jQuery qui se manifeste de la manière suivante quand on écrit dans la barre de recherche :



Voici le code Javascript utilisé pour rendre les recherches dynamiques :

```
<script>
$(document).ready(function() {
  $("#searchBar").keyup(function() {
    var query = $("#searchBar").val();
    $.ajax({
      url: '../models/suggestionProduit.php',
      method: 'POST',
      data: {query: query},
      dataType : 'JSON',
      success: function(data) {
        console.log(data);
        $( "#searchBar" ).autocomplete({
          source: data
        });
      },
    });
  });
</script>
```

Voici le code PHP pour faire les requêtes du côté serveur :

```
<?php
    include_once('../models/ConnexionBdd.php');

    if(isset($_POST['query']) == ""){
        $libelle = "";
    } else {
        $libelle = htmlspecialchars($_POST['query']);
    }

    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare("SELECT libelle FROM `produit` WHERE libelle like '%".$libelle."%'");
    $q->execute();
    $listeNomProduit = array();

    while ($donnees = $q->fetch(PDO::FETCH_ASSOC)) {
        array_push($listeNomProduit, $donnees['libelle']);
    }
    $listeNomProduit = json_encode($listeNomProduit);
    print_r($listeNomProduit);
    return $listeNomProduit;
?>
```

Avec celui-ci nous récupérons les noms des produits en liste puis une fois que l'on a cliqué dessus il suffit de cliquer sur le bouton recherche pour activer le code suivant :

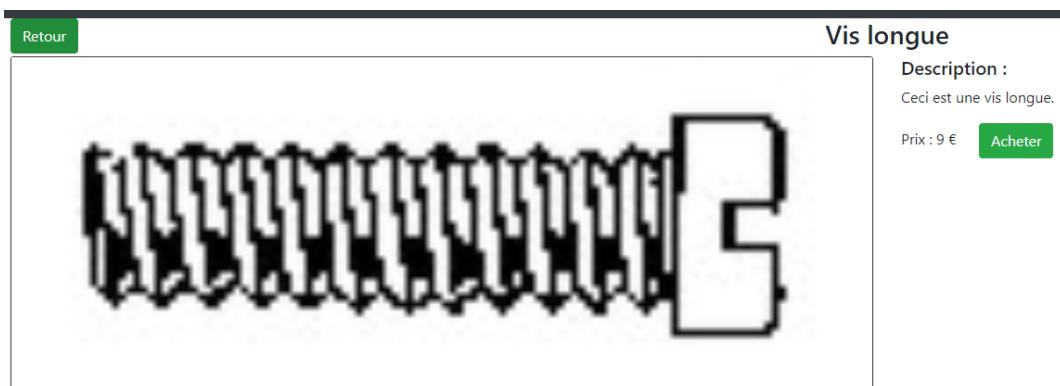
```
<?php
    include_once('../models/ConnexionBdd.php');

    if(isset($_POST['recherche'])){
        if(isset($_POST['searchBar']) && !empty(isset($_POST['searchBar']))){
            $recherche = htmlspecialchars($_POST['searchBar']);

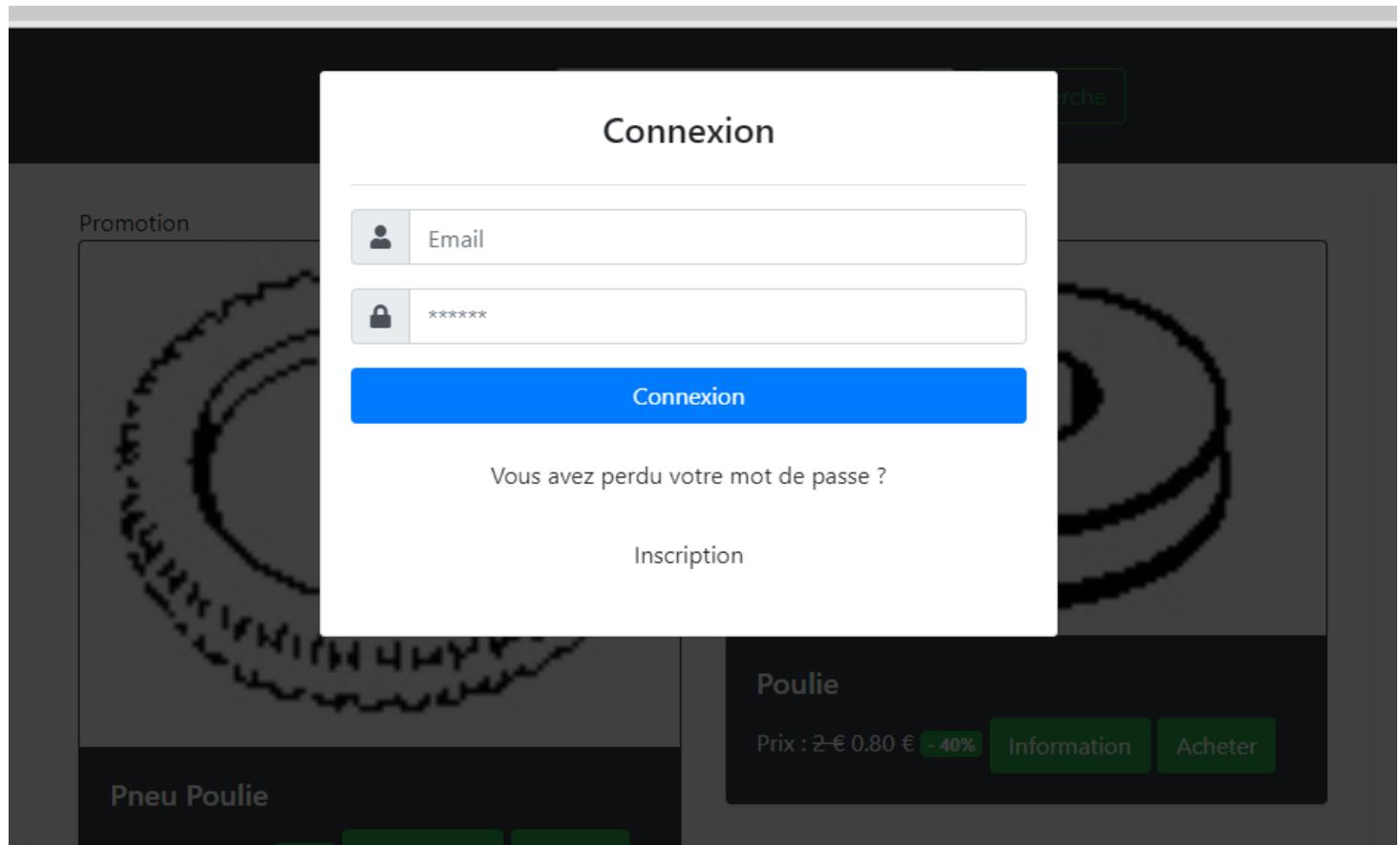
            $DBase = new Connect();
            $db = $DBase->connexion();
            $q = $db->prepare("SELECT id_produit FROM `produit` WHERE libelle like '%".$recherche."%' LIMIT 1");
            $q->execute();
            $produitInfo = $q->fetch(PDO::FETCH_ASSOC);

            header("location:/projetphp/vue/pageProduit.php?id=".$produitInfo['id_produit']);
        }
    }
?>
```

Qui nous redirige sur la page du produit qui se présente comme ceci et sera expliqué dans un autre chapitre :



Chapitre Portail de Connexion



L'affichage de notre portail de connexion s'active par un modal qui est un élément animé du Framework Bootstrap et qui s'active de cette manière :

```
<?php
if(isset($_SESSION['email'])){
    echo('<a class="nav-link" href="../vue/menuUtilisateur.php">');
    echo $_SESSION['email'];
    echo('</a>');
}

} else {
    echo('<a class="nav-link" data-toggle="modal" data-target="#exampleModal">Connexion</a>');
}
?>
</li>
</ul>

if(isset($_SESSION['email'])){
    echo('<li class="nav-item">');
    echo('<a class="nav-link" id="deconnexion" name="deconnexion" href="../public/template/deconnexion.php"><span class="fa fa-power-off fa-2x"></span></a>');
    echo('</li>');
}
?>
</ul>

<!-- connexion --&gt;
&lt;?php
if(isset($_SESSION['email'])){
} else {
    include_once('../public/template/connexion.php');
}
?&gt;</pre>
```

Selon si vous êtes connectés ou non vous ne verrez pas la même chose et cela est géré avec mon code PHP au-dessus. Lorsque vous cliquez sur « Connexion » le portail de connexion apparaît et si vous regardez du côté des classes, il y a un élément qui s'appelle data-target. C'est lui qui va appeler la div viser par un ID qui est ici #exempleModal et qui vise le code suivant :

```
<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="card">
        <article class="card-body">
          <h4 class="card-title text-center mb-4 mt-1">Connexion</h4>
          <hr>
          <form method="POST">
            <div class="form-group">
              <div class="input-group">
                <div class="input-group-prepend">
                  <span class="input-group-text"> <i class="fa fa-user"></i></span>
                </div>
                <input name="mail" id="mail" class="form-control" placeholder="Email" type="email">
              </div>
            </div>
            <div class="form-group">
              <div class="input-group">
                <div class="input-group-prepend">
                  <span class="input-group-text"> <i class="fa fa-lock"></i> </span>
                </div>
                <input class="form-control" name="mdp" id="mdp" placeholder="*****" type="password">
              </div>
            </div>
            <div class="form-group">
              <button type="submit" name="connexion" id="connexion" class="btn btn-primary btn-block">Connexion</button>
            </div>
            <?php
              if (isset($loginErr)) {
                echo('<div class="text-center alert alert-danger">');
                echo $loginErr ;
                echo('</div>');
              }
            ?>
              <p class="text-center"><a href="#" class="btn">Vous avez perdu votre mot de passe ?</a></p>
              <p class="text-center"><a href="../vue/vueInscription.php" class="btn">Inscription</a></p>
            </form>
          </article>
        </div>
      </div>
    </div>
  </div>
```

Cette section une fois appelée nous permettra de nous connecter en remplissant les inputs. Une alerte en cas d'erreur d'identifiant a été mise en place comme vous pouvez le constater à la fin et qui s'affiche comme ceci :

The screenshot shows a mobile application interface. At the top, the word "Connexion" is centered above two input fields. The first field has a user icon and is labeled "Email". The second field has a lock icon and is labeled "*****". Below these fields is a large blue button with the text "Connexion" in white. Underneath the button, a pink rectangular box contains the text "Identifiant ou mot de passe incorrect" in white. At the bottom left, there is a link "Vous avez perdu votre mot de passe ?" and at the bottom right, there is a link "Inscription".

Les inscriptions fonctionnent. Il suffit de cliquer dessus pour arriver sur la page suivante :

The screenshot shows a website header with "P.M.U" and navigation links for "Accueil", "Produit", "Conseils", a search bar, and a shopping cart icon. Below the header, a green "Retour" button is visible. The main content area is titled "Inscription" and contains several input fields: "Email", "Mot de passe", "Nom" (last name), "Prénom" (first name), "Raison social" (business reason), "Adresse", "Ville" (city), "Code postal" (zip code), and "N° de téléphone" (phone number). A green "Inscription" button is located at the bottom right of the form.

Une fois les inputs remplies, vous pouvez cliquer sur « inscriptions ». En cas de non complétiion de l'inscription, vous êtes éjectés sur la page principale. Ensuite le code PHP gère l'inscription :

```
<?php
    session_start();
    include_once('../models/ConnexionBdd.php');

    if(isset($_POST['inscription'])){
        if(empty(isset($_POST['email']))){
            $mailErr = "Il faut un mail pour s'inscrire";
        } else {
            $email = htmlspecialchars($_POST['email']);
        }
        if(empty(isset($_POST['nom']))){
            $nomErr = "Il faut un nom pour s'inscrire";
        } else {
            $nom = htmlspecialchars($_POST['nom']);
        }
        if(empty(isset($_POST['prenom']))){
            $prenomErr = "Il faut un prénom pour s'inscrire";
        } else {
            $prenom = htmlspecialchars($_POST['prenom']);
        }
        if(empty(isset($_POST['raisonSocial']))){
            $raisonSocial = "null";
        } else {
            $raisonSocial = htmlspecialchars($_POST['raisonSocial']);
        }
        if(empty(isset($_POST['adresse']))){
            $adresseErr = "Il faut une adresse pour s'inscrire";
        } else {
            $adresse = htmlspecialchars($_POST['adresse']);
        }
        if(empty(isset($_POST['ville']))){
            $villeErr = "Il faut une ville pour s'inscrire";
        } else {
            $ville = htmlspecialchars($_POST['ville']);
        }
        if(empty(isset($_POST['codePostal']))){
            $codeErr = "Il faut un code postal pour s'inscrire";
        } else [
            $codePostal = htmlspecialchars($_POST['codePostal']);
        ]
        if(empty(isset($_POST['ntel']))){
            $ntel = 123456789;
        } else {
            $ntel = htmlspecialchars($_POST['ntel']);
        }
        if(empty(isset($_POST['mdp']))){
            $mdpErr = "Il faut un mot de passe pour s'inscrire";
        }
    }

    if(empty(isset($_POST['ntel']))){
        $ntel = 123456789;
    } else {
        $ntel = htmlspecialchars($_POST['ntel']);
    }

    if(empty(isset($_POST['mdp']))){
        $mdpErr = "Il faut un mot de passe pour s'inscrire";
    } else {
        $mdp = htmlspecialchars($_POST['mdp']);
    }

    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare("INSERT INTO utilisateur SET `Nom`= '" . $nom . "', `Prenom`= '" . $prenom . "', `email`= '" . $email . "', `MDP`= '" . $mdp . "', `Raison`= '" . $raisonSocial . "', `adresse` ='" . $adresse . "', `ville` ='" . $ville . "', `codepostal` ='" . $codePostal . "', `tel`= '" . $ntel . "', id_cat = 2");
    $q->execute();

    header("location:/projetphp/vue/menuUtilisateur.php");
}

?>
```

Une fois inscrit, vous pouvez vous connecter. Voici la table de la base de données pour les utilisateurs :

The screenshot shows the MySQL Workbench interface with the 'utilisateurs' table selected. The table has 11 columns:

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id	int			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	Nom	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
3	Prenom	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
4	email	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
5	MDP	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
6	Raison	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
7	adresse	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
8	codepostal	int			Non	Aucun(e)			Modifier Supprimer Plus
9	ville	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
10	tel	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus
11	id_cat	int			Non	Aucun(e)			Modifier Supprimer Plus

Below the table, there are buttons for printing, moving columns, normalizing, adding a column after 'id_cat', executing a query, and navigating between tabs like Parcourir, Structure, SQL, Rechercher, Insérer, Exporter, Importer, Priviléges, Opérations, and Déclencheurs.

The 'Index' tab is selected, showing two indexes:

Action	Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
Éditer Supprimer	PRIMARY	BTREE	Oui	Non	id	3	A	Non	
Éditer Supprimer	fk_cat_util	BTREE	Non	Non	id_cat		A	Non	

Il y a une clé étrangère vers la table catégorie utilisateur qui gère les droits utilisateurs et dont la structure est la suivante :

The screenshot shows the MySQL Workbench interface with the 'utilisateur' table selected. The table has 2 columns:

#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
1	id_cat	int			Non	Aucun(e)		AUTO_INCREMENT	Modifier Supprimer Plus
2	libellé	varchar(255)	latin1_swedish_ci		Non	Aucun(e)			Modifier Supprimer Plus

Below the table, there are buttons for printing, moving columns, normalizing, adding a column after 'libellé', executing a query, and navigating between tabs like Parcourir, Structure, SQL, Rechercher, Insérer, Exporter, Importer, Priviléges, Opérations, and Déclencheurs.

The 'Index' tab is selected, showing one index:

Action	Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null	Commentaire
Éditer Supprimer	PRIMARY	BTREE	Oui	Non	id_cat	3	A	Non	

Voici les différents grades utilisateurs :

+ Options

	← T →	id_cat	libellé
<input type="checkbox"/>	Éditer Copier Supprimer	2	Utilisateur
<input type="checkbox"/>	Éditer Copier Supprimer	3	Modérateur
<input type="checkbox"/>	Éditer Copier Supprimer	1	Administrateur

Voici le code PHP qui gère la connexion :

```
<?php

function init_php_session() : bool{
    if(!session_id()){
        session_start();
        session_regenerate_id();
        return true;
    } else {
        return false;
    }
}

function clean_php_session() : void {
    session_unset();
    session_destroy();
}

function is_logged(): bool {
    if(isset($_SESSION['email'])) {
        return true;
    } else {
        return false;
    }
}

function is_admin(): bool {
    return true;
}
```

```
<?php
require('util.php');
init_php_session();

require_once("../models/config.php");

if(isset($_POST['connexion'])){
    if(isset($_POST['mail']) && !empty(isset($_POST['mail'])) &&
    isset($_POST['mdp']) && !empty(isset($_POST['mdp'])) ){
        $mail = htmlspecialchars($_POST['mail']);
        $mdp = htmlspecialchars($_POST['mdp']);

        try {
            $bddConnexion = new mysqli(DBHOST, DBUSER, DBPASSWORD, DBNAME);
        } catch (Exception $e) {
            $loginErr = "Le service est actuellement indisponible : BDD connexion error";
        }

        $userConnexion = $bddConnexion->prepare("SELECT id, email, Nom, Prenom, Raison, adresse, codepostal, ville, tel, MDP,categorie_utilisateur.libellé FROM utilisateur
JOIN categorie_utilisateur ON utilisateur.id_cat = categorie_utilisateur.id_cat WHERE email = ? AND MDP = ?");
        $userConnexion->bind_param("ss", $mail, $mdp);
        $userConnexion->execute();
        $user = $userConnexion->get_result();
        $userInfo = $user->fetch_assoc();

        if ($userInfo['email'] != null) {
            $_SESSION['id'] = $userInfo['id'];
            $_SESSION['email'] = $userInfo['email'];
            $_SESSION['nom'] = $userInfo['Nom'];
            $_SESSION['prenom'] = $userInfo['Prenom'];
            $_SESSION['raison'] = $userInfo['Raison'];
            $_SESSION['adresse'] = $userInfo['adresse'];
            $_SESSION['codePostal'] = $userInfo['codepostal'];
            $_SESSION['ville'] = $userInfo['ville'];
            $_SESSION['tel'] = $userInfo['tel'];
            $_SESSION['mdp'] = $userInfo['MDP'];
            $_SESSION['libellé'] = $userInfo['libellé'];
        }
        else {
            $loginErr = "Identifiant ou mot de passe incorrect";
        }
    }
}
?>
```

Il notifie une erreur au client si il se trompe à la fin, et comme vous pouvez le constater nous n'utilisons pas de classe objet utilisateur car les supers globales de PHP sont amplement suffisantes pour stocker le profil et les données importantes du client en toute sécurité.

Une fois connecté, le site se modifie comme ceci :



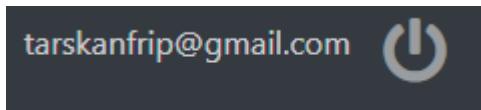
L'adresse mail du client apparaît et un bouton power permet de se déconnecter voici le code :

```
c:\>template> deconnection.php
<?php
session_start(); //On s'assure que tu récupère la bonne session
$_SESSION = array();
session_destroy(); //Paf la session meurt
unset($_SESSION);
header("location:/projetphp/vue/index.php"); //une fois déconnecter on retourne à l'accueil merci bonsoir
?>
```

Notre code commenté parle de lui-même. On s'assure que l'on vise la bonne session puis on détruit pour sortir le client du circuit utilisateur et il retourne à la page d'accueil

Chapitre utilisateur

Lorsque l'utilisateur est connecté, il peut aller consulter ses informations en cliquant sur son adresse mail affichée de la manière suivante :



Une fois ceci fait, vous allez arriver dans la page de gestion utilisateur qui se présente comme ceci :

The screenshot shows a user profile page with the following sections:

- Historique d'achat**: A list of recent purchases (3 days ago). Each item has a heading and a short description.
- Données personnelle du clients**: Personal information fields:
 - Adresse Mail : tarskanfrip@gmail.com
 - Nom complet : Tristan Six
 - Raison social : lolololol
 - Adresse: 7 rue Henri dunant
 - Ville : WATTIGNIES 59139
 - N° de téléphone: 674494580
- Contact**: Help links:
 - Besoin d'aide pour votre compte ?
 - Besoin d'aide pour votre commande ?
 - Une réclamation ?

Comme vous pouvez le voir toutes les informations personnelles du client apparaissent, ainsi que ses achats et des options de contacte en cas de soucis.

Voici le code de la page :

```
<?php
    include_once('../public/template/session.php');
    include_once('../models/ConnexionBdd.php');
    include_once('../models/panier.php');
    include_once('../models/ArticleManager.php');

    $panier = new Panier();
    $article = [];
    $Art_man = new ArticleManager();

    if(empty($_SESSION['email'])){
        header("location:/projetphp/vue/index.php");
    }
?>

<html lang="fr">
    <?php include_once('../public/template/head.php'); ?>

    <body>
        <?php include_once('../public/template/header.php'); ?>

        <main>
            <div class="container mt-5">
                <div class="row">
                    <div class="col-6">
                        <h3>Historique d'achat</h3>
                        <div class="list-group">
                            <a href="#" class="list-group-item list-group-item-action">
                                <div class="d-flex w-100 justify-content-between">
                                    <h5 class="mb-1">List group item heading</h5>
                                    <small>3 days ago</small>
                                </div>
                                <p class="mb-1">Donec id elit non mi porta gravida at eget metus. Maecenas sed diam eget risus varius blandit.</p>
                                <small class="text-muted">Donec id elit non mi porta.</small>
                            </a>
                            <a href="#" class="list-group-item list-group-item-action">
                                <div class="d-flex w-100 justify-content-between">
                                    <h5 class="mb-1">List group item heading</h5>
                                    <small class="text-muted">3 days ago</small>
                                </div>
                                <p class="mb-1">Donec id elit non mi porta gravida at eget metus. Maecenas sed diam eget risus varius blandit.</p>
                                <small class="text-muted">Donec id elit non mi porta.</small>
                            </a>
                        </div>
                    </div>
                    <div class="col-6">
                        <h3>Données personnelle du clients</h3>
                        <ul class="list-group">
                            <li class="list-group-item">Adresse Mail : <?php echo $_SESSION['email']; ?></li>
                            <li class="list-group-item">Nom complet : <?php echo $_SESSION['prenom'] . " " . $_SESSION['nom']; ?></li>
                            <li class="list-group-item">Raison social : <?php echo $_SESSION['raison']; ?></li>
                            <li class="list-group-item">Adresse: <?php echo $_SESSION['adresse']; ?></li>
                            <li class="list-group-item">Ville : <?php echo $_SESSION['ville'] . " " . $_SESSION['codePostal']; ?></li>
                            <li class="list-group-item">N° de telephone: <?php echo $_SESSION['tel']; ?></li>
                        </ul>
                        <div class="text-center">
                            <button onclick="window.location.href='../vue/modifierProfil.php' type="button" class="btn btn-warning mt-4">Modifier les informations</button>
                            <button data-toggle="modal" data-target="#exampleModal" type="button" class="btn btn-danger mt-4 ml-5">Supprimer son compte</button>
                            <?php
                                if($_SESSION['libellé'] == "Administrateur"){
                            ?>
                                <button onclick="window.location.href='../vue/dashboard.php' type="button" class="btn btn-info mt-4">Panel Administrateur</button>
                            <?php
                                }
                            ?>
                        </div>
                    </div>
                </div>
                <div class="row mt-2">
                    <div class="col-6">
                        <h3>Contact</h3>
                        <a>Besoin d'aide pour votre compte ?</a><br>
                        <a>Besoin d'aide pour votre commande ?</a><br>
                        <a>Une réclamation ?</a>
                    </div>
                </div>
            </div>
        </body>
    </html>
```

```

<div class="modal fade" id="exampleModal" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="card">
        <article class="card-body text-center">
          <h4 class="card-title text-center mb-4 mt-1">Êtes vous sur ?</h4>
          <hr>
          <form method="POST">
            <div class="form-group">
              <button onclick="window.location.href='../../controllers/suppProfil.php'" type="button" name="destruction" id="destruction" class="btn btn-danger">Supprimer</button>
            </div>
            <p class="text-center">Cette opération est irréversible !</p>
          </form>
        </article>
      </div>
    </div>
  </div>
</main>

<?php include_once('../public/template/footer.php'); ?>

</body>
</html>

```

Lorsque vous voulez supprimer votre compte, vous n'avez qu'à cliquer sur le bouton supprimer qui fera apparaître un modal, vous demandant de confirmer votre action comme ceci :

Supprimer son compte

Êtes vous sur ?

Supprimer son compte

Cette opération est irréversible !

Une fois que vous cliquez dessus votre compte est supprimé et sauvegardé dans une table spéciale compteSupprimer. Le code suivant s'exécute :

```

<?php
include_once('../models/ConnexionBdd.php');
session_start();

$DBSauvegarde = new Connect();
$db = $DBSauvegarde->connexion();
$q = $db->prepare("INSERT INTO comptesupprimer (`id`, `Nom`, `Prenom`, `email`, `MDP`, `Raison`, `adresse`, `codePostal`, `ville`, `tel`, `id_cat`) VALUES
(" . $_SESSION['id'] . ", '" . $_SESSION['nom'] . "', '" . $_SESSION['prenom'] . "', '" . $_SESSION['email'] . "', '" . $_SESSION['mdp'] . "', '" . $_SESSION['raison'] . "', '" . $_SESSION['adresse'] . "', '" . $_SESSION['codePostal'] . "', '" . $_SESSION['ville'] . "', '" . $_SESSION['tel'] . ", 1)");
$q->execute();

$DBase = new Connect();
$db = $DBase->connexion();
$q = $db->prepare("DELETE FROM `utilisateur` WHERE `id` = " . $_SESSION['id']);
$q->execute();

$_SESSION = array();
session_destroy(); //Paf la session meurt
unset($_SESSION);
header("location:/projetphp/vue/index.php"); //une fois deconnecter on retourne a l'accueil merci bonsoir
?>

```

Le compte est sauvegardé à la fin de l'opération, voici la structure de la table qui est en tout point similaire à la table utilisateur :

	#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action
<input type="checkbox"/>	1	id_compteSupp	int			Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	2	Nom	varchar(50)	latin1_swedish_ci		Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	3	Prenom	varchar(50)	latin1_swedish_ci		Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	4	email	varchar(100)	latin1_swedish_ci		Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	5	MDP	varchar(100)	latin1_swedish_ci		Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	6	Raison	varchar(100)	latin1_swedish_ci		Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	7	adresse	varchar(100)	latin1_swedish_ci		Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	8	codePostal	int			Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	9	ville	varchar(100)	latin1_swedish_ci		Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	10	tel	int			Non	Aucun(e)		Modifier Supprimer Plus	
<input type="checkbox"/>	11	id_cat	int			Non	Aucun(e)		Modifier Supprimer Plus	

Passons à la seconde option, la modification de données :

Modifier les informations

Une fois que vous avez cliqué sur le bouton vous êtes dirigé sur la page suivante :

[Retour](#)

Données personnelle du clients

Email

Mot de passe

Nom <input type="text" value="Six"/>	Prénom <input type="text" value="Tristan"/>
Raison social <input type="text" value="lolololol"/>	
Adresse <input type="text" value="7 rue Henri dunant"/>	
Ville <input type="text" value="WATTIGNIES"/>	Code postal <input type="text" value="59139"/>
N° de téléphone <input type="text" value="674494580"/>	

Modifier les informations

Dont le code est le suivant :

```
<?php
    include_once('../public/template/session.php');
    include_once('../models/ConnexionBdd.php');
    include_once('../models/panier.php');
    include_once('../models/ArticleManager.php');

    $panier = new Panier();
    $article = [];
    $Art_man = new articleManager();

    if(empty($_SESSION['email'])){
        header("location:/projetphp/vue/index.php");
    }
?>

<html lang="fr">
    <?php include_once('../public/template/head.php'); ?>

    <body>
        <?php include_once('../public/template/header.php'); ?>

        <main>
            <div class="container-fluid mt-3">
                <div class="row">
                    <div class="col-2">
                        <button onclick="window.location.href='../vue/menuUtilisateur.php' class="btn btn-success">Retour</button>
                    </div>
                </div>
                <div class="row">
                    <div class="col-4"></div>
                    <div class="col-4">
                        <h3 class="text-center">Données personnelle du clients</h3>
                        <form action="../controllers/modifProfil.php" method="POST">
                            <div class="form-row">
                                <div class="form-group col">
                                    <label for="email">Email</label>
                                    <input type="email" class="form-control" id="email" name="email" value="<?php echo $_SESSION['email']; ?>">
                                </div>
                            </div>
                            <div class="form-row">
                                <div class="form-group col">
                                    <label for="mdp">Mot de passe</label>
                                    <input type="text" class="form-control" id="mdp" name="mdp" required>
                                </div>
                            </div>
                            <div class="form-row">
                                <div class="form-group col-6">
                                    <label for="nom">Nom</label>
                                    <input type="text" class="form-control" id="nom" name="nom" value="<?php echo $_SESSION['nom']; ?>">
                                </div>
                                <div class="form-group col-6">
                                    <label for="prenom">Prénom</label>
                                    <input type="text" class="form-control" id="prenom" name="prenom" value="<?php echo $_SESSION['prenom']; ?>">
                                </div>
                            </div>
                            <div class="form-row">
                                <div class="form-group col">
                                    <label for="raisonSocial">Raison social</label>
                                    <input type="text" class="form-control" id="raisonSocial" name="raisonSocial" value="<?php echo $_SESSION['raison']; ?>">
                                </div>
                            </div>
                            <div class="form-row">
                                <div class="form-group col">
                                    <label for="adresse">Adresse</label>
                                    <input type="text" class="form-control" id="adresse" name="adresse" value="<?php echo $_SESSION['adresse']; ?>">
                                </div>
                            </div>
                            <div class="form-row">
                                <div class="form-group col-8">
                                    <label for="ville">Ville</label>
                                    <input type="text" class="form-control" id="ville" name="ville" value="<?php echo $_SESSION['ville']; ?>">
                                </div>
                                <div class="form-group col-4">
                                    <label for="codePostal">Code postal</label>
                                    <input type="number" class="form-control" id="codePostal" name="codePostal" value="<?php echo $_SESSION['codePostal']; ?>">
                                </div>
                            </div>
                            <div class="form-row">
                                <div class="form-group col">
                                    <label for="tel">N° de téléphone</label>
                                    <input type="number" class="form-control" id="tel" name="tel" value="<?php echo $_SESSION['tel']; ?>">
                                </div>
                            </div>
                            <div class="text-center"><button type="submit" name="modification" id="modification" class="btn btn-success mt-4">Modifier les informations</button></div>
                        </form>
                    </div>
                </div>
            </div>
        </main>
```

```

<?php include_once('../public/template/footer.php'); ?>

</body>
</html>

```

On y retrouve toutes nos données sauf notre mot de passe pour des raisons de sécurité. Vous pouvez à partir de là, modifier vos données personnelles. Voici le code pour la modification :

```

<?php
    session_start();
    include_once('../models/ConnexionBdd.php');

    if(isset($_POST['modification'])){
        if(empty(isset($_POST['mail']))){
            $email = $_SESSION['email'];
        } else {
            $email = htmlspecialchars($_POST['email']);
        }
        if(empty(isset($_POST['nom']))){
            $nom = $_SESSION['nom'];
        } else {
            $nom = htmlspecialchars($_POST['nom']);
        }
        if(empty(isset($_POST['prenom']))){
            $prenom = $_SESSION['prenom'];
        } else {
            $prenom = htmlspecialchars($_POST['prenom']);
        }
        if(empty(isset($_POST['raisonSocial']))){
            $raisonSocial = $_SESSION['raison'];
        } else {
            $raisonSocial = htmlspecialchars($_POST['raisonSocial']);
        }
        if(empty(isset($_POST['adresse']))){
            $adresse = $_SESSION['adresse'];
        } else {
            $adresse = htmlspecialchars($_POST['adresse']);
        }
        if(empty(isset($_POST['ville']))){
            $ville = $_SESSION['ville'];
        } else {
            $ville = htmlspecialchars($_POST['ville']);
        }
        if(empty(isset($_POST['codePostal']))){
            $codePostal = $_SESSION['codePostal'];
        } else {
            $codePostal = htmlspecialchars($_POST['codePostal']);
        }
        if(empty(isset($_POST['ntel']))){
            $ntel = $_SESSION['tel'];
        } else {
            $ntel = htmlspecialchars($_POST['ntel']);
        }
        if(empty(isset($_POST['mdp']))){
            $mdp = $_SESSION['mdp'];
        }
    }

```

```

if(empty(isset($_POST['mdp']))){
    $mdp = $_SESSION['mdp'];
} else {
    $mdp = htmlspecialchars($_POST['mdp']);
}

$DBase = new Connect();
$db = $DBase->connexion();
$q = $db->prepare("UPDATE utilisateur SET `Nom` = '' . $nom . '', `Prenom` = '' . $prenom . '', `email` = '' . $email . '', `MDP` = '' . $mdp . '', `Raison` = '' . $raisonSocial . '', `adresse` = '' . $adresse . '', `ville` = '' . $ville . '', `codepostal` = '' . $codePostal . '', `tel` = '' . $tel . '' WHERE id = '' . $_SESSION['id']);
$q->execute();

$DBase2 = new Connect();
$recup = $DBase2->connexion();
$userConnexion = $recup->prepare("SELECT id, email, Nom, Prenom, Raison, adresse, codepostal, ville, tel, MDP FROM utilisateur WHERE email = '' . $email . ''");
$userConnexion->execute();
$user = $userConnexion->fetch(PDO::FETCH_ASSOC);

if ($user['email'] != null) {
    $_SESSION['id'] = $user['id'];
    $_SESSION['email'] = $user['email'];
    $_SESSION['nom'] = $user['Nom'];
    $_SESSION['prenom'] = $user['Prenom'];
    $_SESSION['raison'] = $user['Raison'];
    $_SESSION['adresse'] = $user['adresse'];
    $_SESSION['codePostal'] = $user['codepostal'];
    $_SESSION['ville'] = $user['ville'];
    $_SESSION['tel'] = $user['tel'];
    $_SESSION['mdp'] = $user['MDP'];
}
header("location:/projetphp/vue/menuUtilisateur.php");
}

?>

```

Comme vous pouvez le voir depuis plusieurs formulaires on utilise htmlspecialchars qui est une autre sécurité qui nous protège des injection SQL dans notre base de données.

Une fois vos données modifiées, vous êtes renvoyé sur votre page utilisateur pour constater vos changements d'information.

Chapitre Panier d'achat

Le cas du panier est spécial car il utilise une classe et à la fois les variables super globales. Voici le code de la classe Panier (qui nous a été fournis par le professeur) avec une fonction mineure ajouter de « destruction de panier » créée par moi-même :

```

<?php
class Panier {
    public function creationPanier()
    {
        if (! isset($_SESSION['panier'])) {
            $_SESSION['panier'] = array();
            $_SESSION['panier']['id_prod'] = array();
            $_SESSION['panier']['quantite'] = array();
            $_SESSION['panier']['prix'] = array();
        }
        return true;
    }
    public function ajouterArticle($id_prod, $quantite, $prix)
    {
        if ($this->creationPanier()) {
            $positionProduit = array_search($id_prod, $_SESSION['panier']['id_prod']);
            if ($positionProduit !== false) {
                $_SESSION['panier']['quantite'][$positionProduit] += $quantite;
            } else {
                array_push($_SESSION['panier']['id_prod'], $id_prod);
                array_push($_SESSION['panier']['quantite'], $quantite);
                array_push($_SESSION['panier']['prix'], $prix);
            }
        }
    }
    public function montantTotal()
    {
        $total = 0;
        if (isset($_SESSION['panier']['id_prod'])) {
            $nbArticle = sizeof($_SESSION['panier']['id_prod']);
            for ($i = 0; $i < $nbArticle ; $i++) {
                $total += $_SESSION['panier']['quantite'][$i] * $_SESSION['panier']['prix'][$i];
            }
        }
        return $total;
    }
    public function getPanier()
    {
        $panier = [];
        for ($i = 0; $i < count($_SESSION['panier']); $i++) {
            $panier += $_SESSION['panier'];
        }
        return $panier;
    }
}

```

```

public function nbArticles()
{
    $total = 0;
    if (isset($_SESSION['panier']['id_prod'])) {
        for ($i = 0; $i < count($_SESSION['panier']['id_prod']); $i++) {
            $total += $_SESSION['panier']['quantite'][$i];
        }
    }
    return $total;
}

public function supprimerArticle($id_prod)
{
    if ($this->creationPanier()) {
        $tmp = array();
        $tmp['id_prod'] = array();
        $tmp['quantite'] = array();
        $tmp['prix'] = array();

        for ($i = 0; $i < count($_SESSION['panier']['id_prod']); $i++) {
            if ($_SESSION['panier']['id_prod'][$i] != $id_prod) {
                array_push($tmp['id_prod'], $_SESSION['panier']['id_prod'][$i]);
                array_push($tmp['quantite'], $_SESSION['panier']['quantite'][$i]);
                array_push($tmp['prix'], $_SESSION['panier']['prix'][$i]);
            }
        }
        $_SESSION['panier'] = $tmp;
        unset($tmp);
    } else
        $this->error = "Un problème est survenu veuillez contacter l'administrateur du site.";
}

public function modifierQteArticle($id_prod, $qteProduit)
{
    if ($this->creationPanier()) {
        if ($qteProduit > 0) {
            $positionProduit = array_search($id_prod, $_SESSION['panier']['id_prod']);

            if ($positionProduit !== false) {
                $_SESSION['panier']['quantite'][$positionProduit] = $qteProduit;
            }
        }
    } else
        $this->error = "Un problème est survenu veuillez contacter l'administrateur du site.";
}

```

```

public function detruirePanier(){
    unset($_SESSION['panier']);
}

```

On l'appelle à chaque page pour être sûr qu'il soit bien initialisé comme ceci :

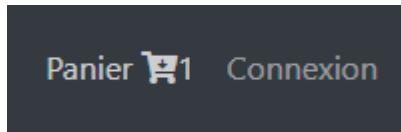
```

include_once('../public/template/session.php');
include_once('../models/ConnexionBdd.php');
include_once('../models/CategorieManager.php');
include_once('../models/panier.php');
include_once('../models/ArticleManager.php');

$categorie = [];
$DBase = new Connect();
$db = $DBase->connexion();
$Cat_man = new CategorieManager();
$panier = new Panier();
$article = [];
$Art_man = new ArticleManager();
$ventes = $Art_man->getMeilleureVente();
$tailleVente = sizeof($ventes);
$articlePromo = $Art_man->lastPromotion();
$tailleArticlePromo = sizeof($articlePromo);
$categorie = $Cat_man->listeCategorie();
$tailleCategories = sizeof($categorie);

```

De ce fait, on peut prendre des articles dans notre panier connecté ou non. Exemple ci-dessous :



Le panier se présente au client comme ceci, il suffit juste de cliquer sur « panier » ou le caddie blanc et il apparaît comme ci-dessous :

Panier d'achat

Produit	Quantité	Prix
Poulie	1	2 €

- + Enlever

Montant Total : 2€ Article Total : 1

Payer

A screenshot of a mobile application showing a shopping cart. The title "Panier d'achat" is at the top. Below it is a table with three columns: "Produit", "Quantité", and "Prix". A single row shows "Poulie" with a quantity of "1" and a price of "2 €". To the right of the table are three buttons: a red minus sign, a green plus sign, and a yellow "Enlever" (Remove) button. Below the table, it says "Montant Total : 2€" and "Article Total : 1". At the bottom is a large grey "Payer" (Pay) button.

S'il y a un article et ensuite sans article :

Panier d'achat

Produit	Quantité	Prix
Panier Vide		

Montant Total : 0€ Article Total : 0

Payer

A screenshot of a mobile application showing an empty shopping cart. The title "Panier d'achat" is at the top. Below it is a table with three columns: "Produit", "Quantité", and "Prix". The table has one row with the message "Panier Vide". Below the table, it says "Montant Total : 0€" and "Article Total : 0". At the bottom is a large grey "Payer" (Pay) button.

Le code HTML pour le panier :

```

<li class="nav-item">
    <a class="nav-link" data-toggle="modal" data-target="#panierAchat">Panier <i class="fas fa-cart-arrow-down"></i><span id="totalCaddie"><?php echo $panier->nbArticles()?></span></a>
</li>
<li class="nav-item">

```

```

<div class="modal fade" id="panierAchat" tabindex="-1" role="dialog" aria-labelledby="panierAchat" aria-hidden="true">
    <div class="modal-dialog" role="document">
        <div class="modal-content">
            <div class="card">
                <article class="card-body">
                    <h4 class="card-title text-center mb-4 mt-1">Panier d'achat</h4>
                    <table class="table">
                        <thead>
                            <tr>
                                <th scope="col">Produit</th>
                                <th scope="col">Quantité</th>
                                <th scope="col">Prix</th>
                                <th scope="col"></th>
                                <th scope="col"></th>
                                <th scope="col"></th>
                            </tr>
                        </thead>
                        <tbody id="affichagePanier">
                            <?php
                                if(!empty($_SESSION['panier']['id_prod'])){
                                    for ($i = 0; $i < count($_SESSION['panier']['id_prod']); $i++) {
                                        $articleDisplay[$i] = $Art_man->getInfo($_SESSION['panier']['id_prod'][$i]);
                                    }
                                    if (isset($_SESSION['panier'])) {
                                        for ($i = 0; $i < count($_SESSION['panier']['id_prod']); $i++) {
                                            $articleDisplay[$i] = $Art_man->getInfo($_SESSION['panier']['id_prod'][$i]);
                                            echo "<tr>";
                                            echo "<th scope='row'></th>";
                                            echo "<td>" . $articleDisplay[$i][0]->getLibelle() . "</td>";
                                            echo "<td>" . $_SESSION['panier'][quantite][$i] . "</td>";
                                            echo "<td>" . $_SESSION['panier'][$i]*$_SESSION['panier']['prix'][$i] . " €</td>";
                                            echo "<td><a class='card-link btn btn-danger modifPanier' href='/projetphp/controllers/modifAchatPanier.php?id= " . $_SESSION['panier']['id_prod'][$i] . "&qt=" . $_SESSION['panier'][$i]*$_SESSION['panier']['prix'][$i] . "' name='modifPanier' id='modifPanier'></a></td>";
                                            echo "<td><a class='card-link btn btn-success ajouterPanier' href='/projetphp/controllers/ajouteAuPanier.php?id= " . $_SESSION['panier']['id_prod'][$i] . "' name='acheterPanier' id='acheterPanier'></a></td>";
                                            echo "<td><a class='card-link btn btn-warning suppPanier' href='/projetphp/controllers/suppProduitAchat.php?id= " . $_SESSION['panier']['id_prod'][$i] . "' name='suppPanier' id='suppPanier'>Enlever</a></td>";
                                            echo "</tr>";
                                        }
                                    }
                                };
                            </tbody>
                        </table>
                    </article>
                <?php
                    if(empty($_SESSION['panier']['id_prod'])){
                        echo "<div id='AVIDER' class='text-center'>";
                        echo "<p>Panier Vide</p>";
                        echo "</div>";
                    }
                ?>
                <div class="text-center">
                    <table class="table">
                        <tr>
                            <td>Montant Total : <span id="totalPrixAchat"><?php echo $panier->montantTotal()?></span>€ </td>
                            <td>Article Total : <span id="totalCaddie2"><?php echo $panier->nbArticles()?></span> </td>
                        </table>
                </div>
                <a href="#" class="btn btn-secondary btn-lg active" role="button" aria-pressed="true">Payer</a>
            </div>
        </div>
    </div>

```

Afin de rendre le panier dynamique pour le client du code javascript a été écrit en conséquence :

```
<script>
(function($){
    $('.modifPanier').click(function(event){
        event.preventDefault();
        $.get($(this).attr('href'),{},function(data){
            if(data.error){
                alert(data.message)
            }
            $('#totalCaddie').empty().append(data.quantite);
            $('#totalCaddie2').empty().append(data.quantite);
            $('#totalPrixAchat').empty().append(data.total);
            $('#affichagePanier').empty();
            for (let i = 0; i < data.donneProd.length; i++) {
                $('#affichagePanier').append(data.donneProd[i]);
            }
        },'json');
        document.getElementById('AVIDER').innerHTML = "";
        return false;
    });
})(jQuery);
</script>
```

Raccordée à ce fichier qui recharge le panier d'achat avec les bonnes valeurs souhaitées :

```
<?php
include_once('../public/template/session.php');
include_once('../models/ConnexionBdd.php');
include_once('../models/ArticleManager.php');
include_once('../models/panier.php');

$panier = new Panier();
$article = [];
$DBase = new Connect();
$db = $DBase->connexion();
$Art_man = new articleManager();

if(isset($_GET['id'])){
    $article = $Art_man->getInfo($_GET['id']);
    $articlePromo = $Art_man->getPromotionArticle($_GET['id']);
}

if($article[0]->getPromotion() == NULL){
    $panier->ajouterArticle($article[0]->getId(), 1, $article[0]->getPrix());
}
else{
    $panier->ajouterArticle($article[0]->getId(), 1, $articlePromo[0]->getPrixPromo());
}

$json['error'] = false;
$json['message'] = 'ça marche';
$json['total'] = $panier->montantTotal();
$json['quantite'] = $panier->nBArticles();

$articleDisplay = [];
$objetProduit = [];

for ($i = 0; $i < count($_SESSION['panier']['id_prod']); $i++) {
    $articleDisplay[$i] = $Art_man->getInfo($_SESSION['panier']['id_prod'][$i]);
}

if (isset($_SESSION['panier'])) {
    for ($i = 0; $i < count($_SESSION['panier']['id_prod']); $i++) {
        $articleDisplay[$i] = $Art_man->getInfo($_SESSION['panier']['id_prod'][$i]);
        $objetProduit[$i] = "<tr><th scope='row'></th>
<td>" . $articleDisplay[$i][0]->getLibelle() . "</td>
<td>" . $_SESSION['panier'][0]['quantite'][$i] . "</td>
<td>" . $_SESSION['panier'][0]['quantite'][$i]*$_SESSION['panier']['prix'][$i] . " €</td>
<td><a class='card-link btn btn-danger supprimerPanier' href='/projetphp/controllers/suppAchatPanier.php?id= " . $_SESSION['panier']['id_prod'][$i] . " name='suppPanier' id='suppPanier'></a></td>
<td><a class='card-link btn btn-success ajouterPanier' href='/projetphp/controllers/ajouteAuPanier.php?id= " . $_SESSION['panier']['id_prod'][$i] . " name='acheterPanier' id='acheterPanier'></a></td>
<td><a class='card-link btn btn-warning suppPanier' href='/projetphp/controllers/suppProduitAchat.php?id= " . $_SESSION['panier']['id_prod'][$i] . " name='suppPanier' id='suppPanier'>Enlever</a></td></tr>";
    }
}

$json['donneProd'] = $objetProduit;
echo json_encode($json);
```

Ces fichiers sont raccordés dans le panier aux trois boutons précédemment montrés dans une image juste au-dessus de la présentation du panier et la fonction ajouter panier est aussi dans le bouton acheter en dessous des produits.

Panel Administrateur :

On va utiliser une barre de navigation horizontale quand on va cliquer sur un des onglets. Il va faire appel à une page php qui apparaîtra avec une animation de fondu.

The screenshot shows the P.M.U Panel Administrator homepage. On the left, a sidebar lists navigation options: Accueil (highlighted in blue), Catégorie, Produit, Conseil, Utilisateur, and Commande. The main content area features a "Dernière promotion :" section with an image of a tire and a "Pneu Poulie" product card. The card includes a price of 2 x 0.60 ct (-30%), an "Information" button, and an "Acheter" button. To the right, a "Meilleur vente :" section shows a single item: "Vis longue". Below these sections, a table titled "Les 5 dernières ventes" displays five recent sales. The table has columns for #, Quantité, Date, Nom, Prénom, Email, Référence, Libelle, and Prix. One row is visible, showing a quantity of 10, date 2020-11-18, name Six, first name Tristan, email sixtristan@orange.fr, reference A701, libelle Vis longue, and price 90 €.

#	Quantité	Date	Nom	Prénom	Email	Référence	Libelle	Prix
1	10	2020-11-18	Six	Tristan	sixtristan@orange.fr	A701	Vis longue	90 €

Les classes d'objet utilisés pour la page d'accueil sont :

- La classe catégorie qui va contenir les catégories de produit :

```
class Categorie
{
    private $_id;
    private $_libelle;

    public function __construct($donnees)
    {
        $this->_id = $donnees['id_cat_prod'];
        $this->_libelle = $donnees['libelle'];
    }

    public function setId($id){
        $this->_id = $id;
    }

    public function setLibelle($libelle){
        $this->_libelle = $libelle;
    }

    public function getId()
    {
        return $this->_id;
    }

    public function getLibelle()
    {
        return $this->_libelle;
    }
}
```

- La classe Produit qui va contenir les produits et leurs promotions :

```
class Article
{
    private $_id ;
    private $_reference = "";
    private $_quantité = 0;
    private $_prix_unitaire = 0;
    private $_libelle = "";
    private $_description = "";
    private $_categorie = "";
    private $_chemin = "";
    private $_nomImage = "";
    private $_id_promotion = "";
    private $_prixPromo;
    private $_pourcentage;
    private $_datefin;
```

```

public function __construct($donnees)
{
    if (isset($donnees['id_produit'])) {
        $this->_id = $donnees['id_produit'];
    }
    if (isset($donnees['reference'])) {
        $this->_reference = $donnees['reference'];
    }
    if (isset($donnees['quantité'])) {
        $this->_quantité = $donnees['quantité'];
    }
    if (isset($donnees['prix_unitaire'])) [
        $this->_prix_unitaire = $donnees['prix_unitaire'];
    ]
    if (isset($donnees['libelle'])) {
        $this->_libelle = $donnees['libelle'];
    }
    if (isset($donnees['description'])) {
        $this->_description = $donnees['description'];
    }
    if (isset($donnees['categorie'])) {
        $this->_categorie = $donnees['categorie'];
    }
    if (isset($donnees['chemin'])) {
        $this->_chemin = $donnees['chemin'];
    }
    if (isset($donnees['image'])) {
        $this->_nomImage = $donnees['image'];
    }
    if (isset($donnees['promotion'])) {
        $this->_id_promotion = $donnees['promotion'];
    }
    if (isset($donnees['prix'])) {
        $this->_prixPromo = $donnees['prix'];
    }
    if (isset($donnees['pourcentage'])) {
        $this->_pourcentage = $donnees['pourcentage'];
    }
    if (isset($donnees['datefin'])) {
        $this->_datefin = $donnees['datefin'];
    }
}

```

```

    public function getPromotion()
    {
        return $this->_id_promotion;
    }

    public function getPrixPromo()
    {
        return $this->_prixPromo;
    }

    public function getPourcentage()
    {
        return $this->_pourcentage;
    }

    public function getDateFin()
    {
        return $this->_datefin;
    }
}

```

```

public function getId()
{
    return $this->_id;
}

public function getReference()
{
    return $this->_reference;
}

public function getQuantite()
{
    return $this->_quantité;
}

public function getPrix()
{
    return $this->_prix_unitaire;
}

public function getLibelle()
{
    return $this->_libelle;
}

public function getDescription()
{
    return $this->_description;
}

public function getCategorie()
{
    return $this->_categorie;
}

public function getImage()
{
    return $this->_chemin;
}

public function getNomImage()
{
    return $this->_nomImage;
}

```

- La classe Conseil qui va contenir les conseils :

```
class Conseil
{
    private $_id;
    private $_titre;
    private $_libelle;
    private $_video;

    public function __construct($donnees)
    {
        $this->_id = $donnees['id_cons'];
        $this->_titre = $donnees['titre'];
        $this->_libelle = $donnees['libelle'];
        $this->_video = $donnees['video'];
    }
}
```

```
public function setId($id){
    $this->_id = $id;
}

public function setTitre($titre){
    $this->_titre = $titre;
}

public function setLibelle($libelle){
    $this->_libelle = $libelle;
}

public function setVideo($video){
    $this->_video = $video;
}

public function getId()
{
    return $this->_id;
}

public function getLibelle()
{
    return $this->_libelle;
}

public function getTitre()
{
    return $this->_titre;
}

public function getVideo()
{
    return $this->_video;
}
```

- La classe Client qui va contenir les utilisateurs :

```
class Client
{
    private $_id ;
    private $_nom = "";
    private $_prenom = "";
    private $_email = "";
    private $_adresse = "";
    private $_cp= "";
    private $_ville = "";
    private $_tel = "";
    private $_categorie = "";
    private $_id_cat = "";

    public function __construct($donnees)
    {
        if (isset($donnees['id'])) {
            $this->_id = $donnees['id'];
        }
        if (isset($donnees['Nom'])) {
            $this->_nom = $donnees['Nom'];
        }
        if (isset($donnees['Prenom'])) {
            $this->_prenom = $donnees['Prenom'];
        }
        if (isset($donnees['email'])) {
            $this->_email = $donnees['email'];
        }
        if (isset($donnees['adresse'])) {
            $this->_adresse = $donnees['adresse'];
        }
        if (isset($donnees['codepostal'])) {
            $this->_cp = $donnees['codepostal'];
        }
        if (isset($donnees['ville'])) {
            $this->_ville = $donnees['ville'];
        }
        if (isset($donnees['tel'])) {
            $this->_tel = $donnees['tel'];
        }
        if (isset($donnees['libellé'])) {
            $this->_categorie = $donnees['libellé'];
        }
        if (isset($donnees['id_cat'])) {
            $this->_id_cat = $donnees['id_cat'];
        }
    }
}
```

```
public function getId()
{
    return $this->_id;
}

public function getNom()
{
    return $this->_nom;
}

public function getPrenom()
{
    return $this->_prenom;
}

public function getEmail()
{
    return $this->_email;
}

public function getAdresse()
{
    return $this->_adresse;
}

public function getCp()
{
    return $this->_cp;
}

public function getVille()
{
    return $this->_ville;
}

public function getTel()
{
    return $this->_tel;
}

public function getCategorie()
{
    return $this->_categorie;
}

public function getId_cat()
{
    return $this->_id_cat;
}
```

- La classe Commande qui va contenir les commandes :

```
class Commande
{
    private $_id ;
    private $_quantite = "";
    private $_date = "";
    private $_nom = "";
    private $_prenom = "";
    private $_email= "";
    private $_reference = "";
    private $_libelle = "";
    private $_prix = "";

    public function __construct($donnees)
    {
        if (isset($donnees['id_commande'])) {
            $this->_id = $donnees['id_commande'];
        }
        if (isset($donnees['quantité'])) {
            $this->_quantite = $donnees['quantité'];
        }
        if (isset($donnees['date'])) {
            $this->_date = $donnees['date'];
        }
        if (isset($donnees['Nom'])) {
            $this->_nom = $donnees['Nom'];
        }
        if (isset($donnees['Prenom'])) {
            $this->_prenom = $donnees['Prenom'];
        }
        if (isset($donnees['email'])) {
            $this->_email = $donnees['email'];
        }
        if (isset($donnees['reference'])) {
            $this->_reference = $donnees['reference'];
        }
        if (isset($donnees['libelle'])) {
            $this->_libelle = $donnees['libelle'];
        }
        if (isset($donnees['prix'])) {
            $this->_prix = $donnees['prix'];
        }
    }

    public function getId()
    {
        return $this->_id;
    }

    public function getDate()
    {
        return $this->_date;
    }

    public function getNom()
    {
        return $this->_nom;
    }

    public function getPrenom()
    {
        return $this->_prenom;
    }

    public function getEmail()
    {
        return $this->_email;
    }

    public function getReference()
    {
        return $this->_reference;
    }

    public function getQuantite()
    {
        return $this->_quantite;
    }

    public function getPrix()
    {
        return $this->_prix;
    }

    public function getLibelle()
    {
        return $this->_libelle;
    }
}
```

- La classe Ucategory qui va contenir les catégories d'utilisateurs :

```
class Ucategorye
{
    private $_id;
    private $_libelle;

    public function __construct($donnees)
    {
        $this->_id = $donnees['id_cat'];
        $this->_libelle = $donnees['libellé'];
    }

    public function setId($id){
        $this->_id = $id;
    }

    public function setLibelle($libelle){
        $this->_libelle = $libelle;
    }

    public function getId()
    {
        return $this->_id;
    }

    public function getLibelle()
    {
        return $this->_libelle;
    }
}
```

Le code du panel admin :

```
<?php
    include_once('../public/template/session.php');
    include_once('../models/ArticleManager.php');
    include_once('../models/ConnexionBdd.php');
    include_once('../models/panier.php');
    include_once('../models/CategorieManager.php');
    include_once('../models/conseilManager.php');
    include_once('../models/ClientManager.php');

    $panier = new Panier();
    $article = [];
    $Art_man = new articleManager();
    $DBase = new Connect();
    $db = $DBase->connexion();
    $Cat_man = new CategorieManager();
    $Art_man = new articleManager();
    $Conseil_man = new conseilManager();
    $Cli_man = new clientManager();

    if($_SESSION['libellé'] != "Administrateur"){
        header("location:/projetphp/vue/index.php");
    }

?>
```

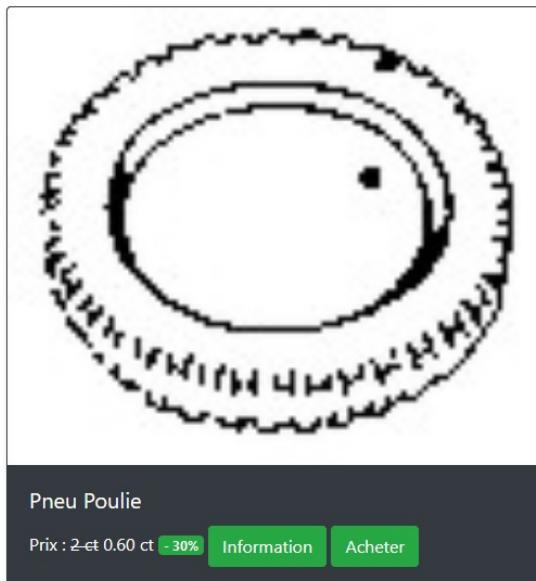
```
<html class="m-0 p-0">
<head>
    <?php include_once('../public/template/head.php'); ?>
</head>
<body>
    <?php include_once('../public/template/header.php'); ?>
    <div class="container-fluid">
        <div class="row">
            <div class="col-2 bg-dark">
                <p class="titre nav-link text-white">Panel Admin</p>
                <div class="nav flex-column nav-pills text-white" id="v-pills-tab" role="tablist" aria-orientation="vertical">
                    <a class="nav-link active p-4" id="v-pills-acceuil-tab" data-toggle="pill" href="#v-pills-acceuil" role="tab" aria-controls="v-pills-acceuil" aria-selected="true">Accueil</a>
                    <a class="nav-link p-4" id="v-pills-categorie-tab" data-toggle="pill" href="#v-pills-categorie" role="tab" aria-controls="v-pills-categorie" aria-selected="false">Categorie</a>
                    <a class="nav-link p-4" id="v-pills-produit-tab" data-toggle="pill" href="#v-pills-produit" role="tab" aria-controls="v-pills-produit" aria-selected="false">Produit</a>
                    <a class="nav-link p-4" id="v-pills-conseil-tab" data-toggle="pill" href="#v-pills-conseil" role="tab" aria-controls="v-pills-conseil" aria-selected="false">Conseil</a>
                    <a class="nav-link p-4" id="v-pills-utilisateur-tab" data-toggle="pill" href="#v-pills-utilisateur" role="tab" aria-controls="v-pills-utilisateur" aria-selected="false">Utilisateur</a>
                    <a class="nav-link p-4" id="v-pills-commande-tab" data-toggle="pill" href="#v-pills-commande" role="tab" aria-controls="v-pills-commande" aria-selected="false">Commande</a>
                </div>
            </div>
            <div class="col-10">
                <div class="tab-content id="v-pills-tabContent">
                    <div class="tab-pane fade show active" id="v-pills-acceuil" role="tabpanel" aria-labelledby="v-pills-acceuil-tab"><?php include_once('adminAcceuil.php') ?></div>
                    <div class="tab-pane fade" id="v-pills-categorie" role="tabpanel" aria-labelledby="v-pills-categorie-tab"><?php include_once('adminCategorie.php') ?></div>
                    <div class="tab-pane fade" id="v-pills-produit" role="tabpanel" aria-labelledby="v-pills-produit-tab"><?php include_once('adminProduit.php') ?></div>
                    <div class="tab-pane fade" id="v-pills-conseil" role="tabpanel" aria-labelledby="v-pills-conseil-tab"><?php include_once('adminConseil.php') ?></div>
                    <div class="tab-pane fade" id="v-pills-utilisateur" role="tabpanel" aria-labelledby="v-pills-utilisateur-tab"><?php include_once('adminUtilisateur.php') ?></div>
                    <div class="tab-pane fade" id="v-pills-commande" role="tabpanel" aria-labelledby="v-pills-commande-tab"><?php include_once('adminCommande.php') ?></div>
                </div>
            </div>
        </div>
    </div>
</body>
</html>
```

Accueil

Affichage de la dernière promotion

On va afficher la dernière promotion ajoutée, puis on va l'afficher sous forme de carte avec une image, le nom du produit son ancien prix et le nouveau, le pourcentage de la promotion et deux boutons un pour avoir plus d'information et un pour acheter le produit.

Dernière promotion :



Le code d'affichage de la dernière promotion.

```
<div class="col-4 shadow-sm m-1 p-3 mb-5 bg-white rounded">
<h2 class="text-center">Dernière promotion :</h2>
<div class="card border border-dark shadow-sm bg-dark text-white rounded">
  <img src=<?php echo '$articlePromo[0]->getImage()' ?> class="card-img-top" alt="<?php echo $articlePromo[0]->getNomImage()?>">
  <div class="card-body">
    <div class="row">
      <div class="col">
        <h5 class="card-title"><?php echo $articlePromo[0]->getLibelle() ?></h5>
          <span class="card-link">Prix : <del><?php echo $articlePromo[0]->getPrix() ?> ct</del> <?php echo $articlePromo[0]->getPrixPromo() ?> ct</span>
          <span class="badge badge-success"><?php echo $articlePromo[0]->getPourcentage() ?>%</span>
        <a href="pageProduit.php?id=<?php echo $articlePromo[0]->getId()?>"><button class="card-link btn btn-success" name="info" id="info">Information</button></a>
        <button class="card-link btn btn-success" name="acheter" id="acheter">Acheter</button>
      </div>
    </div>
  </div>
</div>
```

La fonction qui permet de récupérer la dernière promotion ajouter encore active.

```
public function lastPromotion()
{
  $promotions = [];
  $DBase = new Connect();
  $db = $DBase->connexion();
  $q = $db->prepare('SELECT id_produit,libelle,chemin,image,description,prix_unitaire,quantité,ROUND(prix_unitaire*(pourcentage/100),2)as prix,promotion,pourcentage,datefin
from produit join objet on objet.id_prod = id_produit
join promotion on promotion.id = produit.promotion
where CURDATE() between datedebut and datefin ORDER by promotion.datedebut DESC LIMIT 4 ');
  $q->execute();
  while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
  {
    $promotions[] = new Article($donnees);
  }
  return $promotions;
}
```

Affichage des meilleures ventes

Ici on va juste afficher les meilleures ventes actuelle.

Meilleur vente :

Vis longue

Le code d'affichage des meilleures ventes.

```
<div class="col-6 shadow-sm m-1 p-3 mb-5 bg-white rounded">
    <h2 class="text-center">Meilleur vente :</h2>
    <ul class="text-center">
        <?php
            for ($i=0; $i < $tailleVente; $i++) {
                echo '<li> '. $ventes[$i]->getLibelle() . '</li> ';
            }
        ?>
    </ul>
</div>
```

La fonction qui permet de récupérer les 5 premiers produit par ordre décroissant, produits ayant fais le plus de commande.

```
public function getMeilleureVente()
{
    $ventes = [];

    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('SELECT `libelle` FROM `produit` JOIN commande ON produit.id_produit=commande.id_produit
        GROUP by commande.id_produit ORDER BY COUNT(commande.id_produit) LIMIT 5 ');
    $q->execute();

    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $ventes[] = new Article($donnees);
    }
    return $ventes;
}
```

Affichage des 5 dernières ventes

Ici on va afficher les 5 dernières ventes.

Les 5 dernières ventes

#	Quantité	Date	Nom	Prénom	Email	Référence	Libelle	Prix
1	10	2020-11-18	Six	Tristan	sixtristan@orange.fr	A701	Vis longue	90 €

Le code d'affichage des 5 dernières ventes.

```
<div class="row">
    <div class="col-12 shadow-sm m-1 p-3 mb-5 bg-white rounded">
        <h2 class="text-center">Les 5 dernières ventes</h2>
        <div class="table-responsive-sm m-2 mt-5">
            <table class="table">
                <thead class="thead-dark">
                    <tr>
                        <th scope="col">#</th>
                        <th scope="col">Quantité</th>
                        <th scope="col">Date</th>
                        <th scope="col">Nom</th>
                        <th scope="col">Prénom</th>
                        <th scope="col">Email</th>
                        <th scope="col">Référence</th>
                        <th scope="col">Libelle</th>
                        <th scope="col">Prix</th>
                    </tr>
                </thead>
                <tbody>
                    <?php
                        for ($i=0; $i < $tailleCommande; $i++) {
                    ?>
                    <tr>
                        <th scope="row"><?php echo $commandes[$i]->getId()?></th>
                        <td><?php echo $commandes[$i]->getQuantite()?></td>
                        <td><?php echo $commandes[$i]->getDate()?></td>
                        <td><?php echo $commandes[$i]->getNom()?></td>
                        <td><?php echo $commandes[$i]->getPrenom()?></td>
                        <td><?php echo $commandes[$i]->getEmail()?></td>
                        <td><?php echo $commandes[$i]->getReference()?></td>
                        <td><?php echo $commandes[$i]->getLibelle()?></td>
                        <td><?php echo $commandes[$i]->getPrix()?> €</td>
                    <?php
                        }
                    ?>
                </tbody>
            </table>
        </div>
    </div>
</div>
```

La fonction qui permet de récupérer les 5 dernières ventes par ordre décroissant.

```
public function lastCommande()
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $commandes = [];

    $q = $db->prepare('SELECT id_commande, commande.quantité, date, utilisateur.Nom, utilisateur.Prenom, utilisateur.email, produit.reference, produit.libelle,
    (produit.prix_unitaire * commande.quantité) AS prix FROM commande
    JOIN utilisateur ON commande.id_utilisateur = utilisateur.id
    JOIN produit ON commande.id_produit = produit.id_produit ORDER by date DESC LIMIT 5 ');
    $q->execute();
    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $commandes[] = new Commande($donnees);
    }

    return $commandes;
}
```

Catégorie

On affiche un tableau avec toute les catégories. Grâce au bouton on peut ajouter, modifier ou supprimer une catégorie et tous ceci est sous forme de Modal.

#	Categorie	+
1	Vis	 
2	Poulie	 
5	Roue	 
7	Support	 

Le code d'affichage du tableau des catégories.

```
<table class="table">
  <thead class="thead-dark">
    <tr>
      <th scope="col">#</th>
      <th scope="col">Categorie</th>
      <th scope="col">
        <button type="button" class="btn btn-dark" data-toggle="modal" data-target="#modalAddCat">
          | </i>
        </button>
      </th>
    </tr>
  </thead>
  <tbody>
    <?php
      for ($i=0; $i < $tailleCategories; $i++) {
    ?>
      <tr>
        <th scope="row"><?php echo $categories[$i]->getId() ?></th>
        <td><?php echo $categories[$i]->getLibelle() ?></td>
        <td>
          <button type="button" class=" open-modif-cat btn btn-light" data-toggle="modal" data-id = "<?php echo $categories[$i]->getId()?" data-libelle = "<?php echo $categories[$i]->getLibelle() ?>" data-target="#modalModifCat">
            | </i>
          </button>

          <button type="button" class=" open-delete-cat btn btn-light" data-toggle="modal" data-id = "<?php echo $categories[$i]->getId()?" data-target="#modalDeleteCat">
            | </i>
          </button>
        </td>
      </tr>
    <?php
      }
    ?>
  </tbody>
</table>
```

La fonction qui permet de récupérer toute les catégories.

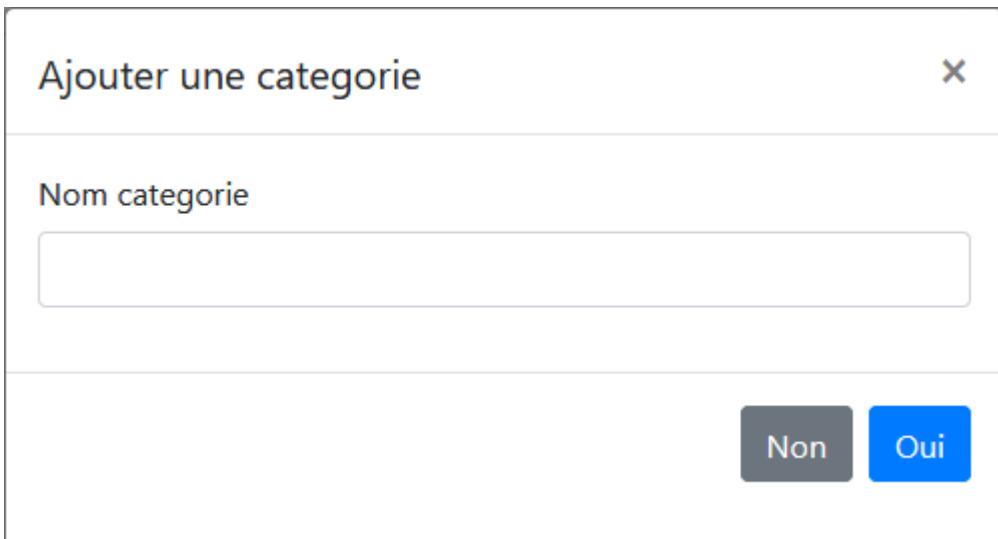
```
//recupere la liste des categories XX
public function listeCategorie()
{
  $DBase = new Connect();
  $db = $DBase->connexion();
  $categories = [];

  $q = $db->prepare('SELECT id_cat_prod,libelle from categorie_produit');
  $q->execute();
  while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
  {
    $categories[] = new Categorie($donnees);
  }

  return $categories;
}
```

Ajouter

Le modal qui permet d'ajouter une catégorie.



Le code du modal d'ajout d'une catégorie.

```
<div class="modal fade" id="modalAddCat" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">Ajouter une categorie</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <form action="../controllers/adminAjouter.php" method="post">
        <div class="modal-body">
          <div class="form-group">
            <label for="libelle">Nom categorie</label>
            <input class="form-control" type="text" id="libelle" name="CAlibelle" >
          </div>
        </div>
        <div class="modal-footer">
          <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
          <button type="submit" class="btn btn-primary">Oui </button>
        </div>
      </form>
    </div>
  </div>
</div>
```

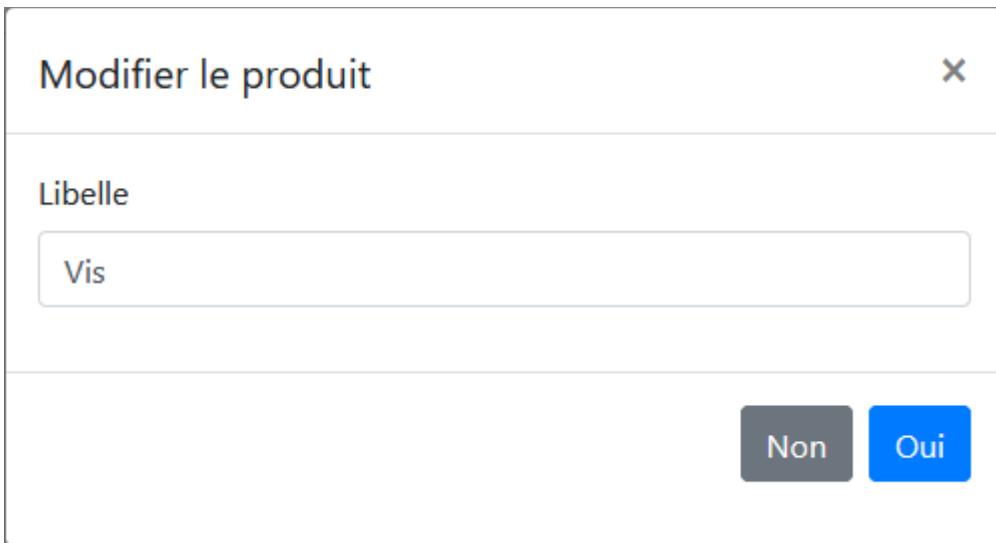
La fonction qui permet d'ajouter une catégorie.

```
if(isset($_POST['CAlibelle'])) {
    $Cat_man->addCategorie($_POST['CAlibelle']);
    header('location:../vue/dashboard.php');
}
```

```
public function addCategorie($libelle)
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('INSERT INTO categorie_produit (libelle) VALUES ("'.$libelle.'");');
    $q->execute();
}
```

Modifier

Le modal qui permet de modifier une catégorie.



La fonction javascript qui permet de transmettre les données au modal.

```
$(".open-modif-cat").click(function() {
    $("#modalModifCat").find("#CMid").val($(this).attr('data-id'));
    $("#modalModifCat").find("#CMlibelle").val($(this).attr('data-libelle'));

    $("#modalModifCat").modal("show");
});
```

Le code du modal de modification d'une catégorie.

```
<div class="modal fade" id="modalModifCat" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Modifier le produit</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <form action="../controllers/adminModifier.php" method="post" >
                <div class="modal-body">
                    <input type="hidden" name="CMid" id="CMid" value="">
                    <div class="form-group ">
                        <label for="CMlibelle">Libelle</label>
                        <input class="form-control" type="text" id="CMlibelle" name="CMlibelle">
                    </div>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                    <button type="submit" class="btn btn-primary">Oui </button>
                </div>
            </form>
        </div>
    </div>
</div>
```

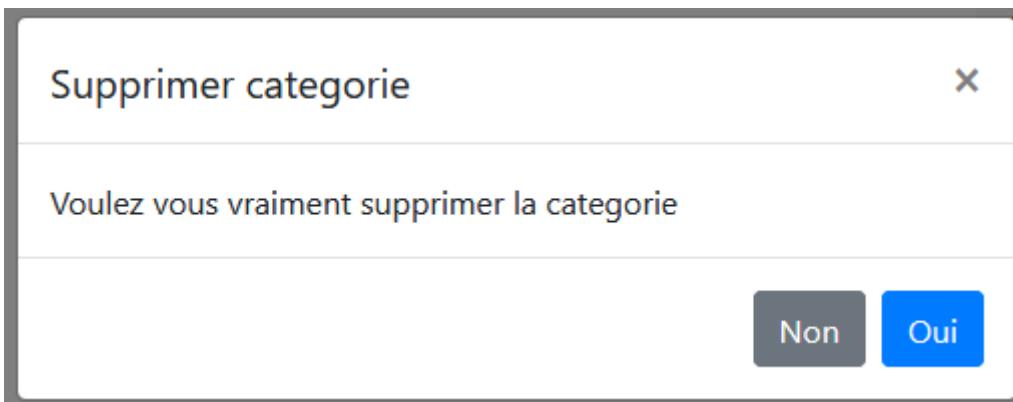
La fonction qui permet de modifier une catégorie.

```
if(isset($_POST['UMid']) && isset($_POST['UMcategorie'])) {
    $Cli_man->upUtilisateur($_POST['UMid'], $_POST['UMcategorie']);
    header('location:../vue/dashboard.php');
```

```
public function modifCategorie($id,$libelle)
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('UPDATE categorie_produit SET libelle = "'.$libelle.'" WHERE id_cat_prod = "'.$id.''");
    $q->execute();
}
```

Supprimer

Le modal qui permet de supprimer une catégorie.



La fonction Javascript qui permet de transmettre les données au modal.

```
$(".open-delete-cat").click(function() {
    $("#modalDeleteCat").find("#CDid").val($(this).attr('data-id'));
    $("#modalDeleteCat").modal("show");
});
```

Le code du modal de suppression d'une catégorie.

```
<div class="modal fade" id="modalDeleteCat" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Supprimer categorie</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                Voulez vous vraiment supprimer la categorie
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                <form action='../controllers/adminSupprimer.php' method="post">
                    <input type="hidden" name="CDid" id="CDid" value="">
                    <button type="submit" class="btn btn-primary">Oui </button>
                </form>
            </div>
        </div>
    </div>
</div>
```

La fonction qui permet de supprimer une catégorie.

```
if(isset($_POST['CDid'])){
    $Cat_man->deleteCategorie($_POST['CDid']);
    header('location:../vue/dashboard.php');
}
```

```
public function deleteCategorie($id)
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('DELETE FROM categorie_produit WHERE id_cat_prod = "'.$id.''");
    $q->execute();
}
```

Produit

On affiche un tableau de tous les produits avec leurs promotions s'ils en ont ou si ces promotions sont actives. Grâce au bouton dans le tableau on peut ajouter, modifier ou supprimer un produit mais aussi ajouter, modifier ou supprimer une promotion sur un produit.

#	Référence	Libellé	Prix Unitaire	Description	Quantité	Categorie	Image	+	Promotion	Date fin promotion	
17		Pneu Poulie	2 €	C'est une poulie pneu	0				30 %	2020-12-17	
2		Vis longue	9 €	Ceci est une vis longue.	0				-	-	
13		Poulie	2 €	Une poulie Simple	0				40 %	2020-12-12	

Le code d'affichage du tableau des produits.

```

$article = $Art_man->getList('%');
$articlePromo=$Art_man->getPromotionCategorie('%');

<?php
$y = 0;
for ($i=0; $i < $tailleArticle; $i++) {
?>
<tr>
<th scope="row"><?php echo $article[$i]->getId() ?></th>
<td><?php echo $article[$i]->getReference() ?></td>
<td><?php echo $article[$i]->getLibelle() ?></td>
<td><?php echo $article[$i]->getPrix() ?> €</td>
<td><?php echo $article[$i]->getDescription() ?></td>
<td><?php echo $article[$i]->getQuantite() ?></td>
<td><?php echo $article[$i]->getCategories() ?></td>
<td></td>
<td>
    <button type="button" class="open-modif-modal btn btn-light" data-toggle="modal" data-id = "<?php echo $article[$i]->getId() ?>" data-reference = "<?php echo $article[$i]->getReference() ?>" data-libelle = "<?php echo $article[$i]->getLibelle() ?>" data-prix = "<?php echo $article[$i]->getPrix() ?>" data-description = "<?php echo $article[$i]->getDescription() ?>" data-quantite = "<?php echo $article[$i]->getQuantite() ?>" data-categories = "<?php echo $article[$i]->getCategories() ?>" data-target="#modalModifProduit"
        <i class="fas fa-pencil-alt"></i>
    </button>
    <button type="button" class="open-delete-modal btn btn-light" data-toggle="modal" data-id = "<?php echo $article[$i]->getId() ?>" data-target="#modalDeleteProduit"
        <i class="far fa-times-circle"></i>
    </button>
</td>
<?php
    if($article[$i]->getId() == $articlePromo[$y]->getId()){
        <td><?php echo $articlePromo[$y]->getPourcentage() ?> %</td>
        <td><?php echo $articlePromo[$y]->getDateFin() ?></td>
        <td>
            <button type="button" class="open-delete-promo btn btn-dark" data-toggle="modal" data-id_produit="<?php echo $articlePromo[$y]->getId() ?>" data-id_promo = "<?php echo $articlePromo[$y]->getPromotion() ?>" data-target="#modalDeletePromo"
                <i class="fas fa-trash"></i>
            </button>
            <button type="button" class="open-modif-promo btn btn-dark" data-toggle="modal" data-id = "<?php echo $articlePromo[$y]->getPromotion() ?>" data-pourcentage = "<?php echo $articlePromo[$y]->getPourcentage() ?>" data-date = "<?php echo $articlePromo[$y]->getDateFin() ?>" data-target="#modalModifPromo"
                <i class="fas fa-edit"></i>
            </button>
        </td>
    <?php $y++;
    }
    else{
?>

        <td>-</td>
        <td>-</td>
        <td>
            <button type="button" class="open-add-promo btn btn-dark" data-toggle="modal" data-id_produit = "<?php echo $article[$i]->getId() ?>" data-libelle = "<?php echo $article[$i]->getLibelle() ?>" data-target="#modalAddPromo"
                <i class="fas fa-percentage"></i>
            </button>
        </td>
        <?php
        }
    ?>
</tr>
<?php
}
?>
</tbody>
</table>

```

La fonction qui permet de récupérer tous les produits.

```

//recupere la liste des categories
public function getList($categorie)
{
    $articles = [];
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('SELECT id_produit,produit.libelle,quantité,description,prix_unitaire,chemin,image,promotion,(categorie_produit.libelle) as categorie
    from objet
    join (produit join categorie_produit on categorie_produit.id_cat_prod = id_cat) on produit.id_produit = id_prod where id_cat LIKE "'.$categorie.'%"');
    $q->execute();
    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $articles[] = new Article($donnees);
    }
    return $articles;
}

```

La fonction qui permet de récupérer toutes les promotions.

```

//Recupere la liste des informations de la promotions pour les articles de la même catégorie
public function getPromotionCategorie($categorie)
{
    $promotions = [];
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('SELECT id_produit,produit.libelle,chemin,image,prix_unitaire,ROUND(prix_unitaire*(pourcentage/100),2)as prix,promotion,pourcentage,datefin from produit
    join categorie_produit on categorie_produit.id_cat_prod = id_cat
    join objet on objet.id_prod = id_produit
    join promotion on promotion.id = produit.promotion where CURDATE() between datedebut and datefin and categorie_produit.id_cat_prod LIKE "'.$categorie.'%"');
    $q->execute();
    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $promotions[] = new Article($donnees);
    }
    return $promotions;
}

```

Article

Ajouter

Le modal qui permet d'ajouter un produit.

The modal has the following structure:

- Reference:** Input field
- Libelle:** Input field
- Prix Unitaire:** Input field
- Description:** Text area
- Quantité:** Input field
- Categorie:** Select dropdown with options "Vis" and "Non"
- Image:** File input field with placeholder "Parcourir..." and message "Aucun fichier sélectionné."
- Action Buttons:** "Non" (grey) and "Oui" (blue)

Le code du modal d'ajout d'un produit.

```

<?php
    include_once('../models/CategorieManager.php');
    $categories=[];
    $Cat_man = new CategorieManager();
    $categories = $Cat_man->listeCategorie();
?>

<div class="modal fade" id="modalAddProduit" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Ajouter un produit</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div>
                <form action="../controllers/adminAjouter.php" method="post" enctype="multipart/form-data">
                    <div class="modal-body">
                        <div class="form-row">
                            <div class="form-group col-md-4">
                                <label for="Ireference">Reference</label>
                                <input class="form-control" type="text" id="Ireference" name="reference" >
                            </div>
                            <div class="form-group col-md-5">
                                <label for="Ilibelle">Libelle</label>
                                <input class="form-control" type="text" id="Ilibelle" name="libelle" >
                            </div>
                            <div class="form-group col-md-3">
                                <label for="Iprix">Prix Unitaire</label>
                                <input class="form-control" type="text" id="Iprix" name="prix" >
                            </div>
                        </div>
                        <div class="form-group">
                            <label for="Idescription">Description</label>
                            <textarea class="form-control" id="Idescription" rows="3" name="description"></textarea>
                        </div>
                        <div class="form-row">
                            <div class="form-group col-md-6">
                                <label for="Iquantite">Quantité</label>
                                <input class="form-control" type="text" id="Iquantite" name="quantite" >
                            </div>
                            <div class="form-group col-md-6">
                                <label for="Icategorie">Catégorie</label>
                                <select class="form-control" id="Icategorie" name="categorie">
                                    <?php
                                        for ($i=0; $i < $tailleCategories; $i++) {
                                            echo '<option value="'.$categories[$i]->getId().'">'.$categories[$i]->getLibelle() . '</option> ';
                                        }
                                    <?>
                                </select>
                            </div>
                        </div>
                        <div class="form-group">
                            <label for="Iimage">Image</label>
                            <input type="file" class="form-control-file" id="Iimage" name="image" >
                        </div>
                    </div>
                    <div class="modal-footer">
                        <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                        <button type="submit" class="btn btn-primary">Oui </button>
                    </div>
                </form>
            </div>
        </div>
    </div>
</div>

```

La fonction qui permet d'ajouter un produit et d'ajouter l'image du produit dans le dossier image du site.

```
public function addProduit($reference,$libelle,$prix,$description,$quantite,$categorie,$image,$chemin)
{
    $max = [];
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('INSERT INTO produit (reference,quantité,prix_unitaire,libelle,description,id_cat)
VALUES ("'.$reference.'","'.$quantite.'","'.$prix.'","'.$libelle.'","'.$description.'","'.$categorie.'")');
    $q->execute();
    $name = $db->lastInsertId();
    $r = $db->prepare('INSERT INTO objet (chemin,image,id_prod)
VALUES ("'.$chemin.'","'.$image.'","'.$name.'")');
    $r->execute();
}
```

Modifier

Le modal qui permet de modifier un produit.

Modifier le produit X

Reference	Libelle	Prix Unitaire
A701	Vis longue	9

Description

Ceci est une vis longue.

Quantité	Categorie
80	Vis

Non Oui

La fonction javascript qui permet de transmettre les données au modal.

```
$(".open-modif-modal").click(function() {
    $("#modalModifProduit").find("#Iid").val($(this).attr('data-id'));
    $("#modalModifProduit").find("#Ireference").val($(this).attr('data-reference'));
    $("#modalModifProduit").find("#Ilibelle").val($(this).attr('data-libelle'));
    $("#modalModifProduit").find("#Iprix").val($(this).attr('data-prix'));
    $("#modalModifProduit").find("#Idescription").val($(this).attr('data-description'));
    $("#modalModifProduit").find("#Iquantite").val($(this).attr('data-quantite'));
    $("#modalModifProduit").find("#Icategorie").text($(this).attr('data-categorie'));

    $("#modalModifProduit").modal("show");
});
```

Le code du modal de modification d'un produit.

```
<div class="modal fade" id="modalModifProduit" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">Modifier le produit</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <form action="/controllers/adminModifier.php" method="post" enctype="multipart/form-data">
        <div class="modal-body">
          <input type="hidden" name="Mid" id="Iid" value="">
          <div class="form-row">
            <div class="form-group col-md-4">
              <label for="Ireference">Reference</label>
              <input class="form-control" type="text" id="Ireference" name="Mreference" value="">
            </div>
            <div class="form-group col-md-5">
              <label for="Ilibelle">Libelle</label>
              <input class="form-control" type="text" id="Ilibelle" name="Mlibelle" value="">
            </div>
            <div class="form-group col-md-3">
              <label for="Iprix">Prix Unitaire</label>
              <input class="form-control" type="text" id="Iprix" name="Mprix" value="">
            </div>
          </div>
          <div class="form-group">
            <label for="Idescription">Description</label>
            <textarea class="form-control" id="Idescription" rows="3" name="Mdescription"></textarea>
          </div>
          <div class="form-row">
            <div class="form-group col-md-6">
              <label for="Iquantite">Quantité</label>
              <input class="form-control" type="text" id="Iquantite" name="Mquantite" value="">
            </div>
            <div class="form-group col-md-6">
              <label for="Icategorie">Categorie</label>
              <select class="form-control" id="Icategorie" name="Mcategorie">
                <?php
                  include_once('../models/CategorieManager.php');
                  $categories = [];
                  $Cat_man = new CategorieManager();
                  $categories = $Cat_man->listeCategorie();
                  $tailleCategories = sizeof($categories);
                  for ($i=0; $i < $tailleCategories; $i++) {
                    echo '<option value="'.$categories[$i]->getId().'">'.$categories[$i]->getLibelle() . '</option> ';
                  }
                <?
                </select>
            </div>
          </div>
          <div class="modal-footer">
            <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
            <button type="submit" class="btn btn-primary">Oui </button>
          </div>
        </form>
      </div>
    </div>
  </div>
```

La fonction qui permet de modifier un produit.

```
if(isset($_POST['Mreference']) && isset($_POST['Mlibelle']) && isset($_POST['Mprix']) &&
isset($_POST['Mdescription']) && isset($_POST['Mquantite']) && isset($_POST['Mid']) && isset($_POST['Mcategorie'])) {
  $Art_man->modifProduit($_POST['Mid'],$_POST['Mreference'],$_POST['Mlibelle'],$_POST['Mprix'],$_POST['Mdescription'],$_POST['Mquantite'],$_POST['Mcategorie']);
  header('location:../vue/dashboard.php');
}
```

```
public function modifProduit($id,$reference,$libelle,$prix,$description,$quantite,$idCat)
{
  $DBase = new Connect();
  $db = $DBase->connexion();
  $q = $db->prepare("UPDATE produit SET reference = '".$reference."', quantité = '".$quantite."', prix_unitaire = '".$prix."', libelle = '".$libelle."', description = '".$description."', id_cat = '".$idCat."' WHERE id_produit = '".$id.'');
  $q->execute();
}
```

Supprimer

Le modal qui permet de supprimer un produit.



La fonction javascript qui permet de transmettre les données au modal.

```
$(".open-delete-modal").click(function() {
    $("#modalDeleteProduit").find("#id_produit").val($(this).attr('data-id'));
    $("#modalDeleteProduit").modal("show");
});
```

Le code du modal de suppression d'un produit.

```
<div class="modal fade" id="modalDeleteProduit" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Supprimer produit</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                Voulez vous vraiment supprimer le produit
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                <form action='../../controllers/adminSupprimer.php' method="post">
                    <input type="hidden" name="id_produit" value="">
                    <button type="submit" class="btn btn-primary">Oui </button>
                </form>
            </div>
        </div>
    </div>
</div>
```

La fonction qui permet de supprimer un produit.

```
if(isset($_POST['id_produit'])){
    $Art_man->deleteProduit($_POST['id_produit']);
    header('location:../vue/dashboard.php');
}

//Supprime un produit
public function deleteProduit($id)
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('DELETE FROM produit WHERE id_produit = "'.$id.''");
    $q->execute();
    $p = $db->prepare('DELETE FROM objet WHERE id_prod = "'.$id.''");
    $p->execute();
}
```

Promotion

Ajouter

Le modal qui permet d'ajouter une promotion.

Ajouter une promotion X

Vis longue

Pourcentage

Date

Non Oui

Le code du modal d'ajout d'une promotion.

```
<div class="modal fade" id="modalAddPromo" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Ajouter une promotion</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <form action="../controllers/adminAjouter.php" method="post" enctype="multipart/form-data">
                <div class="modal-body">
                    <h3 id="PAproduit"></h3>
                    <input type="hidden" name="PAid_produit" id="PAid_produit" value="">
                    <div class="form-group">
                        <label for="pourcentage">Pourcentage</label>
                        <input class="form-control" type="text" id="PApourcentage" name="PApourcentage" >
                    </div>
                    <div class="form-group">
                        <label for="date">Date</label>
                        <input class="form-control" type="date" id="PAdate" name="PAdate" >
                    </div>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                    <button type="submit" class="btn btn-primary">Oui </button>
                </div>
            </form>
        </div>
    </div>
</div>
```

La fonction qui permet d'ajouter une promotion.

```
if(isset($_POST['PAid_produit']) && isset($_POST['PApourcentage']) && isset($_POST['PAdate']))
{
    $pourcentage = htmlspecialchars($_POST['PApourcentage']);
    $date = htmlspecialchars($_POST['PAdate']);
    $Art_man->addPromotion($_POST['PAid_produit'],$pourcentage,$date);
    header('location:../vue/dashboard.php');
}

public function addPromotion($id_produit,$pourcentage,$datefin)
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('INSERT INTO promotion (pourcentage, datedebut, datefin) VALUES ("'.$pourcentage.'",CURDATE(),"'.$datefin.'")');
    $q->execute();
    $name = $db->lastInsertId();
    $p = $db->prepare('UPDATE produit SET promotion="'.$name.'" WHERE id_produit = "'.$id_produit.'"' );
    $p->execute();
}
```

Modifier

Le modal qui permet de modifier une promotion.



La fonction javascript qui permet de transmettre les données au modal.

```
$(".open-modif-promo").click(function() {
    $("#modalModifPromo").find("#PMid_promo").val($(this).attr('data-id'));
    $("#modalModifPromo").find("#PMpourcentage").val($(this).attr('data-pourcentage'));
    $("#modalModifPromo").find("#PMdate").val($(this).attr('data-date'));

    $("#modalModifPromo").modal("show");
});
```

Le code du modal de modification d'une promotion.

```
<div class="modal fade" id="modalModifPromo" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Modifier une promotion</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <form action="../../controllers/adminModifier.php" method="post" enctype="multipart/form-data">
                <div class="modal-body">
                    <input type="hidden" name="PMid_promo" id="PMid_promo" value="">
                    <div class="form-group">
                        <label for="pourcentage">Pourcentage</label>
                        <input class="form-control" type="text" id="PMpourcentage" name="PMpourcentage" >
                    </div>
                    <div class="form-group">
                        <label for="date">Date</label>
                        <input class="form-control" type="date" id="PMdate" name="PMdate" >
                    </div>
                </div>
                <div class="modal-footer">
                    <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                    <button type="submit" class="btn btn-primary">Oui </button>
                </div>
            </form>
        </div>
    </div>
</div>
```

La fonction qui permet de modifier une promotion.

```
if(isset($_POST['PMid_promo']) && isset($_POST['PMpourcentage']) && isset($_POST['PMdate']))
{
    $Art_man->modifPromotion($_POST['PMid_promo'],$_POST['PMpourcentage'],$_POST['PMdate']);
    header('location:../vue/dashboard.php');
}

public function modifPromotion($id_promo,$pourcentage,$datefin)
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('UPDATE promotion SET pourcentage="'. $pourcentage .'" , datefin="'. $datefin .'" WHERE id = "'.$id_promo.'"' );
    $q->execute();
}
```

Supprimer

Le modal qui permet de supprimer une promotion.



La fonction javascript qui permet de transmettre les données au modal.

```
$(".open-delete-promo").click(function() {
    $("#modalDeletePromo").find("#PDid_produit").val($(this).attr('data-id_produit'));
    $("#modalDeletePromo").find("#PDid_promo").val($(this).attr('data-id_promo'));
    $("#modalDeletePromo").modal("show");
});
```

Le code du modal de suppression d'une promotion.

```
<div class="modal fade" id="modalDeletePromo" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Supprimer promotion</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                Voulez vous vraiment supprimer cette promotion
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                <form action='../controllers/adminSupprimer.php' method="post">
                    <input type="hidden" name="PDid_produit" id="PDid_produit" value="">
                    <input type="hidden" name="PDid_promo" id="PDid_promo" value="">
                    <button type="submit" class="btn btn-primary">Oui </button>
                </form>
            </div>
        </div>
    </div>
</div>
```

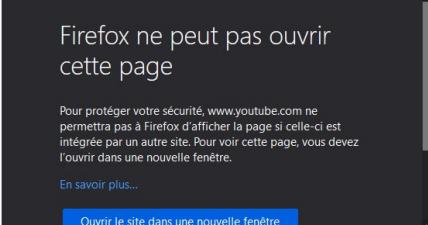
La fonction qui permet de supprimer une promotion.

```
if(isset($_POST['PDid_promo']) && isset($_POST['PDid_produit']))
{
    $Art_man->deletePromotion($_POST['PDid_promo'],$_POST['PDid_produit']);
    header('location:../vue/dashboard.php');
}
```

```
public function deletePromotion($id_promo,$id_produit)
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('DELETE FROM promotion WHERE id = "'.$id_promo.''");
    $q->execute();
    $p = $db->prepare('UPDATE produit SET promotion = NULL WHERE id_produit = "'.$id_produit.''");
    $p->execute();
}
```

Conseil

On affiche un tableau de tous les conseils avec des boutons pour ajouter, modifier ou supprimer des conseils.

#	Titre	Libelle	Video
1	Comment visser une vis	Avant de procéder au vissage, appliquez de la paraffine ou de la bougie sur le filet de la vis pour faciliter sa pénétration. Vissez progressivement : adaptez la vitesse de vissage au fur et à mesure. Regardez ce tuto pour visser comme un pro	 Comment visser facilement une vis ? A regarder ... Partager
2	Quelle clé pour serrer un écrou, un raccord, ou un boulon ?	Vidéo d'aide	 Quelle clé choisir pour la plomberie ? A regarder ... Partager
5	Comment mettre une cheville	Video testsqdqsd	 Firefox ne peut pas ouvrir cette page Pour protéger votre sécurité, www.youtube.com ne permettra pas à Firefox d'afficher la page si celle-ci est intégrée par un autre site. Pour voir cette page, vous devez l'ouvrir dans une nouvelle fenêtre. En savoir plus... Ouvrir le site dans une nouvelle fenêtre

Le code d'affichage du tableau des conseils.

```
<table class="table">
  <thead class="thead-dark">
    <tr>
      <th scope="col">#</th>
      <th scope="col">Titre</th>
      <th scope="col">Libelle</th>
      <th scope="col">Video</th>
      <th scope="col">
        <button type="button" class="btn btn-dark" data-toggle="modal" data-target="#modalAddCons">
          <i class="fas fa-plus"></i>
        </button>
      </th>
    </tr>
  </thead>
  <tbody>
    <?php
      for ($i=0; $i < $tailleConseil; $i++) {
    ?>
    <tr>
      <td><?php echo $conseils[$i]->getId() ?></td>
      <td><?php echo $conseils[$i]->getTitre() ?></td>
      <td><?php echo $conseils[$i]->getLibelle() ?></td>
      <td><iframe width="560" height="315" src=<?php echo $conseils[$i]->getVideo()?> frameborder="0"
        allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture" allowfullscreen class="align-self-start mr-3"></td>
      <td>
        <button type="button" class="open-modif-cons btn btn-light" data-toggle="modal" data-id = "<?php echo $conseils[$i]->getId()?>" data-titre = "<?php echo $conseils[$i]->getTitre() ?>" data-libelle = "<?php echo $conseils[$i]->getLibelle() ?>" data-video = "<?php echo $conseils[$i]->getVideo()?>" data-target="#modalModifCons">
          <i class="fas fa-pencil-alt"></i>
        </button>

        <button type="button" class="open-delete-cons btn btn-light" data-toggle="modal" data-id = "<?php echo $conseils[$i]->getId()?>" data-target="#modalDeleteCons">
          <i class="far fa-times-circle"></i>
        </button>
      </td>
    </tr>
    <?php
      }
    ?>
  </tbody>
</table>
```

La fonction qui permet de récupérer tous les conseil.

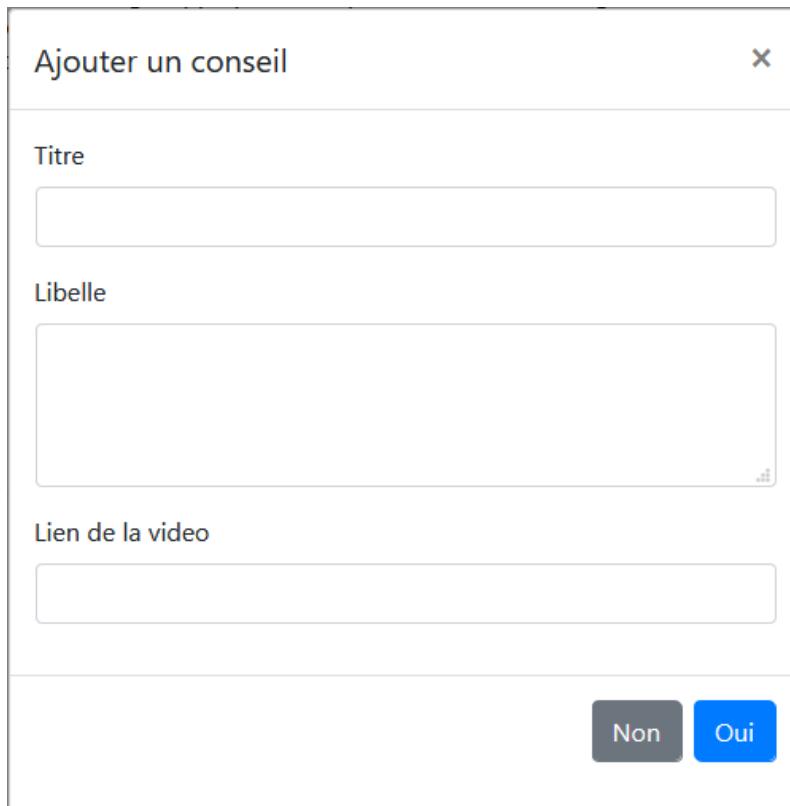
```
//recupere la liste des conseils XX
public function listeConseils() {
  $DBase = new Connect();
  $db = $DBase->connexion();
  $listConseils = [];

  $q = $db->prepare('SELECT * from conseils');
  $q->execute();
  while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
  {
    $listConseils[] = new Conseil($donnees);
  }

  return $listConseils;
}
```

Ajouter

Le modal qui permet d'ajouter un conseil.



Le code du modal d'ajout d'un conseil.

```
<div class="modal fade" id="modalAddCons" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">Ajouter un conseil</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <form action="../controllers/adminAjouter.php" method="post" enctype="multipart/form-data">
        <div class="modal-body">
          <div class="form-group">
            <label for="Ireference">Titre</label>
            <input class="form-control" type="text" id="Ititre" name="CO-Atitre" >
          </div>
          <div class="form-group">
            <label for="Ilibelle">Libelle</label>
            <textarea class="form-control" rows="3" id="Ilibelle" name="CO-Alibelle"></textarea>
          </div>
          <div class="form-group">
            <label for="Ivideo">Lien de la video</label>
            <input class="form-control" type="text" id="Ivideo" name="CO-Avideo" >
          </div>
        </div>
        <div class="modal-footer">
          <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
          <button type="submit" class="btn btn-primary">Oui </button>
        </div>
      </form>
    </div>
  </div>
</div>
```

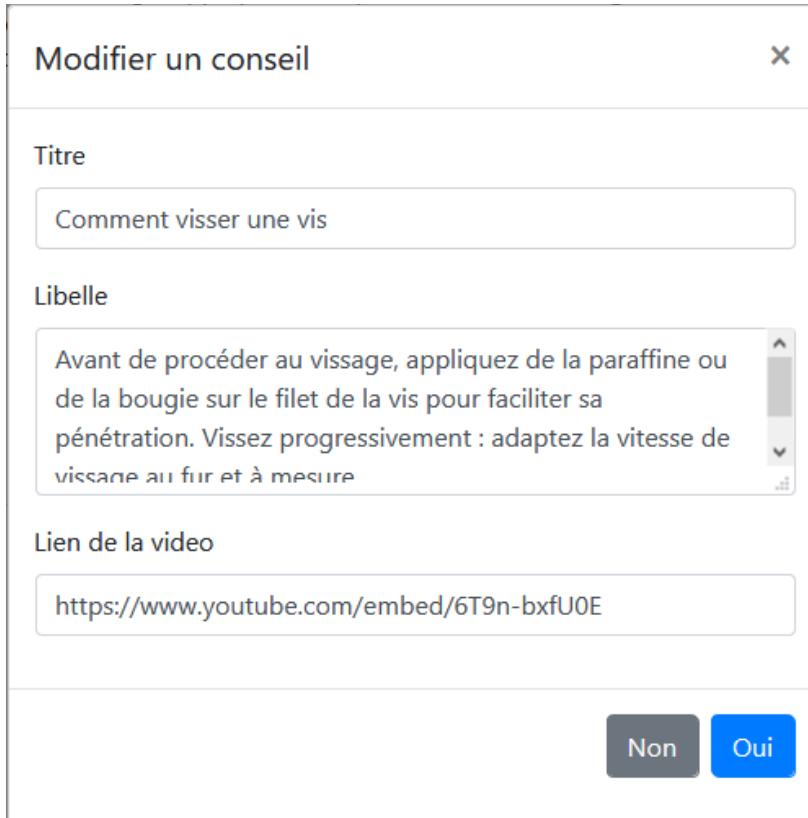
La fonction qui permet d'ajouter un conseil.

```
if(isset($_POST['CO-Atitre']) && isset($_POST['CO-Alibelle']) && isset($_POST['CO-Avideo'])) {
    $Conseil_man->addConseil($_POST['CO-Atitre'],$_POST['CO-Alibelle'],$_POST['CO-Avideo']);
    header('location:../vue/dashboard.php');
}

public function addConseil($titre,$libelle,$video){
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('INSERT INTO conseils(titre, libelle, video) VALUES ("'.$titre.'","'.$libelle.'","'.$video.'")');
    $q->execute();
}
```

Modifier

Le modal qui permet de modifier un conseil.



La fonction Javascript qui permet de transmettre les données au modal.

```
$(".open-modif-cons").click(function() {
    $("#modalModifCons").find("#id").val($(this).attr('data-id'));
    $("#modalModifCons").find("#titre").val($(this).attr('data-titre'));
    $("#modalModifCons").find("#libelle").val($(this).attr('data-libelle'));
    $("#modalModifCons").find("#video").val($(this).attr('data-video'));

    $("#modalModifCons").modal("show");
});
```

Le code du modal de modification d'un conseil.

```
<div class="modal fade" id="modalModifCons" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title" id="exampleModalLabel">Modifier un conseil</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <form action="../controllers/adminModifier.php" method="post" enctype="multipart/form-data">
        <div class="modal-body">
          <input type="hidden" name="CO-Mid" id="id" value="">
          <div class="form-group">
            <label for="Ireference">Titre</label>
            <input class="form-control" type="text" id="titre" name="CO-Mtitre" >
          </div>
          <div class="form-group">
            <label for="llibelle">Libelle</label>
            <textarea class="form-control" rows="3" id="libelle" name="CO-Mlibelle"></textarea>
          </div>
          <div class="form-group">
            <label for="Ivideo">Lien de la video</label>
            <input class="form-control" type="text" id="video" name="CO-Mvideo" >
          </div>
        </div>
        <div class="modal-footer">
          <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
          <button type="submit" class="btn btn-primary">Oui </button>
        </div>
      </form>
    </div>
  </div>
</div>
```

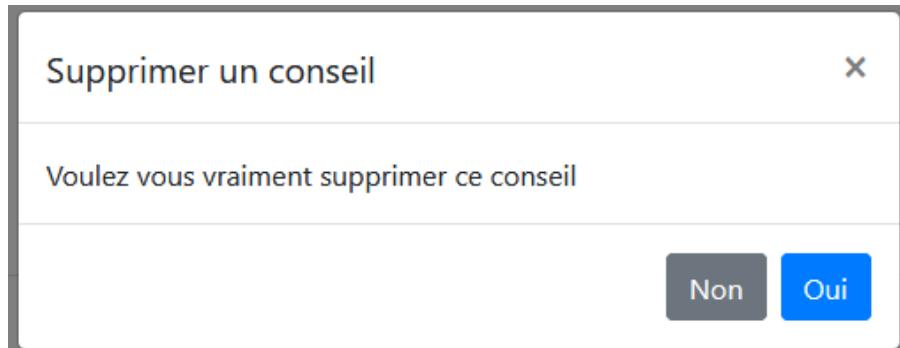
La fonction qui permet de modifier un conseil.

```
if(isset($_POST['CO-Mid']) && isset($_POST['CO-Mtitre']) && isset($_POST['CO-Mlibelle']) && isset($_POST['CO-Mvideo'])) {
    $Conseil_man->modifConseil($_POST['CO-Mid'],$_POST['CO-Mtitre'],$_POST['CO-Mlibelle'],$_POST['CO-Mvideo']);
    header('location:../vue/dashboard.php');
}
```

```
public function modifConseil($id,$titre,$libelle,$video){
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('UPDATE conseils SET titre="'.$titre.'",libelle="'.$libelle.'",video ="'.$video.'" WHERE id_cons = "'.$id.'"' );
    $q->execute();
}
```

Supprimer

Le modal qui permet de supprimer un conseil.



La fonction javascript qui permet de transmettre les données au modal.

```
$(".open-delete-cons").click(function() {
    $("#modalDeleteCons").find("#id").val($(this).attr('data-id'));
    $("#modalDeleteCons").modal("show");
});
```

Le code du modal de suppression d'un conseil.

```
<div class="modal fade" id="modalDeleteCons" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Supprimer un conseil</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <div class="modal-body">
                Voulez vous vraiment supprimer ce conseil
            </div>
            <div class="modal-footer">
                <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                <form action='../controllers/adminSupprimer.php' method="post">
                    <input type="hidden" name="CO-Did" id="id" value="">
                    <button type="submit" class="btn btn-primary">Oui </button>
                </form>
            </div>
        </div>
    </div>
</div>
```

La fonction qui permet de supprimer un conseil.

```
if(isset($_POST['CO-Did'])){
    $Conseil_man->deleteConseil($_POST['CO-Did']);
    header('location:../vue/dashboard.php');
}

public function deleteConseil($id){
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('DELETE FROM conseils WHERE id_cons = "'.$id.''");
    $q->execute();
}
```

Utilisateur

On affiche un tableau de tous les utilisateurs avec un bouton pour chaque utilisateur, pour pouvoir changer le grade.

#	Nom	Prénom	Email	Adresse	Code Postal	Ville	Téléphone	Catégorie	
1234567891	Six	Tristan	sixtristan@orange.fr	dsdq	59139	Wattignies	26522156	Administrateur	
1234567892	Florentin	Duyck	misterfloflo59@gmail.com	44 rue Massena	59199	Hergnies	767689182	Administrateur	

Le code d'affichage du tableau des utilisateurs.

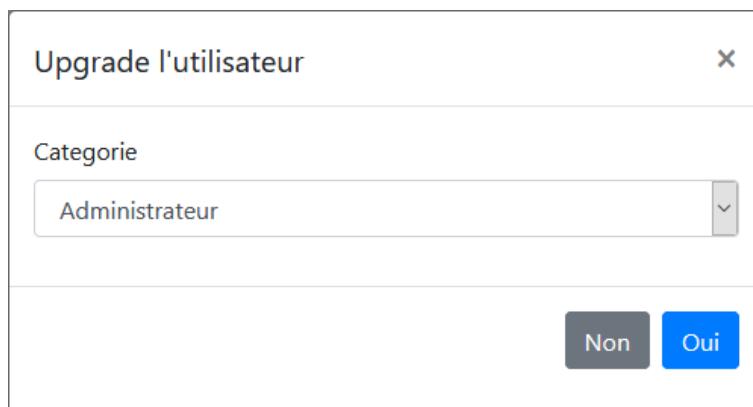
```
<table class="table">
    <thead class="thead-dark">
        <tr>
            <th scope="col">#</th>
            <th scope="col">Nom</th>
            <th scope="col">Prénom</th>
            <th scope="col">Email</th>
            <th scope="col">Adresse</th>
            <th scope="col">Code Postal</th>
            <th scope="col">Ville</th>
            <th scope="col">Téléphone</th>
            <th scope="col">Catégorie</th>
            <th scope="col"></th>
        </tr>
    </thead>
    <tbody>
        <?php
            for ($i=0; $i < $tailleClient; $i++) {
        ?>
        <tr>
            <th scope="row"><?php echo $clients[$i]->getId()?></th>
            <td><?php echo $clients[$i]->getNom()?></td>
            <td><?php echo $clients[$i]->getPrenom()?></td>
            <td><?php echo $clients[$i]->getEmail()?></td>
            <td><?php echo $clients[$i]->getAdresse()?></td>
            <td><?php echo $clients[$i]->getCp()?></td>
            <td><?php echo $clients[$i]->getVille()?></td>
            <td><?php echo $clients[$i]->getTel()?></td>
            <td><?php echo $clients[$i]->getCatégorie()?></td>
            <td>
                <button type="button" class="open-up-Ut btn btn-light" data-toggle="modal" data-id = "<?php echo $clients[$i]->getId()?" data-id-cat = "<?php echo $clients[$i]->getId_cat() ?>" data-target="#modalUpUt">
                    <i class="fas fa-arrow-up"></i>
                </button>
            </td>
        </tr>
        <?php
            }
        ?>
    </tbody>
</table>
```

La fonction qui permet de récupérer tous les utilisateurs.

```
public function listClient()
{
    $clients = [];
    $DBase = new Connect();
    $db = $DBase->connexion();
    $q = $db->prepare('SELECT id,Nom,Prenom,email,adresse,codepostal,ville,tel,categorie_utilisateur.libellé, categorie_utilisateur.id_cat FROM utilisateur
JOIN categorie_utilisateur ON utilisateur.id_cat = categorie_utilisateur.id_cat ');
    $q->execute();
    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $clients[] = new Client($donnees);
    }
    return $clients;
}
```

Augmentation de grade

Le modal qui permet de modifier le grade de l'utilisateur.



La fonction javascript qui permet de transmettre les données au modal.

```
<script>
    $(document).ready(function() {
        $(".open-up-Ut").click(function() {
            $("#modalUpUt").find("#UMid").val($(this).attr('data-id'));
            $("#modalUpUt").find("#UMcategorie").val($(this).attr('data-id-cat'));

            $("#modalUpUt").modal("show");
        });
    });
</script>
```

Le code du modal de modification du grade de l'utilisateur.

```
<?php
    include_once('../models/ClientManager.php');
    $Ucategories=[];
    $Ucat_man = new clientManager();
    $Ucategories = $Ucat_man->listEUCategories();
    $tailleCategories = sizeof($Ucategories);
?>

<div class="modal fade" id="modalUpUt" tabindex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h5 class="modal-title" id="exampleModalLabel">Upgrade l'utilisateur</h5>
                <button type="button" class="close" data-dismiss="modal" aria-label="Close">
                    <span aria-hidden="true">&times;</span>
                </button>
            </div>
            <form action="../controllers/adminModifier.php" method="post" >
                <div class="modal-body">
                    <input type="hidden" name="UMid" id="UMid" value="">
                    <div class="form-group ">
                        <label for="UMcategorie">Categorie</label>
                        <select class="form-control" id="UMcategorie" name="UMcategorie">
                            <?php
                                for ($i=0; $i < $tailleCategories; $i++) {
                                    echo '<option value="'.$Ucategories[$i]->getId().'">'.$Ucategories[$i]->getLibelle() . '</option> ';
                                }
                            <?>
                        </select>
                    </div>
                    <div class="modal-footer">
                        <button type="button" class="btn btn-secondary" data-dismiss="modal">Non</button>
                        <button type="submit" class="btn btn-primary">Oui </button>
                    </div>
                </div>
            </form>
        </div>
    </div>
</div>
```

La fonction qui permet de modifier le grade de l'utilisateur.

```
if(isset($_POST['UMid']) && isset($_POST['UMcategorie'])){  
    $Cli_man->upUtilisateur($_POST['UMid'] , $_POST['UMcategorie']);  
    header('location:../vue/dashboard.php');  
}
```

```
public function upUtilisateur($id,$id_cat)  
{  
    $DBase = new Connect();  
    $db = $DBase->connexion();  
    $q = $db->prepare('UPDATE utilisateur SET id_cat="'.$id_cat.'" WHERE id = "'.$id.'");  
    $q->execute();  
}
```

Commande

On affiche un tableau de toutes les commandes effectuées avec le nom du client et son mail et quel produit il a commandé.

#	Quantité	Date	Nom	Prénom	Email	Référence	Libelle	Prix
1	10	2020-11-18	Six	Tristan	sixtristan@orange.fr	A701	Vis longue	90 €

Le code d'affichage du tableau des commandes.

```
<div class="table-responsive-sm m-2 mt-5">  
    <table class="table">  
        <thead class="thead-dark">  
            <tr>  
                <th scope="col">#</th>  
                <th scope="col">Quantité</th>  
                <th scope="col">Date</th>  
                <th scope="col">Nom</th>  
                <th scope="col">Prénom</th>  
                <th scope="col">Email</th>  
                <th scope="col">Référence</th>  
                <th scope="col">Libelle</th>  
                <th scope="col">Prix</th>  
            </tr>  
        </thead>  
        <tbody>  
        <?php  
            for ($i=0; $i < $tailleCommande; $i++) {  
        ?>  
            <tr>  
                <th scope="row"><?php echo $commandes[$i]->getId()?></th>  
                <td><?php echo $commandes[$i]->getQuantite()?></td>  
                <td><?php echo $commandes[$i]->getDate()?></td>  
                <td><?php echo $commandes[$i]->getNom()?></td>  
                <td><?php echo $commandes[$i]->getPrenom()?></td>  
                <td><?php echo $commandes[$i]->getEmail()?></td>  
                <td><?php echo $commandes[$i]->getReference()?></td>  
                <td><?php echo $commandes[$i]->getlibelle()?></td>  
                <td><?php echo $commandes[$i]->getprix()?> €</td>  
            <?php  
            }  
        ?>  
        </tbody>  
    </table>  
</div>
```

La fonction qui permet de récupérer tout les commandes effectuées.

```
//recupere la liste des categories XX
public function listeCommande()
{
    $DBase = new Connect();
    $db = $DBase->connexion();
    $commandes = [];

    $q = $db->prepare('SELECT id_commande, commande.quantité, date, utilisateur.Nom, utilisateur.Prenom, utilisateur.email, produit.reference, produit.libelle,
    (produit.prix_unitaire * commande.quantité) AS prix FROM commande
    JOIN utilisateur ON commande.id_utilisateur = utilisateur.id
    JOIN produit ON commande.id_produit = produit.id_produit ');
    $q->execute();
    while ($donnees = $q->fetch(PDO::FETCH_ASSOC))
    {
        $commandes[] = new Commande($donnees);
    }

    return $commandes;
}
```

Réalisation technique

Les langages utilisés :

HTML5 : Langage de balisage conçu pour représenter les pages web. C'est un langage permettant d'écrire de l'hypertexte, d'où son nom. HTML permet également de structurer sémantiquement et logiquement et de mettre en forme le contenu des pages, d'inclure des ressources multimédias dont des images, des formulaires de saisie et des programmes informatiques. Il permet de créer des documents. Il est souvent utilisé conjointement avec le langage de programmation JavaScript et des feuilles de style en cascade (CSS).

CSS3 : Afin de manipuler la présentation, nous avons utilisé des feuilles de style CSS.

BOOTSTRAP : Bootstrap est un ensemble de fichiers CSS et JavaScript fonctionnant ensemble et qu'on va pouvoir utiliser pour créer des designs complexes de manière relativement simple et rapide de plus cette solution est gratuite donc privilégiée. Il nous permet d'économiser un temps précieux lors de la réalisation des interfaces de nos pages.

PHP7 : Pour le côté serveur, nous avons choisi le langage PHP, ceci pour plusieurs raisons : Tout d'abord, le PHP gère très bien les requêtes SQL. Ce qui est important pour manipuler les résultats de requêtes SQL.

Enfin, le PHP est un des langages côté serveur le plus utilisé, il existe une grande communauté d'utilisateurs de PHP, ce qui en cas de problème peut vite être résolu par des tutos sur YouTube ou des forums.

SQL : L'association PHP/MySQL est la plus utilisée. C'est pourquoi nous avons choisi le serveur de base de données MySQL contenue dans l'application 3 en 1 qu'est WAMP qui offre une grande flexibilité. Il intègre de nombreuses fonctionnalités et assure une bonne sécurité des données.

JAVASCRIPT : Javascript permet de créer des sites internet dynamiques. Javascript permet donc d'optimiser au maximum l'expérience de l'utilisateur par l'insertion de script dans nos pages PHP, comme recharger des requêtes pour actualiser nos éléments sans recharger la page.

Organisation

Nous nous sommes répartis le travail par secteur chacun s'occupait d'un morceau du site telle que le style, la base de données ou encore l'utilisateur. Nous regroupons notre travail grâce à un dépôt GIT.

Résultats

Les résultats sont atteints sauf pour les commandes, les factures et les paiements.

Les exigences non fonctionnelles qui assurent la qualité et la performance du site ont toutes été développées. L'ergonomie du site est satisfaisante pour les clients.

Améliorations envisagées

Ajouter un système de gestion des factures

Ajouter les paiements des clients

La sauvegarde des factures

Un système de fidélité