

MicroWorld

2022

Documentation technique

Fait par Monchau Victoire Ioamra

Sommaire

1. Présentation générale.....	3
1.1. Contexte.....	3
1.2. Expression des besoins.....	3
1.2.a. Objectifs.....	3
1.2.b. Cibles et caractéristiques.....	3
1.2.c. Description du contenu.....	3
1.3. Exigences fonctionnelles.....	4
1.3.a. Principales composantes :.....	4
1.3.b. Fonctionnalités Front.....	4
1.3.c. Fonctionnalités Back.....	4
2. Analyse.....	5
2.1. MCD / MLD.....	5
2.2. Diagramme de cas d'utilisation.....	5
2.3. Diagramme de classe.....	6
3. Environnement de développement.....	6
4. Les classes et leur méthode.....	7
4.1. Les managers.....	7
4.1.a. AvisClientManager.....	7
4.1.b. CategorieManager.....	8
4.1.c. ClientManager.....	8
4.1.d. CommandeManager.....	9
4.1.e. HistoriqueProduitManager.....	9
4.1.f. ImageProduitManager.....	10
4.1.g. LigneDeCommandeManager.....	10
4.1.h. ProduitManager.....	10
4.2. Connexion à la base de données et autoload.....	11
5. présentation du site.....	12
5.1. page d'accueil.....	12
5.2. Les catégories.....	13
5.3. Page de connexion.....	15
5.4. Page d'inscription.....	16
5.5. Le panier.....	17
5.6. Le profil client.....	18
5.7. Ajouter un produit.....	19
5.8. Modifier un produit.....	20
5.9. Connexion administrateur.....	21
5.10. L'api.....	21
5.10.a. La validation d'un achat.....	21
5.10.b. La recherche de produit par catégorie.....	22
5.10.c. Le changement de disponibilité d'un produit.....	22
5.11. Programmation coté serveur.....	23

1. Présentation générale

1.1. Contexte

La Société Microworld est une entreprise de vente/ réparation de matériel informatique récemment racheté par un entrepreneur. Cette entreprise embauche une vingtaine de personnes réparties en plusieurs équipes centrées sur des tâches différentes.

Ses principales activités sont les suivantes :

- La réparation de matériel informatique
- La vente de matériel informatique
- La prospection de clients
- La prestation de services aux entreprises (maintenance, installation)

Avant le rachat de l'entreprise, cette dernière conservait ses archives uniquement sous format papier.

Le nouveau propriétaire souhaite une évolution avec l'utilisation de solutions logicielles pour rendre son entreprise plus moderne et plus dynamique. Il entreprend de créer une visibilité en ligne par le biais d'un site en ligne pour l'extérieur. Aussi, il souhaite pouvoir offrir les possibilités de se coordonner en interne par le biais de ce même site.

1.2. Expression des besoins

1.2.a. Objectifs

- Vente en ligne de matériel informatique
- Attirer de nouveaux clients. Augmenter le CA et les bénéfices.

1.2.b. Cibles et caractéristiques

- Particuliers (achats de matériel informatique en ligne)

1.2.c. Description du contenu

- Enregistrement sur le site en tant que client.
- Catalogue des objets à vendre classé par catégories.
- Pour chaque catégorie les objets en vente peuvent être recherchés par leur nom, et trier par leur nom, prix et note.
- Gestion du caddie.
- Les objets achetés laissent un historique de commandes pour les clients authentifiés.
- Les employés peuvent ajouter des produits en vente, les modifier et changer leur disponibilité.
- Un client qui a acheté un produit peut le noter et poser un avis.
- Les informations des utilisateurs doivent être modifiable à tous moment.

1.3. Exigences fonctionnelles

1.3.a. Principales composantes :

- Identification des utilisateurs
- Gestion des commandes
- Gestion des avis
- Gestion du caddie
- Gestion de l'espace client
- Gestion du catalogue

1.3.b. Fonctionnalités Front

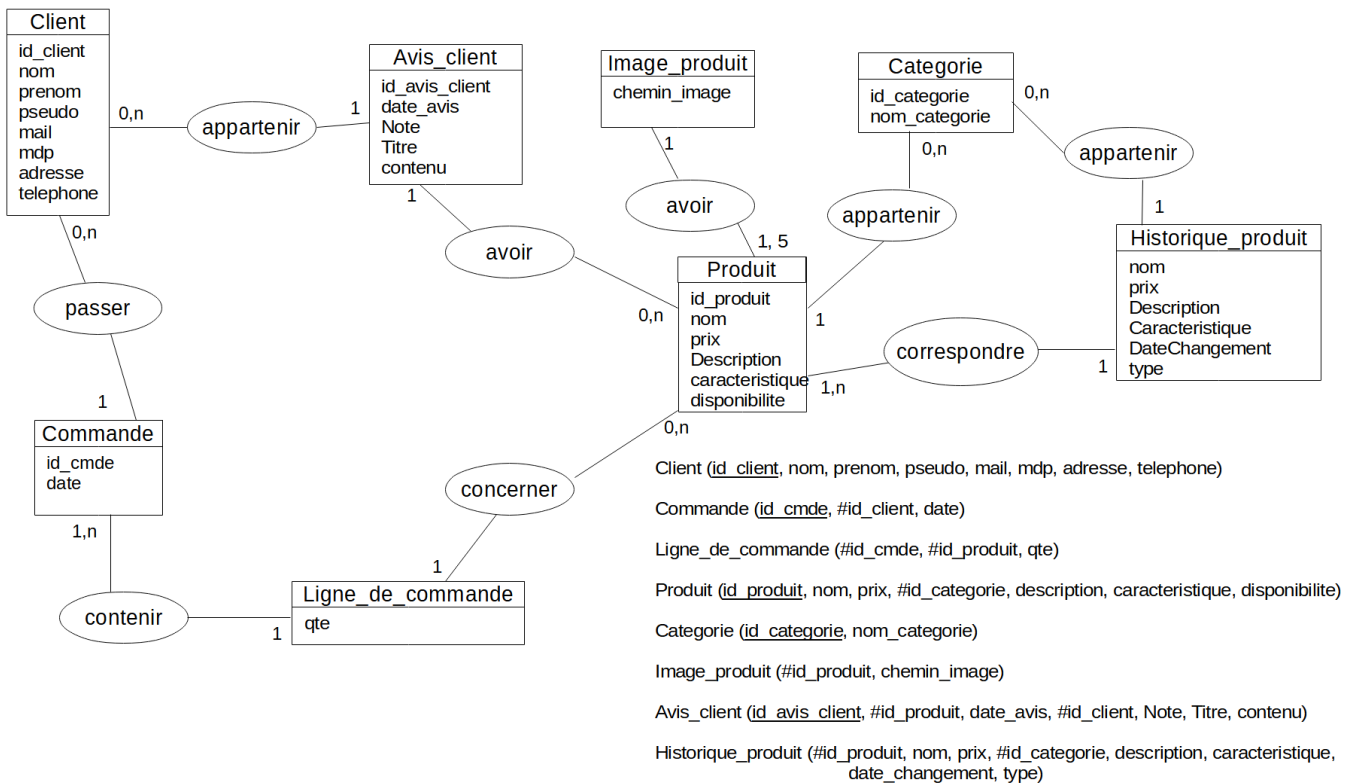
- Identification
- Gestion du catalogue
- Gestion de l'espace client
- Gestion des commandes
- Gestion des avis

1.3.c. Fonctionnalités Back

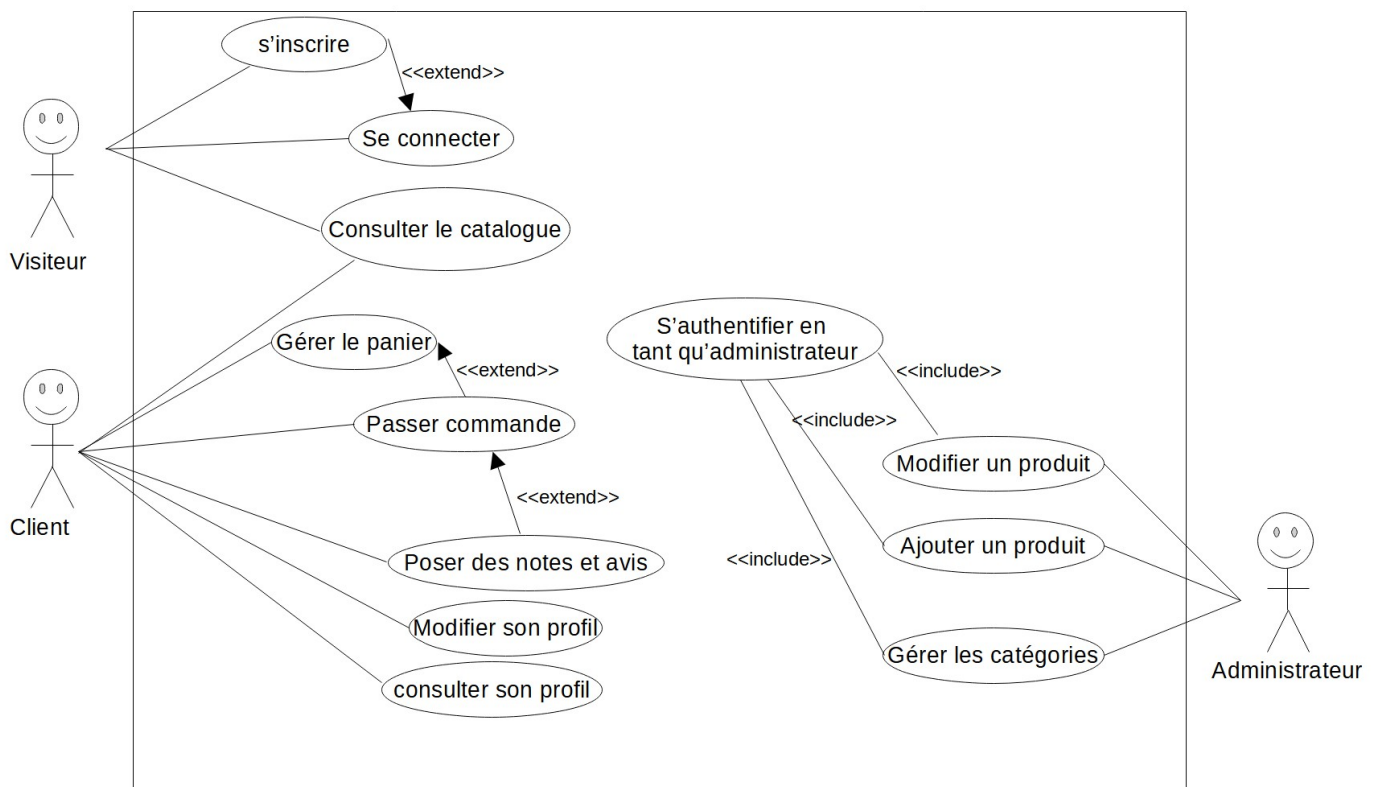
- Gestion des utilisateurs
- Gestion du contenu informationnel
- Gestion du catalogue
- Historique des changement sur les produits

2. Analyse

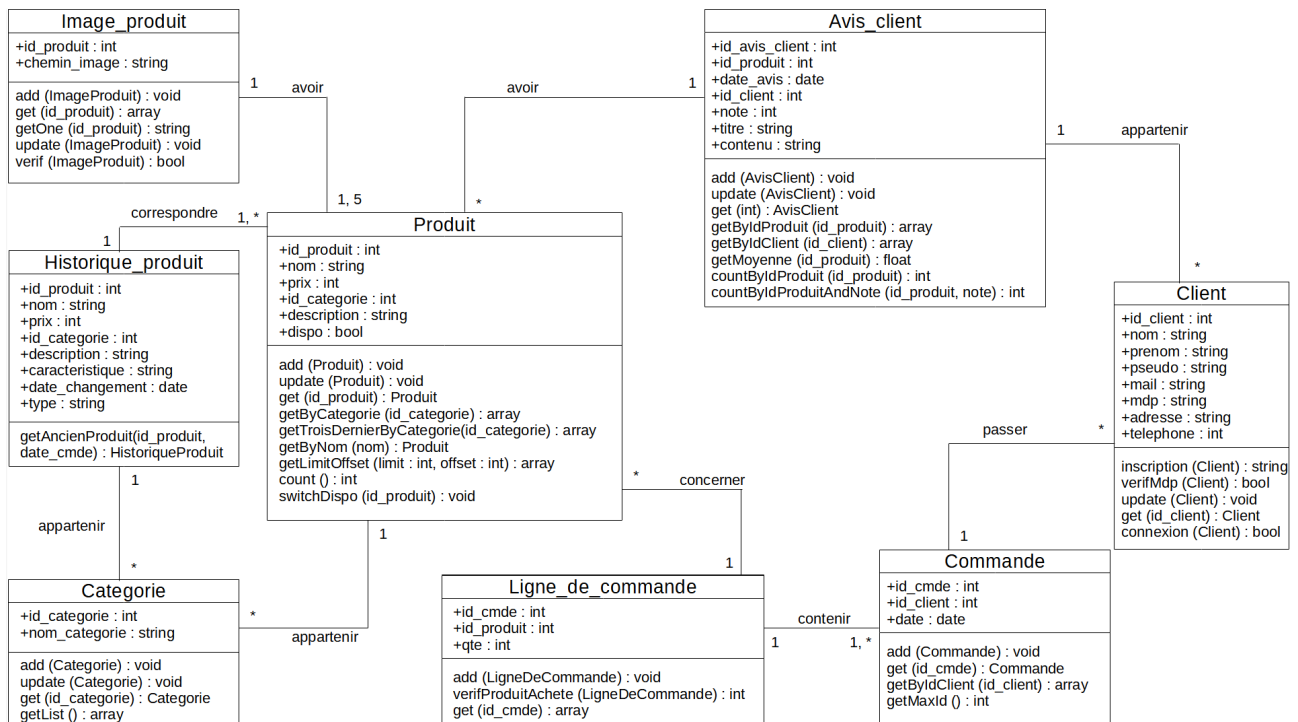
2.1. MCD / MLD



2.2. Diagramme de cas d'utilisation



2.3. Diagramme de classe



3. Environnement de développement

Langage utilisé : HTML , CSS, Javascript, PHP, SQL

Framework/Librairie : JQuery, Bootstrap 5, DataTables 1.11.5

4. Les classes et leur méthode

Chaque table de la base de données à 2 classes associées.

Exemple :

La table produit correspond aux fichiers

produit.class.php et produitManager.class.php contenant respectivement les classes Produit et ProduitManager.

La classe Produit a les mêmes attributs que les colonnes dans la base de données.

```
1  <?php
2  class Produit {
3      public $idProduit;
4      public $nom;
5      public $prix;
6      public $idCategorie;
7      public $descriptionProduit;
8      public $caracteristique;
9      public $dispo;
```

Un getter et un setter pour chaque attribut :

```
/* Getters
public function getIdProduit() {
    return $this->idProduit;
}
```

```
/* Setters
public function setIdProduit($id_produit) {
    $this->idProduit = $id_produit;
}
```

Et la méthode hydrate appelé au constructeur pour créer des objets produit en fonction de nos besoins et de manière plus intuitive.

```
function __construct(array $donnees) {
    $this->hydrate($donnees);
}
```

Toutes les tables de la base de données suivent ce principe.

```
public function hydrate(array $donnees) {
    foreach ($donnees as $key => $value) {
        $method = 'set' . $key;
        if (method_exists($this, $method)) {
            $this->$method($value);
        }
    }
}
```

4.1. Les managers

Chaque manager n'a qu'un attribut et prend pour le constructeur la variable \$bdd, l'objet PDO qui permet la connexion avec la base de données.

4.1.a. AvisClientManager

add (AvisClient avisclient) : void

Ajoute l'objet AvisClient à la base de données.

update (AvisClient avisclient) : void

Met à jour un avis client

get (idAvisClient) : AvisClient

Retourne un objet AvisClient en fonction de son identifiant.

getByIdProduit (idProduit) : array

Retourne un tableau d'objets AvisClient en fonction d'un produit.

getByIdClient (idClient) : array

Retourne un tableau d'objets AvisClient en fonction d'un client

getMoyenne (idProduit) : float
Retourne la moyenne des avis d'un produit.

countByIdProduit (idProduit) : int
Retourne le nombre d'avis sur un produit.

CountByIdProduitAndNote (idProduit, note) : int
Retourne le nombre d'avis d'un produit pour une note.

4.1.b. *CategorieManager*

add (Categorie categorie) : void
Ajoute l'objet Categorie dans le base de données

update (Categorie categorie) : void
Met à jour une catégorie.

get (idCategorie) : Categorie
Retourne un objet Categorie en fonction de l'identifiant.

getList () : array
Retourne une liste d'objets Categorie contenant toutes les catégories.

4.1.c. *ClientManager*

inscription (Client client) : string

```
public function inscription(Client $client) {
    /** Vérification des doublons d'adresse mail
    $req = $this->_db->prepare("SELECT * FROM client WHERE mail = :mail");
    $req->bindValue(':mail', $client->getMail());
    $req->execute();
    $result = $req->rowCount();
    if ($result > 0) {
        return "Cette adresse mail est déjà utilisé.";
    }

    /** Vérification des doublons de pseudo
    $req = $this->_db->prepare("SELECT * FROM client WHERE pseudo = :pseudo");
    $req->bindValue(':pseudo', $client->getPseudo());
    $req->execute();
    $result = $req->rowCount();
    if ($result > 0) {
        return "Ce pseudo est déjà utilisé.";
    }
}
```

Avant de valider l'inscription une vérification est faite pour éviter les doublons d'adresse mail et de pseudo. Si le pseudo ou l'adresse mail est déjà utilisé dans la base de données, la méthode renvoie un message d'erreur (string).

Sinon ajoute un client à la base de données et le connecte (création des variables \$_SESSION pour chaque attribut) et retourne void.

```
session_start();
$_SESSION['id'] = $data['idClient'];
$_SESSION['nom'] = $client->getNom();
$_SESSION['prenom'] = $client->getPrenom();
$_SESSION['pseudo'] = $client->getPseudo();
$_SESSION['mail'] = $client->getMail();
$_SESSION['telephone'] = $client->getTelephone();
$_SESSION['adresse'] = $client->getAdresse();
header("location:index.php");
```

verifMdp (Client client) : bool

Retourne True si idClient et mdp corresponde à un client dans la base de données.
Retourne False s'il n'y a pas de correspondance.

update (Client client)

Mets à jour les informations d'un client.

get (idClient) : Client

Retourne un objet client en fonction de l'identifiant.

connexion (Client client) : bool

Vérifie que l'adresse mail et le mdp correspondent, s'ils correspondent retourne true sinon false.

4.1.d. CommandeManager

add (Commande commande) : void

Ajoute l'objet Commande à la base de données.

get (idCommande) : Commande

Retourne un objet Commande en fonction de son identifiant.

getByIdClient (idClient) : array

Retourne un tableau d'objets Commande pour chaque commande d'un client.

getMaxId () : int

Retourne l'identifiant le plus grand (le plus récent).

4.1.e. HistoriqueProduitManager

getAncienProduit (idProduit, dateCmde)

Retourne un objet HistoriqueProduit correspondant au produit qui est le plus récent par rapport à la dateCmde mais qui ne la dépasse pas. (Pour l'historique d'achat d'un client, permet d'afficher le produit acheter sans prendre en compte les modifications faites entre l'achat et la consultation).

4.1.f. ImageProduitManager

add (ImageProduit imageProduit) : void

Ajoute l'objet imageProduit à la base de données.

get (idProduit) : array

Retourne un tableau associatif avec toute les chemins d'images d'un produit en fonction de l'idProduit.

GetOne (idProduit) : string

Retourne le premier chemin d'image d'un produit en fonction de l'idProduit.

Update (ImageProduit imageProduit) : void

Met à jour les informations d'une image.

verif (ImageProduit imageProduit) : bool

Vérifie si pour un produit l'image associer (1, 2, 3, 4 ou 5) existe ou pas.

Utile lors de la modification d'un produit pour savoir s'il faut add() ou update().

4.1.g. LigneDeCommandeManager

add (LigneDeCommande ligneDeCommande) : void

Ajoute l'objet ligneDeCommande à la base de données.

verifProduitAchete (LigneDeCommande ligneDeCommande) : int

Retourne le nombre de fois qu'un produit à été acheter dans une commande.

Permet de vérifié qu'un utilisateur à déjà acheter un produit.

```
$managerCommande = new CommandeManager($bdd);
$managerLigneDeCommande = new LigneDeCommandeManager($bdd);
$listeCommandeClient = $managerCommande->getIdClient($_SESSION['id']);
foreach ($listeCommandeClient as $li) {
    $res = $managerLigneDeCommande->verifProduitAchete(
        new LigneDeCommande([
            "IdCmde" => $li->getIdCmde(),
            "IdProduit" => $_GET['id']
        ])
    );
    if ($res > 0) $possibilitePoseAvis = true;
```

get (idCmde) : array

Retourne un tableau d'objet ligneDeCommande en fonction d'une commande.

4.1.h. ProduitManager

add (Produit produit) : void

Ajoute un objet produit à la base de données.

update (Produit produit) : void

Met à jour l'objet produit dans la base de données.

get (idProduit) : Produit produits

Retourne un objet produit en fonction de l'identifiant.

getByCategorie (idCategorie) : array

Retourne un tableau de tableau associatif contenant les données des produits en fonction d'une catégorie.

getTroisDernierByCategorie (idCategorie) : array

Retourne un tableau de 3 tableaux associatif contenant les données des 3 derniers produits ajouter en fonction d'une catégorie.

getByNom (nom) : Produit produits

Retourne un objet produit en fonction du nom.

Permet de récupérer l'identifiant d'un produit nouvellement créer.

getLimitOffset (limit, offset) : array

Retourne un tableau d'objet Produit en fonction de limit et offset.

Permet de faire une pagination php dans une page qui affiche tous les produits.

count () : int

Retourne le nombre de produits.

switchDispo (idProduit) : void

Change la disponibilité d'un produit, 0 devient 1 et 1 devient 0.

4.2. Connexion à la base de données et autoload

includes >  autoload.php

```
1  <?php
2  try{
3      $bdd = new PDO('mysql:host=localhost;dbname=MicroWorld;charset=utf8','root','');
4  }catch(PDOException $e){
5      die(print_r("Erreur bdd:".$e->getMessage()));
6  }
7
8  spl_autoload_register(function($class) {
9      require_once("classes/$class.class.php");
10 });
11
```

La connexion à la base de données n'est pas sécurisée, il faudrait créer plusieurs utilisateurs avec des mots de passe différents, et pour chaque utilisateur limiter les différents accès.

Lorsqu'une Classe est appelée, spl_autoload_register va automatiquement require_once le fichier contenant cette classe.

5. présentation du site

5.1. page d'accueil

La page d'accueil contient les 3 derniers produits de chaque catégorie.


Le nom de chaque catégorie est un lien vers celle-ci et de même pour chaque produit.

MicroWorld

PC GamerOrdinateur de bureauPC portablePériphérique


ConnexionInscription

Nos nouveaux produits




VIST PC Gaming AMD Ryzen 5
- RAM 16Go - NVIDIA
GeForce GTX 1650 - SSD
512Go m.2 - Windows 10

775 €




Megaport PC Gamer • AMD
Ryzen 5 3600 6x 3.60GHz •
GeForce GTX1650 • 8Go • 1To
HDD • SANS système
d'exploitation • 303-FR

750 €




MEMORY PC Gamer | AMD
Ryzen 5 3600, 6x 3.60GHz |
4Go GeForce GTX 1650 | 16Go
DDR4 RGB | 240 Go SSD +
2000 Go HDD | Wifi | Win 10
Pr

899 €




PC Lenovo ThinkCentre M73
Tiny G3220T 2.60GHz
8Go/480Go SSD Wifi W10


175 €




PC Portable HP Laptop 17-
cp0274nf - 17.3" HD+ - AMD
3000 Series 3020E / 1.2 GHz -
RAM 4 Go - Stockage 128 Go
SSD - Windows 10




PC Portable - ACER - Aspire
A315-34-C221 - 15.6" FHD -
Intel Celeron N4020 - RAM 4
Go - Stockage 128 Go SSD -
Windows 11 - AZERTY



ACER Nitro VG220Qbmix -
Ecran Gamer 21.5" - FHD -
Dalle IPS - 1ms - 75Hz - VGA /
HDMI - AMD FreeSync



Ecran PC Gamer - SAMSUNG
ODYSSEY G3 -
LF27G35TFWUXEN - 27" FHD
- Dalle VA - 1ms - 144Hz -
FreeSync Premium



Ecran PC Gamer - ACER Nitro
VG240YSbmilpx - 23.8" FHD -
Dalle IPS - 1ms - 165Hz -
2xHDMI / DP - AMD FreeSync
- Noir

Mail : iom.monchau@gmail.com
© Fait par Ioamra Monchau

5.2. Les catégories

Chaque catégorie à une page php et javaScript associé.

La page php contient un tableau d'identifiant 'datatable' qui va accueillir les données du fichier javascript via la librairie DataTable.

Par exemple pour la catégorie PC Gamer :

Le fichier php :

```
1  <?php session_start(); ?>
2  <!DOCTYPE html>
3  <html lang="fr">
4  <head>
5      <title>MicroWorld - PC Gamer</title>
6      <?php require_once "includes/head.php"; ?>
7      <script src="assets/js/pc-gamer.js"></script>
8  </head>
9      <body class="body-bg-grey">
10         <?php
11             require_once "includes/autoload.php";
12             require_once "includes/nav.php";
13             ?>
14         <div class="box-produit">
15             <h1 class="text-center pb-4">PC Gamer</h1>
16             <table id="datatable" class="table table-striped">
17                 <thead>
18                     <tr>
19                         <th class="no-sort"></th>
20                         <th>Nom</th>
21                         <th class="no-sort">Description</th>
22                         <th>Prix</th>
23                         <th>Note</th>
24                     </tr>
25                 </thead>
26             </table>
27         </div>
28
29         <?php require_once "includes/footer.php"; ?>
30     </body>
31 </html>
```

Le fichier JavaScript :

la fonction getProduit est lancée quand le chargement de la page est fini.

Cette fonction fait un appel à api.php avec un paramètre action et idCategorie.

Ici idCategorie = 1 car 1 correspond à PC gamer.

Puis ce résultat est utilisé avec la fonction DataTable de la librairie du même nom, ce qui va créer le contenu du tableau, en ajoutant avec les paramètres correspondant, le tri, la barre de recherche, quel champ sont recherchables, le langage...

```
assets > js > JS pc-gamer.js > ...
1  $(function(){
2      getProduit();
3  });
4
5  async function getProduit() {
6      const result = await fetch('api.php?action=getProduitByCategorie&idCategorie=1');
7      let res = await result.json();
8      $('#datatable').DataTable({
9          data: res,
10         columnDefs: [{
11             orderable: false,
12             targets: "no-sort"
13         }],
14         aaSorting: [],
15         lengthMenu: [10, 20, 40],
16         columns: [
17             {
18                 'data': 'img',
19                 'render': (data,type,row,meta) => {
20                     return '<a href="produit.php?id='+row.idProduit+'"><img style="height:12em; width:a
21                 },
22                 'searchable': false
23             },
24             {
25                 'data': 'nom',
26                 'render': (data,type,row,meta) => {
27                     return '<a class="text-decoration-none text-dark" href="produit.php?id='+row.idProd
28                 },
29             },
30         ]
31     });
32 }
```




La note est afficher sous forme d'étoile vide et creuse.

```
{
    'data': 'note',
    'render': (data) => {
        let roundNote = Math.round(data);
        let resNote = '<div class="etoile">';
        for (let i = 0; i < roundNote; i++) {
            resNote += '★';
        }
        for (let i = 0; i < 5-roundNote; i++) {
            resNote += '☆';
        }
        resNote += '</div>';
        return resNote;
    },
    'searchable': false
}
```

PC Gamer

Afficher 10 éléments

Rechercher :

	Nom	Description	Prix	Note
	MegaPort PC Gamer • AMD Ryzen 5 3600 6x 3.60GHz • GeForce GTX1650 • 8Go • 1To HDD • SANS système d'exploitation • 303-FR	Ce PC gaming de MegaPort saura combler tous vos désirs de gamer. Ce PC est équipé de composants à la pointe de la technologie et dispose de performances inégalées afin de satisfaire les attentes des gamers même les plus exigeants. Autre avantage de cette configuration : le boîtier absolument silencieux au somptueux design. Voici les ingrédients pour un ordinateur qui saura répondre aux attentes toujours grandissantes du monde du gaming. Notre service de retour de marchandises gratuit, valable pendant toute la durée de nos deux ans de garantie constructeur apporte la touche finale à ce pack sér ...	750€	★★★★★
	VIST PC Gaming AMD Ryzen 5 - RAM 16Go - NVIDIA GeForce GTX 1650 - SSD 512Go m.2 - Windows 10	RGB DO GIER AMD Ryzen 5 GTX 1650 16GB SSD 512 m.2 W10 Processeur: AMD Ryzen 5 1600AF - 12 x 3,6 Ghz BOX Mémoire de Ram: DDR4 16GB - Mémoire haute vitesse de 2666 Mhz avec dissipateurs thermiques Disque SSD: SSD 512GB NVMe Pci-E - Plusieurs fois plus rapide que les SSD standard, avec des vitesses de lecture et d'écriture de 3400/2400 Mo/s, il offre de meilleures performances. Alimentation électrique: Silent 650W 140mm - Alimentation électrique robuste certifiée 80+, filtres anti-tension, filtres anti-interférences, grand ventilateur de 140 mm Construction: VIST RGB VR6 - La photo montre un ...	775€	★★★★★
	MEMORY PC Gamer AMD Ryzen 5 3600, 6x 3.60GHz 4Go GeForce GTX 1650 16Go DDR4 RGB 240 Go SSD + 2000 Go HDD WIFI Win 10 Pr	Ce PC de jeu doté d'un processeur AMD Ryzen 5 SixCore et d'une puissante NVIDIA GeForce GTX 1650 avec 4 Go de mémoire vous permet de jouer aux jeux actuels sans problème. La technologie à l'état solide du SSD de 240 Go assure un démarrage ultra-rapide du Memory PC® et tous les autres processus informatiques sont également nettement plus rapides grâce au SSD. Le disque dur supplémentaire de 2000 Go offre également suffisamment d'espace pour toutes vos données. Bien sûr, Windows 10 Pro est déjà installé, il suffit d'activer le système d'exploitation une seule fois. Configuration : Système d'expl ...	899€	★★★★★

Affichage de l'élément 1 à 3 sur 3 éléments

Précédent 1 Suivant

5.3. Page de connexion

Connexion

Adresse Email

Mot de passe

Se connecter

[Vous n'avez pas de compte, inscrivez-vous !](#)

```
if(isset($_POST['mail']) && isset($_POST['mdp'])){
    $mail = $_POST['mail'];
    $mdp = sha1($_POST['mdp']);

    $manager = new ClientManager($bdd);
    $connexionOk = $manager->connexion(
        new Client([
            "Mail" => $mail,
            "Mdp" => $mdp
        ])
    );
    if ($connexionOk == false) $mesError = "Adresse e-mail ou mot de passe invalide.";
}
```

5.4. Page d'inscription

Inscription

Nom

Prenom

Pseudo

Adresse Email

Telephone

Adresse

Mot de passe

Mot de passe

```
if (isset($_POST['submit'])){  
    $nom = htmlspecialchars($_POST['nom']);  
    $prenom = htmlspecialchars($_POST['prenom']);  
    $pseudo = htmlspecialchars($_POST['pseudo']);  
    $tel = htmlspecialchars($_POST['tel']);  
    $mail = htmlspecialchars($_POST['mail']);  
    $adress = htmlspecialchars($_POST['adress']);  
    if ($_POST['mdp1'] == $_POST['mdp2']) {  
        $mdp = sha1($_POST['mdp1']);  
        $manager = new ClientManager($bdd);  
        $mesError = $manager->inscription(  
            new Client([  
                "Nom" => $nom,  
                "Prenom" => $prenom,  
                "Pseudo" => $pseudo,  
                "Mail" => $mail,  
                "Telephone" => $tel,  
                "Adresse" => $adress,  
                "Mdp" => $mdp  
            ])  
        );  
    } else {  
        $mesError = "Les deux mots de passe ne sont pas identique.";  
    }  
}
```


5.5. Le panier

	Nom	Prix unitaire	quantité	prix global
	ACER Nitro VG220Qbmi...	130 €/unité	<input type="button" value="-"/> <input type="text" value="2"/> <input type="button" value="+"/>	260 €
	MEMORY PC Gamer AM...	899 €/unité	<input type="button" value="-"/> <input type="text" value="1"/> <input type="button" value="+"/>	899 €
	AUKEY Clavier Gamer ...	23 €/unité	<input type="button" value="-"/> <input type="text" value="1"/> <input type="button" value="+"/>	23 €
	LOGITECH - Souris Ga...	89 €/unité	<input type="button" value="-"/> <input type="text" value="1"/> <input type="button" value="+"/>	89 €
TOTAL :				1271 €

La livraison est offerte et les colis sont livrés en 2-3 jours ouvrables avec Colissimo.

Vos coordonnées bancaires

Nom du détenteur de la carte	
Numéros de la carte	
Date d'expiration	Code de sécurité

Valider l'achat

La gestion du panier est faite avec la classe Panier dans panier.class.php et l'affichage du panier est fait avec panier.js.

Les méthodes de la classe Panier :

constructor () : void

Récupère dans le localStorage le tableau d'objet panier et l'associe à l'attribut panier.
Si localStorage est vide, l'attribut panier est un tableau vide.

save () : void

Remplace panier dans localStorage par this.panier.

add (produit) : void

Ajoute l'objet produit à this.panier s'il n'est pas déjà dedans, sinon ajoute 1 à sa quantité.
Puis save() pour sauvegarder les changements.

remove (produit) : void

Retire l'objet produit du tableau this.panier puis save().

removeAll () : void

this.panier devient un tableau vide, puis save()

ajoutQte (produit, qte) : void

Change la quantité d'un produit dans this.panier puis save().

getNbProduit () : int
Retourne le nombre de produits dans le panier.

getPrixTotal () : int
Retourne le prix total du panier.

5.6. Le profil client


Depuis son profil un utilisateur peut modifier ses informations, et consulter son historique d'achat.


Profil

Nom :	<input type="text" value="Dupond"/>
Prenom :	<input type="text" value="Michel"/>
Pseudo :	<input type="text" value="Mich"/>
Mail :	<input type="text" value="mich@gmail.com"/>
Telephone :	<input type="text" value="0011223344"/>
Adresse :	<input type="text" value="92 Avenue De Marlioz, Arles 13200"/>
Ancient mot de passe :	<input type="password"/>
Nouveau mot de passe :	<input type="password"/>
Nouveau mot de passe :	<input type="password"/>

Valider les changement[Voir l'historique de mes achats.](#)

N° commande	Date
20	2022-04-30
21	2022-04-30

	Nom	Prix	Quantité	Prix Total
	PC Portable HP Laptop 17-cp027 ...	299 €	x 1	299 €
				299 €

	Nom	Prix	Quantité	Prix Total
	PC Portable - ACER - Aspire A3 ...	315 €	x 1	315 €
				315 €

5.7. Ajouter un produit

Cette page est réservée aux administrateurs.

Le champ caractéristique est déjà pré rempli pour simplifier les différents ajouts.

5 images peuvent être ajoutées pour un produit, la première est obligatoire.

Et chaque image est pré-visualisé en dessous des boutons pour les ajouter.

Ajouter un Produit

Nom

Description

Caractéristique

Categorie

Prix











Images (une image minimum)

Ajouter le produit


```
<table class="table w-100">
  <thead>
    <tr>
      <th class="w-25">Informations générales sur le produit</th>
    </tr>
  </thead>
  <tbody>
  </tbody>
</table>
```

5.8. Modifier un produit

Cette page est réservée aux administrateurs. Elle contient la liste de tous les produits, avec 10 produits par page et un pagination php.
Chaque produit à un bouton à droite pour changer sa disponibilité.
Et si l'on click sur un des produits, le même formulaire que l'ajout de produit apparaît, a la différence qu'il est prérempli.

	EMPIRE GAMING – Dark Rainbow ...	Casque audio	22 €	Disponible
	Casque Gaming Casque Gamer pou ...	Casque audio	20 €	Disponible
	Casque Gaming filaire - LOGITE ...	Casque audio	84 €	Disponible
	LOGITECH G - Casque Gaming san ...	Casque audio	129 €	Disponible
	LOGITECH - Souris Gamer Sans F ...	Souris	89 €	Disponible
	Souris gaming sans fil LIGHTSP ...	Souris	39 €	Disponible
	LOGITECH G - Souris gaming san ...	Souris	85 €	Disponible
	K80 clavier mécanique filaire ...	Clavier	25 €	Disponible
	AUKEY Clavier Gamer Mécanique ...	Clavier	23 €	Disponible
	ACER Nitro VG220Qbmiix - Ecran ...	Ecran	130 €	Disponible




1 2

	Ecran PC Gamer - SAMSUNG ODYSSEY ...	Ecran	220 €	Disponible
Nom				
Ecran PC Gamer - SAMSUNG ODYSSEY G3 - LF27G35TFWUXEN - 27" FHD - Dalle VA - 1ms - 144Hz - FreeSync Premium				
Description				
<p>Ecran LCD à rétroéclairage LED / matrice active TFT Interfaces : VGA, DisplayPort 1.2, HDMI, Casque Résolution native : Full HD (1080p) 1920 x 1080 à 144 Hz Temps de réponse : 1 ms (MPRT) Type de panneau : VA Luminosité : 250 cd/m² Rapport de contraste : 4000:1</p>				
Caracteristique				
</tbody> </table>				
Prix				
220				

Images (une image minimum)



Modifier le produit

	Ecran PC Gamer - ACER Nitro VG ...	Ecran	170 €	Disponible
	PC Portable HP Laptop 17-cp027 ...	PC portable	299 €	Disponible
	PC Lenovo ThinkCentre M73 Tiny ...	Ordinateur de bureau	175 €	Disponible

5.9. Connexion administrateur

La connexion administrateur se fait sur la page co-admin.php, aucun lien ne cible vers cette page. Le mot de passe est enregistré dans le fichier co-admin.php et correspond à 'azerty', mot de passe à changer car bien trop simple.

```
if(isset($_POST['pseudo']) && isset($_POST['mdp'])){
    $pseudo = $_POST['pseudo'];
    $mdp = sha1($_POST['mdp']);
    if ($pseudo == "admin" && $mdp == "9cf95dacd226dcf43da376cdb6cbba7035218921") {
        session_start();
        $_SESSION['id'] = 0;
        $_SESSION['admin'] = "admin";
        header("location:index.php");
    }
}
```

5.10. L'api

Le fichier api.php fonctionne via les paramètres GET. Selon \$_GET['action'], trois actions différentes sont possibles :

5.10.a. La validation d'un achat

Nécessite 2 autres paramètres, idProduit et qteProduit.

Dans ce cas, une nouvelle commande est ajoutée à la base de données puis pour chaque produit, une ligne de commande est ajoutée à la base de données.

```
if ($action == 'validAchat') {
    if (isset($_GET['idProduit'], $_GET['qteProduit'])) {
        $listId = $_GET['idProduit'];
        $listQte = $_GET['qteProduit'];
        $managerCommande = new CommandeManager($bdd);
        $managerLigneDeCommande = new LigneDeCommandeManager($bdd);
        date_default_timezone_set("Europe/Paris");
        $managerCommande->add(
            new Commande([
                "IdClient" => $_SESSION['id'],
                "DateCmde" => date("Y-m-d")
            ])
        );
        $idCmde = $managerCommande->getMaxId();
        for ($i = 0; $i < count($listId); $i++) {
            $managerLigneDeCommande->add(
                new LigneDeCommande([
                    "IdCmde" => $idCmde,
                    "IdProduit" => $listId[$i],
                    "Qte" => $listQte[$i]
                ])
            );
        }
    }
}
```

5.10.b. *La recherche de produit par catégorie*

Nécessite un second paramètre idCategorie.

Affiche sous forme de JSON les données des produits de la catégorie correspondante avec en plus la première image de chaque produit et sa note.

Données qui seront récupéré en JavaScript.

```
if ($action == 'getProduitByCategorie' && isset($_GET['idCategorie'])) {  
  
    $idCategorie = $_GET['idCategorie'];  
    $managerProduit = new ProduitManager($bdd);  
    $managerImageProduit = new ImageProduitManager($bdd);  
    $managerAvisClient = new AvisClientManager($bdd);  
  
    $list = $managerProduit->getByCategorie($idCategorie);  
  
    foreach ($list as $i => $val) {  
        $list[$i]['img'] = $managerImageProduit->getOne($val['idProduit']);  
        $list[$i]['note'] = round($managerAvisClient->getMoyenne($val['idProduit']), 2);  
    }  
    echo json_encode($list);  
}
```

5.10.c. *Le changement de disponibilité d'un produit*

Nécessite un second paramètre idProduit.

Change la disponibilité d'un produit.

```
if ($action == 'switchDispo' && isset($_GET['idProduit'])) {  
    $managerProduit = new ProduitManager($bdd);  
    $managerProduit->switchDispo($_GET['idProduit']);  
}
```

5.11. Programmation coté serveur

Deux Trigger sont utilisé pour sauvegarder les modifications des produits.

Le premier intervient après un ajout de produit et copie le produit dans la table historique_produit en y ajoutant le type 'ajout' et la date du jour.

Détails	
Nom du déclencheur	histoInsert
Table	produit
Moment	AFTER
Évènement	INSERT
Définition	<pre>1 INSERT INTO historique_produit(idProduit, nom, prix, idCategorie, descriptionProduit, caracteristique, dateChangement, type) VALUES (new.idProduit, new.nom, new.prix, new.idCategorie, new.descriptionProduit, new.caracteristique, substr(now(),1,10), 'ajout')</pre>
Créateur	root@localhost

Le second intervient après une modification, créer une nouvelle ligne dans la table historique_produit avec les données mise à jour, le type 'MAJ' et la date du jour.

Nom du déclencheur	histoUpdate
Table	produit
Moment	AFTER
Évènement	UPDATE
Définition	<pre>1 INSERT INTO historique_produit(idProduit, nom, prix, idCategorie, descriptionProduit, caracteristique, dateChangement, type) VALUES (new.idProduit, new.nom, new.prix, new.idCategorie, new.descriptionProduit, new.caracteristique, substr(now(),1,10), 'MAJ')</pre>
Créateur	root@localhost