



**Université Abdelmalek Essaadi Ecole Nationale des Sciences
Appliquées Al Hoceima Département Mathématiques et
Informatique**



Génie Informatique

Module : Développement des applications Web

Site web d'e-commerce

Réalisé par : JAADAR Sami.

Plan :

1. Présentation du projet

1.1 Définition du projet.

1.2 Problématique.

1.3 Objectifs.

2. Spécification et modélisation de l'application.

2.1 Acteurs de l'application.

2.2 Diagramme de cas d'utilisation.

2.3 Diagramme de cas d'utilisation de *l'administrateur*.

2.4 Diagramme de cas d'un visiteur.

2.5 Diagramme de cas d'un client.

2.6 Conception de la base de données.

3. Implémentation

3.1 Outils de développement.

4. Programmation

4.1 Page d'inscription de visiteur.

4.2 Le code PHP pour enregistrer les données.

4.3 Page de connexion.

4.4 Le code PHP de validation de connexion.

4.5 Le Code PHP de déconnexion.

4.6 Le Code PHP d'affichage des produits.

4.7 Code PHP d'affichage du détail des produits.

4.7 Code PHP d'affichage du détail des produits.

5. panier

5.1 le code PHP DU PANIER.

5.2 Le code d'affichage du panier.

5.3 Le code PHP de vérification d'identification.

5.4 Le code de récupération de détails de la livraison.

5.5 Le code PHP d'ajout d'informations de livraison du client.

5.6 Le code de formulaire de remplissage d'information de paiement.

5.7 Le code PHP de mise à jour des informations de paiement.

5.8 L'ajout des informations de la commande.

5.9 Le code PHP de facture de la commande.

5.10 Page d'ajout d'un produit.

5.11 Page de mise à jour des produits.

5.12 Page de suppression d'un produit.

5.13 La page d'affichage des commandes.

5.14 Code PHP de la page des statistiques.

6. Interfaces principales

6.1 Page d'accueil.

6.2 Interface identification.

6.3 Interface inscription.

6.4 Interface d'affichage des produits.

6.5 Interface d'affichage du détail du produit.

6.6 Interface du panier.

6.7 Interface de vérification d'identification.

6.8 Interface de formulaire de livraison.

6.9 Interface de formulaire de paiement.

6.10 Facture PDF de la commande.

7. Interface d'administrateur

7.1 Interface identification.

7.2 Interface principale de la page de l'administrateur

7.3 Interface d'ajouter un produit.

7.4 Interface de mise à jour des produits.

7.5 Interface de suppression des produits.

7.6 Interface des statistiques.

7.7 Interface des commandes.

8. Conclusion

1. Présentation du projet :

1. Définition du projet :

Nous voulons réaliser un projet visant la création d'une application web de la vente en ligne. L'application va gérer les différents produits et les différentes relations avec les clients grâce à une base de données.

2. Problématique :

Dans le commerce traditionnel la procédure d'achat est difficile et épuisante (se déplacer, voir tous les modèles de produit, passer d'un service à un autre pour remplir la commande, la soumettre puis effectuer le paiement passer au magasin de stock pour la restitution de la marchandise).

De ce fait, nous avons souligné plusieurs problèmes auxquels nous tenterons de répondre dans notre projet :

- Difficulté à faire circuler l'information.
- Difficulté de l'accès au produit.
- Complexité de la procédure d'achat,

3. Objectifs :

Ce travail consiste à créer une application web e-commerce permettant l'échange de bien et de services en ligne.

Cette application permettra de réaliser les opérations suivantes :

- Avoir un accès aux produits sans frontières.
- Garantir des transactions financières sécurisées
- Offrir une évaluation pour les différents produits.
- Gérer les relations avec les clients.
- Gérer le stock.
- Gérer les commandes.
- Mettre en place des promotions.
- Gérer les produits (ajouter, modifier ou supprimer des produits).

2. Spécification et modélisation de l'application :

Dans cette partie nous abordons la partie conception et modélisation du projet, dans laquelle nous détaillons les différents éléments de conception

2.1 Acteurs de l'application :

Nous savons que l'acteur est une personne ou un logiciel qui interagit avec le programme de l'extérieur. Dans ce projet, on a défini 3 acteurs :

- L'administrateur : Il assure le dynamisme de l'application et la mise en veille sur les mises à jour des produits, de leurs prix, de leur disponibilité en stock et de la gestion des paiements.
- Le visiteur : C'est un individu qui est entrain de fouiller sur le net, cherchant un produit pour l'acheter ou pour avoir une idée sur les modèles et les prix. Jusqu'à ce stade c'est un utilisateur inconnu donc il n'est pas encore un client.
- Le client : Cet acteur est un visiteur ayant déjà créé un compte sur notre application web. Il peut donc suivre le processus d'achat des produits en toute sécurité sachant que notre système doit être l'unique responsable de la confidentialité des données personnelles de ses clients.

2.2 Diagramme de cas d'utilisation :

Le diagramme de cas d'utilisation est un diagramme UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel. Il est utile pour des présentations auprès de la direction ou des acteurs d'un projet.

Un cas d'utilisation représente une unité discrète d'interaction entre un utilisateur (humain ou machine) et un système.

Nous avons pensé à simplifier notre projet par 3 diagrammes du cas d'utilisation :

- Diagramme de cas d'utilisation de l'administrateur.
- Diagramme de cas d'utilisation de visiteur.
- Diagramme de cas d'utilisation de client.

2.3 Diagramme de cas d'utilisation de l'administrateur :

L'administrateur doit préciser son email et son mot de passe pour accéder à l'application.

L'administrateur à quatre cas d'utilisations :

- **Gérer les produits** : l'administrateur gère le cas ou le produits (ou plusieurs) sera ajouté, supprimé ou modifié.
- **Gérer les comptes** : l'administrateur peut modifier ou supprimer un certain nombre de compte.
- **Gérer les factures** : l'administrateur peut disposer au clients la possibilité d'imprimer leurs factures.
- **Gérer les commandes** : dans le cas où un client passe sa commande, elle ne peut être valider qu'après la validation de toutes les informations qui sont affichées dans une seule interface avant de passer à la phase de payement.

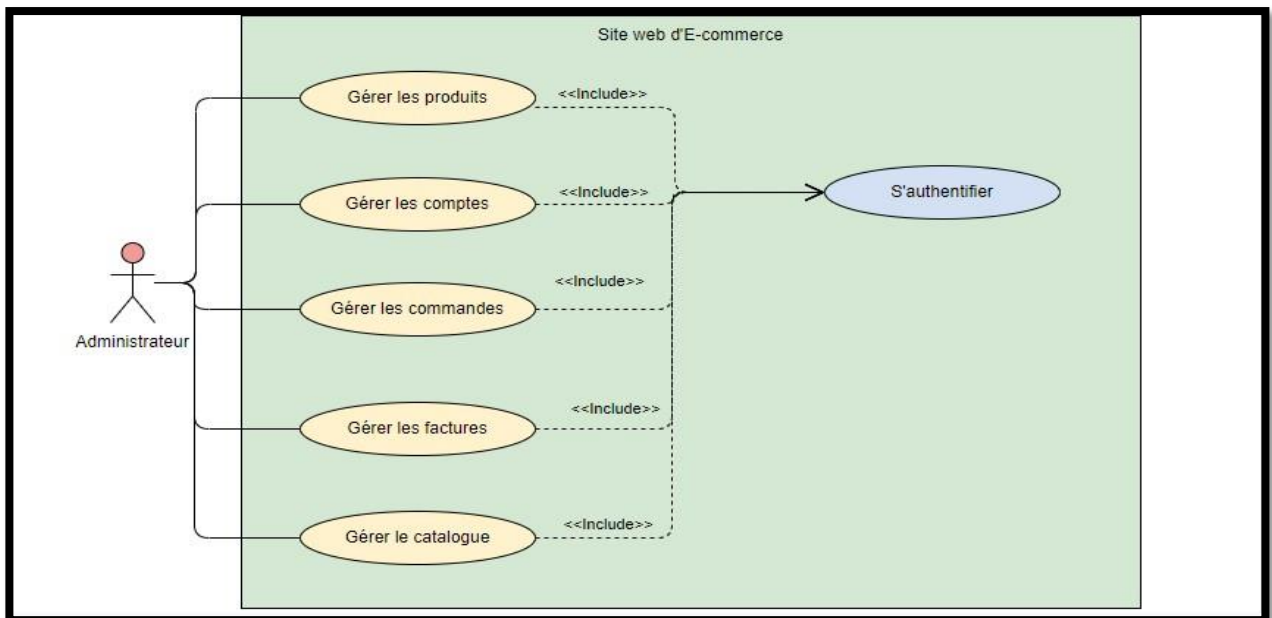


Figure 1: Diagramme de cas d'utilisation de l'administrateur.

2.4 Diagramme de cas d'un visiteur :

Le visiteur a deux cas d'utilisation :

- **Consultation des produits** : le visiteur peut visualiser le catalogue
- **Créer un compte** : pour qu'il soit un client, le visiteur peut remplir le formulaire d'inscription afin d'avoir son propre compte.

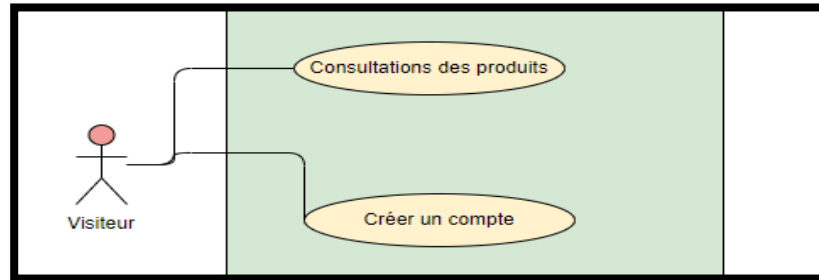


Figure 2: Diagramme de cas d'utilisation de visiteur.

2.5 Diagramme de cas d'un client :

Le client doit préciser son email et son mot de passe.

Le client a trois cas d'utilisation :

- **Achat de produit : le client a la possibilité d'acheter un certain nombre de produit.**
- **Consultation des commandes : le client peut consulter sa commande à tout moment.**
- **Achat de produit : le client a la possibilité d'acheter des produits.**

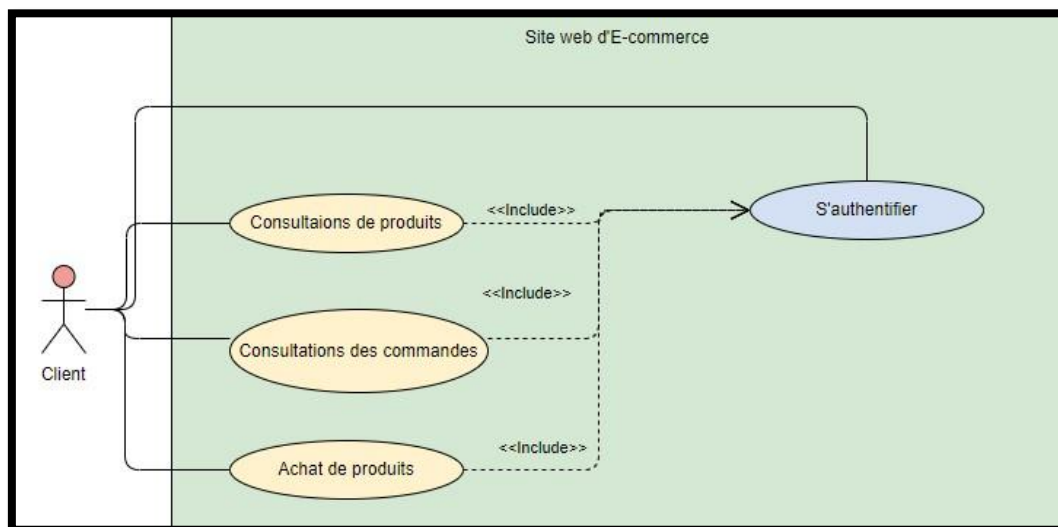


Figure 3: Diagramme de cas d'utilisation de visiteur

2.6 Conception de la base de données :

Le MCD est l'élément le plus connu du MERISE et certainement le plus utile. Il permet d'établir une représentation claire des données de SI et définit les dépendances fonctionnelles de ces données entre elles.

Dans notre application nous avons utilisé 6 entités (client, administrateur, produits, panier, orders , paiement_détails).

La figure ci-dessous représente le MCD de notre application :

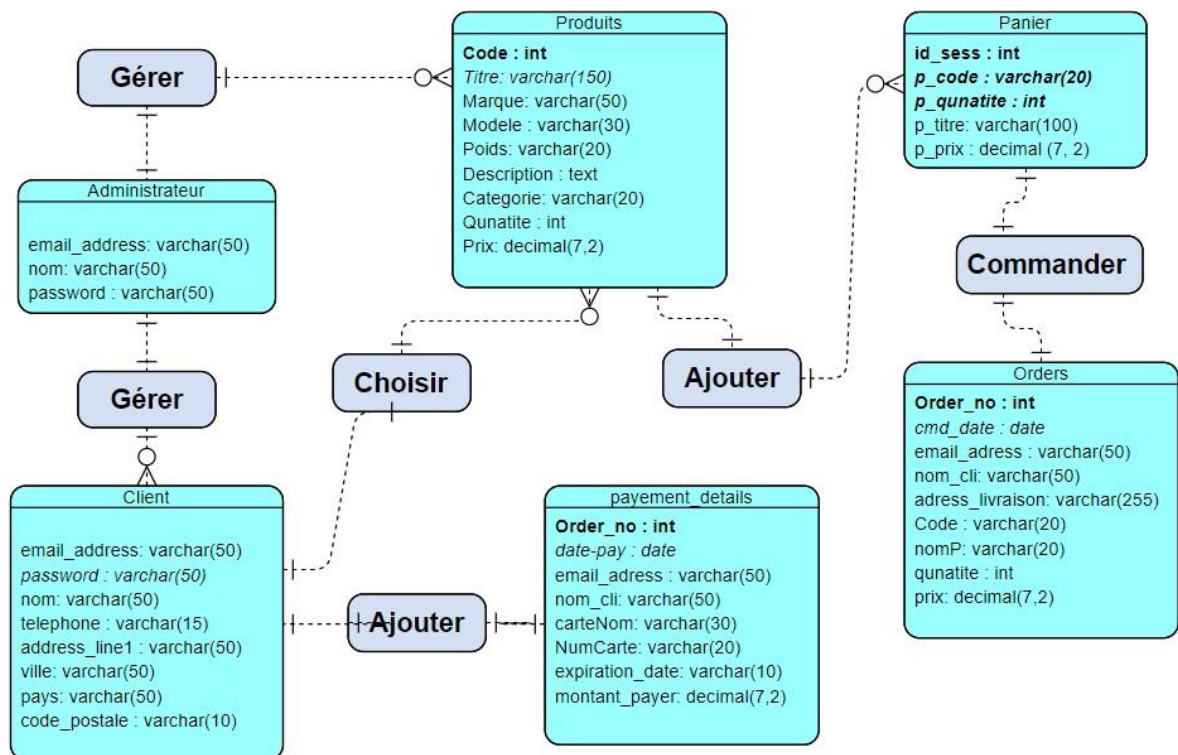


Figure 4:MCD du site web

3. Implémentation :

3.1 Outils de développement :

La mise en place de notre application nécessite l'utilisation de plusieurs langages de programmation et de développement web, les différents langages informatiques et outils que nous avons utilisés dans ce travail sont :

- HTML
- CSS
- PHP
- FPDF
- SQL
- SGBD MYSQL
- WAMP SERVER
- ATOM
- TCPDF

4. Programmation :

Après la réalisation de la base de données on a commencé à traduire les fonctionnalités de notre application sous forme de code à l'aide des langages HTML, CSS et PHP.

Dans cette partie, on va présenter quelques fragments de codes concernant la tâche importante de l'application

4.1 Page d'inscription visiteur :

La figure ci-dessous représente un code HTML qui permet d'afficher le formulaire dans notre page d'inscription.

```
5 <body>
6 <div class="loginbox" style="height:520px">
7 <h1>S'inscrire</h1>
8 <form action="ajouterClient.php" method="post">
9 <p>Nom d'utilisateur</p>
10 <input type="text" name="nom" placeholder="Entrer le nom d'utilisateur">
11 <p>Email</p>
12 <input type="text" name="email" placeholder="Entrer l'email">
13 <p>Mot de passe</p>
14 <input type="password" name="pass" placeholder="Entrer le mot de passe">
15 <input type="submit" name="" value="Login">
16 <p> En créant un compte, vous acceptez les conditions générales et la politique de confidentialité </p>
17 </form>
18 </div>
19 </body>
```

Figure 5: code html pour ajouter un client

4.2 Le code PHP pour enregistrer les données :

La figure ci-dessous représente le code PHP qui permet d'ajouter les informations du client en base de données.

```
<?php
$mysqli = new mysqli("localhost", "root", "", "shopping");
if ($mysqli->connect_error) {
    die(mysqli_error());
}
$email_adresse = $_POST["email"];
$nomcomplet = $_POST["nom"];
$password = $_POST["pass"];
if (!filter_var($email_adresse, FILTER_VALIDATE_EMAIL)) {
    echo "<script> alert(\"Adresse email est invalide\");</script>\n";
    echo "<meta http-equiv=\"Refresh\" content=\"0; url=signup.php\" />";
}
else if (empty($password) || strlen($password)<4){
    echo "<script> alert(\"mot de passe est vide ou tres petit\");</script>\n";
    echo "<meta http-equiv=\"Refresh\" content=\"0; url=signup.php\" />";
}
else{
    $sql = "INSERT INTO client (email_address,password,nom )
VALUES ('$email_adresse',(PASSWORD('$password')), '$nomcomplet')";
    if ($mysqli->query($sql) === TRUE) {
    } else {
        echo "Erreur: " . $sql . "<br>" . $mysqli->error;
    }
}
?>
```

Figure 6:code PHP pour ajouter un client

4.3 Page de connexion :

La figure ci-dessous représente un code HTML qui permet d'afficher le formulaire dans notre page de connexion.

```
5 <body>
6 <div class="loginbox" style="height:420px">
7 <h1>S'identifier</h1>
8 <form class="loginin" action="validerClient.php" method="post" >
9 <p>Nom d'utilisateur</p>
10 <input type="text" name="email" placeholder="Entrer le nom d'utilisateur">
11 <p>Mot de passe</p>
12 <input type="password" name="pass" placeholder="Entrer le mot de passe">
13 <input type="submit" value="Login">
14 <a href="#">Mot de passe oublier?</a><br>
15 <a href="signup.php">Pas de compte?</a>
16 </form>
17 </div>
18 </body>
```

Figure 7:un code HTML de formulaire

4.4 Le code PHP de validation de connexion :

La figure ci-dessous représente le code PHP qui permet de vérifier si les identifications du client sont identiques aux informations en base de données.

```
3 <?php
4 include('topmenu.php');
5 if (session_status() == PHP_SESSION_NONE) { session_start(); }
6 $connect = mysqli_connect("localhost", "root", "", "shopping") or
7 die("Please, check your server connection.");
8 $query = "SELECT email_address, password, nom FROM client
9 WHERE email_address like '' . $_POST['email'] . '' " .
10 "AND password like (PASSWORD('' . $_POST['pass'] . ''))";
11 $result = mysqli_query($connect, $query) or die(mysql_error());
12 if (mysqli_num_rows($result) == 1) {
13 while ($row = mysqli_fetch_array($result, MYSQLI_ASSOC)) {
14 extract($row);
15 echo "<meta http-equiv='Refresh' content='0; url=index.php' />";
16 $_SESSION['email'] = $_POST['email'];
17 $_SESSION['pass'] = $_POST['pass'];
18 } }
19 else {
20 >>
21 <h1 style="text-align:center; color:gray;">Adresse email ou mot de passe invalide</h1>
22 <div class="div">
23 <div class="item-div">
24 Pas Inscrits?<br>
25 <a href="signup.php">Cliquer ici</a> pour inscription.<br> </div>
26 <div class="item-div"> <br>
27 <a href="login.php">Cliquer ici</a> pour ressayer.<br> </div>
28 </div>
```

Figure 8 : un code PHP validation de client.

Après qu'un utilisateur a entré une adresse e-mail et un mot de passe, le code PHP consiste à :

- Confirmer si les informations saisies sont valides.
- Il valide l'utilisateur, c'est-à-dire qu'il détermine si l'utilisateur a entré une adresse e-mail et un mot de passe valides.
- Si une adresse e-mail ou un mot de passe invalide est entré, il affiche un message d'erreur suivi de deux liens qui permettent aux utilisateurs d'entrer leurs informations d'identification ou créer un nouveau compte.
- Si l'utilisateur est authentifié avec succès, il met à jour l'en-tête de le site, en remplaçant le lien de connexion par le nom de l'utilisateur avec le message de bienvenue.
- En outre, le lien d'inscription est remplacé par le lien de déconnexion

4.5 Le Code PHP de déconnexion :

```
1  <?php
2  session_start();
3  if (isset($_SESSION['email']))
4  {
5      unset($_SESSION['email']);
6      session_destroy();
7  }?>
8  <meta http-equiv="Refresh" content="0; url=index.php" />
9
```

Figure 9:code PHP de déconnexion

4.6 Le Code PHP d'affichage des produits :

Le code ci-dessous affiche les produits dont la catégorie est sélectionnée dans le menu déroulant. Ce code est configuré pour afficher uniquement trois produits sur une ligne ; par conséquent, une variable x est utilisée qui fait apparaître seulement trois produits sur une ligne.

Les informations affichées sont l'image du produit, son nom et son prix. Pour permettre aux utilisateurs d'afficher des informations détaillées sur le produit, un lien hypertexte est créé sur l'image du produit et son nom ou le bouton « Voir ».

Cela signifie que si l'utilisateur clique sur l'image, bouton « Voir » ou le nom d'un produit, la navigation s'effectuera vers le code qui affichera plus de détails sur ce produit.

```
<?php
$connect = mysqli_connect("localhost", "root", "", "shopping") or
die(mysql_error());
$Category=$_REQUEST['category'];
$query = "SELECT Code, Titre, Description, Image, Prix FROM produits " .
"where Categorie like '$Category' order by Code";
$results = mysqli_query($connect, $query) or die(mysql_error());
$x=1;
while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) {
if ($x <= 4)
{
$x = $x + 1;
extract($row);
?>
<div class="item"><?php echo "<a href=details.php?itemcode=$Code>";
echo '<img src=' . $Image . ' style="max-height:200px;width:auto;height:auto;"></img><br>';
echo "<h3 style=\"font-weight:bold;color:black;\">".$Titre."</h3><br>";
echo "<h3 style=\"font-weight:bold;color:black;\">".$Prix." &#36;</h3><br>";
?>
<button class="btn">Voir</button></a></div>
<?php }
else { $x=1; }
}
?>
```

Figure 10:code PHP d'affichage des produits

5. PANIER :

Le programme commence par le démarrage de la session. L'ID de session sera utilisé pour se souvenir les articles sélectionnés dans le panier par l'utilisateur particulier.

Le statut de la session est égal à `PHP_SESSION_NONE` si la session est activée et qu'aucune n'existe. Puis une connexion au serveur MySQL est établie et le fichier de base de données d'achat est sélectionné.

La quantité de l'article acheté (stockée dans la variable de quantité) est récupérée à partir du Tableau `$_POST`.

Si l'utilisateur n'a pas spécifié de quantité, la valeur par défaut est une. Il vérifie également que la valeur de la quantité saisie est un nombre non négatif. Le code d'article, le nom d'article, et le prix (stocké dans les variables `icode`, `iname` et `iprice`) envoyés à partir de la page de détails de l'article. Les scripts PHP sont récupérés à l'aide du tableau `$_REQUEST`. Une session est générée et stockée dans la variable `$ sess`.

Cet ID de session est attribué à tous les produits sélectionnés dans le panier pour identifier les articles sélectionnés par un utilisateur particulier.

La valeur de la variable d'action est récupérée et détermine le type d'opération à effectuer appliqué aux articles du panier. La valeur de la variable d'action peut être ajoutée, modifiée ou supprimée.

Lorsque la valeur de la variable action est ajoutée, cela signifie que le produit doit être ajouté au panier table. De même, si la valeur d'action est changer, cela signifie un produit qui est déjà dans le panier doit être modifié, et l'action de suppression signifie que le produit spécifié doit être supprimé du panier.

Lorsque la variable d'action est ajoutée, elle est d'abord vérifiée pour déterminer si le produit est déjà dans le panier. Si le produit est déjà dans le panier, seul le champ de quantité est modifié, c'est-à-dire que la quantité du produit est incrémentée pour indiquer l'addition. Si le produit n'existe pas dans le panier, le produit sera ajouté à la table du panier.

5.1 Le code PHP du panier :

Lorsque le bouton « ajouter au panier » est cliquer, le code ci-dessous ajoute le produit dans la table de panier.

```
case "ajouter":
$query="select * from panier where id_sess = '$sess' and p_itemcode like '$itemcode'";
$result = mysqli_query($connect, $query) or die(mysql_error());
if(mysqli_num_rows($result)==1){
$row=mysqli_fetch_array($result, MYSQLI_ASSOC);
$qt=$row['p_quantite'];
$qt=$qt + $quantite;
$query="UPDATE panier set p_quantite=$qt where id_sess = '$sess' and
p_itemcode like '$itemcode'";
$result = mysqli_query($connect, $query) or die(mysql_error());
}
else
{
$query = "INSERT INTO panier (id_sess, p_quantite, p_itemcode,p_item_name, p_prix)
VALUES ('$sess', $quantite, '$itemcode','$item_name', $prix)";
} break;
```

Figure 12:Code PHP en cas d'ajout

Vous pouvez toujours modifier la quantité de n'importe quel produit en entrant la quantité souhaitée dans la zone de texte, puis en cliquant sur le bouton « Modifier la quantité » trouvée dans la page de détails du produit.

```
case "changer":
if($modified_quantite==0)
{
$query = "DELETE FROM panier WHERE p_sess = '$sess' and p_itemcode like '$itemcode'";
}
else
{
if($modified_quantite < 0)
{
echo "Quantite invalide";
}
else
{
$query = "UPDATE panier SET p_quantite = $modified_quantite WHERE
id_sess = '$sess' and p_itemcode like '$itemcode'";
}
}break;
```

Figure 13:Code PHP en cas de changement de quantité

Vous pouvez également supprimer n'importe quel produit du panier en sélectionnant le bouton « Supprimer ».

```
case "supprimer":
$query = "DELETE FROM panier WHERE id_sess = '$sess' and p_itemcode like '$itemcode'";
break;
case "vider":
$query = "DELETE FROM panier WHERE id_sess = '$sess'";
break;
}
if($query != "")
{
$results = mysqli_query($connect, $query) or die(mysql_error());
}
echo "<SCRIPT LANGUAGE='JavaScript'>updateCart();</SCRIPT>";
}
#calcul du montant totale du panier

$totalquantite=0;
$query="SELECT * from panier where id_sess = '$sess'";
$results = mysqli_query($connect, $query) or die(mysql_error());
```

Figure 14:Code PHP en cas de suppression

Le code ci-dessus calcule le montant totale du panier

```
#calcul du montant totale du panier

$totalquantite=0;
$query="SELECT * from panier where id_sess = '$sess'";
$results = mysqli_query($connect, $query) or die(mysql_error());
while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) {
extract($row);
$totalquantite = $totalquantite + $p_quantite;
}
#affichage du panier
if ( ! isset($MontantTotal)) {
$MontantTotal=0;
}
$totalquantite=0;
$query = "SELECT * FROM panier WHERE id_sess = '$sess'";
$results = mysqli_query($connect, $query) or die (mysql_query());
if(mysqli_num_rows($results)==0)
{
echo "<div style='width:200px; margin:auto;'>Votre panier est vide</div> ";
}
else
{
?>
```

Figure 15:code PHP de calcul du montant totale de panier

5.2 Le code d'affichage du panier :

Après avoir ajouté des produits au panier, on aura besoin d'un code PHP pour afficher tous les produits sélectionnés dans le panier et modifier le contenu du panier si nécessaire.

Le code ci-dessous a pour tâche d'afficher les articles sélectionnés dans le panier et de les gérer.

Une requête SQL est exécutée pour vérifier s'il y a un produit dans le panier avec l'ID de session donné. S'il y en a, cela signifie que le visiteur a déjà ajouté un ou plusieurs produits au panier. Dans ce cas, tous les éléments stockés dans le panier, ainsi que leurs quantités respectives, s'affichent à l'écran.

Si aucun produit ne correspond à l'ID de session indiqué dans la table de panier, cela signifie qu'aucun produit n'est dans le panier. Un message s'affiche à l'écran indiquant que le visiteur a "aucun produit sélectionné dans le panier".

```
102 <?php
103 while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) { extract($row);
104 echo "<tr><td>". $p_itemcode."</td>";
105 echo "<td><form method='POST' action='panier.php?action=changer&icode=$p_itemcode'>
106 <input type='text' name='modified_quantite' size='2'
107 value='$p_quantite'>.</td><td>";
108 echo $p_item_name. "</td><td>";
109 echo $p_prix."</td><td>";
110 $totalquantite = $totalquantite + $p_quantite;
111 $totalprix = number_format($p_prix * $p_quantite, 2);
112 $MontantTotal=$MontantTotal + ($p_prix * $p_quantite);
113 echo $totalprix;
114 echo "</td><td><input type='submit' name='Submit' value='Changer Quantite'></form></td><td>";
115 echo "<form method='POST' action='panier.php?action=supprimer&icode=$p_itemcode'>";
116 echo "<input type='submit' name='Submit' value='Supprimer'></form></td></tr>";
117 echo "<tr color:blue \ " ><td>Totale</td><td>$totalquantite</td><td></td><td></td><td>";
118 $MontantTotal = number_format($MontantTotal, 2);
119 echo $MontantTotal;
120 echo "</td></tr></table><br>";
121 <table border="0" style="margin:auto;">
122 <tr><td style="padding: 10px;">
123 <form method="POST" action="panier.php?action=vider">
124 <input type="submit" name="Submit" value="Vider" style="font-family:verdana; font-size:150%;" >
125 </form></td><td>
126 <form method="POST" action="verification.php">
127 <input id="PanierMontant" name="PanierMontant" type="hidden" value= "<?php echo $MontantTotal ; ?>">
128 <input type="submit" name="Submit" value="Checkout" style="font-family:verdana; font-size:150%;" >
129 </form></td></tr></table> <?php } ?>
```

Figure 16:Code PHP d'affichage du panier

5.3 Le code PHP de vérification d'identification :

Une fois que les visiteurs ont ajouté des articles au panier et souhaitent les acheter, ils peuvent cliquer toujours sur le bouton « check-out » situé au bas du panier. Lorsqu'ils cliquent sur le bouton Commander, le code vérifie si l'utilisateur est connecté ou non.

Pour conserver les informations d'expédition pour la livraison d'un produit, le visiteur doit s'inscrire sur le site et doit être connecté. Si le visiteur n'est pas encore signé, deux liens s'affichent : l'un qui permet à l'utilisateur de se connecter et l'autre de créer un compte si l'utilisateur n'est pas encore inscrit.

En fonction de l'état actuel du visiteur, ils peuvent soit créer un nouveau compte, soit se connecter au site.

```
<?php include('topmenu.php');
if (session_status() == PHP_SESSION_NONE) { session_start();}
$connect = mysqli_connect("localhost", "root", "", "shopping") or die("Serveur non connecter");
$PanierMontant=0;
$PanierMontant = $_POST['PanierMontant'];
$_SESSION['PanierMontant']=$PanierMontant;
if (isset($_SESSION['email'])) { $email_address=$_SESSION['email'];}
if (isset($_SESSION['pass'])) { $password=$_SESSION['pass'];}
if ((isset($_SESSION['email']) && $_SESSION['email'] != "") || (isset($_SESSION['pass']) && $_SESSION['pass'] != ""))
{ $sess = session_id();
  $query="select * from panier where id_sess = '$sess'";
  $result = mysqli_query($connect, $query) or die(mysql_error());
if(mysqli_num_rows($result)>=1){
echo "<meta http-equiv='Refresh' content='0; url=detailsLivraison.php' />";
}
else { echo "Vous pouvez faire des achats en sélectionnant des articles dans le menu";}
}
else{ ?>
  <div class="message">
  <p>Vous n'êtes actuellement pas connecté à notre système.</p><br>
  <p>Vous ne pouvez acheter que si vous êtes connecté.</p><br> </div>
  <div class="div">
    <div class="item-div-check"> Si vous êtes déjà inscrit
    <a href="login.php">cliquer ici</a> pour s'identifier</div>
    <div class="item-div-check">
    Si vous souhaitez créer un compte <a href="signup.php">cliquer ici</a> pour inscrire.
  </p>
  </div></div>
<?php
}
?>
```

Figure 17:Code PHP de vérification d'identification

5.4 Le code de récupération de détails de la livraison :

Si l'utilisateur est déjà connecté au site, sera rediriger vers une autre page pour fournir des informations de livraison pour la livraison des produits.

Lorsque La session est lancée. La connexion est établie avec le serveur MySQL et la base de données d'expédition est sélectionnée.

Une instruction SQL est exécutée pour récupérer tous les détails du visiteur (y compris le nom, adresse, ville, pays, code postal et numéro de téléphone) dans le tableau des clients. Ces informations sont affichées à l'écran. Les informations d'expédition sont également affichées. Les visiteurs peuvent obtenir la livraison du produit en utilisant l'adresse déjà fournie créer le compte ou ils peuvent fournir de nouvelles informations d'expédition.

Après avoir fourni les informations d'expédition, le visiteur peut cliquer sur le bouton « Continuer » pour aller vers la page de paiement.

```
<?php
include('topmenu.php');
if (session_status() == PHP_SESSION_NONE) { session_start(); }
if (isset($_SESSION['PanierMontant'])) {
    $PanierMontant=$_SESSION['PanierMontant'];
}
$conn = mysqli_connect("localhost", "root", "", "shopping") or die("Serveur non connecter");
$email_address="";
if (isset($_SESSION['email'])) { $email_address=$_SESSION['email']; }
if (isset($_SESSION['pass'])) { $password=$_SESSION['pass']; }
if ((isset($_SESSION['email']) && $_SESSION['email'] != "") || (isset($_SESSION['pass']) && $_SESSION['pass'] != ""))
{
    $query = "SELECT * FROM client where email_address like '$email_address'
and password like (PASSWORD('$password'))";
    $results = mysqli_query($conn, $query) or die(mysqli_error());
    $row = mysqli_fetch_array($results, MYSQLI_ASSOC);
    extract($row);
}
?>
<h1 style="text-align:center;padding:30px;">Information Du livraison</h1>
<div class="main">
<form action="detailsPaiement.php" method="post" class="form">
<h2 class="name">Email: </h2><input class="company" type="text" name="email_address" value="<?php echo $email_address; ?>">
<h2 class="name">Nom Complet: </h2><input class="company" type="text" name="nom" value="<?php echo $nom; ?>"></>
<h2 class="name">Adresse: </h2><input class="company" type="text" name="Adresse" value="<?php echo $address_line1; ?>">
<h2 class="name">Ville: </h2><input class="company" type="text" name="ville" value="<?php echo $ville; ?>">
<h2 class="name">Pays:</h2><input class="company" type="text" name="pays" value="<?php echo $pays; ?>">
<h2 class="name">Code postale: </h2><input class="company" type="text" name="code_p" value="<?php echo $code_postal; ?>">
<h2 class="name">Telephone: </h2><input class="company" name="telephone" value="<?php echo $telephone; ?>">
    <button class="btn-p" type="submit">Continuer</button>
</form> </div>
</body>
</html>
```

Figure 18:Code de formulaire de livraison

5.5 Le code PHP d'ajout d'informations de livraison du client :

Le code PHP ci-dessous permet de mise à jour les informations de livraison du client dans la base de données.

```
<?php
include('topmenu.php');
if (session_status() == PHP_SESSION_NONE) { session_start();}
if (isset($_SESSION['PanierMontant'])) { $_PanierMontant=$_SESSION['PanierMontant'];}
$nom=$_POST['nom'];
$adresse = $_POST['Adresse'];
$ville = $_POST['ville'];
$codep = $_POST['code_p'];
$pays = $_POST['pays'];
$telephone= $_POST['telephone'];
$email=$_SESSION['email'];
$connect = mysqli_connect("localhost", "root", "", "shopping") or die("Serveur non connecter");
$sql = "UPDATE client SET address_line1='$adresse',ville='$ville',code_postal='$codep',pays='$pays',telephone='$telephone' WHERE email_address='$email'";
$results = mysqli_query($connect, $sql) or die(mysqli_error());
```

Figure 19:Code PHP de mise à jour des informations de livraison.

5.6 Le code de formulaire de remplissage d'information de paiement :

Le code html ci-dessous permet de recevoir les informations de paiement du client.

```
<body>
<h1 style="text-align:center;padding:30px;">Formulaire de paiement online</h1>
<div class="main">
<form action="commander.php" method="post" class="form" id="payment-form">
<h2 class="name">Numero de la carte: </h2><input class="company" type="text" name="carteNum" value="<?php echo $NumCarte; ?>">
<h2 class="name">CVC: </h2><input class="company" type="text" name="card-cvc" >
<h2 class="name">Nom complet: </h2><input class="company" type="text" name="Nom" value="<?php echo $carteNom; ?>">
<h2 class="name">Expiration (MM/AAAA): </h2><input class="company" type="text" name="Expiration" value="<?php echo $expiration_date; ?>" >
<button class="btn-p" type="submit">Continuer</button>
</form>
</div>
</body>
```

Figure 20:code html de formulaire de paiement.

5.7 Le code PHP de mise à jour des informations de paiement :

Le code PHP ci-dessous permet de mise à jour les informations de paiement du client dans la base de données

```
<?php
$sql4= "UPDATE payment_details SET date_pay='$date', email_address='$email',
nomCli='$nom', carteNom='$Nom', NumCarte='$carteNum', expiration_date='$Expiration_date',montant_payer='$MontantTotal'
Where email_address='$email'";

if ($connect->query($sql4) === TRUE) {
    echo "<meta http-equiv='Refresh' content='0; url=commande.php' />";
} else {
    echo "Erreur: " . $sql4 . "<br>" . $connect->error;
}
?>
```

Figure 21:Code PHP de mise à jour des information.

5.8 L'ajout des informations de la commande :

Tous les enregistrements de la table de panier avec l'ID de session donné (du même utilisateur) sont extraits un par un et stocké dans la table Orders.

Après cela, tous les articles de la table du panier avec l'ID de session donné sont supprimés. L'utilisateur voit la "Commande Acceptation » et l'ID de la commande s'affiche pour une communication future, puis il sera redirigé vers une autre page pour télécharger La facture. À la fin, la session est détruite.

```
<?php
$MontantTotal=0;
if (session_status() == PHP_SESSION_NONE) {session_start();}
if (isset($_SESSION['PanierMontant'])){ $PanierMontant=$_SESSION['PanierMontant'];}
$connect = mysqli_connect("localhost", "root", "", "shopping") or die("erreur.");
$email= $_SESSION['email'] ;
$query = "SELECT * FROM client where email_address like '$email'";
$results = mysqli_query($connect, $query) or die(mysql_error());
$row = mysqli_fetch_array($results, MYSQLI_ASSOC);
extract($row);
$date = date("Y-m-d");
$seSSID = session_id();
$query = "SELECT * FROM panier WHERE id_sess='$seSSID'";
$results = mysqli_query($connect, $query) or die(mysql_error());
while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) {
    extract($row);
    $MontantTotal=$MontantTotal + ($p_prix * $p_quantite);
    $connect = new mysqli("localhost", "root", "", "shopping") or die("Erreur de connection.");
    $sql3 = "INSERT INTO orders (code, nomP, quantite, prix, cmd_date, adressd_livraison, email_address, nom_client)
VALUES ('$p_code', '$p_titre', '$p_quantite', '$p_prix', '$date', '$ville', '$email', '$nom')";
    if ($connect->query($sql3) === TRUE) {
        $sql5 = "UPDATE produits set Quantite=Quantite-'$p_quantite' where Code='$p_code'";
        if ($connect->query($sql5) === TRUE) {
        } else { echo "Erreur: " . $sql5 . "<br>" . $connect->error;
        }
    }
    echo "<script> alert(\"Commande a été passer avec succès\")</script>";
}
else { echo "Erreur: " . $sql3 . "<br>" . $connect->error;
}
}
$query = "DELETE FROM panier WHERE id_sess='$seSSID'";
$delete = mysqli_query($connect, $query) or die(mysql_error());
session_destroy();
?>
```

Figure 22:Code PHP d'enregistrement de la commande.

5.9 Le code PHP de facture de la commande :

Après la validation de la commande le client sera redirigé vers la page PDF de la facture de la commande générer par le logiciel TCPDF.

Cette facture contient les informations de livraison les produits acheter et le montant totale à payer.

```
<?php
$id=$_REQUEST["icode"]; $MontantTotal=$_REQUEST["montant"];
function fetch_data(){
    $Montant=0; $id=$_REQUEST["icode"]; $output = '';
    $conn = mysqli_connect("localhost", "root", "", "shopping");
    $sql = "SELECT * FROM orders where order_no = '$id'";
    $result = mysqli_query($conn, $sql);
    while($row = mysqli_fetch_array($result))
    { extract($row);$Montant=$Montant + ($prix * $quantite);
    $output .= '<tr style="text-align:center;">
    <td>'. $row["code"].'</td> <td>'. $row["nomP"].'</td><td>'. $row["prix"].' $</td><td>'. $row["quantite"].'</td><td>'. $Montant.' $</td></tr> ';
    } return $output; }
$conn = mysqli_connect("localhost", "root", "", "shopping");
$sql = "SELECT * from client INNER join orders where order_no='$id'";
$result = mysqli_query($conn, $sql);
while($row = mysqli_fetch_array($result)) { extract($row);
include('library/tcpdf.php');
$pdf= new TCPDF('P','mm','A4');
$pdf->SetMargins(PDF_MARGIN_LEFT, '20', PDF_MARGIN_RIGHT);
$pdf->setPrintHeader(false);$pdf->setPrintFooter(false);
$pdf->AddPage();
$pdf->SetFont('Helvetica','',14);
$pdf->WriteHTML("<h1 style='text-align:center;'>SAMI STORE </h1> <h2 style='text-align:center; color:green'>Facture de commande</h2>");
$pdf->WriteHTML("<br><h4>Numero commande : ".$id."</h4>");
$pdf->WriteHTML("<h4>Nom client : ".$row["nom_client"]."</h4>");
$pdf->WriteHTML("<h4>Date de la commande : ".$row["cmd_date"]."</h4>");
$pdf->WriteHTML("<h4>Adresse de livraison : ".$row["address_line1"].", ".$row["ville"].", ".$row["pays"].", ".$row["code_postal"]."</h4>");
$content = ' <h1></h1> <table border="1" cellspacing="0" cellpadding="3">
<tr style="font-weight:bold; text-align:center; color:blue;">
<th>Code Produit</th><th>Article</th><th>Prix </th><th> Quantite </th> <th> Montant a payer </th></tr>';
$content .= fetch_data(); $content .= '</table>';
$pdf->writeHTML($content);
$pdf->WriteHTML("<br><h4 style='color:green;'>Montant totale a payer : ".$MontantTotal." $</h4>");
$pdf->output(); }
>>
```

Figure 23:Code PHP de la facture

Ces pages ci-dessous existe seulement dans l'espace administrateur

5.10 Page d'ajout d'un produit :

La figure suivante représente le formulaire html pour ajout de produit.

```
1 <html lang="en">
2 <head>
3   <link rel="stylesheet" type="text/css" href="styles.css">
4 </head>
5 <body>
6 <header><h1>SAMI</h1></header>
7 <h1 style="text-align:center;padding:10px;">Ajouter un produit</h1>
8 <div class="main">
9   <form action="ajouter.php" method="post" class="form">
10    <h2 class="name">Code </h2><input class="company" type="text" name="Code">
11    <h2 class="name">Titre</h2><input class="company" type="text" name="Titre">
12    <h2 class="name">Marque</h2><input class="company" type="text" name="Marque">
13    <h2 class="name">Modele</h2><input class="company" type="text" name="Modele">
14    <h2 class="name">Poids </h2><input class="company" type="text" name="Poids">
15    <h2 class="name">Dimensions </h2><input class="company" type="text" name="Dimensions">
16    <h2 class="name">Categorie</h2>
17    <select class="option" name="Categorie">
18      <option disabled="disabled" selected="selected">--Choisir la Categorie--</option>
19      <option>telephones</option>
20      <option>ordinateurs</option>
21      <option>Cameras</option>
22      <option>livres</option>
23    </select>
24    <h2 class="name">Quantite </h2><input class="company" type="text" name="Quantite">
25    <h2 class="name">Prix </h2><input class="company" type="text" name="Prix">
26    <h2 class="name">Image </h2><input class="company" type="text" name="Image"><br>
27    <button class="btn-p" type="submit">Ajouter</button>
28  </form> </div>
29 </body>
30 </html>
```

Figure 24: Code html de formulaire d'ajout de produits.

La figure suivante représente le code PHP pour ajout de produit dans la base de données.

```
1 <?php
2 $mysqli = new mysqli("localhost", "root", "", "shopping");
3 if ($mysqli->connect_error) {
4   die("Connection failed: " . $mysqli->connect_error);
5 }
6 $Code = $_POST["Code"];
7 $Titre = $_POST["Titre"];
8 $Marque = $_POST["Marque"];
9 $Modele = $_POST["Modele"];
10 $Poids = $_POST["Poids"];
11 $Dimensions = $_POST["Dimensions"];
12 $Categorie = $_POST["Categorie"];
13 $Quantite = $_POST["Quantite"];
14 $Prix = $_POST["Prix"];
15 $Image = $_POST["Image"];
16 $sql = "INSERT INTO produits (Code,Titre,Marque,Modele,Poids,Dimensions,Categorie,Quantite,Prix,Image)
17 VALUES ('$Code','$Titre', '$Marque','$Modele','$Poids','$Dimensions','$Categorie','$Quantite','$Prix','$Image')";
18 if ($mysqli->query($sql) === TRUE) {
19   >>
20   <meta http-equiv="refresh" content="0; url=index.php" />
21   <?php
22 } else {
23   echo "Error: " . $sql . "<br>" . $mysqli->error;
24 }
25 $mysqli->close();
26 >>
```

Figure 25: Code PHP d'ajout de produits

5.11 Page de mise à jour des produits :

La mise à jour concerne les modifications des informations des produits.

La figure suivante représente le formulaire html pour modification de produit.

```
9  ?>
10 <!DOCTYPE html>
11 <html lang="en">
12 <head>
13   <link rel="stylesheet" type="text/css" href="styles.css">
14 </head>
15 <body>
16 <header><h1>SAMI</h1></header>
17   <h1 style="text-align:center;padding:10px;">Modifier un produit</h1>
18 <div class="main">
19   <form action="modifier.php" method="post" class="form">
20     <h2 class="name">Code </h2><input class="company" type="text" name="Code" value="<?php echo $Code; ?>">
21     <h2 class="name">Titre</h2><input class="company" type="text" name="Titre" value="<?php echo $Titre; ?>">
22     <h2 class="name">Marque</h2><input class="company" type="text" name="Marque" value="<?php echo $Marque; ?>">
23     <h2 class="name">Modele</h2><input class="company" type="text" name="Modele" value="<?php echo $Modele; ?>">
24     <h2 class="name">Poids </h2><input class="company" type="text" name="Poids" value="<?php echo $Poids; ?>">
25     <h2 class="name">Dimensions </h2><input class="company" type="text" name="Dimensions" value="<?php echo $Dimensions; ?>">
26     <h2 class="name">Categorie</h2>
27     <select class="option" name="Categorie">
28       <option disabled="disabled" selected="selected">--Choisir la Categorie--</option>
29       <option>telephones</option>
30       <option>ordinateurs</option>
31       <option>Cameras</option>
32       <option>livres</option>
33     </select>
34     <h2 class="name">Quantite </h2><input class="company" type="text" name="Quantite" value="<?php echo $Quantite; ?>">
35     <h2 class="name">Prix </h2><input class="company" type="text" name="Prix" value="<?php echo $Prix; ?>">
36     <h2 class="name">Image </h2><input class="company" type="text" name="Image" value="<?php echo $Image; ?>">
37     <br>
38     <button class="btn-p" type="submit">Modifier</button>
39   </form> </div>
40 </body>
41 </html>
```

Figure 26: Code html de formulaire de mise à jour de produits.

```
1  <?php
2  $mysqli = new mysqli("localhost", "root", "", "shopping");
3  if ($mysqli->connect_error) {
4    die("Connection failed: " . $mysqli->connect_error);
5  }
6  $Code_produit = $_POST["Code"];
7  $Titre_produit = $_POST["Titre"];
8  $Marque_produit = $_POST["Marque"];
9  $Modele_produit = $_POST["Modele"];
10 $Poids_produit = $_POST["Poids"];
11 $Dimensions_produit = $_POST["Dimensions"];
12 $Categorie_produit = $_POST["Categorie"];
13 $Quantite_produit = $_POST["Quantite"];
14 $Prix_produit = $_POST["Prix"];
15 $Image_produit = $_POST["Image"];
16 $sql = "UPDATE produits set Titre='$Titre_produit', Marque='$Marque_produit',Modele='$Modele_produit',Poids='$Poids_produit',
17 Dimensions='$Dimensions_produit',Categorie='$Categorie_produit',Quantite='$Quantite_produit',
18 Prix='$Prix_produit',Image='$Image_produit' where Code='$Code_produit'";
19 if ($mysqli->query($sql) === TRUE) {
20   ?>
21   <meta http-equiv="refresh" content="0; url=index.php" />
22   <?php
23     echo "le produit est ajouter avec succes";
24   } else {
25     echo "Error: " . $sql . "<br>" . $mysqli->error;
26   }
27   $mysqli->close();
28   ?>
```

Figure 27: Code PHP de mise à jour de produits.

5.12 Page de suppression d'un produit :

L'administrateur doit fournir le code produit dans la zone texte fournit pour supprimer un produit.

```
1 <!DOCTYPE html>
2 <html lang="en" dir="ltr">
3   <head>
4     <meta charset="utf-8">
5     <link rel="stylesheet" href="styles.css">
6     <title></title>
7   </head>
8   <body>
9     <header>
10      <h1>SAMI</h1>
11    </header>
12    <h1 style="text-align:center;margin-top:20px;">Supprimer un produit</h1>
13    <div style="margin-top:50px;"class="main">
14      <form action="supprimer.php" method="post" class="form">
15        <h2 class="name">Code </h2><input class="company" type="text" name="id">
16        <button type="submit" class="btn-p">Supprimer</button>
17      </div>
18    </body>
19  </html>
20
```

Figure 28: Code html de formulaire de suppression

Une fois l'administrateur entre le code de produit le script ci-dessous permet de supprimer ce dernier dans la base de données.

```
1 <?php
2 $mysqli = new mysqli("localhost", "root", "", "shopping");
3 if ($mysqli->connect_error) {
4     die("Connection failed: " . $mysqli->connect_error);
5 }
6 $id=$_POST["id"];
7 $sql="DELETE FROM produits WHERE code='$id'";
8 if ($mysqli->query($sql) === TRUE) {
9     ?>
10    <meta http-equiv="refresh" content="0; url=index.php" />
11    <?php
12 } else {
13     echo "Error: " . $sql . "<br>" . $mysqli->error;
14 }
15 $mysqli->close();
16 ?>
17
```

Figure 29:Code PHP de suppression.

5.13 La page d'affichage des commandes :

L'administrateur a le droit de voir toutes les commandes passer par les clients dans un tableau qui contient toutes les informations sur la commande.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <link rel="stylesheet" href="../styles.css">
    <title></title>
  </head>
  <body>
    <header><h1>SAMI</h1></header>
    <div class="container-box">
      <div class="item-2">
        <table class="table-cart" border="1" align="center" cellpadding="5">
          <caption>Statistique des commandes</caption>
          <tr><td>Cmd</td><td>Email</td><td>Date</td><td>Code</td><td>Qauntite</td><td>Produit</td></tr>
          <?php
            $connect = mysqli_connect("localhost", "root", "", "shopping") or die("Serveur non connecter");
            $query = "SELECT * FROM orders";
            $results = mysqli_query($connect, $query) or die (mysql_query());
            while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) { extract($row);
              echo "<tr><td>". $order_no."</td>";
              echo "<td>". $email_address. "</td>";
              echo "<td>". $cmd_date. "</td>";
              echo "<td>". $code."</td>";
              echo "<td>". $quantite."</td>";
              echo "<td>". $nomP."</td></tr>";
            }
          >
        </table>
      </div>
    </div>
  </body>
</html>
```

5.14 Code PHP de la page des statistiques :

L'administrateur reçoit tous les statistiques qui concernent le magasin en ligne.

Il reçoit toutes les informations des produits tel que la quantité vendue de chaque article ainsi que la quantité restante.

Il reçoit aussi toutes les informations des clients y compris le nombre de commande passer.

```
<?php
$totalquantite=0; $totalprix=0; $totalvendue=0; $totalprofit=0;
$connect = mysqli_connect("localhost", "root", "", "shopping") or die("Serveur non connecter");
$query = "SELECT * FROM produits";
$results = mysqli_query($connect, $query) or die (mysql_query());
while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) { extract($row);
    $qtn=0;
    $sql4="SELECT sum(quantite) as somme FROM orders where code='$Code'";
    $results1 = mysqli_query($connect, $sql4) or die (mysql_query());
    $totalprix=$totalprix+($Prix*$Quantite);
    $totalquantite=$totalquantite+$Quantite;
    echo "<tr><td>". $Code."</td><td>". $Titre. "</td><td>". $Prix."</td><td>". $Quantite."</td>";
    if($row1 = mysqli_fetch_array($results1, MYSQLI_ASSOC)){
        extract($row1); echo "<td>". $somme."</td>";
        $profit=$somme*$Prix;
        echo "<td>". $profit."</td></tr>";
    }
    $totalvendue=$totalvendue+$somme;
    $totalprofit=$totalprofit+$profit; }
echo "<tr style='color:#33eeff' \><td>Totale</td><td></td><td>$totalprix</td><td>$totalquantite</td><td>$totalvendue</td><td>$totalprofit</td></tr>";
}> </table> </div>

<div class="item-sta">
<table class="table-cart" border="1" align="center" cellpadding="5">
<caption>Statistiques des clients</caption>
<tr><td>Nom</td><td>Email</td><td>Pays</td><td>Ville</td><td>Nbr de cmd</td></tr>

<?php$connect = mysqli_connect("localhost", "root", "", "shopping") or die("Serveur non connecter");
$query = "SELECT * FROM client ORDER by ville Desc";
$results = mysqli_query($connect, $query) or die (mysql_query());
while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) {
    extract($row);
    $email=$email_address;
    $sql4="SELECT count(*) as cnt FROM orders where email_address='$email' ";
    $results1 = mysqli_query($connect, $sql4) or die (mysql_query());
    if($row1 = mysqli_fetch_array($results1, MYSQLI_ASSOC)){ extract($row1);
    echo "<tr><td>". $nom."</td><td>". $email_address. "</td><td>". $pays."</td><td>". $ville."</td><td>". $cnt."</td></tr>"; }
}
}> </table> </div> </div>
```

Figure 30:code PHP de page de statistique

6. Interfaces principales :

6.1 Page d'accueil :

La page d'accueil est la première page de notre application faite pour nos clients, cette page contient des images et des liens pour permettre à nos clients de bénéficier des différents services offerts par notre application

up to 25% off
FOR ALL LAPTOPS



Produit au meilleur prix.



S10
749.99 \$



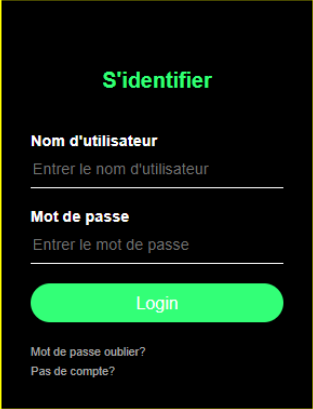
S8
859.42 \$



IphoneX
749.99 \$

6.2 Interface identification :

Cette page permet l'utilisateur d'accéder à l'application après avoir saisi son nom et son mot de passe pour accéder s'il est un client déjà existant.



S'identifier

Nom d'utilisateur
Entrer le nom d'utilisateur

Mot de passe
Entrer le mot de passe

Login

[Mot de passe oublier?](#)
[Pas de compte?](#)

sinon, cette page sera afficher au cas d'erreur d'informations ou le client n'est pas inscrit.



localhost/store/validierClient.php

SAMI

Menu Produits Contact À propos

[S'identifier](#) [S'inscrire](#)

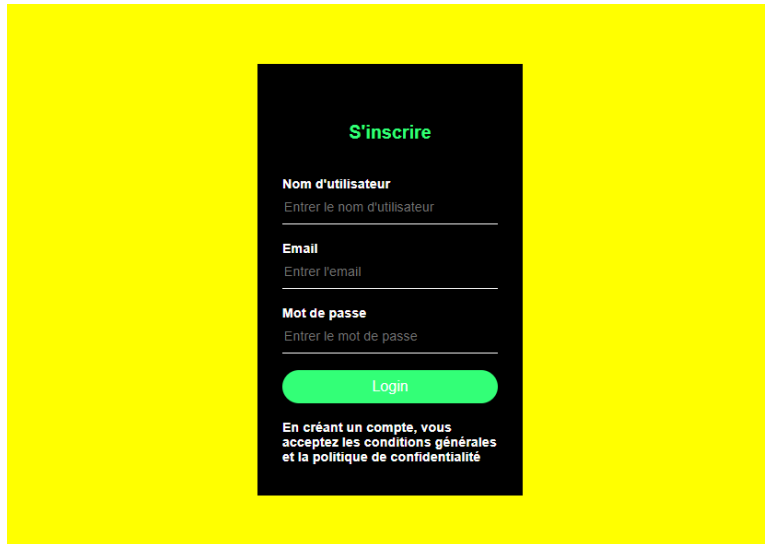
Adresse email ou mot de passe invalide

Pas Inscrits?
[Cliquer ici](#) pour inscription.

[Cliquer ici](#) pour essayer.

6.3 Interface inscription :

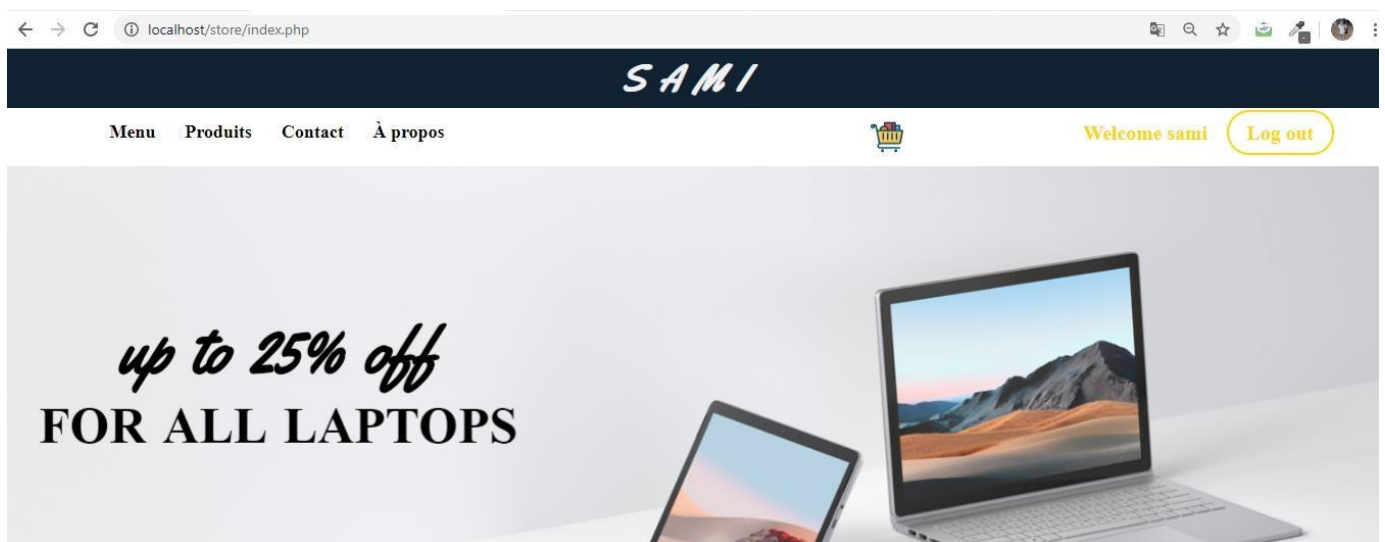
Cette page permet l'utilisateur de s'inscrire à l'application en fournissant son nom, adresse email et un mot de passe.

A registration form titled "S'inscrire" is displayed on a yellow background. The form is contained within a dark gray box. It includes three input fields: "Nom d'utilisateur" (with placeholder "Entrez le nom d'utilisateur"), "Email" (with placeholder "Entrez l'email"), and "Mot de passe" (with placeholder "Entrez le mot de passe"). Below these fields is a green "Login" button. At the bottom of the form, a small text line reads: "En créant un compte, vous acceptez les conditions générales et la politique de confidentialité".

Si email est invalide ou le mot de passe est très court cette page réaffichera avec un message d'erreur.

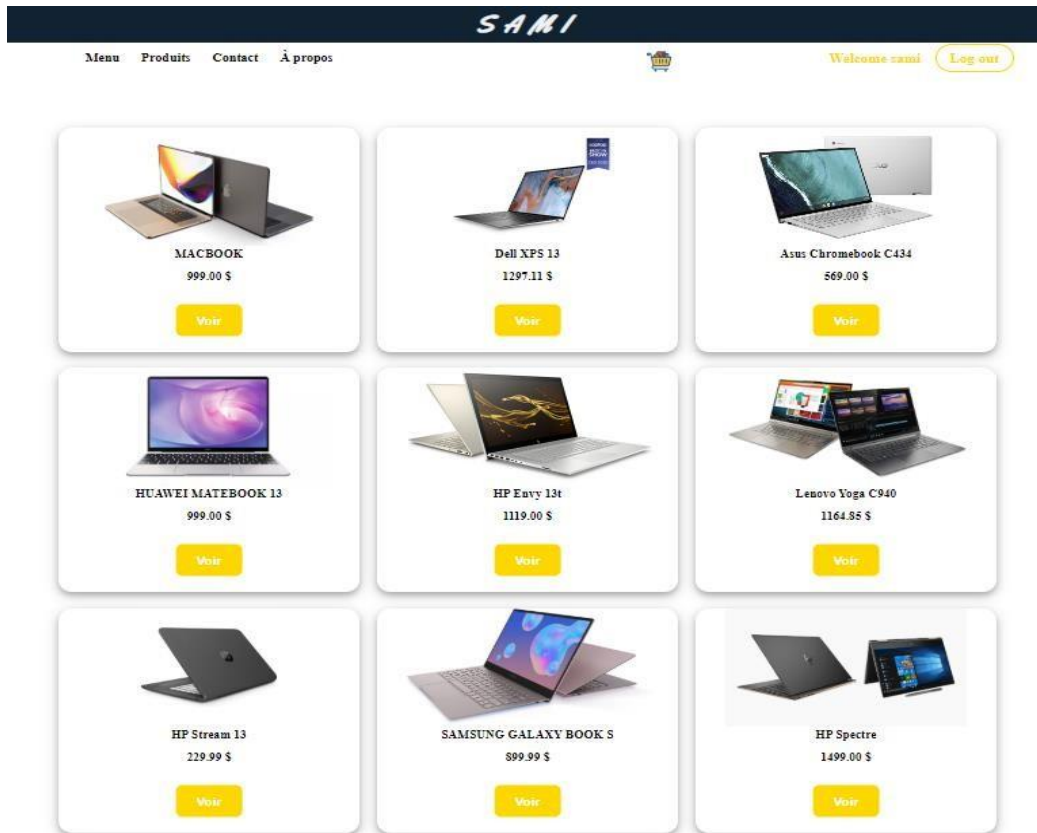
Si l'utilisateur est authentifié avec succès, il met à jour l'en-tête du site, en remplaçant le lien de connexion par le nom de l'utilisateur avec le message de bienvenue.

En outre, le lien d'inscription est remplacé par le lien de déconnexion.



6.4 Interface d'affichage des produits :

Cette page permet à l'utilisateur d'afficher les produits dont la catégorie est sélectionnée dans le menu déroulant



6.5 Interface d'affichage du détail du produit :

Les informations sur le produit récupérées s'affichent à l'écran ainsi que la description et le prix du produit




6.6 Interface du panier :

S'il y a produit dans le panier avec l'ID de session donné. Dans ce cas, tous les éléments stockés dans le panier, ainsi que leurs quantités respectives, s'affichent à l'écran.

Si aucun produit n'est dans le panier. Un message s'affiche à l'écran indiquant que le visiteur a "aucun produit sélectionné dans le panier".

SAMI

[Menu](#) [Produits](#) [Contact](#) [À propos](#)

 [S'identifier](#) [S'inscrire](#)

Code	Quantite	Titre	Prix	Prix Totale		
2	1	S10	749.99	749.99	Changer Quantite	Supprimer
564	2	Canon 600d	279.99	559.98	Changer Quantite	Supprimer
8	3	VIVOBOK S15	694.79	2,084.37	Changer Quantite	Supprimer
3214	1	HP Spectre	1499.00	1,499.00	Changer Quantite	Supprimer
3	2	S8	859.42	1,718.84	Changer Quantite	Supprimer
1234	1	MACBOOK	999.00	999.00	Changer Quantite	Supprimer
Totale	10			7,611.18		

[Vider](#) [Checkout](#)

6.7 Interface de vérification d'identification :

Une fois tu as terminé d'ajouter les produits voulus dans le panier tu peux cliquer sur le bouton « Checkout », le site web vas vérifier si tu es connecté sinon tu vas être rediriger vers la page de connexion ou la page d'inscription si vous êtes pas inscrit.

SAMI

[Menu](#) [Produits](#) [Contact](#) [À propos](#)

 [S'identifier](#) [S'inscrire](#)

Vous n'êtes actuellement pas connecté à notre système.

Vous ne pouvez acheter que si vous êtes connecté.

Si vous êtes déjà inscrit [cliquer ici](#) pour s'identifier

Si vous souhaitez créer un compte [cliquer ici](#) pour inscrire.

6.8 Interface de formulaire de livraison :

Après la vérification d'identification au site tu vas être rediriger vers cette interface pour fournir les informations de livraison.

The screenshot shows the SAMI website header with a dark blue bar containing the logo 'SAMI' and navigation links: Menu, Produits, Contact, À propos. A shopping cart icon is in the center, and 'Welcome sami' with a 'Log out' button is on the right. Below the header, the title 'Information Du livraison' is centered. The main form is an orange rounded rectangle with the following fields: Email (sami), Nom Complet (sami), Adresse (Boulevard Moulay ismail), Ville (Hoceima), Pays (maroc), Code postale (68000), and Telephone (123456). A yellow 'Continuer' button is at the bottom right of the form.

Si le client a déjà passé une commande les informations de livraison vont être afficher dans le formulaire pour les modifier s'il a changé d'adresse.

6.9 Interface de formulaire de paiement :

Après que le client a fournit les informations de livraison le client doit fournir les informations de paiement pour valider la commande.

The screenshot shows the SAMI website header, identical to the previous one. Below the header, the title 'Formulaire de paiement online' is centered. The main form is an orange rounded rectangle with the following fields: Numero de la carte, CVC, Nom complet, and Expiration (MM/AAAA). A yellow 'Continuer' button is at the bottom right of the form.

6.10 Facture PDF de la commande :

Une fois le site reçoive toutes les informations du client, panier, livraison et paiement la commande vas être enregistrer dans la base de donner pour être traiter valider et livrer et le client va reçoie le message de validation de la commande est un fichier PDF téléchargeable de la facture.

SAMI STORE

Facture de commande

Numero commande : 5edd820b7fa3a

Nom client : sami

Date de la commande : 2020-05-15

Adresse de livraison : Boulevard Moulay ismail, Hoceima, maroc, 68000

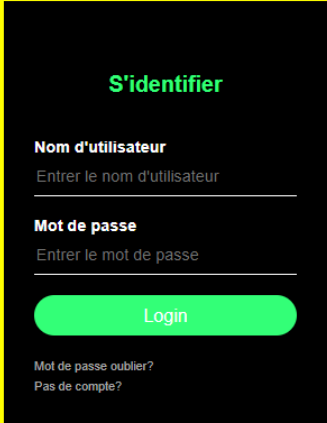
Code Produit	Article	Prix	Quantite	Montant a payer
8	VIVOBOK S15	694.79 \$	2	1389.58 \$
3214	HP Spectre	1499.00 \$	3	5886.58 \$
1234	MACBOOK	999.00 \$	3	8883.58 \$
4	IphoneX	749.99 \$	4	11883.54 \$
3	S8	859.42 \$	4	15321.22 \$
2	S10	749.99 \$	4	18321.18 \$
564	Canon 600d	279.99 \$	1	18601.17 \$

Montant totale a payer : 80286.85 \$

7. Interface d'administrateur :

7.1 Interface identification :

Cette page permet l'administrateur d'accéder à l'application aux interfaces des administrateur après avoir saisi son nom et son mot de passe pour accéder s'il est un déjà un administrateur.

A login form titled "S'identifier" is centered on a yellow background. The form has a dark blue header with the title in green. It contains two input fields: "Nom d'utilisateur" with the placeholder "Entrez le nom d'utilisateur" and "Mot de passe" with the placeholder "Entrez le mot de passe". Below these is a green "Login" button. At the bottom, there are two links: "Mot de passe oublier?" and "Pas de compte?".

Si email est invalide ou le mot de passe est très court cette page réaffichera avec un message d'erreur.

7.2 Interface principale de la page de l'administrateur :

Quand l'administrateur est authentifié avec succès il sera redirigé vers l'interface principale.

Cette interface lui permettra d'ajouter un administrateur, ajouter-supprimer ou modifier un produit ainsi que l'affichage des statistiques des achats et les informations qui concerne les commandes.



7.3 Interface d'ajouter un produit :

Cette interface est consacrée pour l'ajout des nouveaux produits. Elle est accessible à l'administrateur seulement.

Après la saisie des données dans ce formulaire ces derniers s'enregistrent directement à la base de données pour être afficher en site web.

S A M I

Ajouter un produit

Code	<input type="text"/>
Titre	<input type="text"/>
Marque	<input type="text"/>
Modele	<input type="text"/>
Poids	<input type="text"/>
Dimensions	<input type="text"/>
Categorie	<input type="text" value="--Choisir la Categorie--"/>
Quantite	<input type="text"/>
Prix	<input type="text"/>
image	<input type="text"/>
Description	<input type="text"/>

Ajouter

7.4 Interface de mise à jour des produits :

Cette interface est consacrée pour la mise à jour des produits. Elle est accessible à l'administrateur seulement.

S A M I

Modifier un produit

Code

Modifier

Après la saisie du code de produit le formulaire s'affichera avec les informations du produit qui existe déjà, les données dans ce formulaire vont s'enregistrer à la base de données après la validation des modifications.

S A M I

Modifier un produit

Code

Titre

Marque

Modele

Poids

Dimensions

Categorie

Quantite

Prix

image

Description

The MacBook Air in particular has a thin and light design. There are two versions of the MacBook Air:

Modifier

7.5 Interface de suppression des produits :

Cette interface est consacrée pour la suppression des produits. Elle est accessible à l'administrateur seulement.

S A M I

Supprimer un produit

Code

Supprimer

L'administrateur doit fournir le code produit pour le supprimer définitivement de la base de données.

7.6 Interface des statistiques :

Cette interface est consacrée pour afficher les statistiques du site web. Elle est accessible à l'administrateur seulement.

S A M I

Statistique des produits

Code	Titre	Prix	Qnt en stock	Qnt Vendue	Profit
2	S10	749.99	20	38	28499.62
3	S8	859.42	20	43	36955.06
4	IphoneX	749.99	20	56	41999.44
1234	MACBOOK	999.00	20	89	88911
8	VIVOBOK S15	694.79	20	43	29875.97
3214	HP Spectre	1499.00	20	45	67455
564	Canon 600d	279.99	20	84	23519.16
Totale		116643.6	140	398	317215.25

Statistiques des clients

Nom	Email	Pays	Ville	Nmbr de cmd
sami	sami	maroc	tanger	106
user	user@gmail.com	maroc	hoceima	3
adam	adam@gmail.com			13
admin	admin			0
sam	sam			0
adam	adam			0

7.7 Interface des commandes :

Cette interface est consacrée pour afficher les commandes passer par les clients. Elle est accessible à l'administrateur seulement.

S A M I

Statistique des commandes						
Cmd	Email	Date	Code	Quantite	Produit	
5ede8c2ade6cf	sami	2020-05-15	3	1	S8	
5ede8bace7552	sami	2020-05-15	3	1	S8	
5ede8b8b676ad	sami	2020-05-15	3	1	S8	
5ede8b80e5fed	sami	2020-05-15	3	1	S8	
5ede8b4e382c4	sami	2020-05-15	3	1	S8	
5ede8aa7262ea	sami	2020-05-15	3	1	S8	
5ede8a4c7afa2	sami	2020-05-15	3	1	S8	
5ede89ff8cf7a	sami	2020-05-15	3	1	S8	
5ede88434eba6	sami	2020-05-15	3214	1	HP Spectre	
5ede881d10984	sami	2020-05-15	3214	1	HP Spectre	
5ede87fef28e1	sami	2020-05-15	3214	1	HP Spectre	
5ede87ba65143	sami	2020-05-15	3214	1	HP Spectre	
5ede87a40e6f9	sami	2020-05-15	3214	1	HP Spectre	
5ede864830bb6	sami	2020-05-15	3214	1	HP Spectre	
5ede862fc24a3	sami	2020-05-15	3214	1	HP Spectre	
5ede8601d76ca	sami	2020-05-15	3214	1	HP Spectre	
5ede645a1d078	sami	2020-05-15	2	2	S10	
5ede642874a53	sami	2020-05-15	1234	1	MACBOOK	
5ede63d932cc6b	sami	2020-05-15	1234	1	MACBOOK	
5ede5f3ff61ea	sami	2020-05-15	2	1	S10	
5ede5f299666e	sami	2020-05-15	2	1	S10	
5ede5e9f6bf48	sami	2020-05-15	1234	1	MACBOOK	
5ede5e6c66dad	sami	2020-05-15	1234	1	MACBOOK	
5ede5e26f0d50	sami	2020-05-15	2	1	S10	
5ede5e26f0d50	sami	2020-05-15	1234	1	MACBOOK	
5ede5cd50d81c	sami	2020-05-15	2	1	S10	
5ede5ea7230c0	sami	2020-05-15	2	1	S10	

8. Conclusion :

Tout au long de la préparation de ce projet, j'ai essayé de mettre en pratique les connaissances acquises durant ce semestre et cela dans le but de réaliser une application de vente en ligne.