

RAPPORT DU PROJET

Beauty Bliss

ÉTUDIANT : ANDREEA SANDULESCU

GROUPE : 1230 AF

SOMMAIRE

INTRODUCTION	pg. 3
ARCHITECTURE DU SITE	pg. 4
• <i>Technologies et frameworks utilisés</i>	pg. 4
• <i>La structure des fichiers</i>	pg. 4
UTILISATION DE LA BASE DE DONEES ORACLE	pg. 5
• <i>Connexion à la base de données</i>	pg. 5
• <i>Interaction avec la base de données Oracle</i>	pg. 6
• <i>Avantages de l'utilisation de la base de données Oracle</i>	pg. 6
FONCTIONNALITES CLES	pg. 6
• <i>Ajout de produits au panier</i>	pg. 7
• <i>Modification de la quantité des produits dans le panier</i>	pg. 8
• <i>Passation de commande</i>	pg. 9
DESIGN ET EXPERIENCE UTILISATEUR	pg. 10
CONCLUSIONS ET AMELIORATIONS FUTURES	pg. 15

INTRODUCTION

Ce projet représente un site web de produits cosmétiques qui est une plateforme en ligne dédiée à la vente et à la promotion de produits cosmétiques.

Dans le cadre de ce projet, nous avons développé un site Web dynamique et interactif pour offrir une expérience agréable aux utilisateurs tout en leur permettant de naviguer et de sélectionner les produits qui correspondent à leurs besoins et à leurs préférences.

L'un des éléments clés de ce projet est l'utilisation de la base de données Oracle. La base de données joue un rôle essentiel dans le stockage et la gestion des informations relatives aux produits, aux commandes et aux utilisateurs. Grâce à Oracle, nous sommes en mesure de garantir une gestion efficace des données, une sécurité robuste et des performances optimales pour le site.

En utilisant la base de données Oracle, nous pouvons stocker des informations détaillées sur chaque produit, telles que son nom, sa description, son prix et sa disponibilité. Cela permet aux utilisateurs de parcourir les produits avec facilité et d'accéder à des informations précises lorsqu'ils prennent leur décision d'achat. De plus, la base de données enregistre les commandes passées par les utilisateurs, ce qui permet de suivre l'historique des achats et de faciliter la gestion des stocks.

L'intégration de la base de données Oracle dans notre projet présente de nombreux avantages. Elle offre une grande flexibilité en termes de requêtes et de manipulations de données, permettant ainsi une personnalisation et une évolutivité accrues du site. De plus, Oracle offre des fonctionnalités avancées pour la gestion des transactions, ce qui garantit l'intégrité des données et une expérience utilisateur fluide lors du processus de commande.

Dans la suite de ce rapport, nous explorerons plus en détail l'architecture du site, les fonctionnalités clés offertes aux utilisateurs, ainsi que l'interaction

spécifique avec la base de données Oracle. Nous examinerons également les aspects liés au design et à l'expérience utilisateur, en mettant en évidence les points forts du site et en identifiant les possibilités d'amélioration pour l'avenir.

ARCHITECTURE DU SITE

Technologies et frameworks utilisés :

- Le site utilise le langage de programmation PHP pour gérer les fonctionnalités et l'interaction avec la base de données;
- Pour l'interface utilisateur, HTML, CSS et JavaScript sont utilisés pour créer une expérience agréable et interactive.

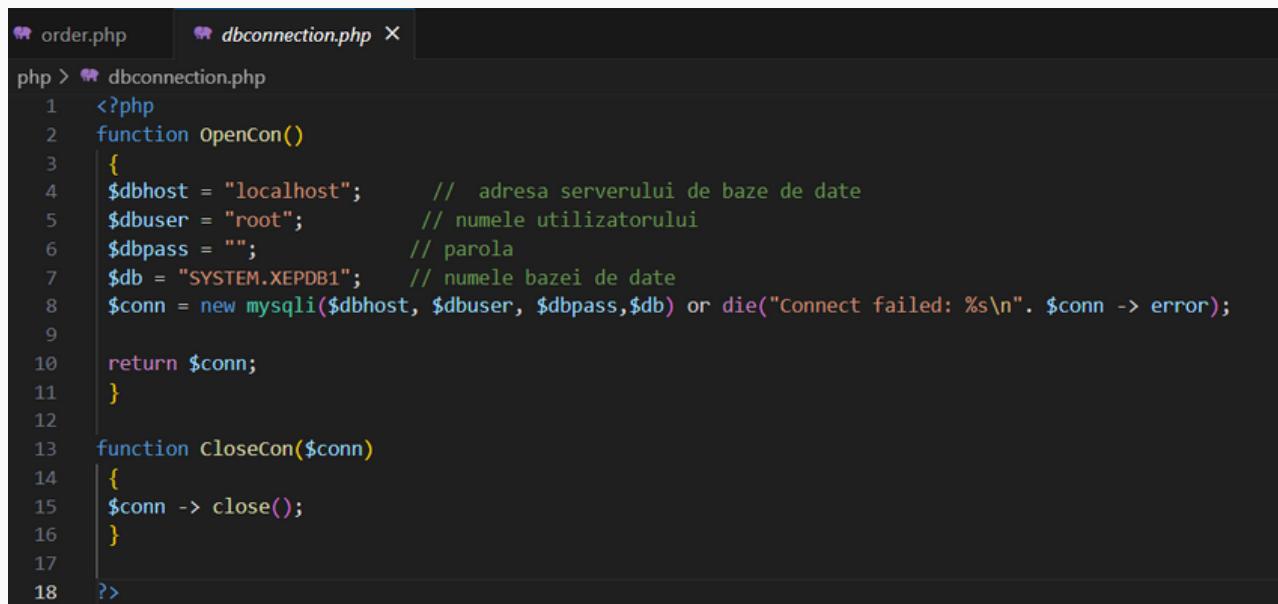
La structure des fichiers :

- Le fichier "dbconnection.php" gère la connexion à la base de données Oracle et fournit des fonctions pour ouvrir et fermer la connexion;
- La page "index.php" représente la page principale du site et affiche les produits cosmétiques disponibles;
- La page "newCart.php" présente le contenu du panier d'achat et permet à l'utilisateur d'ajouter, de diminuer ou de supprimer des produits du panier;
- La page "order.php" gère le processus de passation de commande et l'interaction avec la base de données Oracle pour enregistrer la commande.

UTILISATION DE LA BASE DE DONNÉES ORACLE

Connexion à la base de données :

- Le fichier "dbconnection.php" contient la fonction "OpenCon()" qui gère l'ouverture de la connexion à la base de données Oracle;
- Les paramètres nécessaires pour la connexion à la base de données, tels que l'hôte, le nom d'utilisateur, le mot de passe et le nom de la base de données, sont spécifiés dans ce fichier.



```
order.php dbconnection.php X
php > dbconnection.php
1  <?php
2  function OpenCon()
3  {
4      $dbhost = "localhost";      // adresa serverului de baze de date
5      $dbuser = "root";          // numele utilizatorului
6      $dbpass = "";              // parola
7      $db = "SYSTEM.XEPDB1";     // numele bazei de date
8      $conn = new mysqli($dbhost, $dbuser, $dbpass,$db) or die("Connect failed: %s\n". $conn -> error);
9
10     return $conn;
11 }
12
13 function CloseCon($conn)
14 {
15     $conn -> close();
16 }
17
18 ?>
```

Fig 1 : dbconnection.php

Interaction avec la base de données Oracle :

- Pour enregistrer les commandes des utilisateurs, le site utilise des requêtes PL / SQL pour insérer les informations de commande dans la table appropriée de la base de données Oracle;
- Des requêtes SELECT sont également utilisées pour récupérer des informations sur les produits à afficher sur le site et pour effectuer des vérifications et des validations.

Avantages de l'utilisation de la base de données Oracle :

- Oracle est un système de gestion de base de données réputé, offrant des fonctionnalités avancées de gestion des données, de sécurité et de performance;
- L'utilisation d'Oracle permet de stocker et de gérer efficacement les informations sur les produits, les commandes et les utilisateurs du site;
- Les requêtes PL / SQL peuvent être optimisées pour garantir des performances rapides et une expérience utilisateur fluide.

FONCTIONNALITÉS CLÉS

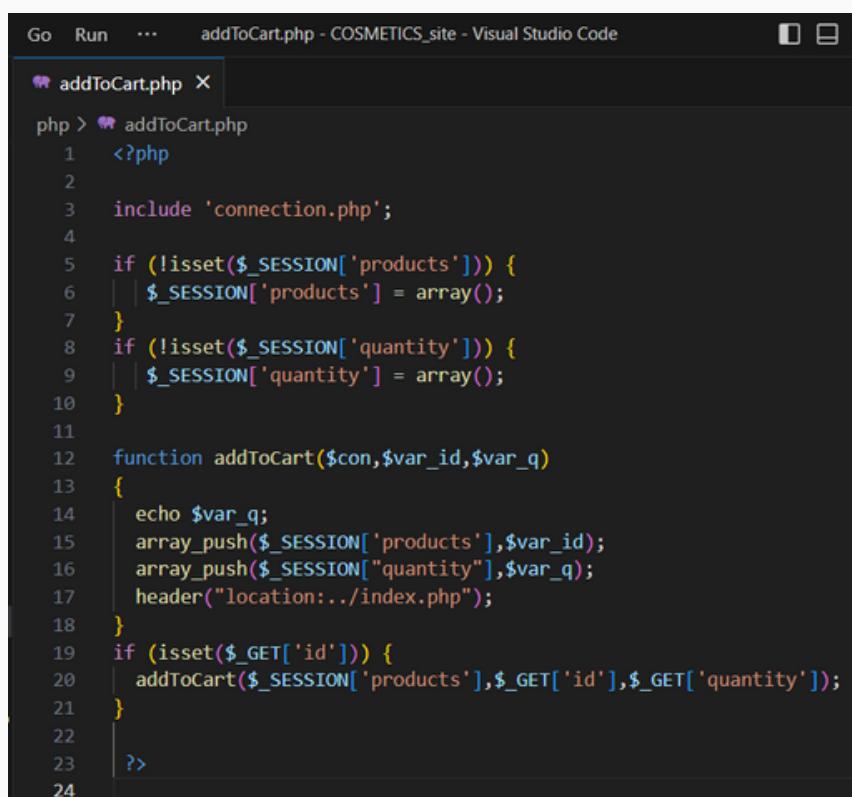
Le site de produits cosmétiques propose plusieurs fonctionnalités essentielles, notamment :

- L'affichage des produits cosmétiques disponibles avec leurs détails et leurs prix;
- La possibilité d'ajouter des produits au panier d'achat et de modifier les quantités;

- Le processus de passation de commande, y compris la saisie des informations de livraison et l'enregistrement dans la base de données Oracle;
- La gestion des commandes précédentes et l'accès à des informations détaillées sur les commandes.

Ajout de produits au panier (fig 2) :

- La fonction "addToCart()" est responsable de l'ajout des produits au panier d'achat;
- L'utilisateur peut ajouter des produits au panier via la page "index.php", où les produits disponibles sont affichés;
- Une fois que l'utilisateur a sélectionné un produit et la quantité souhaitée, ces informations sont transmises à la fonction "addToCart()" via la méthode GET;
- La fonction met à jour les variables de session "\$_SESSION['products']" et "\$_SESSION['quantity']" avec les produits et les quantités correspondantes.

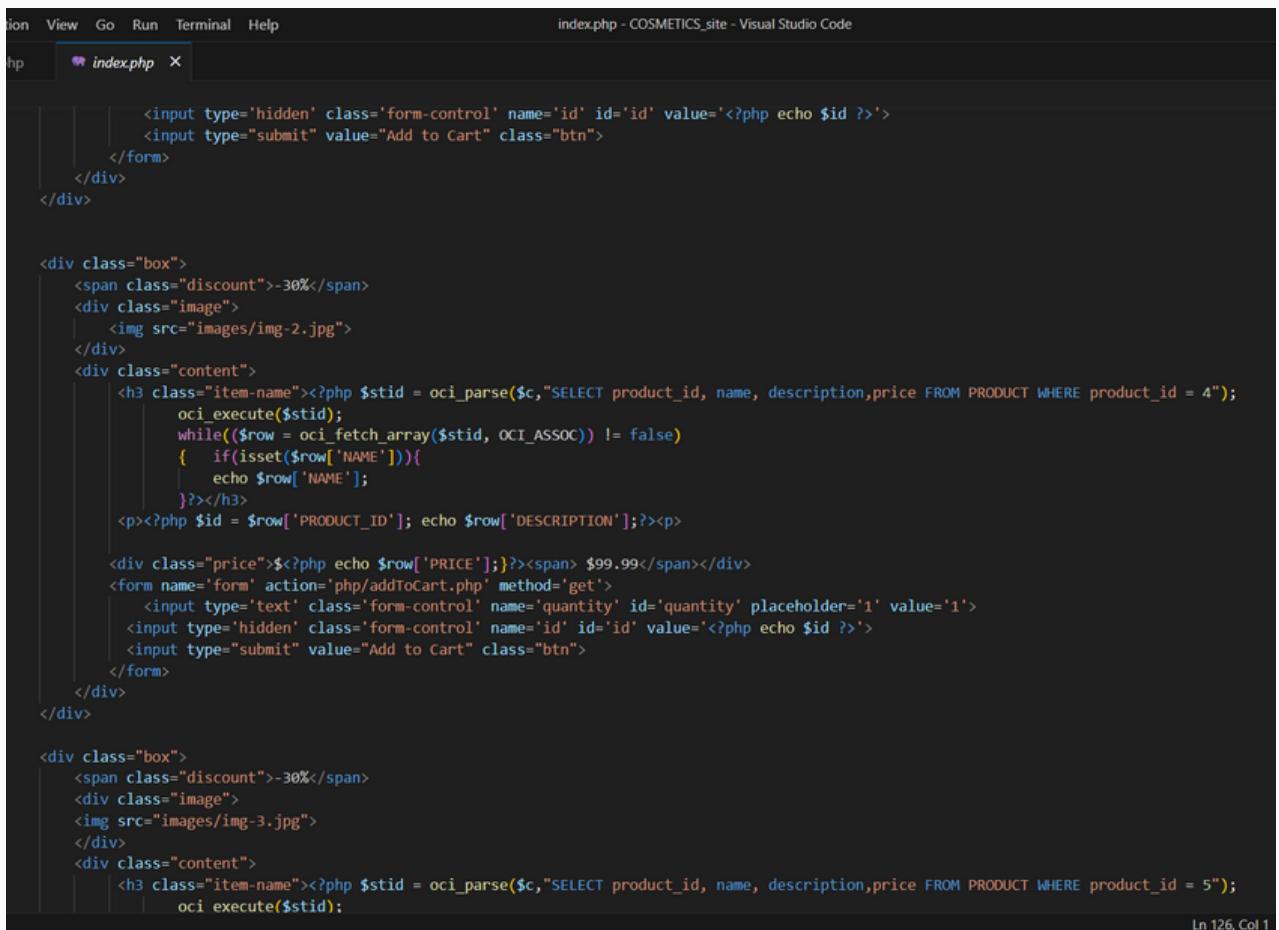


```

Go Run ... addToCart.php - COSMETICS_site - Visual Studio Code
addToCart.php x
php > addToCart.php
1 <?php
2
3 include 'connection.php';
4
5 if (!isset($_SESSION['products'])) {
6   | | $_SESSION['products'] = array();
7 }
8 if (!isset($_SESSION['quantity'])) {
9   | | $_SESSION['quantity'] = array();
10 }
11
12 function addToCart($con,$var_id,$var_q)
13 {
14   echo $var_q;
15   array_push($_SESSION['products'],$var_id);
16   array_push($_SESSION["quantity"],$var_q);
17   header("location:../index.php");
18 }
19 if (isset($_GET['id'])) {
20   addToCart($_SESSION['products'],$_GET['id'],$_GET['quantity']);
21 }
22
23 ?>
24

```

Fig 2 : addToCart.php



```

<?php
    <input type='hidden' class='form-control' name='id' id='id' value='<?php echo $id ?>'>
    <input type="submit" value="Add to Cart" class="btn">
</form>
</div>
</div>

<div class="box">
    <span class="discount">-30%</span>
    <div class="image">
        
    </div>
    <div class="content">
        <h3 class="item-name"><?php $stid = oci_parse($c,"SELECT product_id, name, description,price FROM PRODUCT WHERE product_id = 4");  

            oci_execute($stid);  

            while(($row = oci_fetch_array($stid, OCI_ASSOC)) != false)  

            {  

                if(isset($row['NAME']))  

                    echo $row['NAME'];  

            }?></h3>
        <p><?php $id = $row['PRODUCT_ID']; echo $row['DESCRIPTION'];?><p>
        <div class="price">$<?php echo $row['PRICE'];?><span> $99.99</span></div>
        <form name='form' action='php/addToCart.php' method='get'>
            <input type='text' class='form-control' name='quantity' id='quantity' placeholder='1' value='1'>
            <input type='hidden' class='form-control' name='id' id='id' value='<?php echo $id ?>'>
            <input type="submit" value="Add to Cart" class="btn">
        </form>
    </div>
</div>

<div class="box">
    <span class="discount">-30%</span>
    <div class="image">
        
    </div>
    <div class="content">
        <h3 class="item-name"><?php $stid = oci_parse($c,"SELECT product_id, name, description,price FROM PRODUCT WHERE product_id = 5");  

            oci execute($stid);>

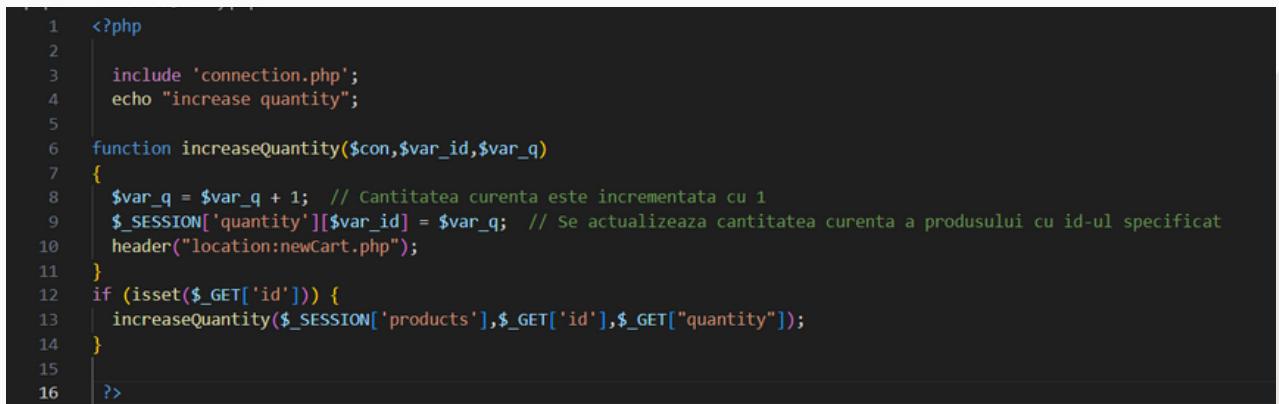
```

In 126 Col 1

Fig 3 : index.php

Modification de la quantité des produits dans le panier (Fig 4 + 5) :

- Les fonctions "increaseQuantity()" et "decreaseQuantity()" permettent à l'utilisateur d'augmenter ou de diminuer la quantité d'un produit dans le panier;
- En utilisant les liens "plus" et "moins" affichés dans la page "newCart.php", l'utilisateur peut modifier la quantité, et les fonctions mettent à jour la variable de session correspondante.



```

1  <?php
2
3      include 'connection.php';
4      echo "increase quantity";
5
6  function increaseQuantity($con,$var_id,$var_q)
7  {
8      $var_q = $var_q + 1; // Cantitatea curenta este incrementata cu 1
9      $_SESSION['quantity'][$var_id] = $var_q; // Se actualizeaza cantitatea curenta a produsului cu id-ul specificat
10     header("location:newCart.php");
11 }
12 if (isset($_GET['id'])) {
13     increaseQuantity($_SESSION['products'],$_GET['id'],$_GET["quantity"]);
14 }
15
16 ?>

```

Fig 4 : increaseQuantity.php

```

<?php

    include 'connection.php';

function decreaseQuantity($con,$var_id,$var_q)
{
    echo $var_q;

    if($var_q > 1){
        echo "Intra in blocul de cod cand cantitatea este mai mare decat 1";
        $var_q = $var_q - 1;
        $_SESSION['quantity'][$var_id] = $var_q; // Se actualizeaza cantitatea curenta a produsului cu id-ul specificat
        header("location:newCart.php");
    } else {
        array_splice($_SESSION['products'],$var_id,1); // Se elimina produsul cu id-ul specificat din lista de produse
        array_splice($_SESSION['quantity'],$var_id,1); // Se elimină cantitatea corespunzatoare produsului cu id-ul specificat din lista de produse
        header("location:newCart.php");
    }
}

if (isset($_GET['id'])) {
    decreaseQuantity($_SESSION['products'],$_GET['id'],$_GET["quantity"]);
}

?>

```

Fig 5 : decreaseQuantity.php

```

newCart.php

1  <?php
2      <thead>
3          <tr>
4              <th scope="col">Product Name</th>
5              <th scope="col">Quantity</th>
6              <th scope="col">Price</th>
7              <th scope="col">Delete</th>
8          </tr>
9      </thead>
10     <tbody>;
11
12     $prod = $_SESSION['products'];
13     $quantity = $_SESSION["quantity"];
14
15     // Parcurge fiecare produs din sesiune si afiseaza detalii in tabel (produsul, cantitatea, pretul si optiunea de stergere)
16     for($i = 0; $i < sizeof($_SESSION['products']);$i++){
17         echo "<tr>";
18         $stid =oci_parse($c, "SELECT * FROM PRODUCT WHERE product_id = '$prod[$i]'");
19         oci_execute($stid);
20         $nrows = oci_num_fields($stid);
21
22         if ($nrows > 0) {
23             while(($row = oci_fetch_array($stid, OCI_ASSOC)) != false) {
24                 echo "<th scope='row'>$i.</th>
25                 <td>".row['NAME']. "</td>
26                 <td><a href = 'decreaseQuantity.php?id=".$i."&quantity=". $quantity[$i]."'>".class= 'btn1'."</a>". $quantity[$i]. "<a href = 'increaseQuantity.php?id=".$i."&quantity=".
27                     ">".row['PRICE']."$_SESSION['quantity'][$i]. "</a>"</td>
28                 <td><a href='removeFromCart.php?id=".$i."' class='btn'>Delete</a></td>";
29                 $_SESSION['total'] = $_SESSION['total']+ $row['PRICE']*$quantity[$i];
30             }
31         }
32         echo "</tr>";
33     }
34     echo "</tbody>
35     </table>";
36     echo "<br><br><br>";
37     echo "<span> Total sum : ".$_SESSION['total']."</span>";
38     echo "<br><br>";
39     <form name='form' action='order.php' method='get'>
40         <button type='submit' class='btn btn-danger'>Order</button>
41     </form>

```

Fig 6 : newCart.php

Passation de commande :

- Le processus de passation de commande est géré par la page "order.php" (Fig 7);
- Après avoir saisi les informations de livraison et confirmé la commande, les données sont insérées dans la base de données en utilisant les fonctions de connexion et de requête;
- La page "order.php" propose également un bouton pour finaliser la commande et rediriger l'utilisateur vers une page de confirmation.

```

1  <?php
2      // Session quantity
3      $quant = $_SESSION['quantity'];
4
5      $date = date('Y-m-d H:i:s');
6
7      $orderId = -1;
8
9      $total = $_SESSION['total'];
10
11     $sql = "INSERT INTO MYORDER (total, order_date) VALUES ('$total', to_date('$date', 'YYYY-MM-DD HH24:MI:SS'))";
12     $result=oci_parse($c,$sql);
13     oci_execute($result);
14
15     // Selecteaza ID-ul comenzii recent inserate
16     $new_sql = "SELECT MYORDER_ID FROM myorder WHERE order_date = to_date('$date', 'YYYY-MM-DD HH24:MI:SS')";
17     $new_result=oci_parse($c,$new_sql);
18     oci_execute($new_result);
19
20     // Verifica daca exista campuri in rezultat
21     if (oci_num_fields($new_result) > 0) {
22         while (($row = oci_fetch_array($new_result, OCI_ASSOC)) != false) {
23             $id = $row['MYORDER_ID'];
24             $orderId = $row['MYORDER_ID'];
25
26             // Parurge lista de produse din sesiune si insereaza fiecare produs in tabelul ORDEREDITEM
27             for ($i = 0; $i < sizeof($_SESSION['products']); $i++) {
28                 $new_sql_req = "SELECT * FROM PRODUCT WHERE product_id = '$prods[$i]'";
29                 $new2_result = oci_parse($c,$new_sql_req);
30                 oci_execute($new2_result);
31
32                 // Verifica daca exista campuri in rezultat
33                 if (oci_num_fields($new2_result) > 0) {
34                     // Insereaza inregistrarea in tabelul ORDEREDITEM cu cantitatea, ID-ul produsului si ID-ul comenzii
35                     $insert_sql = "INSERT INTO ORDEREDITEM (quantity,productID,orderID) VALUES ('$quant[$i]', '$prods[$i]', '$id')";
36                     $insert_result=oci_parse($c,$insert_sql);
37                     oci_execute($insert_result);
38                 }
39             }
40         }
41     } else {
42         echo "0 results";
43     }

```

Fig 7 : order.php

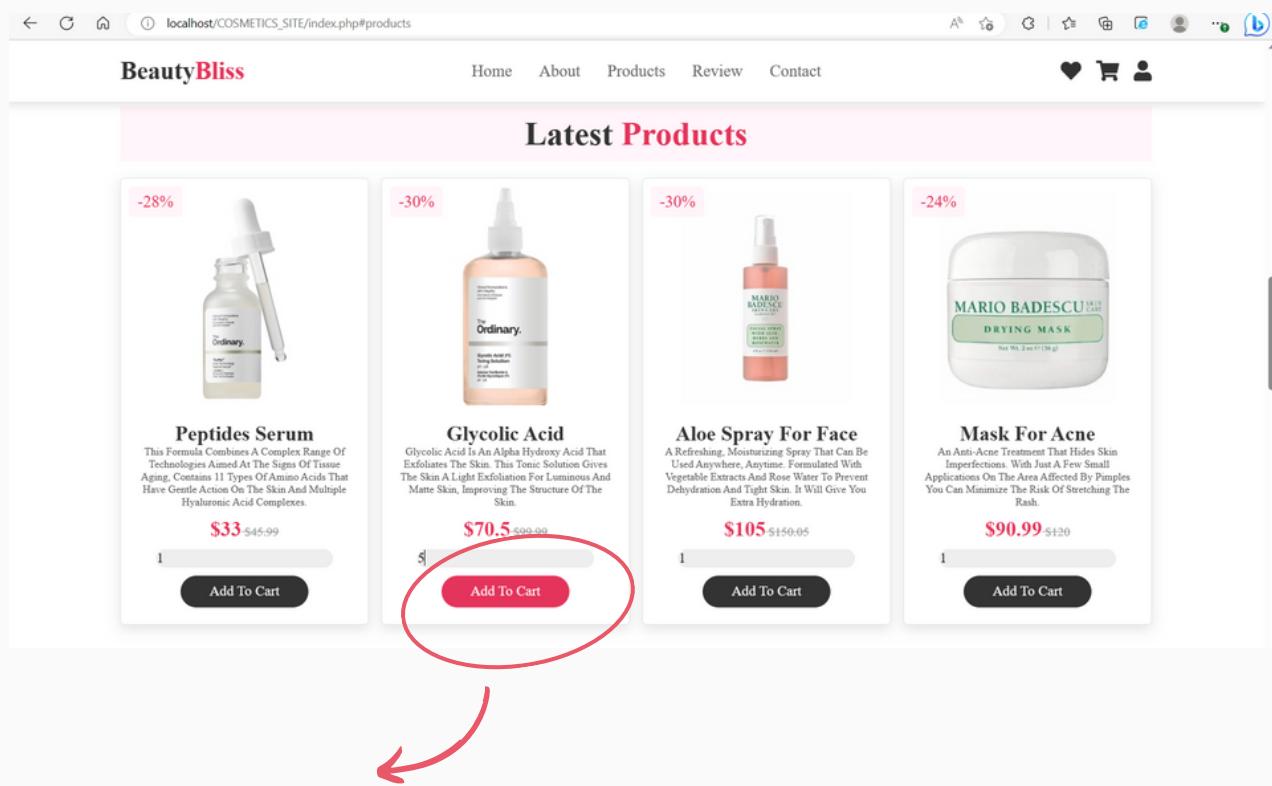
DESIGN ET EXPÉRIENCE UTILISATEUR

"BeautyBliss" offre une interface conviviale et esthétiquement plaisante :

- Le design est adapté à l'industrie des produits cosmétiques, avec des éléments visuels attrayants et une utilisation appropriée des couleurs;
- La navigation est intuitive, permettant aux utilisateurs de trouver facilement les produits, d'ajuster les quantités et de passer une commande;

- Des fonctionnalités interactives, telles que l'ajout et la suppression de produits du panier, améliorent l'expérience utilisateur globale.

Ci-dessous, je vais montrer les étapes à suivre pour passer une commande en tant qu'utilisateur. Ensuite, vous pourrez voir ce que l'administrateur voit dans la base de données après avoir passé une commande.



ÉTAPE 1 :

L'utilisateur choisit le produit dont il a besoin et écrit la quantité souhaitée (par exemple 5), puis il clique sur « Add to Cart ».

localhost/COSMETICS_SITE/php/newCart.php

BeautyBliss

Home About Products Review Contact

MY CART

Id	Product Name	Quantity	Price	Delete
0	Glycolic Acid	- 5 +	352.5	Delete

Total Sum : 352.5

Order

A red arrow points from the 'Order' button to the 'Total Sum' text.

ÉTAPE 2 :

L'utilisateur vérifie son panier où il voit son produit récemment ajouté. Il a la possibilité d'augmenter ou de diminuer la quantité ou même d'abandonner le produit. Cliquez ensuite sur « Order ».

localhost/COSMETICS_SITE/php/orderDetails.php?orderId=132

BeautyBliss

Home About Products Review Contact

Delivery Details

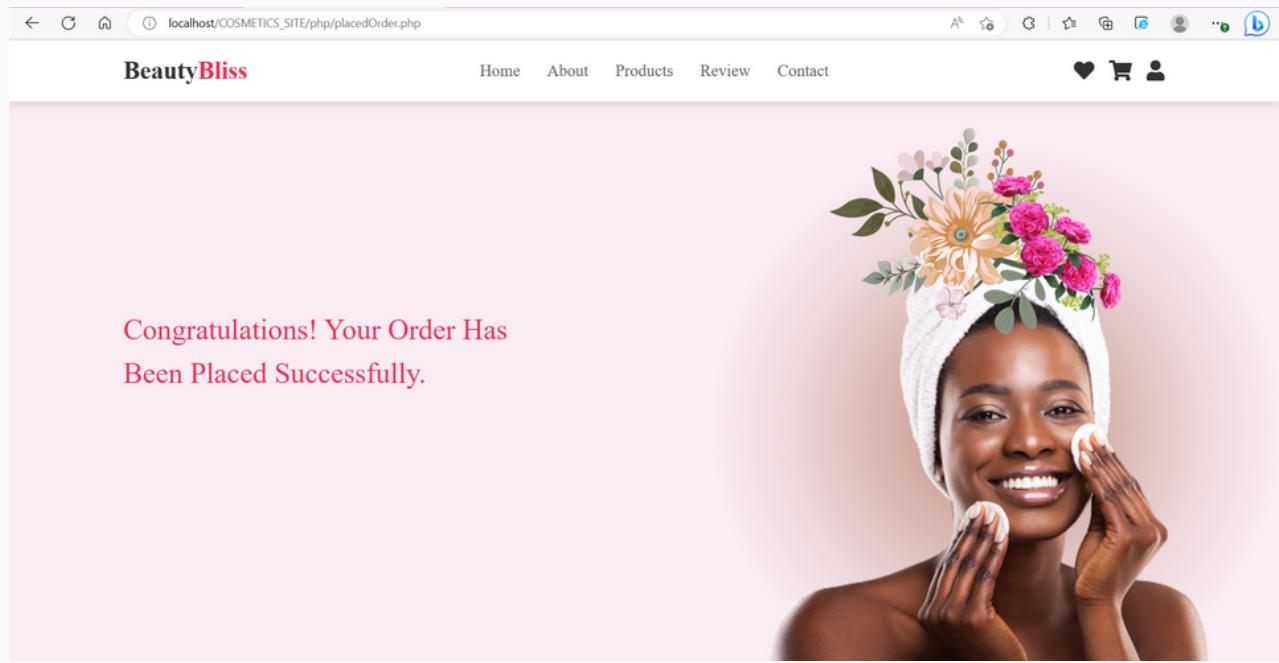
Sonia
Popa
str florilor 15
Pantelimon
sonia2@gmail.com
0722263379

Confirm Details

A red arrow points from the 'Confirm Details' button to the delivery details input fields.

ÉTAPE 3 :

Les champs doivent être remplis avec les détails de livraison personnels. Pour terminer la commande, l'utilisateur clique sur le bouton de confirmation.



ÉTAPE 4 :

L'utilisateur a fini la commande en ligne!

SYSTEM.MYORDER - COSMETICS_site - Visual Studio Code			
tables.sql		Data > SYSTEM.XEPDB1>SYSTEM.PRODUCT	
		Data > SYSTEM.XEPDB1>SYSTEM.MYORDER X	
Save as:	CSV		...
<input checked="" type="checkbox"/>	MYORDER_ID	TOTAL	ORDER_DATE
	131	352.5	22-MAY-23

Fig 8 : MYORDER

Dans la figure 8, l'admin peut voir la commande enregistrée avec le total et la date à laquelle elle a été effectuée.

The screenshot shows a database interface with three tabs at the top: 'tables.sql', 'Data > SYSTEM.XEPDB1>SYSTEM.PRODUCT', and 'Data > SYSTEM.XEPDB1>SYSTEM.ORDEREDITEM'. The current tab is 'Data > SYSTEM.XEPDB1>SYSTEM.ORDEREDITEM'. Below the tabs is a toolbar with icons for saving and exporting to CSV. The main area displays a table with four columns: 'ORDEREDITEM_ID' (checkbox checked, value 83), 'QUANTITY' (value 5), 'PRODUCTID' (value 4), and 'ORDERID' (value 131). The table has a dark header and light body rows.

	ORDEREDITEM_ID	QUANTITY	PRODUCTID	ORDERID
<input checked="" type="checkbox"/>	83	5	4	131

Fig 9 : ORDEREDITEM

La figure 9 représente la base de données des produits commandés en tenant compte de l'ID de commande.

The screenshot shows a database interface with three tabs at the top: 'hp', 'tables.sql', and 'Data > SYSTEM.XEPDB1>SYSTEM.ORDERDETAILS'. The current tab is 'Data > SYSTEM.XEPDB1>SYSTEM.ORDERDETAILS'. Below the tabs is a toolbar with icons for saving and exporting to CSV. The main area displays a table with nine columns: 'ORDERDETAILS_ID' (checkbox checked, value 235), 'FIRSTNAME' (value Sonia), 'LASTNAME' (value Popa), 'ADRESS' (value str florilor 15), 'CITY' (value Pantelimon), 'PHONE' (value 0722263379), 'EMAIL' (value sonia2@gmail.com), and 'ORDER_ID' (value 131). The table has a dark header and light body rows.

	ORDERDETAILS_ID	FIRSTNAME	LASTNAME	ADRESS	CITY	PHONE	EMAIL	ORDER_ID
<input checked="" type="checkbox"/>	235	Sonia	Popa	str florilor 15	Pantelimon	0722263379	sonia2@gmail.com	131

Fig 10 : ORDERDETAILS

Dans ce tableau, l'administrateur enregistre les données de livraison de l'utilisateur.

The screenshot shows a database interface with three tabs at the top: 'delivery.php', 'tables.sql', and 'Data > SYSTEM.XEPDB1>SYSTEM.PRODUCT'. The current tab is 'Data > SYSTEM.XEPDB1>SYSTEM.PRODUCT'. Below the tabs is a toolbar with icons for saving and exporting to CSV. The main area displays a table with four columns: 'PRODUCT_ID', 'NAME', 'PRICE', and 'DESCRIPTION'. A fifth column, 'TOTALQUANTIT', is shown at the top right. The table contains the following data:

PRODUCT_ID	NAME	PRICE	DESCRIPTION	TOTALQUANTIT
3	Peptides serum	33	This formula combines a complex range of technologies aimed at the signs of tissue aging, contains 11 types of amino acids that have gentle action on the skin and multiple hyaluronic acid complexes.	300
4	Glycolic Acid	70.5	Glycolic acid is an alpha hydroxy acid that exfoliates the skin. This tonic solution gives the skin a light exfoliation for luminous and matte skin, improving the structure of the skin.	295
5	Aloe Spray for face	105	A refreshing, moisturizing spray that can be used anywhere, anytime. Formulated with vegetable extracts and rose water to prevent dehydration and tight skin. It will give you extra hydration.	300
7	Mask for Acne	90.99	An anti-acne treatment that hides skin imperfections. With just a few small applications on the area affected by pimples you can minimize the risk of stretching the rash.	300

Fig 11 : PRODUCT

Et ici, on peut observer comment la quantité totale diminue lorsque le produit est commandé.

CONCLUSIONS ET AMÉLIORATIONS FUTURES

Le projet du site de produits cosmétiques, "*BeautyBliss*", intégrant la base de données Oracle, a été réalisé avec succès. Les fonctionnalités clés ont été implémentées et l'expérience utilisateur est satisfaisante.

Cependant, des améliorations peuvent être envisagées pour enrichir davantage le site :

- Mise en place d'un système d'authentification et de gestion des comptes utilisateur pour offrir une expérience personnalisée;
- Ajout d'un moteur de recherche avancé pour faciliter la recherche de produits spécifiques;
- Optimisation des performances du site pour une vitesse de chargement accrue.

