

Date : 15/01/2023

# Rapport - Web

## Appel des procédures stockées & gestion des exceptions

---

Par FERNANDEZ Mickael, DOURLENT Maxime, STRAPUTICARI Luca et  
VIGNAL Alexandre (Groupe G2B-10)



## Sommaire :

<b>Sommaire :</b>	<b>2</b>
Améliorations des pages existantes	3
1. Header (haut de page) - ajustement	3
2. Catégories d'article - optimisation & visuel	3
3. Formulaire de connexion - fonctionnel	4
4. Page index.php - ensemble	6
5. Bonus	7
Ajout de pages fonctionnelles et différentes requêtes	7
1. Inscription d'un client - visuel & fonctionnel	7
2. Modifications des informations d'un client - visuel & fonctionnel	10
3. Page de produit - visuel & fonctionnel	12
4. Page de panier - visuel & fonctionnel	14
5. Ajout d'un produit (artiste) - visuel & fonctionnel	18
6. Mise à jour d'un produit (admin) - visuel & fonctionnel	21
7. Retrait d'un produit (admin) - visuel & fonctionnel	23
Finalisation des pages - gestion des procédures stockées et exceptions	24
1. Page paiement - visuel & fonctionnel	24

## Améliorations des pages existantes

### 1. Header (haut de page) - ajustement

Bien qu'elle soit la page ayant subi le moins de changements, ceux-ci s'avèrent être des changements essentiels. Esthétiquement, tout reste similaire. Pour autant, le sous-en-tête diffère. En effet, après concertation, nous avons jugé qu'une redondance au niveau des pages "posters.php", "figurines.php" ainsi que "peluches.php" pouvait finalement être optimisée car les requêtes ainsi que le corps de ces pages étaient strictement similaires. Seul un paramètre changeait et qui n'était d'autre que la catégorie en question. En optimisant alors ce procédé, ces pages ont été supprimées pour laisser place à une page nommée "categorie.php", traitant alors de ces liens hypertextes. Ainsi, dans l'URL, une variable 'catégorie' est créée afin d'en attribuer le nom de la catégorie selon celle choisie.

**Code :**

```
<div id="subheader">
  <a href="categorie.php?categorie=POSTERS" class="h-items">Posters</a>
  <a href="categorie.php?categorie=FIGURINES" class="h-items">Figurines</a>
  <a href="categorie.php?categorie=PELUCHES" class="h-items">Peluches</a>
</div>
```

### 2. Catégories d'article - optimisation & visuel

Les catégories d'article ont ainsi subi des changements importants, ne laissant alors place qu'à un fichier gérant l'entièreté de ces catégories (auparavant, une page était dédiée à une catégorie spécifique). Le procédé reste dans sa majorité similaire : même requête et même esthétique. Pour autant, pour véritablement faire différer ces pages, ce sera la valeur attribuée au paramètre 'catégorie', passé dans l'URL qui fera toute la différence. Le paramètre nommé de la requête SQL sera alors lié à cette valeur passée dans l'URL, chose qui simplifie et optimise grandement le code ainsi que le nombre de fichiers associés.

**Code :**

```

<?php include("./include/header.php"); ?>

<?php

$recherche = $_GET['categorie'];
$requete = "SELECT * FROM ARTICLE WHERE NOMCATEGORIE LIKE '%' || :categorie || '%'";
$req1 = oci_parse($connect, $requete);
oci_bind_by_name($req1, ':categorie', $recherche);
$result= oci_execute($req1);

echo("<div class='squarE'>");
echo("<h1 class='titleArticles'>Voici les figurines</h1>");
echo("<div id='gridArticles'>");

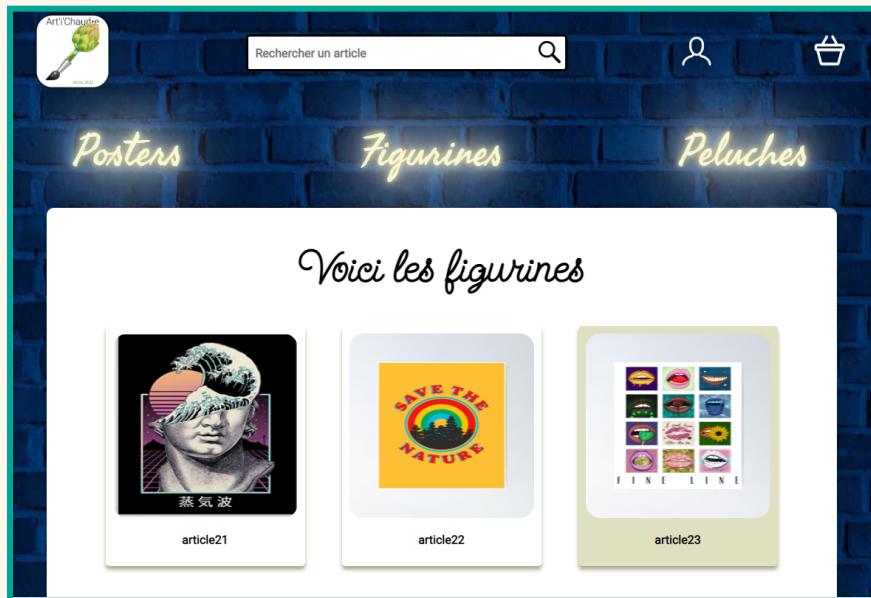
while (($article = oci_fetch_assoc($req1))!= false) {
    echo('<div class="containerArticle" style="cursor: pointer;" onclick="location.href=\\'produit.php?id=' . $article['IDARTICLE']. '\'\">' );
    echo("<img class='imageArticle' src='./include/images/imagesArticles/".$article['NOMARTICLE'].".png'>");
    echo("<br/>");
    echo("<p class='articleText'>".$article['NOMARTICLE'].".</p>");

    echo("</div>");
}
echo("</div>");
echo("</div>");

?>

```

*Site web (exemple de page) :*



### 3. Formulaire de connexion - fonctionnel

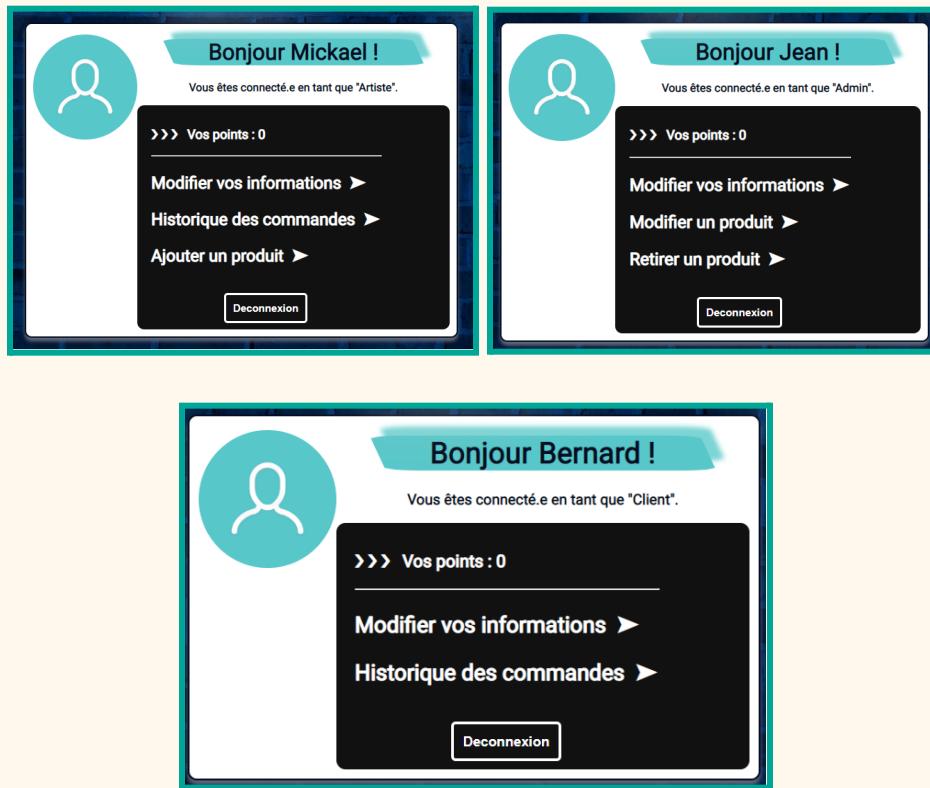
Comme demandé par nos clients de la société Art'i'Chaudé, certains utilisateurs du site possèderont des rôles différents avec des possibilités d'interaction également différentes. Pour un artiste par exemple, celui-ci pourra, en plus de modifier ses informations ou de modifier son mot de passe, ajouter un produit. Pour un administrateur, celui-ci possède des permissions

générales comme le fait de modifier ou de supprimer un produit, dans deux formulaires distincts.

Évidemment, l'ensemble des cas d'erreurs sont traités, notamment si un utilisateur malicieux décide d'accéder à l'un de ces fichiers directement depuis l'URL.

En voici les différentes interfaces et le code associé à la page "connexion.php" :

### *Interfaces :*



### *Code modifié :*

```

echo('
<h3>&#10095; &#10095; &#10095; &#8205; &#8205; Vos points : <strong>'.$_SESSION['ptsfidelite'].'
</strong> </h3>');
echo('<hr>');
echo('<a class="liensUtilisateur" href="modification.php"><h2>Modifier vos informations &#8205; &#x27A4;</h2></a>');
if($_SESSION['utilisateur'] == 'Client') {
    echo('<a class="liensUtilisateur" href="historique.php"><h2>Historique des commandes &#8205; &#x27A4;</h2></a>');
}
if($_SESSION['utilisateur'] == 'Artiste') {
    echo('<a class="liensUtilisateur" href="historique.php"><h2>Historique des commandes &#8205; &#x27A4;</h2></a>');
    echo('<a class="liensUtilisateur" href="addProduit.php"><h2>Ajouter un produit &#8205; &#x27A4;</h2></a>');
}
if($_SESSION['utilisateur'] == 'Admin') {
    echo('<a class="liensUtilisateur" href="updateProduit.php"><h2>Modifier un produit &#8205; &#x27A4;</h2></a>');
    echo('<a class="liensUtilisateur" href="deleteProduit.php"><h2>Retirer un produit &#8205; &#x27A4;</h2></a>');
}

```

Par ailleurs, pour pouvoir modifier l'ensemble des informations d'un utilisateur, il est nécessaire de les récupérer, au préalable, lors de sa connexion. C'est pourquoi le fichier "traitConnexion.php" a également été modifié pour faire que dans des variables de sessions disjointes, l'on puisse récupérer l'ensemble de ses informations :

### **Code modifié :**

```

if(password_verify($mdpActeur, $clientDatabase['MDPACTEUR'])) {
    $_SESSION['id'] = $clientDatabase['IDACTEUR']; //Auto
    $_SESSION['nom'] = $clientDatabase['NOMACTEUR'];
    $_SESSION['prenom'] = $clientDatabase['PRENOMACTEUR'];
    $_SESSION['utilisateur'] = $clientDatabase['ROLEACTEUR']; //Auto
    $_SESSION['adresse'] = $clientDatabase['ADRACEUR'];
    $_SESSION['mail'] = $clientDatabase['MAILACTEUR'];
    $_SESSION['mdp'] = $clientDatabase['MDPACTEUR'];
    $_SESSION['langue'] = $clientDatabase['PREFLANGUE'];
    $_SESSION['theme'] = $clientDatabase['PREFTHEME'];
    $_SESSION['ptsfidelite'] = $clientDatabase['PTSFIDELITECLIENT']; //Auto
    $_SESSION['monnaie'] = $clientDatabase['PREFMONNAIE'];

    if(isset($_POST['memo'])) {
        setcookie('cookIdentifiant', $_POST['adresseMail'], time() + 60);
    } else {
        setcookie('cookIdentifiant', false, -1);
    }

    // Libère toutes les ressources réservées par un résultat Oracle
    oci_free_statement($clientRecup);

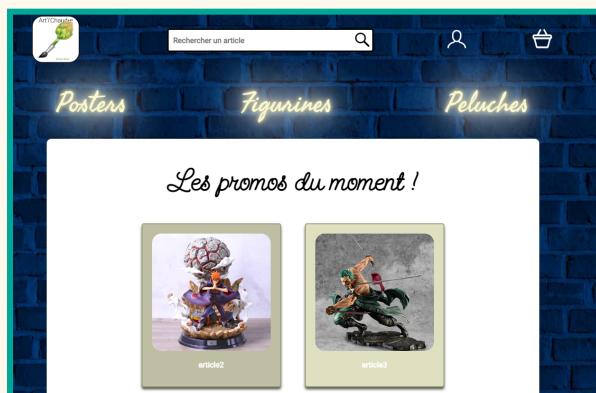
    header('Location: ./index.php');
    exit;
} else {
    header('Location: ./connexion.php?msgErreur=Mot de passe invalide');
    exit;
}

```

## **4. Page index.php - ensemble**

Auparavant, seules les images des produits figuraient sur ces pages. Désormais, ces pages sont plus explicites avec l'ajout du nom de l'article. Par ailleurs, seule l'image était cliquable afin d'accéder à la page "produit.php" ce qui contraignait l'utilisateur dans sa facilité d'interaction avec le site web. Ainsi, ce sera désormais le conteneur qui est sélectionnable, ce qui est nettement plus intuitif et pratique d'utilisation :

### **Site actuel :**



## Code de la page index :

```
<?php
$requete = "SELECT NOMARTICLE, IDARTICLE FROM ARTICLE where IDARTICLE=15 OR IDARTICLE=2 OR IDARTICLE=12 OR IDARTICLE=3";
$req1 = oci_parse($connect, $requete);

$result= oci_execute($req1);

echo("<div id='indexContainer'>");

    echo("<h2 class='titleArticles'>Les promos du moment !</h2>");
    echo("<div id='gridIndex'>");

    while ((($produit = oci_fetch_assoc($req1))!= false) {

        echo('<div class="containerIndex" style="cursor: pointer;" onclick="location.href=\\'produit.php?id='.$produit['IDARTICLE'].'\\\'>');
        echo("<img class='imageArticle' src='include/images/imagesArticles/".$produit['NOMARTICLE'].".png'>");
        echo("<br/>");
        echo('<p class="articleText">'.$produit['NOMARTICLE'].'</p>');

        echo("</div>");
    }

    echo("</div>");
echo("</div>");

?>
```

Ici-même, les articles ont été tirés au sort au sein de notre équipe pour déterminer lesquels seraient en promotion.

## 5. Bonus

Pour faciliter la distinction des onglets d'Art'i'Chaud ou pour donner une meilleure visibilité et esthétique à celui-ci, son logo figure alors dans l'onglet sélectionné :



## Ajout de pages fonctionnelles et différentes requêtes

### 1. Incription d'un client - visuel & fonctionnel

Comme la plupart des pages ou le renseignement d'informations est nécessaire, celui-ci se présente sous la forme d'un formulaire. Ce formulaire est alors constitué de l'ensemble des informations nécessaires qui doivent figurer dans la table 'ACTEUR' de notre base de données, à l'exception de son ID qui lui, sera automatiquement incrémenté.

Le procédé général alliant BD et PHP est relativement similaire entre chaque page, bien que les requêtes ou les informations diffèrent. Concrètement, on vérifie que l'utilisateur ne soit pas connecté, puis, que l'ensemble des informations sont à la fois saisies et valables. De là, le mot

de passe renseigné dans son champ de texte sera hashé afin d'éviter une potentielle attaque en brute forte de la part d'un utilisateur malveillant. Par ailleurs, l'utilisateur sera obligé de renseigner deux fois son mot de passe pour confirmer cette saisie.

Pour autant, la condition essentielle lors d'une inscription est de vérifier si le compte n'existe pas déjà. Sachant qu'une adresse mail est unique par compte, nous allons vérifier si celle renseignée ne figure pas dans notre base de données car si c'est le cas, il sera automatiquement redirigé sur sa page de profil. Autrement, l'ensemble de ses informations seront insérées avec vérification et l'utilisateur sera redirigé vers la page de connexion pour pouvoir tester la connexion de son nouveau compte.

#### ***Maquette associée :***

**S'inscrire :**

*Choisissez votre langue :*

*Entrez votre prénom :*

*Entrez votre nom :*

*Entrez votre adresse :*

*Entrez votre mail :*

*Entrez votre mot de passe :*

*Confirmez votre mot de passe :*

***Code général (`inscription.php` puis `traitInscription.php`) :***

```

echo('<form method="post" id="formInscription" action="traitInscription.php">');
echo('<div id="titleFrame"><h1 id="formTitle">S\'inscrire :</h1></div>');
if(isset($_GET['msgErreur'])){
    echo("<h3 class='errorDatabase'>".$_GET['msgErreur']. "</h3>");
}
echo('<label class="formLabels">Choisissez votre langue :</label>');
echo("<select name='langues' id='selectLangues'>");
    echo("<option class='optionLangues' value='FR'>Français</option>");
    echo("<option class='optionLangues' value='EN'>English</option>");
echo("</select>");
echo('<br/><br/>');
echo('<label class="formLabels">Entrez votre prénom :</label>');
echo("<input type='text' class='inscChamps' name='prenom' required/>");
echo('<br/><br/>');
echo('<label class="formLabels">Entrez votre nom :</label>');
echo("<input type='text' class='inscChamps' name='nom' required/>");
echo('<br/><br/>');
echo('<label class="formLabels">Entrez votre adresse :</label>');
echo("<input type='text' class='inscChamps' name='adresse' required/>");
echo('<br/><br/>');
echo('<label class="formLabels">Entrez votre mail :</label>');
echo("<input type='email' class='inscChamps' name='adresseMail' required/>");
echo('<br/><br/>');
echo('<label class="formLabels">Entrez votre mot de passe :</label>');
echo("<input type='password' class='inscChamps' name='motdepasse' required/>");
echo('<br/><br/>');
echo('<label id="confirmLabel">Confirmez votre mot de passe :</label>');
echo("<input type='password' id='confirmChamp' name='confmdp' required/>");
echo('<br/><br/>');
echo('<input type="submit" name="Inscription" value="Inscription" id="btnInscription"/>');
echo('</form>');

```

```

<?php
// on inclut le fichier de connexion à la base Oracle
require_once('connect.inc.php');
error_reporting(0);

session_start();

if(!isset($_POST['Inscription'])) {
    header('Location: ./inscription.php?msgErreur=Veuillez remplir les champs');
    exit;
}

if (empty($_POST['langues'])){
    header('Location: ./inscription.php?msgErreur=Sélection incorrecte');
    exit;
}

if (! (isset($_POST['prenom']) && isset($_POST['nom']) && isset($_POST['adresse']) && isset($_POST['adresseMail']) && isset($_POST['motdepasse']) && isset($_POST['confmdp'])) {
    header('Location: ./inscription.php?msgErreur=Veuillez remplir les champs');
    exit;
}

$prenomActeur = htmlspecialchars($_POST['prenom']);
$nomActeur = htmlspecialchars($_POST['nom']);
$adresseActeur = htmlspecialchars($_POST['adresse']);
$mailActeur = htmlspecialchars($_POST['adresseMail']);
$mdpActeur = htmlspecialchars($_POST['motdepasse']);
$confmdpActeur = htmlspecialchars($_POST['confmdp']);

$hashedMDP = password_hash($mdpActeur, PASSWORD_DEFAULT);

if($mdpActeur != $confmdpActeur){
    header('Location: ./inscription.php?msgErreur=Mots de passe différents');
    exit;
}

if (!preg_match('#^([a-z0-9_-]+@[a-z0-9_-]{2,}\.[a-z]{2,4}$#, $mailActeur)) {
    header('Location: ./inscription.php?msgErreur=Vérifiez la saisie de votre adresse mail');
    exit;
}

$req = "SELECT * FROM ACTEUR WHERE MAILACTEUR = :pMailActeur";
$acteurRecup = oci_parse($connect, $req);
oci_bind_by_name($acteurRecup, ":pMailActeur", $mailActeur);
$result = oci_execute($acteurRecup);

```

```

if (!$result) {
    $e = oci_error($acteurRecup); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
    $SQLerror = htmlentities($e['message'] . ' pour cette requête : ' . $e['sqltext']);

    header('Location: ./inscription.php?msgErreur=' . $SQLerror);
    exit;
}

if (oci_fetch_assoc($acteurRecup)) {
    header('Location: ./inscription.php?msgErreur=Compte déjà existant');
    exit;
}

oci_free_statement($acteurRecup);

$req = "INSERT INTO ACTEUR (IDACTEUR,NOMACTEUR,PRENOMACTEUR,ADRACKETEUR,MAILACTEUR,MDPACTEUR,PREFLANGUE,PREFTHEME,PTSFIDELITECLIENT,PREFMONNAIE,ROLEACTEUR)
        | VALUES SEQ_ACTEUR_NEXVAL,:pNom,:pPrenom,:pMail,:pMDP,:pLangue,'CLAIR',0,'EUR','Client')";
$acteurInsert = oci_parse($connect, $req);
oci_bind_by_name($acteurInsert, ":pNom", $nomActeur);
oci_bind_by_name($acteurInsert, ":pPrenom", $prenomActeur);
oci_bind_by_name($acteurInsert, ":pAdresse", $adresseActeur);
oci_bind_by_name($acteurInsert, ":pMail", $mailActeur);
oci_bind_by_name($acteurInsert, ":pMDP", $hashedMDP);
oci_bind_by_name($acteurInsert, ":pLangue", $_POST['langues']);

$result = oci_execute($acteurInsert);

if (!$result) {
    $e = oci_error($acteurInsert); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
    $SQLerror = htmlentities($e['message'] . ' pour cette requête : ' . $e['sqltext']);

    header('Location: ./inscription.php?msgErreur=' . $SQLerror);
    exit;
}

oci_commit($connect);

oci_free_statement($acteurInsert);

echo('<script language="JavaScript" type="text/javascript">
    alert("Inscription réussie.");
    location.href = "./connexion.php";
</script>');

```

## 2. Modifications des informations d'un client - visuel & fonctionnel

La modification des informations d'un client suit de manière très ressemblante, dans un aspect visuel, le formulaire d'inscription. Ici, il s'agit de renseigner l'ensemble des informations à modifier ou non pour un utilisateur. S'il ne souhaite pas modifier certaines informations, il n'a pas à interagir avec le champ de texte concerné étant donné que ceux-ci sont remplis automatiquement par les informations de l'utilisateur. Toutefois, le mot de passe étant hashé, l'ancien mot de passe n'est alors pas concis aux yeux de l'utilisateur, bien qu'il reste figuré dans le formulaire. Évidemment, si l'utilisateur n'est pas connecté, il n'aura pas cette possibilité.

Son fonctionnement est strictement similaire à celui de l'inscription, à la différence de la requête SQL où il s'agira de mettre à jour des informations, et de récupérer l'ID de l'utilisateur concerné pour mettre à jour les informations du bon utilisateur.

**Maquette associée et code général (*modification.php* puis *traitModification.php*) :**

Désormais, seules les informations non redondantes seront figurées.

## Modifier ses infos

Veuillez compléter ces informations

**Choisissez votre langue :**

**Entrez votre prénom :**

**Entrez votre nom :**

**Entrez votre adresse :**

**Entrez votre mail :**

**Choisissez votre thème :**

**Choisissez votre monnaie :**

---

### Changement de mot de passe

**Votre ancien mot de passe :**

**Entrez votre nouveau mot de passe :**

**Confirmez votre mot de passe :**

```

echo('
<form method="post" id="formModification" action="traitModification.php">
    <div id="titleName"><h1>Modifier ses infos</h1></div>
    <div id="msgError">' . ($isSet[$_GET['msgError']] ? '<h3>' . $isGet['msgError'] . '</h3>' : '') . '</div>
    <div class="errorDatabase">Veuillez compléter ces informations</div>
</form>
');

```

```

echo('
<div id="littlaTitleName"><h3>Changement de mot de passe</h3></div>
<div id="blockFormLabel">Votre ancien mot de passe </label>
<input type="password" id="oldmdp" name="oldmdp" value="' . $_SESSION['mdp'] . '" disabled/>
<br/>
<div id="confirmLabel">Confirmez votre mot de passe </label>
<input type="password" id="confirmp" name="confirmp" required/>
<br/>
<input type="submit" name="Sauvegarder" value="Sauvegarder" id="btnModification"/>
');

```

```

$Id = $_SESSION['id'];
//On crée une requête paramétrée
$req = "UPDATE ACTEUR
        SET NOMACTEUR = :pNom,
            PRENOMACTEUR = :pPrenom,
            ADRACTEUR = :pAdresse,
            MAILACTEUR = :pMail,
            MOPACTEUR = :pMdp,
            PREFLANGUE = :pLangue,
            PREFTHEME = :pTheme,
            PREFMONNAIE = :pMonnaie
        WHERE IDACTEUR=$Id";
//On prépare la requête
$acteurRecup = oci_parse($connect, $req);
//On lie les valeurs aux paramètres de la requête
oci_bind_by_name($acteurRecup, ':pNom', $nomActeur);
oci_bind_by_name($acteurRecup, ':pPrenom', $prenomActeur);
oci_bind_by_name($acteurRecup, ':pAdresse', $adresseActeur);
oci_bind_by_name($acteurRecup, ':pMail', $mailActeur);
oci_bind_by_name($acteurRecup, ':pMdp', $shashedMdp);
oci_bind_by_name($acteurRecup, ':pLangue', $_POST['langues']);
oci_bind_by_name($acteurRecup, ':pTheme', $_POST['theme']);
oci_bind_by_name($acteurRecup, ':pMonnaie', $_POST['monnaie']);

$result = oci_execute($acteurRecup);

if (!$result) {
    $e = oci_error($acteurRecup); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
    $SQLerror = htmlentities($e['message'] . ' pour cette requête : ' . $e['sqltext']);
    header('Location: ./inscription.php?msgErreur=' . $SQLerror);
    exit;
}

oci_commit($connect);
oci_free_statement($acteurRecup);
session_destroy();

echo('<script language="JavaScript" type="text/javascript">
    alert("Modification réussie. Veuillez-vous reconnecter.");
    location.href = "./connexion.php";
</script>');

```

### 3. Page de produit - visuel & fonctionnel

La page d'un produit est intéressante à analyser puisqu'elle contient un ensemble d'informations diverses et variées, ne nécessitant pas nécessairement l'utilisation d'une seule table car un même produit peut posséder divers coloris ou que les avis des utilisateurs sont figurés dans une table externe à la table 'ARTICLE'.

L'arrangement de cette page s'est réalisé en CSS par l'utilisation des grilles (ou grid). L'avantage est qu'il facilite l'arrangement des objets à faire figurer autant horizontalement que verticalement. Certaines cellules peuvent empiéter sur d'autres (elles seront allongées) et sont facilement personnalisables, au lieu d'utiliser une multitude de conteneur `div`.

Par ailleurs, l'arrangement de l'affichage des artichauts selon la note s'est procédé de la manière suivante : la variable à décrémenter commence de la valeur maximale possible pour un avis : 5. Si l'avis du produit est inférieur à cette variable, ce sera un artichaut noir qui s'affichera, autrement un artichaut classique. Enfin, les couleurs sont affichées selon les couleurs existantes et définies pour l'article dans la base de données. Le tout est traité dans un formulaire (qui n'a plus l'air d'un formulaire) afin de récupérer les informations essentielles du produit, essentielles également pour son affichage (description, prix, etc.).

**Site web (exemple de produit) :**



Le nom de l'article, les prix, les avis ainsi que les descriptions seront changés ultérieurement.

*Code (styles.css et produit.php) :*

```

session_start();
$recherche = $_GET['id'];

//Recup la note moyenne, nom, description, catégorie et prix d'un produit
$reqte1 = "SELECT AVG(AVIS.NOTE) AS NOTE, NOMARTICLE, DESCRIPTION, PRIXARTICLE FROM AVIS,ARTICLE
          WHERE AVIS.IDARTICLE = ARTICLE.IDARTICLE AND ARTICLE.IDARTICLE = :idArticle GROUP BY NOMARTICLE, DESCRIPTION, PRIXARTICLE";
$req1 = oci_parse($connect, $reqte1);
oci_bind_by_name($req1,':idArticle',$recherche);//article correspondant
$result1 = oci_execute($req1);

//Recup le nom et prénom et l'avis de la personne sur un produit
$reqte2 = "SELECT ACTEUR.NOMACTEUR,ACTEUR.PRENOMACTEUR,AVIS.NOTE,AVIS.AVIS FROM ACTEUR,AVIS
          WHERE ACTEUR.IDACTEUR = AVIS.IDACTEUR AND AVIS.IDARTICLE =:idArticle";
$req2 = oci_parse($connect, $reqte2);
oci_bind_by_name($req2,':idArticle', $recherche);
$result2 = oci_execute($req2);

//Recup les couleurs du produit
$reqte4 = "SELECT COULEUR FROM COULEUR,ARTICLE WHERE ARTICLE.IDARTICLE = COULEUR.IDARTICLE AND ARTICLE.IDARTICLE =:idArticle";
$req4 = oci_parse($connect, $reqte4);
oci_bind_by_name($req4,':idArticle', $recherche);
$result4 = oci_execute($req4);

$nom = oci_fetch_assoc($req1);
$product_nom = $nom['NOMARTICLE'];
$pri = $nom['PRIXARTICLE'];
$des = $nom['DESCRIPTION'];
$note = $nom['NOTE'];

echo("<div class='square'>");
echo(' <h1>' . $product_nom . '</h1>');

echo('<form method="post" action="panier.php">');
echo(' <input type="hidden" name="product_id" value="' . $recherche . '">');

echo('<div id="gridProduit">');

echo('<div class="imageCell">');
echo(" <img id='imageProduit' src='include/images/imagesArticles/" . $product_nom . ".png'>");
echo('</div>');

echo('<div class="main-infos">');
echo(' <p>Note globale:</p>');

for($i = 5; $i > 0; $i--){
    if ($note < $i){
        echo('');
    } else {
        echo('');
    }
}
echo('</p>');

echo('</div>');
echo('<div class="main-infos">');

echo('<label for="quantite">Quantité</label>');
echo('<input type="number" id="quantity" name="quantite" min="1" max="5" value="1">');

echo('</div>');

echo('<div class="descContainer">');
echo(' <p>Description</p>');

echo('<div id="descProduit">');
echo(' <p id="descChamp">' . $des . '</p>');
echo('</div>');

echo('</div>');

```

```

echo("<div class='couleurs'>");
echo(' <p>Coloris : </p>');

echo("<select id='selectCouleurs' class='couleur' name='couleur'>");
$couleurs = oci_fetch_assoc($req4);
while($couleurs){
    echo('<option value="'.$couleurs['COULEUR'].'">'.$couleurs['COULEUR'].'</option>');
    $couleurs = oci_fetch_assoc($req4);
}
echo("</select>");

echo("</div>");

echo("<div id='prixProduit'>");
echo(' <p>Prix : '.$pri['euro'] . '€</p>'); //$_SESSION['monnaie'];
echo("</div>");

echo("<div class='payerProduit'>");
echo(' <button name="ajout" id="btnPanier">');
echo(' ');
echo(' </button>');
echo("</div>");

echo("<div class='singleCellProduit'>");
echo(' <p>Avis :</p>');
echo("</div>");

echo("<div id='avisProduit'>");

while (($avis = oci_fetch_assoc($req2))!=false) {
    echo("<div class='containerAvis'>");

    echo('<p class="avisPersonne">'.$avis['NOMACTEUR'].' ' . $avis['PRENOMACTEUR']);

    for($i = 5; $i > 0; $i--){
        if ($avis['NOTE'] == $i){
            echo('');
        } else {
            echo('');
        }
    }

    echo("</p>");
    echo("<br/>");
    echo('<hr>');
    echo('<p class="avisPersonne">'.$avis['AVIS']. '</p>');

    echo("</div>");

    echo('<hr id="separateurAvis"/>');
}

echo("</div>");
echo('</div>');
echo("</form>");

echo("</div>");
```

```

#gridProduit {
    display: grid;
    grid-template-columns: auto auto auto auto; /*columns widths*/
}

.imageCell{
    grid-column: 1 / span 2;
    grid-row: 1 / span 3;
    padding:20px;
    margin-right:20px;
}

#imageProduit {
    width:275px;
    height:275px;
    border-radius:30px;
}

.main-infos {
    grid-column: 3 / span 2;
    display: inline-block;
    vertical-align: middle;
}

.quantite {
    height: 30px;
    width: 70px;
    font-size:20px;
    margin-left: 100px;
    border:2px solid black;
}

.descContainer {
    grid-column: 3 / span 2;
}
```

## 4. Page de panier - visuel & fonctionnel

Comme observé depuis la page d'un produit, un formulaire ayant pour action "panier.php" était à déclarer. Effectivement, celui-ci viendra récupérer les informations figurées depuis la page du produit à même dans le panier. Toutefois, si l'utilisateur n'est pas connecté, un lien hypertexte lui proposant de se connecter sera affiché.

Ainsi, si le même article vient à être ajouté plusieurs fois indépendamment de sa gestion de quantité depuis le panier, celui-ci sera mis à jour afin d'éviter les doublons d'un même article.

Si le panier est vide, un message lui sera affiché pour lui en informer de l'état, autrement, ce sera l'ensemble des caractéristiques de l'article ajouté qui sera affiché. L'utilisateur a également la possibilité de modifier la quantité de l'article en question. Si celui-ci décide de retirer l'article, il lui suffit de définir sa quantité à zéro pour établir la suppression. Autrement, si la

quantité est modifiée, un simple “UPDATE” dédiée à ce changement sera effectué. Ainsi, dès lors de l’ajout d’un produit dans le panier, ce procédé s’effectue également dans la table associée au panier avec pour seule précision, la quantité de celui-ci.

Enfin, si l’utilisateur décide de se déconnecter et de se reconnecter, comme le panier est associé au client, celui-ci pourra retrouver son panier à n’importe quel moment, du moment que celui-ci ne prend pas de décisions à son égard.

L’arrangement du panier suit également un arrangement similaire à la page produit avec un arrangement avec grille.

**Maquette réalisée :**



**Code (*panier.php* et *traitModifPanier.php*) :**

```

session_start();

echo("<div class='squareE'>");
echo("<div id='titleFrame'><h1 id='formTitle'>Panier :</h1></div>");

if(!isset($_SESSION['id'])){

    echo('<a id="linkToConnexion" href="./connexion.php"><p>Veuillez vous connecter pour accéder à votre panier.</p></a>');
}

else {

    if (isset($_POST['ajout'])){

        $product_id = htmlspecialchars($_POST['product_id']);
        $quantite = htmlspecialchars($_POST['quantite']);
        $idacteur = htmlspecialchars($_SESSION['id']);
        $couleur = htmlspecialchars($_POST['couleur']);

        $reqPanier = "SELECT * FROM PANIER WHERE idArticle = :product_id AND idActeur = :idacteur AND couleur = :couleur";
        $panierRecup = oci_parse($connect, $reqPanier);
        oci_bind_by_name($panierRecup,':product_id',$product_id);
        oci_bind_by_name($panierRecup,':idacteur',$idacteur);
        oci_bind_by_name($panierRecup,':couleur',$couleur);
        $resultPanier = oci_execute($panierRecup);

        if (oci_fetch_assoc($panierRecup)) {
            $requeteUpdate = "UPDATE PANIER SET quantiteArticle = quantiteArticle + :quantite WHERE idArticle = :product_id AND idActeur = :idacteur AND couleur = :couleur";
            $reqUpdate = oci_parse($connect,$requeteUpdate);

            oci_bind_by_name($reqUpdate,':product_id',$product_id);
            oci_bind_by_name($reqUpdate,':quantite',$quantite);
            oci_bind_by_name($reqUpdate,':idacteur',$idacteur);
            oci_bind_by_name($reqUpdate,':couleur',$couleur);

            $resultUpdate = oci_execute($reqUpdate);

            if (!$resultUpdate) {
                $e = oci_error($reqUpdate); // on récupère l'exception liée au pb d'exécution de la requête (violation PK par exemple)
                $SQLerror = htmlentities($e['message']) . ' pour cette requête : ' . $e['sqltext'];

                header('Location: ./index.php?msgErreur=' . $SQLerror);
                exit;
            }
        } else{
            $requeteInsert = "INSERT INTO PANIER (idActeur, idArticle, QuantiteArticle, Couleur) VALUES (:idacteur, :product_id, :quantite, :couleur)";
            $reqInsert = oci_parse($connect,$requeteInsert);

            oci_bind_by_name($reqInsert,':product_id',$product_id);
            oci_bind_by_name($reqInsert,':quantite',$quantite);
            oci_bind_by_name($reqInsert,':idacteur',$idacteur);
            oci_bind_by_name($reqInsert,':couleur',$couleur);

            $result = oci_execute($reqInsert);

            if (!$result) {
                $e = oci_error($reqInsert); // on récupère l'exception liée au pb d'exécution de la requête (violation PK par exemple)
                $SQLerror = htmlentities($e['message']) . ' pour cette requête : ' . $e['sqltext'];

                header('Location: ./index.php?msgErreur=' . $SQLerror);
                exit;
            }
        }
    }
}

```

```
$requete1 = "SELECT PANIER.IDARTICLE, NOMARTICLE, PRIXARTICLE, PRIXPROMO, COULEUR, QUANTITEARTICLE FROM ARTICLE,PANIER
WHERE ARTICLE.IDARTICLE=PANIER.IDARTICLE AND PANIER.IDACTEUR=:idacteur";
$req1 = oci_parse($connect,$requete1);
oci_bind_by_name($req1,':idacteur',$_SESSION['id']);
oci_execute($req1);

$totalPanier = 0;
$totalArticle = 0;
$compteur = 0;

$panier = oci_fetch_assoc($req1);

if (!$panier){
    echo('<div id="titleFrame"><h1 id="formTitle">Votre panier est vide</h1></div>');
} else {
    do {
        $compteur++;
        $product_id = $panier['IDARTICLE'];
        $product_nom = $panier['NOMARTICLE'];
        $product_quantite = $panier['QUANTITEARTICLE'];
        $product_prix = $panier['PRIXARTICLE'];
        $product_couleur = $panier['COULEUR'];
        $totalArticle = $product_prix * $product_quantite;

        echo('<div id="gridPanier">');

        echo('<div class="headPanier">');
        echo("<p class='headTextePanier'>Nom produit : ".$product_nom."</p>");
        echo('</div>');

        echo('<div class="caracPanier">');
        echo("<img id='imgPanier' src='include/images/imagesArticles/".$product_nom.".png'>");
        echo('</div>');

        echo('<form id="formQuantite" method="post" action="traitModifPanier.php">');
        //Faire un formulaire pour prendre en compte les modifs

        echo('<input type="hidden" name="productID" value="'.$product_id.'">');
        echo('<input type="hidden" name="productCouleur" value="'.$product_couleur.'">');

        echo('<div class="caracQuantitePanier">');
        echo("<p>Quantité : </p>");
        echo('<input type="number" id="quantity" name="quantity" min="0" max="999" value="'.$product_quantite.'">');
        echo('<input type="submit" name="Modifier" id="modifPanier" value="Modifier">');
        echo('</div>');

        echo('</form>');

        echo('<div class="caracPanier">');
        //Chercher le coloris depuis produit
        echo("<p>Coloris : ".$product_couleur."</p>");
        echo('</div>');

        echo('<div class="caracPanier">');
        echo("<p>Prix unitaire : ".$product_prix.".00€</p>");
        echo('</div>');

        echo('<div class="headPanier">');
        echo("<p class='headTextePanier'>Prix total : ".$totalArticle.".00€</p>");
        echo('</div>');

        $totalPanier += $totalArticle;

        $panier = oci_fetch_assoc($req1);

        echo('</div>');

        echo('<hr id="delimiteurPanier">');
    }
}
```

```
    } while ($panier);

    echo("<p>Prix du panier : ".$totalPanier.".00€</p>");
    echo('<input type="submit" id="commanderPanier" name="Commander" value="Commander &#x27A4;">');
}

echo('</div>');
```

```

$idActeur = $_SESSION['id'];
$couleur = $_POST['productCouleur'];
$product_id = $_POST['productID'];
$quantite = $_POST['quantity'];

if($quantite == 0){
    $reqDelete = "DELETE FROM PANIER WHERE idArticle = :product_id AND idActeur = :idActeur AND couleur = :couleur";
    $panierDelete = oci_parse($connect, $reqDelete);
    oci_bind_by_name($panierDelete, ':product_id', $product_id);
    oci_bind_by_name($panierDelete, ':idacteur', $idActeur);
    oci_bind_by_name($panierDelete, ':couleur', $couleur);
    $resultDelete = oci_execute($panierDelete);

    if (!$resultDelete) {
        $e = oci_error($resultDelete); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
        $SQLerror = htmlentities($e['message'] . ' pour cette requête : ' . $e['sqltext']);
    }

    echo("<script language='JavaScript' type='text/javascript'>
        | alert('Produit retiré du panier');
        | location.href = './panier.php';
        </script>");
} else {
    $requeteUpdate = "UPDATE PANIER SET quantiteArticle = :quantite WHERE idArticle = :product_id AND idActeur = :idActeur AND couleur = :couleur";
    $reqUpdate = oci_parse($connect, $requeteUpdate);

    oci_bind_by_name($reqUpdate, ':product_id', $product_id);
    oci_bind_by_name($reqUpdate, ':quantite', $quantite);
    oci_bind_by_name($reqUpdate, ':idacteur', $idActeur);
    oci_bind_by_name($reqUpdate, ':couleur', $couleur);

    $resultUpdate = oci_execute($reqUpdate);

    if (!$resultUpdate) {
        $e = oci_error($reqUpdate); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
        $SQLerror = htmlentities($e['message'] . ' pour cette requête : ' . $e['sqltext']);
    }

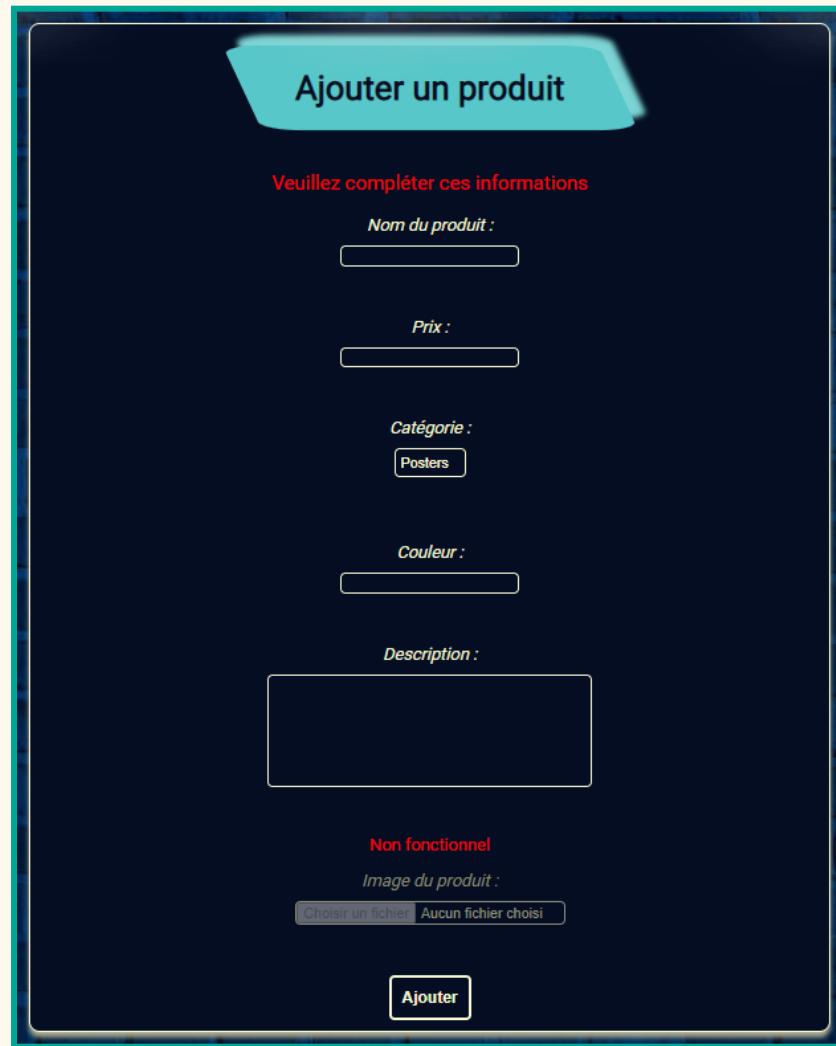
    echo("<script language='JavaScript' type='text/javascript'>
        | alert('Quantité modifiée');
        | location.href = './panier.php';
        </script>");
}
}

```

## 5. Ajout d'un produit (artiste) - visuel & fonctionnel

Uniquement disponible pour un artiste, celui-ci a la possibilité d'ajouter un produit à exposer depuis le site d'Art'i'Chaud. Cette fonctionnalité n'est alors pas disponible pour un administrateur qui n'est tout simplement pas considéré comme un artiste. Le coloris n'est cependant, pas disponible d'être ajouté (mais le fera futurement via un autre lien hypertexte à part entière), tout comme l'image pour des problèmes de droits. Ainsi, les informations à renseigner correspondent aux informations nécessaires à rajouter dans notre base de données dédiée à l'article, à l'exception de l'ID de l'article qui sera également incrémenté automatiquement, tout comme l'inscription d'un utilisateur. Par conséquent, la page concernant la catégorie du produit sera également mise à jour pour faire figurer le produit ajouté.

**Maquette associée :**



Code (`addProduit.php` et `traitAddProduit.php`):

```

<form action="traitAddProduit.php" id="FormInscription" method="post">
<div id='titleFrame'><h1 id='formTitle'>Ajouter un produit</h1></div>
<?php
if(isset($_GET['msgErreur'])){
    echo("<h3 class='errorDatabase'>" . $_GET['msgErreur'] . "</h3>");
} else {
    echo("<h3 class='errorDatabase'>Veuillez compléter ces informations</h3>");
}
?>

<label class="formLabels">Nom du produit :</label>
<input type="text" class="modifChamps" name="name" required>
<br/><br/>

<label class="formLabels">Prix :</label>
<input type="number" class="modifChamps" name="prix" step="0.01" min="0.01" required/>
<br/><br/>

<label class="formLabels">Catégorie :</label>
<select name='categorie' id='selectArticles'>
    <option class="optionArticles" value="POSTERS">Posters</option>
    <option class="optionArticles" value="FIGURINES">Figurines</option>
    <option class="optionArticles" value="PELUCHES">Peluches</option>
</select>
<br/>

<label class="formLabels">Couleur :</label>
<input type="text" class="modifChamps" name="couleur"/>
<br/><br/>

<label class="formLabels">Description :</label>
<textarea id="zoneCollab" class="champsForm" rows="8" cols="40" name="comment" required></textarea>
<br/>

<p class='errorDatabase'>Non fonctionnel</p>
<label id="blockFormLabel">Image du produit :</label>
<input type="file" id="unmodifChamp" name="monImage" disabled/>
<br/><br/>

<input type="submit" name="Ajouter" value="Ajouter" id="btnInscription"/>
</form>

```

```

$req = "INSERT INTO ARTICLE (IDARTICLE,NOMARTICLE,NOMCATEGORIE,PRIXARTICLE,PRIXPROMO,DESCRIPTION)
VALUES(:SEQ_ARTICLE.NEXTVAL,:pNomProduit,:pNomCategorie,:pPrix,0,:pDesc)";
$articleInsert = oci_parse($connect, $req);
oci_bind_by_name($articleInsert, ":pNomProduit", $nomProduit);
oci_bind_by_name($articleInsert, ":pNomCategorie", $_POST['categorie']);
oci_bind_by_name($articleInsert, ":pPrix", $prixProduit);
oci_bind_by_name($articleInsert, ":pDesc", $descProduit);
//oci_bind_by_name($articleInsert, ":pCouleur", $couleurProduit);

$result = oci_execute($articleInsert);

if (!$result) {
    $e = oci_error($articleInsert);
    $SQLerror = htmlentities($e['message'] . ' pour cette requête : ' . $e['sqltext']);

    header('Location: ./addProduit.php?msgErreur=' . $SQLerror);
    exit;
}

oci_commit($connect);

oci_free_statement($articleInsert);

echo('<script language="JavaScript" type="text/javascript">
    alert("Ajout du produit réussi.");
    location.href = "./connexion.php";
</script>');

```

## 6. Mise à jour d'un produit (admin) - visuel & fonctionnel

L'une des fonctionnalités spécifiques à un administrateur est de pouvoir modifier un produit qui figure déjà sur le site web, bien que l'image ne puisse pas être actualisée (par problèmes de paramétrage de celui-ci) ainsi que le coloris qui lui, sera fonctionnel très bientôt.

Autrement, celui-ci peut modifier les caractéristiques d'un produit selon l'ID du produit associé à son nom. Le prix, pour plus de cohérence, sera renseigné avec une virgule. Pour autant, si le prix est un prix exact, il sera nécessaire de faire figurer les "0" de trop dans les pages affichant le prix du produit. Le prix promo correspond à un pourcentage, c'est-à-dire un nombre compris entre 0 et 99. Pour les fonctionnalités non-traitées ou non-fonctionnelles, cette précision est figurée sur le site où les champs ne sont pas activés pour la saisie.

**Maquette associée :**

La maquette de formulaire 'Modifier un produit' est présentée dans un cadre bleu. Le titre 'Modifier un produit' est en haut à gauche. Un message d'erreur rouge 'Veuillez compléter ces informations' est visible au-dessus des champs. Les champs sont organisés comme suit :

- Votre produit :** Champ avec valeur 'article1'.
- Nouveau nom de produit :** Champ vide.
- Catégorie :** Champ avec valeur 'Figurines'.
- Nouveau prix de produit :** Champ vide.
- Nouveau prix promo de produit :** Champ vide.
- Non fonctionnel** : Section marquée par un texte rouge. Elle contient :
  - Image du produit :** Champ avec boutons 'Choisir un fichier' et 'Aucun fichier choisi'.
  - Nouvelle description :** Champ vide.
- Sauvegarder** : Bouton en bas à droite.

### Code (*updateProduit.php* et *traitUpdateProduit.php*) :

```

echo('<form method="post" id="FormModification" action="traitUpdateProduit.php">');
echo('<div id="titleFrame"><h1 id="formTitle">Modifier un produit</h1></div>');
if(isset($_GET['msgErreur'])) {
    echo("<h3 class='errorDatabase'>".$_GET['msgErreur']."</h3>");
} else {
    echo("<h3 class='errorDatabase'>Veuillez compléter ces informations</h3>");
}
echo('<label class="formLabels">Votre produit :</label>');
echo('<select name="nomArticle" id="selectThemes">');

$req = "SELECT IDARTICLE, NOMARTICLE FROM ARTICLE";
$articlesRecup = oci_parse($connect, $req);
$result = oci_execute($articlesRecup);

if ($result) {
    $e = oci_error($articlesRecup); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
    $errorSQL = print htmlentities($e['message']) . ' pour cette requête : ' . $e['sqltext'];
    header('Location: ./updateProduit.php?msgErreur=' . $errorSQL);
    exit;
} else {
    while($nomArticles = oci_fetch_assoc($articlesRecup)) != false) {
        echo('<option class="optionArticles" value="'.$nomArticles['IDARTICLE'].'">' . $nomArticles['NOMARTICLE'] . '</option>');
    }
}

echo('</select>');
echo('<br/><br/>');
echo('<label class="formLabels">Nouveau nom de produit :</label>');
echo('<input type="text" class="modifChamps" name="nomProduit" required"/>');
echo('<br/><br/>');

echo('<label class="formLabels">Catégorie :</label>');
echo('<select name="categorie" id="selectArticles">');
echo('<option class="optionArticles" value="FIGURINES">Figurines</option>');
echo('<option class="optionArticles" value="POSTERS">Posters</option>');
echo('<option class="optionArticles" value="PELUCHES">Peluches</option>');
echo('</select>');
echo('<br/><br/>');

echo('<label class="formLabels">Nouveau prix de produit :</label>');
echo('<input type="number" class="modifChamps" name="prixProduit" step="0.01" min="0.01" maxlength="8" required"/>');
echo('<br/><br/>');

echo('<label class="formLabels">Nouveau prix promo de produit :</label>');
echo('<input type="number" class="modifChamps" name="prixPromoProduit" step="1" min="0" maxlength="2" required"/>');
echo('<br/>');

echo('<p class="errorDatabase">Non fonctionnel</p>');
echo('<label id="blockFormLabel">Image du produit :</label>');
echo('<input type="file" id="unmodifChamp" name="monImage" disabled/>');
echo('<br/><br/>');

echo('<label class="formLabels">Nouvelle description :</label>');
echo('<textarea type="text" class="champsForm" id="zoneCollab" name="descProduit" required/>');
echo('</textarea>');

echo('<br/><br/>');

echo('<input type="submit" name="Sauvegarder" value="Sauvegarder" id="btnModification"/>');

echo('</form>');

```

```

//On crée une requête paramétrée
$req = "UPDATE ARTICLE
        SET NOMARTICLE = :pArticle,
            NOMCATEGORIE = :pNomCategorie,
            PRIXARTICLE = :pPrixarticle,
            PRIXPROMO = :pPrixPromo,
            DESCRIPTION = :pDescription
        WHERE IDARTICLE = '$ancienProduit'";

//On prépare la requête
$nouveauArticleRecup = oci_parse($connect, $req);

//On lie les valeurs aux paramètres de la requête
oci_bind_by_name($nouveauArticleRecup, ":pArticle", $nomProduit);
oci_bind_by_name($nouveauArticleRecup, ":pNomCategorie", $POST['categorie']);
oci_bind_by_name($nouveauArticleRecup, ":pPrixArticle", $prixProduit);
oci_bind_by_name($nouveauArticleRecup, ":pPrixPromo", $prixPromoProduit);
oci_bind_by_name($nouveauArticleRecup, ":pDescription", $descProduit);

$result = oci_execute($nouveauArticleRecup);

if ($result) {
    $e = oci_error($nouveauArticleRecup); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
    $SQLerror = htmlentities($e['message']) . ' pour cette requête : ' . $e['sqltext'];
    header('Location: ./updateProduit.php?msgErreur=' . $SQLerror);
    exit;
}

oci_commit($connect);
oci_free_statement($nouveauArticleRecup);

echo('<script language="JavaScript" type="text/javascript">
    alert("Modification réussie.");
    location.href = "/connexion.php";
    </script>');

```

## 7. Retrait d'un produit (admin) - visuel & fonctionnel

Enfin, une dernière fonctionnalité uniquement accessible si l'utilisateur est un administrateur du site web, il a également la possibilité de supprimer un article parmi les articles exposés chez Art'i'Chaud. Son formulaire est très simple puisqu'à partir de l'ID d'un article, ses caractéristiques seront également effacées bien que ce soit le nom d'un article qui soit figuré depuis la liste déroulante pour plus de visibilité, d'où la requête SQL ne récupérant uniquement les attributs nécessaires.

**Site web :**



**Code (`deleteProduit.php` et `traitDeleteProduit.php`) :**

```

echo( <form method="post" id="formModification" action="traitDeleteProduit.php"> );
echo( "<div id=titleFrame><h1 id='formTitle'>Retrait d'un produit</h1></div>" );
echo( "<h3 class=errorDatabase>Veuillez choisir un produit à retirer</h3>" );
echo( "<label class=formLabels>Votre produit :</label>" );
echo( "<select name='nomArticle' id='selectThemes'>" );

$req = "SELECT IDARTICLE, NOMARTICLE FROM ARTICLE";
$articlesRecup = oci_parse($connect, $req);
$result = oci_execute($articlesRecup);

if ($result) {
    $e = oci_error($articlesRecup); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
    $errorSQL = print htmlentities($e['message'], 'pour cette requête : ' . $e['sqltext']);
    header("Location: ./updateProduit.php?msgErreur=$errorSQL");
    exit();
} else {
    while(($nomArticles = oci_fetch_assoc($articlesRecup)) != false) {
        echo("<option class='optionArticles' value='".$nomArticles['IDARTICLE']."'>" . $nomArticles['NOMARTICLE'] . "</option>");
    }
}
echo("</select>");
echo("<br/><br/>");
echo("<input type='submit' name='Retirer' value='Retirer' id='btnModification' />");

echo( "</form>" );

```

```

$produitARetirer = $_POST['nomArticle'];
echo($produitARetirer);

//On crée une requête paramétrée
$req = "DELETE FROM ARTICLE WHERE IDARTICLE = '$produitARetirer'";
//On prépare la requête
$deleteProduit = oci_parse($connect, $req);

$result = oci_execute($deleteProduit);
oci_commit($connect);
oci_free_statement($deleteProduit);

echo( '<script language="JavaScript" type="text/javascript">
        alert("Retrait réussi.");
        location.href = "./connexion.php";
    </script>');

```

## Finalisation des pages - gestion des procédures stockées et exceptions

### 1. Page paiement - visuel & fonctionnel

Bien que cette page a été avancée grandement, celle-ci ne s'avère pas finalisée (en plus de certaines autres pages).

Concrètement, l'utilisateur aura la possibilité de choisir son mode de paiement parmi le paiement par carte bancaire ou crypto-monnaie. Selon son choix, un deuxième formulaire sera affiché, lui informant de compléter les informations relatives au paiement, et conforme à la table reliée. Les champs respectent également une REGEX (comme l'entièreté des autres champs des autres formulaires), bien qu'ici, il s'agit d'informations très sensibles et nécessitant de correspondre à des données réelles de paiement par carte bancaire ou de crypto-monnaie par exemple.

Toutefois, le procédé se réalisera par le biais d'une procédure qui, lorsque l'utilisateur aura soumis le formulaire de paiement, réalisera :

- La création de la commande concernée ;
- Le fait de basculer les informations du panier dans une table intitulée "DETAILCOMMANDÉ" afin d'en observer un récapitulatif de celui-ci ;
- De vider le panier, une fois la commande réalisée ;
- D'ajouter le mode de paiement selon le choix de l'utilisateur.

Cet aspect est important et nécessite d'être traité avec sérieux. Concrètement, si les informations bancaires de l'utilisateur figurent déjà dans la base de données mais qu'il décide à ce qu'on ne se souvienne pas de lui, ces informations doivent être supprimées. Autrement, si ces informations ne sont pas disponibles dans la base de données mais qu'il décide à ce qu'on se souvienne de son mode de paiement, celui-ci sera alors ajouté.

Pour la crypto-monnaie, ce ne sera simplement qu'un numéro de référence à renseigner. Evidemment, si celui-ci existe déjà, une erreur lui sera affichée. Ces paiements étant axés sur des numéros de référence, ceux-ci s'avèrent être uniques et ne peuvent pas être sauvegardés indépendamment.

**Site web :**This block contains three screenshots of a payment form. The first screenshot shows a "Carte bancaire" section with fields for card support (Visa), card number, name, expiration date, and CVV, along with a "Se souvenir de ma CB ?" checkbox and a "Paiement" button. The second screenshot shows a "Paiement" section with a "Carte bancaire" and "Cryptomonnaie" button. The third screenshot shows a "Cryptomonnaie" section with a "Support de paiement" field (Bitcoin selected), a reference number field, and a "Paiement" button.

**Code (*paiement.php* et *traitPaiement.php*):**

```

echo('<form method="post" id="formModePaiement">');

echo('<div id="titleFrame"><h1 id="formTitle">Paiement</h1></div>');

echo('<div id="gridPaiement">');
    echo('<button name="carteBancaire" class="btnModePaiement">');
        echo('');
        echo('<h3 class="textePaiement">Carte bancaire</h3>');
    echo('</button>');

    echo('<button name="cryptomonnaie" class="btnModePaiement">');
        echo('');
        echo('<h3 class="textePaiement">Cryptomonnaie</h3>');
    echo('</button>');

    echo('</div>');

echo('</Form>');

if(isset($_POST['carteBancaire'])) {
    $_SESSION['moyenPaiement'] = 'carteBancaire';
}

if(isset($_POST['cryptomonnaie'])) {
    $_SESSION['moyenPaiement'] = 'cryptomonnaie';
}

```

```

if(isset($_SESSION['moyenPaiement'])) {
    if($_SESSION['moyenPaiement'] == 'carteBancaire') {

        echo('<form method="post" class="formPaiement" action="traitPaiement.php">');

        echo('<br/>');
        echo('<div id="littleTitleFrame"><h2 id="formTitle">Carte bancaire</h2></div>');

        if(isset($_GET['msgErreur'])){
            echo('<h3 class="errorDatabase">'.$_GET['msgErreur'].'</h3>');
        } else {
            echo('<h3 class="errorDatabase">Veuillez compléter ces informations</h3>');
        }

        echo('<label class="formLabels">Support de paiement :</label>');
        echo('<select name="paientmentC" class="selectPaiement">');
            echo('<option class="optionMonnaies" value="V">Visa</option>');
            echo('<option class="optionMonnaies" value="M">Mastercard</option>');
        echo('</select>');
        echo('<br/>');

        echo('<label class="formLabels">Numéro de carte bancaire :</label>');
        echo('<input type="text" class="paiementChamps" name="numB" maxlength="16" required/>');
        echo('<br/><br/>');

        echo('<label class="formLabels">Prénom et nom sur votre carte bancaire :</label>');
        echo('<input type="text" class="paiementChamps" name="namesB" required/>');
        echo('<br/><br/>');

        echo('<label class="formLabels">Date d\'expiration :</label>');
        echo('<input type="text" class="paiementChamps" name="expiration" maxlength="4" required/>');
        echo('<br/><br/>');

        echo('<label class="formLabels">Cryptogramme :</label>');
        echo('<input type="text" class="paiementChamps" name="cryptogramme" maxlength="3" required/>');
        echo('<br/>');

        echo('<label class="formLabels">Se souvenir de ma CB ?</label>');
        echo('<label id="checkboxLabel">');
        echo('<input type="checkbox" id="checkboxInput" name="memoCB"/>');
        echo('<svg id="checkboxCheck">');
        echo('<polyline points="20 6 9 17 4 12"/></polyline>');
        echo('</svg>');
        echo('<label>');
        echo('<br/><br/>');

        echo('<input type="submit" name="PaiementCB" value="Paiement" id="btnPaiement"/>');

        echo('</form>');

    }
}

```

```

if($_SESSION['moyenPaiement'] == 'cryptomonnaie') {

    echo('<form method="post" class="formPaiement" action="traitPaiement.php">');

    echo('<br/>');
    echo('<div id="littleTitleFrame"><h2 id="formTitle">Cryptomonnaie</h2></div>');

    if(isset($_GET['msgErreur'])){
        echo('<h3 class="errorDatabase">'.$_GET['msgErreur'].'</h3>');
    } else {
        echo('<h3 class="errorDatabase">Veuillez compléter ces informations</h3>');
    }

    echo('<label class="formLabels">Support de paiement :</label>');
    echo('<select name="paientmentcrypto" class="selectPaiement">');
        echo('<option class="optionMonnaies" value="B">Bitcoin</option>');
        echo('<option class="optionMonnaies" value="E">Ethereum</option>');
    echo('</select>');
    echo('<br/>');

    echo('<label class="formLabels">Numéro de référence :</label>');
    echo('<input type="text" class="paiementChamps" name="numRef" required/>');
    echo('<br/><br/>');

    echo('<input type="submit" name="PaiementCR" value="Paiement" id="btnPaiement"/>');

    echo('</form>');

}

```

```

if(preg_match('#^-[0-9]{16}$', $numeroCB)) {
    header('Location: ./paiement.php?msgErreur=Vérifiez la saisie de votre numéro de carte bancaire');
    exit;
}

if(preg_match('#^([0-9]{1}[0-9])|([0-9]{2}[0-9])|([3-9]{3}[0-9])$#', $exp)) {
    header('Location: ./paiement.php?msgErreur=Vérifiez la saisie de votre date d\'expiration');
    exit;
}

if(preg_match('#^-[0-9]{3}#$', $crypto)) {
    header('Location: ./paiement.php?msgErreur=Vérifiez la saisie de votre cryptogramme');
    exit;
}

$req = "SELECT * FROM CARTEBANCAIRE WHERE IDATEUR = :pIdActeur";
$CBRecup = oci_parse($connect, $req);
oci_bind_by_name($CBRecup, ":pIdActeur", $id);
$result = oci_execute($CBRecup);

if ($result) {
    $e = oci_error($CBRecup); // on récupère l'exception liée au pb d'exécution de la requête (Violation PK par exemple)
    $SQLerror = htmlentities($e['message']) . ' pour cette requête : ' . $e['sqltext'];

    header('Location: ./inscription.php?msgErreur=' . $SQLerror);
    exit;
}

if(isset($_POST['memoCB'])) {
    if(empty(oci_fetch_assoc($CBRecup))) {
        oci_free_statement($cryptoRecup);
        echo('<script language="JavaScript" type="text/javascript">
            alert("Okay. Ajout de ta CB en cours... (procédure)");
            location.href = "./index.php";
        </script>');
    }
} else if(!empty(oci_fetch_assoc($CBRecup))) {
    oci_free_statement($cryptoRecup);

    echo('<script language="JavaScript" type="text/javascript">
        alert("Okay. Retrait sur ta CB en cours... (procédure)");
        location.href = "./index.php";
    </script>');
}

echo('<script language="JavaScript" type="text/javascript">
    alert("Transaction effectuée.");
    location.href = "./index.php";
</script>');

```

```

if(isset($_POST['PaiementCR'])) {

    if(empty($_POST['paiementcrypto'])) {
        header('Location: ./paiement.php?msgErreur=Sélection incorrecte');
        exit;
    }

    if(!isset($_POST['numRef'])) {
        header('Location: ./paiement.php?msgErreur=Veuillez remplir le champ');
        exit;
    }

    $numReference = htmlspecialchars($_POST['numRef']);

    if(!preg_match('#^-[1-9][0-9]{1,9}$#', $numReference)) {
        header('Location: ./paiement.php?msgErreur=Vérifiez la saisie de votre référence');
        exit;
    }

    $req = "SELECT * FROM CRYPTOMONNAIE WHERE NUMREF = :pNumReference";
    $cryptoRecup = oci_parse($connect, $req);
    oci_bind_by_name($cryptoRecup, ":pNumReference", $numReference);
    $result = oci_execute($cryptoRecup);

    if (!$result) {
        $e = oci_error($cryptoRecup); // on récupère l'exception liée au pb d'exécution de la
        $errorSQL = print htmlentities($e['message']) . ' pour cette requête : ' . $e['sqltext'];
        header('Location: ./paiement.php?msgErreur=' . $errorSQL);
        exit;
    }

    if(oci_fetch_assoc($cryptoRecup)) {
        header('Location: ./paiement.php?msgErreur=Numéro de référence déjà existant');
        exit;
    }

    oci_free_statement($cryptoRecup);

    echo('<script language="JavaScript" type="text/javascript">
        alert("Okay. Ajout de ta ref en cours... (procédure)");
        location.href = "./index.php";
    </script>');
}

```