**TP 2 ARCHITECTURE MVC PHP**

On retrouve donc les 2 cas d'utilisation du TP 1 :

Consulter les confiseries🡪 c\_voirProduits.php ,

Gérer votre Panier 🡪 c\_gestionPanier.php

La partie ***Gérer votre panier*** va être complétée avec la saisie et l'enregistrement de la commande. Le passage de la commande se fera à partir d’un bouton que l’on rajoutera en fin d’affichage du panier. L’internaute entrera ses données personnelles et les validera.

La base de données contient normalement la structure des tables commande et contenir.

Que contient la table ***contenir*** ? Donnez un exemple avec des valeurs ?

La table contient l’id commande et l’id produit.

Idproduit=1

Idcommande=14

Que doit-on rajouter au scénario du cas **gérer le panier** pour compléter les actions que l’utilisateur peut effectuer

Le scenario à ajouter au cas gérer le panier est le scénario pour passer la commande.

Conformez-vous aux règles indiquées dans la fiche méthodologique pour élaborer ce scenario alternatif

Une partie du formulaire de saisie de la commande a été réalisée. Elle vous est fournie dans le dossier **Vanille\_TP2.**

# Première partie : Passer la commande

**TRAVAIL A FAIRE**

1. Ajouter le bouton nécessaire à l’affichage du formulaire.
2. <ul id="menu">
3. <li>
4. <a href=index.php?uc=gererPanier&action=passerCommande>Passer commande</a>
5. </li>

8. </ul>
9. Incorporer dans l’application le formulaire de saisie fourni.
10. <p>
11. <label for="nom">Nom Prénom\*</label>
12. <input id="nom" type="text" name="nom" value="<?=$nom ?>" size="30" maxlength="45" required>
13. </p>
14. <p>
15. <label for="rue">Rue\*</label>
16. <input id="rue" type="text" name="rue" value="<?=$rue ?>" size="30" maxlength="45" required>
17. </p>
18. <p>
19. <label for="cp">Code postal\* </label>
20. <input id="cp" type="text" name="cp" value="<?=$cp ?>" size="5" maxlength="5" required >
21. </p>
23. <p>
24. <input type="submit" value="Valider" name="valider">
26. </p>
27. Compléter le code pour rendre effectif l’affichage du formulaire de saisie de la commande.
28. Quelle erreur peut se produire ? Que doit-on ajouter au scénario ?

*Lors de la saisie des informations du formulaire par le client il peut y avoir une erreur sur le code postale. Le code postal doit obligatoirement contenir 5 chiffres.*

# Deuxième partie : Confirmer la commande

Après saisie des données du client, la validation devra provoquer l’enregistrement des informations dans la base de données.

La fonction ***creerCommande*** vous est fournie.

Où doit-on l’intégrer dans l’application ?

On doit intégrer créerCommande dans « fonction.inc.php »

**TRAVAIL A FAIRE**

1. Compléter le code pour permettre l’enregistrement de la commande en complétant par les 2 actions suivantes
   * + 1. Le message « commande enregistrée » sera affiché
       2. Le panier sera supprimé
2. Finaliser le formulaire pour disposer de l’ensemble des données du client pour l’enregistrement de la commande.

# Troisième partie : Finalisation de la commande

Que manque t-il pour que l’ensemble de la commande soit enregistrée?

Pour être enregistrée, la commande doit être insérée dans la base de données.

**TRAVAIL A FAIRE**

1. Compléter le code pour permettre l’enregistrement de l’ensemble des éléments de la commande
2. public function creerCommande($nom,$rue,$cp, $lesIdProduit )
3. {
4. $req = "select max(CDE\_id) as maxi from commande";
5. $res = PdoVanille::$monPdo->query($req);
6. $laLigne = $res->fetch();
7. $maxi = $laLigne['maxi'] ;
8. $maxi++;
9. $idCommande = $maxi;
10. echo $idCommande."<br>";
11. $date = date('Y/m/d');
12. $req = "insert into commande values ('$idCommande','$date','$nom','$rue','$cp',null,null)";
13. echo $req."<br>";
14. $res = PdoVanille::$monPdo->exec($req);
15. /\*\* A RAJOUTER INSERTION DANS LA TABLE CONTENIR
16. balayage du tableau des produits
17. \*/
18. foreach( $lespoduits as $lesIdProduit){
19. $req="insert into contenir values('$idCommande','$lesIdProduit[$lesproduits]')";
20. PdoVanille::$monPdo->exec($req);
21. }
22. }

# Quatrième partie : Contrôle de la commande

On vous fournit 2 fonctions permettant de tester le format du code postal (valeur entière et sur 5 caractères) lorsqu’il est saisi par l’internaute.

On vous donne également le code de la fonction ***getErreursSaisieCommande*** permettant de les utiliser.

On avertira l’internaute en utilisant la vue ***v\_erreurs.php*** en cas d’erreurs de saisies détectées.

**TRAVAIL A FAIRE**

1. Intégrer l’utilisation de ces fonctions avant la validation de la commande par l’utilisateur.

# Cinquième partie : Bonus



*VOTRE MISSION SI VOUS L’ACCEPTEZ :*

On désire améliorer l’application en assurant la gestion des quantités de chacun des produits du catalogue.

Compléter votre application avec

* Ajout d’une quantité en stock qui devra être mise à jour à chaque commande de produit.
* Ajout de la quantité commandée par le client, en utilisant le tableau des produits qui devient associatif (id produit devient l'index du tableau et pointe sur la quantité commandée)