**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 ***contenir*** contient les champs : *idcommande* et *idProduit*

Exemple :

- *idcommande* : 001

*-idProduit*: CA01

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

L’utilisateur doit pouvoir passer une commande avec les produits qu’il a choisit et pouvoir ensuite valider sa commande (en renseignant diverses informations personnelles : nom, prénom, adresse, …)

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. Incorporer dans l’application le formulaire de saisie fourni
3. Compléter le code pour rendre effectif l’affichage du formulaire de saisie de la commande.
4. Quelle erreur peut se produire ? Que doit-on ajouter au scénario ?

# 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 ?

**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 ?

**TRAVAIL A FAIRE**

1. Compléter le code pour permettre l’enregistrement de l’ensemble des éléments de la commande

# 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)