



# ISIWEBSShop4

28.01.2019

---

Ineida CARDOSO MORENO  
Smailine VIRARAGAVANE

## Objectifs

1. Développement d'un site web dynamique, pour une commerce.
2. Les clients peuvent commander:
  - a. En se connectant ou non.
  - b. Payer sous différentes formes.
  - c. choisir l'adresse de livraison/facturation.
3. L'administrateur:
  - a. Connecte et gère les commandes en cours.

## Grandes étapes

### I. Création de la base de données et la page de dialogue avec la base.

Le script pour la base de donnée a été fourni par le professeur.

On a pris le soin de rajouter les références d'une table à une autre.

La page de dialogue nous à permis de créer tous les fonctions dont nous avons besoin pour communiquer avec la base de donnée, dans la classe DialogueBD.

### II. Mise en place de différentes vues grâce aux contrôleurs

Grâce à différents contrôleurs comme connexion\_controleur, panier\_controleur ... permettent de gérer l'affichage de chaque vues et chaque actions de l'utilisateur. En effet, pour permettre l'affichage de la liste des produits par exemple, nous avons pour cela mis dans son contrôleur, un traitement permettant de récupérer les produits et ainsi les afficher dan

### III. Mise en place de session

Le grand défi de ce projet était de gérer les comptes qui sont enregistrés et les comptes non enregistrés c'est-à-dire des personnes qui peuvent faire des achats sans pour autant avoir un compte. Pour cela, nous avons utiliser session\_id() qui permet de donné un id pour une personne utilisant le site. De plus, quand nous

avons une session qui est ouverte pour un utilisateur, alors nous ajoutons dans la table customer en mettant à null chaque attribut qui seront bien sûr compléter lors du passage de l'utilisateur dans la partie renseignement du client. Pour gérer aussi cette différence et pour nous faciliter la tâche, nous récupérons l'id du client stocké dans la base pour ainsi faire certains traitement dans la commande par exemple.

## Répartition des tâches:

La répartition des tâches, s'est faite de façon naturelle. Chacune a réalisé une partie: Smailine s'est occupée de la partie concernant les transactions avec le panier. Le paiement, le panier, le renseignement concernant l'achat, et la liste des produits, la suppression d'une commande que ce soit dans un compte connecté ou non connecté.

Tandis que, Ineida s'est occupée de la page d'accueil, d'enregistrement, la génération de la facture, et la partie concernant l'administrateur et la liste des commandes, mise en place de la base de données.

On s'est aidé pour corriger les erreurs, et dans les parties le plus compliqués. Même si on s'est répartie les tâches. Pour la plupart on a participé à deux.

## Difficultés:

Pour la plupart des erreurs et difficultés rencontrées, lorsqu'une n'arrivait pas à les résoudre ou bien ne voyait pas où était le problème, elle faisait appel à l'autre. Les parties que nous ont le plus posé problème sont la gestion de l'achat pour un client qui n'est pas enregistré dans la base et que ne le souhaite pas. Et la génération de la facture.

### 1. La facture:

Réaliser une facture en pdf, n'était pas tellement facile. On ne savait pas comment faire au départ. Puis, on a choisi de faire la réaliser à l'aide de la classe FPDF, qu'on a trouvé sur internet: [www.fpdf.org](http://www.fpdf.org).

On crée une classe pdf qui hérite de la classe fpdf, qu'on a utilisé pour générer la facture. Cependant, la prise en main de cette classe était un peu compliqué, On a eu du mal, à appréhender les méthodes de la classe.

Concernant le design, on a choisit d'utiliser surtout Bootstrap et un peu du css.

## 2. La gestion des comptes non utilisateurs

Nous avons commencé par gérer les actions des utilisateurs connectés via un compte car cela nous paraissait plus facile que de gérer un client non enregistré. Mais en repensant à la fin aux traitements des clients non enregistrés, nous avons perdu du temps car cela entraînait des modifications dans notre code, en mettant plusieurs conditions et vérifications.

Nous avions aussi du mal à comprendre l'utilité de `session_id`.

De plus, nous sommes partis du principe que `l'add1` et `add2` qui se trouve dans `customers` sont les deux adresses donné par l'utilisateur. Et ainsi chacun de ces adresses ont un id dans la table `delivery_addresses`.

## 3. Difficulté à gérer différentes actions

Nous avons eu des difficultés à gérer les différentes actions de l'utilisateur. Nous avons un problème sur l'affichage du panier. En effet, quand nous ajoutons ou supprimons un article ou une commande, la modification sur le panier(dans le vue) n'est pas perçu instantanément. Nous devons rafraichir la page pour constater la modification. Mais nous avons pu y remédier en mettant le `include` du controleur assez haut afin que les traitement puisse passer par lui avant la vue. De plus, **il faut s'assurer que les sessions et les caches soient effacés pour qu'un autre client non enregistré puisse commencer**

**La difficulté principale est la gestion de notre temps car nous nous sommes retardés sur certains détails comme la gestion des sessions, la facture par exemple et ainsi cela s'est répercuté sur l'ensemble du projet.**