

Projet PHP - Présentation technique

Mon projet PHP consiste à regrouper des produits alimentaires en réduction de certains magasins locaux. Les commerçants mettent les articles en solde sur l'application en indiquant le prix de base, le nouveau prix, le pourcentage de réduction qu'il y a, la date de péremption et une brève description de l'article. Grâce aux fonctions de tri sur l'application, les clients peuvent trier leurs recherches par préférence ou mettre un mot clé dans la case search qui se trouve sur la page. En cliquant sur les icônes « Info », ils peuvent connaître les informations sur les fournisseurs et en cliquant sur l'icône « Réserver », ils peuvent réserver un produit. Après cette étape, ils doivent dire qui ils sont afin de pouvoir mettre le produit dans Réserveation.

Projet PHP

Réserveations

Show 10 entries

Search:

Genre	Prix Initial	Prix Rabais	Pourcentage de Rabais	Date de Peremption	Description	Fournisseur	Action
Fruit	30	27	10	2017-03-30	Fruit de la passion - 5kg	Info	Réserver
Fruit	14	10	30	2017-03-23	Manges du Brésil - 6 pièces	Info	Réserver
Poisson	25	19	25	2017-03-09	Saumon d'Irlande - 500gr	Info	Réserver
Poisson	20	10	50	2017-03-09	Colin de Norvège - 200gr	Info	Réserver
Viande	15	8	50	2017-03-02	Viande de boeuf - 500gr	Info	Réserver
Viande	35	26	25	2017-03-08	Viande d'agneaux - 600gr	Info	Réserver
Viande	16	14	10	2017-03-14	Steak de boeuf	Info	Réserver
Viande	23	18	20	2017-03-02	Filet d'agneaux - 200gr	Info	Réserver

Showing 1 to 8 of 8 entries

PreviousNext

Installation :

Afin de créer cette application, j'ai mis en place une base de données MySQL sur phpmyadmin qui contient quatre tables, « article », « client », « fournisseur » et « reservation ». J'ai également utilisé des templates smarty et une connexion à la base de données en PDO.

Justification des choix :

Pour rendre l'interface plus jolie, j'ai choisi de prendre des exemples de Bootstrap qui est un framework qui a beaucoup d'outils pour la création du design de sites et d'applications web, comme par exemple la nav-barre sur mon application, qui est disponible sur le site de Bootstrap. J'ai également utilisé des requêtes JQuery pour le côté dynamique du site et pour le rendre plus joli, j'ai inséré des icônes (« Info » et « Réserver ») qui proviennent du site internet : font-awesome.io. Sur le site de font-awesome, nous pouvons trouver des icônes qui sont vectorisés et qu'on peut modifier comme on veut grâce au CSS. Pour pouvoir trier les articles alimentaires sur mon application, j'ai utilisé DataTables. C'est un plug-in pour le JQuery. C'est un outil très flexible qui permet d'ajouter des contrôles d'interaction avancés à

n'importe quelle table HTML. Avec DataTables on peut avoir un système de pagination, ordonner des colonnes et la fonction « search » que j'ai également utilisé sur mon application.

J'ai utilisé du JavaScript pour le changement de page lors du clique sur le bouton Reservation.

Dans le tableau, j'ai choisi de mettre les prix des articles alignés à droite et pour ce faire, j'ai créé une classe alignRight dans le fichier CSS.

Les noms « fid » et « aid » signifient fournisseur id, l'identifiant du fournisseur et article id, l'identifiant de l'article.

J'ai choisi de mettre des pop-up, des modals, pour rendre l'interface plus jolie. J'ai copié le code du site de Bootstrap et pour l'afficher, je fais des requêtes ajax parce que c'est plus simple.

Pour le modal de la réservation, on va chercher la liste des clients pour savoir qui réserve quel produit. Quand on clique sur le bouton Réserver, ça sauvegarde l'identifiant de l'article et ensuite ça ouvre le modal de la réservation où on doit sélectionner qui l'on est. Quand on appuie sur le bouton Réserver du pop-up, ça fait une requête ajax sur le fichier reserver.php qui lui passe l'identifiant de l'article et l'identifiant de l'utilisateur. Ce fichier php inscrit dans la table reservation le lien du client et de l'article.

Pour l'orienté objet, j'ai uniquement fait la database en orienté objet parce que c'est le seul élément qui se prêtait à ça parce qu'on utilise plusieurs fois la base de données. Elle a une classe database, un constructeur qui construit un objet PDO qu'on construit qu'une fois au début. Pour chaque requête SQL sur la base de données, on récupère le PDO qu'on a construit la première fois avec la fonction getPDO.