

Projet d'Application Web

Site de vente en ligne de jeux vidéo

Master 1 MIAGE (Méthodes Informatiques Appliquées à la Gestion des Entreprises)

Année 2010-2011

Enseignant : Michel BUFFA

Étudiants : Julien LESPAGNARD, Anthony BONIN

Sommaire

Introduction	3
Objectif du document	3
Finalité du projet.....	3
Contexte	3
Déroulement du projet	3
Technologies utilisées	4
Fonctionnalités implémentées	4
Génération du jeu de données	4
Inscription / Connexion	4
Parcours du catalogue / Recherche au sein du catalogue	5
Détails d'un jeu vidéo.....	5
Panier / Commande	6
Console d'administration.....	6
Gestion des utilisateurs.....	6
Gestion du catalogue	6
Gestion des commandes	7
Conclusion.....	7

Introduction

Objectif du document

Le présent document a pour but d'expliquer la démarche suivie pour la réalisation du site, mentionnant notamment les technologies utilisées et les fonctionnalités implémentées.

Finalité du projet

Le projet d'application web consiste au développement d'un site de vente en ligne de jeux vidéo. Le site doit intégrer les fonctionnalités principales suivantes :

- template de présentation au format header + footer + menu ;
- gestion des utilisateurs (clients) ;
- interface de connexion en AJAX ;
- recherche de jeux vidéo ;
- gestion d'un panier pour une commande.

Contexte

Le site de vente en ligne de jeux vidéo a été réalisé par Julien LESPAGNARD et Anthony BONIN dans le cadre d'un projet d'application web pour le second semestre du Master 1 MIAGE (Méthodes Informatiques Appliquées à la Gestion des Entreprises).

Déroulement du projet

Le développement peut se diviser en deux tâches distinctes :

- génération du jeu de données ;
- développement de l'application web.

Anthony s'est chargé de la génération du jeu de données, puis, une fois la tâche terminée, est venu prêter main forte au développement de l'application web.

Julien s'est chargé du développement de l'application web.

Ces deux tâches ont été réalisées en parallèle : il n'y avait aucun lien de dépendance entre elles.

Technologies utilisées

Côté serveur :

- Java : Entity, Session Bean et Servlet ;
- Galssfish.

Côté client :

- JSP (JavaServer Pages) ;
- JSTL (Java Standard Tag Library ;
- Javascript ;
- jQuery ;
- Dojo ;
- AJAX.

Les versions utilisées concernant ces technologies sont les dernières versions « release ».

Fonctionnalités implémentées

Génération du jeu de données

Afin de générer un jeu de données assez conséquent, nous avons fait appel à l'API d'Amazon. Cette dernière fournit un web service permettant de récupérer des produits, présents dans le site, selon des critères de choix (par exemple les jeux vidéo), avec une certaine quantité et un contenu plus ou moins détaillé.

Nous avons développé un « parser » chargé donc, de récupérer le résultat de la requête soumise à l'API d'Amazon et de le transformer sous forme d'entités que nous avons réalisées.

Concernant les utilisateurs, nous utilisons un fichier XML édité avec les données que nous avons générées à l'aide d'un générateur de données en ligne. Il nous a suffi de renseigner le nom et le type des champs voulu, de demander au générateur en ligne de nous fournir un certain nombre de données aléatoire pour ces champs et de les exporter en format XML.

Inscription / Connexion

L'inscription et la connexion se réalisent en AJAX à partir de formulaires Dojo (*dojoType=dijit.form.Form*) au sein de fenêtres de dialogue également Dojo (*dojoType=dijit.Dialog*).

L'appel AJAX se réalise à l'aide de jQuery. Ces appels sont dirigés vers des « servlets » spécifiques à chacune de ces deux fonctions. Une « callback function » est lancé au retour de l'appel de la « servlet » : un message de succès sera alors affiché en cas de réussite d'exécution de la fonction, un message d'erreur sera affiché le cas échéant : l'utilisateur pourra alors effectuer une nouvelle tentative ou annuler en fermant la fenêtre de dialogue.

Une fois l'utilisateur connecté, son profil est stocké en session.

Parcours du catalogue / Recherche au sein du catalogue

Il est possible de parcourir le catalogue de deux manières différentes :

- parcourir le catalogue spécifique à une plateforme ;
- parcourir le catalogue suite à une recherche.

Pour parcourir le catalogue selon une plateforme spécifique, il suffit de cliquer sur le lien dans le menu correspondant à la plateforme.

Autrement, il est possible d'effectuer une recherche de jeux vidéo : pendant la saisie du critère dans le champ de recherche, l'utilisateur se verra proposer des choix en corrélation avec sa saisie. Cette suggestion de recherche est réalisée à l'aide d'un plugin de jQuery : un appel AJAX se fera vers une « servlet » chargée d'émettre les suggestions de recherche, et au retour de cet appel, une fonction d'auto-complétion sera de jQuery sera appelé afin de proposer à l'utilisateur les différentes recherches qu'il peut effectuer.

Lors du parcours du catalogue, que ce soit en parcours par plateforme ou à la suite d'une recherche, si le nombre de produits affichés est supérieur à 10, une pagination se mettra en place de manière à ne pas surcharger la page d'une part, et à gagner du temps en chargeant moins de produits d'autre part. L'utilisateur pourra aisément naviguer de page en page dans les résultats à l'aide de liens-images situés au bas de la liste des produits affichés.

En outre, l'utilisateur peut ajouter dans son panier (et ce même s'il n'est pas connecté) des produits, qu'il pourra par la suite commander. Le panier est stocké en « session ».

Il est possible d'accéder aux informations détaillées d'un jeu : pour cela, il suffit à l'utilisateur de cliquer sur le lien présent sur le nom du jeu dont il souhaite voir la description détaillée.

Détails d'un jeu vidéo

Ici, des informations complémentaires sur le jeu vidéo sont disponibles.

Également, une description riche (avec des images et du style) et des « screenshots » sont accessibles. Ces éléments se trouvent au sein d'un « widget » de Dojo : les onglets (*dojoType=dijit.layout.TabContainer*).

Pour l'affichage des images, nous avons utilisés un plugin : Lightbox (actuellement en version 2.05). Ce plugin permet d'afficher des images dans une fenêtre de dialogue, et de parcourir les images au sein de cette dernière lorsqu'on clique sur une miniature.

Au sein du détail d'un jeu vidéo, l'utilisateur peut, comme lorsqu'il parcourt le catalogue, ajouter le produit à son panier.

Panier / Commande

Au sein de son panier, l'utilisateur peut supprimer des produits qu'il a sélectionnés ou en modifier la quantité voulue : les totaux se mettent alors à jour, cette mise à jour s'effectuant par un appel d'une fonction Javascript.

Un utilisateur peut effectuer une commande seulement s'il est connecté. Bien évidemment, il peut accéder à son panier, mais lorsqu'il voudra passer commande, s'il n'est pas identifié, l'interface de connexion s'affichera.

Une fois connecté, l'utilisateur pourra passer commandes : la quantité des produits commandés diminuera donc. Si le stock de produit est insuffisant, un message d'erreur en avertira l'utilisateur, l'invitant à modifier sa commande.

Console d'administration

Seul un administrateur peut accéder à la console d'administration. Son identification a été codée en dure juste pour l'exemple, il vous suffira donc de vous connecter avec les informations suivantes pour y accéder :

- Email : admin@admin.fr
- Password : admin

La console d'administration vous permettra de gérer les utilisateurs, les commandes et les produits. Pour la gestion de ces derniers, seules les fonctionnalités d'affichage, de suppression et d'ajout (il n'est pas possible d'ajouter une commande) ont été implémentées.

A droite, le menu fait appel à un « widget » de Dojo pour disposer les fonctionnalités énumérées plus haut : l'accordéon (*dojoType=dijit.layout.AccordionContainer*).

Gestion des utilisateurs

Il est possible d'afficher la liste des utilisateurs. La pagination intervient une fois de plus, de manière à ne charger que 10 utilisateurs par 10 utilisateurs.

Un lien-image à droite de chaque utilisateur (sauf l'utilisateur courant qui est l'administrateur) permet sa suppression. Elle se réalise en AJAX, la liste étant rafraîchit au retour de cet appel.

L'administrateur peut également ajouter un utilisateur : le formulaire d'inscription s'affichera alors afin de lui permettre d'ajouter un nouvel utilisateur.

Gestion du catalogue

Même fonctionnalités que pour la gestion des utilisateurs.

La liste des produits est affichée avec une pagination mise en place lorsque le nombre de produit dépasse 10.

Un lien-image à droite de chaque produit permet sa suppression. Elle se réalise en AJAX, la liste étant rafraîchit au retour de cet appel.

Il est possible à l'administrateur d'ajouter un produit : un formulaire de saisie est prévu à cet effet. Au sein de ce formulaire, il pourra ajouter dynamiquement des « screenshots » associés au jeu vidéo.

Gestion des commandes

Idem que pour la gestion des utilisateurs et des produits, mis à part qu'il n'est pas possible pour l'administrateur d'ajouter une commande.

La liste des produits est affichée avec une pagination mise en place lorsque le nombre de produit dépasse 10.

Un lien-image à droite de chaque produit permet sa suppression. Elle se réalise en AJAX, la liste étant rafraîchi au retour de cet appel.

En outre, un lien sur les valeurs des champs « Guest » et « Items » est disponible afin d'afficher, au sein d'une fenêtre de dialogue Dojo, le profil utilisateur si l'administrateur clique sur une valeur du champ « Guest », la liste des produits commandés avec la quantité commandée s'il clique sur une valeur du champ « Items ».

Conclusion

Il s'agissait là d'un projet ayant pour but de nous enseigner le développement web en Java à travers les nouvelles technologies web utilisées aujourd'hui.

Le développement a été formateur pour nous. En effet, bien plus efficace qu'un cours exhaustif sur les technologies web, la prise en main de ces dernières au sein de ce projet nous a permis d'acquérir des compétences solides dans le domaine.

Il est à noter toutefois que, malgré une documentation très complète des APIs Dojo et jQuery, accompagnée bien souvent d'exemples, ces deux technologies sont aussi riches que difficiles à appréhender. Nous nous sommes bien souvent perdus sur la toile dans nos recherches de solutions à certains problèmes que nous rencontrions. Il faut réellement étudier ces APIs de fond en comble pour arriver à un résultat professionnel : malheureusement, dans le cadre du projet, nous ne disposions que de peu de temps pour cela, et il a fallu nous débrouiller comme on pouvait, bien souvent de manière plus fastidieuse mais surtout plus accessible pour notre niveau, afin de réaliser certaines fonctionnalités.

Au final, nous avons réussi à mener à terme notre projet, en intégrant toutes les fonctionnalités de base, mais aussi certaines qui étaient facultatives.