



PROJET WEB

Groupe G

SOULEY Salif
PENG Qixiang
AFIFI Abdourahmane
JAYASHANKAR Koumaar

10 mai 2016

Sommaire

I-	Introduction	1
II-	Les choix techniques	2
1-	La base de données (par Peng)	2
2-	Authentification et traiter_authentification(par Jayashankar)	2
3-	creation_compte et traiter_creation_compte(par Jayashankar)	3
4-	Miseenpage_create et Miseenpage_login(par Peng,Jayashankar)	3
5-	Style_log(par Peng,Jayashankar)	4
6-	Create.sql & Fill.sql(par Peng)	4
7-	Choix.php et traitement.php(par Peng,Affi)	4
8-	Paielement(par Souley)	4
III-	conclusion	5

I- Introduction

Nous avons opté pour un projet libre. Le sujet de notre projet est la création d'un site de gestion de réservation de place d'une salle de théâtre.

Le but de notre projet est de créer un site internet sur lequel des clients peuvent accéder aux différentes représentations qui sont proposées dans un théâtre et de pouvoir effectuer une (des) réservation(s) sur celle-ci. Le théâtre dispose de plusieurs places et propose plusieurs représentations chaque jour. Lorsque le client est sur la page des représentations, il aura le choix de choisir :

- Un type de place : Les petites places, les places moyennes et les places V.I.P
- Une représentation parmi celles proposées dans la journée

Le tarif de chaque type de place dépend à la fois de l'âge du client (-50% pour les clients de moins de 18 ans et de plus de 65 ans, -30% pour un étudiant et tarif plein pour le reste des clients) et des différentes représentations qui sont proposées.

Les représentations sont accessibles en libre accès à tous les internautes mais lorsqu'un client est intéressé par une représentation et souhaite réserver, il doit soit s'identifier s'il possède un compte utilisateur, soit créer un compte utilisateur en entrant toutes ses informations personnelles (prénom, nom, adresse, date de naissance, mot de passe, etc.) pour pouvoir accéder à la page de réservation du site.

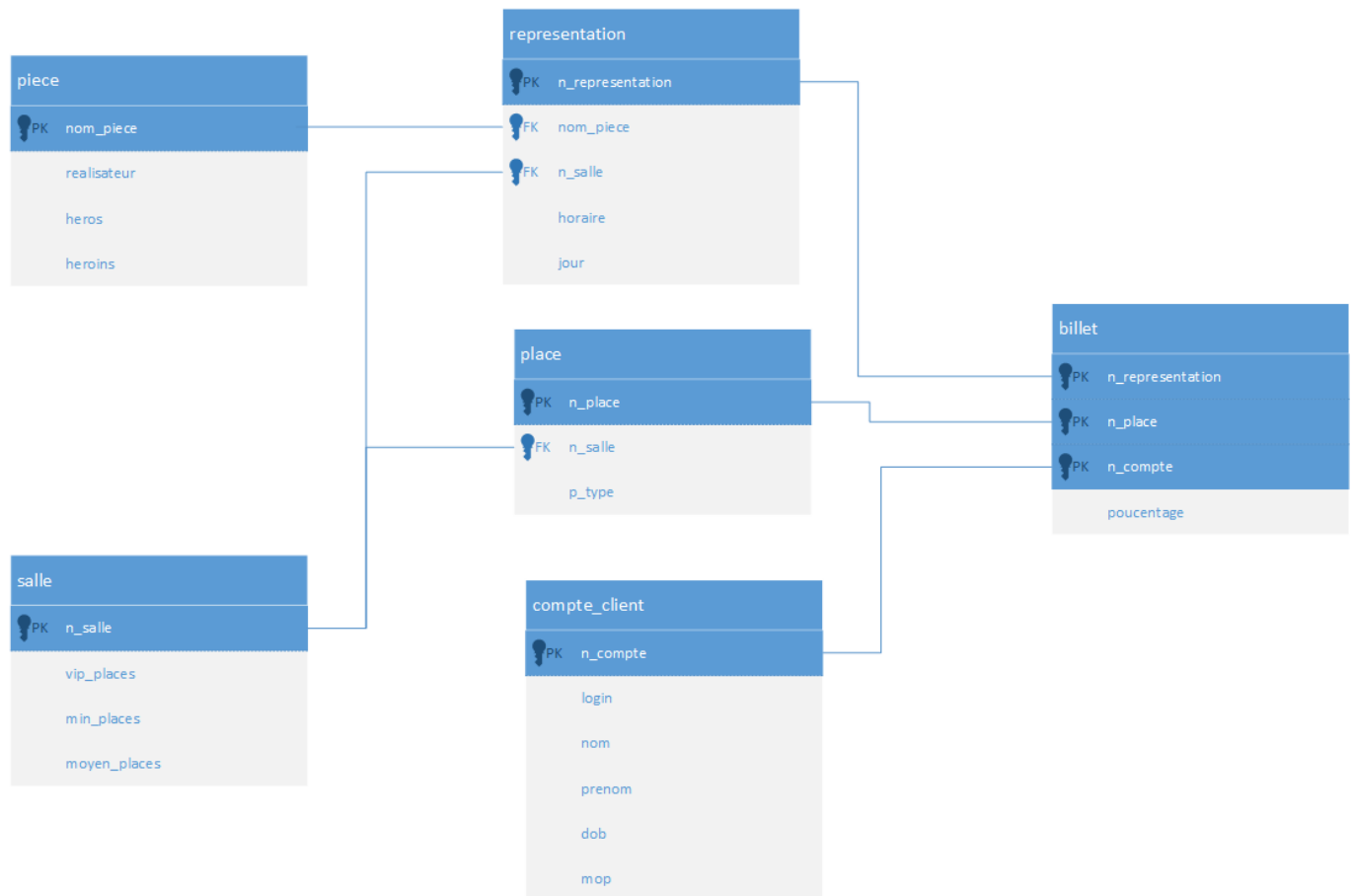
Les informations du client sont stockées dans une base de données.

Pour notre projet, nous avons décidé de programmer dans les langages suivants : **HTML5**, **CSS3**, **PHP5**, **JavaScript** et **PostgreSQL**.

II- Les choix techniques

1- La base de données (par Peng)

Nous avons représenté le schéma UML de notre base de données ci-dessous :



2- Authentification et traiter_authentification(par Jayashankar)

Pages qui permettent pour un membre déjà présent dans le site web, de se connecter à son compte.

Authentification est la partie avec le formulaire où on y trouve des cases pour écrire le login et le mot de passe.

Et puis traiter_authentification vérifie si les données reçues par l'utilisateur sont non vides ou non présent dans la table compte_client. Si c'est le cas on envoie un message d'erreur et on propose un lien vers la page authentification. Sinon on est redirigé vers le home.

Problèmes :

- Vérifier que l'utilisateur a écrit des données
- Redirection vers home

Solutions :

- Utilisez isset(si le variable existe ou non) et !empty(si la valeur de cette variable et non null ou non) pour vérifier les données
- Utilisez header pour un redirection vers une autre page

3- **creation__compte et traiter__creation__compte**(par Jayashankar)

Pages qui permet la création de compte d'un potentiel membre.

Creation__compte contient un formulaire où on peut remplir le login,le nom,le prénom, la date de naissance, le mot de passe et la confirmation de mot de passe.

traiter__creation__compte vérifie si les deux mot de passes sont pareil puis regarde si la base de données comporte déjà ce login. Si c'est le cas il renvoie un message d'erreur et propose un lien vers la page de création. Sinon Il insère ces données dans la table comte_client avec adm=0 et redirige vers le home .

Problèmes :

- Vérifier que l'utilisateur a écrit des données
- Redirection vers home

Solutions :

- Utilisez isset(si le variable existe ou non) et !empty(si la valeur de cette variable et non null ou non) pour vérifier les données
- Utilisez header pour un redirection vers une autre page

On a rencontré les mêmes problèmes dans les deux pages puisque c'est similaire.

4- **Miseenpage__create et Miseenpage__login**(par Peng,Jayashankar)

Page qui contient deux fonction entête(titre) et pied. entête(titre) écrit les début d'un page html (comme <html> <head> et ainsi de suite) avec comme title titre et un background et un référence à un stylesheet. Pied() donne les fins d'un page (comme <body><html>).

5- **Style_log**(par Peng,Jayashankar)

Page qui permet de définir les bars de navigation, et les alignements de texte.

6- **Create.sql & Fill.sql**(par Peng)

create.sql permet de créer les tables du base de données.

fill.sql permet de remplir le tableau avec les admin(adm=1).

7- **Choix.php et traitement.php**(par Peng,Affi)

La page des choix est une page qui permet à l'utilisateur qui navigue, de choisir directement la représentation qu'il souhaite voir, avec le choix de l'horaire et du jour.

Ce choix est obligatoirement réduit à 2 et est directement reliée à la base de données pour gagner du temps et économiser le temps de calcul au serveur. Un formulaire de type post, le type le plus sûr, associé à une action gérée par traitement.php a été utilisé.

Ce formulaire comporte plusieurs types de boutons qui sont reliés à la base de données grâce aux fonctions postgresql.

Cette page est associée à un script JS de validation et le champ de texte est souligné en rouge si certaines conditions liées aux caractéristiques de la BD (taille du texte), ne sont pas respectées. Si c'est le cas on peut préventivement stopper l'envoi.

Le problème principal a été d'implémenter un script de validation JavaScript.

La solution adéquate a été de se servir des notions d'objet de la programmation web et de modifier le style via une instruction CSS

Un problème a été la sécurité des données qui passent à travers un traitement, elles peuvent être exploitées par un utilisateur malveillant (faille XSS), on a remédié à ce problème grâce à la fonction htmlspecialchars qui affiche le code HTML produit, au lieu de l'interpréter.

8- **Paielement**(par Souley)

La page de paiement permet à l'utilisateur, après avoir choisi la représentation qui lui convient, de pouvoir payer en ligne son ticket de théâtre.

L'utilisateur doit saisir son numéro de carte bleu (à 16 chiffres), la date d'expiration ainsi que le cryptogramme (3 chiffres). Si ces informations relatives aux coordonnées bancaires ne sont pas respectées, l'utilisateur est informé que les informations qu'il a saisies ne sont pas bonnes. S'il les respecte, sa demande de paiement est traitée et est autorisée si sa provision est suffisante. L'utilisateur est donc redirigé vers la page d'accueil.

Ce mode de paiement est fictif. On ne vérifie pas pour de vrai si ces coordonnées bancaires existent et si son compte est approvisionné.

III- conclusion

Ce projet est fonctionnel. Cependant, on peut toujours apporter des améliorations.