Description du contenu

Forum de l'Art

Le projet :

Site de partage d’œuvres d'art.
Un artiste (membre promu par l'administrateur) peut mettre en ligne des oeuvres, ou modifier/supprimer les oeuvres qu'il a déjà publiées.
Une utilisateur non-connecté peut se connecter ou créer un compte et peut voir les oeuvres et comptes des membres.
Un utilisateur connecté peut commenter une oeuvre et éditer ou supprimer l'un de ses précédents commentaires.
L'administrateur peut éditer / supprimer n'importe quel commentaire.
Un utilisateur connecté peut également noter une oeuvre (de 0 à 10) une seule fois, si il n'est pas l'artiste qui a publié cette oeuvre.
Un utilisateur connecté a la possibilité de suivre un artiste (reçoit un mail lorsque l'artiste publie une oeuvre).
L'administrateur peut modifier le niveau de privilège d'un compte utilisateur à admin ou membre.
L'administrateur peut supprimer un compte.

Projet web - NFA084 / NFA021 - Juin 2015

Thomas Swank

Description des données :

(tables dans traitement/init.php)

Table user :

Un utilisateur est décrit par son login [clé primaire], son mot de passe (stocké en sha256), son nom, son prénom, son adresse mail, la date de création du compte et des informations le concernant.
4 niveaux d'utilisateurs : Admin, Artiste, Membre, Invité (non-connecté)

- Invité : Voir les œuvres, notes/commentaires et profils des membres. Peuvent créer ou se connecter à un compte.
- Membres : Invité + Possibilité d'ajouter/modifier/supprimer des commentaires., ajouter une note à une oeuvre, et suivre un artiste (reçoit un mail lorsque celui-ci publie une oeuvre).
- Artistes : Membre + peut ajouter/modifier/supprimer des oeuvres.
- Admin : Accès à toutes les informations et possibilité de modification de privilège (l'utilisateur concerné reçoit un mail pour l'en informer) ou suppression de compte et édition/suppression de commentaires.

Table sessions :

Une session est décrite par un id [clé primaire], un login [ref. à user(login)] et une date (mise à jour automatiquement on update).
Connexion :

Lorsqu'un utilisateur se connecte, on crée une entrée avec son login et un id générée aléatoirement avec :
hash('sha256', openssl_random_pseudo_bytes(32))
Côté PHP, le login et l'id(sid) sont stockés dans la variable $_SESSION. Le site impose l'utilisation de cookies.
Vérification :

À chaque nouvelle page ou pour chaque traitement, le couple login/id stocké dans la table sont comparés au couple login/sid de la variable $_SESSION pour authentifier l'utilisateur.
Si l'utilisateur est authentifié, on vérifie que la période d'expiration n'est pas dépassée (1 heure) et on génère un nouveau sid, modifié dans $_SESSION et dans la table.
Déconnexion :

Lorsque le délai de session est dépassé ou qu'un utilisateur se déconnecte, la ligne correspondante est supprimée de la table, le cookie de session est effacé et les variables $_SESSION sont effacées.

Table categories :
Une catégorie est décrite par un id [clé primaire], un nom et un parent [ref. categories(id)].
Les catégories sont hiérarchisées sur 2 niveaux : Les catégories parents ou catégories générales et les catégories enfants ou catégories spécifiques.
Cette structure hiérarchique permet notamment de sélectionner plusieurs catégories spécifiques à partir d'une catégorie générale (cf. pages/listeoeuvres.php)

Table oeuvres :
Une oeuvre est décrite par un id [clé primaire], un artiste [ref. user(login)], une catégorie [ref. categories(id)], un titre, une date d'ajout et une description.
Une oeuvre est liée à 1 à 3 images.

Table images :
Une image est décrite par un id [clé primaire], une oeuvre [ref. oeuvres(id)], un fichier (img grande taille), une miniature, un répertoire et une position.
Les fichiers images et miniatures sont stockées dans les répertoires /images/imgX (X = entier).
Un répertoire contient maximum 50 oeuvres (100 fichiers .jpeg et un fichier index.php de renvoi à pages/index.php).
Lorsqu'il n'y a plus de place dans les répertoires existants : création d'un nouveau répertoire(imgX+1) avec index.php généré automatiquement.
fichier : nom du fichier image, de la forme imgX.jpeg (X = entier)
miniature: nom du fichier miniature, de la forme minX.jpeg (X = entier)
position : position de l'image par rapport aux autres associées à la mm oeuvre (0 à 2).
0 est l'image principale, c'est elle qui est utilisée comme image de présentation de l'oeuvre.

Table commentaires :
Un commentaire est décrit par son id [clé primaire], contenu, auteur [ref. user(login)], oeuvre [ref. oeuvres(id)] et date de publication.
Tous les utilisateurs peuvent voir les commentaires d'une oeuvre.
Un membre connecté à la possibilité de commenter une oeuvre, et d'éditer ou de supprimer l'un de ses propres commentaires.
L'administrateur peut éditer/supprimer n'importe quel commentaire.

Table notes :
Une note est décrite par un auteur [ref. user(login)], une oeuvre [ref oeuvres(id)] et une valeur (0 à 10) [auteur et oeuvre son clé primaire].
Un utilisateur connecté à la possibilité de noter une seule fois une oeuvre.
Un artiste ne peut pas noter ses propres oeuvres.
Une note ne peut être modifiée ou supprimée, à part si l'oeuvre ou l'auteur est supprimé.

Table suivi :
Un suivi est décrit par un id [clé primaire], un artiste [ref. user(login)] et un membre [ref. user(login)].
Un utilisateur connecté à la possibilité de suivre un artiste ou d'annuler un suivi.
Lorsqu'un artiste met en ligne une oeuvre, tous les membres qui le suivent reçoivent un mail les informant de la publication.

N.B : Il y a quatre utilisateurs différents pour la base de donnée correspondants à des niveaux de privilèges spécifiques :
- admin
- artiste
- membre
- invité : utilisateur non-connecté
- check_user : utilisé pour gestion de sessions et login, logout et création de compte.

Fonctionnalités :

(répertoire pages)

Header, En-tête et pied de page

Chargés dans toutes les pages : head.php, menuHaut.php et footer.php.

Se connecter / Se déconnecter

(dans menuHaut.php)
Si utilisateur connecté : liens Se connecter et créer compte
Sinon : liens pour consulter son compte et se déconnecter.
Lorsqu'un utilisateur soumet le formulaire de connexion : vérification de ses identifiants, et si ils sont validés, création d'une session et renvoi sur la page.
Lorsqu'un utilisateur se déconnecte : destruction de la session (dans la table+$_SESSION+cookie) et renvoi sur la page.
Les sessions sont vérifiées et mise à jour à chaque changement de page.

Messages d'erreur de confirmation

(dans menuHaut.php)
Affiche un message indiquant une erreur ou la confirmation de succès d'un traitement. Le message disparaît après 3 secondes (cf. js/login.js)

Message concernant les cookies

(login.js)
Si l'utilisateur n'accepte pas les cookie : affichage d'un message l'invitant à modifier les paramètres de son navigateur.

Accueil

(index.php)
Phrase de présentation du site.
Affiche les 5 oeuvres :
- les plus récentes
- les mieux notées
- les plus commentées

En cliquant sur la vignette d'une oeuvre, un utilisateur est renvoyé vers la page de visualisation correspondante

Afficher un compte
(compte.php)
Affiche les informations / commentaires / lien vers oeuvres (si artiste) d'un utilisateur.
Si un utilisateur consulte son propre compte, il peut en plus consulter les artistes qu'il suit, ou éditer ses informations ou son mot de passe.

Créer un compte
(creerCompte.php)
Formulaire de création de compte : Vérification en ajax que le login est dispo. Mail obligatoire.
En cas de succès de création, l'utilisateur est connecté automatiquement (création de session) et renvoyé à la page précédemment consulté.

Liste des oeuvres
(listeOeuvres.php)
Liste de toutes les oeuvres du site (12 par pages) avec possibilité de filtre par artiste et/ou catégorie.
Les résultats peuvent être trié par date, note, commentaires, titre ou artiste.
En cliquant sur une vignette d'oeuvre, renvoi vers page de visualisation.

Voir une Oeuvre
(voirOeuvre.php)
Page de visualisation d'une oeuvre.
Une partie avec les informations concernant l'oeuvre : titre, date d'ajout, catégorie, artiste (avec lien vers sa page de compte) et description.
Une partie images : 3 tailles d'images (plein écran, principal, miniatures)
Bloc miniature : Si plus d'une image pour cette oeuvre. Au clic, remplace l'image principale par celle correspondant à la miniature.
Bloc principal : Au clic affiche l'image en plein écran.
Bloc plein écran : Possibilité de voir images suivante/précédente et de fermer le bloc.
Une partie notes : Affiche la note moyenne et le nombre de notes. Si utilisateur connecté qui n'a pas encore voté l'oeuvre et qui n'est pas l'artiste correspondant, affichage d'un formulaire pour enregistrer une nouvelle note.
Une partie commentaires : Affiche tous les commentaires déjà publiés et un formulaire d'ajout d'un nouveau commentaire. L'auteur d'un commentaire ou l'administrateur ont la possibilité d'éditer / supprimer un commentaire.
Pour chaque commentaire, lien sur le nom de l'auteur vers sa page de compte.

Liste des artistes
(artistes.php)
Affiche le login de tous les artistes (max 30 par page) avec liens vers le compte correspondant.

Gestion des utlisateurs
(gererUser.php)
Accessible uniquement à l’administrateur.
Permet de voir la liste de tous les utilisateur.
Pour chacun, possibilité de voir le compte.php correspondant, de modifier son niveau de privilège (envoi automatique de mail pour avertir l'utilisateur si promotion à artiste), ou de supprimer le compte.

Gestion des oeuvres
(gererOeuvres.php)
Accessible uniquement aux artistes.
Lien pour ajouter une nouvelle oeuvre.
Affiche la liste des oeuvres qu'il a publié avec pour chacune :
Un lien vers la page de visualisation
Un lien vers formulaire de modification.
Un bouton de suppression.

Ajout d'oeuvre
(addOeuvre.php)
Formulaire d'ajout d'une nouvelle oeuvre.
L'artiste doit associer au minimum une image à l'oeuvre et au maximum 3.
Les images sont redimensionnées en JS et avec la balise canvas pour réduire la taille de l'envoi $_POST et la charge au niveau du serveur -> pas de multipart/form-data

Modifier une oeuvre
(modifOeuvre.php)
Affiche un formulaire avec les informations et images correspondantes à l'oeuvre.
L'artiste a la possibilité de modifier toutes les informations et les images (sauf date d'ajout).

A propos
(apropos.php)
Informations sur le développement du site

N.B : Toutes les pages appellent ../traitement/connect.php :
- Vérifie la session de l'utilisateur.
- Le déconnecte si la session est expirée.
- Met à jour la session si utilisateur connecté
- Établie la connexion à la base de donnée avec niveau de privilèges correspondant

Description des pages de traitement :
(répertoire traitement)

info.php
Variables pour connexion à la base de données.

init.php
Création de la base de donnée, des tables et des utilisateurs.
Ajout des éléments d'exemple à la base donnée.

functions.php
Fonctions générales :
- displayErreur : Renvoi vers page avec notification d'une erreur
- displayMessage : Renvoi vers page avec notification d'un traitement réussi
- saveCurrentUrl : Récupération de l'url de la page actuelle (en enlevant certaines variables $_GET indésirables : erreur et message
- checkSession : Vérifie les variables login et sid stockées en $_SESSION dans la tables sessions. Renvoi le niveau de privilège de l'utilisateur et la date de dernière mise à jour de la session.
- createSession : Crée une session lorsqu'un utilisateur se connecte ou crée un compte
- deleteSession : Détruit une session lorsqu'un utilisateur se déconnecte ou que sa session a expirée
- verifPass : Vérifie la conformité login/mot de passe dans la table user, lorsqu'un utilisateur se connecte ou modifie son mot de passe.
- sendMail : Envoi d'un mail pour suivi artiste ou notifier un utilisateur d'un changement de privilège à artiste.

connect.php
Vérifie si l'utilisateur est connecté.
Établie une connexion avec la base de donnée avec le niveau de privilège correspondant à l'utilisateur.
Si utilisateur connecté :
- Vérifie que la session n'est pas expirée, sinon la détruit
- Met à jour la session si tout va bien

logInOut.php
Traitement du formulaire de connexion et de déconnexion.

addUser.php
Page de traitement du formulaire de création de compte.
Gère également la vérification de disponibilité du login en ajax.

editCompte.php
Gestion des requêtes :
- Modification informations de compte (par l'utilisateur concerné)
- Modification mot de passe (par l'utilisateur concerné)
- Modification du niveau de privilège (par l'admin)
- Suppression de compte (par l'admin)

Si suppression d'un membre ayant mis en ligne des oeuvres, suppressions au préalable des images correspondant aux oeuvres.

oeuvres.php
Traite les requêtes d'ajout, modification, suppression d'oeuvre.

images.php
Traite les requêtes d'ajout et de suppression d'images au niveau de la BDD ainsi que des fichiers correspondant.

commentaires.php
Traite les requêtes d'ajout/suppression/modification de commentaires.

noter.php
Traite les requêtes d'ajout de note.

suivi.php
Traite la mise en place et la suppresion du suivi d'un artiste par un autre utilisateur.

N.B : Lorsque toutes les données devant être insérées dans la base de données ne peuvent pas être vérifiées, l'insertion/modification de la table correspondante se fait avec une requête préparée afin de ne pas risquer d'injection SQL.
De plus, ces données sont échappées avec :
htmlspecialchars($variable, ENT_QUOTES, "UTF-8")
pour éviter que certains caractères soient interprétés en html.

Perspectives d'amélioration :

Administrateur :
- Possibilité d'édition/suppression d'oeuvres et de modification des informations d'un compte.
- Possibilité d'ajout/modification/suppression de catégories
- Possibilité d'empêcher l'accès à certaines fonctionnalités (ex. commentaires) pour certains utilisateurs -> création d'un niveau d'utilisateur supplémentaire.

Artiste :
- Augmenter le nombre d'images maximum par oeuvre.
- Changement de position des images d'une oeuvre sans avoir à les supprimer et rajouter dans le bon ordre.

Membre :
- Possibilité de supprimer le compte.
- Possibilité d'empêcher l'accès des autres utilisateurs à certaines informations.

Utilisateurs non-connectés :
- Réduire les informations visibles des membres dans la page compte

Général :
- Admettre la transmission des sessions par URL avec choix pour chaque membre.
- Vérification de l'adresse mail de chaque membre par envoi d'un mail de validation lors de la création d'un compte.
- Système de messagerie interne pour les membres.
- Ajout d'un forum de discussions.
- Ajout d'un système de partage d'une oeuvre sur des réseaux sociaux.

Sécurité :
- Sécuriser la transmission des formulaires par ajout d'un table avec adresse IP utilisateur / numéro du formulaire / date d'expiration et vérification lors du traitement.
- Limiter le nombre de tentatives de connexion par IP/session
- Protéger la création de compte par CAPTCHA

Bugs connus :
- Les majuscules/minuscules ne sont pas prises en compte lorsqu'un utilisateur se connecte.