

[Projet de développement web]

Enoncé du projet

Gestion de stages d'étudiants

1 Enoncé

Développer un site web en HTML/CSS/JavaScript/PHP permettant de gérer les étudiants en stage.

1.1 Connexion d'un utilisateur

Les utilisateurs seront les étudiants eux-mêmes.

1.1.1 La connexion

Pour se connecter, il faudra saisir son adresse email et son mot de passe, puis cliquer sur le bouton "CONNECTION".

1.1.2 Création d'un compte

Formulaire

Si un étudiant n'est pas encore inscrit sur le site, il pourra alors cliquer sur le bouton "CREATION D'UN COMPTE". Ce bouton fera apparaître un autre formulaire, sous le formulaire de connexion, contenant différentes zones de saisie qu'il devra remplir et un bouton pour effectuer la sauvegarde dans la DB. Les zones de saisie sont les suivantes:

- l'adresse email de l'étudiant
- le mot de passe
- le mot de passe une 2ème fois
- le nom de l'étudiant
- le prénom de l'étudiant

Validations

Pour qu'un nouveau compte soit créé, il faut:

- vérifier que l'adresse email est valide (elle doit contenir au moins une lettre, puis des lettres ou le caractère ".", puis au moins une lettre, puis le caractère "@", puis au moins une lettre, puis un ".", puis au moins une lettre¹).
- le mot de passe fasse au moins 4 caractères
- les 2 mots de passe sont équivalents
- le nom de l'étudiant et le prénom fassent au moins 2 caractères

¹ Normalement, d'autres contraintes sont nécessaires pour la vérification d'une adresse email. Seules celles-ci doivent être vérifiées pour le projet.

1.2 Ecran d'accueil

Sur cet écran, on trouvera:

- un message de bienvenue "Bienvenue XXX YYY" (XXX représentant le nom de l'étudiant et YYY son prénom)
- plusieurs boutons permettront d'accéder aux fonctionnalités suivantes:
 - informations sur les stages choisis (ce bouton n'apparaîtra que si l'utilisateur a déjà choisi un stage)
 - liste des étudiants
 - liste des adresses des entreprises
 - encoder l'adresse d'une entreprise
 - se deconnecter

1.3 Afficher les informations des stages de l'étudiant connecté

Cette page affichera les informations des stages de l'étudiant.

1.4 Liste des étudiants

La liste des étudiants doit être affichée, triée par ordre alphabétique. Pour les étudiants possédant déjà un (ou plusieurs) stage(s), un bouton supplémentaire apparaîtra à droite des infos de l'étudiant.

Une case à cocher permettra d'afficher uniquement les étudiants ne possédant pas encore de stage²

1.5 Afficher les informations des stages d'un étudiant

En cliquant sur le bouton situé à droite des infos d'un étudiant, une nouvelle page doit se charger et afficher les informations des stages de l'étudiant. Sur cette page, un bouton permettra de recharger l'écran de liste des étudiants.

² Les étudiants possédant un stage devront être masqués. Conseil : utiliser les attributs "class".

1.6 Encoder une adresse de lieu de stage

Formulaire

Les utilisateurs peuvent encoder l'adresse d'un lieu de stage en complétant les champs suivants:

- le nom de l'entreprise
- la ville de l'entreprise

Un bouton permettra de sauvegarder les informations, puis rechargera une page avec un formulaire vierge.

Validations

Le nom de l'entreprise doit faire 2 lettres minimum. La ville doit faire au minimum 3 lettres.

Contraintes

Charger les villes dans une liste de choix déroulantes. La liste des villes sera stockée "en dur" dans un tableau dans le code JS³.

1.7 Liste des adresses de lieu de stage

Sur cet écran, on doit trouver la liste des adresses de lieu de stage triée par défaut par nom des entreprises. Sur chaque ligne on trouvera :

- le nom de l'entreprise
- l'adresse
- un bouton permettant de signaler qu'on choisit cette entreprise pour y faire son stage

On y trouve 2 boutons supplémentaires; l'un permettant de trier par entreprise, l'autre permettant de trier par ville⁴.

1.8 Encoder les informations d'un stage

En cliquant sur le bouton signalant qu'on choisit une entreprise pour y effectuer son stage, un formulaire devra être généré sur le même écran, sous le lieu de stage sélectionné. Ce formulaire demandera de saisir :

- le nom du superviseur
- un descriptif du sujet

Un bouton permettra de sauvegarder les informations⁵, puis recharger la page de liste des adresses de lieu de stage.

3 Il est évident que ce n'est pas la meilleure méthode pour gérer la liste des villes. C'est cependant une contrainte du projet.

4 Une idée pourrait être de charger les différentes informations dans des tableaux en JS. Il y a moyen de les créer en manipulant "selectorQueryAll()" pour repérer un ensemble d'éléments du DOM, puis d'envoyer les valeurs de ces éléments dans 1 ou plusieurs tableaux. On pourra alors trier tableaux selon le bouton sur lequel on aura cliqué.

5 Il faudra également signaler sur l'étudiant connecté à trouvé un stage.

2 Base de données

Une base de données MySQL devra être créée. Cette base de données contiendra 3 tables.

2.1 Table des étudiants

Cette table contiendra la liste des étudiants inscrits. Ce sont donc les utilisateurs du site web. La table devra contenir:

- une clé primaire : numérotation automatique
- l'adresse email de l'étudiant
- le mot de passe
- le mot de passe une 2ème fois
- le nom de l'étudiant
- le prénom de l'étudiant
- un champ (booléen) permettant de savoir si l'étudiant a déjà trouvé un stage ou pas

2.2 Table des adresses des entreprises

Cette table contiendra les adresses d'entreprises où il est possible de réaliser son stage. Elle contiendra:

- une clé primaire : numérotation automatique
- le nom de l'entreprise
- la ville de l'entreprise

2.3 Table des sujets de stage

Cette table contiendra les sujets des stages. Elle devra être "liée" à un lieu de stage et à un étudiant. Elle contiendra :

- une clé primaire : numérotation automatique
- le descriptif du sujet du stage
- le nom du superviseur
- l'id du stagiaire (foreign key)
- l'id représentant l'adresse de l'entreprise (foreign key)

3 Remarques et contraintes

- La plupart des pages du site web ne peuvent être accessibles que si on s'est loggué précédemment. Si l'utilisateur tente d'accéder à une telle page sans être loggé, il faut le rediriger automatiquement vers la page de connexion.
- Chaque écran doit posséder un bouton "ACCUEIL" qui permettra de revenir à l'accueil et un bouton "DECONNECTION" qui permettra de vous déconnecter⁶.
- Une fonction générique de validation de la taille d'un champ peut être écrite en JS et en PHP.
- Dans la DB, choisissez les types les plus adaptés pour vos champs. Choisissez quels champs doivent avoir une valeur "unique" et quels champs peuvent être indexés.
- Gérer un maximum d'erreurs possibles⁷
- Il faut gérer le cas où le JavaScript serait désactivé sur le browser : il faut afficher un message à l'utilisateur lui signalant d'activer JavaScript
- Vous pouvez réutiliser la librairie "common.js" distribuée au cours
- Le projet ne doit pas être développé en AJAX

4 Remise du projet

Le projet devra être remis par mail :

- à l'adresse denis.paimparet@hainaut-promsoc.be
- au plus tard le lundi 14 juin 2021 à midi
- avec le sujet suivant: *[2INX - Projet de développement web] Remise du projet*
- sous forme d'un fichier compressé (.zip, .7z ou .rar) contenant :
 - les fichiers HTML, CSS, JS et PHP
 - un fichier texte contenant du code SQL permettant de récréer la base de données⁸
 - un fichier "A_LIRE.txt" dans lequel seront listés les noms des 2 étudiants ayant participé au projet, ainsi que des remarques particulières concernant le projet

⁶ Vous pouvez, si vous le voulez, gérer une barre de menu permettant de naviguer entre les différentes fonctionnalités

⁷ Vous pouvez manipuler les exceptions

⁸ Dans MySQL, il y a possibilité d'exporter une base de données (structures et valeurs) dans un fichier texte.