Projet Javascript

Tous le code doit être en anglais : les variables, routes, views, … .

3 types de user : anonyme ; membre ; administrateur

Les différentes pages :

1. Home page
   * Afficher les 20 questions récemment publié,
   * Un menu déroulant où le user fait une recherche via une catégorie, elle est unique et choisit par nos soins et sera sauvegardé dans la database, et liste toutes les questions de cette catégorie, sinon par défaut ce sont les 20 questions récentes qui apparaissent,
   * Bouton de recherche + texte du titre, faire apparaitre les questions résolu et non résolu à la place de la liste des questions déjà présentes, (un statut devra apparaitre pour les questions résolu à côté ),
   * Le user peut sauvegarder le lien via bookmark et revenir sur le résultat de sa recherche à tout moment.
2. Page des questions
   * Affiche les titres, sujets, catégories , les dates des questions et un bouton de signalement (visible uniquement par les membres et le owner de la question),
   * Chaque réponse est trié chronologiquement avec le sujet et la date de la réponse,
   * Si le owner de la question attribue une bonne réponse, elle sera affiché en tête de liste et aura une marque distinctive, (bouton sera visible pour le owner seulement) sinon chaque réponse aura à côté un bouton si c’est la bonne, visible uniquement pour le owner,
   * En dessous des réponses d’un sujet non résolu, les membres peuvent introduire une réponse dans un champs dédié.
3. Création d’une nouvelle question
   * Formulaire : titre, sujet, catégorie (pas accessible aux visiteurs anonymes),
   * Quand le membre submit la question, il est automatiquement redirigé vers celle-ci.
4. Devenir membre
   * Formulaire : prénom, nom, email, mot de passe (email doit être de De Vinci) si le formulaire n’est pas bien complété, l’email non conforme, ou l’email déjà existant une erreur sera affiché,
   * Si l’enregistrement a été fait alors il sera redirigé vers la home page connecté.
5. Se connecter
   * Champs email et mot de passe pour se connecter, si le mot de passe n’est pas le bon ou mail inexistant afficher une erreur.
6. Page utilisateur
   * Pas accessible pour les anonymes,
   * Afficher 2 listes, une pour les questions ouvertes des utilisateurs, une autre pour les questions résolus,
     + Chaque question montre son titre, si on appuie dessus on y est redirigé
7. Page administrateur
   * Accessible uniquement par l’administrateur,
   * Afficher 2 listes, une pour des questions reportés, l’autre pour les réponses reportés,
     + Chacune d’elle porte 2 boutons, supprimer et garder.
8. Navigation
   * À gauche, le logo ou le nom du site qui redirige vers la home page,
   * À droite, un menu différent pour chaque type d’utilisateur :
     + Les anonymes verront un lien pour se connecter et pour s’enregistrer,
     + Les membres verront un lien pour créer une nouvelle question, un lien pour la page d’utilisateur, et un bouton pour se déconnecter,
     + Les administrateurs verront la même chose que les membres et en plus de ça, un bouton vers la page admin.

Les requis

Responsive design

It is likely for the targeted users to browse the web site on their mobile devices. It is **preferable** to implement a *responsive design1* for ClassNotFound in this version.

Configuration

An example of an config.js file:

module.exports = {

secret: “my special secret”,

dbPath: “./data/project.db”,

...

}

Sécurité et privilèges

* S’assurer que les anonymes ne modifient et ne reportent rien du tout,
* Les membres ont les droits de modifier
* Les administrateurs ont tous les droits.

Base de donnée

* Doit être en SQLite,
* Create and maintain an SQL script to (re)create the database,
* Create and maintain an SQL script for seed data, which is the data to feed into the web site during the installation process for the first time. For instance, the admin account, pre-defined categories...etc…

Développement

Vivement recommander d’utiliser Bootstrap qui adapte automatiquement la page selon l’appareil utilisé pour ouvrir le site.

Soumission 1

app.diagrams.net

Exporter les prototypes en PDF ! jusqu’au 25 mars à 23 :55 !

Soumission 2

Fournir le design de la database, aussi SQL scripts et la remplir avec quelque data.

Utiliser le site ci-dessus pour dessiner le DSD.

Jusqu’au 1 avril à 23 :55

Soumission 3

Les professeurs vont voir nos codes, pour la partie A , les 20 questions récentes du home page,

Pour la partie B, afficher une question

Envoyer en format zip avant le 22 avril à 23 :55

Soumission final

Mettre tous le code dans un dossier nommer : projet\_nomDeFamille1\_nomDeFamille2

Puis le zipper, renommer le zip également de la même manière

À remettre avant le 11 mai à 23 :55

Soumission du rapport

En français ou en anglais, le nom du fichier : rapport\_nomDeFamille1\_nomDeFamille2

Envoyer en PDF sur moodle

À remettre avant le 13 mai à 23 :55