

PROJET PWEB 2019

Le site QCM_Live de questions en ligne

*Projet à faire par équipe de 4 étudiants max, de préférence organisé en 2 sous équipes
Pré-soutenance sur machine et pré-rapport remis durant la dernière semaine de TP.
Travail effectif en 4 semaines.*

Rendu final (rapport et site) début novembre 2019 dans le PUIITS de PWEB à l'IUT, sous la forme d'un zip de nom :

*KF_QCM_2019_vosgroupes_vosnoms.zip ou
JMI_QCM_2019_vosgroupes_vosnoms.zip
suivant l'initial de votre chargé de TPs.*

1. Présentation du projet

Le but est de réaliser un site permettant à un professeur de poser des questions **à la volée** au cours d'un amphi et aux étudiants présents de répondre aux questions posées. Le rôle du professeur sera d'animer le traitement des questions et d'analyser les résultats aux questions. Un tel site permet la réalisation d'amphis inversés, où les connaissances d'un étudiant ayant appris des concepts sont testées au cours des amphis.

Le principe est le suivant,

- les **questions** sont définis sur un **thème** donné, sous forme de **QCM à choix simple ou choix multiple**.
- des **sessions de tests** regroupent un sous ensemble des questions, basiquement définies sur un thème, voire une liste de thèmes.
- le **professeur connecté** peut lancer une **session de questions pour un groupe d'étudiants**.
- Lors d'un **mode interactif élémentaire**, le professeur connecté sélectionnera une des questions de la session sélectionnée et chaque étudiant connecté pourra y répondre. Il aura la liberté de bloquer la prise en compte des réponses, voire de passer à une autre des questions de la session de test. Dans un mode évolué, le professeur pourra sélectionner plusieurs questions et les proposer globalement aux étudiants.
- L'**analyse comptable des bonnes et mauvaises réponses** permettra de situer la qualité globale des réponses pour toute question traitée.
- Le **nombre et la liste des étudiants connectés n'ayant pas (encore) répondu** à une question est une information utile pour la conduite de l'amphi par le professeur.
- Tout étudiant pourra **juger la qualité de ses propres réponses en rapport aux résultats attendus aux questions traitées**.

2. Spécifications fonctionnelles

Le site est accessible via une page d'accueil permettant à un utilisateur inscrit de se connecter. L'inscription d'un utilisateur lui donne accès à un écran dédié suivant son rôle, professeur ou étudiant.

On pourra prévoir différents écrans pour implémenter les différents services de l'application, en se référant par exemple au document powerpoint fourni exhibant un scénario possible de session en amphi.

Architecture du code PHP

L'application devra être développée selon une architecture de type **MVC** vue en cours. Vous pourrez exploiter la série de sites démonstratifs fournis (les démos).

Vous chercherez à minimiser le nombre de templates différents pour réaliser vos écrans.

3. Développement

Sécurité des données entrantes

Des contrôles de saisie seront développés « coté serveur » en PHP, ne permettant l'accès au service qu'à une personne connectée. Le mot de passe saisi ne devra pas être visible en clair et pourra être contrôlée par un format sous forme d'expression régulière.

La fonction **preg_match de PHP** pourra alors vous être utile pour vérifier si le mot de passe correspond au format attendu.

Base de données

Une base de données vous est fournie.

Cependant, après analyse de votre projet, vous pourrez éventuellement apporter des modifications (avec l'accord de votre chargé de TP).

La base de données comprend aussi dans certaines des tables des **booléens de travail** utiles pour le travail synchronisé entre professeur et étudiants : bConnect permet de savoir si un étudiant ou un professeur est connecté ; bActif indique si un test donné est en cours ; bAutorise indique si une question dans un test est affichable sur le poste de l'étudiant ; bBloque permet de bloquer la possibilité de répondre à une question.

Charte graphique et design de l'application WEB

Vous êtes libre de choisir le design de votre application, à condition de respecter quelques contraintes d'importance :

- Apporter un grand soin à cette interface graphique, pour qu'elle s'approche d'une application professionnelle, notamment HTML5 et CSS3, exploitation de bootstrap.
- Aucun attribut de style ne devra être intégré directement dans le HTML. Veiller à respecter et à utiliser les standards d'écriture des spécifications HTML .
- Développer des gabarits de présentation en modules et les inclure suivant les besoins.

Exemples de modules : layout.tpl, header.tpl, footer.tpl, menu.tpl

Pour l'écriture et l'utilisation des template sur la base d'un layout (à savoir un canevas générique de présentation HTML pour l'application), vous pouvez vous inspirer des codes de démonstrations déjà fournis pour mvc : `Demos_mvc`

Fonctionnalités minimales attendues

C'est une application MVC aux fonctionnalités suivantes :

- côté professeur :

- >>> la connexion ou deconnexion du professeur
- >>> la sélection par le professeur connecté d'une session de questions à choix simple et l'enregistrement dans la base que la session est en cours de traitement
- >>> l'affichage par le professeur de la liste des questions de la session et la mise en évidence éventuelle dans la liste des questions déjà traitées
- >>> la sélection par le professeur d'une question non traitée parmi celles non traitées de la session en cours.
- >>> l'enregistrement dans la base que la question est en cours de traitement.
- >>> l'affichage de la liste des étudiants avec leurs résultats pour la session en cours.

- côté étudiant :

- >>> la connexion ou la déconnexion de l'étudiant,
- >>> l'affichage de la question à traiter par l'étudiant connecté
- >>> l'enregistrement de sa réponse dans la base.
- >>> l'affichage de la liste des questions déjà traitées pour la session en cours par l'étudiant avec mise en évidence des réponses de l'étudiant et des réponses attendues.
- >>> l'affichage des résultats de l'étudiant pour la session en cours.

Parmi les fonctionnalités avancées, les suivantes seront très positivement envisagées :

- >>> généralisation de l'application (comme la spécification de session de test à plusieurs thèmes et à plusieurs groupes par étudiants, le traitement de plusieurs questions en simultanée, distribution aléatoire d'un ensemble de questions sur le groupe d'étudiants, le choix automatisé de l'ordre de traitement des questions, la gestion de temporisation pour le temps maximum des réponses).
- >>> export des résultats de la session traitée dans des fichiers excel ou csv (text).

Pour gérer directement excel,

il existe des extensions .dll facilement exploitables en PHP

- >>> édition (création en ligne) par le professeur d'une nouvelle question durant la session
- >>> mise en oeuvre d'autres éléments de back-office pour la gestion du professeur (comme la gestion en ligne de thèmes, questions, sessions, utilisateurs, groupes).

Note sur l'utilisation javascript

Vous pouvez utiliser du javascript et les procédures AJAX pour optimiser votre application, ceci en avance sur le cours de S4 dédié à ces sujets.

Ce point est donc réservé aux équipes les plus avancées du point de vue des concepts. En discuter avec votre chargé de TPs.

Note sur l'utilisation d'un framework

Vous pouvez utiliser un framework pour ce projet. A l'IUT, nous mettrons à votre disposition 2 frameworks parmi les plus utilisés : symfony2 et zend-framework2 .

La mise en oeuvre d'un framework sera appréciée positivement si cela a permis pour votre projet de booster l'activité de l'équipe et les services mis en oeuvre, sans mettre en danger le travail personnel de chacun.

Ce point est donc réservé aux équipes les plus avancées du point de vue des concepts. En discuter avec votre chargé de TPs.

Note sur le Référencement des pages WEB

Dès sa conception le site pourra être optimisé pour le référencement dans les moteurs de recherche :

- Toutes les URL pourront être écrite suivant une présentation uniforme masquant l'existence de paramètres d'URL propre et de processus d'URL *rewriting*. Ainsi, l'url suivante est un exemple d'url propre <http://localhost/utilisateur/ModifContact/3>. Elle pourra être analysée pour retrouver la valeur des paramètres à prendre en compte, comme si l'url avait été écrite sous la forme : <http://localhost/control=utilisateur/action=ModifContact/idU=3>

Note : les notions de réécriture d'url sont classiques et immédiates dans les framework.

Note : les fonctions **explode** et **implode de PHP** vous seront fort utiles pour l'analyse et la formation automatisées de chaîne de caractères

- Réfléchir à la pertinence des noms des nouvelles URL.
- Tous les attributs **ALT** et **TITLE** seront renseignés.
- Vos pages seront bien structurées et l'entête (<head>) renseignée par existence de mots clefs, finalité de la page, titre ...

Note sur les extensions et modules activables de PHP et APACHE

En cas de besoin, n'hésitez pas à nous demander d'activer des modules du serveur web APACHE (à décommenter dans `httpd.conf`) et des extensions de PHP (à décommenter dans `php.ini`).