

Projet de Langages Web 2015 - M1 GIL

Thomas CAPET, Yohann HENRY

January 4, 2016

Contents

1	Présentation du projet	2
2	Mode d'emploi	2
2.1	Installation	2
2.2	Cas d'utilisation	2
3	Fonctionnement global	3
3.1	Routeur	4
3.2	Controleurs	4
3.3	Modèle	4
3.4	Base de données	4
3.5	Vues	4
4	Technologie et langages utilisés	4
4.1	Langages utilisés	4
4.2	Bootstrap	4
5	Améliorations possibles	5

1 Présentation du projet

L'objectif du projet de Langages Web était de développer une plateforme web permettant de concevoir des quiz et d'y jouer.

Une notion d'utilisateur était demandée : l'administrateur a tous les droits, un utilisateur inscrit et connecté peut créer des quiz et y jouer, et un utilisateur anonyme peut jouer aux quiz déjà créés.

Enfin, pour plus de facilité dans le partage des quiz, il devait être possible de transmettre l'URL d'un quiz. Chaque quiz devait donc posséder sa propre URL.

Notre site est adaptatif (*responsive*) ; il s'adapte à la taille de l'écran, et fonctionne également sur portables.

2 Mode d'emploi

2.1 Installation

Pour installer le site, il suffit de lancer la page `Install/install.php`. Si l'utilisateur tente d'accéder au fichier `index.php` alors que le fichier `bdd.php` est inexistant, il est immédiatement reporté sur la page d'installation.

La page est constitué d'un formulaire demandant les identifiants pour le compte d'administration ainsi que les identifiants, l'adresse et le nom de la table pour la base de données. S'ils sont correctement renseignés, l'utilisateur sera reporté sur la page d'accueil. L'installation est terminée. Sinon il sera reportée sur une page avec une liste des raisons pouvant expliquant l'échec de l'installation.

2.2 Cas d'utilisation

Le site possède une barre de menu horizontale, qui sera présente en haut de toutes les pages. Ce menu permet de retourner à l'accueil en cliquant sur **Quiz Constructor**, d'afficher la liste des quiz en cliquant sur **Quiz** et d'afficher la liste des utilisateurs inscrits en cliquant sur **Utilisateurs**. Egalement, si vous n'êtes pas connecté, il vous sera proposé à droite du menu de vous inscrire ou de vous connecter. Si vous êtes connecté, alors vous aurez la possibilité de vous déconnecter ou bien d'accéder à votre page de profil en cliquant sur votre identifiant.

Via la page d'accueil, vous pouvez **Commencer à jouer** et ainsi accéder à la liste de tous les quiz créés.

L'inscription se fait simplement. Après avoir cliqué sur **Inscription**, l'utilisateur arrive sur une page où il doit entrer un identifiant et un mot de passe.

Pour la connexion, il suffit de cliquer sur **Connexion**. L'utilisateur aura juste à rentrer son identifiant et son mot de passe, et la connexion s'effectuera.

Une fois la liste des quiz affichée, il vous est possible de jouer à n'importe lequel d'entre eux en cliquant sur son nom. S'affichera alors la liste de ses questions, à vous d'y répondre. Une fois toutes les questions répondues, le nombre de bonnes réponses vous sera transmis. Chaque quiz possédant une URL unique, un utilisateur peut le transmettre par mail ou par copier/coller. Un utilisateur connecté aura également la possibilité, via la liste des quiz, d'éditer les quiz qu'il a créé. Un administrateur pourra modifier n'importe quel quiz.

La page d'édition des quiz et la page de création se ressemblent beaucoup. Il est possible de modifier la description du quiz, ainsi que les questions enregistrées pour ce quiz. Un utilisateur peut également rajouter des questions à un quiz qu'il a créé. Pour créer un quiz, il suffira de se connecter et d'accéder à la liste des quiz. Un bouton **Créer un quiz** apparaîtra alors.

L'édition et la création des questions d'un sont similaires. L'utilisateur rentre ou modifie un intitulé (l'énoncé de la question), entre de 2 à 4 réponses pour cette question et indique quelle est la bonne réponse parmi les propositions entrées.

Via la liste des utilisateurs, il est possible d'accéder au profil de n'importe quel membre inscrit et, de son profil, aux quiz qu'il a créé.

3 Fonctionnement global

La vue se construit d'après une Url selon le schéma suivant.

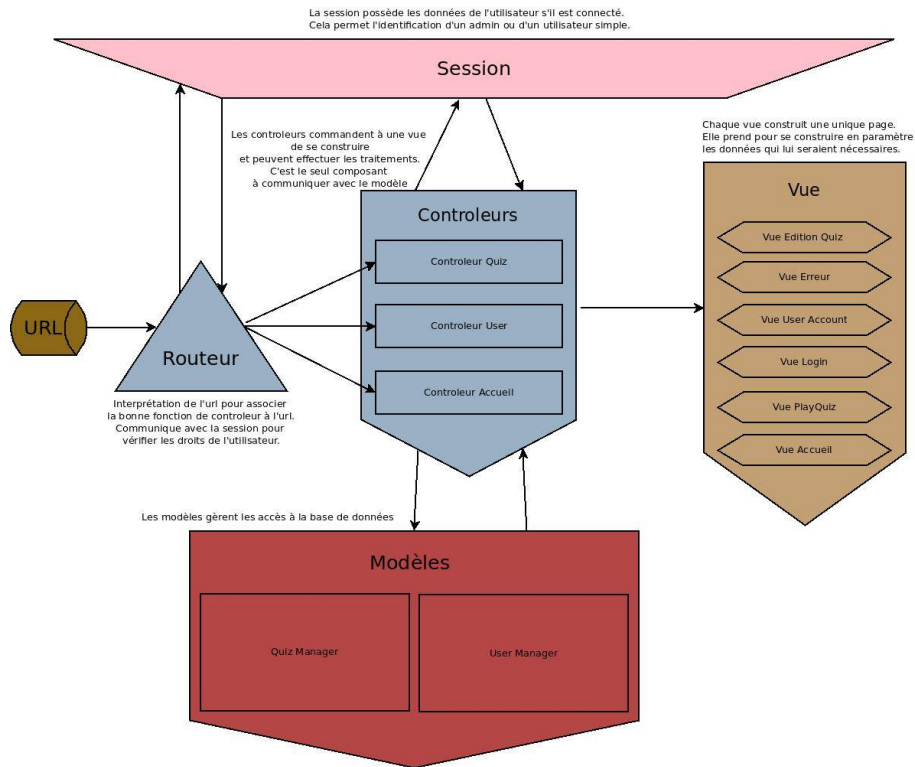


Figure 1: Fonctionnement global

3.1 Routeur

Le routeur est l'objet qui permet de rediriger une requête reçue vers le contrôleur et l'action associée. Le fichier `index.php` est très simple et se contente de demander au routeur d'extraire de la requête attributs `GET` et `POST` afin de déterminer quelle action de contrôleur est demandé. Il vérifie aussi les droits d'admission à certaines actions de contrôleurs en communiquant avec les attributs de session. Prenons l'exemple d'une URL `index.php?action=login`, le routeur va trouver la commande `action=login` et demandera au contrôleur des utilisateurs de lancer sa fonction `login`.

3.2 Contrôleurs

Les contrôleurs sont des classes contenant les actions possibles du site. Que ce soit la création d'un utilisateur, sa connexion, l'affichage de la page d'accueil ou le traitement d'un retour de formulaire, chaque action est représentée par une unique fonction d'un contrôleur spécifique. Ce sont les uniques objets qui pourront interagir avec le modèle de données. Il s'agit aussi ici de la dernière couche de sécurité. C'est dans ces classes que l'on surveille la majorité des entrées de l'utilisateur et les redirections vers la page d'erreur.

3.3 Modèle

Les différents modèles sont les interfaces entre les contrôleurs et la base de données. Chaque action est relativement simple et se compose majoritairement que de requêtes sql. Le modèle d'utilisateurs ne permet que d'ajouter des utilisateurs ou récupérer un ou des lignes de la base de données. Le but est de ne trouver aucune requête ailleurs que dans ces classes.

3.4 Base de données

La base de donnée est composée de trois tables : `quiz`, `question` et `user`.

3.5 Vues

Chaque vue est représentée par un fichier qui sera appelée par la classe `Vue` qui intégrera le contenu dans un modèle commun à toutes les vues. Le contrôleur fournit à la vue par le biais d'un tableau associatif les données extraits des modèles. C'est cette page, une fois construite, qui sera renvoyée au client.

4 Technologie et langages utilisés

4.1 Langages utilisés

Nous avons choisi de programmer en PHP, Javascript, HTML5 et CSS3, car ces quatre langages sont des langages web que nous maîtrisons bien.

Le PHP a l'avantage d'être orienté objet, ce qui nous a permis d'intégrer un patron MVC à notre projet.

4.2 Bootstrap

Bootstrap, ou Twitter Bootstrap, est un ensemble de code HTML et CSS permettant la mise en forme rapide des sites web.

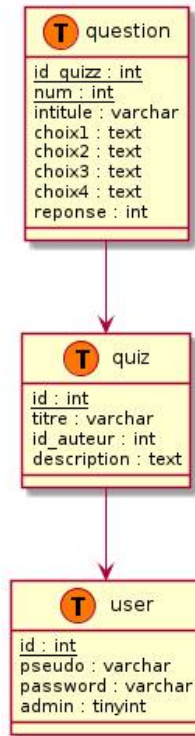


Figure 2: Base de données

Nous avons choisi de l'utiliser car il permet une grande rapidité de programmation en nous permettant de nous concentrer sur le fond plutôt que la forme. De plus, il permet de créer facilement un site web adaptatif (**responsive**), comme demandé dans les objectifs bonus du projet.

Il fonctionne en indiquant un ou plusieurs classes pour un élément. Il ira alors chercher dans son code CSS la mise en page nécessaire. L'imbrication de bloc `div` avec des classes est une fonctionnalité importée dans Bootstrap.

5 Améliorations possibles

Plusieurs améliorations pourraient être envisageables au niveau du site :

- Actuellement, il n'y a pas de moyen de supprimer des utilisateurs ou des quiz, même pour l'administrateur.
- Des statistiques pour chaque quiz (nombre de personnes l'ayant essayé, nombre de personnes l'ayant réussi ...) pourraient être ajoutées.
- Actuellement, le routeur interprète les URL afin de les rediriger vers les vues grâce aux attributs **GET** passés en paramètre dans l'URL.
(exemple : `http://localhost/web2015/index.php?action=quiz`).
Une réécriture d'URL pour cacher les attributs **GET** est envisageable, pour obtenir quelque chose comme `http://localhost/web2015/index.php/quiz`.