

Projet (3-4 personnes)

Vous devez réaliser une interface Web pour un questionnaire en ligne (sondage, quiz, test de personnalité, ce genre de choses...), et une interface d'administration (qui permet d'éditer les questionnaires).

Un questionnaire contient les éléments suivants :

- un titre, par exemple « Plutôt Tomate, Carotte, Radis ou Poireau ? Découvrez votre légume intérieur. » ;
- une liste de question/réponses.

Exemple de question « Quelle est votre couleur préférée ? », avec plusieurs réponses possibles (l'utilisateur doit choisir) :

- « rouge » (donne 1 point pour « tomate ») ;
- « orange » (donne 1 point pour « carotte ») ;
- « rose » (donne 1 point pour « radis ») ;
- « blanc » (donne 1 point pour « radis » et 1 point pour « poireau ») ;
- « vert » (donne 1 point pour « poireau »).

On affiche ainsi une succession de questions (sans indiquer les points gagnés, c'est une surprise !). À la fin du questionnaire, on affiche le résultat au joueur, par exemple « Vous êtes plutôt Carotte (11 points) ! ».

Le site

Une **page d'accueil** permettant d'accéder aux pages suivantes :

1. **Règles du jeu** : une page présentant le fonctionnement du questionnaire et de votre interface (vous pouvez reprendre la présentation ci-dessus).
2. Le **questionnaire** proprement dit.
3. **Crédits** : une page de « crédits » présentant votre équipe de conception et ses membres, ainsi que les langages utilisés, et, le cas échéant, les ressources (images, librairies) utilisées ainsi que leur licence.
4. **Documentation** : une page de présentation de votre projet et de documentation technique.
5. **Administration** : une page, protégée par un mot de passe, comportant un formulaire permettant d'éditer le questionnaire.

Toutes ces pages doivent permettre de revenir à la page d'accueil.

La page du jeu (JavaScript)

Vous devez concevoir l'interface du questionnaire qui affiche le titre du questionnaire, la première question, les réponses possibles ; puis, une fois que l'utilisateur a répondu, la seconde question, etc. Chaque réponse rapporte des points dans une certaine catégorie, par exemple 1 point en « carotte » et 2 points en « poireau ». Après la dernière question, on affiche le résultat.

L'interface de jeu doit utiliser les fonctions du DOM et la technologie AJAX pour interagir avec le serveur. **Il n'est pas question pour cette page d'utiliser du PHP.**

La page de documentation

Cette page explique la configuration, le paramétrage et le fonctionnement de votre site / jeu. Le cas échéant, elle précise les points techniques particuliers. Si le travail n'est pas fini, présente des limites ou des bugs, la documentation doit les pointer. Cette page doit être adaptée à l'impression avec les CSS.

Administration (PHP)

La page d'administration doit utiliser des formulaires et du PHP (pas de JavaScript). Son accès est protégé par un mot de passe. Elle affiche le questionnaire, dans des champs de formulaire éditables : son titre, la liste des questions, les réponses et les points qu'elles rapportent. Outre l'édition de ces champs, on doit pouvoir :

- ajouter et supprimer une question ;
- ajouter et supprimer une réponse.

Un bouton « Envoyer » permet d'enregistrer les modifications sur le serveur.

Si vous n'arrivez pas à créer un formulaire qui permette d'éditer toutes les questions, essayez de créer au moins un formulaire qui permet d'éditer la première question. De manière générale, si vous ne faites pas tout l'énoncé (ici ou ailleurs), pensez à l'indiquer sur la page Documentation.

Données

L'utilisation d'une base de données serait tout à fait légitime pour stocker le dictionnaire, mais pour simplifier le développement, le déploiement et la correction, nous n'en utiliserons pas. Le questionnaire sera stocké sur le serveur sous forme d'un fichier « questionnaire.txt », qui peut contenir plusieurs sortes de lignes :

- Le titre du questionnaire. Cette ligne commence par T| .
- Une question. Cette ligne commence par Q| .
- Une réponse (à la question précédente). Cette ligne commence par R| , et est suivie de la liste des points gagnés, séparés par des virgules (voir l'exemple ci-dessous).

Exemple

T|Plutôt Tomate, Carotte, Radis ou Poireau ? Découvrez votre légume intérieur.

Q|Quelle est votre couleur préférée ?

R|tomate|rouge

R|carotte|orange

R|radis|rose

R|radis,poireau|blanc

R|poireau|vert

Respectez bien ce format ! Vous pouvez ajouter des fonctionnalités, mais assurez-vous que vous êtes compatibles avec ce format (= qu'il n'y a pas de plantage si on utilise ce type de fichier).

Prévoyez un questionnaire d'au moins 5 questions.

Astuces (voir la documentation de PHP)

- La fonction *file* est tout à fait indiquée pour charger le fichier (elle retourne un tableau,

contenant une entrée par ligne du fichier).

- La fonction *file_put_contents* peut servir à enregistrer le fichier (elle permet de sauvegarder une chaîne dans un fichier).
- Les fonctions *implode* et *explode* seront utiles (elles transforment un tableau en chaîne et réciproquement).
- En PHP, le caractère "*\n*" désigne la fin de ligne.

Accès aux données

Vous aurez besoin de créer des services (en PHP) pour lire et écrire ces données :

- un service *getQuestion*, qui prend un paramètre *numero* (le numéro de la question), et affiche la question correspondante. Il n'affiche rien si il n'y a plus de questions (par exemple, si on demande la question numéro 6 alors qu'il n'y a que 5 questions).
- Un service *getReponse*, qui prend un paramètre *numeroQuestion* et un paramètre *numeroReponse*, qui retourne une réponse pour une question donnée. Il n'affiche rien si il n'y a plus de réponses (par exemple, si on demande la réponse numéro 6 de la question numéro 1, qui n'a que 5 réponses).
- un service *setQuestionnaire*, qui prend de nombreux paramètres, définis dans le formulaire de votre page d'administration, et qui enregistre un questionnaire.

Il s'agit d'une suggestion non exhaustive. Vous pouvez créer des services différents.

Organisation du travail

Pour gagner du temps, distribuez les tâches entre vous ! Vous pouvez par exemple distinguer les tâches de « conception » des pages XHTML, et d'habillage avec les feuilles de style. Vous pouvez aussi distribuer les pages entre vous.

Contraintes générales

L'ensemble de votre site et le jeu doivent respecter les normes du XHTML strict (version 1.0 ou 1.1). Votre code doit être valide. Exceptionnellement, le jeu et les formulaires peuvent être codés en HTML 5.

Vous devez respecter au mieux la sémantique du balisage dans vos documents. La mise en page et le design doivent être assurés à l'aide des CSS.

Vos CSS doivent être valides (selon les normes du validateur du W3C).

Votre site doit être portable et consultable au minimum sur Firefox et Chrome.

Éléments d'appréciation du travail

- « Aspect global » du travail : design, mise en page, lisibilité, ergonomie, navigation ... Le graphisme pur, n'étant pas l'objet du cours, n'est pas un aspect primordial (il est possible de faire un site esthétique avec peu d'éléments graphiques).
- Clarté et qualité du code : facilité de lecture (indentation, clarté ...), respect des recommandations actuelles du W3C (je pourrais utiliser les validateurs proposés par le W3C pour vérifier la syntaxe de documents).

- Bonne utilisation du XHTML/HTML et des feuilles de style, qualité du code (en accord avec les éléments exposés en cours).
- Organisation des répertoires et documents, stratégie de nommage des fichiers ...
- Prise en main facile, aspect « professionnel », aspect « fini », rapidité probable des mises à jours futures.
- Fiabilité, portabilité.
- Qualité du code JavaScript, fiabilité.
- Qualité du code PHP, fiabilité.
- Bonus : plusieurs questionnaires.
- Bonus : résultat (pseudo-)personnalisé