

ECOLE NATIONALE SUPÉRIEURE DE COGNITIQUE

1^{ère} année

U.E. Sciences fondamentales
Module Programmation Web

Projet web
du 21/02/20 au 17/04/20

QUIZZFLIX

Adresse du site : <https://quizzflix.000webhostapp.com/>
Identifiants joueur => login : Michel / mot de passe : michel
Identifiants administrateur => login : Amandine / mot de passe : mdp

LAVENSEAU Louis
VALLÉE Amandine

Année universitaire 2019 / 2020

Sommaire

Sommaire	2
Introduction	3
Modélisation	5
Base de données implantée	7
Structure de la base de données	7
COMPTE	7
QUIZZ	8
CATEGORIE	8
QUESTION	8
REPONSE	8
HISTORIQUE	9
Remplissage de la base de données	9
Site web	9
Interface pour un joueur	9
Messages d'erreurs	18
Interface pour un administrateur	22
Plan de test	28
Interface joueur	28
Interface administrateur	28
Conclusion	29

Introduction

Pour ce projet nous avons pour consigne de réaliser une plateforme de quizzes disponible en ligne. Nous avons choisi de nous orienter vers un thème précis : des quizzes concernant les séries disponibles sur Netflix. Nous avons alors nommé notre site Quizzflix.

Notre planning prévisionnel était le suivant :

	semaine du 24/02 au 01/03	semaine du 02/03 au 08/03	semaine du 09/03 au 15/03	semaine du 16/03 au 22/03	semaine du 23/03 au 29/03	semaine du 30/03 au 07/04
Réflexion sur le thème du site, son aspect visuel, son organisation						
Maquettage						
Réflexion à la structuration de la base de données						
Création de quizz						
Rédaction des scripts SQL de création et d'initialisation de la base de données						
Code de l'aspect fonctionnel du site (php et html)						
Code de l'aspect esthétique du site (css)						
Rédaction du rapport						
Mise en ligne du site						

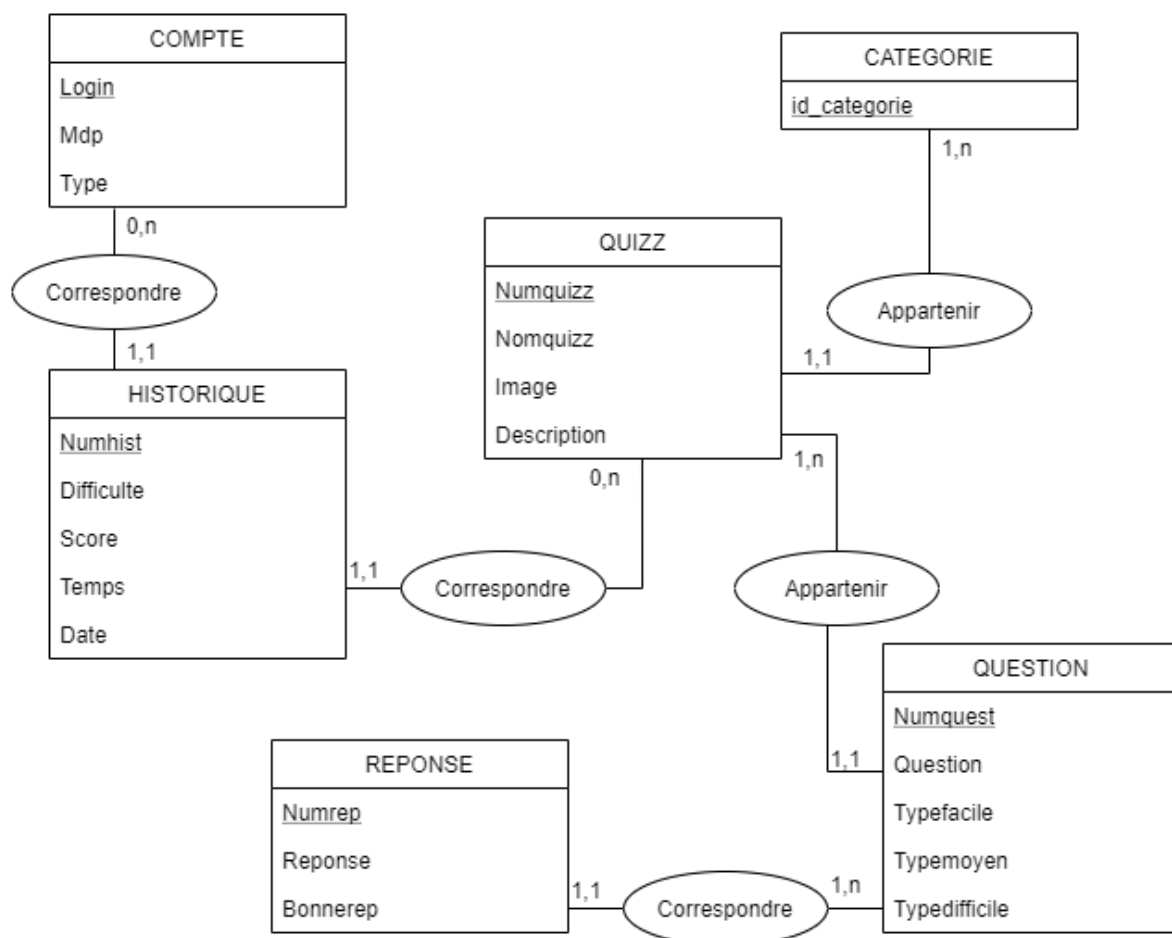
Finalement le planning suivi fut celui-ci :

	semaine du 24/02 au 01/03	semaine du 02/03 au 08/03	semaine du 09/03 au 15/03	semaine du 16/03 au 22/03	semaine du 23/03 au 29/03	semaine du 30/03 au 05/04	semaine du 06/04 au 12/04	semaine du 13/04 au 17/04
Réflexion sur le thème du site, son aspect visuel, son organisation								
Maquettage								
Réflexion à la structuration de la base de données								
Création de quizz								
Rédaction des scripts SQL de création et d'initialisation de la base de données								
Code de l'aspect fonctionnel du site (php et html)								
Code de l'aspect esthétique du site (css)								
Mise en ligne d'une première version du site								
Réalisation d'un tutoriel								
Rédaction du rapport								
Mise en ligne de la version finale du site								

Pour ce projet nous avons essayé de nous répartir les tâches équitablement. Dans un premier temps nous avons travaillé ensemble pour trouver des idées, prendre des décisions et réfléchir à la structuration du site. Puis nous nous sommes réparti chacun la moitié des maquettes à réaliser après nous être mis d'accord sur l'aspect visuel global du site. De la même manière, après avoir décidé ensemble de la composition type d'un quizz, nous avons chacun de notre côté créé plusieurs quizz. Puis nous avons chacun rédigé la moitié des scripts SQL de création de la base de données, puis d'initialisation de celle-ci avec les quizz déjà créés. Pour la partie codage du site, Amandine s'est plutôt penchée sur la partie du site concernant les joueurs, tandis que Louis a travaillé sur la partie réservée aux administrateurs.

Modélisation

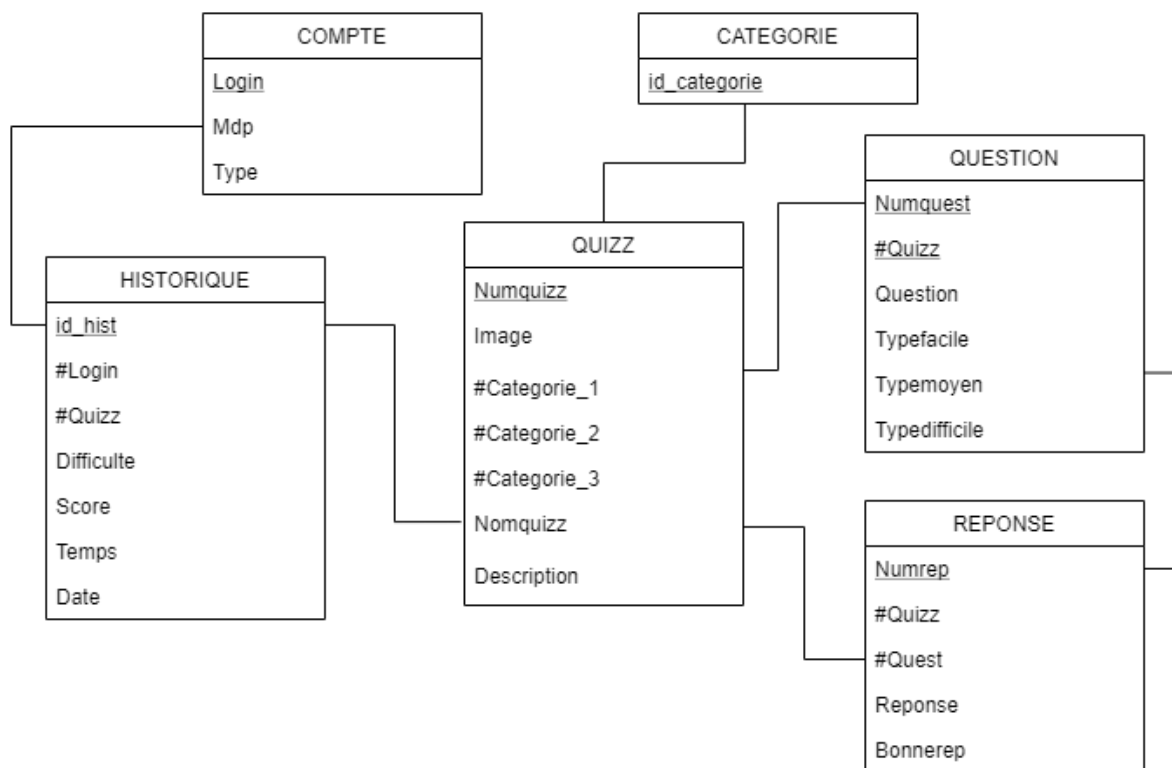
Pour organiser Quizzflix, nous avons adopté le Modèle Conceptuel de Données suivant :



Celui-ci nous a amené à ce dictionnaire de données :

Désignation	Signification	Type
Login	Login d'un joueur ou d'un administrateur	VARCHAR(15)
Mdp	Mot de passe d'un joueur ou d'un administrateur	VARCHAR(20)
Type	Joueur ou administrateur	VARCHAR(10)
Numhist	Identifiant d'une partie enregistrée dans l'historique	INTEGER(5)
Difficulte	Difficulté choisie lors de la partie	VARCHAR(10)
Score	Score obtenu pour une partie	INTEGER(2)
Temps	Durée d'une partie	INTEGER
Date	Date à laquelle la partie a été jouée	DATE
Numquizz	Identifiant du quizz	INTEGER(3)
Nomquizz	Nom du quizz	VARCHAR(30)
Image	Chemin relatif de l'image correspondant au quizz	VARCHAR(30)
Description	Phrase introductive au quizz	TEXT
id_categorie	Identifiant d'une catégorie de quizz	VARCHAR(30)
Numquest	Numéro d'une question dans un quizz	INTEGER(2)
Question	Énoncé de la question	TEXT
Typefacile	Type de question lorsque la difficulté choisie est facile (QCM, ouverte : OUV ou vrai/faux : VF)	VARCHAR(3)
Typemoyen	Type de question lorsque la difficulté choisie est moyen	VARCHAR(3)
Typedifficile	Type de question lorsque la difficulté choisie est difficile	VARCHAR(3)
Numrep	Identifiant d'une réponse possible pour une question	INTEGER(4)
Reponse	Énoncé de la réponse	TEXT
Bonnerep	Indicateur pour dire si la réponse est correcte ou non (oui ou non)	CHAR(3)

Le schéma relationnel correspondant est alors le suivant :



De plus les maquettes des écrans de l'interface sont en annexe. Depuis la réalisation de ces maquettes certains aspects ont changé suite à des changements d'avis d'ordre esthétique et à des contraintes techniques.

Base de données implantée

Structure de la base de données

Pour mettre en place notre base de données nous avons rédigé un script SQL **creationTables.sql**. Dans la suite nous allons expliquer nos choix quand à la structure de chaque table.

COMPTE

Nous avons choisi pour un compte utilisateur donné d'enregistrer dans la base de données un login, pour s'identifier, qui est la clé primaire de la table Compte, et donc qui doit être unique. Nous avons aussi stocké le mot de passe de l'utilisateur, ainsi que le type d'utilisateur, à savoir joueur ou administrateur.

QUIZZ

Nous avons choisi d'identifier un quizz par un numéro, c'est donc la clé primaire de cette table. Puis nous avons évidemment stocké le nom du quizz. Nous avons fait le choix d'ajouter une petite description pour chaque quizz, pour donner envie au joueur de le faire. Nous avons également ajouté une image pour chaque quizz, qui correspond à une affiche de la série. Dans la base de données nous avons alors stocké le chemin relatif de l'image. Enfin nous avons décidé pour rajouter de la structure, d'organiser les quizz par catégories des séries (comédies, drames...) en reprenant les catégories des séries sur Netflix. Nous avons alors décidé qu'un quizz pouvait correspondre à un maximum de trois catégories, c'est pour cela que nous avons stocké trois catégories pour chaque quizz, en sachant qu'une ou plusieurs de ces catégories peuvent être nulles.

CATEGORIE

Dans cette table on retrouve seulement un identifiant correspondant au nom d'une catégorie de quizz.

QUESTION

Chaque question est identifiée par une double clé primaire se composant du numéro de la question dans le quizz (de 1 à 10 en général) ainsi que de l'identifiant du quizz. On retrouve alors dans cette table l'énoncé de la question, puis le type de la question pour chaque niveau de difficulté.

Nous avons décidé pour notre site de proposer trois types de questions différents : les questions à choix multiples(QCM), les vrai/faux (VF) et les questions ouvertes(OUV) où il faut répondre directement en écrivant la réponse sur son clavier. Nous avons choisi de proposer trois niveaux de difficulté pour chacun des quizz (facile, moyen et difficile). Ces niveaux de difficulté influent sur le nombre de réponses lorsque le type de question est QCM (3 réponses possibles pour le niveau facile, 4 réponses possibles pour le niveau moyen et 6 réponses possibles pour le niveau difficile). Mais ils influent aussi sur le type de la question : une question de type QCM en niveau facile peut passer en question ouverte pour les niveaux plus difficiles. D'où notre nécessité de stocker le type de question pour les trois niveaux de difficulté différents.

REPONSE

Chaque réponse correspond à une réponse possible, mais pas forcément bonne, d'une question. Nous avons choisi d'identifier une réponse par un numéro. Chaque réponse correspond à une question d'un quizz, c'est pour cela que nous avons stocké ces deux informations dans cette table aussi. Nous avons évidemment eu besoin de stocker l'énoncé de la réponse également.

Puis la dernière information qui nous était nécessaire était de savoir si la réponse était une bonne réponse à la question, ou non. En sachant qu'une question QCM peut avoir au maximum deux bonnes réponses (car seulement 3 propositions au niveau facile), et que sinon toutes les autres questions ne peuvent avoir qu'une bonne réponse possible. Nous

avons alors stocké cette information dans la colonne Bonnerep qui correspond à un 'oui' ou un 'non'.

HISTORIQUE

Chaque historique correspond à une partie jouée, donc à une réalisation d'un quizz. La table historique va servir à un joueur à avoir accès à ses meilleurs scores sur le site, ou à ses derniers quizz réalisés. Elle va aussi permettre d'afficher le meilleur score ainsi que la personne l'ayant effectué, à la fin de chaque quizz. De plus elle permet de déterminer les quizz les plus populaires du site (qui ont été effectué le plus de fois).

Nous avons décidé d'identifier un historique par un numéro. Pour chaque historique on stocke ainsi le login de la personne ayant effectué le quizz, le numéro du quizz, le score de la personne, le temps qu'elle a mis pour faire le quizz et la date à laquelle il a été effectué.

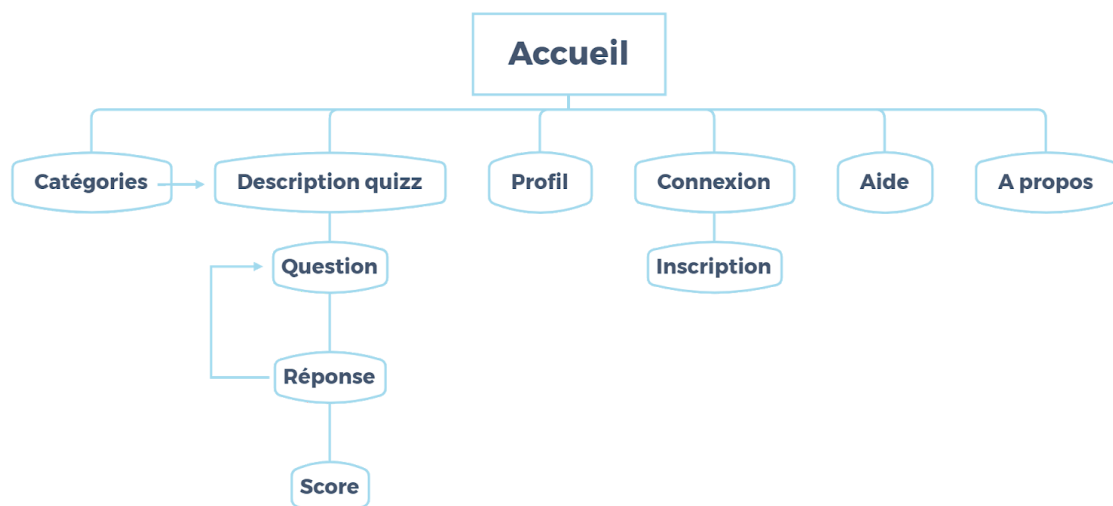
Remplissage de la base de données

Nous avons créé nous-même un total de 15 quizz pour initialiser la base de données. Puis nous avons retranscrit ces quizz sous la forme d'un script de remplissage de la base de données, **initialisation.sql**.

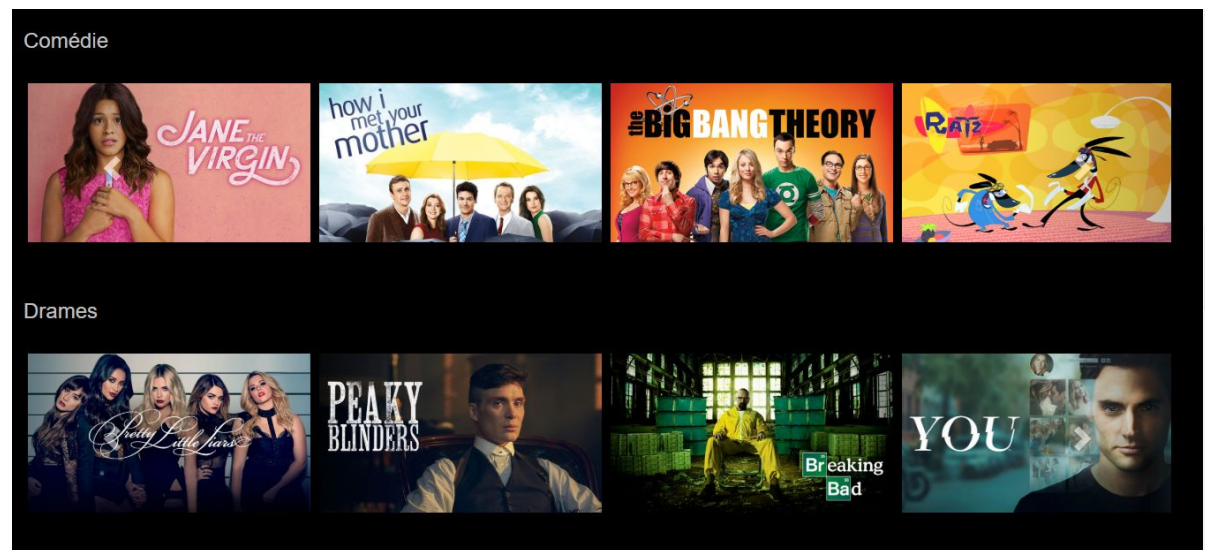
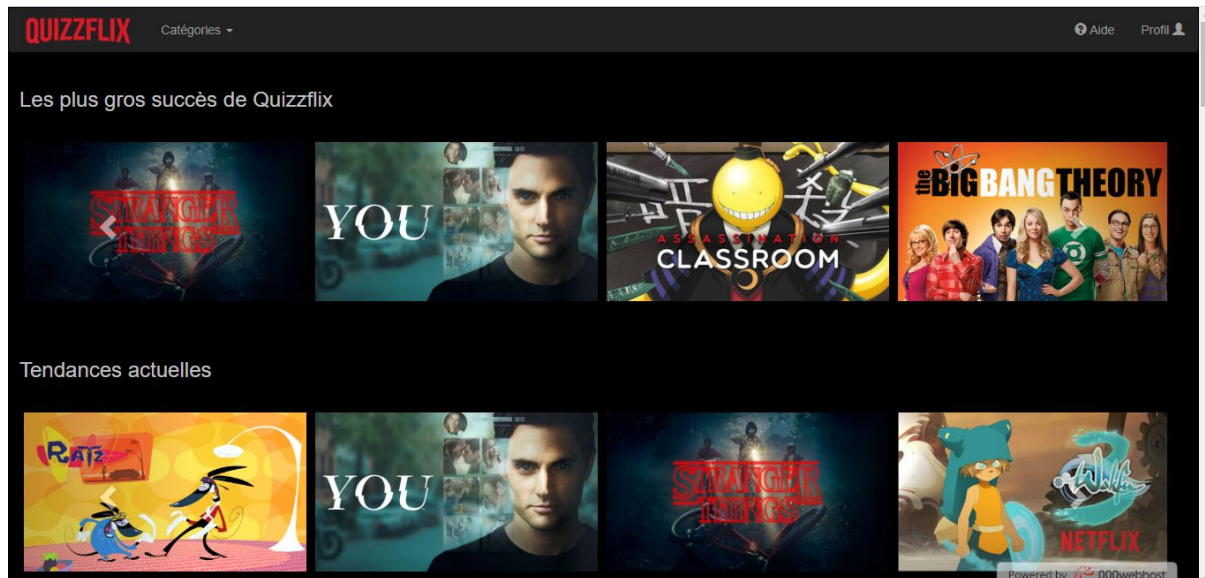
Site web

Interface pour un joueur

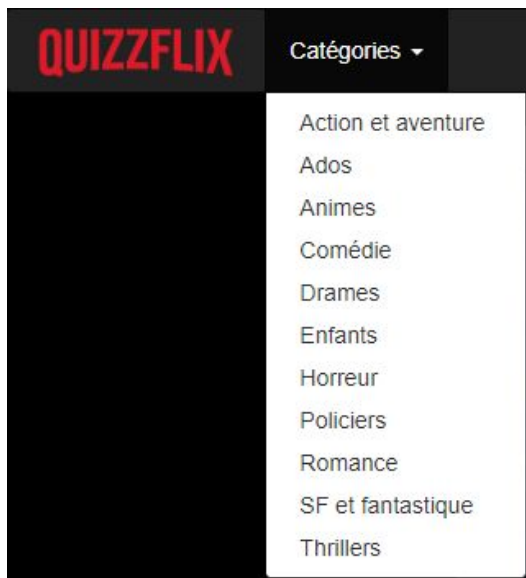
L'arborescence de notre site web est la suivante :



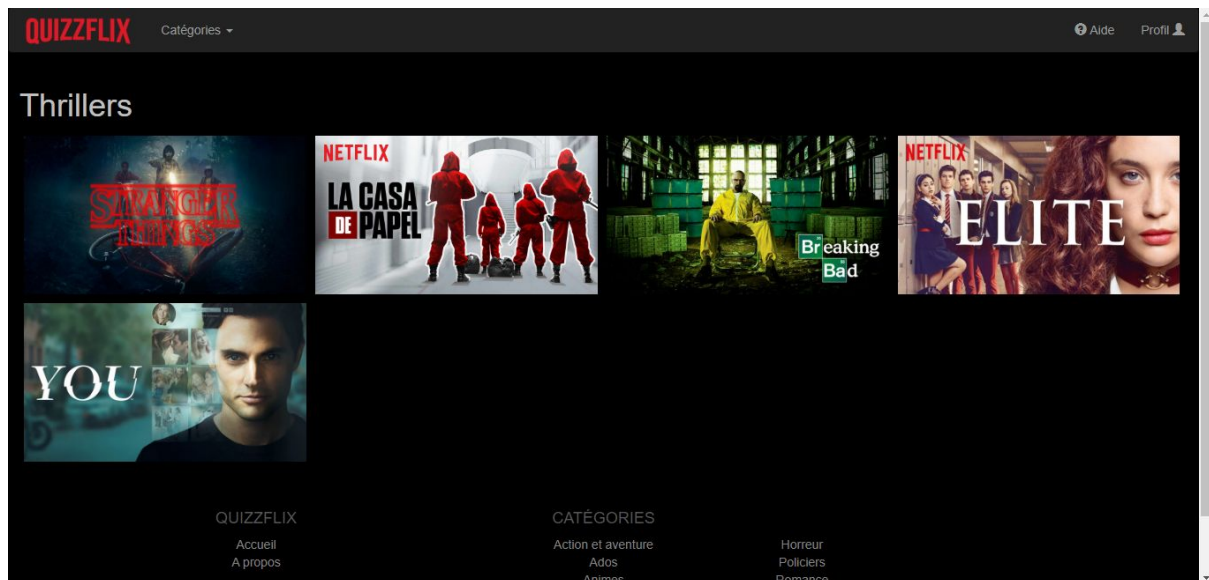
En arrivant sur le site, on se retrouve sur la page **index.php**, qui correspond à l'accueil du site. Sur cette page on va pouvoir retrouver tous les quizz du site selon une organisation précise. Le premier carrousel correspond aux quizz les plus populaires du site, puis le deuxième correspond aux quizz les plus populaires cette dernière semaine. Enfin en descendant sur la page on va retrouver les carrousels correspondant aux différentes catégories de série triées par ordre alphabétique.



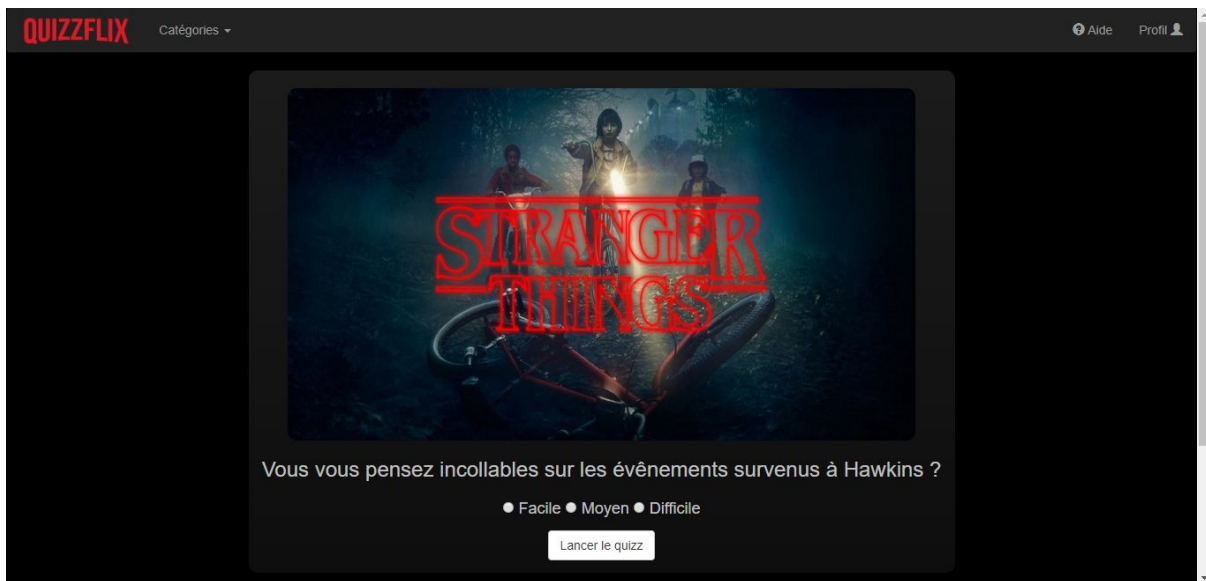
On peut retrouver ces catégories depuis n'importe quelle page du site avec la barre de navigation en haut (**barre.php**) ou alors le bandeau en bas (**footer.html**) :



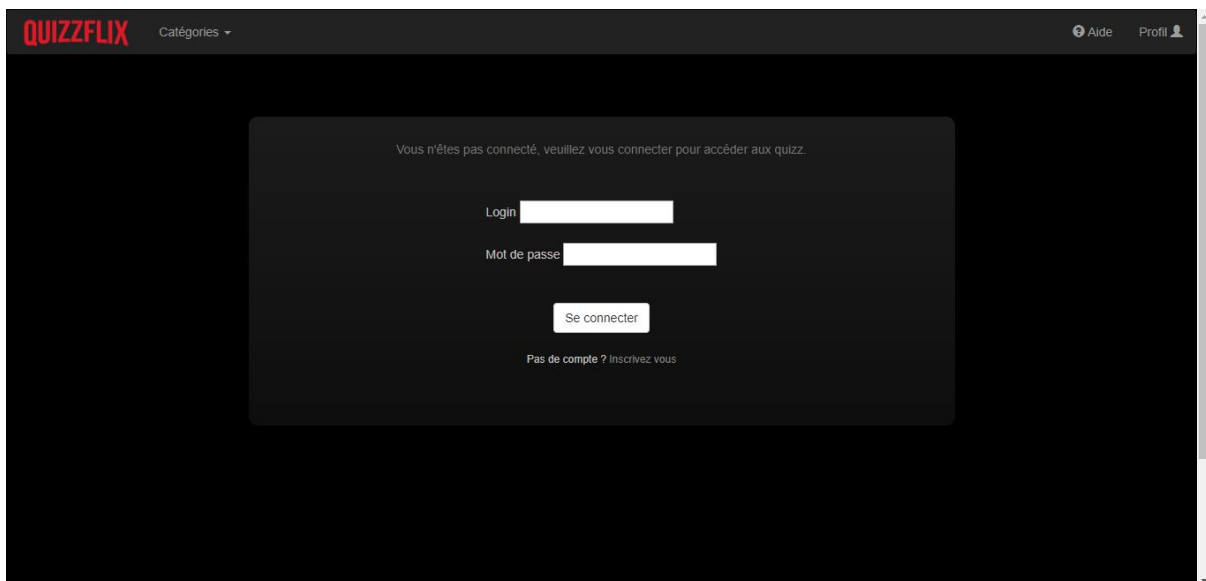
La page consacrée à une catégorie de séries, **categorie.php**, se présente alors ainsi :



Lorsque l'on va cliquer sur l'image d'un quizz depuis la page d'accueil ou de catégorie, on va se retrouver sur une page de présentation du quizz, **DescrQuizz.php** :



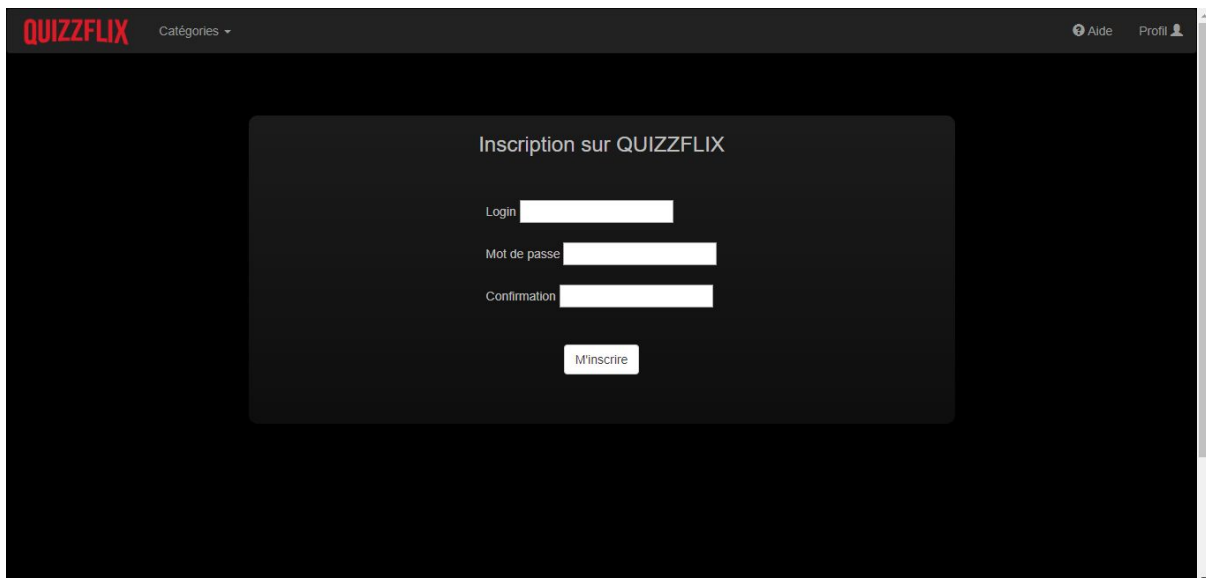
Depuis cette page on va pouvoir choisir la difficulté que l'on souhaite pour le quizz et lancer une partie, ce qui va nous amener à la page **question.php**, mais seulement si on est déjà connecté. Sinon on va être redirigé vers la page **connexion.php** :



Cette page est aussi accessible quelle que soit la page du site sur laquelle on se trouve, depuis la barre de navigation en haut : en cliquant sur "connexion/inscription" si l'on est pas connecté ou bien sur "déconnexion" si l'on était connecté :

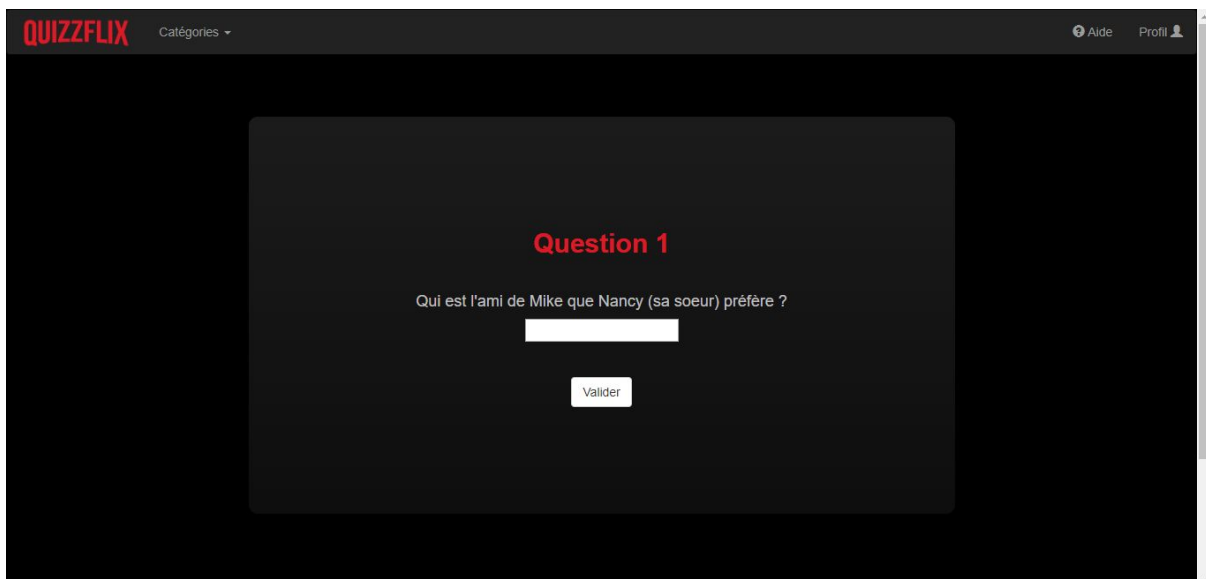


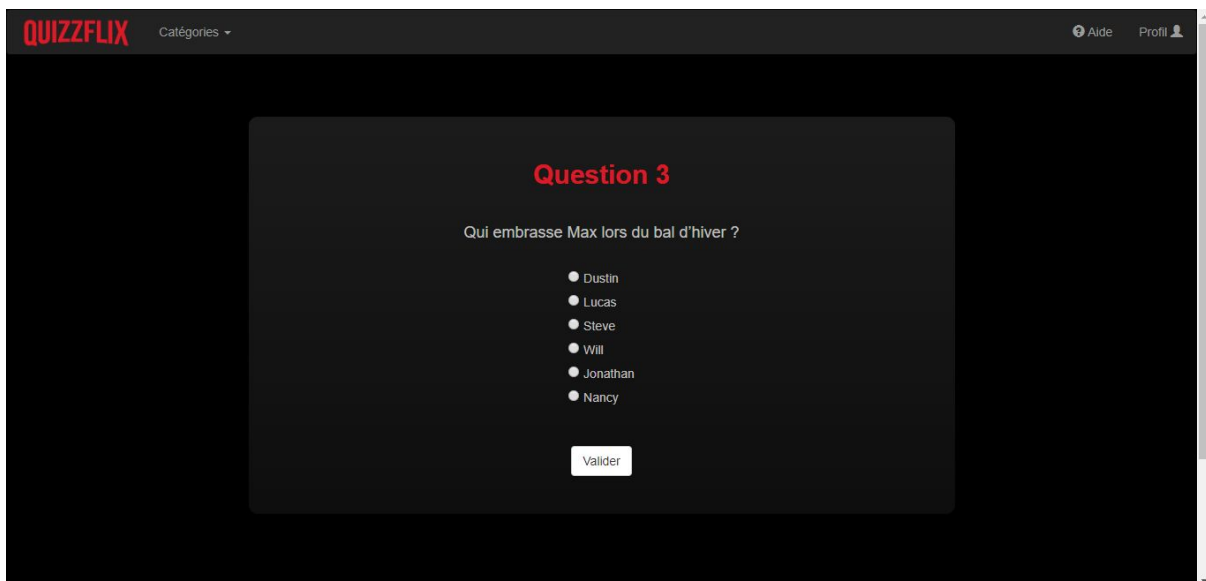
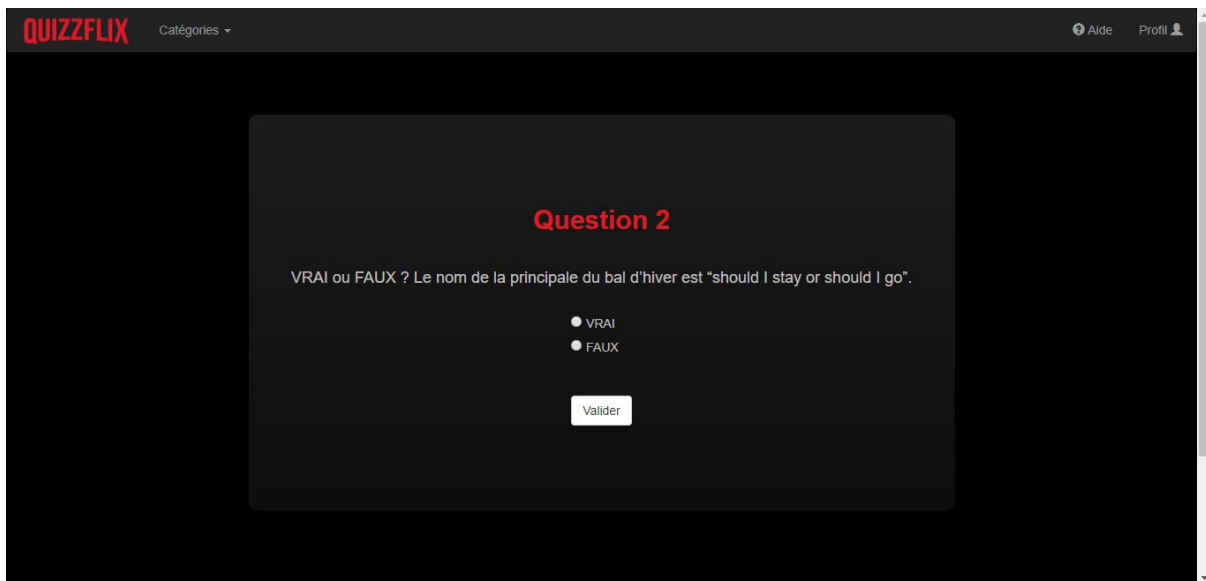
Si l'on n'a pas de compte Quizzflix, en cliquant sur "Inscrivez-vous" depuis la page de connexion on va être redirigé vers la page **inscription.php** :



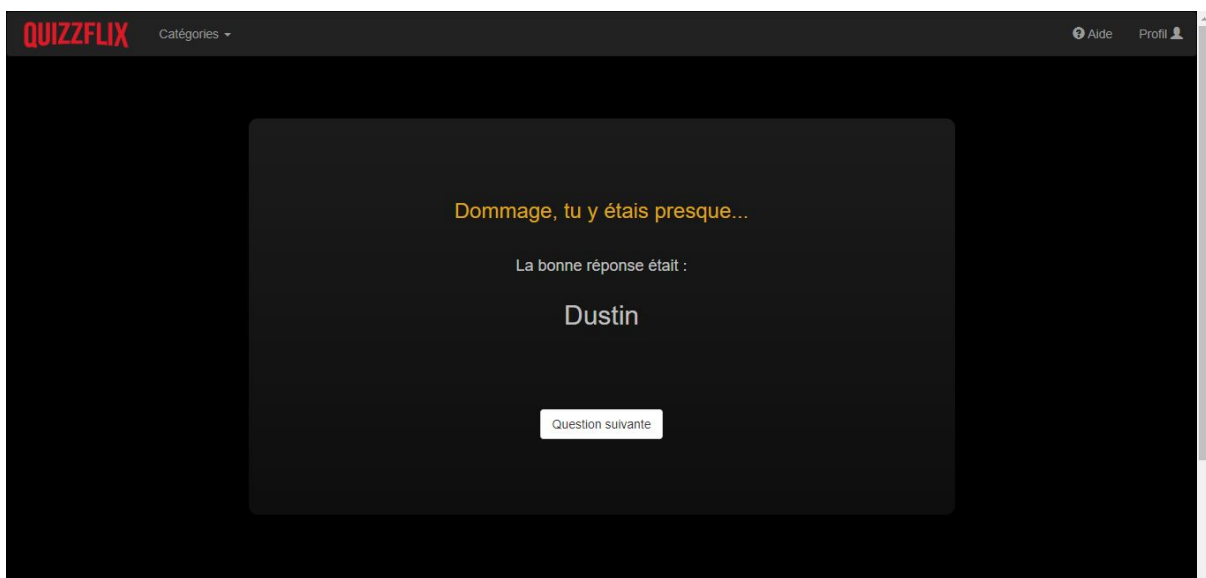
Si l'on se connecte, ou que l'on s'inscrit (ce qui nous connecte en même temps) on est redirigé à l'accueil, sauf si on est arrivé sur la page de connexion en essayant de lancer un quizz, dans ce cas le quizz voulu commence.

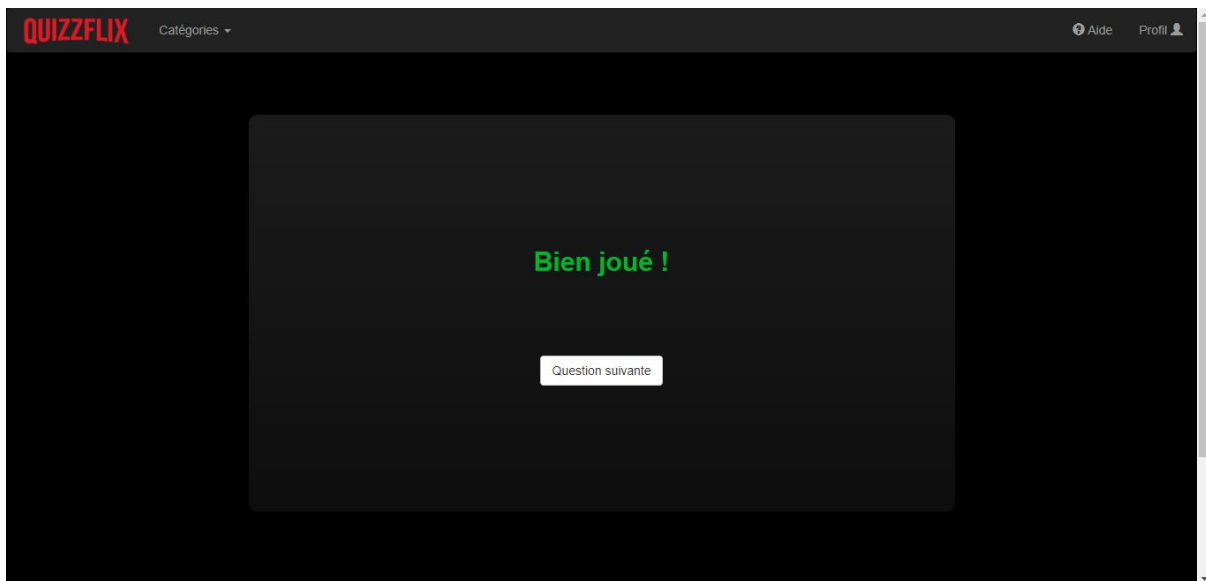
Lorsqu'un quizz se lance, on arrive sur la page **question.php** :



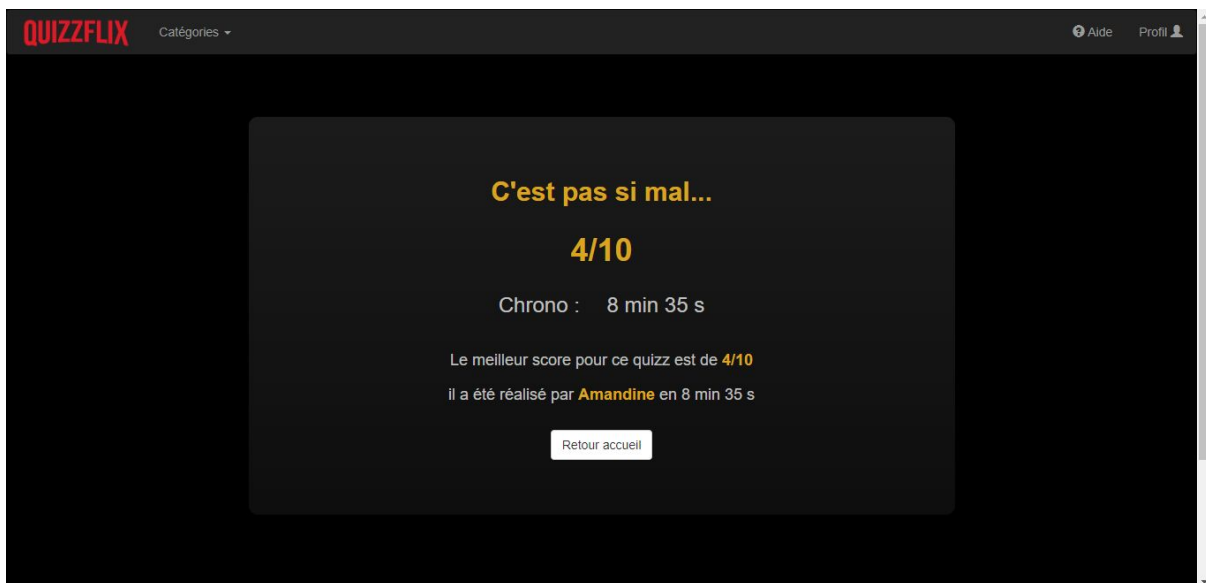


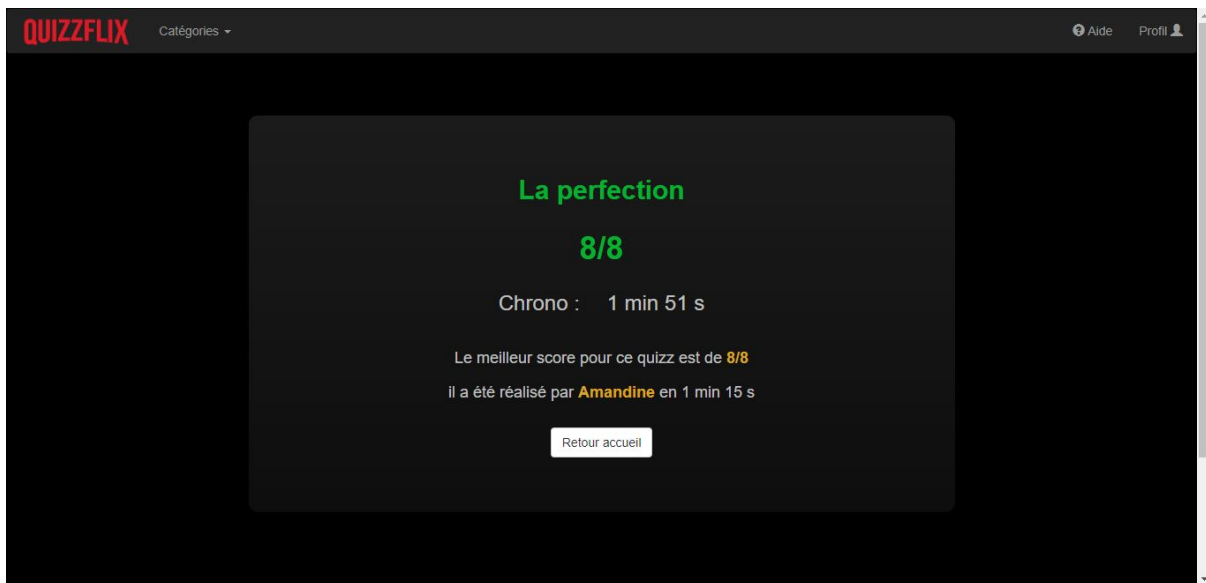
En validant on passe à la page **correction.php** :





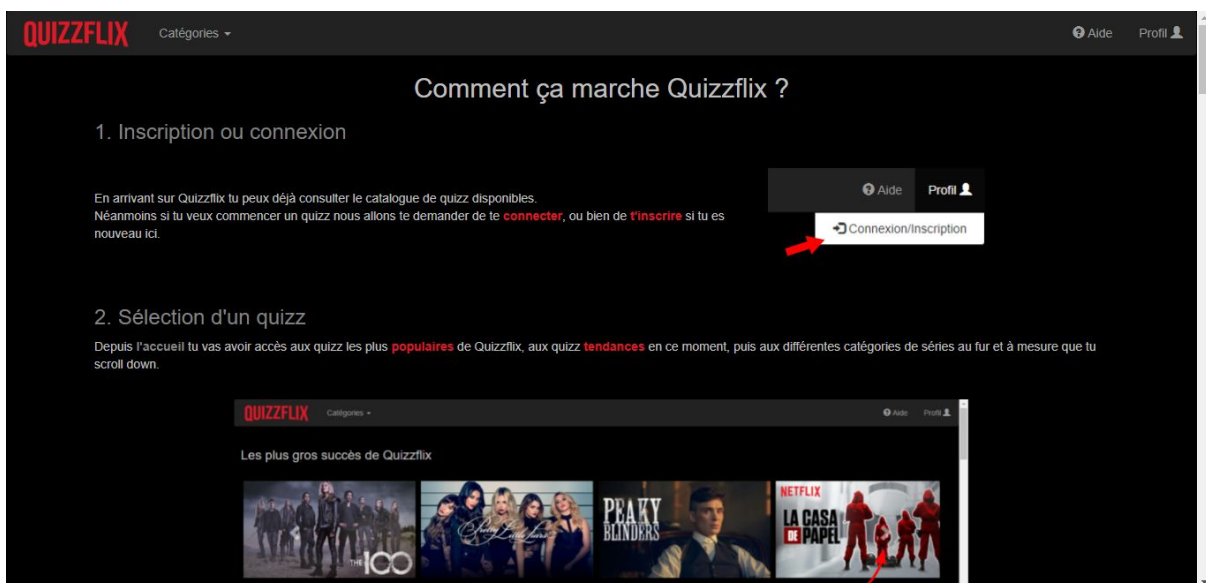
Puis suivant le nombre de questions du quizz les pages de question et correction se succèdent, jusqu'à arriver à la page **score.php** :





Sur cette page on voit notre score et chrono pour le quizz que l'on vient d'effectuer, puis le meilleur score pour ce quizz ainsi que la personne qui l'a fait. Un bouton permet aussi de retourner à l'accueil. Nous avons fait en sorte d'afficher un message "personnalisé" en fonction du score obtenu, selon si c'est un bon score ou moins bon.

La barre de navigation **barre.php** est accessible depuis toutes les pages du site. Elle permet de retourner à l'accueil en cliquant sur le logo de Quizzflix à gauche, ou bien d'aller à une catégorie de quizz précise comme expliqué précédemment avec le menu déroulant "Catégories". A droite en cliquant sur "Aide" on arrive au tutoriel qui explique le fonctionnement du site, **tuto.php** :



Dans les explications on va retrouver des liens vers d'autres pages du site. Par exemple les pages de connexion ou d'inscription vont être accessibles, seulement si l'utilisateur n'est pas déjà connecté. Un lien vers l'accueil est également intégré aux explications. De plus on

va retrouver un lien qui va nous amener tout en haut de la page, et un autre qui au contraire va nous amener tout en bas de celle-ci, ces deux liens ont pour but de montrer où l'on peut retrouver les différentes catégories de quizz (dans la barre du haut et dans celle du bas). Vers la fin des explications un lien est aussi disponible pour se rendre sur son profil (**profil.php**) , uniquement pour les personnes qui sont connectées.

Mais la manière usuelle d'accéder à son profil est depuis la barre de navigation, lorsqu'on est connecté, en cliquant sur "Mon profil" :



La page **profil.php** se présente alors ainsi :

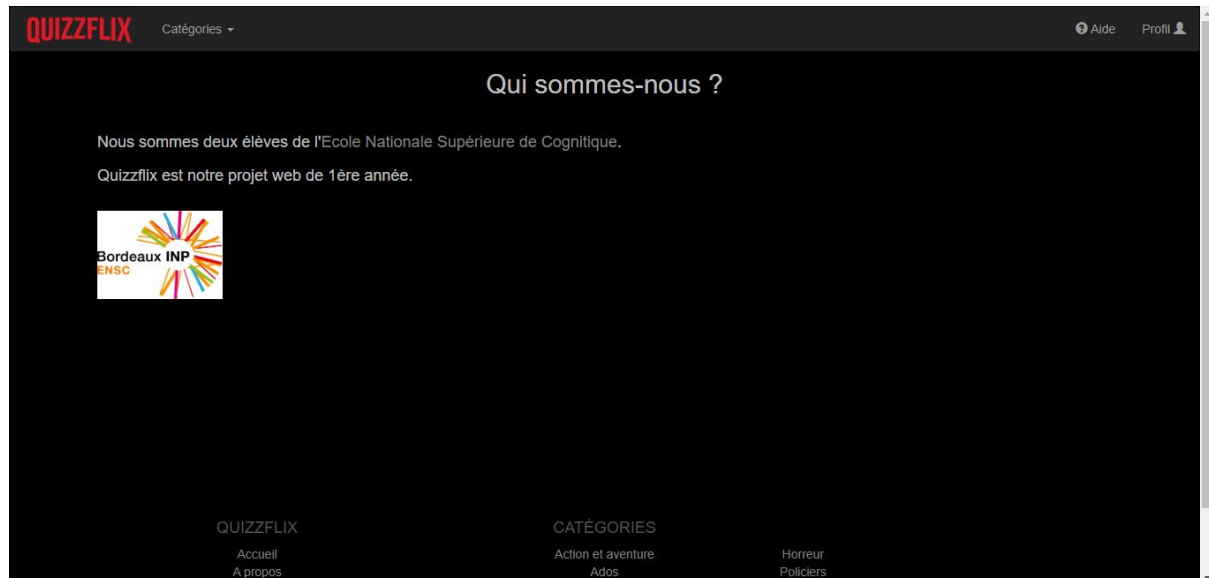


On a un tableau récapitulatif des 5 meilleurs scores du joueur, puis en dessous un tableau se présentant de la même manière avec les 10 derniers quizz effectués par le joueur.

La barre du bas (**footer.php**) est disponible depuis toutes les pages du site, comme dit précédemment.



En plus de permettre d'accéder aux pages des catégories, elle permet de revenir à l'accueil, et d'accéder à la page **apropos.php** en cliquant sur "A propos". Cette page précise que ce site est fait dans le contexte de ce projet, et on y retrouve des liens vers le site de l'école.



Messages d'erreurs

Pour éviter les problèmes liés aux informations non remplies dans les formulaires, dans plusieurs pages nous testons les données rentrées et affichons des messages d'erreurs si ces données ne sont pas conformes.

Sur la page de connexion, **connexion.php**, si l'on tente de se connecter avec un login qui n'est pas dans la base de données on a l'erreur suivante :

The screenshot shows a login form on a dark background. At the top, a message reads: 'Vous n'êtes pas connecté, veuillez vous connecter pour accéder aux quizz.' Below this are two input fields: 'Login' and 'Mot de passe'. A 'Se connecter' button is positioned below the password field. A red error message, 'Ce compte n'existe pas', is displayed below the button. At the bottom, there is a link: 'Pas de compte ? Inscrivez vous'.

De la même manière, lorsqu'on veut s'inscrire, sur **inscription.php**, on ne peut pas choisir un login qui est déjà dans la base de données :



Inscription sur QUIZZFLIX

Login

Mot de passe

Confirmation

Ce login est déjà pris, veuillez en choisir un autre.

On ne peut pas non plus s'inscrire si la confirmation du mot de passe n'est pas correcte :



Inscription sur QUIZZFLIX

Login

Mot de passe

Confirmation

Veuillez écrire le même mot de passe dans les deux zones.

Pour les inscriptions on a donc deux types d'erreurs possible, l'erreur vérifiée en priorité est celle du login. Donc si on prend un login qui est déjà pris, et qu'on se trompe dans l'écriture du mot de passe, seule la première erreur va être affichée.

Ensuite au moment de lancer un quizz il faut obligatoirement sélectionner un niveau de difficulté à la page **DescrQuizz.php** :



Vous vous pensez incollables sur les événements survenus à Hawkins ?

Veuillez sélectionner un niveau de difficulté pour lancer le quizz.

☐ Facile ☐ Moyen ☐ Difficile

Lancer le quizz

Il faut aussi obligatoirement choisir le bon nombre de réponses à la page **question.php** :

Question 1

Qui est l'ami de Mike que Nancy (sa soeur) préfère ?

Il faut choisir une réponse

- ☐ Dustin
- ☐ Lucas
- ☐ Will

Valider

Question 6

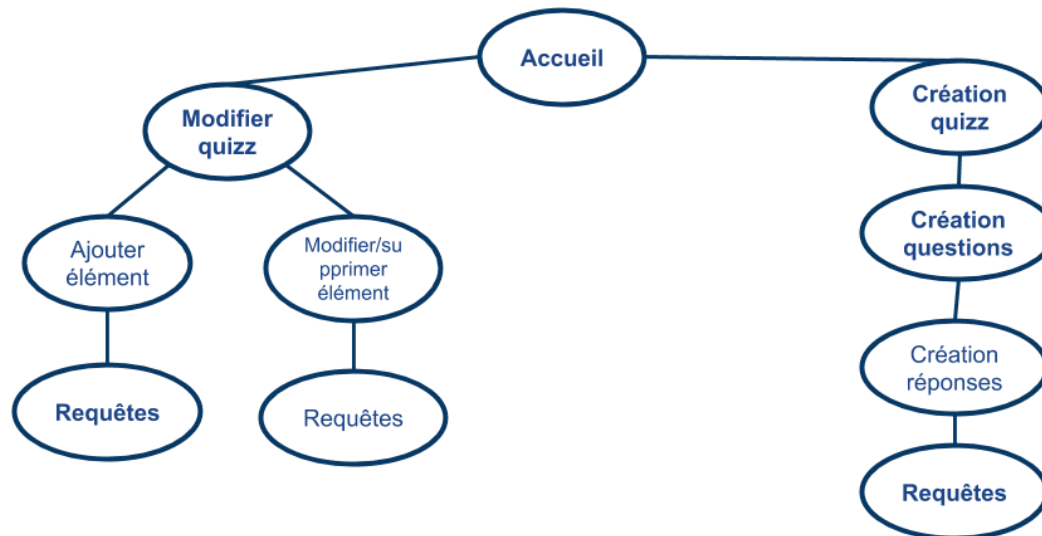
Quels sont les pouvoirs de Onze ? (2 réponses)

Il faut choisir deux réponses

- ☐ Contrôler les choses à distance
- ☐ Voir des gens à distance grâce à des ondes radio et une photo
- ☐ Devenir invisible

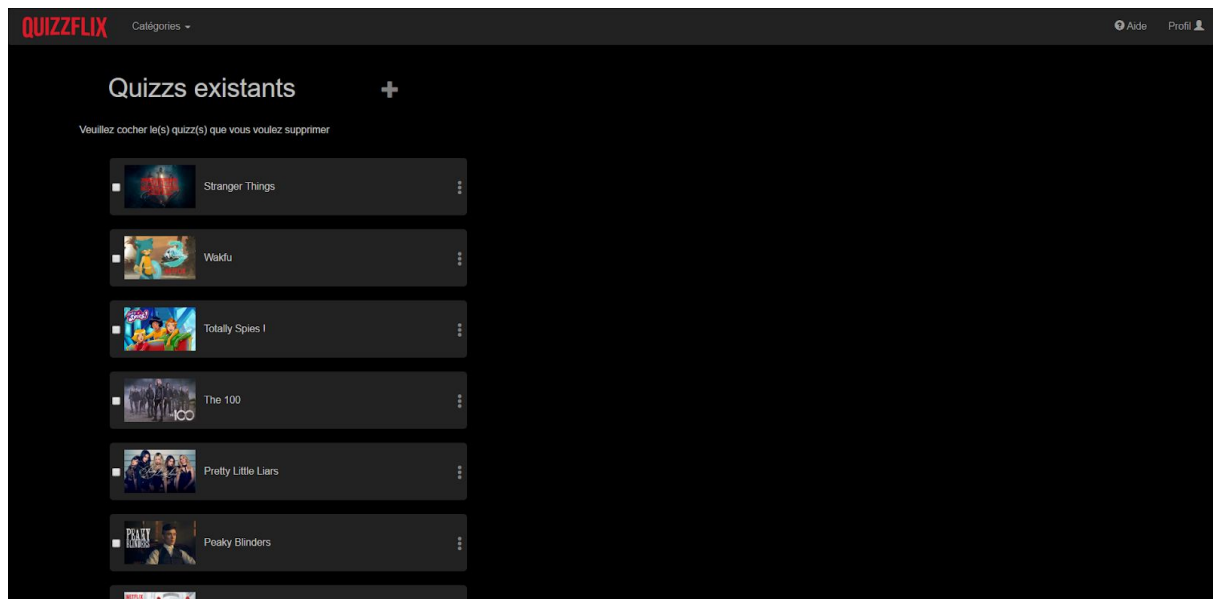
Valider

Interface pour un administrateur

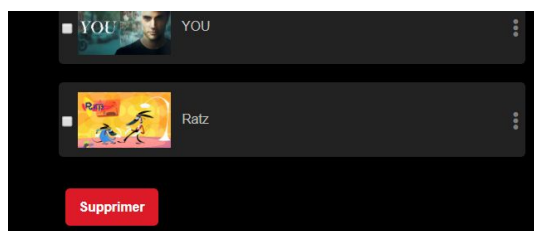


La partie du site réservée pour les administrateurs suit une arborescence en forme de “V” inversé. Nous avons opté pour une arborescence de ce type car elle est adaptée aux fonctions de l’administrateur : supprimer, modifier ou créer un quizz. 3 fonctions qui se placent parfaitement sur les 3 “extrémités” de la forme. Le bas du V correspond parfaitement à une page qui permet la suppression des quizzes, car celle-ci nécessite un accès à tous les quizzes et se fait de manière instantanée, contrairement aux deux autres tâches de l’administrateur. Par conséquent, nous avons envisagé une page d’accueil, sur laquelle sont présents l’ensemble des quizzes présents dans la base de données, et depuis laquelle nous pouvons supprimer les quizzes, ou accéder aux pages qui permettent les deux autres fonctions de l’administrateur.

La partie admin se divise donc grossièrement en trois grandes parties : une page avec un accès à l’ensemble des quizzes, et la possibilité de les supprimer instantanément, un ensemble de pages consacrées pour la création de nouveaux quizzes, et un deuxième ensemble de pages pour la modification de ceux-ci. Lorsque qu’un utilisateur se connecte au site en tant qu’administrateur, il est envoyé sur cette page. Après avoir fini de supprimer, créer ou modifier un quizz, il est ramené sur cette page. Elle correspond au script **accueilAdmin.php** .



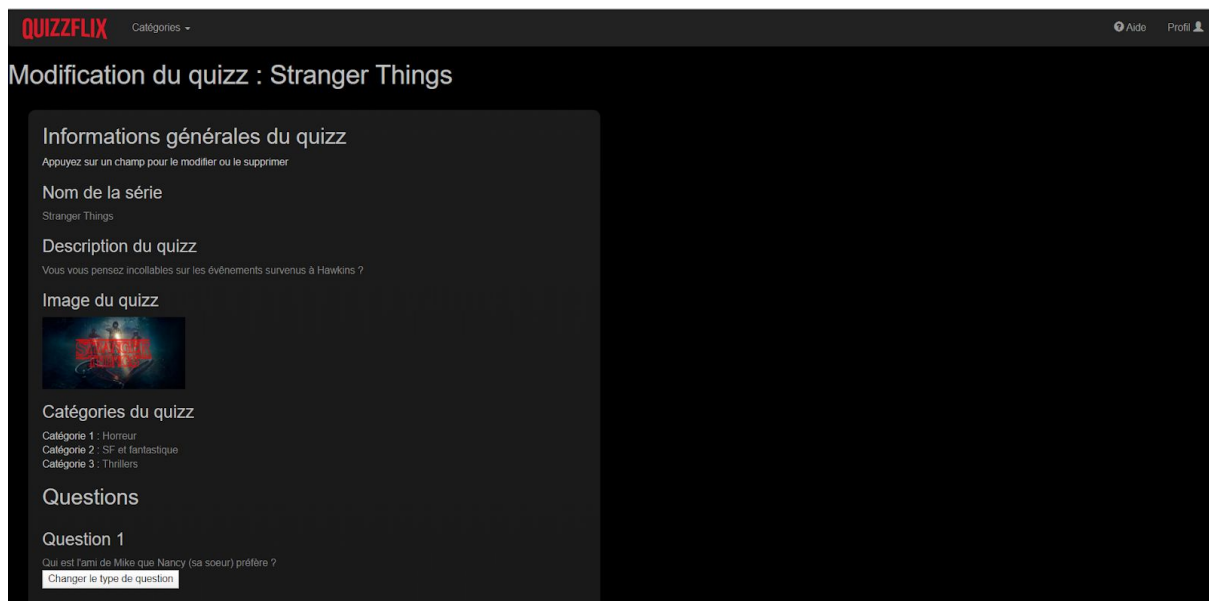
Pour la présentation, nous avons décidé d'opter pour quelque chose de simple mais efficace : une simple liste de balises div en display:flex avec un quizz par balise. Pour chacun de ces quizzes, deux actions sont possibles. Les supprimer, ou accéder à la page de modification du quizz en question. Pour le premier cas, un checkbox est présent sur chaque bloc pour pouvoir permettre à l'administrateur de sélectionner tous les quizzes qu'il souhaite. Il peut alors appuyer sur le bouton supprimer, ce qui les supprime en même temps. C'est pratique lorsqu'il souhaite en supprimer beaucoup en même temps, car il n'a qu'une actualisation de page à subir. Malheureusement, dans le cas où le nombre de quizz est élevé, il peut être moins ergonomique de devoir sélectionner le quizz qu'il veut supprimer et ensuite de descendre tout en bas de la page pour accéder au bouton de suppression.



Un menu déroulant est donc accessible pour chaque quizz depuis lequel l'administrateur peut choisir de supprimer ou de modifier le quizz.

Modification de quizz :

Après avoir appuyé sur le bouton ou sur le nom du quizz, l'administrateur est amené sur la page **modifQuizz.php**.



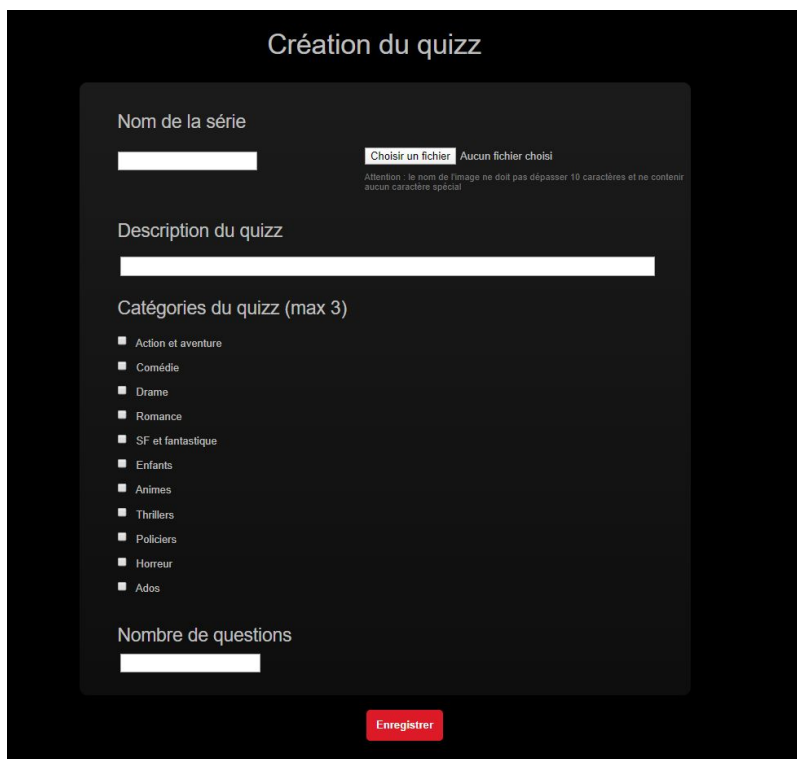
Elle comporte une liste de tous les attributs modifiables de la table quizz, puis une liste de toutes les questions du quizz et de toutes leurs réponses. Nous avons préféré un listage de tous ces éléments sur un script plutôt que plusieurs pour rendre la prise en main de la fonctionnalité facile pour l'administrateur. Pour chaque élément, son champ peut être modifié en cliquant dessus. Nous arrivons alors sur le script **modifElement.php** via un lien et une méthode get. Ce script prend donc une forme adaptée (grâce aux paramètres get) à l'élément qui doit être modifié et à la requête qui est spécifiée. Par exemple, pour les questions et les réponses, si nous cliquons sur le bouton à proximité d'eux, le script nous propose de modifier leurs autres attributs.

A gauche le script propose la modification d'un champ de texte (ex: le titre du quizz) et à gauche soit le changement des types de questions selon le niveau de difficulté pour une question, soit suppression. Nous pouvons aussi supprimer des questions via ce script. Pour la création des questions et des réponses, elles sont aussi faites via une méthode get mais cette fois-ci sur un autre script, **ajouterElement.php**, pour ne pas surcharger le script cité précédemment. Ces deux scripts proposent en fait un formulaire adapté aux requêtes de l'administrateur. Une fois validé, l'utilisateur est amené sur **validationModifElement.php** s'il a voulu modifier

un élément, **validationAjouterElement.php** s'il a voulu en ajouter, ou **validationSupprElement.php** s'il a voulu en supprimer. Ces 3 scripts préparent et exécutent les requêtes sql. Nous avons décidé de les séparer car ils doivent pouvoir traiter l'ensemble des éléments qui caractérisent un quizz. Un seul script aurait été donc trop chargé. Ces scripts informent l'administrateur que la requête a été effectuée sans problème et propose via un bouton de revenir sur la page **modifQuizz.php**.

Création de quizz :

La création de quizz se fait via un glyphicon en forme de plus en haut de la page d'accueil des admins. Il emmène l'administrateur sur la page de **creationQuizz.php**.



The screenshot shows a web form titled "Création du quizz" on a dark background. The form contains several input fields and a list of categories. At the bottom, there is a red "Enregistrer" button.

Création du quizz

Nom de la série

Choisir un fichier | Aucun fichier choisi
Attention : le nom de l'image ne doit pas dépasser 10 caractères et ne contenir aucun caractère spécial

Description du quizz

Catégories du quizz (max 3)

- ☐ Action et aventure
- ☐ Comédie
- ☐ Drame
- ☐ Romance
- ☐ SF et fantastique
- ☐ Enfants
- ☐ Animes
- ☐ Thrillers
- ☐ Policiers
- ☐ Horreur
- ☐ Ados

Nombre de questions

Enregistrer

Elle correspond à tous les attributs de la table quizz. Nous avons décidé de faire la création de quizz en 4 scripts, 3 pour chacune des tables qui caractérisent un quizz, à savoir, quizz, question, réponse, dans l'ordre, et un dernier qui prépare exécute les requêtes sql. La raison est que la création des champs du formulaire à remplir par l'administrateur pour la table réponse nécessite les informations de la question de la table question associée. Nous avons aussi demandé à l'administrateur le nombre de questions qu'il souhaitait créer dans le premier script. Cette information est nécessaire pour créer le formulaire de la table question. Sur ce script est présent notamment l'upload de l'image qui sera par la suite associée au quizz. L'administrateur arrive ensuite sur la page du script **creationQuestions.php**.

Création du quizz : Peaky Blinders

Attention ! Si vous voulez créer une question de type Vrai/Faux, vous devez sélectionner 'Vrai/Faux' pour tous les niveaux de difficulté.
Pour une même question, vous ne pouvez pas non plus choisir une question de type ouverte, puis une question de type QCM pour un niveau plus élevé.

Question 1

Question

Niveau facile

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Niveau moyen

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Niveau difficile

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Question 2

Question

Niveau facile

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Niveau moyen

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Niveau difficile

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Question 3

Question

Niveau facile

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Niveau moyen

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Niveau difficile

- ☐ QCM
- ☐ Vrai/faux
- ☐ Question ouverte

Via une méthode post, autant de champs de questions à remplir que l'administrateur a choisi de questions sont affichés. Le type de celles-ci selon les niveaux de difficulté est traité à ce moment-là. C'est aussi dans ce script que le contrôle de l'image est effectué : contrôle de si l'upload a été réalisé, du type mime du fichier, de sa taille. Le script suivant est **creationReponses.php**, dans lequel un formulaire adapté à chaque question selon son type est proposé.

Création du quizz

Question 1 : Quel est le nom du personnage principal ?

■

■

■

■

■

■

Question 2 : Combien a-t-il de frères et soeurs ?

■

■

■

■

Le dernier script, **validationCreationQuizz.php**, contient toutes les requêtes pour créer les éléments dans les 3 tables en mêmes temps. Nous avons préféré cela par rapport au fait de rentrer les informations dans la base de données au fur et à mesure pour ne pas subir le cas où l'administrateur quitterait la création avant la fin. Ceci impliquerait un affichage sur l'écran d'accueil Quizzflix de quizzes qui en réalité n'est associé à aucune question et/ou réponses.

Plan de test

Interface joueur

Les aspects fonctionnel et esthétique du site ont été testés tout au long de la période de codage en local avec local host. À la mise en ligne de la première version du site nous avons vu apparaître certaines erreurs (code html avant un header, `session_start()` mal positionné...), que nous avons donc pu modifier par la suite.

Nous avons testé le site sur différents navigateurs (Google Chrome, Microsoft Edge, Safari), ainsi que sur différents appareils (ordinateurs sous Windows avec des écrans de différentes tailles, iPhone). Cela nous a surtout aidé pour l'aspect esthétique du site. Par exemple sur Safari nous avons des problèmes pour afficher les boutons, que nous avons pu résoudre en changeant de méthode.

Le fait de tester sur des écrans de plus ou moins grande taille a aussi permis de déceler des problèmes. À l'origine l'accueil du site se présentait de la même manière que les pages catégorie, avec tous les quiz à la suite, et une disposition en "flex-wrap". Seulement ensuite nous avons structuré l'accueil avec des carousels, mais en laissant la disposition du flex-wrap dans le css. Sur un écran d'ordinateur de 14 pouces il n'y avait de la place que pour un carousel par ligne, donc l'erreur passait inaperçue. C'est en testant sur un plus grand écran que l'on a pu s'en rendre compte.

De plus nous avons demandé à quelques personnes de se rendre sur le site et de le tester, pour avoir des retours, et pour déceler les erreurs que nous n'avions pas déjà perçues.

Interface administrateur

Du côté administrateur, nous avons au début tout simplement testé l'efficacité de la création, suppression, et modification des quiz en allant regarder dans la base de données si les attributs étaient bien modifiés comme ils le devaient. Nous avons en plus dû tester un maximum de combinaisons possibles pour les créations et modifications de quiz. Effectivement, nous avons imposé à l'administrateur des règles concernant les quiz, comme par exemple le fait qu'il ne puisse pas avoir pour une question un qcm sur un niveau plus difficile qu'un niveau où il y a une question ouverte (la question ouverte est plus dure que le qcm). Par conséquent nous devons vérifier entre chaque passage d'un script à un autre lors de la création des quiz que les formulaires ne pouvaient pas être remplis selon nos critères. Par exemple, lors du choix des types de questions pour les niveaux de difficultés pour les questions, nous avons essayé toutes les combinaisons possibles pour vérifier que les stratégies d'interception des cas indésirables que nous avons mises en place fonctionnaient. Concernant les requêtes sql, nous avons essayé de rentrer dans les formulaires des bouts de code php pour vérifier qu'elles interceptaient ces cas indésirables et que le site ne pouvait pas se faire pirater. Pour ce qui est de l'upload de l'image sur le site, nous avons essayé aussi d'uploader des fichiers php pour être sûr que le système de contrôle du type du fichier fonctionne bien et que nous ne puissions pas nous faire pirater.

Plus généralement, nous avons validé des formulaires avec des champs vides pour vérifier que dans ce cas les contrôles que nous avons effectués fonctionnaient.

Conclusion

Etant donné les circonstances dues à l'épidémie, notre gestion de projet a été impactée. Malgré le délai supplémentaire nous avons eu des difficultés pour nous organiser en cette période de confinement. De plus nous avons subi des problèmes de matériel au début du confinement.

Nous identifions certaines faiblesses de notre site qui auraient pu être améliorées avec un peu plus de temps. Dans l'historique nous stockons le score effectué par le joueur sous la forme du nombre de bonnes réponses qu'il a eu au quizz, sans prendre en compte le nombre de questions total. Et dans la page de profil nous affichons les meilleurs score du joueur en récupérant avec une requête SQL les 5 plus grands scores. Alors un 9/10 est en fait considéré meilleur qu'un 8/8. Nous nous sommes rendu compte de cette erreur tardivement car nous n'avions que des quizz de 10 questions au début, puis nous en avons rajouté un de 8 questions après. Pour corriger ce problème nous pourrions stocker le rapport du nombre de bonnes réponses sur le nombre de questions total à la place du nombre de bonnes réponses seulement. De cette manière nous pourrions comparer les rapports. Néanmoins ce changement impliquerait des changements dans la base de données et dans le code, et nous nous sommes rendu compte du problème trop tardivement pour avoir le temps de rectifier l'erreur.

D'autres problèmes techniques plus important ont été identifiés trop tardivement concernant les fonctionnalités de suppression en tant qu'administrateur du site. Nous avons mal prévu les choses pour qu'un administrateur puisse supprimer par exemple une question en plein milieu d'un quizz. Il sera possible pour l'administrateur de la supprimer, mais ensuite le quizz ne va plus fonctionner à cause de la manière dont nous avons pensé initialement. La fonctionnalité de suppression est donc opérationnelle mais ne peut être utilisée sans impacter le reste du site que si l'on supprime les choses à la fin (dernières questions, derniers quizz).

De plus certaines erreurs peuvent parfois apparaître sur le site, par exemple quand l'administrateur supprime tous les quizz d'une catégorie, mais en général ce cas de figure ne devrait pas se produire.