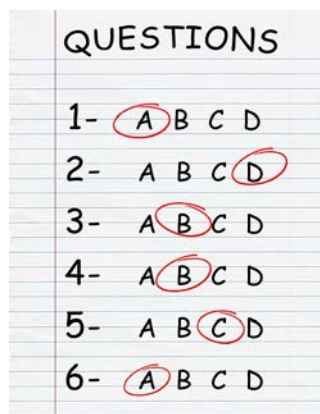


Application web ludique pour jeunes enfants



Harrar Lyes

Sous la direction de M. Lardeux Frédéric

Table des matières

INTRODUCTION	1
CAHIER DES CHARGES	2
1. Présentation.....	3
1.1. Choix du langage.....	3
1.2. Fonctionnement du quiz	4
1.3. Répartition des tâches.....	5
1.3.1. Semaine 1 et 2 : fonctionnement du quiz et planning.....	5
1.3.2. Semaine 3 : base de données	5
1.3.3. Semaine 4 : page d'accueil et page du quiz	6
1.3.4. Semaine 5 : tableau des scores, timer	6
1.3.5. Semaine 6 : l'ajout de questions par l'utilisateur et compte administrateur	6
1.3.6. Semaine 7 et 8 : derniers ajustements et design.....	6
1.4. Répartitions des tâches entre nous	6
2. Solutions adoptées	7
2.1. Le fonctionnement du quiz.....	7
2.1.1. Choix effectués	7
2.1.2. Difficultés rencontrées et solutions apportées	7
2.2. Base de données	8
2.2.1. Choix effectués	8
a) La table « Quiz »	8
b) La table « Nquiz »	9
c) La table « Paramètre ».....	10
d) La table « Score »	10
2.2.2. Difficultés rencontrées et solutions apportées	10
2.3. Tableau des scores	11
2.3.1. Choix effectués	11
2.3.2. Difficultés rencontrées et solutions apportées	11
2.4. Ajout de questions par l'utilisateur.....	11
2.4.1. Choix effectués	11
2.5. Compte administrateur	12
2.5.1. Choix effectués	12
a) La page « ajouter une question »	13
b) La page « toutes les questions »	14
c) La page « valider une question ».....	14
d) La page « paramètre du quiz »	15
3. Quelques points précis à aborder	15
3.1. L'aspect de l'application web	15
3.2. La sécurité.....	16
3.2.1. Eviter les injections SQL.....	16
3.2.2. Côté utilisateur et administrateur	16
3.2.3. Redirection	16
3.2.4. Les commentaires	Erreur ! Signet non défini.
CONCLUSION	17
TABLE DES ILLUSTRATIONS.....	19

Introduction

Ce document a pour but de décrire le déroulement de notre projet dans son intégralité.

Pour notre projet de fin d'année, nous devions faire une sélection des sujets qui nous intéressaient le plus parmi une liste de 29 sujets posés. Ayant un penchant pour le développement web en général (front et back end), nous avons sélectionné des sujets en lien avec notre préférence. Après cela, le sujet qui nous a été attribué fût la confection d'une application web ludique pour jeunes enfants. Ce sujet nous a plu et nous a motivé car de notre point de vue l'apprentissage peut se faire de mille façons, mais une des plus efficace est par le jeu car l'assimilation de la connaissance se fait presque à son insu, en offrant une sensation d'accomplissement et de réussite bénéfique pour la personne.

Dans ce rapport, vous allez voir dans un premier temps comment s'est articulée la réalisation de ce projet en fonction du cahier des charges, notre manière d'aborder le sujet, puis l'organisation autour de celui-ci. Dans un deuxième temps nous décrirons le fonctionnement de l'application web, mais nous traiterons aussi les différentes problématiques en précisant les difficultés ainsi que les solutions apportées. Et dans un troisième temps nous verrons ce que ce projet nous a permis d'accomplir.

Nous espérons que vous prendrez autant de plaisir à lire ce rapport que nous en avons pris durant tout le processus d'élaboration de ce projet.

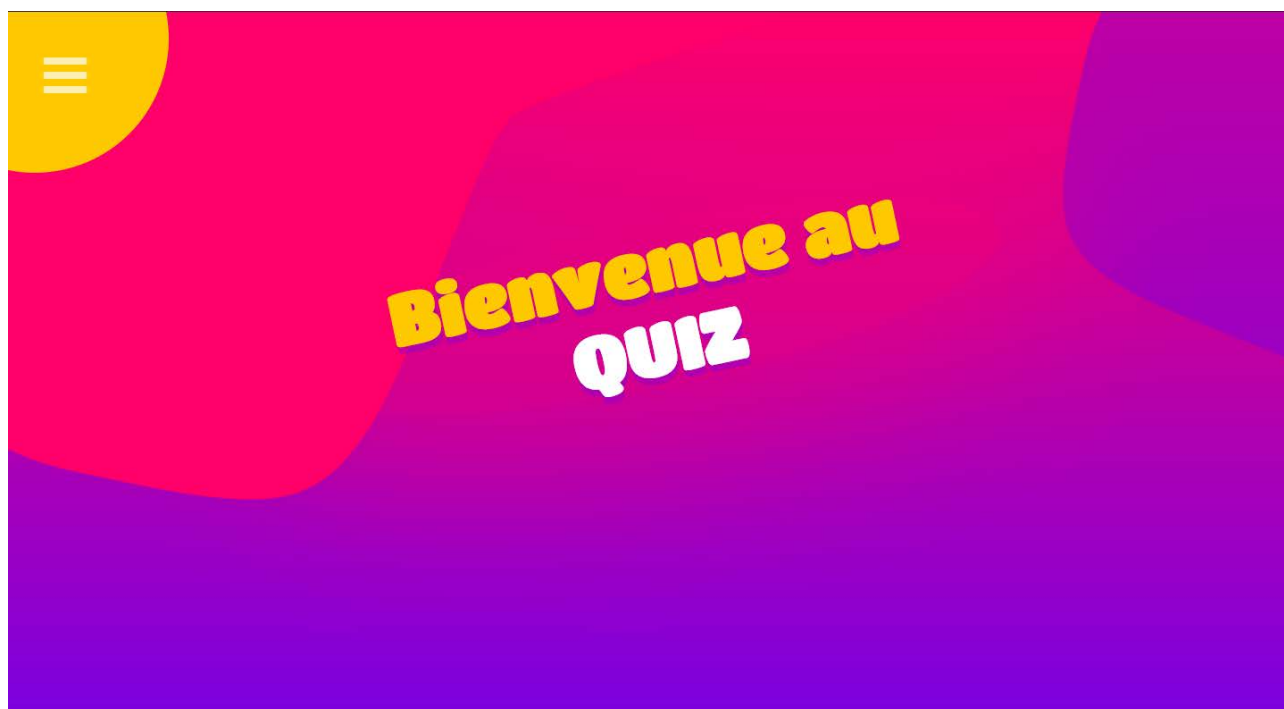


Photo 1 : page d'accueil du quiz

Cahier des charges

Un cahier des charges a pour fonction de formaliser un besoin afin que ce dernier soit compris par l'ensemble des acteurs impliqués dans le projet. Dans notre cas précis, le projet est l'application web ludique pour jeunes enfants et les besoins sont les diverses tâches à accomplir. En outre notre rôle sera de :

Réaliser une application web permettant à des jeunes enfants de tester leurs compétences de manière ludique. Plusieurs modes doivent être possibles :

- Choix multiples : Un questionnaire est passé en paramètre avec différentes réponses (dont au moins une bonne). Toutes s'affichent et l'utilisateur doit en choisir une (réponse en un clic !).
- Addition, soustraction, multiplication et division : un certain nombre d'opérations est proposé et l'utilisateur doit donner la bonne réponse.
- Pour évaluer l'utilisateur, le nombre de bonne réponse est à prendre en compte mais le temps est aussi important.

Voici les différents critères principaux que notre projet doit contenir, néanmoins avec M.Lardeux nous avons réfléchi à différents moyens d'étoffer ainsi que d'améliorer notre projet. Nous avons fini par se mettre d'accord sur d'autres fonctionnalités, à savoir :

- La mise en place d'un côté administrateur afin de gérer (ajouter, supprimer ou modifier) les questions, les propositions de réponse, la bonne réponse, le thème, etc.
- Donner la possibilité à l'utilisateur de proposer sa propre question et de l'introduire dans le quiz.
- L'application web doit être légère, épurée, peu coûteuse en terme d'espace de stockage, et simple d'utilisation pour à la fois profiter d'une expérience aussi rapide et intuitive que possible., mais aussi de pouvoir héberger l'application web gratuitement.

1. Présentation

Tout d'abord, il est essentiel de bien comprendre ce qu'il nous est demandé de faire : il est donc primordial d'analyser les termes importants du sujet. La notion d'application web précisée dans le titre de notre projet est à dissocier d'une application native avec des langages propres à chaque téléphone et tablette, une application web nécessite donc un seul développement quel que soit le périphérique, représentant un gain de temps et de compétence considérable, mais également un gain d'argent, néanmoins c'est un point que l'on ne détaillera pas ici. Une application web, contrairement à une application native, bénéficie d'un chargement plus rapide mais également du référencement naturel : en bref, il est bien plus avantageux de développer une application web plutôt qu'une application native. Par la suite, si l'on prend la définition du mot « ludique » dans le cas de notre application web, cela équivaut à une approche amusante, récréative ou divertissante de l'apprentissage, qui d'ailleurs fonctionne très bien lorsqu'elle est utilisée sur des enfants car elle permet de combiner l'utile à l'agréable.

Pour accomplir au mieux notre mission, nous avons réfléchi à la meilleure manière d'aborder le projet.

1.1. Choix du langage

En premier lieu, le HTML pour afficher le contenu de l'application web pour qu'il soit lisible à partir de n'importe quelle plateforme, puis vient le CSS pour mettre en forme le contenu tout en séparant l'information des détails de la présentation, ce qui permet une maintenance plus aisée, d'éviter des duplications et d'utiliser le même contenu avec différents styles pour différents usages. Pour ce qui est de l'interaction et des animations, JavaScript est un incontournable, et nous avons décidé d'inclure la bibliothèque JQuery pour faciliter l'écriture de scripts côté client dans le code HTML. JQuery se présente comme un unique fichier JavaScript peu volumineux contenant toutes les fonctions de base.

Puis vient la problématique de la sécurité des données et de la création de pages dynamiques à l'aide de PHP 7.1.9 sous WampServer qui est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement des scripts PHP sur nos machines et qui permet aussi l'administration web des bases MySQL en utilisant phpMyAdmin.

Nous avons choisi de ne pas utiliser de frameworks PHP après un début de projet avec Laravel car nous avons très vite remarqué qu'une architecture MVC contiendrait non seulement beaucoup plus de fichiers inutiles à la conception de notre application, mais aussi poserait un problème lors de la gestion et de l'hébergement de celle-ci sachant qu'elle n'était pas destinée à des gestionnaires compétents en informatique. En effet, les librairies dont dispose PHP ne sont pas toutes utiles, surtout dans notre cas

où elles peuvent être très coûteuses. De plus, cela permet un hébergement totalement gratuit de l'application web.

1.2. Fonctionnement du quiz

Notre quiz fonctionne de la manière suivante : l'utilisateur arrive sur l'application web. Avant même de commencer, il sera mis au courant du fonctionnement du quiz, c'est-à-dire :

- Il commence le quiz à 0 point.
- Une question comporte toujours 4 réponses mais une seule est juste : s'il répond correctement à la question, il gagne 1 point, mais s'il répond mal, il ne gagne pas de point et passe automatiquement à la question suivante.
- Pour chaque question il y a du temps, initialisé à 30 secondes : l'objectif est de répondre dans le temps imparti, mais si le temps est écoulé, cela compte comme une faute et le quiz passe à la question suivante.
- Le quiz se termine lorsque l'enfant a répondu à toutes les questions.
- Si son résultat fait partie des 10 meilleurs alors il pourra entrer un pseudo, qui sera ensuite inscrit dans un tableau affichant le score des meilleurs utilisateurs avec classement, et dans le cas contraire l'utilisateur est invité à recommencer le quiz s'il le souhaite et retenter ça chance.

Après avoir pris connaissance de ces règles, il peut appuyer sur commencer pour lancer le quiz. Il va alors voir s'afficher la première question portant sur l'un des thèmes suivant : la culture générale, les dessins animés, l'histoire, la géographie, le français, les mathématiques, la réflexion et le sport. Le but du quiz étant de répondre correctement à un maximum de question.



Photo 2 : exemple de l'affichage d'une question

1.3. Répartition des tâches

Afin de traiter une à une chaque fonctionnalité de l'application web, il est essentiel d'instaurer une répartition des tâches pour finir le projet dans les délais prévus, en l'occurrence fin Mai.

Pour optimiser notre temps, nous avons suivi un planning précis expliqué ci-dessous.

1.3.1. Semaine 1 et 2 : fonctionnement du quiz et planning

Nous avons consacré la première semaine à se mettre d'accord sur le fonctionnement de notre quiz, notamment en s'inspirant de quiz déjà existant sur le web.

La deuxième semaine nous avons listé les différentes choses à faire, puis nous les avons répartis pour avoir une tâche précise à traiter par semaine, et cela tout au long du projet (du 1^{er} Avril jusqu'à fin Mai).

1.3.2. Semaine 3 : base de données

Lors de la troisième semaine, nous nous sommes penché tous les deux sur la base de données de l'application web, plus précisément comment nous allons pouvoir structurer notre base ainsi que nos tables pour pouvoir stocker toutes les informations dont nous avons besoin.

1.3.3. Semaine 4 : page d'accueil et page du quiz

Dans cette quatrième semaine, nous nous sommes occupés de la page d'accueil, de ce qu'elle allait contenir, à savoir un lien qui envoie l'utilisateur sur un formulaire d'ajout de question pour proposer lui-même sa propre question, un côté gestion du quiz accessible seulement par l'administrateur, un tableau des scores, etc, puis de la page qui allait afficher le quiz, avec les différentes questions qui s'affichent une à une, les 4 choix de réponses, le timer, etc.

1.3.4. Semaine 5 : tableau des scores, timer

Tout au long de cette semaine nous avons traité le problème du tableau des scores, celui-ci devant afficher dynamiquement le résultat des meilleurs utilisateurs avec leur pseudo. Nous avons aussi traité le timer, le temps de réponse entre chaque question.

1.3.5. Semaine 6 : l'ajout de questions par l'utilisateur et compte administrateur

Durant cette semaine, nous avons réfléchi au moyen de pouvoir laisser la possibilité aux utilisateurs de proposer une question afin de l'ajouter au questionnaire. Puis nous avons travaillé sur la page consacrée à l'administrateur, donc dans un premier lieu pouvoir se connecter en tant qu'administrateur, et dans un second temps pouvoir maîtriser tout le quiz, comme la vérification des questions ajoutées par les utilisateurs, ou bien l'ajout/la suppression/la modification de questions.

1.3.6. Semaine 7 et 8 : derniers ajustements et design

Pour qu'il puisse plaire, il fallait que le quiz suscite l'intérêt, donc nous sommes retournés sur les différents points traités les semaines précédentes afin de peaufiner certains points, et après cela il nous fallait travailler sur le côté attractif de l'application web, donc sur le design et les animations à mettre en place.

1.4. Répartitions des tâches

HARRAR Lyes		Tous les deux
<ul style="list-style-type: none">- Ajout de questions par l'utilisateur- Base de données- L'ajout des questions	<ul style="list-style-type: none">- Design responsive- La partie administrateur- Fonctionnement du quiz- sécurité de l'application	<ul style="list-style-type: none">- Rapport- Règles du quiz- Page d'accueil- Timer- Tableau des scores

2. Solutions adoptées

Pour chacun des points à traiter, nous allons vous annoncer le raisonnement à travers nos choix, les complications rencontrées au fur et à mesure du projet, quelles en sont les causes, mais également les solutions que nous avons trouvées pour régler les problèmes.

2.1. Le fonctionnement du quiz

2.1.1. Choix effectués

La partie 1.2. dédiée au fonctionnement du quiz explique le fonctionnement du quiz final, mais de notre côté ce ne fût pas la première idée qui nous ait venu à l'esprit, donc nous voulions vous en faire part. L'idée principale du projet est de mettre au point un quiz ludique et attrayant pour que les enfants apprennent tout en s'amusant. Au départ nous sommes partis sur un quiz en version QCM avec affichage des questions en une page et un temps imparti pour cocher les bonne réponses, ce fût plus facile à concevoir et à coder, cependant cette manière de faire nous paraissait peu ludique et fortement présente sur internet ou la quasi-totalité des quiz que l'on a consulté adopte cette méthode. Voulant proposer quelque chose de différent à nos jeunes utilisateurs, nous voulions proposer un quiz plus intuitif : c'est-à-dire une question par page avec à chaque fois 4 choix possibles pouvant s'effectuer obligatoirement en un seul clic. Au début dans notre cas nous avons eu l'idée de mettre en place un quiz illimité, c'est-à-dire un quiz qui ne s'arrête pas tant que le joueur n'a pas commis 3 fautes. Toutefois, nous nous sommes renseignés et il s'avère que la grande majorité des quiz présents sur la toile soient des quiz qui proposent un nombre de question fixe, avec parfois plusieurs thèmes disponibles, mais chaque thème propose toujours les mêmes questions. A l'origine, nous pensions avoir trouvé une bonne idée en créant quelque chose de différent, mais après mûres réflexions nous pensons que ce n'est pas le genre de quiz dont les enfants raffoles. Après coup, nous sommes partis sur un quiz affichant 15 questions au hasard parmi toutes les questions dans la base de données de l'application web. Nous avons choisi ce type de quiz car nous voulions tester globalement les connaissances des enfants en proposant un grand nombre de question, au thèmes variés et adaptés à notre cible principale, en l'occurrence de jeunes enfants.

Selon nous, ce type de fonctionnement convient très bien aux enfants qui ont des facilités car ils seront captivés par le quiz, mais les enfants ayant des difficultés pourront, s'ils le souhaitent, recommencer autant de fois qu'ils le veulent et ainsi améliorer leur score.

2.1.2. Difficultés rencontrées et solutions apportées

Il y a eu plusieurs difficultés rencontrées pour coder le quiz de cette manière. Tout d'abord, pour ne jamais afficher la même question dans un quiz donné et ainsi éviter une éventuelle redondance, il a

fallu mettre au point une manière de sélectionner 15 questions au hasard parmi la base de données : c'est ce que l'on a fait en créant une fonction du nom de `randtableau($taille)` qui prends en paramètre un entier et qui retourne un tableau remplie de nombre de 1 jusqu'au nombre d'enregistrements dans la base de donnée sans redondance, et à chaque fois en les distribuant de manière aléatoire tout le long du tableau. Par exemple `randtableau(5)` retournera quelque chose comme `array(1=>35, 2=>8, 3=>132, 4=>24, 5=>40)`, c'est-à-dire que la question 1 sera notre question 35 dans la table ect..., puis on les afficher une à une au fur et à mesure du quiz. Ensuite vient le problème du timer car il y a diverses choses à prendre en compte : déjà le fait que le timer commence à 30 secondes et fini à 0, et si les 30 secondes sont écoulées alors le quiz passe à la question suivante et pénalise le joueur en ne lui octroyant aucun point. De plus, il y a le problème du bouton de rafraichissement de page sur les navigateurs web qui, comme son nom l'indique, rafraîchit les données d'une page web, donc il a fallu prendre en considération ce problème qui n'en était pas un dans notre ancienne version qcm et empêcher le reset du timer même lors d'un rafraichissement de page ce qui s'est fait entre autre en javascript combiné avec des sessions de php ; Globalement nous nous sommes appuyé sur l'horloge du navigateur qui lui ne change évidemment pas lors d'un rafraichissement d'une page, et donc notre timer peut continuer vu qu'il avance en référence au navigateur. Ensuite, comme notre quiz affiche les questions une à une, il a fallu s'occuper de la transition d'une question à l'autre tout prenant en compte en arrière-plan la réponse de l'utilisateur et le calcul de son score tout au long de la session qui s'est fait bien sûr à l'aide des sessions en php et à la gestion des formulaires notamment savoir jonglé entre les méthodes POST et GET qui se transmettent des données entre elles grâc à un input de type « hidden », c'est-à-dire « caché ».

2.2. Base de données

2.2.1. Choix effectués

Concernant la base de données, nous l'avons construit avec l'outil phpmyadmin. Nous avons optés pour un minimum de tables, quitte à ce qu'elles soient des tables « obèses », et ceci pour la simple et bonne raison que nos tables ne contiendront pas plus d'un millier d'enregistrement : seule la table quiz est à 240 lignes. Ce choix s'avère être plus optimal pour le chargement des pages, et cela évite de se connecter à trop de tables et à devoir jongler entre elles.

Pour faciliter l'administration de MySQL de notre projet. Les données et leurs types sont contenus dans les tables suivantes :

a) La table « Quiz »

quiz

Field	Description	Type	Default	Other
id		int(11)		PK
question		VARCHAR(200)		
option1		VARCHAR(200)		
option2		VARCHAR(200)		
option3		VARCHAR(200)		
option4		VARCHAR(200)		
reponse		VARCHAR(200)		
userreponse		VARCHAR(200)		
theme		VARCHAR(200)		

Photo 3 : descriptif de la table quiz

Cette table est la table principale de l'application web, nous allons par conséquent vous l'expliquer un peu plus en détail pour clarifier les choses.

L'id permet d'obtenir avec certitude une entrée unique pour tous les éléments de la table. Si cette colonne n'est pas créée, nous aurions pu avoir des doublons dans notre base ce qui pourrait à terme provoquer des bugs. Petite remarque, c'est avec l'id que le générateur de question fonctionne, en prenant 15 valeurs au hasard et en ne prenant jamais la même.

La question qui, évidemment, stocke la question, et le thème qui définit le thème de la question. Suivi bien sûr de ses 4 options différentes car c'est de là qu'un quiz tire tout son sens, c'est la difficulté propre à ce jeu et qui le rend intéressant.

La bonne réponse ainsi que la réponse de l'utilisateur sont toutes les deux comparées, donc si les deux valeurs sont identiques alors l'utilisateur gagne 1 point, et dans le cas contraire l'utilisateur n'a pas de point.

b) La table « Nquiz »

nquiz

Field	Description	Type	Default	Other
nquestion		VARCHAR(100)		
noption1		VARCHAR(20)		
noption2		VARCHAR(20)		
noption3		VARCHAR(20)		
nreponse		VARCHAR(20)		

Photo 4 : descriptif de la table nquiz

C'est cette table qui va permettre aux utilisateurs de l'application web de pouvoir proposer leurs propres questions. Ce qu'il faut retenir de cette table est que l'utilisateur indique la question qu'il veut ajouter au quiz, et il indique également les choix de réponses et la bonne réponse. Ces données doivent être bien remplies afin que l'administrateur puisse valider la question et l'incorporer dans la base de données.

c) La table « Paramètre »

parametre

Field	Description	Type	Default	Other
nbquestions		int		PK
temps		int		

Cette table sert uniquement à l'administrateur à modifier à sa guise le nombre de questions ainsi que le temps accordé pour répondre à chaque question : c'est donc ici que la difficulté peut être réglée.

d) La table « Score »

score

Field	Description	Type	Default	Other
classement		int		PK
nom		VARCHAR(20)		
point		int		

Photo 6 : descriptif de la table score

La table score contient le nom et les points des meilleurs utilisateurs, sachant qu'il y aura 10 joueurs inscrit dans ce tableau.

Par ailleurs, le classement se fera à l'évidence selon leurs points, le 1^{er} ayant le plus de points et le 10^{ème} ayant le moins de points.

2.2.2. Difficultés rencontrées et solutions apportées

La table « quiz » n'a pas forcément posé de problèmes particuliers. A l'opposé, la table « Nquiz » qui nous permet de stocker l'ajout de questions par les utilisateurs nous a causé problème car il fallait bien mettre les propositions

Photo 5 : descriptif de la table paramètre

quelque part pour pouvoir les traiter : c'est d'ici que la table « Nquiz » prend tout son sens car elle sert donc de table de transition. Les tables « paramètre » et « score » n'ont pas posé problèmes en elles-mêmes, mais il a fallu réfléchir à la manière dont on allait traiter le tableau des scores, celui-ci étant traité dans une autre partie un peu plus tard dans le rapport.

2.3. Tableau des scores

2.3.1. Choix effectués

Le scoring a pour objectif principal l'obtention d'un score élevé, voire le plus élevé possible. Ici, un score élevé signifie un nombre de bonnes réponses élevé en allant le plus vite possible. Notre quiz permet aux meilleurs d'avoir leur nom et leur score inscrit dans le tableau des scores. Cela permet aussi aux joueurs de vouloir s'améliorer tout au long de leur expérience de jeu pour ainsi avoir un meilleur score et pouvoir être affiché dans le tableau des scores aux yeux de tous. C'est un système qui pousse les utilisateurs vers le haut en les incitant indirectement à s'améliorer.

2.3.2. Difficultés rencontrées et solutions apportées

Tout d'abord, un joueur avec un bon score devra noter son nom pour qu'il puisse être ajouté au tableau des scores. Ensuite la petite difficulté est qu'il a fallu trier les utilisateurs selon leur score dans l'ordre décroissant, donc du plus grand au plus petit. Ensuite lorsqu'un nouveau joueur bat le score d'un ancien joueur dans le tableau des scores, le nouveau joueur passe devant l'ancien joueur, et tous les joueurs situés en dessous de l'ancien joueur sont décalés d'une place, et bien évidemment le joueur situé en 10^{ème} place ne sera plus affiché. Si l'utilisateur est déjà dans le tableau des scores qu'il obtient un meilleur score en rejouant au quiz, son score est mis à jour et il grimpe dans le classement, mais dans le cas contraire si son score est moins bon, son meilleur score ne bouge pas. Il y a aussi le fait qu'un joueur ayant un score moins bon que le dernier du tableau des scores ne sera pas affiché, donc nous ne demandons pas à celui-ci d'entrer son nom.

2.4. Ajout de questions par l'utilisateur

2.4.1. Choix effectués

L'utilisateur a la possibilité de proposer à l'administrateur une question via une page contenant un formulaire, cela permet de créer une interaction en faisant participer les utilisateurs au fonctionnement de l'application web et de la faire vivre.

Photo 7 : page d'ajout d'une question par un utilisateur

On peut constater que cette page est facilement compréhensible par un enfant, donc il n'aura aucun problème à faire sa proposition de question. Lorsque l'enfant va valider l'envoi de sa question, celle-ci va s'ajouter à la base de données, plus précisément dans la table nquiz, et qui, plus tard, sera traitée par l'administrateur via la page « valider une question ».

2.5. Compte administrateur

2.5.1. Choix effectués

Pour gérer le quiz, il fallait introduire à l'application web un système administrateur, accessible à l'accueil du jeu afin de se connecter pour ensuite avoir la mainmise sur l'entièreté du quiz.

Pour mieux comprendre son utilité, nous allons analyser pour vous les différentes fonctions du compte administrateur.

Photo 8 : page du compte administrateur protégée par un mot de passe

En premier lieu, pour avoir accès au compte administrateur, il faut connaître le mot de passe qui est évidemment sécurisé grâce à une pratique qui est le hachage de mot de passe pour empêcher le mot

de passe stocké d'être volé si le support de stockage est compromis, donc cette page sert à vérifier si la personne qui essaie de se loguer au compte administrateur en est bien un. Une fois passé cette étape, vous arriverez sur une page d'accueil qui propose un menu avec différentes pages dont il pourras naviguer entre elles sans avoir à se reconnecter car chaque page vérifiera la session de l'administrateur et lui évitera non seulement d'avoir à se connecter à chaque fois mais aussi empêcher tout autres personnes d'y accéder sans être loguée et ce jusqu'à ce que l'administrateur se déconnecte de la session, nous détaillerons les fonctions de chacune des pages dans ce qui suit.

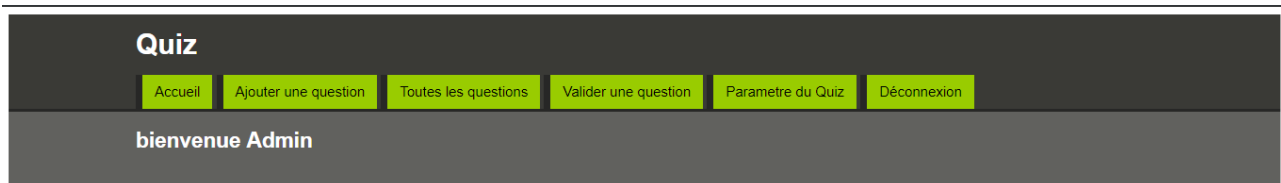


Photo 9 : page d'accueil du compte administrateur

a) La page « ajouter une question »

The image shows the "Ajouter une question..." page in the Quiz application. It features a dark grey header with the word "Quiz" and a horizontal menu with six green buttons: "Accueil", "Ajouter une question", "Toutes les questions", "Valider une question", "Parametre du Quiz", and "Déconnexion". Below the header, the text "Ajouter une question..." is displayed in white. The main content area is a form with several input fields: "Question" (with placeholder "La question ?"), "reponse" (with placeholder "Bonne réponse"), "Choix #1" (with placeholder "Mauvaise réponse 1"), "Choix #2" (with placeholder "Mauvaise réponse 2"), and "Choix #3" (with placeholder "Mauvaise réponse 3"). Below these fields is a dropdown menu labeled "Thème de la question :" with the selected value "culture générale". At the bottom of the form is a "Submit" button.

Photo 10 : page qui permet d'ajouter une question côté administrateur

La page « ajouter une question » est importante car elle permet à l'administrateur de l'application web d'ajouter des questions. Une idée pertinente serait d'ajouter des questions en lien avec ce que les enfants étudient pour leur permettre d'étudier et de tester leur connaissance tout en s'amusant, mais le plus important c'est de faire la différence avec la page « ajouter question » coté utilisateur, qui devait passer les données a une table qui stockerais ces questions et les mettrai en attente d'être validées, dans ce cas l'administrateur ajoute directement la question à la table des questions et donc ne nécessite évidemment aucune validation après ça.

b) La page « toutes les questions »

QUIZ

Accueil

Ajouter une question

Toutes les questions

Valider une question

Parametre du Quiz

Déconnexion

TOUTES LES QUESTIONS DU QUIZ

Nombre de questions : 221

Ici vous pouvez consulter, modifier, ou remplacer les questions de votre choix

Q.NO	QUESTION	OPTION1	OPTION2	OPTION3	OPTION4	RÉPONSE CORRECTE	MODIFIER
220	Comment s'appelle l'île de Vaiana ?	Mahoutou	Matounerie	Materinerie	Mauritane	Matounerie	Modifier
219	Comment s'appelle le camion dans Cars ?	Mac	Mike	Mack	Mik	Mack	Modifier
218	Quel est le nom du père de Pinocchio ?	Figaro	Gettoni	Geppetto	Getama	Geppetto	Modifier
217	Comment s'appelle le père Simpson ?	Moe's	Homer	Papinou	Fifi	Homer	Modifier
216	Qu'est-ce qu'aiment manger Timon et Pumba	De la viande	Ils ne mangent rien	Des insectes	Des plantes	Des insectes	Modifier
215	Comment s'appelle le singe d'Aladdin	Abu	Fred	Jack	Eric	Abu	Modifier
214	Comment s'appelle le père de Simba dans le roi lion	Scar	Mufasa	Remi	Oliver	Mufasa	Modifier
213	Comment s'appelle la soeur d'Elsa dans la Reine des neiges ?	Jenna	Mack	Merida	Anna	Anna	Modifier
212	Qui dit "Chipper arrête de chipper !"	Violetta	Dora	Emile	Mickey	Dora	Modifier

Photo 11 : page qui affiche toutes les questions du quiz

Cette page sert à voir toutes les questions et leurs réponses respectives stockées dans la base de données afin de pouvoir agir dessus à n'importe quel moment. En outre sur cette page on peut :

- Consulter les questions pour voir si elles sont pertinentes (en prenant en compte les compétences des utilisateurs).
- Modifier les questions ou les réponses proposées si elles comportent des erreurs.
- Remplacer les questions si elles ne sont pas cohérentes avec le niveau des utilisateurs par exemple.
- Savoir exactement combien de questions sont stockées dans notre base de données.

c) La page « valider une question »

Cette page sert à voir les questions proposées par les utilisateurs et qui sont en cours de validation. Globalement si la question est plutôt intéressante alors l'administrateur va pouvoir valider la proposition. Si la question comporte des points à revoir comme une question mal formulée, une bonne réponse qui s'avère être inexacte, des erreurs d'orthographe ou des options de réponses inappropriées, alors la question pourra être modifiée dans la page « toutes les questions ». Dans le cas où la question n'est pas pertinente, la question ne sera pas validée et donc pas insérée dans le quiz. Une fois validée, appuyez sur le bouton « vider le tableau » afin de supprimer la demande de validation.

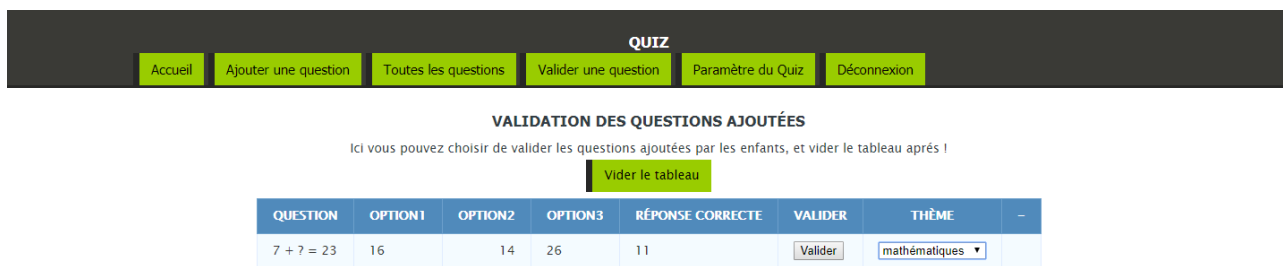


Photo 12 : page qui affiche les questions en attente de validation

d) La page « paramètre du quiz »

Cette page permet à l'administrateur de modifier autant de fois qu'il le souhaite le nombre de questions présentent dans le quiz, mais également le temps fournis pour répondre à une question. Ces paramètres sont idéaux pour rajouter ou bien même enlever de la difficulté au quiz selon le niveau ou les goûts des utilisateurs.

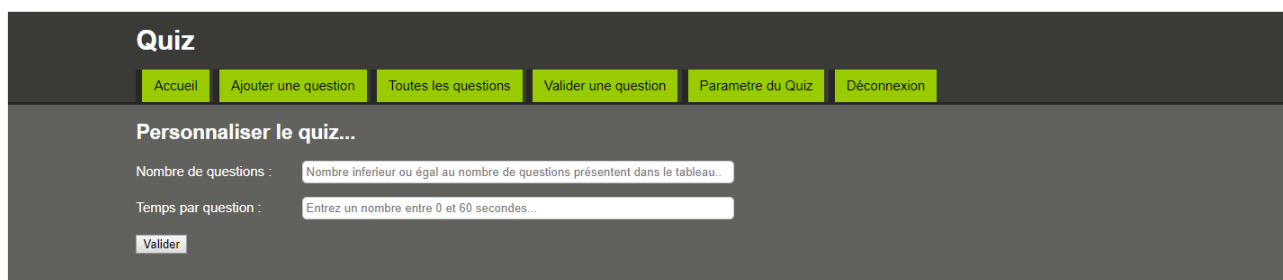


Photo 13 : page qui permet de modifier les paramètres du quiz

3. Quelques points précis à aborder

3.1. L'aspect de l'application web

Pour qu'une application web soit agréable à utiliser et puisse toucher le plus grand nombre de personnes possible, il est nécessaire de prévoir un design responsive, c'est-à-dire un design qui varie selon la résolution et la taille de l'appareil utilisé, cela passe par des redimensionnements, des recadrages, des défilements multidirectionnels de pages, etc. Pour cela, il a fallu que l'on stylise l'application web avec un design fluide pour permettre aux différents blocs contenus de s'adapter à la taille de l'écran facilement et sans bugs d'affichage. Cette méthode prend du temps, mais elle est primordiale afin d'élaborer une application web offrant une expérience de lecture et de navigation optimales pour l'utilisateur.

Pour sublimer notre projet, il a fallu utiliser des couleurs qui attirent l'œil, mais également des animations pour dynamiser et captiver l'utilisateur. Le design a été pensé pour que l'expérience de jeu soit la plus simple et intuitive possible.

3.2. La sécurité

Dans notre projet, nous avons voulu rendre notre code le plus robuste possible et limiter au maximum les risques de piratage. Pour cela, nous avons utilisé diverses techniques expliquées ci-dessous.

3.2.1. Eviter les injections SQL

Les injections SQL sont parmi les failles les plus répandues et les plus dangereuses. Les risques que représente ce genre de faille sont énormes, spécialement lorsqu'une application web telle que la nôtre possède une zone d'administration qui utilise une base de données MySQL les méthodes utilisés pour prévenir cela et l'utilisation des requêtes préparées et en général des méthodes de l'objet PDO qui va appliquer un filtre pour vérifier le type du paramètre et utiliser sa fonction interne pour le "quote" de façon appropriée pour éviter la plupart des injections SQL. Il va au besoins échapper les caractères spéciaux selon les réglages du pilote PDO actif.

3.2.2. Côté utilisateur et administrateur

D'un point de vue sécurité, il est important de séparer le côté utilisateur du côté administrateur. En effet, la zone administrateur de notre projet est protégée par un mot de passe pour éviter toute mauvaise manipulation, qu'elle soit intentionnelle ou non comme interdire l'accès à des pages dont à travers l'URL ce qui testera la présence d'une ou plusieurs sessions et redirigera en conséquence l'utilisateur comme expliqué ci-dessous.

3.2.3. Redirection

L'URL d'une application web est le vecteur permettant d'indiquer la ressource demandée. Il s'agit d'une chaîne de caractère comprenant plusieurs données spécifiques. L'URL peut par exemple permettre de transmettre des paramètres au serveur en faisant suivre le nom de fichier (une des composantes de l'URL) par un point d'interrogation. En manipulant certaines parties de l'URL, cette pratique permet à un pirate d'amener un serveur web à délivrer des pages web auxquelles il n'est pas censé avoir accès. Pour empêcher cela, il est nécessaire d'effectuer une veille sur les vulnérabilités et appliquer régulièrement les correctifs fournis par l'éditeur du serveur web, concernant la partie administrateur, la présence de la session : « `$_SESSION['admin']` » est vérifiée en début de chaque page et redirigera dans le cas où la session n'existe pas l'utilisateur vers la page de connexion.

Conclusion

Ce projet d'une durée de 2 mois a été une très bonne expérience pour nous à différents niveaux : tout d'abord la réalisation de cette application web a été l'occasion de tester nos connaissances en langage de programmation web, de retranscrire les différentes problématiques données en code. Ensuite cela nous a été profitable d'un point de vue organisationnel car traiter un projet en binôme demande sur moyen terme une assiduité et une grande rigueur tout au long de la réalisation pour mener la mission à bien, cela passe par l'élaboration d'un planning et de s'y tenir rigoureusement afin d'honorer les délais fixés, mais aussi par le partage des tâches et la confiance entre les divers collaborateurs d'un projet. Puis il y a aussi le facteur prise de décision, se dire qu'un choix peut avoir un impact sur l'aboutissement du projet, peser le pour et le contre en consultant l'équipe, évaluer les conséquences pour une prise de décision qui permet d'avancer vers l'objectif final.

Nous avons trouvé cet exercice très enrichissant et il nous permettra à l'avenir d'aborder un projet du même type ou d'une autre envergure avec plus de sérénité et de pragmatisme, tout en ayant un recul suffisant et un panel de solutions plus varié en tête.

Table des matières

INTRODUCTION.....	1
CAHIER DES CHARGES.....	2
1. Présentation	3
1.1. Choix du langage.....	3
1.2. Fonctionnement du quiz	4
1.3. Répartition des tâches.....	5
1.3.1. Semaine 1 et 2 : fonctionnement du quiz et planning	5
1.3.2. Semaine 3 : base de données	5
1.3.3. Semaine 4 : page d'accueil et page du quiz.....	6
1.3.4. Semaine 5 : tableau des scores, timer et chance	6
1.3.5. Semaine 6 : l'ajout de questions par l'utilisateur et compte administrateur	6
1.3.6. Semaine 7 et 8 : derniers ajustements et design	6
1.4. Répartitions des tâches entre nous	6
2. Solutions adoptées	7
2.1. Le fonctionnement du quiz	7
2.1.1. Choix effectués.....	7
2.1.2. Difficultés rencontrées et solutions apportées.....	7
2.2. Base de données	8
2.2.1. Choix effectués.....	8
a) La table « Quiz »	8
b) La table « Nquiz »	9
c) La table « Paramètre »	10
d) La table « Score »	10
2.2.2. Difficultés rencontrées et solutions apportées.....	10
2.3. Tableau des scores	11
2.3.1. Choix effectués.....	11
2.3.2. Difficultés rencontrées et solutions apportées.....	11
2.4. Ajout de questions par l'utilisateur	11
2.4.1. Choix effectués.....	11
2.5. Compte administrateur.....	12
2.5.1. Choix effectués.....	12
a) La page « ajouter une question »	13
b) La page « toutes les questions »	14
c) La page « valider une question »	14
d) La page « paramètre du quiz »	15
3. Quelques points précis à aborder	15
3.1. L'aspect de l'application web	15
3.2. La sécurité.....	16
3.2.1. Eviter les injections SQL	16
3.2.2. Côté utilisateur et administrateur	16
3.2.3. Redirection	16
3.2.4. Les commentaires.....	Erreur ! Signet non défini.
CONCLUSION.....	17
TABLE DES ILLUSTRATIONS	19

Table des illustrations

Photo 1 : page d'accueil du quiz	1
Photo 2 : exemple de l'affichage d'une question	5
Photo 3 : descriptif de la table quiz.....	9
Photo 4 : descriptif de la table nquiz.....	9
Photo 5 : descriptif de la table paramètre.....	10
Photo 6 : descriptif de la table score	10
Photo 7 : page d'ajout d'une question par un utilisateur	12
Photo 8 : page du compte administrateur protégée par un mot de passe.....	12
Photo 9 : page d'accueil du compte administrateur.....	13
Photo 10 : page qui permet d'ajouter une question côté administrateur	13
Photo 11 : page qui affiche toutes les questions du quiz.....	14
Photo 12 : page qui affiche les questions en cours de validation.....	15
Photo 13 : page qui permet de modifier les paramètres du quiz	15

ENGAGEMENT DE NON PLAGIAT

Je, soussigné
déclare être pleinement conscient que le plagiat de documents ou d'une
partie d'un document publiée sur toutes formes de support, y compris l'internet,
constitue une violation des droits d'auteur ainsi qu'une fraude caractérisée.
En conséquence, je m'engage à citer toutes les sources que j'ai utilisées
pour écrire ce rapport.

ENGAGEMENT DE NON PLAGIAT

Je, soussigné
déclare être pleinement conscient que le plagiat de documents ou d'une
partie d'un document publiée sur toutes formes de support, y compris l'internet,
constitue une violation des droits d'auteur ainsi qu'une fraude caractérisée.
En conséquence, je m'engage à citer toutes les sources que j'ai utilisées
pour écrire ce rapport.

**Cet engagement de non plagiat doit être signé et joint
à tous les rapports, dossiers, mémoires.**

Présidence de l'université
40 rue de rennes – BP 73532
49035 Angers cedex
Tél. 02 41 96 23 23 | Fax 02 41 96 23 00