

TRAVAIL 2 DÉVELOPPEMENT WEB 3

SECTION 1 : 5% (6 NOVEMBRE)

SECTION 2 : 20% (13 DÉCEMBRE)

Ce travail possède 2 sections avec 2 remises différentes.

CONTEXT DE RÉALISATION

- Le travail doit être réalisé en équipe de 3 ou 4 personnes.
- L'équipe est libre d'utiliser la structure de son choix, mais l'application doit utiliser une forme MVC et être bien structurée.

MISE EN SITUATION

Une entreprise souhaite mettre en place un système de défi évasion (escape room) sur le web. Le système doit contenir les fonctionnalités suivantes :

1. En tant que maître de jeux, je souhaite être en mesure de consulter un menu de gestion afin de configurer mon défi évasion.
 - Ce menu ne doit pas être visible par personne d'autre que le maître de jeux.
 - Le menu doit contenir : création d'énigmes, gestion d'énigme.
2. En tant que maître de jeux, je souhaite être en mesure de créer des énigmes afin de créer mon défi évasion.
 - La page d'ajout doit être accessible par le menu de gestion.
 - Une énigme doit pouvoir(optionnel) contenir du texte qui sert d'indice et guide le joueur.
 - Une énigme doit pouvoir(optionnel) contenir 1 image du format .jpg et seulement jpg.
 - Une énigme doit avoir(obligatoire) une réponse (texte).
 - Il doit y avoir une confirmation que l'ajout de l'énigme a bien fonctionné.
 - Après un ajout fonctionnel, on doit rediriger vers la page de gestion des énigmes.
3. En tant que maître de jeux, je souhaite être en mesure de supprimer une énigme afin de retirer des énigmes qui ne conviennent plus à mon défi évasion.
 - La suppression doit être accessible à l'aide d'un bouton dans la page de gestion des énigmes.
 - Chaque énigme doit avoir un bouton de suppression associé.
 - Une confirmation doit être demandée pour permettre à l'utilisateur d'annuler une suppression.
4. En tant que maître de jeux, je souhaite être en mesure de définir l'ordre des énigmes afin de donner une structure à mon défi évasion.
 - La définition de l'ordre doit se faire dans la page de gestion des énigmes.
 - Il ne peut pas y avoir d'ambiguïté dans l'ordre.
5. En tant que maître du jeu, je souhaite lancer le défi afin d'attendre le moment opportun pour lancer le défi.

- Les participants ne peuvent pas consulter la première énigme avant le lancement.
 - Le démarrage commence le compte à rebours qui débute avec 1heure.
6. En tant que participant, je souhaite visualiser les indices énigmes dans l'ordre défini afin d'être en mesure de réaliser le défi évasion.
- Il est impossible de visualiser l'énigme suivante sans avoir résolu l'énigme courante.
 - Il est toujours possible de consulter les indices des énigmes passés et résolus pour permettre la réutilisation d'indice.
7. En tant que participant, je souhaite voir, et ce en tout temps, le temps restant pour compléter le défi d'évasion afin d'adapter ma stratégie. (Compte à rebours)
- Le compte à rebours se rafraichit 1 fois par 5 secondes.
 - Le compte à rebours fonctionne sans le rafraichissement de la page.
 - Le compte à rebours doit être le bon même si une équipe se connecte en retard.
 - Le compte à rebours doit s'afficher sous le format :
 - 1:00:00
 - 0:59:59
 - Le compte à rebours doit afficher 1:00:00 en vert lorsque la partie n'est pas lancée.
 - Le compte à rebours doit afficher le temps restant en rouge lorsque la partie est lancée.
8. En tant que participant, je souhaite que ma progression soit automatiquement sauvegardée afin de ne pas perdre ma progression si une perte de connexion survient.
- Suite à une reconnexion, l'équipe peut reprendre à l'énigme à laquelle elle était.
9. En tant que participant, je souhaite voir la progression de mon équipe et de mes adversaires (autre équipe) afin d'avoir une compétition intéressante.
- On doit voir visuellement la progression. (Nombre d'énigmes résolues/Nombre d'énigmes total) pour l'ensemble des équipes.
 - Cette progression doit se mettre à jour automatiquement sans rafraichissement de la page, et ce toutes les 10 secondes.
 - On doit voir le nom de l'équipe associé à chaque progression.
10. En tant que participant, je souhaite que d'autres personnes ne puissent pas interférer avec ma session de jeu afin de ne pas voir mon expérience ruinée par des personnes mal intentionnées.
- Le compte de l'équipe est protégé par une authentification qui permet une connexion avec un email et mot de passe.
 - La session expire après 30 minutes.
 - Il est possible de se déconnecter.
11. En tant que participant, je souhaite être en mesure de soumettre mes réponses afin de progresser dans le défi et vérifier si j'ai la bonne réponse.
- Il ne doit pas être possible de soumettre plus d'une réponse par 5 secondes.
 - Une mauvaise réponse doit afficher un petit message générique.
 - Une bonne réponse doit afficher « Click ».
 - La soumission d'une réponse ne doit pas régénérer la page.
 - La soumission d'une bonne réponse doit afficher avec une petite animation un bouton qui dit : « Voir l'énigme suivante ».
 - La réponse est soumise pour l'équipe actuellement connectée.
 - Il est impossible de soumettre une réponse pour une énigme non disponible.
 - Il est impossible de soumettre une réponse pour une énigme déjà réussie.
 - Il est impossible de soumettre une réponse avant le début du défi.
 - Il est impossible de soumettre une réponse si le temps est complètement écoulé.

12. En tant que participant, je souhaite être en mesure d'inscrire mon équipe afin de participer au défi évasion.
- Le formulaire doit être accessible sans connexion.
 - L'équipe doit choisir un nom.
 - Le nom doit être d'une longueur minimale de 3 caractères et maximale de 16 caractères.
 - L'équipe doit définir un email.
 - L'email doit avoir une nomenclature valide.
 - L'équipe doit choisir un mot de passe.
 - Le mot de passe doit contenir un chiffre.
 - Le mot de passe doit contenir une lettre
 - Le mot de passe doit contenir un symbole parmi : !/\$%?&*()
 - L'équipe doit confirmer le mot de passe.
 - La confirmation et le mot de passe doivent correspondre.

SECTION 1 6 NOVEMBRE 5%

La section porte surtout sur les interfaces. L'utilisation de bootstraps Version 4 ou Material design est obligatoire.

MODALITÉ DE REMISE

Votre travail devrait être remis sous format .zip sur LÉA sous l'énoncé de travail « Travail 2 section1 ». La remise doit avoir lieu **avant le 6 novembre 23h59.**

TRAVAIL REQUIS

- Mettre en place le projet (évalué dans la section2)
- Mettre en place un framework CSS.
- Déterminer les interfaces requises.
- Déterminer les URL à utiliser.
- Produire le HTML de ces interfaces. Vous devrez mimiquer les données, car à ce point vous n'avez les données requises pour un affichage authentique. Toutefois, assurez-vous de produire des vues réutilisables.
- Réfléchir la validation(client) des différentes interfaces.
- Effectuer des captures d'écrans des interfaces.
- Remettre un document Word contenant l'ensemble des captures d'écrans.
- Un titre ou une explication doit dire pour quelle story la capture est destinée et quelle URL permettra de l'obtenir. (Préciser également la méthode si nécessaire)
- Les formulaires doivent être suivis d'un tableau qui énonce les validations(client) qui seront mises en place.
- Vous n'avez pas à fournir de capture des messages de confirmations.

CRITÈRE DE CORRECTION

- Le document est propre et bien structuré. (1 point)
- L'ensemble des interfaces requises sont présentes dans le document. (2 points)

- Les captures d'écrans sont identifiées correctement. (1 point)
- L'utilisation des différentes URL est judicieuse (2 points)
- Les interfaces respectent les critères de Bastien et Scapin. (2 points)
- La validation des formulaires est appropriée. (2 points)

PÉNALITÉ

- L'équipe n'utilise pas bootstrap ou Material design.

SECTION 2 13 DÉCEMBRE 20%

MODALITÉ DE REMISE

Votre travail devrait être remis sous format .zip sur LÉA sous l'énoncé de travail « Travail 2 section2 ». La remise doit avoir lieu **avant le 13 décembre 23h59**. **Prévoir du temps pour la présentation de votre projet sur Teams.**

TRAVAIL REQUIS

MCD, GÉNÉRATION DE LA BASE DE DONNÉES

Pour la mise en situation vous devez produire le MCD et faire le script SQL pour la génération de la base de données. Le script devrait être inclus dans votre dépôt git et contenir l'ensemble du code nécessaire pour mettre en place la base de données tous les environnements (qu'il soit nouveau ou non). Ceci inclut :

- La création des tables
- La création des clés étrangères avec leur restriction
- La création des procédures stockées
- L'insertion de données pour tester et faire la démonstration de l'ensemble de l'application est faite.

RÉALISER LES STORIES

- Réaliser les stories en respectant les critères d'acceptation.
- Utiliser une architecture MVC pour structurer le code.
 - Vous pouvez utiliser une approche MVC complètement différente de ce que nous avons fait dans le cadre du cours.

EFFECTUER LA GESTION DES ERREURS

- Afficher une page « introuvable » lorsque l'URL n'est pas valide.
- Rediriger judicieusement vers une page « Vous n'avez pas cette autorisation » ou vers une page de connexion lorsque l'utilisateur n'est pas connecté ou qu'il ne dispose pas des droits.
- Effectuer une validation(serveur) des différentes requêtes.

GESTION D'AUTHENTIFICATION

- Effectuer les vérifications nécessaires (côté serveur) pour effectuer des opérations.

SÉCURITÉ

- Protéger l'application contre la faille XSS.
- Protéger l'application contre l'injection SQL.
- Effectuer une la gestion sécuritaire pour la validation du mot de passe

AJAX

- Utiliser le AJAX lorsque prescrit.

CRITÈRE DE CORRECTION TOTAL 109 POINTS

- MCD et génération de la base de données
 - La base de données est normalisée. (5 points)
 - La base de données contient l'ensemble des informations nécessaires pour le fonctionnement de l'application.
 - Les procédures stockées requises sont présentes. (5 points)
 - Des données de test sont insérées automatiquement. (Pénalité si ce n'est pas fait jusqu'à -10 points)
- Les stories sont complétées et respectent les critères d'acceptation (total : 46 points). Des points seront retirés des stories si la gestion d'erreur y est déficiente.
 - #1 : 1 points
 - #2 : 8 points
 - #3 : 2 points
 - #4 : 5 points
 - #5 : 2 points
 - #6 : 3 points
 - #7 : 5 points
 - #8 : 2 points
 - #9 : 5 points
 - #10 : 2 points
 - #11 : 8 points
 - #12 : 3 points
- Effectuer la gestion des erreurs
 - Afficher une page « introuvable » lorsque l'URL n'est pas valide. (3 points)
 - Rediriger judicieusement vers une page « Vous n'avez pas cette autorisation » ou vers une page de connexion lorsque l'utilisateur n'est pas connecté ou qu'il ne dispose pas des droits. (3 points)
 - Effectuer une validation(serveur) des différentes requêtes. (8 points)
- Gestion d'authentification
 - L'équipe utilise des variables de session pour conserver les informations de connexion ainsi que le rôle d'un usager. (5 points)
- Sécurité
 - Il n'y a pas de faille XSS. (3 points)
 - Il n'y a pas de possibilité d'injection SQL. (3 points)
 - Le nécessaire pour la vérification du mot de passe est conservé dans la base de données d'une manière sécuritaire. (3 points)

- AJAX
 - Utilisation judicieuse et efficace du AJAX (5 points) également évalué par certains critères d'acceptation.
- Qualité du projet (20 points)
 - Respect des responsabilités des objets en MVC.
 - Le code est structuré avec la philosophie objet.
 - Utilisation du polymorphisme et de l'héritage si pertinent.
 - Méthodes courtes (8-12 lignes max). (
 - Nom de variable significatif.
 - Nomenclature constante
 - Pas de mélange français et anglais
 - Snake case ou camel case, mais pas les deux.
 - Exception pour les fonctions en réponse direct aux URL
 - Pas de variable magique.
- Français
 - Les pénalités pour le français s'appliquent comme stipulé par les RDEA.