









## 1 INFORMATIONS GENERALES

Candidat :	Nom : Fernandes		Prénom : Alexandre	
	 : <a href="mailto:alexandre.fernandes2@eduvaud.ch">alexandre.fernandes2@eduvaud.ch</a>		 :	
Lieu de travail :	<input type="checkbox"/> ETML, Rue de Sébeillon 12, 1004 Lausanne <input checked="" type="checkbox"/> CFPV (COFOP-ETML), Avenue de Valmont 28b, 1010 Lausanne			
Orientation :	<input type="checkbox"/> 88601 Développement d'application <input checked="" type="checkbox"/> 88602 Informatique d'entreprise <input type="checkbox"/> 88603 Technique des systèmes			
Chef de projet :	Nom : Costa Lopes		Prénom : Helder	
	 : <a href="mailto:helder.costa@eduvaud.ch">helder.costa@eduvaud.ch</a>		 : 076 442 87 79	
Expert 1 :	Nom : Wolf		Prénom : Benjamin	
	 : <a href="mailto:bw-tpi@hotmail.com">bw-tpi@hotmail.com</a>		 :	
Expert 2 :	Nom : Malherbe		Prénom : Roger	
	 : <a href="mailto:r.malherbe@rmsoft.ch">r.malherbe@rmsoft.ch</a>		 :	
Période de réalisation :	Du <b>jeudi 2 mai 2024 à 13h10</b> au <b>lundi 3 juin 2024 à 13h50</b>			
Horaire de travail :	Lundi	08h00-12h15	13h10-15h45	<i>Pentecôte 20 mai</i>
	Mardi	-	-	<i>Examen CG 27 mai</i>
	Mercredi	08h00-11h25	13h10-16h35	
	Jeudi	-	13h10-16h35	<i>Ascension 9 mai</i>
	Vendredi	08h00-12h15	13h10-16h35	<i>Pont de l'Ascension 10 mai</i>
	<i>Toutes les demi-journées ont une pause obligatoire de 15 minutes.</i>			
Nombre d'heures :	88 heures			
Planning (en H ou %)	Analyse 10 % Implémentation 45 % Tests 10 % Documentation 35%			
Présentation :	Dates retenues : 12 ou 13 juin 2024 semaine 24			

## 2 PROCÉDURE

Le candidat réalise un travail personnel sur la base d'un cahier des charges reçu le 1er jour.

Le cahier des charges est approuvé par les deux experts. Il est en outre présenté, commenté et discuté avec le candidat. Par sa signature, le candidat accepte le travail proposé.

Le candidat a connaissance de la feuille d'appréciation avant de débiter le travail.

Le candidat est entièrement responsable de la sécurité de ses données.

En cas de problèmes graves, le candidat avertit au plus vite les deux experts et son CdP.

Le candidat a la possibilité d'obtenir de l'aide, mais doit le mentionner dans son dossier.

A la fin du délai imparti pour la réalisation du TPI, le candidat doit transmettre par courrier électronique le dossier de projet aux deux experts et au chef de projet. En parallèle, une copie papier du rapport doit être fournie sans délai en trois exemplaires (L'un des deux experts peut demander à ne recevoir que la version électronique du dossier). Cette dernière doit être en tout point identique à la version électronique.

### 3 TITRE :

QuizMaster

## 4 MATÉRIEL ET LOGICIEL À DISPOSITION

### 4.1 Matériel :

- **Ordinateur** : Un ordinateur suffisamment puissant pour exécuter un serveur web local et pour le développement de logiciels.
- **Connexion Internet** : Une connexion Internet pour accéder à des ressources en ligne et pour tester le système de quiz en réseau si nécessaire.

### 4.2 Logiciels :

- **Serveur Web** : Apache, Nginx, XAMPP, WAMP.
- **PHP** : Version récente.
- **Base de Données** : MySQL.
- **Éditeur de Code** : Visual Studio Code, Sublime Text, Atom.
- **Navigateur Web** : Google Chrome, Mozilla Firefox, Microsoft Edge.
- **Terminal ou Invite de Commandes** : Terminal (Mac/Linux), Invite de Commandes (Windows).
- **Client FTP** : FileZilla (Facultatif).
- **Git** : Gestion de Version (Facultatif).

## 5 PRÉREQUIS

Pour créer un système de quiz interactif en PHP, il faut savoir :

- HTML et CSS pour faire une interface utilisateur intuitive
- PHP et MySQL pour gérer les questions et les réponses avec des bases de données
- POO en PHP pour structurer le code

## 6 DESCRIPTIF DU PROJET

Le projet consiste en un système de quiz interactif en PHP. Les utilisateurs pourront faire différents quiz, répondre aux questions et voir leurs résultats tout de suite.

L'application aura une interface utilisateur simple, une base de données MySQL pour garder les données des quiz et une authentification des utilisateurs pour distinguer les administrateurs des utilisateurs ordinaires.

Les fonctionnalités principales seront la création de quiz, le défilement des questions, le calcul du score et l'affichage des résultats.

### 6.1 Fonctions demandées

1. Gestion de la base de données MySQL
  - a. Création de la base de données
  - b. Définition des tables (utilisateurs, questions, réponses)
  - c. Insertion et récupération des données
  - d. Mise à jour des scores des utilisateurs
2. Création de l'interface utilisateur
  - a. Page d'accueil
  - b. Page de connexion/inscription
  - c. Page du quiz
  - d. Page de résultats
3. Intégration PHP-MySQL
  - a. Connexion à la base de données

- b. Exécution des requêtes SQL
  - c. Récupération et affichage des données
- 4. Authentification des utilisateurs
  - a. Inscription (création de compte)
  - b. Connexion (vérification des identifiants)
  - c. Déconnexion
- 5. Calcul du score
  - a. Attribution des points pour les bonnes réponses
  - b. Soustraction des points pour les mauvaises réponses
  - c. Mise à jour du score total
- 6. Affichage des résultats
  - a. Affichage du score à la fin du quiz
  - b. Affichage des bonnes et mauvaises réponses
  - c. Classement des utilisateurs par score

## 7 LIVRABLES

Le candidat est responsable de livrer à son chef de projet et aux deux experts :

- **Rapport** : Ce document, conforme au canevas de l'annexe 3, détaillera l'architecture, les décisions de conception, les solutions aux problèmes rencontrés et les recommandations d'amélioration.
  - **Code Source** : L'ensemble du code source du système de quiz en PHP, incluant les fichiers HTML, CSS et JavaScript pour l'interface utilisateur.
  - **Base de Données** : Le schéma MySQL pour stocker les quiz, questions, réponses et résultats des utilisateurs, avec le script SQL de création des tables.
  - **Journal** : Ce journal combinera la planification initiale du projet et les entrées quotidiennes pour suivre son évolution.

## 8 POINTS TECHNIQUES ÉVALUÉS SPÉCIFIQUES AU PROJET

La grille d'évaluation définit les critères généraux selon lesquels le travail du candidat sera évalué (documentation, journal de travail, respect des normes, qualité, ...). En plus de cela, le travail sera évalué sur les 7 points spécifiques suivants (Point A14 à A20):

Voici cinq points spécifiques qui seront évalués pour ce projet :

1. Analyse du système d'information :
  - Fournir un schéma MCD/MLD conforme à la méthode Merise.
  - Notation :
    - 0 : Schéma absent.
    - 1 : Schéma partiellement conforme.
    - 2 : Schéma conforme.
    - 3 : Schéma complet et conforme.
2. Sauvegarde régulière :
  - Effectuer au moins un commit par jour avec des titres et des descriptions clairs.
  - Notation :
    - 0 : Aucun commit.
    - 1 : Commits rares ou mal documentés.
    - 2 : Commits réguliers mais peu documentés.
    - 3 : Commits quotidiens avec des titres et descriptions claires.
3. Sécurité des données :
  - Utiliser le hachage pour protéger les données sensibles conformément au RGPD/LPD.
  - Notation :
    - 0 : Hachage non utilisé.
    - 1 : Hachage utilisé partiellement.
    - 2 : Hachage utilisé correctement pour la plupart des données.
    - 3 : Hachage utilisé correctement pour toutes les données sensibles.
4. Procédure de test simple :
  - Fournir une procédure simple pour tester la solution.
  - Notation :
    - 0 : Procédure absente.
    - 1 : Procédure complexe ou peu claire.

- 2 : Procédure simple mais incomplète.
  - 3 : Procédure simple et complète.
5. Implémentation des fonctions demandées :
- Vérifier si le candidat a bien implémenté les 6 fonctions demandées.
  - Notation :
    - 0 : Aucune fonction implémentée.
    - 1 : Quelques fonctions implémentées sans respect de la séparation.
    - 2 : La plupart des fonctions sont correctes et respectent la séparation.
    - 3 : Toutes les fonctions sont parfaites et respectent la séparation.
6. Critères : Modularité, réutilisabilité, absence de dépendances croisées.
- Interface responsive :
  - Concevoir une interface utilisateur responsive pour une expérience optimale sur différents dispositifs.
  - Notation :
    - 0 : Interface non responsive.
    - 1 : Interface partiellement responsive.
    - 2 : Interface responsive sur la plupart des dispositifs.
    - 3 : Interface responsive sur tous les dispositifs testés.
7. Gestion des erreurs :
- Mettre en place une gestion efficace des erreurs et des exceptions dans le code.
  - Notation :
    - 0 : Gestion des erreurs non gérée.
    - 1 : Gestion des erreurs partiellement gérée avec des erreurs fréquentes.
    - 2 : Gestion des erreurs bien gérée avec quelques erreurs occasionnelles.
    - 3 : Gestion des erreurs très bien gérée avec très peu ou pas d'erreurs.

**Remarque :**

Le recours à des outils en ligne d'intelligence artificielle (ex. : Chat GPT) doit être mentionné et ne peut servir que d'inspiration à la réalisation. En cas d'abus, l'évaluation du TPI en tiendra compte.

## 9 VALIDATION

	Lu et approuvé le :	Signature :
Candidat :		
Expert n°1 :		
Expert n° 2 :		
Chef de projet :		