









1 INFORMATIONS GENERALES

Candidat :	Nom : PFISTER	Prénom : ANTHONY																				
	 : anthony.pfister@eduvaud.ch	 : 076 589 13 07																				
Lieu de travail :	<input checked="" type="checkbox"/> ETML, Rue de Sébeillon 12, 1004 Lausanne <input type="checkbox"/> CFPV (COFOP-ETML), Avenue de Valmont 28b, 1010 Lausanne																					
Orientation :	<input type="checkbox"/> 88601 Développement d'application <input type="checkbox"/> 88602 Informatique d'entreprise <input type="checkbox"/> 88603 Technique des systèmes																					
Chef de projet :	Nom : SONNEY	Prénom : Gaël																				
	 : gael.sonney@eduvaud.ch	 : + 41 79 629 66 74																				
Expert 1 :	Nom : Berney	Prénom : Daniel																				
	 : daniel.berney@heig-vd.ch	 : +41 79 209 87 93																				
Expert 2 :	Nom : Wolf	Prénom : Benjamin																				
	 : bw-tpi@hotmail.com	 : +41 79 793 34 65																				
Période de réalisation :	Du lundi 8 mai 2023 à 8h au mercredi 7 juin 2023 à 16h35																					
Horaire de travail :	<table border="0"> <tr> <td>Lundi</td> <td>08h00-11h25</td> <td>13h10-16h35</td> <td><i>Pentecôte 29 mai</i></td> </tr> <tr> <td>Mardi</td> <td>-</td> <td>-</td> <td></td> </tr> <tr> <td>Mercredi</td> <td>08h00-12h15</td> <td>13h10-16h35</td> <td></td> </tr> <tr> <td>Jeudi</td> <td>-</td> <td>13h10-16h35</td> <td><i>Ascension 18 mai</i> <i>Examen CG 1^{er} juin</i></td> </tr> <tr> <td>Vendredi</td> <td>08h00-12h15</td> <td>13h10-16h35</td> <td><i>Pont de l'Ascension 19 mai</i></td> </tr> </table> <p><i>Toutes les demi-journées ont une pause obligatoire de 15 minutes.</i></p>		Lundi	08h00-11h25	13h10-16h35	<i>Pentecôte 29 mai</i>	Mardi	-	-		Mercredi	08h00-12h15	13h10-16h35		Jeudi	-	13h10-16h35	<i>Ascension 18 mai</i> <i>Examen CG 1^{er} juin</i>	Vendredi	08h00-12h15	13h10-16h35	<i>Pont de l'Ascension 19 mai</i>
Lundi	08h00-11h25	13h10-16h35	<i>Pentecôte 29 mai</i>																			
Mardi	-	-																				
Mercredi	08h00-12h15	13h10-16h35																				
Jeudi	-	13h10-16h35	<i>Ascension 18 mai</i> <i>Examen CG 1^{er} juin</i>																			
Vendredi	08h00-12h15	13h10-16h35	<i>Pont de l'Ascension 19 mai</i>																			
Nombre d'heures :	89 heures																					
Planning (en H ou %)	Analyse 20%, Implémentation 40%, Tests 10%, Documentation 30%																					
Présentation :	Dates retenues : 14 ou 15 juin 2023																					

2 PROCÉDURE

Le candidat réalise un travail personnel sur la base d'un cahier des charges reçu le 1^{er} 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

Gestion de prêt entre voisins

4 MATÉRIEL ET LOGICIEL À DISPOSITION

- Un PC standard de l'ETML (Windows 10)
- Visual Studio code avec environnement PHP installé
- Serveur web local (uWamp ou autre)
- Suite Microsoft Office pour la documentation
- Un dépôt GIT (GitHub, BitBucket ou autre)

5 PRÉREQUIS

- Connaissances en programmation PHP et en POO (Modules ICT 403, 404, 226, 120, 326, 411, 133)
- Connaissances en modélisation et implémentation de bases de données relationnelles (Modules ICT 104, 105, 153)

6 DESCRIPTIF DU PROJET

Le but de ce projet est de réaliser une application web permettant la gestion de prêt et emprunt entre voisins.

Cette application est destinée à toute personne voulant mettre à disposition et emprunter des articles dans le cadre d'un voisinage. Ces articles sont principalement des objets de bricolage, de jardinage et de cuisine qui ne sont utilisés que temporairement par le propriétaire. Des personnes du voisinage ont la possibilité d'emprunter les objets mis à disposition par un propriétaire. Il y a donc plusieurs utilisateurs qui peuvent être à la fois propriétaire et emprunteur.

Chaque utilisateur peut ajouter des articles et les mettre à disposition. Chaque utilisateur peut emprunter des articles disponibles. Un article est disponible s'il n'est pas emprunté. Chaque utilisateur peut consulter sa liste d'articles prêtés et empruntés.

6.1 Fonctionnalités

Les fonctionnalités présentes dans l'application seront les suivantes :

- Gestion de l'authentification
- Opération CRUD
 - Ajout d'un article
 - Modification d'un article
 - Suppression d'un article
 - Affichage des détails d'un article
- Rechercher un article
- Emprunter un article
- Lister les articles prêtés et empruntés d'un utilisateur
- Mise en place d'un système de notification
- Mise en place de tests automatiques avec Selenium

6.1.1 Gestion de l'authentification

Pour s'authentifier, l'utilisateur doit renseigner son nom d'utilisateur et son mot de passe. Une fois authentifié, l'utilisateur dispose de tous les droits sur l'application. Par contre, un utilisateur non authentifié n'a accès à aucune fonctionnalité. La création d'un compte utilisateur n'est pas demandé dans ce projet. Les utilisateurs seront importés directement dans la base de données (via requêtes SQL ou méthode équivalente, si un framework est choisi par l'élève).

6.1.2 Opérations CRUD

Pour chaque article, un nom, une description et une photo devront être renseignés. Chaque article pourra être disponible ou non disponible. Chaque article est localisé en fonction du lieu d'habitation de son propriétaire.

6.1.3 Rechercher un article

Un utilisateur pourra rechercher un article qu'il voudra emprunter en fonction de son lieu d'habitation. Les articles retournés dans la recherche correspondent au lieu d'habitation de l'emprunteur.

6.1.4 Emprunter un article

Un utilisateur peut emprunter un article en saisissant une date de début et de fin de prêt. Un article prêté devient indisponible.

6.1.5 Lister les articles prêtés et empruntés d'un utilisateur

Un utilisateur peut consulter ses articles mis à disposition et les articles empruntés.

6.1.6 Notifier la fin d'un prêt ou envoyer un rappel de retour

Un système de notification simple doit prévenir la fin d'un prêt. Un bandeau d'affichage sur le profil de l'utilisateur est suffisant.

6.1.7 Tests automatisés avec Selenium

Des tests de non régression seront implémentés pour les principales fonctionnalités.

6.1.8 Ergonomie des interfaces

Des maquettes d'écran devront être réalisées dans l'analyse dans le respect de critères UX (simplicité, cohérence, interaction, crédibilité, etc).

6.2 Points technologiques

6.2.1 Plateforme WEB

Le projet sera réalisé avec une plateforme WEB :

- HTML / CSS / JavaScript coté client
- PHP (version 8)

Si le candidat le souhaite, des frameworks comme **Laravel** et **Tailwind** (css) peuvent être utilisés.

6.2.2 Base de données

Une base de données relationnelle (MYSQL) sera liée à la plateforme et stockera les données liées aux articles et aux prêts.

La modélisation de cette base de données sera effectuée avec la méthode MERISE. Le MCD, MLD et MPD devront être présents dans la documentation du projet.

Les scripts de création de la base de données ainsi que ceux d'imports des données s font partie du livrable.

7 LIVRABLES

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

- Une planification initiale à la fin du premier jour
- Un rapport de projet contenant au minimum :
 - L'analyse du projet (un ou des schémas de principe sont souhaitables pour illustrer les réflexions rédigées).
 - L'analyse devra contenir des informations sur les objectifs, le contenu et le public cible avec les conséquences.
 - La réalisation répondant au cahier des charges (recherches de solutions, fonctionnalités, argumentation sur les choix effectués, obstacles rencontrés, mentions des aides extérieures apportées, etc.), ainsi qu'une charte graphique et une maquette du design. Les schémas de base de données (MCD, MLD, MPD) sont aussi attendus.
 - Les tests planifiés et effectués avec les résultats et analyses de ceux-ci.
 - La conclusion contenant, en outre, des considérations personnelles, des considérations techniques et des considérations de réalisation du projet (comparaison entre ce qui devait être fait et ce qui a été réellement fait, etc.).
- Le rapport de projet en l'état sera livré 2 fois par semaine, le mercredi et le vendredi en fin de journée. Il sera livré sous forme de PDF dans le repo Git dans un dossier dédié à la documentation (/doc par exemple).
- Le journal de travail avec mentions de ce qui a été accompli, les suites à donner chaque fois qu'elles sont envisagées, liens et références des informations collectées ou retranscrites.
- Le code source de l'application et les scripts de base de données (liens vers le dépôt GIT fourni au CDP et aux experts)

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):

1. Choix, modélisation et intégration de la base de données
2. Qualité et lisibilité du code, respect des conventions de nommages standards

3. Les opérations CRUD sur un article (ajout, modification, suppression et afficher les détails d'un article).
4. L'utilisateur doit pouvoir rechercher un article qu'il veut louer, l'emprunter et terminer l'emprunt.
5. L'utilisateur doit pouvoir lister les articles mis à disposition et empruntés et en voir les détails.
6. Un système de notification permettant d'avertir l'utilisateur que la date de fin de prêt est proche.
7. Des tests automatisés sont implémentés avec le framework Selenium.

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 :		