



Elevate

Nom de l'apprenti/-e : Max Langenegger

N° de l'apprenti/-e : 151464

Description générale du projet d'examen :

Une course de marche et vol combine randonnée et parapente. Les participants montent à pied et redescendent en volant, réalisant des épreuves techniques comme des slaloms ou des atterrissages de précision. Cette discipline demande endurance, stratégie et habileté de pilotage.

L'application *Elevate* est une PWA dédiée à la gestion d'une course de marche et vol. Elle permet aux commissaires de course d'enregistrer les performances des concurrents (sommets atteints, slalom, atterrissages) et d'infliger des malus en cas de comportement inadéquat. Un classement en temps réel est accessible aux visiteurs sans authentification.

Les commissaires s'authentifient, sélectionnent leur poste, saisissent un numéro de dossard et enregistrent les résultats. Un mode offline stocke les données localement et les synchronise dès qu'une connexion est disponible.

L'application est développée en HTML, CSS, JavaScript, avec un backend en PHP et MySQL. Son interface épurée privilégie une saisie rapide, avec de grands boutons et champs pour éviter le scrolling inutile. Le design adopte des tons naturels et les mots de passe sont chiffrés pour garantir la sécurité des comptes.

Domaine du TPI : (Cocher la case de couleur correspondante)

<input checked="" type="checkbox"/> Développement	<input type="checkbox"/> Réseau	<input type="checkbox"/> Systèmes
---	---------------------------------	-----------------------------------

Tracé d'activité de l'apprenti/-e durant la dernière année d'apprentissage :

Stage en entreprise et travaux sur mandats (projets) en Phase de professionnalisation

Coordonnées de l'apprenti/-e :

E-Mail : max.langenegger@studentfr.ch
Tél : 078 325 79 50
Prénom : Max
Nom : Langenegger
Adresse : EMF – Ecole des Métiers
Chemin du Musée 2
1700 Fribourg

Le supérieur professionnel

Philippe Galley

Lieu et Date

Fribourg, le 12 février 2025

Table des matières

1	DONNÉE DU PROBLÈME.....	3
2	FORMULATION DU MANDAT – ÉTAT DÉSIRÉ	3
2.1	DESCRIPTION ET OBJECTIFS DU PROJET	3
2.2	EXIGENCES FONCTIONNELLES	10
2.3	ANALYSE.....	11
2.4	CONCEPTION.....	11
2.5	RÉALISATION.....	11
3	INFRASTRUCTURE NÉCESSAIRE.....	11
4	CONNAISSANCES PRÉALABLES	11
5	TRAVAUX PRÉPARATOIRES.....	11
5.1	TRAVAUX À RÉALISER :.....	12
6	STANDARDS D'ENTREPRISE.....	12
7	DOCUMENTATION OBLIGATOIRE	13
8	CRITÈRES D'ÉVALUATION.....	14
9	DÉLAIS	14

1 Donnée du problème

Une course de marche et vol est une discipline combinant randonnée et parapente. Les participants doivent gravir des sommets à pied avant de redescendre en volant, réalisant parfois des épreuves techniques comme des slaloms, des touches sur des zones précises, ou des atterrissages de précision. Cette compétition exige non seulement une excellente condition physique, mais aussi des compétences avancées en pilotage de parapente et une bonne stratégie de gestion de course.

Dans de nombreuses courses comme la Millets Cup, qui se déroule sur les hauteurs de Lessoc, la gestion des résultats repose encore sur un système manuel. Chaque poste (sommet, slalom, touch, cible) est géré par un commissaire, qui enregistre les performances des concurrents sur papier avant de les transmettre à l'organisation. Ce processus est long et sujet aux erreurs : mauvaise retranscription des résultats, oublis, retards dans la mise à jour du classement, et difficulté à rectifier les erreurs.

L'application *Elevate* vise à moderniser et automatiser cette gestion des résultats. Grâce à une saisie numérique en temps réel, chaque commissaire pourra enregistrer directement les performances des concurrents via une interface simple et intuitive, réduisant ainsi les risques d'erreurs et les délais de traitement. Les résultats permettent d'afficher un classement en temps réel accessible à tous. De plus, un mode offline garantit la continuité de la saisie même en l'absence de connexion, avec une synchronisation une fois le réseau rétabli.

En éliminant la gestion papier, *Elevate* apporte gain de temps, fiabilité et accessibilité aux compétitions de marche et vol. Elle permet aux organisateurs de se concentrer sur l'essentiel : le bon déroulement de la course et l'expérience des participants.



Figure 1 - Poste de slalome durant la Millets Cup

2 Formulation du mandat – état désiré

2.1 Description et objectifs du projet

L'objectif du projet est de concevoir et développer une application mobile permettant de consulter et de saisir facilement les résultats d'une compétition de marche et vol. L'application sera une **Progressive Web App (PWA)** utilisable sur l'ensemble des smartphones du marché et reposera sur une **infrastructure backend** assurant la gestion sécurisée des données et des utilisateurs autorisés.

Ce projet a pour ambition de fournir une solution ergonomique, rapide et fiable pour la gestion des résultats, tout en garantissant une utilisation fluide sur le terrain. L'approche PWA assurera une compatibilité optimale sur tous les smartphones sans nécessiter d'installation spécifique, et l'architecture backend garantira une sécurisation des données et une accessibilité en temps réel, même en cas de connexion intermittente.

2.1.1 Affichage et consultation des résultats

L'application devra permettre, dès son ouverture, d'afficher en temps réel le classement de la course. Cette liste présentera les concurrents avec leur photo, numéro de dossard, prénom, nom, code postal et localité de résidence, ainsi que le nombre total de points obtenus. Le classement devra être mis à jour dynamiquement et affiché dans un ordre décroissant des points. La gestion des participants (inscription, modification ou suppression) ne fait pas partie du périmètre du projet ; les données des concurrents seront directement saisies dans la base de données - les photos également.

Depuis la liste des concurrents, il sera possible de sélectionner un participant pour afficher le détail de ses performances. Cet écran devra afficher les résultats enregistrés, avec la mention du type d'épreuve (sommet, touch, cible, etc.), la date et l'heure d'enregistrement, ainsi que le nombre de points attribués. Un bouton permettra de revenir facilement à l'écran d'accueil.

2.1.2 Authentification des commissaires et gestion des accès

Un lien disponible depuis l'écran d'accueil permettra aux commissaires d'ouvrir une page de connexion et de s'authentifier pour accéder à l'interface de saisie des résultats. L'authentification se fera via adresse e-mail et mot de passe, ou via un code PIN, afin de simplifier l'accès rapide sur le terrain. Pour améliorer l'expérience utilisateur, le nom de l'utilisateur devra être stocké dans un cache et récupéré automatiquement lors de connexions ultérieures. Il sera également possible de revenir à l'écran d'accueil sans se connecter.

La gestion des utilisateurs (création, modification, suppression des comptes commissaires) n'est pas incluse dans ce projet. Les informations des commissaires (nom, prénom, e-mail et mot de passe) seront directement enregistrées dans la base de données. Les mots de passe devront être chiffrés pour assurer la sécurité des données.

Une fois connecté, un commissaire pourra se déconnecter à tout moment et revenir à l'écran d'accueil.

2.1.3 Saisie des résultats par les commissaires

Après l'authentification, chaque commissaire devra pouvoir sélectionner un poste parmi ceux auxquels il a été affecté. L'interface affichera uniquement les postes disponibles pour ce commissaire. La gestion des postes et des affectations ne fait pas partie du périmètre du projet et les informations seront directement stockées en base de données. Un poste est défini par un type (ex. "Sommet", "Slalom", "Touch", "Atterrissage"), un nom (ex. "Les Millets", "La Brâ", "Albeuve", etc.) et un nombre de points attribués pour chaque action réussie. De la même manière, le lien entre les commissaires et les postes auxquels ils sont affectés seront directement enregistrés dans la base de données.

Une fois un poste sélectionné, le commissaire pourra enregistrer un résultat pour un concurrent. Il devra saisir le numéro de dossard du participant, ainsi que la date et l'heure du résultat. Par défaut, la date et l'heure courantes seront préremplies, mais un bouton permettra de mettre à jour ces valeurs automatiquement. Un champ de remarque facultative pourra être rempli par le commissaire. Un bouton "Enregistrer" permettra de sauvegarder les informations dans la base de données.

Une fois un résultat enregistré, la page se réinitialisera pour permettre une nouvelle saisie sans perte de temps. Un bouton permettra également de revenir à la liste des postes.

2.1.4 Gestion des malus

L'application devra aussi permettre l'enregistrement de malus, qui sont attribués aux participants en cas de comportement inapproprié (non-respect du fair-play, non-respect des règles environnementales, dépôt de déchets, etc.). Pour attribuer un malus, le commissaire devra saisir une remarque expliquant la raison de la sanction et préciser l'heure à laquelle elle a été enregistrée. Un bouton "Enregistrer" permettra de valider le malus, qui sera intégré dans les résultats du concurrent concerné. Le fonctionnement est identique à la gestion des résultats, un malus n'a toutefois pas de nom. L'accès à la saisie des malus devrait être accessible depuis l'écran contenant la liste des postes affectés aux commissaires.

2.1.5 Gestion des données hors connexion

Une fonctionnalité essentielle de l'application sera la capacité à fonctionner hors connexion, car certains commissaires pourront se trouver dans des zones sans réseau. Il devra être possible de continuer d'enregistrer des résultats même en mode hors ligne et de les synchroniser une fois la connexion rétablie.

Le stockage des résultats hors ligne pourra se faire via une base de données locale ou une autre technologie adaptée aux PWA. Lorsqu'un commissaire saisit un résultat sans connexion, celui-ci devra être stocké en local. Un élément graphique doit indiquer à l'utilisateur que des résultats sont en attente.

Une fois la connexion retrouvée, l'application devra permettre une synchronisation automatique des résultats ou une synchronisation manuelle (à définir lors de la conception) - dans ce cas la le commissaire pourra déclencher l'envoi des résultats restants à partir d'un bouton dédié.

L'interface devra indiquer clairement si des résultats sont en attente de synchronisation et afficher une confirmation lorsque les données seront bien enregistrées sur le serveur.

2.1.6 Infrastructure backend et Web API

L'application reposera sur un backend sécurisé qui fournira une Web API exposant toutes les fonctionnalités nécessaires à la gestion des données de l'application. Ce backend devra gérer :

- L'enregistrement et la récupération des résultats de la course.
- L'authentification des commissaires. Cependant, la gestion complète des utilisateurs (création, suppression, gestion des rôles) n'est pas incluse dans le projet.
- L'accès aux données de classement en temps réel pour l'application mobile.
- L'accès aux données des postes de contrôle et des affectations des commissaires. Toutefois, la gestion de ces informations ne sera pas développée ; les données seront saisies directement en base de données avant l'événement.
- La sécurisation des échanges entre l'application mobile et la base de données, notamment via le chiffrement des mots de passe.
- La contrôle des données pour empêcher l'injection SQL.

L'accès aux méthodes qui gèrent les données propres aux commissaires devra être sécurisé et ne pas permettre d'appels frauduleux et non-authentifiés.

2.1.7 Périmètre et exclusions du projet

Certains éléments sortent du cadre du développement de cette application et ne seront pas pris en charge dans ce projet :

- Gestion des participants (ajout, modification, suppression) : les concurrents seront saisis manuellement en base de données avant l'événement.
- Gestion des comptes utilisateurs pour les commissaires : aucune interface d'administration des utilisateurs ne sera développée.
- Gestion des postes et des affectations des commissaires : ces informations seront stockées directement en base de données avant la course.

2.2 Technologies et schéma de principe

L'application mobile sera basée et développée sur la technologie PWA (Progressive Web Applications) avec notamment l'utilisation des langages HTML, CSS et Javascript. L'utilisation d'un Framework Web tels que React JS, Angular, Vue JS ou un autre est envisageable mais pas obligatoire. La Web API qui est la partie backend sera développée en PHP. Toutes les données utiles à l'application seront enregistrées dans une base de données MySQL.

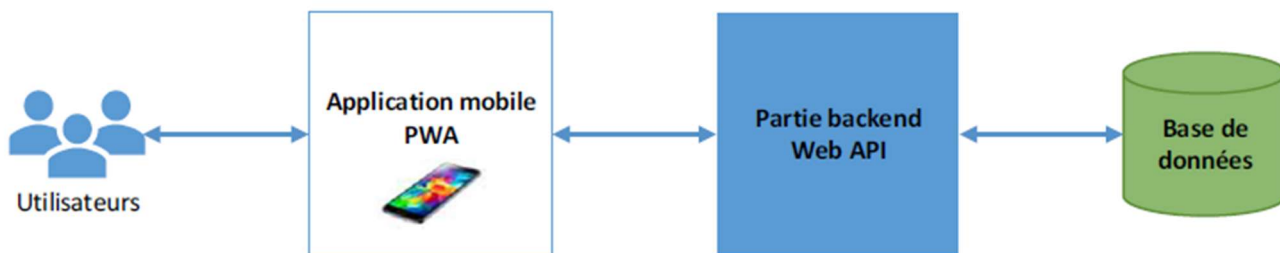


Figure 2 - Schéma de principe

2.3 Application mobile

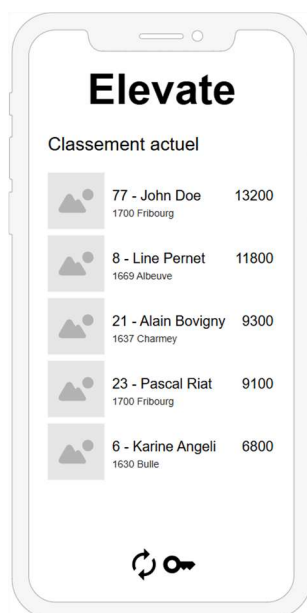
L'application mobile permettra aux utilisateurs d'effectuer les opérations suivantes :

- Visualiser le classement d'une course.
- Visualiser les résultats détaillés d'un concurrent.
- S'authentifier dans l'application.
- Sélectionner un poste afin de saisir des résultats.
- Saisir des résultats.

2.3.1 Maquettes de l'application

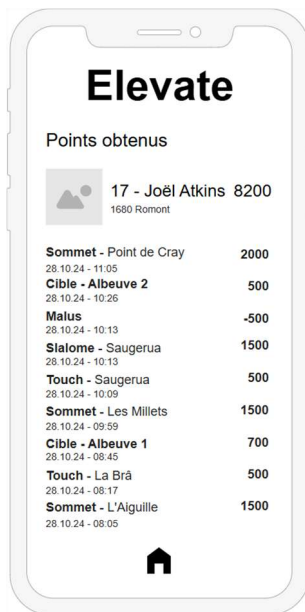
Les maquettes suivantes ont pour but d'éclaircir le fonctionnement attendu de l'application. Les éléments qui y sont représentés sont uniquement indicatifs et leur liste n'est pas exhaustive.

2.3.1.1 Visualisation du classement



- Au démarrage de l'application, l'utilisateur peut directement consulter le classement de la course.
- La liste doit être ordonnée en fonction du total décroissant des points des concurrents.
- La liste doit contenir la photo du concurrent, son numéro de dossard, son prénom, son nom, le code postal et son lieu de résidence ainsi que le nombre total de points obtenus.
- La liste doit être automatiquement mise à jour de manière périodique.
- Lorsque l'utilisateur sélectionne un concurrent, l'écran avec le détail des résultats doit s'ouvrir (cf. 2.3.1.2 Visualiser les résultats détaillés).
- Un premier bouton doit permettre de mettre à jour le classement.
- Un second bouton doit permettre d'ouvrir l'écran d'authentification pour les commissaires (cf. 2.3.1.3 Authentification d'un commissaire).

2.3.1.2 Visualiser les résultats détaillés



- L'écran doit contenir, dans la partie sommitale, la photo du concurrent, son numéro de dossard, son prénom, son nom, le code postal et son lieu de résidence ainsi que le nombre total de points obtenus.
- La liste détaillée de tous les résultats doit être affichée. Cette liste contient le type de poste, le nom du poste, la date et l'heure de saisie des résultats ainsi que le nombre de points attribués.
- Un bouton doit permettre de retourner à l'écran principal, contenant le classement général (cf. 2.3.1.1 Visualisation du classement).

2.3.1.3 Authentification d'un commissaire



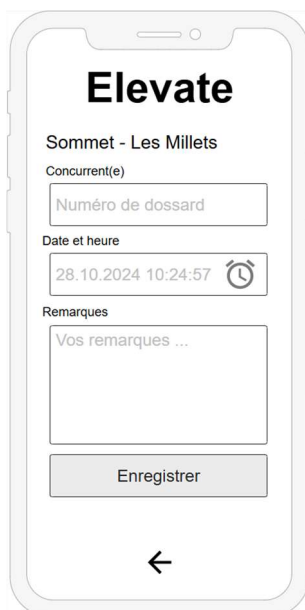
- L'écran de login doit permettre aux commissaires de s'authentifier.
- La page doit contenir un champ pour le login et un autre pour le mot de passe.
- Un bouton doit permettre de s'authentifier.
- La page peut également contenir un logo. Le logo peut être réutilisé pour l'icône de l'application PWA.
- D'une connexion à l'autre, le login du précédent utilisateur doit être gardé en cache et inséré automatiquement dans le champ login afin d'accélérer les connexions ultérieures.
- Un bouton doit permettre de retourner à l'écran principal, contenant le classement général (cf. 2.3.1.1 Visualisation du classement).

2.3.1.4 Sélection d'un poste



- La page doit indiquer le nombre résultats qui n'ont pas pu être transmis et enregistrer dans la base de données à cause d'une connexion réseau interrompue – il s'agit des résultats en attente.
- En fonction des moyens techniques choisis, la transmission des résultats en attente peut être faite automatiquement ou manuellement selon une action exécutée par le commissaire.
- Des boutons doivent permettre de sélectionner un poste sur lequel un commissaire travaille. Les boutons ouvriront l'écran de saisie des résultats (cf. 2.3.1.5 Saisie des résultats). Pour rappel, seuls les postes affectés au commissaire connecté sont visibles.
- Un bouton malus doit permettre d'enregistrer un malus.
- Les boutons sont regroupés en fonction du type de poste.
- Un bouton doit permettre de fermer la session et de retourner à l'écran d'authentification (cf. 2.3.1.3 Authentification d'un commissaire).

2.3.1.5 Saisie des résultats



- Au sommet de l'écran doit être affiché le type de poste et son nom.
- Un champ doit permettre de saisir le numéro du dossard. Seul une valeur numérique doit pouvoir être saisie.
- Un champ doit permettre de saisir la date et l'heure du résultat. Ce champ sera initialisé avec la date et l'heure actuelles.
- Un bouton doit permettre d'ajuster rapidement la date et surtout l'heure actuelle.
- Un champ textuel, sur plusieurs lignes, doit permettre aux commissaires d'insérer une éventuelle remarque liée au résultat.
- Un bouton doit permettre d'enregistrer le résultat sur le serveur. L'application doit demander une confirmation à l'utilisateur avant d'envoyer le résultat.
- Un message d'erreur doit être affiché si le dossard n'existe pas.
- Lorsque le résultat est enregistré, le formulaire de saisie doit être réinitialisé et le curseur doit se retrouver dans le champ pour le dossard.
- Les résultats qui ne peuvent pas être envoyés doivent être mis en attente. L'utilisateur doit être averti.
- Un bouton doit permettre de retourner à l'écran de sélection des postes (cf. 8 Sélection d'un poste).

2.4 Application backend

La tâche principale de cette partie du système est de gérer la sélection, l'enregistrement, la mise à jour et la suppression de toutes les données liées à l'application mobile.

Les données seront sauvegardées dans une base de données MySQL, y compris les images liées aux concurrents.

La sélection et l'enregistrement de toutes les données devront être accessibles à travers un ensemble de méthodes dédiées pour chacune des opérations. Ces méthodes devront être sécurisées et accessibles uniquement pour un utilisateur authentifié.

Les méthodes devront être protégées contre les injections usuelles et les données sensibles (mot de passe) ne devront pas être lisibles.

La partie backend ne comprendra pas d'interface graphique. Développée en PHP, elle pourra être hébergée localement sur un server Wamp (PHP/MySQL) puis déployée sur un serveur AlpNet dédié au travail de TPI. Cet hébergement offrira également la base de données MySQL, pour le stockage des données.

2.5 Exigences fonctionnelles

Exigence fonctionnelle	Description	Priorité Haute Moyenne Basse	Temps estimé (jours)
Création de la base de données	Analyse, conception et réalisation de la base de données	Haute	0.5 jour
Tests	Tests fonctionnels de l'application	Haute	1 jour
Documentation	Rédaction de la documentations - Analyse - Conception - Réalisation	Haute	3 jours
Backend – Mise en place de la base de l'application	Mise en place de la structure de base de l'application qui permettra d'y implémenter les différentes fonctionnalités liées au backend. Documentation.	Haute	0.5 jour
Backend – Gestion du classement	Implémentations des méthodes liées à l'affichage du classement et des résultats détaillés	Haute	0.5 jour
Backend – Gestion des résultats	Implémentation des méthodes liées à la saisie des résultats par les commissaires.	Haute	0.5 jour
Mobile – Mise en place de la base de l'application	Mise en place de la structure de base de l'application mobile qui permettra d'y implémenter les différentes fonctionnalités. Documentation	Haute	0.5 jour
Mobile – Consultation du classement	Implémentation de l'interface qui permettra de consulter le classement de la course. Documentation.	Moyenne	0.5 jour
Mobile – Consultation des résultats détaillés	Implémentation de l'interface qui permettra de consulter les résultats détaillés d'un concurrent. Documentation.	Basse	0.5 jour
Mobile – Login et sélection des postes	Implémentation de la page de login et de la page contenant les postes affectés aux commissaires authentifiés. Documentation.	Haute	0.5 jour
Mobile – Saisie des résultats	Implémentation de la page permettant la saisie des résultats. Documentation.	Haute	1 jour
Mobile – Gestion des résultats en attente	Implémentation du mécanisme de gestion des résultats en attente et l'interface qui permet l'envoi des données. Documentation.	Moyenne	1 jour

2.6 Analyse

Cette phase de travail permet d'analyser les besoins tels qu'ils ont été décrits dans la donnée du problème, en les représentant par des diagrammes ou schémas adéquats.

Durant la phase d'analyse, l'apprenti/-e accomplit les tâches suivantes :

- Analyser les tâches à réaliser.
- Créer le diagramme des cas d'utilisation global.
- Créer les différents diagrammes d'activités pour les cas d'utilisations majeurs de l'application (à définir avec le supérieur professionnel).
- Documenter l'analyse.

2.7 Conception

Durant la phase de conception, l'apprenti/-e accomplit les tâches suivantes :

- Réaliser le schéma relationnel de la base de données.
- Créer le diagramme des classes de l'application.
- Créer les différents diagrammes d'interactions des méthodes de l'application (à définir avec le supérieur professionnel).
- Planifier les tests fonctionnels de l'application.
- Documenter la conception.

2.8 Réalisation

Durant la phase de réalisation, l'apprenti/-e accomplit les tâches suivantes :

- Implémenter une base de données.
- Implémenter un service Web sécurisé en PHP avec l'accès aux données d'une base de données.
- Implémenter une application mobile PWA utilisant une authentification.
- Tester le fonctionnement complet de l'application.
- Documenter la réalisation.

3 Infrastructure nécessaire

Pour la réalisation du projet, l'apprenti/-e disposera de l'infrastructure suivante :

- Un PC Windows 11 connecté à Internet.

4 Connaissances préalables

Toutes les connaissances nécessaires en programmation sont vues durant les modules de formation.

5 Travaux préparatoires

- Installation de WAMP (PHP/MySQL).
- Installation d'un IDE pour le développement du projet.
- Installation d'un logiciel de design UML (Enterprise Architect).
- Lecture de la documentation relative au TPI.

5.1 Travaux à réaliser :

- Tests technologiques pour la mise en place d'une application PWA de base.
- Tests technologiques pour faire fonctionner une application PWA en mode hors ligne (cache des fichiers)
- Tests technologiques pour l'enregistrement d'une image dans une base de données MySQL et la récupération d'une image depuis une base de données afin de l'afficher dans une page HTML.
- Tests technologiques pour établir si une connexion réseau est active ou non dans une application PWA. Analyser les moyens possibles pour savoir si une connexion est perdue ou rétablie.
- Tests technologiques sur les moyens d'enregistrer des données (les résultats en attente) en local et de gérer ces données: cache, base de données locale, fichier, autre.
- Rédaction du document d'analyse pour les points ci-dessus.

6 Standards d'entreprise

Les standards sont les mêmes que les standards de l'entreprise appliqués durant tous les projets de développement :

- Les diagrammes UML permettront de comprendre la structure orienté objet du programme.
- Toutes les méthodes métiers doivent être documentées conformément aux standards.
- Architecture MVC ou parente.
- Les standards de programmation doivent être respectés.
 - Tous les paramètres configurables doivent se trouver dans un fichier de configuration externe.
 - Une logique unique doit être respectée quant au nommage (pertinence des noms et logiques des noms) des variables, classes, méthodes, etc.
 - Les vues et les scripts doivent être séparés.

7 Documentation obligatoire

A la fin du projet, l'apprenti/-e doit fournir les documents suivants :

Une planification

Cette planification doit être réalisée au début du projet avant toute autre action (selon modèle fourni). Elle décrit les étapes importantes du projet ainsi que la durée estimée correspondante. Elle doit être validée par le supérieur professionnel.

Un journal de travail

Ce document décrit les diverses étapes et activités liées au projet (selon modèle fourni).

Une documentation d'analyse

Ce document détermine les exigences et contraintes du projet et permet la justification des choix pour la réalisation du travail demandé. Ce document est composé de :

1. Résumé
 - a. Synthèse de la documentation du mandant.
 - b. Choix définitifs effectués.
2. Tests technologiques (cf. 5 Travaux préparatoires)
3. Analyse UML avec les diagrammes UML qui seront nécessaires :
 - a. Cas d'utilisation (uses case).
 - b. Diagramme d'activité pour les cas d'utilisation majeurs de l'application.

Une documentation de réalisation

La documentation de réalisation a pour objectif de faciliter la maintenance et doit contenir les informations suivantes :

1. Conception
 - a. Architecture proposée.
 - b. Diagrammes de classes de l'application.
 - c. Diagramme d'interactions pour les cas d'utilisation majeurs de l'application.
 - d. Planification des tests fonctionnels de l'application.
2. Implémentation / Réalisation
 - a. Schéma relationnel de la base de données.
 - b. Mise en évidence des éléments pertinents implémentés avec des extraits de code commentés.
 - c. Réalisation des tests fonctionnels de l'application.
3. Remarques
 - a. Problèmes rencontrés
 - b. Limites
 - c. Améliorations possibles
 - d. Extensions possibles et leurs mises en fonction.

Une documentation d'utilisation

Aucune documentation d'utilisation n'est demandée dans le cadre de ce TPI.

Un Web Summary

Ce document a pour objectif de présenter le projet de manière succincte.

La remise de la documentation

La documentation est remise selon les instructions du manuel ICT-FR, partie C, point 1.9.1 « Périmètre du rapport, Tips ».

Le manuel est disponible dans PkOrg sous : Documents ➔ Documents pour tous :

<http://manuel.ict-fr.ch>

8 Critères d'évaluation

<http://criteres.ict-fr.ch>

Compétences professionnelles globales (selon la grille d'appréciation)

Résultat et efficience

1. **Solution / Produit** (selon la grille d'appréciation)
2. **Critères spécifiques au projet**
 - 235 – Projet avec UML
 - 167 – Style de codage - Documentation
 - 166 – Style de codage – Lisibilité du code
 - 132 – Mise en oeuvre, implémentation correcte (BDD)
 - 146 – Satisfaction utilisateur: GUI, utilisation
 - 164 – Codage: Traitement des erreurs
 - 125 – Le programme a-t-il été structuré intelligemment sans procédures et fonctions et sous-procédures/fonctions redondantes
3. **Documentation** (selon la grille d'appréciation)
4. **Journal de travail** (selon la grille d'appréciation)

Présentation et entretien professionnel (selon la grille d'appréciation)

9 Délais

Le projet y compris la remise du projet aux experts se terminera le 2 juin 2025 à 17 heures. Le dernier délai pour la présentation du projet est fixé au 11 juin 2025 à 12:00.

La documentation est déposée sur PkOrg dans les temps.