

Tâche

Le but du Travail Pratique 1 est de créer les documents et artefacts de votre projet. Ceux-ci pourront être utilisés directement pour le développement logiciel du projet par la suite.

Artefacts à aborder dans le projet :

- Description de l'idée (Faisabilité/commercialité/originalité)
- Le cahier des charges fonctionnel
- Les documents de conception (modèles généraux - haut niveau) :
 - Architecture logicielle
 - Conception
 - Tests
- Les diagrammes UML (modèles détaillés - bas niveau) :
 - Cas d'utilisation (use case)
 - Classe (class)
 - Séquence (sequence)
 - Activité (activity)
 - Maquette (wireframe)

Chaque manque aux consignes réduit la note finale du travail de manière proportionnelle à l'ampleur de l'erreur et du total alloué par la section.

Le total est sur 100 points, et converti au pourcentage respectif par la suite.

Remise du projet (Obligatoire)

- Un (1) fichier de rapport en format PDF (.pdf). Vous pouvez utiliser votre logiciel favori, tant que la conversion finale en PDF est possible.
 - Pas besoin de fournir la source des diagrammes ou autres fichiers sources, seulement tout inclure les images dans le rapport.
 - Si vous avez plusieurs PDFs, **il faut les fusionner en un seul PDF**.
- **Seulement une remise par équipe. Les personnes coéquipières n'auront qu'à mettre leur prénom et nom sur le rapport.**
- Le non-respect des consignes de remise octroie la note zéro (0%).

1 Faisabilité/commercialité/originalité de l'idée (/25)

- (25 pts) Répondre aux critères tels que vu dans le cours (les voici par rappel, voir les notes de cours pour plus de détails) :
 - **Création de l'équipe** : Assigner les membres de l'équipes aux rôles techniques dédiés du cours.
 - **Recherche et analyse du marché** : Trouver des compétiteurs de votre idée préliminaire.
 - **Trouver l'idée innovatrice** : Trouver le sujet de votre projet.
 - * L'analyse du marché aide beaucoup pour cette étape.
 - **Collecte des commentaires à la source** : Consulter 2 personnes de votre entourage par membre de l'équipe.
 - * Vous pouvez rendre les résultats anonymes et il est conseillé de ne pas "forger" les résultats soi-même.
 - **Créer un prototype** : Faire le lien avec la stratégie utilisée et les modèles généraux/détaillés produits dans les sections suivantes.
 - **Penser à la commercialisation** : Avoir une idée du marketing et de l'image de marque comme les logos et les vitres/dispositifs (i.e. site Web) grand publique.
 - * Si vous êtes capables de faire le logo dans cette étape (ou les brouillons) c'est une bonne étape pour votre portfolio.

2 Le cahier des charges fonctionnel (/25)

- (15 pts) Rédiger le cahier des charges fonctionnel avec toutes l'information nécessaire **et précis** comme vu en cours (les voici par rappel, voir les notes de cours pour plus de détails) :
 - Introduction
 - Description générale
 - Spécifications détaillées
 - Planification et organisation du projet
 - Critères d'acceptation et de validation
 - Maintenance et évolution
- (10 pts) Respecter les points suivants :
 - Bien structuré et esthétiquement professionnel.
 - Contient une section avec une gestion des risques.
 - Inclure un diagramme de Gantt précis, structuré, et esthétiquement professionnel

3 Les documents de conception (modèles généraux) (/25)

- (15 pts) Choisir et créer les modèles généraux de votre projet (les voici par rappel, voir les notes de cours pour plus de détails) :
 - **Architecture logicielle** : La globalité d'un système (une application) spécifique
 - **Conception** : Méthodologie d'une ou plusieurs applications
 - **Paradigme de tests** : La technique de tests permettant de valider la conception (le code des applications du système)
- (10 pts) Respecter les points suivants :
 - Bien structuré et esthétiquement professionnel.
 - Utiliser des illustrations pour décrire la conception.

4 Les diagrammes UML (modèles détaillés) (/25)

- (15 pts) Créer les diagrammes suivants pour modéliser votre projet (les voici par rappel, voir les notes de cours pour plus de détails) :
 - **Cas d'utilisation (Use case)** : Démontre de manière très générale et haut niveau l'interaction dans une phase de validation d'exigences.
 - * Niveau : Système entier (Fullstack)
 - * Un (1) diagramme pour le système complet.
 - **Classe (Class)** : Démontre les tables de la base de données (i.e. les classes de votre API avec l'utilisation de votre ORM pour générer la DB).
 - * Niveau : Backend (Les classes de la base de données (DB)).
 - * Un (1) diagramme pour représenter toutes les classes ORM (tables) de la DB.
 - **Séquence (Sequence)** : Démontre des cas d'utilisations (interactions) d'un système haut niveau avec des acteurs bien définis et des notions plus techniques.
 - * Niveau : Système entier (Fullstack)
 - * Un (1) diagramme **par membre de l'équipe** pour les différents composants du système.
 - **Activité (Activity)** : Démontre les flux dynamique d'un système de plus bas niveau et technique.
 - * Niveau : Système entier (Fullstack)
 - * Un (1) diagramme **par membre de l'équipe** pour les différents composants du système.
 - **Maquette (Wireframe)** : Démontre le design bref (pas de graphisme avancé) et principalement les fenêtres et liaisons les liaisons de la topologie des interfaces du projet.
 - * Niveau : Frontend
 - * Un (1) diagramme pour la partie frontend.
- (10 pts) Respecter les points suivants :
 - Bien structuré et esthétiquement professionnel.
 - Les diagrammes trop simples seront pénalisées.
 - Les images avec une définition faible seront pénalisées.

Bibliographie (Obligatoire)

- (Obligatoire) Mettre toutes les ressources (liens) utilisées d'Internet/ChatGPT