

# UNIVERSITÉ DU QUÉBEC À MONTRÉAL

## TP3

### Réalisation et contrôle

## PRÉSENTÉ À L'UNIVERSITÉ DU QUÉBEC À MONTRÉAL COMME EXIGENCE DU COURS

INF6150.030 – Génie logiciel III : Conduite de projets  
informatiques (Hiver 2024)

Chargé de cours : Ammar Hamad

### PAR L'ÉQUIPE #06: (LES TOURISTES)

Front-End - KEVIN DA SILVA - ( DASK30049905 ),

Back-End - THAI MINH DANNY NGUYEN - ( NGUT71060209 ),

Back -End - SAAFI ACHREF - (SAAA87070201),

Back -End - BENHAMAD ALI - (BENA21119407),

Back -End - JEAN-CHRISTOPHE CLOUÂTRE - (CLOJ73020201)

Front-End - EMMANUEL EXCELLENT - (EXCE23099402)

Département d'informatique

18 Mars 2024

# Table des matières

A- Mise à jour (MAJ) de la documentation du PRODUIT.....	p.3
B- MAJ des documents de gestion de PROJET.....	p.10
C- Revues d'itération (démonstrations).....	p.15
D- Rapports de suivi.....	p.15

## TP3 – Réalisation et contrôle

### Livrables et leurs éléments demandés

#### A- Mise à jour (MAJ) de la documentation du PRODUIT

##### Documentation du Projet

##### 1) Exigences

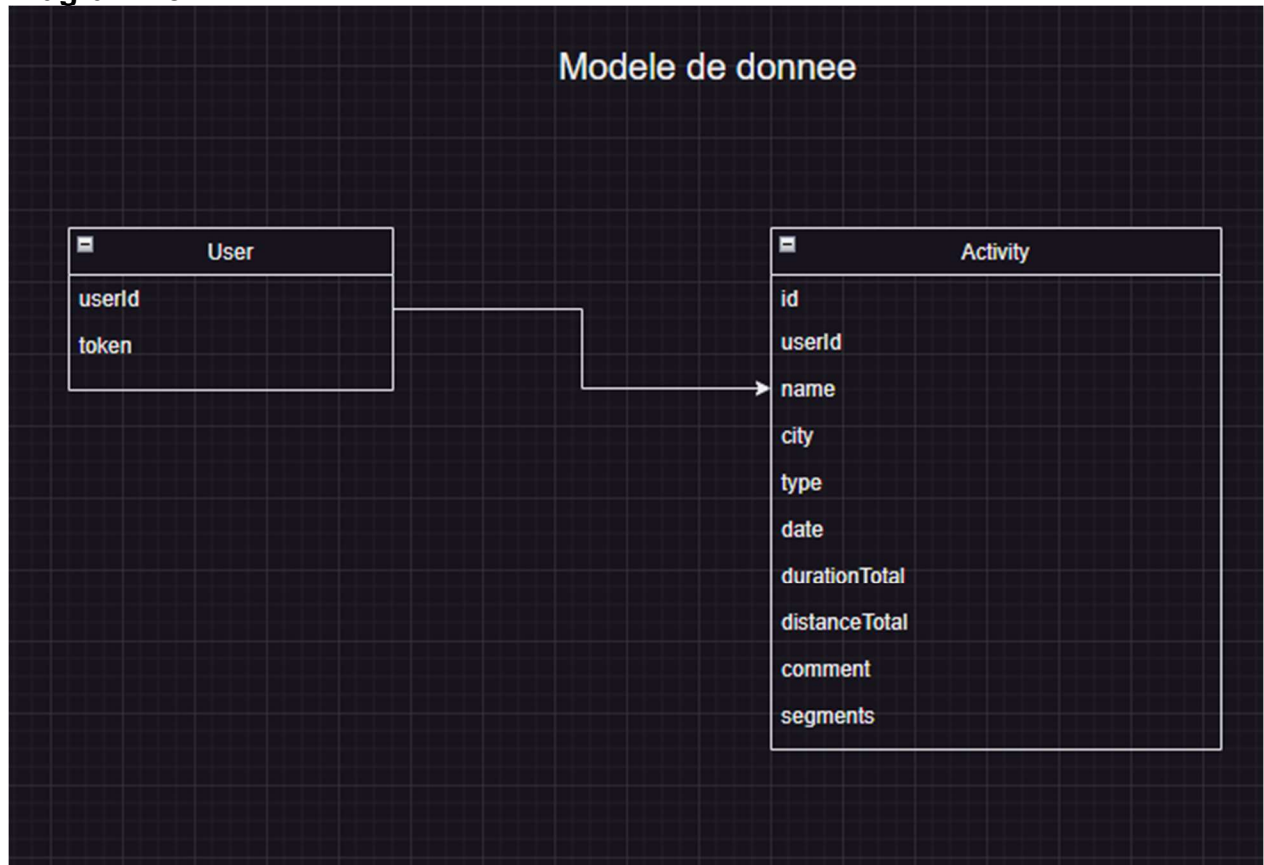
Tout d'abord, le projet vise à développer une application de suivi d'activités physiques permettant aux utilisateurs d'enregistrer leurs activités telles que la course à pied, le vélo et la marche. L'application doit permettre aux utilisateurs de saisir manuellement les détails de leurs activités, tels que le nom, le type, la durée, la distance, la ville et les commentaires. De plus, elle doit offrir la possibilité d'importer des fichiers GPX pour extraire automatiquement les données d'activité à partir de ceux-ci. L'application doit être sécurisée, avec des fonctionnalités d'authentification des utilisateurs via des jetons. La planification et la gestion du projet doivent être bien organisées, avec une attention particulière portée à la communication et à la gestion des changements. La conception de l'application doit prendre en compte une architecture logicielle bien structurée, favorisant l'évolutivité et la maintenabilité. L'implémentation doit suivre des normes de codage cohérentes, avec des tests complets couvrant les différentes fonctionnalités de l'application. Le déploiement doit être documenté et automatisé autant que possible pour assurer des déploiements rapides et fiables. Enfin, la maintenance doit être efficace, avec une identification rapide des problèmes et des correctifs apportés en conséquence, tout en assurant une traçabilité claire des modifications et des correctifs.

**Emplacement dans GitLab :** [https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main?ref\\_type=heads](https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main?ref_type=heads)

## 2) Modèle de Données

**Description textuelle :** Ce modèle de données représente la structure des données clés de notre application. Il comprend les entités principales ainsi que les relations entre elles.

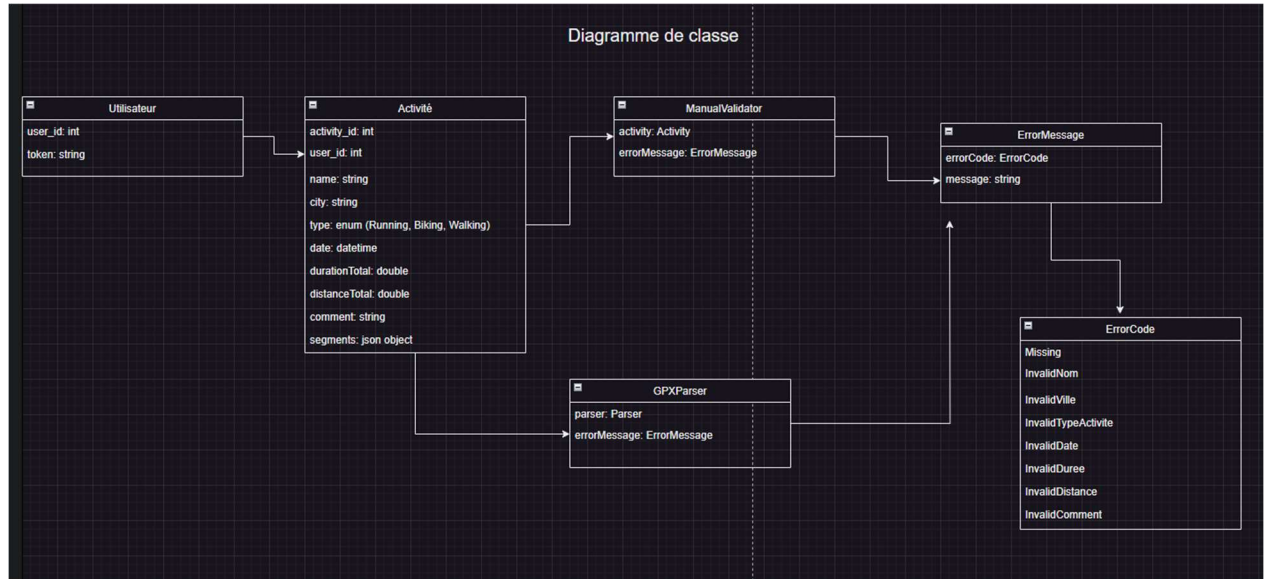
**Diagramme :**



### 3) Classes ou Composants dans le Langage/Framework Utilisé

**Description textuelle :** Ce diagramme de classes illustre les différentes classes ou composants de notre application développée pendant le sprint. Il montre les relations et les interactions entre ces classes dans le système de gestion des activités.

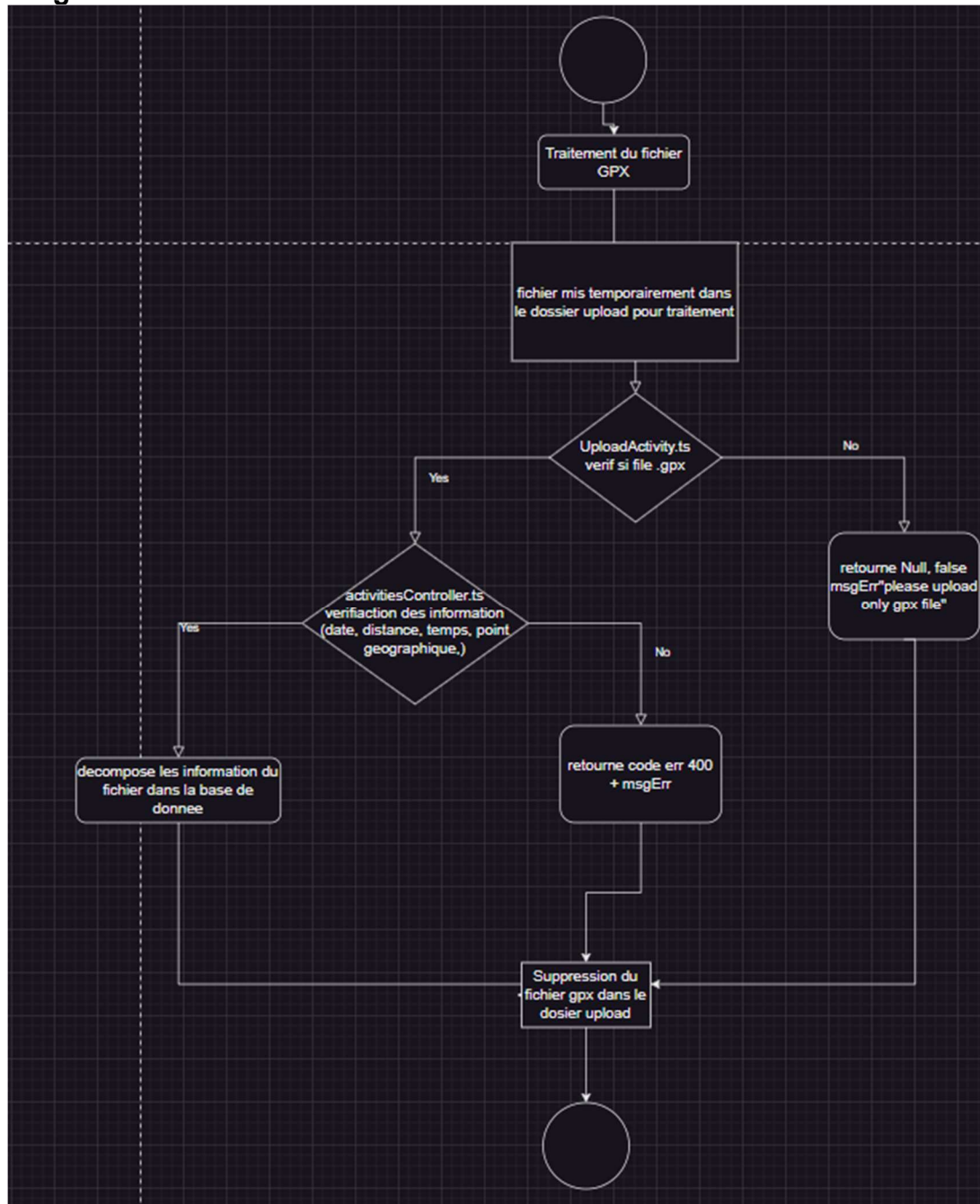
**Diagramme :**



#### 4) Flux Logiques ou Algorithmes

**Description textuelle :** L'organigramme suivant détaille le flux logique impliqué dans notre application. Il décrit comment les différentes actions ou processus de traitement du fichier GPX sont enchaînés pour atteindre un objectif spécifique.

**Diagramme :**



### 3) Documentation du Code

La documentation du code source est essentielle pour assurer sa compréhension, sa maintenance et sa contribution efficaces. Elle comprend des commentaires clairs et concis expliquant la logique, le fonctionnement et l'utilisation du code. Les guides de style de codage, tels que javadoc, svelte kit et eslint, sont également inclus pour garantir la cohérence du code et faciliter la collaboration entre les contributeurs. Cette documentation joue un rôle crucial dans la gestion du projet, en permettant aux développeurs de travailler de manière collaborative et efficace.

#### Emplacement dans GitLab :

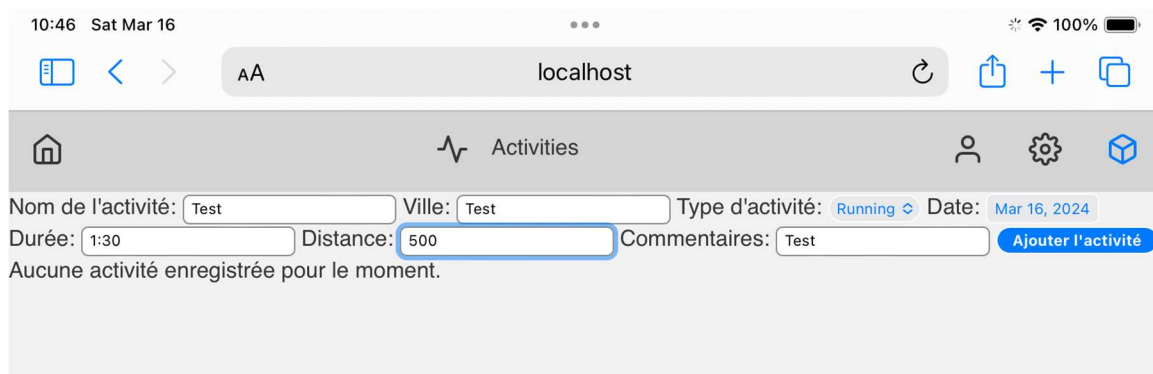
**Front-end Documentation** : [https://gitlab.info.ugam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main/frontend?ref\\_type=heads](https://gitlab.info.ugam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main/frontend?ref_type=heads)

**Back-end Documentation** : [https://gitlab.info.ugam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main/backend?ref\\_type=heads](https://gitlab.info.ugam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main/backend?ref_type=heads)

#### 4) Autres Informations Utiles

Dans cette section, nous présentons des informations complémentaires importantes pour le projet. Nous commençons par un exemple illustratif d'un formulaire rempli avec des données valides, suivi du résultat correspondant dans la base de données MySQL. Ensuite, nous fournissons un exemple de message d'erreur généré lorsque le formulaire est mal rempli, mettant en évidence l'importance de la validation des données pour assurer l'intégrité et la cohérence des données dans le système. Les images ci-dessous illustrent visuellement ces exemples, offrant ainsi une meilleure compréhension des processus impliqués dans le projet.

Page web Activiter (Formulaire exemple) :



10:46 Sat Mar 16

AA localhost

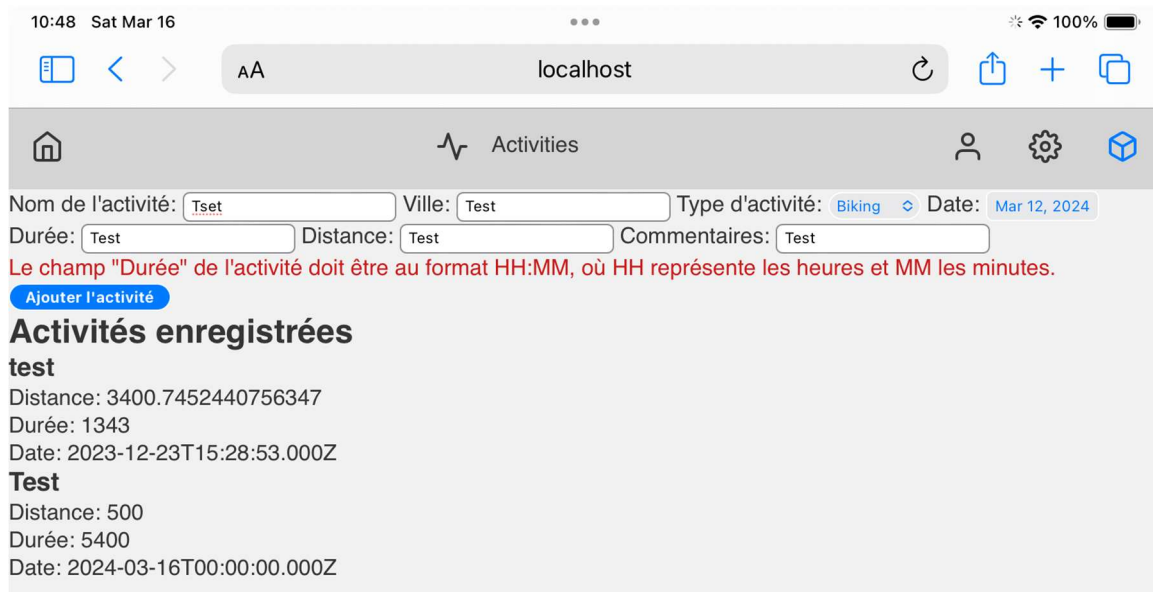
Activities

Nom de l'activité: Test Ville: Test Type d'activité: Running Date: Mar 16, 2024

Durée: 1:30 Distance: 500 Commentaires: Test [Ajouter l'activité](#)

Aucune activité enregistrée pour le moment.

#### Base de Donnee – Back-end



10:48 Sat Mar 16

AA localhost

Activities

Nom de l'activité: Tset Ville: Test Type d'activité: Biking Date: Mar 12, 2024

Durée: Test Distance: Test Commentaires: Test [Ajouter l'activité](#)

**Activités enregistrées**

**test**

Distance: 3400.7452440756347

Durée: 1343

Date: 2023-12-23T15:28:53.000Z

**Test**

Distance: 500

Durée: 5400

Date: 2024-03-16T00:00:00.000Z

Le champ "Durée" de l'activité doit être au format HH:MM, où HH représente les heures et MM les minutes.



Case d'entrer d'erreur :

10:47 Sat Mar 16

localhost

Activities

Nom de l'activité:  Ville:  Type d'activité:  Date:

Durée:  Distance:  Commentaires:  Ajouter l'activité

**Activités enregistrées**

**test**  
Distance: 3400.7452440756347  
Durée: 1343  
Date: 2023-12-23T15:28:53.000Z

**Test**  
Distance: 500  
Durée: 5400  
Date: 2024-03-16T00:00:00.000Z

Si vous souhaitez tester le projet par vous-même, voici les étapes à suivre :

- 1- Cloner le projet depuis GitLab..
- 2- Ouvrir votre IDE (de préférence Visual Studio Code) et charger le projet.
- 3- Lire et suivre les instructions du fichier README.md situé dans le dossier du back-end. Lien : [https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main/backend?ref\\_type=heads](https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main/backend?ref_type=heads)
- 4- Lire et suivre les instructions du fichier README.md situé dans le dossier du front-end. Lien : [https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main/frontend?ref\\_type=heads](https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/tree/main/frontend?ref_type=heads)
- 5- Une fois ces étapes complétées, le projet devrait s'afficher dans votre navigateur, vous permettant ainsi d'interagir avec.

## B- MAJ des documents de gestion de PROJET

### Liste d'activités (Sprint#0)

La liste détaillée des activités du sprint #0 est disponible via ce lien :

<https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/milestones/1#tab-issues>

### Liste d'activités (Sprint#1)

La liste détaillée des activités du sprint #1 est accessible via ce lien. Certaines tâches, plus précisément les suivantes :

- [Implémentation de l'Action de Modification dans la Base de Données - \(5 pts\)](#)
- [Implémentation de l'Action de Suppression dans la Base de Données - \(5 pts\)](#)
- [Affichage des Messages d'Avertissement - \(2 pts\)](#)
- [Affichage des Boutons d'Action - \(2 pts\)](#)
- [Modification d'Activités - \(2 pts\)](#)
- [Suppression d'Activités - \(1 pts\)](#)

Ont été reportées au sprint #2, faute de temps. <https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/milestones/6#tab-issues>

### Liste d'activités future (Sprint#2)

La liste détaillée des activités du sprint #2 sont accessible via ce lien :

<https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/milestones/7#tab-issues>

## **Carnet des Obstacles**

### **1) Identification des Obstacles**

Au cours du sprint, plusieurs obstacles ont été rencontrés, notamment :

- Problème de sécurité lié à la clé de l'API Google Maps.
- Difficulté de compréhension de la structuration des fichiers du serveur.
- Obstacle rencontré lors de la récupération du token du compte utilisateur.
- Obstacle d'adaptation au code TypeScript et à la syntaxe ESLint.

### **2) Description des Obstacles**

- Problème de sécurité lié à la clé de l'API Google Maps : Nous avons dû résoudre ce problème en stockant la clé de l'API dans un environnement sécurisé (env) accessible uniquement aux propriétaires du serveur.
- Difficulté de compréhension de la structuration des fichiers du serveur : Nous avons investi du temps dans l'étude de la documentation de Svelte Kit pour mieux comprendre la structure des fichiers et leur organisation dans le serveur.
- Obstacle rencontré lors de la récupération du token du compte utilisateur : Une analyse approfondie du code de l'autre équipe nous a permis de clarifier les étapes nécessaires et d'implémenter une solution efficace.
- Obstacle d'adaptation au code TypeScript et à la syntaxe ESLint : Nous avons surmonté ce défi en investissant du temps dans l'adaptation au langage TypeScript et à la syntaxe ESLint, notamment en lisant attentivement la documentation associée et en pratiquant régulièrement.

### **3) Actions Prises**

- Pour résoudre le problème de sécurité lié à la clé de l'API Google Maps, nous avons stocké la clé dans un environnement sécurisé.
- Pour surmonter la difficulté de compréhension de la structuration des fichiers du serveur, nous avons étudié la documentation de Svelte Kit.
- Pour résoudre l'obstacle rencontré lors de la récupération du token du compte utilisateur, nous avons analysé le code de l'autre équipe.
- Pour s'adapter au code TypeScript et à la syntaxe ESLint, nous avons investi du temps dans la lecture de la documentation et la pratique régulière.

### **4) Résultats et Leçons Apprises**

- Les solutions adoptées ont permis de résoudre efficacement les obstacles rencontrés.
- Nous avons appris l'importance de l'investissement dans la documentation et la pratique pour surmonter les défis techniques.

- Tirer parti des ressources disponibles, telles que la documentation et l'analyse du code existant, peut grandement aider à résoudre les problèmes rencontrés.
- L'adaptabilité et la volonté d'apprendre de nouvelles technologies sont des compétences précieuses dans un environnement de développement logiciel.

## **Carnet des Risques**

### **1) Identification des Risques**

- Manque de clarté dans la planification initiale du projet.
- Duplication des données dans le système.
- Code source non optimisé et manque de propreté.
- Chemins mal acheminés dans l'application.
- Processus de réalisation des issues effectué avant la résolution des problèmes.
- Absence de documentation détaillée du code.

### **2) Description des Risques**

- Le manque de clarté dans la planification initiale peut entraîner des retards et des erreurs dans la réalisation du projet.
- La duplication des données peut conduire à des incohérences et à une complexité accrue dans le système.
- Un code source non optimisé rendra la maintenance et l'évolution du système plus difficiles.
- Des chemins mal acheminés peuvent entraîner des erreurs fonctionnelles et une mauvaise expérience utilisateur.
- Le processus de réalisation des issues avant la résolution des problèmes peut entraîner des doublons dans les tests et une inefficacité dans le développement.
- L'absence de documentation détaillée peut compliquer la compréhension et la maintenance du code pour les développeurs futurs.

### **3) Probabilité et Impact**

- Manque de clarté dans la planification : Probabilité moyenne, Impact élevé.
- Duplication des données : Probabilité faible à moyenne, Impact moyen à élevé.
- Code source non optimisé : Probabilité moyenne, Impact moyen.
- Chemins mal acheminés : Probabilité faible à moyenne, Impact moyen à élevé.
- Processus de réalisation des issues : Probabilité moyenne, Impact moyen.
- Absence de documentation détaillée : Probabilité moyenne, Impact moyen.

### **4) Plans d'Atténuation**

- Pour le manque de clarté dans la planification, améliorer les définitions de sprint et estimer avec précision les délais.

- Pour la duplication des données, effectuer une analyse approfondie de la base de données et éliminer toute redondance.
- Pour le code source non optimisé, mettre en place une stratégie de refactoring et utiliser des outils d'analyse statique du code.
- Pour les chemins mal acheminés, effectuer une revue de l'architecture de l'application et des tests d'intégration approfondis.
- Pour le processus de réalisation des issues, revoir la séquence de développement pour résoudre les problèmes de doublons dans les tests.
- Pour l'absence de documentation détaillée, introduire une pratique de documentation systématique, telle que l'utilisation de javadoc.

## **5) Suivi et Gestion des Risques**

- Mettre en place un processus de surveillance régulière des risques identifiés.
- Réviser périodiquement les plans d'atténuation pour s'assurer de leur efficacité.
- Encourager une culture de transparence et de communication ouverte pour signaler les risques potentiels à tout moment.
- Intégrer la gestion des risques dans les revues de projet régulières pour garantir une prise de décision éclairée.

## **C- Revues d'itération (démonstrations)**

### **Revues d'itération**

#### **1) Exécution avec succès de l'ensemble des tests d'acceptation**

Lien vers le pipeline : <https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/pipelines?scope=all&page=1>

#### **2) Exécution des Tests d'Acceptation**

Lien vers les tests : [https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/pipelines/89784/test\\_report](https://gitlab.info.uqam.ca/inm5151-h24/touristes-de-salon-expire/-/pipelines/89784/test_report)

## **D- Rapports de suivi**

Version 1 page en photo & version lisible Apres

# Rapport de Suivi de Projet - Itération #1 - Équipe : Les Touristes

## Faits Saillants

- Déploiement réussi des fonctionnalités de création manuelle et d'importation GPX.
- Collaboration efficace au sein de l'équipe, avec une répartition claire des tâches.
- Résolution rapide des problèmes rencontrés lors de l'intégration des fonctionnalités.

**But :** Mettre en place les fonctionnalités de base de l'application, notamment la création manuelle d'activités et l'importation d'activités à partir de fichiers GPX.

## Engagement

Élément du Carnet	Taille Engagée	Taille Réalisée	État
<b>Création Manuelle</b>	38 heures	40 heures	Complété
<b>Importation GPX</b>	35 heures	35 heures	Complété
<b>Affichage d'Activités</b>	4 heures	5 heures	Complété
<b>Modification d'Activités</b>	10 heures	15 heures	En cours
<b>Suppression d'Activités</b>	10 heures	15 heures	En cours
<b>Gestion des Activités</b>	30 heures	35 heures	Complété
<b>Correction de Bugs</b>	18 heures	18 heures	Complété
<b>Total</b>	135 heures	163 heures	

## État des Risques et Obstacles

**Nouveaux** : Risque de duplication des données : Il existe un risque accru de duplication des données dans le système en raison de la structure actuelle de la base de données. Cela pourrait entraîner des incohérences et des erreurs dans les informations enregistrées.

Risque de perte de données en raison d'une sauvegarde insuffisante du système

**Modifiés** : Risque de qualité du code

### Plan d'Amélioration

Amélioration de la documentation du code pour faciliter la compréhension future.

Révision des tests pour une meilleure couverture des fonctionnalités.

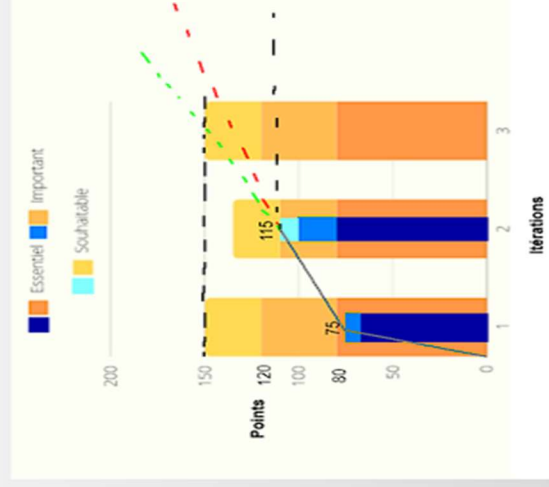
Mise en place d'un processus de revue de code pour assurer la qualité du code produit.

**Supprimés** : Le risque de non-disponibilité des ressources matérielles a été supprimé car toutes les ressources nécessaires sont désormais disponibles.

Tous les risques ont été identifiés et traités avec succès

**Nouveaux Obstacles** : Aucun Obstacle rencontré jusqu'à présent

## État d'Avancement



## État de la Qualité

- Nombre de bogues : 5
- Impact : Mineur, Raisons : Les bogues détectés sont principalement des problèmes d'affichage mineurs et des erreurs de saisie dans les formulaires. Ils n'affectent pas significativement la fonctionnalité principale de l'application, mais nécessitent une correction pour améliorer l'expérience utilisateur.

**Explications des écarts** : Il est possible que certaines fonctionnalités aient été modifiées, retirées ou simplifiées par rapport à l'estimation initiale, entraînant une réduction du temps et des coûts nécessaires pour compléter le projet

État	Effort (hres)	Coûts (\$)
<b>Au début de l'itération</b>	69 heures	8 280,00\$
<b>Cette itération</b>	140 heures	16 800,00\$
<b>Cumulatif à date</b>	209 heures	25 080,00\$
<b>Estimation pour terminer</b>	16 heures	1 440,00\$
<b>Total prévu</b>	225 heures	26 520,00\$
<b>Budget initial</b>	250 heures	30 000,00\$
<b>Écart*</b>	-25 heures	+3 000,00\$



**Rapport de Suivi de Projet - Itération #1 - Équipe : Les Touristes****2. But de l'itération:**

- Le but de cette itération était de mettre en place les fonctionnalités de base de l'application, notamment la création manuelle d'activités et l'importation d'activités à partir de fichiers GPX.

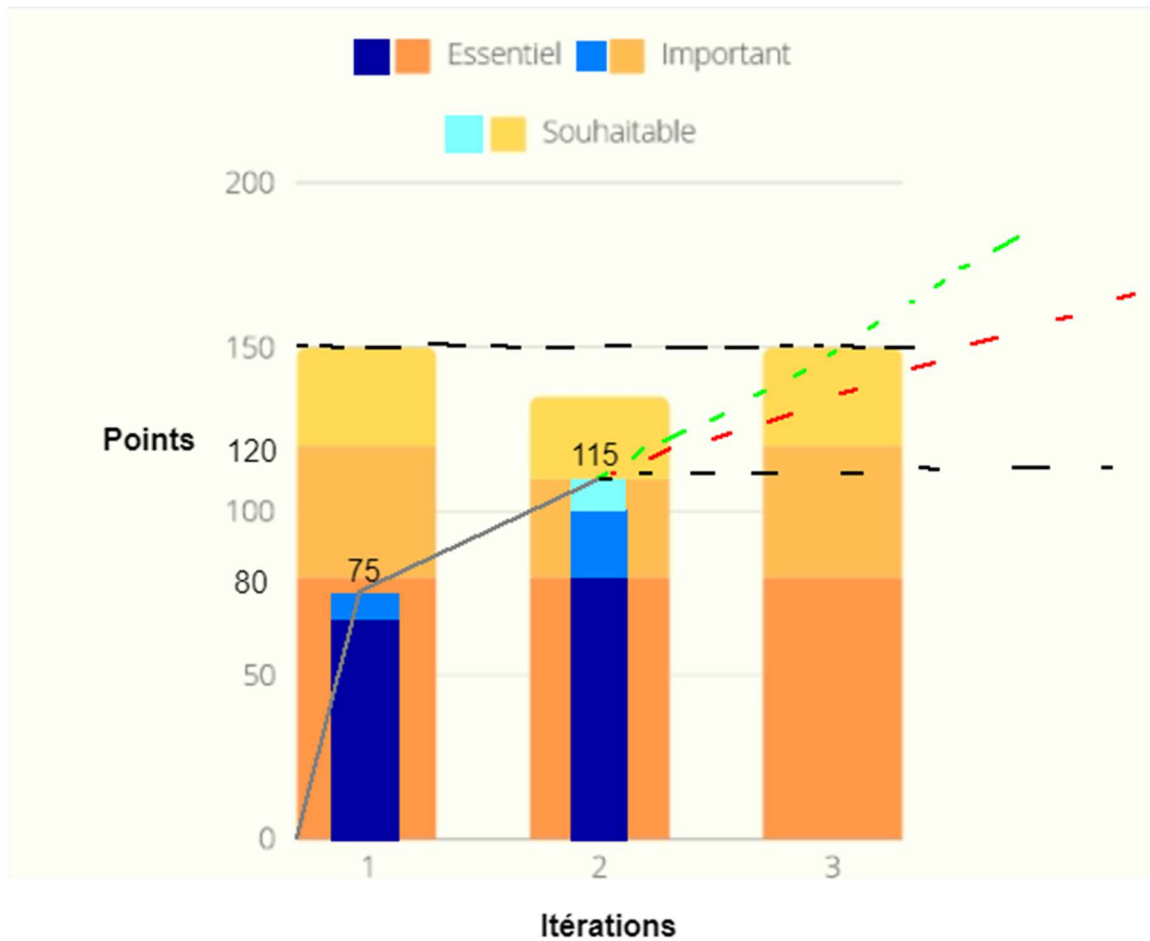
**3. Engagement:**

Élément du Carnet	Taille Engagée	Taille Réalisée	État
Création Manuelle	38 heures	40 heures	Complété
Importation GPX	35 heures	35 heures	Complété
Affichage d'Activités	4 heures	5 heures	Complété
Modification d'Activités	10 heures	15 heures	En cours
Suppression d'Activités	10 heures	15 heures	En cours
Gestion des Activités	30 heures	35 heures	Complété
Correction de Bugs	18 heures	18 eures	Complété
<b>Total</b>	135 heures	163 heures	

**4. Faits Saillants:**

- Déploiement réussi des fonctionnalités de création manuelle et d'importation GPX.
- Collaboration efficace au sein de l'équipe, avec une répartition claire des tâches.
- Résolution rapide des problèmes rencontrés lors de l'intégration des fonctionnalités.

**5. État d'Avancement Graphique:**



## 6. État de l'Effort et des Coûts:

État	Effort (hres)	Coûts (\$)
Au début de l'itération	69 heures	8 280,00\$
Cette itération	140 heures	16 800,00\$
Cumulatif à date	209 heures	25 080,00\$
Estimation pour terminer	16 heures	1 440,00\$
Total prévu	225 heures	26 520,00\$
Budget initial	250 heures	30 000,00\$
Écart*	-25 heures	+3 000,00\$

**Explications des écarts :** Il est possible que certaines fonctionnalités aient été modifiées, retirées ou simplifiées par rapport à l'estimation initiale, entraînant une réduction du temps et des coûts nécessaires pour compléter le projet

## 7. État de la Qualité:

- Nombre de bogues : 5

- Impact : Mineur

- Raisons : Les bogues détectés sont principalement des problèmes d'affichage mineurs et des erreurs de saisie dans les formulaires. Ils n'affectent pas significativement la fonctionnalité principale de l'application, mais nécessitent une correction pour améliorer l'expérience utilisateur.

## **8. État des Risques et Obstacles:**

- Nouveaux Risques:

- Risque de duplication des données : Il existe un risque accru de duplication des données dans le système en raison de la structure actuelle de la base de données. Cela pourrait entraîner des incohérences et des erreurs dans les informations enregistrées.
- Risque de perte de données en raison d'une sauvegarde insuffisante du système.

- Risques Modifiés:

- Risque de qualité du code : Bien que des efforts aient été déployés pour maintenir un code propre et bien structuré, le risque de qualité du code persiste en raison de la complexité croissante de certaines fonctionnalités. Il est important de continuer à surveiller et à améliorer la qualité du code pour éviter les problèmes de maintenance à long terme.

- Risques Supprimés:

- Le risque de non-disponibilité des ressources matérielles a été supprimé car toutes les ressources nécessaires sont désormais disponibles.
- Tous les risques ont été identifiés et traités avec succès

- Nouveaux Obstacles:

- Aucun Obstacle rencontré jusqu'à présent.

## **9. Plan d'Amélioration:**

- Amélioration de la documentation du code pour faciliter la compréhension future.
- Révision des tests pour une meilleure couverture des fonctionnalités.
- Mise en place d'un processus de revue de code pour assurer la qualité du code produit.