

# Analyse Critique du Déroulement du Projet:

## *1. Découpage et Répartition du Travail en Équipe*

Le projet a été divisé en plusieurs tâches, ce qui a permis de mieux organiser le travail et de se concentrer sur des parties spécifiques du système. L'une des forces de ce découpage était la répartition des questions impaires et paires entre nous, les deux membres de l'équipe, permettant à chacun de se concentrer sur un sous-ensemble précis des requêtes.

### **Répartition des tâches :**

- **Imane** : a pris en charge les questions impaires, soit les requêtes concernant la disponibilité des logements, les résidents partageant un logement, les prolongations de séjour et les types de logements les plus demandés. Ces tâches impliquaient de manipuler les données des résidents, des réservations, et des logements, ce qui exigeait une compréhension approfondie des relations entre ces entités.
- **Guilaye** : a pris en charge les questions paires, telles que la gestion des réservations, l'attribution des logements, le suivi des interventions de maintenance et l'organisation des événements communautaires. Ces questions étaient également importantes, mais elles nécessitaient plus d'attention à l'aspect opérationnel du système.
- Et concernant les autres éléments à rendre à la fin du projet comme la reformulation complète du projet, la vidéo, le schéma relationnel ..., nous les avons réalisés ensemble. Vous pouvez tous voir sur notre répertoire sur GitHub pour plus d'informations.

### **Points forts du découpage :**

- **Clarté des responsabilités** : Chaque membre savait précisément ce qu'il devait faire, ce qui a permis de ne pas se marcher sur les pieds.

- **Complémentarité** : Les tâches étaient bien complémentaires, et les deux membres pouvaient travailler sur des aspects différents du projet sans se chevaucher.

#### Points à améliorer :

- **Collaboration sur les détails complexes** : À certains moments, il aurait été utile de se réunir pour discuter des solutions avant de commencer à coder, notamment en ce qui concerne l'optimisation des réservations ou la gestion des prolongations de séjour, qui sont des sujets complexes qui ont nécessité une réflexion collective.

## 2. Difficultés Rencontrées

- **Complexité de l'optimisation des réservations** : L'une des difficultés majeures a été de définir des règles d'optimisation pour l'occupation des logements. Lorsque plusieurs résidents souhaitaient réserver un logement pendant une période, il fallait s'assurer que la gestion des priorités et des prolongations de séjour ne nuise pas à l'occupation optimale des logements.  
=> *Solution* : Nous avons dû expérimenter avec plusieurs approches dans les requêtes SQL et gérer des cas complexes, comme les résidents qui prolongeaient leur séjour à la dernière minute.
- **Gestion des interactions entre résidents** : La gestion des conflits ou des interactions entre résidents a été sous-estimée au départ. L'outil devait permettre de suivre non seulement la présence des résidents, mais aussi leurs activités communautaires et potentiellement des conflits. Une fois ce besoin identifié, il a fallu concevoir un système efficace de suivi.
- **Problèmes de données manquantes** : Lors de la mise en place de la base de données, nous avons constaté qu'il était parfois difficile de garantir la cohérence des données, notamment pour les informations sur les équipements et les événements. Certaines informations étaient manquantes dans les tables de test, ce qui a retardé l'exécution des requêtes et l'analyse.  
=> *Solution* : Utiliser un jeu de données plus complet ou remplir manuellement les tables pour les tests aurait été plus efficace.

## 3. Réflexion sur des Améliorations Éventuelles de l'Organisation des Tâches

Si c'était à refaire, voici quelques ajustements que nous pourrions apporter à l'organisation du projet :

- **Mieux définir les interactions entre les tables dès le départ** : Avant de commencer à écrire des requêtes complexes, il aurait été utile de passer plus de temps à définir les relations entre les différentes tables et à clarifier comment ces relations seraient utilisées dans les requêtes. Une modélisation plus approfondie des interactions entre les entités (comme les interactions des résidents ou les incidents de maintenance) aurait facilité la rédaction des requêtes plus tard dans le projet.
- **Tests et validation des données** : Il aurait été judicieux de tester les requêtes SQL sur un jeu de données plus représentatif dès le début, afin de repérer d'éventuels problèmes de conception ou d'incohérence plus tôt. Cela aurait également permis de vérifier si les requêtes répondaient bien aux besoins définis dans les questions.
- **Réunions régulières de suivi** : Bien que le découpage des tâches ait été clair, avoir plus des réunions ,plus régulières auraient permis de mieux ajuster le travail en cours, surtout sur des tâches complexes, et d'éviter des erreurs d'interprétation sur la portée des différentes questions, mais ceci a été résolu par la communication sur "Discord" .
- **Documentation plus détaillée** : La documentation des étapes clés, des décisions prises, et des résultats obtenus aurait facilité la gestion du projet et la compréhension des choix techniques. Cela aurait également été utile si on devait reprendre le projet à un moment donné.

#### ***4. Aspects Bien Réussis et Éléments Clés de la Réussite***

- **Bonne répartition des tâches** : La division du travail en deux grandes catégories de tâches a permis à chaque membre de se concentrer sur un aspect particulier du projet . Même si ceci rend la gestion du temps plus difficile ,mais les tâches étaient suffisamment bien définies pour ne pas causer de confusion et de retard.
- **Structure claire de la base de données** : La conception de la base de données avec des tables bien définies a permis d'avoir une architecture solide dès le départ. Cela a facilité la création des requêtes SQL nécessaires pour répondre aux questions du projet.
- **Communication efficace** : Malgré les difficultés, notre équipe a bien communiqué pour résoudre les problèmes rencontrés. La flexibilité et la capacité à s'adapter rapidement aux besoins du projet ont permis de surmonter les défis rencontrés, notamment ceux liés à l'optimisation des réservations et à la gestion des prolongations de séjour.

## 5. Conclusion : “Si C’était à Refaire”

Si nous devions refaire ce projet, nous nous concentrerions sur les points suivants :

- **Modélisation plus approfondie en amont** : Consacrer plus de temps à la planification et à la définition des relations entre les entités avant de commencer à coder les requêtes.
- **Meilleure gestion des interactions entre résidents** : Ajouter une gestion des conflits ou des problèmes sociaux entre les résidents dès le début du projet pour ne pas devoir l'intégrer en dernière minute.
- **Test et validation plus rigoureux des données** : Avoir un jeu de données plus réaliste et complet pour tester toutes les fonctionnalités du système de manière plus approfondie.
- **Suivi continu** : Des réunions de suivi régulières pour évaluer l'avancement du projet et ajuster la direction en fonction des obstacles rencontrés.

Dans l'ensemble, bien que le projet ait comporté des défis techniques, il a permis de bien comprendre l'importance de la planification, de la collaboration et de la flexibilité pour mener à bien un projet de cette envergure.