Rapport de Projet Informatique — Jeu de Cartes

# Introduction

Dans le cadre de notre projet informatique, notre groupe, composé de Rayane, Mehdi et Adam, a conçu un jeu de cartes numérique. Le but de ce jeu était de simuler une partie avec plusieurs joueurs, une pioche, une main, des tours de jeu, une défausse et un calcul final de scores.  
Notre progression s’est étalée sur six semaines, et ce rapport présente chaque étape de notre développement, les difficultés rencontrées, les solutions apportées, et la manière dont nous avons réparti les tâches entre les membres.

# Semaine 1 : Lancement du projet et idéation

La première semaine a été consacrée à la définition du sujet et à la compréhension des enjeux du projet.  
Adam a suggéré de développer un jeu de cartes, car cela permettait d’intégrer à la fois de la logique algorithmique (distributions, tours, scores) et des éléments d'affichage. Il a aussi esquissé une structure technique générale, avec l'idée de séparer les fonctions principales pour faciliter le développement par étapes.  
Rayane a contribué à définir le fonctionnement précis du jeu : nombre de joueurs, règles, conditions de victoire. Il s’est assuré que le concept soit simple, clair et techniquement faisable dans les délais.  
Mehdi a commencé à noter les éléments clés pour la rédaction du README final, tout en créant des diagrammes simplifiés pour structurer le travail à venir. Il a aussi commencé à réfléchir à la manière dont on allait documenter les fonctions au fil de leur implémentation.

# Semaine 2 : Codage des premières fonctions

Cette semaine, nous avons entamé la partie technique du projet. L'objectif était d'obtenir une base fonctionnelle : une pioche, des cartes, et la distribution aux joueurs.  
Rayane a développé la fonction de génération de la pioche, en créant les différentes cartes du jeu avec leurs attributs. Il a également codé le mélange de la pioche de façon aléatoire, élément essentiel pour rendre chaque partie unique.  
Adam a implémenté la logique de distribution : chaque joueur devait recevoir le bon nombre de cartes. Il a aussi mis en place les premières structures de données pour stocker les mains de chaque joueur, ainsi que des tests initiaux pour s’assurer de la validité de la distribution.  
Mehdi, de son côté, a suivi les étapes de codage et vérifié la cohérence des noms de variables et des fonctions. Il a commenté certaines parties de code et a commencé à rédiger un premier jet du README, en se concentrant sur la description du projet et des premières fonctionnalités.

# Semaine 3 : Tours de jeu et interactions

Nous avons travaillé sur la gestion des tours : comment chaque joueur joue, comment les cartes sont posées ou utilisées, et dans quel ordre les joueurs interagissent.  
Adam a programmé la logique complète d’un tour de jeu, y compris la gestion des indices de tour et des effets de carte (comme les cartes visibles en début de partie). Il a aussi pensé à des fonctions modulaires pour permettre des ajustements faciles si on voulait changer les règles plus tard.  
Rayane a amélioré l’affichage : au lieu d’avoir uniquement du texte brut, il a encadré chaque carte avec des bordures pour une meilleure lisibilité, rendant l’interface plus agréable. Il a aussi validé visuellement que le déroulement du tour fonctionnait bien.  
Mehdi a poursuivi la documentation en décrivant la structure des tours et les fonctions associées. Il a également commencé à organiser les fichiers pour préparer une version plus propre du code, en vue de la présentation finale.

# Semaine 4 : Conditions de fin de partie et score

Cette semaine, nous avons intégré la fin du jeu, c’est-à-dire le moment où la pioche est vide et les joueurs ne peuvent plus jouer. Il fallait aussi calculer un score pour désigner un gagnant.  
Adam a développé la fonction de calcul des scores en fonction des cartes restantes. Il a également géré les conditions de fin de partie, en prenant en compte le cas où la pioche serait vide ou un joueur aurait terminé ses cartes. Il a centralisé cette logique dans une fonction dédiée pour la réutiliser facilement.  
Rayane s’est chargé de supprimer les portions de code devenues obsolètes après nos modifications successives. Il a également revu certaines boucles pour les simplifier.  
Mehdi, de son côté, a finalisé une première version complète du README, comprenant une explication claire du but du jeu, de son déroulement et de la logique de fin de partie.

# Semaine 5 : Structuration finale du code

Le code fonctionnait, mais nous voulions qu’il soit lisible, modulaire et bien organisé. C’était l’objectif de la cinquième semaine.  
Adam a suggéré de séparer les fonctions dans plusieurs fichiers pour une meilleure organisation (ex : pioche.c, score.c, main.c). Il a ensuite réalisé cette séparation et a fait des ajustements pour que tous les fichiers communiquent correctement entre eux.  
Rayane a enrichi les commentaires dans chaque fichier, afin que le code soit compréhensible même pour quelqu’un qui ne connaît pas le projet. Il a aussi fait quelques corrections sur l’affichage.  
Mehdi a relu tout le code et le README pour s’assurer de la cohérence entre les explications et ce que le programme faisait réellement. Il a aussi ajouté des exemples d’exécution dans le README pour que l’utilisateur sache à quoi s’attendre.

# Semaine 6 : Derniers ajustements et test final

La dernière semaine était dédiée à la finition du projet. Il fallait vérifier le bon fonctionnement du jeu, corriger les éventuels bugs et préparer une présentation propre.  
Adam a fait une passe complète sur le code, détectant une petite erreur dans la condition de fin de partie qu’il a corrigée. Il a également uniformisé les noms de variables et les messages affichés.  
Rayane a réalisé plusieurs tests complets pour s'assurer que toutes les situations étaient prises en compte : égalité, joueurs avec beaucoup de cartes, etc. Il a aussi préparé un court script de démonstration.  
Mehdi a relu une dernière fois le README, corrigé les fautes et veillé à ce que toutes les parties soient bien illustrées. Il a aussi aidé à réorganiser les fichiers dans un dossier propre pour la livraison.

# Conclusion

Ce projet nous a permis de travailler en équipe sur toutes les étapes du développement d’un programme complet, de la conception à la documentation.  
Rayane s’est distingué par sa rigueur dans la structure du jeu et les tests, Mehdi par son sérieux dans la documentation et l'organisation, et Adam par son implication dans la logique du code et l’implémentation technique.  
Ensemble, nous avons réussi à livrer un jeu fonctionnel, clair, bien documenté, et fidèle au cahier des charges.