

Projet 1 :

Développement d'un jeu de Blackjack

4 novemvre 2025

Table des matières

1	Introduction	2
1.1	Présentation générale du projet	2
1.2	Problématique et points clés	2
1.3	Présentation du plan du rapport	3
2	Structuration du projet	4
2.1	Analyse des besoins	4
2.2	Fonctionnalités implémentées	4
2.3	Organisation du travail	4
2.4	Frise chronologique du projet	5
3	Éléments techniques	6
3.1	Structures de données utilisées	6
3.2	Algorithmes notables	6
4	Architecture du projet	7
4.1	Contrôleur : orchestration de la simulation	7
4.2	Vue : interface graphique réactive	7
4.3	Synthèse : une architecture MV-C	7
5	expérimentations et usages	7
5.1	Cas d'utilisation :	7
5.2	Cas d'utilisation :	7
5.3	Cas d'utilisation :	7
5.4	Cas d'utilisation :	7
5.5	Cas d'utilisation :	7
5.6	Cas d'utilisation :	7
5.7	Performances	7
6	Conclusion	8

1 Introduction

1.1 Présentation générale du projet

1.2 Problématique et points clés

- **Créer une interface utilisateur explicite et pédagogique :** L'interface graphique devait à la fois représenter l'état de la mémoire, l'avancement de chaque guerrier, et permettre d'observer l'évolution du match en temps réel. Cela implique une gestion précise des événements et des notifications entre le modèle et la vue (via le pattern Observer).
- **Maintenir la modularité du code :** Enfin, toute cette architecture devait rester claire, maintenable et évolutive. Le respect du modèle **MV-C**, ainsi qu'une séparation stricte des responsabilités, ont été déterminants pour conserver une bonne lisibilité du projet.

1.3 Présentation du plan du rapport

Ce rapport a pour objectif de retracer de manière claire et progressive le développement du projet **Blackjack**. Il est structuré en six sections, chacune correspondant à une phase essentielle du projet :

- **Section 1 – Introduction** : pose le contexte du projet, ses objectifs pédagogiques et les motivations ayant guidé notre équipe.
- **Section 2 – Structuration du projet** : décrit les besoins fonctionnels et techniques, les fonctionnalités envisagées, ainsi que l'organisation du travail collaboratif.
- **Section 3 – Éléments techniques** : détaille les algorithmes et structures de données au cœur de la simulation, notamment le moteur d'exécution, la mémoire circulaire, et les opérateurs génétiques.
- **Section 4 – Architecture du projet** : propose une vue d'ensemble de l'architecture logicielle, articulée autour du patron MVC. Cette partie s'appuie sur des diagrammes UML pour illustrer les relations entre les différentes classes et modules.
- **Section 5 – Expérimentations et usages** : illustre l'utilisation concrète du simulateur à travers des cas d'usage commentés et des captures d'écran, en soulignant les résultats observés.
- **Section 6 – Conclusion** : dresse un bilan général du projet, récapitule les fonctionnalités implémentées et propose plusieurs axes d'amélioration pour la suite.

2 Structuration du projet

2.1 Analyse des besoins

2.2 Fonctionnalités implémentées

2.3 Organisation du travail

Le travail a été réparti entre les trois membres du groupe de manière à valoriser les compétences de chacun :

Une collaboration régulière a été assurée via un dépôt Git, avec un suivi des commits et une validation croisée du code. Des séances hebdomadaires de synchronisation ont permis de maintenir une vision partagée de l'avancement.

2.4 Frise chronologique du projet

La planification et la progression du projet de Blackjack se sont organisées autour de jalons bien définis. La frise suivante illustre les grandes étapes de développement, depuis la compréhension du sujet jusqu'à la rédaction finale du rapport. Chaque étape a mobilisé les membres du groupe selon leur domaine d'expertise et les modules en charge.

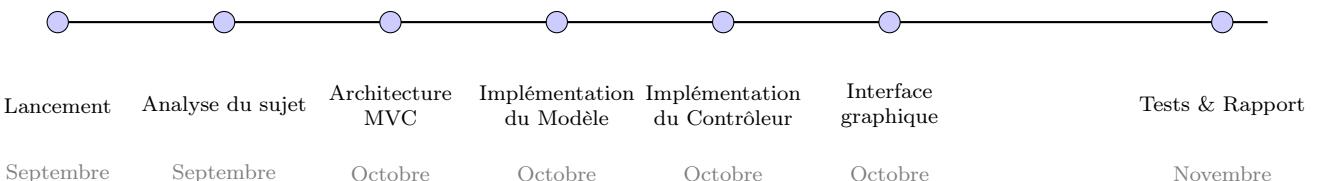


FIGURE 1 – Frise chronologique des grandes étapes du projet

3 Éléments techniques

3.1 Structures de données utilisées

3.2 Algorithmes notables

4. Évolution génétique multi-niveaux

6. Observer Pattern pour l'affichage

4 Architecture du projet

4.1 Contrôleur : orchestration de la simulation

4.2 Vue : interface graphique réactive

Communication modèle ↔ vue

4.3 Synthèse : une architecture MV-C

5 expérimentations et usages

5.1 Cas d'utilisation :

5.2 Cas d'utilisation :

5.3 Cas d'utilisation :

5.4 Cas d'utilisation :

5.5 Cas d'utilisation :

5.6 Cas d'utilisation :

5.7 Performances

6 Conclusion

Récapitulatif des fonctionnalités principales

Propositions d'améliorations