

# Cahier des Charges – Projet Morpion

## Table des matières

<b>1. Introduction.....</b>	<b>2</b>
1.1 Contexte.....	2
1.2 Objectifs .....	2
<b>2. Description du Jeu .....</b>	<b>2</b>
2.1 Règles du Jeu.....	2
2.2 Fonctionnalités Principales.....	2
<b>3. Spécifications Techniques.....</b>	<b>2</b>
3.1 Langage de Programmation.....	2
3.2 Architecture.....	3
3.3 Dépendances et Outils.....	3
<b>4. Contraintes et Normes .....</b>	<b>3</b>
4.1 Contraintes Techniques.....	3
4.2 Normes de Développement.....	3
<b>5. Planification et Organisation .....</b>	<b>3</b>
5.1 Répartition des Tâches .....	3
5.2 Suivi et Dépôt GitHub .....	4
<b>6. Conclusion .....</b>	<b>4</b>

# 1. Introduction

## 1.1 Contexte

Ce projet s'inscrit dans le cadre du cours "Qualité de développement" et a pour objectif de réaliser un jeu de morpion en respectant les bonnes pratiques de développement, les normes de codage et en assurant une couverture de tests adéquate.

## 1.2 Objectifs

- Développer un jeu de morpion fonctionnel en respectant les normes de codage.
- Documenter le projet avec un cahier des charges, un plan de test et une documentation technique.
- Assurer la qualité du code via des tests unitaires et des revues de code sur GitHub.

# 2. Description du Jeu

## 2.1 Règles du Jeu

Le morpion est un jeu de stratégie à deux joueurs qui se joue sur une grille de 3x3 cases. Chaque joueur joue à tour de rôle en plaçant son symbole (X ou O) dans une case vide. Le premier joueur à aligner trois de ses symboles horizontalement, verticalement ou en diagonale remporte la partie. Si toutes les cases sont remplies sans qu'un joueur ne l'emporte, la partie est déclarée nulle.

## 2.2 Fonctionnalités Principales

- Interface utilisateur simple permettant de jouer une partie.
- Gestion des tours de jeu.
- Détection de victoire ou match nul.
- Réinitialisation de la partie.
- Affichage des messages de victoire, défaite ou match nul.

# 3. Spécifications Techniques

## 3.1 Langage de Programmation

Le langage choisi pour le développement du jeu est Python.

## 3.2 Architecture

- Une interface graphique simple.
- Une logique de jeu indépendante de l'interface.
- Gestion des entrées des joueurs.

## 3.3 Dépendances et Outils

- Bibliothèque Pygame utilisée.
- GitHub pour la gestion du code source et des pull requests.
- Bibliothèque Unittest utilisée pour les tests unitaires.

# 4. Contraintes et Normes

## 4.1 Contraintes Techniques

- Le jeu doit être exécutable sur Windows.
- L'interface doit être intuitive et simple d'utilisation.
- Le code doit être bien structuré et commenté.

## 4.2 Normes de Développement

- Respect des conventions de codage du langage choisi.
- Utilisation de branches Git pour chaque tâche.
- Documentation du code et rédaction de tests unitaires.

# 5. Planification et Organisation

## 5.1 Répartition des Tâches

Chaque membre du groupe est responsable d'une partie spécifique du projet :

- Développement de la logique du jeu.
- Implémentation de l'interface.
- Rédaction du cahier des charges et de la documentation technique.
- Conception et exécution des tests unitaires.

## 5.2 Suivi et Dépôt GitHub

Le code source est versionné et hébergé sur GitHub à l'adresse suivante :  
<https://github.com/Bastien113/Deshayes-Soulat-Gramont-Pade-Projet2024>

Des commits réguliers et des revues de code via pull requests sont effectués pour assurer la qualité du projet.

## 6. Conclusion

Ce cahier des charges définit les bases du projet Morpion et sert de référence pour la réalisation du jeu. Il garantit une organisation efficace et une production conforme aux attentes en termes de qualité et de rigueur de développement.