

**Nom, Prénom :**

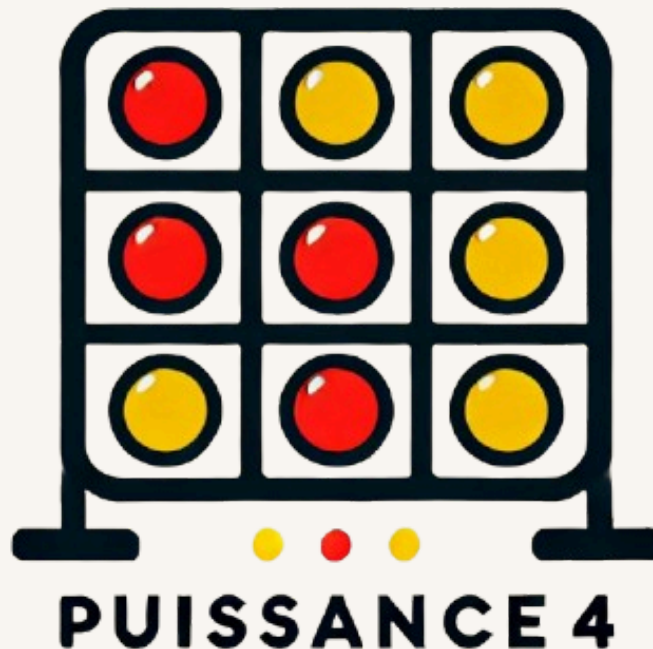
*SOARIZIKY Annaïsse*

*GROULT Marius*

*VATRIN Lorenzo*

**Groupe :**

Dev2



# Cahier des charges

Année : 2024-2025

Puissance 4 : Alignez vos jetons, dominez la partie !"

# **Contexte du projet :**

Le projet consiste à développer un jeu de Puissance 4 en utilisant Windows Forms avec un système de gestion des modes de jeu (joueur contre joueur en local et joueur contre intelligence artificielle). Le code fourni représente une partie de l'interface d'accueil, ainsi que la gestion des événements liés aux interactions de l'utilisateur (survol, clic, fermeture de fenêtre, etc.).

# **Objectifs du projet :**

- Interface graphique conviviale : L'application doit proposer une interface simple, intuitive et esthétique, permettant aux utilisateurs de naviguer entre différents modes de jeu et de configurer leurs parties facilement.
- Deux modes de jeu disponibles :
  - Joueur vs Joueur (local) : Deux joueurs humains jouent à tour de rôle sur le même appareil.
  - Joueur vs IA : Un joueur humain joue contre une intelligence artificielle.
- Gestion des interactions utilisateur : L'application doit répondre de manière fluide aux actions des utilisateurs : survol des boutons, clics, et confirmations lors de la fermeture de l'application.

# Caractéristiques fonctionnelles

## 1. Interface d'accueil :

L'interface d'accueil permet aux utilisateurs de choisir entre deux modes de jeu :

- **Joueur vs Joueur**
- **Joueur vs IA**
  - Composants requis :**
    - Un bouton principal permettant de choisir entre les modes de jeu
    - Un autre bouton permettant de choisir en la taille standard du puissance ou un autre personnalisé
    - 2 compteurs , l'un pour la largeur et l'autre pour la hauteur
  - **Exigences techniques :**
    - La fenêtre d'accueil doit avoir des dimensions dynamiques, ajustées par rapport à ses composants, avec une bordure fixe (FixedSingle).
    - Les compteurs ne doivent apparaître que si on choisi le mode personnalisé

## 2. Mode de jeu Joueur vs Joueur :

Lors du choix du mode Joueur vs Joueur:

- **Une nouvelle fenêtre s'ouvre affichant le plateau de jeu**
- **L'écran d'accueil est masqué.**
  - Composants de la page du plateau :**
    - Nous avons à chaque tour une phrase informant si c'est le joueur Jaune ou Rouge qui doit jouer
    - Le jeu s'arrête lorsque l'un des joueurs gagne ou si ni l'un ni l'autre l'emporte

## 3. Mode de jeu Joueur vs IA (JVIA) :

Lors du choix du mode Joueur vs IA :

- **Une nouvelle fenêtre s'ouvre affichant le plateau de jeu**
- **L'écran d'accueil est masqué.**
- **Composants de la page du plateau :**
  - Nous avons à chaque tour une phrase informant si c'est le joueur Jaune ou Rouge qui doit jouer
  - Le jeu s'arrête lorsque l'un des joueurs gagne ou si ni l'un ni l'autre l'emporte

## 4. Gestion du plateau de jeu Puissance 4 :

Le cœur du jeu consiste à afficher un plateau 7x6 (standard pour Puissance 4) dans lequel les joueurs placent alternativement des jetons.

### Composants du plateau de jeu :

- Grille de 7 colonnes et 6 lignes représentant le plateau
- Indicateur du tour actuel (joueur Rouge ou Jaune )
- Boutons ou zones cliquables pour insérer un jeton dans une colonne

### Fonctionnalités supplémentaires :

- Détection automatique de la victoire ou de l'égalité (vérification des alignements horizontal, vertical et diagonal).
- Animation simple lors du placement d'un jeton.

# Caractéristiques non fonctionnelles

## Interface utilisateur fluide :

- Le changement de page doit se faire sans latence perceptible.
- Les transitions (ouverture/fermeture de pages) doivent être fluides.

## Ergonomie :

- Les boutons et les zones interactives doivent être clairement visibles et bien placés.
- L'affichage doit être centré et bien proportionné, même sur différentes résolutions d'écran.

## Compatibilité :

- Le jeu doit être compatible avec les versions récentes de Windows (minimum Windows 7).
- Il doit être développé en utilisant C# et Windows Forms, avec une bonne compatibilité sur les machines de bureau.

## Qualité de l'IA :

- L'IA doit utiliser des algorithmes de base pour jouer, avec plusieurs niveaux de difficulté.
- Le temps de réponse de l'IA doit être rapide pour ne pas nuire à l'expérience de jeu.

## Contraintes techniques :

- Langage : C# avec la bibliothèque Windows Forms.
- Environnement de développement : Visual Studio (version recommandée : Visual Studio 2022 ou plus).
- Gestion des événements : Utilisation des événements standards (clic, survol, fermeture de fenêtre, etc.).
- Bibliothèque graphique : Utilisation de System.Drawing pour la gestion des couleurs et des éléments visuels.

## Livrables :

**Code source complet**

**Exécutable fonctionnel**

**Documentation :**

- Documentation technique expliquant les fonctionnalités principales du code et son organisation.
- Guide utilisateur expliquant comment installer et jouer au jeu.