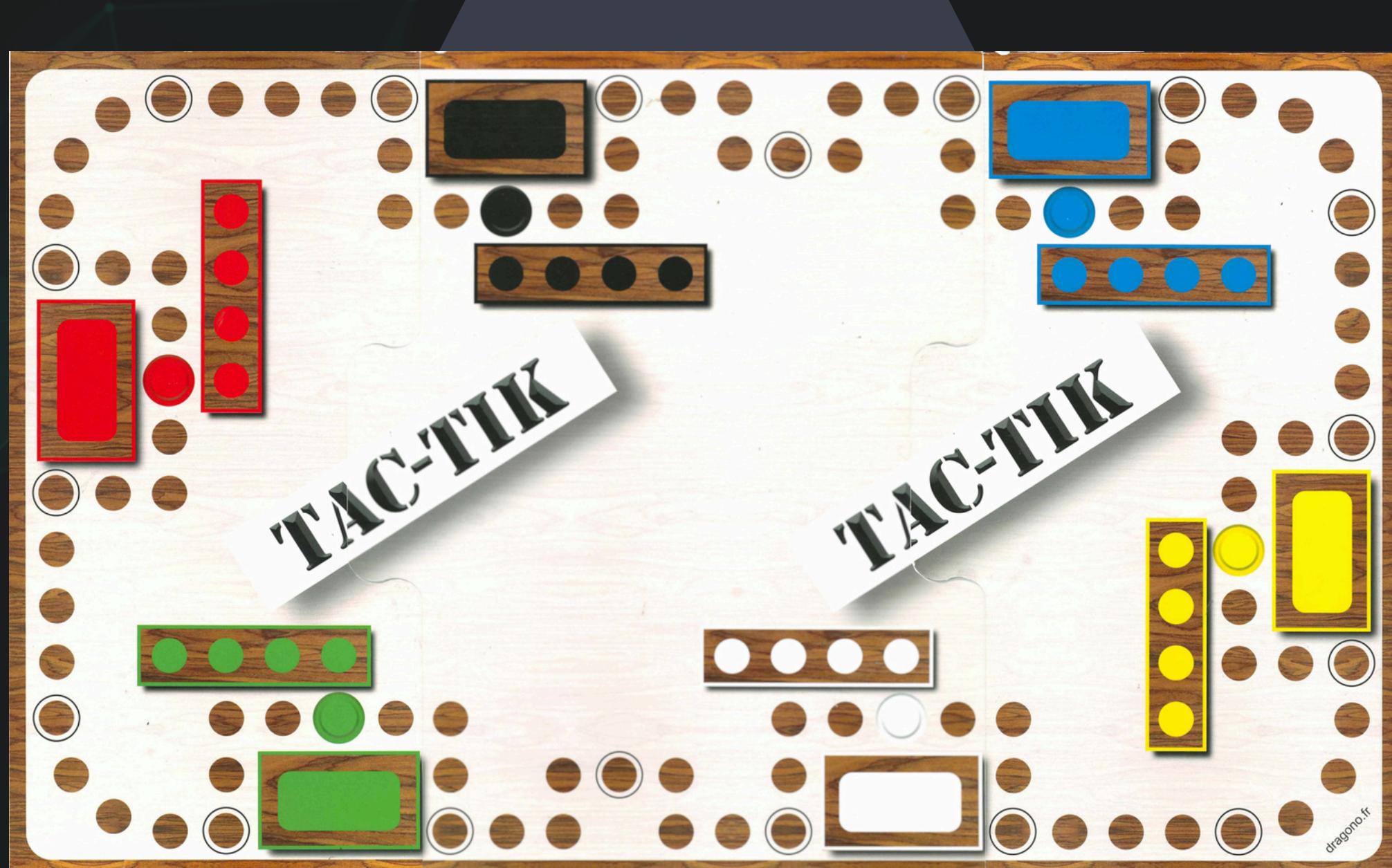


TAC-TIK

MARIUS CISERANE ET VALENTIN LAPORTE

TAC-TIK

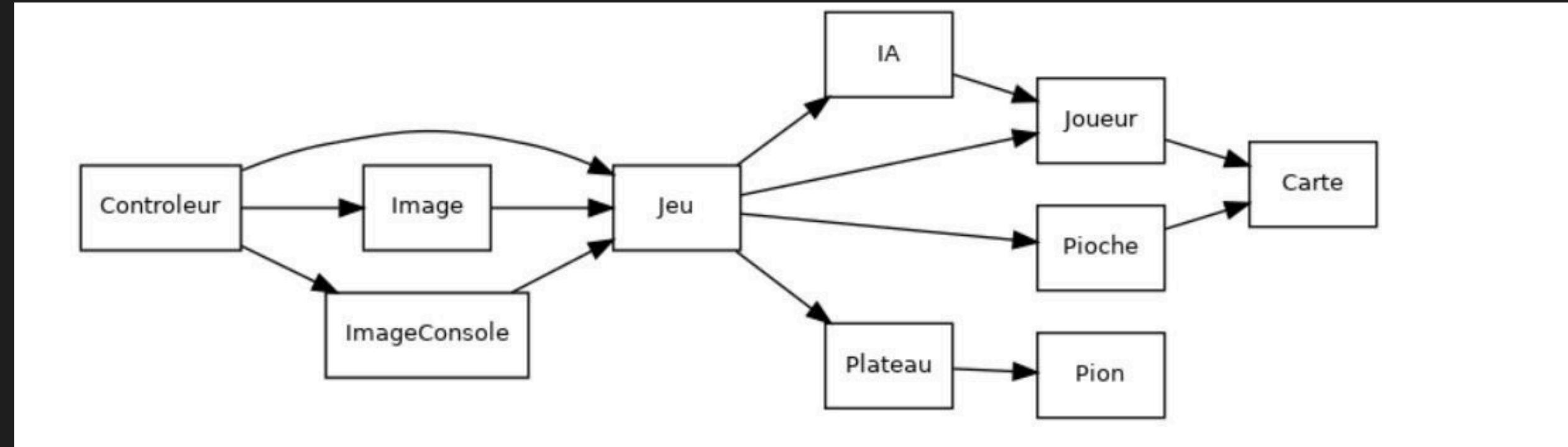
Réécriture en C++ du jeu de société TacTik



Fidèle aux règles du jeu original

Deux versions :
Console (texte)
Graphique (SDL2)

DIAGRAMME DES CLASSES (UML)



Séparation modèle vue contrôleur

CLASSE CONTROLEUR

Attributs

→ Gère l'interface et l'état de la partie :

- Pointeur vers la console et l'image graphique
- Mode graphique actif ou non (versionGraphique)
- Suivi du joueur actif (joueurActif)
- Indicateur de partie en cours (running)

Méthodes

- Initialiser et démarrer la partie (mode graphique ou non)
- Gérer les interactions joueur / IA
- Afficher des messages et menus
- Contrôler les actions
- Déetecter les événements SDL en mode graphique
- Suivre l'état du jeu (joueur actif, partie gagnée...)

| C Controleur | |
|--|--|
| <ul style="list-style-type: none">□ ImageConsole *console□ Image *graphique□ bool versionGraphique□ int joueurActif□ bool running● Controleur()● Controleur(bool affichageGraphique = false)● ~Controleur()● void initJeu(int nbJoueurs, array<bool, 6> IA)● void choixIA(int nbJoueurs, array<bool, 6> &IA)● Jeu& getJeu() const● bool getRunning() const● void setJoueurActif(int indJoueurActif)● int getIdPion(string coutMessage = "")● int getNbCase7x1(string coutMessage = "", int idPion = 0)● int jouerJoker(string coutMessage = "")● char saisirCaractere(string coutMessage = "", int choix = -1)● void afficherMessage(string coutMessage = "")● void attenteTour(bool dev = false)● bool jouerCarte(int valCarte, bool coequipier = false, bool joker = false)● int choixCarte(string coutMessage, const Joueur& joueur)● void echangeDeCartes()● void tourJoueur(bool dev = false)● void afficherVainqueur(int couleurVainqueur) const● int gestionEvent(SDL_Event event, int etapeActuel)● int afficherJeu(int etapeActuel = -1, string coutMessage = "", int choix = 2)● int afficherMenu(string coutMessage = "") | |



CLASSE JEU



Attributs

- **Nombre de joueurs (nbJoueurs)**
- **Plateau de jeu (plateau)**
- **Pioche de cartes (pioche)**
- **Joueurs et pions sous forme de pointeurs**

Méthodes

- **Initialisation et gestion des joueurs et pions**
- **Distribution et manipulation des cartes**
- **Déroulement de la partie**
- **Vérification des règles**
- **Gestion des cas de test avec testRegression()**

CLASSE IMAGE



Attributs

→ Stocke les éléments **SDL** nécessaires à l'affichage :

- **Fenêtre (window), rendu (renderer)**
- **Textures pour le plateau, tas et cartes**
- **Dimensions et coordonnées du plateau et des pions**

Méthodes

- **Initialisation du jeu et des joueurs**
- **Dessins graphiques : cercles, triangles, pions**
- **Gestion des clics et boutons**
- **Affichage des textes et menus**
- **Interactions joueur / IA**

| C Image | |
|---------|---|
| □ | Jeu *jeu |
| □ | SDL_Window *window |
| □ | SDL_Renderer *renderer |
| □ | SDL_Texture *texturePlateau |
| □ | SDL_Texture *textureTas |
| □ | SDL_Texture *textureCartes[4] |
| □ | SDL_Texture *listTexture[15] |
| □ | SDL_Rect RectMain[4] |
| □ | SDL_Rect RectTas |
| □ | TTF_Font *m_font |
| □ | SDL_Surface *textSurface |
| □ | int dimx, dimy |
| □ | int imgWidth, imgHeight |
| □ | float zoom |
| □ | int nbJ |
| □ | int *coordonnees[2] |
| □ | int *coordonneesMaison[2] |
| □ | int *coordonneesReserve[2] |
| ● | Image() |
| ● | ~Image() |
| ● | void initJeu(int nbJoueurs, array<bool, 6> IA) |
| ● | void debugCoordonnees() const |
| ● | Jeu& getJeu() const |
| ● | float getZoom() const |
| ● | int getImgWidth() const |
| ● | int getIndicePion(int posx, int posy) const |
| ● | int getIndiceCase(int posx, int posy, int joueurActif = -1) const |
| ● | int getBouton(int posx, int posy) const |
| ● | void dessineCercle(int couleur, int x, int y) const |
| ● | void dessineTriangle(int couleur, int x, int y) const |
| ● | void afficherPions() const |
| ● | void afficherReserve() const |
| ● | void afficherMaison() const |
| ● | void setTextureCartes(int idJoueur) |
| ● | void grossissement(bool positif) |
| ● | void setTextSurface(string message, int couleur) |
| ● | void setCouleur(int couleur) const |
| ● | void afficherBoutons(int choix) const |
| ● | void afficherJoker(SDL_Rect tab[13]) const |
| ● | int selectionnerValJoker(int joueurActif, string message) |
| ● | int afficherMenu(string message) |
| ● | char choixIA(string message) |
| ● | void afficher(int joueurActif, string message) |

CONCLUSION

1

**Respect du cahier
des charges**

Difficultés rencontrées

2

3

**Améliorations
possibles**

MERCI POUR VOTRE ATTENTION

ANNEXES

Diagramme UML

