

DOCUMENTATION DE RONNDA JEU

Encadre par : Pr. Ikram

Nom: Kamate

Prenom: Boba fousseiny

Nom: Charraj

Prenom: Zakariaa

PLAN

UML	Conception
Part 1:	Description du Projet
Part 2:	Détermination des Fonctionnalités du Système
Part 3:	Les Acteurs
Part 4:	Le Diagramme de Cas d'Utilisation
Part 5:	Le Scénario
Part 6:	Identification des mots et expressions clés
Part 7:	Diagrammes de Classes

App	Ronda Desktop
Part 1:	La presentation des Pages
Part 2:	La transition Entre les pages
Part 3:	Le controlleur de page flexible
Part 4:	L'animation
Part 5:	L'installation sur Winndows

Annexe:	Les Problemes de Projet
---------	--------------------------------

Decription de projet

La **ronda** est un jeu de cartes qui se joue avec des cartes espagnoles ou cartes italiennes (quatre couleurs latines: coupes, épées, bâtons, deniers) en utilisant quarante cartes. On trouve des cartes numérotées de 1 à 7, et trois autres cartes : le Valet (10), le Cavalier (11) et le Roi (12)). On remarquera qu'il « manque » les 8 et 9.

Le jeu se joue à deux, trois ou quatre joueurs.

L'objectif est de ramasser le plus de cartes possible pour marquer le plus de points. La manche est gagnée par le premier joueur ou équipe qui atteint plus de 20 cartes. Quant à la partie, elle est gagnée lorsqu'une équipe totalise la somme de 41 points.

Détermination des Fonctionnalités du Système

Fonctionnalités du jeu Ronda :

1. Distribution des cartes :

- L'application doit être capable de distribuer 3 cartes à l'utilisateur et à l'IA au début de la partie.
- 4 cartes doivent être placées au milieu de la table.

2. Sélection des cartes :

- Les joueurs doivent pouvoir cliquer sur une carte de leur main.
- Les cartes sélectionnées doivent être comparées à la liste des cartes sur la table.

3. Collecte des cartes :

- Si la carte sélectionnée est valide (même numéro ou consécutif à une carte sur la table), la carte sélectionnée et toutes les cartes consécutives doivent être collectées.
- Les points doivent être attribués en fonction du nombre de cartes collectées (1 point par carte).

4. Fin de la partie :

- La partie doit se terminer lorsque tous les joueurs ont joué toutes leurs cartes ou lorsque toutes les cartes de la table ont été collectées.
- Les points de chaque joueur doivent être comptabilisés.
- Le joueur avec le plus grand nombre de points est déclaré gagnant.

Les Acteurs

1. Utilisateur :

- Joue le jeu en sélectionnant des cartes et en prenant des décisions stratégiques.
- Peut accéder aux règles du jeu et choisir la difficulté.

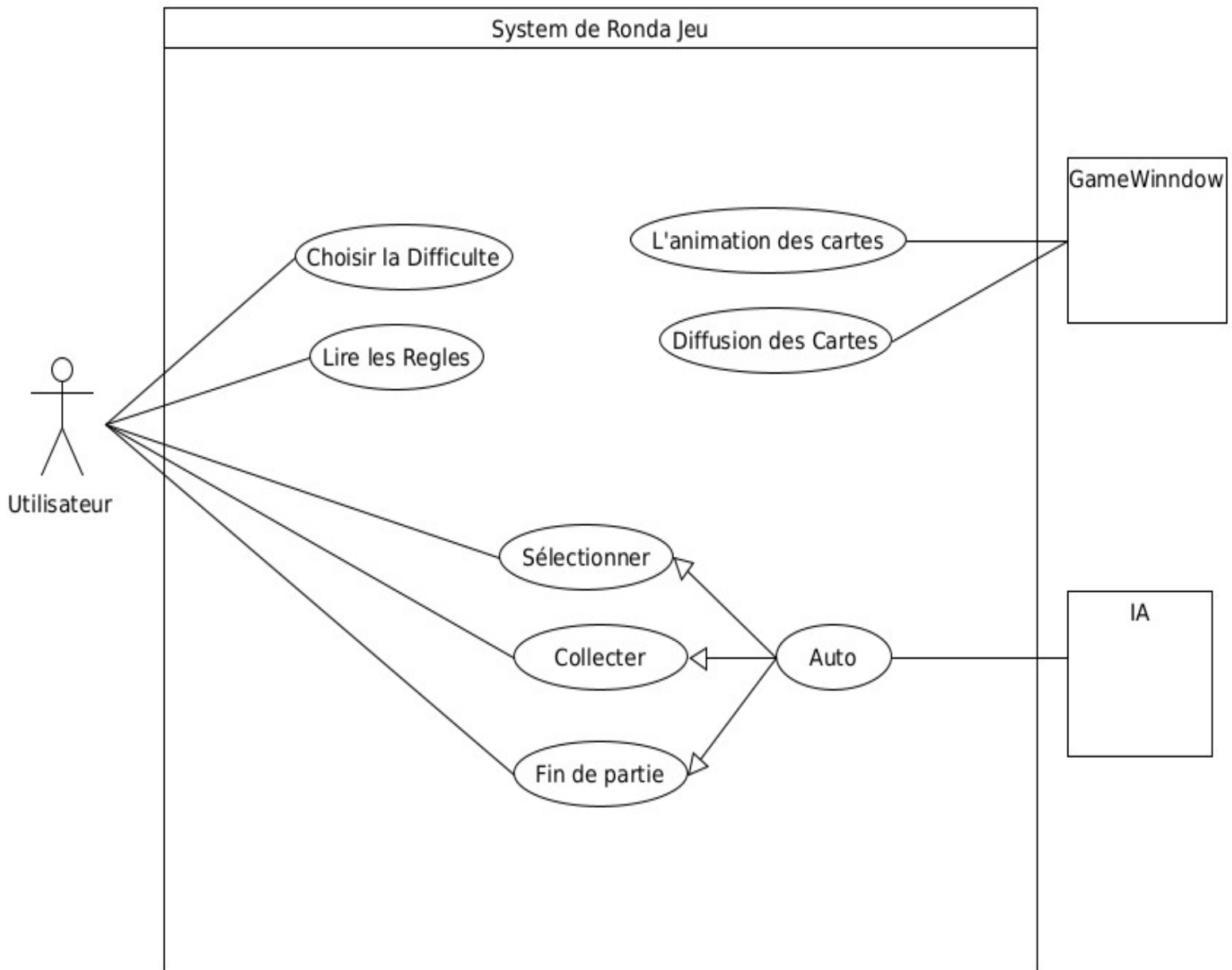
2. IA (Intelligence Artificielle) :

- Joue en suivant des règles prédéfinies, avec une stratégie adaptée à la difficulté choisie.

3. GameWindow (Classe) :

- Gère les animations des cartes.
- Gère la distribution des cartes.

Le Diagramme de Cas d'Utilisation



Le Scénario

Scénario mis à jour du Jeu Ronda :

1. Démarrage de la partie :

- Le joueur lance l'application du jeu Ronda.
- L'utilisateur peut accéder aux règles du jeu depuis le menu principal.
- L'utilisateur choisit la difficulté (facile, moyen, difficile).
- La classe `MainWindow` gère la distribution des 3 cartes à l'utilisateur et à l'IA avec des animations.
- La classe `MainWindow` gère la diffusion de 4 cartes au centre de la table avec des animations.

2. Tour de jeu - Utilisateur :

- L'utilisateur observe ses cartes et les cartes sur la table.
- L'utilisateur sélectionne une carte de sa main en cliquant dessus.
- Le système vérifie si la carte sélectionnée peut être collectée en fonction des règles du jeu.

3. Tour de jeu - IA :

- L'IA analyse ses propres cartes et les cartes sur la table.
- L'IA sélectionne automatiquement une carte en suivant une stratégie adaptée à la difficulté choisie.
- Le système vérifie si la carte sélectionnée par l'IA peut être collectée.

4. **Collecte des cartes :**

- Si la carte sélectionnée par l'utilisateur ou l'IA est valide, le système collecte cette carte ainsi que toutes les cartes consécutives avec des animations.
- Les points correspondants sont attribués au joueur (1 point par carte).

5. **Vérification de la fin de la partie :**

- Le système vérifie si toutes les cartes de l'utilisateur ou de l'IA ont été jouées.
- Ou si toutes les cartes sur la table ont été collectées.
- Si l'une des conditions est remplie, le jeu passe à l'étape suivante.

6. **Fin de partie :**

- Les points de l'utilisateur et de l'IA sont comptabilisés avec des animations.
- Le système déclare le joueur avec le plus grand nombre de points comme le gagnant avec des animations.
- L'application affiche un message de fin de partie.

7. **Option de rejouer :**

- L'utilisateur peut choisir de rejouer une nouvelle partie avec des animations ou aller à la première page.
- Si l'utilisateur accepte, la classe `MainWindow` gère la réinitialisation du jeu, y compris la distribution des cartes avec des animations.
- Sinon, le jeu se termine.

Identification des mots et expressions clés

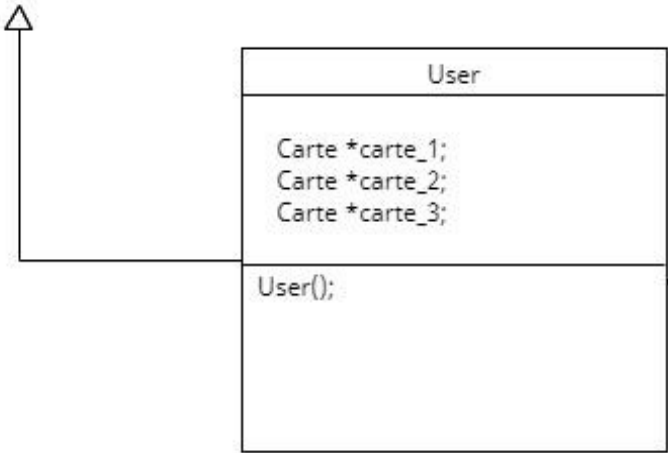
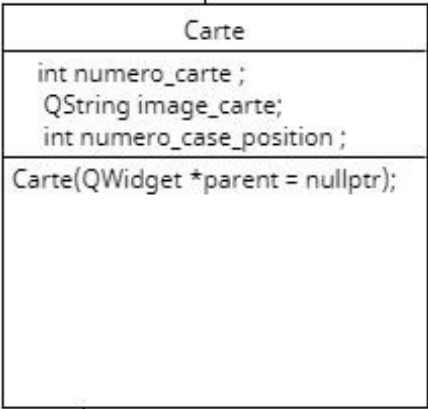
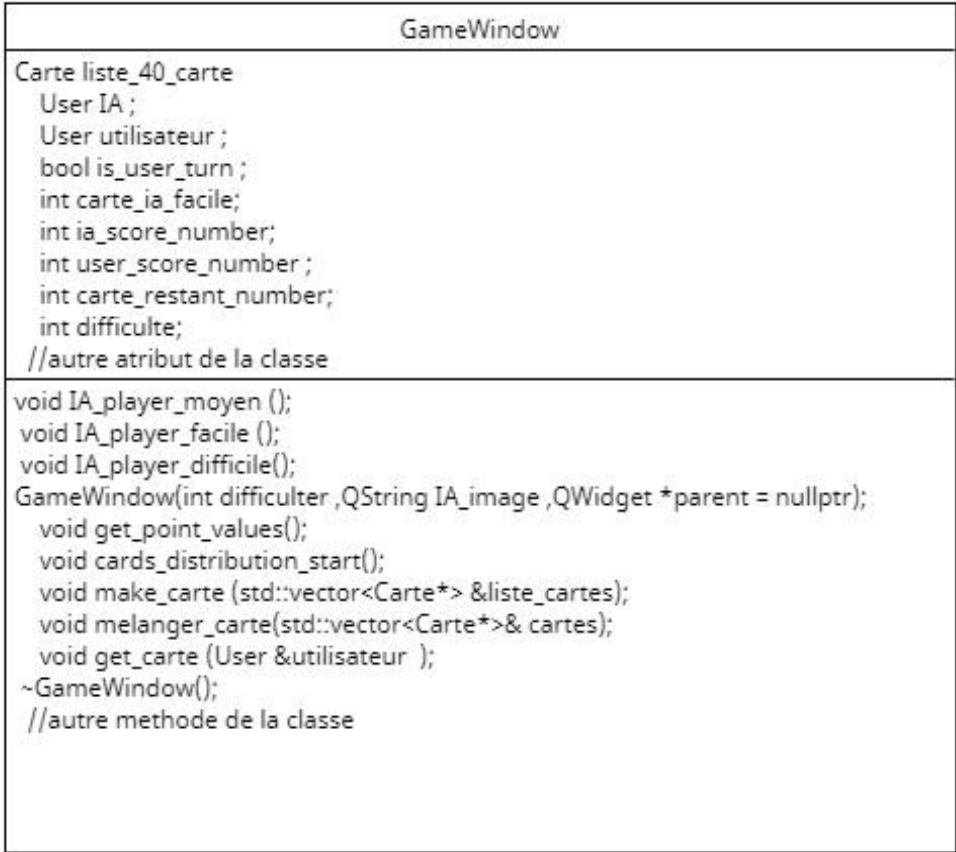
Mot/Expression	Objet	Attribut	Valeur	Methode	Poubelle
get_point_values				X	
cards_distribution_start				X	
make_carte				X	
melanger_carte				X	
get_carte				X	
connection_click_utilisateur				X	
controlleur_medium				X	
appartient_a_medium				X	
hide_carte				X	
make_user_card_animation				X	
get_position				X	
move_animation				X	
make_empty_animation				X	
distrube_user_card_animation				X	
update_user_click				X	
IA_player_moyen				X	
IA_player_facile				X	

desactive_medium_click				X	
clean_medium				X	
update_screen_info				X	
card_img_1_ia_position			X		
card_img_2_ia_position			X		
card_img_3_ia_position			X		
card_img_1_user_position			X		
card_img_2_user_position			X		
card_img_3_user_position			X		
medium_card_1_position			X		
medium_card_2_position			X		
medium_card_3_position			X		
medium_card_4_position			X		
medium_card_5_position			X		
medium_card_6_position			X		
medium_card_7_position			X		
medium_card_8_position			X		
medium_bank_position			X		
card_img_1_ia_widget	X				
card_img_2_ia_widget	X				
card_img_3_ia_widget	X				
card_img_1_user_widget	X				
card_img_2_user_widget	X				
card_img_3_user_widget	X				
medium_card_1_widget	X				
medium_card_2_widget	X				
medium_card_3_widget	X				

medium_card_4_widget	X				
medium_card_5_widget	X				
medium_card_6_widget	X				
medium_card_7_widget	X				
medium_card_8_widget	X				
medium_bank			X		
iste_40_cartes		X			
liste_8_cartes_medium		X			
medium_carte_controlleur		X			
ma_fenetre	?				
IA	X				
Utilisateur	X				
is_user_turn			X		
compteur_de_tour			X		
carte_ia_facile			X		
ia_score_number		X			
user_score_number		X			
carte_restant_number		X			
timer_tour			X		
ia_score	X				
user_score	X				
carte_restant			X		
ImagePathBack			?		
difficulte			?		

Diagrammes de Classes

Objet	Attribut	Methode
Utilisateur	<ul style="list-style-type: none">• Carte *carte_1;• Carte *carte_2;• Carte *carte_3;	<ul style="list-style-type: none">• User()
Carte	<ul style="list-style-type: none">• int numero_carte;• QString image_carte;• int numero_case_position;	<ul style="list-style-type: none">• Carte(QWidget*parent=nullptr);
GameWindow	<ul style="list-style-type: none">• Carte list_40_carte• User IA;• User utilisateur;• bool is_user_turn;• int carte_ia_facile;• int ia_score_number;• int user_score_number;• int carte_restant_number;• int difficile;	<ul style="list-style-type: none">• GameWindow(int difficulter ,QString IA_image ,QWidget *parent = nullptr);• void get_point_values();• void cards_distribution_start();• void make_carte (std::vector<Carte*> &liste_cartes);• void melanger_carte(std::vector<Carte*>& cartes);• void get_carte (User &utilisateur);• ~GameWindow();



Ronda Desktop

→ **Note:**

Cette section est déjà rédigée sous forme de commentaires dans le projet.

Les Problemes de Projet

OOP :

1. **Logique des Types d'Éléments en QT**
2. **Modification des Fonctions Abandonnées**
3. **Ouverture du Contenu d'une Classe sans Fermer la Page Actuelle**
4. **Restructuration des Classes et Création de la Classe Player**

Projet :

5. **Logique des Images en QT et Chemins Relatifs**
6. **Abandon de la Séparation en CPP et CMake**
7. **Difficultés à Trouver des Ressources pour le Projet**
8. **Problèmes avec les Versions de QT et Bibliothèques Manquantes**
9. **Création d'Animations avec les Widgets de QT**