**Documentation du Projet de Jeu de Cartes Marocaines en Qt**

**Realiser  par : El ouriachi Aya**

**Kharchich Soukaina**

**Présentation :**

‘LKBIR YAKOUL SRIR ‘: Projet de Jeu de Cartes Marocaines

LKBIR YAKOUL SRIR est un projet de jeu de cartes marocaines interactif offrant une expérience de jeu unique. Avec deux niveaux de difficulté, les joueurs peuvent choisir entre une partie facile et une partie difficile. Le jeu repose sur la comparaison des numéros des cartes entre le joueur réel et l'ordinateur, et il suit des règles de jeu clairement définies.

**Règles du Jeu :**

Début du Jeu :Chaque partie commence avec le joueur réel.

Jouer une Carte :Lorsque le joueur réel joue une carte, l'ordinateur répond en choisissant immédiatement une carte parmi celles qui restent au joueur. Le choix de la carte par l'ordinateur dépend du niveau de difficulté sélectionné par le joueur.

Cette mécanique de jeu vise à simuler un véritable défi, où le joueur doit prendre des décisions stratégiques pour optimiser ses chances de gagner.

Système de Scores :

-Deux scores sont maintenus tout au long de la partie : un pour l'ordinateur et un pour le joueur réel.

-Chaque joueur voit son score augmenter en fonction du nombre de cartes remportées au cours du jeu.

-Les joueurs doivent être attentifs aux cartes qu'ils jouent et à celles remportées par l'ordinateur, car cela affecte directement les scores.

Fin du Jeu :La partie prend fin dès qu'un des joueurs atteint un score total supérieur ou égal à 70.

Cette règle ajoute une dimension stratégique supplémentaire, car les joueurs doivent non seulement remporter des cartes mais aussi gérer leurs scores pour atteindre la victoire.

Remarque : Le projet LKBIR YAKOUL SRIR propose une expérience de jeu engageante, où le joueur se mesure à l'ordinateur dans une partie de cartes marocaines. Les règles simples, associées à la possibilité de choisir les niveau de difficulté, offrent une expérience adaptée à différents niveaux d'habileté. Jouez avec stratégie, surveillez les scores, et visez la victoire dans ce passionnant jeu de cartes marocaines.

**Annexe :**

La première difficulté à laquelle nous avons été confrontés était le choix du jeu, car il y a plusieurs options et nous étions quelque peu confus en prenant la décision finale. "Kbir Yakol Sghir" est l'un des jeux de cartes les moins connus, alors nous avons pensé que ce serait une excellente décision de le faire découvrir aux gens. La deuxième chose qui a rendu ce jeu difficile à programmer est le travail avec l'interface QT, car même s'il y a des ressources sur YouTube, cela n'a pas été suffisant. grâce à ChatGPT d'avoir facilité l'utilisation des bibliothèques les plus nécessaires. Cependant, la confusion entre l'utilisation de QML et QWidget a rendu les choses encore plus difficiles, mais à la fin, nous avons choisi QWidget car c'était plus facile à travailler, nous permettant de suivre le processus de création du jeu.

Dans la partir codage on avait des difficultés en ce qui concerne Comment créer les cartes de l'ordinateur en les cachant, puis, lorsque nous avons besoin de les afficher, elles montreront les images réelles de ces cartes. Chaque fois qu'une carte est mise sur la table, la carte précédente sera retirée. Cela pose une difficulté temporelle dans l’insertion du son du jeu.

**Fonctionnalités :**

**mousePressEvent(QMouseEvent \*event) :** Cette fonction semble faire partie d'une classe appelée Cartes et est un gestionnaire d'événements associé à l'événement de clic de la souris (QMouseEvent).

**setClickable(bool clickable) :** Cette fonction, nommée setClickable, semble être une méthode d'une classe appelée Cartes. Elle est utilisée pour activer ou désactiver la capacité du widget associé à cette classe à recevoir des clics de souris.

**getImage(const QString& img) :** En résumé, cette fonction prend un chemin d'image, charge l'image, la redimensionne, configure l'objet Cartes actuel pour afficher l'image redimensionnée, et émet un signal pour informer d'un éventuel changement d'image

**creer\_carte(int numero, const QString &symbol, const QPixmap &image):** En résumé, cette fonction crée et configure une nouvelle carte, affiche son image dans un QLabel, connecte le signal de clic de la carte à une fonction spécifique (onCarteClicked), et retourne la nouvelle carte créée. La connexion du signal est conditionnelle sur la présence de la carte dans un certain layout horizontal.

**onCarteClicked () :** ce code gère les interactions liées à la carte cliquée, la suppression des cartes de horizontalLayout et leur ajout à horizontalLayout\_4. En outre, il prend en compte la comparaison des numéros des cartes et ajuste les scores en conséquence.

**on\_pushButton\_clicked() :** semble gérer l'initialisation et la réinitialisation du jeu, en embarquant les cartes, les mélangeant, et les distribuant aux joueurs

**effacerCartesLayout(QHBoxLayout \*layout) :** une méthode efficace pour supprimer tous les widgets d'un QHBoxLayout. Elle utilise une boucle while pour retirer et supprimer chaque widget jusqu'à ce que le layout soit vide.

**Screenshot :**

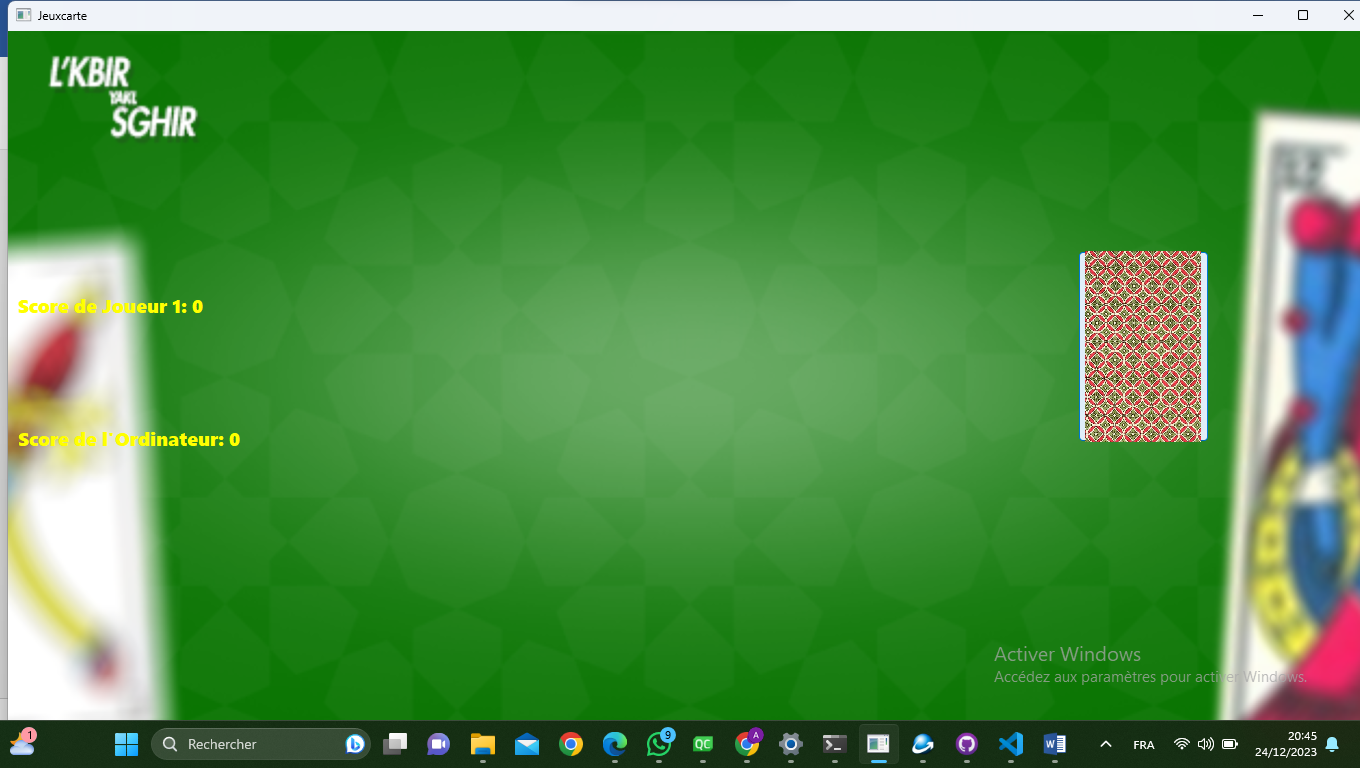
**Page1 : L’ouverture de jeu**

****

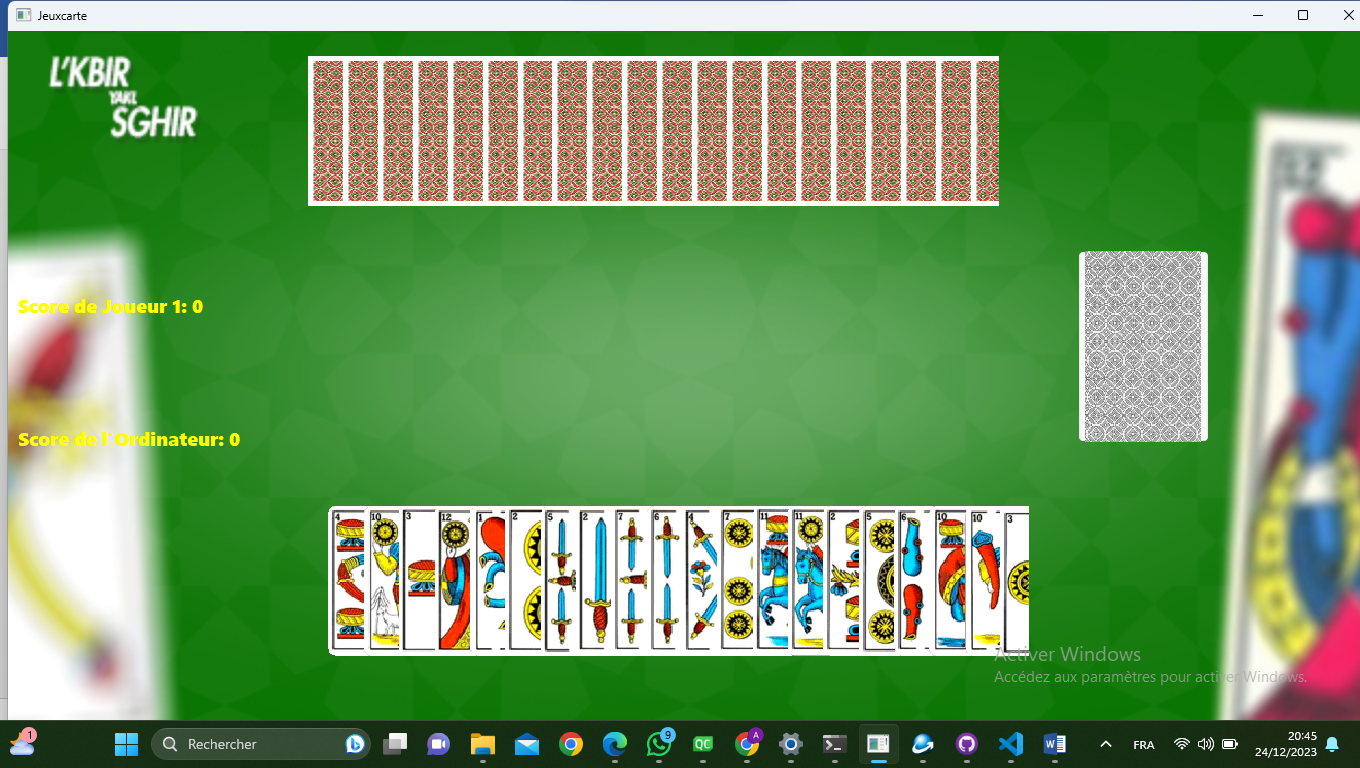
**Page2 :Le choix de difficulté**

****

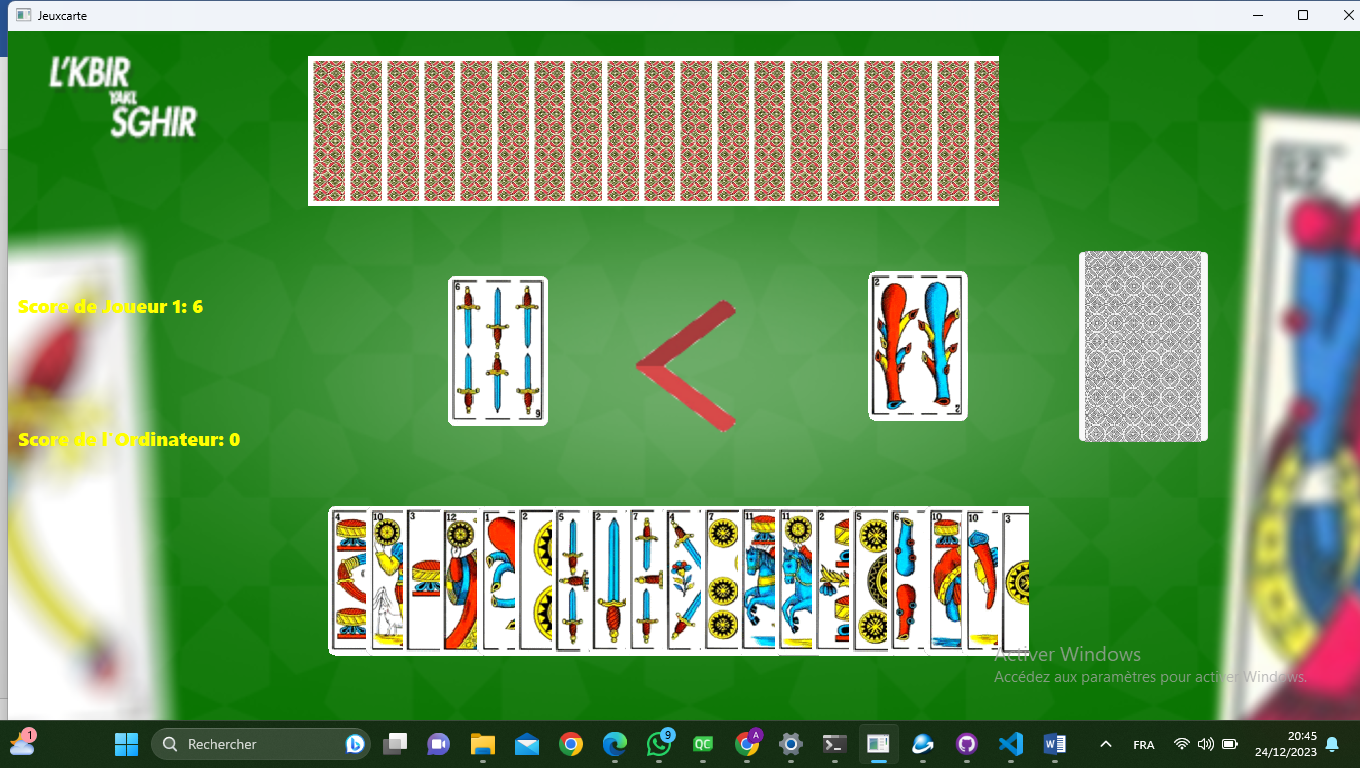
**Page3 :Le bouton avec lequel je peux distribuer les cartes**

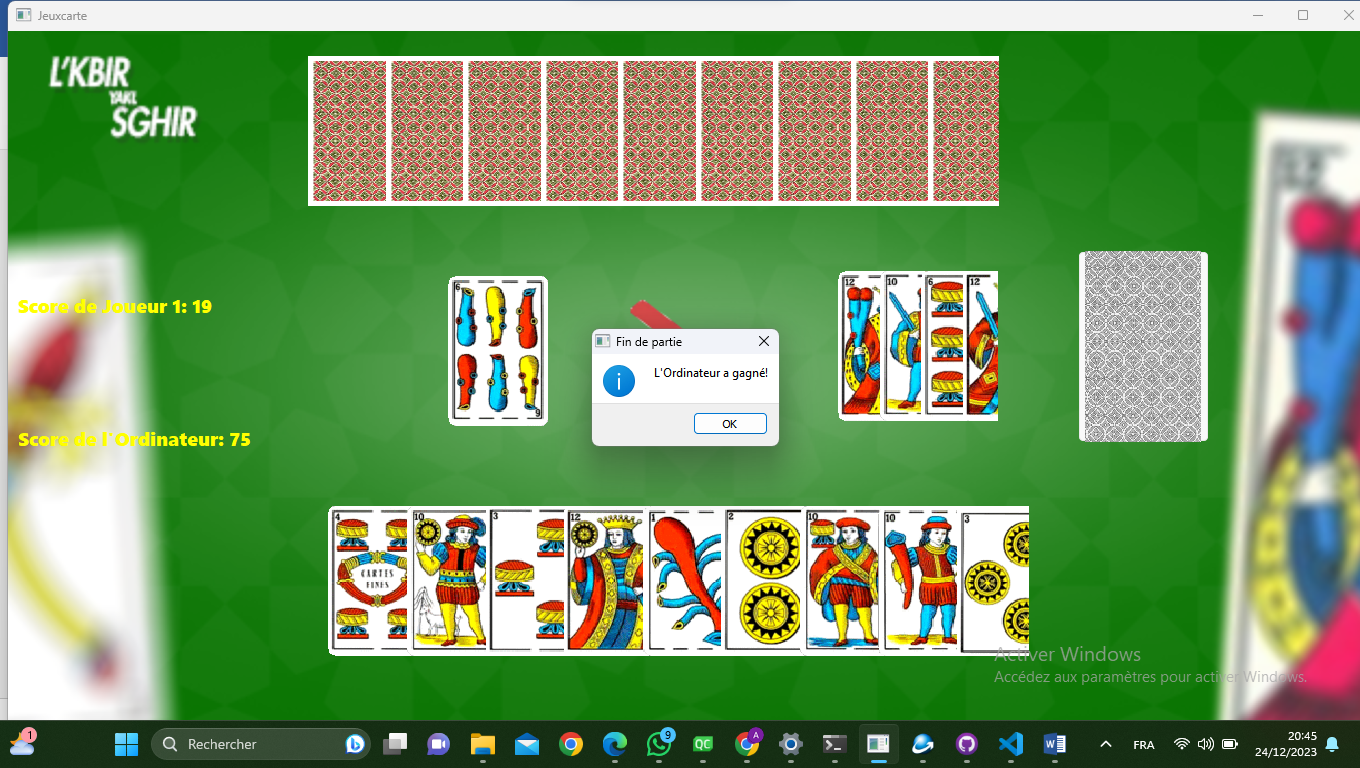
****

**Page4 : La distribution des cartes**

****

**Page5 : Le debut de jeu**

****

**Page 6 : Fin de jeu**