Louis-Félix Veillette

Olivier Savoie

Dictionnaire des classes

Travail présenté à

Charles Jacob

14 avril 2022

Cegep régional de Lanaudière

Table des matières

[Vues : 3](#_Toc99098179)

[Contrôleur : 5](#_Toc99098180)

[Modèles 6](#_Toc99098181)

# Vues :

|  |  |  |  |
| --- | --- | --- | --- |
| FormMenu  Description : Formulaire du menu permettant de commencer une nouvelle partie, de voir le classement des joueurs et de quitter le jeu. | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_controlleur | Echec | Variable permettant de remonter au controller pour le mettre à jour |
| *Méthodes* | | | |
| Public | btnNouvelle\_Click() | void | Événement du bouton « Nouvelle partie » qui actionne une nouvelle partie |
| Public | btnClassement\_Click() | Void | Événement du bouton  « Classement » qui affiche le classement |
| Public | FormMenu\_FormClosed() | Void | Événement lors de la fermeture de ce formulaire (Menu principal) qui ferme tout le jeu |
| Public | btnQuitter\_Click() | Void | Événement du clic du bouton « Quitter le jeu » |

|  |  |  |  |
| --- | --- | --- | --- |
| FormSelection  Description : Formulaire de la sélection des joueurs. | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_controlleur | Echec | Variable permettant de remonter au controller pour le mettre à jour |
| *Méthodes* | | | |
| Public | btnCommencerClick() | void | Événement du clic du bouton « Commencer » qui débute la nouvelle partie |
| Public | FormSelection\_Load() | Void | Affiche le formulaire de sélection avec la liste des joueurs dans chaque ComboBox pour sélectionner les joueurs |
| Public | cmbJoueur1\_SelectedIndexChanged() | Void | Événement de la sélection du premier joueur pour jouer, qui sélectionne le premier joueur et l'enlève de la deuxième liste |
| Public | cmbJoueur2\_SelectedIndexChanged | Void | Événement de la sélection du deuxième joueur pour jouer, qui sélectionne le deuxième joueur et l'enlève de la première liste |

|  |  |  |  |
| --- | --- | --- | --- |
| FormClassement  Description : Formulaire du classement des joueurs. | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_controlleur | Echec | Variable permettant de remonter au controlleur. |
| Privé | \_joueurs | List<Joueur> | Liste des joueurs créés |
| *Méthodes* | | | |
| Public | FormClassement\_Load() | void | Événement de l'affichage du formulaire du classement des joueurs |
| Public | btnAjouter\_Click () | Void | Ajoute le nouveau joueur au classement |
| Public | btnSupprimer\_Click() | Void | Supprime le joueur sélectionné du classement |

|  |  |  |  |
| --- | --- | --- | --- |
| FormPartie  Description : Formulaire de la partie où se trouve l’échiquier et le choix des joueurs pour la partie. | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_controlleur | Echec | Variable permettant de remonter au controlleur. |
| Privé | \_indexPartie | int | Identifiant de la partie créée |
| Privé | clickInitial | Point | Coordonnées de la case initiale |
| Privé | clickDest | Point | Coordonnées de la case destination |
| *Méthodes* | | | |
| Private | btnAbandon\_Click() | void | Événement du clique du bouton « abandonner » pour terminer la partie |
| Private | btnNulle\_Click() | Void | Événement du clique du bouton « Nulle » pour terminer la partie |
| Private | pnlEchiquier\_Paint() | Void | Met à jour l’échiquier et positionne les images des pièces correctement sur l’échiquer ainsi que les messages sur les tours des joueurs. |
| Private | pnlEchiquier\_Click() | Void | Événement des cliques sur l'échiquier. Joue le coup et affiche les messages si possibles. |
| Public | peinturerEchiquier() | Void | Affiche les images des pièces sur l'échiquier sur leurs bons emplacements. |
| Public | finPartie() | void | Termine la partie dépendamment de la raison de cette dernière. |

|  |  |  |  |
| --- | --- | --- | --- |
| FormPromotion  Description : Formulaire qui s’affiche lors d’une promotion d’un pion. Affiche un choix de pièce qui pourra remplacer le pion | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_controlleur | Echec | Variable permettant de remonter au contrôleur. |
| Privé | \_indexPion | Int | Emplacement (index) où se trouve le pion qui s’apprête à être promu |
| *Méthodes* | | | |
| Private | pnl1\_Paint() | void | Dessine l'image de la reine dans le formulaire |
| Private | pnl2\_Paint() | Void | Dessine l'image du fou dans le formulaire |
| Private | pnl3\_Paint() | Void | Dessine l’image du cavalier dans le formulaire |
| Private | pnl4\_Paint() | Void | Dessine l’image de la tour dans le formulaire |
| Private | pnl1\_Click() | void | Événement du clic de l’image sur la reine, ce qui promouvoit le pion en reine |
| Private | pnl2\_Click() | Void | Événement du clic de l’image du fou, ce qui promouvoit le pion en fou |
| Private | pnl3\_Click() | Void | Événement du clic de l’image du cavalier, ce qui promouvoit le pion en cavalier |
| Private | pnl4\_Click() | Void | Événement du clic de l’image de la tour, ce qui promouvoit le pion en tour |

# Contrôleur :

|  |  |  |  |
| --- | --- | --- | --- |
| Echec  Description : Le jeu d’échec permettant de jouer plusieurs parties et de montrer les joueurs. | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_formMenu | FormMenu | Menu principal du jeu |
| Privé | \_formPartie | FormPartie | Formulaire d’une partie |
| Privé | \_formSelection | FormSelection | Formulaire de sélections des deux joueurs qui joueront la partie |
| Privé | \_formPromotion | FormPromotion | Formulaire de promotion des pions |
| Privé | \_unePartie | Partie | L’une des parties d’échec étant commencé |
| Privé | \_listeJoueur | List<Joueur> | Liste des joueurs du jeu |
| Privé | \_listePartie | List<Partie> | Liste des parties étant commencées |
| Privé | \_listeFormPartie | List<FormPartie> | Liste des formulaires des parties étant commencées |
| *Méthodes* | | | |
| Public | nouvellePartie() | Void | Ouvre l formulaire de sélection de joueurs pour la nouvelle partie |
| Public | commencerPartie(string joueur1, string joueur2) | Void | Commence la nouvelle partie et active et garde en mémoire les joueurs qui vont s'affronter |
| Public | jouerCoup (FormPartie monForm, int x1, int y1, int x2, int y2) | Tuple<bool, int> | Vérifie si le coup est valide et fait si possible le déplacement ainsi que la promotion s'il y a lieu. Retourne vrai et un entier représentant que le coup est valide si le coup est valide |
| Public | changerPion(char piece, int indexPion) | Void | Change le pion qui est promu par la pièce que le joueur aura sélectionnée. |
| Public | Victoire\_Abandon(FormPartie monForm) | Void | Abandon de la partie |
| Public | uneNulle(FormPartie monForm) | Void | Partie nulle |
| Public | tour(FormPartie monForm) | Int | Indique le tour du joueur dans la bonne partie. Retourne le tour du joueur. |
| Public | tour(FormPromotion monForm) | Int | Indique le tour du joueur dans la bonne partie. Retourne le tour du bon joueur |
| Public | ajouterJoueur(string nom, int gagné, int perdu, int nulle) | void | Ajoute un nouveau joueur dans la liste des joueurs du classement avec ses statistiques. |
| Public | nettoyerJoueur() | Void | Vide la liste des joueurs |
| Public | ListeJoueur | List<Joueur> | Retourne la liste de joueurs du jeu |
| Public | afficherEchiquier() | String | Affiche l'échiquier avec la position de chaque pièce. Retourne la chaine de caractères représentant l’échiquier. |
| Public | fermerJeu() | void | Ferme le jeu d'échec et sauvegarde |
| Public | quitter() | void | Ferme le jeu d'échec |
| Public | lire() | Void | Ajoute tous les joueurs du fichier sauvegardé dans la liste des joueurs |
| Public | Sauvegarder() | void | Sauvegarde tous les joueurs dans un fichier |
| Public | enleverJoueur(int index) | Void | Enlève le joueur sélectionné de la liste des joueurs |

# Modèles

|  |  |  |  |
| --- | --- | --- | --- |
| Partie  Description : Partie d’échec ayant les informations de la partie. | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_controlleur | Echec | Contrôleur Echec |
| Privé | \_plateau | Plateau | Plateau du jeu avec toutes les 64 cases. |
| Privé | \_nbCoup | int | Compte du nombre de coups joués. |
| Privé | \_coupDepuisPionJoué | Int | Compte le nb de coups depuis le dernier mouvement d’un pion pour prévenir un null |
| Privé | \_joueur1 | Joueur | Le joueur 1 |
| Privé | \_joueur2 | Joueur | Le joueur 2 |
| Privé | \_listeEchiquier | List<string> | Liste permettant de mémoriser les dernières positions des pièces. |
| *Méthodes* | | | |
| Public | victoire\_Abandon() | void | Met à jour les statistiques des joueurs après l'abandon |
| Public | uneNulle() | void | Met à jour les statistiques des joueurs après la partie nulle |
| Public | tour() | int | Retourne le tour du bon joueur |
| Public | determinerCase(int x1, int y1, int, x2, int y2) | Tuple<int, int> | Détermine l'index de la case initiale et de la case de destination du coup. Retourne les index de la case initiale et destination. |
| Public | verifDeplacement (int indexInitial, int indexDesti) | Tuple<bool, int> | Vérifie la validité du déplacement du joueur. Retourne vrai sans entier d'erreur si le coup est valide |
| Public | faireDeplacement(int indexInitial, int indexDesti) | bool | Fais le coup demandé. Retourne vrai si une promotion de pion doit avoir lieu |
| Public | changerPion(char piece, int indexPion) | void | Promouvoit le pion en la pièce que le joueur a sélectionné. |
| Public | verifEchec() | Tuple<bool, int> | Vérifie si un échec ou un échec et mat a lieu. Retourne vrai avec 0 s'il n'y a pas d'échec ni d'échec et mat |
| Public | afficher() | string | Affiche la chaine de caractère de l'échiquier avec les caractères des pièces aux bons endroits. Retourne la chaine de caractères de l'échiquier. |

|  |  |  |  |
| --- | --- | --- | --- |
| Plateau  Description : Plateau (échiquier) de la partie | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_echiquier | Case[] | Échiquier avec ses 64 cases |
| Privé | \_partie | Partie | La partie où se trouve le plateau |
| Privé | \_plateau | String | La chaine de caractères du plateau avec les caractères des pièces dans leurs bonnes positions. |
| *Méthodes* | | | |
| Public | Echiquier | Case[] | Retourne l'échiquier |
| Public | deplacer(int indexInitial, int indexDesti) | Void | Exécute le coup du joueur en mettant la case initiale sur la case destination, puis rend la case initiale à vide. |
| Public | afficher() | String | Affiche la chaine de caractère de l'échiquier. Retourne la chaine de caractères de l'échiquier avec les bonnes positions de pièces. |
| Public | verifierSiPiece(int indexInitial) | Tuple<bool, int> | Vérifie si le joueur a cliqué sur une case non vide. Retourne vrai s'il y a une pièce à la case initiale |
| Public | maPiece(int indexInitial) | Tuple<bool, int> | Vérifie si la case initiale a bel et bien un des pièces du joueur. Retourne vrai si la pièce sélectionné est une pièce du joueur devant jouer. |
| Public | verifTrajectoire(int indexInitial, int indexDestination) | Mouvement | Vérifie la trajectoire de la pièce par rapport aux règles de cette dernière. Retourne le type de mouvement de la pièce |
| Public | verifRoque(int indexInitial, int indexDestination) | Mouvement | Vérifie si le joueur a enclenché un roque et vérifie s'il peut le faire. Retourne le mouvement du roque s'il le fait |
| Public | deplacement(int indexInitial, int indexDesti) | Int | Vérifie le déplacement que la pièce fera sur chaque case. Retourne le nombre de bons de case pour faire chaque mouvement de case en case |
| Public | estCollision(int indexInitial, int indexDestination, int deplacement, Case[] echiquier) | Tuple<bool, int> | Vérifie s'il y a une collision durant le trajet de la pièce. Retourne vrai avec 0 s'il n'y a pas de collision dans le trajet. |
| Public | estCollision(int indexInitial, int indexDestination, int deplacement) | Tuple<bool, int> | Vérifie s'il y a une collision durant le trajet de la pièce. Retourne vrai avec 0 s'il n'y a pas de collision dans le trajet. |
| Public | verifCouleurDesti(int indexInitial, int indexDestination) | Tuple<bool, int> | Vérifie la couleur de la pièce de la case destination pour s'assurer que ce n'est pas une pièce alliée. Retourne vrai avec zéro si la couleur de la pièce qui se trouve sur la destination est alliée. |
| Public | metEnEchecAllié(int indexInitial, int indexDesti) | Tuple<bool, int> | Vérifie si le coup met en échec le roi allié. Retourne vrai avec le message 0 pour confirmer que le coup ne met pas en échec le roi allié. |
| Public | verifPromoPion(int indexDestination) | bool | Retourne vrai lorsqu’un pion joue s’il peut faire une promotion. Retourne vrai si le pion peut être promu. |
| Public | nePeutPlusCharger(int index) | Void | Enlève le droit à la pièce de pouvoir charger. |
| Public | verifPromoPion(int indexDestination) | Bool | Vérifie si une promotion de pion doit avoir lieu. Retourne vrai si une promotion doit avoir lieu. |
| Public | changerPion(char piece, int indexPion) | Void | Change le pion en la nouvelle pièce qu'il deviendra. |
| Public | verifEchec(int indexRoi, Couleur couleur) | Tuple<bool, int> | Vérifie si le coup met en échec le roi ennemi. Retourne vrai, 0 ainsi que l'index de la pièce qui met en échec l'ennemi |
| Public | trouverRoiEnnemi() | Tuple<int, Couleur> | Trouve le roi ennemi. Retourne l'index du roi et sa couleur. |
| Public | verifEchecMat(int indexRoi, int indexAttaquant) | Tuple<bool, int> | Vérifie s'il y a un échec et mat. Retourne vrai et 9 s'il y a un échec et mat. |
| Public | peutBougerRoi(int indexRoi) | Tuple<bool, int> | Vérifie si le roi peut bouger. Retourne vrai s'il peut bouger. |

|  |  |  |  |
| --- | --- | --- | --- |
| Joueur  Description : Les joueurs étant inscrits dans le classement du jeu d’échec avec leurs statistiques. | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_nom | String | Nom du joueur. |
| Privé | \_statsVictoire | Int | Les statistiques de victoire du joueur. |
| Privé | \_statsDefaite | Int | Les statistiques de défaite du joueur. |
| Privé | \_statsNull | Int | Les statistiques de partie nulle du joueur. |
| *Méthodes* | | | |
| Public | Nom | String | Retourne la donnée membre du nom du joueur |
| Public | StatsVictoire | Int | Retourne la donnée membre des statistiques de victoire du joueur. |
| Public | StatsDefaite | Int | Retourne la donnée membre des statistiques de défaite du joueur. |
| Public | StatsNull | Int | Retourne la donnée membre des statistiques de nulle du joueur |
| Public | ajoutVictoire() | Void | Ajoute un victoire au joueur. |
| Public | ajoutDefaite() | Void | Ajoute une défaite au joueur. |
| Public | ajoutNulle() | Void | Ajoute une nulle au joueur |
| Public | ToString() | String | Met en format lisible le joueur avec ses statistiques. Retourne la chaine de caractères des informations du joueur. |

|  |  |  |  |
| --- | --- | --- | --- |
| Case  Description : Case du plateau du jeu pouvant détenir une pièce | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_piece | Piece | Pièce se trouvant sur cette case. |
| Privé | \_estVide | Bool | Esy vrai si la case n’a aucune pièce. |
| *Méthodes* | | | |
| Public | Piece | Piece | Retourne la donnée membre pièce sur la case |
| Public | EstVide | Bool | Retourne vrai si la case est vide |
| Public | couleurPiece() | Couleur | La couleur de la pièce sur la case. Retourne la couleur blanc ou noir. |
| Public | Regles(int indexInitiale, int indexDesti) | Mouvement | Vérifie si les règles de la pièce sont respectés durant le coup. Retourne le type de mouvement de la pièce selon ses règles. |
| Public | ToString() | String | Renvoie le charactère représentant la pièce sur la case. Retourne Le caractère de la pièce. |
| Public | peutEtrePromu() | Bool | Vérifie si la pièce peut être promu. Retourne vrai si la pièce peut être promu. |
| Public | peutRoquer() | Bool | Vérifie si la pièce peut roquer. Retourne vrai si la pièce sur la case peut roquer. |
| Public | nePeutPlusCharger() | Void | Fais en sorte que la pièce ne puisse plus charger. |
| Public | vientDeBouger() | void | Dit à la pièce qu'elle a bougé. |

|  |  |  |  |
| --- | --- | --- | --- |
| Pieces  Description : Pièce du jeu d’échec | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Privé | \_nom | string | Nom de la pièce |
| Privé | \_couleur | Couleur | Couleur de la pièce |
| *Méthodes* | | | |
| Public | Couleur | Couleur | Retourne la couleur de la pièce |
| Public | peutEtrePromu() | Bool | Vérifie si la pièce peut être promu. Retourne faux si la pièce ne peut pas être promu. |
| Public | peutCharger() | Bool | Vérifie si la pièce peut charger. Retourne faux si la pièce ne peut pas charger |
| Public | NePeutPlusCharger() | Void | Fais en sorte que la pièce ne puisse plus charger. |
| Public | peutRoquer() | Bool | Vérifie si la pièce peut roquer. Retourne faux si la pièce ne peut plus roquer. |
| Public | vientDeBouger() | Void | Fais en sorte que la pièce ne puisse plus roquer |
| Public | regles(int indexInitiale, int indexDestination) | Mouvement | Vérifie le mouvement selon les règles de la pièce. Retourne le type de mouvement de la pièce. |
| Public | ToString() | String | Renvoie en chaîne de caractères le caractère qui représente le nom de la pièce. |

|  |  |  |  |
| --- | --- | --- | --- |
| PieceMemoire  Description : Pièce se souvenant qu'elle a bougé | | | |
| *Variables* | | | |
| Permissions | **Nom** | **Type** | **Description** |
| Protégé | **\_aBougé** | **Bool** | **Est true si la pièce a déjà bougé** |
| *Méthodes* | | | |
| Public | ABougé | bool | Gère la propriété qui montre si la pièce a déjà bougé |

|  |  |  |  |
| --- | --- | --- | --- |
| Tour  Description : La pièce Tour | | | |
| Permissions | **Nom** | **Type** | **Description** |
| *Méthodes* | | | |
| Public | peutRoquer() | Bool | Vérifie si la tour peut roquer |
| Public | vientDeBouger() | Void | Dit que la tour a bougé |
| Public | regles(int indexInitiale, int indexDestination) | Mouvement | Vérifie la validité du mouvement selon les règles de la tour. Retourne le type de mouvement de la tour |

|  |  |  |  |
| --- | --- | --- | --- |
| Reine  Description : La pièce reine | | | |
| *Méthodes* | | | |
| Public | regles(int indexInitiale, int indexDestination) | Mouvement | Vérifie la validité du mouvement selon les règles de la reine. Retourne le type de mouvement de la reine. |

|  |  |  |  |
| --- | --- | --- | --- |
| Fou  Description : La pièce du fou | | | |
| *Méthodes* | | | |
| Public | regles(int indexInitiale, int indexDestination) | Mouvement | Vérifie la validité du mouvement selon les règles du fou. Retourne le type de mouvement du fou. |

|  |  |  |  |
| --- | --- | --- | --- |
| Cavalier  Description : La pièce du cavalier | | | |
| *Méthodes* | | | |
| Public | regles(int indexInitiale, int indexDestination) | Mouvement | Vérifie la validité du mouvement selon les règles du cavalier. Retourne le type de mouvement du cavalier. |

|  |  |  |  |
| --- | --- | --- | --- |
| Pion  Description : La pièce du pion. | | | |
| Permissions | **Nom** | **Type** | **Description** |
| *Méthodes* | | | |
| Public | regles(int indexInitiale, int indexDestination) | Mouvement | Vérifie la validité du mouvement selon les règles du pion. Retourne le type de mouvement de du pion. |
| Public | peutEtrePromu() | Bool | Vérifie si le pion peut être promu. Retourne vrai s'il peut être promu. |
| Public | peutCharger() | Bool | Vérifie si le pion peut charger. Retourne vrai s'il n'a toujours pas bougé pour le permettre de charger. |
| Public | nePeutPlusCharger() | Void | Fais en sorte que le pion ne puisse plus charger. |

|  |  |  |  |
| --- | --- | --- | --- |
| Roi  Description : La pièce du roi | | | |
| Permissions | **Nom** | **Type** | **Description** |
| *Méthodes* | | | |
| Public | regles(int indexInitiale, int indexDestination) | Mouvement | Vérifie la validité du mouvement selon les règles du roi. Retourne le type de mouvement du roi |
| Public | peutRoquer() | Bool | Vérifie si le roi peut roquer |
| Public | vientDeBouger() | Void | Dit que le roi a bougé |