

Démineur

Dans le cadre du cours :

Programmation 3

Travail effectué par :

Marc-André Larocque Blais
Jean-François Duchesne

Travail remis à :

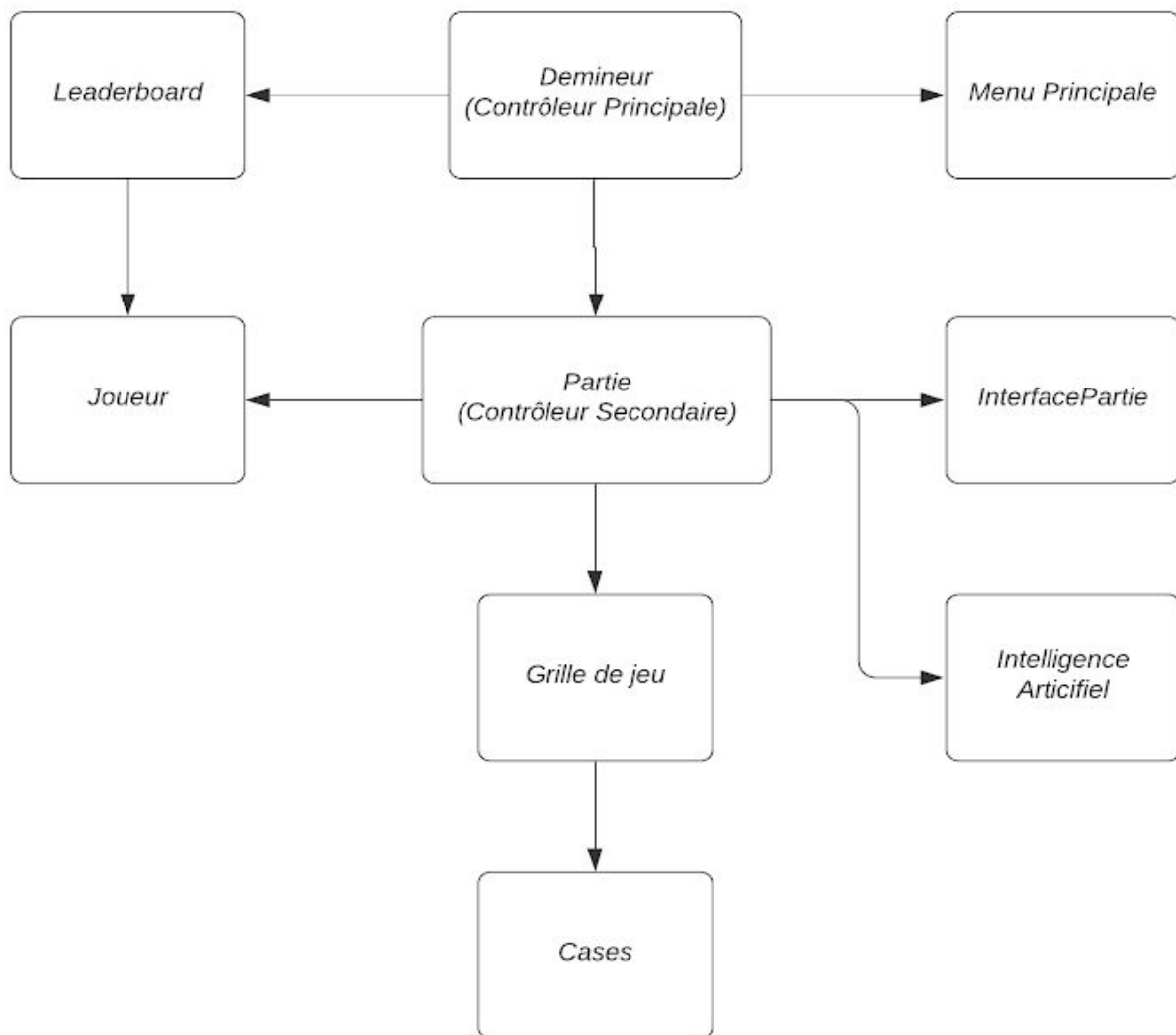
Charles Jacob

Cégep régional de Lanaudière à Joliette
Hiver 2020

Table des matières

Démineur	1
Diagramme des tables	3
Documentation des classes	4
Démineur	4
Menu Principal	4
Leaderboard	5
Partie	5
Interface	5
AI	6
Joueur	6
Grille	7
Case	7

Diagramme des tables



Documentation des classes

Démineur

Nom :	Démineur
Desc :	Point d'entrée du programme. Elle charge son MENU (statique) au démarrage afin de guider le joueur parmi les options. Elle s'occupe d'appeler le Leaderboard afin de charger les informations des joueurs. De là, on peut afficher les scores avec la classe Leaderboard , créer ou continuer une partie avec un nouveau joueur ou un joueur existant.
Donnée membre :	m_Partie : Partie m_EnMarche : Bool [?] m_Leaderboard : Leaderboard
Méthode +:	Aucune

Menu Principal

Nom :	Menu
Desc :	Affiche les options au joueur.(Joueur, Leaderboard, Quitter) Retournera un tableau contenant les paramètres de construction de la partie.
Donnée membre :	Aucune
Méthode +:	AfficherMenu(): char c OptionsDePartie(): char[3] {Grandeur, Difficulté, AI}

- Possibilité de changer le tableau pour du bitWise

Leaderboard

Nom :	Leaderboard
Desc :	Affiche les meilleurs temps (Top 10) par difficulté.
Donnée membre :	m_Joueurs : List
Méthode +:	AfficherLeaderboard(); MettreAJour(NomJoueur, temps, diff);

Partie

Nom :	Partie
Desc :	Instancie une grille de format reçu par le contrôleur démineur et un joueur. Elle agit comme contrôleur de partie en appelant les différentes méthodes nécessaire au bon déroulement de cette dernière.
Donnée membre :	Grille m_Grille Joueur m_Joueur AI m_AI String tempsEcoule
Méthode +:	obtenirMetadonneesDeLaPartieActuellementTerminee(): String partiInfo

Interface

Nom :	Interface
Desc :	Affiche la grille, le chronomètre, le nom du joueur, mode de jeu, nombre de bombes, le bouton AI si l'option AI a été activé lors de la création de la partie.
Donnée membre :	Aucune
Méthode +:	DessinerGrille(string grille); AfficherChronomètre();

AI

Nom :	AI
Desc :	Le AI peut être instancié ou non lors de la création de la partie. Une fois créé, il peut être appelé manuellement afin de jouer un coup pour aider le joueur ou si la partie créée est en mode automatique, il résoudra la grille au complet.
Donnée membre :	Aucune
Méthode +:	jouerUnCoup() : short[2]

Joueur

Nom :	Joueur
Desc :	Instancie un nouveau joueur et s'occupe de la reconstruction des joueurs existants à partir d'un fichier texte. Les scores enregistré sans AI seulement.
Donnée membre :	m_Nom : string m_Scores : string[9] { {temps p_Facile, temps p_Moyen, temps p_Difficile} {temps m_Facile, temps m_Moyen, temps m_Difficile} {temps g_Facile, temps g_Moyen, temps g_Difficile} }
Méthode +:	obtenirNom() : string m_Nom obtenirScore(string m_Nom) : String[] m_Score modifierScore(char grosseur, char difficulte, string temps) : bool modifier

Grille

Nom :	Grille
Desc :	Instancie un tableau de case et disperse les bombes.
Donnée membre :	Case[] champ short nbBombeGrille
Méthode +:	genererGrille(uint grosseur) : Void dispenserBombes(short nbBombeGrille) : Void

Case

Nom :	Case
Desc :	Initialise la valeur d'une case soit bombe ou non et combien de bombe l'entour
Donnée membre :	bool bombe char nbDanger bool[8] casesVoisines
Méthode +:	esTuBombe() : Bool nbDanger() : Char