

Manuel de développement et de maintenance

Développement d'un logiciel de contrôle de l'ordinateur par analyse des mouvements des yeux avec le dispositif Tobii

Table des matières

Table des matières	2
Présentation du logiciel installé	3
L'exécutable du jeu	3
Les fichiers ".dll"	3
Le dossier "log"	3
Le dossier "ressources"	3
Présentation du dossier du projet	4
Les différentes classes	4
La classe Application.	4
La classe Window	5
La classe CursorMovementManager et les classes Cursor, MouseCursor et TobiiCursor	5
La classe InputManager	5
La classe Config	5
La classe StatisticsManager	5
La classe MusicManager	6

Présentation du logiciel installé

La structure du logiciel se découpe en plusieurs dossiers et fichiers.

- L'exécutable du jeu
- Le dossier "ressources"
- Le dossier "log"
- Plusieurs fichier ".dll"

L'exécutable du jeu

Ce fichier est le fichier contenant le code de l'application.

Les fichiers ".dll"

Ces fichiers contiennent le code des bibliothèques utilisées dans le programme. Les bibliothèques sont liées dynamiquement au programme c'est pourquoi ces fichiers sont obligatoires au programme.

Le dossier "log"

Ce dossier contient les fichiers de journalisation de l'application. A chaque exécution du programme, un fichier est créé dans ce dossier. Ce fichier contient le timestamp de la création (pas très lisible au premier coup d'oeil). Pour visionner le dernier, vous pouvez trier par dernière date de modification.

Le dossier "ressources"

Ce dossier contient les différentes ressources utilisées par le programme. Il contient donc les images, les sons etc. Il contient différents fichier et dossier important :

- Le dossier *games*
- Le fichier *config.ini*
- Les images et les sons du jeu, ainsi que la police d'écriture utilisée

Le dossier *games* contient un sous-dossier pour chaque jeu que l'application. Le nom de ces sous-dossiers doivent impérativement être en minuscules. Par exemple, si l'application contient un jeu nommé "Remember", le sous-dossier devra s'appeler "remember".

Ce sous-dossier contient les images et les sons relatif au jeu (pour ne pas surcharger la racine "ressources")

Présentation du dossier du projet

Le dossier de développement du programme se trouve dans le dossier code. Il contient différents dossier et fichier également :

- Le dossier “include”
- Le dossier “lib”
- Le dossier “ressources” (même dossier que pour l’exécution)
- Le dossier “src”
- Les fichiers spécifique à visual studio (.sln, .vcxproj...)
- Un fichier CMakeLists.txt (fichier spécifique à la compilation avec Cmake)
- Un fichier premake5.lua

Le dossier “include” contient tout les fichier “.h” et “.hpp” des bibliothèques de l’application.

Le dossier “lib” contient tout les fichiers nécessaires à la compilation et à l’exécution du programme. C’est-à-dire les fichier “.dll” vu précédemment ainsi que les fichiers “.lib”.

Le fichier “premake5.lua” est un utilitaire utilisant le programme premake permettant de générer les fichiers visual studio plus simplement. Il peut être exécuté via la commande “`premake5 vs2017 --file=./premake5.lua`”.

Le dossier “src” contient tous les fichiers sources de l’application. Veuillez vous référer à la partie suivante pour plus de détail.

Les différentes classes

Les sources se divisent en plusieurs fichiers C++ (des fichier header “.h” et des fichiers sources “.cpp”). Ces fichiers représentent tous des classes du programme. Pour plus de détails, veuillez vous référer au fichier *Diagrams.asta*. Il contient les diagrammes UML du projet et ont été réalisé via le logiciel Astah. Voici la description des classes les plus importantes du programme.

La classe Application.

Cette classe est la classe principal du programme et contient la méthode “run” et “init”.

La méthode “run” permet de lancer le programme. Elle contient la boucle principal du programme. Elle va aussi s’occuper d’appeler la méthode “init”.

La méthode *init* va quand à elle initialiser le programme, c’est-à-dire instancier toutes les classes les plus importantes du programme. Les classes instanciées dans cette méthode le sont souvent qu’une fois. Elles seront ensuite transférées à tous les objets nécessitant une de ces fonctionnalités.

La classe Window

La classe *Window* permet de gérer l’affichage de l’application. Elle permet de générer les différentes textures SDL à partir des images dans le dossier “ressources”. Cette classe est la seule à pouvoir afficher ces textures à l’écran, il est donc indispensable de transmettre la référence de cet objet dans tous les composants affichables à l’écran.

La classe *CursorMovementManager* et les classes *Cursor*, *MouseCursor* et *TobiiCursor*

La classe *CursorMovementManager* permet de gérer l’emplacement des différents curseurs. Les curseurs sont représenté par l’interface “*Cursor*”. Cette interface permet de mettre à disposition une méthode permettant de récupérer la position d’un curseur et d’indiquer par la même occasion si cette information est valide ou si elle ne l’est pas.

Les classes implémentant l’interface *Cursor* sont au nombre de deux : la classe *MouseCursor* et la classe *TobiiCursor*.

La classe *Mouse cursor* permet d’avoir la position de la souris et la classe *TobiiCursor* permet d’interagir avec le dispositif Tobii pour retrouver la position des yeux. Pour plus d’information sur le fonctionnement des algorithmes présent dans cette classe vous pouvez vous référer au rapport du projet.

La classe *InputManager*

Cette classe empruntée à un autre projet est un singleton permettant de gérer les entrées clavier de l’utilisateur. Elle permet aussi de détecter quand l’utilisateur souhaite quitter l’application via la croix rouge disponible dans la barre logiciel de Windows.

La classe *Config*

La classe *Config* est reliée au fichier *config.ini*. Elle permet de lire chaque ligne de ce fichier et de déterminer s’il contient une configuration ou non. Chaque configuration est sous la forme “key=value”. Le fichier de configuration peut aussi contenir des commentaires via le caractère “#” (il ne pourra donc pas apparaître dans une ligne de configuration). Les commentaires peuvent se trouver sur une ligne à part entière ou à la fin d’une ligne de configuration.

La classe *StatisticsManager*

La classe *StatisticsManager* permet de créer et de lire des statistiques sur les jeux. Chaque statistique se trouve dans le sous-dossier du jeu dans le dossier ressources. Elles sont contenue dans le fichier *statistics.txt*. Ce fichier contient des couples clés/valeurs de la même forme que les configurations. Il n’est cependant pas possible de mettre des commentaires étant donné que le fichier est réécrit à chaque exécution.

La classe MusicManager

Cette classe est purement destinée à utiliser des sons dans le programme. Elle permet de lancer une musique en arrière plan ou de lancer des sons plus simple appelés SFX. Elle utilise la bibliothèque *SDL_Mixer*.