

Pour créer une nouvelle lib graphique dans notre en encapsulation nous devons prendre en compte deux parties

Ajouter Une Lib Graphique

I. Le Core et Le Makefile,

Nous devons ajouter la nouvelle librairie en tant que "shared librairies" (.so), dans le Makefile du Core et Graphic et également dans le header du core.cpp que contient les différentes "shared librairies" avec "./lib/" + "my_lib.so"

II. Le Dossier graphique

Il doit se nommer du nom de votre lib, puis ajouter déclarer les fonctions de l'interface IGraphic, qui contient le prototype de toute les fonctions utiles à la création de la lib,

Init() -> crée la lib ainsi que les fenetres/surfaces

_Draw() -> diffère chaque object affichable en fonction de son type pour le draw (text, Image, UIbox)

_DrawImage() -> crée la texture, size, position et affiche sur la fenêtre

_DrawText() -> affiche le texte, la position, la couleur,

_DrawUIbox() -> affiche des sous écran, à hauteur/largeur/posX /posY / la couleur et les bords

_Destroy() -> détruit tous les éléments crée lors de l'initialisation

_PollEvent() -> récupère les inputs en fonctions de la lib et les renvoie sous la forme de namespace ou de caractère ascii

RefreshWindows() -> réaffiche tous les éléments et nettoie la fenetre

Pour créer un nouveau jeuxgraphique dans notre en encapsulation nous devons prendre en compte deux parties

Ajouter Un Jeux

III. Le Core et Le Makefile,

Nous devons ajouter la nouvelle librairie en tant que “shared librairies” (.so), dans le Makefile du Game et du Core et également dans le header du core.cpp que contient les différentes “shared librairies” avec “./lib/” + “my_game.so”

IV. Le Dossier game

Il doit se nommer du nom de votre lib, puis ajouter déclarer les fonctions de l’interface IGraphic, qui contient le prototype de toute les fonctions utiles à la création de la lib,

Init() -> initialise tous les objects nécessaire au bon fonctionnement du jeux

_ Update() -> gère les inputs recus depuis le core et modifiies les propriétés des objects déjà instancier

_end() -> crée la texture, size, position et affiche sur la fenêtre

_DumpObject() -> return les objects Graphics,

_GetgameName() -> Récupère le nom du jeux actuels