Arcade - Documentation

How to implement new graphics libraries ?

To implement a new graphic library, go to **src/lib** directory and create a **new folder** named by the new library you want to implement (ex : *openGL*).

Inside this new folder, create 2 files : ***NewLibraryName.hpp*** and ***NewLibraryName.cpp*** (ex : *OpenGL.hpp* and *OpenGL.cpp*).

In your *NewLibraryName.hpp* file, **create a class** named by your library.
It must **inherit the IGraph class** so you need to **include "../IGraph.hpp".**

As you can see in IGraph.hpp, you need to **implement the following function** :

- **openWindow()** : open a 900x900 window named "Arcade *NewLibraryName*"

- **closeWindow()** : clear and close properly the window.

- **displaySprite(std::vector<std::pair<std::string, Entity*>>)** : This function displays all the sprites. It receives a map of Entities that contains all information requested to display sprites (each entity represents a sprite and its information). You can access to these information thank's to these functions :
    -- getName() -> return the name of the sprite
    -- getPos() -> return the position where the Sprite should be displayed
    -- getAngle() -> return the display angle of the sprite
    -- getSize() -> return the display size of the sprite
    -- getZBuffer() -> return the sprite's number to know which order to display it

Your displaySprite() function should really follow these steps :
    - sort the Entities map by their ZBuffer number
    - clear the previous window
    - Loop in entities map to get or load texture, set position, size and angle then display sprite.

- **displaySprite(std::vector<std::pair<std::string, Text*>>) :** This function has exactly the same goal than the previous displaySprite but for displaying text. It receives a map of Texts that contains all informations needed to display text. You can access to these information thank's to these functions :
    -- getText() -> return the text
    -- getPos() -> return the position where the text should be display
    -- getSize() -> return the display size of the text
    -- getZBuffer() -> return the text's number to know which order to display it

Your displayText() function should really follow these steps :

- sort the Texts map by their ZBuffer number
- Loop in texts map to set a font, get the text, set the position, the color and the size then display the text.

- **getInput()** : return an Input (see **src/Input.hpp** to see the enumeration and get thet event we need to get)
For exemple : if the right click of mouse is pressed, getInput() must return RIGHT_CLICK, if the key "B" of keyboard is pressed it must return PREVLIB, etc.

You can also add all function you want or need in your *NewLibraryName.hpp*

Feel free to look at the other existing files (in folder sfml, sdl etc) for help and inspiration.

Don't forget to extern your library like that

```
extern "C" NewLibraryName *entry()
{
	return new NewLibraryName();
}
```

exemple :

```
extern "C" OpenGL *entry()
{
	return new OpenGL();
}
```

How to implement new games ?

To implement a new game, go to **src/games** directory and create a **new folder** named by the new game you want to implement (ex : pacman).

Inside this new folder, create 2 files : ***NewGameName.hpp*** and ***NewGameName.cpp*** (ex : *Pacman.hpp* and *Pacman.cpp*).

In your *NewGameName.hpp* file, **create a class** named by your game.
It must **inherit the AGame class** so you need to **include "../IGame.hpp".**

KEYBOARD COMMANDS

O Previous game

P Next game

B Previous library

N Next library

Esc Quit

Entrer Confirm

Movements