

May 31, 16 7:55

GraphicElement.cpp

Page 1/2

```
#include "GraphicElement.h"
#include "View.h"
#include "Model.h"
#include "Ninja.h"
#include <iostream>

/*
 * -----
 * "THE BEER-WARE LICENSE" (Revision 42):
 * Antoine BOULANGAM-^I & Pierre_Elliot CABRERA wrote this file. As long as you
retain this notice you
 * can do whatever you want with this stuff. If we meet some day, and you think
 * this stuff is worth it, you can buy me a beer in return
 * -----
 */

//=====
// Constructeurs
//=====
GraphicElement::GraphicElement(const sf::Texture &image, int x, int y, int w, in
t h, int nbSpriteHorizontal, int nbSpriteVertical)
: _x(x), _y(y), _w(w), _h(h){
    this->setTexture(image); // dÃ©finit la texture de la balle
    this->setPosition(sf::Vector2f(_x,_y)); // dÃ©finit la position du sprite de
la balle
    this->resize(w, h, nbSpriteHorizontal, nbSpriteVertical);
}

GraphicElement::GraphicElement(const sf::Texture &image, int x, int y, int w, in
t h, bool semiTransparent)
: _x(x), _y(y), _w(w), _h(h){
    this->setTexture(image); // dÃ©finit la texture de la balle
    this->setPosition(sf::Vector2f(_x,_y)); // dÃ©finit la position du sprite de
la balle
    this->resize(w, h);
    if(semiTransparent)
        this->setColor(sf::Color(255, 255, 255, 128));
}

GraphicElement::GraphicElement(const sf::Texture &image, int w, int h)
: _w(w), _h(h){
    this->setTexture(image); // dÃ©finit la texture de la balle
    this->resize(w, h);
}

//=====
// Dessine sur la fenÃªtre
//=====
void GraphicElement::draw(sf::RenderWindow * window){
    window->draw(*this);
}

//=====
// Synchronise la position de l'image
// suivant la position de l'Ã©lÃ©ment qu'il
// reprÃ©sente
//=====
void GraphicElement::synchronize(int x, int y){
    this->setPosition(x, y);
}

//=====
// Redimensionne une image
//=====
void GraphicElement::resize(int w, int h){
    sf::FloatRect bounds = this->getLocalBounds();
    float width_factor = 1.0 * w / (bounds.width);
    float height_factor = 1.0 * h / (bounds.height);
```

May 31, 16 7:55

GraphicElement.cpp

Page 2/2

```
    this->setScale(width_factor, height_factor);
}

//=====
// Redimensionne une image en fonction
// du nombre de sprites sur l'image
//=====
void GraphicElement::resize(int w, int h, int nbSpriteHorizontal, int nbSpriteVe
rtical){
    // retourne les positions et taille rÃ©elles de s
    sf::FloatRect bounds = this->getLocalBounds();
    // facteur de mise Ã  l'Ã©chelle pour la largeur
    float width_factor = 1.0 * w / (bounds.width / nbSpriteHorizontal);
    // facteur de mise Ã  l'Ã©chelle pour la hauteur
    float height_factor = 1.0 * h / (bounds.height / nbSpriteVertical);
    this->setScale(width_factor, height_factor);
}

//=====
// RedÃ©finie la taille du GraphicElement
// et l'agrandit ou le rÃ©trÃ©cit en fonction
//=====
void GraphicElement::setSize(int w, int h){
    _w = w;
    _h = h;
    this->resize(w, h);
}

//=====
// Accesseurs
//=====
int GraphicElement::getX(){ return this->getPosition().x; }

int GraphicElement::getY(){ return this->getPosition().y; }

int GraphicElement::getH(){ return _h; }

int GraphicElement::getW(){ return _w; }
```