Documentation

Arcade Project

Antony Fantapié / Auguste Thomann / Yann Simon
Maxime Owaller / Killian Pin / Pierre Jeannin

Summary

Drawables

Structure Name	Description
Vector	Vector composed of 2 elements
Vector3	Vector composed of 3 elements
Color	Describes a color (with red, green, blue and alpha field)
Position	Describes a position on the screen (with x and y as a percentage of the screen size)
Size	Describes the size of an element on the screen (with x and y as a percentage of the screen size
Rect	Describes a rectangle shape (with position, size and color field)
Circle	Describes a circular shape (with position, radius and color field)
Text	Describes a text (with position, radius, color and text field)
Sprite	Describes a sprites (with position, size and path to the texture field)

Vector

Public member functions

Return type	Function name	Description
Vector	Vector(T px, T py)	Structure constructor
Vector	operator +(const Vector &other)	Return the sum of the two Vectors
Vector	operator +=(const Vector &other)	Compute the sum of the two vector and assign this value to the first one
Vector	operator -(const Vector &other)	Return the difference of the two Vectors
Vector	operator -=(const Vector &other)	Compute the difference of the two vector and assign this value to the first one

Public fields

Type	Name	Description
T	x	x Value of the vector
T	y	y Value of the vector

```
typedef Vector<float> Vector2f

typedef Vector<int> Vector2i

typedef Vector<unsigned int> Vector2u
```

Vector3

Public member functions

Return type	Function name	Description
Vector3	Vector3(T px, T py, Tpz)	Structure constructor
Vector3	operator +(const Vector3 &other)	Return the sum of the two Vectors
Vector3	operator +=(const Vector3 &other)	Compute the sum of the two vector and assign this value to the first one
Vector3	operator -(const Vector3 &other)	Return the difference of the two Vectors
Vector3	operator -=(const Vector3 &other)	Compute the difference of the two vector and assign this value to the first one

Public fields

Type	Name	Description
T	x	x Value of the vector
T	y	y Value of the vector
T	z	z Value of the vector

Color

Public member fonctions

Return type	Function Name	Description
Color	Color(unsigned char const red, unsigned char const gree, unsigned char const blue, unsigned char const alpha),	Structure constructor
Color	Color(const Color &)	Copy constructor
bool	operator ==(const Color &other)	Compares the color with the 'other' color and return 'true' if they are equal. Return false otherwise
Color	Red()	Creates an instance of Color which corresponds to red
Color	Green()	Creates an instance of Color which corresponds to green
Color	Blue()	Creates an instance of Color which corresponds to blue
Color	Black()	Creates an instance of Color which corresponds to black
Color	White()	Creates an instance of Color which corresponds to white
Color	Yellow()	Creates an instance of Color which corresponds to yellow
Color	Orange()	Creates an instance of Color which corresponds

		to orange
Color	Pink()	Creates an instance of Color which corresponds to pink
Color	Transparent()	Creates an instance of Color which corresponds to transparent

Public fields

Type	Name	Description
unsigned char	red	red Value
unsigned char	green	green Value
unsigned char	blue	blue Value
unsigned char	alpha	alpha Value

Position

(inherited from Vector2f)

Public member functions

Return Type	Name	Description
Position	Position(const float x, const float y)	Structure constructor
Position	Position(const Position &)	Copy Constructor
Position	operator =(const Position &other)	Set the first Position to the same value as the second

Size

(inherited from Vector2f)

Public member functions

Return Type	Name	Description
Size	Size(const float x, const float y)	Structure constructor
Size	Size(const Size &)	Copy Constructor
Size	operator =(const Size &other)	Set the first Size to the same value as the second

Rect

Public member functions

Return Type	Name	Description
Rect	Rect(const Position pos, const Size size, const Color color)	Structure constructor

Public fields

Type	Name	Description
Position	pos	Position the rectangle (as percentage of the window)
Size	size	Size of the rectangle (as percentage of the window)
Color	color	Color or the rectangle

Circle

Public member functions

Return Type	Name	Description
Circle	Circle(const Position pos, const float radius, const Color color)	Structure constructor

Public fields

Type	Name	Description
Position	pos	Position the center of the circle (as percentage of the window)
Float	radius	radius of the circle (as percentage of the window width)
Color	color	Color or the circle

Text

Public member functions

Return Type	Name	Description
Text	Text(const std::string &text, const Position pos, const Size size, const Color color)	Structure constructor
const std::string &	getText()	Return the '_text' field of the structure
void	setText(const std::string &newText)	Set the '_text' field of the structure to 'newText'

Public fields

Type	Name	Description
Position	pos	Position the rectangle (as percentage of the window)
Size	size	Size of the rectangle (as percentage of the window)
Color	color	Color or the rectangle

Private fields

Type	Name	Description
std::string	_text	String contained by the structure

Sprite

Public member functions

Return Type	Name	Description
Sprite	Sprite(const std::string &path, const Position pos, const Size size)	Structure constructor
const std::string &	getTextuePath()	Return the '_texturepath' field of the structure

Public fields

Type	Name	Description
Position	pos	Position the rectangle (as percentage of the window)
Size	size	Size of the rectangle (as percentage of the window)

Private fields

Type	Name	Description
std::string	_texturePath	Path to the texture that the sprite should display

IGame

Class Name	Description
IGame	Interface to use to set up a game. The core uses this interface to interact with the game

IGame

Public member functions

Return type	Function name	Description
void	event(const Event event)	The core sends user events to the game via this function
void	update(int elapsedTime)	This function should be used to update what happened in the game during ‘elapsedTime’ (ex: character movements). It handle the logic of the game
void	render(IGraphicRender &renderer)	This function should be used to display elements on the screen thanks to the IGraphicRender
const int	getScore()	This function the score when the game is over

IGraphics

Define	Description
RESOLUTION_X 1920	Window's width when it is possible
RESOLUTION_Y 1080	Window's height when it is possible
WINDOW_NAME "arcade"	Name of the window
Enumeration name	Description
Event	Enumeration of possible event used by the core and the game
Class name	Description
IGraphicRender	Class that contain all fonction needed to draw on the screen
IGraphic	Class which encapsulates the graphics library

Event

Name	Description	Key
unknown	Unknown event	
up	Move up	up arrow or z
down	Move down	down arrow or s
left	Move left	left arrow or q
right	Move right	right arrow or d
menu	Go back menu	escape
enter	Press enter key	enter
next_game	show next game	v
prev_game	show perv game	c
next_graphic	change current graphic library by the following one	x
prev_graphic	change current graphic library b the previous one	w
restart	restart the game	r
quit	quit arcade	close window or suppr

IGraphicRender

Public member functions

Return type	Function name	Description
void	drawRect(const Rect &rect)	Draws a rectangle on the window according to the one passed as parameter
void	drawCircle(const Circle &circle)	Draws a circle on the window according to the one passed as parameter
void	drawSprite(const Sprite &sprite)	Draw the sprite passed as parameter on the window
void	drawText(const Text &text)	Draw the text passed as parameter on the window
void	refresh()	Call the refresh function of the graphic library to update the rendering
void	clear()	Clear the window

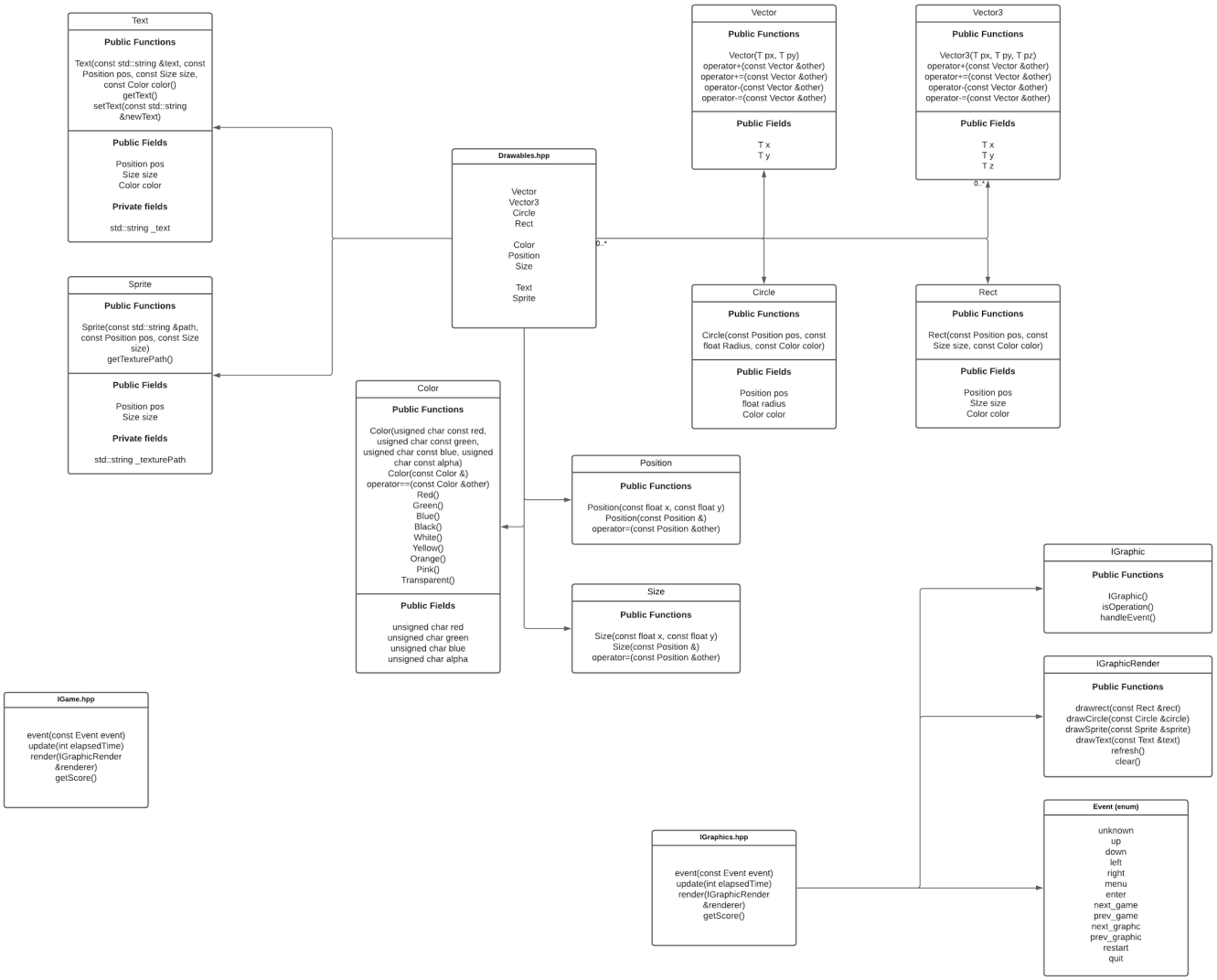
IGraphic

(inherited from IGraphicRender)

Public member functions

Return type	Function name	Description
IGraphic	IGraphic()	Class constructor
bool	isOperational()	Return ‘true’ if the graphic library is ready to use and ‘false’ otherwise
Event	handleEvent()	Return the last event registered by the graphic library

UML


Text
Public Functions
Text(const std::string &text, const Position pos, const Size size, const Color color()
getText()
setText(const std::string &newText)
Public Fields
Position pos
Size size
Color color
Private fields
std::string _text
Sprite
Public Functions
Sprite(const std::string &path, const Position pos, const Size size)
getTexturePath()
Public Fields
Position pos
Size size
Private fields
std::string _texturePath
Drawables.hpp
Vector
Vector3
Circle
Rect
Color
Position
Size
Text
Sprite
0..*
Vector
Public Functions
Vector(T px, T py)
operator+(const Vector &other)
operator+=(const Vector &other)
operator-(const Vector &other)
operator-=(const Vector &other)
Public Fields
T x
T y
Vector3
Public Functions
Vector3(T px, T py, T pz)
operator+(const Vector &other)
operator+=(const Vector &other)
operator-(const Vector &other)
operator-=(const Vector &other)
Public Fields
T x
T y
T z
0..*
Circle
Public Functions
Circle(const Position pos, const float Radius, const Color color)
Public Fields
Position pos
float radius
Color color
Rect
Public Functions
Rect(const Position pos, const Size size, const Color color)
Public Fields
Position pos
SIze size
Color color
Color
Public Functions
Color(usigned char const red, usigned char const green, usigned char const blue, usigned char const alpha)
Color(const Color &)
operator==(const Color &other)
Red()
Green()
Blue()
Black()
White()
Yellow()
Orange()
Pink()
Transparent()
Public Fields
unsigned char red
unsigned char green
unsigned char blue
unsigned char alpha
Position
Public Functions
Position(const float x, const float y)
Position(const Position &)
operator=(const Position &other)
Size
Public Functions
Size(const float x, const float y)
Size(const Position &)
operator=(const Position &other)
IGame.hpp
event(const Event event)
update(int elapsedTime)
render(IGraphicRender &renderer)
getScore()
IGraphic
Public Functions
IGraphic()
isOperation()
handleEvent()
IGraphicRender
Public Functions
drawrect(const Rect &rect)
drawCircle(const Circle &circle)
drawSprite(const Sprite &sprite)
drawText(const Text &text)
refresh()
clear()
IGraphics.hpp
event(const Event event)
update(int elapsedTime)
render(IGraphicRender &renderer)
getScore()
Event (enum)
unknown
up
down
left
right
menu
enter
next_game
prev_game
next_graphc
prev_graphic
restart
quit