arcade

0.1.0

Generated by Doxygen 1.9.1

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 5

Namespace Documentation

5.1 Arcade Namespace Reference

Main namespace for the Arcade project.

Classes

- class IGame

    Game Interface.
- class IRenderer

    Graphic Interface.
- class ISprite

    Sprite Interface.
- class IWindow

    Window Interface.
- class Clock
- class Time
- class Core

    Core class.
- class RuntimeException

    RuntimeException class.
- class Sdl2Renderer

    Sdl2Renderer class.
- class Sdl2Window

    Sdl2Window class.
- class NCursesRenderer

    NCursesRenderer class.
- class NCursesWindow

    NCursesWindow class.
- class SfmlRenderer

    SfmlRenderer class.
- class SfmlWindow

    SfmlWindow class.

Enumerations

- enum class **CoreMode** { **GAME** , **MENU** , **PAUSE** , **QUIT** }
- enum class **GameMode** {
  GAME , **GAME_OVER** , **MENU** , **PAUSE** ,
  QUIT }
- enum class KeyboardEvents {
  UP , **DOWN** , **LEFT** , **RIGHT** ,
  CLICK , **ESC** , **F1** , **ENTER** ,
  NONE }

  Enum for the KeyboardEvents.

5.1.1 Detailed Description

Main namespace for the Arcade project.

5.1.2 Enumeration Type Documentation

5.1.2.1 KeyboardEvents

```
enum Arcade::KeyboardEvents  [strong]
```

Enum for the KeyboardEvents.

Set of enum used for the game event

Chapter 6

Class Documentation

6.1 Arcade::Clock Class Reference

Public Types

- using TimePoint = std::chrono::time_point< std::chrono::high_resolution_clock >

  TimePoint is a type alias for a time point which is a very long and complicated type in the standard library.

Public Member Functions

- Clock ()

  Construct a new Clock object.
- void restart ()

  Restart the clock.
- Time getElapsedTime () const

  Get the elapsed time since the last restart.

6.1.1 Member Function Documentation

6.1.1.1 getElapsedTime()

```
Time Arcade::Clock::getElapsedTime ( ) const
```

Get the elapsed time since the last restart.

Returns

Time The elapsed time

The documentation for this class was generated from the following file:

- Core/include/Arcade/Clock/Clock.hpp

6.2 Arcade::Core Class Reference

Core class.

```
#include <Arcade/Core.hpp>
```

Classes

- class CoreException

  CoreException class.

Public Member Functions

- **Core** (const Core &)=delete
- Core & **operator=** (const Core &)=delete
- **Core** (const Core &&)=delete
- Core & **operator=** (const Core &&)=delete
- int runArcade (const std::string &path)

  Run Arcade.
- void getLibraries (const std::string &libPath)

  Get actives libraries.
- void switchGraphicLibrary ()

  Switch Graphic library.
- template<typename T >
  void switchLib (const std::string &libPath)

  Switch library.
- void mainLoop ()

  Game loop.
- void handleEvents (const KeyboardEvents &event)

  Handle events.
- void setMode (const CoreMode &gameMode)

  Set the game mode.
- void loadMenu ()

  load Menu
- void displayMenu ()

  display menu
- IRenderer ∗ getRenderer ()

  Get the renderer.
- void **moveCursorDown** ()
- void **moveCursorUp** ()
- void **launchGame** ()
- void **setGameIndex** (size_t index)

Static Public Member Functions

- static size_t getNextLibIndex (std::vector< std::string > &libs, size_t currentIndex)

  Get next library index.
- static size_t getPreviousLibIndex (std::vector< std::string > &libs, size_t currentIndex)

  Get previous library index.

6.2.1 Detailed Description

Core class.

Core class for the Arcade project

6.2.2 Member Function Documentation

6.2.2.1 getNextLibIndex()

```
static size_t Arcade::Core::getNextLibIndex (
            std::vector< std::string > & libs,
            size_t currentIndex )  [static]
```

Get next library index.

Parameters

libs	
currentIndex	

Returns

size_t

6.2.2.2 getPreviousLibIndex()

```
static size_t Arcade::Core::getPreviousLibIndex (
            std::vector< std::string > & libs,
            size_t currentIndex )  [static]
```

Get previous library index.

Parameters

libs	
currentIndex	

Returns

size_t

6.2.2.3 getRenderer()

```
IRenderer* Arcade::Core::getRenderer ( )  [inline]
```

Get the renderer.

Returns

The renderer

6.2.2.4 handleEvents()

```
void Arcade::Core::handleEvents (
             const KeyboardEvents & event )
```

Handle events.

Parameters

event	

6.2.2.5 mainLoop()

```
void Arcade::Core::mainLoop ( )
```

Game loop.

Main game loop

6.2.2.6 runArcade()

```
int Arcade::Core::runArcade (
             const std::string & path )
```

Run Arcade.

Parameters

path	

Returns

int

6.2.2.7 setMode()

```
void Arcade::Core::setMode (
             const CoreMode & gameMode )  [inline]
```

Set the game mode.

Parameters

gameMode	

6.2.2.8 switchLib()

```
template<typename T >
void Arcade::Core::switchLib (
             const std::string & libPath )
```

Switch library.

Template Parameters

T	

Parameters

libPath	

The documentation for this class was generated from the following file:

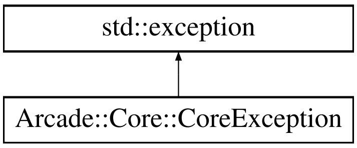
- Core/include/Arcade/Core.hpp

6.3 Arcade::Core::CoreException Class Reference

CoreException class.

```
#include <Arcade/Core.hpp>
```

Inheritance diagram for Arcade::Core::CoreException:



Public Member Functions

- **CoreException** (std::string msg)
- **CoreException** (const CoreException &)=delete
- CoreException & **operator=** (const CoreException &)=delete
- **CoreException** (const CoreException &&)=delete
- CoreException & **operator=** (const CoreException &&)=delete
- const char ∗ **what** () const noexcept override

6.3.1 Detailed Description

CoreException class.

Exception class for the Core class

The documentation for this class was generated from the following file:

- Core/include/Arcade/Core.hpp

6.4 Arcade::IGame Class Reference

Game Interface.

```
#include <Arcade/abstractions/IGame.hpp>
```

Public Member Functions

- virtual void start ()=0

  start the game
- virtual void stop ()=0

  stop the game

6.4.1 Detailed Description

Game Interface.

Set of method to implement for the business logic of the Arcade project

The documentation for this class was generated from the following file:

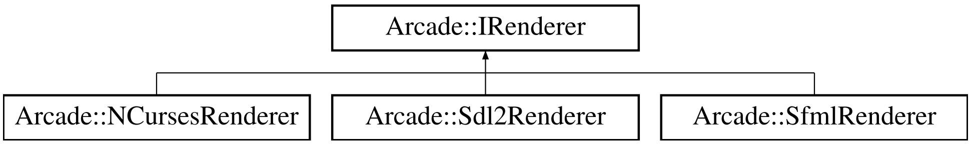
- Core/include/Arcade/abstractions/IGame.hpp

6.5 Arcade::IRenderer Class Reference

Graphic Interface.

```
#include <Arcade/abstractions/IRenderer.hpp>
```

Inheritance diagram for Arcade::IRenderer:



Public Member Functions

- virtual IWindow ∗ getWindow ()=0

  Retrieve the window associated with the renderer.
- virtual KeyboardEvents getEvent ()=0

  Retrieve the current event from the window.
- virtual bool isASCII ()=0

  Check if the renderer supports only ASCII characters.
- virtual bool loadTexture (const std::string &filepath, const std::string &name)=0

  Load a texture from a file.
- virtual bool loadFont (const std::string &filepath, const std::string &name)=0

  Load a font from a file.
- virtual void createSprite (const std::string &textureName, float pos_x, float pos_y, float scale_x, float scale↩ _y)=0

  Create a sprite with the specified parameters.
- virtual void displaySprite (const std::string &spriteName)=0

  Display a sprite with the specified name.
- virtual void createText (const std::string &fontName, const std::string &text, int size, float pos_x, float pos_y)=0

  Create a text object with the specified parameters.
- virtual void displayText (const std::string &textName)=0

  Display a text object with the specified name.
- virtual void **moveText** (const std::string &textName, int pos_x, int pos_y)=0
- virtual void **loadSound** ()=0
- virtual void **stopSound** ()=0

6.5.1 Detailed Description

Graphic Interface.

Set of method for the Arcade project

6.5.2 Member Function Documentation

6.5.2.1 createSprite()

```
virtual void Arcade::IRenderer::createSprite (
             const std::string & textureName,
             float pos_x,
             float pos_y,
             float scale_x,
             float scale_y )  [pure virtual]
```

Create a sprite with the specified parameters.

Parameters

textureName	The name of the texture to use for the sprite
pos_x	The x-coordinate of the sprite's position
pos_y	The y-coordinate of the sprite's position
scale_x	The scale factor for the sprite's width
scale_y	The scale factor for the sprite's height

Implemented in Arcade::SfmlRenderer, Arcade::Sdl2Renderer, and Arcade::NCursesRenderer.

6.5.2.2 createText()

```
virtual void Arcade::IRenderer::createText (
             const std::string & fontName,
             const std::string & text,
             int size,
             float pos_x,
             float pos_y )  [pure virtual]
```

Create a text object with the specified parameters.

Parameters

fontName	The name of the font to use for the text
text	The text to display
size	The size of the text
pos_x	The x-coordinate of the text's position
pos_y	The y-coordinate of the text's position

Implemented in Arcade::SfmlRenderer, Arcade::Sdl2Renderer, and Arcade::NCursesRenderer.

6.5.2.3 displaySprite()

```
virtual void Arcade::IRenderer::displaySprite (
             const std::string & spriteName )  [pure virtual]
```

Display a sprite with the specified name.

Parameters

spriteName	The name of the sprite to display

Implemented in Arcade::SfmlRenderer, Arcade::Sdl2Renderer, and Arcade::NCursesRenderer.

6.5.2.4 displayText()

```
virtual void Arcade::IRenderer::displayText (
             const std::string & textName )  [pure virtual]
```

Display a text object with the specified name.

Parameters

textName	The name of the text to display

Implemented in Arcade::SfmlRenderer, Arcade::Sdl2Renderer, and Arcade::NCursesRenderer.

6.5.2.5 getEvent()

```
virtual KeyboardEvents Arcade::IRenderer::getEvent ( )  [pure virtual]
```

Retrieve the current event from the window.

Returns

The current event

Implemented in Arcade::SfmlRenderer, Arcade::Sdl2Renderer, and Arcade::NCursesRenderer.

6.5.2.6 getWindow()

```
virtual IWindow* Arcade::IRenderer::getWindow ( )  [pure virtual]
```

Retrieve the window associated with the renderer.

Returns

The window associated with the renderer

Implemented in Arcade::SfmlRenderer, Arcade::Sdl2Renderer, and Arcade::NCursesRenderer.

6.5.2.7 isASCII()

```
virtual bool Arcade::IRenderer::isASCII ( )  [pure virtual]
```

Check if the renderer supports only ASCII characters.

Returns

True if the renderer supports ASCII, false otherwise

Implemented in Arcade::SfmlRenderer, Arcade::Sdl2Renderer, and Arcade::NCursesRenderer.

6.5.2.8 loadFont()

```
virtual bool Arcade::IRenderer::loadFont (
            const std::string & filepath,
            const std::string & name )  [pure virtual]
```

Load a font from a file.

Parameters

filepath	The path to the font file
name	The name to associate with the font

Returns

True if the font was loaded successfully, false otherwise

Implemented in Arcade::SfmlRenderer, Arcade::Sdl2Renderer, and Arcade::NCursesRenderer.

6.5.2.9 loadTexture()

```
virtual bool Arcade::IRenderer::loadTexture (
            const std::string & filepath,
            const std::string & name )  [pure virtual]
```

Load a texture from a file.

Parameters

filepath	The path to the texture file
name	The name to associate with the texture

Returns

True if the texture was loaded successfully, false otherwise

Implemented in Arcade::SfmlRenderer, Arcade::NCursesRenderer, and Arcade::Sdl2Renderer.

The documentation for this class was generated from the following file:

- Core/include/Arcade/abstractions/IRenderer.hpp

6.6 Arcade::ISprite Class Reference

Sprite Interface.

```
#include <Arcade/abstractions/ISprite.hpp>
```

Public Member Functions

- **ISprite** (const ISprite &)=delete
- ISprite & **operator=** (const ISprite &)=delete
- virtual void createSprite (const std::vector< std::string > &sprite)=0

  create the sprite
- virtual void draw ()=0

  draw the sprite
- virtual void setPosition (int x, int y)=0

  set the position of the sprite

6.6.1 Detailed Description

Sprite Interface.

Set of method to implement for the business logic of the Arcade project

6.6.2 Member Function Documentation

6.6.2.1 createSprite()

```
virtual void Arcade::ISprite::createSprite (
            const std::vector< std::string > & sprite )  [pure virtual]
```

create the sprite

Parameters

sprite	

create the sprite with the given sprite, can be a filePath, or a ASCII art

6.6.2.2 setPosition()

```
virtual void Arcade::ISprite::setPosition (
             int x,
             int y )  [pure virtual]
```

set the position of the sprite

Parameters

x	
y	

The documentation for this class was generated from the following file:

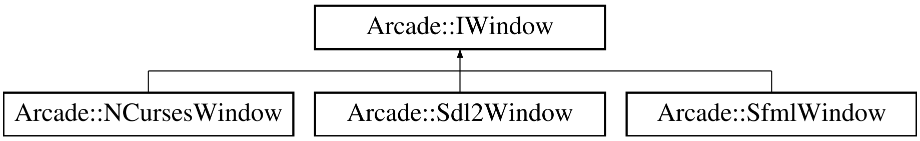
- Core/include/Arcade/abstractions/ISprite.hpp

6.7 Arcade::IWindow Class Reference

Window Interface.

```
#include <Arcade/abstractions/IWindow.hpp>
```

Inheritance diagram for Arcade::IWindow:



Public Member Functions

- virtual void openWindow (unsigned int width, unsigned int height)=0

    Create a window with the specified dimensions.
- virtual void displayWindow ()=0

    Display the current window.
- virtual void clearWindow ()=0

    Clear the contents of the current window.
- virtual void closeWindow ()=0

    Close the current window.
- virtual void setTitle (const std::string &title)=0

    Set the title of the window.

6.7.1 Detailed Description

Window Interface.

Set of method to implement for the business logic of the Arcade project

6.7.2 Member Function Documentation

6.7.2.1 openWindow()

```
virtual void Arcade::IWindow::openWindow (
             unsigned int width,
             unsigned int height )  [pure virtual]
```

Create a window with the specified dimensions.

Parameters

width	The width of the window
height	The height of the window

Implemented in Arcade::SfmlWindow, Arcade::Sdl2Window, and Arcade::NCursesWindow.

6.7.2.2 setTitle()

```
virtual void Arcade::IWindow::setTitle (
             const std::string & title )  [pure virtual]
```

Set the title of the window.

Parameters

title	The new title of the window

Implemented in Arcade::SfmlWindow, Arcade::Sdl2Window, and Arcade::NCursesWindow.

The documentation for this class was generated from the following file:

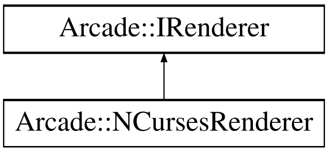
- Core/include/Arcade/abstractions/IWindow.hpp

6.8 Arcade::NCursesRenderer Class Reference

NCursesRenderer class.

```
#include <Arcade/NCursesRenderer.hpp>
```

Inheritance diagram for Arcade::NCursesRenderer:



Public Member Functions

- KeyboardEvents getEvent () override

  Retrieve the current event from the window.
- bool loadTexture (const std::string &texture, const std::string &name) override

  Load a texture from a file.
- void createSprite (const std::string &textureName, float pos_x, float pos_y, float scale_x, float scale_y) override

  Create a sprite with the specified parameters.
- void displaySprite (const std::string &spriteName) override

  Display a sprite with the specified name.
- bool loadFont (const std::string &filepath, const std::string &name) override

  Load a font from a file.
- void createText (const std::string &fontName, const std::string &text, int size, float pos_x, float pos_y) override

  Create a text object with the specified parameters.
- void displayText (const std::string &textName) override

  Display a text object with the specified name.
- void **moveText** (const std::string &textName, int pos_x, int pos_y) override
- bool isASCII () override

  Check if the renderer supports only ASCII characters.
- IWindow ∗ getWindow () override

  Retrieve the window associated with the renderer.
- void **loadSound** () override
- void **stopSound** () override

6.8.1 Detailed Description

NCursesRenderer class.

NCursesRenderer class for the Arcade project

6.8.2 Member Function Documentation

6.8.2.1 createSprite()

```
void Arcade::NCursesRenderer::createSprite (
             const std::string & textureName,
             float pos_x,
             float pos_y,
             float scale_x,
             float scale_y )  [inline], [override], [virtual]
```

Create a sprite with the specified parameters.

Parameters

textureName	The name of the texture to use for the sprite
pos_x	The x-coordinate of the sprite's position
pos_y	The y-coordinate of the sprite's position
scale_x	The scale factor for the sprite's width
scale_y	The scale factor for the sprite's height

Implements Arcade::IRenderer.

6.8.2.2 createText()

```
void Arcade::NCursesRenderer::createText (
            const std::string & fontName,
            const std::string & text,
            int size,
            float pos_x,
            float pos_y )  [inline], [override], [virtual]
```

Create a text object with the specified parameters.

Parameters

fontName	The name of the font to use for the text
text	The text to display
size	The size of the text
pos_x	The x-coordinate of the text's position
pos_y	The y-coordinate of the text's position

Implements Arcade::IRenderer.

6.8.2.3 displaySprite()

```
void Arcade::NCursesRenderer::displaySprite (
            const std::string & spriteName )  [inline], [override], [virtual]
```

Display a sprite with the specified name.

Parameters

spriteName	The name of the sprite to display

Implements Arcade::IRenderer.

6.8.2.4 displayText()

```
void Arcade::NCursesRenderer::displayText (
            const std::string & textName )  [inline], [override], [virtual]
```

Display a text object with the specified name.

Parameters

textName	The name of the text to display

Implements Arcade::IRenderer.

6.8.2.5 getEvent()

```
KeyboardEvents Arcade::NCursesRenderer::getEvent ( )  [override], [virtual]
```

Retrieve the current event from the window.

Returns

The current event

Implements Arcade::IRenderer.

6.8.2.6 getWindow()

```
IWindow* Arcade::NCursesRenderer::getWindow ( )  [inline], [override], [virtual]
```

Retrieve the window associated with the renderer.

Returns

The window associated with the renderer

Implements Arcade::IRenderer.

6.8.2.7 isASCII()

```
bool Arcade::NCursesRenderer::isASCII ( )  [inline], [override], [virtual]
```

Check if the renderer supports only ASCII characters.

Returns

True if the renderer supports ASCII, false otherwise

Implements Arcade::IRenderer.

6.8.2.8 loadFont()

```
bool Arcade::NCursesRenderer::loadFont (
            const std::string & filepath,
            const std::string & name )  [inline], [override], [virtual]
```

Load a font from a file.

Parameters

filepath	The path to the font file
name	The name to associate with the font

Returns

True if the font was loaded successfully, false otherwise

Implements Arcade::IRenderer.

6.8.2.9 loadTexture()

```
bool Arcade::NCursesRenderer::loadTexture (
            const std::string & filepath,
            const std::string & name )  [inline], [override], [virtual]
```

Load a texture from a file.

Parameters

filepath	The path to the texture file
name	The name to associate with the texture

Returns

True if the texture was loaded successfully, false otherwise

Implements Arcade::IRenderer.

The documentation for this class was generated from the following file:

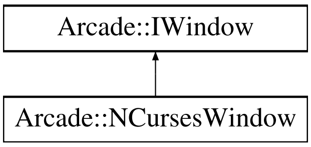
- Graphics/NCurses/include/Arcade/NCursesRenderer.hpp

6.9 Arcade::NCursesWindow Class Reference

NCursesWindow class.

```
#include <Arcade/NCursesWindow.hpp>
```

Inheritance diagram for Arcade::NCursesWindow:



Public Member Functions

- void openWindow (unsigned int width, unsigned int height) override

  Create a window with the specified dimensions.
- void displayWindow () override

  Display the current window.
- void clearWindow () override

  Clear the contents of the current window.
- void closeWindow () override

  Close the current window.
- void setTitle (const std::string &title) override

  Set the title of the window.
- WINDOW ∗ getWindow ()

  Get the current window.

6.9.1 Detailed Description

NCursesWindow class.

NCursesWindow class for the Arcade project

6.9.2 Member Function Documentation

6.9.2.1 getWindow()

```
WINDOW* Arcade::NCursesWindow::getWindow ( )  [inline]
```

Get the current window.

Returns

The current window

6.9.2.2 openWindow()

```
void Arcade::NCursesWindow::openWindow (
             unsigned int width,
             unsigned int height )  [override], [virtual]
```

Create a window with the specified dimensions.

Parameters

width	The width of the window
height	The height of the window

Implements Arcade::IWindow.

6.9.2.3 setTitle()

```
void Arcade::NCursesWindow::setTitle (
             const std::string & title )  [inline], [override], [virtual]
```

Set the title of the window.

Parameters

title	The new title of the window

Implements Arcade::IWindow.

The documentation for this class was generated from the following file:

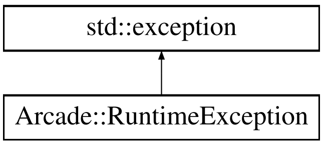
- Graphics/NCurses/include/Arcade/NCursesWindow.hpp

6.10 Arcade::RuntimeException Class Reference

RuntimeException class.

`#include <Arcade/RuntimeException.hpp>`

Inheritance diagram for Arcade::RuntimeException:



Public Member Functions

- **RuntimeException** (std::string msg)
- **RuntimeException** (RuntimeException const &)=delete
- const char ∗ **what** () const noexcept override

6.10.1 Detailed Description

RuntimeException class.

RuntimeException class for the Arcade project

The documentation for this class was generated from the following file:

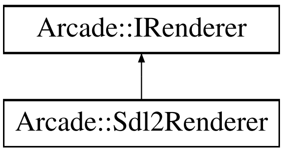
- Core/include/Arcade/RuntimeException.hpp

6.11 Arcade::Sdl2Renderer Class Reference

Sdl2Renderer class.

```
#include <Arcade/Sdl2Renderer.hpp>
```

Inheritance diagram for Arcade::Sdl2Renderer:



Public Member Functions

- KeyboardEvents getEvent () override

  Retrieve the current event from the window.
- bool loadTexture (const std::string &filePath, const std::string &name) override

  Load a texture from a file.
- bool loadFont (const std::string &filepath, const std::string &name) override

  Load a font from a file.
- void createSprite (const std::string &textureName, float pos_x, float pos_y, float scale_x, float scale_y) override

  Create a sprite with the specified parameters.
- void createText (const std::string &fontName, const std::string &text, int size, float pos_x, float pos_y) override

  Create a text object with the specified parameters.
- void displaySprite (const std::string &spriteName) override

  Display a sprite with the specified name.
- void displayText (const std::string &textName) override

  Display a text object with the specified name.
- void **moveText** (const std::string &textName, int pos_x, int pos_y) override
- bool isASCII () override

  Check if the renderer supports only ASCII characters.
- IWindow ∗ getWindow () override

  Retrieve the window associated with the renderer.
- void **loadSound** () override
- void **stopSound** () override

6.11.1 Detailed Description

Sdl2Renderer class.

Sdl2Renderer class for the Arcade project

6.11.2 Member Function Documentation

6.11.2.1 createSprite()

```
void Arcade::Sdl2Renderer::createSprite (
             const std::string & textureName,
             float pos_x,
             float pos_y,
             float scale_x,
             float scale_y )  [override], [virtual]
```

Create a sprite with the specified parameters.

Parameters

textureName	The name of the texture to use for the sprite
pos_x	The x-coordinate of the sprite's position
pos_y	The y-coordinate of the sprite's position
scale_x	The scale factor for the sprite's width
scale_y	The scale factor for the sprite's height

Implements Arcade::IRenderer.

6.11.2.2 createText()

```
void Arcade::Sdl2Renderer::createText (
             const std::string & fontName,
             const std::string & text,
             int size,
             float pos_x,
             float pos_y )  [override], [virtual]
```

Create a text object with the specified parameters.

Parameters

fontName	The name of the font to use for the text
text	The text to display
size	The size of the text
pos_x	The x-coordinate of the text's position
pos_y	The y-coordinate of the text's position

Implements Arcade::IRenderer.

6.11.2.3 displaySprite()

```
void Arcade::Sdl2Renderer::displaySprite (
             const std::string & spriteName )  [override], [virtual]
```

Display a sprite with the specified name.

Parameters

spriteName	The name of the sprite to display

Implements Arcade::IRenderer.

6.11.2.4 displayText()

```
void Arcade::Sdl2Renderer::displayText (
             const std::string & textName )  [override], [virtual]
```

Display a text object with the specified name.

Parameters

textName	The name of the text to display

Implements Arcade::IRenderer.

6.11.2.5 getEvent()

```
KeyboardEvents Arcade::Sdl2Renderer::getEvent ( )  [override], [virtual]
```

Retrieve the current event from the window.

Returns

The current event

Implements Arcade::IRenderer.

6.11.2.6 getWindow()

```
IWindow* Arcade::Sdl2Renderer::getWindow ( )  [inline], [override], [virtual]
```

Retrieve the window associated with the renderer.

Returns

The window associated with the renderer

Implements Arcade::IRenderer.

6.11.2.7 isASCII()

```
bool Arcade::Sdl2Renderer::isASCII ( )  [inline], [override], [virtual]
```

Check if the renderer supports only ASCII characters.

Returns

True if the renderer supports ASCII, false otherwise

Implements Arcade::IRenderer.

6.11.2.8 loadFont()

```
bool Arcade::Sdl2Renderer::loadFont (
             const std::string & filepath,
             const std::string & name )  [override], [virtual]
```

Load a font from a file.

Parameters

filepath	The path to the font file
name	The name to associate with the font

Returns

True if the font was loaded successfully, false otherwise

Implements Arcade::IRenderer.

6.11.2.9 loadTexture()

```
bool Arcade::Sdl2Renderer::loadTexture (
             const std::string & filepath,
             const std::string & name )  [override], [virtual]
```

Load a texture from a file.

Parameters

filepath	The path to the texture file
name	The name to associate with the texture

Returns

True if the texture was loaded successfully, false otherwise

Implements Arcade::IRenderer.

The documentation for this class was generated from the following file:

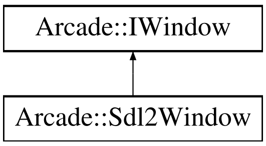
- Graphics/SDL2/include/Arcade/Sdl2Renderer.hpp

6.12 Arcade::Sdl2Window Class Reference

Sdl2Window class.

```
#include <Arcade/Sdl2Window.hpp>
```

Inheritance diagram for Arcade::Sdl2Window:



Public Member Functions

- void openWindow (unsigned int width, unsigned int height) override

  Create a window with the specified dimensions.
- void displayWindow () override

  Display the current window.
- void clearWindow () override

  Clear the contents of the current window.
- void closeWindow () override

  Close the current window.
- void setTitle (const std::string &title) override

  Set the title of the window.
- SDL_Window * getWindow ()

  Get the current window.
- SDL_Renderer * getRenderer ()

  Get the current renderer.
- int getWidth () const

  Get the width of the window.
- int getHeight () const

  Get the height of the window.

6.12.1 Detailed Description

Sdl2Window class.

Sdl2Window class for the Arcade project

6.12.2 Member Function Documentation

6.12.2.1 getHeight()

```
int Arcade::Sdl2Window::getHeight ( ) const  [inline]
```

Get the height of the window.

Returns

The height of the window

6.12.2.2 getRenderer()

```
SDL_Renderer* Arcade::Sdl2Window::getRenderer ( )  [inline]
```

Get the current renderer.

Returns

The current renderer

6.12.2.3 getWidth()

```
int Arcade::Sdl2Window::getWidth ( ) const  [inline]
```

Get the width of the window.

Returns

The width of the window

6.12.2.4 getWindow()

```
SDL_Window* Arcade::Sdl2Window::getWindow ( )  [inline]
```

Get the current window.

Returns

The current window

6.12.2.5 openWindow()

```
void Arcade::Sdl2Window::openWindow (
            unsigned int width,
            unsigned int height )  [override], [virtual]
```

Create a window with the specified dimensions.

Parameters

width	The width of the window
height	The height of the window

Implements Arcade::IWindow.

6.12.2.6 setTitle()

```
void Arcade::Sdl2Window::setTitle (
             const std::string & title )  [inline], [override], [virtual]
```

Set the title of the window.

Parameters

title	The new title of the window

Implements Arcade::IWindow.

The documentation for this class was generated from the following file:

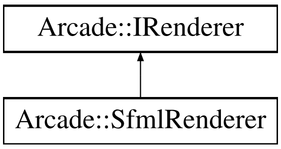
- Graphics/SDL2/include/Arcade/Sdl2Window.hpp

6.13 Arcade::SfmlRenderer Class Reference

SfmlRenderer class.

```
#include <Arcade/SfmlRenderer.hpp>
```

Inheritance diagram for Arcade::SfmlRenderer:



Public Member Functions

- KeyboardEvents getEvent () override

  Retrieve the current event from the window.
- bool loadTexture (const std::string &texture, const std::string &name) override

  Load a texture from a file.
- bool loadFont (const std::string &filepath, const std::string &name) override

Load a font from a file.
- void createSprite (const std::string &textureName, float pos_x, float pos_y, float scale_x, float scale_y) override

  Create a sprite with the specified parameters.
- void createText (const std::string &fontName, const std::string &text, int size, float pos_x, float pos_y) override

  Create a text object with the specified parameters.
- void displaySprite (const std::string &spriteName) override

  Display a sprite with the specified name.
- void displayText (const std::string &textName) override

  Display a text object with the specified name.
- void **moveText** (const std::string &textName, int pos_x, int pos_y) override
- bool isASCII () override

  Check if the renderer supports only ASCII characters.
- IWindow ∗ getWindow () override

  Retrieve the window associated with the renderer.
- void **loadSound** () override
- void **stopSound** () override

6.13.1 Detailed Description

SfmlRenderer class.

SfmlRenderer class for the Arcade project

6.13.2 Member Function Documentation

6.13.2.1 createSprite()

```
void Arcade::SfmlRenderer::createSprite (
             const std::string & textureName,
             float pos_x,
             float pos_y,
             float scale_x,
             float scale_y ) [override], [virtual]
```

Create a sprite with the specified parameters.

Parameters

textureName	The name of the texture to use for the sprite
pos_x	The x-coordinate of the sprite's position
pos_y	The y-coordinate of the sprite's position
scale_x	The scale factor for the sprite's width
scale_y	The scale factor for the sprite's height

Implements Arcade::IRenderer.

6.13.2.2 createText()

```
void Arcade::SfmlRenderer::createText (
            const std::string & fontName,
            const std::string & text,
            int size,
            float pos_x,
            float pos_y )  [override], [virtual]
```

Create a text object with the specified parameters.

Parameters

fontName	The name of the font to use for the text
text	The text to display
size	The size of the text
pos_x	The x-coordinate of the text's position
pos_y	The y-coordinate of the text's position

Implements Arcade::IRenderer.

6.13.2.3 displaySprite()

```
void Arcade::SfmlRenderer::displaySprite (
            const std::string & spriteName )  [override], [virtual]
```

Display a sprite with the specified name.

Parameters

spriteName	The name of the sprite to display

Implements Arcade::IRenderer.

6.13.2.4 displayText()

```
void Arcade::SfmlRenderer::displayText (
            const std::string & textName )  [override], [virtual]
```

Display a text object with the specified name.

Parameters

textName	The name of the text to display

Implements Arcade::IRenderer.

6.13.2.5 getEvent()

```
KeyboardEvents Arcade::SfmlRenderer::getEvent ( )  [override], [virtual]
```

Retrieve the current event from the window.

Returns

The current event

Implements Arcade::IRenderer.

6.13.2.6 getWindow()

```
IWindow* Arcade::SfmlRenderer::getWindow ( )  [inline], [override], [virtual]
```

Retrieve the window associated with the renderer.

Returns

The window associated with the renderer

Implements Arcade::IRenderer.

6.13.2.7 isASCII()

```
bool Arcade::SfmlRenderer::isASCII ( )  [inline], [override], [virtual]
```

Check if the renderer supports only ASCII characters.

Returns

True if the renderer supports ASCII, false otherwise

Implements Arcade::IRenderer.

6.13.2.8 loadFont()

```
bool Arcade::SfmlRenderer::loadFont (
             const std::string & filepath,
             const std::string & name )  [override], [virtual]
```

Load a font from a file.

Parameters

filepath	The path to the font file
name	The name to associate with the font

Returns

True if the font was loaded successfully, false otherwise

Implements Arcade::IRenderer.

6.13.2.9 loadTexture()

```
bool Arcade::SfmlRenderer::loadTexture (
            const std::string & filepath,
            const std::string & name )  [override], [virtual]
```

Load a texture from a file.

Parameters

filepath	The path to the texture file
name	The name to associate with the texture

Returns

True if the texture was loaded successfully, false otherwise

Implements Arcade::IRenderer.

The documentation for this class was generated from the following file:

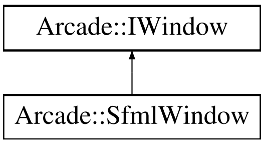
- Graphics/SFML/include/Arcade/SfmlRenderer.hpp

6.14 Arcade::SfmlWindow Class Reference

SfmlWindow class.

```
#include <Arcade/SfmlWindow.hpp>
```

Inheritance diagram for Arcade::SfmlWindow:



Public Member Functions

- void openWindow (unsigned int width, unsigned int height) override

  Create a window with the specified dimensions.
- void displayWindow () override

  Display the current window.
- void clearWindow () override

  Clear the contents of the current window.
- void closeWindow () override

  Close the current window.
- void setTitle (const std::string &title) override

  Set the title of the window.
- sf::RenderWindow ∗ getWindow ()

  Get the current window.

6.14.1 Detailed Description

SfmlWindow class.

SfmlWindow class for the Arcade project

6.14.2 Member Function Documentation

6.14.2.1 getWindow()

```
sf::RenderWindow* Arcade::SfmlWindow::getWindow ( ) [inline]
```

Get the current window.

Returns

The current window

6.14.2.2 openWindow()

```
void Arcade::SfmlWindow::openWindow (
             unsigned int width,
             unsigned int height ) [override], [virtual]
```

Create a window with the specified dimensions.

Parameters

width	The width of the window
height	The height of the window

Implements Arcade::IWindow.

6.14.2.3 setTitle()

```
void Arcade::SfmlWindow::setTitle (
            const std::string & title )  [inline], [override], [virtual]
```

Set the title of the window.

Parameters

title	The new title of the window

Implements Arcade::IWindow.

The documentation for this class was generated from the following file:

- Graphics/SFML/include/Arcade/SfmlWindow.hpp

6.15 Arcade::Time Class Reference

Public Member Functions

- Time (float seconds)

  Construct a new Time object.
- int asSeconds () const

  Transform the time to seconds.
- int asMilliseconds () const

  Transform the time to milliseconds.
- int asMicroseconds () const

  Transform the time to microseconds.

The documentation for this class was generated from the following file:

- Core/include/Arcade/Clock/Time.hpp

Chapter 7

File Documentation

7.1 Core/include/Arcade/abstractions/IGame.hpp File Reference

File declaring Game Interface.

Classes

- class Arcade::IGame

  Game Interface.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.1.1 Detailed Description

File declaring Game Interface.

7.2 Core/include/Arcade/abstractions/IRenderer.hpp File Reference

File declaring Graphic Interface.

```
#include "Arcade/Enum.hpp"
#include "Arcade/abstractions/IWindow.hpp"
```

Classes

- class Arcade::IRenderer

  Graphic Interface.

Namespaces

- Arcade

    Main namespace for the Arcade project.

7.2.1 Detailed Description

File declaring Graphic Interface.

7.3 Core/include/Arcade/abstractions/ISprite.hpp File Reference

File declaring Sprite Interface.

```
#include <iostream>
#include <vector>
```

Classes

- class Arcade::ISprite

    Sprite Interface.

Namespaces

- Arcade

    Main namespace for the Arcade project.

7.3.1 Detailed Description

File declaring Sprite Interface.

7.4 Core/include/Arcade/abstractions/IWindow.hpp File Reference

File declaring Window Interface.

```
#include <iostream>
```

Classes

- class Arcade::IWindow

    Window Interface.

Namespaces

- Arcade

    Main namespace for the Arcade project.

7.4.1 Detailed Description

File declaring Window Interface.

7.5 Core/include/Arcade/Constants.hpp File Reference

File containing constant values used in the Arcade project.

Variables

- constexpr int EPITECH_ERROR = 84

    Exit code for Epitech errors.
- constexpr int SUCCESS = 0

    Exit code indicating that the program has executed successfully.
- constexpr int ERROR = -1

    Exit code indicating that an error has occurred.
- constexpr int WIDTH = 1920

    Width of the window.
- constexpr int HEIGHT = 1080

    Height of the window.
- constexpr char const ∗ **LIB_PATH** = "./lib/"

7.5.1 Detailed Description

File containing constant values used in the Arcade project.

This file provides a set of constant values that are used throughout the Arcade project.

7.6 Core/include/Arcade/Core.hpp File Reference

File declaring Core.

```
#include <memory>
#include <vector>
#include "Arcade/Enum.hpp"
#include "Arcade/abstractions/IRenderer.hpp"
#include "Arcade/abstractions/IWindow.hpp"
#include "Arcade/abstractions/IGame.hpp"
```

Classes

- class Arcade::Core

  Core class.
- class Arcade::Core::CoreException

  CoreException class.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.6.1 Detailed Description

File declaring Core.

7.7 Core/include/Arcade/Enum.hpp File Reference

File declaring Enum.

Namespaces

- Arcade

  Main namespace for the Arcade project.

Enumerations

- enum class **CoreMode** { **GAME** , **MENU** , **PAUSE** , **QUIT** }
- enum class **GameMode** {
  GAME , **GAME_OVER** , **MENU** , **PAUSE** ,
  QUIT }
- enum class Arcade::KeyboardEvents {
  UP , **DOWN** , **LEFT** , **RIGHT** ,
  CLICK , **ESC** , **F1** , **ENTER** ,
  NONE }

  Enum for the KeyboardEvents.

7.7.1 Detailed Description

File declaring Enum.

Set of enum used in the Arcade project

7.8 Core/include/Arcade/RuntimeException.hpp File Reference

File declaring RuntimeException.

```
#include <iostream>
#include <exception>
```

Classes

- class Arcade::RuntimeException

  RuntimeException class.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.8.1 Detailed Description

File declaring RuntimeException.

7.9 Graphics/NCurses/include/Arcade/NCursesRenderer.hpp File Reference

File declaring NCursesRenderer.

```
#include <ncurses.h>
#include "Arcade/abstractions/IRenderer.hpp"
#include "Arcade/NCursesWindow.hpp"
```

Classes

- class Arcade::NCursesRenderer

  NCursesRenderer class.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.9.1 Detailed Description

File declaring NCursesRenderer.

7.10 Graphics/NCurses/include/Arcade/NCursesWindow.hpp File Reference

File declaring NCursesWindow.

```
#include <ncurses.h>
#include "Arcade/abstractions/IWindow.hpp"
```

Classes

- class Arcade::NCursesWindow

  NCursesWindow class.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.10.1 Detailed Description

File declaring NCursesWindow.

7.11 Graphics/SDL2/include/Arcade/Sdl2Renderer.hpp File Reference

File declaring Sdl2Renderer class.

```
#include <map>
#include <SDL2/SDL.h>
#include <SDL2/SDL_ttf.h>
#include <SDL2/SDL_image.h>
#include <SDL2/SDL_mixer.h>
#include "Arcade/abstractions/IRenderer.hpp"
#include "Arcade/Sdl2Window.hpp"
```

Classes

- class Arcade::Sdl2Renderer

  Sdl2Renderer class.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.11.1 Detailed Description

File declaring Sdl2Renderer class.

7.12 Graphics/SDL2/include/Arcade/Sdl2Window.hpp File Reference

File declaring Sdl2Window class.

```
#include <SDL2/SDL.h>
#include <SDL2/SDL_ttf.h>
#include <SDL2/SDL_image.h>
#include "Arcade/abstractions/IWindow.hpp"
```

Classes

- class Arcade::Sdl2Window

  Sdl2Window class.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.12.1 Detailed Description

File declaring Sdl2Window class.

7.13 Graphics/SFML/include/Arcade/SfmlRenderer.hpp File Reference

File declaring SfmlRenderer class.

```
#include <SFML/Graphics.hpp>
#include <SFML/Audio.hpp>
#include "Arcade/abstractions/IRenderer.hpp"
#include "Arcade/abstractions/IWindow.hpp"
#include "Arcade/SfmlWindow.hpp"
```

Classes

- class Arcade::SfmlRenderer

  SfmlRenderer class.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.13.1 Detailed Description

File declaring SfmlRenderer class.

7.14 Graphics/SFML/include/Arcade/SfmlWindow.hpp File Reference

File declaring SfmlWindow class.

```
#include <SFML/Graphics.hpp>
#include "Arcade/abstractions/IRenderer.hpp"
#include "Arcade/abstractions/IWindow.hpp"
```

Classes

- class Arcade::SfmlWindow

  SfmlWindow class.

Namespaces

- Arcade

  Main namespace for the Arcade project.

7.14.1 Detailed Description

File declaring SfmlWindow class.

Index