

# ARCADE

## Project Documentation



### **Team members:**

- Ambroise Jacquement Ramirez
- Etienne Namur
- Lucia Jimenez

# Index

- Description .....2
- Components.....3-5
  - Arcade (executable)
  - Interfaces
    - IObject
    - IEvent
    - IDisplayModule
    - IGameModule
- Usage.....6
- Contributions.....7
- Project Sequence.....8

## Description

The Arcade Project is a modular gaming platform that allows loading and playing various games, as well as interacting with them through graphical and textual interfaces. It provides a flexible architecture for easily adding new games and graphical interfaces.

# Components

## 1- Arcade(executable)

The main executable that loads the shared libraries (.so) of games and graphical/textual interfaces, and manages their execution.

## 2- Interfaces

Interfaces are defined in '`Common/Arcade.hpp`' file and it includes:

### `arcade::IObject`

Interface to represent objects in the game.

#### Methods:

- '`getRGB()`' : Returns the RGB values of the object's color (between 0 and 255 for each color).
- '`getPos()`' : Returns the coordinates (x, y) of the object's position.
- '`getSize()`' : Return the dimensions (width, height) of the object.
- '`getAsset()`' : Returns the name of the asset associated with the object.
- '`getAscii()`' : Return the ASCII value representing the object.
- '`getType()`' : Returns the type of the object (Graphic, Text, Music, Font).

### **arcade::IEvent:**

Interface to represent events.

#### Methods:

- **'type()'** : Returns the type of the event. The most important ones are:
  - "Right"
  - "Left"
  - "Down"
  - "Up"
  - "Switch"
  - "Escape"
  - "Enter"
  - "Back Space"

### **arcade::IDisplayModule:**

Interface for graphical display modules.

#### Methods:

- **'display(const data\_t& data)'** : Displays the game data.
- **'clear()'** : Clears the screen.
- **'getEvent()'** : Returns a vector of events.

## **arcade::IGameModule:**

Interface for game modules.

### **Methods:**

- **'manageEvent(std::vector<std::unique\_ptr<IEvent>> const&)'** : Manages game events.
- **'iterate()'** : Moves to the next step of the game.
- **'getMap()'** : Returns the game map data.

## Usage

To use the Arcade Project, simply compile the `'arcade'` executable and provide the appropriate shared libraries of the fames and graphical/textual interfaces.

Example

```
./arcade ./lib/arcade_sdl2.so
```

# Contributions

The Arcade Project welcomes contributions. If you wish to add new games or graphical interfaces, feel free to create them.

- - -

This documentation provides an overview of the Arcade Project, highlighting its main components and describing usage and contribution to the project.



# Project Sequence

This is an overview of how the main project sequence works:

