

Guide de démarrage

Comment installer, compiler et utiliser le projet ARCADE

Prérequis

Avant de démarrer avec le projet ARCADE, assurez-vous d'avoir les éléments suivants installés sur votre système :

- ✓ Compilateur C++ supportant C++17
- ✓ Make ou CMake
- ✓ Bibliothèque ncurses
- ✓ Bibliothèque SDL2 avec SDL2_image et SDL2_ttf
- ✓ Bibliothèque SFML

Installation des dépendances

Ubuntu/Debian

Copy

```
sudo apt-get update
sudo apt-get install build-essential cmake libncurses5-dev
sudo apt-get install libsdl2-dev libsdl2-image-dev libsdl2-ttf-dev
sudo apt-get install libsfml-dev
```

Fedora

Copy

```
sudo dnf update
sudo dnf install gcc-c++ cmake ncurses-devel
sudo dnf install SDL2-devel SDL2_image-devel SDL2_ttf-devel
sudo dnf install SFML-devel
```

macOS (avec Homebrew)

Copy

```
brew update
brew install cmake ncurses
brew install sdl2 sdl2_image sdl2_ttf
brew install sfml
```

Téléchargement du projet

Clonez le dépôt du projet ARCADE à partir de GitHub :

Terminal

Copy

```
git clone https://github.com/nogbeat/8-OOP-400-COT-4-1-arcade-james.gbetchedji.git
cd 8-OOP-400-COT-4-1-arcade-james.gbetchedji
```

ou Telecharger le Zip du projet :

Téléchargé le projet zip ⓘ

Terminal

Copy

```
unzip arcade
cd arcade
```

Compilation

Le projet ARCADE peut être compilé en utilisant Make ou CMake. Voici comment utiliser chaque méthode :

Compilation avec Make

Terminal

Copy

```
# Compiler l'ensemble du projet
make

# Compiler uniquement le core
make core

# Compiler uniquement les jeux
make games

# Compiler uniquement les bibliothèques graphiques
make graphics

# Nettoyer les fichiers objets
make clean

# Nettoyer les fichiers objets et binaires
make fclean

# Recompiler
make re
```

Compilation avec CMake

Terminal

Copy

```
mkdir -p build && cd build
cmake .. -G "Unix Makefiles" -DCMAKE_BUILD_TYPE=Release
cmake --build .
cd ..
```

Note: L'implémentation CMake est en cours de développement et peut ne pas être complètement fonctionnelle.

Utilisation

Pour lancer l'application ARCADE, vous devez spécifier une bibliothèque graphique au démarrage :

Terminal

Copy

```
# Lancer avec SDL2
./arcade ./lib/arcade_sdl2.so

# Ou lancer avec SFML
./arcade ./lib/arcade_sfml.so

# Ou lancer avec NCurses
./arcade ./lib/arcade_ncurses.so
```

Contrôles

Touche	Action
F5	Changer de bibliothèque graphique (précédente)
F6	Changer de bibliothèque graphique (suivante)
- / -	Changer de jeu (dans le menu)
i / l	Changer de bibliothèque graphique (dans le menu)
Enter	Lancer le jeu sélectionné
R	Redémarrer le jeu
M	Retourner au menu
N	Modifier le nom d'utilisateur
Flèches directionnelles	Contrôler le jeu
Espace	Action (selon le jeu)
Echap	Quitter

Structure des dossiers

Le projet ARCADE est organisé selon la structure suivante :

Structure des dossiers

Copy


```
.
├── Makefile
├── CMakeLists.txt
├── assets
│   ├── fonts
│   └── duckhunt.ttf
├── Ressources
│   ├── bg.png
│   ├── ch.png
│   ├── duckhunt.ttf
│   ├── heart.png
│   ├── pause.png
│   ├── push.ogg
│   ├── push.wav
│   ├── score.png
│   └── ss.png
├── includes
│   ├── Core
│   │   ├── ArcadeCore.hpp
│   │   ├── DLoader.hpp
│   │   ├── EventManager.hpp
│   │   └── ScoreManager.hpp
│   ├── Display
│   │   ├── NCursesDisplay.hpp
│   │   ├── SDL2Display.hpp
│   │   └── SFMLDisplay.hpp
│   ├── Interfaces
│   │   ├── IDisplayModule.hpp
│   │   └── IGameModule.hpp
│   └── Utils
│       ├── Error.hpp
│       └── Types.hpp
├── srcs
│   ├── Core
│   │   ├── ArcadeCore.cpp
│   │   ├── DLoader.cpp
│   │   ├── DisplayGames.cpp
│   │   ├── EventManager.cpp
│   │   ├── Menu
│   │   │   ├── Menu_Ncurses.cpp
│   │   │   ├── Menu_SFML.cpp
│   │   │   └── Menu_sdl2.cpp
│   │   └── ScoreManager.cpp
│   ├── Games
│   │   ├── Hunter
│   │   │   ├── Hunter.cpp
│   │   │   ├── Hunter.hpp
│   │   │   └── HunterModule.cpp
│   │   ├── Nibbler
│   │   │   ├── Nibbler.cpp
│   │   │   ├── Nibbler.hpp
│   │   │   └── NibblerModule.cpp
│   │   ├── Snake
│   │   │   ├── Snake.cpp
│   │   │   ├── Snake.hpp
│   │   │   └── SnakeModule.cpp
│   ├── Graphics
│   │   ├── NCurses
│   │   │   ├── NCursesDisplay.cpp
│   │   │   └── NCursesModule.cpp
│   │   ├── SDL2
│   │   │   ├── SDL2Display.cpp
│   │   │   └── SDL2Module.cpp
│   │   ├── SFML
│   │   │   ├── SFMLDisplay.cpp
│   │   │   └── SFMLModule.cpp
│   └── main
│       └── main.cpp
├── lib
│   ├── Games
│   └── Graphics
└── doc
    ├── index.html
    ├── getting-started.html
    ├── core-arch.html
    ├── graphics.html
    ├── games.html
    ├── api.html
    ├── styles.css
    └── script.js
```

Prochaines étapes

Maintenant que vous avez installé et configuré ARCADE, vous pouvez :

-  [Explorer l'architecture](#)
[Comprendre comment fonctionne le cœur du système ARCADE](#)
-  [Créer une bibliothèque graphique](#)
[Apprendre à implémenter votre propre bibliothèque d'affichage](#)
-  [Créer un jeu](#)
[Développer un nouveau jeu compatible avec ARCADE](#)



 [Accueil](#)

 [Démarrage](#)

 [Architecture](#)

 [Biblio. Graphiques](#)

 [Créer un jeu](#)

 [API](#)

☐ Mode sombre