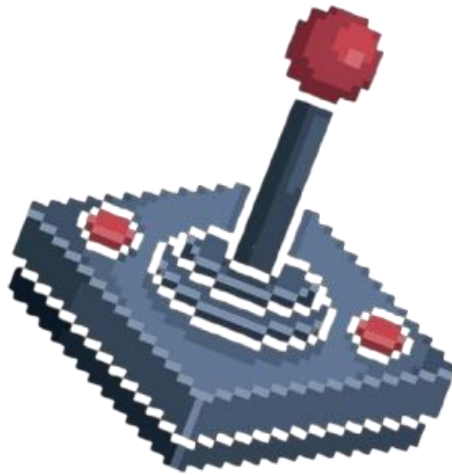


RETRO FIGHTERS ARCADE MANUAL



Retro Fighters

ENGLISH VERSION

TABLE OF CONTENTS

1. Turning on the Arcade
2. Core Selection
3. Selection of games
4. Shutting down the Arcade
5. Frontend Development Guide in C#
6. Database Design Guide
7. Arduino Interfacing Guide
8. Website design and development

TURNING ON THE ARCADE

FIRST STEPS

To turn on the arcade, press the button on the back of the cabinet to turn on the PC inside, at the same time, turn on the monitor using the corresponding button located in the lower right corner. Once the monitor is on, it will indicate this with a yellow light (Awaiting Signal Light) which informs the user that the monitor is on but waiting for the video signal from the PC. Once the PC is on, the monitor will display video and the light will turn green, showing the power status.

CONTINUING WITH THE START UP

After completing the previous steps, you must wait for the PC to complete the Boot process and for the Operating System to start (in this case, Windows 10). Once the boot process is complete, the PC desktop will be displayed with the Retro Fighters wallpaper. You will have to wait a few seconds for the system to automatically start the "retrofighters.exe" file in order to start using the emulator.

COMMON PROBLEMS

Retro Fighters Arcade start up:

- **PC TAKES TOO LONG TO COMPLETE BOOT PROCESS:**
This is completely normal, as this PC's components are considered obsolete compared to the current generation of CPUs, GPUs and other components. In this case, you don't have to do anything as the problem tends to resolve itself. If the OS doesn't start, you should restart the PC by holding down the power button.
- **PC DOES NOT TURN ON:**
If the PC does not turn on, the AC power cable should be checked to make sure they are properly connected to the PC. If this happens and the PC still does not turn on, it should be removed from the cabinet and checked by a technician or teacher for inspection and repair.
- **MONITOR DOES NOT DISPLAY VIDEO OR DOES NOT TURN ON:**
If the monitor does not display video or does not turn on, all the connections must be checked, from the power cable to the VGA cable that runs from the PC to the monitor itself. If the problem persists, consult a technician or a teacher.

SELECTION OF GAMES

HOME SCREEN

Once the program is started, the Arcade home screen will be displayed. This includes the *Retro Fighters Arcade* logo, the icons of each console to be emulated and a main section.



To select a core you must use the console filter.

START OF THE GAME

To start a game you must select the image of the desired game and wait for it to start.

FAQ

Here you can find some common questions and how to solve them:

- **HOW TO USE THE CORE FILTER?**

To use the core filter you must click on the icon of the console you do not want to see (For example: if you want to play PS1 games you must deactivate the rest of the consoles)

- **WHAT DO YOU NEED TO PLAY?**

To play *RetroFighters Arcade* you will need the buttons made by us in Arduino or a keyboard and mouse.

NOTES AND COMMENTS

If there is any problem related to the initial screen of *Retro Fighters Arcade* that is **NOT** specified in this document, you should contact a teacher so that a solution can be provided to said problem not previously specified by the creators of this arcade.

SHUTTING DOWN THE ARCADE

To turn off the Arcade you must close the *retrofightersarcade.exe* program and turn off the PC from the *Windows button* at the bottom left or hold down the button on the PC case.

FRONTEND DEVELOPMENT IN C#

1. Set up the environment

- Install **Visual Studio Community** and select the ".NET Desktop Application Development" workload.
 - Create a **Windows Forms** or **WPF project** .
-

2. Design the interface

- Use the visual designer to drag controls (buttons, labels, etc.).
 - Configure the properties from the corresponding window (text, colors, sizes).
-

3. Add functionality

- Double-click a control to create an event (such as a button click).
-

4. Execute the project

- Press **Ctrl + F5** or click **Run** .
 - Test and debug your program using Visual Studio tools.
-

5. Compile and distribute

- **Publish** option to generate an installer or executable file.

DATABASE DESIGN

1. Database design

- **Model:** An Entity Relationship Model was used to structure the tables and relationships.
- **Main tables:**
 - **Games:** Stores information such as name, platform and description.
 - **Users:** Register players.
 - **Score :** Records the scores
- **Technology used:** MySQL with XAMPP

2. Using the database

- **Insertion and update:** Information is managed through forms in the graphical interface (C#), connected to the database through SQL queries.
- **Queries:** SELECT commands are used to extract data and generate custom views.
- **Report generation:**
 - Reports are created dynamically with data extracted from tables, allowing filtering by platform.

ARDUINO INTERFACE DEVELOPMENT

1. Necessary materials

- **Arduino board** (e.g. Arduino Uno or Mega).
 - **Analog joystick** (two-axis module with push button).
 - **Jumper cables** (male-male or male-female depending on the module).
 - **Protoboard** or connection base.
 - **Additional connectors** if external buttons are needed.
-

2. Connecting the joystick to the Arduino

Identify the joystick pins and connect them to the Arduino:

- **Button 1: “i”; Pin 2 - Works like the z**
- **Button 2: “j”; Pin 3 - Works like the x**
- **Button 3 “k”; Pin 4 - Works like the a**
- **Button 4: “l”; Pin 5 - Works like the b**
- **Button 5: “w”; Pin 6 – Up arrow**
- **Button 6: “a”; Pin 7 - Down arrow**
- **Button 7: “s”; Pin 8 - Left arrow**
- **Button 8: “d”; Pin 9 - Right arrow**

WEBSITE DEVELOPMENT

1. Creation of the website

1. Plan the site layout

- Organize the page into key sections such as:
 - **Home:** Introduction to the project.
 - **Project Description:** Brief Explanation.
 - **Team:** Presentation of the members.
 - **Contact:** Information or form for inquiries.
- Define a clean, easy-to-navigate, mobile-friendly design.

2. Structure the page

- Use a markup language like HTML to create the skeleton of the page.
- Use CSS to customize the visual design: colors, fonts, and element layout.
- Implement a JavaScript script to dynamically switch between languages (Spanish and English), among other features.

3. Create bilingual content

- Write content in both languages and organize texts so that they can be easily selected by the user.
-

2. Page maintenance

1. Content update

- Change texts or add new sections as needed for the project.
- Maintain consistency between the English and Spanish versions.

2. Design settings

- Review the aesthetics and functionality of the site, making changes if the design no longer meets current needs.
- Test to ensure your site is accessible across different devices and browsers.

3. Optimization and performance

- Check the site's loading speed and reduce the size of images or resources if necessary.
 - Make sure navigation is smooth.
-

3. Publication

1. Upload the page to a server

- Use free services like **GitHub Pages** or **Netlify** to host your site.

2. Promote the site

- Share the page link on social media, gaming forums, and tech events.
-

4. Recommendations

- Keep the design simple and don't require frequent changes to the structure.
- Use visual tools such as icons or images that do not require translation to reduce the bilingual workload.
- Conduct regular reviews to ensure content is up-to-date and relevant.