



GAMING HUB

Professional C++ Arcade Game

Course: First Semester C++ Programming

Technology: C++ with raylib Graphics Library

Date: December 2025

1. Project Overview

Gaming Hub is a professional graphical arcade application developed in C++ using the raylib graphics library. It features 5 interactive mini-games with a modern user interface, animations, and mouse/keyboard controls.

Key Features

- **Professional GUI:** Beautiful graphics with animated backgrounds
- **5 Complete Games:** Various game genres included
- **Object-Oriented Design:** Each game is a separate class
- **Score Tracking:** Scores saved to text files
- **Responsive Controls:** Mouse and keyboard support
- **60 FPS Gameplay:** Smooth animations

2. Games Included



Game 1: Tic-Tac-Toe

Type: 2-Player Strategy Game

Objective: Get three X's or O's in a row, column, or diagonal.

Controls: Click on cells to place your symbol.

Game 2: Space Survivor

Type: Real-time Dodge Game

Objective: Survive as long as possible by avoiding falling asteroids.

Controls: A/D or Arrow Keys to move left/right.

Game 3: Break The Code

Type: Mastermind Puzzle

Objective: Guess the 4-digit secret code within 10 attempts.

Controls: Type digits 0-9, press Enter to submit guess.

Game 4: Escape Zombie Land

Type: Turn-based Survival

Objective: Navigate the grid to reach the Safe Zone while avoiding the zombie.

Controls: WASD or Arrow Keys to move.

Game 5: Swap Cards

Type: Memory Match Game

Objective: Find all 8 matching pairs of cards.

Controls: Click cards to flip and match them.

3. Code Structure

The project follows Object-Oriented Programming principles with the following structure:

```
GamingHub.cpp (Main Source File - ~1000 lines) ┌─ Headers & Includes ┌─ Constants
(Screen size, colors) ┌─ Namespace Colors (Color palette) ┌─ Utility Functions | ┌─
DrawTextCentered() | ┌─ DrawButton() | ┌─ DrawAnimatedBackground() ┌─ Game Classes
| ┌─ TicTacToeGame | ┌─ SpaceSurvivorGame | ┌─ BreakTheCodeGame | ┌─
ZombieLandGame | ┌─ SwapCardsGame ┌─ main() - Game Loop
```

Class Structure (Each Game)

```
class GameName { private: // Game state variables public: void Init(); //
Initialize/reset game bool Update(); // Handle input, update logic void Draw(); //
Render graphics };
```

4. Technologies Used

Technology	Purpose
C++	Core programming language
raylib	Graphics library for rendering
OpenGL	Graphics rendering (via raylib)
MinGW g++	Compiler

5. Key C++ Concepts Demonstrated

- **Classes & Objects:** Each game is encapsulated in its own class
- **Namespaces:** Colors namespace for organized constants

- **Vectors:** Dynamic arrays for asteroids, stars, particles
- **File I/O:** Saving scores using fstream
- **Control Structures:** Loops, conditionals, switch statements
- **Functions:** Modular code with reusable functions
- **Enums:** GameState enum for menu navigation
- **Random Numbers:** Game randomization using rand()

6. Setup & Installation Guide

Step 1: Install MinGW Compiler

1. Download MinGW from: <https://winlibs.com/>
2. Extract to c:\mingw64
3. Add c:\mingw64\bin to System PATH
4. Verify: Open CMD and type g++ --version

Step 2: Install raylib Library

1. Download from: <https://www.raylib.com/>
2. Choose: raylib-5.0_win64_mingw-w64.zip
3. Extract to project folder or c:\raylib

Step 3: Compile the Game

```
g++ -o GamingHub.exe GamingHub.cpp -I "raylib-5.0_win64_mingw-w64/include" -L "raylib-5.0_win64_mingw-w64/lib" -lraylib -lopengl32 -lgdi32 -lwinmm -mwindows
```

Step 4: Run the Game

GamingHub.exe

 **Quick Method:** Double-click compile.bat to automatically compile and run the game!

7. Controls Reference

Game/Screen	Controls
Main Menu	Click buttons to select game

All Games	Press M or ESC to return to menu
All Games	Press SPACE to restart
Tic-Tac-Toe	Click cells to place X/O
Space Survivor	A/D or ←/→ to move
Break The Code	Type digits, ENTER to submit
Zombie Land	WASD or Arrow Keys
Swap Cards	Click cards to flip

8. Files Generated

File	Description
GamingHub.cpp	Main source code
GamingHub.exe	Compiled executable
compile.bat	Compilation script
codebreaker_leaderboard.txt	Break The Code scores
zombie_escape_results.txt	Zombie Land results
swapcards_best_score.txt	Swap Cards best scores

9. Code Explanation (Key Sections)

Main Game Loop

```
while (!WindowShouldClose()) { // Update game logic based on current state switch  
    currentState) { case MENU: /* handle menu */ break; case TICTACTOE: if  
        (ticTacToe.Update()) currentState = MENU; break; // ... other games } BeginDrawing();  
    // Draw based on current state switch (currentState) { case MENU: DrawMenu(); break;  
    case TICTACTOE: ticTacToe.Draw(); break; // ... other games } EndDrawing(); }
```

Color Palette (Namespace)

```
namespace Colors { const Color BACKGROUND = {15, 15, 35, 255}; // Dark blue const  
Color PRIMARY = {100, 149, 237, 255}; // Cornflower blue const Color ACCENT = {255,  
215, 0, 255}; // Gold const Color SUCCESS = {50, 205, 50, 255}; // Lime green const  
Color DANGER = {220, 20, 60, 255}; // Crimson red }
```

Game Class Example (TicTacToe)

```
class TicTacToeGame { private: int board[9]; // 0=empty, 1=X, 2=O int currentPlayer;  
// 1 or 2 int winner; // 0=none, 1=X wins, 2=O wins, 3=draw public: void Init() { /*  
reset game state */ } bool Update() { /* handle input, check win */ } void Draw() { /*  
render grid and symbols */ } };
```

10. Screenshots

The game features a modern UI with:

- Animated particle background
- Hover effects on buttons
- Color-coded game elements
- Professional typography
- Smooth 60 FPS animations

11. Conclusion

Gaming Hub demonstrates practical application of C++ programming concepts including object-oriented design, graphics programming, file handling, and game development fundamentals. The project showcases how C++ can be used to create professional, interactive applications beyond simple console programs.

About raylib: raylib is a simple and easy-to-use library for learning videogames programming. It's designed for prototyping and tools development, with a focus on simplicity and ease of use. Website: www.raylib.com

Gaming Hub - Professional C++ Arcade

First Semester C++ Programming Project
Built with C++ and raylib Graphics Library