



C++ for Games Program

Build High-Performance Interactive Systems with C++

C++ is the backbone of most modern game engines and performance-intensive software. It provides **low-level control**, **object-oriented design**, and **high-speed execution**—making it the preferred language for building real-time applications, 3D engines, and complex simulations.

At **Certed Technologies**, our **C++ for Games Program** is designed to give you an in-depth command over C++ programming, with practical application in building **interactive, event-driven, and performance-optimized systems**—used extensively in game engines and real-time content pipelines.



Why Learn C++ in the Interactive World?

C++ powers core systems in Unreal Engine, Unity (via plugins), CryEngine, and custom game frameworks. It's also used for:

- **Gameplay programming**
- **Physics & rendering systems**
- **Multiplayer networking**
- **AI logic and engine architecture**

By learning C++, you unlock access to **engine-level customization**, **VR/AR applications**, and **native plugin development** for performance-critical environments.



Learn from Game-Oriented C++ Experts

Our trainers include C++ programmers with experience in **real-time simulations**, **interactive entertainment**, and **custom engine development**. They'll guide you through everything from **C++ fundamentals** to **designing components used in interactive applications**.



Program Highlights

Feature	Description
Duration	3–4 Months / 200–250 Hours
Delivery Mode	Hybrid (Online + Offline)
Tools & Platforms	C++17/20, Visual Studio, Unreal Engine (Optional), SDL2, Git
Project Work	4+ Projects + 1 Capstone: Build an Interactive 2D/3D System
Certifications	Industry-Recognized + C++ Game Systems Badge
Career Support	Resume, GitHub Setup, Interview Questions for Game Studios

What You'll Learn

Module 1: Core C++ Programming

- Variables, Data Types, Operators
- Control Flow & Loops
- Functions, Arrays, Strings

Module 2: Object-Oriented Design

- Classes, Objects, Inheritance
- Polymorphism, Abstraction
- Static, Dynamic Binding

Module 3: Advanced C++ Concepts

- Pointers & Memory Management
- Constructors, Destructors
- Templates & STL

- Exception Handling

Module 4: Real-Time Programming Concepts

- Game Loops & Frame Updates
- Keyboard/Mouse Event Handling
- Timers, Delta Time, State Machines

Module 5: Graphics & Interactivity (SDL2 / SFML / UE4 Optional)

- Rendering 2D Objects
- Basic Physics & Collision Detection
- Sound Integration & Animations

Module 6: Capstone Project

- Develop a Playable Interactive App
- Game Logic + Input + UI + Feedback
- Code Optimization & GitHub Deployment

Who Should Enroll?

This course is ideal for:

- CS/IT/Engineering students
- Developers aiming to work with Unreal Engine
- Programmers interested in performance-critical applications
- Professionals transitioning into the gaming or simulation domain



Career Outcomes

Gain foundational skills that map directly to job roles like:

- C++ Developer (Game Engine/Simulation)
- Gameplay Programmer
- Game Tools Programmer
- Engine Plugin Developer
- Technical Architect (Performance Systems)
- Simulation Software Engineer

Sectors: Gaming | Simulation | XR | Defence Tech | Embedded Systems | Real-Time Visualizations



Tools & Technologies Covered

C++ | Visual Studio | SDL2 | SFML | Unreal Engine (Blueprint/C++ Integration) | Git & GitHub | Object-Oriented Design | Debugging & Profiling Tools



Sample Project Ideas

- Terminal-based RPG or Maze Game
- SDL2-Based 2D Platformer
- C++ Game Timer with Scoreboard
- Turn-Based Battle Simulator
- Physics-Based Object Interaction Demo



Certification

Get a **Joint Certificate** from Certed Technologies + Industry Mentors along with a **C++ Game Systems Proficiency Badge** that highlights your command over interactive system design using C++.



What Makes This Course Unique?

- ✓ Tailored for Interactive & Game-Oriented Development
 - ✓ Builds Engine-Ready Skills for Unreal, SDL2, Custom Engines
 - ✓ Emphasis on Performance, Structure & Real-Time Concepts
 - ✓ GitHub-Ready Projects for Job Interviews
 - ✓ Compatible with Both Indie and AAA Game Pipelines
-



Upcoming Batch

Batch Start Date	Mode	Status
15 July 2025	Online	Open
22 July 2025	Offline	Few Seats Left
05 August 2025	Hybrid	Open



Custom C++ Training for Institutions & Game Studios

We offer specialized C++ training programs tailored for:

- Game Design/CS Colleges
- Simulation & Training Labs
- Studios building real-time 2D/3D engines


- AR/VR Solution Providers

Customization Options:

- C++ + Unreal Focus
- 2D/3D Engine Plugin Development
- C++ with Networking & AI Scripting
- Debugging & Optimization Workshops

Talk to an Advisor

 Email: support@certedtechnologies.com

 Call/WhatsApp: +91-8920158923 | +91-9009015026

Ready to Build Performance-Critical Interactive Systems?

“Whether it’s a game engine or a flight simulator, C++ is what makes the magic work behind the scenes. Master it and take control.”

 [Enroll Now](#)  [Request a Callback](#)

Frequently Asked Questions (FAQs)

- 1. Is this course beginner-friendly?**
Yes. It starts from scratch and gradually introduces advanced C++ concepts with use cases.
- 2. Is this only for game development?**
No. It's suitable for any performance-focused application like simulations, XR, or engine tools.

3. **Will I work with real-time logic?**

Absolutely. You'll build projects using game loop structures, event handling, and timers.

4. **Is there a capstone project?**

Yes. You'll design and build a complete interactive project using C++.

5. **Can I use this knowledge in Unreal Engine?**

Definitely. Unreal is built on C++—this program prepares you for that ecosystem.

6. **Are job roles available for C++ in games?**

Yes. Studios actively hire C++ programmers for gameplay, physics, tools, and networking.

7. **Is certification included?**

Yes. You'll receive a recognized certificate and a project-backed GitHub portfolio.

8. **Is team-based corporate training available?**

Yes. We customize delivery for studios, colleges, and training divisions.
