

# Eric Azevedo – Game Programmer

---

**E-mail:** eazevedo@digipen.edu **Phone:** (206) 412-1631

**Address:** 12418 110th Lane NE Apt. HH126, Kirkland, WA 98034

## Objective

A full-time game programming position, with an interest in graphics programming.

## Skills

**Languages/APIs:** C/C++ (5 years), C#, DirectX 9.0c, HLSL, Win32 API, DirectInput, Visual Basic, Java, ActionScript, Assembly, OpenGL, Winsock, FMOD.

**Development Tools:** Visual Studio, Perforce, Tortoise SVN, Photoshop, 3ds Max.

**Graphics Programming:** 2D/3D software and hardware rasterizing, HLSL shaders, post processing, shadow mapping, deferred shading, skinned hierarchical mesh animation, inverse kinematics, mipmapping, Phong lighting, bump/normal/environment/specular/opacity mapping, displacement mapping, projective texturing, height-map terrain rendering, reflections and refractions, cel-shading, hierarchical occlusion culling, view frustum culling, BSP trees, KD-trees, ray tracing.

**AI Programming:** A\* pathfinding, reflex agents, minimax with alpha-beta pruning.

**Physics Programming:** 2D collision and response, cloth simulation, PhysX integration.

**Miscellaneous:** STL, Adobe Premiere, Microsoft Office, fluent in written and spoken Spanish.

## Game Projects

**Junior/Senior Game:** *Attack of the 50ft Robot!* (3D action game)

**Game website:** <http://www.attackofthe50ftrobot.com/>

### **Graphics/Tools/Physics Programmer**

- Created graphics engine using DirectX 9.0c.
  - Post processing: Film grain, HDR, motion blur, depth of field.
  - Parallel split / cascading shadow mapping.
  - Deferred shading: Support for many lights simultaneously (directional, point, spot). Dynamic volumetric lights (point and spot lights).
  - Animation: Skinned hierarchical mesh animations. .X file parsing and custom animation controller.
  - Phong-Blinn lighting with bump, normal and specular mapping.
  - Particle effects using both software and hardware accelerated updating.
  - Video rendering.
  - Rendering optimizations: Level of detail algorithms for a large, open, and destructible world. View frustum culling for animations, models, shadows and lights.
  - Destructible environment: Real-time hierarchical mesh separation for destruction simulation of static and dynamic models. 3ds Max modeling solution for defining mesh hierarchies.
  - Anaglyph (3D glasses) image rendering as a post process.
- Worked with artists to integrate art assets, including static/dynamic models and skinned animations.
- Created level editor: 3D picking, controls for translating, rotating, and scaling 3D objects, Win32 API programming, undo history, object grouping, interface to edit game object data.
- PhysX integration: Rigid body simulation with 3ds Max model importing solution for custom compound bounding volumes.

**Sophomore Game:** *Romantic Cross* (2D platformer)

### **Technical Director and Core Engine/Tools Programmer**

- Component-based core engine architecture.
- Level editor: 2D tiles, collision data, file I/O.
- Technical Design Document.

**Freshman Game:** *ASCII Quest* (2D text-based ASCII graphic adventure game)

### **Graphics/Physics Programmer and Lead Tester**

- Built 2D graphics engine using ASCII characters: Scaling and rotations, particles, animation sets.
- 2D physics: collision detection and response.

### **Other Projects:**

- **Chess simulator:** C++ application, AI opponent using minimax algorithm with alpha-beta pruning, DirectX graphics, DirectInput user input.
- **Ray tracing renderer:** C# recursive ray tracing application using global and local illumination, Phong lighting, reflections, refractions, transmissions, soft shadows, torus primitive rendering.

### **Professional Experience**

#### **Software Engineering Intern – Monolith Productions**

May 2009 - Present

- Experience with refactoring and development of systems within a large code base.
- Game database porting to a new interface within the existing code base.
  - Refactoring within existing systems such as AI, Weapons, Characters, Sound, Input, etc.
  - Creation of new systems to provide a new interface to the game database.
- String library testing, cleanup and documentation.
- Added message handling functions for the messaging system as requested by content creators.
- Refactoring for a Vector Processing Library using platform specific intrinsic functions.
- Programming and debugging in a multi-platform environment (PC, PS3 and X360).
- Utilized and participated in the company's Engineering Process to facilitate and document the workflow involving significant additions and modifications of systems for the existing code base.
- WorldEdit visualization modifications to allow for custom rendering on a per object basis.  
Object rendering refactoring to utilize the new visualization interface.

### **Education**

- DigiPen Institute Of Technology  
BS in Real-Time Interactive Simulation (Computer Science)

July 2009