

Computer Science 3371 Computer Game Development I

Student Learning Outcomes

1. Students will create textured 2D models for use in games using XNA 4.0 2D modeling software.
2. Students will manipulate textures for use in 2D game development.
3. Students will apply an object-oriented class hierarchy to the design of their games.
4. Students will utilize different methods to add audio to their games.
5. Students will incorporate concepts of artificial intelligence in customizing the behavior of sprites within their games.
6. Students will create textured 3D models for use in games using XNA 4.0 3D modeling software.
7. Students will incorporate software demonstrating collision detection and response in an interactive 3D graphics application.
8. Students will create a complete interactive 3D graphics game.

Course Content

Textbook: *Learning XNA 4.0*, First Edition, by Aaron Reed. The following chapters are covered (See textbook "Contents").

1. **What's New in XNA 4.0?** The student is taken through the major changes in XNA 4.0 from previous versions.
2. **Getting Started.** The student is introduced to XNA 4.0, the tools needed to develop games in XNA, and the installation of XNA Game Studio 4.0.
3. **Fun With Sprites.** The student is introduced to 2D sprites, transparency, sort order, movement, framerates, sprite sheets, and animation.
4. **User Input and Collision Detection.** Covers user input from keyboards, mice, and Xbox 360 gamepads, as well as the implementation of collision detection.
5. **Applying Some Object-Oriented Design.** Discusses and implements game components and applies an object-oriented class hierarchy to the design of the 2D game.
6. **Sound Effects and Audio.** Introduces the Microsoft Cross-Platform Audio Creation Tool (XACT) and the new simplified audio API as methods to add sound to games on the PC, Xbox 360, and Windows Phone 7.
7. **Basic Artificial Intelligence.** Explains the nature of the science of artificial intelligence and introduces basic artificial intelligence concepts. This chapter also walks through creating customized derived classes with the class hierarchy to implement different behaviors for the game sprites.
8. **Putting It All Together.** Puts the finishing touches on the 2D game, including 2D text, scoring, adding different types of sprites, background images, game states, and power-ups.

- 9. 3D Game Development.** Discusses coordinate systems, cameras, and drawing primitive objects, as well as moving, rotating, and scaling objects in 3D space. Culling and texturing surfaces are also discussed.
- 10. 3D Models.** Introduces 3D models and discusses drawing, rotating, and moving 3D models in 3D space.
- 11. Creating a First-Person Camera.** Walks the student through the creation of a first-person vector-based camera in 3D, which implements forward and backward movement, strafing, and rotation in yaw, pitch, and roll.
- 12. 3D Collision Detection and Shooting.** Delves into the code behind shooting a moving enemy, creating a shot object, moving it in 3D space, and handling collision detection in 3D using bounding spheres. A 3D crosshair HUD (Heads Up Display) and audio effects are also added to the game.
- 13. HLSL Basics.** Introduces High Level Shader Language (HLSL) syntax and implementation, as well as the code required to use HLSL effects in XNA. A number of image manipulation effects are implemented using HLSL.
- 14. Particle Systems.** Walks the reader through the implementation of a custom vertex and a particle used to create an explosion particle effect.
- 15. Wrapping Up Your 3D Game.** Fine-tunes the 3D game, with sections covering splash screens, game states, scoring, and power-ups.
- 16. Deploying To The Xbox 360.** Walks the reader through connecting an Xbox 360 to a PC and deploying the Xbox 360. User input and screen resolution differences between the PC and the Xbox 360 are discussed.

Revised September 2012