

Farm Scene 3D Graphics Project

Project Overview

A 3D farm scene implementation using OpenGL with multiple lighting, collision detection, and interactive features.

Requirements Met

Multiple Object Instances: 3+ copies (trees, barrels, lamps) with different scales/positions

Object Rotation: Windmill, trees, and barrels rotate around axes

Hit Testing: AABB collision detection prevents wall/object penetration

Compilation Instructions

Prerequisites

Install required libraries for your system:

macOS (Homebrew)

```
brew install glfw assimp
```

Ubuntu/Debian

```
sudo apt update
sudo apt install libglfw3-dev libassimp-dev libglm-dev
```

Windows (MSYS2/MinGW)

```
pacman -S mingw-w64-x86_64-glfw mingw-w64-x86_64-assimp
```

Compilation Commands

macOS

```
g++ src/main.cpp src/glad.o \
-I/opt/homebrew/include -Iinclude -I./src \
-L/opt/homebrew/lib -lglfw -lassimp -lz -framework OpenGL \
-std=c++17 -o Farm
```

Linux

```
g++ src/main.cpp src/glad.o \
-Iinclude -I./src \
-lglfw -lassimp -lGL -ldl \
-std=c++17 -o Farm
```

Windows (MinGW)

```
g++ src/main.cpp src/glad.o \  
-Iinclude -I./src \  
-lglfw3 -lassimp -lopengl32 -lgdi32 \  
-std=c++17 -o Farm.exe
```

Project Structure

```
Farm_Project/  
├─ src/  
│   ├── main.cpp          # Main code  
│   ├── glad.o            # OpenGL loader  
│   ├── *.obj             # 3D Models  
│   └── *.jpg/*.png        # Textures  
├─ include/  
│   ├── stb_image.h  
│   └── glm/  
└─ Farm                   # Executable
```

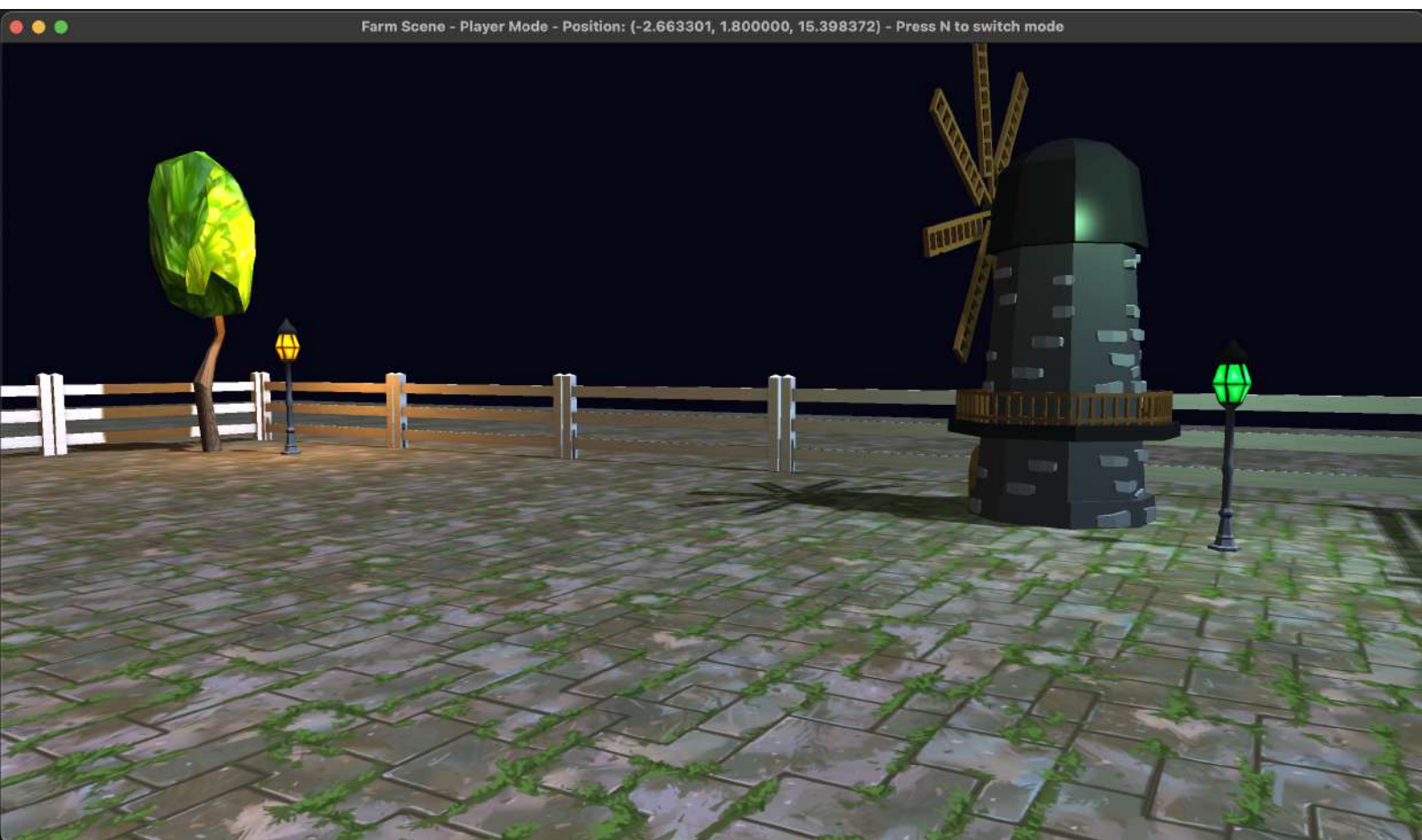
Running the Program

Execute

<code>./Farm</code>	<code># macOS/Linux</code>
<code>Farm.exe</code>	<code># Windows</code>

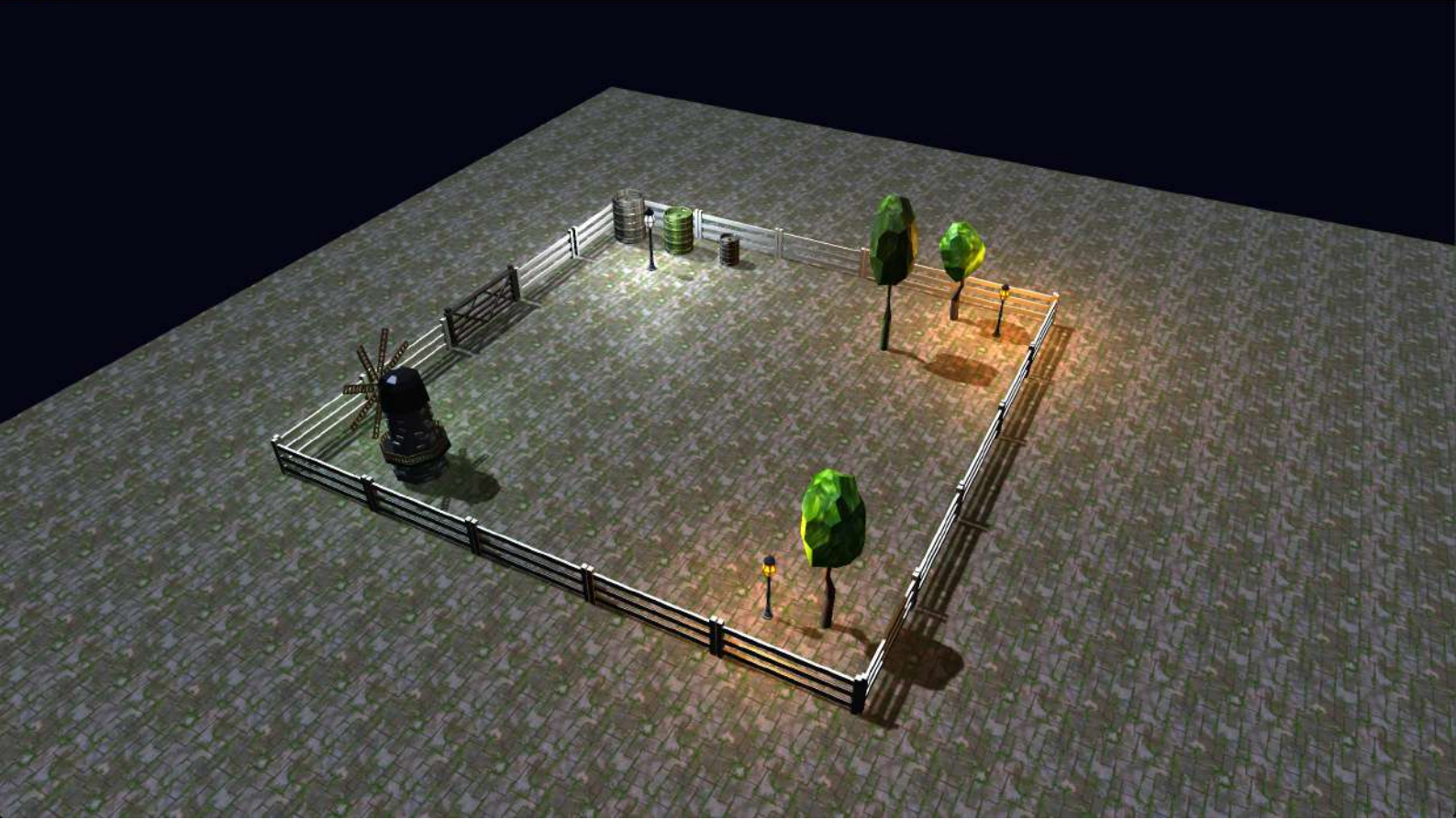
Controls

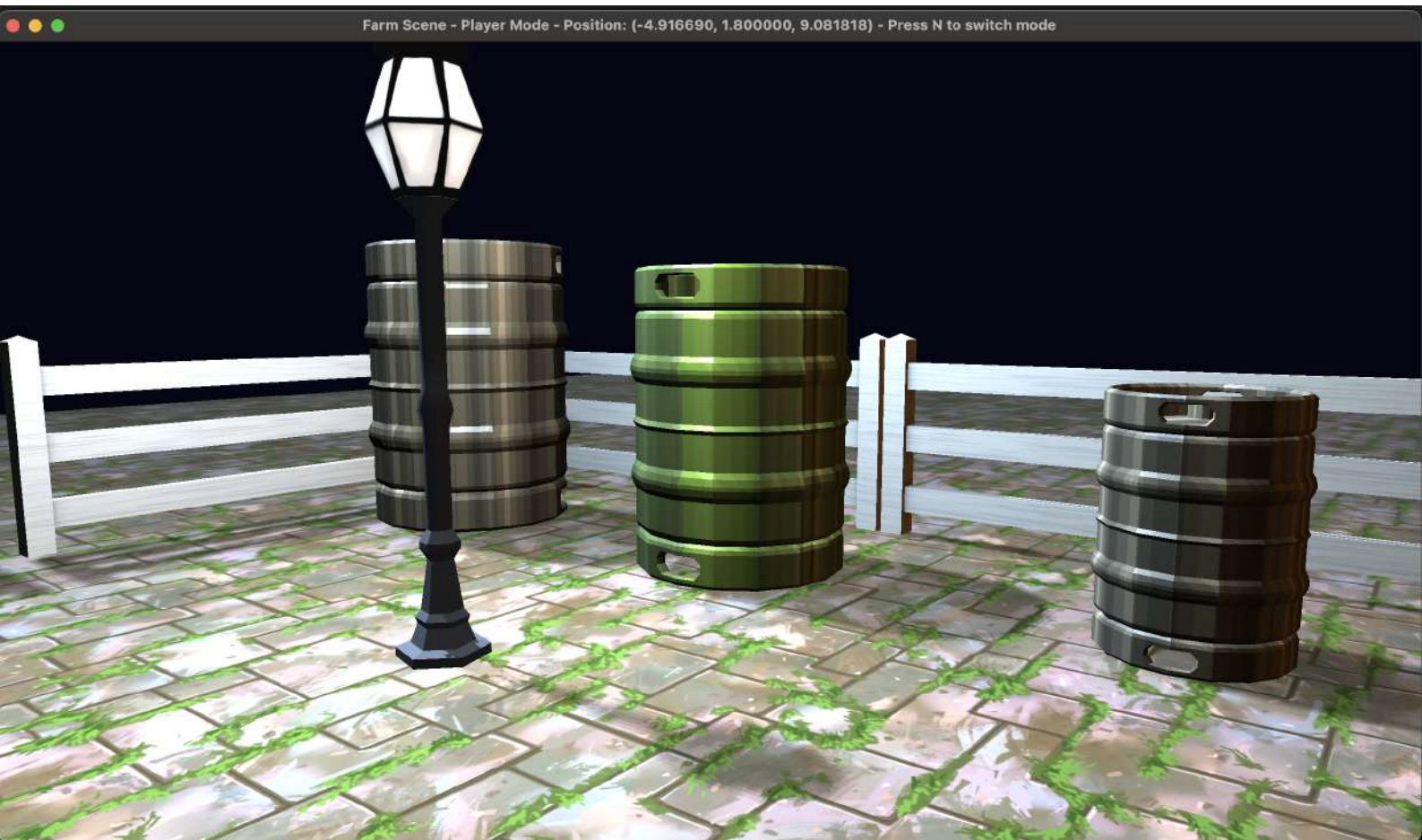
- **WASD**: Move (Player Mode)
 - **Mouse**: Look around
 - **N**: Switch Player/God Mode
 - **ESC**: Exit
-

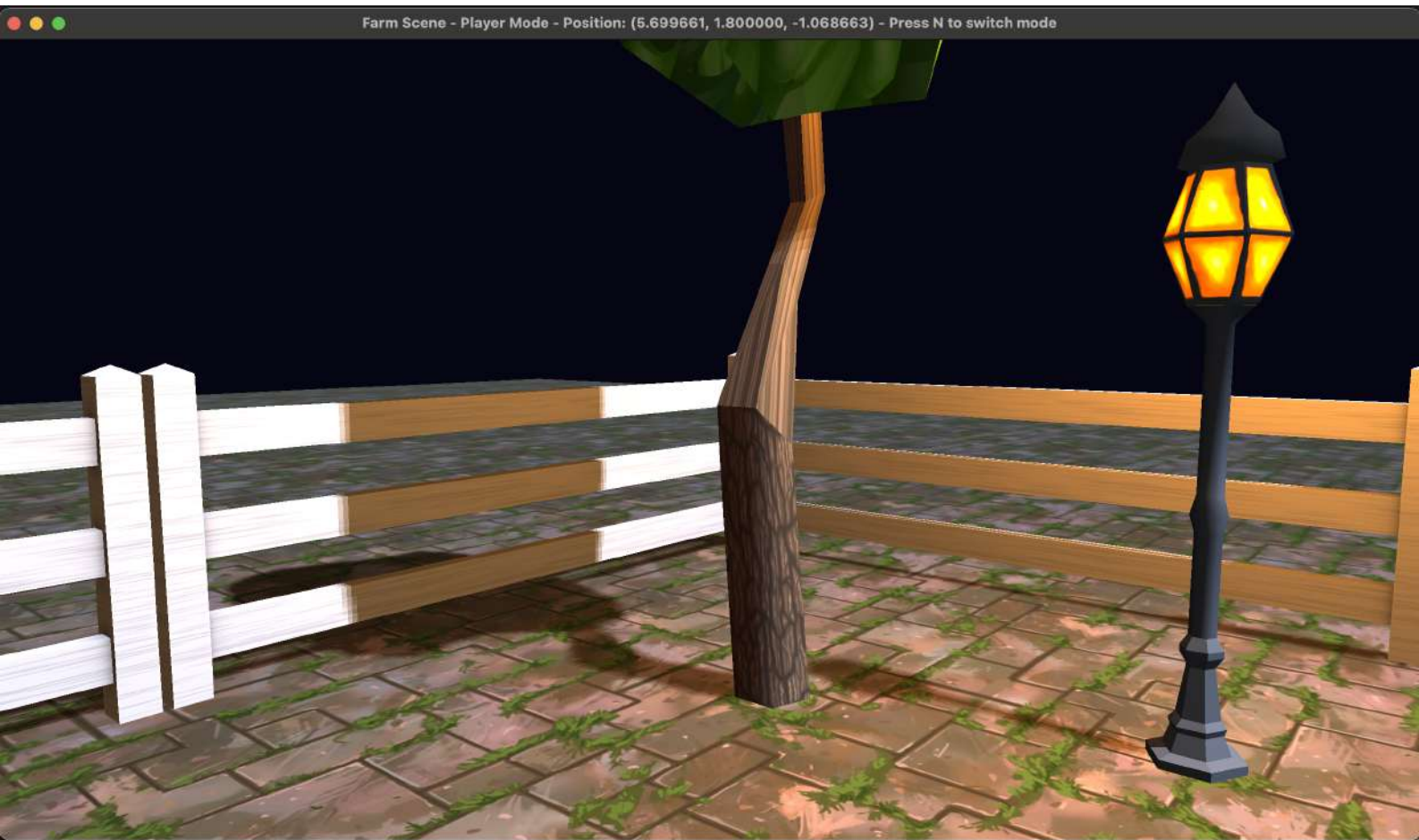


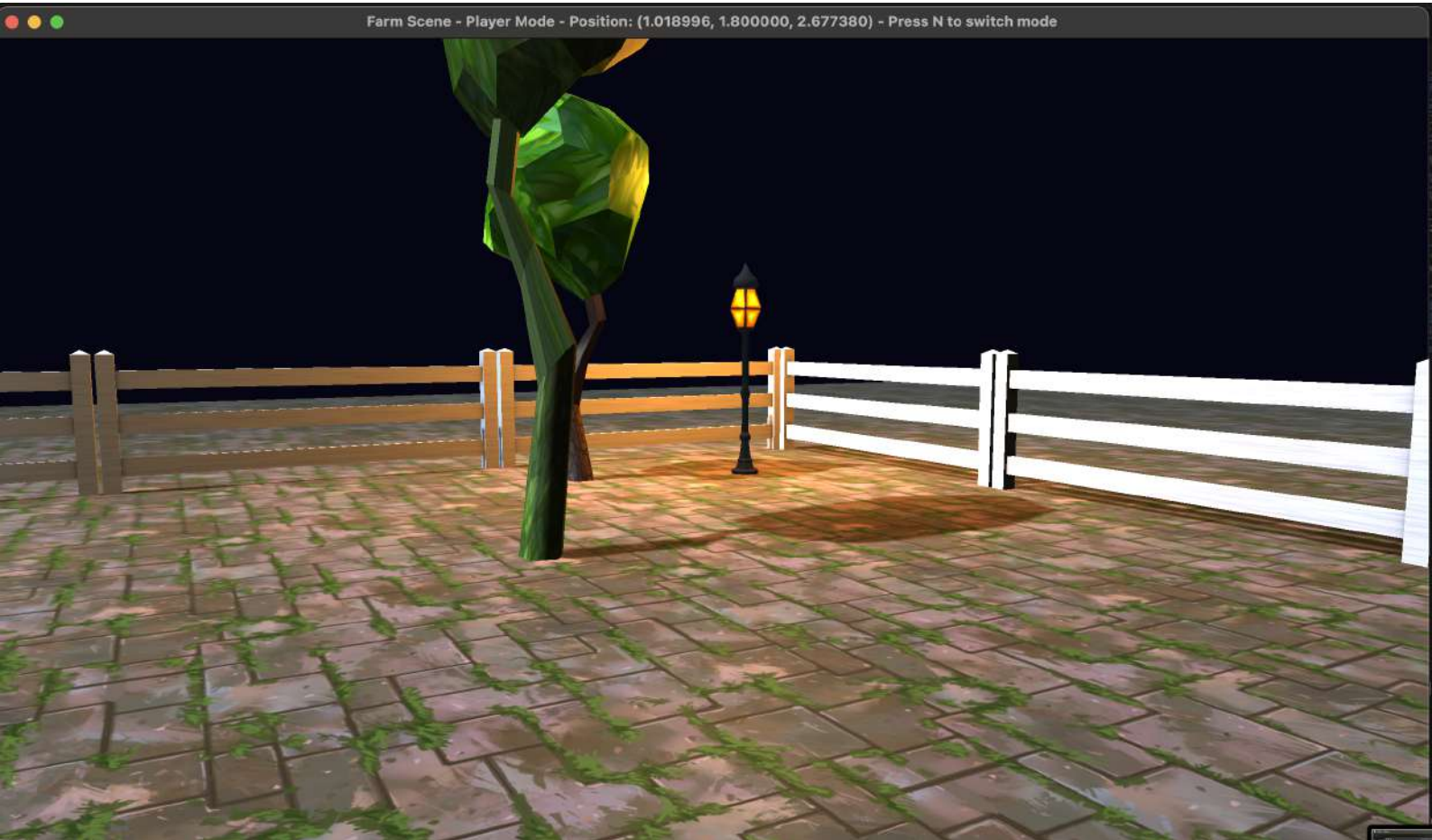


Farm Scene - God Mode - Position: (29.761717, 24.600155, -10.881403) - Press N to switch mode



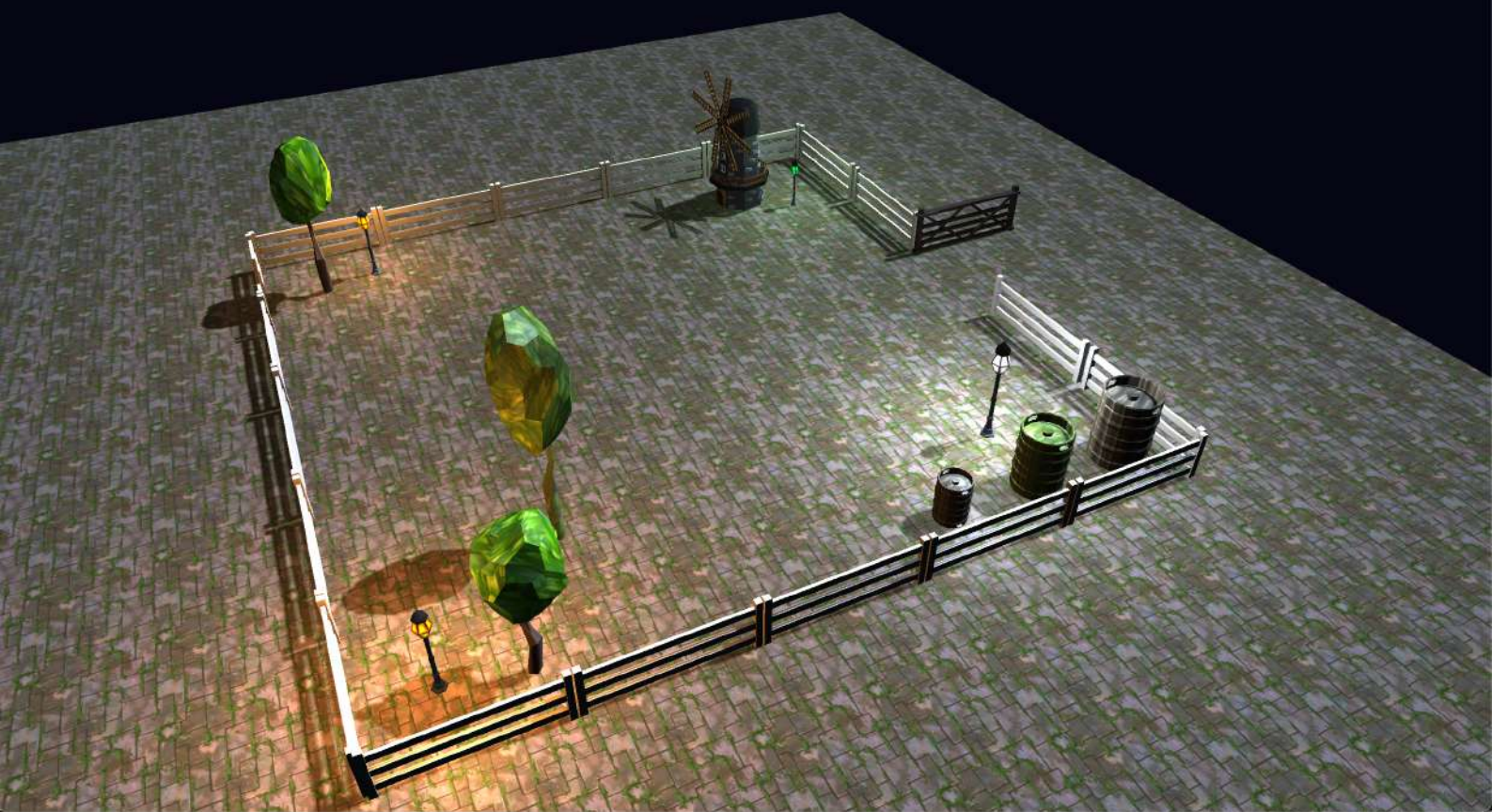








Farm Scene - God Mode - Position: (-26.395868, 20.062643, -3.964766) - Press N to switch mode



Technical Features

Graphics

- **Multiple Point Lights**
- **Shadow Mapping**
- **Phong Lighting** model
- **Texture Blending** effects

Collision System

- **AABB Detection** for all objects
- **Sliding Collision** along walls
- **Farm Boundaries** enforcement
- **Object Avoidance** (trees, barrels, lamps)

Scene Objects

- **Trees** (3x) - normal, blended, rotating
 - **Barrels** (3x) - different sizes, blended, rotating
 - **Lamps** (4x) - different colors
 - **Windmill** - rotating animation
 - **Fence System** with gate
-