# Real Time Graphics Programming Project Report

*Compagnoni Alessandro*

UNIMI A.A. 2023/2024


UNIVERSITÀ
DEGLI STUDI
DI MILANO

# Contents

# 1 Overview

This project implements a 3D environment using OpenGL, allowing users to navigate within 3 square rooms. Inside the rooms various 3D objects are placed, each rendered with a different material based on shaders that showcase various types of noise with different parameters, all freely adjustable by the user thanks to the dedicated UI.

# 2 Design Choices

## 2.1 Technologies

The following technologies were selected to ensure efficient rendering, cross-platform compatibility, and streamlined development for the project:

- OpenGL Version 3.3+:
  OpenGL's modern shader-based architecture was chosen to leverage direct control over vertex and fragment processing. By utilizing GLSL shaders, the pipeline enables complex procedural effects while maintaining high performance.
- GLFW (Graphics Library Framework):
  GLFW provides robust window management and input handling, ensuring consistent behavior across operating systems, also its event-driven architecture simplifies interaction with user inputs.
- GLM (OpenGL Mathematics):
  GLM mathematics library is optimized for graphics programming. It offers essential data types (e.g., vectors, matrices) and prebuilt functions for common transformations (e.g., translation, projection).
- ImGui:
  ImGui is a graphical user interface library for C++. It is simple, fast and portable and allows fast iteration and implementation of creation tools and visualization / debug tools.

## 2.2 Organization

The project is organized as follows:

- C++ file containing the code with the main function and all other functions called "Rooms.cpp", this file is located in the "src" directory.
- Directory called "shaders" containing all the fragment.glsl and vertex.glsl shader files for each of the materials used in the project.
- All the necessary dependencies and library located in the "include", "imgui" and "lib" directories.

## 2.3 Architecture

The scene is organized in 3 different square rooms, where each of the walls, floor and ceiling material's are based on a different shader: the walls' shader is made to make them look like wooden walls, while the ceiling and floor utilize a marble-like shader. The rooms are connected by 2 short corridors that utilize a dark-grey colored shader to appear neutral.

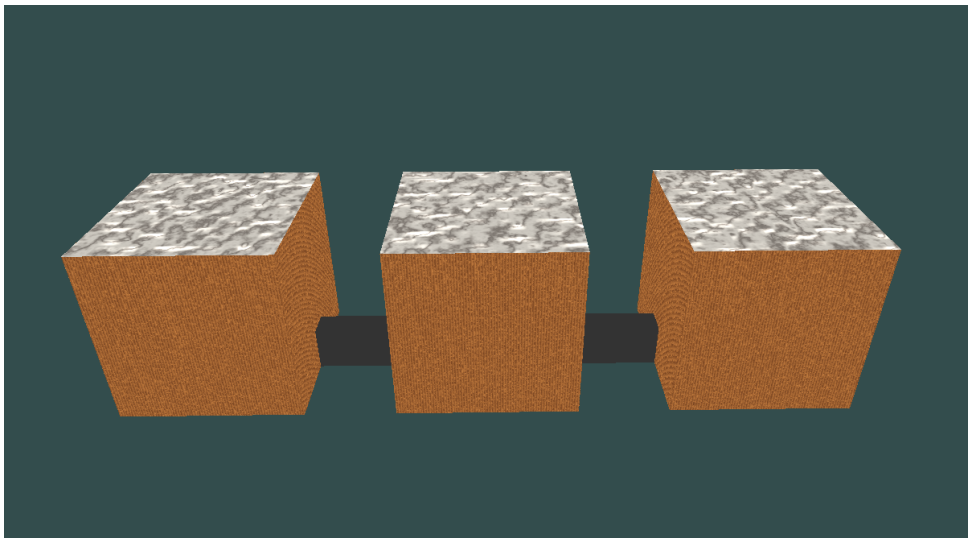

Figure 1: Structure and architecture of the 3D space as seen from the outside

Inside each of the first 2 rooms we can find 3 objects (a sphere, a pyramid and a cube), while in the last room we only find 2 objects of larger dimensions (a sphere and a cube).

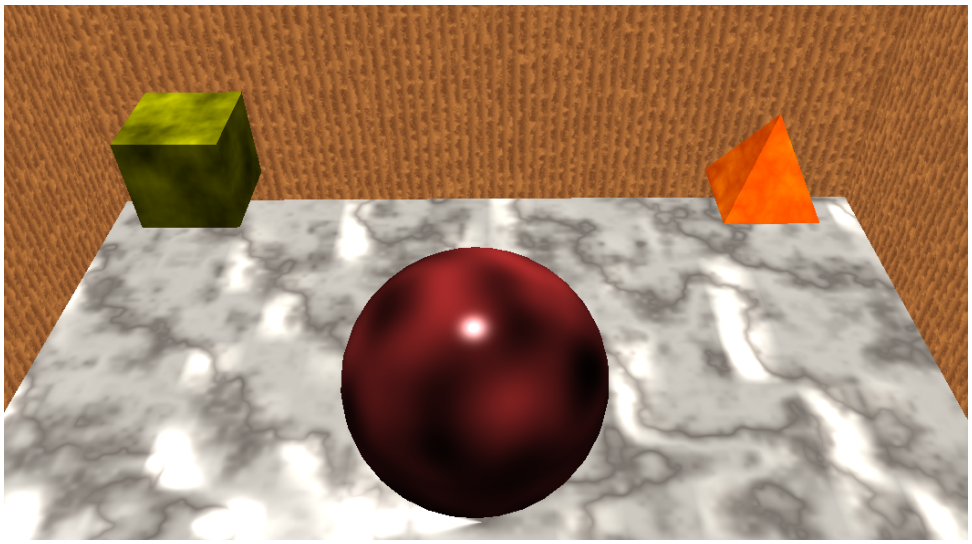

Figure 2: First room

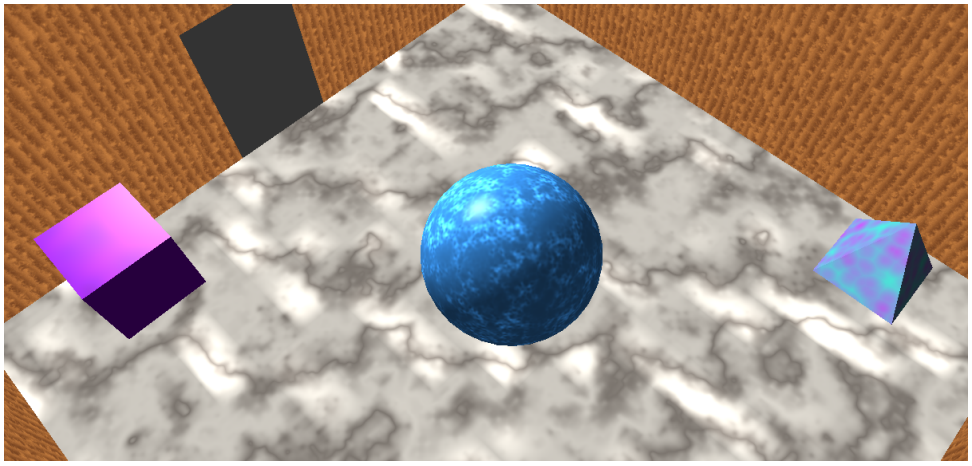

Figure 3: Second room

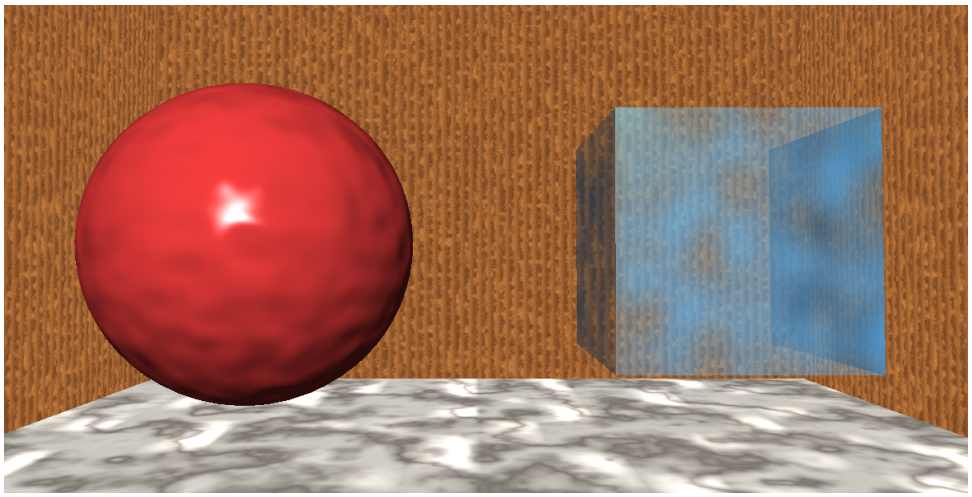

Figure 4: Third room

## 2.4 Controls

The user is able to move and look around in this space by using the WASD keys to move and the mouse to orient the camera.

# 3 Algorithms and Techniques

This section will explain in detail all types of noise utilized and where the user can find them in the project.

## 3.1 Noises

# 4 Implementation Details

# 5 Performance Evaluation