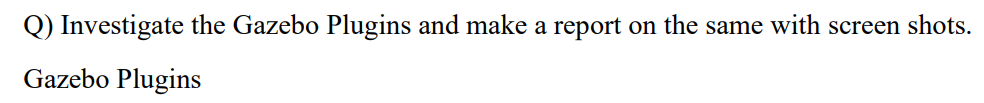
**Lab Sheet 5**

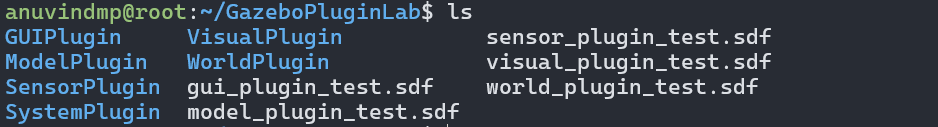
**Exploring Gazebo Plugins**

Name : Aparna Balaji

Roll : AM.EN.U4AIE22005



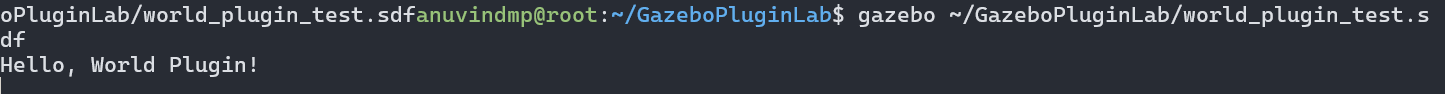
* The project directory, named **GazeboPluginLab**, contains everything except the GUI.
* **SDF files** and the **world file** for the world plugin are placed within the SDF project directory.
* Each plugin directory has its own **build directory** where the cmake command is executed to establish dependencies. This is also where .so files are generated after running cmake .. and make commands.
* The **CMakeLists.txt** file specifies all required dependencies, libraries, and settings needed for the plugin to function correctly.
* The **.cpp file** contains the actual code for the plugin.

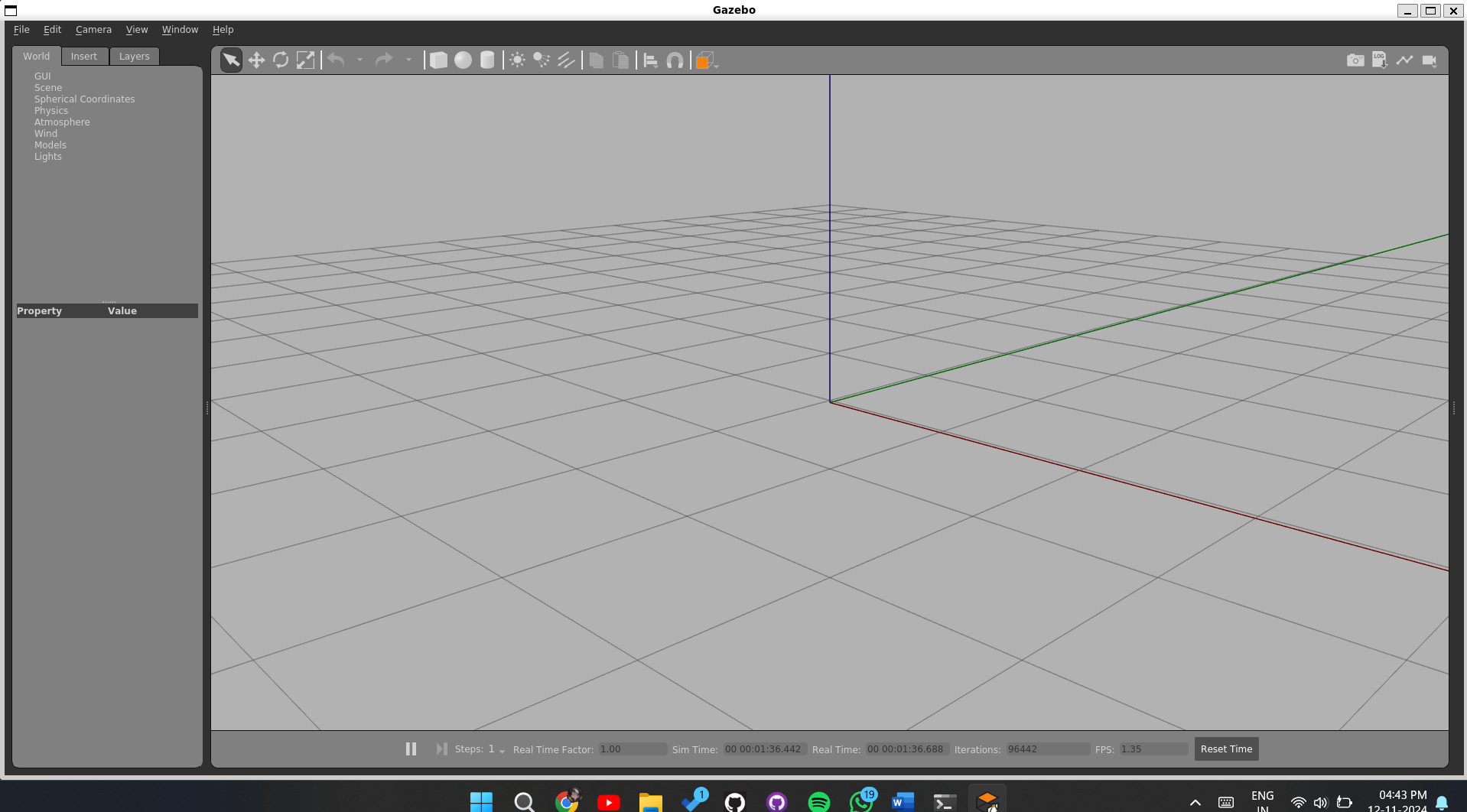
****

1. **World Plugin :**

* The **world plugin** is specifically designed to control elements within the simulation environment.
* It allows developers to add automation or scripted interactions directly within the Gazebo world.
* The **world configuration** and **elements**, such as objects, lighting, and layout, are defined using SDF files linked to the plugin.
* A successful load of the world plugin can be verified by a custom message in the terminal (e.g., "Hello, world Plugin!").
* The plugin’s functionality can include custom behaviors, event handling, and real-time modifications to the simulation.

**SCREENSHOTS :**

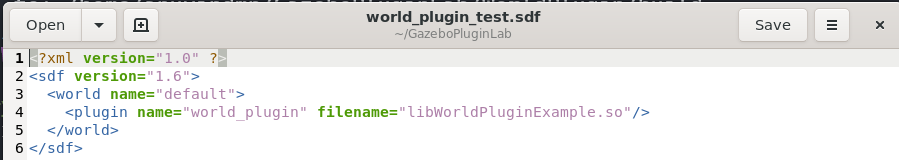




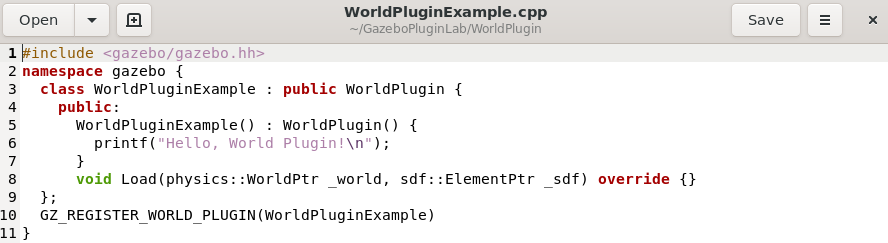
The world generated by the plugin is empty. In the terminal, the message “Hello, world Plugin!” appears, which was added in the C++ code to verify that the world file loaded correctly.

**Code :**

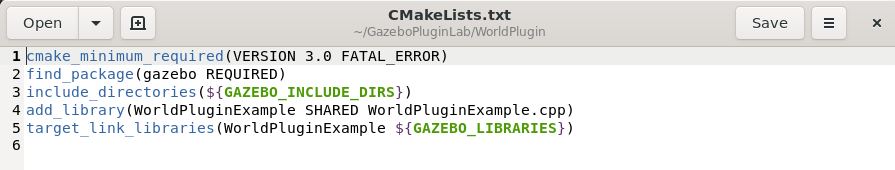
1. **world\_plugin\_test.sdf**

****

1. **WorldPluginExample.cpp**

****

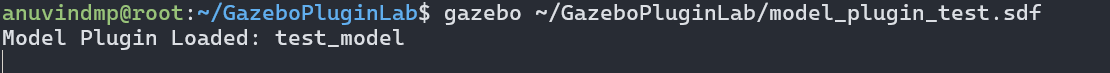
1. **CMakeLists.txt**

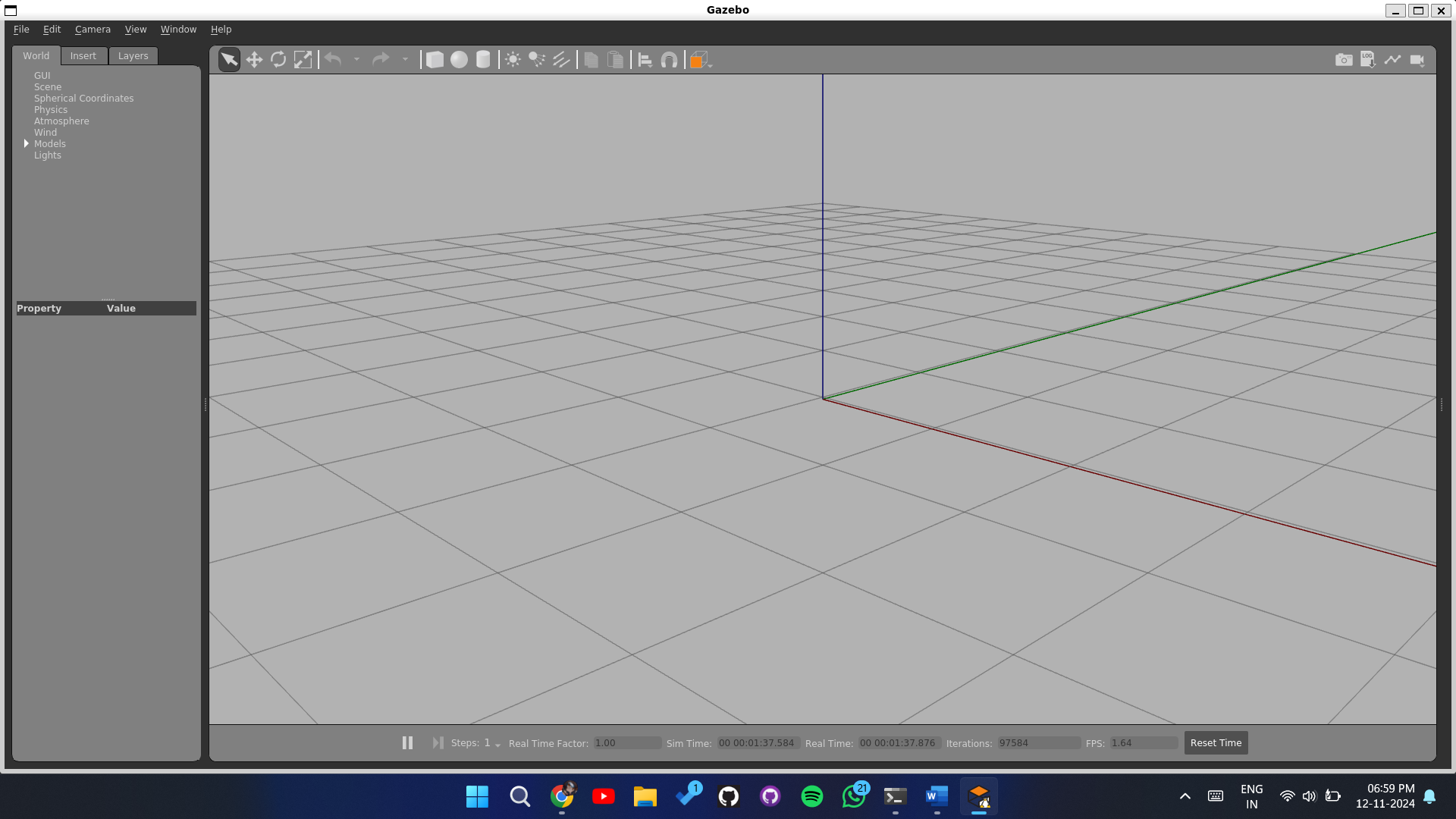
****

1. **Model Plugin**

* The **model plugin** is used to control and manipulate individual models within the Gazebo simulation.
* It provides a way to define specific behaviors for a model, such as motion, response to environmental changes, or interaction with other models.

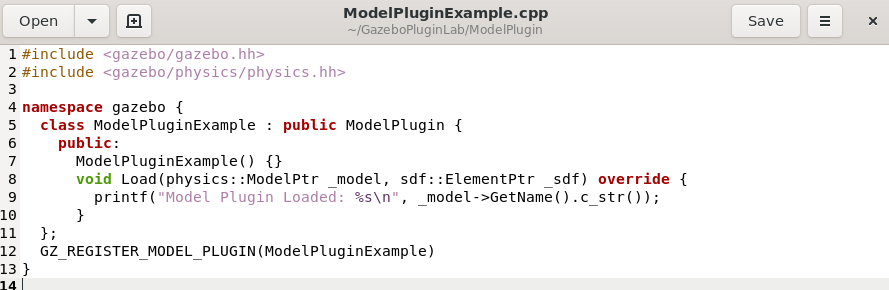
**SCREENSHOTS :**

****

****

**CODE :**

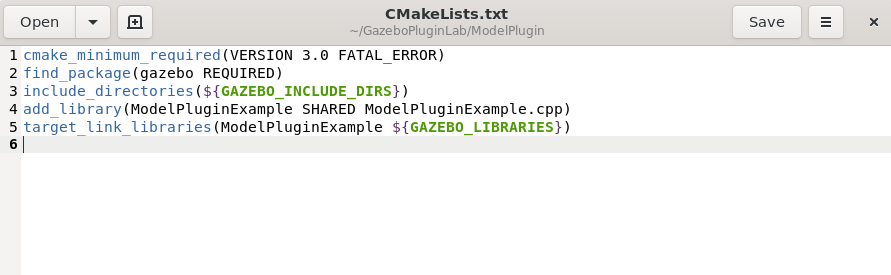
1. **ModelPluginExample.cpp**

****

1. **Model\_plugin\_test.sdf**

****

1. **CMakeLists.txt**

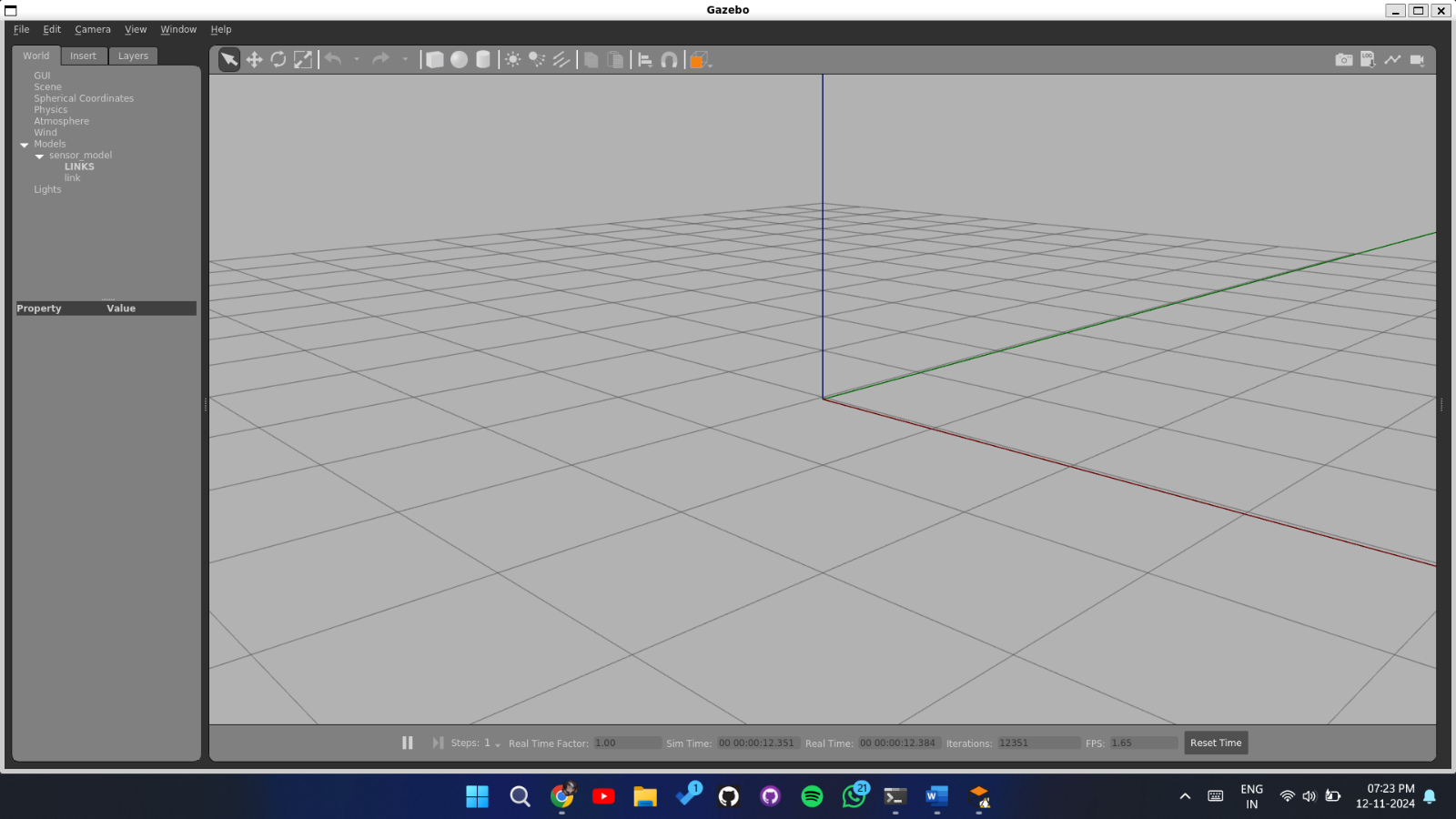
****

1. **Sensor Plugin**

* The **sensor plugin** is used to interface with and control sensors within the Gazebo simulation, such as cameras, lidar, or IMUs (Inertial Measurement Units).
* It allows for real-time data collection from sensors, enabling simulation of sensor readings and interactions with the environment.

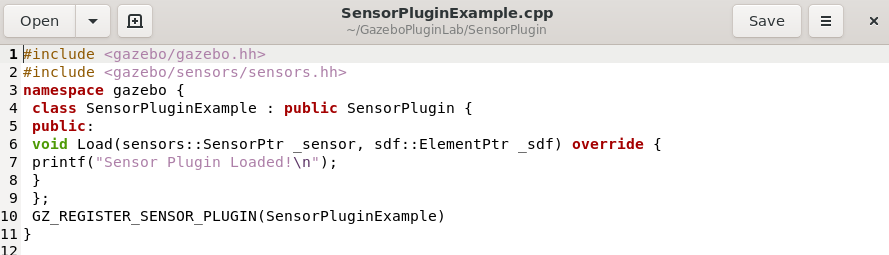
**SCREENSHOTS :**

****

****

**CODE :**

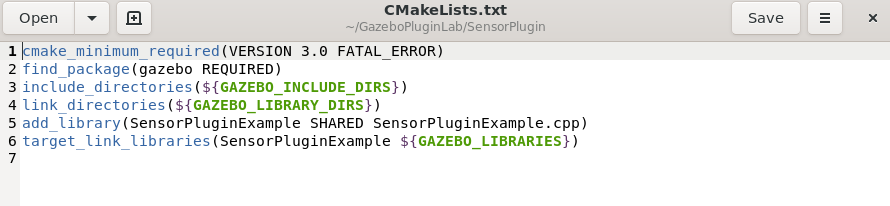
1. **SensorPluginExample.cpp**

****

1. **Sensor\_plugin\_test.sdf**

****

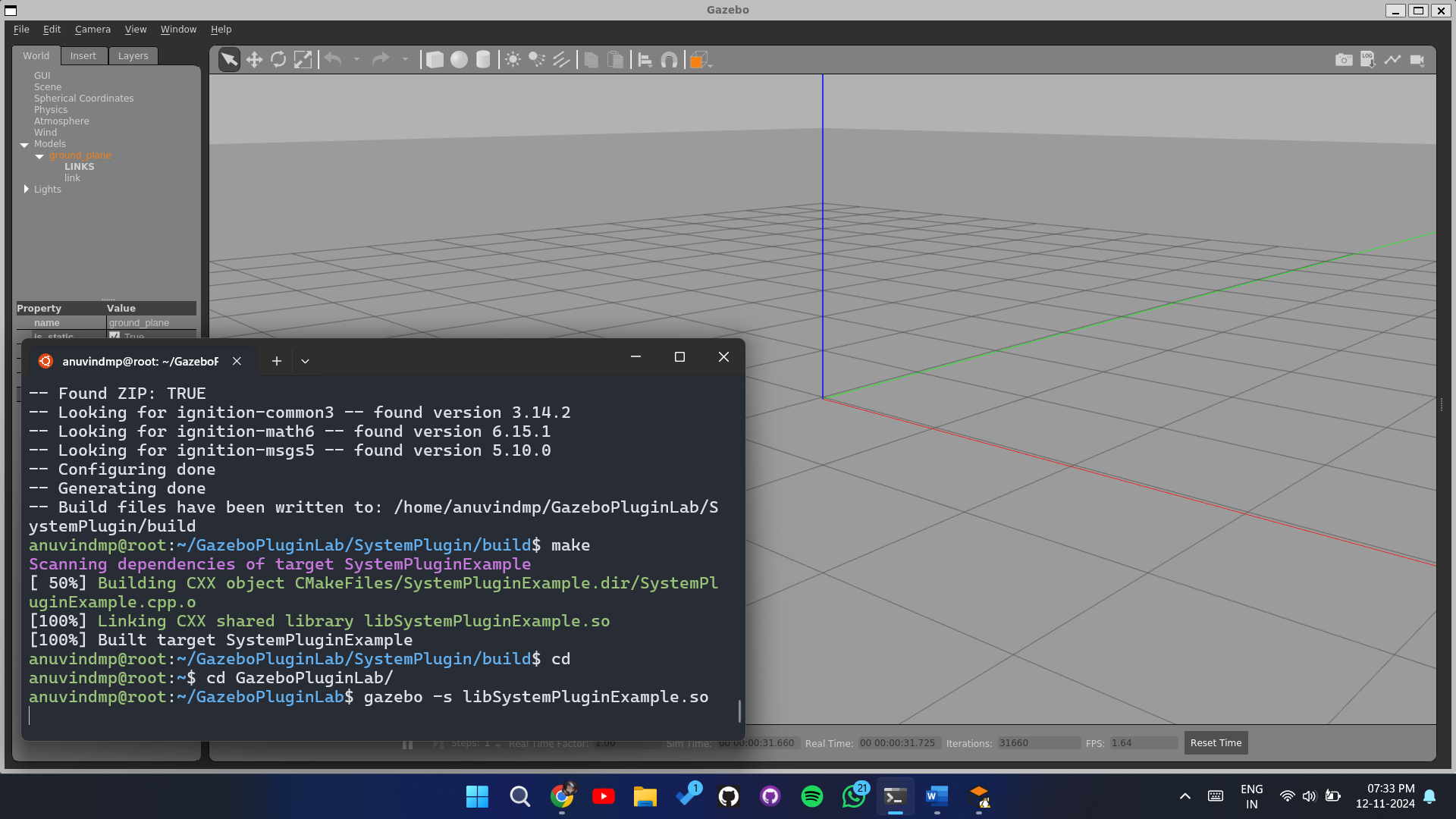
1. **CMakeLists.txt**

****

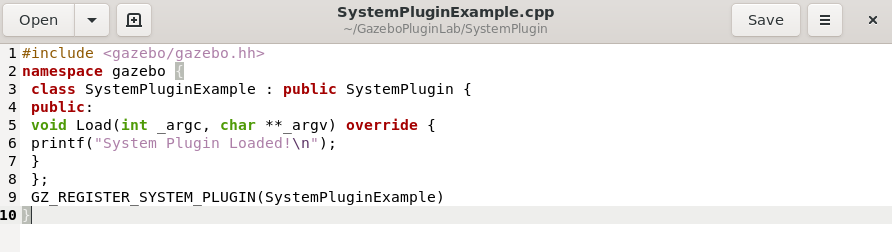
1. **System**

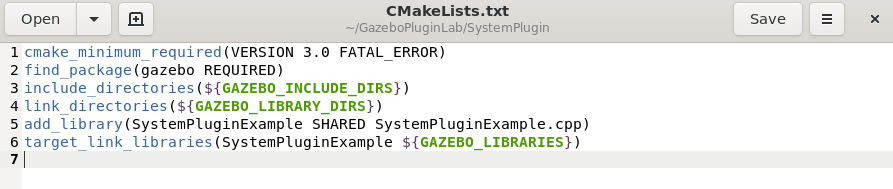
* The **system plugin** operates at the simulation level, allowing for control and customization of the entire Gazebo environment rather than individual models or sensors.
* It can be used to manage global settings, initialize multiple plugins, and handle overarching events or interactions across the simulation.
* Loaded at the start of the simulation, the system plugin provides a foundation for high-level control, ensuring seamless interaction and management of multiple components in the Gazebo world.

**SCREENSHOTS:**

****

**Code :**

****

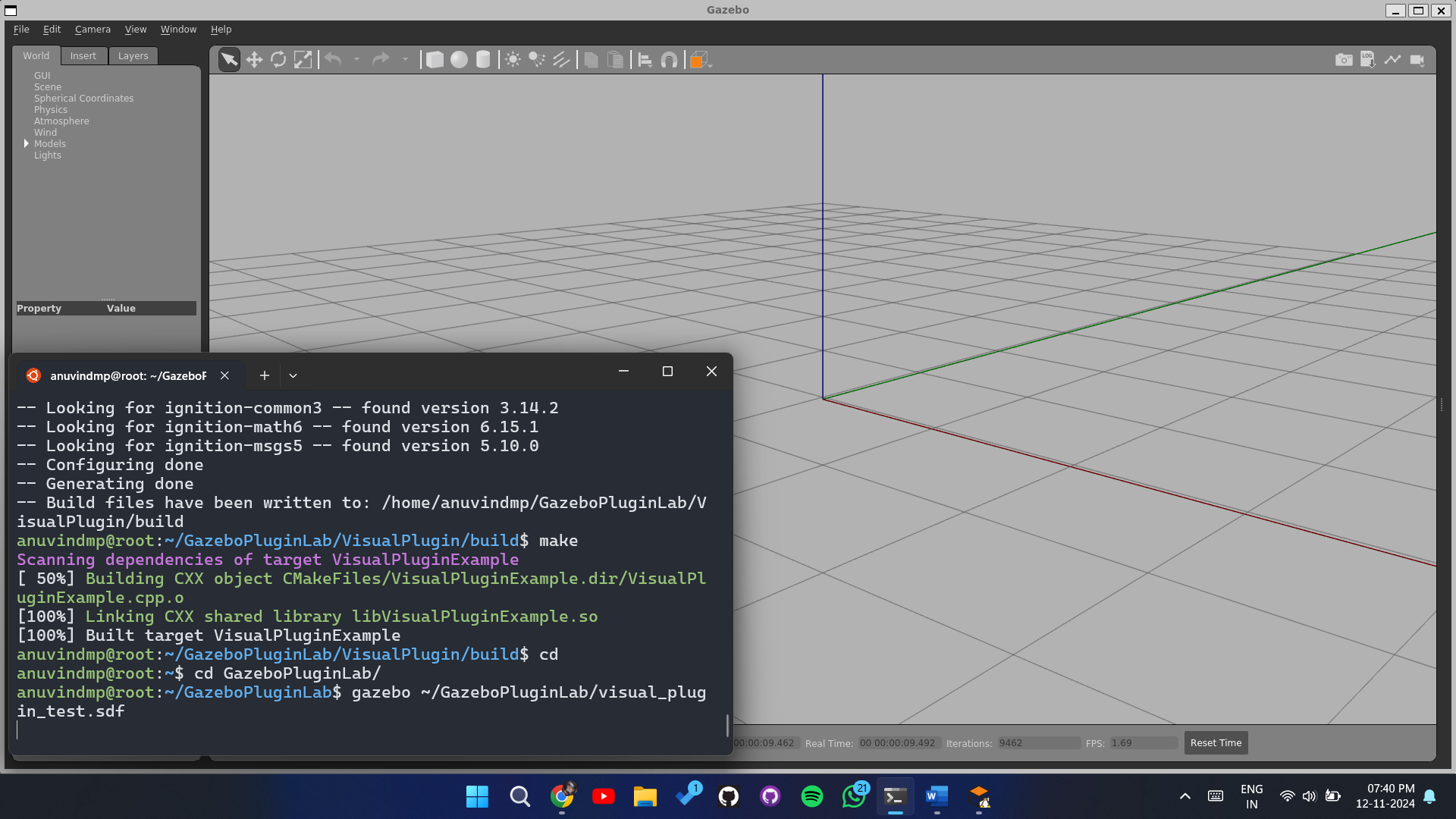
****

***.sdf are not required for this***

1. **Visual Plugin**

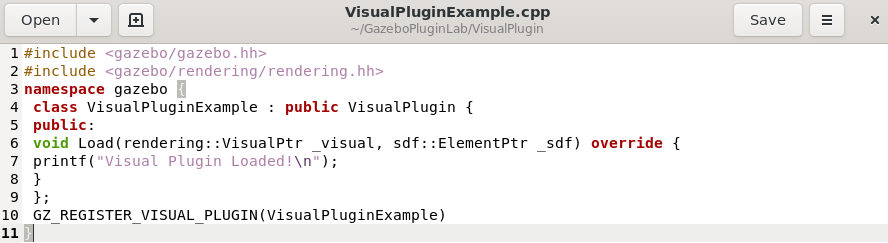
* The **visual plugin** is used to control and enhance the visual appearance of models and elements within the Gazebo simulation.
* It allows for modifications to rendering properties, such as **colors, textures, lighting effects,** and **animations,** providing more detailed and dynamic visuals.
* Once loaded, the visual plugin integrates with Gazebo's rendering engine, enhancing the visual elements and helping to create a more immersive simulation experience.

**SCREENSHOT :**

****

**CODE :**

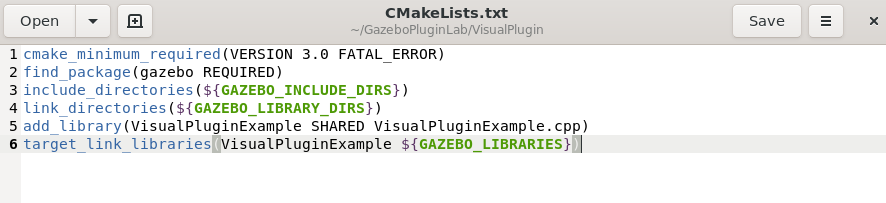
1. **VisualPluginExample.cpp**

****

1. **visual\_plugin\_test.sdf**

****

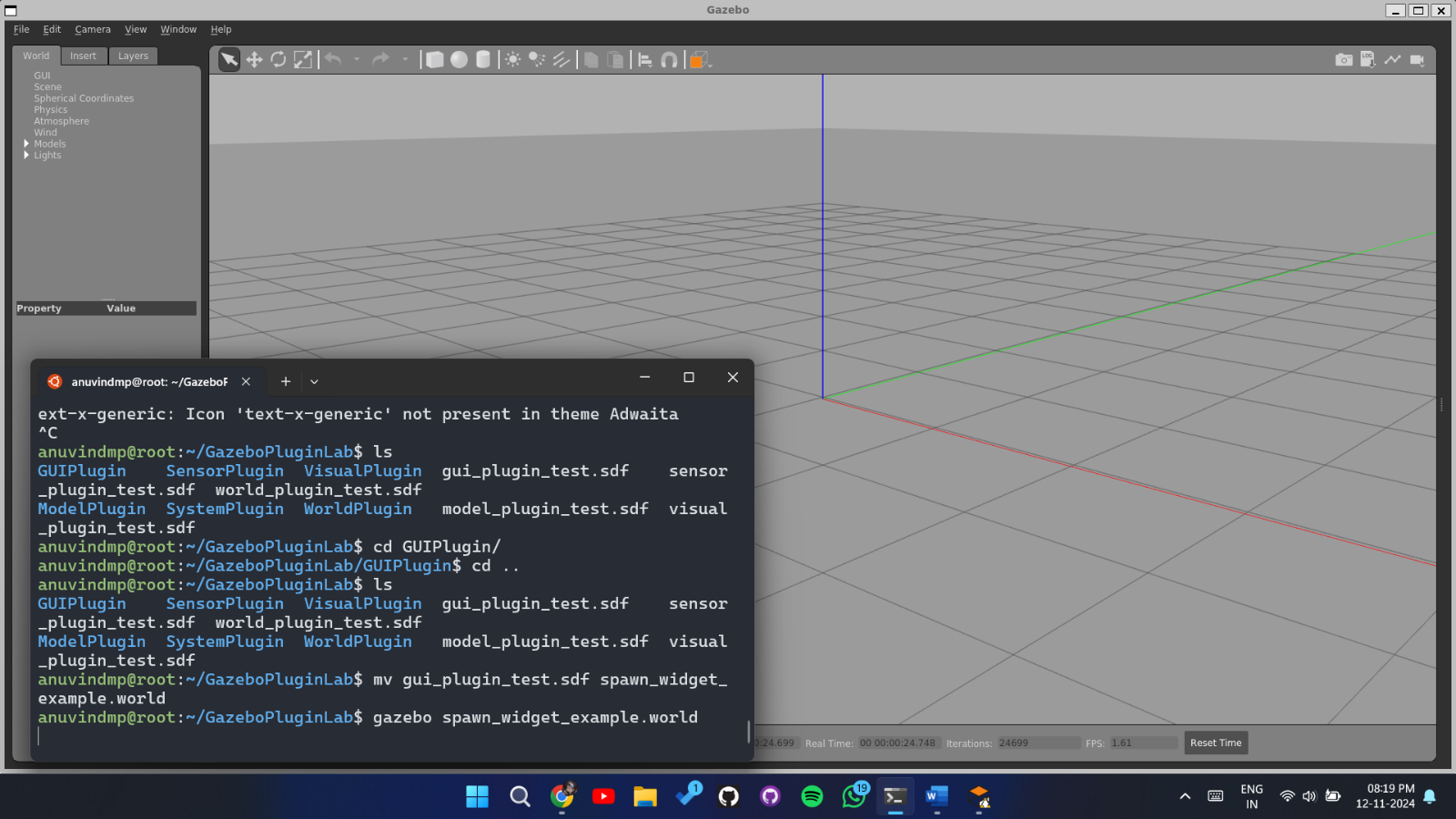
1. **CMakeLists.txt**

****

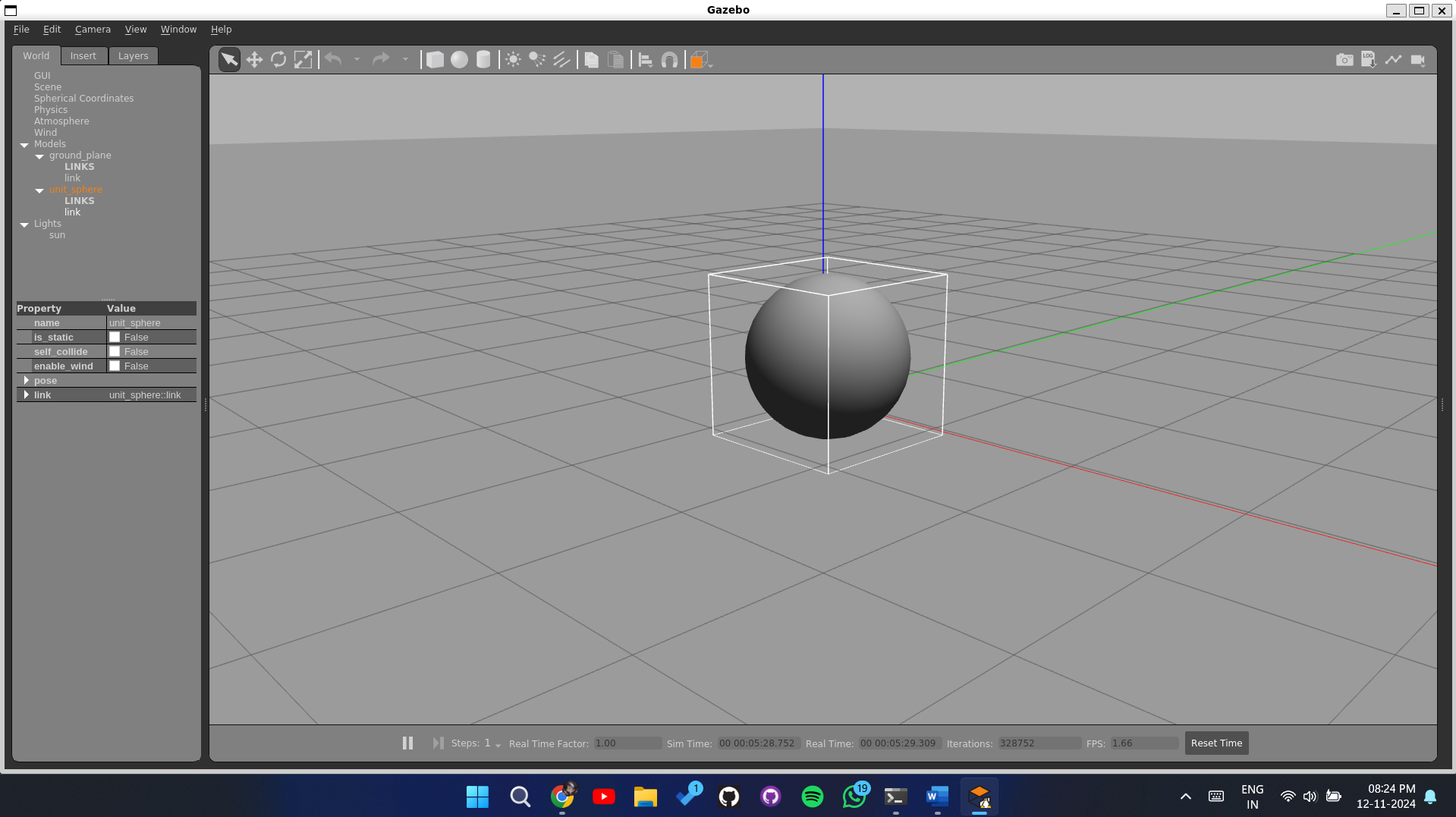
1. **GUI plugin**

* The **GUI plugin** is used to customize and extend the Gazebo user interface, allowing for added interactive controls, widgets, and visualization tools.
* It enables developers to create custom UI elements, such as **buttons, sliders, and panels**, that interact directly with the simulation and enhance user control.
* Loaded alongside the main GUI, the GUI plugin offers a powerful way to make the simulation more interactive and user-friendly, enabling custom functionality beyond the default Gazebo interface.

**Screenshots :**

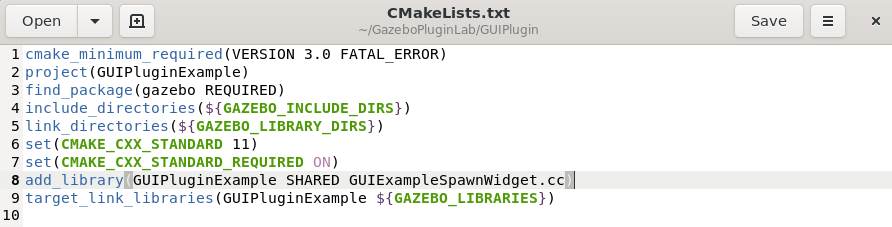
****

**Sphere :**

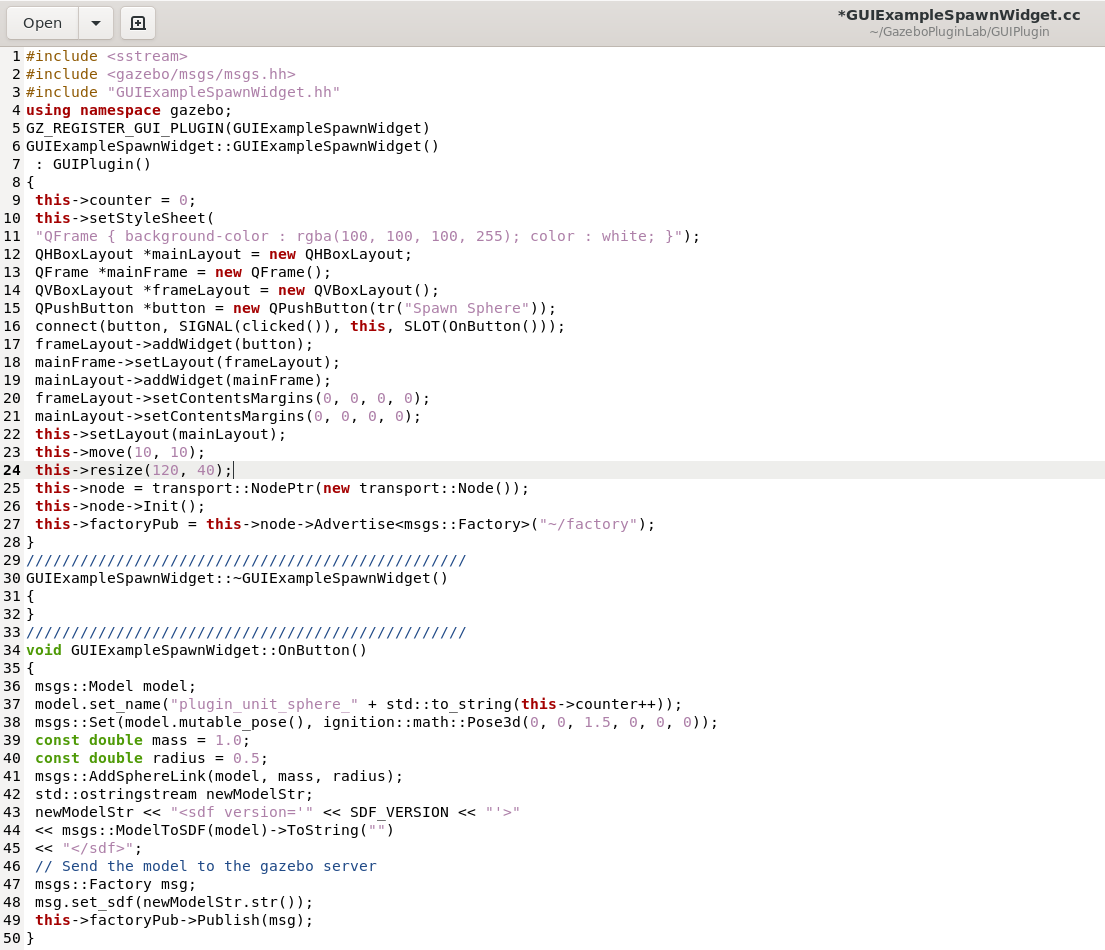
****

**Code :**



****

****

****