In the name of God

First activity:

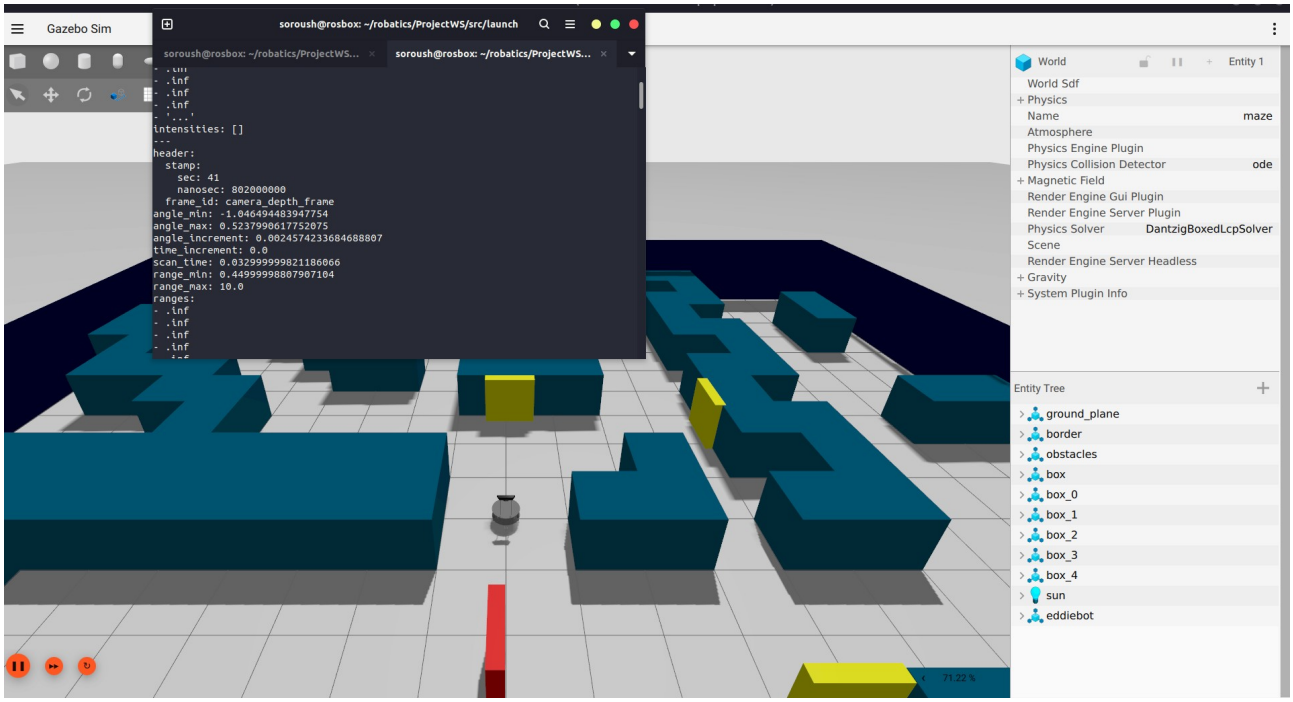
In Lee mode, VelocityControl is a plugin to control the angular and linear velocity, which sends these two items to the /cmd_vel topic. On the other hand, DiffDrive is a plugin that can define several right and left wheels for the robot and It can take the whole structure of the wheel and its main job is that by receiving the linear and angular velocity, it can give us the odometry of the robot. In which position, direction, and new speed the robot is located with a specific speed or acceleration. This plugin subscribes to the /cmd_vel topic and publishes the result in the /odom topic.

About how to use VelocityControl, we must first add it to the SDF file with a topic and initial angular and linear velocity, whose default topic is /model/<model_name>/cmd_vel, which we can change according to our needs. Now, for use when a Twist is sent to the relevant topic, the robot will move according to the new linear and angular speed. In the case of DiffDrive, we must first add it to the xacro file. This plugin has /cmd_vel subscription and in the topic publishes odom. In fact, it takes a Twist published in cmd_vel and by processing it sends the odometry of the robot in the /odom topic. We can use the command

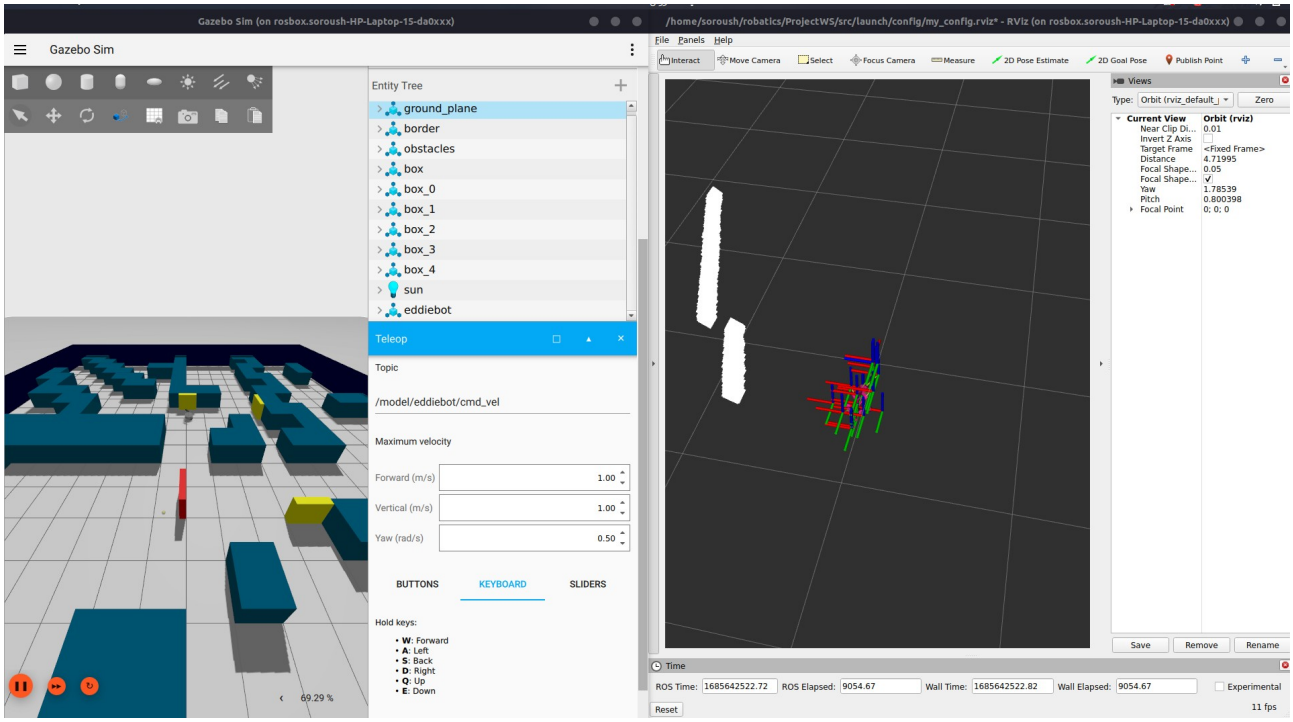
ros2 topic echo /odom Let's find out the contents of this topic.

Second activity:

The name of the header is camera_depth_frame. In the relevant package, an attempt is made to produce a laser scan using deep photos and show all the revelations of obstacles and non-obstacles. Therefore, for this, we need to receive depth photos and send them in two relevant topics, which are mentioned in the case of practice. For this, we need to have a frame and examine depth images in it. In this frame, we examine the surroundings in terms of depth, and a line from this photo is considered depth, and after analyzing a series of numbers that are the distance of the sensor from the wall and other obstacles, it is published in the /scan topic. These distances, which have come in the role of a laser scan, are declared in the camera_depth_frame. As in this frame, we obtained depth photos, and therefore these distances are also presented in this frame, and their connection is that these points are children of this frame.



Result:



Third activity:

You can see the desired tree in this folder. As you can see, base_footprint is selected as the reference of the robot and the rest of the nodes are all its children. base_link is its child, and all the physical parts of the robot are the bases, wheels, sensors, and batteries. Regarding battery_box, we have two other links that are its children, and each of these links is the parent of right_wheel_link and left_wheel_link. Another point is that plate_0_link is the father of the camera, we can guess that the camera frame is installed on plate_0_link. The camera itself is the father of rgb_camera. Also, the tree related to the world is Odom's father. If we want to transform the coordinates from one frame to another frame and express it in another frame, it is necessary to express each frame in relation to its father's frame, and the father's frame in relation to the grandfather's frame, etc., until we reach the root, and in this way we can For example, let's get two frames from two different branches to the root and compare them.