Ain Shams University, Faculty of Engineering

Mechatronics and Automation Program

MCT 431: Autonomous Systems

Milestone 1

Deadline: 16 April, 2023 @11:59 PM

1 Project Objective

The objective of the course project is to develop an autonomous system that is fully capable of operating in an unstructured environment. The project will be simulation based hence, **No** hardware will be needed. This Milestone will be mainly to setup our environment, simulations and packages that will be needed in the preceding milestones.

2 Requirement 1

2.1 Description

1. You are required to install the standalone version of gazebo and run it. After successfully installing it you should test and run gazebo to verify that the installation was successful.

2. You are required to install the gazebo_ros_pkgs which are responsible for the communication and connection of gazebo to ROS. After successfully installing the package you should test and run gazebo from the gazebo_ros package and check the topics and nodes available which are provided by gazebo.

2.2 Submission

For this requirement your going to submit screenshots showing:

1. The standalone gazebo running in the background and a screenshot of the command you wrote on the terminal to launch this version and another screenshot of a terminal showing the available nodes and topics.

2. Gazebo running in the background launched from the gazebo_ros package and a screenshot of the command you wrote on the terminal to run it and another screenshot of a terminal showing the available nodes and topics.

2.3 Hints & Tips

This first requirement is just to setup your simulators and environments. On the google drive folder you will find a PDF file in the **"Project"** folder showing you the necessary commands and steps to install both the standalone gazebo and the gazebo_ros package and also the commands to run them both. The recording of Lab 5 shows you the steps of launching both versions of gazebo and the difference in the outcome of the terminal when checking the nodes and topics.

 Note: If you already have these requirements installed then just run them and take the screenshots.

3 Requirement 2

3.1 Description

1. You are required to install the ready made **Husky** mobile robot package which will be used in the preceding milestones. After successfully installing it you should run the Husky robot in Gazebo to verify that the installation was done correctly.

3.2 Submission

For this requirement your going to submit screenshots showing:

1. Gazebo launched with the Husky mobile robot spawned at the origin.

2. list of topics available when launching Gazebo with the Husky mobile robot

3.3 Hints & Tips

The PDF "Lab 5 Exercises" contains the details and commands necessary for the installation and launching of the Husky mobile robot on Gazebo.

4 Requirement 3

4.1 Description

1. You are required to move the Husky mobile robot in gazebo by sending/publishing the matching message with the matching topic using python code.

4.2 Submission

1. You are required to send a video showing Gazebo running in the background with the Husky robot spawned in it. You are then required to open a new terminal and run your python file and the Husky robot should start moving.

2. You are required to also send at the end the package you created as a zip file.

4.3 Hints & Tips

These are steps to guide you through implementation:

1. Create a new package.

2. Create a python executable file.

3. Check which topic connected to the Husky mobile robot is responsible for moving it. **Hint**:"Similar to the turtlesim topic responsible for moving the turtle."

4. Check the type and details of the message sent through this topic using the commands of Lab 2.

5. In the python file you need to import this type of message and specify values for it, then send it through the suitable topic.

6. **Important:** Check the recording, exercise PDF, Code and slides of Lab 5 to guide you through your implementation. The example covered in Lab 5 is very similar to this requirement.

7. You are **Free** to choose any linear or angular velocity values the Husky moves with.

5 Requirement 4

5.1 Description

You are required to write and implement a service. This service will be written in the same python file that is used to move the Husky robot.

1. This service is responsible for changing/setting the angular velocity of the Husky mobile robot. The service should take as a request a msg of type **"float64"** and return a message of type **"String"**. The request represents the angular velocity we wish our Husky robot to rotate with, the string message is just a message that is printed saying "Angular Velocity Set".

5.2 Submission

You are required to send a video showing Gazebo running in the background with the Husky robot spawned in it. You are then required to open a new terminal and run your python file and the Husky robot should start moving. You are then required to open another terminal and call your service from the terminal and give it any angular velocity you want. Your Husky mobile robot should start turning.

5.3 Hints & Tips

1. Do not forget to create a srv file and modify the CMakelists and package.xml files

2. It is not required for you to create a second python file to call this service. You can simply call this service directly from the terminal using the command "rosservice call /Name_of_service".

3. When calling your service from the terminal you need to enter the request. To do this easily all you need to do after typing the command "rosservice call /Name_of_service" is press the TAB button 3 times. This will auto generate the request message for you and you just need to change the value.

6 Rules and Final submission

1. Cheating in any way is not accepted, any team that copies the Milestone from another team, both teams get a **Zero**.

2. A late submission will result in grade deduction.

3. You are requested to send the entire package you created as a zip file including any codes, folders and files you create inside the package.

4. You are required to send the screenshots you took for each requirement. The screenshots should have a clear view of the terminal (including the lines written), and the name of the computer must be clear and unique for each Team.

5. Send your zipped package, videos and screenshots to the following email: auto.systems.submissions@gmail.com. In the email subject write "Milestone 1 submission". In the body of the email mention your names, IDs and team number. It is required that only **One** person from the team submits the Milestone.