2° Assignment

Carmine Tommaso Recchiuto

Assignment

The package assignment_2_2022: https://github.com/CarmineD8/assignment_2_2022 provides an implementation of an action server that moves a robot in the environment by implementing the bug0 algorithm . You can launch the simulation with *roslaunch assignment_2_2002 assignment1.launch*

What should you do here?

- Create a new package, in which you will develop three nodes:
  - (a) A node that implements an action client, allowing the user to set a target (x, y) or to cancel it. The node also publishes the robot position and velocity as a custom message (x,y, vel_x, vel_z), by relying on the values published on the topic /odom. *Please consider that, if you cannot implement everything in the same node, you can also develop two different nodes, one implementing the user interface and one implementing the publisher of the custom message.*
  - (b) A service node that, when called, prints the number of goals reached and cancelled;
  - (c) A node that subscribes to the robot's position and velocity (using the custom message) and prints the distance of the robot from the target and the robot's average speed. Use a parameter to set how fast the node publishes the information.
- Also create a launch file to start the whole simulation. Set the value for the frequency with which node (c) publishes the information.

Assignment

Additional Requirements:

- ***Only for node (a):*** Create a flowchart of your code, or describe it in pseudocode (Pseudocode Examples (unf.edu))
- Add some comments to the code
- Use functions to avoid having a single block of code
- Publish the new package on your own repository. The flowchart (or the pseudocode) should be added to the ReadMe of the repository. (consider using Markdown syntax to write your readme: Basic Syntax | Markdown Guide)
- **Deadline: 09/01/2023**
- In case you are participating in the 2° exam session, the deadline is postponed to **26/01/2023**

Evaluation

- Code performance
- Code structure and clarity
- Respect of the requirements
- Organization of the repository (e.g., README in which you describe what the code does (possibly with flowchart or pseudocode), how to run the code, possible improvements, …)