Task: Autonomous Docking Using ArUco Markers in ROS2

Objective:
Create a ROS2-based solution for autonomously docking a robot using ArUco markers. This task is designed to assess your ability to navigate and utilize ROS2 documentation, as well as your proficiency in developing, testing, and deploying a basic robotic system.

Task Description:

- **Environment Setup:**
    - Set up a ROS2 workspace on your local machine.
    - Ensure that you have all the necessary tools and dependencies installed for the task.

- **Robot Simulation:**
    - Use or create a URDF file to simulate a differential drive robot in your preferred simulation environment (e.g., Gazebo or RViz).
    - Alternatively, if you have access to a physical robot, set up the necessary ROS2 nodes to interface with the robot.

- **ArUco Marker Detection**:
    - Implement a ROS2 node that can detect ArUco markers using a camera sensor mounted on the robot.
    - The node should publish the pose of the detected markers relative to the robot.

- **Docking Logic:**
    - Develop a ROS2 node that subscribes to the ArUco marker pose topic and computes the necessary control commands to dock the robot to a specific marker.
    - Ensure that the robot can approach the marker, align itself, and stop when it is within a predefined distance (e.g., 10 cm) from the marker.

- **Testing & Validation:**
    - Test the docking system in a simulated environment with multiple ArUco markers placed at different locations.
    - The robot should be able to identify the correct marker and dock precisely.
    - Document any challenges faced during the implementation and how you resolved them.

- **Documentation:**
    - Provide detailed documentation of your solution, including:
    - A brief overview of the ROS2 nodes you used or created.
    - Instructions on how to build and run the solution.
    - A description of the approach taken for marker detection and docking logic.
    - A discussion of potential improvements and extensions.

- **Submission Requirements:**
    - A Git repository link containing all the code and documentation.
    - A short video (2-3 minutes) demonstrating the robot successfully docking to the ArUco marker.
    - A summary report (1-2 pages) highlighting the key aspects of your solution and any insights gained during the task.

- **Evaluation Criteria:**
    - Ability to navigate and utilize ROS2 documentation effectively.
    - Quality and functionality of the ROS2 nodes and overall system.
    - Clarity and completeness of the documentation.
    - Innovation in the approach taken to solve the problem.

Bonus Points:

Implementing a dynamic docking system that can handle moving ArUco markers.
Integration with other ROS2 functionalities, such as obstacle avoidance during docking.

This task is intended to evaluate both your technical skills in ROS2 and your ability to learn and adapt by using available resources. We look forward to seeing your creative solutions!