# Marking Grid

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Feature | P | C | D | HD | Marks Awarded |
| **Robot Digital Twin (Lab 8)** | Robot digital twin in Gazebo  (forward, backward, rotate  left, rotate right) | Implement different robot speeds, use this to modify the movement of the digital twin | Display walls detected by  ultrasonic sensors | Use of the robot compass to correct  rotation | /10 |
| **Drone Digital Twin** | Drone digital twin in Gazebo  (forward, backward, rotate left, rotate right, height) | Use speed from the robot to modify the movement of  the digital twin | Add the SDK curve function to the drone control and use this for the location/orientation of the digital twin | Add the SDK flip function to the drone control. Use an appropriate  image in gazebo and show it flipping in real-time. | /10 |
| **Robot or**  **Drone GUI** | Create a simple  GUI for controlling the  robot or drone | Display sensor readings from the robot or drone e.g. speed, battery, objects. | Add controls to switch the robot or drone from manual to automatic. | GUI should be rich and fully functional (e.g. settings for  speed, height) | /10 |
| **Robot assist the drone** | Display both the robot and drone in gazebo. Move both independently. | Move the drone and robot together so the drone is always above the robot. | Prevent the drone moving forward if an obstacle is detected. | Use all 3  ultrasonic sensors to keep the drone safe. Display objects in gazebo. Movement  should be fluid. | /10 |
| **Use of DJI**  **Tello Mission pads** | Add support for DJI Tello mission pad detection by using state from port 8890 | Implement  distinct actions when a mission pad is detected – e.g. move forward when pad 1 is detected | Use multiple mission pads to determine detect drone location and display it in  Gazebo | Put a mission pad on the robot and have the drone follow the robot  movement,  displaying it in Gazebo. | /10 |
| **Drone video** | Retrieve video | Retrieve video | Implement simple | Use video | /10 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **support** | | from the drone and make it available as a pub/sub topic. Display in a client program. | from the drone and make it available as a service.  Display in a client program. | object detection using OpenCV. Show simple  objects in Gazebo | object  detection to control the  drone or robot |  |
| **Collaborative robotics**  **world (2**  **students)** | | Connect 2 ROS2 systems together so messages can be sent between computers. Demonstrate control and data between the systems. | Display robots or drones from both systems in Gazebo | Use data from one system to control the robot from the other | Implement it in such a way so others can join the world.  Provide instructions and an API | /10 |
| **Robot**  **Obstacle**  **Avoidance** | | Implement a  simple subsumption  architecture which stops the robot moving forward when an object is detected | Add a  subsumption  layer for backing away and trying to navigate around objects | Add a subsumption layer for a search pattern | Visualise the 4 layers of the architecture  with a simple  GUI –  indicating which layer is in control. | /10 |
| **ROS2 robot**  **direct control** | | Use a Raspberry Pi instead of the Arduino to directly control the robot.  Implement all functionality in a  ROS2 node without using a bridge, | | Add support for the ultrasonic sensors | Add support for the robot advanced sensors (compass, gyroscope) | /10 |
| **Mario Kart**  **(2 students)** | | Connect 2 ROS2 systems together so messages can be sent between computers. Demonstrate | Display the location of both robots on a simple GUI or Gazebo. | Add a simple track and race progress. Mario kart works by using waypoints.  Visualise progress. | Implement a simple Mario Kart race. Both students should be able to see progress (no  video necessary) | /10 |
| **Maze solving robot** | | Build a simple maze and implement a ROS2 maze solver | Visualise the maze solving in real-time in Gazebo or in a  GUI | Record a manual solution as the robot moves and enable a ‘replay’ for fast maze  solving | Implement a solution to maze solving that is fast like the maze solving competitions. | /10 |
| **Write a guide** | | Write a guide for | Write a guide | Write a guide in | Write a guide | /10 |
|  |  | any ROS2  functionality | for something that has not been covered in the unit – include code. | the unit template format, including extension exercises and additional resources. | for something that could be used next year for the unit. Chat to the unit chair for  something appropriate. |  |
| Create own 1 | your |  |  |  |  | /10 |
| Create  own 2 | your |  |  |  |  | /10 |
| Create  own 1 | your |  |  |  |  | /10 |
| Overall |  |  |  |  |  | /30 |