ROS

NJRB2 | MVS9 | FA296

# Problem Description

We need a way to communicate with the drone. ROS provides us with convenient message queues that we can subscribe and publish to. This part of the project involves making decisions based on the results produced by the other parts of the project. An example of the workflow for this part of the project would be: Read an image from the drone’s camera, convert the image into an interpretable format and pass the image onto the image analysis modules, use the result returned to make a decision, send a command to the drone based on the decision. This part of the project is also responsible for issuing take-off and land commands alongside the emergency landing command.

# Use cases

These use cases help to identify the different uses for the ROS modules in the project:

* As the ROS component I need to be able to send take-off and land commands to the drone.
* As the ROS component I need to be able to send movement commands to the drone.
* As the ROS component I need to be able to receive data from the drone.
* As the ROS component I need to be able to pass information from the drone to the other project modules.
* As the ROS component I need to be able to make decisions based on the data returned to me from the other project modules.
* As the ROS component I need to monitor key variables such as drone height and rotation.
* As the ROS component I need to ensure that drift is kept to a minimum.

# Milestones

These milestones help to identify different work items during sprint planning meetings:

* Run turtlebot simulations.
* Run drone simulations.
* Drone take-off and land in simulation.
* Control drone via keyboard.
* Read odometry from real drone.
* Drone take-off, fly forward and land in real life.
* Drone hover over Roomba.
* Drone no longer drifts vertically or rotationally.

# Work Done

This section documents the dates on which work was completed. The information is published to Trello during development for ease of access.