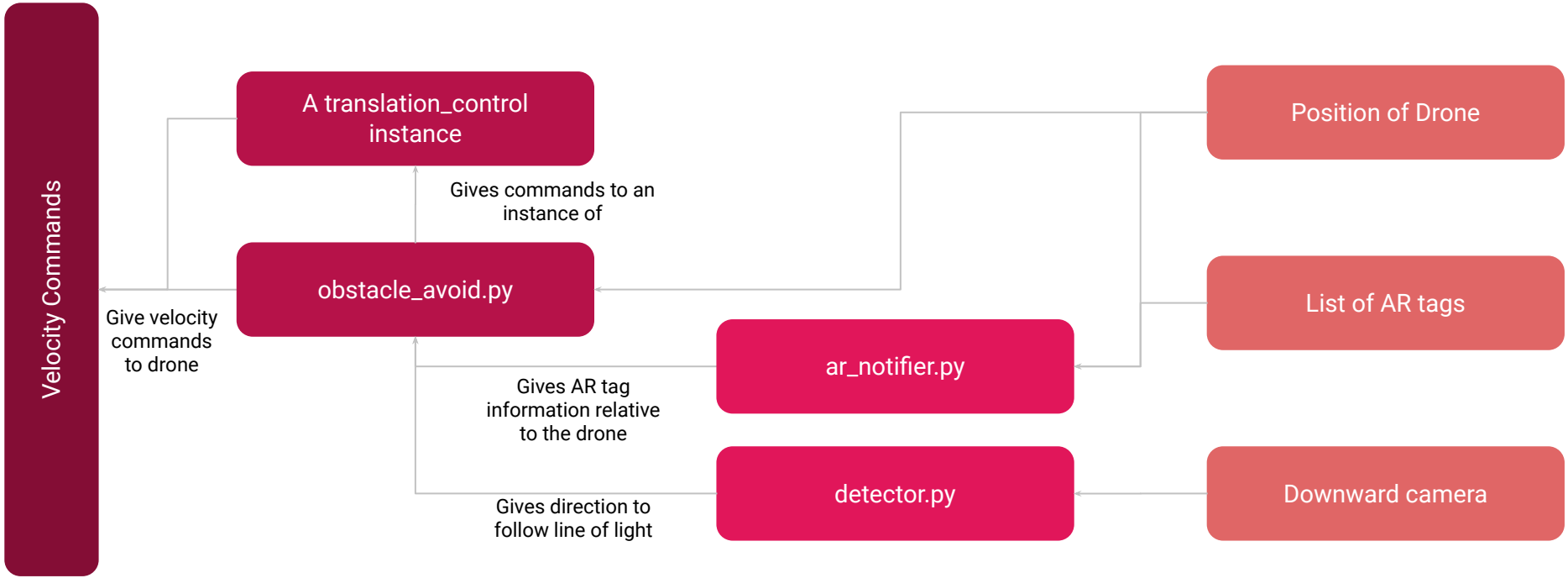
Team Nomad- Code Explained

Rumaisa, Rashi, Manik, Justin

Overview


Velocity Commands
A translation_control instance
Gives commands to an instance of
obstacle_avoid.py
Give velocity commands to drone
Position of Drone
List of AR tags
ar_notifier.py
Gives AR tag information relative to the drone
detector.py
Gives direction to follow line of light
Downward camera

ar_notifier

Detects closest AR tag, gets its x,y,z pos relative to drone

- Subscribes to /ar_pose_marker which gives the list of all the observed AR tags and 3D pos relative to drone
- Subscribes to /mavros/local_position/pose which gives the 3D pos of drone with respect to ground
- Publishes a Position_Holder (we defined as marker.pose.pose.position.x, y, z) to /closest_ar_dists

Challenges:

- When we were creating our custom msg, we had to make sure that we imported it in our code, added it to the CMakeLists, ran catkin make
- Realized we had to add our nodes to launch file

obstacle_avoid

- Initially we tried to implement our obstacle avoid code over our closed loop line follower code
- But we had the code transition to open loop once it read the AR Tag because that method was easier
- We initialized delta_z to zero and then once the drone was <= 1 meter away from the ar tag and 0.75 m above the ground the drone read the AR Tag and either decide to go up or down and past the obstacle
- If the drone's height relative to the center of the AR Tag was lower, then the drone would go below the tag and vice versa

Challenges & Problems

Calibration

- Our drone’s calibration had to be readjusted after we fixed our GPS

Obstacle Avoid

- Our drone wouldn't switch to offboard mode due to some hidden errors in our code (costing us quite some time), and then it wasn’t able to go over or under the AR Tag

Code got deleted

- Code was deleted on multiple occasions, hindered progress and slowed progress.