

(a) All launches file (sensor_init.launch, visualization.launch, dbw_joystick.launch)

- How enable/disable a sensor
- How the functionality of each launch file
- How to give executable permission to a python file

(b) How to correctly get good GNSS signals

Power off the LiDAR, drive the vehicle around and see whether the heading of the vehicle is correct or not on the highbay image.

(c) The PACMod-Joystick demo training

Run the can_start.bash file and launch dbw_joystick.launch. Show all potential issues that fail in launching the demo. For example, one of the emergency buttons is engaged.

(d) Show the useful ROS topics to students and explain them one by one

```
/lidar1/velodyne_points  
/novatel/imu  
/novatel/inspva  
/novatel/bestpos  
/zed2/zed_node/rgb/camera_info  
/zed2/zed_node/rgb/image_rect_color  
/zed2/zed_node/depth/camera_info  
/zed2/zed_node/depth/depth_registered  
/zed2/zed_node/imu/data  
/zed2/zed_node/imu/mag
```

(e) Launch the pure pursuit path tracker demo and explain the code (gem_gnss_tracker_pp.py)

- Explain self.olat and self.olon
- Explain /novatel/inspva topic
- Explain /pacmod/parsed_tx/vehicle_speed_rpt topic
- Explain other PAMod related topics in gem_gnss_tracker_pp.py
- Explain inspva_callback function
- Explain speed_callback function
- Explain heading_to_yaw function
- Explain front2steer function
- Explain read_waypoints function
- Explain wps_to_local_xy function
- Explain get_gem_state function
- Explain the rest of code related to pure pursuit controller