Drone Project Instructions

Note: the university may be judging you on technical merit but at the competition you will be judged on effectiveness of completing your task. The more complicated and/or proprietary the system you develop the more likely the system is to fail in competition. The drones ability to autonomously fly is the most important subsystem in this project and without it all others will not have the chance to be used.

Get your minimum viable subsystem functioning first and then transition to helping/checking in on how your drone operator is doing

Starting materials:
To buy:

- 3 Raspberry Pi 4 kits (one for each member or subsystem)
- 3 Raspberry Pi 4 cooling kits (image recognition will 100% kill their Pi without cooling)
- 3 SD cards
- 3 SD card readers
- 1 logitech webcam
- 1 portable router - extend bandwidth between computer on raspberry pi on competition field

To Check Out:

- Tarot 650 Drone
- Propellers
- 22.2 Volt 6s Lipo Batteries minimum 20C maximum 70 (not sure about max)
- Lipo Battery Smart Charger With 90 XT adapter to charge Lipo Battery
- Transponder
- Taranis X7 Controller

Subsystem #1: **The Drone**
You will be the drone expert. You are in charge of understanding the drone->remote control drone-> flight controller interface, and will be the expert on navigating the mission planner software. **Ask your team-mates for help when you are struggling**

1. Part 107 Drone License: **3 weeks**
    a. Request funding for both a study course and a test your drone operator license
    b. Schedule the test at a testing center
    c. Pass with a 70 or above
    d. Go on the website with your test score and submit it to the FAA requesting to get your official part 107 license.
        i. Approval will take between 2 and 14 days most likely
2. First Round Drone Testing: **2 weeks**
    a. Start applying to fly at Texas A&M Rellis or Main Campus
        i. You will not get approved your first time and are bound to submit something wrong so start **RIGHT NOW**
    b. Download mission planner on laptop

    c. Connect wireless controller to laptop and pair it with the drone
    d. Pair with a controller
        i. Check through Mission Planner to see your controllers bindings, and if there is no land switch this is the first thing you must bind.
    e. Test to see if the following systems are working on the drone through Mission Planner.
        i. Gyroscope
        ii. GPS
        iii. Motors
        iv. Barometer
3. Drone Testing Through Python: **??? weeks**
    a. Flight approval is the first thing you need to get
    b. Test List (perform at least the first round of these with a tether)
        i. Hover test
        ii. Land test
        iii. Fly Straight and Land Test
        iv. Fly Back and Forth Test
        v. Grid Pathing Test

Subsystem #2: **Image Recognition**

1. Find past years github of image recognition: **2 weeks**
    a. Follow the steps to get OpenCV version 4.8.1.78 downloaded
    b. Debug the code on your laptop and study the documentation
2. Connect a camera to Raspberry Pi 4: **2 weeks**
    a. Setup a raspberry pi 4 (internet connected and interfaced with a screen)
    b. Connect this camera to a standard webcam (I used a logitech c915)
        i. DO NOT USE A REALSENSE CAMERA WITH A RASPBERRY PI
    c. Debug to have camera interface with raspberry pi and recognize markers
3.

Subsystem #3: **Water Cannon**

1. Plan the system: **2-3 weeks**
    a. Find water blasting or ball shooting design that may be controlled by a raspberry pi 4 via coding interface like python
        i. INTERFACE WITH FLIGHT RASPBERRY PI IS #1

If a fourth member present

Subsystem #4: **Flight Computer Expert**

The Raspberry Pi 4 is a great and powerful tool when it is set up correctly. Your job could be to set up the raspberry pis early in the semester to save your team mates from the headaches involved in debugging it

1. Setup the Pi OS + Enable Internet + Setup VNC for Remote Viewing: **3 weeks**
    a. Have google handy
    b. Get ready to suffer unless you are lucky

    c. Copy the SD card when you are finished
2. Educate Team-Mates on their flight Computers: **3 weeks**
3. Flex/Professional Debugger: **?? weeks**
    a. You will help your teammates when they inevitably have compatibility issues/are always confused on what is happening with their flight computers

Contact jordanlloyd@me.com for questions