Drone API Initial Design

We provide an Interface class that all modules must implement.

We should provide some kind of version checking, so that as the API is updated, the modules are still useable.

# Data Types

We should probably make some standard data types for distance, speed, and coordinates. They should be clear the the end user what the units are, and should be well documented.

## Coordinate

This should be a basic coordinate, and should be usable with both relative positions and GPS positions. Should be possible to do basic addition/subtraction arithmetic on the coordinate as well.

### Variable

altitude

longitude

latitude

heading

# Parent Class

This is the class that the end user will use. There will need to be a way to choose the module, and this should do the version checking.

## variables:

module: we store the type of module here, so we can use it.

## functions:

setModule(Class): the module that we are going to use, store it in module.

enableCamera(): enable or disable camera

enableLocation(): <- perhaps separate modules here for location? GPS, accelerometer, time etc

enableRangeSensor(); <- perhaps should have separate modules for the range sensors, even amongst the same drone the sensors might be different

takeOff(): makes the drone lift up 3 feet and hover on the spot til ready

land(); forces the drone to land

flyTo(): flys to a given coordinate

# Drone Module Interface

## constants:

versionNumber: the version of the API the module matches. This can be checked against incase a certain function is added later

## functions:

getVersionNumber(): returns the version number

takeOff(): makes the drone lift up 3 feet and hover on the spot til ready

land(); forces the drone to land

flyTo(): flys to a given coordinate

# Camera Module

# Location Module