Agricultural Drone Survey Application

Design Manual

Bernard Steemers
C00235159@itcarlow,ie
Supervisor: Paul Barry
paul.barry@itcarlow.ie

Table of Contents

Introduction

The purpose of this document is to give the reader an overview of the design, development and structure of my final year project. In this document I will discuss and display my project plan, technologies, design and the user interface experience.

Technologies

Olympe:
Olympe is a python framework from the developers of the 'Parrot Anafi' that provides a python controller programming interface for Parrot and simulated drones.

Python:
Python is a general-purpose, high-level programming language. It's popular among software developers due to its simple syntax and readability.

Sphinx:
A simulation environment based on the Gazebo engine used to test drone software.

Google Maps Platform:
Google Maps Platform is a set of APIs and SDKs that allows the integration of maps into mobile apps and web pages in order to retrieve data from Google maps.

Javascript:
Javascript is a client and server side programming language that enables the interactivity or behaviours to webpages.

Geopy:
Geopy is a python library which provides geocoding services and sources such as providing distances between coordinates

Flask:
Flask is a micro web framework. It relies on third party libraries to provide common functions. It is written in Python and enables local hosting.

Linux Ubuntu 18.04 Bionic:
Linux is a family of open source Unix based computer operating systems that uses the Linux kernel. Ubuntu is a free Linux distribution based on Debian.
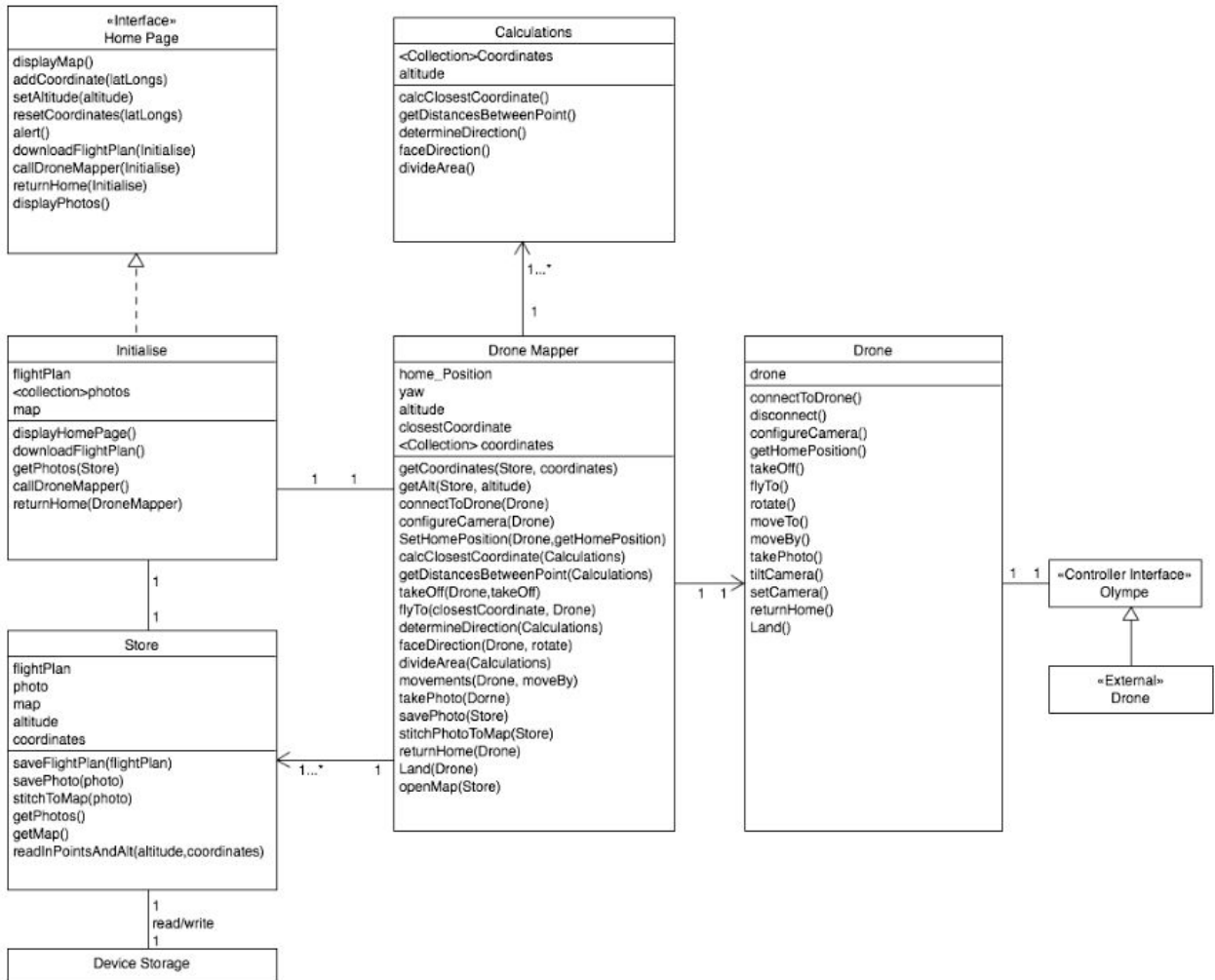
Project Plan

First, I will focus solely on implementing drone commands and instructions in order to establish what features will be of use from the Olympe library. Beginning with establishing a connection to the drone and from there building a library of useful commands and instructions that will shape the backbone of my projects capabilities. I intend to test these commands first inside of the Sphinx virtual environment and then in a real world environment with the Parrot Anafi.

Following some progression through these stages I hope to test the area coverage within photos taken at different altitudes. In doing this I'll be able to establish how much area has been captured within a photo and the distance the drone will have to travel between photos in an effort to avoid image overlap. I will then implement an algorithm to instruct the drones movements dependent on area coverage and altitude selection. Once this has been put in place, as previously, it will be tested both virtually and then physically. As this algorithm becomes more defined I will introduce a simple user interface that allows coordinate and altitude selection. This will be the first user friendly working prototype of my project.

With continuous refinement I will attempt to produce an image stitch algorithm in order to produce an accurate representation of the area covered by the drone. Moving on from here it will be a matter of building upon modules already implemented and introducing new modules to strengthen the structure of the project's design.
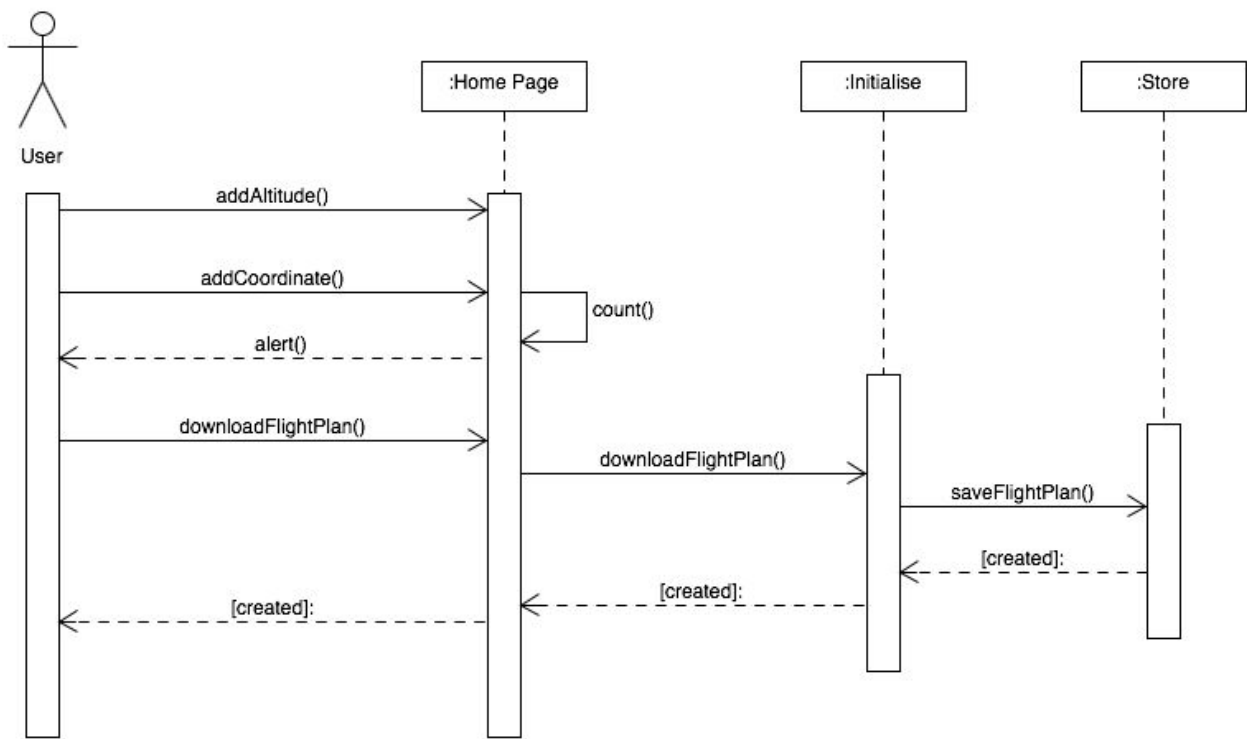Below you will find some current designs for my project structure and implementation.
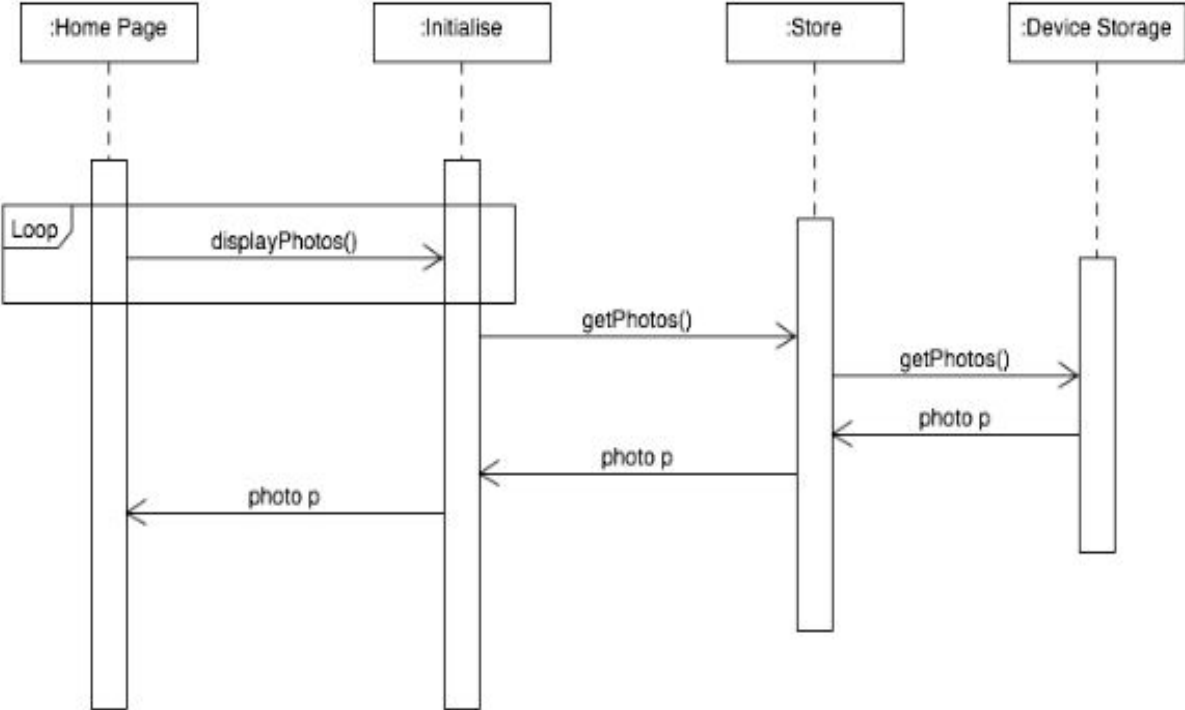
Class Diagram

«Interface» Home Page

displayMap()
addCoordinate(latLongs)
setAltitude(altitude)
resetCoordinates(latLongs)
alert()
downloadFlightPlan(Initialise)
callDroneMapper(Initialise)
returnHome(Initialise)
displayPhotos()

Calculations

<Collection>Coordinates
altitude

calcClosestCoordinate()
getDistancesBetweenPoint()
determineDirection()
faceDirection()
divideArea()

Initialise

flightPlan
<collection>photos
map

displayHomePage()
downloadFlightPlan()
getPhotos(Store)
callDroneMapper()
returnHome(DroneMapper)

Drone Mapper

home_Position
yaw
altitude
closestCoordinate
<Collection> coordinates

getCoordinates(Store, coordinates)
getAlt(Store, altitude)
connectToDrone(Drone)
configureCamera(Drone)
SetHomePosition(Drone,getHomePosition)
calcClosestCoordinate(Calculations)
getDistancesBetweenPoint(Calculations)
takeOff(Drone,takeOff)
flyTo(closestCoordinate, Drone)
determineDirection(Calculations)
faceDirection(Drone, rotate)
divideArea(Calculations)
movements(Drone, moveBy)
takePhoto(Dorne)
savePhoto(Store)
stitchPhotoToMap(Store)
returnHome(Drone)
Land(Drone)
openMap(Store)

Drone

drone

connectToDrone()
disconnect()
configureCamera()
getHomePosition()
takeOff()
flyTo()
rotate()
moveTo()
moveBy()
takePhoto()
tiltCamera()
setCamera()
returnHome()
Land()

«Controller Interface» Olympe

«External» Drone

Store

flightPlan
photo
map
altitude
coordinates

saveFlightPlan(flightPlan)
savePhoto(photo)
stitchToMap(photo)
getPhotos()
getMap()
readInPointsAndAlt(altitude,coordinates)

Device Storage

1...*
1

1

1
1

1   1

1   1

1...*   1

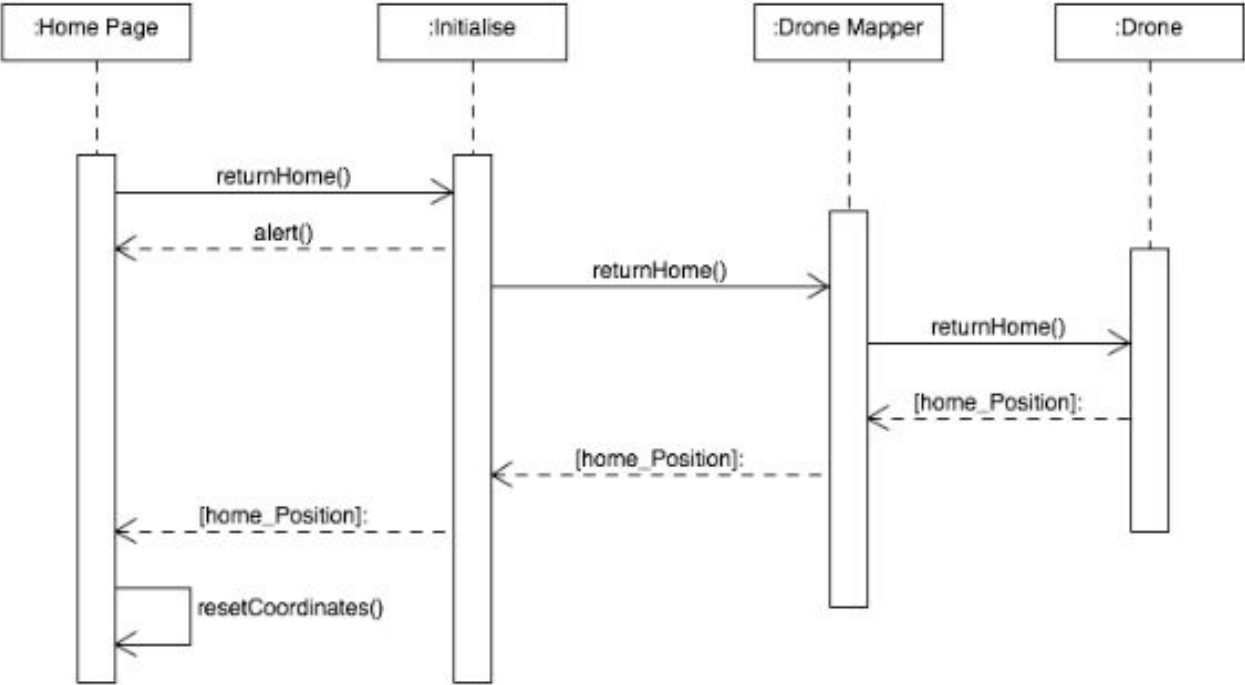1   1

1
read/write
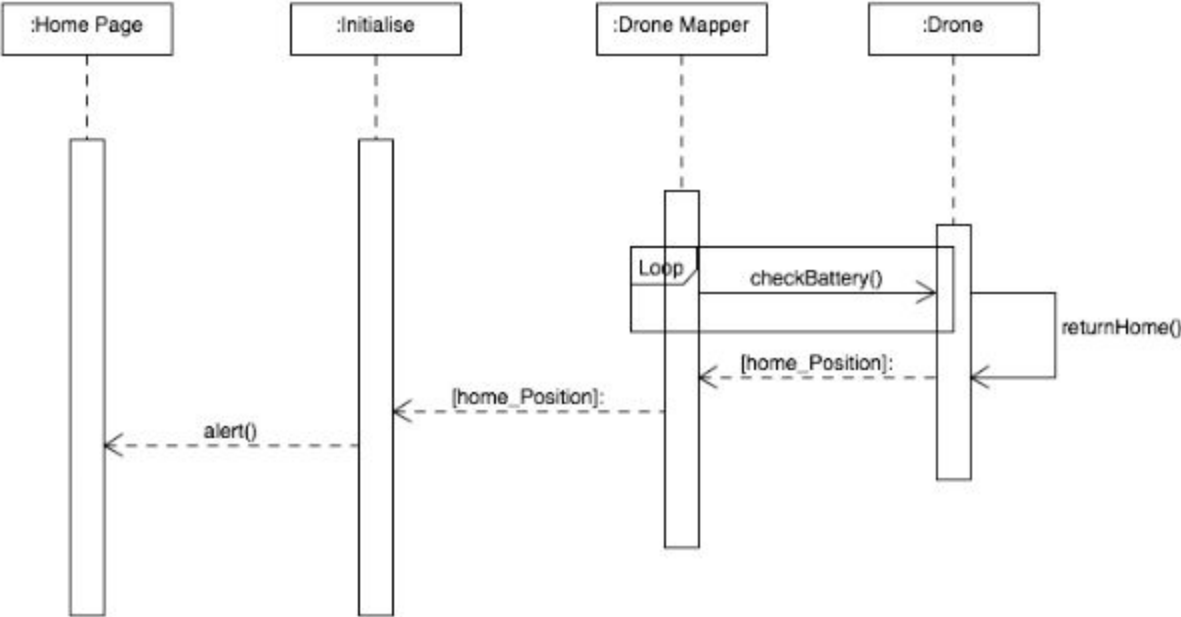1

6

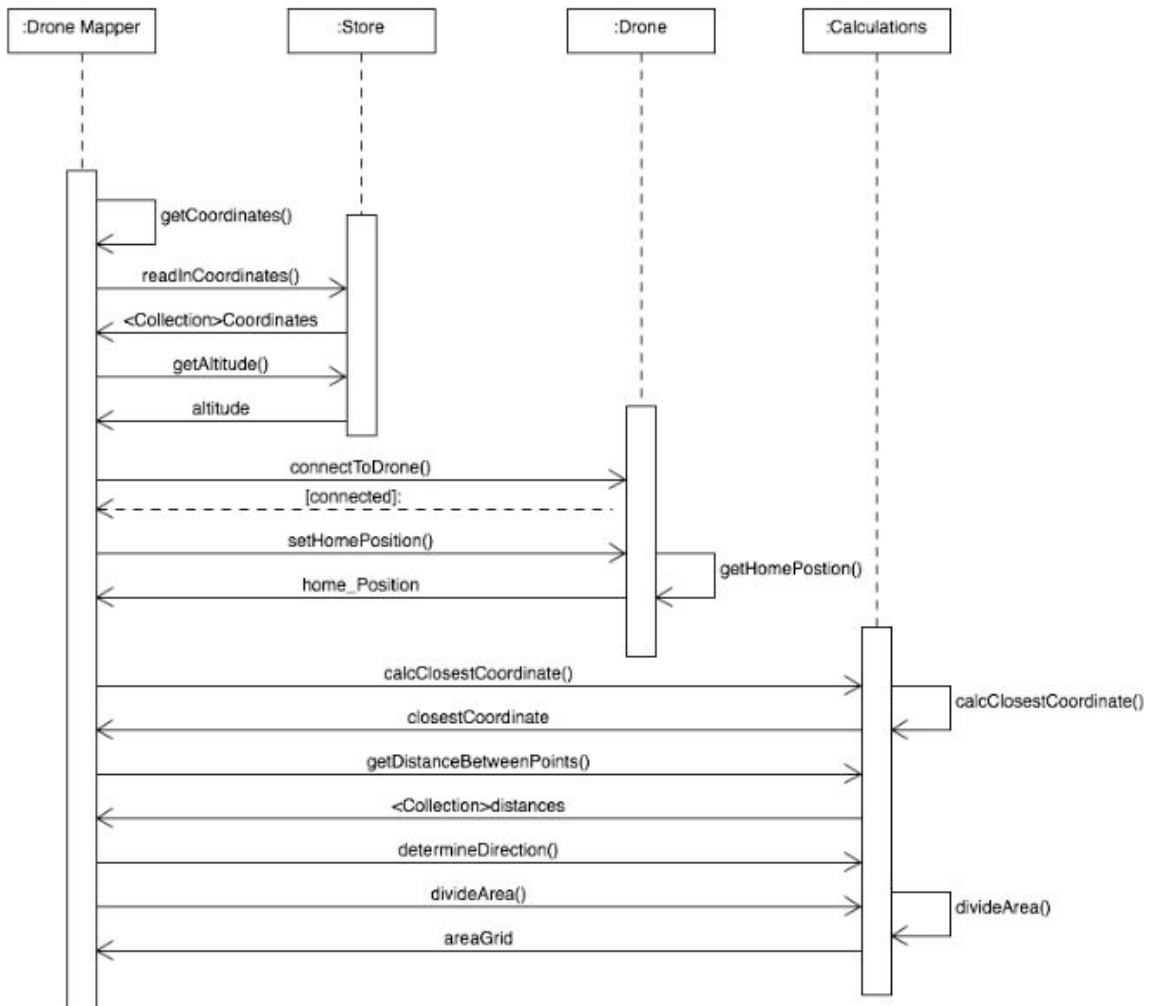# Sequence Diagrams

Create Flight Plan:

View Aerial Images:

Interrupt Flight Plan:

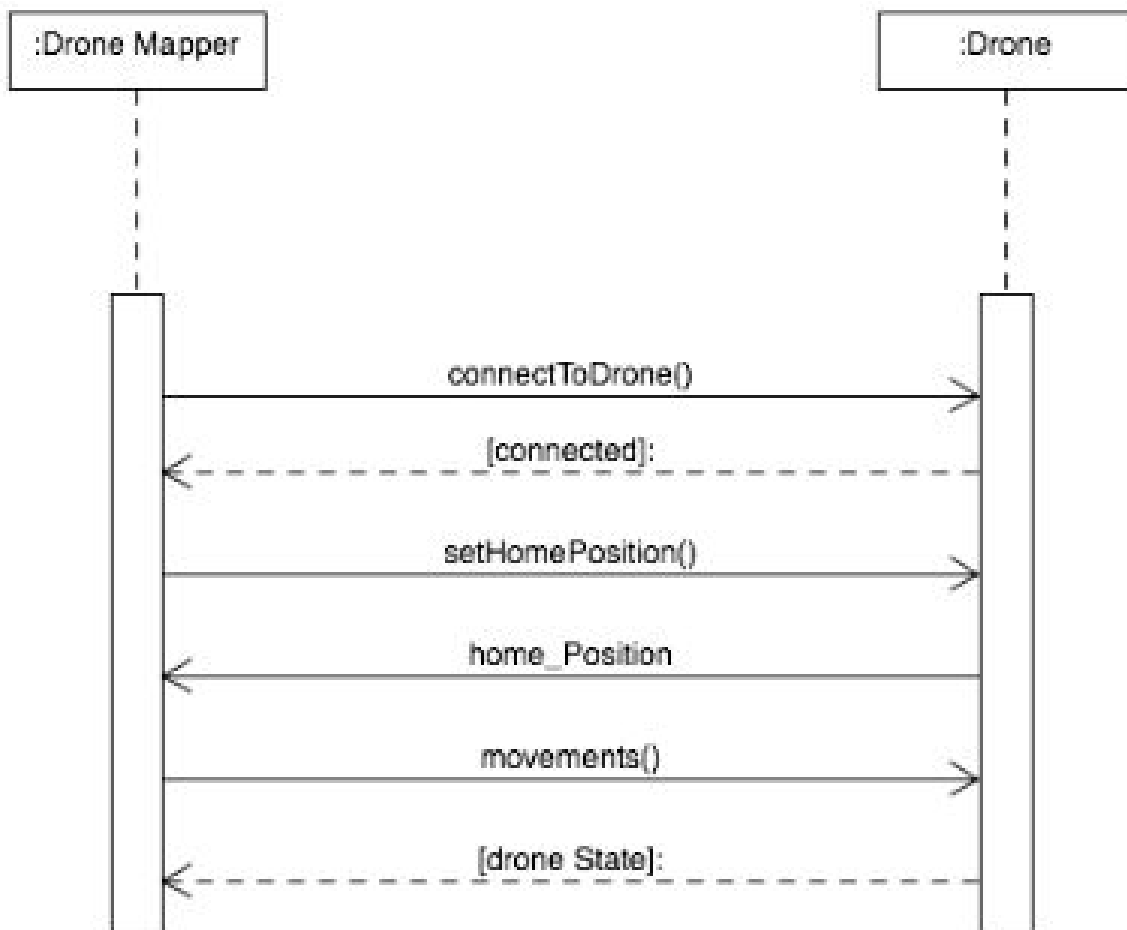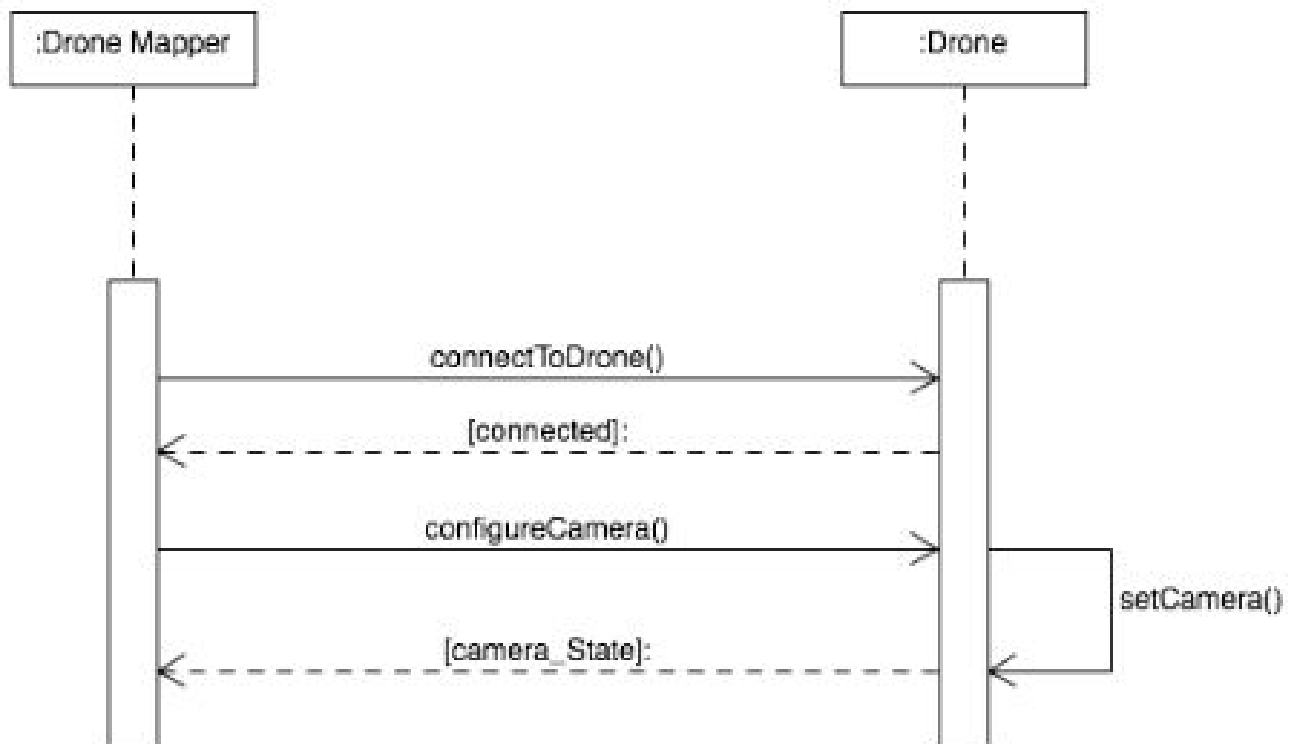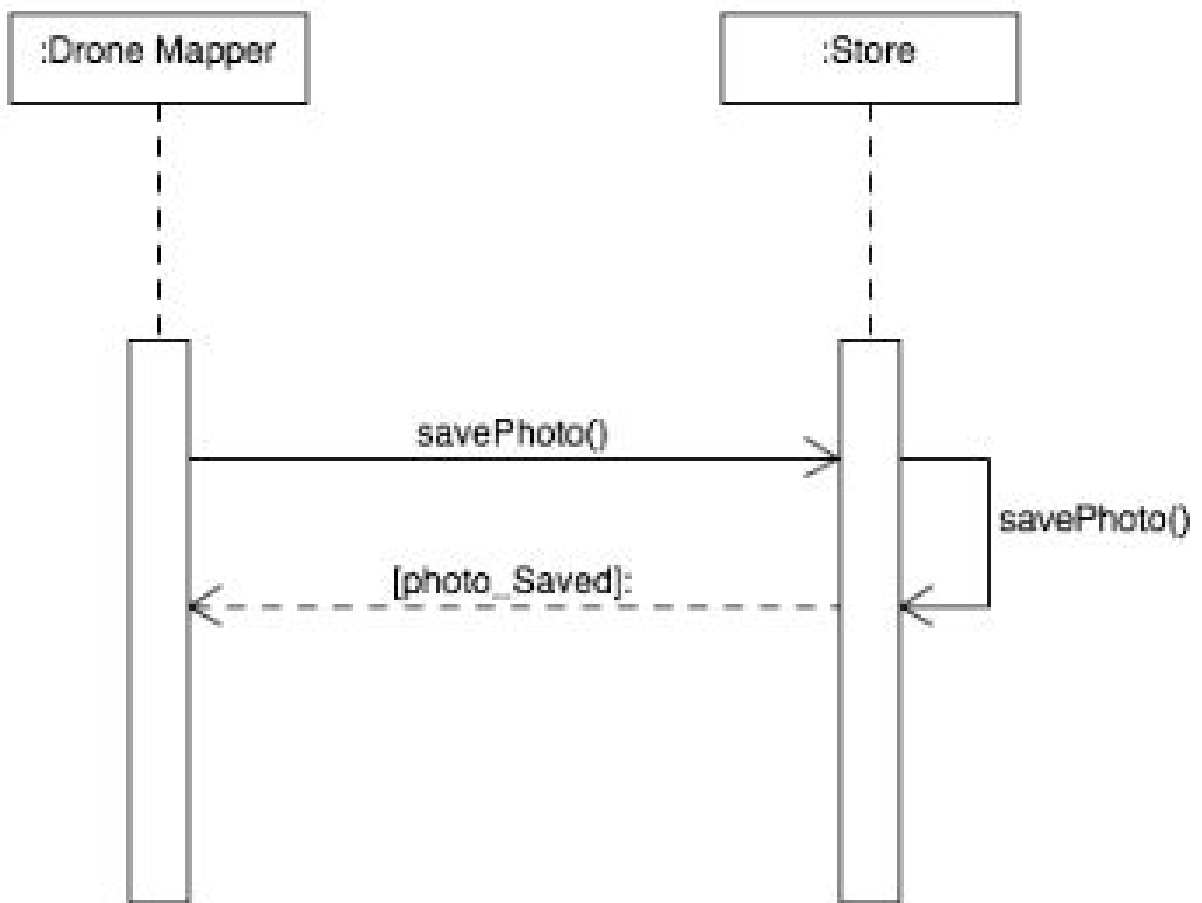# Battery Low RTH:

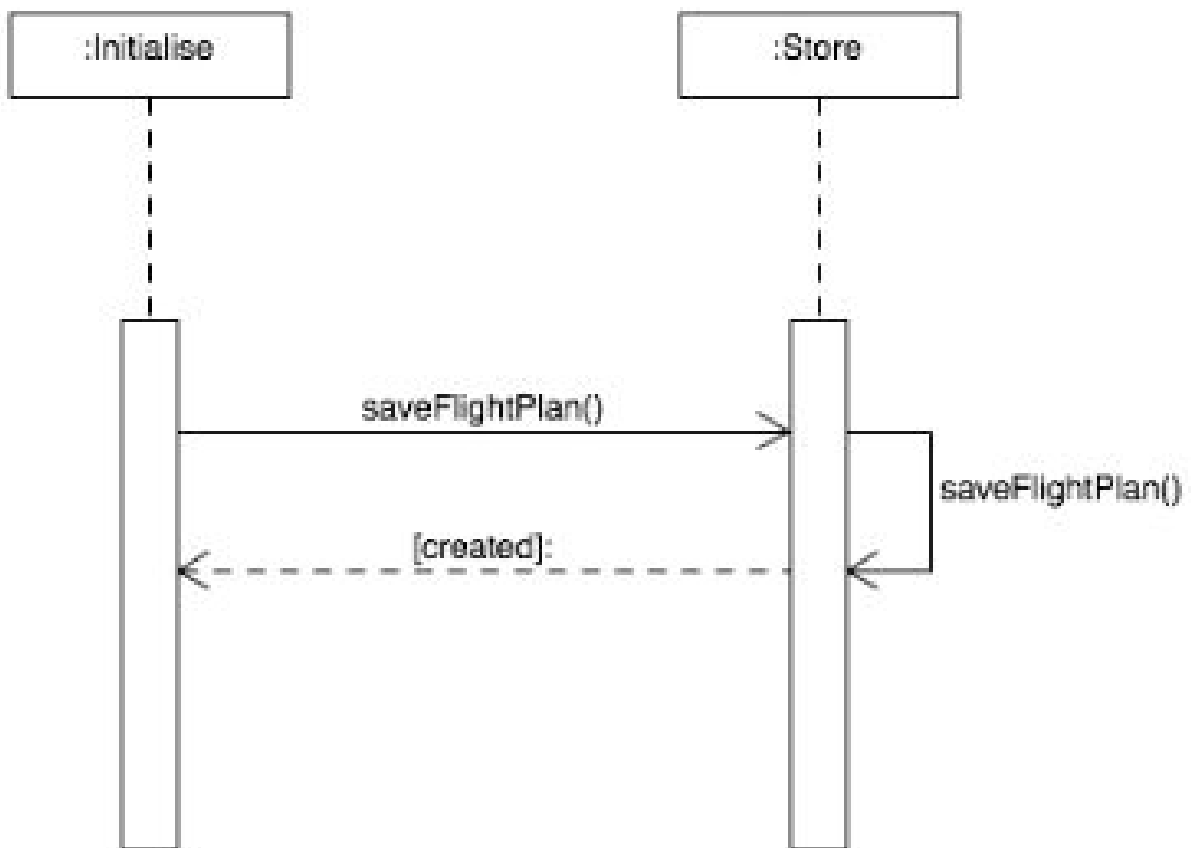# Create Flight Path & Map:

Send Command Instruction:

Configure Camera Settings:

Store Aerial Images:

Store Flight Log Records:

User Interface

Below is an early design of what a user interface might look like for my project. On the left hand side there's an interactive map, from which a user can choose their location and an area to survey.
On the right hand side there's a list of chosen coordinates and a drop down menu to select an altitude from. At the bottom right there's an option to download coordinates and launch the drone.

System Architecture

Below is a diagram showing application layer architecture and the technologies used.
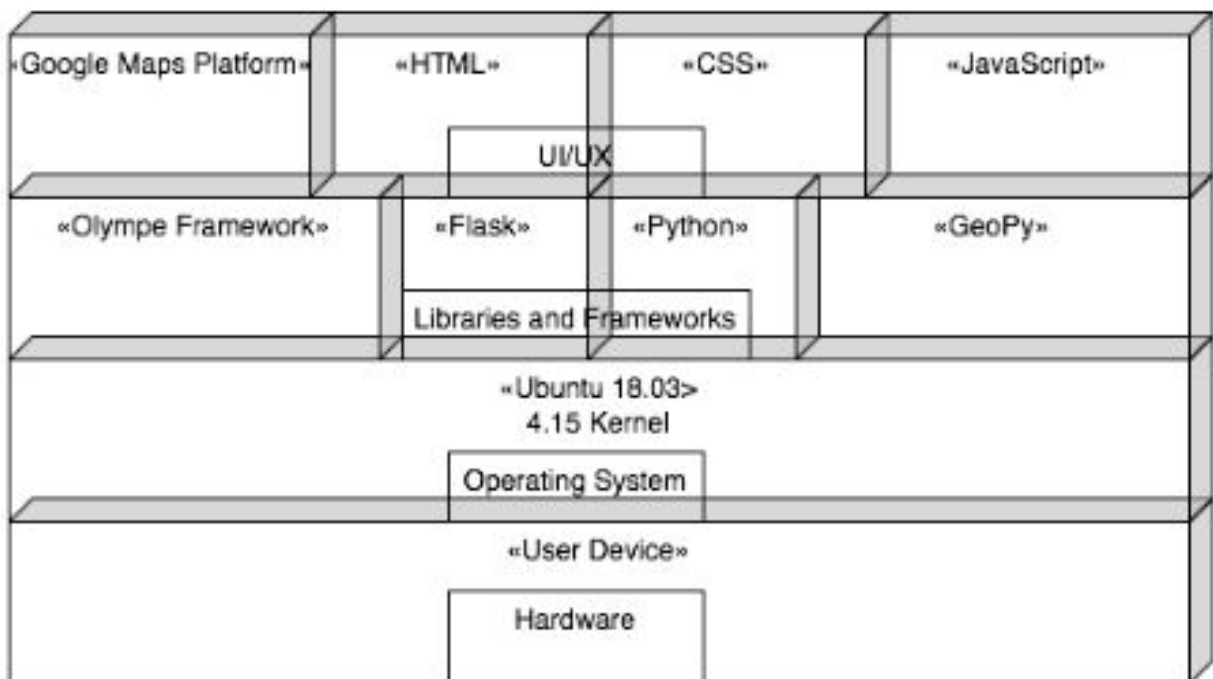
Declaration

- I declare that all material in this submission e.g. thesis/essay/project/ assignment is entirely my/our own work except where duly acknowledged.

- I have cited the sources of all quotations, paraphrases, summaries of information, tables, diagrams or other material; including software and other electronic media in which intellectual property rights may reside.

- I have provided a complete bibliography of all works and sources used in the preparation of this submission.

- I understand that failure to comply with the Institute's regulations governing plagiarism constitutes a serious offense.

Student Name: Bernard Steemers

Student Number: C00235159

Signature:

Bernard Steemers

Date: 20/04/2020