



Peregrine Jet UAV

Software Requirements Specification Document

Sponsor:

Dr. Siddhartha Bhattacharyya

Prepared by:

Nabil Osorio, Christian Cambron, Anthony Messina

9/25/2018

Table of Contents



1. Introduction
 - 1.1 Purpose
 - 1.2 Scope
 - 1.3 Definitions acronyms, and abbreviations
 - 1.4 References
 - 1.5 Overview
2. Overall description
 - 2.1 Product perspective
 - 2.2 Product functions
 - 2.3 User Characteristics
 - 2.4 Constraints
3. Specific requirements
 - 3.1 External interface requirements
 - 3.1.1 User interfaces
 - 3.1.2 Hardware interfaces
 - 3.1.3 Software interfaces
 - 3.1.4 Communications interfaces
 - 3.2 System Features
 - 3.2.1 System Feature 1
 - 3.2.1.1 Introduction/Purpose of feature
 - 3.2.1.2 Stimulus/Response sequence
 - 3.2.1.3 Associated functional requirements
 - 3.2.1.3.1 Functional requirement 1
 - 3.2.1.3.2 Functional requirement 2
 - 3.2.1.3.3 Functional requirement 3
 - 3.2.1.3.4 Functional requirement 4
 - 3.2.2 System Feature 2
 - 3.2.2.1 Introduction/Purpose of feature
 - 3.2.2.2 Stimulus/Response sequence
 - 3.2.2.3 Associated functional requirements
 - 3.2.2.3.1 Functional requirement 1
 - 3.2.2.3.2 Functional requirement 2
 - 3.2.2.3.3 Functional requirement 3
 - 3.2.3 System Feature 3
 - 3.2.3.1 Introduction/Purpose of feature
 - 3.2.3.2 Stimulus/Response sequence
 - 3.2.3.3 Associated functional requirements
 - 3.2.3.3.1 Functional requirement 1
 - 3.2.3.3.2 Functional requirement 2



3.3.3.3.3 Functional requirement 3

3.3.3.3.4 Functional requirement 4

3.3 Performance requirements

3.4 Design constraints

3.5 Software system attributes

3.6 Other requirements

1. Introduction

1.1 The purpose of this document is to outline the requirements for the Peregrine Jet UAV's software components. These components include a functional graphical user interface, control systems, and processing relevant flight data for use by the pilot. The user interface and data



processing will be written in Python, and the control systems will be written in C++. The software will run on Windows OS.

1.2 All software will be housed natively on the ground control station, which will connect remotely to the UAV itself.

1.3 Definitions, Acronyms, and Abbreviations

UAV - Unmanned Aerial Vehicle

GUI - Graphical User Interface

FPV - First Person View

Tx - Transmitter

Rx - Receiver

1.4 References

<https://www.python.org/>

<http://www.cplusplus.com/>

<https://docs.python.org/3/library/tkinter.html>

<https://opencv.org/>

<https://uavcoach.com/infographic-drones-work/>

<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=720574&tag=1>

1.5 This document contains all of the specifics of the software requirements. It contains a list of all technologies we will utilize to make the UAV function properly. We will describe and outline the specific components of the project.

2. Overview Description

2.1 Product Perspective

2.1.1 The Product that is being created is not an independent system. The Software is going to be made to work with Peregrine II Jet UAV. Peregrine II is in unmanned Jet powered aircraft.

2.2 Product function



2.2.1 Controls - The Program is going to pare with transmitters that control Peregrine. The controls include

2.2.2 Autonomous Features - The software is going to need to handle several autonomous features. The first Includes FAA crash Avoidance. FAA protocol states that the Airplane if in a direct crash course with another must turn right. This will have to be stored on Peregrine II. Another second Autonomous feature is that it must be able to turn around to last known signal transmission it receives if the signal is lost. The last feature it must have a fly a pre-programmed flight plan. This might be activated by certain buttons.

2.2.3 Graphical User Interface - Another part of the software is it's going to need a GUI The GUI is going to display speed, altitude, gas level, Orientation, Direction, live video, etc.

2.3 User Characteristics - The user of the program has to be a registered/certified pilot with the FAA.

2.4 Constraints

2.4.1 FAA - The jet must not Fly above 400ft. This will be programmed into the parameters. The jet must not exceed 100 mph this will also be programmed into the parameters.

2.4.2 Hardware Limitations - The signal we will be allowed to transmit is 2.4ghz. This limits the distance of the signal.

2.5 Apportioning of Requirements

2.6.1 - Specific Parts of the program may not be able to be fully functioning until all hardware is received.

3. Specific requirements

3.1 External interfaces

3.1.1 User interface that displays the fpv of the UAV alongside specific data output including, but not limited to, speed, altitude, direction, etc. on the same or separate display.

3.1.2 Hardware interfaces

3.1.2.1 Pixhawk for connection of sensors

3.1.2.2 Joystick for controlling the UAV



3.1.2.3 Computer for receiving/transmitting data and displaying output
onto the GUI

3.1.2.4 Raspberry Pi for taking control of the UAV in the case of
disconnection from the ground station

3.1.3 Software interfaces

3.1.3.1 The software on the ground station shall interact with the software
located within the UAV through the PixHawk or Raspberry Pi to set new coordinates to the UAV
or verify that the UAV still has a connection to the ground station

3.1.4 Communications interfaces

3.1.4.1 The only communication interface between the pilot and the UAV
shall be through the joysticks at the ground station

3.2 System Features

3.2.1 GUI for the pilot

3.2.1.1 The GUI shall display the fpv or the UAV as well as all the
important information sent from the Pixhawk to the ground station

3.2.1.2 The stimulus for the GUI will be the video feed and data gathered
from the sensors. The GUI will display the live video feed and on top of the video, the GUI will
represent the data it has received in a form that the pilot could understand

3.2.1.3 Associated functional requirements

3.2.1.3.1 Display live video feed

3.2.1.3.2 Display airspeed

3.2.1.3.3 Display altitude

3.2.1.3.4 Display direction of flight

3.2.2 Control system

3.2.2.1 This system is the main control for the UAV. The pilot will use
the joysticks that transmit the desired positioning to the UAV

3.2.2.2 Stimulus would include the new coordinates being received by the
system on the UAV and the response would be readjusting the UAV's wings, flaps, and thrust to
allow for such change in direction

3.2.2.3 Associated functional requirements

3.2.2.3.1 The Tx will transmit the specified coordinates or desired
position of the UAV to the Rx located on the UAV. The Rx will then send that information to
the Pixhawk to decide how to what the UAV must do in order to complete that task

3.2.2.3.2 The system must allow the jet engine to start from the
transmitter

3.2.2.3.3 The system must allow for engine shut off or fuel cut off
functionality

3.2.3 Data conversion system

3.2.3.1 This system will receive data sent from the ECU or sensors
mounted on the UAV. This will allow the pilot to monitor the flight condition during test flights



3.2.3.2 Stimulus include RS232 data sent from the ECU and other unknown data types from the various sensors. The system decodes the data and converts it into understandable information to display in the GUI

3.2.3.3 Associated functional requirements

3.2.3.3.1 Receive data in various signal types

3.2.3.3.2 Decode the data

3.2.3.3.3 Send information to the GUI for display

3.2.3.3.4 calculations shall take no less than 5ms

3.3 Performance requirements include but are not limited to latency from the ground station to the UAV shall be no more than 5ms

3.4 Design constraints will be limited by the FAA regulations. More specifically section 107 from the FAA website

3.5 The software must be easily modified for simple testing and future projects, as well as fast in performance, to allow for usage on faster UAVs

3.6 Other requirements include but are not limited to the mobility of the system in regards to switching between RC planes and flight simulators