March 13, 2022

Dr. Mike Hegedus
School of Engineering Science
Simon Fraser University
British Columbia, V5A 1S6

RE: ENSC 405W/440 Design Specifications for Putt from the Sky

Dear Dr. Hegedus,

Improving the golf playing field is the core of what Putt From the Sky values. Bringing consistent, accurate, and detailed green maps will allow both the professional and the amateur to succeed on the course. As such, Putt From the Sky is a golf green mapping service provided to golf courses around the world. The purpose of the attached document is to breakdown and analyze the design choices for the green mapping service.

First, the design document will cover the overview of the system as well as a breakdown of the flow of the components. Then, the image processing pipeline and the green map details are justified. Finally, specifics of the hardware chosen for the design, as well as justification for each selection is explained. Together, these components will compose the entire system.

Putt From the Sky consists of five engineering students spanning electrical, systems, and computer engineering. It is a hard-working and dedicated team that sets ambitious goals and strives to achieve them. The primary contact at Putt From the Sky is Chief Communications Officer, Sam Kwon, who can be reached by email at ockwon@sfu.ca or by phone at 613-614-9010.

Sincerely,

Ryan Stolys
CEO
Putt From the Sky

Design Specifications

Putt from the Sky

ENSC 405: Company 4
Issued on March 13, 2022

Prepared By:

Ryan Stolys
Chief Executive Officer

Jayden Cole
Chief Technical Officer - Software

Matthew Nesdoly
Chief Technical Officer - Hardware

Jiyeong (Sophie) Jeong
Chief Financial Officer - Inventory Manager

Oh Chang (Samuel) Kwon
Chief Communications Officer

Abstract

The following document outlines the design specifications of Putt From the Skys green mapping system. This project will semi-automate the golf green scanning and modeling process using photogrammetry. A drone will acquire images that will be processed through a structure-from-motion image processing pipeline. Finally, a 2D surface map with slopes and terrain contours will be produced for the consumer. This will allow for amateur and pro golfers to improve their putting with high quality green surface knowledge. The project will be considered successful when the maximum allowable green book resolution is attained. However, a less precise model will be iteratively improved until reaching the targeted resolution. The design specification outlines design choices made by the Putt From the Sky team as well as alternative options available to the team and why they were not chosen.

List of Figures

Figure	Caption
2.1.1	System Block Diagram
4.1.1	Software System Architecture
4.4.1	Slope Model Generation Flow
4.4.2	Least Squares Error 2D Plane Example
5.1.1	3-D Model of the Drone
5.1.2	Raspberry Pi with Navio2 Attached to the Pins
5.1.3	Frame Size of the Drone
5.1.4a	CW Propellers
5.1.4b	CCW Propellers
5.1.5	Lithium Polymer Battery 5000 mAh 11.1V
5.1.6	Brushless Motor 1000kv 15A
5.1.7	Electronic Speed Controller (ESC) with 30A Maximum Current Rating
5.2.1	Full Circuit Design of the Drone
5.2.2	Shock/Vibration Absorber for Navio2
5.3.1	Navio2 with UART Connection to Radio Telemetry Device
5.3.2	Mission Planner Software Designed by ArduPilot with GPS Waypoints
5.4.1	Connection Point of the Camera Module and Raspberry Pi
5.5.1	Connection Diagram of Raspberry Pi
7.1	System Interaction Point Block Diagram
7.2.1	Drone Path Mapping Interface
7.2.2	Graphical User Interface for Drone Path Monitoring
7.2.3	Graphical User Interface of the Image Processing Pipeline
7.2.4	High Accuracy Green Diagram with Slopes
7.2.5	Complete 3-D Model of the Drone

List of Tables

Table	Caption
1.4.2	Development Stage Encoding
5.1.1	Calculation of the Total Weight of the Drone
7.1	Engineering Standards

Glossary

The following table includes terms mentioned throughout the document specific to our domain and industry.

Term	Definition
DEM	Digital Elevation Model
CW	Clockwise
CCW	Counter-Clockwise

Contents

Abstract 4

List of Figures 5

List of Tables 6

Glossary 6

1 Introduction & Background 9
- 1.1 Scope 10
- 1.2 Intended Audience 10
- 1.3 Design Specification Classification 10

2 System Overview 11

3 Overall System Design 12

4 Software Design 14
- 4.1 Overall Software Design 14
  - 4.1.1 Overall Software Design 16
- 4.2 Image Parsing Module Design 17
- 4.3 Image to Point Cloud Design 17
  - 4.3.1 Feature Detection using the SIFT Algorithm 18
- 4.4 Slope Generation Module Design 18
  - 4.4.1 Slope Model Generation Algorithm 19

5 Hardware Requirements 21
- 5.1 General Hardware Design 21
- 5.2 Electrical Design 26
- 5.3 Path Mapping and Controlling the Drone 28
- 5.4 Data Acquisition 30

5.5 Data Transfer 31

6 Conclusion 32

7 Appendix A: User Interface Design 33
- 7.1 Introduction 33
  - 7.1.1 Purpose 33
  - 7.1.2 Scope 33
- 7.2 Interface Design 33
  - 7.2.1 User Analysis 34
  - 7.2.2 Technical Analysis 37

        7.2.3 Safety and Sustainability 40
    7.3 Testing and Verification 40
        7.3.1 Empirical Usability Testing 40
        7.3.2 Analytical Usability Testing 41
        7.3.3 Engineering Standards 42
    IEC 61800-5-1 Ed. 2.1 b:2016 42
    7.4 Conclusion 43
    7.5 References 43

8 Appendix B: Test Plan 44
    8.1 Introduction 44
    8.2 Data Collection Unit Testing 45
    8.3 Image Processing Unit Testing 48
    8.4 Overall System Testing 50
    8.5 Consumer Testing 50

9 Appendix C: Alternative Design 50
    9.1 Physical Drone 50
    9.2 Image Data Transfer Interface 51
    9.3 Camera For Data Acquisition 51

10 References 52

1 Introduction & Background

Over the last 20 years, the golf world has seen a substantial increase in the use of technology to help improve performance at the top level of the game. Putt From the Sky intends to add to this expansion by addressing an area of golf that still relies on unreliable and inaccurate practices. This particular area is the yardage book used by amateurs and professionals alike. Yardage books are a set of notes and maps for each of the 18 holes of a golf course that contains information such as distance to specific hazards, reminders for the player, and maps of the green. Except for the professionals playing at the highest level, the green maps in yardage books are handwritten maps that usually lack detail and are not to proper scale. Despite competitor StrackaLine® providing some services related to green mapping, their methods are not widespread. As such, green maps are an untapped target market of the technological expansion in golf.

To do this, Putt From the Sky will use Digital Elevation Models (DEMs), first introduced near the end of the 1900s. Over the past 40 years, DEM quality and application in computer vision projects has improved. Current DEMs have reached sufficient quality to be applicable to model water flow, render 3D visualizations, satellite navigation, surface analysis, and many others [1]. Although there are many ways to acquire the data for a DEM, photogrammetry currently stands out because of its ability to make a high-resolution model from a set of images. Putt From the Sky is determined to add golf green mapping as a DEM application using photogrammetry.

The Putt From the Sky team expects numerous challenges in automating the making of a yardage book as well as digitizing the green. From a hardware perspective, the system must be adaptable to the differing terrain characteristics from hole to hole while maintaining a high quality data acquisition. Image acquisition path planning is a critical aspect of this as well. The solution must be adaptable to great differences in terrain. From a software perspective, there are expected challenges with outlier data points from acquisition noise interfering with smooth surface reconstruction which could significantly affect the resulting model.

1.1 Scope

This document outlines the high-level system and design requirements of the green mapping solution from Putt from the Sky. We have also divided the specifications by their sub-system. Each requirement has been categorized into one of the three following categories: Proof-of-concept, Engineering Prototype and Production Version.

1.2 Intended Audience

This document is intended for the team members of Putt from the Sky, potential partners and the teaching team of ENSC 405W and ENSC 440, namely Dr. Mike Hegedus, Usman Ahmed, Chis Hynes and Ghazal Mirab.

1.3 Design Specification Classification

The requirements in this document are classified using the following convention:

D [Section].[Subsection].[Requirement Number] - [Stage of Development]

The Section and Subsection values will come from headings in this document. The Stage of Development values will come from the following encoding table outline our 3 stages of development.

Code	Stage of Development
A	Proof of Concept
B	Engineering Prototype
C	Final Production

Table 1.4.1 - Development Stage Encoding

2 System Overview

As shown in the block diagram of the system below, the process begins with an operator. The operator will handle all hardware tasks and is responsible for gathering the golf green data on-site. The operator will begin by utilizing the on-site processing unit to create a data collection plan for the data collection device. The processing unit will have software that the operator can interface to design an effective path for a particular golf green. The instructions created in this plan will tell the drone what path to travel, where to take pictures, and at what elevation the images should be taken. The data collection device will have a GPS sensor so that it can move to precise locations determined by the path. This wireless and autonomous system is utilized for the purpose of capturing images with their respective GPS location.

Once the data has been collected, it will be transferred via SD card to the main processing system at another location. From this point, the data will be automatically sorted and formatted for the next stage of the software pipeline. Next, the system will generate a Digital Elevation Model (DEM) which will be utilized to create the final slope map output. The first processing step is the feature detection where the system will detect features that are present in multiple of the photos. A features position can be calculated by triangulation (using GPS data) to produce 3-dimensional point measurements. A point cloud will be generated from this processed data and used to create a digital elevation model. Putt from the Sky will use this DEM to produce a slope map of the green surface which can be interpreted by golfers.

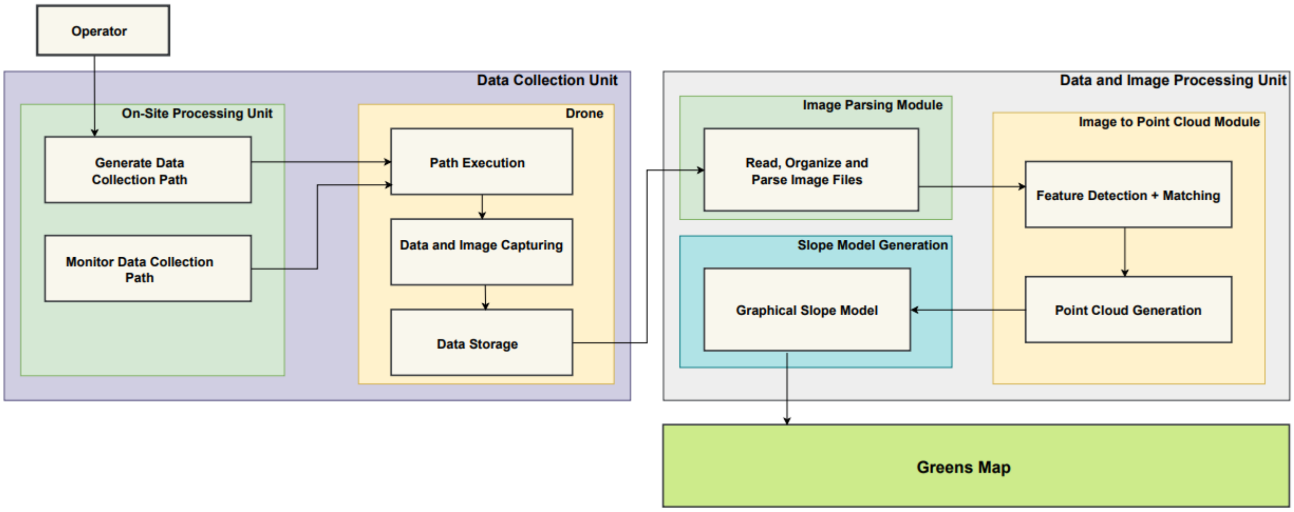


Figure 2.1.1 - System Block Diagram

3 Overall System Design

In this section we will describe overall system design specifications. We will provide rationale for why these decisions were made, and show tables, equations and figures to better describe how each decision will impact the overall system.

D 3.1.1 - B

The system will provide a detailed 2D map of a green, with slope and gradient information. It will be available in paper and pdf format.

Related Requirements: R 3.2.2 , R 4.1.6

Rationale: Throughout the systems process there are several different views of the golf green that could be presented, such as a 3D model. This system will produce a 2D topographic map with slope gradient information because that is how amateur and pro golfers are accustomed to reading information about the greens. A pdf is easy to access online and print out copies can be delivered to customers at a course.

D 3.1.2 - B

The system is designed in a manner that requires a user to have training.

Related Requirements: R 3.2.3, 4.1.2, R 4.1.4

Rationale: The system will require specific knowledge to use effectively and efficiently. As such, the operator must have the knowledge to modify system parameters correctly, place the system in the correct position, ensure the safety of the system and environment, and recognize when part of the dataset is compromised. These elements are key factors to have a productive and useful product.

D 3.1.3 - B

The data collection device will use a camera to collect images, which will be used to perform photogrammetry.

Related Requirements: R 3.1.6

Rationale: This design decision was made so that Putt From the Sky can compare generated outputs to state of the art modeling softwares. A camera as a hardware tool is affordable and easier to use than other applicable data collection devices. Photogrammetry is often used to develop terrain models[1]; as such there is applicable scientific literature to aid in the development of a software system.

D 3.1.4 - B

The data collection device is a quadcopter drone.

Related Requirements: R 3.1.2, R 3.1.4, R 3.2.1, R 3.2.3, R 4.1.1, R 5.1.2, R 5.1.5

Rationale: A drone is capable of capturing images from high vantage points, and is able to work autonomously. It can operate in variable weather conditions and provide a steady and stable environment for a camera. The software used in this system will benefit from the variety of pictures a drone can acquire. A drone can fly and capture images at a fast rate and complete the data collection process swiftly and efficiently.

D 3.1.5 - B

The system will include a graphical user interface to upload path mapping details to the drone and monitor the drone as it flies.

Related Requirements: R 3.2.1, R 3.2.3

Rationale: A GUI is necessary for the operator so they can efficiently set up the drone to fly its path. It is necessary to monitor the drone so that it does not get damaged or hurt anything else. Included in section 7 is the design for this interface. Section 7.2.1.1 to 7.2.1.4 describes what the interface will do, and what it will look like. Section 7.2.2.1 to 7.2.2.4 describes how the operator will interact with the interface and what features of the interface itself affect the operator's experience.

4 Software Design

Putt from the Sky has three main software components in the system: Image Parsing, Image to Point Cloud (I2PC) and Slope Model Generation. This section will describe the design of each of those components in addition to overall design of the software. The software in the system is responsible for transforming data collected from images of a golf green into a slope map.

4.1 Overall Software Design

D 4.1.1 - A
The software shall provide processing progress and output logs to the user through a graphical user interface built using *pysimplegui*
Related Requirements: R 4.1.3, R 4.1.4
Rationale: The software will interface with trained operators and this library will enable developers to build a system which meets the specifications described in Appendix A.

D 4.1.2 - A
The software shall output the slope map in the software systems ‘greens’ folder
Related Requirements: R 4.1.6
Rationale: The system will output the slope map to a common folder. This will simplify the supervision needed while the software is processing the images and generating the greens map.

D 4.1.3 - A
The software shall output the slope map in a printable pdf format
Related Requirements:
Rationale: The slope map will be shared with the consumer in a physical form. A pdf will allow Putt From the Sky to generate and store a digital copy of this map for multiple consumers requesting the same greens map.

D 4.1.4 - A
The software shall be defined in independent modules. The outputs of the preceding module will be the inputs of the next module.
Related Requirements: R 4.1.6
Rationale: Developing independent modules will allow development of each component to be done iteratively and separately. Further, modularization allows testing against outside benchmarks, when available. The software architecture is described in subsection 4.1.1.

D 4.1.5 - A
The software shall be developed using python
Related Requirements: R 4.2.1

Rationale: Running python code involves setting up a python virtual environment. This environment can be set up on any of the hardware processors commonly available today (x86, ARM). This will give Putt from the Sky flexibility in the computer used by the operators of the system.

4.1.1 Overall Software Design

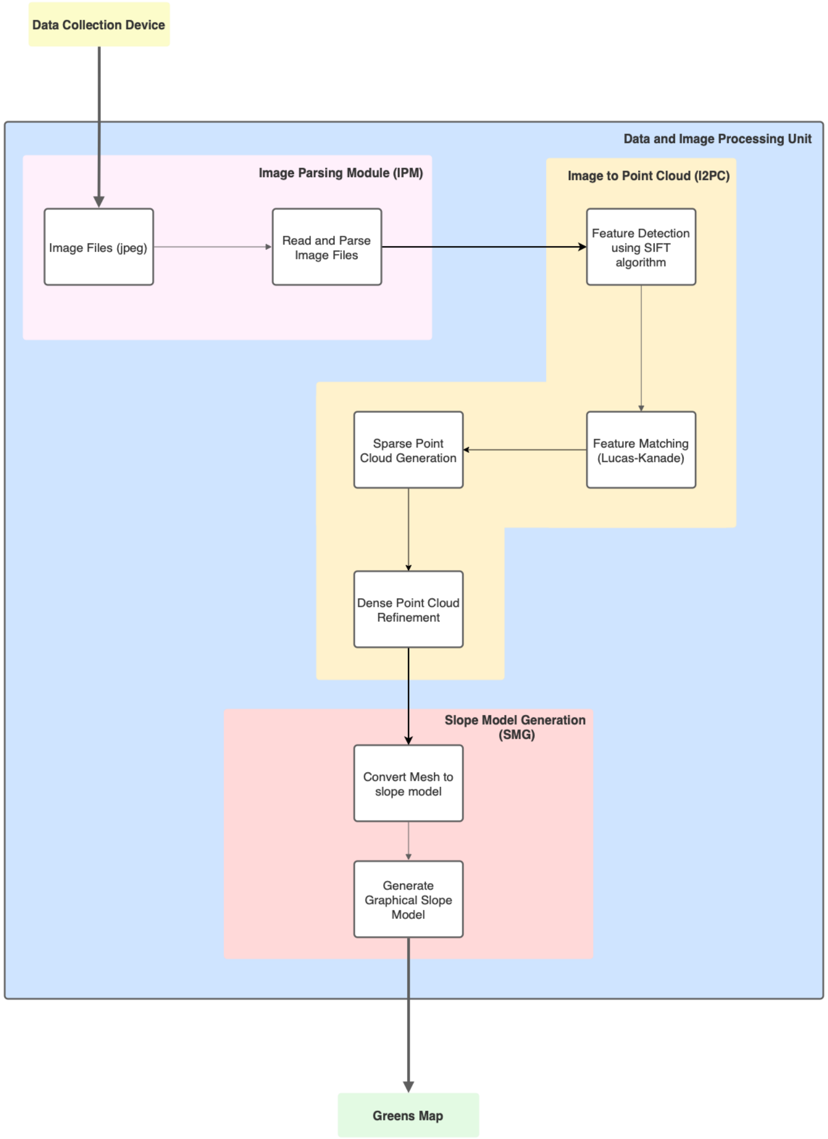


Figure 4.1.1 - Software System Architecture

The software processing task is split into three core components. First is the image parsing module. This module will take in the location of the collected images and corresponding metadata. The module will parse and format the images for the Image to Point Cloud Module. The image to point cloud module will accept the formatted images with their metadata and produce a *Polygon File* (.ply). This file describes the 3D location of each point in the point cloud. The final module, Slope Model Generation Module, takes in the .ply file, produces the slope map of the green and outputs a pdf file to the results folder of the software. Each of these modules are specified in detail in the following sections.

4.2 Image Parsing Module Design

D 4.2.1 - A
The software shall read input images from a variable location on the computer
Related Requirements: R 4.1.2, R 4.1.3
Rationale: When the images are collected and uploaded to the computer, they will be saved to a specific location on the computer. The software will read the images from this folder instead of requiring them to be copied to a secondary location.

D 4.2.1 - B
The software shall combine metadata of the images to be inputted to the I2PC Module.
Related Requirements: R 4.1.2, R 4.1.3
Rationale: The I2PC module will perform the most challenging component of the slope map processing. To reduce complexity, the Image Parsing module will match image metadata with the images instead of requiring the I2PC module to perform this function. The metadata that will be combined with the images is: GPS location, focal length and camera orientation.

4.3 Image to Point Cloud Design

D 4.3.1 - A
The I2PC module shall perform feature detection and matching using the SIFT algorithm
Related Requirements: R 4.1.6
Rationale: SIFT produces a higher accuracy of feature detection compared to other feature detection algorithms such as Speeded Up Robust Features (SURF). Given the general uniformity of the green surface compared to other objects feature detection is commonly used for, a highly sensitive and accurate model is more important than speed. The SIFT algorithm is described in subsection 4.3.1.

D 4.3.1 - A
The I2PC module shall implement the SIFT algorithm using the OpenCV python module.
Related Requirements: R 4.1.6
Rationale: Developing the SIFT algorithm from scratch is challenging and has a high probability of developing less accurate results than previously developed and tested open-source implementations. The SIFT algorithm is described in subsection 4.3.1.

D 4.3.2 - A
The I2PC module will generate the 3D model in the form of a point cloud using the detected feature points using the OpenSfM python module.
Related Requirements: R 4.1.6
Rationale: This module provides both sparse and dense point cloud reconstruction as well as the ability to provide metadata for consideration of images such as gps location, focal length and camera orientation [2].

D 4.3.3 - A
The I2PC module shall output the generated point cloud in the polygon file format (.ply).
Related Requirements: R 4.1.6
Rationale: This file format is standardized and commonly used in structure-from-motion implementations. Additionally, it provides all the necessary information for the slope model generation module to generate the green slope map.

4.3.1 Feature Detection using the SIFT Algorithm

The SIFT algorithm has four main stages. The first stage is scale-space extrema detection. These points are named *keypoints* and are found from taking the difference of successively Gaussian-blurred images and taking the maxima and minima of a Difference of Gaussians that occur at multiple scales [3].

The second stage of the algorithm involves keypoint localization. The previous stage produces too many feature points and ones that are often unreliable. This stage allows the algorithm to reject low contrast points.

The third stage is orientation assignment. The orientations of each keypoint is assigned based on the local image gradient directions. This is the key step in achieving invariance to rotation from camera movement.

The final stage is keypoint descriptor. In order to ensure that *keypoints* can be matched, they need unique description vectors that are highly distinctive based on their calculated local image gradients.

4.4 Slope Generation Module Design

D 4.4.1 - A
The green slope map shall be made using the *matplotlib.pyplot* Contour and Quiver plots.
Related Requirements: R 4.1.6, R 4.1.7

Rationale: The green map design is specified in detail in Appendix A. This library of graphical functionality will enable the creation of the greens map which meets those specifications.

D 4.4.2 - B

The green slope for components of the green will be determined using a least squares error of a fitted 2D plane.

Related Requirements: R 4.1.6, R 4.1.7

Rationale: To determine the slope of a region of the green from a collection of points of the point cloud, the module will compute the least squares error 2D plane of the local point cloud. This will provide an average slope of the region to use for the slope arrow.

D 4.4.3 - B

The slope model generation module will determine the green shape to be shown onto the greens map

Related Requirements: R 4.1.6, R 4.1.7

Rationale: To further assist the consumer in identifying specific regions of the green, the green outline will be needed to better provide context between the slope map and physical world. This will be detected from one or more aerial photographs of the entire green surface.

4.4.1 Slope Model Generation Algorithm

The slope model generation algorithm consists of two main components. The second component (Generate Graphical Slope Model) converts the identified slope vectors of each region of the green into a slope map. The regions of the green are known geographically and the slope arrows are placed onto the map in the corresponding position.

The first component of this module performs the slope vector generation. The first step defines the grid size used for the green. Based on R 4.1.7, the goal is to achieve one grid element per 15cm^2 for the prototype and 5mc^2 for the final product. As the images quality and type improves this grid size will be able to shrink to achieve this precision. To determine the slope of each grid point, only the points within that grid element should be considered, the second stage of this component groups the points into the appropriate regions.

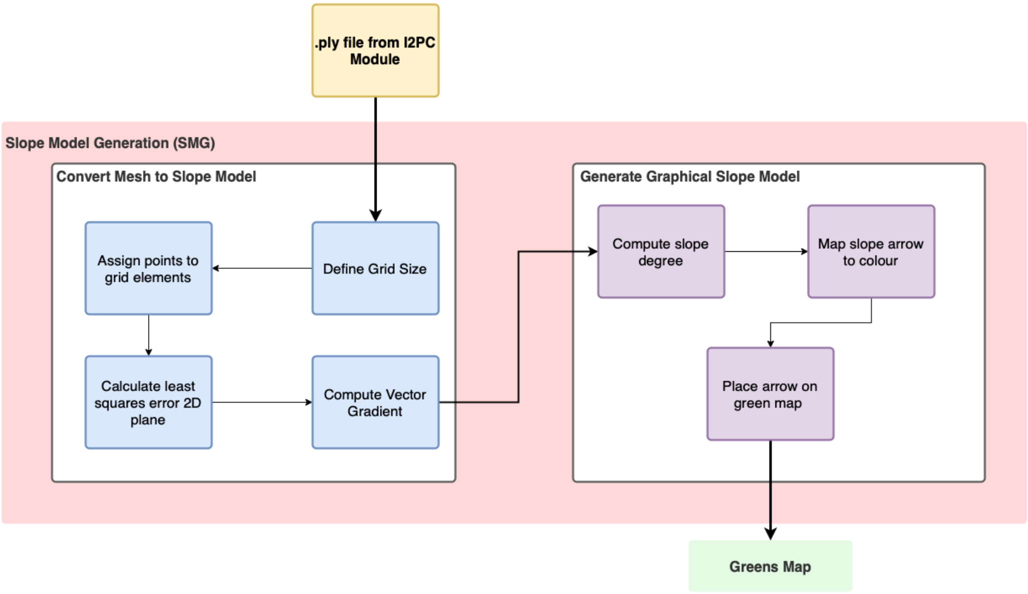


Figure 4.4.1 - Slope Model Generation Flow

To generate a plane for the grid element of the green, a least squares error 2D plane is fitted to the local points. An example of this process is shown in figure 4.4.2. From this plane the gradient vector can be computed and the slope vector easily derived from there.

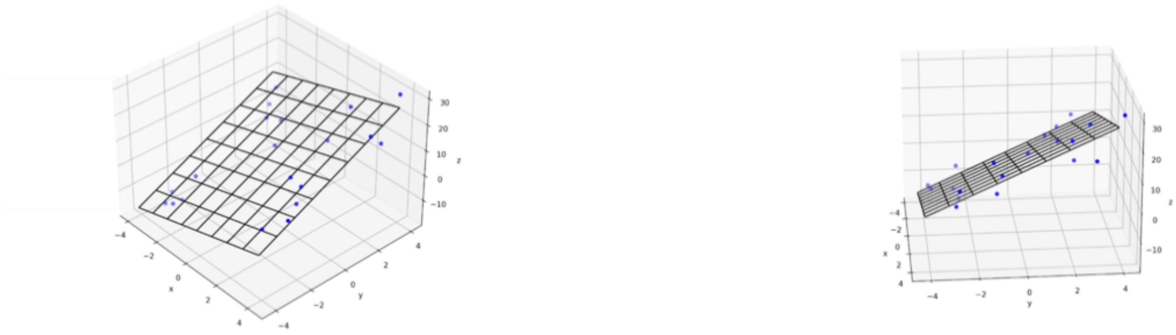


Figure 4.4.2 - Least Squares Error 2D Plane Example

5 Hardware Requirements

5.1 General Hardware Design

In this section, the specifics of the data collection and the rationale behind each design specification is explained.

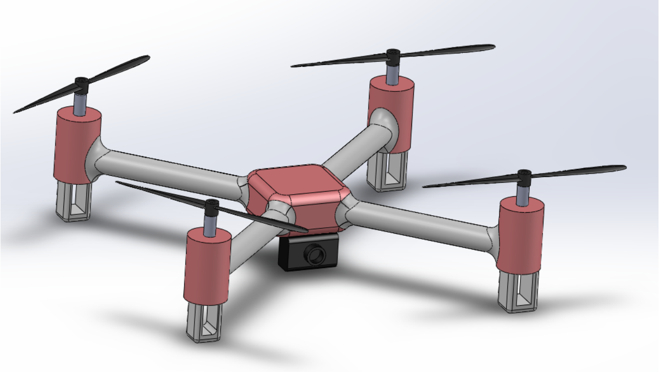


Figure 5.1.1 - 3-D Model of the Drone

D 5.1.1 - A
Raspberry Pi 4B will be used with Navio2 as a flight controller.

Figure 5.1.2 - Raspberry Pi with Navio2 attached to the pins

Related Requirements: R 5.1.2, R5.1.5
Rationale: Using Raspberry Pi and Navio2 as a flight controller was chosen because Navio2 has many sensors such as GPS/ GNSS for location tracking, Dual IMU for orientation and motion sensing, and high resolution barometer which can sense altitude with 10cm resolution. Connecting it to Raspberry Pi increases the function of the drone to control via Wifi, Bluetooth or telemetry. Navio2 is built for autonomous tasks, so it is a perfect fit. Also, it has full-featured open source software providing more flexibility to the product [4].

D 5.1.2 - A

Frame will be at least 30 cm in length and sturdy enough to hold all the electronic components.

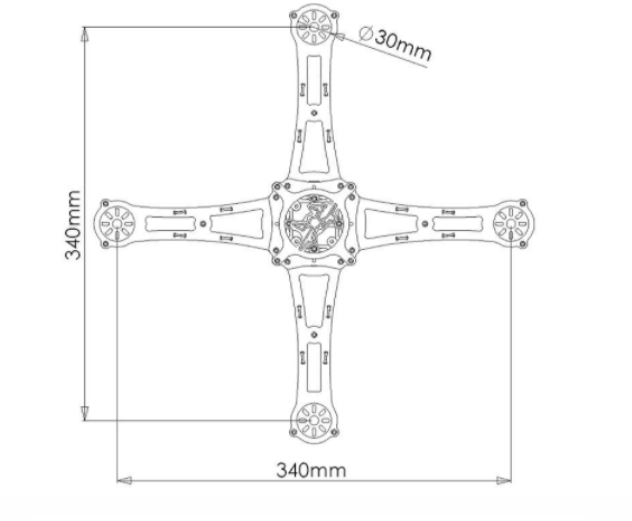


Figure 5.1.3 - Frame Size of the Drone

Related Requirements: R 5.1.3, R 5.1.4

Rationale: A frame is a major component in a drone and it needs to provide enough space to hold all of the hardware components without interfering with the propellers. It will be made with durable material to ensure secure reliable connections.

D 5.1.3 - A

Propellers are 7 inches in length to provide stable flight for image capturing.

Figure 5.1.4a - CW Propellers **Figure 5.1.4b -** CCW propellers

Related Requirements: R 5.1.5

Rationale: The propellers of this size were chosen due to the frame size being at least 30cm in length. Propellers should not interfere with the electronics or other propellers. Longer propellers provide more stable flight, maximizing the propellers size while not interfering with other hardware components is why 7 inch propellers were chosen.

D 5.1.4 - A

Lithium Polymer battery will be at least 5000mah to provide enough flight time to capture green images.

Figure 5.1.5 - Lithium Polymer Battery 5000 mAh 11.1V

Related Requirements: R 3.1.7

Rationale: A battery is also a main component in the drone since it distributes power to all the hardware components. It is important to understand the voltage of the battery, and the weight of the drone. LiPo battery is chosen, because it is light, rechargeable, and cost-effective.

The drone has an approximate weight of 1081g (see table 5.1.1 for calculation) and a 5000 milliamp per hour battery with 11.1V was selected. This will result in a flight time of approximately 15 minutes. This may not reach our intended requirement that the drone must fly 3+ holes, but other limitations had to be considered, namely cost.

Components	No. of Quantity	Mass per quantity (g)	Total Mass (g)
Frame	1	250	426
Battery	1	426	426
Motors	4	52	280
Raspberry Pi	1	45	45
Navio2	1	23	23
ESC	4	14	56
Propellers	4	8	48
Power Module and cables	1	25	25
		Total Mass	1081

Table 5.1.1 - Calculation of the Total Weight of the Drone

Drone flight time was calculated using:

$$Flight\ Time\ =\ \frac{Capacity\ {*}Discharge}{AAD},\ AAD\ =\ \frac{Total\ Drone\ Weight{*}\ Power\ required\ to\ lift\ 1kg}{Battery\ Voltage}$$

- Time - flight time of the drone
- Discharge is set to 80%, because it is never safe to fully discharge the battery.
- AAD - Average Amp Draw of the drone
- Average power required to lift 1kg drone is approximately 170W

Calculating the flight time using a 5000mah, 11.1V LiPo Battery and a total weight of 1081g, it will be approximately 15 minutes.

D 5.1.5 - A

Combined thrust of motors will be at least two times the weight of the drone.

Figure 5.1.6 - Brushless Motor 1000kv 15A

Related Requirements: R 3.1.2

Rationale: The calculation of thrust-to-weight ratio of a quadcopter is essential to ensure the drone can take off and fly stably with total weight. Total weight of the drone is approximately 1081g. For the drone to fly, the combined thrust of motors has to exceed the total weight value, and ideally will exceed at least twice of the total weight of the drone to achieve highly stable flight.

Each thrust of the motors is found in the data sheet provided. Using a 15A motor, the maximum thrust is 920g for each motor. Since the the drone is a quadcopter, the total thrust of motors will be:

$$Total\ Thrust\ =\ 4\ *\ 920g\ =\ 3680g\ >\ 2\ *\ 1081g\ =\ 2162g$$

D 5.1.6 - A

Electronic Speed Controllers (ESC) will be 30A maximum current rating to provide safe and efficient flight.

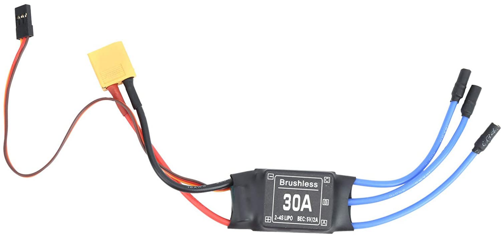


Figure 5.1.7 - Electronic Speed Controller (ESC) with 30A Maximum Current Rating

Related Requirements: R 5.1.1, R 5.2.3, R 5.2.4
Rationale: Electronic Speed Controllers (ESC) control the motor speed as well as directions. The drone will use one ESC for each motor. In choosing an ESC, consideration must be given to the maximum current each motor is supplied.

The motor has a maximum current of 15A and the ESC can take up to 30A. Although higher maximum current ESC could be achieved, that would come with extra weight, size and cost. Thus, higher ESC maximum current rating was chosen to provide lower chance of heating and higher efficiency than the lower ESC maximum current rating [5].

5.2 Electrical Design

D 5.2.1 - A

Flight controller will have enough pins to connect with all other required hardware components

Related Requirements: R 5.2.1

Rationale: Navio is the control piece of the system and controls many hardware components. Navio2 has enough pins to control each motor.

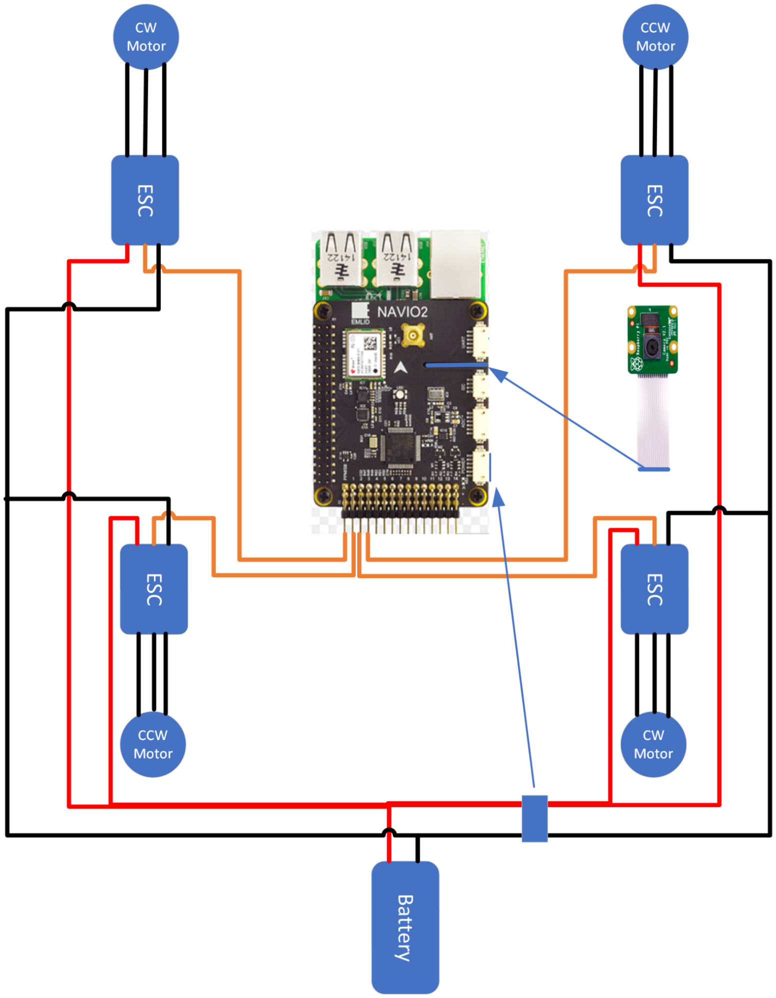


Figure 5.2.1 - Full Circuit Design of the Drone

Figure above is a schematic diagram for all the electrical components of the drone. There should be four motors where two of them are set in clockwise direction and the other two are set in counter clockwise direction. Each motor has electronic speed controllers to control the speed as well as directions. Each electronic controller is connected to GPIO pins on the Navio2 to communicate signals. All the hardware components including Navio2 are then connected to the battery for power source.

D 5.2.2 - A

All connecting points should be soldered properly and have heat shrink between the wires.

Related Requirements: R 5.2.4

Rationale: Connecting points must be durable, so the drone does not fall apart while in a flight. Further, having a heat shrink reduces short circuits.

D 5.2.1 - B

Raspberry Pi with Navio2 will have a shock absorber for accurate positioning and safety during flight.

Figure 5.2.2 - Shock/Vibration Absorber for Navio2

Related Requirements: R 5.1.5

Rationale: Autopilots are sensitive to vibrations and shock. Sensors such as barometers and GPS data can interfere with the drone's position during flight without any vibration damping. Our product will have vibration damping with four shock absorbing balls to provide enough vibration isolation.

5.3 Path Mapping and Controlling the Drone

D 5.3.1 - PC

The Navio2 and Raspberry Pi will connect to the onsite processing unit via Radio Telemetry.

Related Requirements: R 3.2.3, R 5.1.2

Rationale: This choice was made because radio telemetry is a simple and inexpensive way to connect the processing unit wirelessly. The Navio2 has a UART connector that can be configured to communicate with telemetry modules. Telemetry radios are capable of transmitting information over kilometers. This will allow the drone to be monitored from a safe distance.

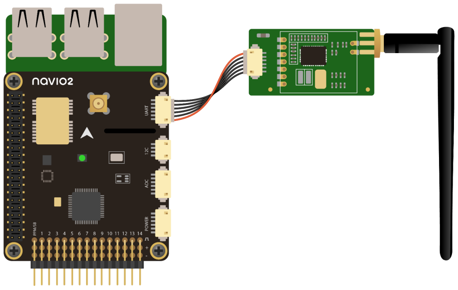


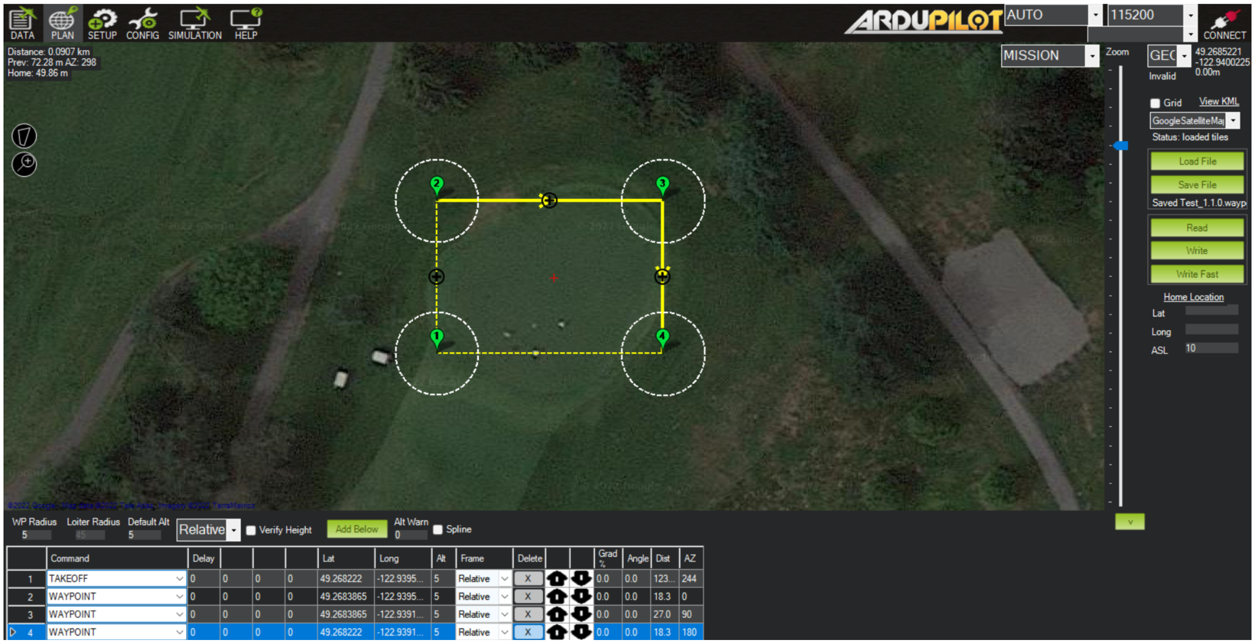
Figure 5.3.1 - Navio2 with UART Connection to Radio Telemetry Device

D 5.3.2 - PC

The onsite processing unit will use "Mission Planner" software provided by ArduPilot to plan the path mapping sequence, and monitor the drone during flight.

Related Requirements: R 5.1.2 (auto), R 4.1.1 (new maps)

Rationale: The system will use Mission Planner because it is free to use and can be easily configured with the Navio2 flight controller. Mission Planner software allows a set of instructions to be created for an autonomous vehicle. A path can be defined by GPS coordinates with instructions to stop and take a picture at every waypoint. Elevation can be defined at each waypoint. During the flight, control and status parameters can be monitored as well as a map showing the position of the drone.



Distance: 0.0907 km
Prev: 72.28 m AZ: 298
Home: 49.86 m

49.2685221
-122.9400225
0.00m
Invalid

Status: loaded tiles
Saved Test_1.1.0.wayp

Lat
Long
ASL 10

WP Radius 5
Loiter Radius 45
Default Alt 5
Relative
Alt Warn 0

	Command	Delay				Lat	Long	Alt	Frame	Delete			Grad %	Angle	Dist	AZ
1	TAKEOFF	0	0	0	0	49.268222	-122.9395...	5	Relative	X	⬆	⬇	0.0	0.0	123...	244
2	WAYPOINT	0	0	0	0	49.2683865	-122.9395...	5	Relative	X	⬆	⬇	0.0	0.0	18.3	0
3	WAYPOINT	0	0	0	0	49.2683865	-122.9391...	5	Relative	X	⬆	⬇	0.0	0.0	27.0	90
4	WAYPOINT	0	0	0	0	49.268222	-122.9391...	5	Relative	X	⬆	⬇	0.0	0.0	18.3	180

Figure 5.3.2 - Mission Planner Software designed by ArduPilot with GPS Waypoints

5.4 Data Acquisition

D 5.4.1 - B
The drone will capture high quality images using a camera as it rotates around the golf green.
Related Requirements: R 3.2.2 - PT
Rationale: In order to capture images of the golf green, the drone with the camera mounted will fly close to the golf green and then fly slightly farther away to take dynamic and a variety of pictures. The drone will rotate a full circle around the golf green while also capturing images from a lower altitude to a higher altitude to capture a wider image at each angle of the circle. The camera resolution will be 5 megapixels, which provides sufficient resolution to construct the slope map.

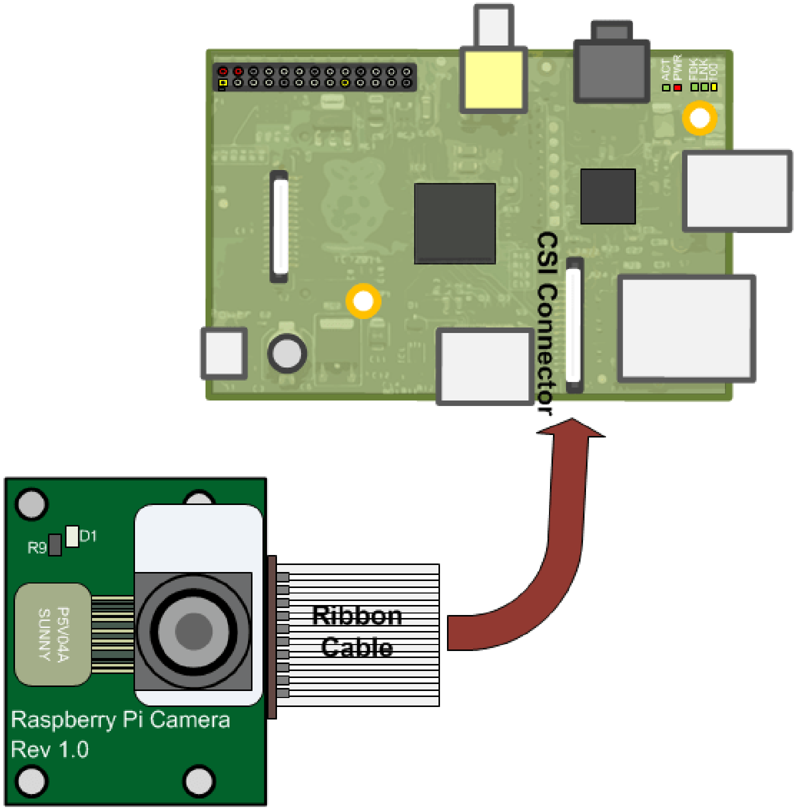


Figure 5.4.1 - Connection Point of the Camera Module and Raspberry Pi

D 5.4.2 - A
The camera will be given a set of instructions to take pictures at each angle around the green as opposed to taking continuous video footage.
Related Requirements: R 3.2.1 - PC
Rationale: The camera, which is connected to the raspberry pi microcontroller will be given commands to snapshot a picture when the drone is stationary at each angle of the circle. Individual pictures are required to construct a digital elevation model, which is used to create the slope map. However, video footage is unable to provide that functionality so the camera will be taking pictures with each command the microcontroller provides.

5.5 Data Transfer

D 5.5.1 - C
The microcontroller will save the pictures to an SD card slot.
Related Requirements: R 5.1.10 - PC
Rationale: The SD card is an easy way to transfer lots of data from the drone to the software processing unit. An SD card slot is provided on the drone and attaches directly into many of today's computers.

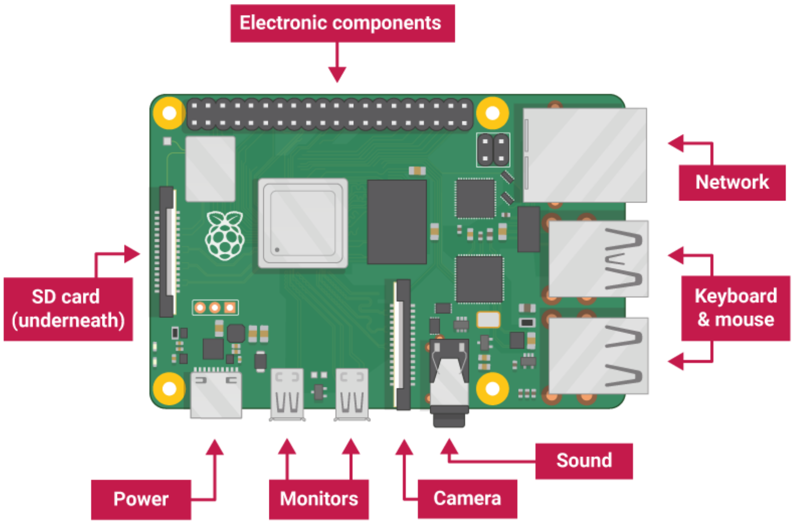


Figure 5.5.1 - Connection Diagram of Raspberry Pi

6 Conclusion

Putt from the sky will benefit golfers of all levels by providing them detailed information of the putting greens they will use on a 2D slope map. Putt From the Sky will acquire and create accurate results directly from the course using a drone maneuvered image acquisition system. This allows the Putt From the Sky team to tackle any terrain that may be found on the golf course to maintain high quality image data. Finally the processing unit will input the images into the image processing pipeline to produce the 2D mapping of the green. The system has been chosen carefully to maximize flexibility of data acquisition to allow Putt From the Sky able to serve the most amount of golf courses.

Building upon Putt From the Sky's Requirements Document feedback, the Design Document has been written in third person with the exception of the Letter of Transmittal. Also, the Abstract was reformatted to provide a concise summary of the document and the design. Further, the encoding of Proof of Concept requirements, Engineering Prototype and Production Version was updated from PoC, EP, and PV to A, B, and C respectively. Justification for each design requirement has been given to support design choices - an additional improvement from the Putt From the Sky Requirements document.

7 Appendix A: User Interface Design

7.1 Introduction

Putt from the Sky is building a service that will create digitized and accurate slope maps of golf greens. These slope maps are provided to consumers in the form of golf greens books which provide the slope information in a domain-specific manner. Collecting and processing this information to generate surface maps will be done by the Putt from the Sky team and will not be a consumer process. The challenge of collecting high-quality and accurate images, as well as the processing time required to produce the slope maps, are major contributing factors in this decision. The design of the system prioritizes functionality along with appropriate safety measures to ensure the system is used as safely and effectively. The system design has also prioritized simplicity of use and intuition in order to allow a new team member to be trained to use the system within one hour as required by requirement R3.1.9.

7.1.1 Purpose

The purpose of this document is to define how the system will interface with the physical world and how it will limit inappropriate uses without sacrificing ease of use.

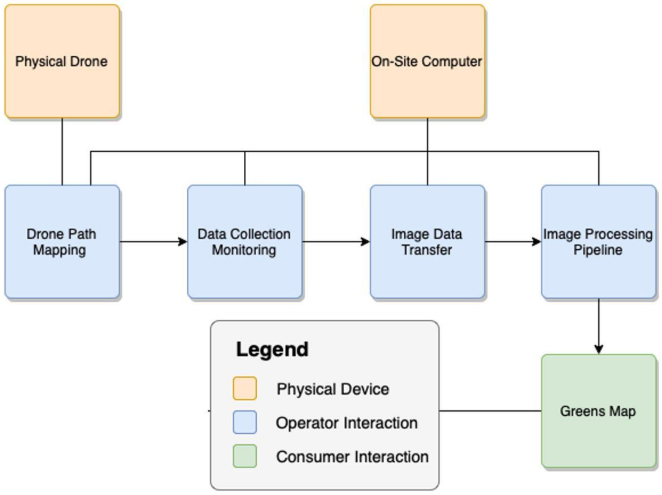
7.1.2 Scope

This document will discuss five main topics, specifically: User Analysis, Technical Analysis, Safety and Sustainability, Usability, and Engineering Standards.

7.2 Interface Design

The majority of the interfaces will be used by employees of Putt from the Sky. The design choices made for each component reflect the abilities of the individual who will use that component. Figure 2.0.1 shows a block diagram visualizing each of the interaction points between an operator and the system. Each of these interaction points are described in the following subsections.

Figure 7.1 - System Interaction Point Block Diagram



7.2.1 User Analysis

This section will describe the interaction of the operator or consumer with the system. Subsections 2.1.1 to 2.1.6 discuss operator interactions and subsections 2.1.7 to 2.1.8 discuss consumer interfaces.

7.2.1.1 Physical Drone

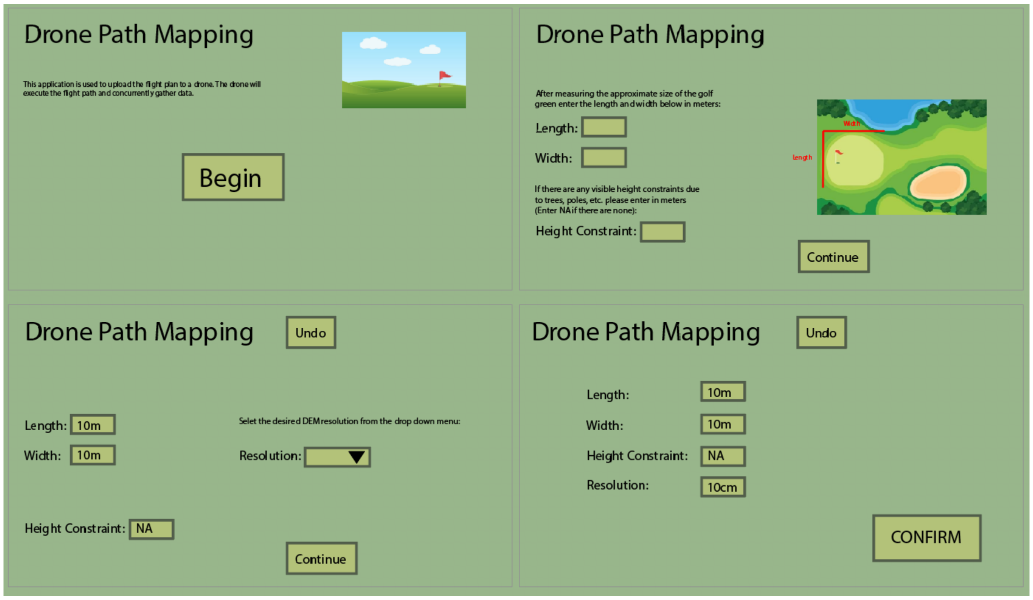
An operator will interface with a drone during image acquisition. With or without prior drone experience, Putt From the Sky employees will be trained on safety procedures and risks associated with flight. For example, ensuring a clear aerial path for the drone is a requirement for the drone to fly. Also, operators will need to be physically capable of transporting the drone across a golf course and be able to keep sight on the drone at all times (not visually impaired).

7.2.1.2 On-Site Computer

The onsite computer is used by an operator to control the data collection process. The software interfaces are described in the following subsections. The computer will run a Windows or Mac OS to rely on operators previous computer experiences.

7.2.1.3 Operator Interaction with Drone Path Mapping GUI

The operator will use the Path Mapping GUI (see Figure 2.1.3, below) to control the pathing of the drone for image acquisition. The employee may have prior experience with similar applications that input data and output a file. Nonetheless, new employees will be exposed to the interface before going on-site. Being a visual interface, visually impaired operators will have difficulties with operating to this interface.



This application is used to upload the flight plan to a drone. The drone will execute the flight path and concurrently gather data.

After measuring the approximate size of the golf green enter the length and width below in meters:

Length:

Width:

If there are any visible height constraints due to trees, poles, etc. please enter in meters (Enter NA if there are none):

Height Constraint:

Length: 10m

Width: 10m

Height Constraint: NA

Selet the desired DEM resolution from the drop down menu:

Resolution:

Length: 10m

Width: 10m

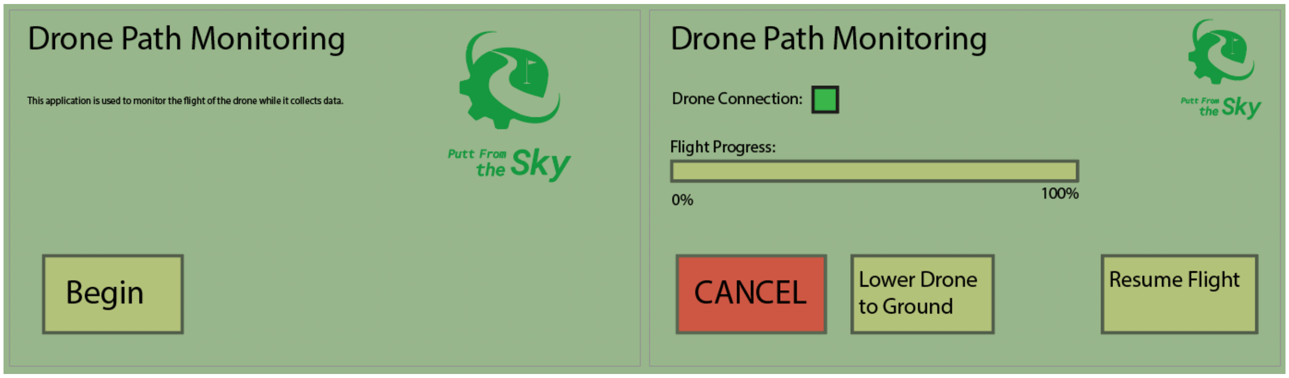
Height Constraint: NA

Resolution: 10cm

Figure 7.2.1 - Drone Path Mapping Interface

7.2.1.4 Operator Interaction with Data Collection Monitoring

As the drone is executing the data collection process, the operator must monitor the area for hazards. A GUI that can pause flight, cancel flight, or return the drone is provided to intervene for these emergencies resulting from those hazards. The operator must understand prior to operation what consists of a safe aerial space to fly and potential future hazards. The operator must physically be able to see the drone at all times and be able to interface with the UI quickly.



This application is used to monitor the flight of the drone while it collects data.

Figure 7.2.2 - Graphical User Interface for Drone Path Monitoring

7.2.1.5 Image Data Transfer Interface

Image data is transferred through an SD card attached to the drone's microprocessor to the on site computer. The operator will physically need to transfer the SD card in a safe manner from the drone to the computer.

7.2.1.6 Operator Interaction with Image Processing Pipeline

The software processing pipeline is executed by an employee using a graphic interface (see Figure 2.1.6). The interface should be familiar to an operator with experience with computer applications and file systems. Selecting an image folder uses the operating system file explorer which should be familiar to operators. Running code and verifying code execution is also a part of the prior knowledge. A graphical interface will make it challenging for a visually impaired operator, but otherwise there are no physical limitations to using this GUI.

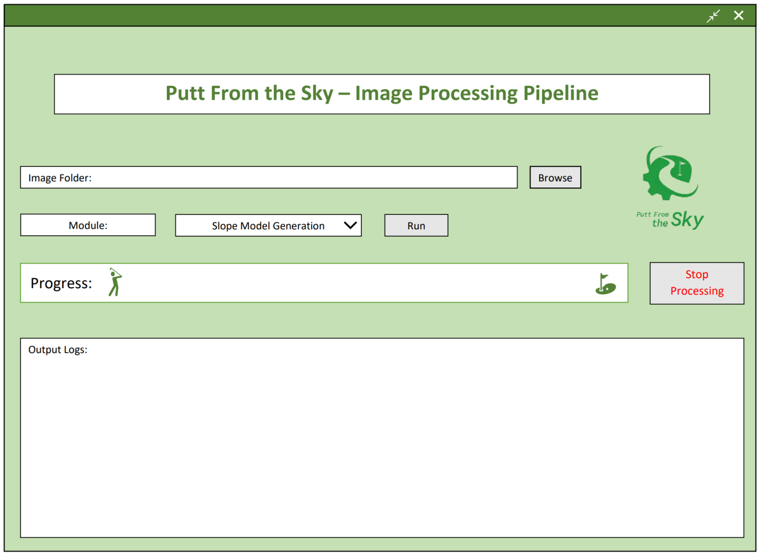


Image Folder:

Module: Slope Model Generation

Figure 7.2.3 - Graphical User Interface of the Image Processing Pipeline

7.2.1.7 Greens Map

The system will output a visual representation of the slope map of a golf green. The map will be available in physical form, printed on paper. This map is intended to be used by golfers, particularly at the competitive level. Given this, there are a few assumptions that can be made about the consumer's knowledge. A description of how to read the greens map and the assumptions of the user are defined in this section.

An example of an extremely high accuracy green diagram is shown in Figure 2.1.7. The Putt from the Sky system will produce a green's map which contains many of the same characteristics, for the purpose of this analysis this figure will be sufficient. In order to read the diagram, the consumer will identify where they are on the green using the grid boxes covering the green surface. Each small grid box represents a 1-yard by 1-yard square of the green (Grid Box Assumption). The consumer will then identify the location of the pin, in the same manner. From that point the user will be able to view the putt in the physical world and on the green map. Each arrow in the map points downhill (Arrow Direction Assumption) and is given an colour to indicate severity (Arrow Colour Assumption). Contour lines are provided on the map to indicate the slope severity and the slope value of each point is also provided beside each arrow. Using this information to understand where to aim the putt requires practice and an understanding of environmental factors on green breaks (Green Reading Assumption). At this point the consumer will know where to aim their putt and they can execute the putt with confidence in their putt read.

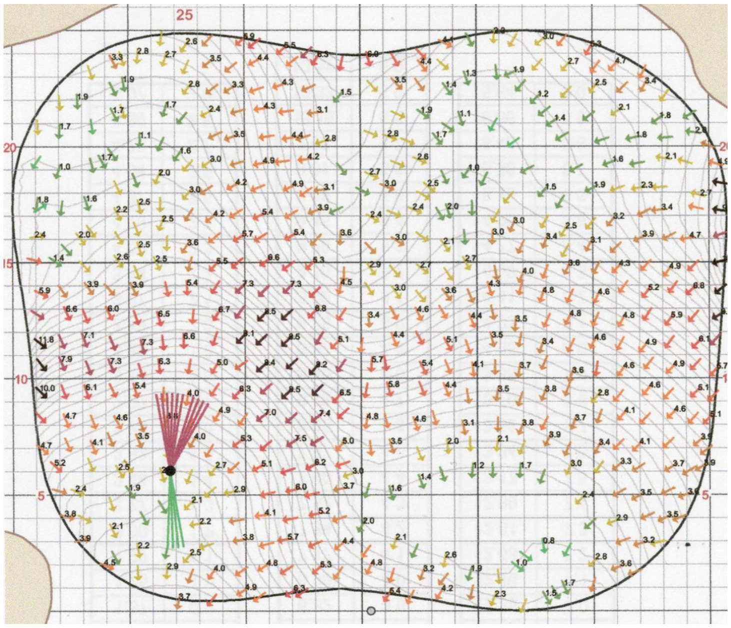


Figure 7.2.4 - High Accuracy Green Diagram with Slopes

Below, the assumptions made to read the green map are explained. For consumers without the prior knowledge required, an example green with explanations will be provided at the beginning of the greens book (a book containing green's maps for all 18 greens of a course). In each of the assumption sections, information included in the example green will be specified.

Grid Box Assumption

In this particular image the major grid lines have been given numbers which indicate their distance (in yards) from the front of the green. Given that between the major grid lines there are five equally spaced boxes the user will know that each small grid box represents 1-yard. The measurement unit (yards) and the grid size will be explained in the example green.

Arrow Direction Assumption

In topography, arrows will always point downhill. Putt from the Sky green maps will also follow this convention. This information will not be provided to the user in the example green since it is a common assumption. Further, since this map will be used in combination with a physical green, the consumer will be able to orient the map to determine the arrow direction if they are unsure.

Arrow Colour Assumption

Green arrows indicate less slope, while red and dark red arrows indicate more slope. Given that the slope values are provided in addition to the arrow colour, the consumer will be able to figure out this relationship. This information will not be provided in the example green.

Arrow Size Assumption

All of the arrows on the greens map are the same size while the colour and numerical slope value indicates the slope severity. It is expected that the consumer will recognize that the arrow size does not provide any slope indication. This is a standard practice in digital green maps. This assumption will be stated in the example green.

Green Reading Assumption

Making effective use of the greens books is a component of golf that requires practice just as any of the other technical skills in golf. Knowing how to use the green map is information that a consumer will learn over time. For consumers without this experience, we will provide an explanation on the example green of how to make basic use of the map.

7.2.2 Technical Analysis

This section will describe the various elements of UI interaction and how each was considered in design choices. The priority of the elements varies between each interface component based on various factors described in the subsections below.

7.2.2.1 Physical Drone

Not all operators at Putt For the Sky will have experience with a drone. Automating the flight path will make the discoverability higher for the operator - moving the focus from controlling the drone to ensuring the safety of the area around the drone. Feedback to the operator about the state of the drone (on, off, low battery, etc) will be communicated through lights on the bottom of the drone.

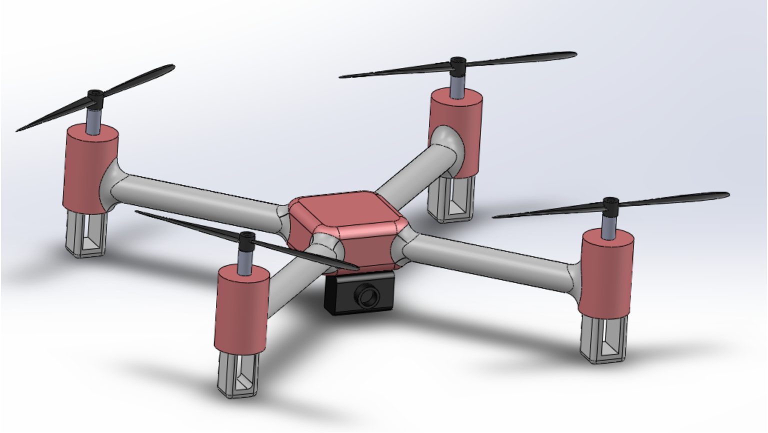


Figure 7.2.5 - Complete 3-D Model of the Drone

7.2.2.2 On-Site Computer

The computer and its OS will not be produced by Putt from the Sky. A technical analysis of the computer and its OS interface will not be provided for that reason.

7.2.2.3 Operator Interaction with Drone Path Mapping

The GUI shown above in Figure 2.1.3 will show what information is required next and how to obtain it. The interface is discoverable as it is designed similar to other computer applications. The interface will provide feedback to the operator through displaying the user inputs on the screen, and requiring the employee to push the "next" button to proceed with the path mapping process. Mappings, including the "undo" button, will undo the user's last input.

7.2.2.4 Operator Interaction with Data Collection Monitoring

The operator is responsible for the safety of the drone, himself and any others in the area. As such the drone control interface must be created in a manner that is easy to understand. The "Stop" button is large and highlighted in red, similar to a common stop sign, to increase discoverability. The "lower drone to ground" button is found next to the stop button because it is most likely scenario in the event of stopping the drone. The "resume flight" is found away from the other options to make the operator consider carefully if the flight can safely be resumed.

7.2.2.5 Image Data Transfer Interface

SD card technology is an interface to transfer data between two objects, for example, the drone and the computer. The 7 elements of UI interaction have been optimized by SD card companies. However, there is a physical constraint of manually disconnecting the SD card from the drone and plugging it into the computer and vice versa.

7.2.2.6 Operator Interaction with Image Processing Pipeline

The GUI for the Image Processing Pipeline (found above in Figure 2.1.6) has been created with a trained operator in mind. The linearly mapped interface is meant to facilitate discoverability. Specifically, each of the four sections: image selection, module selection, process control, and log output are separated vertically from each other top to bottom, as is the cultural norm in English and North America. In terms of operator feedback, the employee would receive visual feedback for the image folder choice, with the path to the desired folder displaying upon selection. The progress bar will provide more visual feedback as it will move from left to right (as is the cultural norm in North America) while the software processes. Finally, the operator will be able to see lower level details recorded in the log output as a final visual cue that the software is progressing.

The module selection drop down menu icon is indicated by a downwards arrow signifier. This is similar to Google Chrome's dropdown tab search feature and is used frequently to indicate a dropping menu. Other signifiers in use are the minimize and exit buttons (top right corner) that are commonly present in computer applications today. Further, mapping the browse button next to the image folder selection indicates to the employee that they can use a familiar operating system GUI to select the desired image folder. Similarly, the run button is mapped next to the module selection tab, indicating that clicking the button will run that module; while the stop button is mapped next to the progress bar as it will stop the progress of the module.

7.2.2.7 Greens Map

The green maps produced by the system are intended for use by a consumer playing golf. The design of the green maps and the overall greens book apply design concepts to this context to help the consumer intuitively understand the information provided.

The greens books are ordered in order of the holes that are played, starting with hole 1 and ending with hole 9 or 18. Each green map has a hole number located at the top of the page to further help the consumer discover the correct green map for their current hole. The green maps are displayed on a to-scale version of the physical green that is being viewed, this helps the user build a conceptual model of the green slopes in the real world. The green's map has grid squares which act as signifiers of the scale of the green on the page in comparison to the actual green size. Lastly, the slope arrows are coloured to provide the consumer with a simple way to read the slopes without needing to read the slope numbers. This mapping is particularly useful for approach shots where specific slope numbers are not as critical as the overall green slope.

7.2.3 Safety and Sustainability

Safety and sustainability are two topics that have been emphasized in the system design. For the data collection process, one drone is used to reduce material needs. The only non-renewable source during image acquisition would be the used battery, which is recharged. In the event of an emergency, it is possible that new parts will be required for replacement. This will be mitigated using safety procedures and the required Transport Canada standards. Over time, the drone's end of life will occur and appropriate e-waste and plastic disposal will be required. Safety wise, the drone controls have an immediate stopping feature as described in section 2.2.4, as well as a pause feature.

For the green books sold to consumers, their cradle-to-cradle cycle will include being created from recycled materials and finally returning to paper recycling at end of life.

7.3 Testing and Verification

7.3.1 Empirical Usability Testing

Since the product that consumers will interact with is a graphical representation of the slopes of the putting green, the testing requirements are dramatically reduced as compared to an interactive device. To ensure the consumers are satisfied with Putt From the Sky's product, the team will assess the quality of the product delivered and inquire feedback from potential customers, specifically competitive golfers. This feedback will be used to improve the final design of the slope map. The team will also conduct research and receive feedback on the operator interfaces of the system. This feedback will come from potential operators who will require training before being subject to the feedback questionnaire.

Questions for Consumers	**Rating Scale: 1 (None) - 10 (Very)**
Challenge to read the green map	
Usefulness of the information provided	
Accuracy was the slope map	
Effectiveness of Slope Colours and Slope Arrows	
Usefulness of example green	
Convenience of accessing greens book	
Questions for Operators	**Rating Scale: 1 (None) - 10 (Very/ High)**
Ease of powering on the drone	

Ease of planning drone flight path	
Usefulness of data collection monitoring UI	
Ease of data transfer from drone	
Quality of the instruction manual	
Overall training provided by the team	
Additional Comments:	

7.3.2 Analytical Usability Testing

The usability of the interface design is key to providing a high quality service. There are two distinct interface groupings within the pipeline, one for consumers using the green books, and one consisting of employees using the data collection and processing pipeline. As such, the testing has been split into two sections.

7.3.2.1 Consumer Analytical Usability Test

The setup for the consumer usability test will include a Putt From the Sky team member as a facilitator of the test and a participant that would potentially use the service. The facilitator will ask the participant to complete the following set of tasks, while providing feedback to the facilitator on the progress and ease of these tasks. These tasks are worded as if coming from the facilitator talking to the participant.

1. You are hitting an approach shot to the 2nd hole green. What page of the booklet do you use to optimize the shot? Why?
2. How wide is the 1st green? How did you get your answer?
3. Placing a ball on green 3, and in a grid box: 4 from the left, 5 from the bottom, which way would the ball roll from this point?
4. How steep is the slope in the previous grid box?
5. Is the arrow an indicator of the slope? Why or why not?
6. Describe the topography of green 15 from left to right.

Each of these questions tests an assumption made by Putt From the Sky in the UI design. The consumer assumptions were grid measurements, arrow directions, arrow colour, arrow size, and green reading. It also tests how quickly a user can find a particular green in the book.

7.3.2.2 Operator Usability Testing

This section will determine how easily a new operator can interact with the system. A facilitator familiar with the system tasks will interview the participant (a potential future employee) to identify problems, uncover improvement opportunities, and learn about the operator's initial behaviors towards the system. The questions are worded from the facilitators point of view.

1. Turn on the path mapping software and create a drone path for green 1 of the Burnaby Mountain Golf Course which has these dimensions - 25x40 yards.
2. Upload the path to the drone.
3. How would you avoid a dangerous situation where the drone is flying towards a person?
4. How would you pause the image acquisition path? How would you resume it?
5. Can you tell whether the computer can communicate to the drone from the interface? How?
6. How much of the flight path has been completed?
7. Can you transfer the collected data to the computer?
8. Process the collected images into a slope map. Describe your thought process through this.
9. Where would you look to see how close the software is to finish the processing?
10. The processing needs to be stopped, how would you do that?
11. Change the image folder for processing
12. Where are outputs from the processing scripts found?

7.3.3 Engineering Standards

The following table shows the Engineering standards applicable to Putt From the Sky. The left-hand column represents the Standard ID given by the following Engineering standards organizations: IEEE, IEC, and ISO. The first 5 relate to device functions, the rest related to UI.

IEC 61800-5-1 Ed. 2.1 b:2016	Adjustable speed electrical power drive systems - Part 5-1: Safety requirements - Electrical, thermal and energy CONSOLIDATED EDITION [2]
ISO/IEC TR 11581-1:2011	Introduces the ISO/IEC 11581 series and provides developers and other icon standards users with an overview of currently available and future anticipated icon standards [3]
IEC 60998-1:2002	Connecting devices for low-voltage circuits for household and similar purposes - Part 1: General requirements [4]
ISO/IEC/IEEE 29119-1:2013	Software and systems engineering — Software testing — Part 1: Concepts and definitions [8]

IEEE 1936.1-2021	Standard for drone applications framework [7]
ISO/IEC TR 20007:2014	Information technology -- Cultural and linguistic interoperability -- Definitions and relationship between symbols, icons, animated icons, pictograms, characters and glyphs [5]
IEEE 1873-2015	Standard for Robot Map Data Representation from two-dimensional (2D) maps to three-dimensional (3D) maps [6]
ISO 9241-161:2016	Ergonomics of human-system interaction — Part 161: Guidance on visual user-interface elements [9]
ISO/CD 9241-115	Ergonomics of human-system interaction — Part 115: Guidance on conceptual design, user-system interaction design, user interface design, and navigation design [10]
IEEE 1621-2004	This standard covers the user interface for the power status control of electronic devices including, but not limited to, office equipment and consumer electronics. Key elements are terms, symbols, and indicators [1]

Table 7.1 - Engineering Standards

7.4 Conclusion

The interface design of the system is a crucial component of the overall system. The system will not work entirely autonomously and requires input from operators. It is important that proper consideration for the design of the system is taken to both reduce training time and reduce operator error when using the system. The analysis of the seven concepts of UI design helps to evaluate the effectiveness of interface design choices and develop a more intuitive and safe system overall. Further to this, given the system’s limited interaction with the consumer directly, it is crucial that the consumer facing interfaces are intuitive and simple. Through empirical and analytical testing of the operator interface and the consumer interface, the system will improve. The system’s proof of concept will lack many of the operator interface elements described in this appendix but the prototype will include these as it is crucial to the system’s overall functionality, safety and ease of use.

7.5 References

[1] “IEEE SA - IEEE 1621-2004 - IEEE standard for user interface elements in power control of electronic devices employed in office/Consumer Environments,” *SA Main Site*. [Online]. Available: https://standards.ieee.org/ieee/1621/3344/. [Accessed: 02-Mar-2022].

[2] “IEC,” *IEC 61800-5-1:2007+AMD1:2016 CSV | IEC Webstore | pump, motor, water management, smart city*. [Online]. Available: https://webstore.iec.ch/publication/25755. [Accessed: 02-Mar-2022].

[3] “ISO/IEC TR 11581-1:2011,” *ISO*, 23-Nov-2011. [Online]. Available: https://www.iso.org/standard/46444.html?browse=tc. [Accessed: 02-Mar-2022].

[4] “IEC,” *IEC 60998-1:2002 | IEC Webstore*. [Online]. Available: https://webstore.iec.ch/publication/4111. [Accessed: 02-Mar-2022].

[5] “IEC,” *ISO/IEC TR 20007:2014 | IEC Webstore*. [Online]. Available: https://webstore.iec.ch/publication/10845. [Accessed: 02-Mar-2022].

[6] “IEEE SA - P2751 - 3D map data representation for Robotics and automation,” *SA Main Site*. [Online]. Available: https://standards.ieee.org/ieee/2751/6994/. [Accessed: 02-Mar-2022].

[7] “IEEE SA - IEEE 1936.1-2021 - IEEE Standard for Drone Applications Framework,” *SA Main Site*. [Online]. Available: https://standards.ieee.org/ieee/1936.1/7455/. [Accessed: 02-Mar-2022].

[8] “ISO/IEC/IEEE 29119-1:2013,” *ISO*, 04-Feb-2022. [Online]. Available: https://www.iso.org/standard/45142.html#:~:text=The%20purpose%20of%20the%20ISO,any%20form%20of%20software%20testing. [Accessed: 02-Mar-2022].

[9] “ISO 9241-161:2016,” *ISO*, 29-Sep-2021. [Online]. Available: https://www.iso.org/standard/60476.html. [Accessed: 02-Mar-2022].

[10] “ISO/CD 9241-115,” *ISO*, 08-Oct-2021. [Online]. Available: https://www.iso.org/standard/80773.html. [Accessed: 02-Mar-2022].

8 Appendix B: Test Plan

8.1 Introduction

This section will outline the testing to be performed on the system as a whole and individual subsystems. All of the tests defined in this test plan will be for a trained operator. A reference will be added for specific user testing which is defined in the Appendix A.

8.2 Data Collection Unit Testing

The following tests will be performed on the data collection unit. These tests will only use system components from the data collection unit.

Test: Drone Component Connections	**Date:**
Procedure:	Each of the below steps will use the continuity testing mode on a multimeter. 1. Verify the Navio2 is connected to the Raspberry Pi as shown in Figure 5.1.2. 2. Verify the Lithium Battery is 5000mAh and is connected to the Raspberry Pi. 3. Verify the camera module is connected to the Raspberry Pi as shown in Figure 5.4.1. 4. Verify the propellers, brushless motors and electronic speed controllers are connected as specified in Figure 5.2.1
Expected Result:	All components are connected as described in corresponding figures
Result:	

Test: Drone Yaw, Pitch, Roll Test	**Date:**
Procedure:	1. Raise the drone 1m off the ground. 2. Rotate the drone 360 degrees 3. Roll the drone 5 degrees left then 5 degrees right. 4. Pitch the drone forward 5 degrees then backward 5 degrees.
Expected Result:	The drone is able to perform the above steps successfully and end up within 1m of the starting physical position.
Result:	

Test: Drone Stability Test	**Date:**
Procedure:	1. Raise the drone 2m off the ground. 2. Apply a wind speed of at least 10km/h in any direction 3. Control the drone to remain in the same ground position
Expected Result:	The drone is able to adjust its hover position to remain stationary over the ground

Result:	

Test: Mission Planning Simple Test	**Date:**
Procedure:	1. Plan a simple test to raise the drone 1m off the ground, hover the drone for 5 seconds and lower the drone to the ground. 2. Upload the flight plan to the drone 3. Execute the flight plan
Expected Result:	The flight plan is uploaded to the drone and the drone rises 1m off the ground and lowers to the ground after 5 seconds.
Result:	

Test: Mission Planning Green Region Test	**Date:**
Procedure:	1. Plan a flight path where the drone will fly in two concentric squares with an area of at least 100m2. The larger square should be 2m larger in width and length. The larger square path should be performed at an altitude of 1m. The smaller square path should be performed at an altitude of 2m 2. Upload the flight path 3. Execute the flight plan
Expected Result:	The flight plan is uploaded to the drone and the drone follows the flight path as described above. (Note any deviation from the expected flight plan in the result)
Result:	

Test: Drone Fall Test	**Date:**
Procedure:	1. Raise the drone to 1m above the ground 2. Cut drone propeller power to cause drone to drop to ground 3. Readjust the drone to an upright position if it is no longer upright 4. Raise the drone to 1m above the ground
Expected Result:	The drone will fall to the ground and remain undamaged so it can continue to function properly.

Result:	

Test: Image Collection Test		**Date:**
Procedure:	1. Plan a flight path to rise to 2m, take a picture, lower to the ground 2. Verify an image was taken and is in JPEG format.	
Expected Result:	The image will be in JPEG format of the intended subject	
Result:		

Test: Green Image Collection Test		**Date:**
Procedure:	1. Plan a pyramid flight path around a region with area at least 250m^2. 2. Take at least 250 images along the flight path 3. Verify the images were all saved on the data storage device	
Expected Result:	The drone will execute the flight path as planned. The drone will take all the images expected and all the images will be available on the data storage device.	
Result:		

Test: Drone Flight Duration Test		**Date:**
Procedure:	1. Verify the drone battery is fully charged 2. Plan a flight test involving at least 5 different altitudes, images and lasting 15 minutes.	
Expected Result:	The drone will execute the flight path and land safely without running out of battery	
Result:		

Test: Drone Flight Pause Test		**Date:**
Procedure:	1. Plan a drone flight path at an altitude of at least 3m. 2. While the drone is executing flight path select *Lower Drone to Ground* on the Drone Path Monitoring GUI	
Expected Result:	The drone will stop execution of the flight path and lower itself to the ground once that button is selected.	
Result:		

Test: Drone Flight Cancel Test	**Date:**
Procedure:	1. Plan a drone flight path at an altitude of at least 3m. 2. While the drone is executing flight path select *Cancel* on the Drone Path Monitoring GUI
Expected Result:	The drone will stop execution of the flight path and hover in place until another instruction is given.
Result:	

Test: Drone Flight Cancel-Resume Test	**Date:**
Procedure:	1. Plan a drone flight path at an altitude of at least 3m. 2. While the drone is executing flight path select *Cancel* on the Drone Path Monitoring GUI 3. Once the drone stops flight path execution and is hovering in place, select the *Resume Flight* button
Expected Result:	The drone will pause execution of the flight path and hover in place until the *Resume Flight* button is selected then the drone will continue its flight path as previously planned.
Result:	

8.3 Image Processing Unit Testing

The following tests will be performed on the image processing unit. These tests will only use system components from the image processing unit.

Test: Image Parsing Module Test	**Date:**
Procedure:	1. Using a subset of the bench test image sample (~5 images) execute the image parsing module and save the result
Expected Result:	The output from the image parsing module will be parsed images with the associated metadata corresponding to each image correctly. These

	images should be verified manually.
Result:	

Test: Feature Detection and Matching Test	**Date:**
Procedure:	1. Using the bench test image sample (~50 images) perform the feature detecting and matching stage of the I2PC module, visualize the results using OpenSfM. 2. Using VisualSFM, perform feature detection and matching on the bench test image sample, visualize the results using VisualSFM 3. Compare the feature points detected in each of implementations
Expected Result:	The I2PC module will produce as many or more accurate feature points as the benchmark from VisualSFM.
Result:	

Test: 3D Reconstruction Test	**Date:**
Procedure:	1. Using the bench test image sample (~50 images) execute the entire I2PC module to produce a .ply file containing the reconstructed images 2. Using VisualSFM, perform a full scene reconstruction and save the output .ply file. 3. Visualize both the reconstructed scenes using OpenSfM 3. Compare the reconstructed scenes
Expected Result:	The I2PC module will produce a scene with more or as much accuracy as the VisualSFM scene. Accuracy will be measured by the percentage of the 3D image that is reconstructed.
Result:	

Test: Slope Module Generation Test	**Date:**
Procedure:	1. Using a full golf green image set, generate a scene reconstruction output file (.ply) using VisualSFM 2. Produce the green slope map using the Slope Generation Module 3. Compare the outputted green map to the green map described in the UI Design Appendix (Appendix A).
Expected Result:	The greens map contains all of the green map details specified in the UI Appendix. These components will be verified manually.

Result:	

8.4 Overall System Testing

The following tests will be performed on the overall system and test both the data collection unit and image processing unit together.

Test: Full System Test		**Date:**
Procedure:	1. Plan a flight path for a golf green. 2. Execute flight plan and save images 3. Upload images to the image processing unit 4. Execute the processing pipeline for the green image set 5. Compare the produced greens map with the physical green surface	
Expected Result:	The greens map produces a to scale slope map of the golf green. The slopes should be accurate to within 2/10th of a degree and can be verified using an electronic level.	
Result:		

8.5 Consumer Testing

The system from Putt from the Sky is not a consumer facing system, rather the system is operated by a trained operator and the output provided to the consumer in the form of a green slope map. Given this, the testing requirements for the consumer fall within the domain of the User Interface testing and as such have been defined there. No additional tests will be given in this section.

9 Appendix C: Alternative Design

9.1 Physical Drone

An operator will interface with a drone during image acquisition. With or without prior drone experience, Putt From the Sky employees will be trained on safety procedures and risks associated with flight. For example, ensuring a clear aerial path for the drone is a requirement for the drone to fly. Also, operators will need to be physically capable of transporting the drone across a golf course and be able to keep sight on the drone at all times (not visually impaired).

Alternatively, a camera on a stick could have been used instead of a drone. With a camera on a stick, taking pictures from high up results in instability as the stick becomes longer and longer. In order to capture stable and high quality pictures at a high altitude, the use of a drone has been selected. An RC car design with camera was also debated, however, in discussions with golf course managers, the greens surface should not be touched by the system operators at any point, if possible, therefore the drone solution became the clear choice to attain this requirement.

9.2 Image Data Transfer Interface

Image data is transferred through an SD card attached to the drone's microprocessor to the on site computer. The operator will physically need to transfer the SD card in a safe manner from the drone to the computer. SD card technology is an interface to transfer data between two objects, for example, the drone and the computer. The 7 elements of UI interaction have been optimized by SD card companies. However, there is a physical constraint of manually disconnecting the SD card from the drone and plugging it into the computer and vice versa. An alternative design would be to wirelessly transfer the data such as: wifi or bluetooth, however, since the service does not need to be delivered on a real time basis, using a physical connection to transfer data is highly effective and reduces the chance of error when transferring data.

9.3 Camera For Data Acquisition

With the goal to construct a slope map, the method of photogrammetry was selected by using a camera as opposed to using a LIDAR scanner. Some of the positives with using a LIDAR scanner includes: it works in the dark and measures the ground using light pulses achieving high accuracy. However, its negatives include being much more expensive than a camera, not being able to easily differentiate the putting green with the rest of the golf course, and that the UV light from the sun can impact the quality. Therefore, utilizing a camera for photogrammetry was selected with its outweighing positives, which include: being more cost effective than a LIDAR scanner, can produce images of the green, and being able to differentiate the putting green between the rest of the golf course. Its negatives include not being able to perform at night, which in this case does not affect the procedure because the service provided is not delivered on a real time basis. A nice day can be selected to produce the images of the green, which a camera can perform great in. Therefore, utilizing a camera with the method of photogrammetry was decided as part of the design to increase functionality and reduce cost.

10 References

[1] E. Peterson, M. Klein, R. Stewart. Whitepaper on Structure from Motion (SfM) Photogrammetry: Constructing Three Dimensional Models from Photography. (2015) pp 2.

[2] "Using," *Using - OpenSfM 0.5.2 documentation*. [Online]. Available: https://opensfm.org/docs/using.html. [Accessed: 12-Mar-2022].

[3] "Scale-invariant feature transform - Wikipedia", *En.wikipedia.org*, 2022. [Online]. Available: https://en.wikipedia.org/wiki/Scale-invariant_feature_transform. [Accessed: 12- Mar- 2022]

[4] “Navio2 – autopilot hat for raspberry pi,” Navio2 by Emlid, 04-Aug-2020. [Online]. Available: https://navio2.emlid.com/. [Accessed: 13-Mar-2022].

[5] “Electronic speed controllers (ESC) for Drones, uavs and Robotics,” Unmanned Systems Technology, 21-Apr-2021. [Online]. Available: https://www.unmannedsystemstechnology.com/expo/electronic-speed-controllers-esc/#:~:text=Electronic%20speed%20controllers%20(ESCs)%20are,the%20speed%20of%20the%20propeller. [Accessed: 13-Mar-2022].