FINAL DEGREE PROJECT

TFG TITLE: Drone configuration for seaside rescue missions
DEGREE: Bachelor's Degree in Air Navigation Engineering
AUTHOR: Jose Miguel Garro Fernández
ADVISOR: Enric Pastor Llorens
DATE: June 15, 2017
Títol: Configuració d’un drone per una missió a línia de costa

Autor: Jose Miguel Garro Fernández

Director: Enric Pastor Llorens

Data: 15 de juny de 2017

Resum

El projecte es basa en la creació d’una eina de primer auxili per un rescat a línia de costa amb un vehicle no tripulat. El sistema estarà format per una part terra, on es trobarà l’estació de terra, i la part aire amb un processador capaç de donar ordres al vehicle no tripulat.

Per als avisos dels possibles ofegaments, la estació de terra rebrà les notificacions que els usuaris hauran publicat mitjançant una aplicació per “smartphone”. Aquesta aplicació permetrà als usuaris capturar imatges de la situació de la possible víctima i enviar-les cap al servidor, que s’encarregarà del seu correcte filtratge cap a la estació pertinent.

El operador serà l’encarregat de la posada en marxa de l’operació. Aquesta operació seguirà eines rutes preestablertes, generades per el processador en aire segons les peticions del operador. Per a la generació de les rutes, primerament serà necessària la generació d’un model d’operació adient per la localització de la missió, aquest model estarà definit per un fitxer amb format “xml”.

La missió constarà de diverses fases, assimilant-se en una gran part a una operació aèria civil, entre les quals destacarà la fase d’escaneig i llançament, on la plataforma cercarà la possible víctima i llençarà un flotador.

El control del vehicle es realitzarà mitjançant una interfície visual, i la comunicació amb el segment aire serà amb comunicació radio, en cas de fallida tindrem una cobertura 4G per continuar l’operació.

L’encarregat de realitzar la missió, serà el vehicle no tripulat que aborda, estarà dotat també per un flotador que es llençarà en una posició pròxima a la víctima per a donar suport. A demés, portarà equipat un pilot automàtic capaç de volar els plans de vol carregats pel processador.

Si durant la missió, sorgeix algun tipus de problema, l’operador podrà sol·licitar una recuperació d’emergència de la plataforma, en el cas de que tot succeeixi amb normalitat podrà recuperar la plataforma, recorrent els passos prèviament volats o de una manera més directe en cas de estar esgotant la bateria.
Title: Drone configuration for seaside rescue missions

Author: Jose Miguel Garro Fernández

Advisor: Enric Pastor Llorens

Date: June 15, 2017

Overview

The project is based on creating a first aid tool to perform a coastline rescue with an unmanned. The System is formed by a ground segment, where the ground station will be placed and an air segment, where the unmanned vehicle, equipped with all the necessary extra devices to perform the mission will be found.

For notices of possible drownings, the ground station will receive notifications that users will publish using a Mobile application. This app will allow users to capture images of the victim location and push them to the server, which will re-direct the information to the station covering that location.

The operator will be responsible of starting the mission. This operation will follow a pre-established route, generated by the processor, according to the requests of the operator. To generate the mentioned routes, first, will be necessary to create an operating model suitable for the location of the mission, this model will be defined with an extended markup language file.

The mission is made up of different stages, similarly to the civil operations. The scan and launch phase stand out among the all steps. In these both paths the vehicle will scan the area to find the victim and deploy the floater once the victim is found.

The vehicle control is done via a visual interface and communication with the air segment will be established by radio. In case of failure, the system will have 4G coverage to continue the operation.

The unmanned aircraft is in charge of performing the mission (controlled by the operator). To operate the mission autonomously, the unmanned vehicle is equipped with an autopilot system, which will perform the flight plans uploaded by the processor. In addition of the radio communication and the autopilot, the vehicle will be equipped with one floater, which will be deployed to give support to the victim.

If during the mission, some kind of problem arises, the operator may request an emergency recovery sequence. Otherwise, in case that everything is working perfectly the platform will be recovered tracing back its previous steps or in a direct recover if the battery level starts to be critical.
For everyone who has believed in me.
CONTENTS

Introduction .......................................................................................................... 1

CHAPTER 1. State of the art .................................................................................. 3
1.1. Evolution of UAS platforms ........................................................................ 3
1.2. Current lifeguard operative for coast rescue ............................................ 7
1.3. Current unmanned systems for coast rescue ............................................. 8

CHAPTER 2. Operating model definition .............................................................. 11
2.1. Introduction of the operation concept ....................................................... 11
2.2. Mission steps ............................................................................................... 12
2.3. Structure of the mission area ....................................................................... 13
   2.3.1. Coastline ............................................................................................... 14
   2.3.2. Land Site Point ..................................................................................... 15
   2.3.3. Entry and exit mission points ............................................................... 16
   2.3.4. Route points ......................................................................................... 19
   2.3.5. Holding points ..................................................................................... 19
   2.3.6. Security Polygon .................................................................................. 20

CHAPTER 3. Drone platform ............................................................................... 23
3.1. Innovations with respect to the current systems ......................................... 23
3.2. Platform ........................................................................................................ 23
   3.2.1. Air-frame ............................................................................................. 23
   3.2.2. Autopilot ............................................................................................ 24
   3.2.3. Payload ............................................................................................... 26
   3.2.4. Weight Budget ................................................................................... 28
   3.2.5. Motor, Propeller and Battery ............................................................... 29

CHAPTER 4. Mission Software ............................................................................ 37
4.1. User connection ............................................................................................ 37
   4.1.1. Mobile App .......................................................................................... 37
   4.1.2. Server decision .................................................................................... 38
LIST OF FIGURES

1.1 Sketch of the Austrian balloon with its different parts tagged. 
1.2 Radio-plane OQ-2 on its launch pad. 
1.3 Aerosonde-UAV. 
1.4 DJI Quadcopter flagship model called Phantom, released in January 2013. 
1.5 Image showing an sketch of the AUXDrone platform manufactured by General Drones. 
1.6 Figure representing the vehicle used by microdrones in the investigation of the mission tactics. 
1.7 Figure displaying the Panthom, manufactured by DJI, and the Ryptide device attached to the vehicle. 

2.1 Simple sketch showing how LifeLine is planned to work. 
2.2 Scan patter used by the platform. 
2.3 Sketch for the launch phase. 
2.4 Simple sketch showing the planned mission flow and the sequences related with them. 
2.5 Castelldefels seashore validation with Google Earth. 
2.6 Landsite, takeoff sequence and hold example in Castelldefels coast. 
2.7 Image showing departure sequence and initial hold with the previous path including landsite, takeoff sequence and hold. 
2.8 Image showing an extract of route points implemented in Castelldefels shoreline and previous paths as departure and takeoff sequences. 
2.9 Image showing departure sequence and initial hold with the previous path including landsite, takeoff sequence and hold. 
2.10 Image showing the security polygon enclosing all the mission paths. 

3.1 Air-frame of the TAROT X6. 
3.2 Flat big plate mounted aboard the platform. 
3.3 Small flat plate which will carry the gimbal and the raspberry. 
3.4 Front view of the Pixhawk 1. 
3.5 Figure showing the Geo-positioning system used on-board the platform. 
3.6 Floating device MRD100 manufactured by Mustang Survival. 
3.7 Image showing the processor Raspberry Pi3 Model B, which is used on-board of the platform. 
3.8 Figure representing the pi camera carried aboard. 
3.9 T-Motor 5208 brush-less motor. 
3.10 Image showing two pair of the Tiger-Motor blades. In the upper part the blade with 16 inch of diameter and 5.4 inch of pitch. In the lower part the 15 inch diameter and 5 inch of pitch blade. 
3.11 Figure where the data obtained from the manufacturer and from the bench is presented. In addition, the three different weights for each battery are displayed. 
3.12 Figure representing the efficiency values for both measuring tools and both blades. 
3.13 Figure representing the values of current consume for both blades and measuring devices.
3.14 Figure representing the values of overall vibration obtained from the testing bench.  
3.15 Figure representing the battery selected. In the image can be seen the constant rate of discharge and the configuration of the cells 6S, the capacity of the battery and the total voltage.  

4.1 General sketch identifying the basic flow between the phone, server and the ground control station.  
4.2 Main screen of the application used to capture the images and send them to the server.  
4.3 Ground Control Form to add a station in the server.  
4.4 Ground Control Form to add a station in the server.  
4.5 Diagram showing the two types of communications protocols established in the platform.  
4.6 Initial main screen display, differentiated in two sections. The section 1, which contains the map, HUD and basic flight button menu, and the section 2, which contains the components for the intended mission.  
4.7 Figure showing the drone icon and the yellow line showing its intention.  
4.8 Button menu for basic flight operations.  
4.9 Emergency button menu.  
4.10 Emergency Dialog for confirmation.  
4.11 Head Up Display, used to control basic information of the flight and about the platform.  
4.12 Messages component, showing the receiving rate frequency of attitude and GPS data.  
4.13 Information showed in the right lower part (below the components), attitude and GPS is presented.  
4.14 Layout of the meteorological component incorporated in the ground station.  
4.15 Layout of the meteorological component incorporated in the ground station, showing how an error is represented.  
4.16 From the left to the right, Wind heading, Current heading, Superficial wave heading.  
4.17 Button menu of the meteorological component showing the light blue color in the Set Manually button.  
4.18 Button menu of the meteorological component showing the dark blue color in the Set Manually button.  
4.19 State of the Mission Manager component after sending the configuration parameters.  
4.20 Specifications of the selected element of services in the list, in that case the Position Trigger.  
4.21 Component showing the operation id at the first leve, the camera id in the second level and the id of the photo taken, in this case just one photo with an id: 0.  
4.22 Track component in which the operator will receive the notifications and commit the properly operation in each moment.  
4.23 Image viewer for the received images coming from the server and taken by the users.  
4.24 Track component in which the operator will receive the notifications and commit the properly operation in each moment.
4.25 Image viewer for the received images coming from the server and taken by the users. .......................................................... 51
4.26 Image showing all the ground tracks received in the control station. .......... 52

5.1 Summary of the information flows when requesting operations from ground. . 53
5.2 Extract of the Maritima Meteo Consult web use to get the needed meteorological information. .................................................. 55
5.3 Figure showing the Stokes drift behavior inside a wave. .......................... 56
5.4 Diagram of the Concept of Operation designed to operate the system. ......... 59
5.5 Figure showing the path of the take off sequence. It also can be interpreted as
the land sequence but in inverse order. ...................................... 60
5.6 Figure showing departure sequence, including the take off path too. ............ 60
5.7 Figure showing the path generated when a request of ToMissionHold to the
selected hold. ....................................................................... 61
5.8 Figure showing two paths. The initial route from the Initial Mission Hold to the
Target To Scan, and then, the search target stage. ............................ 62
5.9 Figure representing the parametrized scan. ....................................... 63
5.10 Flight path when requesting a launch and confirm launch sequence. From the
Holding point of the Scan phase. ............................................. 63
5.11 Figure representing the immediate RTL. ........................................ 64
5.12 Figure representing an example of an emergency land request. ............... 65

6.1 Top view of the 3D solid to show the integration of all the systems aboard. .... 67
6.2 Figure showing an example of the allocation of all the elements in the lower part. 68
6.3 Showing an example of the location of the raspberry and the camera gimbal. .. 68
6.4 Figure exemplifying how the sea is visualized from the camera on board. ....... 69
6.5 Figure representing the water from the camera aboard the platform. ........... 69
6.6 DJI 550 used for the tests in Roda. .......................................... 69
6.7 Figure showing how the victims are seen from the camera on board. ........... 70
6.8 Concept of operation for the test used in Roda. .................................... 70
6.9 Take Off sequence showed in the QgroundControl station. ..................... 71
6.10 QGroundControl representing the drone flying after the take off hold. ....... 71
6.11 Vehicle being displayed in the user interface station. The aircraft is flying towards
the black point. ...................................................................... 71
6.12 Figure showing the scan path in the flight plan. .................................... 72
6.13 Scan patter with the launch path. ................................................. 72
LIST OF TABLES

3.1 Weight Budget separated in parts for the 12.000mAh/16.000mAh/20.000mAh batteries. ........................................ 29
3.2 Table showing for the three different configurations, the cut throttle values, the current consume and the estimated endurance with a 15% reserved for an emergency situation. ................................. 31
3.3 Table showing some values for the operational range of throttle. .................. 31

5.1 Information about wind state given by the AEMET service. ......................... 54
5.2 Information about wave state given by the AEMET service. ........................ 55
5.3 Ratio given by the linear wave theory for identifying the case of study in one of this three groups. .............................. 57
5.4 Table describing every value of the horizontal velocity Equation (5.4). .......... 58
INTRODUCTION

Remotely Piloted Aircraft System (RPAS) are used nowadays for a wide type of applications. From the first road-map created in the United States, to the Spanish legislation created years later, the development of these vehicles has only been growing. Even so, market studies agree that the trend of these platforms is to keep growing.

The most common types of operations currently require the sharpness of a pilot to have the unmanned vehicle properly controlled. That fact implies that much of the success an operation is in hands of the people involved in aircraft control, where the visual and piloting abilities play a big role.

On the other side, focusing on the current operation performed when a maritime emergency is activated, it is known that depending on meteorological conditions and where the emergency happens, rescuers are going to be in a risky situation. For example, close to the breakwater, operation turns out extremely complicated. Consequently, the seriousness of rescue to be performed determines the time factor. Hence, the prompt intervention by the emergency services and the victim life are strongly connected. The loss of body heat, increase up to 25 times more than in regular conditions. In addition, wind and the mood directly affect to the body increasing its heat loss.

Thereby, there are four big problems that lifeguards have to face when performing a rescue, time, meteorological conditions, victim location and current season. These four deals should be overpassed by the created system.

Time, as has been previously said, is one of the most important problems in that kind of mission. Introducing an unmanned aircraft, will drastically reduce the action time. The order of magnitude of the velocities reached by the drone are not comparable with the ones reached by the lifeguard when swimming within the sea. A conventional lifeguard can swim up to 0.88 meters per second, which is considerably below from the 15 meters per second as a normal speed for a drone.

Meteorological conditions are going to be a determinant factor too, because of not all the conditions are feasible for the lifeguards to actuate. This situation could led to increase the time of actuation, if the conditions are in the limit, or if the conditions are that bad, to deny the possibility to the rescuer to actuate. Then, the situation will demand a rescue of the emergency services with boats or vessels.

Therefore, an unmanned vehicle could fly in almost all the weather conditions to save the possible victims on the sea.

On the other hand, the location of the victim will be also a limitation. Some places are going to be unaccessible for the lifeguards, such as the areas close to the breakwater. In that cases, having an unmanned aircraft capable of flying above the breakwater, will be very useful to reach the victim.

If a drowning happens on the winter season, the emergency services will no be available at the most of the coastlines. That fact, turns out a good option to have a system capable of controlling the platform just with a ground control station. Therefore, the aforementioned problems are going to be covered. On one hand, a mobile application has also been designed to allow the people to notify a warning to the ground station, just by taking some pictures and pushing them to a dedicated server. In addition, on the other hand, if the
unmanned aircraft is equipped with a float and the system to be deployed, it will have the capacity of giving support to the possible victim until the emergency boats arriving. To better understand how the devices are going to be integrated, a 3D model made in SolidWorks, containing all the devices aboard the vehicle is going to be shown.

Then, the objective of this project, which has been named **LifeLine**, is to generate the first foundations of a first aid drone system. This vehicle will be controlled by an operator from the ground control station using the developed software for the purpose.

To put in context about unmanned systems development, a brief of the unmanned vehicle history has been explained. Later on, it is detailed how the mission model file has to be created in order to be understood by the software. That model, will be used by the software to generate the necessary routes to perform the operations, which are going to be detailed to better understand how the operator should proceed.

Next on line, the software will be described. Including on it the architecture diagram, and the definition of the data flows among all the components involved in. Then, all the devices that will be carried aboard the unmanned vehicle will be detailed, including an analysis of the motor with data from a measuring bench.

Finally, different test will be performed and described to validate the proper work of all the system.
CHAPTER 1. STATE OF THE ART

1.1. Evolution of UAS platforms

August 22 of 1849 [21], is the first known date of the unmanned aircraft use, in that case by the Austrian army. They used balloons loaded with explosives to attack the Italian city of Venice. The idea was so simple as putting some balloons in a wind flow towards the capital of Veneto region, then with an electromagnetic device attached to a temporizer the bombs would be fired perpendicularly to the ground. Although some of the balloons failed and damaged the Austrian army, some of them worked and successfully harmed Venice.

![Sketch of the Austrian balloon with its different parts tagged.](image)

Figure 1.1: Sketch of the Austrian balloon with its different parts tagged.

After the use of that rudimentary air-vehicles, winged aircrafts started to be developed. The use of that vehicles arose right after World War I. The Ruston Proctor Aerial Target was one of the firsts winged radio controlled aircrafts, which made its first flight in 1916 with a rudimentary radio control. Later on, the Hewitt-Sperry Automatic Airplane, after some improvements driven for military concerns, took off in September 12th of that same year, but controlled by gyroscopes.

Following that tendency, in the early 1930s, the DH.82B Queen Bee a radio controlled aircraft, derived from a modified De Havilland Tiger Moth biplane trainer. It is thought that the term drone started to be used by this time, and also U.S Navy, which was researching in radio-controlled aircrafts, started to use the word drone to identify them.

Soon before the kick off and during World War II, two big parallels branches of investigation
were opened, radio-planes and aerial torpedoes.

In the radio-planes branch, Reginald Denny was one of the prime mover of that research. He and his partners developed a low-cost RC to train the anti-aircraft gunners, named Radioplane OQ-2.

![Figure 1.2: Radio-plane OQ-2 on its launch pad.](image)

On the other hand, torpedoes investigations were also underway, mainly for military reasons, Project Fox could for example.

With the passage of time, radio controlled planes were gaining in relevance, so the investigations during post World War II period, served to develop a new series of piston-powered target drones. As the target drone prospered new kind of applications appeared by this time, such as nuclear tests and reconnaissance platforms, which obviously were focused in military objectives.

Eight B-17 were used in 1946 to collect radioactive data. They were used to gather data from inside the radioactive cloud. In relation to reconnaissance platforms, ADM-20 Quails was the first drone used for this purpose during the 1950s.

In spite of the success of different RC planes, just one spy plane was accessible for the United States in the early 1960s, the Lookheed U-2. But things changed after the shot down of one of the U-2 pilot. There were a fear climate among the pilots, which originated the Red Wagon creation, a highly classified program. During the Vietnam War and successive years, new birds were developed by military services.

While different technologies evolved, people realized how useful were the unmanned aircrafts. As the attitude towards UAV was changing, not just considering these aircrafts such an expensive toys, new researches were focused in achieve higher endurance. These new investigations propitiated the apparition of the new kind of aircrafts.

Be able to maintain aircrafts longer time in air had being a challenge, but it just became an operational reality in the 21st century, although in 1998 the AAI Aerosonde was the first UAV crossing the Atlantic ocean, with a 26 hours flight. Its goal, was to gather data about weather over oceans and remote zones.

Maintaining the same idea as in the Aerosonde, longer times of flight, NASA impulsed a solar-powered aircraft called Pathfinder (1980).

The Pathfinder had two different versions. The initial one, named as Pathfinder, was pow-
Figure 1.3: Aerosonde-UAV.

ered by eight electric motors, firstly powered by batteries but late, when the project was integrated into the ERAST project, solar cells were added covering the extrados surface totally. The second one, called Pathfinder Plus, increased the wing span and incorporated more efficient cells, which were capable of converting 19% of solar energy to power avionics, engines and communications.

Then, with the change of attitude about the unmanned aircrafts, a new meaning of use of that vehicles appeared. Military purposes were still predominant but civil UAS and micro-aerial vehicles (or remotely piloted systems) began to raise.

Nowadays, remotely piloted aircraft system (RPAS) are being used for a handful of applications since its appearance. During the first 20 months (Sept. 2013 - May. 2014) of regulation, the number of authorized operators increased to 1249 and 2241 registered unmanned aircrafts (UA), in view of the State Air Safe Agency from Spain (AESA). In addition, the latest predictions establish a continued increasing trend. The versatility of these platforms offers a diversity of applications that in recent years are being exploited. As different as search and rescue, communications, observation, environmental and lately as couriers.

Most conventional operations are performed with a pilot who is in charge of the aircraft control during the mission time. Regularly, the pilot is on the ground control station with his assistant, better known as pilot on-command. This last one, usually takes care of having an eye on the ground station display to make sure that everything is happening correctly. Optionally, pilot on-command could also provide visual support in case it is required by the pilot, since four eyes allow a better location of the aircraft and greater control over what is happening around it.

This operation is known as visual line of sight (VLOS), whose limitation establish a volume of 120 meters tall and 500 meters long. As an alternative there are two more different operations, extended visual line of sight (EVLOS) and beyond line of sight (BLOS). The procedure to follow in EVLOS mode is quite similar to VLOS. In this case, the pilot in command relies on one or more remote observers to keep the unmanned aircraft in visual sight at all times, relaying critical flight information via radio an assisting the pilot in maintaining safe separation from other aircrafts. Unlike EVLOS, BLOS mode do not need to keep the aircraft in visual line of sight all the time. Instead, the remote pilot flies the aircraft by
instruments from the ground control station.

Figure 1.4: DJI Quadcopter flagship model called Phantom, released in January 2013.

Due to the high number of existing operators, the firms are trying to be focused in one the numerous verticals. The audiovisual, emergency, agricultural, and communications branches are examples of the infinity number of applications that can be performed with an UAS.

Consequently, there are numerous types of frames in the market, as happens with the number of applications, but most popular ones are quadcopters, hexacopters, and octocopters.

Quadcopters are by far the most popular multicopter on the market. These copters use four motors with its propellers to lift up into the air. The most obvious benefit is that with just four motors the weight is reduced considerably, so this fact allows for more payload or overall weight to be added, as an extra battery or a little video camera. This kind of devices can carry up over 400g of payload. In addition, this kind of platform are easy to control and can fly reasonably at a high speed with a decent performance. Its average flight time can be up to 28 minutes, depending on the battery and payload. In terms of price they are relatively cheap because of its accessible manufacture.

Hexacopters are a step up from the quadcopter. These models have six motors and its corresponding propellers. This extra power adds more benefits to the device maintaining pros of the quadrotor. Obviously, in terms of power these models have a great value and reach higher speeds due to the extra two motors. This fact, let hexa reach higher altitudes and carry out heavier payload than previously model. As a safety aspect, having six motors let the platform maintain its flight if one of the motors die, by compensating with the rest of propellers the lack of power. Its flight time will vary depending the carried battery or batteries and the payload, but the average time is between 15 to 20 minutes. Even though having this numerous of pros, these kind of platform have two important drawbacks. The size is increased in comparison with the quad, and now is not as maneuverable in tight sites. Consequently to all of this advantages, the price is elevated than in the previous platform.

On the top of this three models, it is found the octocopter. This archetype shares all the benefits with the hexacopter but with even more power. Higher speed, better control in worse weather conditions, and the capability of lose two or three motors and being capable to maintain the flight. But providing this features is synonym of bigger size and higher price. In terms of flight time, this kind of platforms are very similar to the hexa-propelled vehicle.
1.2. Current lifeguard operative for coast rescue

Not all the coats are equipped with the necessary facilities to establish a control tower for rescuers. Another important factor is the current season too. Normally, during the winter there is no human rescuers, unlike in the summer season, when due to the high number of swimmers is required.

Imagine the case of being in summer season, there are available the needed facilities and someone is in an emergency situation in the sea. At the time that the rescuer notice this circumstance starts the rescue operation. Then, the lifeguard shall go directly to save the victim.

The time that will take the rescuer to reach the victim depends on the sea state and where the victim is located, because is possible that the person would be located in an inaccessible spot for the lifeguard. According to the exam to be selector for the search and rescue team, the requested time to swim 400 meters is about 7 minutes and a half. Hence, if the distance round about 500 meters from the sand, the rescuer will take between 0-10 minutes to reach the location of the possible drowning and save the victim.

If the location is non accessible for the lifeguard, the emergency boats will be in charge of accessing those areas in which the rescuer can not individually. This will increase the operation time considerably, prolonging the distress of the victim.

Thence, victim life will depend on prompt intervention by the emergency services. This fact, such as has been emphasized before, is even more important when analyzing the human body behavior within water. Inside water, the amount of heat loss is about 25 times more than in regular conditions. Besides that, the wind presence, the victim’s nervous mood and the loss of effectiveness from the wet clothes to maintain the body temperature stable, will induce an increasing probability to suffer an hypothermia. Since the studies have determined that a person could resist between 1 to 6 hours when the water temperature is about 10 to 15 degrees, which it coincides with the mean temperature of the Mediterranean sea during the winter.

Figure 1.5: Image showing an sketch of the AUXDrone platform manufactured by General Drones.
1.3. Current unmanned systems for coast rescue

Nowadays there are several platforms remotely piloted and equipped with cameras for the surveillance of the coast.

Due to the increasing number of platforms and the wide number of applications, the security groups have not been left behind. In this case the police of Cunit, a small town of Tarragona, has incorporated the use of drones for the surveillance of the coastlines, but piloted by a licensed police for it and endowed with a camera that transmits the signal in real time to be checked by the same pilot [9].

Figure 1.6: Figure representing the vehicle used by microdrones in the investigation of the mission tactics.

Entrepreneurs endeavors as the AUX Drone project [refauxDrone] impulsed (review Figure 1.5) by the Spanish business General Drones. That project is powered by a quad-copter equipped with two floats that swell when they touch the water. Moreover, this solutions carry aboard a typical visible spectrum camera, being able to carry up an infrared camera in future versions. According to the information available in their web, that system can be flying up to 30 minutes. That device is manually piloted by a pilot from the ground.

Located in Germany and founded in 2005, the firm named microdrones [5] is investigating a new water rescue tactic. Basically their platform (review Figure 1.6) is deployed after receiving a drowning notification and piloted by an specially trained lifeguard, as another rescuer reach the victim by water. Due to the quickly operation of the drone it drops a compact flotation device that automatically inflates after touching the water, then the victim can grab the device while waits to the arrive of the lifeguard boat.

DJI, the big company when talking about micro-aerial vehicles created an accessory for their flagship vehicle, the Phantom. The device is named Ryptide (see Figure 1.7) and is a life saving attachment that can carry an automatic inflater to preserve the victim life. This device only weights 450 grams, and can be set up in less than 1 minute [8]. The phantom with this device aboard have a range of approximately 1 kilometer. The company just have built 5 prototypes, 3 of them being tested in Wisconsin, Louisiana and Florida. Specifically
Figure 1.7: Figure displaying the Panthom, manufactured by DJI, and the Ryptide device attached to the vehicle.

the one beta tester located at Florida, is the one dedicated to a beach rescue during a riptide. This beta tester is formed by a group of students from a private High School in Stamford. As further innovations they want to try the device in a larger drone to implement 4 life rings, controlling the launch for the 4 floats with a code the microprocessor Arduino.

Perseo project [7], managed by a Chilean company, was created for saving lives. Basically their idea is to serve as a float dispatcher. The platform used is an octocopter, equipped with GPS, a GoPro camera and a LED lighting in case is needed. According to their information, the drone will take off from the base, carrying aboard 3 floats that will be dropped near to the victim, in addition an audio system is incorporated if necessary orders are required. The camera carried on board, can livestream its video to the base to be used by the operators.
CHAPTER 2. OPERATING MODEL DEFINITION

2.1. Introduction of the operation concept

The idea of this project is that anyone who sees a possible victim on the beach can notify the emergency services.

![Figure 2.1: Simple sketch showing how LifeLine is planned to work.](image)

For this purpose, a mobile application will allow to take photos of the location of the victim and send them to a central server (see Figure 2.1, where there are two coastlines, “P1” and “P2”, with some mobile phones displayed pushing the info to the central server). That last mentioned, through the geo-location of the telephone, will redirect the notices to the pertinent ground stations (review arrow going from the server to the corresponding ground station in the last mentioned Figure).

At this point, the operator in charge of the system will be who is going to begin the mission. In the case, that two ground stations receive an equal notification, a coordination could be carried out between them if necessary.

Thus, the operation will be carried out in a previously limited area, which is going to be described later in Section 2.3., and in the following Section 2.2., a description of the mission steps. In the Figure 2.1a basic sketch to clarify the planned work of the LifeLine service.

Once the aircraft has reached the victim location using the pre-defined paths (see the sketch inside both mission areas in the Figure 2.1), a scan will be performed to search the person in risk. As can be seen in the Figure 2.2, which is parametrized with the height, width and the number of turns demanded.

After locating the victim during the scan, the launch phase will be activated to deploy the float. The float will be launched close to the victim, as the Figure 2.3 shows and in opposite direction to the wave heading.
2.2. Mission steps

After the reception of a possible drowning notification, the operation should start, being the operator in charge of that. The platform has been designed to operate following these steps:

1. The operator begins the mission checking up the telemetry link with the aircraft, this link is critical for the properly functioning of all system. The GPS position, attitude and battery are critical factors, but current meteorological conditions should be analyzed too in order to assure proper environment for flight. That checks will be also done by the software implemented within the system. In addition, models loaded in air and ground will be checked up to confirm that both are the same file.

2. Once the previous values have been checked up by the operator, the pilot on command should execute the take off immediately. At this point the unmanned aircraft will fly toward the take off hold point following the take off sequence defined in the Subsection 2.3.3.1.

3. Right after arriving at the take-off hold point or even before that happens, the operator could commit the departure request to the air segment. That petition will load to the autopilot the departure sequence, see Section 2.3.3.3. This path will lead the platform to the departure hold, the union point with all the route waypoints. To remind this concept review Section 2.3.4.

4. After departure stage, the pilot in ground, should determine the initial mission hold. The operator will have to select which notification the platform is going to service. Then, the software implemented in the ground control station will compute the closest holding point to the target, so the supervisor just will commit the calculated hold to air. This petition will be executed on air, and the platform will fly towards the selected hold.

5. As the sea is constantly moving, the target will vary its position while the unmanned vehicle reaches its position. To solve this problem, the ground station control implements a meteorological component to download information from a server, or manually fill the meteorological data. Thence, this weather report will be used by the
software implemented in the ground control to estimate the target position after the flight time to the initial route point. Then, once arrived or while flying to the holding point, the operator just will have to click to the scan commit button, which will request a scan to the previous computed point.

6. If is necessary, after the scan step it is possible to perform a confirmation search or directly proceed to the launch stage. For the target confirmation the position committed will be the reported by the detection service, in case of the launch, as in the scan phase, the ground station will estimate the coordinates, reducing the work of the operator in a simple click in both cases.

7. If the confirmation stage has been executed, the final command to request should be the launch. The path for the launch will be uploaded to the autopilot and an order to eject the floats, when overflying the detection position.

8. In case from ground is requested, air segment will perform a simple path where the vehicle will overfly again over the launch point to confirm the correct deployment, this stage is called confirm launch.

9. After the launch or the confirmation phase, the platform will automatically recovered re-flying the same path but inverse order.

Then, to clarify the relationship between the mentioned stages and the sequences designed for the intended mission, a brief sketch (see Figure 2.4) had been created with the planned mission order.

![Figure 2.4: Simple sketch showing the planned mission flow and the sequences related with them.](image)

**2.3. Structure of the mission area**

Classically aircrafts have used pre-established paths to perform their operations in an organized and controlled way. These airways have helped and will continue aiding the controllers to safely manage the traffic flows, either in different kind of operations or at altitudes levels.

A typical airspace track is formed by several points, much of them following a simple path, as the ones defining en-route. But in some cases, complex routes are required to carry out holding patterns, departure and arrival ways for example. In addition, airways are
not defined randomly. These pathways are created taking into account several effects when these routes overfly inhabited zones, as acoustic, atmospheric effects and security reasons. This basic structure is the starting point for defining pathways of the coastline model, in order to protect the swimmers and create an standard operative for search and rescue at the seacoast.

The points that will be used by the software are defined previously and stored in an extended markup language file. This kind of format let easily organize by tags all the model information, which are going to be detailed in the following different subsections. These tags will be parsed by the software and used in the model manager. To see a retailed example of the following descriptions in the Section A.0.1. inside the Appendix can be found one on the Castelldefels coast.

All of the next extended markup language structures must be placed inside the LifeLine tag as can be seen in Listing 2.1, an attribute called name should be specified in order to identify the coast. A second attribute called missionId should be specified, that number needs to be an integer and equally defined either on ground and air, because that integer will be used to check if both models coincide before the platform start. Finally, the third one named checksum, will be used to control that the file is consistent, indicating the number of bytes that software will check up both, in air and ground. These last two, fundamentally are used for assuring that both segments are working with the same models.

Listing 2.1: Definition of the main tag for the model.

```xml
<Lifeline name="modelname" missionId="" checksum=""
    --Structures must be placed here--
</Lifeline>
```

### 2.3.1. Coastline

An important definition for the mission model is the seashore delimitation. Coastline is going to be used for safety reasons in the operative, mainly to prevent the aircraft from flying over the beach and endangering swimmers. Furthermore, as a second reason, when an emergency land is required because a projection of the current position on the seashore will be calculated for a direct path to the sand.

So as to obtain littoral coordinates, the only necessary thing is to select several points defining little segments which outline the coast contour.

#### 2.3.1.1. Coastline XML Tag definition

Once the strand is defined, this information shall be written in the extended markup language file within the corresponding tag and with the properly format. In this case, the identifier for seashore points is coastline. For children inside that tag, latitude, longitude and altitude will be specified following the same organization as is shown in the Listing 2.2 in Page 2.2. The tag position contains the information of one point, with its coordinates and altitude. The location shall be in decimal degrees and altitude as an integer number.

Listing 2.2: Example for Coastline definition in the XML file.

```xml
<coastline>
    <position latitude="" longitude="" altitude=""/>
</coastline>
```
CHAPTER 2. OPERATING MODEL DEFINITION

After selecting the points, which are going to outline the shore limits, the chosen coast will be illustrated similarly to the figure 2.5, in the page 15.

![Figure 2.5: Castelldefels seashore validation with Google Earth.](image)

2.3.1.2. CoastLine heading definition

In order to have information about the coastline heading, an unique tag have been defined for it. The tag name decided to use is `coastLineHeading` and an attribute, called `value`, to establish the heading from the sea to the shoreline. To clarify the explanation Listing 2.3 illustrates an example.

The coastline heading should be specified in **decimal degrees**, for example 29°.

Listing 2.3: Example for Castline heading definition in the XML file.

```xml
<!-- Main heading of the coastline looking from the sea -->
<coastLineHeading value=""/>
```

2.3.2. Land Site Point

Selecting a good point to establish the base for launching and landing the platform is an important decision. Depending on the selected coast to implement the platform this site will vary, but is important to chose the closest one to the sea entry/exit area.

2.3.2.1. Land Site XML Tag Definition

Land site in the XML file shall be defined with the following conditions:

- Tag name: `land`
- Identifier (id) : `land`

As in the coastline coordinates, the values shall be in decimal degrees and altitude as an integer. Listing 2.4 shows an example as how to implement the point.

Listing 2.4: Example for Launch Point definition in the XML file.

```xml
<!-- Land Site - RTL Landing point - Mandatory -->
<land id="land" latitude="" longitude="" altitude="" />
```
2.3.3. Entry and exit mission points

In order to access and return from the maneuvering area, the entry area destined for emergency boats and leisure craft has been the election, in favor of avoiding risky situations due to the aircraft flight above human beings. This area will be used in takeoff, departure and the sequences for going back to launch point, from the end of takeoff and departure respectively.

As in the coastline Subsection (2.3.1.), different tags identify the operations previously named, which are going to be enumerated in the following subsections. For the points that define sequences, *waypoint* is the tag selected for it, unlike the used to define the shore.

2.3.3.1. Takeoff XML Tag Definition

Takeoff is an important operation that later on will be explained in more detail, but first, is essential to define it correctly.

This operation must be stated with the following mandatory conditions:

1. A takeoff hold shall be defined close to the shore, but above the water for safety reasons.

2. The takeoff sequence have to contain at least one point (without including the takeoff hold and landsite), and waypoints must be written in order of operation. This is mandatory in order to perform a speed change before holding at takeoff hold.

The takeoff hold in the XML (example in Listing 2.5) shall be written with:

- Tag name: *takeoffhold*
- Identifier (id): *take-off*

Listing 2.5: Example for Takeoff-Hold Sequence definition in the XML file.

<takeoffhold id="take-off" latitude="" longitude="" altitude=""/>

The takeoff sequence in the XML must be written within the tag name *takeoffSeq* as is represented in Listing 2.6.

Listing 2.6: Example for Takeoff Sequence definition in the XML file.

<takeoffSeq>
  <waypoint latitude="" longitude="" altitude=""/>
</takeoffSeq>

To give a real structure of the previous concepts Figure 2.6 contains a visual example in the Castelldefels coast, where appear tagged the landsite, the minimum mandatory point for takeoff sequence and takeoff hold.
2.3.3.2. Land XML Tag Definition

Land sequence is opposite to the takeoff operation. This basically is the inverse path stated in the takeoff if no obstacle impede to perform a linear path from the landsite, and just one platform is involved in the operation. But in case of some obstacles or more platforms operating at the same time, the land sequence should differ from the takeoff path in order to assure no possible collisions. As can be seen in the Figure 2.6, the takeoff sequence is rectilinear, so the land sequence is the inverse path to the takeoff path as previously explained. The land path consists in going from takeoff hold (to-hold in the image) to the landiste. In addition, the way this path is defined differs from the takeoff, the tag which identify the land is landSeq as Listing 2.7 shows.

Listing 2.7: Example for Land Sequence definition in the XML file.

```xml
<landSeq>
  <waypoint latitude="" longitude="" altitude=""/>
</landSeq>
```

2.3.3.3. Departure XML Tag Definition

Departure sequence is the following action after takeoff, this one will direct the aircraft to mission’s departure hold, which is going to be connected with all the route points. Connection point has been called Initial Hold, identifying it with the following conditions (example in Listing 2.8):

- Tag name: initialHold
- Identifier (id): ihold

The position of this point is not fixed in distance from the shoreline, but should be placed as far as possible to do not endanger people when performing the mission. To give an idea, this point in the Castelldefels model is situated at about 500 meters from the coastline.
As in previous actions, there are some mandatory definitions to establish this route properly. In order to get a clear idea about how to define this sequence, Listing 2.9 has an example for the implementation into the XML. In addition, Figure 2.7 is showing a realistic model in the Castelldefels beach as before.

1. Departure sequence shall be formed by at least one point, without including neither takeoff nor initial hold as in the past sequences. Furthermore, points have to be written in the operation order.

2. Heading of this route must be as far as possible perpendicular to the shoreline, a 15 degrees of deviation is allowed.

3. Tag required to identify this route is `departureSeq`.

Listing 2.8: Example for Initial Hold definition in the XML file.

```
<initialHold id="ihold" latitude="" longitude="" altitude=""/>
```

Listing 2.9: Example for Departure Sequence definition in the XML file.

```
<departureSeq>
  <waypoint latitude="" longitude="" altitude=""/>
</departureSeq>
```

![Figure 2.7: Image showing departure sequence and initial hold with the previous path including landsite, takeoff sequence and hold.](image)

2.3.3.4. Arrival XML Tag Definition

Arrival path is similar in operation as the land sequence, basically a way back from a point to another. Specifically, this track goes from initial hold to the landsite. Hence, this operation includes takeoff hold and the land sequence, in order to keep the land operation as an unique action in case a problem is detected when the takeoff is performed, without to enter emergency mode. For identifying this operation the tag `arrivalSeq` is necessary (see Listing 2.10).

In addition and unlike in the land protocol, arrival shall contain a minimum of three points. At least one from departure sequence, the takeoff hold and a minimum of one from the land sequence. All of this locations must be defined in order of operation, from first to final point, as the previous sequences.
2.3.4. Route points

Route points are an extrapolation of the classical en-route waypoints for conventional airplanes. This set of locations will guide the aircraft designated for the operation to reach the holding points.

The idea of this coordinates is to outline the coast at a determined distance (the one chosen to locate the initial hold), in order to extend the coverage area to all the sea-line.

Regularly entry and exit points of the beach are situated more or less at the center of it, so if the initial hold is set at the middle, two branches of route points will be needed to cover both sides of the strand.

To define a branch next rules must be accomplished:

1. The tag to identify the route points have to be `coastLineRoute`.
2. Each branch must be specified in order and the first point shall be the initial hold.
3. Every single location of a branch have to contain an identification, it is recommended to name the route points with their cardinal direction, followed by an ascending number from the point closer to the initial hold until the last branch point.

To exemplify some route points, Listing 2.11 contains a good extract to completely understand the explanation, in addition, Figure 2.8 contains a model of route points represented in the Google Earth application at Castelldefels coast.

Listing 2.11: Example for define one branch of the Route points in the XML file.

```xml
<coastLineRoute>
    <waypoint id="ihold" latitude="" longitude="" altitude=""/>
    <waypoint id="S1" latitude="" longitude="" altitude=""/>
    <waypoint id="S2" latitude="" longitude="" altitude=""/>
</coastLineRoute>
```

2.3.5. Holding points

Similarly to the route points but in order to do not infer with the en-route operation, parallel points have been defined to perform a hold and wait to future request from the operator, while aircraft maintains its position.

To graphically illustrate these points, Figure 2.9 contains a set of route locations with its holding waypoints.

Each of these holding points are going to be a descendant of one route point. This kind of organization makes a variation in the definition of these 'leafs' within the model file. Unlike the previous route and sequences, every hold point shall be stated with its parent following the next statements. To clarify the rules listed below, Listing 2.12 exemplifies the definition.
Figure 2.8: Image showing an extract of route points implemented in Castelldefels shoreline and previous paths as departure and takeoff sequences.

1. The tag to identify the each branch is `coastLineRoute`.

2. Each hold point must be specified in order and the first point have to be its ascendant.

3. To define the parent, identification is the only requirement but the hold point shall be written as all the previous locations. The hold point Id is recommended to be as its parent Id, but including an H between the letter referencing cardinal point and the number identifying the route point position.

Listing 2.12: Example to define one Hold point in the XML file.

```xml
<coastLineRoute>
  <waypoint id="S1"/>
  <waypoint id="SH1" latitude="" longitude="" altitude=""/>
</coastLineRoute>
```

### 2.3.6. Security Polygon

Once all the model is defined, a security area shall be created in order to use it as a geofencing for the platform. The idea of the geofencing is to limit a zone of operation, where in case that aircraft crosses the border of that defined area, it will automatically execute pre-defined actions to avoid the loss of communication link and consequently the control.

To create this area, the polygon shall enclose all the paths previously defined, in addition the closest side to the sea-line should be prudentially separated from the sand, about 150 to 200 meters where the probability to have a drowning is higher than have it near the sand. Figure 2.10 contains an example implemented in the Castelldefels region.

As previous structures there are some mandatory rules to properly create the file, in that case the needed tag to enclose the polygon points is `securityPol`. Similarly to the coastline definition, each of the polygon’s vertex are going to be defined with the same format, the `position` tag containing `latitude`, `longitude` and `altitude` attributes (see Listing 2.13).
Listing 2.13: Example for defining the security polygon in the XML file.

```
<securityPol>
  <position latitude="" longitude="" altitude=""/>
</securityPol>
```

Figure 2.10: Image showing the security polygon enclosing all the mission paths.

2.3.6.1. Units

In order to have a consistent mission model, it is important to make the units clear. That information has been aforementioned in previous sections in this chapter, but as the units play a big role, they will be clarified again:

- Angles: defined in decimal degrees ($29^\circ$).
- Coordinates: expressed in decimal coordinates or decimal degrees (latitude: 43.256, longitude: 2.345).
- Altitudes: integer number in meters (altitude: 20m).
CHAPTER 3. DRONE PLATFORM

In order to perform the planned rescue mission, it is important to decide which devices are going to be carried aboard the platform. In this chapter, all the tools and necessary attachments, as the autopilot, motors or the float, which are going to be carried aboard will be described. Therefore, including a brief description of the benefits that this system contributes in comparison with those mentioned in the state of the art, specifically in the Section 1.2.

3.1. Innovations with respect to the current systems

Time is the most important factor when comparing the human operative with the current proposed. As mentioned before, lifeguards will take around 10 minutes to reach the victim at 500 meters from the sand. That factor can be dramatically reduced with the unmanned aircraft, which can move faster than the rescuer within the sea water. Normally, this type of aircrafts can reach speeds of 15 m/s (as an average), that means covering 500 meters in more or less 40 seconds, reducing the actuation time in approximately the 95%.

Apart from that, the lifeguard can not enter to the sea in all the meteorological conditions, some of that conditions can be dangerous for the rescuers. Then, LifeLine could assist the victim without endangering lifeguard life. In addition, not only the weather conditions can avoid the rescuer to work, but remote zones or non accessible areas also influence and the system could replace the rescue personal, just with one operator in ground station.

Unlike the current existing unmanned platforms, LifeLine allows to perform the mission without having a pilot. Only one person who knows the stages of the mission will be needed, then simply with clicks the operator will be able to control the aircraft.

3.2. Platform

One of the essential part of the platform is the frame, in this case the TAROT X6. But not only the air-frame is conforming the platform, because the system depends on several devices to work properly. In the following sections these apparatus are going to be described, including the frame, specially for understanding what does the system carry on-board to accomplish the rescue mission.

3.2.1. Air-frame

This model [18] has a wingspan (motor to motor) of 960mm and a clearance space (between the lowest part of the frame and the floor) of 320mm (see measures remarked in the Figure 3.1), which allows the integration of the floats perfectly.

The frame is made off in a high percentage of carbon fiber and high strength plastic, that reduce the weight drastically.

If it is necessary, the landing gear can be retracted. The frame incorporates a motor in order to do that automatically.
The integrated PCB (printed circuit board) Power Hub with connectors let easily integrate the power module just connecting the ESC (electronic speed control) to the central hub, and the several number of connectors on the lower deck allow to eliminate many messy cables from inside the frame.

Attached to the frame are going to be mounted two big plates, and one little flat plate, as can be both seen in the Figure 3.2, for the greatest one, and the Figure 3.3 for the smallest.

3.2.2. Autopilot

In order to control the flying air-frame is very necessary to have an autopilot. That device will be in charge of managing and guiding the aircraft towards indicated coordinates or among the flight plans created for the mission.

The selected autopilot for this purpose has been the Pixhawk 1 [10]. That one was chosen...
because it is an open source platform with a wide support, and the spread community of the DroneCode. That fact, turns out the Pixhawk a really accessible autopilot in terms of price.

That hardware contains a Cortex M4, a high performance embedded 32 bits processor specifically developed for low-cost devices with a broad range of digital signal control. In addition, it has numerous sensing devices.

- Two 3 axis gyroscopes. Used to know the attitude of the aircraft.
- Two 3 axis accelerometers. Used to known acceleration changes, furthermore by integration, velocity and position variations can be estimated.
- One magnetometer which is used to compute the platform orientation towards magnetic north.
- One barometer. Used to know at which altitude above sea level is the aircraft flying.

Apart from the sensors, the AP includes several connections to allow other type of sensors and devices as I2C, SPI or ADC among them.

The Figure 3.4 shows the connections used for the purpose. The arrow with the identification "Telemetry 1", is where the telemetry module is connected to the autopilot and the element that is going to receive the messages of the routes generated by the processor.
The Power pin, needed to supply energy to the autopilot. The battery will supply the required energy, specifically 5 volts after a commutted regulator.

GPS adn I2C pins allow to connect the external geo-positioning system and the magnetometer respectively.

The Main out is going to connect the autopilot to the motor drivers, in order to supply the controlling signals to the motors.

As a kind of security system, are found the buzzer and the switch. The buzzer as references the name, will sound to inform the situation of the platform, armed or not. On the other hand, the switch is a security button. If the button is pressed, the platform can be safely armed, if it is not pressed, the platform will not arm and will not proceed with the mission.

![Figure 3.5](image.jpg)

3.2.2.1. Geo-positioning system - GPS

So as to obtain the current position of the aircraft in every moment, a Geo-positioning system is required. This kind of sensing device basically receives three different signals from three different satellites. Interpreting the arrival times in terms of coordinates that type of device is capable to compute the aircraft's current position.

The GPS module selected for this purpose is **uBLOXNEO-M8N** (review Figure 3.5), in addition is formed up by a magnetometer too. That new feature will let the autopilot to have an external measure of the magnetic field, making that data more precise than the internal one.

Hence, the incorporation of that module allows in the ground station a control of the position of the aircraft in every moment, remarking that GPS information should be transmitted by the telemetry link and improve the measure of orientation.

3.2.3. Payload

In order to perform the search and rescue mission there are some extra parts to incorporate aboard.
3.2.3.1. Float

One of the most fundamental objects is the float. Initially just one float will be mounted aboard but in further versions an extra one could be implemented. The mentioned float is manufactured by Mustang Survival, and the selected model is the MRD100 \cite{19} as can see in the Figure 3.6.

![Figure 3.6: Floating device MRD100 manufactured by Mustang Survival.](image)

That floating device is activated right after touching the water, in addition its shape is similar to a cylinder which allows to be launched from the air and easily describe an uniform drop to the water. The MRD100 has a weight of 553 grams.

3.2.3.2. Processor

Furthermore, in order to generate automatically the routes aboard and in further versions of the platform to analyze the images in live processing, a processor shall be set up on-board. For this purpose a Raspberry Pi3 Model B (shown in the Figure 3.7) is going to be used, due to its economic price and the compatibility with the Java application. That processor will contain the software, developed exclusively for controlling the system.

The Rpi3 is a 64-bit with an ARMv8 (CPU) of four cores, in addition has a 1GB of RAM. Besides that incorporates Wi-Fi, Bluetooth and as can be seen in the Figure 3.7, one Ethernet port, four USB, one HDMI video output, an audio output, forty GPIO pins, an external slot for SD card and a camera and display interface.

Attached to the processor, is found the camera (see Figure 3.8). As in previous decisions, initially the selected is the Rpi3 version 2 camera, although in further versions of the system a thermal camera could be incorporated. The current sensor is a lightweight device of 8 Mega pixels and 3 grams of weight. That image sensor gives to the system the enough image quality to detect the victim within the water.
3.2.4. Weight Budget

When trying to implement some kind of application with a drone platform, one of the restrictive factors is the battery. Currently, that problem is considered the bottleneck and strongly related with the weight that the platform has to lift off the ground. Then, it is so important to do a weight budget.

The part referenced with the mounted frame also includes the motors (6 units), the wires, and the flat plates. In that case two big ones (to hold battery the float), and one little plate to hold the raspberry and the gimbal for the camera (review Section 3.2.3.2.).

As can be seen in the Table 3.1, a budget for three different scenarios have been done. In these cases, everything is the same except the battery. Three different batteries will be analyzed in the following Section, to compare the different performances.
Table 3.1: Weight Budget separated in parts for the 12.000mAh/16.000mAh/20.000mAh batteries.

<table>
<thead>
<tr>
<th>Part</th>
<th>Case 12Ah (g)</th>
<th>Case 16Ah (g)</th>
<th>Case 20Ah (g)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mounted Frame</td>
<td>3759</td>
<td>156</td>
<td>26</td>
</tr>
<tr>
<td>Propellers (x6)</td>
<td></td>
<td>61</td>
<td></td>
</tr>
<tr>
<td>Drops (x2)</td>
<td></td>
<td></td>
<td>26</td>
</tr>
<tr>
<td>4G module</td>
<td></td>
<td></td>
<td>26</td>
</tr>
<tr>
<td>Battery 12Ah</td>
<td>1510</td>
<td>2044</td>
<td>2519</td>
</tr>
<tr>
<td>Floater</td>
<td></td>
<td>553</td>
<td></td>
</tr>
<tr>
<td>Camera</td>
<td></td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>RPi3</td>
<td></td>
<td>95</td>
<td></td>
</tr>
<tr>
<td>GPS</td>
<td></td>
<td>37</td>
<td></td>
</tr>
<tr>
<td>PX4</td>
<td></td>
<td>38</td>
<td></td>
</tr>
<tr>
<td>Telemetry</td>
<td></td>
<td>44</td>
<td></td>
</tr>
<tr>
<td>Wi-Fi antenna</td>
<td></td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>6308</td>
<td>6842</td>
<td>7317</td>
</tr>
</tbody>
</table>

3.2.5. Motor, Propeller and Battery

In the end, the ones which provides the platform the necessary force to lift off the ground and flight are the motors. For this mission, the engine selected, T-Motor 5208 (see Figure 3.9), were the ones with similar specs as the recommended by the manufacturer, but reducing 23 grams of weight in every single propulsion device. Then, every motor weights 145 grams [12].

![T-Motor 5208 brush-less motor.](image)

These motors differ from the rest of loose winding motor reducing the instant current by a 15% and the load current is reduced by a 5-10%, that fact makes this model to have higher efficiency and much torque.
Apart of the motor election, the propellers which are going to be mounted on the motors have to be selected too. The engine manufacturer recommends two different propellers, both characteristics are described in the next paragraphs.

- T-Motor 15*5 propeller. A blade with 15 inch of diameter and 5 inch of pitch.
- T-Motor 16*5.4 blade. A screw with 16 inch of diameter and 5.4 inch of pitch.

To better understand both specifications, a clarified explanation is going to be done. Diameter, as in a basic circumference is the distance between to points crossing the center of it. Otherwise, the pitch is the distance that a propeller would move in one revolution if it were moving through a soft solid.

Figure 3.10: Image showing two pair of the Tiger-Motor blades. In the upper part the blade with 16 inch of diameter and 5.4 inch of pitch. In the lower part the 15 inch diameter and 5 inch of pitch blade.

The aforementioned blades have different performances even if they are mounted in the same motor. Then, to understand this characteristics, the tables provided by the manufacturer and the results extracted from a testing bench \[13\] will be analyzed.

For this analysis, the thrust, the consumed current, and the overall efficiency of the motor versus the power percentage respectively, is going to be reviewed.

It is important to remark that normally, the motors are going to be working around the 50-60% of throttle, because in case an extra power is demanded, the motor could give it without being into the operational limits. Even in the worst case, the autopilot or the pilot in command could increase the throttle up to 40-50%. In addition, at that operational range the motor is working with its maximum efficiency (see Figure 3.12). The range 50-60% provides the highest values of efficiency, in comparison with the upper values of the throttle).

As can be seen the Figures 3.11, 3.13 and 3.12 (which contain the results from the bench and the web page of Tiger Motor), at first sight the results seem pretty similar between
Table 3.2: Table showing for the three different configurations, the cut throttle values, the current consume and the estimated endurance with a 15% reserved for an emergency situation.

<table>
<thead>
<tr>
<th>Blade</th>
<th>Cut Throttle Values</th>
<th>Current Consume (A)</th>
<th>Endurance (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>15-5 Web</td>
<td>57.99%</td>
<td>30.18</td>
<td>20.27</td>
</tr>
<tr>
<td>16-5.4 Web</td>
<td>52.33%</td>
<td>28.13</td>
<td>21.75</td>
</tr>
<tr>
<td>15-5 Bank</td>
<td>57.77%</td>
<td>32.23</td>
<td>18.7</td>
</tr>
<tr>
<td>16-5.4 Bank</td>
<td>53.16%</td>
<td>30.73</td>
<td>20.01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Blade</th>
<th>Cut Throttle Values</th>
<th>Current Consume (A)</th>
<th>Endurance (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>15-5 Web</td>
<td>60.5%</td>
<td>33.73</td>
<td>24.19</td>
</tr>
<tr>
<td>16-5.4 Web</td>
<td>54.4%</td>
<td>31</td>
<td>26.316</td>
</tr>
<tr>
<td>15-5 Bank</td>
<td>59.99%</td>
<td>35.65</td>
<td>22.88</td>
</tr>
<tr>
<td>16-5.4 Bank</td>
<td>55.19%</td>
<td>34.21</td>
<td>23.85</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Blade</th>
<th>Cut Throttle Values</th>
<th>Current Consume (A)</th>
<th>Endurance (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>15-5 Web</td>
<td>62.74%</td>
<td>36.6</td>
<td>27.86</td>
</tr>
<tr>
<td>16-5.4 Web</td>
<td>56.23%</td>
<td>33.94</td>
<td>30.04</td>
</tr>
<tr>
<td>15-5 Bank</td>
<td>62.14%</td>
<td>39.62</td>
<td>25.73</td>
</tr>
<tr>
<td>16-5.4 Bank</td>
<td>56.84%</td>
<td>37.63</td>
<td>27.10</td>
</tr>
</tbody>
</table>

3.2.5.1. Thrust analysis

In the case of the Thrust, can be appreciated at a glance, that the data for the 15 inch diameter propeller in both devices are almost equal. The same happens with the 16 inch diameter blade. For both measuring instruments in the 50-60% range of throttle, they have almost the same values too, although in the upper values of throttle, the green line (representing the 16 inch diameter blade for the values obtained from the bench), tends to differ from the yellow one. This could be a measurement error from the bench.

On the other hand, if the values of thrust (represented as the kilograms the motor can lift off the ground) are checked in the Figure 3.11, can be seen that the blade of 16 inch is capable of lifting off more weight than the 15 inch blade does. In the Table 3.3 are found the numerical values, for both blades and measuring tools, in the operational range (50% - 55% - 60% of throttle).

Table 3.3: Table showing some values for the operational range of throttle.

<table>
<thead>
<tr>
<th>Blade</th>
<th>50%</th>
<th>55%</th>
<th>60%</th>
</tr>
</thead>
<tbody>
<tr>
<td>15-5 Web</td>
<td>4.59 kg</td>
<td>5.67 kg</td>
<td>6.378 kg</td>
</tr>
<tr>
<td>16-5.4 Web</td>
<td>5.7 kg</td>
<td>6.99 kg</td>
<td>8.31 kg</td>
</tr>
<tr>
<td>15-5 Bank</td>
<td>4.40 kg</td>
<td>5.62 kg</td>
<td>6.84 kg</td>
</tr>
<tr>
<td>16-5.4 Bank</td>
<td>5.49 kg</td>
<td>6.78 kg</td>
<td>8.23 kg</td>
</tr>
</tbody>
</table>
Figure 3.11: Figure where the data obtained from the manufacturer and from the bench is presented. In addition, the three different weights for each battery are displayed.

Figure 3.12: Figure representing the efficiency values for both measuring tools and both blades.
As can be seen in the Table 3.3, the values do not fluctuate too much, however is important to take into account the weight budget to calculate the minimum throttle to fly the platform. To obtain these values, three horizontal lines have been inserted in the thrust graphic to identify the cut values (see Figure 3.11). The Table 3.2 contains the exact cut values.

This way, as can be seen in the Table 3.2, the values of throttle are lower in the 16-5.4 blade than in the 15-5 blade for all the scenarios. Because of the 16 inch diameter propeller is bigger than the 15-5 prop, the first mentioned produces more lift, then it needs a lower value of throttle than the 15 inch propeller to lift off the ground the same weight.

In a logic way, if the 16 inch diameter blade requires a lower throttle than the 15·5 blade, it means that in this case the current consume will be lower compared with the 15 inch diameter propeller. To check that affirmation, the current consume is going to be analyzed.

### 3.2.5.2. Current consume analysis

In that case, as can bee seen in the Figure 3.13, the values differ more than in the thrust graphic, where the values were more similar. In order to compute the medium current for every case, the values of the minimum throttle to lift off the platform in the Table 3.2, have been used in the Figure 3.13 to find the respective value of current to fly. These values are found in the last mentioned table, specifically in the **Current Consume** column.

![Image](image.png)

**Figure 3.13:** Figure representing the values of current consume for both blades and measuring devices.

If the results from both measuring tools are compared, at a glance can be identified that the web is more optimist than the results from the power bench. According to the results, the difference between them, oscillates from 2 to 3 amperes in the worst case.

Then, knowing the average current consumed by the platform is possible to calculate the time that the unmanned aircraft could fly. In the following section is going to be explained how the endurance has been calculated.
3.2.5.3. **Endurance analysis**

Thus, to calculate the flight time of the platform according to the type of battery chosen, it will only be necessary to make the following relation between the capacity and the current consumed (review Equation 3.1).

\[ t \ [h] = \frac{C \ [Ah]}{I\cdot Consumed \ [A]} \]  

(3.1)

Then, by applying this equation and establishing a 15% of emergency reserves, the resultant flight times can be found in the Table 3.2, specifically in the **Endurance** column. These results, differ between them due to the differences in the current consumed, as the flight time directly depends on these mentioned value.

Initially, the 16 inch diameter blade could seem the best option, but according to the vibration values this blade is not a good option, due to the high value of vibration, even tough in the low throttles. On the other hand, the blade of 15 inch of diameter has much lower vibrations than the 16-5.4 blade has, as can be seen in the Figure 3.14.

![Figure 3.14: Figure representing the values of overall vibration obtained from the testing bench.](image)

3.2.5.4. **Battery election**

Finally, after presenting the three different scenarios with three types of batteries. It is necessary to decide which one of the three proposed kinds of batteries, best fit to the purpose.

Currently, the selected battery has been the one with a capacity of 12,000 mAh, which has a best trade-off between time and weight, giving an endurance time to the platform more or less about 20 minutes with a 15% of security margin, which is a correct value for the purpose. In opposition, the battery with 16,000 mAh of capacity gives to the unmanned
aircraft just 4 minutes more of endurance but adding 534g to the payload. Something similar happens with the 20,000mAh battery. This one, weights 1009g more than the 12,000mAh [20] battery providing more or less 10 minutes more of endurance.

This selected battery (see Figure 3.15) contains 6 cells and a total of 22.2V. In addition has a constant discharge rate of 15C, it means that the battery can supply a constant current of its capacity times the rate of discharge. In that case the constant rate is 180 amperes. On the other hand, the peak discharge is 30C. It means that the battery could supply 360 amperes in a peak of time if it is required.

Figure 3.15: Figure representing the battery selected. In the image can be seen the constant rate of discharge and the configuration of the cells 6S, the capacity of the battery and the total voltage.
CHAPTER 4. MISSION SOFTWARE

4.1. User connection

Thinking about in periods out of summer season when the lifeguards do not work, the idea was to create a phone app, capable of notifying the possible drownings by taking some photos of the victim situation. This application requires an Ethernet connection to be able to send the images to a server, which would be in charge of redirecting the data to the corresponding ground control station. This is the basic concept, but now each of these parts is going to be described in more detail.

Figure 4.1: General sketch identifying the basic flow between the phone, server and the ground control station.

4.1.1. Mobile App

The phone application, which is going to be used by the users to notify the warnings will look similar than the Figure 4.2.

Figure 4.2: Main screen of the application used to capture the images and send them to the server.

As can be seen in the last mentioned Figure, the main screen contains three buttons and one red line with the text Horizon over it.
The red line with the horizon text, is located on the screen to help the user to correctly frame the photo, adjusting the horizon of the sea in the horizon line showed in the main screen. In addition, that reference will be useful in order to calculate the estimated position of the victim using the photo in further versions.

On the other hand, the ZOOM IN/OUT buttons, will allow to enlarge or shrink the image if is necessary. Finally, once the image is well-framed in the screen the TAKE PHOTO button should be pressed. This button will take 10 pictures and the app will send them to the server. At this moment, the TAKE PHOTO button will be disabled until all the images have been uploaded, this behavior can be understood as an ACK sent by the server, confirming that the images have been received correctly (as can be seen in the Figure 4.1 between the Mobile Phone and the Server).

For a precise work of the application, the phone should have a network connection (Wi-Fi, 3G or 4G) to be able to connect with the server and send the taken pictures.

4.1.2. Server decision

The server, in order to be capable of re-directing the information to one or more ground stations, needs to have them previously registered.

To do that, the server contains a list where the ground control stations can be registered. In order to sign one ground station, is necessary to complete a form with its different characteristics. As can be seen in the Figure 4.3, the server demands:

- The name of the ground control station.
- The coordinates of its position. The coordinates should be in decimal degrees.
- The radius of coverage. This radius means that all the notifications inside the radius will be notified to the ground station. The radius shall be specified in meters.
- The IP direction of the ground control station.
- The Port, where the information is going to be sent to the ground station.

After completing all the requested data, the control station should be submitted to the server by clicking into the Submit button (review Figure 4.3).

Since that moment, where the server contains at least one ground control station, it is suited to re-direct the information received from the users.

The decision will be made depending on the separation between the position where the user has taken the pictures and the ground station position. If the distance is minor than the registered radius it means that the position where the images have been taken is inside the radius coverage of the station. If the distance is bigger than the radius, the server will not re-direct the information.

Subsequently, processing the request and selecting the station where the images have to be sent, the information is pushed to the ground control station. The info will be transmitted by cable between the server and the ground station, in order to minimize the possible loss of packets.
4.2. Software Implemented

After designing the mission model and establishing the platform and the devices that are going to be carried aboard, the software implemented has to be explained.

Firstly, the software architecture will be detailed. Later on, the user interface created for the purpose will be presented and explained.

4.2.1. Software Architecture

The software architecture is separated in components and services. As can be seen in the Figure 4.4, the designed software has 10 big components, with theirs respectively services.

Apart the components division, that groups are also differentiated depending if they belong to air or ground segment. The information, among all the components in each of both sections are going to be transmitted by a network channel, but in the case of the communication between ground and air, a communication by radio will be establish using the MAVLink protocol to code the messages.

Inside the ground section is found the Ground Manager component. The mentioned element, is in charge of processing all the incoming requests from the basic flight button menu and the track component buttons.

Hence, in the same way that the ground manager does, the rest of the components conforming the ground segment will be controlled by their corresponding managers.

The meteorological component (see Section 4.2.3.2. for more detail about its function), which has the capability of getting a weather forecast (review Figure 4.4 to see ServerData Input to the component) or let the user to set it manually (ManualData Input showed in the Figure 4.4), will process all the user petitions by mean of the graphical interface designed. Moreover, this component is able to send all of the weather information to air (represented
in the architecture diagram as output arrows distinguished with, WindState, WaveState and CurrentState).

In the case of the Ground Track Receiver component, is configured to receive the images pushed by the users into the server (review Section 4.1.), and store the pictures within a memory direction (In further versions, these images will be processed in order to compute an estimated position of the victim). In addition, the component will inform to the mission component the memory direction, where the images have been stored, to be visualized in the user interface (see Section 4.2.3.5. to review the user interface).

On one hand, the Mission Trigger Component is the component in charge of setting up the trigger for the external devices. The configuration will be made by a message named, Mission Operation Setup, and the external device will generate a Generic Request Reply to inform the Ground Station. As can be seen in the diagram, that component will send the configuration message to set up the trigger service, and will receive generic request replies when the trigger is executed.

On the other hand, the Image Available element, will establish a direct communication with the external device controller (such as the camera) set up, to obtain the pictures identifications (see Section 4.2.3.4. for detailed information about the component).
Then, after analyzing the ground segment components, the ones integrated in the air segment are going to be described.

As one of the most important components on air is found the Mission Flow Manager. This element is build up of three services:

- State Controller
- Direct Command
- Mission Manager Service

The Mission Manager Service is in charge of controlling and redirecting the requests to the Generator Component, specifically to the Generator Service. In addition, it is also in charge of en-routing the meteorological information generated on ground to all the elements which do need it.

The basic function of the Direct Command service is to send the navigation commands such as arm, take off, loiter, go to or return to land. To send that commands, the services needs to know either the ground requests and the ACK from the Mavlink Component, in order to decide which of the mentioned actions has to be demanded.

By last, the State Controller service is capable of informing the operator, which is the life-line flight mission being flown by the drone at every moment (review Section 5), controlling the index of the navigation commands conforming the flight plans.

Following with the air components, there are found two components that work strongly connected. On one hand, the Generator service (found inside the Trajectory Generator), is capable of decoding what kind of request has been demanded from ground. Thus, in turn, the Generator service will demand to the Route, Scan or Launch generator the needed trajectory.

- The Route service, is in charge of generating the necessary route from the take off to the mission hold, and all the available recovers.
- The Scan service, is able of creating the scan stages to search the possible victims (see Section 5.3.5. for more information about the search).
- And the Launch service, with the meteorological information received from ground, will generate the pertinent route (review Section 5.3.6. for the Launch path) for a correct deployment of the floating devices.

Once the software has generated the flight plan the last step is to push that list of waypoints to the autopilot. In this case, the Mavlink component will be the one in charge of communicating with the controller. That component is able to:

- Send flight plans or navigation commands to the autopilot.
- Receive ACK, either from the flight plans or from the navigations commands sent to the autopilot.
- Inform about the position and attitude of the drone at every moment.
In addition, as the camera is situated on board, a component to control the device and to receive the Message Operation Setup is also implemented in the software. As can be seen in the Figure 4.4 (the component identified with the name, Camera Control Component). This service, will communicate with the Image Available component to inform in the user interface element the identification of the images that have been taken (see Section 4.2.3.4. to understand the visual component).

4.2.2. Communications

4.2.2.1. Ground - Air Link

As aforementioned in the previous part, the communication between ground and air will be established by Wi-Fi. The module of the standard 802.11 aboard the platform and the antenna in ground will allow the communication between both sections.

![Diagram showing the two types of communications protocols established in the platform.](image)

In case that the Wi-Fi link is lost, the platform will have available a 4G communication. As in the previous Wi-Fi link, the ground station and the drone will be equipped with a fourth generation module.

It is important to remark, the device which is going to receive the messages from ground is the processor, equipped with both modules, Wi-Fi and 4G module.

4.2.2.2. MAVLink Component - Autopilot Link

So as to establish the communication between the MAVLink component, which is the one in charge of linking the software with the autopilot. There are three different ways of communicating the messages to the autopilot depending the software configuration:

- User Datagram Protocol (UDP).
- Transmission Control Protocol (TCP).
- Serial Port
Then, these protocols will let the processor to communicate the navigation command lists demanded or the navigation commands. In addition, the autopilot will also send the acknowledges to the MAVLink component.

4.2.3. User Interface

In order to help the operator to control the system an user interface has been created. The initial main display will look as the Figure 4.6. As can be seen in the last mentioned image, there are two differentiated spaces in the principal screen. The big part (identified with a circled 1 in the Figure 4.6), contains the map, and the right section (identified with a 2 in the same previous Figure), is composed of by the different components designed for the mission.

![Figure 4.6: Initial main screen display, differentiated in two sections. The section 1, which contains the map, HUD and basic flight button menu, and the section 2, which contains the components for the intended mission.](image)

In the part identified with the number 1, as is shown in the Figure 4.6, is displayed the map. The map will also contain a drone icon and the designed mission model.

The drone icon, is presented to help the operator to check the position of the platform at every single moment. The software will also shown a yellow line (see black rectangle in Figure 4.7), which the unmanned aircraft intention, and a blue line for the flown track.

On the other hand, the mission model is also painted to aid the operator in the recognition of the points and for a better knowledge about the mission path. The contour of the coastline is presented in a red color. Unlike the shoreline, the rest of the defined model structure (take off sequence, departure sequence, route sequences and holding sequences) is displayed in green, except the points, which are highlighted in yellow with its identification written on the right side.
In the upper right part of the area 1, there is a head up display (HUD). This HUD (remarked with a blue rectangle in the Figure 4.6) shows basic information of the drone. At the right top of that box, is presented the "system id", to identify which drone is performing the intended mission.

Below that identification, the box informs to the operator about the platform altitude and speed (meters and meters per second respectively). Moreover, the attitude information of the unmanned aircraft will be also displayed in order to be checked by the operator.

In the left part of the HUD box, the gray triangle above the CMD letters (see Figure 4.11) is going to be updated while the aircraft is flying indicating the heading of the platform. Below that icon (see rectangle red in the Figure 4.11), two numbers will be showed too. The left number indicates the commands that have been flown, and the right number the ones that still have to be flown.

As the last thing that is going to be presented in the left part of the main screen, is found a button menu (see Figure 4.6, marked with a red rectangle or enlarged Figure 4.8). This mentioned menu, has been created to control the basic flights actions and paths of the platform.
As can see in the Figure 4.8, there are 6 buttons and their functions are going to be explained as follows:

- **Check up** button is going to be used by the operator in order to start the platform and perform the checks of the preliminary stage (see Section 5.3.1.).

- The **Take Off** button requests the generation of the take off sequence to the air segment. After that, the drone will arm, will lift off the ground and then will perform the take off path.

- **To Departure Hold** button (To Dept. Hold in the Figure 4.8), requests the generation of the departure sequence (see Section 2.3.3.3.), that means a new flight plan will be uploaded to the autopilot including the departure sequence to be flown by the unmanned aircraft.

- **Land** button. As aforementioned, the land sequence is reserved to use it in case that something goes wrong at the initial stage of take off. If the aircraft is holding at the take off hold and the operator notices that some system is failing, the supervisor may request a land (review Section 2.3.3.2. to remind the land sequence).

- **RTL**. If the Return to Launch button is pressed, the software will add to the current flight plan the back home path. This sequence is not going to be flown immediately after the click to the button, that path will be flown after completing all the previous requested operations.

- **Immediate RTL**. This request generates a direct path from the current position to the take off hold, moreover contains the path from the last mentioned hold to the land site. This action will be added to the current flight plan.

- **Emergency** button. After pressing the emergency button, a confirmation box appears in order to confirm the request (see Figure 4.10). Then, if this condition is accepted, a button menu as the Figure 4.9 will appear in the main display.

  - Direct Recover request, will generate an equal path as the Immediate RTL, but in this case the recover will be performed immediately after the request.

  - The Emergency Land generates from the current position a perpendicular path to the sand. Right after the click, the platform will avoid all the previous requests and will fly the aforementioned perpendicular path. That petition shall be used in a very much critical situation.

Following with the main display, the right part of the principal screen is going to be described. In that area will be contained the components to perform the planned mission properly. Currently, the software implements the following elements:

- Basic Information Control
- Meteorological Control
- Mission Trigger Control
- Image Available Control
- Ground Control
4.2.3.1. Basic Information Control

The Basic Information Control, contains the primary flight parameters, such as the attitude and the GPS position. This element is identified with the name Messages (see Figure 4.12).

Figure 4.12: Messages component, showing the receiving rate frequency of attitude and GPS data.

If the Messages accordion is unfolded (remarked with red rectangle in the last mentioned Figure) two different names will appear, Attitude and Gpsdata, with their corresponding update rate in Hertz (blue rectangle in the same Figure). Then, if either of them is clicked, in the lower right side of the area identified with a 2 in the Figure 4.6, appears the attitude and GPS information of the platform (see Figure 4.13).

4.2.3.2. Meteorological Control

The meteorological control has been designed in order to collect information about the weather conditions, to assure that the platform can fly securely and to estimate the victim movement in a determined period of time.

That meteorological station allows to download the information from a server, or set the weather conditions manually.

As in the Figure 4.14 can be seen, the meteorological control is divided in four different groups Currents, Winds, Update Info and Waves, a panel below the Update Info section (marked with a yellow rectangle), and an area with four radio-button (identified with a blue rectangle). Further down are found three buttons in order to download the weather forecast, set manually the weather data or upload the information the air segment.

If the user wants to get the information from the server, the button “Get From Server” should be clicked. Some seconds after the button has been pressed, the control will be updated with the corresponding values if they are available on the server.

After downloading the available meteorological data, the control will be full of numbers but is important to interpret them. Each of the four groups are going to be explained.

The Wind box, contains the following information:

- Heading, indicates the direction that the wind flows in degrees (0 deg - 360 deg). It is important to remark this number do not designate that the wind flows towards
the value representing the heading, this number indicates where the wind is coming from. Then, the number indicating 240.0 (see rectangle red in the Figure 4.14) degrees of heading, indicates that the wind is flowing in the 60.0 degrees direction.

- **Force**, represents the wind speed in meters per second.

- The **Type** will designate if the wind is gusty or continuous. This function is not implemented yet.

The **Current** box is going to show the information available of the total sea, not just the wind waves but the full sea propagation. As this information is complicated to obtain, depending if the meteorological spot has the necessary tools to measure it, the current information will be available or not. Every single characteristic is interpreted in the same manner as in the **Wind** box.

In the **Wave** box, the height value, indicates the altitude of the superficial wave in meters. The heading is understood in the same way than before and unlike in the other cases, force is not interpreted as has been in the previous cases. In that element, force is grasped as the horizontal velocity of a particle moving within the current waves. This value, will be helpful for the operator in order to discern if the victim will be moving quickly or not.

The **Update Info** case, displays three different information. Firstly, the source of the meteorological data. Secondly, The date of the last request to the server, including hour, minutes, day and month. And as the third element, the box informs if the information has been uploaded. The text “Not Uploaded” will appear if it has not been sent to air, otherwise “Uploaded” will be written.

Continuing analyzing that component, the information inside the red rectangle indicates if the data presented in the form is the one typed or the downloaded from the server, and the acquisition time of the information (hour, minute, day and month).

In addition, if the operator wants to maintain the current, wind and wave heading information, can maintain these values in the main screen just marking the three radio button (See winds, See waves, See current) inside the blue rectangle in the Figure 4.14 (see Figure 4.16 to see how these values are going to be displayed in the main screen).
48 Drone configuration for seaside rescue missions

The fourth radio button named "Use manual" will activate the button Set Manually. Then, by clicking one time to the button (button will look in dark blue, see Figure 4.18), the desired values can be inserted in the spaces. These numbers are going to be checked up by the software and if the format is wrong, the box will turn red (see Figure 4.15 the box in red). The text boxes just allow to write integers or decimals separated by a dot. Then, if the operator wants to confirm the typed values the Set Manually button should be pressed again (it will turn into light blue again, see Figure 4.17), then the values will be saved in the software.

Finally, once the values has been downloaded or completed manually, is time to decide which ones are going to be sent to air. If the typed values are the ones that are going to be sent, it is important to select the radio button Use Manual (located in the blue rectangle) and have the Set Manually button in light blue. If server forecast is the one that has to be sent, Use Manual shall be not selected. Hence, after deciding the last question, if the button Upload to System is pressed, the information will be sent to air.

4.2.3.3. Mission Trigger Control

The mission trigger control, has been designed to configure the trigger for the attached devices, in that case the camera and the drop devices to deploy the floats, using an operation template.

As can be in the Figure 4.19, the first thing that is found in the interface is a combo box (marked with a red rectangle in the last mentioned Figure) where the operations template generated can be found. In that case the template with the "id" equal to 3 has been selected.

Then, if the Send button is pressed the configurations are going to be set up. After that the Start button shall be pressed in order to start the service.

Just below the template selector, can be seen a blank area where appears "Castelldefels Mission". This message is a brief description to identify the utility of the selected operation.
CHAPTER 4. MISSION SOFTWARE

4.2.3.4. Image Available Control

The Image Available Control, is strongly connected with the mission manager control. In that component (showed in the Figure 4.21), is going to be displayed the identification of the images taken by the camera, according the configured operation template in the mission manager control (review Section 4.2.3.3.).

Once the first image has been taken, the component will generate an accordion with the “Operation Id” (rectangle yellow) stated in the Section 4.2.3.3. If this accordion is unfolded, will appear the identification distinguishing the camera device (in this case 47, blue rectangle), and the identification of the photo taken, in this case just one photo was taken at this point with its "id" equal to 0 (red rectangle).

4.2.3.5. Ground Control

The Ground Control component, has been designed specifically to request the path for the rescue operation. The user interface of that element looks as the Figure 4.22.
50 Drone configuration for seaside rescue missions

Figure 4.21: Component showing the operation id at the first level, the camera id in the second level and the id of the photo taken, in this case just one photo with an id: 0.

In the upper part of that component there are two clocks. The clock located at the left side indicates the hour, minutes and seconds of the current day. On the other hand, the clock placed on the left side will count the operation time, since the Take Off button is pressed.

Figure 4.22: Track component in which the operator will receive the notifications and commit the properly operation in each moment.

As can be seen in the Figure 4.22, there are two big different spaces, GROUND TRACKS and AIR TRACKS. Currently, the second one is not implemented but created for further implementations. The ground track space will receive the notifications of possible drownings explained in the Section 4.1., with an identification and the time that the warning has arrived to the database.

Inside the ground control, if the operator wants to review the images that have arrived, the supervisor should press the button remarked in the Figure 4.22 with a blue square. Right after the click, a window like the Figure 4.23 will appear. In the center of the image viewer, are going to be displayed the images. The two buttons below the picture viewer, allow to review the ten pictures that have been taken by the user. The forward button (arrow pointing to the right), advance the image, in the opposite way, the backward button (arrow pointing to the left), go back to the previous image.

Going back to the ground component interface, there are three more radio button (marked
with a red rectangle in the Figure 4.22).

- Last Ground Track
- Show All Tracks
- Track Estimation

Figure 4.24: Track component in which the operator will receive the notifications and commit the properly operation in each moment.

If the Last Ground Track radio button is selected, inside the main display of the ground station will be shown the information of the selected notification. That info, is conformed by the location where the photo has been taken and the possible location of the victim, connected by a line, as can see in the Figure 4.24 the blue line that connects the points identified with "Camera" and "Target" (indicated with a green rectangle in the same Figure).

In case that the check button (circled in yellow in the Figure 4.22) is selected, the radio button Show All Tracks will be enabled, then if the button is selected, all the ground track notifications will be displayed in the main screen, as the Figure 4.26 shows.

As the last radio button, the Estimated Track has been implemented. When this button is selected, the main screen is going to show two estimated points (see Figure 4.25). The first point, named "Estimation1" and marked with yellow color (see Figure 4.25), indicates the position of the victim, after the time elapsed by the drone until the initial mission hold. The second point, named "Estimation2" and marked with red color (see Figure 4.25), shows the estimated position in the elapsed time by the platform performing the scan.

Therefore, the mission user interface have three buttons to push the requests in order to perform the path of the mission hold, search target, confirm target, launch and confirm launch (which are the main operations involved in the rescue of the victim).

If the Commit Hold Location button is pressed, the mission hold path is being requested. This last point is calculated by the software, and showed in the main display as a black point over the hold point (see green ellipsis in the Figure 4.25). Then, this hold point will be the committed in the request (review Section 5.3.4. to see the mission hold path).
The Commit Search Location and the Commit Launch Location just will be enabled if the meteorological information has been uploaded to the air segment (review Section 4.2.3.2. about how to upload the weather data). Both buttons have two options.

The first time the Commit Search Location is pressed, a search target will be requested. If it is necessary, a second click in the same button will request the confirm target path (review the Section 5.3.5. for search and confirm target path). When requesting these both operations the point, where the search and confirm target will be performed is the "Estimation1".

A similar behavior happens with the Commit Launch Location. Firstly, the software will be requesting a launch path, and the second time the button is pressed, the request is going to be a confirm launch path (review Section 5.3.6. for understanding both paths). In these two cases, the waypoint around where both paths will be generated is the "Estimation2" point.
CHAPTER 5. MISSION PATHS PROCESSING

As soon as the operator decides to start the mission, every single route will be generated automatically by the software. In addition, to estimate the position of the victim, meteorological data will be needed for. Then, that section will be useful to understand all the paths and every single operation involved on it.

5.1. Route generation

After the mission kick off, operations demanded by the operator will be generated automatically by the software. These routes will be directly sent to the autopilot and consequently operated.

Firstly, the requests will be sent to air by radio (coded as a MAVLink message). Then, the air system is in charge of decoding the message and proceed with the necessary steps to serve the request.

Inside the system, specifically, in the software section, one service named Manager Flow service, which is the master on air, canalizes the requests to the Track Service inside the Trajectory Manager. The last one, demands (in the Figure 5.1, msg.r means request, and msg.a mens response or answer) to the DataModel Manager the route, depending if this is a route, scan or a launch path will be received in one of the three generators. After the route generation, the information returns to the Out Flow Service to be sent towards the Autopilot System as a MAVLink message (see Figure 5.1).

Between the autopilot system and the out flow service, the information is directed to the MAVLink component, which will establish the communication link with the autopilot system.

Figure 5.1: Summary of the information flows when requesting operations from ground.

The generation of these paths are not done in a random way, they must follow a proper sequence, because the Track Service will check the order of the requests to assure the perfect work of the system.

The following sections will explain the factors influencing the path generation, as the meteorological effects and the detailed path and operation for every mission step (review Section 2.2. to remind the basic idea).

5.2. Meteorologic effects

Inside a rescue and search mission, meteorological effects are an important limiting factor. Passing from not allowing the rescuer to quickly reach the victim, as to the point of the
appearance of a storm does not allow the lifeguard to access to the water, because of the danger involved in getting into the sea.

Hence, a tool to know the meteorological state of the coast where is happening the situation, will be very helpful. This kind of information is going to be used by the software to estimate the target position. Wind direction and speed, wave direction, speed, height, longitude and period, or even current speed and direction are very helpful for the analysis.

The wind state is going to be profitable for the operator in order to decide if the conditions are good enough to fly the platform. On the other hand, wave state will definitively help to estimate where the victim is moving due to the water movement in a certain period of time, to be more accurately, the time that the platform will take to reach the mission hold and the time the unmanned aircraft will take to perform the scan, as can be seen in Section 4.2.3.5.

5.2.1. Meteorologic data request

To obtain this kind of data, a request to a web page is going to be performed. That solicitation to the page, will let the software to connect with it and download all the information of this web site. After that process, a parser has been created to collect from that source the meteorological data aforementioned.

Right now, there is one meteorological source implemented, named Maritima but before that, State Weather Agency was the first source of meteorological data.

The State Weather Agency lets to obtain a limited number of data and a barely liability. That fact is caused by the low rate of update and the precision of the values obtained, because the agency just provides three different factors named as soft, moderate and strong, without any information of direction.

In the case of the wind intel, only a brief description is provided to identify the breeze (see Table 5.1). Then, to obtain an average value of that conditions the Beaufort scale should be used. That scale, is table that collects empirical values which allows to relate the wind speed to observed conditions at land or sea.

<table>
<thead>
<tr>
<th>AEMET Tag</th>
<th>AEMET Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soft</td>
<td>Clear or light breeze</td>
</tr>
<tr>
<td>Moderate</td>
<td>Normal breeze at the beach</td>
</tr>
<tr>
<td>Strong</td>
<td>Winds of any kind, which raise the sand, and strongly shake the trees</td>
</tr>
</tbody>
</table>

For the wave information, AEMET [14] provides a range for each of the three commented categories, collected in the Table 5.2. Similarly as in the wind data, a conversion is needed to obtain some relevant values to be analyzed. These values were obtained by mean of a table which related the wind speed with the water velocity.

Because of that difficulties to obtain reliable values the source of meteorological information was changed to Maritima Meteo Consult [15].

That last source, solve all the previous problems, inclusive giving a high reliability and updating the meteo every two hours. The Figure 5.2, shows the data that is going to be
### Table 5.2: Information about wave state given by the AEMET service.

<table>
<thead>
<tr>
<th>AEMET Tag</th>
<th>AEMET Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soft</td>
<td>Up to 0.5 meters tall</td>
</tr>
<tr>
<td>Moderate</td>
<td>Between 0.5 to 1.5 meters tall</td>
</tr>
<tr>
<td>Strong</td>
<td>Waves overpassing 1.5 meters tall</td>
</tr>
</tbody>
</table>

obtained from the web. In addition, as can see in the Figure below, the information is divided in 24 hours per day, that fact lets the software to chose the data depending the current time, which increases the precision of the meteorological forecast.

[Figure 5.2: Extract of the Maritima Meteo Consult web use to get the needed meteorological information.]

The wind state in that case is made up by the wind direction and its speed, letting the operator to figure out in which conditions the platform is going to be operating, and if it is necessary abort the mission to avoid greater risks.

On the other hand, wave information is formed by current height, wave height, wave direction, period and longitude. As the wave movement can be similarly understood as sinusoidal waves, waves can be treated like these ones.

- **Amplitude**: difference between the zero line and crest height (m).
- **Wavelength**: Longitude difference between two following crest (m).
- **Period**: Time to perform one cycle, specifically the time the wave takes to reach from a crest to the next crest (s).
• Frequency: Inverse of the period (Hz).

With these characteristics the propagation velocity can be calculated easily, but that speed is not the velocity the people move within the water. To reach an average value for the horizontal velocity of a particle moving inside a wave, the linear wave theory has been used.

5.2.1.1. Target’s movement prediction

The propagation velocity is computed as the wavelength times the frequency or divided by the period (see equation 5.1).

\[ c \ [m/s] = \lambda f = \lambda / T \]  \hspace{1cm} (5.1)

That velocity is the wave speed but no the one that a person inside the water moves, as commented previously. Then, it is necessary to find an expression capable of determining an approximated speed of a particle moving inside a sinusoidal wave. Once arrived at this point, a particle will be extrapolated or treated as a group of infinites particles moving together, assuming that this group is the human body, and that the whole is a big particle moving at the same velocity that all of its formers move.

According to the linear wave theory, its basic concept states that particles moves in elliptical orbits. Due to the \( u \) velocity on top is slightly larger than in the bottom of the ellipsis, the result is demonstrated as a little displacement along the wave direction (see the Figure attached below 5.3). This net motion is called Stokes drift.

![Figure 5.3: Figure showing the Stokes drift behavior inside a wave.](image)

It is important to remark, that the mathematical demonstration is not going to be explained. This is because the project is not focused in understanding how the equations are obtained, but how the equations are applied for the purpose [2].

As a result of the mathematical development of the equations that set up the linear wave theory, different expressions are obtained depending on the sea depth, specifically in three differentiated groups:

• Shallow
• Intermediate
• Deep
This division has been established by giving range to the relation in Equation 5.2. This factor is just a ratio between the depth of the water and the wave length of the studied wave.

\[
\text{ratio} = \frac{d}{\lambda} \quad (5.2)
\]

According to the mentioned theory these three groups are limited by the following values contained in the Table 5.3:

<table>
<thead>
<tr>
<th>Shallow Water</th>
<th>Intermediate Water</th>
<th>Deep Water</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d/\lambda &lt; 1/20)</td>
<td>(1/20 &lt; d/\lambda &lt; 1/2)</td>
<td>(d/\lambda &gt; 1/2)</td>
</tr>
</tbody>
</table>

Different hypothesis have been decided to apply in order to simplify the approximation of the horizontal velocity of the victim moving within the sea.

1. Firstly, the vertical velocity has been neglected, taking the supposition that this is not going to give any information of the horizontal movement, but maintaining in mind that, erasing this velocity component, will focus all the module in the horizontal one, oversizing its real value.

2. Secondly, in order to simplify the election among the three possible cases, it has been assumed that this kind of operation will be performed normally in intermediate waters.

Then, assuming this two hypothesis the number of equations has been reduced to a simply one. Firstly, due to the initial hypothesis of neglect the vertical velocity the three equations (one for each depth division) has been erased. Then, with the medium sea depth assumption, the equations resolving the horizontal velocity for shallow and deep waters have been deleted too. This way, just one equation (see Equation 5.4) has to be taken into account to approximate the horizontal velocity with the previous hypothesis.

\[
z [m] = A \cos(\omega t) \quad (5.3)
\]

\[
u [m/s] = w a \frac{\cosh k(z + d)}{\sinh kd} \sin(\omega t) \quad (5.4)
\]

At this point, that equation, with the mentioned assumption, allows to compute an estimated horizontal velocity at a certain point of the wave. Hence, to compute an average value of the horizontal velocity, it is calculated in steps of meters and after that, the mean value is computed (see Equation 5.5). That mean value, will be the one used to software to estimate the victim movement within the wave, just using the basic showed in the Equation 5.6.

\[
u_m [m/s] = \frac{\sum_{i=0}^{L} u}{N.\ steps} \quad (5.5)
\]
5.3. Operation Flow

After taking a look to the meteorological information, the mission steps are going to be reviewed in a detailed way. Detailing the operations performed inside of each one and being so specific with the flight plans of each stage. Moreover, a diagram of the mission flow is showed in the Figure 5.4.

5.3.1. Preliminary checks

Once, the operator has decided to start up the platform, several checks will be performed in order to assure the properly work of the essential systems.

At this point, it is known that the system will be flying according to a model defined by the user and it is playing a big role in the whole operation. Then, first of all the software will check if ground and air are having the same mission model, to assure a concordance. To be sure about that, the fields specified in the file called, missionId and checksum are going to be sent to air and checked, if they are equal mission will proceed as intended, if not, the operator will receive an advice notifying the situation.

On the other hand, telemetry link will be revised too, being this communication link the most important one in the platform. Finally, after receiving telemetry from air, the position of the vehicle will be checked up to assure it is in the land-site position. This telemetry control will be performed by the software but the supervisor has to check it too.

5.3.2. Take-Off phase

The take off phase is the one used to lift off the ground and do again some security checks before moving away the coast.
By this time, preliminary revisions are over. If the initial checks have gone successfully, the operator will have green flag to take off and de-attaching the unmanned vehicle from the ground.

Hence, the pilot on-command will request a take off operation, if there are no obstacle, the generator service will accept the request, and automatically the unmanned aircraft will be cleared to take off. This clearance is automatic, and after checking that the message requested is equal to the received by the acknowledge (as known as **ACK**) of the autopilot system, the aircraft will lift off and perform the take off phase, ending at the take off hold.

Either before or once the platform has arrived to the take off hold, the operator should request the departure phase if all the services are working properly. After that request, the software will be in charge of canalizing all the data flows to execute petition.

That phase will depend on the designed model created by the user, but in the Figure 5.5, there is a realistic example, on Castelldefels coast.
5.3.3. Departure phase

The departure phase follows the take off sequence, this path will serve to reach the departure hold (see Figure 5.6), which is connected with all the route points and consequently with the hold waypoints, but also let the vehicle to gain altitude until arriving the route height.

As in the previous phase, the path depends on the user creation, but to have an example, the Section 2.3.3.3. contains one sketch on a realistic case in the Castelldefels coast.

5.3.4. Access to Mission Hold

At this time, the unmanned aircraft will be at the departure hold, ready to make the move towards the initial mission hold. It is intended, that after the departure phase, an order will be committed with the information of the desired holding waypoint (among of all the ones defined previously). In that phase, the speed is increased to the cruise speed but the altitude is maintained. Current stage is the one that takes the shorter time due to the high speed.

As can see in the Figure 5.7, the software will generate a route towards the mission hold.
Figure 5.7: Figure showing the path generated when a request of **ToMissionHold** to the selected hold.

point using the defined route points. In that specific example, there are two connected segments. The first, links the departure hold with the parallel route point. The second joins, the route point with the hold selected. In case the desired hold is not the first, the platform will follow the pertinent route points until arriving to the one, which is parallel to the planned hold point.

Meanwhile, the operator should be more concerned about the possible new notifications of the victim and the operation time, directly related with the battery.

In case that the operator has made a mistake in the selection of the initial mission hold, another hold can be selected and committed to air, in order to correct the first decision.

### 5.3.5. Scan phase

Following the generation to the mission hold, the scan phase is the next on line. The scan phase can be made up of two different stages:

- **Search Target**: in this stage, the micro-vehicle will perform an squared scan. This pattern has been parametrized with three characteristics parameters; width, height and the number of scan passes needed (see Figure 5.9). Then while this step is active, the unmanned aircraft will start the camera trigger to capture images. In this stage the aboard processor will process the images to detect the victim.

Initially, the drone will overfly the target (see Figure 5.8 the tag with the name **Target**) and will start to perform the scan. In this case the scan selected is from in to out, and the initial turn is to the left. After performing all the scan, the platform will pass over the **Return Point** (see Figure 5.8), to turn directly to the target location. Once again, the unmanned aircraft will overfly the target point and will come back, perpendicularly to the last past to the hold point (see Figure 5.9 to clarify the explanation). At the hold point, the unmanned aircraft will wait, performing a hold, new requests from the ground station.

Currently, the scan phase will be performed around the first estimated point (review the Section 4.2.3.2. to clarify the meaning of the first estimated point) calculated by the software implemented in the ground control station, and committed by the supervisor.
62 Drone configuration for seaside rescue missions

Figure 5.8: Figure showing two paths. The initial route from the Initial Mission Hold to the Target To Scan, and then, the search target stage.

- **Confirm Target:** is an optional stage designed to re-confirm where the target has been detected by means of the image processing. In this step a little scan is performed too, this scan is parametrized in the same way as the mentioned in the search target stage.

  As the victim detection by image processing is not implemented yet, in order to simulate a real case the confirm target stage is committed with the same point use in the search target stage.

5.3.6. **Launch phase**

Finally, the operation that gives sense to the mission is this one, the launch phase. As the previous phase, this one has also two different stages:

- **Launch:** this step is essentially designed for launching the float or floats. This path has been created to launch the float devices with a minimum security distance of two meters near the victim in the opposite direction of the wave heading. That strategy will use the water movement to get the float closer to the victim.

  The launch path consists in going directly to the committed location of the possible victim. Then, the software, with the meteorological information will generate another point, with the aforementioned security distance in order to avoid the direct impact of the float to the victim.
From the route point the drone will go directly to the target location (Target Location in Figure 5.10), descending its altitude to the configured value in the model. Then, the route generated will guide the platform two meters away the victim location, in the opposite direction of the wave heading (as can be seen in the Figure 5.10 with the name Launch point). In the **Launch** point, a hold of 10 seconds will be performed in order to deploy the float. After deploying the floating device, the drone will go away 13 meters (marked with the name, Away Point in the last mentioned Figure) to come back to the victim location, overflying again the launch point. In that point, the platform will hold awaiting new requests.

Presently, due to the lack of processing image aboard, the launch point is the second estimated point computed by the software included in the ground control station (review the Section 4.2.3.5. to identify the second estimated point). Moreover, a trigger to launch the floating device has been implemented when overflying the mentioned calculated point.

**Confirm Launch:** is an optional step as the confirm target too. During this stage, an equal path as in the launch level will be performed, but in addition some photos
will be taken to confirm that the float has been deployed correctly. These photos will be taken in the same point where the float has been launched.

5.3.7. Platform recovering

Platform recovering has been designed to be performed in three different ways depending on the situation.

- If there is a bad working at take off hold, a **Land** commit should be requested. If the operator recognize a malfunctioning in the unmanned aircraft or with the systems involved in the operation, while the unmanned vehicle is holding at take off hold, the supervisor can request a land. This path will guide the vehicle back to the land site (see Figure 5.5).

![](image)

**Figure 5.11:** Figure representing the immediate RTL.

- If everything is in normal conditions after executing the whole mission, the **RTL** commit shall be requested. This type of recover will re-play all the points, but in the inverse order after completing the requested operations.

- Direct recover after commit **Immediate RTL**. This type of return is created for recover the vehicle in case that the battery is limited to perform a normal back, but still enough to go back in a direct way, then the path generated connect the last mission requested point with the take off hold, and consequently with the land site.

  The Figure 5.11 shows an example of the immediate RTL from the last point of the launch sequence.

- In an emergency situation, two possible decisions can be selected by the operator:

  **Direct recover** request. The platform interpret this request as an emergency situation, this way, the direct recover will directly head the vehicle to the take off hold. Then, will perform the path towards the land site. This action will be executed right after the request, not as the previous ones.
Figure 5.12: Figure representing an example of an emergency land request.

Emergency land request. After this request, the software implemented in the system will calculate the coast projection of the aircraft current position to perform a perpendicular path to the strand. This case of emergency, should be activated in critical situations, due to the proximity with the swimmers when performing the land (see Figure 5.12).
CHAPTER 6. INTEGRATION AND TEST

6.1. Platform integration

As aforementioned in the Section 3, the flight platform will carry aboard various systems and devices, but one important thing is how these ones are going to be integrated on the frame.

Figure 6.1: Top view of the 3D solid to show the integration of all the systems aboard.

In order to better understand the distribution of these instruments, a 3D model has been created with the tool named SolidWorks.

There are two differentiated parts as in the Figure 6.1 can be seen, upper and lower part. In the upper part will be placed the autopilot, the GPS and the telemetry system.

As can be seen in the image, the autopilot is located in the air-frame center of mass and pointing to the front face. The GPS will be also situated in the upper part and slightly lifted from the upper plate in order to avoid or reduce interferences with the motors. On the other hand, telemetry antennas will be placed to in the topmost part, that as previously mentioned will let the knowledge of the drone position on the ground control station.

On the opposite position, the lower part, as can be seen in the Figure 6.2 it is made up by three plates, two big ones and a little tiny plate.

The two big slices are going to be used to locate the battery (the big rectangular above the big plates) and move it to center the mass. In addition, in the inferior part of that two big planes there are attached two devices for ejecting the floating tools.

The devices used to deploy the float (see Figure 6.2) contain a servo and a bar to drop out the wire that will be holding the floating instrument.
68 Drone configuration for seaside rescue missions

Figure 6.2: Figure showing an example of the allocation of all the elements in the lower part.

In the top of the little tiny plate, it is placed the Raspberry Pi3 with the 4G and Wi-Fi modules (not presented in the Figure 6.3) to communicate with the ground control station. Hence, the lower part will let to locate the camera inside the gimbal (see Figure 6.3).

Figure 6.3: Showing an example of the location of the raspberry and the camera gimbal.

6.2. Test

Commonly when developing project, before releasing it to a commercial version, it is needed to do an infinite number of test of every single part and system to assure the correct design and implementation.

6.2.1. Camera test for detecting victims in "Roda de Berà"

During the 2016 Summer five tests were performed within the coast of Roda de Berà. The test consisted on performing the stage of the search target, while the camera was activated
and taking photos.

Photos were post-analyzed in order to see how a victim is visualized from the camera aboard the unmanned vehicle (some attached Figures 6.4, 6.5). For further innovations, these photos could be really helpful in order to try different filters or image processors for the detection of the victims.

![Figure 6.4](image1.png)  ![Figure 6.5](image2.png)

**Figure 6.4:** Figure exemplifying how the sea is visualized from the camera on board.  
**Figure 6.5:** Figure representing the water from the camera aboard the platform.

It is important to remark that the micro-aerial vehicle used in these tests was different to the planned for the LifeLine. In this case was a quadcopter, specifically the **F550** (see Figure 6.6), manufactured by DJI, this one is smaller than the Tarot X6 and consequently is more easy to transport.

![Figure 6.6](image3.png)

**Figure 6.6:** DJI 550 used for the tests in Roda.

Moreover, there were two persons inside the water. They were located at more or less 200 meters from the breakwater (as can see in the Figure 6.7, the two persons squared in green).

![Figure 6.7](image4.png)
6.2.1.1. Test Concept of Operation

The base was placed over the breakwater in the port of Roda. The ground station was made up by a computer, in order to control the position and the intentions of the vehicle. In addition, three supervisors were around the mission area.

On the other hand, a boat carried three more persons, two possible victims and one needed to communicate with the ground control station personal. That communication was basically to notify when the platform was deployed, so as to throw the swimmers into the water to be caught up by the camera aboard the flying vehicle.

While the possible victims were being thrown to the water, the unmanned aircraft was flying to the planned point, proceeding directly to perform the scan (see Figure 6.8). Right after the vehicle reached the intended point, it began to scan whilst the camera was taking photos.

After completing the scan, the vehicle returned to the base and the swimmers were rescued by the boat.
6.2.2. Simulation Test

To assure the proper work of the flight plans generated in each stage of the intended mission, some simulations have been performed inside a computer using "Software in the Loop" and checking the navigations commands with the QGroundControl.

The first request is the take off path. In the Figure 6.9, can be seen how the ground control station shows the drone location.

![Figure 6.9: Take Off sequence showed in the QgroundControl station.](image1)

Following with the missions steps previously mentioned, the departure sequence is activated. As has been said in several times, the departure path will guide the LifeLine vehicle to the connection point attached to the branches containing all the route points. As can be seen in the Figure 6.10, the unmanned vehicle is performing the path, being the departure hold its last point of the sequence.

![Figure 6.10: QGroundControl representing the drone flying after the take off hold.](image2)

![Figure 6.11: Vehicle being displayed in the user interface station. The aircraft is flying towards the black point.](image3)

Committing a mission hold request, will direct the unmanned aircraft to the closest point to the holding point selected (see Figure 6.11).
After that stage, an scan has been requested. In the Figure 6.12, can be seen the flight plan guiding the vehicle to the target point, then the scan starts. In this case, the scan has a measures of 50 meters tall and height, and 5 turns.

![Figure 6.12: Figure showing the scan path in the flight plan.](image)

As the confirm target just will show again a little scan sequence, the launch phase is started (review Figure 6.13). In that case, the launch path is uploaded. As the QGroundControl stores the whole flight plan, the launch plan can not be clear seen.

![Figure 6.13: Scan patter with the launch path.](image)

Finally, a normal recover is requested. This way, the drone will re-flight all the flown paths, following the points state in the go section.
CONCLUSIONS

The development of this project, has allowed to see that the implementation of this kind of rescue platforms would be feasible. A ground station, has been created to control the unmanned aircraft by an operator. Moreover, the platform has been equipped with one floater to be deployed. In addition, the initial objectives planned, when the foundations were stated, have been accomplished.

On one hand, the software designed for the purpose allows to perform a whole rescue mission in the fixed envelope explained in the dissertation. It is clear, that a lot of improvements should be inserted for a final version.

Firstly, the designed and implemented ground control station allows the operator to review what is happening in a log file, not in real time. This should be improved in further versions to be accessible in real time.

As a second fact, the possibility of incorporating a cancel button would be a good option. Currently, just one request of an operation can be requested. It was defined this way, because buttons are activated and deactivated depending the current state, but return and emergency requests are always available. Then, if the operator by mistake presses the return buttons (not the emergency because it has its own confirmation button), the cancel button will allow to return to the back requested state.

Finally, as has been commented in the previous chapters some characteristics have to be implemented.

- The computation of the estimated position of the victim from the photos taken by the users.

- Consequently and also related with the image processing, would have to be implemented the live stream processing of the images taken by the cameras aboard in order to detect the possible victim.

On the other hand, related with the tools integration aboard the vehicle, all the crucial devices for the flight have been mounted and tested.

However, a lot of test have to be performed with the drops and the floater, in order to assure the correct work. Testing this situation, if possible, in different weather conditions. These tests, will serve to learn how the floater behaves depending on the test conditions, so as to select the altitude and security distance of the launch according to the weather state.

Related also with the meteorological conditions, on ground is situated the meteorological station. It is a meteorological station, capable of getting a reliable forecast about winds and superficial waves. Otherwise, the currents are not that reliable and in some cases, there are no information available. Then, the meteorological server, should be improved in further versions to obtain all the possible information with the best reliability in order to have a better estimation of victim position.

As an important part for certifying the proper function are found the tests. Currently, all the ground and air components were tested in a computer. The network communication, has been performed as it will do in a real test, otherwise, the radio communication has been simulated by a TCP port inside the computer. In addition, all the possible flight paths have been simulated with the software in the loop, checking all the information with a ground
control station named, QGroundControl, to assure the proper packets containing all the generated commands had been correctly sent.

Besides that, a lot of more simulations will be needed in real situations. Real simulations, either in the mission situation or in areas to perform drone flights. These flights will serve to know how the vehicle behaves in real conditions.
BIBLIOGRAPHY


APPENDICES
APPENDIX A. APPENDIX

A.0.1. Castelldefels example

The previous sections have explained how to define all the structures which are going to be placed within the extended markup file, but this little subsection will help with a brief real example in Castelldefels coast.

Listing A.1: Example for defining the operation model in Castelldefels coast.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<LifeLine name="Castelldefels-CoastLine" id="04" checksum="1">

<!-- Main heading of the coastline looking from the sea -->
<coastLineHeading value="5"/>

<!-- ToDeptHold - RTL Landing point-->
<land id="land" latitude="41.26508713" longitude="1.96388507" altitude="10"/>

<!-- TakeOff point - Mandatory -->
<takeoffhold id="take-off" latitude="41.26358647" longitude="1.9638879" altitude="20"/>

<!-- At ToDeptHold Hold point-->
<initialHold id="ihold" latitude="41.26060884" longitude="1.96391297" altitude="40"/>

<coastLine>
  <position latitude="41.25790557" longitude="1.91781111" altitude="0"/>
  <position latitude="41.25774628" longitude="1.91889204" altitude="0"/>
  <position latitude="41.25755674" longitude="1.91982276" altitude="0"/>
  <position latitude="41.25712524" longitude="1.9200373" altitude="0"/>
  ...
  Rest of the points
</coastLine>

<takeoffSeq>
  <waypoint latitude="41.264253" longitude="1.9638870" altitude="10"/>
</takeoffSeq>

<landSeq>
  <waypoint latitude="41.26425" longitude="1.96388701" altitude="10"/>
</landSeq>

<departureSeq>
  <waypoint latitude="41.261445" longitude="1.96390652" altitude="40"/>
</departureSeq>

<arrivalSeq>
  <waypoint latitude="41.2614455" longitude="1.9639065" altitude="40"/>
  <waypoint latitude="41.2635864" longitude="1.9638879" altitude="20"/>
  <waypoint latitude="41.264253" longitude="1.96388701" altitude="10"/>
</arrivalSeq>

<coastLineRouteModel>
  <!-- One branch definition -->
</coastLineRoute>
```
<waypoint id="ihold" latitude="41.2606088" longitude="1.9639129"
    altitude="40"/>
<waypoint id="NE1" latitude="41.2606101" longitude="1.96833292"
    altitude="40"/>
<waypoint id="NE2" latitude="41.2606101" longitude="1.97275282"
    altitude="40"/>
... Rest of the branch points
</coastLineRoute>

<!-- Route to holding points definition -->
<coastLineRoute>
    <waypoint id="NEH1" latitude="41.26106199" longitude="1.96832944"
        altitude="40"/>
</coastLineRoute>
<coastLineRoute>
    <waypoint id="NEH2" latitude="41.26106215" longitude="1.97274939"
        altitude="40"/>
</coastLineRoute>
... Complete the file with the rest of the holding points of the first branch
<coastLineRoute>
    <waypoint id="ihold" latitude="41.2606084" longitude="1.96391297"
        altitude="40"/>
    <waypoint id="SW1" latitude="41.26061177" longitude="1.9594652"
        altitude="40"/>
    <waypoint id="SW2" latitude="41.2604373" longitude="1.95505139"
        altitude="40"/>
... Rest of the points of the second branch
</coastLineRoute>

<!-- Route to holding points definition for the second branch -->
<coastLineRoute>
    <waypoint id="SW1"/>
    <waypoint id="SWH1" latitude="41.26106194" longitude="1.95946513"
        altitude="40"/>
</coastLineRoute>
<coastLineRoute>
    <waypoint id="SW2"/>
    <waypoint id="SWH2" latitude="41.26088663" longitude="1.95505139"
        altitude="40"/>
</coastLineRoute>
... Complete the file with the rest of the holding points of the second branch
<securityPol>
    <position latitude="41.271359" longitude="2.061754" altitude="0"/>
    <position latitude="41.267823" longitude="2.063343" altitude="0"/>
    <position latitude="41.261301" longitude="2.030761" altitude="0"/>
    <position latitude="41.258695" longitude="1.999809" altitude="0"/>
    <position latitude="41.258210" longitude="1.946621" altitude="0"/>
... Rest of the security polygon
</securityPol>