

# TFG-Teleco

pedro.arias

December 2019

## Resumen

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.



# Capítulo 1

## Introducción

Este primer capítulo recoge lo esencial del trabajo. En él, se explica el contexto en el cual se enmarca el estudio, el problema a solucionar, la motivación del mismo y los objetivos extraídos del problema. Además, se presenta también cual será la estructura seguida en la memoria.

### 1.1. Contexto

El término *robot* aparece por primera vez en 1920, en la obra teatral *Rossum's Universal Robots* del escritor checo Karel Capek en cuyo idioma la palabra “robota” significa fuerza o servidumbre [29]. El nacimiento de la robótica y los robots surge asociado a la idea de trabajo y producción tras la Segunda Revolución Industrial y a lo largo del siglo XX. La automatización industrial de aquella época da lugar a los primeros sistemas de control automático que se extienden rápidamente a todos los sectores industriales, y que dan lugar a la robótica industrial tal como la conocemos hoy en día [3].

El término *robótica* es acuñado por Isaac Asimov, definiendo a la ciencia que estudia a los robots. El propio Asimov postuló también las Tres Leyes de la Robótica en su libro *Yo Robot* publicado en 1950 [29]. El término ha evolucionado mucho desde sus inicios, hoy entendemos la robótica como la ciencia o rama de la tecnología que estudia el diseño y construcción de máquinas capaces de desempeñar tareas realizadas por el ser humano o que requieren el uso de inteligencia [14]. Como se puede entender, la visión actual es mucho más amplia que en sus inicios y abarca muchas áreas de la ingeniería.

Existen diversas clasificaciones en función de su arquitectura, de su aplicación, de su cronología, etc. Una de estas clasificaciones distingue robots en función de su morfología [9], que suele distinguir los siguientes tipos:

- **Poliarticulados:** Son artillugios mecánicos y electrónicos destinados a realizar de forma automática determinados procesos de fabricación o manipulación. Suelen ser fijos, aunque también pueden realizar desplazamientos limitados y poseen un espacio de trabajo concreto y limitado. Los mejores ejemplos son los robots industriales, manipuladores o cartesianos.
- **Móviles:** Están provistos de algún tipo de mecanismo que les permite desplazarse autónomamente, como patas o ruedas, y reciben información de su entorno a través de sus propios sensores. Son ampliamente utilizados en el transporte de mercancías o en la exploración de lugares de difícil acceso. Pueden ser terrestres, acuáticos, aéreos o espaciales.

- **Androides:** Intentan reproducir total o parcialmente la forma y el comportamiento del ser humano.
- **Zoomórficos:** Son aquellos que trata de reproducir en mayor o menor grado de realismo los sistemas de locomoción de diversos seres vivos.

Este trabajo se enmarca dentro de la robótica móvil, y más en concreto, dentro de la robótica aérea. La robótica aérea es la rama de la robótica que se encarga del estudio del comportamiento autónomo de aeronaves no tripuladas. Se entiende como una aeronave no tripulada (UAV, *Unmanned Aerial Vehicle*, o más recientemente UAVS, *Unmanned Aircraft Vehicle System*) a aquella que es capaz de realizar una misión sin necesidad de tener una tripulación embarcada [2].

A la hora de establecer una clasificación de los UAV es posible atender a diferentes criterios. Siguiendo la clasificación propuesta por *Barrientos et al.* [2] se distinguen aeronaves en función del tipo de despegue, que puede ser vertical o no. A su vez, podemos subdividir las aeronaves en función del origen de su sustentación o del tipo de ala que poseen. Esta clasificación se representa en la Figura 1.1, mientras que en la Figura ?? se muestran los diferentes tipos de UAV.



Figura 1.1: Clasificación de los UAVs [2].

Otros criterios de clasificación pueden responder a las capacidades de vuelo como el alcance, la altitud, la autonomía o la carga máxima. A su vez, también se clasifican las

TEMP TEMP

(a) Prueba 1.

(b) Prueba 2.

Figura 1.2: Tipos de UAV.

aeronaves en función de la actividad que realizan. Las principales aplicaciones se recogen en la siguiente lista:

- Militar: de apoyo, de combate, de reconocimiento, etc.
- Transporte, tanto de mercancías o de personas.
- Seguridad, vigilancia y salvamento.
- Ocio y entretenimiento: en cine, en deporte, etc.
- Educación e investigación.
- Agropecuario: fumigación, control de recursos, etc.
- Cartografía, topografía y fotografía aérea.

La siguiente figura (Fig. 1.3) recoge alguna de las principales aplicaciones mencionadas.

TEMP TEMP

(a) Prueba 1.

(b) Prueba 2.

Figura 1.3: Principales aplicaciones en robótica aérea.

En la actualidad tiende a utilizarse el concepto de UAVS frente al de UAV. La extensión del concepto de vehículo a sistema refleja que el vehículo aéreo autónomo precisa para su funcionamiento de todo un sistema y no solo de la aeronave instrumentada. En concreto la instrumentación embarcada o segmento aire, debe verse complementada con la estación base o segmento tierra, debiéndose considerar las funcionalidades y características de ambos segmentos.

## 1.2. Problema y motivación

Este trabajo se centra en el diseño e implementación del primer prototipo de un software tipo *estación de tierra* que permita la operación automática remota de un Vehículo Aéreo no Tripulado (UAV, *Unmanned Aerial Vehicles*), a los que me dirigiré comúnmente como *drones* a lo largo de esta memoria.

Esta oportunidad surge de la colaboración entre la empresa madrileña CONYCA S.L. [33] y el grupo de investigación de robótica (@RoboticsLabURJC [10]) de la Universidad Rey Juan Carlos (URJC), con el proyecto *ROSpilot, software para operación y navegación de un UAV*, en la cual he tenido la suerte de ser invitado a colaborar.

Profundizando en la estación terrestre requerida, se busca automatizar procesos tediosos relacionados con labores de cartografía y toma de fotografía aérea. En este tipo de misiones es común utilizar drones de ala fija (tipo *Geodrone Mapper* [24], ver figura 1.4), lo cual agranda aún más el desafío propuesto debido a las limitaciones existentes en las operaciones de una aeronave de ala fija.



Figura 1.4: Geodrone Mapper, propiedad de CONYCA S.L., utilizado en misiones de cartografía y fotografía aérea.

La inmensa mayoría de estaciones de tierra propone soluciones generales. Desde el inicio del proyecto se concibe un software íntimamente ligado a resolver y mitigar los problemas principales asociados a la cartografía y a la toma de fotografía aérea.

En las siguientes secciones se describirá más en detalle el problema y como se afronta la solución del mismo.

## 1.3. Objetivos

Tras describir el problema a resolver en la sección anterior (Sección 1.2), en esta sección se quiere aclarar cuales son los objetivos perseguidos desde el inicio del proyecto. El objetivo principal es proporcionar una solución software que permita la operación automática remota a drones de ala fija. En concreto, la aplicación desarrollada será denominada como *UAVCommander* y parte del trabajo previo desarrollado por José Antonio Fernández Casillas en su proyecto fin de carrera *Navegación por posición para un avión autónomo con JdeRobot* [4].

En el proyecto acordado con la empresa se describen una serie de funcionalidades o requisitos que debe incluir la aplicación, los cuales se consideran objetivos secundarios:

- **Caracterización de la aeronave y carga de pago.** El usuario podrá configurar datos operativos de la aeronave y datos de la carga de pago, que consiste en una cámara con la que se tomarán las fotografías para realizar la cartografía.
- **Creación de misiones de vuelo automático.** El usuario podrá crear misiones de tipo multilínea como sucesión de puntos de paso (*waypoints*) o misiones de tipo superficie en las que la herramienta ha de planificar la trayectoria de barrido para cubrir toda la superficie. Para ambas misiones la referencia en la altitud podrá ser fija respecto al punto de despegue o variable según los diferentes puntos de paso.
- **Comprobación de pre-vuelo.** El usuario dispondrá de una lista de comprobaciones de seguridad previas al vuelo, entre las que se encuentran la calibración del tubo de Pitot, el estado de las baterías, etc.
- **Seguimiento del vuelo.** El usuario podrá visualizar la posición de la aeronave en todo momento acompañado de datos de telemetría y/o batería entre otros.
- **Análisis de datos post-vuelo.** El usuario podrá observar, descargar o eliminar los datos del log generado por el autopiloto de la aeronave.

Por último, se consideran otros objetivos ligados al enfoque cartográfico de la aplicación. Es lógico pensar que uno de los motivos por los que surge este proyecto es reducir y facilitar las tareas desempeñadas por el operario con el objetivo de una reducción sustancial en el tiempo empleado en este tipo de tareas. Además, otro objetivo que se persigue es la reducción o eliminación de errores que puedan surgir de una equivocación humana y que produzcan una repetición parcial o total de la misión.

## 1.4. Estructura de la memoria

En esta sección se describe la estructura de la memoria, se introducen los diferentes capítulos y los temas que se tratarán en cada uno de los mismos.

En primer lugar se encuentra un capítulo introductorio (Capítulo 1), en el cual se halla el lector al leer estas líneas, y que se concibe como una serie de aclaraciones previas y necesarias para el correcto entendimiento del trabajo en su conjunto. Incluye un contexto histórico de la robótica, la robótica aérea y las estaciones de tierra y sus aplicaciones, que establecen el marco en el que se sitúa este trabajo (Sección 1.1). A continuación incluye, el problema abordado en este trabajo y la motivación del mismo (Secc. 1.2), es decir, ¿qué se trata de resolver con este trabajo?, y ¿por qué surge la necesidad del mismo?.

En la Sección 1.3 se incluyen los objetivos principales y secundarios que ha de cumplir la solución desarrollada. Por último, se encuentra la sección actual (Secc. 1.4) que muestra una visión global de lo que se presenta en esta memoria.

En segundo lugar, en el Capítulo 2 se detalla la infraestructura utilizada en la aplicación propuesta como solución al problema. Este capítulo recoge en diferentes secciones cada uno de los elementos que entran en juego. Por un lado, en la Sección 2.1 se explica el lado tierra y su composición. Por otro lado, en la Sección 2.2 se detalla el lado aire y sus componentes. Finalmente, en la Sección 2.3 se centra en el protocolo de comunicaciones entre ambos lados, tierra y aire.

En tercer lugar, en el Capítulo 3 se desarrolla el diseño y la implementación de la solución del problema. En una primera sección (Secc. 3.1) se explica detalladamente el diseño elegido y se explican también varias decisiones relevantes a la hora de seleccionar el diseño. En las secciones sucesivas (Seccs. 3.2) se describen la implementación de cada uno de los bloques de código identificados.

En el Capítulo 4, se presentan una serie de casos de uso que tratan de ejemplificar una posible situación real. Las diferentes secciones del capítulo corresponden cada una a un caso de estudio diferente. Los casos de uso que recoge esta memoria son XXXX.

Finalmente, en el Capítulo 5 se recogen las conclusiones extraídas con la finalización del trabajo y se evalúan los distintos objetivos iniciales propuestos (Secc. 5.1). A mayores, se exponen una serie de posibles vías de desarrollo futuro y de mejoras para la aplicación (Secc. 5.2).



# Capítulo 2

## Infraestructura

Durante este capítulo se explican las diferentes herramientas *software* y *hardware* que han servido de ingredientes en la realización de este trabajo. A la hora de proceder a desengranar los diferentes agentes que entran en acción, se realiza previamente una clasificación entre el lado tierra, el lado aire y la comunicación entre estos que se reflejan en las secciones de este capítulo.

En la Figura 2.1 se puede observar un esquema general que representa la arquitectura del sistema. El esquema muestra tanto el hardware como el software que integra el sistema, en los que se profundizará durante las secciones de este capítulo.

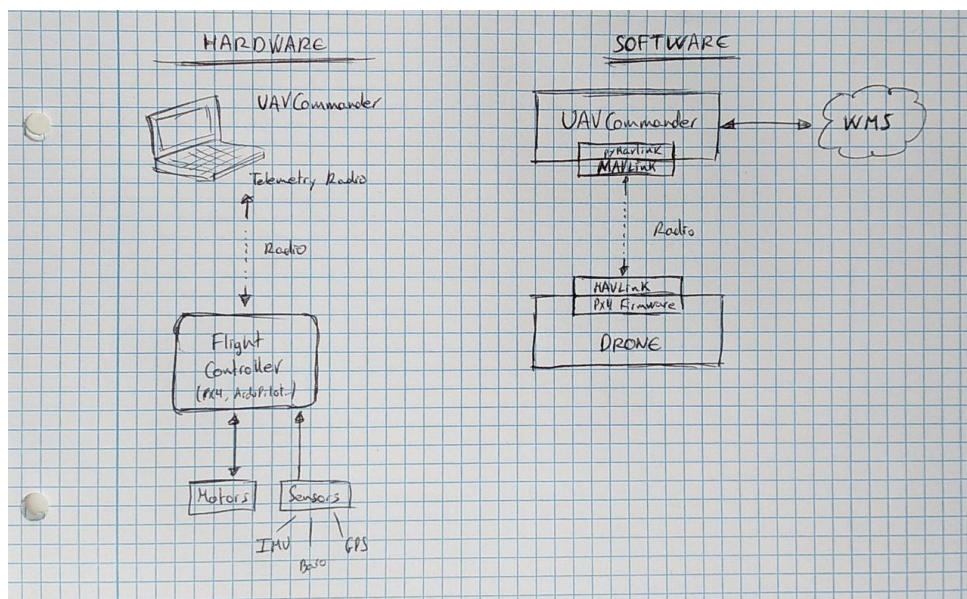


Figura 2.1: Esquema general del sistema.

### 2.1. Lado Tierra

El lado tierra se compone por un ordenador donde se ejecuta *UAVCommander*, la aplicación desarrollada. El software se idea como multiplataforma, por lo que el sistema operativo del ordenador puede ser cualquiera de las principales soluciones en el mercado. Sin embargo, la plataforma utilizada durante el desarrollo y las pruebas ha sido Ubuntu, la distribución de Linux basada en Debian. En concreto, la edición utilizada es la última

versión con soporte de largo plazo, **Ubuntu 18.04.3 LTS (*Bionic Beaver*)** [28].

El lenguaje elegido para el desarrollo de la aplicación ha sido Python [20]. El lenguaje creado a principios de 1990 por Guido van Rossum en los Países Bajos es un lenguaje de programación interpretado, interactivo y orientado a objetos. Sus principales ventajas que han motivado su elección para este proyecto son una sintaxis muy clara y su portabilidad. En concreto, la versión utilizada es **Python v3.6.9**.

Son varias las bibliotecas de Python utilizadas, entre ellas se quiere destacar **PyQt5** [26]. PyQt5 es un *binding* de la biblioteca gráfica Qt5 [7] para el lenguaje de programación Python. Qt es un framework multiplataforma escrito en C++ que permite el desarrollo de interfaces gráficas de usuario de forma sencilla.

Otra biblioteca relevante es **PyMavlink** [1], una implementación en Python del protocolo de comunicaciones MAVLink, el cual se explicará más adelante. El uso de esta librería facilita el uso del protocolo de comunicaciones, simplificando el uso de comandos y reduciendo el riesgo de cometer errores.

Para el manejo de mapas geo-referenciados se han utilizado diversas librerías. Un mapa geo-referenciado es aquel en el que conocemos o podemos calcular la posición real que representa cada píxel del mismo. Por un lado, se ha utilizado servicios de mapa web (WMS, *Web Map Services*) para obtener las imágenes como *Google Maps Platform* [27] u *Open Map Tiles* [25] a través de librerías como *urllib* para el manejo de URLs [23] o *PIL (Python Image Library)* y *Pillow*, para el manejo de imágenes [6].

Por otro lado, para la lectura de imágenes locales geo-referenciadas procedentes del Instituto Geográfico Nacional (IGN) [8] a través del Plan Nacional de Ortofotografía Aérea (PNOA) [30] se ha utilizado la librería **GDAL** [31]. GDAL es una biblioteca que permite leer más de 200 formatos de datos geospaciales ráster y vectoriales, entre los cuales se encuentran formatos como GeoTIFF o ECW, utilizados por la aplicación.

Otras librerías utilizadas son *math* [19], *numpy* [11], *threading* [21], *json* [18], *collections* [16], *datetime* [17], *time* [22] o *qfi* [5].

## 2.2. Lado Aire

El lado aire se compone por el drone. Se distinguen dos posibilidades, el drone puede ser real o simulado. La aeronave real puede ser de diversa índole, como ya se ha visto en el capítulo anterior. A lo largo de este proyecto se han utilizado las siguientes aeronaves:

- **3DR Solo Drone:** Cuadricóptero fabricado por 3DR Robotics [32]. Posee una controladora con firmware Ardupilot [34], una de las soluciones más usada en el mercado. Una de las principales ventajas que decidieron el uso de este drone en las primeras fases de vuelo frente a otros equipos del laboratorio es la existencia de controlador (o mando de control), el cual permite controlar la aeronave ante cualquier problema que no este previsto en el software.
- **PX4:** A completar.
- **Ala fija:** A completar.

Sin embargo, el uso de drone real se ha limitado a las últimas fases de desarrollo, que muy frecuentemente se ha visto sustituido por un drone simulado. Sobre el sistema propuesto en la sección anterior (Ubuntu 18.04.3) se ejecuta una simulación en software

# TEMP TEMP

(a) Drone 1.

(b) Drone 2.

Figura 2.2: Aeronaves utilizadas en el proyecto.

(**SITL**, *Software-In-The-Loop*) de una aeronave. El SITL permite enviar y recibir comandos a una controladora sin necesidad de tener el equipo real y evitando la pérdida de la aeronave en caso de error del programa.

El firmware utilizado como controladora es **PX4** [13], cuyo esquema de SITL se puede observar en la Figura 2.3. Además, la simulación se puede conectar con Gazebo [15] para hacer más completa la experiencia y mostrar el vuelo en un entorno de pruebas con elementos del mundo real. La versión utilizada durante el proyecto es **Gazebo9**.

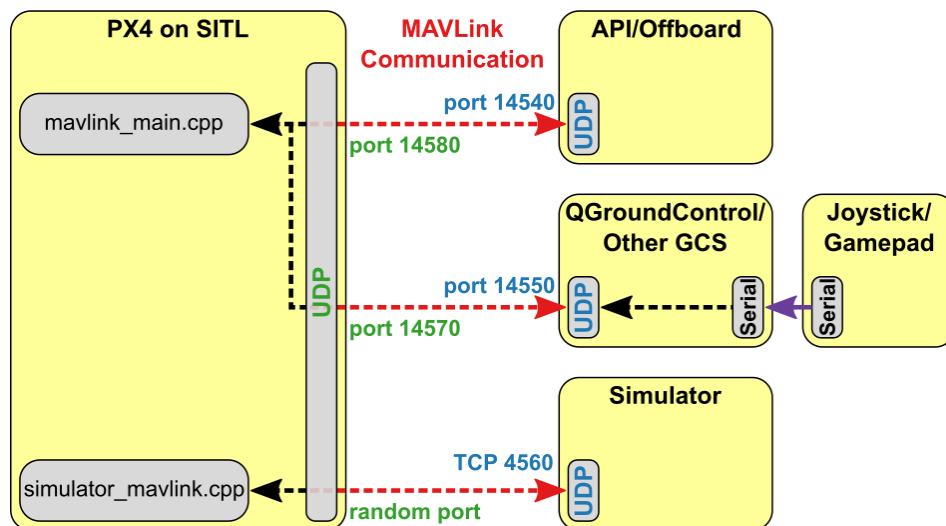


Figura 2.3: Esquema de PX4 sobre SITL [13].

## 2.3. Protocolo de comunicaciones

Tal como se ha adelantado en las secciones anteriores, el protocolo de comunicaciones utilizado es MAVLink [12]. MAVLink son las siglas de *Micro Air Vehicle Link*, un protocolo de comunicaciones muy ligero para el intercambio de mensajes con un drone y sus componentes a bordo. La versión utilizada es **MAVLink v2**.

Los extremos de la comunicación se han explicado en las secciones anteriores, siendo este protocolo el puente de comunicación entre el lado tierra y el lado aire. Es considerado como el protocolo estándar de comunicaciones en robótica aérea y las principales soluciones comerciales lo utilizan. Existen diferentes tipos de mensajes, comandos (de navegación, de *perform*, etc.), enumerados y microservicios.

## Capítulo 3

# Estación de Tierra, diseño e implementación

### 3.1. Diseño

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

### 3.2. Implementación

Sección temporal -¡Bloques de software



# Capítulo 4

## Resultados

### 4.1. Caso de estudio 1: Lorem ipsum...

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

### 4.2. Caso de estudio 2: Lorem ipsum...

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.





# Capítulo 5

## Conclusiones y líneas futuras

### 5.1. Conclusiones

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

### 5.2. Líneas futuras

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.



# Bibliografía

- [1] ArduPilot. Pymavlink. <https://github.com/ArduPilot/pymavlink>. Último acceso: 10-02-2020.
- [2] Antonio Barrientos, J Del Cerro, P Gutiérrez, R San Martín, A Martínez, and C Rossi. Vehículos aéreos no tripulados para uso civil. tecnología y aplicaciones. *Universidad politécnica de Madrid, Madrid*, 2007.
- [3] Aníbal Ollero Baturone. *Robótica: manipuladores y robots móviles*. Marcombo, 2005.
- [4] José Antonio Fernández Casillas. Navegación por posición para un avión autónomo con jderobot. <https://github.com/RoboticsLabURJC/2014-pfc-JoseAntonio-Fernandez>. Último acceso: 24-01-2020.
- [5] Marek Cel and Aitor Martinez (JdeRobot). Qflightinstruments. <https://github.com/JdeRobot/ThirdParty/tree/master/qflightinstruments>. Último acceso: 10-02-2020.
- [6] Alex Clark and Contributors. Pil. <https://pillow.readthedocs.io/en/stable/>. Último acceso: 10-02-2020.
- [7] The Qt Company. Qt5. <https://www.qt.io/>. Último acceso: 10-02-2020.
- [8] Ministerio de Fomento. Instituto geográfico nacional. <https://www.ign.es/web/ign/portal>. Último acceso: 10-02-2020.
- [9] Jonathan Ruiz de Garibay Pascual. Robótica: Estado del arte. *Universidad de Deuston. Número. Fecha*, page 54, 2006.
- [10] Grupo de robótica de la URJC. Twitter @roboticslaburjc. <https://twitter.com/RoboticsLabURJC?lang=es>. Último acceso: 24-01-2020.
- [11] NumPy developers. Numpy. <https://numpy.org/>. Último acceso: 10-02-2020.
- [12] Inc. Dronecode Project. Mavlink. <https://mavlink.io/en/>. Último acceso: 11-02-2020.
- [13] Inc. Dronecode Project. Px4. <https://px4.io/>. Último acceso: 11-02-2020.
- [14] Real Academia Española. Diccionario de la lengua española, 23.º ed. <https://dle.rae.es>. Último acceso: 02-02-2020.
- [15] Open Source Robotics Foundation. Gazebo. <http://gazebo-sim.org/>. Último acceso: 11-02-2020.

- [16] Python Software Foundation. Collections python library. <https://docs.python.org/3/library/collections.html>. Último acceso: 10-02-2020.
- [17] Python Software Foundation. Datetime python library. <https://docs.python.org/3/library/datetime.html>. Último acceso: 10-02-2020.
- [18] Python Software Foundation. Json python library. <https://docs.python.org/3/library/json.html>. Último acceso: 10-02-2020.
- [19] Python Software Foundation. Math python library. <https://docs.python.org/3/library/math.html>. Último acceso: 10-02-2020.
- [20] Python Software Foundation. Pyhton. <https://www.python.org/>. Último acceso: 09-02-2020.
- [21] Python Software Foundation. Threading python library. <https://docs.python.org/3/library/threading.html>. Último acceso: 10-02-2020.
- [22] Python Software Foundation. Time python library. <https://docs.python.org/3/library/time.html>. Último acceso: 10-02-2020.
- [23] Python Software Foundation. Urllib. <https://docs.python.org/3/library/urllib.html>. Último acceso: 10-02-2020.
- [24] Geodrone. Geodrone mapper. <http://geodrone.es/>. Último acceso: 24-01-2020.
- [25] Klokan Technologies GmbH and OSM community. Open map tiles. <https://openmaptiles.org/about/>. Último acceso: 10-02-2020.
- [26] Riverbank Computing Limited. Pyqt5. <https://riverbankcomputing.com/software/pyqt/intro>. Último acceso: 10-02-2020.
- [27] Google LLC. Google map server. <https://developers.google.com/maps/documentation?hl=es>. Último acceso: 10-02-2020.
- [28] Canonical Ltd. Ubuntu 18.04.3 lts. <https://ubuntu.com/download/desktop>. Último acceso: 09-02-2020.
- [29] FM Sánchez Martín, F Millán Rodríguez, J Salvador Bayarri, J Palou Redorta, F Rodríguez Escovar, S Esquena Fernández, and H Villavicencio Mavrich. Historia de la robótica: de arquitas de tarento al robot da vinci (parte i). *Actas Urológicas Españolas*, 31(2):69–76, 2007.
- [30] Instituto Geográfico Nacional. Plan nacional de ortofotografía aérea. <https://pnoa.ign.es/>. Último acceso: 10-02-2020.
- [31] OSGeo. Gdal. <https://www.osgeo.org/projects/gdal/>. Último acceso: 10-02-2020.
- [32] 3DR Robotics. 3dr solo drone. <https://newthreedee.wpengine.com/>. Último acceso: 11-02-2020.
- [33] Conyca S.L. Web. <http://www.conyca.es/>. Último acceso: 24-01-2020.
- [34] Ardupilot Development Team and Community. Ardupilot. <https://ardupilot.org/>. Último acceso: 11-02-2020.