

# Análisis y diseño de entorno sobre móvil para vuelos con drones

Aitor Marco Castro

**Resumen** — En los últimos años se ha producido una gran expansión a nivel comercial de los drones gracias a un gran avance técnico y una reducción considerable del coste de venta. Al mismo tiempo han aparecido un conjunto de propuestas de cuyo objetivo es dotar a los aparatos de nuevas herramientas y utilidades. El objetivo principal del proyecto es analizar, diseñar e implementar, a través de un proceso de Ingeniería del software, una plataforma de control y gestión de drones multipropósito que permita a los desarrolladores aplicar las nuevas ideas evitando crear un hardware y software específico para el U.A.V. Para lograr el objetivo se ha utilizado un dispositivo móvil Android que conectado al drone se utiliza como sistema de procesado y memoria además de aprovechar los sensores ya incorporados. Por otro lado, se utiliza un sistema web como interfaz de usuario comunicado vía red al móvil que permite al usuario controlar y gestionar el drone a lo largo de su recorrido.

**Palabras clave** — U.A.V , drone , móviles, planificador vuelo, sensores drone, multitarea drone.

**Abstract**— For the last years the drones' sale have been increased dramatically because of both a technical improvement and a considerably production's cost reduction. At the same time, some researches, after becoming aware of the drone's potential, have proposed to develop new tools and functionalities such as the forest fire surveillance. The main project's goal is to analyse, design and develop, through a Software Engineering process, a multipurpose platform for managing and controlling a drone. The system has been designed for allowing the developers to build up their ideas without having to make a specific hardware and software. In order to achieve the objective an Android smartphone has been programmed as a drone computer thanks to its powerful processing, memory capacity and its integrated sensors. Furthermore, a user interface web system connected to the drone by internet has been made to allow the users a complete control and manage of the device throughout the route.

**Index Terms**— U.A.V , drone, mobiles, fligh planner, drone sensors, drone multitask



## 1 INTRODUCCIÓN

**D**URANTE los últimos años ha habido un incremento considerable en el uso de dones a nivel comercial . Gracias a ello los aparatos han sido mejorados en fiabilidad, autonomía y habilidad de vuelo junto a una reducción de los costes de producción.

A pesar de la mejora substancial del sistema, el uso de los aparatos se destina principalmente a la grabación, el fotografiado y el control de vuelo aplicado sobre todo al ocio. Tanto el hardware como el software que incorporan son diseñados y desarrollados para cumplir una función específica dejando al margen la posibilidad de incrementar el gran abanico de funcionalidades que pueden aportar.

Existen proyectos cuyo objetivo principal es la búsqueda y el análisis de alternativas de uso ya sea el de rastreo, transporte de objetos o la automatización y toma de decisiones entre otros. Para poder desarrollar las funcionalidades mencionadas se necesita de un hardware y software especializados junto a unos técnicos con unas habilidades muy concretas y variadas, necesitando en última instancia un equipo de trabajo multidisciplinar. En consecuencia, existen diferentes ideas interesantes que no se llevan a cabo por la falta de una plataforma que facilite el desarrollo.

La iniciativa del proyecto radica en la necesidad de poder programar diferentes funcionalidades de un drone sin necesidad de recurrir al desarrollo de un sistema de hardware y un software (sistemas operativos, drivers...) específico para cada funcionalidad.

- 
- E-mail de contacto [aitormarco1990@gmail.com](mailto:aitormarco1990@gmail.com)
  - Mención realizada: Ingeniería del Software.
  - Trabajo autorizado por: Felipe Lumbreras Ruíz (CVC)
  - Curso 2014/15

## 2 ESTADO DEL ARTE

En sus inicios, los drones fueron desarrollados para realizar tareas sin el riesgo de perder vidas humanas. Generalmente, se han utilizado en el ejército en tareas de reconocimiento, soporte logístico y actualmente ataques quirúrgicos.

A nivel comercial, desde finales del siglo XX hasta la actualidad se han desarrollado modelos de aviones y helicópteros no tripulados aunque técnicamente no se incluyen en la familia de los drones ya que no son capaces de aterrizar y despegar automáticamente. Por otro lado, el avance de estos modelos ha permitido el desarrollo de los drones más utilizados actualmente a nivel comercial, los copteros.

Durante la última década, gracias a los avances técnicos acontecidos y a la aparición de los drones copteros se ha iniciado un proceso de desarrollo de dispositivos cuyo objetivo no es el vuelo acrobático o de exhibición si no tareas de fotografiado y filmado entre otras.

Actualmente están apareciendo nuevas iniciativas y modelos de negocio con el objetivo de explotar las ventajas que ofrecen los drones tanto a nivel técnico como económico.

### 2.1 Drones a control remoto

Pertencen al grupo de modelos más extendidos y de uso más generalizado. Existen gran cantidad diferentes de modelos desde el más pequeño y simple a drones de gran tamaño y con una autonomía considerable. El modelo más extendido con diferencia actualmente es el coptero. Tiene como características técnicas principales su gran estabilidad de vuelo y control a la vez que lento y poco acrobático.

Existen diferentes iniciativas para potenciar el uso de estos aparatos más allá de la simple grabación. Se están llevando a cabo investigaciones cuyo objetivo es desarrollar software capaz de analizar las imágenes captadas [1] [2] y en consecuencia obtener una información en diferentes situaciones como incendios, catástrofes naturales o en la investigación arqueológica.

### 2.2 Drones automatizados

Alejados del mundo militar, actualmente, no existen modelos de vuelo automatizado comercializados completos. Como drone automático se considera aquel capaz de realizar una ruta indicada previamente por el usuario sin necesidad de un control directo. A pesar de ser un campo en desarrollo con muchas posibilidades conviene remarcar que la automatización de un sistema de vuelo requiere de un buen control de fallos y de la confianza del cliente. La situación es compleja en lugares como Europa donde existe generalmente una alta densidad de población y en consecuencia un riesgo superior de un fallo con consecuencias críticas.

A pesar de las dificultades, existen diferentes iniciativas con el objetivo de impulsar avances significativos en este campo.

En primer lugar, algunas compañías han creado un hardware dedicado de automatización [3] [4] [5] con el objetivo de ser insertado en un drone radiocontrol. Además de la placa tanto aficionados como las propias empresas del sector han desarrollado software de control y gestor de vuelo [6] que se puede integrar en un PC de manera que se pueden crear tareas más complejas. Como hardware más destacado en el mercado se tiene el Ardupylot.

En segundo lugar, existen equipos que están desarrollando copteros capaces de realizar vuelos y acrobacias complejas mediante la toma de decisiones en tiempo real [7].

Por último, algunos equipos están diseñando drones capaces de trabajar interactivamente entre ellos para realizar tareas más complejas como la vigilancia o el transporte en grupo.

### 2.3 Interacción de dispositivos móviles en drones

Debido a la portabilidad, la potencia además de los diferentes sensores y cámaras de los dispositivos móviles, existen algunos estudios y proyectos que analizan las ventajas que ofrece la integración de los dos elementos.

Hoy en día, a nivel comercial, los dispositivos móviles se utilizan como un método de control simple para drones. Realiza la función de mando teledirigido a partir de una conexión Bluetooth. No obstante, las funciones son bastante limitadas.

A nivel de automatización mediante dispositivos móviles, los estudios o proyectos existentes están en fases muy iniciales. Como proyecto a tener en cuenta, existe un trabajo donde se aplica el dispositivo móvil en un drone como aparato de comunicación, para realizar fotografías y analizarlas [8].

Otro proyecto interesante integra el control de un drone automatizado con Ardupylot mediante móvil en lugar de utilizar el PC [9]. El código es público por lo que se puede trabajar y modificar gracias a las licencias de uso públicas establecidas. La ventaja principal del sistema radica en un incremento considerable de la portabilidad.

En todos los casos son proyectos experimentales que están en fases iniciales o en desarrollo. Como se puede apreciar, no existe una alternativa empresarial, comercial o algún proyecto en fases finales aplique la idea de utilizar un dispositivo móvil con

### 2.4 Iniciativas empresariales

Existen diferentes compañías que están desarrollando drones adaptados propiamente a sus necesidades ya sea automatizados o mediante control remoto. El caso más destacado es el desarrollo llevado por Amazon para utilizar drones como repartidores en zonas con baja densidad de población. Estas iniciativas ponen de manifiesto que existe un interés empresarial por los drones y que aún es un campo con mucho recorrido.

### 3 OBJETIVOS

La motivación del proyecto proviene de la necesidad de crear una plataforma que permita programar tareas diversas y complejas a un drone sin necesidad de desarrollar continuamente un hardware y software específicos. Principalmente la idea nació de la necesidad de aplicar los modelos de visión por computador y el posterior análisis en tiempo de vuelo para tareas varias como la detección y control de incendios o el control de cultivos entre otros.

A partir del concepto se establecen un conjunto de objetivos que orienten al proyecto hacia una solución válida.

En primer término, se ha observado que los dispositivos móviles contienen un hardware potente y a la vez ligero además de un conjunto de sensores y elementos como la cámara que montados sobre un drone permitiría facilitar el cumplimiento del objetivo base sin necesidad de un hardware y software especializados. Por ello, todo el proceso se debe realizar utilizando un dispositivo móvil como controlador del drone y gestor de las tareas.

En segundo término, se requiere que el sistema se diseñe aplicando metodologías de ingeniería del software con el objetivo de conseguir un sistema modular de calidad, cumpliendo las expectativas requeridas esperadas, además de un riguroso análisis y posibles alternativas viables. Por consiguiente todo el proceso debe estar formalizado mediante documentación y diagramas estándar.

En tercer término, el proceso de ingeniería del software debe ofrecer no solo una solución de corto alcance sino el camino que se debería tomar a lo largo de un hipotético desarrollo del proyecto completo, ofreciendo soluciones futuras a problemas, necesidades y funcionalidades planteadas durante el análisis.

En última instancia, aunque el proceso se fundamente en un proceso de análisis y diseño se requerirá como objetivo fundamental el desarrollo de un código de demostración que justifique y pruebe la viabilidad futura del proyecto además de los conceptos y requerimientos planteados más importantes.

### 3 METODOLOGÍA

Se ha utilizado como base los procesos clave de ingeniería del software en especial la fase de análisis y diseño junto a un proceso de desarrollo del software.

Como metodología de trabajo y organización se ha decidido aplicar un sistema mixto entre un desarrollo en cascada y Scrum. Aunque sean dos sistemas prácticamente contradictorios cada uno ofrece ciertas ventajas que se requerían para el proyecto.

La base se ha establecido usando el método en cascada aplicando, en primer término, un estudio de viabilidad, seguido de un proceso de obtención de requerimientos y finalizando con el diseño del proyecto. Todo este proceso se formaliza mediante una documentación y diagramas estándar.

Por otro lado, debido a la inexperiencia previa con ciertas tecnologías y Frameworks es necesario realizar un posterior refinamiento en los diferentes procesos, sobre todo durante la fase de desarrollo donde se pone a prueba todo el sistema planteado. Para realizar el método Scrum se hicieron esfuerzos de una o dos semanas. Cada esfuerzo empieza y finaliza a partir de una reunión con el tutor del proyecto. En la reunión se establecen las bases de las mejoras y cambios que se deben realizar durante el siguiente periodo.

El proceso finaliza cuando se obtiene un código de demostración que cumpla con los requerimientos básicos planteados y los objetivos iniciales.

#### 3.1 División del trabajo

Antes de iniciar el proceso de análisis y desarrollo del sistema se establecen unas bases y una organización que se deben considerar a lo largo de todo el proyecto.

Durante el procedimiento se desarrollan las primeras ideas, conceptos y objetivos clave. Se realiza un estudio preliminar de las diferentes tecnologías, sistemas y recursos disponibles, sin analizarlas en detalle, y se formaliza un primer boceto con los conceptos básicos aplicados.

Por consiguiente, una vez desarrollada la idea preliminar se divide el proyecto en un conjunto de tareas específicas. Cada tarea es descrita y analizada con el objetivo de conseguir una organización del trabajo que facilite el desarrollo del sistema y ayude predecir tanto las necesidades futuras como el alcance del proyecto.

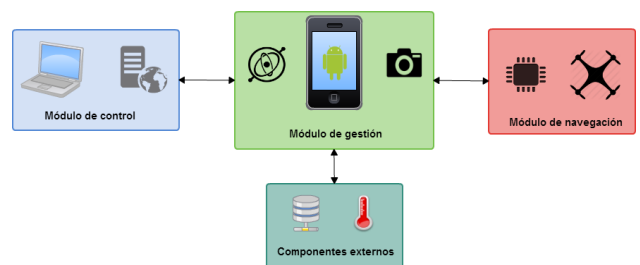


Figura 1: Descripción inicial del sistema con las ideas y conceptos claves desarrollados.

La división principal está formada por tres tareas y cada una de ellas es descompuesta en conjuntos más pequeños y manejables.

En primer lugar se realiza el proceso de análisis que engloba el estudio de viabilidad, el estado del arte y la obtención de requisitos.

En segundo lugar se diseña e implementa el sistema organizando el proyecto a partir de los tres módulos en el que se descompone el sistema. Es el proceso más extenso de todos y su alcance es superior al que se realiza en este estudio.

Finalmente, se realizaría un proceso de test exhaustivo con todo el sistema en funcionamiento. No pertenecen a este grupo las pruebas puntuales o de secciones del sistema cuya realización corresponde en las tareas de desarrollo. Las pruebas corresponden a simulaciones de fallos, pruebas de rendimiento y sobretodo test de fiabilidad del sistema.

### 3.2 Estudio de viabilidad

El proceso llevado a cabo permite establecer las directrices iniciales del proyecto, las tecnologías y alternativas disponibles además de las limitaciones. Para este proyecto, el estudio de viabilidad es básico ya que el éxito depende directamente del correcto uso de un conjunto de tecnologías muy variadas con un uso poco habitual de ellas. Ciertas carencias como la falta de acceso a un dron, las limitaciones en el alcance o la no existencia de proyectos similares hacen necesario la búsqueda de métodos y alternativas que permitan asegurar al máximo posible el éxito antes de iniciar el desarrollo.

Del proceso se pueden extrapolar un conjunto de conclusiones claras y bien justificadas que han permitido orientar el proyecto con seguridad.

En primera instancia, el dispositivo móvil se utilizará junto a un hardware dedicado de control del dron. Los dispositivos móviles, sobre todo si se desarrolla en Android [10], no se pueden considerar fiables a la hora de utilizarlos como controlador directo del dron. Son sistemas que a veces fallan, se colapsan y provocan retrasos en la ejecución o realizan otras tareas prioritarias no controladas. Otro factor importante a tener en cuenta es la falta de control sobre el tipo de hardware que se utilizará. Alguien podría poner un dispositivo móvil o con un hardware antiguo, de baja calidad o simplemente con problemas como la batería, provocando un repentino apagado del sistema.

Por ello ha de existir un hardware dedicado básico que siga funcionando si el sistema falla y que permita controlar el dron hasta alcanzar una situación segura donde se pueda o bien finalizar el vuelo o reiniciar el sistema. Actualmente, como se especifica en el estado del arte, ya existe un proyecto que unifica la placa de vuelo con un dispositivo móvil usado como control remoto, por lo que la viabilidad de esta alternativa, en lo que respecta en la comunicación entre placa y móvil, está prácticamente asegurada en entornos Android.

En segunda instancia, se han analizado las diferentes tecnologías para asegurar que se podría usar un dispositivo móvil como control del dron automático. Android contiene gran cantidad de funcionalidades para controlar y ejecutar tanto los diferentes sensores, cámaras y extras que contenga como realizar cualquier tipo de tarea. El hardware, en general, es suficientemente potente para trabajar con tareas complejas y multi-hilo además de contener una tarjeta gráfica que permitiría optimizar los procesos de análisis de imágenes y video. Uno de los factores más críticos era la posibilidad de utilizar mapas en el sistema de control tanto en el muestreo información como en la planificación de la ruta. Por ello se estudiaron

diferentes alternativas. El resultado del proceso es la utilización vía el API de JavaScript de Google Maps [11]. Como tecnología de desarrollo se optó por Struts 2 [12] [13] un Framework de desarrollo web realizado en JAVA [14].

Con la combinación de las dos tecnologías se consigue un sistema de mapeado muy potente, poder conectarse vía web, siempre y cuando el Framework se ejecute en un servidor web, al sistema de control desde un móvil o una tableta y por último utilizar JAVA como lenguaje base en todo el proyecto.

En todo momento en el documento se reflejan las diferentes alternativas posibles analizando las ventajas, inconvenientes y riesgos.

### 3.3 Casos de uso

Antes de realizar un proceso de obtención de requisitos se establecen un conjunto de funcionalidades del sistema a partir de la actuación de agentes externos. El documento se centraliza en el módulo de gestión como sistema central siendo el módulo de control y sensores los actores externos. El objetivo es diseñar un conjunto de casos claves que constituyan el núcleo del sistema.

Cada caso viene acompañado de una ficha donde se escriben las condiciones previas y posteriores de uso, el conjunto de tareas que conforman el caso o la descripción entre otras. El proceso de análisis se realiza en un nivel de abstracción alto, por lo que se definen las tareas y descripciones de forma general.

De los conceptos obtenidos de este proceso se obtiene la base de algunos de los elementos más importantes del proyecto como el concepto de Triggers y Acciones o el uso de Reflection .

### 3.4 Obtención de requisitos

Durante el proceso de obtención de requisitos, una vez elegidas las tecnologías a aplicar, se establecieron las bases del proyecto. Se han tenido en cuenta tanto las funcionalidades como las restricciones y la estructura con la que se construirá el sistema. Cada requerimiento se identifica con un número, el tipo, la descripción y los requerimientos previos. Este último es muy importante ya que establece el orden en el que se deben ir cumpliendo a lo largo del desarrollo del proyecto. Además durante todo el proceso de diseño y desarrollo se realiza un seguimiento del cumplimiento de los requisitos. La trazabilidad se crea a partir de tres estados definidos. Los estados corresponden a requisito cumplido, no cumplido y un tercero que indica que se está cumpliendo el requisito pero que se debe mantener el control a lo largo del proyecto ya que su cumplimiento definitivo solo se tendrá en cuenta al finalizar el proyecto.

Cabe destacar que se decidió descomponer el sistema en tres módulos principales. Cada módulo realiza una tarea concreta y tiene una mínima interacción con el resto. Toda la organización tanto a nivel de requisitos como de diseño y desarrollo del proyecto se estructura alrededor de la siguiente organización.

El primer sistema corresponde al módulo de control. Corresponde a la interfaz del usuario donde se controla y se da instrucciones al dron. El módulo está formada por el Framework Struts 2 y Google Maps como tecnologías base además de las tecnologías web.

El siguiente a tener en cuenta es el módulo de gestión. Es el sistema incorporado en el dispositivo móvil mediante Android. Se encarga de gestionar las diferentes tareas que debe realizar el aparato a lo largo del vuelo.

Por último, se constituye el módulo de navegación. Este sistema contiene el hardware dedicado de navegación y el software con el que se opera. Aunque es una estructura hecha por terceros suelen incorporar un software altamente configurable y con la posibilidad de ser reprogramados por lo que puede constituir una fase más de desarrollo del proyecto.

Por otro lado también se establecen las bases de la organización de las conexiones mediante la estructura XML [15]. Aunque parezca un proceso menor esta organización es la base de gran parte del diseño posterior. Se establece un conjunto de reglas y recomendaciones de los patrones y configuraciones que se deben seguir al establecer nuevas funcionalidades en la interacción de los dos módulos.

De este proceso se generan el documento de requisitos a partir del estándar ofrecido por la institución IEE y un documento que establece el protocolo a seguir en las comunicaciones

### 3.5 Diseño

El proceso de diseño ha sido la fase más extensa del proceso de Ingeniería del Software. Desde el momento en que se establecieron los objetivos se decidió que el sistema debía estar bien diseñado para poder prever y facilitar futuras añadidos y mejoras además de tener un software de calidad. El proceso se ha realizado principalmente en las fases anteriores a la fase de desarrollo aunque se han hecho sucesivos refinamientos a lo largo de todo el proyecto una vez conocidos los resultados obtenidos. Durante el proceso de diseño se han utilizado diferentes patrones de diseño comunes [16] además de añadir variaciones e ideas clave.

#### 3.5.1 Módulo de gestión

Todos los procesos que se llevan a cabo en el módulo comparten un flujo de ejecución común que se ha tenido en cuenta a la hora de realizar el diseño.

Básicamente, toda acción se realiza a partir de una petición del usuario (módulo de control) o los sensores del dispositivo. La consecuencia de esa petición es una acción cuya respuesta debe ir dirigida al sistema de control o al de navegación.

#### 3.5.1.1 Acciones y Triggers

Uno de los flujos principales de ejecución proviene de los eventos generados por los sensores (ver figura 2). El objetivo de un sensor, planteado para este proyecto, es localizar una variación en el sistema o en su entorno y comunicarlo.

Para este proyecto se trabaja únicamente con sensores integrados en el dispositivo móvil [17] aunque también en un futuro desarrollo, el sistema debería aceptar sensores externos y eventos generados por el módulo de control.

Para conseguir esta estructura básica se han diseñado dos elementos principales, los Trigger y las acciones.

Por un lado los Triggers tienen como objetivo interactuar directamente con los sensores y generar el evento deseado. Cada uno corresponde a un evento único y diferenciado del resto. No hay ninguna limitación en cuanto a la cantidad de ellos que pueden analizar un mismo sensor. Cada trigger puede contener a su vez un número indeterminado de acciones.

Cada acción corresponde a un conjunto de instrucciones cuyo objetivo principal es dar respuesta al evento que la ha activado. Como ejemplo a destacar, un movimiento brusco podría generar una acción de enviar un mensaje de situación peligrosa y a su vez iniciar una grabación o fotografía del suceso. En este caso se tendría un Trigger unido a dos acciones. El patrón base que se ha seguido es el Observer [18], siendo la acción el observador y el Trigger el observado.

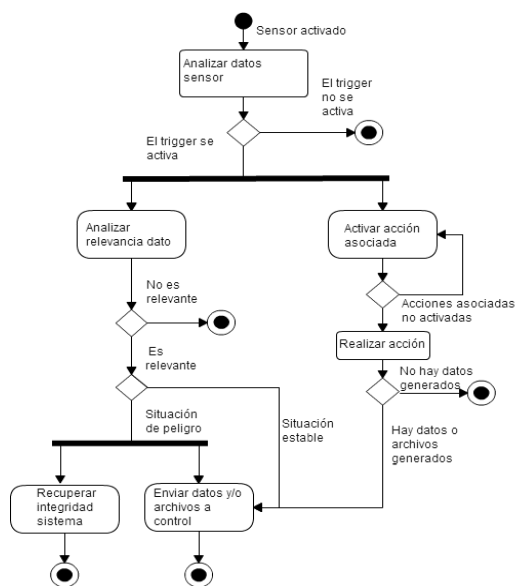


Figura 2: Diagrama de actividades del sistema de Triggers y Acciones. Se muestra desde la generación de un evento hasta la transferencia de la información al módulo de control.

Cada vez que se quisiese programar un Trigger y/o acción el usuario debería extender la clase y programar únicamente la función que analizaría el sensor y la función que se ejecutará en la acción, el resto del proceso se automatiza. Siguiendo el esquema del sistema se cumplen las siguientes reglas.

- Toda acción puede ser unida a cualquier Trigger.
- Una acción puede ejecutarse en el hilo de ejecución principal o en uno independiente.
- Un Trigger puede tener acciones repetidas o diferentes.
- Varios Triggers pueden compartir una misma acción, aunque cada una de ellas corresponde a una instancia diferente.
- Solo existe un Trigger de cada tipo como máximo.
- Una acción puede ejecutarse o una única vez a lo largo del vuelo o indefinidas veces
- Un Trigger puede ejecutarse ilimitadas veces o únicamente una vez durante el vuelo.

Existe, no obstante, un caso excepcional. Para poder generar un evento de geolocalización y que contenga una estructura lógica adecuada se ha decidido que cada Trigger de localización este directamente relacionado a un punto concreto de coordenadas. En esta situación existirán N triggers siendo cada uno un punto en el mapa. Cada uno de ellos únicamente se ejecutará una vez realizando todas las acciones que tienen asignadas. Estos triggers están unidos a un Manager que gestiona tanto el posicionamiento generado por el GPS y la red como el conjunto de Triggers que se puedan ejecutar. Se activará uno de ellos cuando el dispositivo este dentro de un radio de distancia, indicado por el usuario, de las coordenadas indicadas. La activación se realizará una única vez aunque nada impediría realizar múltiples ejecuciones en futuras versiones.

### 3.5.1.2 Entrada de datos

Para gestionar las múltiples opciones que se pueden desarrollar en la entrada de datos se ha decidido utilizar una versión del patrón Interceptor [19]. En este caso se tiene una clase que gestiona un conjunto de filtros. Cada filtro es una extensión de una clase abstracta que conforma la estructura principal.

Cuando se reciba un mensaje externo, principalmente del módulo de control, el gestor enviará a todos los filtros consecutivamente el mensaje. A su vez cada filtro se responsabiliza de analizar una parte correspondiente del mensaje siempre que exista.

Para incrementar la capacidad modular del sistema, el mensaje está estructurado mediante una organización similar a los filtros. El mensaje está formado por un XML de dos niveles, como mínimo, de etiquetas. El primer nivel indica el filtro encargado de analizar la información encapsulada por la etiqueta. El segundo nivel correspon-

de a las etiquetas que definen cada parcela de información que analizará el filtro. Gracias a este sistema se ha creado un proceso de búsqueda automatizado en el que cada filtro se encarga de analizar su parte correspondiente únicamente indicándole que etiqueta debe buscar.

Cada filtro es totalmente independiente del resto y pueden generar tareas totalmente diversas entre ellas como gestión en la seguridad de las comunicaciones, responder a peticiones o simplemente analizar los datos recibidos. Gracias a este sistema se consigue modular la gran cantidad y diversa información que pueda venir del sistema de control focalizando al programador únicamente a la funcionalidad interesada. Además se consigue una organización más clara de la información en general.

### 3.5.1.3 Salida de datos

La salida de datos (ver figura 3) del sistema de gestión debe tratar tanto la información generada por el sistema como los archivos (imágenes, vídeos, audio...). Otro factor importante a tener en cuenta es el orden de prioridad en la transmisión de la información ya que un mensaje de error debe ser transmitido de inmediato mientras que una imagen puede tener un cierto tiempo de espera. Para ello se han utilizado dos canales de transmisión, el primer de ellos utiliza un formato XML y se encarga de transmitir toda la información del estado del sistema, el segundo, por otro lado, se encarga de la transmisión los archivos generados o cualquier otra información extra.

Todo elemento que se quiera enviar debe heredar de la clase Paquete. Este elemento contiene todas las funciones básicas necesarias que permiten transmitir información de manera ordenada, controlada y aprovechando la estructura XML diseñada.

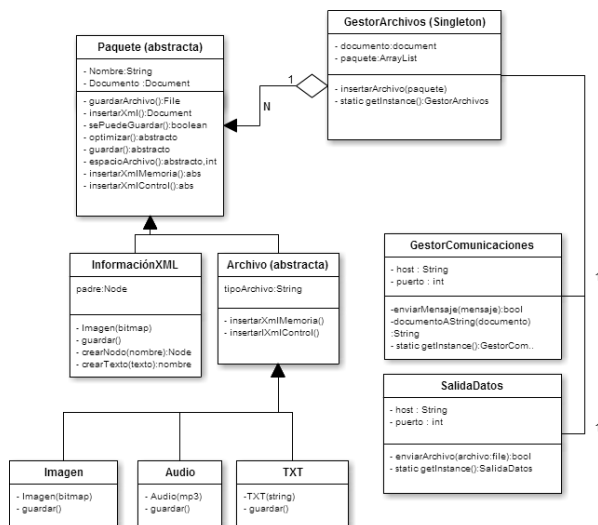


Figura 3: Diagrama de clases del sistema de salida de datos. Se han incluido las funciones públicas más relevantes.

Una especialización de esta funcionalidad se encuentra en la clase Archivo que permite, a través de la implementación de un grupo de funciones abstractas que deben ser completadas en las clases hijas, automatizar, a partir de una cola FIFO, el envío de archivos en diferentes formatos. Además de transmitir, el sistema también detecta las condiciones por las cuales el envío no se puede realizar, sea por un problema de conexión, porque el archivo es demasiado grande o simplemente porque el usuario no lo requiere. En ese caso el archivo es guardado en memoria y la información del estado se enviará una vez el sistema esté disponible. Todo el proceso se realiza en un hilo de ejecución independiente para evitar colapsar el sistema en transmisiones largas.

Todo envío de archivo queda automáticamente reflejado en el estado del sistema. Se indica cuando el archivo está en cola, cuando se ha iniciado la transmisión y la finalización de ella. Según el archivo se pueden personalizar otros estados deseados.

Hay que destacar también que el estado del sistema no se transmite al instante si no bajo un intervalo definido. Esto se hace por dos motivos. En primer término, se evita una saturación en la transmisión comprimiendo toda la información en único mensaje en vez de transmitir cada vez que se genera un estado diferente. En segundo término, permite mantener una conexión constante, en caso de que no se generen estados nuevos, permitiendo al sistema de control saber en todo momento si el sistema de gestión (el dron) sigue operativo o hay conexión.

Por otro lado al ser muy común que la conexión pueda fallar durante el vuelo se ha implementado un sistema de reconexión automática que se realiza una vez se confirme que hay conexión con el sistema.

En conclusión, cuando un programador necesite enviar un archivo o mensaje únicamente deberá seleccionar la clase implementada correspondiente añadir la información básica necesaria, un archivo Bitmap en caso de una imagen por ejemplo y el sistema automatizará todo el resto de decisiones.

#### 3.5.1.4 Reflection

Reflection [20] es una tecnología de JAVA utilizada en los Frameworks y librerías más importantes como Android, Struts 2 o Hibernate. Utilizando la tecnología adecuadamente se puede crear una instancia de clase únicamente mediante una cadena de caracteres que referencie su nombre. Además de la instancia también se puede cargar el constructor, listar las funciones y utilizarlas sin necesidad de conocer previamente como era la clase.

La tecnología mencionada se ha aplicado ampliamente en todas las facetas del módulo para conseguir un sistema altamente modular y evitar que un programador deba tocar la estructura principal cuando quiera añadir una nueva funcionalidad. Para ello se ha programado un ges-

tor que, a partir de un XML que el usuario puede editar siguiendo unas reglas estipuladas, cree la estructura del proyecto deseado sin necesidad de programar código previamente. Es decir, toda la estructura básica del código se organiza mediante un archivo externo de configuración.

Aplicado en los filtros el usuario puede especificar que filtros deben ejecutarse y en qué orden. Si se quiere añadir un nuevo filtro únicamente se debe crear una nueva clase heredada de Filtro e implementar las funciones abstractas. A partir de ahí únicamente se deberá especificar en el XML su existencia sin necesidad de modificar el código de la estructura.

Donde la tecnología mencionada tiene más potencia y recorrido es en el módulo de acciones y Triggers. Cada Trigger y acción existentes se registran en el documento XML identificados por un nombre único y la referencia a la clase.

Cuando el módulo de control y gestión se conecten entre sí se envía el fichero correspondiente con toda la información de la estructura. Con ello el sistema de control puede indicar al usuario que acciones y Triggers existen automáticamente sin necesidad de instanciarlos previamente a nivel de código. De retorno el sistema de control indicará que Triggers deben ejecutarse y que acciones se les asignará. Aprovechando todavía la tecnología Reflection una vez se reciba que triggers y acciones se quieren realizar el sistema creará automáticamente las instancias necesarias.

La consecuencia de todo ello se expresa en el siguiente ejemplo. Un usuario tiene acceso a un conjunto de acciones, triggers y filtros programados por un tercero. Únicamente deberá añadir la clase correspondiente al proyecto y una vez se haga referencia en el XML el sistema podrá crear instancias de los nuevos elementos automáticamente.

### 3.5.2 Módulo de control

Para realizar el módulo de control (ver figura 4) se ha utilizado el Framework Struts 2. Es un conjunto de librerías que permiten realizar un proyecto web con el patrón modelo-vista-controlador utilizando JAVA en el lado del servidor y las tecnologías web, básicamente HTML [21], CSS [22] y JavaScript [23] como tecnologías básicas y la librería JQuery [24] como apoyo para realizar tareas y peticiones dinámicas en el lado del cliente.

#### 3.5.2.1 Organización del servidor

El lado del servidor se ha diseñado utilizando dos estructuras muy concretas que forman el núcleo de la organización.

En primer lugar, las comunicaciones de entrada se han diseñado aplicando el patrón Interceptor (ver 3.3.1.2). En consecuencia se comporta exactamente igual que la entrada de datos del módulo de gestión.

En segundo lugar, toda la información generada se distribuye a lo largo de un conjunto de controladores donde cada uno de ellos representa una estructura de datos como sensores o la estructura de Triggers y acciones. Cada controlador debe estar programado para que responda a peticiones realizadas por el cliente. Cada petición incluye una instrucción que debe realizar el controlador pudiendo ser un envío de información, una petición de información o una orden para ejecutar una rutina concreta. Debido a que la información generada por el módulo de gestión es, por ahora, bastante limitada se ha decidido utilizar archivos en formato TXT para almacenar la información simplificando el proceso ante la opción de crear una base de datos completa.

Por último, se ha añadido una *plugin* de Struts 2 [25] para que se puedan realizar peticiones asíncronas mediante JSON entre cliente y servidor utilizando las tecnologías JAVA y AJAX [26].

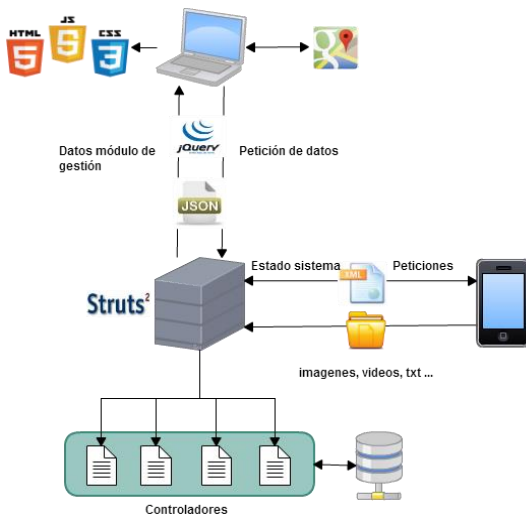


Figura 4: Esquema de la organización del módulo de control y las diferentes tecnologías utilizadas.

### 3.5.2.2 Organización de la interfaz gráfica

El lado del cliente básicamente consta de todo el sistema visual que facilite al usuario la gestión y planificación del vuelo. Para ello se ha utilizado tanto las tecnologías webs más utilizadas en el desarrollo web además de añadir la librería Google Maps. El interfaz de usuario se divide en dos mitades. Por un lado hay todos los formularios de configuración y datos del drone, por el otro un mapa con toda la información geográfica dibujable.

Hay dos estados diferenciados a tener en cuenta, en los dos casos la estructura es la misma pero las funcionalidades difieren considerablemente. En la primera fase el objetivo es crear un sistema de configuración de vuelo (ver Apéndice 1) donde se utiliza el mapa para crear una ruta y los formularios para crear la estructura de triggers y acciones. En el segundo estado en cambio el interfaz pasa a ser un sistema de control de vuelo (ver Apéndice 2). Los formularios se sustituyen por un sistema de mues-

tro de información básica de vuelo. El mapa por otro lado se utiliza para indicar la posición del drone y de los diferentes eventos ocurridos.

### 3.5.2 Desarrollo

En la fase de desarrollo se han implementado los requerimientos más importantes y los diseños realizados a lo largo del proceso de Ingeniería del Software. El sistema además se ha implementado siguiendo todo el proceso de diseño realizado en la fase anterior. Al final del proceso se ha obtenido un código demostración que permite aplicar todas las facetas del sistema e incorpora un conjunto de casos de ejemplos que visualizan el éxito del proceso.

A nivel web gracias a la librería de Google Maps se ha conseguido tener un mapa mundial donde se puede dibujar cualquier ruta deseada tanto mediante mapas geográficos como el servicio de mapeado por satélite. Cada vez que el usuario apriete en el mapa se crea un nuevo punto unido al anterior por una línea. A cada punto el usuario puede seleccionar la acción que quiere realizar. Una vez iniciada la ruta el mapa sitúa al drone en la última posición indicada por el GPS y además indica con un icono todas las acciones importantes realizadas, indicado previamente mediante una acción destinada a ello.

Otro elemento importante en web es la posibilidad de seleccionar cualquier Trigger y acción de una lista que se genera automáticamente. Todas las relaciones entre ellos se muestran en una tabla dinámica.

A nivel visual la web es bastante simple ya que el objetivo principal es tener un sistema de prueba para las funcionalidades diseñadas.

Respecto al módulo de gestión se han creado diferentes acciones y Triggers de demostración. Se ha desarrollado una acción que realiza fotografías con la cámara frontal y las envía a la cola de archivos y una acción que genera la localización donde se ha disparado el Trigger. A nivel de Trigger se han añadido un detector de dispositivos Bluetooth externo que localiza nuevos dispositivos, no analizados anteriormente, un disparador temporal que se activa cada cierto intervalo y un detector de movimientos bruscos que se activa al mover el dispositivo mediante un gesto brusco. Por otro lado el Trigger de localización se activa cada vez que se haya llegado al punto indicado de la ruta marcada, dentro de un intervalo de error definido.

Se han conseguido además una importante cantidad de datos relevantes que permiten demostrar no únicamente que el móvil contiene un mínimo de sensores aptos para obtener información de vuelo sino que estos pueden ser transmitidos y mostrados al interfaz del cliente durante el proceso de vuelo.

A nivel de transmisión los datos a destacar son el tipo de conexión disponible, WIFI, 3G o ambos, y el conjunto de sensores activados, Bluetooth o GPS. Por consiguiente, se tienen un conjunto de datos de posicionamiento como la localización del drone, indicado en el mapa mediante un icono, la velocidad obtenida por GPS o la distancia mínima hasta llegar al siguiente punto. Cabe destacar que en las últimas fases del proyecto se ha encontrado un



sensor que permite establecer el nivel rotación del dispositivo sin recurrir a un giroscopio, sensor no disponible en el dispositivo con el que se trabaja, aunque con menos precisión y ciertas limitaciones. Hay que tener en cuenta además que la autonomía del vehículo es un dato esencial, por ello se han añadido datos sobre el estado de la batería y la existencia de algún cargador auxiliar insertado. Además de todos estos se ha añadido también la velocidad de transmisión del último archivo.

En lo que se refiere a la salida y entrada de datos en los dos módulos hay que destacar la implementación de un sistema de reconexión a la red en caso de fallo de comunicación.

## 4 Resultados

Un proyecto de análisis y diseño del software no se puede valorar mediante unos resultados científicamente cuantitativos que muestren el correcto o mal devenir del proceso. Para valorar los resultados hay que tener en cuenta unos resultados cualitativos mediante el cumplimiento objetivos propuestos y sobretodo los requerimientos establecidos.

En primer término, el objetivo principal es demostrar que se puede utilizar un dispositivo móvil como apoyo de control y gestión de un dron. Por un lado, debido a que fue imposible de poder trabajar con el simulador por estar fuera del alcance real del proyecto, no se puede demostrar al cien por cien que el sistema planteado se podrá integrar en un dron automatizado. Por otro lado, gracias al exigente estudio de viabilidad realizado y los resultados obtenidos en el proyecto demuestran que se podrían conseguir las perspectivas iniciales planteadas si se siguiera con el desarrollo del proyecto.

Básicamente se puede afirmar que ya existen proyectos que están trabajando con la combinación de los dos elementos con el apoyo de un hardware dedicado como Ardupilot. Además, una vez se ha finalizado la estructura del proyecto se puede afirmar que se puede realizar un intercambio de información tanto a nivel de geolocalización como del estado del sistema. Por otro lado el dispositivo puede reaccionar a elementos externos como una posición concreta, un fallo del sistema o un movimiento brusco generando una acción como una fotografía o un mensaje de error.

El uso de Google Maps como elemento de apoyo constituye un punto clave para garantizar el éxito del proyecto. En consecuencia, se ha conseguido obtener la funcionalidad deseada y se han cumplido con las expectativas requeridas a nivel de localización.

A pesar de que la funcionalidad es un factor importante se quería un proyecto construido mediante un proceso de Ingeniería de Software que garantice un sistema modular, la facilidad de ampliación y mejora, además de factores de calidad como la robustez o la facilidad de uso.

Teniendo en cuenta que la mayoría del proceso se realiza en el módulo de gestión es donde se ha trabajado con más esfuerzo para garantizar los objetivos anteriores propuestos. Se puede afirmar que las expectativas iniciales a nivel de diseño han sido claramente cumplidas e

incluso superadas. El objetivo inicial era tener un código bien estructurado y modular. Gracias a los patrones de diseño y el uso de Reflection además de la aplicación de estándares en la comunicación ha permitido crear una estructura cuyo usuario puede ampliar y modificar sin necesidad de comprender todo el proyecto pudiéndose así especificar en un sector. Por otro lado el usuario únicamente debe modificar el código que quiera añadir por lo que la estructura no se verá alterada evitando muchos futuros errores y problemas. Además al estar dividido a partir de una estructura lógica el sistema es bastante simple de entender a partir de los diagramas explicativos realizados.

A nivel del módulo de control se ha dado más prioridad a la funcionalidad ya que, como se ha dicho anteriormente Struts 2 ofrece una organización consistente. No obstante, se ha conseguido un código bien organizado que permite al usuario entender fácilmente donde esta cada sección y como debe proceder para aplicar mejoras. Todo el flujo de ejecución está diseñado a partir de una estructura muy similar al módulo de gestión y al compartir el mismo lenguaje hace que un usuario pueda aprender los dos sistemas con mayor rapidez.

Debido a que el objetivo es crear un sistema altamente automatizado, la robustez es un factor importante. Hay que evitar la situación de un fallo de sistema en pleno vuelo. Para ello se ha intentado predecir diferentes eventualidades como una caída de conexión, un fallo en el Bluetooth o problemas de cobertura del GPS. Además se ha intentado crear un código robusto que no genere excepciones que finalicen la ejecución. Aun así hay que admitir que Android es una librería altamente inestable y se ha comprobado que fácilmente genera errores irre recuperables. Por otro lado el sistema de reconexión permite mostrar, hasta un cierto grado, que el sistema puede auto-recuperarse de situaciones adversas.

## 5 Conclusiones

Se ha conseguido cumplir con las expectativas iniciales creando un código de demostración sólido y bien planificado con unas funcionalidades básicas adecuadas según los requerimientos. En todo momento se ha mantenido unas expectativas realistas dado que el proyecto completo tiene un alcance mucho mayor. No obstante, se ha tenido en cuenta un orden de prioridad que permita asegurar al menos el cumplimiento de la estructura base que permita el añadir futuras mejoras. El código realizado es modular, de fácil mantenimiento y ampliable.

En definitiva, todo el proceso ha servido para demostrar que el sistema es realizable, que tiene un recorrido futuro con un alto porcentaje de éxito en sus funcionalidades. Por otro lado también pone de manifiesto que para conseguir un proyecto totalmente definitivo y apto para el uso real de drones requeriría de un alto esfuerzo de trabajo y un personal con un conocimiento técnico específico en ciertas materias sobre todo en las comunicaciones y control directo con el dispositivo de vuelo.

## 6 Líneas futuras de actuación

Como se ha dicho en el análisis inicial el alcance real del proyecto completo es muy superior a los resultados y la implementación obtenidos en la demostración.

Por ello durante la fase de requerimientos y diseño no se han establecido únicamente las bases para el código demostración sino que se han analizado los requerimientos y necesidades futuras en una hipotética continuación del proyecto.

En primer lugar, es básico implementar una primera versión del módulo de navegación. Siendo el sistema que controla directamente el dron, su realización permitiría mostrar el funcionamiento del sistema mediante pruebas reales. Para ello se necesitaría tanto un dron como el hardware dedicado a la navegación. Por otro lado, durante el proceso de desarrollo del módulo se puede utilizar un emulador de la placa. Es el trabajo más costoso de todo el proyecto y donde con diferencia se requieren unos conocimientos técnicos muy específicos.

Otra actuación a tener en cuenta es la ampliación de las funcionalidades del producto tales como un mayor uso de los recursos y posibilidades que ofrece Google Maps, incrementar las diferentes opciones de Triggers y acciones o mejorar la interfaz del usuario.

A partir de la tecnología Reflection existen diferentes posibilidades realmente potentes. La más destacada consiste en crear acciones, Triggers y filtros configurables vía web. Para ello, el sistema automáticamente puede analizar las funciones de configuración utilizando algún nombre identificativo en la función. Cuando el usuario selecciona el trigger o acción que desea utilizar se le añadirá un formulario con las opciones a rellenar. Un ejemplo claro sería el tiempo de refresco del Trigger de ejecución temporal. Utilizando esta técnica se mantienen todas las ventajas de los sistemas anteriores y añade una gran capacidad de configuración de los elementos.

El proceso de pruebas ha de ser extenso teniendo en cuenta tanto los múltiples dispositivos móviles de gamas diversas como diferentes drones. Hay que probar el sistema en diferentes entornos y situaciones adversas para

### AGRADECIMIENTOS

Mis sinceros agradecimientos a Felipe Lumbreras por su dedicación y esfuerzo en ayudarme, aconsejarme y darme soporte a lo largo del proceso además del tiempo extra dedicado debido a la realización de reuniones semanales.

### BIBLIOGRAFIA

- [1] Bitelli, G., Tini, M. A., Vittuari, L., 2003. Low-height aerial photogrammetry for archaeological orthoimaging production. ISPRS Archives Vol. XXXIV, Part 5/W12, pp. 55-59
- [2] Eugster, H., Nebiker, S., 2008. UAV-based augmented monitoring: real-time georeferencing and integration of video imagery with virtual globes. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 37 (Part B1), 1229-1236.
- [3] Empresa de venta de drones además de software y hardware de navegación automatizado (Arduylot). <<http://3drobotics.com>> [05 marzo 2015].
- [4] Guía del código base de Ardupilot. <<http://dev.ardupilot.com/wiki/learning-the-ardupilot-codebase>> [12 marzo 2015].
- [5] Guía básica de instalación y configuración de Ardupilot. <<http://www.aeromodelismovirtual.com/showthread.php?t=2361>> [05 marzo 2015].
- [6] Tutorial y código de MisionPlanner. <<http://planner.ardupilot.com/>> [15 marzo 2015].
- [7] Conferencia sobre el desarrollo de quadcopters con alta capacidad de vuelo, acrobática y cooperación. <[http://www.ted.com/talks/raffaello\\_d\\_andrea\\_the\\_astounding\\_athletic\\_power\\_of\\_quadcopters](http://www.ted.com/talks/raffaello_d_andrea_the_astounding_athletic_power_of_quadcopters)> [01 marzo 2015].
- [8] Código DroidPlanner. Gestor y control de vuelo mediante Smartphone. <<https://github.com/DroidPlanner/droidplanner>> [12 marzo 2015].
- [9] ISPRS Journal of Photogrammetry and Remote Sensing 79 (2013) 1-18. Feasibility of employing a smartphone as the payload in a photogrammetric UAV System
- [10] Guía para desarrolladores de Android. <<https://developer.android.com/index.html>> [17 mayo 2015].
- [11] Integración de google Maps mediante la API javascript. <<https://developers.google.com/maps/documentation/javascript/?hl=es>> [09 mayo 2015].
- [12] Donald E. Brown (2008). Struts 2. Madrid: Anaya Multimedia.
- [13] Guía de desarrollo básica de Strut 2. <<http://struts.apache.org/docs/guides.html>> [15 marzo 2015].
- [14] Javier Molder (2014). JAVA 8. Madrid: Anaya Multimedia.
- [15] Guía de desarrollo del protocolo XML. <<http://www.w3schools.com/xml/>> [02 mayo 2015].
- [16] Erich et al. Gamma (2002). Design Patterns. Massachusetts : ADDISON WESLEY
- [17] Guía de desarrollo con sensores en dispositivos Android. <[http://developer.android.com/guide/topics/sensors/sensor\\_s\\_overview.html](http://developer.android.com/guide/topics/sensors/sensor_s_overview.html)> [28 mayo 2015].
- [18] Tutorial de uso del patrón Observer. <<http://www.oodesign.com/observer-pattern.html>> [19 marzo 2015].
- [19] Tutorial de uso del patrón Interceptor. <<https://docs.oracle.com/javase/tutorial/reflect/>> [19 marzo 2015].
- [20] Documentación tecnología Reflection. <<https://docs.oracle.com/javase/tutorial/reflect/>> [03 mayo 2015].
- [21] Guía de desarrollo de html. <<http://www.w3schools.com/html/>> [26 mayo 2015].
- [22] Guía de desarrollo de css, <<http://www.w3schools.com/css/>> [06 junio 2015].
- [23] Guía de desarrollo de javascript, <<http://www.w3schools.com/js/>> [05 junio 2015].
- [24] Guía de desarrollo de jquery. <<http://www.w3schools.com/jquery/>> [12 mayo 2015].
- [25] Guía de desarrollo del protocolo JSON. <<http://www.w3schools.com/json/>> [07 mayo 2015].
- [26] Plugin para utilizar JSON en Struts 2 para comunicar cliente con servidor. <<https://struts.apache.org/docs/json-plugin.html>> [12 mayo 2015].

## APÉNDICE

### A1. INTERFAZ DE PREPARACIÓN DEL VUELO

En el proceso de preparación del vuelo el usuario debe iniciar la conexión entre los dos módulos. Para ello se utiliza el panel superior. Siguiendo el orden de izquierda a derecha, en primer término se ha añadido un aviso de conexión. Si el botón esta en verde hay una conexión establecida mientras que rojo indica una pérdida de conexión entre los dos sistemas. Por consiguiente, se puede iniciar la conexión, importar toda la estructura del módulo de control y cerrar la conexión.

En el recuadro del medio se ha creado un log y un pequeño indicador de la distancia que recorrería el dron en la ruta actual marcada.

Por otro lado, en el recuadro inferior se establece toda la configuración de los Triggers y acciones. A la izquierda se implementan todos los elementos que no corresponden con la localización como el detector de Bluetooth o el movimiento brusco. La relación entre ellos se marca mediante una tabla dinámica. A mano derecha únicamente se crea la acción correspondiente a uno de los puntos establecidos. Se puede elegir el punto apretando encima de la etiqueta correspondiente, una vez configurada se cambia el color a verde.

Por último, para crear una ruta únicamente se tiene que utilizar el servicio de Maps implementado. Se aprieta encima del mapa y una línea va uniendo los sucesivos puntos

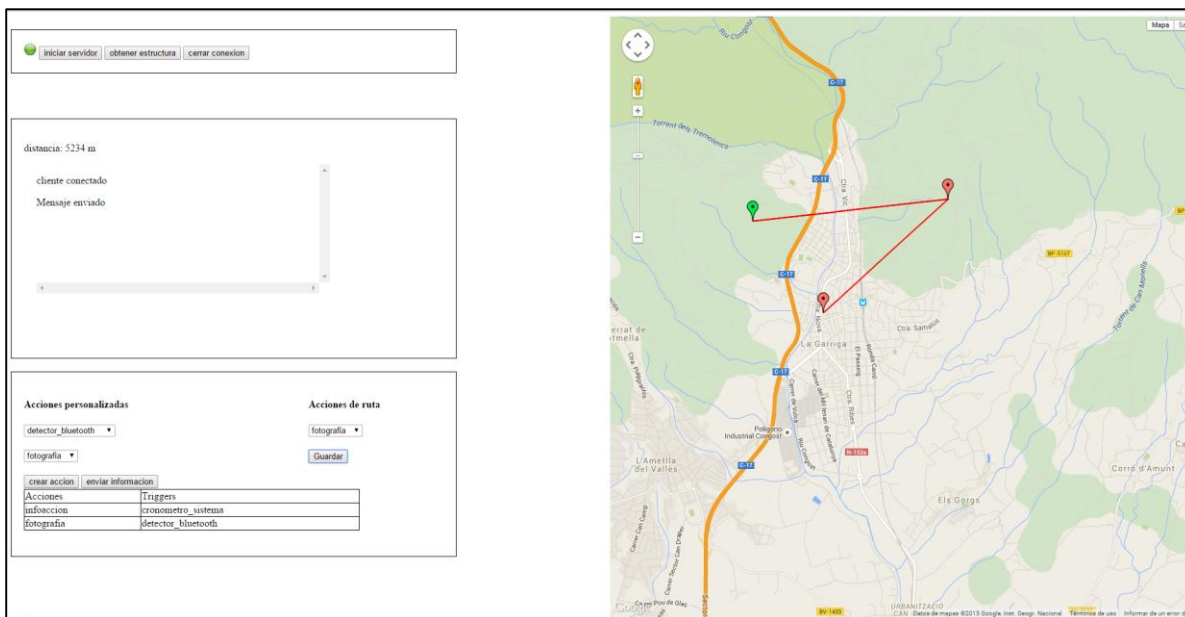


Figura A1.1: Captura realizada durante una prueba del sistema completo. Corresponde a la interfaz gráfica del sistema de preparación anterior al vuelo

## A2. INTERFAZ DE CONTROL DEL VUELO

Durante el proceso de vuelo la interfaz gráfica se convierte en un sistema de control. Para la versión de demostración el usuario únicamente puede visualizar un conjunto de información dado por el dispositivo móvil.

En primer término, se tiene un registro donde el sistema puede escribir diferente información relativa al estado en el envío de imágenes, eventos o cualquier otra información que no se muestre en algún lugar de la interfaz.

Por consiguiente, se tiene un conjunto de datos de localización tales como la posición geográfica del último registro del GPS y la velocidad además de la inclinación del dispositivo a partir de tres ejes.

Por otro lado, se tiene un conjunto de información sobre el estado de los sensores del drone tanto a nivel de conexión, como batería.

Por último, un pequeño sistema de visualización de imágenes nos permite gestionar las diferentes capturas enviadas a lo largo de vuelo.

A nivel del servicio de Google Maps , en este caso se puede ver la ruta marcada , la posición última del drone (indicado por el icono azul) y una marca de un Trigger activado al que se le haya incorporado la acción pertinente (iconos rojos).

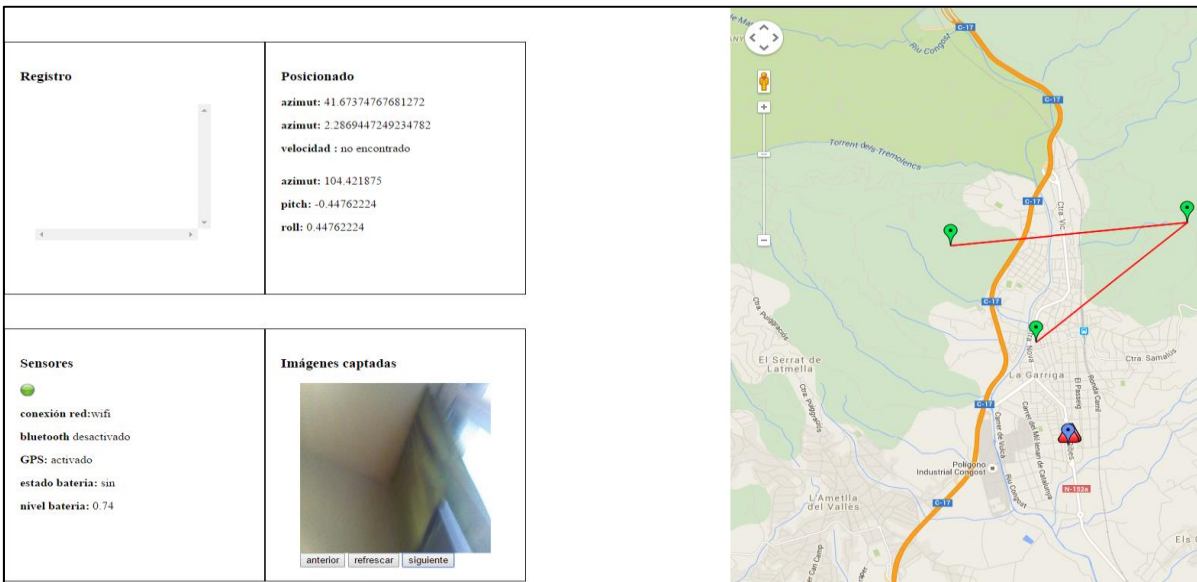


Figura A2.1: Captura realizada durante una prueba del sistema completo. Corresponde a la interfaz gráfica del sistema de control de vuelo