

SD	Sistemas Distribuidos
23/24	Práctica no guiada: Sockets, Streaming de Eventos, Colas y modularidad.
	Art With Drones

Preámbulo

El objetivo de esta práctica es que los estudiantes extiendan y afiancen sus conocimientos sobre el uso de la tecnología de comunicación básica sockets, estudiada durante las sesiones de teoría.

Los sockets son la base de las comunicaciones entre computadores y suponen el punto de enlace básico entre nuestra aplicación e Internet, utilizando muy pocos recursos de red. Como contrapartida, las aplicaciones que utilizan sockets como mecanismo de comunicación deben interpretar los mensajes que intercambian entre sí, es decir, deben establecer un protocolo o acuerdo de comunicación entre ellos. Esta necesidad implica un fuerte acoplamiento entre las aplicaciones distribuidas que utilizan sockets como mecanismo de comunicación, ya que todas las partes deben conocer de antemano la estructura de los mensajes que serán intercambiados y codificarlo en su programación.

Por otro lado, otras tecnologías y arquitecturas actuales orientadas al streaming de eventos en tiempo real, basadas en sockets pero de más alto nivel en términos de construcción y desarrollo, permiten la implementación de soluciones menos acopladas y más resilientes que estos.

Esta práctica establece el marco inicial de un desarrollo que se ampliará durante el cuatrimestre y que permitirá al estudiante crear una aplicación similar a las que se desarrollan hoy en día en los entornos empresariales, poniendo el foco en el uso e integración de distintos paradigmas de comunicación susceptibles de ser utilizados.

Especificación

El objetivo de la práctica a desarrollar es un sistema distribuido que implemente una simulación de una solución para la creación de figuras mediante dispositivos autónomos (drones) manejados en tiempo real.

Descripción funcional

Los estudiantes deberán implementar una solución de control de dispositivos online en tiempo real denominada "ART WITH DRONES".

Alcance

La solución tendrá como objetivo simular un controlador de drones que formen imágenes en el espacio.



Figura 1: Espectáculo Harry Potter de figuras formadas con drones en Universal Orlando

En nuestro caso, el sistema, partiendo de una entrada (fichero, Json, BD) que expresa una figura deberá ser capaz de gobernar una serie de dispositivos (simulando drones) los cuales representarán esa figura en un espacio representado en 2D.

Para ello se basará en una serie de componentes distribuidos que estarán interconectados y que conformarán la solución de manera resiliente, escalable y segura.

Espacio / Mapa

El espacio aéreo será representado por una simple matriz 2D de 20x20 posiciones. Cada posición del mapa (elemento de la matriz) puede contener una de las siguientes posibilidades:

- 1- Un dron: Representado por su identificador (número) y un color (rojo si está moviéndose hacia su posición final, verde si ya ha llegado a la misma)
- 2- Nada: Representado por un “espacio”.

Todas las aplicaciones que representan los drones así como el motor central que los gobierna deberán estar recibiendo y visualizando en todo momento el espacio completo en su estado actual.

El mapa representa una geometría esférica de manera que las posiciones más al este (límite derecha del mapa) están conectadas con las situadas al oeste y viceversa, de la misma forma que las situadas en el límite norte (zona superior del mapa) conectan con el lado sur y viceversa.

Adicionalmente en el espacio se registra un clima el cual puede alterar el comportamiento del sistema según se comenta en los siguientes apartados.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1																				
2																				
3																				
4																				
5																				
6																				
7																				
8																				
9																				
10																				
11																				
12																				
13																				
14																				
15																				
16																				
17																				
18																				
19																				
20																				

Figura 1: Espacio inicial sin ningún dron. Espectáculo no iniciado

Drones

Los drones estarán representados por una aplicación distribuida autónoma.

Disponen de los siguientes atributos:

- Posición: Coordenadas del mapa en las que se encuentra: { int [1..20], int[1..20] }
- ID: Identificador del mismo. Numérico [0..99]-Se usará para representarlo en el mapa.
- Estado: Representado por un color. En movimiento = Rojo, En posición final= Verde.

Cada dron se moverá por el espacio retransmitiendo su posición cada vez que cambie de coordenada. Como si se tratará de un espacio real, solo se puede mover a su siguiente coordenada adyacente en cualquier dirección.

Mecánica de la solución

El sistema comienza mediante la funcionalidad “START” que el estudiante deberá implementar en la aplicación del servidor central (en adelante “Engine”).

El sistema realizará entonces la siguiente secuencia:

- 1- El Engine entrará en modo espera hasta que, en un fichero dispuesto a tal propósito, se encuentre información para la ejecución de una figura.

El formato de fichero será:

```
<FIGURA>
<ID_DRON><COORDENADA_X_DESTINO><COORDENADA_Y_DESTINO>
<ID_DRON><COORDENADA_X_DESTINO><COORDENADA_Y_DESTINO>
...
</FIGURA>
```

- 2- Los drones, por su lado, previo a su inclusión en la acción, deberán haberse registrado en un módulo del sistema desarrollado para tal propósito.
- 3- Una vez realizado el registro, los drones solicitarán al Engine su inclusión en la acción y, una vez autenticados y aceptados en el Engine que comprobará dicha identificación, quedarán a la espera de recibir instrucciones.
- 4- El Engine irá publicando los destinos a los que los drones deben ir moviéndose según la información del fichero comentado en el punto 1.
- 5- Los drones, según reciben sus instrucciones, se irán moviendo progresivamente a su dirección de destino **notificando cada movimiento que realizan**. Este estado del dron equivaldrá al modo “RUN” del dron y se representará en pantalla mediante la expresión en color ROJO de ese dron.

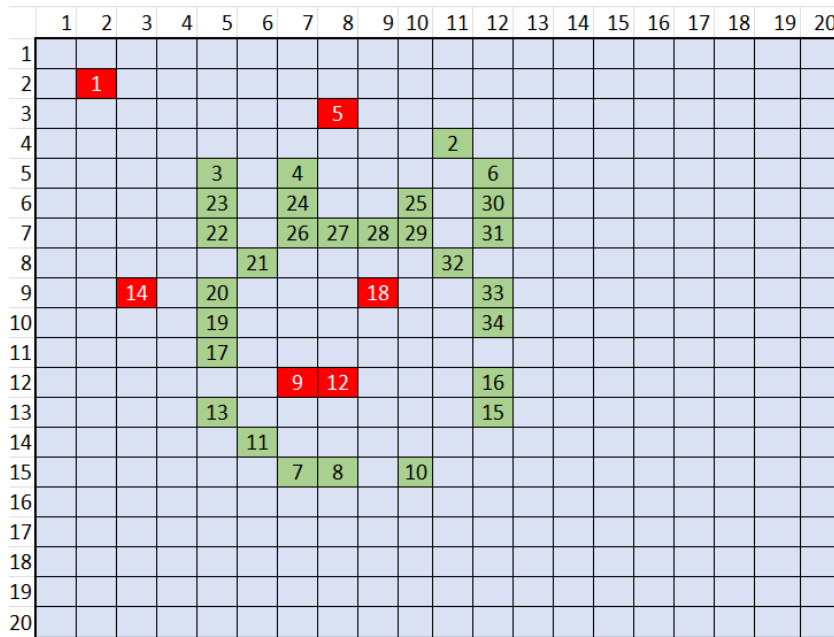


Figura 2: Espectáculo en curso. Drones posicionándose

- 6- El engine recibirá dichos movimientos y volverá a emitir el estado de situación de todo el mapa que todos los drones recibirán a su vez.
- 7- Cuando un dron llegue a su destino deberá permanecer en él cambiando su estado a “END” que se reflejará en pantalla mediante la expresión en color VERDE de ese dron.
- 8- Cuando todos los drones lleguen a su destino el Engine lo detectará y notificará mediante un aviso en pantalla : “FIGURA COMPLETADA”.

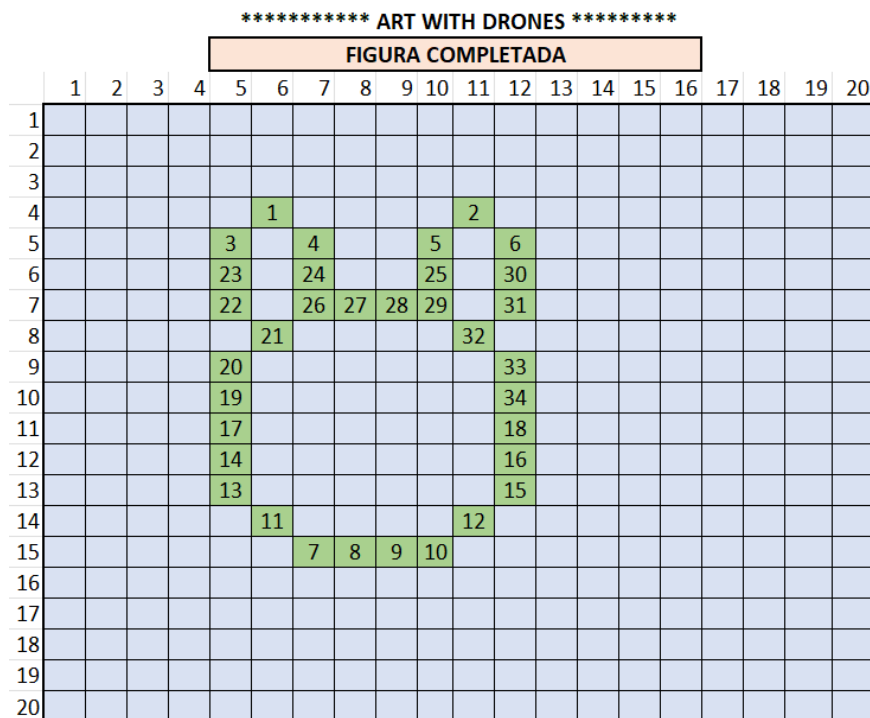


Figura 3: Figura completada. Drones en espera de nuevas instrucciones

- 9- Si hubiera más figuras a realizar en el fichero, el Engine generará una pausa de 5 segundos y procederá a enviar a los drones la siguiente información de destino empezando de nuevo la secuencia de movimientos.
- 10- Si durante 10 segundos no llegan más figuras al fichero, el Engine enviará órdenes a los drones para que vuelvan a su base.

El Engine puede pararse mediante un opción denominada “STOP” ante la cual debe notificar a todos los drones que vuelvan a su base y posteriormente “apagar” todos los mecanismos. Mientras esto no ocurra, el Engine estará siempre a la espera de nuevas entradas en el fichero de figuras.

La acción se va desarrollando de forma asíncrona, es decir, sin “turnos”, de manera que cada dron decide moverse en cualquier momento.

Los drones pueden moverse en cualquier dirección: N,S,W,E,NW,NE,SW,SE

Todos los drones entran y salen de la acción por las coordenadas 1,1 y se mueven hacia su destino con un movimiento de 1 coordenada adyacente cada vez.

En cada movimiento se recalcula la posición del dron. Si un dron falla o su aplicación se bloquea por cualquier causa es eliminado visualmente de la acción. Si se recupera debe volver a autenticarse y volver a moverse a su destino según le indique el Engine.

Finalmente **el clima** puede afectar notablemente al desarrollo del espectáculo. El Engine deberá consultar tanto al inicio de su ejecución como con cierta periodicidad el clima que está teniendo lugar. **Si la temperatura está por debajo de 0º el Engine deberá clausurar el espectáculo.** En caso de que todavía no se hubiera iniciado deberá impedir su inicio y si ya estuviera en curso deberá notificarlo a todos los drones los cuales volverán a su base. En ambos casos, en todas las pantallas se mostrará el texto “CONDICIONES CLIMATICAS ADVERSAS. ESPECTACULO FINALIZADO”.

A continuación, se especifica más detalladamente cada componente.

Núcleo

Contiene todos los módulos que implementan la estructura principal del sistema.

Engine:

Se trata de la aplicación que implementará la lógica fundamental del sistema. El nombre de la aplicación será **obligatoriamente “AD_Engine”**. Para su ejecución recibirá por la línea de parámetros los siguientes argumentos:

- Puerto de escucha
- Número máximo de drones a admitir.
- IP y puerto del Broker/Bootstrap-server del gestor de colas.
- IP y Puerto AD_Weather
- IP y puerto de la BBDD (solo en caso de que sea necesario)

Al arrancar la aplicación quedará a la escucha en el puerto establecido e implementará todas las funcionalidades necesarias para ejecutar la mecánica de la solución expresada en el apartado anterior.

Cada cierto tiempo (X segundos) deberá hacer consultas al servidor de clima para ver el estado del tiempo y aplicar las medidas necesarias.

Para mayor sencillez de implementación, ante cada movimiento de cada dron le responderá con todo el tablero.

Los estudiantes podrán decidir la forma de expresar el mapa aunque se recomienda un array de bytes donde cada elemento de la matriz de bytes es una posición en del mapa.

Registry:

Se trata de la aplicación que implementará el registro de los drones en el Núcleo. El nombre de la aplicación será **obligatoriamente “AD_Registry”**. Para su ejecución recibirá por la línea de parámetros los siguientes argumentos:

- Puerto de escucha
- IP y puerto de la BBDD (solo en caso de que sea necesario)

La aplicación permanecerá a la espera hasta recibir una solicitud de registro de un dron. Los datos que puede guardar¹ de un dron son:

- ID
- Alias

Ante un registro de un dron el AD_Registry devolverá un **token de acceso** con el que el dron se podrá autenticar en el Engine.

¹ En la segunda parte de la práctica se implementarán las medidas de seguridad oportunas añadiendo tanto los datos adecuados en el registro, como la correcta gestión de los tokens y otros aspectos.

Base de Datos:

Contendrá, los datos de los drones, sus credenciales de acceso y cualquier otra información que los estudiantes consideren necesaria para la implementación del sistema.

Servirá de recurso compartido entre los distintos elementos que forman el Núcleo.

Los estudiantes podrán decidir el motor de BD a implementar recomendando los profesores los siguientes: SQLite, MySQL, SQLServer o MongoDB. Igualmente será posible usar simples ficheros para la implementación.

Dron

Aplicación que implementa la funcionalidad del dron. El nombre de la aplicación será **obligatoriamente "AD_Drone"**. Para su ejecución recibirá por la línea de parámetros los siguientes argumentos:

- IP y puerto del AD_Engine
- IP y puerto del Broker/Bootstrap-server del gestor de colas
- IP y puerto del AD_Registry

La aplicación tendrá las siguientes opciones:

- Registrar Dron: Se conectará al módulo AA_Registry para darse de alta en el sistema, editar su perfil o darse de baja.
- Unirse al espectáculo: Usando el token del dron² que se ha recibido durante el registro, este será autenticado en el Engine (comunicación por sockets) estando, de esta forma, en disposición de recibir instrucciones desde el Engine con las acciones de movimiento (coordenadas a las que debe dirigirse) que debe ejecutar (comunicación por streaming). El mapa deberá ir mostrándose en pantalla conforme el dron se mueve **con todos los detalles que muestren el estado del espacio**.
- **Nota:** El estudiante deberá resolver cómo saber si un dron sigue correcto funcionamiento.

Servidor de clima

Aplicación que implementa la funcionalidad. El nombre de la aplicación será **obligatoriamente "AD_Weather"**. Para su ejecución recibirá por la línea de parámetros los siguientes argumentos:

- Puerto de escucha

² En la segunda parte de la práctica se implementará los sistemas de contraseña y seguridad en general.

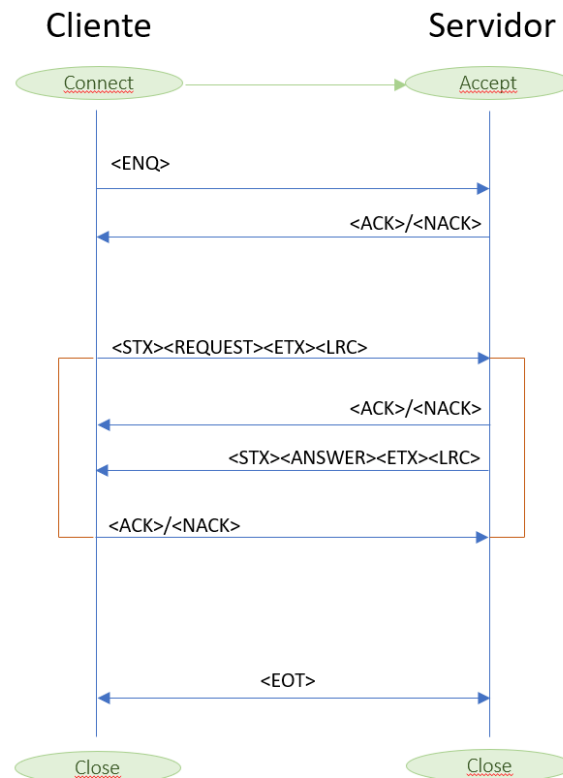
Esta aplicación permanecerá a la escucha indefinidamente esperando a que la aplicación AD_Engine le solicite el estado del clima en la ciudad donde se encuentra ejecutando la acción. La aplicación devolverá un valor de la temperatura que podrá determinarse a voluntad.

La aplicación dispondrá de una BD (un fichero si así se desea) con ciudades del mundo y una temperatura indicada.

Nota: En la segunda parte de la asignatura este servidor de clima será sustituido por una consulta a un servidor real mediante las tecnologías que se indicarán en su momento.

Protocolo de comunicación (solo sockets)

Como se comenta al principio de este documento, la comunicación mediante sockets obliga a los actores involucrados a definir la mensajería y protocolo con precisión. Los estudiantes podrán definir los protocolos de comunicación entre los distintos módulos del sistema a su criterio. **Los profesores recomiendan, aunque no es obligatorio**, usar una mensajería basada en el estándar de empaquetado <STX><DATA><ETX><LRC> que corresponde al siguiente flujo:



Donde:

- <REQUEST> y <ANSWER>: Contienen el mensaje transmitido entre ambos puntos con los campos separados por un carácter determinado:
 - o Ej.: <REQUEST>: Código Operación#campo1#...#campo n
- <LRC>: Se define como el XOR(MESSAGE) byte a byte y sirve para validar que la transmisión del mensaje se ha realizado satisfactoriamente y ha sido recibida de forma correcta por el destinatario el cual responderá al mismo con un <ACK> o <NACK>

Aclaraciones finales

Antes incongruencias funcionales o aspectos no considerados en los anteriores apartados que impidan la correcta implementación de la práctica, los estudiantes deberán implementar la alternativa que entiendan más conveniente previa consulta con el profesor para consensuar dicha implementación evitando un excesivo nivel de complejidad.

Guía mínima de despliegue

Para la correcta evaluación de la práctica es necesario comprobar que la aplicación distribuida solicitada es desplegada en un entorno verdaderamente distribuido. Es por ello por lo que para su prueba es necesario al menos 3 PCs distintos en los que se desplegarán los componentes solicitados. Al menos se ha de desplegar junto con el núcleo y el servidor de clima, 2 drones, proporcionando el siguiente escenario:

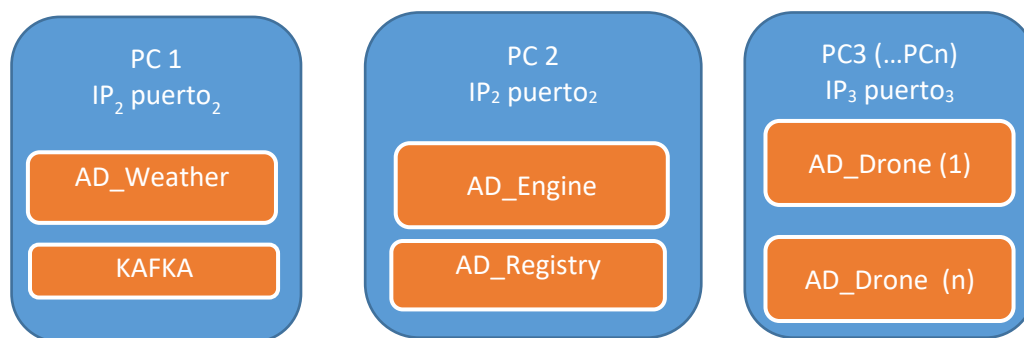


Figura 4. Escenario físico mínimo para el despliegue de la práctica.

Entregables y evaluación

La evaluación de la práctica se realizará en los laboratorios. **Se podrá realizar en grupos de hasta 2 personas sin perjuicio de que, durante el momento de la corrección, el profesor pueda preguntar a cualquier estudiante del grupo por cualquier aspecto de cualquiera de los módulos.** Los estudiantes deben desplegar por ellos mismos la práctica que resuelve el enunciado anterior. Deben desplegar un sistema completo, es decir, un núcleo, los drones y el servidor de clima todos ellos interconectados entre sí. **Este requisito es indispensable para poder realizar la corrección.** Además, deben poderse evaluar positiva o negativamente todos los apartados que aparecerán en la Guía de corrección que se entregará a tal propósito. Cada uno de los apartados puntúa de forma variable, por tanto, cada apartado no implementado o que no pueda comprobarse su correcto funcionamiento no podrá ser tenido en cuenta y por tanto no puntuará. Los estudiantes deberán presentar para la evaluación el documento “Guía de corrección” cumplimentado para que el profesor pueda validar los apartados implementados.

Los estudiantes deberán entregar, además, mediante la funcionalidad de evaluación del UACloud antes de la fecha establecida a su profesor de prácticas una memoria de prácticas, con el código fuente y compilados generados, así como un documento donde se detalle la siguiente información. El formato es libre, pero debe ser un documento ordenado y debidamente formateado, cuidando la redacción y ortografía.

- Portada con los nombres, apellidos y DNI de los estudiantes, año académico y el título de la práctica.
- Un informe donde se indique el nombre de los componentes software desarrollados y una descripción de cada uno de ellos, explicando y enviando además el código fuente de todos ellos.

- El detalle, paso a paso, de una guía de despliegue de la aplicación, que deberá ser la misma que utilice cuando haga la corrección de la práctica.
- Capturas de pantalla que muestren el funcionamiento de las distintas aplicaciones conectadas.

Cada profesor de prácticas podrá solicitar a los estudiantes cualquier otra evidencia que el profesor considere adecuada para poder formalizar la evaluación.

La fecha de entrega será en la semana del 30/10/2023.