

1. Descripción de la aplicación, funcionalidad y actores que intervienen

La práctica ha consistido en el desarrollo de una aplicación Cliente que, mediante una interfaz gráfica pudiese solicitar el tiempo actual y el pronóstico para los próximos días de todas las capitales de provincia españolas. Dichos datos los obtiene realizando peticiones a una aplicación servidor, que se encarga de obtener y enviar los datos meteorológicos al cliente.

Para que los datos fuesen reales y pudiesen estar actualizados, mi servidor a su vez actúa como cliente de un servidor público de datos meteorológicos a través de internet: OpenWeatherMap.

Tanto servidor como cliente están diseñados para escuchar en el puerto **8989**. La aplicación cliente se conecta con la dirección «**localhost**» por defecto, por lo que en principio la aplicación se debe probar ejecutando ambos en la misma máquina. El paso de mensajes entre los procesos se realiza mediante el uso de sockets TCP, lo cual garantiza la seguridad e integridad de la comunicación entre ambos procesos.



1.1 Conexión Cliente/Servidor

Nada más arrancar, el cliente busca al servidor enviando una petición especial “HELLO”. Si el servidor está disponible y establece la conexión correspondiente, responde a la petición con “OK”. Cuando la conexión se ha realizado correctamente entre ambos procesos el cliente carga la interfaz de usuario.

La interfaz del cliente se basa en un mapa interactivo en el que aparecen representadas todas las capitales de provincia españolas de las cuales se puede obtener el tiempo atmosférico actual así como la previsión para los tres próximos días. Cabe destacar que se ha incluido una línea en la parte inferior de la interfaz en donde se detalla la actividad cliente-servidor. El servidor por su parte también comunica todas las comunicaciones con el cliente a través de mensajes en consola.

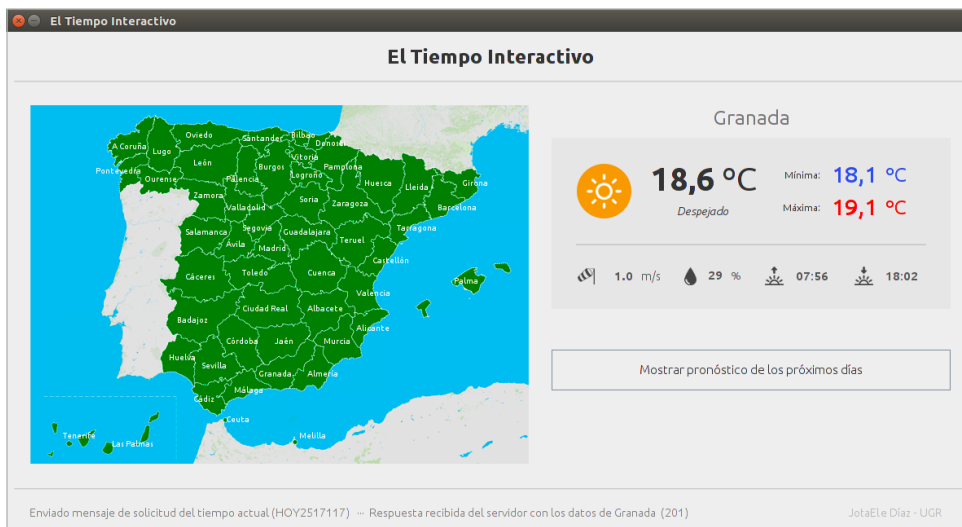


La interfaz de usuario del cliente carga una vez se ha conectado correctamente al servidor. A la derecha podemos ver como el servidor ha recibido la petición y enviado la respuesta «OK».

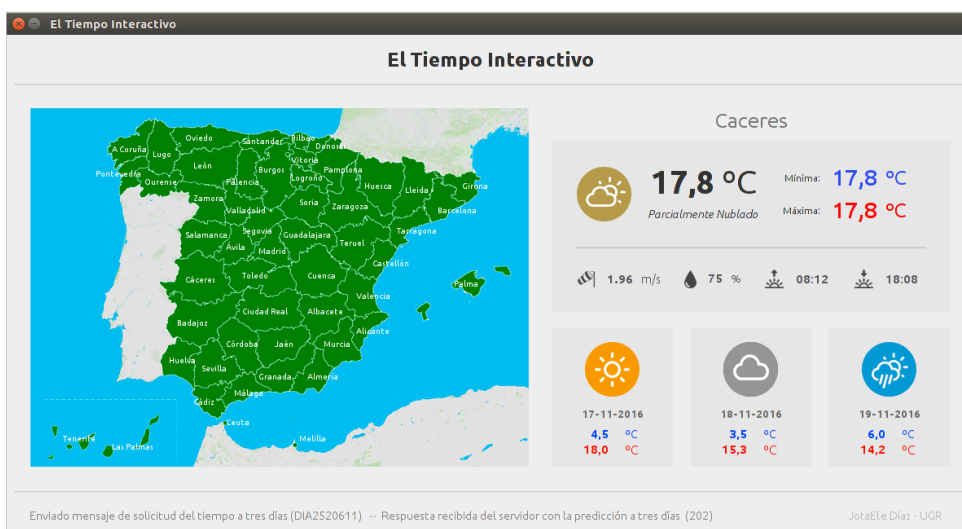
1.2 Descripción de la funcionalidad de la aplicación

El usuario puede obtener el tiempo atmosférico actual de cualquier ciudad haciendo click sobre ella en el mapa. Una vez se ha hecho la selección, el proceso realiza una petición concreta al servidor identificada como “HOY” seguido de un código interno que identifica a la ciudad. Así, por ejemplo, para solicitar los datos de Granada el cliente envía la petición «HOY2517117» al servidor.

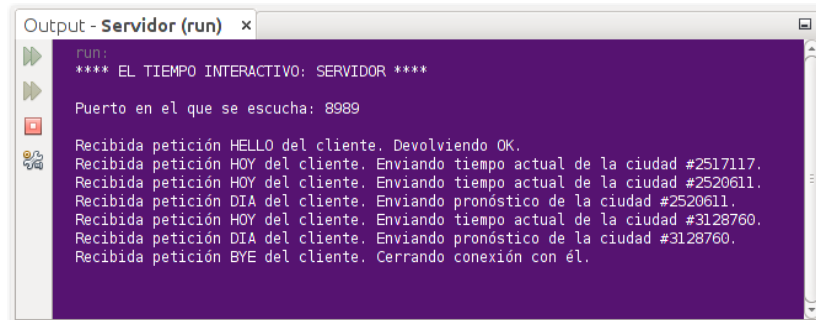
El servidor, una vez ha obtenido de la API de OpenWeatherMap los datos deseados, responde a la petición del cliente con una cadena de texto en formato JSON que contiene los datos solicitados. El cliente se encarga de extraer dichos datos y los muestra en la interfaz. El cliente muestra un icono significativo al estado del tiempo, la temperatura actual, la máxima y mínima, así como otros datos complementarios: La velocidad del viento, humedad y la hora a la que saldrá y se pondrá el Sol.



Una vez obtenidos los datos del tiempo en una ciudad, la interfaz dispone de un botón que permite solicitar el pronóstico de los próximos tres días. Cuando se pulsa, el cliente envía una nueva petición «DIA» al servidor seguida del código de la misma ciudad para la que se había hecho la petición anterior. El servidor obtiene una vez más los datos del pronóstico a tres días y los envía al cliente en formato JSON.



Mediante el cliente, el usuario puede realizar tantas peticiones «HOY» como «DIA» que desee de cualquiera de las ciudades representadas. En cualquier momento se puede finalizar el proceso cerrando la ventana de la interfaz. Se ha programado un disparador que hace que el cliente envíe una señal “BYE” al servidor. El servidor recibe dicha petición y se encarga de cerrar el socket con él, evitando así posibles problemas por datos en el buffer de entrada o salida que hubiesen quedado a medio camino.



```
run:
**** EL TIEMPO INTERACTIVO: SERVIDOR ****

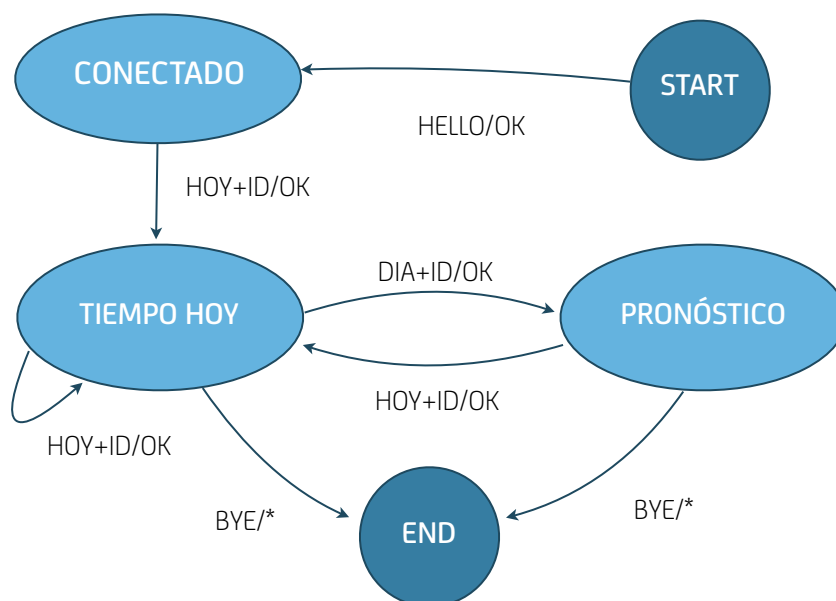
Puerto en el que se escucha: 8989

Recibida petición HELLO del cliente. Devolviendo OK.
Recibida petición HOY del cliente. Enviando tiempo actual de la ciudad #2517117.
Recibida petición HOY del cliente. Enviando tiempo actual de la ciudad #2520611.
Recibida petición DIA del cliente. Enviando pronóstico de la ciudad #2520611.
Recibida petición HOY del cliente. Enviando tiempo actual de la ciudad #3128760.
Recibida petición DIA del cliente. Enviando pronóstico de la ciudad #3128760.
Recibida petición BYE del cliente. Cerrando conexión con él.
```

Registro de peticiones y respuestas que ha registrado el servidor. Podemos comprobar como al desconectarse el cliente ha enviado una última petición «BYE» para finalizar la conexión.

2. Diagrama de Estados del Servidor

A continuación se muestra el grafo de estados del servidor, en el que se detallan en las transiciones de dichos estados las diferentes peticiones y respuestas que recibe del cliente. La sintaxis de dichas transiciones es Petición/Respuesta.



3. Mensajes que intervienen

A continuación se enumeran los distintos mensajes que intervienen entre los procesos cliente y servidor en la aplicación desarrollada. Se definen tanto las peticiones del cliente como las diferentes respuesta que ofrece el servidor.

3.1 Mensajes del Cliente

Código	Cuerpo	Descripción
1000	HELLO	Mensaje que envía el cliente para comprobar la disponibilidad del servidor.
1001	HOY + ID_CIUADAD	Petición al servidor del tiempo metereológico actual para la ciudad identificada por el ID_CIUADAD que se envía.
1002	DIA + ID_CIUADAD	Petición al servidor de la previsión para los próximos tres días del tiempo para la ciudad identificada por el ID_CIUADAD que se envía.
1003	BYE	Señal que se envía al servidor para informar de que se va a finalizar la conexión con él. Tras enviarlo la aplicación se cierra.

3.1 Mensajes del Servidor

Código	Cuerpo	Descripción
200	OK	Mensaje de confirmación de conexión a la petición del cliente.
1002	OK + JSON HOY	Se envía una confirmación de respuesta seguido del JSON que contiene los datos del tiempo actual para la ciudad que ha solicitado el cliente.
1003	OK + JSON DIA	Se envía una confirmación de respuesta seguido del JSON que contiene los datos del pronóstico a tres días para la ciudad que ha solicitado el cliente.
400	*	Señal por defecto que se envía cuando se recibe una petición que no se reconoce.