



UNIVERSIDAD DE GRANADA

3ºA - SUBGRUPO A2

FUNDAMENTOS DE REDES

GRADO EN INGENIERÍA INFORMÁTICA

Ejercicio 5 - Definición e implementación de un protocolo de aplicación

Jose Luis Gallego Peña
Antonio David Villegas Yeguas

15 de noviembre de 2019

Índice

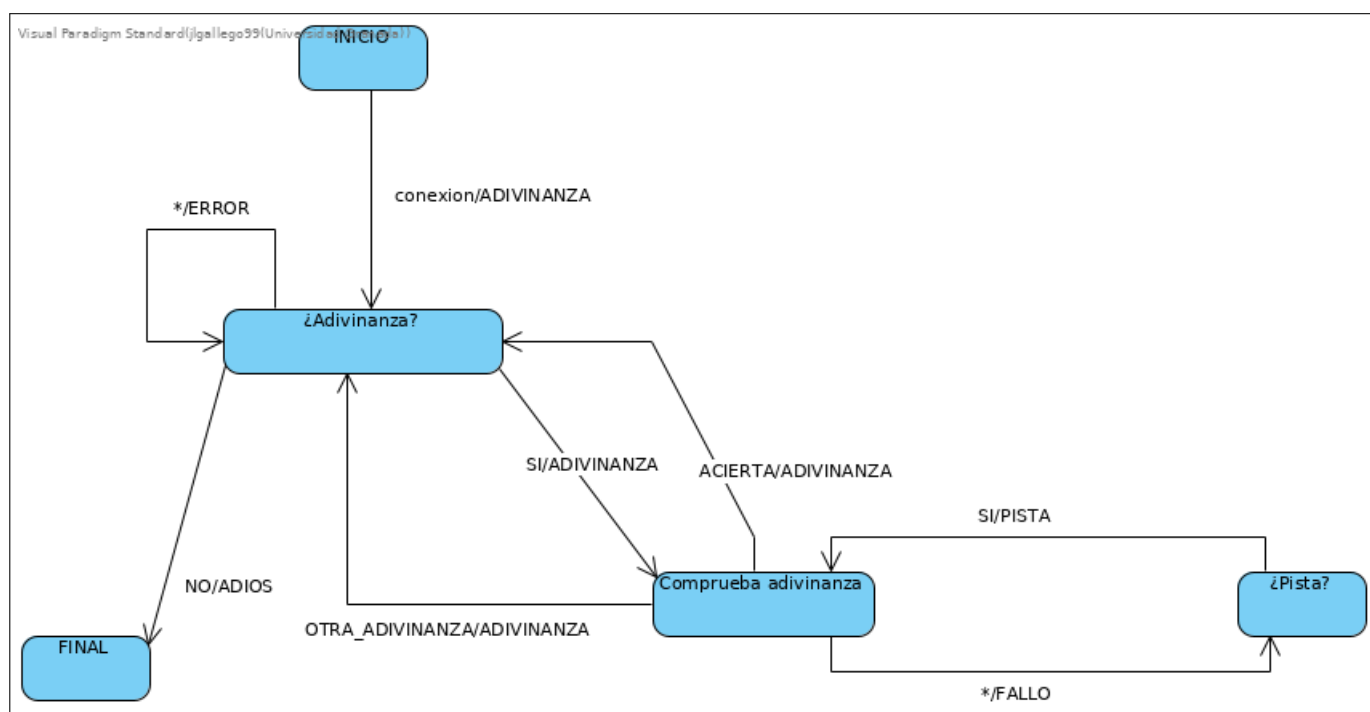
1. Introducción	2
2. Diagrama de estados del servidor	2
3. Mensajes que intervienen	2
4. Funcionamiento de la aplicación	3

1. Introducción

En este ejercicio hemos desarrollado una aplicación en Java que trabaja con envíos de mensajes por un socket TCP. Consiste en un **juego de adivinanzas** en el cual el servidor proporciona varias adivinanzas y los clientes (pueden conectarse varios) deben resolverlas, ganando así puntos, pudiendo jugar cuanto quieran y parar cuando lo deseen.

2. Diagrama de estados del servidor

Definimos el diagrama de estados por los que pasa el servidor de acuerdo a los eventos que puedan surgir o los mensajes que recibe por parte de el/los cliente/s. Las etiquetas siguen la forma: `<evento>/<envia>` siendo `<evento>` lo que sucede en el servidor (el cliente provoca ese evento, y `*` hace referencia a cualquier evento) y `<envia>` lo que el servidor envía al cliente para pasar al siguiente estado.



3. Mensajes que intervienen

Describimos los mensajes y su información asociada que intervienen en el protocolo durante la comunicación entre el cliente y el servidor. Los distintos apartados de la tabla quieren decir lo siguiente:

- **Código:** Identificador único para cada mensaje. Los mensajes del cliente tienen 3 dígitos mientras que los del servidor tienen 4. Los que empiezan por 1 corresponden a los mensajes de conexión, los que empiezan por 2 son mensajes que se intercambian el servidor y el cliente relacionados con el desarrollo del juego, y los que empiezan por 3 son los mensajes de error.
- **Cuerpo:** Funcionalidad en el caso del cliente y respuesta en el caso del servidor.
- **Descripción:** Breve descripción de en qué consiste el mensaje.

Mensajes del cliente

CÓDIGO	CUERPO	DESCRIPCIÓN
100	conexion	El cliente se conecta al servidor
200	NO	El cliente deniega la pregunta del servidor
201	SI	El cliente afirma la pregunta del servidor
202	OTRA_ADIVINANZA	El cliente solicita una nueva adivinanza al servidor
203	ACIERTA	El cliente acierta la adivinanza

Mensajes del servidor

CÓDIGO	CUERPO	DESCRIPCIÓN
3000	ERROR	Se ha producido un error, el cliente no ha introducido una opción correcta
2001	ADIOS	El servidor despide al cliente, indicando que se ha desconectado y cerrando la conexión
2002	OK	El servidor confirma la petición del cliente
2003	ADIVINANZA	El servidor envía al cliente una adivinanza aleatoria
2004	PISTA	El servidor envía al cliente una pista asociada a la adivinanza actual
2005	FALLO	El cliente no ha acertado la adivinanza

4. Funcionamiento de la aplicación

La aplicación se compila mediante un makefile, escribiendo la orden make estando en la carpeta raíz. Una vez hecho esto, se deben abrir dos terminales (mínimo, una para un cliente y otra para el servidor, pero se pueden abrir más para N clientes) y en cada uno de los terminales entrar en la carpeta bin y ejecutar el archivo correspondiente con la orden java:

```
java AdivinaServidorConcurrente para el servidor.
```

```
java AdivinaClienteTCP para el/los cliente/s.
```

Una vez esto el cliente contestará a los mensajes que le envíe el servidor, quedando el funcionamiento completo de la siguiente manera:

```
antonio@antonio: ~/Documentos/Universidad/3o/1er cuatrimestre/FR/practicas/p2/ej5_app/bin
FR/practicas/p2/ej5_app/bin on } master
➔ java AdivinaClienteTCP
¿quieres una adivinanza? (si/no)
si
Son de color chocolate, se ablandan con el calor y si se meten al horno explotan con gran furor. (escribe "otra" para cambiar de adivinanza)
otra
¿quieres una adivinanza? (si/no)
no
Chao pescao, has adivinado 0 adivinanzas.
FR/practicas/p2/ej5_app/bin on } master took 12s
➔
```

```
FR/practicas/p2/ej5_app on } master
➔ make
mkdir ./bin
mkdir: no se puede crear el directorio './bin': El fichero ya existe
make: [makefile:11: init] Error 1 (no tiene efecto)
javac -d ./bin/ ./src/*.java
FR/practicas/p2/ej5_app on } master
➔
```

```
java AdivinaServidorConcurrente
FR/practicas/p2/ej5_app/bin on } master
➔ java AdivinaServidorConcurrente
Recibida conexión de un cliente, mandamos si quiere adivinanza
Recibida respuesta: si
Mandamos una nueva adivinanza
Recibida respuesta de adivinanza: otra
Recibido que quiere otra adivinanza, mandamos si quiere nueva adivinanza
Recibido que no quiere más adivinanza, mandamos despedida y cerramos la conexión
➔
```

```
antonio@antonio: ~/Documentos/Universidad/3o/1er cuatrimestre/FR/practicas/p2/ej5_app/bin
FR/practicas/p2/ej5_app/bin on } master
➔ java AdivinaClienteTCP
¿quieres una adivinanza? (si/no)
si
Son de color chocolate, se ablandan con el calor y si se meten al horno explotan con gran furor. (escribe "otra" para cambiar de adivinanza)
otra
¿quieres una adivinanza? (si/no)
no
Chao pescao, has adivinado 0 adivinanzas.
FR/practicas/p2/ej5_app/bin on } master took 12s
➔ java AdivinaClienteTCP
¿quieres una adivinanza? (si/no)
si
Iba una vaca de lado, luego resultó pescado. (escribe "otra" para cambiar de adivinanza)
vaca
Te has equivocado, ¿quieres otra pista? (si/no)
si
Mucha gente lo suele escribir mal al acabarlo en "--ado"
bacalao
Has acertado, ¿quieres una adivinanza? (si/no)
si
No son flores, pero tienen plantas y olores. (escribe "otra" para cambiar de adivinanza)
planta
Te has equivocado, ¿quieres otra pista? (si/no)
si
Los zapatos les quedan muy bien
flores
Te has equivocado, ¿quieres otra pista? (si/no)
si
Es una parte del cuerpo
pies
Has acertado, ¿quieres una adivinanza? (si/no)
no
Chao pescao, has adivinado 2 adivinanzas.
FR/practicas/p2/ej5_app/bin on } master took 38s
➔
```

```
FR/practicas/p2/ej5_app on } master
➔ make
mkdir ./bin
mkdir: no se puede crear el directorio './bin': El fichero ya existe
make: [makefile:11: init] Error 1 (no tiene efecto)
javac -d ./bin/ ./src/*.java
FR/practicas/p2/ej5_app on } master
➔
```

```
java AdivinaServidorConcurrente
Recibida respuesta: si
Mandamos una nueva adivinanza
Recibida respuesta de adivinanza: otra
Recibido que quiere otra adivinanza, mandamos si quiere nueva adivinanza
Recibido que no quiere más adivinanza, mandamos despedida y cerramos la conexión
➔
Recibida conexión de un cliente, mandamos si quiere adivinanza
Recibida respuesta: si
Mandamos una nueva adivinanza
Recibida respuesta de adivinanza: vaca
Recibido información sobre si quiere pista: si
Mandamos la pista
Leemos nueva respuesta a la adivinanza
Recibido que quiere acierta la adivinanza, mandamos si quiere una nueva
Mandamos una nueva adivinanza
Recibida respuesta de adivinanza: planta
Recibido información sobre si quiere pista: si
Mandamos la pista
Leemos nueva respuesta a la adivinanza
Recibido información sobre si quiere pista: si
Mandamos la pista
Leemos nueva respuesta a la adivinanza
Recibido que quiere acierta la adivinanza, mandamos si quiere una nueva
Recibido que no quiere más adivinanza, mandamos despedida y cerramos la conexión
➔
```