

Ejercicio 5 - Práctica 2

Descripción, funcionalidad y actores que intervienen

Esta aplicación consiste en la elaboración de una “pokédex” haciendo uso del paradigma Cliente/Servidor en el que un usuario puede acceder con sus credenciales al servidor (previo registro). Una vez el usuario tenga una sesión activa, este puede registrar pokémons o ver la lista de pokemons ya registrados. La sesión finalmente acaba cuando el usuario la cierra. Por tanto, el protocolo de comunicación entre cliente y servidor debe encargarse de comunicar la información del cliente hacia el servidor así como del servidor hacia el cliente en forma de credenciales de acceso, nombres de pokemons o errores. Los actores que intervienen en la aplicación son:

- Usuario → Actúa como cliente de la aplicación. Las acciones que puede realizar son registrarse en el servidor o acceder a una cuenta ya registrada. Una vez realizado este paso, puede enviar nombres de pokemons al servidor para que este los registre a su nombre, pedir al servidor un listado de sus pokemons conocidos o cerrar la sesión.
- Servidor → Realiza la escucha pasiva del puerto en el que presta servicio, donde aguarda recibir una petición de conexión por parte de algún usuario. Cuando esto ocurre, abre un socket de servicio y lanza una hebra Procesador. Tras haber lanzado la hebra, vuelve a escuchar el puerto aunque el procesamiento de la petición anterior continúe, permitiendo dar un servicio concurrente a varios usuarios simultáneamente
- Procesador → Es el encargado de realizar todo el procesamiento en el lado del servidor. Maneja toda la comunicación entre usuario y servidor, encargándose de registrar nuevos usuarios o comprobar los credenciales de usuarios ya registrados. También es el encargado de crear los archivos que registran los pokemons de cada usuario así como de construir y enviar el listado de pokemons ya conocidos.

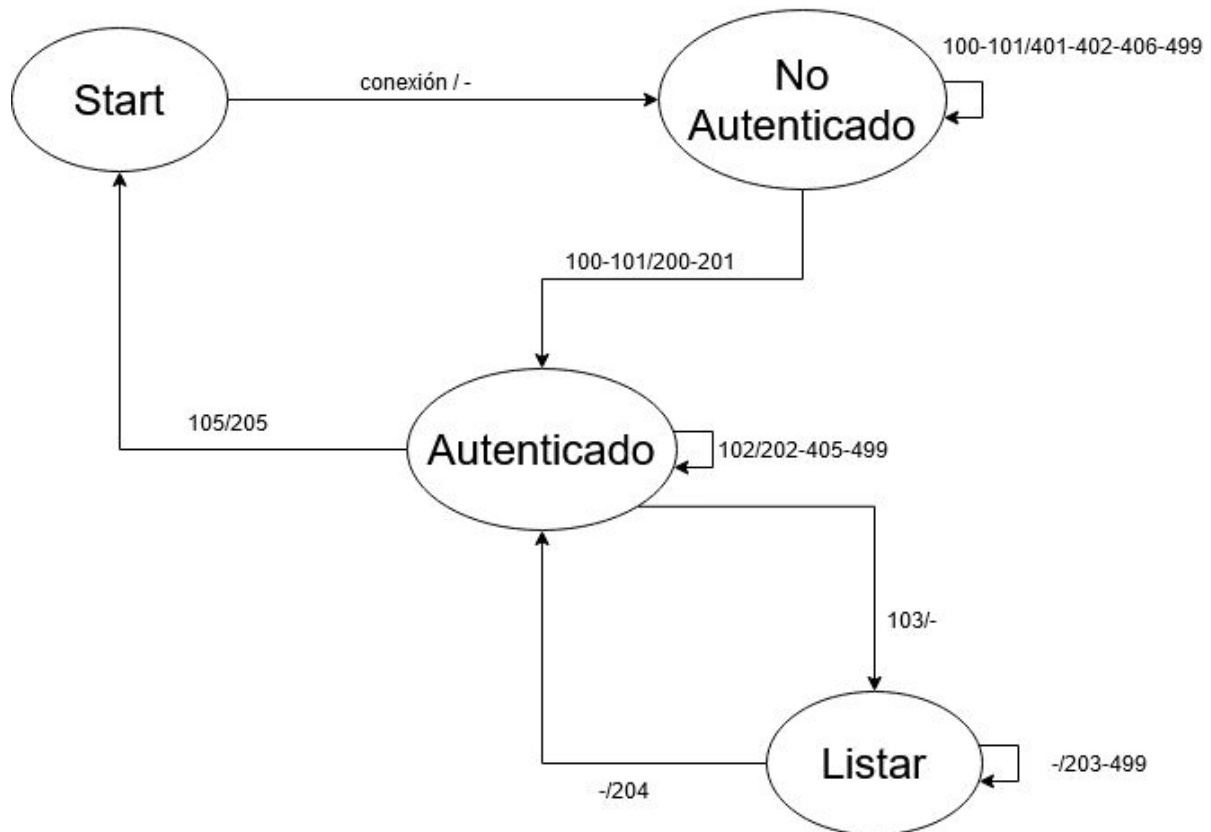
La aplicación creará un directorio Pokedex/ en el que guardará toda la información que necesite para operar. Esta es, un fichero users.pok en el que se recogen los credenciales de todos los usuarios registrados en formato usuario:contraseña, uno en cada línea. Cada vez que se registre un nuevo usuario, la hebra Procesador correspondiente añadirá sus credenciales al final de este archivo. Por cada usuario registrado habrá un subdirectorío con su nombre, en el que se recogerán sus pokemons conocidos. Por cada pokemon nuevo que el usuario registre se creará un archivo en su subdirectorío correspondiente con el nombre del pokemon que haya registrado y extensión .pok. Por ejemplo, si se tiene directorio *./Pokedex/Victor/* en el que están contenidos los archivos *Charizard.pok Lucario.pok Pikachu.pok* significa que el usuario Victor ha registrado en su pokedex los pokemon Charizard, Lucario y Pikachu, los cuales le serán listados si así se lo requiere al servidor.

Para evitar conflictos en el funcionamiento concurrente de la aplicación, cuando una hebra procesador inicia la sesión de un usuario, incluirá a dicho usuario en una lista de usuarios con sesión activa recogida en el servidor, de forma que si otra conexión diferente intenta

iniciar sesión con ese mismo usuario, esta será denegada hasta que no finalice la sesión activa.

Diagrama de estados del servidor

Formato de los mensajes:
Código de los mensajes recibidos / Código de los mensajes enviados



Mensajes que intervienen

Servidor

Código	Cuerpo	Descripción
200	OK-Usuario registrado	El servidor enviará al cliente este mensaje de confirmación para la petición de registro de usuario.
201	OK-Login correcto	El servidor enviará al cliente este mensaje de confirmación para la petición de autenticación de usuario.
202	OK-Pokemon registrado	El servidor enviará al cliente este mensaje de confirmación para la petición de registro de

		pokémon.
203	POK-<pokemon>	El servidor enviará al cliente este mensaje de información con el nombre de un Pokémon.
204	OK-Listado completado	El servidor enviará al cliente este mensaje de confirmación para la petición de una lista con todos los pokemons registrados.
205	LOGOUT-Bye	El servidor enviará al cliente este mensaje de confirmación para la petición de cierre de sesión.
401	ERROR-Ya existe este usuario	El servidor enviará al cliente este mensaje de error para la petición de registro de usuario en caso de que ya exista un usuario con el nombre solicitado.
402	ERROR-El usuario no existe	El servidor enviará al cliente este mensaje de error para la petición de autenticación de usuario en caso de que el usuario o la contraseña no coincidan con los de ningún usuario registrado.
405	ERROR-Pokemon ya registrado	El servidor enviará al cliente este mensaje de error para la petición de registro de Pokémon en caso de que el usuario ya tenga un Pokémon registrado con el mismo nombre.
406	ERROR-Ya hay una sesión activa para este usuario	El servidor enviará al cliente este mensaje de error para la petición de autenticación de usuario en caso de que ya haya iniciado sesión el usuario en otro lado simultáneamente.
499	ERROR-Error en el servidor	El servidor enviará al cliente este mensaje de error en caso de que se produzca algún error en el servidor.

Cliente

Código	Cuerpo	Descripción
100	REGISTER-LOGIN-<usuario> -PASSWORD-<contraseña>	El cliente enviará este mensaje al servidor para solicitar el registro de un nuevo usuario.
101	LOGIN-LOGIN-<usuario>- PASSWORD-<contraseña>	El cliente enviará este mensaje al servidor para solicitar la autenticación de un usuario.
102	POK-<pokemon>	El cliente enviará este mensaje al servidor para solicitar el registro de un nuevo Pokémon.
		El cliente enviará este mensaje al servidor

103	LIST	para solicitar una lista de todos los pokémons registrados en una pokédex.
105	LOGOUT	El cliente enviará este mensaje al servidor para solicitar el cierre de sesión de un usuario.

Evaluación de la práctica

Para la ejecución de esta aplicación será necesario lanzar en primer lugar el servidor, en una terminal aparte, y dejarlo en segundo plano. Para lanzarlo, ejecutaremos las siguientes órdenes:

```
javac Servidor.java // Para compilar el código
java Servidor       // Para ejecutar el servidor
```

```
victor@victor-HP:~/Escritorio/3º/FR/practicass/FR-P2.5-main$ javac Servidor.java
victor@victor-HP:~/Escritorio/3º/FR/practicass/FR-P2.5-main$ java Servidor
```

Una vez lanzado el servidor, abriremos otra terminal (sin cerrar la anterior) donde debemos lanzar el cliente ejecutando las siguientes órdenes, y nos aparecerá el menú de inicio.

```
javac Cliente.java // Para compilar el código
java Cliente       // Para ejecutar el cliente
```

```
victor@victor-HP:~/Escritorio/3º/FR/practicass/FR-P2.5-main$ javac Cliente.java
victor@victor-HP:~/Escritorio/3º/FR/practicass/FR-P2.5-main$ java Cliente
OPCIONES DEL LOGIN
1-Acceder
2-Registrarse
```

A continuación vamos a intentar iniciar sesión sin registrarnos y vemos que nos muestra un error de que no existe el usuario (ERROR 402).

```
OPCIONES DEL LOGIN
1-Acceder
2-Registrarse
1
Introduzca su nombre de usuario:
victor
Introduzca su contraseña:
ej5fr
ERROR El usuario no existe
```

Ahora procedemos a registrarnos, cerramos sesión, y la volvemos a iniciar para comprobar que el usuario se ha guardado correctamente, aunque esto no es necesario ya que el registro supone un inicio de sesión para el nuevo usuario registrado

```

OPCIONES DEL LOGIN
1-Acceder
2-Registrarse
2
Introduzca su nombre de usuario:
victor
Introduzca su contraseña:
ej5fr
MENÚ
1-Registrar Pokemon
2-Ver Pokemon Conocidos
3-Cerrar Sesión
3
victor@victor-HP:~/Escritorio/3º/FR/practicass/FR-P2.5-main$ java Cliente
OPCIONES DEL LOGIN
1-Acceder
2-Registrarse
1
Introduzca su nombre de usuario:
victor
Introduzca su contraseña:
ej5fr
MENÚ
1-Registrar Pokemon
2-Ver Pokemon Conocidos
3-Cerrar Sesión

```

Una vez registrados, vamos a ver los pokémon conocidos para comprobar que la lista está vacía, y después procederemos a registrar 2 pokémon y comprobar que podemos consultarlos perfectamente.

```

MENÚ
1-Registrar Pokemon
2-Ver Pokemon Conocidos
3-Cerrar Sesión
2
Los pokemons recogidos en tu pokedex son:
MENÚ
1-Registrar Pokemon
2-Ver Pokemon Conocidos
3-Cerrar Sesión
1
Introduzca el nombre del pokemon a registrar:
charmander
Se ha registrado a charmander en la pokedex
MENÚ
1-Registrar Pokemon
2-Ver Pokemon Conocidos
3-Cerrar Sesión
1
Introduzca el nombre del pokemon a registrar:
pikachu
Se ha registrado a pikachu en la pokedex
MENÚ
1-Registrar Pokemon
2-Ver Pokemon Conocidos
3-Cerrar Sesión
2
Los pokemons recogidos en tu pokedex son:
pikachu
charmander
MENÚ
1-Registrar Pokemon
2-Ver Pokemon Conocidos
3-Cerrar Sesión

```

Por último, vamos a cerrar sesión, y vamos a comprobar que dentro del directorio actual se ha creado un nuevo directorio Pokedex/. Dentro de este se ha creado un archivo users.pok

donde se almacenan todos los usuarios registrados y una carpeta con el nombre de cada usuario. Dentro de esta hay un archivo .pok con el nombre de cada pokémon registrado por el usuario.

```
3-Cerrar Sesión
3
victor@victor-HP:~/Escritorio/3º/FR/practicas/FR-P2.5-main$ ls
Cliente.class  Cliente.java  Pokedex  ProcesadorPokedex.class  ProcesadorPokedex.java  README.md  Servidor.class  Servidor.java
victor@victor-HP:~/Escritorio/3º/FR/practicas/FR-P2.5-main$ ls Pokedex/
users.pok  victor
victor@victor-HP:~/Escritorio/3º/FR/practicas/FR-P2.5-main$ cat Pokedex/users.pok
victor:ej5fr
victor@victor-HP:~/Escritorio/3º/FR/practicas/FR-P2.5-main$ ls Pokedex/victor/
charmander.pok  pikachu.pok
```