Raúl Rodríguez Pérez

Pedro Iniesta López

**Práctica 2. Ejercicio 5**

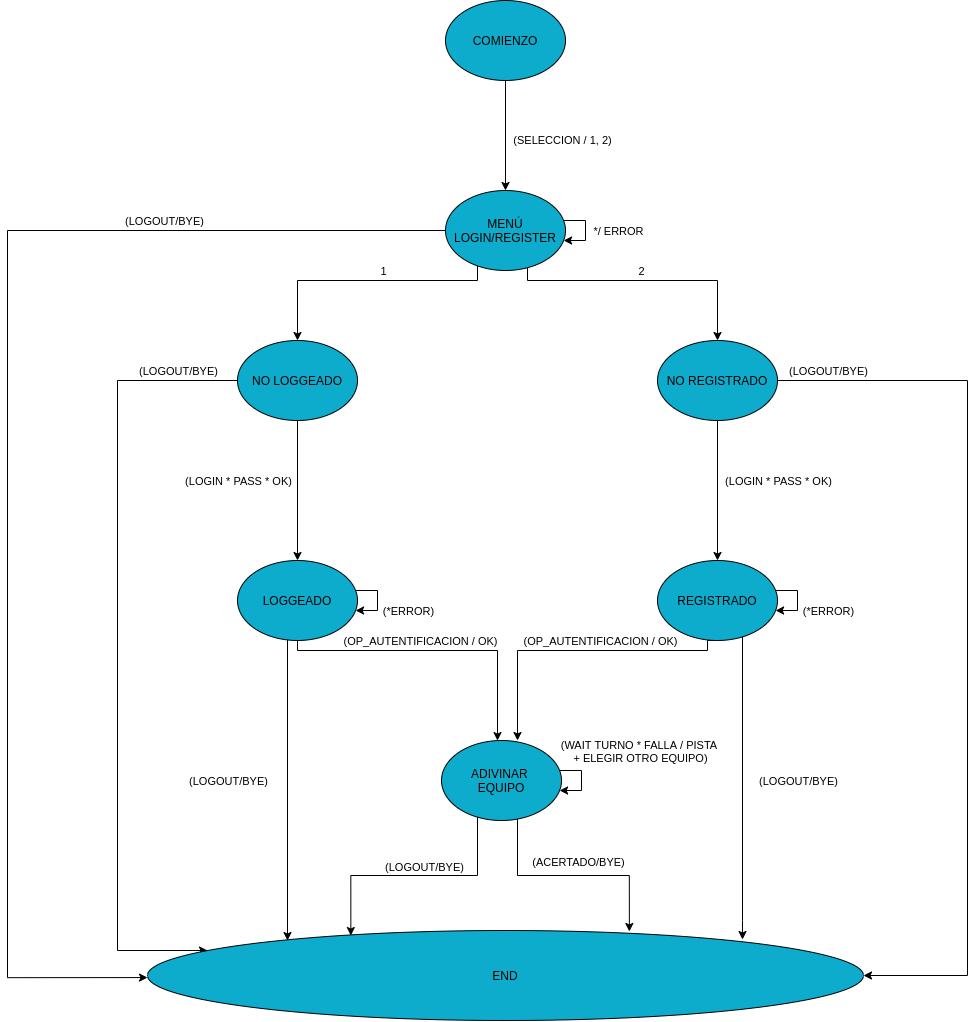
**1. Descripción, funcionalidad y actores**

La aplicación que hemos escogido para realizar el ejercicio 5 es una en la que se adivinan equipos de fútbol de la liga española. El funcionamiento es el siguiente, se conectan 4 jugadores, se elige un equipo aleatoriamente (hay un total de 10 equipos, y cada equipo tiene asociadas 3 pistas). Los jugadores comenzarán adivinando del equipo que se trata, si fallan se le muestra la primera pista. En el momento que se agoten las 3 pistas, se les indicará a los jugadores que no quedan más pistas pero aún así pueden seguir intentándolo. Cuando acierte un jugador, el juego terminará para todos.

***Clases y su funcionalidades***

* Servidor: comienza el juego, es decir, si no está activado el servidor, no se podrán conectar los clientes. Obtiene un equipo aleatorio de la clase Jugadores, crea y llama a un objeto de la clase Principal, donde se lleva a cabo un intercambio de mensajes con el cliente.
* Cliente: lleva a cabo una recogida de datos para iniciar sesión o registrarse, que más tarde son utilizadas para llamar a la clase LoginRegister. Después, guardará la opción del equipo que el cliente cree que es y la envía a la clase Principal, que le devolverá una respuesta de si es correcto, o tiene que mostrarle una pista en caso de que queden.
* LoginRegister: lee y escribe en el fichero de usuarios, donde están los usuarios y sus contraseñas.
* Jugadores: aquí se guarda el número de clientes conectados, los turnos que tiene cada jugador y también es donde se genera el equipo de fútbol que se va a acertar.
* Principal: es la clase donde se lleva a cabo un intercambio de mensajes, tanto con el cliente como con el servidor. Lleva el flujo del juego y comprueba si las respuestas de los usuarios son correctas.

**2. Diagrama de estados del servidor**



**3. Mensajes que intervienen**

***Mensajes del cliente***

|  |  |  |
| --- | --- | --- |
| **Código** | **Cuerpo** | **Descripción** |
| 1000 | SELECCIÓN + (1 O 2)  1.- Log in  2.- Registrarse | Le decimos al servidor cómo queremos entrar, si con log in o con registro |
| 1001 | LOGIN + usuario + contraseña | Se le envía al servidor para entrar haciendo log in |
| 1002 | REGISTER + usuario + contraseña | Se le envía al servidor para registrarse |
| 1003 | EQUIPO + equipo | Le envía el equipo que cree que es para ver si ha acertado |

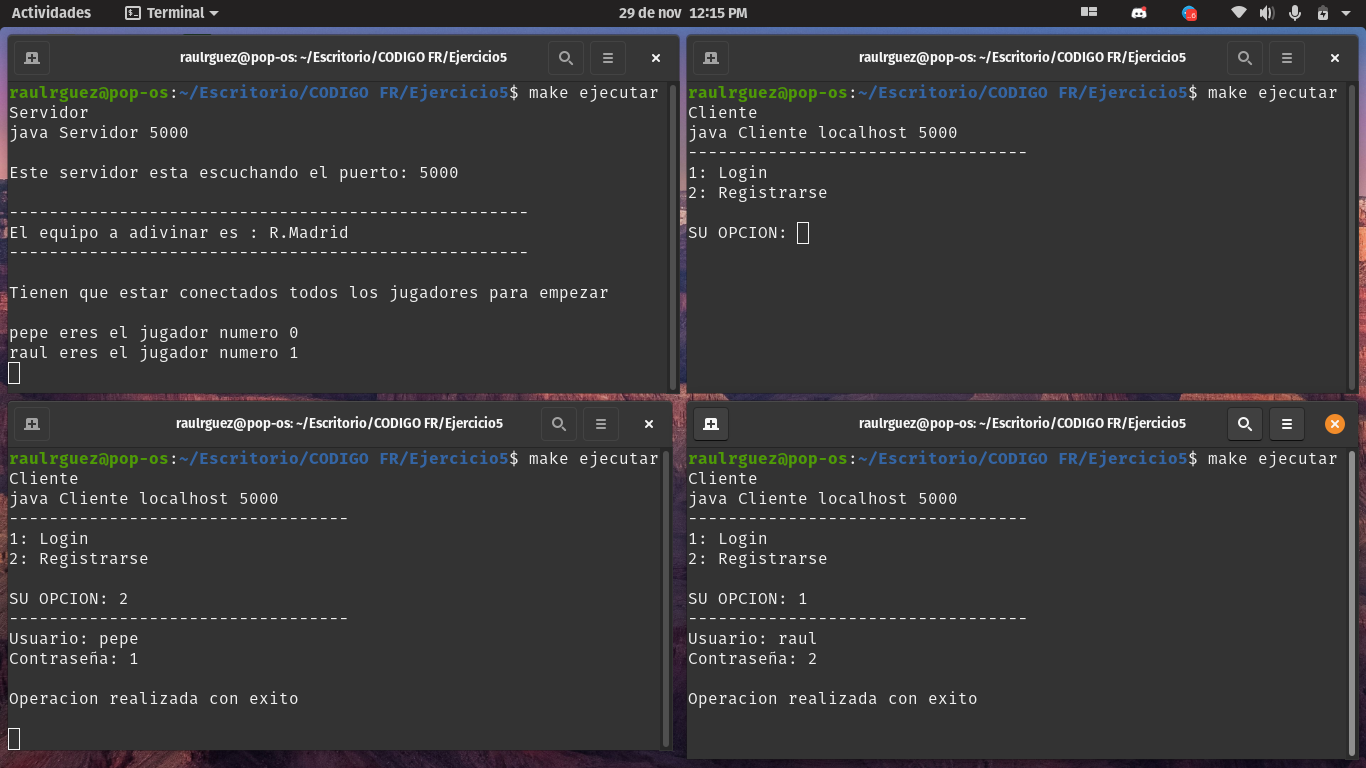
***Mensajes del servidor***

|  |  |  |
| --- | --- | --- |
| **Código** | **Cuerpo** | **Descripción** |
| 200 | “ERROR: usuario y contraseña no existentes” | Mensaje basado en un error de autentificación |
| 201 | OK + “loggeado” | Mensaje basado en un correcto log in |
| 202 | “ERROR: el usuario introducido ya existe” | Mensaje basado en un error de registro sobre el usuario |
| 203 | ERROR + “no se ha podido realizar la operación” | Error al registrarse/loggin en el servidor |
| 204 | OK + “Registrado” | Éxito al registrarse en el servidor |
| 205 | LIMITE JUGADORES + “Límite de jugadores” | Máximo de jugadores permitidos |
| 206 | PISTA + “Pista correspondiente” | Mensaje que envía una pista sobre el equipo al cliente |
| 207 | NO MÁS PISTAS + “Perdedor” | Mensaje que indica cuando no quedan más pistas |
| 208 | Acierto + “Acierto” | Mensaje generado cuando un jugador adivina el equipo |
| 209 | “Turno del jugador: “ + nº jugador | Mensaje enviado para notificar a qué jugador le corresponde que juegue |
| 210 | hayGanador + “Ha ganado el jugador: ” + nºjugador | Mensaje enviado para notificar que jugador ha ganado el juego |

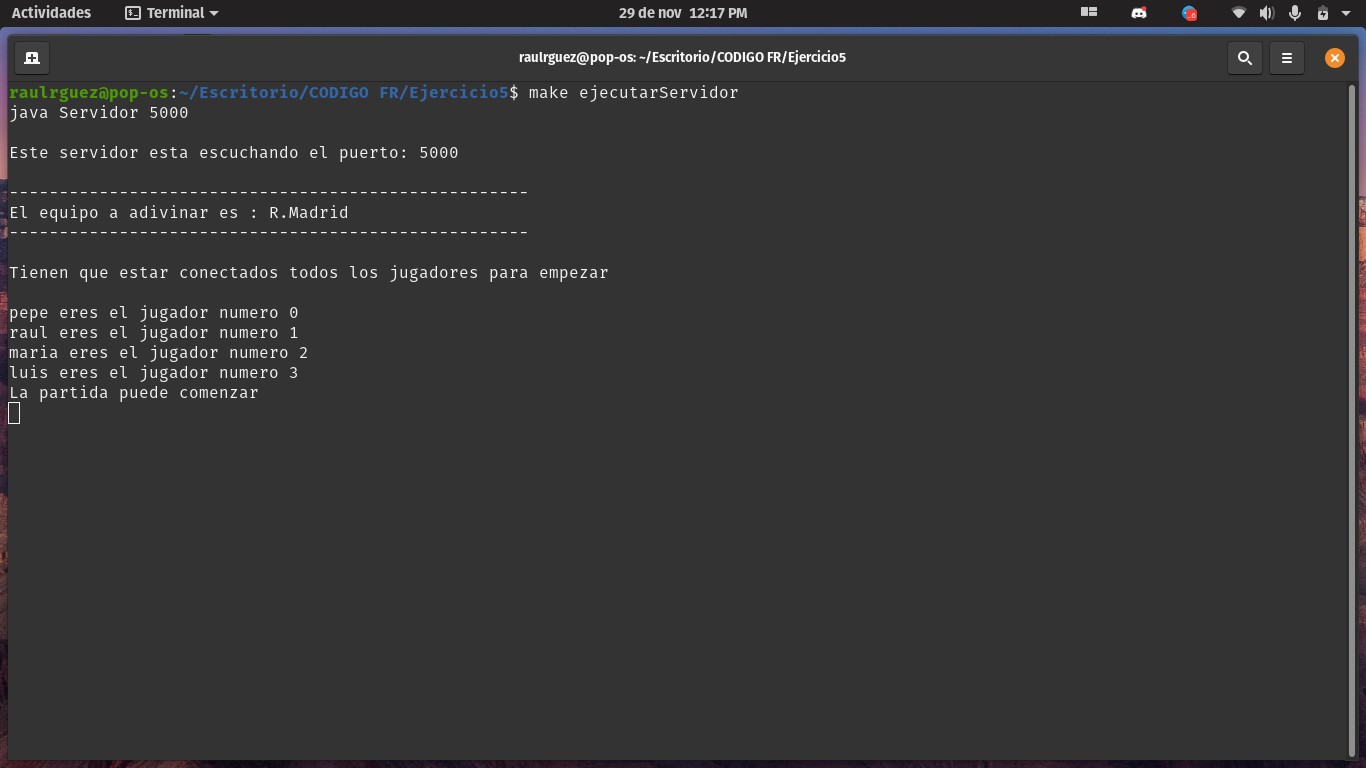
**4. Capturas de pantalla**

Para realizar un ejemplo de uso del juego, vamos a proceder a abrir 5 terminales (1 terminal para el servidor y 4 para los jugadores necesarios para empezar el juego). En primer lugar, en el terminal del servidor hacemos make para compilar todos los archivos necesarios para jugar y, a continuación, ejecutamos la sentencia ‘make ejecutarServidor’. Esto iniciará el juego eligiendo el equipo de fútbol que se ha de adivinar, y se quedará esperando a que los jugadores entren.

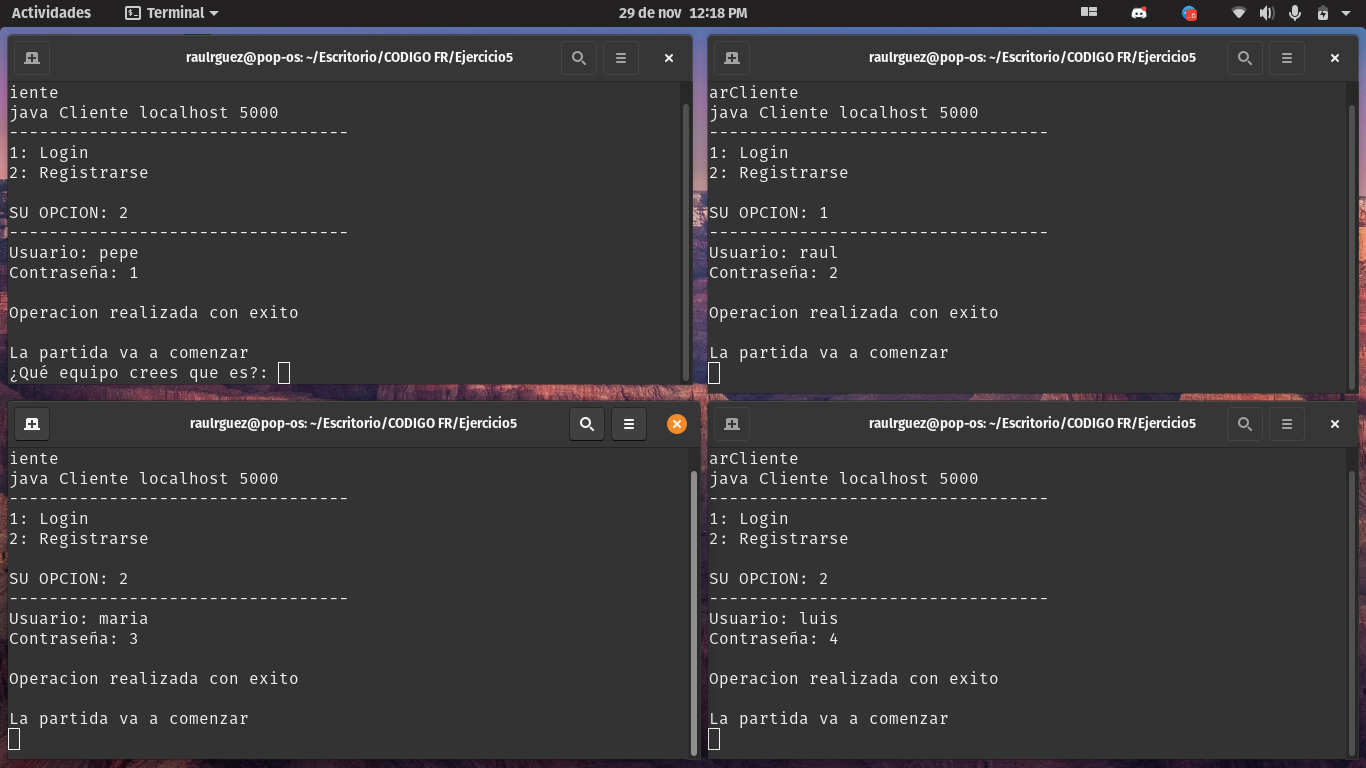
Tras esto, tendremos que ejecutar la sentencia ‘make ejecutarCliente’ en los otros terminales que hemos abierto, y después elegir si queremos iniciar sesión o registrarnos para jugar. Si elegimos la primera opción, es porque ya tenemos nuestro usuario y contraseña creado, es decir, no es la primera vez que jugamos y ya nuestros datos están guardados en usuarios.txt. Por otro lado, si es la primera vez que jugamos, es necesario que nos registremos para poder jugar. Esto consiste simplemente en rellenar nuestros datos (usuario-contraseña).

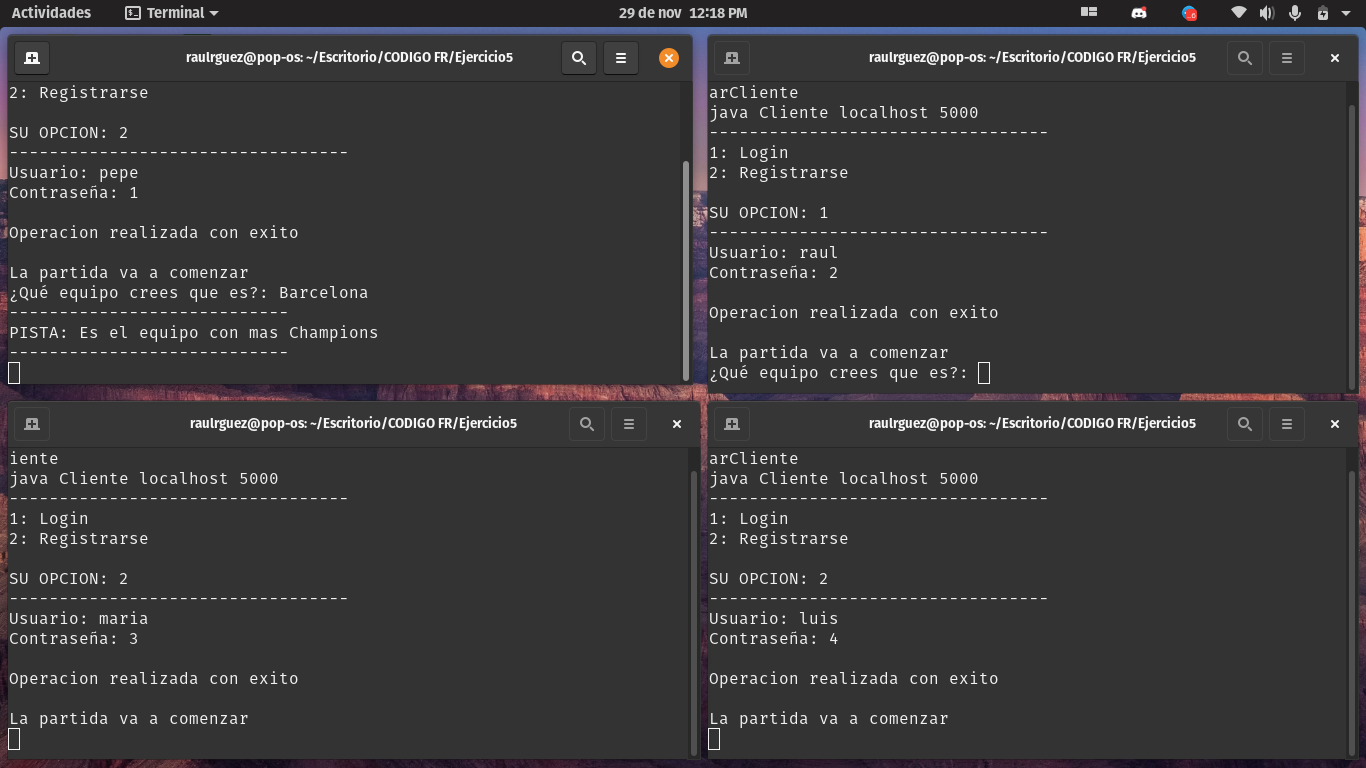


Una vez que todos los jugadores están listos, el servidor nos lo notifica y puede dar comienzo a la partida



A continuación, el juego da comienzo. En la primera ronda, se le pregunta a cada jugador (en orden), cuál cree que es el equipo a adivinar. Si por ejemplo, el jugador 1 responde con un equipo erróneo, el juego le proporcionará una pista, y pasará al turno del jugador 2. Este tipo de ronda se repetirá hasta que algún jugador adivine el equipo. Cabe destacar que el juego le irá proporcionando pistas a los jugadores cada vez que fallen, pero dichas pistas no son infinitas. Para cada equipo existen un número limitado de ellas, por lo que, cuando ya se han notificado todas las pistas a un jugador, este podrá seguir adivinando pero sin recibir más ayuda por parte del juego.





Finalmente, cuando un jugador adivina el equipo, se le notifica tanto a él su victoria, como a los demás jugadores que han perdido.

