**Jugador:**

Nombre: AA\_Player

Parámetros: IP y puerto de escucha de AA\_Engine. IP y puerto de escucha de Kafka. IP y puerto de escucha de AA\_Registry.

Atributos:

* PosiciónX: entero. Puede tomar valores de 1 hasta 20.
* PosiciónY: entero. Puede tomar valores de 1 hasta 20.
* Nivel: entero. Puede tomar valores entre -1 y +infinito.
* Efecto Frío: entero: Puede tomar valores entre -10 y +10.
* Efecto Calor: entero: Puede tomar valores entre -10 y +10.

Funcionalidades:

* Al arrancar: Debe mostrar un menú, con tres opciones posibles: Crear perfil, editar los datos de perfil, unirse a una partida. Se inicializarán a 0 todos los atributos.
* Crear perfil: Solicitará al usuario un nombre y una contraseña. Se conectará al módulo AA\_Registry y enviará los dos datos, quedando a la espera de un mensaje correcto o de error.
* Editar perfil: Solicitará al usuario su nombre, su contraseña actual, y la nueva contraseña. Se conectará al módulo AA\_Registry y enviará los tres datos, quedando a la espera de un mensaje correcto o de error.
* Unirse a la partida: Cuando el usuario elija esta opción, enviará el usuario y la contraseña a la aplicación AA\_Engine, y permanecerá a la espera de respuesta. En caso de recibir un ok, pasará a la funcionalidad de jugar. En caso de recibir un error de autenticación, se le volverá a solicitar el usuario y contraseña.
* Jugar: El sistema leerá el nivel del jugador. Si es igual a 0, mostrará un mensaje de “esperando inicio de partida”. Si el nivel es mayor que 0, ejecutará tres funcionalidades: mostrar mapa, enviar movimiento, y actualizar estado. Si el nivel es menor que 0, el jugador ha muerto durante la partida, y la aplicación finaliza.
* Mostrar mapa: El sistema obtendrá del bróker la información del mapa, y lo dibujará en consola.
* Enviar movimiento: El sistema mostrará un mensaje por pantalla donde indicará al jugador que puede enviar una dirección para moverse, en 8 direcciones posibles, representadas por los puntos cardinales principales y secundarios (N – arriba, S – abajo, E – derecha, W, izquierda, NE – arriba/derecha, NW – arriba/izquierda, SE – abajo/derecha, SW – abajo/izquierda). Además, permanecerá a la espera de que una de estas direcciones sea introducida por teclado. El sistema debe controlar que sólo se introduce una dirección válida. En caso de no ser así, se mostrará un mensaje de error y volverá a pedir una dirección correcta. Cuando la dirección sea correcta, se enviará al servidor.
* Actualizar estado: El sistema obtendrá del bróker la información del estado del jugador, y lo actualizará internamente.

**Servidor de registro:**

Nombre: AA\_Registry

Parámetros: Puerto de escucha

Funcionalidades:

* Al arrancar: Permanecerá indefinidamente a la escucha hasta que llegue una petición de registro o modificación de jugador.
* Registrar nuevo jugador: El sistema leerá el usuario y contraseña del nuevo jugador. Debe comprobar que el usuario no existe en BD. En caso de existir, devolverá un mensaje de error. En caso de no existir, creará un nuevo jugador con nivel 1, resistencia al frío y al calor con valor 0, nombre de usuario y contraseña. Y devolverá un mensaje de éxito.

Modificar datos de jugador: El sistema leerá el nombre de usuario, la contraseña antigua y la nueva contraseña, y realizar los siguientes pasos:

1 – Verificar que el usuario existe

2 – Verificar que la contraseña es correcta

3 – Sustituir la contraseña antigua, por la nueva

4 – Enviar mensaje de ok

En caso de que los puntos 1 ó 2 no se cumplan, se debe enviar un mensaje de error.

Métodos:

* RegistrarJugador(string): La aplicación recibirá un string, que contendrá el nombre de jugador y la contraseña, separados por el carácter punto y coma. El método hará el Split de ambos datos mediante el carácter punto y coma. Una vez tiene los dos datos, se deberá comprobar que el usuario no existe en BD. En caso de existir, devolverá un mensaje de error. En caso de no existir, creará un nuevo jugador con nivel 1, resistencia al frío y al calor con valor 0, y con el nombre de jugador y contraseña especificados en el string. Y devolverá un mensaje de éxito.
* ModificarJugador(string): La aplicación recibirá un string, que contendrá el nombre de jugador, la contraseña antigua y la contraseña nueva, separados por el carácter punto y coma. El método hará el Split de ambos datos mediante el carácter punto y coma. Una vez tiene los tres datos, el proceso es el siguiente:

1 – Verificar que el usuario existe

2 – Verificar que la contraseña es correcta

3 – Sustituir la contraseña antigua, por la nueva

4 – Enviar mensaje de ok

En caso de que los puntos 1 ó 2 no se cumplan, se debe enviar un mensaje de error.

**Servidor de clima:**

Nombre: AA\_Weather

Parámetros: Puerto de escucha

Funcionalidades:

* Al arrancar: Permanecerá a la escucha hasta que le llegue una petición de obtener una ciudad.
* Devolver ciudad: El sistema debe devolver, aleatoriamente, un nombre de ciudad y su temperatura media.

Métodos:

* Al arrancar: Debe leer un fichero de texto de ciudades, y almacenar cada línea del fichero en una casilla de un array llamado “ciudades”. Cada línea del fichero de texto tendrá el formato “NombreCiudad;temperatura”.
* getCiudad(): El sistema calculará un número aleatorio comprendido entre 1 y la cantidad de elementos del array “ciudades”. Accederá al índice correspondiente al número calculado, y devolverá el dato contenido en el array.