 Proyecto Cupi2	<b>ISIS-1205 Algorítmica y Programación</b> <b>Descripción</b>
Ejercicio:	N12_CupiBlog
Autor:	Equipo Cupi2
Semestre:	2019-1

## Enunciado

El equipo Cupi2 desea construir una aplicación que permita manejar un sistema de Blogs llamado CupiBlog, un espacio que les permite a los usuarios crear artículos con contenidos de su interés y calificar artículos de otros usuarios. La información de los usuarios, así como de sus artículos, va a ser persistente en una base de datos.

Para registrarse al sistema, un usuario debe proveer la siguiente información:

- Login de usuario, que será su forma de autenticación. El login debe ser único en el sistema.
- Nombre del usuario.
- Apellido del usuario.

Los usuarios pueden visualizar y publicar artículos sobre diferentes categorías. Un usuario no puede tener dos artículos con el mismo nombre. De los artículos se conoce la siguiente información:

- Login del usuario que escribió el artículo. Para que un usuario pueda registrar un artículo, este debe estar previamente registrado.
- Título del artículo.
- Categoría del artículo. La categoría puede ser: música, televisión, cine, mascotas, comics, viajes, vida diaria u otros.
- Contenido del artículo. Corresponde al escrito del usuario.
- Fecha de publicación.
- Calificación acumulada. Corresponde a la suma de las calificaciones del artículo.
- Cantidad de veces que el artículo ha sido calificado.

La aplicación debe ser construida como un programa distribuido tipo cliente-servidor para permitir el acceso de varios usuarios simultáneos al sistema. Un programa distribuido está compuesto por dos programas: un servidor, encargado de mantener la información de los usuarios registrados y sus artículos publicados; y un cliente, a través del cual los usuarios pueden acceder al CupiBlog.

El programa cliente debe ofrecer las siguientes funcionalidades:

1. Conectarse al servidor: Para esto, el usuario debe suministrar la IP del servidor al que se va a conectar y por defecto, el cliente se conectará al puerto 9999 del servidor con la IP dada.
2. Iniciar sesión: El usuario envía la solicitud de inicio de sesión al servidor con su login. Si el login se encuentra en la base de datos, inicia sesión. Si no se encuentra, entonces debe registrarse.

3. Registrarse en el sistema: El usuario debe proveer sus datos personales: login (que luego usará para iniciar sesión), nombre y apellido. Una vez registrado el usuario, el programa debe enviar una solicitud de inicio de sesión.
4. Publicar un artículo: El usuario publica un artículo dado su título, categoría y contenido.
5. Visualizar un artículo de la lista de artículos.
6. Solicitar la lista de artículos de todo el sistema.
7. Calificar un artículo: El usuario puede calificar un artículo de otro usuario (no puede calificar artículos propios) con valores discretos de 0 a 5.
8. Buscar artículos dada una categoría.
9. Consultar las estadísticas de publicación: Las estadísticas contienen la información sobre el total de artículos que ha publicado el usuario y el promedio de calificación de todos sus artículos.
10. Cerrar sesión: Cuando el usuario cierra el programa cliente, este le anuncia al servidor que va a cerrar la sesión y se termina la comunicación con el servidor.

Por otro lado, el servidor debe:

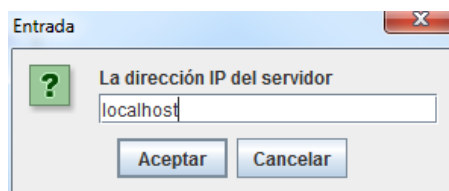
1. **Esperar las conexiones de los usuarios.**
2. **Validar el inicio de sesión de un usuario:** Debe buscar en la base de datos el login del usuario que desea iniciar sesión. Si existe, valida el inicio de sesión y guarda una referencia del usuario que inició sesión. Si no existe, envía un mensaje de error al cliente.
3. **Registrar un usuario al sistema:** Registra un nuevo usuario a la base de datos.
4. **Registrar un artículo:** Cuando el usuario desea publicar un artículo, el programa servidor debe guardar la información de este artículo en la base de datos y actualizar la cantidad de artículos publicados por el usuario.
5. **Registrar la calificación de un artículo:** Cuando el usuario califica el artículo de otro usuario, el sistema debe registrar en la base de datos la calificación y la cantidad de veces que el artículo ha sido calificado.
6. **Consultar la lista de todos los artículos del sistema.**
7. **Consultar la lista de los artículos de una categoría dada.**
8. **Consultar las estadísticas de publicación de un usuario.**
9. **Cerrar la sesión del usuario:** Cuando el usuario indica que va a cerrar sesión, el servidor debe cerrar el canal de comunicación.

Además, el servidor debe ofrecer las siguientes opciones al administrador del CupiBlog

10. **Visualizar la lista de usuarios conectados.**
11. **Visualizar la lista de artículos de un usuario.**

## Interfaz del cliente

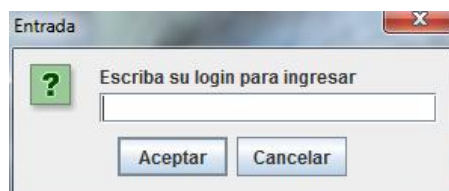
### Conexión al servidor:



**Ilustración 1. Petición de IP del servidor.**

Al iniciar el programa cliente, se abre una ventana que solicita la dirección IP del servidor (como se ve en la Ilustración 1), el usuario ingresa la dirección IP del servidor en la ventana y el programa se conecta al servidor con esa IP al puerto 9999.

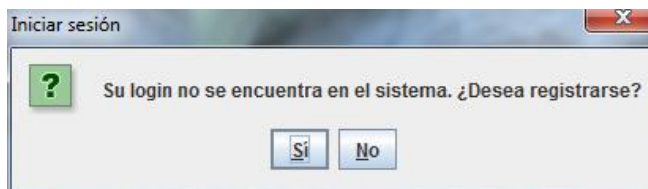
### Inicio de sesión:



**Ilustración 2. Petición de login al usuario.**

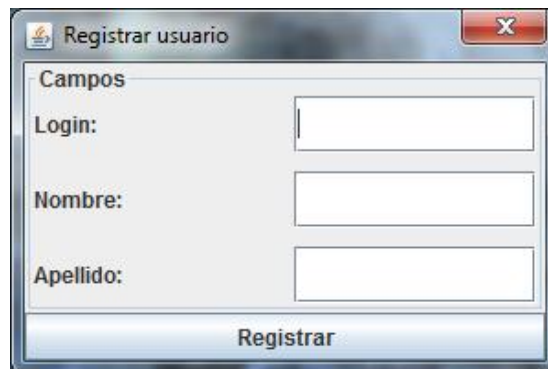
Una vez el usuario ha ingresado la dirección IP del servidor al que se quiere conectar, el programa le solicita el login de usuario para poder iniciar sesión (Ilustración 2). Si el login se encuentra registrado en el sistema, se inicia la sesión, si no, se le solicita al usuario que se registre en el sistema.

### Registro de un usuario:



**Ilustración 3. Mensaje de confirmación para registro del usuario.**

Si el login de usuario no se encuentra en el sistema, el programa cliente muestra un nuevo diálogo (Ilustración 3) preguntando si se desea registrar al sistema. Si el Usuario hace clic en la opción No, se cierra la interfaz, en caso contrario, se despliega el diálogo de registro de usuario.



**Ilustración 4. Diálogo de registro de usuario.**

Si el usuario hace clic en la opción Si del diálogo de registro de usuario (Ilustración 3), entonces se despliega el diálogo presentado en la Ilustración 4, en el cual el usuario debe ingresar su información: Login con el que iniciará sesión en otras ocasiones, su nombre y su apellido.

### Interfaz de sesión:



**Ilustración 5. Interfaz de usuario.**

Una vez el usuario se registra o inicia sesión, se le presenta la interfaz gráfica por defecto (Ilustración 5). Si el usuario desea visualizar un artículo de otro usuario, se activan las opciones de calificación (como se puede apreciar en la Ilustración 6).



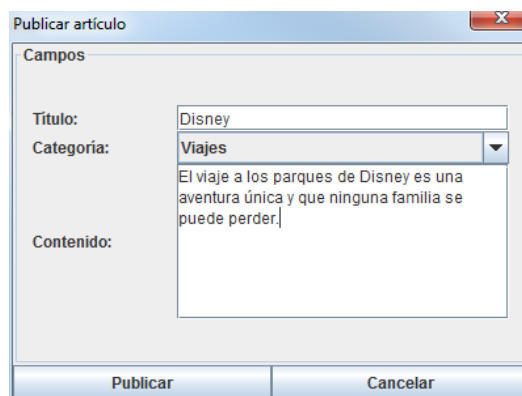
**Ilustración 6. Visualización de un artículo de un usuario.**

Por otro lado, si el usuario desea visualizar un artículo propio, entonces se deshabilitan los botones de calificación de artículo (Ilustración 7).



**Ilustración 7. Visualización de un artículo propio.**

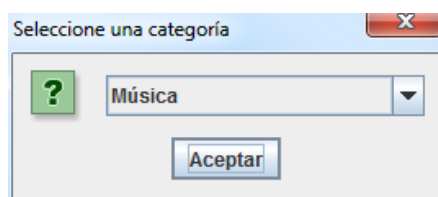
**Publicar un artículo:**



**Ilustración 8. Diálogo para publicar un artículo.**

En la Ilustración 8 se visualiza el diálogo para publicar un artículo. En este, el usuario debe especificar el título, categoría y contenido del artículo.

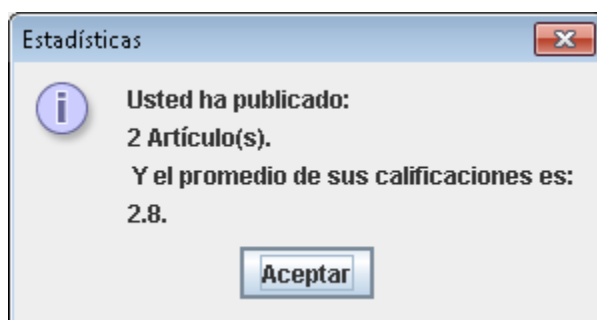
### Buscar un artículo por categoría:



**Ilustración 9. Diálogo para seleccionar una categoría.**

En la Ilustración 9 se visualiza el diálogo para buscar artículos por categoría. En este, el usuario selecciona una categoría, hace clic en el botón de Aceptar y en la interfaz principal (Ilustración 5, Ilustración 6 o Ilustración 7) se actualiza la lista de artículos con los artículos pertenecientes a la categoría buscada.

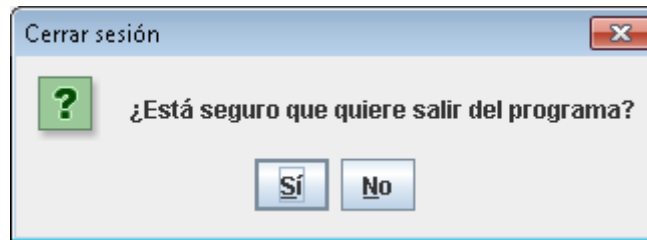
### Estadísticas:



**Ilustración 10. Mensaje con las estadísticas del usuario.**

La Ilustración 10 muestra las estadísticas del usuario, en donde se especifica la cantidad de artículos que ha publicado y el promedio de las calificaciones de todos sus artículos.

### Cierre de Sesión:

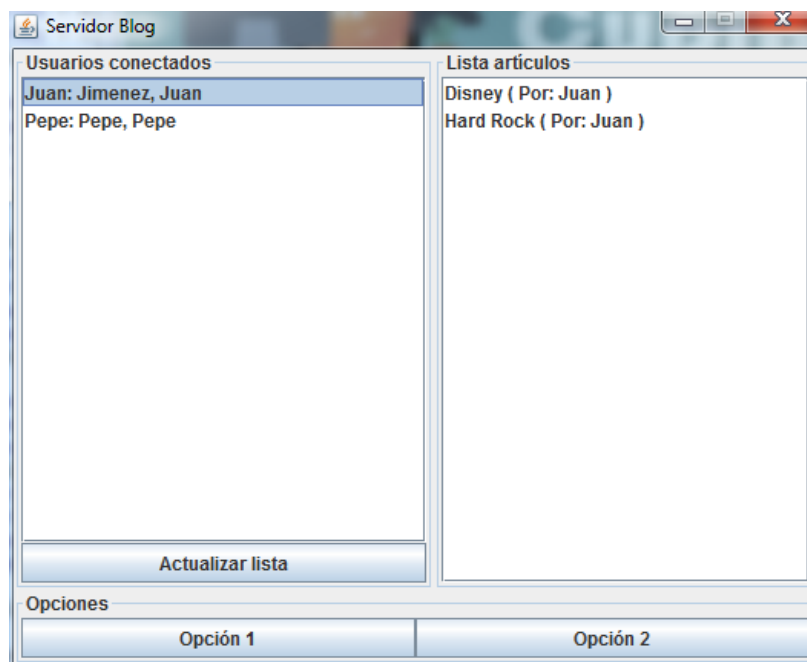


**Ilustración 11. Diálogo de cierre de sesión.**

La Ilustración 11 muestra el diálogo de salida cuando el usuario cierra el programa. Si el usuario acepta salir del programa, se cierra la sesión.



## Interfaz del servidor



**Ilustración 12. Interfaz del servidor.**

La interfaz del servidor (Ilustración 12) permite actualizar la lista de los usuarios actualmente conectados al sistema. Cuando se selecciona uno de los usuario, se visualiza la lista de artículos que han publicado.

## Persistencia

Para el desarrollo de la aplicación CupiBlog, se utilizará como motor de base de datos el sistema (open source) llamado Derby, desarrollado en el proyecto Apache Derby.

Las ventajas de utilizar un motor de base de datos Derby son:

- Está desarrollado completamente en Java y tiene un sistema de persistencia basado en archivos.
- Permite crear un servidor Derby independiente de la aplicación y establecer la comunicación con esta a través de sockets y desde diferentes máquinas.
- Es una base de datos embebida, es decir, que se puede utilizar como si fuera parte de la aplicación.

La base de datos de la aplicación de dos tablas:

- La primera tabla es utilizada para almacenar los datos de los usuarios del sistema. La tabla incluye los siguientes campos: login del usuario, nombre del usuario, apellido del usuario y los artículos publicados. La llave primaria de esta tabla es el login de usuario. Esto quiere decir que no pueden existir dos usuarios con el mismo login.

- La segunda tabla es utilizada para almacenar la información de los artículos del sistema. La tabla incluye los siguientes campos: login del usuario que escribió el artículo, título, categoría, contenido, fecha de publicación, calificación acumulada y la cantidad de veces que ha sido calificado. La llave primaria de esta tabla es una llave compuesta por el login del usuario y el título del artículo. Esto quiere decir que un mismo usuario no puede tener dos artículos con el mismo título.

La Tabla 1 muestra el diseño de las dos tablas de la base de datos con sus respectivos campos y llaves primarias.

Tabla	Campos	Tipo	Llave primaria
usuarios	login	varchar(32)	login
	nombre	varchar(50)	
	apellido	varchar(50)	
	articulos_publicados	int	
articulos	login_usuario	varchar(32)	login_usuario, titulo
	titulo	varchar(32)	
	categoria	varchar(20)	
	contenido	varchar(10000)	
	fecha_publicacion	varchar(20)	
	calificacion_acumulada	int	
	veces_calificado	int	

Tabla 1. Diseño de las tablas para la base de datos.

## Protocolo de comunicación

A continuación se presenta el protocolo de comunicación que establece cuáles mensajes (y en qué orden) se deberán enviar, para realizar cada una de las tareas del sistema. Es responsabilidad tanto del servidor como de los clientes ser capaz de interpretar este protocolo y realizar las tareas necesarias.

En términos generales, los mensajes que se envían tienen el siguiente formato:

**<COMANDO>;;<PARÁMETRO1>;;<PARÁMETRO2>; ... <PARÁMETROn>**

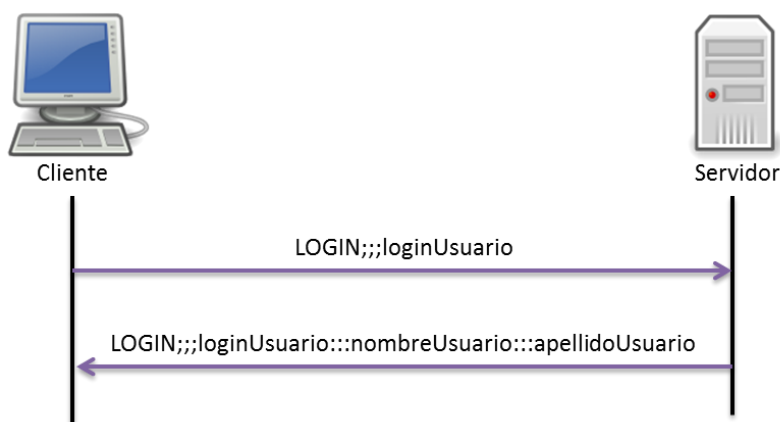
En dónde:

- <COMANDO>: Indica la solicitud del usuario al servidor o la respuesta del servidor al usuario.

- <PARAMETRO>: Indica la información necesaria para resolver la petición.
- Para diferenciar el comando de los parámetros, se utiliza el separador “;;;” (3 “punto y coma”).
- Para diferenciar los parámetros, se utiliza el separador “:::” (3 “dos puntos”).

## Protocolo para iniciar sesión:

**Caso 1:** El usuario inicia sesión correctamente en el servidor:



**Ilustración 13. Protocolo de inicio de sesión.**

En la Ilustración 13 se aprecia el protocolo de inicio de sesión que consiste en que el programa cliente envía un mensaje al servidor con el comando LOGIN seguido del login del usuario que desea iniciar sesión. Si el servidor encuentra el usuario con ese login en la base de datos, entonces envía un mensaje de respuesta con el comando LOGIN y la información de usuario: el login, nombre y apellido. El servidor guarda una referencia del usuario que inició sesión.

Las Tablas 2 y 3 explican la sintaxis de cada mensaje:

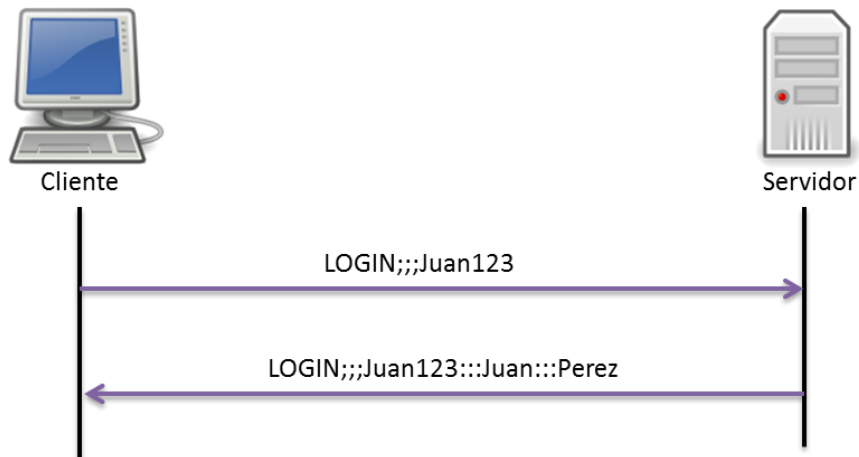
Comando Cliente	Parámetros	Descripción
<b>LOGIN</b>	loginUsuario	Representa el login del usuario que desea iniciar sesión.

**Tabla 2. Sintaxis del mensaje del cliente para el protocolo de inicio de sesión.**

Comando Servidor	Parámetros	Descripción
<b>LOGIN</b>	loginUsuario	Representa el login del usuario que inició sesión.
	nombreUsuario	Representa el nombre del usuario que inició sesión.
	apellidoUsuario	Representa el apellido del usuario que inició sesión.

**Tabla 3. Sintaxis del mensaje del servidor para el protocolo de inicio de sesión.**

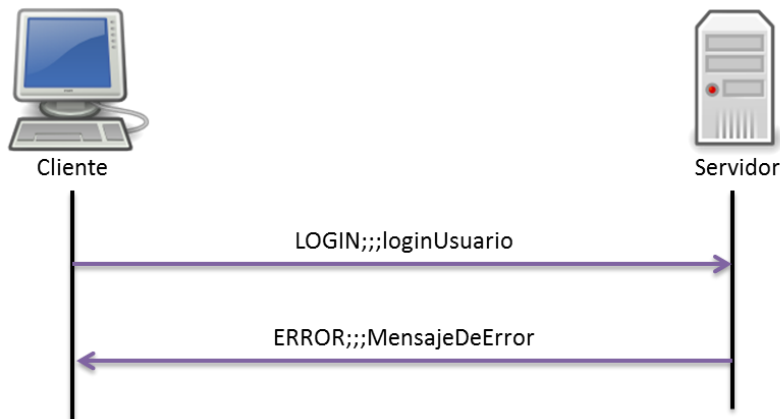
A continuación, un ejemplo concreto:



**Ilustración 14. Ejemplo del protocolo de inicio de sesión.**

En la Ilustración 14 se ilustra el escenario en el que el usuario Juan Pérez desea iniciar sesión, por lo tanto el programa envía el mensaje con comando LOGIN y el login Juan123. El servidor encuentra en su base de datos un usuario con este login y le responde al usuario enviando el mensaje con comando LOGIN, el login Juan123, el nombre Juan y el apellido Pérez.

**Caso 2:** Un usuario que no está registrado al sistema trata de iniciar sesión:



**Ilustración 15. Protocolo de inicio de sesión cuando ocurre un error.**

La Ilustración 15 muestra el protocolo de inicio de sesión en presencia de un error. Esto ocurre cuando el programa cliente envía un mensaje al servidor con el comando LOGIN y su login. El servidor no encuentra en su base de datos un usuario con el login dado, por lo tanto le envía al programa cliente un mensaje con el comando ERROR y el mensaje de error.

Las Tablas 4 y 5 explican la sintaxis de cada mensaje:

Comando Cliente	Parámetros	Descripción
-----------------	------------	-------------

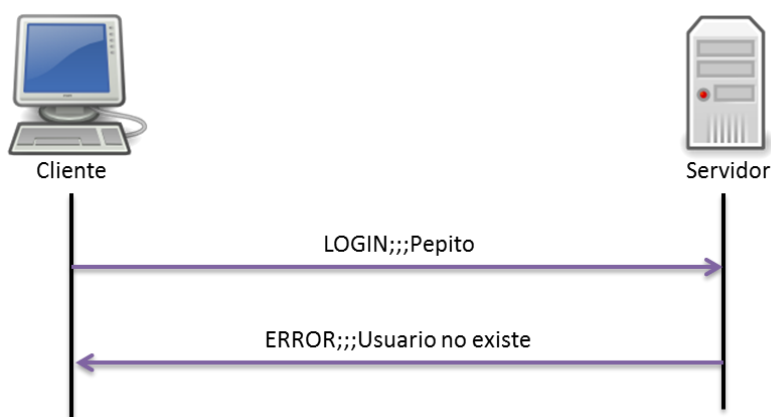
LOGIN	LoginUsuario	Representa el login del usuario que desea iniciar sesión.
-------	--------------	---

**Tabla 4. Sintaxis del mensaje del cliente para el protocolo de inicio de sesión.**

Comando Servidor	Parámetros	Descripción
ERROR	MensajeDeError	Representa el mensaje de error que se quiere informar al usuario.

**Tabla 5. Sintaxis del mensaje del servidor para el protocolo de inicio de sesión (caso de error).**

A continuación, un ejemplo concreto:

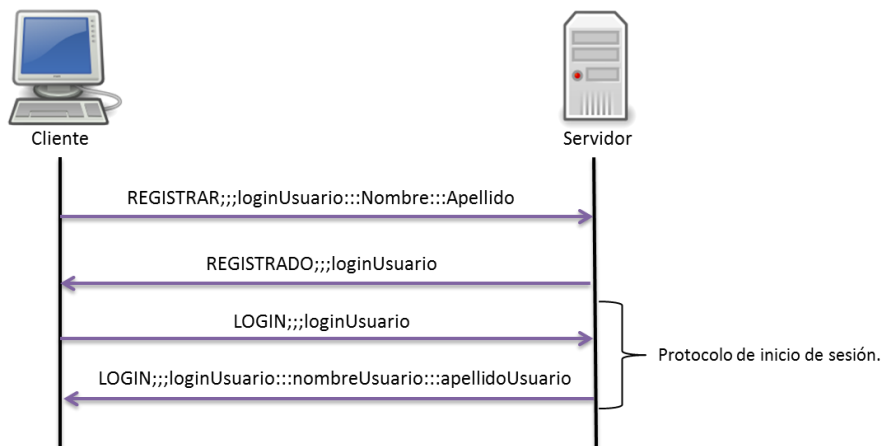


**Ilustración 16. Ejemplo del protocolo de inicio de sesión cuando ocurre un error**

La Ilustración 16 muestra el siguiente ejemplo: el usuario envía una petición de inicio de sesión con el comando LOGIN y login Pepito. El servidor no encuentra un usuario con este login, por lo tanto envía el mensaje de error con comando: ERROR y el mensaje “Usuario no existe”.

## Protocolo para registrar un usuario:

**Caso 1:** Se registra un usuario que no se encuentra en la base de datos del sistema.



**Ilustración 17. Protocolo de registro de un usuario.**

La Ilustración 17 muestra el protocolo de registro de un usuario, este consiste en que el programa cliente envía un mensaje al servidor con el comando REGISTRAR, seguido del login que el usuario desea tener, su nombre y su apellido. El servidor revisa en su base de datos si se encuentra un usuario con el login dado, si no existe, registra al usuario y envía un mensaje de confirmación con el comando REGISTRADO seguido del login. Cuando el cliente recibe el mensaje de confirmación, entonces procede a iniciar la sesión del usuario siguiendo el protocolo de inicio de sesión explicado en la Ilustración 14.

Las Tablas 6 y 7 explican la sintaxis de cada mensaje:

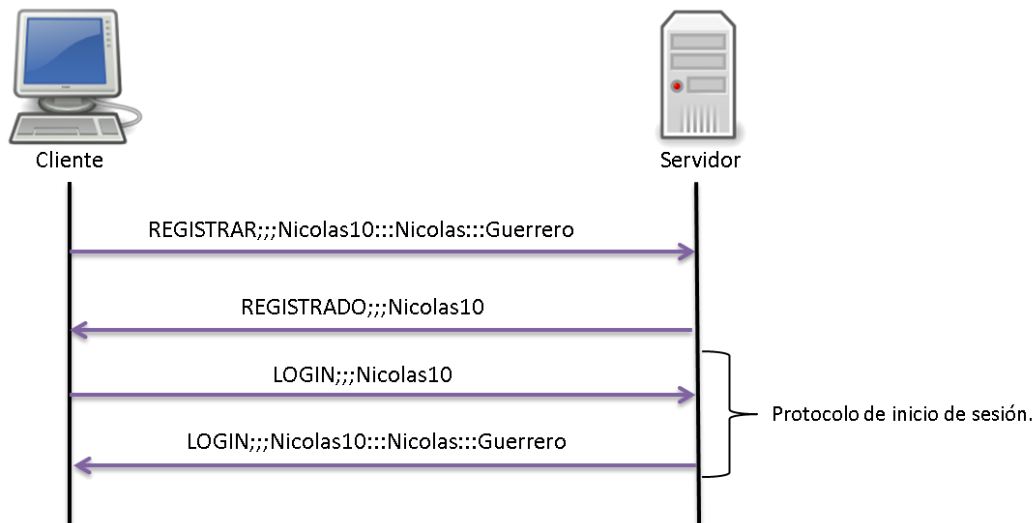
Comando Cliente	Parámetros	Descripción
REGISTRAR	LoginUsuario	Representa el login del usuario que desea registrarse.
	Nombre	Representa el nombre del usuario que desea registrarse.
	Apellido	Representa el apellido del usuario que desea registrarse.
LOGIN	LoginUsuario	Representa el login del usuario que acaba de registrarse y que va a iniciar sesión.

**Tabla 6. Sintaxis de los mensajes del cliente para el protocolo de registro de usuario.**

Comando Servidor	Parámetros	Descripción
REGISTRADO	LoginUsuario	Representa el login del usuario que acaba de registrarse.
LOGIN	LoginUsuario	Representa el login del usuario que va a iniciar sesión.
	NombreUsuario	Representa el nombre del usuario que va a iniciar sesión.
	ApellidoUsuario	Representa el apellido del usuario que va a iniciar sesión.

**Tabla 7. Sintaxis de los mensajes del servidor para el protocolo de registro de usuario.**

A continuación, un ejemplo concreto:



**Ilustración 18. Ejemplo del protocolo de registro de usuario.**

En el ejemplo que muestra la Ilustración 18, el usuario Nicolás Guerrero desea registrarse al sistema, por lo tanto el programa cliente envía el mensaje con comando REGISTRAR seguido del login “Nicolas10” y de los nombres y apellidos: Nicolás Guerrero. El servidor revisa en la base de datos si existe un usuario con login Nicolas10, si no lo encuentra, procede a agregar el usuario y envía el mensaje de confirmación con comando REGISTRADO y el login: Nicolas10. Acto seguido, el programa cliente prosigue a iniciar la sesión de Nicolas10 siguiendo el protocolo de inicio de sesión.

**Caso 2** Se registra un usuario que ya existe en la base de datos.



**Ilustración 19. Protocolo de registro de usuario en caso de error.**

La Ilustración 19 muestra el protocolo de registro de un usuario en caso de error, en dónde el programa cliente envía la solicitud de registro de usuario al servidor con un mensaje con comando REGISTRAR, seguido del login, nombre y apellido del usuario que se desea registrar. El servidor revisa en su base de datos por un usuario con el login dado, en caso de encontrarlo entonces envía un mensaje de error al programa cliente con comando ERROR seguido del mensaje de error.

Las Tablas 8 y 9 explican la sintaxis de cada mensaje:

Comando Cliente	Parámetros	Descripción
REGISTRAR	LoginUsuario	Representa el login del usuario que desea registrarse.
	Nombre	Representa el nombre del usuario que desea registrarse.
	Apellido	Representa el apellido del usuario que desea registrarse.

Tabla 8. Sintaxis del mensaje del cliente para el protocolo de registro de usuario.

Comando Servidor	Parámetros	Descripción
ERROR	MensajeDeError	Representa el mensaje de error que se quiere informar al jugador.

Tabla 9. Sintaxis del mensaje del servidor para el protocolo de registro de usuario en caso de error.

A continuación un ejemplo concreto:

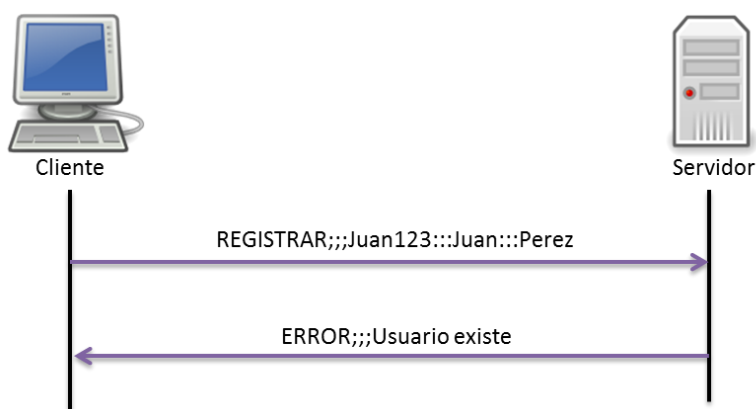
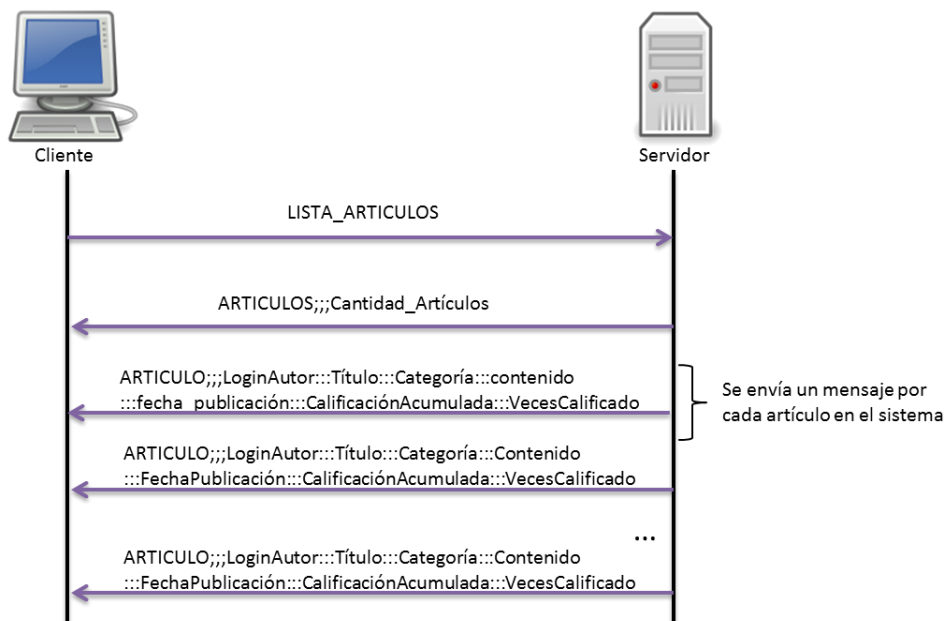


Ilustración 20. Ejemplo del protocolo de registro de un usuario en caso de error.

En la Ilustración 20 se muestra el ejemplo en el que el usuario Juan Perez (que ya existe en el sistema) desea registrarse. El programa cliente envía la solicitud al servidor con el mensaje con comando REGISTRAR seguido del login, nombre y apellido del usuario: Juan123, Juan y Perez respectivamente. El servidor al revisar la base de datos encuentra un usuario con este login, por lo tanto envía al programa cliente el mensaje de error.

## Protocolo para solicitar todos los artículos del sistema:





**Ilustración 21. Protocolo de solicitud de todos los artículos.**

La Ilustración 21 muestra el protocolo de solicitud de todos los artículos del sistema, que consisten en que el programa cliente envía un mensaje con el comando LISTA\_ARTICULOS y el servidor responde de la siguiente manera:

1. Envía un mensaje con comando ARTICULOS seguido de la cantidad de artículos que hay en el sistema.
2. Por cada artículo en el sistema envía un mensaje con comando ARTICULO, seguido del login del autor del artículo, el título del artículo, la categoría, contenido, fecha de publicación, calificación acumulada y las veces que el artículo ha sido calificado.

Las tablas 10 y 11 explican la sintaxis de cada mensaje:

Comando Cliente	Parámetros	Descripción
LISTA_ARTICULOS	Ninguno	Solamente envía el mensaje con el comando LISTA_ARTICULOS para solicitar todos los artículos del sistema.

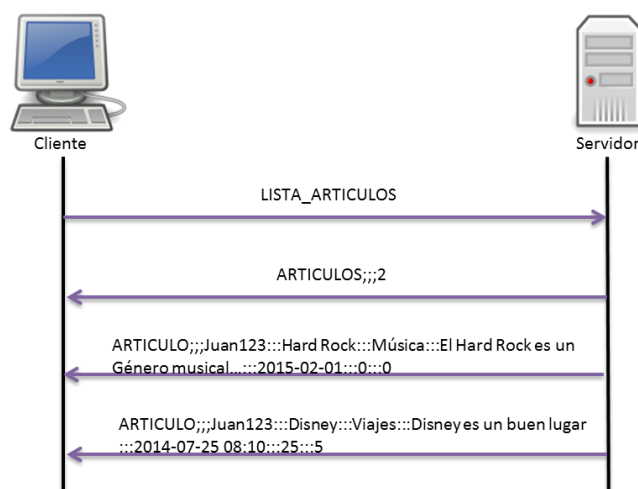
**Tabla 10. Sintaxis del mensaje del cliente para el protocolo de solicitud de todos los artículos.**

Comando Servidor	Parámetros	Descripción
ARTICULOS	CantidadArtículos	Representa la cantidad total de artículos en el sistema.
ARTICULO	LoginAutor	Representa el login del autor del artículo.
	Título	Representa el título del artículo.
	Categoría	Representa la categoría del artículo.
	Contenido	Representa el contenido del artículo.
	FechaPublicación	Representa la fecha de publicación del artículo. Esta está en formato: "yyyy-MM-dd kk:mm".

	CalificaciónAcumulada	Representa la calificación acumulada del artículo hasta el momento.
	VecesCalificado	Representa la cantidad de veces que el artículo ha sido calificado hasta el momento.

**Tabla 11. Sintaxis de los mensajes del servidor para el protocolo de solicitud de todos los artículos.**

A continuación un ejemplo concreto.

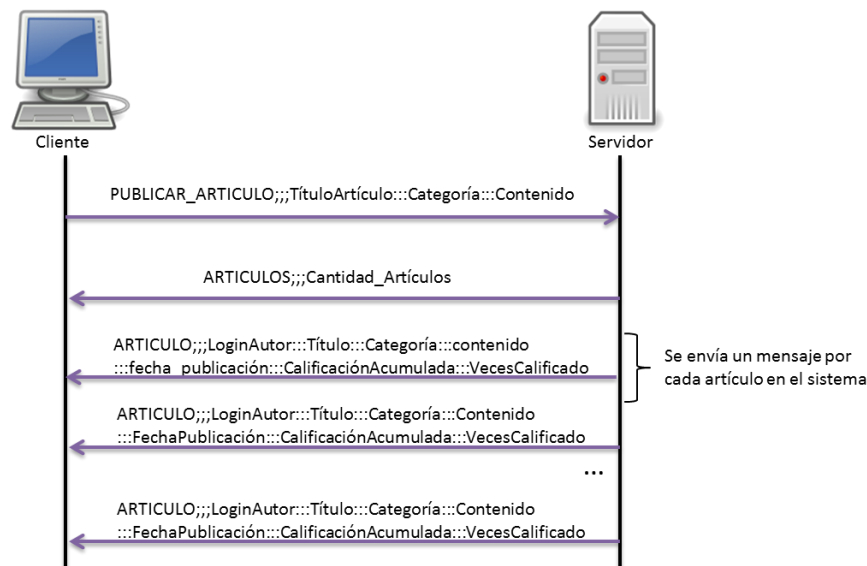


**Ilustración 22. Ejemplo del protocolo de solicitud de todos los artículos.**

En la Ilustración 22 se muestra el ejemplo en el que el programa cliente le solicita al servidor la lista de artículos con el mensaje con comando LISTA\_ARTICULOS. El servidor responde con los siguientes mensajes:

1. El mensaje con comando ARTICULOS seguido del número 2, que indica que hay dos artículos en el sistema.
2. Dos mensajes con comando ARTICULO, seguido del login de su autor dueño, título, categoría, contenido, fecha de publicación, calificación acumulada y la cantidad de veces que ha sido calificado.

### Protocolo para publicar un artículo:



**Ilustración 23. Protocolo para la publicación de un artículo.**

La Ilustración 23 muestra el protocolo de publicación de un artículo, en donde el programa cliente envía el servidor un mensaje con comando PUBLICAR\_ARTICULO, seguido del título del artículo, la categoría y el contenido del artículo. Cuando el servidor recibe el mensaje, registra el artículo a la base de datos con los datos de título, categoría y contenido dados por el cliente. En cuanto a los otros datos, el servidor asume que:

1. El autor del artículo es quien inició sesión.
2. La fecha de publicación es la fecha en la cual recibió la solicitud de publicación.
3. Tanto la calificación acumulada como la cantidad de veces que ha sido calificado están en 0.

Una vez ha registrado el artículo, retorna los siguientes mensajes al cliente:

1. Un mensaje con comando ARTÍCULOS, seguido de la cantidad de artículos que hay en el sistema, incluyendo el artículo que se publicó.
2. Un mensaje por cada artículo que hay en el sistema con comando ARTICULO, seguido del login del autor del artículo, el título, la categoría, el contenido, la fecha de publicación, la calificación acumulada y la cantidad de veces que ha sido calificado.

Las tablas 12 y 13 explican la sintaxis de cada mensaje.

Comando Cliente	Parámetros	Descripción
PUBLICAR_ARTICULO	TítuloArtículo	Representa el título del artículo que se va a publicar.
	Categoría	Representa la categoría del artículo que se va a publicar.
	Contenido	Representa el contenido del artículo que se va a publicar.

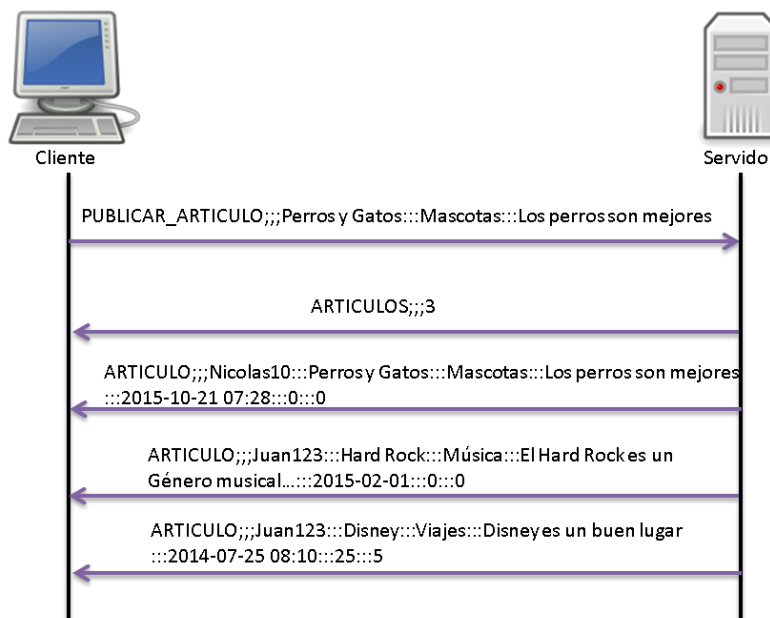
**Tabla 12. Sintaxis del mensaje del cliente para el protocolo de publicación de un artículo.**

Comando Servidor	Parámetros	Descripción
ARTICULOS	CantidadArtículos	Representa la cantidad total de artículos en el sistema.

ARTICULO	LoginAutor	Representa el login del autor del artículo.
	Título	Representa el título del artículo.
	Categoría	Representa la categoría del artículo.
	Contenido	Representa el contenido del artículo.
	FechaPublicación	Representa la fecha de publicación del artículo. Esta está en formato: "yyyy-MM-dd kk:mm".
	CalificaciónAcumulada	Representa la calificación acumulada del artículo hasta el momento.
	VecesCalificado	Representa la cantidad veces que el artículo ha sido calificado hasta el momento.

**Tabla 13. Sintaxis de los mensajes del servidor para el protocolo de publicación de un artículo.**

A continuación un ejemplo concreto:



**Ilustración 24. Ejemplo del protocolo de publicación del artículo.**

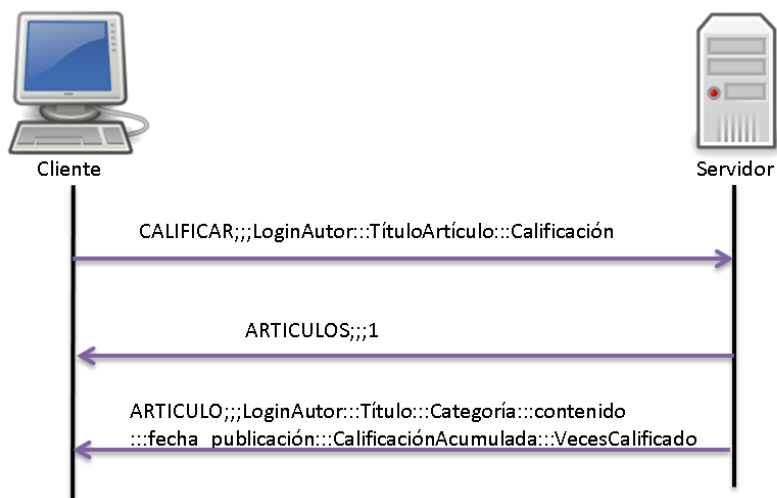
La Ilustración 24 presenta el caso en el que el usuario Nicolas10 desea publicar un artículo. El programa cliente envía el mensaje al servidor con el mensaje con comando PUBLICAR\_ARTICULO, título "Perros y Gatos", categoría "Mascotas" y con contenido "Los perros son mejores". El servidor, al recibir el mensaje, asume que el autor es Nicolas10 (porque guardó su referencia cuando Nicola10 inició sesión), coloca como fecha de publicación, la fecha y hora en la que el mensaje llegó al servidor, asume que la calificación acumulada y la cantidad de veces que ha sido calificado es cero y registra el artículo.

Una vez el servidor registró el nuevo artículo, envía varios mensajes al cliente para retornarle cada uno de los artículos del sistema, incluyendo el que se acabó de publicar:

1. El primer mensaje, con comando ARTICULOS, seguido del número 3, que indica que ahora hay 3 artículos en el sistema.

- Envía tres mensajes con comando ARTICULO, cada uno con uno de los artículos que se han publicado. Vale la pena aclarar que los mensajes se envían en orden de fecha de publicación, del más reciente al más antiguo.

### Protocolo para calificar un artículo:



**Ilustración 25. Protocolo de calificación de un artículo.**

La Ilustración 25 muestra el protocolo de calificación de un artículo que consiste en que el programa cliente envía un mensaje con comando CALIFICAR, seguido del login del usuario autor del artículo que se va a calificar, el título del artículo que se va a calificar y la calificación del artículo. El servidor al recibir la solicitud actualiza en la base de datos la calificación del artículo, y la cantidad de veces que ha sido calificado. Después, retorna al cliente el artículo con la calificación actualizada mediante los siguientes mensajes:

- El primer mensaje con el comando ARTICULOS, seguido del número 1, que indica que solo se va a enviar un artículo (el que se calificó).
- Otro mensaje con el comando ARTICULO, seguido de la información del artículo.

Las tablas 14 y 15 explican la sintaxis de cada mensaje.

Comando Cliente	Parámetros	Descripción
CALIFICAR	LoginAutor	Representa el login del usuario autor del artículo que se va a calificar.
	TítuloArtículo	Representa el título del artículo que se va a calificar.
	Calificación	Representa la calificación que se le va a dar al artículo.

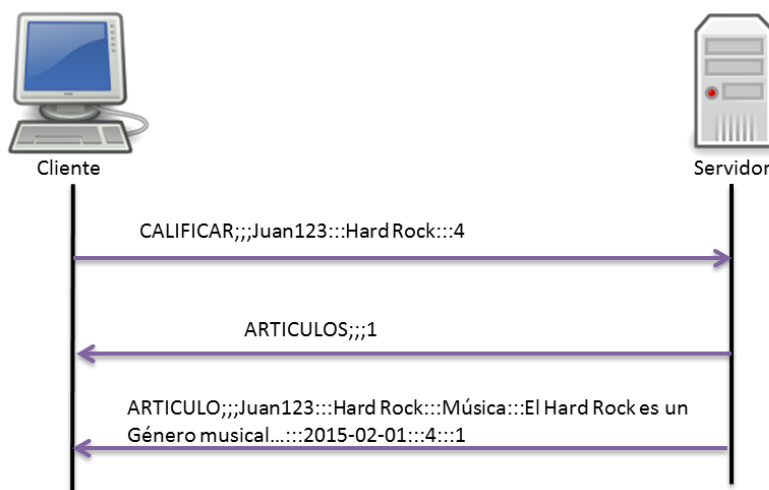
**Tabla 14. Sintaxis del mensaje del cliente para el protocolo de calificación de un artículo.**

Comando Servidor	Parámetros	Descripción
ARTICULOS	1	Solamente se va a enviar el artículo que ha sido calificado.
ARTICULO	LoginAutor	Representa el login del autor del artículo.

	Título	Representa el título del artículo.
	Categoría	Representa la categoría del artículo.
	Contenido	Representa el contenido del artículo.
	FechaPublicación	Representa la fecha de publicación del artículo. Esta está en formato: "yyyy-MM-dd kk:mm".
	CalificaciónAcumulada	Representa la calificación acumulada del artículo hasta el momento.
	VecesCalificado	Representa las veces que el artículo ha sido calificado hasta el momento.

**Tabla 15. Sintaxis de los mensajes del servidor para el protocolo de calificación de un artículo.**

A continuación un ejemplo concreto:



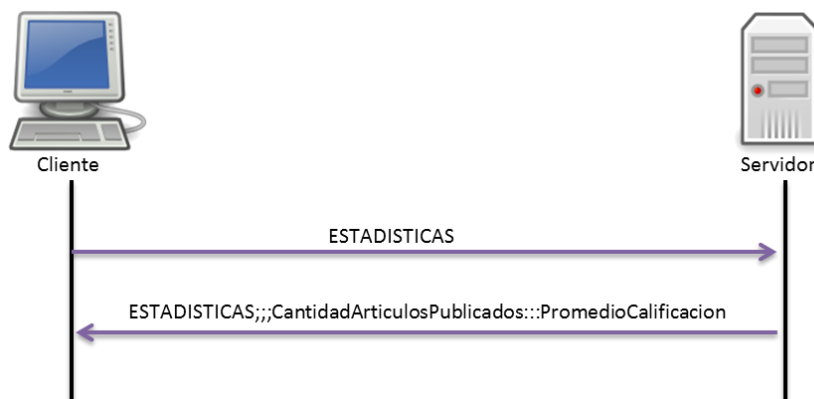
**Ilustración 26. Ejemplo del protocolo de calificación de un mensaje.**

En la Ilustración 26 se muestra el ejemplo en el cual el usuario Nicolas10 desea calificar un artículo del usuario Juan123. El programa cliente envía un mensaje al servidor con el comando CALIFICAR, seguido del login de Juan (Juan123), seguido del título del artículo que se quiere calificar (Hard Rock) y la calificación.

El servidor actualiza la información en la base de datos y le envía al usuario los siguientes dos mensajes:

1. Mensaje con el comando ARTICULOS y el número 1, indicándole al usuario que se va a enviar 1 artículo.
2. Otro mensaje con el comando ARTICULO, seguido de los datos del artículo que se acaba de calificar.

### **Protocolo para solicitar las estadísticas de publicación de un usuario:**



**Ilustración 27. Protocolo de solicitud de estadísticas.**

La Ilustración 27 muestra el protocolo de solicitud de estadísticas, en donde el programa cliente le envía al servidor un mensaje con comando ESTADISTICAS y sin parámetros. El servidor retorna un mensaje con comando ESTADISTICAS, seguido de la cantidad de artículos que el usuario ha publicado y el promedio de calificación de todos los artículos del usuario.

Las tablas 16 y 17 explican la sintaxis de cada mensaje.

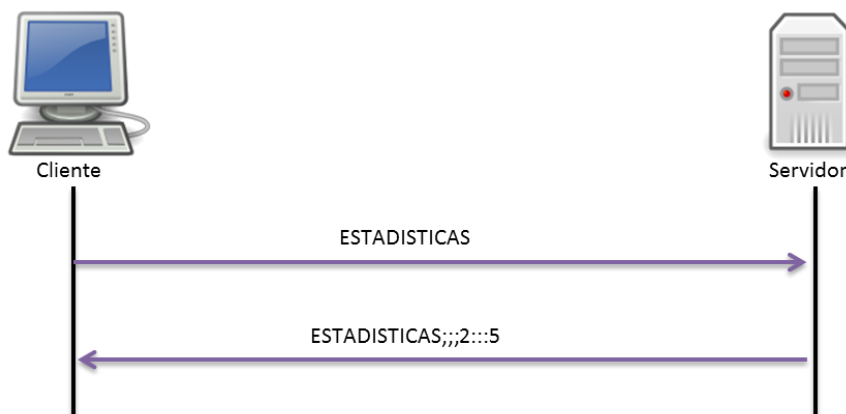
Comando Cliente	Parámetros	Descripción
ESTADISTICAS	Ninguno	Solamente envía el mensaje con el comando ESTADISTICAS para solicitar las estadísticas de publicación del usuario.

**Tabla 16. Sintaxis del mensaje del cliente para el protocolo de solicitud de estadísticas.**

Comando Servidor	Parámetros	Descripción
ESTADISTICAS	CantidadArticulosPublicados	Representa la cantidad de artículos que el usuario ha publicado.
	PromedioCalificación	Representa el promedio de calificación de todos los artículos del usuario.

**Tabla 17. Sintaxis del mensaje del servidor para el protocolo de solicitud de estadísticas.**

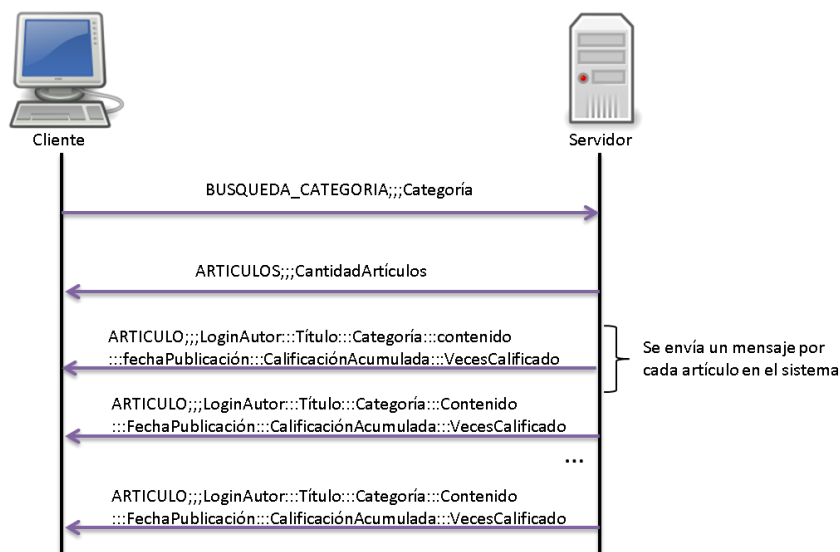
A continuación, un ejemplo concreto:



**Ilustración 28. Ejemplo del protocolo de solicitud de estadísticas.**

En este ejemplo, el usuario Juan123 desea conocer sus estadísticas de publicación. El programa cliente le envía un mensaje al servidor con el comando ESTADISTICAS. El servidor le retorna un mensaje con el comando ESTADISTICAS, seguido del número 2, que indica que Juan123 ha publicado 2 artículos, y su promedio de calificación por esos dos artículos, que en este caso es 5.

### Protocolo para solicitar la lista de artículos de una categoría.



**Ilustración 29. Protocolo de búsqueda de artículos de una categoría.**

La Ilustración 29 muestra el protocolo de búsqueda de artículos dada una categoría. En este protocolo el programa cliente le envía al servidor un mensaje con comando BUSQUEDA\_CATEGORIA, seguido de la categoría que se desea buscar. El servidor busca en la base de datos los artículos que pertenecen a esa categoría y los envía al cliente de la siguiente manera:

1. Envía un mensaje con comando ARTICULOS, seguido de la cantidad de artículos que son parte de la categoría buscada.



2. Por cada mensaje encontrado se envía un mensaje con comando ARTICULO, seguido de la información del artículo.

Las tablas 18 y 19 explican la sintaxis de los mensajes:

Comando Cliente	Parámetros	Descripción
BUSQUEDA_CATEGORIA	Categoría	Representa la categoría de los artículos que se desea buscar.

Tabla 18. Sintaxis del mensaje del cliente para el protocolo de búsqueda de artículos de una categoría.

Comando Servidor	Parámetros	Descripción
ARTICULOS	CantidadArtículos	Representa la cantidad total de artículos en el sistema.
ARTICULO	LoginAutor	Representa el login del autor del artículo.
	Título	Representa el título del artículo.
	Categoría	Representa la categoría del artículo.
	Contenido	Representa el contenido del artículo.
	FechaPublicación	Representa la fecha de publicación del artículo. Esta está en formato: "yyyy-MM-dd kk:mm".
	CalificaciónAcumulada	Representa la calificación acumulada del artículo hasta el momento.
	VecesCalificado	Representa la cantidad de veces que el artículo ha sido calificado hasta el momento.

Tabla 19. Sintaxis de los mensajes del servidor para el protocolo de búsqueda de artículos de una categoría.

A continuación, un ejemplo concreto:

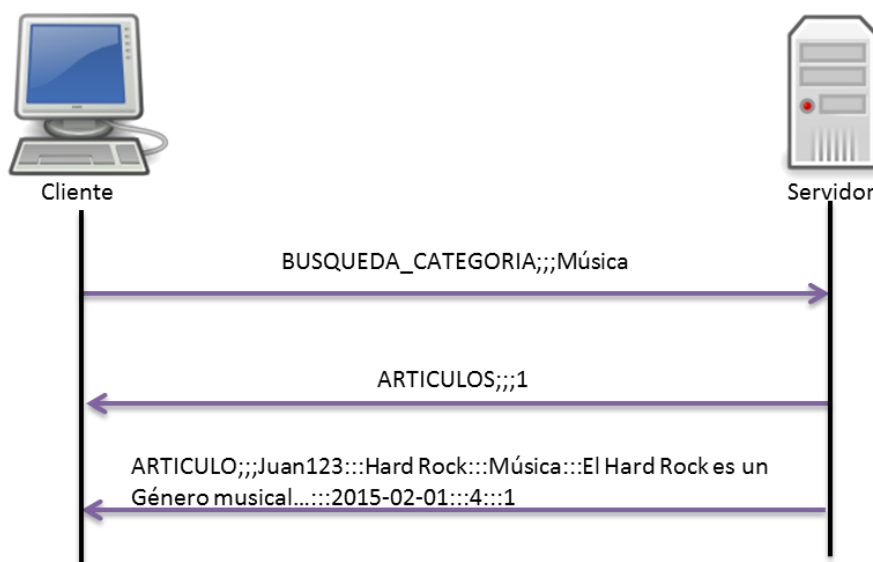
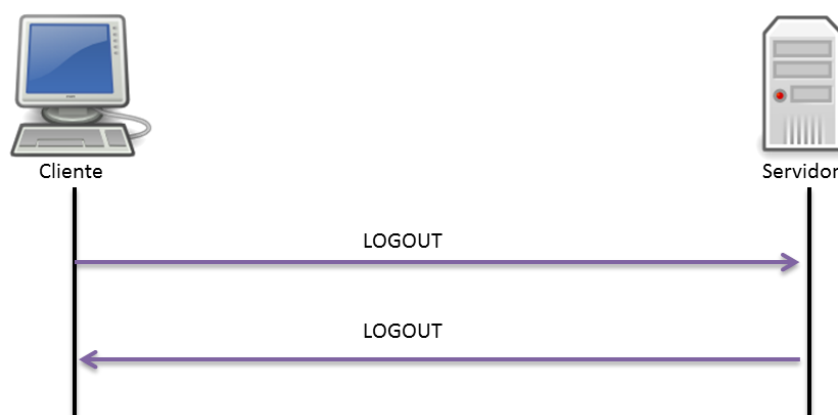


Ilustración 30. Ejemplo del protocolo de búsqueda de artículos de una categoría.

La Ilustración 30 muestra el ejemplo en el que el usuario desea conocer la lista de los artículos que pertenecen a la categoría de música. El programa cliente le envía al servidor un mensaje con el comando BUSQUEDA\_CATEGORIA, seguido de la categoría que se desea buscar: Música. El servidor busca en la base de datos, encuentra un artículo de esta categoría y le envía al cliente los siguientes mensajes:

1. Un mensaje con el comando ARTICULOS y el número 1, indicando que solamente hay un artículo en la categoría Música.
2. Un mensaje con el comando ARTICULO y los datos del artículo con categoría música.

### Protocolo para cerrar sesión:



**Ilustración 31. Protocolo de cierre de sesión.**

El protocolo de cierre de sesión (Ilustración 31) ocurre cuando el usuario cierra la aplicación cliente y se manda un mensaje al servidor con comando LOGOUT. El servidor elimina la referencia del usuario y devuelve otro mensaje al cliente con comando LOGOUT para confirmar el cierre de sesión. El servidor cierra los canales de comunicación con el cliente.

#### **Nota general:**

**Si hay un error en el protocolo de comunicación o en la consulta hacia la base de datos, el servidor debe enviar un mensaje del error al cliente con el comando ERROR, seguido del mensaje de error.**