

## **Tarea 1 de Programación Comunicación de Datos y Redes**

**Entrega solo a través de Adecca, fecha: lunes 22 de Mayo 23:59**

**Semestre 2017/1**

**TAREA es INDIVIDUAL**

### **Objetivos:**

- Alumno aprenda a usar las librerías de java dispuestas para establecer transferencia de datos con protocolos de la capa de transporte de la Internet.
- Alumno diseñe un protocolo de capa de aplicación sencillo
- Alumno desarrolle programa cliente/servidor que implemente su protocolo de comunicación de capa de aplicación

### **Problema:**

La empresa Sofanor Software Inc desea que usted le diseñe un protocolo de comunicación de capa de aplicación que permita implementar una aplicación de red de almacenamiento y consulta de breves mensajes de texto, cuya longitud no exceda los 30 bytes.

Para acceder a este servicio el usuario inicialmente debe proporcionar un seudónimo y una palabra secreta que sea reconocida por el servidor. En el caso de un nuevo usuario, este debe solo proporcionar un seudónimo, el que cual debe ser inmediatamente aceptado si no estuviese registrado en el sistema. En caso contrario, debe el usuario proporcionar un nuevo nombre. Luego de aceptado un nuevo seudónimo, el usuario debe enviar la palabra secreta que será asociada a su seudónimo.

Solo un usuario validado o recién registrado puede acceder a los siguientes servicios que ofrece el servidor:

- Consultar la hora y día actual
- Consultar la hora u día de la última consulta
- Listar los seudónimos de los usuarios registrados
- Guardar un mensaje para un usuario seleccionado
- Consultar, si hay algún mensaje guardado a su nombre (seudónimo), por ende este debe ser transmitido. Si hay varios, cada uno de ellos debe ser transmitido al usuario. En caso que no existan mensajes, este hecho se debe notificar explícitamente al cliente.
- Borrar registro del usuario. Implica borrar su seudónimo y sus mensajes.
- Finalizar sesión de consulta

Considere que no hay restricción del número de mensajes a guardar para un usuario, pero el usuario podría limitar el número de mensajes a descargar. Cuando un mensaje ha sido descargado, este puede ser por tanto eliminado del servidor. Un usuario solo puede acceder a sus mensajes.

En su diseño debe explicitar los pasos que deben ejecutar tanto el cliente como el servidor y tipos de mensajes intercambiados. Estos pasos deben ser indicados tanto al ejecutar un paso de forma exitosa

como también en aquellos que surja un error, por ejemplo cuando un seudónimo existe o la palabra clave sea errónea (en tal caso se debe cerrar el servicio).

Luego debe implementar su protocolo mediante el desarrollo de programas cliente y servidor en lenguaje de programación JAVA. Para el cliente no es obligación que desarrolle alguna interfaz grafica, bastaría con capturar los datos del cliente vía terminal.

Puede implementar su protocolo de capa de aplicación ya sea sobre UDP o TCP. Comente brevemente en su informe el por qué de su elección.

### **Comentarios:**

Es obligatorio que su protocolo siga la siguiente estructura:

- Preámbulo: Autenticación del pseudónimo/palabra secreta o registro del pseudónimo.
- Consulta: permitir el envío de una o más consultas al servidor
- Termina: Cierre de la sesión.

### **Solicitud:**

- Debe escribir un informe donde detalle el protocolo diseñado ( no más de 2 páginas)
- Incluya códigos fuentes debidamente comentados de sus programas cliente/servidor
- Solo emplee como base los programas java vistos en clases a excepción del código que realice para confeccionar la interfaz grafica del usuario.
- Captura de pantalla de al menos un paquete que incluya como contenido un tipo de mensaje enviado por su cliente mediante el software Wireshark. Este sniffer de redes puede ser descargado desde:  
<https://www.wireshark.org/#download>
- Tutorial básico de Wireshark: [https://www.youtube.com/watch?v=vh6q7K\\_2FHW](https://www.youtube.com/watch?v=vh6q7K_2FHW)
- Envío tarea: solo a través de Adecca, adjunte código fuente, informe, archivo README (que explique claramente como compilar y ejecutar tarea) en un solo archivo del tipo tarea\_1\_2017\_1\_redes\_apellido\_alumno\_nombre\_alumno.zip
- El profesor no posee instalado netbeans ni eclipse, de modo que aunque desarrolle su tarea en algún IDE, indique (en archivo README) como compilar y ejecutar su programa por consola/terminal. Se descontaran puntos por el no cumplimiento. El profesor hará el mejor esfuerzo de compilar su tarea pero no la corregirá. Si este paso fallase después de un par de intentos, su tarea será evaluada con la nota mínima.
- LA TAREA ES INDIVIDUAL. Ante la detección de copia, queda a discreción del profesor si está será evaluada con nota 1 o se requerirá de una demo explicativa presencial de su tarea por parte del alumno afectado.
- DEBE INSTALAR (COMPILAR Y EJECUTAR) programa servidor en el servidor de alumnos de ICI/IECI. Detalles serán entregados en clases y publicados prontamente. Emplee un número de puerto superior a 1024 para su servidor. Configure correctamente su cliente.