

# Laboratorio 2: Redes de Computadores

**Profesor:** Jorge Díaz

**Ayudantes:** Juan Cucurella, Nicolás Rodríguez

Agosto 2025

## 1 Objetivos del laboratorio:

- Entender como se comportan las arquitecturas cliente-servidor
- Entender de manera práctica el comportamiento de la capa de aplicación

## 2 Laboratorio 2

### 2.1 Enunciado

En un mundo dominado por la irreverencia y la poca seriedad en la vida, una empresa busca de forma desesperada guardar la información más importante para ellos: frases filosóficas. Para esto lo contratan a usted y a su grupo con el fin de poder guardar esas frases, ya que con su expertiz en la conexión mediante el uso de sockets, pueden establecer un paso de mensajes confiable.

¿Podrá usted y su quipo cumplir con esta misión?, ¿caerá lo máspreciado por esta empresa en el olvido?, cumpla con la misión y recibirá una buena recompensa.

### 2.2 Explicación general

El laboratorio se divide en 3 grandes secciones, en todas y cada una de ellas debe programar clientes que a través del uso de sockets cumplen un rol específico en la interacción del laboratorio. En primera instancia debe programar un cliente TCP, en segunda un cliente UDP, en tercera y última instancia del laboratorio reutilizar o programar un nuevo cliente TCP que sea capaz de enviar solicitudes HTTP.

### 2.3 Pre-laboratorio

- Contar con Python instalado en su sistema.
- Instalar las librerías clave para trabajar el laboratorio (*Consulte sección "Librerías permitidas"*)
- Conocer la dirección y los puertos de los diferentes servicios, los cuales serán dados el día del laboratorio.

### 2.4 Inicio: Cliente TCP

Se debe programar un cliente TCP capaz de contactarse con el servidor tomando en cuenta los siguientes requisitos:

- El cliente debe ser capaz de recibir, leer e imprimir correctamente la información que retorne el servidor.
- El servidor enviará información en respuesta a tres comandos **GET**, **EXIT**, **JOKE**, el cliente debe permitir escribir por entrada estándar el comando que se requiera.
- Permitir durante la interacción, escribir al menos una vez cada comando.

La labor del cliente TCP es obtener los puertos de conexión de los servidores UDP y HTTP.

## 2.5 Búsqueda de la frase real: Cliente UDP

Utilizando la información obtenida a partir de TCP, debe programar un cliente UDP capaz de conectarse al servidor usando el puerto indicado a través de la interacción anterior; este cliente debe cumplir al menos con lo siguiente:

- Permitir el envío del contenido obtenido del comando **JOKE**.
- Esperar la respuesta del servidor a la información anterior e imprimir esta por pantalla.

La labor del cliente UDP es obtener una de las frases filosóficas que se quiere enviar al servidor HTTP.

## 2.6 Registro de la frase: Cliente TCP-HTTP

Utilizando la información recopilada, debe programar un cliente TCP que permita realizar solicitudes HTTP al servidor conectándose a un servicio web expuesto en el laboratorio:

- Mediante el uso de sockets TCP, el cliente debe enviar una solicitud **HTTP POST** al endpoint `"/frase/"` del servicio expuesto.
- La solicitud debe contener la frase obtenida por UDP, y un identificador de grupo.

*Nota: Se recomienda ver la solicitud como un API Request.*

*Nota 2: Con identificador de grupo nos referimos ya sea a sus nombres o rol-usm dentro de la data enviada en la solicitud POST.*

## 2.7 Informe de laboratorio

- ¿Qué es el protocolo HTTP y para que se usa? ¿Por qué es posible realizar solicitudes HTTP a través de un socket TCP? fundamente. En caso de utilizar fuentes externas de información, especifíquelas.
- Realice un diagrama o dibujo de la interacción del laboratorio. Incluya en dicho diagrama la dirección del servidor, los puertos utilizados para conectarse a los servicios (en caso de utilizar más de un puerto) y los protocolos y servicios utilizados.
- ¿Cuál fue la información que obtuvieron de los servidores? Muestre evidencia de esto (screenshots, archivos de texto escritos luego de la interacción con el servidor, etc)
- Nombre otros protocolos de la capa de aplicación y compárelos con HTTP, es decir que puertos usan generalmente, tipo de socket utilizado, y para que sirve el protocolo (nombre al menos 5 que no sean HTTP).

## 3 Librerías permitidas

Se detallan las librerías permitidas para la presente experiencia de laboratorio, en caso de requerir el uso de otra librería que no esté presente en este listado, debe consultarla durante la experiencia.

### 3.1 Python

- **Socket**
- **Asyncio**
- **os**
- **re** (expresiones regulares, para patrones de texto)
- **JSON** (Solo en caso de que lo necesite para interpretar información en el servidor HTTP)

## 4 Reglas de entrega

- El laboratorio se realiza en tríos seleccionados en Aula.
- La fecha de entrega es el día **29 de Agosto** a las 23:59 horas para los grupos del día Lunes y **2 de Septiembre** a las 23:59 hora para los grupos del día Viernes.
- La entrega debe realizarse a través de Aula, en un archivo comprimido **.zip**, indicando el número de Laboratorio y grupo en el siguiente formato: L2-Grupo[Nº Grupo].zip, Ejemplo: **L2-Grupo01.zip**.
- Debe entregar todos los archivos fuente necesarios y solicitados en el presente enunciado para la correcta ejecución de la entrega.
- Debe entregar un README con nombre y rol de cada integrante del grupo, además de las instrucciones necesarias para ejecutar correctamente el laboratorio. (ADVERTENCIA: Si no se entrega dicha información, se colocará un cero a la entrega y posteriormente se tendrá que coordinar una sesión de apelación.)
- Cada hora de retraso penalizará el laboratorio, descontando 15 ptos.
- Cualquier sospecha de copia será notificada debidamente al profesor y evaluada con nota 0. Siendo tomado en cuenta también cualquier copia directa de algún sitio web o foro. Se tendrá un software a mano para realizar dichas comparaciones.
- Cualquier sospecha de uso de herramientas de inteligencia artificial se tendrá en consideración para la evaluación de su entrega, lo que podría conllevar descuentos posteriores.
- Errores de formato involucran una penalización de 15 puntos a la entrega.
- Errores en el nombre de archivo de la entrega involucran una penalización de 15 puntos a la entrega.
- Errores en los roles USM de alguno de los estudiantes del grupo conllevan un descuento de 5 puntos a la entrega.
- Faltas graves de ortografía en el informe de Laboratorio implican penalización de 5 a 15 puntos a la entrega dependiendo de la cantidad de faltas encontradas.
- No especificar fuentes de información o bibliografía conlleva una penalización de 10 puntos a la entrega.
- En ciertos casos de sospecha de actividad maliciosa, copia, uso de IA u otras conductas, es posible que se cite al grupo a una sesión de defensa, en cuyo caso los ayudantes de laboratorio se contactarán vía correo con los grupos correspondientes. En caso de ser citado a defensa y no asistir, se aplicarán las medidas correspondientes y el laboratorio será evaluado con nota 0.