

## ETS: Aplicaciones para Comunicaciones en Red

### Proyecto práctico, aplicación distribuida para búsqueda de archivos

#### Indicaciones:

- El proyecto práctico corresponde al 100% de la calificación del examen ETS.
- El día del examen únicamente se revisará el funcionamiento del proyecto y en ese momento se les dará su calificación.
- Se enviará al correo de enlace que enviaron al coordinador del examen la liga y clave de acceso para la sesión de zoom donde se evaluará de forma presencial el proyecto.
- Pasados 10 minutos de la hora del examen, se revisará el proyecto llamando a los alumnos en orden de lista, cuando se recorra la lista completa, se volverá a recorrer y así, hasta que no existan alumnos a calificar. En ese momento terminará la evaluación y la sesión de zoom.
- Los alumnos permanecerán en la sala de espera de la aplicación y se irá revisando uno por uno. Si en el momento de revisión el alumno no está en su cuenta se continuará con el siguiente alumno y se respetará el orden mencionado de revisión.
- Es responsabilidad del alumno conectarse con la infraestructura suficiente para correr el proyecto, compartir pantalla, el audio y video para conferencia de forma simultánea.
- El examen se revisará por partes, y el funcionamiento correcto de alguna de ellas condiciona el funcionamiento del resto. En caso de que alguna de las partes indicadas más adelante no funcione, se terminará la revisión y la calificación se calculará únicamente con las partes revisadas previamente.

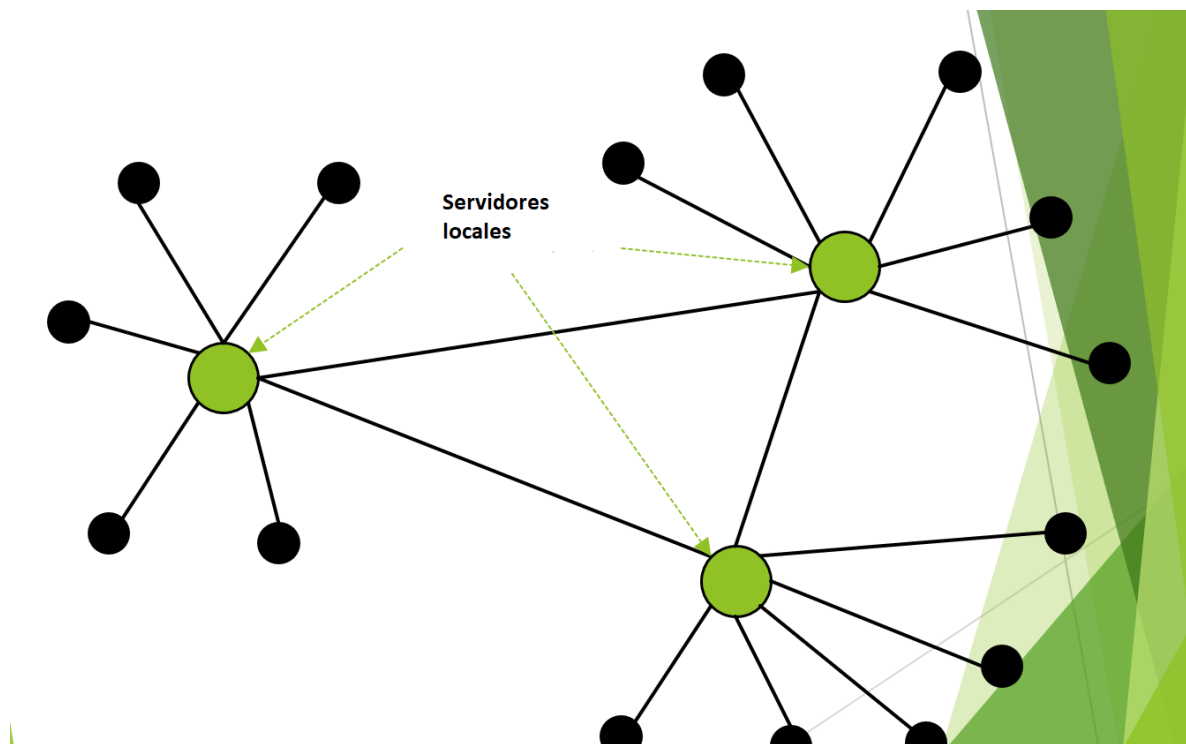
## Objetivo:

Implementar una aplicación en lenguaje orientado a objetos para la búsqueda de archivos en una topología de servidores locales mediante el uso de llamadas a procedimientos remotos, sockets de flujo y multicas, con la ayuda de hilos.

## Instrucciones:

Se deberá de implementar una aplicación con arquitectura cliente-servidor para la consulta y descarga de archivos.

Se creará una red a nivel de capa de aplicación con una topología de múltiples servidores locales, donde cada nodo tendrá interacción únicamente con su servidor de área y donde los servidores interactuaran entre ellos.



La aplicación deberá permitir las búsquedas en todos los nodos mediante los servidores locales, cada nodo solo interactuará con el servidor local que le corresponde. Los servidores locales solo contendrán tablas ordenadas con índices donde se incluirá la información de todos los archivos que comparten los nodos, (el método de control de actualizaciones queda a consideración de los alumnos).

Cada que un nuevo nodo se conecta, le enviara al servidor local que le corresponde los nombres y MD5 de los archivos que comparte, para que, a su vez, este comparta la información a todos los servidores locales de la red, incluyendo la información sobre

el nodo que los ofrece. También enviarán constantemente información sobre los nodos que aún son capaces de recibir para conexión.

En caso de que algún servidor local se desconecte, el resto de los servidores locales esperarán 30 segundos, en caso de no detectar actividad del servidor local, se darán de baja todos los recursos compartidos por dicho servidor local, así como los nodos ligados intentarán conectarse a algún otro servidor local, para seguir compartiendo sus recursos desde otro servidor local.

En caso de que algún nodo se desconecte, el servidor local ligado, esperará 15 segundos y si no hay actividad, dará de baja la información de los archivos que este nodo compartía y le avisará al resto de los servidores locales para que eliminen esos recursos de sus respectivas tablas.

Si algún nodo solicita un archivo y lo encuentra en la lista del servidor local que le corresponde, recuperará la información del identificador o identificadores de los nodos que tienen el archivo.

Una vez obtenida la información, el nodo que solicito el recurso se encargara de recuperarlo desde cada uno de los nodos directamente, es decir, si el archivo se encontró en 3 nodos, le pedirá una tercera parte del archivo a cada nodo al mismo tiempo para recuperar el archivo original.

La confirmación de que se trata del mismo archivo se hará validando el archivo con el algoritmo MD5, si hay más de un archivo con el mismo nombre, pero diferente MD5, le informará al usuario para que pueda elegir el de su preferencia.

## Funcionamiento

Se tendrán dos elementos diferentes, por un lado, se tendrá el programa de los servidores locales y por otro lado el programa de nodos

### Servidor local

Al levantarse un servidor local, preguntará por un número de nodo y obtendrá un identificador único combinando la dirección IP y el número de puerto.

Posteriormente levantará una interfaz gráfica, con el nombre de venta “Servidor local” más su identificador. Contendrá:

- Una lista donde se mostrarán todos los servidores locales activos en la topología, con su ID y el tiempo que tienen sin recibir un refresco de su estado activo.
- Una lista donde se mostrarán todos los nodos que tienen directamente conectados con su ID y el tiempo que tienen sin recibir un refresco de su estado activo.

- Una lista con todos los archivos que se están compartiendo en la topología, su MD5 y su ubicación.

Estás listas deberán de actualizarse de forma dinámica conforme se conecten y desconecten nodos y servidores locales.

Cada servidor local, estará lanzando datagramas al canal multicas 228.1.1.10 cada 5 segundos, donde informaran el número de nodos que aún pueden recibir para conexión, y por ese mismo canal, deberán de escuchar el resto de los servidores locales existentes.

Mediante consultas a llamadas de procedimiento remoto, actualizarán cada 30 segundos los recursos que comparten el resto de los elementos de la red y responderán a consultas realizadas solo por sus nodos conectados directamente.

Cada servidor local, solo permitirá la conexión de 2 nodos, en caso de que reciba otra conexión la rechazará, si después de 15 segundos no recibí un refresco de conexión de alguno de sus nodos, lo dará de baja permitiendo una nueva conexión.

Si existe algún cambio en los archivos compartidos por sus nodos, el servidor local deberá de ser capaz de actualizar la información a los otros servidores locales.

## Nodos

En el momento de activarse, solicitará un número de puerto, y conformará su identificador con su dirección IP y el número de puerto.

Levantará una interfaz gráfica donde el nombre de la ventana estará conformado con la palabra “Nodo” seguido del ID del programa. La interfaz gráfica contendrá los siguientes elementos:

- Un campo con el identificador del servidor local que tiene relacionado.
- Un campo de captura donde se podrá indicar el nombre de texto a buscar.
- Un botón de búsqueda de archivos.
- Un campo de texto que informará sobre los siguientes eventos:
  - Intento de conexiones a nodos y su resultado.
  - La solicitud de búsqueda a su servidor local, así como el resultado de dicho evento.
  - Anunciar si el archivo no se encontró.
  - Anunciar los nodos, los MD5 y la cantidad de nodos donde se encontró el archivo buscado.
  - El rango de bits de la descarga por cada uno de los nodos donde se localizó.
- El procedimiento de selección el archivo a descargar cuando se encuentren más de uno con el mismo nombre, pero diferente MD5, será decisión del programador.

Cada aplicación levantada tendrá relacionada una carpeta donde se contendrán los archivos que maneja el nodo, con el mismo nombre que el número de puerto que se le paso.

El nodo, escuchará por un tiempo de 30 segundos todos los servidores locales anunciados en el canal 228.1.1.10 y de forma aleatoria intentará ligarse a alguno de ellos, en caso de que ese servidor local ya se encuentre sin posibles conexiones, se intentará conectar con el siguiente en la lista de IDs ascendente hasta localizar algún servidor local libre.

Al realizarse la búsqueda de un archivo inexistente, mandará un mensaje de archivo no encontrado, en caso contrario lo descargará en la carpeta del nodo que lo solicito.

También implementara un servicio que permita el ingreso y abandono de la red para nodos. Es decir, cada nodo que este corriendo deberá de ser capaz de ajustar la topología al nuevo número de nodos que estén ejecutándose.

Además de esto, cada que se agreguen, actualicen o eliminen archivos en su carpeta compartida, deberá de ser capaz de actualizar la información en el servidor local al que está conectado.

## Desarrollo del examen

### Servidor local

Los servidores locales constarán de la implementación de los siguientes hilos:

- Un servidor multicas para el anuncio del servidor local.
- Un cliente multicas para generar la lista de servidores locales disponibles.
- Un servidor de llamadas a procedimientos remotos para responder a la búsqueda de archivos por cada uno de sus nodos y para la actualización de los archivos compartidos por sus nodos.
- Un cliente de llamadas a procedimientos remotos, para solicitar información de los otros servidores locales existentes.
- Una interfaz gráfica para integrar todos los elementos necesarios.

### Servidor multicas

Cada 5 segundos se estará anunciando para informar al resto de los nodos que sigue activo.

### Cliente multicas

Estará escuchando constantemente los servidores locales activos, para actualizar las tablas necesarias para mantener funcionando el sistema.

### Servidor de llamadas a procedimientos remotos

Realizará varias funciones de forma simultánea:

- Recibirá por parte de los servidores locales, actualizaciones sobre los recursos compartir por sus respectivos nodos.
- Recibirá solicitudes de búsqueda de archivos por parte de sus propios nodos.
- Recibirá peticiones de conexión por parte de los nodos y mantendrá dichas conexiones activas durante los tiempos ya indicados.
- Recibirá actualizaciones de los recursos compartidos por cada uno de los nodos conectados.

#### Cliente de llamadas a procedimientos remotos

Realizará solicitudes de actualización de información a los otros servidores locales cada 60 segundos para actualizar sus propias tablas.

#### Interfaz gráfica

Deberá cumplir con todas las especificaciones indicadas anteriormente

#### Nodo

El nodo de búsqueda y descarga de archivos consta de la implementación de los siguientes hilos:

- Un cliente multicas para generar la lista de servidores disponibles.
- Un cliente de llamadas a procedimiento remotos que invoque el servicio de búsqueda en cada uno de los servidores disponibles.
- Un servidor de flujo para para el envío de archivos.
- Un cliente de flujo para la descarga de archivos.
- Una interfaz gráfica para integrar todos los elementos necesarios.

A continuación, se describirá la funcionalidad que deberá implementar para cada una de ellas.

#### Cliente Multicas

El cliente deberá unirse a la dirección de grupo “228.1.1.10” y leer todos los datagramas que lleguen. De ellos extraerá la dirección IP del anunciante, y el número de puerto del servidor de llamadas a procedimientos remotos del servidor local anunciante (note que es diferente al número de puerto del servidor de flujo del nodo). El nodo definirá un identificador único (ID), usando la combinación de su dirección IP y el número de puerto de su servidor de datagramas.

Una vez conocidos todos los identificadores, elegirá de forma aleatoria uno de los servidores locales para establecer la conexión,

#### Cliente de servidor a procedimientos remotos

Se encargará de varias funciones de forma simultánea:

- Establecerá y mantendrá la conexión activa con el servidor local seleccionado.
- Enviará actualizaciones sobre sus recursos compartidos en caso de ser necesario.

- Realizará solicitudes de información sobre los recursos ofrecidos por otros nodos a través de sus servidores locales.

#### Servidor de flujo

A este servidor se le asignará de forma automática el número de puerto que se solicita al arrancar el programa (se probará con números a partir del 9000). Una vez hecha la solicitud del cliente, deberá buscar en una carpeta específica (una por cada aplicación que corra) el archivo y el rango de bytes solicitados para enviarlos al cliente.

#### Cliente de flujo

Se encargará de recuperar en paralelo los fragmentos de archivo en cada uno de los nodos donde se encontró el archivo buscando, dividiendo la descarga en cada uno de los nodos donde se localizó.

#### Interfaz de usuario

Desde aquí integrarás todos los servicios de una aplicación de búsqueda y descarga de archivos, ya indicados previamente.

## Orden de la revisión

Para la revisión del funcionamiento de la aplicación, los mensajes indicados en el campo de mensajes deberán de aparecer como se indica, en caso contrario no se continuará con la revisión del proyecto.

1. Primeramente, se solicitará que se levanten entre 2 y 4 servidor locales y se le asignaran números de puerto no consecutivos ni ordenados, y deberán de aparecer en la lista correspondiente de la interfaz gráfica. **Cada uno de los nodos levantados deberá de tener su propia interfaz gráfica de lo contrario no se realizará la revisión del examen.**
2. Se eliminará uno de los servidores locales y después de del tiempo indicado deberá de eliminarse de la lista correspondiente.
3. Se volverá a agregar otro servidor local y deberá de adaptarse a los requerimientos especificados.
4. Se solicitará que se levanten entre 3 y 6 nodos asignando números de puerto no ordenados, y la topología debe adaptarse a la nueva configuración. Esto se verificará con los campos correspondientes mencionados en el párrafo anterior. **Recuerde que cada servidor local solo permitirá dos nodos ligados a él. En caso de que no ocurra así, no se podrá seguir con la revisión.**
5. Se eliminará entre 1 y 3 nodos y la topología deberá de adaptarse de la misma forma que se á indicado previamente. **En caso de que no ocurra no se podrá seguir con la revisión.**
6. Se volverán a agregar nuevos nodos que deberán de adaptarse a los requerimientos especificados.
7. De forma aleatoria se agregará, eliminará o actualizará algún archivo de las carpetas compartidas por los nodos y se deberá de actualizar la información correspondiente en los servidores locales.
8. Se buscará un archivo con el mismo MD5 o equivalente que existe en más de una carpeta de nodo, los archivos a buscar serán del tipo pdf o zip.
9. Se buscará un archivo que no existe en ninguna carpeta.
10. Se revisará en el código que los clientes y servidores implementados correspondan a lo solicitados en el proyecto, en caso de que alguno no coincida, el puntaje correspondiente no será tomado en cuenta.



## Rúbrica

Condición	Calificación		
	No funciona	Funciono parcialmente	Funciono correctamente
En el momento de la ejecución programa del nodo y servidor local solicita el número de puerto inicial y aparece en el nombre de la ventana que se despliega.	0	0	.25
Cada uno de los nodos y servidores locales despliega su propia interfaz gráfica y todos los objetos que se solicitan.	0	0	.25
Se ejecutan de 2 a 4 servidores locales			
Los servidores locales aparecen con su ID en la lista de servidores locales activos, sin duplicarse y cada uno con un número de puerto diferente.	0	.5	1
Se elimina un servidor local al azar			
Las listas actualizan su información conforme los requerimientos.	0	.5	1
Si se agrega un nuevo servidor local la información se actualiza correctamente	0	0	.5
Se ejecutan entre 3 y 6 nodos			
Los nodos se conectas a cada servidor local de forma correcta, respetando las reglas y la información aparece correctamente en las interfaces de nodos y servidores locales.	0	.5	1
Se eliminan de 1 a 3 nodos			
Los nodos se conectas a cada servidor local de forma correcta, respetando las reglas y la información aparece correctamente en las interfaces de nodos y servidores locales.	0	.5	1
Se agregan de 1 a 3 nodos			
Los servidores locales actualizan de forma correcta los archivos compartidos por los nodos.	0	0	.5
De forma aleatoria se agrega, elimina o cambia el nombre de algún archivo en una carpeta de uno de los nodos de la topología			

Los servidores locales ajustan la información de los recursos compartidos.	0	1	2
Desde un nodo al azar, se solicita un archivo existente y se va desplegando los mensajes de forma correcta.			1
Dicho nodo, recupera el archivo solicitado de los nodos donde se encontró.	0	.5	1
Si se busca un archivo inexistente se despliega el mensaje correspondiente	0	0	.5
TOTAL			10 puntos