

# Actividad 14

## Networking

### Instrucciones

Como durante el semestre recibiste tantos anuncios por correo electrónico de github sobre issues, y anuncios de tus queridos ayudantes, tu mail colapsó. Por lo tanto, decidiste crear un sistema propio de mensajería con los conocimientos aprendidos en Programación avanzada.

Un correo electrónico es un servicio de red que permite a los usuarios enviar y recibir mensajes mediante redes de comunicación.

El objetivo de esta actividad es desarrollar un servidor que implemente un sistema de correos electrónicos y un cliente que pueda conectarse al sistema. El servidor debe permitir que varios clientes puedan estar conectados a la vez. Los clientes que se conecten correctamente con el servidor, deben ser capaces de enviar y recibir mensajes.

### Funcionalidades

#### Envío de mensajes y archivos

- Al iniciar el cliente se debe permitir el ingreso de una dirección de correo electrónico.
- Luego de elegir una dirección de correo, se debe mostrar una bandeja de entrada con todos los correos recibidos. Los correos deben estar ordenados por orden de llegada junto con la dirección de correo del emisor.
- Se debe dar la opción de enviar un correo electrónico a uno o muchos usuarios al mismo tiempo.
- {BONUS} Los mensajes deben ser guardados en el servidor para futuras ejecuciones del programa. Queda a su criterio el mecanismo a implementar.

### Requerimientos del Servidor

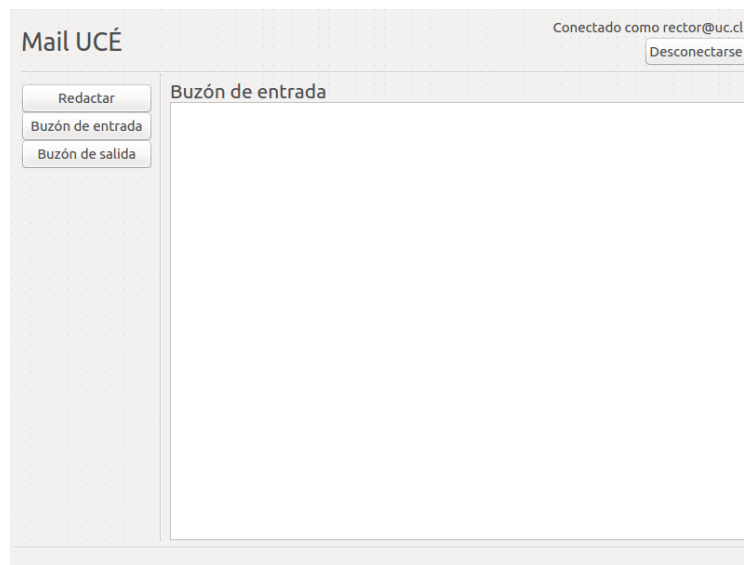
- Debe desarrollar un servidor que implemente un sistema de correo electrónico con todas las funcionalidades especificadas anteriormente.
- Cada vez que el servidor acepte la conexión de un nuevo cliente deberá avisar el evento, junto a la dirección de correo de dicho cliente.
- Cada vez que se reciba algún comando desde alguno de los clientes, el servidor deberá imprimir el comando en la terminal, junto con la dirección de correo del cliente correspondiente.

## Requerimientos del Cliente

- Desarrollar un cliente que pueda conectarse a un servidor. Cuando la conexión se ha completado debe dar aviso de esta conexión a la terminal.
- Implementar un método que permita enviar comandos al servidor.

## Interfaz gráfica

Para el desarrollo del Cliente, se les entregara una interfaz gráfica para facilitar la interacción del usuario con el cliente. Deberá, implementar las funcionalidades de los métodos en la interfaz. Estos métodos estarán comentados en el módulo.



## BONUS

Al iniciar el programa, este deberá ofrecer dos opciones: registrarse en caso de ser un usuario nuevo, o ingresar directamente si es que se trata de uno antiguo. En el modulo `gui` hay un atributo llamado `BONUS` que deberán cambiar a `True` para poder acceder a la ventana inicial de la interfaz gráfica.

- Registro: El sistema deberá pedirle al usuario una dirección de correo electrónico, contraseña y confirmación de la contraseña. Se debe verificar en el servidor que la dirección de correo no esté registrada, es decir, deben ser únicos entre todos los usuarios.
- La base de datos de los usuarios debe mantenerse. Es decir, los usuarios registrados durante una ejecución del programa deberán estar siempre disponibles durante toda ejecución en todo momento. Queda a su criterio el mecanismo a implementar.
- Inicio Sesión: debe pedir al usuario su dirección de correo electrónico y contraseña.

## Notas

- Todas las interacciones del usuario con el cliente, deben ser prácticas y aplicables. Interacciones mal hechas, *i.e.*, que fallen con inputs incorrectos del usuario, etc, serán sancionadas.
- Todo las interacciones debeb ser hechas a través del servidor. El cliente no puede acceder directamente a los datos.
- Debe crear al menos dos archivos por separado: servidor.py y cliente.py.

## To - DO

- Crear la clase Servidor. (4.00 pts)
  - Manejo de conexiones concurrentes con el servidor (Permite tener conectados a varios clientes simultáneamente). (1.00 pts)
  - Correcta implementación de envío de mensajes. (3.00 pts)
- Crear la clase Cliente. (2.00 pts)
  - Iniciar conexión con el servidor. (1.00 pts)
  - Puede enviar y recibir datos del servidor. (1.00 pts)
- Bonus. (2.00 pts)
  - Se guardan los mensajes en servidor. (0.50 pts)
  - Se puede iniciar sesion correctamente. (0.50 pts)
  - Se puede registrar correctamente. (1.00 pts)

## Entrega

- **Lugar:** GIT - Carpeta: Actividades/AC14
- **Hora:** 16:55