

# Proyecto de Sistemas Distribuidos: FTP Centralizado

Nombre del Estudiante

Septiembre 2025

# Alcance del Proyecto

El objetivo del proyecto es implementar un sistema FTP centralizado que cumpla con la mayoría de los comandos definidos en el RFC 959, asegurando compatibilidad con clientes externos como FileZilla. Esta versión servirá como base para la futura implementación distribuida.

## Análisis de Requerimientos

### Funcionales:

- Soportar comandos principales de FTP según RFC 959 (USER, PASS, CWD, PWD, LIST, RETR, STOR, DELE, MKD, RMD, etc.).
- Manejar sesiones de usuario con login y logout.

### No funcionales:

- Compatibilidad con clientes externos.
- Uso de Docker para aislar el entorno de ejecución.
- Preparación para la futura versión distribuida.

## Arquitectura de Software

Los componentes del sistema se organizarán en tres capas principales:

- **Interface Level** : Se encarga de la interacción con el usuario.
- **Processing Level** : Se encarga de implementar el protocolo(envío y recibo de comandos, respuestas, etc).
- **Data Level** : Se encarga de gestionar el sistema de archivos.

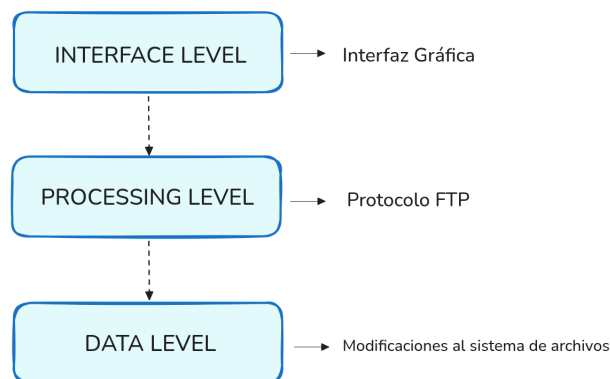


Figure 1: Arquitectura de Software.

# Arquitectura de Sistema

Estas 3 capas estarán físicamente distribuidas en dos máquinas: **cliente** y **servidor**.

La **capa de interfaz** estará completamente contenida en el **cliente** y consistirá en una interfaz gráfica que incluye una **terminal** donde el usuario podrá ingresar los comandos FTP.

La entrada del usuario pasará a una parte de la capa de procesamiento llamada **Módulo de Comunicación** que se encarga de enviar los comandos desde el host del cliente y recibir las respuestas del servidor.

El resto de la capa de procesamiento se encuentra en el **servidor**, con un módulo principal **Connection Manager** el cual se mantiene a la escucha de peticiones por parte de clientes.

Al recibir una petición establece conexión con el cliente y delega la comunicación a otro módulo.

Para procesar los comandos recibidos por los clientes, se cuenta con varios módulos independientes, cada uno con una función específica.

Por último el módulo **FileSystemManager** se encarga de todas las operaciones que se realizan sobre el sistema de archivos.

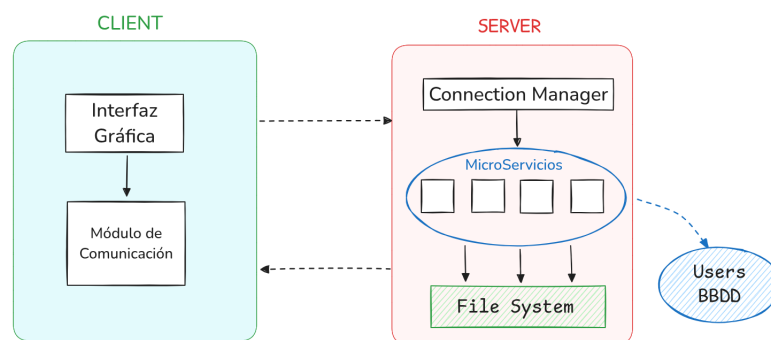


Figure 2: Arquitectura de Sistema.