

Gestor Copias de Seguridad en Java

ERIC

[HTTPS://GITHUB.COM/LAESX/GESTORFTP](https://github.com/LAESX/GESTORFTP)

1. Diseño de la Aplicación

- ▶ La aplicación tiene 2 partes, una que se encarga de subir una carpeta al servidor FTP y otra que se encarga de sincronizar una carpeta local con una carpeta remota en el servidor FTP.
- ▶ A continuación voy a explicar un poco el diseño en general de cada una de las partes de la aplicación.

Clase GestorFTP

La clase GestorFTP es la encargada de gestionar la conexión con el servidor FTP. Esta clase se encarga de conectarse y desconectarse del servidor, así como de realizar operaciones de FTP como subir y descargar archivos.

La podemos usar de 2 maneras:

- ▶ Crear una instancia, pasándole la carpeta que queremos enviar y luego iniciar el hilo, que conectará el servidor FTP, comprimirá y enviará el archivo.
- ▶ Crear una instancia vacía y usar los métodos que nos facilita para enviar archivos, recibirlos, lista directorios, crear directorios, etc.

Clase SincronizadorFTP

- ▶ La clase SincronizadorFTP se encarga de sincronizar una carpeta local con una carpeta remota en el servidor FTP. Para realizar las operaciones de FTP, usa la clase GestorFTP.
- ▶ Primero leemos la lista de archivos locales y remotos y la comparamos, para solo subir los archivos que tengan cambios o que sean nuevos.
- ▶ La sincronización se realiza en un bucle infinito, con un tiempo de refresco configurable.

Clases Main

- ▶ Por último tenemos 2 clases main que usaremos para ejecutar cada una de las partes de la aplicación.
- ▶ El main de sincronización:
- ▶ Y el main de backup:

```
public class MainSincro {  
    ± Laesx  
    public static void main(String[] args) {  
        SincronizadorFTP sincro = null;  
        try {  
            sincro = new SincronizadorFTP( carpetaLocal: "D:\\Test", carpetaRemota: "Test", tiempoRefresco: 5000);  
            sincro.start();  
        } catch (IOException e) {  
            throw new RuntimeException(e);  
        }  
    }  
}
```

```
public class MainBackup {  
    ± Laesx *  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        System.out.print("Introduce el nombre de la carpeta que quieres comprimir (puede ser una ruta relativa o absoluta): ");  
        String folderName = scanner.nextLine();  
        folderName = folderName.replace(target: "\"", replacement: ""); // Elimina las comillas dobles  
        folderName = folderName.replace(target: "'", replacement: ""); // Elimina las comillas simples  
  
        GestorFTP gestorFTP = new GestorFTP(folderName);  
        gestorFTP.start();  
    }  
}
```

2.1. Análisis de funcionamiento (GestorFTP)

- ▶ Iniciamos la Aplicación por el main, que nos pedirá el nombre de la carpeta que queremos comprimir y subir al servidor FTP.
- ▶ Una vez introducido el nombre de la carpeta, se inicia un nuevo hilo con una instancia de la clase GestorFTP para realizar la compresión y la transferencia de la carpeta al servidor FTP.

```
Introduce el nombre de la carpeta que quieres comprimir (puede ser una dirección relativa o absoluta): E:\Video
Comprimiendo carpeta...
Carpeta comprimida correctamente
Enviando archivo tar...
Archivo tar subido correctamente
```

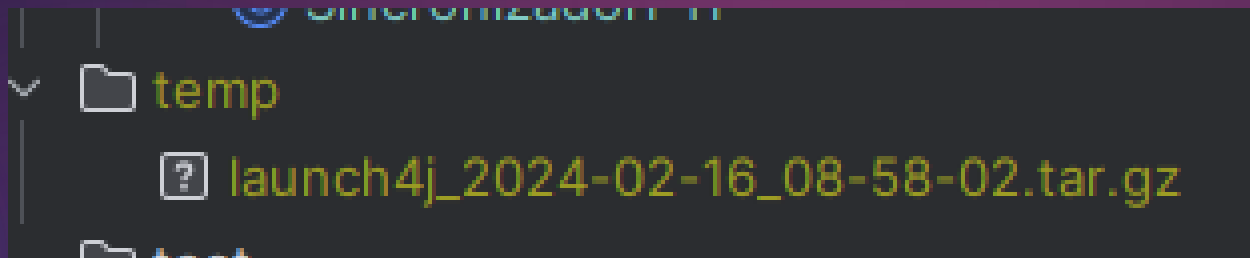
Log de FTP de GestorFTP

- En el log de FTP podemos ver como se realiza la conexión con el servidor FTP y se sube el archivo.

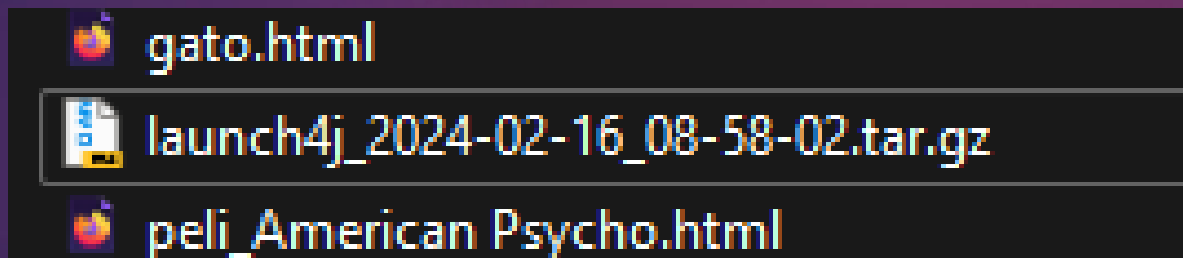
14/02/2024 16:53:24	FTP Session 55 127.0.0.1	Response	220-FileZilla Server 1.8.1
14/02/2024 16:53:24	FTP Session 55 127.0.0.1	Response	220 Please visit https://filezilla-project.org/
14/02/2024 16:53:24	FTP Session 55 127.0.0.1	Command	USER eric
14/02/2024 16:53:24	FTP Session 55 127.0.0.1	Response	331 Please, specify the password.
14/02/2024 16:53:24	FTP Session 55 127.0.0.1	Command	PASS ****
14/02/2024 16:53:24	FTP Session 55 127.0.0.1	Response	230 Login successful.
14/02/2024 16:53:24	FTP Session 55 127.0.0....	Command	TYPE I
14/02/2024 16:53:24	FTP Session 55 127.0.0....	Response	200 Type set to I
14/02/2024 16:53:49	FTP Session 55 127.0.0....	Command	PORT 127,0,0,1,235,183
14/02/2024 16:53:49	FTP Session 55 127.0.0....	Response	200 PORT command successful.
14/02/2024 16:53:49	FTP Session 55 127.0.0....	Command	STOR Video_2024-02-14_16-53-24.tar.gz
14/02/2024 16:53:49	FTP Session 55 127.0.0....	Response	150 Starting data transfer.
14/02/2024 16:53:56	FTP Session 55 127.0.0....	Response	226 Operation successful
14/02/2024 16:53:56	FTP Server	Status	Session 55 ended gracefully.

Archivo creado

- ▶ Aquí podemos ver como nos ha creado un archivo temporal de la carpeta que hayamos indicado.



- ▶ Y como nos ha guardado el zip en el servidor FTP:



2.2. Análisis de funcionamiento (SincronizadorFTP)

- ▶ La clase SincronizadorFTP primero lee todos los archivos del remoto y de la carpeta local y los guarda en 2 sets de archivos.
- ▶ Luego compara los archivos de ambos sets y si hay archivos que no están en el remoto, los sube, si hay archivos que no están en la carpeta local, los descarga.

2.2.1. SincronizadorFTP

Aquí es donde comprobamos cada archivo, creado una lista para cada estado.

Los cuales son, borrado, modificado o creado.

Se comprueba también la fecha de los archivos, y si los archivos del local son más recientes, los actualiza.

```
// Obtener la lista de archivos locales y remotos
Set<String> archivosLocales = obtenerArchivosLocales();
Set<String> archivosRemotos2 = gestorFTP.listarArchivos(carpetasRemotas);

Set<String> archivosRemotos = new HashSet<>();

for (String archivo : archivosRemotos2) {
    archivosRemotos.add(archivo.replace(target: carpetasRemotas + "/", replacement: ""));
}

System.out.println("Archivos locales: " + archivosLocales);
System.out.println("Archivos remotos: " + archivosRemotos);

// Identificar los archivos que no cambian, que han sido modificados o creados, y que han sido borrados
Set<String> archivosNoCambiados = new HashSet<>(archivosLocales);
archivosNoCambiados.retainAll(archivosRemotos);

System.out.println("Archivos no cambiados: " + archivosNoCambiados);

Set<String> archivosModificadosOCreados = new HashSet<>(archivosLocales);
archivosModificadosOCreados.removeAll(archivosNoCambiados);
// Añadimos los archivos locales que se han actualizado
archivosModificadosOCreados.addAll(obtenerArchivosModificados(archivosLocales, archivosRemotos));

System.out.println("Archivos modificados o creados: " + archivosModificadosOCreados);

Set<String> archivosBorrados = new HashSet<>(archivosRemotos);
archivosBorrados.removeAll(archivosNoCambiados);
```

3. Pruebas

- ▶ Hice mucho tipo de pruebas, sobre todo a la clase de sincronización, pasando todo tipo de archivos mientras estaba funcionando para comprobar su robustez.
- ▶ En las siguientes diapositivas enseñé alguna de las pruebas que le hice a cada clase.

Prueba ejercicio a) backup

- ▶ Aquí pruebo varias carpetas podemos ver que sabe identificar carpetas relativas a su ejecución, o rutas absolutas hacia la carpeta.

The screenshot displays a Windows desktop environment with several open windows. At the top, two File Explorer windows are visible. The left window, titled 'FTP', shows the contents of a remote directory, including folders like 'Anime Screenshots', 'Insti', 'Photoshop', and '2024', as well as files like 'launched_2024-02-14_13-28-39.tar.gz'. The right window, titled 'temp', shows a single file 'launched_2024-02-16_08-58-02.tar.gz'. Below these, a terminal window titled 'MainBackup' is open, showing a command prompt with the following text:

```
"C:\Program Files\OpenJDK\jdk-17.0.2\bin\java.exe" "-javaagent:C:\Users\losgu\AppData\Local\Programs\IntelliJ IDEA Ultimate\lib\idea_rt.jar=51216:C:\Program Files\OpenJDK\jdk-17.0.2\bin" -Dfile.encoding=UTF-8
```

At the bottom right, an 'Administration interface - FileZilla Server 1.8.1' window is open, displaying a log of FTP sessions. The log shows a series of commands and responses, including 'LIST Test', 'PORT 127.0.0.1,202,53', and '200 PORT command successful'.

Date/Time	Info	Type	Message
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Command	LIST Test
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Response	150 Starting data transfer.
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Response	226 Operation successful
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Command	PORT 127.0.0.1,202,53
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Response	200 PORT command successful.
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Command	LIST Test/Carpeta
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Response	150 Starting data transfer.
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Response	226 Operation successful
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Command	PORT 127.0.0.1,202,55
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Response	200 PORT command successful.
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Command	LIST Test/Carpeta
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Response	150 Starting data transfer.
16/02/2024 18:11:06	FTP Session 3 127.0.0.1...	Response	226 Operation successful

Prueba ejercicio b) sincronización

- ▶ En este ejemplo añadido un archivo a la carpeta local, a otro le cambio el nombre y el archivo que añadí, lo borro.

The screenshot displays the FileZilla interface with a local directory 'Test' selected. The directory contains three items: 'Carpeta' (a folder), 'Hola.txt' (a text file), and 'New Documento de texto.txt' (a text file). The 'Run' tab at the bottom shows the synchronization status, indicating that the local files are up-to-date with the remote files. The 'Administration interface - FileZilla Server 1.8.1' window is also visible, showing a log of FTP sessions and commands.

Local Directory: Test

Name	Date modified	Type	Size
Carpeta	14/02/2024 18:02	File folder	
Hola.txt	14/02/2024 17:54	Documento de te...	0 KB
New Documento de texto.txt	14/02/2024 17:54	Documento de te...	0 KB

Run: MainSincro

```
Archivos remotos: [Carpeta/New Documento de texto.txt, Carpeta/Fichero.txt, Carpeta/dsgedfghfd.txt, New Documento de texto.txt, Hola.txt]
Archivos no cambiados: [Carpeta/Fichero.txt, Carpeta/New Documento de texto.txt, Carpeta/dsgedfghfd.txt, New Documento de texto.txt, Hola.txt]
Archivos modificados o creados: []
Archivos borrados: []
Sincronización completada
Sincronizando...
Archivos Locales: [Carpeta/Fichero.txt, Carpeta/New Documento de texto.txt, Carpeta/dsgedfghfd.txt, New Documento de texto.txt, Hola.txt]
Archivos remotos: [Carpeta/New Documento de texto.txt, Carpeta/Fichero.txt, Carpeta/dsgedfghfd.txt, New Documento de texto.txt, Hola.txt]
Archivos no cambiados: [Carpeta/Fichero.txt, Carpeta/New Documento de texto.txt, Carpeta/dsgedfghfd.txt, New Documento de texto.txt, Hola.txt]
Archivos modificados o creados: []
Archivos borrados: []
Sincronización completada
```

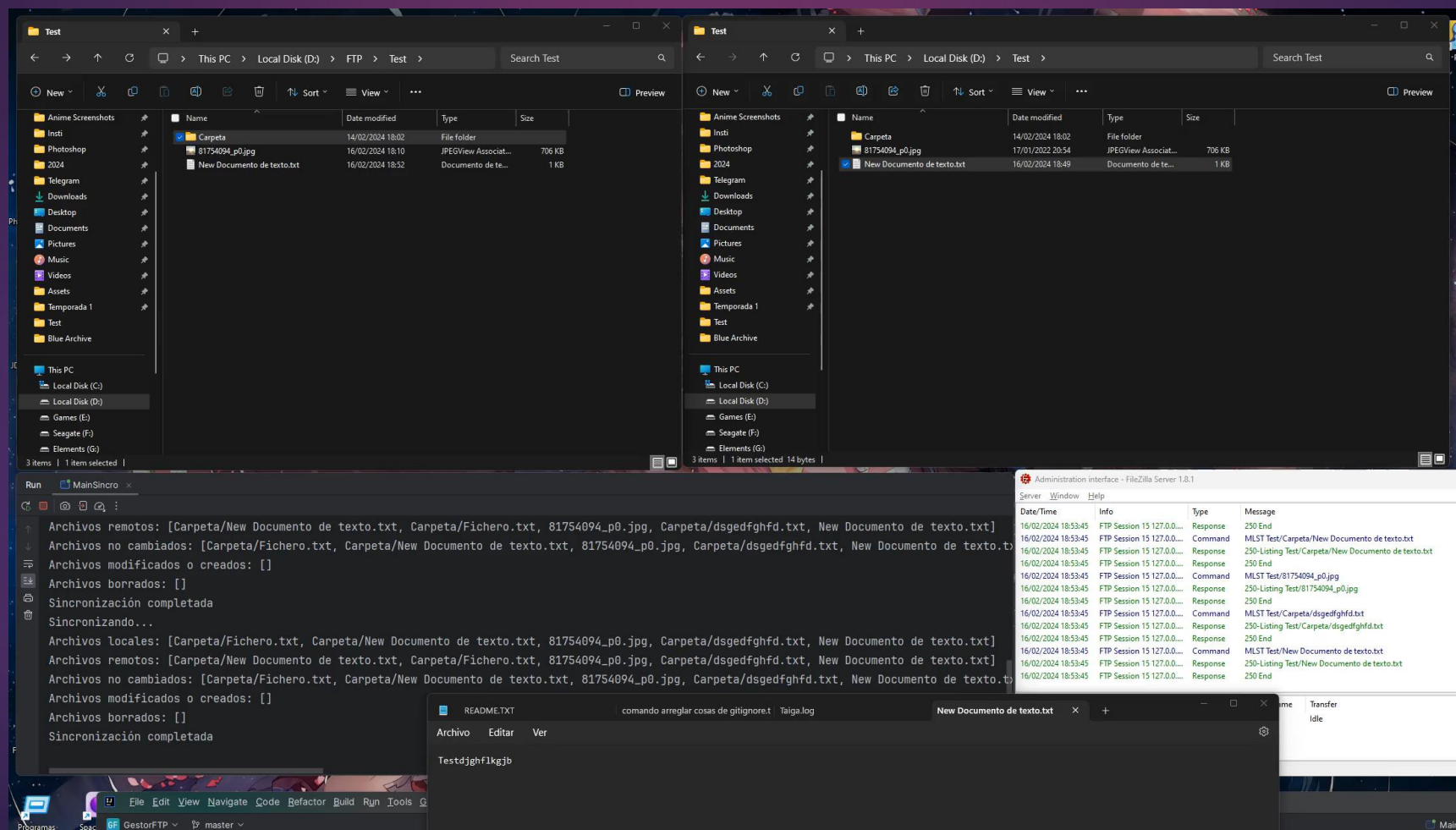
Administration interface - FileZilla Server 1.8.1

Date/Time	Info	Type	Message
16/02/2024 18:10:36	FTP Session 3 127.0.0.1...	Response	150 Starting data transfer.
16/02/2024 18:10:36	FTP Session 3 127.0.0.1...	Response	226 Operation successful
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Command	PORT 127,0,0,1,200,50
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	200 PORT command successful.
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Command	NLST Test
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	150 Starting data transfer.
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	226 Operation successful
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Command	PORT 127,0,0,1,200,52
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	200 PORT command successful.
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Command	LIST Test
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	150 Starting data transfer.
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	226 Operation successful
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Command	PORT 127,0,0,1,200,54
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	200 PORT command successful.
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Command	NLST Test/Carpeta
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	150 Starting data transfer.
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	226 Operation successful
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Command	PORT 127,0,0,1,200,56
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	200 PORT command successful.
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Command	LIST Test/Carpeta
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	150 Starting data transfer.
16/02/2024 18:10:41	FTP Session 3 127.0.0.1...	Response	226 Operation successful

Connected to 127.0.0.1

Prueba 2 ejercicio b) modificado

- ▶ En esta prueba compruebo que, al modificar un archivo, se ha comparado su tiempo de modificación y solo se cambia este.



4. Propuesta de mejoras

- ▶ Una mejora que se podría implementar es que también compruebe el hash de los archivos para saber si los archivos han cambiado o comprobar que todos los archivos se han copiado de forma íntegra en el directorio FTP.
- ▶ También podríamos hacer que guardara una copia de los archivos que se han modificado en un zip por si queremos tener una copia de todos los cambios por los que ha pasado cada archivo.
- ▶ Podríamos hacer también que fuera sincronización de 2 vías, por lo que si el servidor de destino tiene archivos adicionales o más nuevos, que los descargara a nuestro ordenador local.