# **EXPLICACION DE CLASES**

PACKAGE

.JAVA

|  |  |
| --- | --- |
| CONTROLADOR |  |
| **ControladorCliente** | La clase “**ControladorCliente”** es un controlador que implementa la interfaz de “**ActionListener”**. Esta clase se encarga de controlar la interacción entre la vista y el modelo de la aplicación de cliente-servidor que permite subir y descargar archivos a través de un socket. |
| **ControladorServidor** | La clase “**ControladorServidor”** es una clase que implementa la interfaz “**ActionListener”** y se utiliza para manejar la lógica del servidor. |
| **GUI** |  |
| **VistaCliente** | Inicializa la interfaz del Cliente |
| **VistaServidor** | Inicializa la interfaz del Servidor |
| **TASKS** |  |
| **HiloCliente** | La clase “**HiloCliente”** extiende de “**Thread”** que se encargará de manejar las solicitudes de los clientes que se conecten al servidor. Esta clase establece flujos de entrada y salida para comunicarse con el cliente a través de un socket. |
| **WorkerDescarga** | La clase **WorkerDescarga** extiende de la clase **SwingWorker** que se encarga de descargar un archivo desde el servidor y mostrará su progreso mediante una barra de progreso, por último, realiza la tarea de descargar un archivo en segundo plano para no sobrecargar la GUI. |
| **WorkerSubida** | La clase "**WorkerSubida"** extiende la clase **SwingWorker** que se encarga de subir los ficheros al servidor y mostrará su progreso mediante una barra de progreso, por último realiza la tarea de subir un fichero a un servidor en segundo plano para no sobrecargar la GUI. |

# **FUNCIONALIDADES**

|  |  |
| --- | --- |
| OBLIGATORIAS |  |
| Debe permitir conectar con el servidor indicando su IP, y poder obtener la lista actualizada de los ficheros disponibles en el servidor para ser descargados. | Checkmark with solid fill |
| Debe permitir conectar con el servidor indicando su IP, y poder obtener la lista actualizada de los ficheros disponibles en el servidor para ser descargados. | Checkmark with solid fill |
| Debe permitir seleccionar un fichero de su propio sistema y subirlo al servidor. Después de cada transferencia podrá realizar otra. | Checkmark with solid fill |
| Debe notificar la finalización de la transferencia actual en la GUI, asegurando que el acceso a componentes de la GUI se realice en el EDT y la transferencia no sobrecarguen este hilo. | Checkmark with solid fill |
| Debe permitir la conexión y transferencia (subida/bajada) de ficheros desde varios clientes de forma simultánea, manteniendo actualizada la lista de descargas. | Checkmark with solid fill |
| OPCIONALES |  |
| El cliente representa el progreso de la transferencia actual (subidas y bajadas) mediante barras de progreso. | Checkmark with solid fill |
| El servidor dispone de una GUI que muestra la lista actualizada de ficheros disponibles. | Checkmark with solid fill |
| Cada cliente tiene un nombre, y el servidor muestra la lista de los clientes conectados para conocer quién es quién. | Checkmark with solid fill |
| El cliente permitirá pausar y reanudar el progreso de la/s subidas. | Checkmark with solid fill |
| El cliente permitirá pausar y reanudar el progreso de la/s bajadas. | Checkmark with solid fill |

**Repositorio GIT:**  
[**https://github.com/dacm1999/Descargas-Servidor**](https://github.com/dacm1999/Descargas-Servidor)