

Manual de Funcionamiento

Aplicación de Mensajería

Cliente-Servidor con Tuberías con Nombre

1. Introducción:

Este manual describe el funcionamiento de una aplicación de mensajería simple basada en tuberías con nombre, utilizando dos aplicaciones distintas: una aplicación servidor y una aplicación cliente, ambas tienen la misma funcionalidad, ya que permiten enviar y recibir mensajes entre ellas.

2. Requisitos Previos:

- Asegúrate de tener instalado el entorno de desarrollo para C#, en este caso Visual Studio, con todas sus actualizaciones.

3. Ejecución:

- Abre la aplicación PipeFormServidor en Visual Studio y compílala.
- Ejecuta la aplicación compilada del servidor.
- Haz clic en el botón "Activar" en la aplicación del servidor para iniciar la escucha de mensajes.
- Abre la aplicación PipeFromCliente en Visual Studio y compílala.
- Ejecuta la aplicación compilada del cliente.

4. Funcionamiento de la Aplicación Servidor:

- Una vez que la aplicación del servidor está en ejecución, haz clic en el botón "Activar".
- La aplicación servidor mostrará el mensaje "El servidor está activado".
- Ingresa un mensaje en el cuadro de texto de la aplicación cliente y haz clic en "Enviar".
- El mensaje se enviará al servidor que se mostrará en el cuadro de texto de mensajes.

5. Funcionamiento de la Aplicación Cliente:

- Una vez que la aplicación del cliente está en ejecución, haz clic en el botón "Activar".
- Haz clic en el botón "Activar" en la aplicación cliente.
- La aplicación cliente mostrará el mensaje "El cliente está listo para recibir mensajes".
- Ingresa un mensaje en el cuadro de texto de la aplicación servidor y haz clic en "Enviar".
- El mensaje se enviará al cliente, y se mostrará en el cuadro de texto de mensajes de la aplicación cliente.

6. Consideraciones Adicionales:

Si encuentras algún problema durante la ejecución, revisa la consola de salida en Visual Studio para obtener información sobre posibles errores.

La aplicación es mejorable pero por falta de tiempo actualmente voy a recoger en el manual una serie de puntos donde me gustaría mejorarla en el futuro:

Para abrir la aplicación servidor y que esta abra la app cliente, se podría utilizar el método `Start()` de la clase `Process`, con este método se podría crear un método que ejecute la app cliente al que llamaríamos desde el botón activar por ejemplo.

Manejo de Excepciones:

- Se puede implementar un manejo de excepciones más robusto para garantizar que la aplicación sea más resistente a errores.

Hilos Múltiples:

- Se puede considerar el uso de hilos múltiples para manejar conexiones de clientes concurrentes.
- Se puede mejorar para poder cerrar y liberar recursos correctamente cuando un cliente se desconecta.

Interfaz Gráfica más Amigable:

- Se podría mejorar la interfaz gráfica para proporcionar una experiencia de usuario más agradable y amigable.

Manejo de Errores de Conexión:

- Se podrían mejorar los mensajes de error y su disposición.

Asincronía:

- Se podrían usar operaciones asíncronas para mejorar la capacidad de respuesta de las interfaces, por ejemplo, hay un problema que cuando activamos los botones, se bloquea la interfaz hasta que se recibe el mensaje, por lo que tenemos que detener la ejecución para cerrar las aplicaciones.

Manejo de Mensajes Multilínea:

- Para permitir el envío y la visualización de mensajes multilínea para mejorar la capacidad de comunicación.

Control de Versiones:

- Se podría haber usado Git como SCV para tener un control sobre las distintas versiones de las aplicaciones.

