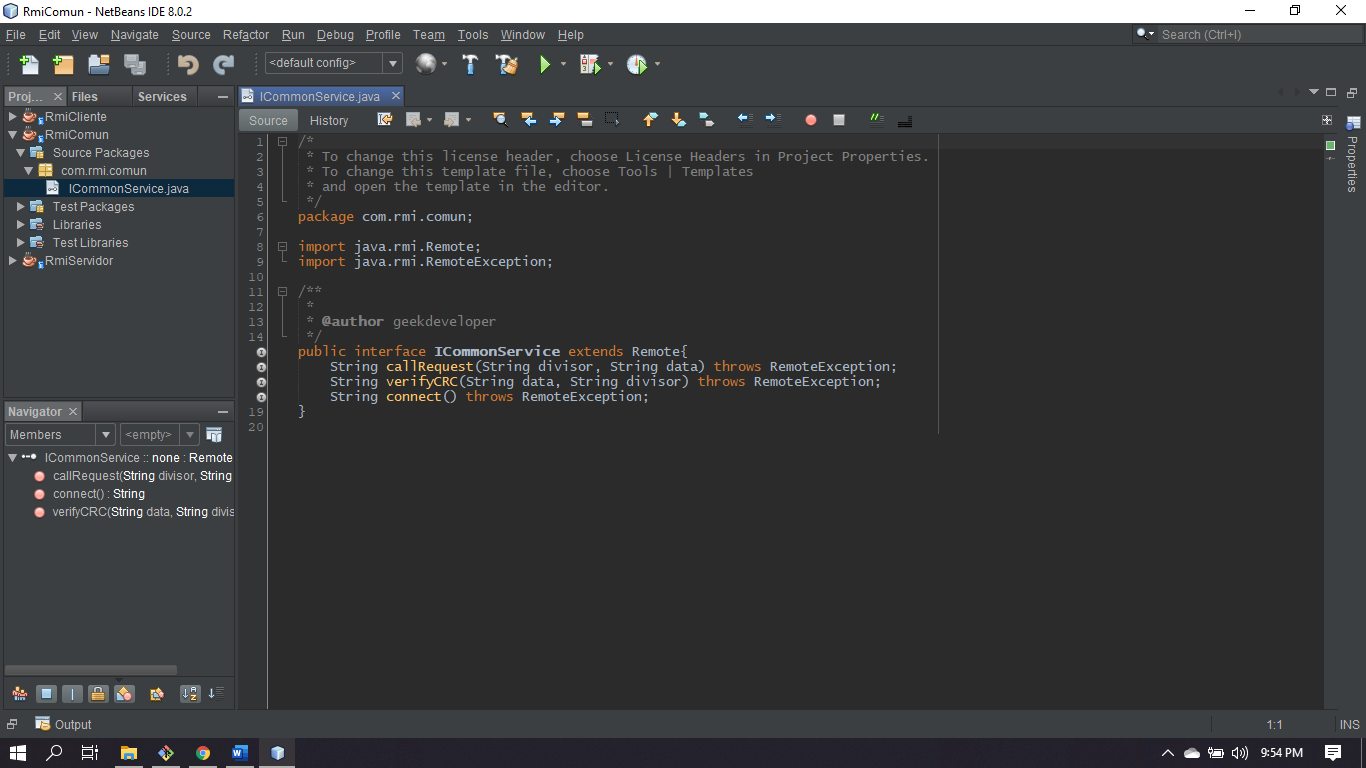
**Guía de uso Algoritmo CRC**

**Integrantes: Gian Carlos Cuello Calle, Rosiris Hernández**

Previamente debe tener instalado un ejecutor o compilador de código Java, o un IDE respectivo para su ejecución (Recomendable NETBEANS).

Se abre el proyecto en Netbeans, los 3, se ejecuta el RmiServidor, y luego el RmiCliente.

El RmiServidor, contiene los codigos fuentes que se van a ejecutar, posteriormente el código CRC, en cada método.

El RmiComun, es una interfaz que se va a implementar en el proyecto RmiServidor.

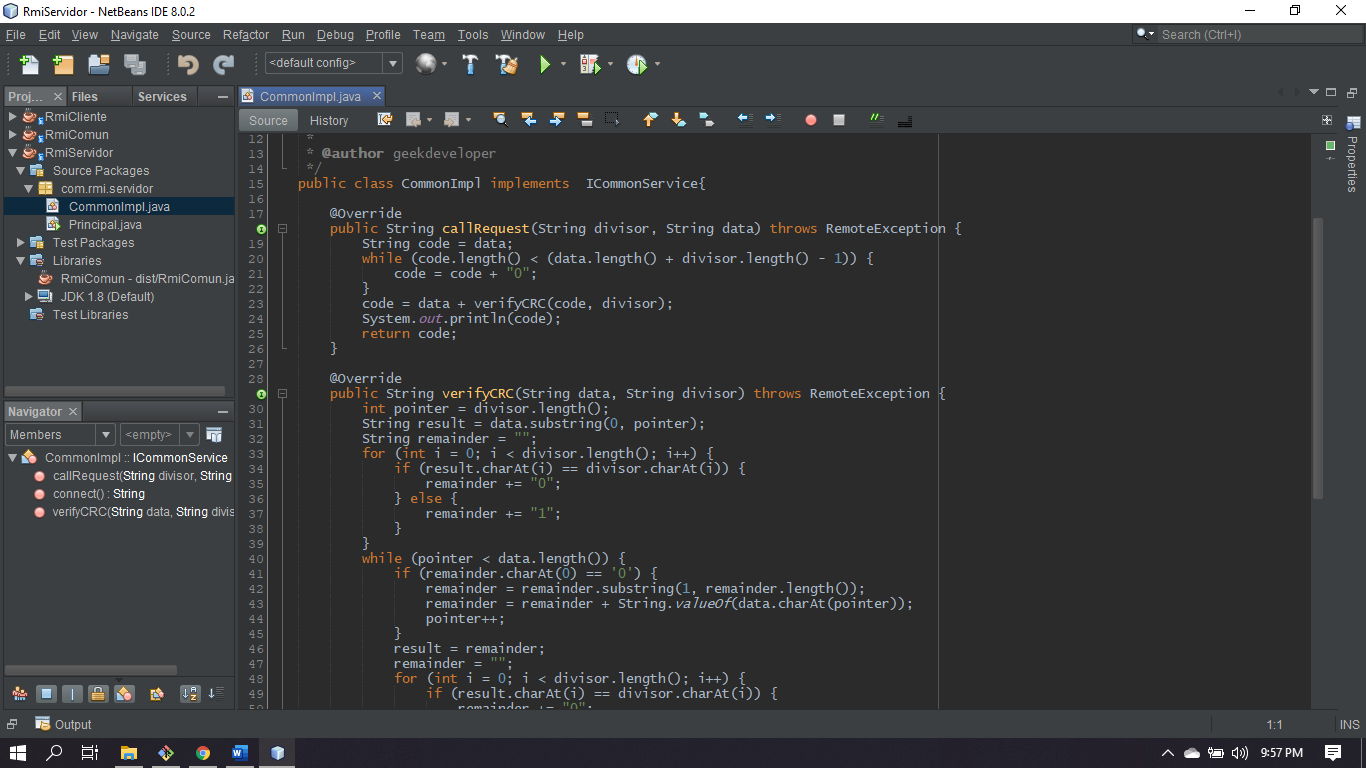
Esta interfaz contendrá 3 métodos que devuelven un String.

**callRequest**

**verifyCRC**

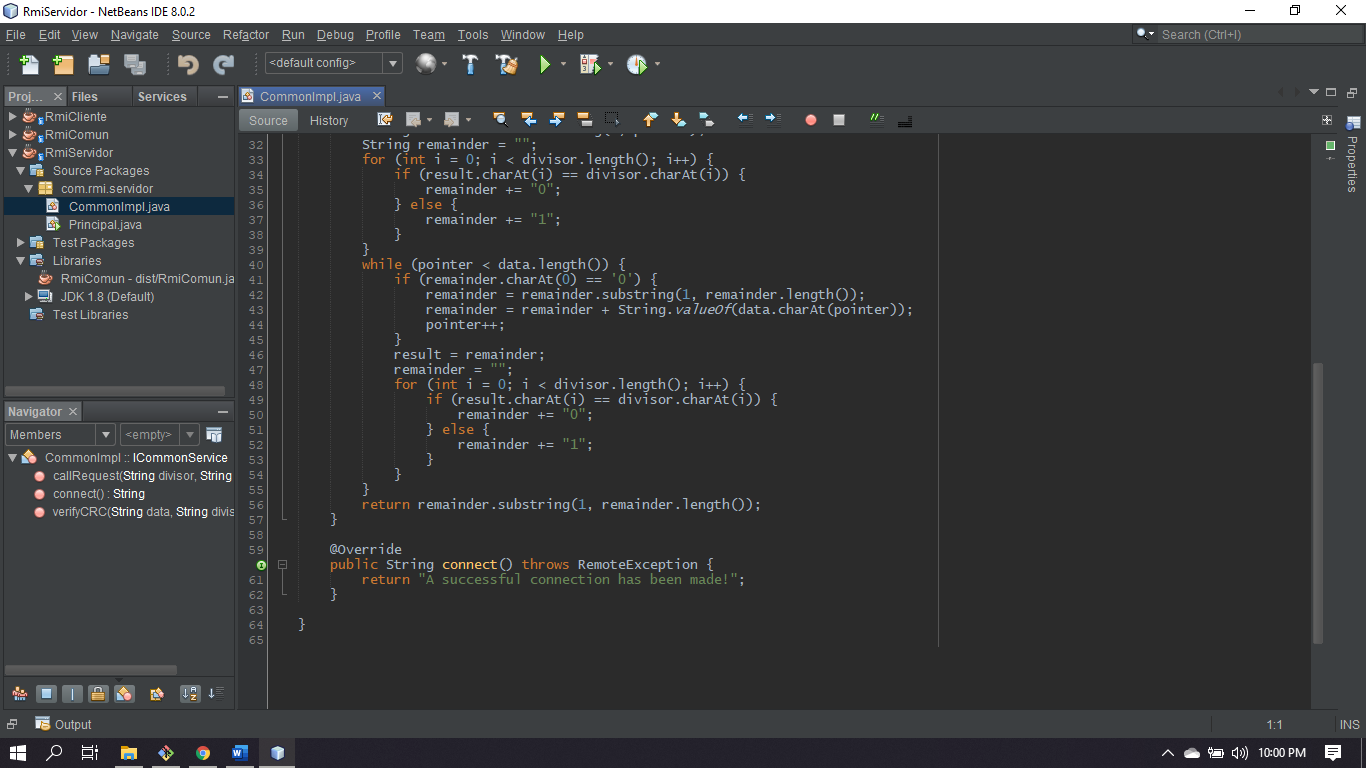
**connect**

**RmiServidor**



Contendra los metodos implementados del proyecto RmiComun, y luego estos metodos se le agrega su funcionalidad.

**callRequest:** Recibe por parámetro los datos del divisor, y el dato a dividir en este caso los datos binarios.

**verifyCRC:** Recibe el binario obtenido con el divisor y retorna el resultado en String.

**connect:** Realiza la conexión entre el Cliente y el servidor.

Muestra de ejecución.

Pasos para ejecutar:

1. Se ejecuta el proyecto RmiServidor.
2. Se ejecuta el proyecto RmiCliente.
3. Se da click en el botón **Connect.**
4. Si la conexión fue exitosa, mostrara el mensaje de Connect del Servidor en la aplicación cliente.
5. Se habilitarán los botones y campos para el envio y verificación de los datos.
6. El el primer campo que dice “**Write your divisor or Generator**”, se escribe el numero por el cual se va a dividir el binario.
7. En el segundo campo que dice “**Enter yout Information to send**”, se escribe el binario que se dividirá.
8. Se presiona el botón “**SendRequest**”.
9. Del servidor nos devolverá una respuesta con el numero obtenido en la división, que se agregará al 3er campo.
10. En la consola nos mostrara el resultado obtenido y se agregara al campo.
11. Luego de esto podemos darle al botón de “**Check**”, y notaremos si el resultado obtenido tiene errores o no.
12. Si el numero es correcto obtendremos un mensaje diciendo “**Sin errores en el código obtenido**”.
13. Si llega a existir un error, lanzara otro mensaje, entonces procedemos a cambiar el numero obtenido agregando o eliminando un 1 o 0.
14. En este ejemplo se agrego un 1 de mas. Y nos lanzo un resultado diciendo “El código obtenido tiene errores”.

