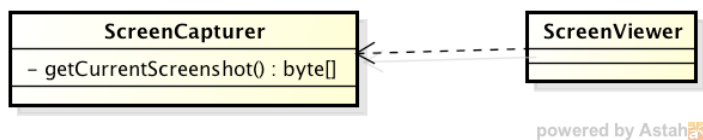


Escuela Colombiana de Ingeniería
Arquitecturas de Software
Sistemas distribuidos – RMI

hector.cadavid@escuelaing.edu.co

En este ejercicio se espera implementar una primera versión de una herramienta que permita la asistencia remota a través de Internet, mediante la visualización a distancia de la interfaz de usuario del sistema operativo de la persona a la que se le prestará asistencia.

La versión actual de la herramienta se limita a capturar las imágenes de la interfaz de usuario local, a través de un objeto de tipo *ScreenCapturer*:



Para que la herramienta funcione correctamente (es decir, que en un equipo se pueda ver lo que se está haciendo en otro equipo), se debe lograr que el objeto *ScreenCapturer* se publique, para su acceso remoto, en el equipo al que se dará asistencia. Para esto, debe realizar lo siguiente:

Parte 0. Descargue la aplicación, ejecútela, y analice su funcionamiento actual.

Parte 1. Servidor (el que publicará el objeto remoto, y que correrá en el equipo al que se le prestará asistencia). Ver figura 1.

1. Cree un nuevo proyecto a partir de los fuentes dados, pero suprimiendo la interfaz gráfica.
2. Definir la interfaz del *stub*, de manera que ésta represente el conjunto de operaciones que ofrecerá, remotamente, el objeto que se publicará en el cliente al que se le prestará un soporte (esta interfaz debe heredar la interfaz `java.rmi.Remote`).
3. Crear una clase que implemente dicha interfaz, y que herede de la clase `UnicastRemoteObject`. Tenga también en cuenta que para que un objeto

tenga un comportamiento consistente -en lo que respecta al manejo de excepciones- cuando es utilizado remotamente, cada uno de sus métodos debe propagar las excepciones de tipo *RemoteException*.

4. Crear el programa principal del servidor, el cual debe publicar en el registro RMI el objeto anteriormente creado, usando las clases Registry y LocateRegistry:

```
String thisHost=NetUtils.getIPAddress();
System.setProperty("java.rmi.server.hostname",thisHost);
LocateRegistry.createRegistry(<puerto>);
Registry reg=LocateRegistry.getRegistry(thisHost,<puerto>);
reg.rebind(<nombre asociado al objeto>,objeto);
```

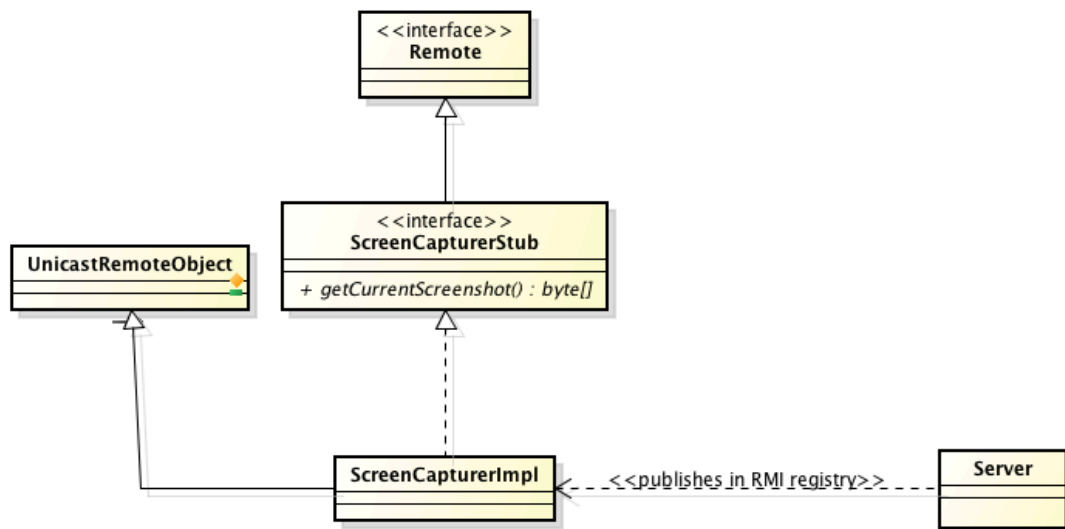


Figure 1 - Modelo de clases - servidor

5. Modifique el archivo de configuración de Maven para que la propiedad 'mainClass' del plugin mave-jar-plugin corresponda al nombre de la clase del programa principal.
6. Genere el jar ejecutable del servidor invocando la fase de empaquetamiento de maven (el jar ejecutable quedará en la ruta /target):

```
mvn package
```

Parte 2. Cliente (el que correrá quien prestará la asistencia remota).

1. Cree un nuevo proyecto a partir de los fuentes dados, dejando únicamente la interfaz de usuario del cliente.
2. A dicho proyecto, agregue una copia de la interfaz *stub* creada para el servidor.
3. Modifique el código para que, en lugar de usar el objeto *ScreenCapturer* local, utilice el que se publicaría remotamente en un servidor

```
//Para obtener el acceso a un registro RMI remoto:  
Registry reg=LocateRegistry.getRegistry(<host>, <puerto>);  
//Para obtener un objeto remoto registrado en dicho registro (este método retorna  
//un proxy que puede tomar la forma del 'stub' de dicha clase):  
reg.lookup(<nombre asociado al objeto remoto>);
```

4. Copie el jar ejecutable del servidor en otro computador y ejecútelo.
5. Configure su programa cliente para que la conexión con el registro RMI se haga usando la dirección IP del otro equipo.
6. Verifique que desde su equipo, se pueda ver lo que está haciendo el equipo de su compañero.

Parte 3.

Modifique lo que haga falta para que la imagen capturada por el servidor (donde la persona asistida), en caso de tener una resolución mayor a la imagen de quien asesora, haga el escalamiento correspondiente. Para esto, será necesario que el cliente envíe como parámetro a la operación del objeto remoto sus dimensiones de pantalla. Si la dimensión enviada por el cliente es menor a la de las imágenes capturadas por el servidor, el servidor retornará una imagen escalada (para revisar cómo hacerlo, puede usar este post: <http://edroidx.blogspot.com/2010/12/how-to-resize-buffered-images.html>).

Parte 4.

Revise las secciones 22.1 y 22.2 de la documentación de Spring:

<http://docs.spring.io/spring/docs/current/spring-framework-reference/html/remoting.html>

Haga una versión alterna del cliente y del servidor en la que se haga uso de Spring para configurar el esquema de acceso remoto al objeto de tipo *ScreenCapturer*.

