**REQUERIMIENTO 3 CORBA**

**MANUAL DE USUARIO**



**Integrantes**

**Eduardo Prado Cañar**

**Yudy Lorena Ramírez Álvarez**

**Profesor**

**Daniel Paz Perafán**

**Universidad del Cauca**

**Facultad de Ingeniería Electrónica y Telecomunicaciones**

**Programa Ingeniería de Sistemas**

**Curso: Sistemas Distribuidos Popayán, Marzo 2016**

Tabla de contenido

[**1.** Introducción 3](#_Toc450335559)

[2. Requerimientos del Sistema: 3](#_Toc450335560)

a) Sistema Operativo 3

[b) Tecnología utilizada……………………………………............................................................................. 3](#_Toc450335562)

[c) Lenguaje de programación 3](#_Toc450335563)

[3. Descripción de la documentación entregada 3](#_Toc450335564)

[4. Descripción de la estructura de directorios, descripción de los archivos de soporte: 4](#_Toc450335565)

[5. Descripción de los archivos fuente (Interfaces, archivos cliente, archivos servidor) 5](#_Toc450335567)

[6. Descripción del proceso de compilación 8](#_Toc450335568)

[7. Descripción del proceso de Ejecución 8](#_Toc450335569)

[8. Descripción del Manejo del Instalador. 8](#_Toc450335570)

# **1.** Introducción

En este documento se encuentran especificados todos los aspectos técnicos a tener en cuenta para el correcto funcionamiento de la aplicación, es decir, el análisis y diseño de la aplicación, lo cual le servirá como referencia al desarrollador de la aplicación.

# 2. Requerimientos del Sistema:

## a) Sistema Operativo: puede ser cualquier sistema operativo que soporte Java, además hay que tener en cuenta que muchas aplicaciones de PC, para que funcionen correctamente es necesario tener instalada Java en el sistema operativo que se esté usando. Es necesario descargar e instalar en la computadora JRE (Java Runtime Environment), conocida como Java Virtual Machine (JVM). Se puede descargar en <http://www.java.com/es/download/> (norfiPc, s.f.)

## b) Tecnología utilizada: Cliente/Servidor usando la especificación CORBA, el cual permite a servicios remotos, es decir que no se encuentran instalados en nuestra máquina necesariamente. La conexión a estos servicios se logra mediante el protocolo de comunicación TCP/IP, donde el cliente debe saber la IP y el puerto al que se desea conectar para alcanzar los servicios remotos.

## c) Lenguaje de programación: se usó Java como lenguaje de .programación ya que este ofrece el mecanismo CORBA que permite interacciones simples y/o complejas (según sea la necesidad), de comunicación entre aplicaciones basadas en la arquitectura CORBA.

# 3. Descripción de la documentación entregada

La entrega del aplicativo, cuenta con tres manuales:

Uno de usuario donde se le ilustra al usuario interesado, la información y guía pertinente para que pueda usar la aplicación. El segundo manual es el técnico que describe los requerimientos técnicos para el desarrollo de la aplicación, los casos de uso que se generan a raíz de estos requerimientos, algunos diagramas de secuencia (los más relevantes), el diagrama de clases y el diagrama de paquetes. Es decir cuenta con toda la información técnica usada y recopilada para desarrollar la aplicación.

El tercer manual es el que se encuentra contenido en este documento, el cual contiene la información necesaria una correcta instalación y/o ejecución de la aplicación.

# 4. Descripción de la estructura de directorios, descripción de los archivos de soporte:

# 

Figura 2. Clientes CORBA 1

Figura 1. Servidores CORBA 1

La estructura general de CORBA en java, para cada **cliente (Administrador/Usuario),** es un paquete “**presentación**” que contiene las clases cliente, es decir las que interactúan con el usuario final de una determinada aplicación; en nuestro caso contiene las GUI de usuario. En el paquete “**controladores**” se encuentran todas aquellas clases usadas para validación de información, gestión de conexión, etc. Por ultimo tenemos el paquete “**sop\_corba**”, que contiene todas las clases generadas por especificación CORBA, y contiene las clases necesarias para los servicios remotos. Por otro lado la estructura general de los servidores CORBA contiene un paquete **“servidor”** que contiene las clases servidor encargadas de gestionar las peticiones de acceso a objetos remotos por parte del cliente, estas son las clases Impl (son las que implementan los métodos definidos en las clases Int del paquete sop\_corba para poder prestar el servicio solicitado). El paquete “**sop\_corba**” y “**sop\_corbaMinBib**” contiene las clases generadas por CORBA a partir de las IDL (interfaces con la definición de los métodos remotos que puede usar el cliente para responder a una solicitud por parte de un usuario).

# 5. Descripción de los archivos fuente (Interfaces, archivos cliente, archivos servidor)

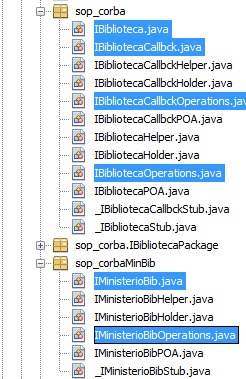


Figura 3. Interfaces Servidor CORBA 2

Las clases en color azul son las interfaces que se consideraron necesarias para el desarrollo de las aplicaciones servidoras, a continuación una breve descripción de cada una de ellas:

Interfaces IBiblioteca, IBibliotecaOperations.

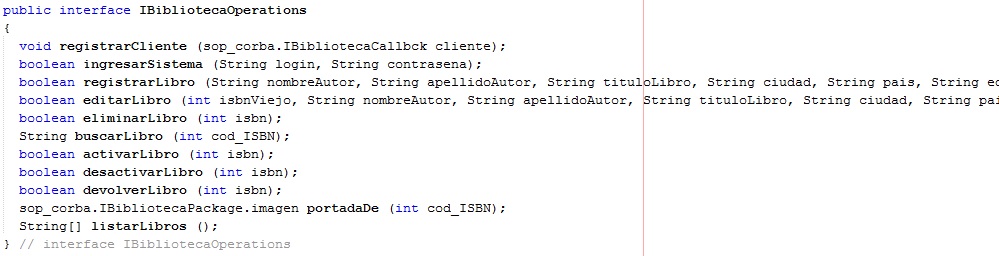


Figura 4. Clase IBibliotecaOperations 1

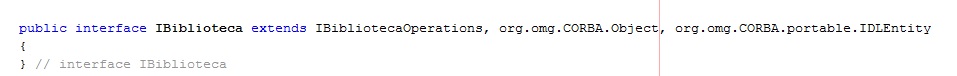


Figura 5. Clase IBiblioteca 1

Estas dos interfaces contienen todos los métodos que permiten a un cliente llevar a cabo los servicios solicitados en este caso el administrador de la biblioteca, en la figura 4 se puede observar los servicios ofrecidos por este servidor Biblioteca.

Interfaces IMinisterioBib, IMinisterioBibOperations

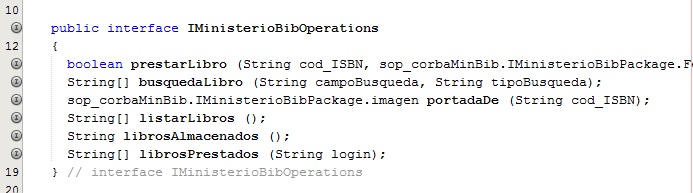


Figura 6. Clase IMinisterioBibOperations 1

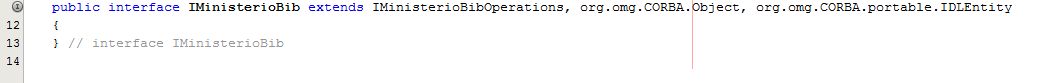


Figura 7. Clase IMinisterioBib 1

Estas dos interfaces contienen todos los métodos que permiten a un cliente, en este caso el mismo servidor ministerio a llevar a cabo los servicios solicitados desde los clientes (usuario, administrados), en la figura 6 se puede observar los servicios ofrecidos por el servidor Biblioteca a el servidor Ministerio.

Nota: Para las interfaces IBibliotecaCllbck, IBibliotecaCllbckOperations, solo contienen un método remoto el cual es notificar() este cuya función es hacer un callback (notificación) al administrador.

Por ultimo las interfaces necesarias para el servidor Ministerio.

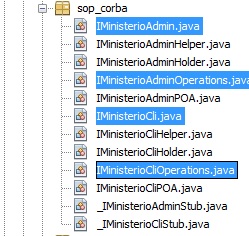


Figura 8. Interfaces Ministerio 1

Interfaces IMinisterioAdmin, IMinisterioAdminOperations

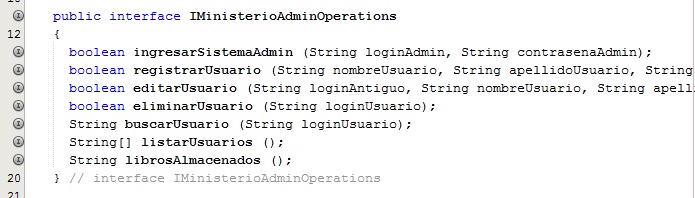


Figura 10. IMinisterioAdminOperation 1

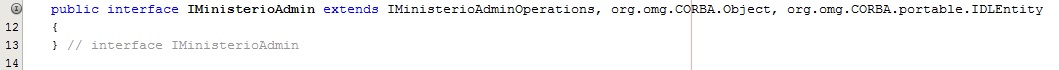


Figura 9. IMinisterioAdmin 1

Estas dos interfaces contienen todos los métodos que permiten a un cliente, en este caso el administrador del ministerio realizar servicios requeridos, en la figura 10 se puede observar todos los servicios ofrecidos por el servidor Ministerio al administrador.

Interfaces IMinisterioCli, IMinisterioCliOperations

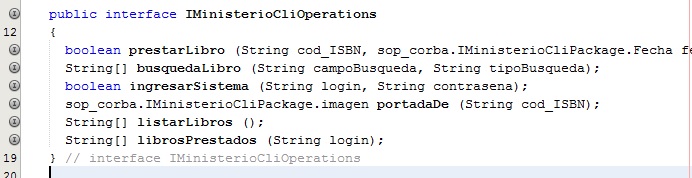


Figura 12. IMinisterioCliOperations 1

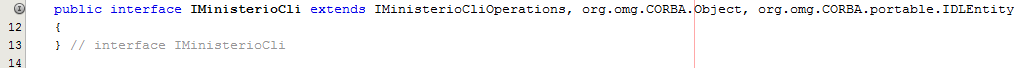


Figura 11. IMinisterioCli 1

Estas dos interfaces contienen todos los métodos que permiten a un cliente, en este caso el usuario del ministerio realizar servicios requeridos, en la figura 10 se puede observar todos los servicios ofrecidos por el servidor Ministerio al usuario.

Cabe resaltar que todos las interfaces son necesarias, sin embargo en este documento se resaltan las interfaces que contienen los servicios ofrecidos. Finalmente las operaciones que se muestran en las figuras 4, 6,10 y 12 son implementadas por clases impl dentro del paquete “**servidor**”.

# 6. Descripción del proceso de compilación

Para los servidores: se ubica dentro de la carpeta src del proyecto, lanza la consola desde esa ubicación y escribe en consola javac –d ..\bin servidor\\*.java

Para los clientes: se ubica dentro de la carpeta src del proyecto, lanza la consola desde esa ubicación y escribe en consola javac –d ..\bin cliente\\*.java

Para los soportes: se ubica dentro de la carpeta src del proyecto, lanza la consola desde esa ubicación y escribe en consola javac -d ../bin sop\_corba/\*.java

De esta forma se generan los .class en la carpeta bin del proyecto.

# 7. Descripción del proceso de Ejecución

Para el servidor: inicialmente lanza la consola desde cualquier carpeta, luego se utilizara el comando “**orbd –ORBInitialHost dir\_IP -ORBInitialPort N\_Puerto**”, donde la dir\_IP es la dirección ip donde se registrara el servidor al igual que el N\_Puerto el puerto que utilizara, para el ORBD el cual es un servicio de nombrado, luego se debe lanzar nuevamente una consola en la direccion \bin del proyecto, en la nueva consola se escribe el comando “**java presentacion.ControladorServidor**”, a continuación se mostrara un menú en consola que le permitirá indicar los valores de IP y de puerto al que se desee conectar con el **ORBD**, al fijar los valores selección la opción 3 para correr el servidor, el mensaje en consola notificara si el objeto remoto fue instanciado y enlazado exitosamente.

Para los Clientes: se debe abrir el proyecto del cliente que se desee utilizar en NetBeans (problemas de librerías al momento de realizar las interfaces graficas), luego presionar la tecla F6 o presionar el icono de ejecutar, lo siguiente que aparecerá será el formulario con la dirección en el que se debe realizar primero la invocación al objeto remoto mediante la dirección ip(en el campo dirección p), y el puerto(en el campo puerto),

Por defecto la url tiene los valores de ip del servidor son localhost con puerto 1099.

# 8. Descripción del Manejo del Instalador.

El proyecto contiene instaladores ejecutables:

EN EL CASO DE LOS CLIENTES: Se distribuyen en dos carpetas como se muestra en la figura 13, en la cual Cliente\_Ministerio contiene tanto el cliente como el administrador del Ministerio según sea su en ingreso en el login. Por otro lado Cliente\_Biblioteca contiene el cliente administrador de la Biblioteca. Para lanzar estos clientes solo es necesario arrancar con doble click los archivos .jar:

* Cliente\_MinCultura.jar
* Cliente\_Biblioteca.jar

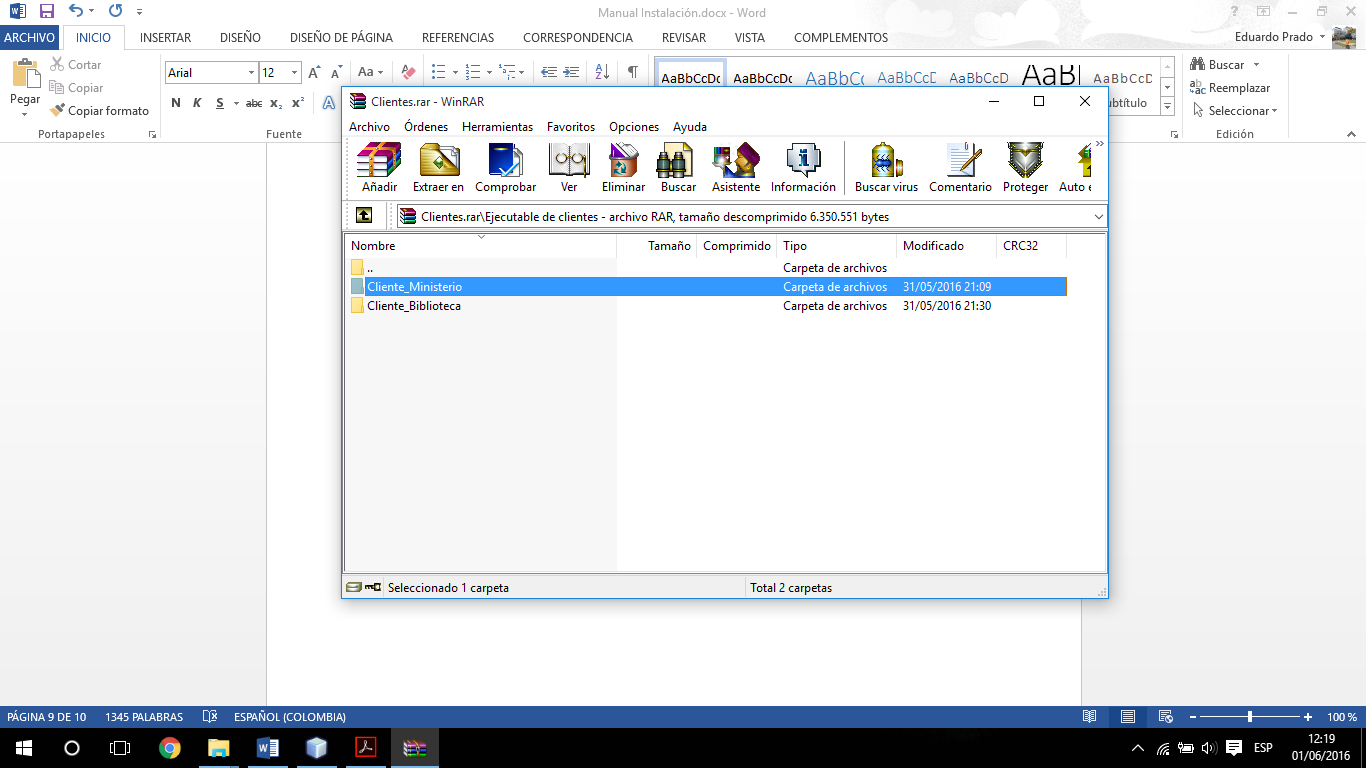


Figura 13. Carpetas Cliente 1

EN EL CASO DEL SERVIDOR: Inicialmente se debe iniciar el servicio de nombrado **ORBD** como se explica en la sección anterior de este documento para cada servidor si es en distintos equipos o para solo una vez si solo es un solo equipo, luego de lanzado el ORBD la distribución de servidores se muestra a continuación en la figura 14.

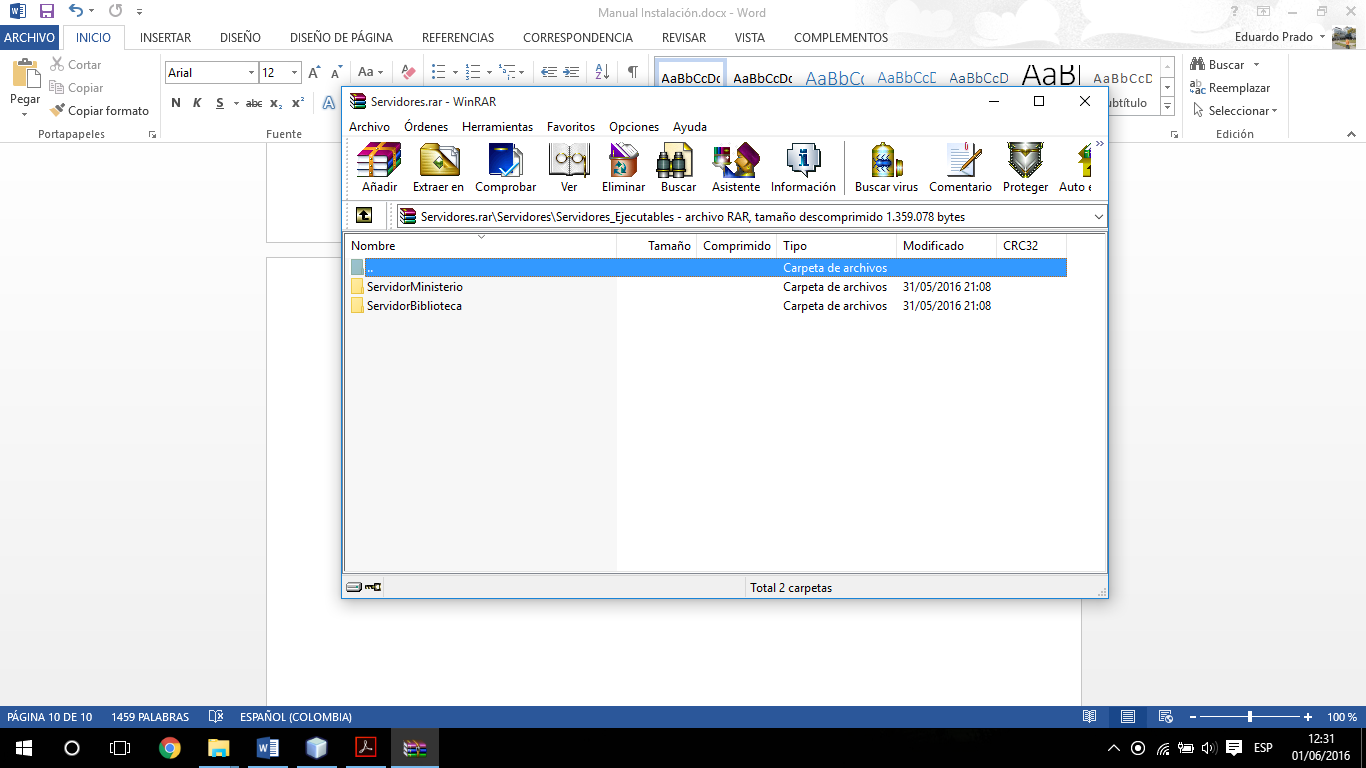


Figura 14. Carpetas Servidores

En la carpeta ServidorBiblioteca se encuentran los archivos necesarios para iniciar el servidor. Este servidor se inicia lanzado la consola del sistema desde este directorio y luego se utiliza el comando “java –jar ServidorBiblioteca.jar”. Luego tendrá un pequeño menú de opciones en consola para modificar la IP y el Puerto donde se encuentra el servicio de nombrado, una opción para correr el servidor y por ultimo una opción para salir.

En la carpeta ServidorMinisterio se encuentran los archivos necesarios para iniciar el servidor. Este servidor se inicia lanzado la consola del sistema desde este directorio y luego se utiliza el comando “java –jar ServidorMinisterio.jar”. Luego de lanzado el servidor mostrara un pequeño menú de consola donde podrá modificar la IP y el Puerto para conectarse al servicio de nombrado, además contara otra opción para modificar la IP y el puerto para conectarse con un servidor Biblioteca, la cuarta opción corre el servidor luego de fijados los valores y por ultimo cuenta con la opción salir para cerrar la aplicación.

Nota: es importante tener en cuenta que para lanzar el servidor Ministerio se debe lanzar antes un servidor Biblioteca para el correcto funcionamiento del mismo.

# Referencias

norfiPc. (s.f.). *norfipc.com*. Obtenido de norfipc.com: https://goo.gl/DCuuk8