<

Raúl Sanchez Rico

Desarrollo de sistemas distribuidos  
4CV2

DESARROLLO DE UN CLIENTE PARA UN SERVICIO WEB RESET

Tarea 8

**Introducción**

Cada alumno deberá desarrollar un programa Java consola que consuma el servicio web que creamos en la tarea 7. El programa deberá desplegar el siguiente menú:

a. Alta usuario  
b. Consulta usuario

c. Borra usuario

d. Borra todos los usuarios  
e. Salir

Opción: \_

La opción "Alta usuario" leerá del teclado el email, el nombre del usuario, el apellido paterno, el apellido materno, la fecha de nacimiento, el teléfono y el género ("M" o "F"). Entonces se invocará el método "alta" del servicio web. Se deberá desplegar "OK" si se pudo dar de alta el usuario, o bien, el mensaje de error que regresa el servicio web.

La opción "Consulta usuario" leerá del teclado el email de un usuario previamente dado de alta. Entonces se invocará el método "consulta" del servicio web. Si el usuario existe se desplegará en pantalla el nombre del usuario, el apellido paterno, el apellido materno, la fecha de nacimiento, el teléfono y el género.

La opción "Borra usuario" leerá del teclado el email de un usuarios previamente dado de alta. Entonces se invocará el método "borra" del servicio web. Se deberá desplegar "OK" si se pudo borrar el usuario

La opción "Borra todos los usuarios" invocará el método "borrar\_usuarios" creado en la actividad individual de la clase pasada. Se deberá desplegar "OK" si se pudo borrar el usuario

La opción "Salir" terminará el programa.

**Desarrollo**

En la ilustración 1 se muestra la maquina virtual hecha en Azure, la cual ya se le ha agregado todo lo mínimo para el funcionamiento, esto implica instalar Java, MySQL, hacer la base datos, así como sus tablas.

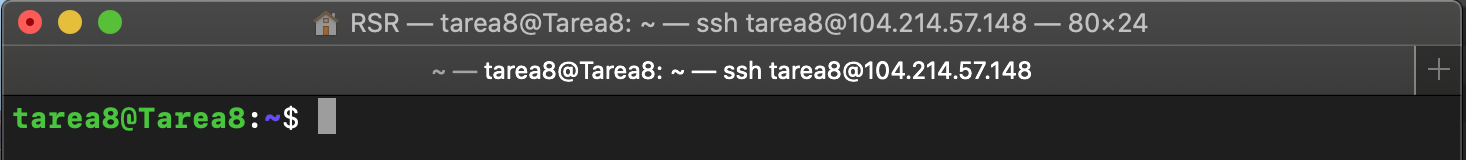


Ilustración 1 Maquina Virtual Tarea8

Posteriormente se hace la compilación de los archivos del cliente donde vamos a mostrar el menú y hacer sus peticiones al servidor, esto se muestra en la ilustración 2. Se tienen 3 archivos los cuales son: Cliente.java, el cual implementa la “interfaz”, HTTPConexiones.java, quien se encarga de realizar las peticiones a nuestro servidor y por último Usuario.java, quien es nuestro objeto.

Una vez terminado se procede a ejecutar la clase Cliente, esto se debe tener en cuenta que se hace como se ha hecho en clase usando el “.jar”. Nos pedirá la IP a la que nos conectaremos, y una vez hecho esto nos mostrará el menú de opciones (Ilustración 2).

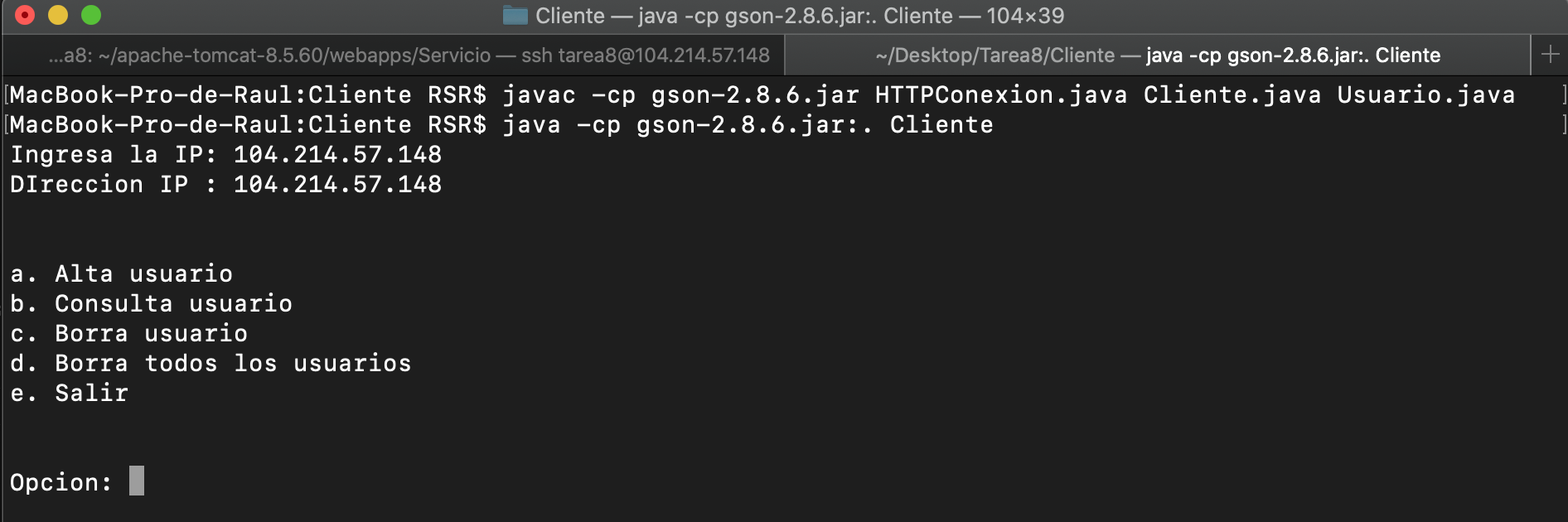


Ilustración 2 Compilación/Ejecución – Menú Principal

En las siguientes ilustraciones se pasará a mostrar cada una de las opciones, para esto se crearán, consultaran, eliminaran 2 usuarios y por ultimo se procederá a salir de la aplicación.

1. **Alta Usuario**

**Usuario (1)**

****

Ilustración 3 Alta Usuario 1

**Usuario (2)**

****

Ilustración 4 Alta Usuario 2

**Usuario (3)**

****

Ilustración 7 Alta Usuario 3

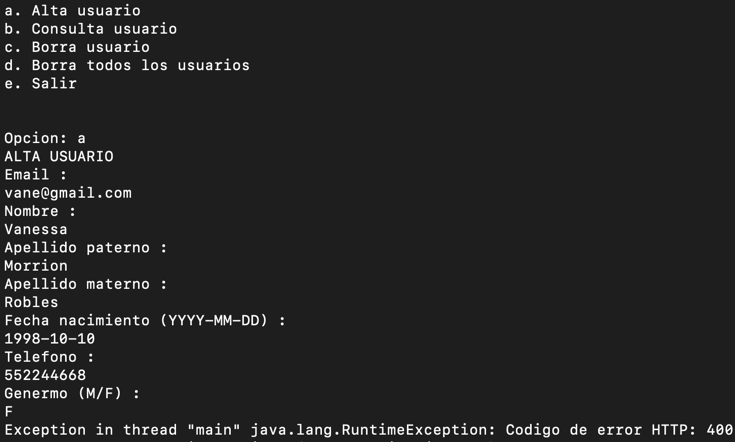
****

Ilustración 8 Alta usuario error HTTP - 400

1. **Consulta Usuario**

**Usuario (1)**

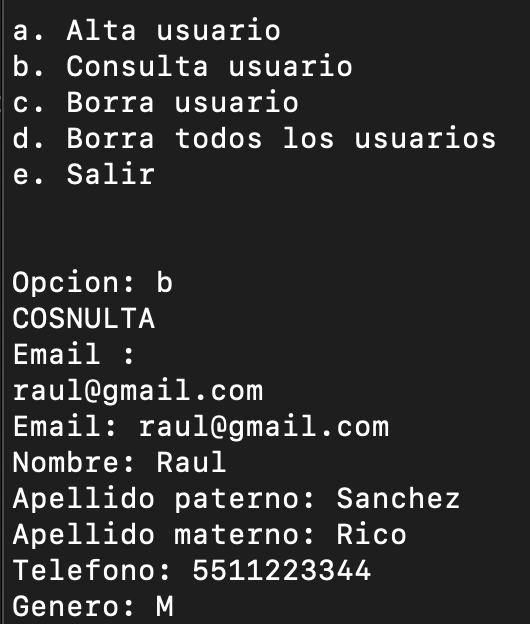
****

Ilustración 5 Consulta Usuario 1

**Usuario (2)**

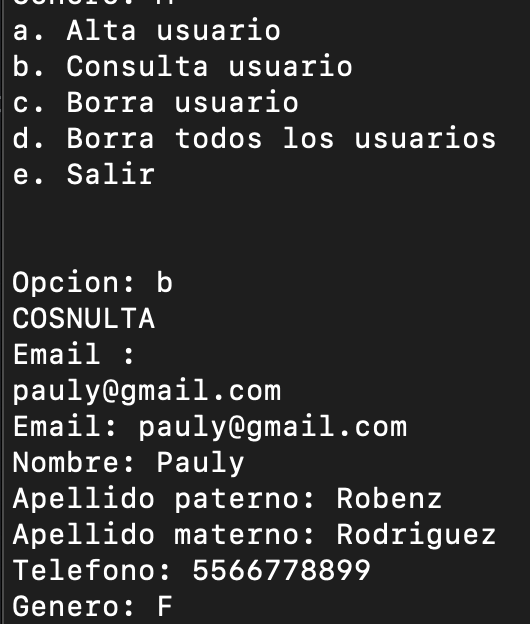
****

Ilustración 6 Consulta Usuario 2

1. **Borrar Usuario**

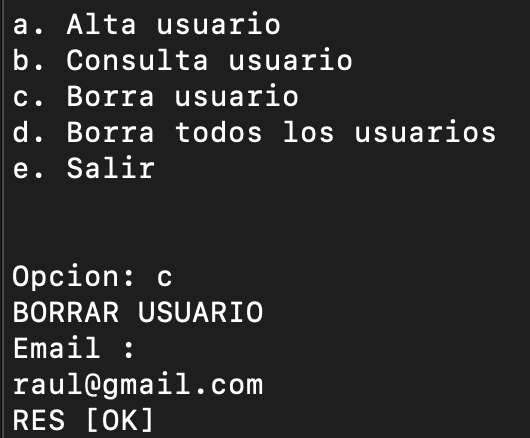
****

Ilustración 9 Borrar usuario 1

****

Ilustración 10 Consulta usuario que no existe

Hay que notar que la ilustración 8 nos muestra un mensaje HTTP: 400, ya que el usuario con email [raul@gmail.com](mailto:raul@gmail.com) ya ha sido borrado.

1. **Borrar Usuarios**

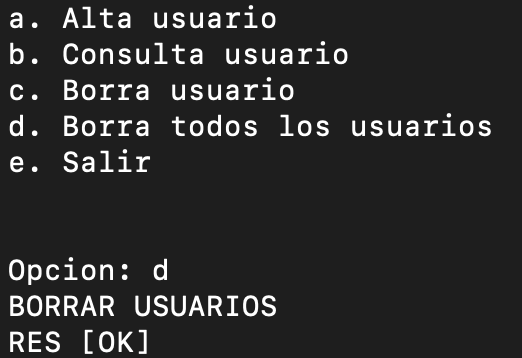
****

Ilustración 11 Borrar Usuarios

1. **Salir**

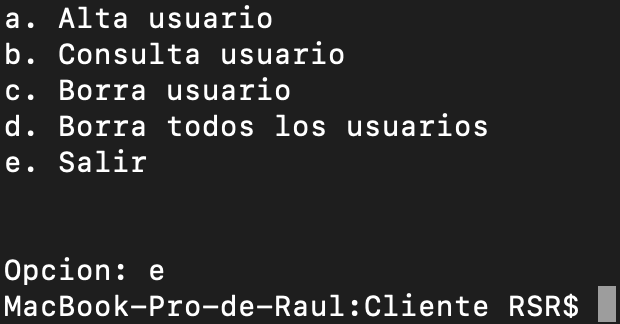
****

Ilustración 12 Salir del programa

**Conclusiones**

El desarrollo de esta practica significa un gran avance ya que se esta entendió mas los conceptos de un servicio web, utilizando JSON, el cual es de gran utilidad no solo para esto si no como reflexión vemos su gran uso, así como la facilidad de este para poder entender estos temas. Aunado a que hace que los proyectos que se desarrollen de esta forma sean mas confiable, rápidos y podamos encontrar errores mas fáciles como ha pasado en esta tarea.