

**INSTITUTO POLITÉCNICO NACIONAL**  
**Escuela Superior de Cómputo**

**ESCOM**

*Practica*

**“RMI”**

*Alumnos*

**Arcos Ayala Jonathan**  
**Zepeda Ibarra Allan Ulises**

*Grupo*  
**4CV3**

## 0.1 Introduccion

En una arquitectura de objetos distribuidos, los componentes del sistema con objetos que proporcionan y requieren un conjunto de servicios. Estos hacen llamadas a los servicios de otros objetos sin hacer distincion lógica entre un cliente y un servidor. Los objetos pueden distribuirse a través de varias computadoras en una red.

RMI (Java Remote Method Invocation) es un mecanismo ofrecido por Java para invocar un método de manera remota. Forma parte del entorno estándar de ejecución de Java y proporciona un mecanismo simple para la comunicación de servidores en aplicaciones distribuidas basadas exclusivamente en Java.

RMI se caracteriza por la facilidad de su uso en la programación por estar específicamente diseñado para Java; proporciona paso de objetos por referencia, recolección de basura distribuida y paso de tipos arbitrarios.

A través de RMI, un programa Java puede exportar un objeto, con lo que dicho objeto estará accesible a través de la red y el programa permanece a la espera de peticiones en un puerto TCP. A partir de ese momento, un cliente puede conectarse e invocar los métodos proporcionados por el objeto.

## 0.2 Desarrollo

En un primer momento nos dedicamos a la creacion de la interfase grafica que nos permitiera hacer las consultas y poder verlas con facilidad, esto fue construido con Java swing en eclipse el codigo se muestra a continuacion:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package p4_rmi_cliente_4cv3;

import com.dsd.interfaces.DireccionInterfaces;
import com.dsd.interfaces.PersonalInterfaces;
import com.dsd.interfaces.exception.ExceptionPersona;
import java.io.IOException;
import static java.lang.System.exit;
import java.rmi.NotBoundException;
import java.rmi.RemoteException;
import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;
import java.util.*;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;

/**
```

```

*
* @author allan
*/
public class cliente extends javax.swing.JFrame {
    /**
     *
     */
    private static final long serialVersionUID = 1L;
    static PersonalInterfaces persona;
    static DireccionInterfaces dir;
    static boolean cargarCompleta=true;
    static boolean altaCompleta=true;
    static boolean bajaCompleta=true;
    static boolean cambioCompleta=true;
    static boolean borrarCompleta=true;
    static boolean compraCompleta= true;
    static HashMap<String , String> personal= new HashMap<>();
    static Exception e=null;
    /**
     * Creates new form cliente
     */
    public cliente() {

        initComponents();
        TNom.setEnabled( false );
        TApp.setEnabled( false );
        TApm.setEnabled( false );
        TCorreo.setEditable( false );
        TDat.setEnabled( false );
        TCalle.setEnabled( false );
        TCol.setEnabled( false );
        TEnt.setEnabled( false );
        TCiu.setEnabled( false );
        TCp.setEnabled( false );
    }

    /**
     * This method is called from within the constructor to initialize the form
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */

    // <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN:
    private void initComponents() {

        dlgConec = new javax.swing.JDialog();
        TIp = new javax.swing.JTextField();
        jLabel11 = new javax.swing.JLabel();
        bynCon = new javax.swing.JButton();
        btnCANI = new javax.swing.JButton();

```

```

jScrollPane1 = new javax.swing.JScrollPane();
LArt = new javax.swing.JList<>();
jPanel1 = new javax.swing.JPanel();
jLabel1 = new javax.swing.JLabel();
TNom = new javax.swing.JTextField();
jLabel2 = new javax.swing.JLabel();
TApp = new javax.swing.JTextField();
jLabel3 = new javax.swing.JLabel();
TApm = new javax.swing.JTextField();
jLNac = new javax.swing.JLabel();
jLabel4 = new javax.swing.JLabel();
TCalle = new javax.swing.JTextField();
jLabel5 = new javax.swing.JLabel();
TCol = new javax.swing.JTextField();
jLabel6 = new javax.swing.JLabel();
jLabel7 = new javax.swing.JLabel();
TCiu = new javax.swing.JTextField();
jLabel8 = new javax.swing.JLabel();
TCp = new javax.swing.JTextField();
TCorreo = new javax.swing.JTextField();
jLabel9 = new javax.swing.JLabel();
BNuevo = new javax.swing.JButton();
BActu = new javax.swing.JButton();
BBorrar = new javax.swing.JButton();
TEnt = new javax.swing.JComboBox<>();
TDat = new com.toedter.calendar.JDateChooser();
btnAgCar = new javax.swing.JButton();
btnSalir = new javax.swing.JButton();
btnIP = new javax.swing.JButton();
jButton4 = new javax.swing.JButton();

dlgConec.setSize(new java.awt.Dimension(400, 150));

jLabel11.setText(" Ip");

bynCon.setText(" Conectar");
bynCon.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        bynConActionPerformed(evt);
    }
});

btnCANI.setText(" Cancelar");
btnCANI.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnCANIActionPerformed(evt);
    }
});

javax.swing.GroupLayout dlgConecLayout = new javax.swing.GroupLayout(dlgConec);

```

```

dlgConec.getContentPane().setLayout(dlgConecLayout);
dlgConecLayout.setHorizontalGroup(
    dlgConecLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(dlgConecLayout.createSequentialGroup()
            .addGap(79, 79, 79)
            .addGroup(dlgConecLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(dlgConecLayout.createSequentialGroup()
                    .addComponent(bynCon)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .addComponent(btnCAInI))
                .addGroup(dlgConecLayout.createSequentialGroup()
                    .addComponent(jLabel11)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .addComponent(Tip, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addContainerGap(92, Short.MAX_VALUE))
        );
dlgConecLayout.setVerticalGroup(
    dlgConecLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(dlgConecLayout.createSequentialGroup()
            .addGap(38, 38, 38)
            .addGroup(dlgConecLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(Tip, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel11))
            .addGap(26, 26, 26)
            .addGroup(dlgConecLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(bynCon)
                .addComponent(btnCAInI))
            .addContainerGap(47, Short.MAX_VALUE))
        );

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setMinimumSize(new java.awt.Dimension(800, 600));

LArt.addListSelectionListener(new javax.swing.event.ListSelectionListener() {
    public void valueChanged(javax.swing.event.ListSelectionEvent evt) {
        LArtValueChanged(evt);
    }
});
jScrollPane1.setViewportViewView(LArt);

jLabel1.setText("Nombre");

jLabel2.setText("Primer Apellido");

jLabel3.setText("Segundo Apellido");

jLNac.setText("Fecha de Nacimiento");

jLabel4.setText("Calle");

```

```

jLabel5.setText(" Colonia ");

jLabel6.setText(" Entidad ");

jLabel7.setText(" Ciudad ");

jLabel8.setText("Codigo Postal");

TCorreo.setEnabled( false );

jLabel9.setText(" Correo ");

BNuevo.setText(" Nuevo ");
BNuevo.setEnabled( false );
BNuevo.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        BNuevoActionPerformed(evt);
    }
});

BActu.setText(" Actualizar ");
BActu.setEnabled( false );
BActu.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        BActuActionPerformed(evt);
    }
});

BBorrar.setText(" Borrar ");
BBorrar.setEnabled( false );
BBorrar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        BBorrarActionPerformed(evt);
    }
});

TEnt.setModel(new javax.swing.DefaultComboBoxModel<>(new String [] {"--",
    javax.swing.GroupLayout jPanellLayout = new javax.swing.GroupLayout(jPa
jPanell.setLayout(jPanellLayout);
jPanellLayout.setHorizontalGroup(
    jPanellLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.L
    .addGroup(jPanellLayout.createSequentialGroup()
        .addGroup(jPanellLayout.createParallelGroup(javax.swing.GroupLayout.A
            .addGap(20, 20, 20)
            .addGroup(jPanellLayout.createParallelGroup(javax.swing.GroupLayout.A
                .addComponent(jLNac, javax.swing.GroupLayout.DEFAULT_SIZE,
                .addGroup(jPanellLayout.createParallelGroup(javax.swing.GroupLayout.A
                    .addComponent(TApm, javax.swing.GroupLayout.Alignment.L
                    .addComponent(jLabel3, javax.swing.GroupLayout.Alignment
                    .addComponent(jLabel2, javax.swing.GroupLayout.Alignment

```



```

        .addComponent(jLabel6)
        .addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addComponent(TApp, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, true)
                .addGap(18, 18, 18))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addComponent(BBorrar)
                .addGap(18, 18, 18))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addComponent(TEnt, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, true)
                .addGap(18, 18, 18))
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel7)
                .addComponent(jLabel3))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addComponent(TCiu, javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, true)
                .addComponent(TApm))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(18, 18, 18))
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel8)
                .addComponent(jLNac))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addComponent(TCp, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, true)
                .addComponent(TDat, javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE, true)
                .addGap(144, 144, 144))
        );

btnAgCar.setText(" Actualizar ");
btnAgCar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnAgCarActionPerformed(evt);
    }
});

btnSalir.setText(" Salir ");
btnSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnSalirActionPerformed(evt);
    }
});

btnIP.setText(" Conexion ");
btnIP.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnIPActionPerformed(evt);
    }
});

```



```

jButton4.setText(" Habilitar ");
jButton4.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jButton4ActionPerformed(evt);
    }
});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane()
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jButton4)
            .addGroup(layout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addComponent(btnAgCar)
                .addGap(60, 60, 60)
                .addComponent(btnSalir)
                .addGap(55, 55, 55)
                .addComponent(btnIP))
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addContainerGap(18, true));
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addContainerGap()
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addGroup(layout.createSequentialGroup()
                .addGap(18, 18, 18)
                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(btnAgCar)
                    .addComponent(btnSalir)
                    .addComponent(btnIP)
                    .addComponent(jButton4))
                .addGap(30, 30, 30))
            .addGap(18, 18, 18))
        .addContainerGap(18, true));

pack();
} // </editor-fold> //GEN-END: initComponents

private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) { //GEN-
// TODO add your handling code here:
exit(0);
}

```

```

} //GEN-LAST:event_btnSalirActionPerformed

private void btnAgCarActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btnAgCarActionPerformed
    try {
        // TODO add your handling code here:
        listarPersonal();
    } catch (RemoteException | ExceptionPersona ex) {
        Logger.getLogger(cliente.class.getName()).log(Level.SEVERE, null, ex);
    }
} //GEN-LAST:event_btnAgCarActionPerformed

private void btnIPActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btnIPActionPerformed
    // TODO add your handling code here:
    dlgConec.setVisible(true);
} //GEN-LAST:event_btnIPActionPerformed
.
.
.
private void btnCANIAActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_btnCANIAActionPerformed
    // TODO add your handling code here:
    dlgConec.setVisible(false);
} //GEN-LAST:event_btnCANIAActionPerformed

private void LArtValueChanged(javax.swing.event.ListSelectionEvent evt) { //GEN-FIRST:event_LArtValueChanged
    try {
        // TODO add your handling code here:
        String n = personal.get(LArt.getSelectedValue());
        Map<String, Object> auxP = persona.getPersona(n);
        TNom.setText((String)auxP.get("nombre"));
        TCorreo.setText((String)auxP.get("correo"));
        TApp.setText((String)auxP.get("App"));
        TApm.setText((String)auxP.get("Apm"));
        TDat.setDate((Date)auxP.get("nacimiento"));
        TCalle.setText((String)auxP.get("calle"));
        TCol.setText((String)auxP.get("colonia"));
        TEnt.setSelectedItem((String)auxP.get("entidad"));
        TCiu.setText((String)auxP.get("ciudad"));
        TCp.setText("cp");

    } catch (RemoteException | ExceptionPersona ex) {
        Logger.getLogger(cliente.class.getName()).log(Level.SEVERE, null, ex);
    }
} //GEN-LAST:event_LArtValueChanged

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_jButton4ActionPerformed
    desbloqueo();
} //GEN-LAST:event_jButton4ActionPerformed

private void BNuevoActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_BTNuevoActionPerformed

```

```
String correo = TCorreo.getText();
String ent = (String)TEnt.getSelectedItem();
if(correo.trim().isEmpty() || ent.trim().equals("---")){
    JOptionPane.showMessageDialog(rootPane, "*Ingresar un correo Valido");
}else{
    try {
        Map<String,Object> per = new HashMap<String , Object >();
        per.put("correo", correo);
        per.put("nombre", TNom.getText());
        per.put("app", TApp.getText());
        per.put("apm", TApm.getText());
        per.put("nacimiento", TDat.getDate());
        per.put("calle", TCalle.getText());
        per.put("colonia", TCol.getText());
        per.put("entidad", ent);
        per.put("ciudad", TCiu.getText());
        per.put("cp", TCp.getText());
        persona.altaPersona(per);
        listarPersonal();
        bloqueo();
        limpiar();
    } catch (RemoteException | ExceptionPersona ex) {
        JOptionPane.showMessageDialog(rootPane, "Error en el Alta de pe");
        Logger.getLogger(cliente.class.getName()).log(Level.SEVERE, null, ex);
    }
}

}

//GEN-LAST:event_BNuevoActionPerformed

private void BActuActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:
String correo = TCorreo.getText();
String ent = (String)TEnt.getSelectedItem();
if(correo.trim().isEmpty() || ent.trim().equals("---")){
    JOptionPane.showMessageDialog(rootPane, "*Ingresar un correo Valido");
}else{
    try {
        Map<String,Object> per = new HashMap<String , Object >();
        per.put("correo", correo);
        per.put("nombre", TNom.getText());
        per.put("app", TApp.getText());
        per.put("apm", TApm.getText());
        per.put("nacimiento", TDat.getDate());
        per.put("calle", TCalle.getText());
        per.put("colonia", TCol.getText());
        per.put("entidad", ent);
        per.put("ciudad", TCiu.getText());
        per.put("cp", TCp.getText());
        persona.cambioPersona(per);
        listarPersonal();
        bloqueo();
```

```
private void BActuActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST_CLICK
    String correo = TCorreo.getText();
    String ent = (String)TEnt.getSelectedItemAt(0);
    if(correo.trim().isEmpty() || ent.trim().equals("---")){
        JOptionPane.showMessageDialog(rootPane, "*Ingresar un correo Valido");
    }else{
        try {
            Map<String, Object> per = new HashMap<String, Object>();
            per.put("correo", correo);
            per.put("nombre", TNom.getText());
            per.put("app", TApp.getText());
            per.put("apm", TApn.getText());
            per.put("nacimiento", TDat.getDate());
            per.put("calle", TCalle.getText());
            per.put("colonia", TCol.getText());
            per.put("entidad", ent);
            per.put("ciudad", TCiu.getText());
            per.put("cp", TCp.getText());
            persona.cambioPersona(per);
            listarPersonal();
            bloqueo();
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(rootPane, ex.getMessage());
        }
    }
} //GEN-LAST_CLICK
```

```

        limpiar();
    } catch (RemoteException | ExceptionPersona ex) {
        JOptionPane.showMessageDialog(rootPane, "Error en la Actualizaci
        Logger.getLogger(cliente.class.getName()).log(Level.SEVERE, nul
    }
}
} //GEN-LAST:event_BActuActionPerformed

private void BBorrarActionPerformed(java.awt.event.ActionEvent evt) { //GEN-
    String correo = TCorreo.getText();
    if (correo.trim().isEmpty()) {
        JOptionPane.showMessageDialog(rootPane, "*Ingresar un correo Valido
    } else {
        try {
            persona.bajaPersona(correo);
            listarPersonal();
            bloqueo();
            limpiar();
        } catch (RemoteException | ExceptionPersona ex) {
            JOptionPane.showMessageDialog(rootPane, "Error en el Alta de pe
            Logger.getLogger(cliente.class.getName()).log(Level.SEVERE, nul
        }
    }
} //GEN-LAST:event_BBorrarActionPerformed

private void bloqueo(){
    TNom.setEditable(false);
    TNom.setEnabled(false);
    TCorreo.setEditable(false);
    TCorreo.setEnabled(false);
    TApp.setEditable(false);
    TApp.setEnabled(false);
    TApm.setEditable(false);
    TApm.setEnabled(false);
    TDat.setEnabled(false);
    TCalle.setEditable(false);
    TCalle.setEnabled(false);
    TCol.setEditable(false);
    TCol.setEnabled(false);
    TEnt.setEditable(false);
    TEnt.setEnabled(false);
    TCiu.setEditable(false);
    TCiu.setEnabled(false);
    TCp.setEditable(false);
    TCp.setEnabled(false);
    BNuevo.setEnabled(false);
    BActu.setEnabled(false);
    BBorrar.setEnabled(false);
}

```

```

private void desbloqueo(){
    TNom.setEditable(true);
    TNom.setEnabled(true);
    TCorreo.setEditable(true);
    TCorreo.setEnabled(true);
    TApp.setEditable(true);
    TApp.setEnabled(true);
    TApm.setEditable(true);
    TApm.setEnabled(true);
    TDat.setEnabled(true);
    TCalle.setEditable(true);
    TCalle.setEnabled(true);
    TCol.setEditable(true);
    TCol.setEnabled(true);
    TEnt.setEditable(true);
    TEnt.setEnabled(true);
    TCiu.setEditable(true);
    TCiu.setEnabled(true);
    TCp.setEditable(true);
    TCp.setEnabled(true);
    BNuevo.setEnabled(true);
    BActu.setEnabled(true);
    BBorrar.setEnabled(true);
}

private void listarPersonal() throws RemoteException, ExceptionPersona{

    Vector<String> nom = new Vector<String>();
    List<Map> aux = persona.getListPersona();
    for (Map next : aux) {
        nom.add((String)next.get("nombre"));
        personal.put((String)next.get("nombre"),(String)next.get("correo"))
    }
    LArt.removeAll();
    LArt.setListData(nom);
}

private void limpiar(){
    TNom.setText("");
    TCorreo.setText("");
    TApp.setText("");
    TApm.setText("");
    TDat.setDate(new Date());
    TCalle.setText("");
    TCol.setText("");
    TEnt.setSelectedIndex(0);
    TCiu.setText("");
    TCp.setText("");
}
/**

```

```

    * @param args the command line arguments
    */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting co
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the
        * For details see http://download.oracle.com/javase/tutorial/uiswing/l
        */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
        } catch (ClassNotFoundException | InstantiationException | IllegalAccessException
            java.util.logging.Logger.getLogger(cliente.class.getName()).log(java
        }
        //</editor-fold>

        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new cliente().setVisible(true);
            }
        });
    }

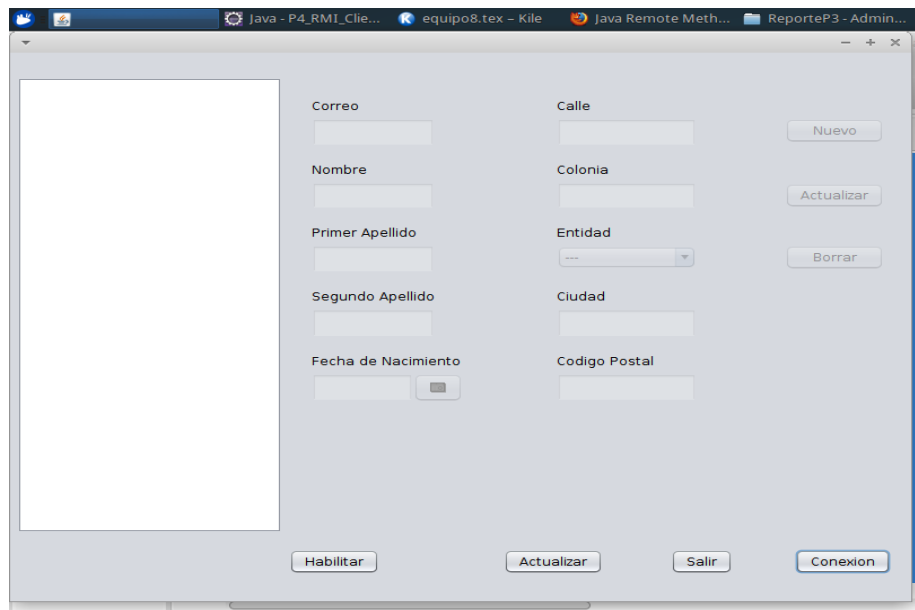
    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton BActu;
    private javax.swing.JButton BBorrar;
    private javax.swing.JButton BNuevo;
    private javax.swing.JList<String> LArt;
    private javax.swing.JTextField TApm;
    private javax.swing.JTextField TApp;
    private javax.swing.JTextField TCalle;
    private javax.swing.JTextField TCiu;
    private javax.swing.JTextField TCol;
    private javax.swing.JTextField TCorreo;
    private javax.swing.JTextField TCp;
    private com.toedter.calendar.JDateChooser TDat;
    private javax.swing.JComboBox<String> TEnt;
    private javax.swing.JTextField TIp;
    private javax.swing.JTextField TNom;
    private javax.swing.JButton btnAgCar;
    private javax.swing.JButton btnCAnI;
    private javax.swing.JButton btnIP;
    private javax.swing.JButton btnSalir;

```

```

private javax.swing.JButton bynCon;
private javax.swing.JDialog dlgConec;
private javax.swing.JButton jButton4;
private javax.swing.JLabel jLNac;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel11;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JLabel jLabel8;
private javax.swing.JLabel jLabel9;
private javax.swing.JPanel jPanel1;
private javax.swing.JScrollPane jScrollPane1;
// End of variables declaration//GEN-END:variables
}

```



Dentro de la funcionalidad que tenemos, hay que pedir la direccion IP a la que nos conectamos esto, para hacerlo de manera manual.

```

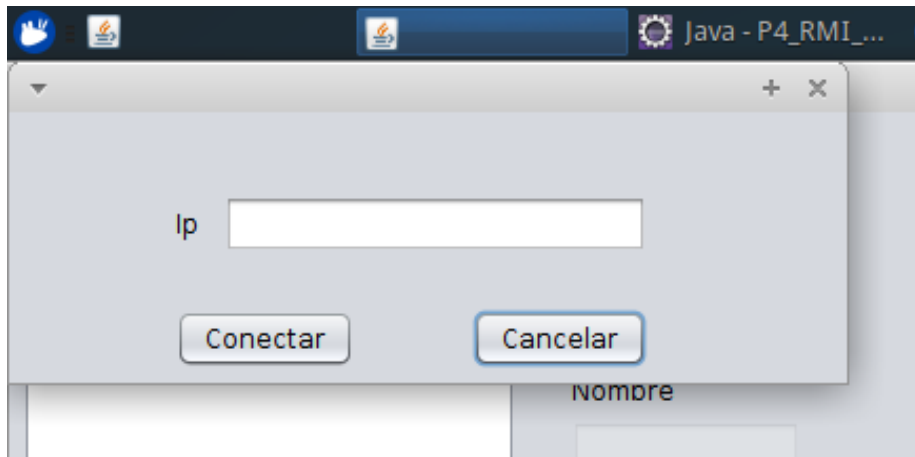
private void bynConActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST
    try {
        System.setProperty("java.security.policy", "seguridad.policy");
        System.setSecurityManager(new SecurityManager());
        final Registry registry = LocateRegistry.getRegistry(TIp.getText(),
            Registry.REGISTRY_PORT);
        System.out.println(TIp.getText());
        persona = (PersonalInterfaces) registry.lookup("rmi://localhost/Persona");
    }
}

```

```

dir = (DireccionInterfaces)registry.lookup("rmi://localhost/Direccion");
listarPersonal();
} catch (IOException ex) {
    JOptionPane.showMessageDialog(rootPane, "Error de Conexion");
} catch (NotBoundException | ExceptionPersona ex) {
    Logger.getLogger(cliente.class.getName()).log(Level.SEVERE, null, ex);
}
dlgConec.setVisible(false);
} //GEN-LAST:event_bynConActionPerformed

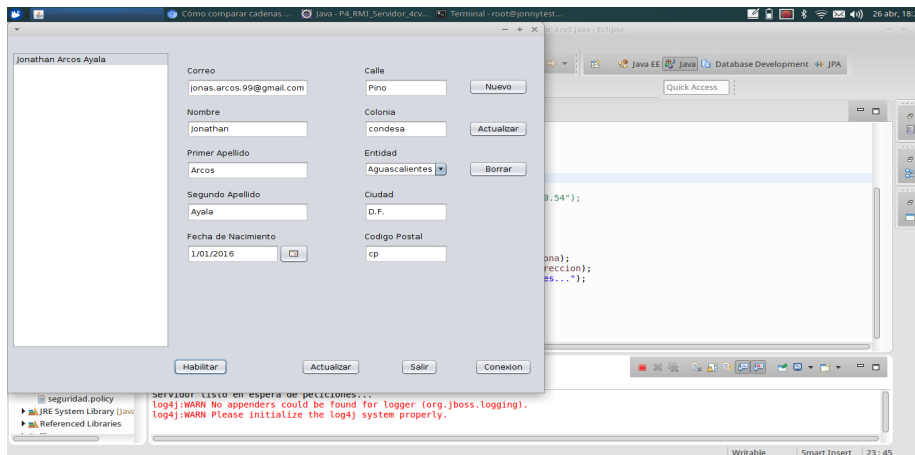
```



Todas las Conexiones al servidores y la alta de la interfase por la que se comunica la aplicacion esta en el metodo de arriba don de se hace el llamado a los metodos RMI correspondientes.

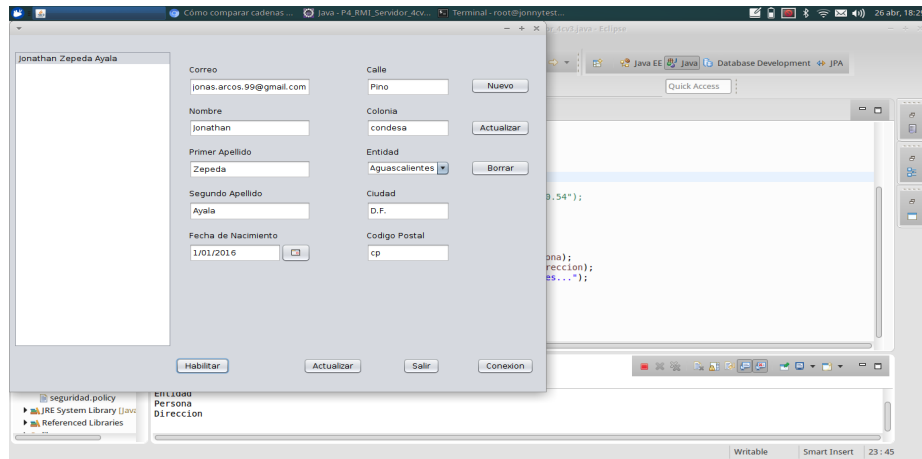
### 0.3 Pruebas

consultas:

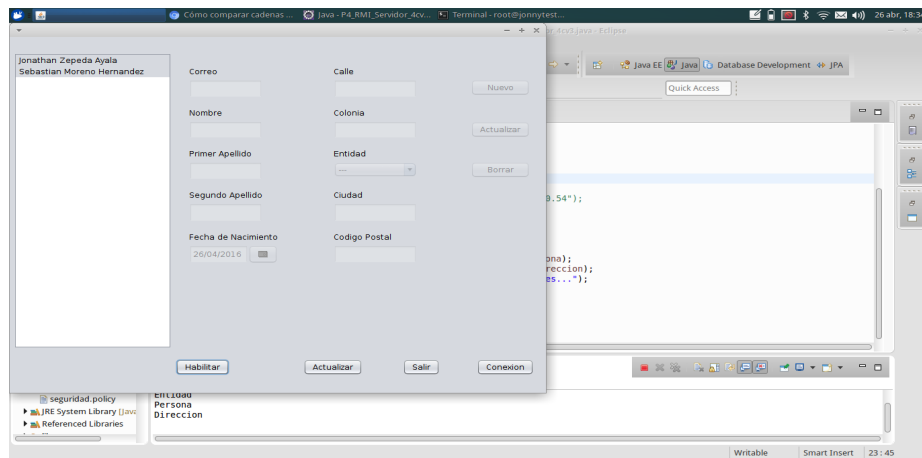
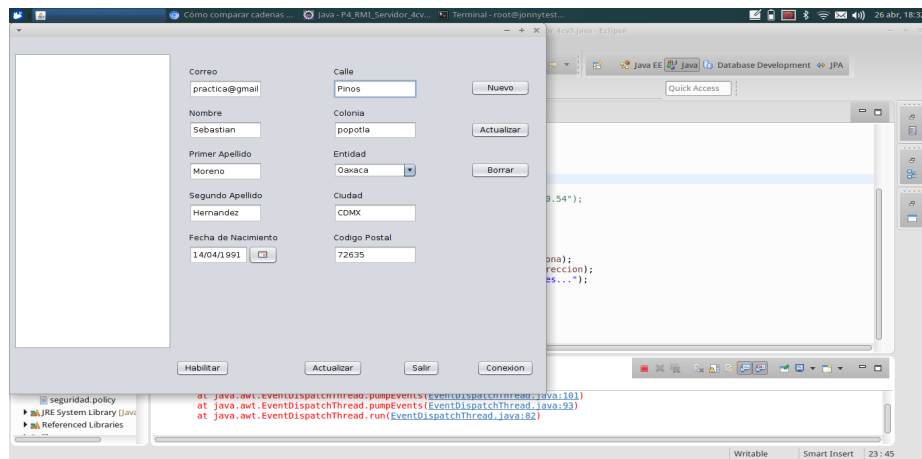




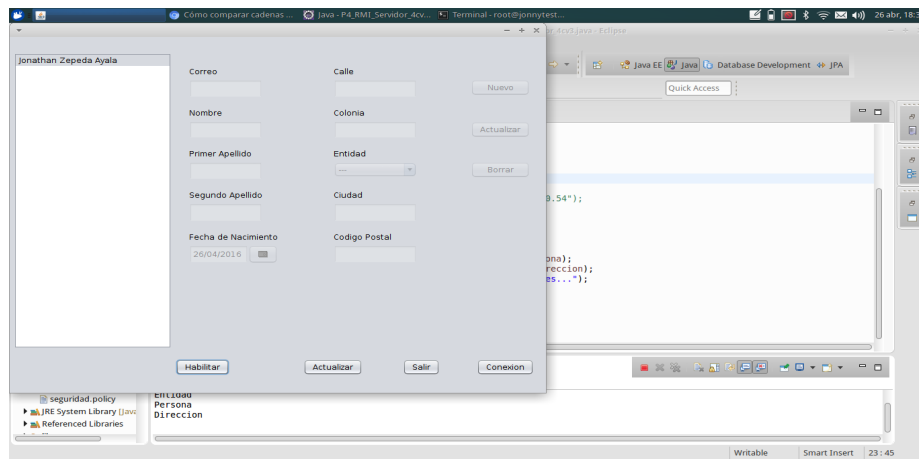
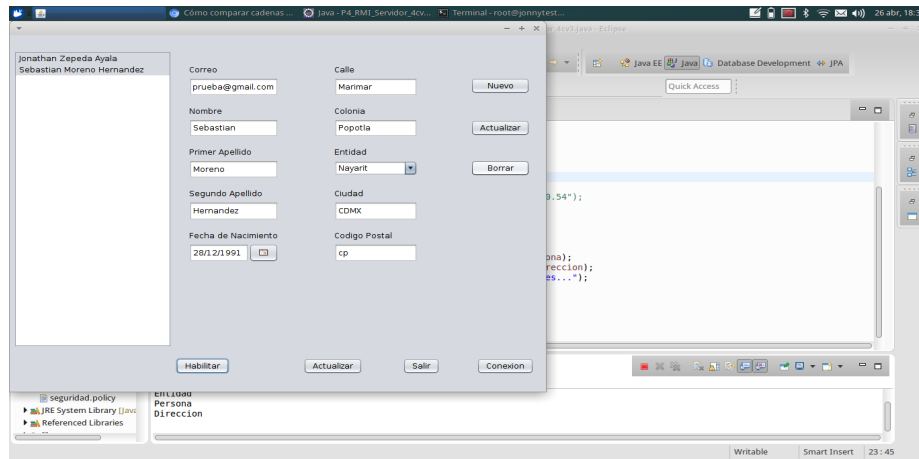
Editar:



Alta:



Borrar:



## 0.4 Conclusiones

Arcos Ayala Jonathan: La comunicación entre procesos en java se lleva acabo de una manera transparente gracias a RMI ya que nos ayuda al paso de información por objetos. Si tenemos una plataforma java, la mejor manera de comunicar procesor en una red es por medio de RMI.

Zepeda Ibarra Allan Ulises: Ya con la interface RMI en java es facil la peticion de metodo de manera remota, y de verdad pareciera que todo esta de manera local ya que si se ven la excepciones en el cliente de la misma forma que si fuera local.