

ANEXO D

CÓDIGOS FUENTE DE LA APLICACIÓN VISOR DICOM

Clase VISOR.java

```
/**
 * Esta clase es la principal y ejecutable; contiene los métodos a
 * llamar desde el exterior
 * @autora Ma José Aguilar Porro
 */

package visor;

import javax.swing.*;
import java.awt.*;
import visor.menus.MenuPpal;
import visor.paneles.PanelPpal;
import visor.nucleo.FuncionesGUI;
import visor.escritores.escriptorError;

public class VISOR extends JFrame{

    /** Cuerpo de la aplicación gráfica */
    public PanelPpal PanelPrincipal = null;
    /** menús de la aplicación gráfica */
    public MenuPpal MenuPrincipal = null;

    /**
     * MAIN PRINCIPAL.
     */

    public static void main (String args[]){
        try{
            VISOR _visor = new VISOR(args);    // creamos una instancia

            if (_visor.MenuPrincipal == null){
                // si no se lanzó interfaz de usuario
                new FuncionesGUI().informar("Saliendo...\n");
                System.exit(0);
            }
        }
        catch (Throwable e){
            new FuncionesGUI().informar("main(): " + new
                escriptorError().escribirError(e));
        }
    }

    /**
     * Constructor sin parámetros. Ajusta su referencia en FuncionesGUI
     */

    private VISOR(){
        try{
            new FuncionesGUI().fijarFrameRaiz(this);
            // así la encontraremos si lo necesitamos
        }
        catch (Throwable e){
            new FuncionesGUI().informar("VISOR(): " + new
                escriptorError().escribirError(e));
        }
    }
}
```

```
/**
 * Constructor con argumentos, lanza la interfaz de usuario
 */

public VISOR (String[] args){
    this();
    try{
        if (args != null && args.length != 0){
            System.out.println("saliendo...");
        }
        else{
            this.interfazGrafica();// lanzamos la interfaz de usuario
        }
    }
    catch (Throwable e){
        new FuncionesGUI().informar("VISOR(String[]): " + new
            escritorError().escribirError(e));
    }
}

/**
 * Este método es el que inicializa toda la interfaz gráfica
 */

private void interfazGrafica(){
    try{
        //título de la ventana
        setTitle("VISOR DICOM");
        // gestión del cierre de la ventana por la "x"
        addWindowListener(new event.WindowAdapter(){
            public void WindowClosing(event.WindowEvent evt){
                System.exit(0);}});
        // fijamos la apariencia como la del sistema donde se ejecuta
        try{
            UIManager.setLookAndFeel(UIManager.
                getSystemLookAndFeelClassName());
            SwingUtilities.updateComponentTreeUI(this);
        }
        catch (Throwable e){
            System.exit(0);
        }

        // la barra de menús
        MenuPrincipal = new MenuPpal();
        setJMenuBar(MenuPrincipal);
        // el contenido
        PanelPrincipal = new PanelPpal();
        getContentPane().add(PanelPrincipal);

        //ajustamos sitio y tamaño de los componentes
        pack();
        setResizable(false);
        //mostramos
        show();
    }
    catch (Throwable e){
        System.exit(0);
    }
}}
```

Clase paneles.PanelPpal.java

```
/**
 * Esta clase define el cuerpo de la aplicación gráfica. Además, es
 * quien va a manejar el DicomObject para extraer y mostrar los datos
 */

package visor.paneles;

/**
 * Esta clase define el cuerpo de la aplicación gráfica. Además, es
 * quien va a manejar el DicomObject para extraer y mostrar los datos
 */

import javax.swing.*;
import java.awt.*;
import java.io.*;
import com.archimed.dicom.*;
import com.archimed.dicom.image.*;
import visor.escritores.*;
import visor.nucleo.*;

public class PanelPpal extends JPanel{

    // En primer lugar, se definen las variables

    public static ImageIcon foto;
    public static JLabel marcofoto;
    public static JButton seleccionArchivo;
    public static JButton cancelar;
    public static JPanel DatosPersonales;
    public static JPanel DatosClinicos;
    public static JPanel DatosPatologias;
    public static JLabel nombrePac;
    public static JLabel nombrePacBis;
    public static JLabel domicilio;
    public static JLabel domicilioBis;
    public static JLabel localidad;
    public static JLabel localidadBis;
    public static JLabel tfno;
    public static JLabel tfnoBis;
    public static JLabel fecnac;
    public static JLabel fecnacBis;
    public static JLabel numss;
    public static JLabel numssBis;
    public static JLabel resenf;
    public static JLabel resenfBis;
    public static JLabel resexpl;
    public static JLabel resexplBis;
    public static JLabel expcomp;
    public static JLabel expcompBis;
    public static JLabel diagnost;
    public static JLabel diagnostBis;
    public static JLabel tratamto;
    public static JLabel tratamtoBis;
    public static JLabel fundclin;
    public static JLabel fundclinBis;
    public static JLabel diabet;
    public static JLabel diabetBis;
    public static JLabel hipert;
    public static JLabel hipertBis;
```

```
public static JLabel anticoag;  
public static JLabel anticoagBis;  
public static JLabel interprev;  
public static JLabel interprevBis;  
public static JLabel enfants;  
public static JLabel enfantsBis;  
public static JLabel alergMed;  
public static JLabel alergMedBis;  
public static JLabel otros;  
public static JLabel otrosBis;  
  
public static DicomImage dcm = new DicomImage();  
public String openFileName;  
public File f;  
public FileInputStream fin;  
public BufferedInputStream bis;  
public JOptionPane parent = new JOptionPane();  
OyenteEventoCancelar oyente1 = new OyenteEventoCancelar();  
OyenteEventoSeleccion oyente2 = new OyenteEventoSeleccion();  
public JPanel panelVisor;  
public JTabbedPane panelTabbed;  
  
/**  
 * Constructor de la clase  
 */  
  
public PanelPpal() {  
  
    panelVisor = new JPanel();  
    foto = new ImageIcon("");  
    marcofoto = new JLabel(foto);  
    seleccionArchivo = new JButton ("Selección de archivo");  
    // Añadimos acceso directo a los botones  
    seleccionArchivo.setMnemonic(event.KeyEvent.VK_S);  
    cancelar = new JButton ("Cancelar");  
    cancelar.setMnemonic(event.KeyEvent.VK_C);  
    cancelar.addActionListener(oyente1);  
    seleccionArchivo.addActionListener(oyente2);  
  
    panelTabbed = new JTabbedPane();  
  
    DatosPersonales = new JPanel();  
  
    nombrePac = new JLabel("Apellidos y Nombre del Paciente: ");  
    nombrePacBis = new JLabel("");  
    JPanel subp1 = new JPanel();  
    subp1.setLayout(new BoxLayout(subp1, BoxLayout.X_AXIS));  
    subp1.add(nombrePac);  
    subp1.add(nombrePacBis);  
    domicilio = new JLabel("Domicilio: ");  
    domicilioBis = new JLabel("");  
    JPanel subp2 = new JPanel();  
    subp2.add(domicilio);  
    subp2.add(domicilioBis);  
    localidad = new JLabel("Localidad: ");  
    localidadBis = new JLabel("");  
    JPanel subp3 = new JPanel();  
    subp3.add(localidad);  
    subp3.add(localidadBis);  
    tfno = new JLabel("Teléfono: ");  
    tfnoBis = new JLabel("");
```

```
JPanel subp4 = new JPanel();
subp4.add(tfno);
subp4.add(tfnoBis);
fecnac = new JLabel("Fecha de Nacimiento: ");
fecnacBis = new JLabel("");
JPanel subp5 = new JPanel();
subp5.add(fecnac);
subp5.add(fecnacBis);
numss = new JLabel("Nª de afiliación a la Seguridad Social: ");
numssBis = new JLabel("");
JPanel subp6 = new JPanel();
subp6.add(numss);
subp6.add(numssBis);

resenf = new JLabel("Resumen de la enfermedad: ");
resenfBis = new JLabel("");
JPanel subp7 = new JPanel();
subp7.add(resenf);
subp7.add(resenfBis);
resexpl = new JLabel("Resumen de la exploración clínica: ");
resexplBis = new JLabel("");
JPanel subp8 = new JPanel();
subp8.add(resexpl);
subp8.add(resexplBis);
expcomp = new JLabel("Exploraciones complementarias realizadas:");
expcompBis = new JLabel("");
JPanel subp9 = new JPanel();
subp9.add(expcomp);
subp9.add(expcompBis);
diagnost = new JLabel("Diagnóstico provisional: ");
diagnostBis = new JLabel("");
JPanel subp10 = new JPanel();
subp10.add(diagnost);
subp10.add(diagnostBis);
tratamto = new JLabel("Tratamiento anterior / actual: ");
tratamtoBis = new JLabel("");
JPanel subp11 = new JPanel();
subp11.add(tratamto);
subp11.add(tratamtoBis);
fundclin = new JLabel("Fundamento Clínico de la petición: ");
fundclinBis = new JLabel("");
JPanel subp12 = new JPanel();
subp12.add(fundclin);
subp12.add(fundclinBis);

diabet = new JLabel("¿Es diabético? ");
diabetBis = new JLabel("");
JPanel subp13 = new JPanel();
subp13.add(diabet);
subp13.add(diabetBis);
hipert = new JLabel("¿Padece hipertensión? ");
hipertBis = new JLabel("");
JPanel subp14 = new JPanel();
subp14.add(hipert);
subp14.add(hipertBis);
anticoag = new JLabel("¿Sigue algún tratamiento con  
anticoagulantes? ");
anticoagBis = new JLabel("");
JPanel subp15 = new JPanel();
subp15.add(anticoag);
subp15.add(anticoagBis);
```

```
enfants = new JLabel("Enfermedades anteriores: ");
enfantsBis = new JLabel("");
JPanel subp16 = new JPanel();
subp16.add(enfants);
subp16.add(enfantsBis);
interprev = new JLabel("Intervenciones previas: ");
interprevBis = new JLabel("");
JPanel subp17 = new JPanel();
subp17.add(interprev);
subp17.add(interprevBis);
alergMed = new JLabel("Alergias a medicamentos: ");
alergMedBis = new JLabel("");
JPanel subp18 = new JPanel();
subp18.add(alergMed);
subp18.add(alergMedBis);
otros = new JLabel("Otros comentarios: ");
otrosBis = new JLabel("");
JPanel subp19 = new JPanel();
subp19.add(otros);
subp19.add(otrosBis);

// Añadimos los componentes a los paneles

DatosPersonales.setLayout(new BoxLayout
    (DatosPersonales,BoxLayout.Y_AXIS));
DatosPersonales.add(subp1);
DatosPersonales.add(subp2);
DatosPersonales.add(subp3);
DatosPersonales.add(subp4);
DatosPersonales.add(subp5);
DatosPersonales.add(subp6);

DatosClinicos = new JPanel();
DatosClinicos.setLayout(new BoxLayout
    (DatosClinicos,BoxLayout.Y_AXIS));
DatosClinicos.add(subp7);
DatosClinicos.add(subp8);
DatosClinicos.add(subp9);
DatosClinicos.add(subp10);
DatosClinicos.add(subp11);
DatosClinicos.add(subp12);

DatosPatologias = new JPanel();
DatosPatologias.setLayout(new BoxLayout
    (DatosPatologias,BoxLayout.Y_AXIS));
DatosPatologias.add(subp13);
DatosPatologias.add(subp14);
DatosPatologias.add(subp15);
DatosPatologias.add(subp16);
DatosPatologias.add(subp17);
DatosPatologias.add(subp18);
DatosPatologias.add(subp19);

panelTabbed.addTab("Datos Personales",DatosPersonales);
panelTabbed.addTab("Datos clínicos",DatosClinicos);
panelTabbed.addTab("Patologías intercurrentes",DatosPatologias);
panelTabbed.setEnabledAt(0,true);
panelTabbed.setEnabledAt(1,true);
panelTabbed.setEnabledAt(2,true);
panelTabbed.setSelectedIndex(0);
```

```
setLayout(new BorderLayout(this, BorderLayout.Y_AXIS));
panelVisor.setLayout(new BorderLayout(panelVisor, BorderLayout.X_AXIS));

JPanel subpanel_1 = new JPanel();
subpanel_1.add(marcofoto);
subpanel_1.setPreferredSize(new Dimension(500, 300));
panelVisor.add(subpanel_1);

JPanel subpanel_2 = new JPanel();
subpanel_2.add(seleccionArchivo);
subpanel_2.add(cancelar);
panelVisor.add(subpanel_2);

add(panelVisor);

add(panelTabbed);

}

// Oyentes para eventos

class OyenteEventoCancelar implements event.ActionListener{
    public void actionPerformed(event.ActionEvent e) {
        System.exit(0);
    }
}

class OyenteEventoSeleccion implements event.ActionListener{
    public void actionPerformed(event.ActionEvent e) {
        JFileChooser chooser = new JFileChooser();
        int returnVal = chooser.showOpenDialog(parent);
        if(returnVal == JFileChooser.CANCEL_OPTION)
            parent.getParent();
        if(returnVal == JFileChooser.APPROVE_OPTION)
        {
            try{
                openFileName = chooser.getSelectedFile().getAbsolutePath();
                // Seleccionamos el archivo que queremos abrir
                f = new File(openFileName);
                fin = new FileInputStream(f);
                bis = new BufferedInputStream(fin);
                // Lo metemos en un objeto DICOM
                dcm.read(bis, true);

                /**
                 * Comenzamos extrayendo los datos de texto que contiene el
                 * diccionario de datos DICOM
                 */
                String paciente = dcm.get(DDict.dPatientName).toString();
                nombrePacBis.setText(paciente);
                String domicil = dcm.get(DDict.dPatientAddress).toString();
                domicilioBis.setText(domicil);
                String localid = dcm.get(DDict.
                    dRegionOfResidence).toString();
                localidadBis.setText(localid);
                String telef = dcm.get(DDict.
                    dPatientTelephoneNumber).toString();
                tfnoBis.setText(telef);
                String fec = dcm.get(DDict.dPatientBirthDate).toString();
                fecnacBis.setText(fec);
```

```
/**
 * Continuamos extrayendo los datos de texto que no están en
 * el estándar
 */

String resEnf = dcm.getS_ge(0010,4001);
resenfBis.setText(resEnf);
String resExpl = dcm.getS_ge(0010,4002);
resexplBis.setText(resExpl);
String expComp = dcm.getS_ge(0010,4003);
expcompBis.setText(expComp);
String diagProv = dcm.getS_ge(0010,4004);
diagnostBis.setText(diagProv);
String trata = dcm.getS_ge(0010,4005);
tratamtoBis.setText(trata);
String fundClin = dcm.getS_ge(0010,4006);
fundclinBis.setText(fundClin);
String nums = dcm.getS_ge(0010,4010);
numssBis.setText(nums);
String diaBet = dcm.getS_ge(0010,4011);
diabetBis.setText(diaBet);
String hiPert = dcm.getS_ge(0010,4012);
hipertBis.setText(hiPert);
String antiCoag = dcm.getS_ge(0010,4013);
anticoagBis.setText(antiCoag);
String interPrev = dcm.getS_ge(0010,4014);
interprevBis.setText(interPrev);
String enfAnts = dcm.getS_ge(0010,4015);
enfantsBis.setText(enfAnts);
String alergias = dcm.getS_ge(0010,4016);
alergMedBis.setText(alergias);
String otroComent = dcm.getS_ge(0010,4017);
otrosBis.setText(otroComent);

/**
 * Y para terminar, extraemos y mostramos la imagen
 */

com.archimed.dicom.image.ImageIO imagenio = new
    com.archimed.dicom.image.ImageIO(dcm);
java.awt.image.ImageProducer imagep =
    imagenio.getImageProducer();
foto = new ImageIcon(devImagen(imagep));
marcofoto.setIcon(foto);
}
catch (Throwable t){
    new FuncionesGUI().informar(new
        escritorError().escribirError(t));
}
}
}

/**
 * El siguiente método devuelve una instancia de la clase Image
 * a partir del ImageProducer que se le pasa como parámetro.
 */

public java.awt.Image devImagen(java.awt.image.ImageProducer imp){
    java.awt.Image im = createImage(imp);
    return im;}}
```

El resto de clases están contenidas en los paquetes *visor.nucleo*, *visor.menus* y *visor.escritores*; como son muy parecidas a las mismas clases en las otras dos aplicaciones, no incluiremos el código aquí, aunque sí que está en el CD correspondiente.