```
import modules.email.action.*;
import modules.user.register.*;
import javax.swing.*;

import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;

import java.io.InputStreamReader;import javax.swing.event.ListSelectionListener;
import javax.swing.event.ListSelectionEvent;

import javax.swing.border.EtchedBorder;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.awt.Toolkit;
import java.awt.Color;


/**
 * email Window
 * This file will generate a window for the ACL2 email system. This is a client based window and
 * will call the shell scripts we generated for the ACL2 program
 *
 * mainly, this is a fancy way of entering data instead of using the console and navigating folders.
 *
 * The Icons in this applications are open source obtained from http://openiconlibrary.sourceforge.net/
 * The license stipulates that the icons are free to use, modify and redistribute.
 *
 * March 13, 2013
 * @version 1.0
 * @author Wesley R. Howell
 *
 */

public class emailWindow extends JFrame {

        private static final long serialVersionUID = 1L;
        private  File [] listofFiles;
        private String [] names;
        private String currentUser     = "matthew.crist";
        private String currentDomain   = "localhost";
        private String currentPassword = "simulation";

        /**
         * Run The Email Window object. Most of the code will be handled here.
         */
        public emailWindow() {
                getContentPane().setBackground(new Color(192, 192, 192));
                setIconImage(Toolkit.getDefaultToolkit().getImage("lib/icon.gif"));
                /**
                 * Close the application when the window closes.
                 */
                setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

                /**
                 * Setup Code
                 */
                setTitle("ACL2 Email System\n");
                getContentPane().setLayout(null);
                final JList list = new JList();
                list.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
                list.setBorder(new EtchedBorder(EtchedBorder.RAISED, null, null));
                final JLabel lblBlah = new JLabel();
                lblBlah.setBackground(Color.WHITE);
                JButton btnGetEmail = new JButton("Get Email");
                btnGetEmail.setIcon(new ImageIcon("lib/email_open.png"));


                /**
                 * Get Email Button Action
                 */
                btnGetEmail.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent arg0) {
                                String s;
//
                                //GetEmail.getEmail("matthew.crist", "localhost", "simulation");
                                GetEmail.getEmail(currentUser, currentDomain, currentPassword);
                                try{
                                Thread.sleep(2000);
                                } catch (InterruptedException e){
                                    e.printStackTrace();
                                }
                                //try {
//                                        Process p = Runtime.getRuntime().exec(
//                                                        "sh /Users/w_howell/code/spring-2013-se2-dijkstra↙
    /client/modules/email/email-action/java-email.sh");
//                                        BufferedReader err = new BufferedReader (
//                                                        new InputStreamReader (p.getErrorStream()));
//                                        while ((s = err.readLine()) != null) {
//                                                System.out.println(s);
//                                        }
//                                  }
//                                  catch (Exception e) {
//                                        e.printStackTrace();
//                                  }

                                //Update Jlist
                                File folder = new File("store/email/inbox");
                                folder.mkdirs();
                        File [] rawContents = folder.listFiles();
                        listofFiles = new File[rawContents.length];
                        names = new String[rawContents.length];
                       // Map<String, File> listforgui = new HashMap<String,File>();
                        for (int i = 0; i < rawContents.length; i++){
                                if(rawContents[i].isFile() && !rawContents[i].isHidden()){
                                        listofFiles[i] = rawContents[i];
                                        names[i] = getEmailHeader(listofFiles[i]);
                                }
                        }
                        list.setListData(names);
                        }
                });
                btnGetEmail.setBounds(16, 6, 117, 29);
                getContentPane().add(btnGetEmail);




                /**
                 * Load the Files in the inbox and add to the list of input
                 */
                File folder = new File("store/email/inbox");
        File [] rawContents = folder.listFiles();
        listofFiles = new File[rawContents.length];
        names = new String[rawContents.length];
       // Map<String, File> listforgui = new HashMap<String,File>();
        for (int i = 0; i < rawContents.length; i++){
                if(rawContents[i].isFile() && !rawContents[i].isHidden()){
                        listofFiles[i] = rawContents[i];
                        names[i] = getEmailHeader(listofFiles[i]);
                }
        }
```

```
            /**
             * Read in the email message when it is clicked
             */
            list.addListSelectionListener(new ListSelectionListener() {
                    public void valueChanged(ListSelectionEvent arg0) {
                            //Open file and get text!

                            BufferedReader reader;
                            try {
                                    StringBuilder  stringBuilder = new StringBuilder();
                                    if(list.getSelectedIndex() != -1){
                                    reader = new BufferedReader( new FileReader (listofFiles[list.     ↙
getSelectedIndex()]));
                                     String         line = null;

                                        String         ls = System.getProperty("line.separator");

                                        while( ( line = reader.readLine() ) != null ) {
                                            stringBuilder.append( line );
                                            stringBuilder.append( ls );
                                        }
                                    }

                                        lblBlah.setText(stringBuilder.toString());
                            } catch (FileNotFoundException e) {
                                    // TODO Auto-generated catch block
                                    e.printStackTrace();
                            } catch (IOException e) {
                                    // TODO Auto-generated catch block
                                    e.printStackTrace();
                            }

                    }
            });
            list.setBounds(16, 44, 315, 612);
            list.setListData(names);
            getContentPane().add(list);


            /**
             * Extra "eye candy" in the window
             */
            JPanel panel = new JPanel();
            panel.setBackground(Color.WHITE);
            panel.setBorder(new EtchedBorder(EtchedBorder.RAISED, null, null));
            panel.setBounds(336, 44, 682, 681);
            getContentPane().add(panel);
            panel.setLayout(null);


            /**
             * Setup for Email Viewer
             */
            lblBlah.setVerticalAlignment(SwingConstants.TOP);
            lblBlah.setBounds(6, 6, 670, 669);
            panel.add(lblBlah);

            JLabel lblNewLabel = new JLabel("");
            lblNewLabel.addMouseListener(new MouseAdapter() {
                    @Override
                    public void mouseClicked(MouseEvent arg0) {
                            try {
                                    String url = "http://en.wikipedia.org/wiki/ACL2";
                             java.awt.Desktop.getDesktop().browse(java.net.URI.create(url));
                            } catch (IOException e) {
                                    // TODO Auto-generated catch block
                                    e.printStackTrace();
                            }
                    }
            });
            lblNewLabel.setIcon(new ImageIcon("lib/logo.gif"));
            lblNewLabel.setBounds(16, 688, 83, 74);
            getContentPane().add(lblNewLabel);
            JLabel lblAcl = new JLabel("<html><h2>ACL2 Email System</h2><p>2013 Team Dijkstra \n        ↙
University of Oklahoma</p></html>\n");
            lblAcl.setVerticalAlignment(SwingConstants.TOP);
            lblAcl.setBounds(121, 688, 183, 74);
            getContentPane().add(lblAcl);

            /**
             * New Email Message Button
             */
            JButton btnNewMessage = new JButton("New Message");
            btnNewMessage.setIcon(new ImageIcon("lib/email_edit.png"));
            btnNewMessage.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                            newMessage msg = new newMessage();
                            msg.setSize(602,487);
                            msg.setVisible(true);
                    }
            });
            btnNewMessage.setBounds(142, 6, 132, 29);
            getContentPane().add(btnNewMessage);

            JLabel lblcodegooglecompspringsedijkstra = new JLabel("<html><font size=4><a href=https:// ↙
code.google.com/p/spring-2013-se2-dijkstra//p/>code.google.com/p/spring-2013-se2-dijkstra</a>\n    </  ↙
font></html>");
            lblcodegooglecompspringsedijkstra.addMouseListener(new MouseAdapter() {
                    @Override
                    public void mouseClicked(MouseEvent arg0) {


                            // open the default web browser for the HTML page
                            try {
                                    String url = "https://code.google.com/p/spring-2013-se2-dijkstra/";
                             java.awt.Desktop.getDesktop().browse(java.net.URI.create(url));
                            } catch (IOException e) {
                                    // TODO Auto-generated catch block
                                    e.printStackTrace();
                            }
                    }
            });

            /**
             * Link to Google Page
             */
            lblcodegooglecompspringsedijkstra.setVerticalAlignment(SwingConstants.TOP);
            lblcodegooglecompspringsedijkstra.setBounds(688, 737, 330, 25);
            getContentPane().add(lblcodegooglecompspringsedijkstra);


            /**
             * Register Email button
             */
            JButton btnRegister = new JButton("Register");
            btnRegister.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    currentUser = JOptionPane.showInputDialog("Input your name");
                    currentDomain  = JOptionPane.showInputDialog("input your domain");
                    currentPassword  = JOptionPane.showInputDialog("Input your Password");            ↙

                    SendRequest.sendRequest(currentUser, currentDomain, currentPassword);
                }
            });
            btnRegister.setIcon(new ImageIcon("lib/user_add.png"));
            btnRegister.setBounds(901, 6, 117, 29);
            getContentPane().add(btnRegister);

    JButton btnNewButton = new JButton("Delete Message");
    btnNewButton.setIcon(new ImageIcon("lib/email_delete.png"));
            btnNewButton.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
            int currentlySelected = list.getSelectedIndex();
            listofFiles[currentlySelected].delete();


            //Update Jlist
                        File folder = new File("store/email/inbox");
            folder.mkdirs();
            File [] rawContents = folder.listFiles();
            listofFiles = new File[rawContents.length];
            names = new String[rawContents.length];

            for (int i = 0; i < rawContents.length; i++){
                            if(rawContents[i].isFile() && !rawContents[i].isHidden()){
                                    listofFiles[i] = rawContents[i];
                                    names[i] = getEmailHeader(listofFiles[i]);
                            }
            }
            list.setListData(names);
                }
            });
            btnNewButton.setBounds(16, 657, 315, 29);
            getContentPane().add(btnNewButton);


    }

    /**
     * Gets the email header for listing in the JFrame list
     * @param file
     * @return
     */
    @SuppressWarnings("resource")
    private String getEmailHeader(File file){
            String rv = "";
            BufferedReader reader;
            try {
                    reader = new BufferedReader( new FileReader (file));
                            String         line = null;
                            String subject = "";
                        String name ="";

                        int number = 0;
                        while( ( line = reader.readLine() ) != null ) {
                            if(number == 1){
                                    subject = line.substring(4, line.indexOf("</h1>"));

                            }
                            if(number == 3){
                                    name = line.substring(11, line.indexOf("</h2>"));
                            }
                            number++;
                        }
                        rv = name + "  -  " + subject +"    -  " + file.getName().substring(4);
            } catch (FileNotFoundException e) {
                    // TODO Auto-generated catch block
                    e.printStackTrace();
            } catch (IOException e) {
                    // TODO Auto-generated catch block
                e.printStackTrace();
            }

            return rv;
        }
    }
```