

Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ciencias y Sistemas

Programación de Computadoras 2 Sección: R

Ing. HERMAN IGOR VELIZ LINARES

Aux. José Eduardo Morales García

Manual de usuario

Pedro Pablo Raúl López Vargas

202004730

Contenido

Introducción	3
Objetivos	4
Objetivo General:	4
Objetivos específicos.....	4
Diccionario de clases	5

Introducción

Se necesita un programa en lenguaje Java que sea de mostrar todas las opciones requeridas para la empresa USAC delivery, mostrando y registrando valores y la entrega de paquetes de la empresa sea registrada de mejor manera y funcionando de manera optima utilizando el programa desarrollado.

Objetivos

Objetivo General:

Aplicar los conocimientos adquiridos en el curso sobre el lenguaje java

Objetivos específicos

Utilizar método y funciones

Manejo de clases

Manejo de listas dinámicas

Diccionario de clases

En el programa se ha utilizado diferentes clases y métodos para llevar a cabo el funcionamiento del programa:

1. Clase proyecto

Esta clase contiene el main que abre la venta principal para mostrar todos los paneles, además de almacenar los arraylist creados.

```
    */
    public class Proyecto1 {

        /**
         * @param args the command line arguments
         */
        static ArrayList <registrocod> listadatos = new ArrayList <registrocod>();
        static ArrayList <regionescod> listaregiones = new ArrayList <regionescod>();
        static ArrayList <departamentoscod> listadeptos = new ArrayList <departamentoscod>();
        static ArrayList <Municipiociod> listamunicipios = new ArrayList <Municipiociod>();
        static ArrayList <kioskocod> listakioskos = new ArrayList <kioskocod>();
        static ArrayList <tarjetacod> listatarjetas = new ArrayList <tarjetacod>();
        static ArrayList <facturacioncod> listafacturacion = new ArrayList <facturacioncod>();
        static ArrayList <cotizacioncod> listacotizacion = new ArrayList <cotizacioncod>();
        static ArrayList <compracod> listacompras = new ArrayList <compracod>();

        public static void main(String[] args) {
            ventana v = new ventana();
            v.setVisible(true);
        }
    }
}
```

2. importes de paneles

En la ventana se importaron todas las clases y las librerías a usar

```

import java.awt.BorderLayout;
import proyectol.elementos.compracod;
import proyectol.elementos.cotizacioncod;
import proyectol.elementos.regionescod;
//import java.util.ArrayList;
import proyectol.elementos.registrocod;
import proyectol.ventanas.paneles.Datosfactupanel;
import proyectol.ventanas.paneles.Kioskopanel;
import proyectol.ventanas.paneles.Municipiopanel;
import proyectol.ventanas.paneles.Regionestab;
import proyectol.ventanas.paneles.adminop;
import proyectol.ventanas.paneles.comprapanel;
import proyectol.ventanas.paneles.cuentapa;
import proyectol.ventanas.paneles.departpanel;
import proyectol.ventanas.paneles.enviossolittab;
import proyectol.ventanas.paneles.login;
import proyectol.ventanas.paneles.menu;
import proyectol.ventanas.paneles.multiplepanel;
import proyectol.ventanas.paneles.registro;
import proyectol.ventanas.paneles.repor2pan;
import proyectol.ventanas.paneles.reporte3pan;
import proyectol.ventanas.paneles.tablaregiones;
import proyectol.ventanas.paneles.tarjetastab;

```

```

/**
 *

```

setRegiones()

Este método sirve para que en las ventas que sea necesaria usar las regiones declaradas pueda ser utilizadas en otras clases

```

public ventana() {
    initComponents();
    cambiarPanel(log);
    Reporteb.setVisible(false);
    regkiosco.setVisible(false);
    regpre.setVisible(false);
    depbut.setVisible(false);
    munibut.setVisible(false);
    mostrarregact.setVisible(false);
    salirbut.setVisible(false);
    cotpaqbut.setVisible(false);
    descargabut.setVisible(false);
    regdatbut.setVisible(false);
    regtarbut.setVisible(false);
    comprabut.setVisible(false);
    verenviosbut.setVisible(false);
    jPanel3.setVisible(false);
    regtab.setRegiones(regionMetropolitana, regionNorte, regionNororient, regionSuroccidente, regionNoroccidente);
    deptotab.setRegiones(regionMetropolitana, regionNorte, regionNororient, regionSuroccidente, regionNoroccidente);
    multipan.setRegiones2(regionMetropolitana, regionNorte, regionNororient, regionSuroccidente, regionNoroccidente);
}

```

Cambiar panel

Este método permite que al ser llamado cualquier panel el panel sea cambiado y se muestre en la ventana principal

```

private void cambiarPanel(javax.swing.JPanel miPanel){
    miPanel.setSize(716, 477);
    miPanel.setLocation(0, 0);
    jPanel1.removeAll();
    jPanel1.add(miPanel, BorderLayout.CENTER);
    jPanel1.revalidate();
    jPanel1.repaint();
}

```

Login

El botón login verificar que exista el dato de correo y contraseña en el arraylist de registro, al mismo tiempo detecta si los valores correctos.

Tiene como funcionan también en verificar si se accede como administrador o como usuario cualquiera, por lo que dependiendo de cual se mostraran las siguientes paneles y las otras estarán ocultas

```

private void LoginActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    for(int i =0;i<reg.getregistrocod().size();i++){
        registrocod temporal = new registrocod();
        temporal=reg.getregistrocod().get(i);
        if(log.getUsuario().equals(temporal.getCorreo()) && log.getClave().equals(temporal.getcontra())){
            if (log.getUsuario().equals("ipcl_202004730@ipcldelivery.com") && log.getClave().equals("202004730")) {
                cambiarPanel(admintab);
                jPanel2.setVisible(true);
                admintab.setVisible(true);
                Reporteb.setVisible(true);
                regkiosco.setVisible(true);
                regpre.setVisible(true);
                depbut.setVisible(true);
                munibut.setVisible(true);
                salirbut.setVisible(true);
                Login.setVisible(false);
                Registrar.setVisible(false);
                regresar.setVisible(false);
                cotpaqbut.setVisible(false);
                descargabut.setVisible(false);
                regdatbut.setVisible(false);
                regtarbut.setVisible(false);
                comprabut.setVisible(false);
                verenviosbut.setVisible(false);
                jPanel3.setVisible(false);
                break;
            }else{

```

Acciones botones

Cada botón al ser presionado mostrara el panel requerido, además que si es necesario usar algún valor de una arraylist de otra clase esta será llamado por el método .set();

```
private void RegistrarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    reg.setListaKioskos(kiostab.getkioskocod());
    reg.crearListaDesplegable4();
    cambiarPanel(reg);
    jPanel2.setVisible(true);
    reg.setVisible(true);
}
}
```

```

    usuarioactivo=log.getUsuario();
    tarjtab.setDato(log.getUsuario());
    factupanel.setDato2(log.getUsuario());
    comprapanell.setDatos0(log.getUsuario());
    enviospan.setDatosf(log.getUsuario());

    cambiarPanel(men);
    Login.setVisible(false);
    Registrar.setVisible(false);
    regresar.setVisible(false);
    jPanel2.setVisible(true);
    men.setVisible(true);
    cotpaqbut.setVisible(true);
    descargabut.setVisible(true);
    regdatbut.setVisible(true);
    regtarbut.setVisible(true);
    comprabut.setVisible(true);
    verenviosbut.setVisible(true);
    salirbut.setVisible(true);
    jPanel3.setVisible(true);
    multipan.setDato1(temporal.getynkiosko());
    multipan.setDato2(log.getUsuario());
//    System.out.println(usuarioactivo);
    break;}
}else{
    etiquetaError.setText("Usuario o Clave incorrectas, intente nuevamente");
}
}
}
```

SetLista y crearlista

Estos métodos funcionan para conseguir los datos de un array list y para que en un deseado combobox se muestren ciertos datos

```
private void comprabutActionPerformed(java.awt.event.ActionEvent evt) {
    comprapanell.setListacoti(multipan.getcotizacioncod());
    comprapanell.crearListaDesplegablec1();
    comprapanell.setListatarj(tarjtab.gettarjetacod());
    comprapanell.crearListaDesplegablec2();
    comprapanell.setListafactu(factupanel.getfacturacioncod());
    comprapanell.crearListaDesplegablec3();
    cambiarPanel(comprapanell);
    jPanel3.setVisible(true);
    jPanel2.setVisible(true);
}
}
```

Tabla HTML

Se estableció una tabla en formato HTML utilizando valores de una arraylist y mostrarlos en un panel

```
private void verenviosbutActionPerformed(java.awt.event.ActionEvent evt) {  
    String textoenvios="<html><head></head><body><table><tr><th>Codigo</th><th>Tipo de servicio</th><th>Destinatario</th><th>Total de envio</th><th>Tipo de pag<  
    for(int rt=0; rt<comprapanel1.getcompracod().size();rt++){  
        compracod temporalr = new compracod();  
        temporalr = comprapanel1.getcompracod().get(rt);  
        for (int yu = 0; yu <multipan.getcotizacioncod().size() ; yu++) {  
            cotizacioncod temporayu = new cotizacioncod();  
            temporayu = multipan.getcotizacioncod().get(yu);  
            if (usuarioactivo.equals(temporayu.getNombreakusuario())) {  
  
                if (temporayu.getCodigo()==temporalr.getPaqcod()) {  
                    textoenvios = textoenvios+"<tr><td>" +temporalr.getPaqcod()+"</td><td>" +temporayu.getTipbreakservicio()+"</td><td>" +temporayu.getDepbreakdestino()+"tempora  
                }  
            }  
        }  
    }  
  
    textoenvios=textoenvios+"</table></body></html>";  
    enviospan.setTextArea(textoenvios);  
    cambiarPanel(enviospan);  
    jPanel2.setVisible(true);  
}
```

Variables en clases

En cada clase se han declarado las variables a utilizar

```
public class departamentoscod {  
  
    private String codreg;  
    private String nregion;  
    private String ndepto;  
    private String coddept;
```

Constructores

Se declaran los constructores vacíos y con valores

```
public departamentoscod() {  
    codreg="";  
    nregion="";  
    ndepto="";  
    coddept="";  
}  
  
public departamentoscod(String codreg, String nregion, String ndepto, String coddept) {  
    this.codreg = codreg;  
    this.nregion = nregion;  
    this.ndepto = ndepto;  
    this.coddept = coddept;  
}
```

Get y set

Se crearon los métodos para obtener los valores del método principal o establecer un valor en ello

```

]      public String getCoddept() {
-      return coddept;
-      }

]      public void setCodreg(String codreg) {
-      this.codreg = codreg;
-      }

```

SetLista

Este método se utiliza para obtener los valores de una arraylist y su metodo respectivo para utilizarlos en diferentes paneles

```

public void setListadat(ArrayList <registrocod> listadatos){
    this.listadatos=listadatos;
}

```

Crea lista desplegable

Este metodo se utiliza para que en determina combobox se muestren valores ya almacenados de un arraylist

```

public void crearListaDesplegable4() {
    jComboBox1.removeAllItems();
    for(int a=0; a<listakioskos.size();a++){
        kioskocod temporal4 = new kioskocod();
        temporal4 = listakioskos.get(a);
        jComboBox1.addItem(temporal4.getNkiosko());
    }
}

```

verificación

Para verificar que algunos datos no estén repetidos se utilizado un metodo que retorna si es verdadero o falso recorriendo la lista

```

public boolean vericorreos() {
    for(int io =0;io<listadatos.size();io++){
        registrocod tempocor = new registrocod();
        tempocor=listadatos.get(io);
        if (tempocor.getCorreo().equals(correos.getText())) {
            JOptionPane.showMessageDialog(null, "El correo ya ha sido ingresado anteriormente","Error", JOptionPane.WARNING_MESSAGE)
            return true;
        }
    }
    return false;
}

```

Determinar rol

Se utilizo esta condición para determinar si el rol de un usuario es usuario o kiosko

```
//  
if (Rol.getSelectedItem().equals("Usuario")) { //asignar valor de kiosko  
    combovalor="NA";  
} else if (Rol.getSelectedItem().equals("Kiosko")) {  
    combovalor=(String) jComboBox1.getSelectedItem();  
}
```