



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Carrera: Ingeniería en Ciencias y Sistemas

MANUAL
“Tecnico”

Nombre: Gerardo Leonel Ortiz Tobar
Carné: 202200196
Fecha de presentación: 15-03-23

Paquetería Usac 2023

Paneles de manejo

1. Método de agregar

```
private void agregarActionPerformed(java.awt.event.ActionEvent evt) {  
    try{  
        String cod = correo.getText();  
        String nom= nombreTarjeta.getText();  
        String num= numeroTarjeta.getText();  
        String fecha = fechaExp.getText();  
  
        if(!(cod.equals( anObject: "" ) || nom.equals( anObject: "" ) || num.equals( anObject: "" ) || fecha.equals( anObject: "" ) || num.length() < 4)){  
  
            listaTarjetas.add(new Tarjetas( correo: cod, nombreTarjeta: nom, numeroTarjeta: num, fechaVencimiento: fecha ));  
            System.out.println("Cod: "+cod +" nombre Tarjeta: " + nom+" numero Tarjeta: " + num +" Fecha Exp: " + fecha);  
            JOptionPane.showConfirmDialog( parentComponent: this, "Se ingreso la tarjeta: " + num, title: "Confirmar acción", optionType: JOptionPane.DEFAULT_OPTION);  
  
            reseteoCampos();  
  
        }else{  
            JOptionPane.showConfirmDialog( parentComponent: this, message: "Ingrese todos los datos", title: "Confirmar acción", optionType: JOptionPane.DEFAULT_OPTION);  
        }  
    }catch (Exception e ){  
        numerolabel.setForeground( fg: Color.red);  
    }  
}
```

En esta parte se está tomando los datos que ingresamos en la parte de las tarjetas.

Se va a la validación si no se han mandado vacíos lo campos y si tira un mensaje que debe de ingresar los datos correspondientes, en dado caso que no hay errores lo va a crear.

En dado caso si hay un error en el tipo dato, mandara una alerta y señalara el error del campo.

2. Método de editar

```
private void editarActionPerformed(java.awt.event.ActionEvent evt) {  
    try{  
        String cod = correo.getText();  
        String nom= nombreTarjeta.getText();  
        String num= numeroTarjeta.getText();  
        String fecha = fechaExp.getText();  
  
        if(!(cod.equals("") || nom.equals("") || num.equals("") || fecha.equals("") || num.length()<4)){  
  
            int i = obtenerPosicion(num);  
            listaTarjetas.set(index:i,new Tarjetas( correo:cod, nombreTarjeta:nom, numeroTarjeta:num, fechaVencimiento: fecha ));  
            System.out.println("Cod: "+cod + " nombre Tarjeta: " + nom+ " numero Tarjeta: " + num + " Fecha Exp: " + fecha);  
            JOptionPane.showConfirmDialog(parentComponent: this, "Se modifiko la tarjeta: " + num, title:"Confirmar acción", optionType:JOptionPane.DEFAULT_OPTION);  
  
            reseteoCampos();  
  
        }else{  
            System.out.println(x:"Ingrese todos los datos");  
            JOptionPane.showConfirmDialog(parentComponent: this, message:"Ingrese todos los datos", title:"Confirmar acción", optionType:JOptionPane.DEFAULT_OPTION);  
  
        }  
    }catch(Exception e ){  
        numeroLabel.setForeground( f:Color.red);  
    }  
}
```

En esta parte se está tomando los datos que ingresamos en la parte de las tarjetas parecido al agregar.

```
public int obtenerPosicion(String num){  
    for(int i = 0; i < listaTarjetas.size(); i++){  
        Tarjetas t = listaTarjetas.get( index:i);  
        if(num.equalsIgnoreCase( anotherString:t.getNumeroTarjeta())){  
            return i;  
        }  
    }  
    return -1;  
}
```

Con una función de obtenerPosicion, se le manda el número de tarjeta y lo buscara en la ArrayList, y si lo encuentra mandara la posición con la que se ubica, para su edición.

Se va a la validación si no se han mandado vacíos los campos y si tira un mensaje que debe de ingresar todos los datos correspondientes, en dado caso que no hay errores lo va a editar

En dado caso si hay un error en el tipo de dato, mandara una alerta y señalara el error del campo.

3. Método de eliminar

```
private void eliminarActionPerformed(java.awt.event.ActionEvent evt) {  
    try{  
        String num= numeroTarjeta.getText();  
  
        listaTarjetas.remove(index:obtenerPosicion(num));  
        JOptionPane.showConfirmDialog(parentComponent:this, message:"Se elimino la tarjeta", title:"Confirmar acción", optionType:JOptionPane.DEFAULT_OPTION);  
    }catch(Exception e){  
        numerolabel.setForeground( fg:Color.red);  
        JOptionPane.showConfirmDialog(parentComponent: this, message:"Se necesita numero de tarjeta para eliminar", title:"Confirmar acción", optionType:JOptionPane.DEFAULT_OPTION);  
    }  
}
```

Para eliminar, se va a requerir los datos del numero de tarjeta, con base a ese numero se va a buscar con la función de obtenerPosicion.

Con el método de remove, se utiliza la posición que encontramos y va hacia el arraylist y eliminar la tarjeta seleccionada.

4. Método de Mostrar

```
private void mostrarActionPerformed(java.awt.event.ActionEvent evt) {  
    try{  
        String cadenaTarjeta="<html><head></head><body><table><tr><th>Nombre Tarjeta</th><th>Numero Tarjeta</th><th>Fecha Exp</th><th>Correo</th></tr>";  
        for(int i=0; i<listaTarjetas.size();i++){  
  
            tar = listaTarjetas.get(i);  
            if(correo.getText().equals(anObject: "")){  
                JOptionPane.showMessageDialog(parentComponent: this, message: "Falta el correo", title: "Confirmar acción", optionType: JOptionPane.DEFAULT_OPTION);  
                correoLabel.setForeground(Color.red);  
  
            }else{  
                if(tar.getCorreo().equals(anObject: correo.getText())){  
                    cadenaTarjeta = cadenaTarjeta+"<tr><td>" + tar.getNombreTarjeta()+"</td><td>" + ocultarNumeros(num: tar.getNumeroTarjeta())+"</td>" + "<td>" + tar.getFechaVencimie  
                }else{  
                    System.out.println("No es del usuario");  
                }  
            }  
        }  
    }  
}
```

En este método, declaramos una variable de cadena de Tarjetas y le ingresamos una media estructura de HTML para la vista del mostrar, después hacemos un recorrido por toda la ArrayList de tarjetas y lo ira almacenando en la variable tar, y se compara el correo de la persona almacenada y la compara con la de la tarjeta y mostrara, en dado caso no lo sea, no se mostrara esa tarjeta, mandara un mensaje en consola.

```
public String ocultarNumeros(String num){  
    String ocultado = "XXXX-XXXX-XXXX-";  
  
    String ultimosCuatro = num.substring(num.length() - 4);  
    String tarjetaOcultada = ocultado + ultimosCuatro;  
    return tarjetaOcultada;  
}
```

se crea otra variable con un texto para ocultar las demás características de la tarjeta.

Con la función de ocultaNumeros, le mandamos el numero de tarjeta y con base al método de substring que corte hasta los últimos 4 dígitos y se guarda en esa variable

Después se concatena con la variable de los cuatro dígitos y se manda por medio de un return.

Después que se hace todo eso ya solo falta terminar la estructura del HTML.

Si en dado caso no se le envía el numero de tarjeta, le tira una señal donde deberá de ingresar ese numero.