**APLICACION DEL METODO DE COBERTURA DE DECISIONES AL PROGRAMA DE PACIENTES**

Para cubrir las decisiones del programa, debemos validar los siguientes elementos:

***El While que realiza el login.***

**while** ((linea = br.readLine()) != **null**) {

datos = linea.split(" ");

***Caso 1:*** Probamos la decisión por falso del bucle leyendo una cadena vacía: {línea = null}.

***Caso 2:*** Probamos la decisión por verdadero del bucle leyendo una cadena con: {línea = Lucas 123}.

***El if que comprueba los datos del login***

**if**(datos[0].compareTo(usr) == 0 && datos[1].compareTo(pass) == 0) {

logueo = **true**;

**break**;

}

}

***Caso 3:*** Probamos la decisión por falso del condional ingresando un usuario inexistente: {usr = “Wanchope”, pass = “”}

***Caso 4:*** Probamos la decisión por verdadero del ciclo ingresando un nombre de usuario y una contraseña existentes: {usr = “Lucas”, pass = “123”}

***La lógica que verifica que se ingresó un usuario y una contraseña correcta.***

**if** (logueo) {

Inicio ingredat = **new** Inicio(usr);

ingredat.setVisible(**true**);

**this**.dispose();

} **else** {

JOptionPane.showMessageDialog(**null**, "Usuario y/o contraseña inválidos");

usuario.setText("");

password.setText("");

}

***Caso 5:*** Probamos la decisión por verdadero del condicional: {logueo = true}.

***Caso 6:*** Probamos la decisión por falso del condicional: {logueo = false}.

***El condicional que pregunta si el código del médico es válido.***

**if** (error == 1 || Integer.parseInt(codMed) < 1 || Integer.parseInt(codMed) > 10000) {

JOptionPane.showMessageDialog(**null**, "El código ingresado no es válido");

codMedico.setText("");

error = 0;

}

***Caso 7:*** Probamos la decisión por verdadero del condicional:

{error = 1, codMed = 0}.

***Caso 8:*** Probamos la decisión por falso del condicional:

{error = 0, codMed = 10}.

***El if que verifica que no se encontró el código ingresado.***

**if** (encontro == 0) {

JOptionPane.showMessageDialog(**null**, "El código ingresado no existe");

codMedico.setText("");

nombremed.setText("");

nombremed.setEnabled(**false**);

table2 = **new** JTable(data, columnNames);

// table2.setBounds(323, 158, 1, 1);

pane = **new** JScrollPane(table2);

pane.setBounds(38, 91, 414, 105);

getContentPane().add(pane);

}

***Caso 9:*** Probamos la decisión por verdadero del condicional: {encontro = 0}.

***Caso 10:*** Probamos la decisión por falso del condicional: { encontro = 1}.

***El While que controla si “lineasitupac” contiene los datos del paciente y el médico que lo tiene que atender.***

**while** ((lineasitupac = brsitupac.readLine()) != **null**) {

datospacmed = lineasitupac.split(" ");

***Caso 11:*** Probamos la decisión por falso del bucle leyendo una cadena vacía: { lineasitupac = null}.

***Caso 12:*** Probamos la decisión por verdadero del bucle leyendo una cadena con: { lineasitupac = 23 23}.

***El if que compara los códigos.***

**if** (Integer.parseInt(datospacmed[1]) == Integer.parseInt(datosmed[0])) {

data[cont][0] = datospacmed[0];

**try** {

archdatospac = **new** File(arch.datospac());

frdatospac = **new** FileReader(archdatospac);

brdatospac = **new** BufferedReader(frdatospac);

String lineadatospac = **new** String();

String datospac[] = **new** String[3];

***Caso 13:*** Probamos la decisión por falso del condional ingresando: {datospacmed[1] = 3 , datospacmed[1] = 2 }

***Caso 14:*** Probamos la decisión por verdadero del condional ingresando: {datospacmed[1] = 3 , datospacmed[1] = 3 }

***El While que pregunta si lineadatospac es distinto de null***

**while** ((lineadatospac = brdatospac.readLine()) != **null**) {

datospac = lineadatospac.split(" ");

***Caso 15:*** Probamos la decisión por falso del bucle leyendo una cadena vacía: { lineasitupac = null}.

***Caso 16:*** Probamos la decisión por verdadero del bucle leyendo una cadena con: { lineasitupac = 3 23}.

***El if que compara el código del paciente con los códigos que atiende el médico.***

**if** (Integer.parseInt(datospac[0]) == Integer.parseInt(datospacmed[0])) {

data[cont][1] = datospac[1];

data[cont][2] = datospac[2];

b**reak**;

}

}

frdatospac.close();

} **catch** (Exception e) {

System.out.println(e.getMessage());

}

cont++;

}

} // aca termina la busqueda de pacientes del medico

***Caso 17:*** Probamos la decisión por falso del condional ingresando:

{ datospac[0]) = 3, datospacmed[0] = 2}

***Caso 18:*** Probamos la decisión por verdadero del ciclo

{ datospac[0]) = 3, datospacmed[0] = 3}

***Se verifica que el código de paciente sea un código válido.***

if (error == 1 || Integer.parseInt(codPac) < 1 || Integer.parseInt(codPac) > 10000) {

JOptionPane.showMessageDialog(null, "El código ingresado no es válido");

codPaciente.setText("");

nombPaciente.setText("");

apePaciente.setText("");

error = 0;

}

***Valida el código ingresado:***

19 Caso: Probamos la decisión por verdadera ingresando un código de paciente válido (numérico) y dentro del rango establecido (entre 1 y 10000): {555}.

20 Caso: Probamos la decisión por falsa ingresando un código de paciente fuera del rango establecido: {0}

if (encontro) {

JOptionPane.showMessageDialog(null, "El codigo de paciente ingresado ya existe");

codPaciente.setText("");

nombPaciente.setText("");

apePaciente.setText("");

error = 0;

}

***Valida si el código del nuevo paciente ya existe:***

21 caso: Probamos la decisión por verdadera ingresando un código de paciente existente

22 caso: Probamos la decisión por falso ingresando un código de paciente nuevo

if (nomPac.matches(regex) == false || apePac.matches(regex) == false || nomPac.equals("") || apePac.equals("")) {

JOptionPane.showMessageDialog(null, "Nombre y/o Apellido inv�lido");

codPaciente.setText("");

nombPaciente.setText("");

apePaciente.setText("");

}

***Valida que se ingrese nombre y apellido válido (solo caracteres y campos no nulos):***

23 caso: probamos la decisión por verdadera ingresando un nombre y apellido válido : {Juan, Perez}

24 caso: probamos la decisión por falso ingresando un nombre y apellido con un caracter inválido: {Juan1, Perez#}

***Método: SituacionPaciente***

if (diag.length() == 0) {

JOptionPane.showMessageDialog(null, "Debe ingresar un diagn�stico");

diagnostico.setText("");

error = 0;

}

***Validación del diagnóstico:***

25 caso: probamos la decisión por verdadera ingresando un diagnóstico: {Angina}

26 caso: probamos la decisión por falso dejando el campo de diagnóstico nulo: {null}

***Método: DatosMedico***

if (espec.matches(regex) == false || espec.equals("")) {

OptionPane.showMessageDialog(null, "Especialidad inv�lida");

codMedico.setText("");

nombMedico.setText("");

apeMedico.setText("");

especialidad.setText("");

}

***Validación de la especialidad:***

27 caso: probamos la decisión por verdadera ingresando una especialidad válida: {Oncólogo}

28 caso: probamos la decisión por falso ingresando una especilidad con caracteres inválidos: {Oncólogo#}