PRUEBA CAJA BLANCA

* Todas las clases, ficheros fxml y resto de recursos están dentro del mismo Package. La estructura del directorio es muy sencillo y simple.
* En los ficheros java, tanto el main como los distintos controladores, no existen comentarios que indiquen algún tipo de información sobre la clase, métodos, funciones o variables que forman parte de ellos.
* Tampoco existen comentarios en los diferentes ficheros fxml.
* Los controladores tienen la misma estructura: imports, y dentro de la clase en primer lugar las variables de los elementos y posteriormente los métodos para controlar el comportamiento y funcionalidad de algunos de estos elementos.

Corrección de errores:

* Se deshabilita la redimensión de la ventana, en la clase main añadimos:
  + primaryStage.setResizable(**false**);
* Se disminuye el tamaño de fuente del título de la ventana principal (CASA DOMOTICA):
  + Desde SceneBuilder ajustamos el ancho de la caja (Label) al ancho de la ventana, reducimos el tamaño de la fuente del contenido a un tamaño de 28px y alineamos dicho contenido al centro de la caja.
* Se amplía el tamaño de la fuente de los label que muestran la temperatura y las luces encendidas en la ventana. Se redistribuyen estos elementos en el layout, para que no haya problemas al mostrar la información. Por ejemplo, la cadena con las luces encendidas es corta, pero si encendemos todas las luces dicha cadena gana en longitud y puede no aparecer correctamente en el layout. Se muestra un ejemplo de cómo quedaría.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

* En la ventana Temperatura se corrige el error que permitía sobrepasar los límites de temperatura. Para ello se añaden dos constantes que controlan la temperatura máxima y mínima.

//variables locales

**private** **final** **static** **int** ***TEMP\_MAXIMA*** = 30;

**private** **final** **static** **int** ***TEMP\_MINIMA*** = 15;

Se modifica la función para que controlar que no se sobrepase.

@FXML

**void** btnBajar(MouseEvent event) {

**if**(*temperatura* > ***TEMP\_MINIMA***) {

*temperatura*--;

String temperaturaMenos = "" + *temperatura*;

lblTemperatura.setText(temperaturaMenos);

}

}

@FXML

**void** btnSubir(MouseEvent event) {

**if**(*temperatura* < ***TEMP\_MAXIMA***) {

*temperatura*++;

String temperaturaMas = "" + *temperatura*;

lblTemperatura.setText(temperaturaMas);

}

}

* En la ventana Llums:
  + Interfaz de usuario gráfica, Aplicación

    Descripción generada automáticamenteSe aumenta el tamaño de fuente del label que muestra las luces encendidas, así como la distribución de dicha información en el layout.

- Para corregir la inserción de las luces encendidas:

* Declaramos 2 variables para controlar si cada una de las luces esta encendida así como una variable de tipo String que rellene las luces encendidas.

//variables globales

**private** **static** **boolean** *interiorEncendida* = **false**;

**private** **static** **boolean** *jardinEncendida* = **false**;

**private** **static** String *cadena* = "";

* Modificamos los métodos y añadimos uno nuevo para comprobar correctamente que se modifica la cadena con las luces encendidas

@FXML

**void** cmbLlumInterior(ActionEvent event) {

*interiorEncendida* = cmbInterior.isSelected() ? **true** : **false**;

comprobarLuces();

}

@FXML

**void** cmbLlumJardi(ActionEvent event) {

*jardinEncendida*= cmbJardi.isSelected() ? **true** : **false**;

comprobarLuces();

}

**void** comprobarLuces() {

*cadena* = "";

**if**(*interiorEncendida*) *cadena* = *cadena* + "INTERIOR";

**if**(*jardinEncendida*) *cadena* = *cadena* + " JARDÍ";

**if**(!*interiorEncendida* && !*jardinEncendida*) *cadena* = "CAP";

lblRespuesta.setText(*cadena*);

}

* Al volver a la ventana principal, la información de las Llums aparece en el label de temperatura, con lo cual modificamos la llamada al método:

control.setLlumns(lblRespuesta.getText());

* El ultimo problema que aparece es que cuando volvemos a la ventana principal, la información de los label se formatea. Es decir, si volvemos desde la ventana Llums, la temperatura se formatea, y a la inversa. Como solución hay que pasar la información de la luz y la temperatura conforme nos movamos en las ventanas.

Para solucionarlo:

En el controlador principal añadimos 2 llamadas a métodos para pasar los datos que recogemos.

@FXML

**void** btnLlums(ActionEvent event) **throws** IOException {

FXMLLoader fxmlLoader = **new** FXMLLoader();

fxmlLoader.setLocation(getClass().getResource("VistaLlums.fxml"));

Stage stage = (Stage) btnLlums.getScene().getWindow();

Scene scene = **new** Scene(fxmlLoader.load());

stage.setScene(scene);

ControladorTemperatura conTemp = (ControladorTemperatura) fxmlLoader.getController();

conTemp.enviaDatos(lblLlums.getText(), lblTemperatura.getText());

}

@FXML

**void** btnTemperatura(ActionEvent event) **throws** IOException {

FXMLLoader fxmlLoader = **new** FXMLLoader();

fxmlLoader.setLocation(getClass().getResource("VistaTemperatura.fxml"));

Stage stage = (Stage) btnTemperatura.getScene().getWindow();

Scene scene = **new** Scene(fxmlLoader.load());

stage.setScene(scene);

ControladorLlums conTemp = (ControladorLlums) fxmlLoader.getController();

conTemp.enviaDatos(lblLlums.getText(), lblTemperatura.getText());

}

* En el controlador de temperatura, cogemos los datos y los almcenamos de la siguiente manera

**public** **void** enviaDatos(String llum, String temp) {

//pasamos la temperatura que hay en la ventana principal

lblTemperatura.setText(temp);

//almacenamos la lluma en una variable global de la clase

*llumGlobal* = llum;

}

* Una vez modificada la temperatura, solo tenemos que devolver a la ventana principal la temperatura modificada y las Llums tal como las hemos recibido.

@FXML

**void** btnAceptar(ActionEvent event) **throws** IOException {

FXMLLoader fxmlLoader = **new** FXMLLoader();

fxmlLoader.setLocation(getClass().getResource("VistaPrincipal.fxml"));

Stage stage = (Stage) btnAceptar.getScene().getWindow();

Scene scene = **new** Scene(fxmlLoader.load());

stage.setScene(scene);

ControladorVistaPrincipal control = (ControladorVistaPrincipal) fxmlLoader.getController();

control.setTemperatura(lblTemperatura.getText());

control.setLlumns(*llumGlobal*);

}

* En el controlador de las llums, realizamos el mismo proceso

**public** **void** enviaDatos(String llum, String temp) {

//mostramos las luces en su label

lblRespuesta.setText(llum);

//alamcenamos la temperatura en una variable global

*tempGlobal* = temp;

}

@FXML

**void** btnAceptar(ActionEvent event) **throws** IOException {

FXMLLoader fxmlLoader = **new** FXMLLoader();

fxmlLoader.setLocation(getClass().getResource("VistaPrincipal.fxml"));

Stage stage = (Stage) btnAceptar.getScene().getWindow();

Scene scene = **new** Scene(fxmlLoader.load());

stage.setScene(scene);

ControladorVistaPrincipal control = (ControladorVistaPrincipal) fxmlLoader.getController();

control.setLlumns(lblRespuesta.getText());

control.setTemperatura(*tempGlobal*);

}

Y con todos estos cambios ya tenemos la app totalmente funcional