package punt;

import java.net.URL;

import java.util.ResourceBundle;

import javafx.beans.binding.Bindings;

import javafx.beans.property.DoubleProperty;

import javafx.beans.property.SimpleDoubleProperty;

import javafx.event.ActionEvent;

import javafx.fxml.FXML;

import javafx.fxml.Initializable;

import javafx.scene.Cursor;

import javafx.scene.control.Label;

import javafx.scene.input.KeyCode;

import javafx.scene.input.KeyEvent;

import javafx.scene.input.MouseEvent;

import javafx.scene.layout.GridPane;

import javafx.scene.shape.Circle;

/\*\*

\*

\* @author Anna

\*/

public class FXMLPuntController implements Initializable {

private Label label;

@FXML

private GridPane graella;

@FXML

private Circle cercle;

private int size;

private static int sizeCol, sizeRow;

private DoubleProperty height\_Grid = new SimpleDoubleProperty();

private DoubleProperty width\_Grid = new SimpleDoubleProperty();

private int celdaX, celdaY;

private double deltaX;

private double deltaY;

public void calculaCelda(double x\_ratoli, double y\_ratoli){

celdaX = (int)(x\_ratoli / (width\_Grid.getValue() / sizeCol)) ;

celdaY = (int)(y\_ratoli / (height\_Grid .getValue()/ sizeRow)) ;

}

@Override

public void initialize(URL url, ResourceBundle rb) {

size = graella.getColumnConstraints().size();

cercle.radiusProperty().bind(Bindings.min(Bindings.divide(graella.widthProperty(),size\*2), Bindings.divide(graella.heightProperty(), size\*2)));

height\_Grid.bind(graella.heightProperty());

width\_Grid.bind(graella.widthProperty());

sizeCol = size;

sizeRow = graella.getRowConstraints().size();

}

@FXML

private void moureCercle(KeyEvent event) {

KeyCode tecla = event.getCode();

switch(tecla) {

case UP:

graella.setRowIndex(cercle, (graella.getRowIndex(cercle) - 1 + size) % size);

break;

case DOWN:

graella.setRowIndex(cercle, (graella.getRowIndex(cercle) + 1 + size) % size);

break;

case RIGHT:

graella.setColumnIndex(cercle, (graella.getColumnIndex(cercle) + 1 + size) % size);

break;

case LEFT:

graella.setColumnIndex(cercle, (graella.getColumnIndex(cercle) - 1 + size) % size);

break;

}

}

@FXML

private void soltatCercle(MouseEvent event) {

cercle.setTranslateX(0);

cercle.setTranslateY(0);

calculaCelda(event.getSceneX(),event.getSceneY());

graella.setRowIndex(cercle, celdaY);

graella.setColumnIndex(cercle, celdaX);

}

@FXML

private void arrastrarCercle(MouseEvent event) {

cercle.setTranslateX(event.getSceneX()- deltaX);

cercle.setTranslateY(event.getSceneY() - deltaY);

}

@FXML

private void eixirCercle(MouseEvent event) {

cercle.setCursor(Cursor.DEFAULT);

}

@FXML

private void entrarCercle(MouseEvent event) {

cercle.setCursor(Cursor.HAND);

}

@FXML

private void polsatCercle(MouseEvent event) {

deltaX = cercle.getLayoutX();

deltaY = cercle.getLayoutY();

}

}