Facultatea Calculatoare, Informatica si Microelectronica

Universitatea Tehnica a Moldovei

Medii Interactive de Dezvoltare a Produselor Soft

Lucrarea de laborator#2

GUI Development

Autor: st. gr. TI-154
Octavian Coroletchi

lector asistent:
Victor Gojin

Chisinau 2017

Laboratory work #1

1 Scopul lucrarii de laborator

Familiarizarea cu GUI

2 Obiective

- Realizeaza un simplu GUI Calculator
- Operatiile simple: +,-,*,/,putere,radical,InversareSemn(+/-),operatii cu numere zecimale.
- Divizare proiectului in doua module - Interfata grafica(Modul GUI) si Modulul de baza(Core Module).

3 Laboratory work implementation

3.1 Tasks and Points

a) Realizarea unui simplu GUI calculator care suporta functiile de baza +, -, /, *

b) Realizarea operatiilor de: putere, radical si InversareSemn (+/-).

c) Adaugarea posibilitatii de utilizare a operatiilor cu numere zecimale.

d) Divizarea proiectului in doua module - Interfata grafica (Modul GUI) si Modulul de baza (Core Module).

3.2 Analiza lucrarii de laborator

Click on ”Link” or copy `https://github.com/OctavianCoroletchiTI154/MIDPS.git` pentru repozitoriul meu.

Task a:
Pentru inceput, m-am familiarizat cu limbajul Java, care l-am folosit pentru implementarea programului meu. Pentru realizarea programului meu, am folosit 2 clase(Main si MainController), 1 fisier .css(application), si 1 fisier .fxml(MainInterface) -¿ vezi Fig a-1. Clasa main este folosita pentru a implementa fereastra si fisierele grafice si functionalitatea -¿ vezi Listing 1. Mai apoi am creat clasa MainController, pentru a realiza codul functionalitatii tuturor elementelor grafice si tehnice. -¿ vezi Listing 2. Fisierul MainInterface.fxml, este creat cu ajutorul framework-ului JavaFX, implementat cu ajutorul JavaFX Scene Builder. -¿ vezi Listing 3 si Fig a-2. Si in final, fisierul application.css, este un fisier cascading style sheets, care realizeaza partea grafica. Cu ajutorul limbajului CSS, putem edita cele mai mici detalii ale interfetei vizuale -¿ vezi codul Listing 4.

Task b:

Operatiile de putere, radical si inversare semn au fost implementate cu ajutorul bibliotecii Math. Am implementat cazuri de exceptie, cum ar fi ”Radical dintr-un numar negativ nu se face” -¿ vezi Fig b-1 sau impartirea la zero -¿ vezi Fig b-2.

Task c:

Pentru posibilitatea functionarii cu numere zecimale, am folosit variabile Double, iar pentru a rezolva problema in care o variabila nu este corect afisata, de ex. in loc de 3.0, este afisata 2.99999. Am rezolvat aceasta problema, cu ajutorul bibliotecii BigDecimal, care rotungeste variabila pentru un anumit numar de cifre zecimale dupa virgula, dupa anumite reguli.

Task d:

Proiectul a fost implementat cu succes in parti diferite, grafica si functional. Modul GUI este MainInterface.fxml, in care dupa cum am specificat anterios, sunt implementate toate obiectele grafice. In fisierul MainController.java, am implementat functionalul acestor obiecte, in care i-am descris fiecaruia dintre ele un anumit scop.

3.3 Imagini

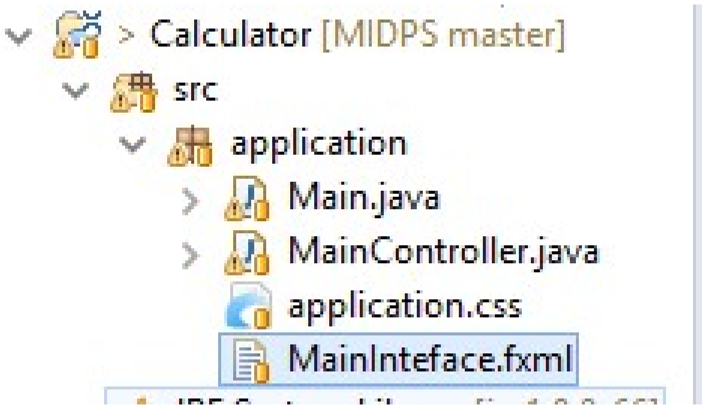

Fig a-1 - ”Ierarhia structurii aplicatiei”

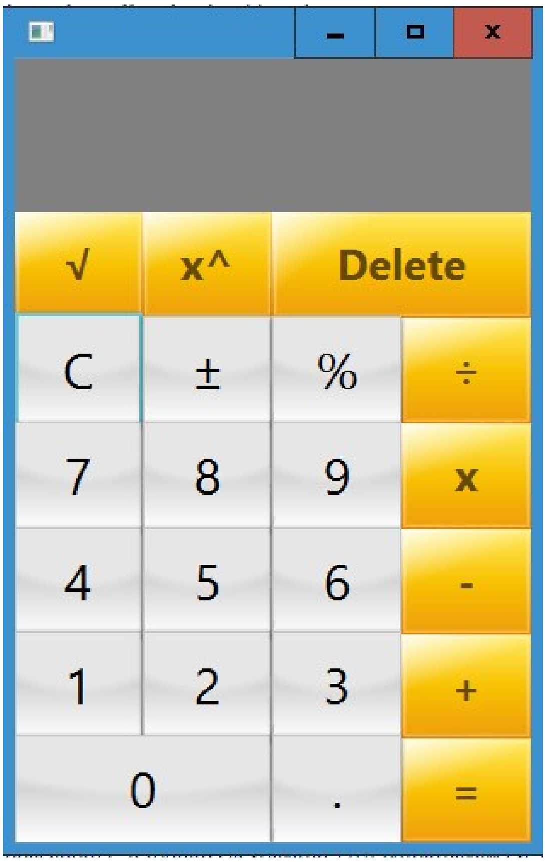

Fig a-2- ”GUI Aplicatiei”

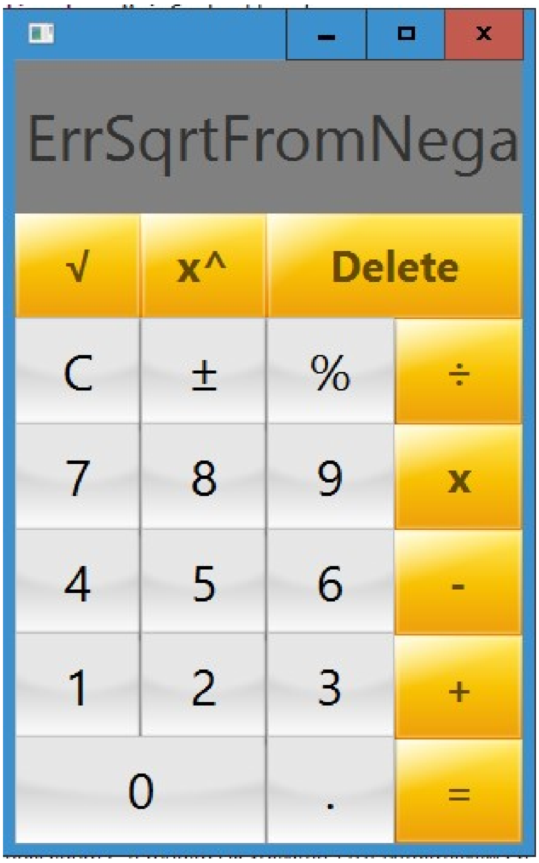
ErrSqrtFromNega

Fig b-1 - ”Exceptia radicalului”

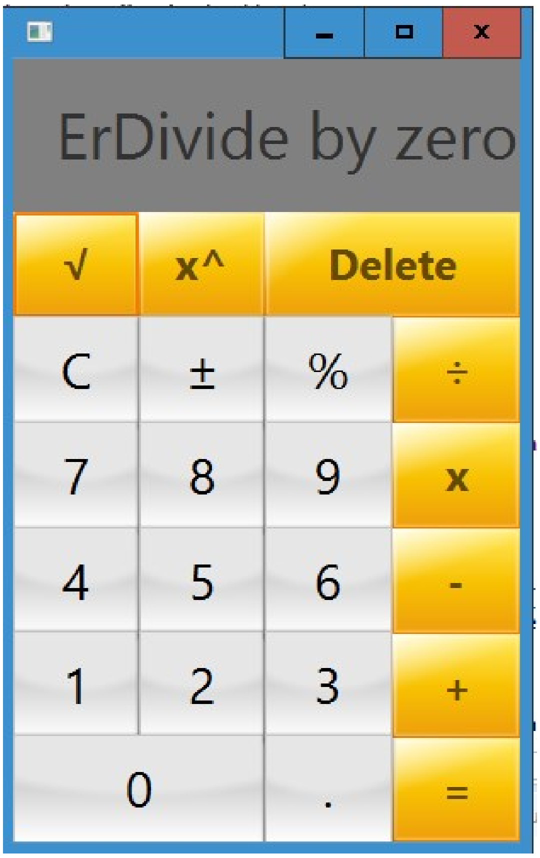
ErDivide by zero

Fig b-2 - ”Exceptia impartirii la zero”

```java
package application;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.stage.Stage;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.layout.BorderPane;


public class Main extends Application {
  @Override
  public void start(Stage primaryStage) {
    try {
      //BorderPane root = new BorderPane();

      Parent root = FXMLLoader.load(getClass().getResource("/application/MainInteface.fxml"));
      Scene scene = new Scene(root);
      scene.getStylesheets().add(getClass().getResource("/application/application.css").toExternalForm());
      primaryStage.setScene(scene);
      primaryStage.show();
    } catch(Exception e) {
      e.printStackTrace();
    }
  }

  public static void main(String[] args) {
    launch(args);
  }
}
```

Listing 1 – Main.java

```java
package application;

import java.math.BigDecimal;
import java.math.RoundingMode;

import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.scene.control.*;

public class MainController {

  @FXML
  private Label display;
  private String output = "";
  private double firstNum = 0;
  private double secondNum;
  private String operatorInput;
  private double result;
  private String text;
  private boolean isDel = false;
  private boolean decimal = false;
  private boolean multiply = false;
  private boolean start = false;

  @FXML
  public void processDel (ActionEvent event){
    if (display.getText().equals("")) return;
    isDel = true;
    processNum(event);
  }

  @FXML
  public void processNum (ActionEvent event) {
    text = ((Button)event.getSource()).getText();
    start = true;
  if (isDel == true){
    text = display.getText();
    text = text.substring(0, text.length() - 1);
    isDel = false;
    display.setText(text);
  }else{
  if(display.getText().equals("0") || display.getText().equals("ErDivide by zero") || display.getText().equals("ErrSqrtFromNegative")){
    display.setText(text);
  }
  display.setText(display.getText() + text);
  text = display.getText();
}}

@FXML
public void processOperators (ActionEvent event) {
  double value = 0;
  decimal = false;

  operatorInput = ((Button)event.getSource()).getText();
  switch (operatorInput){
  case "C":
    display.setText("");
    start = false;
    break;
  case "+/-":
    if (multiply == false){
    value = BigDecimal.valueOf(Double.parseDouble(display.getText())).doubleValue();
    value = value * (-1);
    display.setText(String.valueOf(value));
    }
    else {
      value = BigDecimal.valueOf(Double.parseDouble(display.getText())).doubleValue();
      value = value * (-1);
      secondNum = value;
      display.setText(String.valueOf(value));
    }
    break;
  case "sqrt":
    firstNum = BigDecimal.valueOf(Double.parseDouble(display.getText())).doubleValue();
    if (firstNum < 0){
      display.setText("ErrSqrtFromNegative");
    }else {
    result = Math.sqrt(firstNum);
    if(String.valueOf(result).endsWith(".0"))
      display.setText(String.valueOf((int)result));
    else
      display.setText(String.valueOf(result));
    }
    break;
  case "%":
    if(multiply == true){
      secondNum = BigDecimal.valueOf(Double.parseDouble(display.getText())).doubleValue();
      result = firstNum * secondNum / 100;
      display.setText(String.valueOf(result));
      multiply = false;
    }else{
    firstNum = BigDecimal.valueOf(Double.parseDouble(display.getText())).doubleValue();
    result = firstNum / 100;
    display.setText(String.valueOf(result));
    }
    break;
  case "/":
  case "x":
  case "-":
  case "+":
  case "x^":
    multiply = true;
    firstNum = Double.parseDouble(display.getText());
    result = firstNum;
    display.setText("");
    break;
  }
}

@FXML
public void processDecimal (ActionEvent event) {
  if(decimal == false){
  display.setText(display.getText() + ((Button)event.getSource()).getText());
  decimal=true;
  }
}

@FXML
public void processEqual (ActionEvent event) {
  if(start == false) {
    display.setText("0");
    return;
  }
  if (multiply == true){
  secondNum = Double.parseDouble(display.getText());
  multiply = false;
  }
  switch (operatorInput){
  case "/":
    if(secondNum == 0){
      display.setText("ErDivide by zero");
      return;
    }
    else{
    result = BigDecimal.valueOf(result / secondNum).setScale(9, RoundingMode.
 HALF_UP).doubleValue();
    }
    break;
  case "x":
    result = BigDecimal.valueOf(result * secondNum).setScale(9, RoundingMode.
 HALF_UP).doubleValue();
    break;
  case "-":
    result = BigDecimal.valueOf(result - secondNum).setScale(9, RoundingMode.
 HALF_UP).doubleValue();
    break;
  case "+":
    result = BigDecimal.valueOf(result + secondNum).setScale(9, RoundingMode.
 HALF_UP).doubleValue();;
    break;
  case "x^":
    result = BigDecimal.valueOf(Math.pow(result, secondNum)).setScale(9,
 RoundingMode.HALF_UP).doubleValue();
    break;
  default:
    display.setText("Invalid symbol");
    break;
  }
  if (String.valueOf(result).endsWith(".0"))
    display.setText(String.valueOf((int)result));
  else
  display.setText(String.valueOf(result));

}
}
```

Listing 2 – MainController.java

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.layout.AnchorPane?>
```

```
<?import javafx.scene.layout.ColumnConstraints?>
<?import javafx.scene.layout.GridPane?>
<?import javafx.scene.layout.RowConstraints?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="460.0" prefWidth="303.0" xmlns="http://javafx.com/javafx/8.0.111" xmlns:fx="http://javafx.com/fxml/1" fx:controller="application.MainController">
   <children>
      <Label fx:id="display" alignment="CENTER_RIGHT" layoutY="-32.0" prefHeight="50.0" prefWidth="303.0" style="-fx-background-color: gray;" textOverrun="CLIP" AnchorPane.bottomAnchor="366.0" AnchorPane.leftAnchor="0.0" AnchorPane.rightAnchor="0.0" AnchorPane.topAnchor="24.0">
         <font>
            <Font size="38.0" />
         </font></Label>
      <GridPane layoutY="291.0" prefHeight="370.0" prefWidth="300.0" AnchorPane.bottomAnchor="0.0" AnchorPane.leftAnchor="0.0" AnchorPane.rightAnchor="0.0">
        <columnConstraints>
            <ColumnConstraints hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" />
            <ColumnConstraints hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" />
          <ColumnConstraints hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" />
          <ColumnConstraints hgrow="SOMETIMES" minWidth="10.0" prefWidth="100.0" />
        </columnConstraints>
        <rowConstraints>
          <RowConstraints minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" />
            <RowConstraints minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" />
            <RowConstraints minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" />
            <RowConstraints minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" />
          <RowConstraints minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" />
          <RowConstraints minHeight="10.0" prefHeight="30.0" vgrow="SOMETIMES" />
        </rowConstraints>
         <children>
            <Button id="operator" mnemonicParsing="false" onAction="#processOperators" prefHeight="65.0" prefWidth="90.0" text="sqrt">
               <font>
                  <Font size="30.0" />
               </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processOperators" prefHeight="65.0" prefWidth="118.0" text="C" GridPane.rowIndex="1">
               <font>
                  <Font size="30.0" />
```

```
                </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="111.0" text="7" GridPane.rowIndex="2">
                <font>
                    <Font size="30.0" />
                </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="128.0" text="4" GridPane.rowIndex="3">
                <font>
                    <Font size="30.0" />
                </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="118.0" text="1" GridPane.rowIndex="4">
                <font>
                    <Font size="30.0" />
                </font></Button>
            <Button id="operator" mnemonicParsing="false" onAction="#processOperators" prefHeight="80.0" prefWidth="89.0" text="x^" GridPane.columnIndex="1">
                <font>
                    <Font size="30.0" />
                </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processOperators" prefHeight="65.0" prefWidth="121.0" text="+/-" GridPane.columnIndex="1" GridPane.rowIndex="1">
                <font>
                    <Font size="30.0" />
                </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="119.0" text="8" GridPane.columnIndex="1" GridPane.rowIndex="2">
                <font>
                    <Font size="30.0" />
                </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="156.0" text="5" GridPane.columnIndex="1" GridPane.rowIndex="3">
                <font>
                    <Font size="30.0" />
                </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="126.0" text="2" GridPane.columnIndex="1" GridPane.rowIndex="4">
                <font>
                    <Font size="30.0" />
                </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="184.0" text="0" GridPane.columnSpan="2" GridPane.rowIndex="5">
```

```
               <font>
                  <Font size="30.0" />
               </font>
            </Button>
            <Button id="operator" mnemonicParsing="false" onAction="#processDel" prefHeight="65.0" prefWidth="183.0" text="Delete" GridPane.columnIndex="2" GridPane.columnSpan="2">
               <font>
                  <Font size="30.0" />
               </font></Button>
            <Button id="operator" mnemonicParsing="false" onAction="#processOperators" prefHeight="65.0" prefWidth="111.0" text="/" GridPane.columnIndex="3" GridPane.rowIndex="1">
               <font>
                  <Font size="30.0" />
               </font>
            </Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processOperators" prefHeight="65.0" prefWidth="115.0" text="\%" GridPane.columnIndex="2" GridPane.rowIndex="1">
               <font>
                  <Font size="30.0" />
               </font>
            </Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="135.0" text="9" GridPane.columnIndex="2" GridPane.rowIndex="2">
               <font>
                  <Font size="30.0" />
               </font></Button>
            <Button id="operator" mnemonicParsing="false" onAction="#processOperators" prefHeight="65.0" prefWidth="150.0" text="x" GridPane.columnIndex="3" GridPane.rowIndex="2">
               <font>
                  <Font size="30.0" />
               </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="86.0" prefWidth="100.0" text="6" GridPane.columnIndex="2" GridPane.rowIndex="3">
               <font>
                  <Font size="30.0" />
               </font></Button>
            <Button id="nums" mnemonicParsing="false" onAction="#processNum" prefHeight="65.0" prefWidth="143.0" text="3" GridPane.columnIndex="2" GridPane.rowIndex="4">
               <font>
                  <Font size="30.0" />
               </font></Button>
            <Button id="operator" mnemonicParsing="false" onAction="#
```

```
processOperators" prefHeight="65.0" prefWidth="116.0" text="-" GridPane.columnIndex="3" GridPane.rowIndex="3">
                    <font>
                        <Font size="30.0" />
                    </font>
                </Button>
                <Button id="operator" mnemonicParsing="false" onAction="#processOperators" prefHeight="71.0" prefWidth="76.0" text="+" GridPane.columnIndex="3" GridPane.rowIndex="4">
                    <font>
                        <Font size="30.0" />
                    </font>
                </Button>
                <Button id="operator" cache="true" mnemonicParsing="false" onAction="#processEqual" prefHeight="94.0" prefWidth="101.0" text="=" GridPane.columnIndex="3" GridPane.rowIndex="5">
                    <font>
                        <Font size="30.0" />
                    </font></Button>
                <Button id="nums" mnemonicParsing="false" onAction="#processDecimal" prefHeight="65.0" prefWidth="122.0" text="." GridPane.columnIndex="2" GridPane.rowIndex="5">
                    <font>
                        <Font size="30.0" />
                    </font>
                </Button>
            </children>
        </GridPane>
    </children>
    <padding>
        <Insets top="-28.0" />
    </padding>
</AnchorPane>
```

Listing 3 – MainInterface.fxml

```
/* JavaFX CSS - Leave this comment until you have at least create one rule which uses -fx-Property */

#operator{
  -fx-background-color:
        linear-gradient(#ffd65b, #e68400),
        linear-gradient(#ffef84, #f2ba44),
        linear-gradient(#ffea6a, #efaa22),
        linear-gradient(#ffe657 0%, #f8c202 50%, #eea10b 100%),
        linear-gradient(from 0% 0% to 15% 50%, rgba(255,255,255,0.9), rgba(255,255,255,0));
```

```
    -fx-background-radius: 0;
    -fx-background-insets: 0,1,2,3,0;
    -fx-text-fill: #654b00;
    -fx-font-weight: bold;
    -fx-font-size: 25px;
    -fx-padding: 10 20 10 20;
}
#operator:hover{
  -fx-border-style:solid;
  -fx-border-width:2px;
  -fx-border-color:#f57a00;
  -fx-border-radius:1px;
}
#nums:hover{
  -fx-border-style:solid;
  -fx-border-width:2px;
  -fx-border-color:#62c8d5;
  -fx-border-radius:1px;
}
#nums{
  -fx-background-color:
        #c3c4c4,
        linear-gradient(#d6d6d6 50%, white 100%),
        radial-gradient(center 50% -40%, radius 200%, #e6e6e6 45%, rgba(230,230,230,0) 50%);
    -fx-background-radius: 0;
    -fx-background-insets: 0,1,1;
    -fx-text-fill: black;
    -fx-effect: dropshadow( three-pass-box , rgba(0,0,0,0.6) , 3, 0.0 , 0 , 1 );
}
```

Listing 4 – application.css

Concluzie

In concluzie, am reusit sa creez o aplicatie (Calculator), aplicabila pentru Unix, Windows, ce permite cu usurinta de calculat simple necesitati, dar si adaugari precum radical, ridicare la putere, procent. Am prevazut situatii de exceptie precum impartirea la zero, sau radical dintr-un numar negativ. Intr-un final am obtinut o aplciatie usor de folosit, user-friendly, placuta la vizualizare. Cel mai important, am inteles unele principii in Java, si framework-ul sau JavaFX, iar in continuare voi putea crea lucruri mai complexe si interesante.

References

1 https://docs.oracle.com/javase/tutorial/

2 http://docs.oracle.com/javase/8/javase-clienttechnologies.htm

3 https://www.tutorialspoint.com/java/

4 https://www.youtube.com/