Design of Software Engineering Lesson 2

Content

Use Cases

Use Case 1: UpLoad Data From File.

Participating Actor:

Initiated by Scientist

Entry Condition:

None

Exit Criteria:

Dataset is updated and Cartesian graph is displayed.

Flow of Events:

1. Scientist requests to upload data into the system from a file.
2. System displays LoadDataForm to Scientist.
3. Scientist chooses the data file and submits LoadDataForm.
4. System displays updated dataset and presents Cartesian graph as default.

Use Case 2: Add A Data Pair.

Participating Actor:

Initiated by Scientist

Entry Condition:

None

Exit Criteria:

Dataset and graph are updated.

Flow of Events:

1. Scientist fills the new data pair and makes an adding requests to system.
2. System displays updated dataset and presents updated graph (including trend line, equation if needed).

Use Case 3: Delete Data Pair.

Participating Actor:

Initiated by Scientist

Entry Condition:

Dataset is not empty in MainForm.

Exit Criteria:

Dataset and current graph are updated.

Flow of Events:

1. Scientist chooses data pairs and makes a request to delete them.
2. System responds by displaying DeleteConfirmationForm.
3. Scientist submits his confirmation with DeleteConfirmationForm.
4. System displays updated dataset and presents updated graph (including trend line, equation if needed).

Use Case 4: Edit Data Pair

Participating Actor:

Initiated by Scientist

Entry Condition:

Dataset is not empty in MainForm.

Exit Criteria:

Dataset and current graph are updated.

Flow of Events:

1. Scientist chooses a data pair then requests to update it.
2. System displays updated dataset and presents updated graph (including trend line, equation if needed).

Use Case 5: Switch to Column Graph.

Participating Actor:

Initiated by Scientist

Entry Condition:

Current graph is Cartesian plot

Exit Criteria:

Column graph is displayed

Flow of Events:

1. Scientist requests to switch graph.
2. System hides the Cartesian plot currently displayed and shows the Column graph.

Use Case 6: Switch to Cartesian Plot.

Participating Actor:

Initiated by Scientist

Entry Condition:

Current graph is Column graph

Exit Criteria:

Cartesian plot is displayed

Flow of Events:

1. Scientist requests to switch graph.
2. System hides the Column graph currently displayed and shows a Cartesian plot based on current data set.

Use Case 7: Show Trend Line

Participating Actor:

Initiated by Scientist

Entry Condition:

Cartesian plot is displayed.

Trend line is hidden.

Exit Criteria:

Trend line is shown on the Cartesian plot.

Flow of Events:

1. Scientist requests to show trend line and its equation.
2. System displays linear regression trend line and its equation as default.

Use Case 8: Hide Trend Line

Participating Actor:

Initiated by Scientist

Entry Condition:

Cartesian plot is displayed

Trend line is displayed.

Exit Criteria:

Trend line and its equation are hidden on the Cartesian plot

Flow of Events:

1. Scientist requests to hide trend line and its equation.
2. System hides the trend line and its equation on theCartesian plot.

Use Case 9: Show Equation

Participating Actor:

Initiated by Scientist

Entry Condition:

Cartesian plot is displayed

Trend line is displayed

Trend line's equation is hidden.

Exit Criteria:

Trend line's equation is shown on the Cartesian plot

Flow of Events:

1. Scientist requests to hide trend line and its equation.
2. System shows trend line's equation on the Cartesian plot.

Use Case 10: Hide Equation

Participating Actor:

Initiated by Scientist

Entry Condition:

Cartesian plot is displayed.

Trend line is displayed.

Trend line's equation is displayed.

Exit Criteria:

Trend line's equation is hidden on the Cartesian plot

Flow of Events:

1. Scientist requests to hide trend line or only hide its equation.
2. System hides trend line's equation on the Cartesian plot.

Use Case 11: Show XY Axes Value

Participating Actor:

Initiated by Scientist

Entry Condition:

X and Y axes values are hidden

Exit Criteria:

X and Y axes values are shown.

Flow of Events:

1. Scientist requests to show values on X and Y Axes.
2. System shows X and Y Axes values.

Use Case 11: Hide XY Axes Value

Participating Actor:

Initiated by Scientist

Entry Condition:

X and Y axes values are shown

Exit Criteria:

X and Y axes values are hidden

Flow of Events:

1. Scientist requests to hide values on X and Y Axes.
2. System hides X and Y axis values.

Use Case 12: Show Background Horizontal Lines

Participating Actor:

Initiated by Scientist

Entry Condition:

Background horizontal lines are hidden

Exit Criteria:

Background horizontal lines are shown.

Flow of Events:

1. Scientist requests to show background horizontal lines.
2. System shows background horizontal lines.

Use Case 13: Hide Background Horizontal Lines

Participating Actor:

Initiated by Scientist

Entry Condition:

Background horizontal lines are shown

Exit Criteria:

Background horizontal lines are hidden

Flow of Events:

1. Scientist requests to hide background horizontal lines.
2. System hides background horizontal lines.

Use Case 14: Save Data To File

Participating Actor:

Initiated by Scientist

Entry Condition:

Current data set is not empty.

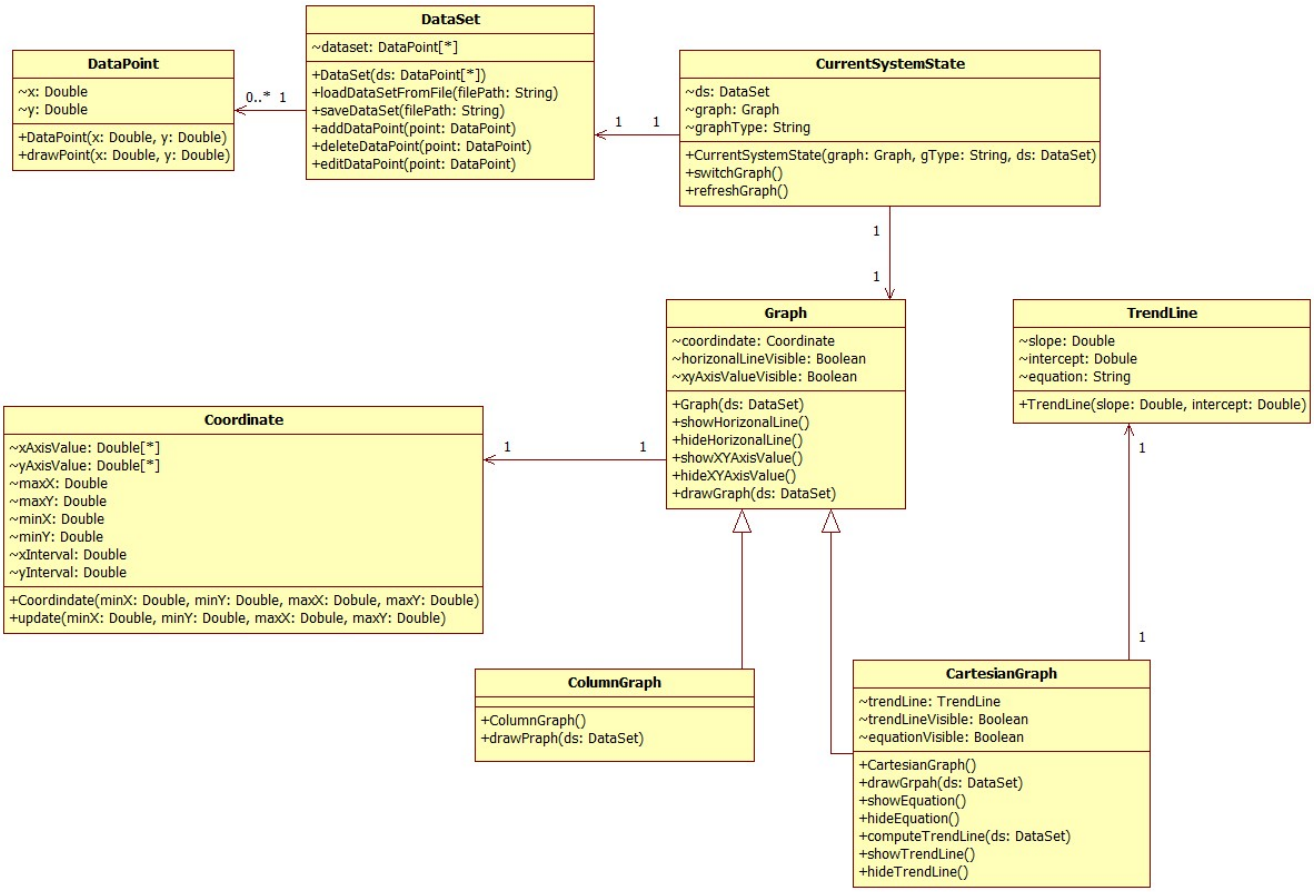
Exit Criteria:

Data is saved to a file.

Flow of Events:

1. Scientist makes a save data request in MainForm.
2. System displays SaveDataForm.
3. Scientists fills in SaveDataForm and submits the form.
4. System writes the dataset to a file chosen by scientist and returns to MainForm.

UML Diagram

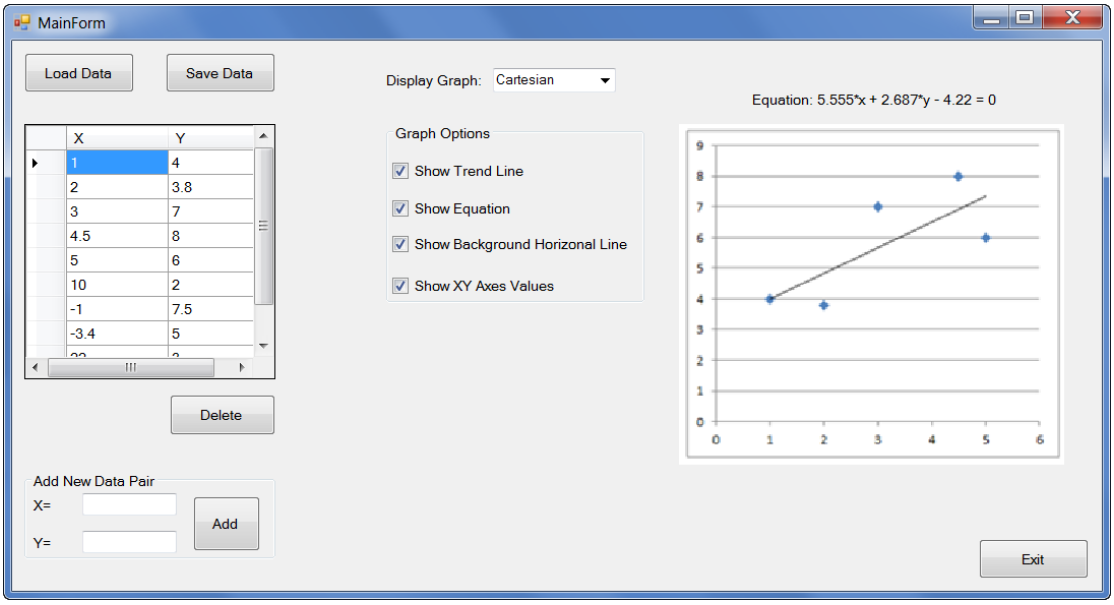

DataPoint
~x: Double
~y: Double
+DataPoint(x: Double, y: Double)
+drawPoint(x: Double, y: Double)
0..*
1
DataSet
~dataset: DataPoint[*]
+DataSet(ds: DataPoint[*])
+loadDataSetFromFile(filePath: String)
+saveDataSet(filePath: String)
+addDataPoint(point: DataPoint)
+deleteDataPoint(point: DataPoint)
+editDataPoint(point: DataPoint)
1
1
CurrentSystemState
~ds: DataSet
~graph: Graph
~graphType: String
+CurrentSystemState(graph: Graph, gType: String, ds: DataSet)
+switchGraph()
+refreshGraph()
1
1
Graph
~coordindate: Coordinate
~horizonalLineVisible: Boolean
~xyAxisValueVisible: Boolean
+Graph(ds: DataSet)
+showHorizonalLine()
+hideHorizonalLine()
+showXYAxisValue()
+hideXYAxisValue()
+drawGraph(ds: DataSet)
TrendLine
~slope: Double
~intercept: Dobule
~equation: String
+TrendLine(slope: Double, intercept: Double)
Coordinate
~xAxisValue: Double[*]
~yAxisValue: Double[*]
~maxX: Double
~maxY: Double
~minX: Double
~minY: Double
~xInterval: Double
~yInterval: Double
+Coordindate(minX: Double, minY: Double, maxX: Dobule, maxY: Double)
+update(minX: Double, minY: Double, maxX: Dobule, maxY: Double)
1
1
1
1
ColumnGraph
+ColumnGraph()
+drawPraph(ds: DataSet)
CartesianGraph
~trendLine: TrendLine
~trendLineVisible: Boolean
~equationVisible: Boolean
+CartesianGraph()
+drawGrpah(ds: DataSet)
+showEquation()
+hideEquation()
+computeTrendLine(ds: DataSet)
+showTrendLine()
+hideTrendLine()

GUI Forms

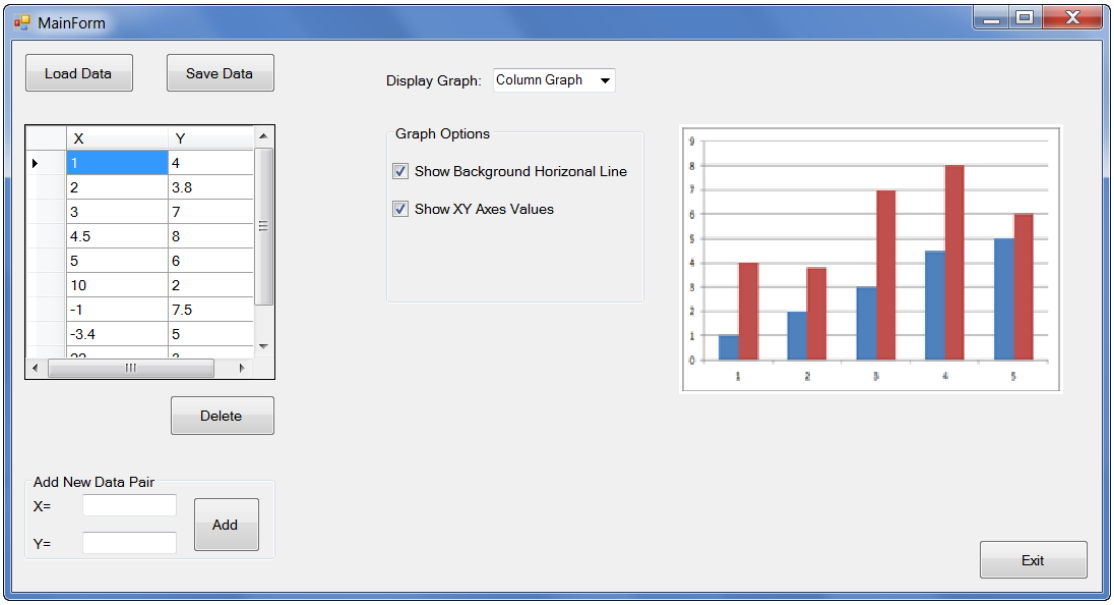
1. MainForm

In the MainForm, the Scientist double clicks the cell to change value.

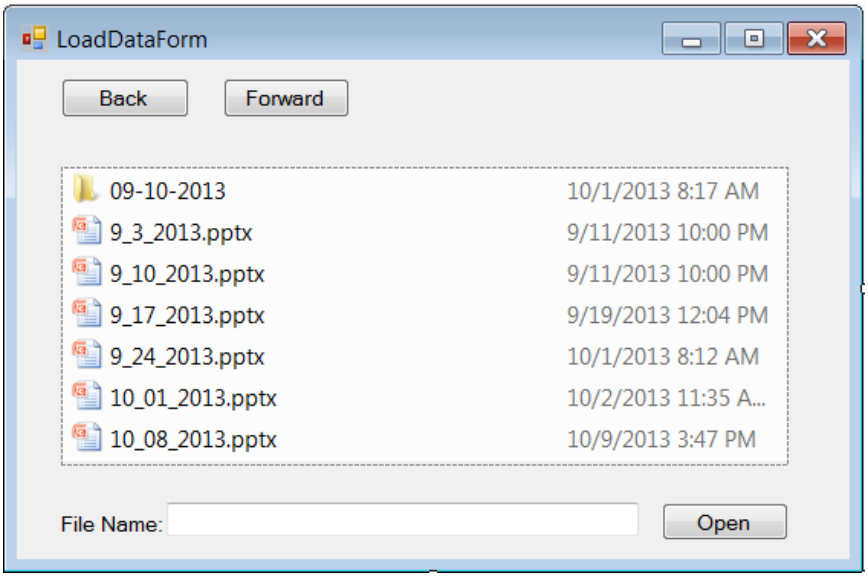
a. MainForm with Cartesian Graph



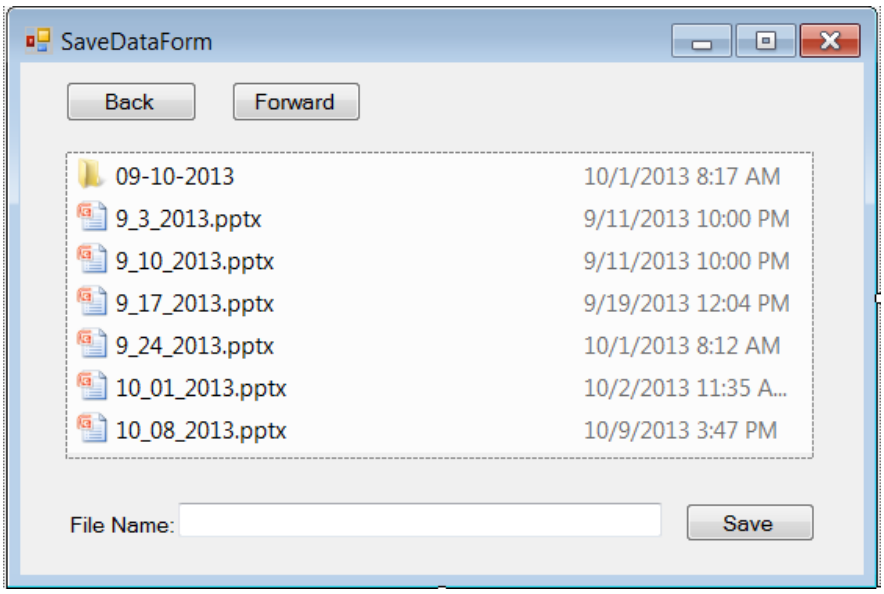
b. MainForm with Column Graph



2. LoadDataForm



3. SaveDataForm



4. DeleteConfirmationForm