EGCO 213

Group Project 3 – GUI Application

The project can be done in a group of <=5 students. Each group must do the project by themselves. This is a freestyle project, so no 2 groups should have the same (or similar) GUIs and code. **Everyone involved in cheating will get ZERO point.**

1. Design and implement a GUI program. The program must open at least 2 frames or dialogs, **JOptionPane not counted**. Use **javax.swing** package for containers and components. **JavaFX forms and components are not allowed**.
    - The class that opens the first frame/dialog must be named **MainApplication** & must be placed separately in file **MainApplication.java** (i.e. the whole program execution starts at main method of this class).

2. The following components must be included in any frame/dialog:
    - JTextField, JPasswordField, or JTextArea
    - JCheckBox or JRadioButton : at least 5 items
    - JComboBox or JList : at least 5 items
    - JButton : one frame/dialog must have a button that opens another frame/dialog
    - An extra component (child class of JComponent) not taught in class, e.g. JSpinner, JSlider, JTable
    - Names and IDs of everyone in your group, as any component

3. Implement at least 4 event handlers. The handlers are counted by method names, e.g.

    A button with MouseListener and mouseClicked(…) method
    A combo box with MouseListener and mouseClicked(…) method } Counted as 1 handler

    A button with MouseListener and mouseDragged(…) method
    A combo box with MouseListener and mouseClicked(…) method } Counted as 2 handlers

    In order to be counted, the handlers must do something substantially (not just printing/showing message). Out of the 4 handlers, one of them must be to handle **KeyEvent** and another one to handle **MouseEvent**. The other 2 handlers can be to handle any event.

4. Make a demonstration clip (< 5 minutes) in .mp4, .m4v, .mov, .mpg or .mpeg, .wmv format. Run your program and explain what it is about. If it is a game, explain how to play it. After watching the clip, user should be able to use your program properly.

5. Package and folder structure must be correct.
    5.1 Your source files (.java) must be in folder Project3_XXX where XXX = ID of the group representative, assuming that this folder is under Maven's "src/main/java" structure. The first lines of all source files must be comments containing English names & IDs of all members.

    5.2 Input files (images, sounds, etc.) must be read from Project3_XXX. If they are placed in subfolders e.g. resources, all subfolders must be inside Project3_XXX. Don't use absolute path that is valid only on your PC. Avoid too big image & sound files. They won't give you any extra point. Instead, you may lose points if they don't load & run properly on my PC.

    5.3 Add readme.txt containing English names & IDs of all members in Project3_XXX.

Grading (20 points)

Points	Description
4	Project presentation & QA. Points are given based on: • The smoothness of your demo & presentation (without technical problems or crashes) • Clear explanation of your project • Questions & answers • Others e.g. punctuality, group participation, etc.
1	Demonstration clips**.** **YouTube link is not accepted –** some clips will be kept as examples for next year classes.
6	Minimum requirements ➔ 2 frames + 6 components + 4 event handlers
1.5	Aesthetic and user interface, with adequate instructions on how to use your program. • The program must run on **Java 21.** Frame size must **not exceed 1366 x 768 resolution.** • Avoid very large & unresizable frames/dialogs or very large images. • Avoid placing components near the border of frames/dialogs – they may be cropped out if the frames/dialogs don't fit my screen.
1.5	Completeness. Everything functions as it should, with correct output/calculation and proper exception handling
6 (4 * D)	• 4 = points for programming in proper OOP + multithreading + GUI + event handling styles
	• D = difficulty multiplier
	0.75 i.e. max score = 4 * 0.75 = 3 GUI with easy tasks (e.g. get input, show output); no thread or any interesting elements (e.g. sounds, images, tables, timers); or threads/interesting elements with easy & non-meaningful tasks just for the sake of having it.
	1 i.e. max score = 4 * 1 = 4 GUI with meaningful tasks, but most are simple or adapted from course materials/exercises; threads or interesting elements (e.g. sounds, images, tables, timers) with simple tasks.
	1.25 i.e. max score = 4 * 1.25 = 5 GUI with creative tasks of your own; threads or interesting elements that require much programming efforts (e.g. graphics, timers).
	1.5 i.e. max score = 4 * 1.5 = 6 GUI with creative tasks of your own that require some complicated calculation or handling; threads or interesting elements that require much programming efforts (e.g. graphics, timers).

Presentation – 20 minutes per group

1. Start the presentation by running your demo clip.
2. Open your NetBeans (or other editors). Show your code and explain the following points:
   - Extra component ➔ its purposes, how it works.
   - Each event handling ➔ event & event handling method, listener, component (to which the listener is attached), how it works.
3. Some QA to check your understanding & whether you did the project by yourselves.
4. Any interesting point in your program to show your extra efforts.
5. No need for slides. No need for everyone to present the project, but at least show some participation.

Submission

1. Put source files (*.java) + related files (images, sounds, etc.) + clip file + readme.txt in folder **Project3_XXX**. After being zipped, the size of **Project3_XXX.zip must not exceed 300 MB.**
   - External libraries/packages (such as com.google, org.alicebot, etc.) are strongly discouraged. I won't have time to set up NetBeans and won't bother trying. If your program doesn't compile or doesn't run properly on my PC due to the lack of such libraries, you may lose some points.
2. Group representative submits Project3_XXX.zip to Google classroom. The others submit only readme.txt.
   - For big submission file, you may give a link to your Google drive. Make sure that you set permission for rangsipan.mar@gmail.com to access your file.
   - Before submitting the project, every member in your group should try downloading & running the project to make sure that it works on any computer.