

CS108 Assignment 7: Advanced Graphical User Interface (GUI) – Survey Project

DUE DATE: TUESDAY 23 NOVEMBER 2004 IN CLASS

Learning Objective

Learn how to build a fairly complicated graphical user interface for an application. Specifically, the GUI will involve multiple dialogs to prompt the user and display results. The GUI should incorporate the use of buttons, textboxes, check boxes, radio buttons, interactive dialogs, and menus. The back-end implementation code will be provided to you.

Problem Statement

You will develop a simple survey-taking application. Refer to the appendix on page 2 for details about how surveys work. Your survey will be on a topic of your choice, and will contain at least 4 separate dialogs (some custom, some can be JOptionPane, but you need to figure out which is which), to ask the respondent at least 4 questions, one question of each type identified in the appendix.

The GUI you will develop should be built using the NetBeans graphical-user-interface drawing palette, and will take the structure of a main window with some menus on it, and several additional dialogs.

- The menu and main window will be responsible for starting and stopping the SurveyProcessor object (to be provided), as well launching the survey dialogs for each new respondent.
- The dialogs will have one question (or more if they are related), and have a *Next* button to exit the dialog and bring up the next question.
-

Don't forget to:

- Display the application name in the title bar.
- Provides intuitively named fields and buttons.
- Make the design as clean and intuitive as possible. You *can* score extra points for making a nice looking GUI.

Design hints

- Start by writing the survey questions. You need to have an idea what it should do before you code it up.
- After the survey question, decide what kind of GUI dialog you need for each question. Draw the dialogs in NetBeans. Don't worry about code yet, just work on the interface (look/feel).
- Use the properties dialog to give each GUI widget a meaningful name (e.g. call a button by it's purpose, "exitButton" to exit the application, not "jButton1").
- Now add the navigation between the dialogs, using a Next button to call the dialog's `.setVisible( boolean )` method as is appropriate. Start with just one dialog, and try building and running it. You'll quickly get the hand of it.
- Finally, connect up the implementation class to record answers to the questions.

Deliverables

- The java code file(s) (*.java).
- An executive summary of no more than 1 page, which will serve to introduce the survey application and identify any important lessons learned from this project.

QUESTIONS: EMAIL ME: azs@bu.edu

Appendix: Basic Survey Design

Surveys are questionnaires used for gathering data about a group or groups of people, for example for marketing studies about product preferences. Each respondent is expected to answer each question once, and the set of responses from each individual responding constitutes a “case.” The results of a survey are comprised of many cases.

A survey can contain an arbitrary number of questions of various types. There are two main categories of survey questions:

1. Open-ended questions allow the respondent to provide any answer of their choosing.

Example: What did you like most about “Lord of the Rings?”

2. Closed-ended questions allow the respondent to select among a fixed list of alternative answers to the question. Several types of closed-ended questions exist:

- a. Questions resulting in a **dichotomous** outcome (e.g. true/false or yes/no questions).

Example: Did you see the first “Lord of the Rings” movie?

Which sports do you watch on TV. Check all that apply: [Football, Baseball, Hockey, Soccer, Basketball]

- b. Questions allowing the respondent to select a single answer among a group of choices (i.e., **multiple choice questions**).

Example: Indicate your class rank in school: [FR, SO, JR, SR, SR++]

- c. Questions allowing the respondent to indicate the strength of their agreement with a clear statement, referred to as a **Likert scale**.

Example: "I found the software easy to use..."

- Strongly disagree
- Somewhat disagree
- Undecided
- Somewhat agree
- Strongly agree