

SE 310 Software Architecture I

Homework Assignment 2 - Part A

NOTE: this is a very demanding project and it will require a lot of effort and a substantial amount of time. For part A of this project, spending a just an hour or two will not be sufficient and will earn you a very low grade.

Also, note, this assignment has substantially changed from previous years. Therefore, be careful to make sure follow this year's instructions.

Instructions

Your task is to develop a generic, console-based survey taking system. The system is to be written in **JAVA**. The following system requirements must be met.

Develop a system that allows a survey to be:

1. Created
2. Modified
3. Stored
4. Loaded
5. Taken
6. Displayed

The survey should be stored in a file system. We require you to use serialization. Each survey created by this system should be stored in an individual file. Each individual's response to a survey should be stored in an individual file. Therefore, if you have a survey created and three sets of responses for that survey, you should have four files.

A survey can be composed of any combination of the following type of questions:

1. True/False
2. Multiple Choice
3. Short Answer
4. Essay
5. Matching
6. Valid Date

Each question can accept a single answer, or if appropriate, multiple answers. Obviously, a True/False question would not accept multiple answers.

This assignment will be given in four parts; two UML assignments and two programming assignments.

For Part A, turn in a PDF or very high-resolution image (acceptable formats: jpg/png) of your UML generated from a computer program. Please DO NOT hand write your UML, you will be

given no credit if you do. It is ok to use a program that adds a watermark. StarUML is a good choice, cross platform, and is free for education, but you are free to use any software of your choice.

TIPS:

Many students get themselves into trouble not because they do not code well, but because of their approach. This is a large, yet not complex, assignment for students at your level. With a good approach, you will save a lot of time. Therefore, please follow these hints to improve your chances of successfully completing the project on time.

1. Do not leave the file saving/loading features as the last task. If you do so, chances are that when a bug arises you will have a hard time figuring out if this was caused by the main code or the file saving/loading code.
2. Learn how to use serialization.
3. Practice Serialization by creating a list of objects; serializing it to a file, and then loading it back into memory. If you understand the syntax and subtleties of serialization before you start you will be far better off.
4. Do this before you work on your UML so you understand what needs to be in your UML for file saving and loading.

Past Questions/Answers

*** READ this section before asking your own questions!

1. When users are filling out a survey, are they allowed to give a null answer to any question regardless of the type of question? On the other hand, should the admin be given the option to make certain questions not require an answer? Or should the overall design allow for such a change to take place in the future? I guess I am asking whether each question has a default answer or not.

No null answers are permitted and there is no such thing as an admin. There are no official users in the system.

2. When the user is filling out a survey is the user allowed to skip questions and return to them later? That is, should the user be allowed to return to previously answered questions before the final submission and modify any of their responses?

You are not allowed to skip questions.

3. Should the user be allowed to quit the survey before completion? If the user does decide to quit, should their previous responses be saved so that they can resume the survey later by selecting from a bank of saved but incomplete surveys?

You should not allow a survey to be quit in the middle.

4. Is the main class, the driver of the application, shown in the UML? If so, is it specified in a

special way?

Yes, the driver/main class must be shown in the UML. There is no special way or notation. The main class is shown in the same way as all the other classes.

5. If one were to implement a Java interface, and have classes implement the interface, how is this represented in UML?

This is shown in the review of JAVA, posted in Bb Learn.

6. What type of questions can have multiple answers?

Multiple choice, short answer, essays, and valid date questions can all have multiple answers.

7. Who decides the number of questions that should be there in the survey?

The user creating the survey.

8. How many T/F or MC or other types of questions should be present in the survey?

The user creating the survey decides how many questions are added to the survey.

9. Are we are supposed to make up the questions on our own?

No, the program allows the user creating the survey to make up any questions they wish.

10. And exactly what should we display as an output?

The program needs to be menu-based allowing you to access all the functionality discussed. You turn in a functional program.

Late Policy

- Assignments submitted 1 hour to 1 week late will receive a 15% penalty.
- Assignments submitted 1 to 2 weeks late will receive an additional 10% penalty.
- Assignments submitted more than 2 weeks late will be subject to an additional 5% penalty for each week.