

MURDOCH UNIVERSITY

ICT373 Software Architectures

Semester 2, 2017

Assignment 2 (20%)

Due Date: 5:00pm Saturday 27/05/2017

All Students: Submit the Assignment via LMS by the due date.

Late penalty: 10% per day penalty for delayed submissions unless prior extension of deadline is obtained from the unit coordinator.

You should keep a copy of your work. Your submission must include a completed assignment cover sheet. An electronic copy of the assignment cover sheet is available at the unit LMS site.

This assignment consists of one question only.

Question [20 marks]

Design and implement (in Java) a basic graphical user interface (GUI) program for recording information about a family tree.

You can assume that a family member has some immediate relatives: i.e., a mother and father (maybe unknown), zero or one spouses, zero or more children and zero or more grandchildren. A family member also has a first name, a surname at birth, a surname after marriage, gender, an address and a paragraph of text describing what is known about their life. Obviously, this is a simplified version of reality.

An address has a street number, street name, suburb and postcode.

A family tree contains this information for a whole bunch of related people.

The program can be started in editing mode or viewing mode.

In viewing mode, the GUI shows details for one person at a time. A family tree will have specified starting person. So if the person has parents and children and grandchildren, the family tree should show the parents (you do not need to show the grandparents or beyond; that means only one level up in the tree) and all descendants on a single page (in a sort of hierarchy so that the people at the same level are together; do not worry about the real look of a tree, a text box is fine).

From the GUI the user will be able to change the displayed person to (display) one of the immediate relatives of the current person.

In editing mode, the user can choose to start a new empty family tree or edit an existing one. A new family member can be added as an immediate relative of an existing person or the names, address, or text for an existing person can be changed. It is not necessary for this basic software to allow the user to be able to link an already existing person in a new role, eg as the mother of another already existing person.

The software should allow a user to conveniently enter their own family tree and have it stored for subsequent editing and viewing.

Set up a couple of demonstration family trees which you submit and give clear instructions in the User Guide for use of those family trees.

Note that your program should compile and run under Java SE 8 and NetBeans IDE.

Please note, you need to use the major items such as serialization, swing components, JFileChooser, and Exception handling (that we have covered in the unit),

Required Documentation for the Question

Please remember to submit the Java source code and executable version of your program (i.e. the whole NetBeans project). The final version of the program should compile and run under Java SE 8 (JDK 8). Internal students should test compilation and running on the University lab machines.

For internal documentation (ie in the source code) we require:

- a beginning comment clearly stating title, author, date, file name, purpose and any assumptions or conditions on the form of input and expected output;
- javadoc and other comments giving useful low-level documentation and describing each component;
- well-formatted readable code with meaningful identifier names and blank lines between components (like methods and classes).

Required External Documentation (One file)

- **Coversheet:** A signed Assignment cover sheet.
- **Title:** a paragraph clearly stating title, author, date, file names, and one-line statement of purpose.
- **Requirements/Specification:** a paragraph giving a more detailed account of what the program is supposed to do. State any assumptions or conditions on the form of input and expected output.
- **User Guide:** instructions on how to compile, run and use the program.
- **Structure/Design:** Outline the design of your program: describe why you chose one approach rather than other possible approaches. Give a written description. Describe how

the classes relate to each other. Describe each class: its purpose, list of methods. Use diagrams, especially UML and STDs. Include a complete description of the design. (Please also include pseudo codes for three of your interesting or complicated algorithms.)

- **Limitations:** Describe program shortfalls (if any), e.g., the features asked for but not implemented, the situations it cannot handle etc.
- **Testing:** describe your testing strategy (the more systematic, the better) and any errors noticed. Provide a copy of all your results of testing in a document saved in Word format. Note that a copy of the sample data and results from a test run of the program is required (copy from the program output window or the GUI windows and paste to the Word file). Your submitted test results should demonstrate a thorough testing of the program.

Source code: Save all your java source code files in appropriate names and directory structure. You must include them in the submission.

Note that all of the above external documentation must be included in a file saved in Word format.

The external documentation together with all relevant files must be compressed in a .zip file before submitting. Make sure that all necessary files are submitted so that the program can be viewed, compiled and run successfully. Note that the whole NetBeans project folder can be zipped.

Warning: Do not use visual tools e.g., bean builders or any type of code generators for your GUI design. You need to write code. [Heavy penalties apply]

Remember to complete and sign the Assignment Cover Sheet and submit it with your work.