Bachelor theses

Write an application for the management of bachelor theses in the faculty, as follows:

1. The information about all the teachers is in a text file. Each teacher has **a unique name**. This file is manually created and it is read when the application starts.
2. Another file contains information about the students. Each student has an **id** (string)**, a name**, **an email**, **the year** he/she has to finalize his/her bachelor thesis, a **bachelor thesis title** (for students who have not yet chosen a title, the default value will be "No title") and **a coordinator** - the name of the coordinating teacher (for students with no coordinator the default value will be "").
3. When the application is launched, a new window is created for each teacher, having as title the teacher's name. The window will show a list with all the students coordinated by the teacher, starting with students who have to defend their thesis in the current year and then all the others, sorted descending by year **(1.5p)**.
4. Any teacher can search for a student, by name or id (search by substring). *Another list* will show all matching students, with all their data. This list is updated as the teacher types the search string. **(1p)** The teacher can select a student from the resulted list and add himself/herself as a coordinator to the selected student only if the student does not already have another coordinator. The student will be added to the teacher's list of coordinated students **(1p).**
5. A teacher can edit their coordinated students' data. A new window will open, showing all student information. Only two attributes can be edited: the email and the thesis title. After the modifications are made, the teacher can save or cancel. If the modifications are saved, the list of students coordinated by the teacher will be updated accordingly **(1.5p).**
6. Another window will show a visual statistic (pie chart, histogram, proportional circles/rectangles) about the number of students each teacher coordinates, including the students without a coordinator. **(1p)**
7. When a modification is made by any teacher, all other teachers will automatically see the modifications (in both the coordinated students and the search students lists). The statistics window will also be updated. **(2p)**

Observations

1. **1p - default**
2. Specify and test the following functions (controller):
    a. Function which adds a coordinator to a student. **(0.5p)**
    b. Function which searches for a student by name or id **(0.5p)**
3. Use layered architecture. Otherwise, you will receive 50% of each functionality.
4. If you do not read data from files, you will receive 50% of file-related functionalities.

Non-functional requirements

1. Use STL to represent your data structures.
2. Use layouts for automatic resizing and repositioning of your widgets.