Software Modeling (CSS 507)

For each question below, you will draw diagrams for the GradCheck System that will be developed. We provide a brief description and a list of requirements for this software.

GradCheck system will aid students in tracking their course schedule, planning future courses, and ensuring that they meet graduation course requirements

Requirements:

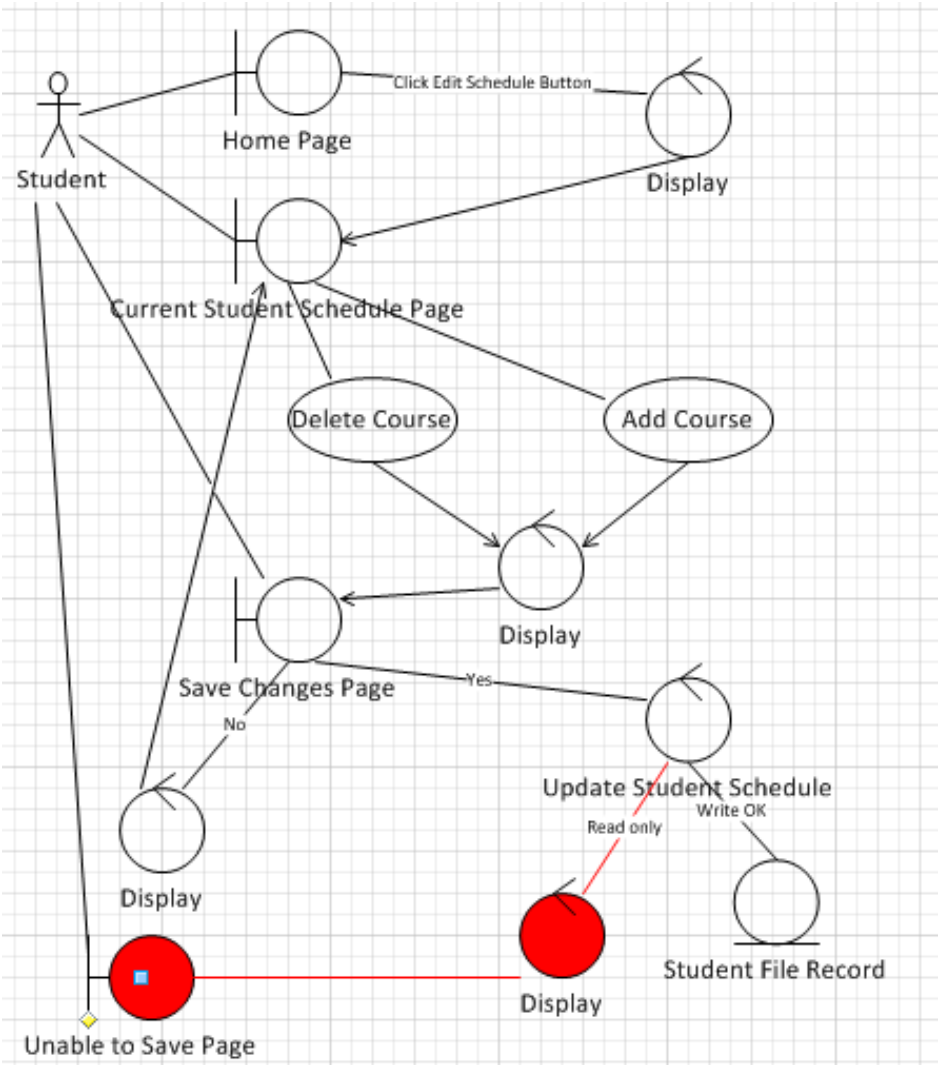
- The GradCheck System (or simply GradCheck) must be web based
- GradCheck shall be used by students and advisors
    - Students shall be able to create a new schedule, add, edit, or deleted courses. Students shall also be able to read the information in the Course Offering and Course Policies
    - Advisors shall be able to view student records as well as edit the information in the Course Offering and Course Policies
    - Students shall be able to enter their name, student ID, major, and course schedule
    - Students shall be able to select courses from the course offering list to add to their course schedule
    - GradCheck shall detect if there are conflicts in the schedule (e.g., different classes that occur at the same time) or policy violations (e.g., if there are pre-requisite courses that have not been satisfied)
    - Advisors shall be able to add, edit, and delete courses in the Course Offering list.
    - Advisors shall be able to specify a course's number, name, number of units, list of pre-requisite courses, quarter and year offered
    - Students shall be able to view required courses for graduation
    - Students shall be able to view the results of the graduation check (e.g., "You are on track", "In danger of academic probation")

1. Robustness diagrams are often used as a bridge between use cases (problem analysis) and sequence diagram (preliminary design). Draw a robustness diagram for the following use case for the GradCheck System. (Note: Please color-code your diagram to distinguish between the basic course and the alternate course.)

    Basic Course
    The student views the Grad Check Home Page and clicks on the Edit Schedule button. The system then displays the Current Student Schedule page. On this page, the student may then choose a course to delete a course or to add a course. After the student performs these tasks, the system asks the user to save the changes on the Save Changes page. If the student selects Yes, the system updates the student record file. Otherwise, the student record is not updated.

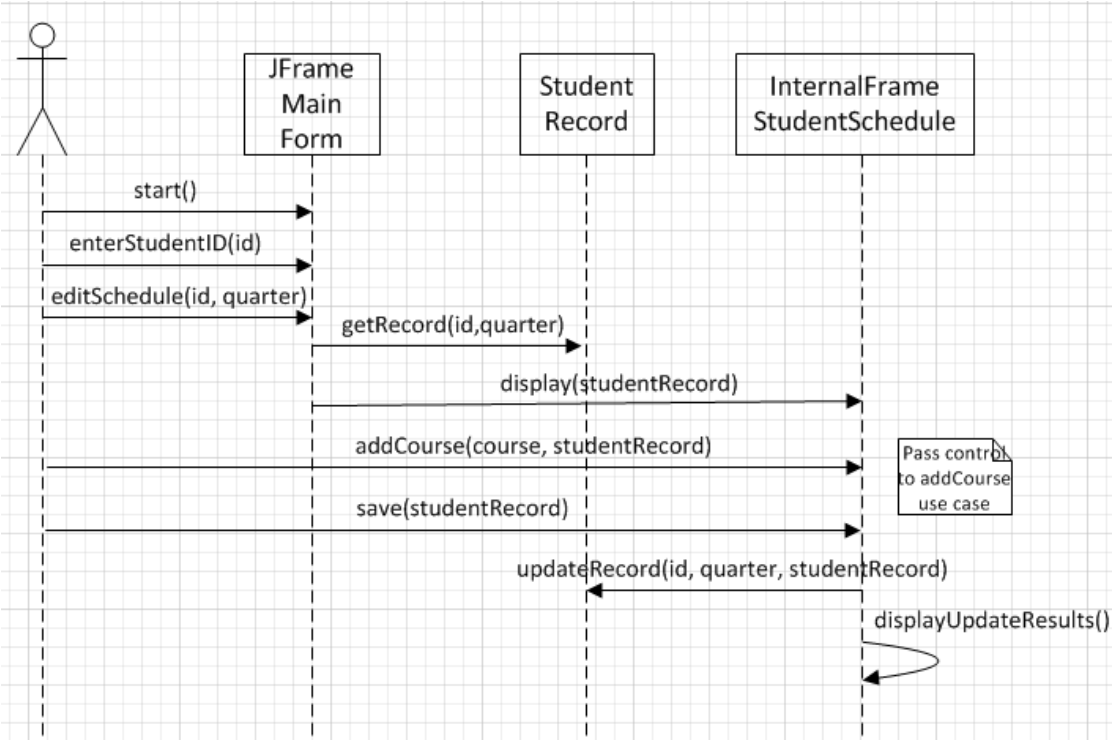
    Alternate Course
    The student record file is unavailable: display Unable to Save page to the Student.

Answer: Here is a sample robustness diagram.



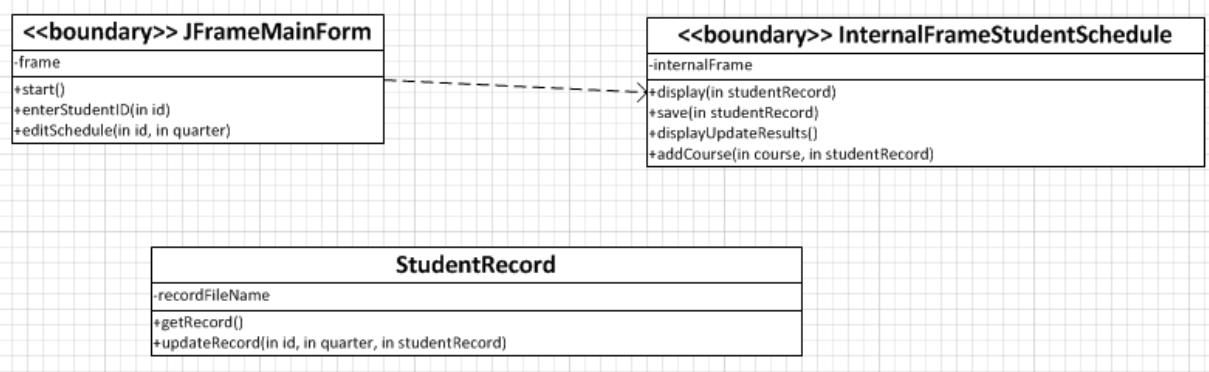
2. From your robustness diagram, draw a sequence diagram. You only need to show the basic course in this diagram.

Answer: Here is a sample sequence diagram.



3. From your sequence diagram, create a simple class diagram. Make sure to include attribute and methods for each class.

Answer: Here is a sample class diagram.



4. Select an architectural style for the GradCheck System. Why did you choose this style?

Answer: Client-Server is a good choice for the system. Logic and data are centralized in a server (or servers in the case of 3-tiered client-server). User interacts with pages served from the server via a browser. Other architectural styles may also work for this system, such as the Model-View Controller.