İTÜ
Computer Engineering Department
Assoc.Prof. Feza BUZLUCA
https://akademi.itu.edu.tr/en/buzluca/

20.03.2024

Object-Oriented Modeling and Design 2nd Assignment
Design with GRASP

Problem:
In this assignment, we will design a part of the **student administration system** that we started to analyze in the first assignment.

Assumptions:
- Do not consider the user interface design. How data is entered and presented is not important in this assignment.
- Assume that the students and officials have already logged in to the system. You do not need to realize the login operation.
- You do not need to consider system initialization. You may assume that the necessary initial operations have already been performed, and information about students and course plans of faculties reside in systems memory in proper data structures.

Consider the following features and operations:
- An authorized official in the registrar's office creates classes to be offered in the current semester and sets their attributes, such as the CRN, instructor, hours, classroom, quota, etc.
  For example, the official selects BLG468E from the faculty's course plan, creates the sections, and sets the attributes.
- Consider the course enrollment (registration) operation that consists of the following suboperations:
  - The student selects a CRN.
  - The system checks the registration rules (prerequisites, quota, time conflicts, etc.) and presents whether the selected class enrollment is accepted or not.
- Students can get information about a course they enrolled in.
  - The student selects the CRN.
  - The system presents the information about the related course:
    - Instructor
    - Classroom
    - Grades entered by the instructor
      The instructor enters the number of exams, assignments, etc. Students will receive grades for each exam or assignment over 100. In addition, each course will have a letter grade.
      How the instructor enters grades is out of the scope of this assignment.

- Information about the courses students enroll in is stored in a database system (for example, SQL server). This database system may be replaced with another system in the future (for example, Oracle). We are sure there will always be only one database in our system at a particular time.

To Do:

- Design the part of the student administration system described above using object-oriented design principles and **GRASP patterns**.

1. Draw your **design model** as a **UML class diagram**.
2. Draw the **UML sequence diagrams** showing how objects interact for the **course enrollment** (registration) operation (including suboperations).
   You may draw the diagrams by hand. In this case, please use a ruler and be neat.

SUBMISSION:

- Upload two files, i.e., class_diagram.pdf and sequence_diagram.pdf, to Ninova by **23.00** on **April 2, 2024, Tuesday**.
- **Late submitted assignments are not accepted.** Do not risk leaving your submission to the last few minutes.
- Do not send your solutions by e-mail. We will only accept files uploaded to the official Ninova e-learning system before the deadline.
- **Cheating** will not be tolerated. Any cheating is subject to the University disciplinary proceedings.
  It is allowed to discuss how to solve a problem with your classmates; however, **this assignment is not group homework. The actual solution should be an independent effort**.