GIT Department of Computer Engineering
CSE 222/505 - Spring 2015
Homework 01
Due date: February 25 2016 – 08:00 AM

SCENARIO:

Design and implement course automation system. The system has users like administrators, teachers, tutors and students. Administrators manage the system by adding, removing users and courses. Teachers who manage courses can add and remove users, tutors and documents to a course. The document can be in different formats like books, files, slides, urls and whiteboard descriptions. Teachers can give assignments. Assignments can be in different formats like quiz, homework or group project. An important part of the assignment is that every assignment has a deadline and a late deadline. A teacher can manage multiple course in a course term. Teachers and tutors can view older courses but cannot make any changes. Tutors only can view their course materials and assignments. A tutor can be a student in a different course. Students can register into system and can upload assignments. She/he can view grades and lecture notes.

OBJECTIVES:

- Preparing object oriented design for the problem
- Creating interfaces
- Applying polymorphism
- Applying method overriding
- Applying error handling
- Applying inheritance
- Applying code documentation
- Applying clean code standards
- Creating javadoc documentation

RESTRICTIONS:

- Use maven standard Project template
- Use only ArrayList data structure
- Can be only one main class in project
- Don’t use any other third part library

GENERAL RULES:

- For any question firstly use course news forum in moodle, and then the contact TA.
- Use maven project management tool. And upload maven project into moodle.
- Code the Project in Java programming language. Java must be 1.8.* or bigger version.
- Any java IDE can be used in coding process.
- Implement all interfaces class

- Add all javadoc documentations for classes, methods, variables …etc. All explanation must be meaningful and understandable.
- Implement clean code standarts in your code;
    - Classes, methods and variables names must be meaningful and related with the functionality.
    - Your functions and classes must be simple, general, reusable and focus on one topic.
    - Use standart java code name conventions.
- Register github student pack and create private project and upload your projects into github.
- Your appeals are considered over your github project process.
- You can submitting assignment one day late and will be evaluated over forty percent (%40).
- Create report which include;
    - Your name, surname, studentid
    - Detailed system requirements
    - The Project usecase diagrams (extra points)
    - Class diagrams
    - Problem solutions approach
    - Test cases
    - Running command and results

GRADING :

- No OOP design : -100
- No maven Project : -100
- No interface : -95
- No method overriding : -95
- No error handling : -95
- No inheritance : -95
- No polymorphism : -95
- No javadoc documentation : -95
- No clean code standard : -95
- No report : -90
- Disobey restrictions : -98
- Your solution is evaluated over 100 as your performance. Don't forget this is performance project.

CONTACT :

- Teacher Assistant : Necmeddin Çarkacı
- Information and Security Lab – 122 – ncarkaci@bilmuh.gyt.edu.tr