

## Homework 2

### CENG431 – Building Software Systems

In this homework you are expected to implement and design a “**Smart Home Simulation**” in Java. You should fulfill the concepts of:

- Mediator Design Pattern
- UML Class Diagrams
- UML Sequence Diagrams
- UML State Diagrams

In this application you are asked to implement a simulation program for a smart home automation system. The system should have a mediator object that manages communication between various components of the smart home, such as sensors, actuators, and a control panel. The components should not communicate with each other directly, but through the mediator object. The simulation program should have the following requirements:

1. The system should have three types of sensors: temperature sensor, light sensor, and motion sensor. Each sensor should periodically send a reading to the mediator object.
2. The system should have three types of actuators: thermostat, light bulb, and door lock. Each actuator should receive commands from the mediator object and perform the corresponding action.
3. The system should have at least one control panel that allows the user to set temperature, turn on/off lights, and lock/unlock doors. The control panel should send commands to the mediator object.
4. The mediator object should process the sensor readings and send commands to the appropriate actuators based on the user's commands and the sensor readings. For example, if the temperature reading is too low or high (20° C - 25° C), the mediator should send a command to the thermostat to lower the temperature.
5. The simulation program should run for a specified duration, during which the sensors should periodically send readings to the mediator and the control panel should send commands to the mediator. The program should output the current state of the system (temperature, light status, door status) to the console at regular intervals (each 1 second).

Your task is to implement the mediator design pattern to manage the communication between the sensors, actuators, and control panel, and to simulate the smart home automation system. You should write a Java program that implements the mediator, sensors, actuators, control panel, and simulation logic, and outputs the system state to the console for 20 seconds. You can randomly decide the values of the sensors and the commands for the actuators (thermostat should be automatic).

You are expected to draw a **UML Class diagram**, a **UML Sequence diagram**, and a **UML State diagram** for your program.

### Important Notes:

1. **Do NOT request inputs in your app.** Printing the results of the simulation will be enough.
2. To support **Turkish characters**, you may need to change your project's text file encoding to UTF8: Right click on your project (in package explorer) → Properties → Text file encoding → Other → UTF8 → Apply.
3. You are expected to write clean, readable, and tester-friendly code. Please try to maximize reusability and prevent from redundancy in your methods.

### Assignment Rules:

1. In this lecture's homework, there are no cheating allowed. If any cheating has been detected, they will be graded as 0 and there will be no further discussion on this.
2. You are expected to submit your homework in groups. Therefore, only one of you will be sufficient to submit your homework.
3. Make sure you export your homework as an **Eclipse project**. You can use other IDEs as well; however, you must test if it supported by Eclipse. If the project import is not accomplished in Eclipse, you will lose points.
4. Submit your homework through Cloud-LMS.
5. Your exported Java Project should have the following naming format with your assigned group ID (which will be announced on MS Teams) as the given below:

**G05\_CENG431\_HW1**

Also, the zip folder that your project in should have the same name.

**G05\_CENG431\_HW1.zip**

6. Please beware that if you do not follow the assignment rules for exporting and naming conventions, you will lose points.
7. Please be informed that your submissions may be anonymously used in software testing and maintenance research studies. Your names and student IDs will be replaced with non-identifying strings. If you do not want your submissions to be used in research studies, please inform the instructor (Dr. Tuglular) via e-mail.