



Hacettepe University

Department of Computer Engineering

Smart Home System

Assignment Report

Course Name:

BBM104 – Introduction to Programming Practicum

Project Assignment – 2

Mert ERGÜN - B2220356062

09.04.2023

Index

1- Introduction	3
2- Analysis of Solution	3
3- Encounters with Problems	4
4- Benefits of Smart Home System	5
5- Benefits of OOP	5
6- Four Pillars of OOP	5
7- UML Diagram	6

1- Introduction

In this assignment we are asked to design a simple working “Smart Devices Complex”.

In this complex, smart devices will be placed and this complex will be managed by using object oriented programming skills together with time control. The difficult parts of this assignment will be to control the properties of the smart objects while providing time control and to handle the inputs that the user may provide incorrectly. All inputs must be checked, erroneous inputs must be handled and the program must terminate without the JVM throwing an error.

2- Analysis of Solution

In the process of solving these problems, object-oriented programming techniques, especially Java, were used. In order to keep the common features of each smart device, an abstract class named Smart was created. All other classes (Plug, Lamp and Camera) were extended from this class. In this direction, the concepts of "Abstraction and Heritage" were used.

Some methods were overridden in the created classes. Some methods were added from scratch. Some attributes that all classes should have in common were taken from Smart class as "super.attribute". All other attributes were created specifically for the classes. At this point, the concept of Polymorphism, which is one of the foundations of object oriented programming, was used.

Since each object can be created with more than one method, more than one Constructor method was created for all classes. Since these methods accept different parameters, the concept of "Overloading" was used at this point.

To handle errors, if else blocks were used at some points and try catch blocks were used at some points. In this way, all kinds of erroneous input that the user could enter were checked and it was ensured that the user did not encounter a JVM error that the user could not understand, and at the same time the program continued to run without crashing.

Calendar object and its methods were used for date data. In this way, the date can be taken in the format entered by the user without the need for extra effort. Some of the reasons for using Calendar instead of Date are; the methods of the Calendar object have replaced the Date methods, the ability to perform operations on the Calendar object is higher and it provides more flexible possibilities.

3- Encounters with Problems

Basically, a study was done on how to control the erroneous inputs that can be entered while creating objects. As a result of this study, which focused more on try catch blocks, the code parts that will be required to handle incorrect inputs were organized with try catch and if else blocks.

One of the big problems encountered afterwards was time control. At this point, the errors received while manipulating the time with the data received from the user were examined in detail and an error-free code was written.

Along with time control, checking the switch times of the devices was also a very challenging part. We tried to find a data type that keeps the switch time data of all devices separately with the name of that device, then this problem was solved by creating a Tuple class.

In addition to switch times, changing the switch status of devices during time hops was also a challenging part. This was solved by using nested for loops and specific if else blocks.

Calculating the energy and storage usage of Plug and Camera devices was the most challenging part of the assignment. It was calculated how much power they consumed in the time interval from the last time the devices were changed until the control date. For this some properties of the Calendar object and cloning of this object were used. In this way, special times were assigned to the devices and operations were performed on these times.

4- Benefits of Smart Home System

Smart house system is a way of controlling home appliances and devices remotely through the internet or a local net. It can make daily tasks easier and more efficient by allowing users to adjust lighting, temperature, security, entertainment, and more with just a few commands. Users can also create schedules and routines for their smart devices to automate certain actions based on their preferences and needs.

Smart house system can also benefit homeowners by saving money on energy and utility bills, enhancing home security and safety, improving comfort and well-being, and reducing insurance costs. Some smart devices can also learn from the user's behavior and optimize their performance accordingly. Users can also customize their smart home according to their personal style and preferences.

5- Benefits of OOP

Speaking specifically about this assignment, there are many benefits of using OOP. Each smart device created is defined as an "object" and operations are performed on these objects. This saves the operations to be performed and the structure to be designed from being abstract. Classifying each object by using override methods also makes it easier to solve the problem.

6- Four Pillars of OOP

Inheritance

Inheritance concept briefly means that classes can have subclasses and super classes. Subclasses are created by extending super classes. This reduces the number of methods and variables that need to be created on each object.

Abstraction

Abstraction is modeling the relevant attributes and interactions of entities as classes to define an abstract representation of a system. It is a way to separate and more easily understand the structure and behavior of an object by hiding unnecessary details of its

internal structure. Abstraction means that you don't need to understand exactly what a method does when you call it. Abstraction abstracts away specific details to make your code reusable, simple, understandable and easily modifiable.

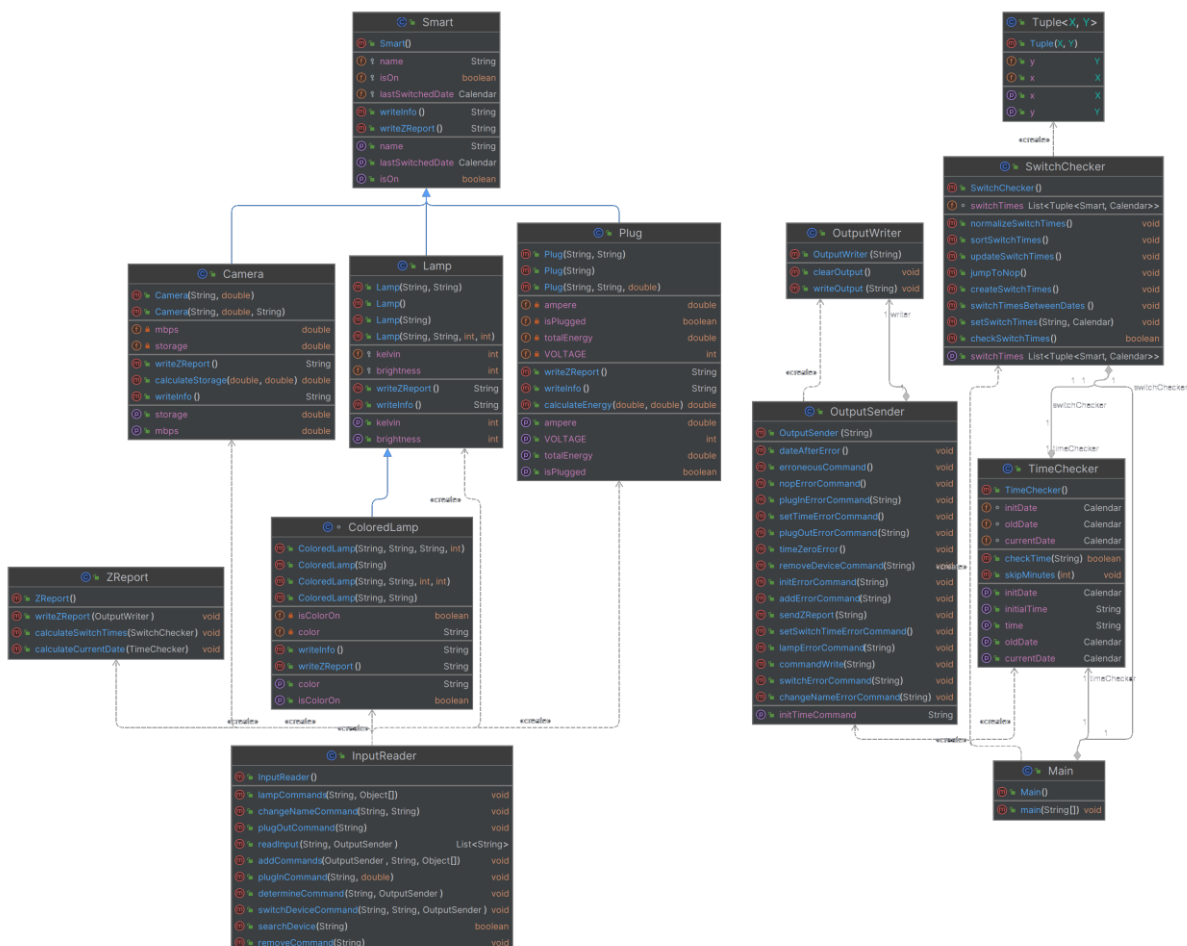
Encapsulation

Encapsulation is a way to protect and organize your data and functions by hiding them from others and only allowing access through some public functions. Encapsulation makes your code more secure and easier to use by keeping everything in one place. Encapsulation is done by using abstraction or encapsulation to hide data and functions.

Polymorphism

Polymorphism is a way to make different objects behave differently even if they have the same name or type. Polymorphism lets you use the same code for different kinds of objects and get different results. Polymorphism is done by using virtual methods that can be overridden by subclasses.

7- UML Diagram



As can be seen in the UML diagram, an abstract class named Smart was first created to create smart devices. Plug, Lamp and Camera classes extending this class were created. A ColoredLamp class was created that extends the Lamp class again.

OutputWriter class was created to print output and OutputSender class was created to print error and success messages.

Created TimeChecker and SwitchChecker classes for time and switch time controls, and a class called Tuple to use in SwitchChecker.

InputReader class was created to read inputs, call error messages according to the situation or take necessary actions.

ZReport class was created to print the Z report.

Main class was created to drive the whole program.