



Ankara Yıldırım Beyazıt University
Department of Computer Engineering

CENG 201 – Object Oriented Programming Course Project

G##: Smart Home Automation

Analysis Report

Bilgenur Erkan,
Buse Köroğlu,
Enes Eldeş,
Enes Mert Aydın,
İdris Aydın

Instructor: Muhammed Abdullah Bülbül

Teaching Assistant: Elif Şanlıalp, Yusuf Şevki Günaydın

Date: 01/11/2024

Table of Contents

1. Introduction	2
2. Requirements	2
2.1. Functional Requirements	2
2.2. Non-Functional Requirements	3
3. System Models	3
3.1. Scenarios	3
3.2. Use Cases	3
3.3. Object and Class Model	4-5
3.4. User Interfaces	6
4. Conclusion	6-7

1. Introduction

The Smart Home Automation System aims to provide users with a unified platform to manage smart devices within their homes. This platform will allow users to remotely control or monitor devices such as lights, security cameras, and thermostats. This report includes an analysis of the requirements (functional and non-functional) needed for the desired system, system models, use cases, and how to integrate it with object-oriented programming. The project's goal is to offer users ease of use by enabling remote control of smart devices at home while ensuring home security.

2. Requirements

2.1. Functional Requirements

1. Device Control
 - a. Turn On/Off
 - b. Set Timer
 - c. View Device Status
2. Timer Setting
3. Alert and Signal Transmission
4. Remote Access
 - a. Web Access
 - b. Mobile Application Support
5. Multi-Device Support

2.2. Non-Functional Requirements

1. Performance
2. Usability
3. Security
4. Flexibility/Scalability
5. Speed
6. Device Compatibility

3. System Models

3.1. Scenarios

Scenario 1: Security Camera Motion Detection and Notification Sending

A security camera detects motion in its designated area and sends a real-time notification to the user's mobile device. The user can view a live stream, ignore the alert, check recorded footage, or take action, like contacting security services. If the internet connection is down, recordings are stored locally and notifications are sent once the connection is restored.

Scenario 2: Thermostat Optimizing Home Temperature with Automatic Settings

The thermostat continuously monitors the home's temperature, adjusting heating or cooling to maintain user-defined preferences. If the temperature falls outside this range, it automatically corrects it. During a power outage, adjustments pause but resume once power is restored. The user can also change settings remotely for immediate adjustments.

Scenario 3: Turning Devices On/Off Through Voice Command or Manual Selection in the App

Users can control devices like the oven, TV and air conditioner via voice commands or the smart home app. When a command is given, the device responds, and the app sends a confirmation notification. If a device fails to respond or an unrecognized command is issued, the system prompts the user for clarification or suggests alternative options.

3.2. Use Cases

In the Smart Home Automation System, two types of users are identified in the Use Case Diagram (Figure 1): Admin and Standard User

1. Admin User: The Admin has full access to the system. They can log in, add or remove users, and manage devices in the system, including adding or removing devices as needed.
2. Standard User: Standard users have control over the devices in the smart home system but do not have permission to add or remove users or devices. Standard users can:
 - View Device Status: Check the current status of devices.
 - Control Devices: Turn devices on or off remotely, for example, turning lights on or off using the "Control Device" option while not at home.
 - Set Timer: Set timers for devices to automate their operation.

Additionally, users can monitor the home security system remotely:

- Monitor Security Cameras: Allows users to view security camera feeds remotely, enabling real-time monitoring of the home.
- Adjust Thermostat: Users can increase or decrease the home temperature remotely for optimal comfort and energy efficiency.

The system also includes alert features:

- Send Alerts: When a detector senses motion, detects a door opening, or identifies light changes, the system will automatically send an alert to the user, notifying them of the activity.

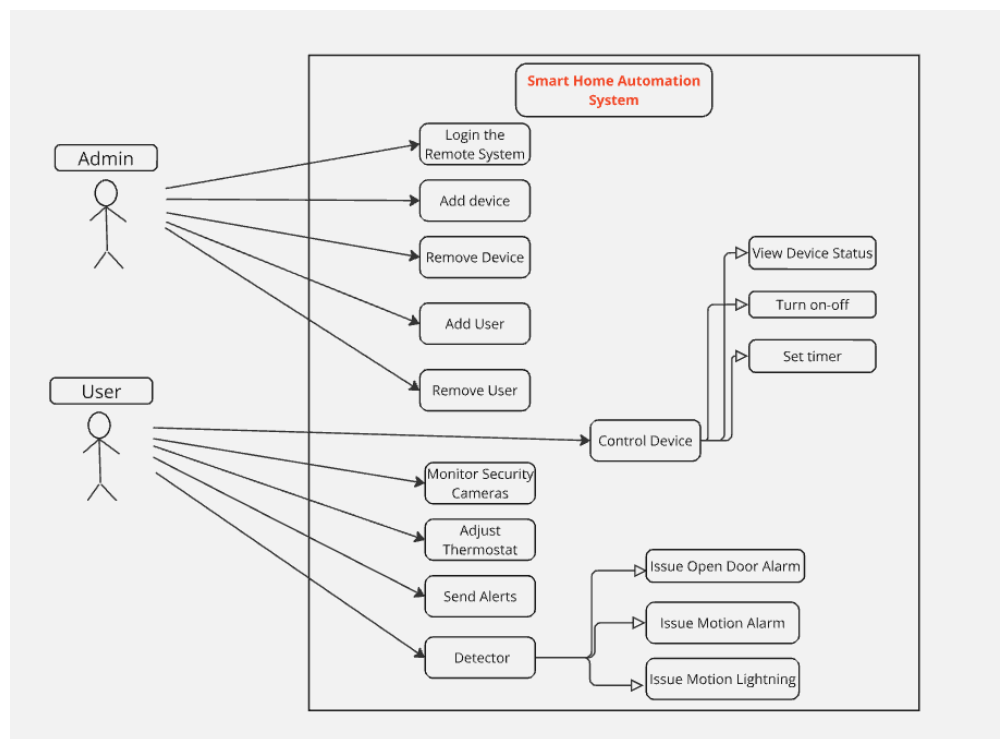


Figure 1: Use cases of the project

3.3. Object and Class Model

The project consists of three main classes: Home, Room, and Device. The top-level class, Home, HAS Rooms, and each Room HAS its own Devices. You can see this structure and the hierarchy of the Device class in Figure 2.

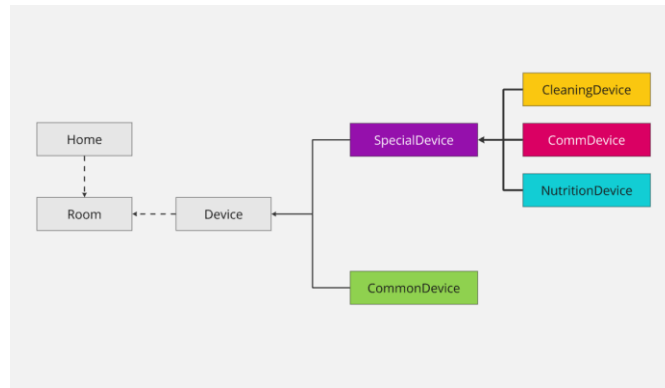


Figure 2: Abstract structure and Device hierarchy

Now, to better visualize the subclasses of the Room class, let's temporarily remove the Device subclasses from the class model. Each orange box in Figure 4 represents a Room subclass. The nested boxes below each subclass indicate the Devices associated with that Room. For the meaning of the boxes, please refer to Figure 3.

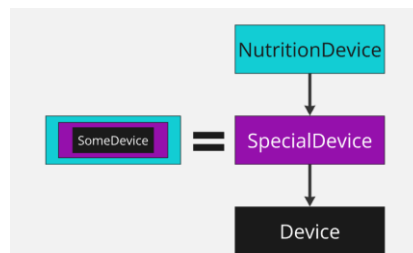


Figure 3: Meaning of the nested boxes

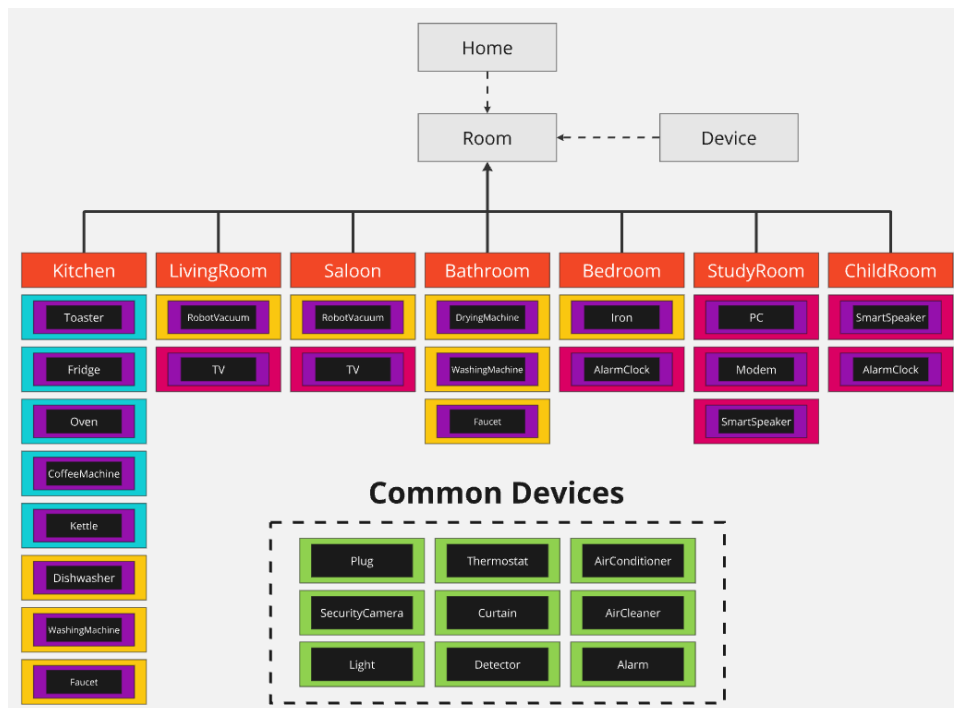


Figure 4: Room (and project) hierarchy

3.4. User Interfaces

The project's user interface is designed to be as simple and understandable as possible. It successfully meets all functional requirements of the project and supports the user experience.

- Accessing rooms and their devices from the home screen (Figure 5).
- Accessing admin screen to remove and add users of the smart home (Figure 6).
- Accessing control panel of the Devices (Figure 7-8).
- View electricity consumption.

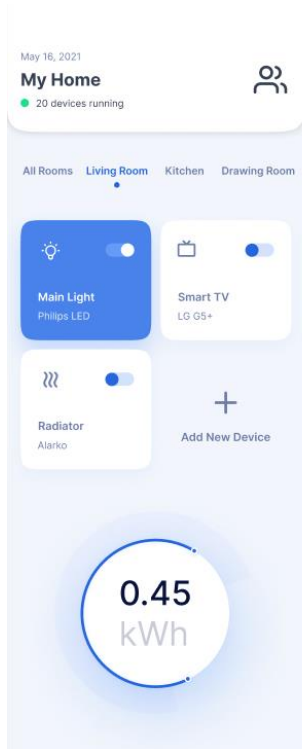


Figure 5: Home screen



Figure 6: Admin screen

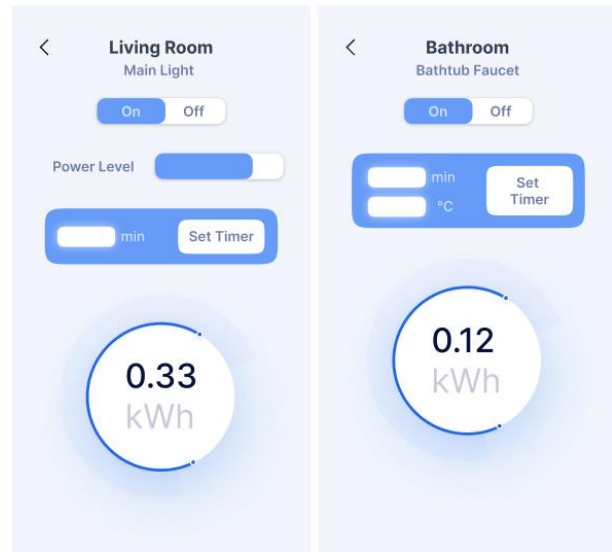


Figure 7-8: Device control screen

4. Conclusion

In summary, the Smart Home Automation System project provides a centralized, user-friendly platform for managing smart devices, with a focus on enhanced security, flexibility, and convenience. Leveraging object-oriented programming principles, the system is designed to be both scalable and adaptable to meet evolving user needs. The functional and nonfunctional requirements outlined in this report aim to improve household efficiency and security, establishing a solid foundation for future smart home technologies.

Each team member played a vital role in the project:

- Buse prepared the Introduction, clearly defining the project's purpose and scope, determined the functional and non-functional requirements of the project and constructed the use cases of the scenarios. (1-2.1-3.2)
- Bilge constructed the detailed scenarios of the project. Each scenario has been carefully structured to meet the project's objectives. (3.1)
- İdris developed the class structures and class diagram, ensuring a well-modeled system foundation. (3.3)
- Enes Aydın compiled the conclusion, effectively summarizing the project goals and anticipated impact. (4)
- Enes Eldeş prepared the user interface and coordinated task distribution, also gathered the shared information to finalize the report. (3.4)

The report was a solid team effort, with each member having their own role but staying in close communication to help each other out where needed. This collaboration allowed the team to complete the work smoothly and cover any gaps together.