Ankara Yıldırım Beyazit University
Department of Computer Engineering

CENG 201 – Object Oriented Programming
Course Project

G09: Smart Home Automation System

Project Stage 2-Design

Furkan Cem Çelik – 23050151020

Zekeriya Damcı – 22050111074

Tunahan Çoban – 22050111010

Murat Kaynar – 17050111048

Ahmed M. A. Albreem Ahmed M. A. Alb -19050141021

Instructor: Muhammed Abdullah Bülbül
Teaching Assistant: Elif Şanlıalp, Yusuf Şevki Günaydın
Date: 07/12/2024

Contents

1. Introduction

The **Smart Home Management System** is designed to empower users to monitor and control a variety of smart home devices, including lights, thermostats, and security cameras. The system provides the ability to perform remote operations and receive real-time alerts, thereby enhancing convenience, security, and energy efficiency for users.

This report includes a comprehensive design of the system, detailing both **Class-Responsibility-Collaboration (CRC) cards** and **UML class diagrams** to illustrate the responsibilities, relationships, and interactions among the system's components. The CRC cards document the primary functionalities and collaborations for each class and interface in the system, while the UML class diagrams provide a visual representation of the system's structure, including its attributes, methods, and relationships.

The design focuses on scalability, modularity, and usability, ensuring that the system can seamlessly integrate with new devices and features in the future. Through this documentation, the system architecture is thoroughly defined, facilitating further development and implementation.

2. CRC Cards

Class: Light

- **Responsibilities**:
  - Turn the light on.
  - Turn the light off.
  - Adjust the brightness level.
- **Collaborators**:
  - Issues commands to the Light class.
  - Implements this interface for basic device operations.

Class: Thermostat

- **Responsibilities**:
  - Set the desired temperature.
  - Retrieve the current temperature.
- **Collaborators**:
  - Manages temperature changes.
  - Displays and retrieves temperature settings from the user.

Class: SecurityCamera

- **Responsibilities**:
  - Detect movement in the camera's range.
  - Notify users of detected motion.
- **Collaborators**:
  - Sends alerts to users.
  - Manages the camera's operations.

Class: NotificationSystem

- **Responsibilities**:
  - notifications to the user.
- **Collaborators**:
  - **SecurityCamera**: Triggers notifications based on motion detection.
  - **Notification**: Implements the notification interface.

Class: ControllerUnit

- **Responsibilities**:
  - Coordinate actions across devices.
  - Interpret and execute user commands.
- **Collaborators**:
  - **SmartDevice**: Manages devices like Light, Thermostat, and SecurityCamera.
  - **UserInterface**: Receives commands from the user.

Class: UserInterface

- **Responsibilities**:
  - Show options to the user.
  - Collect input from the user.
- **Collaborators**:
  - **ControllerUnit**: Sends user commands for processing.
  - **NotificationSystem**: Displays notifications to the user.

Interface: SmartDevice

- **Responsibilities**:
  - Define how to turn the device on.
  - Define how to turn the device off.
  - Provide the device's current status.
- **Collaborators**:

    - **ControllerUnit**: Manages and interacts with devices implementing this interface.

Interface: Notification

- **Responsibilities**:
    - Define how notifications are sent.
- **Collaborators**:
    - **NotificationSystem**: Implements this interface for sending alerts.

Class: CupboardSystem

- **Responsibilities**:
    - Check the stock of items in the cupboard.
    - Alert the user if stock is low.
- **Collaborators**:
    - **NotificationSystem**: Sends low-stock alerts.
    - **ControllerUnit**: Manages stock checking actions.

Class: Window

- **Responsibilities**:
    - Open the window.
    - Close the window.
    - Get the current status (open/closed).
- **Collaborators**:
    - **ControllerUnit**: Sends commands for opening/closing.
    - **SmartHomeSystem**: Integrates window operations.

Class: PlantSystem

- **Responsibilities**:
    - Water the plants.
    - Check soil moisture levels.
- **Collaborators**:
    - **ControllerUnit**: Schedules watering tasks.
    - **ISchedulable**: Implements scheduling for watering

Class: User

- **Responsibilities**:
    - Configure preferences for smart home devices.
    - Receive alerts from the system.
- **Collaborators**:
    - **NotificationSystem**: Sends updates to the user.
    - **ControllerUnit**: Handles user configurations.

Class: SmartHomeSystem

- **Responsibilities**:
  - Add devices to the system.
  - Centralize all notifications.
  - Execute scheduled tasks.
- **Collaborators**:
  - **ControllerUnit**: Manages devices.
  - **NotificationSystem**: Centralized notification hub.

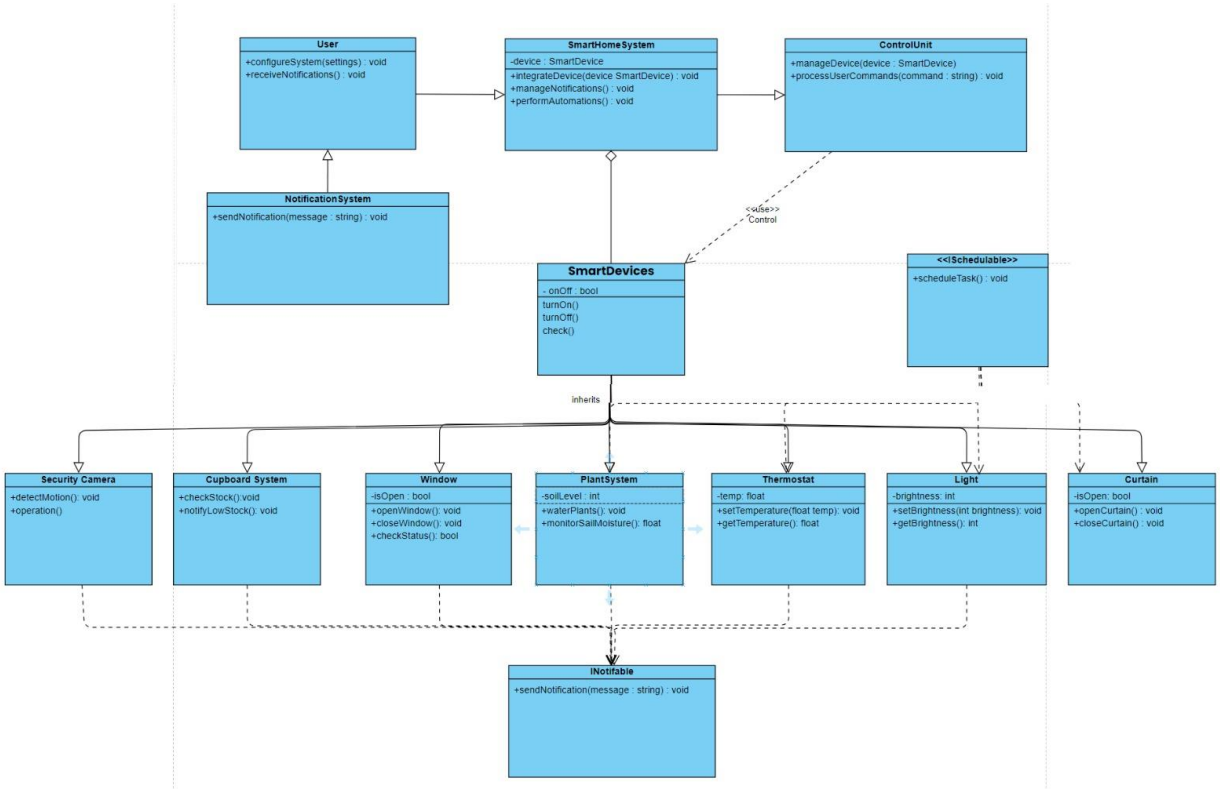
Interface: ISchedulable

- **Responsibilities**:
  - Schedule an actionable task.
- **Collaborators**:
  - **ControllerUnit**: Ensures tasks are scheduled and executed.

Interface: IActionable

- **Responsibilities**:
  - Define specific actions for smart devices.
- **Collaborators**:
  - **SmartDevice**: Implements device-specific actions.

3. Class Diagrams



4. Conclusion

The **Smart Home Management System** design provides a robust framework for managing and interacting with various smart devices, ensuring convenience, security, and efficiency for users. By incorporating lights, thermostats, security cameras, and other devices into a unified system, this design empowers users to perform remote operations and receive real-time alerts.

This report presented a detailed breakdown of the system's design:

- **Class-Responsibility-Collaboration (CRC) Cards** were developed for each class and interface to define their roles, attributes, and interactions.
- **UML Class Diagrams** visually depicted the structure of the system, illustrating relationships, attributes, and methods for all components.

The design was crafted with scalability, modularity, and usability in mind, allowing for future integration of new devices and features. Each group member contributed equally, ensuring thorough analysis and accurate documentation of the system's architecture.

As a result, the Smart Home Management System is well-prepared for implementation, providing a solid foundation for enhancing user experience and expanding smart home capabilities.