 ***Shaheed Zulfikar Ali Bhutto Institute of Science & Technology***

***COMPUTER SCIENCE DEPARTMENT***

***Object Oriented Programming***

***PROJECT PROPOSAL # 02***

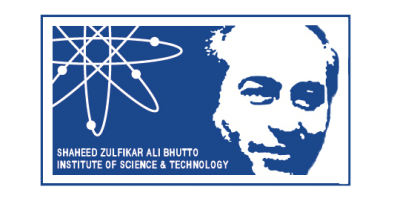
***“Smart Home Simulator”***

***SUBMITTED TO:***

***SAMRA MEHBOOB***

***SUBMITTED BY:***

* ***HAYYA SAYYAD ABBASI (2312398)***
* ***FIZZA KHAN (2312415)***
* ***SIRAJ KHAN (2312411)***

 ***Shaheed Zulfikar Ali Bhutto Institute of Science & Technology***

***TABLE OF CONTENT:***

1. *INTRODUCTION*
2. *PROBLEM STATEMENT*
3. *PROPOSED SYSTEM*
4. *OOP DESIGN*
5. *ERD/CLASS DIAGRAM*
6. *TEST CASES*

***1. Introduction:***

*The Smart Home Simulator project aims to emulate a modern smart home system, demonstrating the interconnectedness of home devices and how they can be controlled and automated. As technology advances, smart home devices such as thermostats, lights, security cameras, and appliances have become increasingly prevalent. This simulator will enable users to interact with a virtual smart home, controlling devices through a menu-driven interface while simulating real-life scenarios such as energy optimization and security alerts.*

***2. Problem Statement***

*Managing a smart home involves coordinating multiple devices efficiently. Real-world issues include:*

* *Difficulty in managing and automating multiple devices simultaneously.*
* *Lack of centralized systems to monitor device status and energy usage.*
* *Inability to simulate smart home operations for testing before real implementation.*

*The absence of a simulator to experiment with smart home configurations can make adoption challenging and costly. This project proposes a solution to emulate a smart home, allowing users to test automation, control devices, and observe interactions before real-world implementation.*

***3. Proposed System***

*The proposed Smart Home Simulator will include the following key features:*

1. ***Device Control***

* *Manage devices like lights, thermostats, security cameras, and smart locks.*
* *Switch devices on/off and adjust settings such as temperature or brightness.*

1. ***Automation and Scheduling***

* *Automate device operations based on time or sensor triggers.*
* *Create routines (e.g., "Good Night" mode turns off lights and locks doors).*

1. ***Energy Optimization***

* *Monitor energy usage for all devices.*
* *Recommend energy-saving configurations.*

1. ***Interactive Menu***

* *A user-friendly interface to interact with the system.*
* *View current status, add/remove devices, and simulate events.*

***4. OOP Design***

*The Smart Home Simulator will employ Object-Oriented Programming principles such as inheritance, composition, and friend functions/classes to create a modular and extensible design.*

***Key Classes and Functions***

* ***Class: Device (Base Class)***

***Attributes:***

* ***id****: Unique identifier for each device.*
* ***status****: On/Off status.*
* ***energyUsage****: Energy consumed by the device.*

***Functions:***

* ***turnOn():*** *Turn the device on.*
* ***turnOff():*** *Turn the device off.*
* ***getStatus():*** *Return the current status of the device.*
* ***Class: Light (Derived from Device)***

***Attributes:***

* ***brightness****: Current brightness level.*

***Functions:***

* ***adjustBrightness(int level)****: Set brightness to a specific level.*
* ***scheduleTimer(int time****): Turn the light off after a set time.*
* ***getDetails():*** *Display light attributes.*
* ***Class: Thermostat (Derived from Device)***

***Attributes:***

* ***temperature:*** *Current room temperature.*

***Functions:***

* ***setTemperature(int temp):*** *Adjust the thermostat to a specified temperature.*
* ***getTemperature():*** *Return the current temperature.*
* ***ecoMode():*** *Activate an energy-saving mode.*
* ***Class: SmartHome (Composition)***

***Attributes:***

* ***devices[]:*** *Array of Device objects.*
* ***energyConsumption:*** *Total energy usage.*

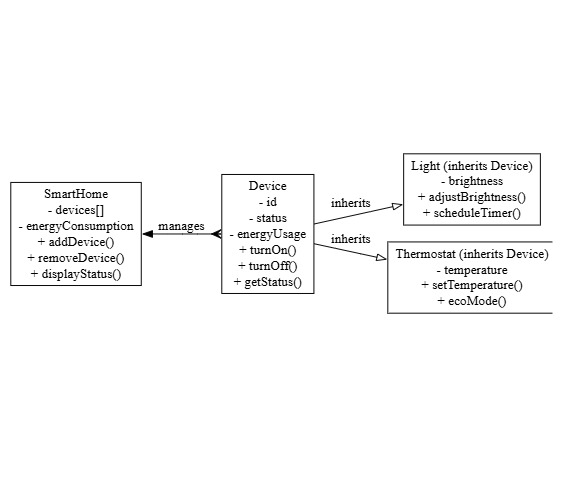
***Functions:***

* ***addDevice(Device d):*** *Add a new device to the system.*
* ***removeDevice(int id):*** *Remove a device by its ID.*
* ***displayStatus():*** *Show the status of all devices.*

***Special OOP Concepts Used***

* ***Inheritance:*** *The Light and Thermostat classes inherit from the Device base class.*
* ***Composition:*** *The SmartHome class integrates multiple objects, managing them collectively.*
* *Device* ***Friend Functions:*** *A friend function in SmartHome directly accesses private attributes of Device for consolidated reporting.*

1. ***ERD/Class Diagram:***



1. ***Test Cases:***

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | | ***Test Case ID*** | ***Description*** | ***Expected Outcome*** | ***Status*** | | *001* | *Initialize the Device class* | *Device is initialized with default values.* | *Pass* | | *002* | *Turn on a Device* | *Device status changes to "On".* | *Pass* | | *003* | *Turn off a Device* | *Device status changes to "Off".* | *Pass* | | *004* | *Adjust brightness of a Light* | *Brightness is set to the specified level.* | *Pass* | | *005* | *Schedule timer for a Light* | *Light turns off after the specified time.* | *Pass* | | *006* | *Set temperature on a Thermostat* | *Temperature is updated correctly.* | *Pass* | | *007* | *Activate ecoMode on a Thermostat* | *Thermostat enters energy-saving mode.* | *Pass* | | *008* | *Add a new device to SmartHome* | *Device is added to the list.* | *Pass* | | *009* | *Remove a device from SmartHome* | *Device is removed from the list.* | *Pass* | | *010* | *Display all devices' statuses* | *Correct status details are displayed.* | *Pass* | |