



# Smart Home Management System

## 1. Project Overview

Project Title: Smart Home Management System.

Project Type: Desktop Application.

Technology Stack: Java, JavaFX, FXML, CSS, JDBC, SQL Server.

The Smart Home Management System is a desktop-based application designed to simulate and manage a smart home environment. The system allows an administration to manage rooms, devices, energy, consumption, and system logs through an interactive and user-friendly graphical interface.

The project focuses on applying Object-Oriented Programming integration, and GUI development using JavaFX.

## 2. Project Objectives

- Provided a centralized dashboard to monitor the smart home system.
- Manage rooms and devices dynamically.
- Track energy usage and visualize it using charts.
- Maintain system logs for actions performed by the admin.
- Apply clean architecture principles (MVC+ DAO+ Services).
- Use a relational database for persistent data storage.

## 3. System Features

### 3.1 Authentication

- Admin login system.

- Validates credentials before granting access.

### 3.2 Dashboard

- Displays system overview.
- shows summary cards:
  - Total number of devices.
  - Total number of rooms.
  - Energy usage Status.
- smooth animations(Fade & Translate transitions).

### 3.3 Rooms Management

- View all rooms in the system.
- Add new rooms dynamically.
- Delete rooms if not needed.
- Each room displays:
  - Room Name
  - Number of devices inside the room.
- Rooms are stored in the Database.

### 3.4 Devices Management

- Devices are linked to rooms
- Each device belongs to specific room.
- Supports adding and removing devices.

### 3.5 Energy Monitoring

- Displays energy consumption data.
- Uses charts (Bar Chart / Line Chart) for visualization.
- Energy data is calculated based on devices usage.

### 3.6 Logs System

- Records important system actions such s:

- Adding rooms.
- Deleting rooms.
- Device operation.
- Logs are stored in database.

## 4. System Architecture

The project follows a layered Architecture:

### 4.1 presentation Layer (GUI)

- JavaFX
- FXML files
- CSS for styling

### 4.2 Controller Layer

- Handles user interfaces.
- Connects UI with business logic (Services).

### 4.3 Model Layer

- Represents system entities (Room, Device, Log, etc).

### 4.4 Data Access Layer (DAO)

- Handles database operation.
- Uses JDBC for SQL Server connection.

## 5. Technologies Used

Technology	Purpose
Java	Core programming language
JavaFX	GUI development
FXML	UI structure

CSS

UI styling

JDBC

Database connectivity

SQL Server

Data storage

IntelliJ IDEA

Development environment

## 6.Database Design

### Rooms Table

- id(Primary Key)
- name
- floor

### Devices Table

- id(Primary key)
- name
- type
- power Consumption

### Room Devices Table

- roomId(Foreign Key)
- deviceId(Foreign key)

### AuditLogs Table

- id
- action
- timetamp

## 7.Key Classes Description

### 7.1 Room Class

- Represents a room in the smart home.
- Contains room details and devices list.

## 7.2 Device Class

- Represents a smart device.
- Stores device properties such as type and power usage.

## 7.3 RoomDAO

- Handles all room-related database operation.
- Support CRUD operation.

## 7.4 SceneManager

- Manages navigation between application scenes.

## 7.5 Controllers

- DashboardController
- RoomController
- DeviceController
- EnergyController
- LogController

# 8.User Interface Design

- Clean and modern design.
- Sidebar navigation.
- Responsive layout using VBox, HBox, BorderPane.
- CSS animation and hover effects.

# 9.Error Handling & Validation

- Try-catch blocks for database operation.
- User-friendly alerts for failures.
- Console logging for debugging.

- Input validation for rooms and devices.
- Database exception handling.

## 10.Challenge Faced

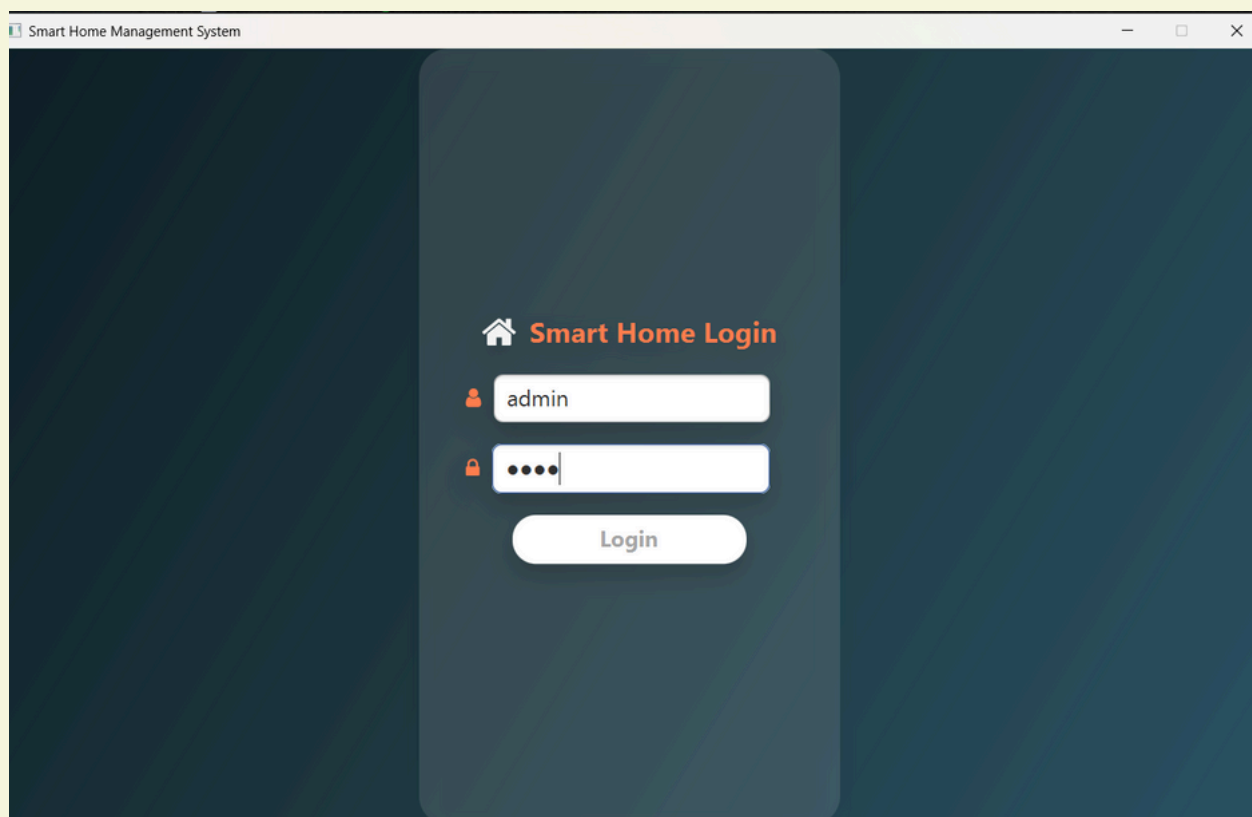
- Managing scene navigation dynamically.
- Synchronizing database data with UI components.
- Handling JavaFX CSS effects correctly.
- Maintaining clean separation between layers.

## 11.Future Enhancements

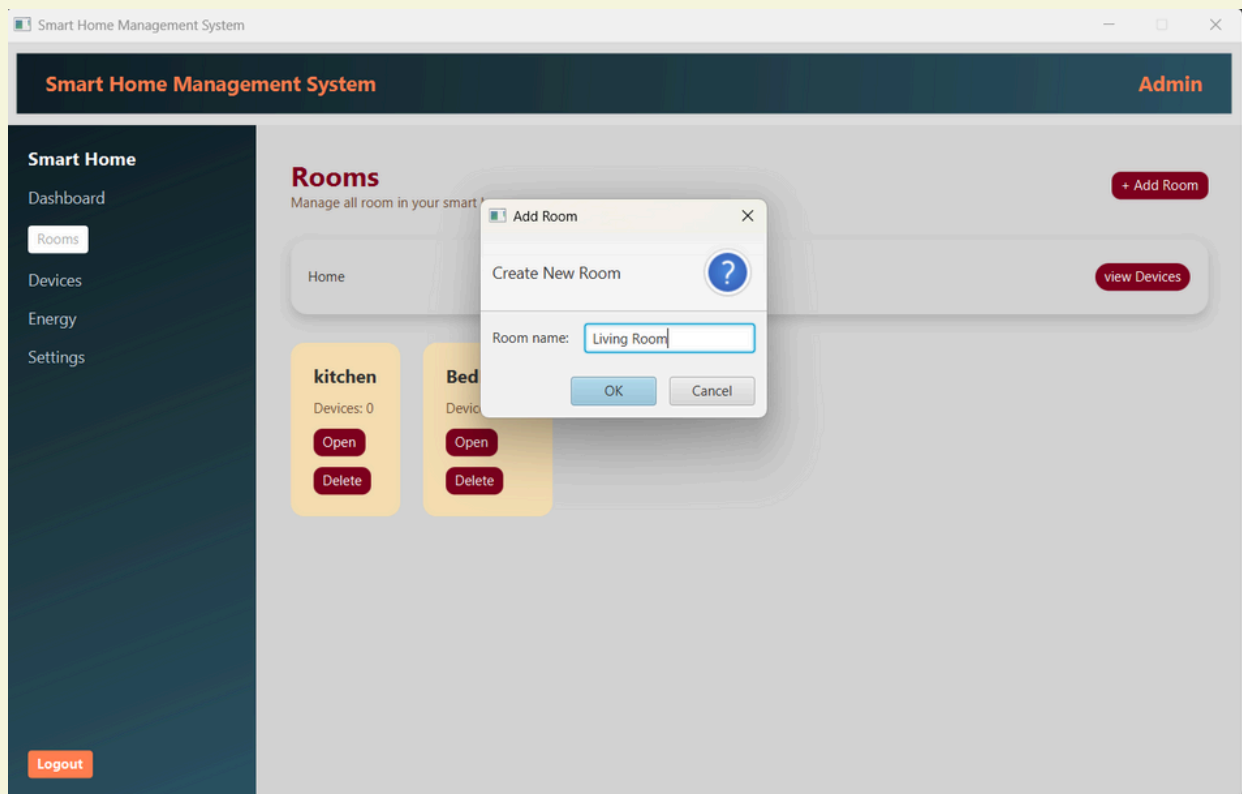
- User roles(Admin / User)
- Real-Time energy monitoring.
- Export logs to file.
- Cloud database integration.

## 12.Use Case Diagram

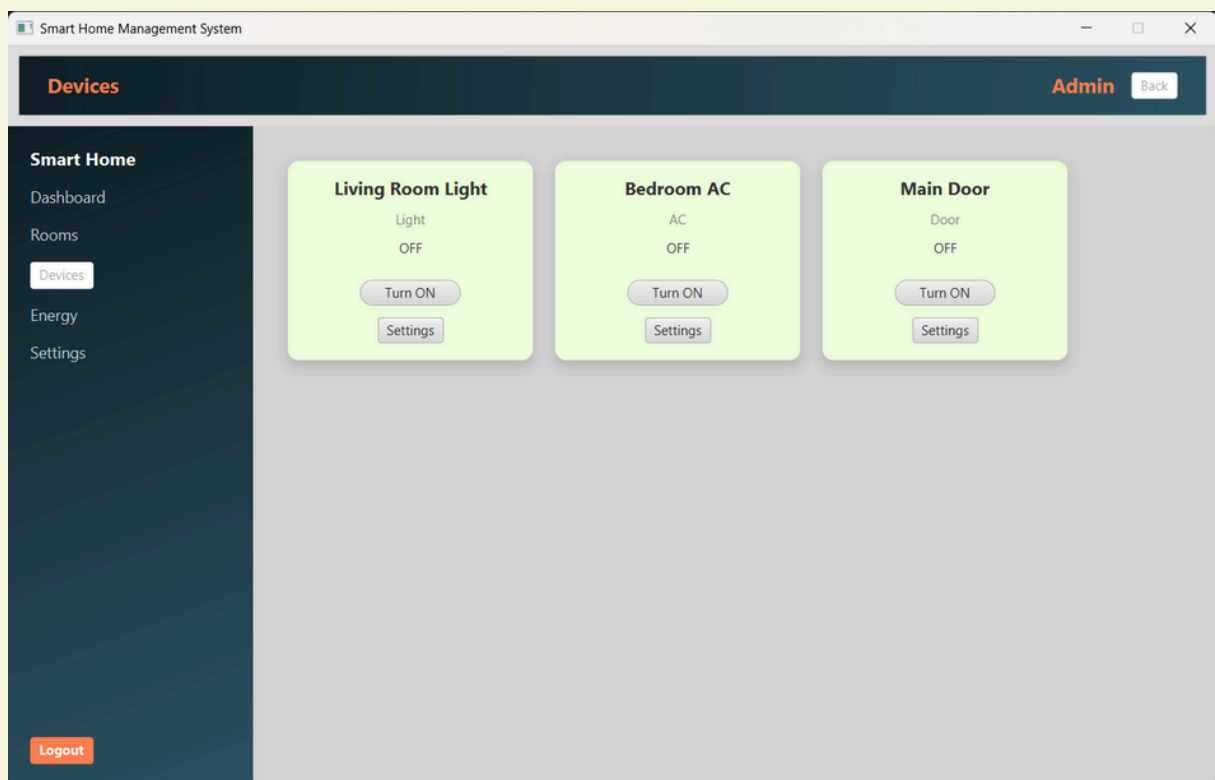
- Admin logs in.



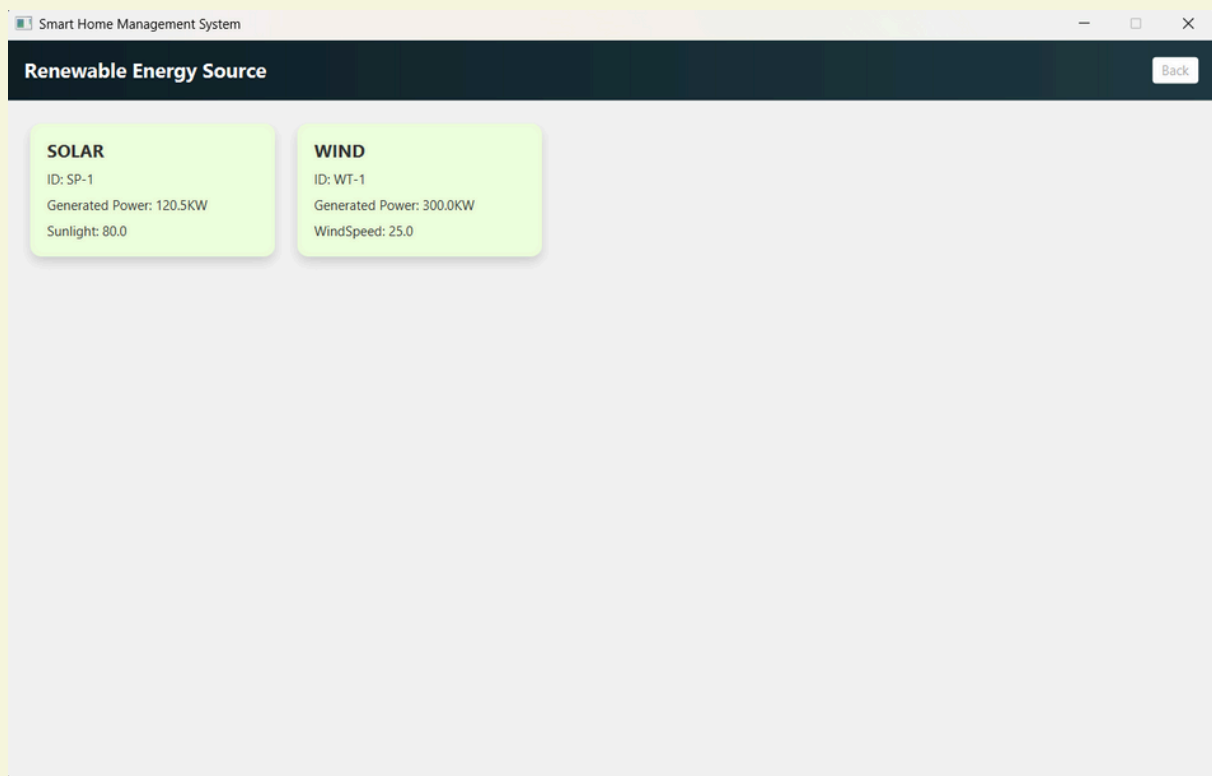
- Admin adds rooms



- Admin assigns devices to rooms



- Admin monitors energy usage



- System records logs automatically

Smart Home Management System

### Audit Logs

ID	User ID	Action	Target	Time
1	1	DEVICE_STOPPED	Device ID: 1	2025-12-18T21:46:10.503
2	1	DEVICE_STARTED	Device ID: 1	2025-12-18T21:46:11.003
3	1	DEVICE_STOPPED	Device ID: 1	2025-12-18T21:46:12.540
4	1	DEVICE_STARTED	Device ID: 1	2025-12-18T21:46:13.327
5	1	DEVICE_STARTED	Device ID: 2	2025-12-18T21:46:15.573
6	1	DEVICE_STOPPED	Device ID: 2	2025-12-18T21:46:16.340
7	1	DEVICE_STOPPED	Device ID: 1	2025-12-18T21:47:11.853
8	1	DEVICE_STARTED	Device ID: 1	2025-12-18T21:47:12.747
9	1	DEVICE_STARTED	Device ID: 2	2025-12-18T21:47:14.190
10	1	DEVICE_STOPPED	Device ID: 2	2025-12-18T21:47:14.907
11	1	DEVICE_STOPPED	Device ID: 1	2025-12-18T21:47:16.593
12	1	DEVICE_STARTED	Device ID: 1	2025-12-18T21:47:17.053
13	1	DEVICE_STOPPED	Device ID: 1	2025-12-18T21:54:05.583
14	1	DEVICE_STARTED	Device ID: 1	2025-12-18T21:54:06.633
15	1	DEVICE_STOPPED	Device ID: 1	2025-12-18T21:55:34.193
16	1	DEVICE_STOPPED	Device ID: 3	2025-12-18T23:34:26.427
17	1	SETTING_CHANGED	Device3 : Mode	2025-12-18T23:35:12.773
18	1	DEVICE_STARTED	Device ID: 3	2025-12-18T23:35:22.403
19	1	SETTING_CHANGED	Device3 : Mode	2025-12-18T23:35:31.710
20	1	SETTING_CHANGED	Device3 : Mode	2025-12-18T23:35:34.793
21	1	DEVICE_STARTED	Device ID: 1	2025-12-19T00:04:41.453
22	1	DEVICE_STOPPED	Device ID: 1	2025-12-19T00:04:42.650
23	1	SETTING_CHANGED	Device1 : Mode	2025-12-19T00:04:51.180

## 13.Non-Functional Requirements.

- Performance.
- Security.
- Scalability.



- Usability.

## **14.Logging Strategy**

- Track user action.
- Ensure system transparency.
- Improve debugging and security.

## **15.Future Scope**

- Mobile App
- Io T real devices.
- AI-based energy prediction.
- Cloud integration.

## **Conclusion**

In conclusion, the Smart Home Management System project successfully delivers a complete and well-structured solution for managing a smart home environment using modern software engineering concepts. The system integrates JavaFX for the user interface, SQL Server for persistent data storage, and a clean MVC-based architecture to ensure scalability, maintainability, and clarity of responsibilities.