Smart Home Energy Management System (SHEMS)

A Java-based Web Application for Monitoring & Optimizing Home Energy Usage

TEAM - 1

Under the guidance

Dr. T.Premalatha

TEAM MEMBERS & ROLES

- Keerthana Sanivada - Team Lead / Project Coordinator
- Thanuja Vanjarapu - Technical lead
- Sri Anjhanee Kunchanapalli - Developer - core module
- Hemanthraj D - Developer - supporting module
- Deepak P - Documentation & Deployment Owner

- Shift Role - Tester / Quality Analyst

Project Overview

Project Objective

- Develop a Smart Home Energy Management System
- Provide secure user authentication
- Enable centralized control of smart devices
- Monitor and manage energy consumption

Implemented Modules

- Module 1 – Authentication & User Access
- Module 2 – Smart Device Management

Folder Structure (How We Created the Project)

Backend Framework:

Spring Boot (Java)

Project Creation

- Generated using **Spring Initializer**
- Language: Java
- Build Tool: Maven

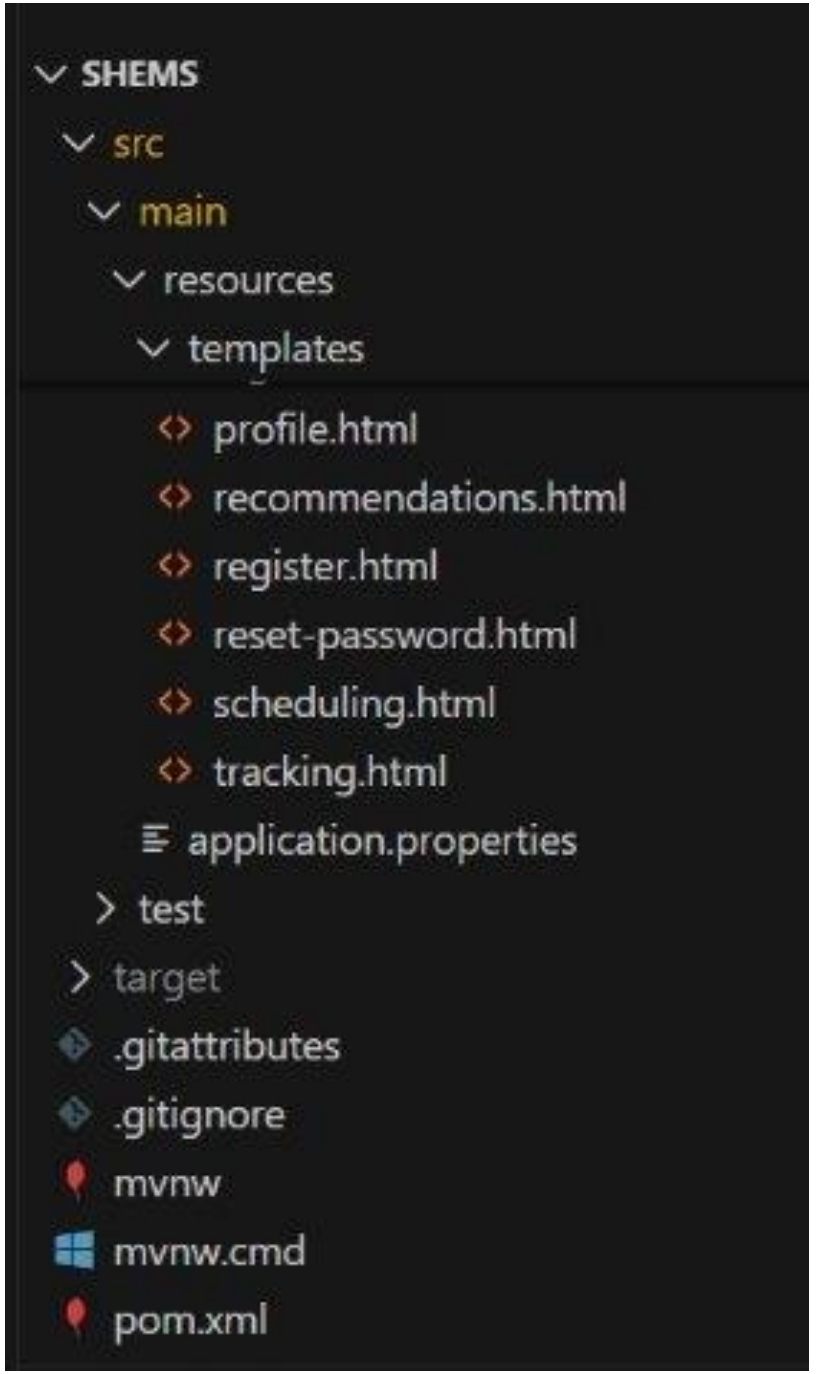
Dependencies Added

- Spring Web (MVC)
- Thymeleaf
- Spring Data JPA
- Spring Boot Starter
- MySQL Driver

Generated Structure

- src/main/java – Java backend code
- src/main/resources
  - static/ – CSS, JS, Images
  - templates/ – Thymeleaf HTML pages
- pom.xml – Maven dependency management


SHEMS
src
main
java\com\shems\shems
model
repository
DeviceRepository.java
UserRepository.java
service
AuthService.java
DeviceService.java
ShemsApplication.java
resources
static
templates
alerts.html
analytics.html
dashboard.html
device.html


SHEMS
src
main
resources
templates
profile.html
recommendations.html
register.html
reset-password.html
scheduling.html
tracking.html
application.properties
test
target
.gitattributes
.gitignore
mvnw
mvnw.cmd
pom.xml


SHEMS
.mvn
.vscode
src
main
java\com\shems\shems
config
controller
AuthController.java
DashboardController.java
DeviceController.java
PageController.java
dto
AuthRequest.java
AuthResponse.java
PasswordResetRequest.java
model
Device.java
User.java
repository

4 Layer Architecture Used

1

Controller Layer

- Handles HTTP requests
- Manages API and page routes

2

Service Layer

- Contains business logic
- Validations and processing

3

Repository Layer

- Database operations using JPA

4

Model Layer

- Entity classes mapped to DB tables

Advantages

- Clean code structure
- Easy maintenance
- Better scalability

Module 1: Authentication

Purpose

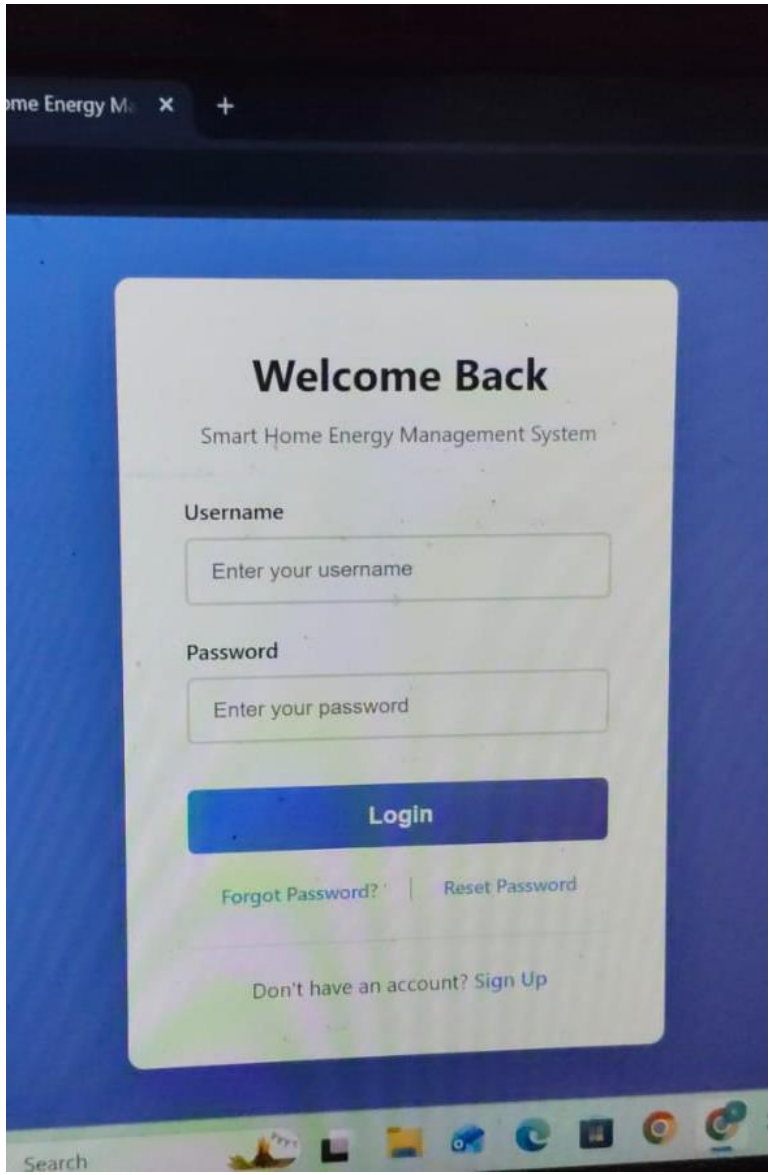
- Secure Login and Signup
- Role-based access
- Session management

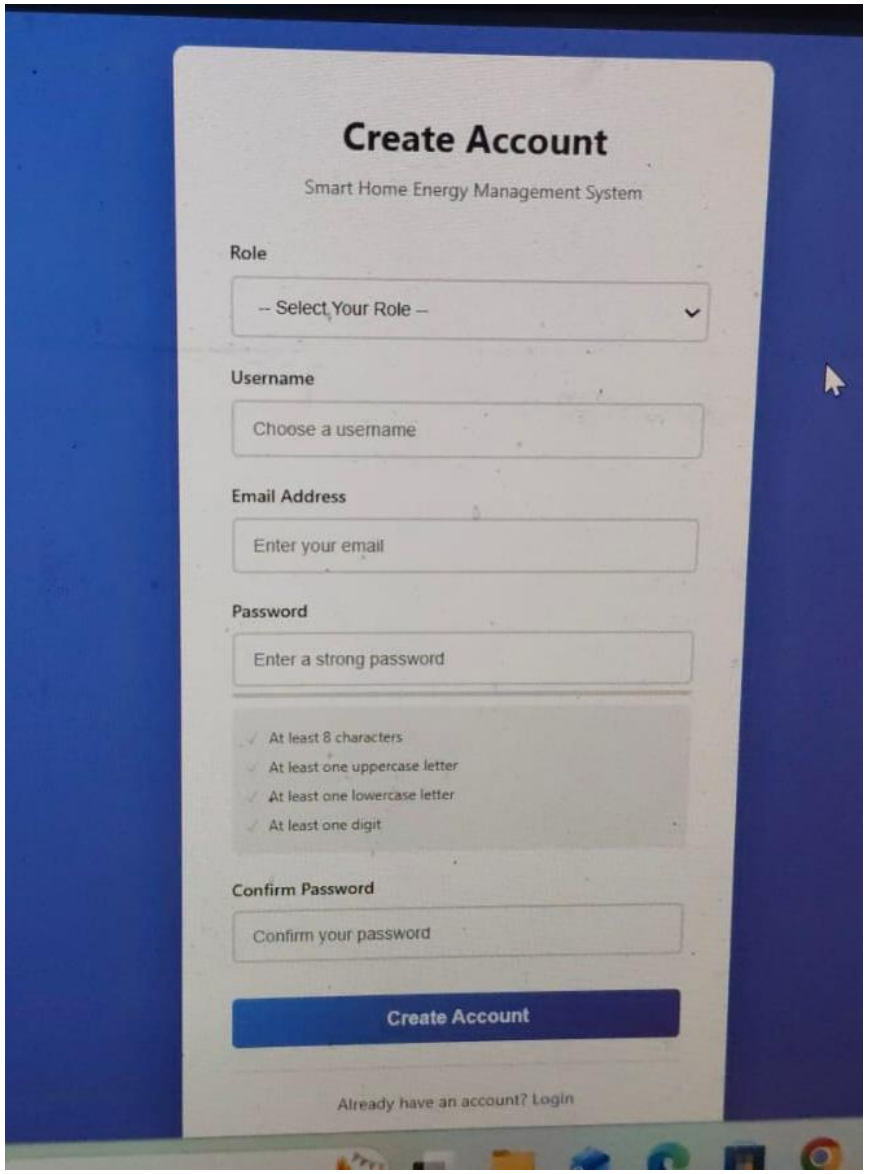
Main Classes

- **AuthController**
  - Handles authentication APIs
  - Manages user sessions
- **AuthService**
  - Login/signup validation
  - Password rules and encoding
- **User (Entity)**
  - Stores user details and roles
- **UserRepository**
  - Database operations for users

DTO Classes

- AuthRequest
- AuthResponse
- PasswordResetRequest


Welcome Back
Smart Home Energy Management System
Username
Enter your username
Password
Enter your password
Login
Forgot Password?
Reset Password
Don't have an account? Sign Up


Create Account
Smart Home Energy Management System
Role
-- Select Your Role --
Username
Choose a username
Email Address
Enter your email
Password
Enter a strong password
At least 8 characters
At least one uppercase letter
At least one lowercase letter
At least one digit
Confirm Password
Confirm your password
Create Account
Already have an account? Login

Module 2: Smart Device Management

Purpose

- Add, update, delete devices
- Control device ON/OFF
- Track energy consumption

Main Classes

- **DeviceController**
  - Manages device operations
- **DeviceService**
  - Business logic
  - Consumption calculation
- **Device Entity**
  - name, type, status, consumption
- **DeviceRepository**
  - JPA database operations

Role-Based Access

- OWNER → Full control
- FAMILY MEMBER → View + ON/OFF
- GUEST → View only

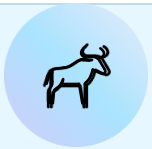
Technology Stack (Frontend)

HTML5 & CSS3

- Web pages and layouts

Bootstrap

- Responsive design
- Tables, buttons, UI components



Thymeleaf

- Server-side template engine
- Connects backend with HTML

Why Thymeleaf?

- Easy integration with Spring Boot
- No complex JavaScript needed

Database & Tools

Database

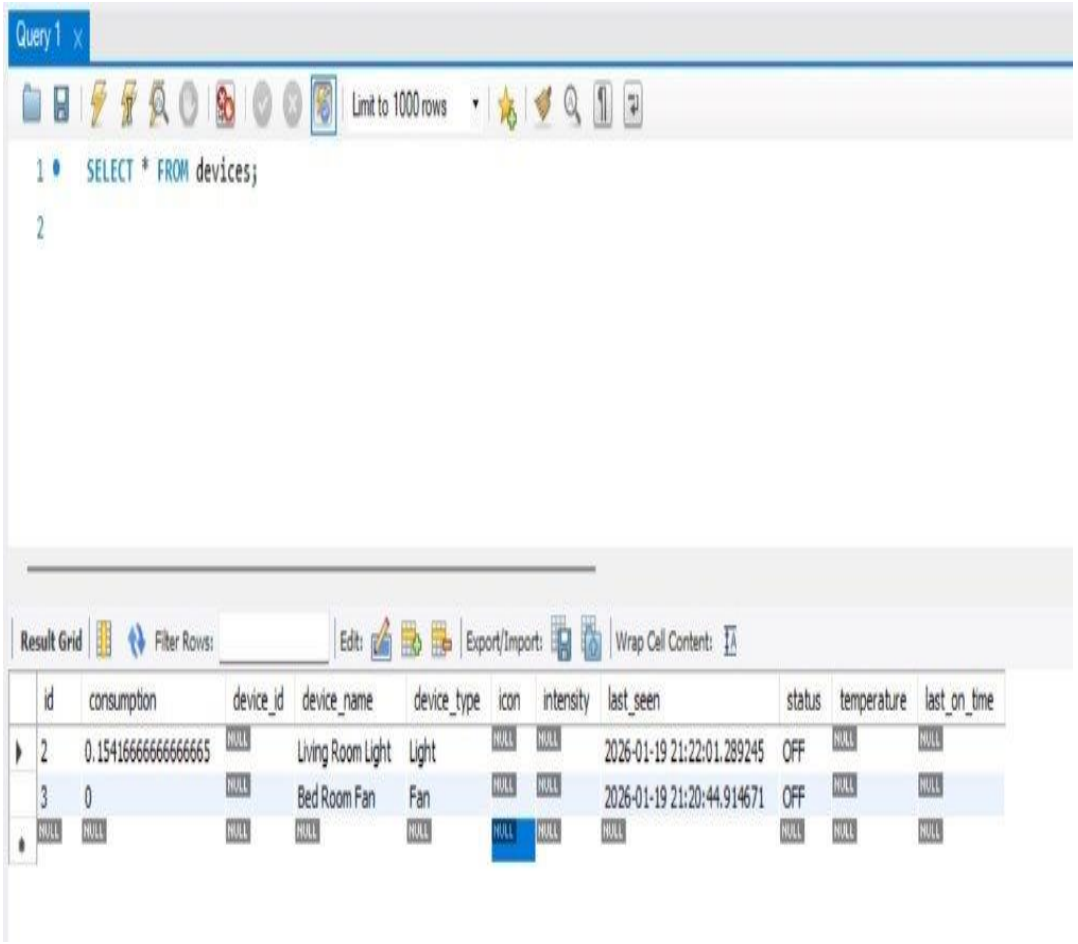
- **MySQL**
  - Stores users and devices
  - Ensures data integrity

Development Tools

- IntelliJ IDEA / STS
- Maven
- Git & GitHub

Version Control

- Separate branches for modules
- Team collaboration enabled



Query 1

Limit to 1000 rows

```
1  Use shems_db;
2  show tables;
3
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Tables_in_shems_db
devices
users

Result 1

Query 1

Limit to 1000 rows

```
1  SELECT * FROM users;
2
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

user_id	created_at	email	is_active	password	role	updated_at	username
1	2026-01-13 13:47:12.767938	323103310216@gvpce.ac.in	1	S2WcnRoYW5hQDE1Ng==	OWNER	2026-01-19 21:21:09.632398	Keerthana
2	2026-01-17 11:52:45.006875	thanuja@gmail.com	1	VGhhbnVqYUAxMjM=	FAMILY_MEMBER	NULL	Thanuja
5	2026-01-19 17:49:39.708468	anjhanee@gmail.com	1	QW5qaGFuZWVAMTIz	GUEST	2026-01-19 19:23:12.374350	SriAnjhanee
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Module Integration

How Modules 1 & 2 Work Together

- Module 1 authenticates user
- Module 2 allows actions after login
- Device access linked to roles
- Unauthorized actions blocked

Key Concept

> “Module 1 decides WHO can access, Module 2 decides WHAT they can control.”

Application Flow

1. User opens application

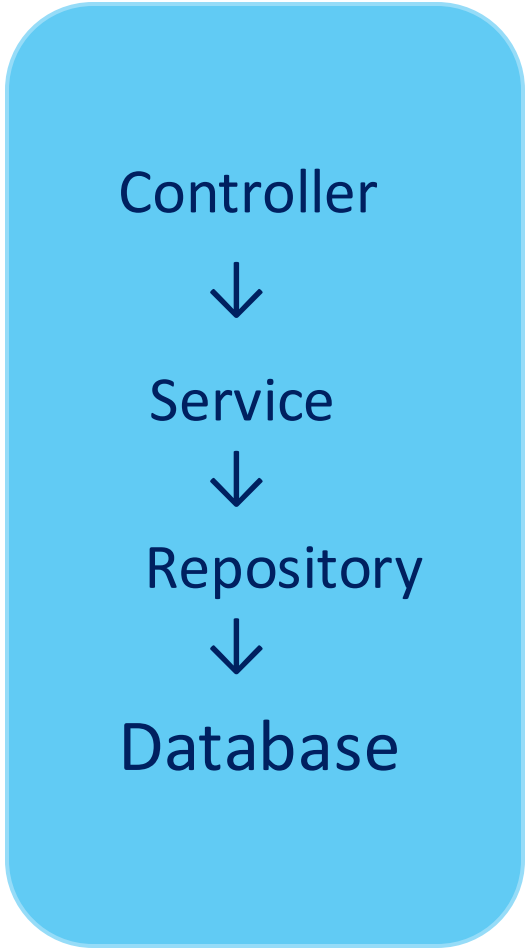
2. Request reaches Controller

3. Controller calls Service

4. Service interacts with Repository

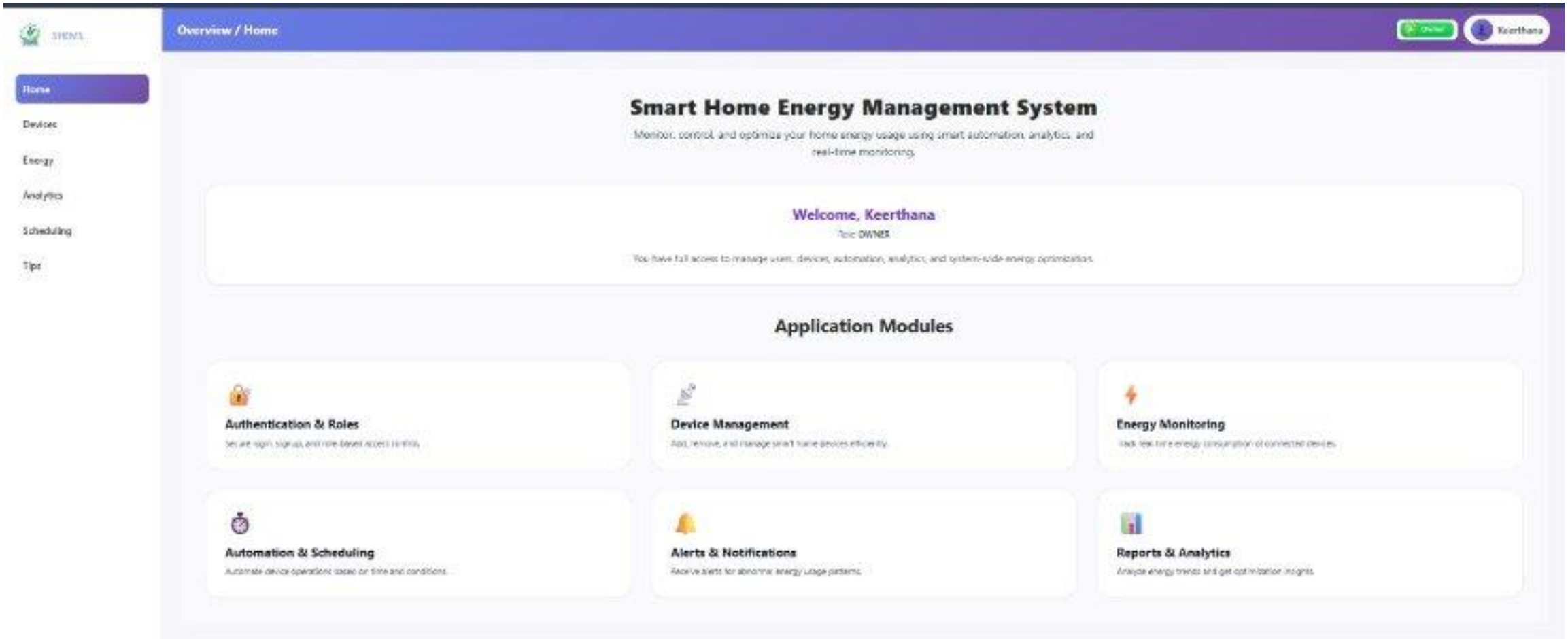
5. Repository accesses MySQL

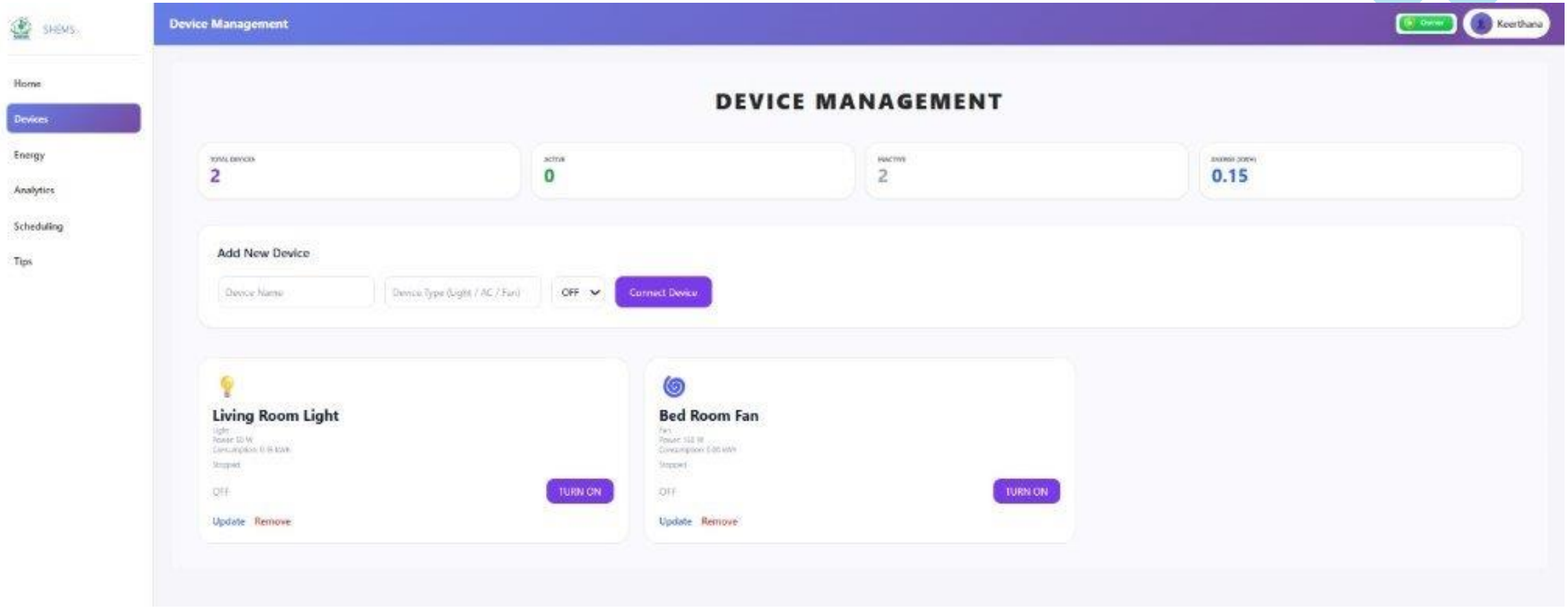
6. Result shown using Thymeleaf



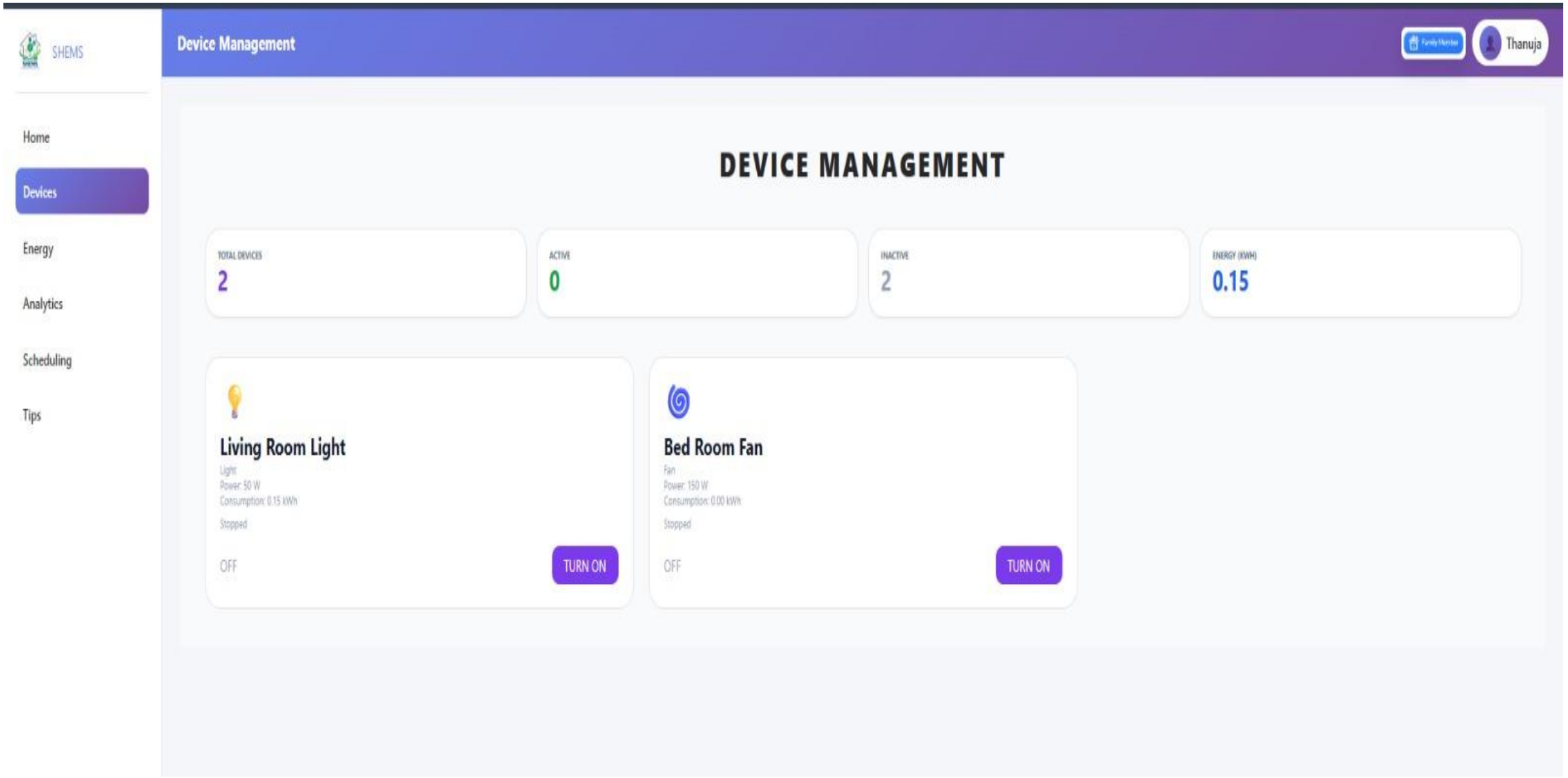

localhost:8081
Smart Home Energy Management System
EN
Login
Signup
Smart Home Energy Management
Monitor, control, and optimize your home energy usage using smart automation.
Application Modules
Authentication & Roles
Secure login, signup, and role based access control.
Device Management
Add, remove, and manage smart home devices.
Energy Monitoring
Track real-time energy consumption.
Automation & Scheduling
Automate devices based on time and conditions.
Alerts & Notifications
Receive alerts for abnormal usage.
Reports & Analytics
View reports and insights for optimization.
© 2026 Smart Home Energy Management System

ADMIN

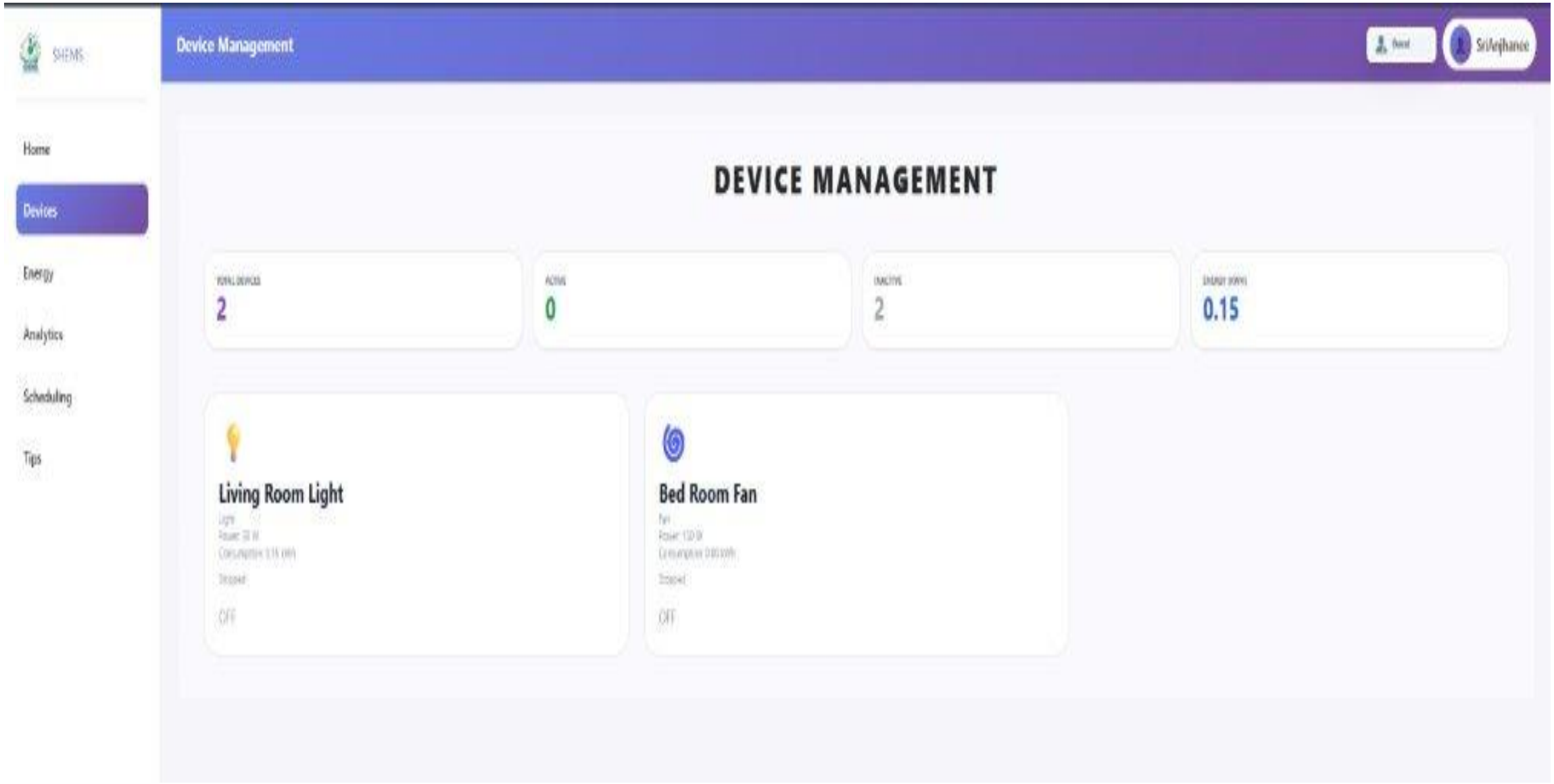

Overview / Home
Keerthana
Home
Devices
Energy
Analytics
Scheduling
Tips
Smart Home Energy Management System
Monitor, control, and optimize your home energy usage using smart automation, analytics, and real-time monitoring.
Welcome, Keerthana
Application Modules
Authentication & Roles
Device Management
Energy Monitoring
Automation & Scheduling
Alerts & Notifications
Reports & Analytics


SHEMS
Home
Devices
Energy
Analytics
Scheduling
Tips
Device Management
Owner
Keerthana
DEVICE MANAGEMENT
2
0
2
0.15
Add New Device
Device Name
Device Type (Light / AC / Fan)
OFF
Connect Device
Living Room Light
Stopped
OFF
TURN ON
Update
Remove
Bed Room Fan
Stopped
OFF
TURN ON
Update
Remove

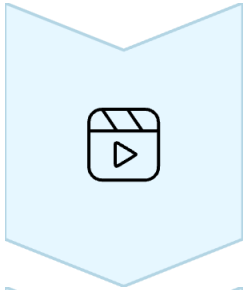
FAMILY MEMBER



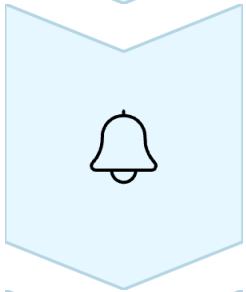
GUEST


SHEMS
Device Management
Home
Devices
Energy
Analytics
Scheduling
Tips
DEVICE MANAGEMENT
2
0
2
0.15
Living Room Light
OFF
Bed Room Fan
OFF

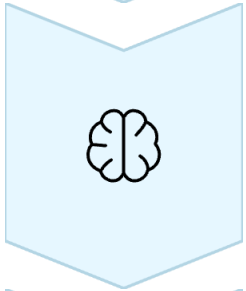
Future Enhancements



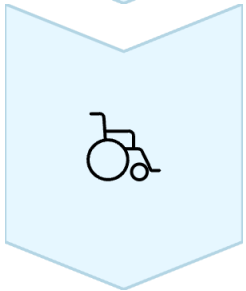
Real-time analytics



Alerts and notifications



AI-based power optimization



Mobile app support

Status

- Module 1 – Completed
- Module 2 – Completed

THANK YOU