Smart Home System Architecture

A S Puneeth Kumar
2024mt13102

Understanding the design and components of intelligent residences

Goals of the Smart Home System

Automation of Tasks

The system automates daily household tasks like lighting and gate management for convenience.

Secure Communication

Ensures secure, reliable communication between devices with user authentication to prevent unauthorized access.

User-Friendly Interface

Provides an easy-to-use interface for control and monitoring of home systems.

High Availability and Interoperability

Supports seamless operation with existing smart home technologies and protocols.

Utility Tree of Top ASRs

ATTRIBUTE	REFINEMENT	EXAMPLE TACTIC
Security	Authentication and access control	SSO, OAuth
Availability	Fault detection and recovery	Redundancy, ping checks
Interoperability	Protocol support	ZigBee, Bluetooth LE

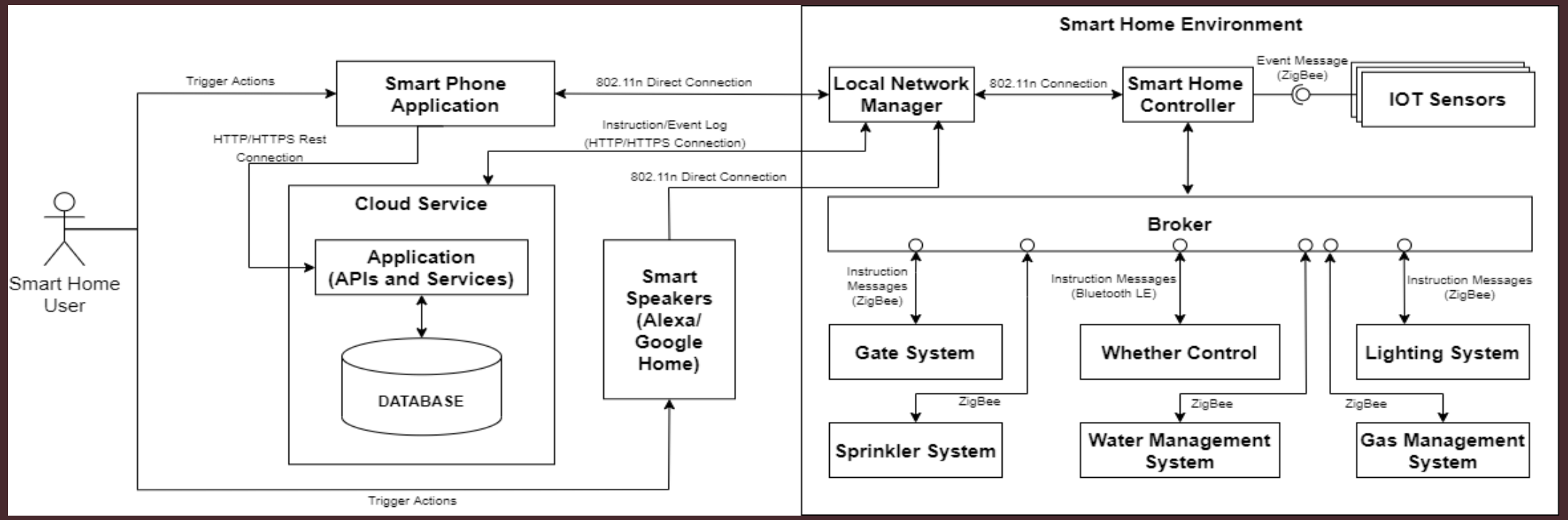
Tactics to Achieve ASRs

QUALITY ATTRIBUTE	SCENARIO	TACTICS	CACHING STRAEGY	TOOLS
Security	Authorized user interaction with role-based permissions	SSO, OAuth, role-based access, encrypted communication	Cache user roles and permissions, Short-Lived tokens	Authentication Service, Redis or Memcached
	Secure Communication Between Devices	Encrypted Communication, Device Authentication, Secure Boot and Firmware Validation	Cache Device credentials and session keys	MQTT with TLS, Hardware Security Modules, Local secure cache

QUALITY ATTRIBUTE	SCENARIO	TACTICS	CACHING STRAEGY	TOOLS
Availability	Fault detection and recovery	Ping/echo checks, redundancy, scalable cloud services	Cache device health status, fallback configurations	Prometheus or AWSCloudWatch, Redis, Load Balancers
	High Availability of smart services	Failover Mechanisms, Service Replication, Graceful Degradation	Cache recent user commands and device states, replicated service metadata	Distributed cache e.g.Redis cluster, Kubernetes or Docker swarm, Local storage

QUALITY ATTRIBUTE	SCENARIO	TACTICS	CACHING STRAEGY	TOOLS
Interoperability	Compatibility with standard protocols	Support for ZigBee, Bluetooth LE, Z-Wave	Cache protocol mappings, store device profiles	Open HAB, Redis or in-memory cache
	Integration with Third party smart devices	API-Based integration, Service Discovery	Cache API endpoints and authentication tokens, device discovery results	MQTT broker, Redis, Service Registry

Component Connector Diagram



Core Smart Home Components

The system includes mobile app, smart hub, IoT sensors, smart speakers, and broker components.

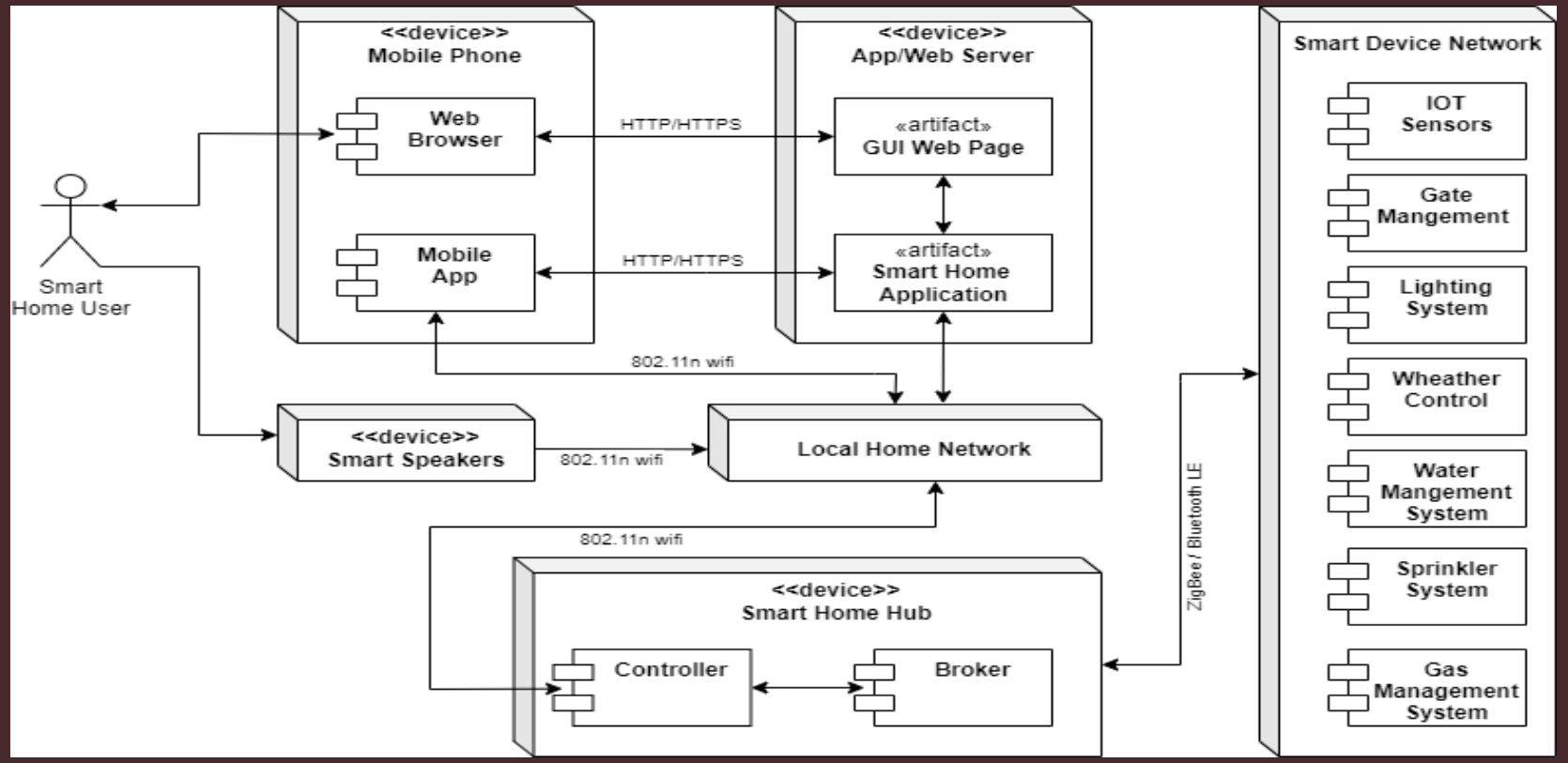
Communication Methods

Instructions are sent via cloud services or local Wi-Fi for device interaction.

Low-Energy Device Protocols

Devices communicate efficiently using low-energy wireless protocols for connectivity.

Deployment Diagram



Visualize System Deployment

Shows how components are deployed across environments including cloud and local hubs.

Operational Infrastructure Insight

Helps teams understand infrastructure setup for cost estimation and deployment planning.

Redundancy and Scalability

Illustrates built-in redundancy and scalability features supporting system reliability.

Labeled Connections & Communication Methods

1. **Mobile App → Smart Hub**
   - **Message**: User commands (e.g., lighting rules, gate control)
   - **Communication Method**:
     - **Cloud-based REST API** (when remote)
     - **Local Wi-Fi** (when on the same network)
   - **Label**: `UserCommandRequest (REST/Wi-Fi)`

2. **Smart Hub → IoT Sensors**
   - **Message**: Configuration updates, polling requests
   - **Communication Method**:
     - **ZigBee / Bluetooth LE / Z-Wave**
   - **Label**: `SensorConfigUpdate (ZigBee/BTLE)`

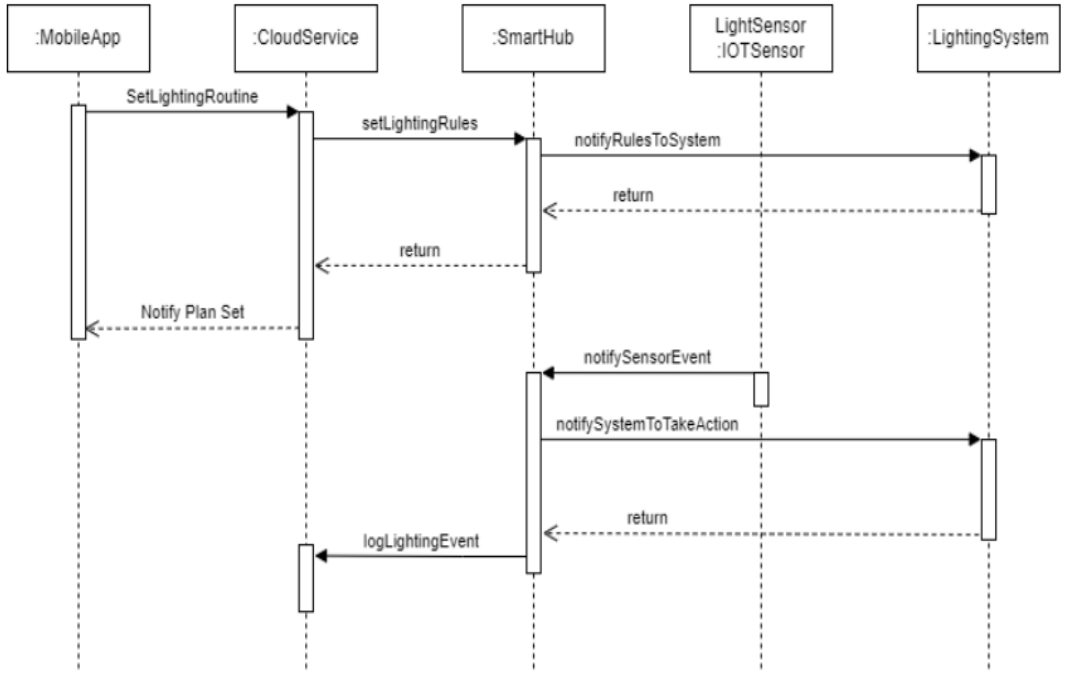
3. **IoT Sensors → Smart Hub**
   - **Message**: Sensor events (e.g., motion detected, temperature change)
   - **Communication Method**:
     - **Low-energy wireless protocols**
   - **Label**: `SensorEventNotification (ZigBee/BTLE)`

4. **Smart Hub → Smart Speakers**
   - **Message**: Voice command processing, status updates
   - **Communication Method**:
     - **Local Wi-Fi / MQTT**
   - **Label**: `VoiceCommandRelay (MQTT/Wi-Fi)`

SEQUENCE DIAGRAM FOR LIGHTING SYSTEM MANAGEMENT BY A SMART HOME SYSTEM

The sequence diagram to the right shows the use case of how rules are set for the lighting system's behavior and how these rules are enforced by the smart hub based on events recorded by the IOT Sensor.

- The authenticated user uses the app to set how the lighting should behave based on the signals given by the IOT sensor events (such as darkness level, or presence of a person etc.)
- The app sends these rules to the smart hub via the hosted application. The Hub lets the lighting system know about the change in it's expected behavior.
- When the optical IOT sensor sees an event, it notifies it to the smart hub. The hub then notifies the system to take the action based on this event.
- The system does the necessary job and notifies the hub which in turn logs the event in the hosted application



User Rule Configuration

An authenticated user sets lighting behavior rules through a mobile application interface.

Hub Communication

The hub receives and relays lighting rules from the app to the lighting system components.

Sensor-Triggered Actions

Lighting system actions are activated in response to sensor input events automatically.

Dynamic Interaction Clarity

The diagram clarifies component interactions and expected system behavior for developers.

Architectural Patterns

Broker Pattern Role

Facilitates decoupled communication between hubs and smart devices for smooth operation.

Client-Server Pattern Use

Manages cloud-hosted app requests linking mobile apps and home hubs efficiently.

Benefits of Patterns

Architectural patterns improve system scalability, reliability, and maintainability.

Key Learnings

High Availability Importance

High availability is critical for systems like gas management to ensure safety and reliability.

Fail-Safe System Design

Systems should default to safe modes or recover quickly from failures to protect users.

Future-Proof Architecture

Design must accommodate evolving smart home technologies for long-term adaptability.

Visual Planning Tools

Component and sequence diagrams help teams plan, estimate costs, and streamline development.