



CS350 Safehome Project

Software Design Specification (SDS) of Safehome

Project Team #6 Members:

20140576	Geunyeong Cheon
20220809	Alan Zhainar
20220814	Yernaz Akhmetov
20256402	Arda Eren

I. Overview.....	7
1. Introduction.....	7
2. Goal.....	7
3. How the design work proceeded.....	7
4. Assumptions.....	7
II. Architectural structure.....	10
1. OVERALL ARCHITECTURE.....	10
2. ARCHITECTURE – INTELLIGENT SECURITY.....	10
3. ARCHITECTURE – LIVE SURVEILLANCE.....	11
4. ARCHITECTURE – SYSTEM AND USER MANAGEMENT.....	11
5. ARCHITECTURE – REMOTE ACCESS AND ACCOUNT.....	12
6. ARCHITECTURE – INDOOR MONITORING AND DEVICE CONTROL.....	13
III. Class Diagram.....	13
1. CLASS DIAGRAM – WHOLE SYSTEM OVERVIEW.....	13
2. CLASS DIAGRAM – INTELLIGENT SECURITY.....	15
2.1 Class diagram - Intelligent security overall.....	16
2.2 Class diagram - Sensor monitoring.....	16
2.3 Class diagram - Incident management.....	16
2.4 Class diagram - Security mode control.....	18
3. CLASS DIAGRAM – LIVE SURVEILLANCE.....	19
3.1 Surveillance overall.....	19
3.2 Camera Viewing and Control.....	20
3.3 Recording and Evidence Management.....	21
3.4 Surveillance settings.....	21
4. CLASS DIAGRAM – SYSTEM AND USER MANAGEMENT.....	22
4.1 Device Management - Complete Class Diagram.....	22
4.2 System Status and Logs - Complete Class Diagram.....	23
4.3 User and Permission Management - UML Class Diagram.....	23
5. CLASS DIAGRAM – REMOTE ACCESS AND ACCOUNT.....	24
5.1 Identity & Profile Model.....	24
5.2 Authentication & Session Model.....	25
5.3 2FA Management Model.....	26
5.4 Verification & Recovery Model.....	27
5.5 RBAC Integration & Audit Model.....	27
6. CLASS DIAGRAM – INDOOR MONITORING & DEVICE CONTROL.....	29
6.1 Class Diagram - Indoor Monitoring & Device Control Overall.....	29
6.2 Class Diagram - Class Diagram - Configuration Core.....	29
6.3 Class Diagram - Device Control Core.....	30
6.4 Class Diagram - Monitoring & Reports.....	31
6.5 Class Diagram - Policy, Logging & Notifications.....	31
IV. CRC cards.....	31
1. CRC CARDS – INTELLIGENT SECURITY.....	31
1.1 CRC CARD - User.....	32
1.2 CRC CARD - AdminHomeOwner.....	32

1.3 CRC CARD - Homeowner.....	32
1.4 CRC CARD - Guest.....	33
1.5 CRC CARD - ExternalSecurityService.....	33
1.6 CRC CARD - IVR/Call Gateway Service.....	34
1.7 CRC CARD - EmergencyContact.....	34
1.8 CRC CARD - SafeHomeHub.....	34
1.9 CRC CARD - CloudServer.....	35
1.10 CRC CARD - MobileApp.....	35
1.11 CRC CARD - Device.....	36
1.12 CRC CARD - Sensor.....	36
1.13 CRC CARD - ContactSensor.....	37
1.14 CRC CARD - ShockSensor.....	37
1.15 CRC CARD - MotionSensor.....	38
1.16 CRC CARD - EnvironmentalSensor.....	38
1.17 CRC CARD - FireSmokeSensor.....	39
1.18 CRC CARD - COSensor.....	39
1.19 CRC CARD - GasSensor.....	39
1.20 CRC CARD - LeakSensor.....	40
1.21 CRC CARD - SoundSensor.....	40
1.22 CRC CARD - Camera.....	41
1.23 CRC CARD - Siren.....	41
1.24 CRC CARD - SmartDoorLock.....	42
1.25 CRC CARD - SmarHomeDevice.....	42
1.26 CRC CARD - SecurityMode.....	42
1.27 CRC CARD - Alarm.....	43
1.28 CRC CARD - Incident.....	43
1.29 CRC CARD - User.....	44
1.30 CRC CARD - ActivityLog.....	45
1.31 CRC CARD - Notification.....	45
1.32 CRC CARD - ConfigurationProfile.....	46
1.33 CRC CARD - AlarmPolicy.....	46
1.34 CRC CARD - NotificationPolicy.....	47
1.35 CRC CARD - CooldownTimer.....	47
1.36 CRC CARD - PanicButton.....	47
2. CRC CARDS – LIVE SURVEILLANCE.....	48
2.1 CRC CARD - Camera.....	48
2.2 CRC CARD - VideoStream.....	49
2.3 CRC CARD - AudioChannel.....	49
2.4 CRC CARD - CameraController.....	49
2.5 CRC CARD - SafeHomeHub.....	50
2.6 CRC CARD - Recording.....	50
2.7 CRC CARD - PlaybackController.....	51
2.8 CRC CARD - EvidenceExporter.....	51
2.9 CRC CARD - StorageRepository.....	51

2.10 CRC CARD - CloudServer.....	52
2.11 CRC CARD - RecordingSettings.....	52
2.12 CRC CARD - NotificationPolicy.....	52
2.13 CRC CARD - SettingsManager.....	53
2.14 CRC CARD - SafeHomeHub.....	53
3. CRC CARDS – SYSTEM AND USER MANAGEMENT.....	54
3.1 Device Classes.....	54
3.2 System Status and Logs Classes.....	56
3.3 User and Permission Management Classes.....	58
3.4 Supporting Classes.....	61
4. CRC CARDS – REMOTE ACCESS AND ACCOUNT.....	64
4.1 CRC CARD - Gateway.....	64
4.2 CRC CARD - LoginService.....	64
4.3 CRC CARD - AuthService.....	65
4.4 CRC CARD - SessionManager.....	65
4.5 CRC CARD - MFAService.....	65
4.6 CRC CARD - PolicyEnforcer.....	66
4.7 CRC CARD - IdentityService.....	66
4.8 CRC CARD - VerificationService.....	66
4.9 CRC CARD - RemoteAccessController.....	67
4.10 CRC CARD - TokenService.....	67
4.11 CRC CARD - DeviceTrustService.....	67
4.12 CRC CARD - NotificationService.....	68
5. CRC CARDS – INDOOR MONITORING AND DEVICE CONTROL.....	68
5.1 CRC CARD - ConfigurationManager.....	68
5.2 CRC CARD - ConfigurationProfile.....	68
5.3 CRC CARD - SafetyZone.....	69
5.4 CRC CARD - Room.....	69
5.5 CRC CARD - DeviceController.....	69
5.6 CRC CARD - Command.....	70
5.7 CRC CARD - CommandResult.....	70
5.8 CRC CARD - SmartHomeDevice (abstract).....	70
5.9 CRC CARD - SmartLight.....	70
5.10 CRC CARD - Thermostat.....	71
5.11 CRC CARD - VentilationController.....	71
5.12 CRC CARD - SmartPlug.....	71
5.13 CRC CARD - AirQualitySensor.....	71
5.14 CRC CARD - PowerMeter.....	72
5.15 CRC CARD - EnergyReport.....	72
5.16 CRC CARD - NotificationPolicy.....	72
5.17 CRC CARD - CooldownTimer.....	72
5.18 CRC CARD - ActivityLog.....	73
5.19 CRC CARD - Log.....	73
5.20 CRC CARD - StorageManager.....	73

5.21 CRC CARD - SystemSettings.....	73
V. State Diagram.....	74
1. STATE DIAGRAM – INTELLIGENT SECURITY.....	74
1.1 State Diagram - SecurityMode.....	74
1.2 State Diagram - Alarm.....	75
1.3 State Diagram - Incident.....	76
1.4 State Diagram - AlarmVerification.....	77
1.5 State Diagram - Sensor.....	78
1.6 State Diagram - PanicButton.....	79
2. STATE DIAGRAM – LIVE SURVEILLANCE.....	79
2.1 State Diagram - CameraLifecycle.....	79
2.2 State Diagram - VideoStream.....	81
2.3 State Diagram - Recording Lifecycle.....	82
2.4 State Diagram - Playback Lifecycle.....	83
2.5 State Diagram - Export Manager.....	84
3. STATE DIAGRAM – SYSTEM AND USER MANAGEMENT.....	84
1. Device Lifecycle State Diagram.....	84
2. Device Registration Process State Diagram.....	85
3. User Account Lifecycle State Diagram.....	86
4. User Session State Diagram.....	86
5. System Status Dashboard State Diagram.....	87
6. Activity Log State Diagram.....	87
7. Device Health Monitor State Diagram.....	88
8. Permission Management State Diagram.....	89
9. Configuration Profile State Diagram.....	90
10. Audit Log State Diagram.....	90
4. STATE DIAGRAM – REMOTE ACCESS AND ACCOUNT.....	92
4.1 State Diagram - Account Lifecycle.....	92
4.2 State Diagram - Remote Access session.....	92
4.3 State Diagram - Login with MFA.....	93
4.4 State Diagram - Password reset flow.....	94
4.5 State Diagram - Token / Session lifecycle.....	95
4.6 State Diagram - Device trust enrollment & check.....	96
5. STATE DIAGRAM – INDOOR MONITORING AND DEVICE CONTROL.....	97
5.1 State Diagram - SmartHomeDevice (Lifecycle).....	97
5.2 State Diagram - SmartLight (Behavior).....	98
5.3 State Diagram - Thermostat (Mode/Control).....	98
5.4 State Diagram - VentilationController (Control Logic).....	99
5.5 State Diagram - AirQualitySensor (Quality Bands with Hysteresis).....	99
5.6 State Diagram - DeviceController (Command Execution).....	100
5.7 State Diagram - ConfigurationManager (Apply Profile).....	100
VI. Design Evaluation.....	101
1. Architectural Design Metric.....	101
a. Design Structure Quality Index (DSQI).....	101

2. CK Metrics.....	103
3. MOOD Metric.....	103
a. MIF (Method Inheritance Factor).....	103
CF = 51 / (35 * 35 – 35) = 0.043.....	105
4. OO Metric Proposed by Lorenz and Kidd.....	105
5. Explanation of Metrics.....	106
VII. WHO DID WHAT.....	106
VIII. MEETING LOGS.....	108
Meeting Log 1: November 5, 2025.....	108
Meeting Log 2: November 11, 2025.....	108
In Question to Professor or TA.....	110
Q1. About same classes.....	110
Q2. Complex Class diagram.....	111
Meeting Log: November 11, 2025.....	113
APPENDIX A. GLOSSARY.....	114
APPENDIX C: Design Clarifications.....	117

I. Overview

1. Introduction

This document describes the design model of SafeHome system proposed in the previous report. Since the design phase of the product is directly connected to the implementation phase, we focus on the well-formed and concrete design of the system. Architectural structure, class diagram, CRC cards, state diagrams, and sequence diagrams are presented to portray the design of the system.

2. Goal

- 1) Completely follow the requirement and the analysis model.
- 2) Achieve low coupling, high cohesion, and modularity.
- 3) Pursue testability, integrity, efficiency, maintainability, and reliability.
- 4) Minimize complexity and consider reusability and flexibility.

3. How the design work proceeded

- 1) Extraction of classes: To achieve correctness of the design model we used the method in chapter 8.7 of SEPA. We reviewed nouns and verbs from use case scenarios to extract classes.
- 2) Creation of architectural structure: Based on the classes extracted and the use case scenario, we created the architectural structure of the SafeHome system.
- 3) Creation of class diagram: On the basis of extracted classes and architectural structure, we created the class diagram considering the implementation.
- 4) Creation of CRC card.
- 5) Refinement of class diagram: By testing the design using CRC card and reviewing the requirement document and the first report, we refined the class diagram. In addition we focused on ways to achieve low coupling and high cohesion. Actual implementation plan became more concrete and added some classes from java and for database access
- 6) Creation of state diagram.
- 7) Refinement of class diagram: Added some missing functions and attributes
- 8) Creation of sequence diagram: Based on the use case scenario we created the sequence diagram. It enabled us to check if the design followed the requirement specification and the first report.
- 9) Review of state and sequence diagram: Tested based on the first report.
- 10) Refinement of class diagram: Added some missing functions and attributes in the view point of implementation.

4. Assumptions

1. Pet Sensor Function(1.1.4) is defined in safehome dialog slide 58-59
2. Alarm Trigger and Instant Notification Function(1.2.1) is defined in safehome dialog slide 5

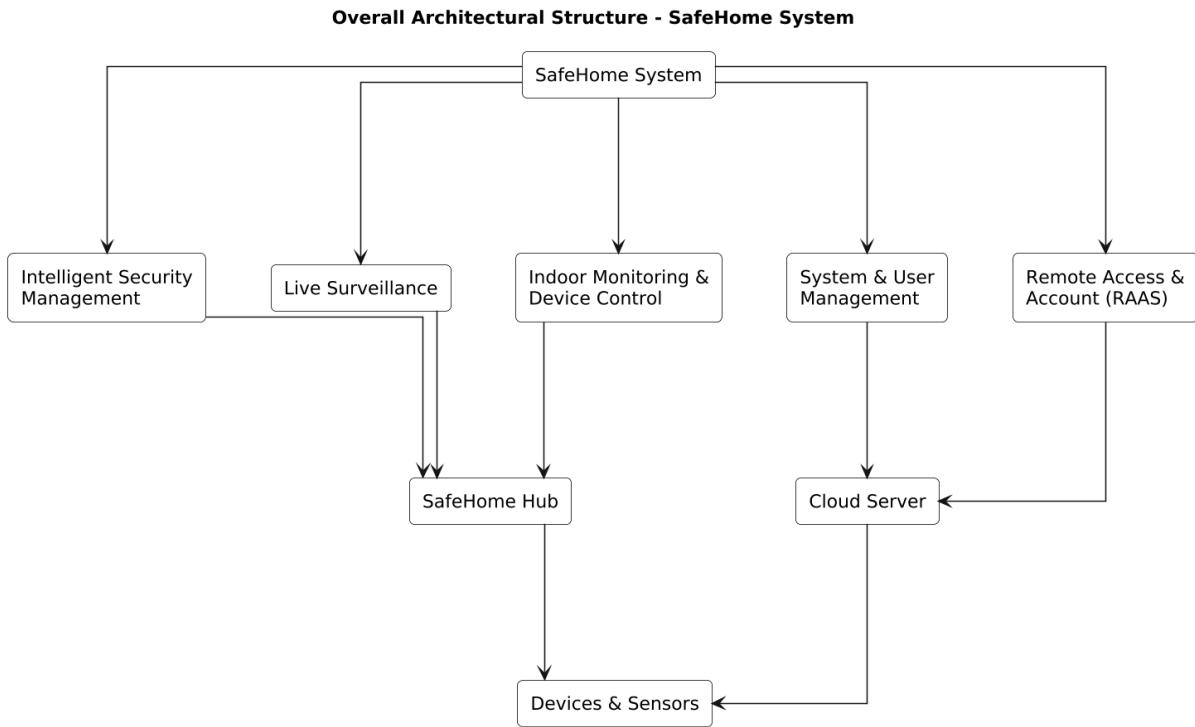
3. Emergency Service Integration and Auto Call Function(1.2.3) is defined in safehome dialog slide 6, 7
4. One-Touch Modes (Away, Home, Sleep) Function(1.3.1) is defined in safehome dialog slide 9
5. Sensor Bypass Function(1.3.2) is defined in safehome dialog slide 10
6. Single Camera Live View Function(2.1.1) is defined in safehome dialog slide 29-31
7. Two-Way Audio Function(2.1.2) is defined in safehome dialog slide 16
8. Camera Lock and Unlock Function(2.1.3) is defined in safehome dialog slide 19-31
9. Search and Playback Recordings Function(2.2.1) is defined in safehome dialog slide 29-31
10. Recording Settings Function(2.3.1) is defined in safehome dialog slide 29-31
11. Add and Place New Devices Function(3.1.1) is defined in safehome dialog slide 58
12. Activity Logs and Timeline Function(3.2.2) is defined in safehome dialog slide 39
13. User Role and Access Control Function(3.3.1) is defined in safehome dialog slide 70
14. Sign Up Function(4.1.1) is defined in safehome dialog slide 41
15. Log In Function(4.1.2) is defined in safehome dialog slide 42
16. Log Out Function(4.1.3) is defined in safehome dialog slide 43
17. Password Recovery and Reset Function(4.1.4) is defined in safehome dialog slide 44
18. Edit Profile Information Function(4.1.5) is defined in safehome dialog slide 45
19. Change Password Function(4.1.6) is defined in safehome dialog slide 46
20. Two-Factor Authentication Management Function(4.1.7) is defined in safehome dialog slide 47
21. Indoor Device Control Function(5.1.1) is defined in safehome dialog slide 39
22. Indoor Air Quality Monitoring and Ventilation Integration Function(5.2.1) is defined in safehome dialog slide 27
23. Real-Time Power Consumption Monitoring and Reporting Function(5.2.2) is defined in safehome dialog slide 29
24. Secure Onboarding (Device Registration Security) Function (Originally 3.1.2) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.
25. OTA Firmware Update Function (Originally 3.1.3) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.
26. Health & Tamper Monitoring (Device Anomaly Detection) Function (Originally 3.2.3) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.
27. Global Priority & Version Policy Function (Originally 3.3.2) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.

28. Policy & Compliance Function (Originally 3.4) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.
29. Data Retention & Deletion Policy Function (Originally 3.4.1) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.
30. Encryption/Transmission Security Policy Function (Originally 3.4.2) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.
31. Privacy Notice & Consent Function (Originally 3.4.3) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.
32. Time Synchronization (NTP) Policy Function (Originally 3.4.4) was added because we determined it is necessary for improving the overall quality of the service. However, we have assumed that it should be developed in the next iteration, as the current SRS is for the first release.
33. It was decided to add the **Physical Intrusion Detection Function(1.1.1)** in the meeting on 10.29 after discussion in the meeting on 10.26.
34. It was decided to add the **Environmental Hazard Detection Function(1.1.2)** in the meeting on 10.29 after discussion in the meeting on 10.26.
35. It was decided to add the **Outdoor Motion Detection Function(1.1.3)** in the meeting on 10.29 after discussion in the meeting on 10.26.
36. It was decided to add the **Alarm Verification Step Function(1.2.2)** in the meeting on 10.29.
37. It was decided to add the **Panic Button Function(1.2.4)** in the meeting on 10.29.
38. It was decided to add the **Sensor Activation and Deactivation Function(1.3.3)** in the meeting on 10.29 after discussion in the meeting on 10.26.
39. It was decided to add the **Camera Activation and Deactivation Function(2.1.4)** in the meeting on 10.29 after discussion in the meeting on 10.26.
40. It was decided to add the **Evidence Sharing and Export Function(2.2.2)** in the meeting on 10.29.
41. It was decided to add the **Notification Policy and Cooldown Function(2.3.2)** in the meeting on 10.29.
42. It was decided to add the **System Status Dashboard Function(3.2.1)** in the meeting on 10.29.
43. Floor plan configuration and hardware deployment is complete and out of the scope of our project.
44. “System administrator” in our use case scenarios is not a person who is in charge of managing the system. It is the system itself acting as a facilitator for the use of system functionalities

45. Between mobile and web, we have decided to make the mobile app our first release.
46. Although the SRS defines device status checks (battery level, connectivity, signal strength) in UC 3.2.1 System Status Dashboard, it does not specify how these metrics are computed. We assume that the system internally maintains a Device Health Monitoring module, which continuously computes diagnostic results, battery metrics, and connectivity quality for each device. References: UC 3.2.1 (System Status Dashboard), Goals – “Ensure high system reliability”
47. The SRS requires a real-time, filterable dashboard but does not describe its implementation structure. We assume the dashboard is implemented as a widget-driven interface where each widget retrieves data from assigned sources and supports customizable layouts for the System Status Dashboard. References: UC 3.2.1 (“Status Dashboard,” device list, filtering, sorting)
48. While UC 3.2.2 focuses on reviewing logs, it does not define internal analysis capabilities. We assume the system includes a Log Analyzer subsystem that identifies patterns, detects anomalies, and generates internal reports to support system monitoring and future enhancements. References: UC 3.2.2 (Activity Logs and Timeline), Goals – “proactive security framework”
49. The SRS explains how devices are controlled (UC 5.1.1 Indoor Device Control), but not how commands are internally structured or executed. We assume an internal Command Abstraction Layer is used to encode device commands, track execution results, and manage retry logic across heterogeneous device types. References: UC 5.1.1 (Indoor Device Control), UC 3.1.1 (Add and Configure New Devices)
50. The SRS describes playback, event logs, and evidence export, but does not define how recorded media is linked to events. We assume the system uses a MediaReference object to provide metadata, storage paths, and linkage between recordings/snapshots and their corresponding events. References: UC 2.2.1 (Search and Playback Recordings), UC 3.2.2 (Activity Logs — “links to associated media”)
51. Device configuration and notification policies (UC 2.3.2) use editable numeric ranges and rules but the SRS does not specify validation logic. We assume the system includes a validation rule engine that enforces data types, valid ranges, read-only constraints, and schema checking for all device and system settings. References: UC 2.3.2 (Invalid input handling), UC 3.1.1 (initial configuration during device setup)

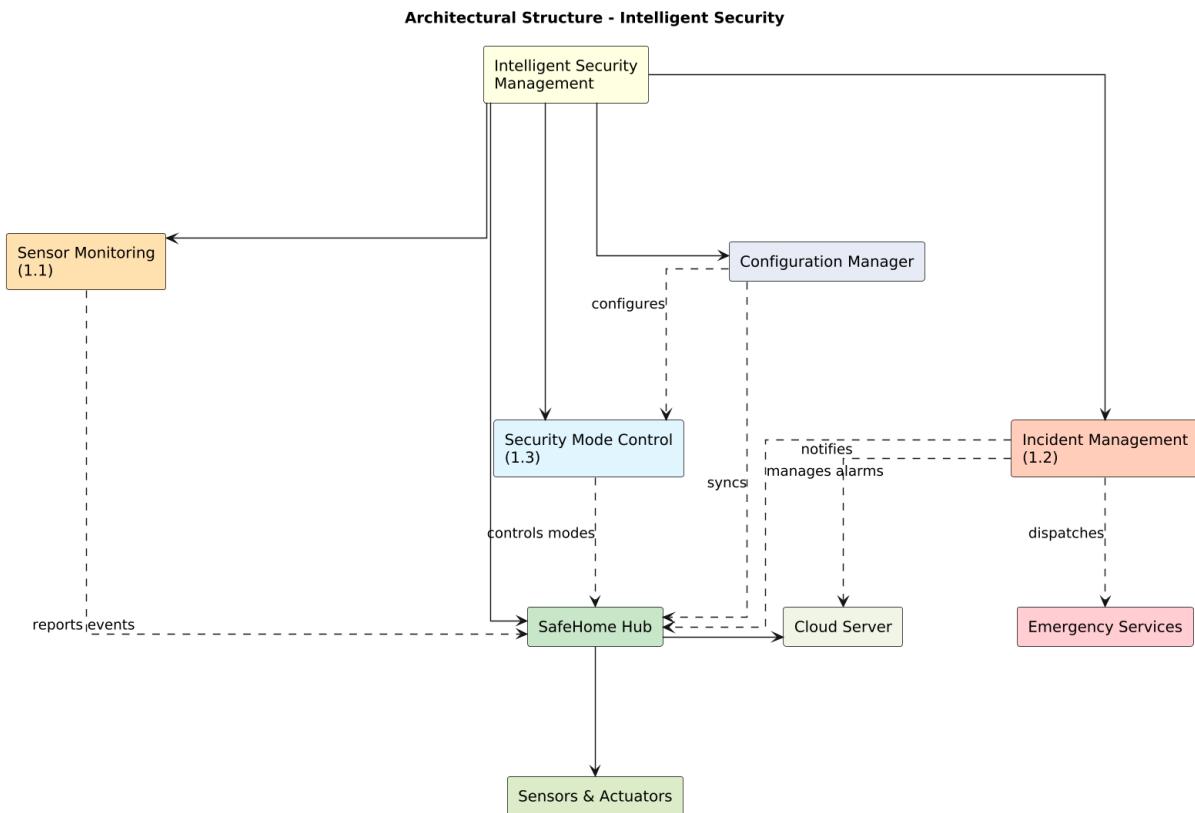
II. Architectural structure

1. OVERALL ARCHITECTURE



2. ARCHITECTURE – INTELLIGENT SECURITY

The number 1.1, 1.2, 1.3 in the diagram means it mainly covers use cases of “Intelligent security” part in the Team1 SRS document.

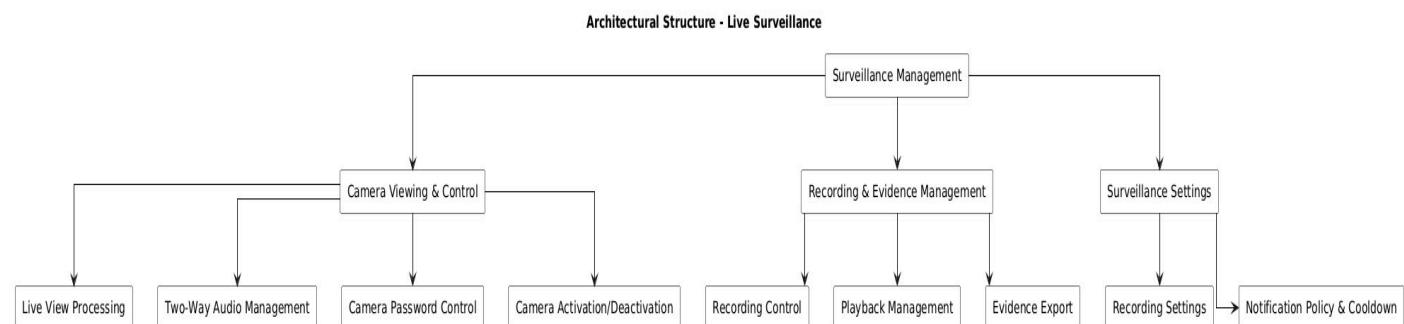


[Class diagram - Intelligent Security class diagram](#)

[Crc cards - Intelligent Security CRC cards](#)

[State diagram - Intelligent Security state diagram](#)

3. ARCHITECTURE – LIVE SURVEILLANCE

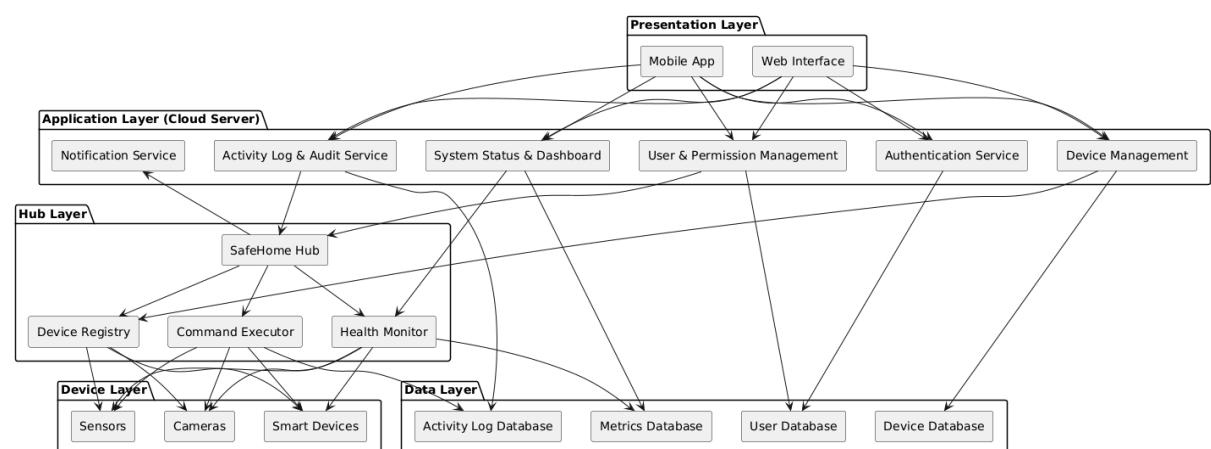


[Class diagram - Live surveillance class diagram](#)

[Crc cards - Live surveillance crc cards](#)

[State diagram - Live surveillance state diagram](#)

4. ARCHITECTURE – SYSTEM AND USER MANAGEMENT

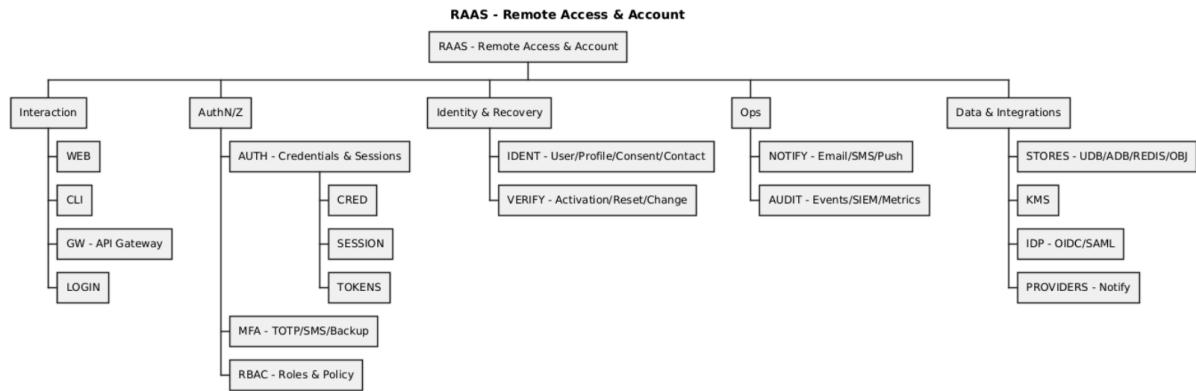


[Class diagram - System and User manager class diagram](#)

[Crc cards - System and User manager crc cards](#)

[State diagram - System and User manager state diagram](#)

5. ARCHITECTURE – REMOTE ACCESS AND ACCOUNT

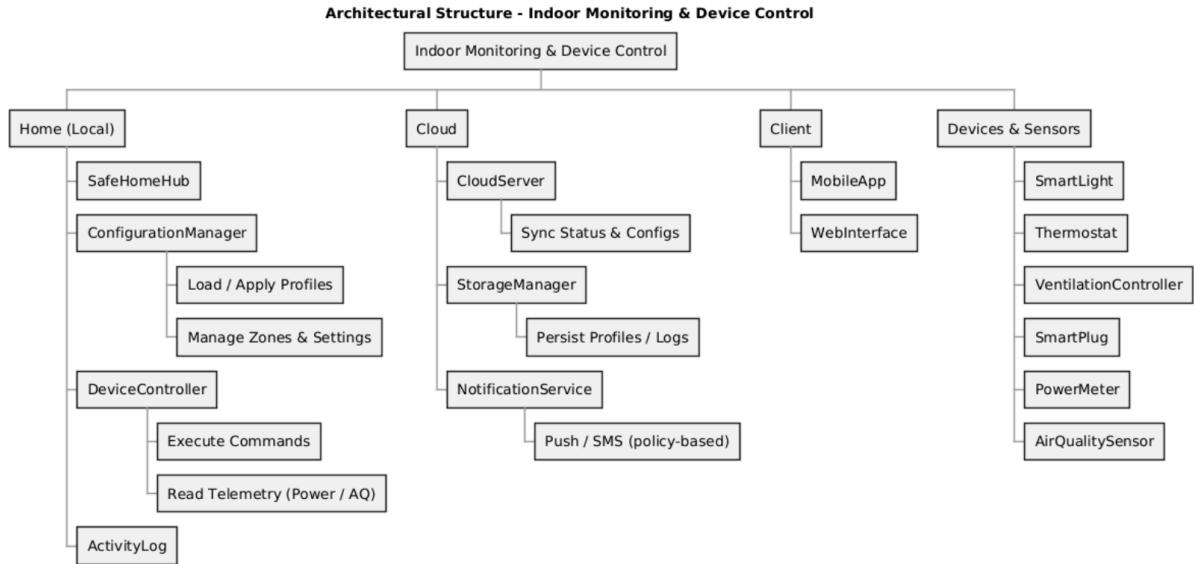


[Class diagram - Remote access and account class diagram](#)

[Crc cards - Remote access and account crc card](#)

[State diagram - Remote access and account state diagram](#)

6. ARCHITECTURE – INDOOR MONITORING AND DEVICE CONTROL



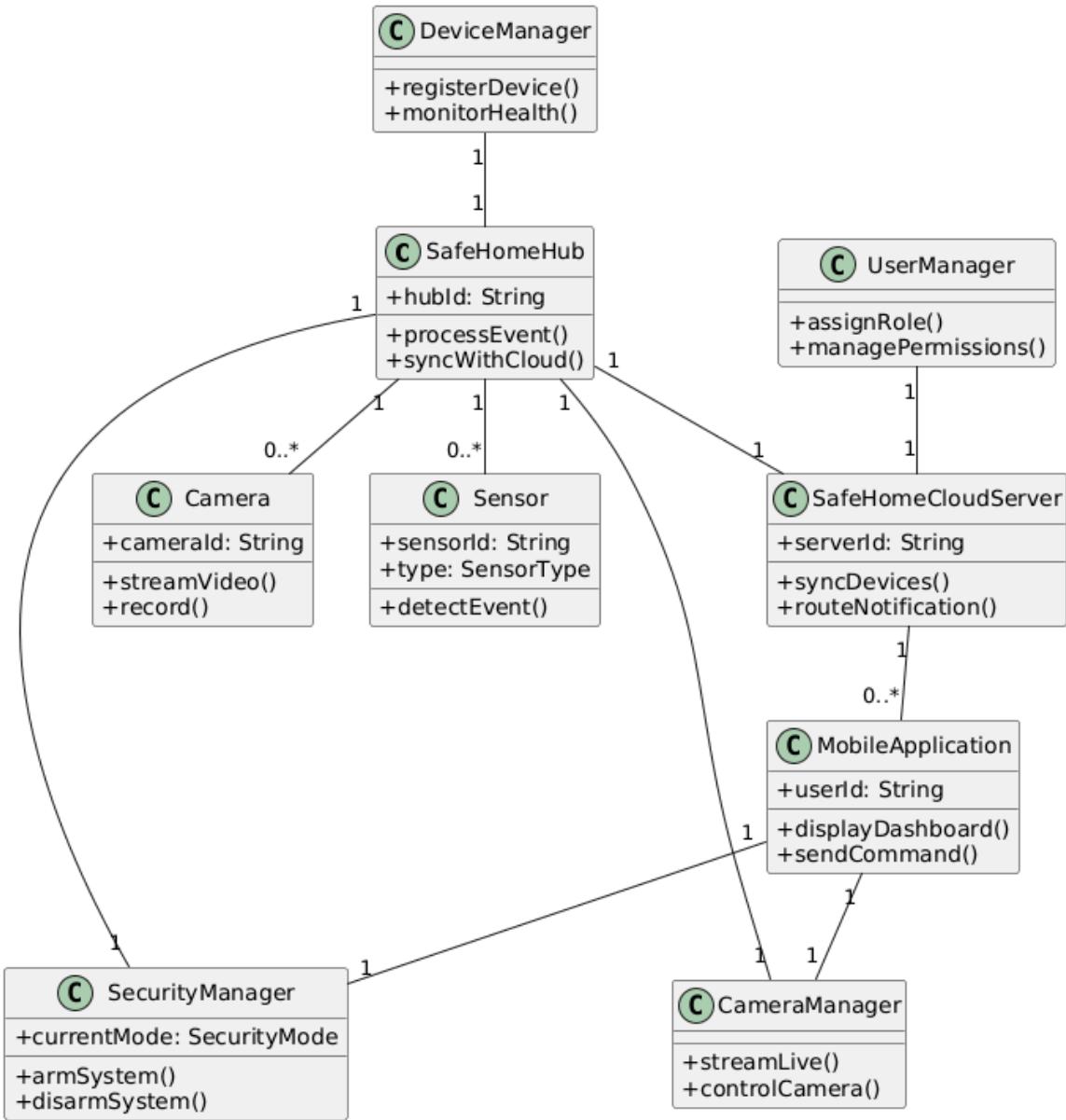
[Class diagram - Indoor monitoring and device control class diagram](#)

[Crc cards - Indoor monitoring and device control crc cards](#)

[State diagram - Indoor monitoring and device control state diagram](#)

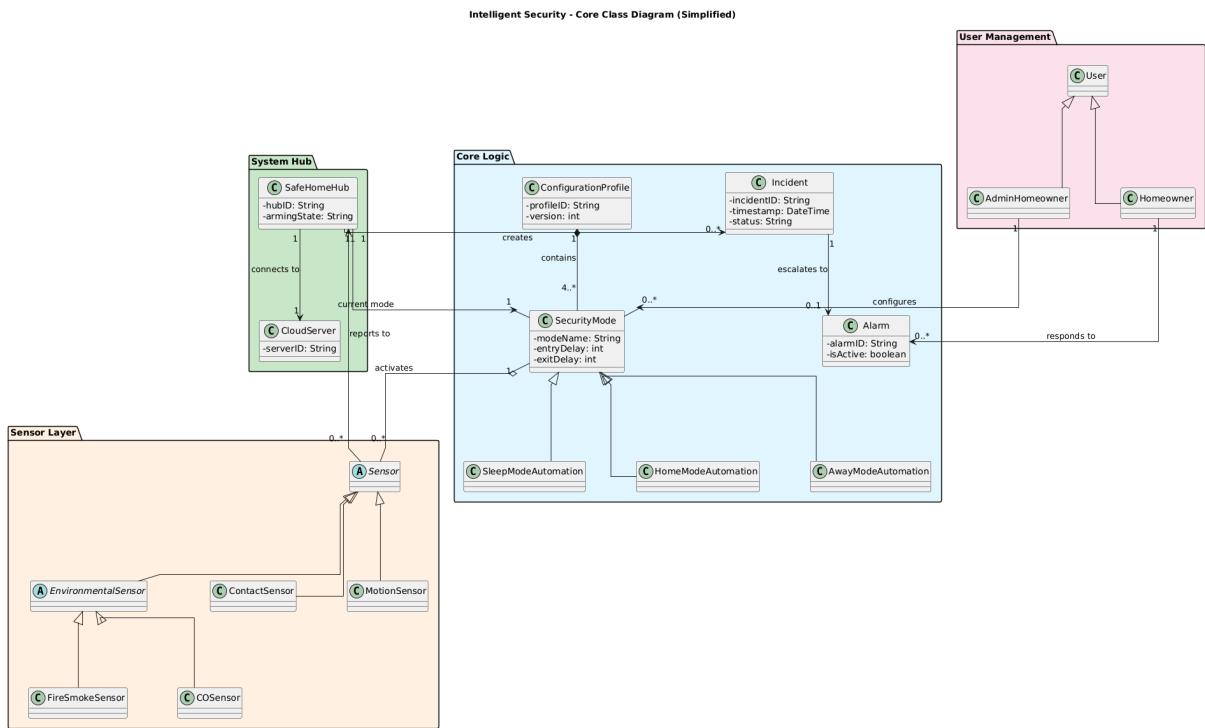
III. Class Diagram

1. CLASS DIAGRAM – WHOLE SYSTEM OVERVIEW

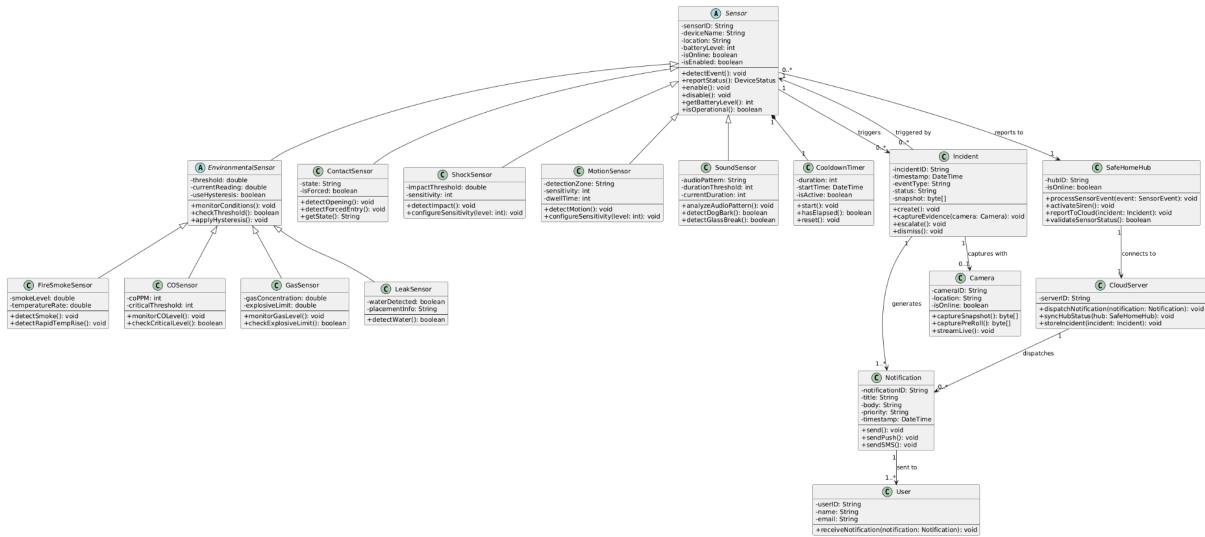


2. CLASS DIAGRAM – INTELLIGENT SECURITY

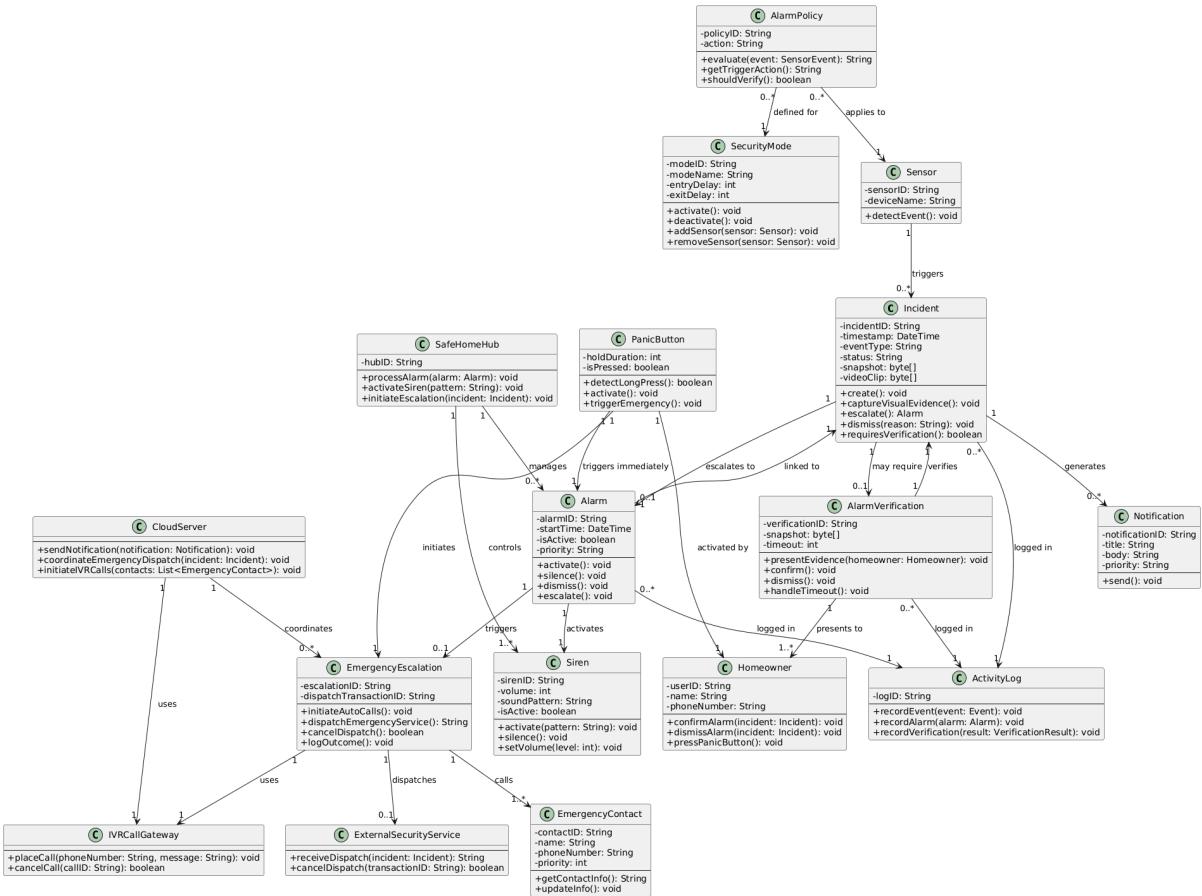
2.1 Class diagram - Intelligent security overall



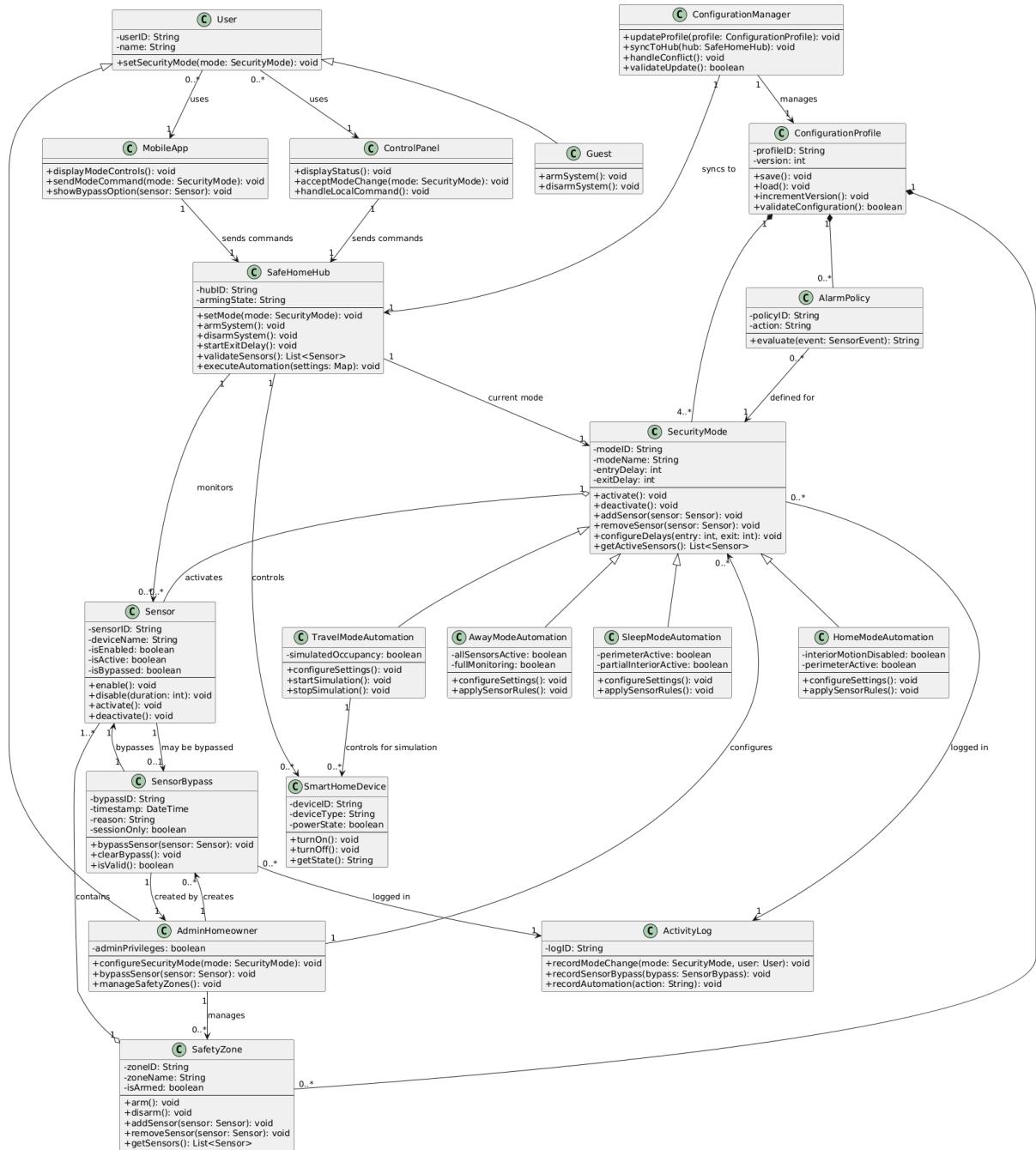
2.2 Class diagram - Sensor monitoring



2.3 Class diagram - Incident management

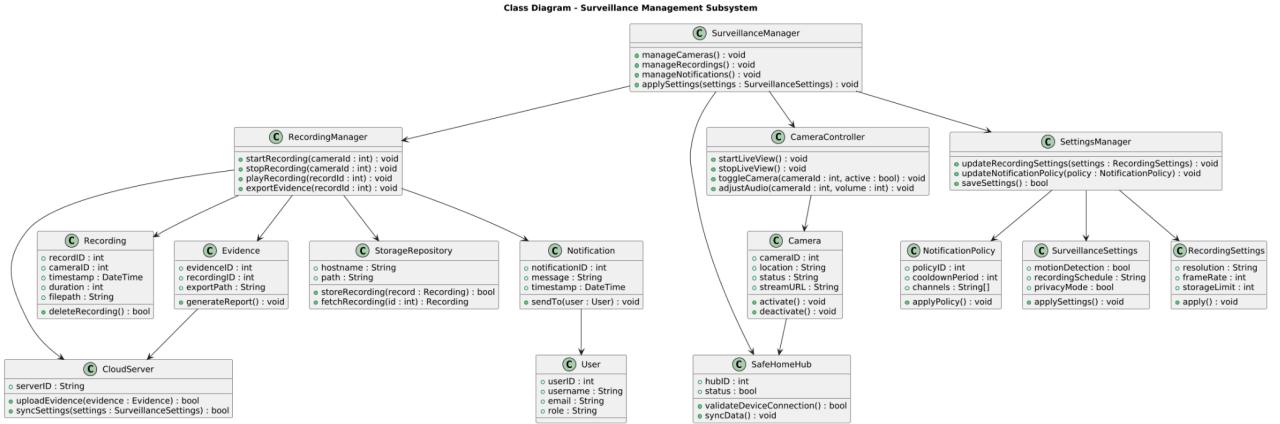


2.4 Class diagram - Security mode control



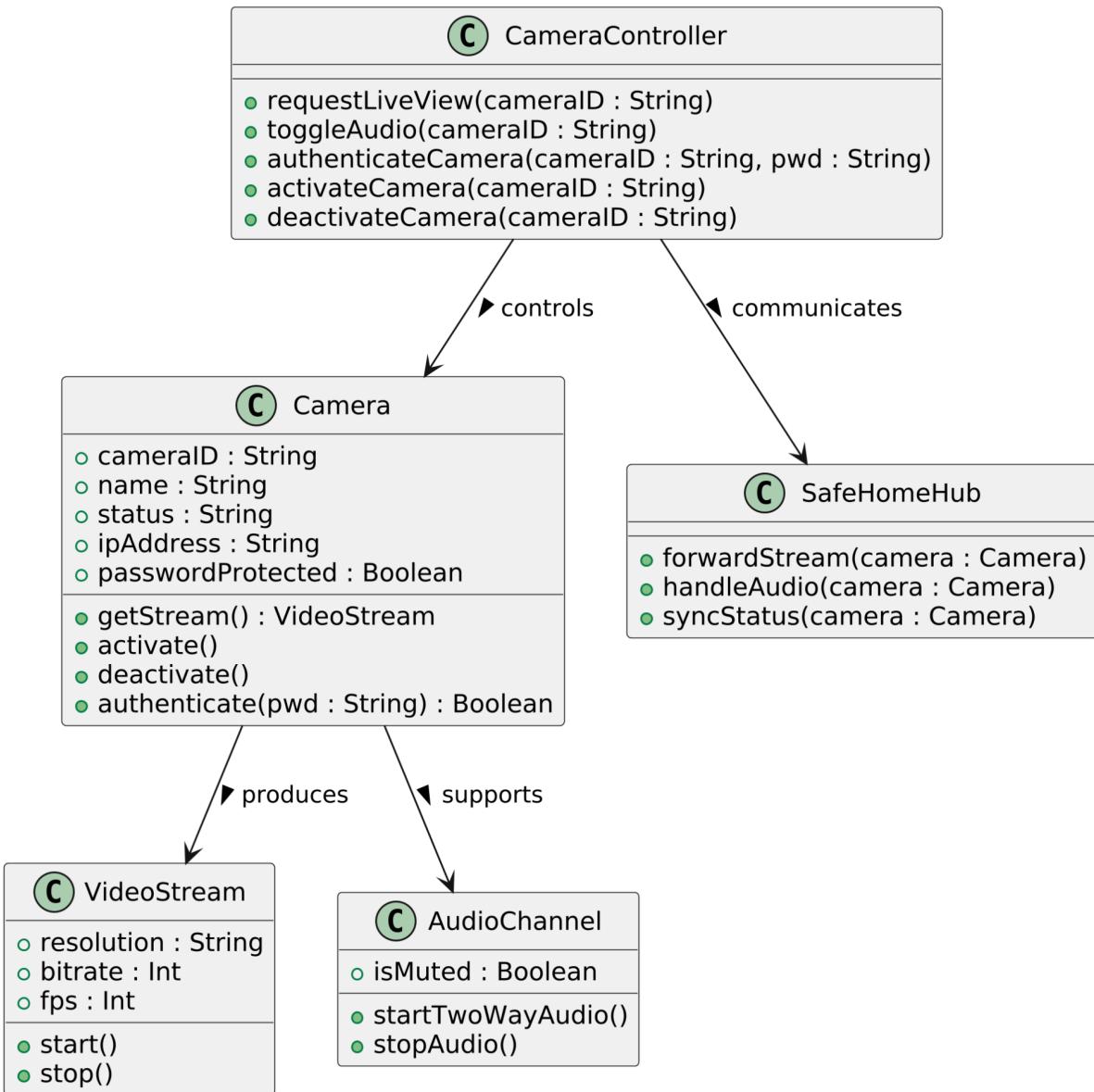
3. CLASS DIAGRAM – LIVE SURVEILLANCE

3.1 Surveillance overall

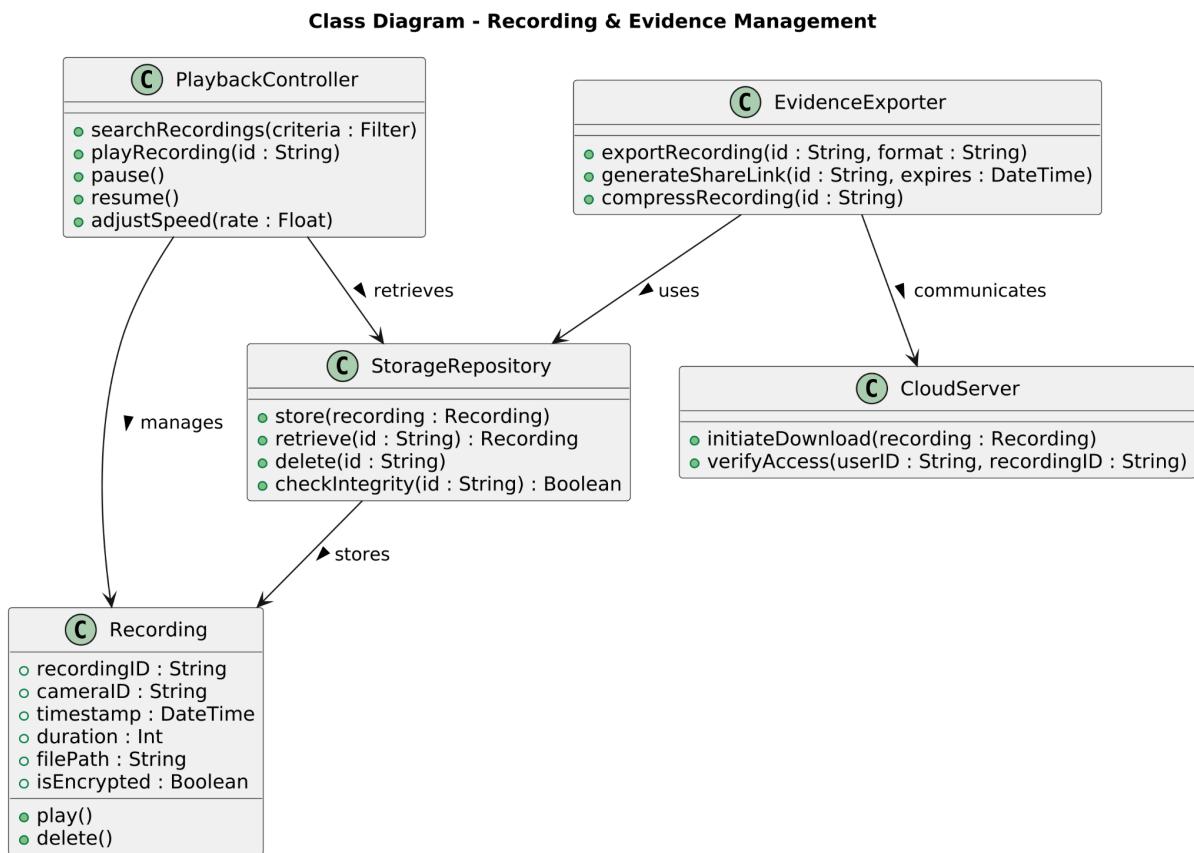


3.2 Camera Viewing and Control

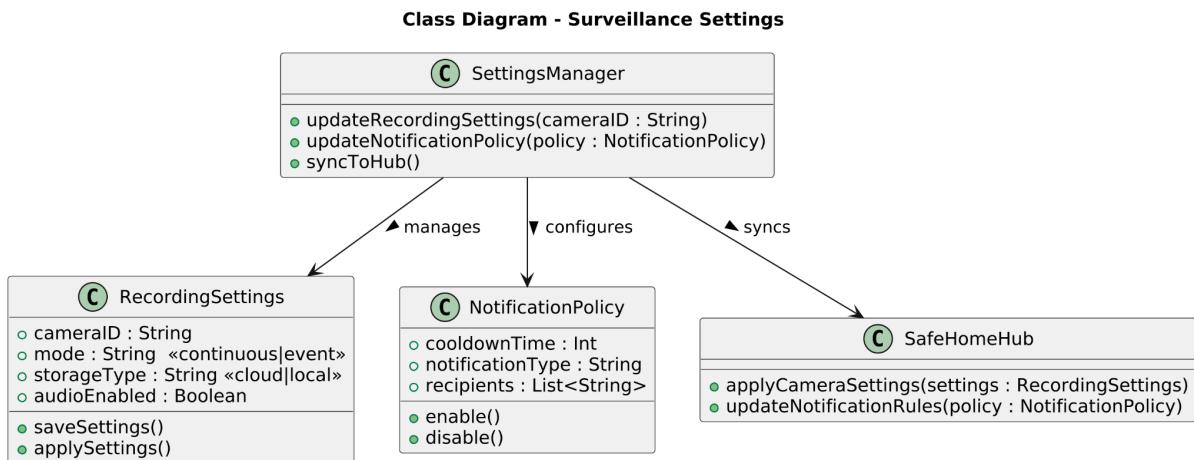
Class Diagram - Camera Viewing & Control



3.3 Recording and Evidence Management

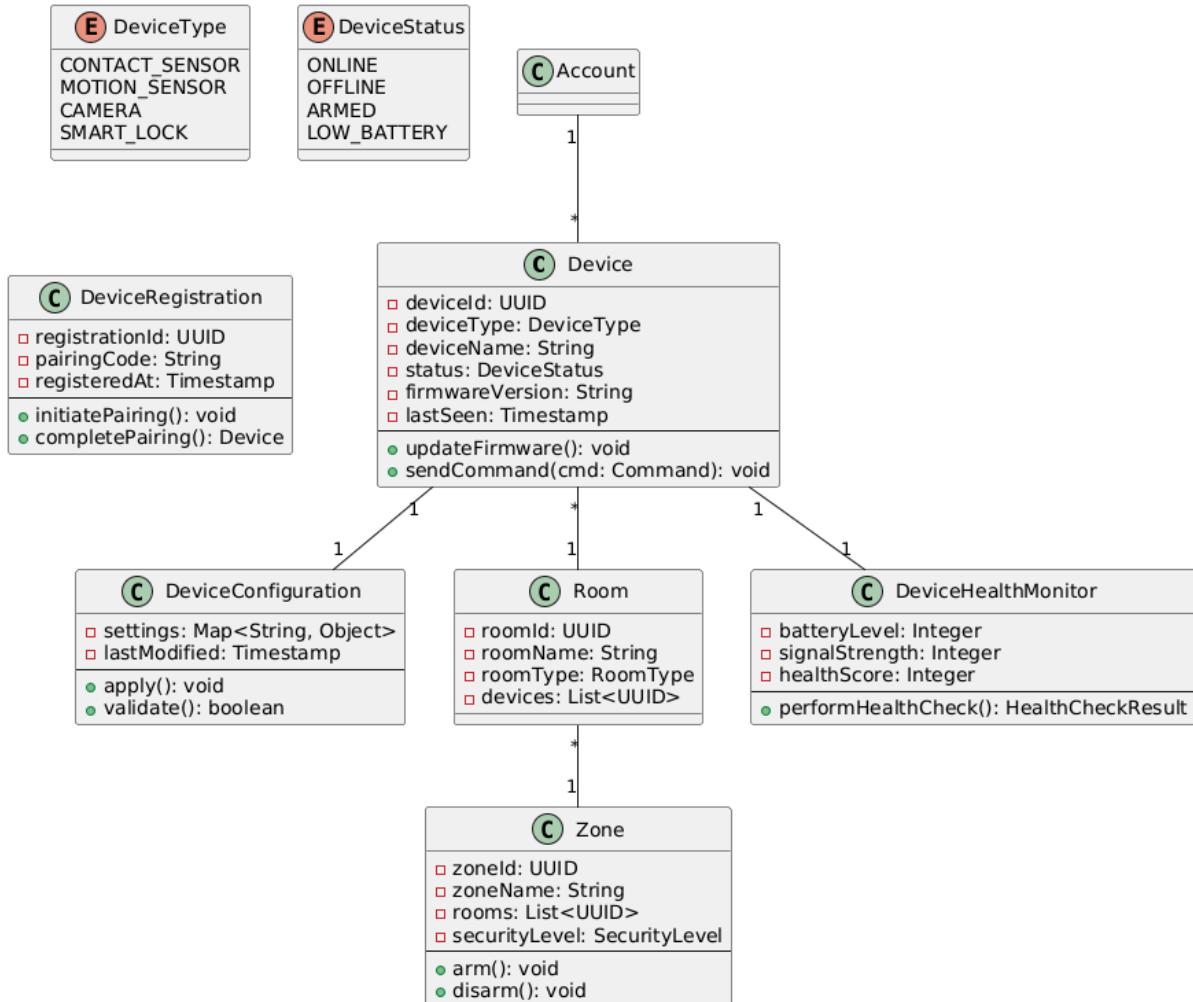


3.4 Surveillance settings

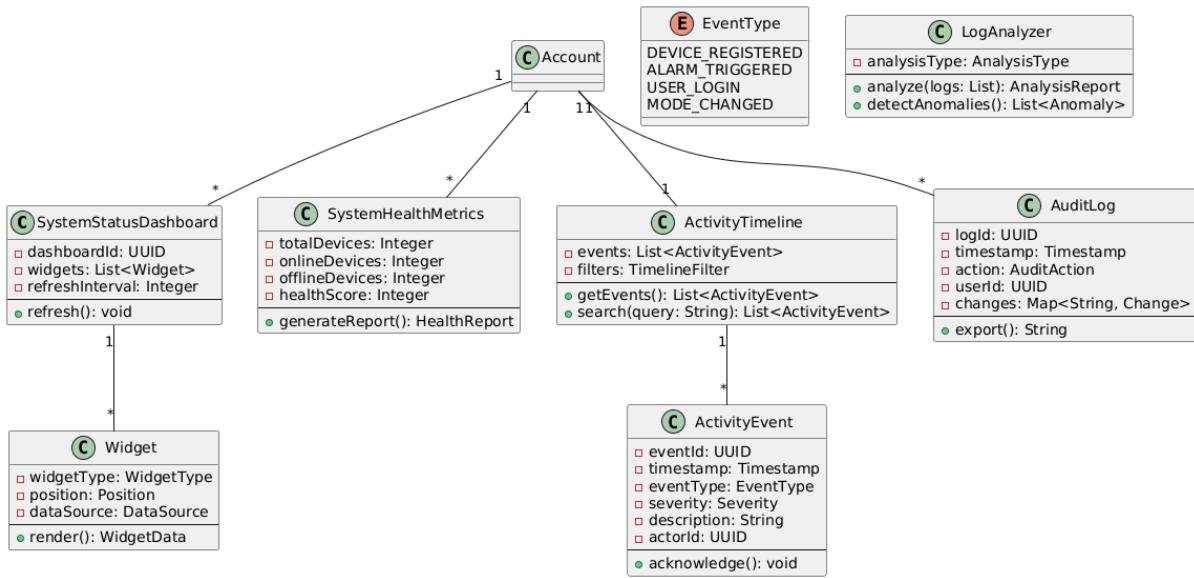


4. CLASS DIAGRAM – SYSTEM AND USER MANAGEMENT

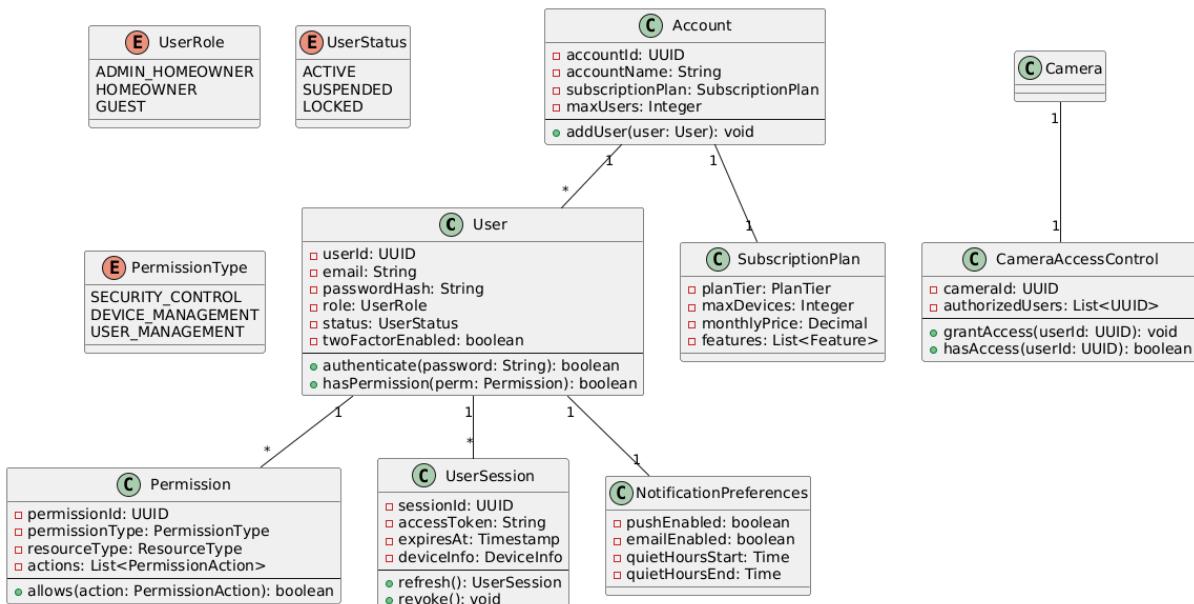
4.1 Device Management - Complete Class Diagram



4.2 System Status and Logs - Complete Class Diagram

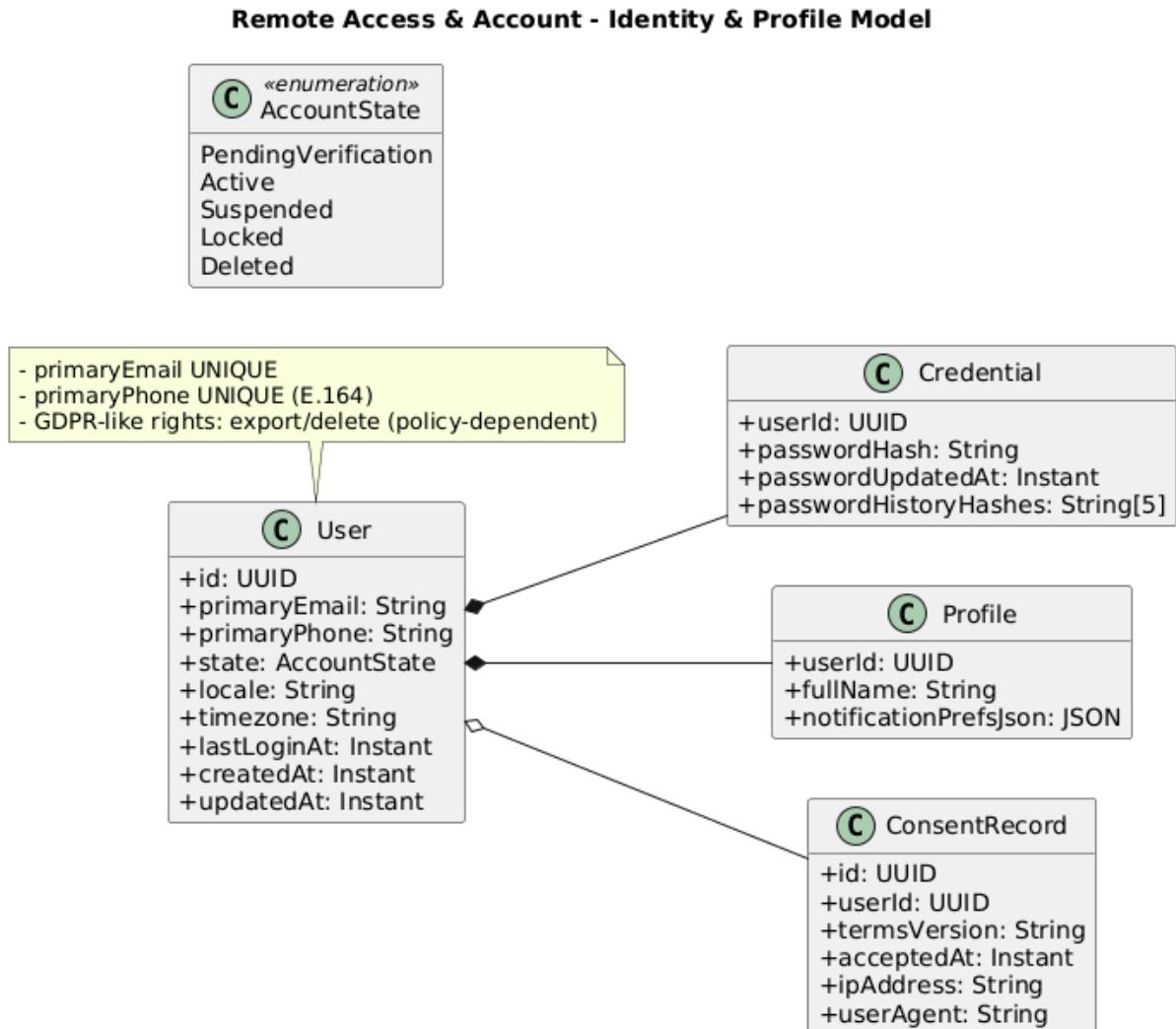


4.3 User and Permission Management - UML Class Diagram



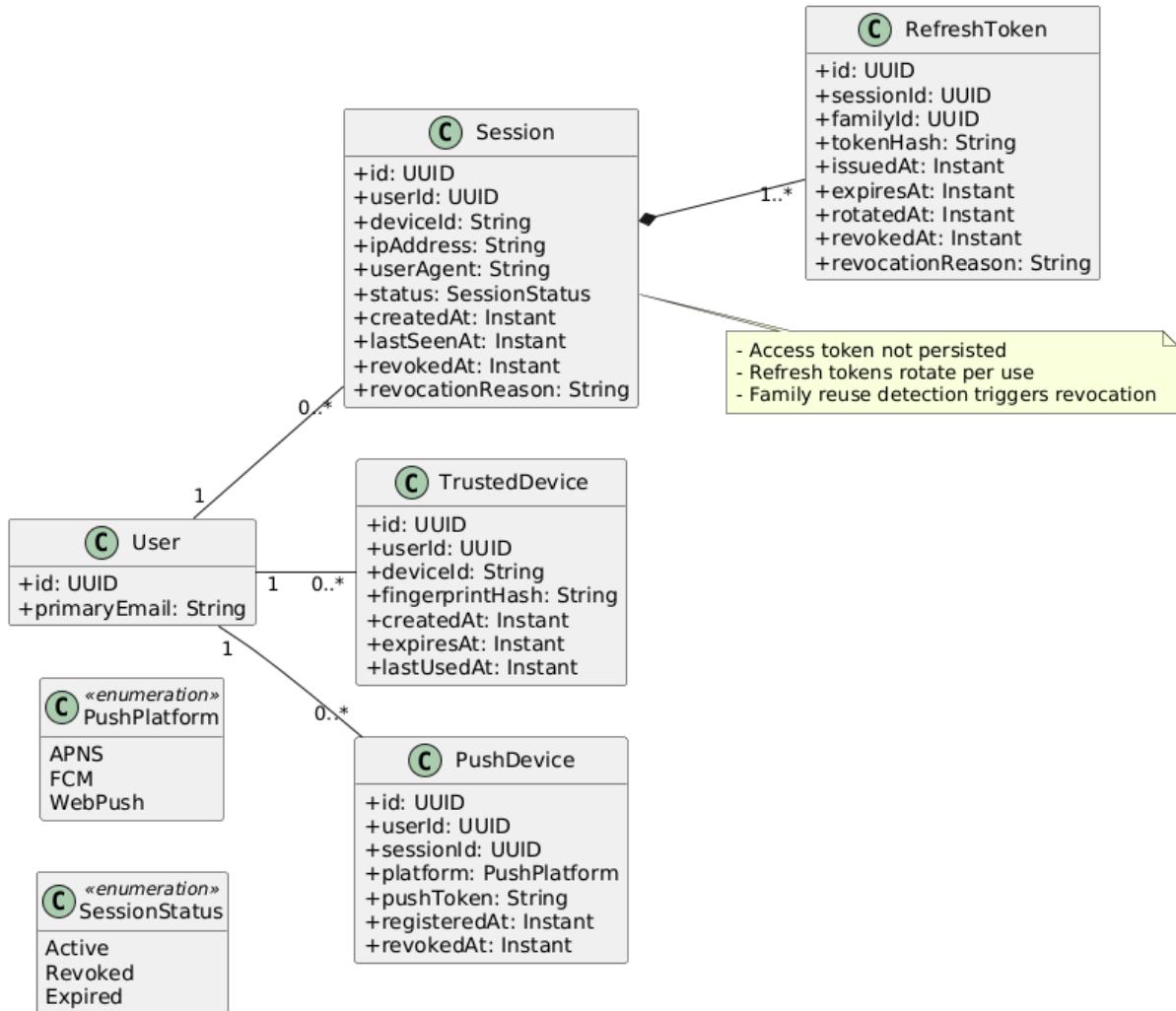
5. CLASS DIAGRAM – REMOTE ACCESS AND ACCOUNT

5.1 Identity & Profile Model



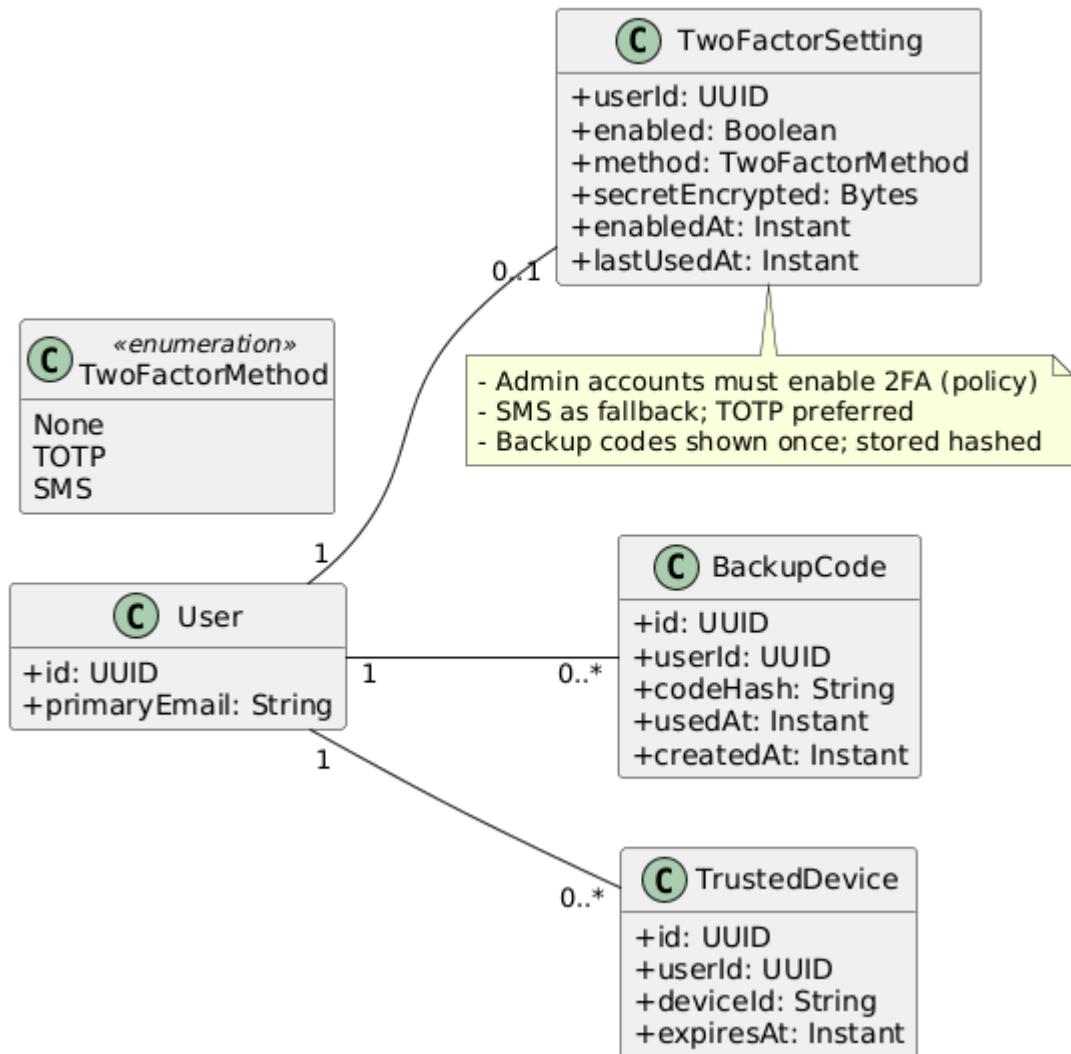
5.2 Authentication & Session Model

Remote Access & Account - Authentication & Session Model

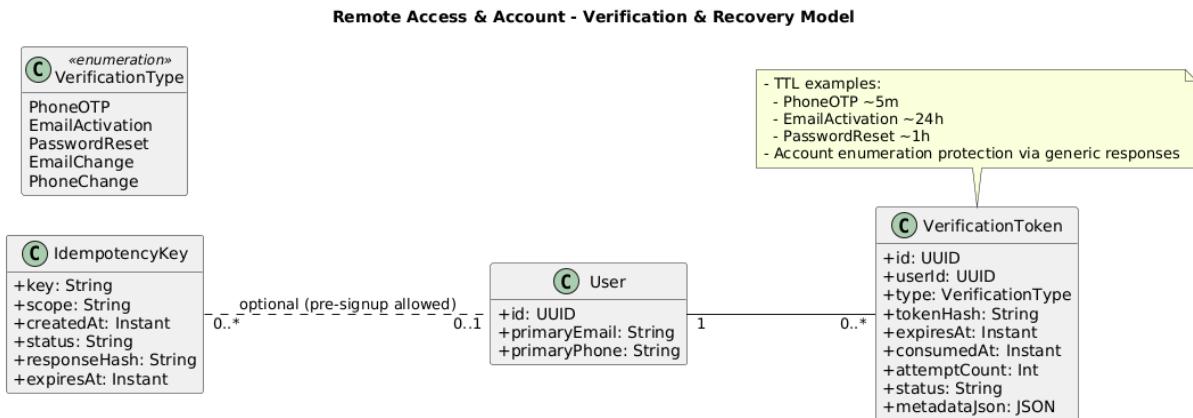


5.3 2FA Management Model

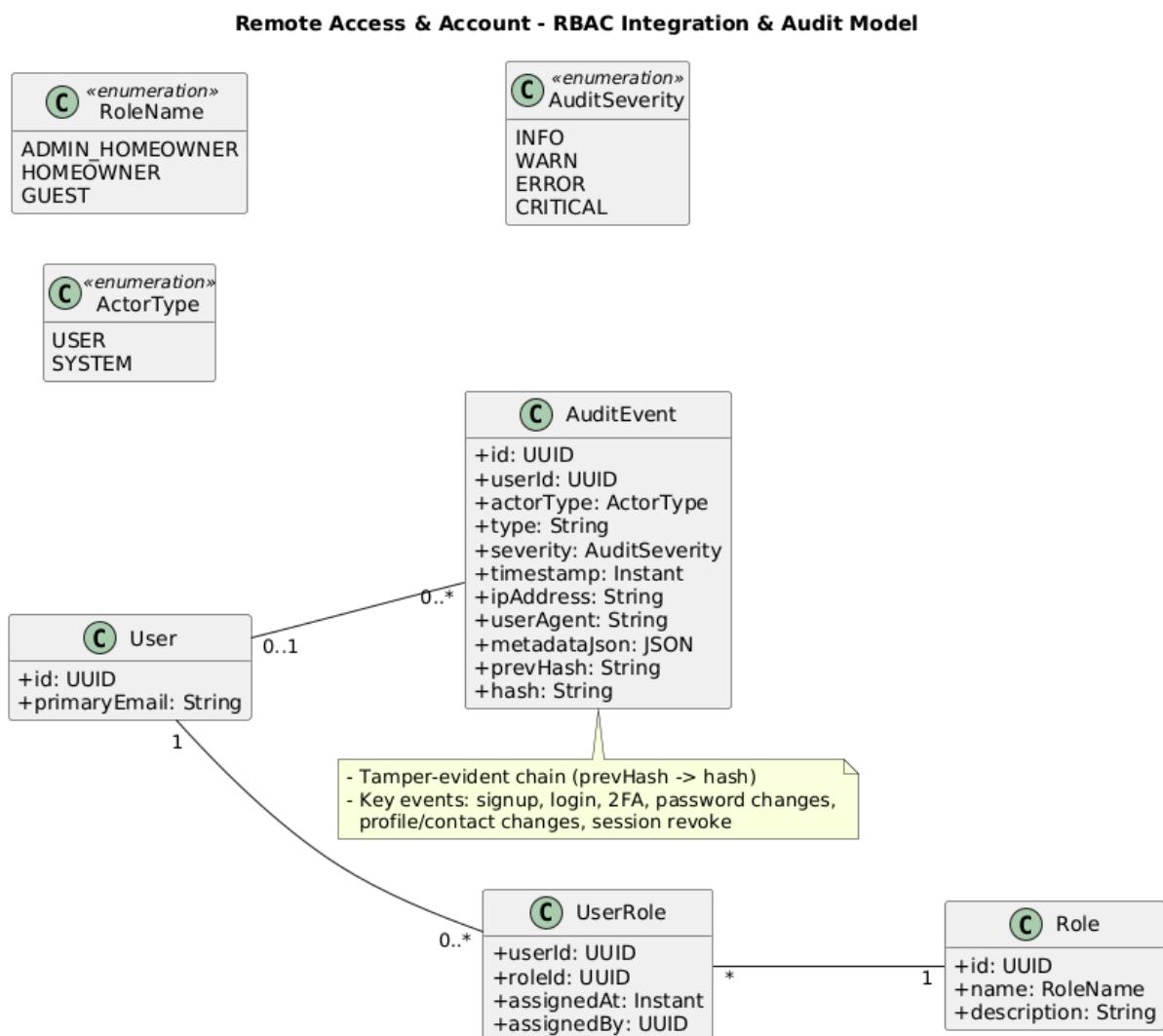
Remote Access & Account - 2FA Management Model



5.4 Verification & Recovery Model

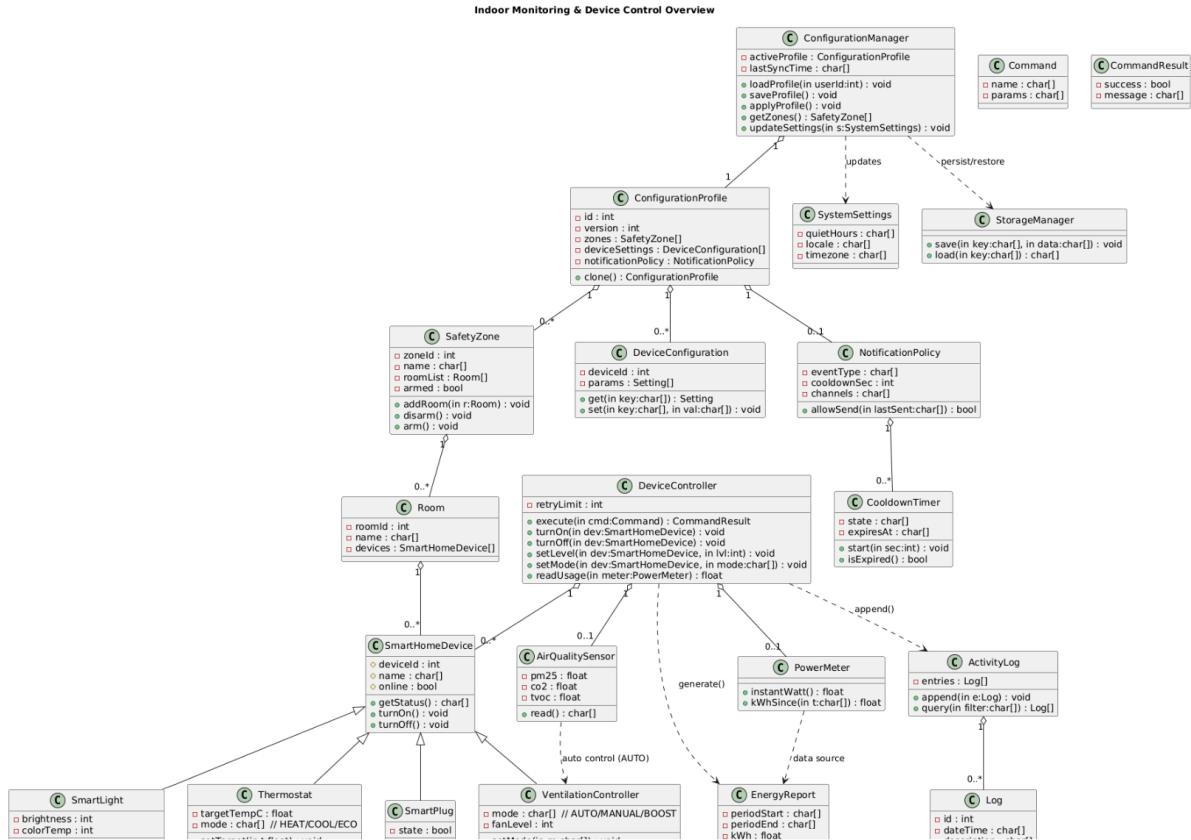


5.5 RBAC Integration & Audit Model

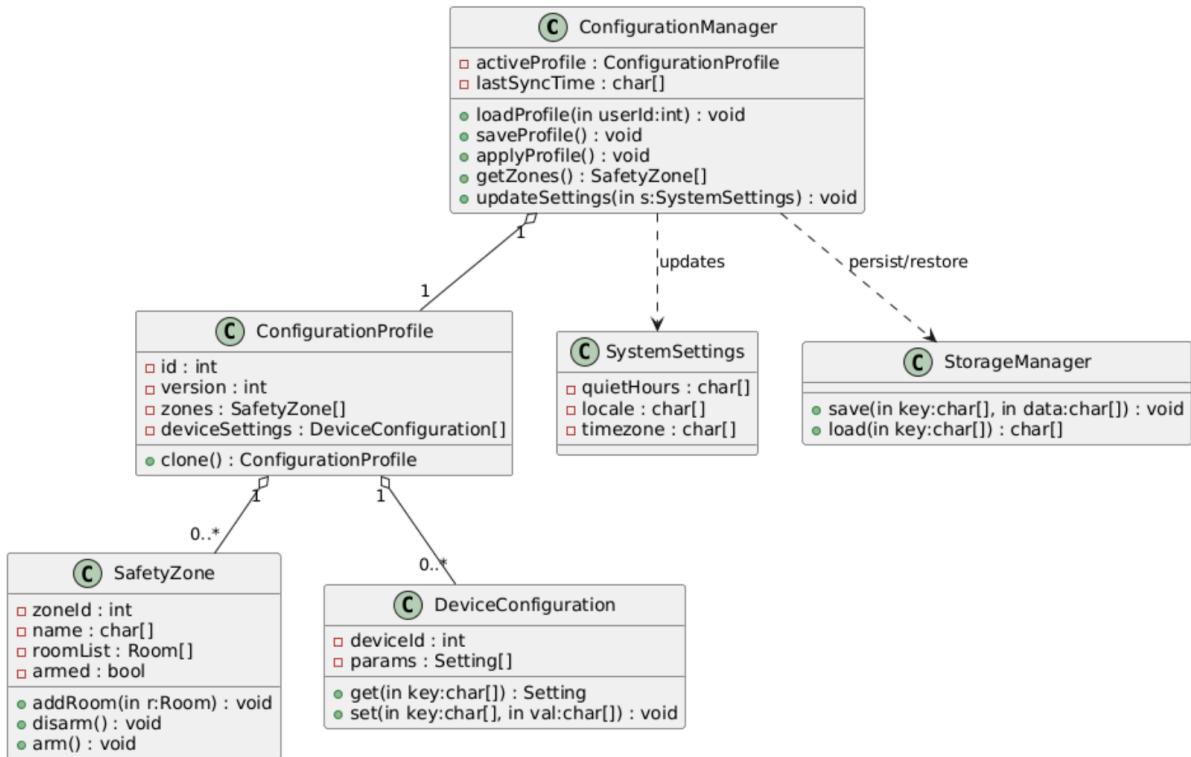


6. CLASS DIAGRAM – INDOOR MONITORING & DEVICE CONTROL

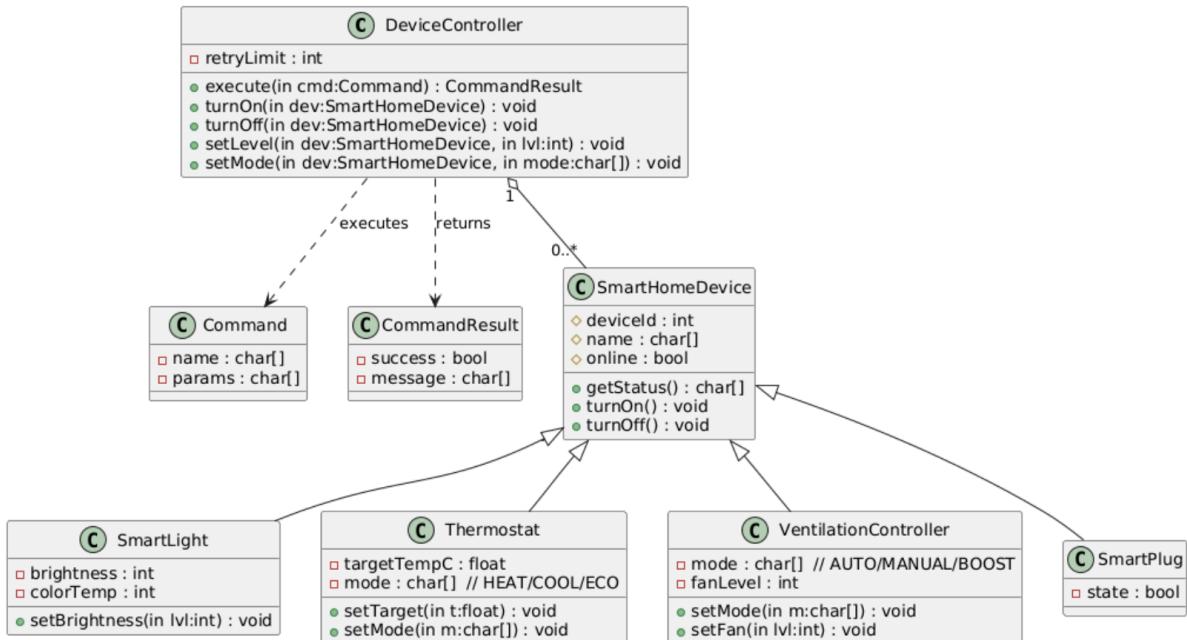
6.1 Class Diagram - Indoor Monitoring & Device Control Overall



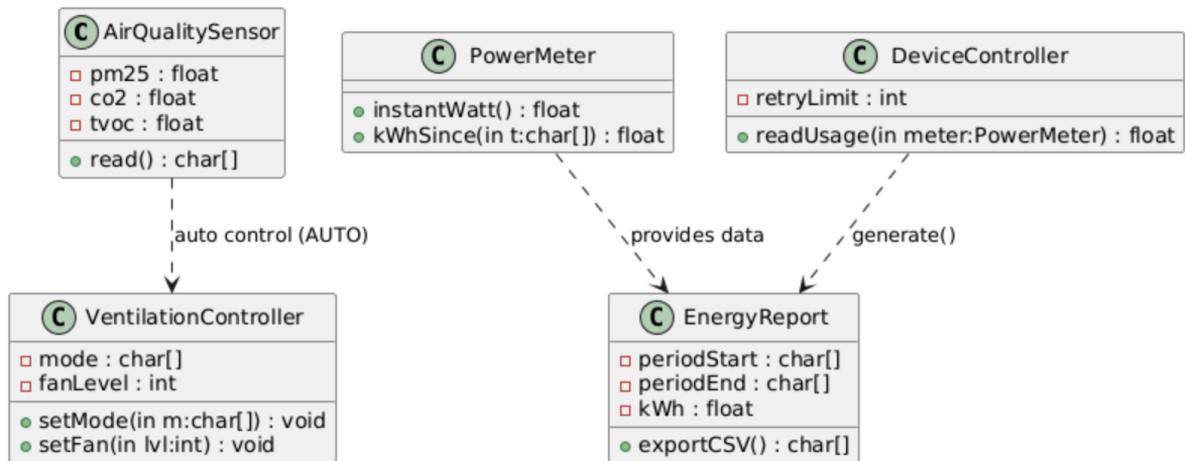
6.2 Class Diagram - Class Diagram - Configuration Core



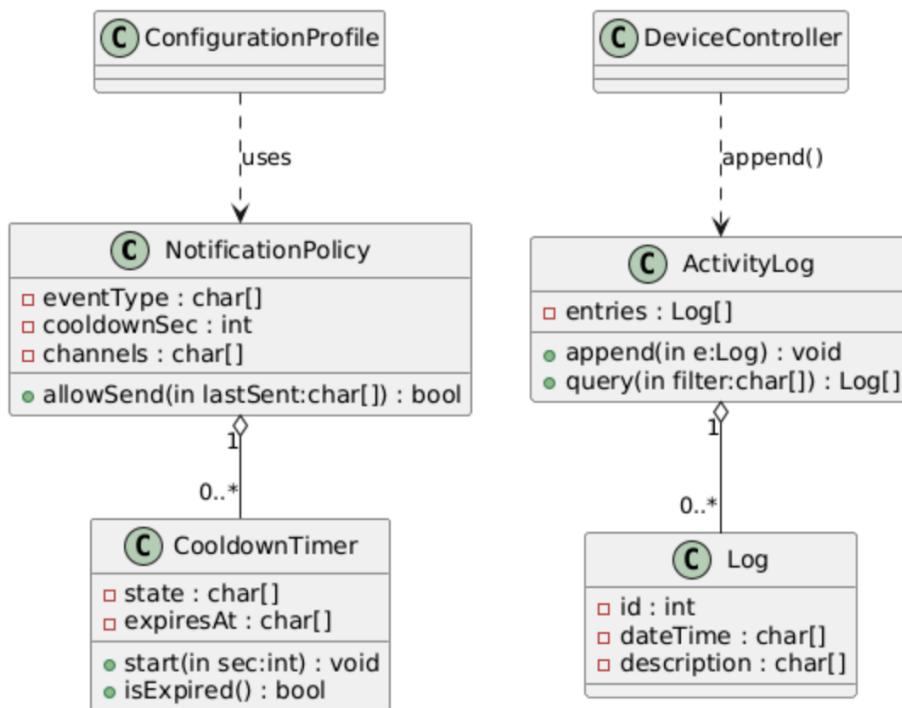
6.3 Class Diagram - Device Control Core



6.4 Class Diagram - Monitoring & Reports



6.5 Class Diagram - Policy, Logging & Notifications



IV. CRC cards

1. CRC CARDS – INTELLIGENT SECURITY

1.1 CRC CARD - User

Class: User	
Base class for all users interacting with the system	
Responsibilities	Collaborators
Store user ID, name, email, hashed password	AdminHomeowner
Provide login functionality	Homeowner
Provide logout functionality	Guest
Provide profile information	

[Ref: SRS Glossary p.129 (Homeowner), p.130 (Guest)]

1.2 CRC CARD - AdminHomeOwner

Class: AdminHomeowner	
Homeowner with full administrative privileges for system settings and user management	
Responsibilities	Collaborators
Manage user accounts	User
Configure system settings	Device
Add devices to system	SystemSettings
Set user permissions and roles	UserPermissionManager

[Ref: SRS Glossary p.129, UC 3.3.1 (User Role and Access Control)]

1.3 CRC CARD - Homeowner

Class: Homeowner

Primary system user with core functionality access (security settings, camera viewing) but without admin rights	
Responsibilities	Collaborators
View camera feeds	User
Set security modes	Camera
Dismiss alarm verifications	SecurityMode
Press panic button	AlarmVerification
	PanicButton

[Ref: SRS Glossary p.129, UC 1.1.1 (Primary Actor)]

1.4 CRC CARD - Guest

Class: Guest	
Temporary user with limited access permissions (e.g., arm/disarm only)	
Responsibilities	Collaborators
Arm system (if permitted)	User
Disarm system (if permitted)	SecurityMode
Store access time window	

[Ref: SRS Glossary p.130, UC 1.3.1 (Primary Actor)]

1.5 CRC CARD - ExternalSecurityService

Class: ExternalSecurityService (Actor)	
External monitoring service that receives automated incident reports	
Responsibilities	Collaborators
Receive dispatch requests	CloudServer
Process incident information	EmergencyEscalation
Return dispatch transaction ID	

[Ref: SRS UC 1.1.1 (Secondary Actors), UC 1.2.3 (Secondary Actors)]

1.6 CRC CARD - IVR/Call Gateway Service

Class: IVR/Call Gateway Service (Actor)	
External service for placing automated voice calls to homeowner or emergency contacts	
Responsibilities	Collaborators
Place automated voice calls	CloudServer
Play pre-recorded alarm messages	EmergencyContact
Handle call cancellation requests	EmergencyEscalation

[Ref: SRS UC 1.2.3 (Emergency Service Integration and Auto Call)]

1.7 CRC CARD - EmergencyContact

Class: EmergencyContact	
Data object containing emergency contact information for incident notifications	
Responsibilities	Collaborators
Store contact name and phone number	User
Store priority order	EmergencyEscalation
Provide contact info for auto-calls	IVR/CallGateway

[Ref: SRS UC 1.2.3 (Scenario 3)]

1.8 CRC CARD - SafeHomeHub

Here SafeHomeHub class represents a class which is related to Intelligent Security:

Class: SafeHomeHub	
Central control unit installed in the home that manages all devices locally and communicates with the cloud	
Responsibilities	Collaborators

Manage all connected devices	CloudServer
Process sensor events	Device
Activate siren	Sensor
Report to cloud	Siren
Sync configuration	ConfigurationManager
Execute security mode transitions	SecurityMode
Operate in offline mode	ActivityLog

[Ref: SRS Glossary p.126 (System Components (Hardware))]

1.9 CRC CARD - CloudServer

Class: CloudServer	
Responsibilities	Collaborators
Authenticate users	SafeHomeHub
Store user accounts	User
Sync hub status	MobileApp
Send notifications	WebInterface
Store activity logs	ActivityLog
Manage session tokens	SessionToken
Coordinate emergency dispatch	ExternalSecurityService
	IVR/CallGateway

[Ref: SRS Glossary p.127 (System Components (Software))]

1.10 CRC CARD - MobileApp

Class: MobileApp	

Primary user interface for iOS/Android platforms providing access to all system functions	
Responsibilities	Collaborators
Display user dashboard	User
Show live camera feeds	Camera
Present security controls	SecurityMode
Display notifications	Notification
Manage user session	SessionToken
Support panic button	PanicButton
	CloudServer

[Ref: SRS Glossary p.127 (System Components (Software)), III. Prototype UI/UX (Page 18-20)]

1.11 CRC CARD - Device

Class: Device (Interface)	
Common interface for all hardware devices connected to the hub	
Responsibilities	Collaborators
Provide device ID and name	SafeHomeHub
Report online/offline status	DeviceStatusManager
Report battery level	DeviceManager
Support device registration	

[Ref: SRS Glossary p.132 (Device Management)]

1.12 CRC CARD - Sensor

Class: Sensor (Abstract)	
Base class for all sensor types in the system. Provides common functionality for device status, communication, and event detection.	
Responsibilities	Collaborators

Detect events and trigger alerts	SafeHomeHub
Maintain connection status	Device
Report battery level	StorageManager
Enable/disable monitoring	ConfigurationManager
Store sensor configuration	SecurityMode
Transmit sensor data to hub	

[Ref: SRS Glossary p.126 (Sensor), UC 1.1 (Sensor Monitoring)]

1.13 CRC CARD - ContactSensor

Class: ContactSensor	
Detects opening/closing of doors and windows for physical intrusion detection	
Responsibilities	Collaborators
Monitor door/window state (open/closed)	Sensor
Detect forced opening events	SafeHomeHub
Report state changes to hub	Incident
Maintain closed/open status	

[Ref: SRS Glossary p.126, UC 1.1.1 (Trigger 1: Forced Opening)]

1.14 CRC CARD - ShockSensor

Class: ShockSensor	
Detects impacts and vibrations on doors/windows indicating forced entry attempts	
Responsibilities	Collaborators
Monitor vibration levels	Sensor

Detect impact exceeding threshold	SafeHomeHub
Configure vibration sensitivity	ConfigurationManager
Report shock events to hub	Incident

[Ref: SRS Glossary p.126, UC 1.1.1 (Trigger 3: Forced Impact)]

1.15 CRC CARD - MotionSensor

Class: MotionSensor	
Detects movement in designated areas for intrusion detection	
Responsibilities	Collaborators
Detect motion in coverage area	Sensor
Configure sensitivity settings	ConfigurationManager
Report motion events to hub	SafeHomeHub
Maintain detection zone information	Incident

[Ref: SRS Glossary p.126, UC 1.1.1 (Trigger 4: Unauthorized Motion), UC 1.1.3 (Outdoor Motion Detection)]

1.16 CRC CARD - EnvironmentalSensor

Class: EnvironmentalSensor (Abstract)	
Base class for environmental hazard sensors (fire, gas, CO, leak)	
Responsibilities	Collaborators
Monitor environmental conditions	Sensor
Detect hazard threshold violations	SafeHomeHub
Maintain threshold configuration	ConfigurationManager
Report critical hazard events	Incident

Use hysteresis to prevent rapid state changes	
---	--

[Ref: SRS Glossary p.126, UC 1.1.2 (Environmental Hazard Detection)]

1.17 CRC CARD - FireSmokeSensor

Class: FireSmokeSensor	
Responsibilities	Collaborators
Monitor smoke levels	EnvironmentalSensor
Detect rapid temperature increase	SafeHomeHub
Report fire events immediately	Incident
Trigger critical life-safety alarm	Alarm

[Ref: SRS UC 1.1.2 (Trigger 1: Fire/Smoke Sensor)]

1.18 CRC CARD - COSensor

Class: COSensor	
Detects dangerous carbon monoxide concentration levels	
Responsibilities	Collaborators
Monitor CO concentration (ppm)	EnvironmentalSensor
Detect critical CO threshold	SafeHomeHub
Report CO hazard events	Incident
Trigger critical life-safety alarm	Alarm

[Ref: SRS UC 1.1.2 (Trigger 2: CO Sensor)]

1.19 CRC CARD - GasSensor

Class: GasSensor	
Detects gas leaks exceeding explosive limit thresholds	

Responsibilities	Collaborators
Monitor gas concentration	EnvironmentalSensor
Detect explosive limit threshold	SafeHomeHub
Report gas leak events	Incident
Trigger critical life-safety alarm	Alarm

[Ref: SRS UC 1.1.2 (Trigger 3: Gas Sensor)]

1.20 CRC CARD - LeakSensor

Class: LeakSensor	
Detects water leaks for property protection	
Responsibilities	Collaborators
Detect water presence	EnvironmentalSensor
Report leak events to hub	SafeHomeHub
Maintain sensor placement info	Incident

[Ref: SRS UC 1.1.2 (Trigger 4: Leak Sensor)]

1.21 CRC CARD - SoundSensor

Class: SoundSensor	
Detects specific audio patterns like dog barking or glass breaking	
Responsibilities	Collaborators
Analyze audio patterns	Sensor
Detect dog barking signature	SafeHomeHub
Configure duration thresholds	ConfigurationManager
Report sound events	Incident
Implement cooldown timer	CooldownTimer

[Ref: SRS Glossary p.126, UC 1.1.4 (Dog Barking Detection)]

1.22 CRC CARD - Camera

Class: Camera	
IP camera providing live streaming, recording, and two-way audio	
Responsibilities	Collaborators
Stream live video feed	Device
Capture snapshots and video clips	SafeHomeHub
Support two-way audio	Homeowner
Start/stop recording	RecordingManager
Enable/disable camera (privacy mode)	ConfigurationManager
Support password protection	CameraAccessManager
Detect motion events	

[Ref: SRS Glossary p.127 (Camera (IP Camera)), UC 2.1 (Camera Viewing and Control)]

1.23 CRC CARD - Siren

Class: Siren	
Physical alarm device that sounds alerts	
Responsibilities	Collaborators
Activate audible alarm	Device
Configure volume and sound pattern	ConfigurationManager
Silence alarm on command	SafeHomeHub
Support distinct patterns for different hazards	Incident
Maintain activation state	

[Ref: SRS Glossary p.128 (Alarm Condition), UC 1.1.1 (Scenario 3)]

1.24 CRC CARD - SmartDoorLock

Class: SmartDoorLock	
Smart lock with password entry and tamper detection	
Responsibilities	Collaborators
Lock/unlock door	Device
Validate entered passwords	SafeHomeHub
Track failed password attempts	Incident
Report tampering after 10 consecutive failures	Alarm
Maintain locked/unlocked state	

[Ref: SRS UC 1.1.1 (Trigger 2: Lock Tampering)]

1.25 CRC CARD - SmarHomeDevice

Class: SmartHomeDevice	
Base class for controllable smart home devices (lights, valves, etc.)	
Responsibilities	Collaborators
Store power state (on/off)	Device
Turn on/off	DeviceController
Report current state	SafeHomeHub
Support automation integration	SecurityMode

[Ref: SRS Glossary p.127 (Smart Home Device), UC 5.1.1 (Indoor Device Control)]

1.26 CRC CARD - SecurityMode

Class: SecurityMode

Defines sensor activation rules for Home, Away, Sleep, and other modes	
Responsibilities	Collaborators
Store mode name and ID	ConfigurationManager
Maintain list of active sensors per mode	Sensor
Define entry/exit delay durations	SafeHomeHub
Store mode in configuration profile	ConfigurationProfile
Add/remove sensors from mode	

[Ref: SRS Glossary p.128 (Security Mode), UC 1.3.1 (One-Touch Modes)]

1.27 CRC CARD - Alarm

Class: Alarm	
Represents an active alarm event in the system	
Responsibilities	
Store alarm ID and start time	Incident
Maintain alarm active status	Siren
Link to triggering incident	SafeHomeHub
Support silence/dismiss operations	AlarmVerification
Record alarm resolution	

[Ref: SRS Glossary p.128 (Alarm Condition)]

1.28 CRC CARD - Incident

Class: Incident

Represents a detected security or hazard event requiring processing	
Responsibilities	Collaborators
Store incident ID and timestamp	Sensor
Record triggering sensor event	Alarm
Capture visual evidence (snapshot)	Camera
Maintain status (PENDING, VERIFIED, DISMISSED)	AlarmVerification
Link to alarm if escalated	SafeHomeHub
Store in activity log	ActivityLog

[Ref: SRS Glossary p.130 (Alarm Verification), UC 1.2.2 (Alarm Verification Step)]

1.29 CRC CARD - User

Class: AlarmVerification	
Manages the verification workflow before full alarm escalation	
Responsibilities	Collaborators
Present visual evidence to homeowner	Incident
Provide "Confirm" and "Dismiss" options	Camera
Manage verification countdown timer	Homeowner
Escalate to full alarm if timeout	SafeHomeHub
Log homeowner decision	ActivityLog

[Ref: SRS Glossary p.129 (Homeowner), p.130 (Guest), p.129 (Admin Homeowner)]

1.30 CRC CARD - ActivityLog

Class: ActivityLog	
Records all system events, user actions, and incidents	
Responsibilities	Collaborators
Store chronological event log	Log
Record user actions with identity	User
Record sensor events	Sensor
Record mode changes	SecurityMode
Store alarm events	Alarm
Support filtering and search	Homeowner
Export logs in PDF/CSV format	EvidenceExport
Persist to storage	StorageManager

[Ref: SRS Glossary p.131 (Activity Log / Timeline), UC 3.2.2 (Activity Logs and Timeline)]

1.31 CRC CARD - Notification

Class: Notification	
Push notification or SMS alert to user	
Responsibilities	Collaborators
Store notification title and body	Incident
Store priority level	CloudServer
Store target user(s)	User
Support push notification	MobileApp
Support SMS fallback	SMSGateway

[Ref: SRS Glossary p.133 (Push Notification), UC 1.1.1 (Scenario 4)]

1.32 CRC CARD - ConfigurationProfile

Class: ConfigurationProfile	
Complete system configuration for a user account	
Responsibilities	Collaborators
Store profile ID and version	User
Contain all security modes	SecurityMode
Contain all safety zones	SafetyZone
Contain alarm policies	AlarmPolicy
Contain notification policies	NotificationPolicy
Contain device settings	Device
Support save/load operations	StorageManager

[Ref: SRS UC 1.2.1 (Configure Alarm Conditions...)]

1.33 CRC CARD - AlarmPolicy

Class: AlarmPolicy	
Defines what action to take when specific sensors trigger in specific modes	
Responsibilities	Collaborators
Map mode + sensor to action	SecurityMode
Define action: TRIGGER_ALARM, TRIGGER_VERIFICATION, LOG_ONLY	Sensor
Evaluate event against policy	IncidentManager
Store in configuration profile	ConfigurationProfile

[Ref: SRS UC 1.2.1 (Configure Alarm Conditions...)]

1.34 CRC CARD - NotificationPolicy

Class: NotificationPolicy	
Defines cooldown periods and alert frequency for event types	
Responsibilities	Collaborators
Store event type and cooldown duration	ConfigurationProfile
Determine if notification should be sent	CooldownTimer
Prevent notification fatigue	SafeHomeHub

[Ref: SRS UC 2.3.2 (Notification Policy and Cooldown)]

1.35 CRC CARD - CooldownTimer

Class: CooldownTimer	
Manages cooldown timers to prevent notification/alert storms	
Responsibilities	Collaborators
Track timer state per event type/camera	NotificationPolicy
Start timer after notification sent	Incident
Check if cooldown period has elapsed	IncidentManager
Reset timer when needed	

[Ref: SRS Glossary p.131 (Cooldown Period), UC 1.1.3 (Trigger)]

1.36 CRC CARD - PanicButton

Class: PanicButton

Emergency button feature for manual alarm activation	
Responsibilities	Collaborators
Detect 3-second press-and-hold	MobileApp
Bypass all verification steps	Homeowner
Trigger highest-priority alarm	SafeHomeHub
Activate siren immediately	Siren
Initiate emergency escalation	EmergencyEscalation
Log panic event with user identity	ActivityLog

[Ref: SRS Glossary p.129 (Panic Button), UC 1.2.4 (Panic Button)]

2. CRC CARDS – LIVE SURVEILLANCE

2.1 CRC CARD - Camera

Class: Camera	
Represents a physical surveillance camera device in the system.	
Responsibilities	Collaborators
Store camera identity, status, and IP information.	VideoStream
Authenticate access when password protected.	AudioChannel
Provide live video and audio stream interfaces.	SafeHomeHub
Handle activation/deactivation of the device.	CameraController

2.2 CRC CARD - VideoStream

Class: VideoStream	
Manages the live video feed produced by a camera.	
Responsibilities	Collaborators
Maintain streaming parameters (resolution, bitrate, fps).	Camera
Start and stop live streaming sessions.	CameraController
Provide encoded video data for UI or cloud forwarding.	

2.3 CRC CARD - AudioChannel

Class: AudioChannel	
Handles two-way audio communication between the user and the camera.	
Responsibilities	Collaborators
Enable two-way audio communication.	Camera
Handle mute/unmute operations.	CameraController
Support audio streaming for connected camera sessions.	SafeHomeHub

2.4 CRC CARD - CameraController

Class: CameraController	
Coordinates all camera operations and user commands.	
Responsibilities	Collaborators
Handle user requests to view, activate, or deactivate cameras.	Camera
Authenticate camera connections.	VideoStream

Coordinate live stream and audio control.	AudioChannel
Communicate with hub for synchronization.	SafeHomeHub

2.5 CRC CARD - SafeHomeHub

Explain some context:

Class: SafeHomeHub	
Responsibilities	Collaborators
Handle user requests to view, activate, or deactivate cameras.	Camera
Authenticate camera connections.	VideoStream
Coordinate live stream and audio control.	AudioChannel
Communicate with hub for synchronization.	SafeHomeHub

2.6 CRC CARD - Recording

Class: Recording	
Responsibilities	Collaborators
Represents a recorded video clip stored locally or in the cloud.	
Store recording metadata (camera, time, duration, path).	PlaybackController
Manage playback and deletion of recordings.	StorageRepository
Ensure integrity and encryption if required.	

2.7 CRC CARD - PlaybackController

Class: PlaybackController	
Controls playback and navigation of recorded videos.	
Responsibilities	Collaborators
Manage playback operations (play, pause, resume, adjust speed).	Recording
Search and retrieve recordings from repository.	StorageRepository
Interface with UI for playback control.	EvidenceExporter

2.8 CRC CARD - EvidenceExporter

Class: EvidenceExporter	
Handles secure export and sharing of recorded evidence.	
Responsibilities	Collaborators
Export selected recordings in specified formats.	StorageRepository
Generate shareable links and compress files for evidence submission.	CloudServer
Coordinate upload with cloud server.	Recording

2.9 CRC CARD - StorageRepository

Class: StorageRepository	
Provides persistent data storage and retrieval for recordings.	
Responsibilities	Collaborators
Handle persistent storage of recordings.	Recording
Support retrieval, deletion, and integrity verification.	PlaybackController

Interface between local system and cloud backup.	EvidenceExporter
--	------------------

2.10 CRC CARD - CloudServer

Class:CloudServer	
Responsibilities	Collaborators
Authorize and validate user access for cloud-stored recordings.	EvidenceExporter
Manage cloud uploads and downloads.	StorageRepository
Enforce access control for shared evidence links.	

2.11 CRC CARD - RecordingSettings

Class: RecordingSettings	
Responsibilities	Collaborators
Stores and manages camera recording configuration preferences.	
Define and store recording modes (continuous/event).	SettingsManager
Manage storage preferences (cloud/local).	SafeHomeHub
Enable or disable audio for recordings.	
Save and apply updated settings.	

2.12 CRC CARD - NotificationPolicy

Class: NotificationPolicy

Defines and controls system notification behavior and cooldown rules.	
Responsibilities	Collaborators
Define notification rules and cooldown times.	SettingsManager
Manage delivery types and recipient lists.	SafeHomeHub
Enable/disable alert policy.	

2.13 CRC CARD - SettingsManager

Class: SettingsManager	
Oversees updating and synchronization of surveillance configuration.	
Responsibilities	Collaborators
Manage all user-configurable surveillance preferences.	RecordingSettings
Update recording and notification settings.	NotificationPolicy
Synchronize updated settings to hub devices.	SafeHomeHub

2.14 CRC CARD - SafeHomeHub

Class: SafeHomeHub	
Acts as the operational core applying and maintaining configuration changes.	
Responsibilities	Collaborators
Apply configuration updates from settings manager.	RecordingSettings
Maintain consistent operational rules across devices.	SettingsManager

Handle synchronization of recording and alert parameters.	NotificationPolicy
---	--------------------

3. CRC CARDS – SYSTEM AND USER MANAGEMENT

3.1 CRC CARD - Device

Class: Device	
Responsibilities	Collaborators
<ul style="list-style-type: none"> • Store device identification and metadata • Track device status and connectivity • Manage firmware version information • Execute commands sent to the device • Report health and operational status 	<ul style="list-style-type: none"> • DeviceConfiguration • DeviceHealthMonitor • Room • Zone • Command • Account

3.2 CRC CARD - DeviceRegistration

Class: DeviceRegistration	
Responsibilities	Collaborators
<ul style="list-style-type: none"> • Initiate device pairing process • Generate and validate pairing codes • Complete device registration • Track registration metadata • Transfer device ownership 	<ul style="list-style-type: none"> • Device • Account • Hub

3.3 CRC CARD - DeviceConfiguration

Class: DeviceConfiguration

Manages device-specific settings, validates changes, and applies configuration updates.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Store device settings and parameters• Validate configuration changes• Apply configuration to devices• Maintain configuration history• Export/import configurations	<ul style="list-style-type: none">• Device• Setting• ValidationRule

3.4 CRC CARD - Room

Class: Room

Organizes devices by physical location and provides room-level device management.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Group devices by physical location• Store room metadata and type• Track devices within the room• Provide device queries by type/status	<ul style="list-style-type: none">• Device• Zone• Account

3.5 CRC CARD - Zone

Class: Zone

Groups rooms for security control and manages armed states and entry/exit rules.

Responsibilities	Collaborators

<ul style="list-style-type: none"> • Group multiple rooms for security purposes • Manage security level and armed status • Configure entry/exit delays • Coordinate room-level security 	<ul style="list-style-type: none"> • Room • Device • Account • SecurityMode
---	---

3.6 CRC CARD - DeviceHealthMonitor

Class: DeviceHealthMonitor

Continuously evaluates device health, connectivity, and predictive failure indicators.

Responsibilities	Collaborators
<ul style="list-style-type: none"> • Track battery level and connectivity • Monitor signal strength and quality • Calculate overall health score • Perform diagnostic health checks • Predict potential failures 	<ul style="list-style-type: none"> • Device • HealthCheckResult • BatteryStatus • ConnectivityStatus

3.7 CRC CARD - SystemStatusDashboard

Class: SystemStatusDashboard

Displays real-time system status using customizable widgets and layouts.

Responsibilities	Collaborators
<ul style="list-style-type: none"> • Display system overview information • Manage dashboard layout and widgets • Control refresh intervals • Provide customizable views • Export dashboard data 	<ul style="list-style-type: none"> • Widget • Account • DashboardLayout

3.8 CRC CARD - Widget

Class: Widget

Visual component that retrieves and displays specific system data within the dashboard.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Render specific data visualization• Fetch data from assigned source• Handle user interactions• Update based on refresh interval• Maintain position and size	<ul style="list-style-type: none">• DataSource• SystemStatusDashboard• WidgetData

3.9 CRC CARD - SystemHealthMetrics

Class: SystemHealthMetrics

Computes overall system health statistics and identifies devices requiring attention.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Calculate system-wide device statistics• Track online/offline device counts• Monitor overall system health score• Identify devices needing attention• Generate health reports	<ul style="list-style-type: none">• Device• Account• HealthReport• HubStatus

3.10 CRC CARD - ActivityTimeline

Class: ActivityTimeline

Maintains and organizes the chronological history of system events.

Responsibilities	Collaborators

<ul style="list-style-type: none"> • Maintain chronological event history • Filter and search events • Paginate event listings • Group events by criteria • Export timeline data 	<ul style="list-style-type: none"> • ActivityEvent • Account • TimelineFilter
---	--

3.11 CRC CARD - ActivityEvent

Class: ActivityEvent

Represents an individual recorded system activity with contextual metadata.

Responsibilities	Collaborators
<ul style="list-style-type: none"> • Record system activity details • Store event metadata and context • Track actor and target information • Link to related media • Support acknowledgment workflow 	<ul style="list-style-type: none"> • User • Device • MediaReference • ActivityTimeline

3.12 CRC CARD - AuditLog

Class: AuditLog

Stores a tamper-proof record of all user actions and system changes for security and compliance.

Responsibilities	Collaborators
<ul style="list-style-type: none"> • Record all user actions and changes • Track resource modifications • Store security-relevant events • Maintain tamper-proof history • Support compliance reporting 	<ul style="list-style-type: none"> • User • Account • Change • ResourceType

3.13 CRC CARD - LogAnalyzer

Class: LogAnalyzer

Analyzes logs to detect unusual patterns, threats, and operational insights.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Analyze event patterns and trends• Detect anomalous behavior• Generate insights and recommendations• Identify security threats• Produce analysis reports	<ul style="list-style-type: none">• ActivityEvent• AuditLog• Pattern• Anomaly• AnalysisReport

3.14 CRC CARD - User

Class: User

Stores user identity and authentication data while enforcing login, MFA, and password policies.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Store user credentials and profile• Authenticate login attempts• Manage two-factor authentication• Track login history and status• Enforce password policies	<ul style="list-style-type: none">• Account• Permission• UserSession• UserPreferences• NotificationPreferences

3.15 CRC CARD - Account

Class: Account

Represents an account's settings, billing, device limits, and lifecycle management.

Responsibilities	Collaborators

<ul style="list-style-type: none"> • Manage account-level settings • Track subscription and billing • Enforce user and device limits • Store account preferences • Control account lifecycle 	<ul style="list-style-type: none"> • User • SubscriptionPlan • BillingInfo • Device • Room • Zone
---	---

3.16 CRC CARD - SubscriptionPlan

Class: SubscriptionPlan

Defines service tiers, resource limits, and pricing for accounts.

Responsibilities	Collaborators
<ul style="list-style-type: none"> • Define service tier capabilities • Specify resource limits • List included features • Store pricing information • Determine feature availability 	<ul style="list-style-type: none"> • Account • Feature • BillingInfo

3.17 CRC CARD - Permission

Class: Permission

Specifies access rights and rules governing what actions a user may perform.

Responsibilities	Collaborators
<ul style="list-style-type: none"> • Define access rights for users • Specify allowed actions on resources • Enforce permission scope • Support conditional permissions • Track permission lifecycle 	<ul style="list-style-type: none"> • User • ResourceType • PermissionAction • PermissionCondition

3.18 CRC CARD - UserSession

Class: UserSession

Manages authentication sessions, tokens, and related device/location context.

Responsibilities	Collaborators
<ul style="list-style-type: none"> Manage authentication tokens Track session lifecycle and expiration Store device and location context Support session refresh Enable session revocation 	<ul style="list-style-type: none"> User DeviceInfo GeoLocation Account

3.19 CRC CARD - CameraAccessControl

Class: CameraAccessControl

Regulates user access to cameras, enforcing permissions and logging access attempts.

Responsibilities	Collaborators
<ul style="list-style-type: none"> Manage camera-specific access Maintain authorized user list Log camera access attempts Enforce password protection Handle access violations 	<ul style="list-style-type: none"> Camera User CameraAccessEvent Account

3.20 CRC CARD - NotificationPreferences

Class: NotificationPreferences

Stores user-specific rules for notification channels, quiet hours, and alert preferences.

Responsibilities	Collaborators
<ul style="list-style-type: none"> Configure notification channels Manage quiet hours settings Define event-specific preferences Control notification frequency Support do-not-disturb modes 	<ul style="list-style-type: none"> User NotificationSetting EventType NotificationChannel

3.21 CRC CARD - Command

Class: Command

Encapsulates an actionable instruction sent to a device along with its execution results.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Encapsulate device commands• Track command execution status• Support command retry logic• Store command parameters• Return execution results	<ul style="list-style-type: none">• Device• User• CommandResult

3.22 CRC CARD - Setting

Class: Setting

Represents a configurable value with validation rules and constraints.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Store individual configuration values• Validate setting values• Track data types and constraints• Support default values• Handle read-only settings	<ul style="list-style-type: none">• DeviceConfiguration• ValidationRule• DataType

3.23 CRC CARD - HealthCheckResult

Class: HealthCheckResult

Stores diagnostic outcomes and identified issues from device health checks.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Store diagnostic test results• Identify health issues• Provide recommendations• Track test execution metadata• Support result export	<ul style="list-style-type: none">• DeviceHealthMonitor• DiagnosticTest• HealthIssue

3.24 CRC CARD - MediaReference

Class: MediaReference

Points to stored media files with metadata and access URLs for playback or download.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Reference stored media files• Provide media metadata• Generate streaming URLs• Support thumbnails• Enable media download	<ul style="list-style-type: none">• ActivityEvent• Camera• Recording

3.25 CRC CARD - UserPreferences

Class: UserPreferences

Holds user interface customization and localization preferences.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Store user interface preferences• Manage localization settings• Configure display options• Maintain theme preferences• Support preference reset	<ul style="list-style-type: none">• User• Theme• TemperatureUnit

3.26 CRC CARD - BillingInfo

Class: BillingInfo

Contains payment details, billing cycles, and renewal settings for an account.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Store payment information• Track billing cycles• Manage billing address• Handle auto-renewal settings• Process payment methods	<ul style="list-style-type: none">• Account• PaymentMethod• Address• SubscriptionPlan

3.27 CRC CARD - TimelineFilter

Class: TimelineFilter

Defines rules to filter and search through timeline events.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Define event filtering criteria• Support multi-attribute filtering• Enable search functionality• Match events against criteria• Support filter persistence	<ul style="list-style-type: none">• ActivityTimeline• ActivityEvent• EventType

3.28 CRC CARD - DeviceInfo

Class: DeviceInfo

Stores metadata about a user's client device, including fingerprinting and trust status.

Responsibilities	Collaborators
<ul style="list-style-type: none">• Store client device metadata• Generate device fingerprints• Track OS and app versions• Manage push notification tokens• Determine device trust status	<ul style="list-style-type: none">• UserSession• ClientDeviceType• OSType

3.29 CRC CARD - PasswordPolicy

Class: PasswordPolicy

Enforces password requirements, strength rules, expiration, and lockout policies.

Responsibilities	Collaborators

<ul style="list-style-type: none"> • Define password requirements • Validate password strength • Enforce password expiration • Prevent password reuse • Configure lockout settings 	<ul style="list-style-type: none"> • Account • User • ValidationResult
---	---

3.30 CRC CARD - UserInvitation

Class: UserInvitation

Manages invitation creation, tracking, and acceptance for onboarding new users.

Responsibilities	Collaborators
<ul style="list-style-type: none"> • Create user invitations • Generate invitation codes • Track invitation status • Handle invitation acceptance • Support invitation expiration 	<ul style="list-style-type: none"> • Account • User • UserRole • Permission

4. CRC CARDS – REMOTE ACCESS AND ACCOUNT

4.1 CRC CARD - Gateway

Class: Gateway	
Entry point: routes requests, enforces prechecks, attaches user context	
Responsibilities	Collaborators
Terminate TLS, route to services	AuthService, RemoteAccessController
Enforce basic auth (bearer/api key) & rate limits	PolicyEnforcer
Attach correlation IDs & user context	SessionManager
Emit request/response audit events	AuditLogger

4.2 CRC CARD - LoginService

Class: LoginService

Sign up / sign in; password hashing & rotation; start verification/MFA	
Responsibilities	Collaborators
Register user, hash credentials	CredentialStore, IdentityService
Primary auth (username/password)	AuthService
Kick off activation / resets	VerificationService, NotificationService
Create sessions on success	SessionManager
Emit security events	AuditLogger

4.3 CRC CARD - AuthService

Class: AuthService	
Central auth: validates creds, issues/validates tokens, device trust hooks	
Responsibilities	Collaborators
Validate credentials & risk checks	CredentialStore, DeviceTrustService
Issue/verify JWT; key rotation	TokenService, KMS
Step-up decisions (require MFA)	PolicyEnforcer, MFAService
Log successes/failures	AuditLogger

4.4 CRC CARD - SessionManager

Class: SessionManager	
Create/refresh/revoke sessions; sliding expiry; cache-backed	
Responsibilities	Collaborators
Create session + refresh tokens	TokenService
Revoke/blacklist on logout/compromise	AuthService, AuditLogger
Store in Redis; TTL mgmt	Redis
Attach session to requests	Gateway

4.5 CRC CARD - MFAService

Class: MFAService	
Enroll/verify TOTP, SMS OTP, backup codes; step-up challenges	
Responsibilities	Collaborators

Enroll factors, generate secrets	IdentityService
Deliver OTPs	NotificationService
Verify codes, rate-limit attempts	AuthService, PolicyEnforcer
Backup code lifecycle	IdentityService
Audit MFA events	AuditLogger

4.6 CRC CARD - PolicyEnforcer

Class: PolicyEnforcer	
Evaluate roles/policies; permit/deny; cache decisions	
Responsibilities	Collaborators
Evaluate RBAC & ABAC rules	RoleStore, IdentityService
Step-up requirements (MFA/reauth)	AuthService, MFAService
Decision caching & invalidation	Redis
Emit policy audit	AuditLogger

4.7 CRC CARD - IdentityService

Class: IdentityService	
User record, profile/prefs, consent, contacts	
Responsibilities	Collaborators
CRUD user/profile/consent/contact	UserRepo
Provide PII to verifications	VerificationService
Apply policy flags/locks	PolicyEnforcer
Emit identity changes	AuditLogger

4.8 CRC CARD - VerificationService

Class: VerificationService	
Email/phone activation; password reset; contact change confirmations	
Responsibilities	Collaborators
Generate signed tokens/links	TokenService, KMS
Send activation/reset/change	NotificationService

Validate tokens; update state	IdentityService, LoginService
Expire/cleanup tokens	Redis/DB
Log verification events	AuditLogger

4.9 CRC CARD - RemoteAccessController

Class: RemoteAccessController	
Establish/maintain tunnels or proxy sessions to targets	
Responsibilities	Collaborators
Policy check & authorization	PolicyEnforcer
Establish tunnel/proxy	Gateway, BackendConnector
Reauth/refresh on long sessions	SessionManager, MFAService
Handle reconnect windows	SessionManager
Emit session lifecycle events	AuditLogger

4.10 CRC CARD - TokenService

Class: TokenService	
Sign/verify JWT; rotate keys; introspect/revoke lists	
Responsibilities	Collaborators
Sign/verify tokens	KMS
Rotate keys; publish JWKS	Gateway
Introspection & revocation checks	SessionManager
Audit token operations	AuditLogger

4.11 CRC CARD - DeviceTrustService

Class: Gateway	
Fingerprint/enroll devices; assess trust/risk	
Responsibilities	Collaborators
Enroll device & bind to user	IdentityService
Score trust; enforce policy	PolicyEnforcer
Store fingerprints/attestations	DeviceStore
Log device events	AuditLogger

4.12 CRC CARD - NotificationService

Class: NotificationService	
Email/SMS/Push delivery with provider failover	
Responsibilities	Collaborators
Template rendering & send	Providers
Failover/retry, rate limit	MFAService, VerificationService
Delivery receipts/metrics	AuditLogger

5. CRC CARDS – INDOOR MONITORING AND DEVICE CONTROL

5.1 CRC CARD - ConfigurationManager

Class: ConfigurationManager	
Temporary user with limited access permissions (e.g., arm/disarm only)	
Responsibilities	Collaborators
Load/save/apply configuration profile	ConfigurationProfile
Update global/system settings	SystemSettings
Manage zones/rooms in active profile	SafetyZone, Room
Log configuration changes	SafeHomeHub, CloudServer
Sync configs with hub/cloud	ActivityLog

5.2 CRC CARD - ConfigurationProfile

Class: ConfigurationProfile	
Container of zones, device configurations, and notification policies.	
Responsibilities	Collaborators
Hold zones and device settings	SafetyZone, DeviceConfiguration
Store notification policy	NotificationPolicy

Version & clone profile	ConfigurationManager
-------------------------	----------------------

5.3 CRC CARD - SafetyZone

Class: SafetyZone	
Logical group of rooms/devices that can be armed/disarmed.	
Responsibilities	Collaborators
Add/remove rooms	Room
Arm/disarm zone	ConfigurationManager
Expose device collections per zone	SmartHomeDevice

5.4 CRC CARD - Room

Class: Room	
Physical location for devices.	
Responsibilities	Collaborators
Hold devices in room	SmartHomeDevice
Query devices by type/status	DeviceController
Report membership to zone	SafetyZone

5.5 CRC CARD - DeviceController

Class: DeviceController	
Executes device commands and reads telemetry.	
Responsibilities	Collaborators
Execute command (turnOn/off/set mode)	Command, CommandResult, SmartHomeDevice
Read power usage	PowerMeter
Coordinate auto-ventilation	AirQualitySensor, VentilationController
Log actions	ActivityLog
Generate energy reports	EnergyReport

5.6 CRC CARD - Command

Class: Command	
Encodes an action + parameters for a device.	
Responsibilities	Collaborators
Carry name/params	DeviceController
Target a device	SmartHomeDevice

5.7 CRC CARD - CommandResult

Class: CommandResult	
Outcome of a command.	
Responsibilities	Collaborators
Report success/message	DeviceController

5.8 CRC CARD - SmartHomeDevice (abstract)

Class: SmartHomeDevice(Abstract)	
Base for controllable devices.	
Responsibilities	Collaborators
Report status & connectivity	DeviceController
Turn on/off	DeviceController
Provide common metadata (id, name)	Room

5.9 CRC CARD - SmartLight

Class: SmartLight	
Dimmable light device.	
Responsibilities	Collaborators
On/Off	DeviceController
Set brightness/color temperature	DeviceController

5.10 CRC CARD - Thermostat

Class: Thermostat	
Temperature control.	
Responsibilities	Collaborators
Set target temperature	DeviceController
Set mode (HEAT/COOL/ECO)	DeviceController
Report ambient temp	DeviceController

5.11 CRC CARD - VentilationController

Class: VentilationController	
Fan/air-exchange controller.	
Responsibilities	Collaborators
Set mode (AUTO/MANUAL/BOOST)	DeviceController
Set fan level	DeviceController
React to air-quality signals (AUTO)	AirQualitySensor

5.12 CRC CARD - SmartPlug

Class: SmartPlug	
Switchable outlet.	
Responsibilities	Collaborators
On/Off control	DeviceController

5.13 CRC CARD - AirQualitySensor

Class: AirQualitySensor	
Reads indoor AQ metrics (PM2.5 / CO ₂ / TVOC).	
Responsibilities	Collaborators
Sample and publish AQ metrics	DeviceController
Classify quality bands (Normal/Warning/Critical)	NotificationPolicy

Trigger ventilation in AUTO	VentilationController
-----------------------------	-----------------------

5.14 CRC CARD - PowerMeter

Class: PowerMeter	
Measures instantaneous and cumulative energy.	
Responsibilities	Collaborators
Provide instant watts	DeviceController
Provide kWh for period	EnergyReport

5.15 CRC CARD - EnergyReport

Class: EnergyReport	
Energy summary for a time range.	
Responsibilities	Collaborators
Aggregate kWh from meter data	PowerMeter
Export CSV / persist report	StorageManager, DeviceController

5.16 CRC CARD - NotificationPolicy

Class: NotificationPolicy	
Rules for when/how to alert users.	
Responsibilities	Collaborators
Evaluate cooldown/quiet hours	CooldownTimer
Select channels (push/SMS)	NotificationService
Bind to profile events	ConfigurationProfile

5.17 CRC CARD - CooldownTimer

Class: CooldownTimer	
Simple timer to avoid alert spam.	

Responsibilities	Collaborators
Start timer with duration	NotificationPolicy
Check expiry	NotificationPolicy

5.18 CRC CARD - ActivityLog

Class: ActivityLog	
Audit log of config changes and device actions.	
Responsibilities	Collaborators
Append entry	DeviceController, ConfigurationManager
Query by filter	ConfigurationManager
Persist/restore log	StorageManager

5.19 CRC CARD - Log

Class: Log	
Single log entry.	
Responsibilities	Collaborators
Store id, timestamp, description	ActivityLog

5.20 CRC CARD - StorageManager

Class: StorageManager	
Persistence for profiles, logs, and reports.	
Responsibilities	Collaborators
Save/load by key	ConfigurationManager, ActivityLog, EnergyReport

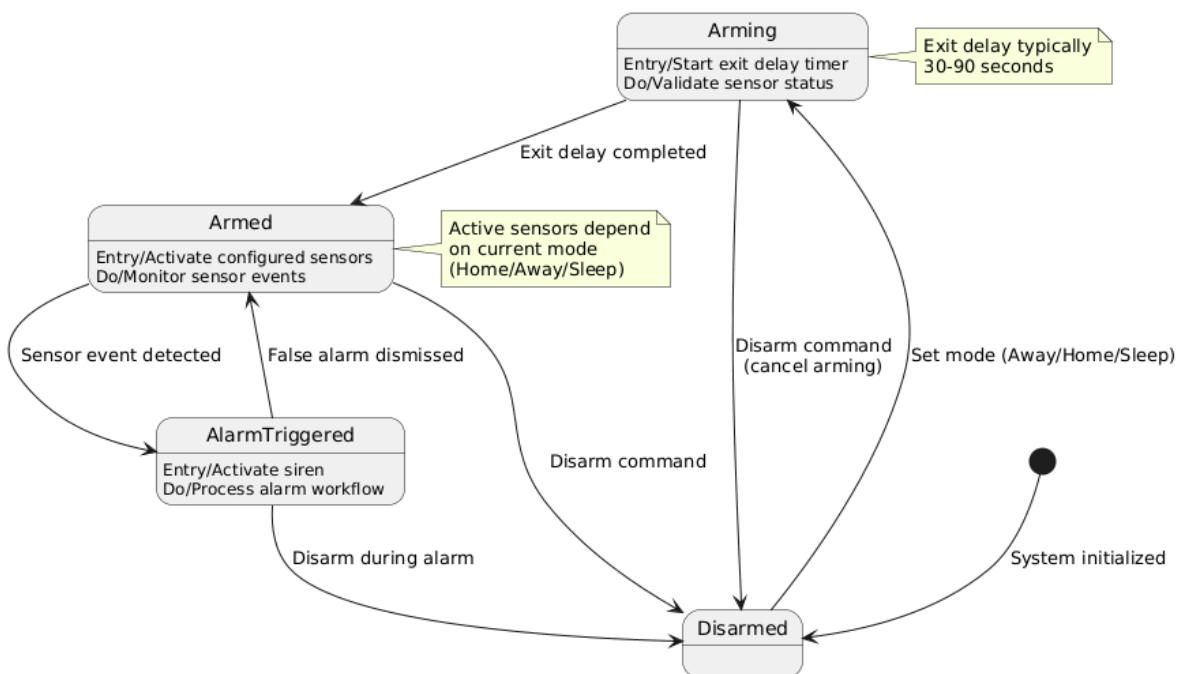
5.21 CRC CARD - SystemSettings

Class: SystemSettings	
Global app/system settings.	
Responsibilities	Collaborators

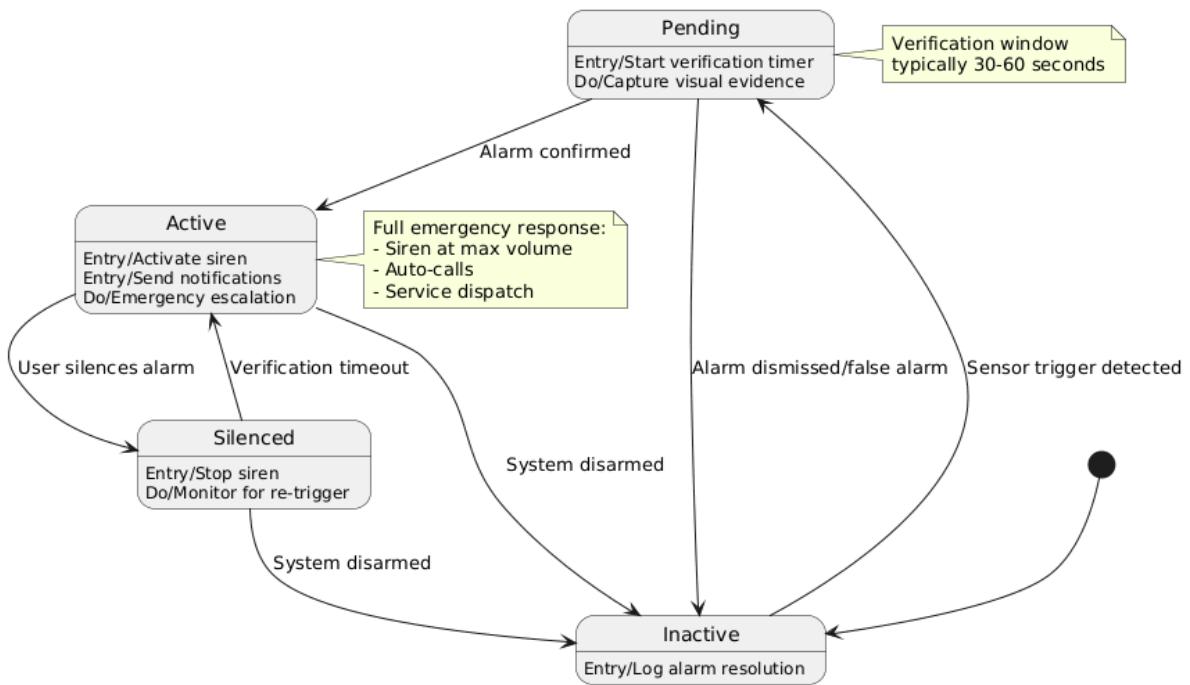
V. State Diagram

1. STATE DIAGRAM – INTELLIGENT SECURITY

1.1 State Diagram - SecurityMode

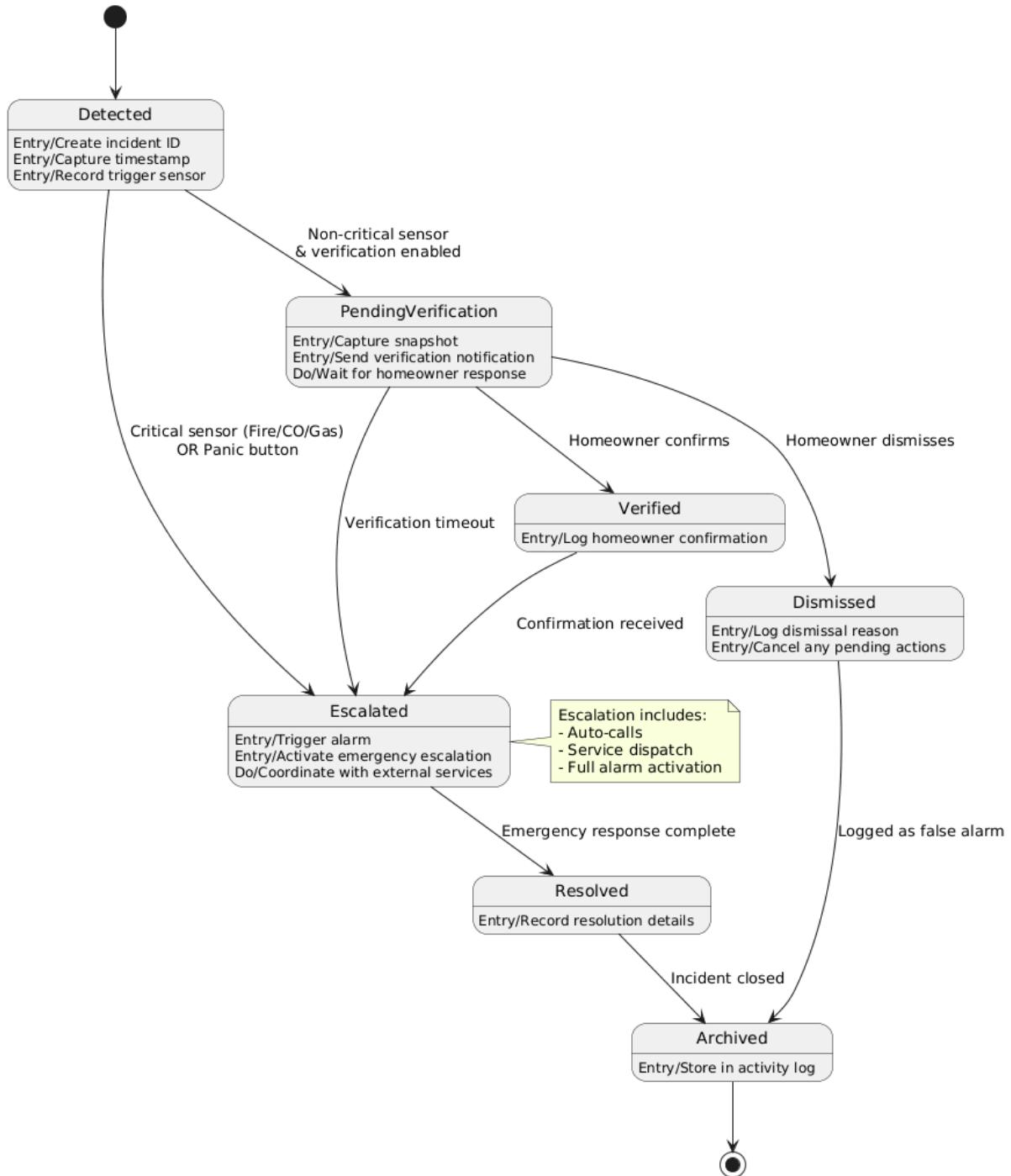


1.2 State Diagram - Alarm



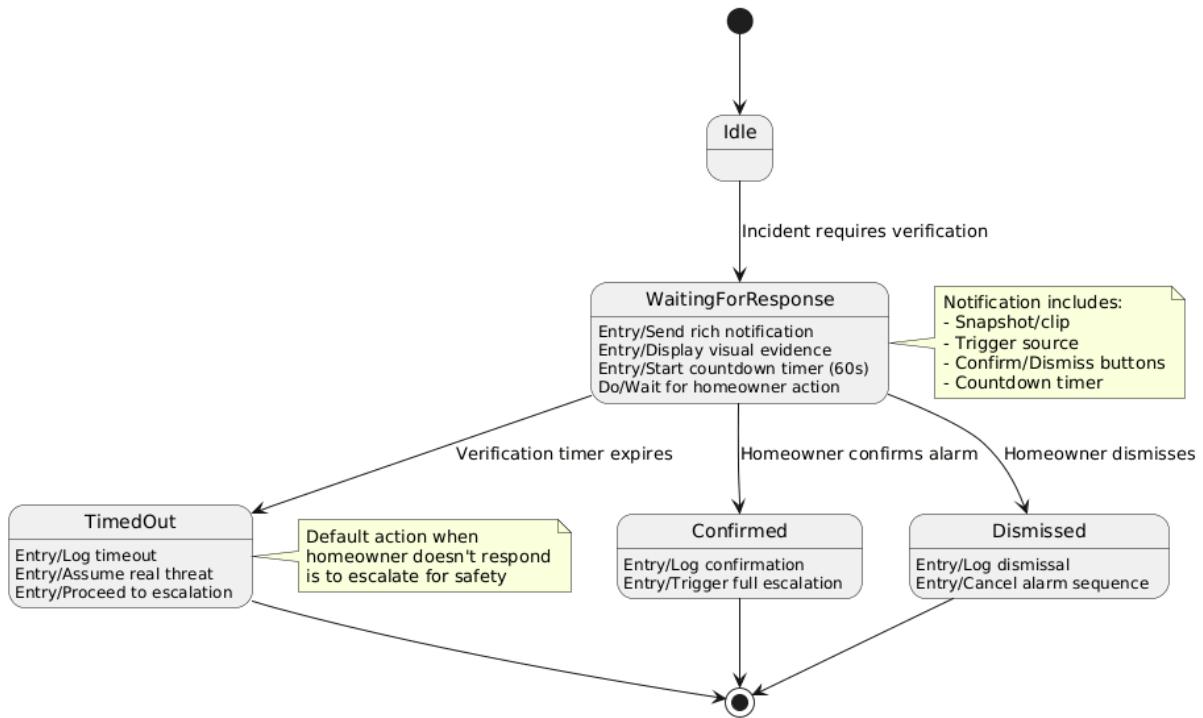
[Ref: SRS Glossary p.128 (Alarm Condition)]

1.3 State Diagram - Incident



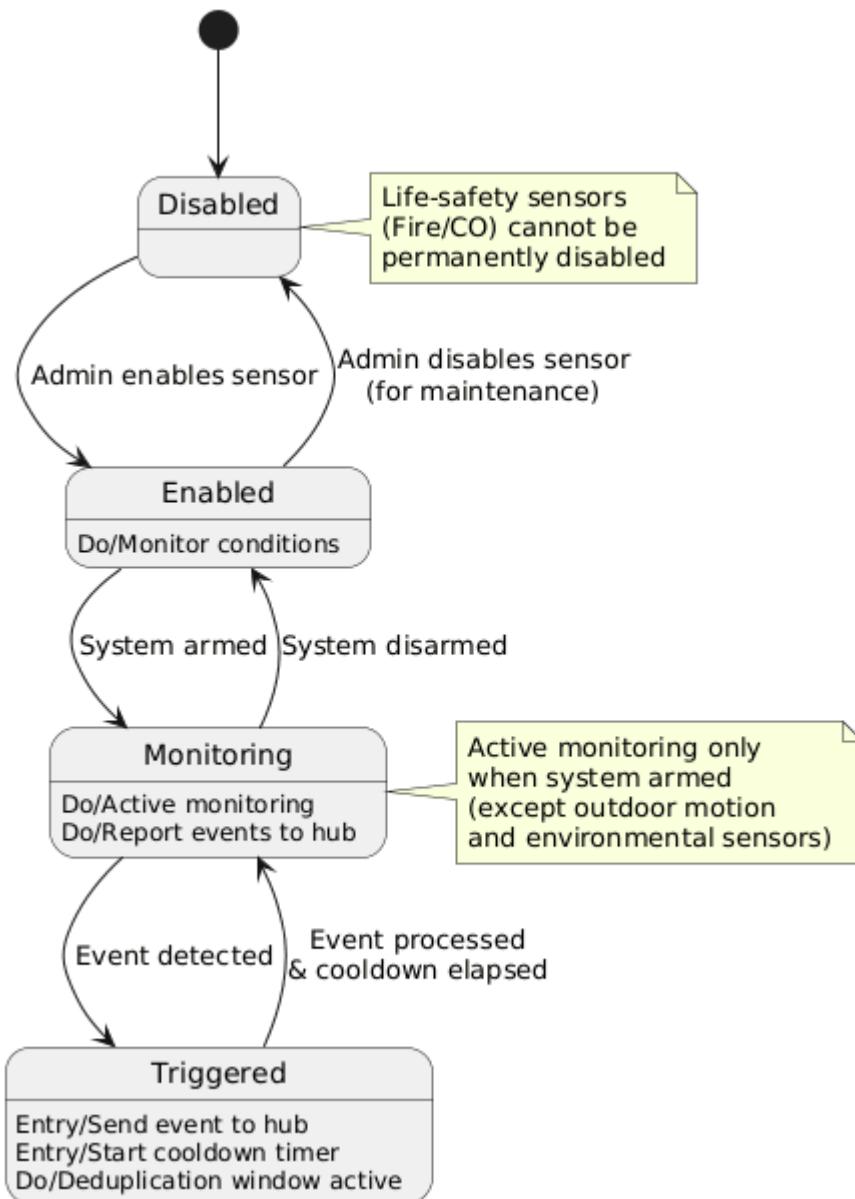
[Ref: SRS Glossary p.131 (Activity Log / Timeline)]

1.4 State Diagram - AlarmVerification



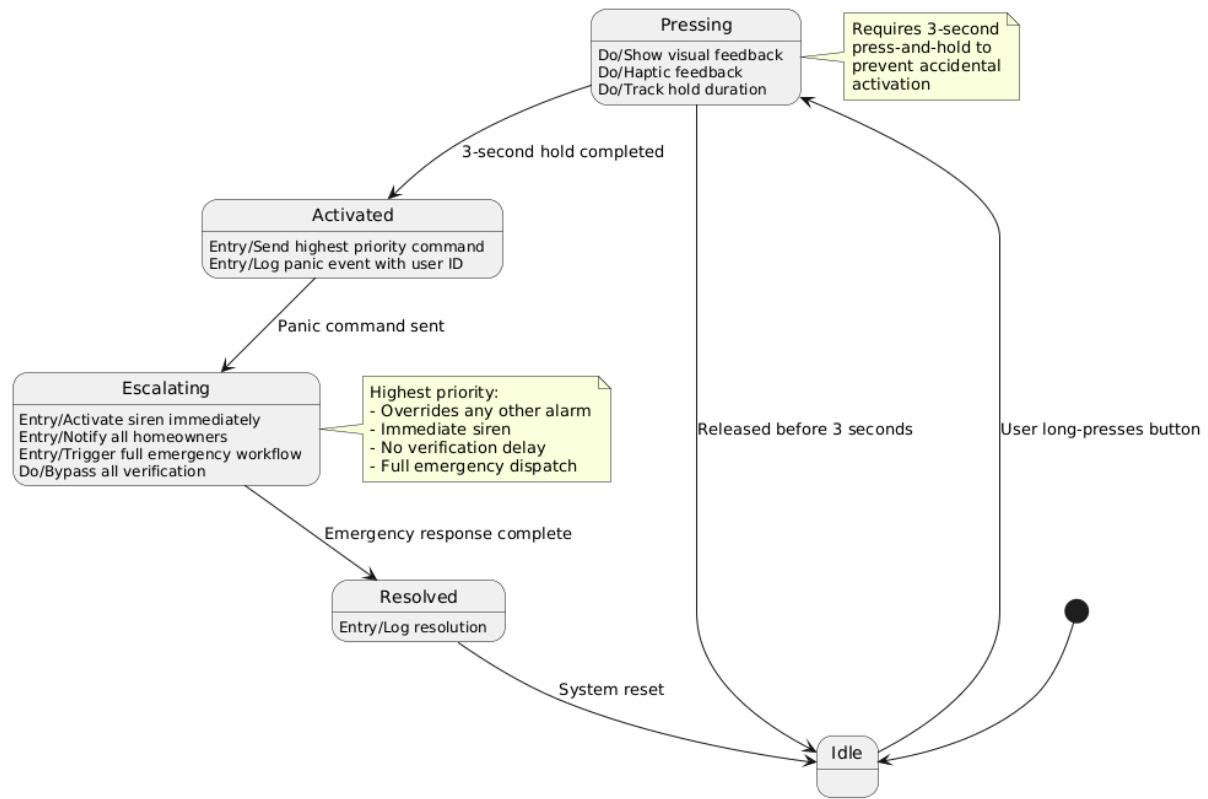
[Ref: SRS Glossary p.130 (Alarm Verification)]

1.5 State Diagram - Sensor



[Ref: SRS Glossary p.126 (Sensor)]

1.6 State Diagram - PanicButton

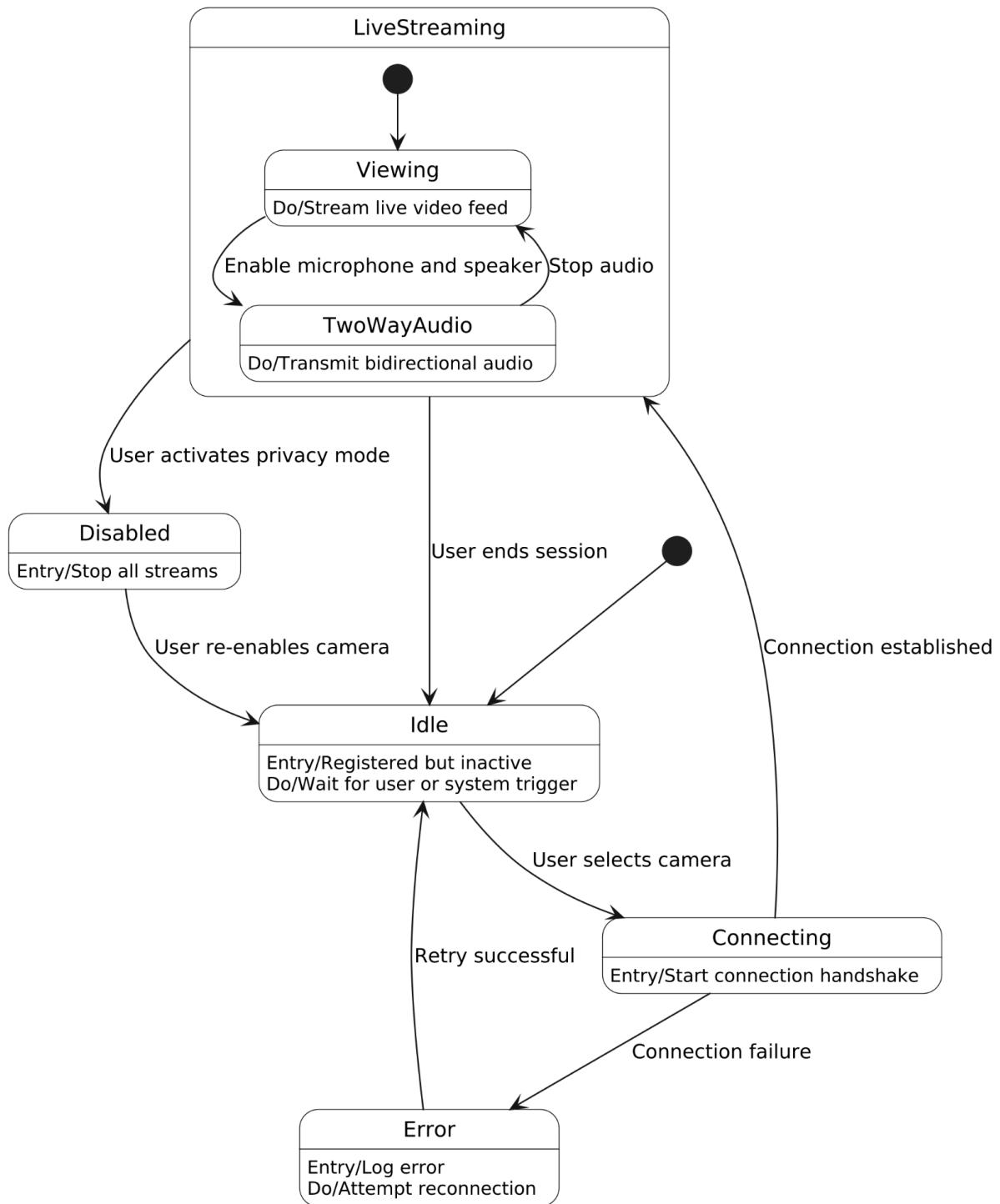


[Ref: SRS Glossary p.129 (Panic Button)]

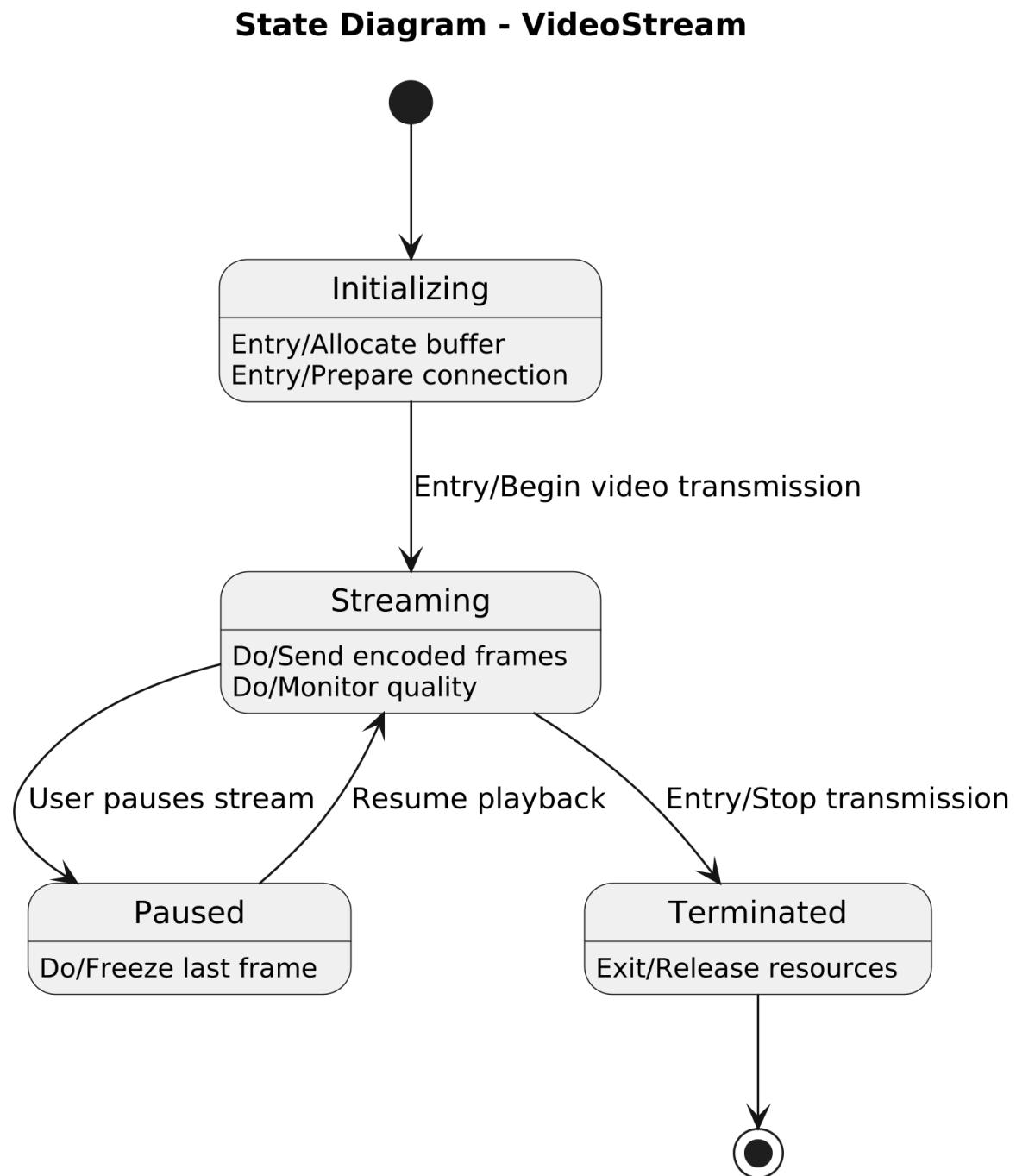
2. STATE DIAGRAM – LIVE SURVEILLANCE

2.1 State Diagram - CameraLifecycle

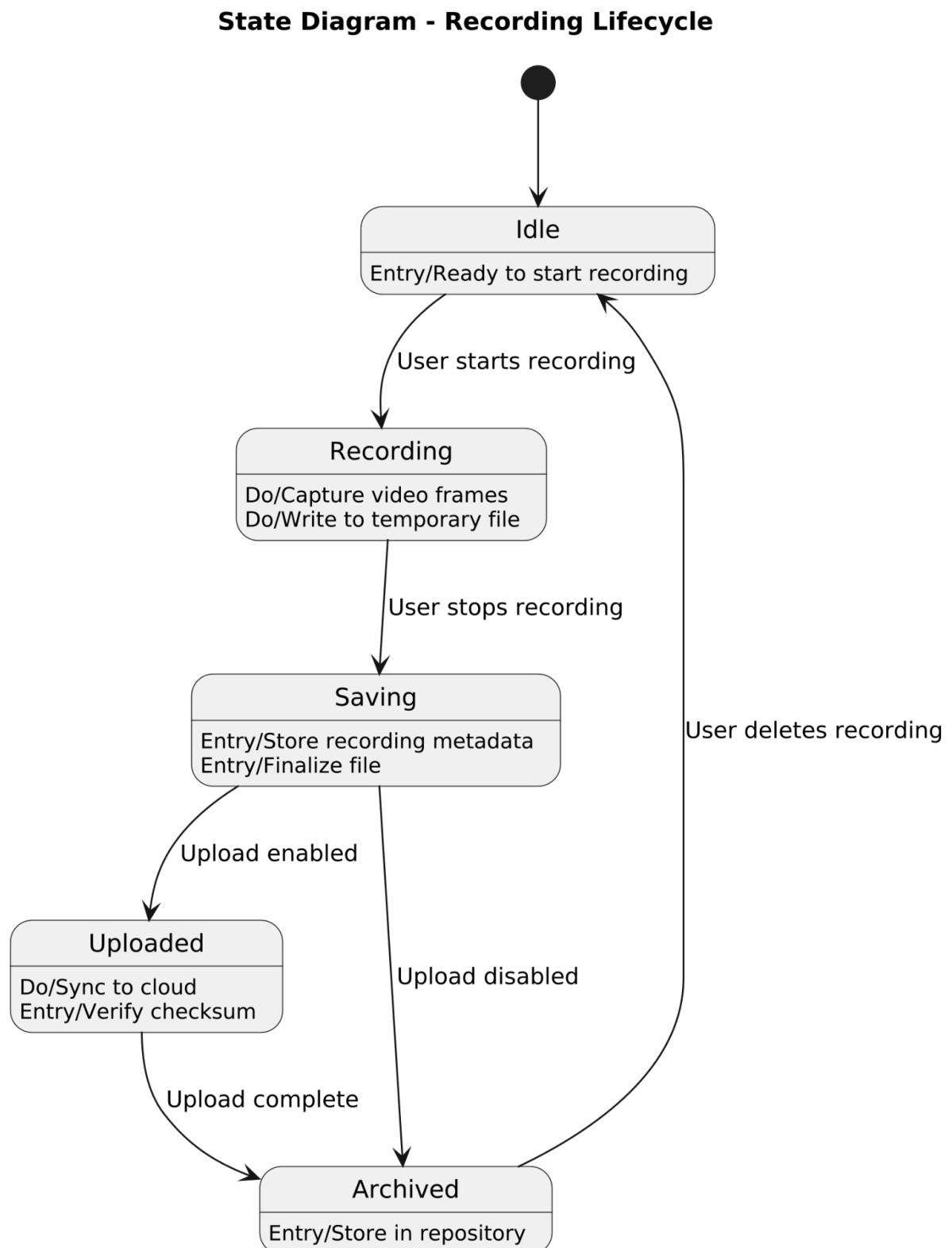
State Diagram - Camera Lifecycle



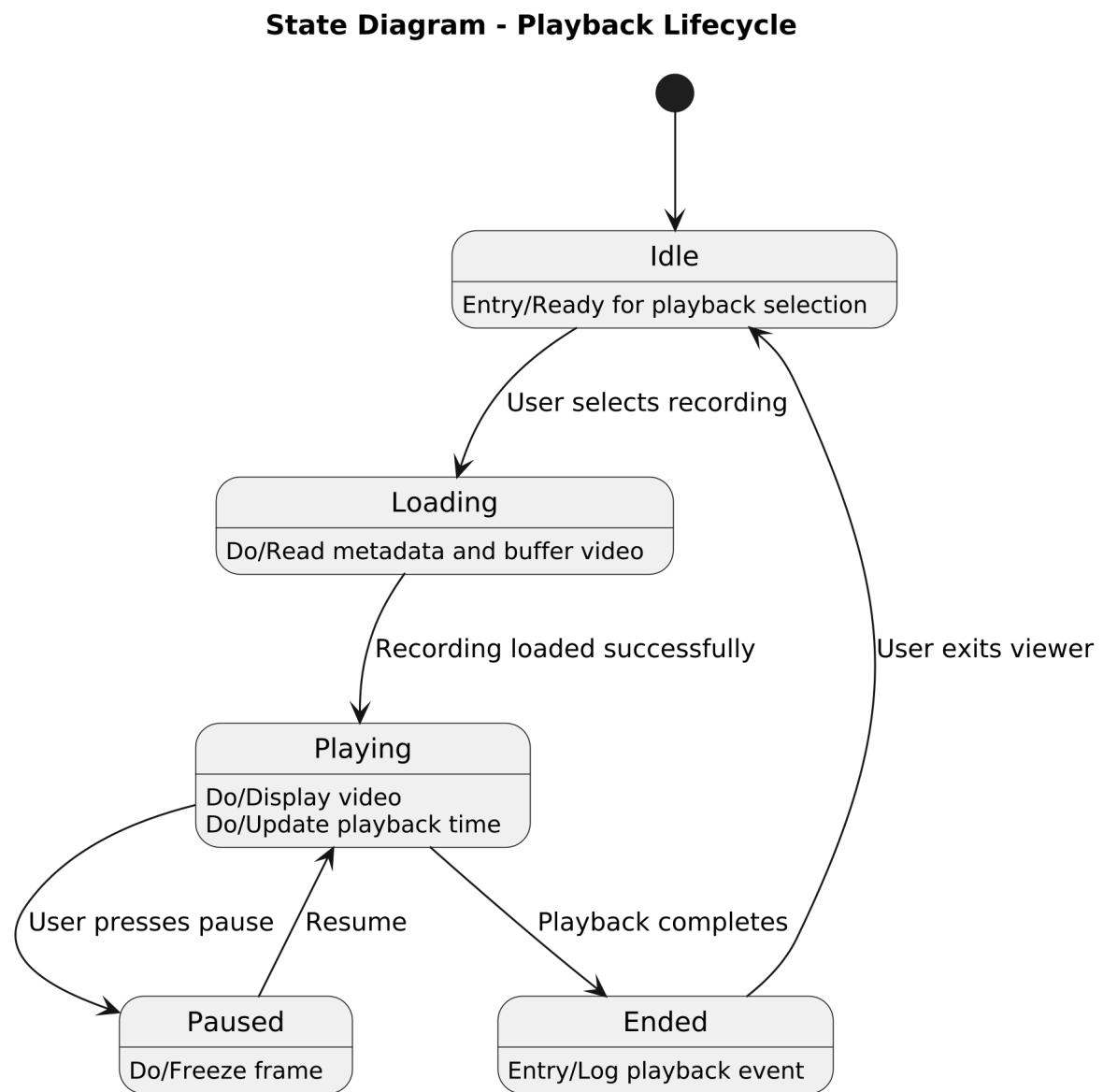
2.2 State Diagram - VideoStream



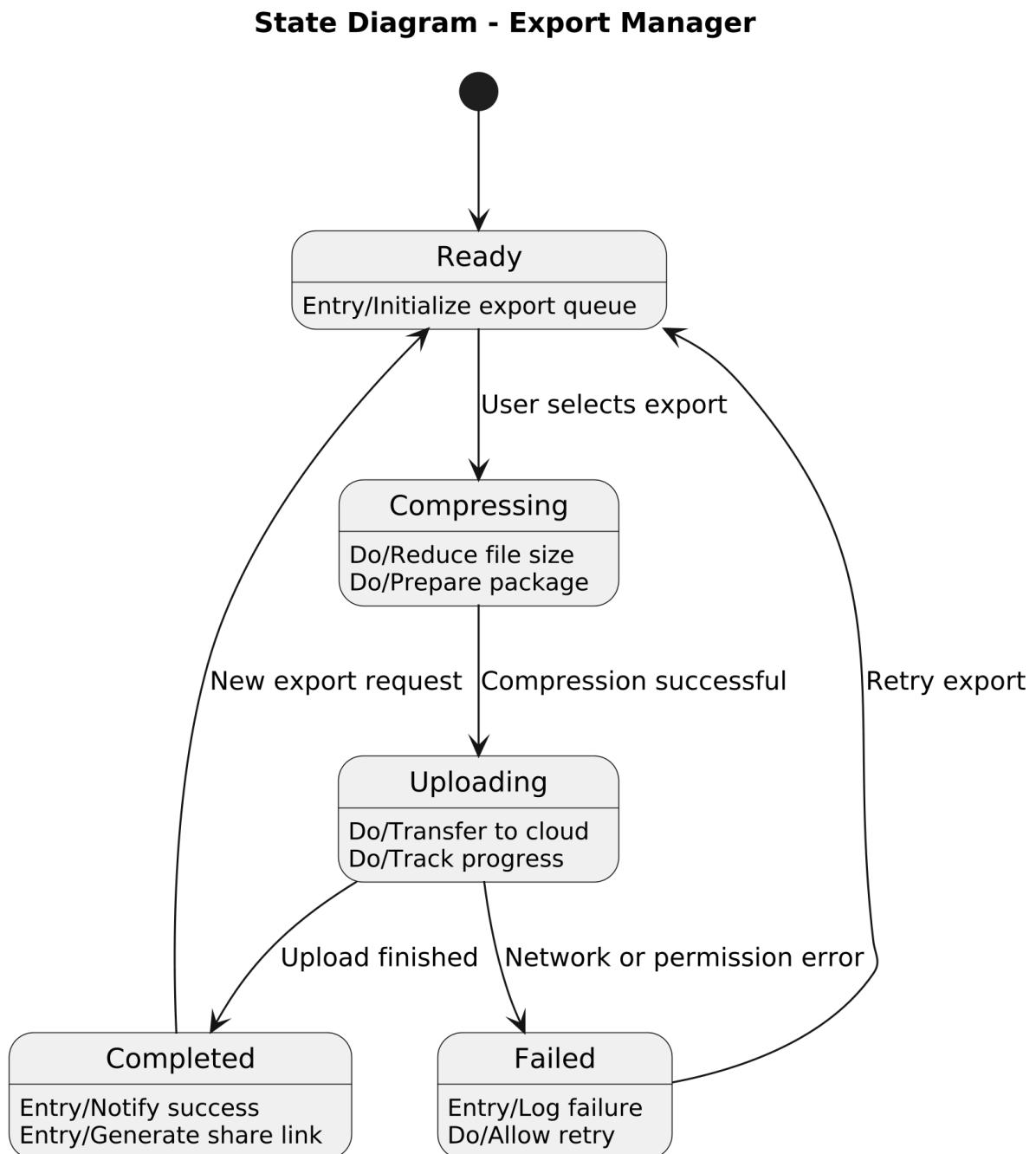
2.3 State Diagram - Recording Lifecycle



2.4 State Diagram - Playback Lifecycle

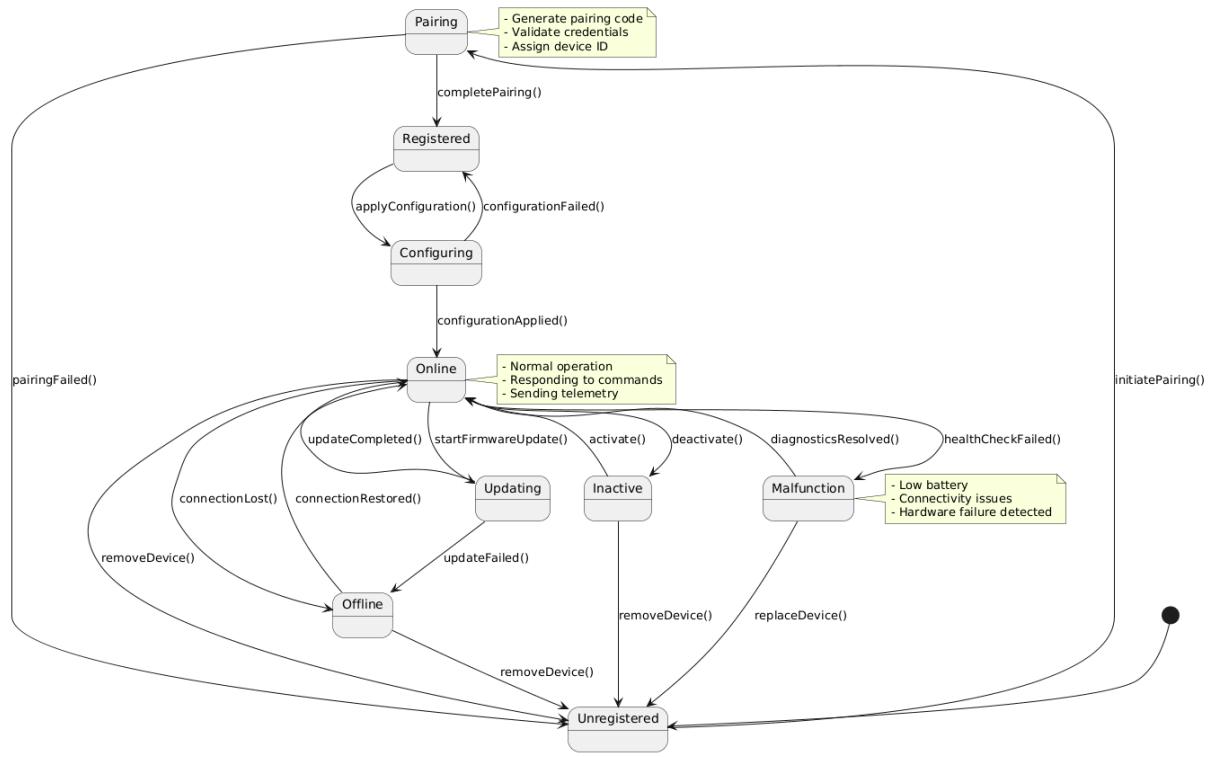


2.5 State Diagram - Export Manager

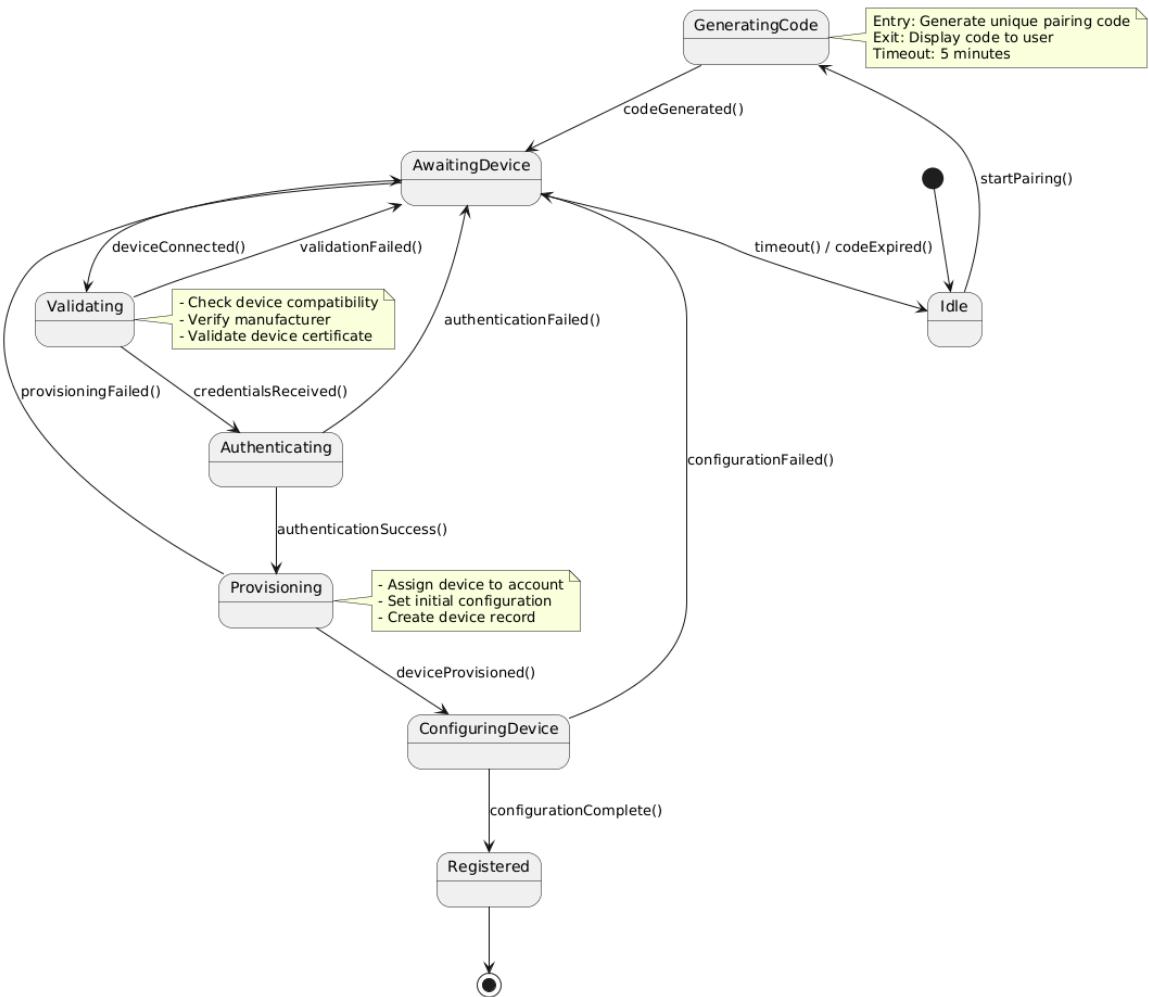


3. STATE DIAGRAM – SYSTEM AND USER MANAGEMENT

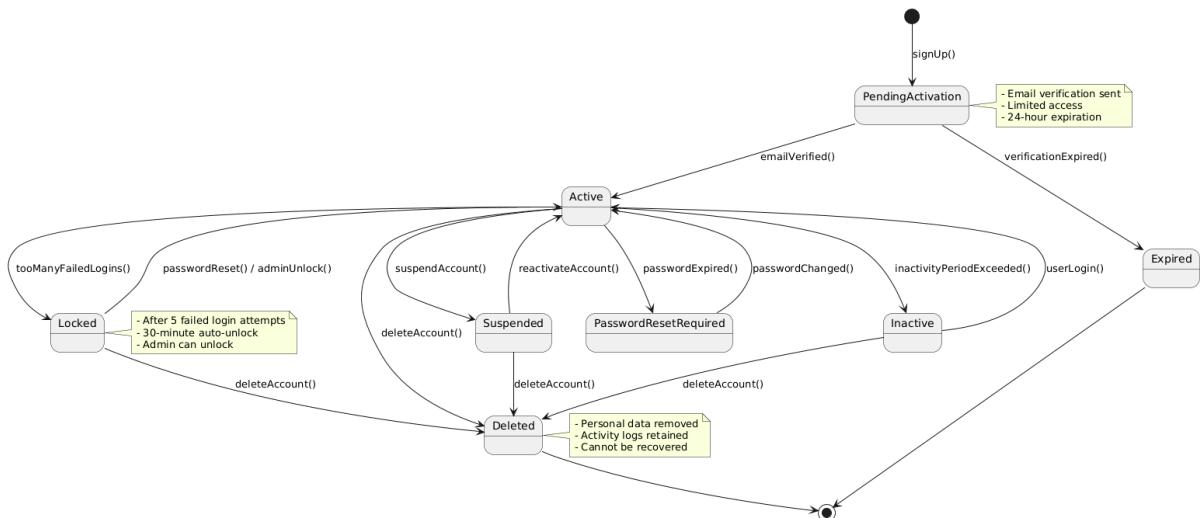
1. Device Lifecycle State Diagram



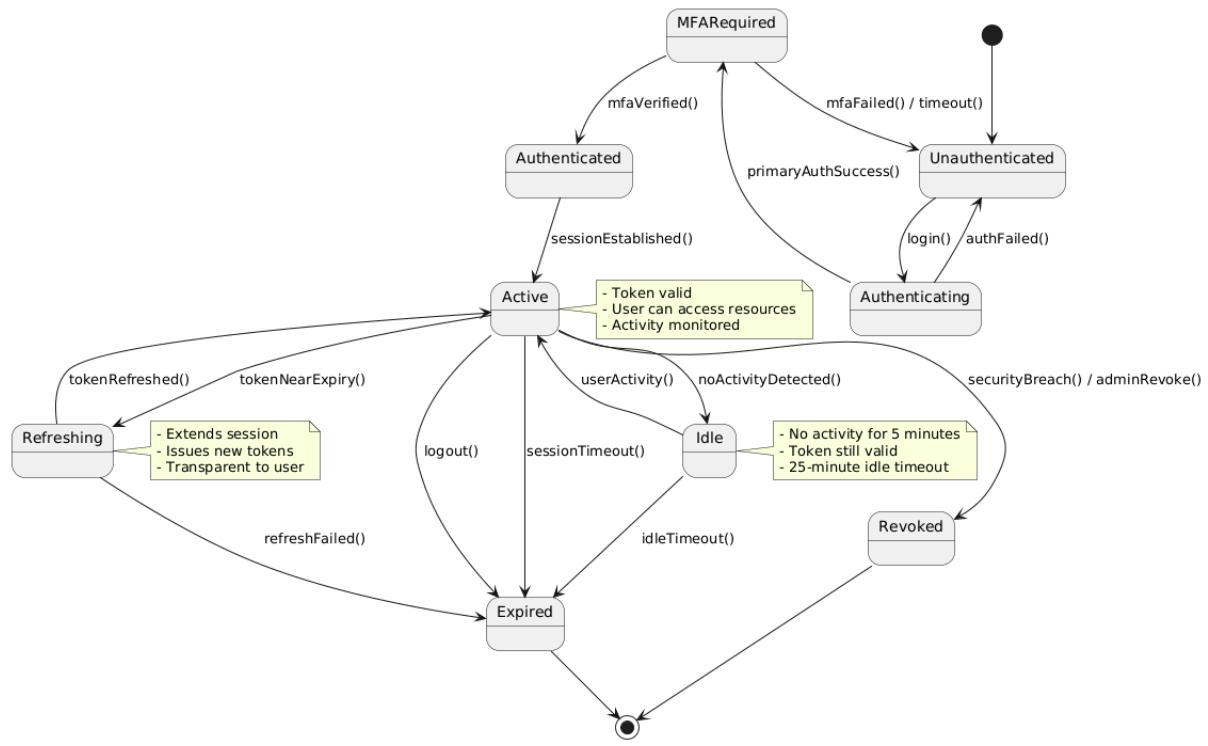
2. Device Registration Process State Diagram



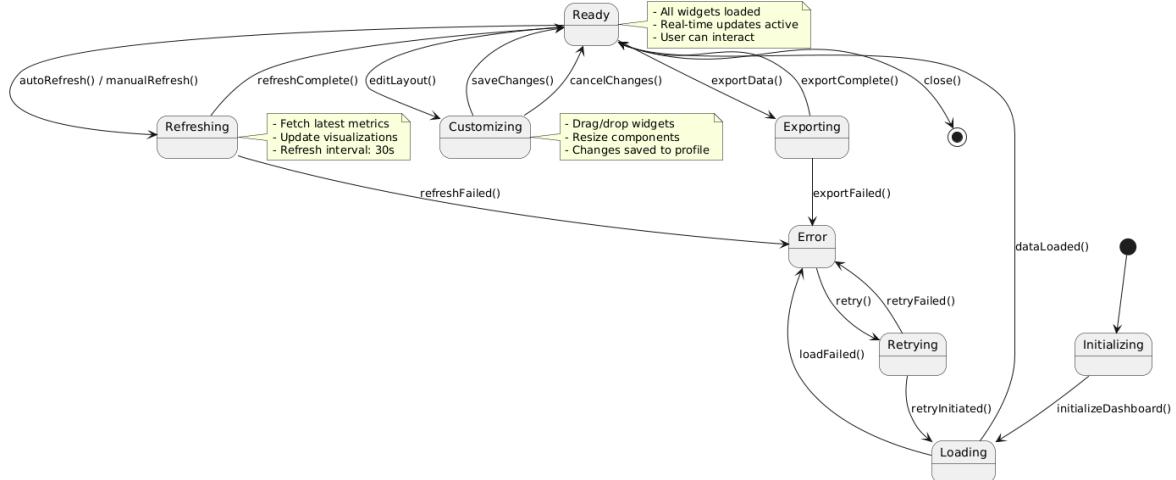
3. User Account Lifecycle State Diagram



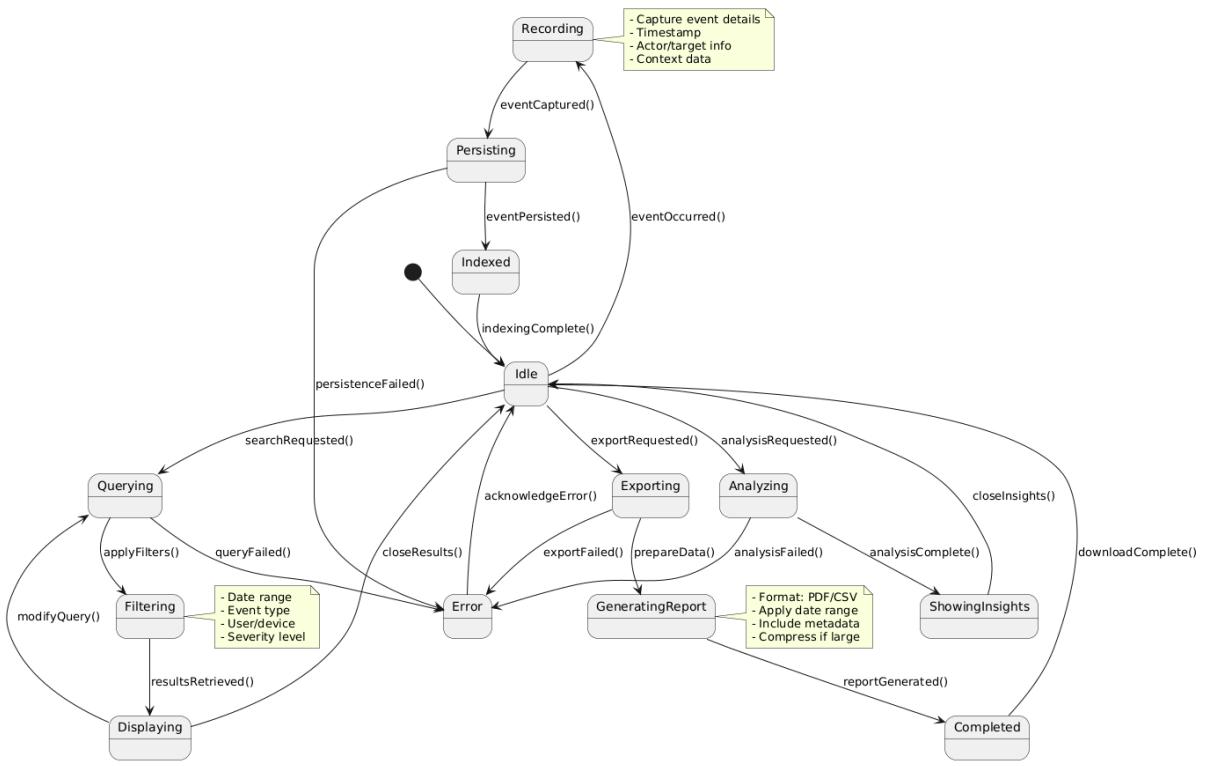
4. User Session State Diagram



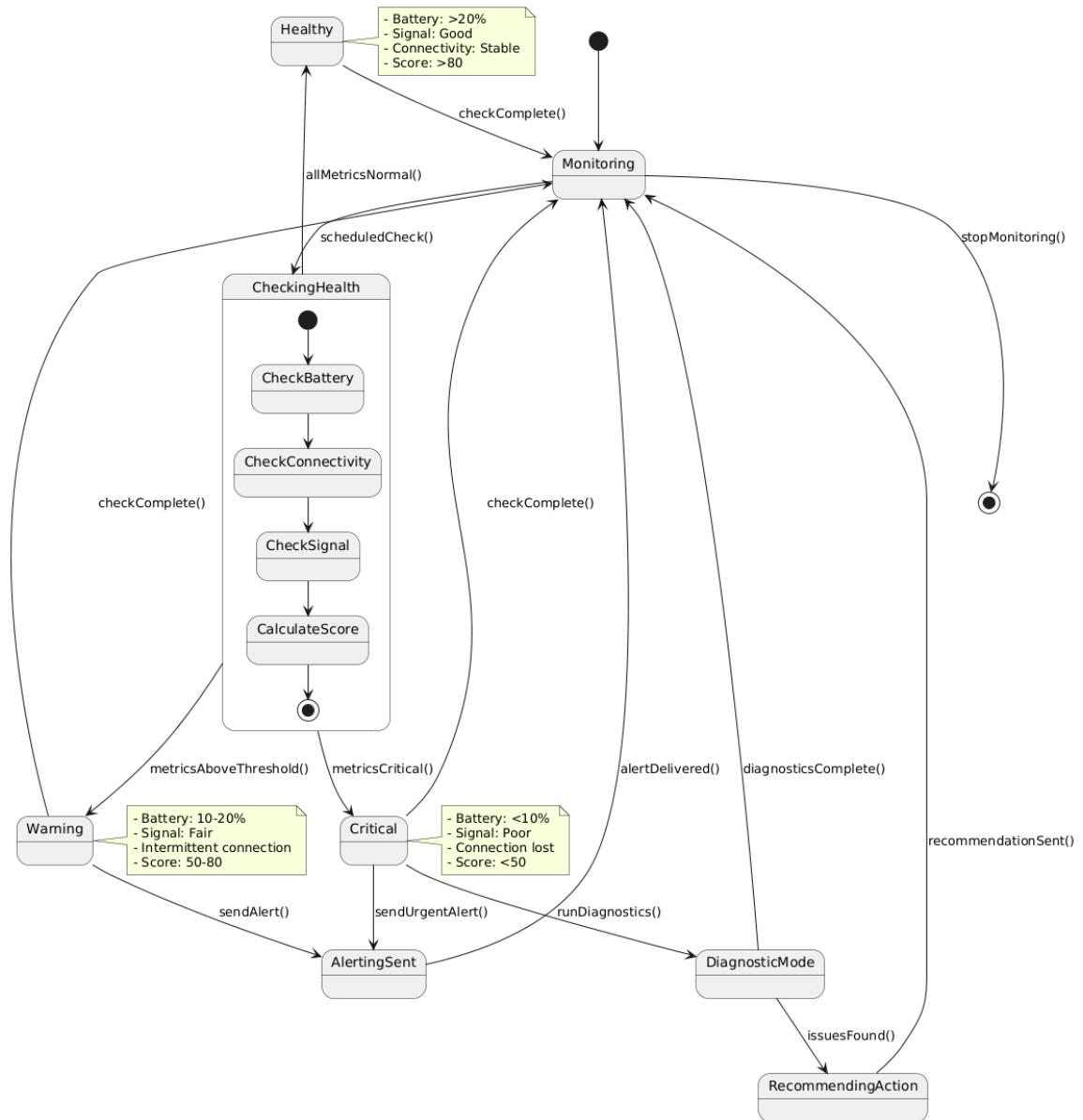
5. System Status Dashboard State Diagram



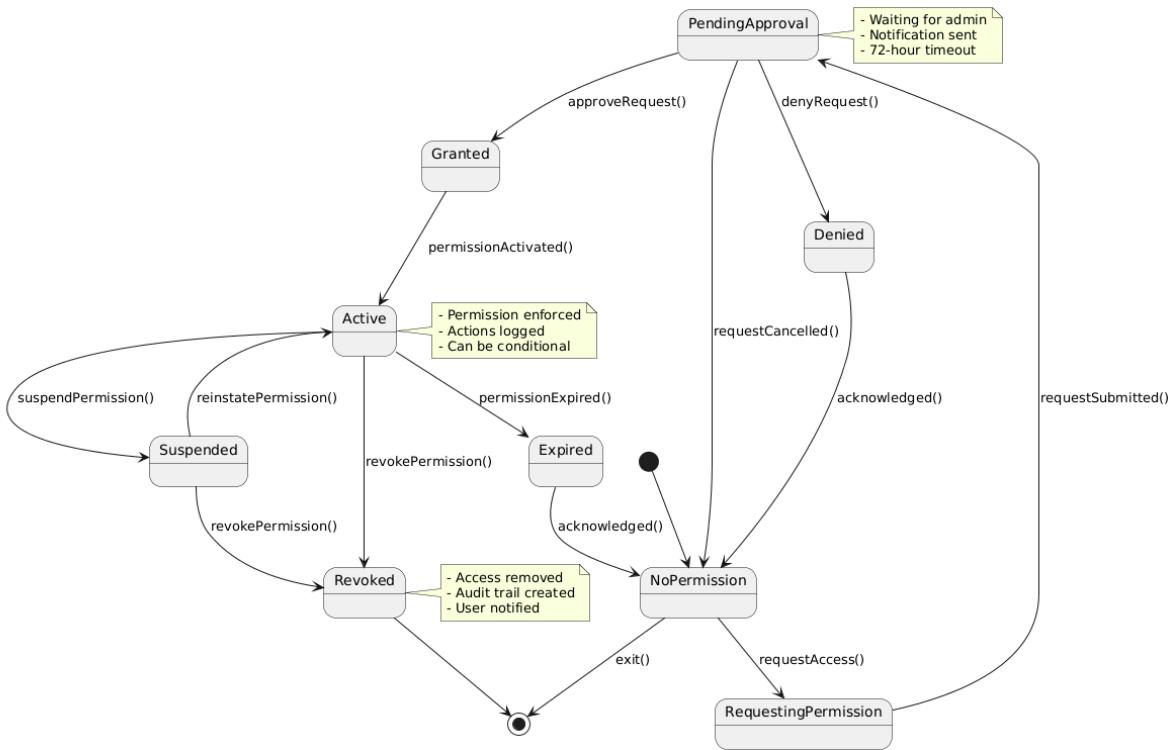
6. Activity Log State Diagram



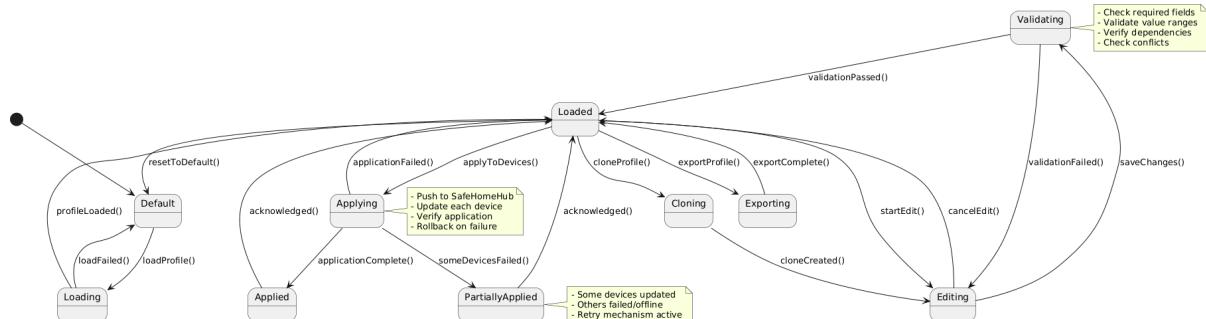
7. Device Health Monitor State Diagram



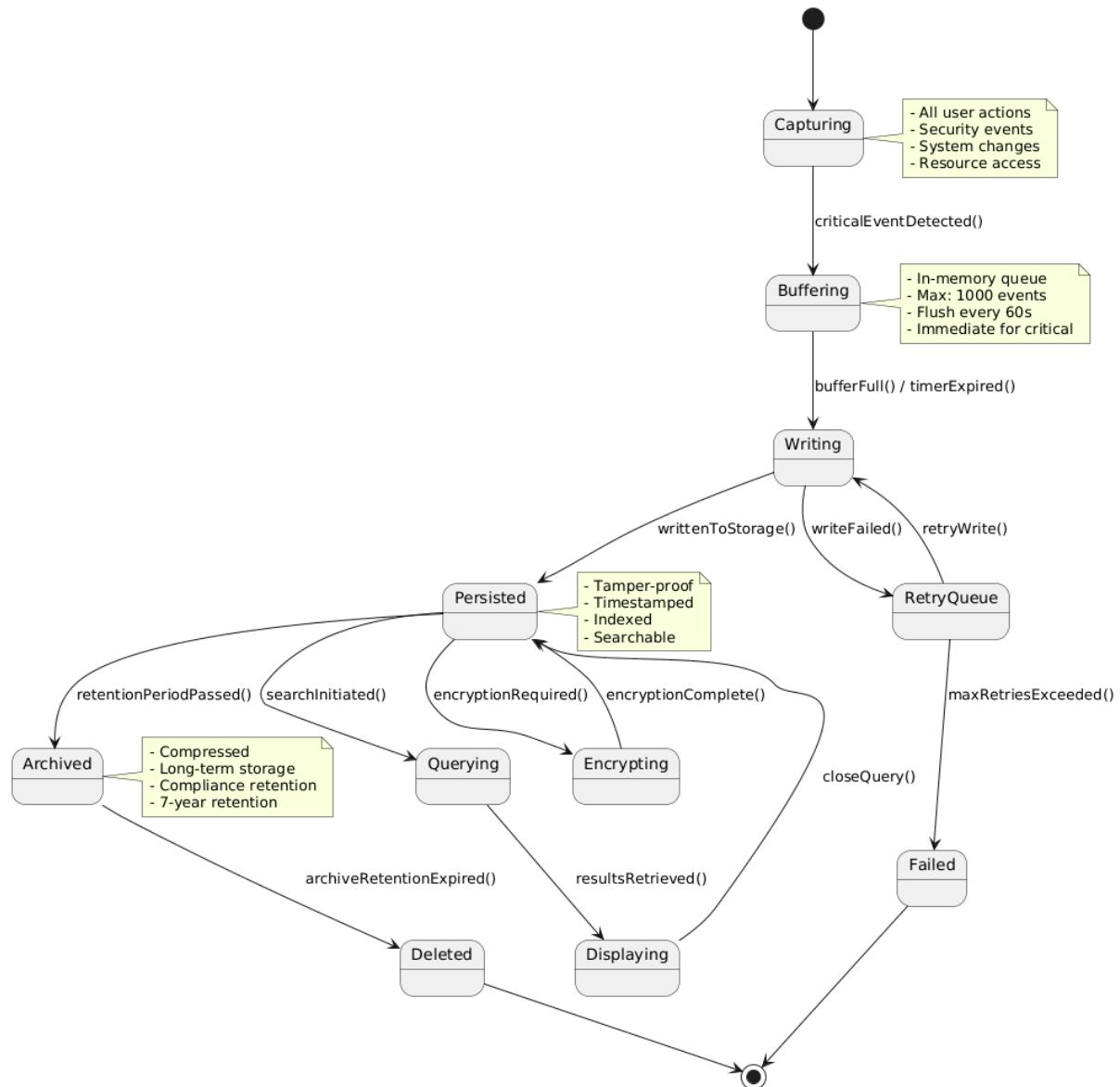
8. Permission Management State Diagram



9. Configuration Profile State Diagram

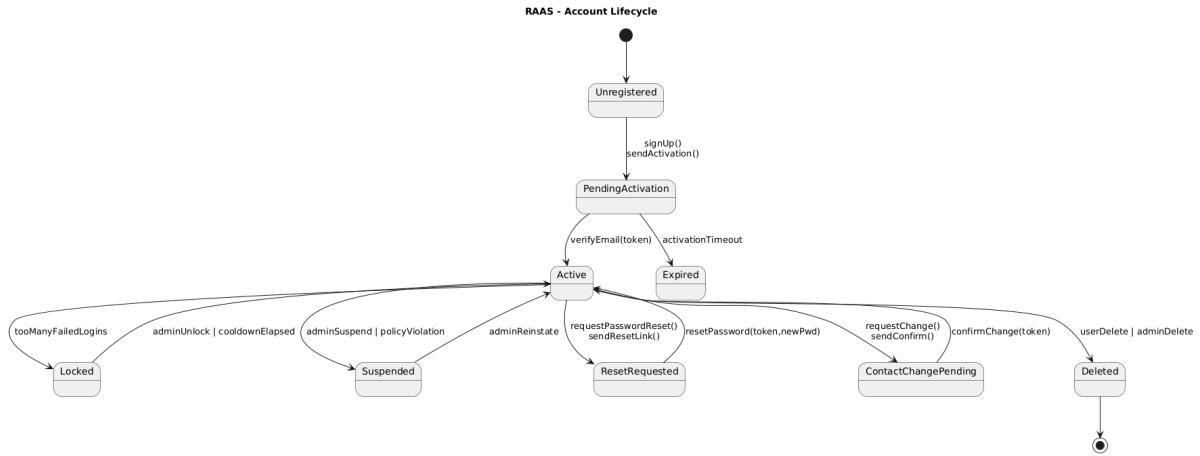


10. Audit Log State Diagram

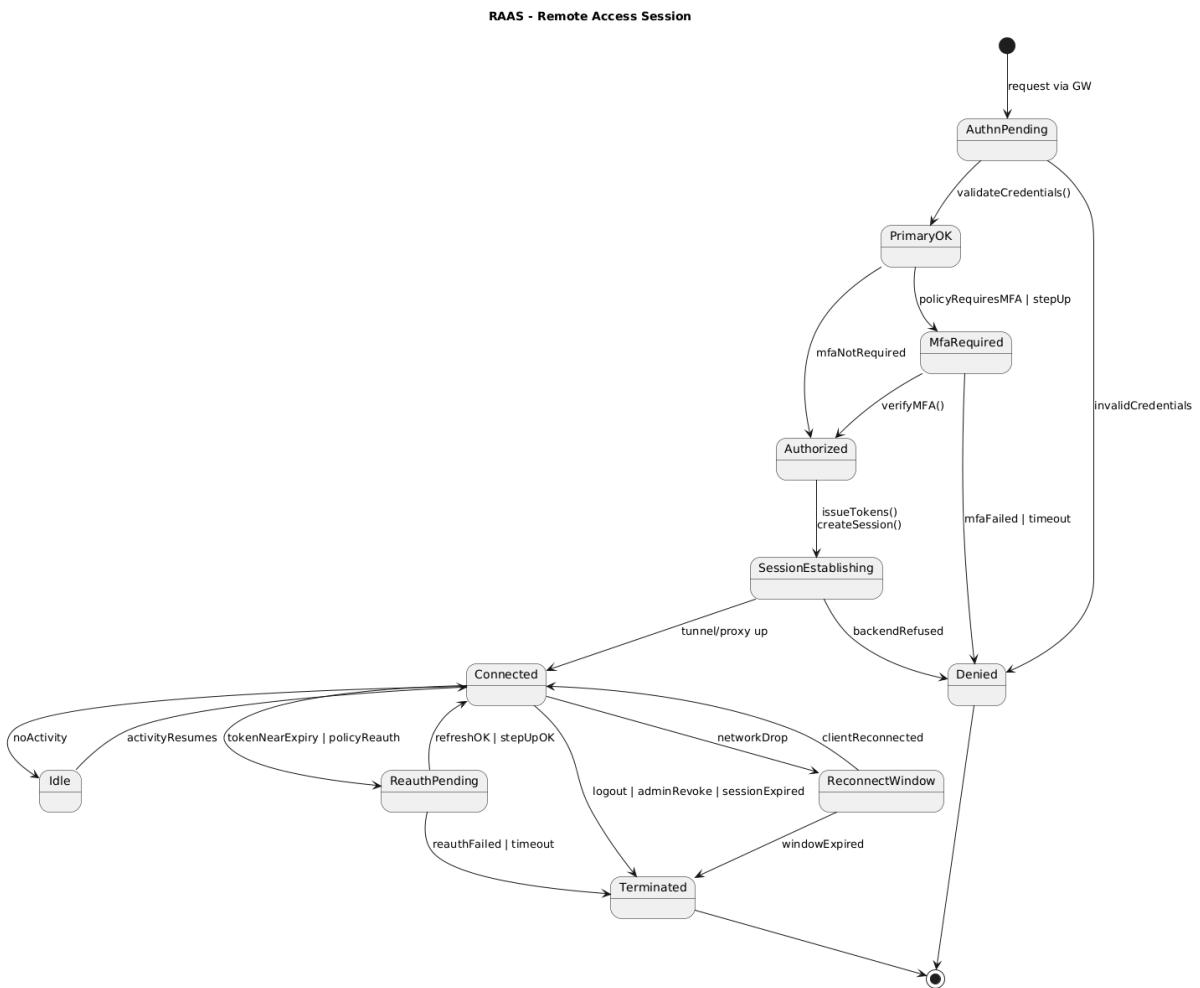


4. STATE DIAGRAM – REMOTE ACCESS AND ACCOUNT

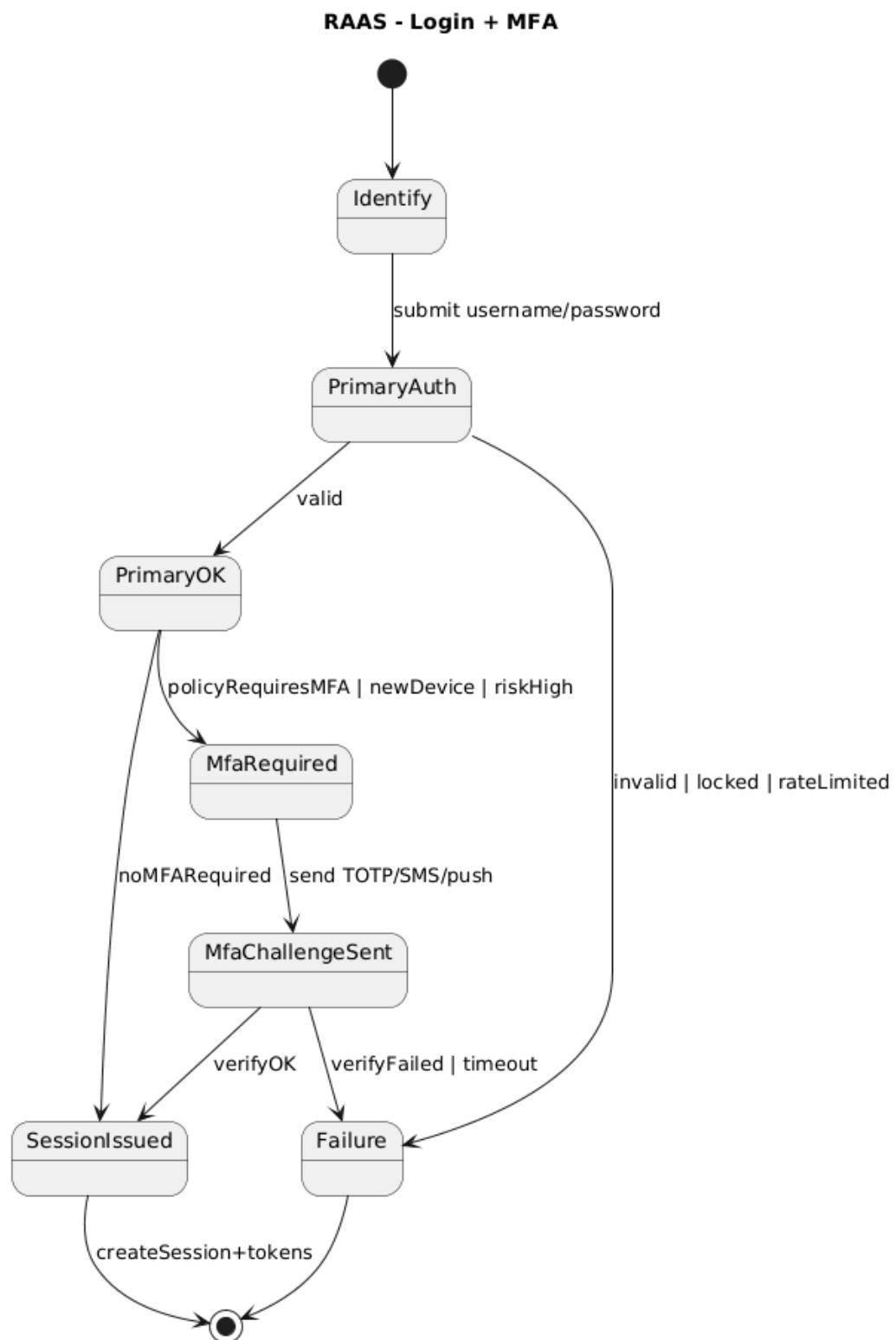
4.1 State Diagram - Account Lifecycle



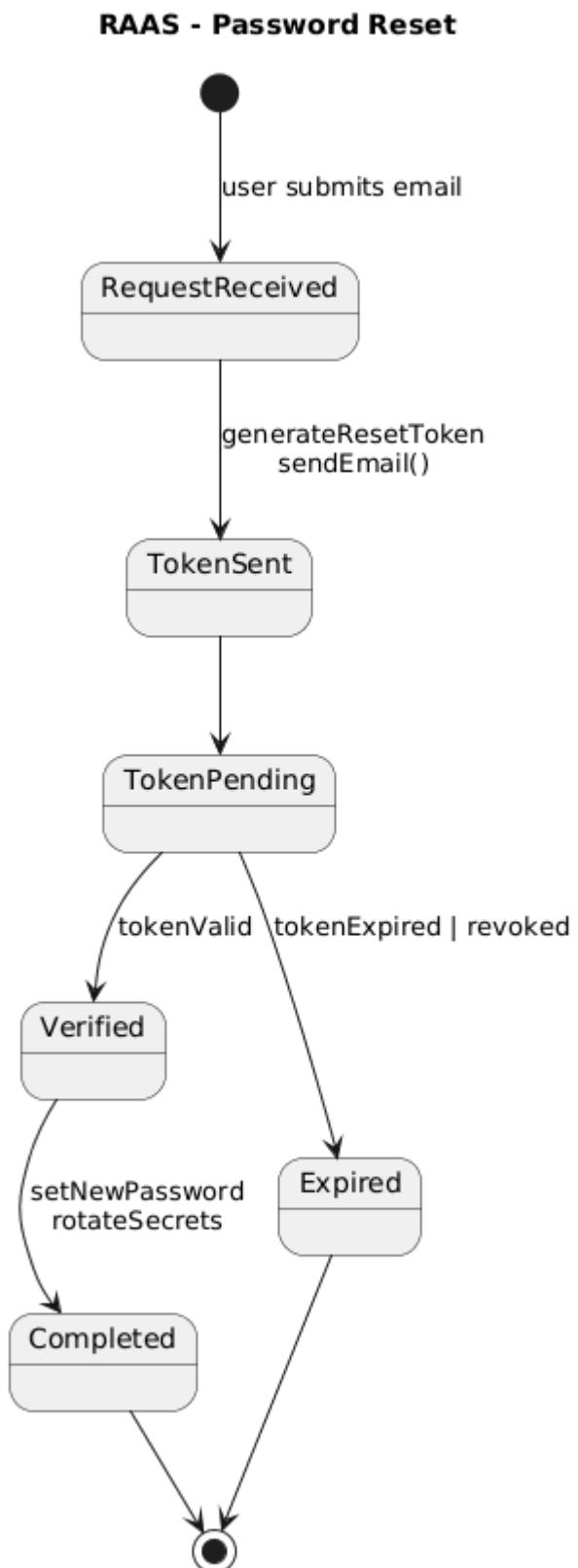
4.2 State Diagram - Remote Access session



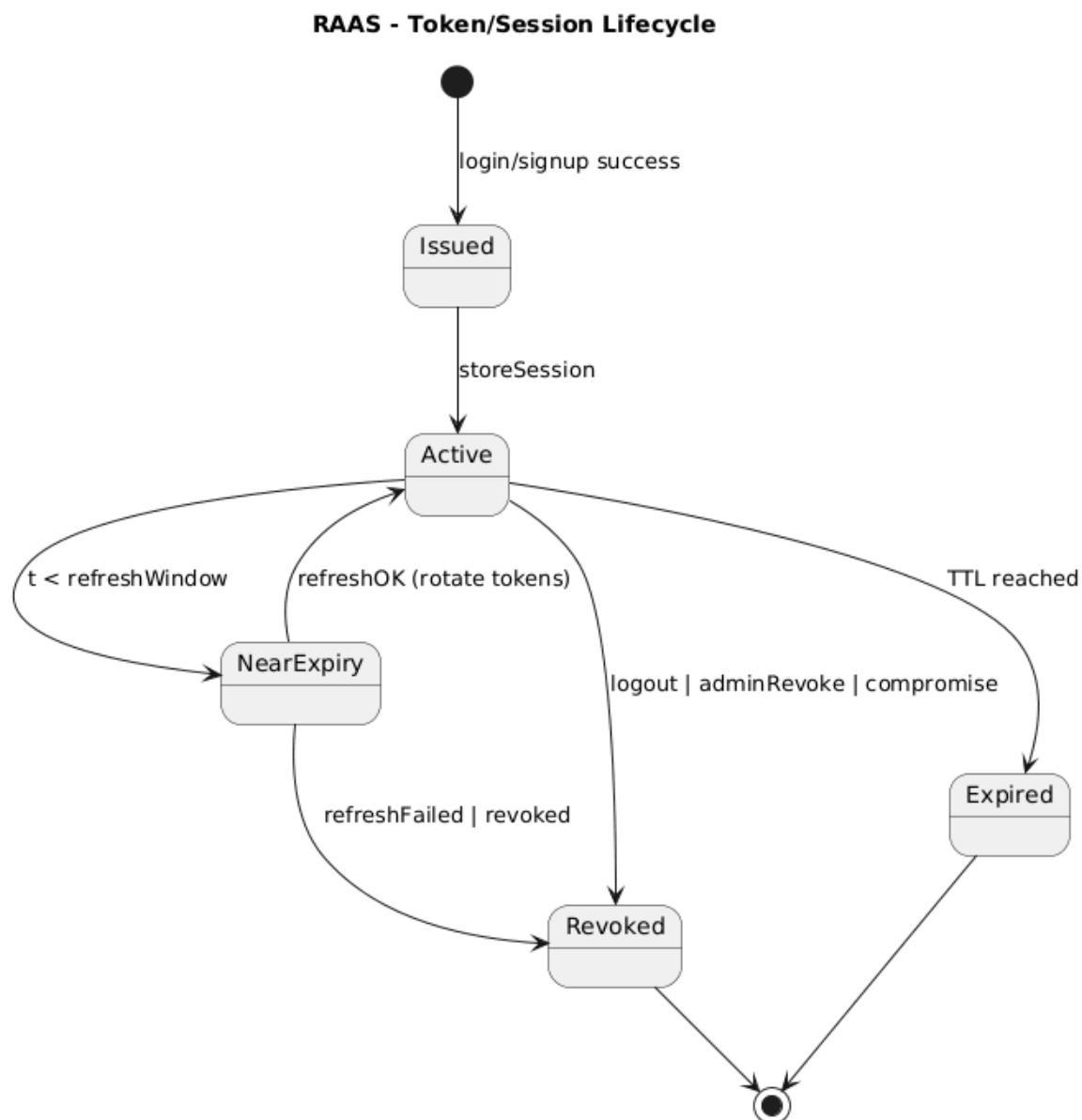
4.3 State Diagram - Login with MFA



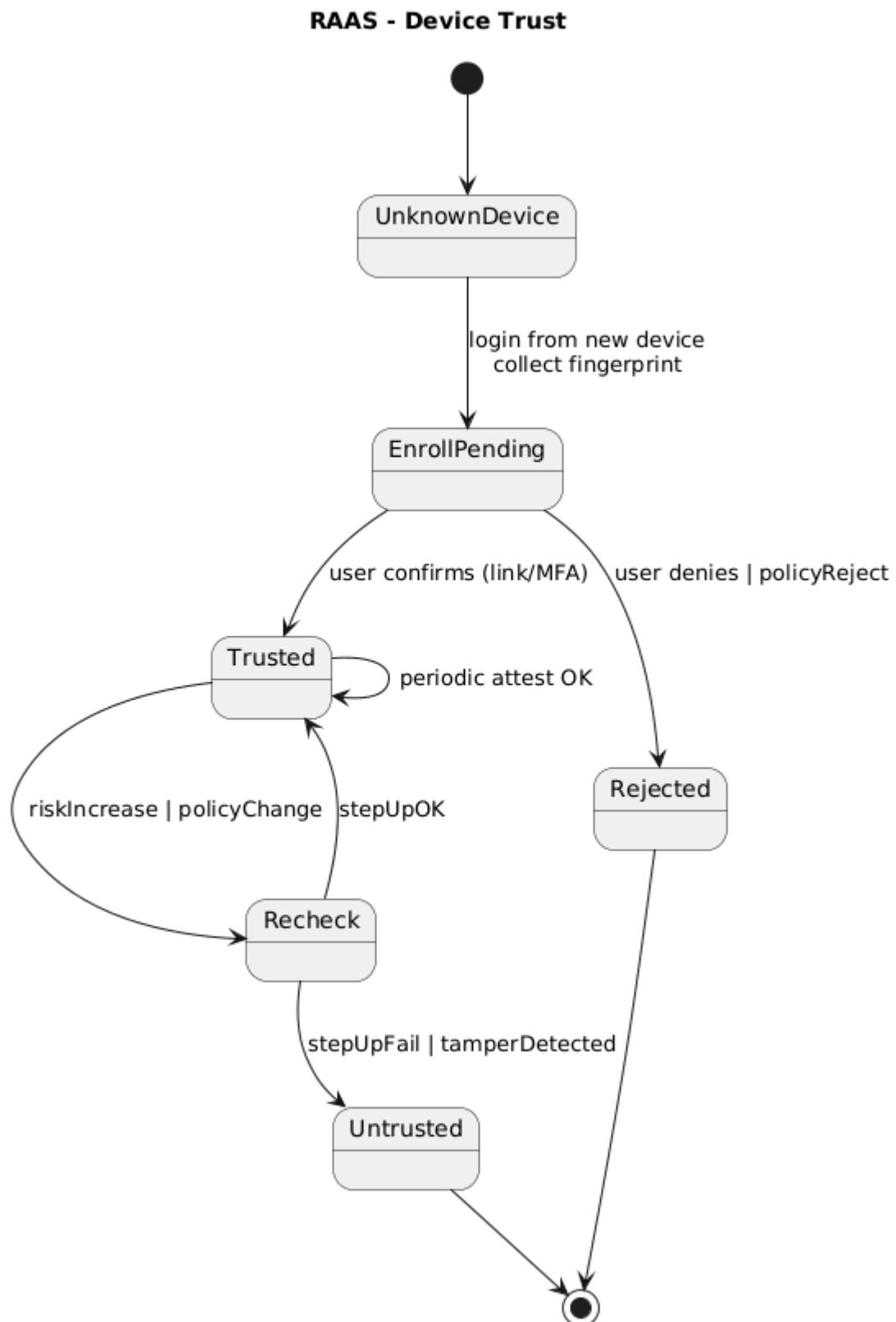
4.4 State Diagram - Password reset flow



4.5 State Diagram - Token / Session lifecycle

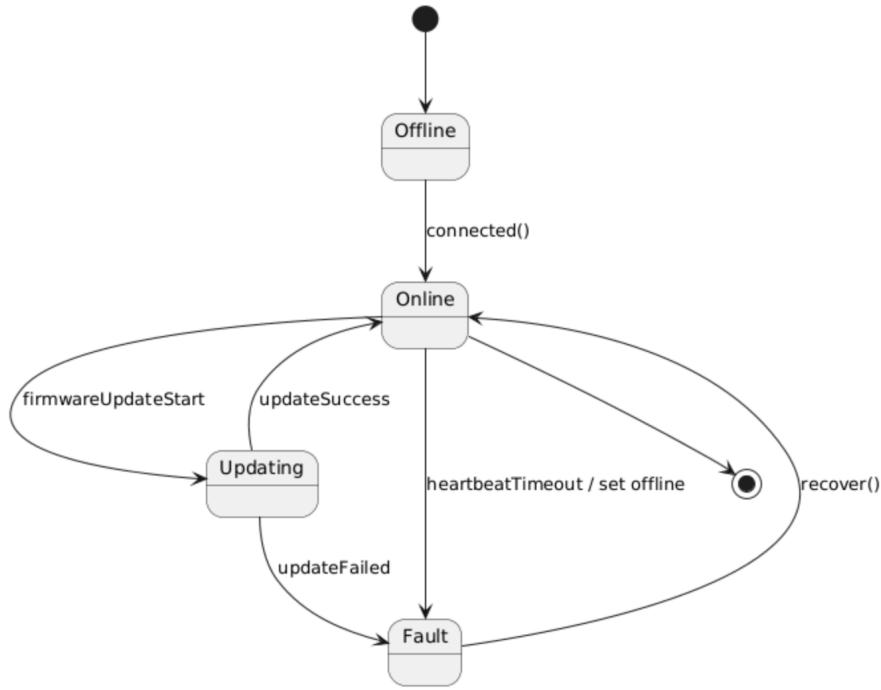


4.6 State Diagram - Device trust enrollment & check

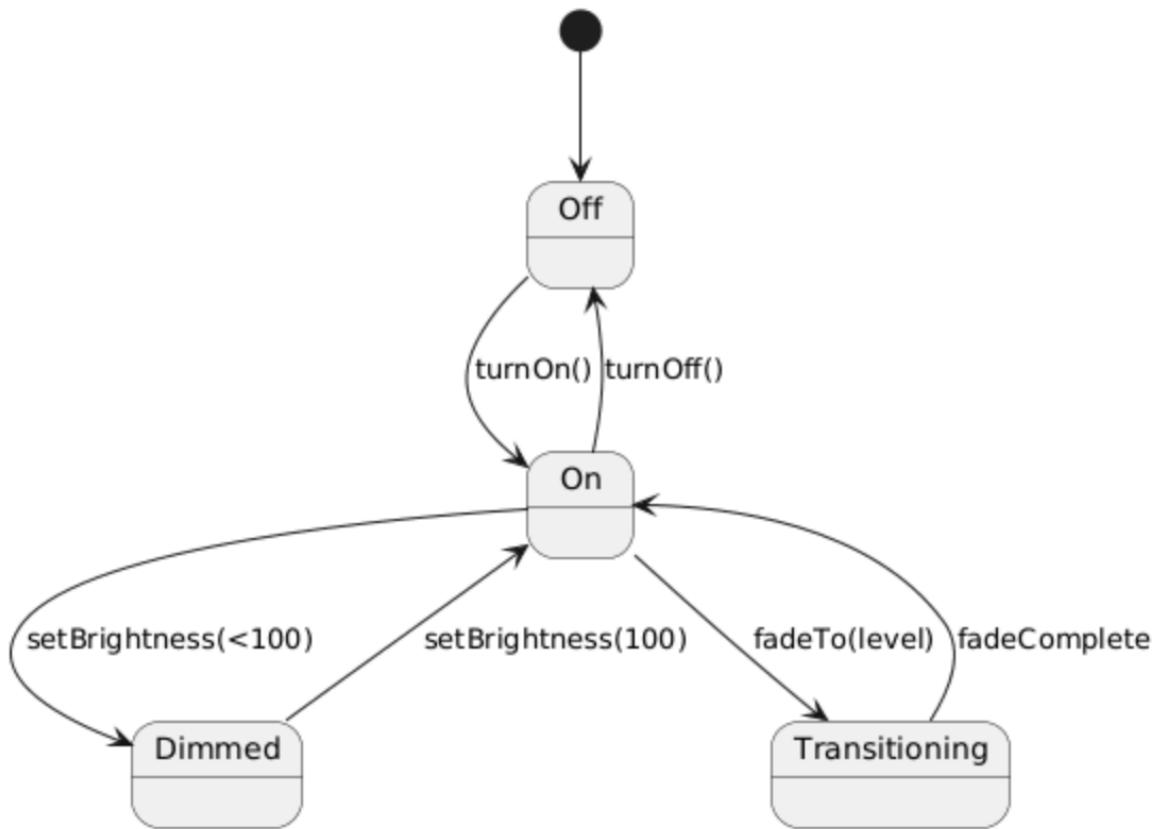


5. STATE DIAGRAM – INDOOR MONITORING AND DEVICE CONTROL

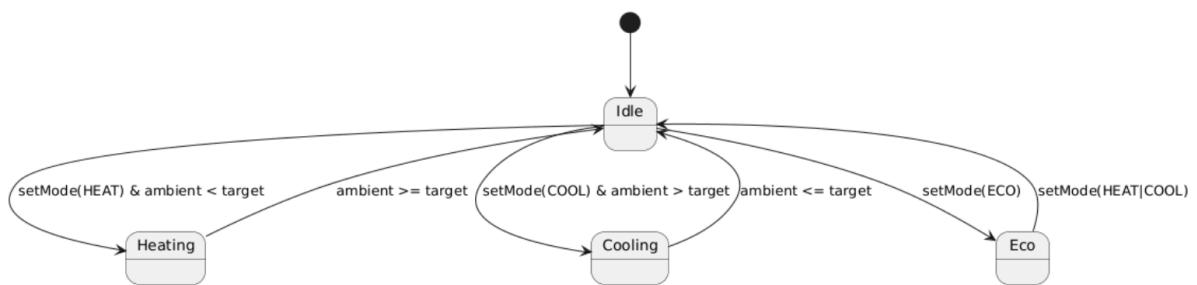
5.1 State Diagram - SmartHomeDevice (Lifecycle)



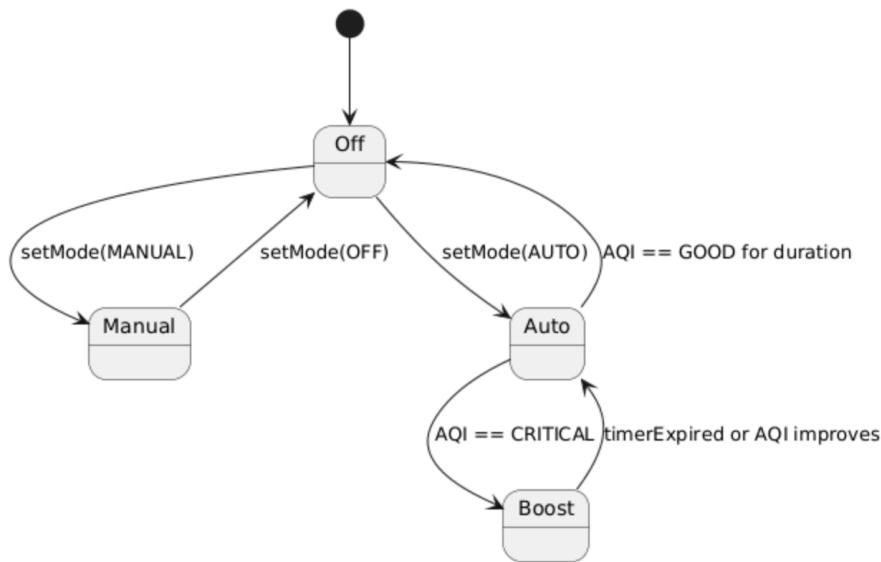
5.2 State Diagram - SmartLight (Behavior)



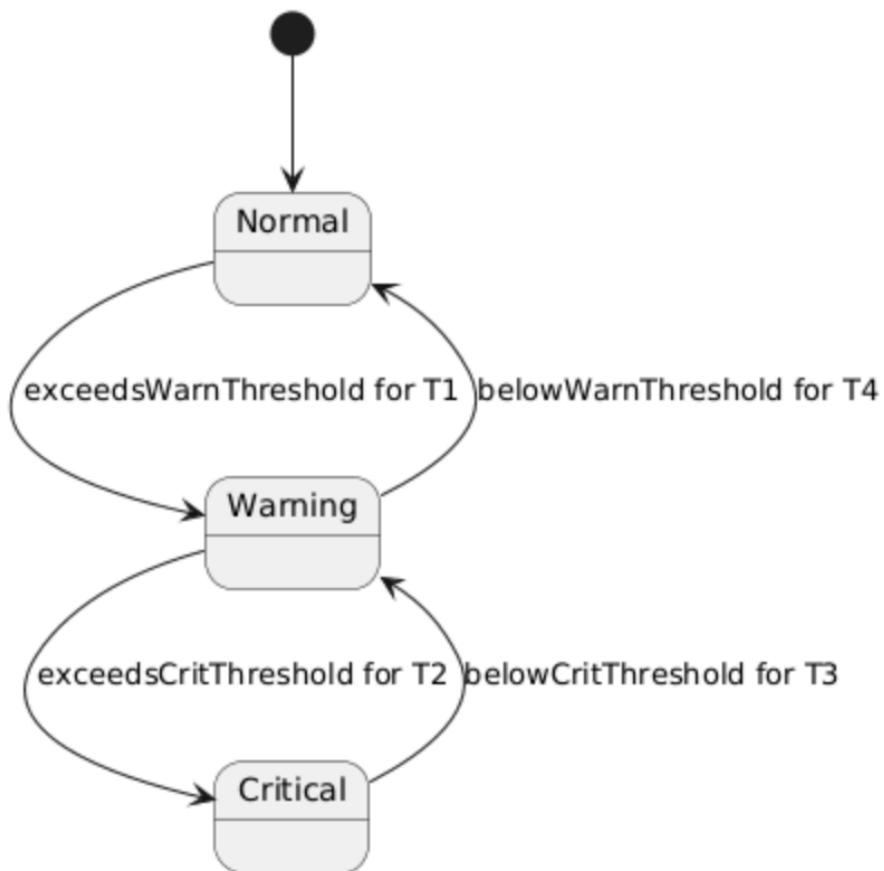
5.3 State Diagram - Thermostat (Mode/Control)



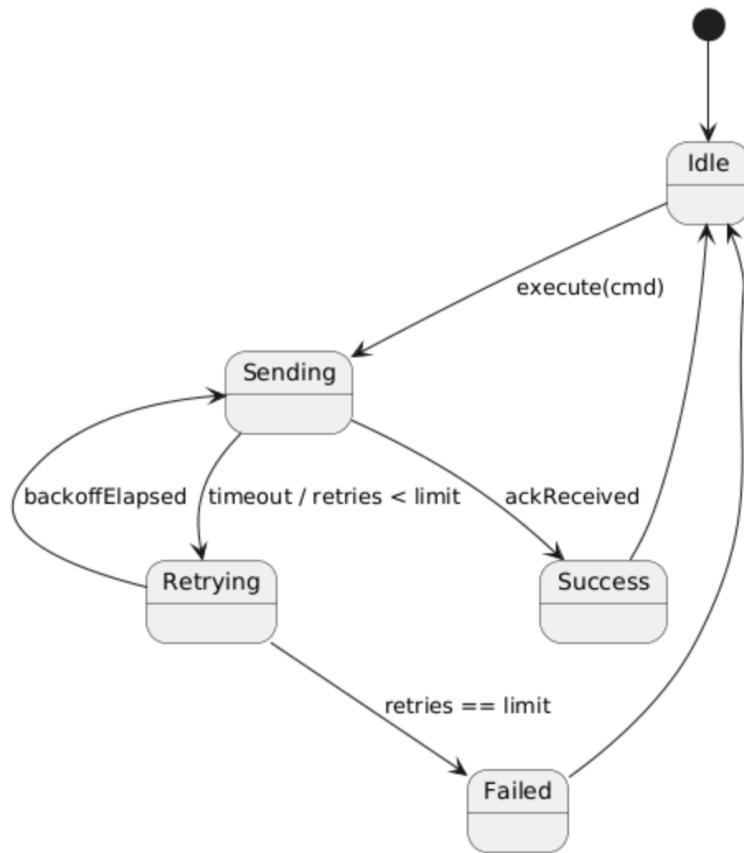
5.4 State Diagram - VentilationController (Control Logic)



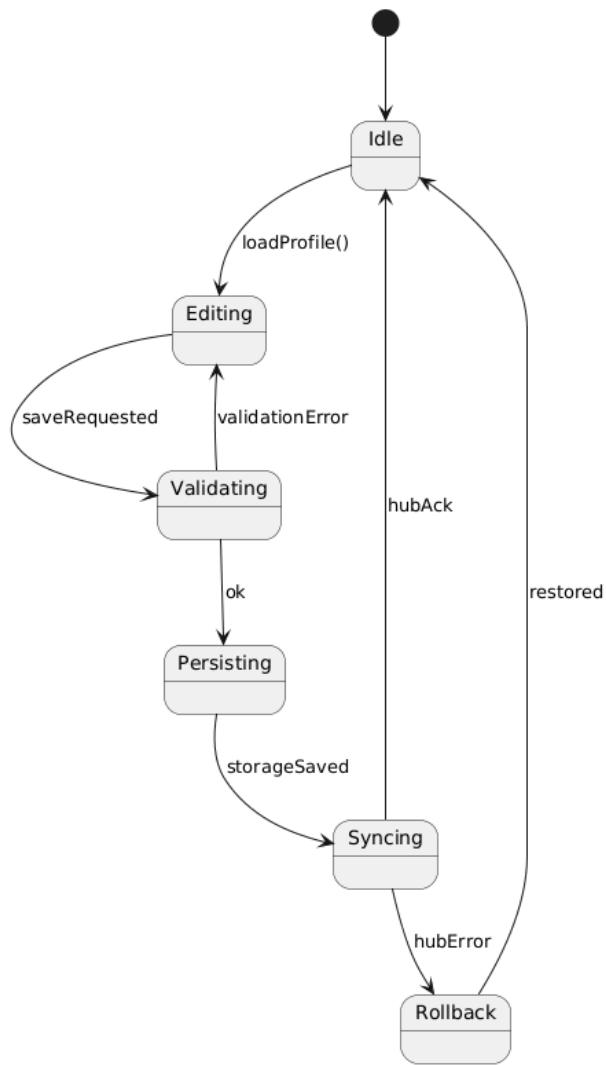
5.5 State Diagram - AirQualitySensor (Quality Bands with Hysteresis)



5.6 State Diagram - DeviceController (Command Execution)



5.7 State Diagram - ConfigurationManager (Apply Profile)



VI. Design Evaluation

1. Architectural Design Metric

a. Design Structure Quality Index (DSQI)

Given values:

- S1 = 82
- S2 = 35
- S3 = 25
- S4 = 16
- S5 = 5
- S6 = 5

- $S7 = 77$

DSQI components:

- $D1 = 1$
- $D2 = 1 - S2 / S1 = 1 - 35/82 = 47/82 \approx 0.573$
- $D3 = 1 - S3 / S1 = 1 - 25/82 = 57/82 \approx 0.695$
- $D4 = 1 - S5 / S4 = 1 - 5/16 = 11/16 = 0.688$
- $D5 = 1 - S6 / S4 = 1 - 5/16 = 11/16 = 0.688$
- $D6 = 1 - S7 / S1 = 1 - 77/82 = 5/82 \approx 0.061$

Weights:

- $W1 = 0.25$
- $W2 = 0.30$
- $W3 = 0.25$
- $W4 = 0.05$
- $W5 = 0.05$
- $W6 = 0.10$

Final DSQI:

DSQI

$$\begin{aligned} &= (W1 \cdot D1) + (W2 \cdot D2) + (W3 \cdot D3) + (W4 \cdot D4) + (W5 \cdot D5) + (W6 \cdot D6) \\ &= (0.25 \times 1) \end{aligned}$$

- (0.30×0.573)
- (0.25×0.695)
- (0.05×0.688)
- (0.05×0.688)
- (0.10×0.061)

$$= 0.25$$

- 0.1719
- 0.17375
- 0.0344
- 0.0344
- 0.0061

DSQI ≈ 0.671

1.2 Fenton's Simple Morphology Metrics

- Node count = 82
- Arc count = 71
- Size = node + arc = 153
- Depth = 5
- Width = 20
- Arc-to-node ratio = arc / node = 71 / 82 ≈ 0.866

2. CK Metrics

Depth of the inheritance tree	5
Maximum Number of Children	5 (Recording Manager)
Average Number of Children	0.429
Maximum Coupling Between Object classes	12 (at System class)
Average Coupling Between Object classes	1.543

3. MOOD Metric

a. MIF (Method Inheritance Factor)

Class	Md(Ci)	Mi(Ci)	Ma(Ci)
User	6	0	6
AdminHomeowner	5	6	11
Homeowner	4	6	10
Guest	3	6	9
Sensor	7	0	7
ContactSensor	3	7	10
ShockSensor	3	7	10
MotionSensor	3	7	10
SoundSensor	4	7	11
EnvironmentalSensor	6	0	6
FireSmokeSensor	3	6	9
COSensor	3	6	9
GasSensor	3	6	9
LeakSensor	3	6	9
SmartHomeDevice	5	0	5
SmartLight	3	5	8
Thermostat	4	5	9
SmartPlug	2	5	7
Camera	5	0	5
SafeHomeHub	8	0	8
CloudServer	7	0	7
Alarm	4	0	4
Incident	5	0	5
ActivityLog	6	0	6
Notification	5	0	5
SecurityMode	5	0	5
ConfigurationManager	6	0	6
Room	4	0	4
SafetyZone	4	0	4
Page	6	0	6
LoginPage	0	6	6
CameraListPage	0	6	6
SingleCameraViewPage	1	6	7
SystemStatusDashboard	5	0	5
DeviceManager	4	0	4

$$\text{MIF} = 97 / 237 = \text{Approximately } 0.41$$

b. CF (Coupling Factor)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	
1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	1	1	1	0	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	1	1	1	1	0	1	1	1	1	0	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	1	1	0	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	1	1	1	0	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	1	1	1	1	0	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	1	1	1	1	1	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	1	1	1	1	1	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	1	1	1	1	1	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	1	1	1	1	1	1	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	1	1	1	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

$$CF = 132 / (34 * 34 - 34) = 0.117$$

4. OO Metric Proposed by Lorenz and Kidd

Subsystem	Class	# of operation	# of attribute	NOA
Intelligent Security	User	3	4	0
Intelligent Security	AdminHomeowner	4	0	1

Intelligent Security	Homeowner	4	0	1
Intelligent Security	Guest	2	1	1
Intelligent Security	SafeHomeHub	7	0	0
Intelligent Security	CloudServer	5	2	0
Intelligent Security	MobileApp	6	0	0
Intelligent Security	Device (interface)	4	0	0
Intelligent Security	Sensor (abstract)	4	2	1
Intelligent Security	ContactSensor	3	1	1
Intelligent Security	ShockSensor	4	0	1
Intelligent Security	MotionSensor	3	1	1
Intelligent Security	EnvironmentalSensor (abstract)	4	1	1
Intelligent Security	FireSmokeSensor	4	0	2
Intelligent Security	COSensor	4	0	2
Intelligent Security	GasSensor	4	0	2
Intelligent Security	LeakSensor	2	1	2
Intelligent Security	SoundSensor	5	0	1
Intelligent Security	Siren	4	1	1
Intelligent Security	SmartDoorLock	3	2	1
Intelligent Security	SecurityMode	1	4	0
Intelligent Security	Alarm	2	3	0
Intelligent Security	Incident	3	3	0
Intelligent Security	AlarmVerification	5	0	0
Intelligent Security	AlarmPolicy	2	2	0
Intelligent Security	NotificationPolicy	2	1	0
Intelligent Security	CooldownTimer	3	1	0
Live Surveillance	Camera	7	0	1
Live Surveillance	VideoStream	2	1	0
System & User Management	ActivityLog	6	2	0
System & User Management	Notification	2	3	0
System & User Management	ConfigurationProfile	1	6	0
Remote Access and Account	LoginInterface	9	19	0
Remote Access and Account	LoginManager	3	0	0
Remote Access and Account	WebInterface	7	2	0
Indoor Monitoring & Device Control	SmartHomeDevice	3	1	1

5. Explanation of Metrics

Overall, the design metrics indicate that the SafeHome system achieves a strong balance between modularity, maintainability, and functionality. The DSQI score of approximately 0.671 reflects a well-structured architecture despite the system's large size, while the CK and MOOD metrics confirm low coupling, appropriate use of inheritance, and clear separation of subsystem responsibilities. The morphology metrics further demonstrate that the class hierarchy and interconnections remain manageable, supporting extensibility across intelligent security, surveillance, user management, and device control. Together, these results show that the design aligns well with the project goals of achieving high cohesion, low coupling, and reliable subsystem integration.

For future improvement, the system could benefit from reducing complexity in coordinator classes with higher coupling, introducing more interface abstractions to isolate subsystem boundaries, and refining inheritance hierarchies to minimize depth where possible. Additional automated analysis tools and performance profiling could also help validate scalability as more devices and features are incorporated into later increments.

VII. WHO DID WHAT

Team member Geunyeong
1. Draw Intelligent security part Architecture Diagram 2. Did CRC cards for intelligent security part 3. Draw Intelligent security part Class Diagram 4. Draw Intelligent security part State Diagram 5. Setup meeting room, Summarized Meeting logs
Team member Alan
1. Did overall architecture diagram 2. Did surveillance class diagram 3. Did CRC cards for surveillance 4. Did State Diagram for Surveillance 5. Completed MOOD metric 6. Added hyperlinks 7. Calculated Coupling factor and matrix
Team member Yernaz
1. Changed Remote Access and Account architecture diagram 2. Did Indoor Monitoring and Device Control architecture diagram 3. Changed Heading format for Remote Access and Account class diagram 4. Did Indoor Monitoring and Device Control class diagram 5. Did CRC cards for Remote Access and Account 6. Did CRC cards for Indoor Monitoring and Device Control 7. Did State Diagram for Remote Access and Account 8. Did State Diagram for Indoor Monitoring and Device Control 9. Made OO Metric Proposed by Lorenz and Kidd
Team member Arda
1. Responsible for maintaining System and User Management. 2. Partially Responsible for maintaining Remote Access and Account in

collaboration with Yernaz.

3. Added complimentary assumptions.
4. Created the architecture diagram for System and User Management
5. Created class diagram for System and User Management
6. Created CRC cards for System and User Management
7. Created State Diagrams for System and User Management
8. Created architecture diagram for Remote Access and Account
9. Created class diagram for Remote Access and Account
10. Calculated the Design Structure Quality Index (DSQI)
11. Calculated the Fenton's Simple Morphology Metrics
12. Wrote down the explanation of metrics.

VIII. MEETING LOGS

Raw logs: [HW2 meeting logs](#)

Meeting Log 1: November 5, 2025

- When: November 5, 2025
- Who (Attendees): Arda, Geunyeong, Alan, Yernaz
- Where: Library Meeting room 3rd floor
- Why (Objective): To review the feedback from the Homework 1 (SRS) document and establish a clear plan, task division, and set of standards for creating the Homework 2 (SDS) document.
- What (Discussion & Decisions):
 - HW1 Feedback Review: The team analyzed the feedback from their first assignment. Key areas for improvement were identified, including missing sequence diagrams (which were confused with swim lane diagrams), inconsistent formatting across use cases, missing references to the source material, and a "Who did what" section that lacked detail.
 - HW2 (SDS) Requirements: The team clarified the expectations for the new assignment based on the professor's lecture. All diagrams (Architecture, Class, CRC, State) must be created based on Team 1's SRS document. The team must create six architecture diagrams (one "overall" and one for each of the five main use case sections) and five state diagrams.
 - Tools & Consistency: To avoid the inconsistencies from HW1, the team agreed to use PlantUML for all diagrams. A central Google Doc will be created to log all class names (e.g., LogManager, Log) to ensure consistency.
 - Metrics (Design Evaluation): The team will calculate metrics (like the Coupling Factor matrix) during the final two days after all diagrams are completed. They will also need to write an analysis of these metrics, not just present the numbers.
 - Assumptions: The team will copy the assumptions from Team 1's SRS but will update them to reference the SRS document itself, rather than the original dialogs.
- How (Task Allocation):
 - Geunyeong: Intelligent Security
 - Alan: Live Surveillance
 - Arda: System and User Management
 - Yernaz: Indoor Monitoring and Device Control
 - Arda & Yernaz: Remote Access and Account (Shared)
 - All Team: Overview, Overall Architecture, and Design Evaluation
 - Deadline: All members agreed to try and complete their individual parts by Sunday/Monday (Nov 10/11) to leave time for a final group review, standardization, and metrics calculation before the Nov 14th deadline.

Meeting Log 2: November 11, 2025

- When: November 11, 2025
- Who (Attendees): Arda, Geunyeong, Alan, Yernaz
- Where: Library Meeting room 3rd floor
- Why (Objective): To conduct a mid-project status check, resolve design blockers, and coordinate the final push to complete the SDS document.
- What (Discussion & Decisions):
 - Status Updates:
 - Geunyeong (Intelligent Security) had finished his CRC, State, and Class diagrams.
 - Arda (SUM) and Alan (Live Surveillance) still had diagrams and formatting to complete.
 - "Remote Access" and "Indoor Monitoring" (Arda & Yernaz) were not yet complete.
 - Document Formatting (How):
 - Arda will replace the static table of contents with Google Docs' automatic, hyperlinked version.
 - All members *must* use the "Heading 1," "Heading 2," etc., styles for their titles so the automatic table of contents will work.
 - Blocker 1: Diagram Readability:
 - The team identified that the overall class diagram for "Intelligent Security" (Section 2.1) was too complex and unreadable.
 - Decision: They will ask the professor or TA for permission to use a simplified, high-level abstraction for the *overall* diagram.
 - Blocker 2: Referencing SRS 1:
 - The team was unsure if they needed to reference Team 1's SRS.
 - Decision: The TA confirmed via message that they must reference Team 1's SRS document.
 - Blocker 3: Duplicate Classes in CRCs:
 - Geunyeong noted the `SafeHomeHub` class appeared in two sections.
 - Decision: It is acceptable to have multiple CRC cards for the same class, as long as each card describes the class's role within a different context or use case. They will add explanations to the document.
 - Blocker 4: "Safety Zone" Concept:
 - Geunyeong was confused about the "Safety Zone" concept mentioned by the professor.
 - Decision: The team clarified it as a feature for grouping sensors in an area to arm/disarm them all at once.
- How (Updated Plan):
 - Geunyeong's Schedule: Geunyeong informed the team he could not attend the Thursday meeting due to personal commitments.
 - New Deadline: All CRC cards must be finished by Wednesday (Nov 12) at midnight.
 - Final Steps: The team will meet on Thursday to assemble the final document, check for consistency, and format everything for submission.

Meeting Log 3: November 13, 2025

- When: November 13th
- Who (Attendees): Arda, Alan, Yernaz (Geunyeong was absent).
- Where: Library 4th floor meeting room
- Why (Objective): To conduct a final "gap analysis" of the SDS document, identify all remaining missing or incomplete sections, and assign the final tasks (like metrics calculation) for completion.
- What (Discussion & Decisions):
 - Document Status Check: Arda led the team through the table of contents to review the document's completeness.
 - Missing Items: They identified that the "System and User Management" (SUM) Architecture Diagram and the State Diagram were still missing.
 - Revisions Needed: The "Intelligent Security" class diagram (Geunyeong's part) needed to be revised for readability. The "Whole System Overview" class diagram was just a blurry template image and needed to be recreated.
 - Formatting Needed: The "SUM" CRC cards were present but needed to be properly formatted.
 - TA Feedback Implementation: The team confirmed the action items from the TA's feedback.
 - They reaffirmed that the complex "Intelligent Security" diagram must be simplified into a high-level version.
 - They confirmed that having multiple CRC cards with the same class name (e.g., SafeHomeHub) is acceptable as long as the context is different.
 - Final Task Allocation:
 - Metrics (Design Evaluation): The team divided the calculation tasks. Arda volunteered for the MOOD metric, and Alan took the DSQI and Fenton's metrics.
 - 'Who did what': They discussed how to write this section to reflect the professor's feedback, ensuring they highlighted collaboration (e.g., Arda and Yernaz shared "Remote Access") rather than having one person do one entire part.
 - 'Assumptions': The plan is to review Team 1's 44 assumptions (deleting or editing as needed) and then add their own new assumptions (starting from #45) related to their design choices.
- How (Final Polish Actions):
 - Add References: The team agreed to add references under their diagrams pointing to the specific pages in the Team 1 SRS document for better traceability.
 - Add Hyperlinks: Arda will add internal document hyperlinks (e.g., from architecture sections to their corresponding class diagrams) to improve navigation.
 - Meeting Logs: The team confirmed they would summarize their meeting logs and add them to the document.
 - TA Gratitude: The team made a specific note to thank the TA (Sejong Jang) in the document for his guidance.

APPENDIX A. GLOSSARY

This glossary defines key terms and classes used throughout the SafeHome Software Design Specification to ensure clarity and consistency.

- **CloudServer:** The central backend system responsible for user authentication, data synchronization, remote access, and processing emergency escalations. It acts as the bridge between the SafeHome Hub and user-facing applications (MobileApp, WebInterface).
- **ConfigurationProfile:** A data object that stores the complete system configuration for a user's home. This includes all security modes, safety zones, device settings, and notification policies.
- **CRC Card:** Stands for Class-Responsibility-Collaborator. A design tool used to define and document the responsibilities of a class and its relationships with other classes.
- **Incident:** An object representing a detected security or hazard event that requires processing. It maintains a status (e.g., PENDING, VERIFIED, DISMISSED) and links to the triggering sensor and any associated alarms.
- **SafeHomeHub:** The central control unit installed in the home. It manages all connected devices (sensors, cameras) locally, processes sensor events, executes security mode transitions, and communicates with the CloudServer.
- **Security Mode:** A set of rules defining which sensors are active and what actions to take based on the system's state (e.g., Home, Away, Sleep). It maintains the list of active sensors and entry/exit delay timings for that specific mode.

UML and Design Terminology

Class Diagram A static structure diagram that shows the system's classes, their attributes, methods, and relationships between classes. Used to visualize the object-oriented design of the SafeHome system.

- Related Sections: Section 2 (Class Diagrams)
- Reference: All subsections 2.1-2.5

State Diagram A behavioral diagram that shows the different states an object can be in and the transitions between those states. Used to model the lifecycle and behavior of key SafeHome classes.

- Related Sections: Section 3 (State Diagrams)
- Reference: All subsections 3.1-3.5

Sequence Diagram A behavioral diagram showing how objects interact with each other over time, illustrating the order of messages exchanged to accomplish a specific use case.

- Related Sections: Already defined in Team 1 SRS
- Reference: Team 1 SRS Section VII

Architecture Diagram A high-level structural diagram that shows the major components of the system and how they interact. SafeHome uses a layered architecture with hardware, business logic, and cloud service layers.

- Related Sections: Section 1 (Architecture)

- Reference: Subsections 1.1-1.6

CRC Card (Class-Responsibility-Collaborator) A design tool that identifies a class's responsibilities and which other classes it collaborates with to fulfill those responsibilities.

- Related Sections: Section 4 (CRC Cards)
- Reference: All subsections 4.1-4.5

Design Patterns and Principles

Observer Pattern A behavioral design pattern used in SafeHome where sensors and devices notify the hub of state changes, and the hub notifies the cloud server and user interfaces.

- Implementation: SensorMonitor, CameraManager
- Reference: Class diagrams for Intelligent Security

Singleton Pattern A creational pattern ensuring only one instance exists. Used for SafeHomeHub and SecurityManager to maintain consistent system state.

- Implementation: SafeHomeHub, SecurityManager
- Reference: System architecture

Factory Pattern Used for creating different types of sensors and devices dynamically during device registration.

- Implementation: DeviceManager
- Reference: UC 3.1.1 (Add and Configure New Devices)

State Pattern Encapsulates varying behavior based on system state (Armed, Disarmed, Arming). SecurityManager uses this pattern.

- Implementation: SecurityManager, SecurityMode
- Reference: State diagrams for Security modes

Coupling and Cohesion

Coupling The degree of interdependence between software modules. SafeHome design aims for loose coupling between subsystems to improve maintainability.

- Measurement: Section 5 (Design Evaluation)
- Reference: Coupling metrics table

Cohesion The degree to which elements within a module belong together. High cohesion is achieved by grouping related functionalities (e.g., all camera operations in CameraManager).

- Measurement: Section 5 (Design Evaluation)
- Reference: Cohesion metrics table

Fan-in / Fan-out

- **Fan-in:** Number of classes that use a given class (measure of reusability)

- **Fan-out:** Number of classes used by a given class (measure of complexity)
- Reference: Section 5 (Design Evaluation)

Architecture Components

Business Logic Layer The layer containing core system functionality through manager classes (SecurityManager, CameraManager, etc.). Sits between the hardware layer and presentation layer.

- Reference: Architecture diagrams, Section 1

Presentation Layer The user-facing layer consisting of mobile application and web interface components.

- Reference: Architecture diagrams, MobileApplication class

Data Access Layer The layer responsible for communication with the cloud server and storage repository.

- Reference: SafeHomeCloudServer, StorageRepository classes

Hardware Abstraction Layer The SafeHomeHub acts as an abstraction layer, providing a unified interface to control diverse hardware devices.

- Reference: SafeHomeHub class, Device management

Technical Implementation Terms

PlantUML A text-based UML diagramming tool used to create all diagrams in this design document for consistency and version control.

- Usage: All UML diagrams in this document
- Reference: Meeting logs (2025.11.11)

Idempotency The property that ensures repeated operations produce the same result. Critical for command reliability in distributed systems.

- Implementation: Cloud server commands
- Reference: Team 1 SRS UC 4.1.1 (Sign Up)

Session Management The process of maintaining user authentication state across multiple requests through session tokens.

- Implementation: AccountManager, MobileApplication
- Reference: Team 1 SRS UC 4.1.2 (Log In), UC 4.1.3 (Log Out)

Event-Driven Architecture An architectural pattern where system components react to events (sensor triggers, user commands). Central to SafeHome's operation.

- Implementation: SensorMonitor, IncidentManager
- Reference: All event processing flows

APPENDIX C: Design Clarifications

This section documents key questions and clarifications that arose during the design process, primarily concerning the interpretation of the SRS and the scope of the design.

1. Regarding Duplicate Classes in CRC Cards (e.g., SafeHomeHub):

- Issue: The `SafeHomeHub` class appears in both "Intelligent Security" (CRC 1.8) and "Live Surveillance" (CRC 2.5) with different responsibilities and collaborators.
- Decision: This is an intentional design choice. The `SafeHomeHub` is a central, complex component with multiple facets. Rather than creating one oversized and unreadable CRC card, we have created separate cards that describe the responsibilities of the `SafeHomeHub` *within that specific context* (e.g., its security-related roles vs. its video-streaming roles).

2. Regarding Class Diagram Complexity (e.g., Intelligent Security Overall):

- Issue: The "Intelligent Security - Complete Class Diagram" (Section 2.1) is highly complex due to the large number of classes (36) in that subsystem.
- Decision: We have provided this complete diagram for thoroughness but have also included simplified, broken-down class diagrams for each sub-module (Sensor Monitoring, Incident Management, Security Mode Control) to improve readability and comprehension. We believe this multi-level approach best serves both completeness and clarity.