Vivekanand Education Society's Education
Institute of Technology

Project presentation on IoT
Application

Forest Fire Detection System

UNDER THE GUIDANCE OF
Dr. Abhay Kshirsagar

The Team

NAME	ROLL NO
Nishtha Joshi	25
Anket Kadam	26
Anushka Karhadkar	27
Payoshni Khekale	28
Kaushik Kotian	29
Tejaswi Koul	30
Priyank Kulkarni	31
Aniket Mahajan	32

Key Requirements should include

Real-time Monitoring

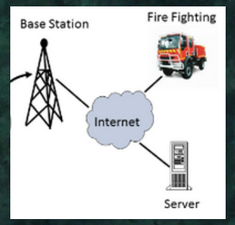
Utilizes sensors, satellite images, and drones for continuous observation to quickly identify fire outbreaks.

Wide Area Coverage

Effective over large forested regions, leveraging various sensor types to enhance reach and effectiveness.

High Accuracy

Reduces false alarms caused by weather conditions, ensuring efficient allocation of firefighting resources.



Data Integration

Combines data from temperature sensors, wind patterns, and satellite imagery for comprehensive fire risk analysis.

Challenges and Problems in existing systems

Environmental and Wildlife Impact:

Some detection systems may unintentionally harm wildlife or disrupt ecosystems due to the installation of physical monitoring devices.

Slow Communication with Authorities:

Delays in alerting authorities, especially in remote areas, can worsen fire situations as timely intervention is crucial for suppression.

Poor Integration of Technologies

Many existing systems use standalone technologies that struggle with cloudy conditions or dense foliage, reducing detection effectiveness.

Delayed Detection

Fires are often detected late, allowing them to grow unchecked and become more dangerous.

False Alarms

High rates of false alarms from dust or fog lead to resource wastage and decreased trust in systems.

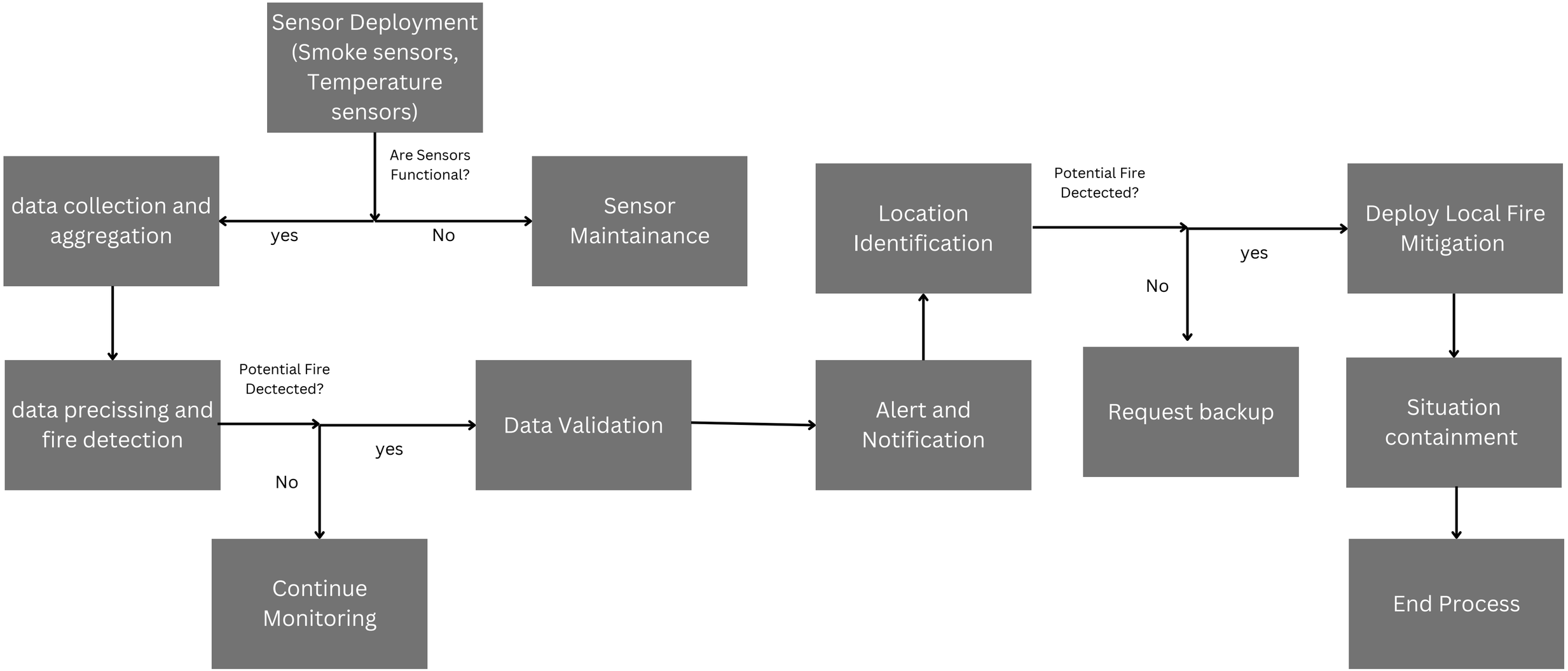
Partial Coverage

Limited reach in remote areas due to infrastructural and cost barriers hampers effective monitoring.

Process Specification

- **Sensor Deployment:** Sensors (temperature, smoke, humidity) placed across the forest. Calibration minimizes false alarms.
- **Fire Detection:** Machine learning detects fires, filtering out false alarms like fog or dust.
- **Data Collection:** Sensors send data to a central hub. Satellite and drone feeds enhance coverage.
- **Alert System:** Monitors temperature.
- **Monitoring:** A live dashboard tracks fire status and risk zones.
- **Response:** Teams are dispatched, and the system monitors fire spread in real-time.

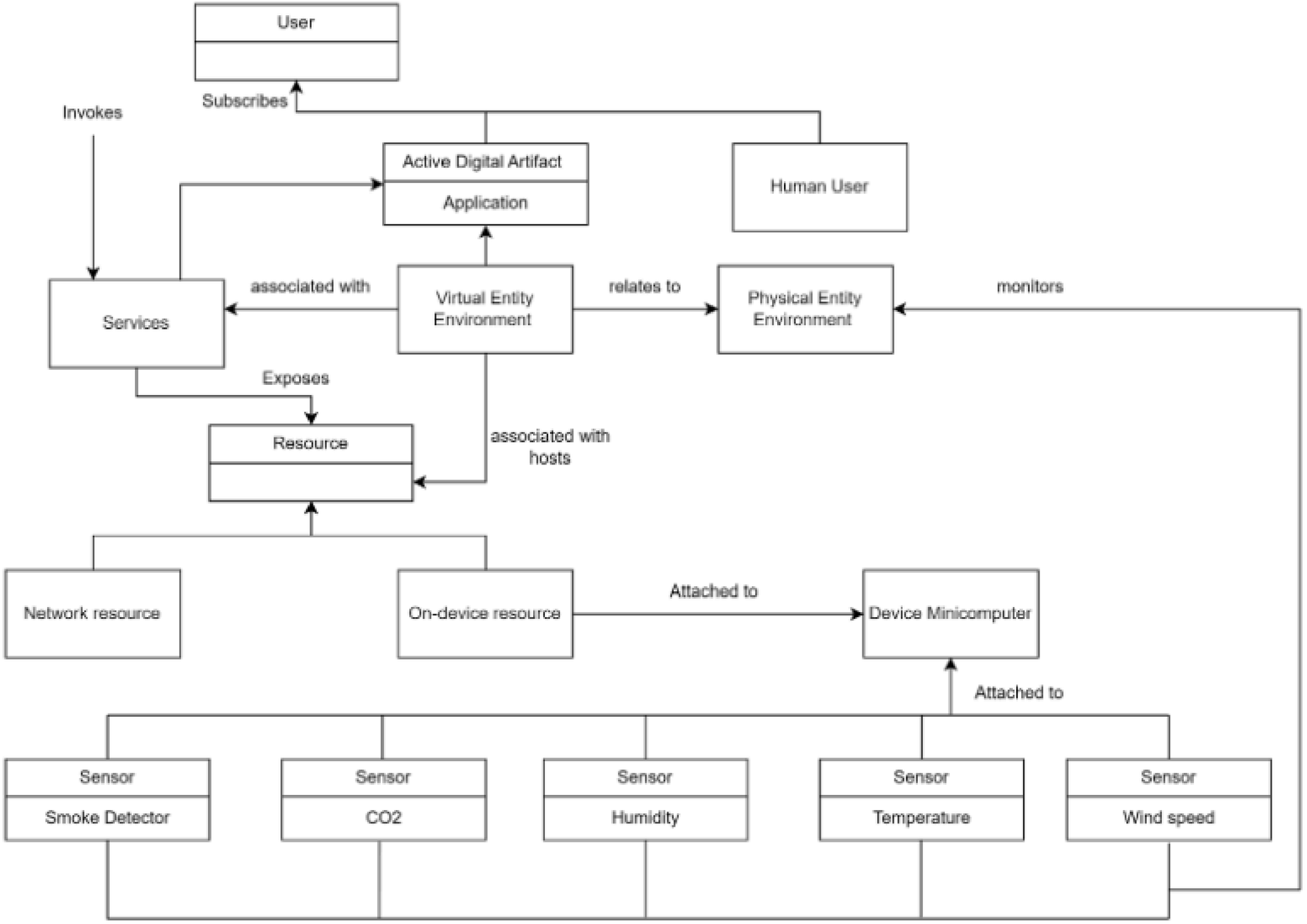
Process Specification



Domain Model Specification

- **Sensors** transmit data to the Sensor Network.
- Data flows to the **Central Monitoring System** for processing.
- **Fire Detection Algorithm** identifies fire risks.
- **Notification Service** alerts emergency responders.
- **Predictive Analytics Engine** forecasts future fire risks.

Domain Model Specification



INFORMATION MODEL SPECIFICATION

- **Virtual Entities:** Sensors (DHT22, MQ7, LM35) and drones are key entities in monitoring.
- Sensors:
- **DHT22:** Measures humidity.
- **MQ7:** Detects smoke levels.
- **LM35:** Monitors temperature.
- **Drones:** Provide aerial data with cameras and thermal sensors to support ground sensors.
- **Attributes:** Sensor type, measurement units, and operational status help assess fire risks.
- **Interconnectivity:** Ground sensors and drones work together to improve fire detection.
- **Data Flow:** Sensors send data to Raspberry Pi (edge device), which forwards it to AWS IoT Core.
- **Event Triggers:** Sensor thresholds (e.g., LM35 temperature limit) activate fire alerts.

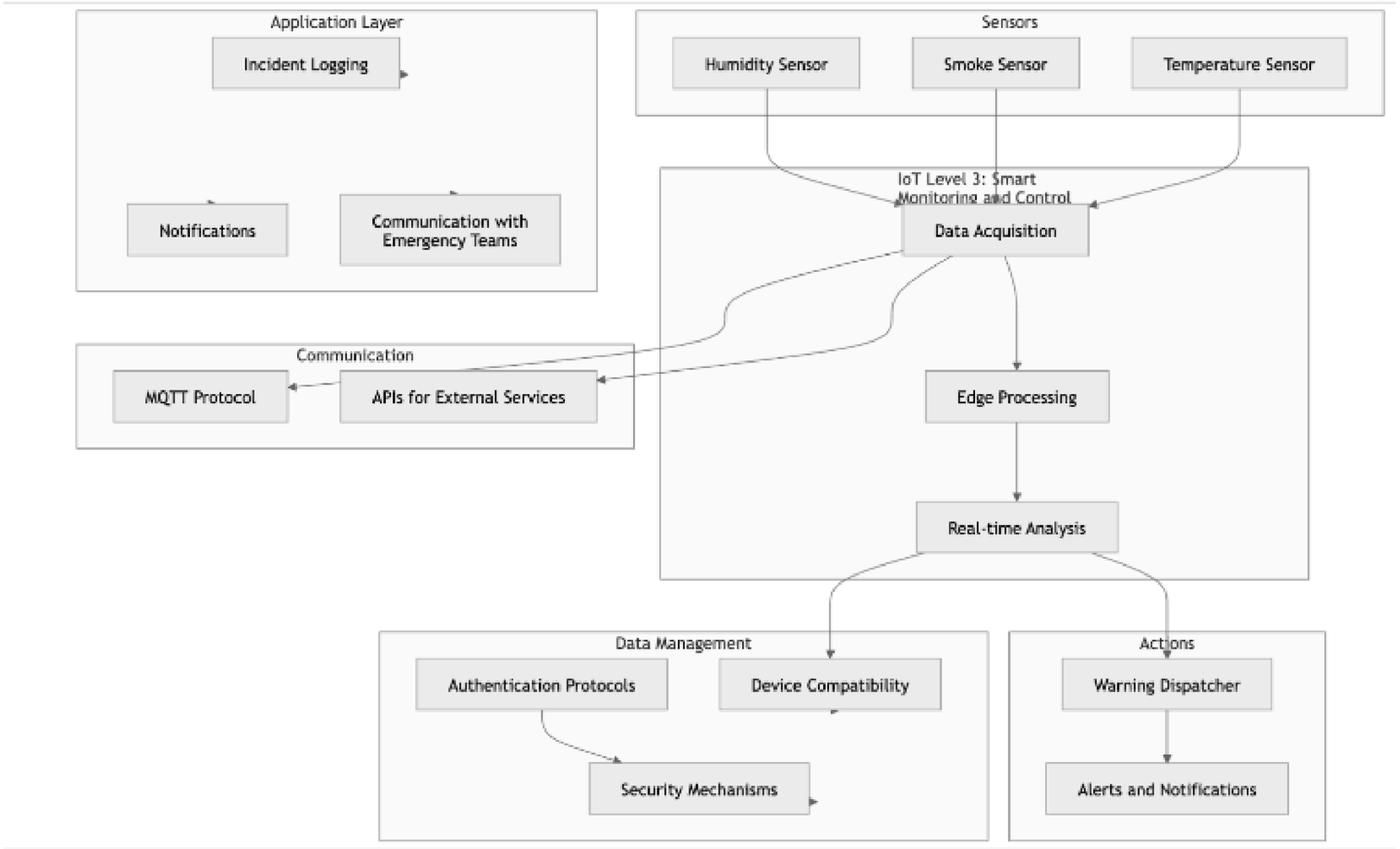
SERVICE SPECIFICATION

- Manages sensor deployment and real-time monitoring.
- Collects and integrates data from various sources.
- Uses AI to detect early fire signs and reduce false alarms.
- Sends alerts and integrates with emergency response systems.
- Predicts fire risks using historical and weather data.

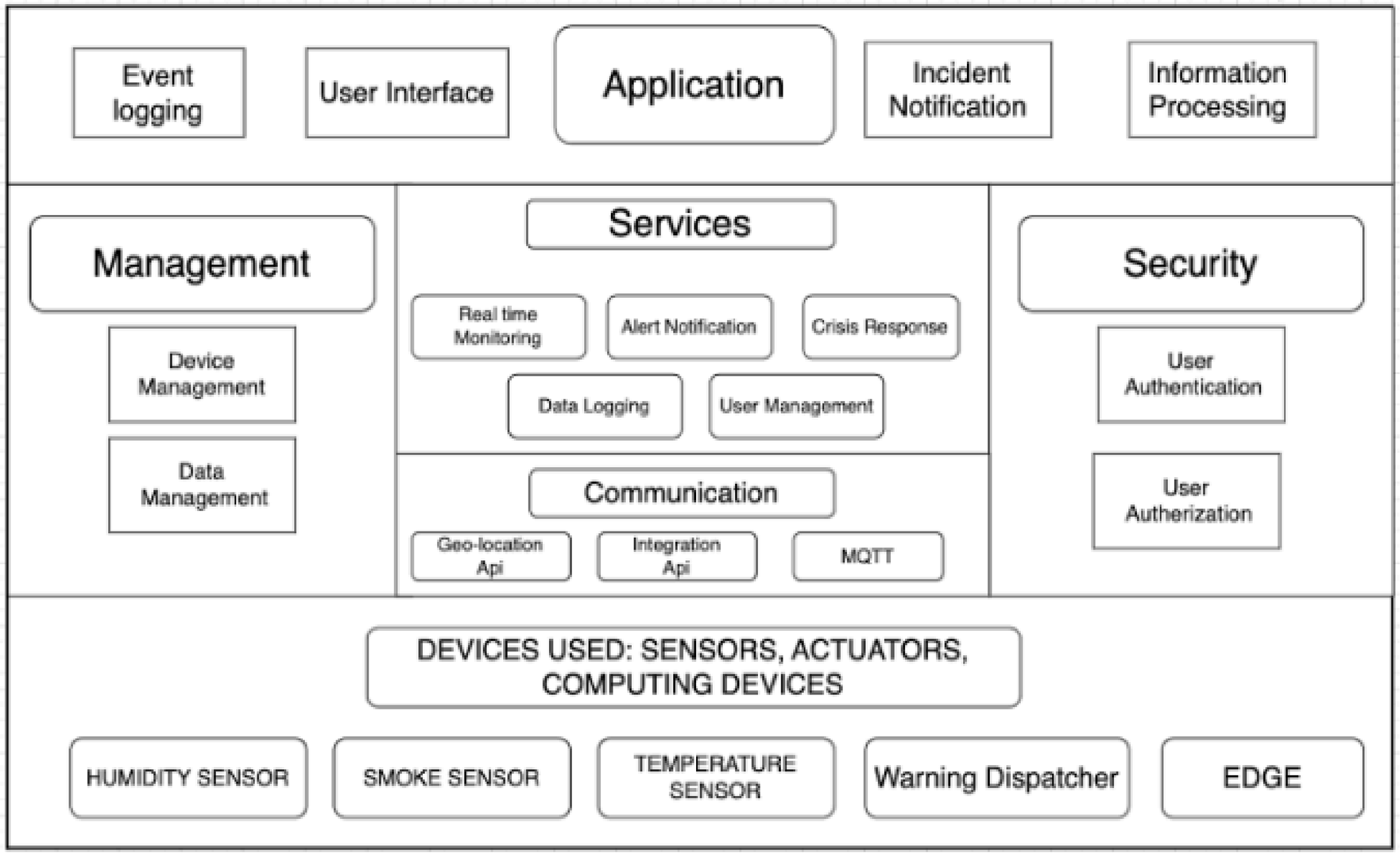
IoT Level Specification

- The forest fire detection system fits under IoT Level 3 specification: due to its real-time data acquisition and edge processing capabilities.
- Edge Processing: Sensors process environmental data locally for immediate decisions.
- Data Relay: Utilizes MQTT for fast data transfer and APIs for external service integration.
- Instant Alerts: Software actuators send notifications when thresholds are exceeded.
- Data Management: Ensures compatibility and security with authentication protocols.

IoT Level Specification



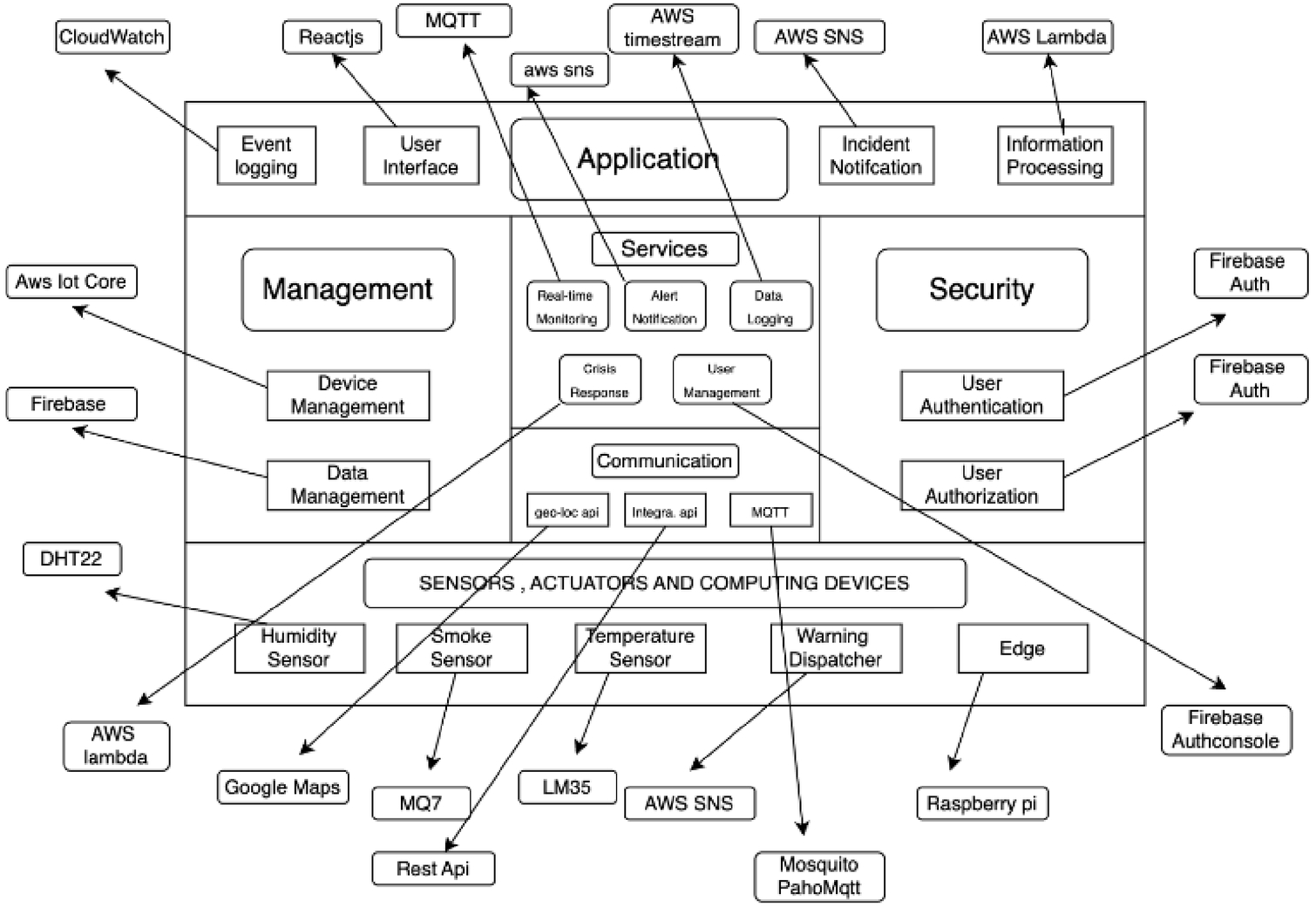
FUNCTIONAL VIEW SPECIFICATION



FUNCTIONAL VIEW SPECIFICATION

- **Devices:** Sensors (humidity, smoke, temperature) and edge devices collect real-time environmental data.
- **Actuators:** Warning dispatcher (software) sends alerts when fire thresholds are reached.
- **Edge Computing:** Locally processes sensor data for quick analysis and timely response.
- **Communication Layer:** Uses MQTT and geo-location APIs to send fire coordinates to control centers.
- **Integration APIs:** Connects system components with third-party services (e.g., fire departments).
- **Services Layer:** Provides real-time monitoring, alerts, and data management for fire response.
- **Management:** Handles sensor data and device compatibility for smooth operation.
- **Security Layer:** Ensures system protection with authentication mechanisms.
- **Application Layer:** Manages user interface, event logs, notifications, and public/emergency team communication.

OPERATIONAL VIEW SPECIFICATION



OPERATIONAL VIEW SPECIFICATION

- **Operational View Design:** Enables real-time monitoring, alerts, and secure data handling.
- **Sensors/Actuators:** DHT22, MQ7, LM35 sensors collect data, Raspberry Pi processes and transmits it via MQTT, drones provide aerial data.
- **Management:** AWS IoT Core and Firebase handle device/data management, AWS CloudWatch logs performance.
- **Application:** Real-time monitoring and alerts via AWS Timestream (data storage) and AWS SNS (alerts), React.js for user interface.
- **Security:** Firebase Auth manages user authentication and authorization.

APPLICATION DEVELOPMENT

- **Technology Stack:** Raspberry Pi, AWS IoT Core, MQTT for seamless data collection and cloud processing.
- **User Interface:** Real-time dashboard with fire alerts, interactive map, and historical data analysis.
- **Mobile Integration:** Mobile app with push notifications and real-time fire risk updates.
- **Scalability & Integration:** Integrated with fire departments and weather services; deployed on AWS Cloud.

Thanks