SIAC-IoT Complete Feature Workflows

🏗️ System Architecture Overview

SIAC-IoT is a comprehensive IoT monitoring platform with 6 microservices:

- **PostgreSQL**: Primary database for IoT data, users, alerts
- **FastAPI Backend**: REST API with ML anomaly detection
- **React Frontend**: Real-time dashboard interface
- **Mosquitto MQTT**: IoT device communication broker
- **InfluxDB**: Time-series metrics database
- **Grafana**: Advanced visualization and monitoring
- **Suricata IDS**: Network intrusion detection

🔐 1. Authentication & User Management

Workflow: User Login

```
1. User enters credentials (username/password)
2. Frontend sends POST /api/v1/auth/login
3. Backend validates against PostgreSQL users table
4. JWT token generated with user role (admin/user)
5. Token stored in localStorage
6. User redirected to dashboard
```

Workflow: User Management (Admin Only)

```
Admin Panel → Create/Edit/Delete Users
├── CRUD operations on users table
├── Password hashing with bcrypt
└── Role-based access control (admin/user)
```

📊 2. IoT Device Management

Workflow: Device Registration

```
1. Admin creates device via POST /api/v1/devices
2. Device metadata stored in PostgreSQL devices table
3. Device appears in device list and monitoring pages
4. Device ready to send telemetry via MQTT
```

Workflow: Device Monitoring

```
ESP32 Device → MQTT Broker → Backend Processing
├── Device publishes telemetry to MQTT topics
├── Backend subscribes and processes messages
├── Data stored in PostgreSQL telemetry table
├── Real-time updates via WebSocket to frontend
└── Dashboard displays live device status
```

Supported IoT Hardware:

- **ESP32 Main Controller**: Central processing unit
- **DHT22 Sensor**: Temperature & humidity monitoring
- **Ultrasonic Sensor**: Distance measurement
- **LED Indicators**: Red (alert) and Green (normal) status

🤖 3. Machine Learning Anomaly Detection

Workflow: Model Training

```
Startup Event → ML Service Initialization
├── Generate 1000 simulated normal telemetry samples
├── Extract 7 features: temp, humidity, tx/rx bytes, connections, time features
├── Train IsolationForest model (contamination=0.05)
├── Save model to model_isolation_forest.pkl
└── Model ready for real-time anomaly detection
```

Workflow: Real-time Anomaly Detection

```
Device Telemetry → Feature Engineering → ML Prediction
├── Raw telemetry received via MQTT
├── Extract 7 features using TelemetryFeatureEngineer
├── IsolationForest predicts anomaly score
├── Score < threshold triggers alert creation
├── Alert stored in PostgreSQL alerts table
└── Real-time notification via WebSocket
```

Feature Engineering (7 Features):

1. **Temperature** (DHT22 reading)
2. **Humidity** (DHT22 reading)
3. **TX Bytes** (log-normalized network transmit)
4. **RX Bytes** (log-normalized network receive)
5. **Active Connections** (current device connections)

6. **Hour of Day** (normalized 0-1)
7. **Day of Week** (normalized 0-1)

🚨 4. Alert Management System

Workflow: Alert Generation

```
Anomaly Detected → Alert Creation → Notification
├── ML service flags anomalous telemetry
├── Alert record created with severity/score
├── Alert stored in PostgreSQL alerts table
├── WebSocket broadcast to connected clients
└── Frontend displays real-time alert notifications
```

Workflow: Alert Handling

```
Dashboard Alerts → Acknowledge/Resolve Actions
├── User views active alerts in Alerts page
├── Click "Acknowledge" → POST /api/v1/alerts/{id}/ack
├── Click "Resolve" → POST /api/v1/alerts/{id}/resolve
├── Alert status updated in database
└── Alert removed from active alerts list
```

Alert Types:

- **ML Anomalies**: Temperature/humidity spikes, unusual network activity
- **Device Offline**: No telemetry received within timeout
- **System Alerts**: Service health issues, connectivity problems

---

🛡️ 5. Network Security (Suricata IDS)

Workflow: Intrusion Detection

```
Network Traffic → Suricata Analysis → Alert Generation
├── Suricata monitors all network interfaces (host mode)
├── Custom rules detect suspicious patterns
├── Alerts logged to infra/suricata/logs/fast.log
├── Backend ingests logs via POST /api/v1/suricata/logs
├── Alerts stored in PostgreSQL suricata_alerts table
└── Real-time display in IDS Alerts dashboard
```

Workflow: Security Monitoring

```
IDS Dashboard → Real-time Security Events
├── GET /api/v1/suricata/logs/recent (last 50 alerts)
├── GET /api/v1/suricata/logs/stats (24h statistics)
├── Auto-refresh every 30 seconds
├── Export functionality (Excel/PDF)
└── Severity-based color coding and filtering
```

Security Rules Monitored:

- **MQTT Protocol**: Connection detection, TLS validation
- **Brute Force**: Failed authentication attempts
- **Network Scans**: Nmap SYN scans, port scanning
- **DoS Attacks**: Flooding and denial of service
- **Intrusions**: Unauthorized access attempts

📈 6. Dashboard & Visualization

Workflow: Main Dashboard

```
Page Load → Data Aggregation → Real-time Display
├── GET /api/v1/dashboard_summary (system overview)
├── GET /api/v1/alerts/recent (last 5 alerts)
├── GET /api/v1/devices (device status)
├── WebSocket connection for live updates
├── Charts update every 30 seconds
└── Interactive device status cards
```

Workflow: IoT Monitoring Page

```
Device Selection → Telemetry Visualization
├── GET /api/v1/telemetry/recent (device-specific data)
├── GET /api/v1/influx/sensor-data (time-series metrics)
├── Recharts.js renders temperature/humidity graphs
├── Real-time sensor status indicators
├── Device-specific monitoring cards
└── Historical data trends and patterns
```

Workflow: Logs Page

```
System Logs → Centralized Viewing
├── GET /api/v1/logs (paginated system logs)
├── Filter by date, level, source
├── Search functionality
├── Export to Excel/PDF
└── Real-time log streaming via WebSocket
```

📊 7. Data Export & Reporting

Workflow: Data Export

```
User Request → Data Retrieval → File Generation
├── Select export type (Excel/PDF)
├── GET /api/v1/export/{type} with filters
├── Backend generates file using pandas/reportlab
├── File download via browser
└── Toast notification on completion
```

Export Types:

- **Telemetry Data**: Historical sensor readings
- **Alert Reports**: Security incidents and anomalies
- **Suricata Logs**: IDS security events
- **System Logs**: Application and system events

🔄 8. Real-time Communication

Workflow: WebSocket Broadcasting

```
Backend Event → WebSocket Broadcast → Frontend Update
├── Alert created → broadcast_websocket_message()
├── Device telemetry → real-time dashboard updates
├── System status changes → live notifications
├── Multiple clients receive simultaneous updates
└── Automatic reconnection on connection loss
```

Workflow: MQTT Device Communication

```
IoT Device → MQTT Broker → Backend Processing
├── Device publishes JSON telemetry to MQTT topics
├── Backend MQTT client subscribes to topics
├── Message processing and validation
├── Data storage in PostgreSQL
├── ML anomaly detection
└── Alert generation if anomalous
```

🏥 9. System Health Monitoring

Workflow: Health Checks

```
Automated Monitoring → Status Dashboard
├── Docker health checks for all services
├── GET /api/v1/health (system status endpoint)
├── Service availability monitoring
├── Database connectivity checks
├── MQTT broker status
└── ML model status and training state
```

Workflow: Service Recovery

```
Service Failure → Automatic Recovery
├── Docker restart policies (unless-stopped)
├── Health check failures trigger restarts
├── Database connection pooling with retries
├── MQTT reconnection logic
└── Graceful degradation for non-critical services
```

🔧 10. Administration Features

Workflow: User Administration

```
Admin Panel → User CRUD Operations
├── List all users with roles and status
├── Create new users with role assignment
├── Update user profiles and permissions
├── Delete inactive users
└── Password reset functionality
```

Workflow: System Configuration

```
Environment Variables → Service Configuration
├── Docker Compose environment variables
├── Database connection strings
├── MQTT broker settings
├── CORS origins configuration
└── Security headers and policies
```

📊 11. Advanced Analytics (Grafana Integration)

Workflow: Metrics Collection

```
System Data → InfluxDB Storage → Grafana Visualization
├── Telemetry data duplicated to InfluxDB
├── Time-series metrics for long-term storage
├── Grafana dashboards for advanced analytics
├── Custom queries and aggregations
└── Historical trend analysis
```

Grafana Dashboards:

- **IoT Device Metrics**: Temperature, humidity, network stats
- **System Performance**: CPU, memory, network usage
- **Security Events**: IDS alerts over time
- **Alert Analytics**: Anomaly patterns and trends

🚀 12. Deployment & Scaling

Workflow: Docker Deployment

```
Docker Compose → Multi-service Deployment
├── docker-compose.yml (production config)
├── docker-compose.override.yml (development)
├── docker-compose.prod.yml (optimized production)
├── Volume management for data persistence
└── Network isolation with siac-network
```

Workflow: Production Scaling

```
Load Balancing → Service Scaling
├── Nginx reverse proxy for frontend
├── Backend horizontal scaling capability
├── Database read replicas (future)
├── MQTT broker clustering (future)
└── InfluxDB high availability (future)
```

📋 API Endpoints Summary

Authentication

- `POST /api/v1/auth/login` - User authentication
- `GET /api/v1/users/me` - Current user profile

Device Management

- `GET /api/v1/devices` - List all devices
- `POST /api/v1/devices` - Create device
- `PUT /api/v1/devices/{id}` - Update device
- `DELETE /api/v1/devices/{id}` - Delete device

Telemetry

- `POST /api/v1/telemetry` - Ingest telemetry data
- `GET /api/v1/telemetry/recent` - Recent telemetry

Alerts

- `GET /api/v1/alerts/recent` - Recent alerts
- `GET /api/v1/alerts/active` - Active alerts
- `POST /api/v1/alerts/{id}/ack` - Acknowledge alert
- `POST /api/v1/alerts/{id}/resolve` - Resolve alert

Machine Learning

- `GET /api/v1/ml/status` - ML model status
- `POST /api/v1/ml/train` - Retrain model

Security (Suricata)

- `POST /api/v1/suricata/logs` - Ingest IDS logs
- `GET /api/v1/suricata/logs/recent` - Recent security events
- `GET /api/v1/suricata/logs/stats` - Security statistics

Dashboard

- `GET /api/v1/dashboard_summary` - System overview
- `GET /api/v1/devices_activity_24h` - 24h activity metrics
- `GET /api/v1/data_volume_7d` - 7-day data volume

System

- `GET /api/v1/health` - System health check
- `WebSocket /ws` - Real-time updates

🔄 Data Flow Architecture

```
IoT Devices (ESP32/DHT22/Ultrasonic/LEDs)
    ↓ MQTT Publish
Mosquitto MQTT Broker
    ↓ Subscribe & Process
FastAPI Backend
├── → PostgreSQL (telemetry, devices, alerts, users)
├── → InfluxDB (time-series metrics)
├── → ML Service (anomaly detection)
└── → WebSocket Broadcast
    ↓ Real-time Updates
React Frontend (Dashboard/Monitoring/Alerts/IDS)
    ↓ User Interactions
Admin Panel (User/Device Management)
    ↓ Export Requests
Data Export (Excel/PDF Reports)
```

🏷️ Technology Stack

Backend

- **FastAPI**: High-performance async API framework
- **PostgreSQL**: Primary relational database
- **SQLAlchemy**: ORM for database operations
- **Pydantic**: Data validation and serialization
- **scikit-learn**: Machine learning (IsolationForest)
- **pandas**: Data manipulation and export
- **reportlab**: PDF generation

Frontend

- **React 18**: UI framework with hooks
- **Vite**: Fast build tool and dev server
- **Tailwind CSS**: Utility-first CSS framework
- **Recharts**: Data visualization library
- **React Router**: Client-side routing
- **Lucide Icons**: Modern icon library

Infrastructure

- **Docker**: Containerization platform
- **Docker Compose**: Multi-container orchestration
- **Mosquitto**: MQTT broker for IoT communication
- **InfluxDB**: Time-series database
- **Grafana**: Advanced visualization platform
- **Suricata**: Network intrusion detection system
- **Nginx**: Reverse proxy and load balancer

Security

- **JWT**: JSON Web Tokens for authentication
- **bcrypt**: Password hashing
- **CORS**: Cross-origin resource sharing
- **Security Headers**: CSP, HSTS, X-Frame-Options

This comprehensive workflow covers all features of your SIAC-IoT platform, from IoT device management and ML-powered anomaly detection to network security monitoring and real-time dashboards. Each feature integrates seamlessly to provide a complete IoT monitoring and security solution.

Last Updated: November 29, 2025 **Version**: 2.0 **Author**: SIAC-IoT Development Team c:\Users\Tanjona\SIAC-IoT\SIAC-IoT_Workflows.md