คู่มือการเชื่อมโยง API Alert Events สำหรับระบบภายนอก

ของระบบ POMS กรมโรงงานอุตสาหกรรม

Service Name : IntegrationAlertEvents

1. การ Authentication

ระบบภายนอกหรือ Worker ทำการ Authentication ด้วย X-API-Key ที่ผู้ดูแลระบบ POMS กำหนดให้

1.1 บริการ API

รายการ	รายละเอียด
Method	POST
Endpoint	`<BASE_URL>/api/v1/integrations/alert-events`
ตัวอย่าง	`http://d-poms.diw.go.th/api/v1/integrations/alert-events`
Security Authentication	X-API-Key
Content-Type	`application/json; charset=UTF-8`
Description	ส่งเหตุการณ์ค่ารายชั่วโมงที่เกินเกณฑ์เข้าสู่ POMS แบบ batch

หมายเหตุ: ควรเรียกใช้งานผ่าน HTTPS เพื่อป้องกัน X-API-Key รั่วระหว่างทาง

2. ชุดข้อมูลส่งไป API

2.1 Header

No.	Parameter	Description
1	`X-API-Key`	Alert Event API Key ที่ผู้ดูแลระบบ POMS กำหนดให้

2.2 Request Root

No.	Parameter	Description
1	`events`	Array ของเหตุการณ์ 1-500 รายการ และต้องผ่าน validation ทั้ง batch

2.3 events[]

No.	Parameter	Description
1	systemType	CEMS หรือ WPMS
2	stationId	รหัสจุดตรวจวัดที่ active และเชื่อมต่อแล้ว
3	pointCode	รหัสจุดตรวจวัด; ไม่ส่งได้ โดยระบบใช้ stationId ค้นหา
4	parameterCode	รหัสพารามิเตอร์ เช่น so2, nox, cod หรือ bod
5	unit	หน่วย เช่น ppm หรือ mg/l
6	eventDate	วันที่ของชั่วโมงที่ตรวจพบ รูปแบบ YYYY-MM-DD
7	time	เวลาเริ่มชั่วโมง รูปแบบ HH:00 ตั้งแต่ 00:00 ถึง 23:00
8	measuredValue	ค่ารายชั่วโมงที่ตรวจวัดได้
9	thresholdValue	ค่าเกณฑ์ที่ใช้เปรียบเทียบ หน่วยเดียวกับ unit
10	thresholdType	STANDARD หรือ EIA

สำคัญ: Contract ปัจจุบันใช้ time เพียงค่าเดียว ฟิลด์ startTime และ endTime จะถูกปฏิเสธด้วย 400 VALIDATION_ERROR

3. สัญญาเวลารายชั่วโมง

Client ส่งวันที่และเวลาเริ่มต้นของชั่วโมง ระบบ POMS จะสร้าง startedAt และ endedAt ให้ครบช่วงชั่วโมงใน timezone +07:00

ค่าที่ Client ส่ง	ค่าที่ระบบสร้าง
eventDate = 2026-03-02	startedAt = 2026-03-02T20:00:00+07:00
time = 20:00	endedAt = 2026-03-02T20:59:59+07:00
รูปแบบที่ไม่ผ่าน	20:30, 24:00, 20, startTime, endTime

เหตุการณ์เดิมที่มี systemType, stationId, parameterCode, alertType และ startedAt ตรงกันจะไม่สร้างแถวซ้ำ

4. ตัวอย่างการเรียกใช้ API

```
curl --request POST \
  --url '<BASE_URL>/api/v1/integrations/alert-events' \
  --header 'X-API-Key: <ALERT_EVENT_API_KEY>' \
  --header 'Content-Type: application/json' \
  --data @alert-event.json
```

4.1 ตัวอย่าง Request Body

```
{
  "events": [
    {
      "systemType": "CEMS",
      "stationId": "S0001",
      "pointCode": "S0001",
      "parameterCode": "so2",
      "unit": "ppm",
      "eventDate": "2026-03-02",
      "time": "20:00",
      "measuredValue": 150,
      "thresholdValue": 60,
      "thresholdType": "STANDARD"
    }
  ]
}
```

ห้ามใส่ API key จริงใน source code, เอกสาร, screenshot หรือ log

5. Response เมื่อประมวลผล Batch

No.	Parameter	Description
1	`success`	`true` เมื่อ request-level validation ผ่าน
2	`data.total`	จำนวนรายการทั้งหมดใน `events`
3	`data.created`	จำนวนรายการที่สร้างใหม่
4	`data.duplicate`	จำนวนรายการที่พบ `idempotencyKey` เดิม
5	`data.failed`	จำนวนรายการที่เกิด business หรือ persistence error
6	`data.results`	ผลลัพธ์เรียงตาม index ของ request

5.1 ตัวอย่าง Success Response ตาม Contract

ตัวอย่างนี้อธิบาย contract และไม่ใช่ผลจาก Production smoke test แบบไม่สร้างข้อมูล

```
{
  "success": true,
  "data": {
    "total": 1,
    "created": 1,
    "duplicate": 0,
    "failed": 0,
    "results": [
      {
        "index": 0,
        "success": true,
        "created": true,
        "duplicate": false,
        "event": {
          "stationId": "S0001",
          "parameterLabel": "SO2 (ppm)",
          "eventDate": "2026-03-02",
          "timeRange": "20.00 - 20.59",
          "startedAt": "2026-03-02T20:00:00+07:00",
          "endedAt": "2026-03-02T20:59:59+07:00",
          "notificationStatus": "AUTO"
        }
      }
    ]
  }
}
```

6. ผลทดสอบจริงจาก Production

Environment: `http://d-poms.diw.go.th` — ทดสอบวันที่ 2 สิงหาคม 2569 โดยใช้ขอบเขต non-writing smoke test

6.1 ตรวจสอบ API Availability

`GET /api/v1`

```
HTTP/1.1 200 OK
{
  "success": true,
  "message": "POMS API",
  "version": "0.1.0"
}
```

6.2 ไม่มี X-API-Key

```
HTTP/1.1 401 Unauthorized
{
  "success": false,
  "error": {
    "code": "UNAUTHORIZED",
    "message": "Missing X-API-Key header"
  }
}
```

6.3 X-API-Key ไม่ถูกต้อง

```
HTTP/1.1 401 Unauthorized
{
  "success": false,
  "error": {
    "code": "UNAUTHORIZED",
    "message": "Invalid integration API key"
  }
}
```

ผลทดสอบยืนยันว่า API ตอบสนองและ authentication guard ทำงาน โดยไม่มี Alert หรือข้อมูล Production ถูกสร้าง

7. ต่าง Error Response

7.1 Request ไม่ผ่าน Validation

```
{
  "success": false,
  "error": {
    "code": "VALIDATION_ERROR",
    "message": "Request validation failed"
  }
}
```

เกิดเมื่อ `time` ไม่ใช่ `HH:00`, `events` ว่างหรือเกิน 500 รายการ หรือมีฟิลด์ที่ไม่อยู่ใน contract เช่น `startTime` และ `endTime`

7.2 รายการใน Batch ประมวลผลไม่สำเร็จ

Request อาจตอบ `200 OK` แต่ `data.failed` มากกว่า 0 เมื่อ `stationId` ไม่ตรงกับ active connected measurement point หรือเกิด business error รายการนั้น ให้ตรวจ `data.results[].error` ก่อน retry

8. หมายเหตุการใช้งาน

- ใช้ API สำหรับระบบภายนอกหรือ Worker และส่งข้อมูลเป็น batch ครั้งละ 1-500 รายการ
- `time` ต้องเป็นต้นชั่วโมงรูปแบบ `HH:00` เท่านั้น ไม่ต้องส่ง `startTime` และ `endTime`
- `stationId` หรือ `pointCode` ต้องตรงกับจุดตรวจวัดที่เชื่อมต่อและ active อยู่ในระบบ POMS
- `parameterCode` จะถูก normalize เป็น lowercase และ response มี `parameterLabel` พร้อมหน่วย เช่น `SO2 (ppm)`
- การส่งเหตุการณ์เดิมซ้ำไม่สร้างแถวใหม่ ระบบรายงานเป็น duplicate ตาม `idempotencyKey`
- ห้ามบันทึก `X-API-Key` ลง source code, เอกสาร หรือ log และควรเรียกผ่าน HTTPS
- ผล Production ในหัวข้อ 6 เป็นการทดสอบแบบไม่สร้างข้อมูล จึงยังไม่ใช่หลักฐานของ authenticated create flow