API Specification — Two Services

Version: 1.0  
Date: 2025-10-02

# 1. Conventions

Content-Type: application/json; charset=utf-8

Encoding: UTF-8

Time format: ISO 8601; timestamps in milliseconds since epoch where applicable.

Authentication: not required by default (should be enforced at the ingress/proxy level if needed).

# 2. Service A — Trainer + Infer (FastAPI)

|  |  |
| --- | --- |
| Параметр | Описание |
| Base URL | http://<host>:8000 |
| API Version | 7.0 |

## 2.1 Endpoints

### GET /health

Service health check.

#### Headers

|  |  |
| --- | --- |
| Параметр | Описание |
| Accept | application/json |

#### Responses

Status 200: application/json

{

"status": "string"

}

##### Example

{

"status": "ok"

}

#### Status codes

|  |  |
| --- | --- |
| Код | Описание |
| 200 | OK |

### POST /train

Triggers training per sensor from Kafka stream.

#### Headers

|  |  |
| --- | --- |
| Параметр | Описание |
| Content-Type | application/json |
| Accept | application/json |

#### Request body (schema)

{

"type": "object",

"properties": {

"seconds": "integer (2..180)"

},

"required": [

"seconds"

],

"example": {

"seconds": 20

}

}

#### Responses

Status 200: application/json

{

"status": "string",

"trained": "[ModelTrainedEvent]"

}

##### Example

{

"status": "ok",

"trained": [

{

"eventType": "MODEL\_TRAINED",

"sensorId": "hvs-ps1-001",

"version": "v0007",

"targets": [

"consumption\_for\_period",

"cumulative\_consumption"

],

"metrics": {

"mae": 0.0123,

"rmse": 0.0456,

"r2": 0.89

},

"timestamp": 1730000000000

}

]

}

Status 400: application/json

{

"detail": "string"

}

##### Example

{

"detail": "Нет данных из Kafka"

}

#### Status codes

|  |  |
| --- | --- |
| Код | Описание |
| 200 | OK |
| 400 | Bad Request |

### POST /predict/{sensor\_id}

Returns prediction for a sensor using the latest history.

#### Headers

|  |  |
| --- | --- |
| Параметр | Описание |
| Content-Type | application/json |
| Accept | application/json |

#### Path parameters

|  |  |
| --- | --- |
| Параметр | Описание |
| sensor\_id | string |

#### Request body (schema)

{

"type": "object",

"properties": {

"minutes": "integer"

},

"required": [

"minutes"

],

"example": {

"minutes": 60

}

}

#### Responses

Status 200: application/json

{

"sensorId": "string",

"version": "string",

"horizon": "integer",

"prediction": "{string:number}"

}

##### Example

{

"sensorId": "hvs-ps1-001",

"version": "v0007",

"horizon": 60,

"prediction": {

"consumption\_for\_period": 0.07,

"cumulative\_consumption": 12346.1

}

}

Status 400: application/json

{

"detail": "string"

}

##### Example

{

"detail": "No features built (not enough history)"

}

Status 404: application/json

{

"detail": "string"

}

##### Example

{

"detail": "Model for hvs-ps1-001 not found"

}

#### Status codes

|  |  |
| --- | --- |
| Код | Описание |
| 200 | OK |
| 400 | Bad Request |
| 404 | Not Found |

## 2.2 Error Model

{

"type": "object",

"schema": {

"detail": "string"

},

"example": {

"detail": "error message"

}

}

## 2.3 Kafka Event — MODEL\_TRAINED

{

"$schema": "http://json-schema.org/draft-07/schema#",

"title": "ModelTrainedEvent",

"type": "object",

"properties": {

"eventType": {

"const": "MODEL\_TRAINED"

},

"sensorId": {

"type": "string"

},

"version": {

"type": "string"

},

"targets": {

"type": "array",

"items": {

"type": "string"

}

},

"metrics": {

"type": "object",

"properties": {

"mae": {

"type": "number"

},

"rmse": {

"type": "number"

},

"r2": {

"type": "number"

}

},

"required": [

"mae",

"rmse"

]

},

"timestamp": {

"type": "integer"

}

},

"required": [

"eventType",

"sensorId",

"version",

"targets",

"metrics",

"timestamp"

]

}

# 3. Service B — Java Spring Controllers

Controllers in scope: EvaluateController, ModeController.

## 3.1 EvaluateController

|  |  |
| --- | --- |
| Параметр | Описание |
| Source file present | True |
| Base path | — |

Method-level endpoint annotations were not detected in the provided file. Only the controller base path is documented.

## 3.2 ModeController

|  |  |
| --- | --- |
| Параметр | Описание |
| Source file present | True |
| Base path | — |

Method-level endpoint annotations were not detected in the provided file. Only the controller base path is documented.

# Appendix A — HTTP Examples (curl)

curl -s http://localhost:8000/health

curl -s -X POST http://localhost:8000/train -H "Content-Type: application/json" -d '{"seconds":30}'

curl -s -X POST http://localhost:8000/predict/hvs-ps1-001 -H "Content-Type: application/json" -d '{"minutes":60}'