

# Aurora – Sistema de Automação Residencial

Documentação Técnica: Banco de Dados, Entidades e Comunicação entre Serviços

## 1. Visão Geral da Arquitetura de Dados

O sistema Aurora é composto por sete microsserviços independentes, cada um responsável por um bounded context distinto do domínio de automação residencial. Todos os serviços compartilham uma única instância PostgreSQL (aurora\_home), operando sobre conjuntos de tabelas dedicadas, decisão pragmática para um ambiente embarcado residencial que reduz overhead de infraestrutura sem comprometer a independência lógica entre os contextos.

### 1.1 Serviços e suas responsabilidades

Serviço	Porta	Responsabilidade
auth-service	8080	Cadastro e autenticação de usuários; emissão de JWT
sensors-service	8081	Recepção, armazenamento e streaming de leituras do ESP32
lighting-service	8082	Gerenciamento do estado das lâmpadas e controle do ESP32
security-service	8083	Gerenciamento do alarme residencial e acionamento do buzzer
rules-service	8084	Motor de regras condicionais: condição → ação
notifications-service	8085	Histórico de notificações e integração com Telegram
schedule-service	8086	Agendamentos cron e pontuais de ações nos dispositivos

## 2. Estrutura de Banco de Dados por Serviço

### 2.1 auth-service

Gerencia identidades de usuários. Única tabela persistida neste contexto.

**Tabela: users**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID do usuário
email	VARCHAR(255)	UNIQUE NOT NULL	E-mail de login
password_hash	VARCHAR(255)	NOT NULL	Senha em hash bcrypt
created_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Data de criação
updated_at	TIMESTAMP	NOT NULL DEFAULT	Última atualização

		NOW()	
--	--	-------	--

## 2.2 sensors-service

Gerencia sensores cadastrados e persiste histórico de leituras de movimento e luminosidade.

**Tabela: sensors**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID do sensor
name	VARCHAR(255)	NOT NULL	Nome do sensor
type	VARCHAR(50)	NOT NULL	Tipo: motion   light
location	VARCHAR(255)	NOT NULL	Localização física
owner_id	VARCHAR(255)	NOT NULL DEFAULT ''	UUID do usuário dono
active	BOOLEAN	NOT NULL DEFAULT true	Sensor habilitado
created_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Data de criação
updated_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Última atualização

**Tabela: motion\_records**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID do registro
sensor_id	VARCHAR(36)	REFERENCES sensors(id)	FK para sensor
user_id	VARCHAR(255)	NOT NULL	UUID do usuário
detected_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Momento da detecção

**Tabela: light\_records**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID do registro
sensor_id	VARCHAR(36)	REFERENCES sensors(id)	FK para sensor
value	DECIMAL(5,2)	NOT NULL	Luminosidade em % (0-100)
raw	INTEGER	NOT NULL	Valor bruto do ADC do ESP32
recorded_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Momento da leitura

**Tabela: devices**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	ID do dispositivo
name	VARCHAR(255)	NOT NULL	Nome do dispositivo
ip_address	VARCHAR(50)	NOT NULL	IP na rede local
port	INTEGER	NOT NULL DEFAULT 80	Porta HTTP do firmware

last_seen	TIMESTAMP	NOT NULL DEFAULT NOW()	Último contato
created_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Data de registro

## 2.3 lighting-service

Persiste o estado das lâmpadas e registra o endereço dos dispositivos físicos.

**Tabela: lights**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID da lâmpada
name	VARCHAR(255)	NOT NULL	Nome da lâmpada
location	VARCHAR(255)	NOT NULL	Ambiente/localização
owner_id	VARCHAR(255)	NOT NULL DEFAULT ''	UUID do usuário dono
state	VARCHAR(10)	NOT NULL DEFAULT 'off'	Estado: on   off
device_id	VARCHAR(255)	NOT NULL	ID:pin do ESP32 (ex: esp32-main:1)
created_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Data de criação
updated_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Última atualização

**Tabela: registered\_devices**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(255)	PRIMARY KEY	ID do dispositivo
ip_address	VARCHAR(50)	NOT NULL	IP na rede local
port	INTEGER	NOT NULL DEFAULT 80	Porta HTTP do firmware
last_seen	TIMESTAMP	NOT NULL DEFAULT NOW()	Último contato
created_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Data de registro

## 2.4 security-service

Persiste o histórico de eventos de alarme (acionamentos e silenciamentos).

**Tabela: alarm\_events**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID do evento
user_id	VARCHAR(255)	NOT NULL DEFAULT ''	UUID do usuário
trigger_type	VARCHAR(50)	NOT NULL	Tipo: motion   manual
sensor_id	VARCHAR(255)	NOT NULL DEFAULT ''	Sensor que originou (se aplicável)
location	VARCHAR(255)	NOT NULL DEFAULT ''	Localização do evento
status	VARCHAR(20)	NOT NULL DEFAULT 'triggered'	Estado: triggered   silenced
triggered_at	TIMESTAMP	NOT NULL DEFAULT	Momento do disparo

silenced_at	TIMESTAMP	NOW() NULL	Momento do silenciamento
-------------	-----------	---------------	--------------------------

## 2.5 rules-service

Persiste as regras de automação condicional criadas pelo usuário.

**Tabela: rules**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID da regra
name	VARCHAR(255)	NOT NULL	Nome da regra
owner_id	VARCHAR(255)	NOT NULL DEFAULT ''	UUID do usuário dono
trigger_type	VARCHAR(50)	NOT NULL	Gatilho: motion   light_low   light_high
trigger_id	VARCHAR(255)	NOT NULL DEFAULT ''	ID do sensor gatilho
trigger_threshold	DECIMAL(5,2)	NOT NULL DEFAULT 30	Limiar de luminosidade (%)
action_type	VARCHAR(50)	NOT NULL	Ação: turn_on_light   turn_off_light   trigger_alarm
action_id	VARCHAR(255)	NOT NULL DEFAULT ''	ID do dispositivo alvo
enabled	BOOLEAN	NOT NULL DEFAULT true	Regra ativa
created_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Data de criação
updated_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Última atualização

## 2.6 schedule-service

Persiste agendamentos do usuário (cron e pontuais) e histórico de execuções.

**Tabela: schedules**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID do agendamento
owner_id	VARCHAR(255)	NOT NULL	UUID do usuário dono
name	VARCHAR(255)	NOT NULL	Nome do agendamento
description	TEXT	NOT NULL DEFAULT ''	Descrição opcional
schedule_type	VARCHAR(20)	NOT NULL	Tipo: cron   once
cron_expr	VARCHAR(100)	NOT NULL DEFAULT ''	Expressão cron (para tipo cron)
run_at	TIMESTAMP	NULL	Horário pontual (para tipo once)
days_filter	INTEGER	NOT NULL DEFAULT 0	Máscara de dias da semana
timezone	VARCHAR(64)	NOT NULL DEFAULT 'UTC'	Fuso horário do agendamento
action_target	VARCHAR(10)	NOT NULL	Alvo: lighting   security
action_type	VARCHAR(50)	NOT NULL	Ação: turn_on   turn_off

			trigger   silence
action_device_id	VARCHAR(255)	NOT NULL DEFAULT ''	ID do dispositivo alvo
action_payload	TEXT	NOT NULL DEFAULT ''	Payload adicional
enabled	BOOLEAN	NOT NULL DEFAULT true	Agendamento ativo
created_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Data de criação
updated_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Última atualização
last_run_at	TIMESTAMP	NULL	Última execução
next_run_at	TIMESTAMP	NULL	Próxima execução prevista

**Tabela: schedule\_executions**

Coluna	Tipo	Restrições	Descrição
id	VARCHAR(36)	PRIMARY KEY	UUID da execução
schedule_id	VARCHAR(36)	REFERENCES schedules(id) ON DELETE CASCADE	FK para agendamento
executed_at	TIMESTAMP	NOT NULL DEFAULT NOW()	Momento da execução
status	VARCHAR(20)	NOT NULL	Resultado: success   error
error_message	TEXT	NOT NULL DEFAULT ''	Mensagem de erro (se houver)

## 2.7 notifications-service

Persiste notificações de eventos do sistema e preferências de Telegram.

**Tabela: notifications**

Coluna	Tipo	Restrições	Descrição
id	UUID	PRIMARY KEY	UUID da notificação
user_id	VARCHAR(255)	NOT NULL	UUID do usuário
type	VARCHAR(50)	NOT NULL	Tipo do evento
title	VARCHAR(255)	NOT NULL	Título da notificação
message	TEXT	NOT NULL	Corpo da mensagem
sensor_id	VARCHAR(255)	NULL	Sensor relacionado (opcional)
location	VARCHAR(255)	NULL	Local do evento (opcional)
read	BOOLEAN	NOT NULL DEFAULT FALSE	Notificação lida pelo usuário
created_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW()	Data de criação

**Tabela: telegram\_preferences**

Coluna	Tipo	Restrições	Descrição
id	UUID	PRIMARY KEY DEFAULT gen_random_uuid()	UUID da preferência

user_id	UUID	NOT NULL UNIQUE	UUID do usuário (1:1)
chat_id	BIGINT	NOT NULL	ID do chat no Telegram
enabled_types	TEXT[]	NOT NULL	Tipos de evento habilitados
active	BOOLEAN	NOT NULL DEFAULT TRUE	Integração ativa
linked_at	TIMESTAMPTZ	NOT NULL DEFAULT NOW()	Data de vinculação

**Tabela: telegram\_link\_tokens**

Coluna	Tipo	Restrições	Descrição
token	VARCHAR(20)	PRIMARY KEY	Token temporário de vinculação
user_id	UUID	NOT NULL	UUID do usuário
expires_at	TIMESTAMPTZ	NOT NULL	Validade do token

### 3. Relacionamentos entre Tabelas

As relações abaixo são definidas por chaves estrangeiras (REFERENCES) ou por convenção de owner\_id que referencia a PK da tabela users.

Tabela origem	Coluna	Tabela destino	Coluna	Tipo de relação
motion_records	sensor_id	sensors	id	N:1 — vários registros por sensor
light_records	sensor_id	sensors	id	N:1 — vários registros por sensor
schedule_executions	schedule_id	schedules	id	N:1 (CASCADE DELETE)
sensors	owner_id	users	id	N:1 — convenção (sem FK explícita)
lights	owner_id	users	id	N:1 — convenção (sem FK explícita)
rules	owner_id	users	id	N:1 — convenção (sem FK explícita)
schedules	owner_id	users	id	N:1 — convenção (sem FK explícita)
notifications	user_id	users	id	N:1 — convenção (sem FK explícita)
telegram_preferences	user_id	users	id	1:1 — convenção (sem FK explícita)
alarm_events	user_id	users	id	N:1 — convenção (sem FK explícita)

## 4. Comunicação entre Serviços

O sistema Aurora utiliza três modalidades de comunicação entre os microsserviços: síncrona (HTTP/REST), assíncrona orientada a eventos (NATS PubSub) e em tempo real (WebSocket). As seções a seguir detalham cada modalidade.

### 4.1 Comunicação Assíncrona via NATS (Pub/Sub)

Eventos de domínio são publicados no NATS e consumidos pelos serviços inscritos no tópico correspondente. O desacoplamento é total: o publicador não sabe quem consome.

Tópico NATS	Publicador	Consumidores	Quando é publicado
sensors.motion.detected	sensors-service	rules-service security-service notifications-service	ESP32 detecta movimento via sensor PIR
sensors.light.changed	sensors-service	rules-service notifications-service	ESP32 envia leitura de luminosidade (LDR)
lighting.state.changed	lighting-service	notifications-service	Estado de uma lâmpada é alterado (ligada/desligada)
security.alarm.triggered	security-service	notifications-service	Alarme é acionado (manual ou por regra)
schedule.executed	schedule-service	notifications-service	Agendamento é executado no horário configurado

### 4.2 Comunicação Síncrona via HTTP (REST)

Chamadas HTTP diretas entre serviços são utilizadas para ações imediatas que exigem confirmação. A autenticação entre serviços é feita via header X-Device-Key.

Serviço chamador	Serviço destino	Endpoint	Quando ocorre
rules-service	lighting-service	POST /lights/{id}/on	Regra com ação turn_on_light é disparada
rules-service	lighting-service	POST /lights/{id}/off	Regra com ação turn_off_light é disparada
rules-service	security-service	POST /alarms/trigger	Regra com ação trigger_alarm é disparada
schedule-service	lighting-service	POST /lights/{id}/on	Agendamento de ligar luz no horário
schedule-service	lighting-service	POST /lights/{id}/off	Agendamento de desligar luz no horário
schedule-service	security-service	POST /alarms/trigger	Agendamento de acionar alarme
schedule-service	security-service	POST /alarms/{id}/silence	Agendamento de silenciar alarme
lighting-service	ESP32 firmware	GET /led/{pin}/{state}	Ligar/desligar LED no pino configurado

security-service	ESP32 firmware	GET /buzzer	Acionar buzzer do ESP32
------------------	----------------	-------------	-------------------------

### 4.3 Comunicação ESP32 → Backend

O firmware do ESP32 se comunica com o backend por HTTP autenticado via header X-Device-Key. Não há comunicação direta do firmware com outros serviços além do sensors-service e do lighting-service.

Origem	Destino	Endpoint	Descrição
ESP32 (Core 0 / ISR)	sensors-service	POST /device/motion	Envia evento de detecção de movimento (PIR)
ESP32 (Core 0 / task)	sensors-service	POST /device/light	Envia leitura periódica de luminosidade (LDR)

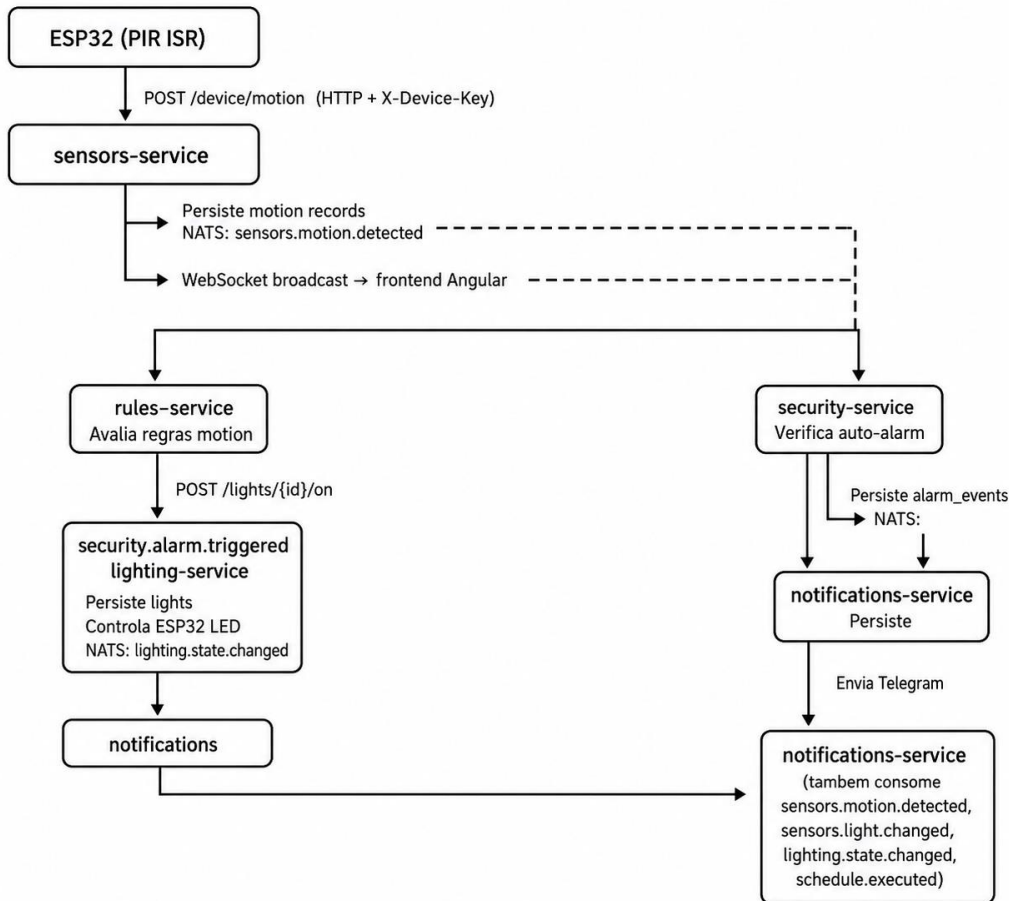
### 4.4 Comunicação em Tempo Real via WebSocket

O frontend Angular mantém conexões WebSocket abertas com os serviços que produzem eventos em tempo real. Os eventos são transmitidos via broadcast para todos os clientes conectados.

Serviço	Endpoint WS	Evento transmitido	Quando
sensors-service	/ws	motion_detected	Movimento detectado pelo PIR
sensors-service	/ws	light_updated	Nova leitura de luminosidade
lighting-service	/ws	light_changed	Estado de lâmpada alterado
security-service	/ws	alarm_changed	Estado do alarme alterado

## 5. Diagrama de Fluxo de Comunicação

O diagrama abaixo representa o fluxo completo de um evento de detecção de movimento, desde o hardware até o frontend e o Telegram.



## 6. Índices de Banco de Dados

Índices criados para otimizar as consultas mais frequentes de cada serviço.

Tabela	Índice	Coluna(s)	Justificativa
users	idx_users_email	email	Busca de usuário por e-mail no login
sensors	idx_sensors_owner_id	owner_id	Listagem de sensores por usuário
sensors	idx_sensors_type	type	Filtro por tipo de sensor
motion_records	idx_motion_records_sensor_id	sensor_id	Histórico de movimento por sensor
motion_records	idx_motion_records_detected_at	detected_at	Ordenação cronológica
light_records	idx_light_records_sensor_id	sensor_id	Histórico de luminosidade por sensor
light_records	idx_light_records_recorded_at	recorded_at	Ordenação cronológica
lights	idx_lights_owner_id	owner_id	Listagem de lâmpadas por usuário
alarm_events	idx_alarm_events_status	status	Filtro por estado do

			alarme
alarm_events	idx_alarm_events_triggered_at	triggered_at	Ordenação cronológica
rules	idx_rules_owner_id	owner_id	Listagem de regras por usuário
rules	idx_rules_trigger	trigger_type, trigger_id	Avaliação de regras por evento NATS
schedules	idx_schedules_owner_id	owner_id	Listagem de agendamentos por usuário
schedules	idx_schedules_enabled	enabled	Carregamento de agendamentos ativos
schedule_executions	idx_executions_schedule_id	schedule_id	Histórico de execuções por agendamento
notifications	idx_notifications_user_id	user_id	Notificações por usuário
notifications	idx_notifications_user_read	user_id, read	Contagem de não lidas

# Aurora - API Agregada (1.0.0)

Download OpenAPI specification: [Download](#)

Documentação agregada dos microsserviços do sistema Aurora.

## Auth

Auth Service API

### Login de usuário

REQUEST BODY SCHEMA: `application/json`  
**required**

Credenciais de login

email	string
password	string

### Responses

- > **200** OK
- > **400** Bad Request
- > **401** Unauthorized
- > **500** Internal Server Error

POST /auth-service/auth/login



## Request samples

### Payload

Content type  
application/json

Copy

```
{
  "email": "usuario@aurora.com",
  "password": "SenhaForte123!"
}
```

## Response samples

200

400

401

500

Content type  
application/json

Copy

```
{
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9..."
}
```

## Registro de novo usuário

REQUEST BODY SCHEMA: application/json  
required

### Dados de registro

email	string
password	string

## Responses

> 201 Created

> 400 Bad Request

> 409 Conflict

> 500 Internal Server Error

POST /auth-service/auth/register

## Request samples

### Payload

Content type  
application/json

Copy

```
{
  "email": "usuario@aurora.com",
  "password": "SenhaForte123!"
}
```

## Response samples

201

400

409

500

Content type  
application/json

Copy

```
{
  "email": "usuario@aurora.com",
  "id": "550e8400-e29b-41d4-a716-446655440000"
}
```

Sensors

## Lista sensores do usuário autenticado

AUTHORIZATIONS: > *BearerAuth*

### Responses

> 200 OK

> 401 Unauthorized

> 500 Internal Server Error

GET /sensors-service/sensors

### Response samples

200

401

500

Content type  
application/json

Copy Expand all Collapse all

```
[
  - {
    "active": "true",
    "createdAt": "string",
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "location": "Sala de estar",
    "name": "Rotina noturna",
    "ownerID": "string",
    "type": "motion",
    "updatedAt": "string"
  }
]
```

# Registra evento de luminosidade via dispositivo (ESP32)

## PATH PARAMETERS

---

→ **id** required string  
ID do sensor

---

## HEADER PARAMETERS

---

→ **X-Device-Key** required string  
Chave de autenticação do dispositivo

---

REQUEST BODY SCHEMA: `application/json`  
required

---

Valor de luminosidade  
object

## Responses

> **200 OK**

> **400 Bad Request**

> **401 Unauthorized**

POST

/sensors-service/sensors/device/{id}/light



## Request samples

Payload

Content type

application/json

Copy

```
{ }
```

## Response samples

200

400

401

Content type

application/json

Copy

```
{
  "event_id": "string",
  "status": "string",
  "value": 1
}
```

## Registra evento de movimento via dispositivo (ESP32)

### PATH PARAMETERS

→ **id** required string  
ID do sensor

### HEADER PARAMETERS

→ **X-Device-Key** required string  
Chave de autenticação do dispositivo

## Responses

> 200 OK

> 400 Bad Request

> 401 Unauthorized

POST /sensors-service/sensors/device/{id}/motion



## Response samples

200

400

401

Content type

application/json

Copy

```
{
  "event_id": "string",
  "status": "string"
}
```

## Histórico de eventos de luminosidade de um sensor

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

→ id **required** string  
ID do sensor

### Responses

> 200 OK

> 400 Bad Request

> 500 Internal Server Error

GET /sensors-service/sensors/{id}/light

### Response samples

200

400

500

Content type

application/json

Copy Expand all Collapse all

```
[
  - {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "raw": 1,
    "recordedAt": "string",
    "sensorID": "string",
    "value": { }
  }
]
```

## Histórico de eventos de movimento de um sensor

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

→ id string  
required ID do sensor

### Responses

> 200 OK

> 400 Bad Request

> 500 Internal Server Error

GET /sensors-service/sensors/{id}/motion

### Response samples

200

400

500

Content type  
application/json

```
[  
  - {  
    "detectedAt": "string",  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "sensorID": "string",  
    "userID": "string"  
  }  
]
```

## Registra evento de movimento via usuário autenticado

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

→ id **required** string  
ID do sensor

### Responses

> 200 OK

> 400 Bad Request

> 401 Unauthorized

POST /sensors-service/sensors/{id}/motion

### Response samples

200

400

401

Content type  
application/json

Copy

```
{  
  "event_id": "string",  
  "status": "string"  
}
```

## Lighting

Lighting Service API

### Registra um dispositivo ESP32

#### HEADER PARAMETERS

---

→ X-Device-Key required	string Chave de autenticação do dispositivo
----------------------------	--

---

REQUEST BODY SCHEMA: application/json  
required

---

Dados do dispositivo  
object

#### Responses

> 200 OK

> 400 Bad Request

> 401 Unauthorized

POST /lighting-service/devices/register



Request samples

## Payload

### Content type

application/json

Copy

```
{ }
```

## Response samples

200

400

401

### Content type

application/json

Copy

```
{  
  "property1": "string",  
  "property2": "string"  
}
```

## Lista todas as luzes do usuário

AUTHORIZATIONS: >

*BearerAuth*

## Responses

> 200 OK

> 500 Internal Server Error

GET /lighting-service/lights



## Response samples

200

500

Content type

application/json

Copy Expand all Collapse all

```
[
  - {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "location": "Sala de estar",
    "name": "Rotina noturna",
    "state": "on"
  }
]
```

## Desliga uma luz

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

→ id required string  
ID da luz

### Responses

> 200 OK

> 400 Bad Request

> 403 Forbidden

> 404 Not Found

POST /lighting-service/lights/{id}/off



## Response samples

200

400

403

404

### Content type

application/json

Copy

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "location": "Sala de estar",
  "name": "Rotina noturna",
  "state": "on"
}
```

## Liga uma luz

AUTHORIZATIONS: >

*BearerAuth*

PATH PARAMETERS

→ id  
required

string  
ID da luz

## Responses

> 200 OK

> 400 Bad Request

> 403 Forbidden

> 404 Not Found

POST /lighting-service/lights/{id}/on



## Response samples

200

400

403

404

#### Content type

application/json

Copy

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "location": "Sala de estar",
  "name": "Rotina noturna",
  "state": "on"
}
```

## Retorna o status de uma luz

AUTHORIZATIONS: > *BearerAuth*

#### PATH PARAMETERS

→ id **required** string  
ID da luz

## Responses

> 200 OK

> 400 Bad Request

> 404 Not Found

GET /lighting-service/lights/{id}/status

## Response samples

200

400

404

#### Content type

application/json

Copy

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "location": "Sala de estar",
  "name": "Rotina noturna",
  "state": "on"
}
```

## Security

Security Service API

### Lista os alarmes recentes

AUTHORIZATIONS: > *BearerAuth*

### Responses

> 200 OK

> 500 Internal Server Error

GET /security-service/alarms



### Response samples

200

500

### Content type

application/json

Copy Expand all Collapse all

```
[
  - {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "location": "Sala de estar",
    "sensor_id": "550e8400-e29b-41d4-a716-446655440001",
    "status": "triggered",
    "trigger_type": "threshold",
    "triggered_at": "string"
  }
]
```

## Dispara um alarme

REQUEST BODY SCHEMA: application/json

Dados opcionais do alarme (location, trigger\_type, sensor\_id)  
object

### Responses

> 200 OK

> 500 Internal Server Error

POST /security-service/alarms/trigger

### Request samples

Payload

### Content type

application/json

Copy

```
{ }
```

## Response samples

200

500

### Content type

application/json

Copy

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "location": "Sala de estar",
  "sensor_id": "550e8400-e29b-41d4-a716-446655440001",
  "status": "triggered",
  "trigger_type": "threshold",
  "triggered_at": "string"
}
```

## Silencia um alarme ativo

AUTHORIZATIONS: >

*BearerAuth*

PATH PARAMETERS

→ id  
required

string  
ID do alarme

## Responses

> 200 OK

> 404 Not Found

POST /security-service/alarms/{id}/silence



## Response samples

200

404

### Content type

application/json

Copy

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "location": "Sala de estar",
  "sensor_id": "550e8400-e29b-41d4-a716-446655440001",
  "status": "triggered",
  "trigger_type": "threshold",
  "triggered_at": "string"
}
```

## Rules

Rules Service API

## Lista regras de automação do usuário

AUTHORIZATIONS: >

*BearerAuth*

## Responses

> 200 OK

> 500 Internal Server Error

GET /rules-service/rules

## Response samples

200

500

Content type  
application/json

Copy Expand all Collapse all

```
[
  - {
    "action_id": "550e8400-e29b-41d4-a716-446655440002",
    "action_type": "turn_on_light",
    "created_at": "2026-06-16T18:00:00-03:00",
    "enabled": true,
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Rotina noturna",
    "trigger_id": "550e8400-e29b-41d4-a716-446655440003",
    "trigger_threshold": "30",
    "trigger_type": "motion"
  }
]
```

## Cria uma nova regra de automação

AUTHORIZATIONS: > *BearerAuth*

REQUEST BODY SCHEMA: application/json  
required

### Dados da regra

action_id	string
action_type	string (Rules_domain.ActionType) Enum: "turn_on_light" "turn_off_light" "trigger_alarm"
name	string
trigger_id	string

trigger_threshold	number
	Para light_low: considerar luz baixa abaixo deste % (0 = usar 30)
trigger_type	string (Rules_domain.TriggerType)
	Enum: "motion" "light_low" "light_high" "schedule"

## Responses

> 201 Created

> 400 Bad Request

> 500 Internal Server Error

POST /rules-service/rules

### Request samples

Payload

Content type  
application/json

Copy

```
{
  "action_id": "550e8400-e29b-41d4-a716-446655440002",
  "action_type": "turn_on_light",
  "name": "Rotina noturna",
  "trigger_id": "550e8400-e29b-41d4-a716-446655440003",
  "trigger_threshold": "30",
  "trigger_type": "motion"
}
```

### Response samples

201

400

500

Content type  
application/json

```
{
  "action_id": "550e8400-e29b-41d4-a716-446655440002",
  "action_type": "turn_on_light",
  "created_at": "2026-06-16T18:00:00-03:00",
  "enabled": true,
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Rotina noturna",
  "trigger_id": "550e8400-e29b-41d4-a716-446655440003",
  "trigger_threshold": "30",
  "trigger_type": "motion"
}
```

## Atualiza uma regra de automação

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

---

→ id **required** string  
ID da regra

---

REQUEST BODY SCHEMA: application/json  
**required**

---

Dados da regra

→ action\_id string

---

→ action\_type string (Rules\_domain.ActionType)  
Enum: "turn\_on\_light" "turn\_off\_light" "trigger\_alarm"

---

→ name string

---

→ trigger\_id string

---

→ trigger\_threshold number  
Para light\_low: considerar luz baixa abaixo deste % (0 = usar 30)

---

→ trigger\_type string (Rules\_domain.TriggerType)  
Enum: "motion" "light\_low" "light\_high" "schedule"

---

## Responses

> 200 OK

> 400 Bad Request

> 403 Forbidden

> 404 Not Found

PUT /rules-service/rules/{id}

### Request samples

Payload

Content type  
application/json

Copy

```
{
  "action_id": "550e8400-e29b-41d4-a716-446655440002",
  "action_type": "turn_on_light",
  "name": "Rotina noturna",
  "trigger_id": "550e8400-e29b-41d4-a716-446655440003",
  "trigger_threshold": "30",
  "trigger_type": "motion"
}
```

### Response samples

200

400

403

404

Content type  
application/json

Copy

```
{
  "action_id": "550e8400-e29b-41d4-a716-446655440002",
  "action_type": "turn_on_light",
  "created_at": "2026-06-16T18:00:00-03:00",
}
```

```
"enabled": true,
"id": "550e8400-e29b-41d4-a716-446655440000",
"name": "Rotina noturna",
"trigger_id": "550e8400-e29b-41d4-a716-446655440003",
"trigger_threshold": "30",
"trigger_type": "motion"
}
```

## Remove uma regra de automação

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

→ id required string  
ID da regra

### Responses

> 200 OK

> 403 Forbidden

> 404 Not Found

**DELETE** /rules-service/rules/{id} ▼

### Response samples

200

403

404

Content type  
application/json

Copy

```
{
  "property1": "string",
  "property2": "string"
}
```

## Ativa/desativa uma regra de automação

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

→ **id** string  
**required** ID da regra

### Responses

> 200 OK

> 403 Forbidden

> 404 Not Found

PATCH /rules-service/rules/{id}/toggle

### Response samples

200

403

404

Content type  
application/json

Copy

```
{
  "action_id": "550e8400-e29b-41d4-a716-446655440002",
  "action_type": "turn_on_light",
  "created_at": "2026-06-16T18:00:00-03:00",
}
```

```
"enabled": true,  
"id": "550e8400-e29b-41d4-a716-446655440000",  
"name": "Rotina noturna",  
"trigger_id": "550e8400-e29b-41d4-a716-446655440003",  
"trigger_threshold": "30",  
"trigger_type": "motion"  
}
```

## Schedule

Schedule Service API

### Lista agendamentos do usuário

AUTHORIZATIONS: > *BearerAuth*

### Responses

> 200 OK

> 500 Internal Server Error

GET /schedule-service/schedules

### Response samples

200

500

Content type  
application/json

Copy Expand all Collapse all

```
[
  - {
    "action_device_id": "550e8400-e29b-41d4-a716-446655440004",
    "action_payload": "{\"brightness\": 80}",
    "action_target": "nats",
    "action_type": "turn_on_light",
    "created_at": "2026-06-16T18:00:00-03:00",
    "cron_expr": "0 18 * * *",
    "days_filter": "mon,tue,wed,thu,fri",
    "description": "Liga as luzes da varanda ao anoitecer",
    "enabled": true,
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "last_run_at": "string",
    "name": "Rotina noturna",
    "next_run_at": "string",
    "owner_id": "string",
    "run_at": "2026-06-16T18:00:00-03:00",
    "schedule_type": "cron",
    "timezone": "America/Sao_Paulo",
    "updated_at": "string"
  }
]
```

## Cria um novo agendamento

AUTHORIZATIONS: > *BearerAuth*

REQUEST BODY SCHEMA: application/json  
**required**

### Dados do agendamento

action_device_id	string
action_payload	string
action_target	string
action_type	string
cron_expr	string
days_filter	integer

description	string
name	string
run_at	string
schedule_type	string (Schedule_domain.ScheduleType) Enum: "cron" "one_shot"
timezone	string

## Responses

> 201 Created

> 400 Bad Request

> 500 Internal Server Error

POST /schedule-service/schedules

## Request samples

Payload

Content type  
application/json

Copy

```
{
  "action_device_id": "550e8400-e29b-41d4-a716-446655440004",
  "action_payload": "{\"brightness\": 80}",
  "action_target": "lighting-service",
  "action_type": "turn_on",
  "cron_expr": "0 18 * * *",
  "days_filter": "mon,tue,wed,thu,fri",
  "description": "Liga as luzes da varanda ao anoitecer",
  "name": "Rotina noturna",
  "run_at": "2026-06-16T18:00:00-03:00",
  "schedule_type": "cron",
```

```
"timezone": "America/Sao_Paulo"
}
```

## Response samples

201

400

500

### Content type

application/json

Copy

```
{
  "action_device_id": "550e8400-e29b-41d4-a716-446655440004",
  "action_payload": "{\"brightness\": 80}",
  "action_target": "nats",
  "action_type": "turn_on_light",
  "created_at": "2026-06-16T18:00:00-03:00",
  "cron_expr": "0 18 * * *",
  "days_filter": "mon,tue,wed,thu,fri",
  "description": "Liga as luzes da varanda ao anoitecer",
  "enabled": true,
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "last_run_at": "string",
  "name": "Rotina noturna",
  "next_run_at": "string",
  "owner_id": "string",
  "run_at": "2026-06-16T18:00:00-03:00",
  "schedule_type": "cron",
  "timezone": "America/Sao_Paulo",
  "updated_at": "string"
}
```

## Pré-visualiza as próximas execuções de um agendamento

AUTHORIZATIONS: > *BearerAuth*

REQUEST BODY SCHEMA: application/json  
**required**

schedule\_id e count  
object

## Responses

> 200 OK

> 400 Bad Request

> 403 Forbidden

POST /schedule-service/schedules/preview

### Request samples

#### Payload

Content type  
application/json

Copy

```
{ }
```

### Response samples

200

400

403

Content type  
application/json

Copy Expand all Collapse all

```
{  
  "next": [ ],  
  "schedule_id": "string"  
}
```

## Retorna um agendamento por ID

AUTHORIZATIONS: >

*BearerAuth*

## PATH PARAMETERS

→ **id** required string  
ID do agendamento

## Responses

> 200 OK

> 403 Forbidden

> 404 Not Found

GET /schedule-service/schedules/{id}

## Response samples

200

403

404

### Content type

application/json

Copy

```
{
  "action_device_id": "550e8400-e29b-41d4-a716-446655440004",
  "action_payload": "{\"brightness\": 80}",
  "action_target": "nats",
  "action_type": "turn_on_light",
  "created_at": "2026-06-16T18:00:00-03:00",
  "cron_expr": "0 18 * * *",
  "days_filter": "mon,tue,wed,thu,fri",
  "description": "Liga as luzes da varanda ao anoitecer",
  "enabled": true,
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "last_run_at": "string",
  "name": "Rotina noturna",
  "next_run_at": "string",
  "owner_id": "string",
  "run_at": "2026-06-16T18:00:00-03:00",
  "schedule_type": "cron",
```

```
"timezone": "America/Sao_Paulo",  
"updated_at": "string"  
}
```

## Atualiza um agendamento

AUTHORIZATIONS: > *BearerAuth*

### PATH PARAMETERS

→ id required	string ID do agendamento
------------------	-----------------------------

REQUEST BODY SCHEMA: application/json  
required

### Dados do agendamento

→ action_device_id	string
→ action_payload	string
→ action_target	string
→ action_type	string
→ cron_expr	string
→ days_filter	integer
→ description	string
→ name	string
→ run_at	string
→ timezone	string

### Responses

> 200 OK

> 400 Bad Request

> 403 Forbidden

> 404 Not Found

PUT /schedule-service/schedules/{id} ▼

## Request samples

### Payload

Content type  
application/json

Copy

```
{
  "action_device_id": "550e8400-e29b-41d4-a716-446655440004",
  "action_payload": "{\"brightness\": 80}",
  "action_target": "lighting-service",
  "action_type": "turn_on",
  "cron_expr": "0 18 * * *",
  "days_filter": "mon,tue,wed,thu,fri",
  "description": "Liga as luzes da varanda ao anoitecer",
  "name": "Rotina noturna",
  "run_at": "2026-06-16T18:00:00-03:00",
  "timezone": "America/Sao_Paulo"
}
```

## Response samples

200

400

403

404

Content type  
application/json

Copy

```
{
  "action_device_id": "550e8400-e29b-41d4-a716-446655440004",
  "action_payload": "{\"brightness\": 80}",
  "action_target": "nats",
}
```

```
"action_type": "turn_on_light",
"created_at": "2026-06-16T18:00:00-03:00",
"cron_expr": "0 18 * * *",
"days_filter": "mon,tue,wed,thu,fri",
"description": "Liga as luzes da varanda ao anoitecer",
"enabled": true,
"id": "550e8400-e29b-41d4-a716-446655440000",
"last_run_at": "string",
"name": "Rotina noturna",
"next_run_at": "string",
"owner_id": "string",
"run_at": "2026-06-16T18:00:00-03:00",
"schedule_type": "cron",
"timezone": "America/Sao_Paulo",
"updated_at": "string"
}
```

## Remove um agendamento

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

→ id	string
required	ID do agendamento

## Responses

> 200 OK

> 403 Forbidden

> 404 Not Found

DELETE /schedule-service/schedules/{id}



## Response samples

200

403

404

### Content type

application/json

Copy

```
{
  "property1": "string",
  "property2": "string"
}
```

## Histórico de execuções de um agendamento

AUTHORIZATIONS: > *BearerAuth*

### PATH PARAMETERS

→ id **required** string  
ID do agendamento

### QUERY PARAMETERS

→ limit integer  
Limite de registros (padrão 50)

## Responses

> 200 OK

> 403 Forbidden

> 404 Not Found

GET /schedule-service/schedules/{id}/history



## Response samples

200

403

404

### Content type

application/json

Copy Expand all Collapse all

```
[
- {
  "error_message": "string",
  "executed_at": "string",
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "schedule_id": "string",
  "status": "success"
}
```

## Ativa/desativa um agendamento

AUTHORIZATIONS: >

*BearerAuth*

PATH PARAMETERS

→ id	string
required	ID do agendamento

## Responses

> 200 OK

> 403 Forbidden

> 404 Not Found

PATCH /schedule-service/schedules/{id}/toggle



## Response samples

200

403

404

### Content type

application/json

Copy

```
{
  "action_device_id": "550e8400-e29b-41d4-a716-446655440004",
  "action_payload": "{\"brightness\": 80}",
  "action_target": "nats",
  "action_type": "turn_on_light",
  "created_at": "2026-06-16T18:00:00-03:00",
  "cron_expr": "0 18 * * *",
  "days_filter": "mon,tue,wed,thu,fri",
  "description": "Liga as luzes da varanda ao anoitecer",
  "enabled": true,
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "last_run_at": "string",
  "name": "Rotina noturna",
  "next_run_at": "string",
  "owner_id": "string",
  "run_at": "2026-06-16T18:00:00-03:00",
  "schedule_type": "cron",
  "timezone": "America/Sao_Paulo",
  "updated_at": "string"
}
```

## Notifications

Notifications Service API

## Health check do serviço

## Responses

> 200 OK

GET /notifications-service/health

## Response samples

200

Content type  
application/json

Copy

```
{  
  "property1": "string",  
  "property2": "string"  
}
```

## Lista notificações do usuário autenticado

AUTHORIZATIONS: > *BearerAuth*

## Responses

> 200 OK

> 500 Internal Server Error

GET /notifications-service/notifications

## Response samples

200

500

### Content type

application/json

Copy Expand all Collapse all

```
[
  - {
    "created_at": "2026-06-16T18:00:00-03:00",
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "location": "Sala de estar",
    "message": "Sensor da sala detectou movimento",
    "read": "false",
    "sensor_id": "550e8400-e29b-41d4-a716-446655440001",
    "title": "Movimento detectado",
    "type": "motion_detected",
    "user_id": "550e8400-e29b-41d4-a716-446655440000"
  }
]
```

## Marca uma notificação como lida

AUTHORIZATIONS: > *BearerAuth*

PATH PARAMETERS

---

→ id string  
required ID da notificação

---

## Responses

> 200 OK

> 403 Forbidden

> 404 Not Found

> 500 Internal Server Error

PATCH /notifications-service/notifications/{id}/read



## Response samples

200

403

404

500

### Content type

application/json

Copy

```
{
  "property1": "string",
  "property2": "string"
}
```

## Gera token para vincular conta Telegram

AUTHORIZATIONS: >

*BearerAuth*

## Responses

> 200 OK

> 500 Internal Server Error

POST /notifications-service/telegram/link



## Response samples

200

500

### Content type

application/json

Copy

```
{  
  "property1": "string",  
  "property2": "string"  
}
```

## Remove vinculacao com Telegram

AUTHORIZATIONS: > *BearerAuth*

### Responses

> 200 OK

> 404 Not Found

> 500 Internal Server Error

DELETE /notifications-service/telegram/link

### Response samples

200

404

500

Content type  
application/json

Copy

```
{  
  "property1": "string",  
  "property2": "string"  
}
```

# Retorna preferencias Telegram do usuario

AUTHORIZATIONS: > *BearerAuth*

## Responses

> 200 OK

> 404 Not Found

GET /notifications-service/telegram/preferences

## Response samples

200

404

### Content type

application/json

Copy

```
{
  "active": "true",
  "chat_id": "123456789",
  "enabled_types": "motion,light",
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "linked_at": "2026-06-16T18:00:00-03:00",
  "user_id": "550e8400-e29b-41d4-a716-446655440000"
}
```

# Atualiza tipos de notificacao habilitados para Telegram

AUTHORIZATIONS: > *BearerAuth*

REQUEST BODY SCHEMA: application/json  
required

enabled\_types

→ property name\*  
additional property    Array of strings

---

## Responses

> 200 OK

> 400 Bad Request

> 500 Internal Server Error

PUT /notifications-service/telegram/preferences

### Request samples

Payload

Content type  
application/json

Copy    Expand all    Collapse all

```
{
  - "property1": [
    "string"
  ],
  - "property2": [
    "string"
  ]
}
```

### Response samples

200

400

500

Content type  
application/json

Copy

```
{  
  "property1": "string",  
  "property2": "string"  
}
```