

# SCADA System

用户手册

版本 1.0 — 2026年4月

# 目录

前言

快速入门 — 15 分钟

目录

**1. 登录注册**

**2. 仪表板**

**3. 模式**

**4. 架构生成器**

**5. 模式查看器**

**6. 设备**

**7. PLC**

**8. 序列**

**9. 配方和变量**

**10. 事件和事故**

**11. 选项**

**12. 脉冲模块**

**13. 批次管理**

**14. 生产线**

**15. 趋势**

**16. 警报**

**17. 数学函数**

**18. I/O 点**

**19. I/O 配置**

**20. 图形状态**

**21. 硬件配置器**

**22. 用户管理**

**23. 角色管理**

**24. 活动日志**

**25. 设置**

**26. 人工智能助手**

**27. PLC 工程导出**

**28. 项目导出和导入**

**29. 坦克**

应用程序导航 (侧边栏)

**全局 UI 元素**

键盘快捷键

实际例子

故障排除和常见问题解答

安全和最佳实践

术语表

附录 A: WebSocket 通道参考

**附录 B: API 参考 (汇总表)**

**附录 C: 错误和警报代码**

文档版本: 1.0

日期: 2026年4月

支持的界面语言: EN, RU, UK, DE, FR, ES, PL, AR, ZH 等。

# 前言

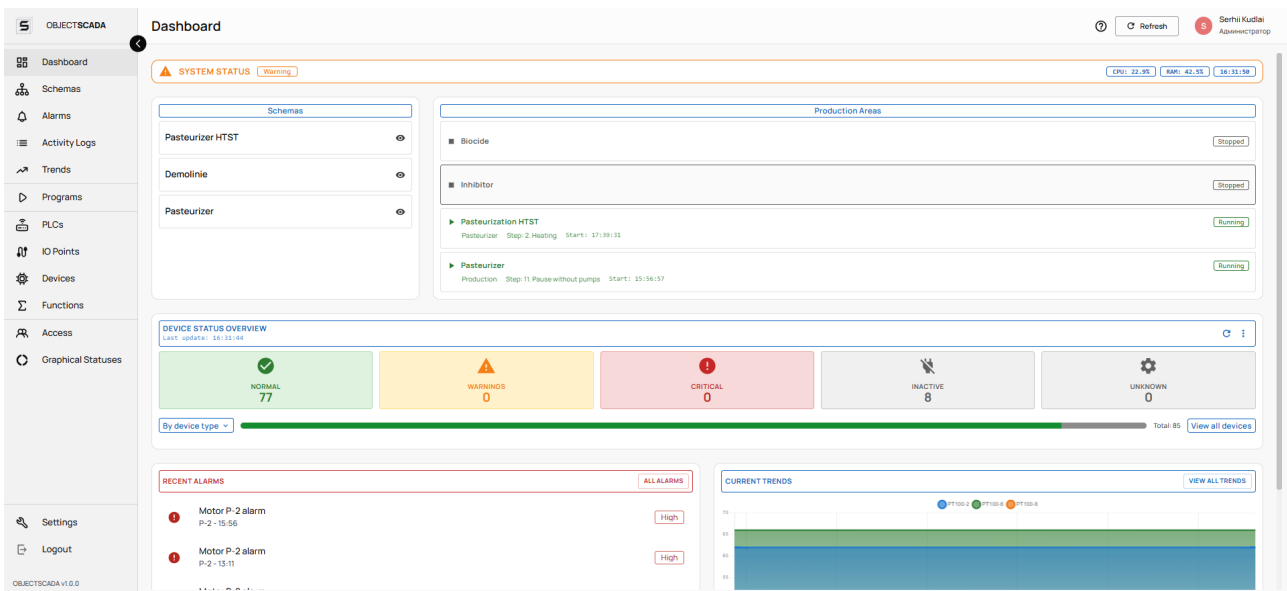
## 关于系统

SCADA 系统  
 是一个现代化的监控和数据采集平台，专为下一代工业自动化而设计。该系统涵盖了从硬件配置、助记词模式设计到生产线实时运行的整个过程控制周期。

与可追溯到桌面应用程序和专有协议时代的传统 SCADA 系统不同，该平台建立在现代 Web 堆栈和微服务原则之上。这意味着操作员可以通过任何浏览器（车间的工业 PC、实验室的平板电脑或办公室的笔记本电脑）使用该系统，无需安装其他软件。

该系统覆盖自动化生产的全生命周期：

- 设计 — 视觉助记符模式构建器、设备配置、控制算法（序列）和配方的创建
- 部署 — 自动 PLC 代码生成（西门子 SCL/XML）、I/O 点配置、通过 TIA Portal Agent 部署
- 操作 — 实时过程监控、设备控制、趋势分析、批次管理
- 维护 — 活动日志、警报、诊断、自动更新、备份



4 个关键系统屏幕的拼贴：仪表盘（左上）、带有运行模式的模式查看器（右上）、带有图表的趋势（左下）、序列编辑器（右下）。展示了功能的广度。

## 目标受众

角色	他们如何使用该系统
过程自动化工程师	创建助记符模式、配置设备和序列、配置 PLC、生成 TIA Portal 代码
工艺技术员	开发生产配方、配置工艺参数、定义警报阈值
线路操作员	实时观察助记符模式、控制设备（启动/停止、设定点）、处理批次和配方
值班经理/领班	通过仪表盘监控所有线路的状态、分析趋势、使用警报日志
系统管理员	管理用户和角色、升级系统、配置 PLC 和网络连接
工厂管理	通过浏览器随时随地查看报告、效率趋势、生产线状态

## 主要优势

### 1. 完全基于网络的平台

该系统可以在任何现代浏览器中运行，无需安装客户端软件。这与传统系统 (WinCC、Citect、InTouch) 有根本区别，传统系统需要在每个工作站上安装客户端，绑定到特定操作系统 (通常是 Windows)，并且每个席位都需要单独的许可证。

SCADA  
SCADA

- 从网络上的任何位置 (Intranet 或 VPN) 访问
- 适用于任何设备: PC、平板电脑、工业终端
- 无操作系统锁定
- 同时向所有用户即时推出 UI

### 2. 数据实时更新

两层 WebSocket 架构可立即向所有连接的客户端提供更改:

- Rust WebSocket 服务器 (端口 8001) — 高性能交付批量设备更新 (每秒最多数千个参数)。用 Rust 编写, 可实现最小延迟和最大吞吐量。
- Django Channels WebSocket — 用于控制命令的双向流量 (电机启动/停止、阀门打开/关闭、PID 设置点更改)。

增量更新仅传输更改的字段, 而批量发送 (每 250 毫秒) 可在数百个参数同时更改时最大限度地减少网络负载。

### 3. 集成 PLC 代码生成

这是大多数 SCADA 系统中没有的独特功能: 根据设备、序列、配方和事故的配置, 系统自动为西门子 TIA Portal 生成完整的项目:

- 11 个功能块 (SCL), 适用于每种设备类型 (电机 FB、阀门 FB、PID FB、AI/AO/DI/DO FB、COS FB、计数器 FB、定时器 FB、储罐 FB)
- 序列 FB — 每个序列的状态机
- 事件/事故 FB — 事件和事故的处理程序
- 数据块 — 设备实例 DB 和配方 DB
- OB1 Main — 主程序以正确的顺序调用每个 FB
- XML 标签表 — SimaticML 标签表可直接导入到 TIA Portal

这将部署新生产线的时间从几周缩短到几小时: 工程师在 SCADA 中配置流程, 单击“生成”, 将 ZIP 上传到 TIA Portal, 然后 PLC 就可以运行了。

### 4. 视觉助记符模式构建器

基于 Konva.js (HTML5 Canvas) 构建的强大模式编辑器无需单独的 SCADA 图形编辑器:

- 设备、管道、文本和图像的拖放放置
- 设备状态自动可视化 (颜色、流动动画)
- 支持背景图像 (P&ID 图纸作为图层)
- 具有可配置盖子形状和自动传感器放置功能的储罐构造器
- 多语言标签和翻译
- 无限缩放和平移
- 多选和分组操作

### 5. 灵活的配方和顺序系统

配方系统遵循 ISA-88 (批量控制) 并清晰地分离:

- 序列 — 算法：步骤顺序和转换条件
- 步骤配方（配方） — 每个步骤的特定设备值
- 生产配方 — 特定产品的一组参数

相同的序列可以与不同的配方一起使用，在相同的设备上生产不同的产品。配方变量参数化过程而不修改算法。

## 6. 批次管理

完整的生产批次管理系统：

- 创建与配方和生产线绑定的批次
- 通过生产线阶段跟踪批次进度
- 具有优先级的批次队列
- 阶段之间自动转移
- 每批次的完整历史记录

## 7. 人工智能助手

基于 LLM 的嵌入式 AI 助手（通过 OpenRouter）极大地加速了系统的工作：

- 自然语言控制：《打开电机MOT\_001》、《哪些设备处于报警状态？》
- 自动 P&ID 分析：上传图像→设备和连接识别→架构上的放置建议
- 上下文相关的帮助和解释
- 使用工具进行代理循环（最多 15 次迭代）以完成复杂任务

## 8. 多语言支持

全面支持 10 多种界面语言，并且能够翻译用户内容，而不仅仅是 UI：

- 接口：EN、UK、RU、DE、FR、PL、AR（带 RTL）等
- 设备、序列、步骤、配方和选项名称 - 通过 “name\_translations” 字段
- 模式文本 — 通过 “translations” 字段
- 模式名称 — 通过 name\_translations 字段
- 即时语言切换，无需重新加载

## 9. 可扩展的基于角色的访问

精细的权限系统可让您精确控制谁可以做什么：

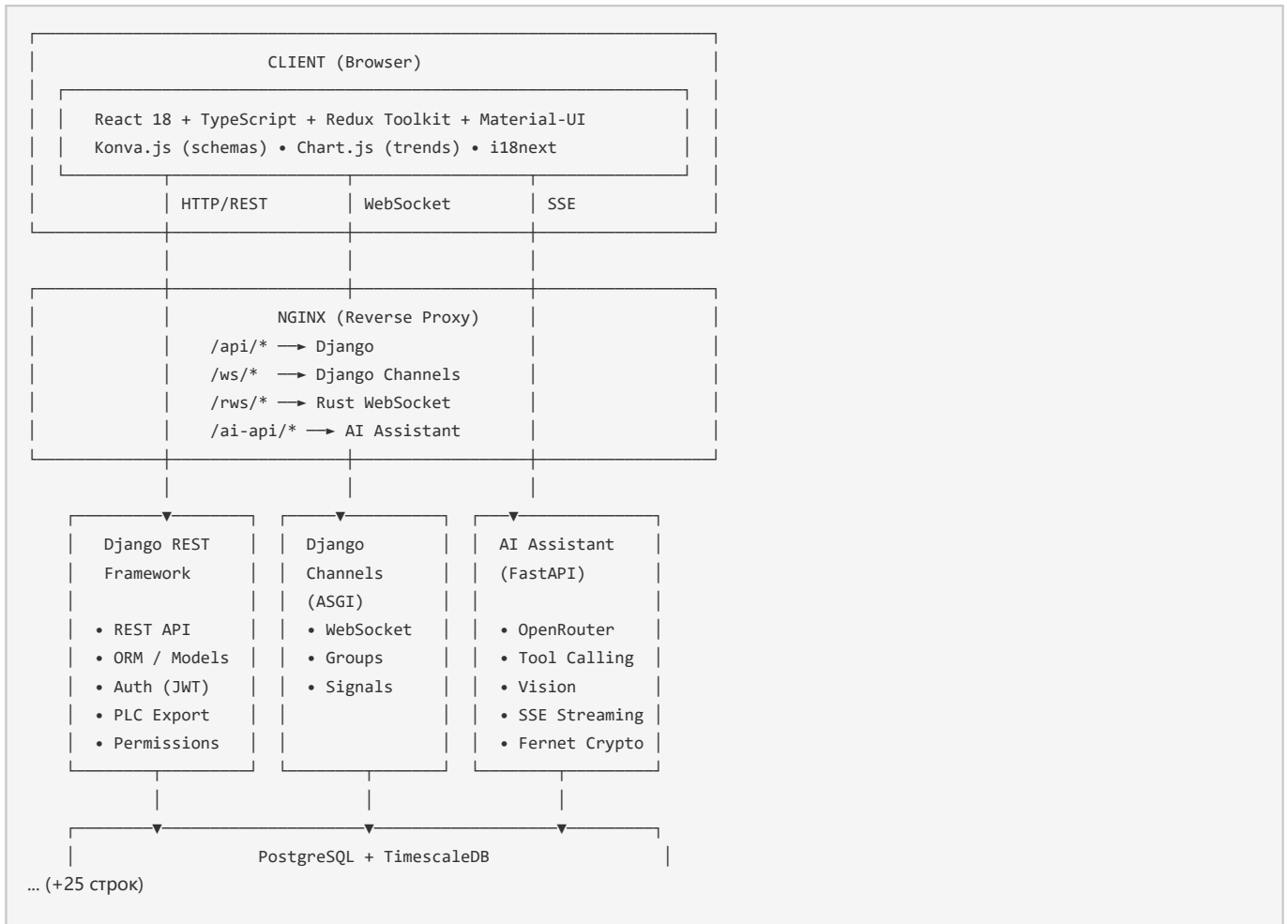
- 具有跨 12 个类别和许多操作的权限矩阵的角色
- 每个模式的访问控制
- 全面审核用户操作
- 用于访问控制的IP过滤

## 10. 工业级可靠性

- 具有指数退避功能的自动 WebSocket 重新连接
- 双 WebSocket：Rust 提高速度 + Django 控制（后备）
- 完整数据库备份和恢复
- 项目导出/导入以在安装之间迁移
- 通过回滚和试运行更新系统
- 升级后冒烟测试

## 系统架构

该系统基于微服务架构构建，职责分离清晰：



## 组件概述

### 前端 (React + TypeScript)

在用户浏览器中运行的客户端应用程序。基于 React 18 构建，使用 TypeScript 实现类型安全，Redux Toolkit 实现状态管理，Material-UI 实现组件。助记符模式是通过 Konva.js (HTML5 Canvas) 绘制的；通过 Chart.js 的趋势。通过 i18next 实现了 10 多种语言的多语言支持，包括 RTL (阿拉伯语)。前端通过 REST API (axios)、WebSocket (本机 API) 和 SSE (用于 AI 流) 与后端进行通信。

### NGINX (反向代理)

所有请求的入口点。它在微服务之间路由流量：

- /api/\* → Django REST API
- /ws/\* → Django 通道 (WebSocket)
- /rws/\* → Rust WebSocket 服务器 (高性能, 只读)
- /ai-api/\* → AI助手 (FastAPI)
- /media/\* → 静态文件 (图标、图像)

### Django REST 框架 (后端 API)

业务逻辑核心：

- 40 多种资源类型的所有 CRUD 操作
- 身份验证和授权 (JWT + RBAC)
- 用户、角色和权限管理
- PLC 代码生成 (11 种功能块类型、状态机、数据块、标签表)
- 项目导出/导入

- 系统维护 (升级、备份、诊断)

## Django 通道 (ASGI WebSocket)

双向实时通信:

- 17+ WebSocket 通道类型 (设备、模式、序列、批次、事故、数学函数、活动、系统)
- 用于群组广播的Redis通道层
- 处理控制命令 (设备启动/停止、批量管理)
- 更新 Django 模型更改的信号

## 人工智能助手 (FastAPI)

独立的人工智能辅助微服务:

- Python 3.11 上的 FastAPI (端口 8080)
- 与 OpenRouter API 集成 (LLM 支持工具调用)
- Orchestrator 代理循环 (最多 15 次迭代), 具有 4 个工具类别
- 通过视觉模型进行 P&ID 图像分析
- SSE 响应流 (令牌、工具调用、工具结果、完成)
- API 密钥加密 (Fernet/AES)
- JWT 身份验证 (通过 Django /api/users/me/ 代理)
- 共享 PostgreSQL 数据库中的专用表 (前缀 "aiassistant")
- 蒸馏器迁移

## PostgreSQL + TimescaleDB

所有服务的单一数据库:

- 核心表 (core\_\*) — 设备、模式、序列、配方、角色、用户
- TimescaleDB 超表 — 具有自动时间分区和压缩功能的历史趋势数据
- AI 助手表 (ai\_assistant\_\*) — 会话、消息、设置

在处理大数据量时, TimescaleDB 的时间序列查询速度比普通 PostgreSQL 快几个数量级。

## Redis

通讯总线:

- Django 通道的通道层 (WebSocket 组)
- Pub/Sub 用于 Django 和 Rust Worker 之间的通知
- 缓存频繁请求

## Rust Worker (scada-worker)

用 Rust 编写的高性能实时处理服务:

- 序列引擎 (engine.rs) — 序列执行: 当前步骤跟踪、转换条件评估、设备激活/停用、配方应用
- 事件处理器 — 序列事件处理
- 事故处理器 — 事故检测、事故记录创建
- 罐容积计算器 — 根据液位和压力传感器数据定期计算罐容积 (每 1000 毫秒)
- 数学函数计算器 — 数学函数重新计算
- DB Batchers — 批量写入 PostgreSQL 更改 (I/O 优化)
- WebSocket Batchers — 批量 WebSocket 更新 (250 毫秒间隔)
- Rust WebSocket Server (端口 8001) — 专用的只读设备状态流服务器。批量更新比 Django Channels 快得多。

该组件选择 Rust 是因为需要最小延迟、可预测的内存使用以及每秒处理数千次更新而无需垃圾收集暂停的能力。

## 相对于传统 SCADA 系统的优势

标准	传统SCADA	这个系统
客户	胖客户端 (安装在每台 PC 上)	Web 浏览器 (无需安装)
操作系统	仅限 Windows	任何 (Windows、Linux、macOS、Android、iOS)
许可	每个座位 (每个工作站)	服务器许可证 (无限客户端)
更新	每台 PC 上的手册	自动、集中 (所有客户端同时)
移动访问	无或单独的模块	开箱即用的响应式界面
PLC 代码生成	无 (手动编程)	自动从SCADA配置
人工智能助手	无	内置 (P&ID 分析、控制、帮助)
多语言	有限 (仅限 UI)	完整 (UI + 内容 + RTL)
实时	专有协议	WebSocket + Rust (亚秒级延迟)
食谱	基础支持	ISA-88 (序列+配方+生产配方)
批次管理	单独的模块 (额外许可证)	内置
趋势分析	内置, 有限	TimescaleDB (PB 级数据, 自动压缩)
审计	基本	包含 WebSocket 更新的完整日志
备份/恢复	外部工具	内置 (备份、导出、导入、回滚)
部署成本	高 (许可证+安装+配置)	低 (Docker、自动化部署)
可扩展性	封闭生态系统	开放REST API + WebSocket + SSE

## 系统要求

### 服务器端

组件	最低	推荐
中央处理器	4 核	8+ 核心
内存	8GB	16+ GB
磁盘	20 GB 固态硬盘	100+ GB SSD (趋势)
操作系统	Linux (Ubuntu 22.04+) 或 Windows Server 2019+	Linux (Ubuntu 22.04 LTS)
码头工人	Docker 引擎 24+	Docker Engine 24+ 与 Docker Compose v2
PostgreSQL	15+ 使用 TimescaleDB	16+ 使用 TimescaleDB 2.x
Redis	7+	7+

### 客户端 (浏览器)

组件	要求
浏览器	Chrome 90+、Firefox 90+、Edge 90+、Safari 15+
屏幕分辨率	最低1280×720, 推荐1920×1080+
网络	与服务器的稳定连接 (LAN/VPN)

## 部署

该系统作为 Docker 容器提供，并使用单个命令进行部署：

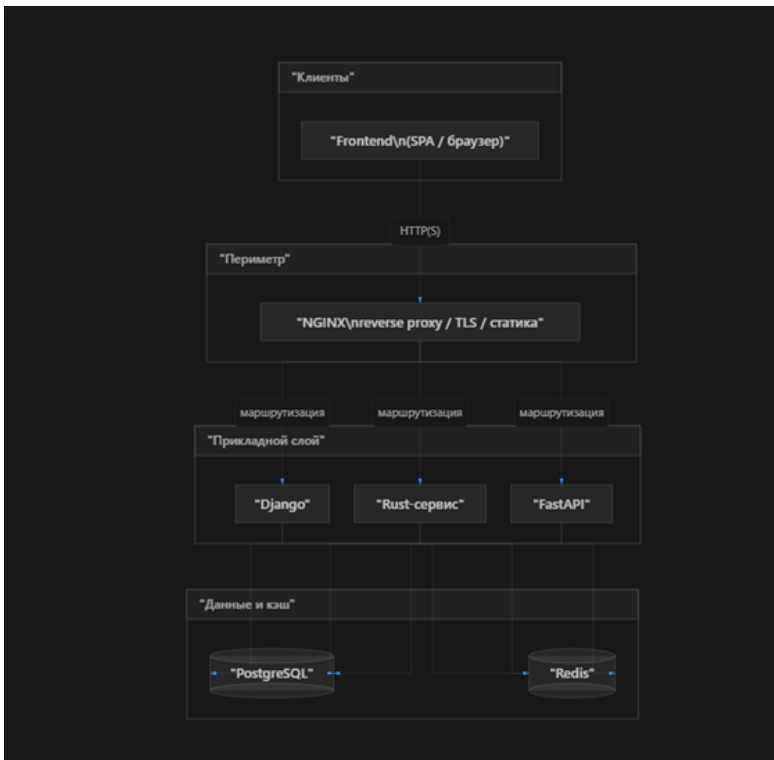
```
docker-compose up -d
```

- 后端 — Django REST API + Django 通道
- scada-worker — Rust Worker
- ai-assistant — FastAPI 人工智能服务
- postgres — PostgreSQL + TimescaleDB
- redis — Redis
- nginx — 反向代理

The screenshot shows the Docker Desktop interface with a list of containers. The 'scada' container is highlighted, and its details are shown in the table below.

Name	Container ID	Image	Port(s)	CPU (%)	Memory usage...	Memory (%)	Actions
scada	-	-	-	261.79%	5.25GB / 47.29GB	231.51%	[Stop] [Refresh] [Delete]
pgbouncer-1	37745af812a4	edoburu/pgbouncer:6432:6432	6432:6432	0.01%	7.02MB / 256MB	2.74%	[Stop] [Refresh] [Delete]
redis-1	bd5f098ba5b7	redis:7-alpine:6379:6379	6379:6379	4.69%	13.77MB / 15.49GB	0.09%	[Stop] [Refresh] [Delete]
qdrant-1	c79f9c64e56b	qdrant/qdrant:6333:6333	6333:6333	0.18%	303.6MB / 512MB	59.29%	[Stop] [Refresh] [Delete]
db-1	b5ee9b813460	scada-db:5432:5432	5432:5432	102.07%	835.2MB / 10GB	8.16%	[Stop] [Refresh] [Delete]
db-maintenance-1	1b65060619e9	scada-db:6001:6001	6001:6001	0%	6.07MB / 64MB	9.48%	[Stop] [Refresh] [Delete]
scada-worker-1	df88d42f6246	scada-worker:8001:8001	8001:8001	81.96%	13.24MB / 512MB	2.59%	[Stop] [Refresh] [Delete]
backend-1	263a6e6fcd62	scada-backend:8000:8000	8000:8000	10.41%	604.2MB / 15.49GB	3.81%	[Stop] [Refresh] [Delete]
frontend-1	6c52a11b8dbd	scada-frontend:3000:3000	3000:3000	59.52%	2.74GB / 4GB	68.4%	[Stop] [Refresh] [Delete]
ai-assistant-1	024774f07b22	scada-ai-assistant:8080:8080	8080:8080	2.95%	788MB / 1GB	76.95%	[Stop] [Refresh] [Delete]

带有 `docker-compose ps` 输出的终端：所有 6 个处于 «Up» 状态的容器、它们的端口和正常运行时间。



系统架构图：通过线路连接的组件块（REST、WebSocket、Redis Pub/Sub、SQL）。颜色编码：蓝色 — 前端、绿色 — Django、橙色 — Rust、紫色 — AI、灰色 — 基础设施。

## 快速入门 — 15 分钟

本节将引导您从干净的安装到使用一台电机和一个阀门的工作助记符模式。如果这是您第一次启动 SCADA，请从这里开始。

### 步骤 1. 首次登录 (2 分钟)

- 在浏览器中打开安装地址（例如 “http://localhost” 或 “http://scada.local” ）。
- 在登录屏幕上输入安装程序创建的管理员凭据（默认为 “admin” ，密码在安装过程中设置）。
- 首次登录时，系统可能会请求激活许可证 - 输入密钥并按 激活。如果没有有效的许可证，大多数功能都会受到限制。
- 成功登录后，您将进入仪表板。

提示。在登录表单的左下角选择界面语言 - 它是按用户保存的。

### 步骤 2. 创建第一个 PLC (2 分钟)

即使您仅进行模拟工作，系统也需要至少一台 PLC 来连接设备。

- 在侧边栏中打开 PLC。
  - 按右上角的 «+»。
- 3、填写：
- 名称：PLC\_DEMO
  - IP地址：192.168.0.100（稍后可以更改）
  - 类型：Siemens S7-1200（或任何可用选项）
  - 机架/插槽：0 / 1
  - 按保存。PLC 状态将为 “离线” ——这对于演示设置来说是正常的。

### 步骤 3. 创建第一个设备 (3 分钟)

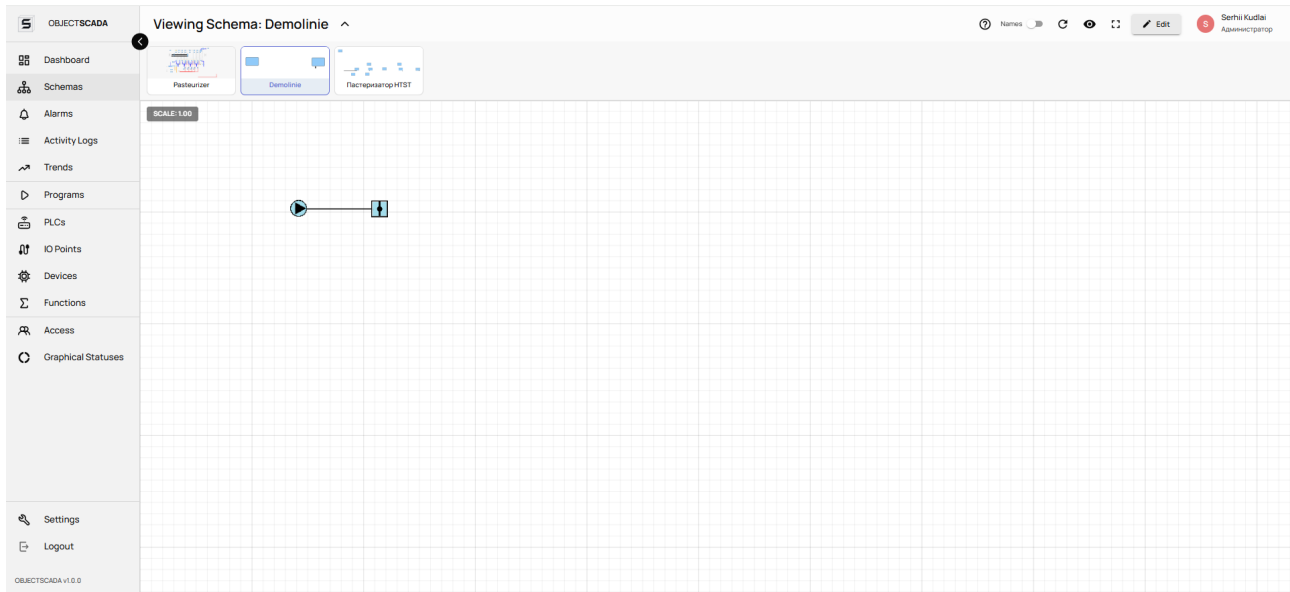
添加一台电机和一个阀门。

- 转至 设备 → «+» → 电机。
- 2、填写：
- ID: mot\_001
  - 名称: 供应泵
  - PLC: PLC\_DEMO
  - 描述: 《演示电机》
  - 按保存。
  - 同样创建一个阀门 («+» → 阀门) :
  - ID: val\_001
  - 名称: 管线阀
  - PLC: PLC\_DEMO

**重要。** 设备 ID 是文本标识符 (mot\_001, 而不是数字)。它用于每个 WebSocket 通道、PLC 导出和转换条件中。预先选择一个命名方案并坚持下去。

## 步骤 4. 创建第一个助记符模式 (3 分钟)

- 打开 架构 → «+»。
- 输入名称: 《演示线路》并按创建。
- 在架构列表中, 单击新架构上的 铅笔 (编辑) 图标 - 架构生成器 打开。
- 在左侧工具栏中按设备 (); 在对话框中选择 mot\_001 → 添加。电机出现在画布上 - 将其拖动到位。
- 再次按设备 → val\_001 → 添加 — 将阀门放置在电机旁边。
- 按管道 (), 单击电机, 然后单击阀门 — 出现连接线。
- 按 Ctrl+S 或顶栏中的 保存 按钮。



简单的演示方案：一台电机（绿色圆圈）通过管道连接到阀门（黄色三角形），背景为画布网格。

## 步骤 5. 运行查看器并进行操作 (3 分钟)

- 在顶部工具栏中按 View — 切换到 Schema Viewer。
- 单击电机图标 — 将打开一个控件弹出窗口。将模式切换为手动, 然后按开始。指示灯变为绿色 — 电机正在“运行”。
- 单击阀门 → 打开。如果在管道上启用了过程监控, 则会出现流动动画。
- 按电机上的停止 — 动画停止。

恭喜您——您拥有了第一个可实时控制的工作助记符模式。

## 第 6 步: 下一步做什么 (2 分钟)

现在你有了一个基本的骨架。接下来的步骤取决于您的目标:

目标	去哪里
自动化流程 (根据条件自动启动/停止)	第 8 节《序列》和《牛奶巴氏灭菌器》实例
为不同产品配置配方	第 9 节《配方和变量》
连接真实的 PLC I/O	第 18 节《I/O 点》和第 19 节《I/O 配置》
配置警报阈值	第 10 节《事件和事故》
生成 TIA Portal 项目	第 27 节《PLC 项目导出》
分析趋势	第 15 节“趋势”

提示。每个页面的标题中都有一个 «?» 按钮 — 它会启动带有元素突出显示的交互式导览。

# 目录

## 简介:

- 快速入门 — 15 分钟

## 主要部分:

- 登录注册
- 仪表盘
- 模式
- 架构生成器
- 模式查看器
- 设备
- PLC
- 序列
- 配方和变量
- 事件和事故
- 选项
- 脉冲模块
- 批次管理
- 生产线
- 趋势
- 闹钟
- 数学函数
- I/O点数
- I/O 配置
- 图形状态
- 硬件配置器
- 用户管理
- 角色管理
- 活动日志
- 设置

## 26. 【人工智能助手】(#26-ai-assistant)

- PLC工程导出
- 项目导出和导入
- 坦克

## 补充部分:

- 实际例子
- 故障排除和常见问题解答
- 安全和最佳实践
- 词汇表

## 附录:

- 附录 A: WebSocket 通道参考
- 附录 B: API 参考
- 附录 C: 错误和警报代码

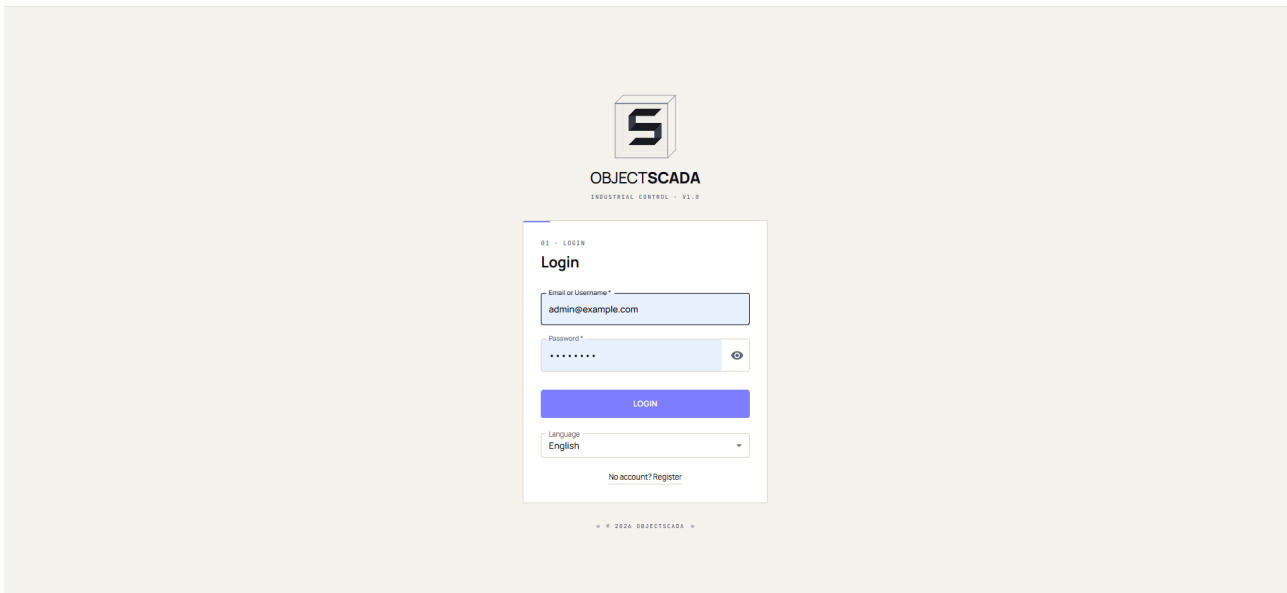
# 1. 登录注册

## 对于用户

系统打开时，您首先看到的是登录屏幕。

屏幕元素：

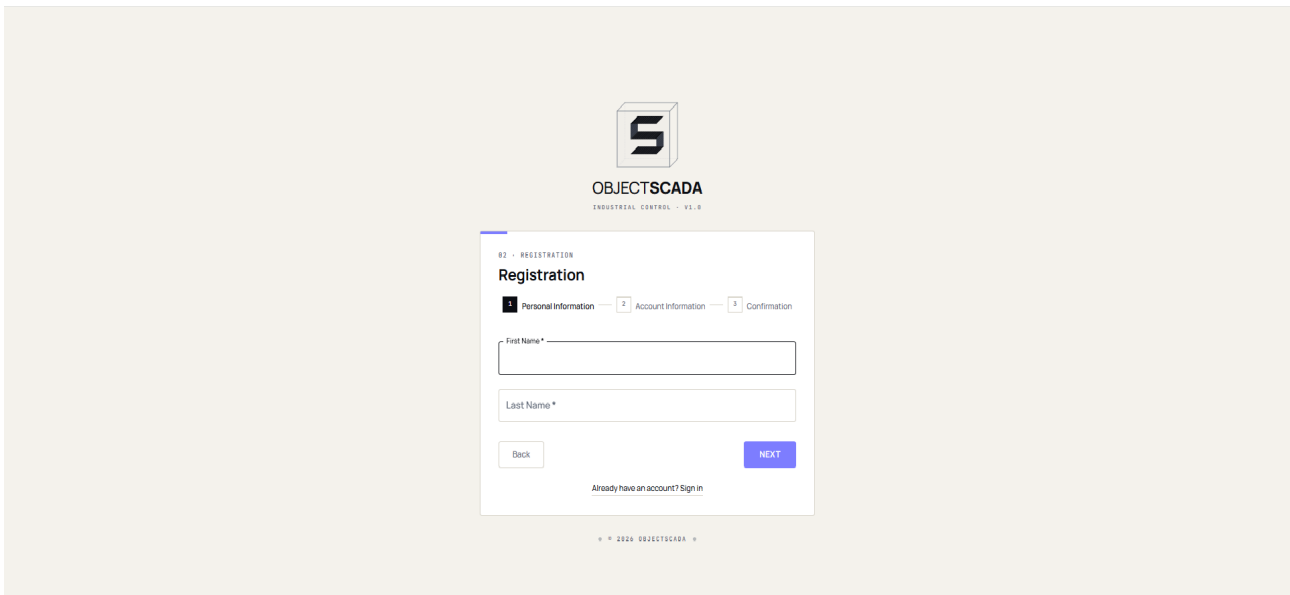
元素	描述
«用户名/电子邮件» 字段	输入您的登录名或电子邮件地址
“密码” 字段	输入密码。右侧的 图标可切换可见性
语言选择器	下拉选择界面语言（登录后保存）
«登录»按钮	执行登录
“忘记密码?” 链接	前往密码恢复
«注册»链接	前往注册表



登录页面：用户名/电子邮件和密码字段（带有眼睛图标）、语言下拉菜单、登录按钮、忘记密码? 和注册链接。深色/浅色主题取决于设置。

注册屏幕：

元素	描述
«用户名» 字段	唯一的登录名
«电子邮件» 字段	电子邮件地址
“密码” 字段	带强度指示器的密码
«确认密码» 字段	密码重复
“全名” 字段	名字和姓氏
«注册»按钮	创建帐户



注册表格：用户名、电子邮件、密码（带强度指示器）、确认密码、全名、注册按钮。

## 技术文档

### 身份验证API:

方法	端点	描述	请求正文
POST	/api/auth/login	登录	{"用户名": "...", "密码": "..."}
POST	/api/auth/register	注册	{ "用户名": "...", "电子邮件": "...", "密码": "...", "名字": "...", "姓氏": "... }
GET	/api/auth/me	当前用户	—
POST	/api/auth/token/refresh/	刷新 JWT 令牌	{"刷新": "..."}

### 登录响应:

```
{
  "access": "eyJhbGciOi...",
  "refresh": "eyJhbGciOi...",
  "user": {
    "id": 1,
    "username": "admin",
    "email": "admin@example.com",
    "roles": [...]
  }
}
```

授权：每个受保护端点都需要标头“授权：承载”。

## 2. 仪表板

### 对于用户

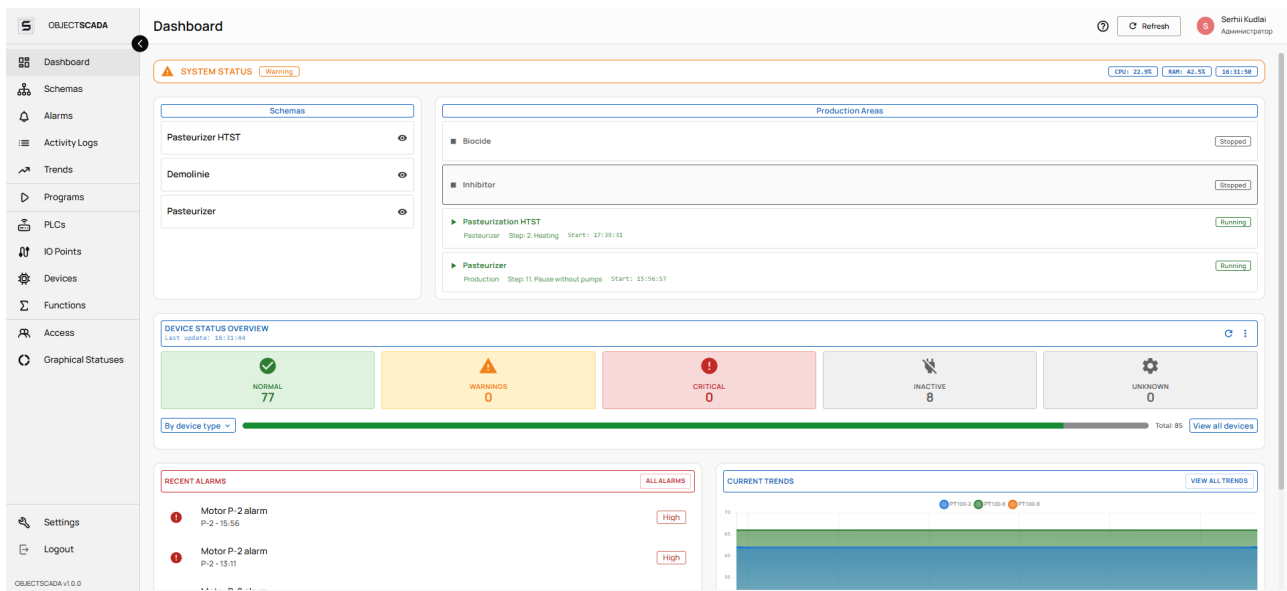
仪表板是系统的中心页面，显示整体状态。

屏幕元素：

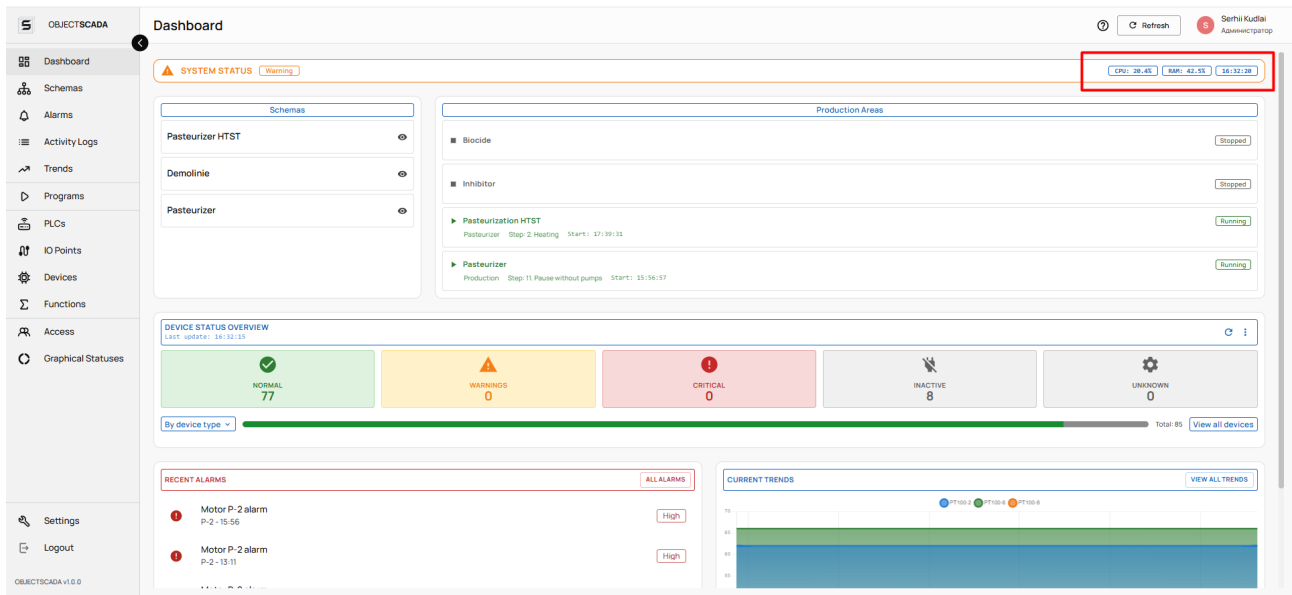
元素	描述
统计卡	快速摘要：设备数量、活动模式、事故、序列
国家概况	设备和系统状态的视觉指示器
报警列表	最近的警报带有严重性颜色编码
小趋势	关键参数微型图表
序列状态	每个运行序列的当前步骤和进度
系统指标	CPU、RAM、正常运行时间、温度、磁盘空间
快速访问按钮	快速导航至核心功能

标题按钮：

按钮	描述
刷新	刷新仪表板数据
旅游	推出仪表板元素的交互式导航
出口项目	打开完整项目导出对话框
导入项目	打开项目导入对话框



完整的仪表板视图：顶部是统计卡，左侧是警报块，右侧是小趋势，下面是序列状态。左侧的导航侧边栏。



系统指标块：CPU 使用率 (%)、RAM 使用率 (%)、磁盘使用率、正常运行时间、网络统计信息。每个参数都有一个颜色指示器（绿色/黄色/红色）。

## 技术文档

### 系统指标API:

方法	端点	描述
GET	/api/system/info/	系统信息
GET	/api/system/health-status/	系统健康状况

### WebSocket: 系统指标 (每 2 秒流一次)

连接: ws://ws/system

```
{
  "type": "system_metrics",
  "cpu": 45.2,
  "memory": 67.8,
  "disk": 34.1,
  "network": {"rx": 1024, "tx": 512},
  "uptime": 86400,
  "temperature": 52.0,
  "lastUpdate": "2026-04-07T12:00:00Z"
}
```

### WebSocket: PLC 状态 (每 2 秒传输一次)

```
{
  "type": "plc_status",
  "is_active": true,
  "status": "running",
  "service_running": true,
  "timestamp": "2026-04-07T12:00:00Z"
}
```

## 3. 模式

### 对于用户

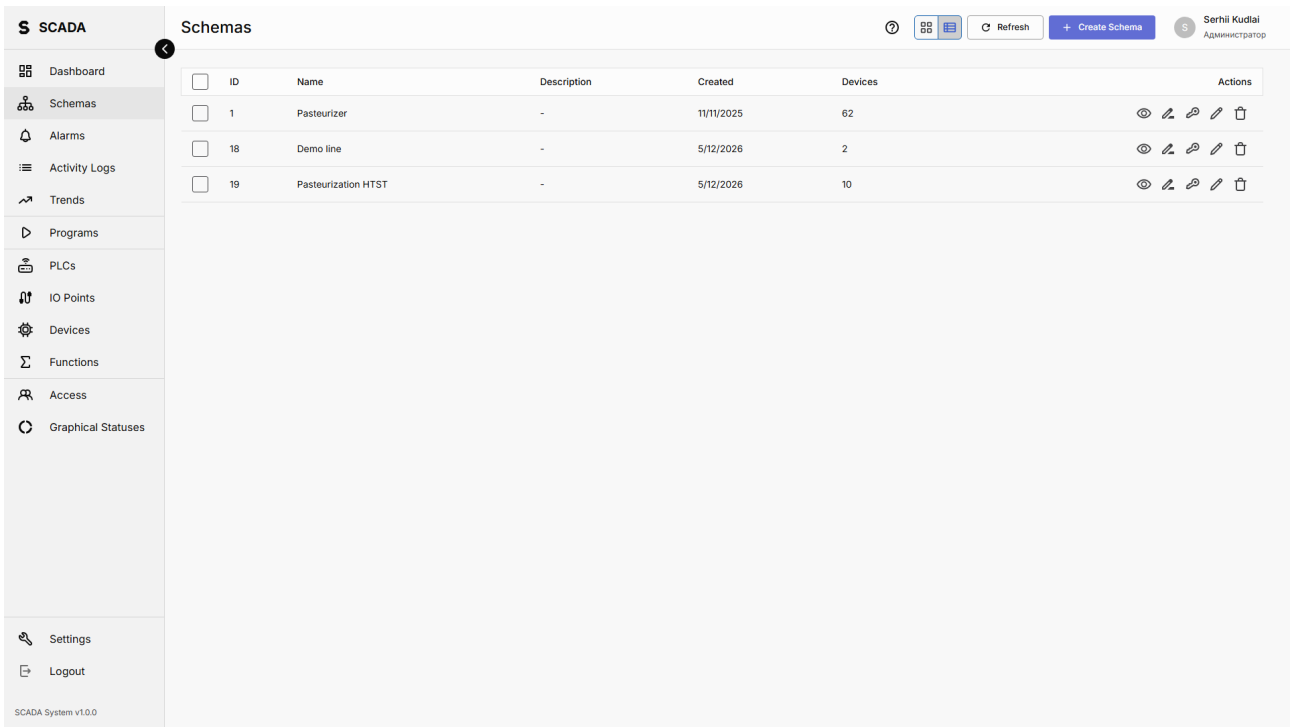
助记图是生产过程的可视化图表。它们是监视和控制的主要工具。

列出屏幕元素：

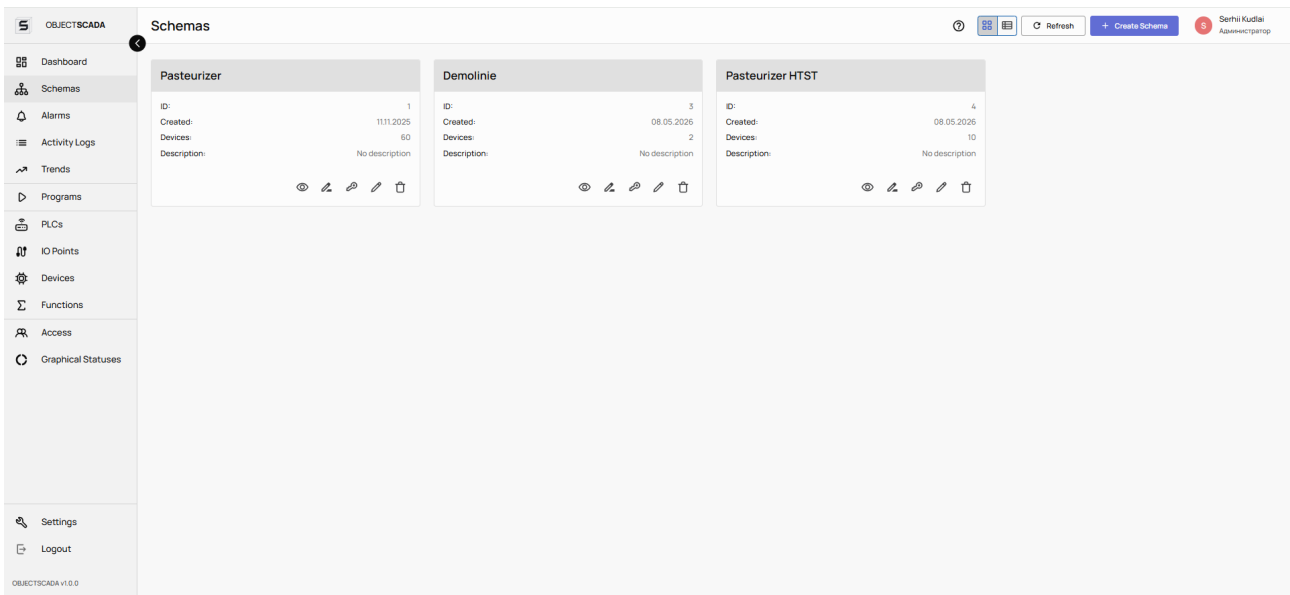
元素	描述
架构列表/网格	具有助记模式的卡片或表格行
查看切换	网格/列表（右上角的图标）
«+»按钮	创建新架构
搜索	按名称搜索
刷新按钮	重新加载列表

每个架构操作：

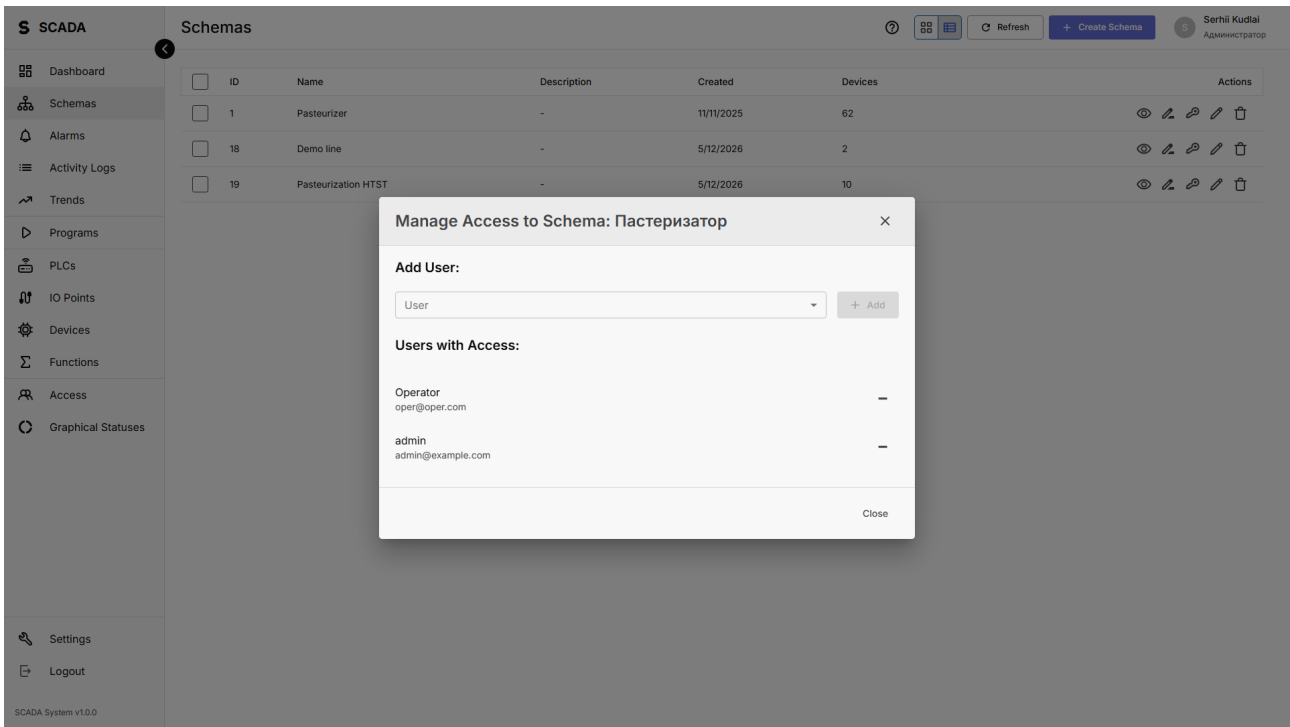
按钮	图标	描述
查看		以视图模式打开架构（架构查看器）
编辑		在编辑器（架构生成器）中打开架构
删除		删除架构（需要确认）
访问管理		配置用户对架构的访问
重命名		更改名称和名称翻译
设置		模式参数（描述、翻译）



架构列表页面作为卡片网格。每张卡片都显示架构预览、名称、修改日期、操作按钮（查看、编辑、删除）。右上角的“+”按钮。



表格形式的同一页面：列“名称”、“描述”、“创建”、“修改”、“操作”。



访问管理对话框：带有复选框、按钮“添加”和“保存”的用户列表。

## 技术文档

### 模式 API:

方法	端点	描述	参数
GET	/api/schemas/	所有架构的列	? 搜索=...&排序=...
POST	/api/schemas/	创建架	{"name": "...", "description": "...", "name_translations": {...}}
GET	/api/schemas/{id}/	检索架	—
PUT	/api/schemas/{id}/	更新架	完整模式对象
PATCH	/api/schemas/{id}/	部分更	{"name": "...", "description": "...", "name_translations": {...}}
DELETE	/api/schemas/{id}/	删除架	—
GET	/api/schemas/{id}/user_access/	具有访问权限	—
POST	/api/schemas/{id}/user_access/	授予访问权限	{"user_id": 1}
DELETE	/api/schemas/{id}/user_access/	撤销访问权限	{"user_id": 1}

## 4. 架构生成器

### 对于用户

模式生成器是一个强大的可视化工具，用于创建和编辑流程图。它在画布上运行，可以对元素进行拖动、缩放和旋转。

#### 左侧工具栏

按钮	图标	描述
选择		选择并移动元素
设备		将目录中的设备添加到架构中
管道/信号		绘制管道（设备之间的连接）
文字	T	添加文字注释
图片		添加背景图像（P&ID 底图）
序列		将序列绑定到架构
选项		添加序列选项按钮
配方变量		添加配方变量显示
生产线		新增生产线
数学函数	▪	添加数学函数
坦克		打开坦克构造器（单独按钮）
创建设备		快速创建新设备

#### 顶部工具栏

按钮	描述
保存	保存架构的当前状态
撤消 (Ctrl+Z)	撤消上一个操作
重做 (Ctrl+Y)	重做已撤消的操作
查看	切换到架构查看器
«名称» 切换	在架构上显示/隐藏设备名称
架构名称	显示当前模式名称

#### 画布控件

按钮	描述
放大 (+)	增加变焦
缩小 (-)	减小变焦
网格	显示/隐藏画布网格
背景颜色	更改架构背景颜色
背景透明度	背景透明度滑块
折叠	折叠/展开左侧工具栏

## 右侧属性面板

当在架构上选择一个元素时，该面板就会出现，并且取决于元素类型：

对于设备 (DevicePropertyPanel)：

物业	描述
位置 X、Y	画布上的坐标
旋转	元件旋转 (0–360°)
尺寸 (宽、高)	宽度和高度
颜色	元素颜色 (调色板)
镜子	水平/垂直翻转
显示名称	设备的开/关标题
图形状态	绑定到状态模板
配置	设备特定参数

对于管道 (PipePropertyPanel)：

物业	描述
名称	管道名称
颜色	线条颜色
厚度	线条粗细 (像素)
姓名可见度	显示/隐藏姓名
过程监控	启用流动动画

对于文本 (TextPropertyPanel)：

物业	描述
文字	内容
字体大小	文字大小
颜色	文字颜色
粗体/斜体	造型
翻译	多语言文本翻译

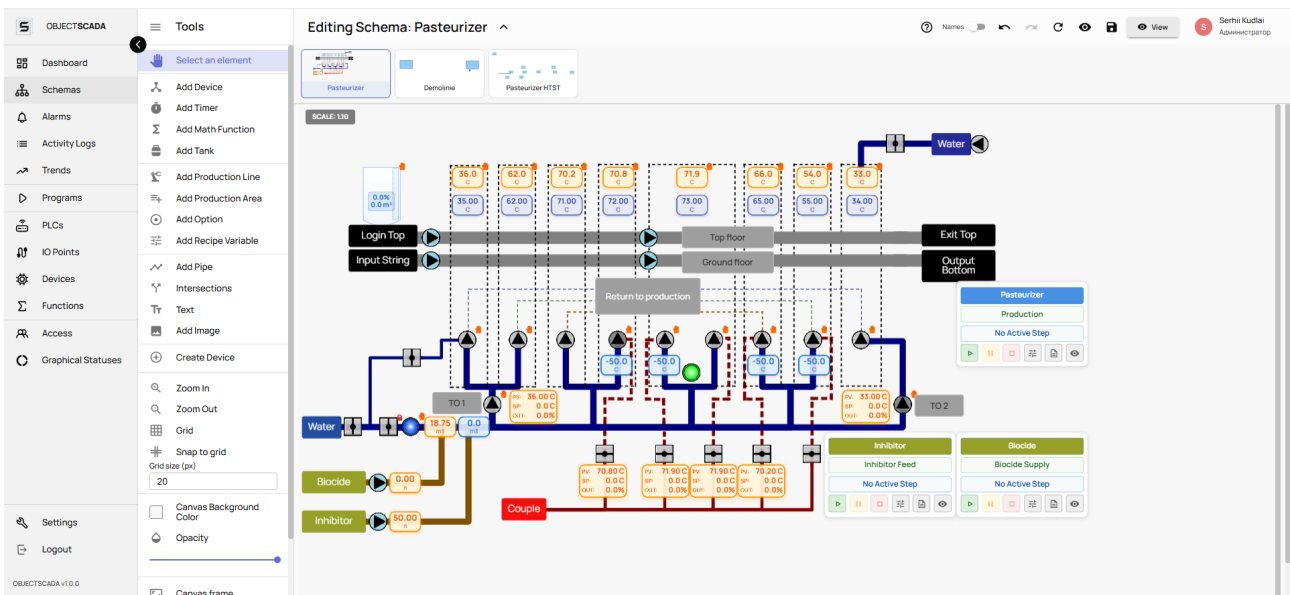
对于序列 (SequencePropertyPanel)：

物业	描述
职位	坐标
尺寸	宽×高
显示	显示设置 (名称、状态、步骤)




## 编辑器对话框

对话	描述
设备对话框	用于放置在架构上的设备目录选择器。按类型过滤。
管道对话框	管道名称、类型和可见性设置
添加序列对话框	选择一个序列来绑定到模式





对话	描述
添加选项对话框	选择一个序列选项
添加生产线对话	挑选一条生产线
添加配方变量对话框	选择一个配方变量
数学函数对话框	选择一个数学函数
储罐构造对话框	储罐构造器：设备选择器、盖子类型 (FLAT/DOME/DISH/CONE)、尺寸、SVG 预览、传感器自动放置
图像对话框	上传背景图片
文本对话框	添加/编辑文本







完整编辑器视图：左侧工具栏带有标志性按钮，顶部栏带有“保存”、“撤消”、“重做”，中央画布带有架构（设备、管道、文本），右侧属性面板用于所选元素。

-  Add Timer
-  Add Math Function
-  Add Tank


---

-  Add Production Line
-  Add Production Area
-  Add Option
-  Add Recipe Variable





---

-  Add Pipe
-  Intersections
-  Text
-  Add Image


---

-  Create Device


---

-  Zoom In
-  Zoom Out
-  Grid
-  Snap to grid
- Grid size (px)

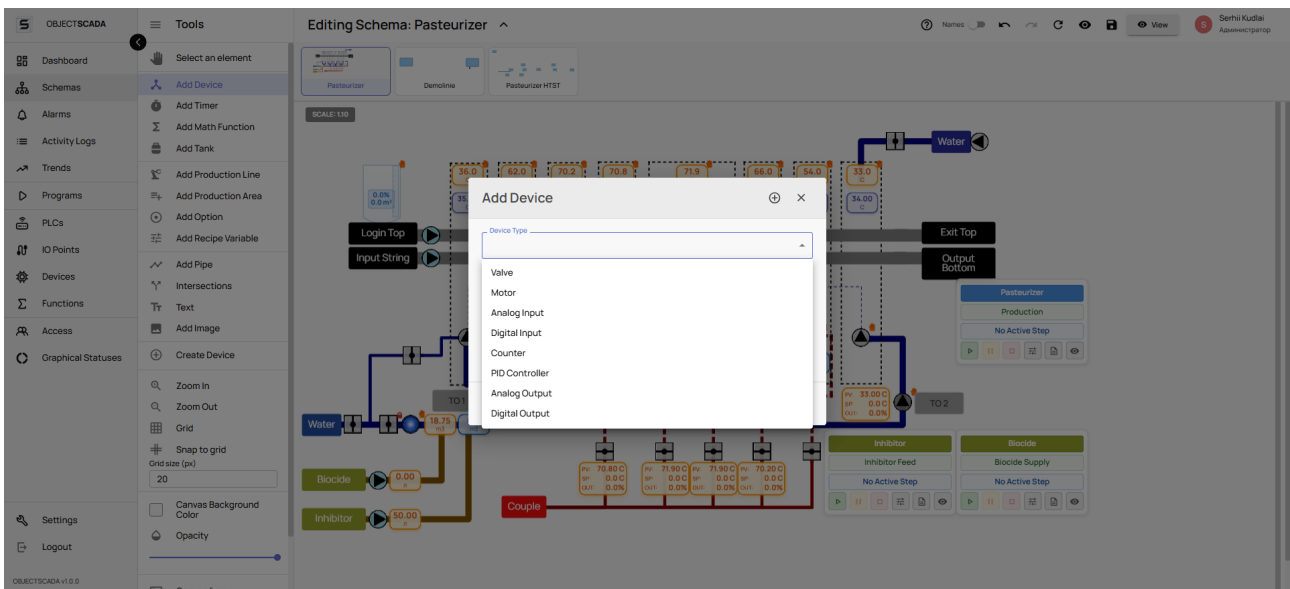
---

- Canvas Background Color
-  Opacity

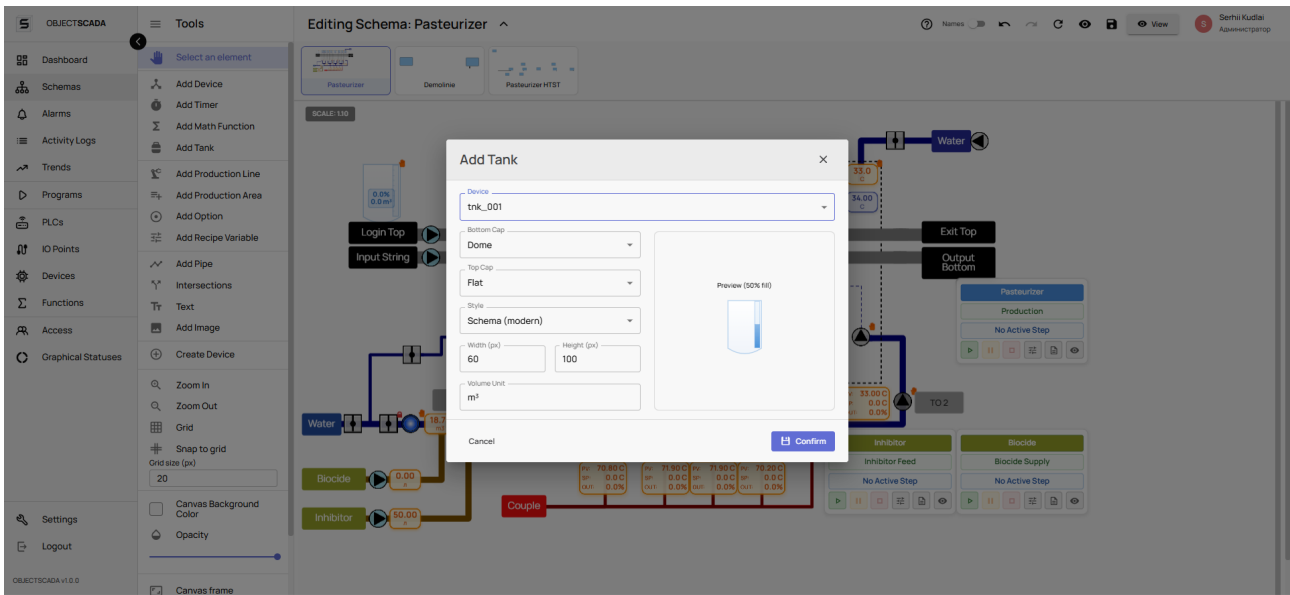
---

-  Canvas frame
- Frame preset

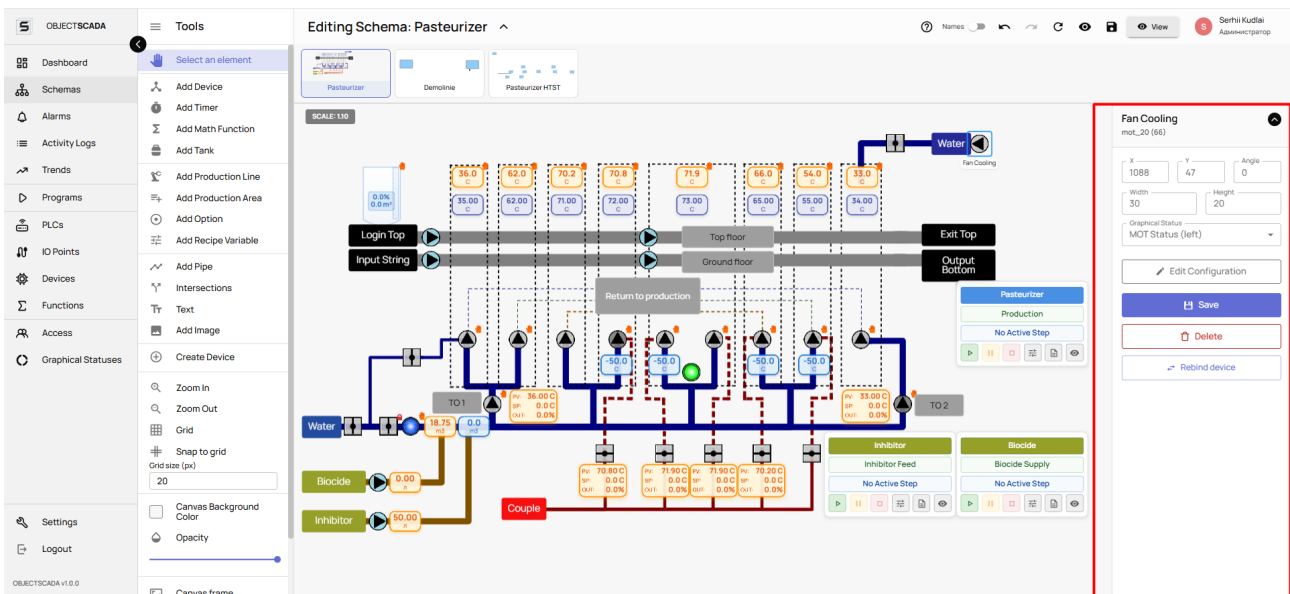
左侧工具栏的特写，其中标有每个按钮（选择、设备、管道、文本、图像、序列、选项、配方变量、生产线、数学函数、罐）。



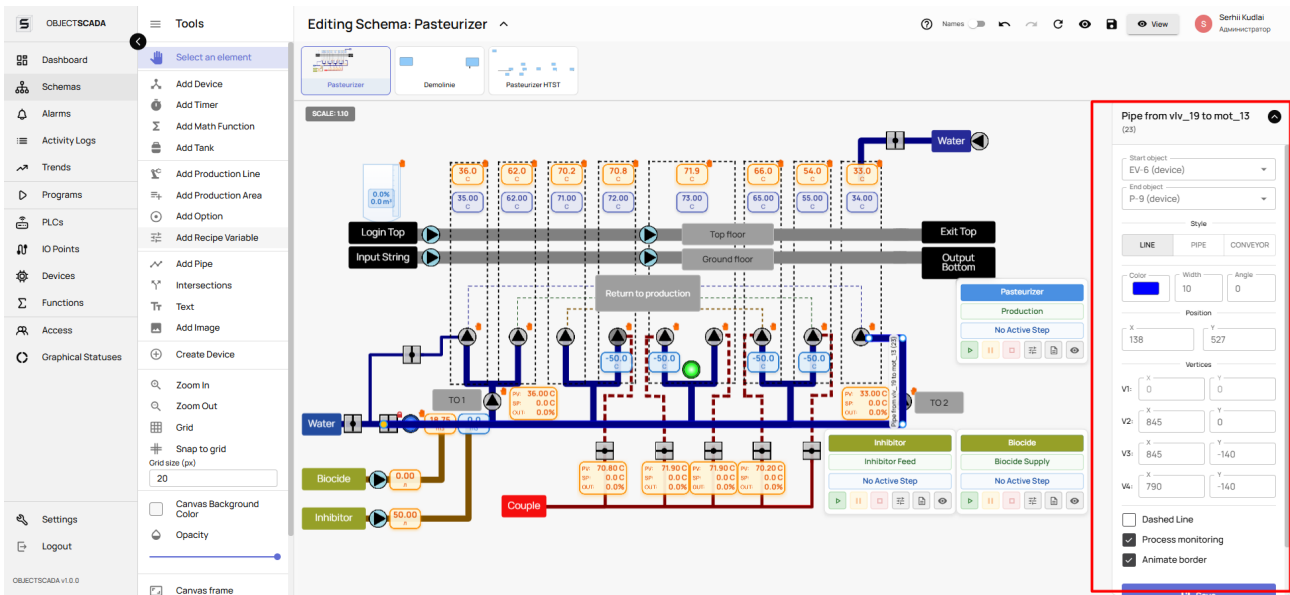
设备选择对话框：按类型过滤的设备列表（电机、阀门、AI、AO、DI、DO、PID、计数器、定时器、储罐、COS）、搜索字段、“添加”按钮。



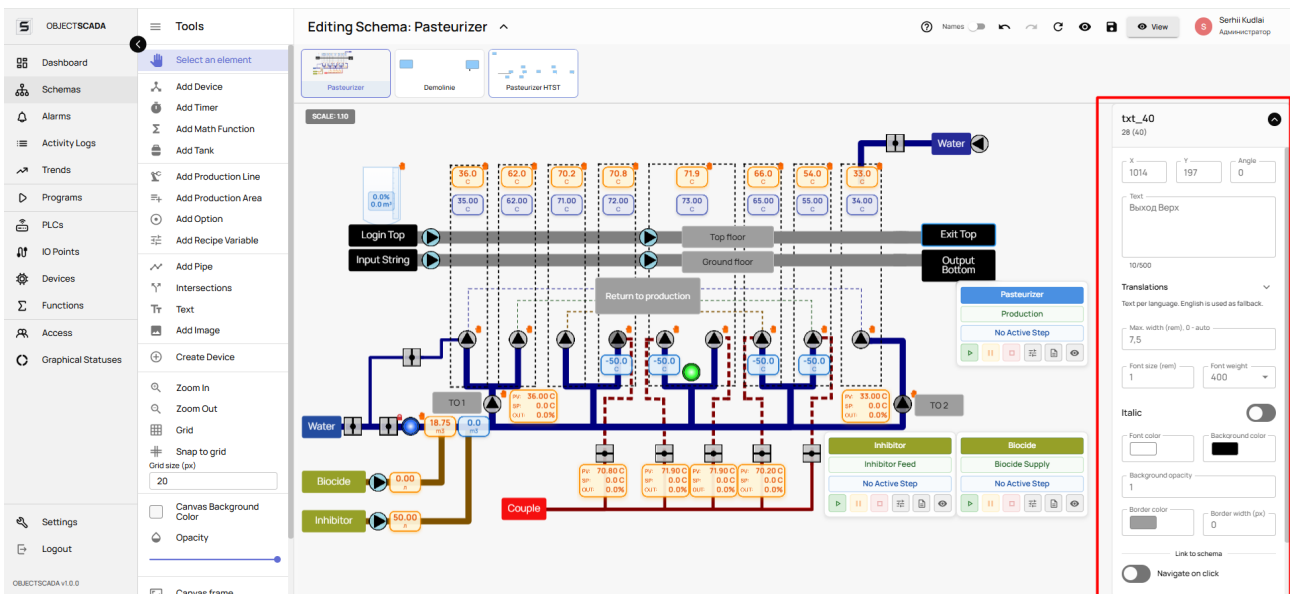
储罐构造器：顶盖和底盖的下拉菜单（平面、圆顶、碟形、圆锥形）、宽度/高度字段、储罐形状的 SVG 预览、储罐设备选择器。



选择设备时的右侧属性面板：位置 X/Y、角度、宽度/高度、颜色选择器、镜像/显示名称切换。



选择管道时的右侧属性面板：名称字段、颜色选择器、厚度滑块、显示名称切换、过程监控切换。



选择文本时的右侧属性面板：文本字段、字体大小、颜色、粗体/斜体切换、包含每种语言字段的翻译块。

## 技术文档

### 架构元素 API:

方法	端点	描述
POST	/api/schemas/{id}/add_motor/	添加电机
POST	/api/schemas/{id}/add_valve/	添加阀门
POST	/api/schemas/{id}/add_analog_input/	添加人工智
POST	/api/schemas/{id}/add_analog_output/	添加 AO
POST	/api/schemas/{id}/add_digital_input/	添加 DI
POST	/api/schemas/{id}/add_digital_output/	添加 DO
POST	/api/schemas/{id}/add_pid_controller/	添加PID
POST	/api/schemas/{id}/add_counter/	添加计数器

方法	端点	描述
POST	/api/schemas/{id}/add_variable_speed_drive/	添加 VSD
POST	/api/schemas/{id}/add_sequence/	添加序列
POST	/api/schemas/{id}/add_option/	添加一个选
POST	/api/schemas/{id}/add_production_line/	新增生产线
POST	/api/schemas/{id}/add_recipe_variable/	添加配方变
POST	/api/schemas/{id}/add-math-function/	添加数学函
POST	/api/schemas/{id}/bulk_add_schema_devices/	批量添加
PATCH	/api/schemas/{id}/update_positions/	批量更新仓
PATCH	/api/schemas/{id}/devices/{deviceId}/	更新元素
DELETE	/api/schemas/{id}/remove_device/	删除一个元素

添加设备请求正文：

```
{
  "device_id": "mot_001",
  "content_type": "motor",
  "position_x": 100,
  "position_y": 200,
  "angle": 0,
  "configuration": {
    "color": "#FF0000",
    "mirror_x": false,
    "mirror_y": false,
    "show_name": true,
    "tank_visual": {
      "bottom_type": "DOME",
      "top_type": "FLAT",
      "width": 60,
      "height": 100
    }
  }
}
```

管道API:

方法	端点	描述
GET	/api/pipes/?schema_id={id}	架构上的所有管道
POST	/api/pipes/	创建管道
PATCH	/api/pipes/{id}/	更新管道
DELETE	/api/pipes/{id}/	删除管道

## 5. 模式查看器

### 对于用户

视图模式提供过程的实时可视化和实时数据更新。

特点：

特色	描述
实时设备状态	颜色和动画实时更新
单击设备	打开一个包含控件和信息的弹出窗口
放大	鼠标滚轮或 «+» / «-» 按钮
潘	右键拖动
报警栏	屏幕顶部/底部 — 当前活动警报
名称切换	显示/隐藏设备标题

### 设备控制弹出窗口

单击设备会打开一个选项卡式弹出窗口：

控制选项卡：

对于每种设备类型：

类型	控制
电机	开始/停止按钮、自动/手动切换、频率控制
阀门	打开/关闭按钮、自动/手动、%、流量
模拟输入 (AI)	自动/手动、数值输入、刻度指示器
模拟输出 (AO)	自动/手动、数值输入、最小/最大
数字输入 (DI)	自动/手动、ON/OFF 指示灯、互锁
数字输出 (DO)	自动/手动、开/关按钮、互锁
PID控制器	设定点、自动/手动、P/I/D 参数、错误和输出
计时器	启动/停止/重置，持续时间设置
柜台	复位、当前值、设定值
坦克	液位、体积、传感器数据

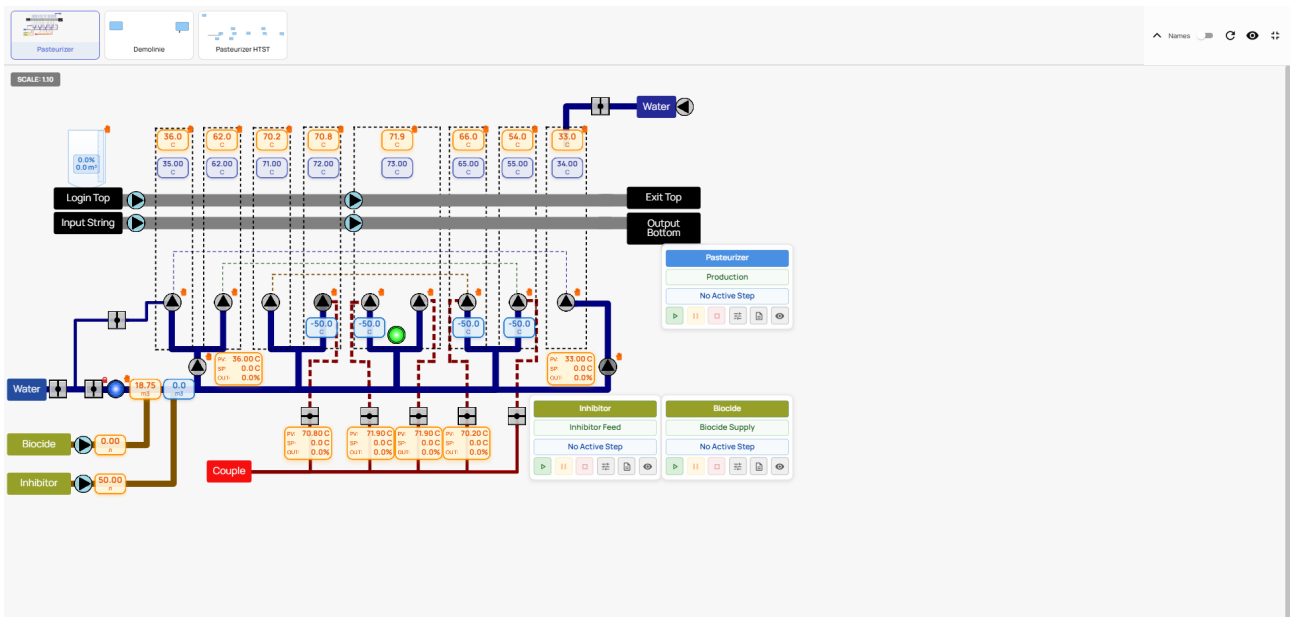
状态选项卡：当前实时状态。

表格选项卡：参数的表格表示。

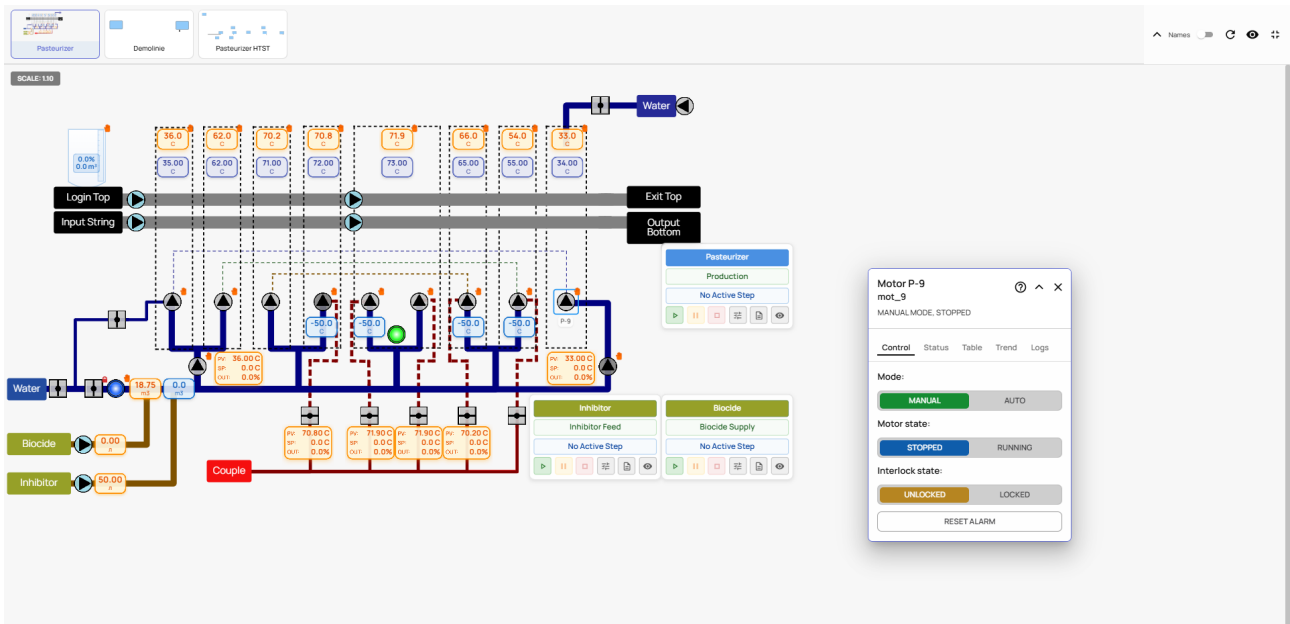
趋势选项卡：价值历史的微型图表。

日志选项卡：设备事件日志。

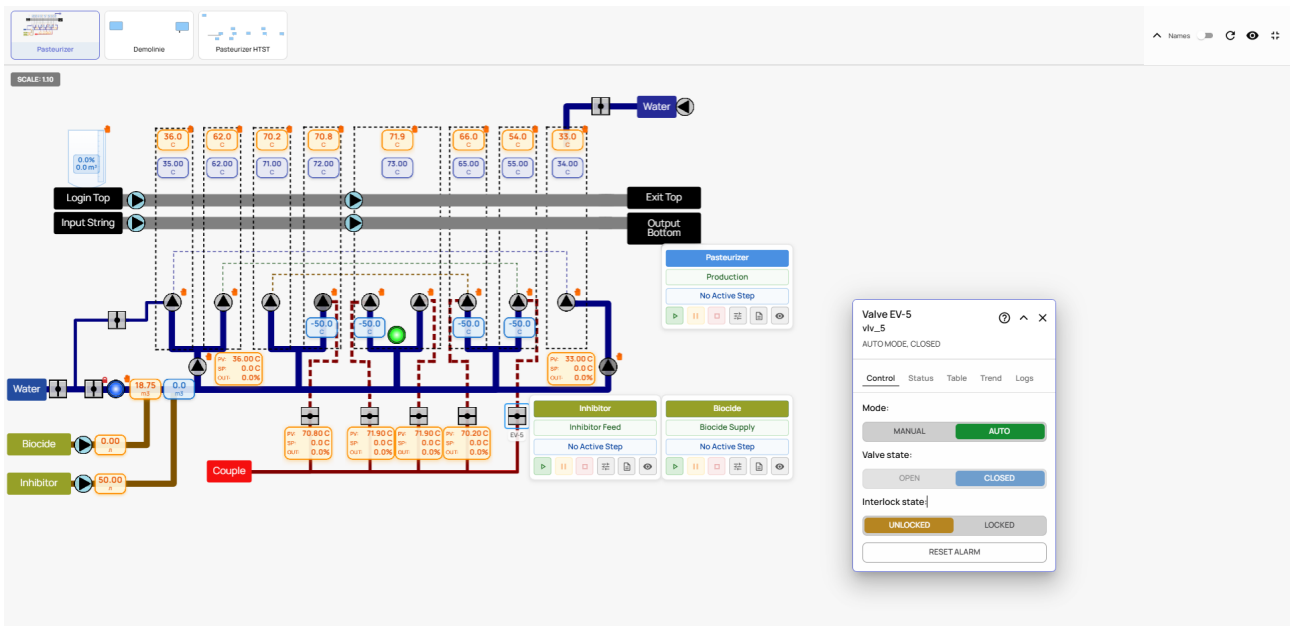
“转换条件”选项卡：活动序列条件（如果设备参与）。



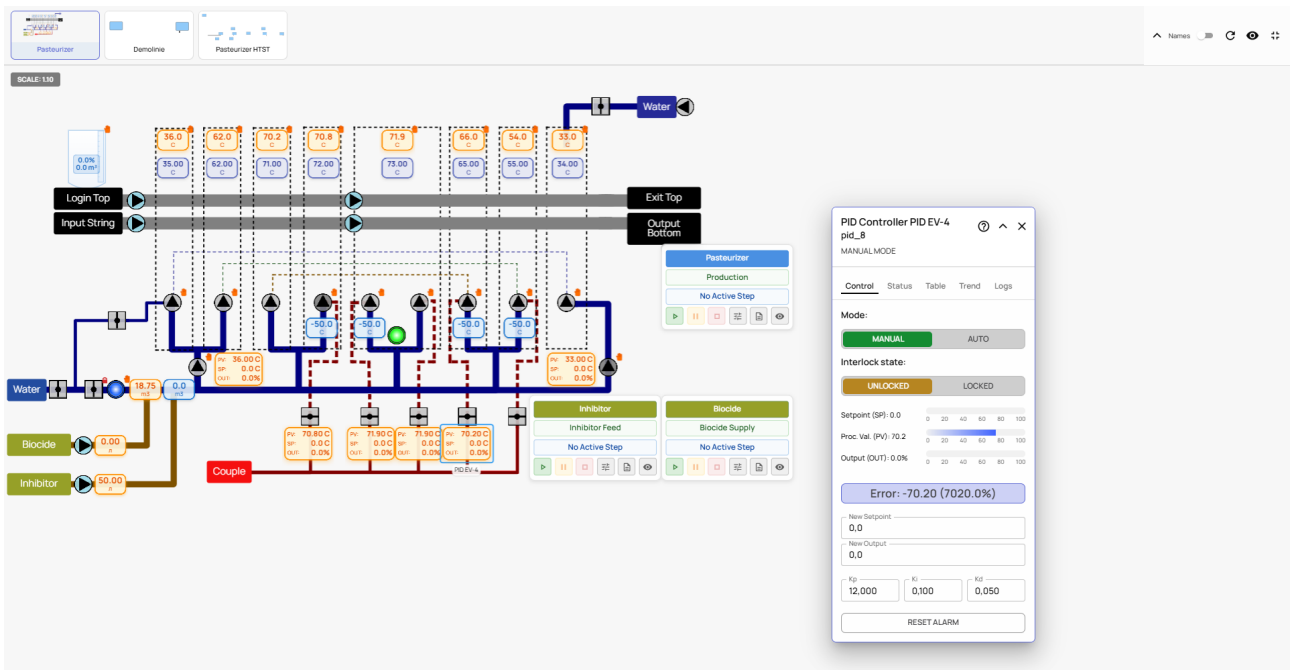
完整模式查看器：带有沿管道流动动画的彩色设备、带有当前步骤的序列块以及顶部的警报栏。



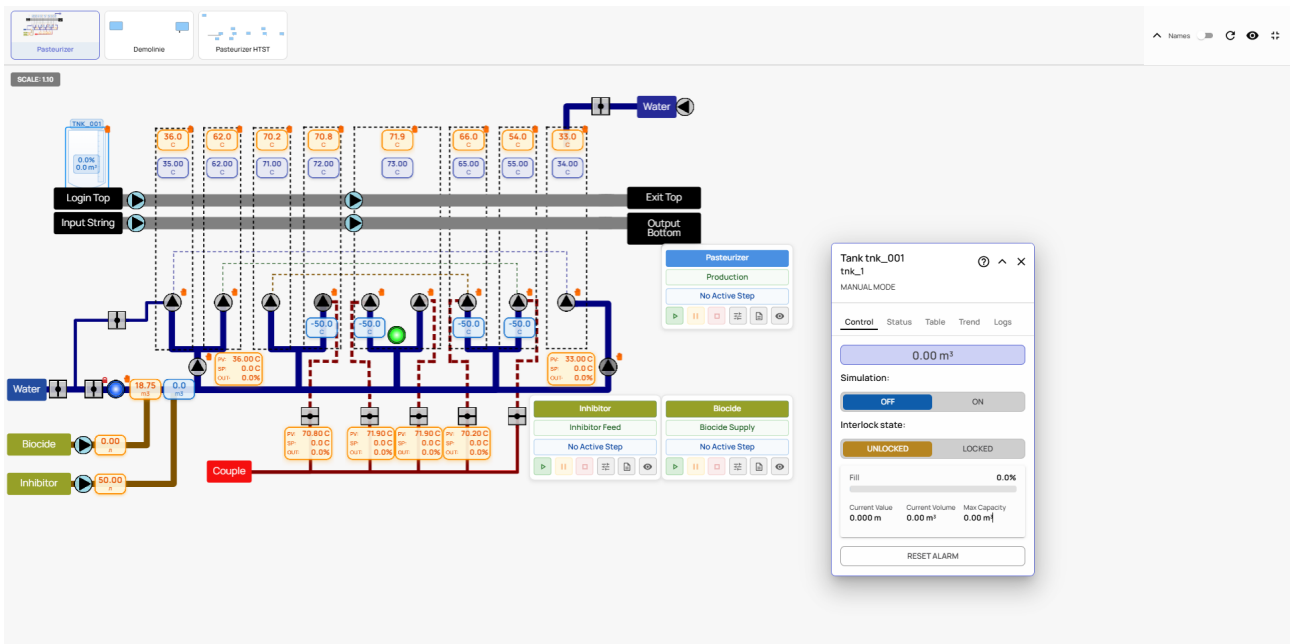
电机弹出窗口：带有启动/停止按钮、自动/手动切换、频率的控制选项卡。绿色指示灯 — 电机正在运行。



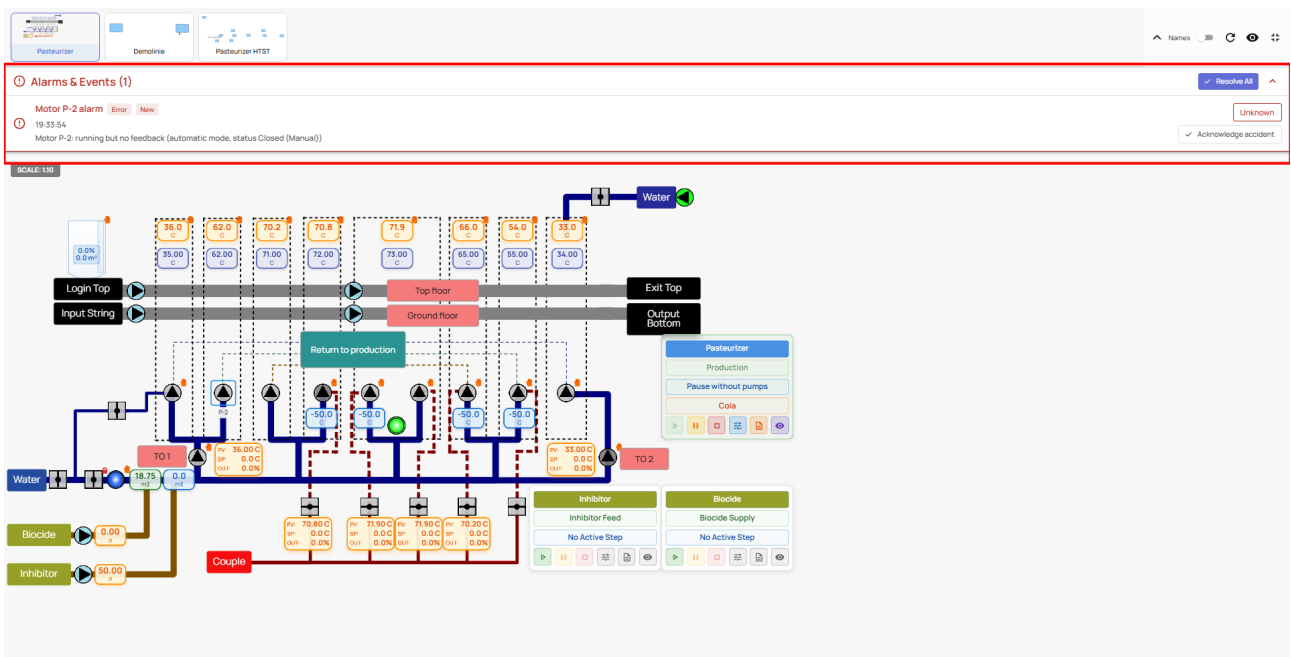
阀门弹出窗口：打开/关闭按钮、打开百分比滑块、自动/手动模式。



PID 弹出窗口：设定点、P、I、D 字段、错误和输出指示器、自动/手动切换。



罐弹出窗口：液位可视化、传感器表（液位、压力）、体积。



顶部报警栏：带有时间戳、严重性和设备名称的红色/黄色消息。

## 技术文档

用于模式查看的 WebSocket 连接：

WebSocket 路径	描述
ws/schema/{schemald}/devices	架构上的所有设备（实时）
ws/schema/{schemald}/pipes	所有管道（流动动画）
ws/schema/{schemald}/sequences	架构上的所有序列
ws/schema/{schemald}/batches	批次和生产线
ws/schema/{schemald}/事故	此架构的事故

## 设备更新事件格式:

```
{
  "type": "schema_devices_status",
  "devices": {
    "mot_001": {
      "device": "mot_001",
      "current_status": {
        "value": 1,
        "is_running": true,
        "mode": "auto",
        "frequency": 50.0,
        "interlock": false
      },
      "status_icon": "/media/icons/motor_running.svg",
      "position_x": 100,
      "position_y": 200,
      "angle": 0,
      "updated_at": "2026-04-07T12:00:00Z"
    }
  }
}
```

## Delta 更新 (仅更改字段) :

```
{
  "type": "device_status_update",
  "device": "mot_001",
  "delta": true,
  "changes": {
    "frequency": 45.0
  }
}
```

## 设备控制API:

方法	端点	描述	身体
POST	/api/{device-type}/{id}/set_mode/	自动/手动	{"模式": "手动"}
POST	/api/{device-type}/{id}/set_manual_value/	设定值	{"值": 50.0}
POST	/api/{device-type}/{id}/set_simulation/	模拟	{"模拟": true}
POST	/api/{device-type}/{id}/set_interlock/	联锁	{"互锁": true}
POST	/api/{device-type}/{id}/update_status/	更新状态	{"current_status": {...}}

## WebSocket 命令 (通过前端 websocketService) :

```
{
  "type": "device_command",
  "device": "mot_001",
  "command": "start",
  "params": {},
  "options": {}
}
```

## 回复:

```
{
  "type": "command_response",
  "device": "mot_001",
  "command": "start",
  "status": "success",
  "data": {...},
  "timestamp": "2026-04-07T12:00:00Z"
}
```

## 6. 设备

### 对于用户

设备页面允许您创建、编辑和删除所有设备类型。

设备类型：

类型	代码	描述
阀门	超低电压	截止/控制阀
电机	交通运输部	电动机
模拟输入	人工智能	模拟传感器 (温度、压力、流量)
模拟输出	AO	模拟输出 (4–20 mA, 0–10 V)
数字输入	DI	开/关信号 (限位开关、就绪信号)
数字输出	做	离散命令
协调信号	COS	系统间协调信号
PID控制器	PID	自动控制回路
专柜	碳纳米管	脉冲计数器
坦克	秋明	具有体积计算功能的容器
定时器	TMR	软件定时器

屏幕元素：

元素	描述
查看切换	网格/表格
«+»按钮	创建新设备
类型过滤器	设备类型下拉列表
搜索	按名称搜索
刷新	重新加载列表

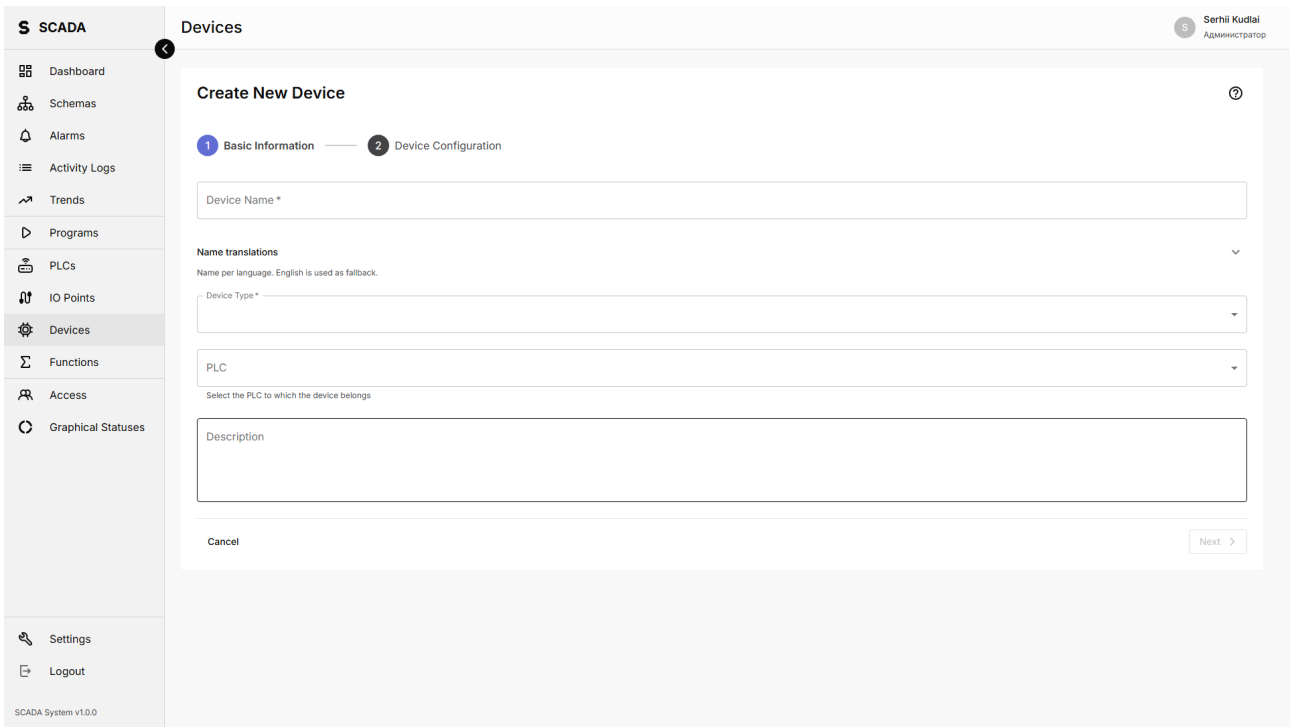
设备形式 (创建/编辑)：

领域	描述
设备 ID	唯一标识符 (例如 “mot_001” )
名称	人类可读的名称
名字翻译	多语言名称
类型	设备类型选择器
描述	文字说明
默认值	初始值
工程单位	欧盟标签
最小/最大	允许范围
缩放	信号转换参数
“保存” 按钮	保存设备
“取消” 按钮	不保存返回

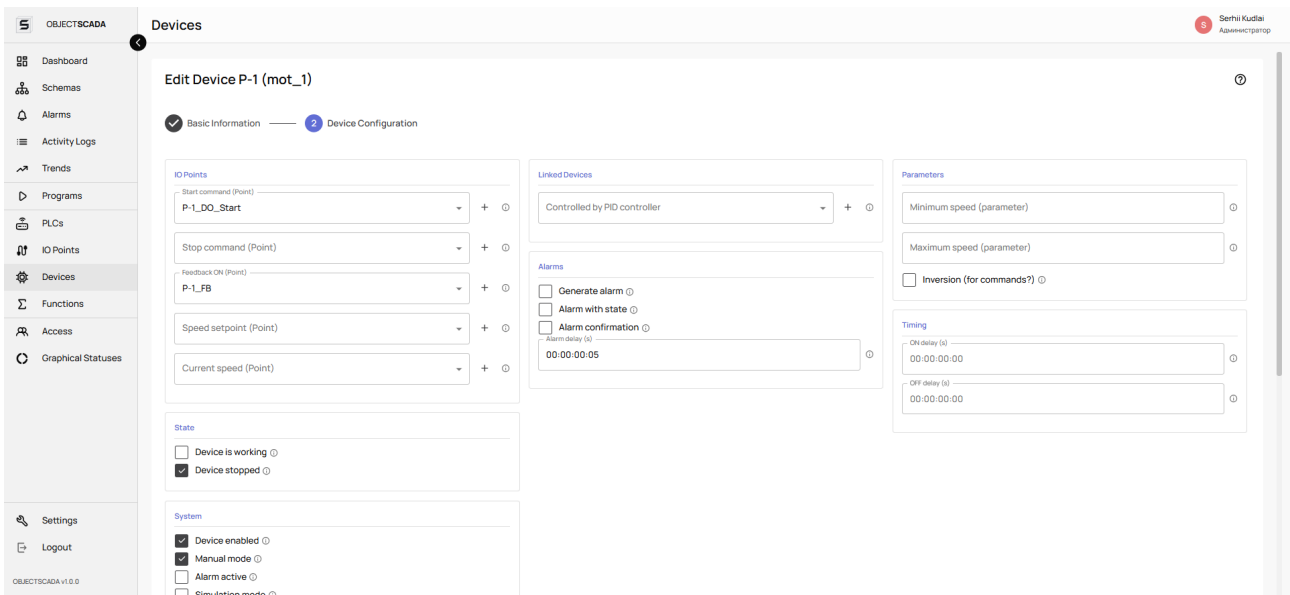
领域	描述
“删除” 按钮	删除设备 (仅限编辑模式)

网格形式的设备页面：带有名称、类型和值的设备类型图标卡（电机、阀门等）。在顶部键入过滤器。

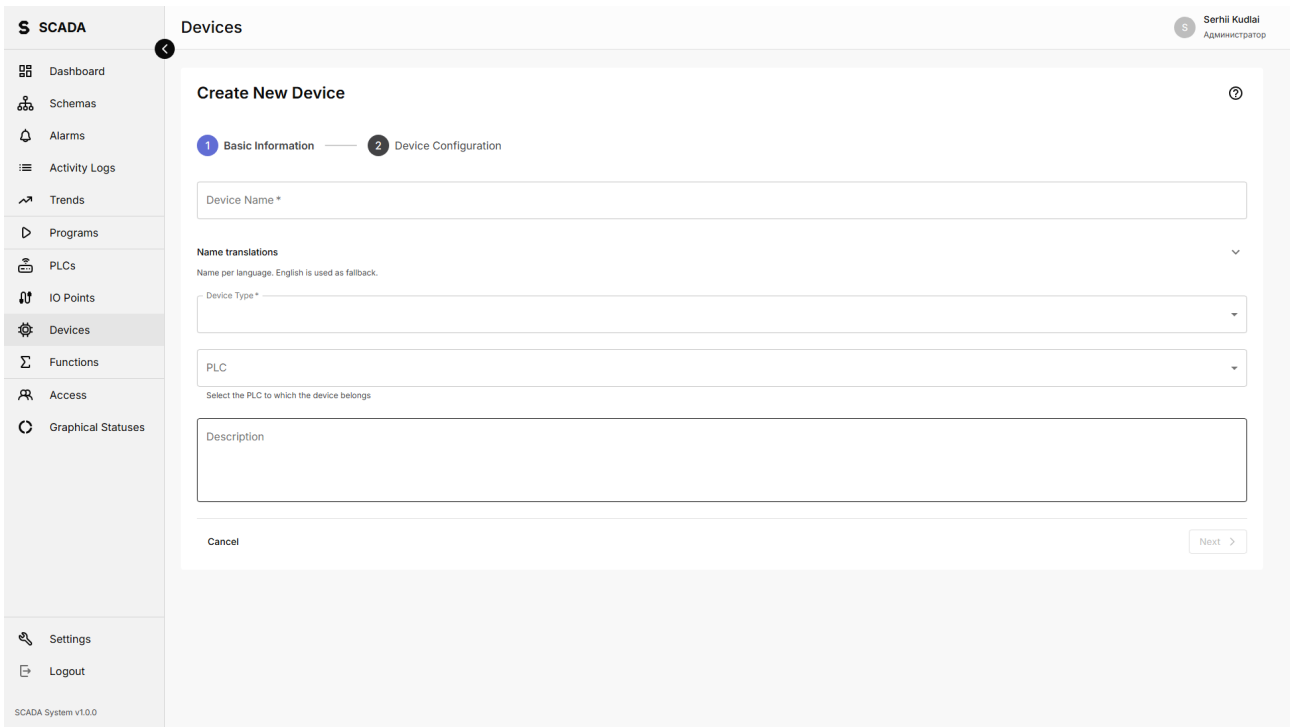
表视图：ID、名称、类型、默认值、单位、操作列。



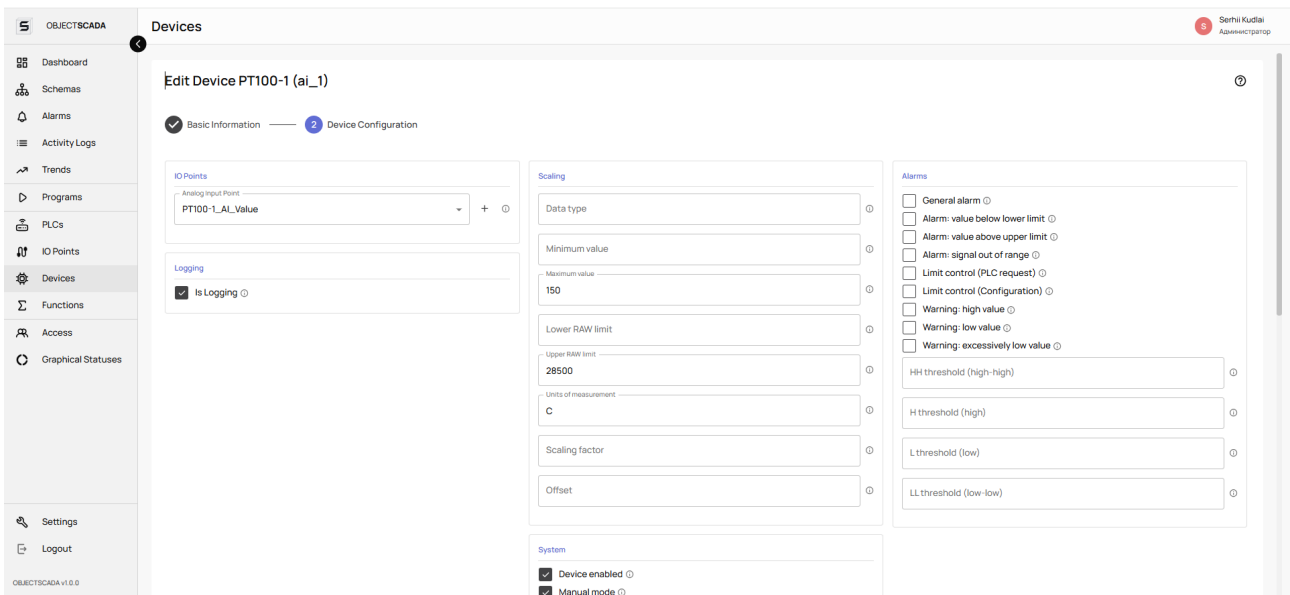
电机创建表单：主要字段 — ID、名称、PLC、描述、设备类型。



电机形式：最小/最大速度、延迟、生成警报、保存按钮。



模拟输入形式：最小值/最大值、单位、缩放比例、原始最小值/最大值。



AI表单：警告阈值、默认值和其余字段。

## 技术文档

设备API (所有类型通用) :

方法	端点	描述
GET	/api/devices/all_devices/	所有类型的所有设备
GET	/api/{device-type}/	某种类型的设备列表
POST	/api/{device-type}/	创建设备
GET	/api/{device-type}/{id}/	检索设备
PUT	/api/{device-type}/{id}/	更新

方法	端点	描述
PATCH	/api/{device-type}/{id}/	部分更新
DELETE	/api/{device-type}/{id}/	删除

端点类型：

- /api/analog-inputs/
- /api/analog-outputs/
- /api/digital-inputs/
- /api/digital-outputs/
- /api/motors/
- /api/valves/
- /api/pid-controllers/
- /api/counters/
- /api/timers/
- /api/coordination-signals/
- /api/variable-speed-drives/

附加操作（每种类型）：

方法	端点	描述
POST	/{id}/set_mode/	切换自动/手动
POST	/{id}/set_simulation/	启用/禁用模拟
POST	/{id}/set_interlock/	设置/清除互锁
POST	/{id}/update_status/	更新当前状态
POST	/{id}/set_manual_value/	设置手动值
POST	/{id}/trigger_websocket_update/	强制 WS 更新 (AllowAnonymous)

WebSocket：设备更新

连接：ws://ws/device/{device\_id}

```
{
  "type": "device_status_update",
  "device": "ai_001",
  "data": {
    "current_status": {
      "value": 23.5,
      "mode": "auto",
      "simulation": false,
      "interlock": false,
      "quality": "good"
    },
    "status_icon": null,
    "updated_at": "2026-04-07T12:00:00Z"
  }
}
```

## 7. PLC

### 对于用户

PLC管理页面。

屏幕元素：

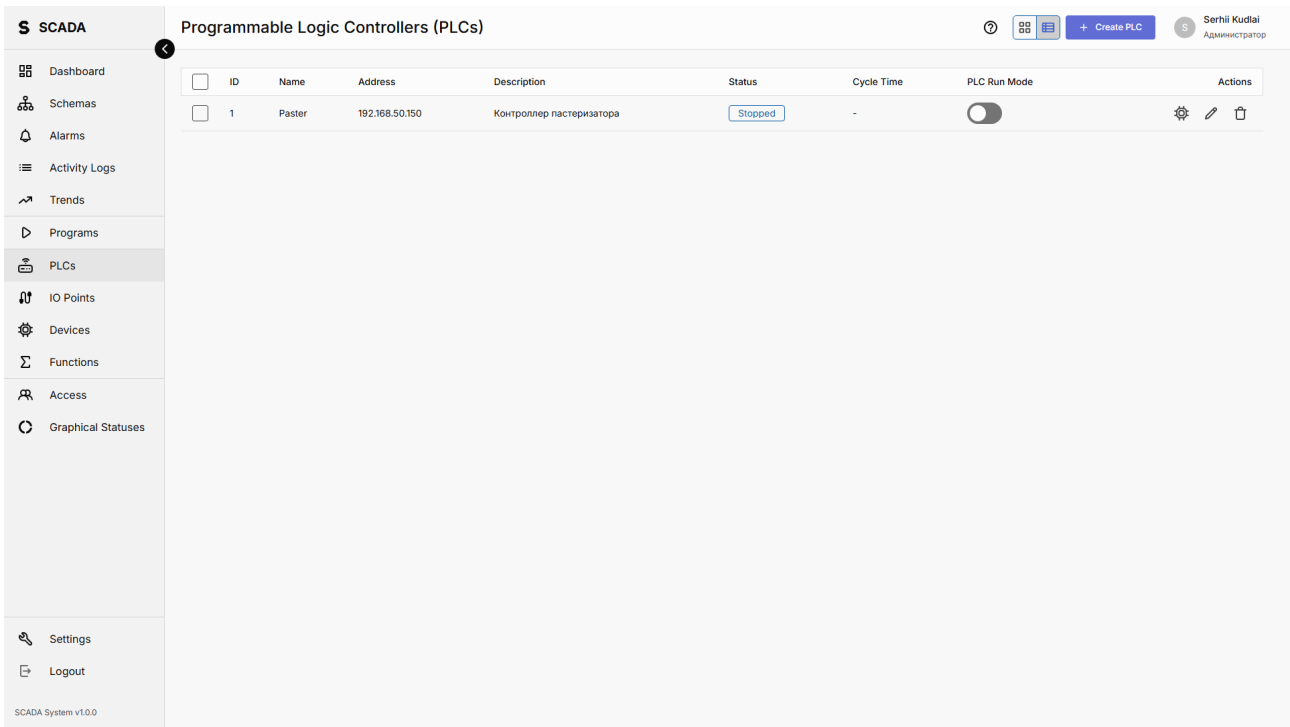
元素	描述
PLC列表	PLC 卡/表
状态指示灯	绿色 — 在线, 灰色 — 离线, 红色 — 错误
«+»按钮	添加新的 PLC
“硬件配置”按钮	转到硬件配置器
运行模式切换	启用/禁用 PLC 运行模式

PLC 卡上的字段：

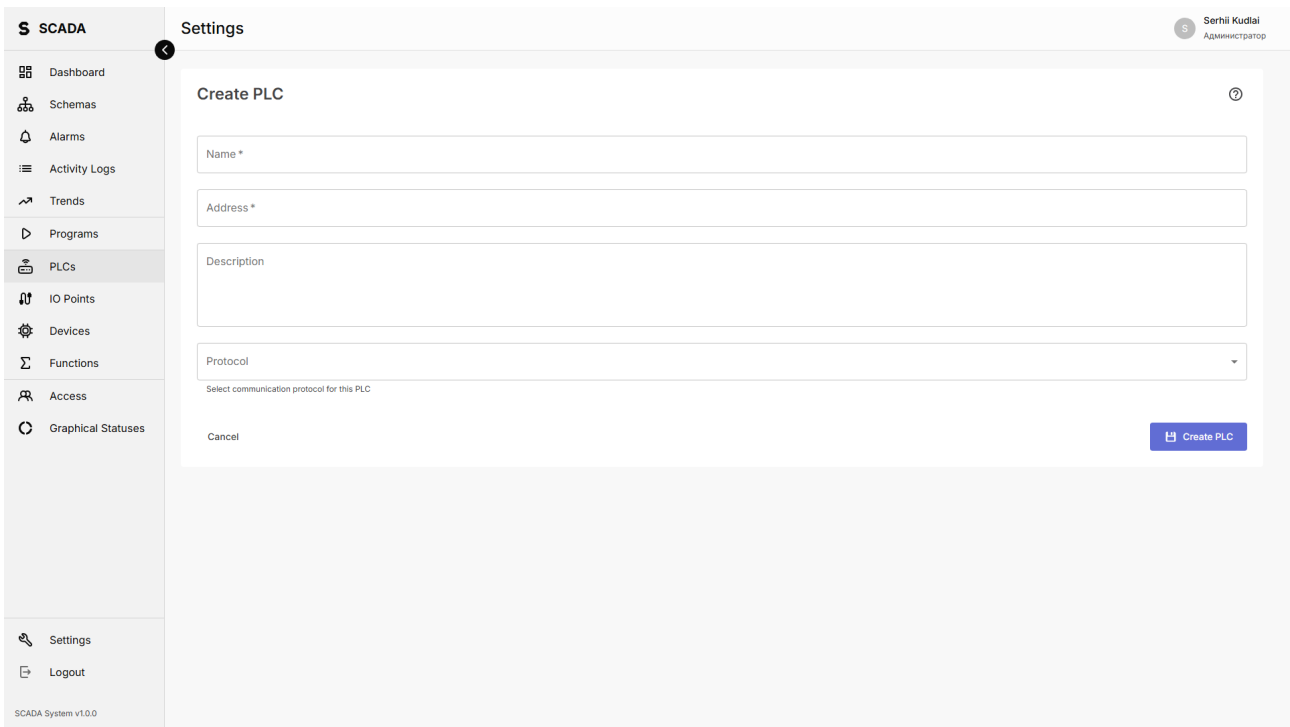
领域	描述
名称	PLC名称
协议	通讯类型 (西门子S7、Modbus、OPC UA)
状态	线上/线下
周期时间	投票率
IP 地址	网络地址
设备数量	连接设备数量

PLC形式：

领域	描述
名称	PLC名称
描述	文字说明
协议	协议选择器
IP / 端口	网络设置
机架/插槽	对于西门子S7
«测试连接»按钮	测试与 PLC 的连接
“保存”按钮	保存配置



PLC 列表：带有状态指示灯（绿色/灰色）、名称、协议、IP、运行模式切换、硬件配置按钮的卡。



PLC 配置表单：名称字段、协议下拉列表、IP 地址、端口、机架、插槽、测试连接按钮。

## 技术文档

### PLC API:

方法	端点	描述
GET	/api/plcs/	列出 PLC
POST	/api/plcs/	创建 PLC

方法	端点	描述
GET	/api/plcs/{id}/	检索 PLC
PUT	/api/plcs/{id}/	更新
DELETE	/api/plcs/{id}/	删除
GET	/api/plcs/{id}/status/	PLC状态
POST	/api/plcs/{id}/set-run-mode/	切换运行模式
GET	/api/plcs/status/	所有 PLC 的状态

## PLC模拟:

方法	端点	描述
POST	/api/plc/simulation/run/	运行模拟
GET	/api/plc/simulation/status/	模拟状态
POST	/api/plc/service/start/	启动PLC服务
POST	/api/plc/service/stop/	停止PLC服务

## 8. 序列

### 对于用户

序列定义了生产设备的操作顺序——步骤、步骤之间的转换以及切换条件。

屏幕元素：

元素	描述
序列表	带过滤功能的卡片/表格
«+»按钮	创建一个新序列
状态指示灯	绿色 — 运行, 灰色 — 停止, 黄色 — 暂停
状态过滤器	全部/正在运行/已停止
搜索	按名称搜索

序列编辑器 (选项卡) :

选项卡	描述
属性	常用设置 (名称、描述、设备绑定)
步骤	执行顺序的步骤列表
过渡条件	步骤之间的转换配置
活动	事件处理
事故	事故处理
选项	选项配置
配方变量	变量定义
制作配方	此序列的食谱
脉冲模块	脉冲模块控制
批次	批次管理

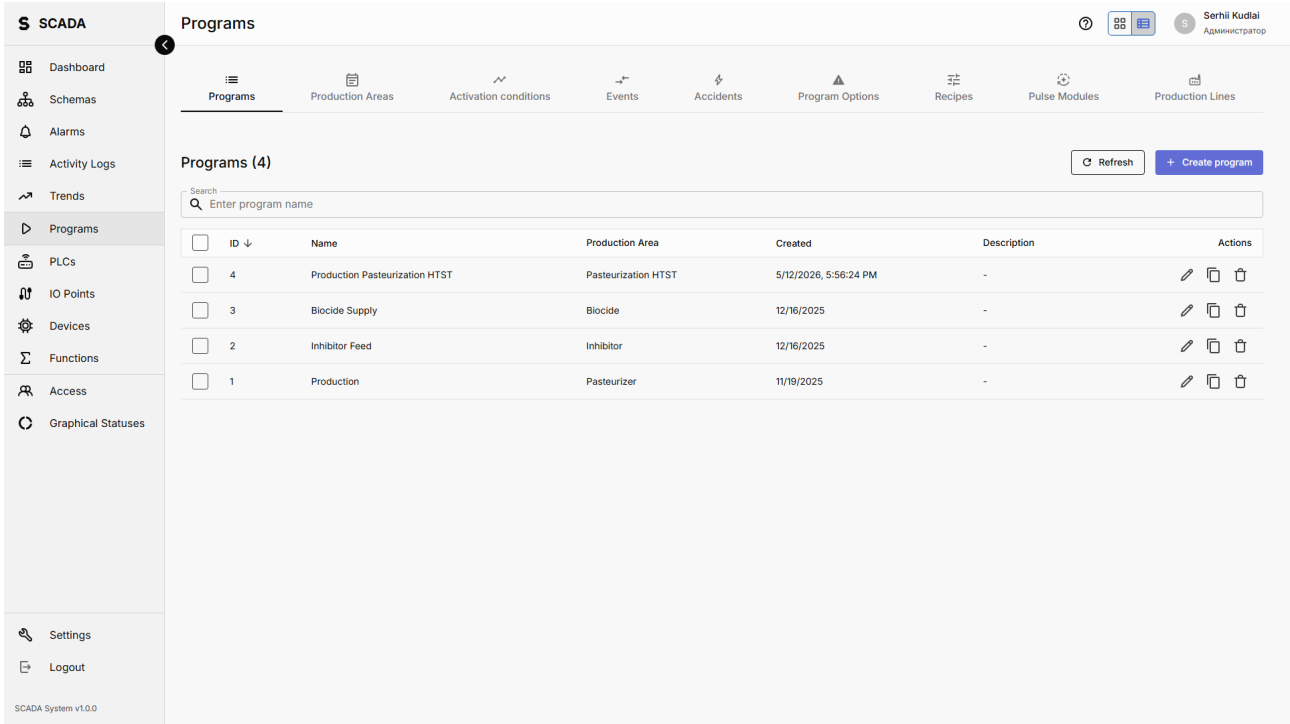
步骤配置：

领域	描述
订单	序列中的步骤号
名称	步骤名称 (+ 翻译)
描述	文字说明
设备	这一步绑定的设备及其值
下一步	转换的 JSON 映射: {"target_step_order": [condition_id, ...]}
批量操作	无/转移/完成

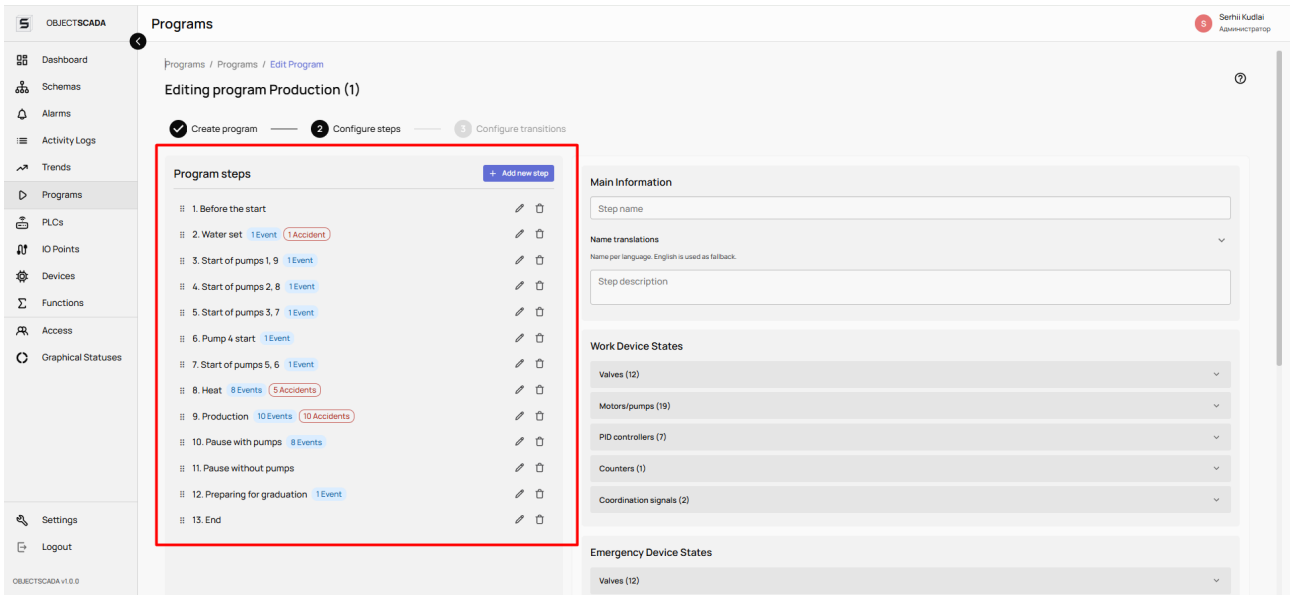
转换条件：

领域	描述
类型	设备 (按值)、计时器 (按时间)、选项 (按操作员操作)
设备	受监控设备 (针对设备类型)
操作员	==、!=、>、=、<=

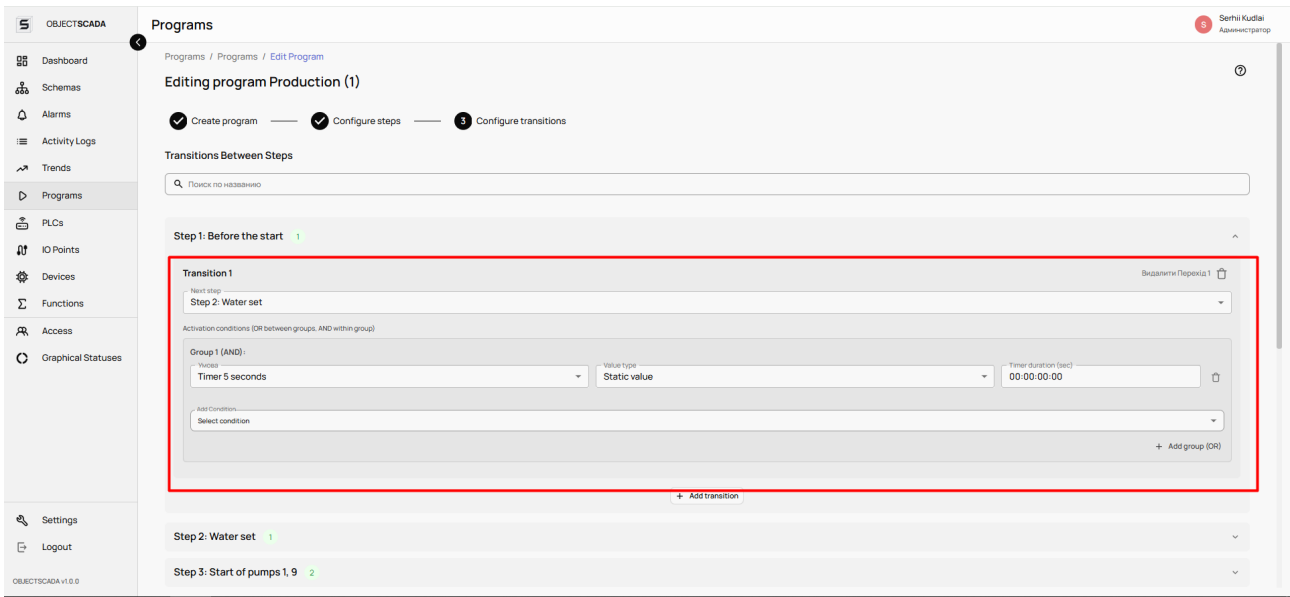
领域	描述
价值	比较值
持续时间	时间 (定时器)
模拟	启用/禁用条件模拟 (用于测试)



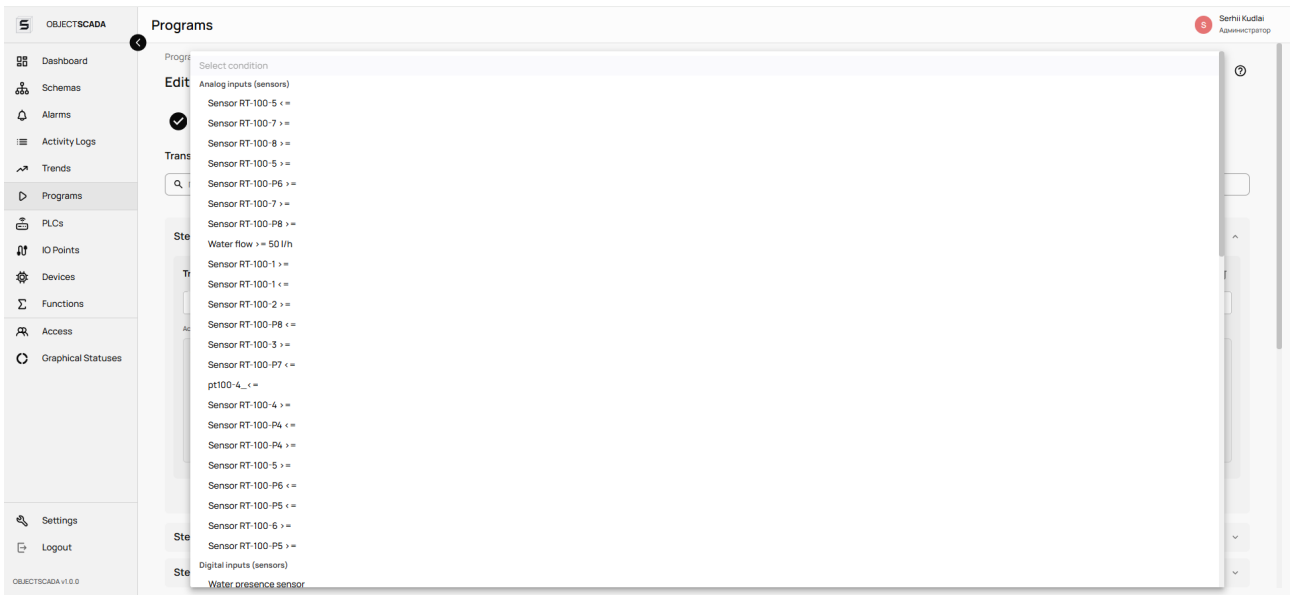
序列列表：带有名称的卡片、状态指示器（绿色/灰色）、当前步骤、编辑/删除按钮。



步骤编辑器：步骤编号列表，每个步骤都有名称、绑定设备、转换箭头、添加/删除步骤按钮。



转换条件编辑器：包含类型、设备、操作员、值、持续时间、模拟切换列表的表。



创建转换条件对话框：类型选择器（设备/定时器/选项）、设备下拉列表、比较运算符、值字段。

## 技术文档

### 序列API:

方法	端点	描述
GET	/api/sequences/	列出序列
POST	/api/sequences/	创建
GET	/api/sequences/{id}/	检索
PUT	/api/sequences/{id}/	更新
DELETE	/api/sequences/{id}/	删除
GET	/api/sequences/{id}/steps/	列出步骤
POST	/api/sequences/{id}/add_step/	添加一个步骤

方法	端点	描述
POST	/api/sequences/{id}/set_steps_order/	重新排序步骤
POST	/api/sequences/{id}/add_transition_conditions/	添加条件

## 步骤API:

方法	端点	描述
GET	/api/sequence-steps/	所有步骤
POST	/api/sequence-steps/	创建步骤
PUT	/api/sequence-steps/{id}/	更新
DELETE	/api/sequence-steps/{id}/	删除

## 转换条件API:

方法	端点	描述
GET	/api/transition-conditions/	所有条件
POST	/api/transition-conditions/	创建
PUT	/api/transition-conditions/{id}/	更新
DELETE	/api/transition-conditions/{id}/	删除
POST	/api/transition-conditions/{id}/simulate/	模拟条件
POST	/api/transition-conditions/clear_simulation/	清晰的步骤模拟

## WebSocket: 序列状态

连接: ws://ws/sequence/{sequence\_id}

```
{
  "type": "sequence_status",
  "id": 1,
  "name": "Production A",
  "is_running": true,
  "is_pause": false,
  "is_stop": false,
  "active_step": 3,
  "active_step_data": {
    "id": 3,
    "order": 3,
    "name": "Heating",
    "name_translations": {"ru": "Нагрев"},
    "description": "..."
  },
  "previous_step": 2,
  "current_step_start_time": "2026-04-07T11:30:00Z",
  "monitored_devices": {
    "ai_001": {"value": 72.3, "mode": "auto"}
  },
  "activated_devices": {
    "mot_001": {"value": 1, "is_running": true}
  },
  "conditions": [
    {
      "id": 10,
      "type": "device",
      "device_id": "ai_001",

```

```
"operator": ">=",  
"value": 80.0,  
... (+9 строк)
```

## 9. 配方和变量

### 对于用户

配方定义序列每个步骤的特定设备参数值。配方变量让您可以对这些值进行参数化。

食谱类型：

类型	描述
食谱 (步骤食谱)	为序列的每个步骤定义设备值
制作配方	特定生产作业的一组变量

步骤食谱形式：

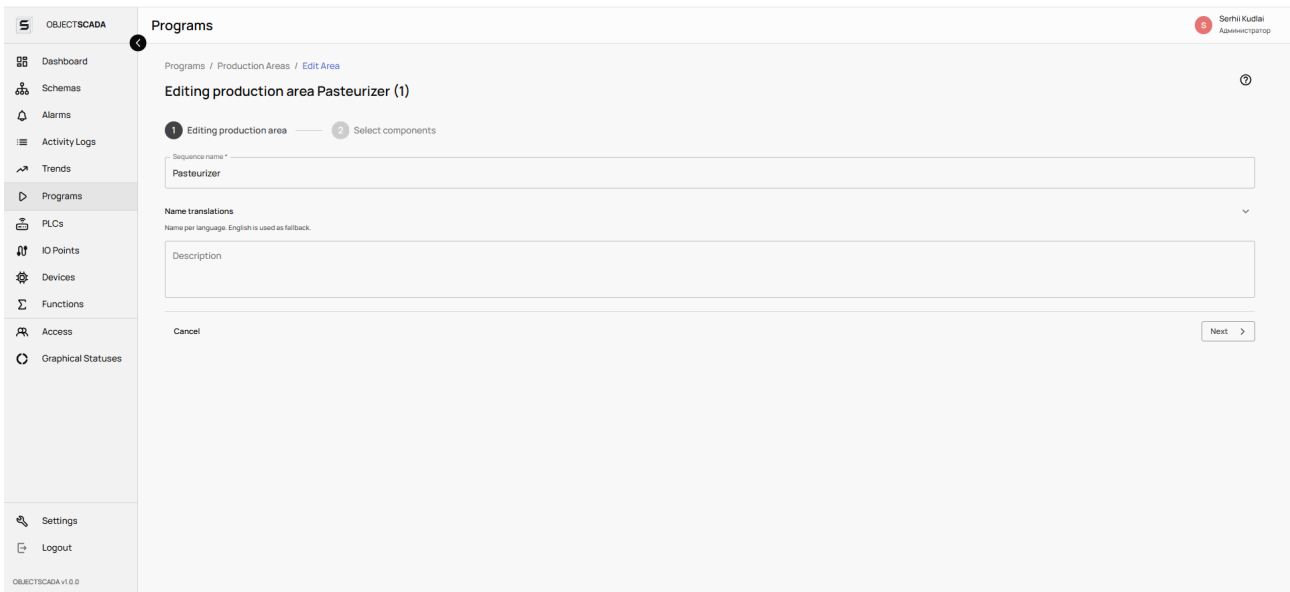
领域	描述
名称	食谱名称 (+翻译)
序列	绑定到序列
步骤	每一步 — 一组“设备 → 值”条目

生产配方形式：

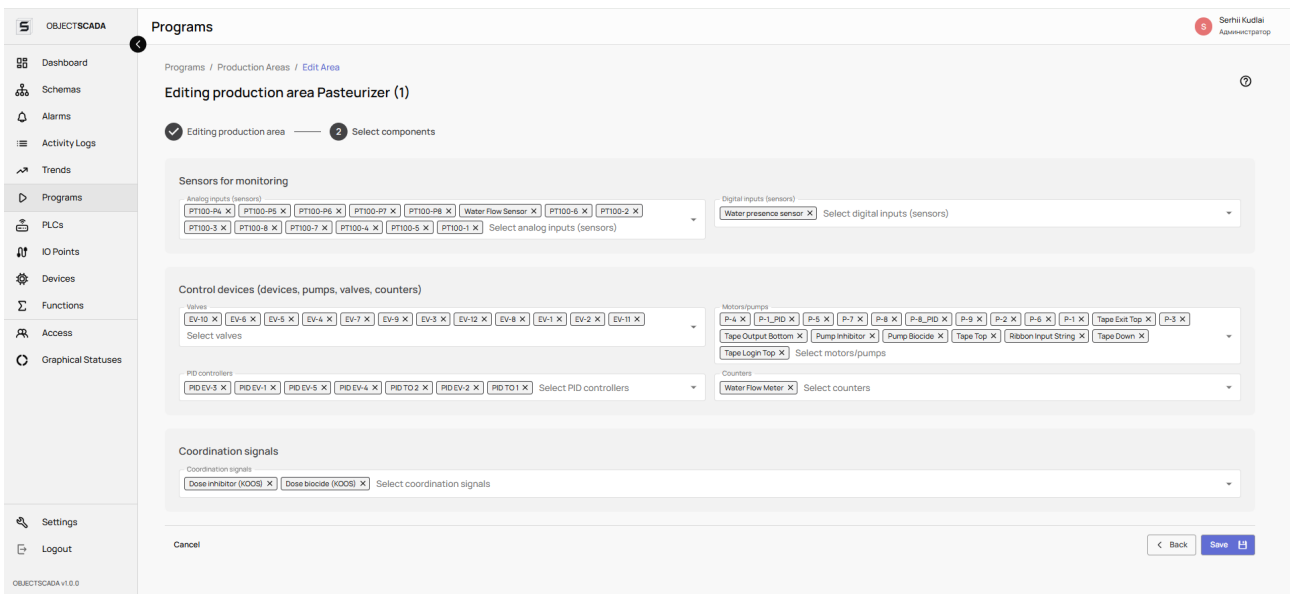
领域	描述
名称	食谱名称
绑定	到序列或到生产线 (XOR)
变量	变量及其值列表

配方变量：

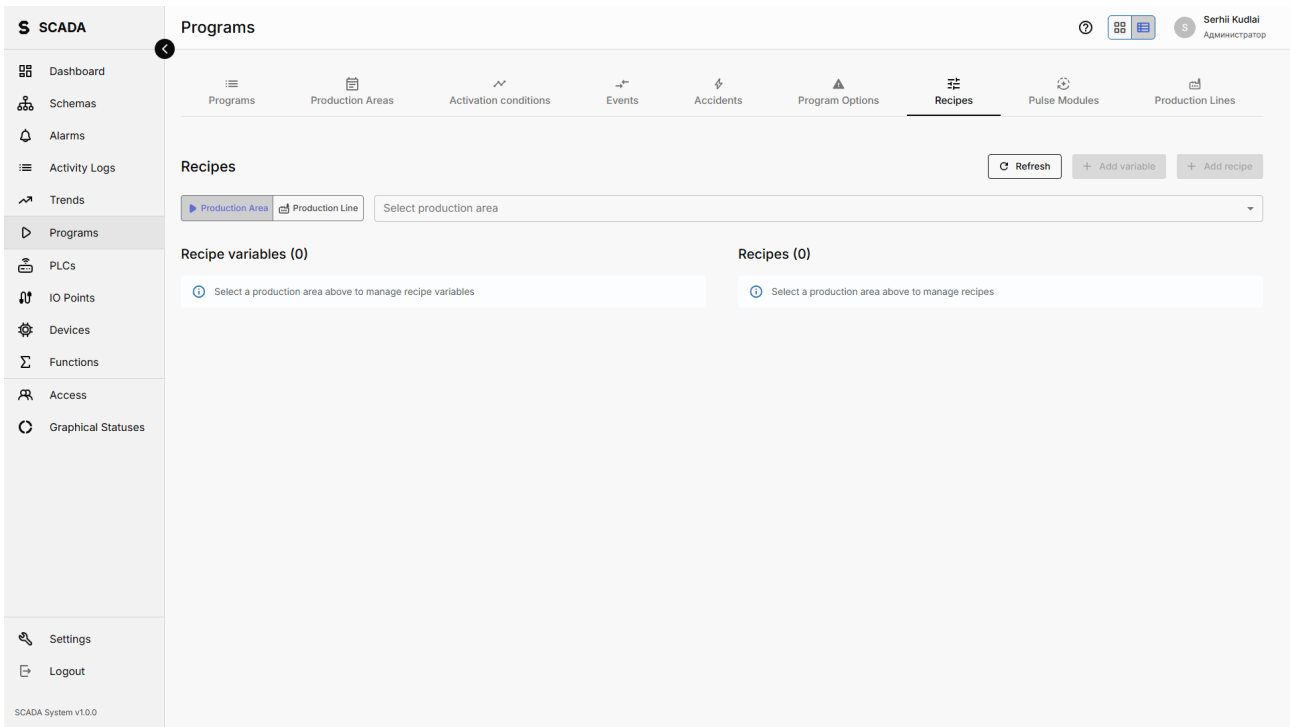
领域	描述
名称	变量名称 (+翻译)
绑定	到序列或到生产线 (XOR)
类型	数字、文本
默认值	初始值
最小/最大	限制



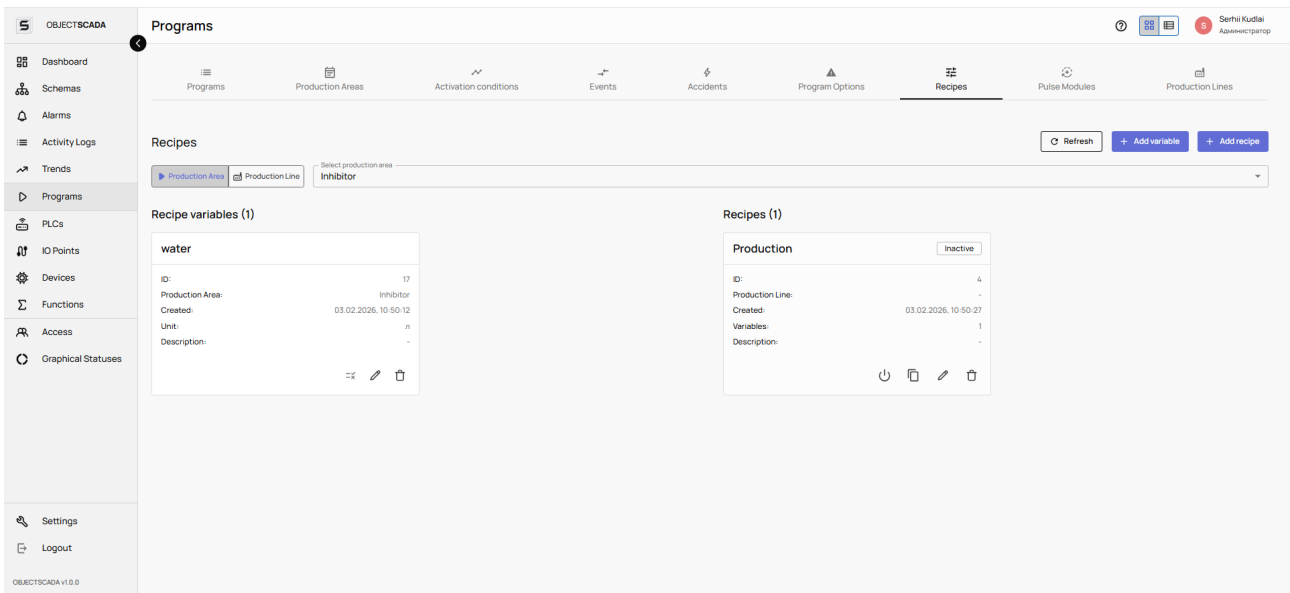
配方创建向导 — 第 1 步：名称和序列绑定。



配方创建向导 — 第 2 步：步骤 → 设备 → 值表。



生产配方管理：带有创建/编辑/删除/复制按钮、序列/生产线切换的配方表。



变量管理：变量列表，包括名称、类型、默认值、最小值/最大值。添加/删除按钮。

## 技术文档

### 食谱API:

方法	端点	描述
GET	/api/recipes/	步骤食谱列表
POST	/api/recipes/	创建食谱
PUT	/api/recipes/{id}/	更新
DELETE	/api/recipes/{id}/	删除

### 生产配方API:

方法	端点	描述	参数
GET	/api/production-recipes/	所有食	?sequence=ID 或 ?生产线=ID
POST	/api/production-recipes/	创建	—
PUT	/api/production-recipes/{id}/	更新	—
DELET	/api/production-recipes/{id}/	删除	—
POST	/api/production-recipes/{id}/select-production-recipe/	激活	—

## 变量API:

方法	端点	描述	参数
GET	/api/production-recipe-variables/	所有变	?sequence=ID 或 ?生产线=ID
POST	/api/production-recipe-variables/	创建	—
PUT	/api/production-recipe-variables/{id}/	更新	—
DELETE	/api/production-recipe-variables/{id}/	删除	—

# 10. 事件和事故

## 对于用户

事件 — 在序列操作期间运行操作的条件（例如，达到某个值时进行切换）。

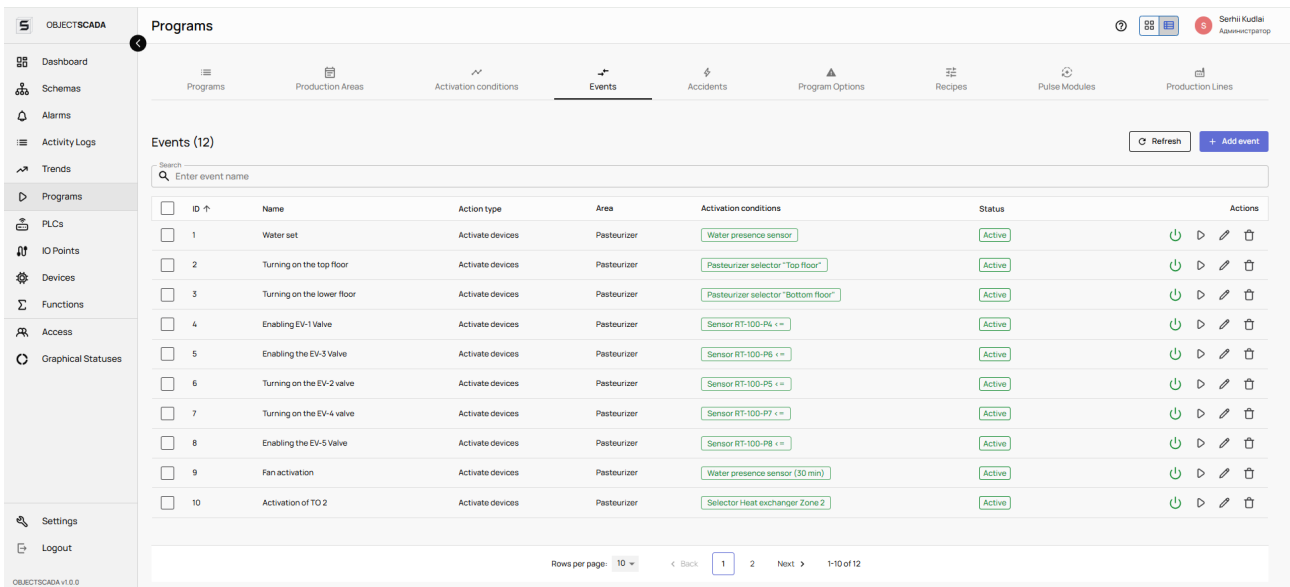
事故——当危险情况出现时，导致过程停止的关键情况。

活动形式：

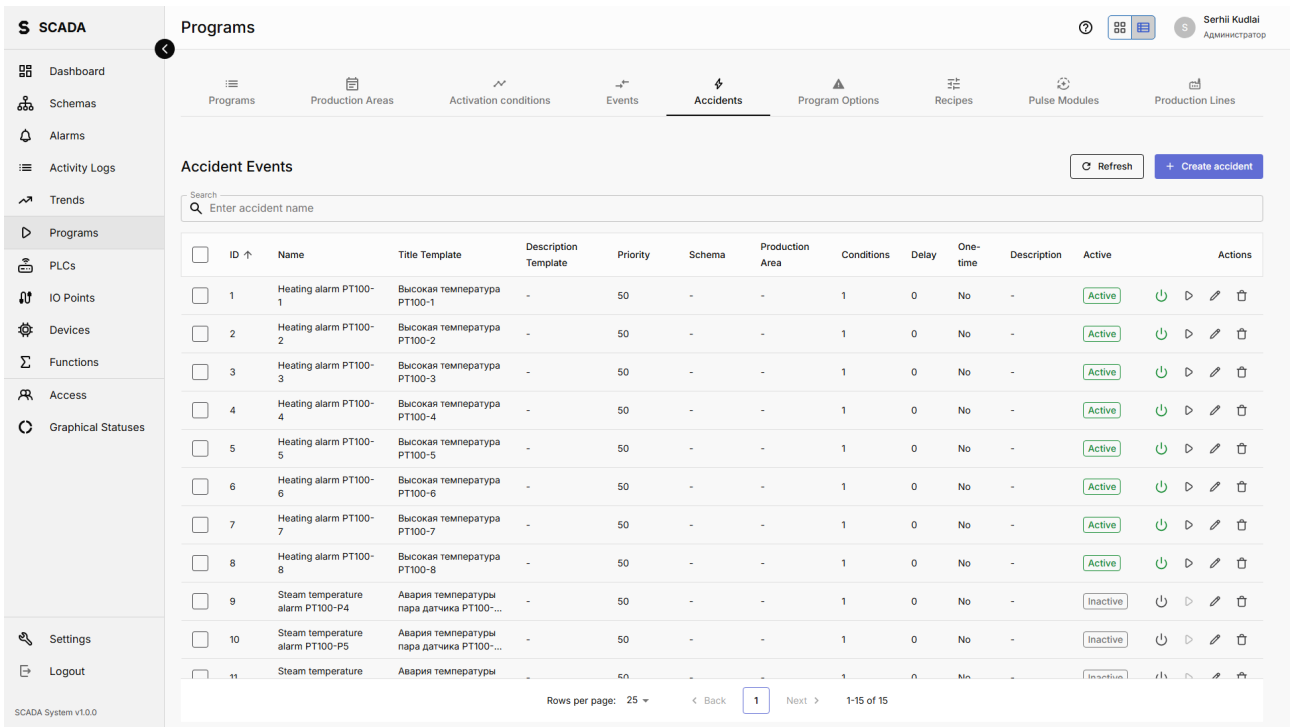
领域	描述
名称	活动名称 (+ 翻译)
类型	触发类型
设备	受监控设备
状况	比较运算符和值
行动	触发器执行什么

事故表格：

领域	描述
名称	事故名称 (+ 翻译)
严重性	信息/警告/错误/严重
设备	受监控设备
状况	运算符和值
联锁	锁定设备



事件管理：事件表，包含名称、设备、条件、操作列、添加/编辑/删除按钮。



事故管理：带有严重性颜色编码的表格（信息蓝色、警告黄色、错误红色、严重深红色）。

## 技术文档

### 事件API:

方法	端点	描述
GET	/api/events/	活动列表
POST	/api/events/	创建
PUT	/api/events/{id}/	更新
DELETE	/api/events/{id}/	删除
POST	/api/events/{id}/set_active/	激活/停用

### 事故API:

方法	端点	描述
GET	/api/accident-events/	列出事故定义
POST	/api/accident-events/	创建定义
GET	/api/accidents/	火灾事故列表
POST	/api/accidents/{id}/acknowledge/	确认
POST	/api/accidents/{id}/resolve/	解决
POST	/api/accidents/resolve-all/	全部解决
GET	/api/accidents/active_events/	目前发生的事故

### WebSocket: 意外

连接: ws:///ws/accidents

```
{
  "type": "accident_created",
  "accident_id": 42,
  "title": "High temperature",
  "description": "AI_001 exceeded the 95°C threshold",
  "status": "NEW",
  "severity_level": "CRITICAL",
  "device_id": "ai_001",
  "device_name": "Temperature Sensor 1",
  "schema_id": 1,
  "occurred_at": "2026-04-07T12:00:00Z"
}
```

# 11. 选项

## 对于用户

选项是操作员在序列执行期间可用的按钮/开关。它们让操作员选择替代执行路径。

选项形式：

领域	描述
名称	选项名称 (+ 翻译)
序列	序列结合
描述	文字说明

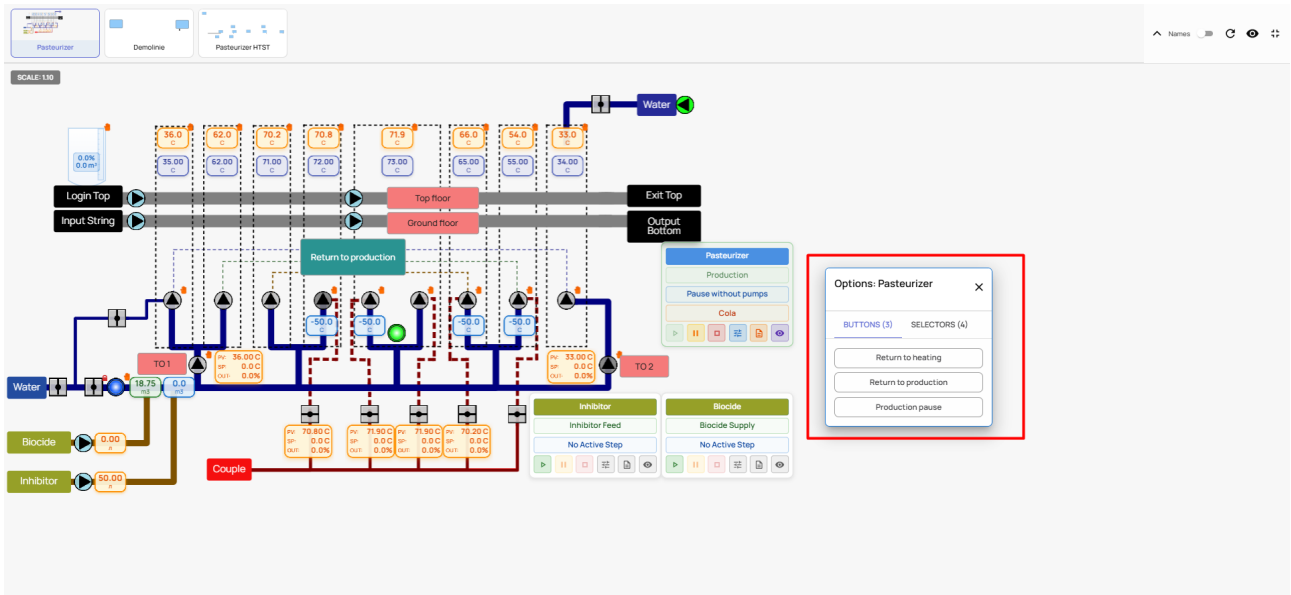
关于架构（架构查看器）：

- 选项显示为按钮
- 按下可切换选项（开/关）
- 按钮颜色随状态变化

The screenshot shows the 'Program Options' management interface. It features a sidebar on the left with navigation options: Dashboard, Schemas, Alarms, Activity Logs, Trends, Programs, PLCs, IO Points, Devices, Functions, Access, Graphical Statures, Settings, and Logout. The main content area is titled 'Programs' and includes a sub-section for 'Program Options'. The 'Program Options' section has a search bar and a table with 7 entries. Each entry includes a checkbox, ID, Name, Description, Type (Button (trigger) or Selector), Area (Pasteurizer), Activated date, Conditions, Availability (Available for interaction), Status (Inactive), and Actions (power, refresh, edit, delete).

ID	Name	Description	Type	Area	Activated	Conditions	Availability	Status	Actions
1	Production pause	-	Button (trigger)	Pasteurizer	06.05.2026, 20:58:42	1	Available for interaction	Inactive	Power, Refresh, Edit, Delete
2	Top floor	-	Selector	Pasteurizer	-	1	Available for interaction	Inactive	Power, Refresh, Edit, Delete
3	Ground floor	-	Selector	Pasteurizer	-	1	Available for interaction	Inactive	Power, Refresh, Edit, Delete
4	Return to production	-	Button (trigger)	Pasteurizer	08.05.2026, 20:00:32	1	Available for interaction	Inactive	Power, Refresh, Edit, Delete
5	Return to heating	-	Button (trigger)	Pasteurizer	24.03.2026, 15:44:42	1	Available for interaction	Inactive	Power, Refresh, Edit, Delete
6	TO 1	-	Selector	Pasteurizer	-	1	Available for interaction	Inactive	Power, Refresh, Edit, Delete
7	TO 2	-	Selector	Pasteurizer	-	1	Available for interaction	Inactive	Power, Refresh, Edit, Delete

选项管理：包含名称、顺序、描述、编辑/删除按钮的选项列表。



模式上的选项按钮：带有名称的矩形按钮，绿色 - 活动，灰色 - 不活动。

## 技术文档

### 选项API:

方法	端点	描述
GET	/api/sequence-options/	选项列表
POST	/api/sequence-options/	创建
PUT	/api/sequence-options/{id}/	更新
DELETE	/api/sequence-options/{id}/	删除

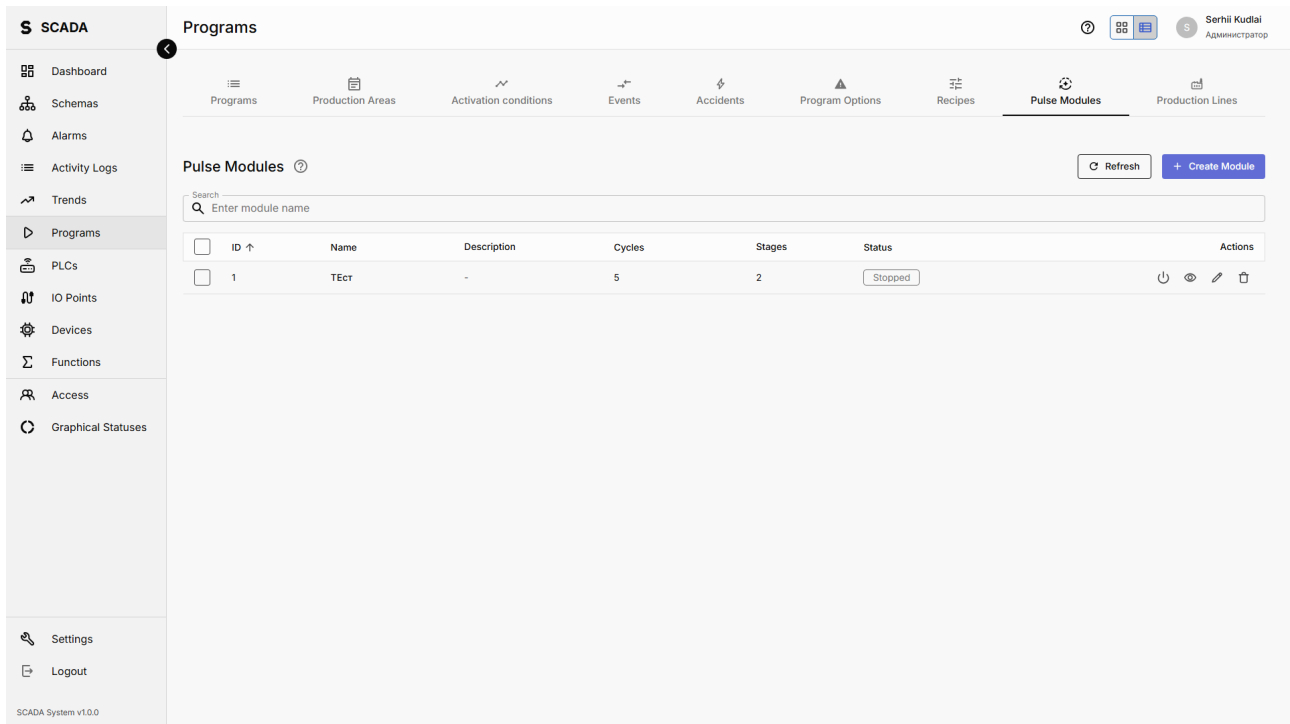
## 12. 脉冲模块

### 对于用户

脉冲模块生成循环开/关设备序列（例如，循环供水）。

结构：

水平	描述
模块	台阶容器
步骤	一个周期的一个阶段
相	步骤内的时间间隔（开/关）
相器	哪个设备已打开/关闭



脉冲模块管理：分层列表——模块→步骤→阶段→设备。每个级别上的添加/编辑/删除按钮。

### 技术文档

脉冲模块API：

方法	端点	描述
GET	/api/pulse-modules/	列出模块
POST	/api/pulse-modules/	创建
POST	/api/pulse-modules/{id}/start/	开始
POST	/api/pulse-modules/{id}/stop/	停止
GET	/api/pulse-module-steps/	模块步骤
GET	/api/pulse-module-step-phases/	步骤阶段

方法	端点	描述
GET	/api/pulse-module-devices/	相装置

## 13. 批次管理

### 对于用户

批次管理系统全程跟踪生产批次。

批处理面板元素：

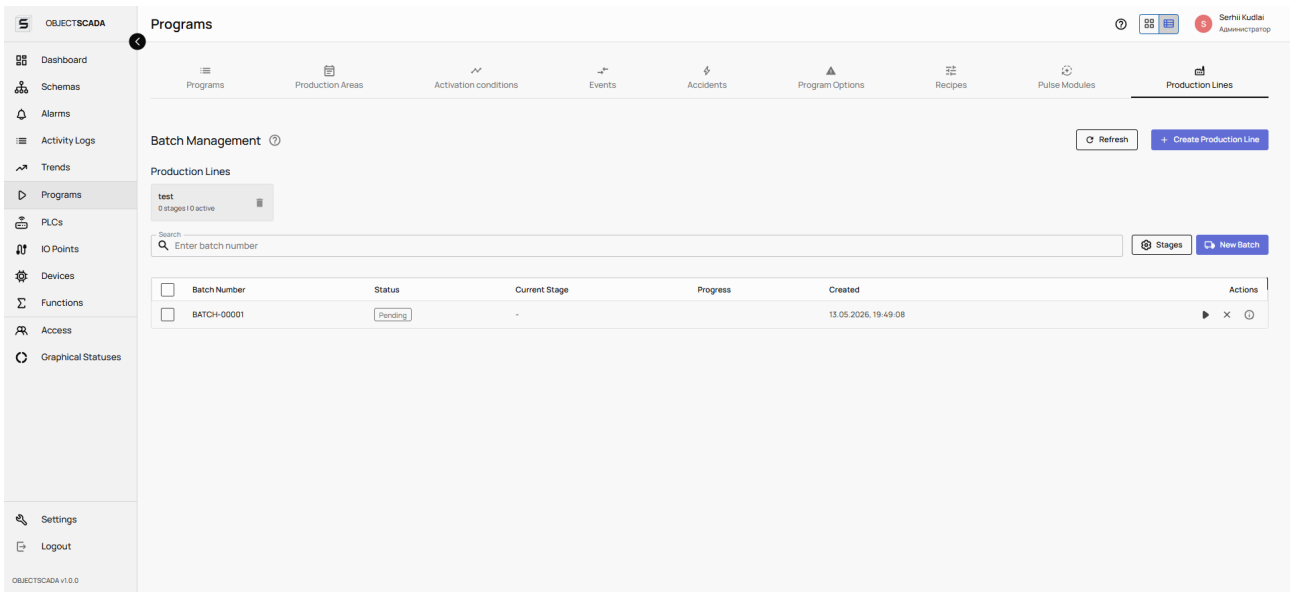
元素	描述
创建批次	创建新批次的按钮
批次表	包含编号、配方、状态、进度的列表
批次状态	待处理/排队/进行中/暂停/已完成/失败/取消
队列	加工订单

批量操作：

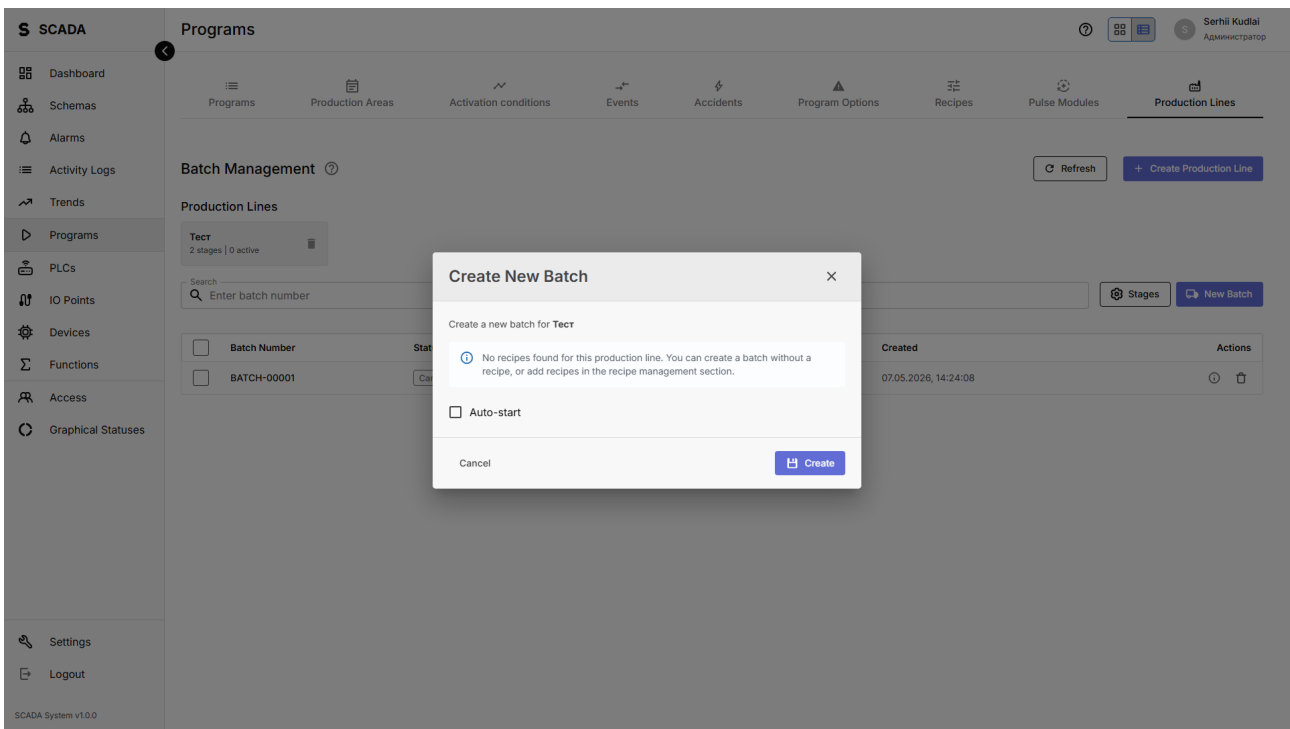
按钮	描述
玩	启动/恢复批次
暂停	暂停
停止	句号
跳过	跳过当前步骤
取消	取消批次
删除	从列表中删除

创建批处理对话框：

领域	描述
生产线	线路选择器
食谱	配方选择器
批号	唯一编号 (BAT-001)
自动启动	创建后立即开始



批次管理面板：带有状态颜色编码的批次表（绿色 - 进行中、灰色 - 待定、蓝色 - 已完成、红色 - 失败）、控制按钮。



创建对话框：生产线和配方选择器、数字字段、“自动启动”复选框、“创建”和“取消”按钮。

## 技术文档

### 批次API:

方法	端点	描述
GET	/api/batches/	所有批次
POST	/api/production-lines/{id}/create_batch/	创建批次
POST	/api/batches/{id}/start/	开始
POST	/api/batches/{id}/pause/	暂停
POST	/api/batches/{id}/resume/	简历

方法	端点	描述
POST	/api/batches/{id}/cancel/	取消
POST	/api/batches/{id}/transfer/	转入下一阶段
GET	/api/batches/{id}/history/	舞台历史
GET	/api/batch-queue/	批量队列
POST	/api/batch-queue/reorder/	重新排序队列

### WebSocket: 批次状态

连接: ws:///ws/schema/{schemald}/batches

```
{
  "type": "batch_status",
  "batch_id": "uuid-...",
  "batch_number": "BAT-001",
  "status": "IN_PROGRESS",
  "current_stage": "Mixing",
  "progress": 65,
  "production_line_id": "uuid-...",
  "recipe_name": "Recipe A",
  "updated_at": "2026-04-07T12:00:00Z"
}
```

### WebSocket 命令 (客户端 → 服务器) :

```
{
  "type": "batch_command",
  "command": "start",
  "batch_id": "uuid-...",
  "production_line_id": "uuid-..."
}
```

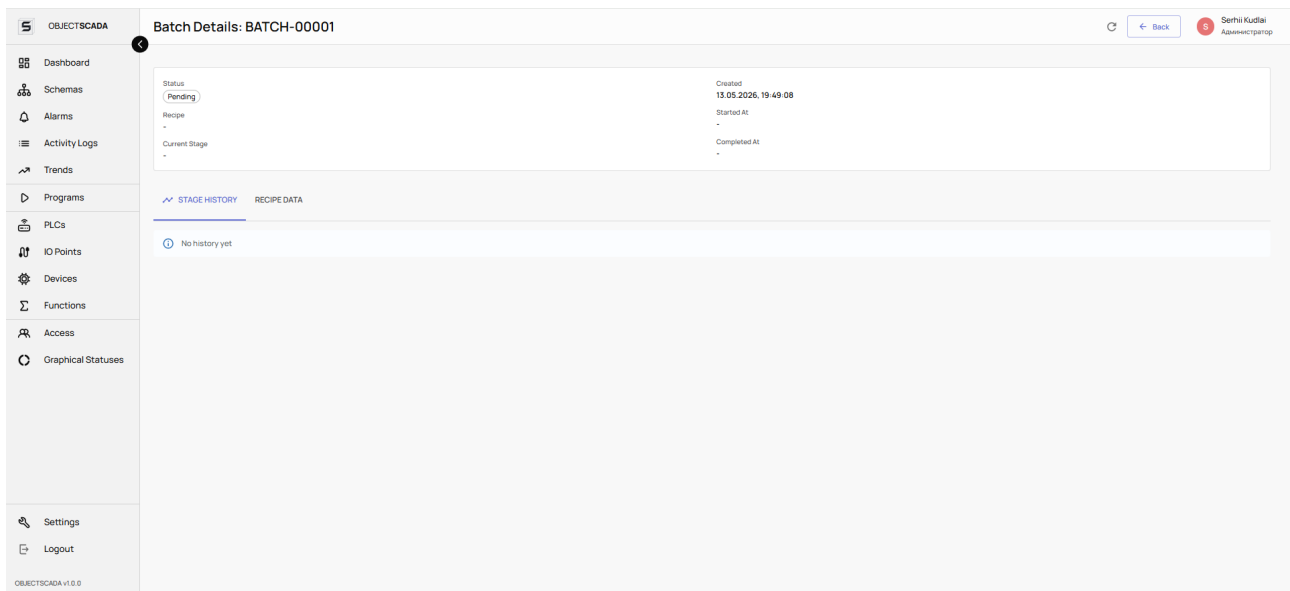
## 14. 生产线

### 对于用户

生产线将多个序列连接在一起，形成具有阶段的单一生产路线。

字段：

领域	描述
名称	线路名称 (+翻译)
阶段	生产线阶段列表，每个阶段都绑定到一个序列
舞台秩序	定义批处理路线
状态	活动/非活动



线路管理：包含名称、阶段计数、状态、编辑/删除按钮的表格。详细视图 - 通过拖放重新排序的阶段列表。

### 技术文档

生产线API：

方法	端点	描述
GET	/api/production-lines/	列表行
POST	/api/production-lines/	创建
PATCH	/api/production-lines/{id}/	更新
DELETE	/api/production-lines/{id}/	删除
POST	/api/production-lines/{id}/activate/	激活
POST	/api/production-lines/{id}/deactivate/	停用
GET	/api/production-line-stages/	所有阶
POST	/api/production-line-stages/	创建舞
PATCH	/api/production-line-stages/{id}/	更新

方法	端点	描述
DELETE	/api/production-line-stages/{id}/	删除

## 15. 趋势

### 对于用户

趋势以图表形式显示历史设备数据。

屏幕元素：

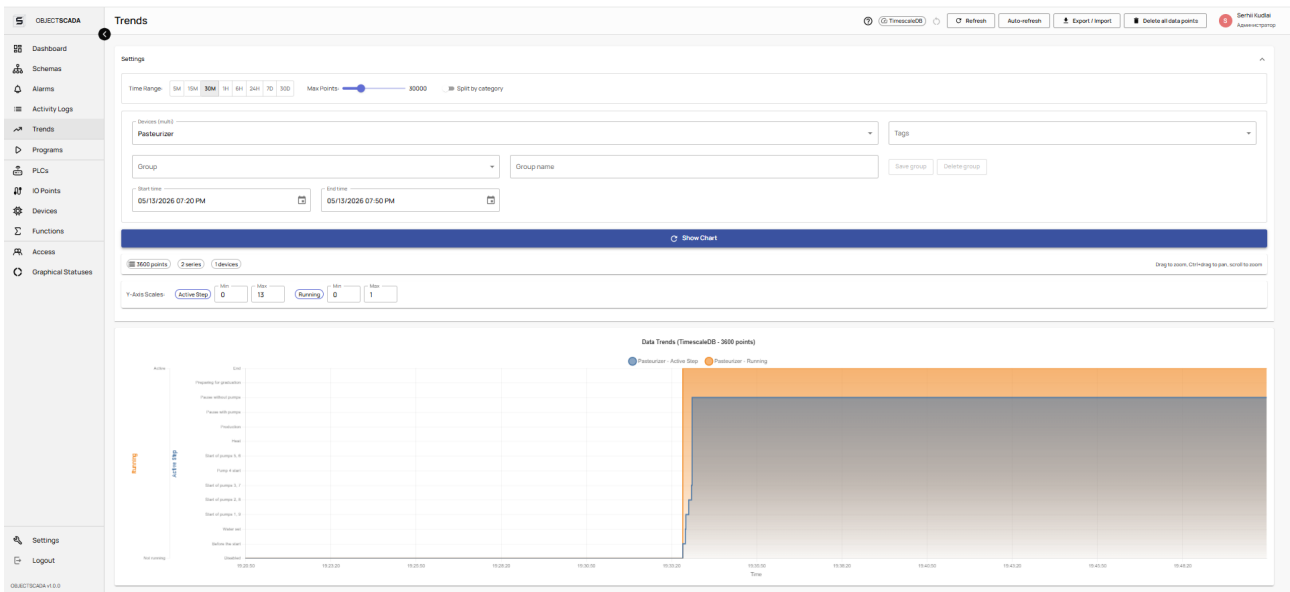
元素	描述
设备选择器	图表的多个设备/参数选择
时间范围	预设：5 分钟、15 分钟、30 分钟、1 小时、6 小时、24 小时、7 天、30 天
定制期	自由日期/时间范围
图表	带缩放功能的 Chart.js 折线图
设备组	保存/加载设备集
存储	磁盘使用信息

图表交互：

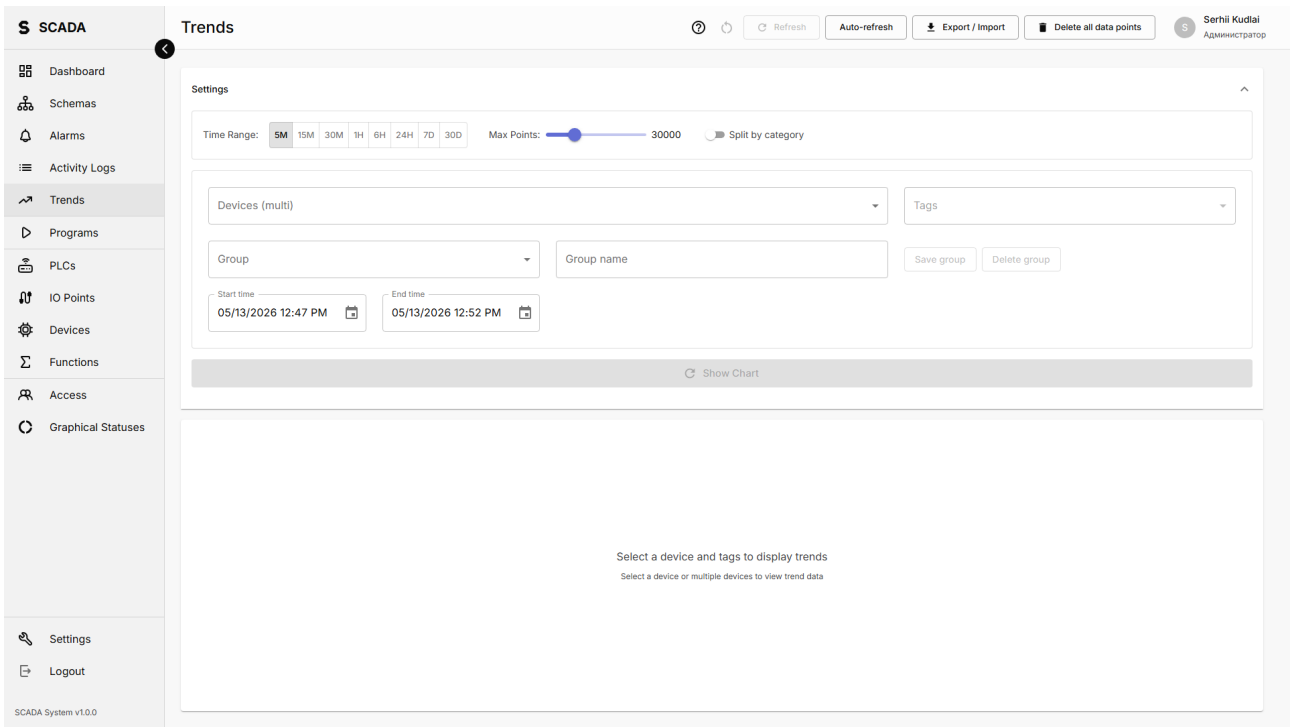
行动	描述
拖动	选择要放大的区域
滚动（鼠标滚轮）	放大
右键拖动	潘
悬停	带有值的工具提示
重置缩放	重置为原始比例

按钮：

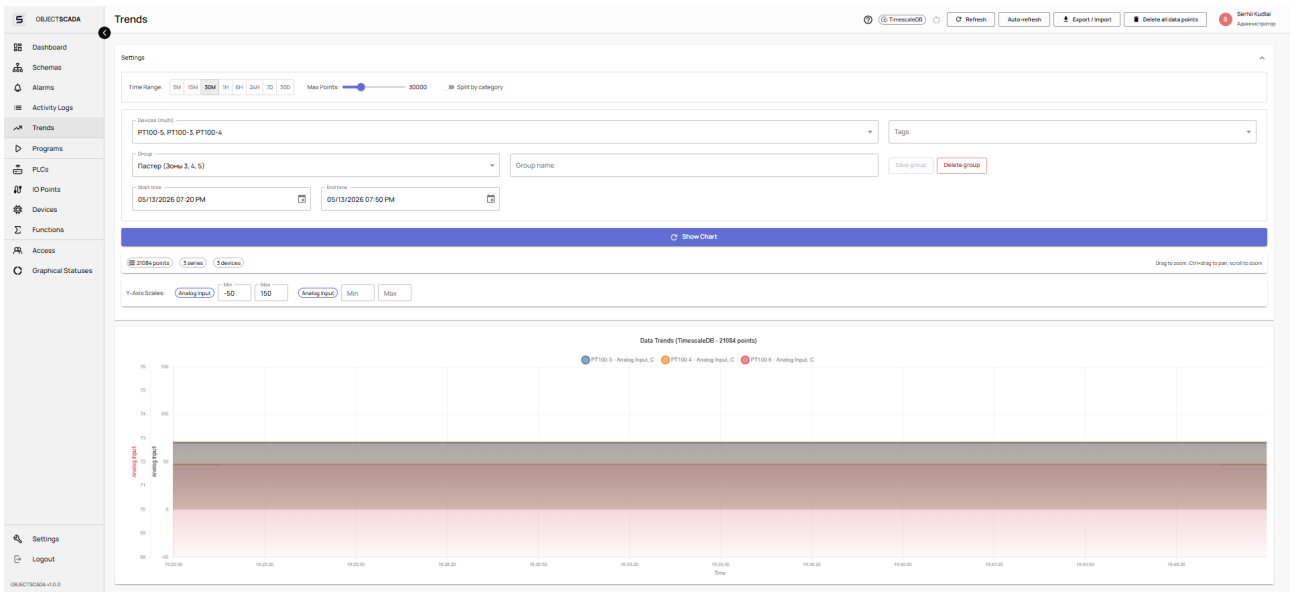
按钮	描述
添加设备	打开设备多选
保存组	保存当前设备集
负载组	选择已保存的组
删除群组	删除已保存的组
下载	导出数据（CSV/JSON）
下载图表	将图表导出为图像
删除数据	删除数据点（超级用户）
刷新	强制数据重新加载



完整趋势视图：顶部 — 设备多选（芯片）、时间范围按钮（5m/15m/1h/6h/24h/7d/30d）和 DateTimePicker。中心 — 具有多条颜色线的折线图。右 — 图例。



缩放后的图表：所选区域放大，重置缩放按钮可见。



保存的设备组下拉列表：《组 1 — 温度》、《组 2 — 压力》、保存/加载/删除按钮。

## 技术文档

### 趋势 API (TimescaleDB):

方法	端点	描述	参数
GET	/api/graphs/graph/	图表数据	?start_time=ISO&end_time=ISO&device_ids=id1,id2
GET	/api/graphs/available-parameters/	可用参数	—
GET	/api/graphs/timescale-status/	TimescaleDB 状态	—
DELETE	/api/graphs/delete-all/	删除所有数据	—
GET	/api/data-points/	数据点	?limit=100&offset=0

### 导出数据:

方法	端点	描述
POST	/api/project/export-graph-data/	导出图表
POST	/api/project/import-graph-data/	导入图表
GET	/api/project/graph-data-stats/	数据统计

## 16. 警报

### 对于用户

系统中所有警报事件的集中日志。

表格列：

专栏	描述
时间	发生日期和时间
严重性	错误（红色）/警告（黄色）
留言	报警说明
设备	来源
状态	主动/确认/解决
行动	控制按钮

过滤器：

过滤	描述
按严重程度	警告/错误/全部
按状态	活动/已确认/已解决/全部
按设备	设备选择器
搜索	文字搜索

按钮：

按钮	描述
确认	确认（注意）
解决	解决（问题已解决）
解决所有问题	解决所有警报
删除	删除条目
刷新	重新加载列表

**S SCADA**
Alarms & Events
Refresh
Serhii Kudlai  
Администратор

- Dashboard
- Schemas
- Alarms**
- Activity Logs
- Trends
- Programs
- PLCs
- IO Points
- Devices
- Functions
- Access
- Graphical Statures
- Settings
- Logout

Priority: All
Status: All
Device Tag: All

Occurrence Time	Device	Accident Type	Description	Severity	Status	Actions
5/12/2026, 8:07:49 PM	P-2	Motor P-2 alarm	Motor P-2: running but no feedback (automatic m...	High	Resolved	
5/12/2026, 7:55:24 PM	Unknown Device	Высокая температура PT100-6		Medium	Resolved	
5/8/2026, 7:59:26 PM	P-4	Motor P-4 alarm	Motor P-4: running but no feedback (manual mod...	High	Resolved	
5/8/2026, 7:59:16 PM	P-4	Motor P-4 alarm	Motor P-4: running but no feedback (manual mod...	High	Resolved	
4/28/2026, 5:24:52 PM	Unknown Device	Высокая температура PT100-4		Medium	Resolved	
4/6/2026, 11:41:30 AM	P-1_PID	Motor P-1_PID alarm	Motor P-1_PID: running but no feedback (automati...	High	Resolved	
4/6/2026, 11:40:40 AM	P-1_PID	Motor P-1_PID alarm	Motor P-1_PID: running but no feedback (automati...	High	Resolved	
4/6/2026, 11:40:09 AM	P-1_PID	Motor P-1_PID alarm	Motor P-1_PID: running but no feedback (automati...	High	Resolved	
4/6/2026, 11:39:39 AM	P-1_PID	Motor P-1_PID alarm	Motor P-1_PID: running but no feedback (automati...	High	Resolved	
4/6/2026, 11:31:12 AM	P-1_PID	Motor P-1_PID alarm	Motor P-1_PID: running but no feedback (automati...	High	Resolved	

Rows per page: 10
 < Back
1
2
3
4
Next >
1-10 of 36

警报页面：带有严重性颜色编码的表格（红色左侧条纹 — 错误，黄色 — 警告）。过滤器在顶部。每行中的确认/解决按钮。

## 技术文档

事故 API：请参阅第 10 节。

WebSocket：ws/accidents — 实时接收新的事件和状态更新。

## 17. 数学函数

### 对于用户

数学函数可让您计算设备数据和配方变量的派生值。

屏幕元素：

元素	描述
功能列表	功能表/卡
«+»按钮	创建一个函数
搜索	按名字

表达式编辑器（基于块）：

表达式是由按顺序添加的块构建的：

块式	描述
数量	常数数值
设备	当前设备值
配方变量	变量值
操作员	+、-、×、÷、%、^
括号	( 和 ) 用于分组
功能	sqrt、log10、圆形、地板、天花板

按钮：

按钮	描述
添加区块	向表达式
删除块	从表达式
验证	语法检查
测试	用当前值评估
保存	保存功能

ID	Name	Expression	Result	Units	Status	Last Calculation	Actions
func_014	Alarm PT-100-5 Cold	Pasteurizer - PT100_5 Pasteurizer - Cold_diff	72.0000	°C	Active	06.05.2026, 16:13:43	[Edit] [Delete]
func_016	RECIPE_70.PT100_1	Pasteurizer - PT100_1	35.0000	°C	Active	06.05.2026, 23:02:03	[Edit] [Delete]
func_020	RECIPE_70.PT100_2	Pasteurizer - PT100_2	62.0000	°C	Active	09.05.2026, 15:38:09	[Edit] [Delete]
func_019	RECIPE_70.PT100_3	Pasteurizer - PT100_3	71.0000	C	Active	06.05.2026, 16:13:43	[Edit] [Delete]
func_018	RECIPE_70.PT100_4	Pasteurizer - PT100_4	72.0000	C	Active	06.05.2026, 16:13:43	[Edit] [Delete]
func_017	RECIPE_70.PT100_5	Pasteurizer - PT100_5	73.0000	C	Active	06.05.2026, 16:13:43	[Edit] [Delete]
func_021	RECIPE_70.PT100_6	Pasteurizer - PT100_6	65.0000	C	Active	06.05.2026, 16:13:43	[Edit] [Delete]
func_022	RECIPE_70.PT100_7	Pasteurizer - PT100_7	55.0000	C	Active	06.05.2026, 16:13:43	[Edit] [Delete]
func_023	RECIPE_70.PT100_8	Pasteurizer - PT100_8	34.0000	C	Active	06.05.2026, 16:13:43	[Edit] [Delete]
func_015	pt100-4_cold	Pasteurizer - PT100_4 Pasteurizer - Cold_diff	71.0000	C	Active	06.05.2026, 16:13:43	[Edit] [Delete]
func_001	Авария PT100-1	Pasteurizer - PT100_1 Pasteurizer - Alarm_diff	90.0000	°C	Active	08.05.2026, 17:39:43	[Edit] [Delete]
func_002	Авария PT100-2	Pasteurizer - PT100_2 Pasteurizer - Alarm_diff	117.0000	°C	Active	09.05.2026, 15:38:09	[Edit] [Delete]
func_003	Авария PT100-3	Pasteurizer - PT100_3 Pasteurizer - Alarm_diff	126.0000	°C	Active	08.05.2026, 17:39:43	[Edit] [Delete]
func_004	Авария PT100-4	Pasteurizer - PT100_4 Pasteurizer - Alarm_diff	127.0000	°C	Active	08.05.2026, 17:39:43	[Edit] [Delete]

功能列表：带有名称、表达式、结果、单位列、编辑/删除/测试按钮的表格。

**Edit Function**

Name \*  
Alarm PT-100-5 Cold

Name translations  
Name per language. English is used as fallback.

Description

Mathematical Expression \*  
Pasteurizer - PT100\_5 + Pasteurizer - Cold\_diff

Units of Measurement  
°C

Update Interval (seconds)  
Leave empty for on-demand calculation

Active

Cancel Save

表达式编辑器：一行块 — [AI\_001] [×] [2.5] [+] [sqrt] [(] [AI\_002] [)]. 底部的块添加按钮。测试结果如右图所示。

## 技术文档

### 数学函数API:

方法	端点	描述
GET	/api/math-functions/	列表功
POST	/api/math-functions/	创建

方法	端点	描述
PUT	/api/math-functions/{id}/	更新
DELETE	/api/math-functions/{id}/	删除
POST	/api/math-functions/{id}/calculate/	计算
POST	/api/math-functions/{id}/validate_expression/	验证
POST	/api/math-functions/{id}/test_expression/	测试
POST	/api/math-functions/validate_all/	验证全部

### WebSocket: 数学函数更新

连接: ws://ws/math-functions

```
{
  "type": "math_function_update",
  "function_id": 5,
  "name": "Total Flow",
  "expression": "ai_001 * 2.5 + sqrt(ai_002)",
  "result": 157.3,
  "units": "L/h",
  "last_calculated": "2026-04-07T12:00:00Z",
  "calculation_error": null,
  "is_active": true
}
```

# 18. I/O点

## 对于用户

I/O 点定义物理 PLC 信号和逻辑 SCADA 设备之间的链接。

标签:

选项卡	描述
模拟输入	模拟输入 (4-20 mA, 0-10 V)
模拟输出	模拟输出
数字输入	离散输入
数字输出	离散输出

I/O 点形式 (IoPointDialog):

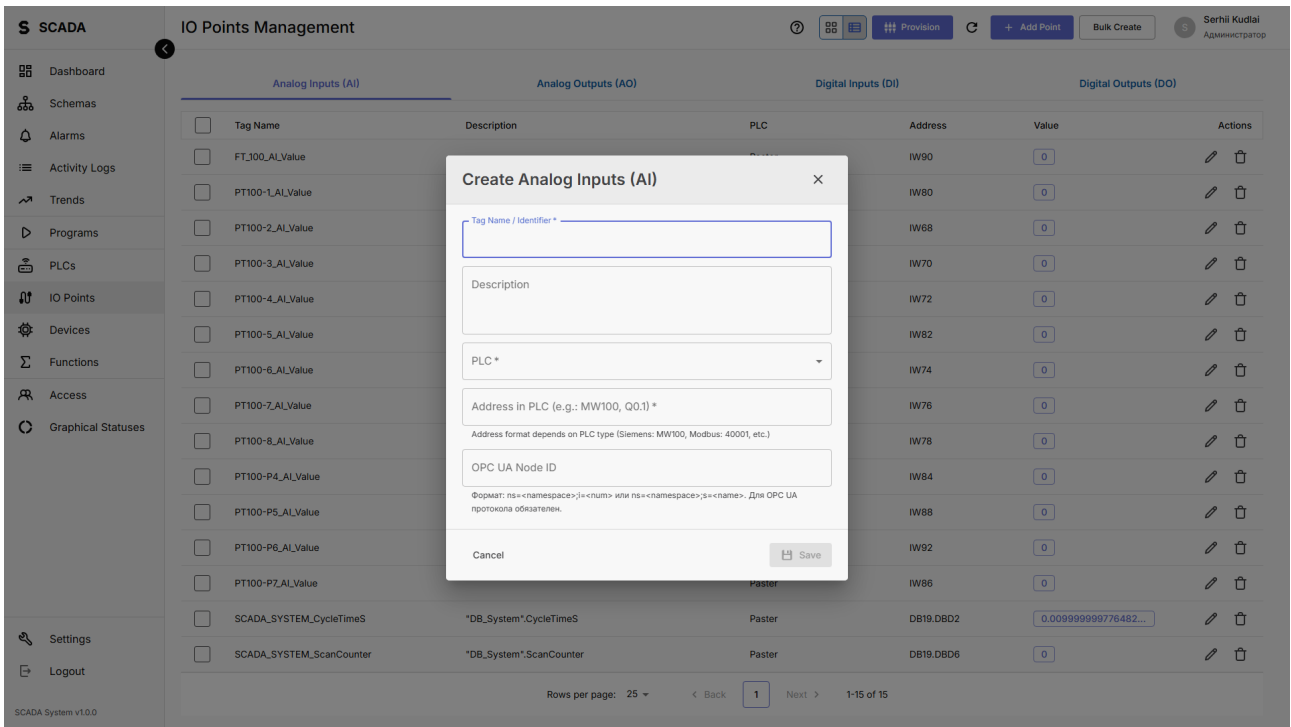
领域	描述
可编程逻辑控制器	控制器选择器
号码/地址	PLC地址 (IW0、QW0、IO.0、QO.0)
设备	绑定到逻辑设备
缩放	最小/最大 (模拟)
数据类型	整数、实数、布尔
单位	工程单位

The screenshot displays the 'IO Points Management' interface. It features a sidebar on the left with navigation options: Dashboard, Schemas, Alarms, Activity Logs, Trends, Programs, PLCs, IO Points, Devices, Functions, Access, Graphical Statuses, Settings, and Logout. The main content area is titled 'IO Points Management' and includes a top navigation bar with buttons for 'Provision', '+ Add Point', and 'Bulk Create'. Below this, there are four tabs: 'Analog Inputs (AI)', 'Analog Outputs (AO)', 'Digital Inputs (DI)', and 'Digital Outputs (DO)'. The 'Analog Inputs (AI)' tab is active, showing a table with the following data:

Tag Name	Description	PLC	Address	Value	Actions	
<input type="checkbox"/>	FT_100_AI.Value	Paster	IW90	0		
<input type="checkbox"/>	PT100-1_AI.Value	Paster	IW80	0		
<input type="checkbox"/>	PT100-2_AI.Value	Paster	IW68	0		
<input type="checkbox"/>	PT100-3_AI.Value	Paster	IW70	0		
<input type="checkbox"/>	PT100-4_AI.Value	Paster	IW72	0		
<input type="checkbox"/>	PT100-5_AI.Value	Paster	IW82	0		
<input type="checkbox"/>	PT100-6_AI.Value	Paster	IW74	0		
<input type="checkbox"/>	PT100-7_AI.Value	Paster	IW76	0		
<input type="checkbox"/>	PT100-8_AI.Value	Paster	IW78	0		
<input type="checkbox"/>	PT100-P4_AI.Value	Paster	IW84	0		
<input type="checkbox"/>	PT100-P5_AI.Value	Paster	IW88	0		
<input type="checkbox"/>	PT100-P6_AI.Value	Paster	IW92	0		
<input type="checkbox"/>	PT100-P7_AI.Value	Paster	IW86	0		
<input type="checkbox"/>	SCADA_SYSTEM_CycleTimeS	*DB_System*.CycleTimeS	Paster	DB19.DBD2	0.009999999776482...	
<input type="checkbox"/>	SCADA_SYSTEM_ScanCounter	*DB_System*.ScanCounter	Paster	DB19.DBD6	0	

At the bottom of the table, there is a pagination control: 'Rows per page: 25', '< Back', '1', 'Next >', and '1-15 of 15'.

I/O 点页面: 4 个选项卡 (AI、AO、DI、DO)、包含 PLC、地址、设备、比例、单位列、添加/编辑/删除按钮的表格。



I/O 点对话框：PLC 下拉列表、地址字段、设备下拉列表、最小/最大刻度字段、单位。

## 技术文档

### I/O点API:

方法	端点	描述
GET	/api/points/analog-inputs/	AI点
POST	/api/points/analog-inputs/	创建人工智能
GET	/api/points/analog-outputs/	AO点
POST	/api/points/analog-outputs/	创建 AO
GET	/api/points/digital-inputs/	DI 点
POST	/api/points/digital-inputs/	创建 DI
GET	/api/points/digital-outputs/	DO 点
POST	/api/points/digital-outputs/	创建 DO

对于每种类型：标准 CRUD (GET、POST、PUT、PATCH、DELETE) + /validate/ 和 /test/。

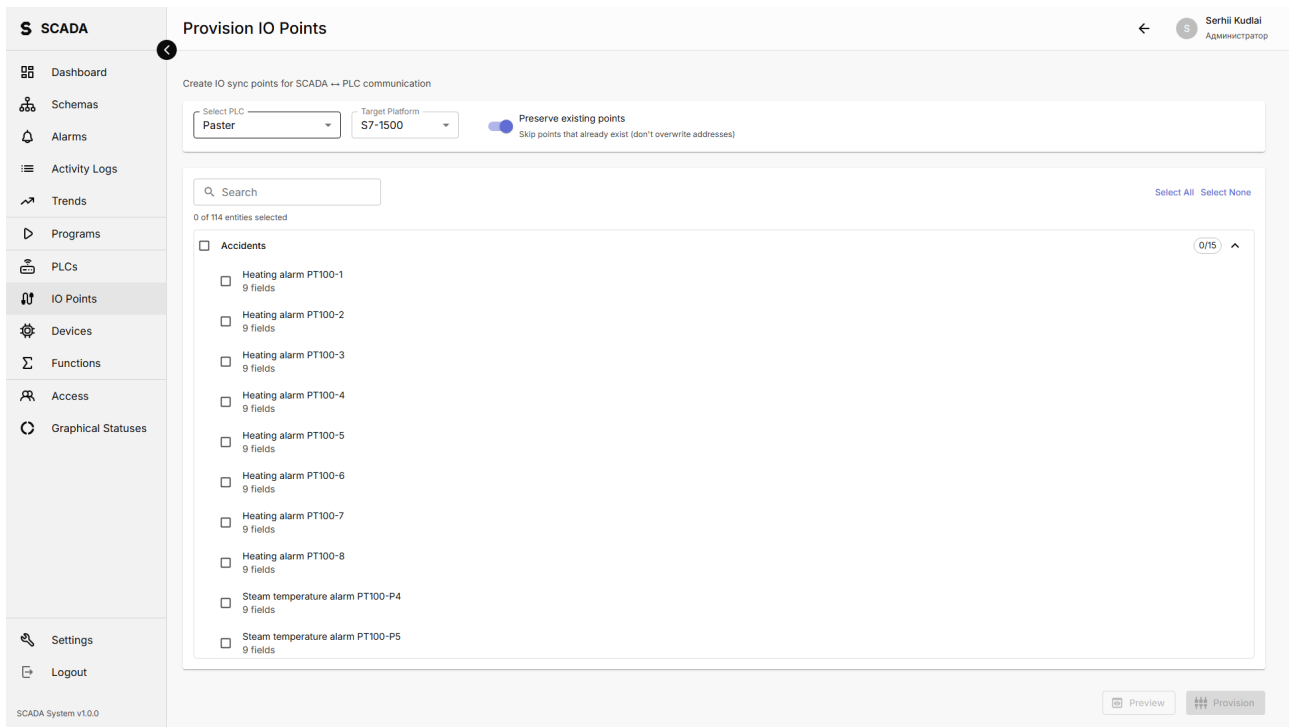
# 19. I/O 配置

## 对于用户

根据设备配置批量创建 I/O 点。

屏幕元素：

元素	描述
PLC选择器	目标控制器
PLC类型	S7-1200 或 S7-1500
保留地址	复选框 — 不覆盖现有地址
实体清单	每种类型的复选框 (电机、阀门、PID、AI、AO、DI、DO、COS、计数器、计时器、水箱、序列、事件、事故)
预览	结果预览
规定	运行配置



配置页面：PLC 下拉列表、S7-1200/S7-1500 无线电、带有复选框的实体列表、预览和配置按钮。

## 技术文档

配置API：

方法	端点	描述
POST	/api/plc-export/provision-io-points/preview/	预览
POST	/api/plc-export/provision-io-points/	执行
GET	/api/plc-export/provision-io-points/entities/	可用实体

## 20. 图形状态

### 对于用户

图形状态是在助记符模式上显示设备状态的图标模板。

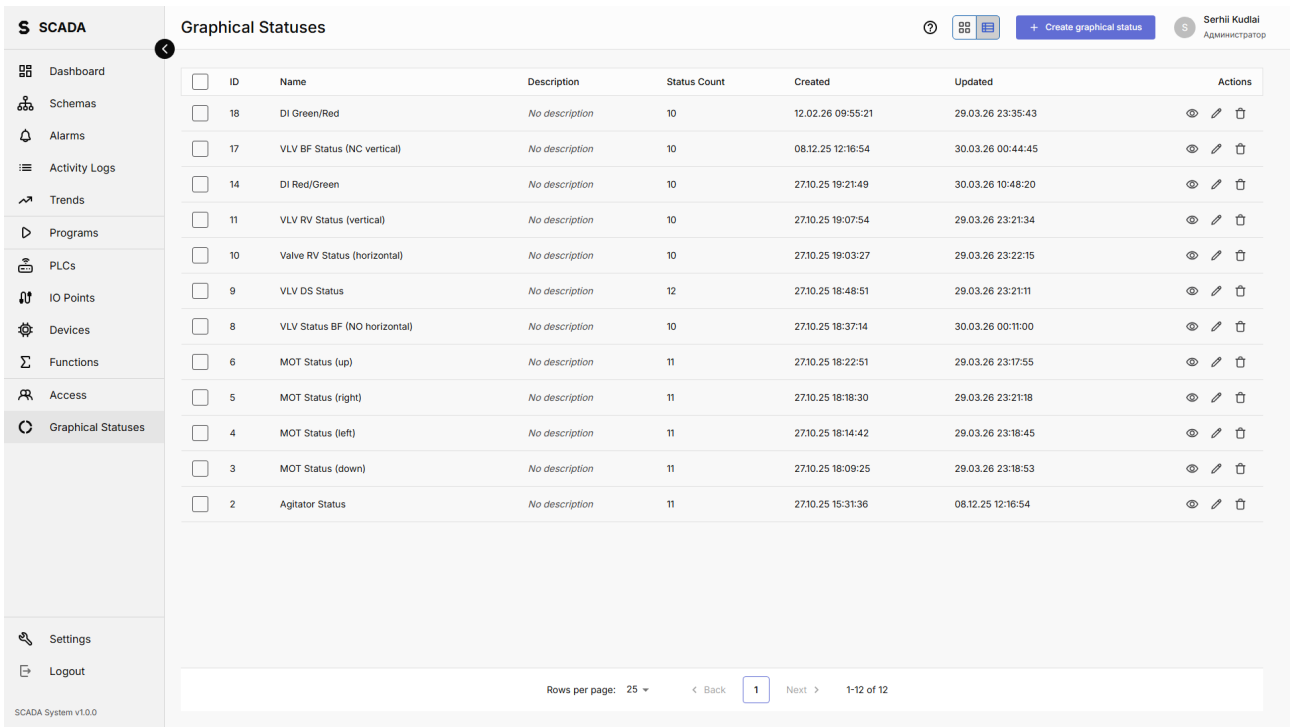
要素：

元素	描述
模板列表	卡片/桌子
«+»按钮	创建模板

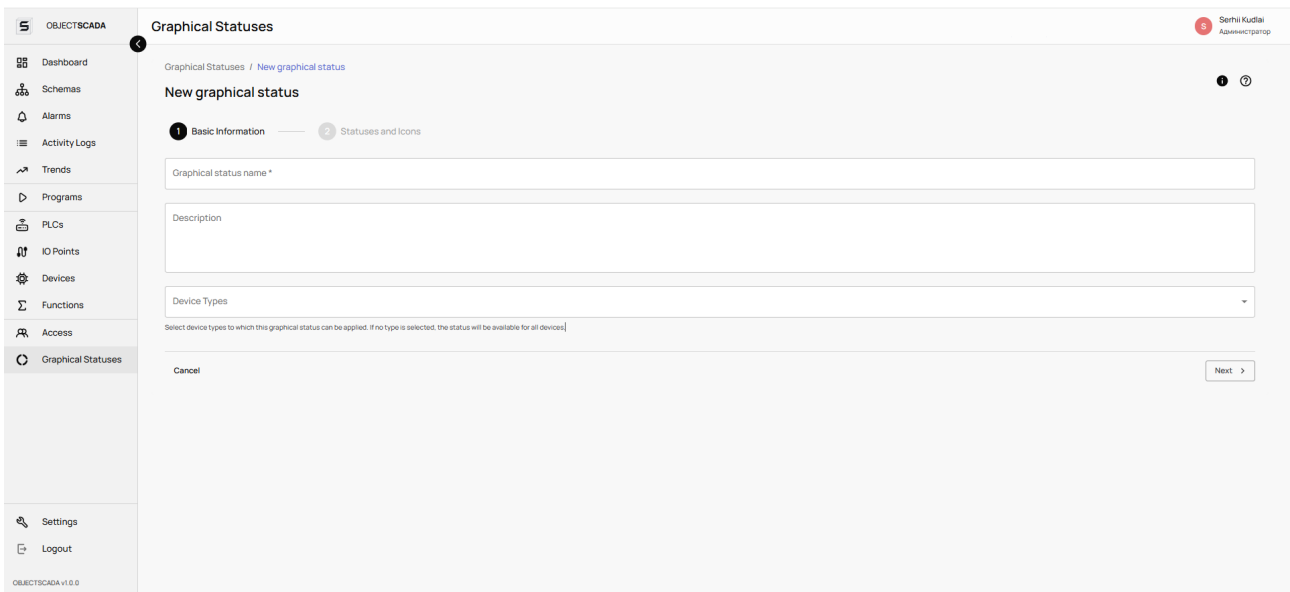
创建向导：

- 第 1 步：名称、描述、设备类型
- 第 2 步：状态和图标
- 状态代码 → 图标 (URL 或上传)
- 可用图标库
- 上传自定义图标

按钮	描述
添加状态	新的“代码→图标”行
删除状态	删除一行
上传图标	上传图标文件
画廊	从之前上传的中挑选
上一页/下一页	向导导航
保存	保存模板



图形状态列表：带有图标预览、模板名称、设备类型的卡片。



创建向导：左—状态列表（代码+图标预览），右—图标库供选择。

## 技术文档

### 图形状态API:

方法	端点	描述
GET	/api/graphical-statuses/	列表模板
POST	/api/graphical-statuses/	创建
PUT	/api/graphical-statuses/{id}/	更新
DELETE	/api/graphical-statuses/{id}/	删除
POST	/api/graphical-statuses/upload-icon/	上传图标（多部分）

方法	端点	描述
GET	/api/graphical-statuses/list-icons/	列表图标

## 21. 硬件配置器

### 对于用户

用于 TIA Portal 部署的 PLC 硬件配置的可视化编辑器。

标签：

选项卡	描述
机架视图	带有模块的可视化机架编辑器
PROFINET 拓扑	设备网络拓扑
地址表	I/O地址表

机架编辑器：

元素	描述
机架插槽	机架中模块的可视化表示
模块目录	带有可用模块的侧面板（拖放）
属性面板	选定模块配置
添加模块	从目录中拖动
删除模块	从插槽中移除

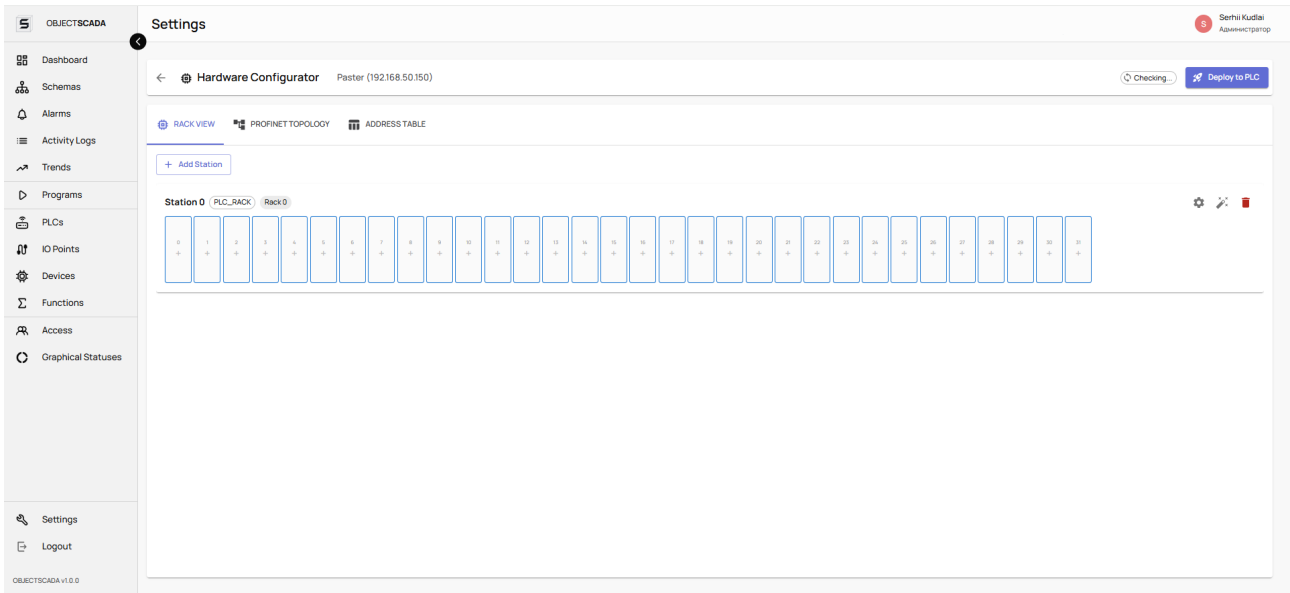
工具栏按钮：

按钮	描述
部署	将配置部署到 TIA Portal
下载	下载配置
进口GSD	导入设备描述符 (GSD/GSDML)
TIA 代理状态	TIA Agent 连接指示灯
返回	返回PLC列表

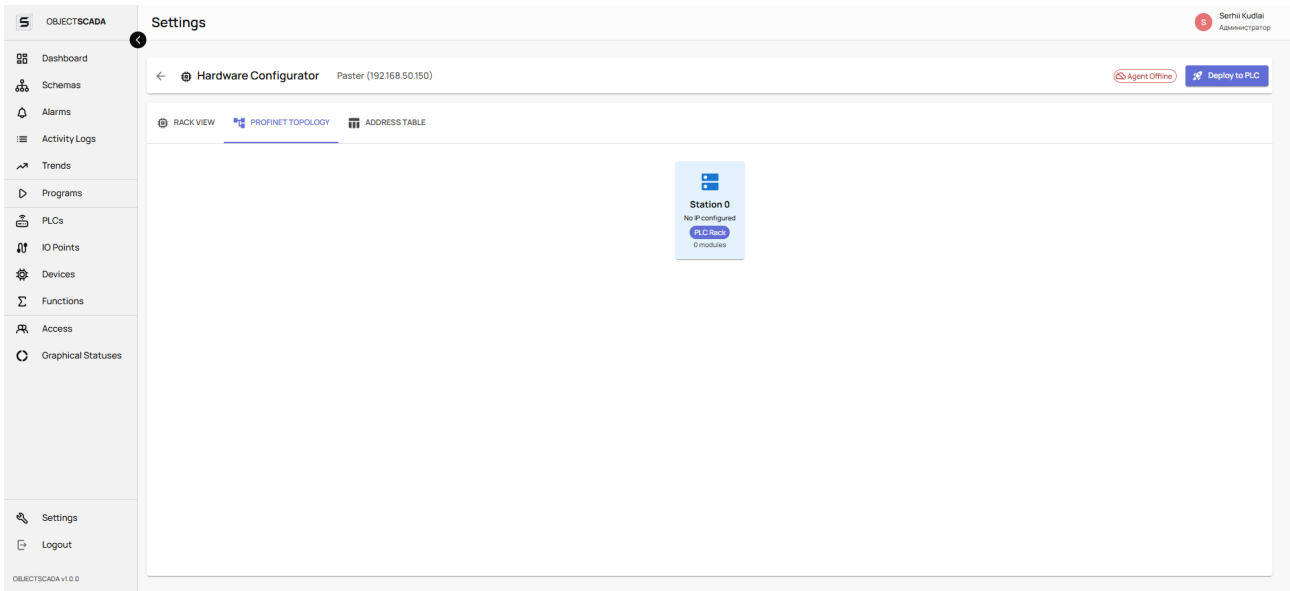
对话框：

对话	描述
TiaDeployDialog	部署：目标系统、进度、状态
GsdImportDialog	GsdImportDialog
网络配置对话框	网络设置：IP、掩码、网关、PROFINET 参数

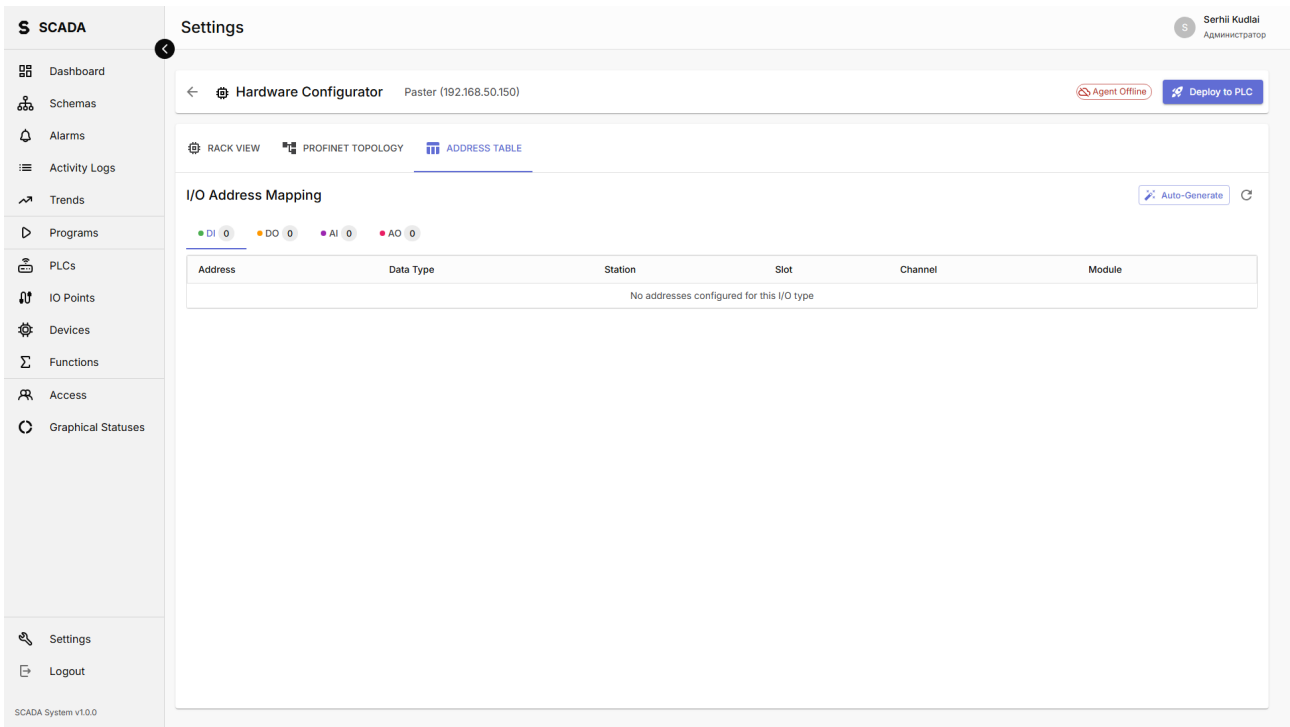
GSD导入：



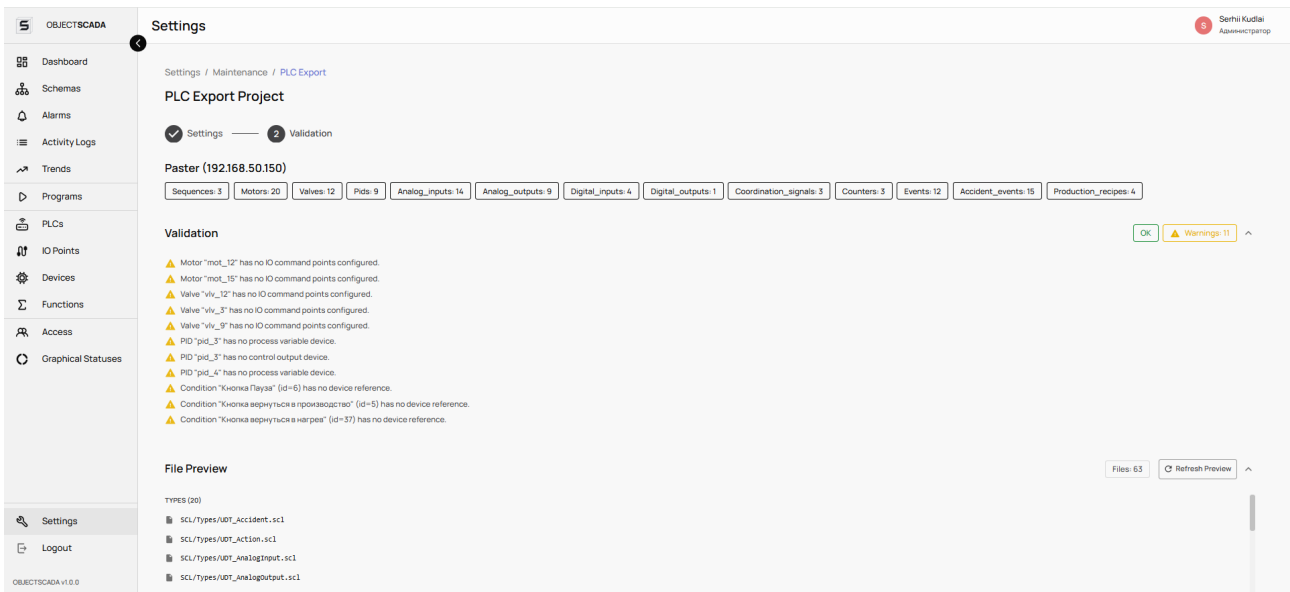
机架编辑器：插槽中带有模块的可视化机架，左侧为模块目录（滑出面板），右侧为属性面板，工具栏中为“部署”按钮。



PROFINET 拓扑：带有设备节点及其之间链路的网络图。



地址表：模块、插槽、通道、地址、类型 (I/O)、设备列。



部署对话框：TIA Agent 选择器、进度条、操作日志、部署/取消按钮。

## 技术文档

### 配置器API:

方法	端点	描述
GET	/api/hw-catalog/	模块目录
POST	/api/hw-catalog/import-gsd/	导入 GSD (多部分)
GET	/api/hw-stations/	车站
POST	/api/hw-stations/	创建一个站
POST	/api/hw-stations/{id}/auto-assign-addresses/	自动分配地址

方法	端点	描述
GET	/api/hw-slots/	老虎机
POST	/api/hw-slots/	创建插槽
POST	/api/hw-slots/bulk-update/	批量更新
GET	/api/hw-network/	网络配置
POST	/api/tia/deploy/	部署到 TIA Portal
GET	/api/tia-deploy-jobs/{id}/progress/	部署进度
GET	/api/tia-agent-config/	TIA 代理配置
GET	/api/tia-agent-config/{id}/check-connection/	连接测试

## 22. 用户管理

### 对于用户

创建、编辑和删除用户帐户。

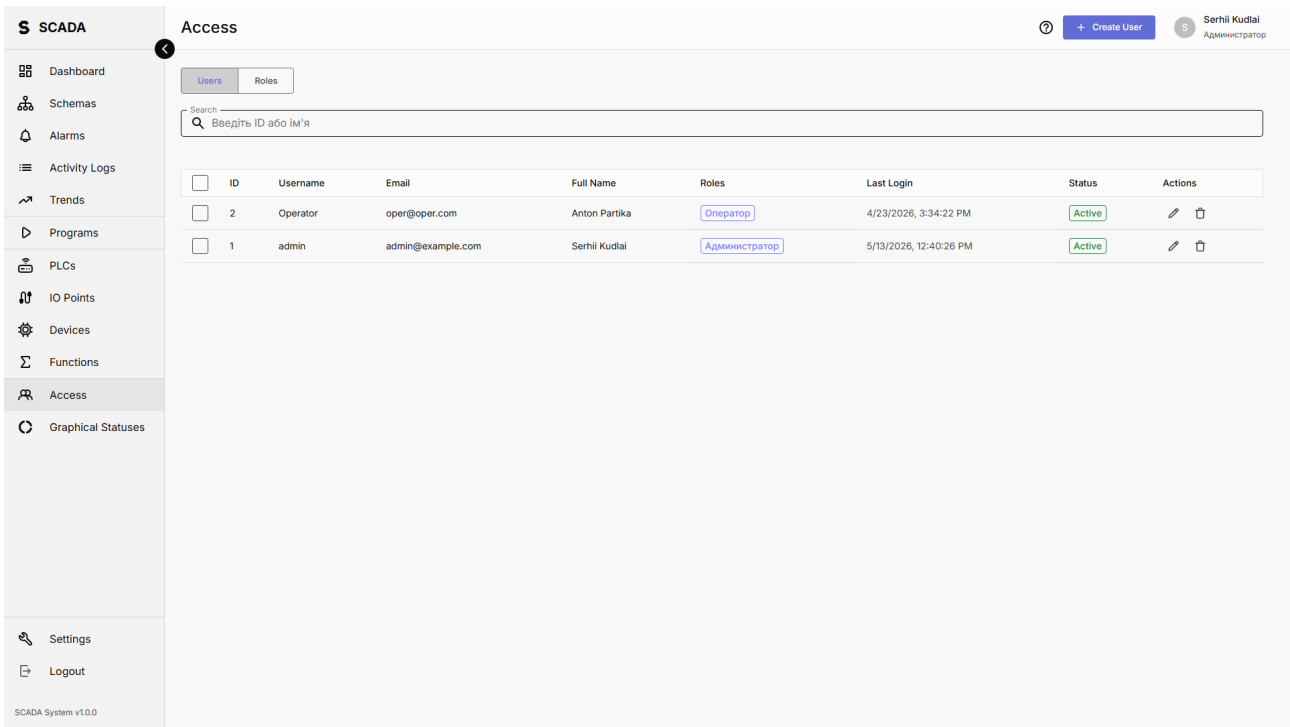
用户表：

专栏	描述
用户名	登录
全名	名字和姓氏
电子邮件	电子邮件地址
角色	分配的角色
上次登录	日期和时间
状态	活动/非活动
行动	编辑/删除

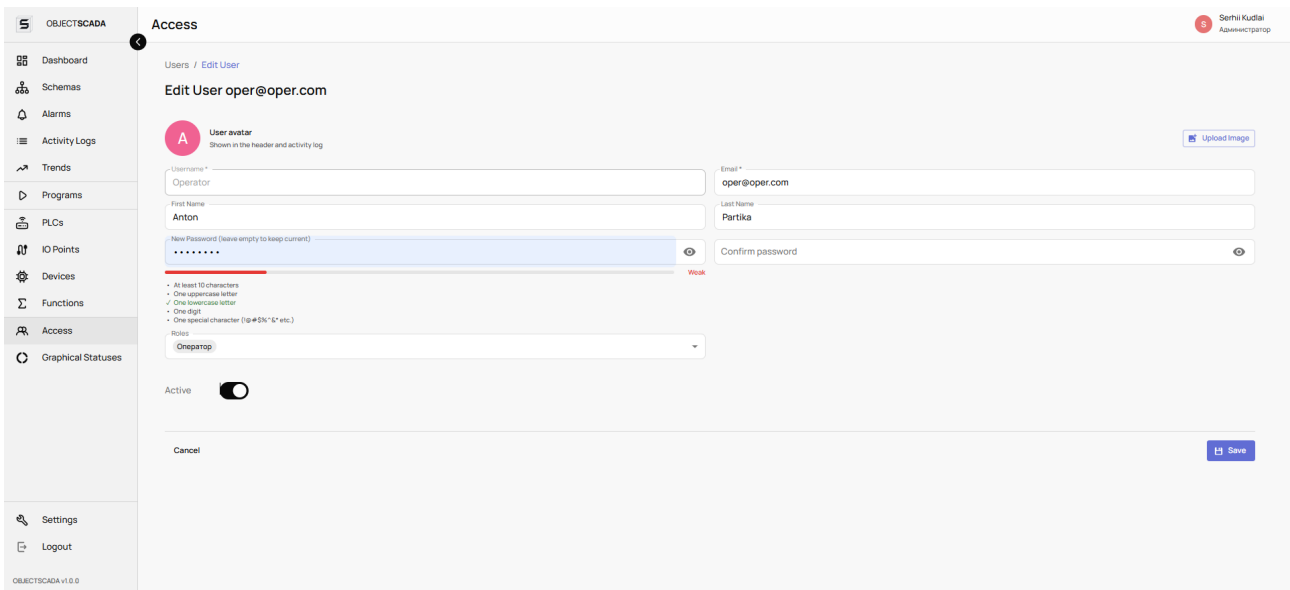
用户表单：

领域	描述
用户名	独特的登录
电子邮件	电子邮件地址
密码	密码（创建时）
确认	密码重复
名字/姓氏	真实姓名
头像	个人资料照片上传
角色	角色多选
状态	活动/非活动切换

按钮	描述
保存	保存更改
取消	不保存返回
删除	删除帐户（仅限编辑模式）
更改密码	更改用户密码



带有列、头像、角色颜色徽章、状态指示器的用户表。



创建/编辑表单：用户名、电子邮件、密码、角色选择器、头像上传、状态切换。

## 技术文档

### 用户API:

方法	端点	描述
GET	/api/users/	列出用户
POST	/api/users/	创建
GET	/api/users/{id}/	检索
PUT	/api/users/{id}/	更新
DELETE	/api/users/{id}/	删除

方法	端点	描述
GET	/api/users/me/	当前用户
POST	/api/users/{id}/set_password/	更改密码
POST	/api/users/{id}/upload-avatar/	上传头像 (多部分)
GET	/api/users/{id}/roles/	用户角色
POST	/api/users/{id}/roles/	分配角色
DELETE	/api/users/{id}/roles/	撤销角色

## 23.角色管理

### 对于用户

基于角色的访问控制。每个角色定义一组权限。

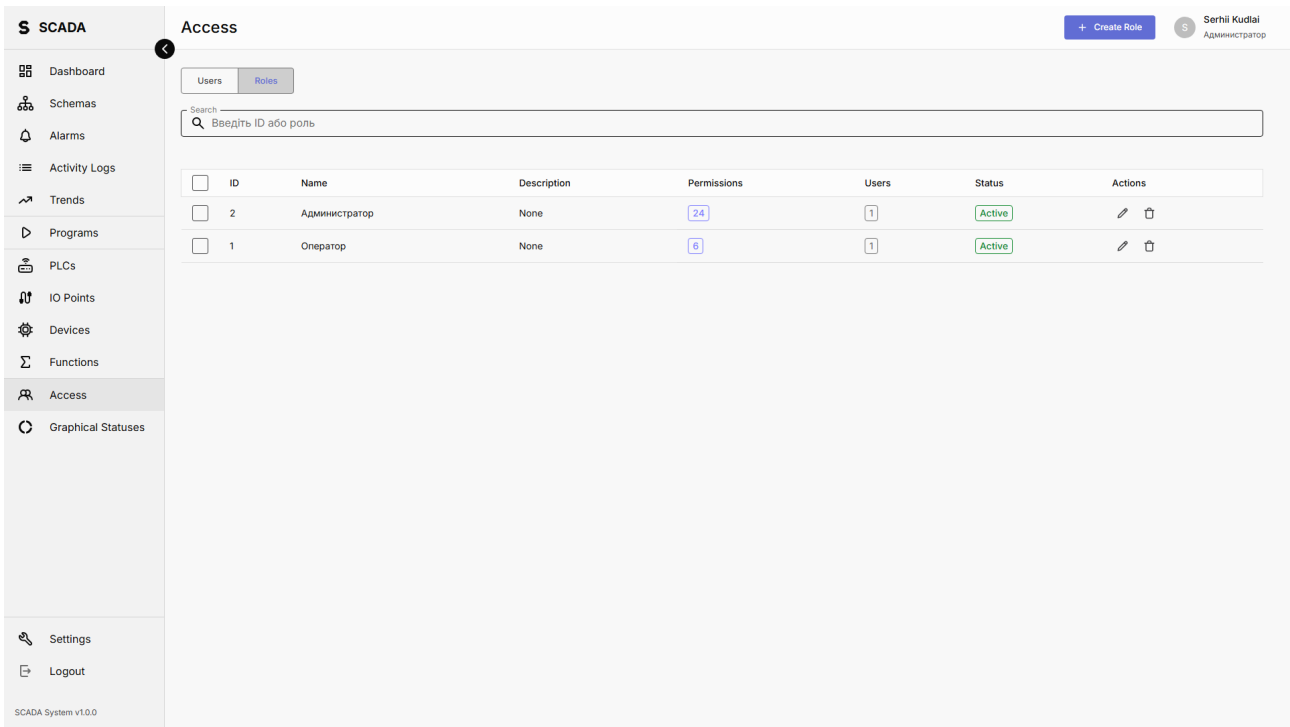
角色表：

专栏	描述
名称	角色名称
描述	文字说明
用户	具有此角色的用户数量
行动	编辑/删除

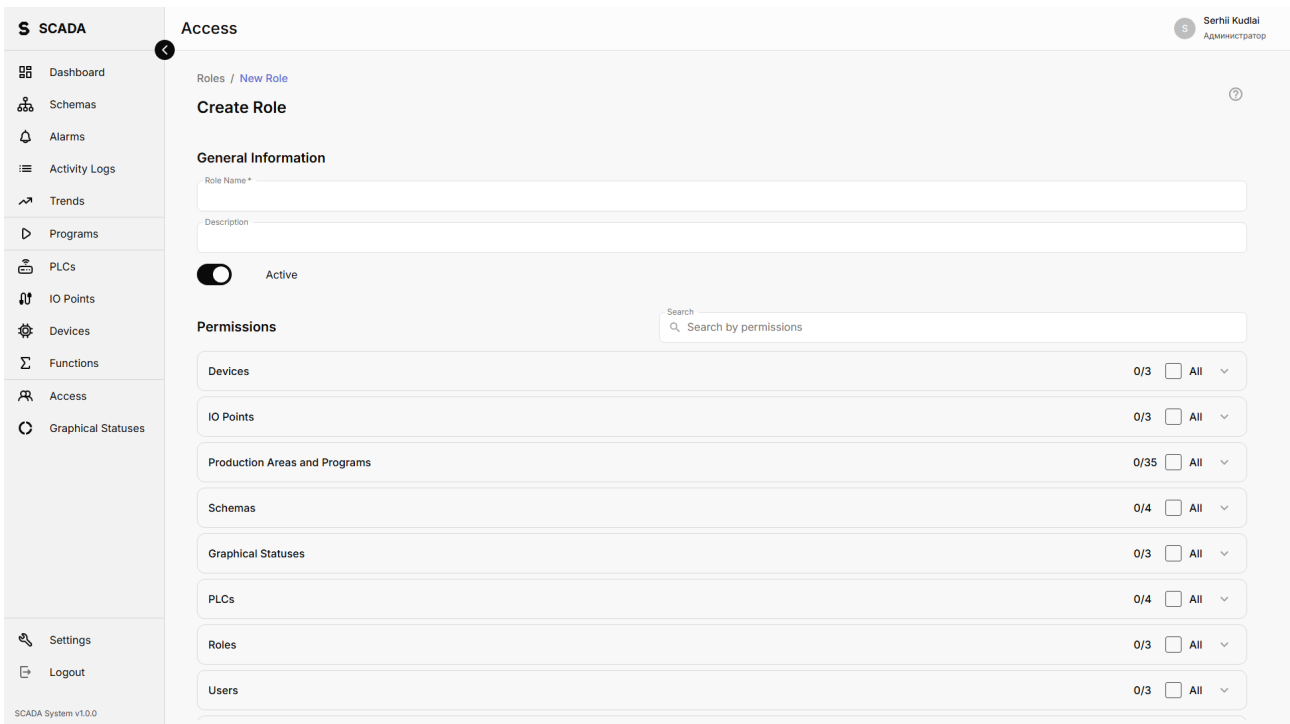
权限矩阵（以角色形式）：

类别	权限
设备	创建、查看、编辑、删除
PLC	创建、查看、编辑、删除、硬件配置
序列	创建、查看、编辑、删除、复制
架构	创建、查看、编辑、删除
I/O 点	创建、查看、编辑、删除
用户	创建、查看、编辑、删除
角色	创建、查看、编辑、删除
事故	确认、解决、删除
趋势	查看、导出、删除数据
图形状态	创建、查看、编辑、删除
管理	连接、许可、更新

按钮	描述
选择全部	选择每个权限
取消全选	清除所有权限
保存	保存角色



角色列表：包含名称、描述、用户计数、编辑/删除按钮的表。



角色形式：名称、描述、按类别划分的复选框矩阵（设备、PLC、序列、模式...）。全选/取消全选按钮。

## 技术文档

### 角色API:

方法	端点	描述
GET	/api/roles/	列出角色
POST	/api/roles/	创建

方法	端点	描述
GET	/api/roles/{id}/	检索
PATCH	/api/roles/{id}/	更新
DELETE	/api/roles/{id}/	删除
GET	/api/roles/{id}/permissions/	角色权限
GET	/api/permissions/categories/	权限类别
GET	/api/permissions/actions/	权限操作

## 24. 活动日志

### 对于用户

审核系统中的所有用户操作。

屏幕元素：

专栏	描述
时间	行动日期和时间
用户	谁执行了该操作
行动	操作类型（创建、更新、删除、登录...）
描述	行动详情
IP 地址	来源IP

过滤器：

过滤	描述
按用户	用户选择器
通过行动	操作类型
按日期	日期范围
搜索	文字搜索

统计数据：

- 动作总数
- 用户总结
- 按行动类型总结

活动日志：顶部带有过滤器的表格，右侧为统计信息（按操作类型和用户列出的图表）。底部分页。

## 技术文档

### 活动日志API:

方法	端点	描述	参数
GET	/api/user-activity-logs/	日志列表	?user_id=X&action_type=X&limit=100&offset=0
GET	/api/user-activity-logs/recent/	过去 24 小时	—
GET	/api/user-activity-logs/statistics/	统计	—
DELETE	/api/user-activity-logs/clear/	清除（管理员）	—

### WebSocket: 实时日志

连接: ws:///ws/activity-log

```
{
  "type": "activity_log_entry",
  "id": 1234,
  "action_type": "device_update",
  "description": "Updated motor MOT_001 mode to manual",
  "user": {"id": 1, "username": "admin"},
  "content_object": {"type": "motor", "id": "mot_001"},
  "metadata": {},
  "created_at": "2026-04-07T12:00:00Z"
}
```

## 25. 设置

### 对于用户

设置页面有多个选项卡：

#### 常规选项卡

参数	描述
主题	配色方案选择器 (蓝色、绿色、橙色、粉色、紫色、红色)
深色模式	浅色/深色切换
语言	界面语言选择器 (EN、UK、RU、DE、FR、PL、AR...)
设计版	V1 / V2
人工智能助手	启用/禁用AI助手

#### 通知选项卡

参数	描述
电子邮件通知	开/关
推送通知	开/关
短信通知	开/关
严重性过滤器	通知的最低严重性

#### 连接选项卡

元素	描述
TIA 代理商名单	TIA Portal 连接配置
添加	新的 TIA Agent (主机、端口)
测试	测试连接
状态指示灯	绿色/红色

#### 许可证选项卡

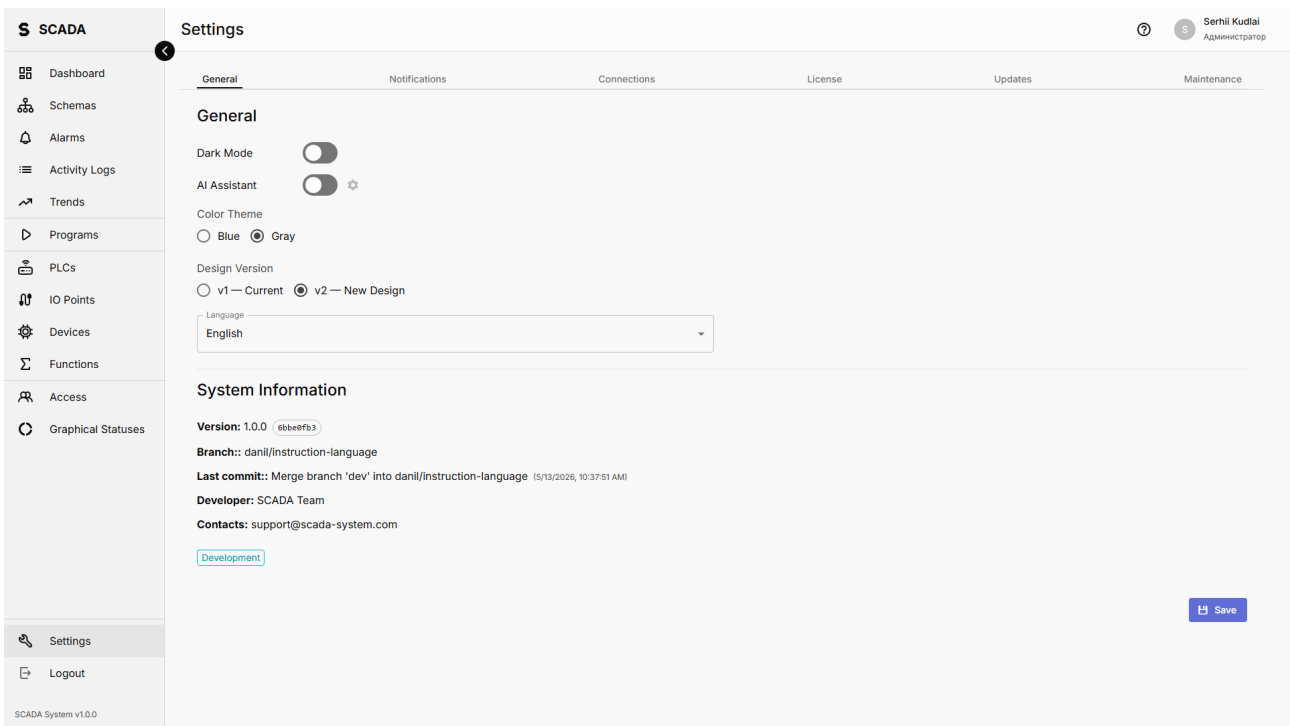
领域	描述
状态	有效/无效/试用
类型	终身/试用/订阅
剩余天数	倒计时
公司	公司名称
有效期	许可证有效期
安装 ID	唯一的安装标识符

#### 更新选项卡

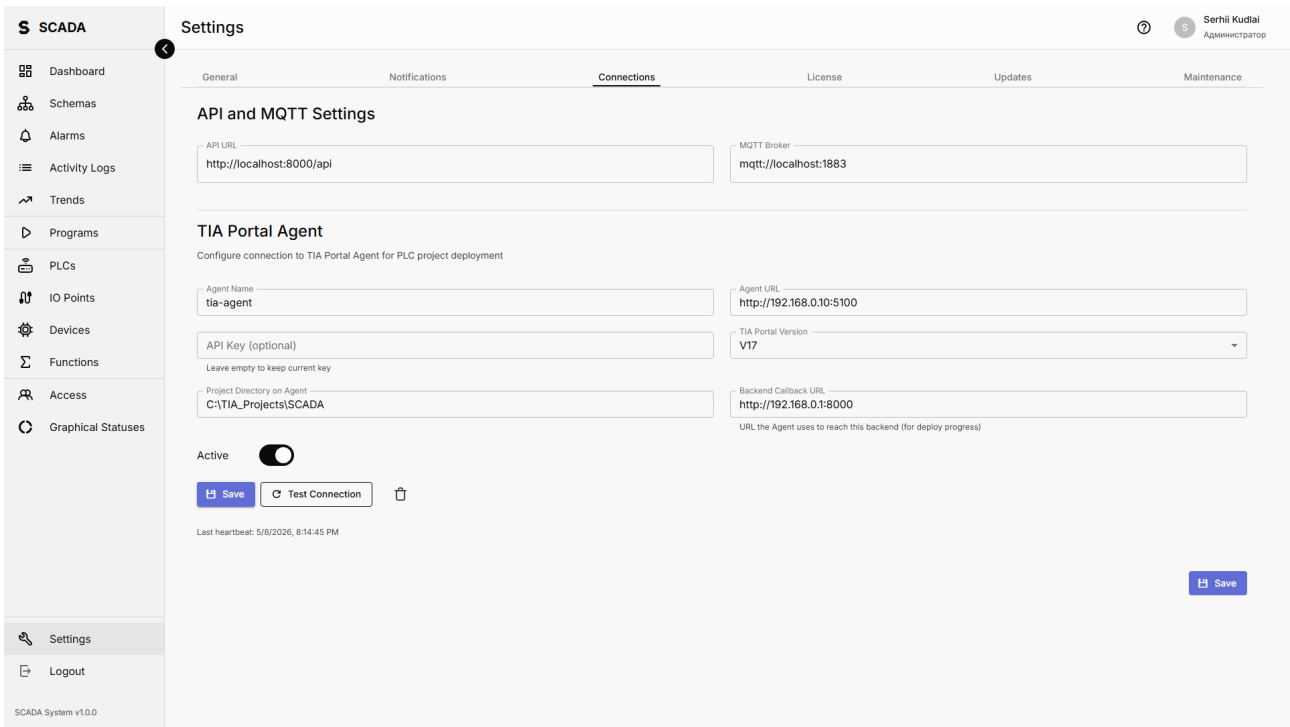
元素	描述
当前版本	当前版本号
检查更新	检查按钮
下载更新	下载按钮
进展	更新进度指示器
自动更新	切换

### 维护选项卡

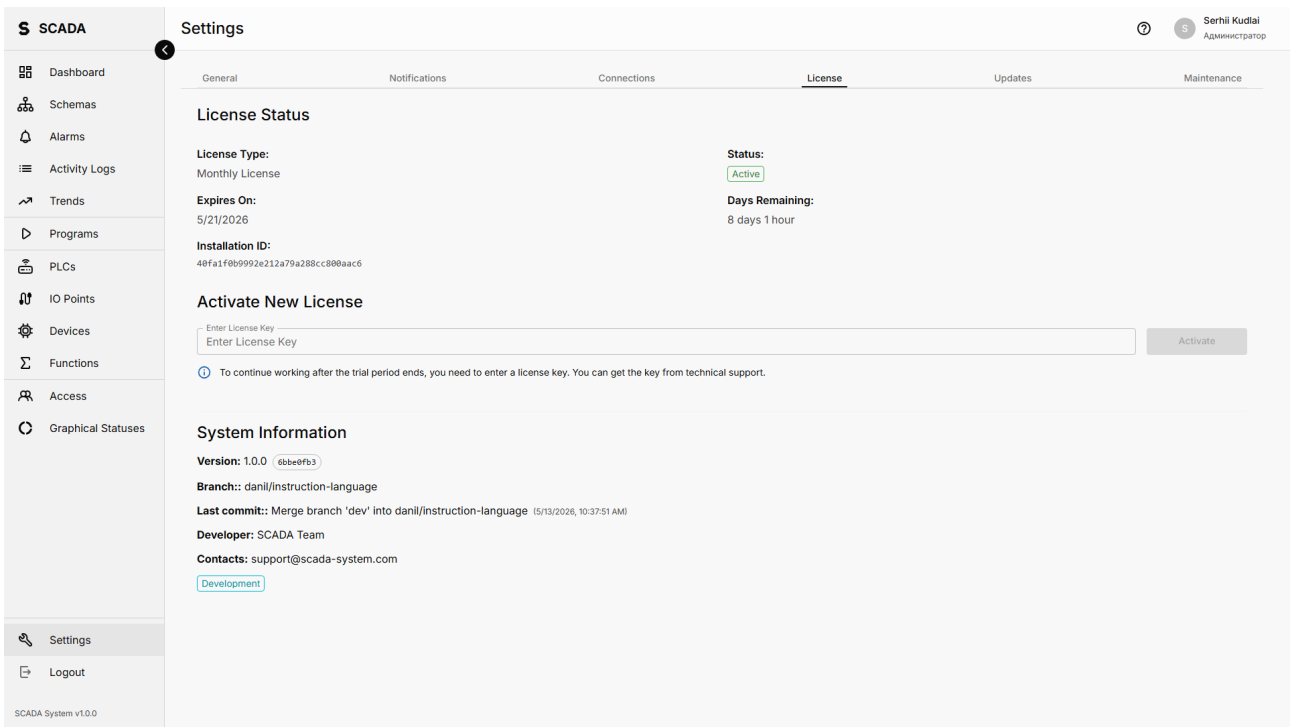
元素	描述
数据库	DB维护操作
缓存	清除缓存
日志	日志管理
系统健康状况	健康检查
备份/恢复	备份与恢复
磁盘清理	删除临时文件
烟雾测试	运行时健康检查



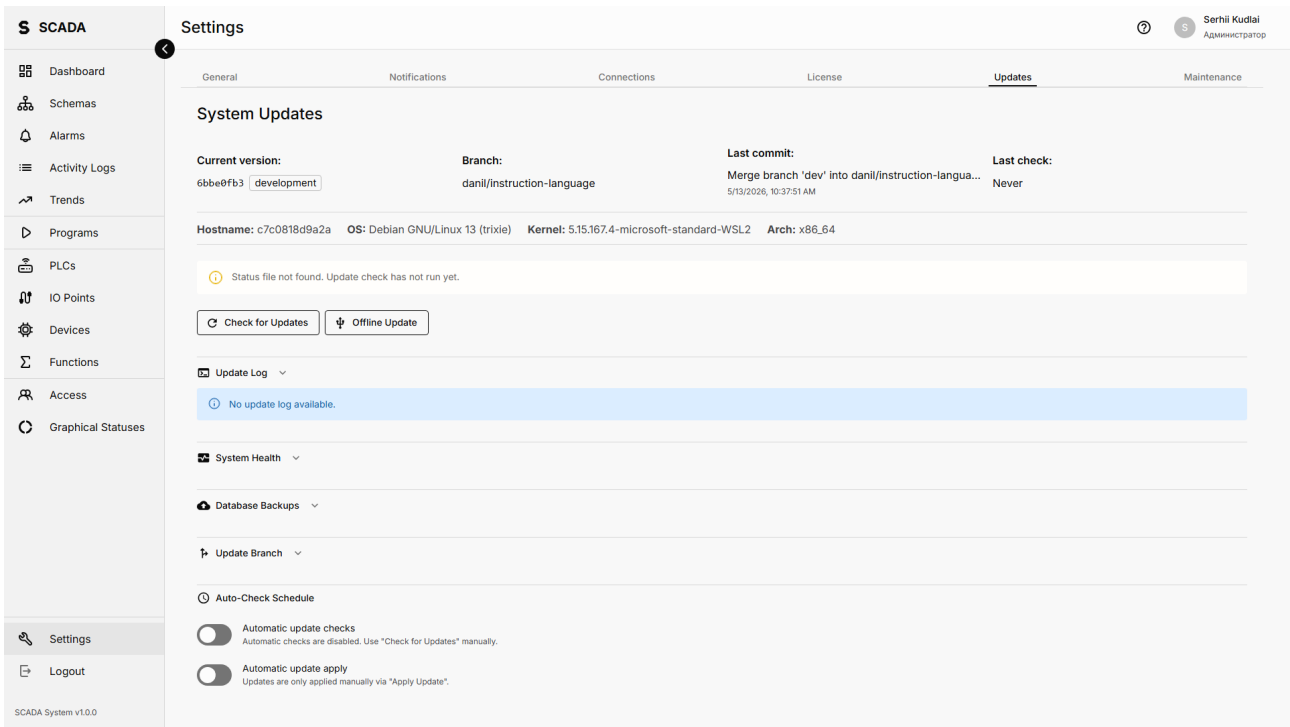
常规选项卡：深色模式切换、颜色选择器（6种颜色）、语言下拉菜单、设计 V1/V2 切换、AI 助手切换。



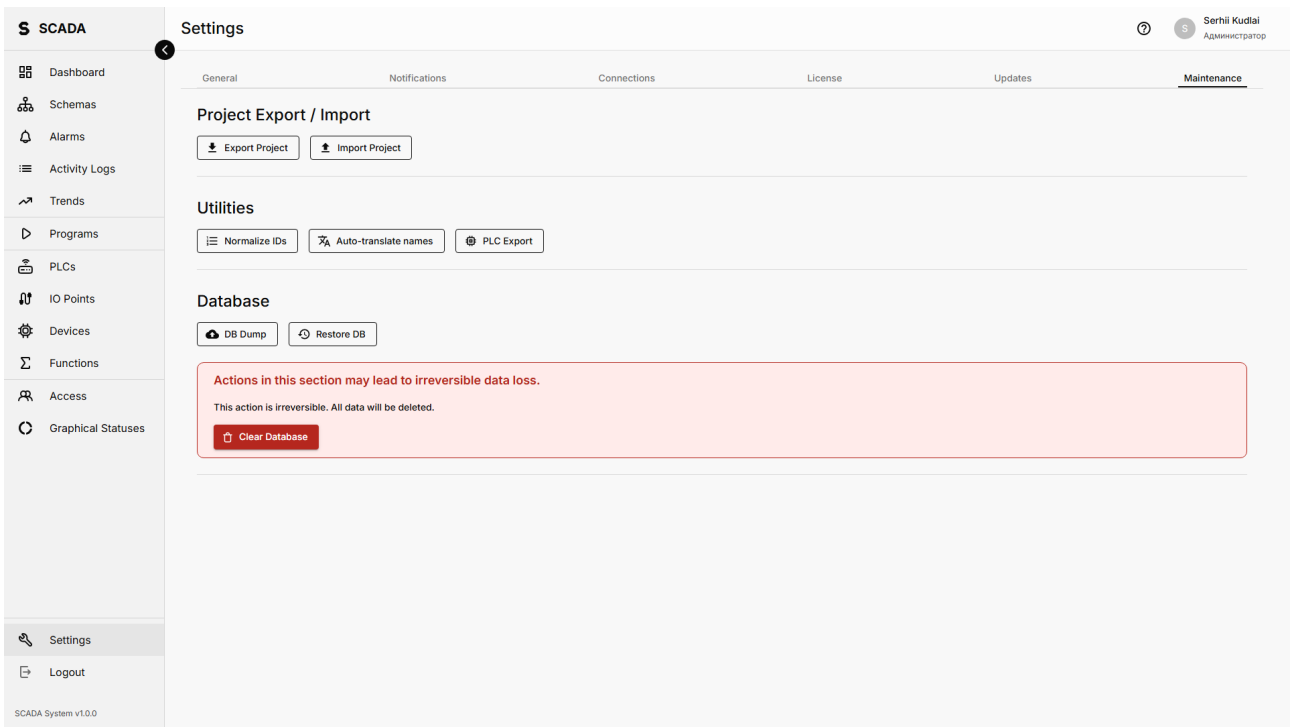
连接选项卡：TIA Agent 列表，包含主机/端口、添加/编辑/删除/测试按钮、状态指示器。



许可证选项卡：许可证状态（绿色徽章）、类型、剩余天数、公司、到期日期。



更新选项卡：当前版本、检查更新按钮、进度条、更新日志。



维护选项卡：备份数据库、恢复数据库、清除缓存、磁盘清理、冒烟测试按钮及说明。

## 技术文档

### 用户设置API:

方法	端点	描述
GET	/api/user/language/	获取语言
POST	/api/user/language/	设置语言

方法	端点	描述
GET	/api/user/dark-mode/	获取深色模式
POST	/api/user/dark-mode/	设置深色模式

## 许可证API:

方法	端点	描述
GET	/api/license/status/	许可证状态
POST	/api/license/activate/	激活
POST	/api/license/generate/	生成密钥

## 更新API:

方法	端点	描述
GET	/api/system/update-status/	更新状态
POST	/api/system/update-check/	检查更新
POST	/api/system/update-apply/	应用更新
GET	/api/system/update-progress/	进展
POST	/api/system/update-cancel/	取消
POST	/api/system/update-rollback/	回滚
GET	/api/system/update-history/	更新历史
GET	/api/system/update-backups/	备份
POST	/api/system/update-backups/{filename}/restore/	从备份恢
POST	/api/system/disk-cleanup/	磁盘清理
GET	/api/system/smoke-test/	烟雾测试
GET	/api/system/diagnostics/	诊断下载

## WebSocket: 更新通知

```
{
  "type": "system_update",
  "event": "update_progress",
  "progress_percent": 45,
  "current_step": "Applying migrations...",
  "status": "in_progress"
}
```

## 26. 人工智能助手

### 对于用户

内置 AI 助手，可让您通过自然语言管理 SCADA。

要素：

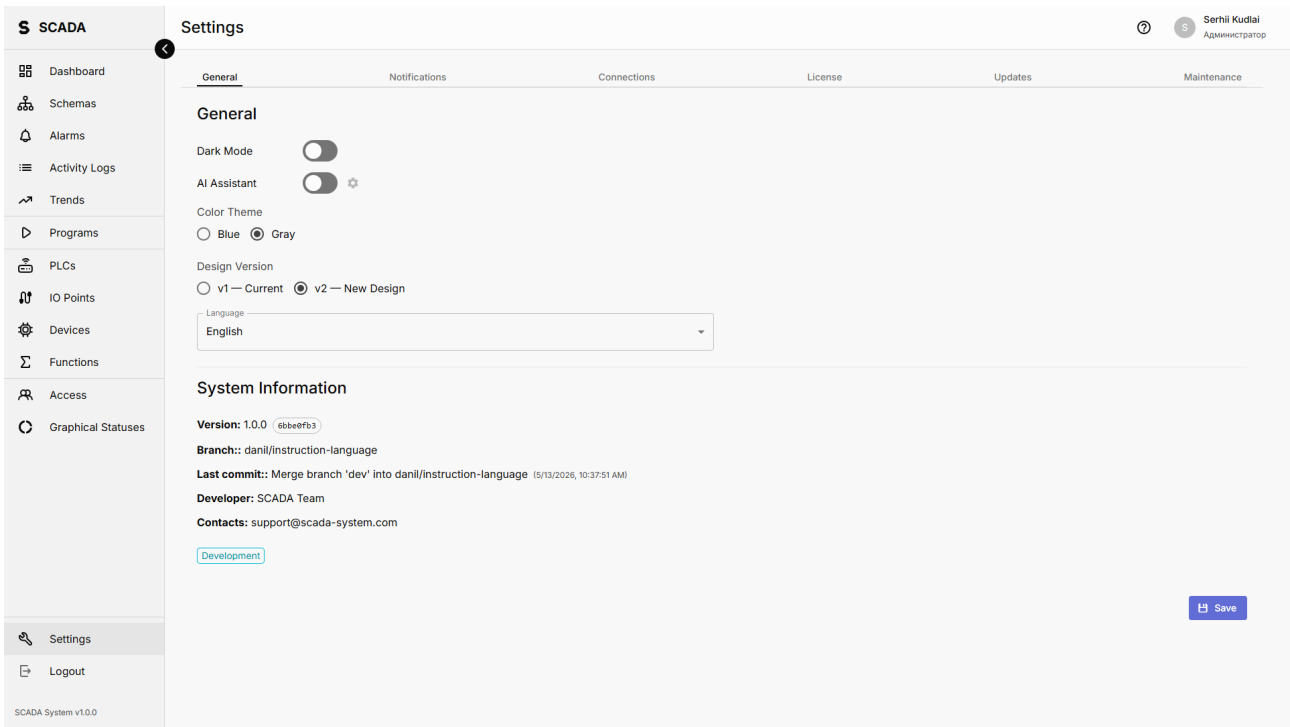
元素	描述
AI按钮 (FAB)	右下角浮动按钮
聊天面板	滑出/可拖动面板
会议列表	带有对话历史记录的面板
输入字段	消息文本字段
发送按钮	发送消息
附加按钮	上传图像 (P&ID 绘图)
设置	AI 配置 (型号、API 密钥)

人工智能能力：

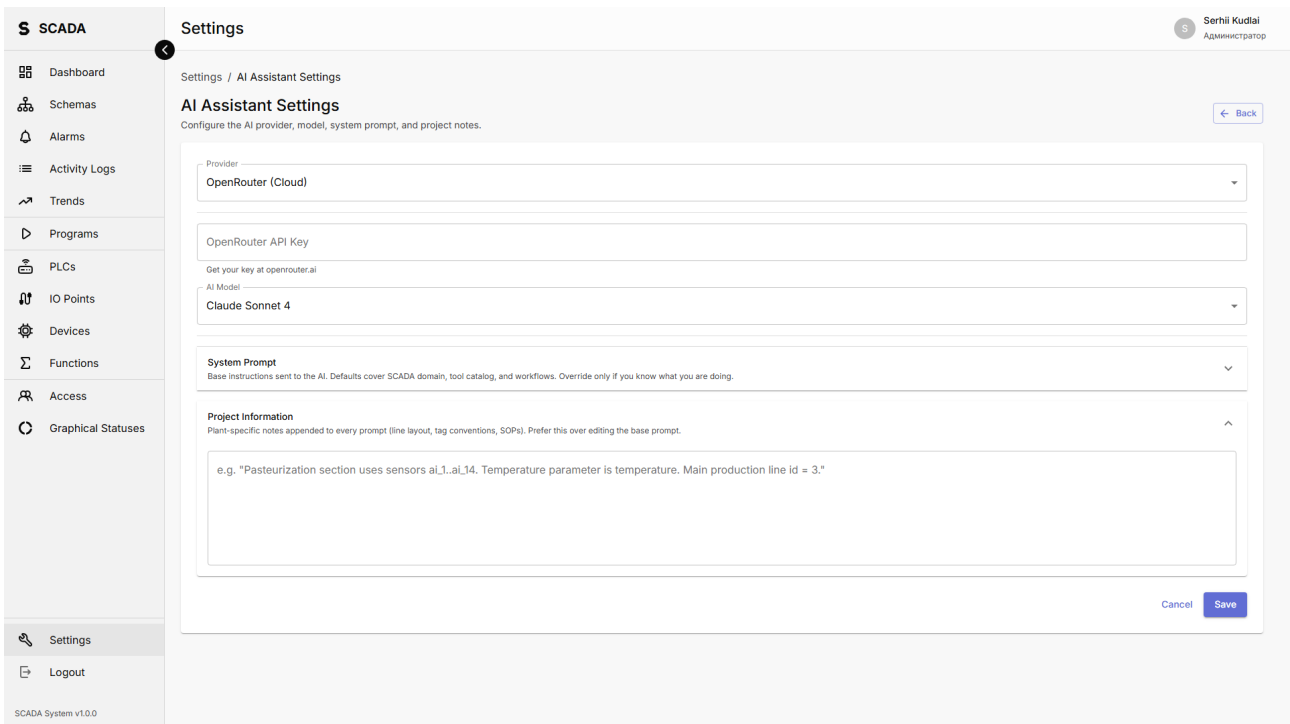
功能	描述
设备控制	«打开电机 MOT_001»
信息查询	“哪些设备当前处于警报状态？”
元素创建	«创建新的模拟输入 AI_010»
P&ID 分析	上传图片→元素识别→放置建议
帮助	系统问答

人工智能设置 (AiSettingsDialog) :

领域	描述
API 密钥	OpenRouter 密钥 (使用 AES/Fernet 加密)
型号	LLM模型选择器
温度	回应创造力 (0-1)
最大代币数量	最大响应长度



右下角浮动AI按钮：圆形，带有机器人/聊天图标。



AI 聊天面板：左侧 — 会话列表，右侧 — 对话（用户和 AI 消息），底部 — 使用“发送”和“附加”按钮输入。

## 技术文档

AI API (FastAPI微服务, BaseURL: `ai-api/v1`) :

方法	端点	描述
POST	/会话	创建会话
GET	/会话	列出会话

方法	端点	描述
DELETE	/sessions/{id}	删除会话
PATCH	/sessions/{id}	重命名
GET	/sessions/{id}/messages	会话消息
POST	/聊天/{id}/消息	发送消息 (SSE 流)
POST	/图像/分析	分析图像 (多部分)
POST	/图像/应用	申请提案
GET	/设置	获取设置
PATCH	/设置	更新设置
GET	/健康	健康检查

SSE 事件 (stream=true) :

活动	描述
令牌	响应文本的标记 (流式传输)
工具调用	AI调用工具 (devicetools、schematools、query_tools)
工具结果	工具调用结果
错误	错误
完成	回复完毕

SSE 格式:

```

event: token
data: {"content": "Motor "}

event: token
data: {"content": "MOT_001 "}

event: tool_call
data: {"name": "get_device_status", "arguments": {"device_id": "mot_001"}}

event: tool_result
data: {"result": {"status": "running", "value": 1}}

event: token
data: {"content": "is running now."}

event: done
data: {}
    
```

## 27. PLC工程导出

### 对于用户

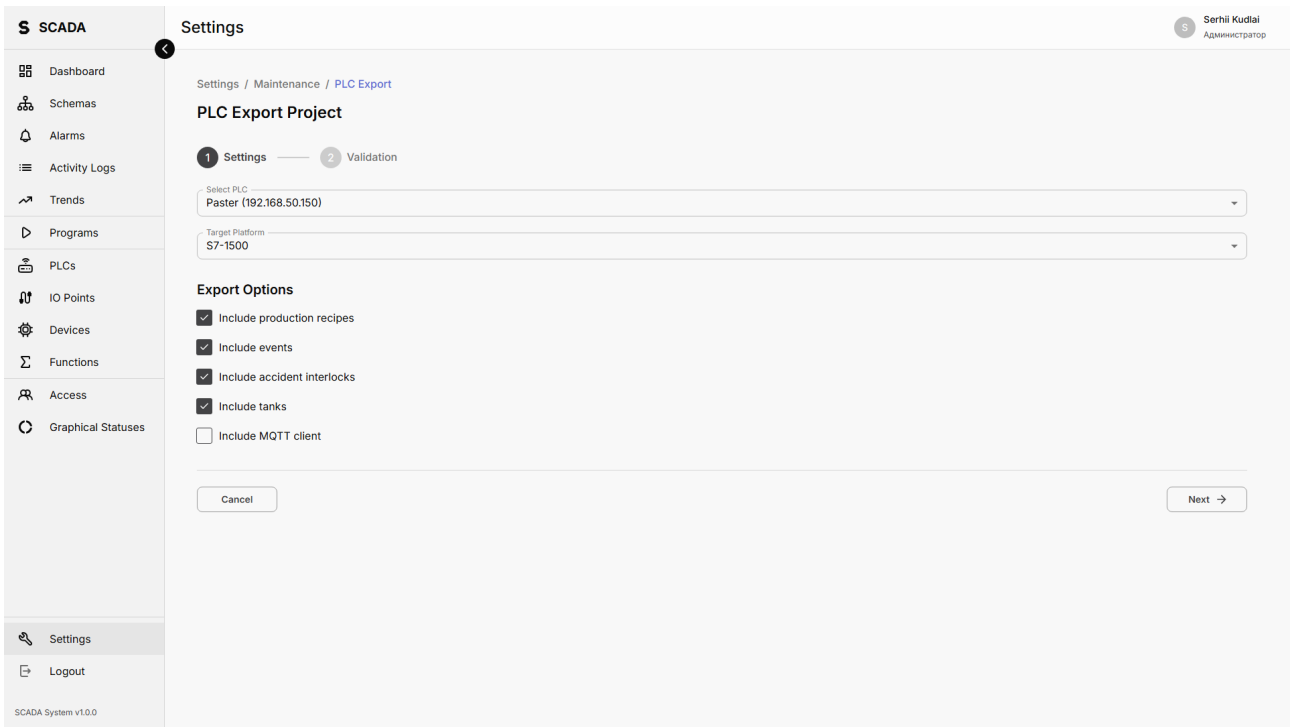
从 SCADA 配置生成西门子 TIA Portal 项目。

导出对话框 (PlcExportDialog) :

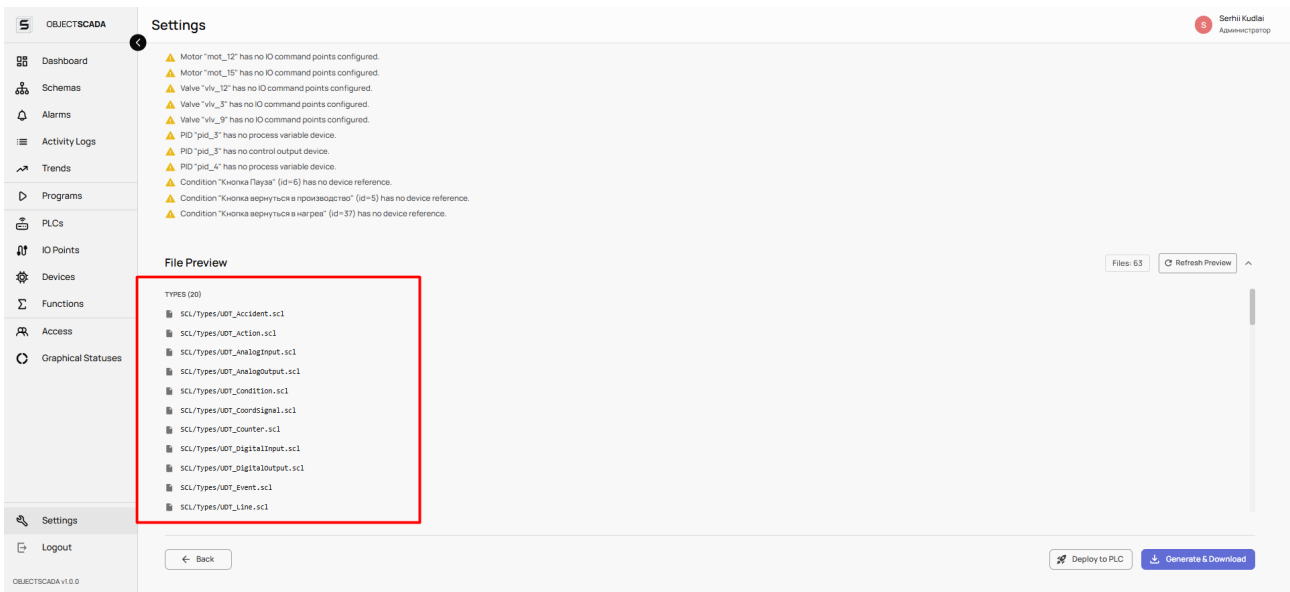
元素	描述
PLC选择器	目标控制器
PLC类型	S7-1200 / S7-1500
包括食谱	复选框-添加食谱
包括活动	复选框 — 添加事件
包括事故	复选框 — 添加事故
包括坦克	复选框 — 添加坦克
包括 MQTT	复选框 — 添加 MQTT 通信
MQTT 代理 IP	MQTT 代理地址
MQTT 主题前缀	主题前缀
预览	SCL 代码预览
生成	下载项目 ZIP
部署到 TIA	通过 TIA Agent 进行部署

生成的项目内容:

- 功能块的 SCL 代码 (电机 FB、阀门 FB、PID FB、AI FB、AO FB、DI FB、DO FB、COS FB、计数器 FB、定时器 FB、水箱 FB)
- 序列FB (状态机)
- 事件/事故 FB
- 数据块 (实例 DB、配方 DB)
- OB1 Main (调用每个 FB)
- XML 标签表 (SimaticML)



导出对话框：PLC 下拉列表、S7-1200/S7-1500 无线电、包括复选框（配方、事件、事故、坦克、MQTT）、MQTT 配置字段、预览/生成/部署按钮。



SCL 代码预览：左 — 文件树（FB、DB、标签表），右 — 带语法突出显示的 SCL 文件内容。

## 技术文档

### PLC 导出 API:

方法	端点	描述
GET	/api/plc-export/preview/	代码预览
POST	/api/plc-export/validate/	验证
POST	/api/plc-export/generate/	生成 ZIP
GET	/api/plc-export/address-map/	地址地图

## 28. 项目导出和导入

### 对于用户

完整的 SCADA 项目导出/导入，以便在安装之间进行迁移。

导出对话框 (ExportProjectDialog) :

元素	描述
对象复选框	选择要导出的类型：设备、模式、序列、配方、事件、事故、选项、趋势、警报、用户、角色
选择全部	全选
包括图表	导出趋势数据
搜索	对象过滤器
出口	下载导出文件
进展	进度指示器

按钮：

按钮	描述
出口项目	开始导出选定的对象
导入项目	上传文件→预览→应用
数据库备份	创建数据库备份
恢复数据库	从备份恢复

The screenshot displays the 'Settings / Maintenance / Export Project' configuration page. It features a sidebar on the left with navigation icons for various system components. The main area is titled 'Export Project' and contains a checkbox for 'Include graph data points' which is checked. Below this, there is a section 'Select objects to export' with a 'Selected: 195' indicator and a 'Deselect all' button. A list of object types is shown, each with a selection count and a dropdown arrow:

- Schemas: 3/3
- Sequences: 4/4
- Motors: 21/21
- Valves: 15/15
- PID controllers: 11/11
- Variable speed drives: 0/0
- Counters: 3/3
- Timers: 0/0
- Analog inputs: 17/17
- Analog outputs: 9/9
- Digital inputs: 4/4
- Digital outputs: 1/1
- Coordination signals: 3/3

导出对话框：带有复选框的类型列表、全选按钮、“包含图表”切换、导出按钮、进度条。

## 技术文档

项目API:

方法	端点	描述
GET	/api/project/export-models/	列出可出口型号
POST	/api/project/export/	出口
POST	/api/project/import/preview/	导入预览
POST	/api/project/import/	进口
POST	/api/project/backup-db/	数据库备份
POST	/api/project/restore-db/	恢复
POST	/api/project/normalize-ids/	规范化 ID
POST	/api/project/clear-database/	清除数据库
POST	/api/project/export-graph-data/	导出图表
POST	/api/project/import-graph-data/	导入图表
GET	/api/project/graph-data-stats/	图表统计

## 29. 坦克

### 对于用户

储罐是根据传感器数据自动计算体积的容器。

罐体参数：

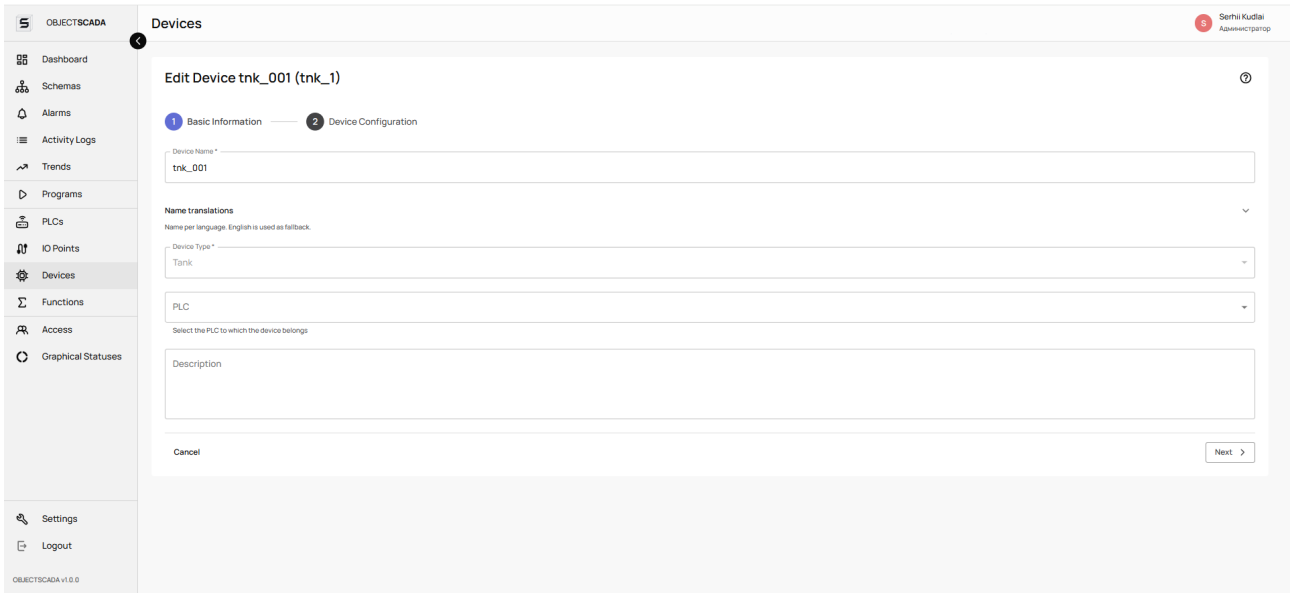
领域	描述
液位传感器	绑定到 AI 设备（液位传感器）
顶部压力传感器	顶部压力（可选）
底部压力传感器	底部压力（ $\Delta P$ ）
警报级别	HH、H、L、LL — 警报阈值

体积计算方法：

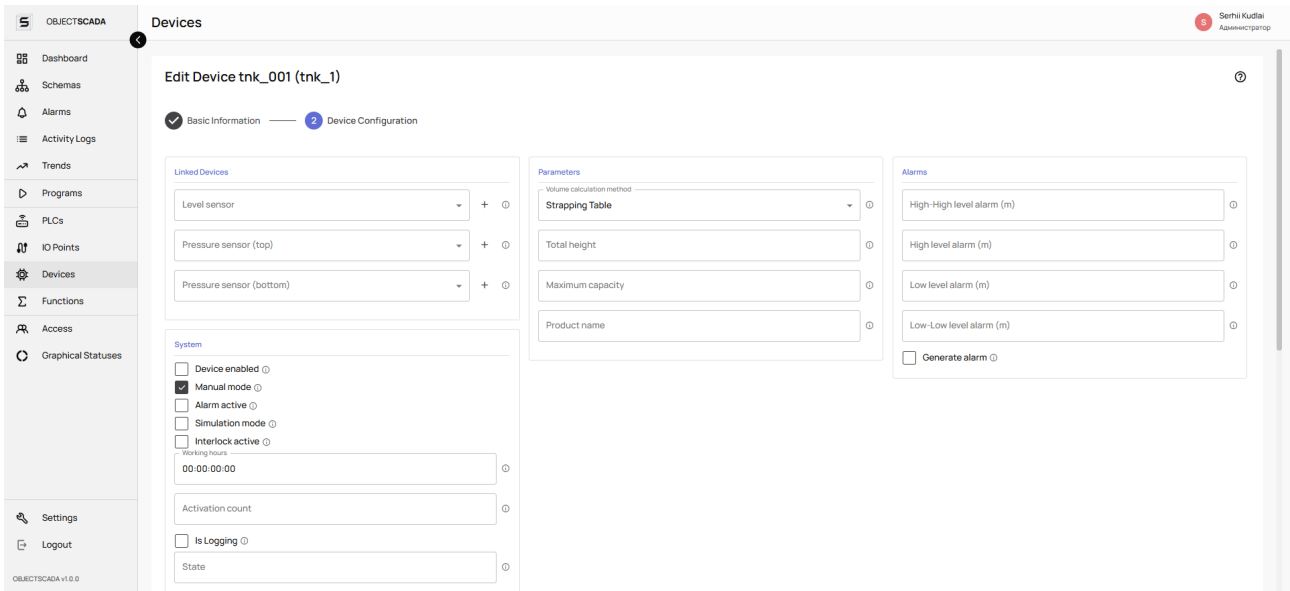
方法	描述
捆扎表	校准表（插值传感器值 $\rightarrow$ 体积）
几何	几何（截面：圆柱体、圆锥体、圆顶、碟形）

架构上的储罐构造函数：

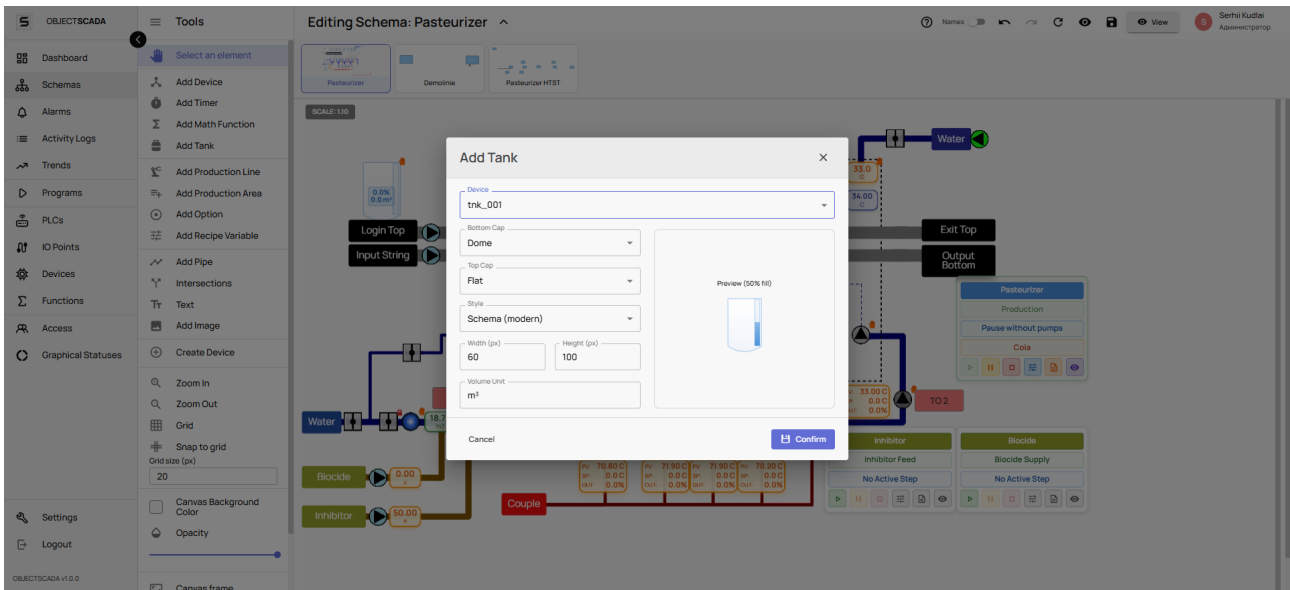
元素	描述
设备选择器	现有 TNK 设备的下拉菜单
底盖类型	平面/圆顶/碟形/锥体
顶盖类型	平面/圆顶/碟形
宽度/高度	视觉尺寸
SVG 预览	坦克形状预览
自动放置传感器	传感器沿着右墙放置



罐体形式：传感器绑定（液位传感器下拉菜单、压力顶部、压力底部）、警报阈值（HH、H、L、LL — 数字字段）。



捆扎表：两列 — 传感器值和音量、添加行、删除行、导入 CSV、导出 CSV 按钮。



坦克构造器：左 — 设置（设备、盖子、尺寸），右 — 所选形状的 SVG 预览。

## 技术文档

### 坦克API:

方法	端点	描述
GET	/api/tanks/	列出坦克
POST	/api/tanks/	创建
PUT	/api/tanks/{id}/	更新
DELETE	/api/tanks/{id}/	删除
POST	/api/tanks/{id}/calculate_volume/	重新计算体积
POST	/api/tanks/{id}/import_strapping_table/	导入表 (CSV)
GET	/api/tanks/{id}/export_strapping_table/	导出表
GET	/api/tanks/{id}/volume_at_level/	音量水平 (?sensor_value=X)
GET	/api/tank-sections/	部分 (?tank=ID)
GET	/api/tank-strapping-entries/	表行 (?tank=ID)

### Rust Worker — 定期计算（每 1000 毫秒）：

文件 `scada-worker/src/tasks/tank_volume.rs`

轮询液位和压力传感器，根据捆扎表或几何形状计算体积，将结果写回数据库并通过 Redis 发布。

## 应用程序导航（侧边栏）

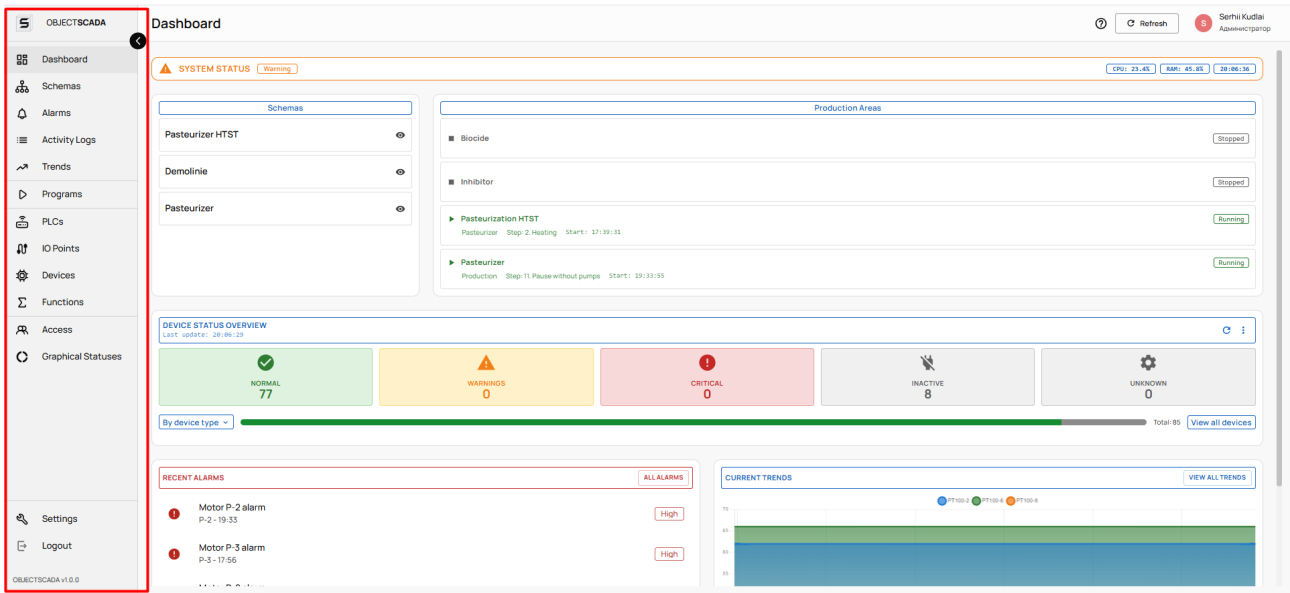
### 对于用户

每个页面都可以使用侧边栏。菜单项的显示或隐藏取决于用户的权限。

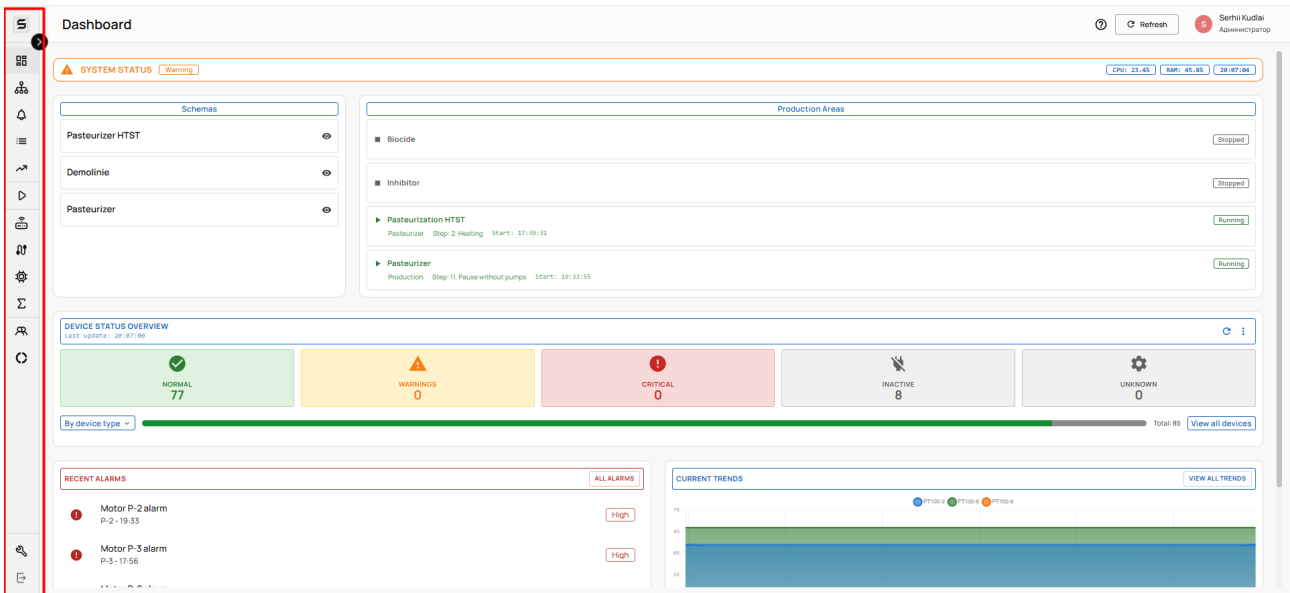
菜单项	页	所需许可
仪表盘	主面板	—（始终可用）
架构	助记符模式	—
警报	警报	—
活动日志	活动日志	—
趋势	趋势	—
序列	序列	序列食谱
PLC	PLC	plc
I/O 点	I/O点数	io_points
设备	设备	设备
功能	数学函数	设备
用户	用户	用户
角色	角色	角色
图形状态	图形状态	图形状态
设置	设置	—（始终可用）
退出	退出	—

附加侧边栏元素：

元素	描述
用户头像	照片和姓名
折叠/展开	将侧边栏缩小为图标
版本号	当前应用程序版本



侧边栏展开：用户头像位于顶部，菜单项带有图标和标签，注销位于底部。



侧边栏折叠：只有菜单图标。

# 全局 UI 元素

## 对于用户

标题:

元素	描述
页面标题	当前部分名称
帮助按钮	推出导赏团
更新指标	更新可用时的徽章
连接指示灯	WebSocket/MQTT 连接状态
情境行动	特定于当前页面的按钮

小吃店:

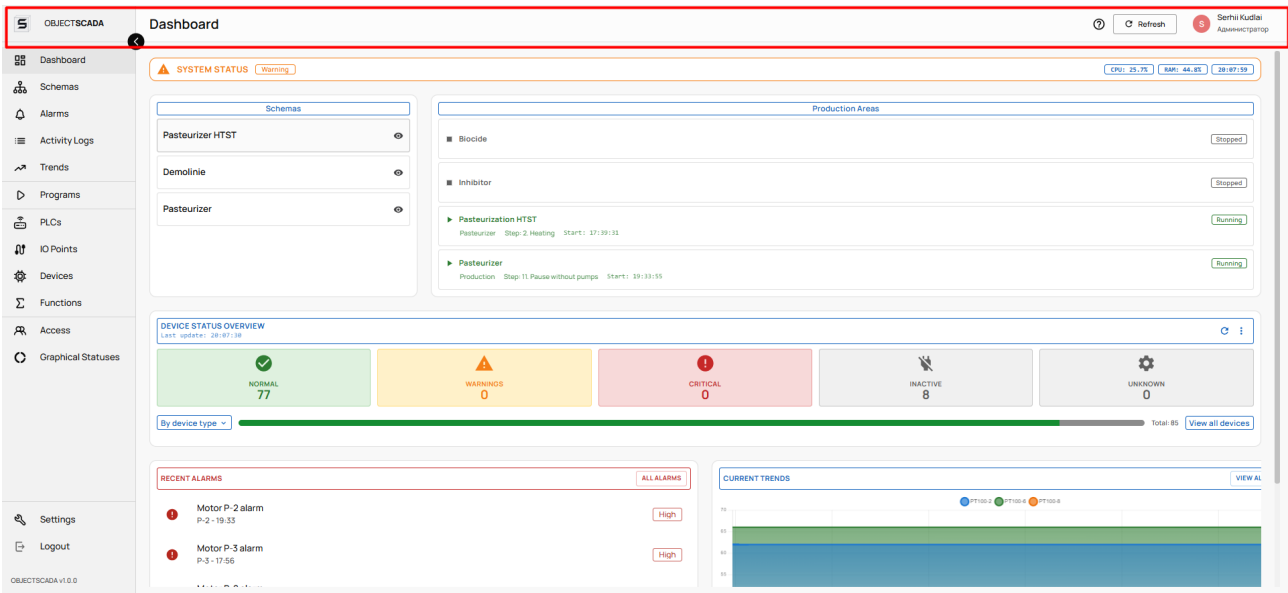
类型	描述
成功 (绿色)	操作成功
错误 (红色)	错误
警告 (黄色)	警告
信息 (蓝色)	信息

横幅:

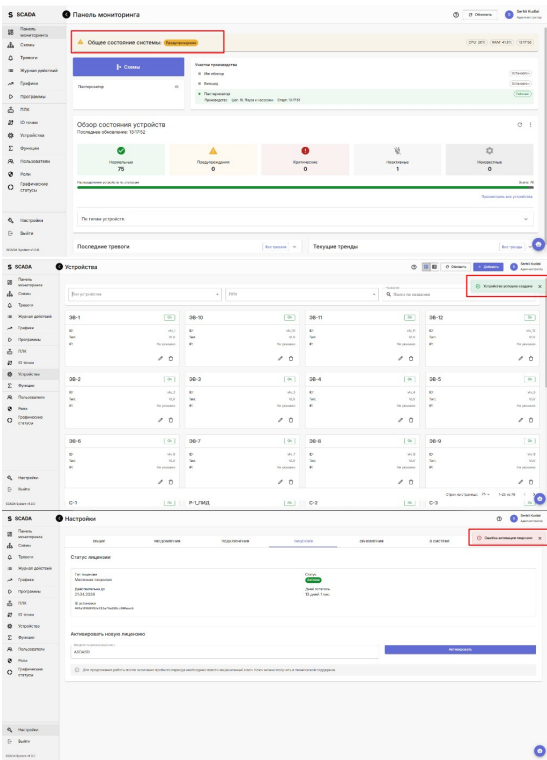
横幅	描述
连接状态	连接丢失时出现黄色横幅 (15 秒后)
保养	维护模式开启时的蓝色横幅
版本不匹配	前端/后端版本不同时的橙色横幅
可用更新	当有更新可用时, 标题中会显示绿色徽章

导游服务:

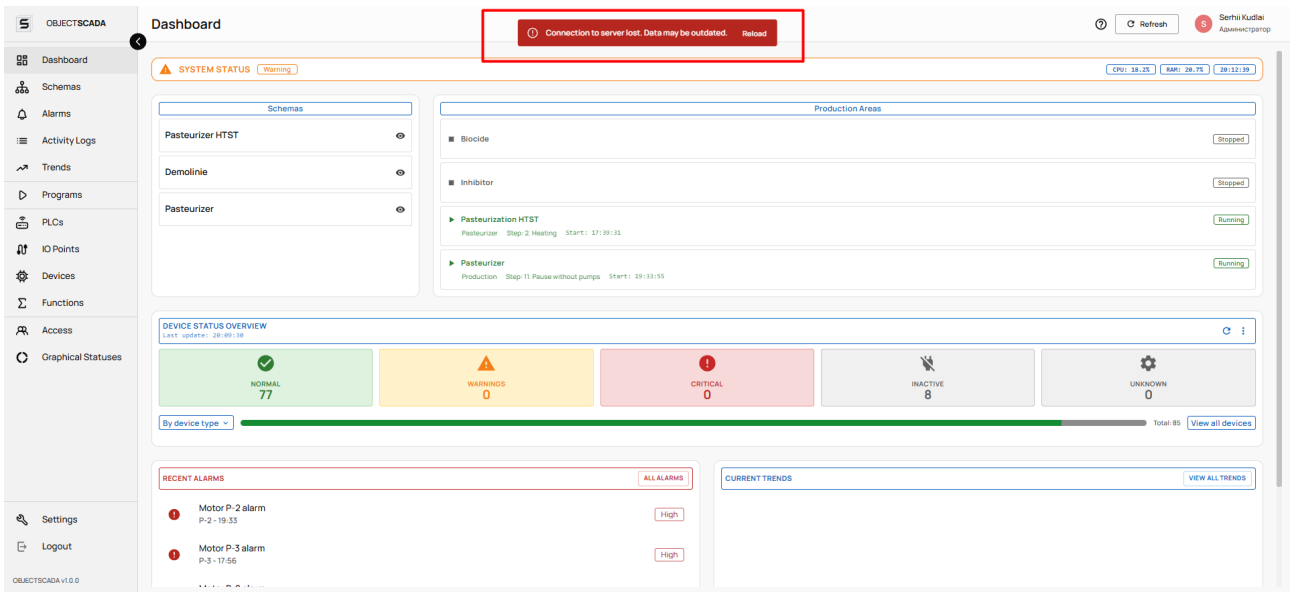
通过 “?” 可在每个页面上找到标题中的按钮。带有元素突出显示和描述的分步交互式指导。



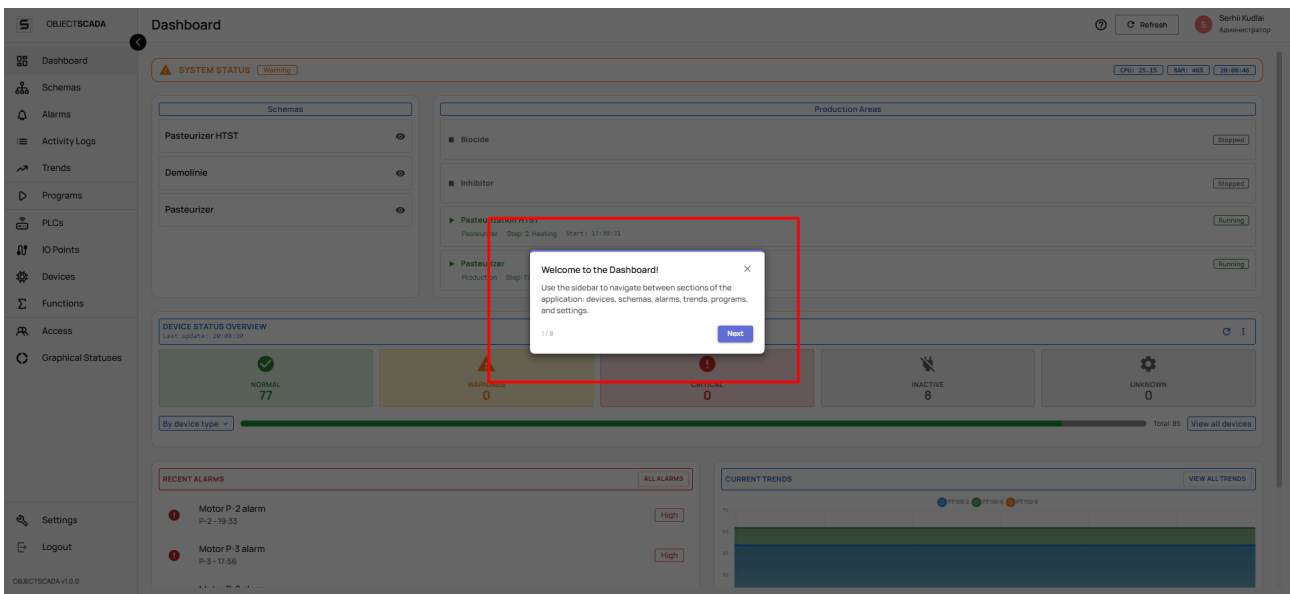
应用程序标题：左侧页面标题、游览按钮 ( )、更新指示器 (红色徽章)、连接状态 (绿色/红点)。



示例通知：绿色“设备已成功保存”、红色“错误：连接丢失”、黄色“警告：许可证将在 7 天内过期”。



黄色顶部横幅: «连接丢失。正在尝试用动画重新连接...»。



引导浏览: 带有工具提示的突出显示元素、“下一步”、“后退”、“跳过”按钮、进度条。

## 键盘快捷键

关键	行动	工作地点
Ctrl+Z	撤消	架构生成器
Ctrl+Y	重做	架构生成器
Ctrl+S	保存	架构生成器
删除	删除元素	架构生成器
Ctrl+A	全选	架构生成器
Ctrl+C	复制	架构生成器
Ctrl+V	粘贴	架构生成器
'逃脱'	清除选择/关闭对话框	无处不在
鼠标滚轮	放大	架构生成器/查看器、趋势

## 实际例子

本节包含端到端自动化场景，展示设备、助记符模式、序列、配方和事件如何组合在一起以实现实际流程。

### 示例 1. 自动化牛奶巴氏灭菌器

过程目标。将牛奶加热至 75 °C，保持 15 秒，然后冷却至 4 °C。这是经典的 HTST（高温短时）工艺，广泛应用于乳制品行业。

工艺设备：

设备	身份证	类型	目的
供应泵	mot_001	电机	将牛奶推过热交换器
蒸汽进口阀	val_001	阀门	向加热部分供应蒸汽
冷水机阀门	val_002	阀门	供应冷冻水
加热温度传感器	ai_001	模拟输入	加热段后温度
保持温度传感器	ai_002	模拟输入	保温管后温度
出口温度传感器	ai_003	模拟输入	冷水机后温度
加热控制器	pid_001	PID控制器	保持加热温度（设定值 → val_001）
冷却控制器	pid_002	PID控制器	保持出口温度（设定值 → val_002）
接收罐液位	tnk_001	坦克	储罐填充监控
三通换向阀	val_003	阀门	返回未煮熟的牛奶以重新加热

### 第 1 阶段. 设备创建

通过 Devices → «+» 创建所有设备。对于 PID 控制器 “pid\_001”：

```
POST /api/pid-controllers/
{
  "id": "pid_001",
  "name": "Heating controller",
  "plc": 1,
  "setpoint": 75.0,
  "kp": 2.5,
  "ki": 0.1,
  "kd": 0.05,
  "output_min": 0,
  "output_max": 100
}
```

```
POST /api/tanks/
{
  "id": "tnk_001",
  "name": "Receiving tank",
  "plc": 1,
  "volume_method": "STRAPPING",
  "level_sensor": "ai_004",
  "alarm_hh": 95.0,
  "alarm_h": 85.0,
  "alarm_l": 15.0,
  "alarm_ll": 5.0
}
```

## 第 2 阶段. 序列

整个巴氏灭菌循环分为 5 个步骤:

#	步骤	目的	下一步的转移条件
1	空闲	等待操作员启动	选项 “开始”
2	供暖	加热至设定值	ai_001 >= 75.0 持续 5 秒
3	持有	保持温度	定时器 15 秒
4	冷却	冷却至 4 °C	ai_003 <= 4.0
5	完成	完成	返回 “空闲”

通过 API 创建序列:

```
POST /api/sequences/
{
  "name": "Pasteurization HTST",
  "name_translations": {"en": "Pasteurization HTST", "ru": "Пастеризация HTST"},
  "description": "Heat 75°C → hold 15 s → cool 4°C"
}
```

添加步骤:

```
POST /api/sequence-steps/
{
  "sequence": 1,
  "order": 2,
  "name": "Heating",
  "name_translations": {"en": "Heating"},
  "next_steps": {"3": [10]}
}
```

过渡条件《温度 ≥ 75°C》:

```
POST /api/transition-conditions/
{
  "id": 10,
  "type": "device",
  "device_id": "ai_001",
  "content_type": "analoginput",
  "operator": ">=",
  "value": 75.0,
  "duration_seconds": 5
}
```

## 第三阶段. 生产配方

我们将与产品相关的参数提升为配方变量，以便相同的序列可以运行不同的乳制品（全脂奶、奶油、脱脂奶）。

配方变量:

```
POST /api/production-recipe-variables/
{
  "name": "target_heating_temp",
  "name_translations": {"en": "Target heating temperature"},
  "sequence": 1,
  "default_value": 75.0,
  "min_value": 70.0,
  "max_value": 90.0
}
```

```
POST /api/production-recipe-variables/
{
  "name": "holding_time_sec",
  "name_translations": {"en": "Holding time, seconds"},
  "sequence": 1,
  "default_value": 15.0,
  "min_value": 10.0,
  "max_value": 30.0
}
```

### 全脂牛奶配方：

```
POST /api/production-recipes/
{
  "name": "Whole Milk 3.5%",
  "sequence": 1,
  "variables": {
    "target_heating_temp": 75.0,
    "holding_time_sec": 15.0
  }
}
```

### 奶油配方（更严格的方案）：

```
POST /api/production-recipes/
{
  "name": "Cream 30%",
  "sequence": 1,
  "variables": {
    "target_heating_temp": 85.0,
    "holding_time_sec": 20.0
  }
}
```

## 第 4 阶段：事故事件

### 定义两种严重事故：

#### 1. 过热（高于 90 °C）：

```
POST /api/accident-events/
{
  "name": "Overheat protection",
  "device_id": "ai_001",
  "content_type": "analoginput",
  "operator": ">",
  "value": 90.0,
  "severity_level": 4,
  "action": "STOP_SEQUENCE"
}
```

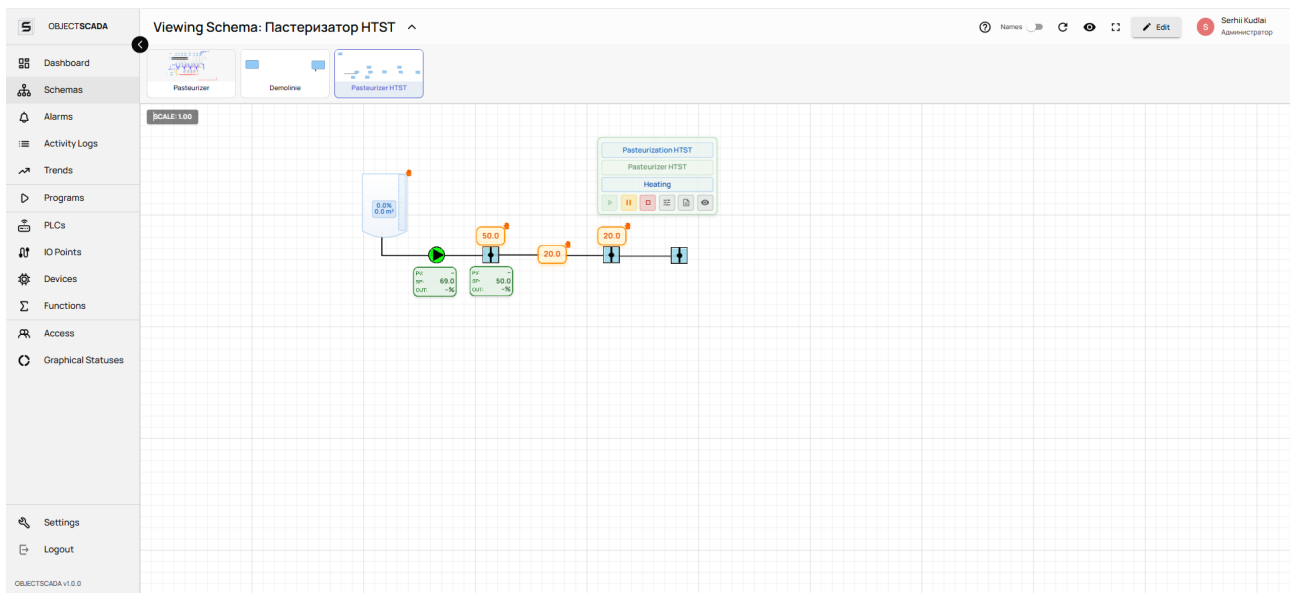
severity\_level: 4` 为严重 — 序列立即停止。

2. 巴氏灭菌不足（保存期间低于 73 °C）：

```
POST /api/accident-events/
{
  "name": "Insufficient pasteurization",
  "device_id": "ai_002",
  "content_type": "analoginput",
  "operator": "<",
  "value": 73.0,
  "severity_level": 3,
  "action": "TRIGGER_DIVERT"
}
```

## 阶段 5. 助记符模式

- 创建一个模式 “巴氏灭菌器 HTST”。
- 在架构生成器中：
- 通过 Tank 接收坦克 tnk\_001 — 在构造函数中选择形状 FLAT / DOME 和尺寸 80×120。
- 泵 “mot001”、阀门 “val001”、“val002”、“val003”、PID “pid001”、“pid002”。
- 传感器 “ai001”、“ai002”、“ai\_003”。
- 连接管道：水箱→泵→加热器 (val\_001和ai\_001) →保温管 (ai\_002) →冷水机 (val\_002、ai\_003) →出口。
- 绑定序列：序列 → 选择 “巴氏灭菌 HTST”。
- 添加选项按钮 “开始”、“暂停”、“停止”（单独的选项块）。



巴氏灭菌器助记图：左侧接收罐及其液位，一条管道通向泵，然后穿过带有温度传感器和蒸汽阀的加热部分、保温管和带有冷冻水阀的冷却部分。右侧 — 突出显示活动 “加热” 步骤的序列块。

## 第 6 阶段. 运行流程

- 在架构查看器中打开架构。
- 在序列块上选择活动配方 - 例如 “全脂牛奶 3.5%”。
- 按 开始 — 序列从 “空闲” 移动到 “加热”。
- 现场观看过程：
- 活动步骤突出显示（黄色 → 绿色表示成功转换）。
- 传感器上的当前温度在弹出窗口中可见。
- val\_001 的打开百分比（由 PID 驱动）。

- 一旦达到 75 °C 并保持 5 秒，就会自动过渡到“保持”。
- 15 秒后 — 过渡到“冷却”。
- 当出口温度  $\leq 4^{\circ}\text{C}$  → “完成” → 返回“空闲”。

## 第 7 阶段. 批次后分析

- 打开趋势 → 创建图表。
- 添加趋势线: ai\_001、ai\_002、ai\_003、pid\_001.output。
- 将范围设置为最近 30 分钟。
- 检查:
  - 加热曲线的形状 (达到设定点的时间)。
  - 保持温度的稳定性 (预期波动  $\leq 0.5^{\circ}\text{C}$ )。
  - 冷却速度。
- 将图表导出为 PNG 或 CSV 格式以进行报告。

## 第 8 阶段: 生成 PLC 项目

- 打开 仪表板 → PLC 导出 按钮 (或设置 → PLC 导出)。
- 在对话框中选择:
  - PLC: PLC\_DEMO
  - 装置、顺序、配方和事故——全部。
- 按预览 — 系统列出要生成的文件:
  - Motor\_FB.scl、Valve\_FB.scl、PID\_FB.scl — 设备功能块
  - Sequence\_Pasteurization\_FB.scl — 状态机
  - Accident\_FB.scl — 事故处理程序
  - OB1\_Main.scl — 主程序
  - Tags\_Inputs.xml、Tags\_Outputs.xml — TIA Portal 的标签表
  - 验证 — 验证配置的正确性。
  - 生成 — 生成用于 TIA Portal 导入的 ZIP。

## 示例 2. 采用批次管理的 CIP 清洗

工艺目标。

生产班次后自动就地清洁加工设备 (储罐、管道)。多阶段: 预漂洗→碱洗→中间漂洗→酸洗→最终漂洗。

此示例的显着特点:

- 使用批次管理 — 每个清洁周期都记录为一个批次。
- 使用带阶段的生产线。
- 操作员可以通过选项跳过各个阶段 (kurzes Reinigen)。
- 演示计时器和计数器。

### 第一阶段. 装备

身份证	类型	目的
tnk_cip_01	坦克	碱溶液罐 (NaOH 2%)
tnk_cip_02	坦克	酸溶液罐 (HNO 1.5%)
tnk_water	坦克	循环水箱
mot_cip_pump	电机	CIP循环泵
val_cip_supply	阀门	供应阀进入洗涤回路
val_cip_return	阀门	回流阀

身份证	类型	目的
val_alkali	阀门	供碱阀
val_acid	阀门	供酸阀
val_water	阀门	供水阀门
ai_cip_temp	模拟输入	溶液温度
ai_cip_cond	模拟输入	电导率 (浓度检查)
cnt_cip_flow	专柜	供应流量计
tim_cip_stage	定时器	当前阶段定时器

## 第 2 阶段. 生产线和阶段

创建一条具有五个阶段的生产线 - 每个阶段都有自己的顺序。

```
POST /api/production-lines/
{
  "name": "CIP Line 1",
  "description": "Cleaning of production 1 tanks"
}
```

阶段 (生产线阶段) :

#	名称	序列	批量传输动作
1	预冲洗	seq_cip_rinse (水, 90 秒)	转移到下一阶段
2	碱洗	seq_cip_alkali (NaOH, 75 °C 20 分钟)	转让
3	中间冲洗	seq_cip_rinse	转让
4	酸洗	seq_cip_acid (HNO, 65 °C 下 15 分钟)	转让
5	最后冲洗	seq_cip_rinse	完整

每个序列最后一步的 “batchtransferaction” 字段控制转换:

- TRANSFER — 批次自动移至下一阶段。
- COMPLETE — 批次已完成。
- NONE — 批次保持在当前阶段 (手动控制)。

## 第 3 阶段. “碱洗” 顺序

步骤:

- Fill — 用碱填充回路 (val\_alkali = OPEN、mot\_cip\_pump = ON)。
- Heat — 加热至 75 °C (ai\_cip\_temp >= 75.0)。
- Circulate — 循环 20 分钟 (计时器 tim\_cip\_stage)。
- Drain — 排水 (val\_cip\_return = OPEN 至下水道)。

配方《碱洗 — 标准》:

```
POST /api/production-recipes/
{
  "name": "Alkali wash - standard",
  "production_line": 1,
  "variables": {
    "alkali_temp": 75.0,
    "alkali_duration_min": 20,
    "acid_temp": 65.0,
  }
}
```

```
    "acid_duration_min": 15  
  }  
}
```

## 第 4 阶段. 操作员选项

添加用于加速或调整过程的选项按钮：

选项	描述	效果
SKIP_ACID	跳过酸洗	跳过第 4 阶段（直接跳至第 5 阶段）
双重冲洗	双重最终冲洗	第 5 阶段重复两次
低温	温和模式（较低温度）	变量 alkali_temp = 60, acid_temp = 50

选项通过 序列选项 添加并作为按钮放置在架构上。

## 阶段 5. 创建并启动批处理

- 在架构查看器中打开 CIP 线路架构。
- 在生产线 元素上按 创建批次 — 将打开 “CreateBatchDialog” 。

3、填写：

- 批次名称：CIP-2026-04-18-001
- 配方：《碱洗 — 标准》
- 阶段：预冲洗（初始阶段）
- 优先级：正常
- 创建 — 批次加入队列。
- 从 “BatchManagementPanel” 激活批次 → 开始。

等效API：

```
POST /api/production-lines/1/create_batch/
{
  "name": "CIP-2026-04-18-001",
  "recipe_id": 5,
  "starting_stage_id": 1,
  "priority": "NORMAL"
}
```

## 第 6 阶段. 监控批次

“BatchManagementPanel” 显示：

- 当前阶段（例如 “碱洗” ）。
- 阶段内进度（倒计时器）。
- 主要传感器的当前读数（温度、电导率、流量）。
- 所有阶段的总体批次进度。

阶段转换存储在 “BatchStageHistory” 中——稍后可以检索完整的时间线：

```
GET /api/batches/{id}/history/
```

## 第 7 阶段. 成功的洗涤验证

HACCP/ISO 22000 需要清洁证明文件。使用反向事故事件：

```
POST /api/events/
{
  "name": "CIP validation - conductivity check",
  "device_id": "ai_cip_cond",
  "operator": "<",
  "value": 0.5,
  "duration_seconds": 30,
  "action": "LOG_EVENT"
}
```

该日志可在活动日志中使用，并具有序列/批量过滤功能，并且可以导出为 CSV 格式以进行报告。

## 新流程设置清单

配置任何新流程时使用此模板：

### 1.设计（进入系统前）：

- 绘制过程 P&ID 图
- 已准备好具有唯一 ID 的设备列表
- 所有模拟和数字信号均已定义
- 记录操作逻辑（步骤和转换条件）
- 定义警报阈值和操作

### 2.在 SCADA 中配置：

- 使用正确地址创建的 PLC
- 创建并连接到 PLC 的所有设备
- 配置 PID 参数 (Kp、Ki、Kd)
- 配置储罐捆扎表（如适用）
- 使用放置的设备和管道创建的助记符模式
- 使用步骤创建的序列
- 创建转换条件
- 配方变量已创建
- 至少创建一个具有默认值的配方
- 事故事件配置了适当的严重级别

### 3.测试（投入生产之前）：

- 每个设备都可以手动控制（手动模式）
- 转换条件正确触发（使用 POST /api/transition-conditions/{id}/simulate/）
- 顺序在手动中端到端地传递所有步骤
- 意外发生并正确停止进程
- 配方适用 — 设定值更新
- 趋势记录每个关键变量的数据
- PLC 导出通过验证，没有错误

### 4.调试：

- 操作员经过培训
- 项目备份已完成 (POST /api/project/export/)
- 分配角色和权限（操作员、值班经理、工程师）
- 第一批在减少的参数上启动，作为健全性检查
- 工作场所文件已准备好

## 故障排除和常见问题解答

本节介绍典型问题以及如何解决这些问题。

### 连接和身份验证

问：黄色的“连接丢失”横幅一直显示。

可能的原因：

- Rust Worker 或 Django 后端已关闭 — 运行 `docker-compose ps`；所有 6 个容器必须处于“Up”状态。
- Nginx 不代理 WebSocket — 打开 DevTools → Network → WS。关闭代码1006或502指向nginx。检查 `nginx.conf`：需要 `proxy_http_version 1.1` 以及 `Upgrade/Connection` 标头。
- JWT 令牌已过期 — 注销并重新登录。在正常情况下，令牌会在过期前 1 分钟自动刷新。
- 网络防火墙阻止端口 8001 (Rust WebSocket)。系统回退到 Django WebSocket，但会增加延迟 - 打开端口。

问：升级后，会出现橙色“版本不匹配”横幅。

前端和后端版本不同。使用缓存旁路刷新浏览器

(`Ctrl+Shift+R`)。如果这没有帮助，请检查“`docker-compose.yml`”：“`frontend`”图像标签必须与“`backend`”和“`scada-worker`”匹配。

问：我无法登录 — «无效的凭据»。

- 检查键盘布局 (CAPS LOCK)。
- 确保您使用的是用户名，而不是电子邮件（如果设置需要特定的用户名）。
- 如果密码丢失且未配置 SMTP — 在服务器上重置密码：

```
docker-compose exec backend python manage.py changepassword <username>
```

### 架构和设备

问：架构上的设备保持灰色且其值不会更新。

- PLC 离线 — 检查 PLC 中的状态。如果不是“在线”，则设备不会获取任何数据。
- 设备未连接到 PLC — 打开设备卡，检查“PLC”字段。
- WebSocket 通道未订阅 — 检查 DevTools → Network → WS：应该有一个 `ws/schema/{id}/devices` 连接。如果没有，请重新加载页面。
- 您缺乏设备权限 — 请管理员检查您的角色。

问：尽管液位传感器工作，但水箱显示容积为 0。

- 未配置捆扎表（对于“STRAPPING”方法）。打开水箱 → «捆扎表» 选项卡 → 添加至少 2 个点 (`sensor_value` → `volume`)。
- 错误的计算方法 — 检查 `volume_method: STRAPPING` 或 `GEOMETRIC`。“`GEOMETRIC`”需要“`TankSection`”条目（圆柱体、圆锥体等）。
- Rust Worker 未更新卷 — 检查“`docker-compose` 日志 `scada-worker` |” `grep Tankvolume``。默认情况下，每秒计算一次体积（“`TANKVOLUMEINTERVALMS`”）。

问：即使电机正在运行，管道也不会产生流动。

在架构生成器的右侧属性面板中为管道启用进程监控。默认情况下它是关闭的。

问：当我拖动设备时，它会跳回来。

当 WebSocket

更新同时到达时会发生这种情况。如果您在架构生成器中，编辑会阻止实时更新；移动后立即保存 (Ctrl+S)。

## 序列和配方

问：即使满足条件，序列也不会转换到下一步。

- 条件不绑定到步骤 — 打开当前步骤，检查 “next\_steps”。目标步骤必须列出条件 ID。
- `duration\_seconds` 尚未过去 — 条件必须在配置的时间内持续保持为真。查看条件弹出窗口中的 “连续” 字段。
- 使用另一个值模拟条件 — 检查 WebSocket 条件有效负载中的 “issimulated” 标志。使用 POST /api/transition-conditions/clearsimulation/和{"step\_id": }重置。
- 设备价值未交付 — 设备离线 (见上文)。

问：我更改了配方，但 PID 设定值没有改变。

确保：

- 配方被激活 — POST /api/production-recipes/{id}/select-production-recipe/。
- 配方变量绑定到序列编辑器中的设备字段 (“步骤” 选项卡 → 检查每个步骤的设备值)。
- 序列不在 is\_pause = true 中 — 暂停会冻结所有转换和配方应用程序。

问：我想在没有真实设备的情况下测试逻辑。

- 使用转换条件模拟 — POST /api/transition-conditions/{id}/simulate/ 和 {"simulated\_value": 80.0}。该条件评估提供的值而不是实际值。
- 对于模拟输入，启用 模拟模式：POST /api/analog-inputs/{id}/set\_simulation/ 和 {"is\_simulated": true, "simulated\_value": 75.0}。
- 测试后，始终关闭模拟——这在生产中是危险的。

## 批次

问：批次挂在舞台上并且无法继续移动。

- 检查当前阶段序列的最后一步的 “batchtransferaction” ——它必须是 “TRANSFER” (继续) 或 “COMPLETE” (完成)。
- 检查下一个阶段是否存在并且处于活动状态 (is\_active: true)。
- 检查 Rust 工作日志 — “批量传输失败” 消息解释了原因。
- 手动传输：POST /api/batches/{id}/transfer/ 和 {"next\_stage\_id": }。

问：批次队列顺序错误。

首先按优先级排序，然后按创建时间排序。更改优先级：POST /api/batch-queue/{id}/set\_priority/ 为 {"priority": "HIGH"}。有效值：“低”、“正常”、“高”、“紧急”。

## PLC导出

问：按 “验证” 会生成错误列表。

常见验证错误：

- “设备 ID 包含无效字符” — ID 必须仅与 “[a-z0-9]” 匹配，不能包含破折号或空格。
- “序列引用丢失设备” — 设备已删除，但步骤中的引用仍然存在。打开序列编辑器并修复。
- PID output\_max <= output\_min — 检查 PID 范围。
- “储罐捆扎条目必须是单调的” - “sensor\_value” 必须严格增加。
- “没有将 PLC 分配给设备 X” — 将设备连接到 PLC。

问：生成的 SCL 无法在 TIA Portal 中编译。

- 检查 TIA Portal 版本 - 生成器面向 TIA Portal v16+。

- 确保两个 XML 标记表（输入和输出）在 SCL 之前导入。
- 阅读 TIA Portal 编译日志 — 它指向特定的 FB 和行。

## 趋势

问：即使设备正在工作，图表也是空的。

- 时间范围太小或在未来 — 重置为“过去 1 小时”。
- 时间刻度离线 — 检查“GET /api/graphs/timescale-status/”。如果它返回“disconnected”，则说明 TimescaleDB 已关闭或无法访问。
- 未收集数据点 — 在 数据点 部分中，设备必须具有有效的收集订阅。

问：大范围内趋势缓慢。

使用下采样。如果样本仍然过多，请缩小范围或增加聚合间隔。对于长期分析，可导出为并在外部进行处理。

CSV

## 人工智能助手

问：AI助手返回“验证失败”。

- 在设置 → AI助手中查看OpenRouter API密钥。密钥在保存时使用 Fernet 进行加密 - 该错误意味着密钥过期或数据库条目损坏。
- 检查您的 OpenRouter 余额 — 当资金用完时，请求将被拒绝。
- 检查日志：docker-compose 日志 ai-assistant |尾-50。

问：人工智能很慢。

该模型通过工具调用运行代理循环最多 15 次迭代。复杂的请求（“分析上周的所有事故并提出优化建议”）可能需要 30-60 秒。为了快速得到答案，请保持问题简短而具体。

15

30-60

## 性能

问：UI 在具有 100 多个设备的架构上滞后。

- 在不参与活动进程的管道上禁用流动动画。
- 在 设置 → 性能 中启用“限制非关键更新”（非活动设备每 2 秒更新一次，而不是 250 毫秒）。
- 为 Docker 提供更多 RAM（至少 4 GB 以保证舒适运行）。

问：数据库增长很快。

主要来源是活动日志和历史趋势。在设置 → 数据管理中配置保留：

- 活动日志：默认为 90 天。
- 趋势数据：可通过数据点对每个设备进行配置。

## 安全和最佳实践

本节收集有关安全系统操作的建议。遵循这些标准可降低事故风险并有助于遵守质量标准（ISO 9001、HACCP、GMP）。

### 使用自动/手动模式

手动模式让操作员可以直接控制旁路顺序和警报联锁。它是一种强大的工具，也是一种危险的工具。

手册的正确使用：

- 仅用于调试和诊断。
- 仅在 停止 设备上或在明确为其设计的状态下。
- 得到运行线上值班经理的口头批准。
- 在活动日志中记录一个条目（这会自动发生，但操作员应该知道）。

手册的错误使用：

- 绕过安全条件以加快进程。
- 在不知道当前批次状态的情况下手动启动设备。
- 轮班结束时将设备保留为手动状态。

规则。当您在手动模式下使用完设备后，请务必将其放回自动模式。系统不会为您做这件事。

### 模拟：仅测试

条件模拟、模拟输入和计时器是工程师的工具，而不是操作员的工具。

场景	允许吗？
在测试台上测试序列逻辑	是的
在模拟器中验证事故点火	是的
“跳过” 生产过程中的一个条件以更快地启动	绝对不是
由于真实传感器损坏而模拟传感器	否 — 修复传感器或停止进程

生产过程中的任何实时模拟都是事故的潜在原因。在启动生产批次之前，运行检查 “GET /api/transition-conditions/?is\_simulated=true” ——该列表不得包含与活动序列相关的任何内容。

### 备份和恢复

要备份什么：

- 项目配置 — 设备、模式、序列、配方、事故。使用 “POST /api/project/export/” → ZIP。
- 数据库 — 完整的 PostgreSQL 转储（包含日志、历史趋势、批量数据）：

```
docker-compose exec postgres pg_dump -U scada scada_db > backup_$(date +%Y%m%d).sql
```

- nginx 和 docker-compose.yml — 保存在 Git 下。

多久一次：

- 项目配置 - 在任何重大更改之前。
- 数据库——每日自动化+系统升级前。

- 升级 SCADA 版本之前 — 两种备份类型都是必需的。

测试您的恢复。每季度一次，将备份恢复到测试台并进行验证。从未恢复过的备份是不可靠的备份。

## 访问控制

最小权限原则。每个用户仅获得其工作所需的权限。

典型角色矩阵：

角色	权限
操作员	查看模式+自动控制设备+确认事故。没有编辑权限。
值班经理	操作员可以做的一切+启动/停止批次+从允许的列表中更改配方。
技术专家	以上所有内容+创建/编辑配方和变量。
自动化工程师	以上所有内容+编辑模式、序列、设备。
管理员	完全访问，包括用户管理和系统设置。

还有：

- 按模式访问 — 可以对不在该区域工作的用户隐藏单个模式。
- IP 过滤 — 限制对内部网络的访问（192.168.0.0/16 或 VPN）。
- 定期审核 — 每季度审核用户中的活跃用户列表并删除离开者。

## 系统升级

正确的升级顺序：

- 规划一个窗口 — 升级需要 5-15 分钟的生产停机时间。
- 进行备份（配置+数据库）。
- 阅读变更日志 — 可能会发生重大 API 更改或迁移。
- 运行试运行 — 系统验证迁移而不应用它们。
- 应用升级 — 通过设置 → 系统 → 更新。
- 等待冒烟测试 — 系统自动运行基线检查。
- 如果出现问题，请使用回滚（升级后 24 小时内可用）。

切勿在批量运行期间或存在未解决的严重事故时进行升级。

## 处理事故

发生事故时该怎么办：

- 保持冷静。阅读警报栏中的事故描述。
- 前往触发它的设备。
- 目视评估状态（泄漏、过热、异常噪音）。
- 如果安全，请按警报中的确认。
- 排除原因。
- 修复后按解决。

不该做的事：

- 无需审查即可批量确认所有警报（“全部解决”适用于特殊情况，仅由轮班经理决定）。
- 生产运行时从配置中删除事故事件。
- 降低严重性级别以消除嘈杂的警报 - 相反，设置正确的阈值。

## 活动日志

---

活动日志是您的安全网和事实来源。它会自动记录：

- 登录和退出。
- 每个设备控制命令（启动、停止、打开、关闭）。
- 食谱和顺序的更改。
- 火灾事故。
- 批量启动/停止/传输。

建议：

- 保留日志至少 1 年（许多质量标准的要求）。
- 定期导出为 CSV 以便存档。
- 调查事件时，首先查看事件周围  $\pm 1$  小时窗口的日志。

## 命名建议

---

随着项目的发展，一致的命名约定会带来回报。

设备：

- 按类型前缀：“mot”（电机）、“val”（阀门）、“ai”（模拟输入）、“pid”、“tnk\_”（水箱）。
- 以零填充的编号：“mot001”、“mot002”（最多“mot\_099”）。
- 按区域分组：val\_tank1\_in、val\_tank1\_out。

序列：

- 处理前缀：seq\_pasteurize\_htst、seq\_cip\_alkali。

食谱：

- 带参数的产品名称：Milk\_Whole\_3.5%、Cream\_30%\_HotFill。

## 术语表

系统和本手册中使用的术语的参考。

### 一般自动化

SCADA (监控和数据采集) ——一类用于监视和控制工业过程的软件系统。

PLC (可编程逻辑控制器) ——实时执行控制逻辑的工业计算机。

HMI (人机界面)。在该系统中, HMI 是 Schema Viewer。

P&ID (管道和仪表图) — 显示管道和仪表的流程图。

I/O — 输入/输出, PLC 和现场设备之间的通道。

标签 — 已命名的 PLC 信号。在 SCADA 中, 它映射到读取/写入值的内存地址 (例如, 西门子的 “DB10.DBDO” )。

### 设备

AI (模拟输入) — 接受连续信号 (例如 4–20 mA、0–10 V) 并将其转换为工程单位。

AO (模拟输出) — 发出连续控制信号 (VSD 设定点、阀门位置)。

DI / DO (数字输入/输出) — 逻辑 0/1 信号 (按钮、限位开关、灯、接触器)。

PID (比例-积分-微分) — 一种具有三个组成部分的控制算法: 比例 (Kp)、积分 (Ki)、微分 (Kd)。

设定值 (SP) — 受控变量的目标值。

过程值 (PV) — 受控变量的当前测量值。

VSD (变速驱动器) — 控制电机速度的变频器。

COS (协调信号) ——用于跨进程同步的逻辑变量 (例如 “第 1 行准备好批量传输” )。

计数器 — 脉冲/事件计数器。用于流量累计、圈数、循环。

联锁 — 阻塞条件 (例如 “阀门关闭时请勿启动泵” )。

### 坦克

捆扎表 — 将液位传感器读数映射到罐容积的表格。用于具有非标准几何形状的储罐。

HH / H / L / LL — 高高/高/低/低低。储罐有四个警报级别: 极高、高、低、极低。

静压液位 — 根据上部压力传感器和下部压力传感器之间的压力差 ( $\Delta P$ ) 得出的液位。

### 序列和配方

序列 — 描述流程的有序步骤的状态机。

步骤 — 序列的一种状态。

转换 — 具有触发条件的两个步骤之间的链接。

转换条件 — 触发器: 设备 (设备值比较)、计时器 (经过的时间)、选项 (操作员按钮)。

配方 — 序列每个步骤的特定设备值。

生产配方 — 特定产品的一组变量。

配方变量 — 其值取决于产品的参数。

选项 — 操作员驱动的控制信号（开始/暂停/跳过按钮）。

## 批次和生产线

批次 — 经历该过程的生产单位。该概念来自ISA-88。

生产线——生产商品的一套设备。

生产线阶段 — 生产线上的一个加工阶段（例如清洗 → 干燥 → 包装）。

批量传输 — 一批从一个阶段移动到下一个阶段。由序列步骤上的“batchtransferaction”字段驱动。

批队列 — 等待上线的批次队列。

ISA-88 — 配方驱动生产中批量控制的国际标准。

## PLC出口与西门子

TIA Portal — 全集成自动化博途，西门子的 PLC 配置 IDE。

SCL（结构化控制语言）— 西门子 PLC 语言（IEC 61131-3 结构化文本）。

FB（功能块）— 具有逻辑及其自身实例数据的程序单元。

DB（数据块）— PLC 变量的存储结构。

OB（组织块）— PLC 代码的入口点。

OB1 — 主循环 OB，每次扫描时调用。

SimaticML — 用于 TIA Portal 项目导出/导入的 XML 格式。

GSD 文件（通用站描述）— PROFIBUS/PROFINET 的硬件模块描述符。

## WebSocket 和 API

REST API — 通过 HTTP 进行 CRUD 操作的代表性状态传输。

WebSocket — 一种双向 TCP 协议，支持服务器到客户端推送。

JWT（JSON Web 令牌）— 一种以签名形式携带用户数据的身份验证令牌。

SSE（服务器发送事件）— 单向服务器到客户端流式传输。由 AI 助手用于流式回答。

增量更新 — 仅发送对象的已更改字段。

批处理 — 将多个更新分组到一条消息中以减少网络负载。

## 行业标准

HACCP（危害分析和关键控制点）——食品安全方法。

GMP（良好生产规范）——一种生产管理标准，特别是在制药和食品领域。

CIP（就地清洗）——无需拆卸即可自动清洗加工设备。

HTST（高温短时）—高温短时巴氏灭菌（72–75 °C，15–20 秒）。

ISO 9001 — 国际质量管理标准。

## 附录 A: WebSocket 通道参考

频道	描述	方向
ws/device/{id}	设备状态	服务器→客户端
ws/device/{id}/control	设备命令	双向
ws/schema/{id}/devices	一个模式的所有设备	服务器→客户端
ws/schema/{id}/pipes	架构管道	服务器→客户端
ws/schema/{id}/sequences	模式序列	服务器→客户端
ws/schema/{id}/batches	架构批次	双向
ws/schema/{id}/accidents	架构事故	服务器→客户端
ws/sequence/{id}	序列状态	服务器→客户端
ws/序列/全部	所有序列	服务器→客户端
ws/pipe/{id}	管道状态	服务器→客户端
ws/系统	系统指标	服务器→客户端
ws/plc/状态	PLC状态	服务器→客户端
ws/设备状态	全球现状	服务器→客户端
ws/cos/全部	协调信号	服务器→客户端
ws/数学函数	数学函数	服务器→客户端
ws/活动日志	活动日志	服务器→客户端
ws/事故	所有事故	服务器→客户端

Rust WebSocket 服务器 (端口 8001, 通过 `/rws/ws/` 代理) :

- 批量设备更新时只读
- 自动回退到 Django WebSocket
- URL查询参数中的JWT认证

## 附录 B: API 参考 (汇总表)

资源	基本网址	增删改查	额外行动
授权	/api/auth/	—	登录、注册、我、刷新
用户	/api/users/		设置密码、上传头像、角色
角色	/api/roles/		权限
权限	/api/permissions/		类别、行动
设备 (全部)	/api/devices/	—	alldevices、batchwebsocket_update
模拟输入	/api/analog-inputs/		设置模式、设置模拟、设置互锁、设置手动值
模拟输出	/api/analog-outputs/		(如上所述)
数字输入	/api/digital-inputs/		(如上所述)
数字输出	/api/digital-outputs/		(如上所述)
电机	/api/motors/		(如上所述)
阀门	/api/valves/		(如上所述)
PID控制器	/api/pid-controllers/		(如上所述)
专柜	/api/counters/		(如上所述)
定时器	/api/timers/		(如上所述)
COS	/api/coordination-signals/		(如上所述)
变频驱动	/api/variable-speed-drives/		(如上所述)
坦克	/api/tanks/		计算体积, 导入/导出_strapping
坦克部分	/api/tank-sections/		—
坦克捆扎	/api/tank-strapping-entries/		—
PLC	/api/plcs/		状态、设置运行模式
架构	/api/schemas/		添加/删除设备、序列、选项、生产线
管道	/api/pipes/		—
序列	/api/sequences/		步骤、转换、位置
顺序步骤	/api/sequence-steps/		—
过渡条件	/api/transition-conditions/		模拟, clear_simulation
食谱	/api/recipes/		—
生产食谱	/api/production-recipes/		选择生产配方
配方变量	/api/production-recipe-variables/		—
活动	/api/events/		设置活动
事故事件	/api/accident-events/		—
事故	/api/accidents/		承认、解决、解决所有
序列选项	/api/sequence-options/		—
生产线	/api/production-lines/		激活、停用、create_batch
生产阶段	/api/production-line-stages/		—
批次	/api/batches/		开始、暂停、恢复、取消、转移
批量队列	/api/batch-queue/		重新排序, 设置优先级
脉冲模块	/api/pulse-modules/		开始、停止

资源	基本网址	增删改查	额外行动	
数学函数	/api/math-functions/		计算、验证、测试	
I/O 点 (AI)	/api/points/analog-inputs/		验证、测试	
I/O 点 (AO)	/api/points/analog-outputs/		(如上所述)	
I/O 点 (DI)	/api/points/digital-inputs/		(如上所述)	
I/O 点 (DO)	/api/points/digital-outputs/		(如上所述)	
图形状态	/api/graphical-statuses/		上传图标、列表图标	上传图标
图表	/api/graphs/		数据、时间刻度状态	
数据点	/api/data-points/		—	
硬件目录	/api/hw-catalog/		导入-gsd	
硬件站	/api/hw-stations/		自动分配地址	
硬件插槽	/api/hw-slots/		批量更新	
汉王网络	/api/hw-network/		—	
TIA 部署	/api/tia/	—	部署、自动地址、地址映射	
TIA 代理配置	/api/tia-agent-config/		检查连接、代理状态	
活动日志	/api/user-activity-logs/	研发	最近, 统计, 清晰	
IP 接入	/api/ip-access/allowed-ips/		checkip、myips	
许可证	/api/license/	—	状态、激活、生成	
系统	/api/system/	—	信息、健康、更新、维护	
项目	/api/project/	—	导出、导入、备份、恢复	
PLC出口	/api/plc-export/	—	预览、验证、生成	
人工智能会议	/ai-api/v1/sessions		留言	
人工智能聊天	/ai-api/v1/chat	—	消息 (上交所)	
人工智能图像	/ai-api/v1/images	—	分析、应用	
人工智能设置	/ai-api/v1/settings	R/U	—	

## 附录 C: 错误和警报代码

用户在日志、WebSocket 有效负载和 API 响应中遇到的代码和状态的参考。

### 事故严重程度

“AccidentType” 或 “accidentevent.severitytype” 上的字段 “severity\_level” 。

代码	名称	颜色	推荐回应
1/低	低	蓝色	信息性的。没有立即采取行动。记录以进行统计。
2 / 中	中等	黄色	需要操作员尽快注意。生产可以继续。
3/高	高	橙色	需要立即引起注意。潜在的产品质量下降或设备损坏风险。
4 / 关键	关键	红色	立即序列停止。对人员或设备造成风险。

事件严重性类型（事件的 “SeverityType” ）：

代码	名称	使用
信息	信息	状态消息（开始、阶段完成）
警告	警告	无威胁性偏差
错误	错误	需要干预的关键问题

### 事故状态

“事故” 的 “状态” 字段。生命周期：

NEW → ACKNOWLEDGED → IN\_PROGRESS → RESOLVED → CLOSED

|-----|-----|-----| | “新” |新 |刚刚提出，未获承认。显示在警报栏中。 | | 已确认 |已确认  
|操作员确认收货 (POST /api/accidents/{id}/acknowledge/)。保持活跃但不再是首要任务。 || 进行中 |进行中  
|根本原因工作正在进行中。 || 已解决 |已解决 |根本原因已修复 (POST /api/accidents/{id}/resolve/)。 || 关闭  
|关闭 |已解决 + 延迟后的最终状态（用于报告）。 |

### 序列状态

Sequence 模型上的字段：

领域	类型	意义	描述
正在运行	布尔	真实	序列激活，步骤切换。
is_pause	布尔	真实	暂停；转换被冻结，设备保持状态。
is_stop	布尔	真实	停止了；设备进入安全状态。

组合：

- is\_running=true, is\_pause=false, is\_stop=false — 正在运行。
- is\_running=true, is\_pause=true, is\_stop=false — 暂停（可恢复）。
- is\_running=false, is\_pause=false, is\_stop=true — 完全停止。

## 批次状态

代码	名称	描述
待处理	待定	已排队, 未开始
运行	跑步	活跃批次
暂停	暂停	被运营商暂时停止
已完成	已完成	成功完成所有阶段
取消	取消	手动中止
失败	失败	因严重事故而中止

优先事项:

代码	排队顺序
"紧急"	首先运行
"高"	2
正常	3 (默认)
低	4

## 设备模式

适用于电机、阀门、PID、AI/AO/DI/DO:

代码	描述
自动	由顺序/系统逻辑控制
手册	操作员直接控制
模拟	模拟值 (仅测试)
互锁	通过联锁锁定 (值无法更改)

## API 错误代码

典型的HTTP代码及其在本系统中的含义:

代码	意义	典型原因
200 好	成功	请求已完成
201 已创建	创建	创建新对象
204 无内容	成功, 无身	删除
400 错误请求	无效请求	验证失败。正文是 {"field": ["error msg"]}
401 未经授权	未经验证	JWT 缺失或无效
403 禁止	没有权限	已认证但未授权
404 未找到	没有找到	ID不存在
409 冲突	冲突	例如。使用现有 ID 创建设备
423 已锁定	锁定	对象被另一个进程锁定 (例如, 在序列运行时保存模式)
500 内部错误	服务器错误	错误或异常。检查 docker-compose 日志后端
502 错误网关	代理错误	Nginx 无法到达后端/Rust Worker

代码	意义	典型原因
503 服务不可用	不可用	维护模式或过载

## WebSocket 关闭代码

当WebSocket连接关闭时，可能会观察到以下代码：

代码	描述
1000	正常关闭
1001	服务器消失（重新启动）
1006	异常关闭（连接丢失）。前端自动重新连接。
1008	违反策略（例如 URL 查询参数中的 JWT 无效）
1011	内部服务器错误
4001	自定义：身份验证失败
4003	自定义：禁止（无频道权限）
4004	自定义：找不到频道

## 转换条件类型

代码	描述
设备	基于设备值（与阈值相比）
定时器	基于持续时间（进入步骤后）
选项	基于操作员选项按

比较运算符（对于 type=`device`）：

符号	意义
==	平等
!=	不等于
>	大于
<	小于
>=	大于或等于
<=	小于或等于

## 步骤上的批量操作 (batch\_transfer\_action)

代码	描述
无	没有行动；批次停留在当前阶段
转移	将批次转移至下一阶段
完成	完成批次（最终状态）

## 罐盖形状类型

对于字段 “SchemaDevice.configuration.tank\_visual” :

底部 (`bottom\_type`) :

代码	描述
平	平
圆顶	球形 (凸向下)
菜	准球形 (椭圆形)
'锥体'	圆锥形

顶部 (`top\_type`) :

代码	描述
平	平
圆顶	球形
菜	准球形

基于源代码分析生成的文档。对于屏幕截图占位符：打开应用程序，导航到指示的屏幕并拍摄与所描述的元素匹配的屏幕截图。