基于iMaster NCE-CampusInsight的大数据疫情监控分析系统

设计文档

所在赛道与赛项：（*填写“A/A-ST/B-EP1/B-EP2”其中一项）。*

1. 目标问题与意义价值

*说明作品的应用领域，解决或关注的问题，实现的目标与基本功能，以及作品的理论意义或应用价值。*

去年年初疫情突发，新型冠状病毒具有极强的人传人能力，为了疫情监控政府推出健康码，行程码等手段，但这些只能大概的定位用户去过哪些地方，得到一个大体的行程路线。当患者进入某些封闭或半封闭的园区（非公共管辖范围）时，则无法获取到更小精度的行动轨迹，导致该园区的所有人员都是高风险人员。

因此本系统在传统的SDN系统基础上，添加了疫情监控、用户行为分析等功能。本系统可以在此基础上弥补“最后一公里“的不足，通过输入确诊者的手机唯一识别号（mac）准确的定位出患者在该系统覆盖范围内具体去过哪些地方并且生成出一条可视化的轨迹路线，对于防疫工作来说具有更细的精度和更加准确的摸排。除了可视化路线外，该系统还会返回患者轨迹附近有可能与患者接触的人员名单。该名单可以帮助相关部门更快的找到疫情传播轨迹、源头等信息。并且通过接触人员名单，通过热力图等形式直观的显示出疫情的扩张趋势、有可能被污染的区域。

其次该系统还可以接入政府相关平台获取现存确诊名单，通过比对数据库接入数据可以实现当有患者曾来过或正滞留在园区时快速响应告警。如相关部门需求，该系统还可以提供相应的api接口给有关部门调用，有助于疫情的防控。

此外，本系统还有人群热力图功能，通过用户接入信息、用户行为数据可以帮助管理者分析出系统范围内用户的画像，帮助管理者更好的改善和管理园区。比如改善园区人流的导流、改善交通规划等等

1. 设计思路与方案

**技术选型：**

因为该系统是基于华为提供的iMaster NCE-CampusInsight接口（下称华为接口）实现的，因此我们需要先从华为接口获取原始数据，加工处理后返回给用户。此外还计划实现多端设备查看园区网络数据和将园区数据推送或提供接口给相关部门调度。故这里选择前后端分离的架构。

**框架与工具选择：**

前端框架方面，JavaScript框架选择Vue.js、网络请求库使用Axios、UI框架使用element UI。

后端框架选择的是APIFlask框架，该框架是一个基于Flask和marshmallow-code的轻量级web框架。ORM框架方面选择的是SQLAlchemy，鉴权方面使用的是APIFlask集成的Flask-HTTPAuth扩展。网络请求方面用的是aiohttp和requests库。

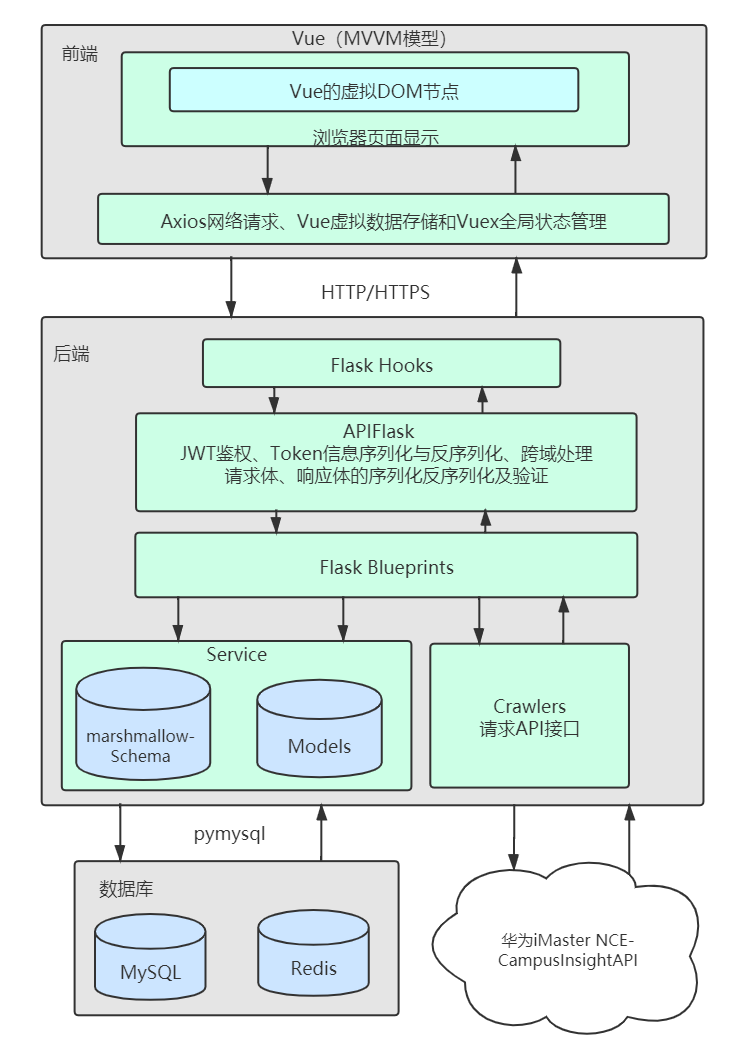
数据库方面才用MySQL作为持久化存储，Redis作为缓存数据库。

**详细设计：**

基础端口（不需要计算加工）后端根据前端参数进行请求

加工端口: 后端根据前端请求，将任务添加到任务队列里面，定时任务定时从任务队列里面拿任务加工处理，前端一段时间后再请求相应接口获取加工好的数据。

**整体设计流程如图所示：**

****

1. 方案实现

1. 前端设计

1.1 页面解构设计

总体结构按照以下结构进行设计

1）系统状态（系统总览）

1. 站点选择

2. 时间选择

3. 系统信息概览

2）用户列表

1. 用户列表

2. 用户数据分析

3）健康度分析

1. 接入成功率

2. 接入耗时

3. 信号干扰

4. 漫游达标率

4) 园区热力地图

5）用户轨迹图

1.2 组件设计

1.2.1 站点数设计

功能：能展示站点的树形结构，站点数据的获取与处理逻辑在组件内部进行

用途：在需要使用的页面，提供给用户选择站点功能。

1.2.2 时间选择器

功能：能让用户方便的选择日期，组件默认选择近一周。

用途：各个需要依赖Unix时间戳参数的功能

参数：start->起始时间；end->结束时间

组件事件:

i. changes：事件被更改后触发，参数一作为数据对象

ii. 参数一：

type：Array

describe：以数组形式返回开始时间和截止时间

1.3 页面功能设计

1.3.1 系统总览（或者叫系统状态）

功能：浏览系统各个维度的信息

布局：页面以弹性布局为主，每个维度的状态单独封装成一个盒子组件（卡片组件）自动根据页面宽高进行自适应。

展示内容：

1、接入设备数量、AP数量、历史接入设备数量（状态概览卡片）

2、登录用户信息；（账号信息卡片）

3、接入失败信息；（echarts饼图）

4、站点总速率；（echarts 仪表盘）

5、站点状态概览；（echarts 雷达图）

6、服务器cpu使用情况；（echarts 仪表盘）

7、服务器内存使用情况。（echarts饼图）

1.3.2 用户列表

相关API：获取用户数据分析信息

功能：接入园区网络所有用户（设备）的信息 （对站点下的用户数据进行统计和分析）.

展示内容：历史接入用户的概览以及详细信息、可以指定页码跳转并且能查看选中用户的轨迹路线。

布局：表格展示概况、详细卡片展示单独用户的详细情况、分页控制器（每页可选5,10,20,.40,100,200）

1.3.3 健康度分析

功能：展示园区站点接入信息以及各种健康度指标。

展示内容：漫游达标率、信号干扰、接入耗时、接入成功率等。

1.3.4 位置地图（热力图）

相关API：取热力图

数据处理方法（二选一）：

1. 直接计算映射权重：(count – 最小count) / (最大count – 最小count)

2. 选择性精准扶贫法：

i. 定义：

1. 太小：值小于平均值的5%

2. 一般小：值大于平均值5%但小于15%

ii. 处理：

1. 若数据太小，则丢弃

2. 若数据一般小，映射权重设为平均值的50% + 原值

3. 其他情况映射权重与原值相同

1.3.5用户轨迹图

功能：查看用户的园区内的移动轨迹。

展示内容：

1. 显示ap节点信息

2. 显示ap节点健康信息（绿色、黄色、红色表示）

3. 显示用户的移动轨迹

1.3.6 日志分析

功能：查看选择站点的详细日志信息

内容：各个时刻的关联总数、关联成功数、关联失败数、认证通过数、认证失败数、DHCP成功数、DHCP失败数等信息

布局：表格展示，内容作为表格的字段

1.3.7 账号管理

功能：查看SDN系统平台的账号以及管理相关账号的权限，登录状态和新增、删除账号等功能。

布局： 表格布局，账号状态用check button控制开关（开表示可登录，关表示不可登录）。

2. 后端设计

2.1 模块设计

后端系统采用工厂模式，各模块如下设计：

1. 蓝图模块（Blueprints）

2. 模型模块（Models）

3. 验证模块（Schemas）

4. 爬虫模块（Crawlers）

5. 错误处理模块（Exceptions）

6. 日志模块（Logger）

7. 配置模块（Settings）

8. 辅助函数模块（Utils）

2.2 爬虫系统设计：

1. 基础crawler

功能：请求华为API，获取原始数据（raw data）并返回

私有字段：token（表示华为API的token）

方法：

1) 生成http请求头，方法名：generate\_header

2) 处理GET请求的API，方法名: fetch

3) 处理POST请求的API , 方法名: post

4) 处理PUT请求的API，方法名: put

5) 获取华为API的token，方法名: get\_token

6) 上锁获取华为API的token，方法名：mtx\_token (loop\_token)

错误处理

1) 鉴权错误（没有token或者token过期）

处理方法：抛出token鉴权失败错误（错误类型从错误处理模块中导入）

2）连接错误

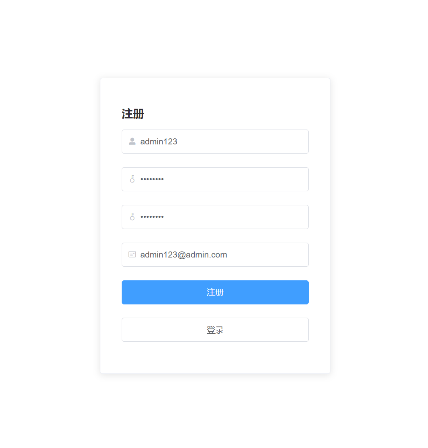
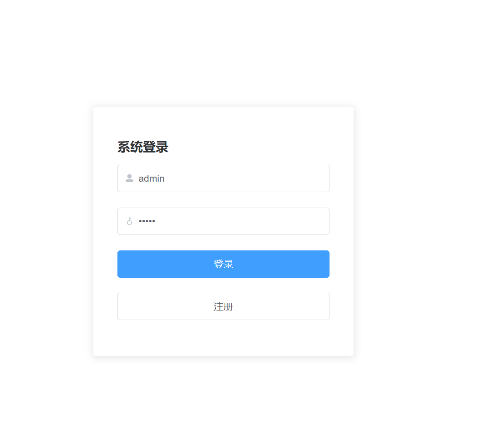
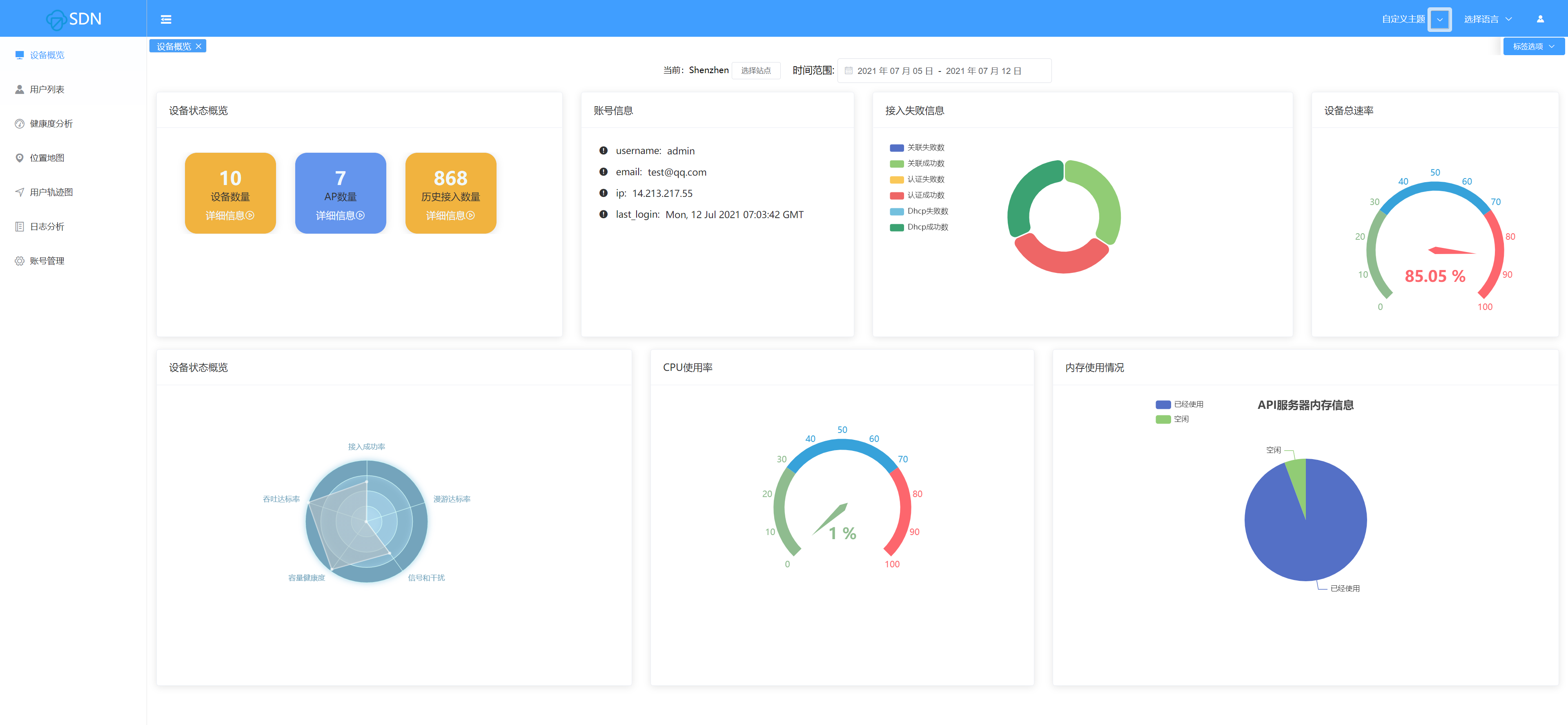
修改原始错误信息，抛到最外层处理handler处理。

2. 实现模块

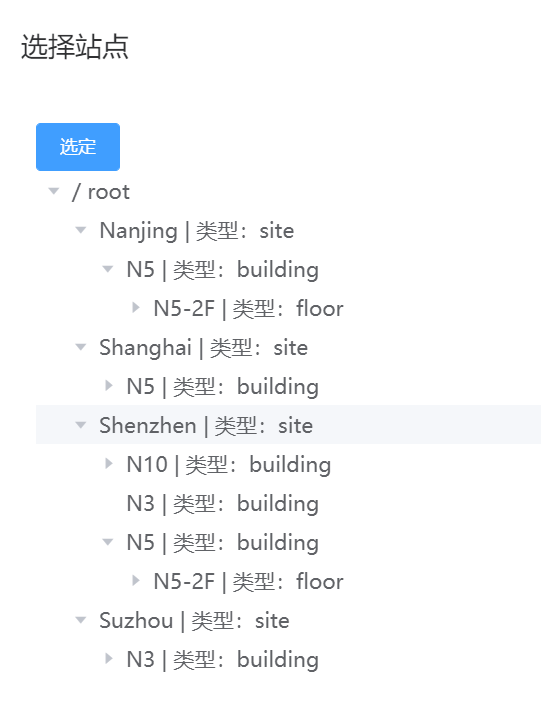
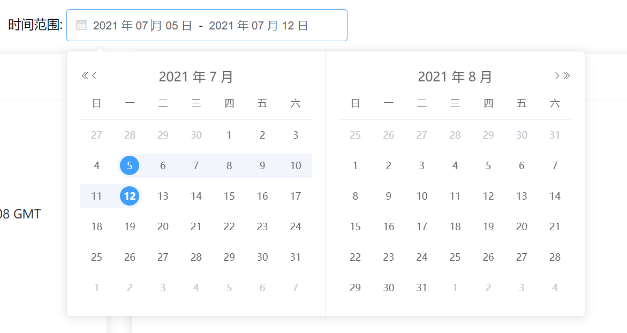
功能：根据华为API文档，请求具体数据。

处理token错误：利用基础crawler的mtx\_token（上锁获取token）方法获取token后再次请求接口数据。

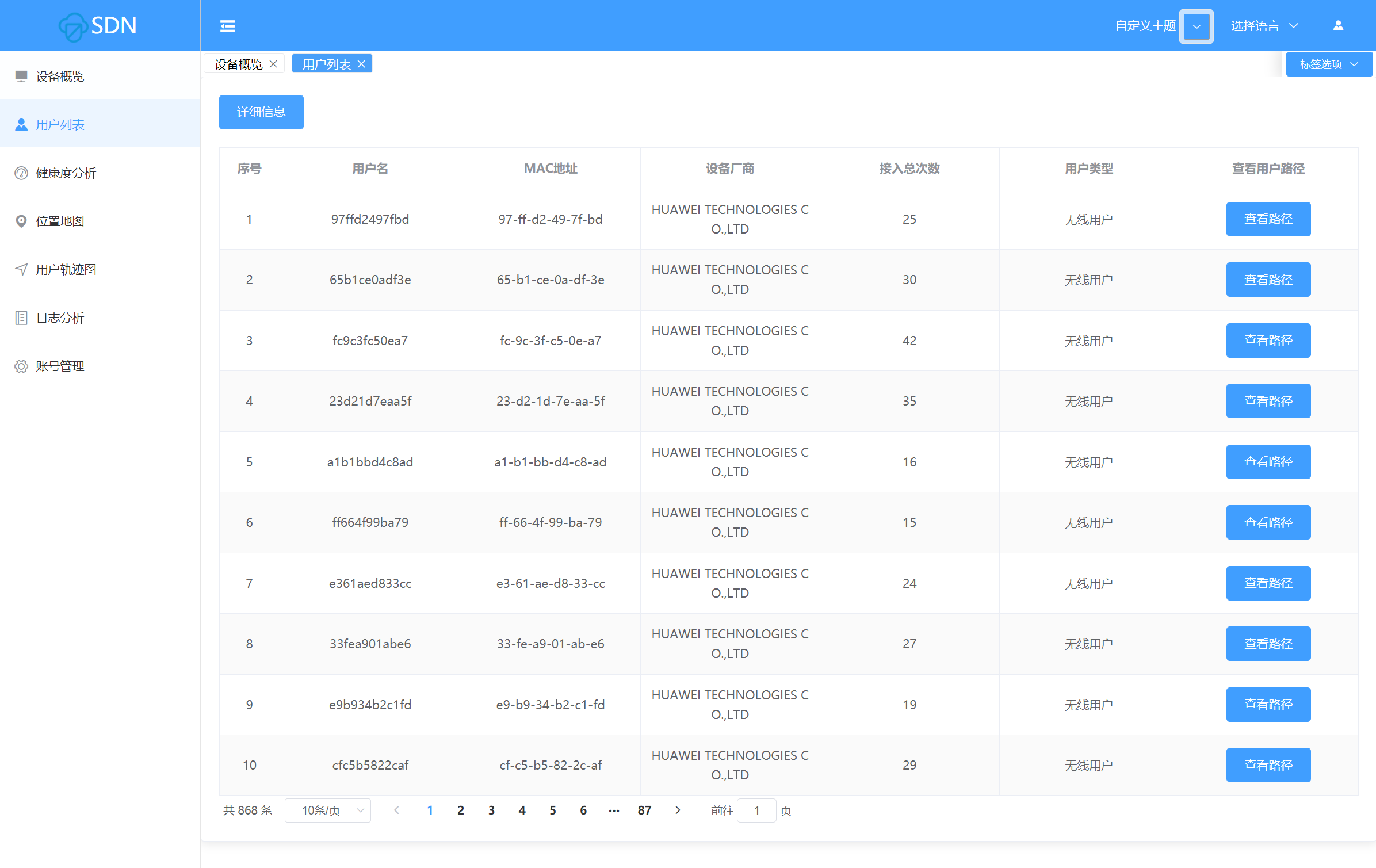
1. 运行结果/应用效果

选择器：

用户列表：



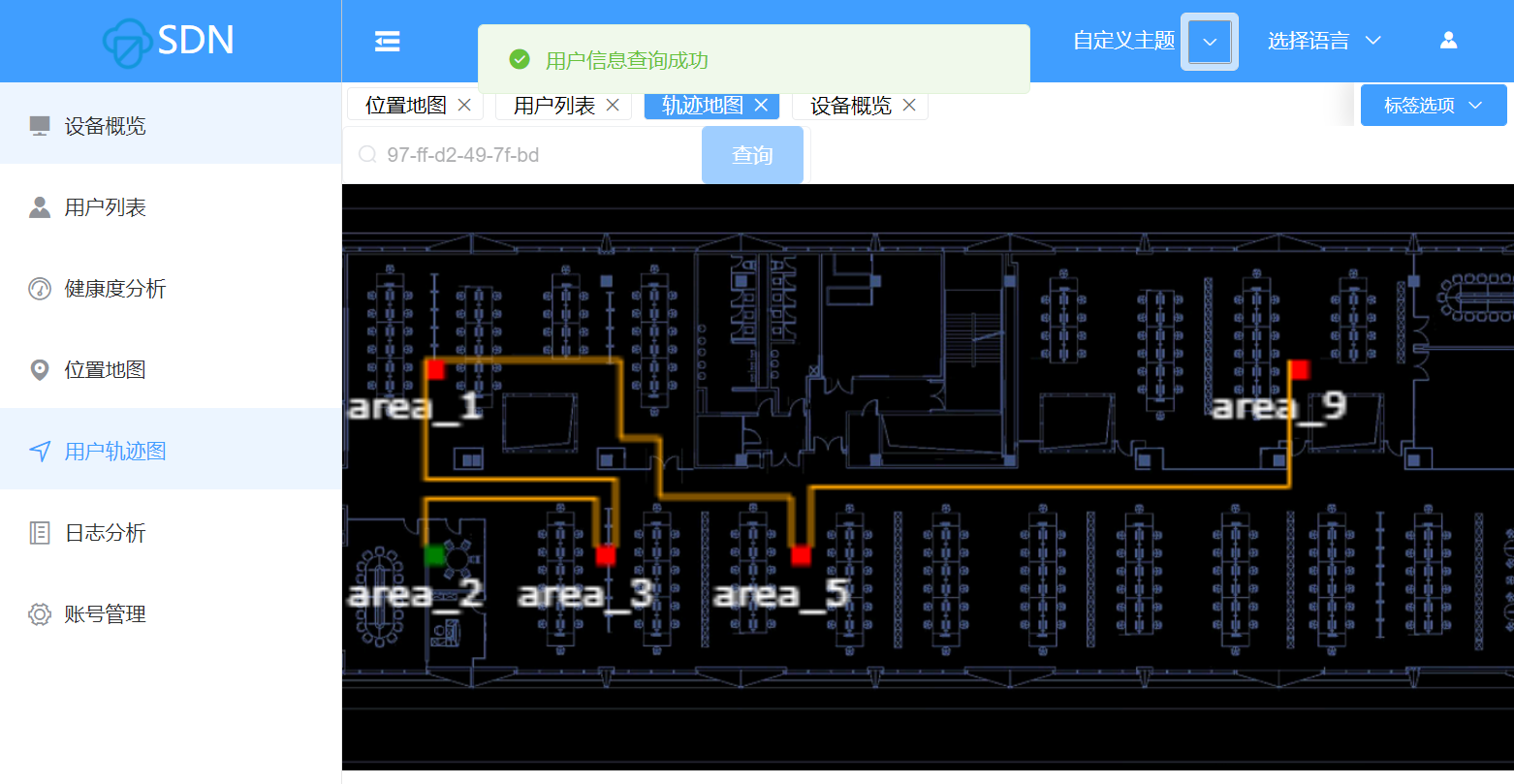
健康度：



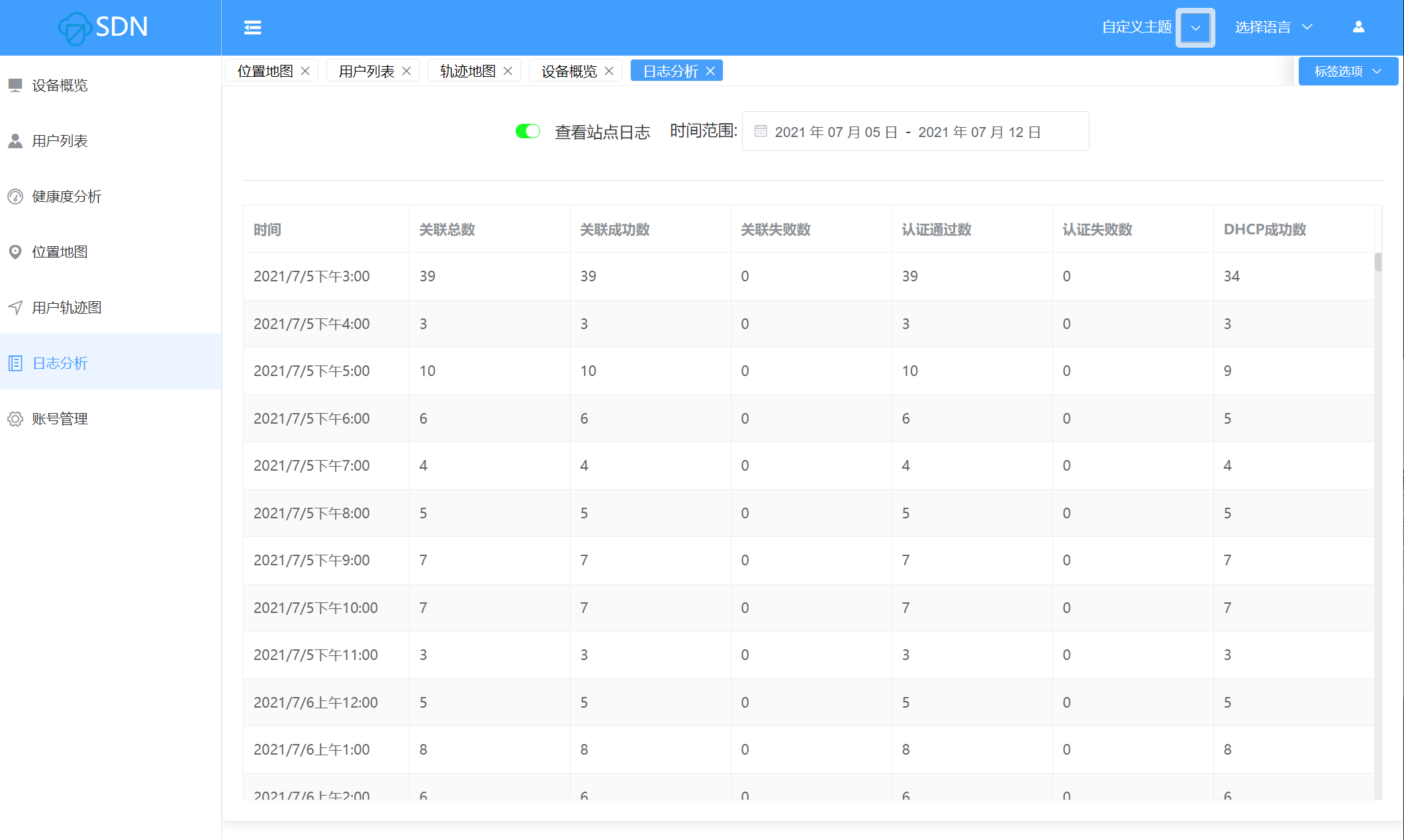
热力图：



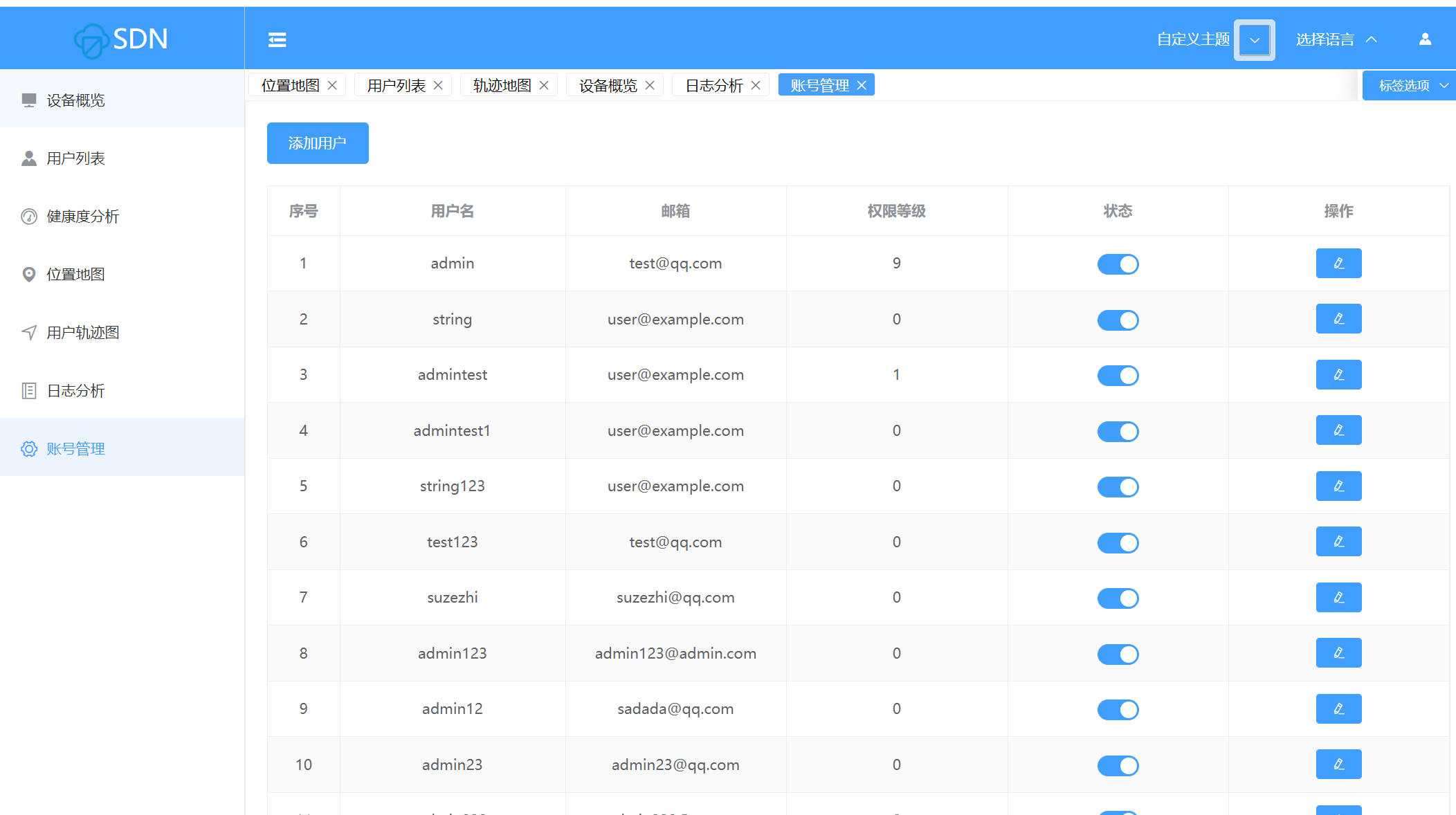
轨迹图：



日志分析：



用户管理：



1. 创新与特色

1. 与传统SDN相比，本系统加入了用户轨迹图功能，热力图功能，能更直观的看到园区人流信息、人流轨迹，方便管理园区。

2. 本系统能接入政府相关部门的平台，能及时的响应、上报疫情详情。

3. 本系统内置了中英双语及多种UI主题。

4. 后端使用了APIFlask框架和aiohttp，利用协程来处理华为API 的请求。