8班E组_分布式温控系统的解决方案

分布式温控计费系统

组别：308E

组长：2018211408 林振宇

组员：2018211404 余子劲

2018211393 王一丰

2018211405 甘鹏飞

2018211406 黄河阳

用户需求和系统目标

制作分布式温控系统，实现酒店各个房间空调调节与计费，并且酒店前台、空调管理员、酒店经理可以看到对应的数据。

分析该系统目标和功能如下：

1. 空调系统由中央空调和房间空调两部分构成；
2. 中央空调是冷暖两用，根据季节进行工作模式调整。
   a）当设置为供暖时，供暖温度控制在25°C～30°C之间；
   b）当设置为制冷时，制冷温度控制在18°C～25°C之间。
   c）计费标准1元/度
3. 中央空调具备开关按钮，只可人工开启和关闭，中央空调正常开启后处于待机状态。
   a）缺省工作温度为25°C
   b）当关闭后，不响应来自房间的任何温控请求；
   c）当有来自从控机的温控要求时，中央空调开始工作；
   d）当所有房间都没有温控要求时，中央空调的状态回到待机状态。
4. 房间内有独立的从控空调机，但没有冷暖控制设备。
   a）从控机具有一个温度传感器，实时监测房间的温度，并与从控机的目标设置温度进行对比，并向中央空调机发出温度调节请求。
   b）温度调节按钮连续两次或多次指令的时间间隔小于1s时，从控机只发送最后一次的指令参数；
   c）如果温度调节按钮连续两次的时间间隔大于1s时，从控机将发送两次指令参数；
5. 房间目标温度达到后，从控机自动停止工作，同时发送停止送风请求给中央空调。
   a）房间温度随着环境温度开始变化，当房间温度超过目标温度1 °C时，重新启动；
   b）到达目标温度后，客户端请求停止送风
   c）关机状态下，每分钟温度变化0.5°，直到初始温度
6. 从控机只能人工方式开闭，并通过控制面板设置目标温度，目标温度有上下限制。
   a）从控机开机后动态获取房间温度，并将温度显示在控制面板上；
   b）从控机开机后需要与中央空调进行连接认证，用户输入房间号+身份证号后，从控机从中央空调获取工作模式，并将工作模式和缺省工作温度显示在控制面板上；
7. 客户端有温度传感器，控制面板的温度调节可以连续变化也可以断续变化：
8. 中央空调能够实时监测各房间的温度和状态，并要求实时刷新的频率能够进行配置；
9. 要求从控机的控制面板能够发送高、中、低风速的请求，中风速每分钟变化0.5°，高风速每分钟变化率提高20%，低风速每分钟变化率减少20%
10. 中央空调实时计算每个房间所消耗的能量以及所需支付的金额，并将对应信息发送给每个从控机进行在线显示，以便客户可以实时查看用量和金额。
11. 系统中央空调部分具备计费功能：可根据中央空调对从控机的请求时长及高中低风速的供风量进行费用计算；
    a）每分钟中速风的能量消耗为1度/2分钟标准功率消耗单位；
    b）低速风的每分钟功率消耗为1度/3分钟标准功率；
    c）高速风的每分钟功率消耗为1度/1分钟标准功率；
    d）并假设，每一个标准功率消耗的计费标准是1元/度。
12. 资源调度,可以设计一个定时任务队列，可以分别设置不同角色的不同优先级，采用RR或者FIFO机制实现资源分配
13. 系统管理员，酒店前台，酒店经理，不同的角色拥有不同的权限
14. 要跟别的小组之间互相通信协作
15. 中央空调监控具备统计功能，可以根据需要给出日报表、周报表和月报表；报表内容如下：房间号、从控机开关机的次数、温控请求起止时间（列出所有记录）、温控请求的起止温度及风量大小（列出所有记录）、每次温控请求所需费用、每日（周、月）所需总费用。

需求分析

产品的功能需求：

1. 服务器需求分析

- 从机控制

  可通过对用户客户端发出指令来手动代开、调整或关闭某房间空调（从机）

- 监控从机状态

  服务器内置一个cache来保存各个房间的实时状态数据，使得管理员能够随时查看空调的状况。

- 与后端数据库的交互

  每当用户客户端发出指令改变空调状态时，服务器将相关数据发送至后端数据库中进行存储，当酒店经理需要查看详细表目以及用户退房需要打印详细账单时，从后端数据库中调出相关数据。

2. 从控机（用户客户端）需求分析

  - 显示功能

    从机（用户客户端）应该能够显示当前的状态，包括模式、风速、房间温度，以及实时能耗及费用。

  - 控制功能

    用户能根据需求选择不同的模式，包括制冷、制热，同时可以调节空调的温度和风速。

3. 用户对象分析

  本系统的面向用户有三类：入住顾客，中央空调管理员和酒店管理员

  - 入住顾客

    顾客可以根据自身需求打开或关闭空调，之后根据自身需求调节空调的温度或风力大小，可以查看已用信息（模式，风速，温度）。

  - 中央空调管理员

    中央空调管理员可以对中央空调进行特定操作（查看状态，控制从机），也可以设置中央空调计费标准。

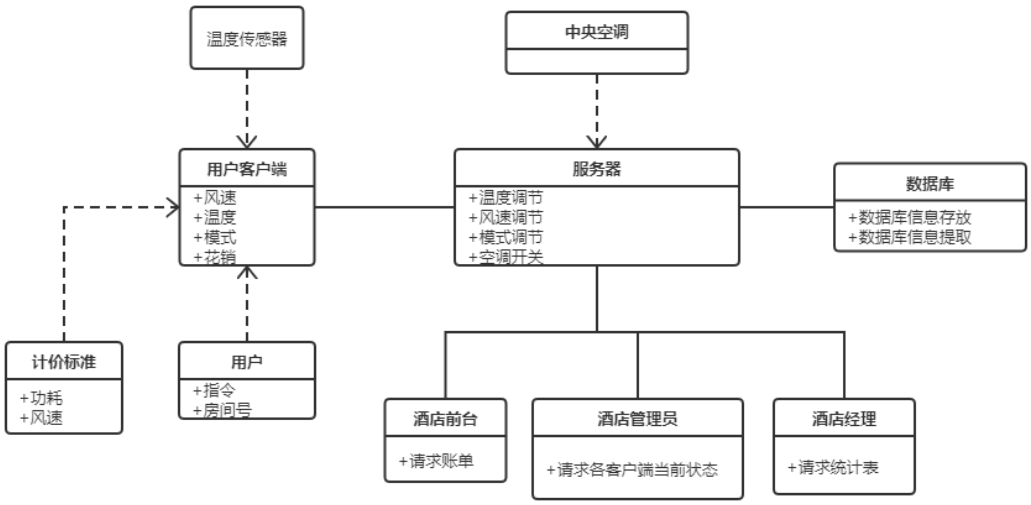
  - 酒店管理员

    酒店管理员可以对顾客查看顾客使用后的费用并生成报表。

产品的非功能性需求

  - 可靠性需求
    - 数据正确性:检验数值范围，是否存在异常
    - 数据保存:当服务器出现宕机，客户端出现故障，要恢复状态，另外要注意数据一致性
  - 安全保密需求
    - 用户操作权限限制，抵御恶意攻击
    - 用户密码通过加salt和SHA256方式加密
  - 性能需求
    - 高可用：保证系统快速响应，快速完成任务队列中的任务，即时显示真实数据
    - 可拓展：保证服务一定数量的客户端
  - 界面需求
    - 用户友好
    - 简洁美观
    - 数据可视化程度高

领域模型



技术路线

1. **编程环境**：Windows10系统，Ubuntu20.04， IDEA2019，jdk1.8。
2. **编程语言**：客户端前端javascript,html,css， 客户端后端java，前后端分离设计。服务器采用java语言实现，数据库采用SQL。
3. **具体设计框架**：客户端采用B/S架构，客户端与服务端采用C/S架构。
4. **前端**：使用Vue框架前端界面，在客户端上实现可视化，方便客户，中央空调管理员，酒店管理员进行操作，前端使用npm包管理工具完成依赖管理。
5. **后端**：使用SpringBoot进行后端开发，添加缓存功能用于降低服务器与数据库的IO压力，使用分布式锁解决客户与管理员对数据库读写的不一致问题，后端使用maven包管理工具完成依赖管理。
6. **前后端通信**：前后端之间基于TCP,利用WebSocket协议通信。
7. **服务器与客户端的通信**：用http+json实现客户端和服务端之间的通信。
8. **数据库**：计划刚开始在服务端采用缓存+数据库方式来存储数据，缓存或许采用Redis，数据库使用SQL Server。
9. **代码文档：**使用ProcessOn画图软件，结合Graphviz生成文档
10. **在后期改进中：**将尝试实现分布式存储，完成数据备份、节点动态扩展等高可用问题，或许将服务端改成集群，做好主从备份，异常恢复等功能，另外针对大型酒店，还要考虑高并发问题。

通信模式与数据库模式

1. 服务器的通信端口为6666。
2. 客户端四个操作，加温减温，增风减风，分别设为0 1 2 3，一字节存储。

3. 客户端给服务器发送“房间号（一字节）操作（一字节）”，其中房间号按ascii存，支持256个房间。

例：“+0”，意为43号房间加温。

4. 服务器收到消息后加上Datetime存到Log表中。

5. 退房的时候，查Log表，筛出信息，计算费用。

6. 经理查一段时间的详细信息时，查log表，按时间筛。

7. 客户端实时计费,在客户端完成，服务器不做此功能。

8. 数据库模式字段名

a) table_Rooms

```
{
    RID int
    TMode int default 0 //设定的温度模式：0关机，1缺省，2制冷，3制热
    EMode int //设定的耗电标准：1低风，2中风，3高风 。
    primary key RID
}
```

b) table_Log

```
{
    CID int
    RID int
    TMode int default 0
    EMode int
    OccurTime Datetime //操作发生的时间
    foreign reference Rooms
    foreign reference Customers
}
```

c) table_Customers

```
{
    CID int
    Name char(10)
primary key CID
}
```