

智能家居系统设计文档

一、系统概述

随着人们对生活品质的日益提高，智能化家居已经成为了现代家庭生活中的趋势和需求。智能家居通过传感器技术和互联网的结合，实现了家庭设备的智能化控制和管理，提高了用户的便利性和舒适度。本文将介绍一个智能家居物联网仿真系统的设计和实现，包括系统架构、数据存储设计、用户界面设计和系统功能描述等。

二、系统架构

2.1 虚拟传感器模拟端

通过 Java 语言开发，使用 Maven 架构，通过随机数生成算法模拟真实环境下动态变化的温度、湿度、光照、空气质量等各类传感器生成的数据。
使用网络协议 MQTT 将模拟数据发送到后端服务器进行数据分析。

2.2 后端数据处理

选择 java 语言和 Spring Boot 框架来实现后端服务。
接收并处理来自传感器模拟端的数据。
构建消息队列服务的架构，以管理和调度数据流。
提供数据接口供前端调用，并管理数据的存储和更新。
接口如下：

请求方式：GET

请求地址：/senserData

请求参数：null

响应参数：

```
{
  dateTime: 当前日期和时间
  temperature: 当前室内温度值，单位° C
  humidity: 当前室内湿度值，单位%
  light: 当前室内光照情况，单位 lux
  airQuality: 当前室内空气质量
}
```

例如：

```
[{"dateTime":"2024-06-04T14:58:22","temperature":19.952893715810003,"humidity":48.49389155546847,"light":362.14439109852617,"airQuality":9
```

```
3. 02129217746108}, {"dateTime": "2024-06-04T14:58:27", "temperature": 20.037700681737093, "humidity": 50.46591009497797, "light": 539.5111474540657, "airQuality": 102.56673020914602}, {"dateTime": "2024-06-04T14:58:32", "temperature": 20.409839984376372, "humidity": 49.91102915794134, "light": 478.10880024410966, "airQuality": 101.98625843944495}, {"dateTime": "2024-06-04T14:58:37", "temperature": 20.26981696469334, "humidity": 51.70677947108769, "light": 384.98913798807496, "airQuality": 106.22486229765373}, {"dateTime": "2024-06-04T14:58:42", "temperature": 18.883618049218438, "humidity": 48.90716343544343, "light": 477.43707538482704, "airQuality": 94.42660339657367}]
```

2.3 前端可视化界面

使用 HTML 为主体，内嵌 JavaScript 脚本以及 Chart.js 的数据可视化库实现用户界面。从后端获取数据并展示，使用折线图根据后端获得的数据动态显示，以便用户能够直观地查看和分析数据。

三、数据存储设计

3.1 数据库

使用关系型数据库 MySQL 来存储传感器数据和其他系统相关数据。数据库结构如下：

sensormessage

🔑

dateTime: dateti...

temperature: var...

humidity: varchar...

light: varchar(255)

airQuality: varcha...

字段	索引	外键	触发器	选项	注释	SQL 预览
名						类型长度小数点不是 null 虚拟键注释
dateTime						datetime00 <input checked="" type="checkbox"/> <input type="checkbox"/> 🔑1
temperature						varchar2550 <input type="checkbox"/> <input type="checkbox"/>
humidity						varchar2550 <input type="checkbox"/> <input type="checkbox"/>
light						varchar2550 <input type="checkbox"/> <input type="checkbox"/>
airQuality						varchar2550 <input type="checkbox"/> <input type="checkbox"/>

根据需求，设计合适的数据表结构，包括传感器数据表、用户表、设备表等。

3.2 消息队列服务

使用消息队列服务 **ActiveMQ** 来管理和调度数据流。
将传感器数据通过消息队列进行异步处理，提高系统的可伸缩性和性能。

四、用户界面设计

页面布局：
采用多个折线图并行显示的方式，使用户可以直观的看到各项数据的数值以及变化趋势。



五、系统功能描述

5.1 传感器模拟端功能

模拟各类传感器生成的数据，如温度、湿度、光照强度等。
通过网络协议将模拟数据发送到后端进行数据分析。

5.2 后端数据分析功能

接收来自传感器模拟端的数据。
对数据进行处理和分析，如聚合、过滤、转换等。
将分析结果存储到数据库，并提供数据接口供前端调用。
构建消息队列服务，实现异步处理和调度数据流。

5.3 前端界面功能

从后端获取数据并展示，如图表、图形、仪表盘等。
实现用户交互功能，如数据显示、数据更新等