

《BS 体系软件设计》大程

1 实验：物联网应用网站

1.1 实验目的

任选 Web 开发技术实现一个物联网应用的网站

1.2 实验要求

需要实现的基本功能如下：

1. 搭建一个 mqtt 服务器，能够接收指定的物联网终端模拟器发送的数据。
2. 实现用户注册、登录功能，用户注册时需要填写必要的信息并验证，如用户名、密码要求在 6 字节以上，email 的格式验证，并保证用户名和 email 在系统中唯一，用户登录后可以进行以下操作。
3. 提供设备配置界面，可以创建或修改设备信息，包含必要信息，如设备 ID、设备名称等
4. 提供设备上报数据的查询统计界面
5. 提供地图界面展示设备信息，区分正常和告警信息，并可以展示历史轨迹
6. 首页提供统计信息（设备总量、在线总量、接收的数据量等），以图表方式展示（柱状体、折线图等）

增强功能：

7. 样式适配手机端，能够在手机浏览器/微信等应用内置的浏览器中友好显示。

为了提交作业方便，项目使用的数据库，建议使用 mysql 或 mangodb，提交作业时同时附带建库建表脚本文件。

2 文档要求和评分标准

独立完成。

要求界面友好，提供必要的文档，包括设计文档和使用手册等其他文档。

设计文档于 2021 年 5 月 7 日前提交，占 20%最终总评成绩。打包上传学在浙大或发送到 hu_xj@zju.edu.cn

2021 年 6 月 30 日前提交程序代码和实验报告。打包上传学在浙大或发送到 hu_xj@zju.edu.cn

1、文档包括：

- （1）实验报告封面（见下页）
- （2）设计文档
- （3）其他文档（如使用手册、测试报告、开发体会、小结等），
- （4）源代码文件

浙江大学

本科实验报告

课程名称: B/S 体系软件设计

姓 名: 程诗卓

学 院: 计算机科学与技术

系: 计算机科学与技术

专 业: 计算机科学与技术

学 号: 3180106177

指导教师: 胡晓军

2021 年 6 月 30 日

浙江大学实验报告

课程名称： B/S 体系软件设计 实验类型： 期末大程

实验项目名称： 物联网应用网站

学生姓名： 程诗卓 专业： 计算机科学与技术 学号： 3180106177

同组学生姓名： _____ 指导老师： 胡晓 1 军

实验地点： _____ 实验日期： 2021 年 6 月 20 日

目录

序言

设计文档

- 1 技术栈
- 2 网站架构
 - (1) 前端
 - ① 布局设计
 - ② 界面路由与组件逻辑
 - ③ 可视化
 - ④ 表单与网络请求
 - (2) 后端
 - ① MySQL 建表
 - ② Mqtt 消息持久化
 - ③ Spring-boot + mybatis
 - 1) Mapper
 - 2) Service
 - 3) Controller

使用说明 README

1. MySQL 建表
2. 启动 spring-boot 服务
3. 启动 iot-client
4. 启动 React 前端

测试报告

1. 登录注册
2. 地图路径标注
3. 统计信息可视化
4. 设备概览

开发小结

序言

本实验是 B/S 体系软件设计的期末作业，实现了包括注册登录、前端交互、数据可视化、数据持久化、前后端通信等功能
手机端样式：



设计文档

技术栈

- 1 React
- 2 Antd 组件库
- 3 Spring-boot
- 4 Mybatis
- 5 EMQX

网站架构

前端

- 布局设计与组件逻辑
登录页，居中表单，右上角浮动回到主页的按钮

 IoT Lord

IoT Lord是一个物联网管理系统, 前端采用React, 后端采用php,java.spring boot

* 账号

13163230059

* 密码

登录

没有账号, 去注册

github@weewenao2008
by 程停桌 3180106177
当前时间: 下午10:06:25

注册页类似

 IoT Lord

IoT Lord是一个物联网管理系统, 前端采用React, 后端采用php,java.spring boot

* 账号

13163230059

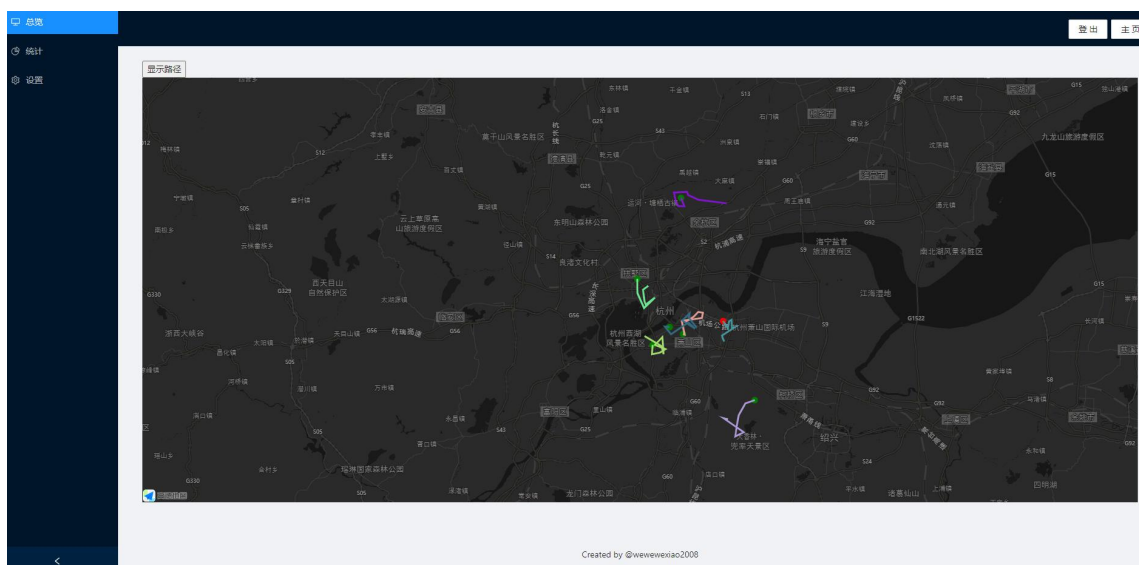
* 密码

* 邮箱

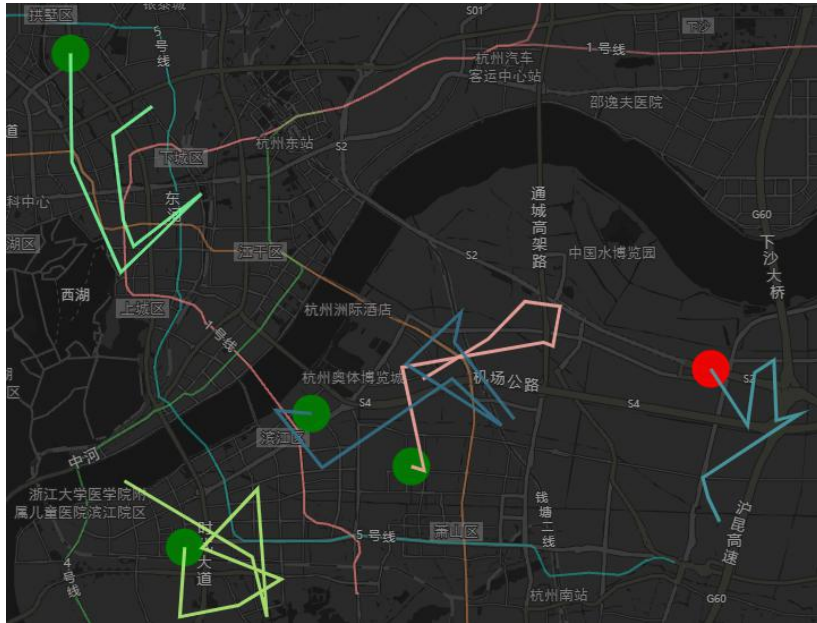
请输入邮箱

注册

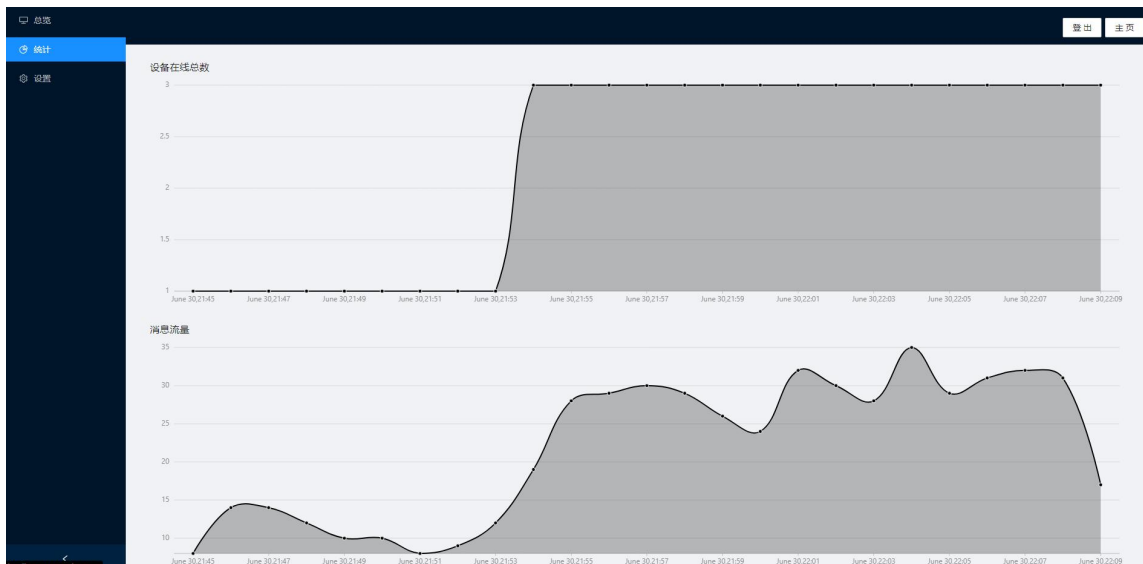
登录后的主界面左侧为导航栏, 可收缩, 右上角有登出和回到主页的按钮
地图上方的按钮点击可以显示轨迹, 再次点击可以刷新轨迹, 颜色随机, 默认显示最后 10 个地点
采用高德地图在 react 中的封装 amap 作为地图组件, 能够添加线条和圆形



圆形的颜色作为告警信息（绿、红）



设备统计页面右边两张面积表作为在线设备和消息流量的统计，需要左侧切换导航栏刷新
采用 BizCharts 作为可视化组件



设备配置页能够获得设备最后一次日志的信息，同时能在上方信息栏中填写设备 id 进行设备名和拥有者的更新，若设备 id 不存在则会报错。需要手动刷新。

clientid	name	owner	Last info	lat	ing	devicevalue	alert
device0001	asdas		Device Data 2021/06/30 22:13:00	30.115833169817932	120.05163072764873	38	0
device0002			Device Data 2021/06/30 22:13:02	30.303020580410955	120.39597558736804	36	0
device0003			Device Data 2021/06/30 22:12:51	30.073274430036538	120.12185949087153	95	0
device0004			Device Data 2021/06/30 22:12:58	30.243150631785404	120.4151162689924	76	1
device0005			Device Data 2021/06/30 22:12:57	30.47609950602054	120.28055622458463	11	0
device0006			Device Data 2021/06/30 22:13:02	30.164756188392634	120.38724620044232	49	0
device0007			Device Data 2021/06/30 22:12:52	30.28907381534576	120.21893500983714	86	0
device0008				0	0	0	0
device0009				0	0	0	0
device0010				0	0	0	0

后端

- MySQL 建表

User 表，用户名为主键，password 非空，email 唯一

```
create table users(
  username varchar(255) primary key,
  passwd varchar(255) not null,
  email varchar(255) unique);
```

Devices 表

clientid 主键，其他默认为空，alert 用于标志最后一次告警状态

```
create table devices(
  clientid varchar(255) primary key,
  devicename varchar(255) default null,
  ownername varchar(255) default null,
  alert bool
);
```

Log 表

logid 自动增长，clientid 作为外键关联 devices，logtimestamp 为 Unique 属性（这样 1 秒就只能接受一个 message，但是会避免因联合 UNIQUE 造成的数据库性能问题和奇怪的重复问题），同时 UNIQUE 可以自动添加索引

```
create table history_log(
  logid int primary key auto_increment,
  clientid varchar(255),
  logtimestamp timestamp unique,
  info varchar(255),
  lat double,
  ing double,
  devicevalue int,
  alert int,
  FOREIGN KEY (clientid) references devices(clientid)
);
```

- Mqtt 消息持久化

项目名 iotclient

IOTListener.java

成员包含 MqttClient，作为客户端订阅模拟器发布的主题

修改 MqttCallbackExtended 回调函数中接受到信息的动作为加入数据库

DBHelper.java

使用 JDBC 接口进行数据库连接与持久化

DBService.java

服务类内的函数被 mqtt 回调函数调用，生成对应的 Sql 语句，然后调用 DBHelper 执行需要执行

1. 无论设备是否存在，都进行一遍 Insert 加入到 devices 表中
2. Update 一次该 device 信息
3. 在 loghistory 表中 insert 一条 log message 数据

- Spring-boot + mybatis

项目名 iot-mybatis

使用 spring 与 mybatis 结合，分为 controller, domain, mapper, service 四个层

Domain

定义基本数据结构，需要加上 Getter、Setter

@Data

```
public class Devices {  
    private String clientid;  
    private String devicename;  
    private String ownername;  
    private Boolean alert;  
}
```

Mapper

定义映射，需要注解 @Mapper，然后再内部设置映射到的 SQL 语句

@Mapper

```
public interface DevicesMapper {  
    @Insert("INSERT INTO devices" +  
        "clientid, devicename,ownername " +  
        "VALUES" +  
        "(#{clientid},#{devicename},#{ownername})")  
    int insert(Devices device);  
}
```

Service

需要 @Service 注解，@Autowired 自动装配，分为接口和实现层，调用 Mapper 中的接口，返回 domain 中类的实例如：

@Service

```
public class LogServiceImpl implements LogService {  
    @Autowired  
    private HistorylogMapper historylogMapper;  
  
    @Override  
    public int insert(Historylog historylog) {  
        return historylogMapper.insert(historylog);  
    }  
}
```



```
}
```

Controller

需要注解`@RestController`、`@CrossOrigin`，还有`@RequestMapping`用于处理 request，`@GetMapping`用于处理前端的 get 请求，`@RequestParam`可以获得传来的参数，然后调用 Service 里面的函数，返回一个 Result 对象以 json 格式传回浏览器，如：

```
@RestController
@RequestMapping("/api/user")
@CrossOrigin
public class UserController {

    @Autowired
    private UserService userService;

    @GetMapping("/get")
    public ResultObject getUserByName(@RequestParam String username){
        User user = userService.getUserByName(username);
        ResultObject resultObject = new ResultObject(0, "success", user);
        return resultObject;
    }
}
```

README

见项目文件夹

测试报告：

见视频