

浙江大学

本科实验报告

课程名称: B/S 体系软件设计

姓 名: 肖瑞轩

学 院: 计算机科学与技术学院

系: 计算机科学与技术

专 业: 计算机科学与技术

学 号: 3180103127

指导教师: 胡晓军

2021 年 6 月 29 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: 个人上机实验

实验项目名称: 物联网应用网站运行指南

学生姓名: 肖瑞轩 专业: 计算机科学与技术 学号: 318010327

同组学生姓名: 无 指导老师: 胡晓军

实验地点: 曹西 503 实验日期: 2021 年 6 月 29 日

运行指南

3180103127 肖瑞轩

运行指南

一、Iot Client虚拟客户端

1.1 代码编译

1.2 运行

二、数据库

三、mqtt接受服务器

四、后端

五、前端

一、Iot Client虚拟客户端

1.1 代码编译

```
mvn clean package
```

1.2 运行

- 将iot.properties文件放在iotclient-1.0.0.jar同一目录
- 修改iot.properties配置
- java -jar iotclient-1.0.0.jar

二、数据库

数据库使用mysql，初始的建库脚本在create_db.mysql中，代码如下

```
1 create database myapp;
2
3 use myapp;
4
5 create table my_device
6 (
7     id          int unsigned auto_increment primary key,
8     cid         varchar(256) default '' not null,
9     name        varchar(256) default '' not null,
10    cdes         text,
11    create_time  datetime      default CURRENT_TIMESTAMP not null,
12    user         varchar(256) default '' not null
13 );
14
15 create table my_message
16 (
17     id          int unsigned auto_increment primary key,
18     alert       int           default 0          not null,
19     cid         varchar(256) default ''          not null,
```

```

20     info      varchar(256) default ''          not null,
21     lat       float      default 0.0000        not null,
22     lng       float      default 0.0000        not null,
23     timestamp datetime    default CURRENT_TIMESTAMP not null on update CURRENT_TIMESTAMP,
24     value     int         default 0             not null
25 )charset = utf8;
26
27 create table user_info
28 (
29     id        int unsigned auto_increment      primary key,
30     name      varchar(256) default '' not null,
31     password  varchar(256) default '' not null,
32     email     varchar(256) default '' not null,
33     gender    varchar(256) default '' not null,
34     birthday  varchar(256) default '' not null
35 );

```

后端中的数据库引擎部分也需要改为自己的密码与对应的端口

```

1 engine = create_engine("mysql+pymysql://(用户名):(密码)@localhost:3306/myapp", encoding="utf-8")

```

三、mqtt接受服务器

mqtt接受服务采用mosquitto服务，下载mosquitto并开启服务，在backend文件夹下，运行指令

```

1 python mymqttserver.py

```

来接收Iot client虚拟客户端发送的消息并保存到数据库中

四、后端

后端使用flask框架进入backend文件夹，运行对应的文件即可。

```

1 cd backend
2 python mybackendserver.py

```

运行在localhost:5000上

五、前端

进入frontend文件夹，之后用npm install npm start进行构建就可以了。

```

1 cd frontend
2 npm install
3 npm start

```

运行在localhost:3000，通过设置proxy与axios进行跨域访问后端端口

若全部过程均没有报错，之后在浏览器中打开<http://localhost:3000>就可以对物联网平台进行访问啦！