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api简单说明

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

后端api简单说明

3180103127 肖瑞轩

/get30Message：得到最近的30条消息

```python
@app.route('/get30Message', methods=['GET'])
def get30Message():
    messages = Message.query.all()[-30:]
    result = {
        "code": 0,
        "data": []
    }
    for message in messages:
            mess = {
                "cid": message.cid,
                "alert": message.alert,
                "info": message.info,
                "lat": message.lat,
                "lng": message.lng,
                "timestamp": message.timestamp,
                "value": message.value
            }
            result.get("data").append(mess)
    print(result)
    return result
```

/getMessagebyCid：根据设备id得到最近的该设备的50条消息

```python
@app.route('/getMessagebyCid', methods=['GET'])
def getMessagebyCid():
    Id = request.args.get("Cid")
    print(Id)
    messages = Message.query.filter(Message.cid==Id).all()[-50:]
    result = {
        "code": 0,
        "data": []
    }
    if (len(messages)==0):
        return jsonify(code=-1, msg="设备不存在!")
    for message in messages:
        tmpmessage = {
            "alert": message.alert,
            "info": message.info,
            "lat": message.lat,
            "lng": message.lng,
            "timestamp": message.timestamp,
            "value": message.value
        }
        result.get("data").append(tmpmessage)
    print(result)
    return result
```

/getUserbyEmail：通过邮箱得到用户信息

```
1   @app.route('/getUserbyEmail', methods=['GET'])
2   def getUserbyEmail():
3       email = request.args.get("Email")
4       print(email)
5       user = User.query.filter(User.email == email).all()
6       if user is None:
7           code = -1
8           msg = "Not exist the user!"
9           return jsonify(code=code, msg=msg)
10      code = 0
11      msg = "get user success!"
12      return jsonify(code=code, msg=msg, data=user[0].name)
```

/editPass：修改个人信息

```
1   @app.route('/editPass', methods=['POST'])
2   def editPass():
3       data = json.loads(request.data)
4       email= data["email"]
5       new_name = data["name"]
6       new_passwd = data["passwd"]
7       print(email)
8       print(new_name)
9       print(new_passwd)
10      tmp = User.query.filter(User.email == email).all()
11      print(tmp)
12      old_name=tmp[0].name
13      tmp[0].name=new_name
14      tmp[0].password=new_passwd
15      devices = Device.query.filter(Device.user == old_name).all()
16      for device in devices:
17          device.user = new_name
18      db.session.commit()
19      return jsonify(code=0, msg="更改个人信息成功!")
```

/register：注册方法

```
1   @app.route('/register', methods=['POST'])
2   def register():
3       data = json.loads(request.data)
4       name = data['name']
5       password = data['passwd']
6       email = data['email']
7       gender = data['gender']
8       birthday = data['birthday']
9       print(gender,birthday)
10      result = {
11          "code": 0,
12          "msg": "Register success!"
13      }
```

```
14        new_id = 0
15        users = User.query.all()
16        for user in users:
17            if name == user.name or email == user.email:
18                result['code'] = -1
19                result['msg'] = "邮箱或用户名已经存在!"
20                return result
21            new_id = user.id
22        new_user = User(id=new_id + 1, name=name, password=password,
     email=email,gender=gender,birthday=birthday)
23        db.session.add(new_user)
24        db.session.commit()
25        return result
```

/login：登录方法

```
1  @app.route('/login', methods=['POST'])
2  def login():
3      data = json.loads(request.data)
4      password = data["password"]
5      email = data["email"]
6      print(email, password)
7      code = 0
8      msg = "login success!"
9      user = User.query.filter(User.email == email).all()
10     if user is None:
11         code = -1
12         msg = "用户不存在!"
13         return jsonify(code=code, msg=msg)
14     print(user[0].password)
15     if user[0].password != password:
16         code = -2
17         msg = "密码错误!"
18         return jsonify(code=code, msg=msg)
19     token_back = mytoken.create_token(user[0].id)
20     print(token_back)
21     print(type(token_back))
22     return jsonify(code=code, msg=msg, data=token_back)
```

/getDeviceByUser：得到用户拥有的所有设备

```
1  @app.route('/getDeviceByUser', methods=['GET'])
2  def getDeviceByUser():
3      username = request.args.get("username")
4      devices =  Device.query.filter(Device.user == username).all()
5      result = {
6          "code": 0,
7          "data": []
8      }
9      for device in devices:
10         dev = {
11             "id": device.id,
12             "Cname": device.cid,
13             "Cid": device.name,
14             "CDes": device.cdes,
15             "Ctime": device.create_time,
```

```
16            "user": device.user,
17        }
18        result.get("data").append(dev)
19        print(dev)
20    return result
```

/getDevice：得到所有设备

```
1  @app.route('/getDevice', methods=['GET'])
2  def getDevice():
3      devices = Device.query.all()
4      result = {
5          "code": 0,
6          "data": []
7      }
8      for device in devices:
9          dev = {
10             "id": device.id,
11             "code": device.cid,
12             "name": device.name,
13             "description": device.cdes,
14             "create_time": device.create_time,
15             "user": device.user,
16         }
17         result.get("data").append(dev)
18         print(dev)
19     return result
```

/getRecentDevice:得到最近新建设备

```
1  @app.route('/getRecentDevice', methods=['GET'])
2  def getRecentDevice():
3      time_index = []
4      count = [0, 0, 0, 0, 0, 0, 0,0,0,0]
5      today = datetime.datetime.today()
6      for i in range(0, 10):
7          daytmp = today - datetime.timedelta(days=i)
8          dayt = daytmp.replace(hour=0, minute=0, second=0, microsecond=0)
9          dayt = datetime.datetime.date(dayt)
10         time_index.append(str(dayt)[5:])
11     time_index.reverse()
12     devices = Device.query.all()
13     for device in devices:
14         device_day = str(datetime.datetime.date(device.create_time))[5:]
15         for i in range(len(time_index)):
16             if time_index[i] == device_day:
17                 count[i] += 1
18
19     print(time_index, count)
20     return jsonify(code=0, msg="getRDsuccess!", day=time_index, count=count)
```

//editDevice：编辑设备

```
1  @app.route('/editDevice', methods=['POST'])
2  def alterDevice():
3      data = json.loads(request.data)
4      Cid = data["Cid"]
5      Cname = data["Cname"]
```

```
6         Cdes = data["Cdes"]
7         username = data["user"]
8         device = Device.query.filter(Device.name == Cid).filter(Device.user==username).all()
9         if(len(device)==0):
10            return jsonify(code=-1, msg="修改失败!")
11        device[0].cid = Cname
12        device[0].name = Cid
13        device[0].cdes = Cdes
14        db.session.commit()
15        return jsonify(code=0, msg="修改成功!")
```

/createDevice：新建设备

```
1   @app.route('/createDevice', methods=['POST'])
2   def createDevice():
3       data = json.loads(request.data)
4       Cid = data["Cid"]
5       CName = data["Cname"]
6       CDescription = data["Cdes"]
7       CUser = data["user"]
8       result = {
9           "code": 0,
10          "msg": "create success!"
11      }
12      new_id = 0
13      devices = Device.query.all()
14      for device in devices:
15          if Cid == device.name:
16              result['code'] = -1
17              result['msg'] = "出现了重名!"
18              return result
19          new_id = device.id
20      new_device = Device(id=new_id + 1, cid=CName, name=Cid, cdes=CDescription,
21                          create_time=datetime.datetime.now(), user=CUser)
22      db.session.add(new_device)
23      db.session.commit()
24      return result
```

其他更多api接口详见源代码。