流动警务站监控

项目文档

第三项目组

2019年9月11日

目录

[1.引言 2](#_Toc12142)

[1.1 目的 2](#_Toc31040)

[1.2 项目背景 2](#_Toc26828)

[2. 任务概述 3](#_Toc27856)

[2.1 系统业务流程图 3](#_Toc31677)

[3.需求分析 4](#_Toc1340)

[3.1 内部需求 4](#_Toc28761)

[3.2 用户需求 4](#_Toc22760)

[3.3 模块划分 4](#_Toc8056)

[4. 系统结构 5](#_Toc4419)

[5. 系统功能模块详细设计 7](#_Toc22749)

[5.1 登录模块 7](#_Toc8913)

[5.2 地图显示模块 7](#_Toc15265)

[5.3 排班表模块 7](#_Toc16989)

[5.4 打卡模块 7](#_Toc23312)

[5.5 搜索模块 7](#_Toc26864)

[6.数据字典 8](#_Toc23384)

[7. 数据库设计 9](#_Toc1083)

[7.1 数据库表结构 9](#_Toc19635)

[8. 接口设计 10](#_Toc8085)

[8.1 外部接口 10](#_Toc253)

[8.2 内部接口 10](#_Toc29127)

[9. 页面设计 11](#_Toc14033)

[9.1 登录页面 11](#_Toc24110)

[9.2 注册页面 11](#_Toc3737)

[9.3 主页与打卡 12](#_Toc11264)

[9.4 搜索页面 13](#_Toc8517)

[10. 项目功能测试 14](#_Toc3259)

[10.1 测试环境 14](#_Toc3450)

[10.2 功能测试 14](#_Toc4869)

# 1.引言

## 1.1 目的

开发一个针对警员的，具有打卡、排班功能的流动警务站监控系统。基于Node.js和MYSQL等技术开发此系统，通过本次开发练习运用所学技术。

## 1.2 项目背景

开发人员表1.2

|  |  |  |
| --- | --- | --- |
| 角色 | 组员 | 备注 |
| 项目经理 | 王文金 |  |
| 后端 | 王程鑫 王文金 |  |
| 前端 | 杨昊 施一航 |  |
| UI | 刘伟业 |  |
| 文档/测试 | 雷雪煜 |  |

# 任务概述

开发一个可以查看执勤信息、执勤情况的登录打卡系统。使用Node.js、Jquery和数据库等技术进行开发搭建，并从中获得项目管理的经验，运用学过的技术，也是发起本项目的一个主要原因。

## 2.1 系统业务流程图

否

是

登出

打卡失败，请前往此区域

打卡成功

显示地图、排班表

登录/注册

打卡

是否到达地点

否

# 3.需求分析

## 3.1 内部需求

**3.1.1 注册和登录**

用户查看信息必须进行注册和登录。

**3.1.2 查看信息**

根据用户需要可以查询自己的排班信息，展示排班表，可以按时间查询当日执勤人员信息。

**3.1.3 打卡系统**

点击打卡之后，如果用户在执勤地点内则显示打卡成功，如果不在执勤地点则显示请立即前往执勤地。

## 3.2 用户需求

用户需要一个快速注册和登录的页面，登录完成后方便查询用户的排班信息，了解执勤地点和时间。在点击打卡之后，如果用户在执勤地点内则显示打卡成功，如果不在执勤地点则显示请立即前往执勤地。

## 3.3 模块划分

**3.2.1 用户注册和登录**

点击进入网站首页，在页面上显示有用户登录入口，每个登录的用户都应先注册成功并登录之后才能查看具体的排班信息等等。如果已经注册过的用户，直接输入账号密码进行登录即可。

**3.2.2 地图显示**

登录成功后，进入首页，显示出执勤地点及其周边的地图，在用户进行打卡之后，地图上会显示出执勤情况。

**3.2.3 排班管理**

用户可以查看自己的排班情况，也可以按日期查询排班情况。排班表将显示执勤的日期、人员名称、执勤地点还有执勤人员的打卡情况。用户无法修改排班表。

**3.2.4 打卡系统**

用户查看自己排班信息之后，前往执勤地点，如果不在执勤地打卡失败，请立即前往执勤地点；只有在执勤点打卡才能成功记录，并且地图上会弹出显示打卡成功。

**3.2.5 搜索功能/查看功能**

用户登录成功之后进入主页，可以输入警员名和日期点击搜索按钮进行搜索查找到需要的信息，搜索到的警员信息在排班表里进行高亮。

# 系统结构

系统详细结构图如下图4.1 详细系统结构图所示；

流动警务站监控系统

地图显示

排班表

打卡系统

岗岗位显示

岗执勤情况

岗执勤日期

岗执勤人员

岗执勤地点

岗打卡情况

岗到岗情况

图4.1 详细系统结构图

地图实体属性图如下图4.2 地图属性图所示；

地图

执勤位置

执勤情况

图4.2 地图属性图

排班实体属性图如下图4.3 排班表属性图所示；

排班表

执勤日期

执勤人员

执勤地点

打卡情况

图4.3 排班表属性图

打卡实体属性图如下图4.4 打卡属性图所示。

打卡系统

到岗情况

图4.4 打卡属性图

# 系统功能模块详细设计

## 5.1 登录模块

1. 功能：实现登录功能，输入用户名称和用户密码，正确则进入地图页，不正确则返回原来的页面；
2. 性能：由登出功能调用；
3. 输入项目：用户名称和用户密码；
4. 输出项目：正确则进入地图页，不正确则返回原来的页面。

## 5.2 地图显示模块

1. 功能：为用户显示执勤岗位信息和执勤情况查询功能；
2. 性能：为用户提供执勤查看功能；
3. 输入项目：执勤人员；
4. 输出项目：执勤岗位信息和执勤情况。

## 5.3 排班表模块

1. 功能：为用户显示个人排班情况及所有排班情况；
2. 性能：为用户提供信息查询功能；
3. 输入项目：执勤人员或日期；
4. 输出项目：执勤日期、执勤人员、执勤地点和打卡情况。

## 5.4 打卡模块

1. 功能：实现考勤信息增加功能；
2. 性能：为用户提供考勤信息录入；
3. 输入项目：打卡人员信息；
4. 输出项目：打卡成功或失败，失败请立即前往执勤地。

## 5.5 搜索模块

1. 功能：搜索查询用户所需信息；
2. 性能：为用户提供查询功能；
3. 输入项目：警员名和日期；
4. 输出项目：搜索的警员的相关信息。

# 6.数据字典

**USER表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 中文名称 | 数据类型 | 允许非空 | 自动递增 | 备注 |
| user\_id | 用户ID | int | N | Y |  |
| user\_name | 用户名 | varchar（40） | N |  |  |
| user\_password | 用户密码 | varchar（40） | N |  |  |
| user\_desc | 用户描述 | varchar（40） |  |  |  |
| cop\_site\_x | 坐标 | int |  |  |  |
| cop\_site\_y | 坐标 | int |  |  |  |

数据文件：警员信息表

文件组成：用户ID,用户名,用户密码,用户描述,日期

数 据 项：用户ID

数据类型：整型

数据长度：8

数 据 项：用户名

数据类型：可变长度字符串

数据长度：40

数 据 项：用户密码

数据类型：可变长度字符串

数据长度：40

数 据 项：用户描述

数据类型：可变长度字符串

数据长度：40

数 据 项：x轴坐标

数据类型：整型

数据长度：40

数 据 项：y轴坐标

数据类型：整型

数据长度：40

数据流定义：

注册

名    称：用户注册

简    述：记录用户信息

数据来源：用户

数据去向：警员信息表

数据组成：用户ID+用户名+用户密码+用户描述

打卡

名    称：用户打卡

简    述：用户到达指定位置后点击打卡

数据来源：用户

数据去向：用户打卡情况

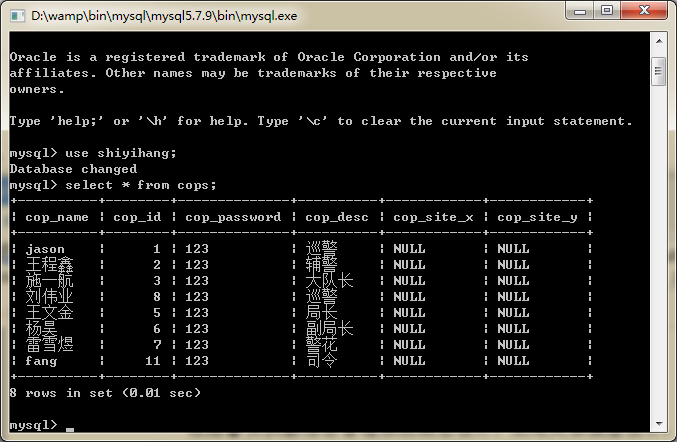
# 数据库设计

## 7.1 数据库表结构

根据警员执勤信息管理系统概念设计，采用关系模型设计，系统包括三个实体对象，一个关系表，并使用存储过程和函数，触发器等内容对数据库进行操作。

根据用户属性，user表包含四个字段，其字段名、属性、类型和约束如表7.1 user表所示。用户对象用用户ID做主键。

**USER表**



|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 属性 | 数据类型 | 约束 |
| cop\_id | 用户ID | int | Not NULL |
| cop\_name | 用户名 | varchar（40） | Not NULL |
| cop\_password | 用户密码 | varchar（40） | Not NULL |
| cop\_desc | 用户描述 | varchar（40） |  |
| cop\_site\_x | 坐标 | int |  |
| cop\_site\_y | 坐标 | int |  |

**表7.1 user表**

# 接口设计

## 8.1 外部接口

（1）用户注册接口：

输入：用户注册账号的信息

输出：添加用户信息到数据库的USER表

1. 用户登录接口：

输入：用户登录账号信息

输出：登录成功时，跳转到地图页面

1. 用户登出接口：

输出：点击注销按钮，跳转到登录页面

## 8.2 内部接口

1. 排班表接口：

输入：从数据库调出警员信息，进行排班

输出：显示排班情况到页面上

1. 打卡系统接口：

输入：获取打卡人员信息

输出：打卡成功或打卡失败，请立即前往执勤地点

8.2.1用户登录接口:

|  |  |  |  |
| --- | --- | --- | --- |
| 接口名称 | 参数 | 返回值 | 备注 |
| /getUserInfo | {userId} | {userName, userAge, userXX:[]} | get/json/noCach/synyse  必填项:[] |
| /getUserInfo | {userName, userAge, userXX:[]} | {boolean} | post/json/noCach/synyse  必填项:[] |
| /user/info/get |  |  |  |
| /user/info/post |  |  |  |
| /user/auther |  |  |  |

# 页面设计

## 9.1 登录页面

用户登录页面如下图9.1 登录所示，输入账号密码点击登录按钮即可登录；

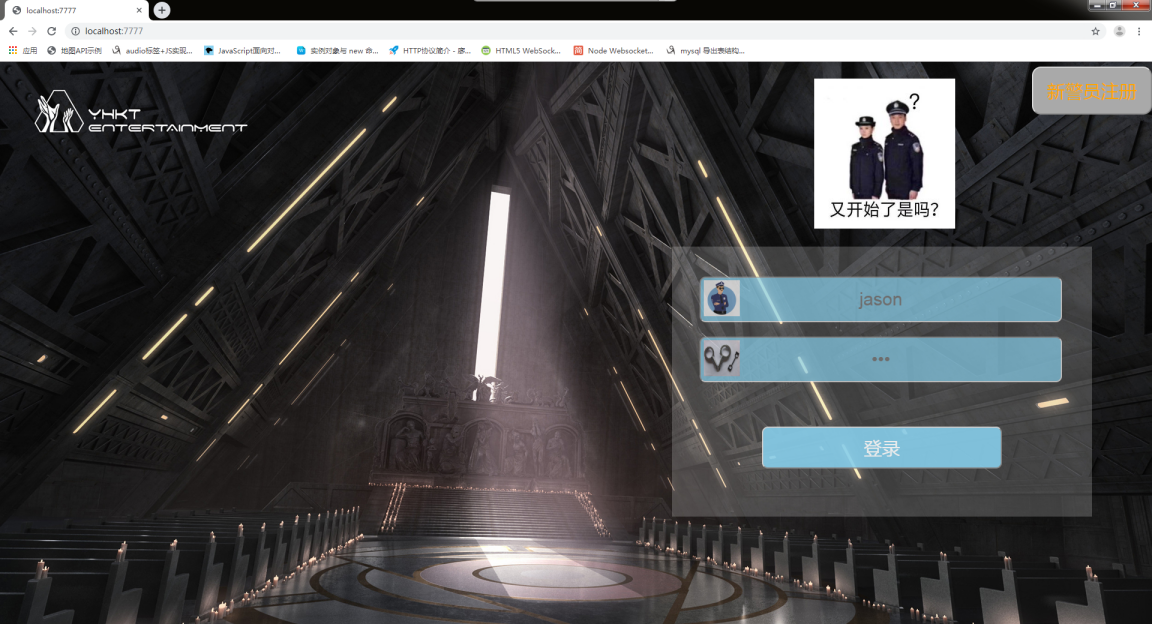


图9.1 登录

## 9.2 注册页面

点击注册按钮进入注册页面，输入警员姓名和密码即可注册，注册成功后跳转到主页，注册页面如下图9.2 注册页面所示；

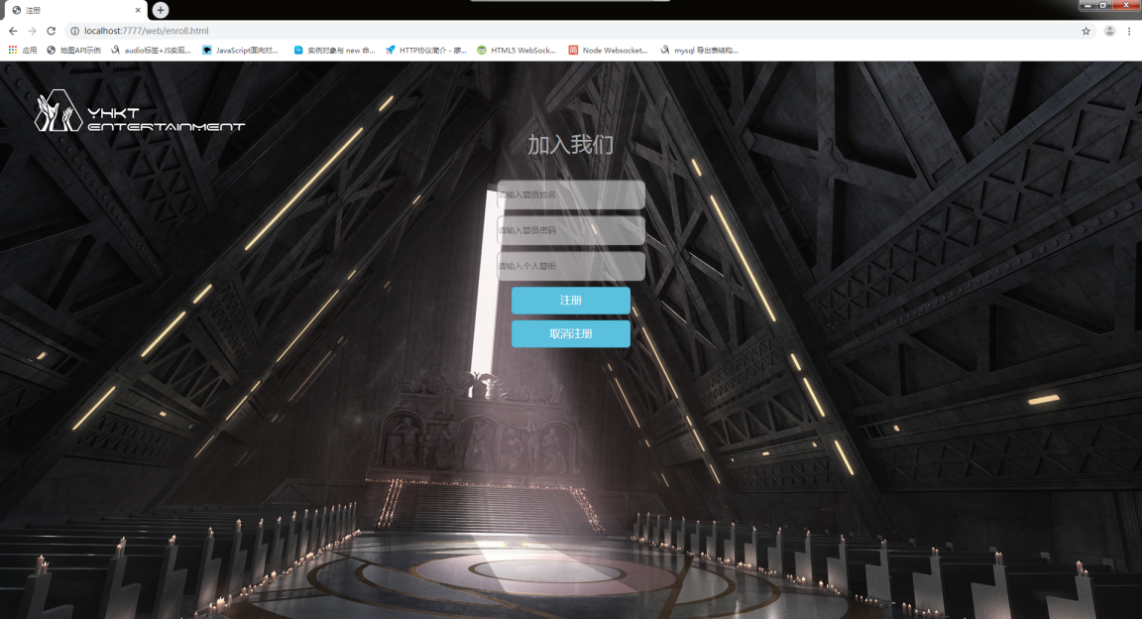


图9.2 注册页面

## 9.3 主页与打卡

登录完成后，进入主页面，主页面显示有地图和打卡按钮，用户查看信息后到达执勤地时可以打卡，如不在执勤地则弹窗显示请立即前往执勤地，之后再进行打卡。首页展示如图9.3.1 主页所示，打卡功能如图9.3.2 打卡功能所示；

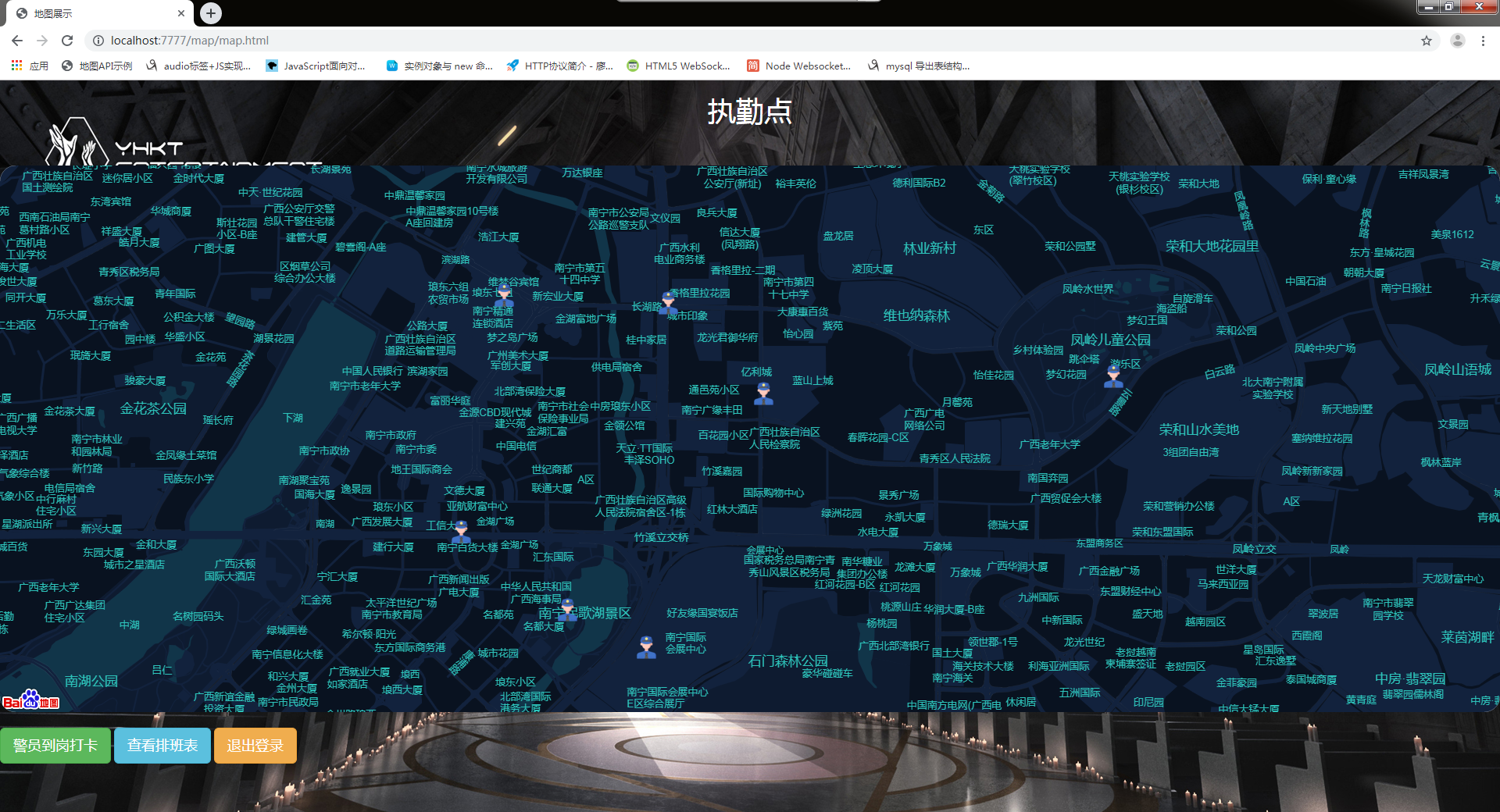


图9.3.1 主页

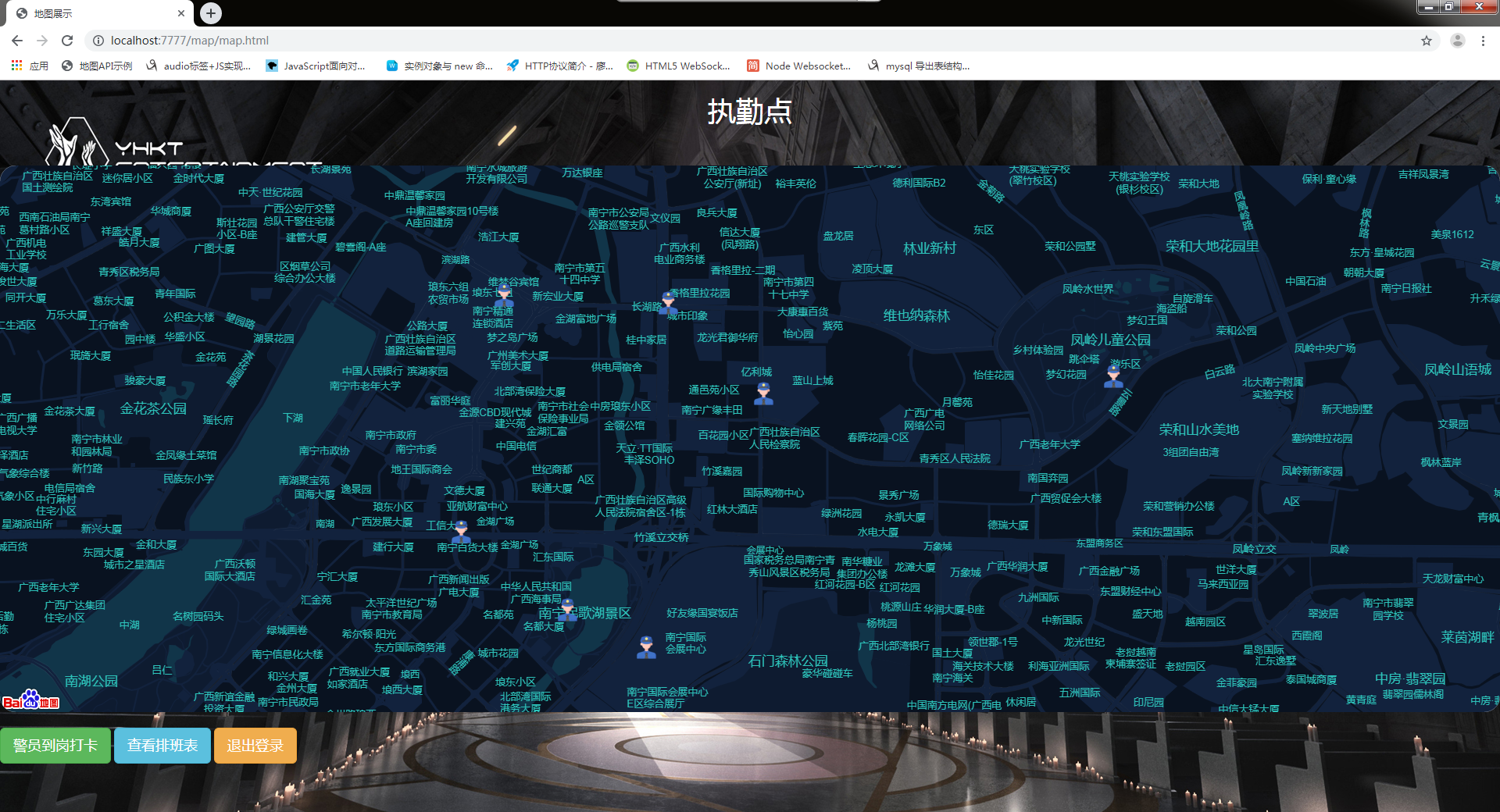


图9.3.2 打卡功能

## 9.4 搜索页面

用户进入主页后显示搜索栏，用户输入需要查找的警员名和日期即可查找到相关人员信息，搜索页面如下图9.4.1 搜索所示。

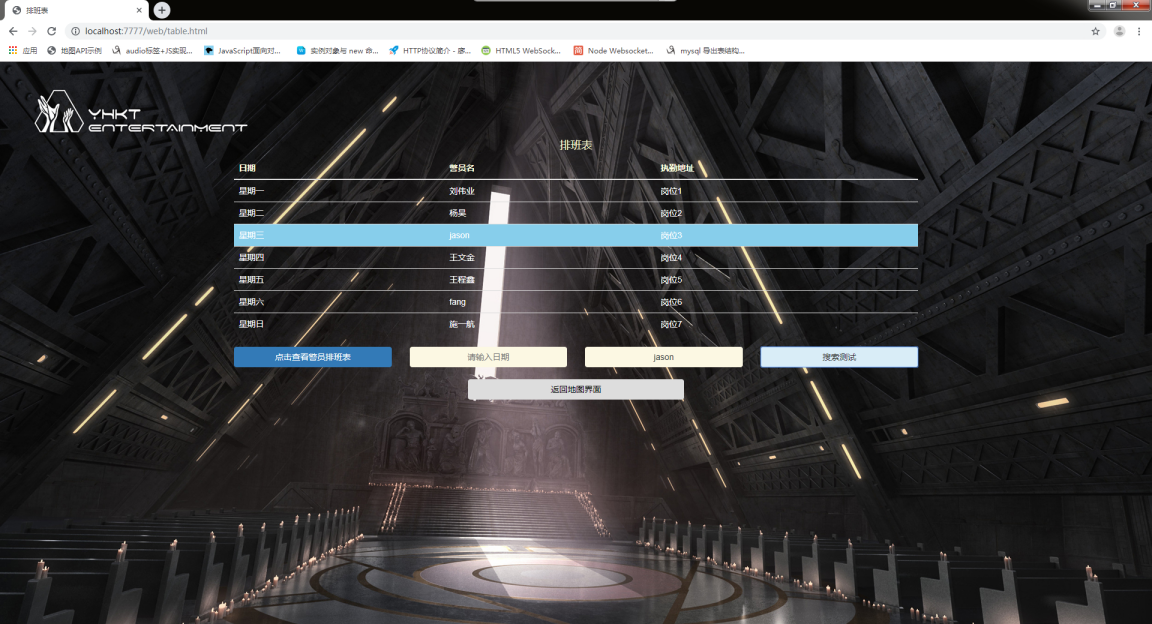


图9.4.1 搜索

# 项目功能测试

## 10.1 测试环境

wamp自带MYSQL数据库

Google Chrome浏览器

## 10.2 功能测试

**10.2.1模块功能测试**

测试方案：按照流程先进行注册或者登录之后进入主页，查看各种排班信息后完成打卡；

测试结果：经过反复测试之后，页面跳转流畅，各模块功能可以正常运行。

**10.2.2数据与数据库**

测试方案：在登录或者注册页面新注册用户，查看新用户信息是否进入数据库记录；

测试结果：经测试后，新用户数据能成功进入数据库。

**10.2.3 代码强度测试**

1. bug记录
2. 注册可以相同用户名
3. 点击排班表按钮时，重复快速点击会无限自增表格
4. 排班表排序重复问题
5. 排班表的执勤岗位无法正常显示
6. 登录密码错误也能进入主页面
7. 登录之后用户无法查看自己的账号等信息
8. 解决方案

解决A方案：在注册模块进行重名容错解决；

解决E方案：如果密码输入错误则返回登录页

解决F方案：在页面上显示自己的账号名等信息