**二维码签到系统文档1.0.0**

**一、开发文档**

**数据库表及其关系**

上课时段 starttime

编号 sid

上课开始时间 stime

老师编号 tid

老师出勤状态 iswork

请假记录 qingjiarecord

编号 qid

学号 sno

事由 qshiyou(0表示事假1表示病假)

日期 qdate

是否审核 qisread (1表示未读 0表示已读)

审核结果 qresult (0 同意 1不同意 )

请假老师编号 tid

消息记录 messagerecord

编号 mid (唯一)

消息主体 mcontent

日期 mdate

发送者 mfronid

接收者 mtoid

是否阅读 misread

签到记录 qdrecord

编号 qid (唯一)

学号 sno

签到日期 qdate

签到时间 qtime

上课时间 qstarttime

缺勤状态 qteachermsg (有则表示缺勤)

系统账号 account

登录的手机号 sphone

登录的密码 spass

最后一次登录日期 slastlogtime

数据字典 dictionary

编号 did (唯一)

学院名称 dxyname

系名称 dxiname

专业名称 dzyname

年级 dnjname

班级名称 dbjname

学生 student

学号 sno （唯一）

姓名 sname

性别 ssex

电话 sphone （唯一）

密码 spassword

班级编号 classno

老师班级 teacherclass

编号 tcid

老师编号 tid

字典编号 did

老师 teacher

编号tid （唯一）

电话tphone（唯一）

姓名tname

性别tsex

密码tpassword

**前后台组成结构图**

Html页面

页面中的效果用css和JavaScript实现

其中的操作流程也是css和JavaScript结合实现

Servlet 逻辑、应用、服务。。。

数据 这里是数据库存储中的数据

一般数据存储在内存、硬盘。

**二、使用文档**

**页面分析**

二维码签到系统主要有以下几个页面：

1. index.jsp

该系统的登录页面。输入账号和密码，选择管理员或者老师进行登录。

功能模块：忘记密码和记住密码

（1）忘记密码

这里设计的是短信验证机制。输入相应的手机号，也就是系统账号。系统后台会向这个手机号码发送一个验证码。这个验证码是随机生成的，在一定时间内有效，超时则失效。用户输入验证码，系统进行匹配。成功后就要求用户输入修改后的密码，两次密码输入相同则会对该用户数据库中的密码信息进行修改。

这里关于密码，应该设计一定的规则，限制长度，必须字母和数字组合。大小写不区分等等。然后需要对密码加密，数据库存储的是加密之后的密码信息。这样保证用户信息安全。

（2）记住密码

暂时先不做。和忘记密码同时做界面会协调一点。

1. leader.jsp

系统账号是领导的账号，且成功登录的用户则会进入这个页面。这个是领导管理老师的操作界面。

左侧菜单选项分别是：数据统计、教师管理和设置

1. 数据统计

这里采用图表插件，从后台获取数据进行绘制图表显示数据。

查询某学院某系某专业某班级的人数

1. 教师管理

教师管理模块细分为两个功能

添加老师信息和出勤记录的查询

首先是添加老师信息：默认而言，领导管理老师，老师管理学生。领导通过老师管理学生。添加老师信息这里需要输入老师的信息有：姓名、性别、电话、密码、选择老师要代课的班级，

选择代课班级示意图：



这里需要考虑的问题是：多选之后不能重复选择某一个班级。所以还需要加上条件限制。

代课班级信息是从字典信息查询而来。正确的流程顺序是从设置开始的。不过，在系统投入使用之前，数据库中的数据字典表的数据应该是包含目前该学校所有的从学院、系、专业、年级、班级的数据。在系统投入使用中进一步完善，使系统能够很好的适用与该学校。

保存老师信息后则会将这条数据添加至数据库。老师就可以用账号和密码登录系统，完成他所对应的操作。

然后是出勤记录的查询：

根据下拉列表选择老师，查询该老师的出勤记录

这里的功能做的很不细致。而且这块在系统设计之初并没有详细的考虑。未来

的规划是查询该老师一个月的出勤情况，根据条件限制查询。比如只查询老师的缺勤记录。

1. 设置

设置是三个模块功能中比较重要的一个环节。

设置需要进行的操作有：数据字典(专业班级)数据的添加，上课时段的安排，发送消息。其中最主要的是前面两个操作。

数据字典数据的添加：

示意图如下



这里的数据信息之间应该有级联关系。即选择了某一个学院之后，只能选择该学院之下的某个系，以此类推直到班级。添加一个专业班级信息后应该也要能够修改和删除。所以这里 还需加上这两个按钮。

下面是老师上课时段的安排



选择老师，点击表格中周日到周六的时间表格，选择时间给老师安排。这里不理会是什么课。而是只考虑在这个时间点上老师有课。这里的时间应该是从开学到学期末结束。将一个礼拜的课程时间安排好了之后，以此类推，下个礼拜还是如此。如果遇到放假或者其他情况，则应修改这个时间的上课安排。这个也是在最后在做。

后续思考：

上面的逻辑似乎是没有问题，但是好像代码量有点多，况且这个功能并不是这个系统最主要的功能。换一种思考，只是安排这个老师一个礼拜中有哪些时段有课，关于iswork字段换成这个老师的总学时，就是这个老师这个学期带的课的学时之和。每次上课前有个签到，不仅是学生自己签到，还有老师自己手动签到。每次老师展示二维码让学生签到，自己完成这次操作后签到，这样有个问题，就容易让老师作弊签到。就像是考试之前就已经知道考试的试卷的内容了。所以，仔细分析这个漏洞还是不能用代码避免的。想到这里，逻辑不通，代码就不用写了。如果有更好的思路，再继续完善。

发送消息这块，最开始是设计是发送邮件给某个老师。但是需要知道老师的邮箱。这个明显不现实，以抛弃。如果是发送消息，则是选择想要发送消息的老师，然后输入消息存储在数据库中，接收消息的老师登录系统之后，会有消息提示，然后阅读消息。不过这样不好。还不如直接电话联系老师或者利用社交工具QQ或者微信比较好一点。所以这个功能不做的几率比较大。

1. teacher.jsp

接下来是老师页面的分析。

左侧菜单是设置和相应代课班级。

代课班级是查询数据库老师的代课信息查询而来。

首先是设置

设置这里有上课时间、服务器地址、二维码接口地址三个苏剧需要选择或者填写。

之后就点击代课的班级，生成二维码图片供学生扫描签到。

这个又有三个选项：考勤、出勤记录、消息。

考勤则是显示生成的二维码图片。下方实时显示签到的人数。

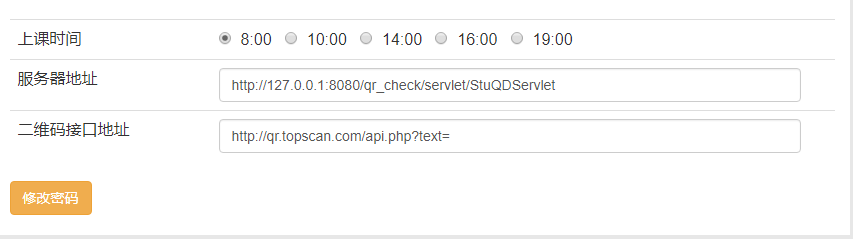
出勤记录

显示学生的出勤情况。可以更具时间日期进行筛选查询。也可以到处excel文件。

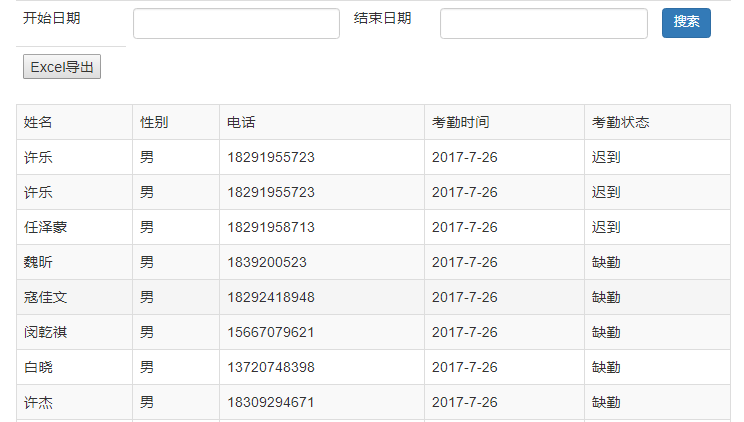
消息

这里是学生需要请假，发送给代课老师请假的原因，老师查看这个消息后，操作选择同意或者不同意，该学生本堂课的出勤记录进行相应的修改。





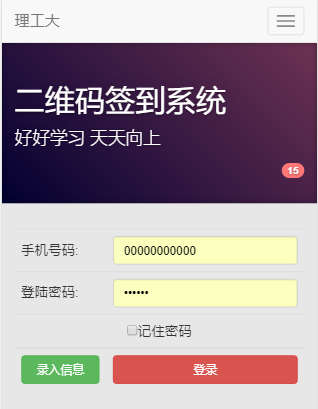






1. studentPhone.jsp

扫描二维码之后，如果已经的话，则会自动签到。否则需要登录后才能够签到。登陆之前，必须要有账号才可以，这里就需要在studentAdd页面内操作。



1. studentAdd.jsp

学生个人输入信息进行注册。这里需要注意，电话号码需要加上唯一的限制。否则出现同一个号码，不同的人。还要不能重复注册。



1. studentLogin.jsp

输入账号和密码登录。信息在cookie中存储。下次登录则不再需要重新输入账号密码登录。



**二维码签到系统文档1.0.1**

目前这个项目系统总体上是没有什么问题的。但是如果添加新的功能模块的时候，很难进一步开发，而且开发难度到后期是越来越大，代码量增加，模块划分不明确，代码混杂，这就导致现有的系统并不能随着时间的改变而有所调整。所幸现在的系统并不是很复杂。现在打算重新选择技术重新构建这个系统项目。

首先：划分移动端，PC端。

PC端则是：管理页面，打算采用vue书写。加上移动端适配。Flex布局

React-Native:简易的查询 APP

后台则是Java，提供接口。

如果还是原先的思路，基本上也就是CRUD的思路 数据交互变化实现一些逻辑。

技术难点基本上没有难度，重构过程中，发现这个问题，而且，也就是重新用一门技术实现原来的流程的过程。由于时间很紧张，决定将主要的功能实现。

首先是登录的功能模块

关注的重点是：如何加密，安全验证的问题，并发性，错误类型判断，设计。

其次是生成二维码的功能模块

之前是后端实现二维码图片的生成，现在采用前端库实现。

生成的图片的分辨率的问题。假设一张图片被放大之后，如何解决扫描的问题。

再之是简易APP的上线

之前的想法有点问题，执行力不够，打算使建立简单雏形之后，日后再不断完善。

最后的优化，交给后面再思考。

变动：又node开启的服务创建api接口。

总之，做就要做到个人的极致。

试着用工程化的角度思考问题。比如搬砖这种劳动任务很明确，至于搬砖的方式，可以很粗糙，也可以很优雅。一个技术点掌握之后，再次做这种事情无非会使自己对这个知识很熟悉，很有经验。能够很快解决某些问题。

个人理解，工作也就是重复性的一些工作。换个名字，本质应该还是那些东西。