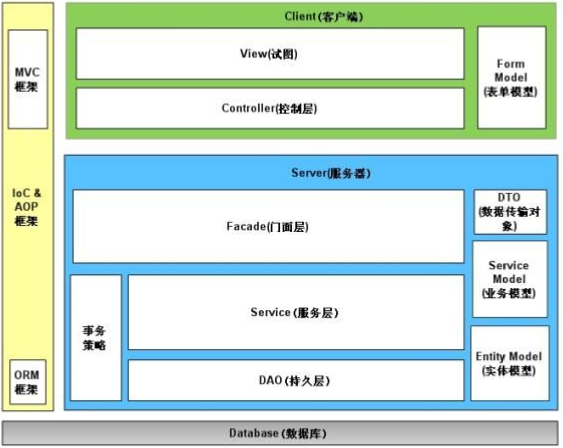
**开发技术说明**

1. 架构图



到云

学生

任课老师

教务管理员

查看本人基本信息

在线签到

查看上课签到信息

查

看

个人信

息

和

修

改

个

人

密

码

管理所教班级学生的签到信息

查看所教班级学生签到信息

管理学校院系年级专业班级的添加、删除和修 改

更新每学期每个班级课程及任课老师的安排

查看学生上课签到信息

管理系统的签到信息

二、数据字典

(1)用户表daoyunuser

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **字段名称** | **字段类型** | **允许空** | **字段描述** |
| 1 | id | int | N | 编号 |
| 2 | name | VARCHAR(64) | N | 姓名 |
| 3 | avatar | VARCHAR(64) | Y | 昵称 |
| 4 | userid | VARCHAR(64) | Y | 用户id |
| 5 | email | VARCHAR(64) | Y | 电子邮箱 |
| 6 | signature | VARCHAR(64) | Y | 签名 |
| 7 | gender | VARCHAR(64) | Y | 性别 |
| 8 | school | VARCHAR(64) | Y | 学校 |
| 9 | education | VARCHAR(64) | Y | 学历 |
| 10 | scholarship | VARCHAR(64) | Y | 学院 |
| 11 | address | VARCHAR(64) | Y | 地址 |
| 12 | phone | VARCHAR(64) | N | 电话号码 |
| 13 | password | VARCHAR(64) | N | 密码 |
| 14 | role\_id | int | N | 角色id |
| 15 | status | VARCHAR(64) | Y | 状态 |
| 16 | geographic | VARCHAR(64) | Y | 位置 |
| 17 | tags | VARCHAR(64) | Y | 标签 |

(2)角色表(daoyunrole)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **字段名称** | **字段类型** | **允许空** | **字段描述** |
| 1 | id | int | N | 编号 |
| 2 | name | VARCHAR(64) | Y | 角色名 |
| 3 | note | VARCHAR(64) | Y | 注释 |
| 4 | uniqeKey | VARCHAR(64) | Y | 唯一项 |
| 5 | status | VARCHAR(64) | Y | 状态 |

(3)课程表（daoyuncourse）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **字段名称** | **字段类型** | **允许空** | **字段描述** |
| 1 | id | VARCHAR(64) | N | 编号 |
| 2 | coursename | VARCHAR(64) | Y | 课程名称 |
| 3 | teachename | VARCHAR(64) | Y | 老师姓名 |
| 4 | time | VARCHAR(64) | Y | 时间 |
| 5 | intro | VARCHAR(64) | Y | 课程介绍 |
| 6 | info | VARCHAR(64) | Y | 课程详情 |

(4)选课记录表（courseRecord）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **字段名称** | **字段类型** | **允许空** | **字段描述** |
| 1 | id | VARCHAR(64) | N | 编号 |
| 2 | courseId | VARCHAR(64) | N | 课程编号 |
| 3 | userId | VARCHAR(64) | Y | 用户编号 |
| 4 | courseExp | VARCHAR(64) | Y | 课程经验 |
| 5 | JoinTime | VARCHAR(64) | Y | 加入课程时间 |
| 6 | score | double | Y | 最终成绩 |

(5)菜单(authroutes)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **字段名称** | **字段类型** | **允许空** | **字段描述** |
| 1 | id | int | N | 编号 |
| 2 | name | VARCHAR(64) | Y | 名称 |
| 3 | path | VARCHAR(64) | Y | 路径 |
| 4 | AUTHORS | VARCHAR(64) | Y | 权限 |
| 5 | pid | int | N | 父编号 |

(6)数据字典(dict)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **字段名称** | **字段类型** | **允许空** | **字段描述** |
| 1 | dictId | int | N | 字典编号 |
| 2 | key | VARCHAR(64) | Y | 关键字 |
| 3 | value | VARCHAR(64) | Y | 值 |
| 4 | dictName | VARCHAR(64) | Y | 字典名 |

三、技术简介

ASP.NET又称为ASP+，不仅仅是ASP的简单升级，而是微软公司推出的新一代脚本语言。ASP.NET基于.NET Framework的Web开发平台，不但吸收了ASP以前版本的最大优点并参照Java、VB语言的开发优势加入了许多新的特色，同时也修正了以前的ASP版本的运行错误。ASP.NET具备开发网站应用程序的一切解决方案，包括验证、缓存、状态管理、调试和部署等全部功能。在代码撰写方面特色是将页面逻辑和业务逻辑分开，它分离程序代码与显示的内容，让丰富多彩的网页更容易撰写。同时使程序代码看起来更洁净、更简单。

采用MVC 编程模式，MVC 是一种使用 MVC（Model View Controller 模型-视图-控制器）设计创建 Web 应用程序的模式：Model（模型）表示应用程序核心（比如数据库记录列表）。View（视图）显示数据（数据库记录）。Controller（控制器）处理输入（写入数据库记录）。MVC 模式同时提供了对 HTML、CSS 和 JavaScript 的完全控制。

前端UI框架使用layui，layui（谐音：类UI) 是一款采用自身模块规范编写的前端 UI 框架，遵循原生 HTML/CSS/JS 的书写与组织形式，门槛极低，拿来即用。其外在极简，却又不失饱满的内在，体积轻盈，组件丰盈，从核心代码到 API 的每一处细节都经过精心雕琢，非常适合界面的快速开发。

数据交互使用React：使创建交互式 UI 变得轻而易举。为应用的每一个状态设计简洁的视图，当数据改变时 React 能有效地更新并正确地渲染组件。

后端使用了MyBatis框架：很大程度上的简化了开发,由其是它的自动化配置,让大大的提高了开发效率。

数据库存储数据使用MySQL：MySQL是最流行的关系型数据库管理系统，在WEB应用方面MySQL是最好的关系数据库管理系统应用软件之一。使用MySQL可以更加便捷管理数据库数据，实现数据库数据的增删查改。

项目部署在华为的云服务器上：可以通过链接直接访问到单页面应用的网站，同时后端也都部署到了上面。

Axure：Axure RP是美国Axure Software Solution公司旗舰产品，是一个专业的快速原型设计工具，让负责定义需求和规格、设计功能和界面的专家能够快速创建应用软件或Web网站的线框图、流程图、原型和规格说明文档。作为专业的原型设计工具，它能快速、高效的创建原型，同时支持多人协作设计和版本控制管理

AngularJS：AngularJS诞生于2009年，由Misko Hevery 等人创建，后为Google所收购。是一款优秀的前端JS框架，已经被用于Google的多款产品当中。AngularJS有着诸多特性，最为核心的是：MVC（Model–view–controller）、模块化、自动化双向数据绑定、语义化标签、依赖注入等等。