

杭州电子科技大学

HANGZHOU DIANZI UNIVERSITY



《软件开发实践 2》

系统开发报告

课程名称:	《软件开发实践 2》
题目:	《课表查询系统》
组号:	第 6 组
姓名:	马家成
任课教师:	穆海伦

完成时间：2022 年 07 月 05 日

一、系统概述

课表查询系统，基于 Vue 框架开发，能够为用户提供快速、准确的课程查询服务。系统根据不同用户的需求赋予用户权限，可实现个人课程查询、教学班课程查询、年级课程查询以及课程情况查询等多项功能。查询内容包括课程时间安排、课程班级成员、班级成员课程空闲时间等方面。

二、需求分析

在日常教学工作中，用户对于课程查询功能需求不尽相同，为保障系统服务功能齐全、服务对象广泛，同时避免系统结构过于复杂。我们最终确定系统功能如下：

1. 简洁、美观的操作界面。
2. 为不同用户设计合适的查询权限。
3. 满足学生、任课教师、辅导员三个主要用户的查询需求。
4. 学生查询：个人信息查询、按学期查询课表、按教学周次查询课表、课程详细信息查询。
5. 教师查询：本学期课程查询、当日课程查询、教学班学生信息查询、教学班学生课表查询、学生课程空余时间查询。
6. 辅导员查询：年级学生信息查询、年级学生课表查询、年级学生选课情况查询。

三、系统设计

前端设计：

课表查询系统前端搭建采用 Vue 框架，包括登陆界面（Login）、学生界面（Student）、教师界面（Teacher）、辅导员界面（Instructor）四个主界面以及若干二级界面，结构关系如图 3.1 所示：

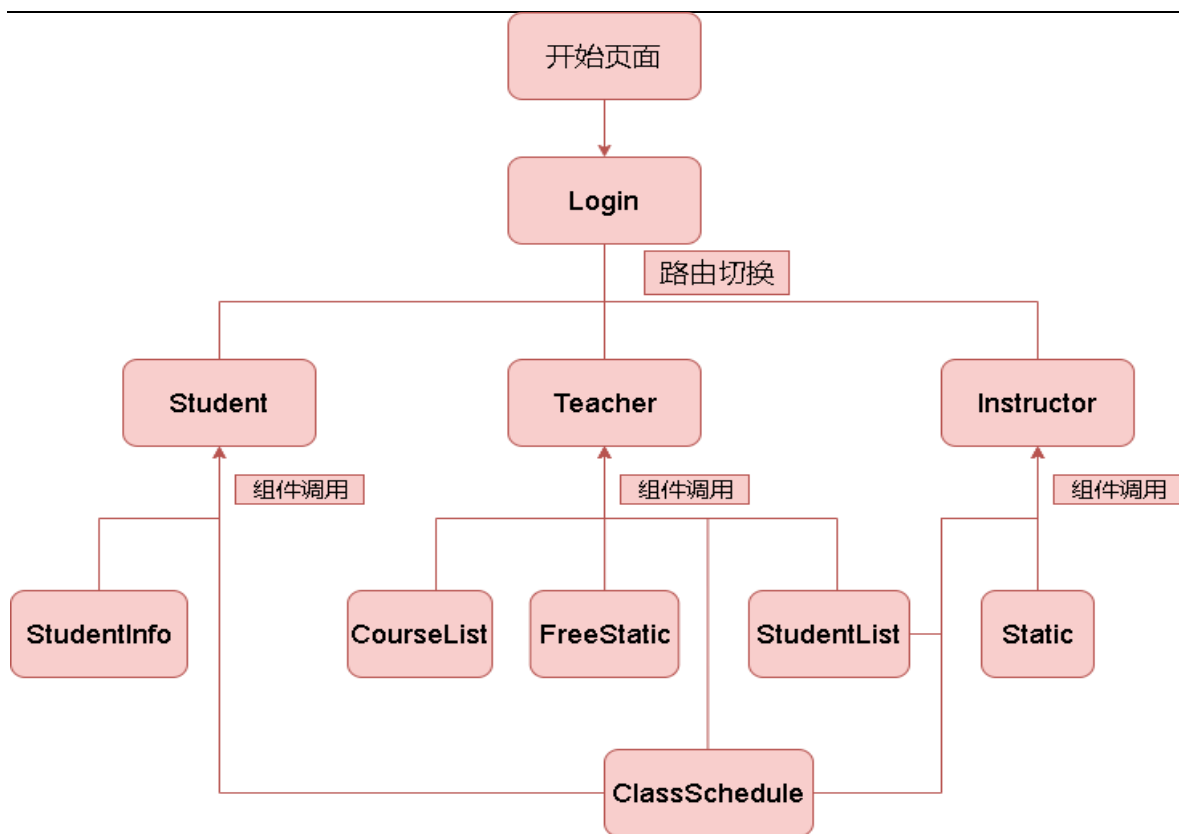


图 3.1

前端界面 UI 采用 Element-UI 组件设计，该 UI 组件已封装好，可直接调用使用，在开发过程中更加简便快捷。

接口设计：

本项目采用 Axios 作为前后端接口，在前端发送 Get/Post 请求，后端响应，并执行数据查询等操作。如下图所示，前端调用 Axios 接口，根据后台服务器地址发送请求，并将返回的数据保存。

```
axios
  .post('http://localhost:3000/api/user/studentcourses', {
    // 传递参数
    sno: this.student.sno
  })
  .then((response) => {
    this.totalCourse = Array.from(response.data)
    console.log('查询学生课程信息成功!')
  })
```

图 3.2

```

// 获取管理的学生信息
axios
.post('http://localhost:3000/api/user/instructorstudents', {
  tno: this.instructor.tno
})
.then((response) => {
  console.log(response)
  this.ManageStudent = Array.from(response.data)
  // 获取管理学生的课程信息
  var AllStudentCourse = []
  for (var i in this.ManageStudent) {
    var student = this.ManageStudent[i]
    axios
      .post('http://localhost:3000/api/user/studentcourses', {
        sno: student.sno
      })
      .then((response) => {
        AllStudentCourse.push(Array.from(response.data))
      })
  }
  this.AllStudentCourse = AllStudentCourse

  // 获取管理学生的上课时间段信息
  var AllStudentBusyTime = []
  for (var k in this.ManageStudent) {
    var stu = this.ManageStudent[k]
    axios
      .post('http://localhost:3000/api/user/studenttermbusytime', {
        sno: stu.sno,
        cterm: '2021-2022-2'
      })
      .then((response) => {
        AllStudentBusyTime.push(Array.from(response.data))
      })
  }
  this.AllStudentBusyTime = AllStudentBusyTime

```

图 3.3

后端设计:

后端采用基于 Node js 的 express 框架, 通过 MySQL 库连接数据库, 响应前端发送的请求, 对数据库进行操作。

后端连接数据库:

```

module.exports = {
  ...
  mysql: {
    host: '127.0.0.1',
    user: 'school',
    password: '20052010',
    database: 'school',
    port: '3306'
  }
}

```

后端搭建服务器，配置路由，监听端口：

```
const userApi = require('./api/userApi')
const bodyParser = require('body-parser')
const express = require('express')
const app = express()

// 采用设置所有均可访问的方法解决跨域问题
app.all('*', function (req, res, next) {
  // 设置允许跨域的域名，*代表允许任意域名跨域
  res.header('Access-Control-Allow-Origin', '*')
  // 允许的header类型
  res.header('Access-Control-Allow-Headers', 'content-type')
  // 跨域允许的请求方式
  res.header('Access-Control-Allow-Methods', 'DELETE,PUT,POST,GET,OPTIONS')
  if (req.method.toLowerCase() === 'options') {
    res.send(200)
  } else {
    next()
  }
})
app.use(bodyParser.json())
app.use(bodyParser.urlencoded({
  extended: false
}))
// 后端api路由

app.use('/api/user', userApi)

// 监听端口
app.listen(3000)
console.log('success listen at port:3000.....')
```

建立语句映射，实现对数据库的各种操作：

```
// sql语句
// sqlMap.js—SQL 语句映射文件，供 API 调用
var sqlMap = {
  select: {
    AllStudents: 'select sno,sname,ssex,sdepartment,smajor,inyear from student',
    AllCourses: 'select cno,cname,cterm,chours,ccredit,cstartweek,cendweek,cclassroom,ctime,ctimeslot,tno fr',
    AllTeachers: 'select tno,tname,tdepartment,tttitle from teacher',
    CourseStudents: 'select student.sno,sname,ssex,sdepartment,smajor,inyear from student,selection where se',
    StudentCourses: 'select course.cno,cname,cterm,chours,ccredit,cstartweek,cendweek,cclassroom,ctime,ctime',
    TeacherCourses: 'select cno,cname,cterm,chours,ccredit,cstartweek,cendweek,cclassroom,ctime,ctimeslot,tn',
    InstructorStudents: 'select student.sno,sname,ssex,sdepartment,smajor,inyear from student where tno=?',
    StudentTermBusyTime: 'select day,ctimeslot from busystudent where sno=? and cterm=?'
  },
  register: {
    Teacher: 'select tno,tname,tdepartment,tttitle,instructor from teacher where teacher.tno=? and teacher.tp',
    Student: 'select sno,sname,ssex,sdepartment,smajor,inyear from student where student.sno=? and student.s'
  }
}
module.exports = sqlMap
```

数据库设计：

数据库使用 MySQL 数据库。创建表以及字段含义如下：

学生表字段含义:

```
studentKeys: {
  'sno': '学号',
  'spassword': '密码',
  'sname': '姓名',
  'ssex': '性别',
  'inyear': '入学年份',
  'sdepartment': '学院',
  'smajor': '专业',
  'tno': '辅导员工号'
},
```

sno	spassword	sname	ssex	sdepartment	smajor	inyear
20180001	23ewde23	王磊	男	会计学院	会计学	2018
20180002	j9m08u0	刘洋	男	法学院	法学	2018
20180003	1n23nl2	王艳	女	计算机学院	计算机科学与技术	2018
20180004	asd9w0a0	王勇	男	电子信息学院	电子信息工程	2018
20180005	wfwerdw	李军	男	电子信息学院	电子信息工程	2018
20180006	wqdwqew70ds	张勇	男	会计学院	会计学	2018
20190001	d9qw8d09w	李杰	男	计算机学院	软件工程	2019

教师表字段含义:

```
teacherKeys: {
  'tno': '教师号/辅导员工号',
  'tname': '姓名',
  'tdepartment': '院系',
  'tttitle': '职称'
},
```

tno	tname	tdepartment	tttitle
200006001	元尚武	马克思主义学院	教授
200106002	张辉	马克思主义学院	讲师
200202002	江州	外国语学院	教授
200302002	张丹	法学院	教授

课程表字段含义：

```
courseKeys: {  
  'cno': '课程号',  
  'cname': '课程名',  
  'cterm': '学期',  
  'chours': '学时',  
  'ccredit': '学分',  
  'cstartweek': '开始周',  
  'cendweek': '结束周',  
  'cclassroom': '教室',  
  'ctime': '上课时间',  
}
```

四、系统开发环境

代码编辑器：Visual Studio Code（版本号 1.68.1）

开发框架：前端：Vue（版本 2.x）

后端：Node.js（版本 16.15.1）

数据库：MySQL

浏览器：Microsoft Edge / Google Chrome

安装教程：

Visual Studio Code：下载地址 <https://code.visualstudio.com/>

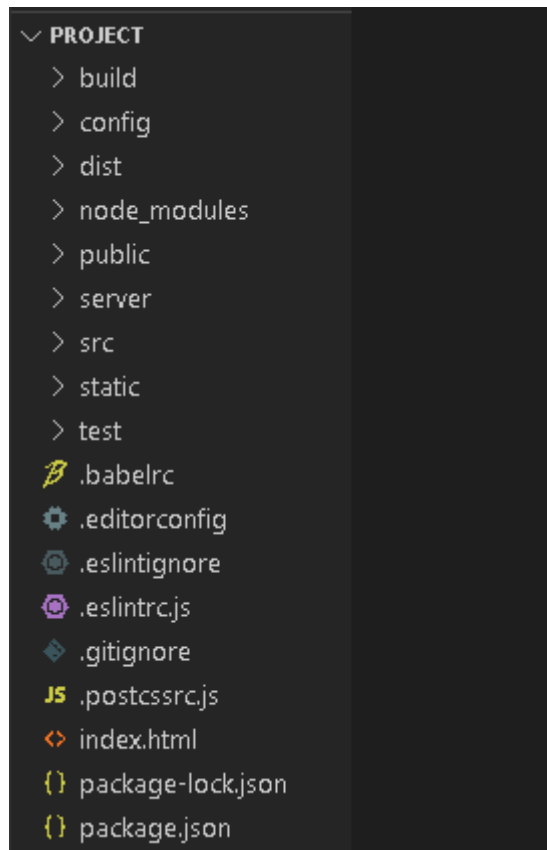
Node.js：下载地址 [Node.js \(nodejs.org\)](https://nodejs.org/)

MySQL：下载地址 <https://www.mysql.com/>

Vue 框架：安装完成 Node.js 后，Win + R 键打开运行窗口，输入 cmd 打开管理员终端，输入以下指令 npm install -g cnpm --registry=https://registry.npm.taobao.org 待进度条满即可完成安装。

五、系统操作手册

1. 参照上文步骤，安装 Node.js、Vue 框架、VSCode 等必要组件。
2. 将项目压缩包解压到磁盘。
3. 打开 VSCode 软件，打开解压完成的文件夹，如下图所示：



4. 打开终端，进入 Project 下的 server 目录，终端输入 `node index.js` 执行 `index.js` 文件，启动服务器，连接云数据库。如下图所示：

```
PS D:\报告\软件实践2\test\Project\server> node index.js
success listen at port:3000.....
```

5. 新开终端，输入 `npm run dev` 指令，在 Project 根目录下启动 Vue 项目。如下图所示：

```
DONE Compiled successfully in 11633ms
I Your application is running here: http://localhost:8080
```


6. 按住键盘 Ctrl 键，点击项目启动后的地址，即可进入系统。如图所示：

