

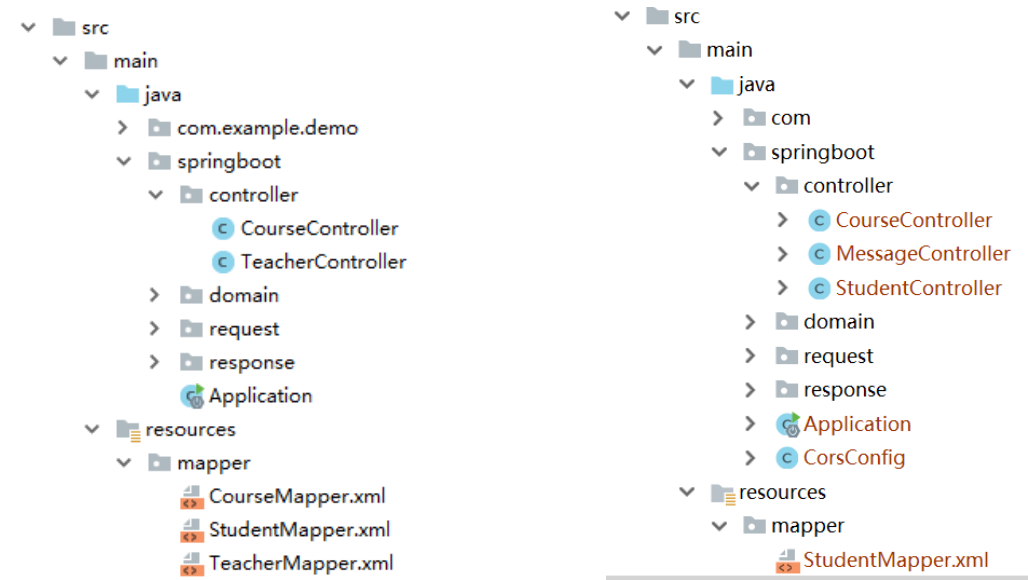
# 基于微信小程序的教学内容管理平台设计文档

## 一、教师服务器端

- 1、设计思想：前后端分离
- 2、项目设计：前端使用 ajax 向后端发送请求，后端使用 Springboot 框架提供 restful 接口，处理前端发回来的请求。
- 3、项目实施

### 教师后端

- 1、后端使用框架：Springboot：其他技术：Mybatis
- 2、后端项目结构：



### 3、后端的包及主要功能

后端主要的包及功能		
controller 包		
CourseController	@RequestMapping (value = "/findAllCourse", method = RequestMethod.GET) public @ResponseBody Object listAll() 列举出所有的课程，并返回响应体 (以下省略 requestMappping 和请求方法)	主要的地址映射， 处理请求的方法
	findCourseById 根据 CourseID 获取课程信息	
	findTeacherNameByCourseID 根据 CourseID 查找老师姓名	

	findChapterByCourseID 根据课程 id 获取章节名称	
	findCommentByCourseID 根据课程 id 获取评论	
	findStudentsByCourseID 根据课程 id 获取学生名单	
	findHomeworkByCourseID 根据课程 id 获取作业完成情况	
	findHomeworkTokenByID 根据作业 id 获取作业完成情况	
	findStudentNameByCourseID 根据学生 id 获取学生姓名	
	findCoursesByTeacherID 根据教师 id 获取开课列表	
	updateIntroduction 根据课程 id 更新课程介绍	
	updateChapter 根据课程 id 更新章节	
	addCourse 开设新课程	
	addHomework 布置新作业	
	TeacherController	
login 教师登录验证		
teacherOnline 根据 ID 获取教师个人信息		
updateInfo 更新教师个人信息		
StudentController	listAllStudent 获取所有学生信息	
	findStudentByNick 通过学生昵称查询对应详细信息	
	modify 修改学生个人信息	
MessageController	recvMsg 对话式教学记录	
	sendMsg 学生发送消息	
domain 包		
Teacher、	类中属性的 getter,setter 方法	对应数据库中的

Student、 Course、 等实体类		主要实体类
<b>request 包</b>		
TeacherLogin、 StudentRegister、 CourseRequest、 等实体类	类中属性的 getter,setter 方法	定义请求体，在 controller 中使用
<b>response 包</b>		
ErrorResponse、 MessageResponse、 SuccessResponse 等实体类	类中属性的 getter,setter 方法	定义响应体，在 controller 中使用
<b>mapper 包</b>		
CourseMapper.xml	对数据库中课程的增删改查的 sql 语句	主要数据库操作的 sql 语句，对应 controller 中的调用
TeacherMapper.xml	对数据库中教师的增删改查的 sql 语句	
StudentMapper.xml	对数据库中学生的增删改查的 sql 语句	

## 教师前端

- 1、前端使用框架：vue 其他技术：ajax, bootStrap
- 2、前端项目结构

```

> css
v data
  blog.js
  blogCourse.js
  index.js
  myCourse.js
  register.js
  userInfo.js
> fonts
> images
> js
> sass
  blog.html
  blogCourse.html
  chat.html
  createNewCourse.html
  index.html
  myCourse.html
  register.html
  unregister.html
  userInfo.html

```

### 3、前端主要的包及功能

前端主要的包及功能		
<b>data 包</b>		
主要的 js 文件，处理对应同名的 html 文件页面的操作和请求，使用 cookie 记录当前登录的用户。		
<b>位于根目录下 html 文件</b>		
Index.html	用户登录入口	用户登录，验证用户是否注册过，密码是否正确
Register.html	用户注册	用户使用用户名，邮箱，密码进行注册。邮箱作为唯一标识。
blog.html	主页：展示所有课程	展示数据库中所有的课程，可以浏览
blogCourse.html	课程详细页面	课程详情页面，开设课程的老师有权限修改信息
Mycourse.html	我开设的课程页面	获取当前登录用户开设的所有课程
createNewCourse.html	创建新课程页面	当前登录用户创建新的课程
userInfo.html	用户个人信息页面	获取当前登录用户的个人信息，提供修改功能

### 微信小程序端简介

我们项目主要使用 mpvue 来写小程序的前端框架，后端使用 springboot 框架，前后端之间的联系大致是 springboot 写好需要从数据库读取的数据，就是一个 restful 接口，然后小程序通过调用一个 flyio 跟 restful 接口对接，实现前端读取后端的数据。

#### 1、前端（学生端）的实现部分

**CSS/目录**---主要是用来存放公共的样式

- 在这里我们存放 app.css, butto.css 的样式，为了统一美观，同时也可以减少代码量。

**Pages/目录**---就是小程序的页面，每一个页面对应一个文件夹。

- chat 目录: 对话式教学界面
- courseMsg 目录: 显示学生收藏的所有课程的页面。
- index 目录: 小程序首页，展示搜索框，轮播以及课程信息。
- learn 目录: 显示学生所学习的课程，包括课程名称，教师，上课时间，老师发布的作业以及作业完成状态。
- login 目录: “我的”界面，内容可以看到登录信息，比如头像、昵称，还有学员手册、我的收藏、个人信息。
- manual 目录: 学员手册信息界面，里面的内容就说我们小程序项目的大致内容。
- takeCourses 目录: 参与课程的界面，就是学生查看自己添加的已参与课程的界面
- test 目录: 课程详情页面，首页点击某一课程跳到该课程详情页面。

- userInfo 目录: 个人信息修改的界面, 就是学生登陆后完善个人信息。

### **Utils/目录---工具包**

- 存放 flyio 所使用的 request.js。

### **微信小程序未解决的技术问题**

- mpvue 框架使用 flyio 获得后端数据之后不能正确排版到页面上, 不能获得返回数组的  
中指定的一条数据, 因为 mpvue 与微信小程序语法有所差异。