常州信息职业技术学院

学生毕业设计论文

二 级 学 院： 软件与大数据

专 业： 软件技术

班 级： 软件精英1712

学 生 姓 名： 胡旺生

学 生 学 号： 17091230334

论 文 题 目： 基于node.js与Koa打造在线教学

指 导 教 师： 曾世邦

设 计 地 点： 常州信息职业技术学院

起 迄 日 期： 2019-5-6

**摘 要**

近年来随着互联网的兴起，短短几年的时间势如破竹般的影响了人们的方方面面，例如出行，购物，医疗，教学等等，无一例外。当然这也同样是一种机会，高新技术的超速发展和全球化的市场经济导致企业之间的竞争空前激烈，这种竞争归根结底是人才的竞争，谁能有效地管理和运用互联网，无疑谁就能掌握知识经济时代的生产力。

本文基于node.js和Koa框架打造在线教学实例,实现了用户在线学习和交流,利用琐碎的时间对自己进行学习,提高自我技能。主要实现了用户的增删改查和观看学习视频。

**关键字:web,在线教学，koa，node.js**

**Abstract**

In recent years, with the rise of the Internet, the short period of time has affected people's various aspects, such as travel, shopping, medical care, teaching, etc., without exception. Of course, this is also an opportunity. The rapid development of high-tech and the globalized market economy have led to unprecedented competition among enterprises. This competition is ultimately the competition of talents. Who can effectively manage and use the Internet? Can master the productivity of the knowledge economy era.

This article builds online teaching examples based on node.js and Koa frameworks, enabling users to learn and communicate online, using trivial time to learn and improve their skills. It mainly realizes the addition, deletion and change of the user and the viewing of the learning video.

Keywords: web, online teaching, koa, node.js

目录

**[第一章 前言](#_Toc16348_WPSOffice_Level1)** **[4](#_Toc16348_WPSOffice_Level1)**

[1.1设计背景](#_Toc21563_WPSOffice_Level2) [4](#_Toc21563_WPSOffice_Level2)

[1.2设计目的](#_Toc10907_WPSOffice_Level2) [5](#_Toc10907_WPSOffice_Level2)

[1.3设计的意义](#_Toc24757_WPSOffice_Level2) [5](#_Toc24757_WPSOffice_Level2)

**[第二章 需求分析](#_Toc21563_WPSOffice_Level1)** **[6](#_Toc21563_WPSOffice_Level1)**

[2.1 可行性分析](#_Toc8036_WPSOffice_Level2) [6](#_Toc8036_WPSOffice_Level2)

[2.2 用户需求](#_Toc23118_WPSOffice_Level2) [6](#_Toc23118_WPSOffice_Level2)

[2.3 安全可行性分析](#_Toc14043_WPSOffice_Level2) [7](#_Toc14043_WPSOffice_Level2)

**[第三章 实现技术](#_Toc10907_WPSOffice_Level1)** **[8](#_Toc10907_WPSOffice_Level1)**

[3.1 node.js简介](#_Toc19470_WPSOffice_Level2) [8](#_Toc19470_WPSOffice_Level2)

[3.2 monogodb和Mongoose简介](#_Toc17185_WPSOffice_Level2) [9](#_Toc17185_WPSOffice_Level2)

**[第四章 详细设计](#_Toc24757_WPSOffice_Level1)** **[10](#_Toc24757_WPSOffice_Level1)**

[4.1 前端界面设计](#_Toc31318_WPSOffice_Level2) [10](#_Toc31318_WPSOffice_Level2)

[4.2 后端功能实现](#_Toc16218_WPSOffice_Level2) [14](#_Toc16218_WPSOffice_Level2)

[4.3 博达在线教学网站功能实现简介](#_Toc12439_WPSOffice_Level2) [15](#_Toc12439_WPSOffice_Level2)

**[第五章 在线教学系统测试](#_Toc8036_WPSOffice_Level1)** **[30](#_Toc8036_WPSOffice_Level1)**

[5.1系统测试的目标](#_Toc23412_WPSOffice_Level2) [30](#_Toc23412_WPSOffice_Level2)

[5.2具体测试](#_Toc24784_WPSOffice_Level2) [31](#_Toc24784_WPSOffice_Level2)

**[第六章 总结与展望](#_Toc23118_WPSOffice_Level1)** **[33](#_Toc23118_WPSOffice_Level1)**

**[致谢词](#_Toc14043_WPSOffice_Level1)** **[34](#_Toc14043_WPSOffice_Level1)**

**[参考文献](#_Toc7702_WPSOffice_Level1)** **[35](#_Toc7702_WPSOffice_Level1)**

# 第一章 前言

## 1.1设计背景

教学是小到一个大学最基本的业务，大到乃至一个国家的根本，教学是提高人口的总和素质和能力的最基本体现，是一个大学的核心工作，担负着培养一代代优秀学生的使命。

传统的教学方式存在很多繁琐的地方，也存在居多的局限性，由于学生的重多，很多时候专业选课的冲突，上课地点的变动，教学老师的人数，上课时间的冲突，查看学生个人有关信息等等，都存在着一定的不灵活性，或者说是局限性，随着互联网的到来，冲击了传统的教学，时间，空间，人力不在成为制约教学和资源分配的主要问题。

基于以上的原因，本文在对基于node.js和Koa框架打造在线教学的研究和实现中，选择了前后端分离的思想来构建，旨在对基于node.js和Koa框架打造在线教学进行研究的同时，为基于node.js和Koa框架打造在线教学的实现进行有益的探索。

## **1.2设计目的**

由于互联网的大规模覆盖，资源的分享已经成为了某种意义上的生活必需品，然而这也就导致了在资源整合和使用的时候由于大量信息的原因，资源变得臃肿和复杂，不利于查找和浏览。在这里本文将选取教学系统来进行资源的整合，实现资源的查找和修改。这将大大的有利于我们的查询信息的速度，我们基于这一情况来进行在线教学的研究将具有重要的意义。

## **1.3设计的意义**

由于信息化的社会快速发展，人们的方方面面也都有所改变，不管是生活，出行，学习更新的速度尤为之快，人们需要不断的去学习进步，从而不让这个社会所淘汰，但是工作中的人又没有那么多大把的时间用来学习，所以基于以上种种原因，萌生出开发一个在线网站，给工作中的人群利用上下班或者休息等一些琐碎的时间给自己充电。这款在线教学网站将完美的实现人们生活，娱乐，放松，学习的需要。

# 第二章 需求分析

## **2.1 可行性分析**

### 2.1.1经济可行性分析

经济可行性分析主要是对网站的开发项目所使用的成本和效益进行评估，包括时间投入成本，人力投入成本，固态资源等等。

1. 网站的费用。建设站费用一般包括：设备费用，开发费，运行费，维护费，时间成本的投入等。
2. 网站的收益。网站的收益有两个方面，一个是直接受益，另一个是间接收益。直接受益网站大部分是指有偿服务，通过运行产生收益。间接收益一般包括网站的使用和维护运行，方便内部人员自己的使用和增加网站的人流量，利用出租摊位摆放广告来产生收益。

这里将从两个方面来进行经济可行性分析，一个是成本，另一个是效益。

本网站基于教育背景开发的在线教学投入的成本主要有：时间成本，人力成本，硬件成本和购买域名和服务器等费用。网站的收益主要有：用户访问产生流量和出租广告教学有关的广告摊位来获取经济的收益。

**2.1.2技术可行性分析**

技术方面可行性分析，就是根据现有技术条件，并分析是否可以达到网站建设的基本要求，从硬件性能方面，软件的性能，环境和能源条件。技术可行性主要做以下的工作：

1. 细分网站目标，分析技术的可行性。
2. 从多个方面进行进行网站的分析。
3. 从多台服务器上进行分析网站的交互性。
4. 分析网站的压力测试。可以采用黑盒测试和白盒测试以及同时很多用户进行进入网站，看网站的抗压能力。

本网站是基于node.js，node.js是服务端的javaScript环境，node.js为基础的一个后台框架，koa具有丰富的中间件以及强大的类库。可以用来轻松完成复杂的功能。如果想了解详细信息请看第三章，这里就不多介绍了。

## **2.2 用户需求**

用户需求针对的是一家公司或者一家企业而言，描述的是公司如何去解决用户提出的问题的，如何去满足用户的欲望，并且将利益最大化，不断地去追求用户的极度体验，把用户拖住，稳住人流量，在进行规模化发展。

1. 如何做用户需求？
2. 对自己的产品做一个定位；概括自己的产品特征，属性。
3. 寻找初步需求；利用调查法，头脑风暴法等方法。
4. 对调查数据进行分析，缩小寻找的圈子，找到最合适的群体并对其进行精准调查，询问需求并加以记录。

基于教育教学网站背景，主要是要页面简洁化但是功能齐全，用户的课程评价，个人中心保存观看历史记录和用户水平测试，满足用户最基本需求。

## **2.3 安全可行性分析**

用户的信息安全，用户注册的账号安全，用户的浏览信息安全，系统（或者网站）的安全使用，有没有漏洞，有没有防御外来入侵的功能。

基于对网站的安全性进行分析，从用户和管理者信息安全，注册，浏览信息以及系统的使用安全性和是否存在漏洞，不安全的因素。有没有可能防御外来的入侵等等。

第一：在线教学的网站对后台管理者安全性做了充分的考虑和拦截，如果某个人知道后台信息的网址，是否直接进去就可以，无需管理员登录，绕过密码的登录，基于这一情况对后台代码做了一系列的拦截，如果你没有登录就算即使知道后台地址，也会被自动拦截并跳转到后台管理员登录页面。在这里使用了koa框架中koa-session，用来获取session和cooker判断管理员的权。

首先，第一步：安装模块npm install koa-session；

第二步：const session = require('koa-session');

第三步：使用

第二：考虑到管理员的密码安全在这里使用了MD5加密算法，对管理者的密码进行加密，当然所有的事情都有好的一面，自然也有不好的一面，因为这个密码的设置是不可逆的。

第三：在管理员登录界面添加了验证码功能，一方面是为了当管理员登录时候给登陆者感觉加载速度不慢，这里使用了koa中的svg-captcha用于验证码模块的使用。

第一步：安装模块

npm install --save svg-captcha

第二步：加载验证模块

const captcha = svgCaptcha.create({

size:4,

fontSize: 50,

width: 120,

height:34,

background:"#cc9966"

});

第三步：生成英文字母或者数字相加

const captcha = svgCaptcha.createMathExpr()

const captcha = svgCaptcha.createMathExpr()

# **实现技术**

## **3.1 node.js简介**

node.js是基于Chrome javaScript运行时候建立的平台,node.js的框架是基于Chrome的一个V8引擎进行封装,是目前为止速度最快的javaScript引擎,node.js是使用事件驱动,非阻塞I/O模型,轻量级,高效,可以很快地处理数据并且运行在不同的设备上,是java做web开发的速度快上好多倍,实现了后端语言可以实现的所有功能,可以媲美于PHP,Java,Python,等后台语言。优点主要有:node.js的学习成本低,上手容易,只要会javaScript语言就可以立马上手,它打破了javaScript过去只能在浏览器中运行的局面,可以处理高并发事件,主要是因为node.js不需要为每一个用户建立线程而仅仅只是触发一个事件,假设一个8GB大小的内存服务器,基本上大约上可以处理超过40000用户的同时进行访问,可以应用于多个领域,如下图所示。

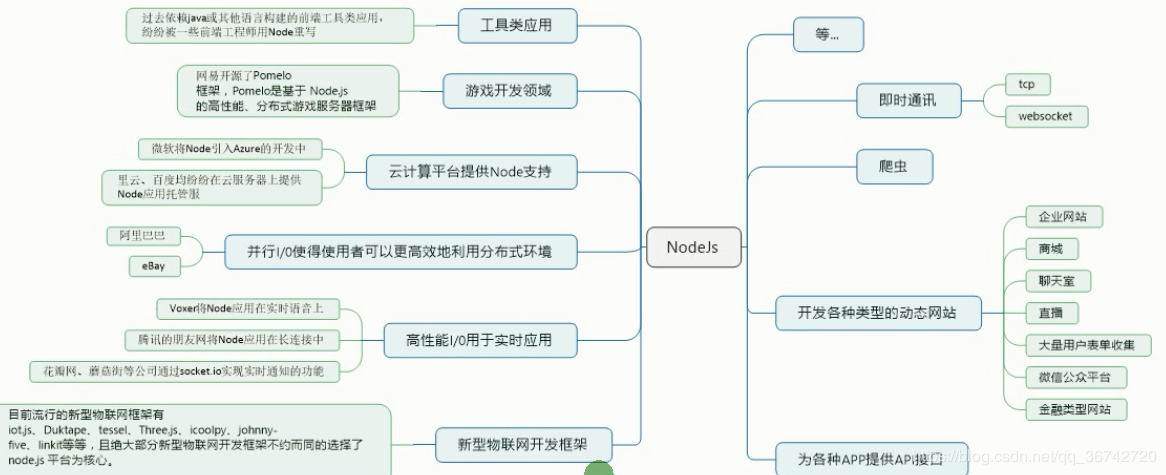
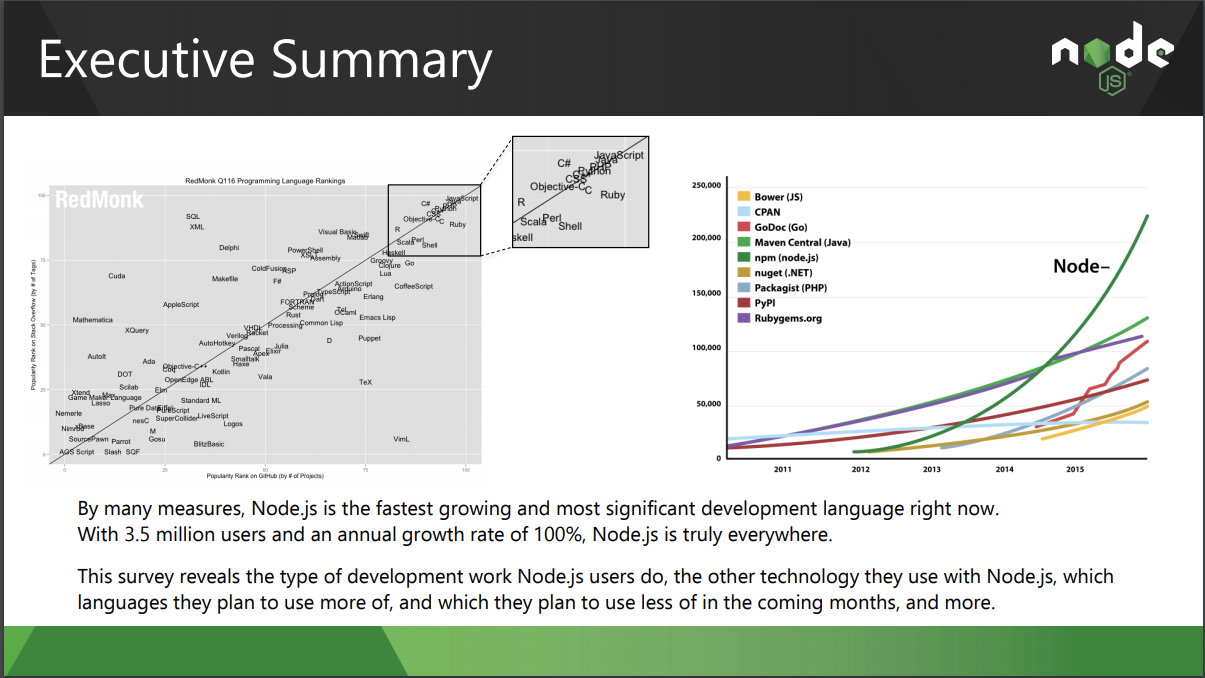
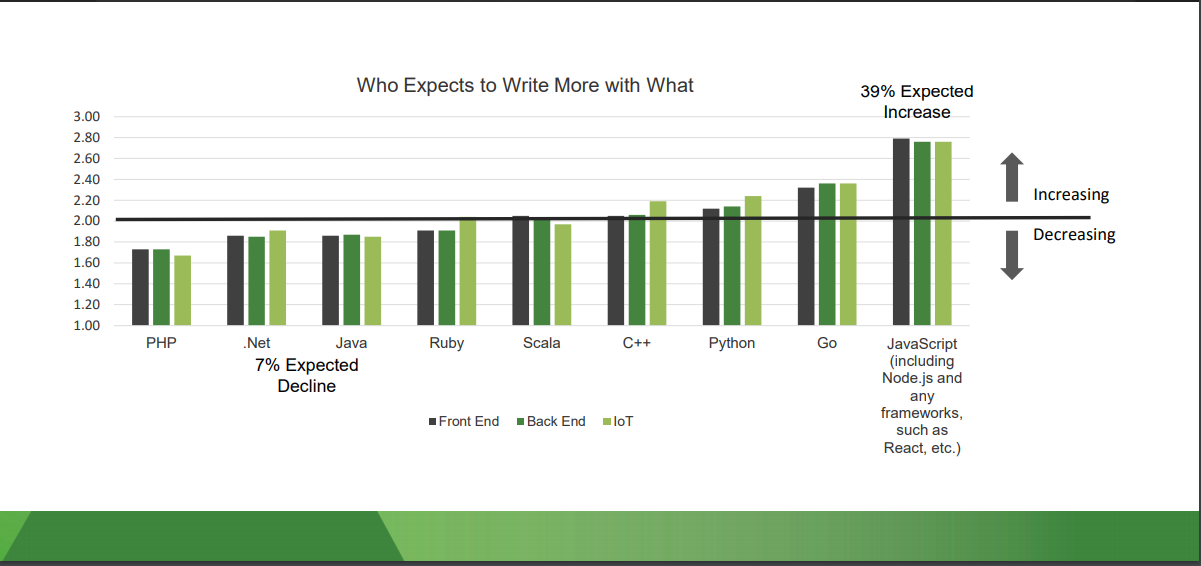
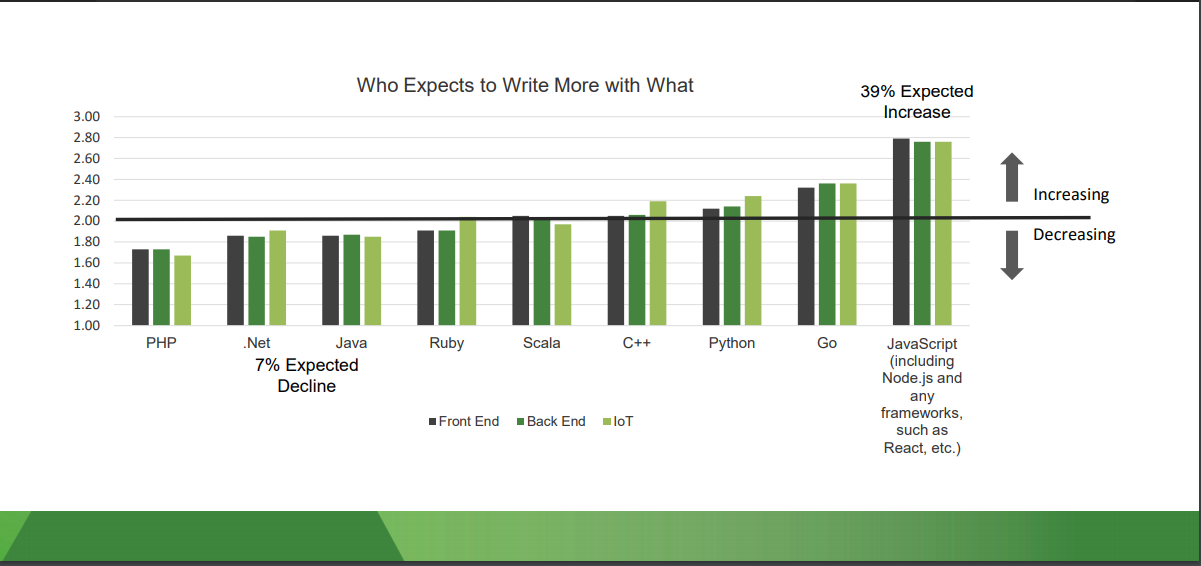


图3-1 node.js功能

随着时间的不断推移，node.js的功能性强大和完善的生态以及学习上手的成本低，很快的发展了起来，很多的大公司例如腾讯，Google都有使用，包块一些小的创业公司，由于创立出去资金有限，也会选择node.js来进行企业网站的开发，如下图分析所示。

图3-2 node.js发展趋势

 图3-3 语言对比

## **3.2 monogodb和Mongoose简介**

Mongodb数据库大部分的开发者都有听过，但是并没有太多人去使用过这样的一个数据库。Mongodb是介于关系数据库和非关系数据库之间的存储工具，是基于分布式文件存储的开源的数据库系统，最大的特点就是查询速度特别快，尤其是海量的数据进行查找并支持建立数据索引，优点有：1.快速！将热性能数据存储在物理存储中，从而使得热数据的读写数据变得十分快速。2.高扩展性！物理机器的添加，mongodb的扩展会达到很高的高度。3.Json的格式进行存储！Mongodb的Json和Bson存储格式和文档格式的储存和查询。与传统型的关系数据库相比：mongodb数据库是非关型数据库，传统数据库(Mysql)是关系型数据库。mongodb存储的方式是虚拟内存+持久化，而传统数据数据库不同引擎有不同的存储方式。Mongodb查询语句独特的查询方式，传统数据库存储都是差不多的sql语句。两者的架构不同：前者使用副本集以及分片来实现高可用性，后者使用M-S,MHA,MMM等架构方式。Mongodb数据库在海量的数据情况下速度可以很快，而传统型数据库在海量数据查询的时候会明显力不从心。

MongoDB的缺点：

①mongodb不支持事务操作。

②mongodb占用空间过大。

③MongoDB没有MySQL工具使用成熟，这是个值得注意的地方。

Mongodb的安装与使用也很方便，下面介绍mongodb的安装

1. 进入官网下载：<https://www.mongodb.com/>(根据电脑的系统配置进行载)
2. 安装完成配置环境变量
3. 在一个盘的根目录下建立一个data文件夹
4. 启动mongodb服务命令 ：服务端：mongod 开启数据库服务 mongod --dbpath C:\mongodb
5. 打开窗口命令行输入mongo连接数据库

# 第四章 详细设计

本章将详细介绍网站的功能以及实现的技术和思路的讲解。本网站是基于node.js的koa框架实现的在线教学网站。分为前端和后端两个部分开发，就像传统的ASP，JSP和PHP，所有用来展示数据都是后端通过异步的接口调用(AJAX/JSONP)的方式提供的数据，而前端只是用来数据的展示。前端主要负责View和Controller层，后端主要实现Model层，业务数据和逻辑处理。这样采用前后端开发开发可以更好的分配人力资源，做到“术业有专攻”，另一方面可以提高开发效率，是前后端职责清晰，不存在推卸责任等，下面将详细介绍。

## **4.1 前端界面设计**

前端主要使用HTML和CSS，JQuery，Bootstrap，JavaScript等技术。

前端页面主要包括：首页，用户登录，用户注册，课程中心，个人中心，关于我们和联系我们主要基本页面，基本结构如下图所示。

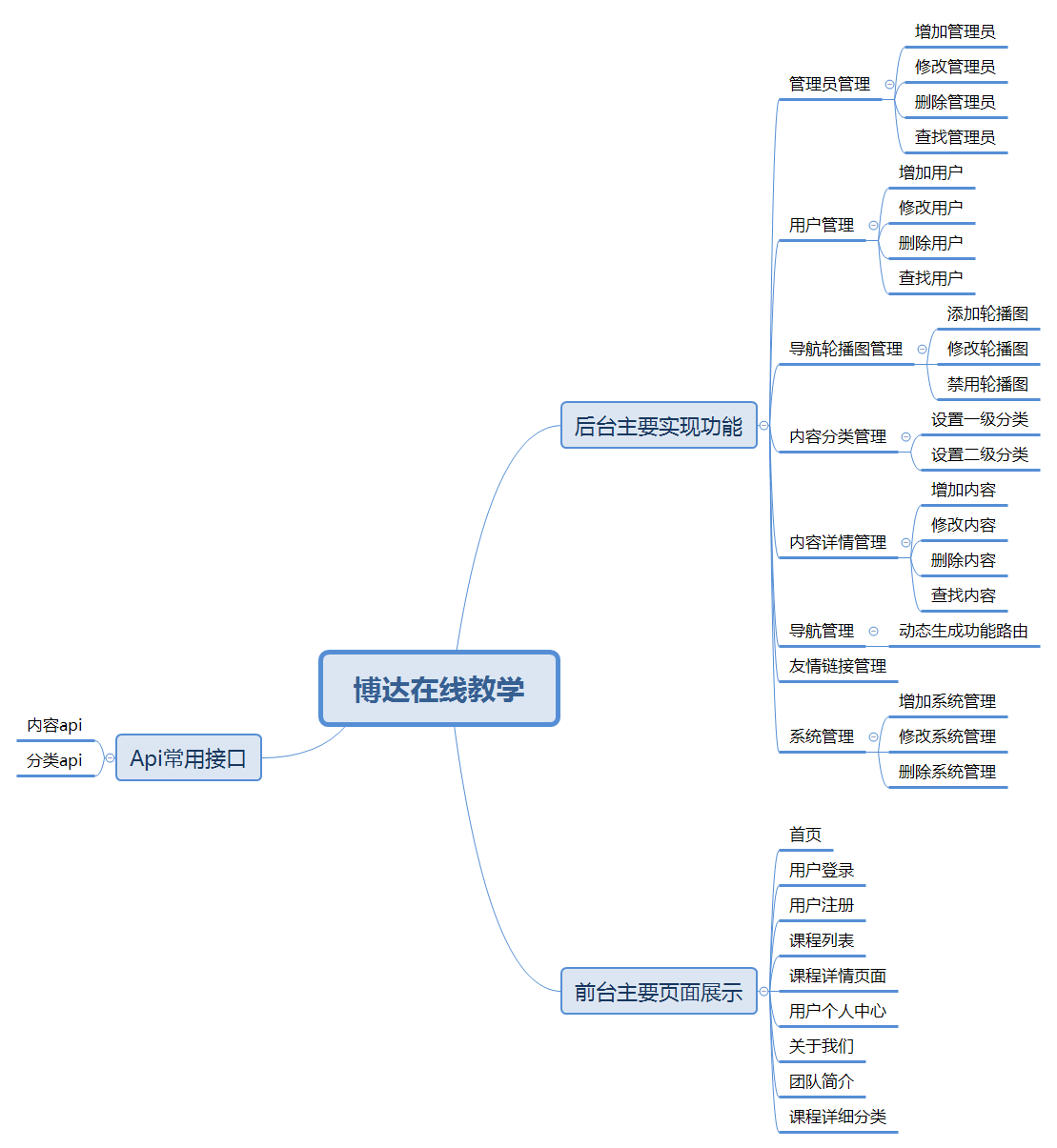


图4.1逻辑业务功能设计

前端主要实现的功能：

1 用户的登录，注册，修改密码和个人相关信息。

2 课程的详细信息以及各类不同教学视频的分类等。

下面是前端部分截图展示：



图4.1-1前端首页

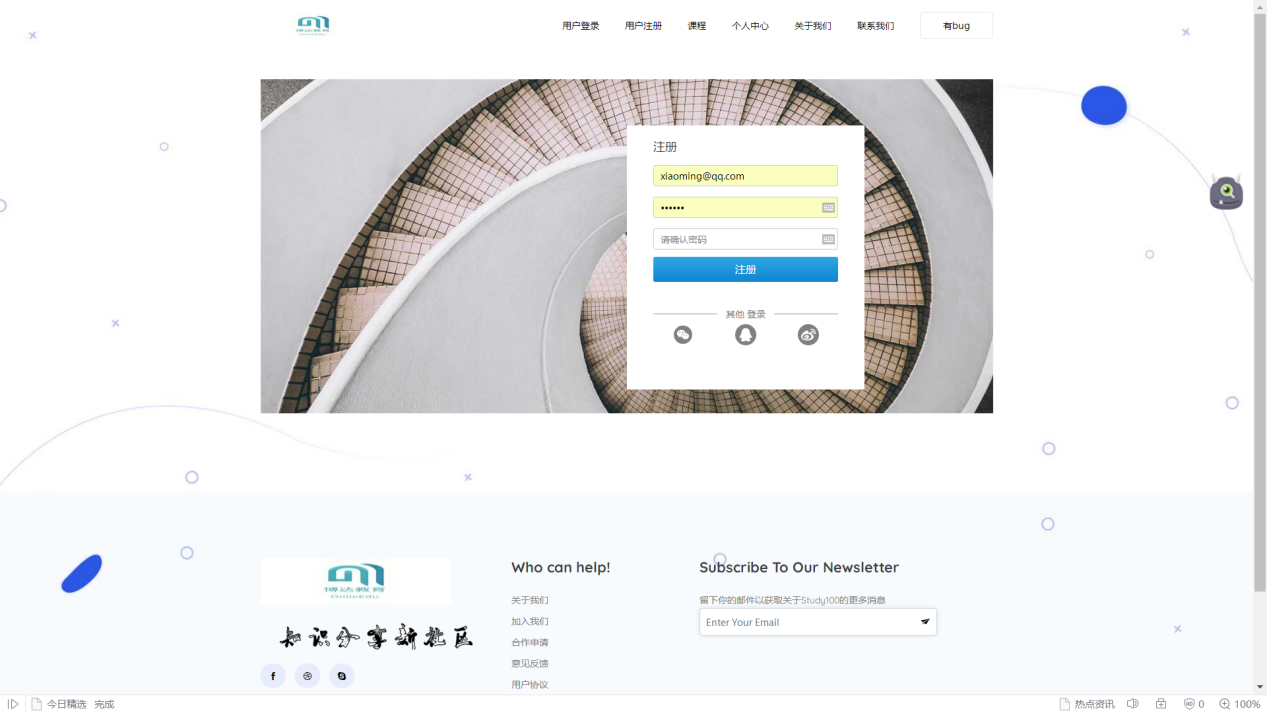


图4.1-2前端用户登录

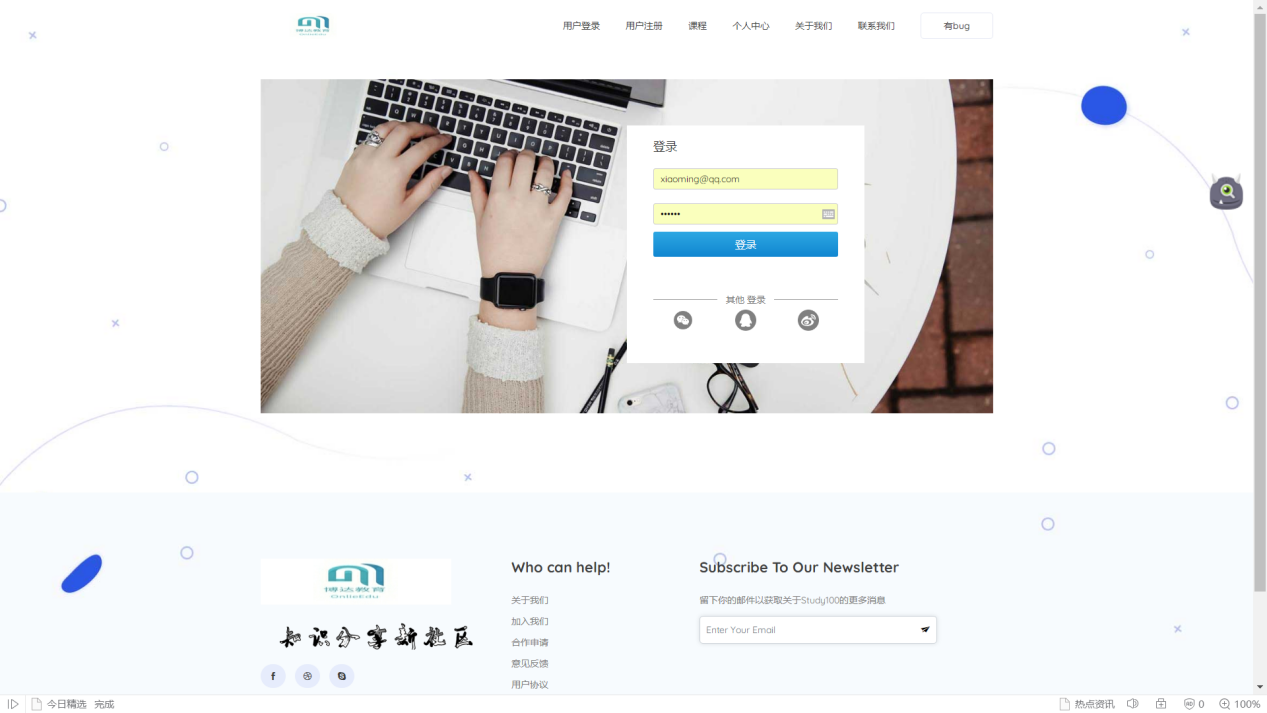


图4.1-3前端用户注册

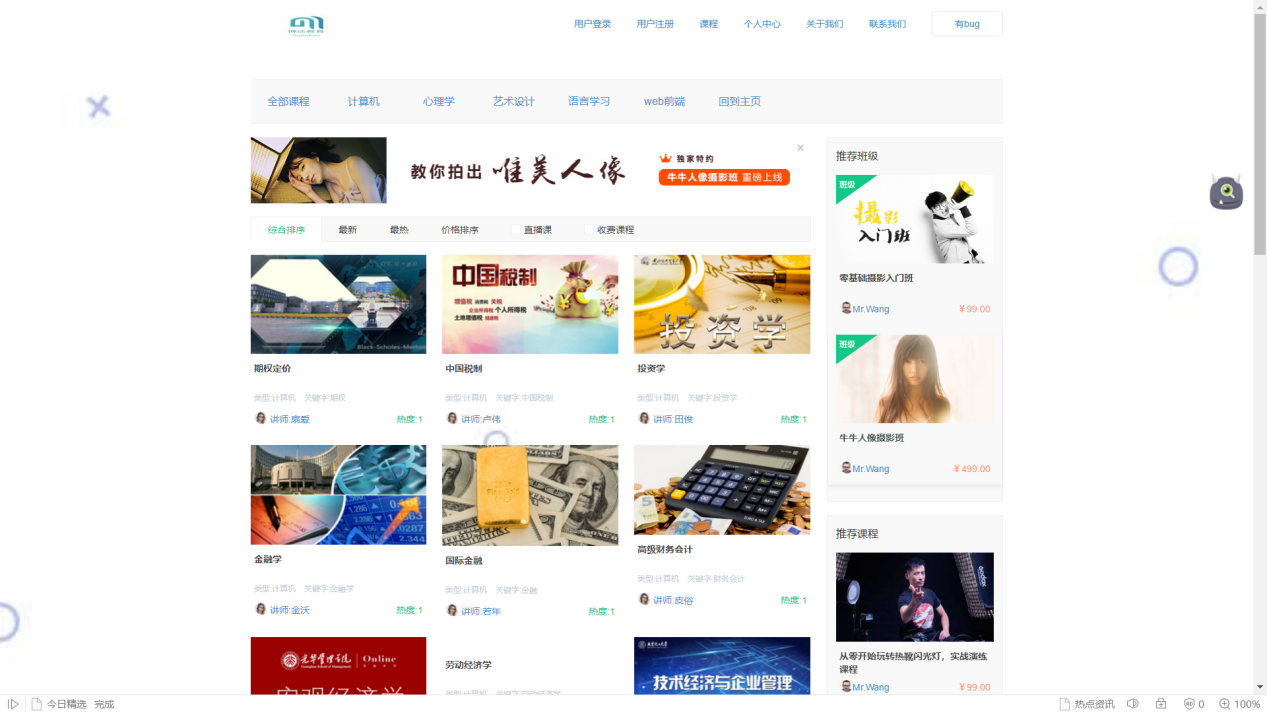


图4.1-4课程中心

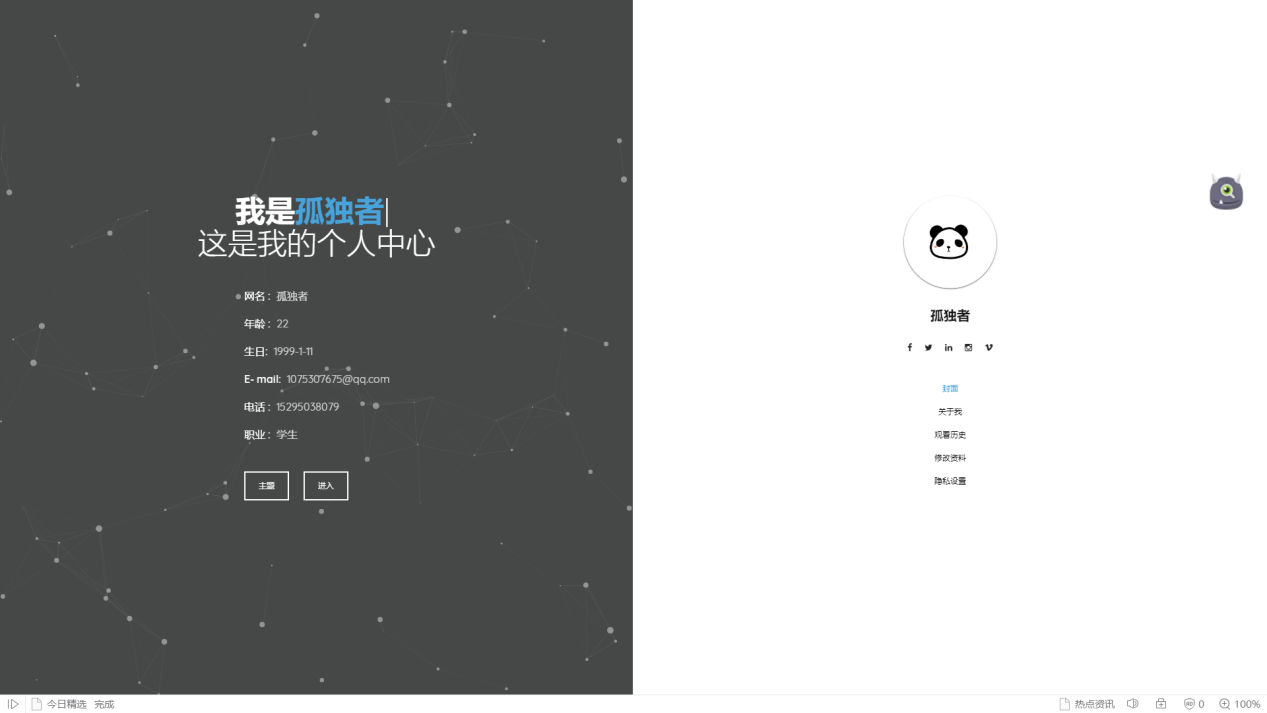


图4.1-5用户个人中心

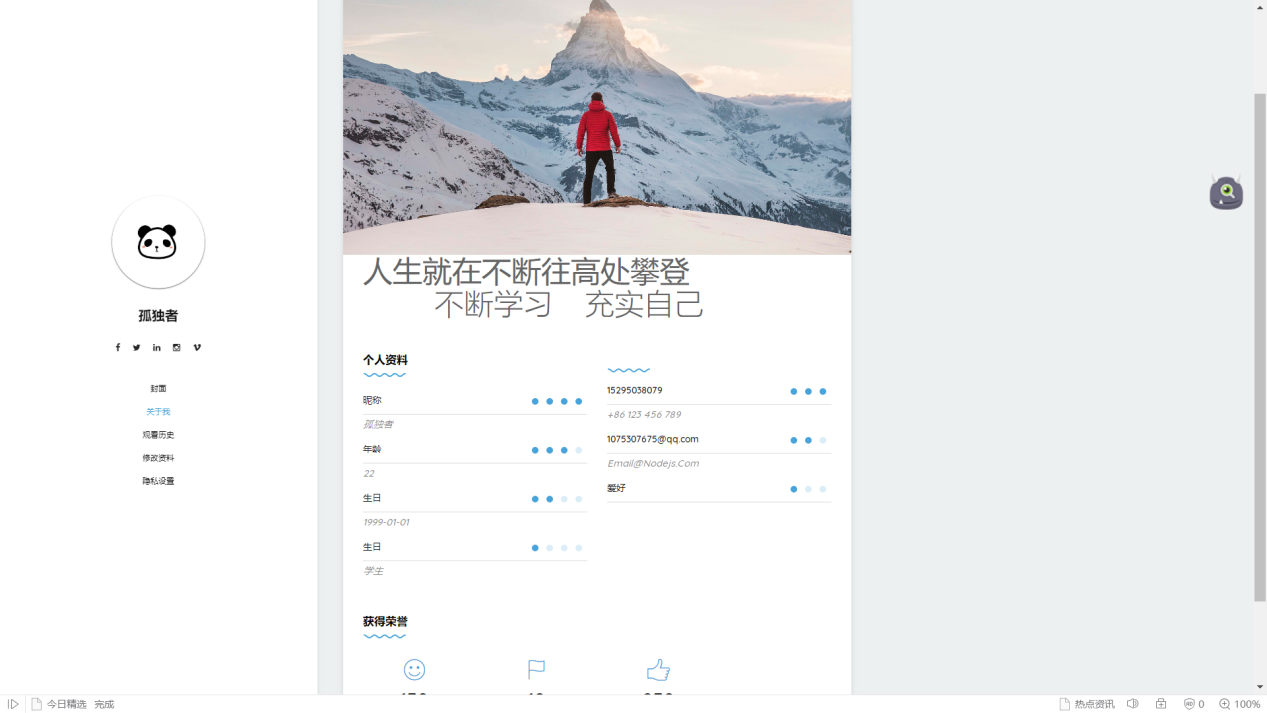


图4.1-6用户基本信息中心

## **4.2 后端功能实现**

后台开发是基于node.js运行的koa的一个成熟框架，是基于javaScript开发的。优点是拥有很多完善的中间组件，使得node.js使用少的代码去实现业务复杂的功能。

由于本网站是一个在线教学网站，所以开发之前进行了功能实现的详细计划和设计，主要业务功能有管理员管理，用户管理，导航轮播图管理，内容分类管理，内容详情管理，导航管理，友情链接管理以及系统管理，如下图所示。

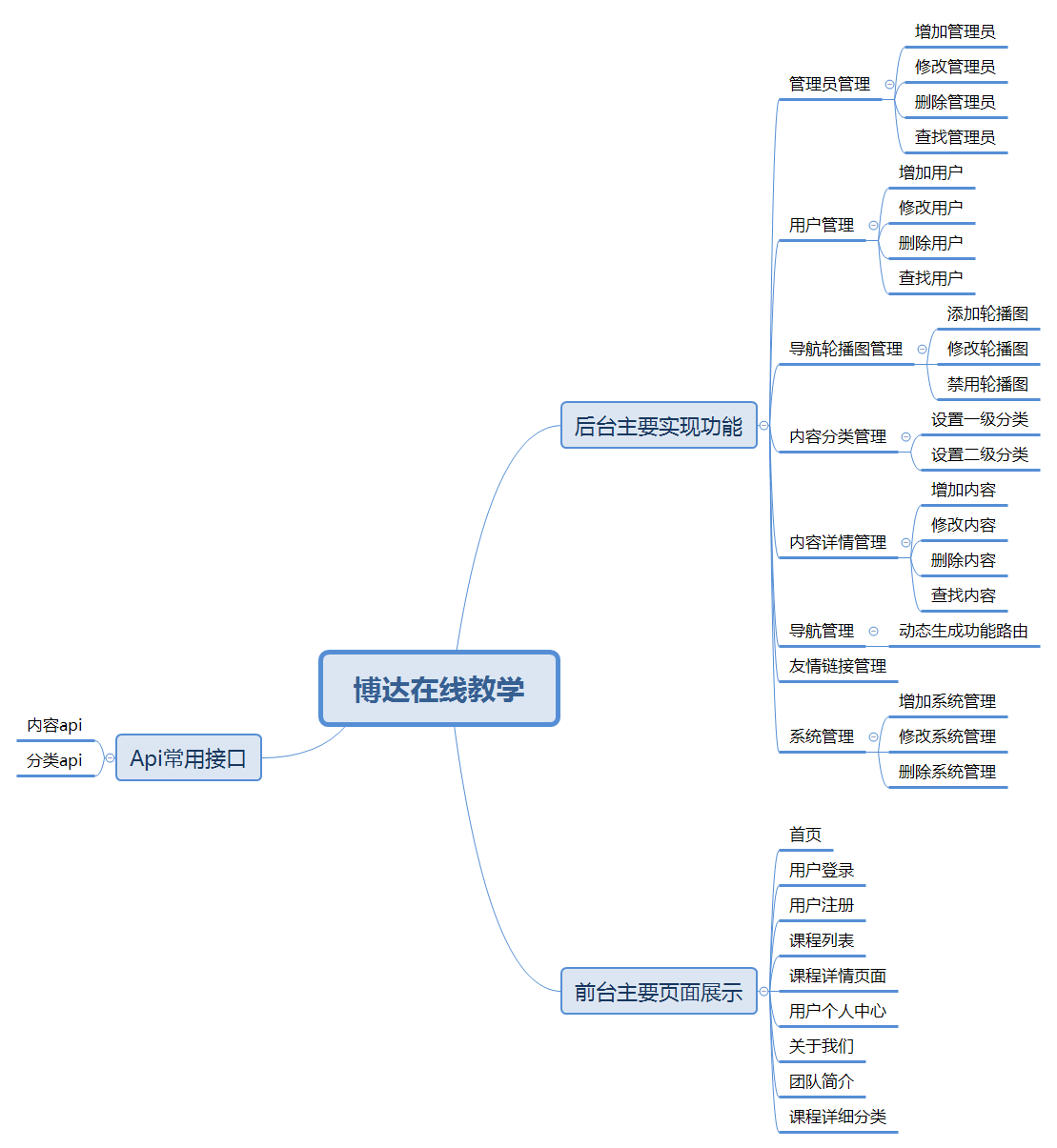


图4-2-1逻辑业务功能设计

## 4.3 博达在线教学网站功能实现简介

从上述介绍可知后台业务逻辑主要分为八个模块，下面将对八个模块的业务逻辑以及实现技术和思路进行详细讲解。

4.3.1管理员管理

管理员模块主要实现管理员的登录，删除，修改和添加功能。另一方面为了登陆体验，在管理员登录界面添加了字母匹配验证码功能，如下图所示。



图4.3.1-1管理员登录页面

这里主要使用了一个koa的svg-captcha中间件用于验证码的使用模块，主要代码使用如下图所示

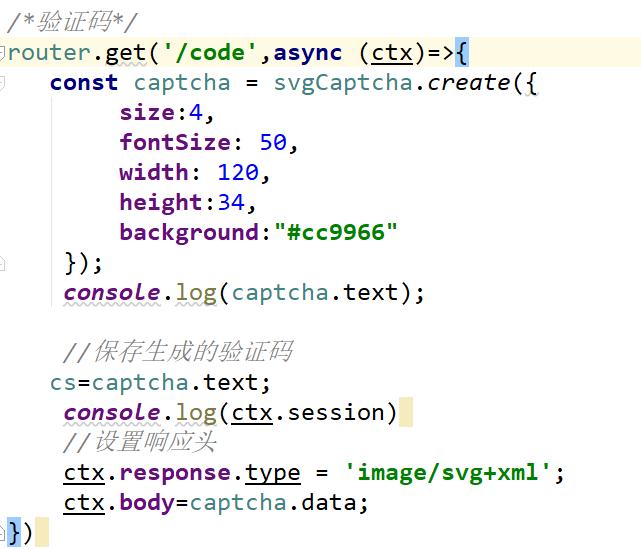


图4.3.1-2验证模块

除此之外对管理者的登录cookie进行了存储和消除，简单来说就是当管理员登录账户，cookie会自动保留管理员账户信息(保留的时间长短根据实际需要而定)，避免了不小心退出后再一次需要从新登录。当管理员退出时会自动清除保留的cookie信息，避免了存在安全性问题。这里是用到了koa的koa-session中间件，第一步需要按照：npm install koa-session，第二步导入模块：const session = require('koa-session');第三步使用，第四步使用session：

//全局配置变量

router.use(async(ctx,next)=>{

ctx.state.\_HOST\_="Http://"+ctx.header.host;

//登录继续向下匹配路由

if(ctx.session.userinfo){

await next();

}else {

//没有登录就跳转到登录页面

if(ctx.url=="/admin/login"||ctx.url=="/admin/doLogin"){

await next();

}else {

ctx.require("/admin/login");

}

}

})



图4.3.1-3配置session中间件

下图为后台管理员列表展示

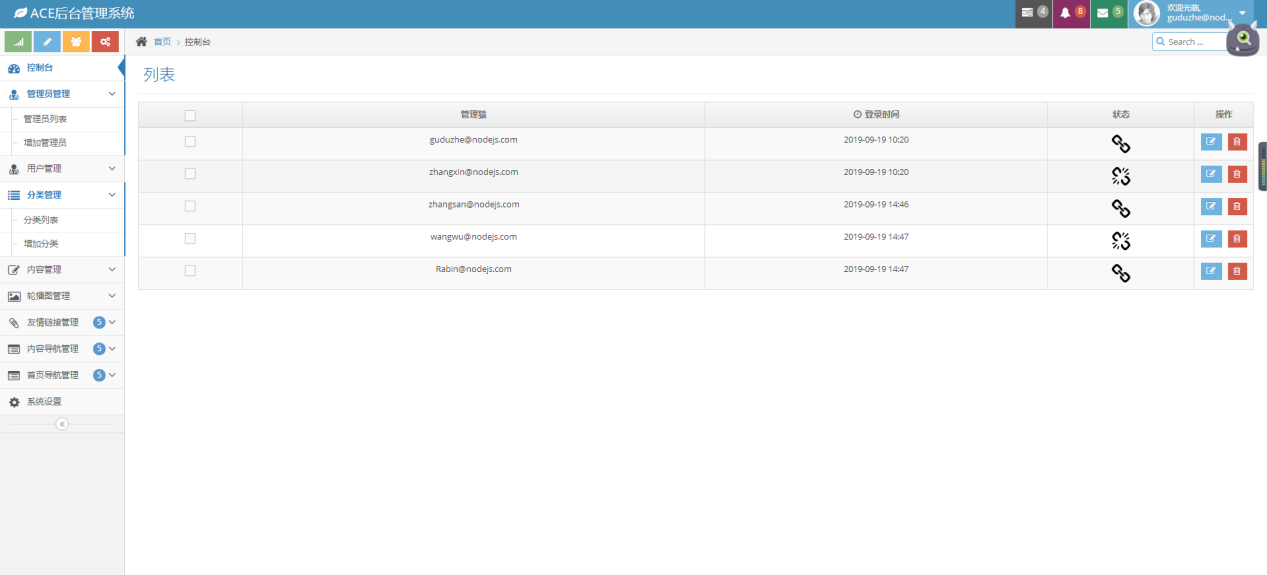


图4.3.1-4管理员列表

下图为管理员admin表的主要内容，包括管理员账户，密码，最后一次登录时间，以及状态字段。

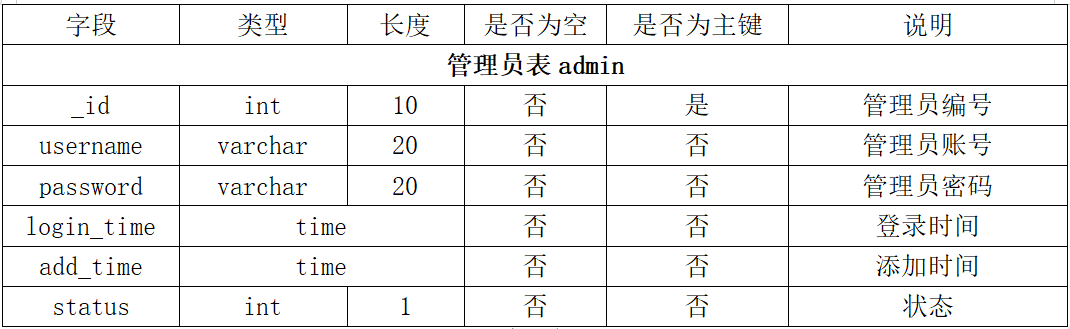


图4.3.1-5管理员表

4.4.2用户管理和用户注册

用户模块主要实现的内容：增加用户，删除用户，修改用户密码，修改用户的状态。

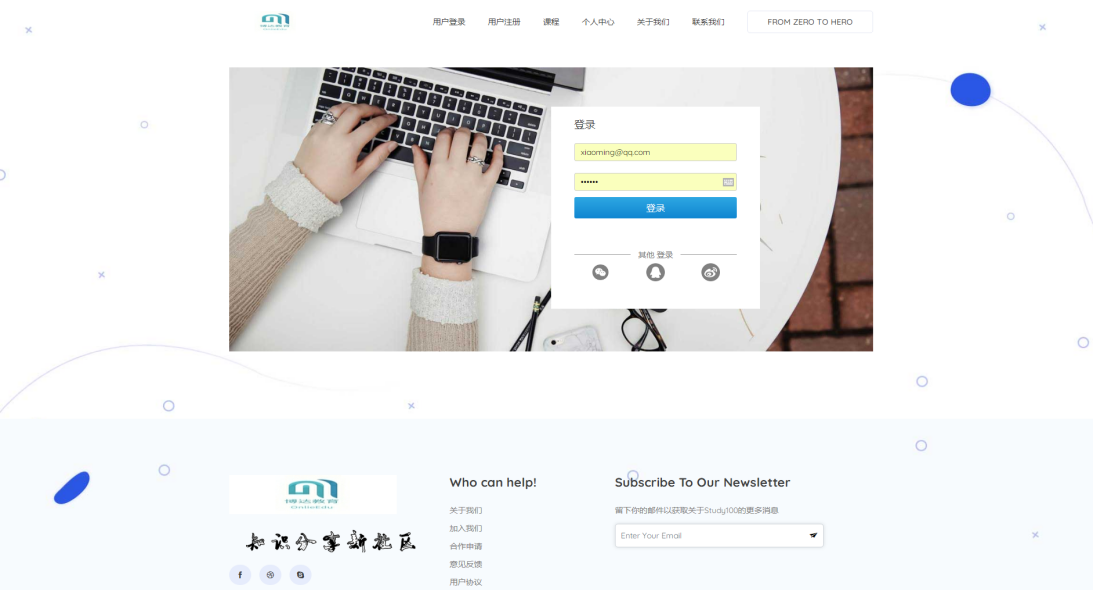


图4.3.2-1用户登录页面

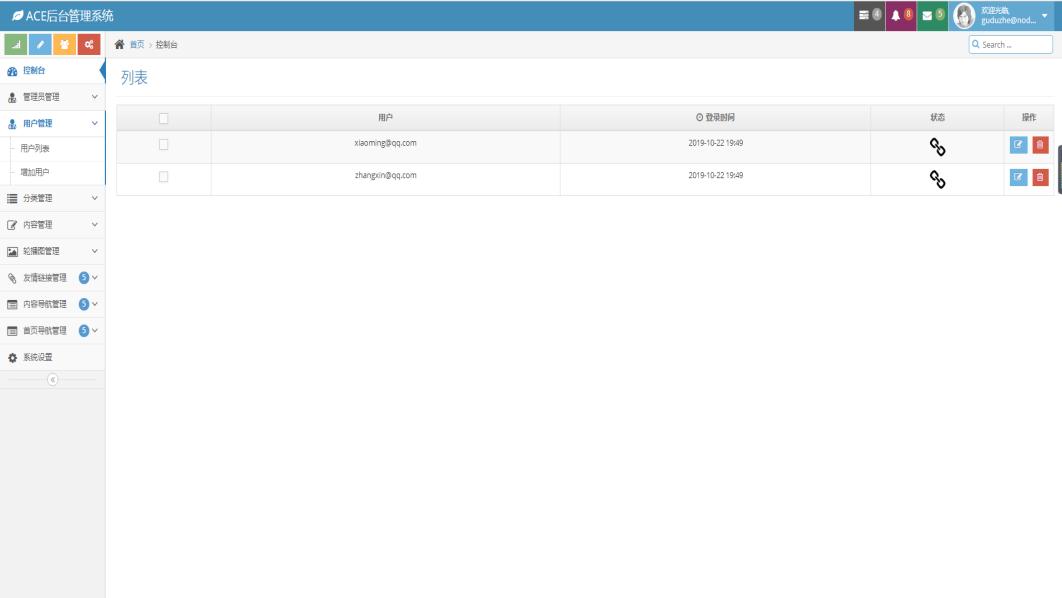


图4.3.2-2用户后台管理列表

下图所展示的表为用户表，包括用户的账号(必须为邮箱格式)，用户密码，最后登录时间，状态，用户性别，用户爱好，用户网名，和一些用户基本信息。

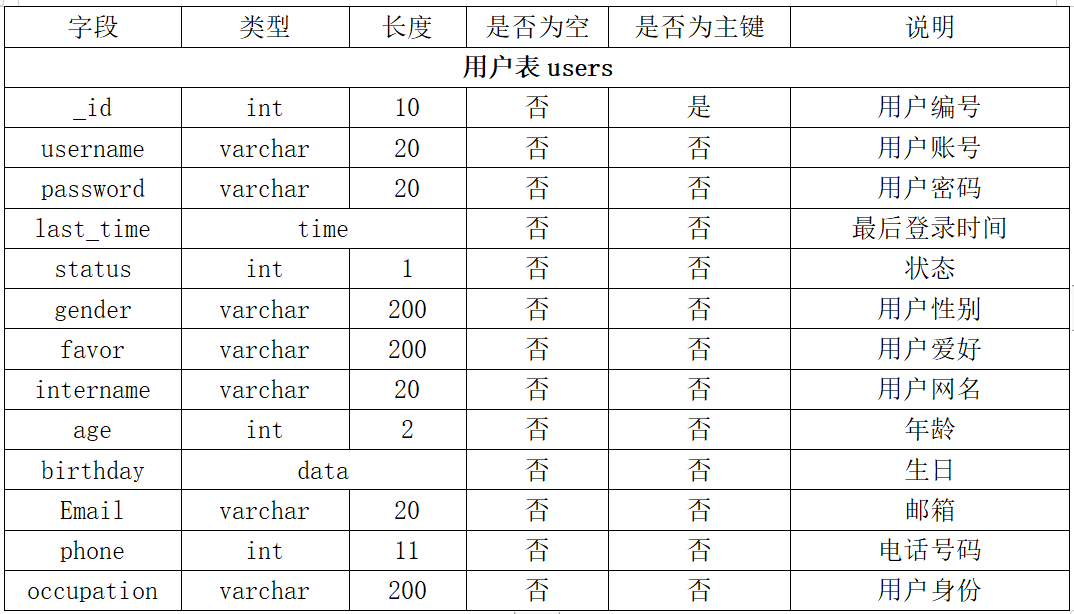


图4.3.2-3用户表

当用户第一次使用本网站的时候如果想看我们的教学课程，必须要进行登录，如果没有登陆账号，就需要用户进行注册，由于现实的需要，我们对用网站的安全性进行了多方面考虑。第一：如果用户不登录账号而要点击我们的课程进行观看，我们对这样的用户进行了拦截，跳转到登录页面。第二：用户注册时候的账号格式为了方便管理统一邮箱格式，电话号码和其他的一些基本信息必须填写，否则将注册不成功。用户的注册代码主要分为三部分，第一部分：获取用户注册表单的信息，第二步：验证表单数据的安全性和规则(不符合要求将会注册不成功)

第三步：在数据库查询当前要增加的用户是否存在，第四步：添加用户到users表中，详细代码如下





图4.3.3-4用户注册验证

4.4.3导航轮播图管理

导航轮播图的作用主要是管理网站首页的动态轮播图，可以根据需要随时改变图片，名称，链接地址，状态(去掉或者添加一个轮播图)。如下图所示。



图4.3.3-5导航轮播表

4.4.4内容导航管理

这里的内容导航是前端页面用来给用户登录，注册，个人中心，关于我们等进入模块的入口，也是动态的在后台进行设置的路由，如下图所示

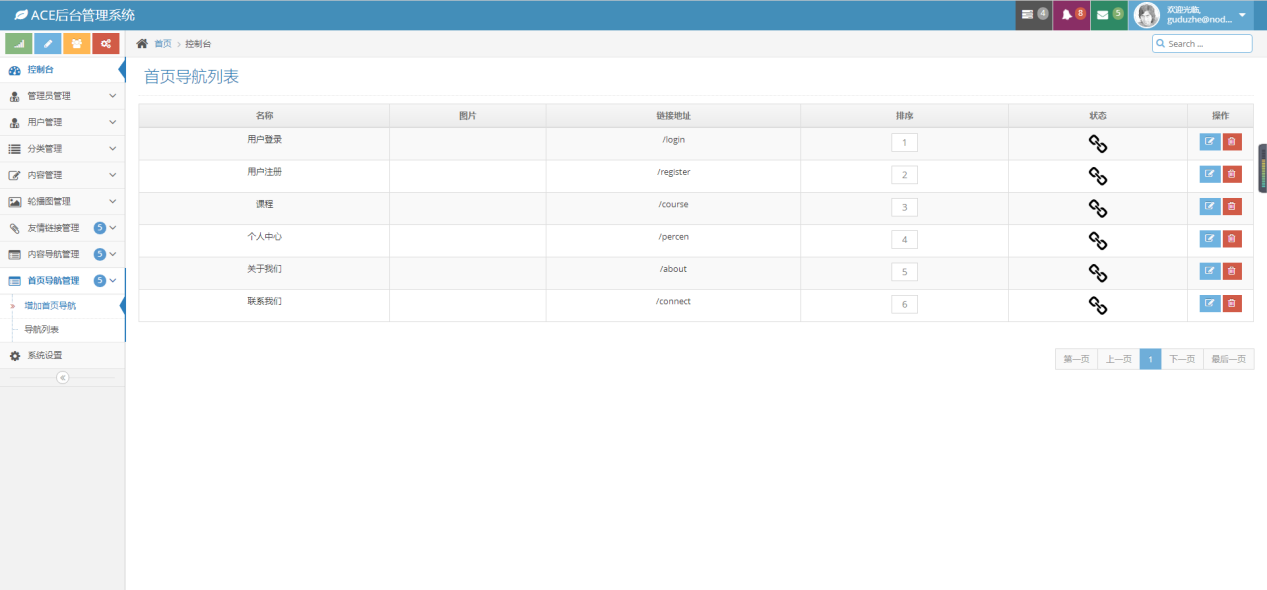


图4.3.4-1内容导航表

添加首页导航第一步要接收post通过form表单提交的数据，要注意在form单中配置enctype="multipart/form-data"，第二步把接受的数据插入到classes表中(安装Json格式),第三步实现跳转到列表。详细代码如下图所示。

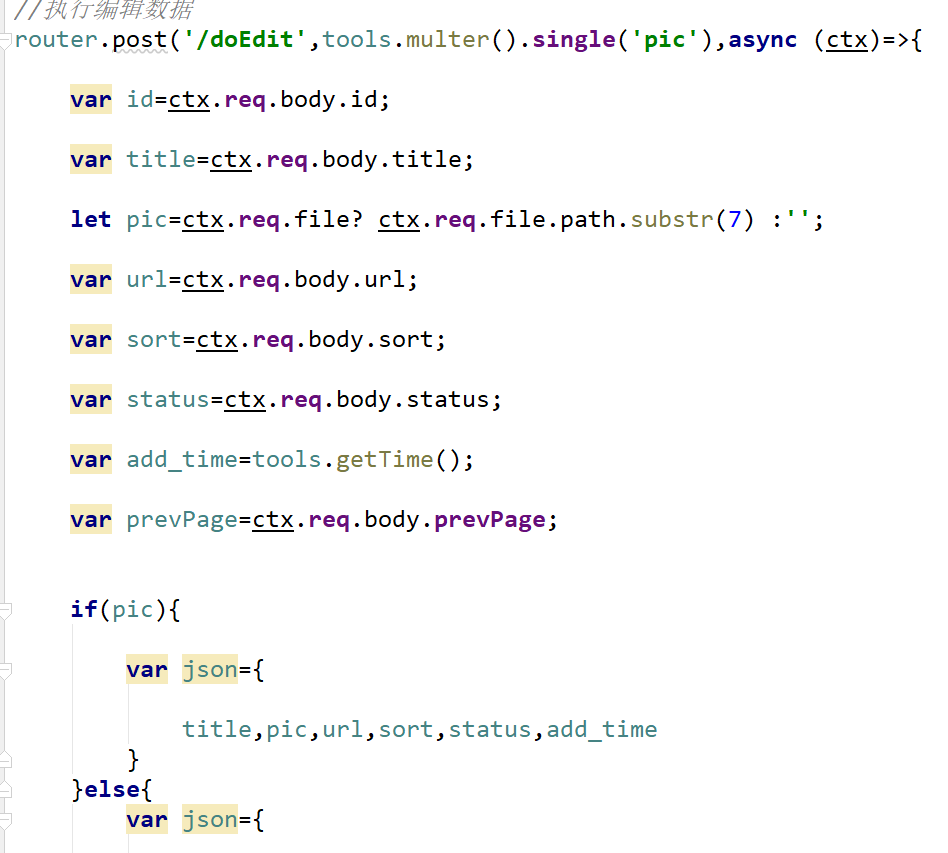


图4.3.4-2添加导航



图4.3.4-3 form表单提交数据

修改首页导航第一步要要拿到对应的id，第二步根据id去数据库中查找，并赋给result，第三步把对应的result里面的数据传给list数组，第四步在前端页面分别显示对应的内容。在编辑完数据是需要进行三步走，当跳转到修改页面之后，程序提交修改后的post表单。第一步重新把list中的数据进行重写，第二步然后把程序写好的数据写成json格式，再进行数据库的更新操作，第三步通过隐藏的表单域存储上一页的地址，修改之后跳转回上一个页面。





4.3.4-4修改数据

下图所示为首页导航表，主要字段包括课程标题，导航链接地址，链接时间，导航设置排序，以及状态。



图4.3.4-5内容导航表

4.4.5内容详情管理

内容详情管理模块功能主要有增加课程，删除课程，修改课程，课程顺序排序，状态的设置，添加课程视频等功能，对课程的添加也分为三步走，第一步要接收post通过form表单提交的数据，要注意在form单中配置enctype="multipart/form-data"，第二步把接受的数据插入到article表中(安装Json格式),第三步实现跳转到列表。详细代码如下图所示。

router.post('/doAdd', upload.single('img\_url'),async (ctx)=>{

let pid=ctx.req.body.pid;

let catename=ctx.req.body.catename.trim();

let title=ctx.req.body.title.trim();

let author=ctx.req.body.author.trim();

let pic=ctx.req.body.author;

let status=ctx.req.body.status;

let is\_best=ctx.req.body.is\_best;

let is\_hot=ctx.req.body.is\_hot;

let is\_new=ctx.req.body.is\_new;

let keywords=ctx.req.body.keywords;

let description=ctx.req.body.description || '';

let content=ctx.req.body.content ||'';

let img\_url=ctx.req.file? ctx.req.file.path.substr(7) :'';

let add\_time=tools.getTime();

//console.log(img\_url);

//属性的简写

let json={

pid,catename,title,author,status,is\_best,is\_hot,is\_new,keywords,description,content,img\_url,add\_time

}

var result=DB.insert('article',json);

//跳转

ctx.redirect('/admin/article');

})



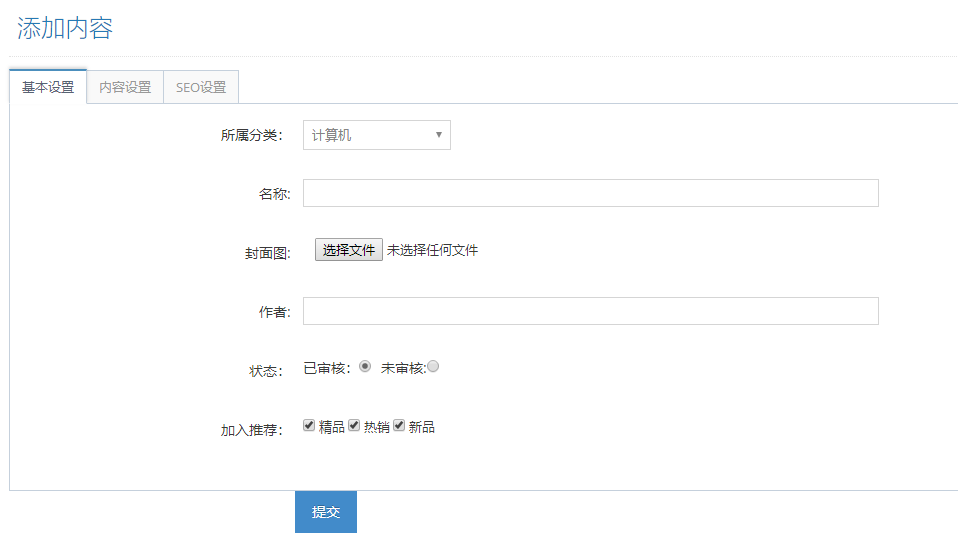


图4.3.5-1课程内容添加列表

对课程详细模块进行编辑和修改分为两部分，一部分是把要修改的数据显示出来，第二部分是对要修改的数据内容进行修改之后进行数据库更新。下面先介绍第一部分的代码思路：当管理员点击要修改的数据时候获取到对应的id，根据获取的id去数据库article表中找到这条数据，然后把数据内容全部显示出来。

详细代码如下

router.get('/edit',async (ctx)=>{

//查询分类数据

var id=ctx.query.id;

//分类

var catelist=await DB.find('articlecate',{});

//当前要编辑的数据

var articlelist=await DB.find('article',{"\_id":DB.getObjectId(id)});

await ctx.render('admin/article/edit',{

catelist:tools.cateToList(catelist),

list:articlelist[0],

prevPage :ctx.state.G.prevPage /\*保存上一页的值\*/

});

})

第二部分是对显示出来的数据进行修改并更新数据库article表中的数据。代码思路：首先获取通过第一部分拿到要修改数据的id所有的数据，管理员修改完数据之后，程序获取所有的字段数据，组成json格式，重新更新article表并返回列表。详细代码如下

router.post('/doEdit', upload.single('img\_url'),async (ctx)=>{

let prevPage=ctx.req.body.prevPage || ''; /\*上一页的地址\*/

let id=ctx.req.body.id;

let pid=ctx.req.body.pid;

let catename=ctx.req.body.catename.trim();

let title=ctx.req.body.title.trim();

let author=ctx.req.body.author.trim();

let pic=ctx.req.body.author;

let status=ctx.req.body.status;

let is\_best=ctx.req.body.is\_best;

let is\_hot=ctx.req.body.is\_hot;

let is\_new=ctx.req.body.is\_new;

let keywords=ctx.req.body.keywords;

let description=ctx.req.body.description || '';

let content=ctx.req.body.content ||'';

let img\_url=ctx.req.file? ctx.req.file.path.substr(7) :'';

//属性的简写

//注意是否修改了图片 var let块作用域

if(img\_url){

var json={

pid,catename,title,author,status,is\_best,is\_hot,is\_new,keywords,description,content,img\_url

}

}else{

var json={

pid,catename,title,author,status,is\_best,is\_hot,is\_new,keywords,description,content

}

}

DB.update('article',{"\_id":DB.getObjectId(id)},json);

//跳转

if(prevPage){

ctx.redirect(prevPage);

}else{

ctx.redirect('/admin/article');

}

})

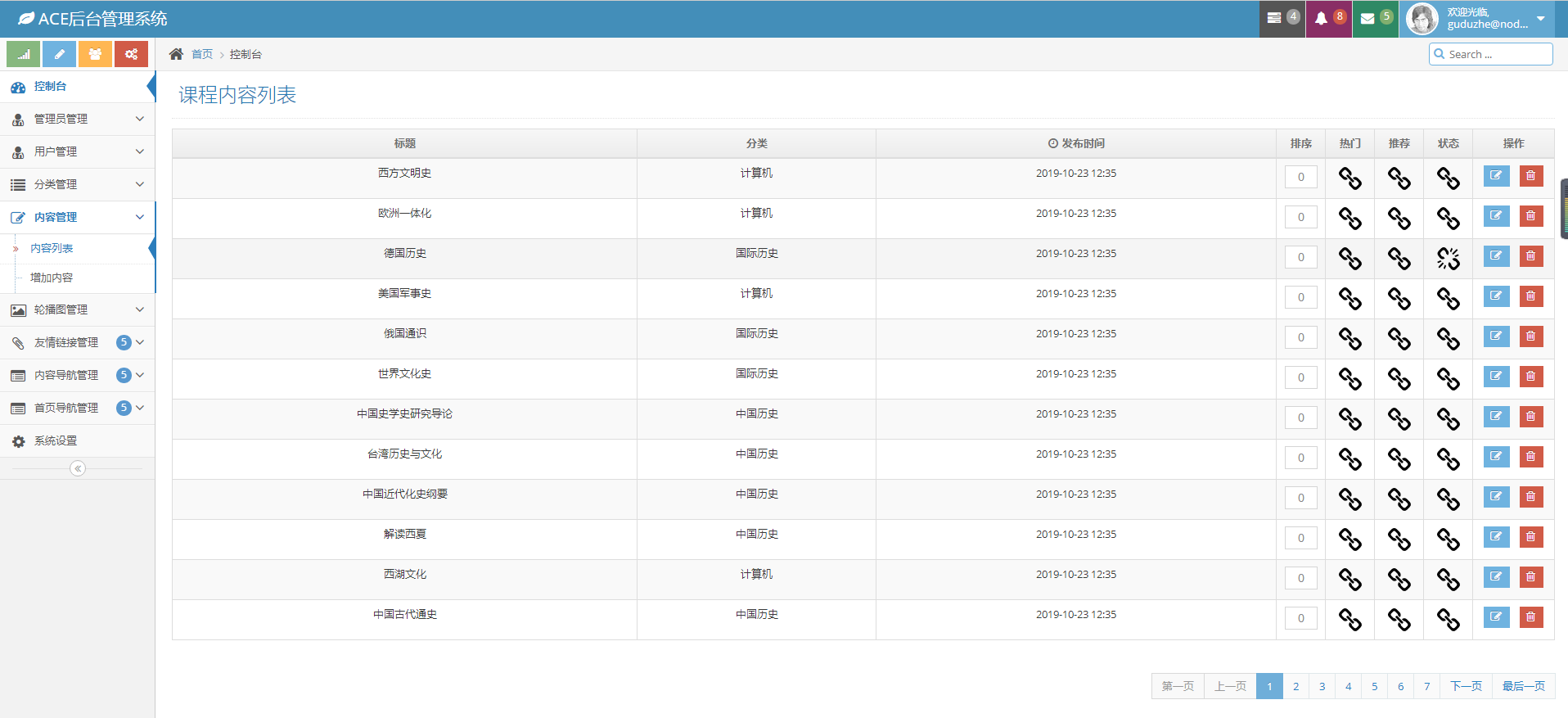


图4.3.5-2课程内容列表

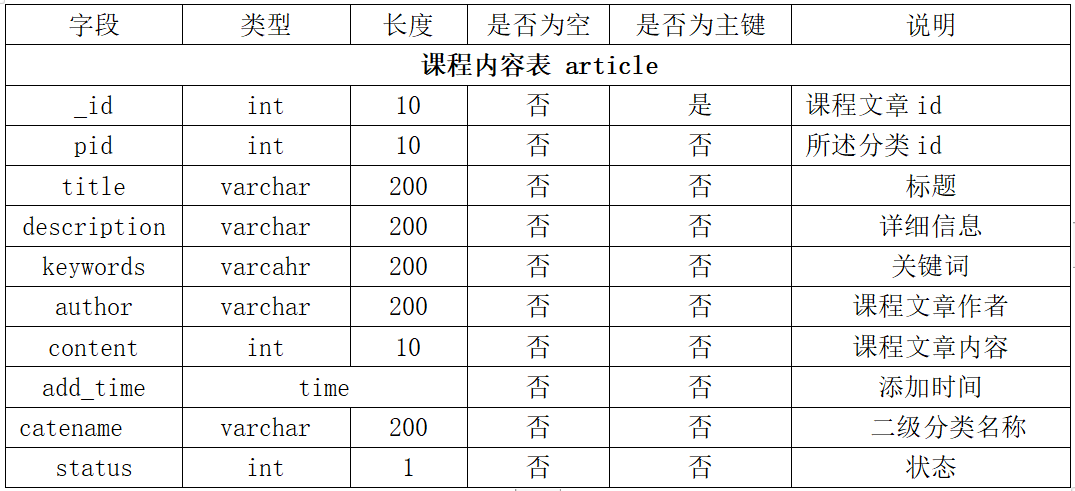


图4.3.5-3课程内容表

4.4.6内容分类管理

内容分类管理主要功能使用来处理分类和管理作用，例如计算机科目下面有前端课程，后端课程等，前端课程下面包括H5+CSS从入门到精通这门课，后端课程有node.js,springmvc课程等等，如下所示

var obj=[

{

'title':'计算机',

'list':[

{

title: '移动开发',

},

{

title: '网站开发',

}

]

}

]

一共分为二级目录。这样可以很好的处理当课程很多的时候寻找起来浪费时间的情况，当然这里需要说明一下，每一个二级分类都会带有一级分类的id字段，通过id字段来处理分类，下面将会详细分类二级目录的代码思路。首先是展示一级分类和二级分类，通过查询articlecate表中总数量，设置为result(设置为全局变量，嵌入到html代码中)，在html中通过遍历的方式把分类展示处理。

router.get('/',async (ctx)=>{

var page=ctx.query.page ||1;

var pageSize=12;

//查询总数量

var count= await DB.count('articlecate',{});

var result=await DB.find('articlecate',{},{},{

page:page,

pageSize:pageSize,

sortJson: {

'add\_time': -1

}

});

console.log(tools.cateToList(result));

await ctx.render('admin/articlecate/index',{

page:page,

totalPages:Math.ceil(count/pageSize),

list: tools.cateToList(result)

});

})



图4.3.6-1一级分类

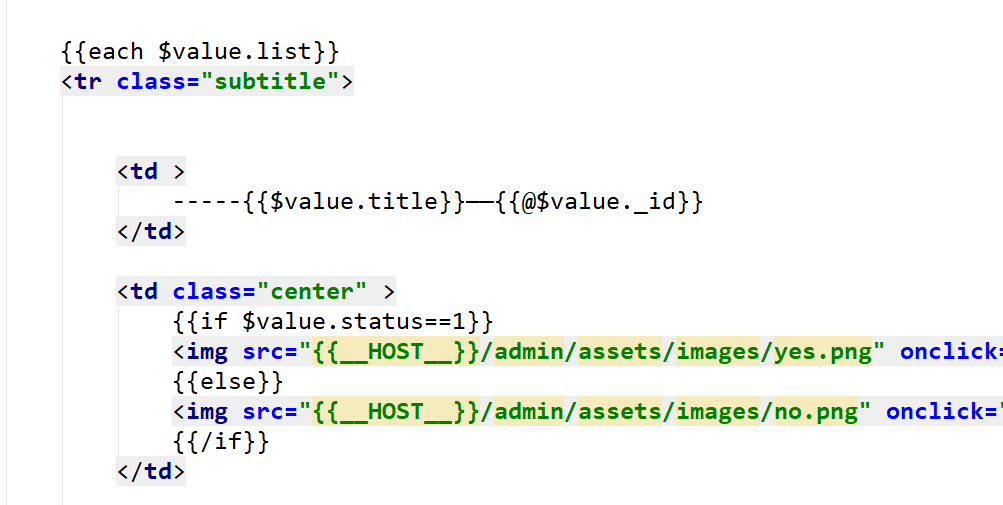


图4.3.6-2二级分类

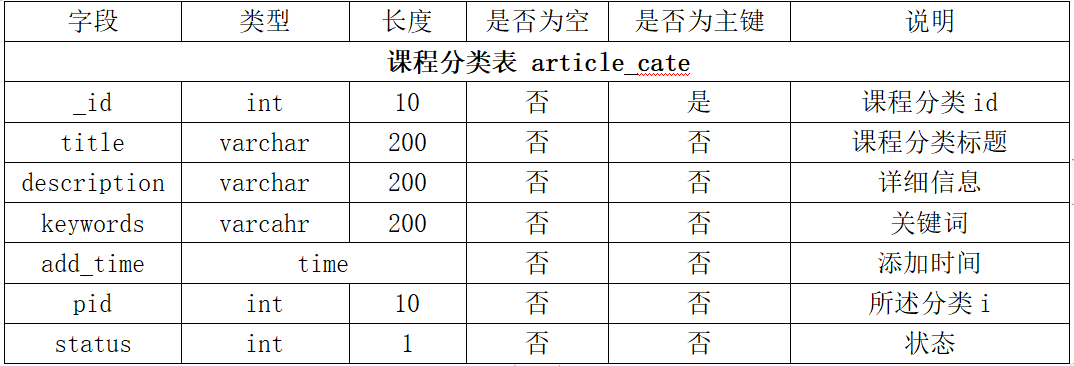


图4.3.6-3课程分类表

4.4.7网站系统设置管理

网站系统设置管理主要功能包括一些网站的基本信息，网站logo设置，网站描述(网站描述主要用来给浏览器进行检索网站的关键字，使用户可以快速并且准确的找到我们的网站)，一些基本的联系方式和公司地址等。

图4.3.7-1系统设置列表

下面是setting表的主要字段，包括id，网站标题，网站logo，搜索网站的关键字，网站描述，网站联系方式，地点等基本信息。

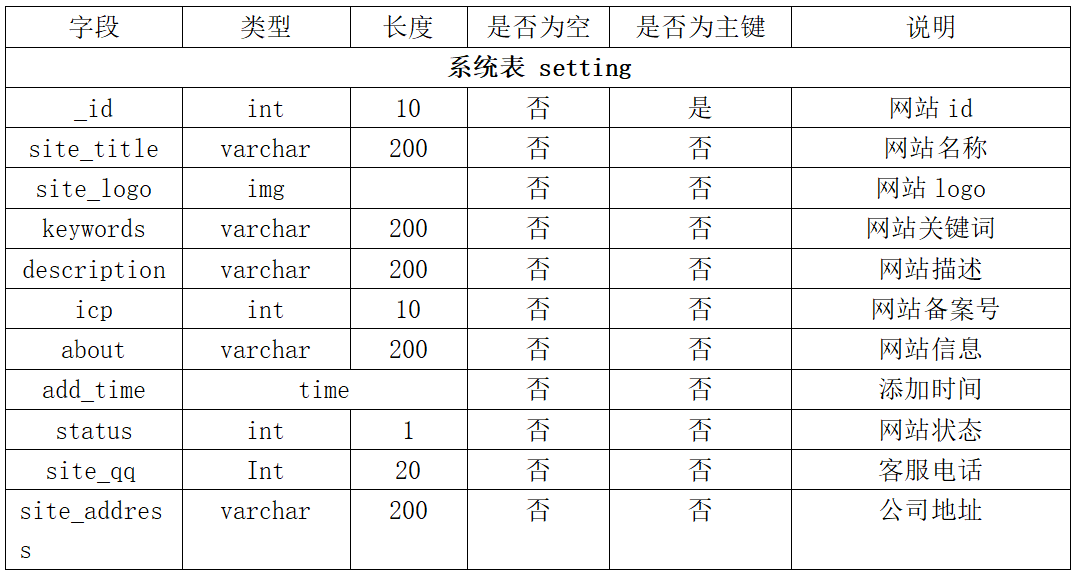


图4.3.7-2系统表

综合以上的主要七大模块可以看出了整个数据库表中的所有内容，同时你也会发现数据库中的表很少存在表与表之间的关联关系，这样就是mongodb数据库的特征属于非关系性数据库。下图是对数据库中表的一个总结。

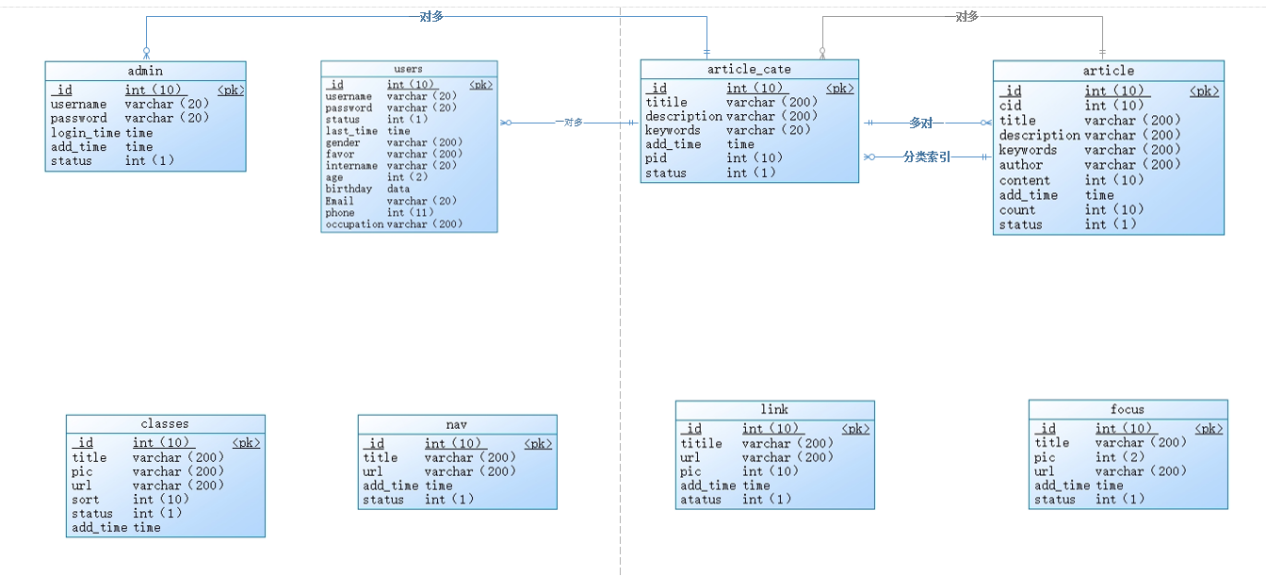


图4.3-数据库表关联

# **第五章 在线教学系统测试**

## 5.1系统测试的目标

软件的测试是在软件投入使用前对软件需求分析,设计规格和功能测试等进行最后的复查，是保证软件上线可靠性的关键步骤。软件测试的主要内容是根据需求和内部的结构和实现的功能进行设计若干测试用例，运行这些测试用例，找出程序中可能隐藏的错误。测试的最重要目的就是在系统上线之前，尽可能多地发现系统中的错误。

测试的目标：

1. 测试是为了发现程序中的错误。
2. 成功的测试不是没有发现错误，而恰巧相反成功的测试是发现迄今为止未发现的错误。

测试主要分为两种：如果知道程序的功能，可以通过测试每个功能是否正常使用；如果知道程序的内部代码结构，可以通过检查模块内部的设计，使用测试数据运行并比较和事先预想的是否一致。前一种方法称为黑盒测试，后一种方法称为白盒测试。

这一章将从两个角色(用户和管理员)来进行网站的测试功能，包括功能测试，性能测试，安全性测试，业务流程测试，合理性方面等进行测试。

## 5.2具体测试

这一节将从两个角度(用户和管理员)来进行网站的测试功能，包括功能测试，性能测试，安全性测试，业务流程测试，合理性方面等进行测试。

### 5.2.1 功能测试

首先对于用户而言主要的业务流程进行分析，如图5.1-1所示。

测试项：分为从使用者不同进行测试，分为两大模块：用户和管理员

用户测试：用户的登录，注册，观看视频，个人中心信息等等

管理员测试：分为八大小模块，分别从管理员，用户，轮播图，系统设置。导航栏等进行功能测试。

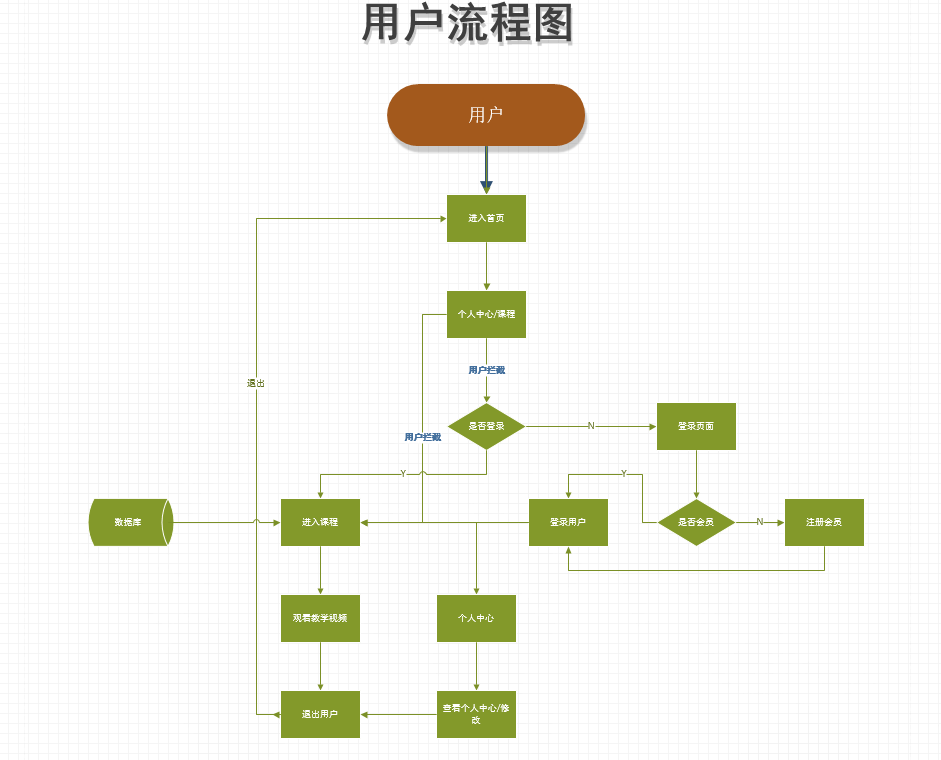


图5.1-1用户流程图

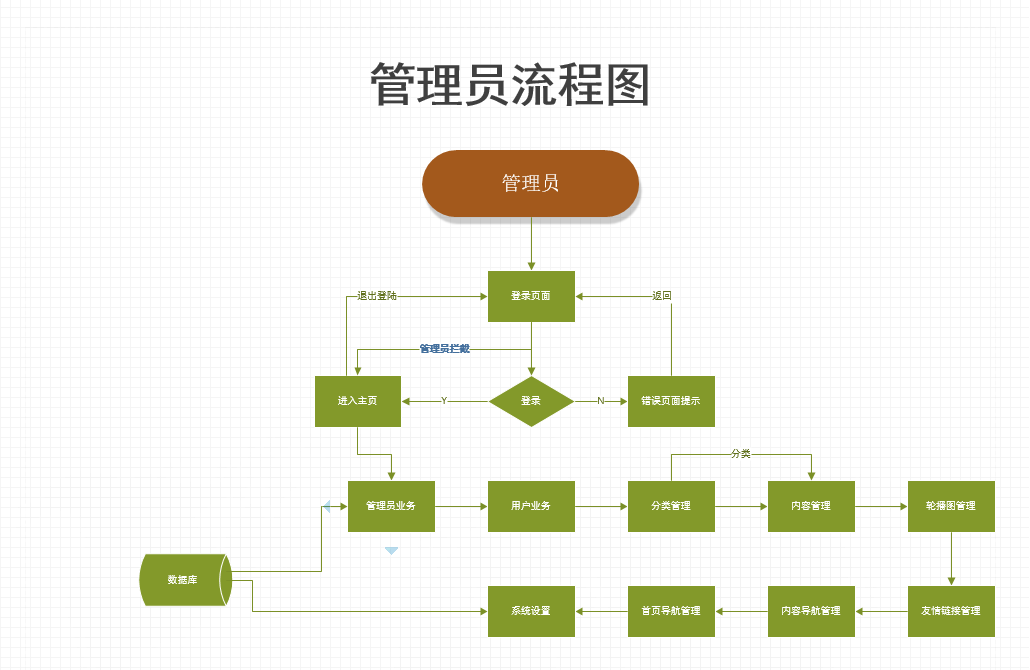


图5.1-2管理员流程图

### 5.2.2设置测试数据

|  |  |  |  |
| --- | --- | --- | --- |
| 功能项 | 功能编号 | 测试点 | 预期结果 |
| 用户登录 | 1. | 使用已注册的账号和密码 | 登陆成功，并可以观看视频和进入个人中心 |
| 2. | 使用已注册的账号，密码错误 | 登录失败并提示用户密码错误请重新登录 |
| 用户注册 | 3. | 注册账号非邮箱格式，其余格式都正确 | 注册失败，注册账号不符合要求，请重新注册，并跳转到注册页面 |
| 4. | 注册账号格式正确，注册密码和确认注册密码不一致，其余格式都正确 | 注册失败，注册账号密码不一致，请重新注册，并跳转到注册页面 |
| 5. | 注册账号格式和密码都正确，其余基本信息有漏填写 | 注册失败，请填写完用户信息，请重新注册，并跳转到注册页面 |
| 6. | 注册账号和密码都正确，并且信息都填写了，但是电话号码不等于11位 | 注册失败，请输入有效电话号码，请重新注册，并跳转到注册页面 |
| 7. | 注册账号和密码都正确，并且信息都填写了而且都符合要求，电话11位，邮箱也正确 | 用户注册成功 |

表5.1-3用户白盒测试

|  |  |  |  |
| --- | --- | --- | --- |
| 功能项 | 功能编号 | 测试点 | 预期结果 |
| 管理员登录 | 1. | 使用已注册的账号和密码 | 登陆成功，进入后台管理 |
| 2. | 使用已注册的账号，密码错误 | 登录失败并提示管理员密码错误请重新登录 |
| 3 | 使用已注册的账号，密码正确，输入验证码不匹配 | 登录失败并提示管理员验证码输入不正确请重新登录 |
| 管理员操作管理员信息 | 1. | 添加新的管理员 | 添加成功 |
| 2. | 修改管理员 | 修改成功 |
| 3. | 删除管理员 | 删除成功 |
| 管理员操作用户信息 | 4. | 删除用户 | 删除成功 |
| 5. | 禁用用户 | 禁用成功 |
| 管理员操作课程添加信息 | 6. | 添加课程的基本信息和视频 | 添加成功 |
| 7. | 修改课程的基本信息和视频 | 修改成功 |
| 8. | 删除课程的基本信息和视频 | 删除成功 |

表5.1-3管理员黑盒测试

# 第六章 总结与展望

到这里本文也该结束了，基于node.js打造的在线教学网站，分别从用户和管理员的不同角色对网站前后端进行了设计和功能的实现，用户的登录，注册，个人中心，个人信息查询，学习课程等基本的功能。后台管理部分分别从九大模块对网站进行了管理和动态运维，实现了管理员的增删改查，用户的增删改查，轮播图的设计和修改，导航栏的动态设置和修改，课程的分类和详细课程信息的增删改查，课程详细信息表和课程分类表通过id进行了关联，也是本项目唯一的实现表与表之间的关联。还有对网站的系统设置，动态的添加logo，网站地址，电话，网站描述和关键字，这主要用来浏览器进行检索准备的，方便用户通过关键字来搜索到我们的网站。

对于本项目的设计和功能的实现前前后后也花了很长时间，也走了很多弯路，从一点都不了解node.js这个技术，到一点一点的学习，一步一步的走到现在，也感谢坚持的自己，慢慢的把项目给架构了出来。项目做完了自己也学会了很多，懂得了很多很多，以前不是很懂得东西和词语，都是听老师说的，自己并不是懂。就比如什么叫“前后端分离项目”,以前的理解只是肤浅的停留在表面，当自己去真正的接触前后端项目的时候才真真切切的懂得了什么才叫前后端项目，懂得了如何把数据库表中的数据遍历出来嵌入到html页面里面和配置全局的前端变量来渲染全端页面。明白了前后端的各自的主要工作，前端主要是来渲染后端传过来的数据，把数据嵌入到html，javaScript和css页面中，后端主要是来实现模块功能实现与前端进行交互的作用，前后端彼此是相互独立的，但又存在关联。但是，由于多方面的原因，本课题的研究还有很多不足之处，今后的工作主要体现在以下一些方面:

(1)前端的页面设计还有待提高，用户友善做的不够好。

(2)cookie的设置也存在问题，不管是用户的登录还是管理员的登录之后都会进行保存，并且可以互相访问前后端，用户只要登录了账号不仅可以访问课程页面同时如果知道后端地址则不用登录就会进入到后台管理员后台，给网站带来的很大的安全性问题。

通过几个月的付出完成了这个在线教学项目，一方面转变了开发网站的思路和看法，也懂得了如何去构建一个网站和准备工作。当然也存在不足的地方，有些功能方面的实现是存在的安全性问题和缺少健壮性，并不适合大量的用户进行同时访问会导致网站的崩掉，还存在代码的大量重复和复杂性，代码的思维能力有很大的提升空间。希望在日后的学习中有意识的去提高代码思维和加强自我学习的能力。

# **致谢词**

本论文最终能够顺利完成，首先要感谢我的指导老师曾世邦自始至终给予的关系和技术指导，不管是论文的开题，选题，写作中遇到的问题，曾老师都悉心指导，导师的一丝不苟的态度和治学作风以及求实的工作太多都给了我深深的启发和做人的道理。另一个要感谢我的搭档在合格过程给予我的帮助和灵感，一个优秀有潜力的前端设计师，对美感，视觉用户体验都有独特讲解。

最后，再次感谢所有关心和爱护过我的老师、亲人、同学和朋友！发自内心的感谢大学期间传道授业授惑的老师们，谢谢你们把我从一个对电脑都没有接触过的高中生培养成一个合格的程序员，谢谢你们的付出，谢谢你们教会我做人的道理。

# **参考文献**

[1] 黄综捷，李超编著.电子商务概论[M].北京：中国财经经济出版社，2002

[2] 张海宽等编著,电子商务概论[M]，北京：电子工业出版社，2003

[3] node.js官网：[https://node.js.org/static/documents/2016-survey-report.pdf](https://nodejs.org/static/documents/2016-survey-report.pdf)

[4] 美George Reese著.石永薪,宋隆译.《JDBC与Java数据库编程》北京:中国电力出版社,2002(3),34.

[5] 徐心平，远程教育及其分类.计算机世界，1999(7)

[7] 新叶网络教学平台. http：//202.112.88.32/aboutvclass/index.html 北京师范大学现代教育技术研究所

[8] 廖泉文,人力资源招聘系统[M].山东人民出版社,1999.

[9] mongodb官网https://www.mongodb.com/

[9] node.js官网https://node.js.org/en/ m/