地表最强慕课网v0.3

**一、创意来源及背景**

如今web领域大放异彩，但是Web方面的知识太零散，我们想利做这样一个网站来提供大众一个分享学习web领域的平台。然后不同的年轻人可以在上面分享和交流。

加上当前Nosql 和 nodejs 的快速发展和优越的性能，我们决定用nodejs结合express框架写后台，用mongodb做我们的数据库，用bootstrap样式写前端。来搭建一个灵活可交互的慕课网站。

项目目前已经部署在云服务器上方便查看 (www.huangqingyi.cn)

6月专家指导后，我们将导航和主页进行了修改，让页面与用户权限结合，显示的更加灵活

**二、系统功能架构图**

**项目采用MVC模式，模型、视图、控制器彻底分离开**

**view视图层:**

**layout.jade 公共头**

**index.jade 首页（包含登录和注册两个模态框）**

**admin.jade 后台录入页**

**category\_admin.jade 分类录入页**

**categorylist 分类list页面**

**list.jade 文章list页面**

**detail.jade 文章详情页**

**results.jade 分类详情页**

**signin.jade 登录页面**

**signup.jade 注册页面**

**aboutUs.html 关于我们**

**search.jade 查询页面**

**404.html 404页面**

**Schemas定义模型 models实现模型 模型层:**

**Category.js 文章分类模型**

**Comment.js 评论模型**

**Passage.js 文章模型**

**User.js 用户模型**

**Controller 管理层：**

**Category.js 调用分类的各种方法**

**Comment.js 评论的方法和处理逻辑**

**Index.js 首页处理**

**Passage.js 文章相关处理和方法**

**User.js 登录注册逻辑**

**App.js 项目入口**

**App.js require 了大部分依赖 连接MongoDB：27017端口的shumei数据库 的以及启动3000端口，**

**路由管理 config/router.js**

**数据库 mongodb data/db**

**部分静态文件Public**

**Libs 放的是jq 和bs库 和iconfont图标和一些自定义css样式**

**Js 放的是一些 删除，评论时对页面样式控制的脚本**

**Img 放的是经过压缩后的图**

**NPM的Package.json**

**里面放的是项目依赖文件目录，经过命令行npm install 会下载依赖文件，没有经过命令行NPM install 是无法运行 node app的**

**Gulpfile**

**为了方便查看，未对代码进行压缩，图片已经压缩过了**

查询逻辑：正则匹配数据库里的已有数据，并且render给页面，并且运用分页展示；

**三、团队分工界面**

（展示不同模块界面及对应代码，注明组内分工情况）

前端视图层，前端的页面，动态操作时候的脚本样式。(前端页面和bs样式和jq脚本)由唐婷婷完成

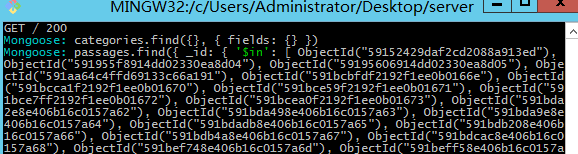
Nodejs相关后台逻辑，mongodb数据库操纵和前后台交换由黄卿怡完成

**四、代码质量审查机制**

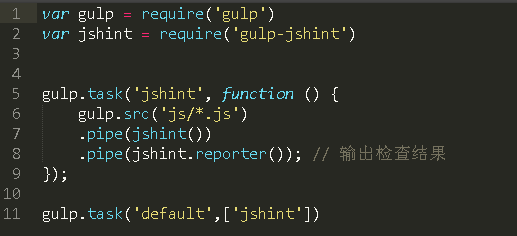
用包管理工具加载依赖项 morgan 对项目有一个log日志的输出，在项目整个过程操作和bug都会打印到git cmd 上根据问题修改

App引入

C:\Users\Administrator\AppData\Roaming\Tencent\Users\297947067\TIM\WinTemp\RichOle\{O{5A@(EWH[[N(X%H(F(9FE.png



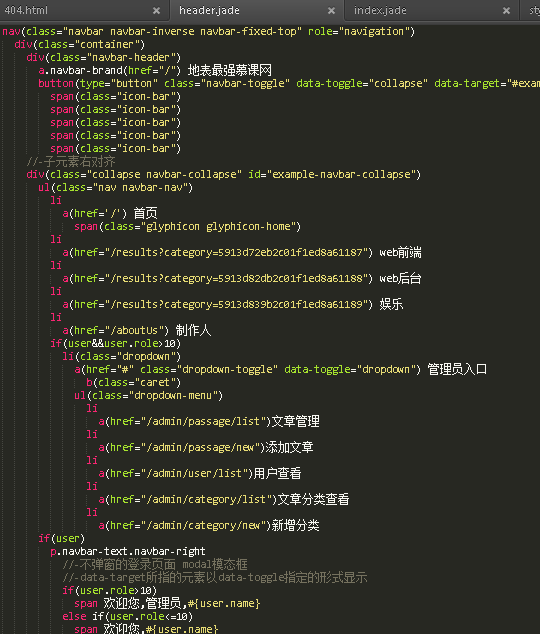
Gulp工具用来检查js代码是否有错误



**五、技术亮点**

**1**

**我们前端视图都采用的jade模版引擎，让代码的可维护性，可扩展性，其继承性让前端大大减少代码量，同时减少了dom操作的异常，运用bootstrap，让前端开发更加高效**



2

用bcrypt.js 对用户保存密码之前将密码和当时设置的hash值以及结合系统时间对用户密码加盐

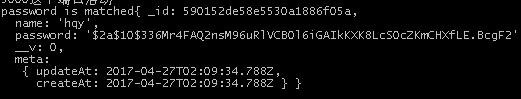
代码：在user模型定义的时候

引入bcryptjs 设置hash值为10

C:\Users\Administrator\AppData\Roaming\Tencent\Users\297947067\TIM\WinTemp\RichOle\QDO3%XQ[{K$5YMANVI1H7TF.png

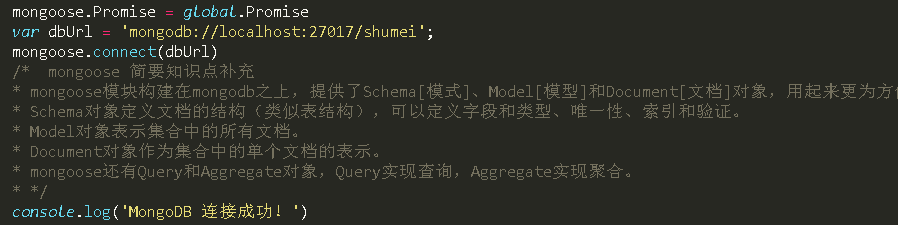
在存储用户密码前都会执行



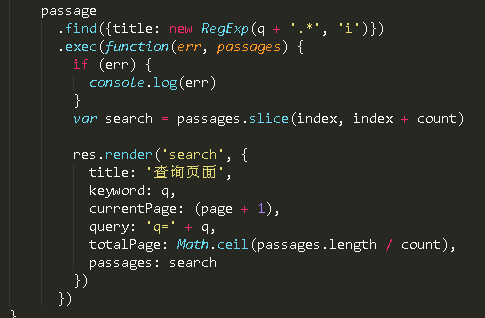


3，采用Nosql的非关系型数据库，提高程序的并发性，针对nodejs 用mongoose 插件，mongoose为mongodb封装了几个对象，能够让node 通过操纵这些对象从而能够方便的操作mongodb,

C:\Users\Administrator\AppData\Roaming\Tencent\Users\297947067\TIM\WinTemp\RichOle\WFV)58`[$LVGGOL)]}M6`DF.png



4，正则表达式从数据库挖掘用户想要查询的数据，实现模糊查询



**六、难点攻关**

遇到的困难实在是太多了，攻克难点主要通过google 百度 和github 和nodejs社区搜索，

例如：开始做的时候：我们不知道如何从页面上拿到用户请求信息中的\_id 老是提示不存在，好久才发现原来是传入这个对象的时候传错了，一个错误的对象，

对jade模版不熟悉，老是出现\*\* is self closing and should not have content错误，在逛了github里的社区，通过国外大牛的帖子解决了各种格式问题。

数据fetch并且渲染到页面的，传入从数据库里面取出的值，却怎么也不显示。这个就分了好几类情况，1存储的时候模型数据与现在输出定义的格式的不一样 2模型写错，根本就没存到数据库里3全局变量污染，因为我们都是萌新，所以前后端交互不是很熟悉。这次让我们学到了不少东西

用户登录逻辑，用户登录是拿到库里的进行对比，对比上了就登录成功了，然而我们上网搜了不少方法，由于node相关的资源不是很多，还是没有php java这些老语言存在时间长，各种各样的方法基本都是针对java 或者php. 所以通过外国人的网站，我们才获得了想要的东西，最后还是采用bcrypt自带的compare方法，通过回调函数反馈了是否match,再通过与服务端建立的session，从客户端浏览器拿到cookies 保持用户与服务器的持续会话。

期间出了各种各样问题，好在最后通过互联网和大佬，解决了，

**反思这些难点应该都是自己经验不足导致的。**

**七、软件使用**

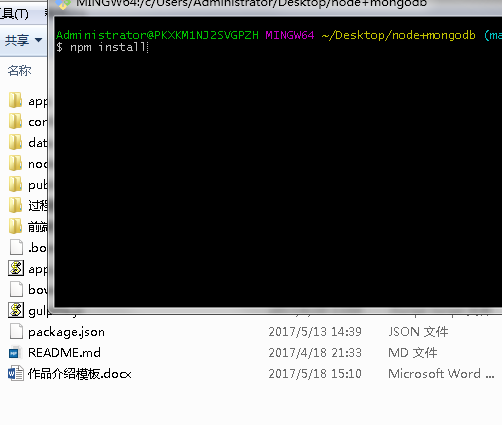
1.我们项目需要安装node js v6.10.0

2.下载数据库 mongodb for windows就行

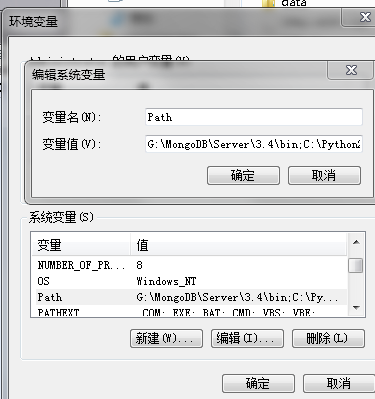
（<https://www.mongodb.com/download-center?jmp=nav#community>）

3.一般安装了nodejs后都会自带npm包管理工具 对项目根目录进入命令行

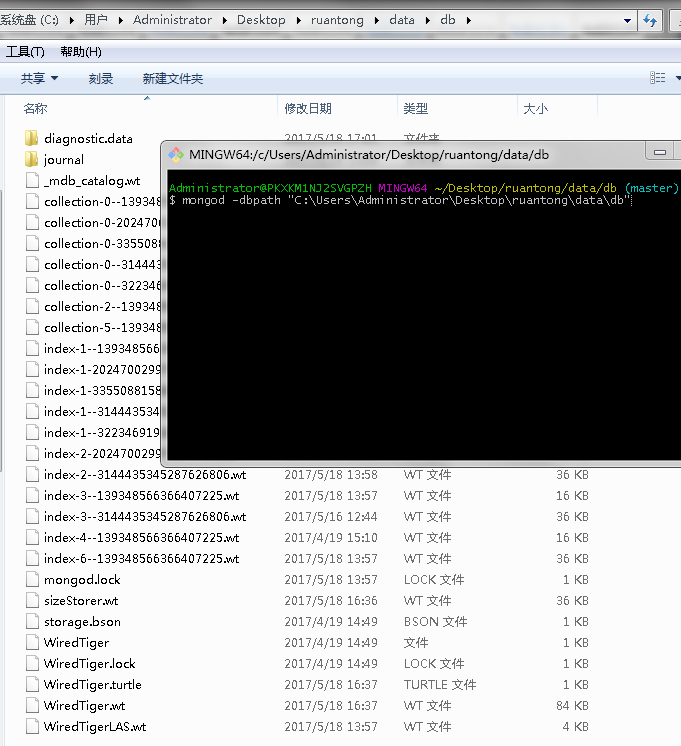
输入 npm install 就会根据package.json 下载依赖包了(如果很慢推荐使用淘宝NPM镜像,如果出现gyp错误不管，因为它是c++依赖)

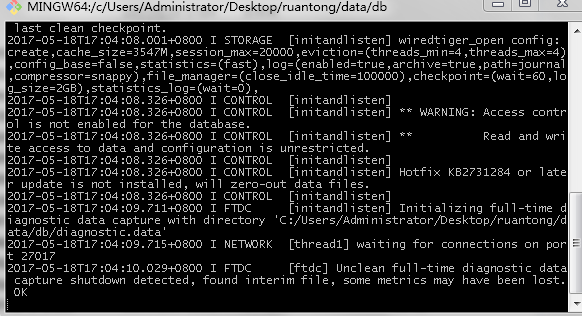


4我们把mongodb 添加到环境变量



5进入项目根目录的data文件里的db文件将地址复制下来

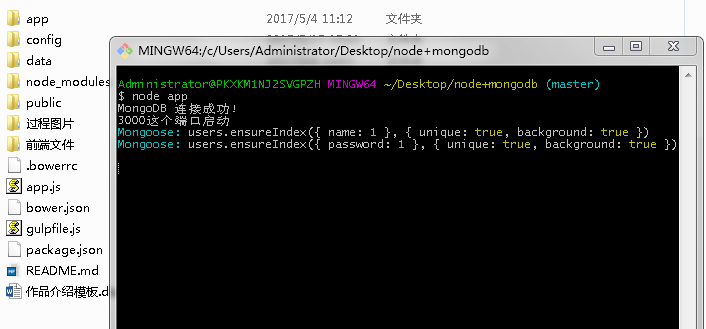




出现这样就成功了

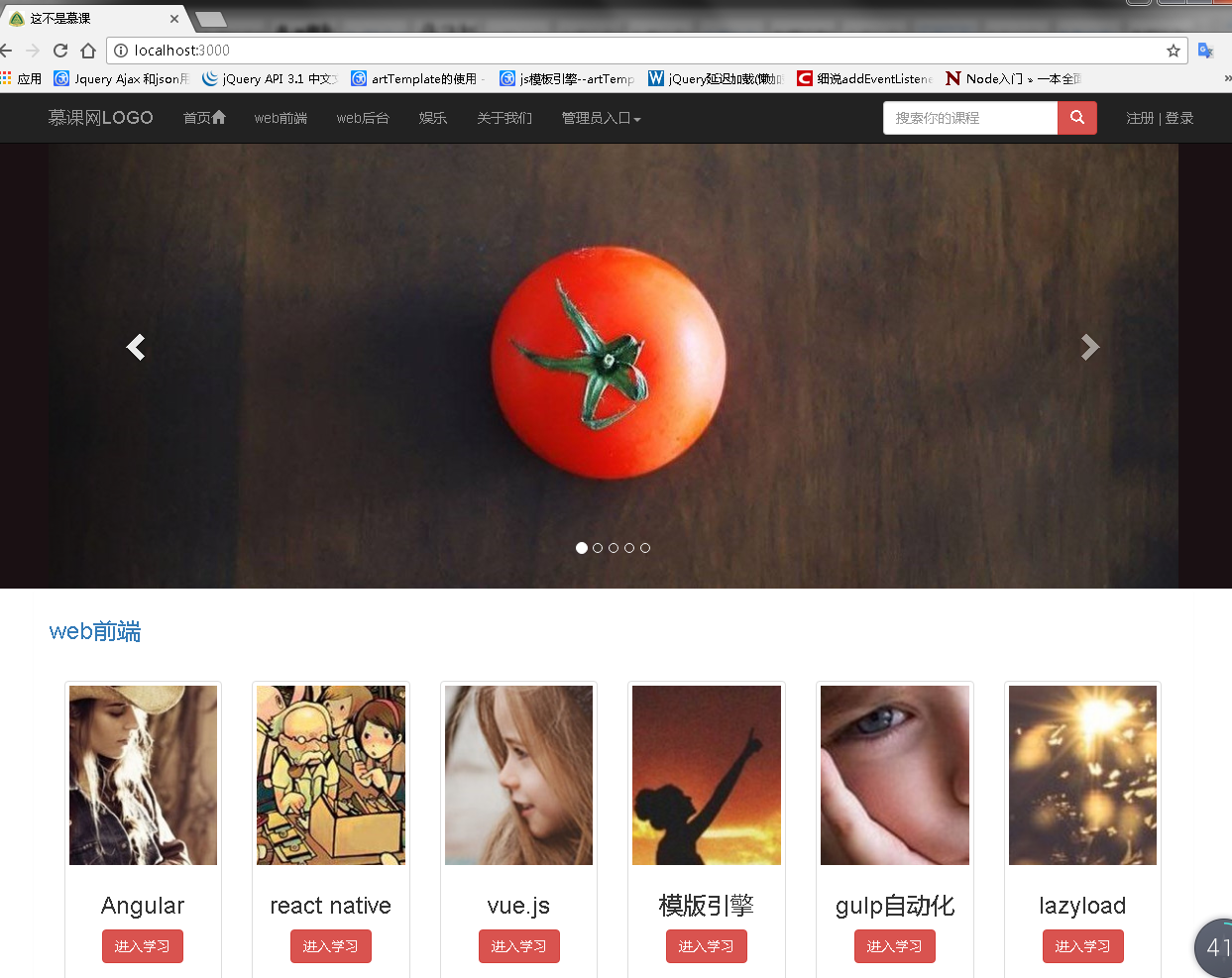
然后我们进入根目录

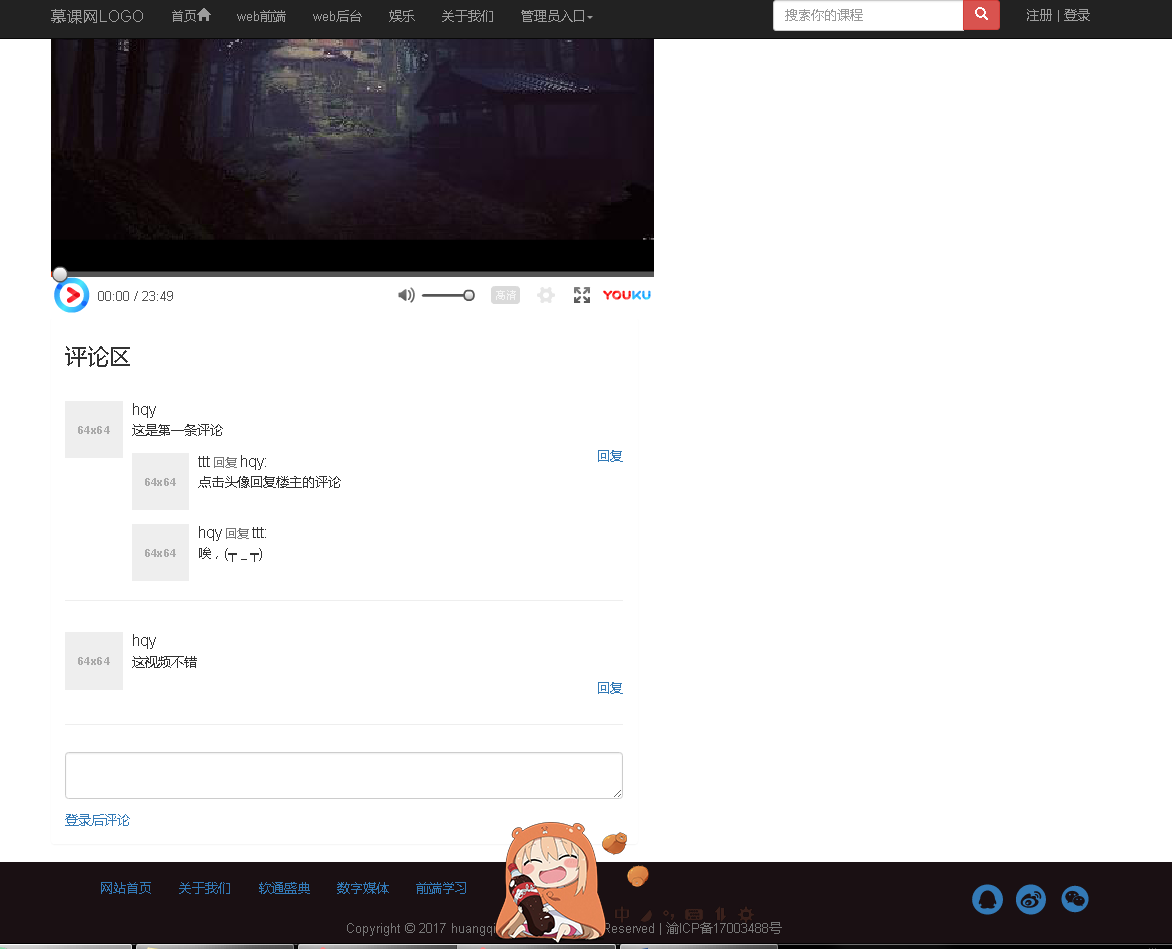
命令行 node app



打开浏览器 localhost:3000就可以看到作品了

（我们的作品已经部署到了[www.huangqingyi.cn](http://www.huangqingyi.cn)上，提供查看）





录入新数据格式：



封面图片可以是网上的，也可以是本地的/static/img/pt1-pt16.jpg