**选题的意义与目的**

随着计算机网络的发展，技术的日新月异，博客作为一种新的交流方式，是作者思想的结晶。现在已经成为网络日志必不可少的一种工具，也是一种简单有效提供网络用户之间进行在线交流的网络平台，通过其可以结交更多的朋友，表达更多的想法，它可以随时发布日志，方便快捷。因此想搭建一个属于自己的一款精美的博客应用。

采用nodejs+mongodb+jade+bootstrap框架去开发一个个人博客应用，实现文集、标签、自定义URL，响应式布局、代码高亮的富文本编辑器、评论，回复功能。Node.js技术使JavaScript来实现服务端的应用成为可能，其中Node.js采用事件驱动、异步编程，并且为网络服务而设计的特点,使它可以简单的实现web应用，而Node.js的单线程运行模式，是一个基于GoogleV8引擎的Javascript运行环境，使Node.js实现的应用并发量大，占用内存少，性能佳。综合Node.js技术特点，选择这个技术搭建一个博客应用为首选。

通过搭建一个属于自己的博客应用，掌握Node.js开辟了另外一条道路跟其他程序员进行技术上的交流，解决自己在项目中遇到的问题，在交流中学习并得到启发，记录自己学习、思考、总结的过程；分享自己的故事、所得、感想、经验。

需求分析

系统功能需求

改系统分为两个角色，一个是访客，一个是博主

访客拥有的权限：

页面板块设计

首页

最新文章 热门文章 热议文章 标签云 文集云 博主个人简历

系统功能需求

该系统分为两个角色，一个是访客，另外一个是博主。其不同角色对应有不同的功能。

博主功能介绍：

（1）登录

这部分功能又分为博主登录、博主退出两个部分。

登录：主要用于验证博客网站用户信息的真实身份，以便对博客网站进行管理和维护。网站检测用户的用户名，密码判断其是不是博客的博主，并给予其相应的权限对博客网站进行操作。

退出：博主通过退出，可以释放自己所占有的各种信息资源。

（2）文章发布

博主可以发表自己的文章，文章包括标题、正文、表情、图片、标签、自定义URL等信息，博主通过各种元素来展示自己的想法和思想。系统接受这些信息并且存储在服务器端的数据库中。

（3）文章管理

文章的删除：博主可以删除自己已经发表的文章内容和各项信息，系统自动在服务器端数据库中删除这些记录。

文章的修改：博主可以修改自己已经发表的文章内容和各项信息，系统自动在服务器端数据库中更新这些信息。

（）评论管理

评论的回复：博主回复访客的评论。系统自动将这些评论存储在服务器端的数据库中，并且可供博主以及其它读者浏览。

评论的删除：博主可以删除访客的评论，系统自动在服务器端数据库中删除这些数据。

（3）文集管理

由于当代是一个信息膨胀的社会，需要博主拥有编辑文集的权限，通过对自己所写文章进行归类，让读者可以更快速的找到自己所想要看到文章，所想要了解的信息。

可以新建文集，或把文章归类到已有的文集下面。归类好的文集可以进行上下移动跟删除。

访客功能介绍：

访客可以根据文集，标签，最新文章，热门文章，热议文章来快速定位到自己想要看的文章以及需要了解的知识。同时可以通过文集，查看博主对文章的分类，得到对一类知识的大体了解。对所浏览的文章进行评论，实现与博主跟其他人的交流跟沟通。同时可以用手机终端对文章进行浏览。

个人简历模块功能

个人简历是对博主的介绍。分为四大板块

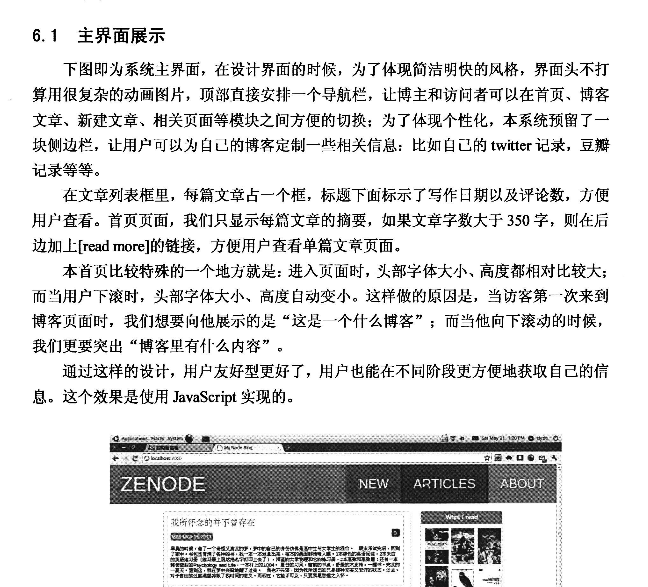
1. 关于板块 介绍博主的一些基本信息。比如目前就职公司，爱好及职业
2. 作品展示板块 对博主已完成的作品进行展示
3. 能力板块 对自己现在已经会的技术，以及熟练程度进行介绍
4. 联系板块 显示自己的联系方式

根据系统功能需求对主要页面进行布局跟设计

总体设计

对每个页面增加进度条。让读者知道页面的加载进度。

首页设计



头部

Banner图片轮播

标签展示

三个tab页切换

文集展示

尾部

当进入首页。为增加页面的动感效果。Banner及tab页采用从左侧切入，标签展示及文集展示从右侧切入的效果。

Banner图片轮播，主要是博主对一些自己语录的记录。可以自动切换，也可以手动切换

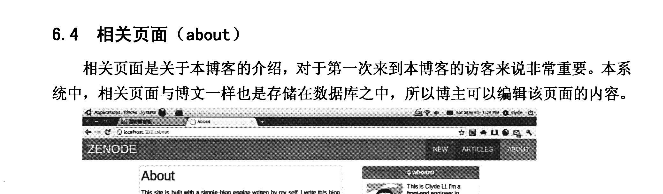
三个tab页切换，分别是对最新文章，热门文章，热议文章进行切换，tab页对列出对于的文章列表。当鼠标悬浮在文章上面时，该文章区域浮现阅读全文，标签，文集的相关信息。

左侧标签展示，文集展示采用标签云跟文集云的效果，当鼠标不在改区域的时候，标签跟文集会自由滚动。当鼠标停在特定的文集跟标签上时，该标签或文集停止滚动。点击特定的文集跟标签，跳到指定的标签跟文集汇总页面。

文集汇总页面设计

以列表的形式显示文集名及文集下面对应的文章汇总，同时当鼠标悬浮在文章上面时，该文章区域浮现阅读全文，标签，文集的相关信息的效果

博主个人简历设计

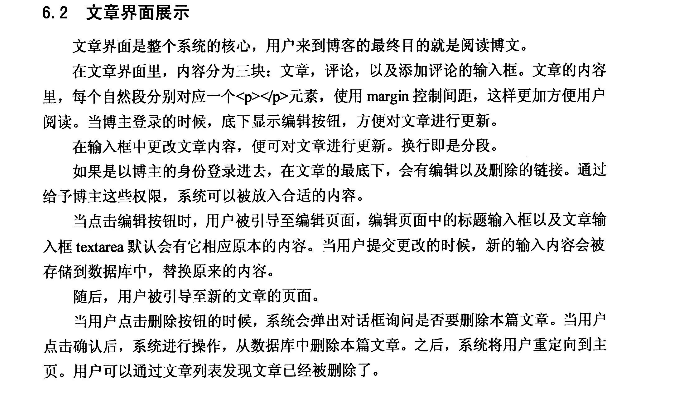


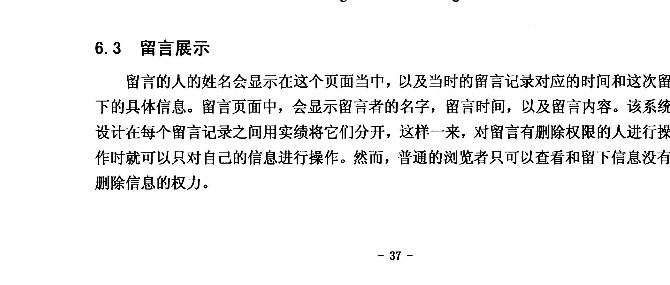
整个页面由四个模块构成。全面的介绍了博主的信息。同时通过头部的关于，能力，作品，联系可以定位到特定的模块。页面的字体显示均采用动态字体显示效果。

标签详细页和文集详细页展示

显示指定标签或文集下的文章。

文章详细页





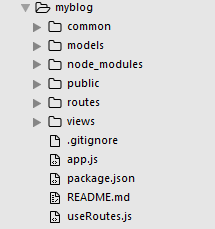
阅读文章的内容。显示该文章的浏览量，评论量，所属文集及标签。同时显示评论区，了解其他读者对这篇文章的理解，与其他读者讨论。

后台管理页

发布文章，对文章进行归类，可以对评论进行回复跟删除，对自己的发表的文章进行编辑跟删除。

具体的实现

目录结构



1. public 项目公共目录，其中又包含了images、javascripts、stylesheets目录，也就是说要求我们把网站所有的静态资源、公共资源放到此目录下，因为只有public目录下的文件可以通过URL直接访问  
     
   2、routes 路由控制器目录

3、views 视图/模版目录，就是MVC的view层，采用 jade模版引擎

4、app.js 项目入口文件，程序就是从这里开始运行  
  
5、package.json 项目包文件定义，其中最重要的部分就是依赖关系的定义，可以让我们非常方便的使用NPM进行项目的发布安装。方便项目的移植

node\_modules：用来存放项目的依赖库。

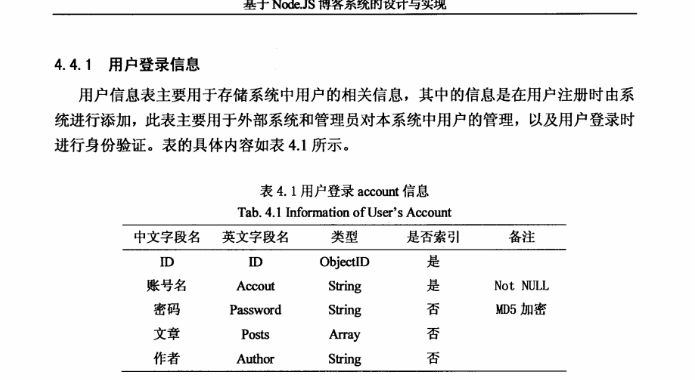
common 存放整个项目公共的配置文件，工具类

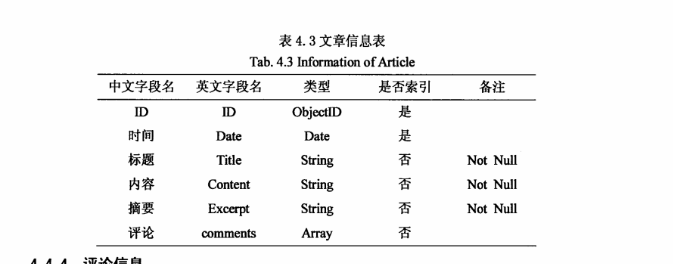
model 存放数据库模型

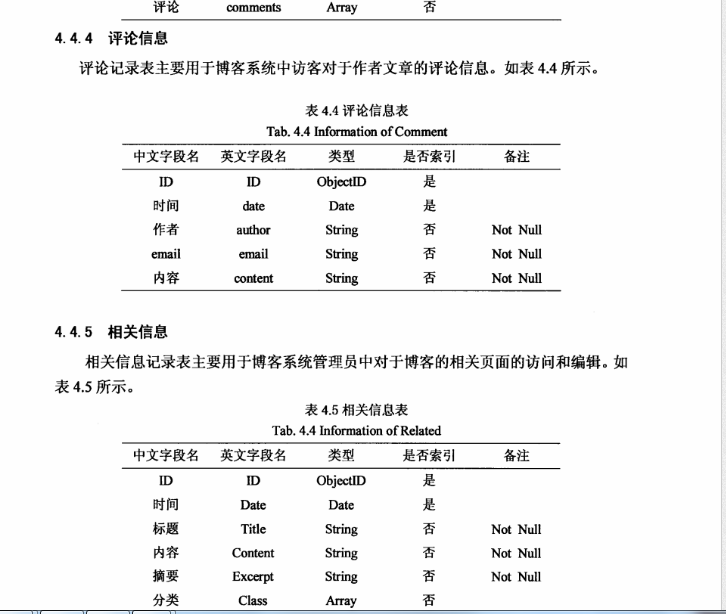
.gitignore 由于这个项目采用的版本管理工具为git，这个配置文件用于配置不需要加入版本管理的文件,配置好该文件可以为我们的版本管理带来很大的便利

README.md 来对整个项目进行介绍的一个文件

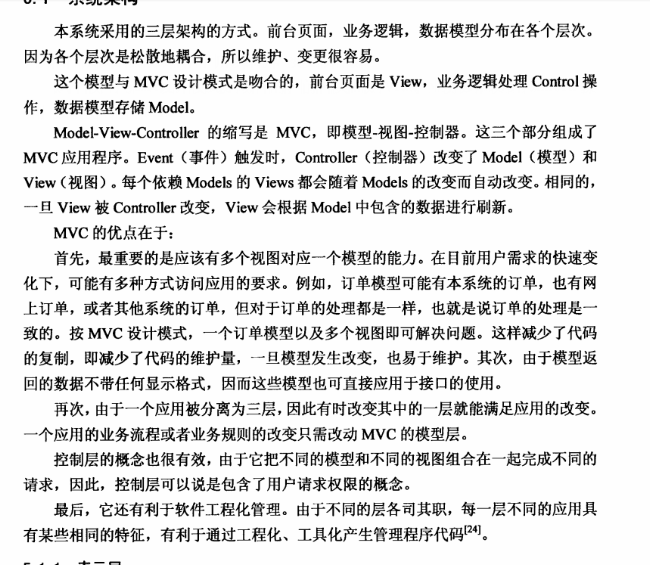
userRoutes.js 将路由控制全部挂载在express应用上







系统架构介绍



MVC

表示层（view层）

表示层为用户在浏览器中看到的页面。这里采用jade作为模板引擎。通过jade的include和继承实现代码复用，使代码的冗余信息缩减到最小。采用res.render的方法，传入数据，对表示层进行渲染，从而最后传给浏览器，显示给用户。

实现会话持久

var express = require('express'); //加载express框架

var session = require('express-session'); //加载express-session框架

var app = express(); //实例化一个express对象

app.use(session({

secret: 'yy', //用来对session数据进行加密的字符串.这个属性值为必须指定的属性。

resave: false, //是指每次请求都重新设置session cookie，假设你的cookie是6000毫秒过期，每次请求都会再设置6000毫秒。

saveUninitialized: true //是指无论有没有session cookie，每次请求都设置个session cookie ，默认给个标示为 connect.sid

}));

Model层（数据模型层）

数据库设计

博主

