

基于 Vue+Node+MongoDB 的博客网站设计与实现

王名培 黄媛媛

四川大学锦城学院 计算机与软件学院 四川 成都 611731

【摘要】: 现代社会随着手机电脑等各种科技设备的普及,人们之间除了线下聚会,还增加了线上交流方式。为了适应这一变化,各种社交类网站应运而生。本网站基于前端的渐进式框架 vue, MongoDB 和 Node 三个技术构建了一个生活分享网站。本文对该网站登陆注册、用户基本信息头像修改、文章管理、文章点赞评论、文章推送、私信回复以及博客主个人页面浏览等模块提出设计方案与实现,并阐述所使用的技术。

【关键词】: 前端; 网站开发; VUE; Node; MongoDB

1 引言

随着硬件技术,互联网以及浏览器的快速发展,当今人们对线上交流的需求提升,人们开始在网络上分享自己的生活,记录自己的生活,因此需要一个博客分享平台,创造展示自己的生活在他人,在网络中找到自己相同爱好的朋友。本网站在一定程度上满足了这个需求,让用户成为生活美的寻找者,提供了登陆注册,博客发表,点赞回复私信,博客浏览等功能,让用户有自己的个人空间去分享生活体验生活。本文通过介绍该网站的关键架构决策、可行性分析、需求分析、系统功能结构设计,来介绍此项目的设计思路和实现的过程。

2 关键技术选型介绍

2.1 Vue.js

Vue.js 是一套构建视图层页面的开源框架,它与其他两个前端主流框架相比,其拥有轻巧、性能快、单页组件化等特点。对于开发者而言,其采用的 MVVM 模式,让你只需要简单的代码即可实现数据绑定视图组合组件复用等开发,使得学习成本大量减少。同时其拥有丰富的第三方库,通过组件可以精简的实现重复的页面模块和动效的功能,让开发者更专注于开发功能而非复杂的数据绑定和重复的代码重写。

2.2 Node.js

Node.js 是一个基于 V8 引擎的 JS 的运行期系统,它使得 JavaScript 能在服务器端运行。他还使得 js 具有文件读写操作能力,弥补了浏览器不能直接操作本地文件的缺点,让 JavaScript 成为能跟主流的 PHP, Java, Python 等语言平起平坐^[1]。让前端程序员不需要学习新的语言既可以创建一个服务器进行测试和接口开发。Node.js 对不同于一般的用例采取一定手段进行改变,提供替代的 API 接口,使其拥有响应速度快、易于扩展的网络应用,其拥有 express 等第三方模块的支持,让开发者轻松实现一个后台的创建以及实时通信等

可扩展的网络应用程序。

2.3 MongoDB

MongoDB 是一个由 C++编写,基于大量 PC 服务器通过网络连接对外提供的一个整体文件存储服务系统的数据库。它是一个介于采取关系模式的数据库和采取非关系模式存储的数据库间的一个数据库。它可以存储大部分类型的数据。而在 Node 中, MongoDB 为 Node.js 提供了便利的第三方库 Mongoose,使得在 Node.js 中只需要 npm 包安装以后进行引用,即可快速实现对 MongoDB 数据库的增删改查功能。

2.4 Ajax

Ajax 是一种通过与服务器进行数据交换,只改变一部分页面的异步 JavaScript 和 XML 技术。它对比于传统网页,使得互联网的应用程序小而快,让用户体验更加友好。

3 可行性分析

3.1 技术可行性

Vue.js 是目前主流框架中十分轻量,性能强大^[2],数据绑定简单,组件开发容易,其拥有中文开发文档说明清晰,方便学习。他扬长避短,通过脚手架和 npm 包的支持,有成熟的 ui 库,能轻松的搭建一个美观,功能性强,交互效果不错的网站,同时其强大的单页组件特性,使得开发者协作容易,维护成本大幅度降低。同时其支持 axios 并采用 MVVM 模式,使得 ajax 请求的数据并回显这个操作更加方便。

Node.js 是基于 ChromeV8 引擎的 JavaScript 运行环境,它使得 JavaScript 能在服务端运行,让前端开发者不需要学习新的编程语言即可创建中小型服务器。本项目采用 express 这个小规模的 web 应用程序开发框架,让前后端进行分离。因此后端不做页面模板的渲染和路由的导航工作,只需要提供 api 接口给前端进行 ajax 调用然后通过 vue 自身来处理页面数据渲染和前端页面导航,满足于本项目对于基本数据获

取操作的接口需求。

MongoDB 为 Node 提供了 mongoose 这个第三方库，能使用 json 等格式存储，同时拥有成熟的可视化工具，只需要调用简单的 api 即可对 MongoDB 数据库中数据进行操作，国内网络拥有大量的教程，方便学习，处理数据速度快，适用于本网站对数据库部分需求。

综上，技术可行。

3.2 运行环境可行性

本次项目开发中，Vue,MongooseDB,Node 可以在 window、linux 和 mac 系统都可以进行开发，同时无论在 vscode 的命令行还是系统自身命令行都可以进行代码的启动和调试，而如果将代码更新到 git 上，若更换电脑安装完 node, mongoose 和 vue 环境后，进行一条 npm 命令安装相关库即可快速的运行。因此，运行环境可行。

4 需求分析

对于用户来说，使用本网站，可以拥有一个个性化的博客，可以充分展示自己的爱好，了解更多的人，看到不一样的世界，只需要手指轻轻一点即可上传图片向更多的人记录分享自己的生活。对于该网站我们主要采用 PC 端网页开发，用简单明了的页面，满足用户上手快速的需求下同时满足当下网页页面设计审美。参考如今比较热门的同类网站，需求主要有用户注册，用户登录，浏览博客，用户基本信息设置，头像上传，文章发表，文章点赞，浏览个人页面，删除文章，评论文章，私信等。

5 模块设计

5.1 功能模块

本网站的主要分有用户注册，用户登录，浏览博客，用户基本信息设置，头像上传，文章发表，文章点赞，浏览个人页面，删除文章，评论文章，私信几大模块进行设计，如图 1 所示：

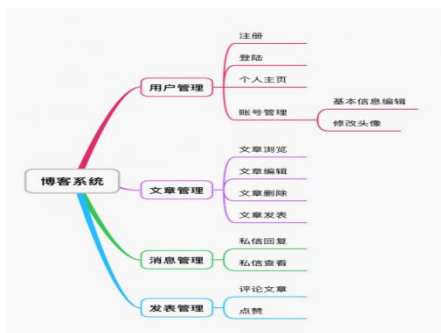


图 1 功能模块图

用户模块提供了用户的注册登陆个人信息查看账号基本信息管理等功能，为该博客系统提供一个区分游客和登陆用户的接口，用于区分某些功能的使用权限。文章管理用于提供文章基本的增删改查等功能，也是该网站的主要模块。消息管理和发表管理提供了私信评论点赞等功能，给用户提供了一个相互交流的空间。

5.2 功能描述

5.2.1 用户注册/登录/退出登录

- (1) 当用户没有账号的时候需注册账号才能进行登录操作。
- (2) 注册页面和登陆页面有相互切换的按钮，进入注册页面输入基本信息后发送请求。
- (3) ajax 请求到服务器服务器判断当前邮箱是否被注册过，然后存储到数据库返回一个状态码。
- (4) 前端收到注册成功状态码自动转到登录页面输入之前注册的邮箱和密码即可登录。
- (5) 登陆状态时之前登陆按钮转变为退出按钮。
- (6) 点击退出登陆从登录状态变为非登录状态。

5.2.2 文章管理

- (1) 该功能只面向已登陆用户。
- (2) 用户点击导航栏“发表”进行文章发表页面，输入文章标题、内容上传需要上传的图片，点击发表按钮，后台接收到文章信息进行存储返回发表成功状态码，然后跳转到首页，即可找到刚刚发表的文章。
- (3) 用户进入自己的个人页面，可以浏览自己发过的文章，同时能对其进行删除操作。
- (4) 用户可以对用户文章进行点赞评论，点赞评论主要代码如下：

a)前端发送请求：

```

submit:function(){
    this.$http.get("/api/mine").then(res=>{
        this.user_id=res.data.data._id;
        this.userid=res.data.data.userid;
        let datas={
            author:this.user_id,
            userid:this.userid,
            content:this.NewComment,

```

```
essayid:this.valueData._id,
  contenttime:new Date()
};
this.$http.post("/api/commits",item).then(res1=>{
  this.commentvalue.push(res1.item.data[0]);
});
});
},
b):后端处理请求:
router.post('/',function(req,res,next){
  data=req.body;
  var newCommits=new Comment(data);
  newCommits.save((errs,data)=>{
    let [resultCode,msg]=[0,"评论失败"]
    if(!errs){
      Comment.find({_id:data._id})
        .populate({path:'author',select:'_id username introduction'})
        .exec().then(user=>{
          resultCode=1;
          msg="评论成功"
          res.json({resultCode:resultCode,msg:msg, data:user})
        })
      }else{
        res.json({resultCode:resultCode,msg:msg, data:data})
      }
    })
  })
})
```

5.2.3 消息管理

(1) 进入其他用户个人主页，上方有一个“私信”按钮，在非游客状态下点击进入界面。

(2) 在私信页面输入内容，该消息会发送给你要发的用户。

(3) 收到私信的用户可以在自己的私信页面浏览别人

给你发送的私信。

(4) 点击回复可以回复发送给你私信的用户。

5.2.4 用户管理

(1) 用户点进设置页面可以对自己的个人信息进行基本设置。

(2) 点击修改头像页面可以对自己的用户头像进行修改。

(3) 在修改头像页面点击修改头像框内任意位置，弹出文件选择页面。

(4) 选择一张图片点提交，前端会发送 ajax 请求让图片上传到服务器，服务器保存成功会将图片路径回传到之前修改头像框位置进行回显。

(5) 前端收到服务器端上传成功的响应并弹出 Toast 模态框显示上传成功。

5.3 部分页面展示

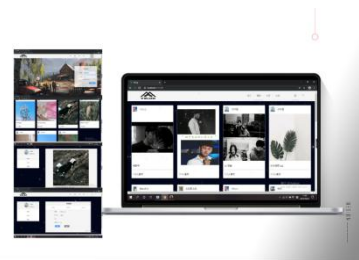


图2 5.3 部分页面展示

结语

目前国内网络市场中此类型网站数量较多，但其中一些网站操作不够便捷，且页面繁复，本网站以操作简单，页面清爽，功能完善的目标进行设计和开发。

本文给出了基于 Vue，Node 和 MongoDB 三大主流技术构建的博客系统。目前本网站依然存在需要不断完善的地方，同时还拥有一部分需要添加的功能，将在后续过程中不断优化，不断提升自己的产品竞争力。同时本网站未来还会响应当今主流的 WebApp 技术，开发移动端产品，致力于在各个设备之间都可以为用户带来良好的博客体验。

参考文献:

- [1] 张钊源,刘晓瑜,鞠玉霞.Node.js 后端技术初探[J].中小企业管理与科技(上旬刊),2020(08):193-194.
- [2] 唐斌斌,叶奕.Vue.js 在前端开发应用中的性能影响研究[J].电子制作,2020(10):49-50+59.