

本科毕业论文

JavaScript游戏分享平台的设计与实现

|  |  |
| --- | --- |
| **二 级 学 院** | **数学与计算机学院** |
| **专 业** | **软件工程** |
| **年 级** | **2014级** |
| **学 号** | **1410819041** |
| **学 生 姓 名** | **韦文耐** |
| **指 导 教 师** | **任子亭** |
| **职 称** | **讲师** |
| **完 成 时 间** | **2018 年03 月28 日** |

**独 创 性 声 明**

本人郑重声明：所呈交的毕业论文（设计）是本人在指导老师指导下取得的研究成果。除了文中特别加以注释和致谢的地方外，论文（设计）中不包含其他人已经发表或撰写的研究成果。与本研究成果相关的所有人所做出的任何贡献均已在论文（设计）中作了明确的说明并表示了谢意。

签名：

　　　　　　　　　　　　　　　　　　年　　月　　日

**授权声明**

本人完全理解贺州学院有关保留、使用本科生毕业论文（设计）的规定，即：学院有权保留并向国家有关部门或机构送交毕业论文（设计）的复印件和磁盘，允许毕业论文（设计）被查阅和借阅。本人授权贺州学院可以将毕业论文（设计）的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编论文（设计）。

本人论文（设计）中有原创性数据需要保密的部分为：无。

签 名：

　　　　　年　　月　　日

指导教师签名：

年　　月　　日

**目 录**

[摘 要 1](#_Toc11597)

[绪 论 2](#_Toc5346)

[1 系统需求分析 3](#_Toc11336)

[1.1 研究目标 3](#_Toc23762)

[1.2 可行性分析 3](#_Toc881)

[1.3 需求分析 3](#_Toc20882)

[1.4 功能分析 6](#_Toc8483)

[2 开发环境搭建 8](#_Toc17608)

[2.1 Node环境的安装 8](#_Toc8764)

[2.2 Visual Studio Code安装 8](#_Toc29283)

[2.3 编码环境搭建 9](#_Toc17242)

[3 系统设计 10](#_Toc23398)

[3.1 总体结构 10](#_Toc17626)

[3.2 功能设计 10](#_Toc15456)

[3.3 用户界面设计 12](#_Toc8220)

[3.4 数据库设计 16](#_Toc9261)

[3.5 文件结构说明 19](#_Toc8087)

[4 系统实现 22](#_Toc5687)

[4.1 运行环境 22](#_Toc14516)

[4.2 开发工具概述 22](#_Toc15187)

[4.3 功能实现 22](#_Toc15776)

[5 总结 24](#_Toc32490)

[5.1 主要工作和心得 24](#_Toc28263)

[5.2 遇到的问题和存在的不足 24](#_Toc8725)

[5.3 展望 25](#_Toc27662)

[参考文献 26](#_Toc23752)

[Abstract 1](#_Toc26011)

[致 谢 2](#_Toc9550)

游戏分享系统的研究与实现

作 者 韦文耐

指导教师 任子亭

摘 要**:**目前互联网已深深地影响着我们的生活，使我们的生活方式变得更加便利快捷。有那么一些人，在使用互联网的同时，深深地为互联网所着迷，想要探索互联网中的奥妙。为了增加学习探索的趣味性，部分人选择在游戏中学习，此时，一个可以供编程爱好者使用的游戏分享系统应势而生，为很多编程游戏爱好者提供一个可交流学习的平台。

本文主要介绍该游戏分享平台的设计与实现。本平台主要是针对使用JavaScript编写小游戏的编程爱好者进行分享使用。本文主要介绍该游戏平台的登录、注册、首页展示、留言管理、评论管理、我的好友及提交作品和查看作品的功能。该平台主要是实现前后端分离的思想，前台页面使用HTML+CSS，后台使用NodeJS实现服务器端技术，用MySQL作为后台数据库，结合VUe.JS框架与ElementUI组件库，实现基于MVVM框架的游戏分享平台。

本平台需要设置相应的权限：在管理中心、好友中心及发布作品页面需要登录才能访问，在首页和查看作品页面则不需要设置相应的权限。

关键词:小游戏；分享系统；NodeJS；JavaScript；

绪 论

（1）研究内容

文章研讨的内容是通过运用JavaScript编程语言，基于NodeJS操作平台实现一个可分享、可交流的游戏分享系统。

1. 该毕业设计的开发背景

目前的社会是“互联网+”的社会，科学技术一直在发展，人们的生活水平提高，人们都在跟进时代的潮流，努力接受新事物。在这个过程中，部分人选择学习互联网，即编程。在游戏中学习是很多人学习编程的一种重要方式，这种方式给枯燥的学习增加了趣味性，大大提高了初学者的效率。特别是近年来在奥巴马“全民写代码”的呼吁下，学习编程的风头更盛，国内外网友都纷纷开始学习编程。与此同时，在上千种编程语言中，JavaScript比较简单易学，该系统为编程爱好者提供了一个有趣味性、共同学习，互相进步的平台。

（3）研究目的和意义

本文研究的目的就是利用大众跟进潮流、学习新事物的心态，开发出一款让编程游戏爱好者相互交流，相互沟通，相互学习的分享系统。

通过查阅资料，研究并开发出该游戏分享系统。该系统以发布作品与查看分享的作品的源码为主线，留言、评论、个人作品中心的查看与操作为副线，让用户得以分享学习。

1. 根据调查得知，在国内尚未有类似专门用于游戏分享的系统或网站出现，而在不久的将来，学习游戏编程的人会越来越多，怎样才能满足此类人群的需求呢？此时，研究并开发一个游戏分享网站的意义更加重大。

（5）系统的特色和创新点

该系统的特点在于选题的新颖，能跟紧时代的脚步，努力做到为用户提供便利，服务大众，不只为毕业而作。创新点在于使用的技术较为前沿，前端部分使用的是近年来三大框架之一的Vue，后台部分使用的是把JavaScript推向新一股浪潮的NodeJS。

1. 系统需求分析

1.1 研究目标

本系统研究的目标有以下几个方面：

（1）理解和掌握NodeJS实现异步非阻塞程序的设计和开发过程；

（2）利用各个组件开发系统页面及功能。

（3）本系统是采用JavaScript语言进行编写的，是基于Visual Studio Code开发平台的一个游戏分享系统。

1.2 可行性分析

1.2.1 社会可行性

伴随着计算机科学技术和移动互联网技术的革新，互联网的用户也越来越多，人们对计算机的操作越来越纯熟，逐渐开始学习游戏编程，此时如果有一个游戏分享系统将更加利于人们学习及使用。软件设计是依法进行的，没有受到法律以及政策的约束。

1.2.2 技术可行性

该系统主要实现前后端相分离的理念。前端主要使用Vue.JS框架进行项目的架构搭建，并配合ElementUI组件库编写页面样式；后端主要使用NodeJS进行项目搭建，使用NodeJS的插件Express进行路由控制，通过对MySQL数据库数据的增删改查以处理前端提交的数据；前后端通过Vue.JS插件axios访问接口发送AJAX请求进行连接，提交或请求数据以实现对数据的增删改查操作。

1.3 需求分析

1.3.1 需求概述

打开系统进入首页，可以阅览首页的最新作品、推荐作品、推荐作者信息，这是未登录用户可进行的最基本操作。该系统需求根据登录用户与未登录用户进行区分。

未登录用户需求如下：浏览首页信息，查看作品详情，查看其他用户作品列表。未登录用户用例图如图1-1所示。

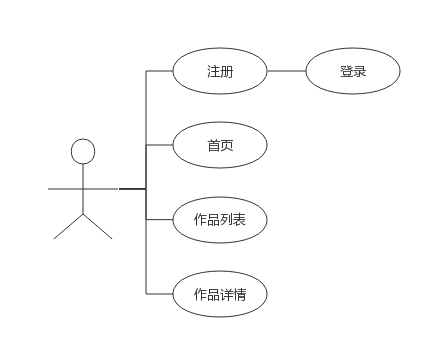


图1-1 未登录用户用例图

登录用户主要需求如下：

第一，发布作品。可通过点击菜单栏“发布作品”项进入发布作品页面，填写信息点击发布时对数据进行验证，验证通过后提交后台，后台进行保存入数据库。

第二，作品管理。主要查看用户个人的作品列表，用户可查看其他用户的作品列表，当前用户可实现作品的删除操作。

第三，管理中心。通过点击菜单栏“管理中心”进入管理中心页面，在该页面可实现留言管理与评论管理操作。其中留言管理可对留言信息进行回复操作，点击“回复”则弹出文本框进行输入内容，点击提交把内容提交后台存入数据库。评论管理则显示相应的评论信息，点击信息进行跳转到相应的作品详情评论部分，在该部分点击“回复”，弹出文本框进行输入内容，点击提交把内容提交后台存入数据库。

第四，好友中心。通过点击菜单栏“好友中心”进入好友中心页面，该页面显示好友信息，可点击“关注他”则实现对该好友的关注操作，对应信息存入数据库，此时按钮修改为“取消关注”；点击“取消关注”则实现对该好友的取消关注操作，对应信息存入数据库，此时按钮修改为“关注他”。

第五，修改密码与退出。登录后右上角才显示对应的用户信息，显示修改密码与退出按钮，点击修改密码跳转到修改密码页，输入当前密码和新密码后点击提交，后台对数据进行验证并返回相应数据。点击退出则清除登录信息，并刷新页面。

登录用户用例图如图1-2所示。

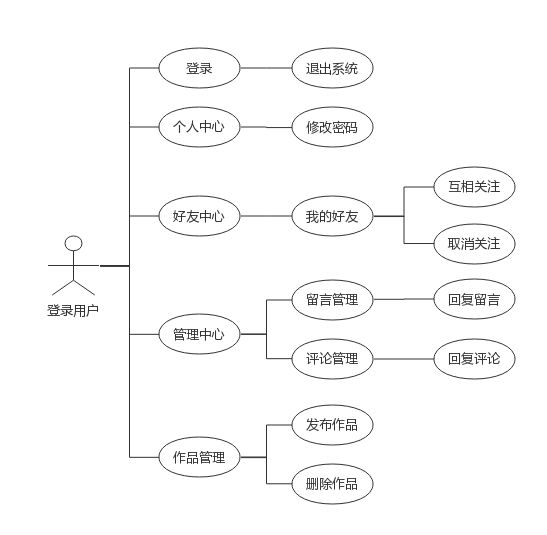


图1-2 登录用户用例图

1.3.2 功能分析

该系统主要有登录注册、发布删除作品、评论留言管理、粉丝关注管理的功能，一下进行分段讲解。

登录功能有相应的权限，例如浏览首页和查看作品详情时并不需要登录，但在功能中需要登录权限，例如发布删除作品和管理留言评论与好友关注取消操作都要求登录。

发布与删除作品需要登录权限。发布作品从前台获取数据提交数据到后台，后台进行存储到数据库，并返回数据到前台提示发布成功。删除作品从个人作品列表进行删除，并把当前数据的ID返回后台，后台从数据库进行删除并返回成功提示给前台。

评论管理分有给我的评论与我给别人的评论。在给我的评论中有回复功能，把回复内容提交后台，后台返回数据进行刷新页面，并把回复的内容渲染在页面上；我给别人的评论仅作查看功能。留言管理同我的评论，具有回复功能。

粉丝管理主要可以跳转查看粉丝主页和关注他。关注操作判点击并把数据提交后台，后台判断点击则更改数据库数据，重新返回更改的数据。关注管理同粉丝管理，有取消关注与查看他的主页的功能。

1.4 功能分析

该系统通过Vue.JS组件进行组件化开发，严格遵照软件工程的相关标准，采用软件设计技术来降低模块之间耦合，主要功能模块有：

（1）发布作品，对作品的发布。

（2）管理中心，主要是对留言与评论的展示与回复操作。

（3）好友中心，对好友的关注与取消关注操作

（4）作品详情页，展示作品详情与回复评论。

（5）作品管理，展示作品列表与删除作品。

（6）个人中心，能够进行登录注册，修改密码，退出操作。

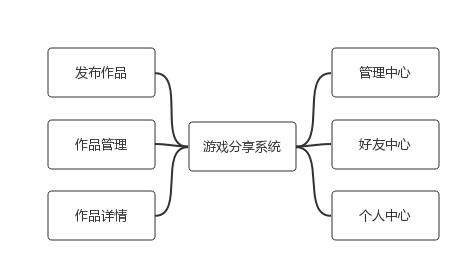


图1-1 功能模块图

详细流程图如图1-2所示。

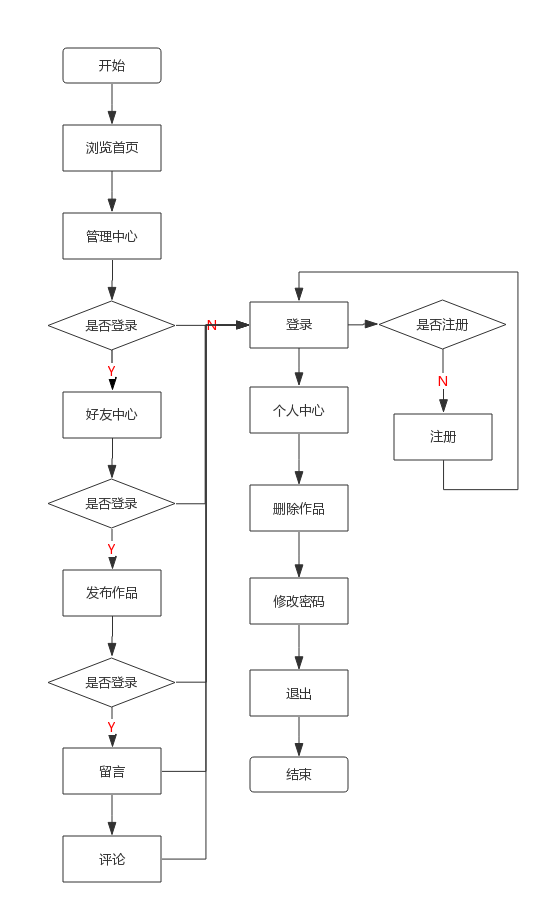


图1-2 详细流程图

1. 开发环境搭建

2.1 Node环境的安装

首先进入官网下载安装包，按照步骤进行安装，安装完成点击“finish”，如图2-1所示。

图2-1 node安装完成图

在命令行输入“node -v”检测版本号，不报错，检测成功则表示node安装成功，安装成功会自动生成npm包管理工具。如图2-2所示。

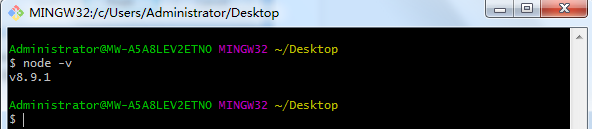


图2-2 node完成完成示意图

2.2 Visual Studio Code安装

官网下载安装包完成后，设置好安装路径后点击安装即可安装。安装完成后，点击“完成”即可。

安装完成新建或打开一个文件，如图2-3所示。

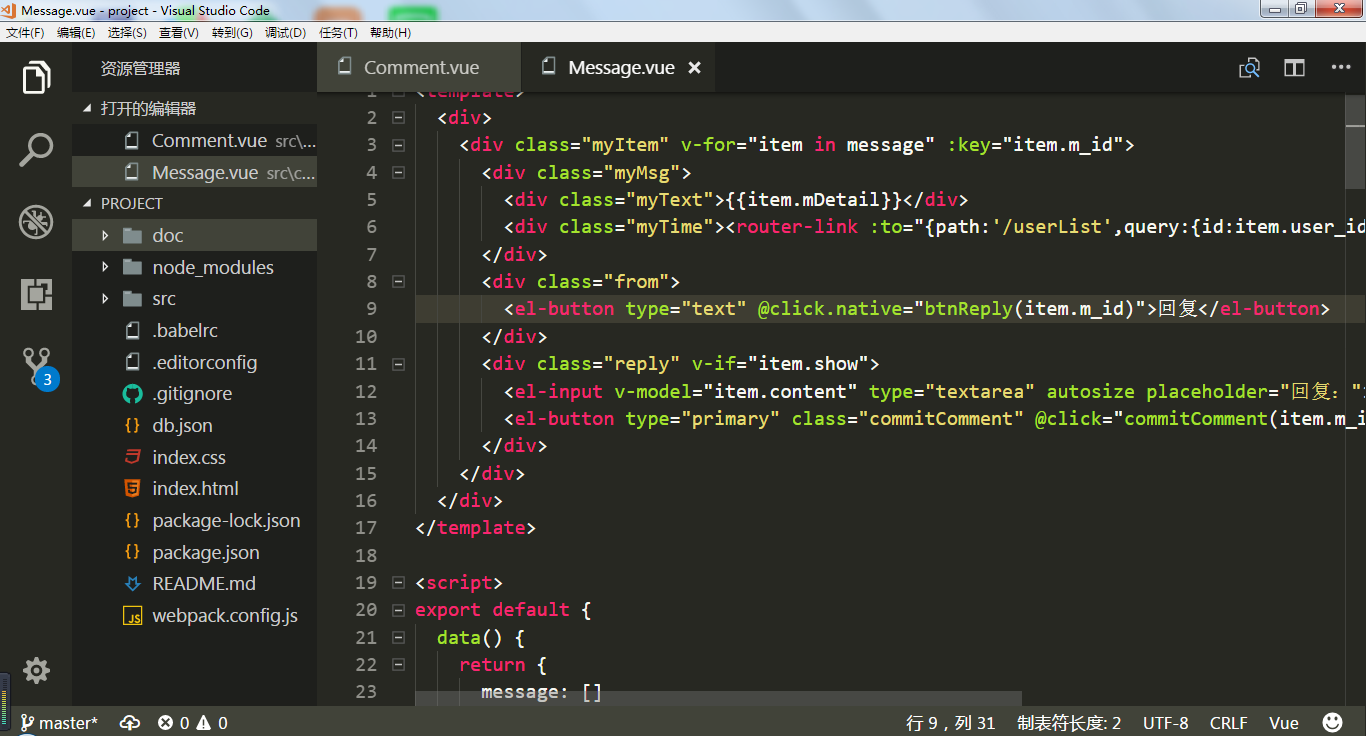


图2-3 Visual Studio Code打开文件夹

2.3 编码环境搭建

在命令行中输入“npm install vue vue-cli -g”进行全局安装Vue环境，成功后在选择一个目录执行“vue init webpack project”搭建前端项目结构，生成项目结构在project下。

3 系统设计

系统设计在整个系统的开发过程中起着非常重要的作用。系统分析是决定系统应该“干什么”的问题，而设计阶段则是解决系统应该“怎么干”的问题。它从系统的目标启程，建立系统的模型，确定系统的总体结构，规划系统的规模，建立各个基础部分，并阐明它在整个系统中的作用及彼此之间的关系。

在本文中，系统设计阶段主要任务包括：数据库设计、系统结构设计、系统流程图设计、系统功能模块结构设计等。

3.1 总体结构

系统的总体结构主要由前端Vue.JS框架及后台NodeJS组成，通过后台Express框架创建接口，前台发送ajax请求获取数据。Express 是一个基于 Node.js 平台的简单、灵活的 web 应用开发框架，它提供很多非常强大的特性。具备丰富的 HTTP 快捷方式和任意排列组合的 Connect 中间件，让创建健壮、友好的 API 变得既快速又简单。Express 不对 Node.js 已有的特性进行二次抽象。Node.js 事实上是一个基于 Chrome V8 引擎的 JavaScript 运行环境，使用了一个事件驱动、非阻塞式 I/O 的模型，让它变得轻量又高效。应用npm作为包管理器 ，是全球最大的开源库生态系统。Vue.JS是用于构建用户界面的一套渐进式框架，与其它框架不同的是，Vue 是一个自底向上逐层应用。Vue 的核心库主要关注视图层，不光易于上手，而且便于与既有项目整合，还能够为复杂的单页应用提供驱动。

3.2 功能设计

根据系统功能，可对管理中心、好友中心、作品管理、个人中心功能进行详细描述。

3.2.1 管理中心模块设计

管理中心主要是对留言与评论的管理。留言相当于私信，有查看留言与回复留言的功能；评论同时具有查看和回复功能，不同的是评论针对某篇文章，在其他用户浏览文章时可以被查看，而留言只能被当前的用户查看，表示留言是留言对象双方的，评论是可多人的查看的。查看评论时可点击评论信息进行跳转到评论的文章页面再进行回复。管理中心模块功能模块图如图3-1所示。

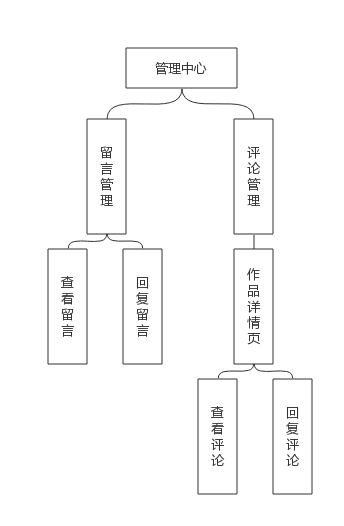


图3-1 管理中心模块图

3.2.2 好友中心模块设计

好友中心主要是对好友的管理，可以进入他的主页进行查看他的作品，也可以执行关注他与取消关注操作。好友中心功能模块功能模块图如图3-2所示。

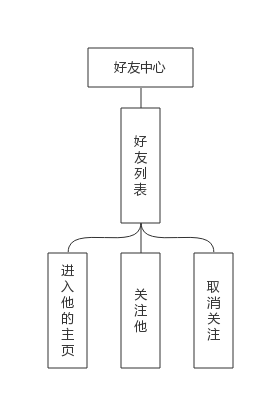


图3-2 好友中心功能模块图

3.2.3 个人中心模块设计

个人中心模块主要功能有登录、注册、修改密码、退出账号。个人中心模块功能模块图如图3-3所示。

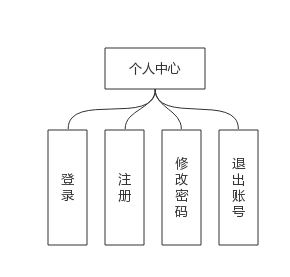


图3-3 个人中心模块功能模块图

3.2.4 作品管理模块设计

作品管理模块主要是对作品的增删查操作，主要功能有查看作品、发布作品、删除作品。其中查看作品不需要登录权限，而发布作品与删除作品需要登录权限，只能用户本人才能执行操作。作品管理模块功能模型如图3-4所示。

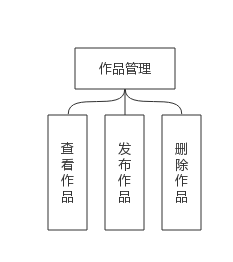


图3-4 作品管理功能模块图

3.3 用户界面设计

3.3.1 首页界面设计

首页模块主要实现展示功能，头部点击进行跳转到相应的界面；中间左部展示最新作品，点击可跳转到相应的详细作品信息；中部轮播图展示部分宣传的作品界面；推荐作品为获赞数较前的作品，点击可跳转到相应的详细作品信息；中部右侧为获赞数较高的作者的排行。以上信息都通过向后台发送Ajax请求获取，在后台中，最新作品通过发布的时间进行排序，推荐作品通过获赞数进行排序，作者排行通过作品的获赞总数进行排序。

系统在服务器上启动后，用户在浏览器地址栏中输入‘http://localhost:8080’后回车可进入首页，可浏览最新作品、推荐作品、优秀作者。首页主要内容如图3-5所示。

图3-5 主页主要内容

3.3.2 登录界面设计

用户在首页点击登录按钮，在注册页面点击已有账号或者在未登录的情况下访问管理中心、好友中心、发布作品都可跳转到登录页面。用户输入后发送Ajax请求判断账号密码是否正确。如果账号密码未填点击登录给出提示，账号密码不正确也给出相应的提示。登录页面如图3-6所示。



图3-6 登录页面

3.3.3 注册界面设计

注册模块主要是实现用户注册操作，在点击注册的瞬间执行以下操作：

1. 验证用户名与密码是否已填，
2. 验证密码与确认密码是否相同，
3. 向后台发送Ajax请求，
4. 后台把账号密码存储入数据库，
5. 返回请求是否成功信息，
6. 如果前台接收到注册成功的信息则跳转到登录页，如果不成功则给出提示。

注册模块界面设计如图3-7所示。



图3-7 注册界面设计

3.3.4 上传作品界面设计

上传作品代码方面使用wangeditor富文本编辑器，提交作品把编辑器里的源代码进行提交给后台，后台进行保存入数据库。上传作品界面设计如图3-8所示。

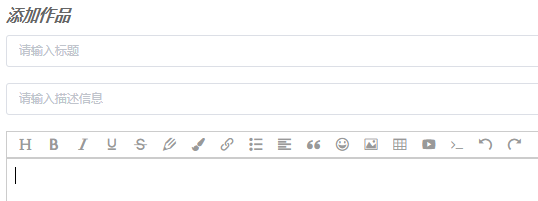


图3-8 上传作品界面设计

3.3.5 管理中心界面设计

管理中心左侧菜单使用ElementUI的菜单栏，子菜单有评论管理与留言管理，点击分别修改路径，在右侧显示相对应的内容，如果为评论管理则显示我的评论与给我的评论，如果为留言管理则显示我的留言与给我的留言。右侧为ElementUI的Tab栏组件，可实现相互却换操作。数据通过向后台发送Ajax请求获取，后台读取数据库并返回。管理中心模块界面设计如图3-9所示。



图3-9 管理中心界面设计

3.3.6 好友中心界面设计

好友中心模块主要是显示我的粉丝与我的关注，都是发送Ajax请求从后台获取数据，从数据库读取数据进行显示。在我的粉丝界面可以在右边点击进行关注他操作，我的关注界面可以进行取消关注操作。好友中心界面设计如图3-10所示。

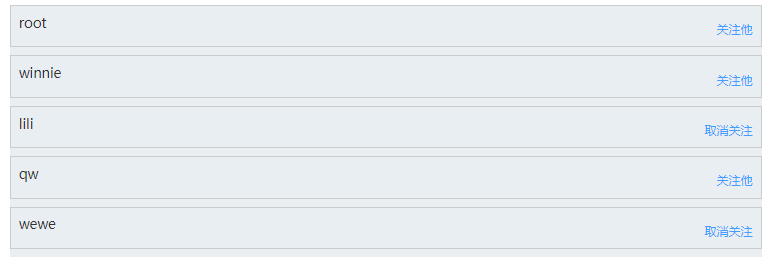


图3-10 好友中心模块设计

3.4 数据库设计

本系统采用MySQL数据库进行数据的获取和存储。其中主要有用户信息、作品信息、管理信息及好友信息。

用户数据包括用户的ID、账号、密码。用户的E-R图3-11所示。

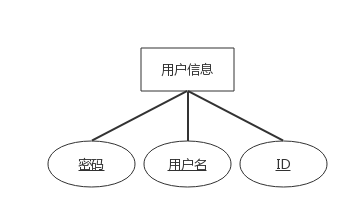


图3-11 用户E-R图

评论表包括表的ID、评论的用户ID、评论的文章ID、评论的内容、当前登陆用户ID及是否已经回复的标识。评论表的E-R图如图3-12所示。

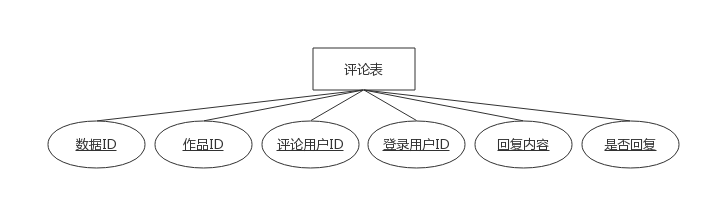


图3-12 评论E-R图

好友表包括列表ID、当前登陆用户ID、好友ID和是否关注的标识，已经关注则为1，未关注为0。好友表的E-R图如图3-13所示。

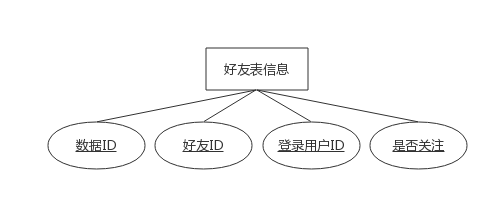


图3-13 好友E-R图

留言表包括留言ID、留言用户ID、当前用户ID、留言信息及是否回复的标识。留言表的E-R图如图3-14所示。

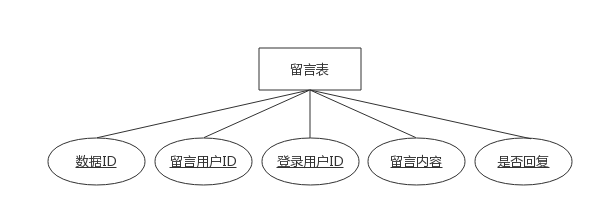


图3-14 留言E-R图

作品表包括作品ID、用户ID、作品名称、作品描述、点赞数、发布时间、作品编码及是否已经删除的标识。作品表E-R图如图3-15所示。

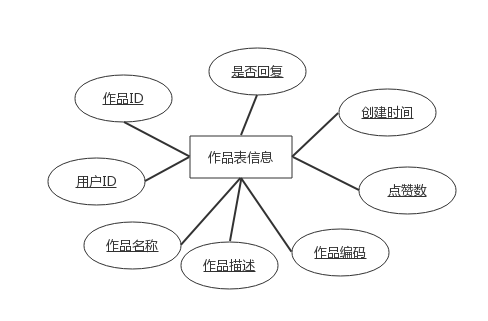


图3-15 作品表E-R图

数据库一共有五个表，表结构设计如下。

用户表主要用以存储用户的账号密码，用户表如表3-1所示。

表3-1 用户表结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否主键 | 描述 |
| user\_id | Int | 是 | 用户ID |
| userName | Varchar(10) |  | 用户名 |
| userPwd | Varchar(18) |  | 用户密码 |

作品表主要用以存储作品信息，如作品名称、作品描述、作品编码、点赞数及创建时间，作品表如表3-2所示。

表3-2 作品表结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否主键 | 描述 |
| w\_id | Int | 是 | 作品ID |
| user\_id | Int |  | 用户ID |
| workName | Varchar(30) |  | 作品名称 |
| workDetail | Varchar(100) |  | 作品描述文字 |
| count | Int |  | 获赞数 |
| createTime | dateTime |  | 作品创建时间 |
| workCode | longtext |  | 作品的代码 |
| isDel | bit |  | 是否删除 |

评论表用以存储用户的评论内容，评论表如表3-3所示。

表3-3 评论表结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否主键 | 描述 |
| c\_id | Int | 是 | 我的评论ID |
| w\_id | Int |  | 评论的作品ID |
| chDetail | Varvhar(100) |  | 评论内容 |
| user\_id | int |  | 评论用户ID |
| Login\_id | int |  | 登录用户ID |
| isShow | bit |  | 是否回复 |

留言表用以存储用户的留言内容，评论表如表3-4所示。

表3-4 留言表结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否主键 | 描述 |
| m\_id | Int | 是 | 留言ID |
| user\_id | Int |  | 留言用户的ID |
| mDetail | Varchar(100) |  | 留言内容 |
| Login\_id | int |  | 登录用户ID |
| isShow | bit |  | 是否回复 |

好友表用以存储用户的粉丝及关注的好友，评论表如表3-5所示。

表3-4 好友表结构

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名 | 数据类型 | 是否主键 | 描述 |
| follow\_id | Int | 是 | 关注列表ID |
| user\_id | Int |  | 关注对象ID |
| status | bit |  | 是否互相关注 |
| Login\_id | int |  | 登录用户ID |

3.5 文件结构说明

由于该系统使用前后端相分离的理念，因此代码分别写在不同的文件夹，真正的实现前后端分离因此以下文件夹将逐个进行说明。

3.5.1 前端文件结构说明

前段架构主要是使用Vue.JS的脚手架vue-cli生成，其中页面需要的插件配置项写在webpack.config.js文件内，所有的页面代码写在src文件夹下，以App.vue作为路由主页面，main.js作为入口文件。在component文件夹下编写各个子组件，在assets文件夹下存放使用的各类资源，如CSS文件，图片等。router文件夹编写路由控制，以路由控制页面的渲染。页面需要的依赖包存放在node\_modules文件夹下，在package.json文件中进行存储依赖项，需要的依赖项可根据package.json在命令行中执行“npm install”进行下载。前端文件结构如图3-16所示。

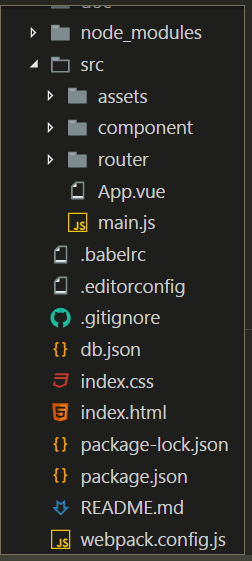
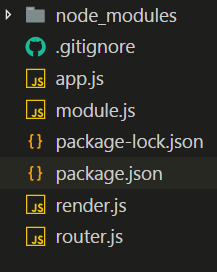


图3-16 前端文件结构

3.5.2 后端文件结构说明

后端文件与前端文件类似，都是通过npm安装依赖项或插件进行使用，所以都有package.json文件与node\_modules文件夹，依赖项保存在package.json文件内，依赖包则保存在node\_modules文件夹内。Node.JS作为后台，以本地形式创建服务器。App.js作为入口文件，定义服务器的端口供前台使用。Router.JS主要做路由控制，编写相应的端口响应相应的请求。Module.js文件编写SQL语句直接操作数据库数据，并把结果返回给render.js文件，在render.js文件进行获取前台返回数据进行处理。



4 系统实现

4.1 运行环境

服务器：node 8.9.1

操作系统：Windows 7

数据库：MySql 11.1.13

网络通信协议：TCP/IP

开发工具：Visual Studio Code 1.17.2

4.2 开发工具概述

该系统使用的开发工具是Visual Studio Code，这是一个可以运行于Windows之上的，针对编写Web应用的跨平台源代码编辑器，集成了大部分现代编辑器所应该具备的特性，包括语法高亮、可定制的热键绑定、括号匹配和代码片段收集是一款全特性的IDE。

开发语言使用JavaScript，JavaScript是一种直译式脚本语言，也是一种动态弱类型、基于prototype的语言，它的解释器被称为一种引擎，是浏览器的一部分。

前台页面开发使用Vue.js+ElementUI框架，Vue.js是构建用户界面的一套渐进式框架，自身采用自底向上、增量开发的设计，能够驱动一个复杂的单页应用，即使该应用是采用单文件组件和Vue生态系统支持的库开发的；ElementUI是基于Vue 2.0的一套桌面端组件库帮助快速搭建网站，使用这个组件库可以大大减少研发时间成本。

后台部分使用的是Node.js，Node.js是一个Javascript运行时环境，是一个基于Chrome运行建立的平台， 用于快速地搭建一个响应速度快、易于扩展的网络应用。Node.js 是使用事件驱动的， 从而使非阻塞I/O 模型得以轻量和高效并适合在分布式设备上运行密集型的数据。

4.3 功能实现

首先，在后台部分对数据库操作的SQL语句进行封装，查询语句得出的结果作为回调通过module.exports传递给render.js文件对应方法，在方法内对通过response获取前台传递的数据进行处理，根据render.js里的方法在router.js内编写路由，生成相对应的接口供前台调用。

在前台部分使用Vue的脚手架搭建的项目架构，并搭配使用ElementUI进行搭建样式结构。通过使用axios插件向后台发送ajax请求，以获取后台数据供页面展示，在功能型页面中，需要获取用户输入的数据，并提交给后台，后台进行验证并返回正确或错误信息，前台需要进行展示该信息。

在用户登录方面需要前后台配合对登录用户的验证，在获取后台数据库信息的同时进行返回是否已经登陆的验证，如果前台获取到的信息显示用户未登录而访问需要登录的页面，则给出未登录的提示并跳转到登录页面。

5 总结

5.1 主要工作和心得

经过几个月的不懈努力，游戏分享系统终于初步完成。本次毕业设计中，按照预期设想画出原型图后，进行设计静态页面，在页面完成之后设计数据库添加数据。然后进行后台controller层操作，读取数据库，分析数据进行增删查操作，把数据返回给前台页面进行渲染。在这个作品中，发布作品与查看作品是项目的核心功能，其实是对数据库的增、查操作，其他操作例如评论与留言功能是彰显项目的目的--方便学习交流。

在这过程中，不仅全方位学习了一个网站的制作过程，这更是一个提高自身技术的过程。在完成必要工作的同时也学习了新技术，例如对Node.JS与Vue.JS的熟练使用。更是对路由有了更深入的了解。经过反复的测试、修改，逐渐完成成现在的模样，虽然没有达到预期完美的效果，但也是经过努力做出来的，过程可谓艰辛，毕竟独自完成一个功能性网站不容易。

5.2 遇到的问题和存在的不足

因为开发的时间比较短及技术等方面条件的限制，与其他分享网站相比，该系统还是存在着很多不足之处的。主要有有以下几点不足：

（1）界面不够友好与美观。在系统的开发过程中即使使用的ElementUI组件库元素样式设计得都还不错，但是整体样式有所欠缺，例如整体色调方面不美观，有些杂乱，logo的设计不太美观等。

（2）代码的编制。对于开发时间比较紧凑的，文件及变量命名不是很规范，没有严格按照代码的编制要求完成命名，在代码风格上也有些杂乱，例如文件引入的方式及代码缩进等不够规范，维护起来不够方便。

（3）未接受过专门的测试，在验收过程中只是进行了简单的白盒测试，并未进行专门的测试，如黑盒测试，也未让用户测试等，离成为一个真正可用的网站还有一定的距离，还需要进行一个全面的测试。

（4）系统的功能比较简单。因为各种条件的限制，系统的功能完成得比较粗略，部分想要的功能虽然主要部分已经完成，但是一些其他部分尚未完成。例如预期的作品修改功能尚未完成。

（5）在项目开始的时候应该制定需求变更标准，都应该严格按照标准实施，能够在完成开发后减少因需求不同导致的问题，从而降低沟通成本。

5.3 展望

软件设计与实践，总结老师在课堂上和自己在课外时间所学的知识，做了简单总结。凡事都有两面性，有做得好的地方也有存在缺陷和不足地方。

学校所学习知识可以用在工作上面的可能不会很多，但我们学到的是一个学习的方法，因为无论在人生的任何阶段，人们总是需要不断学习。在软件开发这一工作中，需要充分的锻炼自己的思维和敏锐的洞察能力，不断提高自身的技术，培养良好的学习习惯及学习能力，因为软件开发更新的速度是非常快速的，要不断学习新的知识来武装自己。

在技术方面，该系统中使用的技术是JavaScript，后台使用NodeJS。近几年JavaScript的热门程度越来越高，在近一年中，NodeJS异步非阻塞方式把JavaScript推向一个新的浪潮。虽然JavaScript已经逐渐被大众接受使用，但也还有很多不足需要更改，希望能在之后不断完善。

参考文献

[1] 许会元,何利力.NodeJS的异步非阻塞I/O研究[J].工业控制计算机,2015(03).

[2] 朴灵.深入浅出Node.js[M].人民邮电出版社,2013.

[3] (美)泽卡斯(Zakas.JavaScript高级程序设计[M]. 人民邮电出版社,2012.

[4] 张桂元,贾燕枫,姜波.征服Ajax[M]. 人民邮电出版社,2006.

[5] 曾亮.嵌入式浏览器Javascript引擎的分析与优化[D].电子科技大学,2011.

[6] 周明明,董亚波,朱淼良.基于MVC的Web控制层的开发模型设计[J].计算机应用与软件,2005(12).

[7] 袁婷.浅谈Web前端开发[J].无线互联科技,2017(02).

[8] 霍福华.Web前端开发技术以及优化方向[J].晋城职业技术学院学报,2017(02).

[9] 朱二华.基于Vue.js的Web前端应用研究[J].黔南民族师范学院,2015.

[10] 程桂花.MVVM前后端数据交互中安全机制的研究与实现[D].浙江理工大学,2017.

[11] 冯杰.基于MVVM模式的体系结构一致性测试[D].中国海洋大学,2015.

[12] Xuyuan Xu,Lai-Man Po,Terence Chun-Ho Cheung,Kwok-Wai Cheung,Litong Feng,Chi-Wang Ting,Ka-Ho Ng.Adaptive depth truncation filter for MVC based compressed depth image[J].Signal Processing:Image Communication,2014(3).

Research and development of game sharing system

Abstract**:** At present, the Internet has deeply influenced our lives and made our lives more convenient and faster. Some of you are fascinated by the Internet while using the Internet, and want to explore the secrets of the Internet. In order to increase the interest of learning exploration, some people choose to learn in the game. At this time, a game sharing system that can be used by the programming enthusiasts should be born, providing a platform for the vast majority of programming game enthusiasts to learn.

This paper mainly introduces the design and implementation of the game sharing platform. This platform is mainly for JavaScript programming fans to share the use of small games. This article mainly introduces the login, registration, home page display, message management, comment management, my friends and the works and the functions of the work. The platform is mainly to realize the idea of front and back separation, the front page uses HTML+CSS, the background uses NodeJS to realize the server end technology, uses the MySQL as the background database, combines the VUe.JS framework and the ElementUI component library, and realizes the game sharing platform based on the MVVM framework.

This platform needs to set the appropriate authority: the management center, the friend center and the publication page need to log in to access, in the home page and view the work page does not need to set appropriate permissions.

**Keywords:** small game; sharing system; NodeJS; JavaScript;

致 谢

在整个设计实现的过程中，由于是用的新学的框架，遇到了许多的难题，系统出现过挺多的bug。但都在指导老师和同学的帮助下顺利解决了。指导老师用渊博的知识和严谨的治学作风使我受益匪浅，对我顺利完成本课程设计起到了极大的作用。在此向他表示我最衷心的感谢！感谢老师在课题设计研究初期给予我一定的帮助，才得以有一个很好的一个开始，即使在设计的过程中发现有很多的不足但是都能够得到老师相应的指导并解决问题。在完成设计的过程中，我不仅得到了其他老师指导还得到了许多同学的热心帮助，我想对老师的培养表示深深的感谢。编程其实是一件挺苦的事，独立完成一个系统更是难上加难，从一开始的界面设计，到一次次的讨论，分析，修改，测试。我接触到了许多过去不曾遇到的问题，学习了到了许多。

论文的完成是一个里程碑，标志着我大学生活的结束，同时也标志着我新的社会生活的开启。在此，诚挚的感谢培养了我四年的母校，感谢她接纳了曾经稚嫩的我，培养我；同时，也感谢这四年辛苦付出的上课老师，指导老师，辅导员，校领导。希望母校越来越好。