

分类号_____

编 号_____

U D C_____

密 级_____



南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

本科生毕业设计（论文）

题 目： 基于 Spring Boot 的校园论坛

姓 名： 陈仕镛

学 号： 11811202

系 别： 计算机科学与工程系

专 业： 计算机科学与技术

指导教师： 唐珂

2022 年 6 月 2 日

诚信承诺书

1. 本人郑重承诺所呈交的毕业设计（论文），是在导师的指导下，独立进行研究工作所取得的成果，所有数据、图片资料均真实可靠。
2. 除文中已经注明引用的内容外，本论文不包含任何其他人或集体已经发表或撰写过的作品或成果。对本论文的研究作出重要贡献的个人和集体，均已在文中以明确的方式标明。
3. 本人承诺在毕业论文（设计）选题和研究内容过程中没有抄袭他人研究成果和伪造相关数据等行为。
4. 在毕业论文（设计）中对侵犯任何方面知识产权的行为，由本人承担相应的法律责任。

作者签名：陈仕锦

2022 年 6 月 2 日

基于 Spring Boot 的校园论坛

陈仕镛

(计算机科学与工程系 指导教师：唐珂)

[摘要]：随着网络技术和和互联网的发展，网页和软件逐渐出现并改变着人们的生活和交流方式。论坛，即 BBS，是一种网络交流平台，在网络世界有着深刻的影响。通过校园论坛，学生们可以进行问答、交流生活体验、表达情感、钻研学术、分享工作心得和就业信息等等；校园官方机构可以认证用户、发布动态、组织活动等等。一个功能完善的、有组织管理的论坛可以满足学生和学校的各方面需要，增加学生生活的便利性并提高学校的知名度。本校园论坛系统是基于 Web 的动态网页。前端使用 Vue 作为框架，设计并美化动态页面。后端使用 Spring boot 框架进行快速开发，并使用 Maven 快速整合第三方常用框架。数据库使用 MySQL 关系型数据库进行数据管理。项目前端与后端分离，能在云服务器上运行。

[关键词]：Spring Boot；校园；论坛；Vue。

A Bulletin Board System based on Spring Boot

Chen shipu

(Computer Science and Engineering supervisor: Tang Ke)

[ABSTRACT]: With the development of network technology and the Internet, web pages and software gradually appear and change people's life and communication. A forum, or BBS, is a kind of network communication platform which has a profound influence on the network world. Through the campus forum, students can ask questions, exchange life experiences, express emotions, study academics, share work experience, employment information, etc. Campus official institutions can authenticate users, publish news, organize activities, etc. A well-organized forum can meet the needs of students and schools in all aspects, increase the convenience of students' lives, and improve the school's visibility. The campus forum system is a dynamic web page based on the web. The front-end uses Vue as a framework to design and beautify dynamic pages. The back-end utilizes the Spring Boot framework for rapid development and Maven to quickly integrate third-party frameworks. The database uses a MySQL relational database for data management. The front-end and back-end of the project are separated and can be run on cloud servers.

[Keywords]: Spring boot; campus; BBS; Vue.

目录

1. 绪论	1
1.1 背景	1
1.2 目的	1
1.3 调研	1
2. 需求分析	1
2.1 创新功能	1
2.2 普通用户需求	2
2.3 认证用户需求	2
2.4 管理需求	2
3. 前端设计	2
3.1 样式设计	2
3.2 路由设计	3
3.3 连接设计	3
3.4 测试设计	4
3.5 前端工具	4
4. 后端设计	5
4.1 连接设计	5
4.1.1 连接前端	5
4.1.2 连接数据库	5
4.2 封装设计	5
4.3 权限设计	6

4.4 测试设计	6
4.5 后端工具	6
5. 数据库设计	7
5.1 表单设计	7
5.2 数据库工具	10
6. 服务器设计	10
6.1 服务器配置	10
6.2 服务器部署	11
6.3 服务器工具	12
7. 总结和展望	12
参考文献	14
致谢	15
附录	16

1. 绪论

1.1 背景

学校里没有官方的论坛。通过网络和以前的同学，我了解到国内知名的高校基本都有自己内部的官方论坛，可以供学生和官方使用。经过在南方科技大学四年的学习生活体验，我感觉需要一个类似论坛的平台，供师生交流，供学校机构发布信息。

1.2 目的

通过校园论坛，学生们可以进行问答、交流生活体验、表达情感、钻研学术、分享工作心得和就业信息等等。通过校园论坛，校园官方机构认证用户、发布动态、组织活动等等。一个功能完善的论坛可以满足学生和学校的各方面需要，增加学生生活的便利性和提高学校的知名度。

1.3 调研

对国内其他高校的论坛进行了一定程度的调研，如北大的北大未名，北邮的北邮人论坛，浙大的 cc98 等等。主要是观察它们的前端页面、使用了哪些前端框架，以及实现了哪些论坛功能。通过调研来获取论坛设计的大体思路和思考需要实现哪些功能。

2. 需求分析

论坛的需求大致来自四个方面，一是来自普通学生方面，他们的需求就是正常的论坛功能，例如登录、搜索、修改资料、查看帖子、发布帖子等等论坛功能。二是来自官方机构方面，他们需要有特定的标识，发布并设置一些特殊的帖子。三是来自论坛管理方面，论坛的管理人员，在收到诸如不良言论、不良行为、恶意攻击他人、泄露隐私信息等方面的投诉时，应当可以及时地禁言相应的用户或是关闭相应的帖子。四是来自游客方面，论坛可能考虑对游客开放某些板块或是开放某些功能，但出于对保持论坛环境的考虑，暂时只能用学号注册用户。

2.1 创新功能

与传统论坛不同的是，本论坛添加了认证官方用户、关注动态、接受通知等功能。因为出于论坛运作的实际需要，论坛管理方需要认证权威用户以便于同时

认证其发布信息的安全性和可靠性。普通用户可以关注认证用户或普通用户，以及时接收被关注者发布的帖子信息。如果普通用户被其他用户@，即被其他用户回复，被回复的用户的通知列表里应当也显示此通知。

2.2 普通用户需求

以下是普通用户的需求：

（1）可以查看并进入不同板块；（2）可以查看帖子列表；（3）可以按照指定的指标对帖子进行排序；（4）可以发布帖子并设置帖子属性；（5）可以发表评论；（6）可以回复他人的评论；（7）可以设置头像；（8）可以改名字；（9）可以改密码；（10）可以注销；（11）可以按照名字搜索帖子；（12）可以关注其他用户；（13）可以接收关注用户动态；（14）可以删除关注你的用户；（15）可以认证官方机构；（16）需要使用学号注册；（17）需要登录进入网站，未登录不能进入网站。

2.3 认证用户需求

- （1）可以在用户名后显示认证图标；
- （2）应当具备普通用户所有的功能。

2.4 管理需求

- （1）可以在用户名后显示管理图标；
- （2）可以管理帖子。
- （3）可以管理用户。

3. 前端设计

3.1 样式设计

前端的样式设计使用到了 element ui、html、css、js 等工具，主要使用的是 element ui 中的各类组件，如 Layout 布局，Container 布局容器，Icon 图标，Button 按钮，Input 输入框，Upload 上传，Form 表单，Table 表格，Pagination 分页，Message 消息提示，Dropdown 下拉菜单，Card 卡片等等。以下是各组件的简单介绍和在本项目中的应用：

- （1）Layout 布局：将页面 24 分栏，可以设置组件占据的分栏和组件之间的

分栏间隔。

(2) **Container 布局容器**：主要分为 **header** 顶端容器，**aside** 侧端容器，**main** 主体区域容器和 **footer** 底端容器。本项目中使用了顶端容器、侧端容器和主体区域容器。

(3) **Icon 图标**：包含一套常规的图标样式，主要用于选项的可视化。

(4) **Button 按钮**：包含一套常规的按钮样式，主要用于点击触发事件。

(5) **Input 输入框**：显示为一条细长的段框，使用键鼠能够在其中输入字符，本项目中在登录、注册、修改信息，以及提交表单中使用该组件。

(6) **Upload 上传**：显示为一个加号，通过鼠标点击或输入文件路径上传文件。本项目中用于上传头像和认证信息。

(7) **Form 表单**：本项目中主要用于发表帖子、登录等功能的实现。

(8) **Table 表格**：本项目中主要用于展示用户列表、帖子列表等等，并在表格中提供自定义操作。

(9) **Pagination 分页**：当前端返回的数据量过多时，可以使用分页组件分割数据，以便于用户查看。本项目中将分页和表格组合使用。

(10) **Message 消息提示**：常用于主动操作后的反馈提示。本项目中合法的操作成功后会以绿色的消息提示，非法的操作失败后以红色的信息提示。

(11) **Dropdown 下拉菜单**：本项目中用于折叠用户菜单。

(12) **Card 卡片**：卡片组件用于将同一信息体的数据聚合，不同信息体的数据分割。本项目中使用此组件分割和聚合数据信息。

3.2 路由设计

本项目主要使用 Vue 安装的 **vue-router** 库来管理项目的路由。在前端根目录下创建了 **router** 文件夹，在其中存储 **route** 信息。一个 **route** 信息体包含 **path** 路径，**name** 名字，**component** 组成部件，**children**（可选）子路由等信息。配置好 **route** 的父子关系后，声明一个 **router** 全局变量，即可调用 **router** 来使用路由，列如 **router.push({路径})**，跳转至路径。也可以通过 **router** 来存储和传递信息。

3.3 连接设计

因为后端与数据库连接，请求后端，即可由后端代理请求数据库，故前端只需要与后端进行直接连接，而不需要与数据库进行直接连接。

本项目中前端使用的端口是 8080，而后端使用的端口是 8081，当 8080 向 8081 发起请求时，这个请求被称为跨域请求，处于网络安全方面的考虑，此类跨域请求一般被禁止，故需要使用工具对此类跨域请求进行过滤，即合法的跨域请求可以通过过滤器，而非法的跨域请求不能通过过滤器。

本项目使用 `axios` 处理跨域请求。使用 `axios` 配置后端基础地址和超时时延，并在发送请求前和接受响应前分别添加拦截器。发送请求前拦截器用于添加请求头、判断请求是否合法等等。接受响应前拦截器用于判断接受数据是否合法等等。

3.4 测试设计

运行 `npm run serve` 指令可以在本地 8080 端口运行前端项目，并查看样式。`Chrome`（谷歌）浏览器中按 `F12` 可以调出开发者工具。开发者工具可以选取前端元素，并在元素选项中查看元素的各项数据，如边框间距、颜色等等。在开发者工具的控制台选项中可以查看错误类型，前端项目中使用 `console.log()` 函数可以在控制台选项中打印日志。在开发者工具的网络选项中，可以查看前端向后端发起的请求的请求头、请求体。在开发者工具的应用选项中，可以查看前端项目中使用 `localStorage` 在本地浏览器存放或删除的数据。

3.5 前端工具

以下是项目中使用的前端编译器：

（1）`VSCode`:是一个开源的，不收费的编辑器。

以下是项目中使用的前端语言：

（2）`html`:即超文本标记语言，浏览器对其进行渲染后显示为网页。

（3）`css`:指的是层叠样式表，用于定义如何显示 `html` 元素。

（4）`JavaScript`:是一种可内置于 `html` 中的脚本语言。本项目中使用其进行前端的动态设计。

以下是项目中使用的 `vue` 安装的工具库：

（1）`Element-UI`: 提供了一套简单方便的组件库。本项目中使用其包含的组件实现前端样式的快速开发。

（2）`Vue CLI`: 是用于进行快速开发 `Vue` 项目的系统库。本项目中使用其快速组建和打包项目。

（3）`less, less-loader`: `less` 用于定义变量，`less-loader` 是用于 `less` 编译为 `css`

的加载器。

(4) **vue-router**: 是 **vue** 用于控制路由的工具库。本项目中使用其对前端进行路由管理。

(5) **axios**: 是一个发送 **http** 请求的工具库。本项目中试使用其对跨域请求进行拦截和过滤。

4. 后端设计

4.1 连接设计

后端作为数据传输的中间枢纽，需要分别和数据库以及前端建立连接。

4.1.1 连接前端

前文提到前端与后端之间的请求为跨域请求，前端需要对跨域请求进行拦截和过滤。同理，后端也需要使用一个过滤器对此类请求进行拦截和过滤。即所有向后端发送的请求都必须经过此过滤器。过滤器可以通过 **addAllowedOrigin** 函数添加信任的源地址，通过 **addAllowedHeader** 函数添加信任的请求头，通过 **addAllowedMethod** 函数添加信任的请求方法。

后端的 **controller** 类中向前端提供了接口，前端可以通过设置 **url** 来向指定接口发起请求。前端可以在 **url** 中设置参数，以此传递给后端：一是前后端规定某子路径用于传递值，例如规定 **{路径名}/{id}**，则 **{路径名}/1** 表示前端传递的 **id** 值为 1；二是在 **url** 末尾设置 **{key}={value}**，例如 **{路径}/id=1** 表示前端传递的 **id** 值为 1。

4.1.2 连接数据库

连接数据库需要：安装数据库驱动，本项目中为 **MySQL** 的 **jdbc** 驱动；设置对应的数据源名称和数据库的连接地址；设置数据库用户名和密码。为了 **MyBatis** 实现映射，需要设置：**mapper** 文件夹路径；实体类文件夹路径。（以上在 **.properties** 文件中设置，**.properties** 文件是本项目的配置文件之一）

同时，**MyBatis-plus** 也需要进行对应的设置。

4.2 封装设计

Spring Boot 分层如下。**controller**（控制层）为后端总的控制器，用于接受前端向后端发起的请求。**service**（业务层）用于实现业务的核心逻辑和算法。**mapper**

（数据层）用于实现数据库 SQL 语句。**entity**（实体层）用于存放与数据库表单相对应的数据体。

定义 **Result** 类来封装后端对前端请求的响应。**code** 变量表示响应类型，例如 200 表示成功，400 表示参数错误，401 表示权限不足，500 表示系统错误等等；**msg** 变量表示返回的文本提示信息，与消息提示组件结合使用；**data** 变量表示返回的数据信息。定义 **GlobalExceptionHandler** 类对系统异常进行封装。同理加入 **code** 变量表示异常类型。

4.3 权限设计

后端将用户分成多类型的用户，如普通用户、认证用户和管理者。不同用户应当具有不同的权限。在设计数据库时，用户表应当有一个标志，来表示此用户是何种类型的用户。当用户没有登录却尝试进入网站时，应当拦截此用户请求。

后端主要使用 **JWT** 来实现此类权限管理。前端向后端发起登录请求时，后端会根据登录的用户名和密码，生成一个 **token** 返回给前端（含有效期）。当前端向后端继续发送请求时，会将此 **token** 放入请求头中以表明已经进行过用户鉴权。若后端收到不含 **token** 或含错误 **token** 的请求时，会返回一个错误信息，以指示前端返回登录页面进行登录。

4.4 测试设计

本项目使用 **Swagger**、**Postman** 对后端提供的接口进行测试。IDEA 中可以编写 **Junit** 测试以及查看日志和错误信息。同时，使用 **Chrome**（谷歌）浏览器的开发者工具，在网络选项下可以查看后端响应的内容。

4.5 后端工具

以下是后端项目使用的编译器：

（1）**IntelliJ IDEA**：简称 **IDEA**，功能强大，操作便捷，是工业界常用的 **java** 开发工具，本项目中使用其作为 **java** 集成开发环境。

以下是后端项目使用的工具和库：

（1）**Spring Boot**：是用来简化项目初始搭建和开发过程的框架。本项目使用其搭建网络框架。

（2）**Maven**：可以通过简短的代码，为项目进行工具库的版本管理。本项

目中使用 Maven 进行项目组件的管理，将工具库的<dependency>代码段复制到项目的 pom 文件中，然后进行项目的重新加载，IDEA 就会下载对应的工具库。

(3) Postman: 主要用来模拟各种 HTTP 请求，本项目中使用其测试后端提供的接口，与 Postman 功能类似，作为对比。

(4) Swagger: 提供了具有网页样式的 WEB 应用服务。本项目中使用 Swagger 测试后端提供的接口，与 Postman 功能类似，作为对比。

(5) MyBatis: 是一个基于 Java 的持久层框架。本项目中使用 MyBatis 来简化复杂的 sql 语句。

(6) MyBatis-plus: 是一个 MyBatis 的增强工具，在 MyBatis 的基础上只做增强不做改变，为简化开发、提高效率而生。本项目使用中 MyBatis 简化复杂的 SQL 语句。

(7) Velocity: 是一个基于 java 的模板引擎。本项目中将 Velocity 与 MyBatis-plus 结合，作为代码生成器来使用。

(8) Devtools: 是 Spring Boot 热部署工具。本项目中使用其快速运行后端项目（保存即可运行，不需要重启）。

(9) Hutool: 是轻量的 Java 工具类库。本项目主要调用其函数进行开发。

(10) JWT: 是一种用于前后端之间传递安全信息表述性声明规范。本项目中主要使用其进行用户登录的鉴权。

5. 数据库设计

主要是使用 Mybatis 框架来连接 MySQL 数据库。建立了 user 表（存储用户信息）、relation 表（存储关注粉丝关系信息）、section 表（存储板块信息）、post 表（存储帖子信息），comment 表（存储通知信息）、notice 表（存储通知信息）等表单。

5.1 表单设计

各表单列名、属性、类型如下表 1-6 所示。各表单之间的外键、主键联系如下图所示图 1 所示。

图 1 表单关系

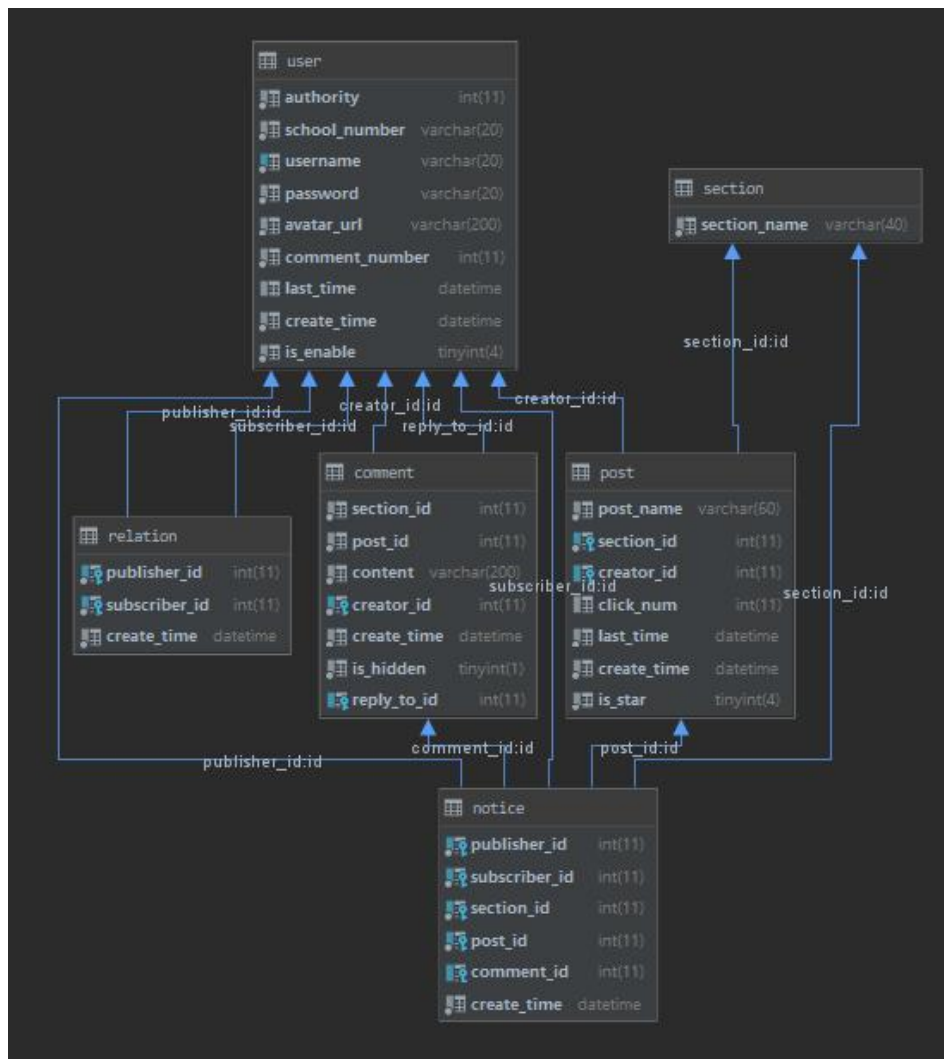


表 1 user 表

属性名	类型	说明
id	int	用户 ID
authority	int	权限
school_number	varchar	学号
username	varchar	用户名
password	varchar	密码
avatar_url	varchar	头像路径
comment_number	int	评论数
last_time	datetime	最后活跃时间

create_time	datetime	创建时间
is_enable	tinyint	启用

表 2 relation 表

属性名	类型	说明
publish_id	int	被订阅者
subscriber_id	int	订阅者
create_time	datetime	时间

表 3 section 表

属性名	类型	说明
id	int	板块 ID
section_name	varchar	板块名称

表 4 post 表

属性名	类型	说明
id	int	帖子 ID
post_name	varchar	帖子名字
section_id	int	所属板块
creator_id	int	创建者
click_num	int	点击量
last_time	datetime	最近回复
create_time	datetime	创建时间
is_star	tinyint	星标

表 5 comment 表

属性名	类型	说明
id	int	评论 ID
section_id	int	所属板块
post_id	int	所属帖子
creator_id	int	创建者
reply_to_id	int	回复评论 ID
content	varchar	评论内容

表 6 notice 表

属性名	类型	说明
id	int	帖子 ID
publisher_id	int	被关注者 ID
subscriber_id	int	粉丝 ID
section_id	int	所属板块
post_id	int	所属帖子
comment_id	int	所属评论
create_time	datetime	创建时间

5.2 数据库工具

MySQL: MySQL 是最流行的关系型数据库管理系统，关系型数据库将信息分割，存储在不同的表单中，以减少数据的耦合性。MySQL 提供命令行界面，输入用户名和密码后，可以对权限中的数据库进行操作。本项目中使用 MySQL 作为底层数据库管理系统。

Data Grip: Data Grip 用于实现数据库的可视化，以及可以快速连接各类型数据库、建立表单、修改表单、插入表单信息、实现事务管理、快速测试数据集等等。本项目中使用 Data Grip 软件对数据库进行操作，没有使用 MySQL 自带的命令行操作，是因为 Data Grip 中操作更加方便快捷，提供了可视化接口，而 MySQL 命令行需要复杂的指令进行操作。

6. 服务器设计

本项目使用了腾讯云轻量应用服务器进行项目部署。

6.1 服务器配置

服务器使用了广州地区的轻量应用服务器，操作系统经过重装后为 CentOS 7.6（使用服务器的系统镜像），公网 IP 为 43.138.166.217。

服务器具体配置如下：CPU 为两个核，内存为 2GB，使用的系统盘为 40GB 的 SSD 云硬盘，带宽为 4Mbps，流量包为 300GB/月。

由于服务器有防火墙对流量进行控制，为了项目在服务器的正常运行，特别添加了若干规则。具体规则如下：为了前端项目的正常运行，开放了 TCP 协议

下 8080 端口；为了后端项目的正常运行，开放了 TCP 协议下 8081 端口；为了 MySQL 数据库的正常运行，开放了 TCP 协议下 3306 端口；为了 HTTP 协议和 HTTPS 的正常使用，开放 TCP 协议下 80 端口和 443 端口；开放 TCP 协议下 22 端口用于用户进行 Linux 登录；开放 TCP 协议的 3389 端口使用户可以进行 Windows 远程登录；为了 Ping 协议的正常执行，开放了 ICMP 协议下所有端口。若要继续添加规则可在腾讯云轻量应用服务器控制台中配置。

6.2 服务器部署

本项目购买了腾讯云轻量应用服务器进行项目部署。购买后的服务器需要进行配置，卸载了原先运行的 Windows 操作系统，重新安装了 CentOS 操作系统用于项目的运行。出于服务器的安全设计，购买服务器之后无法得到原先的密码，需要重置密码才能使用新的密码进入服务器。

前端项目使用 Nginx 代理服务器进行部署，需要在本地对前端项目进行打包，使用指令 `npm run build` 可以调用 Vue CLI 框架进行快速打包。前端项目的根目录下会出现 `dist` 文件夹，此文件夹包含前端项目运行所需的文件。打包完成后，将得到的 `dist` 文件夹通过 Xftp 软件向服务器进行远程文件传输，即可将 `dist` 文件夹传输到服务器端。Xshell 软件可以以命令行的指令使服务器下载、安装 Nginx，使用 Vim 可以打开 Nginx 的配置文件进行配置修改。在 Nginx 程序目录下使用指令 `./nginx` 即可在服务器端运行前端项目。

后端需要使用 maven 的 `package` 打包指令（或在命令行中输入 `mvn clean package` 指令）在本地进行打包。后端项目的根目录下会出现 `jar` 文件包，此 `jar` 文件包中包含后端项目运行所需的所有文件。打包完成后，将得到的 `jar` 文件包通过 Xftp 软件向服务器进行远程文件传输，即可将 `jar` 文件包传输到服务器端。由于重新安装的 CentOS 系统没有自带 Java 环境，需要将本地的 `jdk` 文件通过 Xftp 软件向服务器进行远程文件传输，然后使用 Xshell 命令行安装 `jdk`。Java 环境配置完成后，使用指令 `java -jar {jar 文件包名称}` 即可在服务器端运行后端项目。

数据库不需要在本地向服务器端传输本地文件。使用 Xshell 命令行下载并运行 docker 和 MySQL 后，使用指令 `docker ps` 可以查看 MySQL 的进程号，然后使用指令 `docker exec -it {进程号} /bin/bash` 可以在 docker 容器中运行 MySQL。使用指令 `mysql -u{用户名} -p{密码}` 可以通过用户登入进入 MySQL 界面。设置

连接数据库的用户名和密码后，在本地数据库的编译器 Data Grip 中可以向服务器发起请求（需要服务器公网 IP，用户名，密码），连接成功后，可以在本地对服务器端的数据库进行远程操作。

6.3 服务器工具

（1）Xshell: Xshell 用来在 Windows 界面下访问远端不同系统下的服务器，从而实现较好地远程控制终端的目的。Xshell 是一个本地端模拟软件，它支持各种网络协议。本项目中，本地的操作系统是 Windows，服务器端的操作系统是 CentOS（linux），Xshell 提供用户远程登录和命令行界面，在本地使用 Xshell 命令行界面，输入 linux 指令可对远程服务器进行操作。

（2）腾讯云: 腾讯云是腾讯公司旗下的产品，为开发者及企业提供云服务、云数据、云运营等整体一站式服务方案，具体包括云服务器、云存储、云数据库和弹性 web 引擎等基础云服务。本项目中使用了腾讯云轻量应用服务器进行项目部署，腾讯云提供了轻量应用服务器的控制平台，在控制平台上可以进行开关机、重装系统、重置密码、查看服务器运行情况、查看公网和内网 IP 等等。

（3）Nginx: Nginx(engine x) 是一个高性能的 HTTP 和反向代理 web 服务器，同时也提供了 IMAP/POP3/SMTP 服务。本项目中使用 Nginx 来部署前端项目。

（4）Xftp: Xftp 是一个功能强大的 SFTP、FTP 文件传输软件。使用了 Xftp 以后，MS Windows 用户能安全地在 UNIX/Linux 和 Windows PC 之间传输文件。本项目中，Xftp 主要用于本地（Windows 操作系统）向服务器（linux 操作系统）传输文件。Xftp 在下载、安装完成后，被内置于 Xshell 软件中，可以在 Xshell 界面被调用。打开 Xftp 软件后，会显示本地和服务器两端的文件资源管理器，只需要拖动文件（或文件夹），即可实现本地和服务器之间的文件传输。

7. 总结和展望

本项目使用了多种工具独立开发了前端、后端、数据库和服务端。前端实现主要是为了体现设计的美感、给用户优质的使用体验。后端实现主要是为了权限管理和提供数据。数据库设计遵循数据库三范式，降低数据之间的耦合性。服务端提供通过 IP 地址访问网站的途径。

本项目完成了论坛系统的基本框架开发，但仍有细节之处等待完善。论坛的管理和运行需要大量人力物力，论坛的应用和发展未来可期。

参考文献

- [1]阳小兰, 罗明. 基于 Spring+SpringMVC+MyBatis 网上论坛的设计与实现[J]. 黑龙江科技信息, 2016(36):279-280.
- [2]刘广升. 基于 Java 的 BBS 论坛系统的设计与实现[D]. 山东大学, 2013.
- [3]林玲. “我国分布式供能系统发展论坛” 举行[J]. 上海节能, 2012(10):52.
- [4]彭冬, 蔡皖东. 面向 Web 论坛的网络信息获取技术及系统实现[J]. 计算机工程与科学, 2011, 33(01):157-160.
- [5]杨健. 基于 MVC 的论坛网站的设计与实现[J]. 计算机技术与发展, 2006(11):81-83.
- [6]李海峰. 基于 J2EE 技术开发高性能 BBS 论坛[D]. 华东师范大学, 2006.
- [7]杨世文, 侯超钧. 基于 SSM 框架的学术论坛管理系统设计与实现[J]. 计算机时代, 2021(02):25-28+33. DOI:10.16644/j.cnki.cn33-1094/tp.2021.02.007.
- [8]高云峰. 对基于 Java 的 BBS 论坛系统的设计与实现分析[J]. 电子技术与软件工程, 2018(04):202.
- [9]王宏玉, 徐步步. 基于 Java 的 BBS 开发[J]. 电脑知识与技术, 2016, 12(28):81-82. DOI:10.14004/j.cnki.ckt.2016.3654.
- [10]胡秉玺. 网上论坛系统设计与实现[D]. 西安电子科技大学, 2014.
- [11]裴博杰. 大学生学习交流论坛的设计与实现[D]. 电子科技大学, 2013.
- [12]俸学文. 基于 MVC 的校园论坛系统的研究与实现[J]. 电脑知识与技术, 2011, 7(17):4081-4082.
- [13]张丽华, 李德荣, 王冬星. BBS 论坛的设计与实现[J]. 电脑学习, 2009(06):48-50.
- [14]于士涛. 基于问答网络论坛知识体系的自动问答系统研究[D]. 南开大学, 2009.

致谢

首先衷心感谢对本论文做出悉心指导的唐珂老师, 尽管老师平时很忙, 依然抽空给予了耐心的指导, 提出了很多建议。在论文期间, 唐珂老师不断的鼓励给了我很大的帮助, 也让我重拾了信心。同时感谢陈文杰老师在修改论文时给予的帮助和指导性建议。

同时我还要感谢在大学期间教导过我的所有的老师们, 以及感谢实验课上的老师和助教, 在我遇到困难时给予了很多帮助。同时感谢致诚书院辅导员对我学习和生活上的关怀。

这次毕业设计是大学期间屈指可数的单人项目, 我曾经参与过小组的网站开发, 但此次的技术栈更广, 工作量更大。在项目进行过程中, 我学到了很多知识, 也得到了来自各方面的很多帮助。最后要感谢大学四年一直陪伴我的同学们, 尤其是创新实践小组内的同学, 在学习和生活上给予了我很多帮助, 在此衷心感谢。

附录

附录 A

以下是项目涉及的网址：

- (1) GitHub 仓库: <https://github.com/shiyel999/campus-bbs>
- (2) Vue: <https://cn.vuejs.org/>
- (3) 腾讯云: <https://cloud.tencent.com/>
- (4) MyBatis-Plus: <https://baomidou.com/>
- (5) Spring Boot: <http://c.biancheng.net/spring boot/>
- (6) Maven 仓库: <http://mvnrepository.com/>
- (7) elementUI: <https://element.eleme.cn/#/zh-CN>