

基于 JavaEE 的开发者论坛的设计与实现

中文摘要

在互联网技术的迅猛发展的今天，技术的更新换代也是非常的快，需要不断学习和进步才能适应知识的更新。互联网作为人们获取、发布和传播信息的重要途径，在人们的交流和学习中发挥着重要作用。所以开发一个论坛可以让用户互相学习的同时方便查找专业知识，这是很有必要的。它不局限于时间和空间，用户可以在论坛上发布他们的观点，还可以发布问题与众多用户一同讨论，并能快速解决。大家学习的效率将能大大提升，这也顺应了信息化发展的越来越快。

在敏捷开发的思想下，用户的基本需求可以从个人的角度快速确定，软件开发是以迭代、循序渐进的方式进行的。前端开发基于 bootstrap 框架，后端基于 Spring Boot+MyBatis 框架。采用 MySQL 数据库。

论坛主要的前台功能是文章、话题、用户、评论、配置和广告等模块。这些模块是相互关联的。配合相应的管理系统，完善对前端信息的监控。大量的代码封装和组件开发不仅提高了代码的可读性，也为将来的扩展奠定了基础。本文系统地描述了基于软件开发生命周期的整个项目的开发过程。

关键词：论坛；Spring Boot；MySQL

目 录

前 言.....	1
1 绪论.....	1
1.1 项目研究背景.....	1
1.2 项目研究意义.....	2
1.3 当前研究现状.....	2
1.4 论文研究方法.....	2
2 开发技术和工具.....	2
2.1 开发工具.....	2
2.1.1 IntelliJ IDEA.....	2

2.1.2 Navicat.....	3
2.2 主要开发技术	3
2.2.1 Spring Boot.....	3
2.2.2 Spring Security.....	3
2.2.3 MyBatis.....	4
2.2.4 Bootstrap.....	4
2.2.5 JWT.....	4
2.2.6 I18N.....	4
2.2.7 Thymeleaf.....	4
2.2.8 Websocket 协议	4
3 系统分析.....	5
3.1 系统功能需求概述	5
3.2 系统功能需求分析	5
3.2.1 用户功能需求分析.....	5
3.2.2 管理者功能需求分析.....	6
3.3 技术经济分析	6
3.3.1 技术可行性分析.....	6
3.3.2 经济可行性分析.....	6
3.3.3 社会可行性分析.....	6
3.4 用例分析	6
3.4.1 前台论坛用例分析.....	6
3.4.2 后台管理系统用例分析.....	7
4 系统设计.....	8
4.1 系统架构设计	8
4.2 系统功能模块设计	8
5 数据库设计.....	9
5.1 数据库设计原则	9
5.2 数据库 ER 图	9
5.3 数据库物理设计	10

6 系统详细设计与实现.....	17
6.1 前端主要功能实现	17
6.1.1 登录注册模块.....	17
6.1.2 文章列表模块.....	19
6.1.3 文章详情&评论模块.....	20
6.1.4 发布模块.....	22
6.1.5 话题模块.....	24
6.1.6 在线聊天模块.....	24
6.1.7 用户个人中心模块.....	26
6.1.8 设置模块.....	26
6.2 后台管理页面展示	27
6.2.1 登录界面.....	27
6.2.2 首页界面.....	28
6.2.3 用户管理页.....	28
6.2.4 文章管理页.....	29
7 系统测试.....	30
7.1 系统测试概述	30
7.2 接口测试用例	30
7.3 集成测试用例	31
7.4 测试结果	32
7.5 关于测试的思考	32
结 论.....	32
参考文献.....	35

前 言

目前，在互联网信息技术的强力推动下，不论我们来自何方只要在 Internet 可以接入的范围内，动动手指全球的知识和信息唾手可得。这是一个大数据的时代，无论是在生活、工作还是学习各个方面，统统离不开数据信息。在这样的背景下催生出熟人社交和陌生人社交以及特定群体社交三种形式，在这样的需求下，互联网上出现了各种类型的社交平台。

另一方面各种高级语言发展越来越成熟，比如 JAVA 语言，互联网技术迅猛的发展。J2EE 企业级技术给 Web 应用程序的开发提供了更多灵活的选择，以及 github 等开源社区的不断壮大，优秀的框架体系给开发人员提供了一种基于组件的设计和开发 Web 应用的模块化方法。再加上 Spring 的出现更是极大程度降低了企业应用开发的复杂性，2014 年以“约定大于配置”为宗旨的 Spring Boot 的提出更是让 Spring 家族再次焕发新春。

后端技术发展迅猛，各种前端框架的发展更是日新月异，如 Angular、React、Vue 等。前端框架最重要的就是页面的响应速度以及美化程度，这两点直接决定了能否留住用户。

随着互联网的迅速发展，使得虚拟社交逐渐超过现实社交成为主流，从六度空间理论（通过六个陌生人之间的关系可以认识世界上任何一个人）分析其结果就是：现实社交的弱化导致六度空间外拓（认识世界上的任何一个人需要更多的中间人），人与人之间的沟通层级增加，一种新的基于陌生人的避免信息泛滥的社交体系有待建立，以此来满足人们突破沟通层级，进行更多社交活动的欲望。

1 绪论

1.1 项目研究背景

在互联网技术的迅猛发展的今天，出现众多 IT 爱好者以及有志于从事 IT 行业的人员。然而，技术就是硬道理，技术的更新换代也是非常的快，需要不断学习和进步才能适应知识的更新。互联网作为人们获取、发布和传播信息的重要途径，在人们的交流和学习中发挥着重要作用。一些技术论坛纷纷出现，此种学习方式不受地点和时间限制，可以为开发者提供便捷的帮助。但是由于许多用户专研学习的专业方向不同，会出现问题不能及时发现和解决的情况。所以开发一个论坛可以让用户互相学习的同时方便查找专业知识，这是很有必要的。开发者论坛系统是以在线网站的形式，为用户提供简便并且快捷的交流空间，在论坛中用户通过发帖、回复的方式进行不仅限于技术学习方面的各种交流，结合管理员的协调与管理，能够成为一个稳定的、以开发技术为主题的技术交流平台。

1.2 项目研究意义

互联网已经渗透进了我们生活的方方面面,信息化的飞速发展给我们的生活带来无数便利,极大的提高了人们生活水平和质量。互联网不仅是一个信息来源,而且是一个虚拟空间,我们可以在里面相互交流。论坛正是其中之一。它不局限于时间和空间,用户可以在论坛上发布他们的观点,还可以发布问题与众多用户一同讨论,并能快速解决。大家学习的效率将能大大提升,这也顺应了信息化发展的越来越快。

1.3 当前研究现状

随着信息化的普及,国内的论坛数量数量惊人,各个领域都有相应的论坛,内容越来越广泛。各大门户网站都拥有自己的论坛,专业类的论坛也逐渐在其领域发展扩大。比如互联网技术类的有:csdn 是全国最大的开发者技术社区, chinaunix 是我国最大的 Linux/unix 技术论坛,还有 ITPUB 技术论坛也是与 csdn 并肩的;国外也要 Stack Overflow、SitePoint 等。

1.4 论文研究方法

本项目拟实现基于 Java EE 的开发者论坛,开发者论坛是一个网上 IT 技术交流的系统,它是以 Java 和 MySQL 数据库作为开发平台。根据各个用户角色需求的分析,去设计并实现的,系统分为后台与前台两大模块。前台功能主要是页面显示,帖子的浏览、发布帖子和回复、点赞、关注等;后台功能则是根据论坛后台管理员对论坛管理的需求去实现;论坛系统必定存在一个核心业务流程——帖子,本人通过对不同角色对帖子的需求分析,实现了前后台对帖子的功能需求,实现了前后台数据的交互,且数据都是实时更新的。

整个系统开发过程中,第一阶段:可行性分析,确定以现有条件可以完成本项目;第二阶段:通过调研,完成本项目的需求分析,编写需求文档;第三阶段:根据需求提取数据字典,设计数据库表,设计数据流图完成项目架构设计;第四阶段:编码实现预期的项目功能;第五阶段:对完成的项目功能和性能进行测试。

2 开发技术和工具

2.1 开发工具

2.1.1 IntelliJ IDEA

IntelliJ IDEA (Integrated Development

Environment, IDE) 是 JetBrains 公司旗下的一款 Java 集成开发环境工具软件, 正如其官方网站介绍的那样该工具的开发者们致力于打造一款世界上最强大, 最有效的开发人员工具, 该工具可以极大的加快生产速度, 是开发人员可以更快速的创建和编码, 该工具提供了大量的快捷键, 让开发人员很大程度可以摆脱鼠标的依赖, 极大的提高生产力。还有其智能的编码辅助, 该工具在建立源码索引后会在各种情况下提供建议, 他提供了开箱即用的体验, 2016.1 版本后自带 JDK1.8。它相对于 eclipse 来说更加智能, 而且 IntelliJ IDEA 自动保存代码, 在运行时先编译再运行, Eclipse 需要手动保存同时编译, 在运行时直接运行。最显著的功能是调试 (Debug), 可以调试 java 代码、JavaScript、jQuery、Ajax 和其他技术。您可以通过代码补全指出支持的方法。最后, 在多线程调试的情况下, 登录控制台功能有助于检查多线程执行的状态。

2.1.2 Navicat

Navicat 是一款可视化的数据库管理工具, 价格便宜、功能丰富。其所拥有的功能可以满足专业开发者的需求, 并且对数据库初学者来说也非常容易学习和上手。

2.2 主要开发技术

2.2.1 Spring Boot

该框架是由 Pivotal 团队与 2013 年研发, 主要用来简化 Spring 开发框架的开发、配置、调试、部署工作, 同时在项目内集成了大量易于使用且实用的基础框架[4], 它的出现, 极大程度的减轻了 Spring 一直被诟病的配置过多问题, Spring Boot 提供开箱即用和约定大于配置的策略, 大量的注解代替了 Spring 框架的 XML 配置, 如果想要自定义配置, 鼓励以代码形式编写配置类, 完全替代了 XML 文件。

2.2.2 Spring Security

Spring security 是一个安全框架, 它为基于 Spring 的企业应用系统提供声明性安全访问控制解决方案。从 Spring Security 5 开始, 要求您必须手动设置密码加密方法。Spring 官方明确指出 SHA 和 MD5 都不安全, 并推荐使用 BCrypt 加密。

BCrypt 加密: 是加 salt 的加密方案。当您对相同的密码时, 加密 MD5 会生成相同的哈希值。有些 MD5 加密方法可以被暴力破解。而 BCrypt 加密与之完全不同, 每次使用完全一样的密码生成出来的 hash 值都不一样。每次加密时, 生成第一个随机盐数, 然后通过密码对随机数进行 hash, 检索 hash 值并将其存储在数据库中, 使用的函数是 BCrypt.hashpw (password, BCrypt.gensalt ())。当用户登录时输入的是明文密码, 此时, 当 springSecurity.xml 文件身份验证管理器配置 BCrypt 加密方案时, 将从数据库中删除 hash 值并将其分开。

前 22 位加盐，然后，它通过组合由前端输入的随机数和密码来找到 hash 值，并确定它们是否相同。使用的函数是：`match BCrypt.checkpw (candidate, hashed)`。

2.2.3 MyBatis

MyBatis 的最强大功能之一是其动态 sql 语句。如果您以前使用过 JDBC，您将看到连接 sql 语句有多么困难，不要漏掉空格并且避免在 columns 列后面省略逗号等。而动态语句可以完全解决这些难题。Hibernate 采用面向对象的思想，通过封装 JDBC 操作实现 ORM（实体关系映射），但是在复杂的查询操作上增加了开发人员的学习成本，MyBatis 需要用户写更多的 SQL 代码，更多的通过用户逻辑来解决复杂查询问题，此外，MyBatis 在处理大量数据和高并发的情况表现更好。

2.2.4 Bootstrap

Bootstrap 框架是当下最为流行的响应式布局的前端开发技术，多用于移动端的开发。其开发出来的页面能同时适应 pc 电脑、平板电脑以及手机，还兼容主流的浏览器。使用 bootstrap 开发 web 非常的快捷，大大简化工作量，组件和 js 插件也非常的丰富，足以满足一般的项目开发。而且技术要求不高，只要会基本的 HTML 及 CSS 知识，再对照中文文档，即可快速上手。

2.2.5 JWT

JWT (JSON Web Token)，是当前最主流的跨域认证解决方案，解决了传统的 Session 认证扩展性差、耗费服务器内存资源的问题。官方提供 7 个字段，可以自定义配置，通过 Base64URL 算法将 JSON 对象序列化，这一部分是可逆的，不可存放敏感信息。签名部分是前两个部分的签名，以防止数据被篡改。

2.2.6 i18n

在信息领域，国际化 (i18n) 意味着能够在不显著改变产品的情况下适应不同语言和地区的语言需求。能够不修改程序内部的代码，也可以按语言或地区查看相应的界面。

2.2.7 Thymeleaf

Thymeleaf 是一个服务器端 Java 模板引擎，用于开发 Web 和独立环境项目。Spring 正式支持的模板呈现模板中没有包含 JSP。但是，Thymeleaf 和 SpringMVC 的自动配置集成，非常完美，而且成本不高。Thymeleaf 与 SpringBoot 完全集成在一起。SpringBoot 提供了 Thymeleaf 的默认配置，并设置了 Thymeleaf 的查看解析器。你可以像以前使用 jsp 一样操纵 Thymeleaf。代码没有什么不同。只是模板语法有所不同。

2.2.8 WebSocket 协议

由于 HTTP 协议存在着固定的缺陷——请求只能从客户端发起，是一种“你问我答”

的模式，导致 HTTP 协议不能主动的将信息推送给客户端，这种单向请求的特点，就导致请求只能由客户端发起然后服务端对客户端请求进行相应，而服务端有了变化客户端却无从知晓，比如：想要实现天气有变化实时通知给用户，过往通过客户端定期发送请求来解决，虽然后来开发人员对“轮询”的方式进行了升级出现了“长轮询”，但是“长轮询”相较于“轮询”不同是——在服务器有变化的时候给客户端相应，而不是之前“你问我答”的方式，但还是一种长连接，而 HTTP 是一个发送简单信息的昂贵的网络协议，这会导致无论服务器是否有新消息传送，都要建立昂贵的连接，导致资源极度浪费。因此 WebSocket 协议就诞生了。使用 WebSocket 协议的服务器和客户端都可以主动推送消息，双向平等的对话机制。可以通过多种 HTTP 代理服务器进行传递。可以发送文本或者以二级制的方式发送视频文件等，并且数据格式轻量，速度更快，效率更高，但是 Websocket 需要更高的浏览器规范，并且会给服务器带来延迟，并且还存在着一些问题，比如使用使用 HTTP 进行安全和混合内容处理[9]。

3 系统分析

3.1 系统功能需求概述

该论坛是用户之间交流的一个很好的平台，信息量有效分类，形式自由。首先是论坛前台方面，不像各种管理系统那样，必须登录账号才能访问，于是我就设计成对接口进行保护，需要登录账户才能访问的页面和 API 接口，在未登录的情况下无法访问。用户权限分为游客（未登录）只能访问开放的一些页面和服务器端的接口，当用户登录时，所有功能都可以访问，所有服务器端接口也会打开。同时必须要有对应的后台管理系统。

3.2 系统功能需求分析

3.2.1 用户功能需求分析

（1）游客功能需求分析

①游客不需要注册和登录就可以浏览论坛的文章列表、话题列表、查询指定用户。

②查询指定文章、查询指定话题、从数据库的层面看，游客只有部分查询操作，没有增删改的操作。

（2）注册用户功能需求分析

①用户注册后登录即可解锁全部功能，包括对文章的评论、点赞，对文章作者的关注，关注后的人将会出现在他的关注列表中，关注人的文章也会出现在关注人文章列表中。

②用户可以创建文章，上传图片，并且可以选择文章的分类和文章所属的话题分类。

③用户可以和其他在线用户一同进入聊天室发送文字进行在线聊天。

④用户可以修改大部分个人信息，唯一不可以修改的是用户的登录名，因为这作为用户的登录信息，在用户表中是和 id 一样的唯一标识。

3.2.2 管理者功能需求分析

(1) 因为是后台管理系统，所以必须登录才能访问。

(2) 管理者的身份在数据库中定义，管理员开放全部接口，对文章进行查看和删除，对用户的权限修改和重置用户密码，管理页面广告位的广告投放。

(3) 管理员除了可以管理前台信息以外，还能对后台运行的日志进行查看、下载和删除。

3.3 技术经济分析

3.3.1 技术可行性分析

本次开发采用 Java 语言，Java 语言凭借着其强大的特性在高级开发语言中经久不衰，市场占有率一直稳居开发语言排行榜前茅。配合近年流行的 Spring Boot 和 Mybatis 等框架编写出简洁的服务端代码，前端使用 bootstrap 框架，后台管理端使用 layui 能够快速构建出前端页面，并不需要自己动手写“轮子”，（软件领域中轮子理论指的他人已经做过的项目或者功能，直接拿来使用）。

现阶段我已经对以上所需技术基本熟悉，通过开发过程中一步一步地将技术不断打磨。通过计划开发周期安排，初步能确定在规定时间内能够完成需求分析确认下来的功能需求。Java 作为常用开发语言，遇到问题及 bug 在各大网站、博客等都可提供技术支持。所以有了这些技术上的支持，成功的开发一个论坛系统没有任何的技术风险。

3.3.2 经济可行性分析

本次设计开发的项目规模很小，目前项目的复杂程度和商业项目相距甚远，开发人员仅有我一人，因此仅需要一台电脑和相应的网上付费产品。而且开发出来的论坛运行起来所需要的电脑硬件要求不高，所以总的来说成本费用非常低廉。所以经济上是可行的。

3.3.3 社会可行性分析

本论坛的设计与实现过程中所使用的工具软件均为开源软件，并且本次开发不作商业用途，开发过程中如有借鉴到的代码全部是开源的，不会触及到国家、任何群体、个人的利益。所以社会、法律上也可行。

3.4 用例分析

参与者、用例以及这两者之间的关系是 UML 用例图的三个组成部分，首先要找到一个系统的参与者，并根据参与者需要系统做的事情建立用例，用例的名称可以命名为简单且易于使用。它通常是一个带动作性的词。

3.4.1 前台论坛用例分析

(1) 参与者主要有用户和游客。

(2) 用例主要涉及到注册、登录；文章的查询、点赞、关注、搜索、评论；个人中心主要涉及到修改个人相关的信息已经获取到个人在使用期间的一些数据；用户与用户之间的在线聊天等。用例如图 3.1 所示。

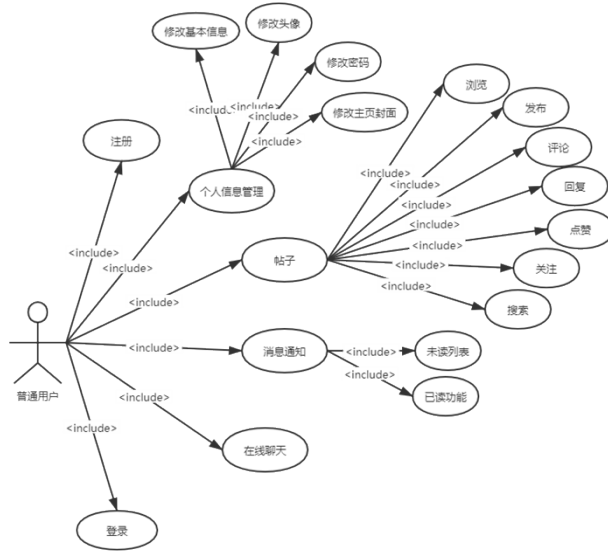


图 3.1 普通用户用例图

3.4.2 后台管理系统用例分析

(1) 参与者主要是后台的管理员。

(2) 用例包括管理员登录；对用户账号密码的重置和权限的修改；管理员对帖子的查看和屏蔽；对广告和新闻的增加删除；对论坛数据的统计分析。用例如图 3.2 所示。

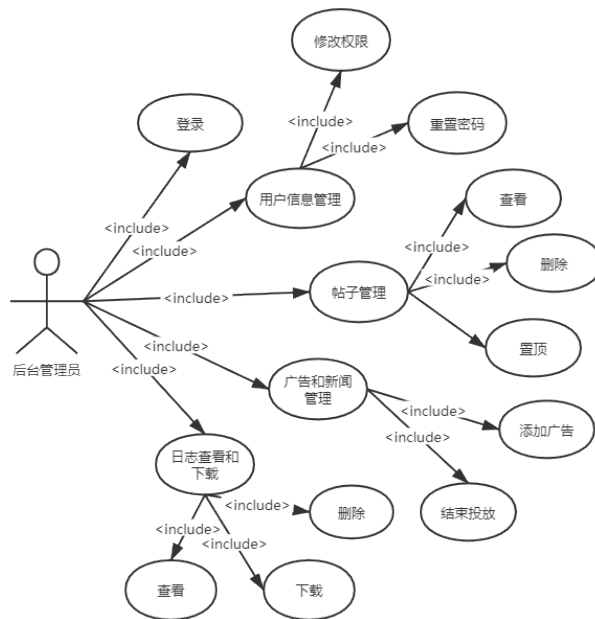


图 3.2 后台管理员用例图

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。

如要下载或阅读全文，请访问：

<https://d.book118.com/798007132025006116>