

杭州电子科技大学

毕业设计（论文）开题报告

题 目	基于 Spring Boot 的响应式技术博 客的设计和实现
学 院	计算机学院
专 业	软件工程
姓 名	朱文赵
班 级	14108413
学 号	14108337
指导教师	马虹

一、综述本课题国内外研究动态，说明选题的依据和意义

最早的博客是作为网络“过滤器”的作用出现的，即挑选一些特别的网站，并作简单的介绍。因此有人认为浏览器发明人 Marc Andreessen 开发的 Mosaic 的 What's New 网页就是最早的博客网页^[1]。Justin Hall 的黑社会链接网页也是最早的博客网站原型之一。

Blog 是继 Email、BBS、ICQ 之后出现的第四种网络交流方式，是网络时代的个人“读者文摘”，是以超级链接为武器的网络日记，是代表着新的生活方式和新的工作方式，更代表着新的学习方式^[2]。具体说来，博客(Blogger)这个概念解释为使用特定的软件，在网络上出版、发表和张贴个人文章的人。

现在市面上有很多开发博客的方法：

第一种方式，使用用现有的框架直接搭建，比如 GitHub Pages。首先 Github Pages 有一个很好的优点，就是它轻量级的博客系统，没有麻烦的配置，使用标记语言，比如 Markdown^[3]。还有就是无需自己搭建服务器，使用 Jekyll 模板系统，相当于静态页发布，特别适合博客，文档介绍。但是缺点也很明显，首先它的动态程序的部分相当局限，比如没有评论。再者就是由于网络的原因，很多同学可能很难访问。最后一个就是他有内存限制，每个人只有 300M 的空间。

第二种方式，自己注册一个其他平台的账号，然后开始编写属于自己的博客。这种呢，是很多同学都喜欢的一种方法，有点很明显，简单易用，比其他方式来的都快。但是普遍有人认为，各大博客的功能差别太大，除了基本 blog 的编写，其他做的真的不是天差地别，比如说有的人喜欢漂亮的界面(简书)，有的人选择内容的丰富、优质(CSDN)，有的人喜欢别的种类等(博客园)。很难把喜欢的一些功能放在一起，对很多人造成了困扰。

第三种方式，采用其他技术整合来搭建技术博客，比如现有的 SSM 整合，Python Web，Node 系列，这种方式，比较自由，能够使用自己想用的技术。但是同时难度也比较大，相比于 SpringBoot 开发流程比较繁琐，或者框架还没有到 Java 这种这么成熟，不能很成熟的操作其他的技术栈^[4]，比如说 NOSQL，又或者技术变化太快，自己无法跟上更新迭代的速度。

技术博客在众多博客类型中又有着一个不一样的地位。这些博客记录了博主曾经走过的路，是如何考研，做项目，学习算法，如何读书，如何思考，如何去准备面试，如何进入谷歌，facebook，微软这样的公司实习或在学习或者工作中学到的新技术或是碰到的技术难题以及解决之道。同时记录了他们对待各种技术的看法，每个人可能做得时间不一样，但他们对此有相同的态度，通过坚持，都有了自己的收获。如我一样，很多人在遇到一些“疑难杂症”的时候，都会去网上苦苦寻求，当读到他们读到了他们这些文字，这些曾经他们也遇到过同样的问题的时候。就会看到方向，不再迷茫，学会了知识，懂得了记录的重要性。

这些博客各有特点。比如说国内的 CSDN，简书；国外的 Tumblr，Github Pages，The Hacker Blog^[5]。同时还有一些个人站点，如：程序猿 DD，廖雪峰的官方网站等。这些都不失为值得学习借鉴的博客。对比他们的技术特点，不难总结出以下几点：

- 有的在界面上很出众，注重“表面”的开发，同时后台也不差，能够把内容更好的展示出来。一般国外的技术博客都有这个特点，相比之下国内的有些博客就不是很注重外观，究其原因可能是技术不愿意更新换代^[6]；

- 国内的博客对于国内程序员，很友好，对他们的成长提供了很大的帮助，比如说有时会提供一些外文文档的翻译，或者一些对国外新技术的介绍和使用；然后国外呢，一般都是国内现有答案无法满足用户的搜索要求，大家会采用谷歌来搜索自己想要的问题答案，比如在 Stack OverFlow 中进行搜索，或者到相关社区进行提问，等待别人的解答；

- 有的会提供自定义代码模块，让用户能够自己小规模建站^[7]，让他们的博客与众不同；

- 有的有机器学习模块，能够更加智能的根据用户的行为，如点击，浏览，收藏，转发等，智能推荐给用户相应的模块或者内容；

- 有的不提供注册，只能个人使用，如一些个人站点；

由此可见，这些技术博客各有优势，也有不足，难以满足中国技术人员的个性化需求。因此，本毕业设计将建立一个技术博客，能够让中国技术人员记录下自己在学习、工作中遇到的一些问题、难点、心得等。并且提供注册，开放给其他人使用，努力将其建立一个程序员交流学习的好地方。同时使用最近火热的技术 SpringBoot 来作为技术支持，让建站变得更加容易，更易于部署，更加容易和其他技术栈结合，以便能够专注于开发逻辑，技术的选用，而不在搭建和繁杂的配置项。

二、研究的基本内容，拟解决的主要问题：

2.1 本课题研究的基本内容

本毕业设计选用 SpringBoot 框架，结合 Thymeleaf，SpringData，SpringSecurity，Elasticsearch 等技术，旨在为技术人员设计并实现一款用于记录并分享技术文档的技术博客。通过该技术博客，方便技术人员记录自己工作和学习过程中的点滴，不断地进行技术的总结和积累，从而提升自己的综合能力，并通过博客这一平台，把自己的知识、经验、教训分享给大家，为志同道合者提供一个相互交流、共同学习的平台，促使更多的人共同进步。

本毕业设计中将主要实现用户模块、角色模块、权限模块、博客模块、评论模块、点赞模块、分类模块、标签模块、搜索模块等功能模块。会根据用户平常阅读的喜好，

搜索的习惯，智能推荐给用户相应的优质内容。方便技术人员能够更加自由的书写博客，以及分享自己的学习心得，结识一些有相同爱好的极客。后期也可根据用户的需求做适当调整。

具体的功能如图 2.1 所示：



图 2.1 功能模块图

（1）用户管理模块：用户可以进行登录、注册和搜索其他用户，超级管理员具有用户的增删改查的超级权限。

（2）全文搜索模块：此模块为本项目的一个亮点，采用大数据分析平台 elastic-Search，将后台数据库中的博客信息，用户信息，标签信息，分类信息，评论信息等同步到改平台中。用户在全文搜索的时候，后台直接从 es 中进行查询，增加了响应速度和响应精确性^[8]。比直接用 sql 语句中的 like 查询面积要广，而且还具有分词效果，匹配度高的显示在前面，低的显示在后面。

（3）博客管理模块：此模块是本系统的一个重点。用户可以自由的对博客增删改查，同时可以对博客设置相应的标签。还可以对博客进行最新最热排序。同时系统还会对每一个博客进行阅读量的统计，让别的技术人员能够更加直观的感受该博文的受欢迎程度^[9]，同时也给全文搜索给了一个指标。

（4）评论管理模块：用户可以对博客进行评论，让更多的人看到自己的想法，能够与别人进行交流和互动。

（5）点赞管理模块：通评论模块一样，用户可以对自己喜欢的博文进行点赞，或者

对自己已经不感兴趣的博文进行取消点赞。同时也给全文搜索提供一个硬性指标。

(6) 安全设置模块：用户只能对自己编写的博客才有删除的权限，对其他的无法进行删除操作。用户只能在自己拥有该权限的时候，才能进行相应的操作，这个通过 SpringSecurity 来进行管理^[10]。

(7) 个性推荐模块：会通过采集用户的行为信息，比如阅读时长，点赞，收藏，创建的标签等，前期通过后台的一定逻辑，清洗这些数据，最后形成一个简易的用户画像。后期可能会使用一些算法，如协同过滤，逻辑回归等，来简历数据模型，并持续训练，致力于向用户推荐更加精准的博文^[11]。

2.2 拟解决的主要问题

(1) SpringBoot 与其他技术栈的整合，致力于打造一个无 xml 配置的系统。

(2) NOSQL 的使用，如 Redis 在分布式 session 中的操作，以及分布式锁的实现，用来解决存在情况不是很高的高并发的场景^[12]。又或者 es 的使用，用来处理全文搜索的技术难点，又或者 mongodb 用来存放文件，用作图片、文件服务器等技术问题。

(3) Thymeleaf 模板引擎的使用，以及结合如何使用 Swagger 来编写 RESTful 接口^[13]，形成一个完整的前后端交互接口文档。

(4) 数据库表结构的设计与编写，还有 mysql 中的数据与 es 的同步问题，这些都需要一个个解决。

(5) 权限中使用 Spring AOP 先过滤掉大量的请求^[14]，再使用 SpringSecurity 来拦截用户具体的权限。更加合理地设置用户的权限等等。

(6) 对 python 的重新学习，学习相关数据结构与算法，设计出一个有深度学习的技术博客。

(7) 对 Java8 中的流式操作，进行一个实战，把自己所学到的设计模式、设计理念和编程方法^[15]，运用到改博客系统中，将自己所有所学所用都进行一个实战。

(8) 克服线上服务器的搭建，以及域名的选购，还有 https 的申请等问题。

(9) 要合理安排毕设编写时间和工作时间，做到二者都不互相打扰，争取把它们都做好，努力学习，正常毕业，持续提升自己的能力。

三、研究步骤、方法及措施：

1. 研究的步骤

1.1 搜集用户需求

通过百度，谷歌搜索一些技术人员在使用博客的时候的一些心得，结合自己的所做的调查问卷的搜集结果，总结出用户真实的需求。

1.2 分析用户需求

得到用户需求后，将其解析成为程序员能读懂的文字，后用相关的工具软件列出要开发的系统的大功能模块，每个大功能模块有哪些小功能模块，对于有些需求比较明确相关的界面时，在这一步里面可以初步定义好少量的界面^[16]。

1.3 概要设计

软件系统进行概要设计，即系统设计。概要设计需要对软件系统的设计进行考虑，包括系统的基本处理流程、系统的组织结构、模块划分、功能分配、接口设计、运行设计、数据结构设计和出错处理设计等^[17]，为软件的详细设计提供基础。

1.4 详细设计

在概要设计的基础上，需要再进一步进行软件系统的详细设计。在详细设计中，描述实现具体模块所涉及到的主要算法、数据结构、类的层次结构及调用关系，需要说明软件系统各个层次中的每一个程序(每个模块或子程序)的设计考虑，以便进行编码和测试。应当保证软件的需求完全分配给整个软件^[18]。详细设计应当足够详细，能够根据详细设计报告进行编码。

1.5 编码

开始具体的程序编写工作，分别实现各模块的功能，从而目标系统达到预定的功能、性能、接口、界面等方面的要求。

1.6 测试

在开发过程中，一边开发一遍进行功能模块测试；开发结束后，根据需求文档，进行功能和性能测试。

1.7 维护

若要上线，需要对服务器的压力进行监测，用户的反馈进行修复，提供 [github](#) 的交流社区。

2. 研究的方法

开发时候需要使用很多方法。比如说使用访谈法、问卷调查法以及网上搜索来搜集用户需求；使用面向对象的方法，结合多种设计模式来实现自己的系统；也可使用敏捷开发结合瀑布模型的方法^[19]，快速开发，让自己有东西可测，有东西可调整。

3. 研究的措施

- 1.搜集国内外文献，了解最新动态，通过问卷，访谈等形式透彻分析需求。
- 2.学习 [Spring Boot](#) 相关技术栈，丰富自己的技术能力；同时学习相应的设计模式，设计理念，把程序的可复用最大最大化，对代码要有一种高可用、高性能的追求。
- 3.进一步学习关系型数据库和非关系型数据库^[20]，将理论付诸于实践，边做边学。
- 4.学习国内外相关博客的设计理念。比如界面设计，UI 交互等。

5.尝试使用推荐算法，如协同过滤，逻辑回归、DNN、因子分解机和梯度提升决策树等^[21]。就算用不上，也要把 python 重新捡起来，不能荒废。

6. 完成基于 Spring Boot 的博客系统，持续测试，积极修复 bug。并支持注册，欢迎广大技术爱好者前来使用。

四、研究工作进度:

序号	时间	内容
1	2017.12.8——2017.12.20	查阅文献, 外文翻译, 撰写文献综述
2	2017.12.21——2018.1.1	课题调研, 收集资料, 撰写开题报告
3	2018.1.2——2018.1.31	需求分析, 编写需求文档
4	2018.2.1——2018.2.20	拟定技术实现方案
5	2018.2.21——2018.3.20	产品概要设计和详细设计
6	2018.3.21——2018.4.10	系统编码实现
7	2018.4.11——2018.4.20	代码调试优化
8	2018.4.21——2018.4.30	产品测试、改进
9	2018.5.1——2018.5.31	撰写毕业设计论文
10	2018.6.1——2018.6.17	毕业论文修订、定稿,准备毕业设计答辩

五、主要参考文献:

- [1] 马雄. 基于微服务架构的系统设计与开发[D]. 南京: 南京邮电大学,2017.
- [2] 陈涛,叶荣华.基于 Spring Boot 和 MongoDB 的数据持久化框架研究[J]. 电脑与电信, 2016(Z1): 71-74.
- [3] 王永和,张劲松,邓安明,周智勋.Spring Boot 研究和应用[J]. 信息通信, 2016(10): 91-94.
- [4] 郑彬彬. 基于微服务的 OJ 系统重构与优化[D]. 上海: 东华大学,2017.
- [5] 程化梅. 基于 React Native 的即时通讯应用的设计与实现[D]. 武汉: 武汉邮电科学研究院,2017.
- [6] 张峰.应用 SpringBoot 改变 web 应用开发模式[J]. 科技创新与应用, 2017(23): 193-194.
- [7] Pivotal 团队. Spring Boot Reference Guide1.5.3.RELEASE[OL]. 2017.
- [8] 朱荣鑫. 基于微服务架构的游戏商城服务端的设计与实现[D]. 南京: 南京大学,

2017.

[9] 汪云飞. JavaEE 开发的颠覆者: Spring Boot 实战[M]. 北京: 电子工业出版社, 2016: 1-15.

[10] 杨家伟. 基于 Spring Boot 的 web 设计与实现[J]. 轻工科技, 2016, 32(07): 86-89.

[11] Bhimani J, Yang Z, Mi N, et al. Docker Container Scheduler for I/O Intensive Applications running on NVMe SSDs[J]. 2018, 2(99): 1-1.

[12] Reddy K S P. Testing Spring Boot Applications[M]. Beginning Spring Boot 2. Apress, Berkeley, CA, 2017: 221-246.

[13] Reddy K S P. Getting Started with Spring Boot[M]. Beginning Spring Boot 2. Apress, Berkeley, CA, 2017: 21-33.

[14] Walls C. Spring Boot in action[M]. Manning Publications Co., 2016: 21-34

[15] Felipe Gutierrez. Spring Boot, Simplifying Everything[M]. Apress: 2014-06-15: 02-06

[16] Miller R A, Heitkamp E M. Spring boot: U.S. Patent Application 12/435,432[P]. 2010-11-11.

[17] Balalaie A, Heydarnoori A, Jamshidi P. Microservices architecture enables devops: Migration to a cloud-native architecture[J]. IEEE Software, 2016, 33(3): 42-52.

[18] Balalaie A, Heydarnoori A, Jamshidi P. Migrating to cloud-native architectures using microservices: an experience report[C]. European Conference on Service-Oriented and Cloud Computing. Springer, Cham, 2015: 201-215.

[19] Webb P, Syer D, Long J, et al. Spring boot reference guide[J]. Part IV. Spring Boot features, 2013, 10(3): 24-25.

[20] Prieto Barreiro I, Varela F, Mandilara E. Monitoring of CERN's Data Interchange Protocol (DIP) System[J]. 2018, 3(33): 3-5.

[21] Varela F, Gonzalez Corral M, Podgorski S, et al. MARS: Easing Maintenance and Interventions for CERN Controls[J]. 2018, 3(11): 3-4.

六、指导教师审核意见：

指导教师签字：_____

____2018____年____3____月____17____日

七、系（教研室）评议意见：

系（教研室）主任签字：_____

____2018____年____3____月____17____日

八、开题小组评审意见：

开题小组负责人签字：_____

_____2018 年 3 月 18 日

九、学院领导审核意见：

1. 通过； 2. 完善后通过； 3. 未通过

学院领导签字：_____

_____2018 年 3 月 19 日