**中国科学技术大学**

**工程硕士研究生学位论文**

**开题报告**

**论文题目：基于Restful架构的供应链电子发票系统的设计与实现**

**学 生 姓 名： 杨堃**

**学 校 导 师： 李春杰**

**企 业 导 师： 谢弘**

**工 程 领 域： 软件工程**

**领 域 代 码：** 430013

**研 究 方 向： 软件系统设计**

**所 在 单 位： 软件学院**

**中国科学技术大学研究生院**

**填表日期： 2017 年 04 月 06 日**

**说 明**

1. **工程硕士学位论文的开题报告是保证论文质量的一个重要环节，为了加强对工程硕士研究生培养的过程管理，规范其学位论文的开题报告，特制此表。**
2. **工程硕士学位论文开题报告，应该在工程硕士学位授予点或培养单位组织的学术报告会上报告，听取意见，论证后再填写此表。**
3. **此表一式两份经导师和培养单位负责人签字后，交培养单位研究生教学管理办公室存档。**
4. **工程硕士研究生在申请学位论文答辩时，必须提交该学位论文开题报告。**

**一、 简况**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **研究生简况** | **学 号** | | **SA15226379** | | | | | | **姓 名** | **杨堃** | | | | **姓名拼音** | **YangKun** | |
| **性 别** | | **男** | | **身份证号** | | | **142302199306070013** | | | | | | **出生年月** | **1993.06** | |
| **工程领域** | | **软件工程** | | | | | | | | | | **研究方向** | **软件系统设计** | | |
| **入学时间** | | **2015.9** | | | | **录取方式** | | | | **统考** | | | **培养方式** | | **脱产** |
| **本科毕业时间** | | | | | **2015．7** | | | | | | **本科毕业学校** | | **山西大学** | | |
| **论文类型与性质** | **名称** | **中文** | | **基于 Restful 架构的供应链电子发票系统的设计与实现** | | | | | | | | | | | | |
| **英文** | | **Design and implementation of electronic Invoicing system for supply chain based on Restul architecture** | | | | | | | | | | | | |
| **类别** | **1．技术攻关研究 2．工程项目策划**  **√3．工程设计或技术改造 4．新工艺、新材料、新设备、新产品的研制与开发** | | | | | | | | | | | | | | |
| **形式** | **√1．工程设计**  **2．研究论文** | | | | | | | | | | | | | | |
| **性质** | **√1．****应用技术研究**  **√2．技术开发** | | | | | | | | | | | | | | |
| **论文内容和意义** | **摘 要** | 进入二十一世纪以来，全球经济一体化体系越来越成熟，各个企业都逐渐拥有了稳定的产品供应商群体。随着采购收货开票业务量的迅速增长，传统的企业管理系统已经渐渐不能胜任处理这些数据关联性极强的业务需求。 以传统的处理方式，通常会通过人工的方式对采购-收货-支付流程中产生的收货单、发票进行数据核对。这样的方式在处理大量发票信息的时候，不免会增加人力、时力上的消耗。因此，企业对于电子发票开票前数据协商和自动化匹配的需求也将显著上升。  本系统是基具有 Restful 架构的Jersey平台开发，采用 B/S 系统架构，在企业用户接入网络的情况下，用户可以随时登录系统网页，并方便的完成电子发票相关的操作。使用本系统，能够及时就电子发票的各个字段、数据与供应商沟通修改，而且还可以自动以收货单为标准，核对电子发票中每件商品在价格和数量上是否正确，极大的节约了企业用户在人力和时力上的损耗。弥补了目前电子发票领域中在这一块的功能缺失。  本文着重介绍一套基于Restful 架构的供应链电子发票系统的设计与实现，该系统将基于B/S结构，采用 Java和Postgres+Elasticsearch编程来实现对文档数据的存储和操作。在后端开发模型中，遵照 MVC 模式进行开发。网络数据交互则遵照 HTTPS协议来实现与服务器端的通信。在后台开发环境与运行环境上选用 Postgres+Elasticsearch作为后台数据库服务器，采用 Jetty作为服务器端容器，最终完成具有基础功能的电子发票系统，满足用户在开票前的业务需求。 | | | | | | | | | | | | | | |
| **主题词** | **主题词数量不多于三个，主题词之间空一格（英文用“/ ”分隔）** | | | | | | | | | | | | | | |
| **中文** | | **电子发票 UBL商业文档 Restful 架构** | | | | | | | | | | | | |
| **英文** | | **Electronic Invoice /UBL/Restful** | | | | | | | | | | | | |

**二、选题依据**

|  |
| --- |
| 1. **阐述该选题的研究意义，或工程设计的价值和意义，国内外概况和发展趋势，选题的先进性和实用性，技术难度及工作量。**   **1.1选题的研究意义**  发票是单位和个人所开具和收取的业务凭证，也是审计机关、税务机关执法检查的重要依据。在我国当前的税收体制下，不同于欧美等发达国家以直接税作为税收主要来源，间接税占据了我国税收更高的比例。间接税是指纳税义务人不是税收的实际负担人，纳税义务人能够用提高价格或提高收费标准等方法把税收负担转嫁给别人的税种。2014年我国消费税、营业税、增值税以及进口环节增值税与消费税占据了60.38%的比例，而美国则是个人所得税与企业所得税占据了近60%的比例。由于开具发票的商品服务交易一般涉及到的税种为增值税、营业税、消费税等间接税，因此发票对应的是我国实现税收的重要凭证。  电子发票是指在购销商品、提供或者接受服务以及从事其他经营活动中，使用税务机关确定的标准开票软件开具、按照主管部门要求格式和方式存储和使用的电子收付款凭证。但是在国内，虽然有电子发票这一概念，但在电子发票的推广和使用上上还与国外有一定的差距。一方面，因为我国当前税收体制与其它国家有一定的区别，另一方面，电子发票系统在国内还没有形成一个完整稳定的生态系统。对于不同企业针对纸质发票的不同需求，在电子发票的设计与开票过程中，都有着不同的规范要求。因此，如何根据不同企业用户之间的规范要求来进一步约束保证纸质开票流程的准确无误是开发电子发票系统的一个关键点。这也是本文所需要解决的问题。  该系统在设计中将发票开票流程中，将企业用户对发票格式、字段的规范要求放在首要的考虑位置，通过及时的信息传递以及可修改的电子发票版式进一步加强了在纸质发票开票流程前，企业用户之间的沟通协商能力，保证了最终纸质发票开具的准确无误。  除此之外，与传统纸质发票相比，电子发票具有“无纸化、网络化、自动化”的特点，无纸化节省传统纸质发票的印刷成本，网络化提升税管部门信息管税的水平，自动化节约企业因手工申请、领用、开具、流转、检验造成的纳税成本，全方面提升管税水平。  **1.2国内外概况和发展趋势**   * **模拟电子发票国外发展现状**   在电子商务行业内，SAP 公司是全球最大的企业管理和协同化电子商务解决方案供应商。目前，SAP在120多个国家和地区拥有18,800多家客户、56,000多个系统安装点以及1000万多名最终用户，世界500强中80%以上的公司都在使用SAP的管理解决方案。因此，调研国外电子发票发展状况以 SAP 公司为主具有一定的代表性和全面性。  经深入调研分析，国外的SAP思爱普公司针对发票管理解决方案开发了一款产品，该产品主要由采购订单协同、发票协同、发票工作流、审批管理等几个功能组成。企业通过该产品可以有效控制开票流程，降低成本。通过电子单据集成到发票流程中，可以提高发票处理速度和准确性，并确保发票准时送达指定目的地。功能上相对来说是比较完善的，能够提供丰富的发票管理方面的功能（自动验证、匹配发票与采购单和收货单），最大程度的满足企业用户的工作需求。  但是针对中国企业的业务需求， SAP 目前的产品也存在一定的不足：1. SAP 的产品还没有着重考虑到开票前企业与供应商很可能会就发票信息进行频繁协商的情况。2. SAP 生态系统的互相关联导致其在第三方平台的移植、兼容性方面表现不友好。3. 基于中国发票特有的限额拆分特点，SAP 尚未针对这一特点提出有效的解决方案和相关产品。4. SAP 尚未设计出基于中国发票的电子发票格式。5. 该产品基于 C/S 架构，在部署、跨区域方面灵活性较差。  对比国外主流电子商务公司 SAP 的产品，本文所提出的供应链电子发票系统针对以上不足，设计出符合中国发票字段要求的电子发票样式，并提出了可行的解决方案。不仅在设计中考虑到企业与供应商就电子发票信息协作沟通的业务需求，而且基于 B/S 架构能够做到及时快速部署、友好兼容企业与供应商自带的 ERP 系统。此外，针对中国发票特有的限额拆分特点，在模拟电子发票模块实现了详细的拆分法则。   * **模拟电子发票国内发展现状**   目前，随着电子商务的快速发展，国内从事电子发票业务相关的企业也逐渐增多，比较典型的有航天信息股份有限公司、百望股份有限公司、汉特等等。  经调研分析，航天信息股份游戏那公司的发票管理系统以金税盘和电子发票平台为中心，以电子发票开具为基础，涵盖了发票开具、查验、推送等功能，但是由于基于 C/S 架构，在系统部署、配置、跨平台方面存在一定的约束条件，无法满足企业多平台、快速部署的需求。百望股份有限公司电子发票管理系统也基本覆盖了发票管理的所有功能，此外，因为该系统基于 B/S 架构，在平台兼容性、部署方面具有一定优势，但由于技术陈旧，目前也逐渐不能适应企业的新需求扩展。汉特公司开发的发票管理系统通过建立管理平台和数据接口，利用从 ERP 中导出的数据和扫描发票获得的数据，直接实现自动开票，消除发票流程中的手工劳动，保证开票的及时性和准确性。帮助企业建立统一的电子发票管理系统平台，将发票类型统一为增值税电子发票，实现电子发票自动化开具、集中存储管理、多渠道发票递送等功能。具有批量开具、查验下载、信息回传 ERP 系统等特点，并且在技术选型上，较为超前，但该产品对于设备的要求较高，系统稳定性较差，因此在企业客户中并没有得到很好的普及。  因此，通过深入调研分析，可以总结出目前国内电子发票管理系统具有如下的特点与不足：1. 绝大多数主流产品适配 Windows 系统。2. 产品技术架构陈旧、结构冗余，系统稳定性较差。3. 基于 C/S 的系统架构，产品具有复杂的安装步骤，在安装阶段、需要企业用户配置过多的用户信息。4. 移植性较差，大多数的产品系统未能集成企业已有的 ERP 系统。5. 国内产品主要考虑了开票和电子发票管理功能，尚未针对开票前企业用户与供应商用户沟通协作以及电子发票限额拆分做出可行的解决方案。  而本文结合国内产品的优点与不足，在系统适配方面，以 B/S 的轻量级系统 架构能够很好的适配 Windows、MacOS 两种操作系统。免去了繁琐的应用安装和配置过程，用户可以通过导入配置的方式做到快速配置。系统以基于 Restful 架构的 Jersey 轻量级框架作为后端业务处理的主要框架结构，保证了业务流程的稳定性。结合 Postgres+Elasticsearch 的独特优势，提升了数据的处理以及搜索能力。此外，在具体的业务场景中，着重考虑企业用户整个采购-收货-支付流程中在开票前，对于如何开具电子发票、开具后发票争议字段的协商、如何自动化核对电子发票数据以及发票限额拆分场景进行详细的讨论并提出有效的解决方案，弥补了国内电子发票管理系统在这方面的功能空白。  **1.3选题的先进性和实用性**  本系统结合国内外发展现状，基于供应链管理场景，设计并实现该场景下一种可行的电子发票整体解决方案，弥补目前供应链管理中电子发票流程缺失的开票前发票协商、自动化核对流程。以浏览器作为客户端，可以适配大多数的企业办公环境，并且对外暴露有集成企业 ERP 系统的接口，更方便企业用户直接导入商业数据，快速部署和维护，具有一定的先进性。  使用本系统，可以简化企业在采购收货付款开票流程中的人力时力损耗。并且针对收货单、纸质发票格式、字段、商品价格、商品数量等数字信息可以基于该系统与供应商进行及时的沟通和修改，从而保证最后开票和付款的准确无误，具有一定的实用性。  **1.4选题的技术难度和工作量**  该选题技术难度适中，能够满足研究生论文的研究要求；工作量适中，能够在预定的时间内完成课题研究的内容。  **2.主要参考文献（列出作者、论文名称、期刊名称、出版年月）**  [1]裘江南，刘燚，仲秋雁. UBL在供应链信息交换中的应用[J]. 微型机与应用, 2005, 24(9):46-48.  [2]赵博文. 解析Restful Web Service架构[J]. 光盘技术, 2008(10):4-4.  [3]潘冰. 面向资源的RESTful Web应用研究[J]. 网络新媒体技术, 2010, 31(7):38-43.  [4]雷映喜, 习淑婷, 彭俊峰,等. XML与JSON在WEB中对数据封装解析的对比[J]. 价值工程, 2013(9):210-211.  [5]金理清, 殷国真, 张正做. 基于B/S架构办公管理信息系统设计与开发[J]. 电脑编程技巧与维护, 2010(16):50-52.  [6]寇毅, 吴力文. 基于MVC设计模式的Struts框架的应用方法[J]. 计算机应用, 2003, 23(11):91-93.  [7] Monson-HaefelR,ChappellD A.Java message service[M].Cali- fornia:O'Reilly,2001.  [8]邓以克，王灿. 基于JMS的安全通信模型研究与设计[J]. 计算机工程与设计, 2009, 30(15):3526-3530.  [9]关辉, 李四海. 用HornetQ实现异步消息传输[J]. 软件导刊, 2010, 09(12):33-34.  [10]林河水, 程伟, 孙玉芳. PostgreSQL存储管理机制研究[J]. 计算机科学, 2004, 31(12):76-80.  [11]陈俊杰, 黄国凡. 应用Elasticsearch重构图书馆站内搜索引擎[J]. 情报探索, 2014(11):114-119.  [12]Shari Lawrence Pfleeger,Joanne M.Atlee.Software Engineering Theory and Practice.Higher Education Press.2009-12  [13]Thomas H.Cormen Charles E.Leiserson Ronald L.Rivest Clifford Stein.Introduction to Algorithms.China Machine Press.2004-01 |

**三、课题内容及具体方案**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1. 课题内容**  本课题主要是针对企业与供应商在开具纸质发票前信息沟通不方便的问题，而设计的一个UBL 格式电子发票系统。OASIS 的 UBL(Universal Business Language)作为一个基于 XML的信息交换通用语言标准可以解决上述问题。它为商业文档定义一个通用的XML交换格式, 并可以通过扩展满足特殊领域的要求[1]。  该系统设计主要包含以下4点内容：  （1）基于 B/S架构的网页客户端的设计与实现。  （2）基于 UBL 通用商业文档的多种特定文档数据类型的设计与实现。  （3）使用 HornetQ 消息队列处理文档数据状态的发送和变更。  （4）服务器端数据处理。  **2. 系统需求分析（应用软件工程专业描述工具描述）**  2.1 概述  世界经济发展越来越成熟，对于每一个成熟的企业，都有若干的供应商与其对应，保持着稳定的采购供应往来。随着企业业务规模的不断扩大，企业的采购需求也随之增加，在企业和供应商之间大批量的商业货物交易，在极大地推对企业发展的同时，也带来了一系列的问题。大批量的交易往来导致短期内收货发货信息的急剧增加。面对这样的问题，大多数的企业还是依靠人力或者是简单的表格系统来处理繁杂的货物信息，这样的处理方式不仅花费较多的时间，而且对于货物信息中的时间、价格等关键信息的互相检查、核对有着很大的疏忽遗漏风险。这是数据核对方面的问题，此外，在供应商发货后导出开票的过程中，由于供应商与企业缺乏有效的沟通机制，常常不能及时地就发票字段、折扣、货物数量、金额等字段进行商定，从而导致最后纸质发票数据不吻合的情况。  随着互联网技术的发展，B2B的软件架构已经越来越成为主流。通过集成企业与供应商自己的信息管理系统，将需要管理和比对的数据放在单独的模拟电子发票系统中进行处理是一种可行的解决方案。通过供应链电子发票系统，企业和供应商可以随时跟进每一份收货单、模拟电子发票和纸质发票的状态。此外，供应链电子发票系统提供了快捷的沟通服务，能够基于当前需要修改的文档，及时发送修改信息给对方，待双方确认无误后可以正确规范纸质发票的各字段格式和数值，从而在整个采购到供应的流程中规范发票流程，极大地便捷了企业和供应商的操作，减少了双方在这方面的资源消耗。[6]  2.2功能需求  供应链电子发票系统中包括了企业用户、供应商用户和系统管理员三种用户，根据其不同的角色对系统有不同的需求。    图2-1 企业用户和供应商用户收货单用例图  供应链电子发票系统主要服务对象是企业用户和供应商用户。对于企业用户，可以通过 FTP 上传接口将需要处理的收货单批量上传到模拟电子发票系统。企业用户可以在该系统中通过筛选，选择需要发送的收货单，在点发发送后，系统可以自动根据每一份收货单中的公司信息来找到正确的供应商，将对应的收货单发送给正确的供应商。供应商用户可以接收到企业用户发送来的收货单信息，并在该系统中筛选查看。可以根据需要勾选对应的收货单来直接导出模拟电子发票，其中的商品数据会从收货单中直接提取，从而保证了模拟电子发票在创建的时候就保持与收货单商品数据的正确对应。企业用户和供应商用户收单用例图如图2-1 所示。    图2-2 企业用户和供应商用户模拟发票用例图  模拟发票是供应链电子发票系统的主要数据文档，所有与纸质发票相关联的信息都在模拟发票中一一对应，任何一个针对模拟发票的操作在最后导出开票时，均会影响到纸质发票的票面信息。供应商用户在勾选若干收货单后，系统会根据这些收货单中的信息创建一份模拟发票，并同时更改对应收货单的状态，并且可以通过筛选选择对应的模拟发票发送给企业用户。双方可以将这份模拟电子发票视为电子版的纸质发票，在样式、字段方面均保持一致。对于该模拟发票中的差异信息，企业用户和供应商用户可以通过该系统的信息沟通服务来随时发送修改信息以征求对方的确定。这种及时的沟通方式极大的方便了双方的信息交换，简化了时间和人力的花费。企业用户和供应商用户模拟发票用例图如图2-2 所示。  系统管理员主要是对整个系统的配置信息进行管理。系统管理员可以根据企业用户的要求来配置各个公司用户的个性化配置。  本需求着重描述供应链电子发票系统的设计与实现。此系统主要包括收货单模块、模拟发票模块、状态转换模块、消息队列模块等。  2.3非功能性需求  1）浏览器系统：系统需要一定的浏览器版本支持，使系统在企业和供应商中都能稳定应用，目前支持 IE8、Chrome 45及以上版本、Safari、Firefox 等主流浏览器。  2）界面需求：对于企业用户和供应商用户，因为要处理大量的文档数据，界面应该设计友好，结构清晰，操作方便，有良好的用户体验；而对于管理员用户，界面需要能方便访问，对管理项目清晰明了，操作简单。  3）通信网络：根据系统要求，能在公共互联网上通信，访问系统服务器，实现数据的传送和接收。  4）数据存储：系统需要对所有的数据进行安全的存储，方便各种用户对数据的操作。在大量用户同时访问时，保证数据提取的准确性和高效性  **3、系统概要设计（应用软件工程专业描述工具描述）**  3.1 目标与概述  本设计将实现基于Restful 架构的供应链电子发票系统。Restful Web 服务是符合 REST (Representational State Transfer)风格的Web服务架构。Restful Web服务架构意味着,方法信息都在HTTP方法里; 作用域信息都在URI里。方法信息放到HTTP方法里就是指通过GET、PUT、DELETE等标准的HTTP方法来区分获得数据、修改数据或删除数据[2]。在REST中,整个Web被看作一组资源的集合,资源是一个基于URI(Universal Resource Identifier)的实体,包括实体对象和业务逻辑,对资源进行的操作由客户端指定的URI和 HTTP协议动词的组合来实施的。REST这种基于资源的设计改变了传统的基于动作的设计思想[3]。  系统将综合上传文档、搜索文档、创建并修改模拟发票以及消息通信功能于一身，为企业用户和供应商用户提供快捷的采购-收货-对账-开票服务。本设计需要在设计完成后，能实现以下功能：  1. 用户可以在浏览器端实现用户登录及用户注册功能，并进入供应链电子发票系统的主界面。  2. 在主界面可以实现上传文档、搜索文档、查看文档、创建文档、修改文档、发送文档等一系列功能。  3. 用户点击进入特定文档进入详情查看界面，在该界面不仅显示该文档的发送人、接收人、商品等必要信息外，还可以看到消息沟通的模块。通过该模块，用户可以及时发送修改并告知对方。  另外，本设计在性能上要达到安全性、可维护性和稳定性三项要求。安全性就是要求对系统管理员、企业用户和供应商用户实现对用户操作和数据库访问权限进行管理与控制，对用户的合法权进行保护。可维护性就是要求在系统开发过程中，系统文档、指示代码体系遵循软件开发的规范要求。稳定性就是要求系统必须保持性能稳定，在用户访问时能保证较好的响应速度。  3.2 系统总体设计设计  基于Restful架构的供应链电子发票系统主要是为用户在商业采购活动中能够快捷的处理众多数据繁杂的商业文档，并实现文档数据的准确迁移。因此系统要求能在企业的日常办公网络中正常使用。    图3-1 供应链电子发票系统B/S软件结构  供应链电子发票系统的总体架构如图 3-1 所示。用户通过浏览器访问供应链电子发票系统，发送需要的操作请求，请求数据在通过浏览器发送后，在服务器端会根据该请求路径解析成对应于一个特定对外接口的访问操作，同时，服务器端会从该请求中获取到访问接口必要的请求参数。通过定位的访问接口和请求参数，服务器端调用 Service 层业务逻辑处理方法；在 Service 层，具体的数据请求会通过 DAO 层间接操控数据库，从而获取到对应的返回数据。在经过一系列的逻辑处理后，该接口将结果数据以 Response、JSON 或者 XML的格式返回给前端，前端将响应数据解析后，发送回浏览器。其中XML是 ExtensibleMarkup Language(可扩展标记语言，有时候也写作eXtensible MarkupLanguage)的简写形式。可以用来标记数据和定义数据类型，是一种允许用户对自己的标记语言进行定义的源语言。JSON(JavaScriptObjectNotation)是一种数据交换格式，是以JavaScrip为基础的数据表示语言[4]。  该系统通过消息队列模型实现双方用户在文档数据、文档状态等方面的一致性，下图是系统的数据交换流程图。    图3-2 供应链电子发票系统整体数据交换图  3.2.1 Web端开发架构  B/S 架构，即浏览器/服务器（Browser/Server，简称 B/S）结构，在这种架构下，用户界面完全通过Web浏览器实现，一部分事务逻辑在前端实现，主要事务逻辑在服务器端实现，形成所谓三层结构。这种结构客户机只要安装一个浏览器如 Internet Explorer 或 Chrome、Firefox 等即可，服务器安装 SQL Server 或 Oracle、Sybase、MySQL 等数据库。浏览器通过 Web 服务器同数据库进行数据交互。考虑到采用 B/S 架构体系具有易开发、易移植、易维护、易使用等优点，本系统采用 B/S 架构[5]。  本系统的 Web 端的体系结构如图 3-3 所示，采用 B/S 模式，即浏览器/服务器模式。在 Browser/Server 的系统中，用户可以通过浏览器向网络上的服务器发出请求。Browser/Server 结构可分为三个单元：表示层（Web 浏览器）、功能层(具有应用程序扩展功能的 Web 服务器)、数据层（数据库服务器）。  为了方便用户在浏览器上进行操作，以及系统各个模块的相对独立，Web 端采用MVC（Model - View – Controller，模型-视图-控制器）设计模式，将数据显示和数据业务处理分开。MVC设计模式来源于面向对象语言 Smalltalk ,它的主要思想是将应用程序分为三个部分 :模型 (Model) 、视图 (View)和控制器 (Controller) ,以提高应用程序的可扩展性[6]。在设计对外访问接口的过程中，主要是有了具有 Restful 风格的 Jersey 框架，从而使各个对外接口都有准确无误的语义和独立性，使系统在今后的开发中更容易加入新的业务或新类型数据界面，增强系统的可扩展性。    图3-3 基于 B/S 结构的 Web 端架构图  3.2 功能模块设计  系统在浏览器界面的主要工作流程如图 3-4 所示。从图中可以看出，用户首先登录系统，或者注册后登录系统，分别可以进行查看文档、搜索文档操作，然后通过点击列表中的文档，进入对应文档的详细信息界面。针对每一份文档的详细操作都位于该界面中，用户可以检查文档数据是否正确，采用数据匹配模块来由系统进行自动核对账单，如果用户对某一条数据有修改操作，在修改后，文档详情界面的右侧会有消息通知模块，负责将该修改操作告知文档的另一方用户。当这些数据符合用户的要求时，用户可以在该界面确认接收，系统会将接收状态同时通过消息队列发送给文档的供应商用户，供应商用户在此之后便可以调用各地的税务系统导出纸质发票，从而保证了纸质发票各个字段、折扣、价格、数量上的准确无误。接下来的章节将分用户详细介绍收货单模块、模拟电子发票模块以及消息队列模块。    图3-4 供应链电子发票系统整体工作流程图  3.2.1 收货单模块的设计  收货单模块主要负责从企业采购部门获取 csv 格式的收货统计表格数据，并针对这些数据的一系列操作，不同用户下该模块工作流程如图 3-5 所示。    图3-5 供应商用户下收货单模块的流程图  从图中可以看到，供应商用户进入系统界面后，可以查看到企业用户发给它的收货单信息，默认界面显示的是当前收到的所有收货单列表。在列表界面，供应商用户可以设置搜索条件来过滤特定的收货单。供应商用户对目标收货单进行勾选，点击界面的创建模拟发票按钮，在这一操作中，后端会解析被勾选的收货单，并查询数据库找到模拟发票的发送方和接收方，以及必要的关联数据，然后创建一份新的模拟发票，界面会直接跳转到这份模拟发票创建完成的详细界面。    图 3-6 企业用户下收货单模块流程图  在收货单模块中，企业用户的主要工作是负责将清点后的收货核对数据表以 csv 格式通过供应链电子发票系统的FTP 上传接口上传到该系统中。在成功上传后，可以在收货单列表界面中看到已上传的收货单，企业用户可以根据搜索或直接点击进入收货单详细界面。在详情界面，用户可以点击发送按钮，将当前收货单发送给对应的供应商用户，期间需要的发送关联数据会从先前企业用户管理俺的供应商信息中获取。  3.2.2 模拟发票模块的设计  模拟发票模块是业务的主要部分，供应商用户通过勾选收货单生成模拟电子发票。之后在该发票被企业用户拒绝前，所有的主要业务都将基于该电子发票展开。供应商用户和企业用户对模拟发票的操作基本一致，但也有少许的区别，下面分用户介绍该模块。    图 3-7 供应商用户下模拟发票模块流程图  因为本系统是针对企业用户，所以在模拟发票模块中，主要的业务操作由供应商用户来负责，从而最大化的减轻企业用户的工作流程。对于供应商用户，可以在模拟发票列表中搜索已创建的模拟发票草稿，或者在接收企业用户发送过来的收货单信息后，通过筛选，选择需要的收货单创建生成模拟电子发票。在创建过程中，为确保数据的一致性，商品数量、价格、折扣等直接从收货单中获取。在此基础上，供应商用户可以针对每个字段信息进行修改，但这一修改会反馈给企业用户。在创建完成后，供应商用户可以将该模拟发票暂存为草稿、丢弃。在确认无误后，点击发送，系统即会通过消息队列将该模拟发票的一份原样拷贝发送给企业用户，等待企业用户的接收，或者修改反馈。在企业用户接受该文档后，供应商用户便会基于该模拟发票调用当地税务系统开具真正的纸质发票，之后将纸质发票的必要四字段信息回填到模拟发票中，发给企业用户进行最后的纸质发票与模拟发票的核对验证。    图 3-8 企业用户下模拟发票模块流程图  在该系统中，企业用户主要负责查看模拟发票信息是否正确，并给出相应的反馈状态。在供应商用户发送模拟发票后，企业用户可以在模拟发票列表中找到对应的文档信息。企业用户进入该文档后，如果发票信息数据量较少，用户可以通过自主核对后就可以给出反馈状态，如果数据量较大，则可以在该模拟发票界面，点击两单匹配，出发数据匹配模块，该模块会自动以模拟发票中的行为级别，与对应的收货单信息进行商品数量、商品价格两方面的比较，如果比较通过，那么系统会默认该模拟发票已符合企业用户的各项要求，会自动接受该模拟发票，并将接受状态回传给供应商用户。如果匹配不通过，系统会高亮显示具体的不匹配信息的行位置，企业用户可以基于该不匹配信息进行修改反馈，通过系统的会话模块与供应商用户进一步的沟通。在企业用户接受该模拟发票后，等待供应商用户开具的纸质发票和回填四字段的模拟发票，最后一步通过核对纸质发票与模拟发票来保证整个开票流程中各项数据的准确无误。  3.2.3 消息队列模块设计  本系统的消息队列模块主要采用 JMS+HornetQ 整合的方式实现，从而实现多个用户间的数据状态传递。  Java 消息服务(Java message service,JMS) 是 SUN 公司提出的旨在统一各种消息中间件接口的规范,它 提供了一组与具体实现无关的接口[7]。提供了如消息持久化,消息过滤和事务的服务，允许Java应用程序访问现有的各种消息中间件。但为了对底层的实现进行抽象。一个特定的 JMS 实现可以提供基于 TCP/IP, HTTP, UDP 或其它的协议[8]。  JBOSS的 HornetQ 可以看做是JMS服务一种实现，JMS 规定了消息传递的数据结构、调用方式等，而在 HornetQ 等这些消息队列中，依据 JMS 协议，实现了消息传递模型。[HornetQ可以用于构建多协议、嵌入式、高性能、集群及异步的消息系统。HornetQ 的核心只依赖于JDK，可以通过依赖注入框架如:Spring 或者 Google Guice 对其进行实例化[9]。  JMS支持两种消息传递模型：点对点（point-to-point，简称PTP）和发布/订阅（publish/subscribe,简称pub/sub）。这两种消息传递模型非常相似，但有以下区别:  a. PTP消息传递模型规定了一条消息只能够传递费一个接收方。  b. Pub/sub消息传递模型允许一条消息传递给多个接收方。  在该系统中采用的是点对点的消息传递模型，大致的模型图如下：    图 3-9 点对点消息队列模型  在该系统中，点对点的消息队列模型分别定义消息生产者和消费者，生产者把消息发送到JMS Provider的某个目标地址（Destination），消息从该目标地址传送至消费者。消费者一方通过创建一个监听器来异步接收消息，一般而言，异步消息消费者的执行和伸缩性都优于同步消接收者，主要体现在：  1. 异步消息接收者创建的网络流量比较小。单向对东消息，并使之通过管道进入消息监听器。管道操作支持将多条消息聚合为一个网络调用。  2. 异步消息接收者使用线程比较少。异步消息接收者在不活动期间不使用线程。同步消息接收者在接收调用期间内使用线程，结果线程可能会长时间保持空闲，尤其是如果该调用中指定了阻塞超时。  3.对于服务器上运行的应用程序代码，使用异步消息接收者几乎总是最佳选择，尤其是通过消息驱动Bean。使用异步消息接收者可以防止应用程序代码在服务器上执行阻塞操作。而阻塞操作会是服务器端线程空闲，甚至会导致死锁。阻塞操作使用所有线程时则发生死锁。如果没有空余的线程可以处理阻塞操作自身解锁所需的操作，这该操作永远无法停止阻塞。  3.3 数据库设计  对于供应链发票系统，主要的处理对象是 UBL 商业文档，需要存储文档信息、文档状态信息、消息转发信息等。数据库能实现配置管理系统的数据持久化，是系统的重要组成部分。本文将选用 Postgres+ Elasticsearch数据库进行后台数据库开发，建立各部分数据表。Postgres是一个跨平台的开源关系型数据库管理系统，具有支持复杂对象数据类型、支持用户自定义数据类型、支持“活跃”数据库和规则、减少系统崩溃后的恢复代码和代价等特点[10]。因此，本系统的设计采用 Postgres 作为主要的数据库进行数据库开发。  Elasticsearch是一款稳定高效的分布式搜索和分析引擎，它的底层基于Apache Lucence，提供了完整的 Restful API 来对数据进行操作。除此之外，Elasticsearch高可用(Highly-available)、高伸缩(Scalable)、分布式，这三个特性都是源于其简易但强大的分布式支持[11]。并且Elasticsearch对文档全局搜索有很好的性能，而这一点，也符合本系统大量文档类型数据的搜索要求。  各个表的功能与结构设计如下：  1.文档基本信息表（ documents）：   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 序号 | **字段名** | **字段说明** | **类型** | **是否为空** | **说明** | | 1 | uuid | 文档标识 ID | uuid | N | PK | | 2 | created | 创建时间 | timestamp | Y |  | | 3 | lastedit | 最近修改时间 | timestamp | Y |  | | 4 | actorid | 用户 ID | uuid | Y |  | | 5 | groupid | 公司 ID | uuid | N |  | | 6 | audited |  | varchar | N |  | | 7 | state | 文档状态 | varchar | N |  | | 8 | targettype | 文档类型 | varchar | N |  | | 9 | dispatchid | 转发 ID | uuid | Y |  | | 10 | documenttype | 文档详细类型 | varchar | N |  | | 11 | deleted | 已删除 | bool | Y |  | | **补充说明** | | documenttype: 包含文档类型的版本信息 | | | |   2.文档详细信息表（ portabledocs）：   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 序号 | **字段名** | **字段说明** | **类型** | **是否为空** | **说明** | | 1 | documented | 文档ID | uuid | N | PK | | 2 | description | 文档描述 | varchar(100) | Y |  | | 3 | displayed | 文档号码 | varchar | N |  | | 4 | issuedate | 发送时间 | date | Y |  | | 5 | total | 文档总金额 | numeric | Y |  | | 6 | currency | 货币单位 | varchar | Y |  | | 7 | otherparttenantid | 第三方公司 ID | uuid | Y |  | | 8 | otherpartactorid | 第三方公司用户 ID | uuid | Y |  | | 9 | otherpartname | 第三方公司名称 | varcahr | Y |  | | 10 | typeid | 文档类型 ID | int | N |  | | 11 | totalminustax | 含税价 | numeric | Y |  | | **补充说明** | | displayed：用户自定义的文档 Id，不是系统生成的 UUID 类型 | | | |   3.文档类别表( documenttypes)：   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 序号 | **字段名** | **字段说明** | **类型** | **是否为空** | **说明** | | 1 | type | 文档类别 | varchar | N | PK | | **补充说明** | |  | | | |   4.文档对象标识表( objecttypes)：   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 序号 | **字段名** | **字段说明** | **类型** | **是否为空** | **说明** | | 1 | id | 文档对象ID | int4 | N | PK | | 2 | type | 文档类型 | varchar | N |  | | 3 | classname | 文档类名 | varchar | Y |  | | **补充说明** | | classname：对应在后端中每一种文档类型对应的 Java 类名 | | | |   5.文档号码表（documentnumbers）：   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 序号 | **字段名** | **字段说明** | **类型** | **是否为空** | **说明** | | 1 | groupid | 公司 ID | uuid | Y |  | | 2 | documenttype | 文档类型 | int4 | Y |  | | 3 | number | 文档号码 | varcahr | Y |  | | 4 | documented | 文档号码对应文档 ID | uuid | Y |  | | **补充说明** | | number：表示当前文档类型所用的最大文档号码 | | | |   6.文档状态属性表（ documentmetadata）：   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 序号 | **字段名** | **字段说明** | **类型** | **是否为空** | **说明** | | 1 | documentid | 文档 ID | uuid | N | PK | | 2 | key | 文档属性名 | varchar | N |  | | 3 | value | 文档属性值 | text | Y |  | | 4 | system | 系统属性标识 | bool | N |  | | 5 | largevalue | 最大值 | text | Y |  | | **补充说明** | |  | | | |   7.文档内容表( documentcontent)：   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 序号 | **字段名** | **字段说明** | **类型** | **是否为空** | **说明** | | 1 | id | 文档序号 | bigserial | N | PK | | 2 | documentid | 文档 UUID | uuid | N |  | | 3 | type | 文档类型 | varchar | Y |  | | 4 | mimetype | 文档 MIMETYPE | varchar | N |  | | 5 | content | 文档内容 | bytea | Y |  | | 6 | original | 来源 | bool | N |  | | 7 | externalid | 扩展 id | varchar | Y |  | | 8 | ubl | ubl 格式 | bool | Y |  | | **补充说明** | |  | | | |   **4、拟采用的设计方法、环境等**  在网页客户端，采用 AngularJS 作为前端的开发框架，并通过 Ajax 技术实现前台和后台数据的异步逻辑处理及通信。  在服务端，因为该系统以 Restful API 为设计前提，所以在开发框架上选取对 Restful 风格支持较好的 Jersey 框架，并采用spring JDBC数据库连接池对数据库连接进行维护。在消息队列方面，采用 JMS 服务器+HonetQ 作为文档及状态转发的通信机制。  开发环境为Mac OS+IntellJ Idea，Web容器为Jetty server，数据库为Postgres+Elasticsearch，项目模块管理采用 Maven 控制。  **5. 技术难度及特色分析**  根据企业用户的调研访问，确定该系统不需要特别的扩展功能，希望在文档转发、状态变更通知、修改即时生效、数据一致性方面能够保持较高的稳定性和可靠性。因此该系统采用相对成熟的后端开发技术，用以保证系统开发的速度和稳定性。  该系统面向的用户是企业用户，供应商用户只是企业用户的商品供应方，并不是本系统的服务对象，只是提供给供应商用户一个与企业用户进行开票前数据核对沟通的系统。因此，本系统以企业用户为主，但在设计各模块的功能中，需要以用户权限来区别和限定两种用户的不同操作方式。  本系统技术难度分析  1. 基于 UBL 商业文档格式的收货单和模拟发票数据字段设计的合理性，需要最大程度上匹配大多数企业用户的字段要求（最好的方式是：如何设计成为可配置的）。  2. 对于企业用户和供应商用户一对多的情况，如何保证后端数据在当前用户下唯一性问题（文档在序号、文档标识、公司标识等字段组合搭配如何做到唯一性确定）。  3. 平台各模块包之间如何做到解耦，（基于 Maven 的项目模块管理）频繁的导入依赖可以做到各模块之间的互相调用，但有可能会造成循环依赖的问题，因此采用 eventbus 机制来处理模块间解耦问题。  4. 通过 HornetQ 消息队列随时保证企业用户和供应商用户文档数据的一致性问题。  5. 对于中国发票特有开票上限这一问题，如何针对数额较大的发票进行适当的拆分，以及拆分后如何保证相关数据的关联性。  6. 发票中经常会有折扣的概念，如何针对大数额的发票在后端进行精确的折扣计算，以减少不必要的计算误差。  本系统拥有很多特色   1. 本系统采用了Postgres+Elasticsearch 作为数据存储和搜索的存储方式。在全局文档搜索方面系统具有较高的性能。 2. 本系统作为基于浏览器端的系统，具有广泛的兼容性，可以适用多种主流操作系统。 3. 系统支持与企业自身 ERP 系统集成，可进一步减轻企业采购收货支付流程中的工作量。 4. 后端API 设计具有 Restful 风格特点，对前后端开发人员具有较好的友好度，而且具有一定的可扩展性，方便针对特定业务进行 API 接口上的扩展。 5. 服务器处理、响应速度快，且相当稳定。   **6、本人主要工作描述**  供应链电子发票系统是一个基于 UBL 商业文档的，企业用户和供应商用户共同操作的发票系统。整个流程中主要的操作是针对 UBL 商业文档的操作，因此设计具有中国风格的发票是本系统开发的前提。因此在该系统中，我的工作量主要是：（涉及收货单、模拟发票、消息队列三个模块）   * 负责设计收货单 UBL 文档、模拟发票 UBL 文档。 * 基于设计好的收货单、模拟发票 UBL文档类型，映射为 Java 类去处理相关的业务逻辑（相关文档的创建、修改、替换、删除等，以及针对文档数据的计算）。 * 在两种用户间的数据交换，需要用到 JMS+HonetQ 作为消息队列来时刻转发和监听各个文档的状态，因此基于 JMS+HonetQ 创建消息转发和消息监听也是我的一部分工作。 |

**四、工作进度的大致安排**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **应包括文献调研，工程设计，新工艺、新材料、新设备、新产品的研制和调试，实验操作，实验数据的分析处理，撰写论文等。**   |  |  |  | | --- | --- | --- | | 时间 | 工作 | 阶段成果 | | 2016.12 - 2017.2 | 完成对供应链电子发票系统的市场调研，分析项目的可行性。 | 可行性分析报告，市场调研，开题报告 | | 2017.2 - 2017.3 | 完成前期课题调研，研究分析相关技术，并掌握项目总体框架的设计 | 需求分析报告 | | 2017.3 - 2017.5 | 掌握项目各模块具体的原理以及自己相关工作的重点和难点，熟悉系统架构的每个细节 | 概要设计报告 | | 2017.5 - 2017.8 | 根据设计文档完成自己相关模块的工作的详细设计，部署好开发环境，完成代码框架的编写 | 详细设计报告 | | 2017.8 - 2017.9 | 完成项目编码与测试工作 | 项目代码 | | 2017.9 - 2017.10 | 整理相关文档，撰写毕业论文，进行毕业论文答辩 | 毕业论文初稿 | | 2017.10 - 2017.11 | 修改和完善毕业论文 | 毕业论文定稿 | |

|  |
| --- |
| **预期成果**   1. 完成基于 Restful 架构的供应链电子发票系统的设计与实现 2. 撰写工程硕士学位论文 |
| **导师意见**  **导师签名：**  **年 月 日** |
| **培养单位负责人意见**  **培养单位负责人签名：**  **年 月 日** |