

文章编号: 2096-1472(2016)-08-30-04

通用Web表单数据采集系统的设计与实现

程 佳, 陈 涛, 王 成

(武汉工商学院信息工程学院, 湖北 武汉 430065)

摘 要: 面对日常工作中种类繁多的表单数据采集的需要, 设计并实现了一个基于Web和数据库的通用表单数据采集系统。本文介绍了系统的基本处理流程, 详细分析了基于MVC经典模式及Struts2、Hibernate和Bootstrap作为主要框架的程序架构设计, 探讨了动态表单结构、数据存储与解析、Web数据交互、常用表单管理等关键技术的实现。所实现的系统能够解决日常工作中表单数据采集的问题, 灵活性好, 同时这对Web表单自动的生成也提供了一种较好的解决方案。

关键词: 表单; 数据采集; Web; 程序架构

中图分类号: TP319 **文献标识码:** A

Design and Implementation of General Web Form Data Collection System

CHENG Jia, CHEN Tao, WANG Cheng

(College of Information Engineering, Wuhan Technology and Business University, Wuhan 430065, China)

Abstract: In the face of the needs of a wide variety of form data collection in daily work, a general data collection system based on web and database is designed and implemented. This paper introduces the basic process of the system, analyzes in detail the program architecture design based on the MVC classic model and mainstream frameworks such as Hibernate, Struts2 and Bootstrap. The implementation of some key techniques are discussed, like data storage and analysis, Web data interaction and general form management. The system can solve the problems of form data collection in daily work, which provides good flexibility and a suitable solution to the automatic generation of web forms.

Keywords: form; data collection; Web; program architecture

1 引言(Introduction)

在日常工作中, 管理部门需要采集某些数据时, 往往设计各类表单, 经过层层下发、填报、汇总, 并对汇总数据的有效性和完整性进行审核, 如此繁琐的过程费时、费力, 比如高校统计教材征订信息等等。为简化工作流程和提高工作效率, 可以开发一个通用Web表单数据采集系统, 该系统能根据采集数据的需求来简单、灵活的设计相应表单, 用户通过Web填报数据即可。目前, 关于Web表单的自动生成方法有些少量的研究, 主要是基于模板和规则的, 具体利用XML或数据库技术, 相关技术在办公自动化系统或企业管理信息系统中有些应用, 但其实现过程较为复杂或专业人员才能胜任^[1-4]。另外, 专门的通用表单数据采集系统很少。因此, 设计并实现一个通用Web表单数据采集系统是非常有意义的, 这能给日常的数据采集工作带来很大的便捷性。

该通用表单数据采集系统的用户包括系统管理员、普通管理员和一般用户, 主要功能包括用户管理、表单任务设计、表单收藏、表单查看、数据填报、数据审核与导出。系统管理员能够管理用户基本信息。普通管理员根据采集数据的需要, 可视化的进行表单任务设计, 这包括表单任务名

称、备注说明、有效填报时间、表单字段名称、样式及字段约束等; 而且能对常用表单进行收藏, 避免以后再次使用时重复设计; 此外, 能够对用户提交的数据进行审核、统计与导出。一般用户可以通过Web查看所需完成的表单任务列表, 选择具体表单任务进行数据的填报、修改和提交工作。所有数据存储在数据库中, 查看历史的采集数据很方便。从系统整体架构角度讲, 系统基于B/S模式, 用户通过浏览器进行操作, 主要特点是操作非常简单、功能通用性强, 数据规范性好。

2 系统总体设计(Overall design of the system)

2.1 处理流程设计

该系统的主要功能是表单数据的采集, 涉及的主要工作流程: 首先由管理员定义表单任务并发布; 其次一般用户可以在系统中查看已发布的表单任务, 填写表单内容, 并提交表单; 最后由管理员审核与导出用户提交的表单数据。

(1) 管理员定义表单

当需要采集某些数据时, 管理员需要根据采集数据的需求来定义表单任务, 具体定义流程如图1所示。管理员首先创建表单任务, 接着设置基本信息, 包括表单任务名称、备注

信息和有效填报时间；其次配置表单字段，包括字段名称、输入形式、字段约束，其中设置字段约束时可以选择已定义的常用约束或定义一个新的约束并应用；最后，发布表单任务。

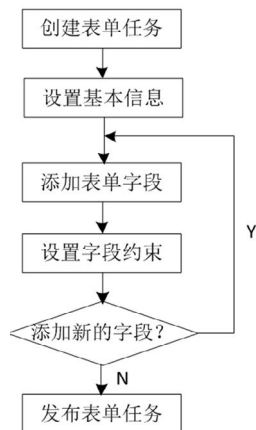


图1 表单定义流程图

Fig.1 Form definition flow chart

(2)用户填写表单数据

一般用户登录后可以看到所需填写的表单任务列表，具体填写流程如图2所示。首先选择具体的表单任务，则进入相应表单页面；其次填写表单内容对应的相关信息；最后提交表单数据，此时如果用户填写的内容与字段约束不匹配，系统会提示不匹配信息，需要用户进行数据修改，如果用户提交的表单数据符合对应的字段约束，才显示提交成功。

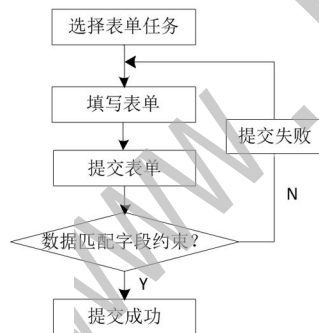


图2 表单填写流程图

Fig.2 Form filling flow chart

(3)管理员审核表单

用户提交表单数据后，普通管理员可以对用户提交的数据进一步审核，确保数据准确，此次审核主要是对数据实际意义审核。管理员可以查看用户提交的表单数据，并对表单进行审核处理，对合格的表单数据可以审核通过，对不符合要求的表单数据审核不通过。一般用户可以查看数据审核状态，若没有审核或审核不通过则可以修改表单数据再提交，若审核通过则不能再修改。

2.2 程序架构设计

系统基于MVC经典模式，使用Struts2、Hibernate和Bootstrap框架作为基本开发架构。程序构架模块如图3所示，MVC模式(Model-View-Controller)把系统分为三个基本部分：模型(Model)、视图(View)、控制器(Controller)。三个模块相对独立，降低了系统模块的耦合性，提高了程序的可维护性^[5,6]。

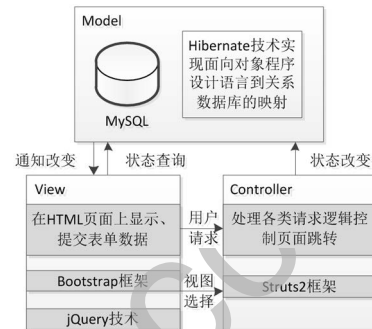


图3 程序构架设计

Fig.3 Program architecture design

Model层实现系统中的业务逻辑，主要为功能逻辑的具体编码。在系统中，Model层采用数据库持久化框架Hibernate，通过这一轻量级的O/R Mapping框架，完成用户对象、表单对象、字段对象、字段内容对象等到关系数据库的映射，完成对表单数据的提取与转换，提高了系统开发效率。

View层主要用于显示数据和提交数据，提供用户交互界面。比如，对于表单显示查询请求的结果，系统以Web方式展示给用户，并且使用Bootstrap框架中的分页组件美化了表单列表分页显示，通过jQuery操作页面元素以及结合AJAX达到异步刷新页面的效果。Bootstrap是目前很受欢迎的前端框架，它在jQuery的基础上进行了更为个性化和人性化的完善，形成一套自己独有的网站风格^[7]。

Controller层用于控制业务逻辑和页面的跳转，接受用户的输入并调用模型和视图去完成用户的需求。在Struts2框架下，根据需求编写Action类用于处理逻辑，编写JSP页面用于展示用户界面以及在struts.xml中配置映射关系。

例如，在struts.xml中关于表单显示请求的配置代码如下：

```

<action name="task_list" class="com.test.action.task.TaskListAction"
method="task_list">
<result name="success" type="json">
<param name="root">json</param>
</result>
</action>
  
```

其中, action标签用于定义处理请求URL为task_list. action的Action, result标签用于定义处理结果字符串和资源之间的映射关系, param用于对返回的结果进行配置, 其文本内容json表示返回的是json对象, 属性值root表示返回对象的层级为根部。

3 关键技术的实现(Implementation of key technology)

3.1 动态表单结构生成

由于每次所采集的数据很有可能是各式各样的, 因此不能事先设计固定的Web表单, 必须要根据具体情况设计动态表单结构。自定义一个表单主要涉及到表单基本信息、表单字段、字段约束等信息的配置。在Bootstrap、jQuery、HTML、CSS等前端技术的支持下, 为用户提供了向导式的人机交互界面, 使之适应于各类人群的使用, 比如在需要填写截止日期时, 系统会自动调用Bootstrap的日历插件。就系统实现而言, 系统采用MySQL数据库, 并进行数据表和相关字段的设置。在数据库的设计中, 对于表单的生成主要涉及表单任务、表单字段、字段约束这三个实体。关系模型如下:

表单任务(表单ID, 表单名称, 备注说明, 有效起始时间, 有效截止时间, 是否被收藏)

表单字段(表单字段ID, 字段名称, 所属表单ID, 字段约束ID)

字段约束(字段约束ID, 规则名称, 规则的正则表达式, 规则说明)

动态表单结构相关实体间的关系如图4所示。一个表单任务包含多个表单字段, 一个表单字段只属于一个表单任务; 一个字段约束可用于限定多个表单字段, 一个表单字段只能指定一个字段约束。



图4 表单结构相关实体

Fig.4 Entities related to form structure

例如, 在学校教学管理工作中, 每学期开学前学校需要向各科教师采集教材订阅信息。这时需创建一个表单用于采集教材订阅信息, 表单任务表中就需存储表单名称, 数据填写的备注说明以及有效填报时间等信息。对于表单字段的设置可存储在表单字段表中, 在该表单中有书籍ISBN、书籍名称、价格、出版社名称、使用班级、订购数量等字段。为了实现系统自动审核表单内容, 在表单字段约束的设计中, 主要利用正则表达式来规范对应内容的格式。正则表达式是

对字符串进行规范格式的一种逻辑表达式, 用事先定义好的一些特定字符及这些特定字符的组合, 组成一个“匹配字符串”, 这个“匹配字符串”用来表达对字符串的一种过滤逻辑。比如, 每本书会有个ISBN字段, 根据ISBN组成格式, 含有13位数字, 可以设计一个正则表达式: $[0-9]\{13\}$, 来进行合法性验证。字段约束表中存储有系统默认定义的一些常见的字段约束规则, 用户也可以新增。

3.2 数据存储与解析

对于数据的存储, 系统使用了另外两个数据表, 其中一个为表单概要表, 存储某个用户提交某个表单任务的概要信息, 另一个为表单详情表, 表单字段对应的详细信息保存在该表中。比如, 学校通过教材订阅信息表单采集教材需求信息时, 用户ID、表单ID、附件信息等存储在表单概要表中, 而表单中书籍ISBN、书籍名称、价格、出版社名称、使用班级、订购数量等信息存储在表单详情表中。关系模型如下:

表单概要(概要信息ID, 用户ID, 表单ID, 附件, 是否审核通过)

表单详情(详细信息ID, 概要信息ID, 字段ID, 字段值)

由于表单结构多样, 这使得数据的存储与解析流程的较为复杂, 系统设计了通用的处理程序来存储与解析表单数据。

3.3 Web数据交互实现

当表单创建完成, 用户(包括普通管理员或一般用户)再次请求表单任务列表时, 则从数据库中读取表单相关设置数据并传递到前端页面, 表单显示页面对表单结构数据进行解析、显示, 并且运用jQuery的分页插件实现表单列表分页显示。另外, 用户在请求某个具体的表单任务时, 采用Ajax+JSON技术与服务器交换数据。比如, 在教材订阅信息表单实例中, 当学校管理员或教师请求教材订阅信息表时, 通过用户界面发起AJAX异步刷新请求, 后台与服务器进行少量数据交换, AJAX可以使网页数据实现异步更新。这意味着不必采用会中断交互的完整页面刷新, 就可以动态的更新Web页面。

AJAX的工作原理相当于在用户和服务器之间加了一个中间层, 使用户操作与服务器工作异步化^[5-8]。AJAX的Web模型如图5所示。作为AJAX异步请求的传输对象, 后台通过Hibernate从数据库读取数据, 请求的返回结果是JSON对象。系统中大多通过AJAX技术与服务器交换数据实现网页部分刷新, 将对象转换为JSON的封装类使用了较为流行的JSON解析库Jackson, 类中通过单例模式来获取ObjectMapper对象, 从而序列化和反序列化JSON数据。用户界面获取到JSON数据后进行解析, 通过jQuery的eval方法将JSON数据转换成

JSON对象,然后像调用数组元素一样调用JSON对象的数据,以显示在页面列表中。

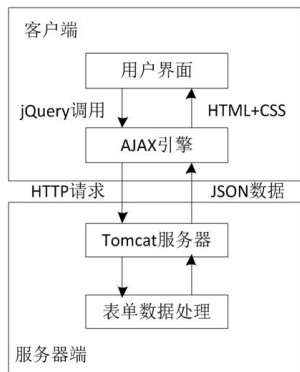


图5 AJAX的Web模型

Fig.5 Web model of AJAX technology

3.4 常用Web表单管理

根据用户的实际使用情况来说,有的表单模板可能会被多次重复使用。比如,在学校采集教材订阅信息的例子中,该表单在每个学期开学前都会用到,为提供更好的用户体验,系统设计了常用表单收藏的功能。此外,当系统中收藏的表单逐渐增多时,也可以通过关键词模糊搜索或分类标签类进行查找。

4 系统运行效果(Operation effect of the system)

服务器端系统的主要功能包括任务管理、知识库管理、用户管理三大模块。在任务管理模块,管理员可以新增表单,查看、搜索、修改及删除已定义的表单,以及配置表单字段,如图6所示。该界面上方和左方是功能模块栏目,右边是信息展示窗口,所显示的表单列表是按截止时间的先后、是否有提交数据以及是否审核通过排序,页面过渡、翻页、搜索等前端效果都是通过Bootstrap框架实现的。当新建表单任务完成后,可以配置相应的表单字段,字段配置界面如图7所示。该界面右边显示的是关于教材订阅信息表单的相关字段配置,有书籍ISBN、书籍名称、价格、出版社名称等,可以任意增加、删除、修改表单字段,根据要求添加相应的字段约束。



图6 表单任务管理界面

Fig.6 Form task management interface



图7 表单字段配置界面

Fig.7 Form field configuration interface

5 结论(Conclusion)

根据日常工作中的表单数据采集需求,实现了系统的基本功能,包括用户管理、表单任务设计、表单收藏、表单查看、数据填报、数据审核与导出等,系统的实现主要采用如今流行的Struts2+Hibernate+Bootstrap框架技术,扩展性较好。系统功能性较为实用、灵活性较好。而且,系统的用户可以是很大众化的,不要求具备程序设计方面的知识。但系统也还可以进一步改进。比如,提高个性化设置和通用性扩展,对于不同工作需求的用户,能自定义配置程序和系统参数。功能细节也可更加完善,使表单输入框具有足够的弹性,以接受多种有效数据形式。

参考文献(References)

- [1] Strme kiD,Rado evi D,Magdaleni I.Web Form Generators Design Model[C].Cecis,2015.
- [2] Magdaleni I,Rado evi D,Orehova kiT.Autogenerator: Generation and Execution of Programming Code on Demand[J].Expert Systems with Applications,2013,40(8): 2845-2857.
- [3] NegametzyanovA,Lau S L,Ng C F.Web-based Interactive Form Generator for Public Kiosks[C].IEEE Conference on Open Systems,2015.
- [4] 吴贺,及俊川,李新.基于XML的动态表单快速生成技术[J].计算机系统应用,2010,19(9):60-63.
- [5] 夏仙.基于SSI框架的通用表单开发工具的设计与实现[D].北京:北京邮电大学,2013.
- [6] 胡丽媛,黎杰.MVC模式及Struts框架的研究与应用[J].计算机与信息技术,2011(10):9-12.
- [7] 吴海.Bootstrap模板在后台管理系统应用中的经验探讨[J].信息通信,2015(2):101-103.
- [8] 郭庆燕,张敏,杨贤栋.JQuery Ajax异步处理JSON数据实现气象图片的显示[J].计算机应用与软件,2016,33(6):20-22;67.

作者简介:

程 佳(1994-),女,本科生.研究领域:软件工程.

陈 涛(1979-),男,硕士,讲师.研究领域:计算机网络.(本文通讯作者)

王 成(1993-),男,本科生.研究领域:软件工程.