

基于 JSP 和 Android 的歌手竞赛信息交互系统设计

黄磊¹, 杨武军¹, 张鹏^{1,2}, 程远征¹

(1. 西安邮电大学 通信与信息工程学院, 西安 710121; 2. 西安邮电大学 移动互联网研究所, 西安 710121)

摘要: 为了打造校园歌手选拔的线上“生态系统”, 完成线上与线下的互动闭环, 提出一种基于 Android 平台与 MVC 模式的歌手竞赛系统的方法; 该系统是客户端/服务器结构的, 使用 Servlet 和 JSP 技术, 并采用了改进的 MVC 设计模式, 将 Spring MVC 与 MyBatis 技术结合起来; 服务端利用 Tomcat 搭建 Web 容器, 用开发工具 Eclipse 对服务器编程, 并用 MySQL 数据库提供数据层应用; 该系统在移动终端上进行了测试, 可以实现线上报名, 上传作品, 互动分享, 查看排名, 以及对选手进行评论, 投票等功能; 该系统的实现可以使线上歌手竞赛随时随地的进行, 从而使参赛过程更加便利和高效。

关键词: 安卓平台; 歌手竞赛系统; MVC 模式

Design of Information Interaction System for Singer Competition Based on JSP and Android

Huang Lei¹, Yang Wujun¹, Zhang Peng^{1,2}, Cheng Yuanzheng¹

(1. School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710121, China; 2. Institute of Mobile Internet, Xi'an University of Posts and Telecommunications, Xi'an 710121, China)

Abstract: In order to build the online “ecosystem” of campus singer selection, and to complete the interaction between online and offline closed loop, a singer competition interaction system based on Android platform and MVC model has been proposed. This system was C/S structured and use Servlet as well as JSP technology, also adopted a modified MVC design pattern, combined Spring MVC and MyBatis technology together. The server included a Web container built by Tomcat, and used Eclipse development tools to program, meanwhile, it used MySQL database to provide data layer application. The system was tested on mobile terminal, can realize the online registration, display works, interactive sharing, checking the player rankings, as well as to the comment, vote and so on. The implementation of the system can make the online singer competition of anytime and anywhere, thus to make the process more convenient and efficient.

Keywords: Android platform; singer competition system; MVC model

0 引言

近年来, 随着移动互联网的迅猛发展, 手机上网已经成为生活中不可或缺的一部分, 以用户为中心的移动 APP 进入了一个快速发展的阶段。同时用户也越来越希望能在任何时间, 任意地点体验个性化服务。诸如中国好声音, 中国梦之声等选秀节目的兴起, 为有着音乐梦想的年轻人提供了一个展现自己, 实现梦想的舞台。

然而传统的线下海选对于深处象牙塔的大学生来说, 无论参赛时间还是比赛地点, 都有其局限性, 不能满足大学生群体张扬个性的需求, 线下参赛门槛仍然很高。因此本文基于 Android 和 JSP 技术^[1], 以 Android 移动设备作为客户端载体, 设计了一个歌手竞赛系统, 为校园音乐爱好者提供了一

个线上的展示自己的平台, 实现了选手线上比赛, 线下互动的功能。

1 Android 平台简介

随着信息化和互联网技术的发展, 智能手机越来越多地被广大消费者使用, 各类手机的操作系统应运而生。Android 系统从一发布, 就因为它开源的优势而迅速占领市场, 目前 Android 系统的市场份额已经达到了一半以上^[2]。技术的进步带动了整个 Android 软件产业的蓬勃发展, 越来越多的用户开始追捧日益丰富的 Android 软件应用。Android 采用开源的 Linux 操作系统, 分别采用 C 和 Java 语言技术实现底层和应用层^[3-4]。由于 Android 系统开放性这一特点, 使得在其上的软件开发相对较为容易, 普通用户可以根据自己的需求定向开发属于自己的应用软件, 同时由于应用的逐渐丰富, Android 手机用户的数量不断增长。考虑到 Android 系统庞大的用户群体和巨大的发展潜力, 所以选择它作为应用程序的运行环境。

2 系统架构设计

2.1 系统架构

该系统的设计研发, 提出了一种建立在 Android 平台上,

收稿日期: 2015-08-06; 修回日期: 2015-10-08。

基金项目: 2014 陕西省国际科技合作项目(2014KW02-02); 2014 陕西省工业攻关(2014K09-14)。

作者简介: 黄磊(1988-), 男, 河南安阳人, 研究生, 主要从事移动互联网技术方向的研究。

杨武军(1969-), 男, 陕西西安人, 副教授, 硕士研究生导师, 主要从事移动互联网方向的研究。

结合 Java Web 技术的方法。本系统采用 C/S 结构，为了使软件具有结构性，便于开发，维护和管理，本系统采用了 MVC 的设计模式，将不同的功能模块独立，使其具有松耦合性的特点，系统架构如图 1 所示。

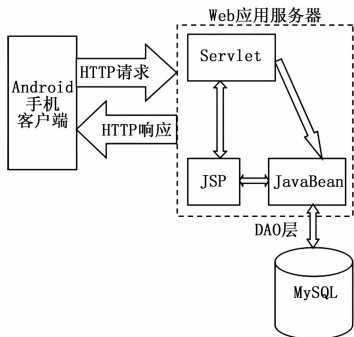


图 1 系统架构

Servlet 主要负责控制程序的流转，属于程序的控制层；JavaBean 负责封装和处理数据，与数据库进行交互；JSP 负责在页面展示数据。这种开发模式下的程序具有很好的结构性，因为各个层只做本身最擅长的工作。

图中 MVC 设计模式是一种基于模型—视图—控制器的软件设计模式，一种软件设计的规范。可以实现将视图与模型的代码进行分离，改变某一模块时不需要对其它模块进行改动，方便代码的替换与复用。M 指的是业务模型 Model，功能是数据库的存取数据；V 是指用户界面 View，处理的是数据显示的部分；C 则是控制器 Controller，负责从视图读取数据，并将后台数据传输到前台展示。

该系统客户端与服务端采用了 HTTP 协议进行交互，HTTP 协议是互联网上应用最为广泛的一种网络协议。由客户端发起一个 HTTP 请求，建立一个到服务器指定端口的 TCP 连接，服务器则在那个端口监听客户端发送过来的 HTTP 请求，一旦收到请求，服务器根据不同的情况响应一个状态码。在 HTTP 协议下，客户端每发起一次请求都需要服务器端进行一次响应。包括 POST 请求和 GET 请求，其中 POST 请求向服务器传送的数据对用户不可见；GET 请求是把参数放到 URL 中，对用户可见。

2.2 系统功能设计

服务端是基于 JSP 开发的系统，其功能如下：

1) 选手报名：填写报名信息，上传参赛歌曲或视频，后台将选手填写的姓名，手机等字段信息设置到 Model 层，保存到数据库的 player_info 表中；

2) 选手详细信息：选手详细信息页面展示的字段包括姓名，所在院校，得票数，评论数等。用户可以在线收听选手上传的音乐或视频作品，并可为自己喜欢的选手投票，同时可以在下面的讨论区对选手进行评论；

3) 选择赛区：页面采用三级联动技术，可选择省份、城市和学校。点击搜索后将地区条件加入到筛选条件中，重新跳转到排行榜页面，显示按照地区学校的选手排名列表信息；

4) 搜索选手：点击首页的“搜索”按钮跳转到选手搜索页面，后台的数据库查询规则采用的是模糊查询，将填写的姓名加入到查询语句中，通过 DAO 层的 id 映射到对应的查询语句，将结果返回并分页展示在前端页面；

5) 用户评论：用户可在线对选手的作品进行评论，后台根据用户的 permit 即手机号，从数据库中检索出当前用户的信息，并传递头像给页面并展示在评论列表中。当选手点击用户头像时，从前台返回一个“Username”参数给后台，后台重新拼接回复内容，实现选手对特定用户进行回复的功能；

6) 用户投票：每个用户每天可以为选手投一张票，点击详情页左下角的心形 ICON 后，就可以对选手进行投票，后台会根据注册用户的手机号进行判断，当前用户是否已经投过票，并将页面显示的投票数加一。系统功能模块如图 2 所示。

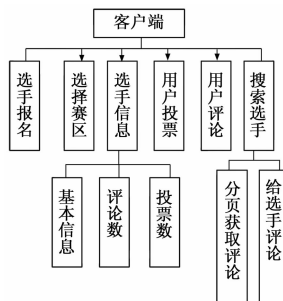


图 2 系统功能模块

2.3 开发工具与框架

该系统采用 Eclipse 对服务器端代码进行开发，Eclipse 集成了众多开发工具，有助于项目开发方便快速地进行。

本系统采用的 Spring MVC+MyBatis 是目前主流的框架，Spring MVC 配置较为简单，通常在项目中会涉及到几十上百的组件，如果采用 xml 的对象定义来配置的话，就会增加配置文件的大小，不方便查找和代码维护，项目中采用的 Spring 注解利用反射机制原理可以解决如上的问题。Spring 的组件自动扫描机制，可以实现在配置文件中添加几行配置文件代码，即可实现在类路径下寻找标注的功能。如我们可以用@Controller 标注控制层组件，@Repository 来标注数据访问组件，即 DAO 组件。

MyBatis 是一个自定义 SQL，存储过程和高级映射的持久层框架，它用于在 bean 和数据库之间传输数据，同时保证对象，数据库和数据映射层本身的独立。MyBatis 通过在 XML 中进行配置和定义 SQL 语句，用 Mapper 来绑定 XML 与程序之间的关系，这样便将数据库语句和对象隔离了起来。

本系统根据用户竞赛交互系统的实际需要，保证平台的稳定运行，选择使用 MySQL 数据库。MySQL 是目前主流的关系型数据库管理系统，具有体积小、速度快、总体成本低、安全可靠等特点，适用于各种平台。同时采用数据库可视化软件 SQLyog 对数据库进行可视化，方便地操作。

3 服务端的设计与实现

3.1 服务端程序构成

本系统创建的 Web 服务器是采用 Spring MVC 架构，在于客户端交互上，为了使得数据传输效率更高，系统采用了 JSON 格式的数据用于前后台的数据传输^[5-7]。JSON 是一种轻量级的字符串数据格式，采用的文本格式易于阅读和编写，同时也易于机器解析和生成，不仅可以传输一个数组，还可以传

输一个对象。

为了完成整个服务端的功能, 服务器端程序一共有 7 个包, 如图 3 所示。

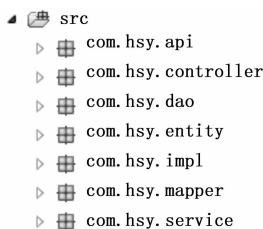


图 3 服务端包

DAO 包主要是一些接口, 封装了数据库的增删改查的操作, Service 包主要用来调用 DAO 层的一些方法, 同时可以操作多个 DAO 包, ServiceImp 是对 Service 接口的具体实现, 这里会调用 DAO 的一些方法, Controller 层负责控制程序的流转, 通过调用 Service 中的方法做一些操作, 可以将查询到的数据传入前台进行展示。

3.2 数据库的建立与连接

根据功能需要, 数据库包含 3 张表: 选手详情信息表 player_info, 选手评论信息表 player_comment 及投票信息表 player_vote 等, 采用主流的 MyBatis 持久层框架, 当 DAO 层接收到调用请求时, 会接收到 SQL 的 ID 和传入对象, MyBatis 会根据 SQL 的 ID 找到对应的 MapperStatement, 然后根据传入参数对象对其进行解析, 最终得到要执行的 SQL 语句和参数。

选手信息表包含选手 ID 主键、选手姓名、选手联系方式、性别、报名头像地址、报名时间和所在学校 ID, 如表 1 所示; 选手评论信息表包括用户 ID、所评论的选手 ID、评论内容、评论类型及评论时间, 如表 2 所示; 投票信息表包括投票用户 ID、所投票选手 ID、投票数目及投票时间等, 如表 3 所示。

表 1 选手详情信息表

列名	数据类型	主键	备注
PlayerID	Int	是	主键 ID
PlayerName	Varchar(50)	否	选手姓名
Permit	Varchar(50)	否	联系方式
Sex	Varchar(50)	否	性别
PhotoUrl	Varchar(50)	否	头像地址
CreateDate	DateTime	否	报名时间
CampusID	Int	否	学校 ID

表 2 选手评论信息表

列名	数据类型	主键	备注
UserID	Int	是	主键 ID
PlayerID	Int	否	所评论的选手 ID
Comment	Varchar(100)	否	评论内容
CommentType	Int	否	评论类型
CommentDate	DateTime	否	时间

表 3 投票信息表

列名	数据类型	主键	备注
UserID	Int	是	主键 ID
PlayerID	Int	否	所投票的选手 ID
Count	Int	否	投票数目
CommentDate	DateTime	否	投票时间

在这个过程中, Controller 层不再直接访问数据库, 而将其转化为由 DAO 封装对数据库的增删改查的操作。这样就将低级别的数据访问逻辑与高级别的业务逻辑分离了, 体现了松耦合的思想。

4 实验与分析

各个模块设计开发完毕后, 为了使用户有更好的应用体验, 需要将代码部署到 linux 服务器上, 本次开发用的 linux 远程管理器是 Xshell+Xftp, Xshell 是一个强大的安全终端模拟软件, 它支持 SSH 以及 Microsoft Windows 平台的 TELNET 协议。Xftp 是一个基于 MS windows 平台的功能强大的 SFTP, FTP 文件传输软件。

项目开发完毕后, 部署到正式服务器上, 将打好的项目的 war 包通过 Xftp 传输到服务器上。

接着打开 Xshell, 输入 linux 命令: sudo cp -a /home/remote/isp.war./ 将 isp 的 war 包复制到当前文件夹下, sudo unzip isp.war 解压缩, 接着 sudo /ect/tomcat7 restart 重启 tomcat 服务器, 这样项目就部署在了服务器上, 就可以通过无线网络访问系统服务器, 同时也可以通过访问后台来获取数据信息。

实现结果如图 4 所示, 其中系统主界面如图 4 (a) 所示, 排行榜界面如图 4 (b) 所示, 选手详情界面如图 4 (c) 所示。



(a) 主界面 (b) 排行榜界面 (c) 详情界面
图 4 实现结果

5 结论

本文提出了基于 JSP 和 Android^[8-9] 的歌手竞赛系统, 客户端与服务端的交互采用 HTTP 协议, 上传视频文件时将字符串转化为字节流传输到后台, 播放时再读取文件流并还原成视频文件。该系统部署在了 Linux 远程服务器上, 经过测试, 能够满足多用户同时注册报名登录浏览的需求。

本文的特色之一是移动 APP 平台的创新运用, 从使用的便携性上来说, 用户通过 APP 即可利用碎片时间参加线上竞技, 同时也可以线下邀请好友注册为自己拉票助威。采用的线上竞技线下互动的 O2O 模式, 掌上突围, “云”端加冕, 线上线下齐互动, 移动互联网平台的运作使之变得更加高效和便捷。