第19章 ： 电子拍卖系统

**本章要点**

* Android应用与传统应用整合的意义
* Android应用整合传统应用的方式
* 基于HttpClient，Json数据交换的整合方式
* JSON基本知识
* JSON语法
* 开发服务器端生成JSON响应的Servlet
* 开发Android客户端界面
* 使用HttpClient发送请求
* 使用HttpClient获取服务器响应
* 将服务器响应转换成JSON对象或数组
* 通过Android客户端加载服务器响应

本章介绍了一个实用的Android应用：电子拍卖系统。它是一个Android + Struts + Spring + Hibernate 整合应用。Struts + Spring + Hibernate 提供一个B/S结构电子拍卖系统，对于使用PC用户而言，可以使用浏览器来访问该系统；对于使用Android手机的用户而言，可以选择安装Android客户端程序这样即时通过手机来使用该拍卖系统。

Android客户端通过网络与服务器端的控制器组件交互。

Android系统发展的大抛是与传统服务器应用系统整合，因为手机的硬件资源毕竟是有限的：计算能力有限，存储能力也有限。所以Android应用更适合作为应用的客户端：手机携带方便，可以随时开机运行应用，而且可以随时访问网络，并通过网络与服务器端应用交互，获取服务器端的数据。在未来的几年内，运行在手机上的电子商务客户端（目前已经有如淘宝已经有了手机客户端）、证券系统客户端，金融系统客户端将会大量出现。

本章的介绍重点是Android开发，因此将会详细介绍Android应用的开发，心及如何让Android应用与远程服务器交互；

# 1 系统功能简介和架构设计

## 1.1 系统功能简介

本章所介绍的系统是一个功能不太复杂的电子拍卖系统，本系统从实际电子商务平台上抽取，只取出其中部分核心功能实现，以作为示范应用，展示一种良好的程序架构。

电子拍卖系统其实就是一个电子商务平台，只要将该系统部署在互联网上，全球的客户都可以在该系统上发布想售出的商品，也可以对拍卖中的商品参与竞价。

如果系统中提供与电子银行的接口，将可以通过电子银行的操作，实现买家对专家的自动付款。一早付款成功，就可以利用全球物流供应系统将拍卖物品发送到买家手中。

该电子拍卖系统模拟了淘宝系统部分功能，抽取了实际电子拍卖系统部分功能，但没有提供如个人身份谁，信用管理等细节问题，本系统主要实现了电子拍卖系统中的核心功能。

本项目的服务器端是一个完整的Java EE项目，服务器采用控制器层，业务逻辑层，DAO层的分层架构。Android客户端应用只负责与服务器的控制器组件交互，Android应用采用Apache HttpClient向服务器端的控制器发送请求并获取服务器响应，这样即可实现Android系统与电子拍卖系统之间的通信。

本系统要求用户参与拍卖之前，必须登录系统。登录验证在服务器端通过Filter实现，Filter拦截用户请求，并判断Session中是否保存了当前用户ID，如果保存了用户ID，即该用户已经登录，否则没有登录。

对于物品的管理，本系统可以查询拍卖物品，添加拍卖物品，增加物品各类，竞价处理，以及发送邮件通知用户所参与的竞价。

* 注册用户可以添加用户物品，添加物品各类。添加前必须登录系统
* 注册用户可以浏览当前拍卖中的物品，以及流拍的物品
* 注册用户可以参与竞价，参与的竞价系统将通过邮件通知用户。

## 1.2 系统架构设计

本系统的服务器采用JavaEE的分层结构，分为视图层，控制器层，业务逻辑层和DAO层。分层体系将业务规则，数据访问等工作放到中间层处理，客户端不直接与数据库交互，而是通过控制器与中间层建立连接，再中中间层与数据库交互。

中间层采用SpringMVC ＋　Hibernate,为了分离挖掘层与业务逻辑层，又可细分为

* 控制器层，就是MVC中的C，负责表现层与业务逻辑层的交互，调用业务逻辑层，并将业务数据返回给表现层来显示
* Service 层（业务逻辑层），负责实现业务逻辑，对DAO对象进行正面模式的封闭
* DAO层 （数据访问对象层），负责与持久化对象交互，封闭了数据的add,delete,select,update原子操作
* PO层(持久化对象层) 通过实体/关系映射工具将关系型数据库的数据映射成对象，实现以面向对象方式操作数据库，这个系统采用Hibernate

数据库系统使用MySql。

服务器响应将会采用JSON数据格式——可以有