目 录

[第1章 电力交易动态推荐系统的设计与实现 1](#_Toc500419327)

[1.1 需求分析(未完成) 1](#_Toc500419328)

[1.1.1 功能性需求分析 1](#_Toc500419329)

[1.1.2 非功能需求分析 1](#_Toc500419330)

[1.2 总体设计(未完成) 1](#_Toc500419331)

[1.2.1 系统架构设计 1](#_Toc500419333)

[1.2.2 用户体验设计原则 1](#_Toc500419334)

[1.3 详细设计 1](#_Toc500419335)

[1.3.1 移动端功能模块设计 1](#_Toc500419337)

[1.3.2 数据库设计 1](#_Toc500419338)

[1.3.3 移动端业务逻辑类设计\* 1](#_Toc500419339)

[1.4 系统展现 2](#_Toc500419340)

[1.5 本章小结 2](#_Toc500419345)

# 电力交易推荐系统移动端设计与实现

如今移动互联网技术更迭日新月异，移动智能设备远远超过个人微型计算机的数量。大多数互联网产品都扩展了面向移动互联网的产品线。随着智能设备的普及，主流移动操作系统呈现竞争趋势，目前市场占有量最大的是Android操作系统。本章根据相关电力改革背景，设计并实现基于Android平台下的电力交易推荐系统原型，遵循软件工程的思路阐述电力交易推荐系统移动端原型的设计和实现过程。

## 需求分析(未完成)

软件需求分析是软件设计至关重要的阶段。该部分是软件开发的方向性的指导方针。主要任务是对软件的需要和要求进行归纳和整理。本节将分析电力交易推荐系统的功能需求。

使用本系统的角色包含三类：大用户、发电企业、电网企业。其中，电力交易推荐系统移动端的使用角色仅包含大用户。大用户可以在移动端浏览查询交易信息，接收推荐信息，管理合同信息，以及进行个人用户信息管理等。因此移动端需求可通过以下用例图【图】来描述：



1）交易管理：含有交易公告、交易申报、交易结果等主要功能需求。主要实现当前用户参与的包括交易公告详细内容、申报状态、交易发布时间、申报截止时间、交易附件；申报数据结果，根据不同交易类型展示数据申报结果，主要内容包括电价、电量、时间段等信息以及包括交易类型、交易名称、交易形式、成交电量、成交均价等交易结果。

2）兴趣管理：包含当前用户感兴趣的售方展示的功能需求。主要实现根据用户的行为进行推荐，为用户推送感兴趣售方供用户选择。

3）合同管理：含有当前合同、历史合同、合同分析等功能需求。主要实现对当前用户的合同信息查询，并且提供当前合同的执行追踪情况及统计分析。

4）基本信息管理：含有市场成员信息、机组信息、用电单元信息、联系人信息、准入用户信息、市场成员历史信息、机组历史信息、用电单元历史信息以及统计信息等功能，实现了对当前用户的信息全生命周期统计查询。

5）软件设置：含有密码修改、客户端缓存清除等需求。主要实现密码修改功能，清除客户端缓存功能。

## 总体设计(未完成)

总体设计是软件开发过程中的一个设计性的工作，主要包含对系统架构的设计。本节介绍电力交易推荐系统的系统架构设计情况。



### 服务端架构设计

服务端的架构总体包含三个层次：数据层、业务层和视图层。如图【图】所示。



视图层是与客户之间直接发生交互的移动端程序，数据层用来存储用户行为的数据，业务层是负责视图层和数据层发生交互的中间程序，提供业务支持，包括对用户行为数据的采集系统，即日志系统，还包括本系统的核心服务，即推荐系统。三者之间发生交互，相互合作，共同为用户提供推荐服务。

推荐服务主体有推荐系统产生，由用户行为驱动。当用户产生行为数据，首先由日志系统对这部分数据进行收集，本部分应该采用流式数据处理引擎做日志处理（如Flume）。收集后的用户行为日志数据会存储到数据层，使用HDFS提供的分布式文件存储服务完成。由于本系统的用户实时性不那么明显，没有实时推荐需求，因而对获得的日志文件通过推荐系统的离线分析来生成推荐结果。生成的推荐列表由服务层发送到视图层，即用户的Android智能设备上。

### 移动端架构设计

在移动端整体开发遵循MVC架构模式。移动端架构设计如图【图】。



视图层(View)是对用户显示数据的一个展示层，主要负责显示界面，包括显示用户的注册，登录等操作功能界面。针对本系统的移动端视图，在res/layout目录下完成界面的布局，在android.view包下完成界面的组合封装。视图层的开发使用Xml语言和Java语言共同完成。

控制层(Controller)是调用模型层层的相关代码来实现客户端的业务逻辑控制。用于捕获用户请求并控制请求转发。在移动端的架构中，控制层发送HTTP协议请求到服务器端获取相应数据，并根据模型层的回调把数据响应到视图层，用户每次对视图层的触发会产生一个Action给控制层，控制层调用模型层代码实现业务逻辑控制。本系统客户端程序使用android的Activity和Fragment共同完成控制层的编写。

模型层(Model)主要是负责业务逻辑以及数据库的交互。在客户端会有一个本地数据库，在Android系统中，采用的是SQLite，一个基于MySQL的关系型数据库。使用本地数据库可以提高系统的响应速度，存储本地系统的数据模型。服务端每次的响应会发送出JSON数据，模型层捕获JSON数据加以封装，存储于本地数据库SQLite中，为控制层提供数据模型。

### 网络架构设计

电力交易移动应用同时提供wifi和移动网络两种接入方式，参与市场交易的市场成员用户采用VPN方式接入，可以保证数据传输安全，公众大用户不需要使用VPN通道，直接接入外网代理服务器。外网部署日志服务器和代理服务器，对外统一提供代理服务，内网部署移动应用服务器提供电力交易应用服务，内外网隔离方式有效保证内网数据的安全。内网移动应用服务器与电力交易系统相连接，实时交互业务数据。网络架构示意图如图【图】所示：



## 详细设计

本节详细介绍移动端原型的开发流程，包括功能模块设计、数据库的逻辑模型和物理模型设计等。



### 移动端功能模块设计

由系统的需求分析可以获取到五大根本的模块：交易管理、兴趣管理、合同管理、基本信息管理和软件设置。在这些模块中，可以划分出若干个子功能。移动端的功能模块设计如图【图】。其中，最核心的推荐功能部署在兴趣管理模块，该模块主要负责对推荐结果的展示功能。



### 数据库逻辑模型设计

### 数据库物理模型设计

## 系统展现



## 本章小结

# 