

基于蓝牙技术的手机投票系统

李琳, 郑晓静

(洛阳理工学院, 河南 洛阳 471023)

摘要: 该文描述了一种基于蓝牙技术的手机投票系统设计方案。整个投票系统分为服务器端和客户端并利用蓝牙进行数据传输。手机作为客户端用于发送投票信息; 计算机作为服务器端用于接收客户端发送的投票结果。该文详细介绍了整个系统的设计结构及实现方法, 以 NetBeans IDE 作为 JAVA 开发环境。该投票系统的测试平台是 Compaq nx8220 计算机, Sony-Ericsson W810i 手机和 Nokia 6630 手机。

关键词: 蓝牙; J2ME 程序; 用户界面设计; 服务器端; 客户端

中图分类号: TP311 **文献标识码:** A **文章编号:** 1009-3044(2013)01-0056-03

Mobile Voting System Based on Bluetooth

LI Lin, ZHENG Xiao-jing

(Luoyang Institute of Science and Technology, Luoyang 471000, China)

Abstract: This report describes how to develop a mobile voting system based on Bluetooth technology. The voting system runs on PC platform, which is treated as server; and cellular phones, which are treated as client. The work has been focused mainly on Bluetooth protocol used in exchanging data between server and client. During the development, Netbeans IDE with mobility pack is used to create J2ME application. Sun wireless toolkit 2.0 is the best option as build tools and device emulator. A detailed explanation on architecture of this project is discussed including the system structure, communication handler, question syntax, etc. The demonstration platform for this voting system was Compaq nx8220 laptop, Sony-Ericsson W810i and Nokia 6630 mobile device.

Key words: bluetooth; J2ME application; user interface design; server; client

为了解决在很多小型场所投票方式落后的问题, 笔者设计出了一种基于蓝牙技术的手机投票系统。在衡量了各种技术的优缺点后, 投票系统采用蓝牙和 J2ME 作为系统的技术支持。

1 投票系统的结构设计

1.1 系统的功能分析

根据投票系统的实际应用, 可以将这个系统细化为以下两部分:

表 1 系统功能表

服务器端	客户端
通过蓝牙发布服务并且创建服务记录	寻找蓝牙设备和服务
等待客户端的服务请求	在发现服务后建立服务连接并且发送数据
更新服务记录	
读取来自于客户端的数据并且反馈结果给客户端	

1.2 系统的整体框架

根据上述部分的功能分析, 进而可以得出整个系统的结构框架。

- 1) 设计一个基于蓝牙技术的程序运行于服务器端(PC机)用于“侦听”与之相连接的设备的投票情况并将投票结果显示出来。
- 2) 设计一个运行于手机客户端的 J2ME 程序用于发送投票结果。

2 投票系统的实现

整个投票系统的设计可分为以下几个模块: 用户界面设计、服务器模块、用户端模块、建立连接、发送信息收集回馈信息。

收稿日期: 2012-11-18

作者简介: 李琳(1981-), 女, 四川德阳人, 助教, 硕士, 主要研究方向为移动计算; 郑晓静(1962-), 女, 河南洛阳人, 助理实验师, 本科, 主要研究方向为计算机应用。

2.1 用户界面设计

整个系统的界面由两部分组成:一个是计算机上的服务器端界面;另一个是手机终端的用户端界面。

1)服务器端界面功能分为四部分:

- 1)启动服务应用 (“Select” 按钮)
- 2)显示系统状态 (文本框内显示)
- 3)用柱状图表示投票结果 (“View Result” 按钮)
- 4)退出应用 (“Exit” 按钮)

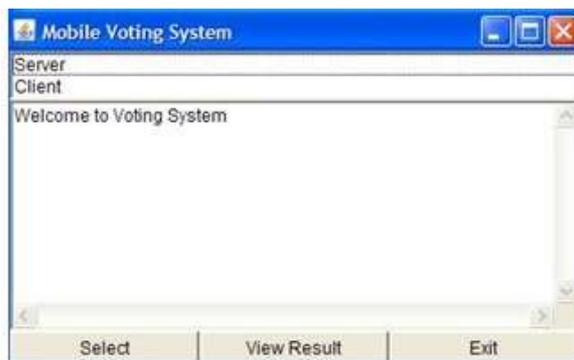


图1 服务器端初始界面

2)用户端界面功能分为三部分:

- 1)内容显示:该投票系统要求为每一个提问提供回答选项,例如:非常好、基本不错等。
- 2)导航系统:提供按钮进入下一界面
- 3)帮助文件:阻止用户操作错误,例如当用户输入错误时能有帮助信息及时提醒。

2.2 服务器模块设计

服务器模块主要发布服务给蓝牙设备并且允许其它用户端连接此服务。另外服务器需要组织数据的输入流,输出流并且根据从客户端反馈的数据显示结果,这个过程可以分为三步:

1)指定 UUID 值:蓝牙设备上指定服务的唯一值。不同的服务有不同的 UUID 值。该投票系统中 RFCOMM 协议被用于数据的通信,所以在该设计中 UUID 的值被指定为 0x0003。

2)设置蓝牙协议:投票系统采用的蓝牙协议是 RFCOMM 协议,用于发送和接受数据流,它以如下形式定义:public static final UUID RFCOMM_UUID = new UUID (0x0003);

3)注册服务:在客户端使用服务器端发布的服务前,服务器必须首先注册服务,其关键代码如下:

```
//创建 localDevice 对象并得到信息
localDevice = LocalDevice.getLocalDevice ();
//设置 localDevice 为可被发现模式
localDevice.setDiscoverable (DiscoveryAgent.GIAC);
//创建一个指定 URL 地址的服务
notifier = (StreamConnectionNotifier) Connector.open (url);
```

2.3 客户端模块设计

当服务器端运行了服务程序后,服务已经被注册了。对于客户端来说,它的任务就是发现蓝牙设备和服务。DiscoveryListener 接口用于实现以上功能,这里有四个方法被应用于此接口:

- 1)void deviceDiscovered (RemoteDevice btDevice, DeviceClass cod)
- 2)void servicesDiscovered (int transID, ServiceRecord [] servRecord)
- 3)void serviceSearchCompleted (int transID, int respCode)
- 4)void inquiryCompleted (int discType)

2.4 建立连接

在建立连接这部分主要考虑如何在服务器和用户之间建立连接、如何发送数据流、如何接收数据流、如何拆除连接等问题。具体划分到服务器端应该实现的功能是等待用户初始信息、发送问题给用户、接收来自于用户端的信息、服务器关闭连接;而用户端应实现的功能是打开连接、发送初始信息流、接收来自于服务器的问题、发送回复信息给服务器、客户端关闭连接。

2.5 发送数据

在发送数据环节所应该考虑的发送的问题的结构是如何的?问题是采取一个一个发送还是把所有问题封装在一个数据流中一起发送。该系统中采用的是第二种方案即服务器将所有问题通过一个数据流发给所有用户端手机,数据显示在手机屏幕上再由

用户点击“Next”按钮自行控制阅读下一个投票问题。之所以采用此方案是因为每个人的阅读速度有差异,如果采用第一种方案,控制权交由服务器端就会因此而引起许多不必要的麻烦。作为服务器来说,发送数据的关键代码如下:

```
//将投票的问题设置在 msg1 中
String msg1 = "Q1: How satisfied are you with our service in the canteen? |Q2: How do you like Chinese food?| Q3: How satisfied are you with facility in the LRC?| Q4: How do you think the quantity of books in the LRC?";
//创建输出流并且发送到用户端
OutputStream outputStream = conn.openOutputStream ();
outputStream.write (msg1.length ());
outputStream.write (msg1.getBytes ());
//关闭数据流
outputStream.close ();
```

2.6 整理结果

服务器利用数组【】来存放所有接收到的各用户端回馈信息。例如来自于用户1的答案 ABCD 存放于 store[0],A 是用户1收到的第1个问题的回馈答案,B 是用户1接收到的第2个问题的答案,以此类推。当用户点击服务器上的“View Result”按钮,所有答案会被手机重新整理,关键代码如下:

```
else if ( command.equals ("View Result"))
{
// 建立一个新的 GraphDisplay 对象
GraphDisplay gd = new GraphDisplay ();
//收集答案
For (int n=0; n< echoServer.i; n++)
{String s=echoServer.store [n];
gd.collectResult (s);
}
}
```

3 结论

该系统满足了多数场合的投票功能需求。与其它设计方案相比,成本低,更快捷、易实现。通过测评,上述设计方案实现了利用移动终端进行投票的功能。实验结果如下:

表2 测试结果(服务器:Compaq 计算机;用户端:Nokia 6630 和 Sony-Ericsson w810i 手机)

	接收来自于服务器的信息	接收来自于用户端的信息	平均回路延迟
测试 1	用户 1: 17s	用户 1:14s	32s
	用户 2: 18s	用户 2:16s	
测试 2	用户 1: 12s	用户 1:13s	28s
	用户 2: 15s	用户 2: 15s	

参考文献:

- [1] Bruce Hopkins,Ranjith Antony. Bluetooth for Java[M]. New York:Apress, Inc, 2003.
- [2] C Bala Kumar, Paul Kline, Tim Thompson. Bluetooth Application Programming with the Java APIs[M].Morgan Kaufmann,2003.
- [3] Sing Li,Jonathan Knudsen. Beginning J2ME: From Novice to Professional, Third Edition[M]. Apress, Inc,2005.
- [4] Qusay H Mahmoud. Wireless Application Programming with J2ME and Bluetooth[J]. Sun J2EE white papers,2003.
- [5] 郭克华. Java ME 移动开发实例精讲[M].北京:清华大学出版社,2010.