



江 蘇 大 學

JIANGSU UNIVERSITY

本 科 毕 业 论 文

## 基于 Android 的 PPT 控制软件的设计与实现

Design and Implementation of PPT Control Software

Based on Android

学院名称: 计算机科学与通信工程学院

专业班级: 物联网 1201 班

学生姓名: 周 率

指导教师姓名: 王 新 宇

指导教师职称: 副 教 授

2016 年 5 月

# 基于 Android 的 PPT 控制软件的设计与实现

专业班级：物联网 1201 班

学生姓名：周率

指导教师：王新宇

职称：副教授

**摘要** 在智能手机迅猛发展的今天，手机软件已经成为人们日常生活中不可少的一部分。把手机软件运用在教学上的例子有很多，但在实现 PPT 控制方面的软件就很少，并且免费且高效便捷的软件少之又少。为解决 PPT 演示时可能出现的一些问题，包括限制演讲者的操作自由以及 PPT 翻页笔不可控的因素，设计一款免费的 PPT 控制助手工具软件是十分需要的。

我们采用 C/S 架构，在 eclipse 开发环境下使用 Java 语言开发了服务端，在 Android studio 开发环境下开发了 Android 客户端。客户端与服务端之间使用 socket 进行通信。客户端的设计严格按照 MVC 架构，采用模块化设计的编程思想。服务端的核心技术由 Java Robot 类提供。我们系统实现的功能有：对 PPT 文稿进行控制操作，包括前一页、后一页、开始播放、退出播放、标记模式。此外还提供了模拟鼠标操作的功能。

在设计系统时，我们充分考虑了实际使用情况，对出现的问题有针对性的进行了开发，这使得我们的课题非常具有实用价值。此外，我们的产品在 PPT 控制方面具有功能完备，操作简单，界面友好等特点。

**关键词：**C/S 架构    Java 语言    Android 开发    Socket 通信    PPT 远程遥控

## **Design and Implementation of PPT Control Software Based on Android**

**Abstract** Nowadays, due to the rapid development of the smart phones, software of mobile phone has become one part of people's daily life. Many examples that software have been used in teaching can be found but there is fewer in the aspect of manipulating PPT especially the free, efficient and practical software. In order to solve some problems may occur in PPT presentation including restrictions on freedom of the operation to the speaker and some uncontrollable factors may occur in the pen of page control. It's necessary for us to design a free software of PPT control assistant tool.

C/S architecture is used and Java language is used to develop the Server based on the eclipse development environment as well as the Android studio development environment is used to develop the Android client. The communication between Client and Server is achieved by Socket technology. The design of Client is strictly followed the rules of MVC architecture and the program idea of modularization design is used. The core technology of Server is provided by the class of Java Robot. The functions of the system include PPT manipulations such as page up, page down, start playing, stop playing and tag mode and mouse simulations.

When designing this system, we take into full consideration of the actual situation. For the problems we take the special development which makes our scheme very practical value. In addition, our product have many characters in the control of PPT including full-featured, easy to operate and friendly interface and so on.

**Key words** C/S architecture Java Android Socket PPT remote control

# 目 录

<b>第一章 概述</b>	<b>1</b>
1.1 课题研究的背景	1
1.2 课题的研究目的和意义	1
1.3 研究现状	2
1.4 论文结构	2
<b>第二章 系统相关技术介绍</b>	<b>4</b>
2.1 软件结构	4
2.1.1 C/S 软件结构及工作模式	4
2.1.2 C/S 软件结构的优点	4
2.2 基于安卓的客户端开发环境	5
2.2.1 安卓操作系统	5
2.2.2 安卓的平台优势	5
2.3 Java 开发语言	5
2.3.1 Java 简介	5
2.3.2 Java 的主要特性	6
2.3.3 java.awt.robot 类	6
2.4 Socket 通信	7
2.4.1 Socket 的概念	7
2.4.2 Socket 的基本原理	7
2.5 开发工具	7
<b>第三章 系统分析</b>	<b>9</b>
3.1 系统需求分析	9
3.2 系统可行性分析	9
3.2.1 技术可行性分析	9
3.2.2 经济可行性分析	10
3.2.3 法律可行性分析	10

3.2.4 环境可行性分析.....	10
3.2.5 操作可行性分析.....	10
<b>第四章 系统设计 .....</b>	<b>11</b>
4.1 系统的结构设计 .....	11
4.2 系统模块设计 .....	12
4.2.1 服务器端设计.....	12
4.2.2 客户端设计.....	12
<b>第五章 系统实现 .....</b>	<b>16</b>
5.1 服务端实现 .....	16
5.2 客户端实现 .....	18
5.2.1 连接模块.....	18
5.2.2 功能选择模块.....	20
5.2.3 PPT 控制模块 .....	21
5.2.4 标记模块.....	23
5.2.5 鼠标控制模块.....	24
5.2.6 断开连接模块.....	28
5.3 服务端数据处理 .....	29
<b>第六章 总结和展望 .....</b>	<b>31</b>
6.1 总结 .....	31
6.2 展望 .....	31
<b>致谢 .....</b>	<b>33</b>
<b>参考文献 .....</b>	<b>34</b>

## 第一章 概述

近年来，由于移动通信技术的大幅进步，促使了移动平台的飞速发展。为满足用户需求，各移动设备厂商纷纷大力发展自己移动操作系统。其中典型的有诺基亚公司的塞班系统，微软公司基于 Windows 开发的手机操作系统，谷歌公司的安卓系统和黑莓公司的黑莓操作系统。这些移动端的操作系统中，只有安卓是开源的操作系统。安卓操作系统最先是使用在手机上，随后，安卓操作系统逐渐发展到其他的领域上，比如平板电脑，电视，数码相机等设备。据统计，从 2011 年开始，安卓系统在全球的智能手机市场脱颖而出，其销售额和用户量都跃居为全球第一。到 2013 年，基于安卓系统的手机的市场份额已达到 78.1%。迄今为止，全世界采用这款操作系统的设备数量已突破 10 亿。正是由于安卓操作系统的兴起，导致老牌的移动厂商走向没落，随着诺基亚和摩托罗拉商业帝国的坍塌，标志着我们已经走向了智能设备的时代。因此，基于安卓手机操作系统的开发具有广阔的前景。

### 1.1 课题研究的背景

随着科技的快速发展，特别是嵌入式硬件方面的技术的提升，电子设备给以往人们的生活方式带来了很大的冲击，这种冲击是有利于人类生活发展的，能为人们的生活提供各种便利。其中，手持终端智能设备对人们生活的影响尤为突出，智能终端设备以其高技术含金量，多样的用途场景，丰富的市场应用以及较合适的市场价格广受当代人们的青睐。据调查，我国智能手机的普及率已达到 76%且覆盖用户年龄范围广，受欢迎程度高，因此，智能手机在我国有巨大的用户规模。在智能手机市场，采用安卓操作系统的智能手机以其便捷的操作系统，高性价比的特性，占据着大半的市场份额。由此可见，安卓手机庞大的用户群将会对移动软件的发展提供良好的契机，正因为这样，大量的安卓软件应运而生，犹如为安卓智能手机的发展插上了翅膀。在这些大量的软件中，手机控制类软件备受欢迎，这些应用与传统的应用不同，他们是物联网时代的产物。实践证明，这类应用会给人们的生活带来极大的便利。

### 1.2 课题的研究目的和意义

近年来，由于智能化水平的提高，越来越多的学校借助多媒体来教学。这种教学方式

图文并茂，能极大提高学习者的学习热情，同时对教学者也提供了极大的便利。因此，使用 PPT 演示授课已成为主要的教学方式。但我们发现，在授课的过程中，对 PPT 的播放操作还是存在着许多问题的。例如，我们在听 PPT 演讲者做相关资料展示的时候，讲演者为了提高讲解质量，并不会一直停留在电脑前，而是来回走动，这种方式带来了一个问题，那就是当演讲者将要讲完一页 PPT 内容的时候，他需要走回放置电脑的地方，去翻动 PPT。即使后来有了 PPT 翻页笔来解决这种问题，但是使用翻页笔依然存在一些问题，例如：翻页笔忘记携带或者丢失，又或者翻页笔没有电量以及一些无法预知的故障等。还有一些硬性问题，比如我想在 PPT 上画图怎么办，那就又得被电脑鼠标控制，限制了演讲者的演讲自由。

为解决上述问题，我们决定基于安卓手机操作系统，开发一款能够远程控制 PPT 控制的软件，来解放演讲者，使 PPT 演示教学的效果达到最大化。这也是本可以的研究目的和意义所在。

### 1.3 研究现状

随着物联网的发展，手机能够做的事情越来越多。手机已经不是一个简单的通讯工具了，它能够给人们提供许多的帮助。其中在手机遥控方面的应用已取得了很好的效果。例如我们可以通过手机遥控电视，空调冰箱等日常家电。我们也可以用手机监控安防，检测环境等。但据我们调查，把手机遥控应用在 PPT 多媒体教学上的例子并不是很多。市场上也出现了一些这方面的应用，如 Mobel CC, Home Center 等，这些软件虽然功能强大，但是不够轻量级，过多的功能也会使用户眼花缭乱。还有一些付费软件，这类软件固然不错，可是在中国用户使用软件氛围下，这类软件能否被广泛应用还是个问题，毕竟用户大都喜欢用免费的软件。因此，研发出一款免费的轻量级的 PPT 多媒体教学控制软件还是很有用处的。

### 1.4 论文结构

第一章介绍安卓手机操作系统的发展概况，分析本课题研究的背景，介绍本课题研发的目的和意义，并介绍课题相关软件的发展现状，对市场上的一些应用进行分析。

第二章介绍安卓操作系统、C/S 软件开发架构、socket 通信和 Java 编程语言，并简要介绍在系统实现中所使用的一些开发工具。

第三章主要介绍系统的任务目标，进行系统的需求分析和可行性分析(包括对技术、经济、法律、环境和操作的可行性)。

第四章介绍系统的整体设计，其中包括系统流程设计、模块设计、交互设计，数据接收发送和处理。

第五章介绍各个模块实现方法，给出核心代码和系统运行界面等。

第六章总结自己的工作，指出特色与不足，给出改进建议。



## 第二章 系统相关技术介绍

本课题采用 C/S 软件结构，客户端采用 Java 语言开发，服务端采用基于安卓的操作系统开发。客户端与服务端使用基于 TCP/IP 协议的 Socket 通信方式。本章将简单介绍相关技术。

### 2.1 软件结构

#### 2.1.1 C/S 软件结构及工作模式

C/S(Client/Server)结构，即我们熟知的客户端/服务端结构<sup>[1]</sup>。它的基本工作原理是把计算机的任务分解，把一个大的任务分解成多个子任务，然后分配给多台计算机完成，即使用“功能分布”的原则。客户端负责数据的处理，对数据的表示以及为用户提供接口；服务端完成对数据库的管理功能。这种客户端请求服务，服务端提供处理方法的方式能够提高数据处理响应速度，使客户端更加轻量。

客户端程序的任务主要是将用户要处理的数据提交给服务器端，服务器接收到数据后，将会对数据进行处理，然后将处理的结果以特定的方式反馈给客户端。

#### 2.1.2 C/S 软件结构的优点

C/S 结构的优点是能够发挥出客户端的数据处理能力，很多工作都是可以通过客户端的处理后提交给服务器。具体表现在以下两点：

(1) 减轻服务端运行数据的压力。最简单的 C/S 结构的应用由两个部分组成，即客户端程序和服务器端程序。也就是我们熟知的前台程序和后台程序。当启动服务端程序后，服务器就等待客户端发来的请求。用户在客户端对数据进行操作时，客户端就会向服务端发送数据请求，服务端经过预定义好的格式对数据进行处理，执行相应的操作。

(2) 数据的储存管理更加透明。因为在进行数据的存储时，服务器程序和客户端程序是分开进行的，并且把一些前台不能处理的一些复杂情况放在服务端集中管理，例如对访问者权限的控制，对客户身份的验证等情况。所有这些对于在前台的用户而言，是“透明的”，他们无需知道背后的过程，就可以完成自己想要做的事情。也就是说在 C/S 体系下，数据库受到了专门的管理。

## 2.2 基于安卓的客户端开发环境

### 2.2.1 安卓操作系统

安卓系统是一个完全开源的操作系统，它是基于 Linux 系统开发<sup>[2]</sup>。通过 Android SDK 提供的开发文档（API）以及配套的开发工具，人们可以很方便的开发基于安卓的应用程序。安卓系统由安卓运行库、应用程序、应用程序框架、应用程序库和 Linux 内核等五个部分组成。安卓操作系统中也有一部分内置程序，包括日历、地图、通讯录和邮件等，这些程序都是由 Java 编写的。因为应用程序有着良好的程序框架，使得组件很容易实现重用，所以我们能够根据 API 提供的函数接口，很容易地实现我们想要的功能。

### 2.2.2 安卓的平台优势

#### （1）开放性

安卓操作系统因其开源性，对市场有很好的兼容。因此吸引了大批的移动终端厂商的加盟，同时也涌入了大批的开发人员。随着用户规模的扩大，平台应用的日益丰富，安卓平台也走向了成熟<sup>[3]</sup>。

#### （2）自由性

在之前，手机上的应用受到许多制约，例如受到运营商的网络限制，平台应用的匮乏。随着近年来移动网络通信的发展和安卓平台应用的井喷式爆发，这些限制将不复存在。

#### （3）丰富性

由于安卓系统的开放性，每家厂商都会开发自己的产品，这些产品功能各异，给用户提供了更多的选择。同时，在软件开发方面，安卓平台提供了一个自由的环境，第三方开发商不受约束，争相开发产品，虽然丰富了市场，但也会导致应用的质量下降，缺乏新颖度。

#### （4）强大的支撑

我们知道，安卓操作系统是谷歌公司开发的。一个强大的企业，能够保证平台的壮大和更新，为手机终端开发商和应用开发者提供了有力的支撑。

## 2.3 Java 开发语言

### 2.3.1 Java 简介

Java 是一种跨平台、面向对象的应用程序设计语言。Java 语言的开发技术具有显著地

通用性，广泛应用在游戏控制、数据中心、PC、互联网及物联网中。同时，Java 开发语言有着全球最大的开发者社群。

### 2.3.2 Java 的主要特性

#### (1) 安全性

由于 Java 通常使用在网络环境中，为防止恶意代码的攻击，Java 提供了一个安全的机制。除了 Java 语言具有安全性，对一些从网络中下载类，Java 也具有安全防范机制。如为防止网络中下载类的名字替代本地的类，Java 会为这些类分配不同的名字空间并会进行代码检查，并提供安全管理机制，给采用 Java 的应用设置安全哨兵。

#### (2) 可移植性

由于 Java 的体系结构是中立的，所以 Java 语言具有良好的可移植性。此外，Java 还对各种基本数据类型的长度进行了严格的限制。Java 系统本身就具有很强的一致性，可以用 Java 实现 Java 编译器，可以用 ANSI C 实现 Java 的运行环境。

#### (3) 支持多线程

线程是 Java 语言中的一种特殊对象，它可由 Thread 类或其子类创建。创建线程一般有两种方法：其一，使用 Thread (Runnable) 把 Runnable 接口的对象封装成一个线程；其二，创建一个类继承 Thread 类，在它里面中重写 run 方法。使用该类创建的对象称之为线程。需要注意的是 Thread 类已经实现了对 Runnable 的接口，所以，任何一个子类均有其 run 方法，所要运行的代码都包含在 run 方法中。用一组方法来控制线程的活动。Java 语言支持并发执行，并提供多线程同步（关键字为 synchronized）。

#### (4) 动态性

在设计之初，Java 语言就考虑了适应动态的环境。Java 程序中需要使用的类能够动态地载入到运行环境，也可以把需要用的类通过网络载入。这种做法有利于软件的升级。此外，基于运行时（runtime）系统，Java 中的类有运行时的表示，能检查运行时的类型。

因为 Java 语言以上等优良的特性，使得用 Java 开发大的应用具有健壮性和可靠性，这能够有效减少系统后期的维护费用。Java 对面向对象编程的支持和内嵌的强大的 API 大大缩短了程序开发的周期时间，降低了成本。Java 编译一次，可被移植到多平台上使用，极大地减少了多个平台之间的信息传递成本<sup>[4]</sup>。

### 2.3.3 java.awt.robot 类

Robot 类能够给测试自动化、自运行演示程序和其他需要控制鼠标和键盘的应用程序

生成本机系统输入事件。**Robot** 类的主要目的是便于 **Java** 平台实现自动测试。我们可以通过此类来完成系统的关键操作，并且系统的实现也都是基于此类完成的<sup>[5]</sup>。**Robot** 类的功能在对电脑控制方面十分强大，我们只是用了其中的一小部分。

## 2.4 Socket 通信

### 2.4.1 Socket 的概念

**Socket**(套接字)是通信的基石，它是 **TCP/IP** 网络通信协议中的基本操作单元。它是把网络通信过程中端点进行抽象的表示，其中包括了进行网络通信时需要的五种信息：连接协议、本机 **IP** 地址、本地进程的协议端口、远程主机 **IP** 地址和远程主机提供的协议端口<sup>[6]</sup>。

**Socket** 可以对应用程序的数据进行接受和发送，通过 **Socket**，可以把应用程序添加到网络中并实现和连接在同一网络中的应用程序进行通信。

### 2.4.2 Socket 的基本原理

在使用 **Socket** 通信时，服务器端会先生成 **ServerSocket** 对象，它的参数是整形的，表示着一个主机上未被占用的端口号，然后使用 **ServerSocket** 的 **accept()**方法，对客户端发送来的数据进行接收。**accept()**方法相当于一个死循环，一直处于堵塞状态，监听客户端发送来的数据。

客户端需要创建一个 **Socket** 对象来实现通信，它有两个参数，一个用来表示服务器端的 **IP** 地址，另一个来表示端口号。**Socket** 对象使用 **inputstream** 流读取服务器返回的数据，在处理完数据以后，通过 **outputstream** 流把数据发送给服务端。

## 2.5 开发工具

在本课题的项目中，使用 **Eclipse** 进行服务端的 **Java** 语言开发。安卓客户端采用 **google** 开发的 **Android Studio** 作为开发工具。

**Eclipse** 是 **IBM** 公司开发的一个继承编译环境，它是为了方便的开发 **Java** 软件而设计的。开发时代码的写入、编译、调试和运行都可以在同一个程序中完成。并且，**Eclipse** 有很多的插件，通过使用插件，能够完成许多功能，例如图形界面开发。

谷歌在 2013 年 **I/O** 大会上推出了最新研发出的安卓开发环境 **Android Studio**。根据这

个最新的环境，开发者能够很方便的进行屏幕适配，同时还支持多语言。相对于之前在 eclipse 上进行安卓开发，Android Studio 对比 Eclipse 具有更大的优势，提高了编译速度。现如今，Android Studio 的开源社区异常火热，为安卓的发展增添了不少的贡献。

## 第三章 系统分析

按照软件的设计原则，我们需要在系统设计之前，对系统进行需求分析和可行性分析，来确保系统能够正常设计。在本章中，我们对系统的需求进行了深入分析，并对可行性进行了分析（分别从技术、经济、法律、环境、操作可行性进行分析）。

### 3.1 系统需求分析

PPT 演示是如今技术讨论、信息报告和知识讲授的常用演示方式。但是我们发现，在演示 PPT 的时候，演讲者往往因为需要操控 PPT 而不能离开讲台，不方便与听众互动，即使现在有了翻页笔来解决这个问题，但是使用翻页笔依然存在一些问题，例如：翻页笔忘记携带或者丢失，又或者翻页笔没有电量以及一些无法预知的故障等。还有一些硬性问题，比如我想在 PPT 上画图怎么办，那就又得被电脑鼠标控制，限制了演讲者的演讲自由。为解决这些问题，一个基于 Android 的 PPT 控制软件应具有如下需求。

（1）PPT 控制功能：通过软件实现利用手机控制 PPT 的一些操作，包括开始播放、翻页和退出播放等。

（2）PPT 标记功能：实现对 PPT 进行标记的相关操作，包括开始标记、完成标记、撤销标记和返回 PPT 播放等。

（3）鼠标控制功能：根据鼠标的特性做出一个虚拟的鼠标，主要包括模拟鼠标移动和提供鼠标的左右键操作。

### 3.2 系统可行性分析

根据本课题的特点，我们对系统的可行性进行了分析，主要从技术可行性，经济可行性、法律可行性、环境可行性和操作可行性等几个方面来进行分析。

#### 3.2.1 技术可行性分析

本系统采用 C/S 架构。其中客户端我们是基于安卓操作系统开发的。因为安卓系统的开源特性，使我们能够很方便的进行开发。服务端，我们采用 Java 语言进行开发，其跨平台的特性能够使我们的程序具有很好的移植性。对于服务端与客户端的通信，我们采用的是基于 TCP/IP 协议的 Socket 通信。这些成熟的技术给我们提供了强大的支撑，使我们完

全能够开发出一款界面友好，操作便捷，实用高效的应用。

### 3.2.2 经济可行性分析

随着计算机行业的发展，相关从业人员的人数增加，使得软件市场的竞争日趋激烈，也使得软件研发技术日益成熟，软件市场更加规范。软件开发成本的降低，使得我们能够更加便捷的开发出自己设计出的产品。设计软件的目的也是为了改变用户的生活习惯，为用户提供不一样的便利。

本系统采用的是安卓操作系统和 Java 语言，以及 eclipse，Android studio 工具进行的开发。众所周知，这些软件和操作系统及语言都是免费的，所以，我们系统的开发成本极低。

### 3.2.3 法律可行性分析

本软件采用的开发语言是免费开源的，开发技术都是公开的。本系统的设计和实现独立完成，均未对任何其他产品和公司造成侵犯。因此，从法律角度看本产品是完全可行的。

### 3.2.4 环境可行性分析

本软件属于控制类软件的范畴，控制类的软件的目的就是为了使繁琐的操作变的简单，减少中间过程，从侧面来说，无意是高效环保的。并且，本系统的使用不会对环境造成任何危害。基于此，可以说我们的系统完全是节能环保的绿色软件。

### 3.2.5 操作可行性分析

基于安卓平台开发的手机应用，大都有着简单便捷的操作和友好的界面。我们根据用户使用习惯和实际情况，开发出了各种模块，其功能简单明了，具有良好的操作性。

## 第四章 系统设计

一个好的系统源自系统良好的设计。在本章中，我们对软件结构进行了设计，对软件的通信原理进行了探讨，并且对系统进行了模块化的分析设计。

### 4.1 系统的结构设计

本课题采用的是 C/S(Client/Server)系统架构，这是软件开发中的一种常用的两层架构，也就是客户端-服务端架构。在本文中，客户端是指运行在安卓手机上的程序；服务端有两种，一种采用的是基于数据库的服务器端，客户端经过网络连接，能对服务器端的数据库进行访问；另一种是 Socket 服务器端，客户端的应用通过 Socket 与服务端建立网络连接后，能和服务端进行通信<sup>[7]</sup>。C/S 架构具有良好的界面操作，并具有一定的安全性，可保证信息的安全传递，实现多层认证。本课题采用的服务器端是 Socket 服务器，Socket 的通信过程是：服务端建立 Socket 连接，对目的 IP 和端口号进行监听，等待客户端的接入。客户端接入后，服务端和客户端就建立连接。客户端与服务端的通信是输入/输出流。当完成通信需要断开，客户端向服务端发送关闭消息，服务端断开连接，关闭 Socket<sup>[8]</sup>。Socket 通信过程如图 4-1 所示。本系统采用的是 C/S 架构，客户端与服务端通过局域网进行数据通信<sup>[9]</sup>。也就是基于 TCP/IP 协议的 Socket 通信，系统的模型如图 4-2 所示。

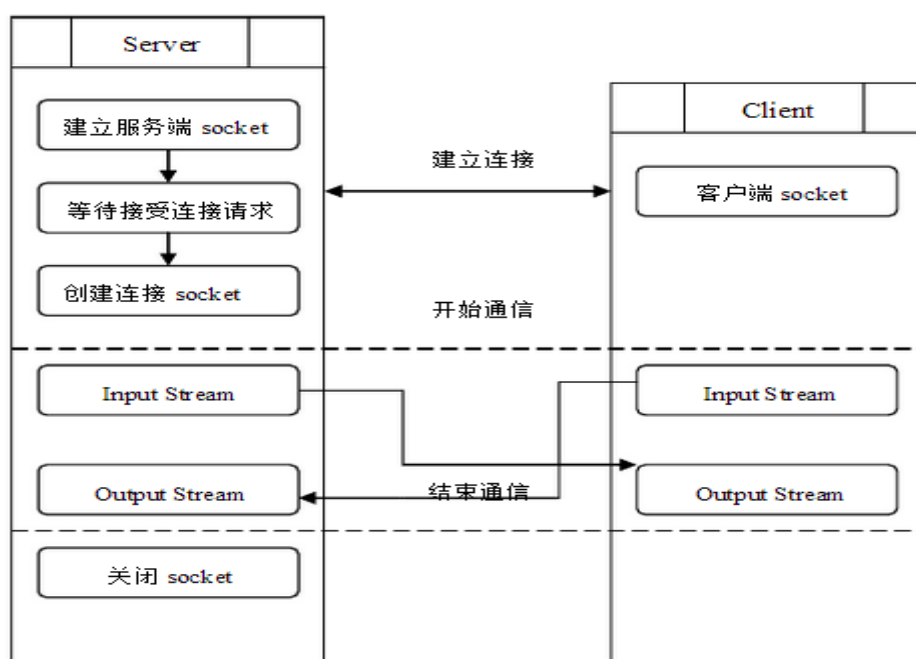


图 4-1 Socket 通信模型



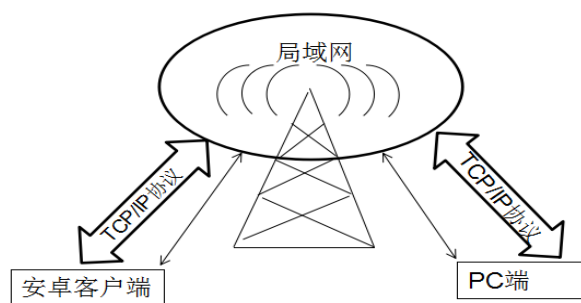


图 4-2 系统模型

## 4.2 系统模块设计

### 4.2.1 服务器端设计

基于 eclipse 开发环境，我们使用 Java 语言开发了服务器端。在服务器端，我们的详细设计流程如下：

（1）Socket 连接：因为需要与客户端进行通信，所以我们创建 Socket 对象，配置好 Socket 通信相关信息。

（2）消息处理：连接建立后，通过读取输入流信息，获取客户端发送的请求并采用 Robot 类对信息进行处理。

（3）界面设计：设计一个连接界面，负责提示用户和辅助用户对客户端进行连接。我们以窗口的形式显示相关信息。

### 4.2.2 客户端设计

客户端基于 Android Studio 开发，根据需求，我们对客户端功能的设计如下：

#### （1）连接模块

连接模块即用户登录界面。根据服务器的可视化的信息展示，我们按照操作在登录界面进行操作。首先，在客户端界面内输入 IP 地址和端口号，然后进行连接确认，如果连接成功，会对页面进行跳转，如果连接失败，需要重新进行操作进行确认。连接模块流程图如图 4-3 所示。

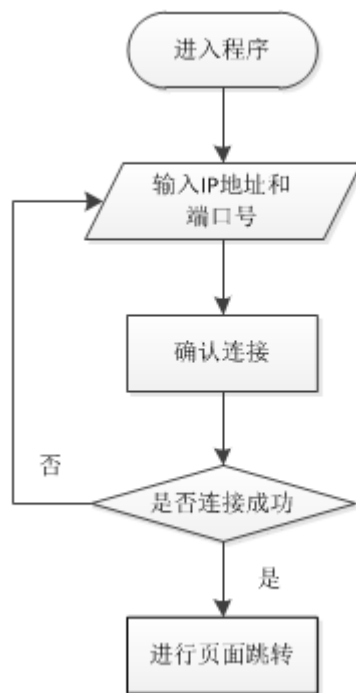


图 4-3 连接模块流程图

## （2）功能选择模块

当客户端与服务器成功建立连接后，就会跳转功能选择模块。在此模块中，有两个功能可供选择：**PPT 控制功能**和**鼠标控制功能**。当选择 **PPT 控制模式**进入 **PPT 控制**页面；当选择**鼠标控制模式**，将会转到**鼠标操作**界面。功能选择模块设计如图 4-4。

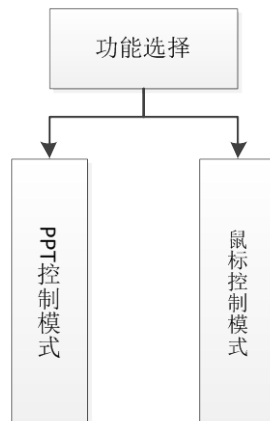


图 4-4 功能选择模块

## （3）PPT 控制模块

当选择 **PPT 控制模式**，将会跳转至 **PPT 控制**界面。在这个模块中，我们根据实际情况，设计符合用户操作习惯、满足基本操作需求的功能。在客户端打开需要使用的 **PPT** 文件后，我们即可对 **PPT** 文件进行操作。提供的功能有包括：（a）上一页，对 **PPT** 进行上翻页操

作；（b）下一页，对 PPT 进行下翻页操作；（c）开始播放，放映打开的 PPT 文件；（d）退出播放，退出正在放映的 PPT 文件。

此外还将提供标记功能，当用户选择标记功能时，将会跳转至标记界面。在此标记页面中，可以选择的功能包括：（a）开始标记，选择 PPT 上需要标记地方的开始位置；（b）进行标记，我们采用手势识别的方式来获取用户手指滑动的位置，用户在指定的区域进行标记操作，会在 PPT 页面上显示出标记的内容；（c）完成标记，在完成标记后，点击完成，此时程序会退出标记模式，标记的内容会保留在 PPT 页面上；（d）撤销标记，完成标记后，如果需要取消标记的内容则选择此功能，点击后会将标记删除。

以上操作在点击后客户端都会把指令发送给服务端，服务端得到数据后进行解析，得到操作指令，然后根据 robot 类来进行操作。PPT 控制模块设计如图 4-5 所示，标记模式的功能设计如图 4-6 所示。

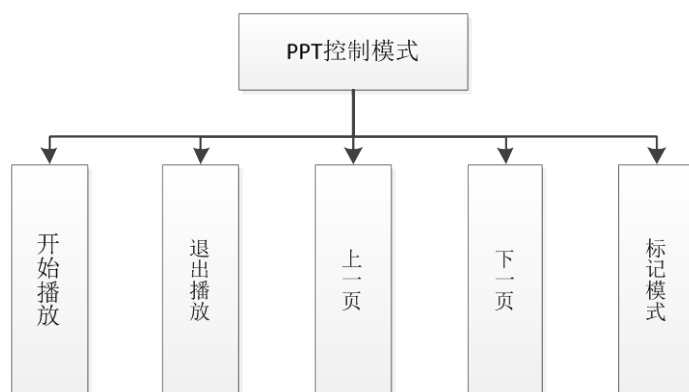


图 4-5 PPT 控制模块

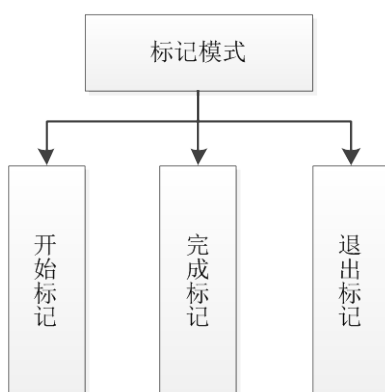


图 4-6 标记模块

#### （4）鼠标控制模块

当用户选择鼠标控制模式时，页面将会跳转至鼠标控制操作界面。我们根据鼠标的特点模拟鼠标的左键单击与右键点击功能，并且根据服务端的屏幕尺寸，换算出相对于客户

端的尺寸比，这样在给定的手势滑动区域，就相当于模拟出了一个屏幕。用户可以按照页面提供的按钮选择要实现的操作。当点击左键时，客户端会根据点击信息，向服务端发送数据，服务端获取数据后对数据进行解析，取得用户想要进行的操作，然后使用 `robot` 类执行相应的操作。鼠标右键的实现与滑动的实现也是如此。图 4-7 为鼠标控制模式的功能设计图。

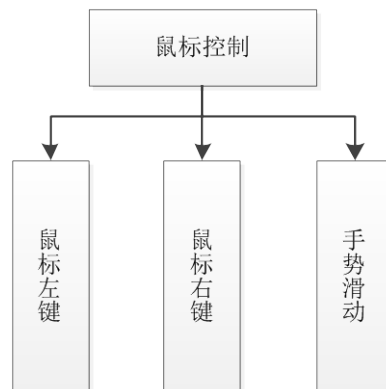


图 4-7 鼠标控制模块

## 第五章 系统实现

经过对系统的设计、关键技术的分析和模块相关接口的设计，我们已经设计好了项目的结构。在本章中，我们对具体实现的相关细节及核心技术进行阐述。

### 5.1 服务端实现

客户端连接模块的实现是采用 `JFrame` 的子类实现。`JFrame` 类是 Java 语言中可视化编程的一部分，它相当于一个容器，开发者可以将需要使用的组件添加进去，然后 `JFrame` 会重新组织这些组件然后可视化的显示出来。`JFrame` 类是 Java 界面开发中属于最底层的容器，在操作系统中的显示是以窗口形式，所以我们可以看到这些界面与操作系统自带的程序具有相同的窗口特性。图 5-1 是客户端运行界面图。

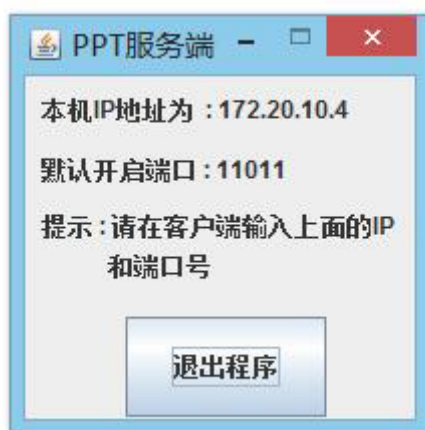


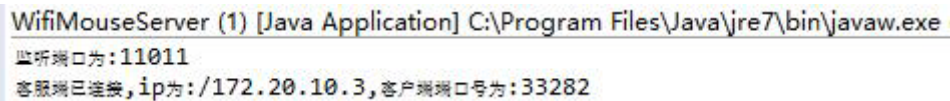
图 5-1 客户端运行界面

在以上界面中，我们可以看到此界面为我们提供了本机的 IP 地址信息和默认提供的端口号。我们先用 `JLabel` 类分别创建两个对象 `IPLabel` 和 `PortLabel`。使用 `getHostAddress()` 方法获取本机的 IP 地址，获取后用一個字符串进行接受，然后把 `IPLabel` 的 `text` 属性设置为此字符串，这样就可以给用户提示本机 IP 地址。端口号也是类似，只不过我们使用默认端口号 11010。

当运行服务端程序后，服务器端会开始进入监听模式，此时我们需要设置此 `Socket` 对象为长连接状态，即让 `Socket` 连接一直处于监听的状态，这样就能及时对客户端发送的消息进行处理。

当运行客户端程序后，此时服务器就会进入监听状态，等待客户端的接入，此时服务端监听的正是自己提供的默认端口号 11010。当开始连接后，服务端会处理客户端发来的

数据信息。图 5-2 为连接成功提示的信息。



```
WifiMouseServer (1) [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe
监听端口为:11011
服务端已连接,ip为:/172.20.10.3,客户端端口号为:33282
```

图 5-2 连接成功提示信息

连接成功后，这时如果客户端进行操作，就会向服务器端发送信息，服务器端会对这些信息进行处理。服务端读取部分核心代码如下：

```
public void getReqData(InputStream is){//读取数据
    int bufferSize = 200;
    byte[] buffer = new byte[bufferSize];
    int eof;
    try {
        while((eof = is.read(buffer)) != -1){
            String str = new String(buffer,Charset.forName("UTF-8"));
            String cmd = checkIfCmd(str);
            buffer = new byte[bufferSize];
            if(!"".equals(cmd))doAction(cmd);
        }
    } catch (IOException e1) {
        e1.printStackTrace();
    }
}
```

在进行数据读取之前我们会设置一个判断，即如果我们在服务器端提供的操作与客户端传来的指令匹配的话就执行，不匹配的话会提示错误。创建 **Pattern** 对象 **pattern**，对 **pattern** 进行赋值为：

```
Pattern.compile("\\[(up|down|left|right|enter|esc|close|double|start|done|undo)\\]");
```

匹配代码如下：

```
Matcher matcher = pattern.matcher(str);
if(matcher.find()){
    return matcher.group(1);
}
```

```

    }else{
        System.out.println("操作无匹配");
    }
    return "";
}

```

当用户使用完毕时，可以选择关闭此程序，客户端与服务端会断开连接，并释放服务端所占用端口。我们采用一个 **button** 来触发这个事件。当用户按下按键时，此动作被认为是发生了一个事件，会被事件监听器捕获到，随后，事件监听器会对这个事件进行处理，去调用对应的实例方法。

## 5.2 客户端实现

我们对客户端按照功能模块进行划分，分别有连接模块，功能选择模块，PPT 控制模块，标记模块，鼠标控制模块。以下我们对各个模块的详细实现进行依次描述。

### 5.2.1 连接模块

客户端的连接主要就是根据服务端界面提示的信息，依次输入主机 IP 地址和主机端口号。如果正确输入的话，页面会进行跳转，即连接成功，若无任何响应则说明连接未成功，需要重新确认信息进行连接。连接界面如图 5-3 所示。

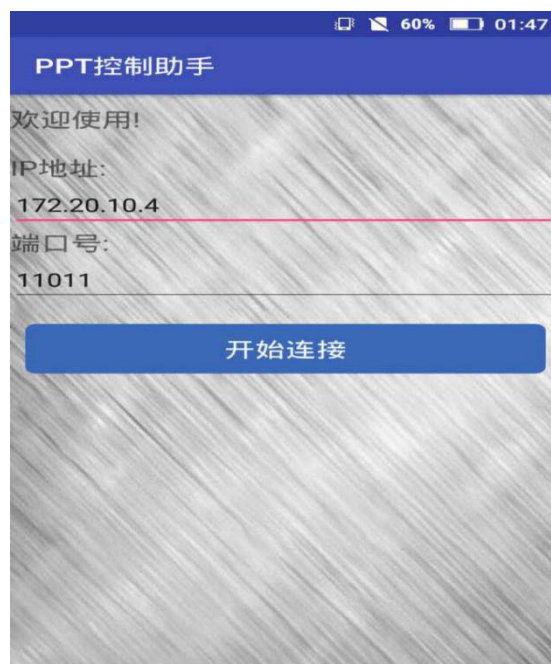


图 5-3 连接模块界面

因为要进行 Socket 连接,本程序中只有一个 Socket 对象,为了方便使用,我们把 Socket 对象设置为单例,这样就能全局使用,并能对 Socket 对象进行操作。创建单例的代码如下:

```
public Socket socket;

private static NetUtils instance = new NetUtils();

private NetUtils(){
    super();
}

public static NetUtils getInstance(){
    return instance;
}
```

此外,在移动开发中,对网络请求的处理一般放在子线程中进行处理,因为此类操作一般都是比较耗时的,放在主线程会造成程序卡死。所以我们创建了 SocketThread 类来实现子线程的开辟。开辟子线程的代码如下:

```
private Socket s;

// 定义向 UI 线程发送消息的 Handler 对象
private Handler handler;
private String ip,port;
Utils.CallBack callBack;
public SocketThread(Handler handler,String ip,String port,Utils.CallBack callBack){
    this.handler = handler;
    this.ip = ip;
        this.port = port;
        this.callBack = callBack;
    }
    public void run()
    {
        try
        {
            NetUtils.getInstance().connect(ip, port,callBack);
            Looper.prepare();
            Looper.loop();
        }
    catch (Exception e)
    {
        e.printStackTrace();
    }
}
```



```
}
```

这样我们就可以在 MainActivity 中进行 socket 连接处理了。

### 5.2.2 功能选择模块

在成功连接后，页面会跳转到功能选择界面。为使用户更加方便的选择，我们在此页面中提供了两个 button 按钮。设置 button 的 id 分别为 ppt 和 mouse，然后根据此 id 来判断是哪个 button 进行了点击。功能选择是在 SelectActivity 中进行的。运行结果如图 5-4 所示。

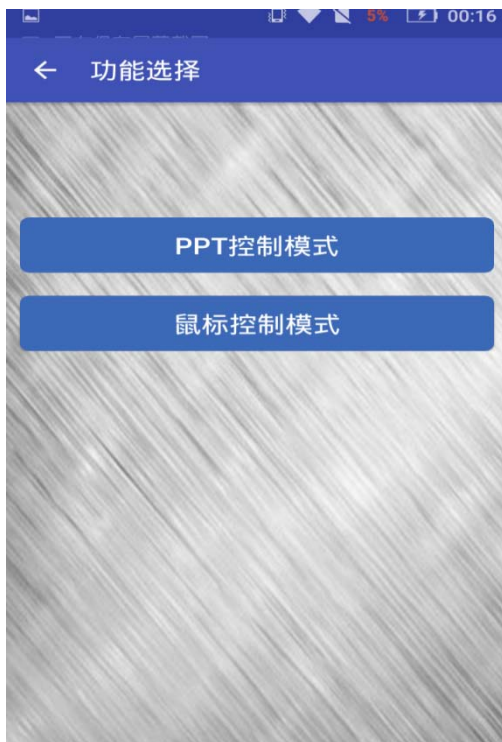


图 5-4 功能选择界面

关键代码如下：

```
ppt = (Button) this.findViewById(R.id.ppt);
ppt.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        startActivity(new Intent(SelectActivity.this,PPTActivity.class));
    }
});
mouse = (Button) this.findViewById(R.id.mouse);
mouse.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        startActivity(new Intent(SelectActivity.this,MouseActivity.class));
    }
});
```

### 5.2.3 PPT 控制模块

在 PPT 控制页面中，可以对 PPT 进行控制操作，本页面中有 5 个 button，分别为上一页，下一页，进入标记，开始播放，退出播放。每个 button 都设置了监听事件，当点击后会调用相应的方法，向服务端发送信息。因客户端与服务端已经建立 Socket 连接，此时在服务端操作系统中打开需要演示的 PPT，点击相应的按钮，即可实现对应的操作。当点击了开始播放，客户端通过 Socket 发送了 Enter 信息，此时服务端解析数据后，会调用键盘的 F5 键进入播放模式。其他的操作同理，上一页对于 Up, 下一页对于 Down, 退出播放对应 Esc。点击进入标记是跳转到标记模式。图 5-5 为 PPT 控制界面。

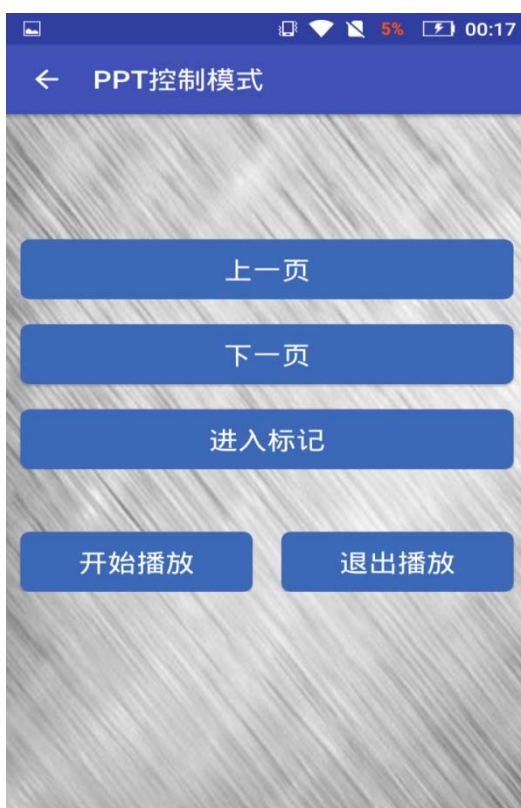


图 5-5 PPT 控制界面

客户端 PPT 控制界面核心代码如下：

```
switch (v.getId()){
    case R.id.handle_up:
        try {
            NetUtils.getInstance().send("[up]");
        } catch (Exception e) {
            e.printStackTrace();
        }
        break;
    case R.id.handle_down:
        try {
```

```

        NetUtils.getInstance().send("[down]");
    } catch (Exception e) {
        e.printStackTrace();
    }
    break;
case R.id.handle_enter:
    try {
        NetUtils.getInstance().send("[enter]");
    } catch (Exception e) {
        e.printStackTrace();
    }
    break;
case R.id.handle_esc:
    try {
        NetUtils.getInstance().send("[esc]");
    } catch (Exception e) {
        e.printStackTrace();
    }
    break;
case R.id.handle_mark:
    startActivity(new Intent(PPTActivity.this, Mark.class));
    break;
default:
    break;
}

```

PPT 控制操作流程如下。首先在服务端端打开 PPT，图 5-6 为 PPT 打开后的初始状态。

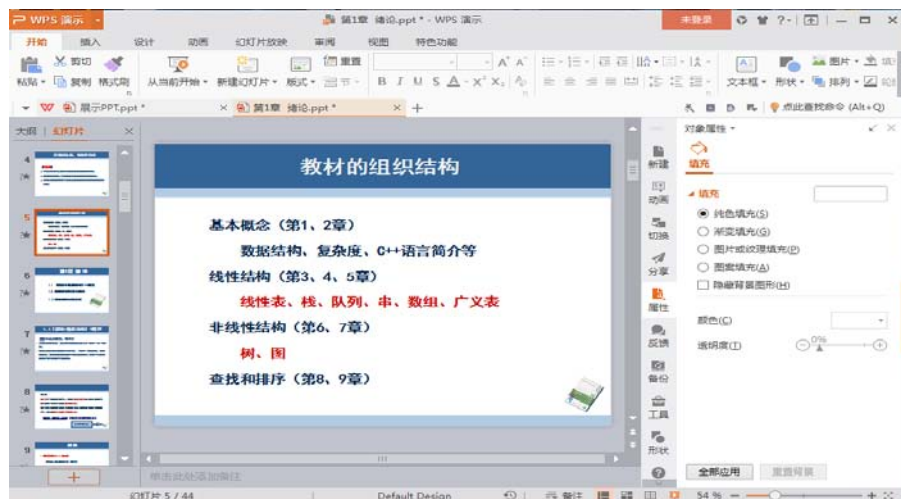


图 5-6 PPT 初始状态

当在客户端点击“开始播放”后，PPT 即进入播放状态，如图 5-7 所示。

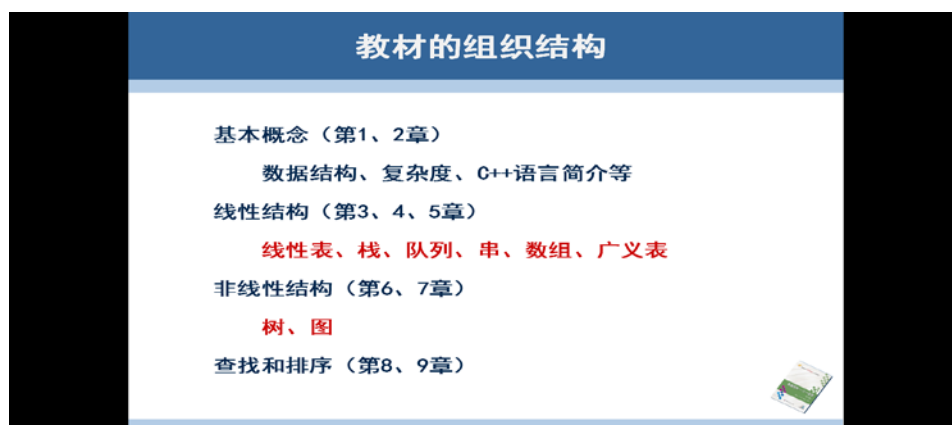


图 5-7 点击播放后效果

若在客户端点击“下一页”，则 PPT 进行下翻页操作，如图 5-8 所示。



图 5-8 点击下一页操作后效果

若在客户端点击“上一页”，则 PPT 进行上翻页操作。当点击“退出播放”按钮时，PPT 会退出全屏并回到播放起点位置。

#### 5.2.4 标记模块

此模块是嵌套在 PPT 控制模式中，但此功能相对其他操作独立，故把此部分抽离出来进行介绍。当点击标记模式后，客户端会进入标记模式界面，如图 5-9 所示。

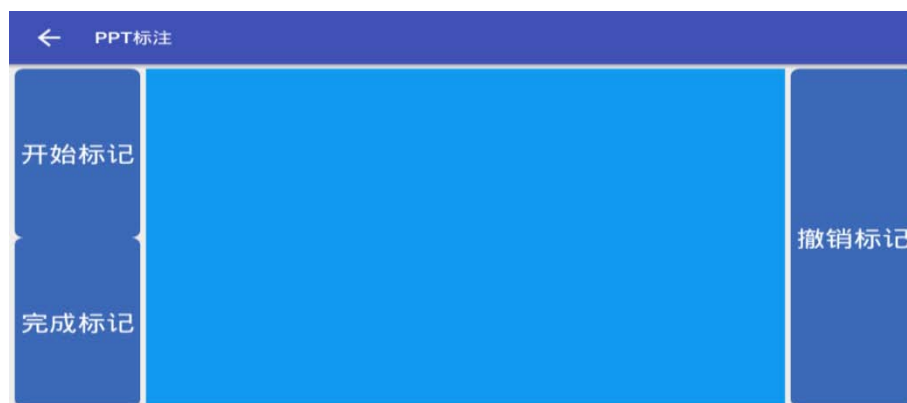
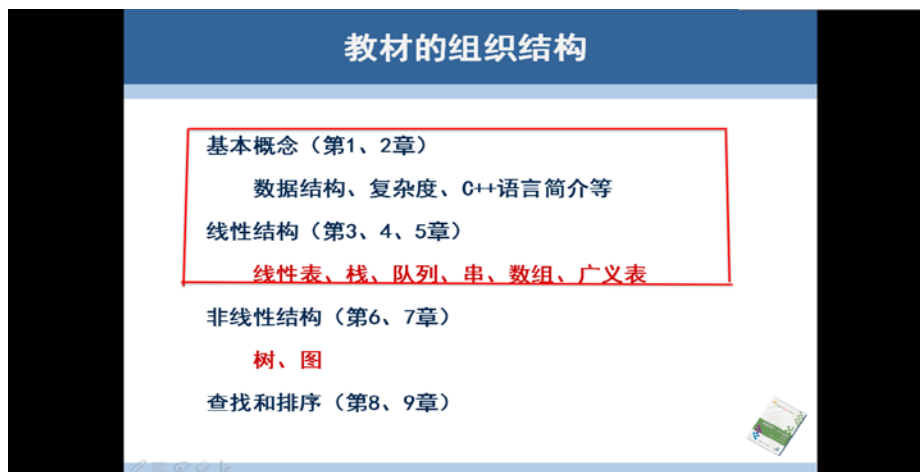


图 5-9 标记模式

在客户端此界面分为三个部分，“开始标记”设置为 start 按钮，“完成标记”为 done，“撤销标记”为 undo。当点击“开始标记”即会向服务端发送信息，服务端执行 Ctrl+P 键的操作进入标记，此时客户端屏幕的空白区域就对应服务端的屏幕区域，我们可以在客户端屏幕的空白区域进行标记，标记效果如图 5-10 所示。



5-10 标记效果

当选择“撤销标记”，服务端将执行键盘 E 键操作，标记的内容会被撤销掉。如果需要保留标记内容，则点击“完成标记”按钮，此时会保留标记内容，并且会退出标记模式。客户端功能选择操作和 PPT 控制模式下操作类似，我们就不再赘述。标记功能中的核心是手势滑动的处理，为此，我们写了一个 onTouch 方法，用 onTouch 方法来获取手势触摸的位置。我们首先创建 MotionEvent 的一个对象 event 来处理手势相关操作，并用两个浮点类型的变量 x 和 y 来接收手势坐标；然后根据不同的操作获取手势操作的位置，并将这个位置发送给服务端，服务端收到数据后对数据解析获取内容，也就是发送过来的地址，经过坐标转换后执行鼠标移动操作，能够将鼠标指针移动到指定的位置。

### 5.2.5 鼠标控制模块

根据鼠标的结构特点及基本操作属性，我们设计了鼠标控制模式。为模拟鼠标，我们创建了两个 button 分别代表鼠标左键和鼠标右键，对于鼠标左键 button，点击时触发监听事件，在鼠标左键的监听事件中，向服务端发送“Left”信息，客户端接受消息后，在鼠标所在的位置，调用 robot 类对鼠标左键进行单击操作，此时相当于鼠标左键单击了该区域。对于鼠标右键 button，则执行的是鼠标右键操作。为获取用户手势位置，我们提供一个 view，用户可在该区域进行滑动，客户端捕捉到滑动位置后会将该位置发送给服务端，服务端对位置进行换算，和本机显示器尺寸相匹配。其界面如图 5-11 所示。

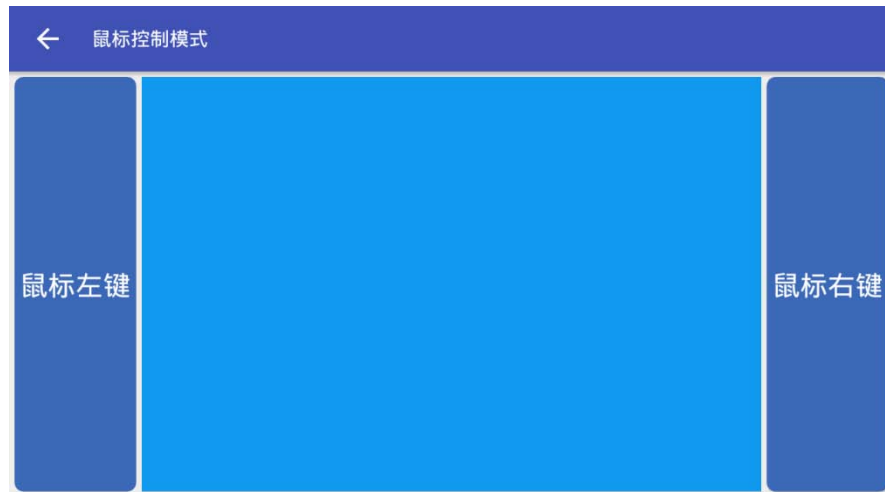


图 5-11 鼠标控制模式

鼠标的操作有单击、双击、右击和移动这四个功能。双击的功能实质上就是做两次单击，即单击中间有个短暂的延迟，因此可以在服务端对鼠标单击进行两次执行来实现双击操作。客户端的鼠标控制代码如下。

(1) 鼠标按键操作的核心代码

```
switch (v.getId()){
    case R.id.handle_left:
        long currentTime=System.currentTimeMillis();
        if(currentTime-oldTime<2*1000){
            Utils.showToast(this,"双击");
            try {
                NetUtils.getInstance().send("[double]");
            } catch (Exception e) {
                e.printStackTrace();
            }else{
            try {
                NetUtils.getInstance().send("[left]");
            } catch (Exception e) {
                e.printStackTrace();
            }
            oldTime=currentTime;
        }
        break;
    case R.id.handle_right:
        try {
            NetUtils.getInstance().send("[right]");
        } catch (Exception e) {
            e.printStackTrace();
        }
        break;
}
```

```

        default:
            break;
    }
}
(2) 鼠标移动操作的核心代码
float x = event.getX();
float y = event.getY();
try{
    switch(event.getAction())
    {
        case MotionEvent.ACTION_DOWN:
            focused.setText(x + "" + y + " down");
            Log.e("Mouse", "x-" + x + " y-" + y + " state-down");
            NetUtils.getInstance().send(x + "", y);
            break;

            case MotionEvent.ACTION_UP:
                focused.setText(x + "" + y + " up");
                Log.e("Mouse", "x-" + x + " y-" + y + " state-up");
                NetUtils.getInstance().send(x + "", y);
                break;

            case MotionEvent.ACTION_MOVE:
                focused.setText(x + "" + y + " move");
                Log.e("Mouse", "x-" + x + " y-" + y + " state-move");
                break;
    }
    return true;
}
catch(Exception e)
{
    Log.v("touch", e.toString());
    return false;
}

```

我们对鼠标操作进行测试：当在客户端选定鼠标操作模式，界面会跳转至鼠标操作模式。此界面模拟了鼠标左右键功能和滑动功能。当在滑动区域进行滑动操作时，在客户端显示器可见同步的操作，并且滑动区域与显示器屏幕适配。为进一步检验，我们对滑动区域进行有目标性滑动，先滑动至左上方对应服务器端屏幕“我的电脑”，单击左键表示选定服务端操作系统桌面上的我的电脑。然后单击右键操作，此时服务端操作为对“我的电脑”的属性进行查看，继续滑动至“属性”选项，再次选定后单击，即进入到详细界面。操作界面如下图 5-11 和 5-12 所示，图 5-13 为操作效果。



图 5-11 鼠标单击选定操作



图 5-12 鼠标右键操作

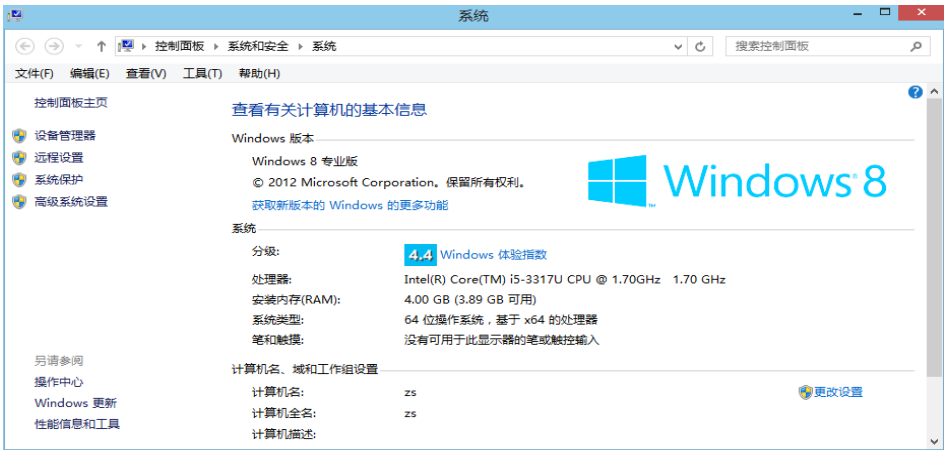


图 5-13 操作结果



### 5.2.6 断开连接模块

程序使用完毕后，需要进行断开连接操作，释放客户端被占用的端口号。当退出至初始界面，将有断开连接的按钮，此按钮在进行成功连接后出现，如图 5-14 所示。点击断开连接，程序会弹出一个提示框进行操作提示，表示用户已经成功断开连接，如图 5-15 所示。点击确定后即可退出程序。我们在此部分对页面数据进行了缓存处理，方便用户再次连接时，不需要重复输入信息。



图 5-14 断开连接操作



图 5-15 断开成功

### 5.3 服务端数据处理

服务端接收客户端发送的数据后，对数据进行解析，获取到操作指令。本系统将执行的操作封装在一起集中处理，提高了工作效率，也使代码量大大减少。因为在客户端的实现方法中我们已经对接收数据做出了处理，故在服务端接受指令后，我们对给定的正则表达式进行编译，用字符串接受操作指令，并且把字符串与正则表达式进行匹配，如果操作指令匹配成功，程序会执行操作，若指令不匹配，则无响应。故我们只需要匹配出这些操作即可。以下给出客户端操作处理的核心代码。

```
public void doAction(String str){
    if("left".equals(str)){ //单击
        robot.mousePress(InputEvent.BUTTON1_MASK);//按下左键
        robot.mouseRelease(InputEvent.BUTTON1_MASK);//释放左键
    }else if("enter".equals(str)){ // 进入全屏
        robot.keyPress(KeyEvent.VK_F5);
    }else if("right".equals(str)){ //右击
        robot.mousePress(InputEvent.BUTTON3_MASK);//按下右键
        robot.mouseRelease(InputEvent.BUTTON3_MASK);//释放右键
    }else if("up".equals(str)){ // 上一页
        robot.keyPress(KeyEvent.VK_UP);
    }else if("double".equals(str)){ // 双击
        robot.mousePress(InputEvent.BUTTON1_MASK);//按下左键
        robot.mouseRelease(InputEvent.BUTTON1_MASK);//释放左键
        robot.delay(100);//停顿 100 毫秒,即 0.1 秒
        robot.mousePress(InputEvent.BUTTON1_MASK);//按下左键
        robot.mouseRelease(InputEvent.BUTTON1_MASK);//释放左键
    }else if("down".equals(str)){ // 下一页
        robot.keyPress(KeyEvent.VK_DOWN);
    }else if("esc".equals(str)){ // 退出全屏
        robot.keyPress(KeyEvent.VK_ESCAPE);
    }else if("start".equals(str)){ // 标记模式
        robot.keyPress(KeyEvent.VK_CONTROL);
        for(int i=1;i<=2;i++)
        {
            robot.keyPress(KeyEvent.VK_P);
            robot.keyRelease(KeyEvent.VK_P);
        }
        robot.keyRelease(KeyEvent.VK_CONTROL);
        robot.mousePress(InputEvent.BUTTON1_MASK);//按下左键
    }else if("undo".equals(str)){ // 撤销标记
        robot.keyPress(KeyEvent.VK_E);
    }
```

```

}else if("done".equals(str)){ // 完成标记
    robot.mouseRelease(InputEvent.BUTTON1_MASK);//释放左键
    robot.keyPress(KeyEvent.VK_ESCAPE);
}else if("close".equals(str)){ // 关闭
}else { // 移动
    StringTokenizer st = new StringTokenizer(str,",",false);
    double x = Double.parseDouble(st.nextToken()) * 1.55;
    double y = Double.parseDouble(st.nextToken()) * 1.26;
    System.out.println(x+""+y);
    robot.mouseMove((int)x, (int)y);
    robot.setAutoDelay(0);
}
}

```

## 第六章 总结和展望

### 6.1 总结

针对目前演示 PPT 文稿时出现的对演讲者造成不便的情况，设计和实现了基于 Android 的 PPT 控制软件。本文首先介绍了研发的背景，对课题相关软件的开发现状进行了分析，确立了课题目标和意义；然后，针对课题内容介绍了开发中所用到的关键技术；接着对系统进行需求分析，确定了系统的开发结构，对系统能实现什么效果提出了明确的目标，并对系统的可行性进行了分析，最后重点介绍了系统的详细设计和功能实现。

本课题采用了 C/S 软件架构，服务端基于 Eclipse 采用 Java 语言进行开发，客户端使用 Android studio 开发工具进行开发，服务端与客户端的通信采用 Socket 通信。在开发的过程中采用模块化的设计思想，对功能模块进行抽离，严格按照低内聚，高耦合的代码原则。基于上述技术和开发原则，设计并开发出一个满足日常 PPT 演示要求，使用非常便捷的软件，解决了 PPT 演示的一些问题，提高了演示效率。本课题的主要研究成果如下：

(1) 根据实际使用情况，基于 C/S 架构，分别设计了 windows 平台的服务端和安卓平台的客户端。

(2) 实现了服务端的信息传递以及信息处理功能，并设计出了友好的操作界面，给用户做出了操作提示。

(3) 设计了针对 PPT 的主要操作的功能，实现了对 PPT 进行“上一页”，“下一页”，“进入播放”，“退出播放”和“标记模式”等功能。

(4) 由于安卓智能手机的普及，设计出了安卓客户端，对用户界面和交互都做出了良好的实现。在实际的使用过程中，用户可以很方便地对软件进行安装与操作。同时，本软件是免费并且是极其轻量化的，这是本软件的一个优势。

### 6.2 展望

根据课题设计目标，本系统已经基本满足设计要求。但是，系统中依然存在着如下所述的需要解决的问题。

(1) 服务端的操作相对繁琐，用户在使用时，需要启动服务器，获取提示信息，然后手动将信息在客户端进行输入。

(2) 客户端的鼠标功能不完善，缺少鼠标滚轮设计。

(3) 标记步骤操作繁琐，每进行一次操作，需要跳转页面，然后点击开始、标记、完成操作。

对于这些问题，会在今后有针对性的进行改进，可供参考的方案如下：

(1) 将服务端信息生成二维码，用户只需要扫描一下二维码就行，这样就不用在客户端输入连接信息。

(2) 鼠标增加滚轮功能，实现滚轮的前后滚动，这样能够支持对文本的浏览操作等，进一步完善鼠标控制功能。

(3) 对标记功能进行优化，减少标记操作的步骤，以便提高用户体验。

智能手机的普及给我们提供了很多方便，在今后的开发过程中，将针对现实中存在的一些问题，结合手机开发，做出更多能够提高工作效率的软件。

## 致 谢

在王新宇老师的悉心指导下，本人经过一个学期的努力，完成了课题的设计的目标和论文的撰写。能够顺利的完成这些工作，特别要感谢王新宇老师。老师为人随和，教学严谨，对学生严格要求、认真负责。在开发中遇到的问题，老师会认真提出指导建议，使我学习到了很多知识。不仅是学习上的帮助，在平时的生活中老师也给了我足够的理解和支持。在此，向老师表示由衷的感谢！

此外，感谢我的朋友和同学，在毕业设计期间给予了我很多有效的帮助。

最后，对参加论文审查和答辩的各位老师表示由衷的感谢！

## 参考文献:

- [1] 郭宏志. Android 应用开发详解[M]. 北京: 电子工业出版社, 2010
- [2] 熊刚. 基于 Android 的智能手机的设计与实现[D]. 武汉: 武汉理工大学, 2010
- [3] 徐诚. 零点起飞学 Android 开发[M]. 北京: 清华大学出版社, 2013
- [4] 耿祥义, 张跃平. Java2 实用教程[M]. 北京: 清华大学出版社, 2012
- [5] 李刚. 疯狂 JAVA 讲义[M]. 北京: 电子工业出版社, 2013
- [6] 曾宇, 曾兰玲, 杨治. 计算机网络技术[M]. 北京: 机械工业出版社, 2013
- [7] 韩希义. 计算机网络基础[M]. 北京: 高等教育出版社, 2004
- [8] 周恒民译. Java TCP/IP Socket 编程(原书第 2 版)[M]. 北京: 机械工业出版社, 2009
- [9] 张良银. 浅论 C/S 和 B/S 体系结构[J]. 工程地质计算机应用, 2006(4): 20-23