**基于Android的手机日程管理系统的设计与实现**

**详细设计说明书**

**目录**

1、引言 1

1.1、编写目的 1

1.2、背景说明 1

1.3、定义 1

1.4、参考资料 1

2、程序系统的结构 2

3、功能模块设计说明 2

3.1、程序描述 2

3.2、功能 3

3.3、性能 4

3.4、输人项 4

3.5、输出项 4

3.6、算法 4

3.7、接口 6

3.8、限制条件 6

4、界面设计说明 6

4.1、程序描述 6

4.2、功能 6

4.3、性能 6

4.4、输人项 6

4.5、输出项 6

4.7、限制条件 8

5、安全设置设计说明 8

5.1、程序描述 8

5.2、功能 8

5.4、输人项 9

5.5、输出项 9

5.6、算法 9

5.7、接口 10

5.8、限制条件 10

5.9、尚未解决的问题 11

6、其他需求设计 11

# 1、引言

## 1.1、编写目的

本文档用于说明本软件开发的详细过程，使毕业设计审核组了解软件的内部结构，从而对本日程管理系统有更深入的了解，便于审核。

## 1.2、背景说明

待开发软件系统的名称：基于Android的手机日程管理系统

用户：android手机用户

项目开发环境：eclipse、Android系统平台

## 1.3、定义

Android 手机日程 详细设计

## 1.4、参考资料

[1]赵亮;张维;;基于Android技术的界面设计与研究[J];电脑知识与技术;2009年29期.

[2]陈昱;江兰帆;;基于Google Android平台的移动开发研究[J];福建电脑;2008年11期.

[3]蔡罗成;;Android后台监听实现机制浅析[J];信息安全与通信保密;2010年06期.

[4]王会进,方锦亮;基于JAVA的自定义控件应用技术研究[J];微型机与应用;2005年05期.

[5]陈璟;陈平华;李文亮;;Android内核分析[J];现代计算机(专业版);2009年11期.

[6]姚昱旻;刘卫国;;Android的架构与应用开发研究[J];计算机系统应用;2008年11期.

[7]方明;田野;户银龙;;基于Android平台的嵌入式软件项目组织方法[J];电脑知识与技术;2011年34期.

[8]叶炳发;Android操作系统移植及关键技术研究[D];暨南大学;2010年.

[9]岳传真;Android系统移植和应用程序开发[D];复旦大学;2010年.

[10]刘敏;移动终端的Android移植与应用程序设计[D];西安电子科技大学;2011年.

[11]胡思捷;基于Android平台的触摸屏系统设计及实现[D];南开大学;2011年.

[12]施金兰;基于Android平台的网络共享研究与实现[D];华东师范大学;2011年.

[13]李凯;Android操作系统分析与移植[D];华南理工大学;2011年.

[14]刘瑞顺;基于Android平台的智能手机输入法研究与设计[D];汕头大学;2011年.

# 2、程序系统的结构

手机日程管理系统结构设计如下：



图1 手机日程管理系统结构设计图

# 3、功能模块设计说明

## 3.1、程序描述

主要功能：用于用户的可视化界面操作，有助于用户快速熟悉软件。

可扩展性：能够适应应用要求的变化和修改，具有灵活的可扩充性。

## 3.2、功能

主要有设置课程、查看课程、新增日程、查看日程4个选项

设置课程：



图2 设置课程程序流程图

通过点击课程表界面，用户可以输入课程名和地点。设计每天天共有8节课程和一节选修课，用户可输入星期一到星期五的所有课程安排信息。

查看课程：



图3 查看课程程序流程图

用户可查看输入的课程信息，设计只要左右滑动界面就可以查看星期一到星期五的所有课程信息。

新增日程：



图4 新增日程程序流程图

点击新增日程，就跳转用户可输入的新增日程界面，通过点击下拉菜单可选择日程类型是会议还是约会，在内容框里输入日程信息，输入所花时间，点击评分就是日程的优先级。

查看日程：



图5 查看日程程序流程图

用户点击查看日程后，输入正确密码，程序自动调用系统内部的日历模块，即显示日历，点击今天的日期，即可查看今天的日程安排，点击今天的日程信息，即可跳转编辑此次日程信息界面。

## 3.3、性能

当用户选择菜单时能够快速反应，以满足用户的要求。

## 3.4、输人项

设置课程：

课程名和地点

查看课程：

用户通过点击输入操作命令

新增日程：

类型、内容、耗时、评分（日程优先级）

查看日程：

密码

## 3.5、输出项

设置课程：

无

查看课程：

课程表信息，课程名和地点

新增日程：

无

查看日程：

密码、类型、内容、耗时、评分（日程优先级）

## 3.6、算法

//定义菜单类，用于接收用户的单击消息。

**public** **class** MainActivity **extends** Activity

//设计4个按钮用于接收按钮消息，并将它们初始化：

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*main\_activity*);

Button btn = (Button)findViewById(R.id.*btnSetSallybus*);

btn.setOnClickListener(onBtnSetSallybusClickListener);

btn = (Button)findViewById(R.id.*btnAddNote*);

btn.setOnClickListener(onBtnAddNoteClickListener);

btn = (Button)findViewById(R.id.*btnViewSallybus*);

btn.setOnClickListener(onBtnViewSallybusClickListener);

btn = (Button)findViewById(R.id.*btnViewNote*);

btn.setOnClickListener(onBtnViewNoteClickListener);

在OnClick函数中对响应进行处理：

1跳转设置课程表界面

**public** View.OnClickListener onBtnSetSallybusClickListener = **new** View.OnClickListener(){

@Override

**public** **void** onClick(View view) {

Intent intent=**new** Intent(MainActivity.**this**, ScheduleInsert.**class**);

MainActivity.**this**.startActivity(intent);

}

} ;

2跳转新增日程界面

**public** View.OnClickListener onBtnAddNoteClickListener = **new** View.OnClickListener(){

@Override

**public** **void** onClick(View view) {

//To change body of implemented methods use File | Settings | File Templates.

Intent intent=**new** Intent(MainActivity.**this**, DiaryNew.**class**);

MainActivity.**this**.startActivity(intent);

}

} ;

3跳转查看课程表界面

**public** View.OnClickListener onBtnViewSallybusClickListener = **new** View.OnClickListener(){

@Override

**public** **void** onClick(View view) {

Intent intent=**new** Intent(MainActivity.**this**, ScheduleShow.**class**);

MainActivity.**this**.startActivity(intent);

}

} ;

4跳转查看日程界面

具体在安全设计中介绍

## 3.7、接口

API函数

## 3.8、限制条件

无明显限制条件

# 4、界面设计说明

因概要设计中有具体页面效果图，这里只举一例说明

## 4.1、程序描述

主要功能：用于为用户提供可视化的操作界面，增加了程序的可读性。

可扩展性：能够适应应用要求的变化和修改，具有灵活的可扩充性。

## 4.2、功能

显示主界面、课程表界面、日程界面、日历界面、密码界面。

## 4.3、性能

能够快速的显示用户所需要的操作界面

## 4.4、输人项

无输入

## 4.5、输出项

操作界面

4.6、页面布局代码

<?xml version=*"1.0"* encoding=*"utf-8"*?>

<LinearLayout xmlns:android=*"http://schemas.android.com/apk/res/android"*

android:orientation=*"vertical"*

android:layout\_width=*"fill\_parent"*

android:layout\_height=*"fill\_parent"*

android:layout\_gravity=*"center\_vertical"*

>

<LinearLayout android:orientation=*"horizontal"*

android:layout\_width=*"fill\_parent"*

android:layout\_height=*"wrap\_content"*

android:layout\_weight=*"1"* android:layout\_gravity=*"center"* android:gravity=*"center"*>

<Button

android:layout\_width=*"120dp"*

android:layout\_height=*"120dp"*

android:text=*"设置课程表"*

android:id=*"@+id/btnSetSallybus"* android:layout\_gravity=*"center"*/>

<Button

android:layout\_width=*"120dp"*

android:layout\_height=*"120dp"*

android:text=*"查看课程表"*

android:id=*"@+id/btnViewSallybus"* android:layout\_gravity=*"center"*/>

</LinearLayout>

<LinearLayout android:orientation=*"horizontal"*

android:layout\_width=*"fill\_parent"*

android:layout\_height=*"wrap\_content"*

android:layout\_weight=*"1"* android:gravity=*"center"* android:layout\_gravity=*"center"*>

<Button

android:layout\_width=*"120dp"*

android:layout\_height=*"120dp"*

android:text=*"新增日程"*

android:id=*"@+id/btnAddNote"* android:layout\_gravity=*"center"*/>

<Button

android:layout\_width=*"120dp"*

android:layout\_height=*"120dp"*

android:text=*"查看日程"*

android:id=*"@+id/btnViewNote"* android:layout\_gravity=*"center"*/>

</LinearLayout>

</LinearLayout>

## 4.7、限制条件

无明显限制条件

# 5、安全设置设计说明

## 5.1、程序描述

主要功能：当用户点击查看日程时需输入安全密码才可查看信息，起到保护用户信息的功能。

可扩展性：能够适应应用要求的变化和修改，具有灵活的可扩充性。

## 5.2、功能

安全设置程序流程图：



图4 安全设置程序流程图图

5.3、性能

能够让用户输入密码，并保存进内部数据库。

## 5.4、输人项

用户输入密码。

## 5.5、输出项

无

## 5.6、算法

**public** View.OnClickListener onBtnViewNoteClickListener = **new** View.OnClickListener(){

@Override

**public** **void** onClick(View view) {

**if**(!Security.*Logined*)

{

LayoutInflater inflater = getLayoutInflater();

**final** View inputView = inflater.inflate(R.layout.*input\_string*,(ViewGroup)findViewById(R.id.*etInputString*));

**new** AlertDialog.Builder(MainActivity.**this**).setTitle("请输入查看日程密码")

.setView(inputView)

.setNeutralButton(R.string.*ok*, **new** DialogInterface.OnClickListener() {

@Override

**public** **void** onClick(DialogInterface dialogInterface, **int** i) {

EditText etPwd = (EditText) inputView.findViewById(R.id.*etInputString*);

String strInputPwd = etPwd.getText().toString();

SharedPreferences prefs = getSharedPreferences("user\_info",*MODE\_PRIVATE*);

String strPwd = prefs.getString("password","");

**if**(!strPwd.equals("") )

{

**if**(strPwd.equals(strInputPwd)) {

Security.*Logined* = **true**;

}

}

**else**{

Security.*Logined* = **true**;

}

**if**(!Security.*Logined*){

**new** AlertDialog.Builder(MainActivity.**this**)

.setTitle("提示")

.setMessage("密码不正确")

.setPositiveButton("确定", **null**)

.show();

**return**;

}

}

})

.show();

**return**;

}

**if**(!Security.*Logined*){

**new** AlertDialog.Builder(MainActivity.**this**)

.setTitle("提示")

.setMessage("密码不正确")

.setPositiveButton("确定", **null**)

.show();

**return**;

}

Intent intent=**new** Intent(MainActivity.**this**, CalendarActivity.**class**);

MainActivity.**this**.startActivity(intent);

}

} ;

## 5.7、接口

API函数

## 5.8、限制条件

无明显限制条件

## 5.9、尚未解决的问题

不能进行密码重置，当用户忘记密码时，只能内部查看。当查看日程时输入一次密码后，必须在此点击查看日程按钮才能查看日程。

# 6、其他需求设计

详见需求规格说明书。