娄底职业技术学院

《基于Android开发框架基础》

实训报告

|  |
| --- |
| 题目：基于Android计算器设计与实现 |

类型：☑产品设计 □工艺设计 □方案设计

|  |  |
| --- | --- |
| 学 号： | 201702050308 |
| 班 级： | 17软件3班 |
| 姓 名： | 龙哲学 |
| 指导老师： | 邓伟华 |

目录

[第1章 系统概述 3](#_Toc11230662)

[1.1 实训目的及要求 3](#_Toc11230663)

[1.1.1实训目的 3](#_Toc11230664)

[1.1.2要求 3](#_Toc11230665)

[1.2 前置准备工作 3](#_Toc11230666)

[1.3 开发环境 3](#_Toc11230667)

[1.4项目技术背景 4](#_Toc11230668)

[1.4.1 Java简介 4](#_Toc11230669)

[1.4.2 Android Studio简介 4](#_Toc11230670)

[1.4.3计算器的背景 4](#_Toc11230671)

[第2章 系统分析 5](#_Toc11230672)

[2.1 功能分析 5](#_Toc11230673)

[2.2 界面分析 5](#_Toc11230674)

[2.3 模块划分 5](#_Toc11230675)

[2.4进度安排 5](#_Toc11230676)

[第3章 系统设计 6](#_Toc11230677)

[3.1 布局一设计 6](#_Toc11230678)

[3.1.1 登陆界面设计 6](#_Toc11230679)

[3.1.2 登陆界面代码 7](#_Toc11230680)

[3.2布局二设计 9](#_Toc11230681)

[3.2.1计算界面设计 9](#_Toc11230682)

[3.2.2 计算界面代码 9](#_Toc11230683)

[3.3活动一设计 15](#_Toc11230684)

[3.3.1属性成员设计 15](#_Toc11230685)

[3.3.2方法设计 15](#_Toc11230686)

[3.4配置文件设计 16](#_Toc11230687)

[第4章 系统测试与部署 16](#_Toc11230688)

[4.1 项目开发文件夹 16](#_Toc11230689)

[4.2 系统运行界面 17](#_Toc11230690)

[第五章 实训总结 18](#_Toc11230691)

[参考文献 19](#_Toc11230692)

[致谢 20](#_Toc11230693)

# 第1章 系统概述

## 1.1 实训目的及要求

### 1.1.1实训目的

实训的是为了让我们好的掌握本学期的学习的成果，巩固所学知识。让大家更好的找到自己不足的地方，加以改正。让我们能更好的理解和熟悉Android移动开发这一门课程。

### 1.1.2要求

运用相关知识，查阅相关资料，编写一个简单的计算器，能够实现简单的基本的加减乘除以及登录界面，在计算器工作的时候利用键盘或者鼠标进行相应的操作。程序基本功能要求实现完整，并有简单的验证。在计算器运行中，输入数据时如果输入错误的情况，能够通过界面的C进行删除，并且重新输入正确的数据。

## 前置准备工作

1. 查找项目开发资料
2. 搭配环境
3. 在网上查找关于Android计算器的做法

## 1.3 开发环境

操作系统：windows7或windows10

开发工具：Android Studio 3.0

开发语言：Java

## 1.4项目技术背景

### 1.4.1 Java简介

java是一种高级的面向对象的程序设计语言 。

面向对象：java语言的基础。java将一切问题都看做对象与对象之间的交互，将对象抽象成方法与属性的集合。

分布性：包含操作分布性与数据分布性两个方面。操作分布性是指由多个主机共同完成一项功能，数据分布性是分布在多台主机上的数据当做一个完成的整体处理。

跨平台：java语言编写的应用程序，可以由一种平台迁移到另一种平台。

解释型：使用java语言编写的源码被转化为字节码，JVM将字节码翻译成机器码就可以被计算机执行。

安全性：java语言的底层设计可以有效避免非法操作。

健壮性：java提供了许多机制防止运行时出现严重错误，如编译时类型检查、异常处理。

多线程：java支持进程内部多个线程同时工作。

### 1.4.2 Android Studio简介

Android Studio是Google发布的用于Android的开发平台。系统基于 IntelliJ IDEA。类似 EclipseADT，Android Studio 提供了集成的 Android 开发工具用于开发和调试。Android Studio的开发环境和模式更加的丰富和便捷，能够支持多种语言，还可以为开发者提供测试工具和各种数据分析。开发者可以在编写程序的同时看到自己的应用在不同尺寸屏幕中的样子。

谷歌对开发者控制台进行了改进，增加了五个新的功能，包括优化小贴士、应用翻译服务、推荐跟踪、营收曲线图、用版测试和阶段性展示。

### 1.4.3计算器的背景

随着科技的发展，信息技术进入人类的生活，遍布各个领域，大型的计算是人脑不可以胜任的，所以这时科学计算器显得尤为重要。以前的计算器只能实现加减乘除四则运算，并且只能进行十进制的运算，遇到很长的式子，人们只能一步步输入很多，并且没有之前输入的提示，显示栏里只有结果。这样对整个计算的过程都没有显示，功能很单一，而且大多数计算器只有一些简单的四则计算功能。传统的计算器只能满足很小的学生计算并且使用。随着科技的进步，社会的发展，十进制不能满足人们的计算，所以我们需要二进制八进制的转化，传统的计算器不在被人们广泛需要，给人们带来了不方便。然而我们的这个项目是面对Android手机应用的开发，使人们更加的方便使用、便捷。

# 第2章 系统分析

## 2.1 功能分析

能够实现人们普遍的加法、减法、乘法、除法。能够精确的算出结果。还在界面前加了一个登陆界面，实现登陆模块。

## 2.2 界面分析

整体界面简单明了，美观，使人们更加方便使用。更加直观的了解计算器，还在其中添加了登陆界面。

## 2.3 模块划分

|  |
| --- |
|  |

|  |
| --- |
|  |

1. 登陆模块
2. 计算模块

## 2.4进度安排

选择项目，查找相关资料 6月1号

创建项目，登陆界面设计 6月2号

功能设计 6月4号

实现主要功能 6月5号

修改错误 6月6号

实训总结 6月7号

# 第3章 系统设计

## 3.1 布局一设计

3.1.1 登陆界面设计

以下是登陆界面的功能实现页面：



3.1.2 登陆界面代码

以下是登陆界面主要代码：

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="账号："

android:textColor="@color/colorheise"

android:textSize="30dp" />

<EditText

android:id="@+id/zh"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textColor="@color/colorheise"

android:textSize="30dp" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="密码："

android:textColor="@color/colorheise"

android:textSize="30dp" />

<EditText

android:id="@+id/mm"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

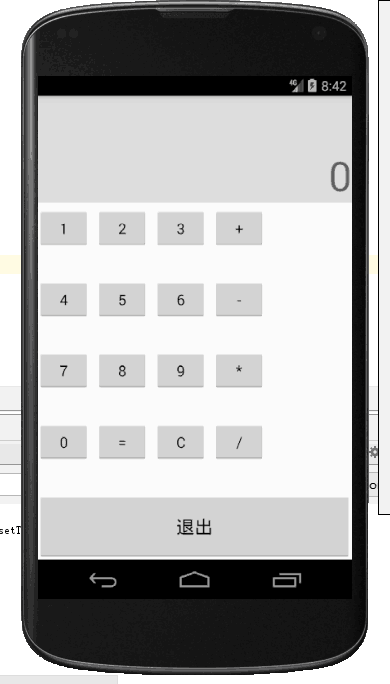
android:textColor="@color/colorheise"

android:textSize="30dp" />

</LinearLayout>

## 3.2布局二设计

### 3.2.1计算界面设计



### 计算界面代码

主要实现了人们生活中的加减乘除，方便大家计算使用，主要代码如下

<Button

android:id="@+id/button1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="1"

android:gravity="center"

></Button>

<Button

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="2"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="3"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button11"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="+"

android:onClick="jishu"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button4"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="4"

android:gravity="center"

></Button>

<Button

android:id="@+id/button5"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="5"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button6"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="6"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button12"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="-"

android:onClick="jishu"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button7"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="7"

android:gravity="center"

></Button>

<Button

android:id="@+id/button8"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="8"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button9"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="9"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button13"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="\*"

android:onClick="jishu"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button10"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="0"

android:gravity="center"

></Button>

<Button

android:id="@+id/button15"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="="

android:onClick="jishu"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button16"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="C"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

<Button

android:id="@+id/button14"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="/"

android:onClick="jishu"

android:layout\_marginLeft="15px"

android:gravity="center"

></Button>

## 3.3活动一设计

### 3.3.1属性成员设计

public class

protected void onCreate

public void onClick

### 3.3.2方法设计

public void onClick(View v) {

if (zh.getText().toString().equals("user") && mm.getText().toString().equals("123456")) {

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

startActivity(intent);

} else {

EditText ts = (EditText) findViewById(R.id.ts);

ts.setText("账号或密码错误");

}

}

## 3.4配置文件设计

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

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

package="com.example.calculator">

<application

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:roundIcon="@mipmap/ic\_launcher\_round"

android:supportsRtl="true"

android:theme="@style/AppTheme">

<activity android:name=".MainActivity">

</activity>

<activity android:name=".Login">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

</manifest>

# 第4章 系统测试与部署

## 4.1 项目开发文件夹

Calulator\app\src\main\res\layout\ activity\_login.xml

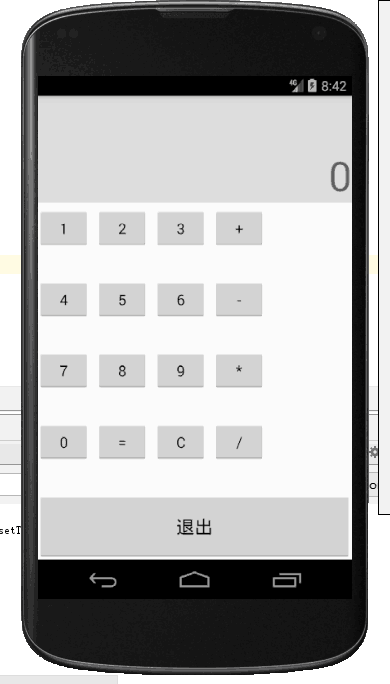
Calulator\app\src\main\res\layout\ activity\_main.xml

Calulator\app\src\main\java\com\example\calculator\Login.java

Calulator\app\src\main\java\com\example\calculator\MainActivity.java

## 4.2 系统运行界面

下面是系统运行之后得到结果的界面：



# 第五章 实训总结

为期一周的android数据库实训已经结束了，从这一周中，有了很多的感悟。从学到和掌握到的东西来说，在书本上学到的东西非常不牢固，然而实训真的让我受益匪浅!实训第一天到教室时，看到老师给我们讲实训的内容与要求，然后告诉我们一些要完成的任务与作业，然后根据实训的内容与要求授课，让我们从实践中去体会所学的知识。虽然这样的实训才一周，但是这样的一周实训可以相当于是一个学期的知识，在实训中要充分的利用自己在网络上和书上查找解决方法，以及询问老师。

在实训的过程中了解到了很多的代码是如何使用，也学会了一些没出现在课堂上的代码，正所谓是师傅领进门，修行靠个人，在实训过程中充分的培养了自己的自学能力以及同学之间的互帮互助，互相解决问题能力，同时也培养了设计能力，以及了解代码的规范性设计。

这次实训，不仅让我们学到了许多有关android的知识和编程的知识，老师也给我们讲了很多内容，让我们在不同的角度了解这个专业。让我们在今后的学习中更有动力的充实自己，增加自己的知识面和锻炼自己各方面能力。

# 参考文献

【1】《Android移动开发基础案例教程》黑马程序员编著.北京:人民邮电出版社.2017

# 致谢

实训也接近尾声了，我们的实训也圆满完成了。经历了一周的努力，终于完成了计算器的设计与实现。感谢邓老师平时在课堂的幸勤付出，以及在实训期间的耐心指导，帮我们解决了一个又一个的难题，让我们最终实现了自己的项目。最后非常感谢邓老师在实训期间耐心的教导和帮助。