HIT学生管理系统

设计方案

项目成员： 1163710223-张凌宇

1163710214-刘文佳

1163710219-张浚哲

2018年11月

目录

[1、项目概述 3](#_Toc531201379)

[1.1 项目的目的和意义 3](#_Toc531201380)

[1.2 现状分析 3](#_Toc531201381)

[2、主要业务流程分析 3](#_Toc531201382)

[3、需求分析 3](#_Toc531201383)

[4、系统功能设计 3](#_Toc531201384)

[5、系统UI设计 3](#_Toc531201385)

[6、系统数据库设计 3](#_Toc531201386)

[7、系统体系结构设计 4](#_Toc531201387)

[8、统一接口设计 4](#_Toc531201388)

[9、其它需要说明的设计要点 4](#_Toc531201389)

[10、详细设计与成果 4](#_Toc531201390)

[11、项目体会 4](#_Toc531201391)

# 1、项目概述

## 1.1 项目的目的和意义

建立学生成绩管理系统，利用手机对学生成绩进行管理，进一步提高现代化水平。使学生能够快速、准确地输入、修改和查询自己的成绩，更好地了解自己的学业成绩，并方便对教师的管理。让学生更快地查询自己的学业成绩。教师可以更快更好地了解学生的学习情况，从而进一步提高学校的教学质量。

## 1.2 现状分析

学校对学生成绩的管理曾经还停留在使用手工操作上。随着学校规模的扩大，学生人数逐年增加。随着学生成绩管理数据量的不断增加，一些学校不得不依靠不断增加的人力物力来进行学生成绩管理。但是，人工管理存在效率低、易出错、信息检索速度慢、难以为学校管理提供决策信息等缺点。学校不能充分掌握每个学生的学习情况，这不利于学校的教学工作，学校的教学质量一直停滞不前，所以我只想开发一个适用的绩效管理体系，使学生能够充分掌握学业的成绩和课程，以便于学生了解情况。

# 2、主要业务流程分析

成绩管理系统可以输入课程和成绩数据，并可以完成添加、修改、删除等查询，并根据各种条件基本满足师生的需要。成绩管理系统主要包括学生的基本信息、年龄、电话等方面的信息，内容比较简单。成绩管理系统主要用于学生成绩信息的输入，具有课程注册、成绩查询、学生信息注册等功能。一般来说，它具有编辑、查询、学生成绩管理等功能。

学生端登陆该系统后，进入个人信息页面可以查看自己的个人信息，进入课程管理页面可以对已选课程的课程成绩查询，显示其学号、科目名称和成绩，进入课程论坛页面可以进入现有话题查看和跟帖，也可以创建新的讨论话题。

教师端登录系统后可以进入个人信息页面可以查看自己的个人信息，还可以先选择已开放课程，之后进行成绩录入、成绩修改、成绩查询、成绩分析、删除课程的操作，也可以增加新开放的课程。成绩录入先选择需要录入的人数，再添加其学号与成绩。成绩修改先输入需要修改的成绩的学生学号进行查询，然后输入其姓名，科目，任课教师，成绩，然后提交修改。成绩查询可查询已选此门课程的学生的学号、姓名和成绩。成绩分析可以选择分数比例分析图查看饼状统计图，或者选择分段人数分析图查看条状统计图。删除课程就会删除此门开放课程。

# 3、需求分析

功能需求：

(1)成绩管理系统可以提供一个完善的登陆界面，使学生或教师都可以通过此窗口登陆。

(2)成绩管理系统可以提供完善的学生信息和课程信息管理界面，学生或管理员可以查看和管理学生信息与课程信息。

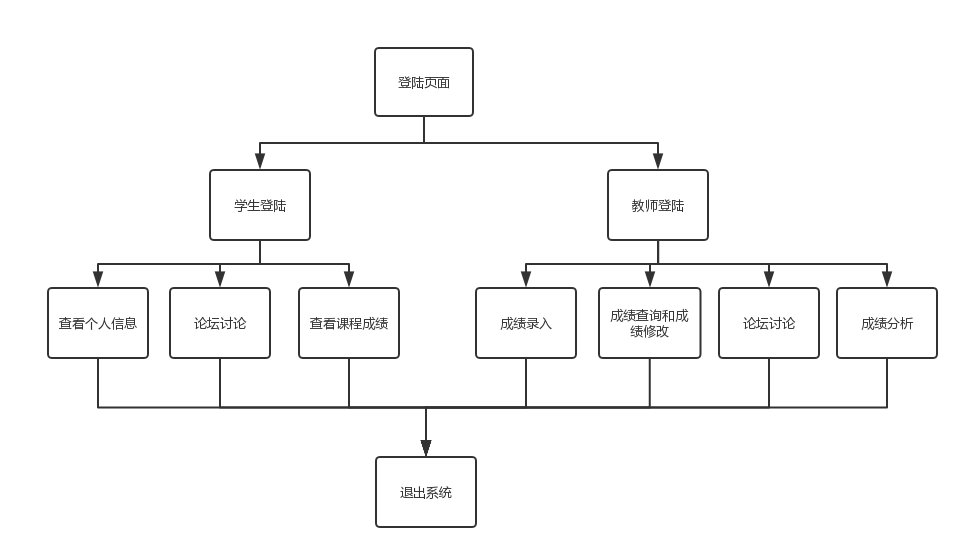
(3)成绩管理系统可以提供完善的课程信息管理界面，教师或管理员可以查看和管理课程信息，进行成绩录入、成绩修改、成绩查询、成绩分析。

(4)成绩管理系统有一个后端管理页面，可以作为管理员登陆查看与修改学生信息，教师信息，课程信息和论坛信息。

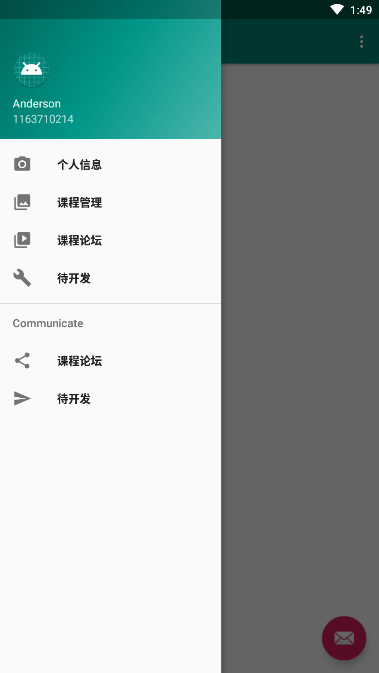
非功能需求：

1. 登陆时采用的用户名 和密码都采用MD5算法保护，再存入数据库存储。
2. 该系统能够在android4.0.13以上版本的安卓系统上正常运行。

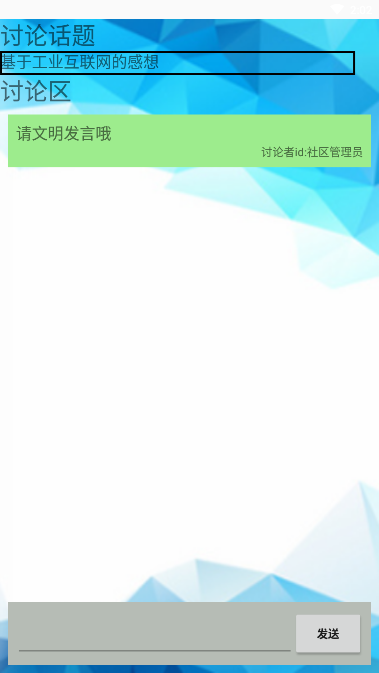
# 4、系统功能设计

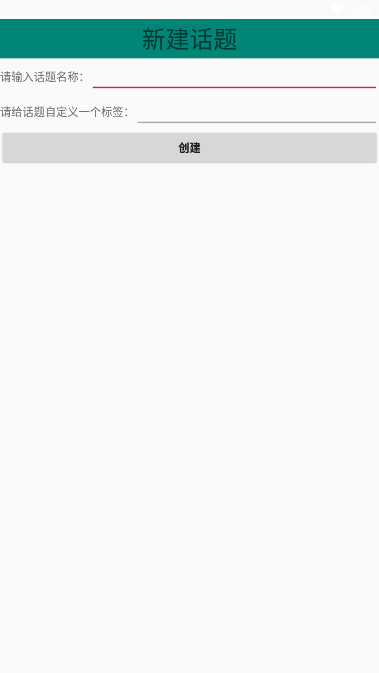


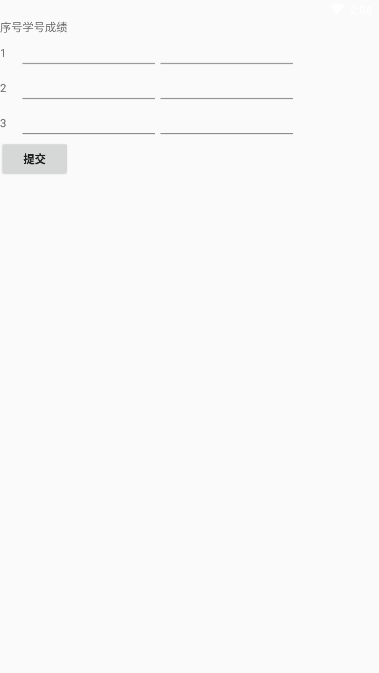
# 5、系统UI设计



# 6、系统数据库设计

/\*

服务器hostname： www.wuyunian.tk

密码： wuyunian1998

学号：

8位数是老师，10位数是本科生

\*/

/\*学生

学号 密码 姓名 入学年份 手机号

\*/

drop table if exists `student\_id`;

CREATE TABLE IF NOT EXISTS `student\_info`(

`id` VARCHAR(100) NOT NULL,

`password` VARCHAR(100) NOT NULL,

`name` VARCHAR(100) NOT NULL,

`year` int not null,

`phonenum` varchar(11) not null,

PRIMARY KEY ( `id` )

)ENGINE=InnoDB DEFAULT CHARSET=utf8;

/\*老师

职工号 密码 姓名

\*/

drop table if exists `teacher\_info`;

CREATE TABLE IF NOT EXISTS `teacher\_info`(

`id` VARCHAR(100) NOT NULL,

`password` VARCHAR(100) NOT NULL,

`name` VARCHAR(100) NOT NULL,

`phonenum` varchar(11) not null,

PRIMARY KEY ( `id` )

)ENGINE=InnoDB DEFAULT CHARSET=utf8;

/\*课程

课程编号 课程名称 任课教师职工号 学生学号 学生成绩

\*/

drop table if exists `subject`;

CREATE TABLE IF NOT EXISTS `subject`(

`subject\_id` VARCHAR(100) NOT NULL,

`subject\_name` VARCHAR(100) NOT NULL,

`teacher\_id` VARCHAR(100) NOT NULL,

`student\_id` VARCHAR(100) ,

`score` double

)ENGINE=InnoDB DEFAULT CHARSET=utf8;

/\*讨论话题

讨论话题编号 讨论的话题 教师职工号 学生学号 老师发言 学生发言

\*/

drop table if exists `discussion`;

CREATE TABLE IF NOT EXISTS `discussion`(

`discussion\_id` int NOT NULL auto\_increment,

`topic` VARCHAR(100) NOT NULL,

`name` VARCHAR(100) NOT NULL,

`teacher\_id` VARCHAR(100) ,

`student\_id` VARCHAR(100) ,

`teacher\_words` VARCHAR(200) ,

`student\_words` VARCHAR(200) ,

PRIMARY KEY ( `discussion\_id` )

)ENGINE=InnoDB DEFAULT CHARSET=utf8;

/\*插入学生数据 \*/

INSERT INTO

student\_info (id, password, name,year,phonenum)

VALUES("1163710214", "123456", "Anderson",2018,18846135525);

INSERT INTO

student\_info (id, password, name,year,phonenum)

VALUES("1163710215", "123456", "刘文佳",2018,15512931688);

INSERT INTO

student\_info (id, password, name,year,phonenum)

VALUES("1152625346", "123456", "佳佳大魔王",2018,15512931688);

INSERT INTO

student\_info (id, password, name,year,phonenum)

VALUES("1554651223", "123456", "克格勃佳佳",1976,15512931688);

/\*插入老师数据\*/

insert into teacher\_info(id,password,name,phonenum)

values("20181234","123456","奥巴马","18846135525");

insert into teacher\_info(id,password,name,phonenum)

values("19982356","123456","普京","11913582155");

/\*插入课程数据\*/

insert into subject(subject\_id,subject\_name,teacher\_id,student\_id,score)

values("cs001","Data constructure And Algorithm","20181234","1163710214",90.6);

insert into subject(subject\_id,subject\_name,teacher\_id,student\_id,score)

values("cs001","Data constructure And Algorithm","20181234","1163710214",90.6);

insert into subject(subject\_id,subject\_name,teacher\_id,student\_id,score)

values("cs002","Software Constructure","19982356","1163710214",85.9);

insert into subject(subject\_id,subject\_name,teacher\_id,student\_id,score)

values("math001","gaoshu","19982356","1163710215",85.9);

/\*插入讨论数据\*/

insert into discussion(topic,name,teacher\_id,student\_id,teacher\_words,student\_words)

values("工业互联网","基于工业互联网的感想","20181234","1163710214","我觉得不行","我觉得ok");

insert into discussion(topic,name,teacher\_id,student\_id,teacher\_words,student\_words)

values("测试大数据导论","基于hadoop进行数据分析","20181234","1163710214","我觉得不行","我觉得ok");

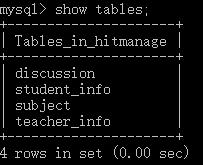
insert into discussion(topic,name,teacher\_id,student\_id,teacher\_words,student\_words)

values("人工智能","价值一亿美元的人工智能代码竟然是用java写的","19982356","1163710214","我觉得不行","我觉得ok");

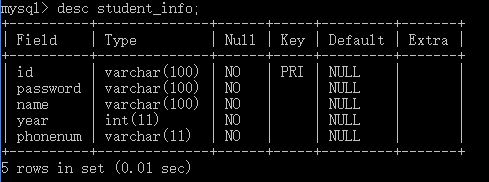
insert into discussion(topic,name,teacher\_id,student\_id,teacher\_words,student\_words)

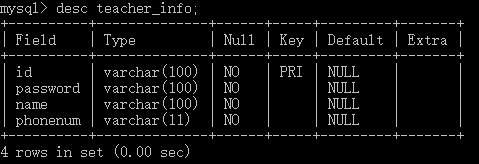
values("人工智能","AI","19982356","1163710214","我觉得不行","我觉得ok");

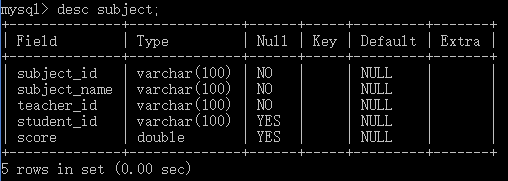
主要的表：

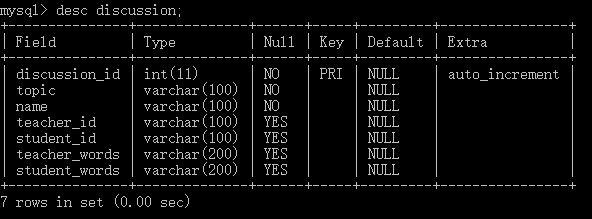


各个表的结构：





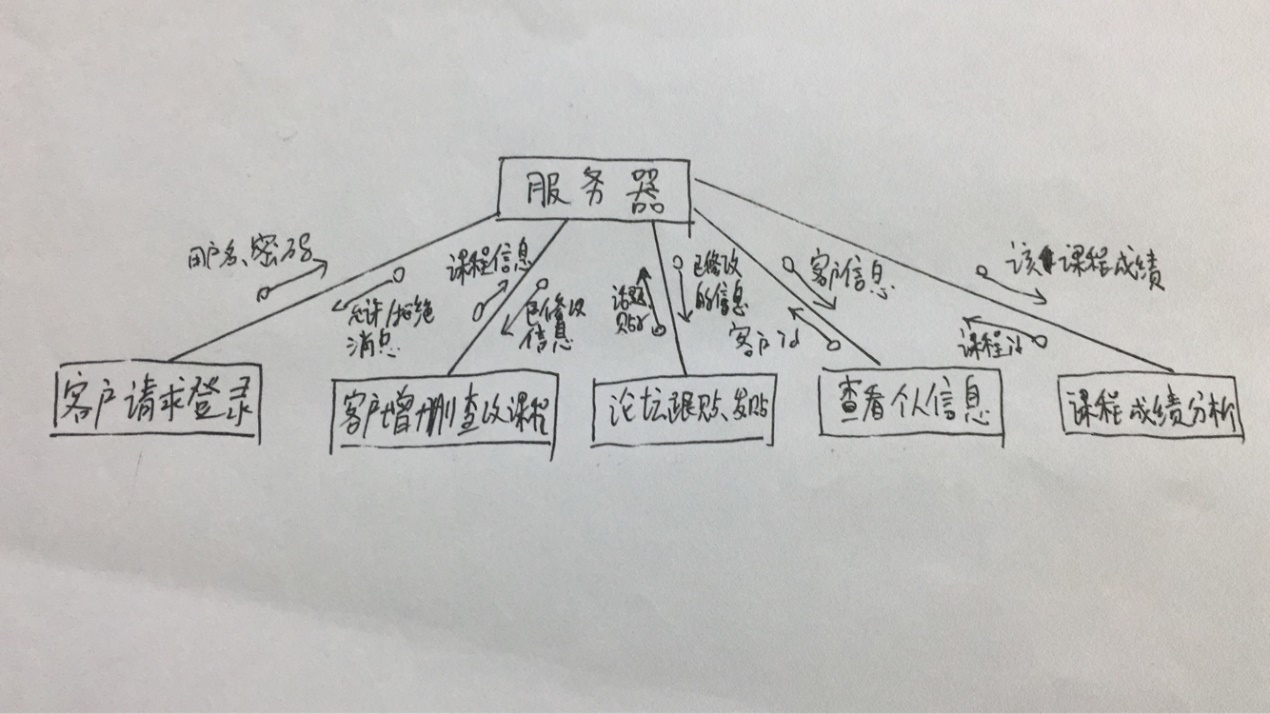




# 7、系统体系结构设计

后端采用SSM框架，采用spring+springmvc+mybatis的技术方案，实现后端快速开发

前端基于Navigation Drawer Activity和LoginActivity两个基础Activity，使用sharedpreferences，HTTP，本地化存储等技术进行开发。



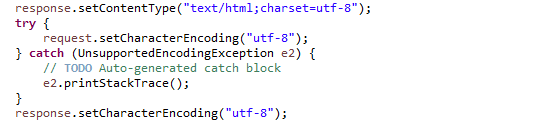
# 8、统一接口设计

| **序号** | **url** | **方式** | **返回值** | **返回格式** | **备注** |
| --- | --- | --- | --- | --- | --- |
| 1 | <http://localhost:8080/HITManage/login?id=1163710215&password=123456> | post | true or false | json | 登录验证 |
| 2 | <http://localhost:8080/HITManage/reqstumsg?id=1163710215> | get | student\_info | json | 请求学生信息 |
| 3 | <http://localhost:8080/HITManage/reqteamsg?id=20160101> | get | teacher\_info | json | 请求老师信息 |
| 4 | <http://localhost:8080/HITManage/reqallsubject> | get | Arraylist<String> | json | 请求所有课程名称 |
| 5 | <http://localhost:8080/HITManage/reqsubject?subjectname=gaoshu&student_id=1163710215> | get | subject | json | 请求某学生某课程的课程信息 |
| 6 | <http://localhost:8080/HITManage/reqsubscore?id=1> | get | Arraylist<Subject\_info> | json | 请求某课程所有的成绩 |
| 7 | <http://localhost:8080/HITManage/reqalldis> | get | Arraylist<String> | json | 请求所有讨论名称 |
| 8 | <http://localhost:8080/HITManage/reqdis?disname=>**\*\*\*\*\*** | get | Arraylist<discussion> | json | 请求某讨论所有内容 |
| 9 | <http://localhost:8080/HITManage/addStudent?id=116&password&name&year&phonenum> | post | ? | ? | 添加学生账号 |
| 10 | <http://localhost:8080/HITManage/addScore> | post | 格式：Arraylist<subject> | json | 添加学生成绩 |
| 11 | <http://localhost:8080/HITManage/addDiscussion> | post | 格式：Arraylist<discussion> | json | 添加讨论信息 |
| 12 | <http://localhost:8080/HITManage/reqSubid?subname=gaoshu> | get | string | json | 请求某课程的编号 |
| 13 | <http://localhost:8080/HITManage/reqSubByTeaId?teacherid=1163710215> | get |  | json |  |
| 14 | <http://localhost:8080/HITManage/newDiscussion?topic=>人工智能&name=啥玩意啊 | post | true or false | json | 添加讨论圈子 |
| 14 | <http://localhost:8080/HITManage/addDiscussion>? | post | discussion | json | 添加某话题下的讨论内容 |
| 15 | <http://localhost:8080/HITManage/delDiscussion>? | post | diccussion | json | 删除某条讨论内容（先不实现） |
| 16 | <http://localhost:8080/HITManage/reqSubByStuId?studentid=1163710215> | get | status boolean data List<Subject> | json | 请求某学生的课程信息 |
| 17 | <http://localhost:8080/HITManage/modifyPassword?id=1163710215&password=123456> | post | staues true | json | 修改密码 |

| **序号** | **url** | **方式** | **返回值** | **返回格式** | **备注** |
| --- | --- | --- | --- | --- | --- |
| 12 | <http://localhost:8080/HITManage/reqSubjectByAndroid?subjectname=gaoshu&student_id=1163710215> | get | subject | json | 请求某学生某课程的课程信息 |
| 13 | <http://localhost:8080/HITManage/addStudentByAndroid?id=116&password=123456&name=Anderson&year=3&phonenum=18846135525> | post | status--boolean msg--String | json | 添加学生账号 |

# 9、其它需要说明的设计要点

中文编码问题：由于在前后端传输数据时，若传输中文数据容易变为乱码，因此在前后端进行中文传输时，函数要在开始前进行编码统一。（如下图）



# 10、详细设计与成果

先介绍android前端的设计思路，首先设计的是login活动，该活动登陆时读取用户输入的信息，在本地进行加密，加密后送至服务器进行匹配，并返回是否允许登陆的信息。再次登陆时利用sharedpreference存储历史登陆信息。第二个是注册活动，该活动还允许学生注册新的账户与密码，发送至服务器并保存。第三个是修改密码活动，输入用户名，新密码旧密码，以发送至服务器并修改。第四个是显示主页面的活动，通过侧边栏接入更多的功能。第五个是个人信息活动，也是通过请求后端数据库以活动该用户的信息以显示。第六个是学生端的课程信息活动，该活动可以查询自己所有已选课程的成绩，需要通过添加适配器动态地显示多条信息。第七个是教师端的课程信息活动，教师可以增删查改各门课程的成绩，同样是请求后端对数据库进行增删查改，教师端还有一个成绩分析的活动，该活动根据对后端请求所得的特定课程的所有成绩信息进行分段，同样该活动也使用了适配器，然后分别用饼状图和柱状图具体的展示给教师。第八个是论坛讨论功能，先使用一个适配器展示所有的讨论话题，点击一个话题后利用第二个适配器展示所有该话题下的所有讨论内容。

然后是web管理段的设计思路，该页面相当于将所有存在数据的内容，包括学生信息，教师信息，课程信息，论坛信息，一并的显示于页面上，并且可以自由的增删查改，该页面当然也与后端数据库连接。

后端采用ssm框架。在后端中有简单的页面：老师界面、学生界面、讨论区界面、课程界面，每个界面实现了基本的增删改查功能。

在框架中，也贯穿了软件设计过程中的MVC思想。model体现在pojo这个包中，这个包实现了讨论、老师、学生、课程这几个基本类。views体现在web界面的jsp中。control体现在controller这个包中。

spring利用控制反转和依赖注入，将对象之间的依赖关系交由spring进行控制，避免硬编码所造成的过度程序耦合。

springMVC实现了利用注解简化了配置，实现可视化

mybatis与数据库进行连接，将数据库操作放在xml文件中，避免出现在java代码中，极大地降低了数据库操作语言的书写过程和连接等过程。

在后端接受来自前端jsp或者android传来的get或者post请求，对请求中的内容进行解析，如json数据、String类型等，调用数据库操作，最后将得到的结果、状态信息等封装到json中返回给请求方。

# 11、项目体会

了解到了如何将SSM框架应用于后端的开发，在与他人合作编写代码时，要提前商量并确定下来一定的书写规范，可以保证代码的整洁度，也可以为后续的整合以及修改工作提供便利。在编写后端逻辑功能时，要时刻关注前端到后端再到数据库的数值传输，以此来确保模块功能的实现，同时减少代码编写过程中产生的bug。简而言之，代码的书写思路必须要清晰。在android 最初学习阶段，通常会遇到很多问题，Andorid 学习过程中知识点繁多，难以熟练的应用，不经过很长一段时间很难以熟练地运用自己所学的知识。其实Andorid 并不是很难，无非就是布局，Activity 中获取布局，对布局进行传值操作。并且学习Android 开发 fragment ViewPager 适配器是必不可少的，在任意一款app 中都有所应用。