**题目：****Android平台医院收费管理系统程序的应用开发**

**The Design and Development of the Android-based Hospital Application**

**Android平台医院收费管理系统程序的应用开发**

**摘 要：**（200字左右）

医院收费管理系统是一个基于Android Stduio来实现的安卓App系统。

医院门诊收费系统（Hospital Information System简称HIS）是一门容医学、信息、管理、计算机等多种学科为一体的边缘科学，在发达国家已经得到了广泛的应用，并创造了良好的社会效益和经济效益。医院门诊收费系统是现代化医院运营的必要技术支撑和基础设施，实现医院门诊收费系统的目的就是为了以更现代化、科学化、规范化的手段来加强医院的管理，提高医院的工作效率，改进医疗质量，从而树立现代医院的新形象，这也是未来医院发展的必然方向。

**关键词：**医院管理；数据库；Android；

**目录**

[1. 概述 ４](#_Toc43376853)

[（一） 任务和目标 ４](#_Toc43376854)

[（二） 需求概述 ４](#_Toc43376855)

[（三） 运行环境 ４](#_Toc43376856)

[（四） 系统项目简介 ４](#_Toc43376857)

[2. 系统整体设计 ９](#_Toc43376858)

[（一） 功能模块划分及说明 ９](#_Toc43376859)

[3. 模型层设计 １０](#_Toc43376860)

[4. 系统详细设计 １１](#_Toc43376861)

[（一） 登陆注册模块 １１](#_Toc43376862)

[（二） 会员管理模块 １４](#_Toc43376863)

[（三） 部门管理模块 １７](#_Toc43376864)

[（四） 医疗项目管理模块 ２０](#_Toc43376865)

[（五） 订单管理模块 ２４](#_Toc43376866)

[总结 ２７](#_Toc43376867)

1. **概述**
   1. 任务和目标
   2. 需求概述

对所开发软件的概要描述, 包括主要的业务需求、输入、输出、主要功能、性能等。

* 1. 运行环境
  2. 系统项目简介

1. 项目概况

* 工程名称：Hospital
* 所用架构：Android Studio project-module架构
* 所用技术：Sqlite数据库技术,Android SDK,XML,JAVA语言,虚拟机
* 项目依赖：Java,SQLite,Android Stduio
* 权限要求：管理员,会员

1. 目录结构

* 视图层文件：

所有的界面布局对应的xml文件说明。

例如：

account\_register.xml: 用户注册界面

dep\_fragment.xml: 用户医院fragment界面

department\_item.xml: 管理界面部门item

department\_save.xml: 部门保存界面

login\_layout.xml: 登陆界面

manage\_main.xml: 管理主界面

meb\_fragment.xml: 用户会员fragment界面

member\_item.xml: 管理界面会员item

member\_save.xml: 会员保存界面

myspinner: spinner界面

order\_fragment.xml: 订单fragment界面

order\_item.xml: 管理界面订单item

order\_save.xml: 订单保存界面

proj\_fragmen.xml: 项目fragment界面

project\_item.xml: 管理界面项目item

project\_save.xml: 项目保存界面

user\_dep\_item.xml: 用户界面部门item

user\_main.xml: 用户主界面,用于存放framgent

user\_order\_item.xml: 登陆界面布局

user\_project\_item.xml: 用户界面项目item

user\_update\_pwd.xml: 用户界面修改密码

* 模型层文件：

业务模型相关的java类说明。数据库有关的java类也属于这一层。

Account.java: 账户实体

Member.java: 会员实体

Doctor.java: 医生实体

Project.java: 项目实体

Order.java: 订单实体

Account.java: 账户持久层

Member.java: 会员持久层

Doctor.java: 医生持久层

Project.java: 项目持久层

Order.java: 订单持久层

HospitalSQLiteOpenHelper.java: 数据库helper类

* 控制层文件：

模型层除外的java文件，包括Activity，Service, Fragment等。如果有一些是工具类Util类型的java，请放在工具类文件中说明。

DepartmentAdapter:管理部门界面recyclerview适配器

MemberAdapter:管理会员界面recyclerview适配器

OrderAdapter:管理订单界面recyclerview适配器

ProjectAdapter:管理项目界面recyclerview适配器

UserDepAdapter:用户部门界面recyclerview适配器

UserOrderAdapter:用户订单界面recyclerview适配器

UserProjAdapter:用户项目界面recyclerview适配器

DepFragment:管理部门界面recyclerview适配器

MemberAdapter:管理会员界面recyclerview适配器

OrderAdapter:管理订单界面recyclerview适配器

ProjectAdapter:管理项目界面recyclerview适配器

* 工具类文件：

如果没有工具类，就删除这一部分。

BasicTimesUtils.java:用于存放时间格式及工具的类

GobalVar.java:用于存放登陆用户的类

MyTools.java:用于存放MD5,UUID的相关生成工具

RecycleViewDivider.java:用于存放recyclerview分割线等数据

* 资源文件：

所有其他资源文件说明。

手机屏幕截图

描述已自动生成

以上为项目内用到的图片,图标及控件背景

1. **系统整体设计**
   1. **功能模块划分及说明**

例如：

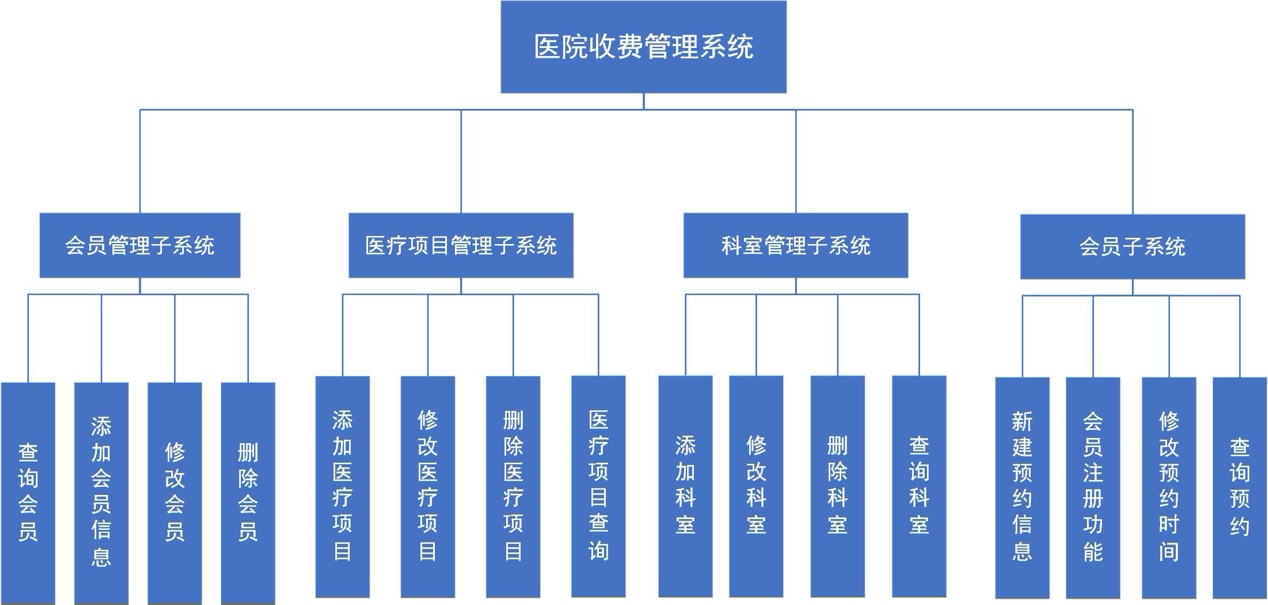


图1 功能模块图

管理员可控制会员,医疗项目,科室订单等模块

会员可进行个人信息修改,注册,预约等操作

1. **模型层设计**

**模型层Java类说明。数据库有关的表格等等设计也在此说明。**

表2.3.1 账户表

|  |  |  |  |
| --- | --- | --- | --- |
| ***列名*** | ***数据类型*** | ***可否为空*** | ***说明*** |
| userID | varchar | not null | 账号 |
| Pwd | varchar | not null | 密码 |
| Kind | int | not null | 用户级别 |

表2.3.2 会员表

|  |  |  |  |
| --- | --- | --- | --- |
| ***列名*** | ***数据类型*** | ***可否为空*** | ***说明*** |
| mebID | Varchar | not null | 就诊卡号 |
| mebName | Varchar | not null | 姓名 |
| sex | Char |  | 性别 |
| phone | Varchar |  | 联系电话 |
| resdate | Datetime | not null | 注册日期 |

表2.3.3 科室表

|  |  |  |  |
| --- | --- | --- | --- |
| ***列名*** | ***数据类型*** | ***可否为空*** | ***说明*** |
| depID | Varchar | not null | 科室ID |
| depName | Varchar | not null | 科室名 |
| phone | Varchar |  | 科室联系电话 |

表2.3.4 医疗项目表

|  |  |  |  |
| --- | --- | --- | --- |
| ***列名*** | ***数据类型*** | ***可否为空*** | ***说明*** |
| projID | Varchar | not null | 项目ID |
| projName | Varchar | not null | 项目名 |
| depID | Varchar |  | 所属科室 |
| unit | Varchar |  | 单位 |
| price | Money | not null | 项目价格 |
| notes | Varchar |  | 备注 |

表2.3.5就诊订单表

|  |  |  |  |
| --- | --- | --- | --- |
| ***列名*** | ***数据类型*** | ***可否为空*** | ***说明*** |
| orderID | Varchar | not null | 订单流水号 |
| mebID | Varchar | not null | 就诊卡号 |
| projID | Varchar | not null | 项目ID |
| num | Int | not null | 数量 |
| appotime | datetime | not null | 预约时间 |
| state | varchar | Not null | 订单状态 |

1. **系统详细设计**
   1. **登陆注册模块**
2. 界面设计

以下分别是登陆界面,注册界面,用户,管理登陆主界面组件树

手机屏幕的截图

描述已自动生成 手机屏幕截图

描述已自动生成

手机屏幕的截图

描述已自动生成手机屏幕截图

描述已自动生成

1. IPO图

****

**图2 登陆注册模块IPO**

1. 主要算法

关于登陆的：

SharedPreferences是Android平台上一个轻量级的存储类。在此活动中我们用来缓存用户信息和登录状态，以此来改善用户体验。

首先从SharedPreferences读取用户信息，检查有没有已经登陆过的账号密码，有则自动填充登录用户名和密码；

MD5加密：

用户注册修改登录均进行md5加密

MD5是一个安全的散列算法，输入两个不同的明文不会得到相同的输出值，根据输出值，不能得到原始的明文，即其过程不可逆；所以要解密MD5没有现成的算法，只能用穷举法，把可能出现的明文，用MD5算法散列之后，把得到的散列值和原始的数据形成一个一对一的映射表，通过比在表中比破解密码MD5算法散列值，通过匹配从映射表中找出破解密码所对应的原始明文

数据库事务:

AccountDao中注册账户和删除账户均为修改两张表的内容，应使用事务进行数据库修改，以防止只有单张表进行了修改

数据库事务是访问并可能操作各种数据项的一个数据库操作序列，这些操作要么全部执行,要么全部不执行，是一个不可分割的工作单位。事务由事务开始与事务结束之间执行的全部数据库操作组成。

1. 运行效果

手机屏幕截图

描述已自动生成手机屏幕截图

描述已自动生成

手机屏幕截图

描述已自动生成手机屏幕的截图

描述已自动生成

* 1. **会员管理模块**

1. 界面设计

以下分别是会员item,会员fragment,修改信息界面,修改密码界面组件树

手机屏幕截图

描述已自动生成 手机屏幕截图

描述已自动生成

手机屏幕截图

描述已自动生成

1. IPO图

****

**图3会员模块IPO**

1. 主要算法

关于会员ID获取：

使用类Gobalvar的静态变量对登录账户的ID和类别进行存储

静态变量：Java 中被 static 修饰的成员称为静态成员或类成员。它属于整个类所有，而不是某个对象所有，即被类的所有对象所共享。静态成员可以使用类名直接访问，也可以使用对象名进行访问。

Recyclerview：

RecyclerView是Android一个更强大的控件,其不仅可以实现和ListView同样的效果,还有优化了ListView中的各种不足。其可以实现数据纵向滚动,也可以实现横向滚动(ListView做不到横向滚动)。

新建item布局用于表示单个视图的内容，新建适配器绑定recyclerview，并在holder中为item内容进行对应赋值，增加监听器以实现相关功能。达到列表的展示。

1. 运行效果

手机屏幕截图

描述已自动生成 手机屏幕的截图

描述已自动生成

 手机屏幕截图

描述已自动生成

* 1. **部门管理模块**

1. 界面设计

以下分别是用户部门item,部门item,部门保存界面组件树

手机屏幕的截图

描述已自动生成手机屏幕截图

描述已自动生成

手机屏幕的截图

描述已自动生成

1. IPO图

****

**图4部门模块IPO**

1. 主要算法

隐式intent

不是像显式的那样直接指定需要调用的Activity，隐式不明确指定启动哪个Activity，而是设置Action、Data、Category，让系统来筛选出合适的Activity。

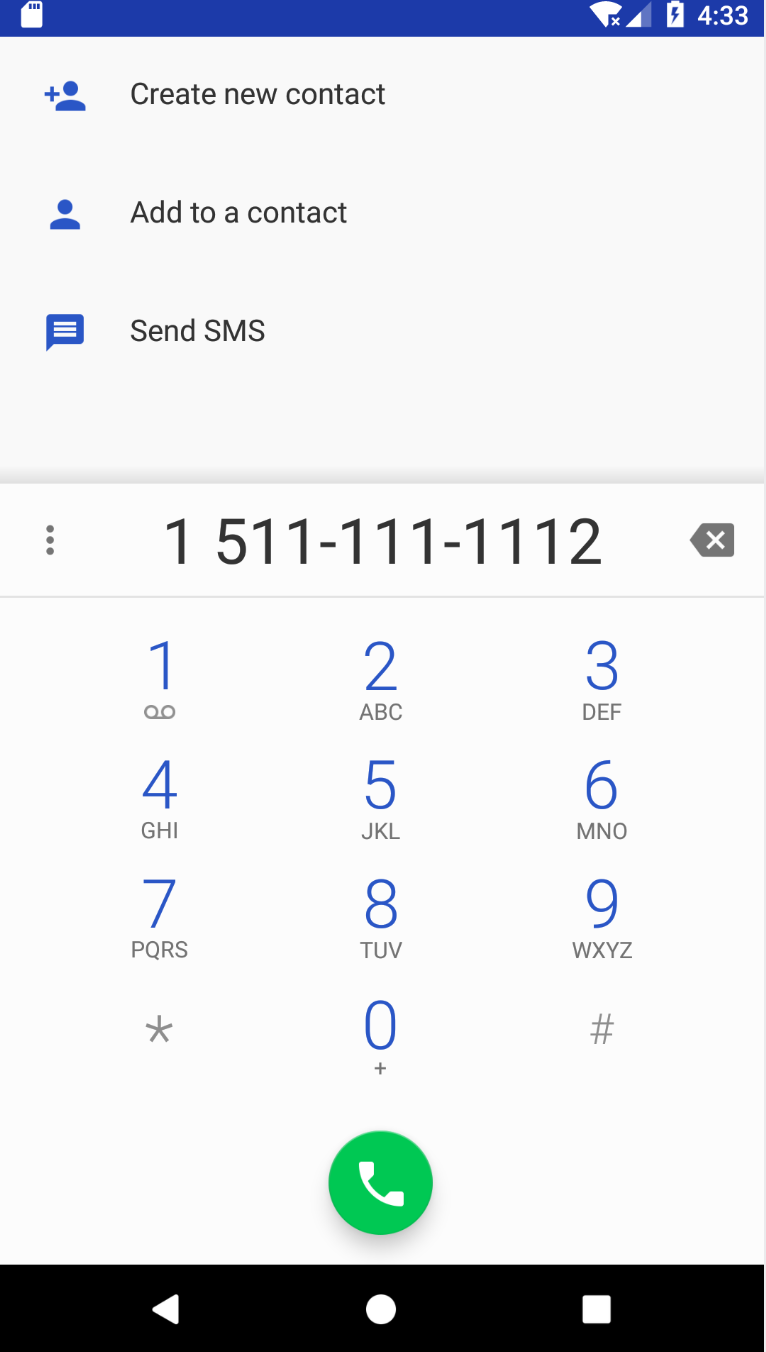
筛选是根据所有的<intent-filter>来筛选。

数据库操作

创建SQLiteOpenHelper和持久层Dao类,调用后通过recyclerView将内容显示,并提供修改新增的接口

1. 运行效果

手机屏幕的截图

描述已自动生成

手机屏幕截图

描述已自动生成 手机屏幕截图

描述已自动生成

* 1. **医疗项目管理模块**

1. 界面设计

以下分别是项目item,项目保存,用户项目item组件树

手机屏幕的截图

描述已自动生成手机屏幕的截图

描述已自动生成

手机屏幕截图

描述已自动生成

1. IPO图

****

**图5 项目模块IPO**

1. 主要算法

下拉列表框：

用adapter给Spinner绑定dao中的list列表获取所有下拉信息,将实体toString方法重写,用于显示表述,提交时取出对应的value值

Spinner提供了从一个数据集合中快速选择一项值的办法。默认情况下Spinner显示的是当前选择的值，点击Spinner会弹出一个包含所有可选值的dropdown菜单，从该菜单中可以为Spinner选择一个新值。

1. 运行效果

手机屏幕的截图

描述已自动生成 手机屏幕截图

描述已自动生成

手机屏幕截图

描述已自动生成

* 1. **订单管理模块**

1. 界面设计

以下分别是订单item,订单用户item,订单保存界面组件树

手机屏幕的截图

描述已自动生成手机屏幕的截图

描述已自动生成

手机屏幕的截图

描述已自动生成

1. IPO图

****

**图6 订单模块IPO**

1. 主要算法

用户界面切换:

用户界面四个子菜单均为fragment,通过4个radiobutton将布局中的占位layout用fragment代替,实现四个不同的用户界面

Fragment是一种可以嵌入在活动中的UI片段，能够让程序更加合理和充分地利用大屏幕的空间，出现的初衷是为了适应大屏幕的平板电脑，可以将其看成一个小型Activity，又称作Activity片段。

使用Fragment可以把屏幕划分成几块，然后进行分组，进行一个模块化管理。Fragment不能够单独使用，需要嵌套在Activity中使用，其生命周期也受到宿主Activity的生命周期的影响

手机屏幕的截图

描述已自动生成手机屏幕截图

描述已自动生成

手机屏幕截图

描述已自动生成

**总结**

对于下拉框获取的使用开始不了解,后经过资料的查询,成功用adapter将实体类绑定在spinner上并实现了功能.

对于md5加密,使用了工具类将功能实现

不足:起初计划实现的用时间选择控件实现日期选择未实现