Android软件设计课程实验

Android软件设计课程实验报告

学号 SA20225506

姓名 谢雪松

日期 2020.10.27

|  |  |
| --- | --- |
| **实验名称** | 实验二：实现Activity跳转以及控件的综合使用 |
| **实验内容** | 要求：1. Button跳转页面；2. ListView使用；3. ImageView使用；4. 布局综合使用；  评分：1. 完成上述4项基本UI设计和功能（7分）；  2. 可根据UI美化效果适当加分或模仿微信或手机QQ的UI和框架设计； |
| **实验完成情况**  （包括完成的实验内容及每个实验的完成程度。注意要贴出每个实验的  核心代码） | 完成的实验内容   1. 完成了Button跳转页面。 完成程度： 2. 使用到了ListView，还使用到了开源的room类进行数据持久化操作 3. 多次使用到了ImageView 4. 通过RadioButton、Fragment配合，使得HomeActivity内通过底部导航按钮切换不同的Fragment。 |
| **实验中的问题**  （包括在实验中遇到的问题，以及解决问题的方法） | 1. 如何通过数据库保存和读取用户的头像。 2. 如何对所有应用列表进行搜索，方便用户查找到含有关键词的应用。 3. 如何让不同用户有不同的收藏列表。 4. 如何让用户的收藏列表为空时，提示如何添加。 |
| **实验结果**  **及成果展示**  （包括实验完成后的源码和打包文件的说明） | 成果展示和完成后的源码   1. 登录界面   LoginActivity.java  package com.rom471.main; import android.content.Intent; import android.graphics.drawable.Drawable; import android.os.Bundle; import android.text.Editable; import android.text.TextWatcher; import android.view.Gravity; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.ImageView; import android.widget.TextView; import android.widget.Toast;  import androidx.appcompat.app.AppCompatActivity;  import com.rom471.lab1.R; import com.rom471.userdb.UsersDBHelper;  public class LoginActivity extends AppCompatActivity implements View.OnClickListener {  EditText name\_et; //用户名输入框  EditText password\_et;//用户密码输入框  UsersDBHelper db; //用户数据库操作类  ImageView avatar\_img; //用户头像图片  TextView register\_tv; //注册按钮  TextView reset\_tv; //重设密码按钮   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*login\_layout*);  //绑定资源  name\_et=findViewById(R.id.*Acount*);  avatar\_img=findViewById(R.id.*pic*);  password\_et=findViewById(R.id.*passwd*);  Button login\_bt=findViewById(R.id.*loginbutton*);  register\_tv=findViewById(R.id.*register\_tv*);  reset\_tv=findViewById(R.id.*forget*);   //设置按钮的点击事件监听器  login\_bt.setOnClickListener(this);  register\_tv.setOnClickListener(this);  reset\_tv.setOnClickListener(this);  //根据名字输入框动态的查找用户的头像  name\_et.addTextChangedListener(new TextWatcher() {  @Override  public void beforeTextChanged(CharSequence s, int start, int count, int after) { }  @Override  public void onTextChanged(CharSequence s, int start, int before, int count) {  String name=s.toString();//获取用户输入的用户名  if(!name.equals("")){  Drawable avatar\_drawable = db.getAvatarByName(name);  if(avatar\_drawable==null){  avatar\_img.setImageResource(R.drawable.*boy*);  }  else {  avatar\_img.setImageDrawable(avatar\_drawable);  }  }  else {  avatar\_img.setImageResource(R.drawable.*boy*);  }  }  @Override  public void afterTextChanged(Editable s) { }  });   //初始化数据库  db=new UsersDBHelper(this);  db.InitialWithTestData();   }   @Override  public void onClick(View v) {  switch (v.getId()){  case R.id.*loginbutton*:  //执行登录操作  login\_action();  break;  case R.id.*register\_tv*:  //跳转到注册界面  Intent intent = new Intent(LoginActivity.this, RegisterActivity.class );  startActivity(intent);  break;  case R.id.*forget*:  //跳转到重设密码界面  Intent intent2 = new Intent(LoginActivity.this, ResetPasswordActivity.class );  startActivity(intent2);  break;  }  }  //登录操作  private void login\_action(){  String name= name\_et.getText().toString();  String password=password\_et.getText().toString();  //先判断数据库里是否存在该用户  if(!db.haveUser(name)){  toast("该用户不存在！");  return;  }  if(db.loginWith(name,password)){  db.close();  // 登录成功，跳转到Home界面  Intent intent = new Intent(LoginActivity.this, HomeActivity.class );  intent.putExtra("name",name);  startActivity(intent);  this.finish();  }  else {  toast("密码错误");  }  }    @Override  protected void onPause() {  super.onPause();  db.close();  }   @Override  protected void onResume() {  super.onResume();  db=new UsersDBHelper(this);  }  //---------toast  public void toast( String text){  Toast toast=Toast.*makeText*(LoginActivity.this, text, Toast.*LENGTH\_SHORT*);  toast.setGravity(Gravity.*CENTER*, 0, 0);  toast.show();  } }  login\_layout.xml  <?xml version="1.0" encoding="utf-8"?> <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:focusableInTouchMode="true"   >   <ImageView  android:id="@+id/pic"  android:layout\_width="370dp"  android:layout\_height="150dp"  android:layout\_centerHorizontal="true"  android:layout\_margin="40dp"  android:scaleType="fitCenter"  android:src="@drawable/boy" />    <EditText  android:id="@+id/Acount"  android:layout\_width="match\_parent"  android:layout\_height="65dp"  android:layout\_below="@+id/pic"  android:layout\_marginLeft="30dp"  android:layout\_marginRight="30dp"  android:layout\_weight="1"   android:background="@drawable/edittext\_selector"  android:hint="@string/account"  android:selectAllOnFocus="false"  android:singleLine="true"  android:textAlignment="center"  android:visibility="visible" />   <EditText  android:id="@+id/passwd"  android:layout\_width="match\_parent"  android:layout\_height="65dp"  android:layout\_below="@+id/Acount"  android:layout\_marginLeft="30dp"  android:layout\_marginTop="10dp"  android:layout\_marginRight="30dp"  android:layout\_marginBottom="35dp"  android:layout\_weight="1"  android:background="@drawable/edittext\_selector"  android:hint="@string/passwd"  android:inputType="textPassword"  android:selectAllOnFocus="false"  android:singleLine="false"  android:textAlignment="center"  android:visibility="visible" />   <Button  android:id="@+id/loginbutton"  android:layout\_width="80dp"  android:layout\_height="50dp"  android:layout\_below="@+id/passwd"  android:layout\_centerHorizontal="true"  android:background="@drawable/button\_selector"  android:text="@string/login" />   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_alignParentBottom="true"  android:layout\_marginBottom="20dp"   android:orientation="horizontal">   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_weight="1"  android:orientation="vertical">   <TextView  android:id="@+id/forget"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_gravity="center\_horizontal"  android:hint="@string/forget"  android:textSize="10pt" />   </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:layout\_weight="1"  android:orientation="vertical">   <TextView  android:id="@+id/register\_tv"  android:layout\_width="wrap\_content"  android:layout\_height="match\_parent"  android:layout\_gravity="center\_horizontal"  android:hint="@string/signin"  android:textSize="10pt" />  </LinearLayout>  </LinearLayout>   </RelativeLayout>   1. 注册界面源码     RegisterActivity.java  package com.rom471.main;  public class RegisterActivity extends AppCompatActivity implements View.OnClickListener {  Button register\_btn; //注册按钮  EditText name\_et; //用户名输入框  EditText password\_et;//密码输入框  EditText email\_et; //用户邮箱输入框  ImageView avatar\_img; //用户头像图片  UsersDBHelper db; //用户数据库操作类  User User =new User(); //用户对象  public final static int *PICTURE*=10; //通过相册获取用户头像的请求码  public final static int *CAMERA* =11; //通过相机获取用户头像的请求码  boolean userNameValid=true; //记录输入的用户名是否合法  boolean registerFinish=false; //注册是否完成  @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_register*);  //绑定组件  register\_btn=findViewById(R.id.*register\_button*);  name\_et=findViewById(R.id.*name\_et*);  password\_et=findViewById(R.id.*passwd\_et*);  email\_et=findViewById(R.id.*email\_et*);  avatar\_img=findViewById(R.id.*avatar\_img*);  //设置点击事件监听器  avatar\_img.setOnClickListener(this);  register\_btn.setOnClickListener(this);  //设置用户名输入框的监视器  name\_et.addTextChangedListener(new TextWatcher() {  @Override  public void beforeTextChanged(CharSequence s, int start, int count, int after) { }  @Override  public void onTextChanged(CharSequence s, int start, int before, int count) {  String name=s.toString();//获取用户输入的用户名  if(db.haveUser(name)){ //该用户名已经存在  name\_et.setTextColor(Color.*RED*);  userNameValid=false;  }  else {  name\_et.setTextColor(Color.*BLACK*);  userNameValid=true;  }  }  @Override  public void afterTextChanged(Editable s) { }  });  //给用户数据库操作类赋值  db=new UsersDBHelper(this);   }   @Override  public void onClick(View v) {  String name=name\_et.getText().toString();  String password=password\_et.getText().toString();  String email=name\_et.getText().toString();  switch (v.getId()){  case R.id.*register\_button*:  if(registerFinish){  Intent intent = new Intent(RegisterActivity.this, HomeActivity.class );  intent.putExtra("name",name);  startActivity(intent);  this.finish();  }  if(!userNameValid){  toast("该用户名已经被使用！");  return;  }  User.setName(name);  User.setPassword(password);  User.setEmail(email);  db.insertUser(User);  MyUtils.*toast*(this,"注册成功");  db.close();  registerFinish=true;  hideViews();  //this.finish();  break;  case R.id.*avatar\_img*:  MyUtils.*verifyStoragePermissions*(this);  getPhotoOrPicture();  break;  }  }  //注册成功后删除一些view  private void hideViews(){  ViewGroup parent =(ViewGroup) name\_et.getParent();  parent.removeView(email\_et);  parent.removeView(password\_et);  name\_et.setFocusable(false);  register\_btn.setText("登录");  }  //弹出对话框，选择从哪里获得头像  private void getPhotoOrPicture(){  final CharSequence[] options = { "从相机", "从相册","取消" };  AlertDialog.Builder builder = new AlertDialog.Builder(RegisterActivity.this);  builder.setTitle("选择一个头像");  builder.setItems(options, new DialogInterface.OnClickListener() {  @Override  public void onClick(DialogInterface dialog, int item) {  if (item==0){ // 打开系统相机获取头像图片  Intent camera = new Intent(MediaStore.*ACTION\_IMAGE\_CAPTURE*);  startActivityForResult(camera, *CAMERA*);  }  else if (item==1){ //打开系统相册获取头像图片  Intent intent = new Intent(Intent.*ACTION\_PICK*,android.provider.MediaStore.Images.Media.*EXTERNAL\_CONTENT\_URI*);  startActivityForResult(intent, *PICTURE*);  }  else if (item==2) { //取消对话框  dialog.dismiss();  }  }  });  builder.show();  }  @Override  protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {  super.onActivityResult(requestCode, resultCode, data);  if(requestCode==*PICTURE* && resultCode==*RESULT\_OK* && data!=null){  String img\_path=null;  Uri uri=data.getData();  img\_path=uri.getPathSegments().get(1);  MyUtils.*log*(img\_path);  Bitmap decodeFile= BitmapFactory.*decodeFile*(img\_path);  Bitmap zoomBitmap = MyUtils.*zoomImg*(decodeFile,200,200);  Drawable avatar\_drawable=new BitmapDrawable(zoomBitmap);  avatar\_img.setImageDrawable(avatar\_drawable);  User.setAvatar(avatar\_drawable);  }  else if(requestCode== *CAMERA* && resultCode==*RESULT\_OK* && data!=null){  Bundle bundle = data.getExtras();  // 获取相机返回的数据，并转换为图片格式  Bitmap bitmap = (Bitmap) bundle.get("data");  Bitmap zoomBitmap = MyUtils.*zoomImg*(bitmap,200,200);  Drawable avatar\_drawable=new BitmapDrawable(zoomBitmap);  avatar\_img.setImageDrawable(avatar\_drawable);  User.setAvatar(avatar\_drawable);  }  }  //---------toast  public void toast( String text){  Toast toast=Toast.*makeText*(RegisterActivity.this, text, Toast.*LENGTH\_SHORT*);  toast.setGravity(Gravity.*CENTER*, 0, 0);  toast.show();  }  }  activity\_register.xml  <?xml version="1.0" encoding="utf-8"?> <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:orientation="vertical"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:focusableInTouchMode="true"  android:gravity="center|top"   >   <ImageView  android:id="@+id/avatar\_img"  android:layout\_height="100dp"  android:layout\_width="100dp"  android:layout\_centerHorizontal="true"  android:layout\_marginVertical="20dp"  android:layout\_marginHorizontal="40dp"  android:src="@drawable/default\_avatar"/>  <EditText  android:id="@+id/name\_et"  android:layout\_width="match\_parent"  android:layout\_height="60dp"  android:layout\_marginVertical="10dp"  android:layout\_marginHorizontal="40dp"  android:background="@drawable/edittext\_selector"  android:hint="@string/name"  android:selectAllOnFocus="false"  android:singleLine="false"  android:visibility="visible"  android:textAlignment="center"/>   <EditText  android:id="@+id/passwd\_et"  android:inputType="textPassword"  android:layout\_width="match\_parent"  android:layout\_height="60dp"  android:layout\_marginVertical="10dp"  android:layout\_marginHorizontal="40dp"  android:background="@drawable/edittext\_selector"  android:hint="@string/passwd"  android:selectAllOnFocus="false"  android:singleLine="false"  android:visibility="visible"  android:textAlignment="center"/>  <EditText  android:id="@+id/email\_et"  android:inputType="textEmailAddress"  android:layout\_width="match\_parent"  android:layout\_height="60dp"  android:layout\_marginVertical="10dp"  android:layout\_marginHorizontal="40dp"  android:background="@drawable/edittext\_selector"  android:hint="@string/email"  android:selectAllOnFocus="false"  android:singleLine="false"  android:visibility="visible"  android:textAlignment="center"/>  <Button  android:id="@+id/register\_button"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:layout\_marginVertical="10dp"  android:layout\_marginHorizontal="40dp"  android:layout\_below="@+id/email\_et"  android:text="@string/register"  android:background="@drawable/button\_selector"/> </LinearLayout>  3、重设密码界面  ResetPasswordActivity.java  package com.rom471.main; import androidx.appcompat.app.AppCompatActivity; import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.ImageView; import com.rom471.lab1.R; import com.rom471.userdb.MyUtils; import com.rom471.userdb.UsersDBHelper;  public class ResetPasswordActivity extends AppCompatActivity implements View.OnClickListener {  Button resetPassword\_btn;  EditText name\_et;  EditText email\_et;  EditText newPassword\_et;  ImageView avatar\_img;  UsersDBHelper db;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_reset\_password*);  //绑定组件  resetPassword\_btn=findViewById(R.id.*register\_button*);  name\_et=findViewById(R.id.*name\_et*);  newPassword\_et=findViewById(R.id.*passwd\_et*);  email\_et=findViewById(R.id.*email\_et*);  avatar\_img=findViewById(R.id.*avatar\_img*);  //设置监听器  avatar\_img.setOnClickListener(this);  resetPassword\_btn.setOnClickListener(this);  //初始化数据库操作类  db=new UsersDBHelper(this);   }   @Override  public void onClick(View v) {  if (v.getId() == R.id.*register\_button*) {  String name = name\_et.getText().toString();  String password = newPassword\_et.getText().toString();  String email = email\_et.getText().toString();  if(!db.haveUser(name)){  MyUtils.*toast*(this, "该用户不存在！");  return;  }  //修改密码成功会产生变化  int i = db.updatePasswordByNameAndEmail(name, email, password);  if(i<=0){ //没有变化，代表邮箱不匹配  MyUtils.*toast*(this, "邮箱信息不匹配");  return;  }else {  // 密码重置成功  MyUtils.*toast*(this, "密码重设成功");   }    }  }  }  布局文件activity\_reset\_password.xml  <?xml version="1.0" encoding="utf-8"?> <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:focusableInTouchMode="true"  >  <ImageView  android:id="@+id/avatar\_img"  android:layout\_height="120dp"  android:layout\_width="120dp"  android:scaleType="fitXY"  android:layout\_centerHorizontal="true"  android:layout\_margin="40dp"  android:src="@drawable/changepassword"/>  <EditText  android:id="@+id/name\_et"  android:layout\_width="match\_parent"  android:layout\_height="65dp"  android:layout\_marginLeft="30dp"  android:layout\_marginRight="30dp"  android:layout\_below="@+id/avatar\_img"  android:layout\_weight="1"   android:background="@drawable/edittext\_selector"  android:hint="@string/name\_reset"  android:selectAllOnFocus="false"  android:singleLine="false"  android:visibility="visible"  android:textAlignment="center"/>  <EditText  android:id="@+id/email\_et"  android:inputType="textEmailAddress"  android:layout\_width="match\_parent"  android:layout\_height="65dp"  android:layout\_marginTop="10dp"  android:layout\_marginLeft="30dp"  android:layout\_marginRight="30dp"  android:layout\_marginBottom="10dp"  android:layout\_below="@+id/name\_et"  android:layout\_weight="1"  android:background="@drawable/edittext\_selector"  android:hint="@string/email\_reset"  android:selectAllOnFocus="false"  android:singleLine="false"  android:visibility="visible"  android:textAlignment="center"/>  <EditText  android:id="@+id/passwd\_et"  android:inputType="textPassword"  android:layout\_width="match\_parent"  android:layout\_height="65dp"  android:layout\_marginTop="10dp"  android:layout\_marginLeft="30dp"  android:layout\_marginRight="30dp"  android:layout\_marginBottom="10dp"  android:layout\_below="@+id/email\_et"  android:layout\_weight="1"  android:background="@drawable/edittext\_selector"  android:hint="@string/passwd\_new"  android:selectAllOnFocus="false"  android:singleLine="false"  android:visibility="visible"  android:textAlignment="center"/>  <Button  android:id="@+id/register\_button"  android:layout\_width="80dp"  android:layout\_height="50dp"  android:layout\_below="@+id/passwd\_et"  android:layout\_centerHorizontal="true"  android:text="@string/passwd\_reset"  android:background="@drawable/button\_selector"/> </RelativeLayout>  4、主界面 主界面由我的收藏和所有应用两个板块组成，通过下方的按钮可以实现切换。     1. 我的收藏Fragment 我的收藏由 2. 所有应用Fragment |