

桂林航天工业学院

安卓大作业报告

音乐相册 APP 设计报告

课程名称	移动应用软件开发（B）
指导老师	张新伦
开课学期	2019-2020 第一学期
班 级	软件工程 4 班
姓 名	钟 祯
学 号	2017070030429

1. APP设计的目的

现今社会生活紧张，而欣赏音乐是其中最好的舒缓压力的方式之一，随着手机市场的迅速发展，各种音频资源也在网上广为流传，通过手机音乐播放器放松心情已经渐渐成为人们生活中必不可少的一部分了。不少人在购买手机的时候都会关心手机的音乐播放的能力，这也足以看出目前大家对音乐播放功能的重视。

在欣赏音乐的同时，还能欣赏自己所拍摄的美照，自定义摆放，播放方式，用户通过该系统可完成照片的选取，对于处理后的照片可添加背景音乐，制作成具有动态效果的电子相册，该软件可操作性强、方便实用。

所以综上所述，实现的功能有：

- 用户的注册功能
- 用户的登录功能
- 图片自动轮播功能
- 图片手动切换功能
- 图片列表展示功能
- 背景音乐播放功能

2. APP设计的主要步骤

2.1 设计方法

整个程序基于 Android Studio 开发，对所要实现的功能大概分为以下模块：

- 登录模块
- 注册模块
- 图片自动轮播模块
- 图片手动切换模块
- 图片列表展示模块
- 背景音乐播放模块

登录模块：用户需要输入正确的账户密码才能登录本软件正常使用。若用户是第一次使用本软件，可以使用注册模块来拥有自己的账号和密码。该功能主要是运用了 SQLite 的数据插入和查询功能。

当用户登录后，首先进入的就是我们软件的主界面，软件的主界面主要运用了 Fragment 嵌套使用，底部是导航栏。

用户在浏览后，可以根据自己的喜好玩转 APP。

2.2 登录界面



图 2.1 登录主页



图 2.2 登录账户不存在提示



图 2.3 密码错误提示

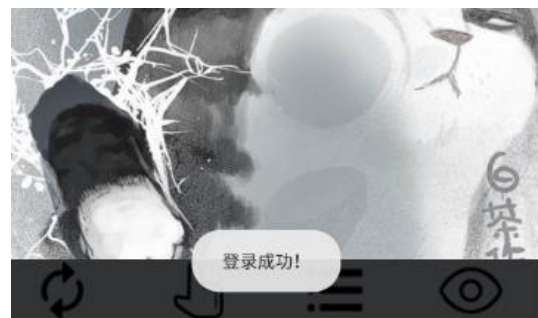


图 2.4 登录成功提示

2.2.1 登录布局代码

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/im_car"
    android:gravity="center"
    android:orientation="vertical">

    <EditText
        android:id="@+id/User_id"
        android:layout_width="230dp"
        android:layout_height="40dp"
        android:layout_marginTop="20dp"
        android:background="@drawable/edit_bg"
        android:gravity="left"
        android:hint="手机号/邮箱"
        android:padding="10dp"
        android:textColor="#131212"
        android:textColorHint="#6A6A6A" />

    <EditText
        android:id="@+id/User_pwd"
        android:layout_width="230dp"
        android:layout_height="40dp"
        android:layout_marginTop="10dp"
        android:background="@drawable/edit_bg"
        android:gravity="left"
        android:hint="密码"
        android:inputType="textPassword"
        android:padding="10dp"
        android:textColor="#131212"
        android:textColorHint="#6A6A6A" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```

        android:layout_marginTop="30dp"
        android:gravity="center"
        android:orientation="horizontal">

        <Button
            android:layout_width="100dp"
            android:layout_height="40dp"
            android:background="@drawable/button_bg"
            android:onClick="login"
            android:text="登录" />

        <Button
            android:layout_width="100dp"
            android:layout_height="40dp"
            android:layout_marginLeft="20dp"
            android:background="@drawable/button_bg"
            android:onClick="register"
            android:text="注册" />
    </LinearLayout>

```

2.2.2 后端实现代码

```

public class LoginActivity extends AppCompatActivity {

    //初始化数据库类
    private UserInfoDB userInfoDB;
    private SQLiteDatabase sqLiteDatabase;

    //初始化 EditText 控件，用于获取前端输入的值
    private EditText UserId, password;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        //找到控件的 id
        UserId = findViewById(R.id.User_id);
        password = findViewById(R.id.User_pwd);
    }
}

```

```

        //打开数据库操作
        userInfoDB = new UserInfoDB(this, "UserInfo.db", null, 1);
        SQLiteDatabase = userInfoDB.getReadableDatabase();
    }

    //跳转到注册界面
    public void register(View view) {
        startActivity(new Intent(LoginActivity.this,
RegisterActivity.class));
    }

    //登录界面
    public void login(View view) {
        query();
    }

    //查询数据库是否存在该用户
    public void query() {
        //查询用户 id
        @SuppressWarnings("Recycle") Cursor cursor =
        SQLiteDatabase.rawQuery("select * from UserInfo where User_id=?", new
String[] {UserId.getText().toString()});
        //查询用户密码
        @SuppressWarnings("Recycle") Cursor pwd =
        SQLiteDatabase.rawQuery("select * from UserInfo where password=?", new
String[] {password.getText().toString()});

        //判断是否存在该用户
        if (cursor.getCount() > 0) {
            //判断密码是否正确
            if (pwd.moveToFirst()) {
                start();
            } else {
                Toast.makeText(this, "账号密码不匹配",
Toast.LENGTH_LONG).show();
                password.setText("");
            }
        }
    }

```

```

        } else {
            AlertDialog.Builder builder = new
AlertDialog.Builder(this);
            builder.setTitle("提示");
            builder.setMessage("用户不存在");
            builder.setPositiveButton("前往注册", new
DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which)
{
                    startActivity(new Intent(LoginActivity.this,
RegisterActivity.class));
                }
            });
            builder.setNegativeButton("重新输入", new
DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which)
{
                    }
            });
            builder.setCancelable(true);
            builder.create();
            builder.show();
        }
    }
}

```

//当前界面结束时关闭数据库连接

```

@Override
protected void onDestroy() {
    if (sqliteDatabase != null) {
        sqliteDatabase.close();
        userInfoDB.close();
    }
    super.onDestroy();
}

```

//存入数据，传递参数，跳转界面


```
public void start() {  
    String id = UserId.getText().toString().trim();  
    Intent intent = new Intent();  
    intent.setClass(LoginActivity.this, MainActivity.class);  
    //存储用户 id 信息，传递到主界面  
    intent.putExtra("User_id", id);  
    startActivity(intent);  
    LoginActivity.this.finish();  
    Toast.makeText(this, "登录成功!", Toast.LENGTH_LONG).show();  
}  
}
```

2.3 注册界面



用户注册

电话/邮箱

用户昵称

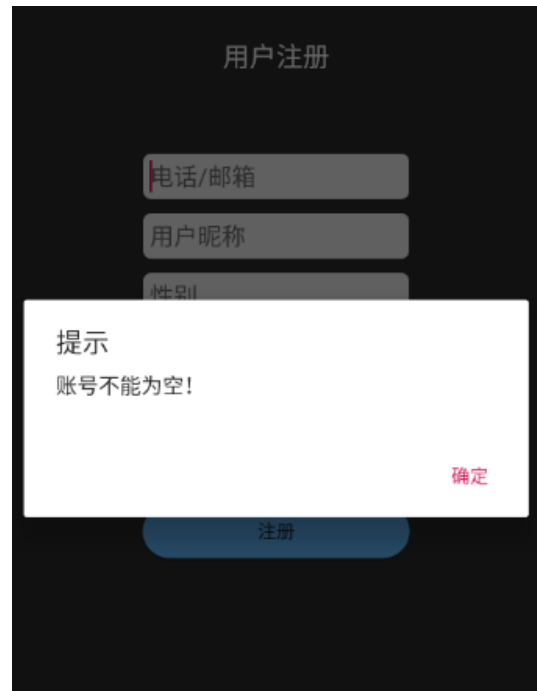
性别

密码

确认密码

注册

图 2.5 注册首页



用户注册

提示

账号不能为空!

确定

注册

图 2.6 错误提示



用户注册

33333333

33333

3333

提示

两次密码不一致!

确定

注册

图 2.7 错误提示

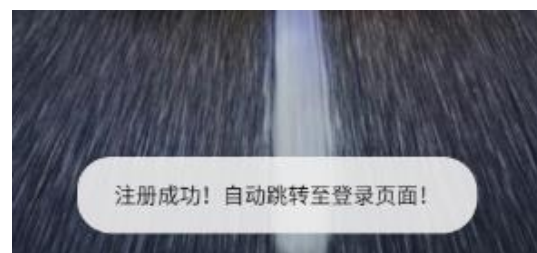


图 2.8 注册成功提示

2.3.1 注册布局代码

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#2B2B2B"
    android:orientation="vertical"
    tools:context=".RegisterActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:layout_marginTop="50dp"
        android:gravity="center"
        android:padding="10dp"
        android:text="用户注册"
        android:textColor="#fff"
        android:textSize="20sp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="400dp"
        android:gravity="center"
        android:orientation="vertical">

        <EditText
            android:id="@+id/User_id"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:background="@drawable/edit_bg"
            android:hint="电话/邮箱"
            android:inputType="phone|textEmailAddress"
            android:padding="5dp"
            android:textColor="#131212"
            android:textColorHint="#6A6A6A" />
```

```
<EditText
    android:id="@+id/User_name"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:background="@drawable/edit_bg"
    android:hint="用户昵称"
    android:padding="5dp"
    android:textColor="#131212"
    android:textColorHint="#6A6A6A" />
```

```
<EditText
    android:id="@+id/User_sex"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:background="@drawable/edit_bg"
    android:hint="性别"
    android:padding="5dp"
    android:textColor="#131212"
    android:textColorHint="#6A6A6A" />
```

```
<EditText
    android:id="@+id/password"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:background="@drawable/edit_bg"
    android:hint="密码"
    android:inputType="textPassword"
    android:padding="5dp"
    android:textColor="#131212"
    android:textColorHint="#6A6A6A" />
```

```
<EditText
    android:id="@+id/check"
    android:layout_width="200dp"
```

```

        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:background="@drawable/edit_bg"
        android:hint="确认密码"
        android:inputType="textPassword"
        android:padding="5dp"
        android:textColor="#131212"
        android:textColorHint="#6A6A6A" />

        <Button
            android:layout_width="200dp"
            android:layout_height="40dp"
            android:layout_marginTop="50dp"
            android:background="@drawable/button_bg"
            android:onClick="checkInfo"
            android:text="注册" />
    </LinearLayout>

</LinearLayout>

```

2.3.2 后端实现代码

```

public class RegisterActivity extends AppCompatActivity {
    //初始化控件
    private EditText User_id, password, check, UserName, sex;
    //初始化数据库类
    private UserInfoDB userInfoDB;
    private SQLiteDatabase sqLiteDatabase;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);

        //找到控件的 id
        User_id = findViewById(R.id.User_id);
        password = findViewById(R.id.password);
        check = findViewById(R.id.check);
        UserName = findViewById(R.id.User_name);
    }
}

```

```

sex = findViewById(R.id.User_sex);

//创建数据库名字
userInfoDB = new UserInfoDB(this, "UserInfo.db", null, 1);
sqliteDatabase = userInfoDB.getReadableDatabase();
}

//注册按钮
public void checkInfo(View view) {
    //调用注册限制
    String result = Info();
    if (result != null) {
        //设置弹窗对象
        AlertDialog.Builder builder = new
AlertDialog.Builder(this);
        //设置弹窗标题
        builder.setTitle("提示");
        builder.setMessage(result);
        builder.setPositiveButton("确定", null);
        builder.setCancelable(true);
        builder.create();
        builder.show();
    } else write();
}

//注册限制
private String Info() {
    //初始化三个 String 类型，用于接收参数
    String id = User_id.getText().toString().trim();
    String pwd = password.getText().toString().trim();
    String checkPwd = check.getText().toString().trim();
    //查询数据库是否存在已存在注册的 id
    @SuppressWarnings("Recycle") Cursor cursor =
sqliteDatabase.rawQuery("select * from UserInfo where User_id=?", new
String[] {
        id
    });
    //判断查询结果，如果 getCount 大于 0 说明数据库已存在该对象

```

```

        if (cursor.getCount() > 0) {
            return "用户已存在! ";
        } else {
            if ("".equals(id)) {
                return "账号不能为空! ";
            }
            if (pwd.length() < 6 || pwd.length() > 16) {
                return "密码数字应在 6 到 16 位之间! ";
            }
            if (sex.getText().equals("boy") ||
sex.getText().equals("girl")) {
                return "性别有误! ";
            }
            if (!checkPwd.equals(pwd)) {
                return "两次密码不一致! ";
            }
        }
        return null;
    }

    //把注册信息写入注册表
    public void write() {
        SQLiteDatabase.execSQL("insert into UserInfo
values(null,?, ?, ?, ?, ?)", new String[] {
            User_id.getText().toString(),
            UserName.getText().toString(),
            password.getText().toString(),
            sex.getText().toString()
        });
        Toast.makeText(this, "注册成功! 自动跳转至登录页面!",
Toast.LENGTH_LONG).show();
        startActivity(new Intent(RegisterActivity.this,
LoginActivity.class));
        RegisterActivity.this.finish();
    }

    //当前界面结束时关闭数据库
    @Override

```

```
protected void onDestroy() {  
    if (sqliteDatabase != null) {  
        userInfoDB.close();  
        sqliteDatabase.close();  
    }  
    super.onDestroy();  
}  
}
```


2.4 图片轮播主页

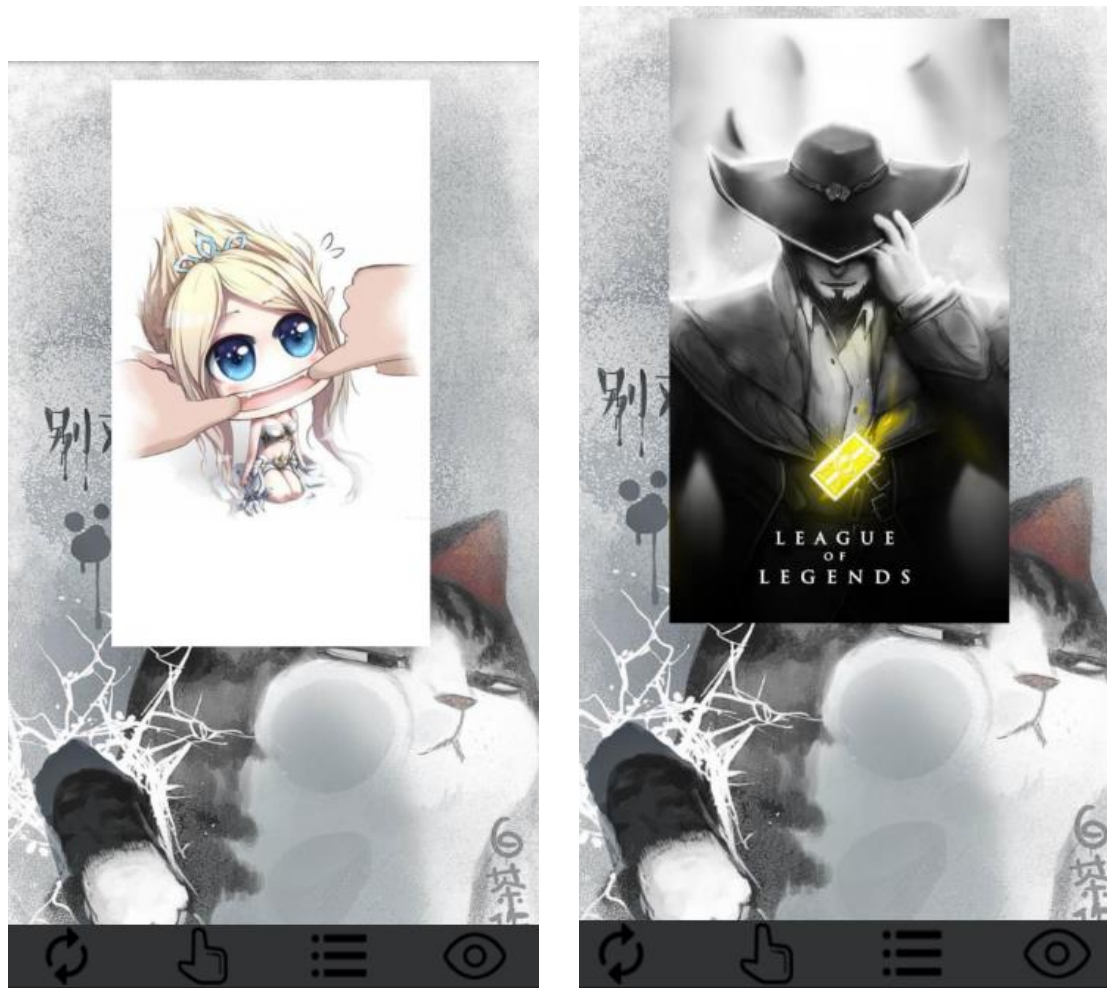


图 2.9 图片轮播主页

2.4.1 轮播页布局代码

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingTop="15dp">

    <ImageSwitcher
        android:id="@+id/banner"
        android:layout_width="match_parent"
        android:layout_height="450dp"
        android:inAnimation="@anim/slide_in_right"
```

```
        android:outAnimation="@anim/slide_out_left" />
```

```
</LinearLayout>
```

2.4.2 后端实现代码

```
public class AutoFragment extends Fragment {
    private View view;

    private int[] images = {R.drawable.im_auto1, R.drawable.im_auto2,
        R.drawable.im_auto3,
        R.drawable.im_auto4, R.drawable.im_auto5};

    private int num = 1;
    private ImageSwitcher banner;

    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater,
        @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
        view = inflater.inflate(R.layout.activity_auto, container,
            false);
        StartBanner();

        banner = view.findViewById(R.id.banner);
        banner.setFactory(new ViewSwitcher.ViewFactory() {
            @Override
            public View makeView() {
                return new ImageView(view.getContext());
            }
        });

        banner.setImageResource(images[0]);
        return view;
    }

    //初始化图片数据
    private void StartBanner() {
        new CountDownTimer(3 * 1000, 1000) {
```

```

@Override
public void onTick(long l) {

}

@Override
public void onFinish() {
    banner.setImageResource(images[num]);
    num++;
    if (num > 4) {
        num = 0;
    }
    StartBanner();
}
}.start();
}
}

```

2.5 手动切换图片页

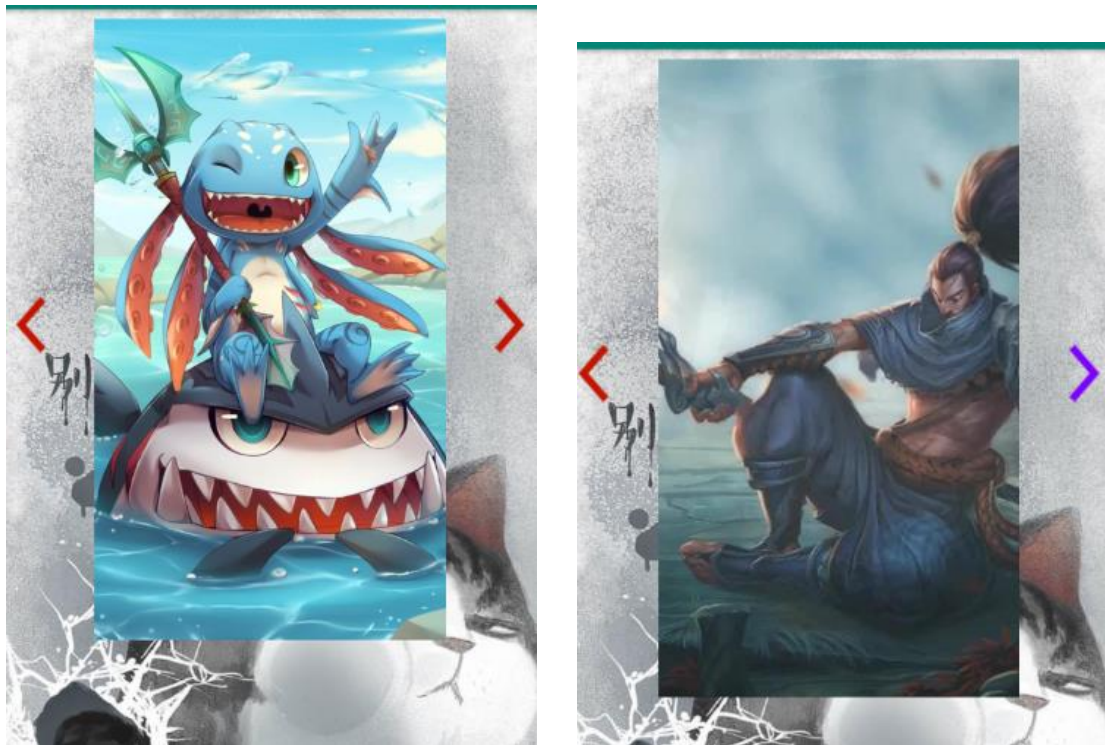


图 2.10 手动切换图片页

2.5.1 手动切换页布局代码

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center_vertical"
    android:orientation="horizontal">

    <ImageButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#00000000"
        android:id="@+id/left"
        android:src="@drawable/left" />

    <ImageSwitcher
        android:id="@+id/mSwitcher">
```

```

        android:layout_width="0dp"
        android:layout_marginTop="5dp"
        android:layout_height="500dp"
        android:layout_weight="1"
        android:inAnimation="@android:anim/slide_in_left"
        android:outAnimation="@android:anim/slide_out_right" />

<ImageButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="#00000000"
    android:id="@+id/right"
    android:src="@drawable/right" />
</LinearLayout>
<!-- 中间的分割线 -->
<View
    android:layout_height="2dp"
    android:layout_marginBottom="10dp"
    android:layout_marginTop="5dp"
    android:background="#fff"
    android:layout_width="match_parent" />

<!-- 用来显示评论的 list view -->
<ListView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/commentList"/>

</LinearLayout>

```

2.5.2 后端实现代码

```

public class HandFragment extends Fragment {

    private View view;
    private ImageSwitcher mSwitcher;
    private int currentIndex = 0;
    private int[] images = {R.drawable.im_hand1, R.drawable.im_hand2,
R.drawable.im_hand3,

```

```

        R.drawable.im_hand4, R.drawable.im_hand5});

private ImageButton left, right;

@Nullable
@Override
public View onCreateView(@NonNull LayoutInflater inflater,
@Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {

    view = inflater.inflate(R.layout.activity_hand, container,
false);

    left = view.findViewById(R.id.left);
    right = view.findViewById(R.id.right);
    mSwitcher = view.findViewById(R.id.mSwitcher);
    mSwitcher.setFactory(new ViewSwitcher.ViewFactory() {
        @Override
        public View makeView() {
            return new ImageView(view.getContext());
        }
    });
    mSwitcher.setImageResource(images[currentIndex]);
    return view;
}

@Override
public void onActivityCreated(@Nullable Bundle savedInstanceState)
{
    left.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            mSwitcher.setInAnimation(view.getContext(),
R.anim.slide_in_left);
            mSwitcher.setOutAnimation(view.getContext(),
R.anim.slide_out_right);
            currentIndex = (currentIndex - 1 + images.length) %
images.length;

```

```

        mSwitcher.setImageResource(images[currentIndex]);
    }
});

right.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        mSwitcher.setInAnimation(view.getContext(),
R.anim.slide_in_right);
        mSwitcher.setOutAnimation(view.getContext(),
R.anim.slide_out_left);
        currentIndex = (currentIndex + 1) % images.length;
        mSwitcher.setImageResource(images[currentIndex]);
    }
});

super.onCreate(savedInstanceState);
}
}

```

2.6 图片列表页



图 2.11 图片列表页

2.6.1 列表页布局代码

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical">

    <ListView
        android:id="@+id/picture"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />

</LinearLayout>
```


2.6.2 后端实现代码

```
public class ListFragment extends Fragment {
    private List<Map<String, Object>> data = new ArrayList<>();

    private int[] images = {R.drawable.im_list1, R.drawable.im_list2,
R.drawable.im_list3,
        R.drawable.im_list4, R.drawable.im_list5};

    private String[] names = {"凯隐", "金克丝", "锐雯", "联盟", "亚索"};

    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater,
@Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.activity_list, container,
false);
        ListView picture = view.findViewById(R.id.picture);

        initData();
        SimpleAdapter simpleAdapter = new
SimpleAdapter(view.getContext(), data, R.layout.picturelist,
            new String[]{"image", "text"},
            new int[]{R.id.picture, R.id.text});
        picture.setAdapter(simpleAdapter);
        picture.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> adapterView, View
view, int i, long l) {
                Intent intent = new Intent();
                intent.setClass(view.getContext(),
InfoActivity.class);
                intent.putExtra("index", String.valueOf(i));
                startActivity(intent);
            }
        });
        return view;
    }
}
```

```

    }

    //初始化数据方法
    private void initData() {
        int i;
        int length = images.length;
        for (i = 0; i < length; i++) {
            Map<String, Object> item = new HashMap<>();
            item.put("image", images[i]);
            item.put("text", names[i]);
            data.add(item);
        }
    }
}

```

2.7 读取相册页

2.7.1 读取相册页布局代码

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <ImageView
        android:id="@+id/readImage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="24sp" />

</LinearLayout>
```

2.7.1 后端实现代码

```
public class ReadActivity extends AppCompatActivity {
    private String path;
    private ImageView readImage;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_read);
        readImage = findViewById(R.id.readImage);
        path =
Environment.getExternalStorageDirectory().getAbsolutePath() +
"/123.jpg";
        boolean fileExist = fileIsExists(path);

        if (fileExist) {
            readImageClass(readImage);
        }
    }
}
```

```

public void readImageClass(View view) {
    Bitmap bitmap = BitmapFactory.decodeFile(path);
    readImage.setImageBitmap(bitmap);
}

public boolean fileIsExists(String strFile) {
    try {
        File f = new File(strFile);
        if (!f.exists()) {
            return false;
        }
    } catch (Exception e) {
        return false;
    }
    return true;
}
}

```

2.8 主页布局代码

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/im_fitness"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <LinearLayout

        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <FrameLayout

            android:id="@+id/mainFragment"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_weight="1" />

        <LinearLayout

            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:layout_alignParentBottom="true"
            android:background="#323335"
            android:orientation="horizontal">

            <ImageView

                android:layout_width="120dp"
                android:layout_height="50dp"
                android:layout_weight="1"
                android:onClick="Auto"
                android:src="@drawable/ic_auto"
                android:text="自动" />
```

```
<ImageView
    android:layout_width="120dp"
    android:layout_height="50dp"
    android:layout_weight="1"
    android:onClick="Hand"
    android:src="@drawable/ic_hand"
    android:text="手动" />
```

```
<ImageView
    android:layout_width="120dp"
    android:layout_height="50dp"
    android:layout_weight="1"
    android:onClick="listImage"
    android:src="@drawable/ic_list"
    android:text="列表" />
```

```
<ImageView
    android:layout_width="120dp"
    android:layout_height="50dp"
    android:layout_weight="1"
    android:onClick="Read"
    android:src="@drawable/ic_read"
    android:text="读取" />
```

```
</LinearLayout>
```

```
</LinearLayout>
```

```
</RelativeLayout>
```

2.9 主页后端代码

```
public class MainActivity extends AppCompatActivity {
    private MediaPlayer playerMusic;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        getSupportFragmentManager().beginTransaction().replace(R.id.mainFragm
ent, new AutoFragment()).commit();
        startMusic();
    }

    public void Auto(View view) {

        getSupportFragmentManager().beginTransaction().replace(R.id.mainFragm
ent, new AutoFragment()).commit();
    }

    public void Hand(View view) {

        getSupportFragmentManager().beginTransaction().replace(R.id.mainFragm
ent, new HandFragment()).commit();
    }

    public void Read(View view) {
        Intent intent = new Intent();
        intent.setClass(this, ReadActivity.class);
        startActivity(intent);
    }

    public void listImage(View view) {

        getSupportFragmentManager().beginTransaction().replace(R.id.mainFragm
ent, new ListFragment()).commit();
    }
}
```

```
public void startMusic() {  
    playerMusic = MediaPlayer.create(this, R.raw.wind);  
    playerMusic.setLooping(true);  
    playerMusic.start();  
}  
  
@Override  
protected void onDestroy() {  
    super.onDestroy();  
    playerMusic.stop();  
}  
}
```

2.10 图片列表详情页



图 2.12 图片列表详情页

2.10.1 图片列表详情页布局代码

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".InfoActivity">

    <TextView
        android:id="@+id/info"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        android:textColor="#3A819B"
        android:textSize="40sp" />

</LinearLayout>
```

2.10.2 后端实现代码

```
public class InfoActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_info);
        TextView info = findViewById(R.id.info);
        Intent intent = getIntent();
        if (intent != null) {
            String index = intent.getStringExtra("index");
            assert index != null;
            if (index.equals("0")) {
                info.setText("我是凯隐");
            }
            if (index.equals("1")) {
                info.setText("我是金克丝");
            }
            if (index.equals("2")) {
                info.setText("我是锐雯");
            }
        }
    }
}
```

```
    }  
    if (index.equals("3")) {  
        info.setText("我是联盟");  
    }  
    if (index.equals("4")) {  
        info.setText("我是亚索");  
    }  
    }  
    }  
}
```

3. APP设计的不足

本次的设计虽然完成了基本的功能模块（登录、注册、图片自动轮播、图片手动轮播、图片列表、图片列表项点击详情）功能，但是一个完整的 app 这些功能还是不够的。目前主要缺陷如下：

- 读取相册功能未实现
- APP 只能在用户个人手机上使用，不能联网
- 缺少服务终端

通过这次安卓开发的设计，我学到了很多。因此也发现自己的知识能力是远远不够的，还有很多深奥复杂的功能还不能自己实现，还需要查找相关资料来辅助完成 app 的开发。在此之后，我需要更加努力学习。