

项目 5



时事阅读

本项目通过鸿蒙系统开发工具 DevEco Studio, 基于 Java 开发一款阅读类 App, 实现获取随机笑话、今日新闻、历史上的今天。

5.1 总体设计

本部分包括系统架构和系统流程。

5.1.1 系统架构

系统架构如图 5-1 所示。



图 5-1 系统架构

5.1.2 系统流程

系统流程如图 5-2 所示。

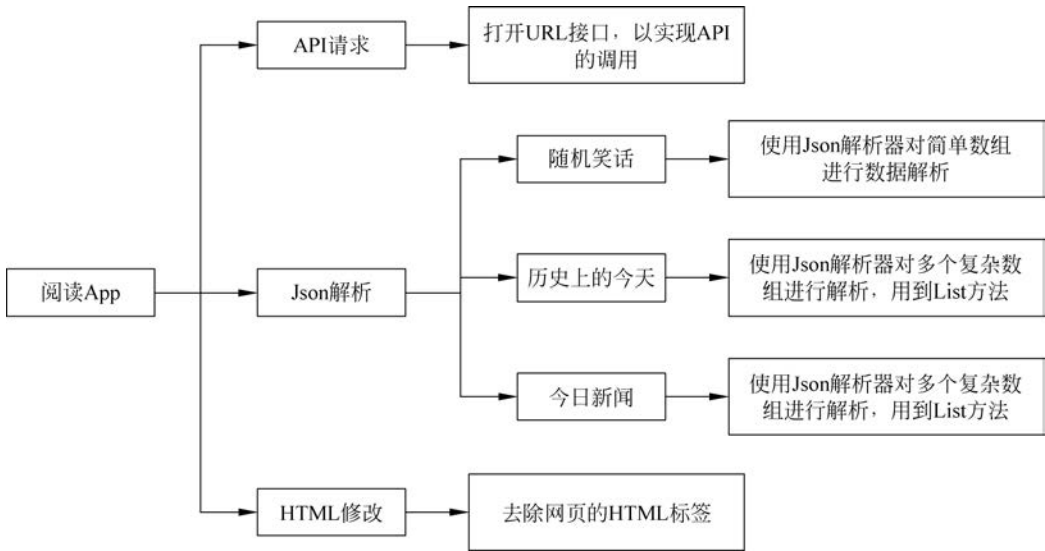


图 5-2 系统流程

5.2 开发工具

本项目使用 DevEco Studio 开发工具, 安装过程如下。

(1) 注册开发者账号, 完成注册并登录, 在官网下载 DevEco Studio 并安装。

(2) 模板类型选择 Empty Feature Ability, 设备类型选择 Phone, 语言类型选择 Java, 单击 Next 后填写相关信息。

(3) 创建后的应用目录结构如图 5-3 所示。

(4) 在 src/main/java 目录下进行阅读 App 的应用开发。

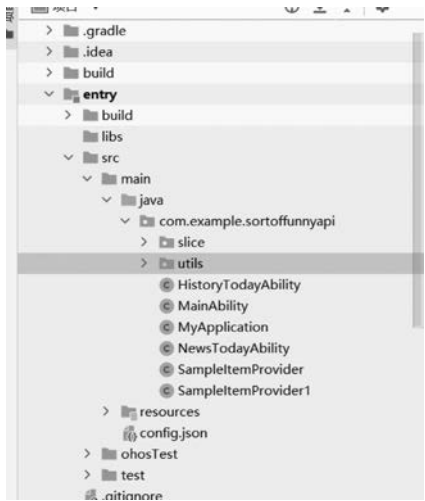


图 5-3 应用目录结构

5.3 开发实现

本部分包括界面设计和程序开发, 下面分别给出各模块的功能介绍及相关代码。

5.3.1 界面设计

本部分包括图片导入、界面布局和完整代码。

1. 图片导入

首先,将选好的界面图片导入 project 文件;然后,将 picture(.jpg 格式)保存在 main/resources/base/media 文件夹下,如图 5-4 所示。



图 5-4 图片导入

2. 界面布局

本部分包括主界面、今日新闻和历史上的今天。

(1) 主界面。

```
<?xml version = "1.0" encoding = "utf - 8"?>
<DirectionalLayout xmlns:ohos = "http://schemas.huawei.com/res/ohos"
    ohos:height = "match_parent"
    ohos:width = "match_parent"
    ohos:alignment = "top"
    ohos:orientation = "vertical"
    ohos:background_element = " $ graphic:background_ability_main"
>
//App 名称显示在主界面中
<Text
    ohos:height = "80vp"
    ohos:width = "match_parent"
    ohos:text = "悦读"
    ohos:text_size = "60vp"
    ohos:text_font = "sans - serif - condensed - medium"
    ohos:bottom_margin = "20vp"
    ohos:top_margin = "0fp"
/>
```

(2) 今日新闻。

```
<?xml version = "1.0" encoding = "utf - 8"?>
<DirectionalLayout
    xmlns:ohos = "http://schemas.huawei.com/res/ohos"
    ohos:height = "match_parent"
    ohos:width = "match_parent"
    ohos:alignment = "center"
    ohos:orientation = "vertical">
//ListContainer 容器的设置
<ListContainer
    ohos:id = " $ + id:list1"
    ohos:height = "match_parent"
    ohos:width = "match_parent"
    ohos:orientation = "vertical"
/>
</DirectionalLayout>
```

(3) 历史上的今天。

```
<?xml version = "1.0" encoding = "utf-8"?>
<DirectionalLayout
    xmlns:ohos = "http://schemas.huawei.com/res/ohos"
    ohos:height = "match_parent"
    ohos:width = "match_parent"
    ohos:alignment = "center"
    ohos:orientation = "vertical">
//ListContainer 容器的设置
    <ListContainer
        ohos:id = "$ + id:list"
        ohos:height = "match_parent"
        ohos:width = "match_parent"
        ohos:orientation = "vertical"
    />
</DirectionalLayout>
```



文件 6

3. 完整代码

界面设计完整代码请扫描二维码文件 6 获取。

5.3.2 程序开发

本部分包括打开 URL 接口、Json 数据解析、HTML 解析、主界面、今日新闻界面、历史上的今天界面和 ListContainer 适配器。

1. 打开 URL 接口

打开 URL 接口以便后续进行 URL 调用。

```
public class APIRequest {
    static final HiLogLabel LABEL = new HiLogLabel(HiLog.LOG_APP, 0x00201, "MY_TAG");
    public static String URLRequest(String myurl,String Reqmethod){
        NetManager netManager = NetManager.getInstance(null);
        if (!netManager.hasDefaultNet()) {
            return "";
        }
    }
//可以获取网络状态的变化
    NetHandle netHandle = netManager.getDefaultNet();
    HttpURLConnection connection = null;
    String resText = " ";
    StringBuilder response = new StringBuilder();
    try {
        URL url = new URL(myurl);
        URLConnection urlConnection = netHandle.openConnection(url,
            java.net.Proxy.NO_PROXY);
        if (urlConnection instanceof HttpURLConnection) {
            connection = (HttpURLConnection) urlConnection;
        }
        connection.setRequestMethod(Reqmethod);
        connection.connect();
```

```

//可进行 URL 的其他操作
InputStream in = connection.getInputStream();
BufferedReader reader = new BufferedReader(new InputStreamReader(in));
String line; while((line = reader.readLine()) != null){
    response.append(line);
}
HiLog.info(LABEL, response.toString());
} catch(IOException e) {
    e.printStackTrace();
} finally {
    if (connection != null){
        connection.disconnect();
    }
}
}
resText = response.toString();
return resText;
}
}

```

2. Json 数据解析

使用 Gson 解析 Json 数据,包括简单数组数据和复杂的多个数组,相关代码请扫描二维码文件 7 获取。



文件 7

3. HTML 解析

HTML 解析去除界面的 HTML 标签。

```

public class HTMLChange {
    public static String delHTMLTag(String htmlStr) {
        String regEx_script = "<script[^>]*?>[\\s\\S]*?<\\/script>";
        //定义 script 的正则表达式
        String regEx_style = "<style[^>]*?>[\\s\\S]*?<\\/style>";
        //定义 style 的正则表达式
        String regEx_html = "<[^>]+>"; //定义 HTML 标签的正则表达式
        Pattern p_script = Pattern.compile(regEx_script, Pattern.CASE_INSENSITIVE);
        Matcher m_script = p_script.matcher(htmlStr);
        htmlStr = m_script.replaceAll(""); //过滤 script 标签
        Pattern p_style = Pattern.compile(regEx_style, Pattern.CASE_INSENSITIVE);
        Matcher m_style = p_style.matcher(htmlStr);
        htmlStr = m_style.replaceAll(""); //过滤 style 标签
        Pattern p_html = Pattern.compile(regEx_html, Pattern.CASE_INSENSITIVE);
        Matcher m_html = p_html.matcher(htmlStr);
        htmlStr = m_html.replaceAll(""); //过滤 HTML 标签
        return htmlStr.trim(); //返回文本字符串
    }
    //删除标签
    public static String stripHtml(String content) {
        //<p>段落替换为换行
        content = content.replaceAll("<p.*?>", "\\r\\n");
        //<br><br/>替换为换行
        content = content.replaceAll("<br\\s*/?>", "\\r\\n");
        //去掉其他<>之间的内容
    }
}

```



文件 8

```

        content = content.replaceAll("\\<. * ?>", "");
        //还原 HTML
        //content = HTMLDecoder.decode(content);
        return content;
    }
}

```

4. 主界面

调用上述三种方法进行线程管理及线程间的通信,相关代码请扫描二维码文件 8 获取。

5. 今日新闻界面

```

public class NewsTodayAbilitySlice extends AbilitySlice {
    private ListContainer list1;
    private List < JsonParse.NewsToday > newsToday;
    @Override
    public void onStart(Intent intent) {
        super.onStart(intent);
        super.setUIContent(ResourceTable.Layout_ability_news_today);
        String data = intent.getStringParam("newsToday");
        //getStringParam 方法获取界面传递的参数
        newsToday = JsonParse.parseNewsToday(data);
        list1 = findComponentById(ResourceTable.Id_list1);
        //对应 XML 界面中的参数
        initListContainer();
    }
    //适配器代码从官方文档中获取
    private void initListContainer() {
        SampleItemProvider1 sampleItemProvider1 = new SampleItemProvider1(newsToday, this);
        list1.setItemProvider(sampleItemProvider1);
    }
    @Override
    public void onActive() {
        super.onActive();
    }
    @Override
    public void onForeground(Intent intent) {
        super.onForeground(intent);
    }
}

```

6. 历史上的今天界面

```

public class HistoryTodayAbilitySlice extends AbilitySlice {
    private ListContainer list;
    private List < JsonParse.HistoryToday > historyToday;
    @Override
    public void onStart(Intent intent) {
        super.onStart(intent);
        super.setUIContent(ResourceTable.Layout_ability_history_today);
        String data = intent.getStringParam("historyToday");
        //getStringParam 方法获取界面传递的参数
    }
}

```

```

        historyToday = JsonParse.parseHistoryToday(data);
        list = findComponentById(ResourceTable.Id_list);
        initListContainer();
    }
    //适配器代码从官方文档中获取
    private void initListContainer() {
        SampleItemProvider sampleItemProvider = new SampleItemProvider (historyToday,
this);
        list.setItemProvider(sampleItemProvider);
    }
    @Override
    public void onActive() {
        super.onActive();
    }
    @Override
    public void onForeground(Intent intent) {
        super.onForeground(intent);
    }
}

```

7. ListContainer 适配器

ListContainer 每行可以是不同的数据,因此需要适配不同的数据结构,使其都能添加到 ListContainer 上,相关代码请扫描二维码文件 9 获取。



文件 9

5.4 成果展示

打开 App,应用初始界面如图 5-5 所示;今日新闻功能如图 5-6 所示;历史上的今天功能如图 5-7 所示;随机笑话功能如图 5-8 所示。



图 5-5 应用初始界面



图 5-6 今日新闻功能

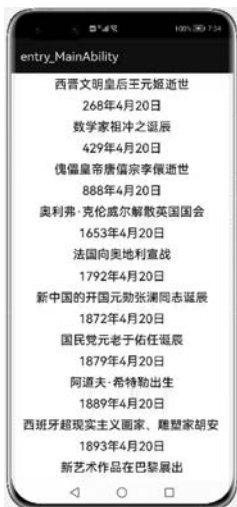


图 5-7 历史上的今天功能



图 5-8 随机笑话功能