《移动互联网编程实践》

课程大作业

版本:1.1



华为技术有限公司

# 作业导读

## 作业导读

本篇有2个作业可供选择，分别为商城和信息流，旨在帮助学生熟练的掌握程序设计以及如何分析和设计程序的功能。

完成作业需经历五个阶段：分析设计、绘制商品页面、加载页面、支持列表滑动、支持更多品类。

当然，如果同学们有自己的想法完成自己设计的软件内容，也可以自由选题。但注意，自由选题的同学们额外需要自己仿照下述内容，设计绘制必要的UML图表、明确自己选题的内容任务难点、详述软件理想效果。

使用环境：本作业要求在鸿蒙真机/模拟器上进行，以此让学生体验解决问题的过程。

使用操作系统：HarmonyOS Next。

使用工具：DevEco Studio。

作业提交要求：

（1）需求分析：针对设计制品的应用需求，详细描述其可能涉及的功能性与非功能性的各项需求，通过绘制必要的UML图表（如用例图、类图、状态图等），确保需求明确、完整、可验证。

（2）架构设计：涵盖系统总体设计架构、各模块详细设计、数据库设计、接口设计、界面设计等内容，需绘制必要的UML图表（如构件图、活动图、顺序图等），为系统开发提供清晰指导。

（3）系统测试：展示测试的全方位规划、具体用例设计以及最终测试结果和结论，证明系统达到预期质量标准，并且可进行量化测试的内容都需要进行量化验证。

（4）项目管理：假设项目执行期为2个月，项目组成员为分组实际人员情况，策划包括项目计划、团队成员分工说明、进度跟踪记录，以及风险管理等内容的项目管理文档，要求体现项目管理的规范性和有效性。

（5）系统原型与实现代码：系统原型的关键技术评估点包括：提交材料：代码和数据文件。

（6）各组组长额外提供1份关于各个组员的角色分工及其工作情况说明，组长和所有组员每人提供一份400字以内的个人实践心得。

（7）功能展示视频，需要录制一份展示功能实现完整的视频。

# 商城

## 功能要求

HarmonyOS ArkTS提供丰富的接口和组件，开发者可以根据实际场景和开发需求，选用不同的组件和接口。我们使用Scroll组件、List组件以及LazyForEach组件实现一个商品列表的页面，并且拥有下拉刷新、懒加载和到底提示的效果。

核心业务： ①商品浏览（有缩略图、关键词介绍、首页瀑布流、高矮图共存等）

②分类Tab（三步内找到目标品类：点 Tab→看子类→进列表浏览商品）

③搜索（输入关键词→点结果）

④筛选（让用户“按意愿排序”，同时可叠加多重过滤条件，缩小选择范围）

## 作业要求

目的：完成上述功能实现

作业涉及如下**技术要求：**掌握基础的ArkTs程序编写，掌握基础的DevEco Studio使用，掌握基础的开发编译以及调试的使用，禁止在作业内容中混用JS/JAVA等其他编程语言。

相关概念

[Scroll](https://developer.huawei.com/consumer/cn/doc/harmonyos-references-V5/ts-container-scroll-V5" \t "_blank)：可滚动的容器组件，当子组件的布局尺寸超过父组件的视口时，内容可以滚动。

[List](https://developer.huawei.com/consumer/cn/doc/harmonyos-references-V5/ts-container-list-V5" \t "_blank)：列表包含一系列相同宽度的列表项。适合连续、多行呈现同类数据，例如图片和文本。

[Tabs](https://developer.huawei.com/consumer/cn/doc/harmonyos-references-V5/ts-container-tabs-V5" \t "_blank)：一种可以通过页签进行内容视图切换的容器组件，每个页签对应一个内容视图。

[LazyForEach](https://developer.huawei.com/consumer/cn/doc/harmonyos-guides-V5/arkts-rendering-control-lazyforeach-V5" \t "_blank)：开发框架提供数据懒加载（LazyForEach组件）从提供的数据源中按需迭代数据，并在每次迭代过程中创建相应的组件。

## 代码介绍 ，

├──entry/src/main/ets // 代码区

│ ├──common

│ │ └──CommonConstants.ets // 常量集合文件

│ ├──entryability

│ │ └──EntryAbility.ets // 应用入口，承载应用的生命周期

│ ├──pages

│ │ └──ListIndex.ets // 页面入口

│ ├──view

│ │ ├──GoodsListComponent.ets // 商品列表组件

│ │ ├──PutDownRefreshLayout.ets // 下拉刷新组件

│ │ └──TabBarsComponent.ets // Tabs组件

│ └──viewmodel

│ ├──InitialData.ets // 初始化数据

│ └──ListDataSource.ets // List使用的相关数据加载

└──entry/src/main/resources // 资源文件目录

参考链接：

<https://developer.huawei.com/consumer/cn/codelabsPortal/carddetails/tutorials_NEXT-List>

### 页面布局

页面使用Navigation与Tabs做页面布局，使用Navigation的title属性实现页面的标题，Tabs做商品内容的分类。

页面分为“精选”、“手机”、“服饰”、“穿搭”、“家居”五个模块，推荐同学们将“精选”部分单独编写代码，其余模块使用ForEach遍历生成。无论生成方式，需要保证各模块Tab样式相同，各Tab中需要添加SelectedComponent、Banner、WaterfallList、RefreashLayout组件，注意组件之间的层叠关系以及数据与事件的传递。

### 商品列表的懒加载

使用Scroll嵌套List做长列表，让各组件的高度保持统一，实现Scroll与List的联动。

商品列表往往数据量很多，如果使用ForEach一次性遍历生成的话，性能不好，所以这里使用LazyForEach进行数据的懒加载，推荐使用剩余6条商品时就提前进行拉取。

当向下滑动，需要加载新的数据时，再将新数据加载出来，生成新的列表。通过onTouch事件来触发懒加载行为，当商品列表向下滑动，加载新的数据。

推荐使用20条作为首次加载，后续每次20条作为增量更新，保持≥40fps帧率，中间不掉帧即可。

### 下拉刷新与到底提示

下拉刷新同样使用TouchEvent做下拉判断，当下拉的偏移量超出将要刷新的偏移量时，就展示下拉刷新的布局，同时使用条件渲染判断是否显示下拉刷新布局。这里额外提出要求，需要考虑异常保护，下拉的动画如果遇到了网络延迟如何应对。效果如图所示：

列表到底提示“已经到底了”并回弹的效果使用了Scroll的edgeEffect来控制回弹，尽量使用系统级弹簧，就无需手写动画，但需要自己控制回弹距离。效果如图所示：



最终效果如下：



代码请下载：

<https://developer.huawei.com/consumer/cn/codelabsPortal/carddetails/tutorials_NEXT-List>

# 信息流

## 功能要求

本场景解决方案主要面向于新闻类页面开发人员，指导开发者从零开始构建一个新闻类首页面，包含地址选择、tabs和tabContent切换的动态图标和流畅动效、下拉刷新上拉加载、首页feed流等常见功能，及功能的流畅体验。

核心业务： ①地址选择（要求支持最先使用GPS返回当前城市，之后可以进行手动切换，运用首字母索引导航或模糊搜索）

②Tab&TabContent动态图标+流畅动效（顶部Tab支持手动拖拽排序，底部具有多项主导航，切换时跟随手部操作，添加图标的自适应）

③下拉刷新/上拉加载（首页 Feed 支持下拉刷新整页，上拉加载更多；刷新/加载均有 Lottie 动画）

④首页Feed流（图文卡片+视频卡片混排，视频自动播放、支持无限列表、懒加载、预加载 3 屏）

## 作业要求

目的：完成上述功能实现

作业涉及如下**技术要求：**掌握基础的ArkTs程序编写，掌握基础的DevEco Studio使用，掌握基础的开发编译以及调试的使用，禁止在作业内容中混用JS/JAVA等其他编程语言。

## 代码介绍

├──entry/src/main/ets/

│ ├──common

│ │ └──lottie // 动画

│ ├──constants

│ │ ├──BreakpointConstants.ets // 断点常量

│ │ ├──CommonConstants.ets // 常用常量

│ │ └──HomeConstants.ets // 主页常量

│ ├──entryability

│ │ └──EntryAbility.ets // Ability的生命周期回调内容

│ ├──pages

│ │ ├──CitySearch.ets // 城市查询

│ │ └──Index.ets // 首页

│ ├──util

│ │ ├──BreakpointType.ets // 断点类型

│ │ └──ResourceUtil.ets // 路由数据

│ ├──view

│ │ ├──CityView.ets // 城市列表组件

│ │ ├──Home.ets // 主页组件

│ │ ├──HomeContent.ets // tab内容组件

│ │ ├──HomeHeader.ets // 主页头部组件

│ │ ├──NewsChannel.ets // 新闻渠道组件

│ │ ├──PullToRefreshNews.ets // 拉取刷新新闻组件

│ │ ├──SearchView.ets // 搜索组件

│ │ └──TabBar.ets // 标签栏组件

│ └──viewmodel

│ ├──CityDetailData.ets // 城市详细数据

│ ├──NewsData.ets // 新闻数据

│ ├──NewsDataSource.ets // 新闻数据源

│ ├──NewsTypeModel.ets // 新闻类型模型

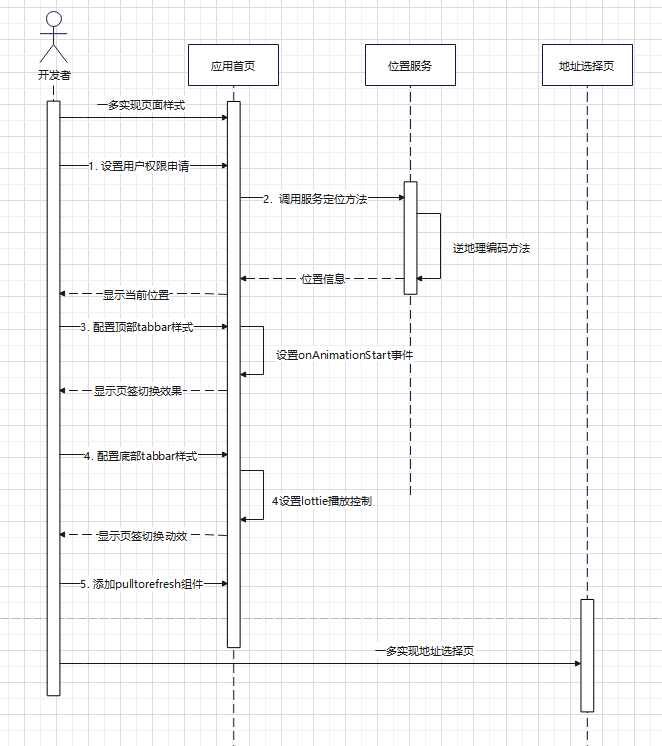
│ └──NewsViewModel.ets // 新闻视图模型

└──entry/src/main/resources // 应用静态资源目录

### 整体场景介绍

主要介绍用户操作应用的流程，展示用户进入首页通过页签切换页面内容，上拉加载和下拉刷新页面，从首页地址进入地址选择页更换地址等功能效果。

#### 实现应用的主要流程图



### 场景分析

实现方案如下表：

| **场景名称** | **描述** | **实现方案** |
| --- | --- | --- |
| 导航栏切换动效流畅 | 点击页签或者滑动切换页面，页签同步切换 | tab组件添加动画开始时触发事件 |
| 底部页签跳转精致流畅 | 底部页签切换具有动画效果 | 添加lottie动画 |
| 上拉加载下拉刷新 | 上拉加载更多的新闻内容，下拉刷新整个页面，均具有加载动效 | pullToRefresh组件 |
| 首页feed流 | 首页展示流畅图文列表 | 使用LazyForEach对子组件进行渲染，实现懒加载功能 |
| 地址选择页 | 提供地址选择，定位，地址首字母定位及模糊查询功能 | 位置服务与AlphabetIndexer组件 |

### 场景实现

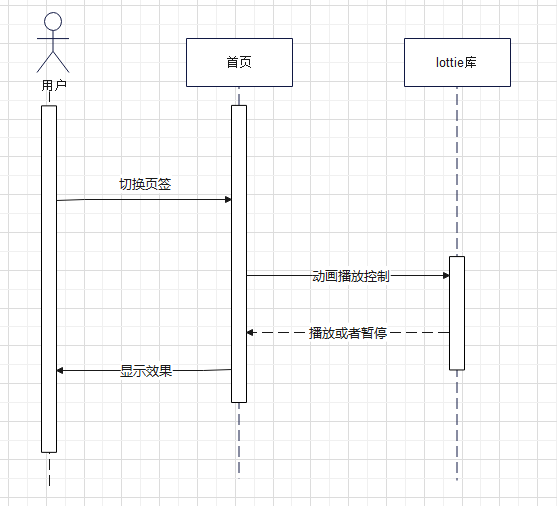
#### 导航栏切换动效流畅

通过添加tab组件动效触发事件，实现页面内容切换与页签样式切换的同步触发效果。

#### 底部页签跳转精致流畅

底部页签样式添加lottie动画使跳转精致流畅。

底部页签跳转功能时序图



#### 上拉加载下拉刷新

通过三方库组件pullToRefresh实现下拉刷新页面，上拉加载更多数据效果。

上拉加载下拉刷新功能时序图



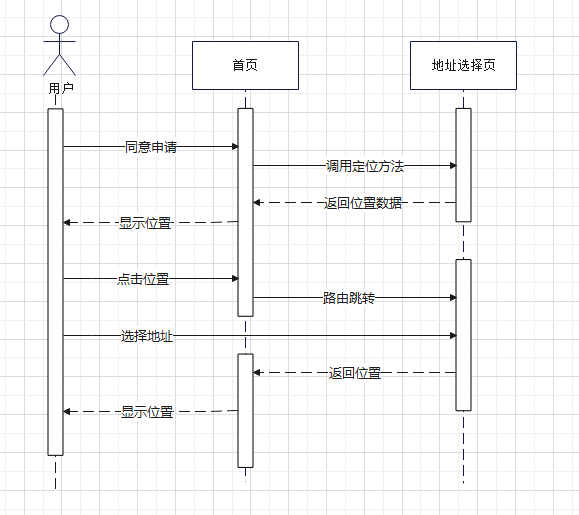
#### 首页feed流

通过懒加载实现首页feed流快速渲染与流畅滑动。

#### 地址选择页

使用原生位置服务实现定位功能，AlphabetIndexer组件实现地址首字母定位导航条。

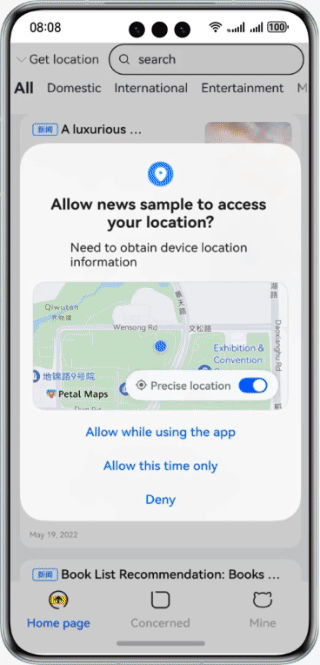
地址选择页效果功能时序图



更多内容可以参考：

<https://developer.huawei.com/consumer/cn/doc/best-practices-V5/bpta-news_homepage-V5>

最终效果如下：



代码请下载：

**<https://gitee.com/harmonyos_samples/fluent-news-homepage>**