电影资讯APP开发文档

## 项目简介

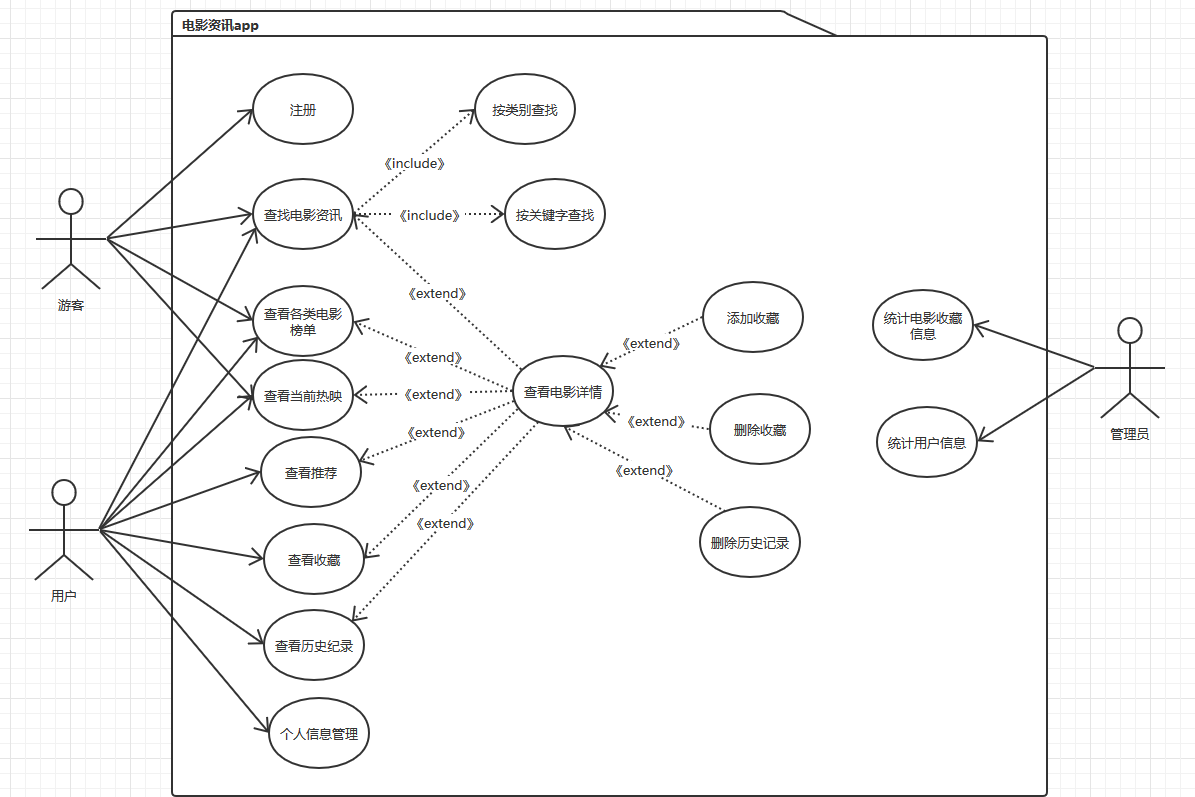
社会近年来的迅猛发展，使得我国人民的生活质量飞速提高，电影也逐渐成为了大众不可或缺的娱乐需求。本文论述的就是一个基于android移动客户端设计的影视信息类APP，借助目前国内应用最为普遍的安卓手机移动终端实现一个可以在手机端查看电影资讯的功能性APP，它可以向你提供关于你想了解的影片的详细信息，包括相关人员信息、评分、影评等，你可以通过关键字或分类进行搜索，你还可以查看各大影视榜单、建立自己的收藏列表、接收系统为你提供的个性化推荐等等。

## 项目开发方式

Android端APP，采用原生的开发模式，使用java语言，适用于Android API19及以上版本。

服务端使用node.js 的express框架，使用MongoDB进行数据存储。暂时没有部署到可以公网访问的服务器上。

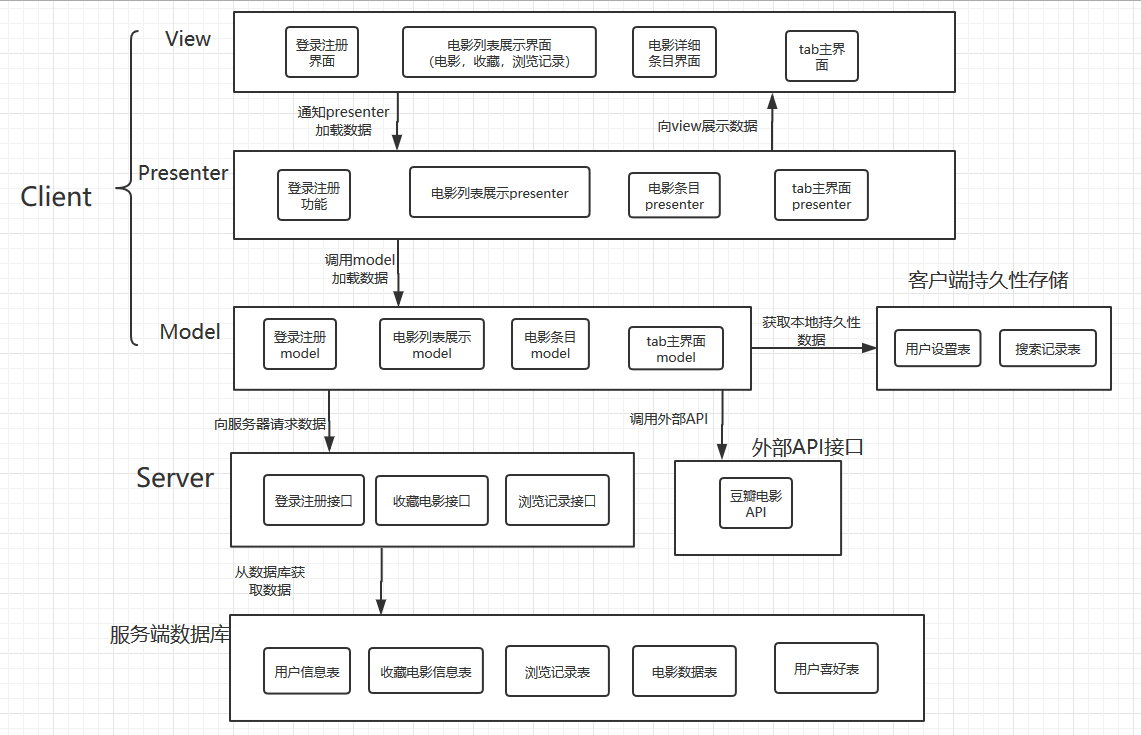
## 实现的用例

实现了图中共十五个用例

## 项目架构方式

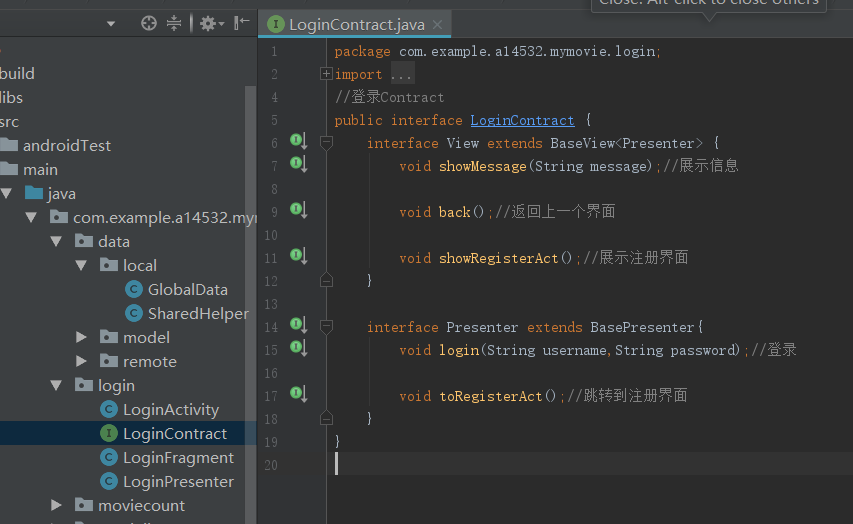
项目总体采用了CS架构，分为服务端与客户端

在客户端即Android端采用了MVP架构



## 项目代码风格

项目采用了MVP架构模式，使用了一个契约类Constract, 有关Presenter和View函数已经功能都注释在Constract中了。



## 网络访问

APP中使用了Retrofit+OkHttp框架实现网络访问功能，主要分为两部分，一部分是获取豆瓣电影API的电影数据，另一部分是与后端服务器进行访问，为用户登陆注册，收藏，浏览历史，推荐电影等操作提供接口。

## 数据存储

本系统的服务端数据是使用MongoDB数据库进行数据存储。它总共有users, collect, history, moviedata, typicaluser五张表，用来存储五种不同的类型与用途的数据。

客户端使用SharedPreference进行本地的持久性存储，SharedPreferences是Android平台上一个轻量级的存储类，用来保存应用的一些常用配置。在我们的系统中，主要是用它来存储用户名和用户头像，这样当用户关闭APP后重新打开，就可以免去登陆的步骤。

## 电影推荐算法实现

基于内容的推荐算法（Content-Based Recommendations CB）:

CB的过程一般包括以下三步：

物品表示（Item Representation）：为每个电影抽取出一些特征（也就是电影的年代，类型，语种等）来表示此电影；

特征学习（Profile Learning）：利用一个用户过去喜欢（及不喜欢）的电影的特征数据，来学习出此用户的喜好特征（即用户喜好表）；

生成推荐列表（Recommendation Generation）：通过比较上一步得到的用户喜好表与候选电影的特征，为此用户推荐一组相关性最大的电影。

基于协同过滤的用户画像推荐算法，包括以下三个步骤:

确定用户的属性：年龄段，性别，学历。

根据所有用户的收藏情况，生成用户画像（即各个性别，年龄段，学历的用户会喜欢什么样的电影），存在typicalUser用户画像表中。

生成推荐列表：通过用户画像中某类用户的电影喜好表来给用户推荐电影。

基于导演的电影推荐：

这是一个比较简单但实用的电影推荐算法，向用户推荐他收藏列表中的导演执导的其他电影。

## 使用的开源库

网络访问：Retrofit, Okhttp3

图片加载：gilde

列表展示：recyclerview

TAB搭建：com.flyco.tablayout.FlycoTabLayout

轮播图：com.bigkoo.connevnientbanner

搜索框：com.carson\_ho.searchLayout

悬浮按钮FAB:ddz.materialdesign.floatingActionButton

图标：MPAndroidChart