

天津财经大学

本科毕业论文



中文题目： 基于 ARCore 的 3D 购物 APP
的研究与实践

英文题目： Research and Practice of 3D Shopping
Based on ARCore

院系名称： 理工学院信息科学与技术系
专业班级： 计算机科学与技术
学 号： 2016111__
姓 名： 李 健
指导教师： 严冬梅

2020 年 5 月 8 日

内容摘要

随着网络技术的发展与电子商务的兴起，线上购物已经成为人们消费的主要渠道。但现有的基于 2D 图片的购物方式存在着缺陷，它的表现能力有限，不能向在线消费者提供足够的有关产品的直接信息，屏幕上的图片可能具有“欺骗性”。当没有足够的可用信息时，消费者将没有信心进行购买。3D 购物可以有效地解决这个问题。3D 购物可以让消费者直观感受到商品的大小与形状，传达更多更有用的信息。然而，在线上展示商品 3D 信息对设备要求较高，受限于技术，3D 购物一直没有得到有效的实践。近年来，随着 5G 与 AR 技术的发展，基于 AR 技术的 3D 购物方式可以流畅地运行在移动端，3D 购物的实践条件已经成熟。

本文尝试将 AR 技术与线上购物方式结合，以解决当前电子商务模式的缺陷。使用 Google 公司提供的 AR 开发工具 ARCore 和 Flutter 移动应用开发技术实现一款移动端基于 Android 的 3D 购物 APP。本文实现的 APP 将以家具售卖为例，尝试提出一种新的销售形式——“购物计划”，以适应使用 AR 技术的 3D 购物方式。

关键词：3D 购物；体验式营销；ARCore；Flutter；Android

Abstract

With the development of network technology and the rise of e-commerce, online shopping has become the main channel of people's consumption. But the existing 2D-based shopping method is flawed, its performance is limited, it does not provide online consumers with enough direct information about the product, the image on the screen can be deceptive. When there is not enough information available, consumers will not have the confidence to make a purchase. 3D shopping can effectively solve this problem. 3D shopping allows consumers to visualize the size and shape of the product, conveying more and more useful information. However, displaying product 3D information on the internet requires higher equipment requirements, limited by technology, 3D shopping has not been effective practice. In recent years, with the development of 5G and AR technology, 3D shopping mode based on AR technology can run smoothly on the mobile side, 3D shopping practice conditions have matured.

This paper attempts to combine AR technology with online shopping to solve the defects of the current e-commerce model. Use ARCore, a Google-supplied AR development tool, and Mobile App Development technology to implement a mobile Android-based 3D shopping APP. The APP implemented in this paper will use furniture sales as an example to try to propose a new form of sales - "shopping plan" to adapt to the use of AR technology 3D shopping.

Key Words: 3D Shopping; Experiential marketing; ARCore; Flutter; Android;

目 录

一、绪论.....	1
(一) 研究背景及现状	1
(二) 研究内容	2
(三) 研究目的与意义	2
(四) 技术路线	3
(五) 论文结构	3
二、相关理论与技术.....	4
(一) ARCore 技术简介	4
(二) Flutter 技术简介	4
(三) 开源 3D 模型平台	5
三、系统分析.....	6
(一) 总体需求分析	6
(二) 系统功能结构设计	1
四、系统设计.....	9
(一) 系统架构设计	9
(二) 系统业务流程设计	10
(三) 主要模块功能设计	10
(四) 购物计划功能详述	13
(五) 数据库设计	14
五、系统实现.....	16
(一) 单模型 AR 展示功能实现	16
(二) 多模型 AR 展示功能实现	21
(三) 购物计划与富文本解决方案	22
(五) Flutter 与 Java 的混合开发.....	25

六、总结与展望..... 27

 （一）课题总结 27

 （二）不足与展望 27

基于 ARCore 的 3D 购物 APP 的研究与实践

一、绪论

（一）研究背景及现状

1. 研究背景

随着网络技术的发展与电子商务的兴起,目前线上购物已经成为人们主要的消费渠道。据统计,2013 年至 2018 年间,中国网购交易金额从 2679 亿元增长至 57370 亿元,复合增长率为 84.6%。仅在 2019 上半年,超四成中国消费者每周一次及以上线上购物,25%的消费者每月 2-3 次,16.7%的消费者每月一次线上购物,15.3%的消费者每季 1-3 次线上购物,仅有 1.1%的消费者每季 1 次以下线上购物,电子商务拥有蓬勃的发展动力,线上购物是未来发展的趋势。但基于 2D 图片展示的电子商务方式存在着缺陷,受限于仅仅依靠图片、文字、视频的表现形式,不能向消费者提供足够的有关商品的直接信息,屏幕上的图片可能具有欺骗性。当没有足够的可用信息时,消费者将没有信心进行购买。为了解决这一问题,研究者们提出了 3D 购物解决方案。3D 购物是一项利用数字化手段,在线上向消费者展示商品虚拟模型的技术,通常使用虚拟现实(Virtual Reality,简称 VR)与增强现实(Augmented Reality,简称 AR)作为实现的技术手段。

近年来 AR 技术发展迅速,苹果公司推出了 ARKit, Google 公司推出 ARCore,使得复杂的 AR 模型得以在算力不足的移动设备实现,并提供了动态追踪、环境理解、光估计等丰富的功能。这些技术的推出使得 AR 的呈现方式不再是孤零零的图片,而是生动的,可以与环境交互的模型。2019 年是 5G 的商用元年,5G 的出现为 AR 的发展带来了更大的机遇,仅在 AR 与电子商务融合的 3D 购物领域,5G 的高速性使得 AR 得以在移动设备上呈现更精细的模型,从而更加拟真,带来更具沉浸式的购物体验。

2. 研究现状

使用 AR 作为 3D 购物解决方案(简称 AR 购物),解决传统线上购物缺陷和增强购物体验一直以来深受学者与工程师们的关注,在国内外众多学者都在研究这一问题。

矫春龙在论文“基于 Kinect 的 3D 虚拟试衣系统”中实现了增强现实在服装零售行业的应用,利用 AR 技术,消费者在没有实际商品的情况下,试穿不同的衣服。陈若雯在论文“电子商务中基于 AR 的家具展示平台的设计与实现”中实现了利用 AR 展示家

具。除了在技术上的探索，赵小军在“增强现实网购中的技术接受心理模型：基于大尺度空间环境的 B2C 系统研究”中通过调查论述了消费者对 AR 购物的消费意愿强于传统购物方式，并指出在 AR 与电子商务的结合中体现了体验式营销的影子。除了学术界的探索研究，工业界也对 AR 与电子商务的融合进行了广泛实践。“宜家家具”在其移动 APP “IKEA Place” 提供了家具在家实景体验、摆放的 AR 功能，但由于建模的成本，仅提供了少量商品。“美图有颜”是一个美妆电商平台，提供了 AR 试妆功能。不仅是美图、京东这样的电商平台，欧莱雅、雅诗兰黛以及梅西百货这样的零售商也开始使用 AR 美妆提高其销售转化。在国外一款名叫“Houzz”的家具电商平台十分火爆，它提供 AR 展示家具并销售。无论在国内还是国外，AR 与电子商务的融合已经成为了趋势。

（二）研究内容

1. AR 与电子商务融合实现 3D 购物

本文研究的 3D 购物是 AR 与电子商务的融合，利用 Google 公司提供的 AR 移动设备开发工具包——ARCore，实现将线上售卖的商品以 3D 模型的形式虚拟到现实中的功能。使得消费者可以更全面的、直观的了解商品的信息，如形状、大小、颜色。同时将虚拟商品与现实环境做比较，获得更好得购物体验。

2. 适应 AR 购物模式，增强购物体验的功能设计

区别于传统线上购物，消费者购买物品，加入购物车、下订单的流程，在基于 AR 技术的 3D 购物应用中消费者将更关注于与商品交互的体验而不是简单的下单，所以传统线上购物的购物车，订单流程已不能满足新的 3D 购物需求。在商品交互体验中，消费者关注的不仅仅是商品与现实环境的交互，还包括商品与商品间的交互，本文尝试提出一种新的销售方式——“购物计划”，引入体验式营销理念，增强购物体验。“购物计划”功能支持消费者体验其他消费者组合的购物方案，将一系列商品导入现实与环境进行比较，通过身临其境的购物体验增强消费者的购买意愿。以家具售卖为例，通过导入其他消费者组合的家具布置方案，将这些家具导入现实与房间环境进行比较，从而更好的判断是否符合自己的购买意愿。

（三）研究的目的是与意义

为解决基于 2D 图片的传统线上购物无法为消费者带来完美购物体验的问题上，本文在现有 3D 购物实践基础上，将 ARCore 技术应用于 3D 购物，在移动端实现解决传

统线上购物购物体验感差的问题，弥补了现有电子商务模式的缺陷。本课题的研究不仅弥补了 3D 购物应用在实际上的空缺，而且应用开发的编程方案和功能设计也可为其他开发者提供借鉴。

（四）技术路线

1. 前端 APP

本文实践系统的前端基于 Android 系统开发，采用原生 Android 与最新的全平台开发框架 Flutter 混合开发的方式。本系统的亮点在于 AR 与电子商务的融和，AR 功能由 Google 公司的 AR 开发工具 ARCore 提供。ARCore 基于原生 Android，使用 Java 语言开发。前端 APP 的其他功能界面则由 Flutter 框架实现。Flutter 基于 Dart 语言，是全平台开发框架，有利于未来对系统的拓展，仅需改变 AR 功能的支持为 ARKit 便可在 iOS 端流畅运行。

2. 基于 Linux 服务器的 Java 后台

后台服务并不是本文实践的重点，后台服务使用阿里云提供的 Ubuntu 系统，后台框架使用 Java，通过 JavaServlet 联通前端请求与后台的数据库。数据库使用 MySQL 关系型数据库，存储商品和用户的信息。

（五）论文结构

第一部分是论文的绪论，介绍了本文的研究背景、目标与意义，阐述了国内与国外在 AR 与电子商务融合领域所作的研究与实践，并介绍了实践选用的技术与原因。

第二部分是相关理论技术介绍，这一节将介绍本文会用到的相关技术，包括 ARCore、Flutter 和 3D 模型开源平台。主要介绍他们的功能特性、使用方法与选择原因。

第三部分是系统分析，这一节将进行需求分析，介绍符合消费者期望的 3D 购物 APP 应具有怎样的功能。

第四部分是系统设计，这一节主要介绍本文实践的 3D 购物 APP 的整体架构设计，主要模块功能设计，并详细介绍“购物计划”功能的优势与出现的原因。

第五部分是系统的实现，这一节会介绍主要功能与关键技术的实现思路与代码。

第六部分是总结和展望。

二、相关理论与技术

（一）ARCore 技术简介

ARCore 是谷歌（Google）公司推出的增强现实（AR）开发工具包，官方网站是 <https://developers.google.com/ar/develop/>。它支持在 Android, iOS, Unity, Unreal 多种平台中开发使用，它本身封装了一套本地 API，通过它可以实现一些最基础的 AR 效果，包括：增强图像、增强脸部、云锚点。ARCore 最重要的特点是，它可以运行在移动设备和网页端这种计算能力不足的平台，而且呈现效果不打折扣，虚拟内容看起来就像位于真实表面或者处于现实世界中的位置一样。它拥有一些非常强大的特性从而达到了这样的效果。

1. 运动追踪

当手机移动时，ARCore 会使用称为并行测距与映射（或 COM）的过程来理解手机与周围世界的相对位置，通过检测 Camera 图像中的视觉差异特征计算位置变化，并利用这些视觉信息与设备 IMU 的惯性测量结果估测出设备随着时间推移而相对于真实世界的姿态。使得开发者能从正确的角度渲染虚拟内容，使虚拟内容看起来像现实世界的一部分。运动追踪使得 ARCore 的 AR 效果不再是镜头前简单的模型拖拽，而是可以在真实世界的不同角度观察的虚拟内容，沉浸感大大提升。

2. 环境理解

ARCore 可以查找看起来位于常见水平或垂直表面上的特征点集群，并将这些表面用作放置物体的平面，并通过检测特征点和平面来不断改进它对真实世界环境的理解。环境理解使得虚拟内容可以与现实环境做简单互动，虚拟内容会附着在真实世界存在的平面上，真实感大大提升。

3. 光估测

ARCore 可以检测物体周围环境光线的相关信息，并对给定摄像头图像进行平均光强度和色彩校正，使用同样的光线照亮虚拟物体。这可以让虚拟物体在现实环境中显示的更加真实。

ARCore 可以运行在移动平台，而且效果惊艳，最重要的是它完全面向开发者开源。使用 ARCore 可以无需投入太多的成本就能取得很好得效果，这是本课题选择它得原因。

（二）Flutter 技术简介

随着开发技术的成熟，越来越多的厂商开始追寻跨平台技术解决方案，从 Cordova、

Xamarin，到 React Native 和 Weex，每种框架都有自己的优点与缺点，但目标都是提升应用的开发效率，降低研发成本，一套代码运行多个平台。Cordova 通过 webkit 进行 UI 渲染，因此其性能和原生相比还有很大差距。React Native 和 Weex 基于 js 开发，需要通过桥接器来驱动原生渲染，导致其在性能上与原生还是有一定差距。基于以上框架的缺点，Flutter 框架应运而生。Flutter 是谷歌 2015 年推出的移动 UI 框架，是一跨平台技术解决方案，使用 Dart 语言，可以快速在 iOS 和 Android 系统上构建高质量的媲美原生开发的用户界面。Flutter 可以与现有原生代码一起开发工作，无需重新开发。Flutter 正在被世界上越来越多的开发者和组织使用，并且是完全免费和开源的。

Flutter 拥有五大优秀特性，是使得它成为本课题前端开发首选技术的原因。

1. 热重载

由于 Flutter 选用了 Dart 作为其开发语言，Dart 既可以是 AOT (Ahead Of Time) 编译，也可以是 JIT (Just In Time) 编译，其 JIT 编译的特性使 Flutter 在开发阶段可以达到亚秒级有状态热重载，只需使用开发工具将代码写入设备一次，就可以随时更改并在设备上得到体现，无需向过去一样每一个改变都要重新加载，从而大大提升了开发效率。

2. 性能优越

使用自带的高性能渲染引擎 (Skia) 进行自绘，渲染速度和用户体验堪比原生。富有表现力的精美 UI。使用 Flutter 内置美丽的 Material Design 和 Cupertino (iOS 风格) widget、丰富的 motion API、平滑而自然的滑动效果和平台感知，为用户带来全新体验。而且所有的 API 都是响应式的。

3. 支持原生功能与 SDK

Flutter 虽然使用不同于 iOS 和 Android 平台的开发语言却允许复用现有的 Java、Swift 或 ObjC 代码，可以访问原生系统的功能与 SDK，让应用变得更加强大易用。这是本课题选用它的原因之一，Flutter 可以和 ARCore 有效结合。

4. 丰富的第三方库

依托于谷歌公司，Flutter 有着强大的社区，全世界的开发者充实着它的代码库。通过 <https://pub.flutter-io.cn>，可以轻松的使用众多功能的第三方库，大大增强开发效率。

Flutter 拥有精美的 UI 和高效的开发效率，并且可以与 ARCore 技术完美结合，这是本课题选择它的原因。

(三) 开源 3D 模型平台

本文实践主要使用了两个 3D 模型平台的资源。Google 的 poly 和微软的 3D library。

Poly 是一个 3D 模型网站，同时也是一个在线 library，人们可以在其中浏览，共享和重新混合 3D 资源。支持所有格式的 3D 模型上传。许多模型根据 CC BY 许可获得许可，这意味着开发人员可以在创建者允许的情况下免费在其应用程序中使用它们。大部分模型都可以免费下载，官网是 poly.google.com。Poly 拥有各个分类的 3d 模型包括本文实践需要的家具模型，poly 模型的特点是卡通化，不够拟真，但 poly 支持 ARCore，并为开发者提供了 API，使得开发者可以直接在程序中向 poly 服务器请求资源，而无需搭建自己的服务器。

微软的 3D library 由 win10 自带的 3D View 提供，资源相对较少，但模型拟真，用 AR 展示的效果更加好。下载资源的方式很简单，通过 3D view 可以直接下载。

三、系统分析

（一）总体需求分析

目前线上购物已经成为人们消费的主要渠道，越来越多的人习惯于在网上购买物品，然后等待快递的送达。随着电子商务领域技术的成熟，消费者不再满足于在线上简单的购买到商品，开始期待在线上获得更多商品信息，得到更好的购物体验，这是基于 2D 图片的电子商务模式所缺乏的。赵小军在其博士学位论文《增强现实网购中的技术接受心理模型：基于大尺度空间环境的 B2C 系统研究》中指出不同于传统基于 2D 图片展示的线上购物，AR 购物使得消费者与商品同步于同一空间关系中，这种消费方式的改变体现了现代体验式营销的影子，研究表明消费者对 AR 购物的购买意愿强于传统线上购物。本文研究 3D 购物 APP 是 AR 与电子商务的结合，其特点是利用 AR 技术展示线上商品的 3D 模型，使得消费者可以更直观地感受到更多更有效的信息，从而收获更好的购物体验。本文实践的 APP 是在传统电子商务 APP 的基础上增加了 AR 展示功能和一个新的商品浏览销售方式——“购物计划”。

（二）系统功能结构设计

1. 系统总体功能结构

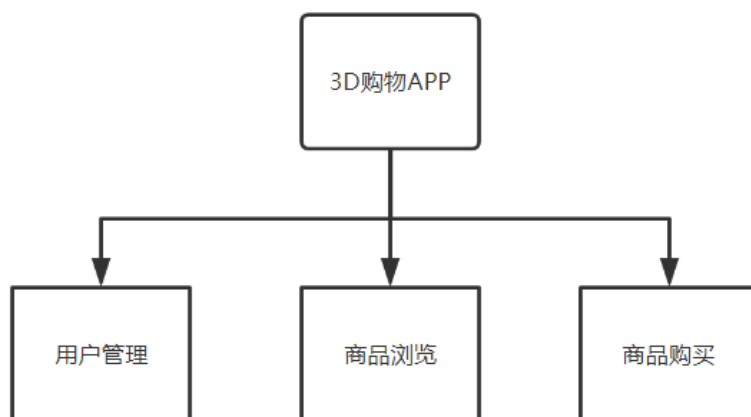


图 3.1 系统总体功能结构图

2. 用户管理功能分析

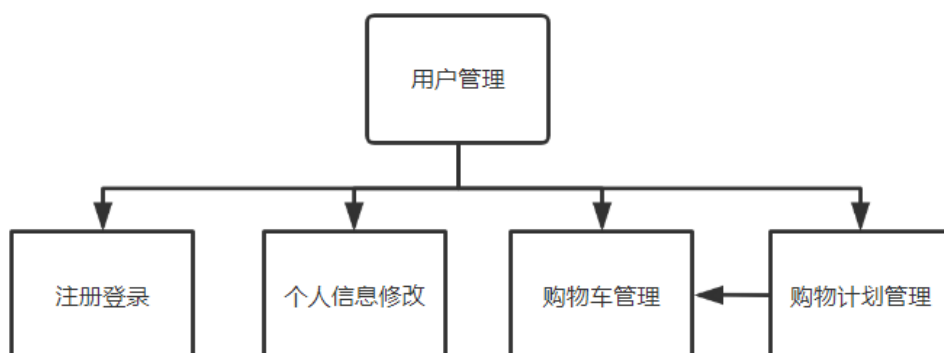


图 3.2 用户管理功能结构图

用户管理功能实现注册登录和个人信息的修改，个人信息的修改包括用户名，密码重置等功能。

购物车管理实现对购物车内的商品进行增添与删除，并负责沟通后台下单的功能。

购物计划管理实现对购物计划的增添，删除和对购物计划内容进行修改，并实现将购物计划内的商品加入购物车的功能。

3. 商品浏览功能分析

商品浏览功能提供商品的分类展示，从而方便消费者快速的找到自己所需的商品。展示的信息包括图片、名称、价格等简要信息，方便消费者挑选。

商品详情功能向消费者展示商品的文字、图片信息，同时提供 AR 展示功能的接口。

购物计划是本文实践的特色功能。购物计划要向消费者展示由官方或其他消费者分享的购物方案。在本课题的实践中，购物计划会向消费者展示房间内家具的布置方案。

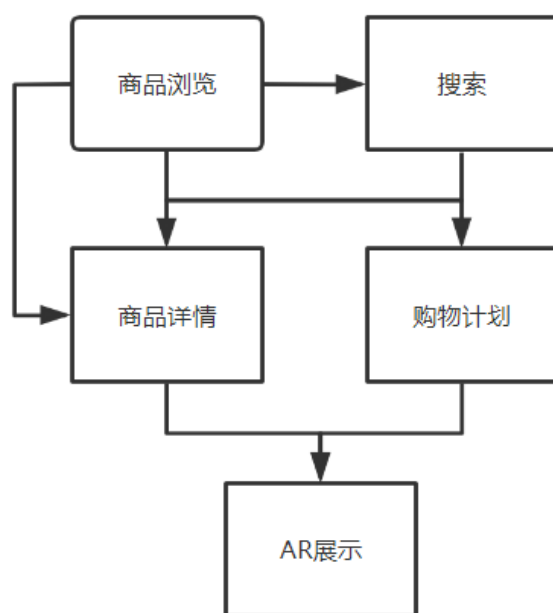


图 3.3 商品浏览功能结构图

AR 展示是本课题的特色功能。如上图所示 AR 展示功能有 2 个链入箭头，在本次实践中 AR 展示功能有两种，单一的商品展示和多商品展示。单一的商品展示由商品详情链入，提供将商品 3D 模型虚拟到现实空间的功能，向消费者展示更具体的商品信息，并允许消费者将线上的商品与实际环境作比较，判断是否符合自己的购买意愿。多商品展示由购物计划链入，与单商品的区别在于可以将多个商品虚拟到现实空间中，这使得商品模型不仅可以与现实空间作比较还可以与其他商品模型作比较。其意义在于，消费者不仅可以参考其他消费者的布置方案，还可以在现实空间中直接体验，从而找到适合自己房间的布置方案。

4. 商品购买功能分析

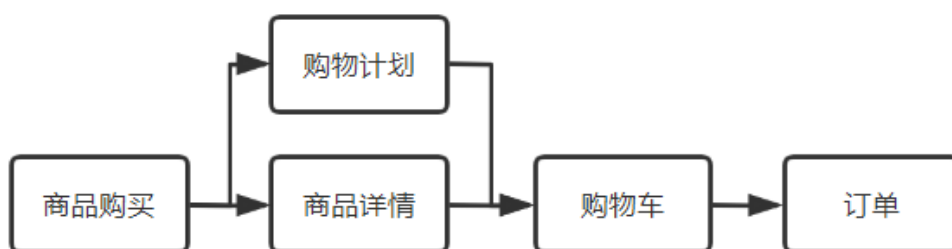


图 3.4 商品购买功能图

商品购买功能实现了从商品详情和购物计划中将商品加入购物车，并从购物车中下单，沟通后台生成订单的功能。

四、系统设计

(一) 系统架构设计

如图 4.1 所示本文实践的 3D 购物 APP 采用 C/S 架构,C/S 架构的特点是交互性强,响应速度快,能处理大量的数据。在本次实践中前端 APP 负责与消费者进行交互,而具体功能的实现则由服务器完成(AR 功能除外),服务器负责操纵数据库向前端应用传递数据。这样的设计可以避免消费者直接修改数据库,保障数据的安全性。



图 4.1 C/S 架构图

本文的实践更加关注 AR 与电子商务的结合对消费者浏览和购买体验上的影响,所以本文实践的系统更加关注前端应用的功能,服务器程序则负责对前端功能提供支持。前端应用中的注册登录、修改个人信息、搜索、商品浏览,购物计划探索、购物车管理、购物计划管理和下单功能都需要服务器后台的支持。本文实践的系统整体架构如图 4.2 所示。

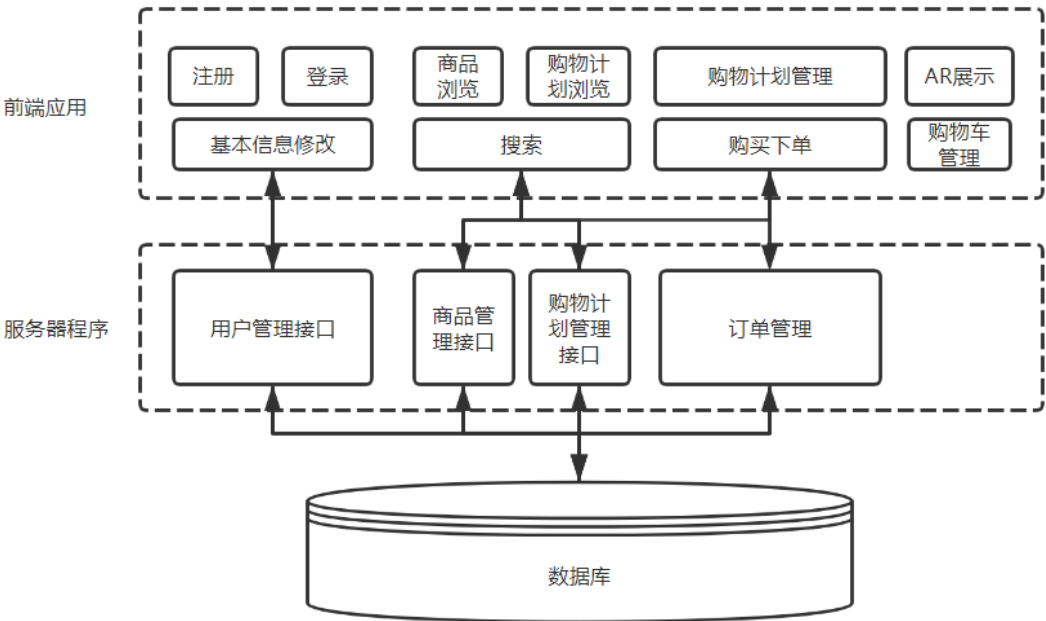


图 4.2 系统整体架构

（二）系统业务流程设计

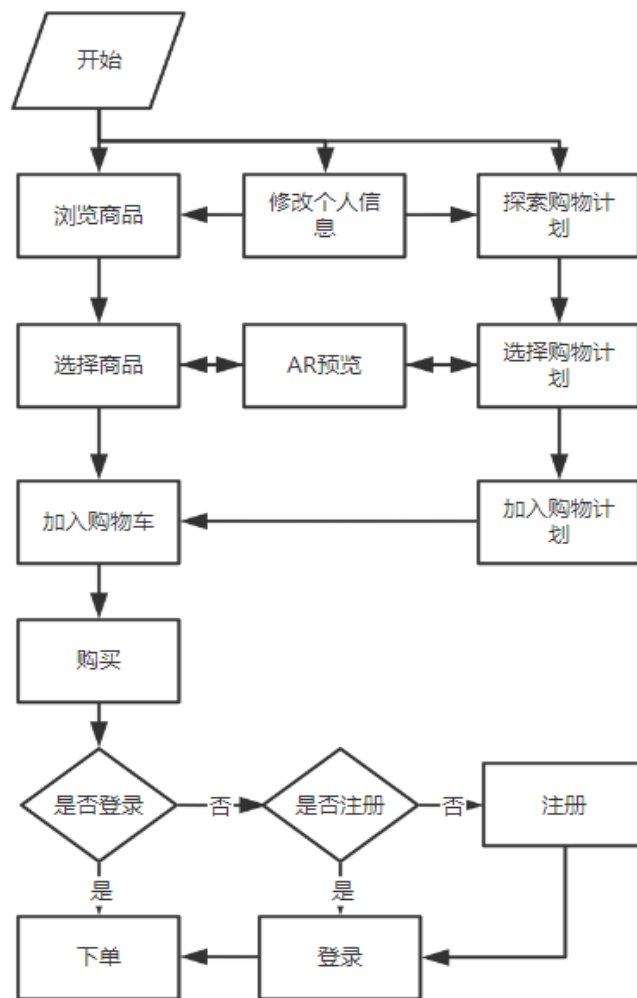


图 4.3 系统业务流程

（三）主要模块功能设计

1. 登录注册模块设计

本课题实践的 APP 无需启动时登录，只有在下订单或修改个人信息和访问购物计划时登录，同时也支持主动登录。如图 4.4 所示，当用户登录时需要输入 ID 与密码，登录成功会跳转到程序的主页。注册功能的接口在登录页中，注册需要输入用户名，手机号与密码。注册成功会跳转到登录页重新登录。

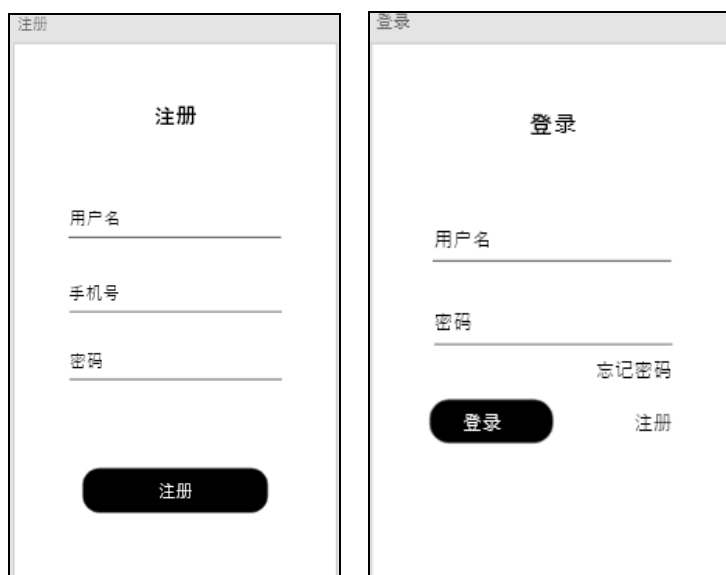


图 4.4 登录注册功设计图

2. 用户管理模块设计

如图 4.5 所示，个人信息管理放在侧边栏中，用户可以更改用户名，手机号，重置密码与查看历史订单。用户通过向右滑动进入侧边栏，侧边栏中包含用户头像，修改个人信息，购物计划，查看订单，重置密码。点击修改个人信息可以修改用户名与手机号，点击重置密码可以修改密码，点击订单可以查看历史订单。

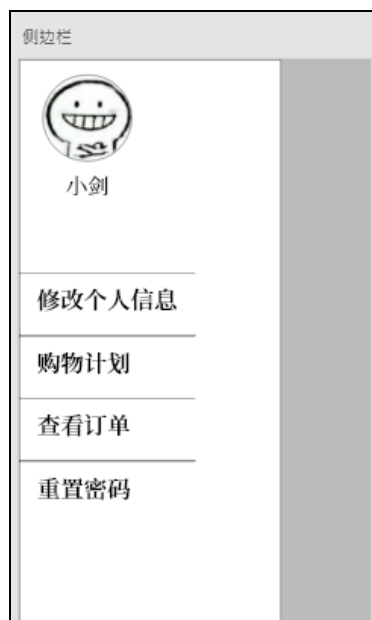


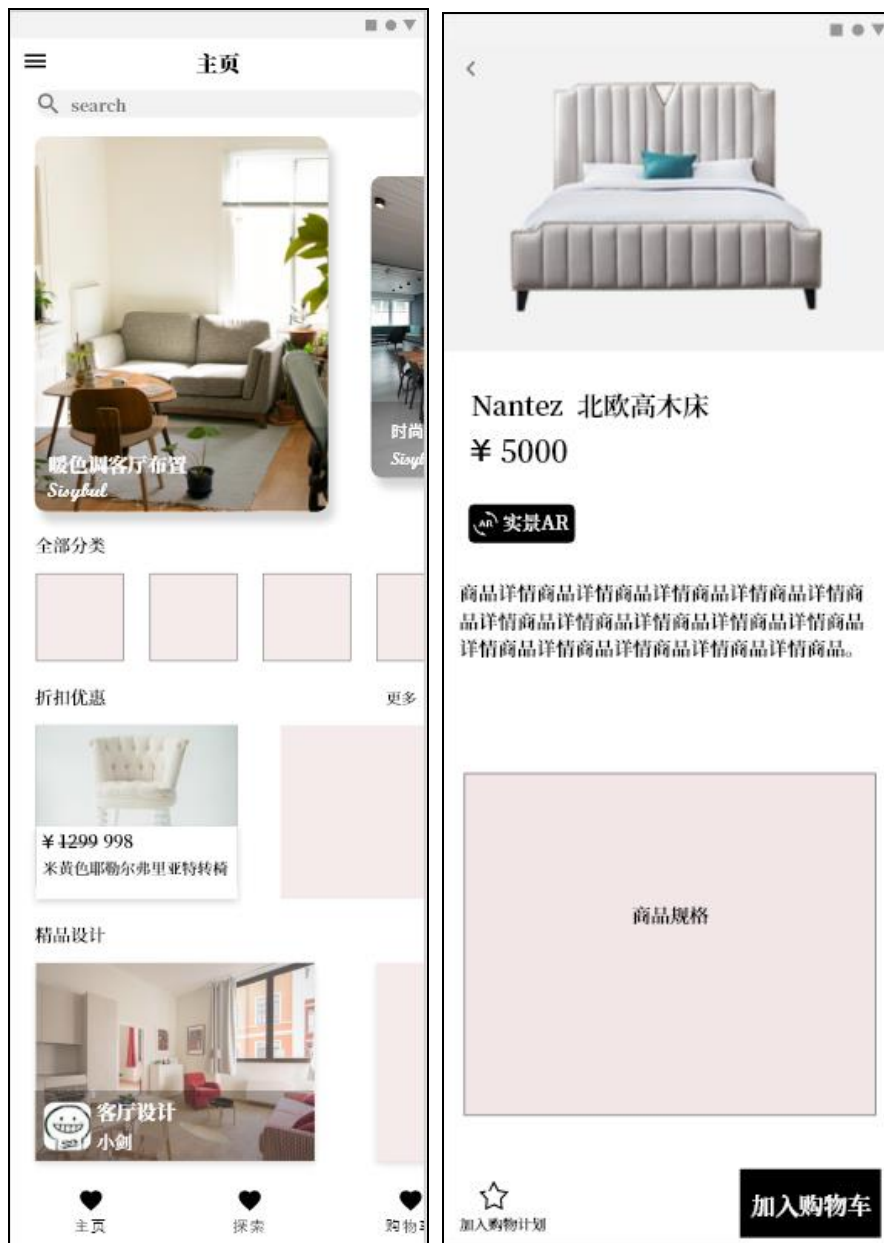
图 4.5 用户管理模块侧边栏设计图

3. 商品浏览功能设计

如图 4.6 所示，商品浏览分为三种。第一，从主页浏览，主页的轮播图和折扣推荐会推荐一些商品，并支持链入商品详情页。第二，分类浏览，从主页点击分类菜单可进

入商品浏览页。分类包括客厅设施、厨房用品、卧室家具、浴室用品、学习办公和其他。

第三，搜索浏览，输入关键字并选择分类信息可搜索到匹配的商品。



(a) 商品主页

(b) 商品详情

图 4.6 商品浏览功能设计图

4. 商品购买功能设计

本文实践中购买功能通过购物车实现，用户可以删除购物车里的内容，并点击下单完成购买流程。本次实践中，购物车的内容有两个来源一是在商品浏览中选择商品加入购物车，二是将购物计划中的一个购物方案整体加入购物车，购物车界面如图 4.7 所示。

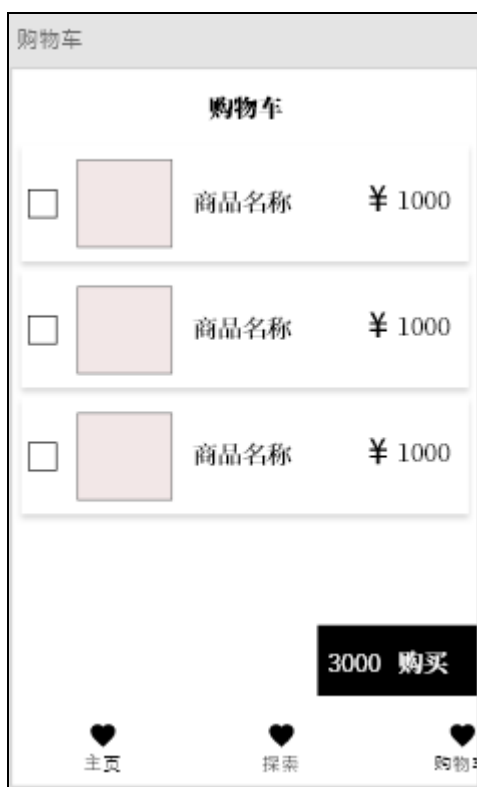


图 4.7 商品购买功能设计图

(四) 购物计划功能详述

1. 购物计划与体验式营销

购物计划是本文实践的特色功能，目的是提供更具沉浸感的购物体验。购物计划在本次实践中有两个功能，探索和购物计划管理。购物计划管理是存放用户自己创作和保存的其他用户创作的房间布置方案的地方，用户可以选择最喜欢的一款布置方案加入购物车。探索是用户浏览其他用户分享的布置方案的地方，探索允许用户阅读其他用户分享的购物方案并利用 AR 功能虚拟到实际环境中体验。

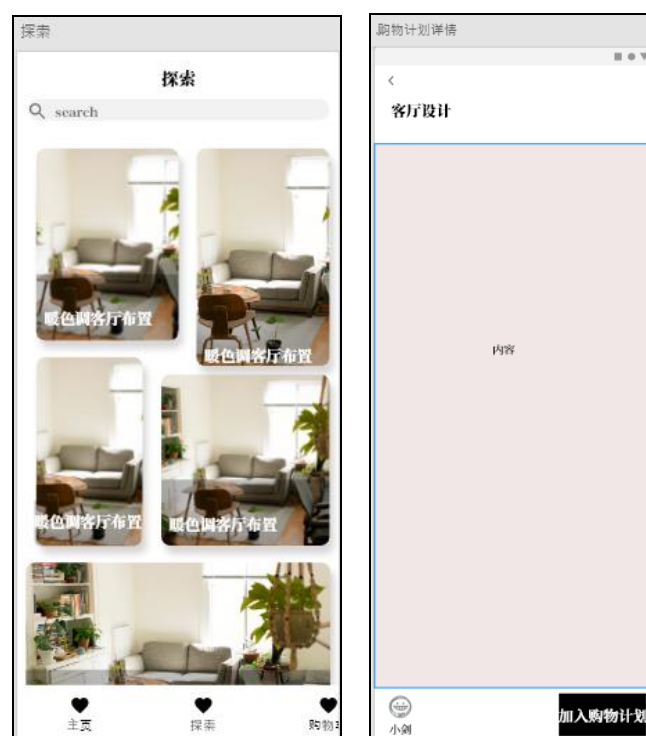
购物计划的灵感来自于宜家家具，是一种体验式营销的手段。体验式营销通过消费的体验行为，将消费者的个性化需求明晰并满足这种需求。宜家将每一个人心目中“家”的样子实地的展示在消费者面前，用想象和感受刺激消费，每一个宜家店都是一个消费者亲身体验的现场和展示的空间。购物计划就是宜家方案在线上实现的一种形式。购物计划将精心设计的“家”的布置方案展现在每一个消费者面前，消费者不仅可以通过文字和图片想象“家”的样子，还可以用 AR 在房间中布置、体验，这是购物计划的价值所在。

购物计划不仅仅是体验式营销手段，还实际的解决了消费者购物的痛点。购买家具

与购买其他商品不同，在家具购买活动中不仅仅是消费者与家具二者之间的互动，还是家具与家具、家具与实际环境之间的互动，因为家具需要与环境匹配且家具一般成套购买。购物计划的多模型 AR 展示功能和房间布置方案分享功能解决了消费者购买到家具却与房间不匹配和想要布置房间却不知道购买什么家具的窘境。

2. 购物计划功能设计

如图 4.8 所示，探索页与主页平级，用于浏览购物计划。探索页的展示采用瀑布流，点击可进入购物计划详情。购物计划详情展示家具布置方案详情，中间展示内容，底部按钮可将该布置方案添加入自己的购物计划。



(a) 探索 (b) 购物计划详情

图 4.8 购物计划功能设计图

(五) 数据库设计

1. 商品管理模块数据库设计

商品管理包括商品表、商品分类表、品牌分类表、场景分类表。所有分类表都包含 id 与名称 2 个属性。商品分类包括但不限于桌、椅、柜、床、灯，等。场景分类包括但不限于客厅、厨房、卧室、浴室、其他。商品表包括 id、名称、场景、品牌、分类，数量、价格、详情。其中详情是 json 字符串，存储了图片、文字、模型的信息与地址。

商品表与商品分类表、品牌分类表是一对一关系。商品表与、场景分类表是一对多

关系。

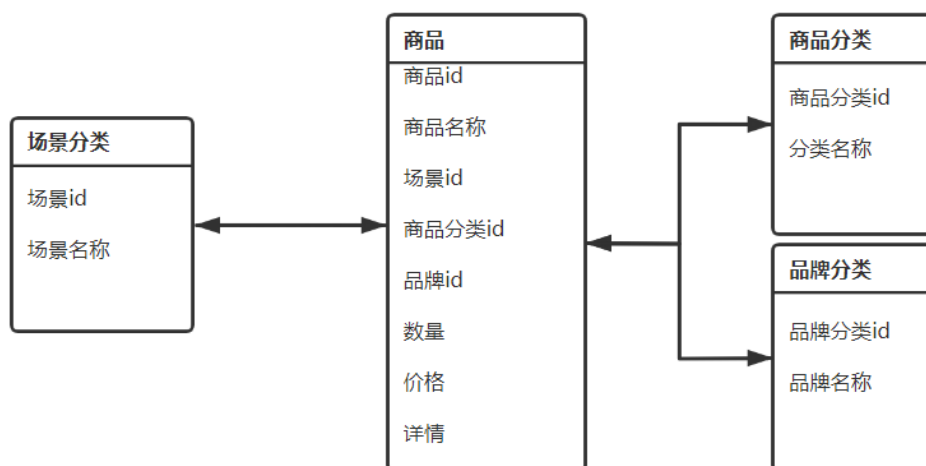


图 4.9 商品管理模块数据库设计图

2. 订单管理模块数据库设计

订单管理包括订单表与用户表。本文实践系统没有单独的用户管理，每个模块都需要用户表。订单包括订单 id、日期、价格、收获地址、用户 id、购物清单。其中购物清单是一个 json 字符串，包含所购商品的 id 与价格。订单表与用户表是一对一关系。

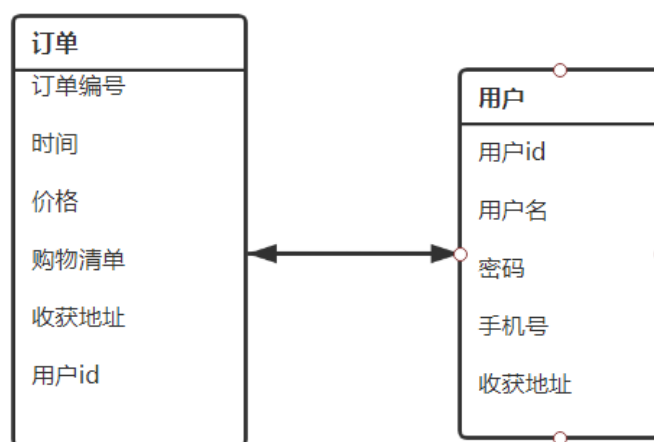


图 4.10 订单管理模块数据库设计图

3. 购物计划管理模块数据库设计

购物计划管理包含购物计划表，用户表。购物计划表保存所有创建的房间布置方案即购物计划，包括计划 id、用户 id、标题、内容、商品清单、私有状态。内容是文件地址，购物计划的内容使用 Markdown 语言，内容保存在 Markdown 文件。商品清单是 json 字符串，存有商品 id。私有状态代表此购物计划是否在探索中可见。购物计划表与用户表是多对一关系。

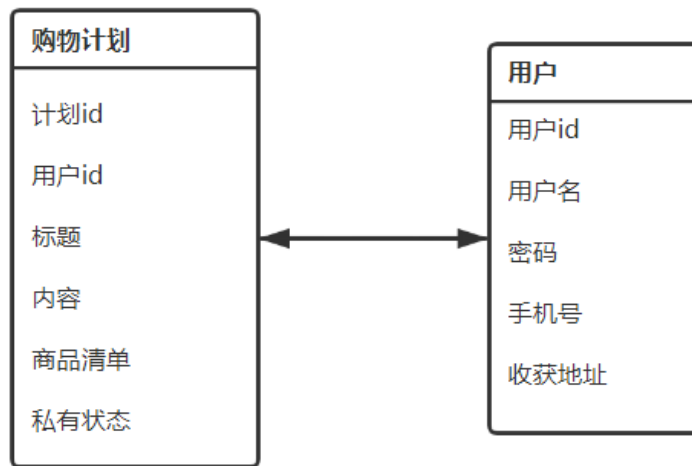


图 4.11 购物计划管理数据库设计图

五、系统实现

(一) 单模型 AR 展示功能实现

1. 使用 Sceneform

Sceneform 是 ARCore 体系的一部分。利用 Sceneform 可以轻松地在 AR 和非 AR 应用中渲染逼真的 3D 场景，而无需学习 OpenGL。

首先配置项目中的 build.gradle 文件，项目的 build.gradle 包括 Google 的 Maven 代码库。

```
allprojects {
    repositories {
        google()
    }
}
```

更新项目中的 build.gradle 以添加最新的 ARCore 和 Sceneform 用户体验依赖项，确保项目的设置与两个库都兼容。

```
android {
    defaultConfig {
        minSdkVersion 24
    }
    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
    }
}
```

```

        targetCompatibility JavaVersion.VERSION_1_8
    }
}
dependencies {
    implementation 'com.google.ar:core:1.5.0'
    implementation 'com.google.ar.sceneform.ux:sceneform-ux:1.5.0'
    implementation 'com.google.ar.sceneform:core:1.5.0'
}

```

修改项目中的 AndroidManifest.xml 文件更改 ARCore 和摄像头访问权限：

```

<uses-permission android:name="android.permission.CAMERA" />
<uses-feature android:name="android.hardware.camera.ar" />
<application>
    <meta-data android:name="com.google.ar.core" android:value="required" />
</application>

```

2. 导入 3D 模型

导入 3D 模型前首先要安装 Google Sceneform Tools (Beta)插件。如图 5.1 所示在 Android Studio 中，打开 Plugins 设置：

Windows: File > Settings > Plugins

macOS: Android Studio > Preferences > Plugins

搜索并安装 Google Sceneform Tools (Beta)。

Sceneform 支持 OBJ、FBX 和 glTF 格式的 3D 模型。导入 3D 模型前，在项目主目录建立 **Sampledata** 文件夹，这是程序自动生成的文件夹，如果没有需要手动建立。我们将 3D 模型的各种文件存放在 **Sampledata** 文件夹，通过网络请求的文件则存储在手机的内存中。注意 3D 模型文件不可放在 **assert** 或 **res** 文件内。

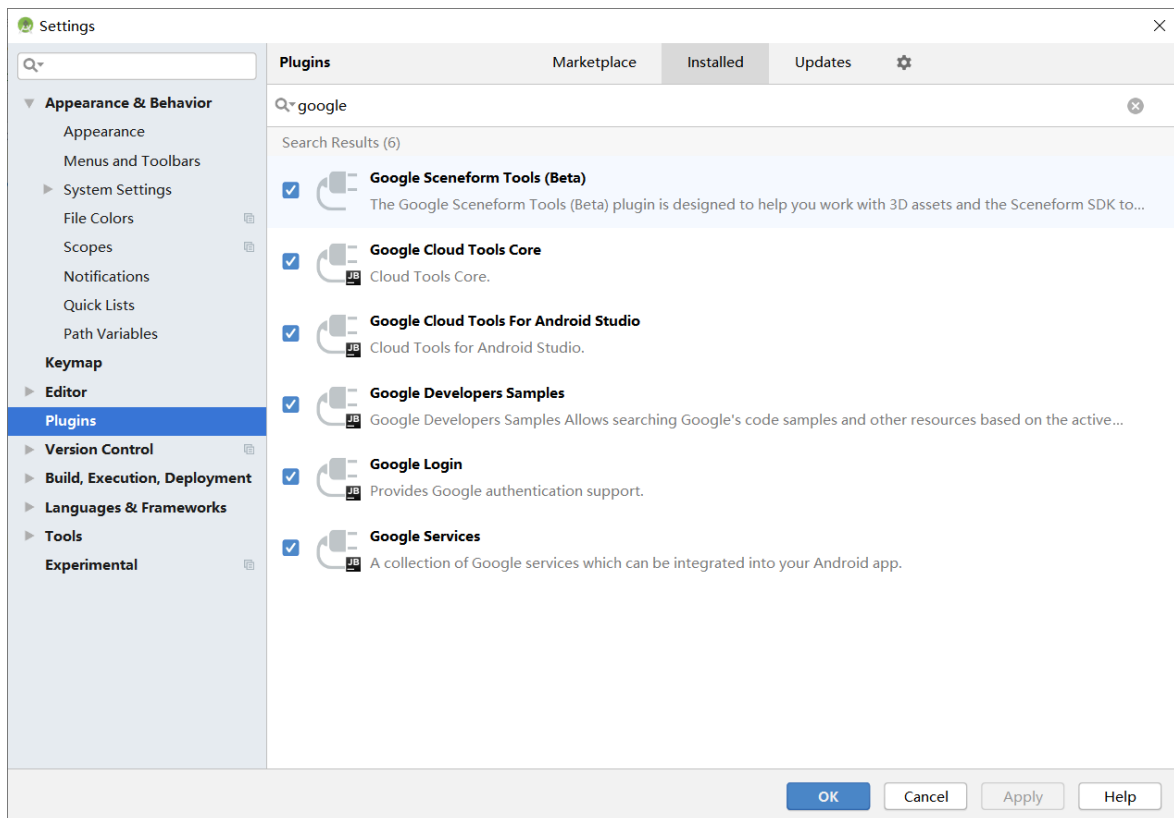


图 5.1 Sceneform 安装设置图

更新您的应用 build.gradle 文件，为新导入的资源添加一个 apply plugin 行和一个 Sceneform.asset() 条目：

```
apply plugin: 'com.google.ar.sceneform.plugin'

sceneform.asset('sampledata/models/andy.obj',
    'default',
    'sampledata/models/andy.sfa',
    'src/main/res/raw/andy')
```

3. 使用 AR

本节将介绍详细的编程方案。使用 ARCore 实现模型的放置很简单，本课题使用 ARCore 提供的 Java API 实现。

为实现 AR 效果，需要 Layout 和 Activity 类。Layout 是视图文件，使用 XML 语言构建视图界面。支持 AR 的视图需要使用 Sceneform 提供的 Fragment。AR 效果是通过 Fragment 实现的，代码如下图所示，不论是多模型还是单模型在 Layout 上没有区别。

```
<?xml version="1.0" encoding="utf-8"?>

<android.support.constraint.ConstraintLayout
```

```

...
tools:context=".ARActivity">

<fragment android:name="com.google.ar.sceneform.ux.ArFragment"

    android:id="@+id/arFragment"

    android:layout_width="match_parent"

    android:layout_height="match_parent" />

</android.support.constraint.ConstraintLayout>

```

Layout 负责绘制视图, Activity 则负责实现逻辑, Activity 与 Layout 一一对应。Activity 是一个 Java 类继承 AppCompatActivity。实现 AR 的逻辑首先 import 相关的类。

```

import com.google.ar.core.Anchor;

import com.google.ar.sceneform.AnchorNode;

import com.google.ar.sceneform.rendering.ModelRenderable;

import com.google.ar.sceneform.rendering.ViewRenderable;

import com.google.ar.sceneform.ux.ArFragment;

import com.google.ar.sceneform.ux.TransformableNode;

```

以 Anchor 开头的类提供锚点功能。锚点通过环境理解, 绘制周围环境的地图, 从而固定模型放置的位置, 锚点类同时还提供平面检测功能, 当开始运行 AR 时, 平面检测会检测到可以放置模型的平面。Rendering 类提供模型渲染功能, ArFragment 则是 AR 实现的基础。

首先创建 ArFrament 的对象, 与刚刚创建的 Layout 绑定。

```

arFragment = (ArFragment)getSupportFragmentManager()

    .findFragmentById(R.id.arFragment);

```

注意在开发中代码会显示为红色, 代码中红色部分在正常情况下是有错误的, 但本文使用的是 Flutter 与 Java 混合开发的方式, 虽然显示红色但并不影响运行。只有绑定的 Fragment 才能对屏幕镜像操作。创建屏幕的监听, 寻找平面, 监听用户点击操作创建锚点。

```

arFragment.setOnTapArPlaneListener(((hitResult, plane, motionEvent) ->{

    监听到点击操作后, 创建锚点并在锚点上渲染模型, 首先创建渲染模型的方法。

private void addModelToScene(Anchor anchor, ModelRenderable modelRenderable){

    AnchorNode anchorNode = new AnchorNode(anchor);

```



```

TransformableNode transformableNode = new TransformableNode(
    arFragment.getTransformationSystem());
transformableNode.setParent(anchorNode);
transformableNode.setRenderable(modelRenderable);
arFragment.getArSceneView().getScene().addChild(anchorNode);
transformableNode.select();
}

```

这个方法中通过传递进来的锚点和模型，将模型渲染到锚点上，但当这个函数执行完毕后模型并不会出现在屏幕中，调用 `show` 方法时才会真正的渲染。继续未完成的代码。

```

Anchor anchor = hitResult.createAnchor();
ModelRenderable.builder()
    .setSource(this, Uri.parse(MainActivity.model+".sfb"))
    .build()
    .thenAccept(modelRenderable ->
        addModelToScene(anchor,modelRenderable))
    .exceptionally(throwable -> {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setMessage(throwable.getMessage())
            .show();
        return null;
    });
});

```

这段代码在监听函数内部。通过这些代码就可以实现将一个模型渲染到镜头中了。下图是实际渲染效果。



图 5.2 单模型 AR 实机渲染图

（二）多模型 AR 展示功能实现

多模型 AR 展示与单模型的操作类似，区别在于 Activity 文件的程序不同，所以这里就不再介绍 Sceneform 和导入 3D 模型了。

多模型展示的原理是在单模型展示的基础上，在渲染过程中改变模型。需要展示的模型以 ListView 的形式展示，通过点击改变即将渲染的模型。首先注册 ListView，并设置监听。

```
Private List<modelbin> list = new ArrayList<modelbin>();  
adapter models = new adapter(planActivity.this,R.Layout.mAR,list);  
ListView listView = (ListView)findViewById (R.id.plangoods_listview);  
listView.setAdapter(models)
```

adapter 是数据适配器，负责向 ListView 装配数据，而 ListView 负责展示与传递数据。下面设置监听。

```
listView.setOnItemClickListener(new AdapterView.OnItemClickListener(){  
    @Override  
    Public void onItemClick(AdapterView<?>,View view, int position , long id){  
        modelbin item = list.get(position);  
        modelname = item.name;  
    }  
}
```

})

每次点击都会更改 modelname，modelname 存储模型的存储路径。下面设置渲染函数，与单模型 AR 一样，只需要将模型路径改为 modelname。



图 5.3 多模型 AR 实机渲染图

（三）购物计划与富文本解决方案

1. 购物计划探索功能实现

探索功能是与主页平级的功能页，探索功能通过瀑布流的方式展示购物计划。为实现瀑布流，本课题采用 flutter_staggered_grid_view 插件。首先更改 pubspec.yaml，安装插件。

dependencies:

flutter_markdown: ^0.2.0

然后创建存储购物计划数据的实体类。

```
class PlanEntity {  
    int design_id;  
    String title;  
    String contentUrl;  
    String imageUrl;  
    String name;  
    String userImg;  
    PlanEntity(  
        this.title,  
        this.name,
```

```

        this.imageUrl,
        {this.design_id=0,this.userImg="asset/creator.jpg",
        this.contentUrl="asset/demo.md"}
    );
}

```

最后实现瀑布流。

```

Widget build(BuildContext context) {
  return Expanded(child: RefreshIndicator(
    onRefresh: _onRefresh,
    child: StaggeredGridView.countBuilder(
      crossAxisCount: 4,
      padding: EdgeInsets.all(8),
      itemCount: _PlanList.length,
      itemBuilder: (BuildContext context, int index) {
        PlanEntity item = _PlanList[index];
        return PlanItem(item);},
      staggeredTileBuilder: (int index) =>
        new StaggeredTile.count(2, index == 0 ? 2.5 : 3),
      mainAxisSpacing: 8.0,
      crossAxisSpacing: 8.0,)),
  );
}

```

瀑布流的实现使用插件提供的模板示例。它的核心在于 `itemCount` 和 `itemBuilder`。`itemCount` 的 `_PlanList` 是 `PlanEntity` 的列表存储的需要展示的数据。`itemBuilder` 对应的函数设置展示数据的 UI，核心是 `PlanItem`，`PlanItem` 的代码与普通的界面编写一样，在这里不进行展示。



图 5.4 探索功能实机演示图

2. 富文本解决方案

房屋布置方案是一种富文本形式。为更好的展示房屋的布置，内容要包括文字和图片，并且文字与图片的位置大小允许自定义。为实现一个对编辑友好，易于展示且低成本的富文本方案，本课题采用 Markdown 语法。Markdown 是一种轻量级标记语言，它允许人们使用易读易写的纯文本格式编写文档。本课题使用 flutter_markdown 插件展示 Markdown 文件。

安装 flutter_markdown。更改 pubspec.yaml。

dependencies:

flutter_markdown: ^0.2.0

展示 Markdown 的思路是现在网上下载布置方案的 Markdown 文件，再用 flutter_markdown 的方法 Markdown()展示。

```
body: Container(
  child: FutureBuilder(
    future: rootBundle.loadString(res),
    builder: (BuildContext context, AsyncSnapshot snapshot){
      if(snapshot.hasData){
        return Markdown(data: snapshot.data);
      }
    }
  )
)
```

```

}else{
    return Center(
        child: Text("加载中..."));
}

```

下图是实机演示。



图 5.5 购物计划详情实机展示图

（五）Flutter 与 Java 的混合开发

ARCore 的功能是基于原生 Android 系统开发的，和 Flutter 不在同一体系中。为实现在 Flutter 中调用 ARCore 的功能，必须在 Flutter 的框架内调用原生 Android，这就是混合开发。要实现混合开发，首先要把原生安卓的项目文件放到 Flutter 项目主目录下。这个项目文件要提前调试好，是可以单独运行的 APP。

在项目内调用原生安卓使用广播的方式，广播是安卓一种通信方式，向所有应用广播信息。使用广播前要注册。注册在原生安卓项目中进行。作为嵌入 Flutter 中的安卓项目，MainActivity 是没有界面的，并且继承 FlutterActivity。利用 MainActivity 进行注册。

```

public class MainActivity extends FlutterActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}

```

```
GeneratedPluginRegistrant.registerWith(this);
```

注册方法在另一个类实现。

```
public static void registerWith(PluginRegistry registry) {  
    if (alreadyRegisteredWith(registry)) {  
        return;  
    }  
  
    private static boolean alreadyRegisteredWith(PluginRegistry registry) {  
        final String key = GeneratedPluginRegistrant.class.getCanonicalName();  
        if (registry.hasPlugin(key)) {  
            return true;  
        }  
        registry.registrarFor(key);  
        return false;  
    }  
}
```

注册成功后，原生端会开启广播，等待接受广播，当接受到广播后启动拥有 AR 功能的界面，接受广播的方法在 MainActivity 中。

```
new MethodChannel(getFlutterView(), "flutter.ar").setMethodCallHandler(  
    new MethodChannel.MethodCallHandler() {  
        @Override  
        public void onMethodCall(MethodCall call, MethodChannel.Result result){  
            model=call.method ;  
            startar();  
            result.success(success);  
        }  
    }  
);  
  
private void startar() {  
    startActivity(new Intent(this,ARActivity.class));  
}
```

最后一步是在 Flutter 端发送广播。Flutter 实现广播相对简单，拥有成熟的 API。

```
void callAr() async {  
    String result = await methodChannel.invokeMethod(goods.model);  
    print(result);  
}
```

六、总结与展望

（一）总结

本文主要分析了国内外在电子商务与 AR 融合领域的研究现状，研读了一些国内学者在该领域所做的实践。AR 助力电子商务一直深受国内外学者的关注，利用 AR 可以有效解决基于 2D 线上购物的体验感问题，并可提供更多的销售手段。然而 AR 对设备的要求较高，市场上流通的移动设备无法满足 AR 的流畅运行，这阻碍了 AR 与电子商务融合的普及与实践。近年来 Google 推出的 ARCore 技术解决了这一问题，ARCore 使得 AR 可以在移动设备下流畅运行，并提供系统且全面的开发工具包。

本文以家具售卖为例，通过 ARCore 技术研究并实现了一款 AR 与电子商务融合的 3D 购物 APP。全面学习 ARCore 的开发方法，实现在移动端对商品进行 AR 展示的功能，解决 Flutter 与 Java 的混合开发问题，并设计了利用 AR 技术将体验式营销策略与线上销售结合的新功能——购物计划。购物计划支持用户浏览其他用户分享的房间布置方案并支持使用 AR 体验布置方案，购物计划解决了用户的购物体验问题和不知道购买什么家具的购物窘境。本文所做实践亮点在于 AR 商品展示与购物计划，重点在前端的功能设计和实现，对后端的涉及不多。本文对实践的系统进行了需求分析，设计了 AR 家具售卖系统的功能、前端界面和数据库，并完成在服务器的部署。

（二）不足与展望

线上购物已经成为时代主流，AR 与电子商务的融合必将带来巨大的价值。AR 突破了虚拟与现实的界限，为消费者提供更多更全面的信息和更好的购物体验，兼具线下购物与线上购物双方的优点，拥有巨大的优势，可以预见 AR 与电子商务的融合必将成为未来主要研究方向之一。

AR 与电子商务的融合依然面临着一些不足与困难。第一，建模成本巨大，为实现逼真的效果需要精细的模型，而建模依然依靠人工，制约了 AR 与电子商务融合的发展；第二，复杂的现实环境，当前的 AR 技术需要通过检测平面固定模型的渲染位置，而仅在家具售卖领域，房屋中需要展示的装饰与物品摆放的环境不仅仅只有平面。

参 考 文 献

- [1] 陈若雯, 电子商务系统中基于 AR 的家居展示平台的设计与实现[D]. 重庆大学, 2017.
- [2] 矫春龙, 基于 Kinect 的 3D 虚拟试衣系统[D]. 西安电子科技大学, 2014.
- [3] 蒋先梅, 基于虚拟现实技术的电子商务展示平台的研究与实现[D]. 江西科技师范学院, 2010.
- [4] 乔秀全, 任沛, 商彦磊. 关于增强现实技术潜在发展方向的思考[J]. 中兴通讯技术, 2017, 23(06): 37-40.
- [5] 钱叶明, 我国 B2C 企业基于体验经济的商品展示研究[D]. 苏州大学, 2014.
- [6] 夏杰, 尤志翔, 李凯, 安平. 基于 AR 技术的实时互动体验电商系统[J]. 电视技术, 2018, 42(02): 76-82.
- [7] 朱淼良, 姚远, 蒋云良. 增强现实综述[J]. 中国图象图形学报, 2004(07): 3-10.
- [8] 张珊, 基于移动增强现实的厨电零售应用用户体验探究与设计[D]. 浙江大学, 2017.
- [9] 赵小军, 增强现实网购中的技术接受心理模型[D]. 陕西师范大学, 2015.
- [10] 周宣彤, 基于增强现实的场景化数字零售 App 设计研究[D]. 哈尔滨工业大学, 2019.
- [11] Daugherty, Terry & Biocca, Frank. (2005). Experiential Ecommerce: A Summary of Research Investigating the Impact of Virtual Experience on Consumer Learning. Online Consumer Psychology: Understanding and Influencing Consumer Behavior in the Virtual World. 10. 4324/ 9781410612694.