

天津大学

《软件工程综合实践》结课报告

饿了吧项目实践报告



学	院	<u>软件工程学院</u>
专	业	<u>软件工程</u>
年	级	<u>2022级</u>
姓	名	<u>杨宇鑫 谢帛洋</u>
学	号	<u>3022207128 3022206049</u>

2024 年 9 月 22 日

目 录

第一章	软件需求规格说明书	1
1.1	引言	1
1.2	任务概述	1
1.3	功能需求	2
1.4	数据需求	6
1.5	数据流条目	7
1.6	数据存储条目	11
1.7	性能需求	12
1.8	运行需求	13
第二章	项目设计文档	15
2.1	数据库设计	15
2.2	前端设计	18
2.3	后端设计	21
第三章	项目测试文档	28
3.1	项目背景	28
3.2	编写目的	28
3.3	测试人员	28
3.4	测试目标	28
3.5	测试要求	28
3.6	测试方法	28
3.7	测试内容	28
3.8	测试结果	31

3.9	测试用例	31
第四章	部署文档	32
4.1	项目背景	32
4.2	系统需求	33
4.3	项目部署的系统需求	33
4.4	配置说明	34
4.5	部署的步骤和流程	35
第五章	人员分工及项目进度	37
第六章	项目分析及开发过程	38
6.1	项目分析	38
6.2	开发过程	42
6.3	gitee上团队成员贡献	42
第七章	开发过程	45
7.1	Bug修复以及简单功能的完善	45
7.2	饿了么V2.0版本的创新点	46
7.3	部署成果	53
第八章	问题，解决对策及反思	55
8.1	浮点数精度	55
8.2	页面显示不全	55
8.3	历史订单的价格不能随着新的商品价格变化而变化	55
8.4	商品详情页的图片不能显示	55
8.5	搜索	55
8.6	手机号和密码的强度检验	56

8.7	点赞，收藏，评论的存储	56
8.8	文心一言的接口	56
8.9	部署到云端	56
第九章	成果与展望	57

第一章 软件需求规格说明书

1.1 引言

1.1.1 编写目的

本文档的目的是详细地介绍饿了么APP所包含的需求，以便客户能够知晓产品的确切需求以及开发人员能够根据需求进一步进行扩展开发，以下叙述将结合文字描述、数据流图、ER图等来描述饿了么APP的功能、性能、用户界面、运行环境、外部接口以及针对用户操作给出的各种响应。本文档的预期读者有客户、开发人员等。

1.1.2 背景

该项目适用于喜欢吃餐厅菜肴但不喜欢出门或不善于做饭的各年龄段人群，由谢帛洋和杨宇鑫二人团队进行开发和部署。

1.1.3 参考资料

[1] 奚万峰. 软件工程方法与实践(第三版). 北京: 机械工业出版社, 2016

[2] 普莱斯曼. 软件工程: 实践者的研究方法(原书第8版). 北京: 机械工业出版社, 2016

1.2 任务概述

1.2.1 项目概述

1.2.1.1 项目简介

饿了么APP是一款针对各年龄段各类人群的外卖软件，大家可以在这里浏览各个餐厅的美食，以及足不出户享受菜肴。软件最大的特色是添加了AI智能咨询服务，输入需求就可以得到饮食建议，方便使用者进行选择。

1.2.1.2 项目目标

该项目的市场目标为以年轻人群体为主、外卖软件市场，应用目标为实现外卖以及饮食咨询工作。

1.2.1.3 具体功能概述

(1) 搜索功能：通过餐厅名、餐品名或地址模糊搜索显示结果列表。

(2) 点餐功能：进入餐厅后可以选择餐品加入购物车，点击购物车可以看到餐品明细，继而可以进行餐品支付。

(3) 咨询功能：可以询问智能AI得到推荐食谱。

(4)订单功能：可以显示未支付和已支付的所有订单，并能查看订单明细。

(5)我的信息：查看个人基本资料，可以进行个人资料的修改（包括电话、头像等）。

(6)商家功能：可以进行商家注册以及登录，登陆后可以上架商品，编辑商品信息等。

(7)点赞、收藏和评论功能：在商家列表页可以进行点赞、收藏以及评论功能。

1.2.2 用户特点

本产品的用户主要是喜爱点外卖年轻人群体，接受新鲜事物的能力比较强，他们比较容易接受外卖菜肴。并且对于智能AI的认可度也比较高。尤其是当代有许多人比较‘宅’，这类人群是我们的重点目标客户。

1.2.3 假定和约束

(1)人力的时间的约束：本APP开发过程中需要考虑到人力的时间的约束,相较于一些软件的开发团队来说人员较少时间较短。

(2)技术发展的约束：计算机技术和发展的日新月异，将会给信息处理带来更多手段，同时也会带来更加丰富的信息表达形式，例如心形的人工智能等等，可能导致我们的搜索时候没有那么智能，这就要求软件在设计时要考虑技术变化的可能性，为可能的变化预留一定的处理能力。

1.3 功能需求

1.3.1 功能划分

1) 饿了么的顶层数据流图



图 1饿了么顶层数据流图

描述：如图1所示，用户可以扮演两种角色——个人用户、商家用户。当用户选择个人用户时，可以向饿了么发送事务（如修改资料等），同时个人用户可以浏览饿了么返回的事务，即个人用户与饿了么有双向的数据流动。当用户选择商家用户时，可以向饿了么系统发送事务上架商品，同时可以浏览饿了么返回的事务，即商家用户与饿了么系统也有双向的数据流动。

(2) 饿了么的0层数据流图

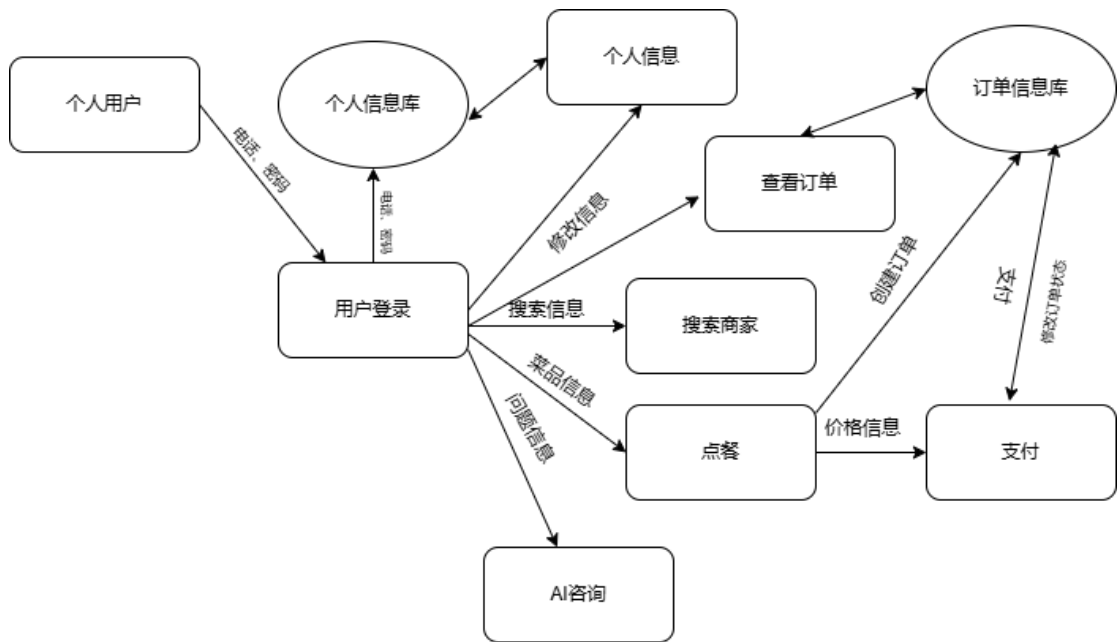


图 2 饿了么 0层数据流图

描述：如图2所示，个人用户通过提交身份信息向用户登录事务发送请求。用户登录事务从用户信息库中读取相应的用户信息进行匹配判断登录结果。用户登录成功后，用户可以进行个人信息管理、搜索商家、点餐、AI咨询、查看订单等操作。用户进行搜索操作时，用户提供的搜索信息流动到搜索事务，搜索事务对搜索信息进行相应的处理后得到商家信息。用户进行个人信息管理时，用户的修改信息传入信息修改事务中，修改个人信息库并返回修改后的新信息。进行点餐操作时发送菜品信息并进入支付事务，支付事务修改数据库创建新订单。其他操作同上述基本一致。

1.3.2 功能描述

(1)用户登录

功能描述：如图3所示，用户登录可以分为注册和登陆。注册时用户提供新用户注册信息发往注册事务，注册事务根据新用户注册信息得到新用户信息存入用户信息库，同时流动出用户信息。登录时用户提供用户名和密码发往登录事务，登录事务将得到的用户名和密码与用户信息库中的信息匹配，同时流动出用户信息。

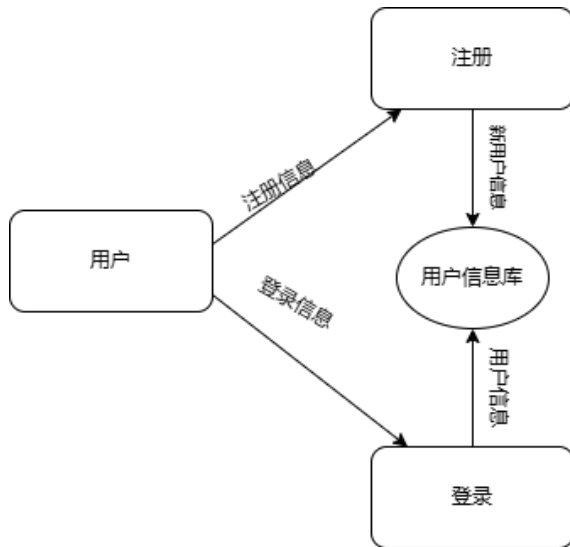


图 3 用户登录功能细化数据流图

(2)个人信息管理

功能描述：用户登录后可以进行相应操作进入个人信息管理界面，用户可以在此页面修改自己的个人信息。

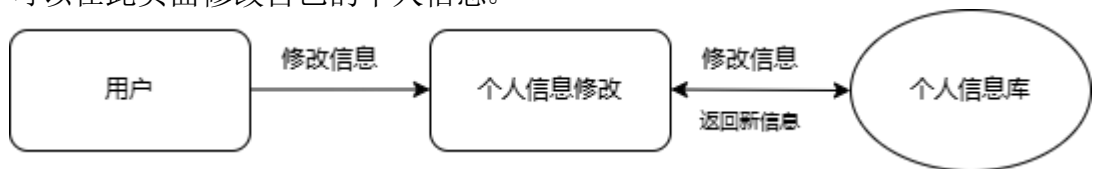


图 4 个人信息管理功能细化数据流图

(3)搜索商家

功能描述：如图4所示，用户在搜索栏输入相应的搜索信息。搜索信息可以是商家名称或菜品名称。用户进入搜索商家事务，搜索商家事务处理传入的搜索信息流动出商家内容。

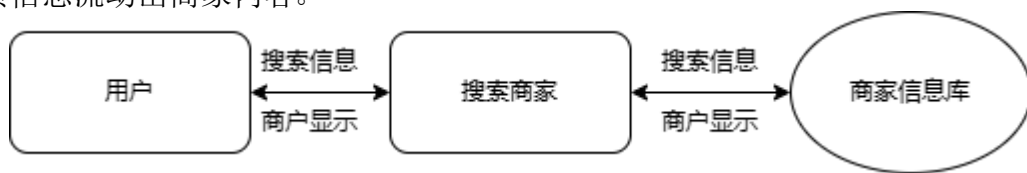


图 5 搜索功能细化数据流图

(4)点餐功能

功能描述：用户选择菜品后提交菜品信息触发点餐事务，创建订单修改订单信息库，接着触发支付事务改变订单状态值。

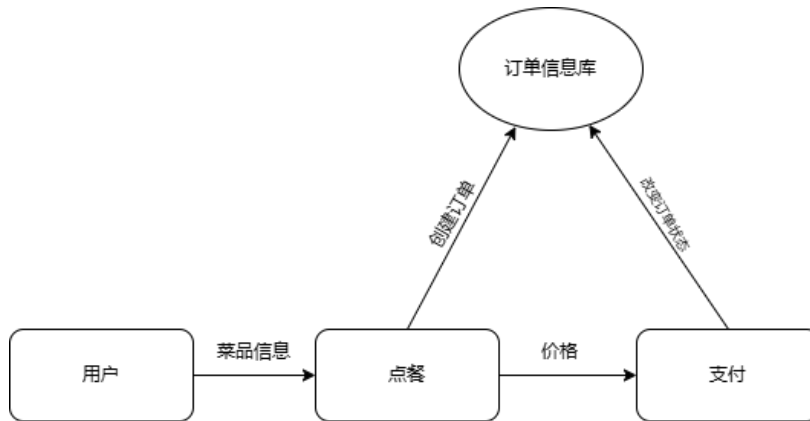


图 6 点餐功能细化数据流图

(5)咨询功能

功能描述：用户可以向智能AI传递问题信息，触发咨询事务，返回AI的回答。

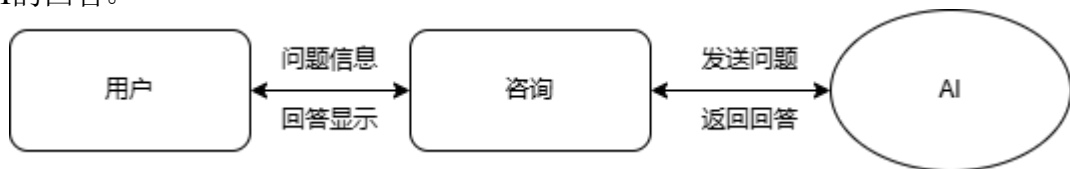


图 7 咨询功能细化数据流图

(6)订单功能

功能描述：用户点击订单后触发订单查看事务，事务从订单数据库查找订单数据并分为已支付和未支付，进行显示。

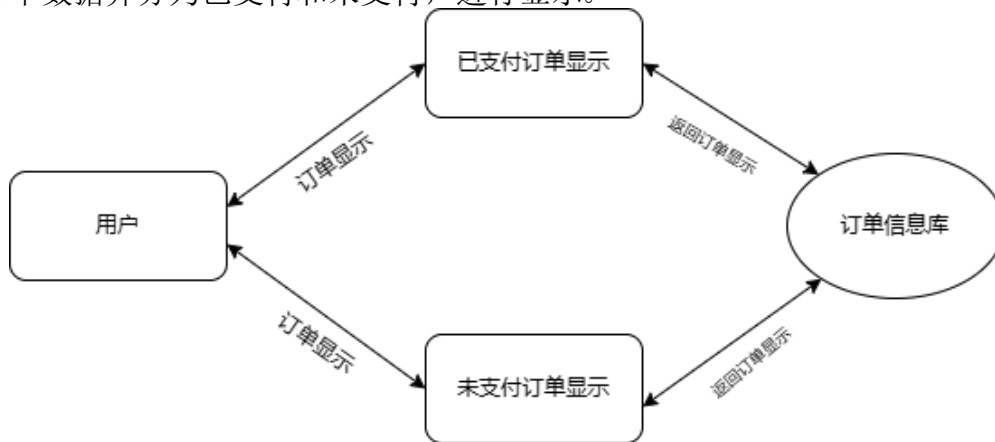


图 8 订单功能细化数据流图

(7)点赞、评论和收藏功能

功能描述：在商家列表处可以进行点赞、收藏、评论操作。点赞时用户触发点赞事务，向点赞信息库中加入信息，收藏同上。评论时用户触发评论事务，向评论信息库中加入信息，并且显示评论信息库中的每一条评论，即评论信息库向用户返回评论信息。

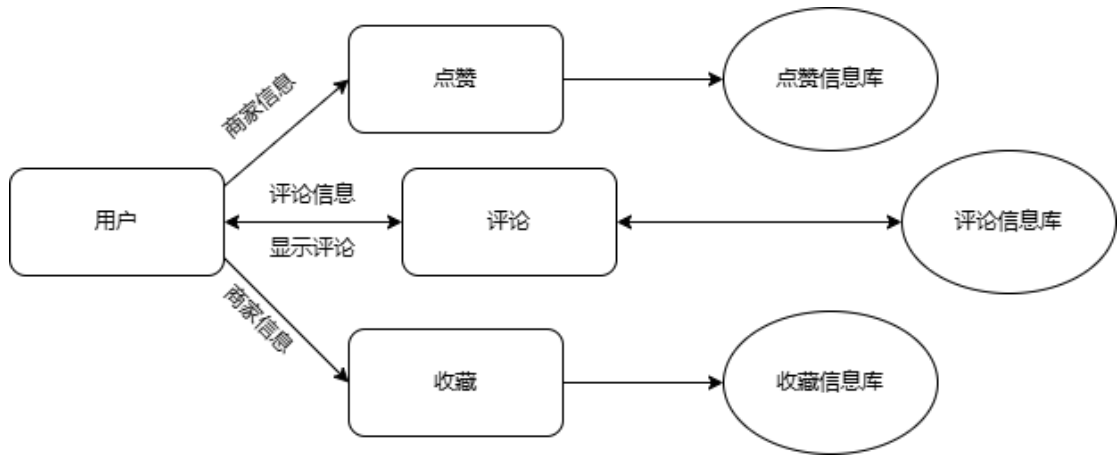


图 9 点赞、评论和收藏功能细化数据流图

(8)商家功能

功能描述：用户提交电话和密码触发商户登录事务，如果未进行注册则先进行注册事务，提交电话和密码至用户信息库。若进行过注册则进行比对，一致则登录成功。登陆事务出发后用户可以进行商户信息修改和上架商品等功能。

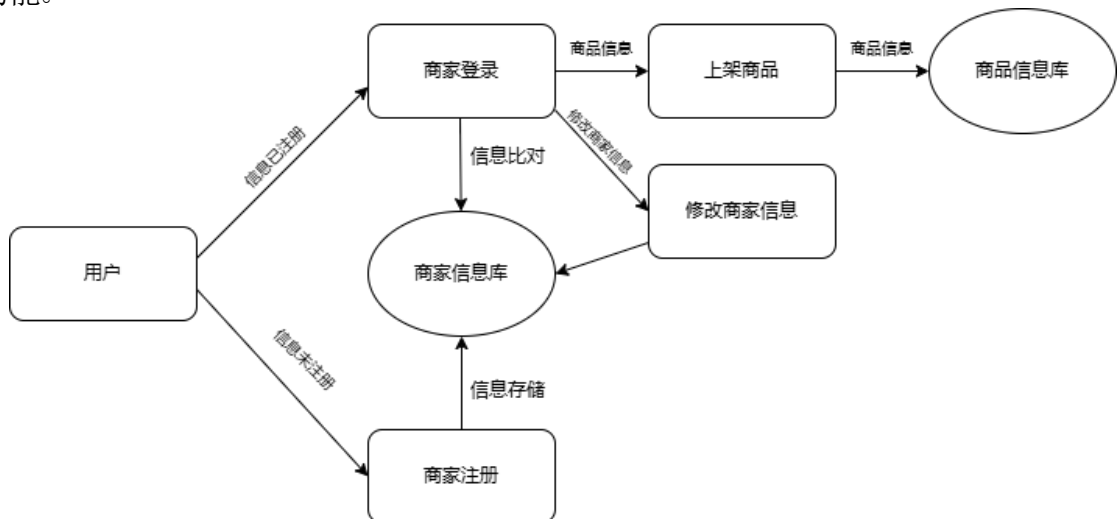


图 10 商家功能细化数据流图

1.4 数据需求

1.4.1 静态数据

用户信息、商户信息、点赞信息、收藏信息、评论信息、商品信息

1.4.2 动态数据

用户输入的搜索信息，输入框内的信息

1.4.3 数据字典

1.5 数据流条目

(1)身份信息

名称	身份信息
简述	个人用户和商家用户的身份
来源	个人用户和商家用户
去处	用户登录

(2)用户名

名称	用户名
简述	用户登录的账号
类型	varchar
长度	1024
来源	用户登录
去处	用户信息库

(3)密码

名称	密码
简述	用户登录的账号对应的密码
类型	varchar
长度	1024
来源	用户登录
去处	用户信息库

(4)搜索商家

名称	搜索信息
简述	用户发出搜索商户的信息
来源	用户输入
去处	搜索问题

(5)注册信息

名称	新用户注册信息
----	---------

简述 新用户进行注册的信息
来源 用户输入
去处 注册（用户信息库）

(6)点赞信息

名称 点赞信息
简述 用户点赞时的信息
来源 用户
去处 点赞信息库

(7)收藏信息

名称 收藏信息
简述 用户收藏时的信息
来源 用户
去处 收藏信息库

(8)评论信息

名称 评论信息
简述 用户评论时的信息
来源 用户输入
去处 评论信息库

(9)商家列表

名称 商家列表
简述 商家的信息列表
来源 商家
去处 查看商家

(10)商家信息

名称 商家信息
简述 用户点单时展示的商家信息
来源 商家

去处 用户点单

(11)支付信息

名称 支付信息
简述 进行支付时的信息
来源 用户点单
去处 用户支付

(12)订单信息

名称 订单信息
简述 用户点单后产生的订单信息
来源 用户点单
去处 订单信息库

(13)AI咨询信息

名称 AI咨询信息
简述 用户输入的咨询问题
来源 用户输入
去处 AI咨询

(14)个人信息

名称 个人信息
简述 用户的个人信息展示
来源 用户登录、用户信息库
去处 我的信息功能

(15)修改的个人信息

名称 修改的个人信息
简述 用户输入的新用户信息
来源 我的信息中用户输入
去处 我的信息功能

(16)商户用户名

名称	商户用户名
简述	商户的电话号码
来源	商户注册时的用户名
去处	商户登录

(17)商户注册

名称	商户注册信息
简述	商户注册时信息
来源	商户输入
去处	商户登录

(18)商户密码

名称	商户密码
简述	商户登录的账号对应的密码
类型	varchar
长度	1024
来源	商户登录
去处	商家信息库

(19)商品信息

名称	商品信息
简述	商户商品的详细信息
来源	商户
去处	商品信息库

(20)上架商品信息

名称	上架商品信息
简述	商户想要上架的商品信息
来源	上架商品
去处	商品信息库

(21)地址信息

名称	地址信息
简述	外卖配送地址
来源	地址添加
去处	点餐功能

1.6 数据存储条目

(1) 用户信息

名称	用户信息
简述	描述用户的信息
组成	用户名+密码+昵称+头像+Id
组织方式	以Id为关键字

(2) 地址信息

名称	地址信息
简述	描述配送地址的信息
组成	名称+电话+地址+编号
组织方式	以编号为关键字

(3) 商户信息

名称	用户信息
简述	描述商家的信息
组成	编号+图片+地址+类别+名称+商品+电话+密码
组织方式	以编号为关键字

(4) 商品信息

名称	商品信息
简述	描述商品的信息
组成	商品名称+编号+商户+图片
组织方式	以编号为关键字

(5) 点赞信息

名称	点赞信息
简述	描述商家获得点赞的信息
组成	用户编号+商家编号+点赞编号
组织方式	以点赞编号为关键字

(6)收藏信息

名称	收藏信息
简述	描述商家获得收藏的信息
组成	用户编号+商家编号+收藏编号
组织方式	以收藏编号为关键字

(7)评论信息

名称	评论信息
简述	描述商家获得评论的信息
组成	用户编号+商家编号+评论编号
组织方式	以评论编号为关键字

1.7 性能需求

1.7.1 数据精度

字段	精度	备注
用户名	char型	符合中国大陆电话格式
密码	char型	8位以上，大小写、符号和数字都要包含
昵称	char型	8位以下
用户是否存在	map型	前端传过来含有用户名和密码的json对象，后端接受到之后在数据库中匹配
用户ID	int型	自动赋值

1.7.2 时间特性

- (1) 响应时间：用户任意操作后2秒内系统给予反馈信息。
- (2) 更新处理时间：由系统运行状态来决定。
- (3) 数据的转换和传送时间：能够在5秒内完成。

5.3 灵活性

当需求发生某些变化时，该软件的基本操作、数据结构、运行环境等等基本不会发生变化，只是对系统的数据库的文件和记录进行处理，就可以满足需求。

1.8 运行需求

1.8.1 用户界面

(1) 注册：用户填写该页面的“用户名”、“昵称”、“密码”、“确认密码”、“头像”信息后点击提交即可成功注册，返回“注册是否成功的消息”。

(2) 登录：用户填写该页面的“用户名”、“密码”信息后点击登录即可成功登录，如果用户没有账号可以点击下方按钮进行注册。

(3) 主页：该页面展示各个不同种类的美食，底部显示主页、发现、订单、我的四个按钮，右上角显示登录注册按钮，可以进入不同的页面。

(4) 个人中心：点击修改资料可以修改资料；中间提供展示该用户的基本数据信息；点击最下方的退出可以登出用户。

(5) 点单流程：点击后展示每个类别的商家列表，再进一步点击可以展示商家的商品信息，并可以进行点单，进入支付页面并完成支付。

(6) AI咨询：点击首页发现键进入AI界面，可以输入想要咨询的问题并发送，返回的内容会被显示在下方。

(7) 我的订单：订单界面分别显示已支付订单和未支付订单，并且可以点击下拉按钮显示订单明细，对于未支付订单可以点击去支付完成订单的支付。

(8) 商户信息：进行商户登陆后会显示该商户的基本信息，如图片、名称以及电话和菜品等。

(9) 上架商品：商家登陆后可以上架新商品，输入新商品的图片、价格、名称和简介即可上架。

(10) 搜索商户：用户可以在首页的搜索栏输入搜索内容，点击搜索后下方展示搜索出的商家，并可以点击商家进行点餐。

1.8.2 软件接口

1.操作系统：Microsoft Windows系统

2.软件设备：VScode、IntelliJ IDEA、MySQL8.0、eclipse

1.8.3 质量属性

- 1.可用性：用户可以使用
- 2.可靠性：在给定时间内可以满足无错运行的要求
- 3.可维护性：服务器重启、写进日志
- 4.安全性：对用户的密码加密
- 5.可移植性：移动端移植

第二章 项目设计文档

2.1 数据库设计

2.1.1 business表

Field	Null	Key	Default	Extra
businessId	NO	PRI	NULL	auto_increment
phoneNumber	NO		NULL	
password	NO		NULL	
businessName	NO		未命名商家	
businessAddress	YES		NULL	
businessExplain	YES		NULL	
businessImg	YES		NULL	
orderTypeId	YES		NULL	
starPrice	YES		0	
deliveryPrice	YES		0	
remarks	YES		NULL	

2.1.2 cart表

Field	Type	Null	Key	Default	Extra
cartId	int	NO	PRI	NULL	auto_increment
foodId	int	NO		NULL	
businessId	int	NO		NULL	
userId	varchar(20)	NO		NULL	
quantity	int	NO		NULL	

2.1.3 chats表

Field	Type	Null	Key	Default	Extra
currentUserId	varchar(50)	NO		NULL	
senderUserId	varchar(50)	NO		NULL	
receiverUserId	varchar(50)	NO		NULL	

message	varchar(255)	YES	NULL
---------	--------------	-----	------

2.1.4 deliveryaddress表

Field	Type	Null	Key	Default	Extra
daId	int	NO	PRI	NULL	auto_increment
contactName	varchar(20)	NO		NULL	
contactSex	int	NO		NULL	
contactTel	varchar(20)	NO		NULL	
address	varchar(100)	NO		NULL	

2.1.5 favorite表

Field	Type	Null	Key	Default	Extra
userId	varchar(20)	NO		NULL	
businessId	int	NO		NULL	

2.1.6 food表

Field	Type	Null	Key	Default	Extra
foodId	int	NO	PRI	NULL	auto_increment
foodName	varchar(30)	NO		NULL	
foodExplain	varchar(30)	NO		NULL	
foodImg	mediumtext	NO		NULL	
foodPrice	decimal(5,2)	NO		NULL	
businessId	int	NO		NULL	
remarks	varchar(40)	YES		NULL	

2.1.7 likes表

Field	Type	Null	Key	Default	Extra
userId	varchar(20)	NO	PRI	NULL	
businessId	int	NO	PRI	NULL	

2.1.8 orderdetail表

Field	Type	Null	Key	Default	Extra
odId	int	NO	PRI	NULL	auto_increment
foodName	varchar(40)	YES		NULL	
orderId	int	NO		NULL	
foodId	int	NO		NULL	
quantity	int	NO		NULL	
priceAtThatTime	decimal(5,2)	NO		0	

2.1.9 orderdetail food表

Field	Type	Null	Key	Default	Extra
odId	int	YES		NULL	
orderId	int	YES		NULL	
foodId	int	YES		NULL	
quantity	int	YES		NULL	
priceAtThatTime	decimal(5,2)	YES		NULL	
foodName	varchar(255)	YES		NULL	
foodExplain	varchar(255)	YES		NULL	
foodImg	varchar(255)	YES		NULL	
businessId	int	YES		NULL	
remarks	varchar(255)	YES		NULL	

2.1.10 orders表

Field	Type	Null	Key	Default	Extra
orderId	int	NO	PRI	NULL	auto_increment
userId	varchar(20)	NO		NULL	
businessId	int	NO		NULL	
orderDate	date	YES		NULL	
orderTotal	decimal(7,2) unsigned zerofill	NO		0	
daId	int	YES		NULL	
orderState	int	NO		0	

2.1.11 remarks表

Field	Type	Null	Key	Default	Extra
remark	varchar(255)	YES		NULL	
businessId	int	YES		NULL	
remarkDate	date	YES		NULL	
userId	varchar(20)	YES		NULL	
remarkId	int	NO	PRI	NULL	auto_increment
userName	varchar(20)	YES		NULL	

2.1.12 searchhistory表

Field	Type	Null	Key	Default	Extra
userId	varchar(40)	NO		NULL	
searchContent	varchar(255)	NO		NULL	

2.1.13 user表

Field	Type	Null	Key	Default	Extra
userId	varchar(20)	NO	PRI	NULL	
password	varchar(20)	NO		NULL	
userName	varchar(20)	NO	MUL	NULL	
userSex	int	NO		1	
userImg	mediumtext	YES		NULL	
delTag	int	NO		1	

2.2 前端设计

2.2.1 底部导航栏

首页：返回主界面。

我的：包含用户的个人中心，如订单、收藏、账户设置等。

发现：包含发现新商家或新活动的功能。

订单：查看和管理用户的订单。

2.2.2 首页

顶部导航栏：

登录和注册按钮，允许用户进行账户管理。

天津大学北洋园校区的下拉菜单，用于选择不同的校区或位置。

搜索栏：

提供搜索功能，用户可以搜索饿了么平台上的商家或商品名称。

分类标签：

早餐美食：可能包含早餐相关的食品选项。

跑腿代购：提供代购服务，如汉堡披萨、甜品饮品等。

速食简餐：快速方便的食品选择。

地方小吃：提供地方特色小吃。

米粉面馆：提供米粉、面条、包子、粥、炸鸡、炸串等食品。

品质套餐：提供搭配齐全的套餐选项。

促销活动：

立即抢购：可能是限时抢购或特价活动的入口。

超级会员：提供会员服务，每月享受超值权益。

商家推荐：

推荐商家列表，显示商家名称、评分、月销量、配送信息等。

筛选和排序功能：

筛选：允许用户根据综合排序、距离最近、销量最高等条件筛选商家。

排序：提供排序功能，用户可以根据不同的标准对商家进行排序。

商家详情：

万家饺子（软件园E18店）：显示商家的详细信息，包括评分、月销量、配送费用、起送价、配送距离和预计送达时间。

蜂鸟专送：表明配送服务由蜂鸟专送提供。

2.2.3 登录

密码输入区域：

两个密码输入框，用于输入密码和确认密码。

操作按钮：

登录按钮：用于提交登录信息。

去注册链接：引导新用户注册账户。

2.2.4 商家列表

标题：

商家列表：标明这是商家列表的页面。

商家信息：

万家饺子（软件园E18店）：商家名称，包括分店信息。

商品种类：各种饺子，说明商家提供的主要商品。

小锅饭豆腐馆（全运店）：商家名称，包括分店信息。

商品种类：小锅套餐，说明商家提供的主要商品。

米村拌饭（浑南店）：商家名称，包括分店信息。

商品种类：拌饭、拌饭套餐、串，说明商家提供的主要商品。

申记串道（中海康城店）：商家名称，包括分店信息。

商品种类：烤串、炸串，说明商家提供的主要商品。

2.2.5 商家信息

食品名称：食品的名称，包括烹饪方式。

食品描述：描述食品的主要食材和特点。

价格信息：食物的价格

配送费用：另需配送费，说明除了食品价格外，还需要额外支付的费用。

起送金额：标明用户下单的最低金额要求。

2.2.6 商单信息

订单标题：

确认订单：标明这是用户需要确认的订单信息。

配送信息：订单配送至：提示用户订单将被配送到的地址，具体的配送地址，收货人的联系电话。

商家信息：订单来自的商家名称和分店信息。

订单明细：用户所点的食品名称和数量，食品的价格，订单的配送费用。

支付按钮：用户点击此按钮进行支付。

订单总额：包括食品价格和配送费用的订单总金额。

2.2.7 支付信息

支付标题：

在线支付：标明这是用户进行在线支付的页面。

订单信息：

订单信息：提示用户这是他们即将支付的订单详情。

商家及订单金额：

订单来自的商家名称和分店信息。

订单的总金额，可能包括食品价格和配送费用。

支付方式：

支付宝：提供支付宝作为支付选项。

微信支付：提供微信支付作为支付选项。

支付按钮：用户点击此按钮以确认支付订单。

2.3 后端设计

2.3.1 “我的”页面新增api

3.1.1.头像与昵称的获取接口：（原接口）

UserController / getUserByIdByPass

参数：userId、password

返回值：user对象

3.1.2.修改头像：（新增接口）

UserController / changeUserAvatar

参数：userId , base64 (头像的base64编码)

返回值：int（影响的行数）

3.1.3.修改昵称：（新增接口）

UserController / changeUserName

参数：userId , userName

返回值：int (影响的行数)

3.1.4.修改密码:（新增接口）

UserController / changeUserPassword

参数：userId , oldPassword , newPassword

返回值：int (影响的行数)

3.1.5 UserController/saveUser

参数：userId , password , userName , userSex , userImg (可选)

返回值：int（影响的行数）

功能：向用户表中添加一条记录

2.3.2 评论和展示评论

3.2.1 RemarkController / listRemarksByBusinessId

参数：businessId

返回值：remark数组

功能：根据商家编号查询所属商家的所有评论信息

3.2.2 RemarkController / saveRemarks

参数：remark,userId,userName,businessId

返回值：int（评论编号）

功能：向评论表中添加一条记录，并返回评论编号

3.2.3 RemarkController / removeOneRemark

参数：userName userId businessId remark

返回值：int 影响的行数

功能：用户删除在某商家下的一条评论

2.3.3 收藏商家与展示收藏列表

3.3.1 FavoriteController / listFavoriteByUserId

参数：userId

返回值：businessId数组

功能：查询用户收藏的所有商家

3.3.2 FavoriteController / saveFavoriteBusinessId

参数：userId , businessId

返回值：int （影响的行数）

功能：当前用户收藏该商家（向收藏表中添加一条数据）

3.3.3 FavoriteController / removeFavoriteBusinessId

参数：userId , businessId

返回值：int （影响的行数）

功能：当前用户取消收藏该商家

2.3.4 点赞

3.4.1 LikesController / getLikesBybusinessId

参数：businessId

返回值：int (此商家的点赞总数量)

功能：查询某商家的点赞总数量

3.4.2 LikesController / saveLikes

参数：userId , businessId

返回值：int （影响的行数）

功能：用户点赞某一个商家

3.4.3 LikesController / removeLikes

参数：userId , businessId

返回值：int （影响的行数）

功能：用户取消点赞某一个商家

3.4.4 LikesController / getLikesByUserIdByBusinessId

参数：userId , businessId

返回值：int （0：之前此用户对此商家未点赞

1：之前此用户对此商家点过赞）

功能：判断此用户对此商家之前有没有点过赞，

若点过赞，则再点一次是取消点赞

若未点过赞，则点一次是点赞

2.3.5 私信

3.5.1 ChatsController / removeChatsAllByCurrentUserId

参数: `currentUserId`

返回值: `int` (影响的行数)

功能: 清空当前用户的私信列表

3.5.2 ChatsController / removeChatsByTwoUserIdByMessage

参数: `currentUserId` , `receiverUserId` , `message`

返回值: `int` (影响的行数)

功能: 删除当前用户`currentUserId`的与
用户`receiverUserId`的内容为`message`的消息

3.5.3 ChatsController / recallChatsByTwoUserIdByMessage

参数: `currentUserId` , `receiverUserId` , `message`

返回值: `int` (影响的行数)

功能: 撤回当前用户`currentUserId`的发送给
用户`receiverUserId`的内容为`message`的消息

3.5.4 ChatsController / getChatsByUserId

参数: `currentUserId`

返回值: `chats`数组

功能: 查询当前用户的所有私信列表

3.5.5 ChatsController / saveChats

参数: `senderUserId` , `receiverUserId` , `message`

返回值: `int` (影响的行数)

功能: 新增私信记录到总私信表中

2.3.6 商家登陆

3.6.1 BusinessController / saveBusiness

参数: `businessName` , `password`

返回值: `int` (商家编号`businessId`)

功能: 商家注册并返回代表商家的唯一编号作为账号

3.6.2 BusinessController / updateBusiness

参数: `businessId` , `businessAddress` ,

`businessExplain` , `businessImg` ,

`starPrice` , `deliveryPrice` , `orderId`

返回值: `int` (影响的行数)

功能: 更改(完善)商家的信息

3.6.3 FoodController / addFood

参数: businessId , foodName , foodExplain ,
foodImg , foodPrice

返回值: int (食品的标号foodId)

功能: 商家可以上架自己的商品

3.6.4 BusinessController / getBusinessByIdByPass

参数: businessId , password

返回值: int (影响的行数)

功能: 商家登录

2.3.7 搜索与搜索历史

3.7.1 SearchController / listBusiness

参数: searchContent , userId (需要更新userId的这条历史搜索)

返回值: business数组

功能: 根据关键词搜索商家列表

3.7.2 SearchController / getHistoryByUserId

参数: userId

返回值: searchContent

功能: 查询用户上一次的一条搜索记录

3.7.3 BusinessController / listBusinessByOrderTypeId

参数: orderTypeId

返回值: business数组

功能: 根据点餐分类编号查询所属商家信息

3.7.4 BusinessController / getBusinessById

参数: businessId

返回值: business对象

功能: 根据商家编号查询商家信息

3.7.5 RemarkController / listRemarksByBusinessId

参数: businessId

返回值: remark数组

功能: 根据商家编号查询所属商家的所有评论信息

2.3.8 原项目基础API

3.8.1 business

3.8.1.1 BusinessController/listBusinessByOrderTypeId

参数: orderTypeId

返回值: business数组

功能: 根据点餐分类编号查询所属商家信息

3.8.1.2 BusinessController/getBusinessById

参数: businessId

返回值: business对象

功能: 根据商家编号查询商家信息

3.8.2 food

3.8.2.1. FoodController/listFoodByBusinessId

参数: businessId返回值: food数组

功能: 根据商家编号查询所属食品信息

3.8.3 cart

3.8.3.1 CartController/listCart

参数: userId、businessId (可选)

返回值: cart数组 (多对一: 所属商家信息、所属食品信息) 功能: 根据用户编号查询此用户所有购物车信息

根据用户编号和商家编号, 查询此用户购物车中某个商家的所有购物车信息

3.8.3.2 CartController/saveCart

参数: userId、businessId、foodId返回值: int (影响的行数)

功能: 向购物车表中添加一条记录

3.8.3.3 CartController/updateCart

参数: userId、businessId、foodId、quantity返回值: int (影响的行数)

功能: 根据用户编号、商家编号、食品编号更新数量

3.8.3.4 CartController/removeCart

参数: userId、businessId、foodId (可选) 返回值: int (影响的行数)

功能: 根据用户编号、商家编号、食品编号删除购物车表中的一条食品记录根据用户编号、商家编号删除购物车表中的多条条记录

3.8.4 deliveryAddress

3.8.4.1 DeliveryAddressController/listDeliveryAddressByUserId

参数: userId

返回值: deliveryAddress数组

功能: 根据用户编号查询所属送货地址

3.8.4.2 DeliveryAddressController/getDeliveryAddressById

参数: daId

返回值: deliveryAddress对象

功能: 根据送货地址编号查询送货地址

3.8.4.3 DeliveryAddressController/saveDeliveryAddress

参数: contactName、contactSex、contactTel、address、userId返回值: int (影响的行数)

功能: 向送货地址表中添加一条记录

3.8.4.4 DeliveryAddressController/updateDeliveryAddress

参数: daId、contactName、contactSex、contactTel、address、userId返回值: int (影响的行数)

功能: 根据送货地址编号更新送货地址信息

3.8.4.5 DeliveryAddressController/removeDeliveryAddress

参数: daId

返回值: int (影响的行数)

功能: 根据送货地址编号删除一条记录

3.8.5 orders

3.8.5.1 OrdersController/createOrders

参数: userId、businessId、daId、orderTotal返回值: int (订单编号)

功能: 根据用户编号、商家编号、订单总金额、送货地址编号向订单表中添加一条记录, 并获取自动生成的订单编号,

然后根据用户编号、商家编号从购物车表中查询所有数据, 批量添加到订单明细表中, 然后根据用户编号、商家编号删除购物车表中的数据。

3.8.5.2 OrdersController/getOrdersById

参数: orderId

返回值: orders对象 (包括多对一: 商家信息; 一对多: 订单明细信息)

功能: 根据订单编号查询订单信息, 包括所属商家信息, 和此订单的所有订单明细信息

3.8.5.3 OrdersController/listOrdersByUserId

参数: userId

返回值: orders数组 (包括多对一: 商家信息; 一对多: 订单明细信息) 功能: 根据用户编号查询此用户的所有订单信息

新增:

3.8.5.4.OrdersController / payOk

参数: orderId

返回值: int (影响的行数)

功能: 改变订单属性, 将未支付0转成已支付1

3.8.6 user

3.8.6.1 UserController/getUserByIdByPass

参数: userId、password返回值: user对象

功能: 根据用户编号与密码查询用户信息

3.8.6.2 UserController/getUserById

参数: `userId`

返回值: `int` (返回行数)

功能: 根据用户编号查询用户表返回的行数

第三章 项目测试文档

3.1 项目背景

饿了么是一款面向各年龄段人群的外卖软件，让目标客户足不出户享受形形色色各个菜系的美食，并且让用户方便地了解不同商家的信息和菜品。

3.2 编写目的

该文档的目标人群是软件的开发团队以及使用软件的目标客户，以及软件的功能审核人员。给出测试方式和测试结果，方便审核人员对软件功能的进一步审核，也能够使用户对软件的功能更加了解。

3.3 测试人员

本次测试的主要人员是谢帛洋和杨宇鑫。

3.4 测试目标

在用使用软件之前，尽可能的发现软件中存在的错误和不合理之处，排除软件中存在的错和不合理之处，排出软件中潜在的错误，最终把高质量的软件系统交付给用户。系统的测试覆盖范围：功能、性能、UI、安全性、兼容性。

3.5 测试要求

测试系统	Windows11
数据库版本	Mysql 8.0
其他	VsCode、IDEA、Eclipse

3.6 测试方法

系统的功能测试选用了手工测试，运用黑盒测试中的等价类划分、边界值分析、错误推断、因果图法。

UI方面的测试包括：易用性测试、规范性测试、帮助设施测试、合理性测试、美观与协调性测试、独特性测试、快捷方法组合组合测试。

系统的安全性、兼容性、配置测试也是手工测试

单元测试采用方法是白色测试，功能测试采用黑盒测试

3.7 测试内容

3.7.1 单元测试

首先依照系统、子系统和模块进行划分名单时最终的单元必须是功能模块，

或者面向对象过程中的若干类，单元测试是对功能模块进行正确性验证的测试工作，也是后续测试的基础。目的在于发现各模块内部可能存在的各种差错，因此需要从程序内部结构出发设计测试用例，着重考虑以下五个方面：

模块接口：对所测模块的数据流进行测试。

局部数据结构：检查不正确不一致的数据类型说明、适用尚未赋值或者尚未初始化的变量、错误的初始值或者缺省值

路径：设计测试用例查找由于不正确计算（算法错、表达式的符号不正确、运算精度不够等）不正确的比较或者不正常的测试流（包括不同数据类型的相互比较、不适当地修改了循环变量、错误的或不可能的循环终止条件等）检查模块有没有对于常见的条件设计比较完善的错误处理功能，保证其逻辑上的正确性，注意设计数据流、控制流中刚好等于、大于或小于确定的比较值的用例。

3.7.2 集成测试

集成测试也叫组装测试、联合测试。通常在单元测试的基础上需要将所有的模块按照设计要求组装系统，这时需要考虑的问题如下：

- （1）把各个模块连接起来，模块接口的数据是否会丢失
- （2）一个模块的功能是否会对另一个模块的功能产生不利的影响
- （3）全局数据结构是否有问题
- （4）单元模块的误差积累起来，是否会放大，从而达到不能接受对策程度。

我们在组装的时候可以参考采用一次性组装方式或者增值式组装方式

3.7.3 系统测试

（1）功能测试

验证系统功能是否符合其需求规格说明书，核实系统功能上是否完整，没有冗余和遗漏功能。详细介绍如下表：

测试范围	验证数据精确度、数据类型、业务功能等相关方面的正确性
测试目标	核实所有功能均已正常实现、即是否与需求一致
技术	采用黑盒测试、边界测试、等价类划分测试方法
工具与方法	手工测试
开始标准	开发阶段对应的功能完成并且测试用例设计完成
完成标准	测试用例通过并且高级缺陷全部解决
测试范围	验证数据精确度、数据类型、业务功能等相关方面的正确性

3.7.4 用户界面测试

测试用户界面是否具有导航性、美观性、规范性、是否满足设计中客户要求的执行功能、详细介绍如下边UI测试：

测试范围	保证窗口运行、提示信息、页面跳转、易于使用等方面的正确性
测试目标	核实各个窗口的风格（包括颜色、字体、提示信息、图标、title等）均与需求保持一致
技术	Web 测试通用方法
工具与方法	手工测试、目测
开始标准	界面开发完成
完成标准	UI 符合可接受标准，能保证用户界面的友好性，易操作性，而且符合用户操作习惯

3.7.5 性能测试

测试相应时间、事务处理效率和其他时间敏感的问题。介绍如下表：

测试范围	多用户长时间在线操作时性能方面的测试
测试目标	核实系统在大流量的数据与多用户操作时软件性能的稳定性，不造成系统崩溃或者相关数据丢失
技术	手动测试、自动化测试
开始标准	系统测试
完成标准	系统满足用户需求的性能要求

3.7.6 兼容性测试

测试软件在不同平台上的使用的兼容性。介绍如下：

测试范围	1. 使用不同版本的浏览器、分辨率、操作系统分别进行测试 2.不同操作系统、不同设备、浏览器、分辨率等各种条件的组合测试
测试目标	核实系统在不同软件和硬件配置中运行稳定
技术	黑盒测试 手工测试
开始标准	系统测试
完成标准	在各种不同版本不同类型浏览器、操作系统或者其组合下均能正常实现其功能(测试根

3.7.7 安全性测试

测试范围	用户、管理员的密码安全、权限
测试目标	用户、管理员密码管理、应用程序级别的安全性、核实用户只能操作其所有权限操作
技术	黑盒测试
工具与方法	手工测试
开始标准	系统测试

3.7.8 配置测试

测试在不同网络、服务器、工作站的不同软硬件配置条件下，软件系统的质量，详细说明见下表：

测试范围	不同网络、服务器、不同软硬件配置条件
测试目标	核实系统在不同的软硬件配置条件下系统的质量是否达到标准
技术	黑盒测试
工具与方法	手工测试
开始标准	系统开发完成后
完成标准	达到相关要求
测试重点与优先级	测试优先级以测试需求优先级为参照
需考虑的特殊事项	软硬件设备问题

3.7.9 回归测试

测试范围	所有功能、用户界面、兼容性、安全性等测试类型
测试目标	核实执行所有测试类型后功能、性能、等均达到用户需求所要求的标准
技术	黑盒测试
工具与方法	手工测试、自动化测试
开始标准	每当被测试的软件或其开发环境改变时，在每个核实的测试阶段上进行回归测试
完成标准	95% 的测试用例执行通过并通过系统测试
测试重点与优先级	测试优先级以测试需求的优先级为参照
需考虑的特殊事项	软硬件设备问题

3.8 测试结果

3.8.1 测试中存在的问题

3.8.1.1 AI咨询功能在前端显示时会出现无返回结果的问题

3.8.1.2 修改头像约有10%的数据传不到数据库

3.8.2 项目评价

在测试的过程中，测试团队并没有发现较大的bug，所述功能均可以基本实现，包括用户注册、登录以及基本的点餐流程、修改个人资料等功能。并且网页前端设计简洁清晰，对用户的引导性强，同时兼具了一定的美观性考量。

3.9 测试用例

第四章 部署文档

4.1 项目背景

4.1.1 市场需求分析

随着互联网技术的飞速发展和人们生活节奏的加快，外卖订餐服务已成为现代都市生活中不可或缺的一部分。越来越多的人选择通过手机应用程序或网站来订购美食，享受便捷的送餐服务。据统计，近年来外卖市场呈现出快速增长的趋势，市场规模不断扩大，用户数量持续攀升。因此，开发一款高质量的在线订餐平台具有巨大的市场需求和发展潜力。

4.1.2 用户痛点分析

尽管市场上已经存在多家知名的外卖平台，但在实际使用过程中，用户仍然面临诸多不便：

4.1.2.1 选择困难

面对众多的餐厅选项，用户往往难以快速做出决策。

4.1.2.2 配送延迟

高峰时段订单量大，导致配送时间延长，影响用餐体验。

4.1.2.3 食品安全

部分小餐馆卫生条件差，食品质量难以保证。

4.1.2.4 个性化推荐不足

现有的推荐算法不够精准，无法满足用户的个性化需求。

4.1.3 商家需求分析

对于餐饮商家而言，加入一个高效的外卖平台同样具有重要意义：

4.1.3.1 拓宽销售渠道

通过线上平台吸引更多的顾客，提高销售额。

4.1.3.2 降低运营成本

减少线下宣传费用，利用平台流量优势增加曝光度。

4.1.3.3 提升服务质量

借助平台提供的数据分析工具，优化菜品结构和服务流程。

4.1.4 项目目标

基于以上分析，“饿了么”项目旨在打造一个高效、便捷且安全的在线订

餐平台，具体目标包括：

4.1.4.1 丰富菜品选择

整合周边优质餐厅资源，提供多样化的菜品供用户选择。

4.1.4.2 优化配送流程

引入先进的物流管理系统，缩短配送时间，提升用户体验。

4.1.4.3 保障食品安全

严格筛选合作商家，确保所有上线餐厅均符合卫生标准。

4.1.4.4 智能推荐系统

利用大数据分析技术，根据用户喜好和历史订单记录进行个性化推荐。

通过实现这些目标，“饿了吧”项目将为用户提供更加优质的订餐体验，同时也为商家带来更多的商业机会，实现双赢的局面。

4.2 系统需求

4.3 项目部署的系统需求

4.3.1 前端系统需求

4.3.1.1 操作系统

- 支持主流的操作系统，包括 Windows、macOS 和 Linux。

4.3.1.2 浏览器兼容性

- 支持最新版本的 Chrome、Firefox、Safari 和 Edge 浏览器。

4.3.1.3 移动设备

- 支持 iOS 和 Android 操作系统的移动设备。

4.3.2 后端系统需求

4.3.2.1 服务器配置

- CPU：至少 2 核心
- 内存：至少 2 GB
- 硬盘：至少 5 GB 可用空间

4.3.2.2 数据库

- MySQL 5.7 或更高版本

4.3.2.3 中间件

- Nginx 作为反向代理服务器
- Redis 用于缓存和会话管理
- RabbitMQ 用于消息队列

4.3.2.4 其他依赖

- Java 8 或更高版本
- Spring Boot
- Maven

4.3.3 网络需求

4.3.3.1 IP 地址

- 具备公网 IP 地址，以便外部访问。

4.3.4 安全性需求

4.3.4.1 数据加密

- 使用 HTTPS 协议，确保数据传输的安全性。

4.3.5 运维需求

4.3.5.1 持续集成/持续部署 (CI/CD)

- 使用宝塔面板实现构建、测试和部署流程。

4.3.5.2 日志管理

- 收集和分析系统日志，快速定位问题。

4.4 配置说明

4.4.1 数据库配置

4.4.1.1 数据库类型

- MySQL

4.4.1.2 数据库连接信息

- 主机: `http://140.143.151.181/`
- 端口: 3306
- 数据库名称: `elm`
- 用户名: `admin`
- 密码: 123456

4.4.1.3 数据库表结构

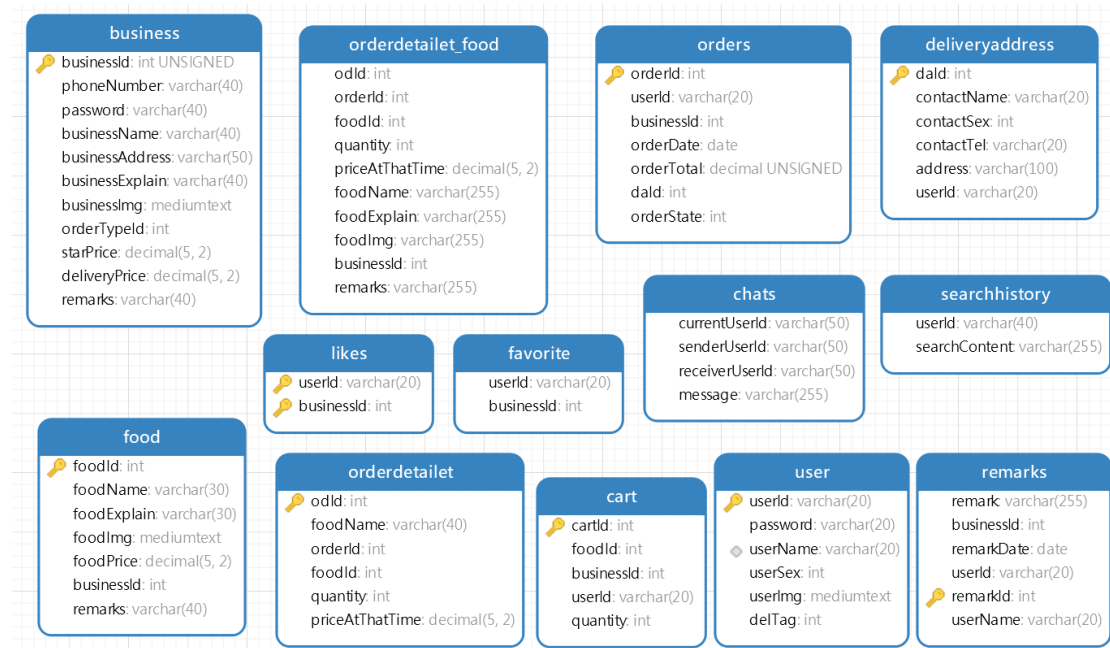


图 4-1 数据库表结构

4.5 部署的步骤和流程

4.5.1 服务器配置

1. 安装 Ubuntu 20.04 LTS 服务器。
2. 安装 Nginx 和 MySQL。
3. 安装 Java 和 Spring Boot。
4. 安装 Redis 和 RabbitMQ。

4.5.2 前端部署

1. 将前端代码部署到服务器上。
2. 配置 Nginx，将前端请求重定向到服务器。

4.5.3 后端部署

1. 将后端代码部署到服务器上。
2. 配置 Nginx，将后端请求重定向到服务器。

4.5.4 数据库部署

1. 将数据库部署到服务器上。

4.5.5 其他配置

1. 安装宝塔面板，方便操作
2. 配置 Nginx，将前端和后端请求重定向到服务器。
3. 配置日志管理，快速定位问题。

第五章 人员分工及项目进度

- 人员分工：
 - 杨宇鑫：负责后端与部署
 - 谢帛洋：负责前端与测试
- 项目进度：
 - 第一周：学习了项目一的内容，复盘了项目一的代码；
 - 第二周：学习了项目二到项目四的内容，复盘了项目二到项目四的代码，饿了吧V1.0项目复盘完成；
 - 第三周：创新了原项目，饿了吧V2.0项目完成。

第六章 项目分析及开发过程

6.1 项目分析

6.1.1 项目一: JDBC

elm_admin 是饿了么 **JDBC** 版项目，采用了 **JDBC+MySQL** 开发，是纯后端的字符界面操作数据库的命令行应用程序。

6.1.1.1 项目技术架构

- JDK 1.8
- JDBC
- MySQL 数据库

6.1.1.2 开发工具

- STS (spring-tool-suite)
- mysql-5.5.62-winx64
- Navicat Premium 8

6.1.1.3 安装部署指南

- 安装 jdk、STS、MySQL
- 在 MySQL 数据库中创建数据库 **elm_admin**，使用数据库脚本 **elm_admin.sql** 创建数据库和初始数据。
- 在 STS 中导入 JavaSE 项目。
- 打开 **com/neusoft/elm/util/DBUtil** 修改数据库密码
- 本项目有两个入口：管理员入口、商家入口。
 - 运行 **ElmAdminEntry** 中的 **main** 函数为管理员入口。
 - 运行 **ElmBusinessEntry** 中的 **main** 函数为商家入口。

6.1.1.4 整体要求

1. 项目技术架构
 - JDK8
 - JDBC
 - MySql
2. 开发工具
 - STS (SpringToolSuite4)
 - mysql-5.5.62-winx64
 - navicat

3. 涉及的技术点

- 封装 JDBC
- 封装 DAO
- 领域模型中的实体类
- 增删改查操作
- 多条件模糊查询
- JDBC 事务管理
- 表的主外键关系

6.1.2 项目二: Vue3

饿了么前端版项目是采用 **HTML**、**CSS**、**JavaScript** 开发的前端静态网页项目。

6.1.2.1 项目前端技术架构

- HTML
- CSS
- JavaScript

6.1.2.2 开发工具

- HBuilder
- Chrome 浏览器

6.1.2.3 安装部署指南

- 安装 HBuilder、Chrome
- 将工程导入到 HBuilder 中
- 在 Chrome 浏览器中运行 index.html 文件
- 在 Chrome 浏览器中使用 Toggle device toolbar 模拟手机浏览

6.1.2.4 整体要求

1. 项目技术架构
 - HTML5
 - CSS3
 - JavaScript (ES6 以上)
2. 开发工具
 - HBuilder
 - Chrome 浏览器
3. 涉及的技术点
 - HTML5 标签的使用
 - CSS3 样式的使用

- JS 对 DOM 的基本操作
- DIV+CSS 布局基础
- 移动端布局基础
- viewport 设置
- 弹性布局
- 边框盒子模型
- vw 与 vh 的使用
- 图片按比例自适应
- CSS3 小图标的使用
- 第三方字体库的使用

6.1.3 项目三: Servlet

饿了么 **Servlet** 版本

6.1.3.1 项目演示

- 运行“饿了么项目”，演示应用程序效果，演示“点餐业务线”整体流程。

- 本项目参照“饿了么官网网页版”制作。饿了么网页版：<http://h5.ele.me/>

- 本项目专注于完成点餐业务线功能，“饿了么官网”中的其它功能暂不涉及。

6.1.3.2 项目目标

- 本项目为课程级贯穿项目中的第三个项目（JDBC项目、前端项目、JavaWeb项目）。

- 本项目完成后，学员将能够使用 Vue + Servlet + AJAX 技术开发前后端分离的 Web 应用程序。

6.1.3.3 项目中所涉及到相关知识点

- AJAX 的使用
- Servlet 的使用
- Session 的使用
- 简单 MVC 封装
- Service 层事务管理
- DAO 层批量操作
- 多对一与一对多的映射
- 服务器端 JSON 数据转换
- VueCLI 的使用
- 多条件模糊查询的使用

- SVN、Git 版本控制工具的使用

6.1.3.4 数据库设计

- 本项目完成后，学员将能够使用 Vue + Servlet + AJAX 技术开发前后端分离的 Web 应用程序。

6.1.3.5 整体要求

1. 项目技术架构

- JDK8
- Servlet
- Tomcat5.5
- MySQL
- Vue

2. 开发工具

- HBuilder
- STS (SpringToolSuite4)
- mysql-5.5.62-winx64
- Tomcat8.5

3. 涉及的技术点

- AJAX 的使用
- Servlet 的使用
- Session 的使用
- 简单 MVC 封装
- Service 层事务管理
- DAO 层批量操作
- 多对一与一对多的映射
- 服务器端 JSON 数据转换
- VueCLI 的使用
- 多条件模糊查询的使用

6.1.4 项目四: SpringBoot

饿了么 **SpringBoot** 版本

6.1.4.1 整体要求

1. 项目技术架构

- JDK8
- SpringBoot
- MyBatis

- MySQL
- Vue
- 2. 开发工具
- HBuilder
- STS (SpringToolSuite4)
- mysql-5.5.62-winx64
- Tomcat8.5
- Maven

3. 涉及的技术点

- AJAX 的使用
- SpringBoot 框架的使用
- MyBatis 框架的使用
- 封装 Mapper
- Service 层事务管理
- 数据层批量操作
- 多对一与一对多的映射
- 服务器端 JSON 数据转换
- VueCLI 的使用
- 多条件模糊查询的使用

6.2 开发过程

- 项目进度：
 - 第一周：学习了项目一的内容，复盘了项目一的代码；
 - 第二周：学习了项目二到项目四的内容，复盘了项目二到项目四的代码，饿了吧V1.0项目复盘完成,并完成了前后端的部署；
 - 第三周：创新了原项目，饿了吧V2.0项目完成，前后端数据库都部署在了腾讯云服务器上，并制作了简易的手机apk

6.3 gitee上团队成员贡献



图 6-1 Gitee提交数据1



图 6-2 Gitee提交数据2



图 6-3 Gitee提交数据3



图 6-4 Gitee提交数据4

第七章 开发过程

7.1 Bug修复以及简单功能的完善

- **Bug修复:**
 - 修复总价浮点数精度 bug;
 - 修复页面显示不全的 bug。
- **功能完善:**
 - 手机号的正则表达式检验;
 - 强密码检验。

7.2 饿了么V2.0版本的创新点

7.2.1 实现前端 Vue2 到 Vue3 的升级

7.2.2 实现“我的”页面



图 7-1 实现“我的”页面

7.2.3 实现修改昵称，修改头像，修改密码功能



图 7-2 实现修改昵称，修改头像，修改密码功能

7.2.4 实现收藏功能：收藏商家，取消收藏，收藏列表

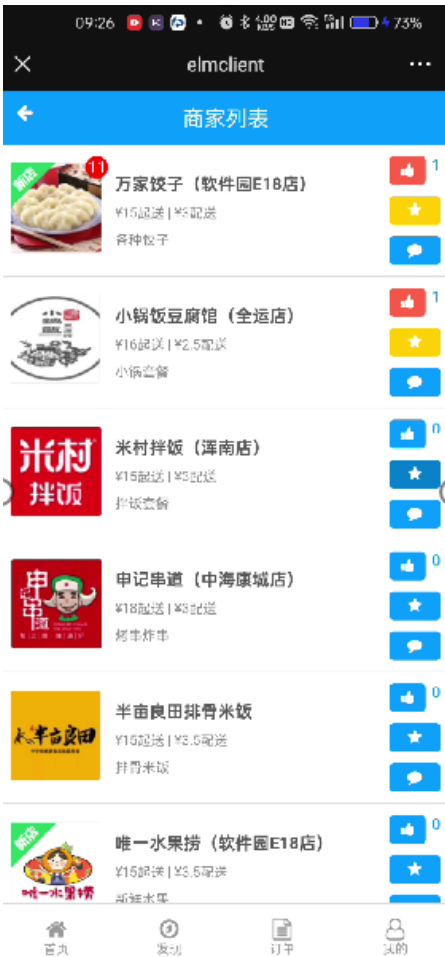


图 7-3 实现收藏功能：收藏商家，取消收藏，收藏列表

7.2.5 实现点赞功能：点赞，取消点赞，展示商家总点赞量

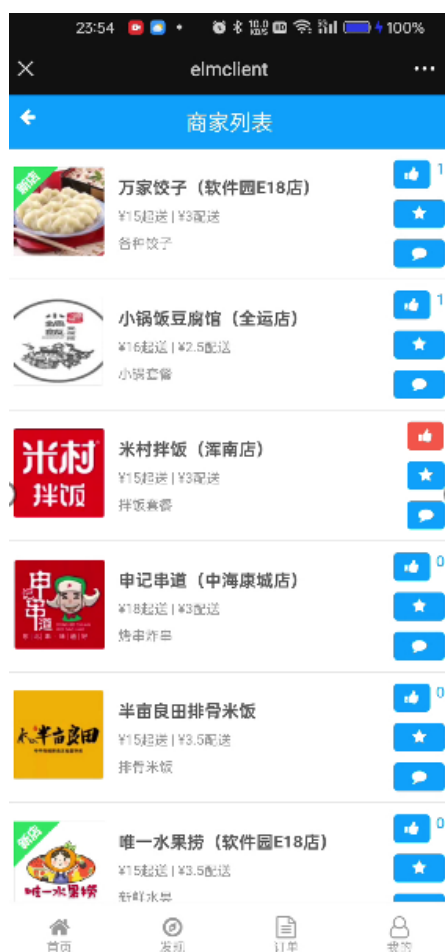


图 7-4 实现点赞功能：点赞，取消点赞，展示商家总点赞量

7.2.6 实现评论功能：新增评论，展示评论列表，删除评论



图 7-5 实现评论功能：新增评论，展示评论列表，删除评论

7.2.7 实现搜索功能：模糊匹配，可以搜索商家或商品



图 7-6 实现搜索功能：模糊匹配，可以搜索商家或商品

7.2.8 实现历史记录功能：搜索的时候展示搜索历史



图 7-7 实现历史记录功能：搜索的时候展示搜索历史

7.2.9 实现商家账号注册，登录，上架商品，修改商家个人信息

←

提交商家信息

食品名称:

食品名称

食品简介:

食品简介

上传食品图片:

选择文件

未选择任何文件

食品价格:

0

上架商品

图 7-8 实现商家账号注册，登录，上架商品，修改商家个人信息

7.2.10 集成文心一言接口，充当智能客服，方便用户对食品的咨询

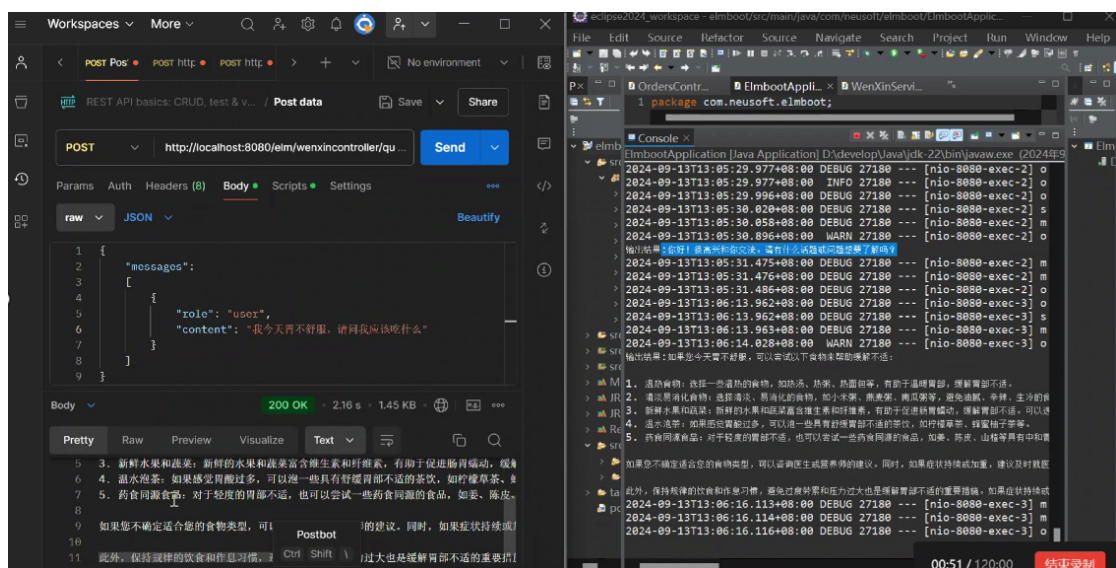


图 7-9 集成文心一言接口，充当智能客服，方便用户对食品的咨询

7.3 部署成果

7.3.1 在腾讯云服务器上部署：前端，后端，数据库

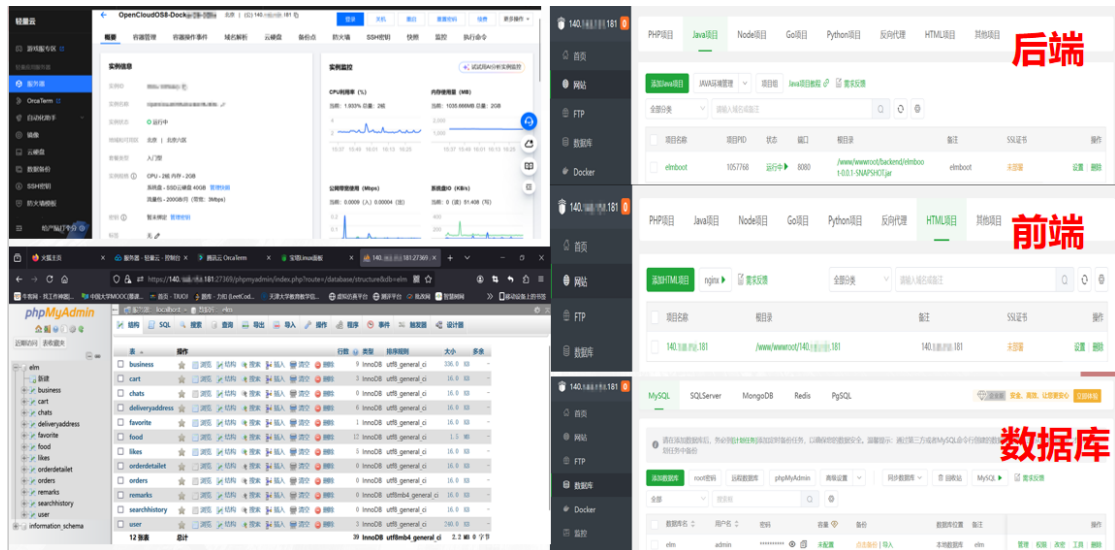


图 7-10 在腾讯云服务器上部署：前端，后端，数据库

7.3.2 创建手机 APK



图 7-11 创建手机 APK

第八章 问题，解决对策及反思

8.1 浮点数精度

问题：如果出现3个0.1相加的情况，得到的结果就是0.3000004

解决方案：用字符串表示小数，然后通过字符串相加的算法，得到最终结果

8.2 页面显示不全

问题：有些机型在页面下拉的时候，由于底部有遮挡，没有办法下拉到最下边

解决方案：底部适当留白

反思：站在用户的角度，完善页面布局，给用户最好的体验

8.3 历史订单的价格不能随着新的商品价格变化而变化

问题：当改变数据库一个商品价格的时候，之前已经购买的商品的价格也会变化

解决方案：在原有的订单明细表中，增加一个属性：商品的priceAtThatTime，记住商品在购买时的价格，当展示订单明细的时候，不用在数据库中去找现在的商品，而是直接在订单明细表中拿到当时的商品价格

反思：当想要保留一个数据的时候，可以直接在数据库中增加属性，在创建的时候，可以直接将值保留进数据库，在想用的时候直接从数据库中拿到想要的值就可以了

8.4 商品详情页的图片不能显示

问题：商品详情页的图片不能显示，因为图片的url是相对路径，需要将图片的url改为绝对路径

解决方案：在数据库中，将图片的url改为绝对路径，在商品详情页中，通过绝对路径来获取图片

反思：在数据库中，将图片的url改为绝对路径，在商品详情页中，通过绝对路径来获取图片，这样，在商品详情页中，图片的url就无需再通过服务器来获取了

8.5 搜索

问题：搜索功能，只能搜索到商家，无法搜索到想要的商品

解决方案：强化搜索的sql语句：先将商家与商品按照商家的id进行左连接，

并筛选出带有关键词的商家or商品，将筛选出满足条件的所有商家的id，再显示所有的商家id在（in）刚才筛选出的id中的商家即可

反思：优化sql语句，可以实现更强大的功能

8.6 手机号和密码的强度检验

问题：如果没有检验，用户使用的手机号和密码将会泛滥，失去控制

解决方案：使用正则表达式进行手机号与密码的强度判断

8.7 点赞，收藏，评论的存储

问题：用户的点赞，收藏，评论如何永久保留

解决方案：在数据库中新建点赞，收藏，评论的表结构，并约束好主键外键，设计索引以便更方便的查询数据库中的数据

8.8 文心一言的接口

问题：在javaweb工程中，如何调用文心一言的接口

解决方案：通过查询文心一言api的官网，根据教程与代码，在javaweb工程中，通过HttpClient调用文心一言的接口，并获取返回结果

8.9 部署到云端

问题：如何将本地的javaweb工程部署到云端，并让云端能访问到前端后端以及数据库

解决方案：注册腾讯云账号，开通一个云服务器，因为云服务器上的操作系统是linux，不方便操作，所以安装宝塔面板，通过数据库部署，html前端部署，java后端部署以及一些配置完成云端部署，并可以通过公网ip访问到云端的项目

第九章 成果与展望

经过团队成员共同努力，“饿了吧”项目不仅按期完成了既定目标，还超预期地实现了多项创新功能。未来我们将继续关注，不断迭代优化产品，力求提供更加优质的服务。同时，我们也将积极探索新技术的应用，如AI推荐算法等，以期进一步提升平台智能化水平

“饿了吧”项目的成功离不开每一位参与者的辛勤付出与智慧贡献。它不仅是对我们现有技术水平的一次检验，更为重要的是它代表了我们对未来发展方向的一种探索。希望以此为契机，能够激励更多人投身于技术创新之中，共同推动行业进步。