基于vue的校园二手交易app的设计 --- 以广东海洋大学为例

发现现象—分析现象，定义问题—对解决方案做需求分析--对解决方案做可行性分析—对app做系统设计—编写代码，实现功能—调试app，完善软件

设计总说明

绪论 > 设计背景及意义 + 二手交易app的现状和发展趋势 ---> 项目需求

软件开发及相关环境 > 开发环境 + 开发工具 + 开发相关技术

技术分析 > vue + vant + nodejs

前端静态实现 + 接口开发实现 + 数据库数据管理实现

项目布局与分析 > 总体开发流程与设计 + 页面UI设计 + 界面设计 + 项目目录结构

测试与发布 > 项目打包上线

设计总结

鸣谢

参考文献

3 可行性分析

3.1 经济可行性：

本软件面向的是海大所有学子，每个用户既是买家也是卖家，这不仅为由闲置物品的同学亦或是想入手物美价廉的物品的同学提供了一个渠道，而且没有中间商赚取差价；在方便同学们做交易的过程中，既能让买家清楚商品的品质，也能让卖家处置闲置物品。而且该软件占内存较小，面向的用户量不大，维护成本不高，并且推广后，可以十分方便地解决闲置物品处置的问题。这种便利受到同学们欢迎的成数是可观的。因此，该软件在经济上是完全可行的。

3.2 技术可行性：

本软件采用的是vue框架以及vant组件框架开发的前端页面，采用nodejs开发接口，采用mongodb数据库做数据管理，实现前后端数据交互，是项目开发常用技术，不需要其他任何复杂的技术或者应用软件的支持。综上，开发该软件是具有技术可行性的。

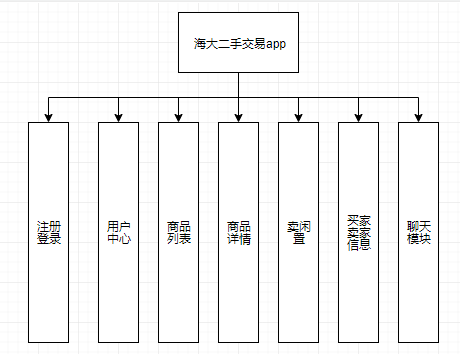
3.3 操作可行性：

本软件采用了滚动与分栏相结合的界面架构，所有操作都是生活中常用的操作，而且每个操作都是一目了然的，功能操作清晰可见，界面简洁清晰，操作方便。因此，该软件具备了操作可行性。

4 需求分析

4.1 功能需求

4.1.1 功能块划分



4.1.2 各功能模块概述

4.1.2.1 注册登录模块

1）用户通过学号、手机号、手机验证码注册并输入新密码成为平台用户

2）用户可以通过学号或手机号、密码登录到本app，从而进行商品浏览或者商品上传等操作

3）当用户忘记密码时，可通过注册手机的手机验证码的方式进行密码重置。

4.1.2.2 用户中心模块

1）用户可通过用户中心页面进入登录注册界面

2）登陆之后，可在用户中心设置个人姓名、宿舍、学号等信息

3）可查看个人已买、已卖、正在售卖、正在交易的商品

4）可对已买商品做评价操作，对正在售卖商品做编辑操作；

4.1.2.3 商品列表模块

1）针对不同类型的商品以标签切换的方式展示，通过点击不同类型进入各类商品列表

2）列表页展示一类商品的图片、几成新、卖家所在区域、以及价格

3）用户可通过下拉菜单选取查看东区、西区、中区各区的商品

4.1.2.4 商品详情模块

1）商品详情展示的是商品的照片、几成新、卖家转手原因、卖家名字、卖家所在区域、商品价格

2）通过点击卖家姓名可查看卖家信息

3）通过点击聊天图标可与卖家联系

4）通过点击商品详情折叠面板查看更多商品照片

4.1.2.5 卖闲置模块

1）可通过上传照片、编辑商品标题、价格、所在区域、转手原因、商品详情进行商品上架

4.1.2.6 买家卖家信息模块

1）展示用户的学号、所在区域、所在宿舍，以及用户以往交易过的商品所产生的评价记录

4.1.2.7 聊天模块

1）买家在有意愿与卖家做交易时，可通过聊天功能，与卖家协商好交易时间与交易地点等信息

6 app开发及相关环境

6.1 开发环境

本app开发是基于vue框架开发的webapp，其相关开发工具在window10运行，前期测试在chrome和firefox浏览器。

6.2 开发工具

6.2.1 Visual Studio Code（简称vsCode）

vsCode是一个轻量且强大的跨平台开源代码编辑器，支持windows、mac、linux系统。内置JavaScript、TypeScript和Nodejs支持，而且拥有丰富的插件生态系统，通过安装插件，几乎支持所有主流的编程语言的语法高亮、代码补全、自定义快捷键、括号配对。

vsCode内置了git版本控制，我们可以方便的提交、回滚、拉取和推送，而不需要自己创建各种‘最终版’文档。也可以通过快捷键（ctrl+shift+D）对代码进行调试，可以在图形界面上执行单步调试和多步调试。

6.2.2 mongodb数据库管理工具

MongoDB是一个基于分布式文件存储的数据库，即文件系统管理的物理资源是通过计算机网络和节点相连的，是一个介于关系数据库和非关系数据库之间的产品，是最像关系数据库的功能最为丰富的非关系数据库。MongoDB所支持的数据结构分散，支持类似于json的bson格式，所以可以存储类型复杂的数据。MongoDB最大的特点就是其强大的查询功能，其语法类似于面向对象语言，其特点为性能高、易于部署、易于使用、扩展性高。其主要功能特性为面向‘集合’存储（其基本单位是文档，一个文档即一条数据，是以‘键值对’的方式组成文档存储数据的，集合即一组文档），易于存储对象类型的数据，支持动态查询、完全索引，对于存储在数据库中的文档，我们不需要知道数据的结构。MongoDB支持windows、mac和linux系统。

6.2.3 Robo 3T数据可视化工具

Robo 3T是一款MongoDB数据库的数据可视化工具，可用于管理数据库内容以及内置了一个代码编辑区域，在数据库代码编辑方面提供了开发优化方案，可提高开发和数据保存效率。

6.2.4 NPM包管理工具

NPM（全称Node Package Manage），是nodejs官方的一种包管理工具，一般是随同nodejs一起安装的，常用于从NPM服务器下载第三方包到本地使用、从NPM服务器下载并安装别人已经编写好的命令行程序到本地使用，能够方便得让前端开发人员下载、安装、删除以及管理已经安装的包。

6.3 开发相关技术

6.3.1 HTML语言

HTML（Hyper Text Markup Language）超文本标记语言，它不是一种编程语言，而是一种标识性语言，是一门由一些标签组成的用来制作网页的语言。HTML语言是可以跨平台的，且支持不同数据格式的文件嵌入。

6.3.2 JavaScript语言

JacaScript是一种解释性脚本语言，它可以直接嵌入HTML页面，在绝大多数浏览器的支持下，可以在windows、mac、linux系统上运行，主要用来向HTML页面添加交互行为，也可以实现web页面的人机交互。

6.3.3 vue框架

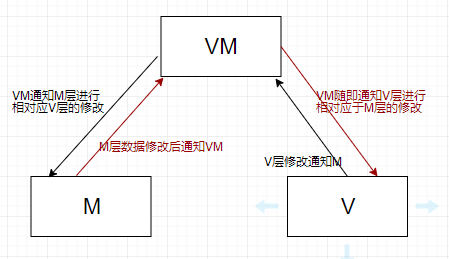
vue是一个用JavaScript写的库，是一个MVVM渐进式框架，MVVM是vue的设计模式，在vue框架中数据会自动驱动视图。

6.3.3.1 MVVM设计模式

view是视图，就是DOM；对应视图也就是html部分—代表UI，它负责将数据模型转换为UI在界面展示出来；model是模型，就是vue页面里的data—代表数据模型，也可以在模型里定义数据修改和操作的业务逻辑；viewModel主要用于监听模型数据，也就是data的改变和控制视图行为、处理用户交互，简单理解就是一个同步view和model的对象。

在MVVM模式下，view层和model层之间并没有直接的联系，而是通过viewModel作为桥梁进行交互，view层和model层之间的交互是双向的，所以，view层的数据变化会同步到model层，model层的数据变化也会同步到view层。

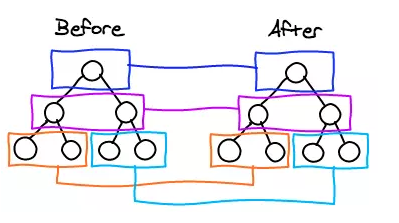
viewModel通过双向数据绑定把view层和model层连接了起来，而view层和model层之间的同步工作完全是自动的，所以，开发中开发者只需要关注业务逻辑而不需要手动的去操作dom节点，也不需要关注数据状态的同步问题，复杂的数据状态维护完全由MVVM模式来统一管理。



6.3.3.2 虚拟DOM（Virtual DOM）和diff算法

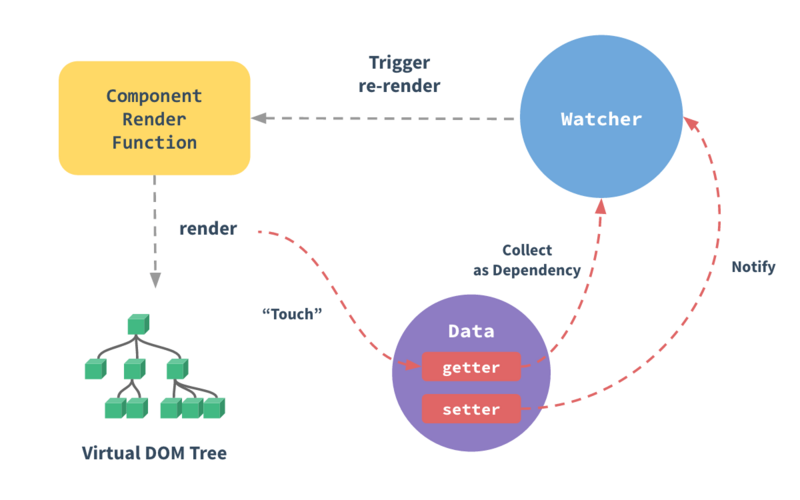
大量的DOM操作是很耗时的，vue框架是数据驱动视图的，用虚拟DOM解决节点操作的时效性。虚拟DOM是用js模拟的结构类似于对象的节点，是可以保存当前视图中所有DOM节点的属性和节点间的关系的树结构。

diff算法是进行新旧虚拟节点元素的对比并返回一个用来存储两个节点不同的地方的patchs对象，最后再用patchs记录消息去局部的更新视图层的DOM。新旧两个树节点完全比较的话需要的时间复杂度为，考虑到效率问题，在diff算法中只会对同级元素进行比较，其时间复杂度仅为，其实就是深层遍历并比较本层级的节点。diff算法的本质就是判断两个虚拟DOM的差异，并将差异更新到真实DOM中。



6.3.3.3 响应式原理

响应式，即数据发生改变时视图会重新渲染，匹配更新为最新的值。vue的响应式原理的核心是通过object.defindeProperty中的访问器属性中的getter和setter方法来拦截数据变化。每个vue实例都会拥有一个专属的观察者watcher，可用于实例更新。vue在视图层展示UI时，通过getter方法获取逻辑层的数据，所以当逻辑层的数据发生改变时，可通过getter方法通知视图层做视图更新；当逻辑层的数据改变时，vue通过属性的setter方法拦截数据变化，setter方法触发当前组件重新渲染，实现视图更新。



6.3.4 nodeJS

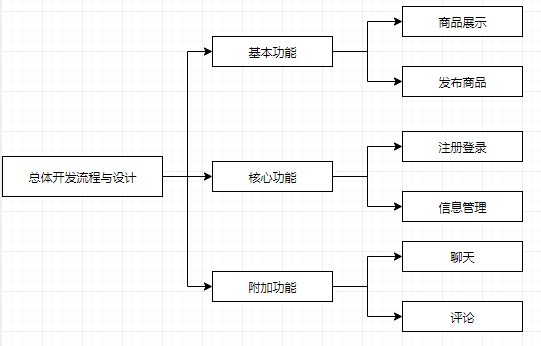
JavaScript是一门脚本语言，它需要一个运行环境。在nodeJS之前，JavaScript的运行环境是浏览器，也就是JavaScript在网页中才能跑起来。nodeJS的出现使JavaScript多了一个运行环境。nodeJS封装了一些i/o的线程池管理和网络的i/o操作，可以控制系统文件的读写和监听网络的输入输出，所以nodeJS又可被单纯的认为是一个可以运行JavaScript的服务器。

nodeJS大多被用于前端开发的开发环境，vue-cli也是运行在nodeJS环境下的脚手架，因为有了文件读写的能力，所以vue-cli可以帮我们打包vue核心模块，将代码转义、压缩，nodeJS可用来做服务器、移动端混合应用、移动端原生应用。

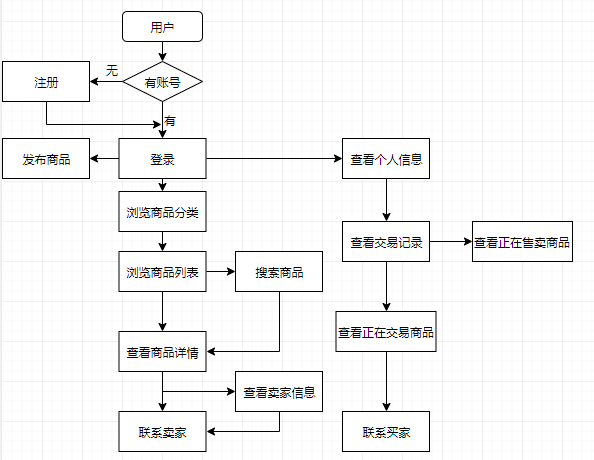
6.3.5 mongodb数据库操作

5 app布局设计与功能实现

5.1 总体开发流程与设计



5.2 app业务流程



5.3 app各页面设计、功能介绍和实现

5.3.1 首页

5.3.2 分类页

5.3.3 商品列表页

5.3.4 商品详情页

5.3.5 卖闲置页

5.3.6 收藏页

5.3.7 个人中心页

5.3.8 登录注册页