

寻宝猫软件需求规格说明(SRS)

目录:

1 范围	4
1.1 标识.....	4
1.2 系统概述	4
1.3 文档概述	4
1.4 基线.....	4
2 引用文件	4
3 需求	4
3.1 所需的 状态和方式	4
3.2 需求概述	5
3.2.1 目标.....	5
3.2.2 运行环境	6
3.2.3 用户的特点	6
3.2.4 关键点	6
3.2.5 约束条件	6
3.3 需求规格	6
3.3.1 软件系统总体功能/对象结构	6
3.3.2 软件子系统功能/对象结构	9
3.4 CSCI 能力需求	12
3.4.1 数据容量	12
3.4.2 并发需求	13
3.4.3 图形界面	13
3.5 CSCI 外部接口需求	13
3.5.1 用户接口	13
3.5.2 硬件接口	14
3.5.3 软件接口	14
3.5.4 通信接口协议	14
3.5.5 接口标识和接口图	14
3.6 CSCI 内部接口需求	15
3.7 CSCI 内部数据需求	15
3.8 安全性需求	15
3.9 计算机资源需求	16
3.9.1 硬件需求	16
3.9.2 软件需求	16
3.10 数据	16
3.10.1 输入数据	16
3.10.2 输出数据	17
3.11 操作	20
3.12 故障处理	21
3.13 需求的优先次序和关键程度	21
4 合格性规定	21
5 需求可追踪性	21
6 风险评估	22
6.1 需求风险	22
6.2 技术风险	22

6.3 过程风险	22
6.4 设计和实现风险	22
6.5 产品风险	23
6.6 回避风险方式	23
7 尚未解决的问题	23
8 注解	23

1 范围

1.1 标识

“寻宝猫”以 APP 的形式呈现，主要标识为以“山大红”标记的“方正圆体”“寻宝猫”字样，以及怀抱元宝、右手半举的招财猫 logo。

1.2 系统概述

“寻宝猫”是一款基于"Android", "IOS"多平台的移动端 APP，是一款旨在为在校同学提供二手交易平台的公益性产品。本项目在学校的监管下，由在校学生自主策划、自主设计、自主开发、自主维护，具有较高的独立性和较专一的使用场景和范围。

1.3 文档概述

本文档的读者是“寻宝猫”开发团队成员，包括项目经理、项目组成员、测试组成员、质量保证组成员。本系统的使用将极大提高工作效率，简化手工作业流程，降低手工工作量和错误率。

1.4 基线

本说明书的主要焦点在于确定产品的准确定位，服务范围，以及开发需求，包括：运行环境，用户特点分析，约束条件分析。同时还需要具体落实 CSCI 能力需求、CSCI 外部接口需求、安全性需求、计算机资源需求等。最后还将确定需求可追踪性以及尚未解决的问题及未来展望。

2 引用文件

本文档参考的所有文档的编号、标题、修订版本和发行日期陈列如下：
《软件设计文档国家标准》GB8567-2006

3 需求

3.1 所需的状态和方式

1. 空闲：如果当前访问人数较少，系统进入低能耗状态以便于节约能源，同时，一般在此时进行系统的维护和更新。
2. 准备就绪：在准备就绪阶段，系统准备好足够的网络资源和软件资源，来应对用户对系

统的频繁访问。

3. 活动：优化活动发布方式，及时更新最新活动

4. 事后分析：在迭代阶段，根据系统的运行情况，获得各种运行信息，以此来进行系统的优化和调整。

5. 培训：对寻宝猫项目组成员进行项目维护以及信息发布方式的培训。

6. 降级：对于部分配置较低的手机，进行系统降级，牺牲部分特性，进行系统精简，保证系统可以正常运行。

7. 紧急情况：当系统更新出现较大 BUG 等紧急情况，发布公告，提供之前的无 BUG 版本下载方法，并且进行紧急维修。

8. 后备：采用周期性记录当前数据的方式进行后备技术，当原始系统数据万一丢失时启用副本的建立和启动。

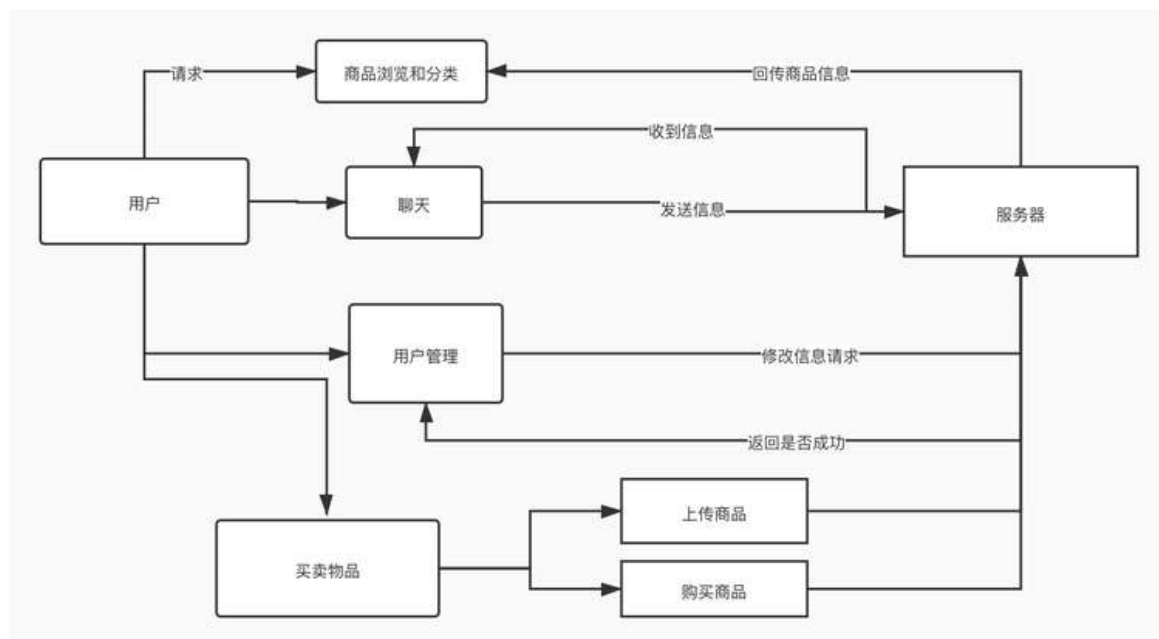
3.2 需求概述

3.2.1 目标

1. 开发目的：本系统打算设计实现一个校园二手交易平台，平台基于 ReactNative 以及 Nestjs 实现，在设计过程中，针对系统特点，兼顾卖家和买家的功能需求，对交易流程进行简化，保证交易的合理以及便捷。

2. 系统主要功能以及数据流程：功能主要包括用户登陆，个人中心，商品发布，商品搜索，聊天通信，订单查询等功能。

3. 外部接口和数据流的系统高层次图：



3.2.2 运行环境

产品在移动手机平台进行运行，因此需要 Android 以及 IOS 系统。

3.2.3 用户的特点

主要用户是学校的学生以及教师，这类人员有二手物品交易的需求，并且在校内方便进行交易。

3.2.4 关键点

1. 关键功能：系统关键功能在于对用户买卖商品的推荐以及管理。
2. 关键算法：为了处理服务器响应速度，采用了高效的数据管理算法，对后台数据库进行优化。
3. 关键技术：React-Native, Nestjs, Redux, MVVM, 数据库维护，网络技术

3.2.5 约束条件

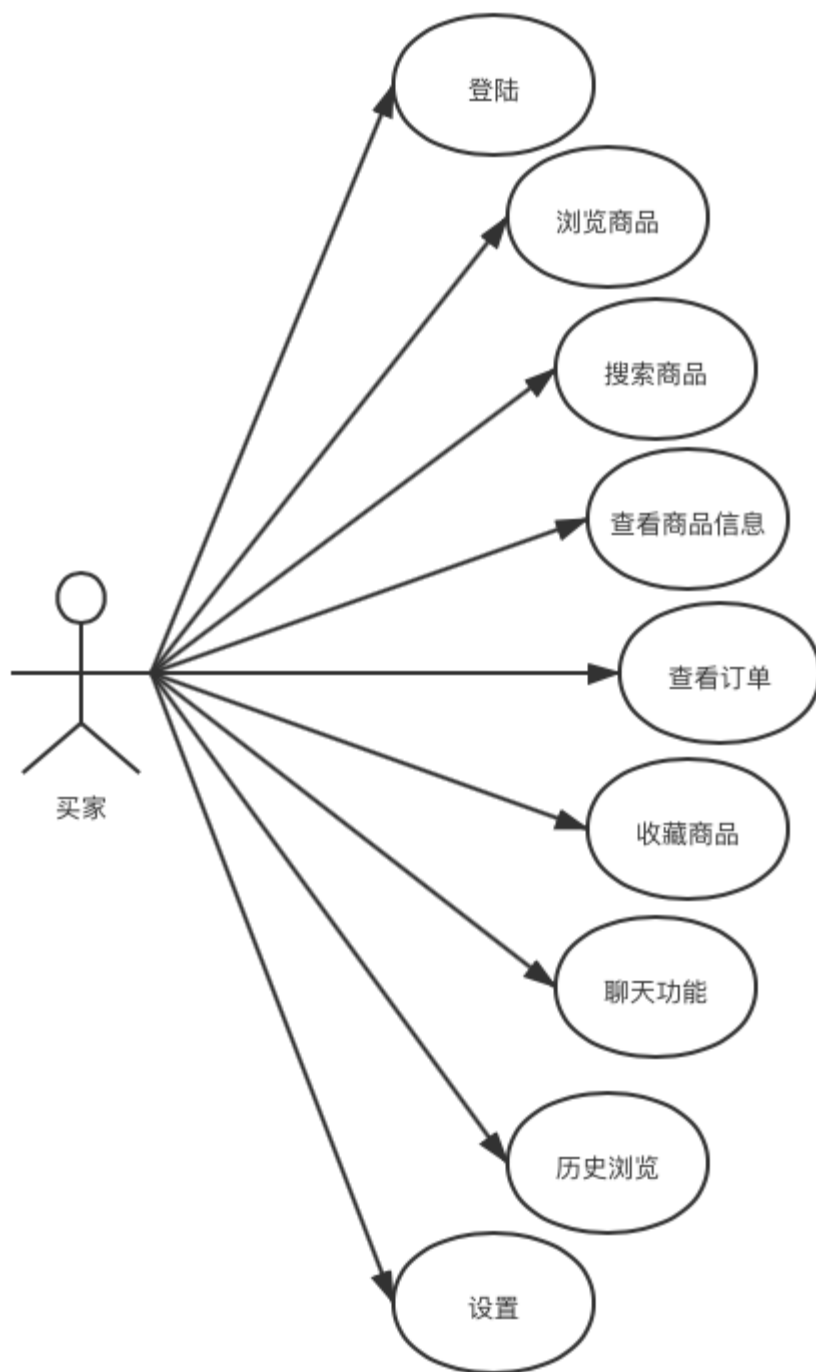
1. 经费限制：由于软件开发成员为在校学生，因此没有经费支持。
2. 开发期限：在本学期内完成。
3. 法律法规：开发过程中需要遵守《中华人民共和国著作权法》、《软件产品管理办法》等规定

3.3 需求规格

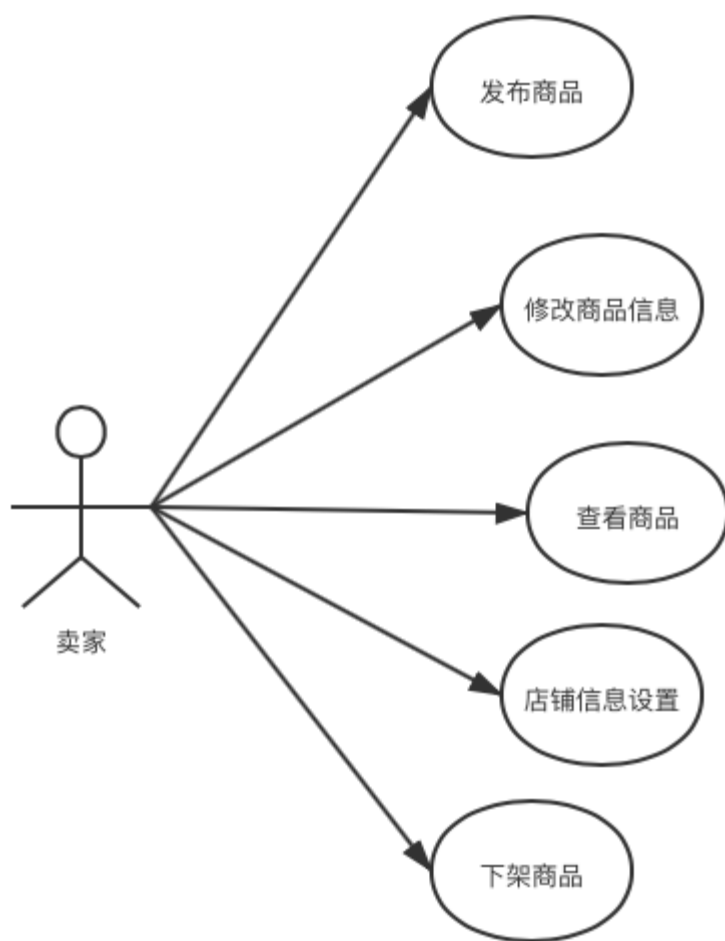
3.3.1 软件系统总体功能/对象结构

系统总体功能包括用户登陆，个人中心，商品发布，商品搜索，订单查询，聊天交流等功能。我们从用户角色来进行功能的分类：

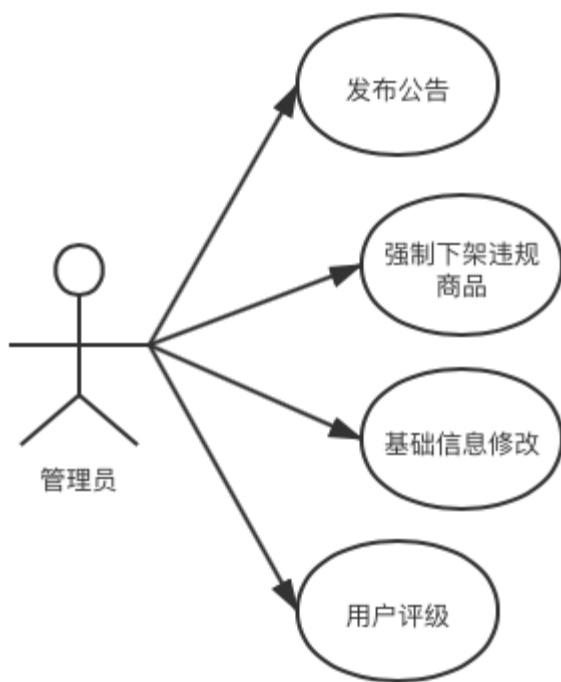
对于买家来说：



对于卖家，除了卖家具有的功能，还有以下功能：



对于管理员，有以下功能：

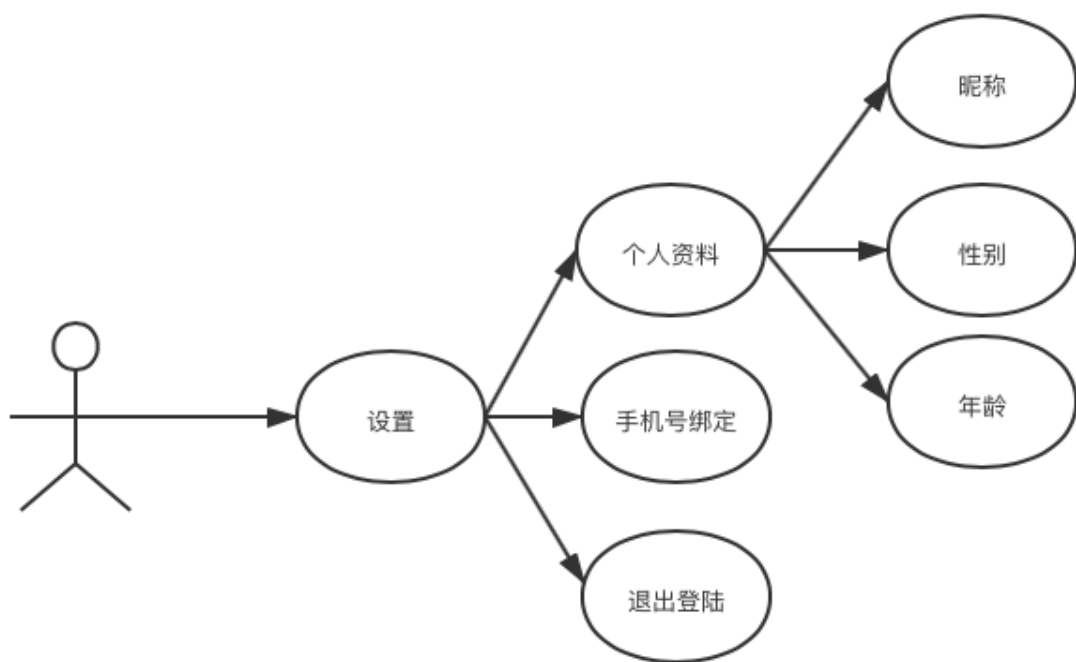


3.3.2 软件子系统功能/对象结构

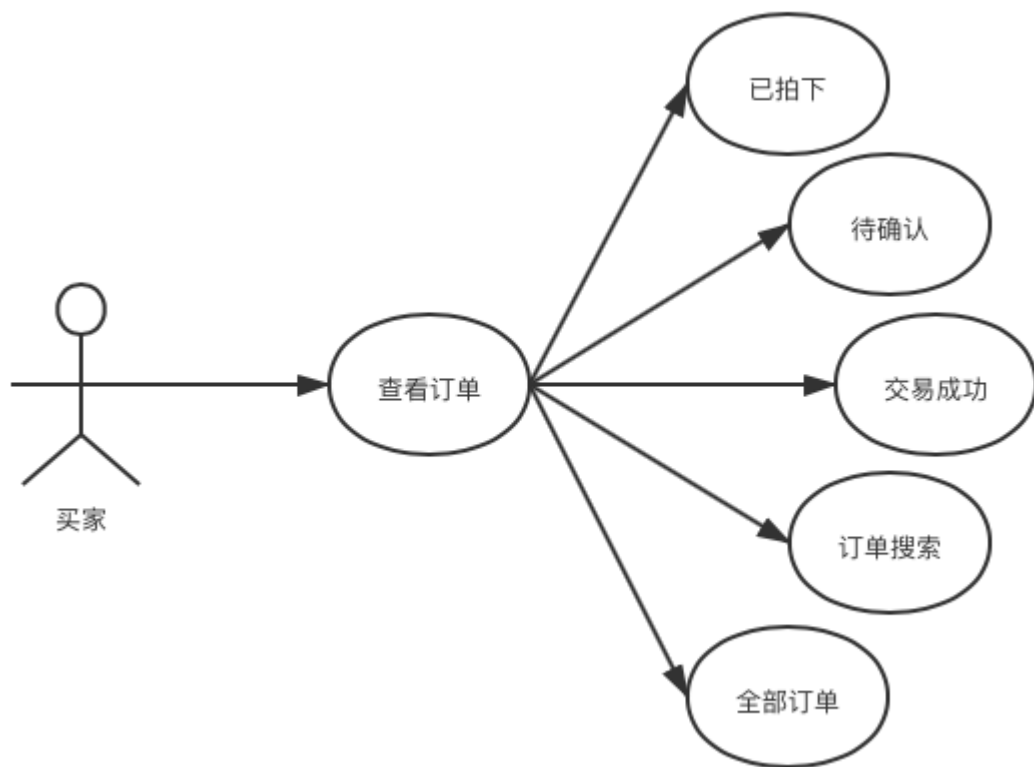
登陆:



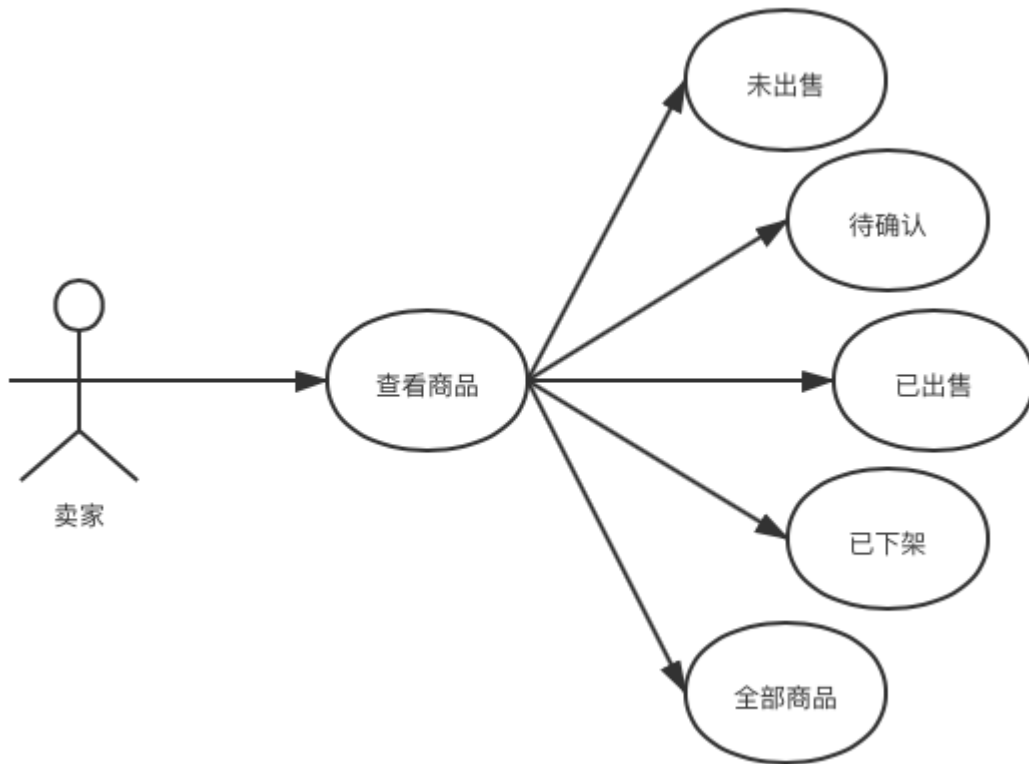
设置:



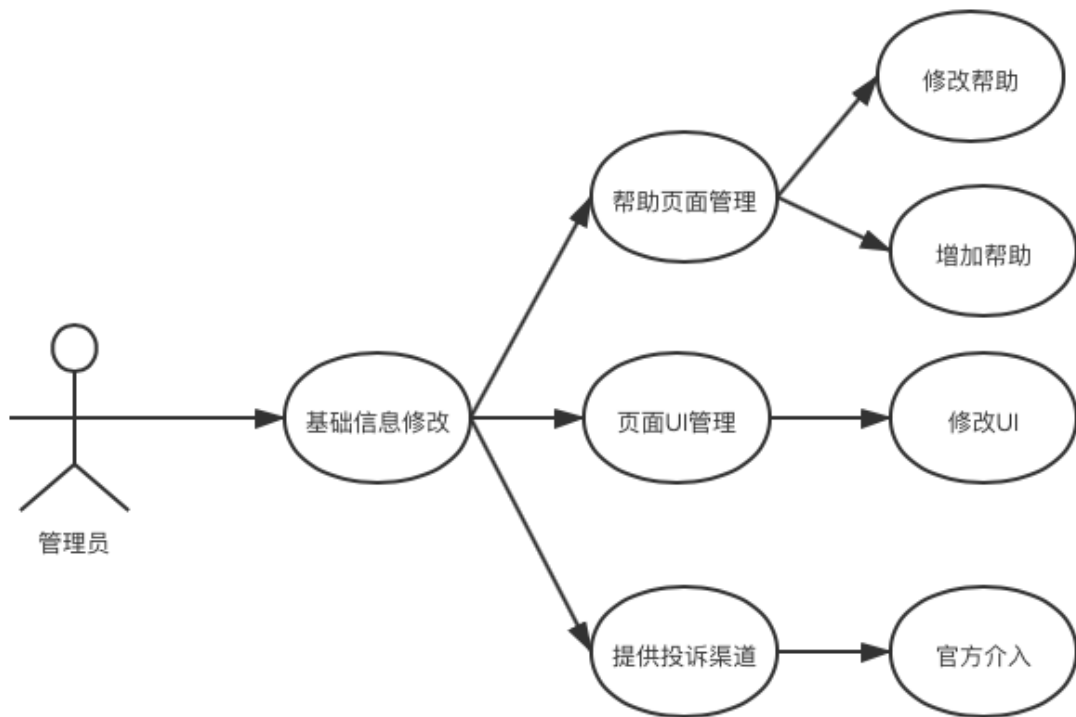
查看订单:



查看商品：



基础信息修改：



3.4CSCI 能力需求

3.4.1 数据容量

(1) 说明

数据容量是指存储论坛中用户信息、商品信息等数据的总的容量；数据容量与开发的规模息息相关,在论坛不断扩大的过程中,必须保证数据容量足够大,以容纳 用户在日常使用时产生的数据。

(2) 输入

数据的输入源主要为用户注册时产生的用户信息，用户发布的商品信息以及用户的交流信息，在前期数据规模可以 MB 计，当用户数量变大时，数据的规模可能会以 GB 计；对于数据的输入，要检测用户在注册/登录时是否遵循预先设定的规则，并且要检测用户的商品内容是否有不符合规范的信息，以保证数据的整洁性和规范性。

(3) 处理

对数据进行有效性检查后，对于有效的数据应当通过数据库的接口将数据存储到数据库中，以保证系统的正常运行，对于不符合规范的数据输入操作，应显式的提示用户，并且在特定情况下禁止严重违反规范的 IP 地址登录。

(4) 输出

用户在调用某些数据时，对于具有合法权限用户的请求，应当同意用户的操作，并且将指定数据展现给用户(主要以 json 数据回传并解析成数据显示在 APP 内的形式)，同时，管理员应具有查看并管理数 据库的权限。

3.4.2 并发需求

(1) 说明

对于并发的需求是一个工程的基本需求，特别是对于类似 BBS 的交流软件而言。因此，系统需要提供足够的并发度，以保证用户的正常使用。

(2) 输入

可以将用户的每一个登录状态看作一个输入，我们需要考虑到很多输入的情况。考虑到时段问题，我们可以在用户活跃的峰值区间增加资源，以提高并发，并且在低谷阶段减少资源，以降低成本。

(3) 处理

为了解决高并发问题，我们可以提高接口响应时间，把平均响应时间做到几十毫秒甚至几毫秒，这个方法需要一定的技术支持，并且提升空间不大，但是几乎没有成本的增加，由于我们的 APP 的初期预估人数在千人规模，因此可以采用负载均衡(load balance)+web 服务器（如 Wamp Server）的配置即可实现

(4) 输出

管理员应具有查看系统当前并发程度的权限，如服务器 CPU 的利用率、系统当前数据吞吐量等，在一些高峰阶段定时查看，并在必要时请求技术人员控制系统的并发能力。

3.4.3 图形界面

(1) 说明

当今软件,用户交互是很重要的一部分,并且直接决定的软件的存亡。因此,本 CSCI 能力主要是提供一个图形界面,主要采用的是 React-Native 技术。匹配此 CSCI 需要支持 APP 的操作系统, IOS14 与 Android 8.0 及以上。

(2) 输入

输入源是用户的 APP 操作，具体的是，对于用户在 APP 中的操作，软件可以相应的做出响应与用户交互，本质上 APP 要有处理请求的能力，同时对于用户的信息请求，APP 需要传给服务器请求，并收取相应的结果。数量方面，一般来说，请求不会超过一秒百万.度量单位，每一个用户作为一个单位考虑输入。

有效输入范围：有效的 APP 页面与交互。所以，从输入的方面考虑，匹配此 CSCI 能力至少需要 安装了满足要求版本的移动设备，以及需要搭载着相应的输入设备。

(3) 处理

本 CSCI 的能力的数据计算需要在客户端的处理器与服务端进行，在客户端如果数据处理或者网络状态发生异常，需要给用户相应提示信息供用户使用；在服务端，如果数据库与服务器网络发生异常，应有相关的报错提示。

(4) 输出

软件的运行者作为数据的接收端，必须要对数据进行处理并显示在 APP 上，因此需要匹配相应的具有图像处理能力和显示设备的移动设备。

3.5CSCI 外部接口需求

3.5.1 用户接口

用户接口包括用户调用微信、qq 进行转发的接口，用户通过点击转发按钮即可调用这一外

部接口。

3.5.2 硬件接口

处理器：多核处理器
内存：2G 及以上
外存：16G 及以上
GPU：能够处理普通动画即可
移动网络：4G 或 5G 均可
显示器：手机屏幕
传感器：无要求
电池：有内置电池
摄像头：作为拍照的设备

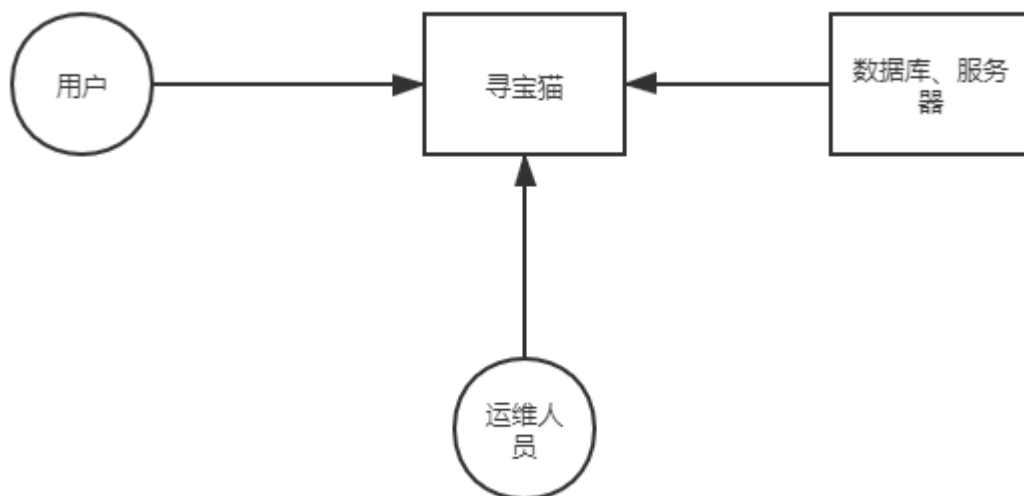
3.5.3 软件接口

系统：Android 或 iOS

3.5.4 通信接口协议

提供 TCP/IP 协议。

3.5.5 接口标识和接口图



3. 6CSCI 内部接口需求

内部接口包括用户登录接口，注册接口，以及用户对自己商品进行操作的一系列接口，这里以登录接口为例：

- 请求路径: login
- 请求方法: post
- 请求参数 login

参数名	参数说明	备注
user_id	用户名	不能为空
password	密码	不能为空

- 响应参数

参数名	参数说明	备注
user_id	用户名	
authority	用户权限	
token	令牌	基于 jwt 的令牌

- 响应数据

```
{
  "data": {
    "user_id": 500,
    "token": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1aWQiOiJ1aWwMcWicm1kIjowLCJpYXQiOiE1MTI1NDQyOTksImV4cCI6MTUxMjYzMDY5OTk0fQ."
  },
  "meta": {
    "msg": "登录成功",
    "status": 200
  }
}
```

3. 7CSCI 内部数据需求

关于内部数据的需求，由于在撰写本文档时，整个程序只有架构是确定的，内部数据的具体内容已经确定，但是在各种细节并没有完善，也没有建立起各种数据的数据库和数据之间的联系关系。因此，关于 CSCI 内部数据需求需要留待设计时确定。

3.8 安全性需求

本平台的保密性需求主要包含以下几个方面：

- 1.使用本平台需要在数据库中保存用户的个人信息，其中包含用户的姓名，密码等很多重要的信息；此外，对于用户发布到平台的商品，为了保证交易的双方能够清楚的查看信息，使得交易能够顺利进行，商品的信息在保存时不应该出现信息缺失，错误的情况。因此，本平台需要保证数据库信息的完整性和安全性。
- 2.在使用本平台进行交易的过程中，用户之间会进行聊天，用以确定交易的细节。在聊天时，聊天记录中会出现一些涉及到用户隐私的信息。为了保护用户的隐私安全，本平台需

要保证用户之间聊天记录的私密性，防止泄露。

3.在平台运行的过程中，有非常大的可能会出现数据丢失的情况。为了减少数据丢失带来的影响，需要定期备份数据库。

3.9 计算机资源需求

3.9.1 硬件需求

根据软件的规模，我们预计最低的手机硬件配置如下：

处理器：HiSilicon Kirin 970 以上

运行内存：1GB 以上，建议使用 2GB 内存

手机存储：至少要有 1GB 存储空间

3.9.2 软件需求

本软件运行所需要的软件需求如下：

操作系统：Android 8.1.0 以上

数据库管理软件：

开发软件：Visual Studio Code

网络通信协议：TCP/IP 协议，FTP 协议

3.10 数据

3.10.1 输入数据

在输入数据方面，本平台在以下功能需要输入数据：

1.用户登录：

输入数据的定义及数据类型为：

数据名称	数据类型	数据大小
学号	字符串	12B
密码	字符串	6-16B

输入数据总计 18B-28B，每次需要处理 1 条这样的表单数据。

2.搜索商品：

输入数据的定义及数据类型为：

数据名称	数据类型	数据大小
商品名称	字符串	1-100B

输入数据总计 1B-100B，每次需要处理 1 条这样的表单数据。

3.发布商品：

输入数据的定义及数据类型为：

数据名称	数据类型	数据大小
商品图片	图片	约 2MB
商品名称	字符串	1-100B
商品价格	浮点数	4B
商品类别	字符串	2-20B
新旧程度	字符串	5B
商品库存	整数	4B
商品描述	字符串	1-1000B

输入数据总计约 2MB,每次需要处理一条这样的表单数据。

4.聊天：

输入数据的定义及数据类型为：

数据名称	数据类型	数据大小
聊天内容	字符串	1-1000B

输入数据总计 1B-1000B，每次需要处理 1 条这样的数据。

5.修改个人信息：

输入数据的定义及数据类型为：

数据名称	数据类型	数据大小
用户头像	图片	2MB
昵称	字符串	1-20B
性别	字符串	2B
手机号	字符串	11B

输入数据总计约 2MB，每次需要处理 1 条这样的表单数据。

3.10.2 输出数据

在输出数据方面，本平台在以下功能需要输入数据：

1.显示搜索结果:

输出数据的定义及数据类型为:

数据名称	数据类型	数据大小
商品名称	字符串	1-100B
商品价格	浮点数	4B
查看次数	整数	4B

输出数据总计 9B-108B，每次需要输入所有满足搜索结果的数据。

2.显示用户发布的所有商品:

输出数据的定义及数据类型为:

数据名称	数据类型	数据大小
商品图片（已压缩）	图片	约 32K
商品名称	字符串	1-100B
商品价格	浮点数	4B
查看次数	整数	4B

输出数据总计 41B-140B，每次需要输入所有满足搜索结果的数据。

3.显示用户收藏的所有商品:

输出数据的定义及数据类型为:

数据名称	数据类型	数据大小
用户名	字符串	1-20B
商品图片（已压缩）	图片	约 32K
商品名称	字符串	1-100B
商品价格	浮点数	4B

输出数据总计 38B-156B，每次需要输入所有满足搜索结果的数据。

4.显示用户浏览商品历史:

输出数据的定义及数据类型为:

数据名称	数据类型	数据大小
商品图片（已压缩）	图片	约 32K
商品名称	字符串	1-100B

商品价格	浮点数	4B
------	-----	----

输出数据总计 37B-136B，每次需要输入所有满足搜索结果的数据。

5.显示用户订单：

输出数据的定义及数据类型为：

数据名称	数据类型	数据大小
用户名	字符串	1-20B
商品图片（已压缩）	图片	约 32K
商品名称	字符串	1-100B
商品价格	浮点数	4B
状态	字符串	6-8B

输出数据总计 44B-164B，每次需要输入所有满足搜索结果的数据。

6.显示用户已发布商品：

输出数据的定义及数据类型为：

数据名称	数据类型	数据大小
商品 ID	字符串	10B
商品图片（已压缩）	图片	约 32K
商品名称	字符串	1-100B
商品价格	浮点数	4B
状态	字符串	6-8B

输出数据总计 53B-154B，每次需要输入所有满足搜索结果的数据。

7.显示商品详细信息：

输出数据的定义及数据类型为：

数据名称	数据类型	数据大小
商品 ID	字符串	10B
商品图片	图片	约 2MB
商品名称	字符串	1-100B
商品价格	浮点数	4B
商品类别	字符串	2-20B

新旧程度	字符串	5B
商品库存	整数	4B
商品描述	字符串	1-1000B

输出数据总计约 2MB,每次需要处理一条这样的数据。

8.显示聊天用户信息:

输出数据的定义及数据类型为:

数据名称	数据类型	数据大小
用户名	字符串	1-20B
用户头像	图片	约 2MB
最近消息	字符串	1-20B

输出数据总计约 2MB,每次需要处理所有满足条件的数据。

9.显示聊天记录:

输出数据的定义及数据类型为:

数据名称	数据类型	数据大小
聊天内容	字符串	1-1000B

输出数据总计约 1B-1000B,每次需要处理所有满足条件的数据。

3.11 操作

对于本系统而言,所包含的操作大概可以分为三类:常规操作,特殊操作和恢复操作。

1.对于常规操作,即本软件需要经常性处理的操作的合集,包括:查询满足条件的结果,显示商品的详细信息,发布/撤回/购买商品,发送/查看/撤回聊天信息,查看/修改个人信息等。这些操作使用非常频繁,因此,要求处理这些操作的速度尽可能的快,由于不同手机配置的不同,相应的操作时间上限会有所不同,所以对于这一类操作,完成一条操作所需要的平均时间不能超过 5 秒。

2.对于特殊操作,即本软件很少需要处理的操作,包括:登陆,注册,找回密码等。这些操作很少被用户使用,而且相应的操作过程也比较的繁琐,因此,要求对处理这些操作的速度要求不高,根据不同手机配置的不同,对于这一类操作,完成一条操作所需要的平均时间不能超过 10 秒。

3.对于恢复操作,即本软件在出现运行错误和故障时所采取的补救操作。这类操作的使用频率最低,并且类操作的目的是需要保证系统的正确性,因此操作过程最为繁琐。所以,对于这类操作来说,速度要求最低,根据不同手机配置的不同,对于这一类操作,完成一条操作所需要的平均时间不能超过 20 秒。

3.12 故障处理

本平台运行时，可能出现的故障如下：

1.

错误类型：由于网络原因，导致数据上传失败。

错误信息：弹出提示框，内容为：“网络连接异常，上传数据失败，请重试。”

补救措施：无。

2.

错误类型：由于网络原因，导致数据下载失败。

错误信息：弹出提示框，内容为：“网络连接异常，请刷新。”

补救措施：程序不断尝试从服务器下载数据。

3.

错误类型：由于系统问题，导致运行闪退。

错误信息：无。

补救措施：回退到闪退之前的状态。

3.13 需求的优先次序和关键程度

本软件的优先次序如下：

1.本软件的主体是一个二手交易软件，因此，交易功能的实现对于整个系统是至关重要的，而交易功能中最重要的功能是搜索商品，查看商品和聊天。同时，这几个功能对于安全性，保密性的要求也是最高的。因此，这三个功能的优先级最高。

2.用户使用本平台需要登陆，而对于交易的管理也是必要的，因此，用户登陆和订单显示这两个功能的优先级中等，应在高优先级功能实现后紧接着实现。

3.其他功能对于整个软件的主体来说不算很重要，优先级最低。

4 合格性规定

“寻宝猫”APP 验收标准为：实现所有功能需求、满足非功能性需求、系统设计文档完整，且符合规范

、代码符合规范，且与系统设计一致。

5 需求可追踪性

在软件开发过程中建立和维护软件制品之间的关联关系，并利用这些关系对软件项目进行一系列分析。包括建立 向前需求规范和向后需求规范、前向和后向可追踪性、水平和垂直可追踪性。

(1) 向前需求规范和向后需求规范 。既包括从需求规范到用户需求的追踪，也包括是从需求规范到实现需求的软件设计、源代码间的追踪。

(2) 前向和后向可追踪性。包括从当前软件制品到其后继派生制品的追踪，如从需求到实现该需求的源代码之间的追踪，以及从当前制品到派生出该制品的源制品的追踪，如从源代码中某功能函数到 UML 类中某操作的追踪。

(3)水平和垂直可追踪性。包括对处于相同的项目开发阶段或拥有相同抽象层次的制品间的追踪，例如与系统性能相关的所有需求间的追踪，特定需求在不同时刻不同版本间的追踪，及对不同阶段拥有不同抽象层次制品间的追踪，如软件需求和设计制品间的追踪。

6 风险评估

6.1 需求风险

1. 需求已经成为项目基准，但需求还在继续变化；
2. 需求的优先程度规划问题；
3. 需求定义欠佳，而进一步的定义会扩展项目范畴；
4. 产品定义含混的部分比预期需要更多的时间；
5. 缺少有效的需求变化管理过程。

6.2 技术风险

1. 缺乏培训
2. 对方法、工具和技术理解的不够
3. 应用领域的经验不足
4. 对新的技术和开发方法应用不熟悉

6.3 过程风险

1. 大量的纸面工作导致进程比预期的慢；
2. 前期的质量保证行为不真实，导致后期的重复工作；
3. 太不正规(缺乏对软件开发策略和标准的遵循)，导致沟通不足，质量欠佳，甚至需重新开发；
4. 过于正规(教条地坚持软件开发策略和标准)，导致过多耗时于无用的工作；
5. 向管理层撰写进程报告占用开发人员的时间比预期的多；
6. 风险管理粗心，导致未能发现重大的项目风险。

6.4 设计和实现风险

1. 设计质量低下，导致重复设计；
2. 一些必要的功能无法使用现有的代码和库实现，开发人员必须使用新的库或者自行开发新的功能；
3. 代码和库质量低下，导致需要进行额外的测试，修正错误，或重新制作；
4. 过高估计了增强型工具对计划进度的节省量；
5. 分别开发的模块无法有效集成，需要重新设计或制作。

6.5 产品风险

1. 矫正质量低下的不可接受的产品，需要比预期更多的测试、设计和实现工作；
2. 开发额外的不需要的功能(镀金)，延长了计划进度；
3. 严格要求与现有系统兼容，需要进行比预期更多的测试、设计和实现工作；
4. 要求与其他系统或不受本项目组控制的系统相连，导致无法预料的设计、实现和测试工作；
5. 在不熟悉或未经检验的软件和硬件环境中运行所产生的未预料到的问题；=
6. 开发一种全新的模块将比预期花费更长的时间；
7. 依赖正在开发中的技术将延长计划进度。

6.6 回避风险方式

1. 需求变更需要经过统一的负责人提出,并且要用户需求的审核领导认可,需求变更应该是定期而不是随时的提出；
2. 控制系统的复杂程度，过于简单的系统结构，对用户来使用比例会有明显的折扣，甚至造成软件寿命过短。反之，软件结构的过于灵活和通用，必然引起软件实现的难度增加，系统的复杂度会上升，这又会在实现和测试阶段带来风险。适当控制系统的复杂程度有利于降低开发的风险；
3. 不断对软件系统进行版本升级，在确保可维护性的前提下逐步扩展系统；
4. 设定应急计划，每个开发计划都至少应该设定一个应急预案去应对出现突发情况和不可预知的风险。
5. 借助相关的风险管理工具。如 GS risk，Deltak Risk+等。其中 RPM 通过为风险和响应集中编目，并生成措施或任务来识别与记录项目有关的问题并做出相应，从而能够提供从计划，监控到控制阶段过程有效的风险管理，同时其强大的风险数据库还可以为项目风险提供模板。

7 尚未解决的问题

在软件逻辑设计方面还存在以下几个待优化问题：

对于买家，想要查看需要的商品十分麻烦，需要通过查看以往的聊天记录进行搜索，同时过长时间的消息无法确定商品是否还在，需要一一询问。

对于卖家，发售的商品过一段时间或许会被其他商品或者其他的水群记录给“刷屏”，并且部分以图片形式介绍的商品极其容易被人所忽略，因此出售物品也较为困难。

除此之外，交易群聊中人员有限，且群聊容纳人数有限，这就导致需求双方人数较少，因此交易也比较困难。

8 注解

API：一组预先定义的接口，程序可以通过这个接口访问一组例程，但不需要了解实现该进程的源码。

React-Native: 前端框架，使用 JS 语言支撑原生 App 的开发。

Redux: 前端数据层，使用 JS 语言实现数据在 App 中的流动。

Nestjs: 后端开发框架，支持服务器数据的管理并为前端开发提供 HTTP 访问方法和接口。

MVVM: 一种软件架构模式，有助于将图形用户界面的开发与业务逻辑或后端逻辑的开发分离。

Web 开发: 指开发出的软件是由一个网页呈现的。