# 第3章.需求分析

## 3.1需求分析的目的

需求分析就是为了满足所面向用户的要求的过程。通过客户所描述的需求来确定系统所要实现的功能，为之后的总体设计详细设计打下基础，避免出现编写系统用户不满意的情况。

只有只要用户需要什么，才能做出让用户所期望的产品。这是后期软件开发的基础所在。如果不能满足用户需求，无论系统的设计与编码做得如何出色，制作出来的产品还是失败的，开发者费时费力，用户还很失望。

本章节是根据当前市场上的小说阅读应用与小说阅读用户进行充分调研之后整理所得。描述小说推荐、小说搜索和小说阅读以及用户需求的文档。

## 3.2 确定软件的具体要求

本软件的具体要求主要包括系统的功能需求、系统的性能需求、系统开发环境和运行环境的需求以及系统可靠性和可用性需求。本章节的目的一方面是展示用户的业务需求，另一方面是作为后续开发设计业务需求的指导文档。

### 3.2.1 系统的功能需求

本项目是基于微信小程序的小说阅读系统，是一个用户用户阅读网络小说的平台。因此本软件应该满足用户阅读网络小说的一些操作。从整体上考虑，本系统应该实现以下功能：

1. 登录系统功能：为了获取用户信息，分析了解用户喜好，提供更好的阅读体验。设置一个登录功能，登录时候获取微信所能提供的用户信息，为之后提升用户体验打下基础。
2. 小说排行榜功能：为了用户可以找到自己喜欢的小说。根据点击、评论等要素对小说进行的排行对用户找到自己喜欢的小说有很大帮助。
3. 小说分类功能：为了用户可以找到自己喜欢的小说。根据小说的类别进行查找可以让用户更加容易寻找到自己喜欢的小说。
4. 小说书架功能：用户可以继续阅读未阅读完的小说，此功能是小说阅读的基本所在。
5. 小说阅读功能：此功能是本系统的核心功能，包括章节列表查询、章节内容阅读、章节跳转等。

### 3.2.2 系统的性能需求

1. 准确性。在软件设计和开发过程中，不但要保证本次数据处理结果的准确性，还要考虑本次数据处理结果对整个系统的影响，从而保证整个系统的正确性。
2. 稳定性。软件的每次运行虽然时间上不同，处理器可能处于不同的状态，但也应该保持系统的反应相对稳定，不能出现两次相同或类似的操作、数据输入下系统响应时间和响应效果差异明显。
3. 及时性。及时性就是要求软件在日常处理中的响应速度要达到实时要求，以及实时反馈信息。在软件设计和开发过程中，还要充分考虑系统当前和将来可能承受的工作量，使系统处理能力和响应时间能够满足用户对信息处理的要求。
4. 实用性。软件展现出的功能必须可以给用户以舒适的用户体验，用简单快捷的操作完成期望的功能。也就是说，用户能够快速上手熟悉软件，轻易地操纵软件，而不是感觉软件难以应用，无法上手。
5. 可扩展性。软件在开发过程中，应该充分考虑以后的可扩展性。这些要求软件提供足够手段进行功能的调整和扩充。要实现这一点，应通过系统的开放性来完成，即系统应是一个开放的系统，只要符合一定的规范，可以简单的加入和减少系统的模块、配置系统的硬件。通过软件的修补、替换完成系统的升级和更新换代。可以说，具有良好的扩展性的软件才是优秀的软件，这是一个软件应具有的基本要求。
6. 健壮性。用户在使用软件过程中会出现各种各样的情况，误操作也是避免不了的。面对不同的操作带来的不同情况，软件应具备一定的适应能力和容错能力，对于用户的误操作可以进行提示，而不是将错就错进行执行。

### 3.2.3 系统开发环境和运行环境的需求

系统的开发环境具体配置如下：

1. 硬件环境：

* CPU：Intel(R) Core(TM) i3 2.53GHz；
* 硬盘：320G；
* 内存：8G。

1. 软件环境：

* 操作系统：Centos7.0 ；
* 系统类型 64 位系统；
* 分辨率：推荐使用分辨率为1366×768；
* 数据库：MySQL数据库；
* 开发工具：Sublime Text 3,微信公众开发平台；
* 开发语言：GoLang，JavaScript；
* 建模工具：Microsoft Office Visio 2010。

### 系统可靠性和可用性需求

1. 数据传输可靠性：对前端请求数据采用Json串传输；
2. 数据保存可靠性：对后端数据进行备份，后端数据的定期更新；
3. 数据响应可靠性：对数据库读写分离保证响应的高效；
4. 客户端界面可用性：清爽舒适的界面设计，易用性强；
5. 交互设计可用性：对于各种操作，提供较为明显的响应。始终以用户响应为第一优先，提供良好的操作感

## 3.3可行性分析

随着网络文学的飞速发展，版权意识的完善，阅读系统的竞争越来越激励。很多阅读器从添加广告、内容收费等方面给用户带来了不好的体验，微信小程序的出现无疑提供了一个快捷、有效的服务平台。本节分别从技术、经济和操作的角度分析了基于微信小程序的小说阅读系统的开发和应用的可行性。

### 3.3.1 技术可行性

构建基于微信小程序的小说阅读系统分为前后端两个部分。系统前端是使用微信所提供的API，使用JavaScript脚本以及CSS布局，开发工具使用微信自带的微信公众号开发工具。服务器端采用的是Google发布的开源GoLang语言，数据库使用的是MySQL。经过不断的学习与动手练习，现在已经能够熟练的运用这几项开发技术。还有百度，谷歌的搜索与指导老师提供帮助。因此，开发该系统在技术层面上具有可行性。

### 3.3.2 经济可行性

基于微信小程序的小说阅读系统的成本包括两部分：一部分是硬件设备成本，包括后台服务器、域名和作为测试机的移动终端。随着电子产品的普及，其价格也在不断下降。硬件部分的费用还是可以接受；另一部分来自与开发系统的成本，Go语言有很多优秀的开源框架在成本上可以节省很多。加上程序上架所带来的收益可以弥补开发的成本。所以该系统在技术上具有可行性。

### 3.3.3 操作可行性

在我国微信的普及率十分大。对于网络小说阅读爱好者来说，对于各种App应用操作十分熟练，所以对于以简洁著称的小程序更不在话下。设计本系统最重要就是简洁、免费。让更多的网络文学爱好者可以读到自己喜欢的小说。所以，开发基于微信小程序的小说阅读系统在操作上具有可行性。

## 3.4本章小结

本章主要介绍了基于微信小程序的小说阅读系统的需求分析，主要包括系统的功能需求、系统的性能需求、系统开发环境和运行环境的需求以及系统可靠性与可用性需求。同时在从技术、经济与操作三个方面分析了本系统的可行性。一份详细的需求分析在开发之中起着非常重要的作用，它决定着系统能否正常交付。

# 第4章 总体设计

## 4.1 总体设计概述

经过需求分析对本系统需要“做什么”已经比较清楚了，需求分析的结果作为本系统的需求抽象信息结构，现在进入了该“如何去做”，即该如何去实现本系统的阶段。将需求分析的逻辑模型，变换成总体设计的物理模型，着手去实现系统的需求。

总体设计首先是设计系统的总体功能结构，也就是要确定系统中每个程序是由哪些模块组成，还有这些模块相互之间的关系和相互之间的调用过程。由于模块之间是相互独立的，所以可以在编码的时候单独的设计、编写、测试和修改，从而降低了错误在模块中的蔓延，提高了系统可靠性和可维护性。

其次，是进行数据库的设计，数据库的设计是本系统重要的部分，设计好数据库，才能让系统运行更流畅。在系统的开发初期，必须要完善的规划数据库、设计数据库中的各个数据集、规范数据库间数据的联系。

## 4.2系统总体模块设计

基于微信小程序的小说阅读系统分成微信小程序客户端和Web服务器端两大部分。其中微信小程序客户端分为数据展示模块、数据处理模块。Web服务器端分为请求处理模块、数据爬取模块。各个模块的具体功能如下：

1. 数据展示模块：数据展示模块中包含本小程序的用户提供的各个功能，当用户进入本系统之后可以使用这些功能，包括首页推荐、书架功能、分类推荐和搜索书籍等操作。

* 首页推荐：用户在进入到系统中之后首先看到的就是首页推荐，用户在界面中可以浏览到各种热门小说推荐，用户可以选择感兴趣的小说查看介绍，小说介绍部分可以看到小说封面、最新章节、内容介绍等信息。用户可以选择对该小说进行阅读、查看目录、加入书架等操作。
* 书籍阅读：用户从书架或者首页推荐等入库进入阅读某小说，阅读按照章节顺序进行。在阅读过程中可以查看小说目录、章节跳转、小说字体大小修改、背景颜色修改等操作。
* 书架：通过首页推荐、分类查找、搜索小说等功能可以找到自己感兴趣的小说。为了下次阅读更加方便可以将小说放入书架。对于已经阅读完的小说或者已经不感兴趣的小说可以将其进行移除书架操作。点击书架上的小说可以跳转到上次阅读到的章节界面。
* 分类查看：分类查看提供了更加细致的各个类别的小说列表。按照小说的种类对小说进行细分，满足不同用户的需求。
* 搜索书籍：在搜索页面提供了搜索查看小说的功能。在该页面同时提供热门搜索内容、用户最近搜索记录等功能以方便用户。

1. 数据处理模块:数据请求模块包括小程序端对于数据的操作。包括请求发送、数据接收和本地数据处理等操作。

* 请求发送：在初始加载页面以及用户点击发起的请求，通过微信小程序自带的API像服务器端进行请求。
* 数据接收：将请求回来的Json数据进行处理。存放到本地或者反馈给用户阅读。
* 本地数据处理：为了小程序的流畅程度，选择把书架等部分数据存放到本地。用户的操作感觉更加流畅。

1. 请求处理模块：请求处理模块是根据前台发送过来的不同请求，通过验证其正确性之后进行相应的数据库查询、网页数据爬取等操作。本模块包含推荐信息请求、分类信息请求、搜索书籍请求、书籍内容请求等。

* 推荐信息请求和分类信息请求：接收小程序客户端发送过来的请求与数据并验证，通过查询数据库或爬取网页内容获取相应的数据并返回给小程序客户端。
* 搜索书籍请求：接收小程序客户端发送过来的请求与数据并验证，通过查询数据库获取相应的数据并返回给小程序客户端，如果数据库没有相应信息，通过网页内容爬取来更新数据库同时返回相应数据。
* 书籍内容请求：接收小程序客户端发送过来的请求与数据并验证，通过查询数据库获得书籍所在网址。通过爬虫获取章节列表、章节内容等数据返回至小程序客户端。

1. 数据爬取模块：数据爬取模块是使用爬虫技术查找并爬取网页上的有用数据。通过维护数据库来跟新数据网页的地址。主要包括爬取分类信息、爬取书籍信息、更新基本信息等操作。

* 爬取分类信息：使用爬虫对多个小说网站进行数据爬取。包括各个类别的小说内容等信息。将基本信息以及小说地址写入数据库以减少用户请求等待时间。
* 爬取书籍信息：使用数据库中已有的地址，或者更新之后的小说地址对小说的内容进行爬取。将获取的内容返回至小程序客户端。书籍的信息并不存储在数据库中。
* 更新基本信息：固定一个时间，每隔这个时间对数据库中的数据进行更新。包括书籍的最新章节、地址等等。减少用户请求时才进行更新的次数。

系统总体功能模块图如下所示：

基于微信小程序的小说阅读系统

小程序客户端

Web后端服务器

数据展示模块

数据处理模块

数据爬取模块

请求处理模块

更新信息

爬取分类信息

爬取书籍信息

书籍内容请求

搜索书籍请求

分类信息请求

推荐信息请求

本地数据处理

数据接收

请求发送

搜索

书籍

分类查看

书架

书籍阅读

首页推荐

图4-1 系统功能模块图

## 4.3 核心功能模块设计

### 4.3.1 搜索书籍模块设计

搜索书籍模块是本系统十分重要的一个模块。具体过程为：首先在搜索界面输入要搜索的书籍名称（可以不完整），请求至后端服务器进行数据验证，验证成功之后依据名称查询数据库看是否有数据，如果有返回数据，如果没有使用爬虫爬取网页数据，爬取成功更新数据库返回书籍信息，失败弹出未查找到提示。流程图如图4-2所示。



图4-2 搜索书籍模块流程图

### 4.3.2 数据更新模块设计

数据更新模块是为了及时更新数据库中的数据，使用户得到的数据一直是最新的。具体过程为：首先在配置上设置更新的时间间隔。设定一个计时器，未到时间时等待，到达设定时间时遍历数据库中的书籍数据，依次更新。完成之后继续等待下一个时间点的到来。流程图如图4-3所示。



图4-3 数据更新模块流程图

### 4.3.3 推荐/分类信息请求模块设计

推荐/分类信息请求模块是为了使用户接收到更多的书籍信息，从而找到自己感兴趣的书籍。具体过程为：用户打开界面时自动发起请求，服务器端收到请求之后验证其正确性，不正确的话返回错误提示。通过查询数据库来取得需要的数据比如推荐信息或者某一个类别的小说列表，获取完成之后返回数据至前端。流程图如图4-4所示。



图4-4 推荐/分类信息请求模块流程图

### 4.3.4 书籍信息查看模块设计

书籍信息查看模块是为了展示书籍的基本信息，用户可以通过本模块来了解书籍的简单内容，从而找到自己感兴趣的书籍。具体过程为：用户点击查看书籍，发送请求至服务器端，服务器端验证请求正确性，不正确返回错误提示，正确的话查询数据库寻找书籍内容。找到书籍直接返回书籍数据，找不到内部调用搜索模块更新数据库。更新失败返回错误提示。成功返回书籍信息。流程图如图4-5所示。



图4-5 书籍信息查看模块流程图

### 4.3.5 章节阅读模块设计

章节阅读模块是本系统的核心内容之一，用户可以阅读自己喜欢书籍的章节，本模块包括章节之间的跳转、章节内容的获取等。具体过程为：点击阅读书籍之后，发送阅读章节请求。后天验证数据正确性，不正确返回错误提示，正确的话使用爬虫爬取该章节内容信息，还包括上一章地址、下一章地址、书籍信息地址、章节列表地址等信息。如果爬取失败返回错误提示。成功的话，通过这些信息，用户可以进行章节之间跳转，章节列表的获取等操作。流程图如图4-6所示。



图4-6 章节阅读模块流程图

## 4.4 数据库设计

## 4.5本章小结

本章主要介绍了如何实现需求分析，把需求分析的逻辑模型转化成可以实现的物理模型，对系统的各个模块进行设计，详细介绍了各个模块的功能，还有核心的模块的流程，应该如何去实现各个核心模块，为中期的编码打下了基础；还介绍了数据库的设计，包括数据库的概念设计，为以后数据库和系统的连接做准备。