# “无微不至”的借阅伴侣

# 需求规格说明书

编写者： 活力小组

目录

[1导言 3](#_Toc486504065)

[1.1目的 3](#_Toc486504066)

[1.2范围 3](#_Toc486504067)

[2系统定义 3](#_Toc486504068)

[2.1项目来源及背景 3](#_Toc486504069)

[2.2项目要达到的目标 3](#_Toc486504070)

[2.3系统整体结构 4](#_Toc486504071)

[3应用环境 4](#_Toc486504072)

[3.1系统运行网络环境 5](#_Toc486504073)

[3.2系统运行硬件环境 5](#_Toc486504074)

[3.3系统运行软件环境 5](#_Toc486504075)

[4功能规格 6](#_Toc486504076)

[4.1角色定义 6](#_Toc486504077)

[4.1.1用户 6](#_Toc486504078)

[4.1.2管理员 6](#_Toc486504079)

[4.1.3超级管理员 7](#_Toc486504080)

[4.2客户端子系统 7](#_Toc486504081)

[4.3管理端子系统 8](#_Toc486504082)

[5性能需求 8](#_Toc486504083)

[5.1界面需求 8](#_Toc486504084)

[5.2响应时间需求 8](#_Toc486504085)

[5.3可靠性需求 8](#_Toc486504086)

[5.4开放性需求 9](#_Toc486504087)

[5.5系统安全性需求 9](#_Toc486504088)

[6产品提交 9](#_Toc486504089)

# 1导言

## 1.1目的

本文档是关于“无微不至”的借阅助手借书系统的功能和性能的要求，重点描述了网上招聘系统的设计需求，将作为在概要设计阶段的主要参考资料。

本文档的预期读者是：

* 设计人员
* 开发人员
* 项目管理人员
* 测试人员
* 用户

## 1.2范围

该文档是借助于当前系统的逻辑模型导出目标系统的逻辑模型，解决整个项目系统的“做什么”的问题。在这里，对于开发技术并没有涉及，而主要是通过建立模型的方式来描述用户的需求，为客户、用户、开发方等不同参与方提供一个交流的渠道。

# 2系统定义

## 2.1项目来源及背景

在科技高度发达，信息大爆炸的今天，人们的生活习惯渐渐地发生了改变。阅读就是其中一个很典型的例子。如今我们阅读的渠道很多：报纸、刊物、书籍、邮件、微博、微信、今日头条、腾讯新闻、澎湃、简书等等不胜枚举，不管是在家里还是在单位，我们几乎每时每刻都有机会阅读。但问题是，我们是否还记得曾几何时，我们抱着一本书，完完整整地看完一遍又一遍？我们被碎片化了，信息时代让我们的见识广了，但是我们中的大部分却渐渐丧失了独立思考的能力了。实际上，我们成天处于一种被各种信息“洗脑”的状态。

有人说可以订立读书计划，买纸质书来读。不过老话说得好，“书非借不能读也”，一旦买下来，最终大多束之高阁，不了了之。好的习惯往往需要一些“逼迫”的意味在里面。那种害怕错失的那种感觉会让人在这个节奏飞快的时代里为阅读安排出宝贵的时间。

## 2.2项目要达到的目标

1.本系统是基于微信开发一套图书馆借阅系统。

2.系统包括微信小程序、管理员应用及相关业务后台。

3.用户可以通过关注微信小程序，进入本系统，用户可以通过自己的手机号码进行注册，4.注册完成后，用户通过浏览本系统查找自己喜欢的书籍本将其加入自己的借阅栏目中，当用户要借阅书籍时可以到图书馆中找到相应的书籍，并到前台出示自己的借阅二维码，图书馆的管理员通过扫描用户出示的二维码，读出用户将要借阅的书籍，并与实物进行比对，如果用户借阅的书籍与实物一样，管理员允许用户借阅。用户交完押金可将该书籍借走,用户的借阅期限是15天，期间系统会提醒用户何时归怀该图书。如用户超过期限还未归还，则用户将支付相应的费用。

5.当用户在期限内归还图书，用户到图书馆出示自己的归还图书的二维码，管理员扫描后得到用户需要归还的图书，确认无误后管理员确认归还。

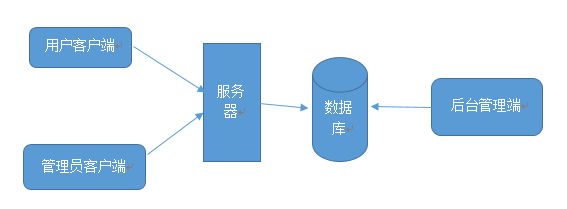
6用户对于自己想要借阅的书籍可以通过搜索等操作查找并加入借阅栏，等待借阅。对于想要借阅但图书已被借的情况，用户可以提前预定，等到图书馆有书时，可以通知用户前去取书。

## 2.3系统整体结构

本系统分为客户端和管理端，客户端主要是用户查询借阅图书，以及管理员对用户借书还书进行扫码。

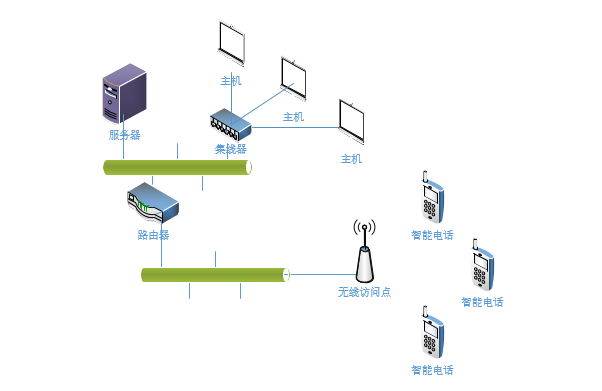
管理端主要管理后台图书的增删改查，管理员信息的录入删除，以及用户信息的查看。

关系图如下：



# 3应用环境

## 3.1系统运行网络环境



## 3.2系统运行硬件环境

本系统的硬件环境如下：

* 客户机：普通PC
  + CPU：P4 1.8GHz
  + 内存：256MB以上
  + 分辨率：推荐使用1024\*768像素
* WEB服务器
  + CPU：P4 1.8GHz
  + 内存：256MB以上
* 数据库服务器
  + CPU：P4 1.8GHz
  + 内存：256MB以上

## 3.3系统运行软件环境

* 操作系统：Windows 2000
* 数据库：SQL Server 2000
* 开发工具包：JDK Version 1.4.2
* JSP服务器：Tomcat
* 浏览器：IE6.0

# 4功能规格

我们采用面向对象分析作为主要的系统建模方法，使用UML(Unified Modeling Language)作为建模语言。UML为建模活动提供了从不同角度观察和展示系统的各种特征的方法。在UML中，从任何一个角度对系统所作的抽象都可能需要几种模型来描述，而这些来自不同角度的模型图最终组成了系统的映像。

Use Case描述的是“actor”(用户、外部系统以及系统处理)是如何与系统交互来完成工作的。Use Case模型提供了一个非常重要的方式来界定系统边界以及定义系统功能，同时，该模型将来可以派生出动态对象模型。

设计Use-case时，我们遵循下列步骤：

第一步，识别出系统的“actor”。Actor可以是用户、外部系统，甚至是外部处理，通过某种途径与系统交互。重要的是着重从系统外部执行者的角度来描述系统需要提供哪些功能，并指明这些功能的执行者(Actor)是谁。尽可能地确保所有Actor都被完全识别出来。

第二步，描述主要的Use Case。可以采取不断地问自己“这个Actor究竟想通过系统做什么？”来准确地描述Use Case。

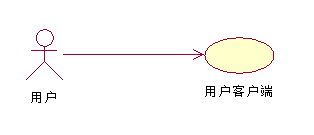
第三步，重新审视每个Use Case，为它们下个详尽的定义。

## 4.1角色定义

角色或者执行者（Actor）指与系统产生交互的外部用户或者外部系统。

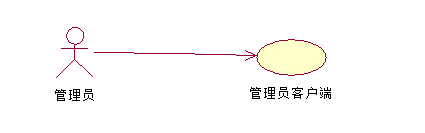
### 4.1.1用户

用户可以在本系统的用户客户端注册和登录，查看书籍，预定书籍，借阅和归还书籍。



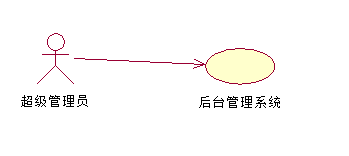
### 4.1.2管理员

管理员可以通过管理员账号和密码登录管理员客户端，管理用户书籍的归还借阅等操作。

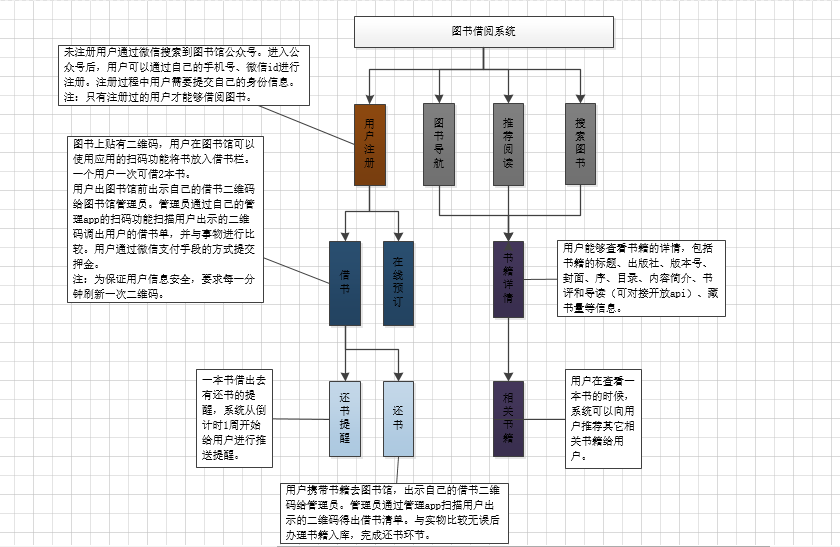


### 4.1.3超级管理员

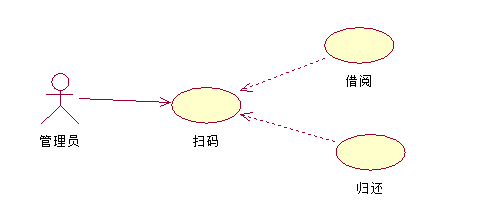
超级管理员可以登录后台管理系统，查看用户和管理员的信息，并对书籍的信息的进行维护，并对管理员有增加和删除的权限。



## 4.2客户端子系统



## 4.3管理端子系统



# 5性能需求

## 5.1界面需求

系统的界面要求如下：

１）页面内容：主题突出，站点定义、术语和行文格式统一、规范、明确，栏目、菜单设置和布局合理，传递的信息准确、及时。内容丰富，文字准确，语句通顺；专用术语规范，行文格式统一规范。

２）导航结构：页面具有明确的导航指示，且便于理解，方便用户使用。

３）技术环境：页面大小适当，能用各种常用浏览器以不同分辨率浏览；无错误链接和空链接；采用CSS处理，控制字体大小和版面布局。

４）艺术风格：界面、版面形象清新悦目、布局合理,字号大小适宜、字体选择合理，前后一致，美观大方；动与静搭配恰当,动静效果好；色彩和谐自然,与主题内容相协调。

## 5.2响应时间需求

无论是客户端和管理端，当用户登录，进行任何操作的时候，系统应该及时的进行反应，反应的时间在5秒以内。系统应能监测出各种非正常情况，如与设备的通信中断，无法连接数据库服务器等，避免出现长时间等待甚至无响应。

## 5.3可靠性需求

系统应保证7X24内不当机，保证20人可以同时在客户端登录，系统正常运行，正确提示相关内容。

## 5.4开放性需求

系统应具有十分的灵活性，以适应将来功能扩展的需求。

## 5.5系统安全性需求

系统有严格的权限管理功能，各功能模块需有相应的权限方能进入。系统需能够防止各类误操作可能造成的数据丢失，破坏。防止用户非法获取网页以及内容。

# 6产品提交

提交产品为：

a) 应用系统软件包

b) 数据库初始数据

c) 系统开发过程文档

d) 系统使用维护说明文档