

**基于 MVC 设计模式的在线图书馆管理系统项目开发总结
报告**

**Online Library Management System Based On MVC Design
Pattern Software Project Development Summary Report**

2019-1 发布

目录

1	范围.....	1
1.1	标识.....	1
1.2	系统概述.....	1
1.3	文档概述.....	1
2	任务来源与研制依据.....	1
3	软件概述.....	1
4	CSCI 体系结构设计.....	2
4.1	软件研制过程概述.....	2
5	软件满足任务指标情况.....	3
6	质量保证情况.....	4
6.1	质量保证措施实施情况.....	4
6.2	软件重大技术质量问题和解决情况.....	5
7	配置管理情况.....	6
7.1	软件配置管理要求.....	6
7.2	软件配置管理实施情况.....	6
7.3	软件配置状态变更情况.....	6
8	测量和分析.....	6
9	结论.....	7
10	注释.....	7

1 范围

1.1 标识

中文名称：《项目开发总结报告》。

英文名称：“Project Development Summary Report (PDSR)”。

文档版本：“1.0” 。

文档编号：“OLMS-PDSR-1.0” 。

1.2 系统概述

基于 MVC 设计模式的在线图书馆管理系统(Online Library Management System Based On MVC Design Pattern)适用于各版本的 Windows 系统，本软件用于对图书馆进行管理，大致包括：图书管理员管理，读者管理和书籍管理三个方面。具有一定的并发性，支持多人同时进行操作，功能较为完备，系统可用性、可靠性高，易于维护，具有较高的效率。

1.3 文档概述

本文档对基于 MVC 设计模式的在线图书馆管理系统(Online Library Management System Based On MVC Design Pattern)中的整个研制和开发情况进行了总结。包括任务来源与研制依据、软件概述、软件研制过程、软件指标满足情况、质量保证情况、配置管理情况、测量分析和结论。

2 任务来源与研制依据

在信息化高速发展的今天，“时间效率”以及“用户满意度”这样的名词正主导着人们的生活和发展，有必要设计开发一个在线图书馆管理系统。通过系统功能有效的解决读者查询书籍信息过慢，借阅操作流程过于繁杂，图书管理员管理书籍、读者信息太过麻烦等问题，从而提高所有系统用户的时间效率。

3 软件概述

随着应试教育的普及，文盲率的不断下降，人们对书本的需求也越来越大，随着图书馆注册用户人数的增多，以及各类书目的出版，仅仅凭借手工记录信息已经越来越难以满足当代社会的需求。在国家“互联网+”政策的推动下，创建一个帮助图书馆工作人员管理读者及书籍信息，以及图书馆读者查看个人借阅记录与查询书籍信息的系统是大势所趋、势在必行的。

基于 MVC 设计模式的在线图书馆管理系统（OLMS）利用现代软件工程技术，直击要害，完美地解决了上述的问题，它既方便了图书馆工作人员管理读者、书籍信息，又方便了读者查询书籍、自己的借阅记录，有效的提高了工作人员的办事效率，显著的增强了用户的满意度，比如图书馆工作人员可以方便地通过系统查看用户的信息，对用户进行增删改查；用户可以通过系统便捷地查看需要的

书目的具体位置，系统也能够自动的向即将要缴纳罚金的用户发送提示信息。

它的建设将有效地提高用户对图书馆满意度，让更多的人选择去图书馆借阅实体图书，它能够成为一套新兴的图书馆管理方法。

4 CSCI 体系结构设计

4.1 软件研制过程概述

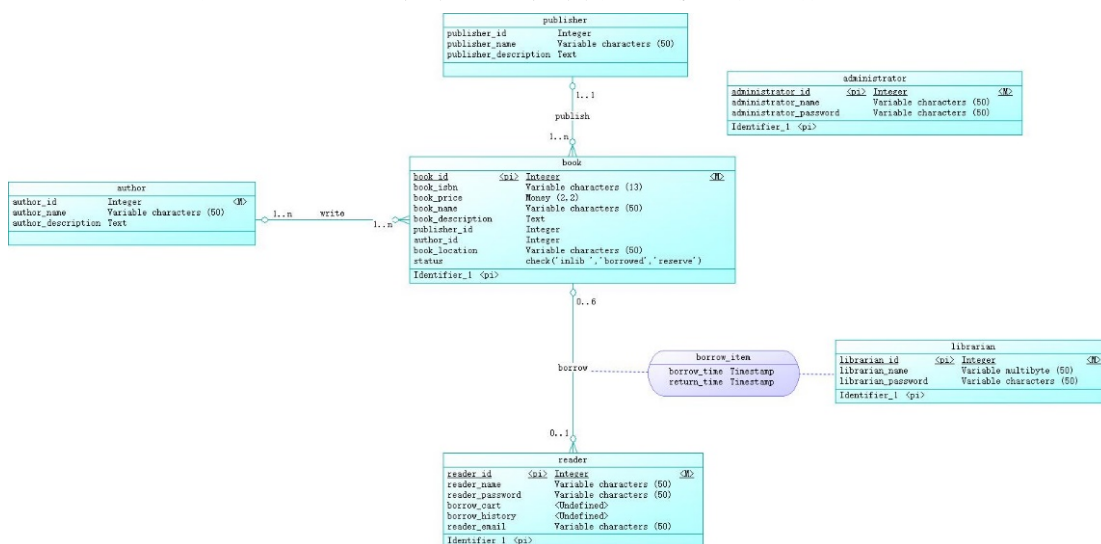
a) 系统要求分析和设计

本在线图书馆管理系统根据 MVC 设计模式，主要包含对数据库中表的数据的增删改查。

b) 对于用户的任意输入，均需要判断输入的合法性，在输入数据的合法性的基础上进行相应的操作，操作后也要返回相应的数据以表示操作是否成功或者失败。

c) 对于用户的输入通过正则表达式进行判断以保证输入的合法性

d) 对于不同的用户显示不同的数据，其数据库 ER 模型如下所示：



图一. 数据库 ER 模型

对于系统管理员可以对图书管理员的表数据进行读写，读者可以对图书相关的表数据进行读操作，对读者的表数据可以进行读写操作，图书馆管理员可以对图书馆管理员表数据，图书相关表数据，读者表数据进行读写操作

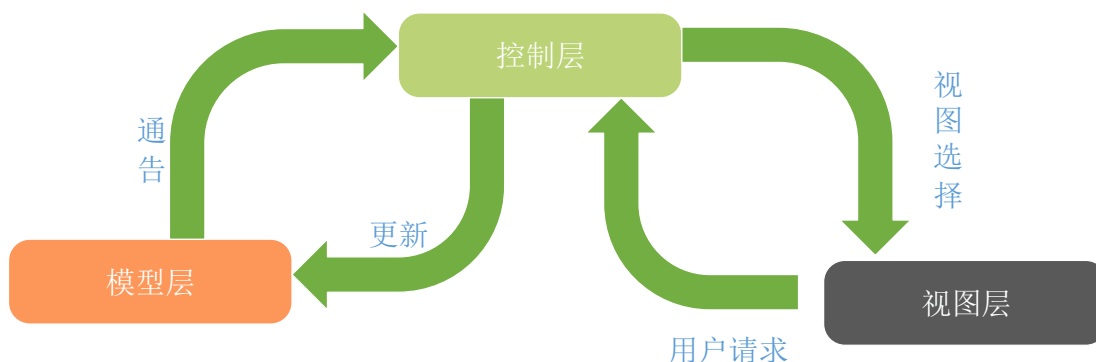
e) 为保证系统的安全性针对不同的用户设置不同的账号区间，区间排列顺序为系统管理员、图书馆管理员、读者。同时用户的密码也需要进行保护，统一采用 MD5 码对用户的密码进行加密。

f) 软件需求分析

系统管理员，图书馆管理员，读者根据自身实际身份登陆基于 MVC 设计模式的在线图书馆管理系统，如果登录成功，则启动相应的管理系统，以及相应的权限，实现各项功能。

g) 软件设计

基于 MVC 设计模式的图书馆管理系统将系统拆分成三大部分，分别是模型层，视图层，控制层，其中用户的操作在视图层，用户在视图层的输入传入控制层，由控制层调用模型层的方法实现数据的持久化存储。调用结果如下所示：



图二. 调用关系

为进一步分离不同用户之间的操作，在视图层与控制层分别拆分出系统管理员，图书馆管理员与读者的部分，对于不同的用户进行不同的操作。

h) 软件实现和单元测试

软件实现过程采用前后端分离的方式，单元测试采用白盒测试的方法。

i) 软件集成与测试

软件集成测试采用自底向上的测试方式。

5 软件满足任务指标情况

软件满足任务指标情况如下表所示：

	功能要求	功能满足情况	性能满足情况
Admin	登入系统	完全实现	满足用户常规使用
	查询 librarian 的信息	完全实现	满足用户常规使用
	删除 librarian 的账号	完全实现	满足用户常规使用
	修改 librarian 信息	完全实现	满足用户常规使用
	注册 librarian	完全实现	满足用户常规使用
	修改密码，名字	完全实现	满足用户常规使用
	退出系统	完全实现	满足用户常规使用
Reader	登入系统	完全实现	满足用户常规使用
	搜索图书	完全实现	满足用户常规使用
	查询个人信息	完全实现	满足用户常规使用
	借阅历史	完全实现	满足用户常规使用

Librarian	修改名字，邮箱，密码	完全实现	满足用户常规使用
	添加图书到借阅车	完全实现	满足用户常规使用
	退出系统	完全实现	满足用户常规使用
	登入系统	完全实现	满足用户常规使用
	修改密码，名字	完全实现	满足用户常规使用
	修改 Reader 信息(名字，密码，邮箱，账号状态)	完全实现	满足用户常规使用
	增加一个读者	完全实现	满足用户常规使用
	删除一个读者	完全实现	满足用户常规使用
	查询读者信息	完全实现	满足用户常规使用
	借书操作	完全实现	满足用户常规使用
	查看借阅车，审批借阅申请(以申请借阅次序)	完全实现	满足用户常规使用
	增加图书	完全实现	满足用户常规使用
	删除图书	完全实现	满足用户常规使用
	查询图书信息	完全实现	满足用户常规使用
	修改图书信息	完全实现	满足用户常规使用
	还书操作	完全实现	满足用户常规使用
	退出系统	完全实现	满足用户常规使用

表一. 软件满足任务指标

6 质量保证情况

6.1 质量保证措施实施情况

质量保证组织：

由五名开发人员担任质量保证组织成员。

保证制度：

***负责管理员全部功能，图书管理员对读者的账户的增删改。

***负责读者部分功能与过滤器，图书管理员对购物车的审批。

***负责界面设计，图书管理员对图书的修改查询。

***负责读者部分功能与过滤器，图书管理员对图书的增删。

***负责输入正则判断，主页推荐与新闻显示。

软件研制各个阶段中的各项质量保证安排：

测试完成的功能并向其提交错误报告；

测试完成的功能并向其提交错误报告；

测试完成的功能并向其提交错误报告；

测试完成的功能并向其提交错误报告；

测试完成的功能并向其提交错误报告。

6.2 软件重大技术质量问题 and 解决情况

1) 界面显示问题

故障现象：在界面中的中文会出现乱码。

故障产生的机理：字符在不同的编码格式转换到另一个编码格式会导致乱码。

解决措施：统一编码格式为“UTF-8”。

验证情况：页面显示正常。

2) 安全问题

故障现象：用户的密码在明文存储的时候会导致用户信息泄露。

故障产生的机理：安全度不高。

解决措施：对用户输入的密码进行 MD5 码加密。

验证情况：MD5 码为不可逆算法，安全性提高。

3) 数据库时间与本地时间不一致

故障现象：客户端的时间传到数据库中会产生时差。

故障产生的机理：数据库的时区默认为 UTC 时区比东八区晚 8 个小时。

解决措施：连接数据库时设置时区为客户端时区。

验证情况：时间显示正常。

4) 读写冲突

故障现象：同一操作由多用户同时进行产生读写冲突。

故障产生的机理：没有保证用户的读写顺序。

解决措施：将数据库的连接使用单一实例化的设计模式连接数据库。

验证情况：解决读写冲突。

5) 页面表单内容过多

故障现象：页面中表单的内容过多，导致界面无法加载过慢。

故障产生的机理：页面显示的内容将数据库中的所有内容都展现出来。

解决措施：每次从数据库中去一部分内容出来展现。

验证情况：页面加载速度正常，内容数量合理。

版本号规则：各配置版本标识格式为“s.xy.m”，s 为 1-9 数字，xy 为 0-99 数字，m 为任意长度数字。S 为主版本号，初始版本号为 1，配置项在产品库中变更时，版本号加 1；xy 为受控版本号，初始受控版本号为 00，当体系文件在一次受控库中的变更出库时，受控版本号加 1；m 为开发库版本号，开发库版本号可

根据实际情况省略不写。

7 配置管理情况

7.1 软件配置管理要求

7.2 软件配置管理实施情况

配置项清单：

序号	配置项名称	配置项描述
1	软件开发计划	基于 MVC 设计模式的在线图书馆管理系统软件开发计划
2	概要设计文档	基于 MVC 设计模式的在线图书馆管理系统总体设计方案
3	详细设计文档	基于 MVC 设计模式的在线图书馆管理系统详细设计方案
4	软件需求规格说明	基于 MVC 设计模式的在线图书馆管理系统软件需求规格说明
5	软件交付说明书	基于 MVC 设计模式的在线图书馆管理系统软件交付说明
6	源代码	基于 MVC 设计模式的在线图书馆管理系统源代码
7	可执行文件	在线图书馆管理文件可部署的工程

表一. 配置项清单

7.3 软件配置状态变更情况

无。

8 测量和分析

a) 进度执行情况：

第一版：在检查时，完成度为预期的 90%，原因是在任务分配时，第一版分配任务过多，且在完成过程中，对整体结构进行修改，导致时间延后。

第二版：在检查时，完成度为预期的 115%，原因是在架构确定的情况下，第二版的部分功能与第一版部分功能有相似结构，通过代码复用，可使得计划超额。

第三版：在检查时，完成度为预期的 110%，原因是软件相关功能需求大多数在第一版、第二版时已完成，且代码风格较好，漏洞较少，为测试节省了很多时间。

b) 实际费用使用在计划费用以内。

c) 第一版：实际工作量大于计划工作量。计划工作量未考虑到重新架构的

情况，而在实际完成过程中，发现根据需求，架构需要重新设计，导致工作量加大。

第二版：实际工作量略大于计划工作量。因第一版

第三版：实际工作量小于计划工作量。计划工作时，考虑到测试时会出现较多漏洞，预留了一定的修复工作计划，实际测试时，因前期交流充足，代码风格较好，故模块合并时产生错误较少，使得实际工作量小于计划工作量。

- d) 缺陷 65%分布在设计阶段，20%分布在编码阶段，15%分布在测试阶段。在设计阶段，由于对需求实现过程的不清晰，导致数据库的设计出现一定歧义，需重新设计。编码阶段，由于技术、交流上仍有所欠缺，功能接口、功能实现上产生了一定的缺陷。

9 结论

在开发过程中，团队在理论为指导的基础下，实现技术创新发展的实用化。通过有序开发将过程更加的具体。软件功能、性能指标达到软件任务的要求，软件可以交付需方使用。

10 注释

MVC: model, view, control 的缩写，分别代表模型、视图、控制器。