



绵阳城市学院
MIANYANG CITY COLLEGE

校园博客

总体设计说明书

创建日期：	<u>2021/10/19</u>	文档名称：	<u>总体设计</u>
项目名称：	<u>校园博客</u>	撰写人员：	<u>刘晓虎</u>



绝顶聪明 2.0 组



日期	版本	说明	作者	评审日期	评审人员
2021/10/19	1.0	初稿	刘晓虎		

目录

1 引言.....	1
1.1 编写目的.....	1
1.2 背景.....	1
1.3 定义.....	1
1.4 参考资料.....	2
2.1 需求规定.....	2
2.2 运行环境.....	2
2.3 基本设计概念和处理流程.....	3
2.4 结构.....	4
2.4.1 用户模块.....	4
2.4.2 收藏模块.....	5
2.4.3 检索模块.....	5
2.4.4 文章模块.....	6
2.4.5 管理员模块.....	8
2.5 功能需求与程序的关系.....	8
3 接口设计.....	9
4 运行设计.....	9
4.1 运行模块组合.....	9
5 系统数据结构设计.....	9
5.1 逻辑结构设计要点.....	9
5.2 物理结构设计要点.....	11
5.3 局部 E-R 图.....	11
5.4 完整 E-R 图.....	13
5.5 E-R 图转为关系模式.....	13
6 系统出错处理设计.....	14
6.1 出错信息.....	14
6.2 补救措施.....	15

6.2.1 需求变更.....	15
6.2.2 进度风险.....	15
6.2.3 技术风险.....	15
6.2.4 质量风险.....	15
6.3 系统维护设计.....	16

1 引言

1.1 编写目的

撰写本文档的目的在于明确系统的设计，确定软件所具有的功能和性能，并划定系统边界，重点描述了校园博客系统的设计，将作为项目的开发、测试、验收提供相关依据。

本文档预期读者为：项目开发人员，甲方人员

1.2 背景

项目名称：校园博客

甲方：叙利亚战争组

开发者：绝顶聪明 2.0 组

本项目背景是为了满足甲方需求，为本校大学生开发的一个校园博客系统。学校不仅是学生学习的天地，更是同学之间交流的地方。随着网络信息技术的发展，网络渐渐成了当今在校大学生交流信息的重要渠道。本项目旨在为在校大学生提供一个资源共享信息交流的平台，以达到大学生之间的充分交流。

1.3 定义

名称	含义	备注
CRUD	对相关数据的增加(Create)、查询(Retrieve)、更新(Update)和删除>Delete)	
相关文档	需求分析，概要设计，详细设计，数据库设计，网络接口设计(按需)，部署说明手册，用户操作使用手册	

1.4 参考资料

【1】《软件需求说明书（GB856T——88）》

【2】《校园博客需求分析文档》

【3】《校园博客数据库设计文档》

【4】《校园博客接口设计文档》

【5】《校园博客测试文档》

【6】《校园博客问题反馈文档》

【7】《校园博客用户手册文档》

2.1 需求规定

参考《校园博客需求分析文档》

2.2 运行环境

1. PC 电脑

处理器：I3 以上

内存空间：256M 以上

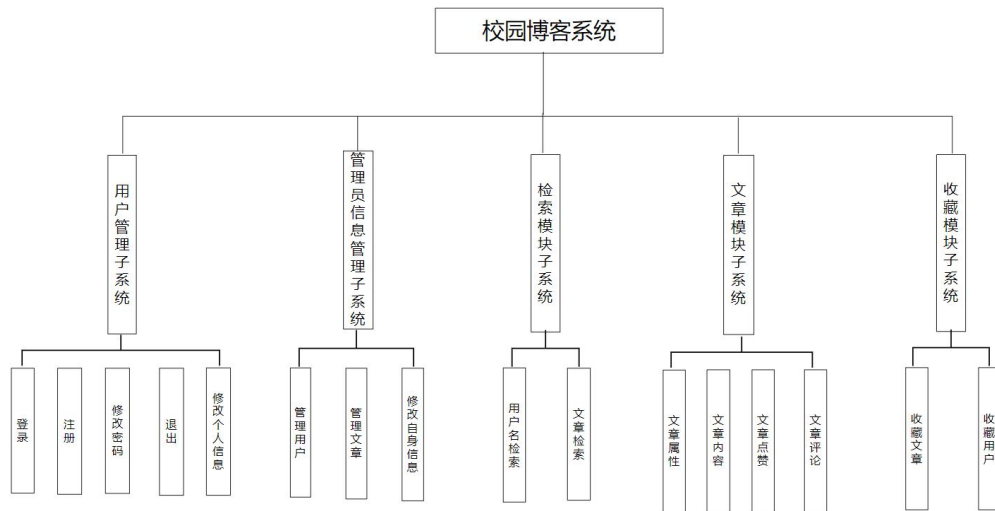
硬盘空间：10G 以上

其他硬件需求，请参照能使 java 语言程序运行的环境和能访问网络的基本需求确定。

2. 手机

要求能够上网，可以用键盘或屏幕进行基本的指定区域准确点击确认，以及信息输入，能够完成信息交互。

2.3 基本设计概念和处理流程



主要模块：用户模块、管理员模块、检索模块、文章模块、收藏模块

2.4 结构

2.4.1 用户模块



● 前端设计

用户注册时需要判断昵称和密码是否为空、是否存在非法字符，用户名前端不用涉及，性别采取复选框的形式，头像支持 jpg、png 等常用格式，注册可选择邮箱或手机验证，邮箱需要判断字符串格式是否是邮箱格式（以@为标识），手机长度应为 11 位（暂不支持海外用户）。

用户登录时需要判断用户名和密码是否存在非法字符和是否为空，使用验证登录则向后端发起验证请求，由后端判断验证码是否正确。

● 后端设计

后端根据前端发送到用户基本信息对用户进行封装，用户密码选用

base64 加密，当前端发起邮箱和手机验证时，向用户发送对应的验证码并判断从前端发送到验证码是否一致。

2.4.2 收藏模块



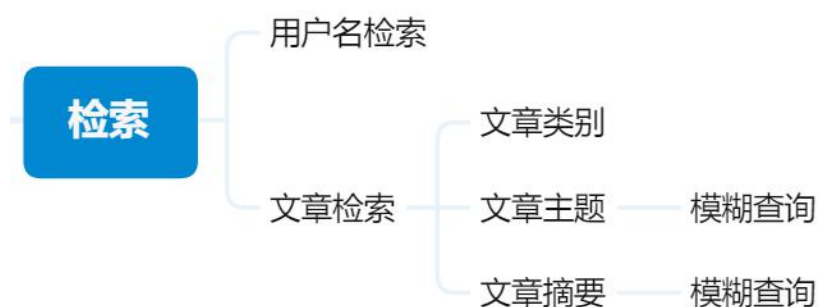
- 前端设计

用户点击文章的★按钮时向后端发送收藏请求，用户可以在个人信息页面查看收藏的文章。

- 后端设计

根据前端发送到请求做出相对应的业务逻辑处理，当用户查看收藏列表时返回该用户收藏的所有文字。

2.4.3 检索模块



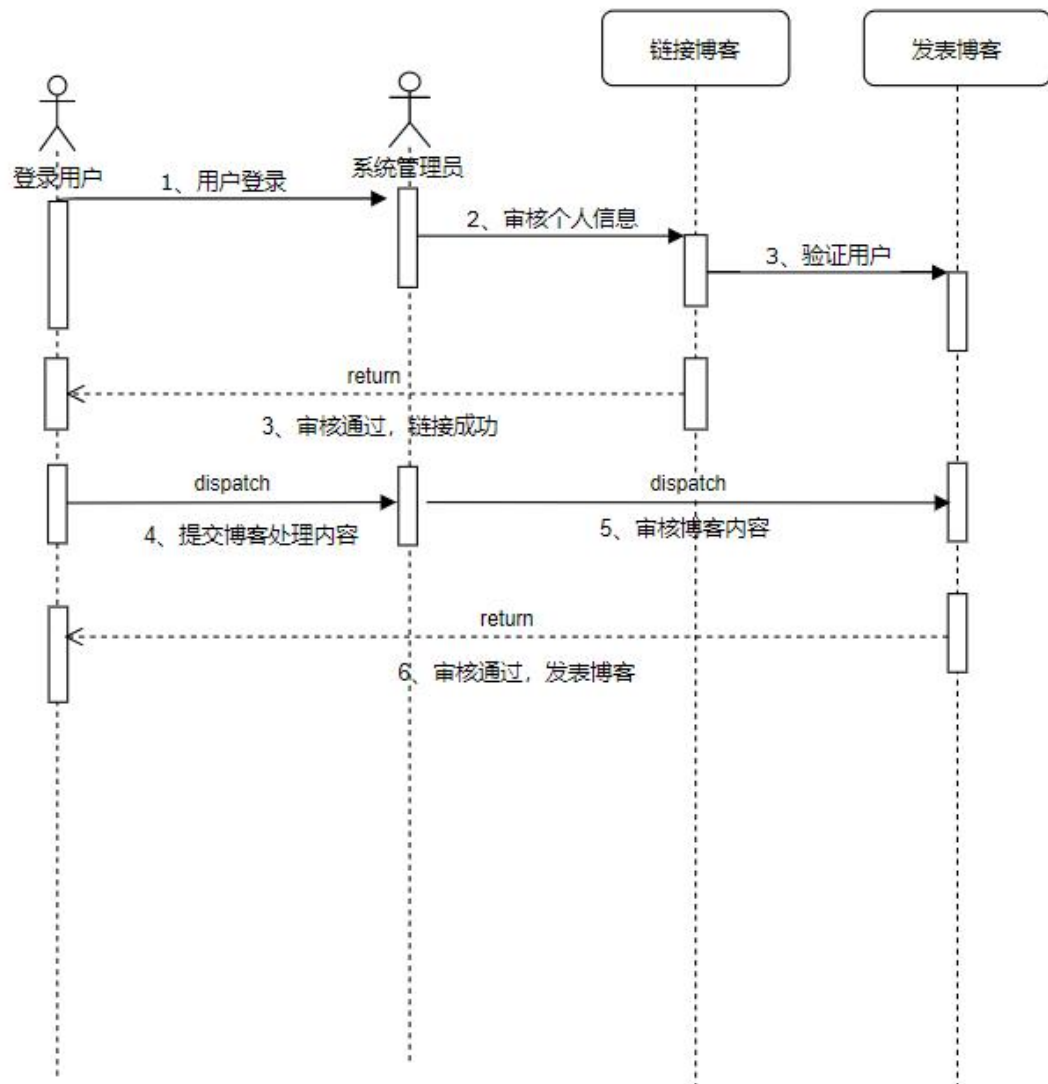
- 前端设计

用下拉框让用户选择搜索用户（用户 ID）还是文章（类别、主题、摘要），前端通过输入框获取用户输入发送给后端并显示后端返回的数据。

- 后端设计

根据前端请求的数据进行查找并返回对应的数据。

2.4.4 文章模块



2.4.5 管理员模块



- 前端设计

管理员从管理员登录界面设计后可以查看所有用户的信息，可以重置用户的密码。可以查看用户的文章，可以驳回文章与删除文章。

- 后端设计

返回所有用户信息与文章信息，如果文章存在敏感字符则进行标注再返回。

2.5 功能需求与程序的关系

本条用一张如下的矩阵图说明各项功能需求的实现同各块程序的分配关系：

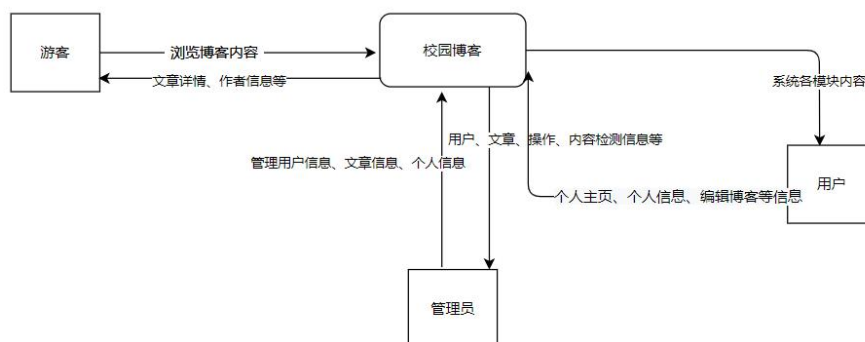
	用户模块	管理员模块	文章模块
用户登录注册	√		
管理员管理		√	
发布文章	√		√
浏览文章	√	√	√
修改文章	√		
邮箱登录	√		

3 接口设计

参考接口文档《校园博客接口设计文档》

4 运行设计

4.1 运行模块组合



5 系统数据结构设计

5.1 逻辑结构设计要点

用户模块具体功能：

- ①用户的注册登录

用户可以通过用户名或者邮箱的方式进行注册或者登录进入本系统；

②用户对文章的管理（CRUD）

用户对文章可以进行基本的增删改查的操作，必须具备有登录权限的用户才可以实现其功能；

③用户对文章的评论

用户可以通过对文章发布评论，该用户发布的评论将会上传到后端，后端与数据库相连，因此管理员可以对其评论进行管理和监控，以防不正当言论的产生，管理员才能做到及时删除；

④用户对文章的收藏

用户可以对自己喜欢的博客进行收藏的功能，收藏之后的文章可以在我的个人界面的收藏里面可以看到。

管理员模块

①管理员对用户的管理

管理员有权限对用户的个人信息进行增删改查，以防用户忘记密码时进行重置密码，对用户的资料进行修改等操作；

②管理员对文章的管理

管理员可以对文章的内容进行查看功能，在用户发表不当文章时进行删除的管理功能。

文章模块

①文章内部信息的管理

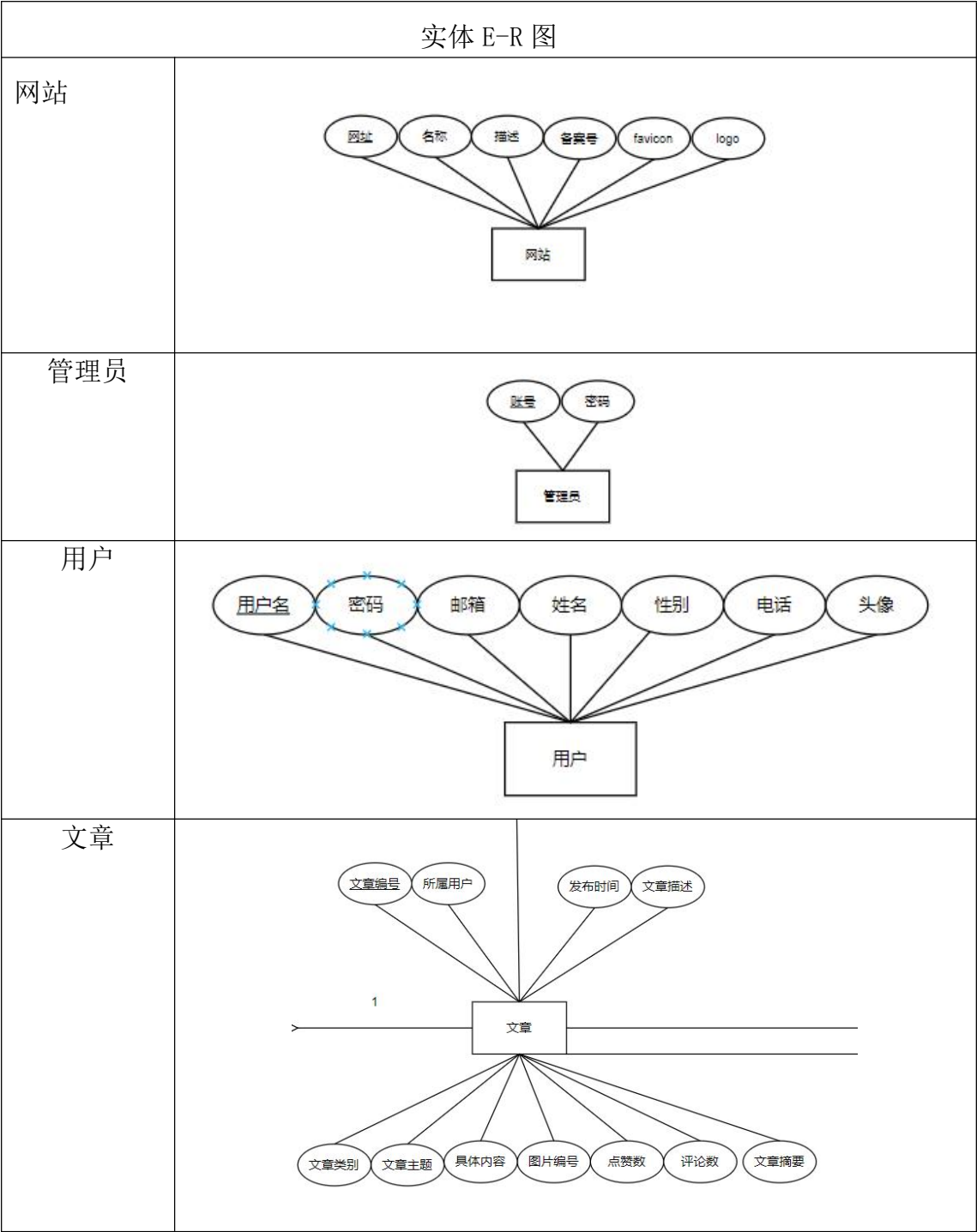
对文章内部信息管理主要是指对文章作者、文章类别、文章主题、发布时间、文章内容等进行管理。

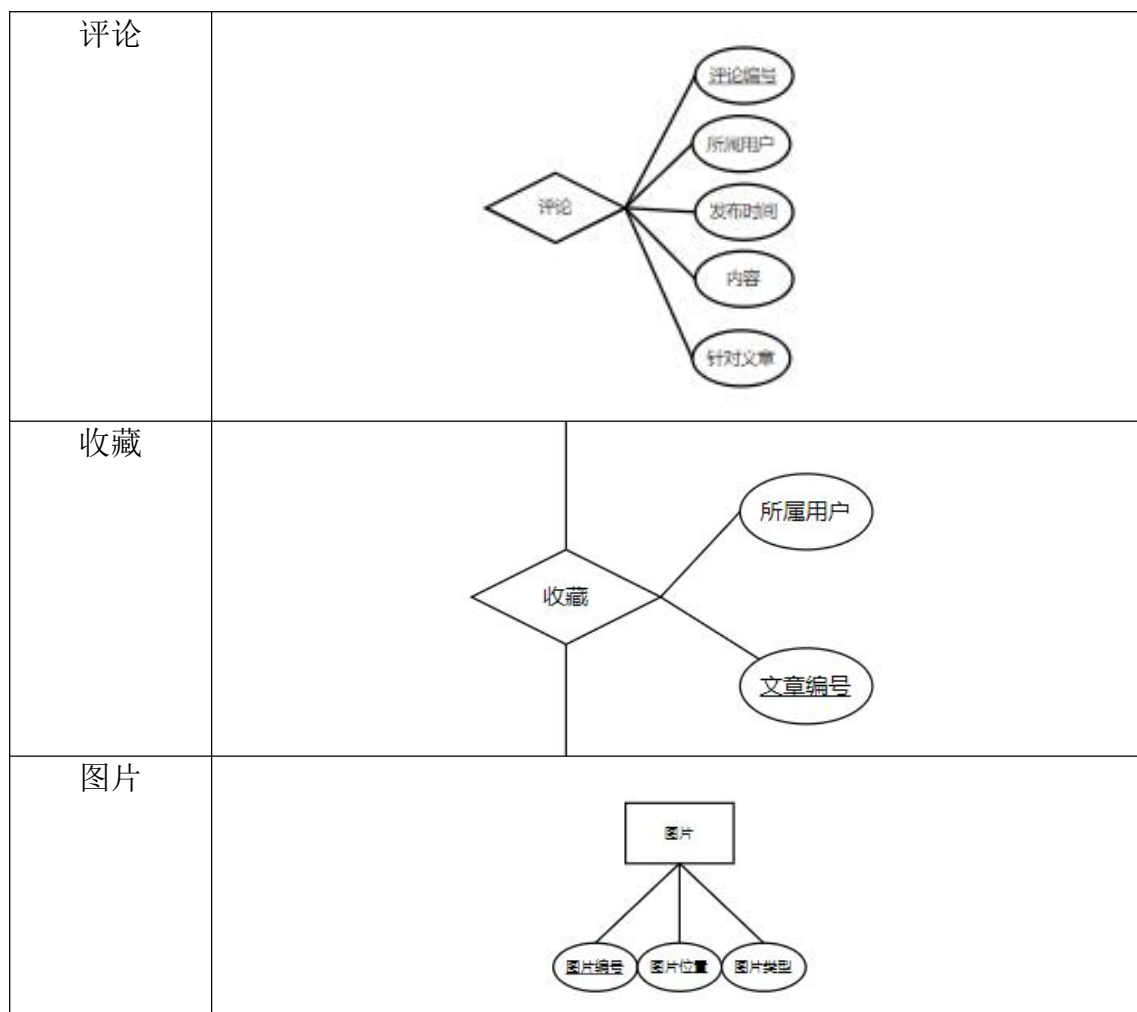
②文章外部信息的管理

对文章外部的信息管理主要是指对文章的用户点赞、用户评论进行管理。

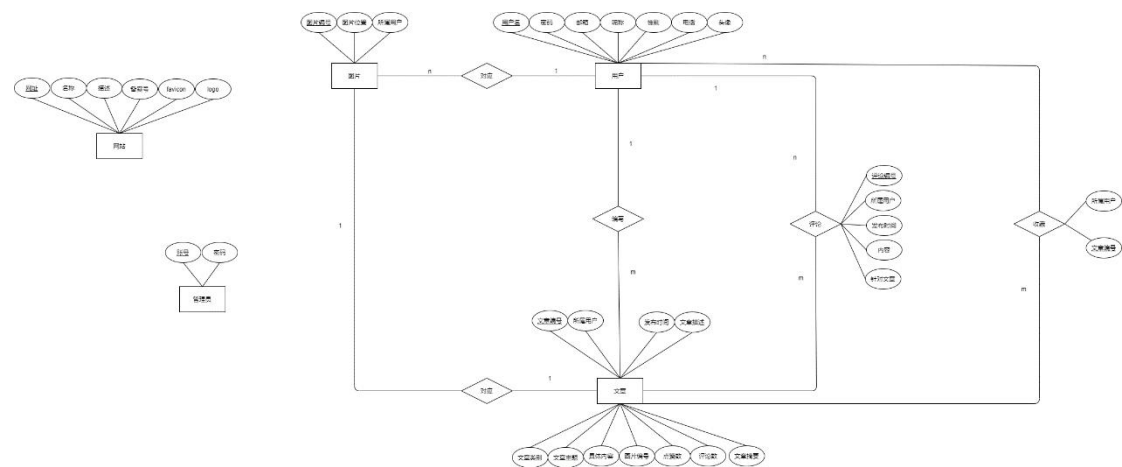
5.2 物理结构设计要点

5.3 局部 E-R 图





5.4 完整 E-R 图



5.5 E-R 图转为关系模式

(1) 网站 (网站, 名称, 描述, 备案号, logo, favicon)

主键: 网站

(2) 用户 (用户名, 密码, 邮箱, 姓名, 性别, 电话, 头像图片)

主键: 用户名 外键: 头像图片

(3) 管理员 (账号, 密码)

主键: 账号

(4) 文章 (文章编号, 所属用户, 文章类别, 发布时间, 文章主题, 文章描述, 具体内容, 修饰图片, 点赞数, 评论数)

主键: 文章编号 外键: 所属用户, 文章类别, 修饰图片

(5) 文章评论 (评论编号, 所属用户, 针对文章编号, 内容, 发布时间)

主键: 评论编号 外键: 所属用户, 针对文章编号

(6) 文章类别 (类别编号, 类别名称, 类别描述, 文章数量)

主键: 类别编号

(7) 图片: (图片编号, 图片位置, 图片类别)

主键: 图片编号

(8) 文章收藏: (文章编号, 所属用户)

联合主键 (文章编号, 所属用户)

6 系统出错处理设计

6.1 出错信息

错误类型	子项	错误原因
数据库错误	连接	连接超时
		连接断开
	数据库本身	数据库代码错误
		数据库溢出
TCP 连接错误	连接	连接超时
		连接断开
	其他 TCP 错误	Socket 自身错误
系统部分自定义错误	权限错误	管理员权限设置故障
	输入错误	用户 ID 错误/为空
		用户密码错误/为空
		验证码错误/为空
	查找错误	为查找到符合要求的记录
链接错误	内部链接错误	网页链接错误
		文件/图片链接错误
	外部链接错误	页面休整或者其他错误无法打开
		友情连接网址维护或其他错误无法打开

补救措施 说明故障出现后可能采取的变通措施，主要包括：对于软错误，需要在添加 /

6.2 补救措施

6.2.1 需求变更

预防这种风险的办法是需要团队成员的高度配合和密切协作的阶段,在进行需求分析的时候要仔细分配团队成员的工作,要求需求分析阶段的团队按照项目管理中典型的矩阵式结构来开展,这种结构能够有效的利用项目资源,减少条块分割的冲突,增加了沟通和协调的机会,降低了项目的执行成本,能够充分发挥项目经理和个分组人员的积极性。在需求分析工作中,重点要解决好优先级别更高的需求项的调研及确认工作。可最大限度地降低需求变更发生的可能性,将变更造成的影响减小到最小。

6.2.2 进度风险

在项目实施的时间进度管理上,需要充分考虑各种潜在因素,适当留有余地;任务分解要详细,便于考核;在执行过程中,应该强调项目按照进度执行的重要项,再考虑任何问题时,都要将保持进度作为先决条件;同时,合理利用赶工期及快速跟进等方法,充分利用资源。

6.2.3 技术风险

对新技术的使用要谨慎,要循序渐进,尽量采用成熟的技术方案完成软件开发工作。在技术创新与技术风险之间进行平衡,并做好创新技术的研究和试验工作。需要对软件项目过程中使用的各种技术进行评估,软件项目管理在制定软件开发计划时必须考虑这些因素,并做出合理的权衡决策。

6.2.4 质量风险

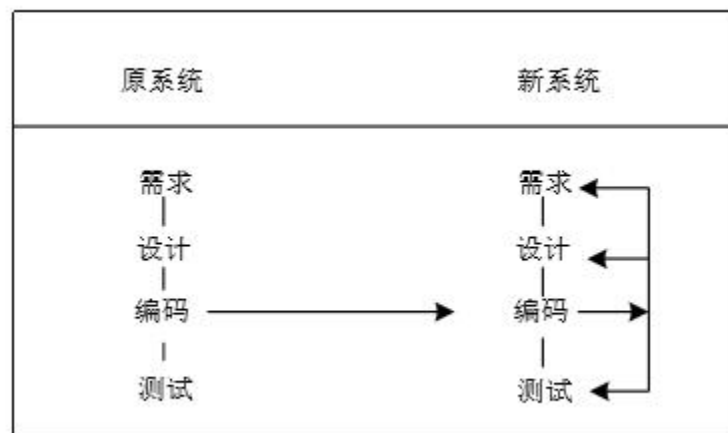
经常和用户交流工作成果、品牌管理采用符合要求的开发流程、认真组织对产出物的检查和评审、计划和组织严格的独立测试等。

6.3 系统维护设计

6.3.1 快速修复模型

基本过程：修改编码，然后对相应的文档(被修改影响的)进行必要的修改

存在的问题：受时间限制，软件维护缺乏合适的计划、设计、影响分析和回归测试；软件系统的可维护性会下降



6.3.2 迭代维护方法

过程：系统维护从分析现有系统的需求、设计、编码和测试文档开始，进而修改受影响的高级说明文档，再将这种修改应用到所有的系统文档；进化过程的每一步都是根据对现有系统的分析而对系统进行再设计的

特点：文档可以随着代码的改变而进行更新

