**System Design Specification**

**\_\_\_XXX\_\_\_\_系统设计说明书**

|  |  |  |  |
| --- | --- | --- | --- |
| Prepared by  拟制 | 张三 | Date  日期 | 填写日期 |
| Reviewed by  评审人 | XXX（讲师） | Date  日期 | 填写日期 |
| Approved by  批准 | XXX（学术主任） | Date  日期 | 填写日期 |

**Revision Record修订记录**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date**  **日期** | **Revision Version**  **修订版本** | **Change Description**  **修改描述** | **Author**  **作者** |
| 填写日期 | 1.0 | 第一稿 | 组员姓名 |
| 填写日期 | 1.5 | 修订稿 | 组员姓名 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目 录

[1. Introduction  简介 4](#_Toc448395722)

[1.1 Purpose  文档目的 4](#_Toc448395723)

[1.2 Scope  本文档适用范围 4](#_Toc448395724)

[1.3 Name 软件名称 4](#_Toc448395725)

[1.４Applications软件应用领域 4](#_Toc448395726)

[2. High Level Design概要设计 5](#_Toc448395727)

[2.1系统功能设计 5](#_Toc448395728)

[2.2 系统架构设计 5](#_Toc448395729)

[2.2.1 C/S架构 5](#_Toc448395730)

[2.2命名设计 5](#_Toc448395731)

[3 Database数据库设计 7](#_Toc448395732)

[3.1表关系 7](#_Toc448395733)

[3.2数据表设计 7](#_Toc448395734)

[4类详细设计 8](#_Toc448395735)

[４.1 Common Class 公共类 8](#_Toc448395736)

[４.2 DBHelper Class 数据库操作类 8](#_Toc448395737)

[４.3 User Class 用户实体类 9](#_Toc448395738)

[5.Low Level Model Design 模块详细设计 10](#_Toc448395739)

[5.1 用户登录 10](#_Toc448395740)

[5.2用户管理（用户表的增删改查） 11](#_Toc448395741)

# 1. Introduction  简介

## **1.1 Purpose  文档目的**

编写此文档的目的是在系统的需求分析的基础上，做概要及详细设计。

概要设计主要解决了实现该系统需求的程序模块设计问题。包括如何把该系统划分成若干个模块、决定各个模块之间的接口、模块之间传递的信息，以及数据结构、模块结构的设计等。同时制定规范：代码体系、接口规约、命名规则。这是项目小组今后共同作战的基础，有了开发规范和程序模块之间和项目成员彼此之间的接口规则、方式方法，大家就有了共同的工作语言、共同的工作平台，使整个软件开发工作可以协调有序地进行。在以下的概要设计报告中将对在本阶段中对系统所做的所有概要设计进行详细的说明。

详细设计说明书的目的是为了使开发人员在完成概要设计说明书的基础上完成规定的各个功能块的具体实现的设计工作，通过所产生的功能模块进行过程描述，开发一个可以直 接转化程程序语言的软件标示，这样可以使软件设计人员很好的了解软件的功能，并对软件的所有模块有一个更深的了解，对系统的各个模块、程序、子系统分被进行实现层面上的要求和说明，为进行后面的实现和测试做准备，检测本系统的各项性能制表，发现纰漏及时修补。

## **1.2 Scope  本文档适用范围**

适用范围：程序员、项目经理、项目总监、测试人员．

## **1.3 Name 软件名称**

软件名称:XX系统

## 1.４Applications软件应用领域

本软件适用于何种领域．

# 2. High Level Design概要设计

## **2.1系统功能设计**

本项目主要划分为后台登录、用户管理、栏目管理、信息管理、首页、信息分类、信息搜索和信息展示九个模块。其中各个模块中包含不同的系统功能，其关系可以通过如图10.9所示的系统功能结构图进行描述。

## **2.2 系统架构设计**

### 2.2.1 三层架构

*本项目采用三层架构进行开发，其优势在于便于分工合作、易于扩展、便于维护，其三层之间的关系如图17.14所示。其中表示层用于与用户进行交互；业务逻辑层负责处理项目业务流程；数据访问层负责操作数据库；Model层中的实体类是三层之间的数据载体，用于在三层之间传递数据；*



## **2.2命名设计**

解决方案名：Pic

项目名：Pic,Model,DAL,BLL

命名空间：Pic

其他命名规则：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **模块名称** | **功能** | **优先级** | **涉及网页** | |
| 管理端 | 5.1管理员登录 | A | 管理员登录 | Login.aspx |
| A | 后台管理主窗体 | Default.aspx  Index |
| 5.2用户管理 | A | 用户管理 | 后台:aspxUser(Add,Del,Update,List).aspx |
|  | 前台:User.aspx  User.html |
| A | 新增用户 | UserInsert |
| B | 修改用户 | UserUpdate |
| 模块一 |  |  |  |
|  |  |  |
|  |  |  |
| 模块二 |  |  |  |
|  |  |  |
|  |  |  |

2.窗体中的交互控件应按照控件命名规范完成命名，比如文本框应命名为txtXXX

3.项目中的类、结构体、函数（方法）、枚举遵循Pascal命名法，首字母大写

4.局部变量遵循骆驼命名法，首字母小写

5.数据库名、表名、字段名遵循Pascal命名法，首字母大写

6.以上命名均需采用有意义的单词实现

# 3 Database数据库设计

## **3.1**表关系

数据库中的表关系图或者PD设计图。

## **3.2数据表设计**

本项目数据库（XXX）共包含X个表，其表结构如下表所示。

项目所有数据库表见表3.1。

**表3.1 表设计**

|  |  |
| --- | --- |
| 表名 | 功能说明 |
| Users | 用户表 |
|  | 图片分类表 |
|  | 图片信息表 |
|  |  |
|  |  |

Users用户表

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name  字段名称 | Field Type  字段类型 | Field Restrain  字段约束 | Descriptiion  注释 |
| UserID | Nvarchar(10) | 主键 | 登录名 |
| UserName | Nvarchar(20) | 非空 | 真实姓名 |
| UserPass | Nvarchar(20) | 非空 | 密码 |
| IsAdmin | bit | 非空，默认1 | 是否管理员  1-管理员，0-用户 |

# **4类详细设计**

本项目中共包含X个类，项目所有类见下表。

**类设计**

|  |  |
| --- | --- |
| 表名 | 功能说明 |
|  |  |
| DBHelper | 数据库操作类 |
| User | User,UserDAL(增删改查,登录),UserBLL |
| Type | Type,TypeDAL(增删改查),TypeBLL |

## **４**.1 Common Class 公共类

1) OverView简介

该类用于保存项目相关全局信息

**2) Attributes 属性**

|  |  |  |  |
| --- | --- | --- | --- |
| Visibility  可见性 | Name  属性名 | Type  类型 | Description  描述 |
| Public static | UserID | String | 登录用户名 |
| Public static | UserName | String | 登录用户真实姓名 |

## **４**.2 DBHelper Class 数据库操作类

1) OverView简介

该类用于设置数据访问相关的公共字段和方法

**2) Attributes 属性**

|  |  |  |  |
| --- | --- | --- | --- |
| Visibility  可见性 | Name  属性名 | Type  类型 | Description  描述 |
| Public static | ConnString | String | 数据库链接字符串 |
|  |  |  |  |

**3) Methods 方法**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Visibility  可见性 | Name  方法名 | Type  返回类型 | Parameter  参数描述 | Description  描述 |
|  |  |  |  |  |
| Public static | CloseConnection | void |  | 关闭数据库连接池对象 |
| Public static | GetDataReader | SqlDataReader | Sql语句 | 查询，获取DataReader |
| Public static | GetDataTable | DataTable | Sql语句 | 查询，获取DataTable |
| Public static | ExecuteNonQuery | bool | Sql语句 | 执行增改删语句 |
| Public static | ExecuteScalar | object | Sql语句 | 执行集合函数 |

## **４**.3 User Class 用户实体类

1) OverView简介

该类为用户实体类（）

**2) Attributes 属性**

|  |  |  |  |
| --- | --- | --- | --- |
| Visibility  可见性 | Name  属性名 | Type  类型 | Description  描述 |
| Public | UserID | String | 登录名 |
| Public | UserName | String | 姓名 |
| Public | UserPass | String | 密码 |
| Public | IsAdmin | int | 是否管理员，1-管理员，0-用户 |

# 

# 5.Low Level Model Design 模块详细设计

## 5.1 用户登录



**(1)Process analysis业务流程分析**

后台登录模块的业务流程较为简单，分为以下步骤进行。

1.首先在登录界面中输入用户名和密码

2.点击“登录”按钮实现登录功能，

|  |  |  |
| --- | --- | --- |
| **页面内容** | | |
| **信息字段** | **说明** | **数据来源** |
| **用户名** | 用户的名称 | 手工输入 |
| **密码** | 用户的密码 | 手工输入 |

|  |  |
| --- | --- |
| **页面操作** | |
| **登录** | 验证正确的用户名和密码，跳转到主界面  提示登录失败  登录成功?  是  否  进入主界面  非空验证成功?  查询用户  查询用户表select  提示不能为空  是  否  获取输入用户名和密码  **查询SQL语句：**  Select count(\*) from 用户表 where 用户名=’输入的用户名’ and 密码=’输入的密码 |
| **注册新用户** | 跳转到注册页面 |

’