《一锤定音》软件系统设计说明书

项目总负责人：张海

项目开发组长：杨霄

项目测试组长：龙思

项目小组成员：崔菁菁、张盛吉、黄瑾行、杨寿 才、王佳琦

2021年4月22日

目录

[1 引言 3](#_Toc1072)

[1.1 编写目的 3](#_Toc28417)

[1.2 背景 3](#_Toc21985)

[1.3 定义 3](#_Toc31542)

[1.4 参考资料 4](#_Toc19845)

[2 系统功能设计 5](#_Toc24352)

[2.1 总体结构的分析与设计 5](#_Toc25190)

[2.1.1 设计目的和原则 5](#_Toc31554)

[2.1.2 设计策略 7](#_Toc10195)

[2.1.3 系统设计UML图 7](#_Toc23278)

[2.2 模拟法庭模块设计 12](#_Toc14423)

[2.2.1 设计目的和原则 12](#_Toc19414)

[2.2.2 设计思想 13](#_Toc29592)

[2.2.3 基本处理流程 14](#_Toc25921)

[2.2.4 数据字典 16](#_Toc26215)

[3 数据库设计 18](#_Toc28096)

[3.1 数据库关系E-R图 18](#_Toc20419)

[3.2 数据库表格设计 18](#_Toc14400)

[4 接口及过程设计 21](#_Toc12071)

[4.1 用户接口 21](#_Toc25996)

[4.2 外部接口 22](#_Toc11396)

[4.2.1用户界面 22](#_Toc9941)

[4.2.2软件接口 22](#_Toc24872)

[4.2.3通信接口 22](#_Toc12687)

[4.3 内部接口 23](#_Toc4263)

[5 界面设计 24](#_Toc22310)

[5.1 用户界面设计规范 24](#_Toc6723)

[5.2 界面整体风格 27](#_Toc1872)

[5.2.1 页面风格 27](#_Toc28266)

[5.2.2 按钮风格 28](#_Toc31246)

[5.2.3 文字风格 30](#_Toc28987)

[5.2.4 表单设计 31](#_Toc26265)

[5.2.5 图像风格 33](#_Toc14908)

[6 系统出错处理设计 35](#_Toc5837)

[6.1 出错信息 35](#_Toc25246)

[6.2 补救措施 36](#_Toc25294)

[7 系统维护设计 36](#_Toc24260)

# 1 引言

## 编写目的

在本系统项目的前一阶段，也就是需求分析阶段中，已经将系统用户对本系统的需求作了详细的阐述，这些用户需求已经在上一阶段中的调研中获得，并在需求规格说明书中得到详尽的叙述和阐明。

本阶段已在系统的需求分析的基础上对系统进行概要设计,详细的介绍系统的各个模块，解决了系统需求的程序模块设计问题，如：如何把系统划分为各个模块、决定各个模块的接口、以及数据结构、模块结构的设计等。在以下的系统设计报告中将对本阶段中对系统的所有概要设计进行详细的说明。

在下一阶段的详细设计中，程序设计员可参考此系统设计报告，在系统设计对本系统所做的模块结构设计的基础上，对系统进行详细设计。在以后的软件测试以及软件维护阶段也可以参考此说明书，以便了解在系统设计过程中所完成的各模块设计结构，或在修改时找出本阶段设计的不足或错误。

## 背景

系统设计文档是在需求设计说明书的基础上进一步明确系统结构，它能详细的介绍系统的各个模块，方便成员更好的了解系统情况，使工作开展的各个过程合理有序。

## 定义

|  |  |
| --- | --- |
| 专门术语 | |
| MySQL | 关系型数据库管理系统 |
| 事务流 | 数据进入模块后可能有多种路径进行处理 |
| 主键 | 数据库表中的关键域，值互不相同 |
| 外部主键 | 数据库表中与其他表主键关联的域 |
| ROLLBACK | 数据库的错误恢复机制 |
| 缩写 | |
| 系统 | 若未特别指出，统指本《一锤定音》软件 |
| SQL | Structured Query Language(结构化查询语言) |
| ATM | Asynchronous Transfer Mode (异步传输模式) |
| UML | 统一建模语言、是一套用来设计软件蓝图的标准建模语言，是一种从软件分析、设计到编写程序规范的标准化建模语言 |
| UDP | User Datagram Protocol 是无连接的传输层协议 |
| 分布式代理 | 可隐藏服务器ip ，减少服务器的危险 |
| 服务器代理 | 可验证用户数据的正确性，以及安全性，进行处理 |
| 三级代理 | 减轻服务器压力，可实现智能作弊系统 |

## 参考资料

文档格式要求按照我国GB/T8567-1988国家标准和IEEE/ANSI830-1993标准规范要求进行。包括以下文件与书籍：

* 模拟法庭流程系统需求说明书
* 软件工程项目开发文档范例
* 软件工程国家标准文档
* 软件需求说明书编写规范
* 软件系统设计说明书编写规范
* 《软件项目管理》 朱少民，韩莹 编著，人民邮电出版社。
* 《软件项目管理》 Rajeev T Shandilya编著，科学出版社。
* 《JAVA web编程基础》任玉刚 编著，科学出版社。

# 2 系统功能设计

## 2.1 总体结构的分析与设计

### 2.1.1 设计目的和原则

设计采用模块化的设计方案，模块化是根据系统分析的逻辑模型设计应用软件系统的物理结构。在结构化系统总体结构设计中,整个应用软件系统由多个功能模块组成,通过合理的划分模块,正确处理模块之间与模块内部的联系,达到使整个系统具有良好的可用性、可读性、可修改性、易于调试和维护的目的。

**系统设计需要包含以下内容主要内容：**

（1）系统总体结构设计：

* 系统总体结构设计包括两方面的内容:
* 系统网络结构设计;
* 系统模块化结构设计。

（2）代码设计：

代码设计就就是通过设计合适的代码形式,使其作为数据的一个组成部分，用以代表客观存在的实体、实物与属性,以保证它的唯一性便于计算机处理。

（3）数据库(文件)设计：

根据系统分析得到的数据关系集与数据字典,再结合系统处理流程图,就可以确定出数据文件的结构与进行数据库设计。

（4）输入/输出设计：

输入/输出设计主要就是对以纪录为单位的各种输入输出报表格式的描述，另外,对人机对话各式的设计与输入输出装置的考虑也在这一步完成。

（5）处理流程设计：

处理流程设计就是通过系统处理流程图的形式,将系统对数据处理过程与数据在系统存储介质间的转换情况详细地描述出来。

（6）程序流程设计：

程序流程设计就是根据模块的功能与系统处理流程的要求,设计出程序模框图,为程序员进行程序设计提供依据。

（7）系统设计文档：

系统标准化设计就是指各类数据编码要符合标准化要求,对数据库(文件)命名、功能模块命名也要标准化。描述系统设计结果就是指系统设计说明书,程序设计说明书,系统测试说明书以及各种图表等,要将她们汇集成册,交有关人员与部门审核批准;

拟定系统实施方案设计就是在系统设计结果得到有关人员与部门认可之后，拟定系统实施计划,详细地确定出实施阶段的工作内容、时间与具体要求。

另外,为了保证系统安全可靠运行,还要对数据进行保密设计,对系统进行可靠性设计。包括:代码设计;数据库设计;输出设计;输入设计;处理流程设计;程序流程设计。

**软件项目设计时应采用以下四个原则：**

（1）系统性

从整个系统的角度进行考虑,系统的代码要统一,设计规范要标准,传递语言要尽可能一致,对系统的数据采集要做到数出一处、全局共享,使一次输入得到多次利用。

（2）灵活性

系统应具有较好的开放性与结构的可变性,采用模块化结构,提高各模块的独立性,尽可能减少模块间的数据偶合,使各子系统间的数据依赖减至最低限度。

（3）可靠性

可靠性就是指系统抵御外界干扰的能力及受外界干扰时的恢复能力。一个成功的管理信息系统必须具有较高的可靠性,如安全保密性、检错及纠错能力、抗病毒能力等。

（4）经济性

经济性指在满足系统需求的前提下,尽可能减小系统的开销。一方面,在硬件投资上不能盲目追求技术上的先进,而应以满足应用需要为前提;另一方面,系统设计中应尽量避免不必要的复杂化,各模块应尽量简洁,以便缩短处理流程、减少处理费用。

### 2.1.2 设计策略

采用结构化设计的基本思想:

将多个复杂的系统可以看成由许多相对独立部分组成，再经过层层划分，分解为若干组成部分进行设计。

结构化设计的基本思想的要点为：模块化、由顶向下,逐步求精、上层模块分解为下层模块。有三种不同的结构形式:顺序结构、分枝结构和循环结构、模块之间的通信只限于其直接上、下级模块,同级模块、其它上、下级模块不得直接通信。

结构化的设计策略应参考下表进行设计：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 联结方式 | 相互影响程度 | 容易修改性 | 可读性 | 通用性 |
| 数据耦合 | 弱 | 好 | 好 | 好 |
| 控制耦合 | 中 | 不好 | 不好 | 不好 |
| 公共耦合 | 强 | 不好 | 差 | 差 |
| 内容耦合 | 最强 | 差 | 差 | 差 |

### 2.1.3 系统设计UML图

（1）成员类图



图2.1 成员类图

（2）工具类图



图2.2 工具类图

（3）数据用例图



图2.3 数据用例图

（4）系统整体用例图



图2.4 系统整体用例图

（5）用户资料用例图



图2.5 用户资料用例图

（6）顺序图



图2.6 系统操作顺序图

（7）活动图



图2.7 系统操作活动图

（8）数据流图



图2.8 第一层数据流图



图2.9 第二层数据流图

## 2.2 模拟法庭模块设计

### 2.2.1 设计目的和原则

学生选择典型案例，模拟担当法官、公诉人、原告、被告、代理人、证人、书记员等各种角色，通过对所选择案件的模拟庭审全过程，使学生熟悉司法审判的实际程序，熟悉与案件相关的实体法和程序法，加深学生对有关程序法的准确理解，锻炼学生在真实的环境里从事法律实务的能力；通过“亲身、亲历”参与模拟法庭活动，培养和锻炼学生发现问题、分析问题和解决问题的能力，提高学生语言表达能力、组织协调能力，提升学生的专业技能和实践技能，同时还检验学生专业知识学习成果、法学理论掌握程度，学生反应能力与思辩能力等。

采用单一职责原则，即一个类只允许有一个职责，只有一个导致该类变更的原因。如果一个类具有多种职责，就会有多种导致这个类变化的原因，从而导致这个类的维护变得困难。在软件开发中，随着需求的不断增加，可能会给原来的类添加一些本来不属于它的一些职责，从而违反了单一职责原则。如果我们发现当前类的职责不仅仅有一个，就应该将本来不属于该类真正的职责分离出去。不仅仅是类，函数也要遵循单一职责原则。

### 2.2.2 设计思想

**（1）体系结构风格**

风格是一种带有倾向性的模式，本软件采用的是浏览/服务器（B/S）风格。具体结构为：浏览器/Web服务器/数据库服务器。

在这种架构下，用户工作界面通过WWW浏览器来实现，极少部分事务逻辑在前端实现，但是主要事务逻辑在服务器端实现，形成三层3-tier结构，WEB浏览器是客户端最主要的应用软件。用户通过客户端上的浏览器输入相应网址，即可打开软件《一锤定音》，使用模拟法庭功能创建法庭房间和加入法庭房间，选取自己所想扮演的法庭评审角色，也可以选取自己所需模拟的具体案件并开始模拟法庭的运作。系统安装、修改和维护全在服务器端解决，这种模式统一了客户端，将系统功能实现的核心部分集中到服务器上，简化了系统的开发、维护和使用。客户机上只要安装一个浏览器，服务器安装MySQL数据库。浏览器通过Web Server同数据库进行数据交互。这样就大大简化了客户端电脑载荷，减轻了系统维护与升级的成本和工作量，降低了用户的总体成本。

**（2）体系结构框架**

框架是特定应用领域问题的体系结构模式，本软件采用的是**J2EE体系结构框架**，J2EE是分层结构，中间的表示层、业务层、集成层包含应用程序构件，客户层和资源层处于应用程序的外围。

客户层：用户通过客户层与系统交互，本软件采用的是浏览/服务器（B/S）风格，因此客户层为Web浏览器客户端。

资源层：在本软件中，资源层为服务器上的数据库。

表示层：用户通过表示层来访问应用程序。在本软件中，表示层由用户界面和运行于Web服务器上的过程组成。

业务层：业务层负责确认和执行业务规则和事务，并管理从资源层加载到应用程序高速缓存中的业务对象，主要表现为确认软件各项功能是否正确运行。

集成层：负责建立和维护与数据源的连接，主要体现在通过JDBS与数据库进行通信。

**（3）设计模式**

设计模式为软件系统的子系统、构件或者构件之间的关系提供一个精练之后的解决方案，描述了在特定环境下，用于解决通用软件设计问题的构件以及这些构件相互通信时的各种结构。本软件采用的是观察者模式。

观察者模式中关键的对象为主题和观察者，它定义了对象间的一种一对多的依赖关系，当主题的状态发生改变时，所有依赖于它的观察者都得到通知，并被自动更新。在本软件中，所有使用模拟法庭功能的用户都是主题，同时也是其他用户的观察者。观察者模式降低了目标与观察者之间的耦合关系，两者之间是抽象耦合关系，符合依赖倒置原则。并且在目标与观察者之间建立了一套触发机制。

### 2.2.3 基本处理流程



图2.10 模拟法庭功能数据流图



图2.11 模拟法庭功能用例图



图2.12 模拟法庭功能顺序图



图2.13 模拟法庭功能类图



图2.14 模拟法庭功能活动图

### 2.2.4 数据字典

**（1）数据项**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据项含义 | 数据类型 | 取值范围 | 取值含义 |
| A1 | 用户昵称 | 用户在软件中展示的昵称 | Char |  | 用户自拟 |
| A2 | 用户ID | 每位用户的唯一标识 | Integer | 0~999999 | 用户的唯一ID，自适应递增 |
| A3 | 用户类型 | 整型 | 1位 | ‘0’或‘1’ | ‘0’为管理员，  ‘1’为普通用户 |
| A4 | 案件标题 | 每个案件的标题 | Char |  | 对案件的分类及梗概 |
| A5 | 案件角色 | 案件中每个人物的姓名 | Char |  | 标识不同角色 |
| A6 | 案件信息 | 每个案件的案情 | Char |  | 对案情的描述 |
| A7 | 案件角色信息 | 案件中每个人物的简介 | Char |  | 标识不同角色 |
| A8 | 庭审角色 | 庭审时法庭中的各项角色 | Char |  | 发挥每位庭审角色特定职能 |
| A9 | 房间信息 | 案件信息及用户信息 | Char |  | 记录此房间的各项信息 |
| A10 | 庭审结果 | 由法官宣布的最终评审结果 | Char |  | 庭审结果显示本案获胜方 |

**（2）数据流**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据流名 | 组成 | 来源 | 去向 |
| F1 | 用户信息 | A1+A2+A3 | E1+E3+E4 | C1 |
| F2 | 案件 | A4+A5+A6+A7 | E2 | C2 |
| F3 | 角色 | A6+A7 | E2 | C2 |
| F4 | 房间 | A1+A2+A3+A4+A5+A6+A7 | E1+E2+E3+E4 | C1+C2 |
| F5 | 庭审 | A1+A2+A3+A4+A5+A6+A7+A8+A9 | E1+E2+E3+E4 | C1+C2 |

1. **数据存储**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 数据项名称 | 组成 | 组成方式 | 查询要求 | 简述 |
| C1 | 用户表 | A1+A2+A3 | A1升序 | 立即 | 暂无 |
| C2 | 案件表 | A4+A5+A6+A7 | A4升序 | 立即 | 暂无 |
| C3 | 庭审表 | A1+A2+A3+A4+A5+A6+A7+A8+A9 | A1升序 | 立即 | 暂无 |

1. **外部输入**

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 数据项名称 | 输出数据流 | 简述 |
| E1 | 用户 | F1+F4+F5 | 暂无 |
| E2 | 案件信息 | F2+F3+F4+F5 | 暂无 |
| E3 | 注册系统 | F1 | 暂无 |
| E4 | 登录系统 | F1 | 暂无 |

# 3 数据库设计

## 3.1 数据库关系E-R图



图3.1 E-R图

## 3.2 数据库表格设计

Create table user(

user\_id integer primary key auto\_increment not null ,

username varchar2(50) not null,

sex varchar(2) DEFAULT '男' check(sex in('男','女')),

phone varchar2(10),

address varchar(20),

code varchar(15)，

user\_outline varchar2(100)

);

表3.1 用户表数据结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 说明 | 数据类型 | 长度 | 是否为空 | 特殊要求 |
| user\_id | 用户账号 | Integer | 9 | 否 | 自适应增加 |
| username | 用户昵称 | Varchar2 | 50 | 否 | 无 |
| sex | 用户性别 | Varchar2 | 2 | 是 | 无 |
| phone | 用户电话 | Varchar2 | 10 | 是 | 无 |
| code | 用户密码 | Varchar2 | 15 | 否 | 必须含有字母数字且大于8位 |
| address | 用户所在地区 | Varchar2 | 20 | 是 | 无 |
| user\_outline | 用户简介 | Varchar2 | 100 | 是 | 无 |

create table room(

room\_id varchar2(10)primary key not null,

room\_name varchar(5),

case\_id varchar(10),

total number(8),

func varchar(10) DEFAULT '普通房间' check(func in ('热门案件模拟房间','最新模拟案件房间','冷门模拟案件房间')),

foreign key (case\_id) references cases(case\_id)

);

表3.2房间表数据结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 说明 | 数据类型 | 长度 | 是否为空 | 特殊要求 |
| room\_id | 房间号 | Varchar2 | 10 | 否 | 无 |
| room\_name | 房间名称 | Varchar2 | 5 | 否 | 无 |
| case\_id | 案件编号 | Varchar2 | 10 | 是 | 只能填“男”或“女” |
| total | 总数量 | Varchar2 | 8 | 是 | 无 |
| func | 功能 | Varchar2 | 10 | 否 | “普通房间”“热门案件模拟房间”“冷门模拟案件房间”“最新模拟案件房间” |

create table cases(

case\_id varchar2(10) primary key not null,

case\_name varchar(10),

case\_num number(8),

case\_category varchar(10),

case\_outline varchar2(500),

);

表3.3 案件表数据结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 说明 | 数据类型 | 长度 | 是否为空 | 特殊要求 |
| case\_id | 案件编号 | Varchar2 | 10 | 否 | 无 |
| case\_name | 案件名 | Varchar2 | 10 | 否 | 无 |
| case\_num | 案件被模拟数 | Varchar2 | 8 | 是 | 案件总模拟数 |
| case\_category | 案件类型 | Varchar2 | 10 | 是 | “民事案件”和“刑事案件”和“经济案件”和“非诉讼案件”和“其它” |
| case\_outline | 案件简述 | Varchar2 | 500 | 否 | 无 |

create table role(

role\_name varchar2(10) primary key not null,

employment varchar2(10),

room\_id varchar2(10),

case\_id varchar2(10),

role\_outline varchar2(500),

foreign key (case\_id)references cases(case\_id),

foreign key (room\_id)references room(room\_id)

);

表3.4 角色表数据结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 列名 | 说明 | 数据类型 | 长度 | 是否为空 | 特殊要求 |
| role\_name | 角色名称 | Varchar2 | 10 | 否 | 无 |
| employment | 职业 | Varchar2 | 10 | 否 | 无 |
| room\_id | 房间号 | Varchar2 | 10 | 是 | 无 |
| case\_id | 案件类型 | Varchar2 | 10 | 是 |  |
| role\_outline | 角色简述 | Varchar2 | 500 | 否 | 无 |

# 4 接口及过程设计

## 4.1 用户接口

（1）用户接口需具有可视性：

* 所有控件应该是可见的。
* 提供及时反馈以表示控件正在响应。
* 控件有外观应该体现它的功能-控件的使用目的。

（2）系统开发人员应该使用已出版的界面设计标准和指导原则：

* 尽量保持一致性为老用户提供快捷键提供有效反馈
* 设计完整的对话过程提供简单的错误处理机制允许撤销动作
* 提供控制的内部轨迹减轻短期的记忆负担

（3）编制对话设计文档：

* 与其他系统活动同时进行
* 基于需要用户交互的输入和输出
* 用于定义菜单层次
* 允许用户导航到每个对话框-提供整个系统结构。设计标准窗体和浏览器窗体的指导原则，每个对象可能需要几个Windows窗体。

（4）广泛应用的标准表单：

* 标准窗体: Java
* 浏览器: HTML, JavaScript, ASP, JavaServlets

（5）确定窗体对象及相关数据域一用原型工具构造窗体

* 窗体布局与格式化的一致性
* 标题、标签和标志
* 字体大小、亮度对比和颜色
* 文本框和按钮的次序
* 数据键控及数据输入(使用标准控件)
* 文本框、列表框、组合框等等
* 导航和支持控件
* 帮助支持-教程、索引、上下文敏感

## 4.2 外部接口

### 4.2.1用户界面

1. 系统的总体结构采用菜单调用窗体，利用窗体来对信息进行管理、数据加工、信息。
2. 界面菜单：需要统一的菜单风格，包括菜单图片、颜色，菜单栏目的划分，各种功能的菜单标题等。
3. 屏幕布局：对于不同分辨率的网页进行适配
4. 操作方式：触控操作与鼠标操作
5. 色彩方案：清新
6. 快捷方式：快捷方式，显示快捷方式，自定义快捷方式
7. 导航方式：无导航方式
8. 错误显示标准：弹窗显示错误

### 4.2.2软件接口

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口标识 | 简要描述 | 所需服务 | 通信定义 | 需求来源 | 请求 | 返回 |
| MC-01 | 与数据库数据互通 | Mysql数据库 | 共享数据 | 登录服务 | 用户id  用户密码 | 查询数量  用户信息 |
| MC-02 | 语音交互接口 | Navigator.getUserMedia() | 交换数据 | 对话服务 | 调用服务 | 语音信息 |
| MC-03 | 与服务器交互 | Centos系统服务 | 交换数据 | 系统服务 | 无 | 无 |
| MC-04 | 网站托管接口 | TomCat | 交换数据 | 系统服务 | 无 | 无 |

### 4.2.3通信接口

1. 调用TCP/IP通信协议接口
2. 调用GSM/CDMA无线通信协议接口
3. 调用SMS短消息通信协议接口
4. 调用联通网关通信协议接口
5. 调用防火墙通信接口
6. 调用路由器通信接口
7. 调用交换机通信接口
8. Web浏览器：IE8.0以上或Chrome内核浏览器
9. 数据传输速率：512KB/S以上
10. 加密与通信：md5加密用户信息

## 4.3 内部接口

接口的数据一般都采用JSON格式进行传输,内部接口的变化遵循以下原则：

* 数据的变化，比如增加了旧版本不支持的数据类型
* 参数的变化，比如新增了参数
* 接口的废弃，不再使用该接口了
* 每个接口有各自的版本，一般为接口添加个version的参数。
* 整个接口系统有统一的版本，一般在URL中添加版本号

内部接口概要设计如下表所示：

|  |  |  |
| --- | --- | --- |
| 编号 | 名称 | 作用 |
| 1 | void updateUserState() | 更新用户状态 |
| 2 | Int getUserState() | 获取用户状态 |
| 3 | void restartSystem() | 重启系统 |
| 4 | void reportResult() | 反馈结果 |

# 5 界面设计

## 5.1 用户界面设计规范

**（1）用户界面设计原则**

本系统坚持图形用户界面（GUI）设计原则，界面直观、对用户透明：用户接触软件后对界面上对应的功能一目了然、不需要多少培训就可以方便使用本应用系统。

**（2）一致性原则**

在界面设计中应该保持界面的一致性。一致性既包括使用标准的控件,也指使用相同的信息表现方法,如在字体、标签风格、颜色、术语、显示错误信息等方面确保一致。

**（3）布局合理性原则**

应注意在一个窗口内部所有控件的布局和信息组织的艺术性,使得用户界面美观。   
 在一个窗口中按tab键,移动聚焦的顺序不能杂乱无章,tab 的顺序是先从上至下，再从左至右。一屏中首先应输入的和重要信息的控件在tab顺序中应当靠前,位置也应放在窗口上较醒目的位置。布局力求简洁、有序、易于操作。

**（4）鼠标与键盘对应原则**

应遵循的是可不用鼠标的原则:应用中的功能只用键盘也应当可以完成,即设计的应用中还应加入一些必要的按钮和菜单项。

但是,许多鼠标的操作,如双击、拖动对象等,并不能简单地用键盘来模拟即可实现。例如在一个列表框中用鼠标双击其中一项可以表示选中该项内容。为了用键盘也能实现这一功能,必须在窗口中定义一个表示选中的按钮,以作为实现双击功能的替代（或其它方式）。又如在一个窗口中有两个数据窗口,可以用鼠标从一个数据窗口中将一项拖出然后放到另一个中。如果只用键盘,就应当在菜单中设置拷贝或移动的菜单项。

**（5）向导（WIZARD）使用原则**

对于应用中某些部分的处理流程是固定的，用户必须按照指定的顺序输入操作信息，为了使用户操作得到必要的引用应该使用向导，使用户使用功能时比较轻松明了，但是向导必须用在固定处理流程中，并且处理流程应该不少于3个处理步骤。

**（6）系统响应时间**

系统响应时间包括两个方面：时间长度和时间的易变性。用户响应时间应该适中，系统响应时间过长，用户就会感到不安和沮丧，而响应时间过短有时会造成用户加快操作节奏，从而导致错误。系统响应时间的易变性是指相对于平均响应时间的偏差。即使响应时间比较长，低的响应时间易变性也有助于用户建立稳定的节奏。因此在系统响应时间上坚持如下表5.1和表5.2原则：

表5.1 响应时间长度界面设计原则

|  |  |
| --- | --- |
| 响应时间长度 | 界面设计 |
| 0-10秒 | 鼠标显示成为沙漏 |
| 10到18秒 | 由微帮助来显示处理进度 |
| 18秒以上 | 显示处理窗口，或显示进度条 |
| 一个长时间的处理完成时 | 应给予完成警告信息 |

表5.2 响应时间易变性的界面设计原则

|  |  |
| --- | --- |
| 响应时间的易变性 | 界面设计 |
| 用户感觉不到 | 不考虑 |
| 用户稍微感觉到 | 由微帮助提供易变性说明 |
| 容易性大而且时间绝对差别大 | 显示易变性提示 |

**（7）用户帮助设施**

常用的帮助设施有两种：集成的和附加的。集成的帮助设施一开始就是设计在软件中的，它与语境有关，用户可以直接选择与所要执行操作相关的主题。通过集成帮助设施可以缩短用户获得帮助的时间，增加界面的友好性。附加的帮助设施在系统建好以后再加进去的。通常是一种查询能力比较弱的联机帮助。

本系统提供这两种帮助设施，设计和实现时遵循以下原则：

1. 进行系统交互时，提供部分帮助功能，即：提供主要操作的帮助；
2. 用户可以通过帮助菜单、F1键和帮助按钮（如果有的话）访问帮助；
3. 表示帮助时根据需要提供三种方式的选择：另一个窗体、微帮助和指出参考某个文档；
4. 用户如何回到正常交互方式有两种选择：返回键和功能键；
5. 帮助信息的构造：采用分层式帮助；
6. 微帮助提供：由状态栏提供，或控件上的提示文本。

**（8）出错信息和警告**

出错信息和警告是指出现问题时系统给出的坏消息，本系统对于出错信息和警告应该遵循以下原则：

1. 信息以用户可以理解的术语描述；
2. 信息应提供如何从错误中恢复的建设性意见；
3. 信息应指出错误可能导致那些不量后果，以便用户检查是否出现了这些情况或帮助用户进行改正；
4. 信息应伴随着视觉上的提示，如特殊的图像、颜色或信息闪烁；
5. 信息不能带有判断色彩，即任何情况下不能指责用户。

**（9）命令交互**

由于本系统用户是网页用户，故本系统不提供命令交互。

**（10）一般交互原则**

本系统一般交互遵循以下原则：

1. 一致性：菜单选择、数据显示以及其它功能都应使用一致的格式；
2. 提供有意义的反馈；
3. 执行有较大破坏性的动作前要求确认；
4. 在数据录入上允许取消大多数操作；
5. 减少在动作间必须记忆的信息数量；
6. 在对话、移动和思考中提高效率；
7. 允许用户非恶意错误，系统应保护自己不受致命作物的破坏；
8. 按功能对动作分类，并按此排列屏幕布局，设计者应那里提高命令和动作组织的内聚性；
9. 提供语境相关的帮助机制。

**（11）信息显示原则**

本系统信息显示遵循以下原则：

1. 只显示与当前用户语境环境有关的信息；
2. 不要用数据将用户包围，使用便于用户迅速吸取信息的方式表现信息；
3. 使用一致的标记、标准缩写和可预测的颜色，显示信息的含义应该非常明确，用户不必再参考其它信息源；
4. 产生有意义的出错信息；
5. 使用缩进和文本来辅助理解；
6. 使用窗口分隔控件分隔不同类型的信息；
7. 高效地使用显示器的显示空间。

**（12）数据输入原则**

本系统数据输入遵循以下原则：

1. 尽量减少用户输入动作的数量；
2. 维护信息显示和数据输入的一致性；
3. 交互应该时灵活的，对键盘和鼠标输入的灵活性提供支持；
4. 在当前动作的语境中使不合适的命令不起作用；
5. 让用户控制交互流，用户可以跳过不必要的动作、改变所需动作的顺序（如果允许的话）以及在不退出系统的情况下从错误状态中恢复；
6. 为所有输入的动作提供帮助；
7. 消除冗余输入。可能的话提供缺省值、绝不要让用户提供程序中可以自动获取或计算出来的信息。

## 5.2 界面整体风格

### 5.2.1 页面风格

页面总体风格的主色调为蓝色、并且运用统一的浅色背景图、添加标志、采用适用的图象分辨率、分帧等。

另外，对于不同的页面，对可能出现的页面进行分类，并分别描述各类页面应包含的内容。分类的方式可以按照页面层次分，也可以按照页面种类分，分类如下：

按层次分：

* 一级页面——模拟法庭房间展示界面
* 二级页面——加入、查询、创建、删除模拟法庭房间界面
* 三级页面——查询案件、法律法规界面
* 四级页面——其它深层次页面

按种类分：

* 主页
* 用户注册页
* 查询页
* 查询结果页
* 出错提示页
* 用户信息提交成功页
* 帮助页
* 其它

### 5.2.2 按钮风格

**（1）行为召唤按钮风格**

在设计中行为召唤按钮通常会提示用户注册/立即购买等。在产品设计中如果强烈建议用户应该做的事情应该使用CTA按钮，并且使用圆形按钮，更引入注目。

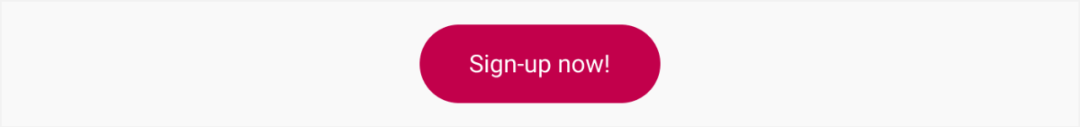


图5.1 行为召唤按钮

**（2）主要操作按钮风格**

主按钮应该是一个强大的视觉指示器，可以帮助用户完成他们的旅程。主按钮应该在用户可能想要“下一步”、“完成”、“开始”等的情况下使用。

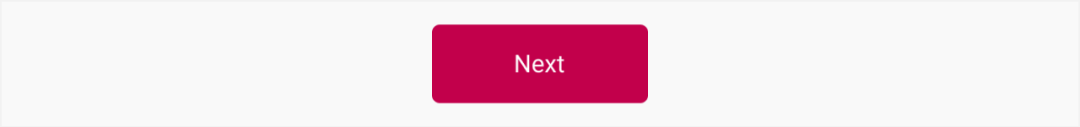


图5.2 主要操作按钮

**（3）次要操作按钮风格**

从“返回”的辅助按钮到“下一步”的主按钮，或从“编辑”的辅助按钮到“保存”的主按钮。辅助按钮是我们为用户提供的主要操作的备选方案。



图5.3 次要操作按钮风格

**（4）三级操作按钮风格**

三级按钮通常用于其他操作，可能暂时不是用户想要做的事情。比如“添加朋友”、“查找”、“编辑”或“删除”之类的内容，前提是它们不是主要操作，使用较小或较不突出的按钮样式。

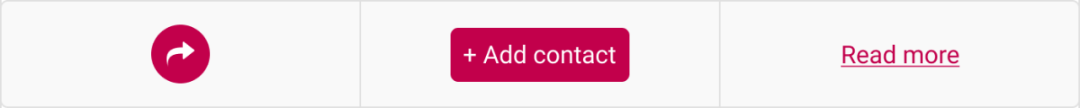


图5.4 三级操作按钮

**（5）按钮颜色选择**

不同功能对应的按钮颜色应该具有明显的对比度，以便用户较为清晰分辨出不同的功能进行对应的操作。

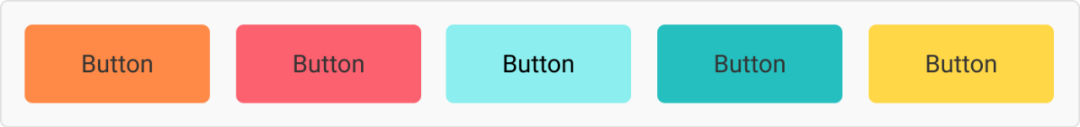


图5.5 不同颜色按钮

另外，在选择颜色时应该考虑色彩心理学。红色按钮多用于删除，黄色按钮多用于警示等。

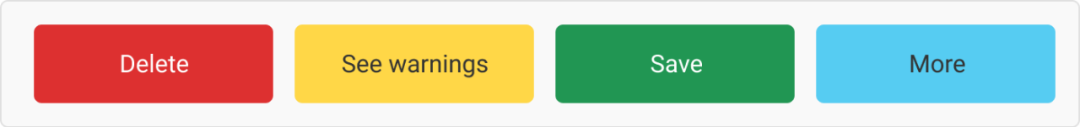


图5.5 “删除”、“警告”、“保存”和“更多”按钮

**（6）图标按钮风格**

由于图标按钮没有标签，只是一个图标，因此采用具体化图标设计按钮。

****

图5.6 图标按钮风格

**（7）文本链接设计**

单一的文本链接无法实例化地表示功能的具体操作，应添加具体功能图标，表示文本链接的跳动功能，便于提示用户操作。



图5.7 文本链接按钮设计

**（8）超链接按钮设计**

使用蓝色对文本链接设置超链接，容易识别，即便是色盲的人也通常能识别出蓝色超链接按钮。



图5.8 超链接按钮设计

### 5.2.3 文字风格

文字是UI设计中一个重要的组成元素，文字使用的好坏会极大影响产品的用户体验，为了使用户能够高质量地使用我们的产品，文字风格设计应该建立视觉层级。一个页面中不同元素的重要性是不一样的，有优先级之分。文字也是一样，有些文字比较重要，我们更希望用户看到，有些文字比较鸡肋我们不希望用户看到。为了达到这个目的，界面的文字应该建立视觉层级，将文字大致分为三类：主标题，次标题和正文。

1. 主标题

主标题是对于整个页面内容的一个总结，合格的主标题用户看了一眼就应该知道这个页面的大致内容。主标题是用户进入一个页面第一眼就应该看到的文字。主标题应该用字号要足够的大，要加粗，这样才可以更好的吸引用户的注意力。此外，为了更好的节约用户时间，主标题应该简练。根据Jacob Nielsen的一项研究表明，主标题5-6个单词(英文状态下)最合适，最长不要达到一句话的长度。如采用图片文字、或方正正中黑。



图5.9 方正正中黑字体

我们强调要突出主标题来吸引用户注意力，但是不要过度突出。因为用户对于具象元素(插画，icon,图像或摄影图等)的感知能力远比文字要强得多。如果我们想宣传一款产品，那么最好的方案就是直接给用户展示产品图片。文字和图片搭配使用的时候，文字起到的只是辅助说明的作用。我们不能过度放大主标题的尺寸造成对图片的遮盖，这样是本末倒置的。

1. 次标题

次标题的要求和主标题类似，都要求文字简洁干练，概括性强。和主标题一样，次标题也要进行加粗处理，当然为了和主标题区分，字号要稍微小点。

1. 正文

正文是提供详细说明和解释的文字，从页面层级的角度来说重要性要低于主标题和次标题。用长文字来展示特定内容的详细信息或者用户不太熟悉的内容(需要用户仔细阅读)。字体采用宋体、字号为小四号。

### 5.2.4 表单设计

**（1）表单构成**

一个完整的表单由类别、标签、表单字段、占位符、帮助信息、操作按钮、及时验证等元素构成。各类元素如图5.10所示。

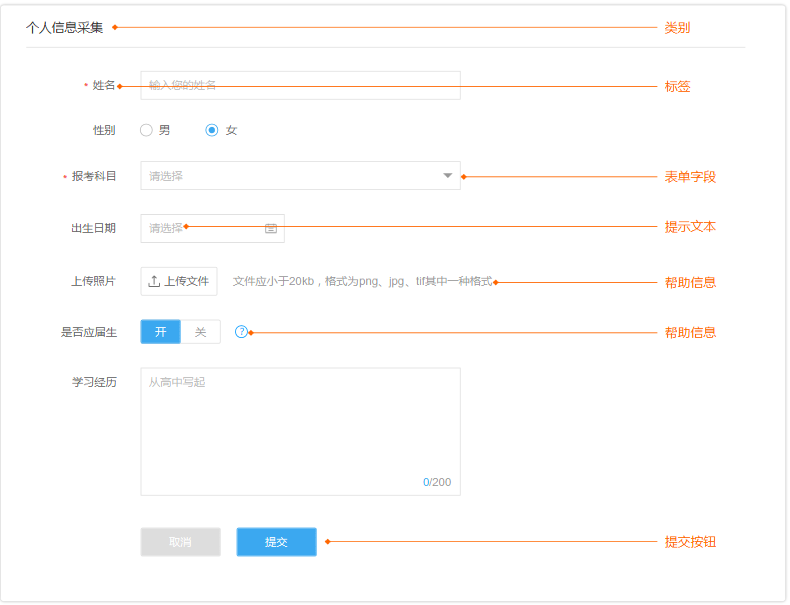


图5.10 表单各类元素显示

**（2）表单外部标签对齐方式**

采用右对齐方式，右对齐标签适用于表单占用垂直屏幕空间有限，同时要求快速完成表单填写，是Web端主流的表单标签样式。右对齐标签优点是输入框联系较为紧密，并且减少占用垂直空间，缺点是标签可读性不强，降低快速浏览完表单的效率。

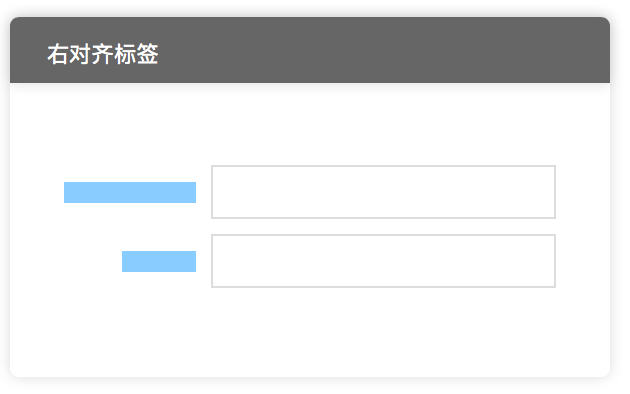


图5.11 标签对齐方式

**（3）表单要简短**

表单在设计过程中只保留必要的字段，做到不重复；删掉不必要的字段能减少用户完成表单耗费的精力、提高表单提交率、提高表单与用户之间的对话效率。

**（4）合理组织信息**

表单设计时保证视觉流遵循从左到右、从上到下的阅读习惯，提高阅读速度；对于较为复杂、填写项多的表单，如果不对信息进行组织，很容易显得繁杂混乱，让用户在一开始就产生抵触的情绪，甚至选择放弃，所以建议设计者在设计表单时多从信息的组织方面考虑是否采用分步表单或者采用分组表单。合理有层次的组织信息，可以利用框线、空间间隔、颜色的不同，按照不同信息的类别、属性和相关性进行区块的划分，达到信息的合理组织。

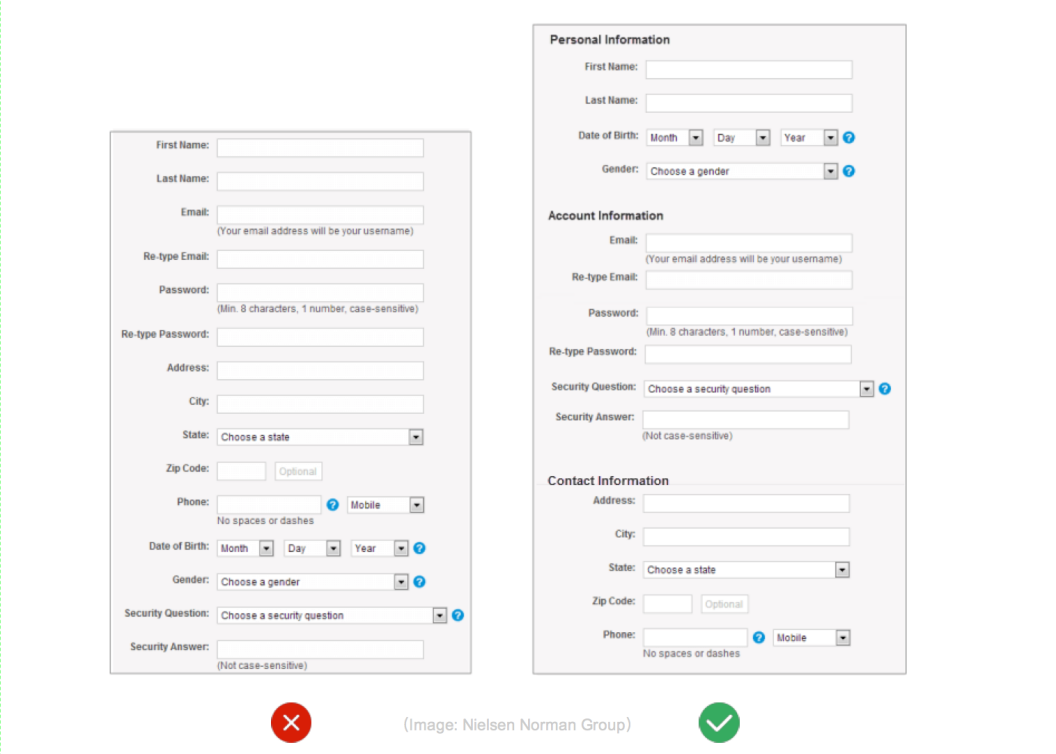


图5.12 表单非合理组织与合理组织对比图

**（5）合理运用颜色**

由设计心理学色彩对视觉的影响我们可以知：通知为蓝色，警告为黄色，错误为红色，成功确认为绿色。



图5.13 颜色使用规则

### 5.2.5 图像风格

（1）图片大小限制

图片大小不超过2MB，大型图片存储在云服务器上。

（2）图片格式

采用标准的.gif、.jpg、.png等格式，不能采用.bmp格式。

（3）图片命名

先分类后命名，如bg-01，bg-02，banner-01，banner-02，button-open，button-close等。

# 6 系统出错处理设计

## 6.1 出错信息

用一览表的方式说明每一种可能出错的情况出现时，系统输出信息的形式、含义级处理的方法。由于输入信息不符合规范(如:已有房间数量为负数等)，称之为软错误;由于硬件方面的错误(如:网络传输超时、硬件出错等)，称之为硬错误; 对于一些关键的操作(如:删除操作)，应该提供提示确认机制;对于数据、测试文档，都是要提供相应的保密措施设置;

采用错误提示窗口向用户提示错误，并友好地处理错误。例如,用户登录失败时，根据失败原因进行提示，用户输入不正确时，进行适当提示。

|  |  |  |
| --- | --- | --- |
| 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 | 管理员权限设置错误 |
| 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 |
| 友情链接网址维护或者其他错误无法打开 |

## 6.2 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/修改操作中及时对输入数据进行验证，分析错误的类型，并且给出相应的错误提示语句，传送到客户端的浏览器上;
* 对于硬错误，错误类型不较少而且比较明确，所以可以在可能出错的地方中输出相应的出错语句，并将程序重置，最后返回输入阶段;
* 定期建立数据库备份,一旦服务器数据库被破坏，可以使用最近的一份数据库副本进行还原。
* 为防止服务器故障，预备另外一台服务器,只要主服务器出现故障,可以迅速启动预备服务器运行系统。
* 恢复及在启动技术说明将使用的恢复再启动技术，是软件从故障点恢复执行或是软件从头开始重新开始运行的方法。

# 7 系统维护设计

此项主要是对服务器上的数据库以及相关文件进行维护。数据库维护：我们使用数据库的维护功能机制，比如，定期备份数据库，定期检测数据库的一致性，定期查看操作日志等；文件方面：对于过期文件等，定期删除相关文件，减少数据量。

（1）系统应用程序维护

**出错输出信息：**

* 数据库无法连接：请联系系统管理员；
* 用户名错误：无此用户；
* 用户密码错误：密码错误，请检查你的密码；
* 验证码错误：请输入正确的验证码；
* 管理员用户名错误：无此用户；
* 管理员密码错误：密码错误，请检查你的密码；
* 用户未登录：请您登陆后在进行访问；
* 网页链接错误：请刷新后重试；
* 文件/图片链接错误：请刷新后重试。

**出错处理对策：**

* 用户未登录，系统输出提示信息“请先登录再进行操作”，并提供超链接返回登录页面。
* 用户未创建新模拟法庭房间，系统输出提示信息“请创建新模拟法庭房间”，并提供返回到创建页面。
* 网页、文件与图片的链接错误，系统输出提示信息“请刷新网页并重试”，若仍无效，则返回上级页面并生成错误报告提交至系统管理员。
* 现有应用软件系统功能模块的维护，包括对各功能模块运行中发现的BUG进行修改完善。
* 协助用户使用，听取各方面操作人员的修改意见，对所有意见汇总整理，对在实施维护范围内的有效意见予以修改完善。

（2）安全保密设计

前台程序的用户数量多，地点比较分散，应限制非法用户对本系统的使用。拟采用WEB服务器端对网站安全性的实现手段。主要使用限制客户的IP地址的方法。后台管理程序的使用权控制通过页面认证表单来实现。也可以利用WEB服务器的安全认证来实现。如果对安全性的要求很高的话，可以通过在服务器、客服机中安装数字证书来提供安全性。本系统的后台权限验证功能目的是防止管理员的错误操作。

（3）数据维护

① 制定出完备的备份方案，每日进行应用软件系统的数据备份，每周检查备份是否成功，每月初将上月的备份文件存储到移动存储介质当中。含硬盘备份及移动存储介质的备份。将在意外情况下数据丢失的影响最小化。

② 维护安全性，使数据库不被无权人员随意打开查看。

③ 合理管理和组织数据库的存储空间，每周定期压缩和整理数据库，使数据库保持有效性和简洁性，提高存取速度。

④ 数据库升级安装与配置，根据需要，如果用户数据库升级，完成数据库系统的升级安装和配置。

⑤ 应急恢复系统：在1小时内完成应用及数据库的恢复工作。

（4）软件维护

① 为满足同时在线需求，每周进行压力测试，应确保至少10万人能同时在线，以满足同时在线1万人的最低指标。同时数据最大储存不少于200Mb。

② 维护系统初始化参数，确保各功能模块的正常运行。

③ 根据用户需求，进行人员注册、注销权限、人员调整等操作。用户执行增加删除等操作时，数据库响应时间要求在0.8秒之内。

④ 每日一次检查各功能模块的使用情况和数据库状况，确保各功能模块信息的有效性和运行的高效性。

（5）系统检查

系统定期巡检是指对应用系统实施现场检查，及时发现系统运行中出现的隐患，减少系统发生故障的概率，保证系统的稳定运行。在系统巡检过程中发现的问题，依据故障分类标准，由小组开发人员进行相应的故障排除。本项目中巡检服务主要指软件检查，内容包括：数据完整性检查、性能指标检查、功能正确性检查、软件可用性与安全性检查等。