**财税信息统计管理平台 --设计说明书**

# 第一章 前言

## 1.1 遵循原则

**设计原则**

分析方法：客户端以原型方法代替用例分析法驱动功能分析，体验更直观；服务端以数据流分析法为主。

模块独立性原则：避免重复定义，处理完整，输入输出定义明确；边界设计原则：可扩展部分接口化，客户端脚本与服务端Java代码的接口要符合命名习惯，接口设计友好。

扩展原则：可变部分分离成为配置文件，以便系统具有较好的灵活性。客户端功能插件化，使功能的扩展与维护简便、合理。鉴于人力与时间都比较紧迫的状况，可扩展部分也要设计出来，方便后期进行功能增强。

性能原则：客户端尽量减少访问服务端获取大量数据的次数，降低通讯压力；客户端缓存数据要进行合理规划，对数据增加访问安全限制的同时，要尽量少的在客户端缓存与呈现无关的数据。

**命名规则**

遵循普遍被Java或JavaScript业界所接受的编码方式与命名规则。客户端JavaScript与服务端Java代码均要有完整接口注释（类功能、方法功能、返回值说明与参数列表说明），有复杂或难以理解的代码出现时，要有详细的代码注释。

1.2 系统概述

通过财税信息统计管理平台，企业可以有效收集、存储、加工和传输会计信息，实现对企业经济活动进行全面地控制。会计业务电子化、移动化，快速高效，大大提高了工作效率。

# 第二章 总体设计

## 2.1 界面效果

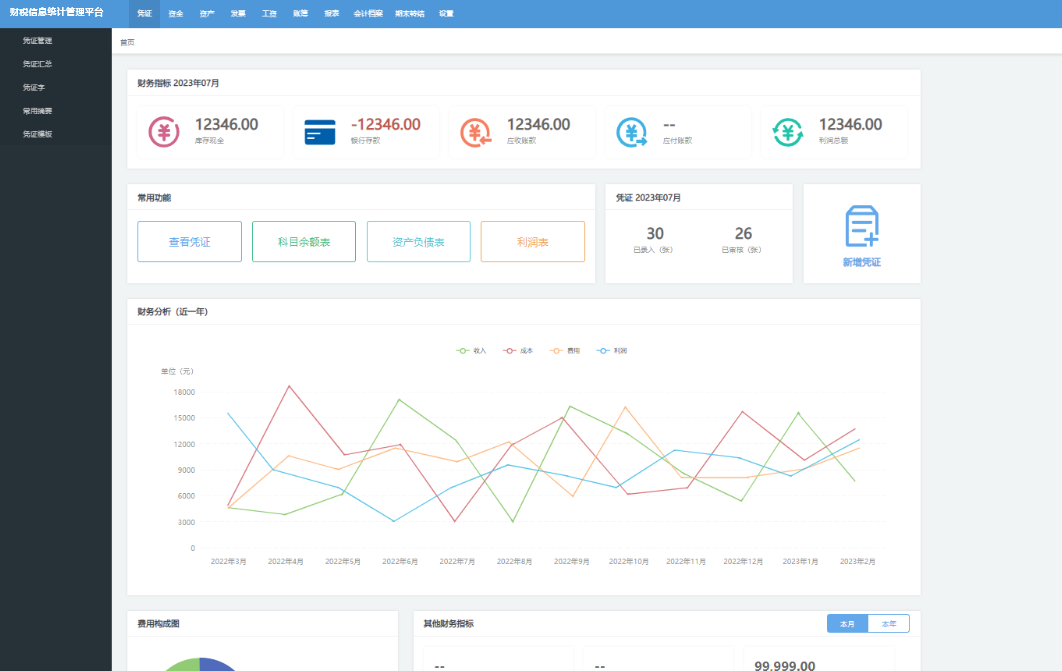
**2.1.1 登录系统**

在浏览器输入系统地址，点击【enter】，进入系统登陆页面，用户依次输入自己的账号、密码、校验码后，点击【登录】按钮，进入财税信息统计管理平台。



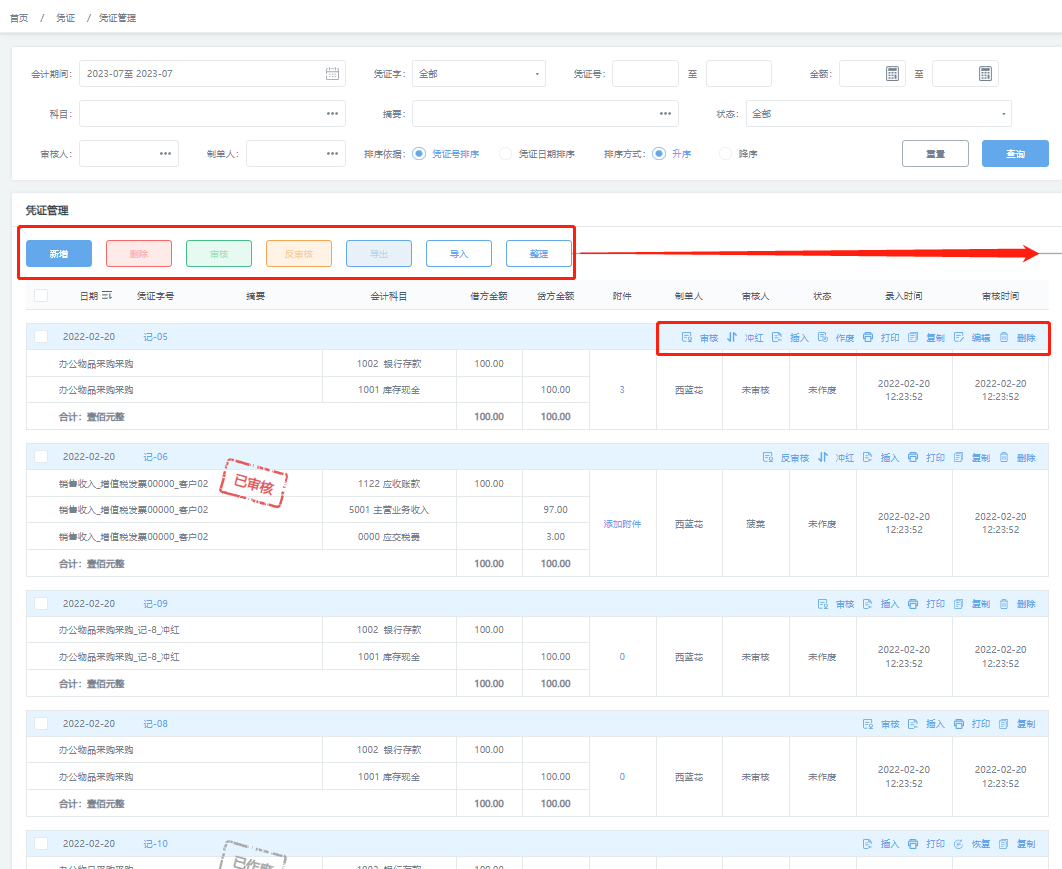
**2.1.2 首页**

登陆后进入系统首页，如下图所示，展示数据指标、数据分析及常用功能等。



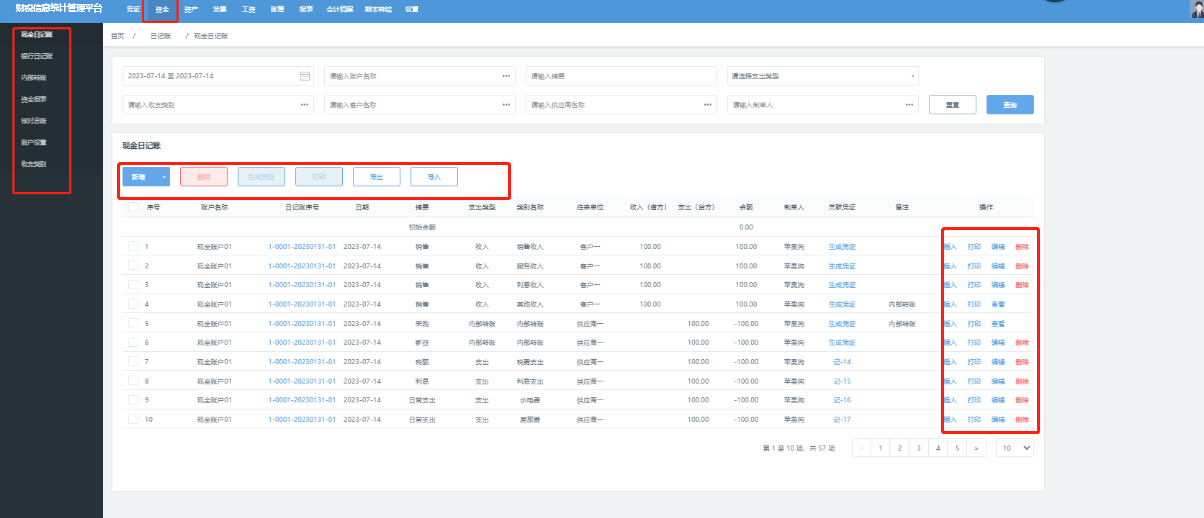
**2.1.3 会计凭证**

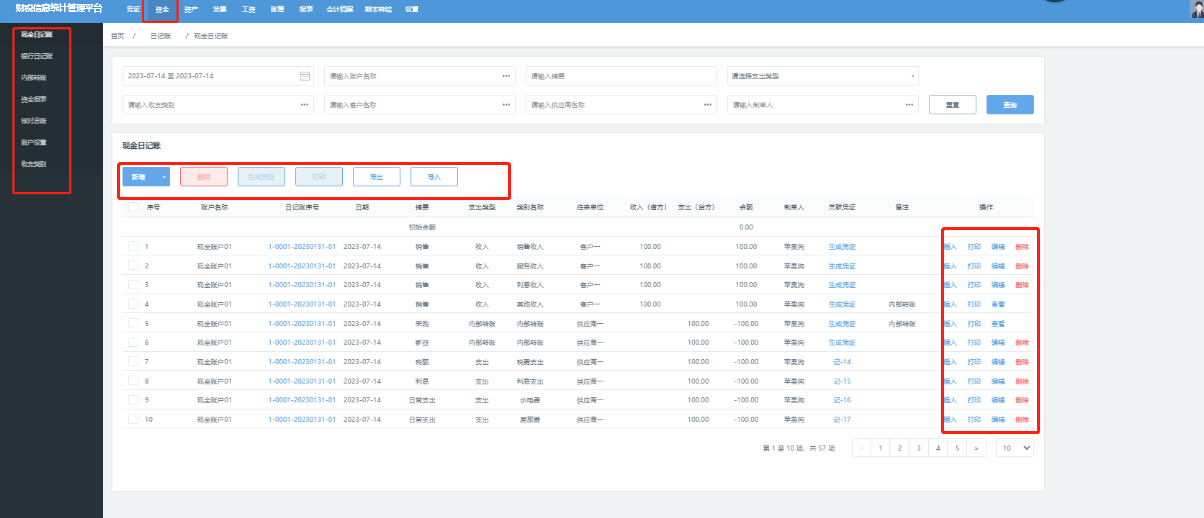
如下图，点击顶部菜单栏凭证后，在左侧菜单栏点击凭证管理，可进入凭证管理列表页。会计凭证既是记录经济业务发生或者完成情况的书面证明，也是登记账簿的依据。除对会计凭证进行增删改查外，还可以进行凭证的审核、作废、打印、冲红登财务操作。



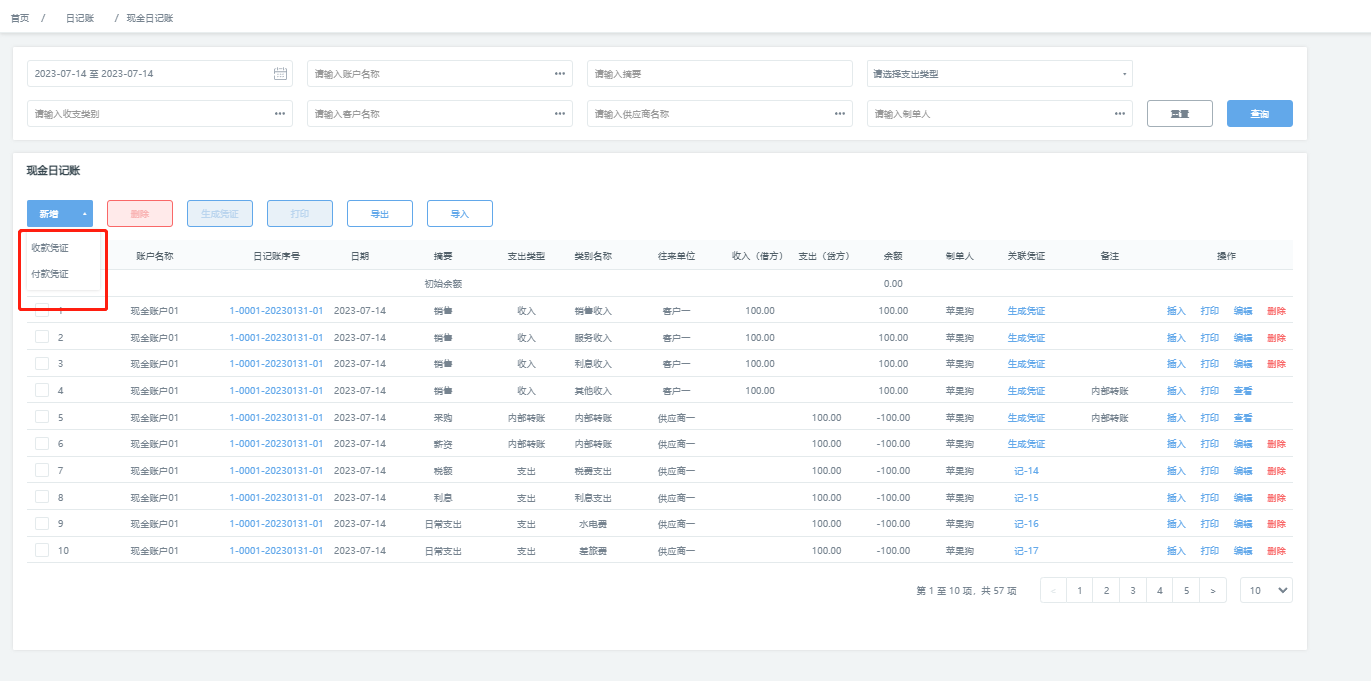
**2.1.4资金管理**

账点击顶部菜单资金后，左侧菜单栏展示资金下面的所有菜单，包括现金日记账、银行日记账、内部转账、资金报表、核对总账、账户设置、收支类别。如下图所示：

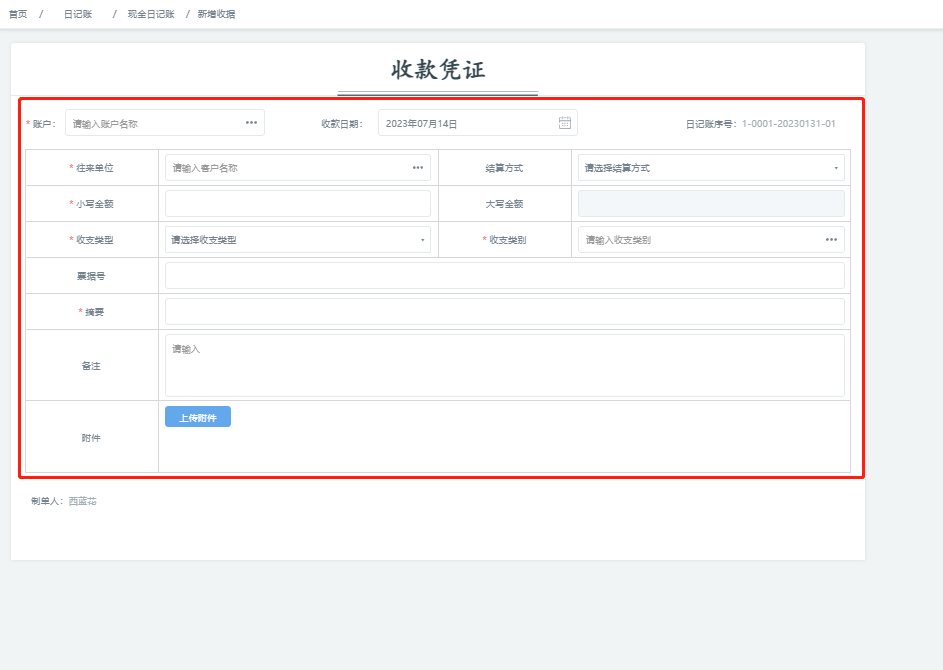
现金日记账：现金日记账列表展示现金日记账的详细数据，可以进行数据的增删改查、插入、打印、导入/导出及生成凭证。



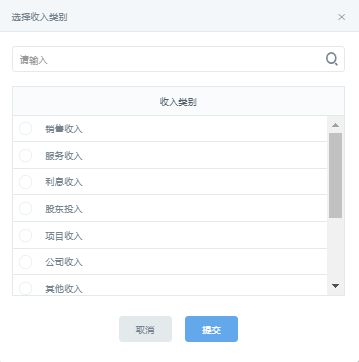
新增现金日记账：点击现金日记账列表中的新增按钮，下拉选项展示收款凭证、付款凭证，点选收款凭证，如下图所示：



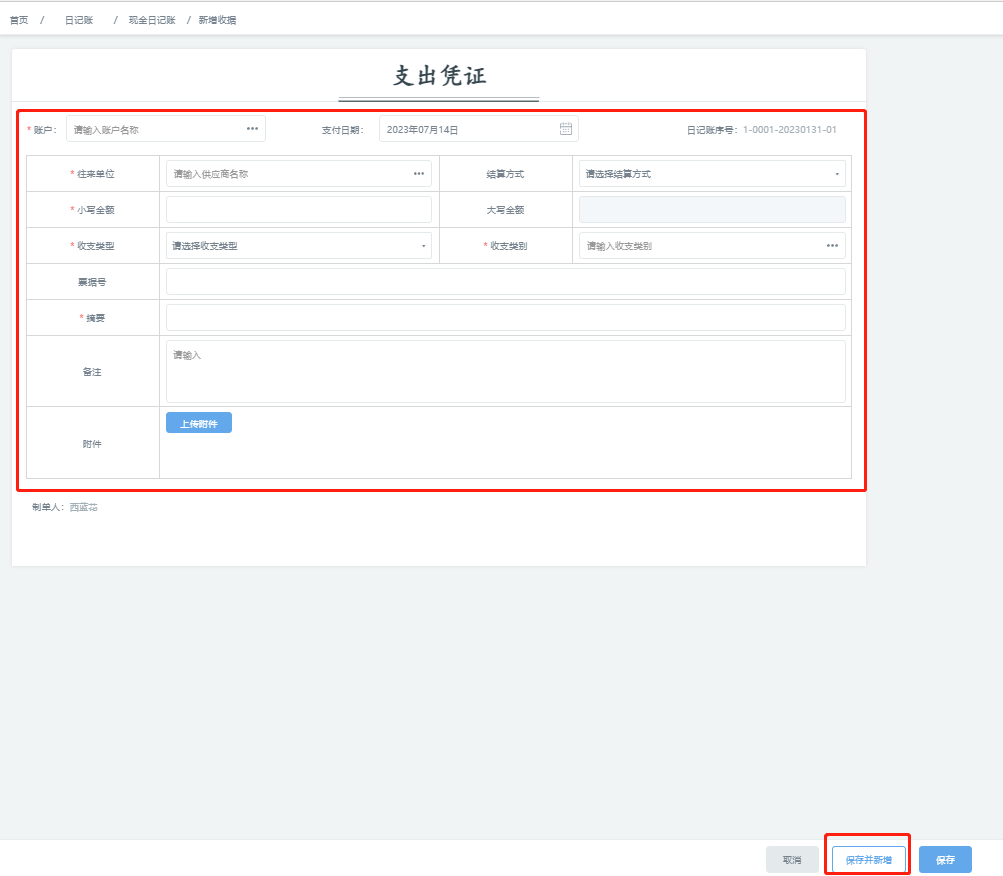
新增收款凭证：点选收款凭证后，进入收款凭证新增页面，如下图所示。输入账户名称、往来单位名称，选择结算方式，分别输入大小写金额、选择收支类型和收支类别，输入票据号、摘要，选填备注和附件，最后点击保存并新增即可，仅保存只是保存草稿。



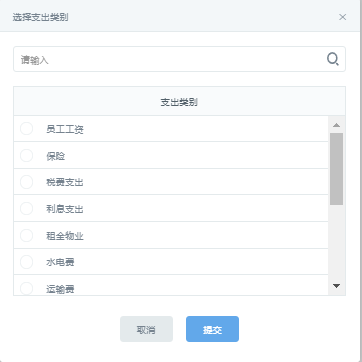
收入分类的选择：选择 收入分类时，点击“...”打开收入分类弹窗，列表展示说有的收入类别，可以直接找到后点选，也可以根据关键字检索后进行选择，最后点击提交按钮，既可以完成收入分类的选择。



新增支出凭证:点选付款凭证后，进入付款凭证新增页面，如下图所示。输入账户名称、往来单位名称，选择结算方式，分别输入大小写金额、选择收支类型和收支类别，输入票据号、摘要，选填备注和附件，最后点击保存并新增即可，仅保存只是保存草稿。



支出类别的选择：选择 收支类别时，点击“...”打开收入分类弹窗，列表展示说有的支出类别，可以直接找到后点选，也可以根据关键字检索后进行选择，最后点击提交按钮，既可以完成收入分类的选择。



# 第三章 表设计

## 3.1 用户设计表

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否必须 | 注释 |
| user\_id | string | 是 | 用户ID |
| dept\_id | string | 是 | 部门ID |
| user\_name | string | 是 | 用户账号 |
| nick\_name | string | 是 | 用户昵称 |
| user\_type | varchar | 是 | 用户类型 |
| email | varchar2 | 是 | 用户邮箱 |
| phonenumber | number | 是 | 手机号码 |
| sex | string | 是 | 用户性别（0男 1女 2未知） |
| avatar | string | 是 | 头像地址 |
| password | number | 是 | 密码 |
| status | string | 否 | 帐号状态（0正常 1停用） |
| del\_flag | string | 是 | 删除标志（0代表存在 2代表删除） |
| login\_ip | string | 是 | 最后登录IP |
| login\_date | string | 是 | 最后登录时间 |
| create\_by | string | 是 | 创建者 |
| create\_time | string | 是 | 创建时间 |
| update\_by | string | 是 | 更新者 |
| update\_time | string | 是 | 更新时间 |
| app\_id | string | 是 | 应用id |

## 3.2凭证管理表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名** | **数据类型** | **允许NULL** | **注释** |
| voucher\_id | String | 否 | 凭证序号 |
| acount\_name | varchar | 否 | 账户名称 |
| document\_num | int | 否 | 单据数 |
| abstract | int | 否 | 摘要 |
| subject | int | 否 | 会计科目 |
| amount\_debit | float | 否 | 解放金额 |
| amount\_credit | float | 否 | 贷方金额 |
| create\_by | string | 是 | 创建者 |
| create\_time | string | 是 | 创建时间 |

## 3.3现金日记账表

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段类型** | **是否必须** | **注释** |
| journal\_id | String | 否 | 记账序号 |
| acount\_name | varchar | 否 | 账户名称 |
| type | int | 否 | 收支类型 |
| time | datetime | 否 | 收付款时间 |
| settle\_type | int | 否 | 结算方式 |
| amount | float | 否 | 金额 |
| bill\_id | String | 否 | 票据号 |
| abstract | varchar | 是 | 摘要 |
| Notes | varchar | 是 | 备注 |
| create\_by | string | 是 | 创建者 |
| create\_time | string | 是 | 创建时间 |

# 第四章 接口设计

## 4.1 登录接口

登录接口

请求地址

/user/login.do

请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否必须 | 注释 |
| password | String | 是 | 密码 |
| username | String | 是 | 用户名 |
| code | String | 是 | 检验码 |

返回字段

|  |  |  |
| --- | --- | --- |
| 字段名称 | 字段类型 | 注释 |
| flag | boolean | 异常标识 |
| code | int | 错误码 |
| msg | string | 错误信息 |
| data | Object | 返回数据 |
| sex | string | 用户性别 |
| userName | string | 登录名称 |
| nickName | string | 用户姓名 |
| email | string | 用户邮箱 |
| status | string | 帐号状态（0正常 1停用） |
| delFlag | string | 删除标志（0存在 2删除） |
| loginIp | string | 最后登录IP |

## 4.2 添加凭证接口

请求地址

/voucher\_manage/add.do

请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否必须 | 注释 |
| voucher\_id | String | 否 | 凭证序号 |
| acount\_name | varchar | 否 | 账户名称 |
| document\_num | int | 否 | 单据数 |
| abstract | int | 否 | 摘要 |
| subject | int | 否 | 会计科目 |
| amount\_debit | float | 否 | 解放金额 |
| amount\_credit | float | 否 | 贷方金额 |
| create\_by | string | 是 | 创建者 |
| create\_time | string | 是 | 创建时间 |

返回字段

|  |  |  |
| --- | --- | --- |
| 字段名称 | 字段类型 | 注释 |
| flag | boolean | 异常标识 |
| code | int | 错误码 |
| msg | string | 错误信息 |
| data | Object | 返回数据 |

## 4.3 删除凭证接口

删除凭证接口

请求地址

/voucher\_manage/delete.do

请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否必须 | 注释 |
| voucher\_id | String | 否 | 凭证序号 |
| IsDelete | Strintring | 是 | 是否删除 |

返回字段

|  |  |  |
| --- | --- | --- |
| 字段名称 | 字段类型 | 注释 |
| flag | boolean | 异常标识 |
| code | int | 错误码 |
| msg | string | 错误信息 |
| data | Object | 返回数据 |

## 4.4 编辑凭证接口

编辑凭证接口

请求地址

/voucher\_managee/edit.do

请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段类型 | 是否必须 | 注释 |
| voucher\_id | String | 否 | 凭证序号 |
| edit\_by | edit\_time | 是 | 编辑人 |
| edit\_time | datetime | 是 | 编辑时间 |

返回字段

|  |  |  |
| --- | --- | --- |
| 字段名称 | 字段类型 | 注释 |
| flag | boolean | 异常标识 |
| code | int | 错误码 |
| msg | string | 错误信息 |
| data | Object | 返回数据 |

# 第五章 非功能性设计

## 5.1 性能设计

**响应时间**

为了能够快捷地提供查询服务，系统应该能够快速地相应查询请求。用户最终得到结果的响应时间除了与系统响应速度有关之外，还与网络情况有关。一下提出的是对WEB查询页面查询相应速度的需求：

|  |  |  |
| --- | --- | --- |
| 时间段 | 种类 | 响应时间（秒） |
| 平时 | 新增业务数据 | 2 |
| 高峰期 | 4 |
| 平时 | 简单查询 | 2 |
| 复杂查询 | 10 |
| 高峰期 | 简单查询 | 8 |
| 复杂查询 | 20 |

注：简单查询是指涉及单个条件的严格匹配查询；复杂查询是指涉及多个条件，或使用模糊匹配的查询及统计；高峰期指并发用户高于系统支持最大并发用户的60%时。

**处理能力**

系统处理能力主要考虑系统能承载的最大并发用户数，按照实际情况的规划，系统至少能承载的最大并发用户数要求达到400。

## 5.2 安全性设计

应按照信息系统安全等级保护第二级相关要求对本项目涉及的应用、数据、主机、网络、物理等安全进行建设，并对管理提出有效方案。

重要系统的应用、主机，关键网络节点，应做好冗余技术处理。重要数据应加强保护，避免未授权访问和篡改，并做好备份工作。

由于网络的互联性和开放性，使得数据传输面临种种危险，也由此提出了相应的系统安全要求。保证系统的安全，对于信息的安全性、完整性都非常重要。因此在构建系统时，必须考虑安全因素，制定一套信息安全的整体解决方案。

系统对安全要求包括信息保密性、身份的确定性、身份的不可否认性、文件的不可修改性、访问控制、身份验证和授权等。每一部分都非常重要。具体要求如下：

实现信息传输的加密，以密文方式传输，以防止数据被监听。

对接收后数据需进行必要的完整性校验，保证信息的完整性。

对保密程度较高的信息，在信息存储时实现信息的加密存储。

实现有效的用户身份认证，并支持多种认证方式，最大限度的保证了数据传输的可靠性。

由于组织机关文件的严肃性，实现信息确认发出后的不可否认性。

实现信息确认发出后的不可篡改性，保证整个数据传输过程的严肃和公正。

提供完善的、灵活的授权管理机制，实现不同权限的用户只能访问自己权限内的资源，保证系统资源的安全和完整性。

提供完善的日志管理功能，能完整的记录系统资源使用情况、用户登录及访问等应用情况。通过日志信息的分析可以及时发现系统级和应用级的安全问题，消除安全隐患。

## 5.3 其他非功能性设计

**易用性需求**

（1）应用系统应采用图形化人机界面，操作指令应易于理解；

（2）所有操作菜单、提示信息全部使用简体中文；

（3）应用系统的界面风格应与当前主流工具软件的风格一致；具备丰富的；

（4）鼠标双击、鼠标右键点击功能，具备灵活的动态窗口功能；

（5）应用系统应具备完备的联机帮助功能；

（6）应用系统应具备用户操作提醒功能；

（7）系统应自动维护不同数据库相同字段数据的一致性；

（8）能根据需要灵活定制统计内容和报表格式；

（9）一项功能的多个计算机指令操作尽可能在同一界面完成，确实需要进行屏幕切换的，屏幕切换次数不应超过2次。

**可靠性需求**

(1) 应用系统应具备满足平台全面应用的条件下的高可靠性，应用系统试运行期间，在其他软件运行正常的情况下，应用系统中断的故障次数不应超过3次/月；在系统保证期内，应用系统中断的故障次数不应超过1次/月。应用系统应具备完善的错误处理机制，对系统运行过程中的错误处理有明确的提示信息，指导用户操作；

(2) 系统应具备数据的转存和恢复机制；

(3) 应用系统应具备恢复保障机制，在软硬件平台正常运行的情况下，应用系统故障的恢复时间不超过30分钟。

**可扩展性需求**

系统的可扩展性设计表现在以下几个方面：

1. 数据可扩展

系统数据库是建立一个统一的可扩展的数据模型基础之上的，这是建立在对系统数据需求进行全面综合分析的基础之上，该数据模型能够建立不同空间数据间的、不同业务数据间的、以及空间数据与业务数据间的、管理数据与运行数据间存在的各种关系。即使现在建设时，许多数据并不完整或完善，可以通过模型的可扩展性，可以在将来逐步完善，并添加新的数据。

2.功能可扩展

系统采用“微内核”＋“插件”的开发技术，并采用组件化、构件化技术开发插件，应用支撑平台提供了一致的接口规范。这样使得需要优化某一系统功能时只需在原来的接口规范基础上，改进组件内部的设计，以提供功能更为强大的组件。当需要增加新的功能时，通过开发包含新功能的插件，经通过简单的配置就可以集成到系统中。系统还提供有二次开发接口，也可以通过二次开发来增强系统的功能。

3.业务可扩展

系统的业务都是通过定制来实现，因而业务数据和流转条件等信息与系统程序之间并没有紧密绑定，而是通过读取或传输数据流、业务流来实现。因此，系统需要改变业务流程，新增或者删除某个业务对系统本身没有影响。系统采用这种平台框架，对业务进行定制，应用与设计无关，所以很容易扩展新的业务。

4. 服务可扩展

系统对外提供服务方式包括数据交换、订阅/发布、远程数据访问和WebService等多种方式。尤其是通过建立基于WebService的数据交换接口，能够定制出针对不同对象的不同种类、格式不同内容的数据服务，实现服务的可扩展。

5.灵活性设计

系统的灵活性设计是出于对系统业务的可变、伸缩性而进行的考虑。要求系统能够灵活的配置业务流与数据流的流向，能够灵活的处置业务环节的变化。这一切都是通过实现的工作流引擎和表单报表定制平台完成的。系统的灵活性和可伸缩性表现在：

1.>业务流程灵活性——由于职能调整而改变业务流

由于职能的调整与变化，引起职能的执行部门发生变化，业务流向发生改变，系统只需更改工作流实例的执行用户即可。

2.> 数据交换的灵活性——业务变化引起数据需求变化

由于业务变化，引起所需的数据发生相应的改变，需要获取别处的数据，系统可以修改工作流节点的数据源，通过图形化自定义的表单来获取新的数据。

3.> 用户功能的灵活性配置——岗位职责变化，岗位对系统功能变化

由于岗位职责的调整与变化，相同的岗位根据业务职能的调整，需要的系统功能会有调整。系统可以根据统一的角色权限系统来给用户统一配置所需的功能和权限。

# 第六章 使用说明

## 6.1 安装及初始化及软件主要参数设置

一、安装与初始化及软件主要参数设置

软件环境：

JDK：1.8Maven：3.4.9 Mysql:5.7 nginx：1.2.0

软件编译：

[root@localhost local] cd /usr/server/ [root@localhost local] mvn clean install

软件部署(启停命令)：

[root@localhost local] ./bin./start.sh [root@localhost local] ./bin/shutdown.sh

二、软件故障排除

1）查看服务进程

[root@localhost local] ps –ef | background-server

2）查看进程是否存在

# 如果不存在，进入[root@localhost local] cd /usr/server/ #启动服务：[root@localhost local] ./bin./start.sh

3）查看日志

[root@localhost local] cd /usr/server/logs[root@localhost local] tail -f logs.log

三、安装部署步骤

如果没有，先执行SQL文件夹下的 "SYS\_USER.sql"文件和" SYS\_CONFIG.sql"文件，然后在数据库里再执行 "初始化数据.sql" 确保执行成功 。

## 6.2 运行环境

**软件环境**

WEB服务器：

1、 操作系统：CentOS 6.5

2、 应用服务器：Tomcat 8.5

3、 Java运行环境：JDK8.0

DB服务器：

4、 操作系统：CentOS 6.5

5、 数据库：MySQL5.7

6、 数据库：MongoDB 3.X

Redis服务器：

7、 操作系统：CentOS 6.5

8、 Redis服务器：Redis4.0

微服务注册中心服务器：

9、 操作系统：CentOS 6.5

10、 Eureka-Server: Eureka2.x

客户端：

11、 操作系统：Microsoft Windows 7，8,10

12、 浏览器：Microsoft Internet Explorer 9，10,11，Chrome60+

**网络环境**

需要提供互联网和行内网访问系统，互联网带宽要求10M。

**硬件环境**

服务器端：

1. WEB服务器建议最低配置：云服务器ECS CPU：4核；内存：16G；  (共10台)

2. MySQL服务器建议最低配置：云数据库MySQL CPU：8核；内存：16G；

3. MongoDB服务器建议最低配置：云数据库MongoDB CPU：8核；内存：64G；

4. Redis服务器建议最低配置：云服务器Redis CPU：4核；内存：32G；

5. 文件存储服务器建议最低配置：对象存储服务OSS

6. 负载均衡服务器建议最低配置：负载均衡SLB

客户端：

PC建议配置：CPU I7；内存 8G；显卡支持  1920\*1080

PC最低配置：CPU I5；内存 4G；显卡支持  1366\*900

备注：如系统运行环境达不到最低配置要求，系统可能出现运行缓慢等现象。

## 6.3 技术特点

1. 采用了j2ee+tomcat+mysql的模式进行开发；

2. 可伸缩性和可用性高，采用企业级数据库功能；

3. 易于安装、部署和使用，可重用性高；

4. 采用BS架构，性能提升高，易于管理，易于维护。