**××系统详细设计说明书**

[版本号]

**大唐软件**

目 录

[1. 引言 3](#_Toc352230375)

[**1.1.** 产品描述 3](#_Toc352230376)

[1.2. 产品开发背景和内容概要 3](#_Toc352230377)

[1.3. 相关资料、缩略语、定义 3](#_Toc352230378)

[1.3.1. 相关文档 3](#_Toc352230379)

[1.3.2. 缩写词和名词定义 3](#_Toc352230380)

[1.3.3. 参考资料 3](#_Toc352230381)

[2. 系统功能详细设计 3](#_Toc352230382)

[2.1. [子系统1] 3](#_Toc352230383)

[2.1.1. [功能模块1] 3](#_Toc352230384)

[2.2. [子系统2] 3](#_Toc352230385)

[2.3. [子系统3] 3](#_Toc352230386)

[3. 接口设计 3](#_Toc352230387)

[3.1. 与外系统的接口 3](#_Toc352230388)

[3.1.1. 外部接口1 3](#_Toc352230389)

[3.1.1.1. 概述 3](#_Toc352230390)

[3.1.1.2. 接口流程 3](#_Toc352230391)

[3.1.1.3. 接口约定 3](#_Toc352230392)

[3.1.1.4. 接口数据结构 3](#_Toc352230393)

[3.1.1.5. 接口服务描述 3](#_Toc352230394)

[3.1.1.5.1. 服务1 3](#_Toc352230395)

[3.1.1.5.2. 服务2 3](#_Toc352230396)

[3.1.1.6. 接口异常处理 3](#_Toc352230397)

[3.2. 内部接口 3](#_Toc352230398)

[3.2.1. 内部接口1 3](#_Toc352230399)

[3.2.1.1. 概述 3](#_Toc352230400)

[3.2.1.2. 接口流程 3](#_Toc352230401)

[3.2.1.3. 接口约定 3](#_Toc352230402)

[3.2.1.4. 接口数据结构 3](#_Toc352230403)

[3.2.1.5. 接口服务描述 3](#_Toc352230404)

[3.2.1.5.1. 服务1 3](#_Toc352230405)

[3.2.1.5.2. 服务2 3](#_Toc352230406)

[4. 系统异常设计 3](#_Toc352230407)

[**4.1.** 系统异常设计规范 3](#_Toc352230408)

[**4.2.** 系统异常设计 3](#_Toc352230409)

[5. 系统监控设计 3](#_Toc352230410)

[6. 修订记录 3](#_Toc352230411)

引言

产品描述

（产品编号、产品名称、操作系统平台、数据库系统平台、开发语言、功能版本、用户名称、产品开发项目组、产品交付项目组）

## 产品开发背景和内容概要

（产品提出、产生、完善的发展背景；产品简介、应用概述）

## 相关资料、缩略语、定义

相关文档

（描和本文档有直接关系的相关文档，包括本文档的输入文档，以及以本文档作为输入的相关文档）

缩写词和名词定义

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **术语** | **英文** | **缩写** | **同义词** | **描述** |
|  |  |  |  |  |
|  |  |  |  |  |

参考资料

（本小节应完整列出本文档中其他部分所引用的任何文档。每个文档应标有标题、报告号（如果适用）、日期和出版单位。列出可从中获取这些参考资料的来源。这些信息可以通过引用附录或其他文档来提供。）

系统功能详细设计

## [子系统1]

[功能模块1]

#### 模块生产方式

(包括自行开发、外包委托开发、购买、复用模块等方式，如果不做说明，默认为自行开发

复用方式:设计复用标准，模块清单、修改比例和估计工作量；

购买方式：比如报表工具等商业现货时，要制订维护采购模块的开发维护计划，商业现货软件参照《项目采购商业现货过程》；

外包委托开发：参照《项目外包过程》。)

#### [功能1]

(给出类图的结构，要分层摆放类的位置，从上到下分别是：展现层、业务逻辑层、领域层、数据访问层、数据持久层)

##### **功能描述**

##### **模型设计**

##### **业务处理流程**

##### **界面设计**

##### **展现层设计描述**

（同业务逻辑层设计描述类似）

##### **服务层设计描述**

（同业务逻辑层设计描述类似）

##### **业务逻辑层设计描述**

（对类的实现给出详细的描述，包括类的标识、简介（概念、功能）、属性、方法等，可以参照如下模板：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 数据类型 | 可见属性 | 含义及用法 |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **方法名称** |  | | | | | |
| **修改历史** | 修改时间 | 修改人 | | 修改原因 | | 修改内容 |
|  |  | |  | |  |
|  |  | |  | |  |
| **功能说明** |  | | | | | |
|  | 参数名称 | | 数据类型 | | 含义 | |
| **输入参数** |  | |  | |  | |
|  | |  | |  | |
| **输出参数** |  | |  | |  | |
|  | |  | |  | |
| **返回值** |  | |  | |  | |
| **调用的领域及数据访问组件** | 组件名称（字母） | | | | 调用的方法 | |
|  | | | |  | |
| **异常处理** | 异常处理的详细描述，对每个异常都要给出系统显示的详细信息提示内容。 | | | | | |
| **方法实现描述**  **(要把代码的整体逻辑结构都要描述清楚，也就是业务逻辑的主要路径以及每个步骤的实现要求，也就是出伪代码结构)** | Public void delBundle (ProductBundle productBundle)throws SQLException  {  //检查该产品包是否被商品所引用，如果已引用抛出异常；  //调用产品包DAO的delete方法删除产品包对象  //如果出错，抛出异常；  } | | | | | |

）

##### **数据访问层设计描述**

（同业务逻辑层设计描述类似）

##### **数据持久层设计描述**

（同业务逻辑层设计描述类似，根据实际情况确定是否需要在详细设计中具体描述实现）

##### **数据存储层设计描述**

（同业务逻辑层设计描述类似）

## [子系统2]

## [子系统3]

接口设计

(如果有《接口详细设计说明书》对接口设计进行单独描述，此章可裁剪，否则不可裁剪)

## 与外系统的接口

### 外部接口1

### 概述

（简要描述与本接口相关系统背景，及本接口功能、接口设计理念等）

### 接口流程

（描述接口处理流程，对处理流程的每个环节的业务逻辑进行详细描述。此处必须有处理流程图来说明）

### 接口约定

（主要描述在接口协商过程中定义的一些约定内容）

### 接口数据结构

### 接口服务描述

### 服务1

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 |  | | |
| 方法名 |  | | |
| 服务说明 |  | | |
|  | 参数名称 | 数据类型 | 含义 |
| 输入参数 |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| 返回值 |  |  |  |
| 异常处理 | 异常处理的详细描述，对每个异常都要给出系统显示的详细信息提示内容。 | | |
| 方法实现描述**(**要把代码的整体逻辑结构都要描述清楚，也就是业务逻辑的主要路径以及每个步骤的实现要求，也就是出伪代码结构) | Public void delBundle (ProductBundle productBundle)throws SQLException  {  //检查该产品包是否被商品所引用，如果已引用抛出异常；  //调用产品包DAO的delete方法删除产品包对象  //如果出错，抛出异常；  } | | |
| 使用系统 |  | | |

### 服务2

### 接口异常处理

## 内部接口

(描述软件内部子系统之间、模块之间的接口)

### 内部接口1

### 概述

（简要描述与本接口相关系统背景，及本接口功能、接口设计理念等）

### 接口流程

（描述接口处理流程，对处理流程的每个环节的业务逻辑进行详细描述。此处必须有处理流程图来说明）

### 接口约定

（主要描述在接口协商过程中定义的一些约定内容）

### 接口数据结构

### 接口服务描述

### 服务1

|  |  |  |  |
| --- | --- | --- | --- |
| 类名 |  | | |
| 方法名 |  | | |
| 服务说明 |  | | |
|  | 参数名称 | 数据类型 | 含义 |
| 输入参数 |  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
| 返回值 |  |  |  |
| 异常处理 | 异常处理的详细描述，对每个异常都要给出系统显示的详细信息提示内容。 | | |
| 方法实现描述**(**要把代码的整体逻辑结构都要描述清楚，也就是业务逻辑的主要路径以及每个步骤的实现要求，也就是出伪代码结构) | Public void delBundle (ProductBundle productBundle)throws SQLException  {  //检查该产品包是否被商品所引用，如果已引用抛出异常；  //调用产品包DAO的delete方法删除产品包对象  //如果出错，抛出异常；  } | | |
| 使用系统 |  | | |

### 服务2

系统异常设计

系统异常设计规范

（系统异常处理的详细设计，包括但不限于：

异常代码的规范，异常代码在各个模块的分配。

异常的类型，如系统级异常、应用级异常等，异常类型可以在细分，如系统级异常可以再细分为中间件层异、数据库层异常等；常应用级异常可再细分为操作异常、数据异常等；

异常发生的层次：在表现层、业务逻辑层、领域层、数据访问层、数据持久层、数据存储层等；

异常的级别：如致命错误、严重错误、一般错误、告警消息等

异常发生的位置：具体的程序位置，系统执行时的调用层次关系，异常抛出的层次关系

异常的信息展现：代码、异常信息、解决方案提示等）

系统异常设计

（具体的异常类的设计及描述）

系统监控设计

（系统的每个组件，除了要提供对应的业务处理功能外，还需要提供组件运行的性能参数统计数据，以便系统能够对系统的运行情况进行详尽的监控，但是这个设计是可选的设计，根据系统的情况确定是否真正需要，因为除了增加系统复杂度外，还会影响系统的运行性能

可以做成开关，如果开关打开，则系统组件运行的同时，可以产生运行性能统计参数数据，在系统性能测试、性能调优等方面都会发挥作用）

修订记录

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本** | **简要变更说明** | **变更人** | **变更日期** | 【**文件更改单**】**号** |
| A | 新建，参考原软件公司《系统详细设计说明书》修改而来 | 王静兰 | 2013-3-28 |  |
|  |  |  |  |  |
|  |  |  |  |  |