




成绩

中国农业大学

课程设计

(2019-2020 学年春季学期)

题 目： 概要设计报告

课程名称： 软件工程

任课教师： 李林

班 级： 计算机 172

姓 名： 张靖祥

目录

1 HJAC 的应用.....	1
1.1 用户组别与页面关系表.....	1
1.2 动物档案修改业务页面 HJCA 技术应用.....	2
1.2.1 页面文件设计.....	2
1.2.2 页面技术点设计.....	2
1.2.3 业务校验.....	2
1.2.4 数据提取.....	3
1.3 实验数据修改业务页面 HJCA 技术应用.....	3
1.3.1 页面文件设计.....	3
1.3.2 页面技术点设计.....	4
1.3.3 业务校验.....	4
1.3.4 数据提取.....	4
1.4 审核页面 HJCA 技术应用.....	4
1.4.1 页面文件设计.....	4
1.4.2 页面技术点设计.....	5
1.4.3 数据提取.....	5
1.5 日志整理页面 HJCA 技术应用.....	5
1.5.1 页面文件设计.....	5
1.5.2 页面技术点设计.....	6
1.5.3 数据提取.....	6
2 Spring MVC 在动物档案管理系统中的应用.....	7
2.1 web.xml 配置设计.....	7
2.2 Controller 类设计.....	7
2.2.1 动物档案修改 Controller 类设计.....	7
2.2.2 实验数据修改 Controller 类设计.....	8
2.2.3 审核修改 Controller 类设计.....	9
2.2.4 日志整理 Controller 类设计.....	9
2.3 springMVC 配置表.....	10
2.4 数据类型转换表设计.....	11
2.5 数据校验表设计.....	11
2.6 国际化应用设计.....	12
2.7 动物档案管理系统拦截器设计.....	13
2.7.1 动物档案修改拦截器设计.....	13

2.7.2 实验数据修改拦截器设计	13
2.8 操作日志设计	14
2.9 SpringMVC 标签库设计	14
3 Mybatis 在动物档案管理系统中的应用	15
3.1 Mybatis 应用配置设计	15
3.1.1 Mybatis 服务配置全局设计	15
3.1.2 JDBC 数据源用户角色表设计	15
3.2 会话 Mapper 设计表	16
3.3 实体 Domain 设计	16
3.4 Mybatis 安全设计	16
4 界面元素设计	17
4.1 动物档案修改页面元素设计	18
4.1.1 动物档案修改页面元素设计表	19
4.1.2 界面设计与界面元素说明	19
4.2 实验数据修改页面元素设计	20
4.2.1 实验数据修改页面元素设计表	21
4.2.2 界面设计与界面元素说明	22
4.3 操作日志页面元素设计	22
4.3.1 操作日志页面元素设计表	23
4.3.2 界面设计与界面元素说明	23
5 状态实现	24
5.1 动物档案修改转态改变表	24
5.2 实验数据修改转态改变表	25
6 数据结构详细设计	25
6.1 动物档案修改数据元素结构设计	25
6.1.1 表示层数据元素结构设计	25
6.1.2 控制层数据元素结构设计	27
6.1.3 业务逻辑层数据元素结构设计	28
6.2 实验数据修改数据元素结构设计	29
6.2.1 表示层数据元素结构设计	29
6.2.2 控制层数据元素结构设计	31
6.2.3 业务逻辑层数据元素结构设计	32
7 系统运行详细设计	32
7.1 活动图在系统详细设计中的体现	32
7.2 系统运行设计	33

7.2.1 定位活动对应的架构层次.....	33
7.2.2 定位不同的返回值页面.....	33
7.2.3 异常处理.....	35
8 算法设计.....	35
8.1 分析类间关系.....	35
8.2 分析算法场景.....	37
8.3 描述实现数据结构.....	37
8.4 业务算法设计.....	39
8.5 确定并描述程序算法.....	39

1 HJAC 的应用

1.1 用户组别与页面关系表

为了动态生成显示页面，我们将用户进行了划分，并在每个用户在验证成功后，动态生成系统页面，并执行相应的操作。我们采用了对用户进行分组的方法，然后不同的用户组分别赋予不同的用户权限，每个用户组按照其业务类型分别赋予多个执行任务。在判断了不同的用户组别后，系统将定位到相应的用户页面，页面与用户组别有关系。

表 1.1

用户组别	代码	页面名称	说明
动物转平台部门组	AnimalTransferUser	V_AnimalTransferUserIndex.jsp	
实验室部门组	ExperimentUser	V_ExperimentUserIndex.jsp	
档案审核员组	ArchiveInspectWorker	V_ArchiveInspectWorkerIndex.jsp	
档案信息维护组	ArchiveMaintainWorker	V_ArchiveMaintainWorkerIndex.jsp	

用例图如下

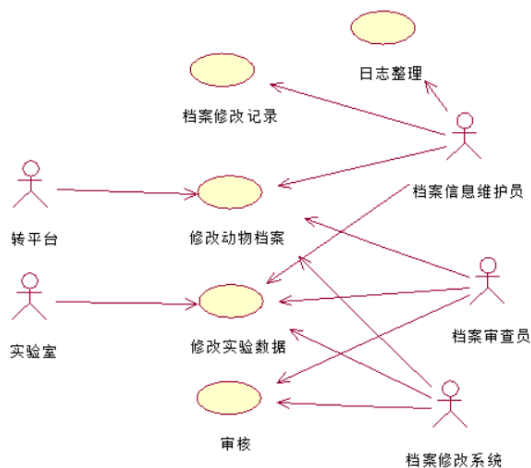


图 1.2

1.2 动物档案修改业务页面 HJCA 技术应用

档案修改业务我们只针对转平台开放，转平台部门在进行动物的转运之后，需要来修改动物的基本档案。共同参与的人员为转平台的用户、档案审查员，需要的中间系统为档案修改系统进行参与完成。需要完成对于动物档案表与档案更新日志的更新操作。

1.2.1 页面文件设计

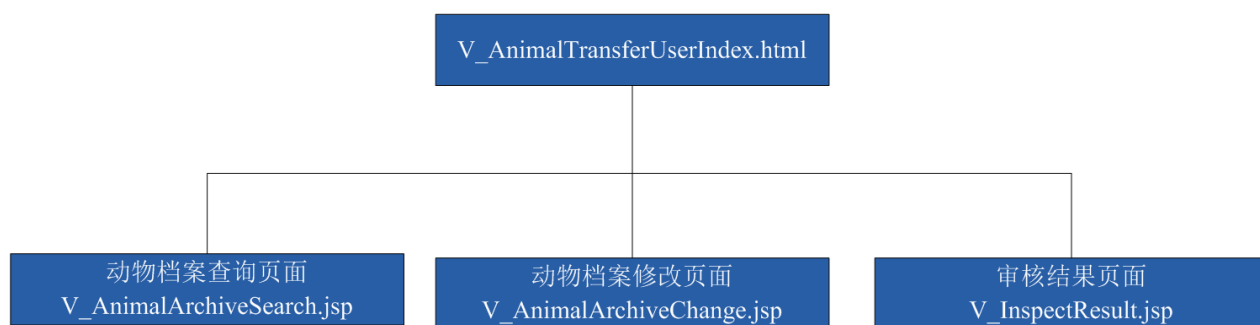


图 1.2.1

1.2.2 页面技术点设计

1. 动物档案查询信息不是固定的数据记录行，是动态增加的。

解决办法：采用 JavaScript

2. 关于数据的组织问题：数据检索部分，只要输入相应的参数值，然后通过控制层以及业务逻辑层获取数据值，然后投保书建档页面做出相应的处理。但是 Html 中无法嵌套定义 form。

解决办法：在提交每一个 submit 请求，并且要求 Action 层处理时，都需要通过 JavaScript 来进行重新组织。

1.2.3 业务校验

业务校验有三种解决办法

1. 通过 validate.xml 文件配置（通过配置文件设置检验规则）
2. 通过 Controller 层、MYBatis 层完成业务逻辑校验（服务器后端检验）
3. 通过 Ajax 技术处理数据业务规则校验（网页前端校验）

实验动物档案管理项目中的动物档案修改业务有两类校验规则：

1. 静态业务规则：对输入的动物档案修改数据信息的数据类型、长度、数据最大值、数据最小值，采用 validate.xml
2. 动态业务规则：随着政策、法规、产品变化等进行规则变化，具有很强的共享性，采用数据表验证

1.2.4 数据提取

表 1.2.4 动物档案修改页面数据提取

页面功能	技术描述
查询员工工号	使用 JavaScript，动态进行页面显示
查询档案编号	使用 JavaScript，页面跳转并动态显示档案
修改动物档案信息	使用 JavaScript，将修改结果自动提交申请
查询审核结果	使用 JavaScript，在刷新页面后显示申请结果

1.3 实验数据修改业务页面 HJCA 技术应用

修改实验数据业务我们只针对实验室开放，实验室部门在进行动物的试验之后，有可能需要对试验的记录进行调整，需要来修改实验数据档案。共同参与的人员为实验室的用户、档案审查员，需要的中间系统为档案修改系统进行参与完成。需要完成对于动物档案表与档案更新日志的更新操作。

1.3.1 页面文件设计

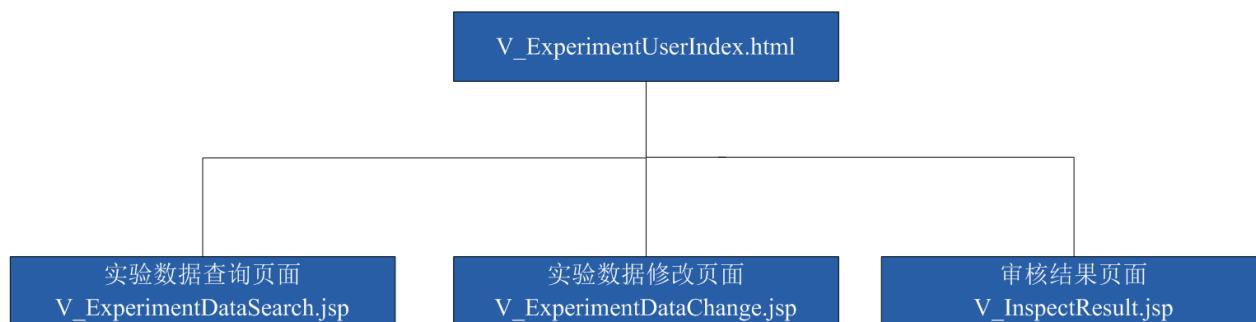


图 1.3.1

1.3.2 页面技术点设计

1. 实验数据查询信息不是固定的数据记录行，是动态增加的。

解决办法：采用 JavaScript

2. 关于数据的组织问题：数据检索部分，只要输入相应的参数值，然后通过控制层以及业务逻辑层获取数据值，然后投保书建档页面做出相应的处理。但是 Html 中无法嵌套定义 form。

解决办法：在提交每一个 submit 请求，并且要求 Action 层处理时，都需要通过 JavaScript 来进行重新组织。

1.3.3 业务校验

实验动物档案管理项目中的实验数据修改业务有两类校验规则：

1. 静态业务规则：对输入的实验数据修改数据信息的数据类型、长度、数据最大值、数据最小值，采用 validate.xml
2. 动态业务规则：随着政策、法规、产品变化等进行规则变化，具有很强的共享性，采用数据表验证

1.3.4 数据提取

表 1.3.4 实验数据修改页面数据提取

页面功能	技术描述
查询员工工号	使用 JavaScript，动态进行页面显示
查询实验数据	使用 JavaScript，页面跳转并动态显示实验数据
修改实验数据信息	使用 JavaScript，将修改结果自动提交申请
查询审核结果	使用 JavaScript，在刷新页面后显示申请结果

1.4 审核页面 HJCA 技术应用

1.4.1 页面文件设计

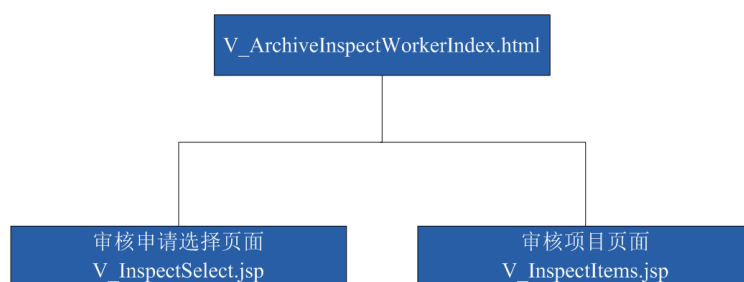


图 1.4.1

1.4.2 页面技术点设计

1. 待审核的信息不是固定的数据记录行，是动态增加的。
解决办法：采用 JavaScript
2. 为方便审核员使用，审核界面在完成审核后不进行跳转，直接开始审核下一条记录
解决办法：采用 AJAX 技术

1.4.3 数据提取

表 1.4.3 审核页面数据提取

页面功能	技术描述
查询审核记录	使用 JavaScript，动态进行页面显示
提交审核结果	使用 AJAX，动态完成页面提交与刷新

1.5 日志整理页面 HJCA 技术应用

1.5.1 页面文件设计

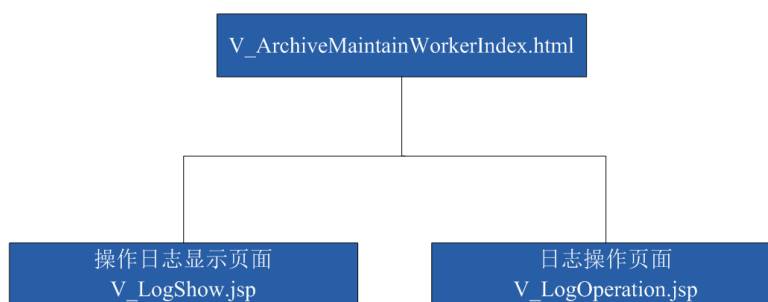


图 1.5.1

1.5.2 页面技术点设计

1. 档案日志显示的信息不是固定的数据记录行，是动态增加的。
解决办法：采用 JavaScript
2. 为方便日志整理员使用，每一条日志的整理操作是动态完成并记录后台数据库的。
解决办法：采用 AJAX 技术

1.5.3 数据提取

表 1.4.3 日志整理页面数据提取

页面功能	技术描述
查询日志记录	使用 JavaScript，动态进行页面显示
对日志进行整理	使用 AJAX，动态完成页面提交与刷新

2 Spring MVC 在动物档案管理系统中的应用

2.1 web.xml 配置设计

动物档案管理系统采用的 Spring MVC 框架的核心 filter 实现类：

Org.apache.spring.dispatcher.FilterDispatche

Spring MVC 的标签库配置如下：

Spring MVC 标签库的 URI:/S

标签库定义文件位置：/WEB-INF/struts-tags.tld

欢迎页面：V_Login.jsp

表 2.1 动物档案修改系统 Servlet.xml 文件规划表

文件名	描述
AnimalTranserUser_Servlet.xml	动物档案修改子系统 Controller 及互相映射描述
ExperimentData_Servlet.xml	实验数据修改子系统 Controller 及互相映射描述
ArchiveInspectWorker_Servlet.xml	审核申请子系统 Controller 及互相映射描述
ArchiveMaintainWorker_Servlet.xml	日志整理子系统 Controller 及互相映射描述

2.2 Controller 类设计

2.2.1 动物档案修改 Controller 类设计

表 2.2.1

动物档案修改页面 (V_AnimalArchivePage)	调用 Action 或调用页面				功能说明
	类名	方法	参数/类型	返回值	
1、查询动物档案记录表 QueryAnimalArchive	ImplementsAnimalArchiveAction	QuerybyAnimalNumber	Number/String	AnimalArchiveBaseInfo /Entity	查询动物档案记录
6、提交动物档案修改信息 SaveAnimalArchive	ImplementsAnimalArchiveAction	SaveAnimalArchive	AnimalArchiveBaseInfo /Entity	.t.or.f/boolean	保存动物档案修改记录

8、验证用户权限 QueryUserAuthority	ImplementsUserAction	QueryByUserNumber	Number/String	UserBaseInfo /Entity	查询用户对表的操作权限
9、验证动物档案修改内容 QueryAnimalArchive	ImplementsAnimalArchiveAction	QuerybyAnimalNumber	Number/String	AnimalArchiveBaseInfo /Entity	审核动物档案修改内容
10、保存动物档案修改记录 SaveAnimalArchive	ImplementsAnimalArchiveAction	SaveAnimalArchive	AnimalArchiveBaseInfo /Entity	.t.or.f/boolean	保存修改记录
11、登记动物档案修改日志 SaveChangLog	ImplementsLogAction	Save	ExperimentRecordInfo/Entity	.t.or.f/boolean	保存修改日志

2.2.2 实验数据修改 Controller 类设计

表 2.2.2

实验数据修改页面 (V_ExperimentPage)	调用 Action 或调用页面				功能说明
	类名	方法	参数/类型	返回值	
1、查询实验数据记录表 QueryByExperimentRecordNumber	ImplementsExperimentRecordAction	QueryByExperimentRecordNumber	Number/String	ExperimentRecordInfo/Entity	按实验档案编号进行查询
6、提交实验数据修改信 SaveExperimentArchive	ImplementsExperimentRecordAction	SaveExperimentData	ExperimentRecordInfo/Entity	.t.or.f/boolean	暂存用户修改记录
8、验证用户权限 QueryUserAuthority	ImplementsUserAction	QueryByUserNumber	Number/String	UserBaseInfo /Entity	查询用户对表的操作权限
9、验证实验数据修改内容 QueryByExperiment	ImplementsExperimentRecordAction	QueryByExperimentRecordNumber	Number/String	ExperimentRecordInfo/Entity	验证用户的修改内容是否合

RecordNumber					规
10、保存实验数据修改记录 SaveExperimentArchive	ImplementsExperimentRecordAction	SaveExperimentData	ExperimentRecordInfo/Entity	.t.or.f/boolean	保存修改记录
11、登记实验数据修改日志 SaveChangLog	ImplementsLogAction	SaveExperimentData	ExperimentRecordInfo/Entity	.t.or.f/boolean	保存修改日志

2.2.3 审核修改 Controller 类设计

表 2.2.3

审核修改页面 (V_InspectPage)	调用 Action 或调用页面				功能说明
	类名	方法	参数/类型	返回值	
1、查询实验数据记录表 QueryByChangingArchiveNumber	ImplementsChangingArchiveAction	QueryByChangingArchiveNumber	Number/String	ExperimentRecordInfo/AnimalArchiveBaseInfo /Entity	按实验档案编号进行查询
2、提交实验数据修改信 SaveChangingArchive	ImplementsChangingArchiveAction	SaveInspectResult	ExperimentRecordInfo/AnimalArchiveBaseInfo /Entity	.t.or.f/boolean	保存审核数据

2.2.4 日志整理 Controller 类设计

表 2.2.4

日志整理页面 (V_LogMaintainPage)	调用 Action 或调用页面				功能说明
	类名	方法	参数/类型	返回值	

1、查询日志记录 QueryByLogNumber	ImplementsLogMaintainAction	QueryByLogNumber	Number/String	ArchiveChangingLog/Entity	按实验档案编号进行查询
2、提交实验数据修改信 Save	ImplementsLogMaintainAction	SaveLogChanging	ArchiveChangingLog/Entity	.t.or.f/boolean	保存日志整理

2.3 springMVC 配置表

1. 确定谁要做什么事的问题，表示层具体的 JSP 文件，JSP 页面中由何种事件或方法体完成。
2. 确定每个事件和方法将要关联的 controller 对应的类以及这些类中的具体方法。
3. 选择 controller 返回类型，并且确定返回值所对应的页面。

表 2.3

View (包: IUW.JSP.JSPTellerSystem)		Controller 类		
PageName	Submit	Name	Class	Method
AnimalArchivePage	QueryAnimalArchive	QuerybyAnimalNumber	ImplementsAnimalArchiveAction	QuerybyAnimalNumber
	SaveAnimalArchive	SaveAnimalArchive	ImplementsAnimalArchiveAction	SaveAnimalArchive
ExperimentPage	QueryExperimentData	QueryByExperimentRecordNumber	ImplementsExperimentRecordAction	QueryByExperimentRecordNumber
	SaveExperimentData	SaveExperimentData	ImplementsExperimentRecordAction	SaveExperimentData
InspectPage	QueryChangingArchive	QueryByChangingArchiveNumber	ImplementsChangingArchiveAction	QueryByChangingArchiveNumber
	SaveInspectResult	SaveInspectResult	ImplementsChangingArchiveAction	SaveInspectResult
LogMaintainPage	QueryLogNumber	QueryByLogNumber	ImplementsLogMaintainAction	QueryByLogNumber

	SaveLogChangi ng	SaveLogChangin g	ImplementsLogMai ntainAction	SaveLogChanging
--	---------------------	---------------------	---------------------------------	-----------------

2.4 数据类型转换表设计

Web 应用程序的交互是建立在 HTTP 之上的，互相传递的是字符串。而在 web 应用的对象中，使用多种不同的类型，如整数、浮点数、日期或自定义数据类型，因此在服务器端必须将字符串转换成合适的数据类型。

目前的解决办法有 3 种：

1. Spring MVC 框架提供了一些简单类型的转换器，如转换为 int，float 等简单数据类型。
2. 复杂类型和自定义类型需要定义转换器。
3. 在转换过程中，会出现一些异常，Spring MVC 的 conversionError 拦截器会自动处理异常，并在页面上生成提示信息。

表 2.4

数据集	数据集合名称	组织形式	类型转换
动物档案数据集	AnimalArchive	ArrayList	自定义转换
实验信息数据集	ExperimentData	ArrayList	自定义转换
档案更新操作日志	ArchiveUpdateLog	ArrayList	自定义转换
用户数据集	UserInformation	ArrayList	自定义转换

2.5 数据校验表设计

数据的组织形式有两种，1 是 Form 类型的数据，针对这种表单类型的数据，采用 controller 进行校验；2 是 ArrayList 类型的数据，针对这种数组形式的数据，采用 JavaScript 校验。

表 2.5

数据集	数据集合名称	组织形式	校验类型
动物档案数据集	AnimalArchive	ArrayList	Struts 自定义校验
实验信息数据集	ExperimentData	ArrayList	Struts 自定义校验
档案更新操作日志	ArchiveUpdateLog	ArrayList	Struts 自定义校验

用户数据集	UserInformation	ArrayList	Struts 自定义校验
-------	-----------------	-----------	--------------

2.6 国际化应用设计

动物档案管理国际化种类：

标题性语言提示语句要求严格按照动物档案管理系统数据字典提示，国际化语言 key 值严格按照数据字典中的英文标识编写，有多个单词组成的 key 值，需要去掉空格并且每个单词第一个字母大写。

提示语，如错误信息提示，需要专门编写语言对应表。

表 2.6

Key 值	中文提示	英文提示
MC	数据类型必须是字符串	Datatype must be String
CL	字符串长度	Character long must be value
CI	数据类型必须是整数型	Datatype must be Integer
DL	数据类型必须是 long 型	Datatype must be long
DS	数据类型必须是 short 型	Datatype must be short
DD	数据类型必须是日期型	Datatype must be date
DF	数据类型必须是浮点型	Datatype must be float
DDB	数据类型必须是 double 型	Datatype must be double
DB	数据类型必须是 boolean 型	Datatype must be Boolean
DBT	数据类型必须大于\$变量，小于\$变量	Numerical value is between max value and min value
DN	输入数据不能为空	Input data no null
DO	数据长且仅长\$变量	Long of data must be \$ value
DM	数据必须大于&变量	Data must be greater than \$ value
DG	数据必须小于\$变量	Data must be smaller than \$ value
DZ	数据最大值\$变量	Numerical value max value is \$ value
DMI	数据最最小值\$变量	Numerical value min value is \$ value
DEM	不符合 E_main 规范	Data is not conform to standards of E_mail
DUR	不符合 URL 规范	Data is not conform to standards of URL

2.7 动物档案管理系统拦截器设计

拦截器的作用：

1. 做验证、国际化、获取参数等；
2. 验证是为了保证系统运行者的合法性而设计的
3. 具有相同功能的 controller 在多个 controller 中使用，代码重用

2.7.1 动物档案修改拦截器设计

图 2.7.1

拦截 Action 类及方法	验证方法说明
UserNameValidate	对用户登录的姓名进行验证
UserPwdValidate	对用户登录的密码进行验证
UserAuthorityValidate	对用户权限进行验证
AnimalArchiveNumberValidate	对用户输入的查询档案编号进行验证
AnimalArchiveRangeValidate	对用户输入的模糊查询进行验证
AnimalArchiveChangeValidate	对用户提出的档案修改申请进行验证

2.7.2 实验数据修改拦截器设计

图 2.7.1

拦截 Action 类及方法	验证方法说明
UserNameValidate	对用户登录的姓名进行验证
UserPwdValidate	对用户登录的密码进行验证
UserAuthorityValidate	对用户权限进行验证
ExperimentDataNumberValidate	对用户输入的实验数据编号进行验证
ExperimentDataRangeValidate	对用户输入的模糊查询进行验证
ExperimentDataChangeValidate	对用户提出的实验数据申请进行验证

2.8 操作日志设计

操作日志的作用：能够实时查询投保书的操作状态，能够实时跟踪投保书操作进度，方便查询所需信息。

操作日志信息内容：操作员工号、姓名、操作时间、所在部门、完成的操作和操作方法。

表 2.8

Action 类名称	方法	时间点
ImplementsAnimalSearchAction (动物档案检索日志)	Save	保存成功后,完成动物档案查询操作日志保存
ImplementsExperimentSearchAction (档案检索日志)	Save	保存成功后,完成实验数据查询操作日志保存
ImplementsUserAction (用户操作日志)	Save	保存成功后,完成用户登录操作日志保存
ImplementsAnimalArchiveAction (动物档案修改日志)	Save	保存成功后,完成动物档案修改操作日志保存
ImplementsExperimentDataAction (实验数据修改日志)	Save	保存成功后,完成实验数据修改操作日志保存
ImplementsLogAction (日志操作日志)	Save	保存成功后,完成日志操作日志保存

2.9 SpringMVC 标签库设计

表 2.9

标签名称	动物档案管理系统中的应用场景分析
a	用于创建动物档案管理系统中的页面超链接
Controller	在数据库中动态提取数据,包括动物答案编号、实验数据编号等基本的查询信息
ControllerError	用于获取 Controller 异常信息
ControllerMessage	显示 Controller 相关信息

3 Mybatis 在动物档案管理系统中的应用

3.1 Mybatis 应用配置设计

3.1.1 Mybatis 服务配置全局设计

动物档案管理系统采用 mysql，Mybatis 服务配置的全局设计要结合 mysql。

表 3.1.1 Mybatis 参数表

参数名称	参数值	说明
Mybatis 服务器	UIWSysMybatisServer	提供 Mybatis 服务
BusinessLogic 服务器	UIWBusinessLogisServer	业务逻辑服务器
数据库连接配置	JDBC	提供数据库连接服务
数据库名称	UIWSysDBServer	数据库服务器名称

3.1.2 JDBC 数据源用户角色表设计

1. 不同用户角色，具有不同的操作权限。
2. Mybatis 配置 JDBC 连接时，一个 JDBC 连接只能设置一个用户连接。

表 3.1.2 动物档案管理系统 JDBC Data Source 用户角色表

数据库名称	使用角色及操作权限
JDBC_AnimalArchiveManager	部门业务数据查询，动物转平台部分数据查询业务
JDBC_ExperimentDataManager	部门业务数据查询，实验室平台部分数据查询业务
JDBC_LogWorker	业务及数据查询和录入，档案审查员进行相关查询
JDBC_InspectWorker	业务及数据查询和录入，日志整理员进行相关查询

3.2 会话 Mapper 设计表

表 3.2 Mapper 设计表

Mapper 名称	状态	JNDI 名称	说明
ImplementsAnimalArchiveSearchMapper	Statesless	ImplementsSP/remote	动物档案插查询会话 mapper
ImplementsExperimentMapper	Statesless	ImplementsSP/remote	实验数据查询会话 mapper
ImplementsInspectMapper	Statesless	ImplementsSP/remote	档案审查业务会话 mapper
ImplementsLogOperationMapper	Statesless	ImplementsSP/remote	档案日志操作会话 mapper
ImplementsUserStateMapper	Statesless	ImplementsSP/remote	用户状态维持会话 mapper

3.3 实体 Domain 设计

实体 Domain 的配置与数据库有着密切关系，在动物档案管理系统中实体 Domain 承担了数据库设计中数据结构定义的职责，而这个职责必须和系统数据表结构有直接的关联，否则会话 mapper 与数据库的操作过程中会因为与数据结构的不一致而导致数据无法操作，所以实体 Domain 必须是在解读数据表结构设计的基础上。另外，domain 不一定与数据表结构之间有一一对应关系。

表 3.3 动物档案管理系统 domain 与数据表或视图对应关系

Domain 名称	对应表或视图	说明
JB_AnimalArchive	T_AnimalArchive	动物基本信息表
JB_ExperimentData	T_ExperimentData	实验数据信息表
JB_UserInformation	T_UserInformation	用户信息表
JB_InspectData	T_InspectData	审核数据表
JB_LogInformation	T_LogInformation	日志信息表

3.4 Mybatis 安全设计

用户角色的设计过程

1. 确定系统用例中哪些操作是属于哪些参与者
2. 确定在某个用例中，某个参与者需要完成的动作行为，以及这些动作行为操作的对象
3. 根据领域类图，确定参与者所操作的领域类的属性，然后，根据这些属性，定义数据结构，并且对这些数据结构进行操作
4. 综合汇总，编写 Mybatis 中的会话 mapper 或 BusinessLogic 中操作的用户组，编写 mybatis 用户权限表。

表 3.4 mybatis 用户权限表

Bean 名称	方法体	访问角色	说明
ImplementsAnimalArchive	QueryByAnimalArchiveNumber	AnimalTransferDepartment	对转平台部门开放，用于对动物档案进行修改查询操作
	Save	AnimalTransferDepartment	
ImplementsExperimentData	QueryByExperimentDataNumber	LabDepartment	对实验室部门开放，用于对动物档案进行修改查询操作
	Save	LabDepartment	
ImplementsInspectUpdateApplication	QueryByAll	ArchiveInspectStaff	对档案审查员开放，用于对提交的申请进行审核
	Save	ArchiveInspectStaff	
ImplementsLogOperationApplication	QueryByLogNumber	LogOperator	对档案日志操作员开放，用于对更改的日志信息审核
	Save	LogOperator	
	UpdateByLogNumber	LogOperator	
	DeleteByLogNumber	LogOperator	

4 界面元素设计

界面设计原则：

1. 遵循标准一致的原则
2. 通信作为界面设计的核心
3. 能够很好地提供帮助
4. 操作简单
5. 具有较强的容错功能

4.1 动物档案修改页面元素设计

<input type="checkbox"/>	ID	品种	描述	性别	体重(/kg)	体长(/cm)	操作
<input type="checkbox"/>	1	猪	沙子岭猪	公	48	136	编辑 删除
<input type="checkbox"/>	2	猪	沙子岭猪	母	52	137	编辑 删除
<input type="checkbox"/>	3	猪	沙子岭猪	公	55	138	编辑 删除
<input type="checkbox"/>	4	猪	沙子岭猪	公	52	139	编辑 删除
<input type="checkbox"/>	5	猪	沙子岭猪	公	50	140	编辑 删除
<input type="checkbox"/>	6	猪	沙子岭猪	母	52	141	编辑 删除
<input type="checkbox"/>	7	猪	沙子岭猪	公	54	142	编辑 删除

当前第 1 页 总 9 页 总 58 条记录

[首页](#) [«](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [»](#) [末页](#)

动物档案修改

动物id: 4

动物品种: 猪

性别: 公

基本描述: 沙子岭猪

耳标编号: 4

建档日期: 2020-1-7

负责人编号: 4

关键字: 沙子岭猪,育种方案优化

档案描述: 对沙子岭猪育种方案进行优化,以获得目前条件下最优育种方案

体重

52

体长

139

取消

保存

4.1.1 动物档案修改页面元素设计表

表 4.1.1 动物档案修改页面元素设计表

序号	事件	事件名称	事件定位
1	失去焦点	AnimalArchiveNumber onBlur()	动物档案编号 Text 框
2	单击按钮	Select OnClick()	查询动物档案
3	失去焦点	AnimalGender onBlur ()	动物档案编号 Text 框
4	失去焦点	AnimalWeight onBlur ()	动物性别 Text 框
5	失去焦点	AnimalLength onBlur ()	动物体重 Text 框
6	失去焦点	AnimalFeature onBlur ()	动物体长 Text 框
7	失去焦点	HealthRecordDate onBlur ()	特征描述 Text 框
8	失去焦点	HealthDescription onBlur ()	健康记录日期 Text 框
9	失去焦点	HealthLevel onBlur ()	健康描述 Text 框
10	失去焦点	AnimalGender onBlur ()	健康水平 Text 框
11	单击按钮	Reset OnClick()	重置动物档案
12	单击按钮	Submit OnClick()	提交档案修改申请档案
13	单击按钮	Save OnClick()	暂存当前修改档案
14	单击按钮	InspectResult OnClick()	查看审核结果

4.1.2 界面设计与界面元素说明

界面布局，遵循工作习惯，将数据界面排列按照从左到右、从上到下的原则进行页面布局和数据项顺序布局。

表 4.1.2 动物档案修改页面界面元素说明表

标题	标识	数据类型	是否为空	值域	值
动物档案编号	AnimalArchiveNumber	String	否		
动物耳标编号	AnimalEartagNumber	String	否		
动物性别	AnimalGender	String	否	雌/雄	
动物体重	AnimalWeight	Double	否		大于 0，小于 500

动物体长	AnimalLength	Double	否		大于 0，小于 10
特征描述	AnimalFeature	String	否		
健康记录日期	HealthRecordDate	Date	否		
健康描述	HealthDescription	String	否		
健康水平	HealthLevel	String	否		

4.2 实验数据修改页面元素设计

<input type="checkbox"/>	ID	实验者编号	实验日期	实验目的	实验地点	操作	✕批量删除
<input type="checkbox"/>	1	1	2020-3-1	获得目前条件下最优沙子岭猪育种方案	实验室101	 编辑	 删除
<input type="checkbox"/>	2	2	2020-3-1	获得目前条件下最优沙子岭猪育种方案	实验室101	 编辑	 删除
<input type="checkbox"/>	3	3	2020-3-1	获得目前条件下最优沙子岭猪育种方案	实验室101	 编辑	 删除
<input type="checkbox"/>	4	1	2020-3-1	获得目前条件下最优沙子岭猪育种方案	实验室101	 编辑	 删除
<input type="checkbox"/>	5	2	2020-3-1	获得目前条件下最优沙子岭猪育种方案	实验室101	 编辑	 删除
<input type="checkbox"/>	6	3	2020-3-1	获得目前条件下最优沙子岭猪育种方案	实验室101	 编辑	 删除
<input type="checkbox"/>	7	1	2020-3-1	获得目前条件下最优沙子岭猪育种方案	实验室101	 编辑	 删除

当前第 1 页 总 8 页 总 50 条记录

实验数据修改

档案编号: 3

实验者编号: 3

实验日期: 2020-3-1

实验目的
获得目前条件下最优沙子岭猪育种方案

实验方法
根据测定数据, 对沙子岭猪的育种方案进行优化, 选择最佳优化组合的

实验记录
对沙子岭猪育种资料的整理,对沙子岭猪群体进行基础性研究,包括生长发育性状校正公式的制定、遗传参数的估算、边际效益的计算及简化综合育种值

实验结果
Saturation模型为拟合沙子岭猪早期生长动态的最优模型

取消

保存

4.2.1 实验数据修改页面元素设计表

表 4.2.1 动物档案修改页面元素设计表

序号	事件	事件名称	事件定位
1	失去焦点	ExperimentDataNumber onBlur()	实验数据编号 Text 框
2	单击按钮	Select OnClick()	查询实验数据档案
3	失去焦点	ExperimentPurpose onBlur()	实验目的 Text 框
4	失去焦点	ExperimentLocation onBlur()	实验地点 Text 框
5	失去焦点	ExperimentApproach onBlur()	实验方案 Text 框
6	失去焦点	ExperimentReflection onBlur()	实验反应 Text 框
7	失去焦点	ExperimentRecord onBlur()	情况记录 Text 框
8	失去焦点	ExperimentOtherNote onBlur()	其他备注 Text 框
9	失去焦点	ExperimentDate onBlur()	实验日期 Text 框

10	单击按钮	Reset OnClick()	重置动物档案
11	单击按钮	Submit OnClick()	提交档案修改申请档案
12	单击按钮	Save OnClick()	暂存当前修改档案
13	单击按钮	InspectResult OnClick()	查看审核结果

4.2.2 界面设计与界面元素说明

表 4.2.2 实验数据修改页面界面元素说明表

标题	标识	数据类型	是否为空	值域	值
实验数据编号	ExperimentDataNumber	String	否		
实验目的	ExperimentPurpose	String	否		
实验地点	ExperimentLocation	String	否		
实验方案	ExperimentApproach	String	否		
实验反应	ExperimentReflection	String	否		
情况记录	ExperimentRecord	String	否		
其他备注	ExperimentOtherNote	String	否		
实验日期	ExperimentDate	Date	否		

4.3 操作日志页面元素设计

ID	操作员	部门	操作原因
30	3	实验室	更新数据
31	4	动物房	更新数据
32	5	动物房	更新数据
33	6	动物房	更新数据
34	7	动物房	更新数据
35	14	转平台	更新数据
36	15	转平台	更新数据

4.3.1 操作日志页面元素设计表

表 4.3.1 操作日志页面元素设计表

序号	事件	事件名称	事件定位
1	失去焦点	LogNumber onBlur()	档案编号 Text 框
2	单击按钮	Select OnClick()	查询动物档案
3	失去焦点	UserInformationName onBlur()	用户姓名 Text 框
4	失去焦点	UserInformationPhone onBlur()	用户联系方式 Text 框
5	失去焦点	UserInformationType onBlur()	用户类别 Text 框
6	失去焦点	UserInformationNumber onBlur()	用户编号 Text 框
7	失去焦点	UserInformationName onBlur()	用户姓名 Text 框
8	失去焦点	UserInformationType onBlur()	用户类别 Text 框
9	失去焦点	ArchiveUpdateDepartment onBlur()	档案更新所在部门 Text 框
10	失去焦点	ArchiveUpdateReason onBlur()	档案更新原因 Text 框
11	失去焦点	ArchiveUpdateType onBlur()	档案更新的类型 Text 框
12	失去焦点	ArchiveUpdateItems onBlur()	档案更新详情 Text 框
13	失去焦点	ArchiveUpdateDate onBlur()	档案更新日期 Text 框
14	单击按钮	Reset OnClick()	重置日志
16	单击按钮	Save OnClick()	保存修改档案

4.3.2 界面设计与界面元素说明

表 4.3.2 操作日志页面界面元素说明表

标题	标识	数据类型	是否为空	值域	值
档案编号	LogNumber	String	否		
用户姓名	UserInformationName	String	否		
用户联系方式	UserInformationPhone	String	否		
用户类别	UserInformationType	String	否		

用户编号	UserInformationNumber	String	否		
用户姓名	UserInformationName	String	否		
用户类别	UserInformationType	String	否		
档案更新所在部门	ArchiveUpdateDepartment	String	否		
档案更新原因	ArchiveUpdateReason	String	否		
档案更新的类型	ArchiveUpdateType	String	否		
档案更新详情	ArchiveUpdateItems	String	否		
档案更新日期	ArchiveUpdateDate	Date	否		

5 状态实现

一个对象的状态表现形式有两种：

1. 作为状态作为历史记录永久保存下来，持久状态
2. 在系统运行时的临时状态

动物档案管理系统基础状态是“已建档”，只有在“已建档”状态才可以进行对于动物的一系列操作，操作成功后，在操作日志中添加相应的状态。

5.1 动物档案修改转态改变表

表 5.1 动物档案修改转态改变表

状态名称	类	方法	策略	表	字段
查询	ImplementsAnimalArchivementsMapper	QueryByAnimalArchiveNumber	锁定	T_AnimalArchive	AnimalArchiveState
修改	ImplementsAnimalArchivementsMapper	Save	记录	T_AnimalArchive	AnimalArchiveState

5.2 实验数据修改转态改变表

表 5.2 实验数据修改转态改变表

状态名称	类	方法	策略	表	字段
查询	ImplementsperimentDataMapper	QueryByExperimentDataNumber	锁定	T_ExperimentData	T_ExperimentData
修改	ImplementsperimentDataMapper	Save	记录	T_ExperimentData	T_ExperimentData

6 数据结构详细设计

以子系统为设计主题，选择每个类包含方法体的输入参数、输出参数

针对每个方法体所要完成的功能，并且分析这些方法体中所操作的数据有哪些，以提高性能为目标而确定它们的组织形式，包括 Java 集合，基本数据类型和 mapper 对这些数据元素命名

6.1 动物档案修改数据元素结构设计

6.1.1 表示层数据元素结构设计

页面名称：V_AnimalArchivePage.jsp

1. AnimalArchiveNumber onbul()

AnimalArchiveNumber onbul(), 它表示输入动物档案编号，用于检索动物档案。

(1) 输入:

数据类型为 string

参数名称为 AnimalArchiveNumber, 动物档案编号

(2) 返回:

数据类型为 AnimalArchiveBaseInfo

2. AnimalGender onblur ()

AnimalGender onbul(), 它表示输入动物性别, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 AnimalGender, 动物性别

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

3. AnimalWeight onblur ()

AnimalWeight onbul(), 它表示输入动物体重, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 AnimalWeight, 动物体重

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

4. AnimalLength onblur ()

AnimalLength onbul(), 它表示输入动物体长, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 AnimalLength, 动物体长

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

5. AnimalFeature onblur ()

AnimalFeature onbul(), 它表示输入动物特征, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 AnimalFeature, 动物特征

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

6. HealthRecordDate onblur ()

HealthRecordDate onbul(), 它表示输入动物健康记录日期, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 HealthRecordDate, 动物体重

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

7. HealthDescription onblur ()

HealthDescription onbul(), 它表示输入动物描述, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 HealthDescription, 动物描述

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

8. HealthLevel onblur ()

HealthLevel onbul(), 它表示输入动物健康等级, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 HealthLevel, 动物健康等级

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

6.1.2 控制层数据元素结构设计

Controller 程序名称: ImplementsAnimalArchiveController.java (动物档案基本信息)

1. 查询动物档案

功能描述: 按照动物档案编号进行档案查询。

(1) 输入:

数据类型为 string

数据名称为 AnimalArchiveNumber

数据元素为动物档案编号 (AnimalArchiveNumber)

(2) 返回:

数据类型为 AnimalArchiveBaseInfo

数据元素为动物档案信息

2. 保存动物档案

功能描述: 按照动物档案编号进行档案保存。

(1) 输入:

数据类型为 AnimalArchiveBaseInfo

数据名称为 AnimalArchiveBaseInfo

数据元素为动物档案信息 (AnimalArchiveBaseInfo)

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

6.1.3 业务逻辑层数据元素结构设计

程序名称: ImplementsAnimalArchiveInfomapper

功能描述: 包括动物档案的查询和修改功能

(1) 输入:

数据类型为 Java

数据结构名称为 JB_AnimalArchiveBaseInfo

数据元素为: 动物档案编号+动物耳标编号+责任研究员工号+所属 PI+动物性别+动物体重+动物体长+特征描述+健康记录日期+健康描述+健康水平

(2) 输出

数据类型为 String

数据元素为 “True” 或 “false”

(3) 操作数据表: T_AnimalArchiveBaseInfo

6.2 实验数据修改数据元素结构设计

6.2.1 表示层数据元素结构设计

页面名称: V_ExperimentDataPage.jsp

1. ExperimentDataNumber onbul()

ExperimentDataNumber onbul(), 它表示输入实验档案编号, 用于检索实验档案。

(1) 输入:

数据类型为 string

参数名称为 ExperimentDataNumber, 实验档案编号

(2) 返回:

数据类型为 ExperimentDataBaseInfo

2. ExperimentPurpose onblur()

ExperimentPurpose onbul(), 它表示输入实验目的, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 ExperimentPurpose, 动物实验目的

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

3. ExperimentLocation onblur()

ExperimentLocation onbul(), 它表示输入实验地点, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 ExperimentLocation, 实验地点

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

4. ExperimentApproach onblur()

ExperimentApproach onbul(), 它表示输入动实验方法, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 ExperimentApproach, 实验方法

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

5. ExperimentReflection onbul()

ExperimentReflection onbul(), 它表示输入实验反应, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 ExperimentReflection, 实验反应

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

6. ExperimentRecord onbul()

ExperimentReflection onbul(), 它表示输入实验记录, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 ExperimentReflection, 实验记录

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

7. ExperimentOtherNote onbul()

ExperimentOtherNote onbul(), 它表示输入其他信息, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 ExperimentOtherNote, 其他信息

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

8. ExperimentDate onBlur()

ExperimentDate onBlur(), 它表示输入实验日期你, 用于保存数据。

(1) 输入:

数据类型为 string

参数名称为 ExperimentDate, 实验日期

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

6.2.2 控制层数据元素结构设计

Controller 程序名称: ImplementsExperimentDataController.java (动物档案基本信息)

1. 查询动物档案

功能描述: 按照实验档案编号进行档案查询。

(1) 输入:

数据类型为 string

数据名称为 ExperimentDataNumber

数据元素为动物档案编号 (ExperimentDataNumber)

(2) 返回:

数据类型为 ExperimentDataBaseInfo

数据元素为动物档案信息

2. 保存实验数据

功能描述: 按照实验档案编号进行档案保存。

(1) 输入:

数据类型为 ExperimentDataBaseInfo

数据名称为 ExperimentDataBaseInfo

数据元素为动物档案信息 (ExperimentDataBaseInfo)

(2) 返回:

数据类型为 boolean

数据名称为 true or false, 格式正确, 则返回值为.t., 否则返回.f.

6.2.3 业务逻辑层数据元素结构设计

程序名称: `ImplementsExperimentDataInfomapper`

功能描述: 包括实验数据的查询和修改功能

(1) 输入:

数据类型为 `Java`

数据结构名称为 `JB_ExperimentDataBaseInfo`

数据元素为: 动物档案编号+用户编号+用户姓名+用户联系方式+用户类别+实验目的+实验地点+实验方案+实验反应+情况记录+其他备注+实验日期

(2) 输出

数据类型为 `String`

数据元素为 “True” 或 “false”

(3) 操作数据表: `T_ExperimentDataBaseInfo`

7 系统运行详细设计

7.1 活动图在系统详细设计中的体现

1. 状态图的实现, 最后由业务逻辑层来完成
2. 活动图由表现层完成, 定位到表现层某个方法体来完成。
3. 业务逻辑层只负责业务逻辑运算, 将业务逻辑的结果值返回到 `controller` 层中, 如何跳转由 `controller` 层负责调度。
4. 页面层的规则校验策略: 不符合输入规则的, 通过弹出对话框的方式提示用户重新输入, 并且返回到输入框的原焦点上; 对于功能项连接, 建议不使用网页连接的方式, 而是将连接定义为一种事件, 然后再调用 `controller` 层, 由 `controller` 指定连接, 封装了网络地址细节, 提高安全性。
5. 异常处理: 不符合业务逻辑而出现的异常, 通常采用在页面端弹出异常对话框, 然后由用户进行再次处理, 指导输入正确的业务逻辑数据为止或终止系统运行; 程序逻辑异常, 对异常归类, 按照架构设计中异常处理机制, 将异常信息保存在数据库中, 并且通过中文方式显示异常类型, 对已经操作的数据回滚处理, 最后结束运行程序。

7.2 系统运行设计

7.2.1 定位活动对应的架构层次

1. 动物档案修改业务

(1) 档案查询验证:

由业务逻辑层检索, 并且判断是否为空, 如果为空, 则调用 Error.jsp, 并且传递参数“档案查询信息为空”于 Error.jsp, 如果不为空, 则通过 Servlet 判断档案是否存在, 正确则完成页面跳转, 否则 Error.jsp, 并且传递参数“档案信息不存在”于 Error.jsp

(2) 档案修改验证:

对用户修改的档案内容进行合法性检验, 由页面层完成校验, 通过 SpringMVC 中 Validate 负责校验。

2. 实验数据修改业务

(1) 档案查询验证:

由业务逻辑层检索, 并且判断是否为空, 如果为空, 则调用 Error.jsp, 并且传递参数“实验数据信息为空”于 Error.jsp, 如果不为空, 则通过 Servlet 判断档案是否存在, 正确则完成页面跳转, 否则 Error.jsp, 并且传递参数“实验信息不存在”于 Error.jsp

(2) 实验数据修改验证:

对用户修改的实验数据进行合法性检验, 由页面层完成校验, 通过 SpringMVC 中 Validate 负责校验。

7.2.2 定位不同的返回值页面

1. 动物档案修改业务

(1) 动物档案查询

页面层方法体: V_AnimalArchive.AnimalArchiveInfoQuery

控制层方法体: ImplementsAnimalArchiveAction.QueryByAnimalArchiveNumber

业务层方法体: ImplementsAnimalArchiveBean.QueryByAnimalArchiveNumber

controller 返回值类型及页面跳转:

If Resultlist.isEmpty()

{Error.jsp “查无此动物档案” }

Else

{V_AnimalArchivePage 显示 ResultList}

(2) 动物档案修改

页面层方法体: V_AnimalArchivePage.Save

控制层方法体: ImplementsAnimalArchiveController.Save

业务层方法体: ImplementsAnimalArchiveMapper.Save

controller 返回值类型及页面跳转:

If SaveState

{返回 V_AnimalArchivePage.jsp}

Else

{Error.jsp “保存失败” }

2. 实验数据修改业务

(1) 实验数据查询

页面层方法体: V_ExperimentData.ExperimentDataInfoQuery

控制层方法体: ImplementsExperimentDataAction.QueryByExperimentDataNumber

业务层方法体: ImplementsExperimentDataBean.QueryByExperimentDataNumber

controller 返回值类型及页面跳转:

If Resultlist.isEmpty()

{Error.jsp “查无此实验档案” }

Else

{V_ExperimentDataPage 显示 ResultList}

(2) 动物档案修改

页面层方法体: V_ExperimentDataPage.Save

控制层方法体: ImplementsExperimentDataController.Save

业务层方法体: ImplementsExperimentDataMapper.Save

controller 返回值类型及页面跳转:

If SaveState

```
{返回 V_ExperimentDataPage.jsp}
```

```
Else
```

```
{Error.jsp “保存失败” }
```

7.2.3 异常处理

1. 程序逻辑异常的返回值全部为.f.
2. 要求将异常信息保存到数据库的同时，调用 Error.jsp “异常信息”。

8 算法设计

8.1 分析类间关系

类间关系如下图

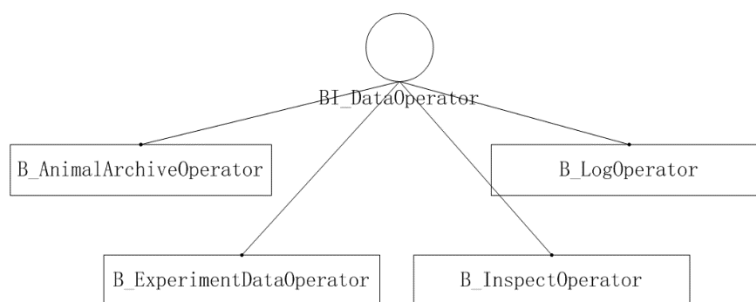


图 8.1

(1) BI_DataOperator 数据处理接口方法体描述

```
Public interface BI_DataOperator
```

```
{
```

```
List QueryByIdNumber(String Number) //按照 ID 查找
```

```
List QueryByDate(Date QueryDate) //按照日期查找
```

```
List QueryByDepartment(String DepartmentName) //按照部门查找
```

```
List QueryByName(String Name) //按照名字查找
```

```
Boolean SaveChange()    //保存数据库  
}
```

(2) B_AnimalArchiveOperator 动物档案操作

```
Public class B_AnimalArchiveOperator implements BI_DataOperator  
{  
List QueryByIdNumber(String Number) //按照 ID 查找  
List QueryByDate(Date QueryDate)    //按照日期查找  
List QueryByDepartment(String DepartmentName) //按照部门查找  
List QueryByName(String Name) //按照名字查找  
Boolean SaveChange()    //保存数据库  
}
```

(3) B_ExperimentDataOperator 实验档案操作

```
Public class B_AnimalArchiveOperator implements BI_DataOperator  
{  
List QueryByIdNumber(String Number) //按照 ID 查找  
List QueryByDate(Date QueryDate)    //按照日期查找  
List QueryByDepartment(String DepartmentName) //按照部门查找  
List QueryByName(String Name) //按照名字查找  
Boolean SaveChange()    //保存数据库  
}
```

(4) B_InspectOperator 审查操作

```
Public class B_AnimalArchiveOperator implements BI_DataOperator  
{  
List QueryByIdNumber(String Number) //按照 ID 查找  
List QueryByDate(Date QueryDate)    //按照日期查找  
List QueryByDepartment(String DepartmentName) //按照部门查找  
List QueryByName(String Name) //按照名字查找  
Boolean SaveChange()    //保存数据库  
}
```

(5) B_LogOperator 日志操作

```
Public class B_AnimalArchiveOperator implements BI_DataOperator
```



```
{  
List QueryByIdNumber(String Number) //按照 ID 查找  
List QueryByDate(Date QueryDate) //按照日期查找  
List QueryByDepartment(String DepartmentName) //按照部门查找  
List QueryByName(String Name) //按照名字查找  
Boolean SaveChange() //保存数据库  
}
```

8.2 分析算法场景

目标：提高保险公司工作效率和业务水平而设计。

数据的交互以数据库交互为主，包括：数据排序，检索，数据结构组织、数据排序、数据检索，科学合理使用 SQL 语句。

在本系统中，主要使用到的算法应该为数据检索与数据排序。因此应当合理的使用 SQL 语句，尽量减少笛卡尔积操作。

8.3 描述实现数据结构

（1）实验动物档案系统输入参数

本系统中的输入参数为

- A. 动物档案相关信息
- B. 实验数据相关信息
- C. 审查文件相关信息
- D. 日志操作相关信息

（2）操作数据表

- A. 动物档案表查询

操作表：动物信息视图 V_AnimalInformation

关联表：无

- B. 动物档案表修改

操作表：动物信息视图 V_AnimalInformation

关联表：无

C. 实验数据表查询

操作表：实验信息视图 V_ExperimentData

关联表：无

D. 实验数据表修改

操作表：实验信息视图 V_ExperimentData

关联表：无

E. 审查文件表生成

生成表：审核文件表 T_InspectResult

关联表：无

F. 日志表生成

生成表：实验信息视图 T_Log

关联表：无

(3) 输出数据结构

A. 动物档案表查询数据元素

数据类型：ArrayList

数据元素：动物档案编号+动物耳标编号+责任研究员工号+所属 PI+动物性别+动物体重+动物体长+特征描述+健康记录日期+健康描述+健康水平

B. 动物档案表修改数据元素

数据类型：ArrayList

数据元素：动物档案编号+动物耳标编号+责任研究员工号+所属 PI+动物性别+动物体重+动物体长+特征描述+健康记录日期+健康描述+健康水平

C. 实验数据表查询数据元素

数据类型：ArrayList

数据元素：动物档案编号+用户编号+用户姓名+用户联系方式+用户类别+实验目的+实验地点+实验方案+实验反应+情况记录+其他备注+实验日期

D. 实验数据表修改数据元素

数据类型：ArrayList

数据元素：动物档案编号+用户编号+用户姓名+用户联系方式+用户类别+实验目的+实验地点+实验方案+实验反应+情况记录+其他备注+实验日期

E. 审查文件表生成数据元素

数据类型：ArrayList

数据元素：审核编号+审核项目种类+审核结果+审核意见+审核日期+审核项目

F. 日志表生成数据元素

数据类型：ArrayList

数据元素：日志编号+修改种类+修改结果+修改项目

8.4 业务算法设计

由于动物档案修改业务不涉及计算问题，因此没有计算规则

（1）输入参数规则

A. 动物档案修改业务类

AnimalArchiveNumber（动物档案编号） \leq 动物档案最大编号

AnimalGender（动物性别）= "雌" 或 "雄"

AnimalWeight（动物体重）介于 0 到 200 kg

AnimalLength（动物体长）介于 0 到 5m

HealthRecordDate（健康记录日期） \leq 当前日期

HealthLevel（健康等级）为正整数且小于等于 5

B. 实验数据修改业务类

ExperimentDataNumber（实验数据编号） \leq 实验数据最大编号

ExperimentDate \leq 当前日期

C. 审核与日志类

InspectDate（审查日期）= 当前日期

LogDate（日志日期）= 当前日期

8.5 确定并描述程序算法

从程序逻辑角度来分析算法，程序逻辑算法需要考虑问题：空间复杂度和时间复杂度。

（1）检索和排序

① 排序问题：

在数据库中读取数据的过程中将数据无排序的读取到内存中，然后按照各种算法进行排序；

在数据库读取数据的过程中，直接使用 SQL 语句中的 ORDER BY 或 GROUP BY 语句进行排序，然后按照数据集顺序采取先进先出或后进后出的方法存储到内存中

② 检索问题：

将数据读取到本地内存中，然后按照一定的程序算法将其进行重新检索

在数据库中创建索引，然后根据索引来获得检索

（2） 数组数据的存储和检索

在空间复杂度方面，尽量减少从数据库中提取数据记录的数量，为了降低空间复杂度，在数据检索过程中，做到了对数据提取的准确定位。

为了减少时间复杂度，在准确定位检索条件的前提下，对于方法体返回的数组严格按照实际列宽数据元素数量长度进行定义，保证了开辟内存一点都没有浪费。

（3） 算法选择

在动物档案管理系统中，并不存在复杂的算法问题。因此只需要按照常规的方法进行数据库操作即可。