Project Name		Confidentiality Level
Contract Management System		Only for Recipients Reference
Project ID	Version	Document Code
BP	1.0	Project ID_SD_001

Contract Management System
Software System Design Specification

Prepared by		Date	
Reviewed by		Date	
Approved by		Date	

软酷卓越实验室

Copyright Restricted and No Reproduction Allowed
Copyright © Ruankosoft Technologies(Shenzhen), Co., Ltd.
. All Rights Reserved

Catalog

软酷卓越实验室

1 Introduction

1.1 Purpose

This document describes the Contract Management System's design process, including general design and detailed design, in order to conduct project team to implement coding and unit testing. Expected reader of this specification is intermediate users (refers to project team member, client representative, testing staff, QA and etc.).

1.2 Scope

1.2.1 Name

Contract Management System.

1.2.2 Functions

Software functions please refer to the requirements specification document: requirement analysis of Contract Management System.

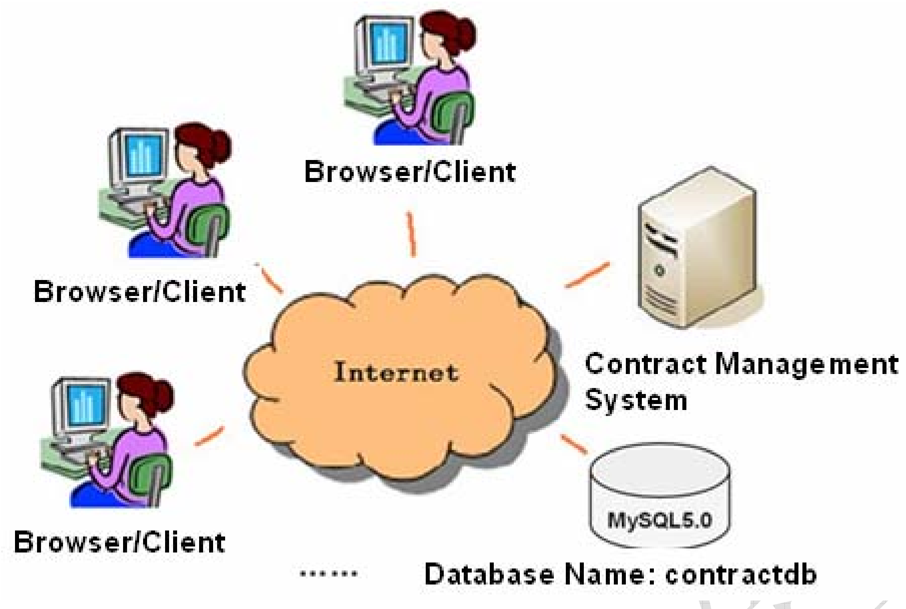
1.2.3 Applications

This software is a web application. The system uses the b/s (Browser/Server)structure, users only need to access the website through the browser.

1.3 Software System Context Definition

Contract Management system is a system of B/S structure, the access structure of system as follow:

软酷卓越实验室



1.4 Design Considerations

1.4.1 Design Alternatives

1. three layer structure of system
A system can be divided into view layer, business logic layer, and data access layer, packets respectively are (com.ruanko.web),(com.ruanko.service),(com.ruanko.dao), they call business model class(is com.ruanko.modelthe class of packets) to transfer the data, some of classes are used publicly in several places, as a tool, Then put them together in a com.ruanko.utils packets.

2. the development model of project and three layer structure
A system use Model 2 development Model: JSP+Servlet+JavaBean(MVC), it combines with three layer structure, servlet (controller) and JSP page (view) pose as view layer, JavaBean is model (including business logic, data access and entity model class and so on) pose as business logic layer and data access layer.

3. the directory structure of project
the packet hierarchical structure of combine three layer structure with Model 2:

project structure	directory
View Layer(UI)	JSP Files
	com.ruanko.web
Business Logic Layer(BLL)	com.ruanko.service
Data Access Layer(DAL)	com.ruanko.dao com.ruanko.dao.impl
Entity	com.ruanko.model

软酷卓越实验室

the directory of folder:

folder	Brief descriptions
css	to store style files
images	to store picture files
js	to store script files
upload	to store files to be uploaded

1.4.2 Design Constraints

1.4.2.1 Standards compliance

Could expand the specifications that do not exist in the following requirements, but it cannot contrary to the standard. It follows: <COE technical requirement standard of Ruanko Lab>, <COE projectming standard requirement of Ruanko Lab>.

1.4.2.2 Hardware Limitations

The minimum configuration of the hardware:
CPU: 1GHZ
RAM: 128MB
To ensure that the system can run smoothly, the configuration best meet the following requirements:
CPU: 1.8GHZ
RAM: 1G

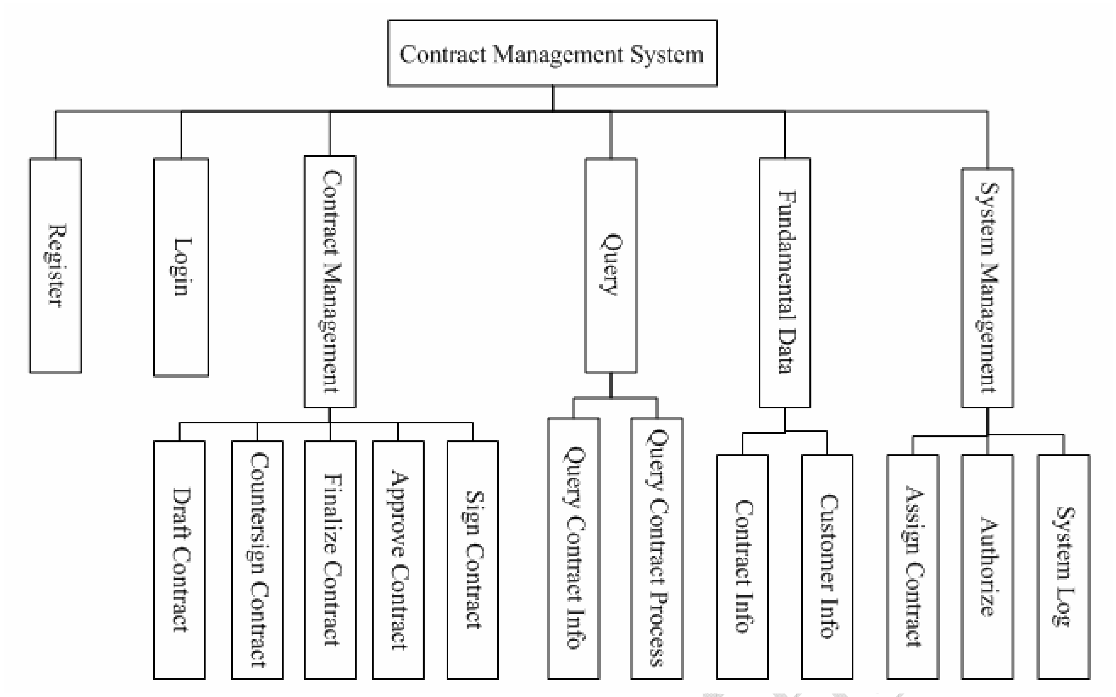
1.4.2.3 Technology Limitations

Database: MySQL5.X
Coding standard: COE projectming standard requirement of Ruanko Lab.

软酷卓越实验室

2 Level 1 Design Description

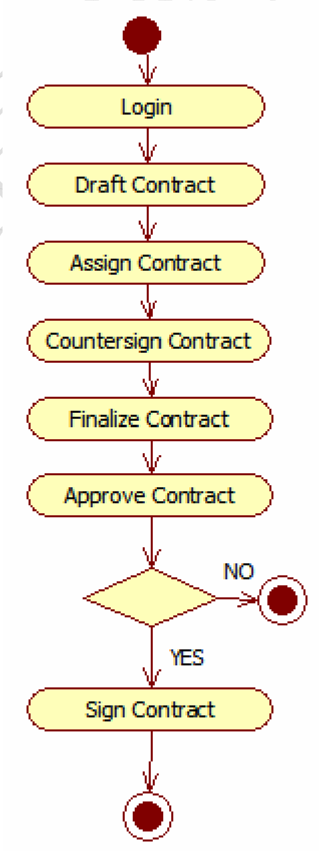
1.System Architecture
The functions are divided into several modules to manage and develop respectively. System is divided into 6 modules: Register, Login, Contract Management, Query, Fundamental Data Management, System Management.The detailed modules of system are as follows:



2. Representation of the Business Flow

the flow of contract management: draft contract, assign contract,countersign contract,finalize contract,approve contract and sign contract.

the flow of contract management as follow:

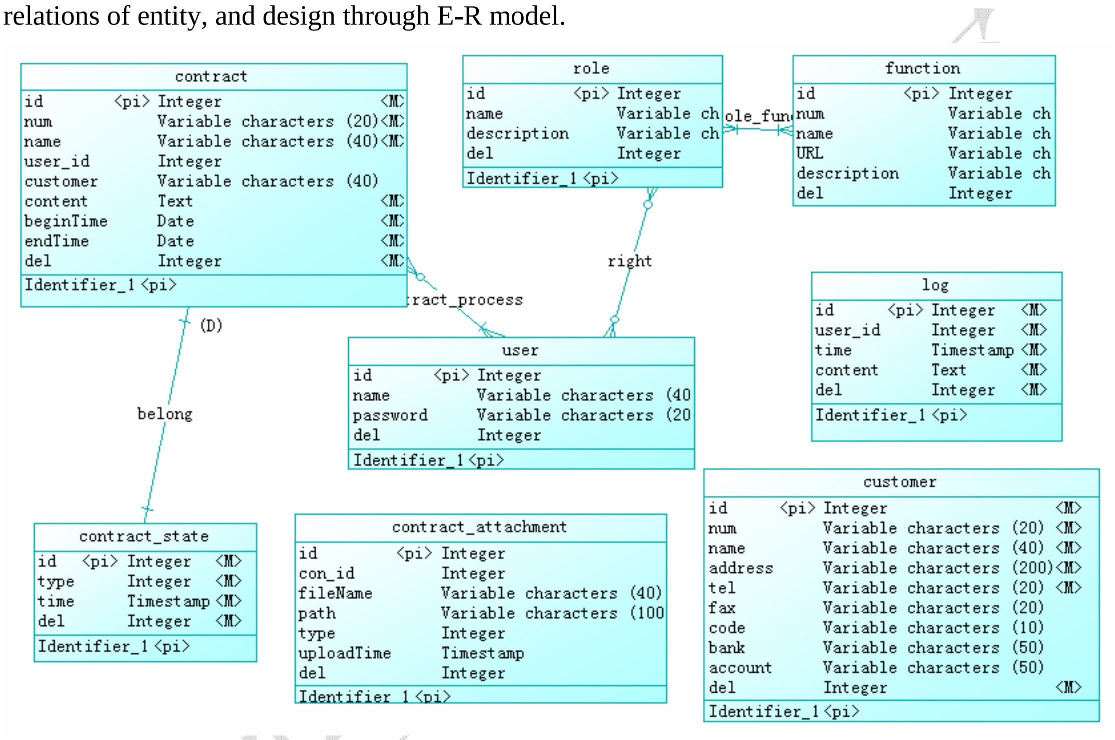


3 Database Design

According to the requirements and business process of contract management system, the system data information may includes the operation of page, dispose process, save the data, etc, extracting and analyzing the data information which refer to these processes, design the project database.

3.1 Conceptual Data Model

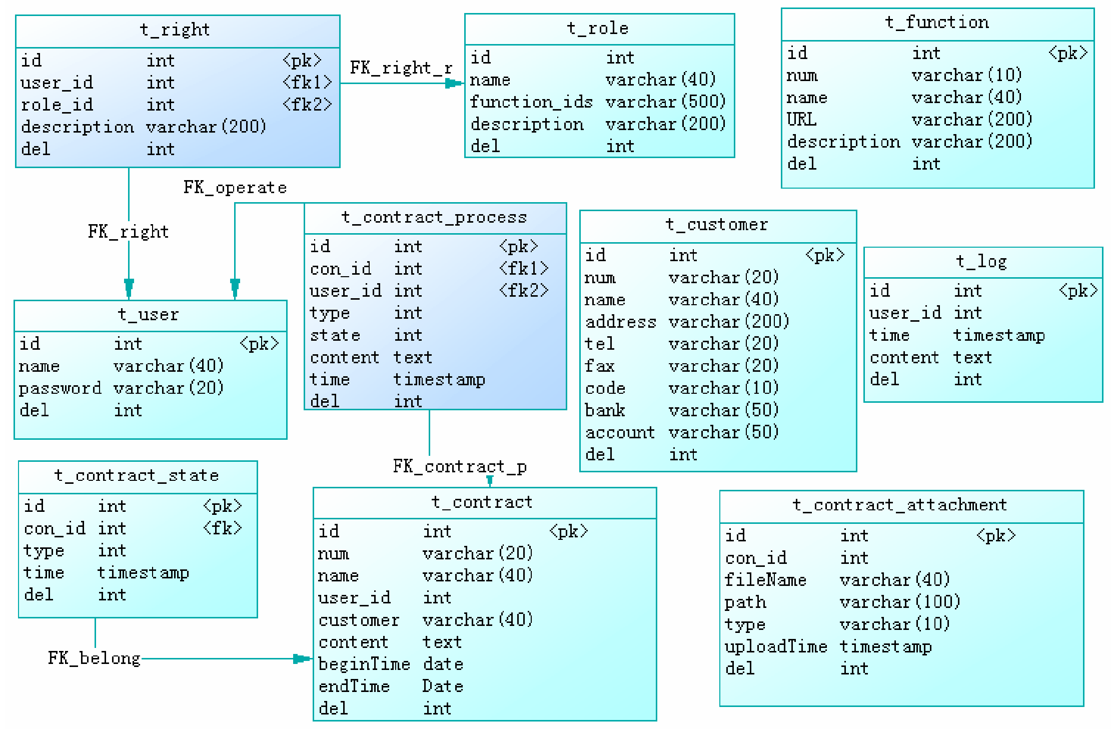
Based on the data dictionaries in extracting, identify the entity, the properties of entity and relations of entity, and design through E-R model.



软酷

3.2 Physical Data Model

The following is based on the MySQL physical data model.



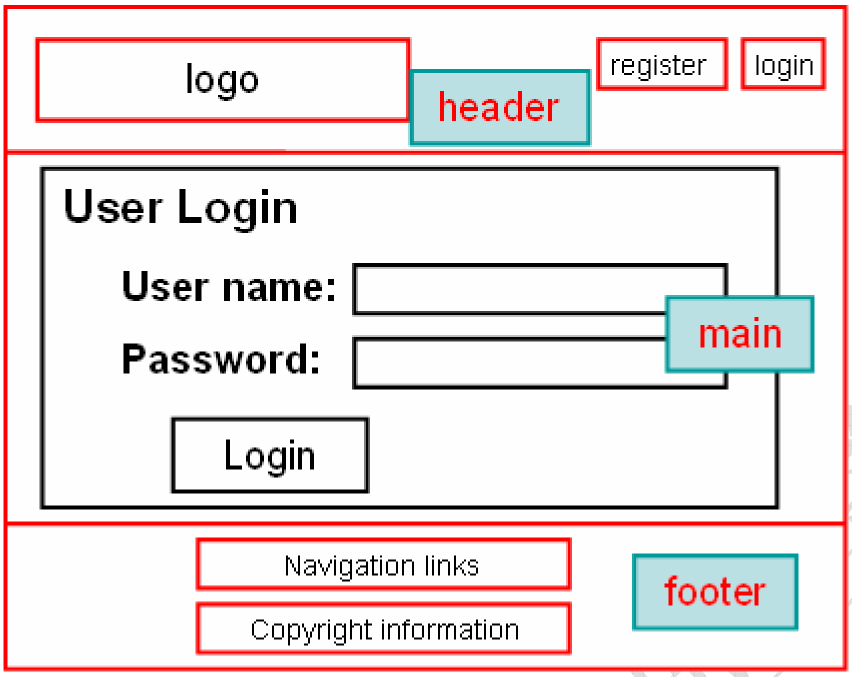
4 UI Design

According to the requirements, design the following pages of Contract Management System:
(1) Register Page and Login Page
(2) Contract Management Page
The pages of Draft, Countersign, Finalize, Approve, Sign, etc. Contract Details Page, Contract List Page, Countersign Comment Page.
(3) System Management Page
Assign Contract Page, Permission Management Page, Log Management Page.

4.1 Login Page

1. The original interface

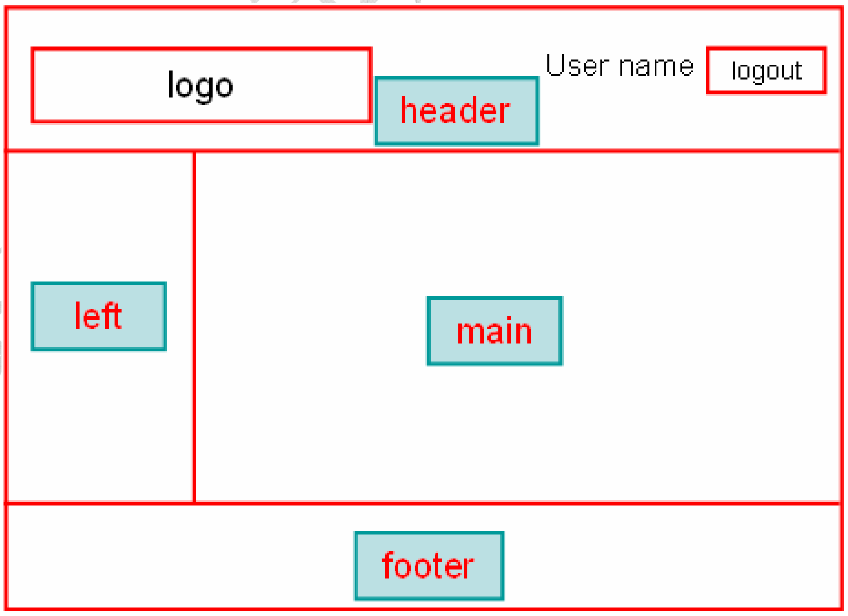
软酷卓越实验室



2. Interface specification
The body is the corresponding form to receive the register information and login information.

4.2 Management Page

1. The original interface

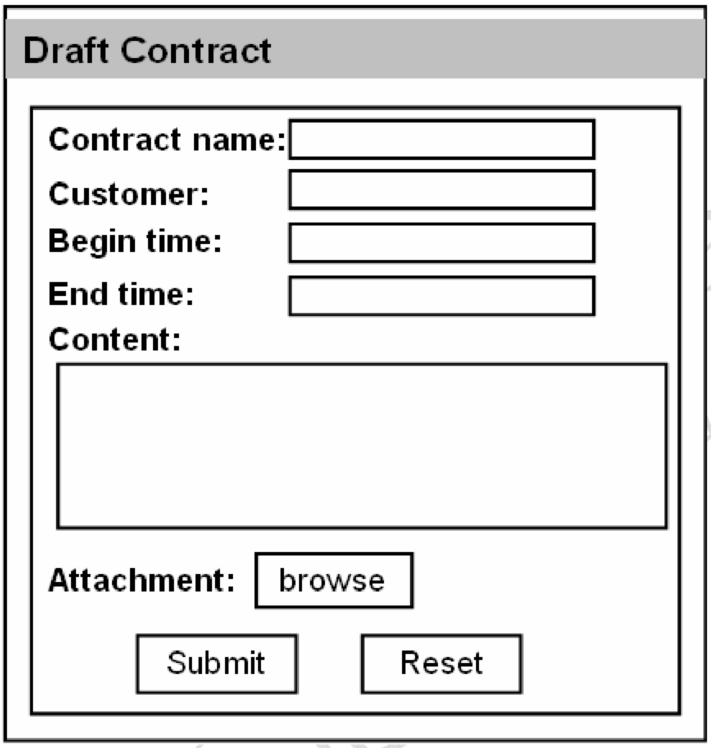


2. Interface specification
The layout of Management Page is implemented with HTML framework page Frame.
The body is divided into the left and right.

When the left management hyperlink is clicked, load the corresponding function page in the right.

4.3 Add Contract Page

1. The original interface

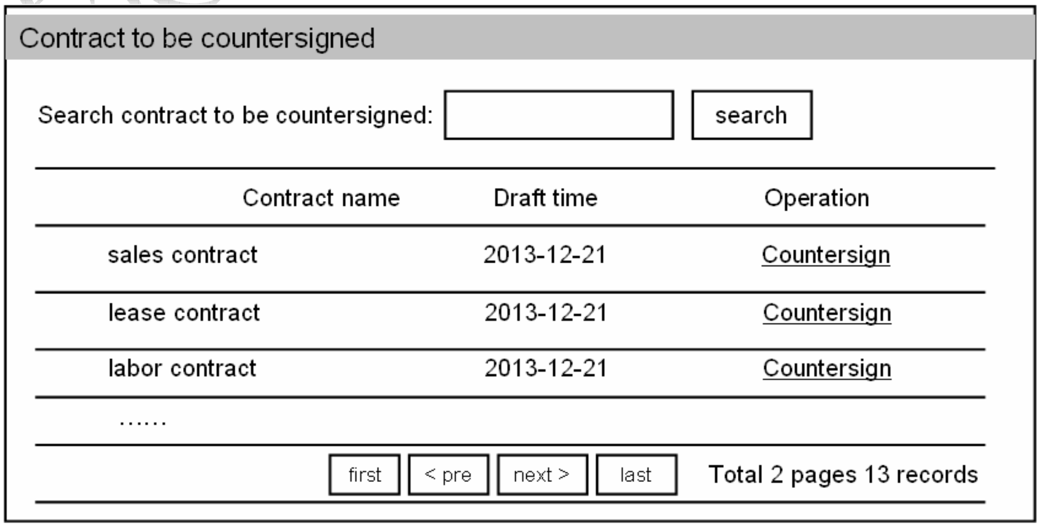


2. Interface specification

The body is the contract form. Its layout is implemented with <table>.

4.4 Contract List To Be Countersigned Page

1. The original interface


Contract to be countersigned
Search contract to be countersigned:
search

Contract name	Draft time	Operation
sales contract	2013-12-21	Countersign
lease contract	2013-12-21	Countersign
labor contract	2013-12-21	Countersign
……		

Total 2 pages 13 records

2. Interface specification
Display the contract information as list. The contract can be queried according to the condition.

5 Detailed Design of Module

5.1 Entity Class Design

The entity class is used to transfer data between layers, one part of entity class correspond to database table, the others based on the requirement of actual business, set up business entity class. The corresponding entity of database table:
Class:User, Contract, ConAttach,ConProcess,ConState,Customer,Function,Log,Right and Role.
Based on the requirements of business, set up the entity class of business:
ConBusiModel, ConDetailBusiModel, CsignatureOpinion and PermissionBusiModel, etc.

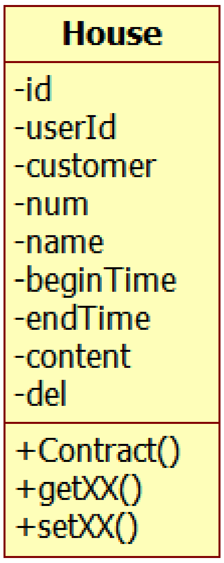
In the designing of entity class, the class consists of three parts.
(1) Each class includes private member variables.
(2) The function of getXX() and setXX(): used for access the private member variable of class (properties), set up respective private variable corresponding to the function of getXX() and setXX().
(3) No parameters constructed function: used to initialization private member variable, give a member variable an initial value.
The entity class is saved in the com.ruanko.model package.

The design of each entity class is consistent, The following Contract entity class as an example to design.

1. Overview
Contract entity class: used to set up the mapping relation between database table (contract basic information table), declare private properties and corresponding field in database table, provide initialize constructed function, setter and getter function is provided for external call, the entity class in project as the carrier of data transmission.

2. Class Diagram

软酷卓越实验室



3. Attributes

The common data structure should be described here.

Visibility	Name	Type	Brief descriptions
private	id	int	ID
private	userId	int	user id
private	customer	String	customer
private	num	String	contract number
private	name	String	contract name
private	beginTime	Date	begin time
private	endTime	Date	end time
private	content	String	contract content
private	del	int	delete status(0-not deleted, 1-deleted)

4. Methods

In entity class can provide setter and getter for external call for these private properties, not call directly through property name. Through the setter function to assignment for the entity attribute. Use setter function gets the property value of entity. In the following, let’s take the contract id for example to illustrate, the other property are similar.

(1) setId()

① Method Descriptions

Prototype	public void setId()
Description	sets the attribute's value
Calls	
Input	id
Output	void
Return	void
Exception	void

软酷卓越实验室

② Implementation Description

If the contract is an example of Contract: contract.setId(1) indicates set the id property of contract as 1.

(2) getId()

① Method Descriptions

Prototype	public int getId()
Description	gets the attribute's value
Calls	
Input	void
Output	void
Return	id
Exception	void

② Implementation Description

Assume Contract is an example of Contract: Contract.getId() indicates get the property value of contract id.

(3) Contract()

① Method Descriptions

Prototype	public Contract()
Description	assigns initial values to object attributes
Calls	
Input	void
Output	void
Return	void
Exception	void

② Implementation Description

Define a constructed function of entity, the function name is the same as the class name:public Contract(), in constructed function can assign a specifying value to property according to the requirements.

软酷卓越实验室

5.2 View Layer Class Design

View layer, call the methods of business logic layer, as deal with the requirement of user, including handle users, Servlet of request for contract operations ect, operation, Servlet for deal with page jump.

Servlet as a controller in the view layer, Servlet class save in com.ruanko.web package.

5.3 Business Logic Layer Class Design

Call data access layer methods, handle users, contract operation etc, provide service for view layer, including UserService,ContractService, etc.
Business logic layer class is saved in com.ruanko.service packet.

5.4 Data Access Layer Class Design

Define the methods of database operation though interface in the data access layer, the detailed entity class of the interface achieves logic mehtod, with the interactions of the database, achieve the accessing operation of data.
Including the interface and corresponding realization of user register, log in, contract management and system management,ect.
Interface/Classes:UserDao/UserDaoImpl,ContractDao/ContractDaoImpl,ConProcessDao/ConProcessDaoImpl,ConStateDao/ConStateDaoImpl,RightDao/RightDaoImpl,RoleDao/RoleDaoImpl.

The interface is saved in com.ruanko.dao package, the corresponding implementation class saves in com.ruanko.dao.impl package.

5.5 Util Class Design

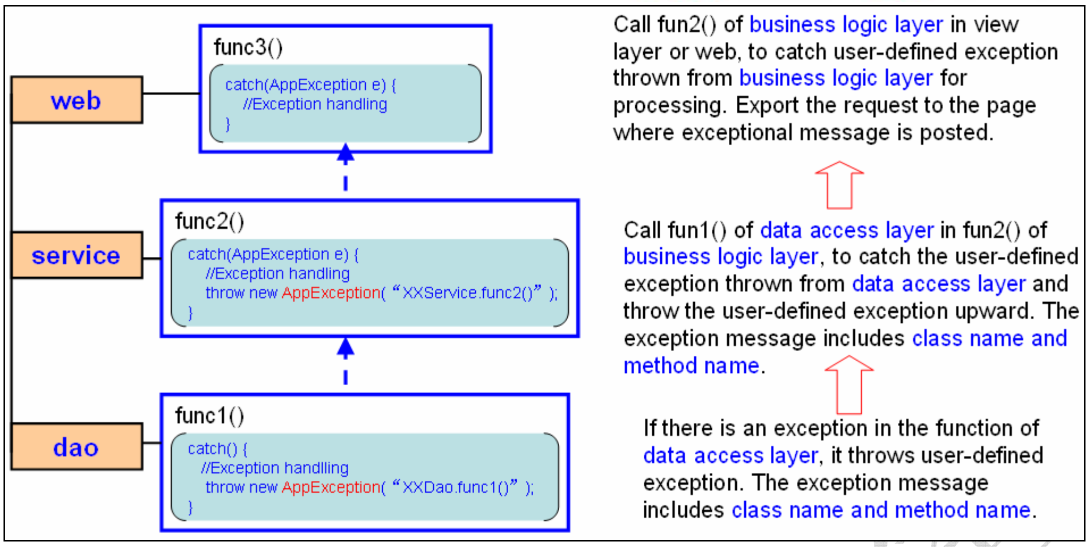
Extract the public function to form to util class, for the other classes to use, including database tool class DBUtil, the constant define class Constant, user-defined exception class AppException.

Util classes are saved in com.ruanko.utils.

6 Error Design

Uniform to handle exceptions of the view layer, business logic layer and data access layer, define AppException class as handle exceptions' tool class. Use try,catch,throw to handle the exceptions, when the exceptions occurs in the bottom layer, upthrow the user-defined exception, and including the exception information, the format of exception information as: the exception class, the function name of exception. Until the view layer handle uniformly.
In project, the unified exception class is AppException,which is saved in com.ruanko.utils package of the project.

软酷卓越实验室



```
func3()
catch(AppException e) {
    //Exception handling
}
```

```
func2()
catch(AppException e) {
    //Exception handling
    throw new AppException( “XXService.func2()” );
}
```

```
func1()
catch() {
    //Exception handlling
    throw new AppException( “XXDao.func1()” );
}
```

Call fun2() of business logic layer in view layer or web, to catch user-defined exception thrown from business logic layer for processing. Export the request to the page where exceptional message is posted.

Call fun1() of data access layer in fun2() of business logic layer, to catch the user-defined exception thrown from data access layer and throw the user-defined exception upward. The exception message includes class name and method name.

If there is an exception in the function of data access layer, it throws user-defined exception. The exception message includes class name and method name.

软酷卓越实验室