Bookstore

说明文档

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| 2015.6.28 | 1.0 | 最初版本 | 张子辰 |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 功能 4

1.3 构架目标和约束 4

2. 架构 4

2.1 架构风格 4

2.2 包结构一览 4

3. 第一次迭代 4

3.1 SimpleLoginModule 4

3.2 数据加密 5

4. 第二次迭代 5

4.1 Ajax和json 5

4.2 事务管理 5

4.3 Xss防御 5

4.4 MDB 5

4.5 聊天室 5

5. 第三次迭代 5

5.1 memcached 5

5.2 国际化 5

5.3 集群策略 5

5.4 Aop 5

5.5 Web service 5

6. 第四次迭代 6

6.1 Lucene 6

6.2 大数据 6

说明文档

# 简介

## 目的

本文档详细说明了Bookstore项目的演化过程、架构风格以及所使用的组件。

## 功能

该项目的主要功能有图书管理、购物车、订单处理、以及在线聊天。

## 构架目标和约束

开发语言：Java。

开发工具：Eclipse。

服务器：GlassFish 4

设计模式：面向对象的UML设计模式。

框架/组件：ejb

架构风格：MVC

数据库：Mysql。

团队结构：单人项目。

# 架构

## 架构风格

本项目从se223课程的大作业演化而来，自始至终都严格遵循MVC架构风格。项目大致分为四个部分：view层是一些jsp，model层是实体类和数据库中的数据。Controller层分为两部分，一部分是控制逻辑，主要是servlet类；另一部分是业务逻辑，原先是spring中的dao类，使用ejb之后改为了会话bean。

当浏览器的请求发到服务器软件时，服务器软件会根据配置好的路由（WebServlet）选择并加载相应的servlet。Servlet验证参数之后访问会话bean，会话bean再访问数据库。servlet拿到数据之后处理数据，并传给jsp，jsp渲染出页面之后返回给浏览器。另外有一些servlet用于ajax调用，它们不把参数传给jsp，而是直接组装成json返回。

Servlet、ejb和数据库三个部分可以部署在不同的主机上面。

## 包结构一览

下面列出了主要的包，以及它们的作用：

bookstore.servlet：存放servlet。

bookstore.sessionbean：存放会话bean。

bookstore.remote：存放会话bean依赖的remote接口。

bookstore.entitybean：实体类，数据库中数据的封装。用于c层与m层，c层与v层传递数据。

bookstore.utility：功能性的函数或者用于jdbc连接的配置。

bookstore.websocketbot：存放聊天室所用的websocket服务端。

bookstore.meesagebean：存放消息驱动bean。

bookstore.ws：存放webservice。

bookstore.jaas：存放简单登录认证模块。

# 第一次迭代

## SimpleLoginModule

项目中实现了一个符合jaas规范的登录认证模块，存放在bookstore.jaas包下面。然而在实际部署时，配置文件无法找到而加载失败，项目中模块以直接new出来的方式使用，并没有交给jaas组件加载。

## 数据加密

Servlet和ejb之间的订单信息采用加密方式传输。Ejb开放了一个返回订单加密信息的接口，这个接口返回一个用discuz的authcode算法加密的json字符串。

# 第二次迭代

## Ajax和json

BookServlet中提供了图书增、删、查的接口，OrderServlet中提供了订单查询的接口，他们的返回数据均为json格式。图书页面（/index?action=book）和(/index?action=order)在访问时先返回大致的html结构，之后再由ajax加载具体内容。图书页面的增加和删除操作也是以ajax方式交互的。

## 事务管理

项目的所有数据库连接都采用了命令式的事务管理。具体在代码中，数据库连接之后设置为不自动提交。等到一系列的数据库操作进行完毕之后，手动调用commit来提交。隔离级别方面，所有带写入的操作都是串行化的。只有查询的事务，由于它们都只访问一次数据库，均为已提交读。

## Xss防御

所有jsp中输出字符串的部分，都用方法HtmlEnco转义。这个方法是我自己写的，位于bookstore.utility.Common中。它把左尖括号、右尖括号、反斜杠、单引号和双引号五个字符转义为html实体。

## MDB

项目中下达订单的业务逻辑是用mdb而不是会话bean实现的。调用mdb的代码位于bookstore.servlet.CartServlet内，doOrder方法中。

## 聊天室

项目中实现了一个小型的聊天室，代码在bookstore.websocketbot中。链接在导航栏中有显示，通过/chatroom.jsp可以访问。为了顺利部署，服务器从jboss切换为glassfish。

# 第三次迭代

## memcached

存储购物车的地方从stateful bean 变成了 memcached。

## 国际化

图书页面（index?action=book）实现了国际化，最下方有个下拉框供选择语言，点击旁边的按钮可以刷新页面。项目中通过url中的查询参数设置语言，通过cookie读入和保存语言。

## 集群策略

在一个局域网内架设多台主机，每台主机上面都部署bookstore应用。采用被动连接模式，当一个请求过来时，负载均衡器一个一个去试所有主机，找到能够连接的主机就把请求转送给它。另外，可以配置一台监测用的主机，每隔一段时间向其余主机发送心跳包，以确认是否正常工作。

## Aop

独立项目AspectjExample中应用了aspectj。这个项目是一个n queens模拟生成工具，并且所有的方法的调用前后都会打出log，以便开发者查看调用栈的情况。

## Web service

bookstore.ws包下面，BookWsdl是wsdl形式的web service，BookRestful是restful形式的web service，提供对图书增加、删除、查询三个操作的接口。

# 第四次迭代

## Lucene

独立项目LuceneExample中演示了如何使用lucene制作索引和查询。Index子目录下面是制作好的索引，rsrc子目录下面是制作索引用的三个文本文件。

## 大数据

为了满足大数据的需求，静态文件比如图片、js、css交给cdn来托管，数据库里的数据转移至nosql，并分布存储与多个服务器中，同时做冗余。