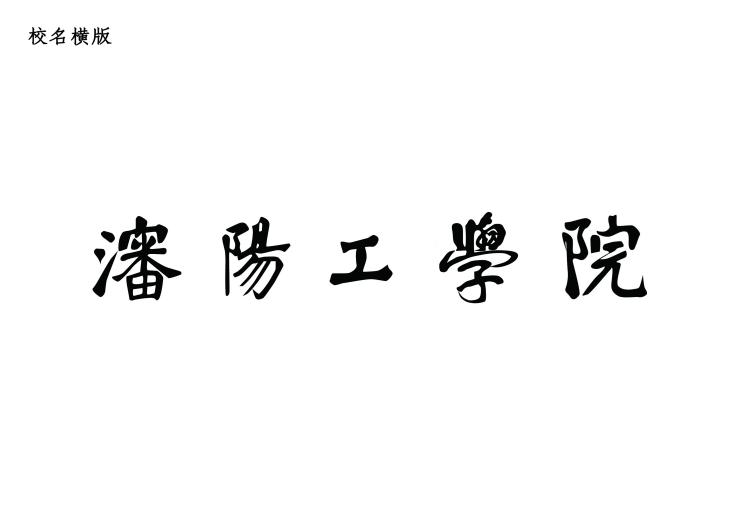
****

****

**企业实训报告**

**学 院： 信息与控制学院**

**专业班级: 计算机科学与技术一班**

**学 号：1530110129**

**学生姓名：王威人**

**成 绩：**

沈阳工学院

**企业实训日志**

**2017年12月18日 　　　星期一**

简单熟悉了企业团队开发的流程，了解了网站开发的整个过程。确定了这几天需要做的项目，对网上书店商城进行了基本的了解，并对网上书店商城项目进行功能结构的设计。了解了团队开发的重要性，并确定每组中每个人需要做的事情，对功能模块的确定。设计出系统需要的数据库。数据库设计如下:

用户表：

create table users

(

id int(10) NOT NULL AUTO\_INCREMENT,

username varchar(40) not null unique,

password varchar(40) not null,

phone varchar(40) not null,

email varchar(40) not null,

ulimit int not null,

primary key(id)

);

用户收货信息表：

Create table reward

(

id int(10) NOT NULL AUTO\_INCREMENT,

address varchar(100) not null,

user\_id varchar(40) not null,

primary key(id)

)

图书信息：

create table books

(

id int(10) NOT NULL AUTO\_INCREMENT,

name varchar(100) not null unique,

author varchar(40) not null,

price double not null,

image varchar(100),

description varchar(255),

category\_id int,

primary key(id)

);

订单表：

create table orders

(

id int(10) NOT NULL AUTO\_INCREMENT,

ordertime datetime not null,

price double,

state boolean,

user\_id int(10),

primary key(id)

);

订单项：

create table orderitem

(

id int(10) NOT NULL AUTO\_INCREMENT,

quantity int,

price double,

order\_id int(10),

book\_id int(10),

primary key(id)

);

**2017年12月19日 　　　星期二**

对本次实训项目有了整体的了解，安装mysql数据库。搭建项目，将文件结构拷贝至eclipse中，配置tomcat服务器，指定工作区间。了解Spring框架在开发中的重要性，并使用Spring框架整合、Struts2框架和Hibernate框架。对项目功能进行编写，我负责对购物车功能进行编写。主要代码如下:

Action中添加至购物车代码:

public String buy() throws Exception {

//验证用户是否登录

UsersVo user = (UsersVo) session.get("user");

if(user == null) {

request.put("message", "对不起，请先登录！！！");

return "nologin";

}

//查询出指定的图书信息

int id = Integer.parseInt(bookid);

List<BooksVo> list = bookDao.findById(id);

//遍历出图书信息

BooksVo book = null;

for(BooksVo b : list) {

book = new BooksVo();

book = b;

}

//获取用户购物车

Cart cart = (Cart) session.get("cart");

if(cart == null) {

cart = new Cart();

session.put("cart", cart);

}

cart.add(book);

return SUCCESS;

}

添加至购物车的方法，代码如下：

public void add(BooksVo book){

CartItem item = map.get(book.getId());

//如果原本购物车中没有该图书

if(item == null){

item = new CartItem();

item.setBook(book);

item.setQuantity(1);

map.put(book.getId(), item);

}else{

item.setQuantity(item.getQuantity() + 1);

}

}

删除购物车中的指定图书的方法，代码如下：

public void deleteBook(int id) {

CartItem item = map.get(id);

if(item.getQuantity() > 1) {

item.setQuantity(item.getQuantity() - 1);

}else {

map.remove(id);

}

}

**2017年12月20日 　　　星期三**

对昨天分发的任务在小组中进行检查，并将已经编写好的代码整合在一起进行测试，将所有的代码整合到一个项目中去，会遇到500这样的错误，通过调试、检查程序，发现有的包名并不正确，导入最新的的jar包即解决问题，在前端界面的设计上，以.html格式的文件并没有什么问题，但是整合到.jsp中时，有时会出现引用的css并没有起作用，通过一番更改最终解决。最后分发别的模块任务，涉及代码，订单模块功能实现，主要代码如下：

配置文件中主要代码：

<bean id="orderDao" class="com.dhee.dao.OrderDao">

<property name="hibernateTemplate" ref="hibernateTemplate" />

</bean>

<bean id="orderAction" class="com.dhee.action.OrderAction">

<property name="orderDao" ref="orderDao"></property>

</bean>

添加订单，查看订单的Dao层代码：

public void addOrder(OrdersVo order) throws Exception{

this.hibernateTemplate.save(order);

}

public List<OrdersVo> findOrder(int id) throws Exception{

String hql = "from OrdersVo where user\_id=?";

List<OrdersVo> list = this.hibernateTemplate.find(hql, new Object[] {id});

return list;

}

控制逻辑层中，action的主要代码：

public String confirm() throws Exception{

Cart cart = (Cart) session.get("cart");

int id = Integer.parseInt(userid);

OrdersVo order = new OrdersVo();

order.setPrice(cart.getPrice());

order.setState(false);

order.setUser\_id(id);

order.setOrdertime(new Date());

orderDao.addOrder(order);

//订单项没写

request.put("message", "订单已生成，请等待发货！！！");

session.remove("cart");

return SUCCESS;

}

public String findOrder() throws Exception{

int id = Integer.parseInt(userid);

List<OrdersVo> list = orderDao.findOrder(id);

request.put("orders", list);

return SUCCESS;

}

**2017年12月21日 　　　星期四**

整合所有功能并完善功能，对程序进行测试，解决一些非逻辑上的Bug。完善项目中的登录注册功能，浏览图书功能以及订单功能，并按照需求分析对某一功能进行进一步的修改和完善，并在组员的讨论下增加用户添加地址这一功能，使功能更加完善，项目更加完善并且实用。并添加全局错误显示界面，即当程序在运行时遇到逻辑上的错误也不会在页面中显示500这种用户看不懂的错误信息，而是以非常友好的形式提示用户当前有误，请尽快联系管理员。主要代码如下：

用户添加地址、查询地址，主要代码如下：

public String add() throws Exception{

reward.setAddress(new String(reward.getAddress().getBytes("iso-8859-1"),"utf-8"));

rewardDao.addAddress(reward);

request.put("message", "地址添加成功！！！");

return SUCCESS;

}

public String find() throws Exception {

int id = Integer.parseInt(userid);

List<RewardVo> list = rewardDao.findAddress(id);

session.put("reward", list);

return SUCCESS;

}

持久层中，将用户的地址保存至数据库中，对于已经添加过地址的用户，将从数据库中查询出来传至action中，主要代码如下：

public List<RewardVo> findAddress(int userid) throws Exception{

String hql = "from RewardVo where user\_id=?";

List<RewardVo> list = this.hibernateTemplate.find(hql, new Object[] {userid});

return list;

}

public void addAddress(RewardVo reward) throws Exception{

this.hibernateTemplate.save(reward);

}

在配置文件中，配置全局的错误显示界面，以给用户一个友好提示，主要代码如下：

<package name="struts-error" extends="struts-default">

<global-results>

<result name="error">/error.jsp</result>

</global-results>

<global-exception-mappings>

<exception-mapping result="error" exception="java.lang.Exception"/>

</global-exception-mappings>

</package>

**2017年12月22日 　　　星期五**

对项目进行最后一次综合测试，将所有功能之间进行整合，对于最后一天的小组审核进行分工，完成PPT，并对整个项目的代码进行统计，按要求，注释行要达到代码总行的14%至15%，并且小组内的每一个成员对其完成的功能进行总结，对于解决不了的问题提出来，大家一起解决，最终上台进行小组评审，并演示PPT及程序，对于老师和同学提出的问题进行解答。最后老师讲解了计算机行业的发展前景，并对程序测试进行了单独的讲解，了解到国内与国外计算机行业的不同，国外非常重视测试这一环节，而在国内，测试这一环节相较国外来说略有不足，老师详细的说明了，在整个项目中测试的重要性。今天涉及的代码如下：

对用户的收货地址的bug进行了修改:

<div class="shdz">

<s:if test="#session.reward.size==0">

<form action="addAddress" method="post">

<table>

<td>收货地址：</td>

<td>

<input placeholder="输入地址" name="reward.address" class="address" type="text" required="">

<input type="hidden" name="reward.user\_id" value="${user.id }">

</td>

<td><input type="submit" value="提交"/></td>

</table>

</form>

</s:if>

<s:else>

<table>

<td>收货地址：</td>

<s:iterator var="re" value="#session.reward">

<td><s:property value="#re.address"/></td>

</s:iterator>

</table>

<div class="tijiao">

<a href="${pageContext.request.contextPath }/confirmOrder?userid=${user.id}" class="tijiao">确认订单</a>

</div>

</s:else>

</div>

增加了页面自动跳转的功能:

<script type="text/javascript">

var i = 3;

var intervalid;

intervalid = setInterval("fun()",1000);

function fun() {

if(i == 0){

window.location.href = "${pageContext.request.contextPath }/login.jsp";

clearInterval(intervalid);

}

document.getElementById("mes").innerHTML = i;

i--

}

</script>

**实训心得**

为期一周的企业实训中，我们了解了企业开发中的相关流程，对于一个项目的整体开发流程有了基本的了解，同时了解了华信公司的企业文化，并且懂得了编码规范，编写好的代码不仅对自己受益，对他人也有不小的帮助。

了解了瀑布式的开发模式，由上至下，最大的特点是不可逆，知道了软件开发的声明周期：

1. 需求分析：用户想要做什么？
2. 概要设计：设计者能提供什么？
3. 详细设计：设计者让程序做什么？
4. 编码：程序员让计算机做什么？
5. 测试：测试人员运行结果是否正确？

以上这五点在平时的学习和编写代码中往往是我们不会考虑的，我们总是上来就是编写代码，并没有对编码前的三步进行详细的设计。本次企业实训，我们就严格的按照企业的软件开发模式，从老师提出的要求，我们做了系统的规划，从需求分析到程序测试，严格的按照流程进行。懂得了团队协作开发的重要性。

本次企业实训，使我对Struts2,Spring,Hibernate三大框架有了更好的理解。Struts2用于控制层，Spring用于业务逻辑层，Hibernate用于持久层。这三个框架中，Spring框架显得尤为重要，它可以降低程序的耦合性。Spring框架的特点：

1. 方便解耦，简化开发。Spring就是一个大工厂，可以将所有对象创建和依赖关系维护，交给Spring管理。
2. AOP编程的支持。Spring提供面向切面编程，可以方便的实现对程序进行权限拦截、运行监控等功能。
3. 声明事务的支持。只需要通过配置就可以完成对事务的管理，而无需手动编程。
4. 方便程序的测试。Spring对Junit4支持，可以通过注解方便的测试Spring程序。
5. 方便集成各种优秀框架。pring不排斥各种优秀的开源框架，其内部提供了对各种优秀框架（如：Struts2、Hibernate、MyBatis、Quartz等）的直接支持。
6. 降低了JaveEE API的使用难度。Spring 对JavaEE开发中非常难用的一些API（JDBC、JavaMail、远程调用等），都提供了封装，使这些API应用难度大大降低。

本次企业实训也提高了自身的编码能力，独立解决问题的能力，以及团队协作开发的能力。同时了解到当前计算机行业的就业前景，以及发展方向，对当前的一些公司使用的技术也有所了解。同样也明确了自己是否符合这个行业，是否真正的喜欢计算机行业。掌握了自己以前不了解的技术以及编码规范，对于自己的技术欠缺也有所了解，明确了自己需要努力的方向，如何提高自己，投资自己，为以后步入社会，步入计算机行业打下良好的基础。