# E-COMMERCE WEB APPLICATION

**Project Abstract:**

Building a one stop e-commerce web application, wherein sellers can sign up and post their products. Customers can also sign-up as well and order the products of their choice and on successful order placement of a product, confirmation is sent on the email to the user.

**Roles included:**

* Admin – Manage sellers on the website and product inventory.
* Seller – Manage their products on the website to be sold and their quantity based on popularity of product.
* Customer – Buy products of their choice from the web app. Also keep items in cart till their approval.

**Planned Scope:**

Admin can validate the seller’s authenticity and manage inventory and well as seller’s based on their reviews and seller’s popularity based on its products on the web application. Sellers can post different products and manage their inventory based on product’s popularity. Customers on the other side can buy products, comment about them and keep orders in their cart till they have made up their mind. A confirmation email will be sent to the user on successfully ordering an item from the web application.

PROJECT TILL NOW:



CONTROLLER DATA:

1). ADMIN CONTROLLER

package com.me.controller;

import java.util.ArrayList;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.beans.factory.annotation.Qualifier;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.WebDataBinder;

import org.springframework.web.bind.annotation.InitBinder;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.servlet.ModelAndView;

import com.me.dao.AdminDAO;

import com.me.dao.SellerDAO;

import com.me.pojo.Seller;

import com.me.validator.AdminValidator;

import com.me.validator.SellerValidator;

@Controller

public class AdminController {

@Autowired

@Qualifier("adminDao")

AdminDAO adminDao;

@Autowired

@Qualifier("adminValidator")

AdminValidator adminvalidator;

@InitBinder

private void initBinder(WebDataBinder binder) {

binder.setValidator(adminvalidator);

}

@RequestMapping(value = "/validatesellers.htm", method = RequestMethod.GET)

protected ModelAndView ValidateSellers() throws Exception {

System.out.println("Validate Sellers by Admin clicked!");

ArrayList<Seller> sellerList = adminDao.getNewSellerList();

return new ModelAndView("validate-sellers","sellerlist",sellerList);

}

@RequestMapping(value = "/admin-home.htm", method = RequestMethod.GET)

protected ModelAndView AdminHomePage() throws Exception {

return new ModelAndView("admin-home");

}

}

2). AJAXDATAPROVIDER

package com.me.controller;

import java.io.IOException;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.Map;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import org.hibernate.HibernateException;

import org.hibernate.Query;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.bind.annotation.ResponseBody;

import com.me.dao.DAO;

import com.me.exception.AdminException;

import com.me.pojo.Seller;

/\*\*

\* Servlet implementation class AjaxDataProvider

\*/

@Controller

public class AjaxDataProvider extends DAO {

public AjaxDataProvider() {

super();

// TODO Auto-generated constructor stub

}

@RequestMapping(value="/sellerlist.htm", method=RequestMethod.POST)

public @ResponseBody Seller getAttr(@PathVariable(value="sellername") final String id,

@RequestParam String someAttr) {

ArrayList<Seller> sellerList;

Seller seller1 = new Seller();

System.out.println("Inside AJAXDATAPROVIDER CONTROLLER!!");

try{

String sql = "from Seller where name = someAttr";

Query query = getSession().createQuery(sql);

System.out.println("query for selecting seller details from db ran in ajax controller!!");

sellerList = (ArrayList<Seller>)query.list();

for(Seller seller:sellerList){

seller1 = seller;

}

}catch (HibernateException e) {

rollback();

System.out.println("Error retreiving seller list - AJAX CONTROLLER!!");

}

System.out.println("Seller name:\t"+seller1.getName());

return seller1;

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

// protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

// // TODO Auto-generated method stub

// boolean isValid;

// String seller = request.getParameter("sellername");

// System.out.println("Seller data:\t"+seller);

// if(seller!=null){

// isValid = true;

// }

// }

}

3). CATEGORY CONTROLLER

package com.me.controller;

import javax.servlet.http.HttpServletRequest;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.beans.factory.annotation.Qualifier;

import org.springframework.stereotype.Controller;

import org.springframework.validation.BindingResult;

import org.springframework.web.bind.WebDataBinder;

import org.springframework.web.bind.annotation.InitBinder;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.servlet.ModelAndView;

import com.me.dao.CategoryDAO;

import com.me.pojo.Category;

import com.me.validator.CategoryValidator;

import com.me.exception.CategoryException;

@Controller

public class CategoryController {

@Autowired

@Qualifier("categoryValidator")

CategoryValidator categoryValidator;

@Autowired

@Qualifier("categoryDao")

CategoryDAO categoryDAO;

@InitBinder

private void initBinder(WebDataBinder binder) {

binder.setValidator(categoryValidator);

}

@RequestMapping(value = "/AddCategories.htm", method = RequestMethod.GET)

protected ModelAndView AdminAddCategories() throws Exception {

System.out.println("Admin home Add Categories clicked!");

return new ModelAndView("category-add-form","category",new Category());

}

@RequestMapping(value = "/add.htm", method = RequestMethod.POST)

public ModelAndView addCategory(@ModelAttribute("category") Category category, BindingResult result,HttpServletRequest Request) throws Exception {

System.out.println("Inside Add Category Controller!!!");

categoryValidator.validate(category, result);

if (result.hasErrors()) {

return new ModelAndView("category-add-form", "category", category);

}

try {

category = categoryDAO.create(category.getTitle());

} catch (CategoryException e) {

System.out.println(e.getMessage());

return new ModelAndView("error", "errorMessage", "error while login");

}

Request.getSession().setAttribute("success", "Successfully added Category!!");

return new ModelAndView("category-add-form", "category", new Category());

}

}

4). LINK CONTROLLER

package com.me.controller;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

import org.apache.commons.mail.DefaultAuthenticator;

import org.apache.commons.mail.Email;

import org.apache.commons.mail.SimpleEmail;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.beans.factory.annotation.Qualifier;

import org.springframework.stereotype.Controller;

import org.springframework.validation.BindingResult;

import org.springframework.web.bind.WebDataBinder;

import org.springframework.web.bind.annotation.InitBinder;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.servlet.ModelAndView;

import com.me.exception.SellerException;

import com.me.exception.UserException;

import com.me.pojo.Seller;

import com.me.pojo.User;

import com.me.validator.SellerValidator;

import com.me.validator.UserValidator;

import com.me.dao.SellerDAO;

import com.me.dao.UserDAO;

@Controller

public class LinkController {

@Autowired

@Qualifier("userDao")

UserDAO userDao;

@Autowired

@Qualifier("userValidator")

UserValidator validator;

@InitBinder

private void initBinder(WebDataBinder binder) {

binder.setValidator(validator);

}

@RequestMapping(value = "/home.htm", method = RequestMethod.GET)

protected ModelAndView backtohome() throws Exception {

System.out.println("homepage clicked!");

return new ModelAndView("home");

}

@RequestMapping(value = "/userlogin.htm", method = RequestMethod.GET)

protected ModelAndView LoginPageTranfer() throws Exception {

System.out.println("Login/SignUp clicked!");

return new ModelAndView("login");

}

@RequestMapping(value = "/userregister.htm", method = RequestMethod.GET)

protected ModelAndView RegisterPageTransferUser() throws Exception {

System.out.println("Register clicked!");

return new ModelAndView("register","user",new User());

}

@RequestMapping(value = "/userlogin.htm", method = RequestMethod.POST)

protected ModelAndView loginUser(HttpServletRequest request) throws Exception {

HttpSession session = (HttpSession) request.getSession();

try {

System.out.println("loginUser");

User u = userDao.get(request.getParameter("username"), request.getParameter("password"));

if(u == null){

System.out.println("UserName/Password does not exist");

session.setAttribute("errorMessage", "UserName/Password does not exist");

return new ModelAndView ("error.jsp");

}

session.setAttribute("user", u);

//Create userhome.jsp in views

return new ModelAndView("user-home");

} catch (UserException e) {

System.out.println("Exception: " + e.getMessage());

session.setAttribute("errorMessage", "error while login");

return new ModelAndView ("error");

}

}

@RequestMapping(value = "/register.htm", method = RequestMethod.POST)

protected ModelAndView registerNewUser(HttpServletRequest request, @ModelAttribute("user") User user, BindingResult result) throws Exception {

System.out.println("Inside register controller!!!");

validator.validate(user, result);

System.out.println("Checking for user existance");

boolean flag = userDao.checkIfExists(user);

System.out.println("Checked for user existance!!");

if(!flag){

System.out.println("Same user found!!");

return new ModelAndView("register", "user", user);

}

if (result.hasErrors()) {

return new ModelAndView("register", "user", user);

}

System.out.print("registerNewUser");

String receiverEmail = user.getEmail().toString();

User u = null;

try {

u = userDao.register(user);

request.getSession().setAttribute("user", u);

Email email = new SimpleEmail();

email.setSmtpPort(465);

email.setAuthenticator(new DefaultAuthenticator("contactapplication2017@gmail.com", "springmvc"));

email.setHostName("smtp.gmail.com");//if a server is capable of sending email.

email.setSSL(true);//setSSLOnConnect(true);

email.setFrom("chintankoticha@gmail.com");

email.setSubject("SUCCESSFUL SIGN UP!!!");

email.setMsg("This is system generated mail, do not reply to this email.\n\n"+"\nYOUR USERNAME IS:\t"+user.getUsername()+"\nKINDLY LOGIN TO THE WEBSITE!!");

email.addTo(receiverEmail);

email.setTLS(true);//startTLS.enable.true

email.send();

} catch (UserException e) {

System.out.println("Exception: " + e.getMessage());

return new ModelAndView("error", "errorMessage", "error while login");

} catch (Exception e) {

System.out.println("Exception e:" + e.getMessage());

}

System.out.println("EMAIL SENT TO NEW USER!!");

return new ModelAndView("user-home", "user", u);

}

}

5). LOGIN CONTROLLER

package com.me.controller;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.beans.factory.annotation.Qualifier;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.WebDataBinder;

import org.springframework.web.bind.annotation.InitBinder;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.servlet.ModelAndView;

import com.me.dao.UserDAO;

import com.me.exception.UserException;

import com.me.pojo.User;

import com.me.validator.UserValidator;

@Controller

public class LoginController {

@Autowired

@Qualifier("userDao")

UserDAO userDao;

@Autowired

@Qualifier("userValidator")

UserValidator validator;

@InitBinder

private void initBinder(WebDataBinder binder) {

binder.setValidator(validator);

}

@RequestMapping(value = "/user/login", method = RequestMethod.POST)

protected ModelAndView loginUser(HttpServletRequest request) throws Exception {

HttpSession session = (HttpSession) request.getSession();

try {

System.out.print("loginUser");

User u = userDao.get(request.getParameter("username"), request.getParameter("password"));

if(u == null){

System.out.println("UserName/Password does not exist");

session.setAttribute("errorMessage", "UserName/Password does not exist");

return new ModelAndView("error");

}

session.setAttribute("user", u);

return new ModelAndView("user-home");

} catch (UserException e) {

System.out.println("Exception: " + e.getMessage());

session.setAttribute("errorMessage", "error while login");

return new ModelAndView("error");

}

}

}

6). SELLER LINK CONTROLLER

package com.me.controller;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

import org.apache.commons.mail.DefaultAuthenticator;

import org.apache.commons.mail.Email;

import org.apache.commons.mail.SimpleEmail;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.beans.factory.annotation.Qualifier;

import org.springframework.stereotype.Controller;

import org.springframework.validation.BindingResult;

import org.springframework.web.bind.WebDataBinder;

import org.springframework.web.bind.annotation.InitBinder;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.servlet.ModelAndView;

import com.me.dao.SellerDAO;

import com.me.exception.SellerException;

import com.me.exception.UserException;

import com.me.pojo.Seller;

import com.me.validator.SellerValidator;

@Controller

public class SellerLinkController {

@Autowired

@Qualifier("sellerDao")

SellerDAO sellerDao;

@Autowired

@Qualifier("sellerValidator")

SellerValidator sellervalidator;

@InitBinder

private void initBinder(WebDataBinder binder) {

binder.setValidator(sellervalidator);

}

@RequestMapping(value = "/sellerregister.htm", method = RequestMethod.GET)

protected ModelAndView RegisterPageTransferSeller() throws Exception {

System.out.println("Register clicked!");

return new ModelAndView("seller-register","seller",new Seller());

}

@RequestMapping(value = "/registerseller.htm", method = RequestMethod.POST)

protected ModelAndView registerNewSeller(HttpServletRequest request, @ModelAttribute("seller") Seller seller, BindingResult result) throws Exception {

System.out.println("Inside register controller for Seller!!!");

sellervalidator.validate(seller, result);

// System.out.println("Checking for user existance");

// boolean flag = userDao.checkIfExists(user);

// System.out.println("Checked for user existance!!");

// if(!flag){

// System.out.println("Same user found!!");

// return new ModelAndView("register", "user", user);

// }

if (result.hasErrors()) {

return new ModelAndView("seller-register", "seller", seller);

}

System.out.println("registerNewSeller");

String receiverEmail = seller.getEmailAddress();

Seller s = null;

try {

s = sellerDao.register(seller);

request.getSession().setAttribute("seller", s);

Email email = new SimpleEmail();

email.setSmtpPort(465);

email.setAuthenticator(new DefaultAuthenticator("contactapplication2017@gmail.com", "springmvc"));

email.setHostName("smtp.gmail.com");//if a server is capable of sending email.

email.setSSL(true);//setSSLOnConnect(true);

email.setFrom("chintankoticha@gmail.com");

email.setSubject("SUCCESSFUL SIGN UP!!!");

email.setMsg("This is system generated mail, do not reply to this email.\n\n"+"\nYOUR USERNAME IS:\t"+seller.getUsername()+"\n"

+ "KINDLY LOGIN TO THE WEBSITE ONCE YOU APPROVED BY THE ADMIN!!");

email.addTo(receiverEmail);

email.setTLS(true);//startTLS.enable.true

email.send();

} catch (SellerException e) {

System.out.println("Exception: " + e.getMessage());

return new ModelAndView("error", "errorMessage", "error while login");

} catch (Exception e) {

System.out.println("Exception e:" + e.getMessage());

}

System.out.println("EMAIL SENT TO NEW SELLER!!");

return new ModelAndView("seller-home", "seller", s);

}

@RequestMapping(value = "/sellerlogin.htm", method = RequestMethod.POST)

protected ModelAndView loginSeller(HttpServletRequest request) throws Exception {

HttpSession session = (HttpSession) request.getSession();

try {

System.out.println("loginSeller!!");

Seller u = sellerDao.get(request.getParameter("username"), request.getParameter("password"));

if(u == null){

System.out.println("UserName/Password does not exist");

session.setAttribute("errorMessage", "UserName/Password does not exist");

return new ModelAndView ("error");

}

session.setAttribute("seller", u);

//Create userhome.jsp in views

return new ModelAndView("seller-home");

} catch (SellerException e) {

System.out.println("Exception: " + e.getMessage());

session.setAttribute("errorMessage", "error while login");

return new ModelAndView ("error");

}

}

}