Project Report

Phonebook

Sunit Bail

[bail.s@northeastern.edu](mailto:bail.s@northeastern.edu)

001577432

Submit date: 12/18/21

Due Date: 12/18/21

Table of Contents

[1.0 Summary 1](#_Toc90713095)

[1.1 Data Discussion 2](#_Toc90713096)

[1.2 Users 2](#_Toc90713097)

[1.3 Technologies Used 2](#_Toc90713098)

[1.4 Data Model 3](#_Toc90713099)

[1.5 Implemented Functionality with screenshots 3](#_Toc90713100)

[1.6 Controller Code 10](#_Toc90713101)

# 1.0 Summary

Phonebook’s primary purpose is to allow a client to maintain a backup of their contact list. They would be able to maintain a list of all their contacts on an online portal. For this they can use a form or upload a CSV file with their contacts that would be persisted to a MySQL database. Access to the contact details would be allowed by appropriate logins. A user can also consent to having their profile added to a public registry that is accessible even without a login. User would also have the option of downloading their contacts in a CSV format to maintain a permanent copy. Addition and deletion options would be available with the end user for their specific set of contacts.

# 1.1 Data Discussion

The data would have attributes like name, phone number(s) and address of a client’s contacts. The client would be the sole consumer of their own data. The client also has the option of adding themselves to a public registry, which can be consumed by other users. An admin would have the option of deleting a user and their contacts.

# 1.2 Users

*End User Type 1*

Create an account and manipulate their own contact list on the portal

Can add to a public registry if they so desire.

*End User Type 2*

Can access a public registry of users.

*Admin*

Can deregister users from the public registry.

# 1.3 Technologies Used

* Spring Web MVC
* Hibernate
* JSP Files

# 1.4 Data Model

Graphical user interface

Description automatically generated

# 1.5 Implemented Functionality with screenshots

* Registration for new users

Graphical user interface

Description automatically generated

* Login for existing users

Graphical user interface, application

Description automatically generated

* Login for administrator

Graphical user interface, application

Description automatically generated

* + Dashboard for administrator

Graphical user interface, text

Description automatically generated with medium confidence

* + Current enrolled users list available to Administrator only

Graphical user interface, application

Description automatically generated

* + Delete enrolled users with their contact lists available to Administrator only

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

* + Logout success screen

Graphical user interface, text, application, email

Description automatically generated

* Public Registry of enrolled users available to all users without creating account

Graphical user interface, application, email

Description automatically generated

* Post Login for existing users
  + Dashboard for users

A picture containing application

Description automatically generated

* + User Profile Update capability

Graphical user interface, application

Description automatically generatedGraphical user interface, application, email

Description automatically generated

* + Contact List with download capability

Table

Description automatically generated

* + Add contact individually or with file upload

Graphical user interface, application

Description automatically generated

* + Update Contact

Table

Description automatically generatedGraphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

* + Search Contact

Graphical user interface, application

Description automatically generated

* + Delete Contact

Table

Description automatically generated

Graphical user interface, text, application

Description automatically generated

# 1.6 Controller Code

package com.neu.phonebook.controller;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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

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

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.Contact;

import com.neu.phonebook.model.User;

@Controller

public class AddContactController {

@Autowired

ContactDao contactDao;

@Autowired

UserDAO userDAO;

@GetMapping(value = "/addContact.htm")

public ModelAndView addContact(ModelMap model, Contact contact, HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("username") == null) return new ModelAndView("redirect:/");

model.addAttribute("contact", contact);// Command Class

return new ModelAndView("addContact");

}

@PostMapping(value = "/addContact.htm")

public ModelAndView addContact(@ModelAttribute("contact") Contact contact, HttpServletRequest request) {

String username = (String) request.getSession().getAttribute("username");

System.out.println("Username" + username);

User user = userDAO.get(username);

contact.setUser(user);

Contact committed = contactDao.create(contact);

if (committed != null)

return new ModelAndView("addContactSuccess");

else {

return new ModelAndView("addContactError");

}

}

}

package com.neu.phonebook.controller;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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

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

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

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

import com.neu.phonebook.model.User;

@Controller

@SessionAttributes("admin")

public class AdminController {

@GetMapping(value = "/admin")

public String showForm(ModelMap model, User user, HttpServletRequest request) {

request.getSession().invalidate();

user.setUsername("admin");

user.setPassword("admin");

model.addAttribute("user", user);

return "admin";

}

@GetMapping(value = "/adminDashboard")

public String showDashboard(HttpServletRequest request) {

if (request.getSession(false) != null)

return "dashboard";

else {

return "admin";

}

}

@PostMapping(value = "/admin")

public String showSuccess(@ModelAttribute("user") User user, HttpServletRequest request) {

if (user.getUsername().equals("bailsunit.admin") && user.getPassword().equals("password")) {

HttpSession hs = request.getSession();

hs.setAttribute("name", "Sunit");

return "dashboard";

}

else {

return "invalid";

}

}

}

package com.neu.phonebook.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.Contact;

import com.neu.phonebook.model.User;

@Controller

public class ContactListController {

@Autowired

UserDAO userDAO;

@Autowired

ContactDao contactDao;

@GetMapping(value = "/contactList.htm")

public ModelAndView contactList(HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("username") == null) return new ModelAndView("redirect:/");

String username = (String) request.getSession().getAttribute("username");

User user = userDAO.get(username);

List<Contact> contacts = contactDao.getContactList(user);

request.setAttribute("contactList", contacts);

return new ModelAndView("contactList");

}

}

package com.neu.phonebook.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

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

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.Contact;

import com.neu.phonebook.model.User;

@Controller

public class DeleteContactController {

@Autowired

ContactDao contactDao;

@Autowired

UserDAO userDAO;

@GetMapping(value = "/delete.htm")

public ModelAndView deleteContact(HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("username") == null) return new ModelAndView("redirect:/");

String username = (String) request.getSession().getAttribute("username");

User user = userDAO.get(username);

List<Contact> contacts = contactDao.getContactList(user);

request.setAttribute("contactList", contacts);

return new ModelAndView("delete");

}

@PostMapping(value = "/delete.htm")

public ModelAndView deleteContactSpecific(HttpServletRequest request) {

String contactId = request.getParameter("contact");

Contact contact = contactDao.get(Integer.parseInt(contactId));

contactDao.delete(contact);

return new ModelAndView("deleteSuccess");

}

}

package com.neu.phonebook.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

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

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.Contact;

import com.neu.phonebook.model.User;

@Controller

public class DeleteUserController {

@Autowired

ContactDao contactDao;

@Autowired

UserDAO userDAO;

@GetMapping(value = "deleteUser.htm")

public ModelAndView deleteContact(HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("name") == null) return new ModelAndView("redirect:/admin");

List<User> users = userDAO.get();

request.setAttribute("userList", users);

return new ModelAndView("deleteUser");

}

@PostMapping(value = "deleteUser.htm")

public ModelAndView deleteContactSpecific(HttpServletRequest request) {

String userId = request.getParameter("user");

User user = userDAO.get(Integer.parseInt(userId));

List<Contact> contacts = contactDao.getContactList(user);

for(Contact contact : contacts) {

contactDao.delete(contact);

}

userDAO.delete(user);

return new ModelAndView("deleteUserSuccess");

}

}

package com.neu.phonebook.controller;

import java.io.File;

import java.io.FileWriter;

import java.io.IOException;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

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

import org.springframework.stereotype.Controller;

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.Contact;

import com.neu.phonebook.model.User;

import com.opencsv.CSVWriter;

@Controller

public class DownloadContactListController {

@Autowired

UserDAO userDAO;

@Autowired

ContactDao contactDao;

@GetMapping(value = "/download")

public ModelAndView downloadContactList(HttpServletRequest request) {

String username = (String) request.getSession().getAttribute("username");

User user = userDAO.get(username);

List<Contact> contacts = contactDao.getContactList(user);

writeContactsFile(contacts);

return new ModelAndView("downloadSuccess");

}

private void writeContactsFile(List<Contact> contacts) {

File file = new File("/Users/sunitbail/Downloads/contacts/contacts.csv");

try {

FileWriter outputfile = new FileWriter(file);

CSVWriter writer = new CSVWriter(outputfile);

String[] header = { "Name", "Phone", "Address" };

writer.writeNext(header);

for (Contact contact : contacts) {

header[0] = contact.getName();

header[1] = contact.getPhone();

header[2] = contact.getAddress();

writer.writeNext(header);

}

writer.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

package com.neu.phonebook.controller;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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

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

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

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

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.User;

@Controller

@SessionAttributes("login")

public class HomeController {

@Autowired

UserDAO userDAO;

@GetMapping(value = "/")

public String showForm(ModelMap model, User user, HttpServletRequest request) {

request.getSession().invalidate();

user.setUsername("Username");

user.setPassword("password");

model.addAttribute("user", user);// Command Class

return "home";

}

@GetMapping(value = "/dashboard")

public String showDashboard(HttpServletRequest request) {

if (request.getSession(false) != null)

return "login";

else {

return "home";

}

}

@PostMapping(value = "/")

public String showSuccess(@ModelAttribute("user") User user, HttpServletRequest request) {

User committed = userDAO.get(user.getUsername());

if (committed != null && committed.getPassword().equals(user.getPassword())) {

HttpSession hs = request.getSession();

String[] fullName = committed.getName().split(" ");

hs.setAttribute("name", fullName[0]);

hs.setAttribute("username", committed.getUsername());

return "login";

}

else {

return "invalid";

}

}

}

package com.neu.phonebook.controller;

import javax.servlet.http.HttpServletRequest;

import org.springframework.stereotype.Controller;

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

import org.springframework.web.servlet.ModelAndView;

@Controller

public class LogoutController {

@GetMapping(value = "/logout.htm")

public ModelAndView userList(HttpServletRequest request) {

request.getSession().invalidate();

return new ModelAndView("logout");

}

}

package com.neu.phonebook.controller;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.User;

@Controller

public class ProfileController {

@Autowired

ContactDao contactDao;

@Autowired

UserDAO userDAO;

@GetMapping(value = "/profile")

public ModelAndView updateProfile(ModelMap model, User user, HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("username") == null) return new ModelAndView("redirect:/");

String username = (String) request.getSession().getAttribute("username");

user = userDAO.get(username);

model.addAttribute("user", user);

String consent = user.getIsPublic().equals("true") ? "Yes" : "No";

System.out.println(consent);

request.setAttribute("consent", consent);

return new ModelAndView("updateProfile");

}

@PostMapping(value = "/profile")

public ModelAndView updateProfileFinal(ModelMap model, User user, HttpServletRequest request) {

String selected = request.getParameter("consent");

User refresh = userDAO.get((String) request.getSession().getAttribute("username"));

refresh.setName(user.getName());

refresh.setEmail(user.getEmail());

refresh.setName(user.getName());

refresh.setPhone(user.getPhone());

refresh.setAddress(user.getAddress());

refresh.setPassword(user.getPassword());

if (selected.equals("truth"))

refresh.setIsPublic("true");

else

refresh.setIsPublic("false");

userDAO.update(refresh);

request.setAttribute("user", refresh);

String consent = refresh.getIsPublic().equals("true") ? "Yes" : "No";

request.setAttribute("consent", consent);

return new ModelAndView("updateProfileSuccess");

}

}

package com.neu.phonebook.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

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

import org.springframework.stereotype.Controller;

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.User;

@Controller

public class PublicListController {

@Autowired

UserDAO userDAO;

@GetMapping(value = "/yellowPages.htm")

public ModelAndView yelloPages(HttpServletRequest request) {

List<User> userList = userDAO.getUserList();

request.setAttribute("userList", userList);

return new ModelAndView("yellowPages");

}

}

package com.neu.phonebook.controller;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

import org.hibernate.exception.ConstraintViolationException;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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

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

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.User;

@Controller

public class RegisterController {

@Autowired

UserDAO userDAO;

@GetMapping(value = "/register.htm")

public ModelAndView addUser(ModelMap model, User user, HttpServletRequest request) {

model.addAttribute("user", user);// Command Class

return new ModelAndView("register");

}

@PostMapping(value = "/register.htm")

public ModelAndView addUser(@ModelAttribute("user") User user, HttpServletRequest request) {

String selected = request.getParameter("consent");

try {

if (selected.equals("truth"))

user.setIsPublic("true");

else

user.setIsPublic("false");

User committed = userDAO.create(user);

if (committed != null && committed.getPassword().equals(user.getPassword())) {

HttpSession hs = request.getSession();

String[] fullName = committed.getName().split(" ");

hs.setAttribute("name", fullName[0]);

hs.setAttribute("username", committed.getUsername());

return new ModelAndView("login");

} else {

return new ModelAndView("registrationError");

}

} catch (ConstraintViolationException e) {

return new ModelAndView("usernameError");

}

}

}

package com.neu.phonebook.controller;

import java.util.ArrayList;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

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

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.Contact;

import com.neu.phonebook.model.User;

@Controller

public class SearchContactController {

@Autowired

ContactDao contactDao;

@Autowired

UserDAO userDAO;

@GetMapping(value = "/searchContact.htm")

public ModelAndView searchContact(HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("username") == null) return new ModelAndView("redirect:/");

return new ModelAndView("searchContact");

}

@PostMapping(value = "/searchContact.htm")

public ModelAndView searchContactResult(HttpServletRequest request) {

String contactname = (String) request.getParameter("keyword");

String parameter = request.getParameter("parameter");

User user = userDAO.get((String)request.getSession().getAttribute("username"));

List<Contact> contactList = (ArrayList<Contact>)contactDao.getLike(contactname, user,parameter);

if(contactList.isEmpty()) {

request.setAttribute("contactname", contactname);

return new ModelAndView("searchResultsEmpty");

}

else {

request.setAttribute("contactList", contactList);

return new ModelAndView("searchResults");

}

}

}

package com.neu.phonebook.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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

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

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.Contact;

import com.neu.phonebook.model.User;

@Controller

public class UpdateContactController {

@Autowired

ContactDao contactDao;

@Autowired

UserDAO userDAO;

@GetMapping(value = "/update.htm")

public ModelAndView updateContact(HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("username") == null) return new ModelAndView("redirect:/");

String username = (String) request.getSession().getAttribute("username");

User user = userDAO.get(username);

List<Contact> contacts = contactDao.getContactList(user);

request.setAttribute("contactList", contacts);

return new ModelAndView("update");

}

@PostMapping(value = "/update.htm")

public ModelAndView updateContactSpecific(ModelMap model, @ModelAttribute("contact")Contact contact, HttpServletRequest request) {

String source = request.getParameter("source");

if (source.equals("firstForm")) {

String contactId = request.getParameter("contact");

Contact con = contactDao.get(Integer.parseInt(contactId));

contact = con;

model.addAttribute("contact", contact);

return new ModelAndView("updateContact");

} else if (source.equals("finalForm")) {

Contact refresh = contactDao.get(contact.getId());

refresh.setName(contact.getName());

refresh.setPhone(contact.getPhone());

refresh.setAddress(contact.getAddress());

refresh.setId(Integer.parseInt(request.getParameter("id")));

contactDao.update(refresh);

request.setAttribute("contact", refresh);

return new ModelAndView("updateSuccess");

}

return new ModelAndView("login");

}

}

package com.neu.phonebook.controller;

import java.io.File;

import java.io.FileReader;

import java.io.IOException;

import java.util.ArrayList;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.ContactDao;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.Contact;

import com.neu.phonebook.model.User;

import com.opencsv.CSVReader;

@Controller

public class UploadContactListController {

@Autowired

UserDAO userDAO;

@Autowired

ContactDao contactDao;

@GetMapping(value = "/upload")

public ModelAndView downloadContactList(HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("username") == null) return new ModelAndView("redirect:/");

String username = (String) request.getSession().getAttribute("username");

User user = userDAO.get(username);

List<Contact> contacts = readContactsFile();

if(contacts==null) return new ModelAndView("fileUploadError");

for(Contact contact : contacts) {

contact.setUser(user);

contactDao.create(contact);

}

return new ModelAndView("redirect:/contactList.htm");

}

private List<Contact> readContactsFile() {

File file = new File("/Users/sunitbail/Downloads/contacts/contacts.csv");

List<Contact> contacts = new ArrayList<Contact>();

Contact contact;

if(!file.exists()) return null;

try {

FileReader outputfile = new FileReader(file);

CSVReader reader = new CSVReader(outputfile);

String[] nextRecord;

while((nextRecord = reader.readNext()) != null) {

if(!nextRecord[0].equals("Name")) {

contact = new Contact();

contact.setName(nextRecord[0]);

contact.setPhone(nextRecord[1]);

contact.setAddress(nextRecord[2]);

contacts.add(contact);

}

}

reader.close();

return contacts;

} catch (IOException e) {

return null;

}

}

}

package com.neu.phonebook.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

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

import org.springframework.web.servlet.ModelAndView;

import com.neu.phonebook.dao.UserDAO;

import com.neu.phonebook.model.User;

@Controller

public class UserListController {

@Autowired

UserDAO userDAO;

@GetMapping(value = "/userList.htm")

public ModelAndView userList(HttpServletRequest request) {

HttpSession hs = request.getSession(false);

if(hs.getAttribute("name") == null) return new ModelAndView("redirect:/");

List<User> userList = userDAO.getUserList();

request.setAttribute("userList", userList);

return new ModelAndView("userList");

}

}