Федеральное государственное бюджетное образовательное учреждение   
высшего образования

|  |  |
| --- | --- |
| Gerb-BMSTU_01 | «Московский государственный технический университет  им. Н.Э. Баумана (национальный исследовательский университет)»(МГТУ им. Н.Э. Баумана) |

ФАКУЛЬТЕТ – Информатика и управления

КАФЕДРА – Информационные системы и телекоммуникации

РАСЧЁТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе на тему

Разработка механизма хранения данных пользователей в СУБД для веб-сервиса Eclipse Che

Студент группы ИУ3-73 (подпись) 16.02.2017 А.А. Пирогов

Руководитель курсовой работы (подпись) 16.02.2017 А.М. Иванов

Москва, 2017

Рабочий код программы:

/\*\*

\*

\*/

package ru.agentlab.user.service.ds.host;

import java.io.IOException;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.Collection;

import java.util.Dictionary;

import java.util.HashMap;

import java.util.Iterator;

import java.util.List;

import java.util.Map;

import org.osgi.service.cm.ConfigurationException;

import org.osgi.service.cm.ManagedService;

import org.osgi.service.component.ComponentContext;

import org.osgi.service.component.annotations.Activate;

import org.osgi.service.component.annotations.Component;

import org.osgi.service.component.annotations.Deactivate;

import org.osgi.service.component.annotations.Modified;

import ru.agentlab.user.service.IUserService;

import ru.agentlab.user.service.model.Group;

import ru.agentlab.user.service.model.StringRequest;

import ru.agentlab.user.service.model.User;

import ru.agentlab.user.service.model.UserGroupRequest;

/\*\*

\*

\* User service realisation

\*

\* @author Pirogov

\*

\*/

@Component(enabled = true, immediate = true,

property = { "service.exported.interfaces=\*", "service.exported.configs=ecf.jaxrs.jersey.server",

"ecf.jaxrs.jersey.server.urlContext=http://localhost:8080", "ecf.jaxrs.jersey.server.alias=/userservice",

"service.pid=ru.agentlab.user.service.ds.host.UserServiceComponent" })

public class UserServiceComponent

implements IUserService, ManagedService {

private Collection<User> users;

private Map<User, List<Group>> usersGroups;

@Override

public User getUser(StringRequest login) {

return getUser(login.getValue());

}

@Override

public List<Group> getUserGroups(StringRequest login) {

for (User user : usersGroups.keySet())

{

if (user.getLogin().equals(login.getValue()))

{

return usersGroups.get(user);

}

}

return null;

}

@Override

public void addUser(User user) {

if (!users.contains(user))

{

users.add(user);

}

}

@Override

public void removeUser(StringRequest login) {

Iterator<User> userIterator = users.iterator();

while (userIterator.hasNext())

{

User user = userIterator.next();

if (user.getLogin().equals(login.getValue()))

{

userIterator.remove();

}

}

}

@Override

public void addUserGroup(UserGroupRequest request) {

User user = getUser(request.getLogin());

if (user != null)

{

List<Group> userGroups = usersGroups.get(user);

if (userGroups == null)

{

usersGroups.put(user, new ArrayList<>(Arrays.asList(request.getGroup())));

}

else

{

userGroups.add(request.getGroup());

}

}

}

@Override

public void removeUserGroup(UserGroupRequest request) {

User user = getUser(request.getLogin());

if (user != null)

{

List<Group> userGroups = usersGroups.get(user);

if (userGroups != null)

{

userGroups.remove(request.getGroup());

}

}

}

@Activate

public void activate(ComponentContext context) throws IOException {

users = new ArrayList<>();

usersGroups = new HashMap<>();

System.out.println("UserService service started"); //$NON-NLS-1$

}

@Deactivate

public void deactivate(ComponentContext context) {

System.out.println("UserService service stopped"); //$NON-NLS-1$

}

@Modified

public void modify() {

System.out.println("UserService service modified"); //$NON-NLS-1$

}

@Override

public void updated(Dictionary<String, ?> properties) throws ConfigurationException {

// Does nothing

}

private User getUser(String login) {

for (User user : users)

{

if (user.getLogin().equals(login))

{

return user;

}

}

return null;

}

}

/\*\*

\*

\*/

package ru.agentlab.user.service.ds.host;

import java.io.IOException;

import java.util.Dictionary;

import org.osgi.service.cm.ConfigurationAdmin;

import org.osgi.service.cm.ConfigurationException;

import org.osgi.service.cm.ManagedService;

import org.osgi.service.component.ComponentContext;

import org.osgi.service.component.annotations.Activate;

import org.osgi.service.component.annotations.Component;

import org.osgi.service.component.annotations.Deactivate;

import org.osgi.service.component.annotations.Modified;

import org.osgi.service.component.annotations.Reference;

/\*\*

\*

\* User service configurer.

\*

\* @author Pirogov

\*

\*/

@Component(enabled = true, immediate = true)

public class UserServiceConfigurer implements ManagedService {

@Reference

public void bindCm(ConfigurationAdmin configAdmin) throws IOException {

// Does nothing

}

public void unbindCm(ConfigurationAdmin configAdmin) {

// Does nothing

}

@Activate

public void activate(ComponentContext context) {

System.out.println("UserServiceConfigurer service started"); //$NON-NLS-1$

}

@Deactivate

public void deactivate(ComponentContext context) {

System.out.println("UserServiceConfigurer service stopped"); //$NON-NLS-1$

}

@Modified

public void modify() {

System.out.println("UserServiceConfigurer service modified"); //$NON-NLS-1$

}

@Override

public void updated(Dictionary<String, ?> properties) throws ConfigurationException {

// Does nothing

}

}





