package com.admin.service.impl;

import java.util.ArrayList;

import java.util.Collections;

import java.util.List;

import org.apache.commons.mail.EmailException;

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

import org.springframework.stereotype.Service;

import com.admin.dao.AccountRepository;

import com.admin.dao.ChequeBookRequestsRepository;

import com.admin.dao.SaccountRepository;

import com.admin.dao.TransferRepository;

import com.admin.dao.UserDisplayRepository;

import com.admin.dao.UserRepository;

import com.admin.model.ChequebookRequest;

import com.admin.model.Transfer;

import com.admin.model.User;

import com.admin.model.UserDisplay;

import com.admin.service.AdminService;

@Service

public class AdminServiceImpl implements AdminService{

@Autowired

private TransferRepository transDAO;

@Autowired

private ChequeBookRequestsRepository chequeBookDAO;

@Autowired

private UserRepository userDao;

@Autowired

private UserDisplayRepository displayDao;

@Autowired

private AccountRepository accDao;

@Autowired

private SaccountRepository sAccDao;

@Autowired

private MailServiceImpl mailService;

@Autowired

private AccountsCreationImpl accService;

@Autowired

private SaccountCreationImpl sAccService;

@Override

public void authorizeUser(String username) {

userDao.authorizeUser(username);

System.out.println("error here top");

User currUser = userDao.findByUsername(username);

int userId = currUser.getId();

System.out.println("error here 1");

accService.newAccount(username,userId);

System.out.println("error here 2");

sAccService.newAccount(username,userId);

}

@Override

public void cancelAuthorization(String username) {

userDao.cancelAuthorization(username);

}

@Override

public void acceptChequebookRequest(long accNo) {

String username = "";

chequeBookDAO.setChequebookInfoByAccount(accNo);

if(Long.toString(accNo).length() == 7) {

username = accDao.findByAccno(accNo).getUsername();

}

else {

username = sAccDao.findByAccno(accNo).getUsername();

}

try {

mailService.sendChequebookConfirmedEmail(username);

} catch (EmailException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

@Override

public void enableUser(String username) {

userDao.enableUser(username);

}

@Override

public void disableUser(String username) {

userDao.disableUser(username);

}

@Override

public List<UserDisplay> getAllUsers() {

return displayDao.getAllUsers();

}

@Override

public List<Transfer> getAllTransactions(long accountNo) {

List<Transfer> sender=transDAO.findBySaccount(accountNo);

List<Transfer> receiver=transDAO.findByRaccount(accountNo);

List<Transfer> merged=new ArrayList<>();

merged.addAll(sender);

merged.addAll(receiver);

Collections.sort(merged);

return merged;

}

@Override

public List<ChequebookRequest> getAllChequebookRequests() {

return chequeBookDAO.findAllChequebookRequests();

}

@Override

public List<User> getAllUnauthorizedUsers() {

return userDao.findAllUnauthorizedAccounts();

}

@Override

public void setUserFeatures(String username, int featureId) {

userDao.setUserFeatureStatus(username,featureId);

}

static boolean isNumber(String s)

{

for (int i = 0; i < s.length(); i++)

if (Character.isDigit(s.charAt(i))

== false)

return false;

return true;

}

@Override

public UserDisplay searchUser(String userDetail) {

if(isNumber(userDetail)) {

return displayDao.getUserDetailsByAccountNo(Long.parseLong(userDetail));

}

else {

return displayDao.getUserDetailsByUsername(userDetail);

}

}

}