package com.icinbank.service.impl;

import java.time.LocalDate;

import java.util.List;

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

import org.springframework.stereotype.Service;

import com.icinbank.dao.AccountRepository;

import com.icinbank.dao.ChequeBookRepository;

import com.icinbank.dao.SaccountRepository;

import com.icinbank.model.Account;

import com.icinbank.model.ChequebookRequest;

import com.icinbank.model.Saccount;

import com.icinbank.response.ChequeResponse;

import com.icinbank.service.ChequebookService;

@Service

public class ChequebookServiceimpl implements ChequebookService{

@Autowired

private ChequeBookRepository dao;

@Autowired

private AccountRepository adao;

@Autowired

private SaccountRepository sdao;

@Override

public ChequeResponse createrequest(ChequebookRequest chequebook) {

ChequeResponse response=new ChequeResponse();

long account = chequebook.getAccount();

List<ChequebookRequest> prevRequests = dao.findByAccount(account);

if(!prevRequests.isEmpty()) {

for(int i=0;i<prevRequests.size();i++) {

if(prevRequests.get(i).isRequestStatus()==false) {

response.setResponseMessage("Your previous chequebook request is still pending.");

response.setStatus(false);

response.setAccount(account);

return response;

}

}

}

LocalDate today = LocalDate.now();

if(isprimary(account)) {

try {

Account account1 =adao.findByAccno(account);

response.setAccount(account1.getAccno());

response.setStatus(true);

response.setResponseMessage("Request submitted successfully");

chequebook.setAccType("Primary");

chequebook.setDate(today);

chequebook.setRequestStatus(false);

//dao.save(chequebook);

dao.save(chequebook);

}

catch(Exception e) {

response.setAccount(account);

response.setStatus(false);

response.setResponseMessage("account number is incorrect");

}

}

else {

if(isSecondary(account)) {

try {

Saccount saccount=sdao.findByAccno(account);

response.setAccount(saccount.getAccno());

response.setStatus(true);

response.setResponseMessage("Request submitted successfully");

chequebook.setRequestStatus(false);

chequebook.setAccType("Secondary");

chequebook.setDate(today);

dao.save(chequebook);

}

catch (Exception e) {

response.setAccount(account);

response.setStatus(false);

response.setResponseMessage("account number is incorrect");

}

}

else

{

response.setAccount(account);

response.setStatus(false);

response.setResponseMessage("account number is incorrect");

}

}

return response;

}

@Override

public List<ChequebookRequest> getRequests(long account) {

return dao.findByAccount(account);

}

public static boolean isprimary(long account) {

String s = Long.toString(account).substring(0, 10);

String check="3914918201";

if(s.equals(check)) {

return true;

}

else

{

return false;

}

}

public static boolean isSecondary(long account) {

String s = Long.toString(account).substring(0, 10);

String check="3914918202";

if(s.equals(check)) {

return true;

}

else

{

return false;

}

}

}