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

import org.springframework.security.core.GrantedAuthority;

import org.springframework.security.core.userdetails.UserDetails;

import org.springframework.security.core.userdetails.UserDetailsService;

import org.springframework.security.core.userdetails.UsernameNotFoundException;

import org.springframework.stereotype.Service;

import java.util.Collection;

import java.util.stream.Collectors;

@Service

public class UserDetailsServiceImpl implements UserDetailsService {

@Autowired

private UserRepository userRepository;

@Override

public UserDetails loadUserByUsername(String username) throws UsernameNotFoundException {

User user = userRepository.findByUsername(username)

.orElseThrow(() -> new UsernameNotFoundException("User not found with username: " + username));

return UserDetailsImpl.build(user);

}

}

import org.springframework.security.core.GrantedAuthority;

import org.springframework.security.core.authority.SimpleGrantedAuthority;

import org.springframework.security.core.userdetails.UserDetails;

import java.util.Collection;

import java.util.List;

import java.util.stream.Collectors;

public class UserDetailsImpl implements UserDetails {

private Long id;

private String username;

private String password;

private Collection<? extends GrantedAuthority> authorities;

public UserDetailsImpl(Long id, String username, String password,

Collection<? extends GrantedAuthority> authorities) {

this.id = id;

this.username = username;

this.password = password;

this.authorities = authorities;

}

public static UserDetailsImpl build(User user) {

List<GrantedAuthority> authorities = user.getRoles().stream()

.map(role -> new SimpleGrantedAuthority(role.getName()))

.collect(Collectors.toList());

return new UserDetailsImpl(

user.getId(),

user.getUsername(),

user.getPassword(),

authorities

);

}

// Implement other UserDetails methods...

}