

```

package com.example.springsecurityapplication.services;

import com.example.springsecurityapplication.models.Person;
import
com.example.springsecurityapplication.repositories.PersonRepository;
import com.example.springsecurityapplication.security.PersonDetails;
import org.springframework.beans.factory.annotation.Autowired;
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 org.springframework.transaction.annotation.Transactional;

import java.util.Optional;

@Service
@Transactional(readonly = true)
public class PersonDetailsService implements UserDetailsService {

    private final PersonRepository personRepository;

    @Autowired
    public PersonDetailsService(PersonRepository personRepository) {
        this.personRepository = personRepository;
    }

    @Override
    public UserDetails loadUserByUsername(String username) throws
UsernameNotFoundException {
        // Получаем пользователя из таблицы person по логину с формы
аутентификации
        Optional<Person> person = personRepository.findByLogin(username);

        // Если пользователь не был найден
        if(person.isEmpty()){
            // Выбрасываем исключение что данный пользователь не найден
            // Данное исключение будет поймано спринг секьюрити и
сообщение будет выведено на страницу
            throw new UsernameNotFoundException("Пользователь не
найден");
        }
        // Возвращаем объект пользователя
        return new PersonDetails(person.get());
    }
}

```