

Информациска безбедност

Домашна работа 2

Професор:

Др. Димитрова Весна

Подготвил:

Стефан Стошевски, 195085

Ноември 2023 година

Подготвил: Стефан Стошевски

Документација

Во оваа лабораториска вежба користам spring security како главен систем за login/register и за безбедност на апликацијата. Користам custom-authentication based модел каде имам неколку класи и методи кои што детално ќе ги опишам подолу.

1. Model

- **Class**- Имам класа User во која чувам:
 - username кое исто така ми е и id во мојата база на податоци.
 - password кој го енкодирам со BCrypt со должина 10.
 - чувам name I surname кое ги користам при регистрација на корисник.
 - користам и неколку приватни променливи кое се потребни при користење на ваква аутентикација на корисник.
- **Exception**
 - Користам неколку exceptions со цел да се прикаже error на корисник при регистрација/login во смисла лошо внесен password.
- **Enum**
 - Користам енумен поле Role во кој имам 2 променливи role_user и role_admin каде одредувам кој какви привилегии има во мојата апликација.

2. Repository

- Користам spring JPA за зачувување на податоците во база и имам уште 2 методи:
- **findByUsername**- оваа функција има за цел да го најде корисникот во база според username.
- **findByUsernameAndPassword**- оваа функција има за цел да го најде корисникот во база според username и password.

3. Service

- -Имам `AuthService` и `UserService` и нивна имплементација. Имам неколку методи кои ги дефинирам подолу.
- **User login(String username, String password)**- прима 2 аргументи и овозможува логин функција
- **User register(.....)**- прима `username`, `password`, `repeat password`, `name`, `surname` и `role`. Ова е за регистрација на корисникот.
- Во **AuthServiceImpl** имам метода која што се вика `public User login`. Оваа метода има за цел да го овозможи логирањето на корисникот на мојата апликација. Во случај `username` да не е внесен или `password` да не е внесен ќе фрли `error InvalidArgumentException` која што е класа која се наоѓа во `exception` пакетот кој што го дефинирав погоре.
- Во **UserServiceImpl** имам метода која што се вика `public User register`. Оваа метода има за цел да го овозможи регистрањето на корисникот на мојата апликација. Во случај `username` да не е внесен или `password` да не е внесен ќе фрли `error InvalidArgumentException`, ако пасвордот што е внесен не е ист со пасвордот за проверка ќе фрли `PasswordDoNotMatchException`, а ако веќе постои таков `username` ќе фрли `UsernameAlreadyExistsException`.

4. Controller

- Во овој дел имам неколку контролери кои што ќе ги дефинирам подолу:
- **HomeController**-Овој контролер служи за прикажување на главната содржина на мојата апликација кое е мапирана на `/home`.
- **LoginController**-Овој контролер служи за овозможување на функционирање на логин формата, односно овозможување на корисникот да се најави на апликацијата.
- **LogoutController**-Овој контролер служи за одјавување на корисникот од апликацијата.
- **RegisterController**-Овој контролер служи за регистрање на корисникот на апликацијата.

5. Config

- -Во овој дел имам 2 класи кои детално ќе објаснам за нив.
- **WebSecurityConfig**-Во овој имам 2 методи кои се:
- **configure(http)**- Имам дефинирано неколку пермисии кој како може да пристапи на апликацијата. За логин најава имам дозволено на сите, имам поставено доколку не е добро внесено да даде error во урлто, при успешна најава да ме однесе на home странат, да го избрише cookie и да овозможи logout.
- **configure(AuthenticationManagerBuilder)**- Оваа тука презема се што е дефинирано во CustomUsernamePasswordAuthentication
- **CustomUsernamePasswordAuthentication**- Имам 2 методи кои детално ќе објаснам за нив.
- **Authenticate**-Овој метод служи за аутентикација на корисникот кој што ги зема username и password и проверува дали добро се внесени и дали може корисникот да се најави.
- **Supports**-Ова се користи за да се зема токенот за username и password.

Користена литература: stackoverflow, google, github, знаење од веб програмирање.