



Tutorial 08 – Web Security

CSIM603026 – Arsitektur dan Pemrograman Aplikasi Perusahaan

Semester Genap 2017/2018

Ruth Nolytha Putri
1606955006

Ringkasan

- Autentikasi dan Otorisasi diperlukan dalam pengembangan aplikasi untuk melakukan verifikasi akses pengguna terhadap aplikasi dan untuk mengatur hak/fitur pengguna yang telah terverifikasi
- Anotasi @Controller diperlukan pada setiap class Controller aplikasi. Semua class Controller pada aplikasi akan digunakan/dijalankan methodnya sehingga tidak boleh ada path @RequestMapping yang sama walau pada class Controller yang berbeda
- Autentikasi pada aplikasi bisa dilakukan secara in-memory (disimpan pada memory) maupun database. Spring menggunakan AuthenticationManagerBuilder untuk membantu menangani masalah autentikasi pada aplikasi
- Thymeleaf menyediakan security basic bagi spring untuk menangani login dan error pages. Misalnya penggunaan HttpSecurity pada method configure class WebSecurityConfig untuk handle login page

Latihan

1. Buatlah hal yang sama dengan menambahkan tombol Sign Out dan cetak nama user untuk halaman student/view/{npm}

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
  <head>
    <title>View Student by NPM</title>
  </head>
  <head th:replace="fragments/fragment :: assets" />
  <body>
    <div th:replace="fragments/fragment :: header"></div>
    <h2 th:text="'Login as ' + ${#httpServletRequest.remoteUser}">Login as</h2>
    <form th:action="@{/logout}" method="post">
      <input type="submit" value="Sign Out"/>
    </form>
    <h3 th:text="'NPM = ' + ${student.npm}">Student NPM</h3>
    <h3 th:text="'Name = ' + ${student.name}">Student Name</h3>
    <h3 th:text="'GPA = ' + ${student.gpa}">Student GPA</h3>
    <div th:replace="fragments/fragment :: footer"></div>
  </body>
</html>
```

Fungsi logout pada latihan ini diletakkan pada method view dengan RequestMapping "student/view". User dengan role yang diassign akan bisa mengakses fitur ini. Selain itu, pada Latihan ini juga dilakukan pemanggilan nama user untuk ditampilkan pada view dengan menggunakan fungsi yang disediakan Thymeleaf yaitu `httpServletRequest.remoteUser`. Hasil latihan dapat dilihat pada gambar berikut:

Login as user

Sign Out

NPM = 141021354001

Name = Afifah

GPA = 2.98

Mata Kuliah APAP

2. Buatlah user baru dengan username "user", password "user", dan role "USER". Coba apakah bisa login dengan akun tersebut

```
@Autowired
public void configureGlobal(AuthenticationManagerBuilder auth) throws Exception{
    auth.inMemoryAuthentication()
        .withUser("admin").password("admin").roles("ADMIN");
    auth.inMemoryAuthentication()
        .withUser("user").password("user").roles("USER");
}
```

User baru dengan username, password, dan role tertentu bisa ditambahkan untuk digunakan pada aplikasi dengan menggunakan in-memory authentication seperti pada gambar diatas

User Name :

Password :

Sign In

User baru dengan username yang telah ditambahkan pada configureGlobal dapat digunakan untuk login pada aplikasi

Welcome To Tutorial 08 APAP

NPM :

Cari

Mata Kuliah APAP

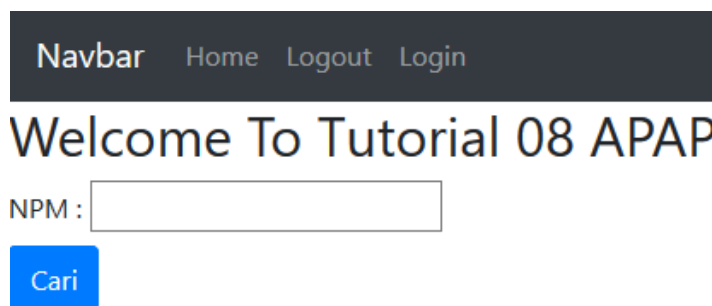
Setelah user berhasil diverifikasi, user bisa mengakses fitur dan halaman yang diizinkan untuk diakses dengan role user tersebut

3. Tampilkan menu view student by npm beserta input field dan button untuk memasukan nomor npm student yang ingin dicari pada halaman index user dengan role USER

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <title>Index</title>
</head>
<head th:replace="fragments/fragment :: assets" />
<body>
  <div th:replace="fragments/fragment :: header"></div>
  <h2>Welcome To Tutorial 08 APAP</h2>

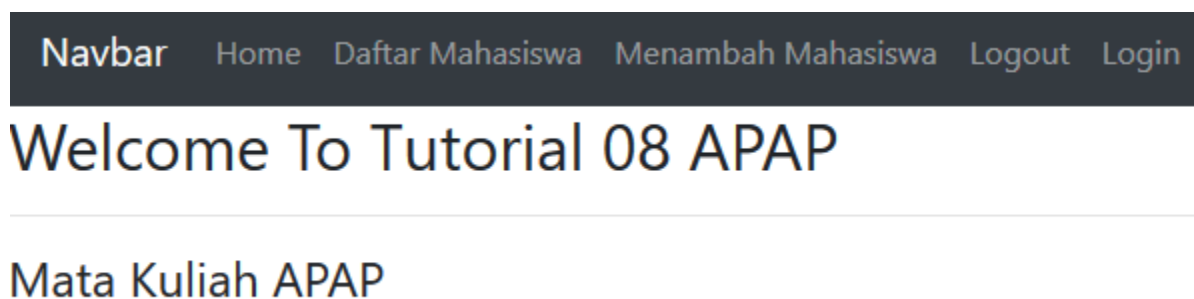
  <form action="/student/view" method="get" th:if="${#httpServletRequest.isUserInRole('USER')}">
    <div>
      <label>NPM : <input type="text" name="npm"></label>
    </div>
    <div class="text-left">
      <button type="submit" class="btn btn-primary">Cari</button>
    </div>
  </form>
  <div th:replace="fragments/fragment :: footer"></div>
</body>
</html>
```

Untuk membuat form pencarian mahasiswa berdasarkan npm hanya bisa dilihat oleh role tertentu, maka digunakan fungsi Thymeleaf `httpServletRequest.isUserInRole` yang mengembalikan Boolean yang mengindikasikan apakah user tersebut ter-autentikasi dengan role yang dispesifikasikan.



Mata Kuliah APAP

Gambar diatas merupakan halaman index aplikasi dengan user yang ter-autentikasi memiliki role USER



Gambar diatas merupakan halaman index aplikasi dengan user yang ter-autentikasi memiliki role ADMIN. Form pencarian mahasiswa berdasarkan NPM tidak akan ditampilkan

4. Buat agar user dengan role ADMIN dapat melihat view all student dan view student by npm juga

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Index</title>
</head>
<head th:replace="fragments/fragment :: assets" />
<body>
    <div th:replace="fragments/fragment :: header"></div>
    <h2>Welcome To Tutorial 08 APAP</h2>

    <form action="/student/view" method="get" th:if="${#httpServletRequest.isUserInRole('ADMIN')}
    ||#httpServletRequest.isUserInRole('USER')}">
    <div>
        <label>NPM : <input type="text" name="npm"></label>
    </div>
    <div class="text-left">
        <button type="submit" class="btn btn-primary">Cari</button>
    </div>
    </form>
    <div th:replace="fragments/fragment :: footer"></div>
</body>
</html>
```

Agar user dengan role ADMIN juga bisa melihat form pencarian mahasiswa, maka ditambahkan `httpServletRequest.isUserInRole` yang untuk digunakan role ADMIN.

```
@Configuration
@EnableWebSecurity
public class WebSecurityConfig extends WebSecurityConfigurerAdapter{
    @Override
    protected void configure(HttpSecurity http) throws Exception{
        http.authorizeRequests()
            .antMatchers("/").permitAll()
            .antMatchers("/student/viewall").hasRole("ADMIN")
            .antMatchers("/student/view/**").hasAnyRole("ADMIN", "USER")
            .anyRequest()
            .authenticated()
            .and()
            .formLogin()
            .loginPage("/login").permitAll()
            .and()
            .logout().permitAll();
    }
}
```

Pada controller role ADMIN juga perlu diberikan authorize untuk mengakses RequestMapping "student/view"

Navbar Home Daftar Mahasiswa Menambah Mahasiswa Logout Login

Welcome To Tutorial 08 APAP

NPM :

Cari

Mata Kuliah APAP

Gambar diatas adalah tampilan halaman index untuk user dengan role ADMIN