

Tutorial 8

Pada tutorial ini yang dipelajari adalah sistem authentication dan authorization. Authentication berhubungan dengan verifikasi apakah user tertentu memiliki hak akses untuk masuk ke dalam sistem, yang digunakan adalah autentikasi username dan password. Sedangkan authorization berhubungan dengan privilege user yang telah di autentikasi untuk dapat mengakses fitur yang ada di aplikasi, yang digunakan adalah role. Contohnya role admin biasanya memiliki akses full ke semua fitur aplikasi, sedangkan role user hanya beberapa fitur saja yang berhubungan dengan user. Pada tutorial ini menggunakan dua hierarchy role user yaitu admin dan user. Admin dapat mengakses fitur view all student dan view student by npm. Sedangkan role user hanya dapat mengakses view student by npm. Implementasi menggunakan dependency bawaan spring boot yaitu spring boot security. Jadi terdapat dua mekanisme untuk autentikasi yaitu set manually user dan password yang bisa login ke aplikasi menggunakan method configureGlobal, method override dari class WebSecurityConfigurerAdapter. Kemudian mekanisme autentikasi yaitu menggunakan data user dari database.

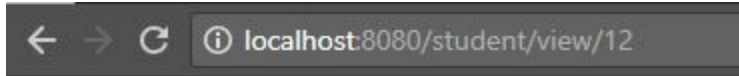
Latihan

1. Menambahkan tombol signout yang akan redirect ke login dan menambahkan nama user yang login di page view student by npm.
Code yang ditambahkan adalah di view.html

```
<body>
  <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>
</body>
```

Yang dilakukan hanya menambahkan informasi user yang sedang login di view yaitu dengan code `#httpServletRequest.remoteUser`. Kemudian button Sign Out yang akan redirect ke `/logout`

Tampilan view nya adalah sebagai berikut:



Login as admin

[Sign Out](#)

NPM = 12

Name = Ayu

GPA = 2.2

2. Menambahkan user baru yaitu user dan password user dengan role USER

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

    auth.inMemoryAuthentication()
        .withUser("user").password("user")
        .roles("USER");
}
```

Code diatas ditambahkan pada class WebSecurityConfig di method configureGlobal.

User baru ini akan di simpan static sebagai inMemoryAuthentication, dan ketika menggunakan user ini untuk login akan berhasil karena telah di daftarkan sebagai user yang authenticated.

3. Menambahkan view student by npm untuk role USER dan menambah form untuk search by npm. Berikut adalah code di index.html

```

<html xmlns:th="http://www.thymeleaf.org">
<head>
<script type="text/javascript" th:src="@{/js/jquery-1.12.4.js}"></script>
<title>Index</title>
</head>
<body>
<h2>Welcome</h2>
<a href="/Login">Login</a><br>
<a th:if="${#httpServletRequest.isUserInRole('ADMIN')}" href="/student/viewall">View All Student</a><br>
<label th:if="${#httpServletRequest.isUserInRole('USER')}">View student By NPM</label><br>
<form id="myForm" th:if="${#httpServletRequest.isUserInRole('USER')}" th:action="@{/student/view/}" method="POST">
<input type="text" name="npm" id="npm" placeholder="Please input NPM"/>
<button type="submit" id="search">Search</button>
</form>
</body>

<script th:inline="javascript">
$(document).ready(function() {
var $form = $("#myForm");
var $idField = $("#npm");

console.log("test");
$form.submit( function( event ) {
    $form.attr( 'action', $form.attr('action') + $idField.val() );

    $idField.prop( "disabled", true );

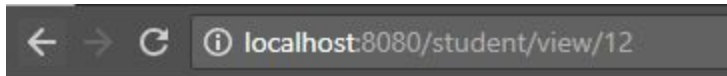
    return;
});
} );
</script>
</html>

```

Jika user yang login memiliki role USER maka akan ditampilkan form untuk search student by npm. Fungsi di controller yang dipanggil adalah `/student/view/{npm}`. Karena controller ini menggunakan pathVariable maka perlu ditambahkan npm di url yang akan diakses, maka digunakan javascript ketika form di submit maka ditambahkan npm pada url endpoint nya. Tampilannya adalah sebagai berikut



Ketika di search maka akan redirect ke view by npm page



Login as user

[Sign Out](#)

NPM = 12

Name = Ayu

GPA = 2.2

4. User admin dapat mengakses view student by npm
Yang dilakukan adalah menambah akses url view student by npm untuk role USER yaitu sebagai berikut

```
http
    .authorizeRequests()
    .antMatchers("/").permitAll()
    .antMatchers("/student/view/**").hasRole("USER")
    .antMatchers("/student/view/**", "/student/viewall").hasRole("ADMIN")
    .anyRequest().authenticated()
    .and()
    .formLogin()
    .loginPage("/login")
    .permitAll()
    .and()
    .logout()
    .permitAll();
```

Tampilannya adalah sebagai berikut

← → ↻ ⓘ localhost:8080/student/view/12

Login as admin

Sign Out

NPM = 12

Name = Ayu

GPA = 2.2