

MỤC TIÊU

- Đăng nhập từ mạng xã hội Google
- Sử dụng @PreAuthorize để phân quyền truy xuất

BÀI 1: ĐĂNG NHẬP TỪ GOOGLE (4 ĐIỂM)

Sao chép lại dự án của Lab2.1 và bổ sung, hiệu chỉnh để hoàn thành bài tập này theo yêu cầu sau:

- Tích hợp OAuth2 vào dự án
Khai báo Dependency vào pom.xml để tích hợp thư viện cần thiết vào dự án

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-oauth2-client</artifactId>
</dependency>
```

- Lấy Client ID và Secret từ Google Cloud và khai báo vào application.properties
Tạo và lấy Client Id và Client Secret từ Google Cloud và khai báo vào application.properties để ứng dụng web kết nối với Google trong quá trình đăng nhập

```
spring.security.oauth2.client.registration.google.client-id=<<client-id>>
spring.security.oauth2.client.registration.google.client-secret=<<client-secret>>
```

- Cấu hình đăng nhập từ mạng xã hội
 - SecurityFilterChain: Bổ sung đoạn mã sau đây để cấu hình đăng nhập từ mạng xã hội

```
http.oauth2Login(login -> {
    login.permitAll();
});
```

- Bổ sung liên kết sau đây vào form đăng nhập (login.html)

```
<a href="/oauth2/authorization/google">Google</a>
```

- Chạy và kiểm thử nhấp vào link Google để đăng nhập. Kiểm tra thông tin tài khoản sau đăng nhập.
- Tạo đối tượng Authentication mới thay thế đối tượng Authentication chứa thông tin đăng nhập từ mạng xã hội.

Hãy bổ sung đoạn mã sau đây vào vị trí ??? của http.oauth2Login(login -> {login.permitAll(); ???}) hoàn thiện yêu cầu

```
login.successHandler((request, response, authentication) -> {
    // Đọc UserDetails chứa thông tin người dùng từ Google
})
```



```

DefaultOidcUser user = (DefaultOidcUser) authentication.getPrincipal();
String username = user.getEmail();
String role = "OAUTH";

// Tạo đối tượng Authentication mới và thay thế đối tượng Authentication hiện tại
UserDetails newUser = User.withUsername(username).password("{noop}").roles(role).build();
Authentication newauth = new
    UsernamePasswordAuthenticationToken(newUser, null, newUser.getAuthorities());
SecurityContextHolder.getContext().setAuthentication(newauth);

// Chuyển về trang yêu cầu bảo mật hoặc trang chủ
HttpSession session = request.getSession();
String attr = "SPRING_SECURITY_SAVED_REQUEST";
DefaultSavedRequest req = (DefaultSavedRequest) session.getAttribute(attr);
String redirectUrl = (req == null) ? "/" : req.getRedirectUrl();
response.sendRedirect(redirectUrl);
});

```

- Chạy và kiểm thử nhấp vào link Google để đăng nhập. Kiểm tra thông tin tài khoản sau đăng nhập lại 1 lần nữa.

BÀI 2: SỬ DỤNG @PREAUTHORIZE (3 ĐIỂM)

Sao chép lại dự án của Lab2.1 và bổ sung, hiệu chỉnh để hoàn thành bài tập này theo yêu cầu sau bằng

- @Configuration: Kích hoạt chế độ phân quyền qua phương thức

```

@Configuration
@EnableWebSecurity
@EnableMethodSecurity(prePostEnabled = true)
public class SecurityConfig {...}

```

- @Bean SecurityFilterChain: Cấu hình cho phép truy cập tất cả

```

@Bean
public SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {
    // Bỏ cấu hình mặc định CSRF và CORS
    http.csrf(csrf -> csrf.disable()).cors(cors -> cors.disable());

    // Phân quyền sử dụng
    http.authorizeHttpRequests(req -> {
        req.anyRequest().permitAll();
    });
    ...
}

```

- @Controller: phân quyền truy cập cho các phương thức


```
@Controller
public class MyController {
    @GetMapping("/{", "/poly/url0"})
    public String method0(Model model) {
        model.addAttribute("message", "/poly/url0 => method0()");
        return "page";
    }
    @PreAuthorize("isAuthenticated()") @GetMapping("/poly/url1")
    public String method1(Model model) {
        model.addAttribute("message", "/poly/url1 => method1()");
        return "page";
    }
    @PreAuthorize("hasRole('USER')") @GetMapping("/poly/url2")
    public String method2(Model model) {...}
    @PreAuthorize("hasRole('ADMIN')") @GetMapping("/poly/url3")
    public String method3(Model model) {...}
    @PreAuthorize("hasAnyRole('USER', 'ADMIN')") @GetMapping("/poly/url4")
    public String method4(Model model) {...}
}
```

- Chạy và kiểm thử các liên kết

BÀI 3: GIẢNG VIÊN CHO THÊM (3 ĐIỂM)