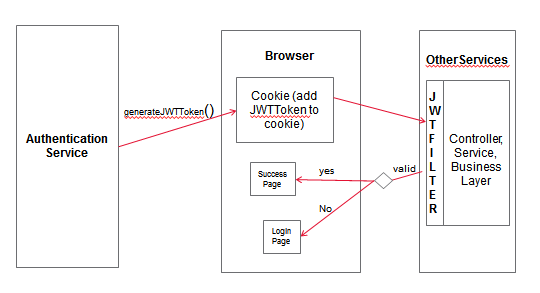
**SpringBoot with JWT:**



**Steps:**

1. In AuthenticationService add maven dependency for JWT in pom.xml.

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt</artifactId>

<version>0.6.0</version>

</dependency>

1. Configure the CorsRegistry in main class.

@Bean

**public** WebMvcConfigurer corsConfigurer() {

**return** **new** WebMvcConfigurerAdapter() {

};

}

1. Create a class file and add following method which generates the token using JWTBuilder.

**public** **static** String generateToken(String signingKey, String subject) {

**long** nowMillis = System.*currentTimeMillis*();

Date now = **new** Date(nowMillis);

JwtBuilder builder = Jwts.*builder*()

.setSubject(subject)

.setIssuedAt(now)

.signWith(SignatureAlgorithm.***HS256***, signingKey);

**return** builder.compact();

}

1. After Token is generated add the token in browser cookie. Now this cookie has to be read by other microservices . Once the token is validated we can redirect to the next page else to the Login Page.
2. Configure JWTFilter in the main class of second microservice . This will read the cookie which has be added by AuthenticationService.

@Bean

**public** ~~FilterRegistrationBean~~ jwtFilter() {

**final** ~~FilterRegistrationBean~~ registrationBean = **new** FilterRegistrationBean();

registrationBean.setFilter(**new** JwtFilter());

registrationBean.setInitParameters(Collections.*singletonMap*("next.page", orderpg));

**return** registrationBean;

}