Authentication and authorization   
::same answer

Key things to consider : input field validation

Auditing and logging

1 done out of 8

Here the entire flow is like , first there is request and response

Then once request is being thrown then there is a layer of filters , amongst the filter there is one filter named as authentication filter `

It gives the request details come along with the request send to the authentication manager

Manager has authentication provider , it has two tools named as passwordEncoder and userDetails service

Here passwordEncoder uised to save the string format password to its encoded format.

Then userdetails service loads the data by id , which comes along with request ,

In authentication provider request details and user details service are checked if its matched theall the data collected by the user details service is sent to response level.

2 done out of 7.

Add spring security. Dependency.

For cors handling along with the security

@Bean  
public SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {  
 http.cors() // Enable CORS  
 .and()  
 .csrf().disable() // Disable CSRF for APIs  
 .authorizeHttpRequests(auth -> auth  
 .requestMatchers("/users/\*\*").permitAll() // Allow public access to /users  
 .anyRequest().authenticated()  
 );  
  
 return http.build();  
}  
  
  
@Bean  
public CorsFilter corsFilter() {  
 CorsConfiguration config = new CorsConfiguration();  
 config.setAllowedOrigins(List.*of*("http://localhost:4200")); // Angular dev server  
 config.setAllowedMethods(List.*of*("GET", "POST", "PUT", "DELETE", "OPTIONS"));  
 config.setAllowedHeaders(List.*of*("\*"));  
 config.setAllowCredentials(true);  
  
 UrlBasedCorsConfigurationSource source = new UrlBasedCorsConfigurationSource();  
 source.registerCorsConfiguration("/\*\*", config); // Apply to all endpoints  
  
 return new CorsFilter(source);  
}

Default login and logout is set

To hit logout , go to /logout

We can add password statically in app.properties

Likwise spring.security.user.password = amol!@123

spring.security.user.name = amol

3 end