

Date : 28-Jun-21

Spring Boot 9AM

Mr. RAGHU

Open Authorization - BackEnd App

*) Model class

=> User (C) and AuthProvider(E)

In case of Any type of Authentication we can store user data in DB
but unique emailId.

AuthProvider - LOCAL, FACEBOOK, GOOGLE, GITHUB.

*) Repository :- UserRepository(save,fetch,delete)
get user data based on emailId
check given emailId exist or not.

*) Service : CustomUserDetailsService

At the time of login either using OAuth/JWT read current user
and load Db object as 'UserPrincipal' that holds both local + OAuth
Data.

=> On OAuth Login success we may get additional data
(image,emailId,Display name
..etc) inside "Map<String, Object> attributes".

=> Default role is 'ROLE_USER' for every user who logged in using Form
or OAuth.

*) AuthenticationEntryPoint : In case of invalid user/login failed
UNAUTHORIZED (401)

---Google OAuth Response-----

```
{
  "sub"    : "1542659633",
  "name"   : "RAGHU S RAGHU",
  "email"  : "javabyraghu@gmail.com",
  "picture":
"https://lh3.googleusercontent.com/ogw/ADea4I5hZ7GON3VHssZfGZVDRjU9Z3J
ohM6F7vuTNieISw=s32-c-mo",
}
```

---Facebook OAuth Response-----

```
{
  "id"      : "332323232323",
  "name"    : "RAGHU S RAGHU",
  "email"   : "javabyraghu@gmail.com",
  "picture": "https://scontent.fhyd2-2.fna.fbcdn.net/v/t1.18169-
9/10437437_1447435218838262_7626051183616333537_n.jpg?
_nc_cat=111&ccb=1-3&_nc_sid=09cbfe&_nc_ohc=-
Zg4mHHfbieAX8vdNMQ&_nc_ht=scontent.fhyd2-
2.fna&oh=6ab7357d4c5f4ee6105a1fd4af7a139e&oe=60DDB113"
}
```

---Github OAuth Response-----

```
{
  "id"      : "3434342",
  "name"    : "RAGHU S RAGHU",
  "email"   : "javabyraghu@gmail.com",
  "avatar_url" : "https://avatars.githubusercontent.com/u/15054479?v=4"
}
```

*) OAuth2UserInfoFactory :- is creating OAuth2UserInfo(AC) impl class based on AuthProvider type given by client application.

*) CustomOAuth2UserService : is using OAuth2UserInfoFactory and creating OAuth2UserInfo object and validating emailId, if already exist with any one provider, then don't allow same emailId for same/other providers.

If exist update name/Image URL else register into DB using UserModel class

*) Return User data with Attributes to Client App back.

*) A cookie is created by server and sent to browser using response to store some data, while sending request again we need to send all cookies using request header.

<https://www.mediafire.com/file/w5x9w5vcmkwkkdv/RaghuSirNareshITJavaPdfs.zip/file>

*) AdvJavaNitRaghuSirNareshIT.pdf
Page # 121

=====

<https://github.com/javabyraghu/react-social>
<https://github.com/javabyraghu/spring-social>

OAuth2UserInfo