# Spring Boot -Spring Security -login with Facebook and Github from scratch

What is OAuth 2.0 and how it works?

**OAuth** doesn't share password data but instead uses authorization tokens to prove an identity between consumers and service providers. **OAuth** is an authentication protocol that allows you to approve one application interacting with another on your behalf without giving away your password.

What is OAuth 2.0 used for?

The **OAuth 2.0** specification defines a delegation protocol that is useful for conveying authorization decisions across a network of web-enabled applications and APIs. **OAuth** is **used in** a wide variety of applications, including providing mechanisms for user authentication.

What is OAuth flow?

**OAuth** 2 is an authorization framework that enables applications to obtain limited access to user accounts on an HTTP service, such as Facebook, GitHub, and DigitalOcean. ... **OAuth** 2 provides authorization **flows** for web and desktop applications, and mobile devices.

When should you use OAuth?

**You should** only **use OAuth** if **you** actually need it. If **you** are building a service where **you** need **to use** a user's private data that is stored on another system — **use OAuth**

What is OAuth in REST API?

**OAuth** is an authorization framework that enables an application or service to obtain limited access to a protected HTTP resource. To use **REST APIs** with **OAuth** in Oracle Integration, you need to register your Oracle Integration instance as a trusted application in Oracle Identity Cloud Service.

Is OAuth a SSO?

**OAuth** (Open Authorization) is an open standard for token-based authentication and authorization which is used to provide single sign-on (**SSO**). **OAuth** allows an end user's account information to be used by third-party services, such as Facebook, without exposing the user's password.

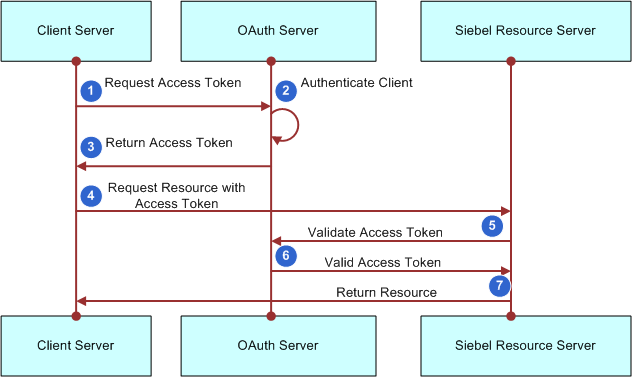
Is OAuth safe?

It's the most **secure** flow because you can authenticate the client to redeem the authorization grant, and tokens are never passed through a user-agent. There's not just Implicit and Authorization Code flows, there are additional flows you can do with **OAuth**. Again, **OAuth** is more of a framework.

What does OAuth stand for?

Open Authorization

OAuth, which stands for “**Open Authorization**,” allows third-party services to exchange your information without you having to give away your password.



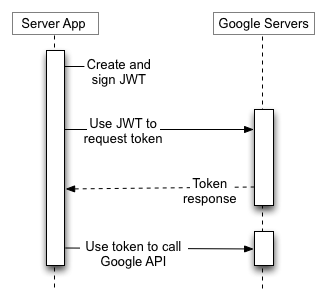
OAuth 2.0 defines a protocol, i.e. specifies how tokens are transferred, JWT defines a token format.

OAuth 2.0 and "JWT authentication" have similar appearance when it comes to the (2nd) stage where the Client presents the token to the Resource Server: the token is passed in a header.

But "JWT authentication" is not a standard and does not specify how the Client obtains the token in the first place (the 1st stage). That is where the perceived complexity of OAuth comes from: it also defines various ways in which the Client can obtain an access token from something that is called an Authorization Server.

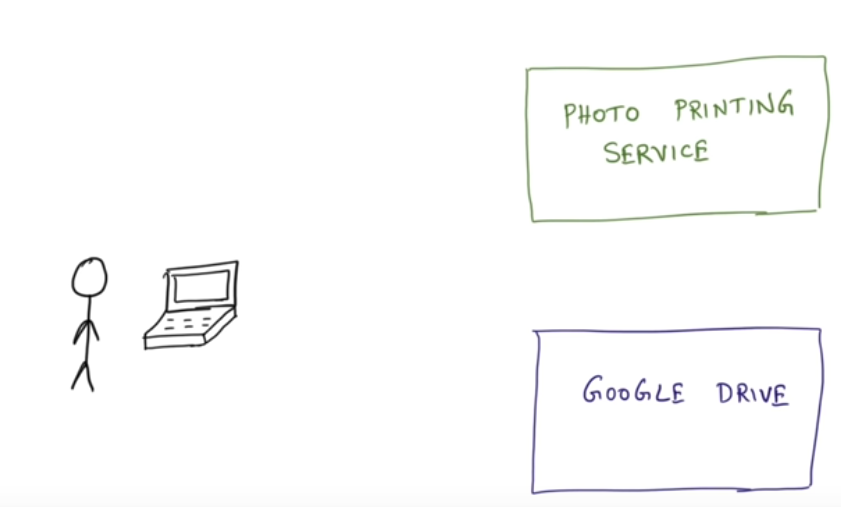
So the real difference is that JWT is just a token format, OAuth 2.0 is a protocol (that may use a JWT as a token format).

Firstly, we have to differentiate JWT and OAuth. Basically, JWT is a token format. OAuth is an authorization protocol that can use JWT as a token. OAuth uses server-side and client-side storage. If you want to do real logout you must go with OAuth2. Authentication with JWT token can not logout actually. Because you don't have an Authentication Server that keeps track of tokens. If you want to provide an API to 3rd party clients, you must use OAuth2 also. OAuth2 is very flexible. JWT implementation is very easy and does not take long to implement. If your application needs this sort of flexibility, you should go with OAuth2. But if you don't need this use-case scenario, implementing OAuth2 is a waste of time.



We will using only for authentication not for authorization

- will it possible only authenicate with out authorization



Ex :

-user trying to photoprinitng service in order to access file in google drive. The way photoprinting service access google drive is by oauth bcz service really cant get google credentials in order authorize the req

- here assumption user already authenticated to photoprinting service and already authticated to google drive and problem is to server to talk to each other

- we have 2 service both trust the user but this 2 service dnt trust each other

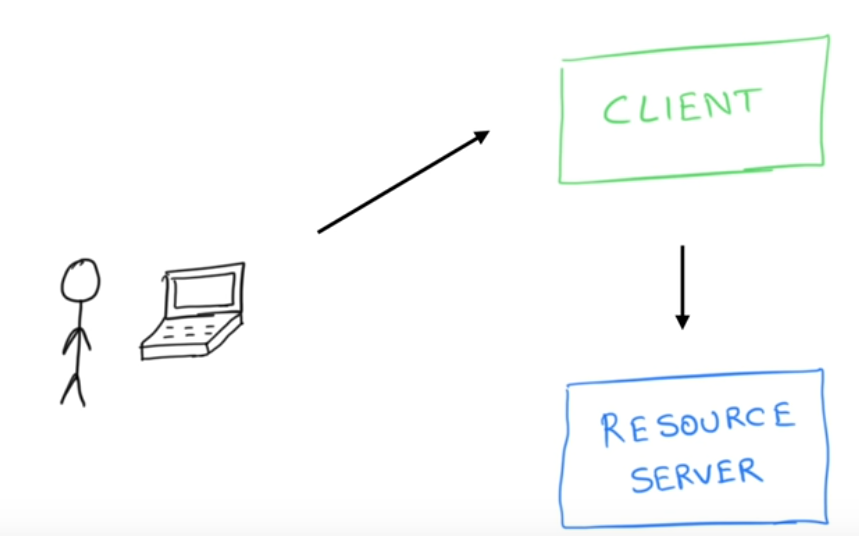
- this is the intended usage of oauth service to authorize with other services



If ur building client service (photoprinting service) trying to access Resource Server

-1 st we need to set authentication to ourself in photosprinting service we need to authenticate our user first cleint service

- when client is authenticated now please acess Resource Server on my behalf then cleint will have ability to access the resource server



- going to use google facebook etc I can get inforamtion about users of google account, git, facebook we can use oauth to get it using resource server

- what we do when Resource server use oauth request we are basically validating that user Google account resource server is google

- now we know google id of user

- is that case why I need to authenticate all over again user already authenticated with google already have reliable authentication why again I need to login

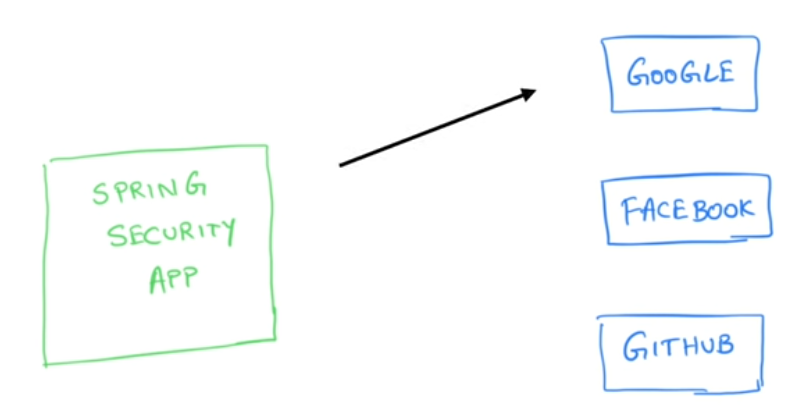
- resource servers are like popualr google , facbook we can deligate the authnetication

- able to aouth successfuly to one of this service get an emaild id service says it is authenticated

- can have one to one mapping create a account on my end or relay on google account

- cleint cant avoid authnecation on their own relaise on google or face book etc that basic we use oauth that is originaly ment for authorization of API

- using this to leverage some authentication services and avaoiding building authentication on our own



- will build Spring Security app instead of building authentication on our own will use oauth call to google this person has google account

- make oauth call to google - google might say I know or I don’t know this person if know is a legimit user here is his email id I ahve been verified

- now we building spring security app will be gud so I dnt have to build authentication so this make oauth call to get user profile information once we get user info

- our spring secuity app will put that in spring security context to simulate a logged in user. Its almost like user logged in but basically not forcing user to login using google oauth api to get profile info

- tech wt authentication is all about is user give his credentials get user profile and save it in security context for this we will use oauth instead of our authentication

We will create Spring Security app from Scrath and make this whole this work

- Spring Security has a annotation which lets do oauth single sign on functionality

- we don’t want any api from google

- photo printing service might want access photo from google drive

Spring Io

Spring\_Boot\_oauth\_google\_facbook\_git

Add dependecny of web and Spring security for now

There is oauth dependecny that <https://start.spring.io> does nt support wil add in pom once work space set

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-security</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<!-- https://mvnrepository.com/artifact/org.springframework.security.oauth.boot/spring-security-oauth2-autoconfigure

give access to single sign on ability-->

<dependency>

<groupId>org.springframework.security.oauth.boot</groupId>

<artifactId>spring-security-oauth2-autoconfigure</artifactId>

<version>2.3.3.RELEASE</version>

</dependency>

Once adding all dependency

Make application an oauth sso applicaiton

Just do oauth based authentication by adding one annotaion @EnableOauth2Sso - this is when we want leverage oauth for authentication

- with above annotaiton now application is ready for single sign on this is not an application where init login capabilitty we will using single sign on

Note: Not @EnableOauth2Client - this will be used when we want make an api call and dnt want levearage oauth for authentication

@SpringBootApplication

@~~EnableOAuth2Sso~~

public class SpringBootOauthGoogleFacbookGitApplication {

public static void main(String[] args) {

SpringApplication.run(SpringBootOauthGoogleFacbookGitApplication.class, args);

}

}

Now where it will get external inforamation to login

- application.yml will be set configuration properties this where spring security will look for oauth details who am I have to call whom I ahve to access to get it done for oauth inforamtion.

- add block config that will used to authenticate wth facebook , google , git etc

<https://developers.facebook.com/>

Takes fb login

Create a new app if not created app name : sachin\_oauth\_app

Will get KEY generated

Go to setting=>basics =: will see appid

App id is what uniquley identifies the app for facebook

Every identiy provider as similar concept

|  |
| --- |
| App ID  4145039038901281 |

App secrt

2f58379142e9dd3249498073c1360ec4

Only visible to developer

We need this values to for spring security to configure

**security:**

**oauth2:**

**client:**

**clientId:** 4145039038901281

**clientSecret:** 2f58379142e9dd3249498073c1360ec4

**accessTokenUri:** https://graph.facebook.com/oauth/access\_token

**userAuthorizationUri:** httpa://www.facebook.com/dialog/oauth

**tokenName: oauth\_token**

**authenticationScheme:** query

**clientAuthenticationScheme:** form

**resource:**

**userInfoUri:** <https://graph.facebook.com/me>

Never share client id and secret id to any one it need to secured

- we run the application spring security will say this is on oauth application look at yml file make call to facebook

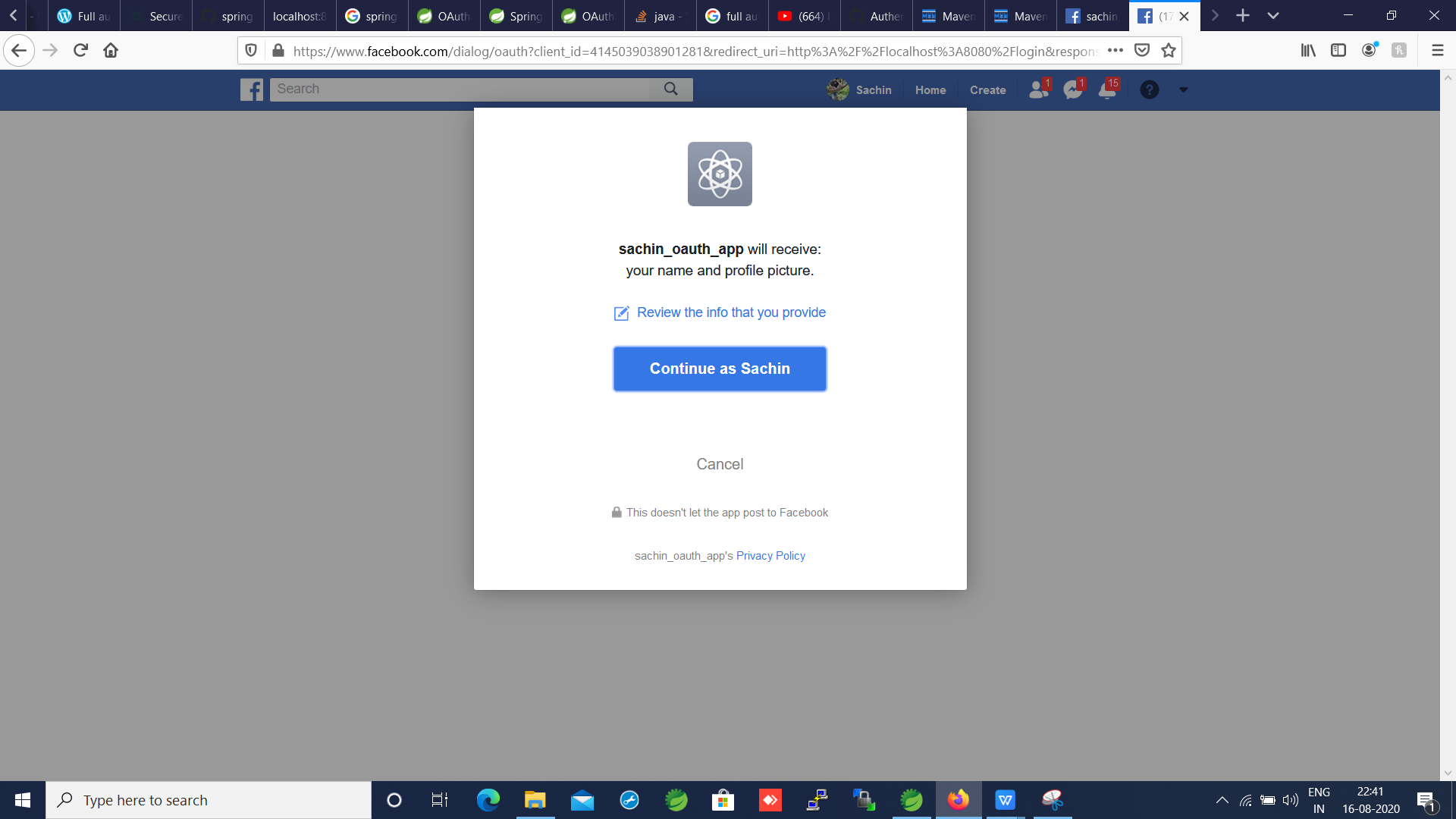
- will make oauth call with authorization token and gets access token after that this is the authorzation flow that follows

- create one html static file to c if authentication goes successfull

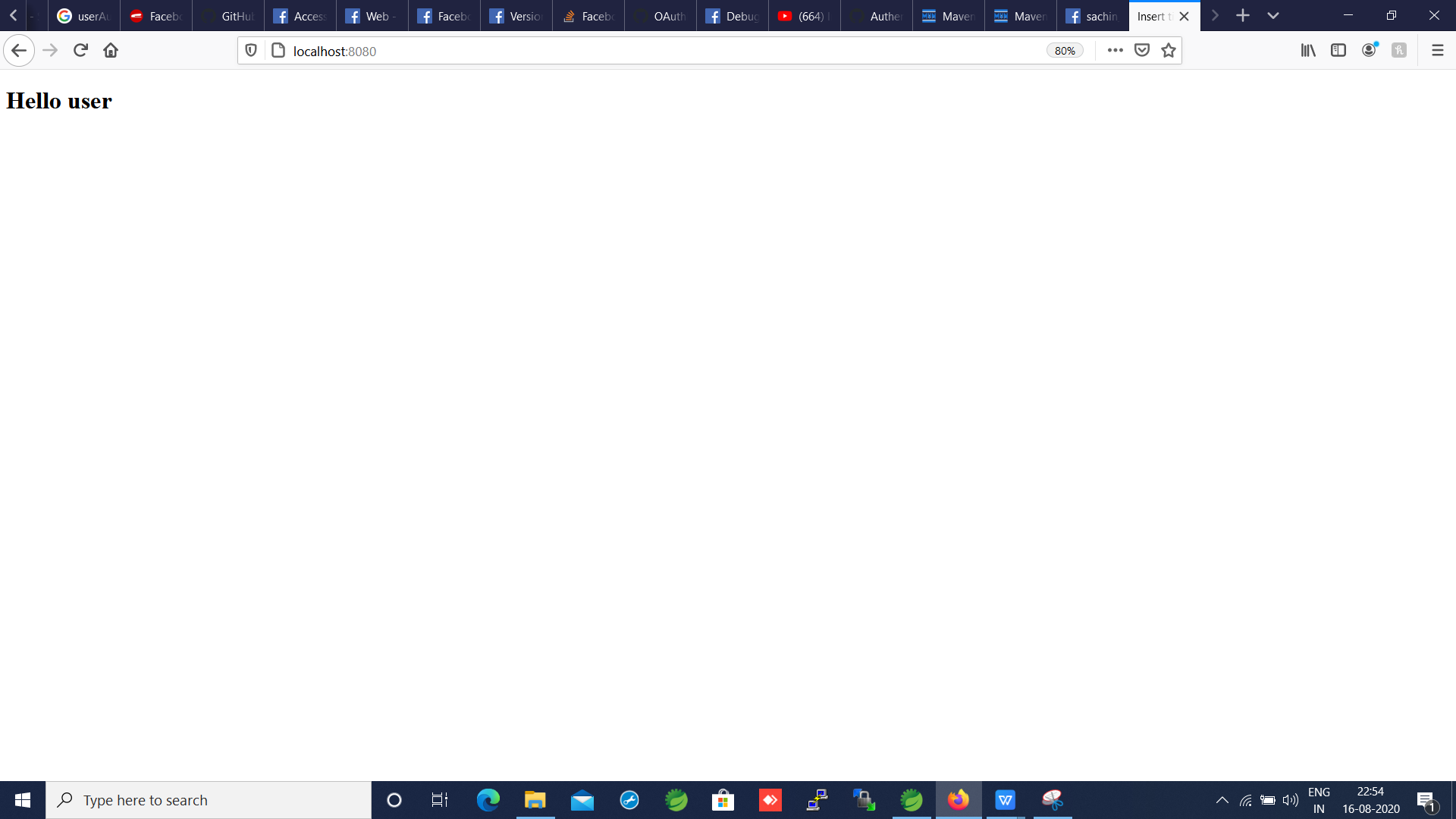
Index.html

Now start application and access it using localhost:8080

By default spring security will not give default login page it will facebook login permission



After ok

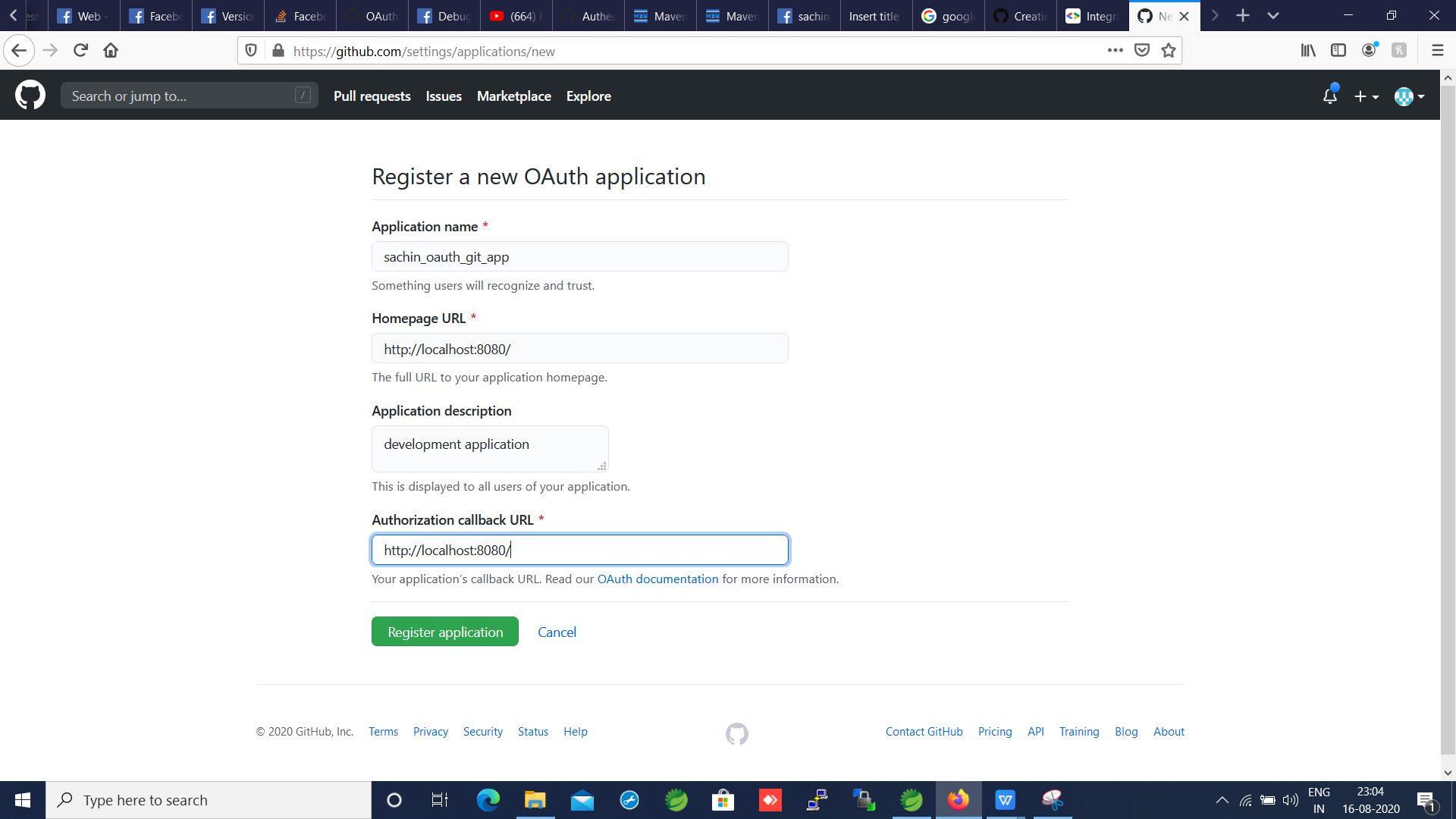


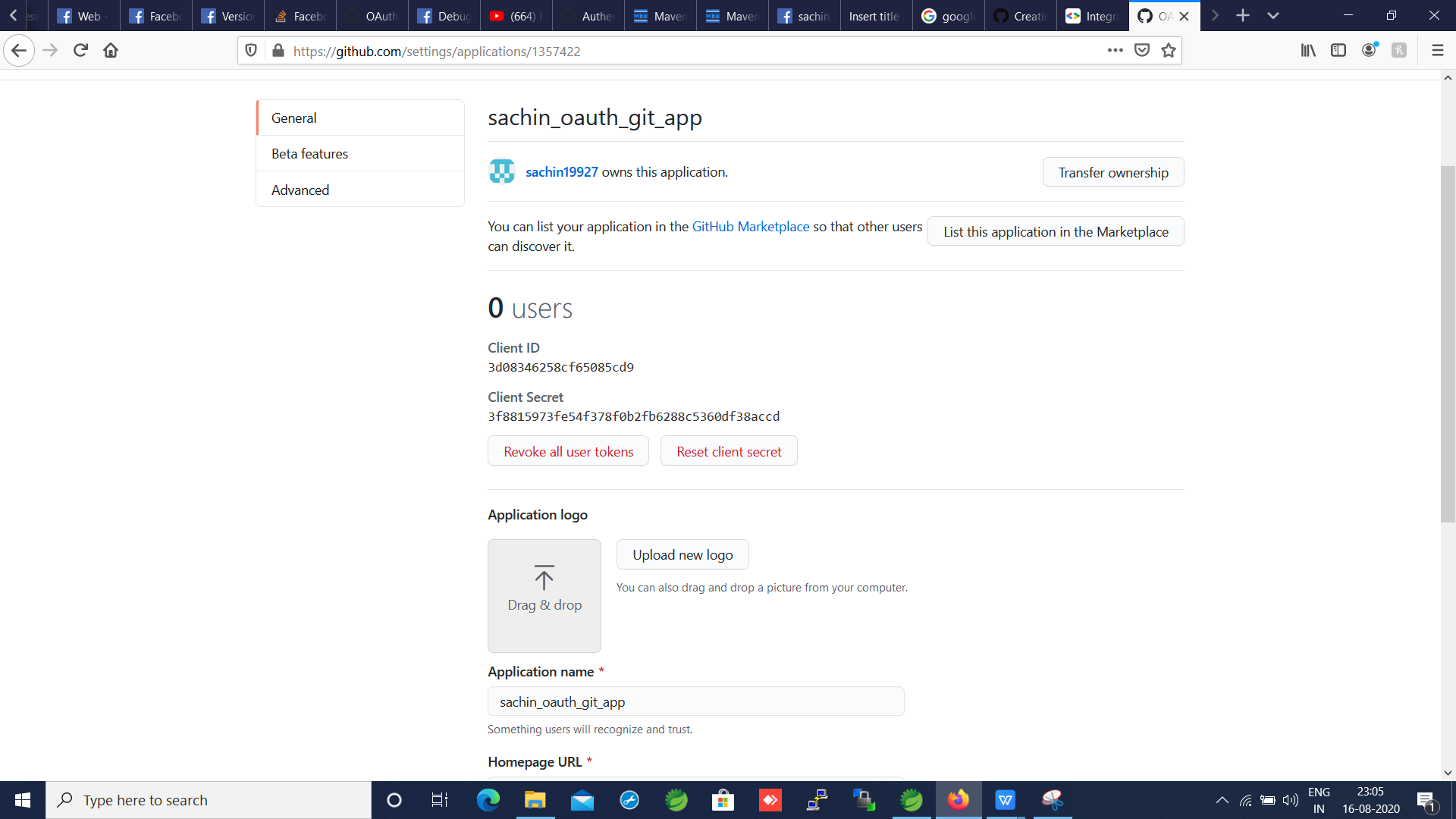
Its logged in

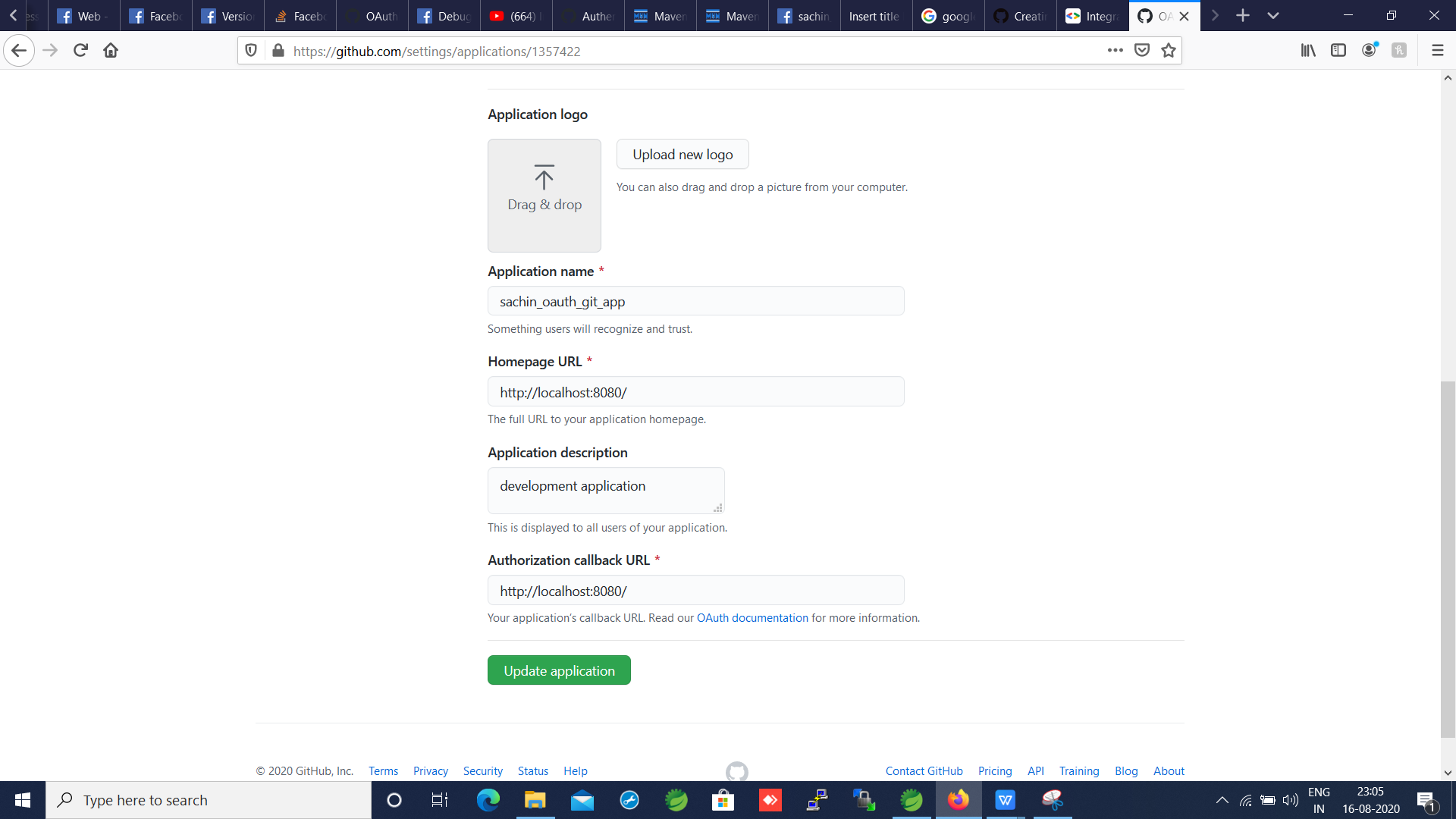
Ref : <https://developers.facebook.com/docs/>

Now try to do with Git

Go to Git account => setting => Developer Setting= Oauth apps

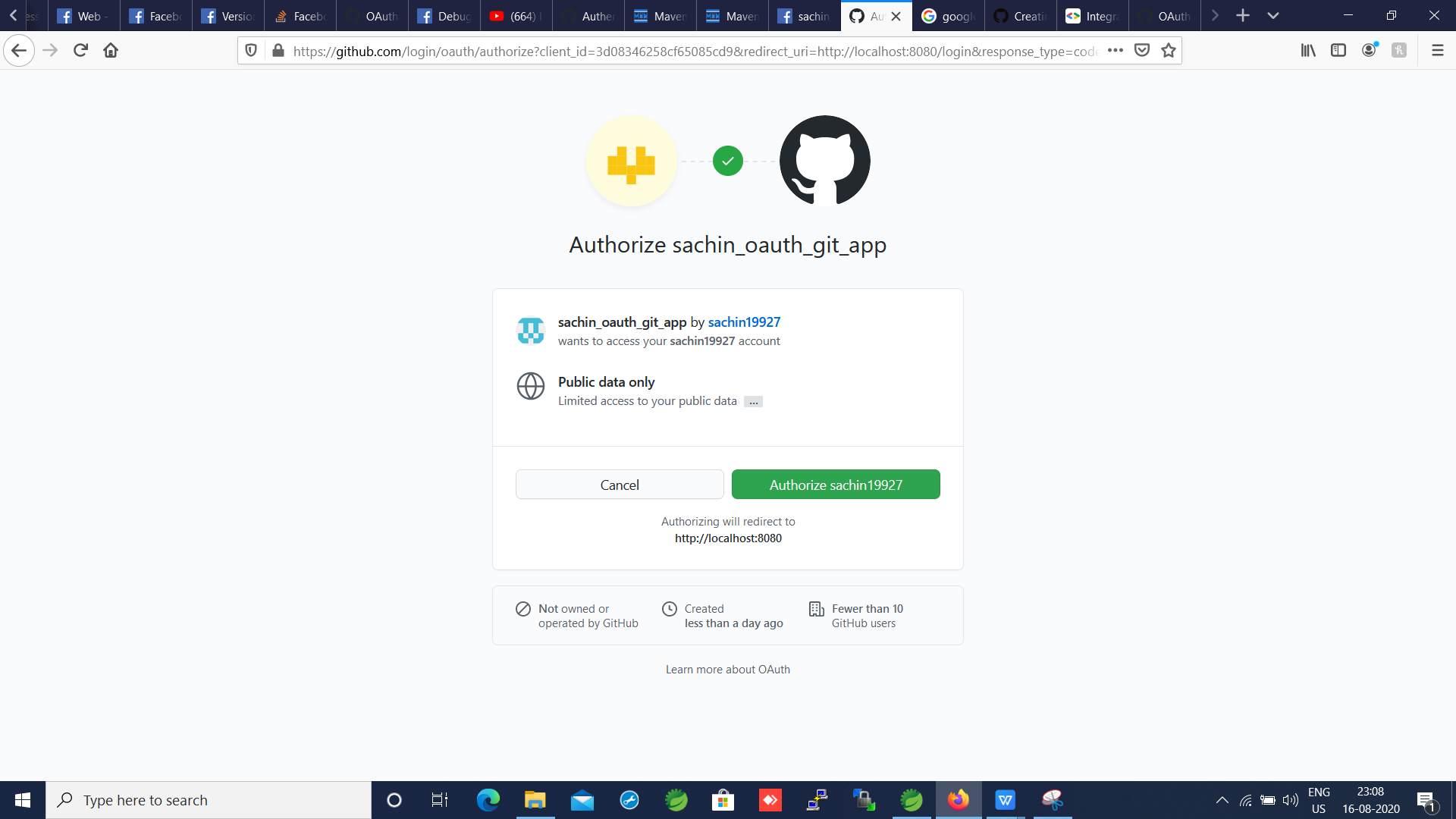






Start app

localhost:8080



Always go to docs and get latest URI’s for auth config

Ref : <https://docs.github.com/en/rest/reference/activity#events>

Spring Boot -Spring Security -login with Google from scratch

N

Save

Then create Credentials with Client name

Create new project Spring\_Boot\_oauth\_google adding following dependency

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-oauth2-client</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-security</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

@Configuration

public class SecurityConfig extends WebSecurityConfigurerAdapter {

/\*\*

\* define which endpoints require authentication whcih dnt

\*

\* antMatcher("/\*\*").authorizeRequests() => match our all end points and

\* authorize all requests

\*

\* antMatchers("/").permitAll() => endpoints which user dont need to be logged

\* in

\*

\* .anyRequest().authenticated() => other request need to be authenticated

\*

\* and().oauth2Login() and login will based on oauth2login type redirect to

\* oauth2 base login

\*

\*

\*/

@Override

protected void configure(HttpSecurity http) throws Exception {

http.antMatcher("/\*\*").authorizeRequests().antMatchers("/").permitAll().anyRequest().authenticated().and()

.oauth2Login();

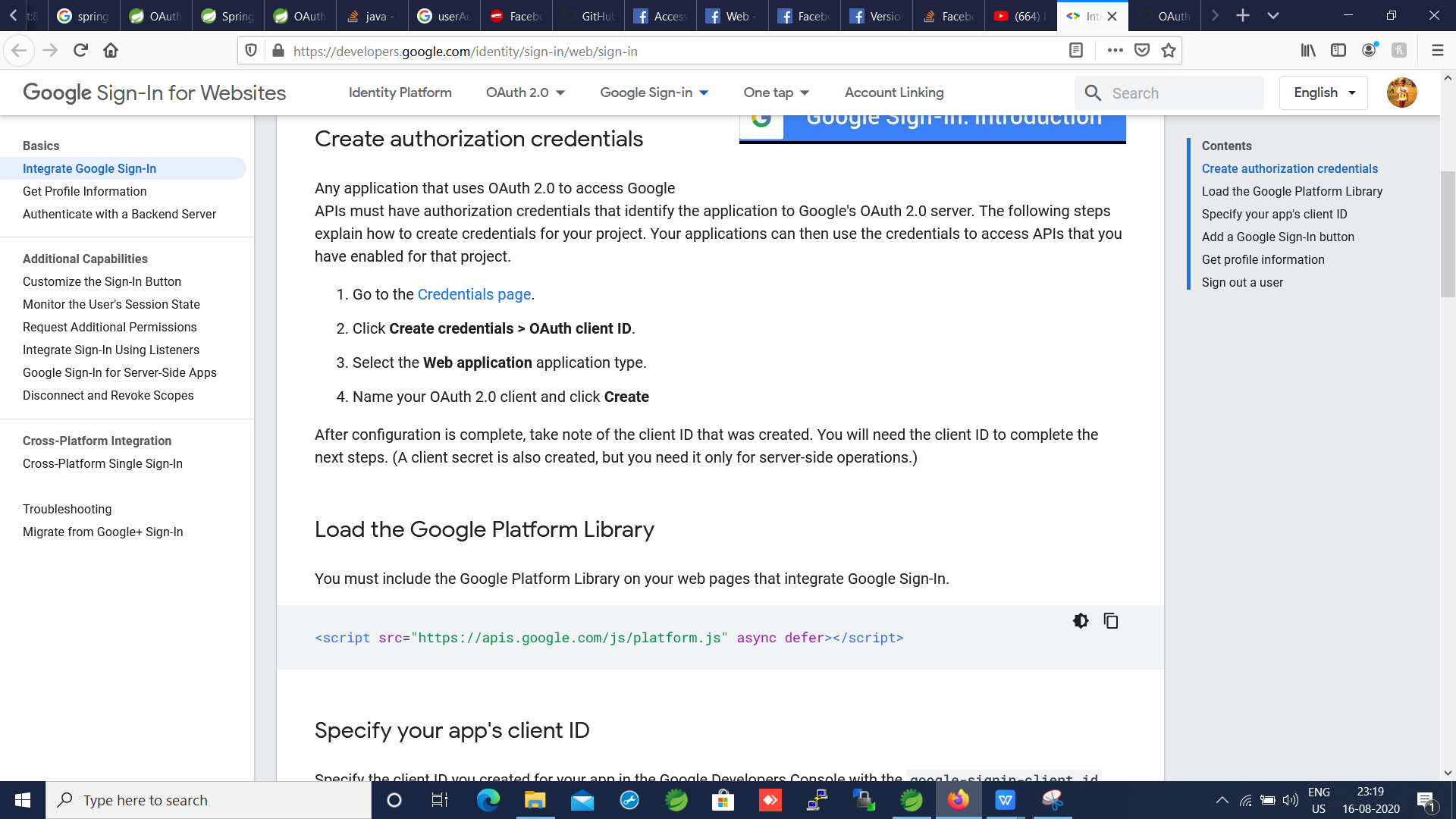
}

}

Google Oauth

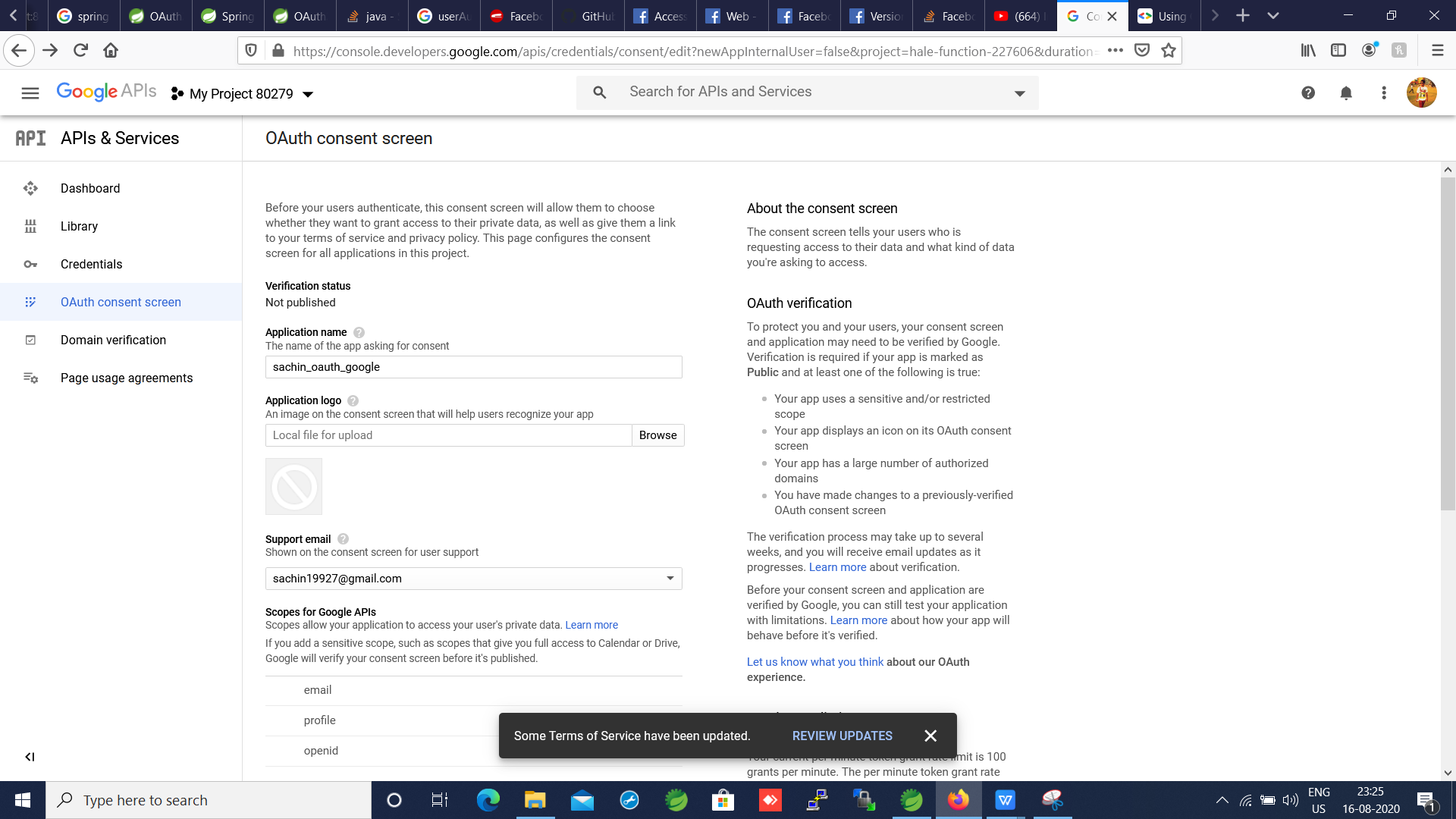
<https://console.developers.google.com/apis>

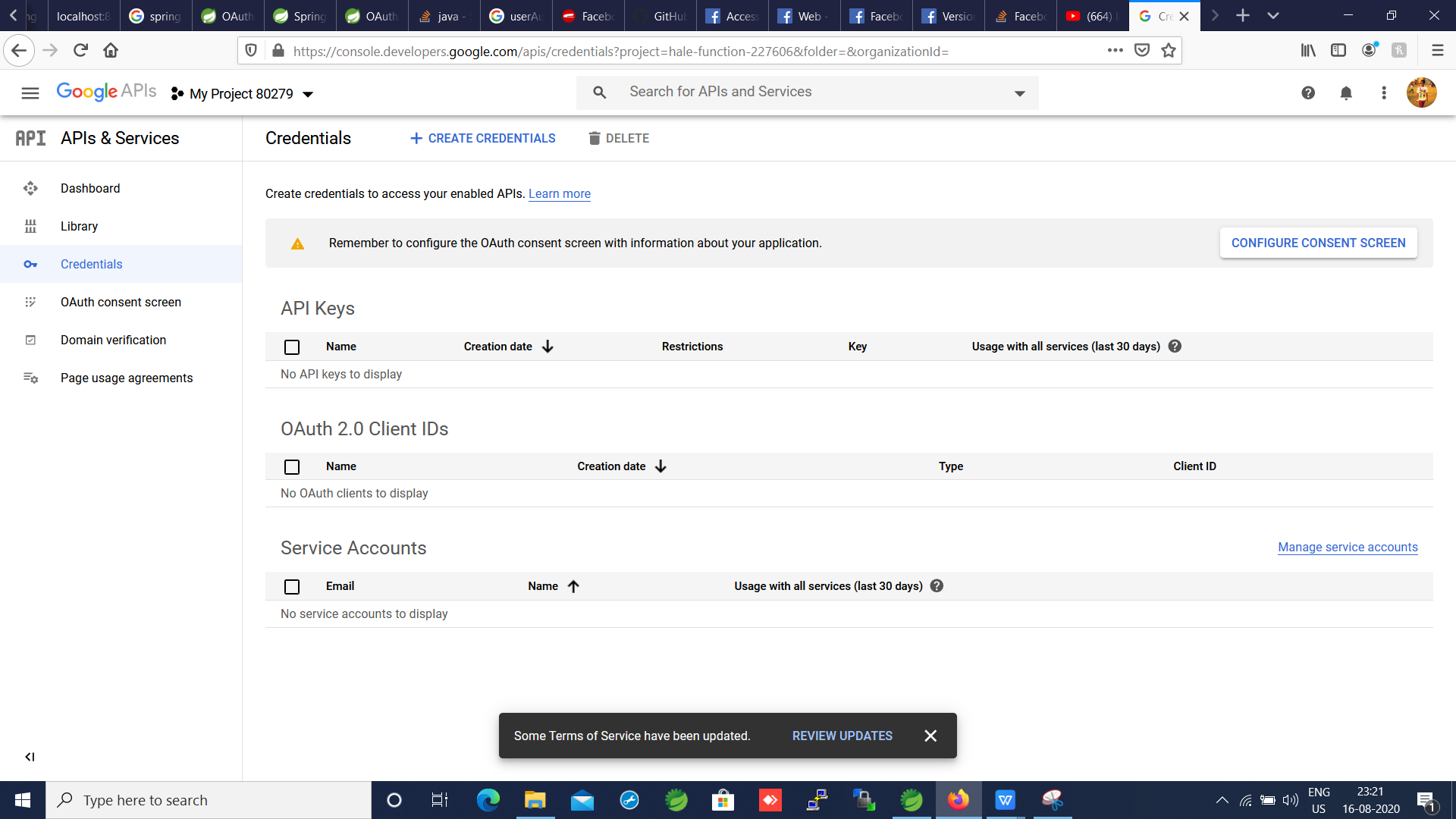
Create Credentials



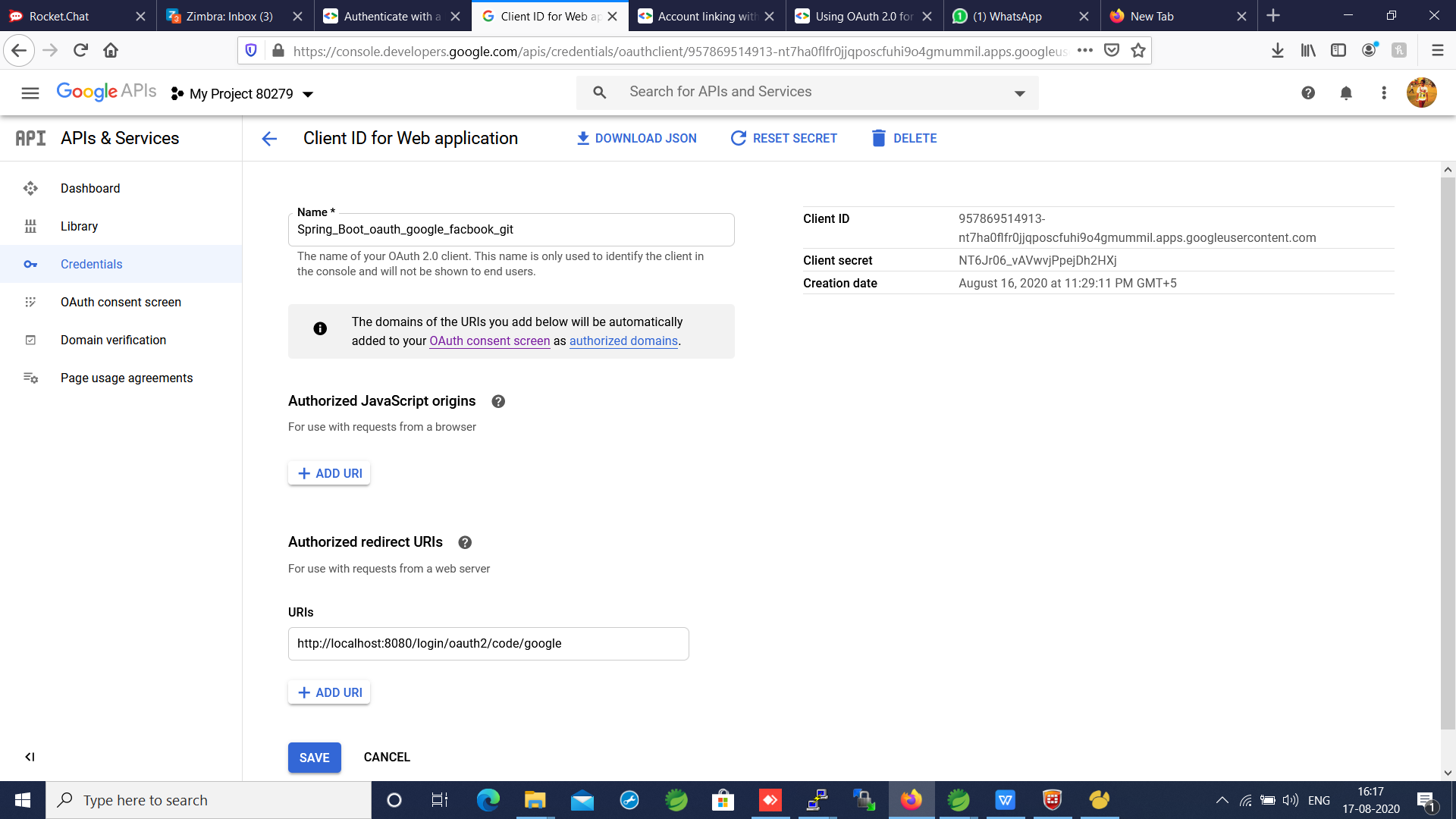
Save

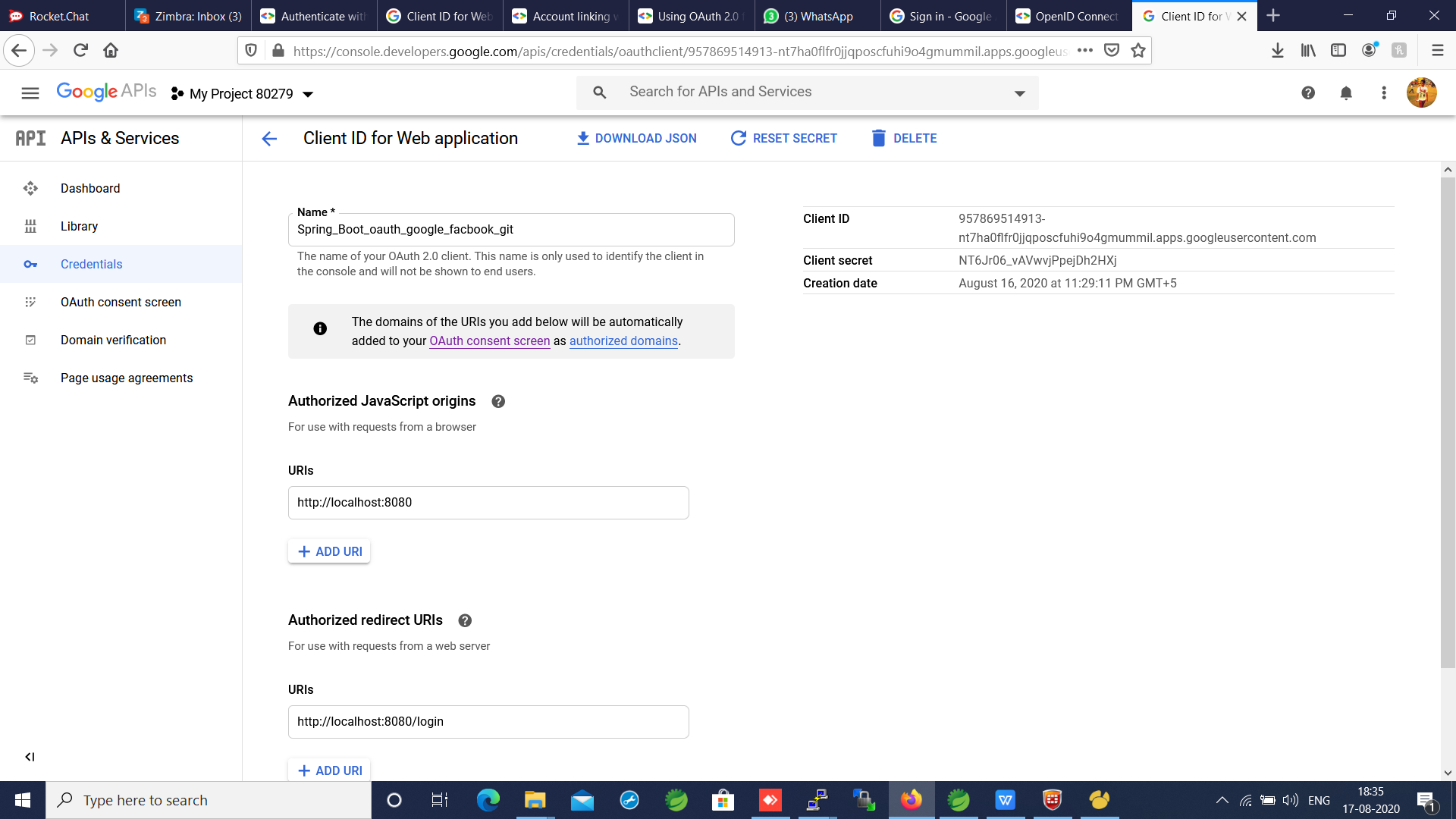
Then create Credentials with Client name

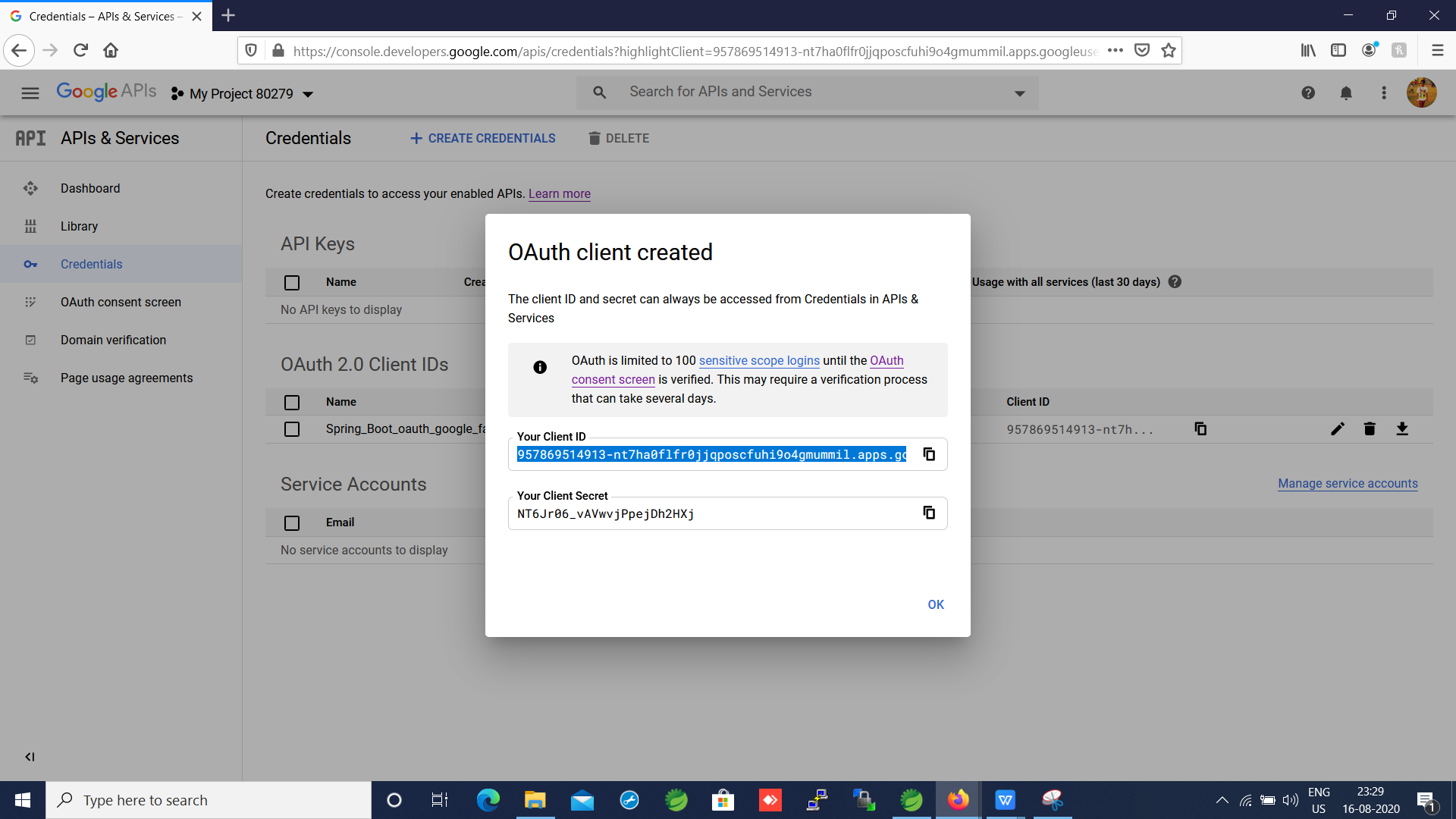




N

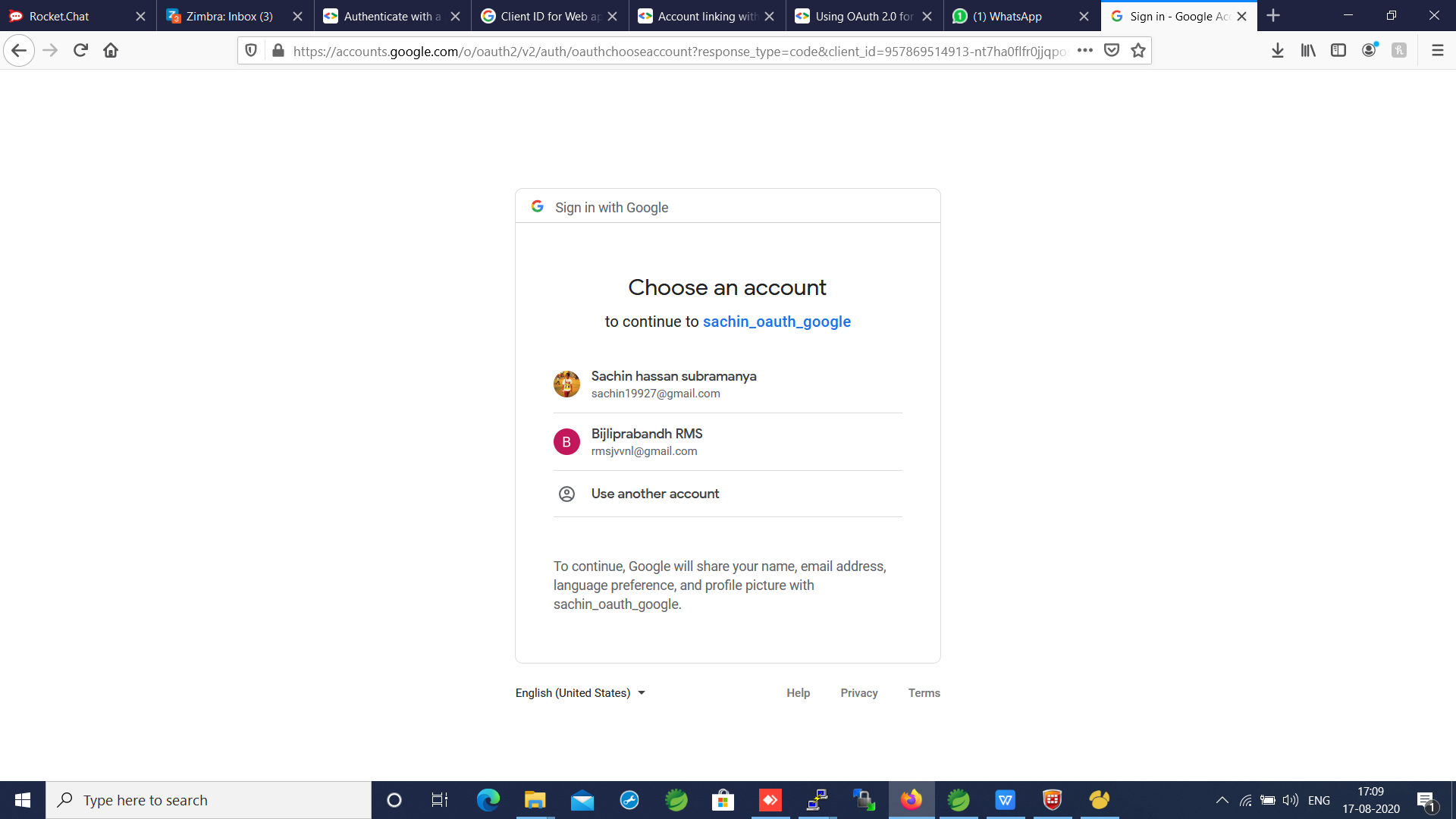


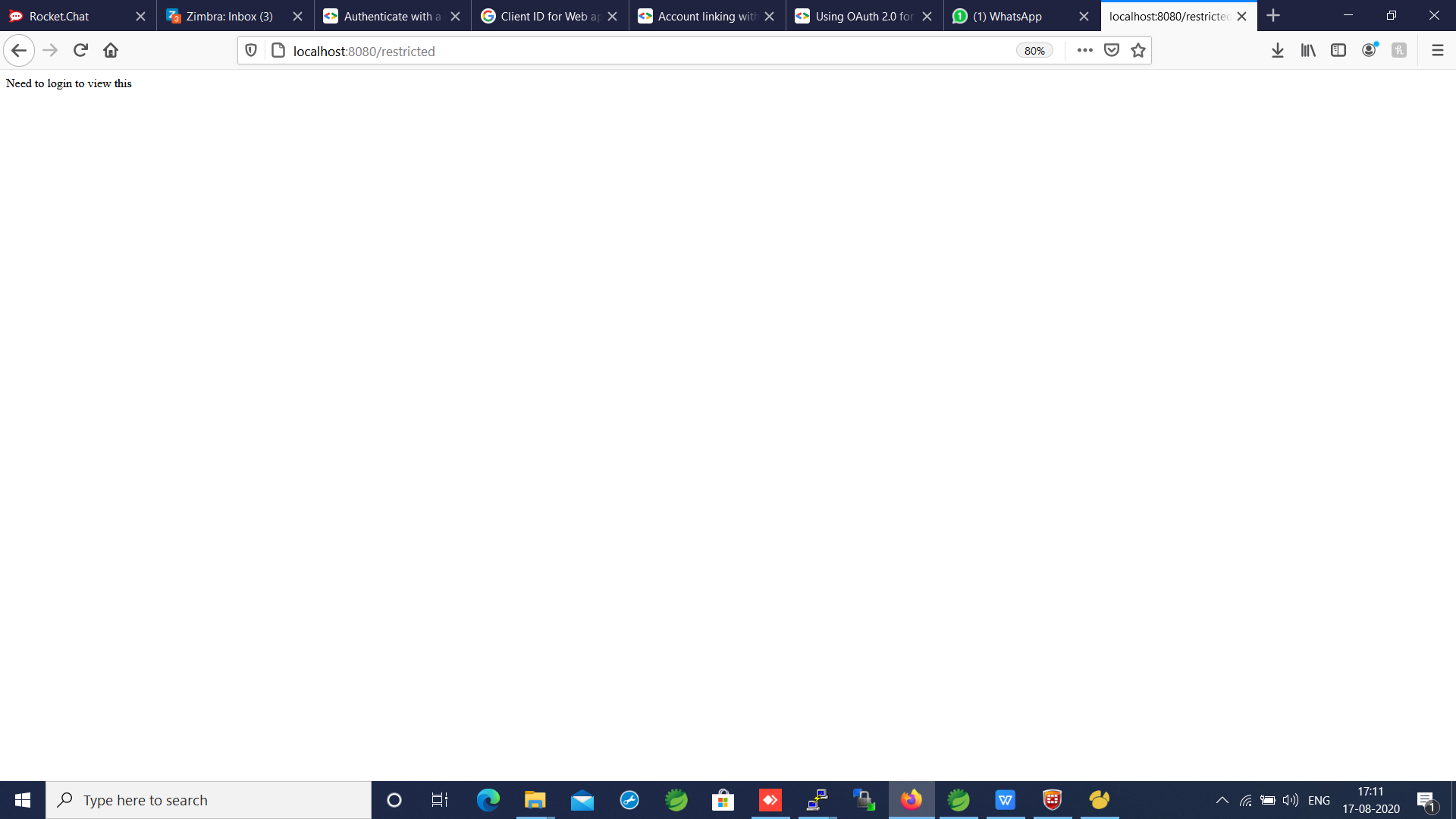




<http://localhost:8080/> => will be accessed with out any login

<http://localhost:8080/>restricted => will be accessed with google login





Disadvantage

-since its an oauth there will be authorization and access token so some body login to google and confident about security

- if the application I am accessing I am not confident about then user good reason to provide access to google if access token is lost there is always posibility that some one can use it

- when oauth exchange happen thng that validating if the user has logged into google that point of time when u do sign in that time how ever if applciation is holding on to that token probly login user from week now but not valid may that user logged from google or facebook r deleted . so if application using aouth tokens to authenticate then it has to do every time make sure not realy on authentcation done a while back.