**What is OAuth2?**

“This is an authorization framework that enables applications to obtain limited access to user accounts on an HTTP service. It works by delegating user authorization to access the user account. This authorization flows for we and desktop applications and mobile devices” [1].

**What are OAuth Roles?**

* **Resource Owner: User**

An entity capable of granting access to a protected resource

* **Resource Server: API**

The server hosting the protected resources, capable of accepting and responding to protected resource requests using access tokens.

* **Client: Application**

An application making protected resource requests on behalf of the resource owner

* **Authorization Server**

The server issuing access tokens to the client after successfully authenticating the resource owner and the obtaining authorization [1].

**What is the OAuth Protocol Flow?**

Below you can see a detailed flow of how our app works

**Google Drive**

**OAuth Service**

**User**



Sign in (Request)



[This Photo](http://hul78.deviantart.com/art/Isometric-Walk-Cycle-473946539) by Unknown Author is licensed under [CC BY-NC-ND](https://creativecommons.org/licenses/by-nc-nd/3.0/)

Choose account (Response)



Choose gmail account(Request)

Choose gmail account (Request)



[This Photo](http://hul78.deviantart.com/art/Isometric-Walk-Cycle-473946539) by Unknown Author is licensed under [CC BY-NC-ND](https://creativecommons.org/licenses/by-nc-nd/3.0/)



[This Photo](https://commons.wikimedia.org/wiki/File:Google_Drive_Ana.jpg) by Unknown Author is licensed under [CC BY-SA](https://creativecommons.org/licenses/by-sa/3.0/)

Confirm credentials (Response)

Confirm credentials (Response)



Click confirm (Request)



[This Photo](http://hul78.deviantart.com/art/Isometric-Walk-Cycle-473946539) by Unknown Author is licensed under [CC BY-NC-ND](https://creativecommons.org/licenses/by-nc-nd/3.0/)



Upload files (Request)

User interface (Response)



[This Photo](http://hul78.deviantart.com/art/Isometric-Walk-Cycle-473946539) by Unknown Author is licensed under [CC BY-NC-ND](https://creativecommons.org/licenses/by-nc-nd/3.0/)



Upload files (Request)



[This Photo](https://commons.wikimedia.org/wiki/File:Google_Drive_Ana.jpg) by Unknown Author is licensed under [CC BY-SA](https://creativecommons.org/licenses/by-sa/3.0/)

The user will run the app and click sign in.

The user then can select the account from the suggested accounts which he needs to sign in to the app.

The OAuth2 will suggest to confirm the credentials again.

After confirming the credentials, the page will be directed to the main page where you can upload the files.

User can click upload files and can upload several files from the explorer and click upload.

The uploaded files can be views in the Google Drive.

**What are the grant types?**

The grant types depend on the method used by the application to request authorization, and the grant types supported by the API. There are four grant types which are listed below, and which can be used in different cases.

* **Authorization** **Code:** used with server-side Applications
* **Implicit:** used with mobile apps or Web Applications
* **Resource** **Owner Password Credentials:** used with trusted applications
* **Client Credentials:** used with applications API access [1]

**Explanation of the app implementation.**

You should be having a google account where you can register the app with google which is the service provider.

Steps to follow

**Step1: Create Project**

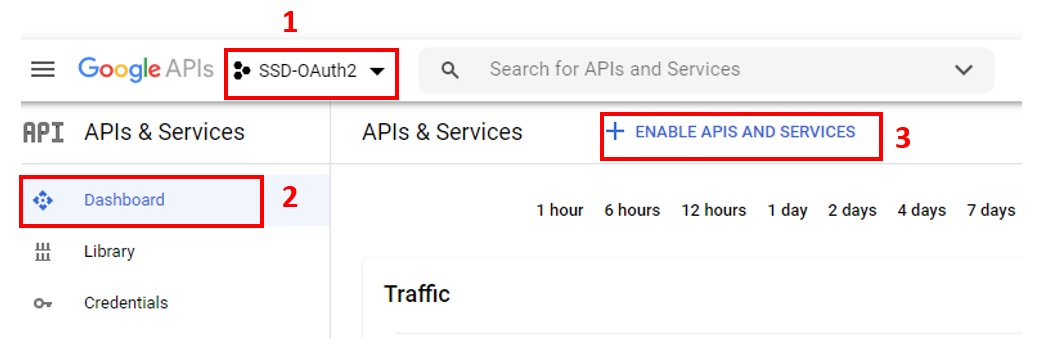
Login to google developer console : <https://console.developers.google.com/>

Click new project and create a new project as shown in below figure.

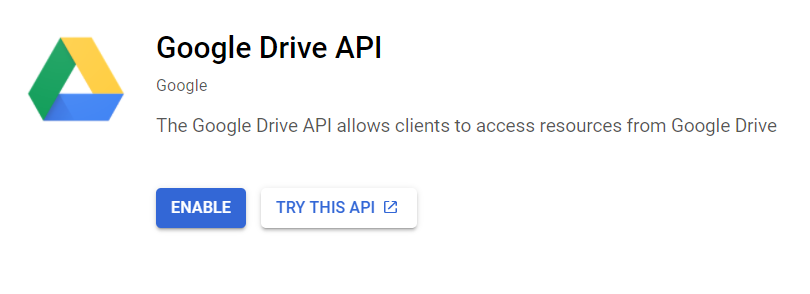


**Step 2: Enable google drive API**

After selecting the newly created project in dashboard and click ‘ENABLE APIS AND SERVICES’.



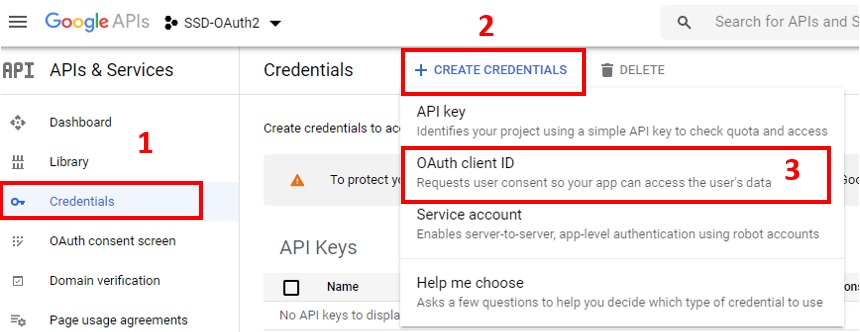
Search for Google Drive API and click ENABLE.



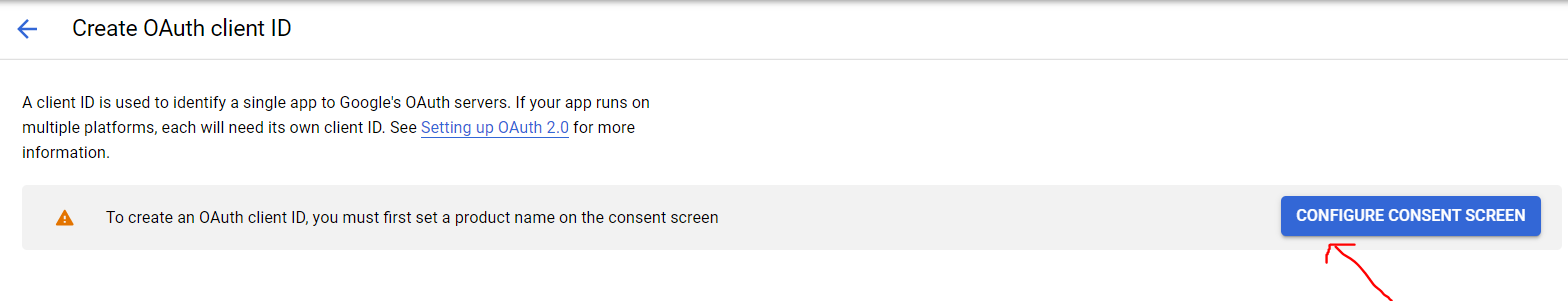
**Step 3: Create Credentials**

This is to configure Credential for the project in the cloud developer platform console.

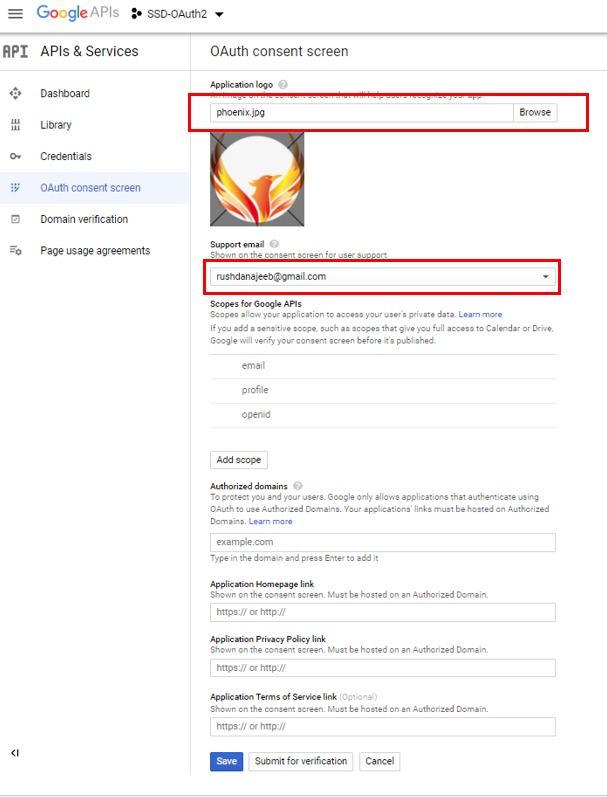
Click Credentials in the navigation panel and click create credentials and select OAuth client ID.



Onclick a warning appears to configure the oauth consent screen as shown below.

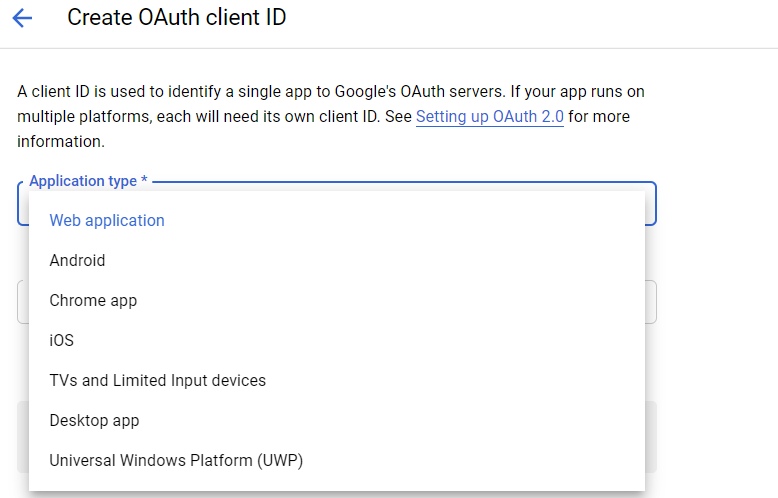


**Step 4: Configuring the OAuth consent Screen**

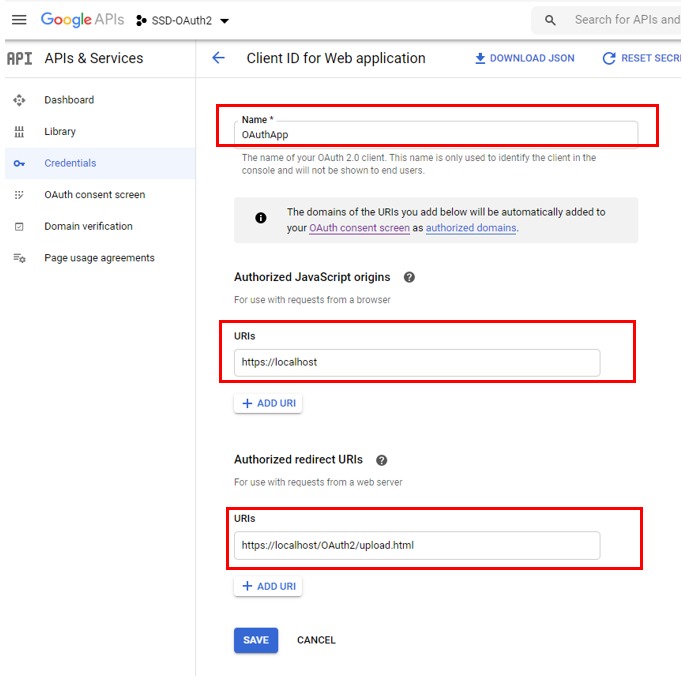
Fill in the required details as below and click save. Required: Application name, support email

**Step 4: Create Credentials – continue**

Navigate back to credential menu and click OAuth client ID as shown in step 3. Then proceed by selecting the application type as web application.



Next fill in the Required details – name of the OAuth client, authorized javascript origin URI, and the authorized redirect URI as shown below and click save.

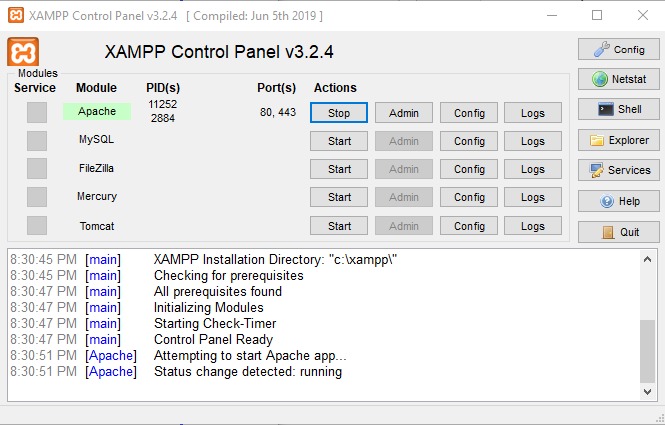


After saving the client ID and the client secret credentials will be provided as shown below.

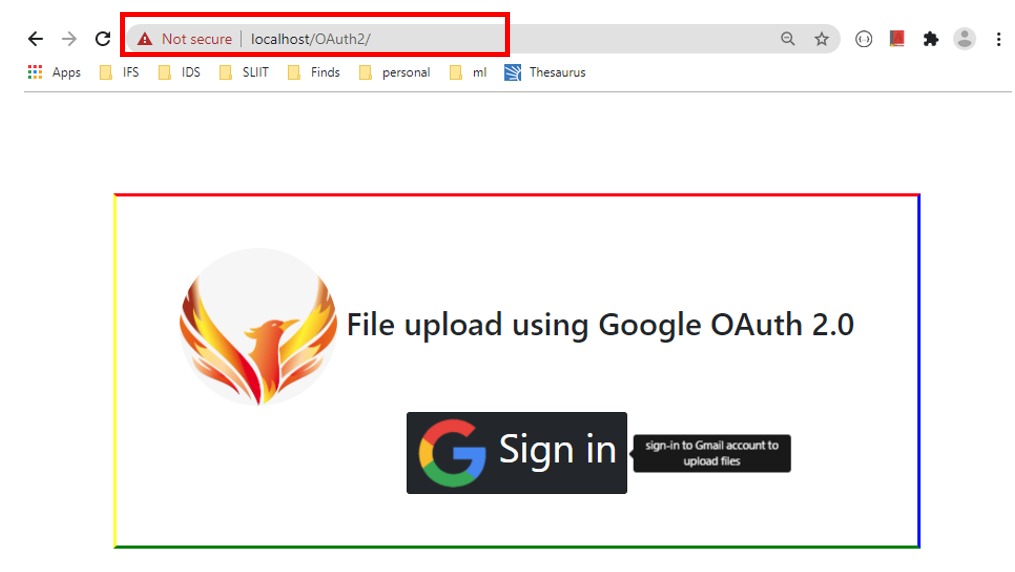


**How our project works**

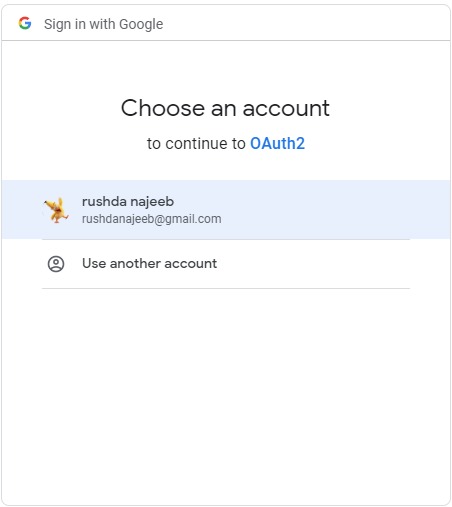
An Apache server was used as shown below to run the application.



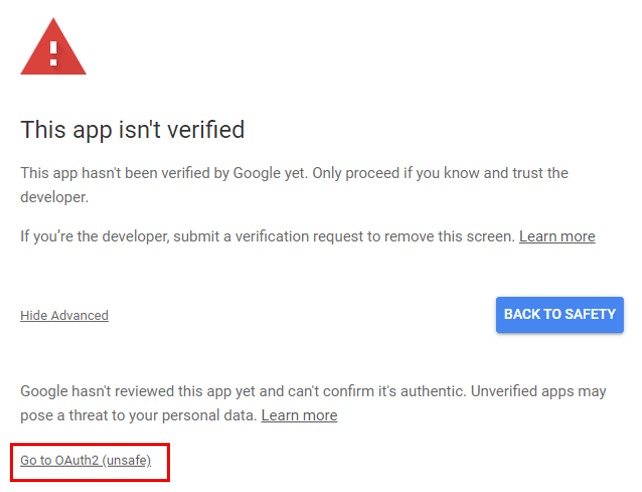
After running the application, the home page can be accessed by <https://localhost/OAuth2/> . A snap of the home page is shown below.



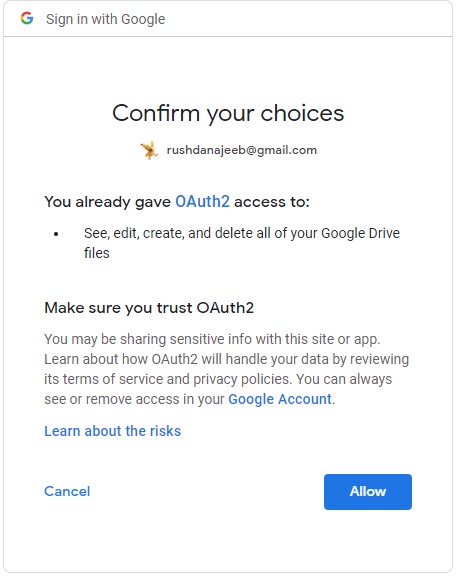
Click sign in and select your preferred account



You’ll be getting a warning stating app isn’t verified. To continue click go to OAuth2(unsafe) and you’ll be directed to the page to confirm permission.

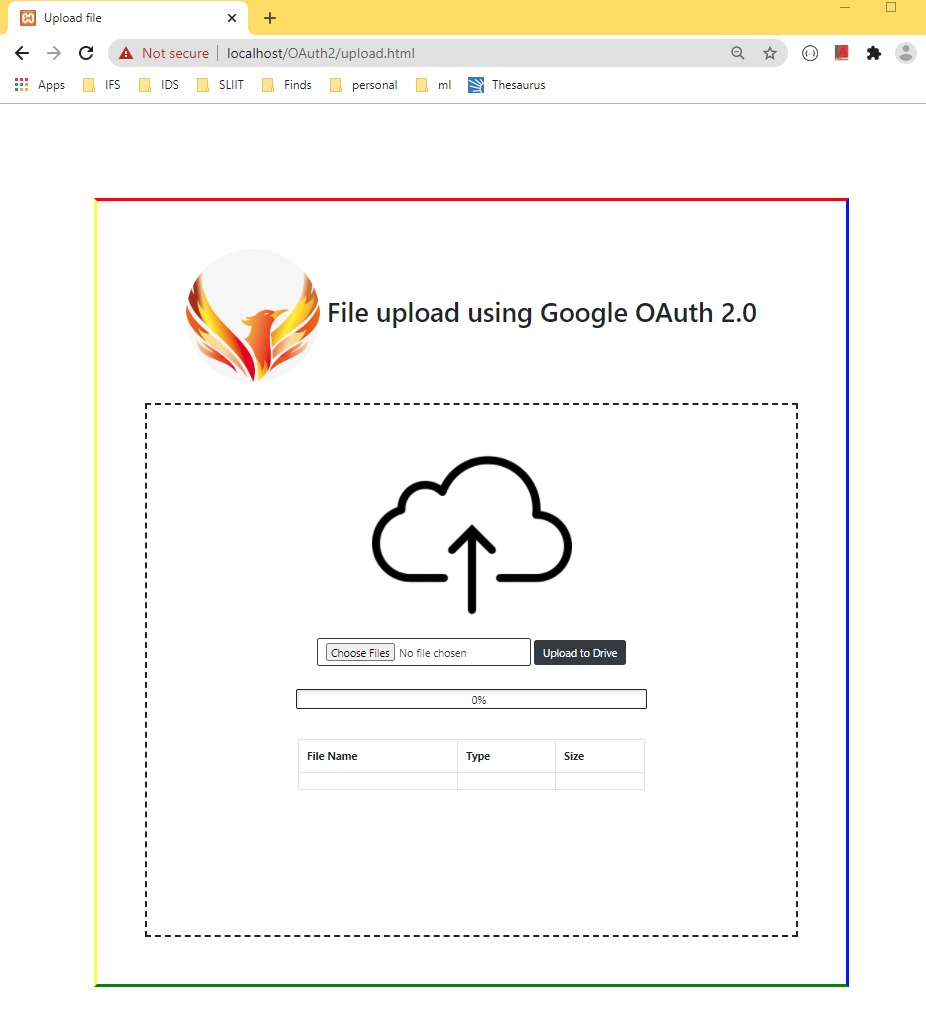


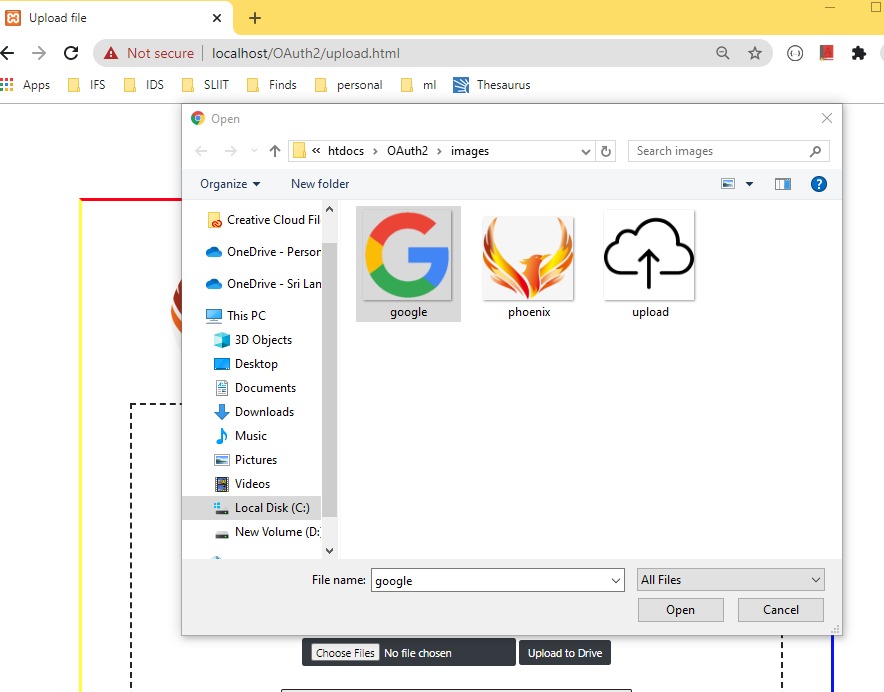
1

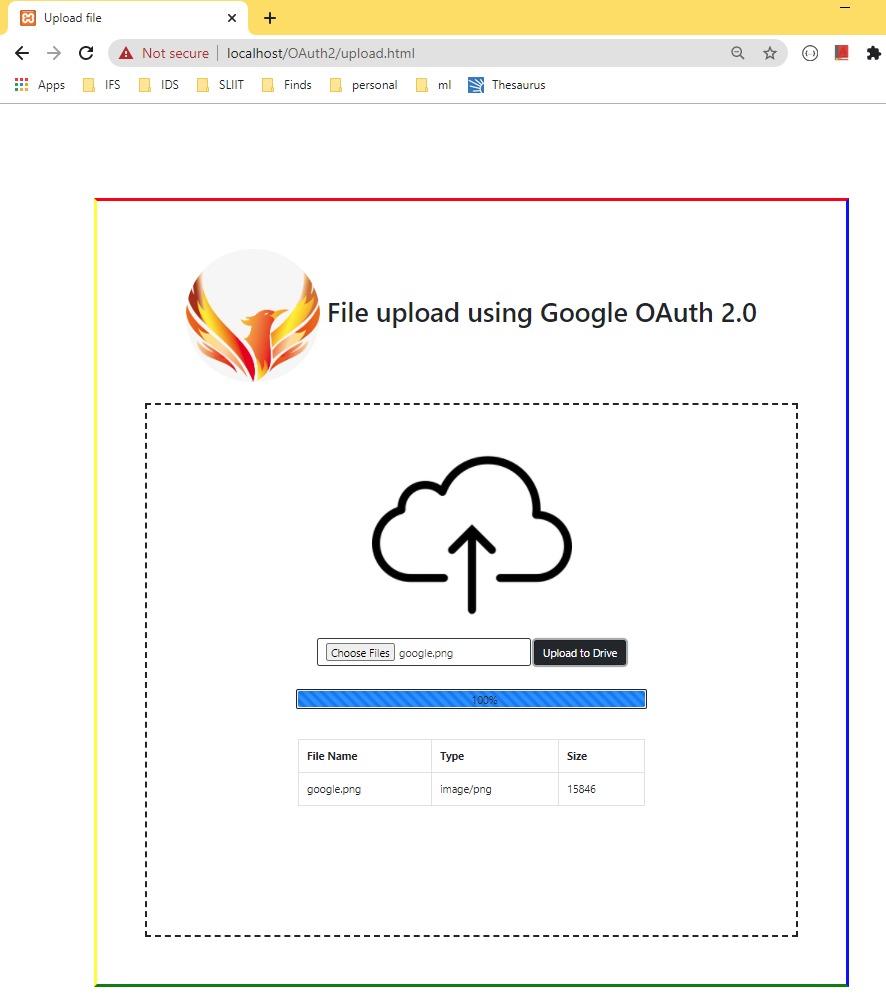


2

After confirming permissions you’ll be redirected to the upload page: <https://localhost/OAuth2/upload.html> . (configured in step 4)



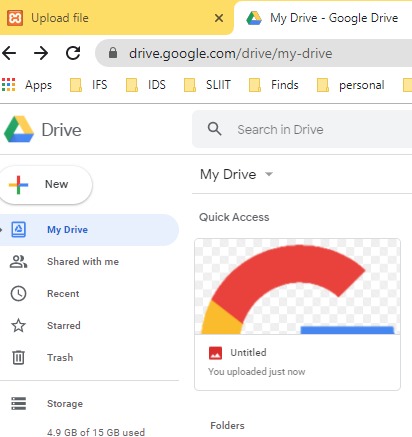
To upload a file / many files to the drive click choose files. Select the uploading image or file and click open.

The selected file and it will be visible in the table with information containing filename, file type and file size.

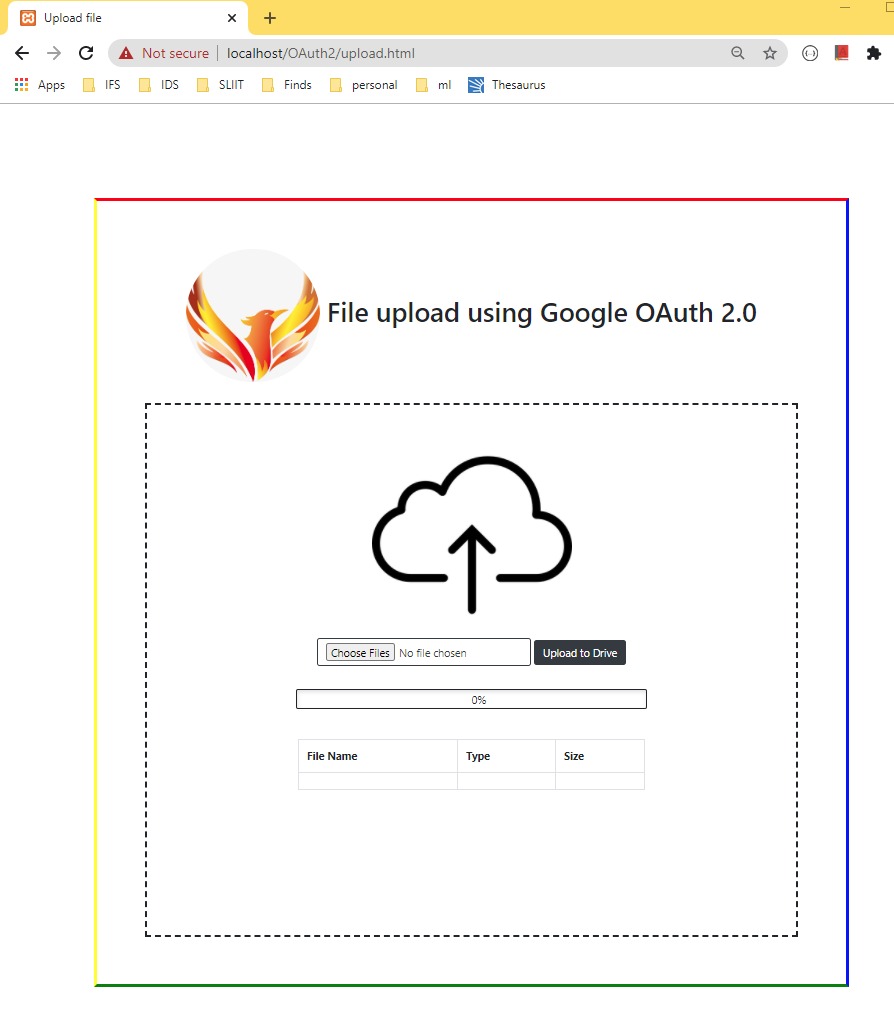
Then click upload to drive. An alert notification pops up when uploading is completed.



You can see the uploaded file in the drive as shown below.



After a successful file upload to the drive the page auto re-freshens back to the upload file page.



# References

|  |  |
| --- | --- |
| [1] | M. Anicas, "Community," DigitalOcean, 2020. [Online]. Available: https://www.digitalocean.com/community/tutorials/an-introduction-to-oauth-2. [Accessed 4 10 2020]. |