



Sri Lanka Institute of Information Technology

OAuth related assignment document.

Module Name: SSD.

SSD Assignment 2

Group Members Details

Registration No	Name
IT17166720	WICKRAMAARACHCHI W.P.C.K
IT17171298	KATHRIARACHCHI A.D

SSD Assignment 2 OAuth2

This application is about share google sheets with outside of organization and access sheets using a local application. You must create google oauth2 credentials and service account. The process as follow:

The left screenshot shows the 'Client ID for Web application' configuration page with the 'Authorized JavaScript origins' section. It includes a note: 'The domains of the URIs you add below will be automatically added to your OAuth consent screen as authorized domains.' Below this, there are two input fields for URIs: 'http://localhost:5000' and 'http://127.0.0.1:5000'. A '+ ADD URI' button is at the bottom.

The right screenshot shows the same page but with the 'Authorized redirect URIs' section expanded. It includes a note: 'For use with requests from a web server'. Below this, there are six input fields for URIs: 'http://localhost:5000/login', 'http://localhost:5000/authorize', 'http://127.0.0.1:5000/authorize', 'http://localhost:5000/request-api', 'http://127.0.0.1:5000/request-api', and 'http://localhost:5000/sheetauth'. A '+ ADD URI' button is at the bottom. At the very bottom of the right screenshot, there are 'SAVE' and 'CANCEL' buttons.



Then copy client id and secret id and add it creating .env file.


Create a google sheet, copy the sheet name, and add it to .env file as SHEET_NAME. Then .env file will be as follow:

```
.env
1 client_id=
2 client_secret=
3 SAMPLE_SPREADSHEET_ID=
4 SHEET_NAME='user'
```


Add client email in client.json to share access of google sheet.

id	first_name	last_name	email	gender	ip_address
1	Katherina	Hulk	khulk0@simpl	Female	132.49.127.173
2	Courtney	Betke	cbetke1@para	Female	113.138.108.61
3	Alex	Prandin	aprandin2@ac	Male	45.205.237.255
4	Kory	Ausher	kausher3@ft.c	Male	222.71.247.164
5	Ieci	Earnworth	ifarnworth4@i	Female	160.37.749.113

 Share with people and groups 




Owner

 sheet-426

Editor ▾

[Send feedback to Google](#) [Done](#)

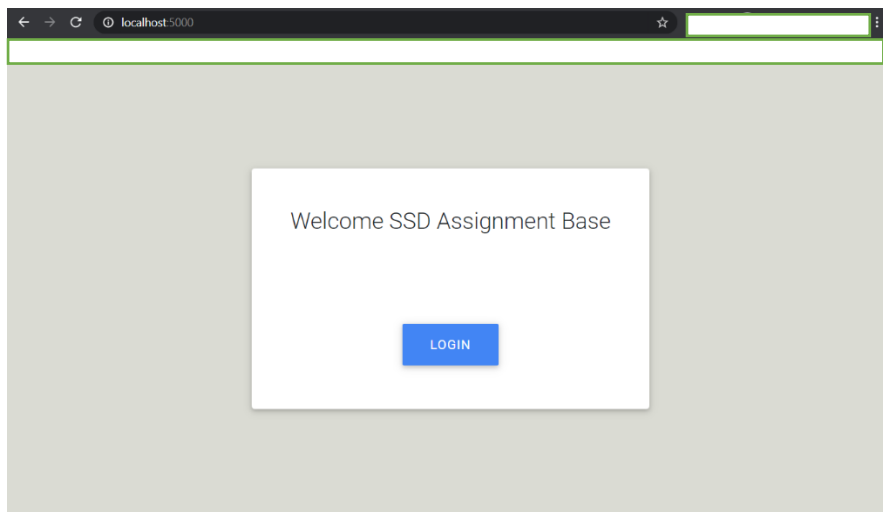
 Get link

Restricted Only people added can open with this link

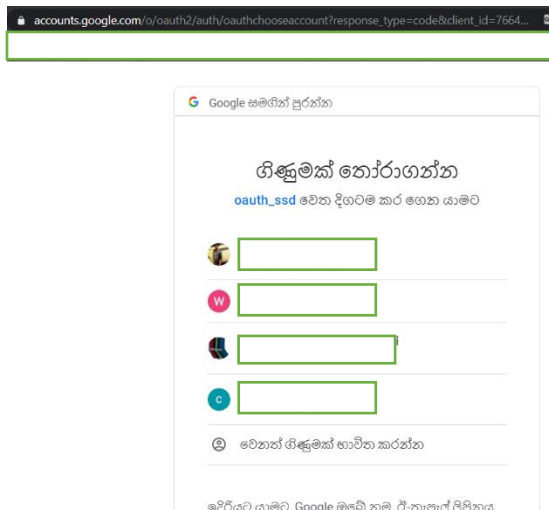
[Change to anyone with the link](#) [Copy link](#)

```
{ client.json > client_email
1 {
2   "type": "service_account",
3   "project_id": "[REDACTED]",
4   "private_key_id": "[REDACTED]",
5   "private_key": "[REDACTED]",
6   "client_email": "[REDACTED]",
7   "client_id": "[REDACTED]",
8   "auth_uri": "https://accounts.google.com/o/oauth2/auth",
9   "token_uri": "https://oauth2.googleapis.com/token",
10  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1",
11  "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata"
12 }
13
```

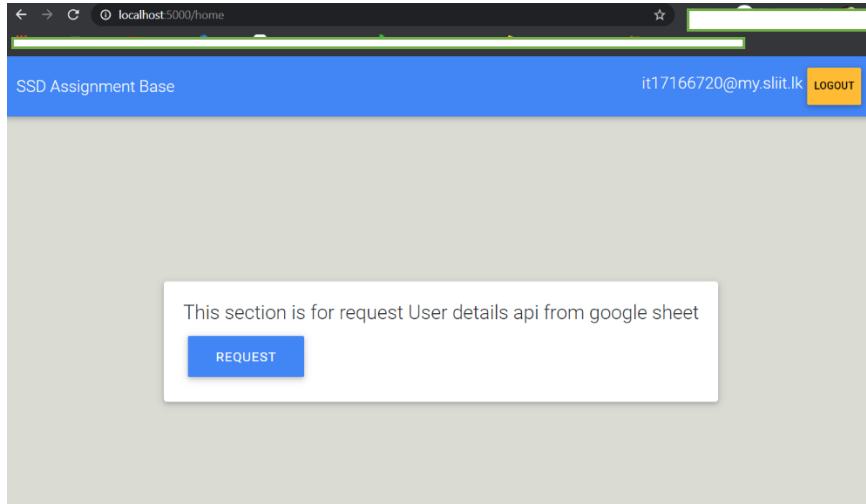
In the application you can simply run the development mode in locally using 'flask run' command and open the application in <http://127.0.0.1:5000/> . Then you will display the welcome page as follow:



Then you have login to the system, and you will redirect to google authentication page as follow:

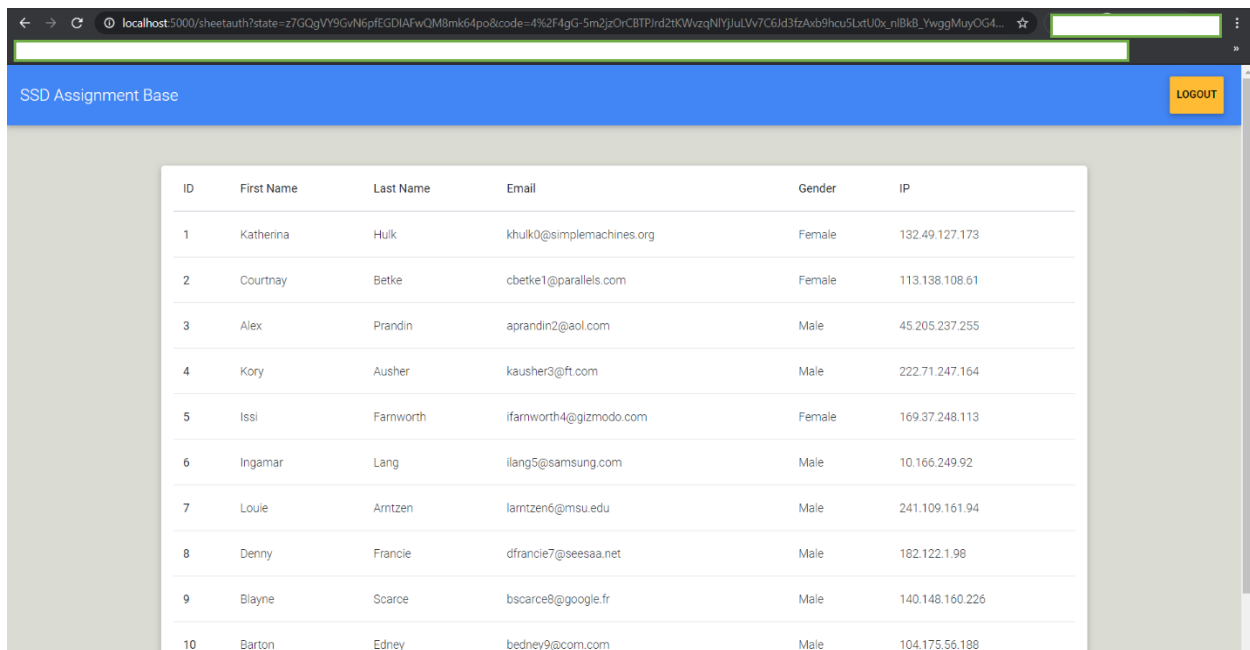


Then you will redirect to home page after authenticated successfully.



Then you need to request for google sheet. In this phase this application is only developed for access a one sheet. Further this application can be developed to access multiple sheets which can be edit or with limited access. For this step we used google service account and get an email which is used for giving share access via email.

Finally, you will be display sheet data in local application using data tables representation.



ID	First Name	Last Name	Email	Gender	IP
1	Katherina	Hulk	khulk0@simplemachines.org	Female	132.49.127.173
2	Courtney	Betke	cbetke1@parallels.com	Female	113.138.108.61
3	Alex	Prandin	aprandin2@aol.com	Male	45.205.237.255
4	Kory	Ausher	kausher3@ft.com	Male	222.71.247.164
5	Issi	Farnworth	ifarnworth4@gizmodo.com	Female	169.37.248.113
6	Ingamar	Lang	ilang5@samsung.com	Male	10.166.249.92
7	Louie	Arntzen	larntzen6@msu.edu	Male	241.109.161.94
8	Denny	Francie	dfrancie7@seesaa.net	Male	182.122.1.98
9	Blayne	Scarce	bscarce8@google.fr	Male	140.148.160.226
10	Barton	Edney	bedney9@com.com	Male	104.175.56.188

We have presented our workflow using chart as follow:

