OAuth 2.0 Authorization Framework
Software Security – Assignment 2

Ishadi Malshani Gulawita
Faculty of Computing, Sri Lanka Institute of Information Technology,
New Kandy Rd, Malabe 10115, Sri Lanka
ishadimg@outlook.com, ms20907266@my.sliit.lk

Introduction

OAuth is an authorization method to provide access to resources over the HTTP protocol. It can be used for authorization of various applications as well as for the manual user access. The general way it works is allowing an application to have an access token (which represents a user’s permission for the client to access their data) which it can use to authenticate a request to an API endpoint. There are two versions of OAuth authorization OAuth 1 (using HMAC-SHA signature strings) and OAuth 2 (using tokens over HTTPS).

In any OAuth 2.0 flow we can find the following roles:

1. **The Third-Party Application: "Client"**
   The client is the application that is trying to get access to the user's account. It needs to get permission from the user before it can do so.
2. **The API: "Resource Server"**
   The resource server is the API server used to access the user's data.
3. **The Authorization Server**
   This is the server that presents the interface where the user approves or denies the request. In smaller implementations, this may be the same server as the API server, but larger scale deployments will often build this as a separate component.
4. **The User: "Resource Owner"**
   The resource owner is the person who is giving access to some portion of their account.

The OAuth 2.0 Authorization Framework specification defines four flows to get an Access Token. These flows are called grant types. Deciding which one is suited for the case depends mostly on the type of the application. The first one is Authorization Code. It is used by Web Apps executing on a server. This is also used by mobile apps, using the Proof Key for Code Exchange (PKCE) technique. The second on is Implicit. It is used by JavaScript-centric apps (Single-Page Applications) executing on the user's browser. Resource Owner Password Credentials is the third one and it is used by trusted apps. Finally, Client Credentials: used for machine-to-machine communication. Also, when considering OAuth 2.0 framework, it is very important to understand **Authorization Code Flow**.

Web Application/

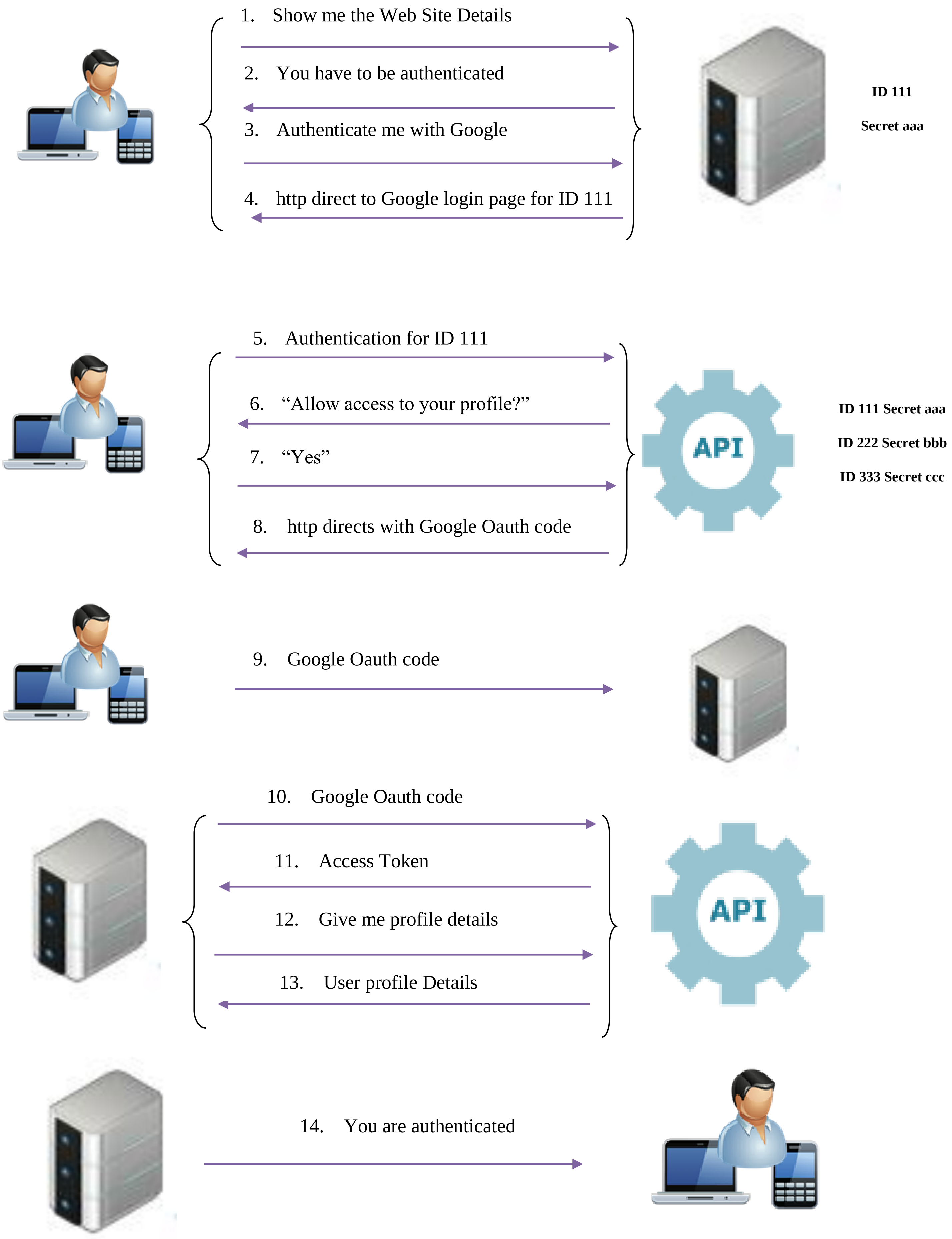
Client Application

Authorization Server

Resource Server

User

Resource Owner



1. Show me the Web Site Details
2. You have to be authenticated
3. Authenticate me with Google
4. http direct to Google login page for ID 111

ID 111

Secret aaa

5. Authentication for ID 111
6. “Allow access to your profile?”
7. “Yes”
8. http directs with Google Oauth code

ID 111 Secret aaa

ID 222 Secret bbb

ID 333 Secret ccc

9. Google Oauth code

10. Google Oauth code
11. Access Token
12. Give me profile details
13. User profile Details

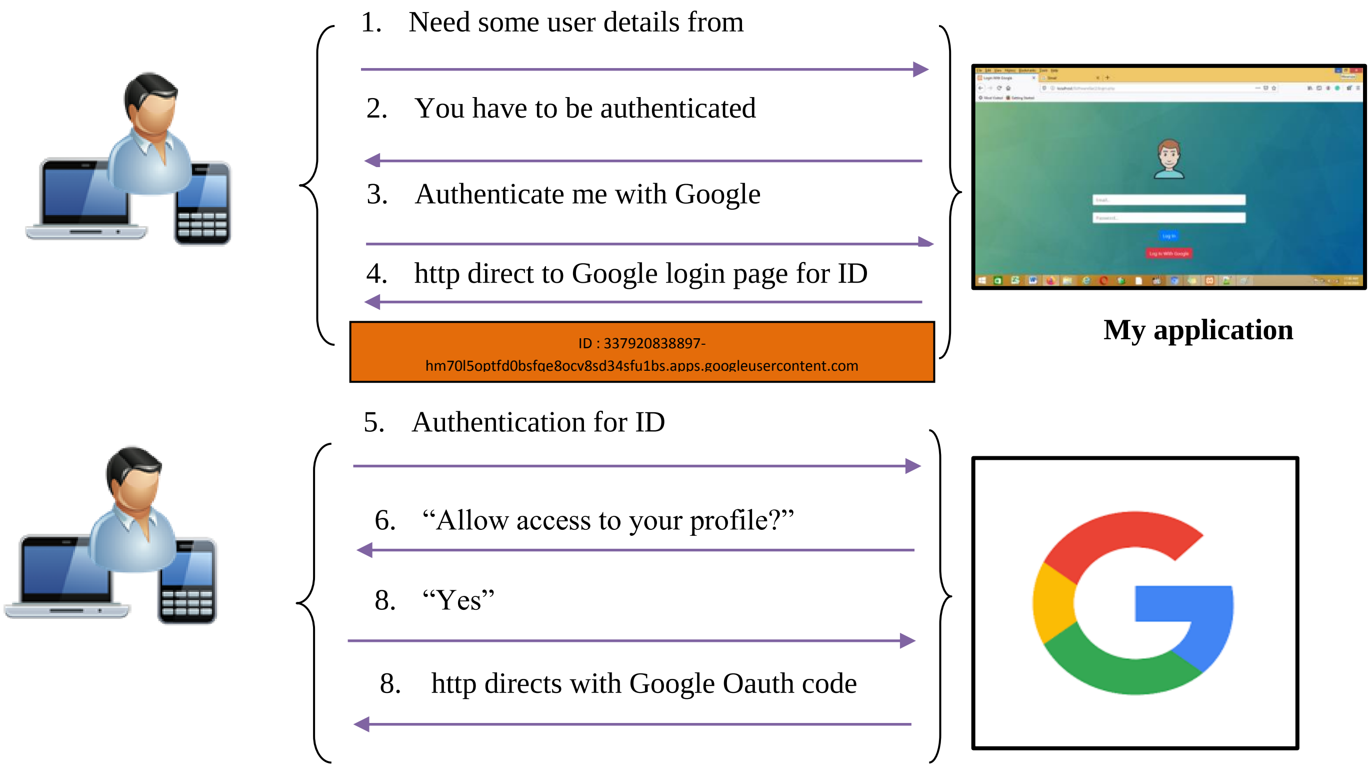
14. You are authenticated

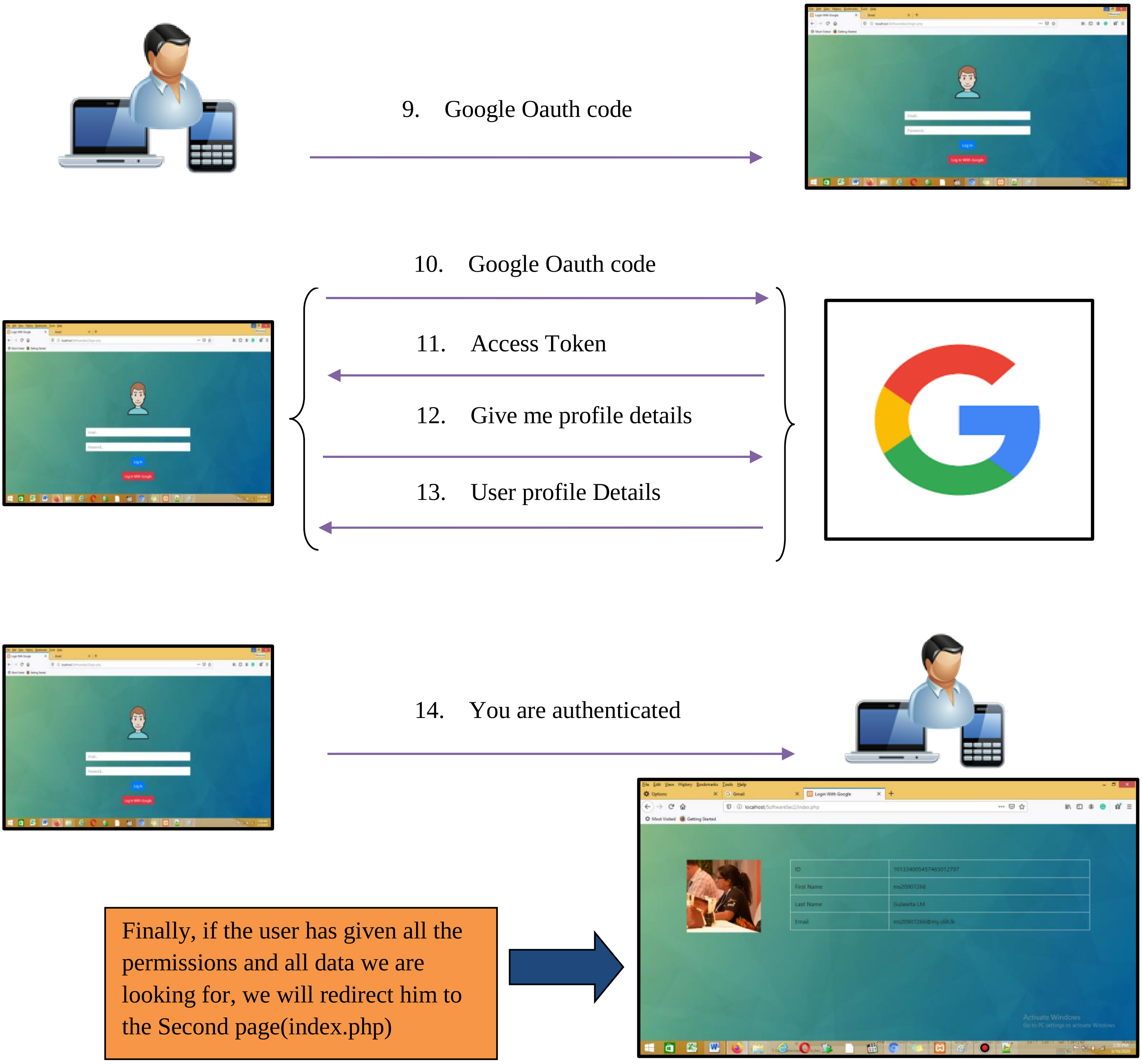
First the client application requests authorization to access service resources from the user. If the user has not already authenticated with the authorization server, the user will be required to login and consent at the server. After successful authentication, user has to authorize the client or user at the authorization server. The server responds to the request with an authorization code.Then the client can exchange the authorization code with an access token by using client secret. In return the authorization server will issue an access token and a refresh token in addition to that,with the acquired access token, the client can send requests to the resource server and access resource server API and user data presenting the access token as the means of authentication.

Selected Scenario

Use OAuth 2.0 protocol to access Google API and authenticate user to display some of user details in a website.

In this snenario, A web application is created for get some details(Eg : Name, First name, Last Name Email and etc) of a user in to a second web page .So there is first page login page(login.php) and login button. It asks user email password. I want that file is only accessible for the people who are logged in.So now at the moment people can login to this page only if they have valid account data.But I also want to enable those people to sign in with the Google account.So what I'm going to do is I will just add one button here "Log in with google" and once someone press the button we will redirect the person to the Google account and accuire account details. And now on the Google they will have to accept the permission that we are asking for and once they accept google will redirect them to the one page that I'm going to call Google callback page and now the key thing will happen inside this page we are going to check few details from the Google and if the user has given all the permissions and all data we are looking for we will redirect him to the Second page(index.php) and if he is not we will just redirect to the login page.


1. Need some user details from
2. You have to be authenticated
3. Authenticate me with Google
4. http direct to Google login page for ID
ID : 337920838897-hm70l5optfd0bsfqe8ocv8sd34sfu1bs.apps.googleusercontent.com
My application
5. Authentication for ID
6. "Allow access to your profile?"
8. "Yes"
8. http directs with Google Oauth code



9. Google Oauth code

10. Google Oauth code

11. Access Token

12. Give me profile details

13. User profile Details

14. You are authenticated

Finally, if the user has given all the permissions and all data we are looking for, we will redirect him to the Second page(index.php)

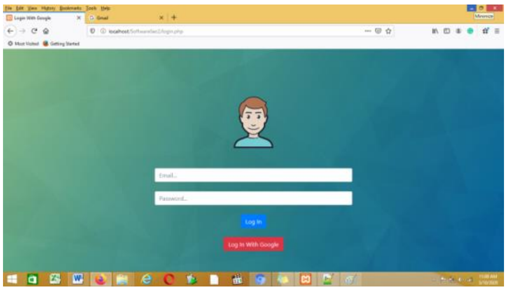
ID	101334005457465012797
First Name	ms20907266
Last Name	Gulawita I.M
Email	ms20907266@my.sliit.lk

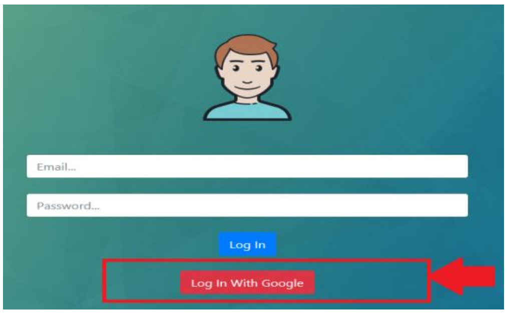
Here I'm describing all the steps I followed for this activity. The process of Creating the Client ID and Client Secret Code is attached as Appendix inthis report.

- Desing My login page 9 (login.php)
- Downloaded Google API Library
- Creating "config.php"
- Get Client ID and Client Secret Code (Appendix)
- Get permission for Google Account
- Create google php file and get code. (g-callback.php)
- Generate Google OAuth 2.0 Code
- Generate Access Token
- Display Google Account data inside my websit

1. Design my login page

http://localhost/SoftwareSec2/login.php



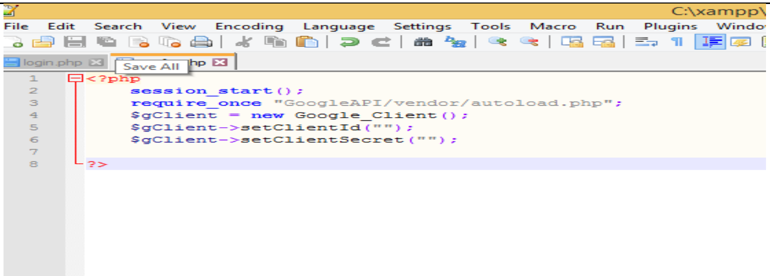


Here my main objective is implementing the funtion of “Login with Google” Buttun.

```
<!doctype html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>Login With Google</title>
   <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vslU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">
</head>
<style>
body {
  background-image: url('images/logo1.jpg');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: cover;
}
</style>
<body>
   <div class="container" style="margin-top: 100px">
      <div class="row justify-content-center">
         <div class="col-md-6 col-offset-3" align="center">

            <img src="images/logo2.png"><br><br>

            <form >
               <input placeholder="Email..." name="email" class="form-control"><br>
               <input type="password" placeholder="Password..." name="password" class="form-control"><br>
               <input type="submit" value="Log In" class="btn btn-primary"><br><br>
               <input type="button" onclick="window.location = '<?php echo $loginURL ?>';" value="Log In With Google" class="btn btn-danger">
            </form>

         </div>
      </div>
   </div>
</body>
</html>
```

2. Then downloaded Google API Library (https://developers.google.com/api-client-library/). Named the that File as “Google API” and added it in my local server folder.

3. Creating “config.php”

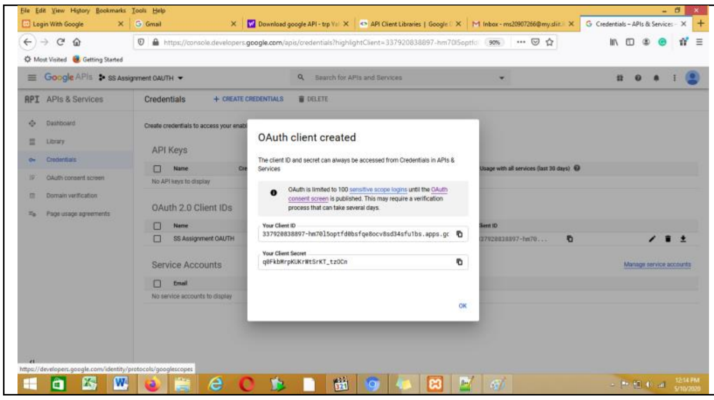

C:\xampp\
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Windo
login.php
Save All

```
<?php
    session_start();
    require_once "GoogleAPI/vendor/autoload.php";
    $gClient = new Google_Client();
    $gClient->setClientId("");
    $gClient->setClientSecret("");
?>
```

```
<?php
        session_start();
        require_once "GoogleAPI/vendor/autoload.php";
        $gClient = new Google_Client();
        $gClient->setClientId("");
        $gClient->setClientSecret("");
?>
```

4. Get ClientId and ClientSecret from Google API

- Added them in config.php and complete php file.



OAuth client created

The client ID and secret can always be accessed from Credentials in APIs & Services

OAuth is limited to 100 sensitive scope logins until the OAuth consent screen is published. This may require a verification process that can take several days.

Your Client ID
337920838897-hm70l5optfd0bsfqe8ocv8sd34sfu1bs.apps.gc

Your Client Secret
q0FkbMrpKUKrWtSrKT_tzOCn

```
<?php
        session_start();
        require_once "GoogleAPI/vendor/autoload.php";
        $gClient = new Google_Client();
        $gClient->setClientId("337920838897-
hm70l5optfd0bsfqe8ocv8sd34sfu1bs.apps.googleusercontent.com");
        $gClient->setClientSecret("q0FkbMrpKUKrWtSrKT_tzOCn");
        $gClient->setApplicationName("SS Assignment OAUTH");
        $gClient->setRedirectUri("http://localhost/SoftwareSec2/g-
callback.php");
        $gClient-
>addScope("https://www.googleapis.com/auth/plus.login
https://www.googleapis.com/auth/userinfo.email");
?>
```

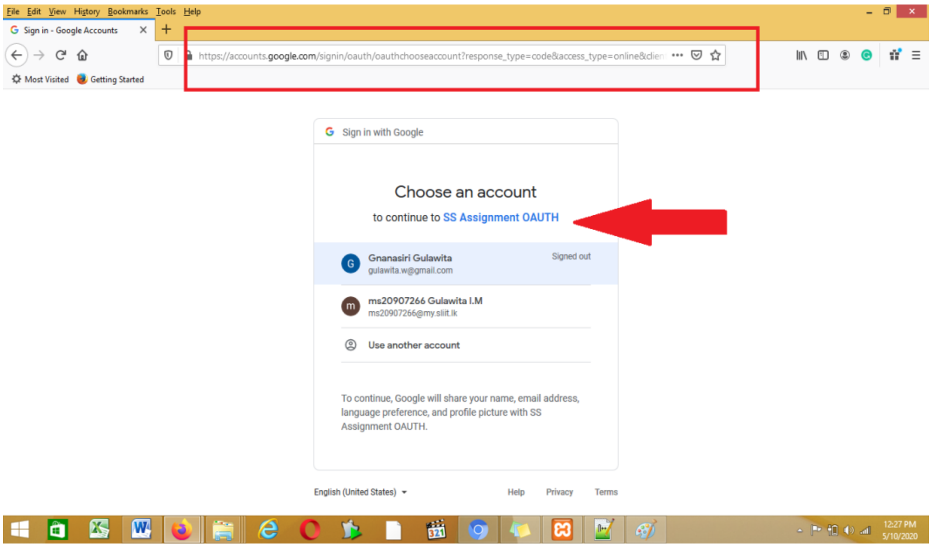
- Again update login.php (Changes are marked in differnet color)

```
<?php
   require_once "config.php";

          $loginURL = $gClient->createAuthUrl();
?>

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
          content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
    <meta http-equiv="X-UA-Compatible" content="ie=edge">
    <title>Login With Google</title>
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vslU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">
</head>
<style>
body {
  background-image: url('images/logo1.jpg');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: cover;
}
</style>
<body>
    <div class="container" style="margin-top: 100px">
        <div class="row justify-content-center">
            <div class="col-md-6 col-offset-3" align="center">

                <img src="images/logo2.png"><br><br>

                <form >
                    <input placeholder="Email..." name="email" class="form-control"><br>
                    <input type="password" placeholder="Password..." name="password" class="form-control"><br>
                    <input type="submit" value="Log In" class="btn btn-primary"><br><br>
                    <input type="button" onclick="window.location = '<?php echo $loginURL ?>';" value="Log In With Google" class="btn btn-danger">
                </form>
            </div>
        </div>
    </div>
</body>
</html>
```

5. Get permission for Google Account

Go to the http://localhost/SoftwareSec2/login.php and clicled Log in with Google

- Generate Access Token



https://accounts.google.com/signin/oauth/oauthchooseaccount?response_type=code&access_type=online&clie...

Sign in with Google

Choose an account

to continue to SS Assignment OAUTH

Gnanasiri Gulawita
gulawita.w@gmail.com
Signed out

ms20907266 Gulawita I.M
ms20907266@my.sliit.lk

Use another account

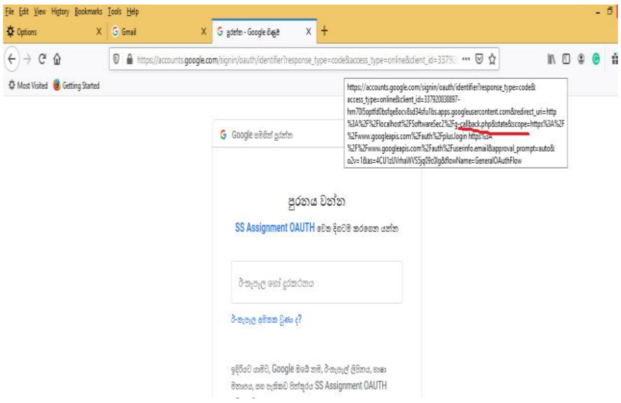
To continue, Google will share your name, email address, language preference, and profile picture with SS Assignment OAUTH.

English (United States) Help Privacy Terms

File Edit View History Bookmarks Tools Help
Sign in - Google Accounts
Most Visited
Getting Started
12:27 PM
5/10/2020

https://accounts.google.com/signin/oauth/oauthchooseaccount?response_type=code&access_type=online&client_id=337920838897-hm70l5optfd0bsfqe8ocv8sd34sfu1bs.apps.googleusercontent.com&redirect_uri=http%3A%2F%2Flocalhost%2FSoftwareSec2%2Fg-callback.php&state&scope=https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fplus.login%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fuserinfo.email&approval_prompt=auto&o2v=1&as=6Qxsis7hyQ_mX0A4wGbDGg&flowName=GeneralOAuthFlow

6. Create google php file and get code. (g-callback.php)



https://accounts.google.com/signin/oauth/identifier?response_type=code&access_type=online&client_id=33792...

https://accounts.google.com/signin/oauth/identifier?response_type=code&access_type=online&client_id=337920838897-hm70l5optfd0bsfqe8ocv8sd34sfu1bs.apps.googleusercontent.com&redirect_uri=http%3A%2F%2Flocalhost%2FSoftwareSec2%2Fg-callback.php&state&scope=https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fplus.login https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fuserinfo.email&approval_prompt=auto&o2v=1&as=4CU1zUVrhaWVS5jq09c0lg&flowName=GeneralOAuthFlow

Google සමඟින් පුරන්න

පුරනය වන්න

SS Assignment OAUTH වෙත දිගටම කරගෙන යන්න

ඊ-තැපැල හෝ දුරකථනය

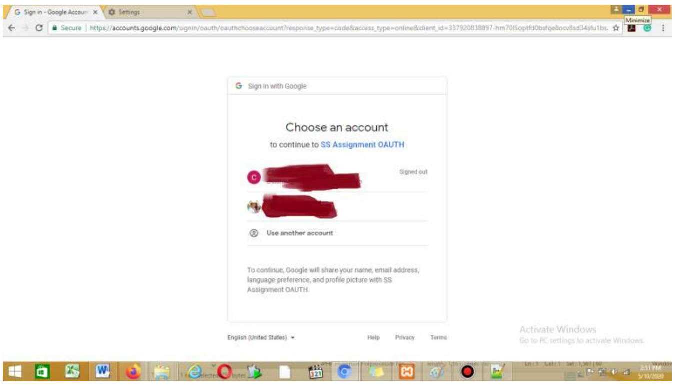
ඊ-තැපැල අමතක වුණා ද?

ඉදිරියට යාමට, Google ඔබේ නම, ඊ-තැපැල් ලිපිනය, භාෂා මනාපය, සහ පැතිකඩ පින්තූරය SS Assignment OAUTH

```
<?php
		require_once "config.php";

		if (isset($_SESSION['access_token']))
			$gClient-
>setAccessToken($_SESSION['access_token']);
		else if (isset($_GET['code'])) {
			$token		=		$gClient-
>fetchAccessTokenWithAuthCode($_GET['code']);
			$_SESSION['access_token'] = $token;
		} else {
			header('Location: login.php');
			exit();
		}

		$oAuth = new Google_Service_Oauth2($gClient);
		$userData = $oAuth->userinfo_v2_me->get();

		$_SESSION['id'] = $userData['id'];
		$_SESSION['email'] = $userData['email'];
		$_SESSION['gender'] = $userData['gender'];
		$_SESSION['picture'] = $userData['picture'];
		$_SESSION['familyName'] = $userData['familyName'];
		$_SESSION['givenName'] = $userData['givenName'];

		header('Location: index.php');
		exit();
?>
```

7. Display Google Account Data inside web application

- Create the redirecting page (index.php)



https://accounts.google.com/signin/oauth/oauthchooseaccount?response_type=code&access_type=online&client_id=337920838897-hm70t5optfd0bsfqe8ocv8sd34sfu1bs.

Choose an account

to continue to SS Assignment OAUTH

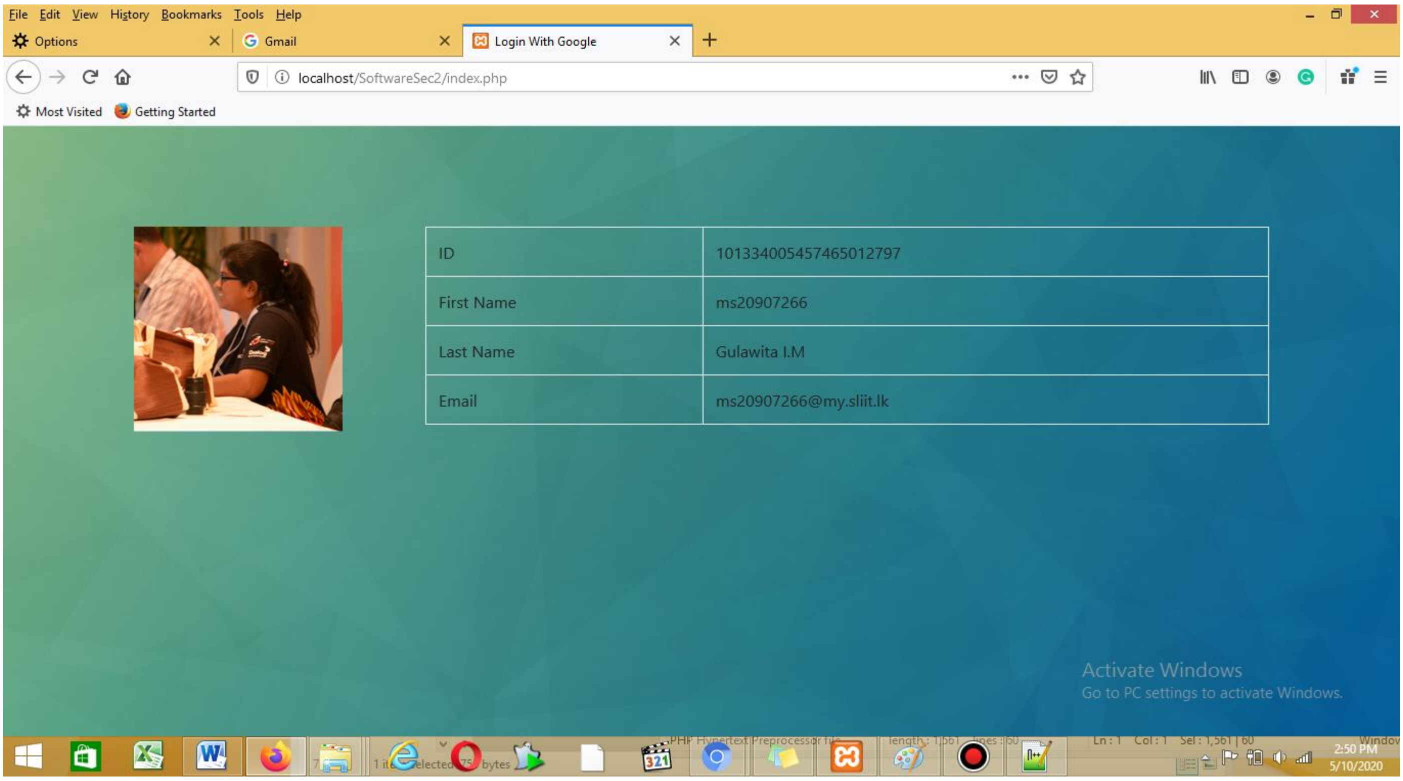
[redacted] Signed out

[redacted]

To continue, Google will share your name, email address, language preference, and profile picture with SS Assignment OAUTH.

```
<?php
		session_start();

		if (!isset($_SESSION['access_token'])) {
			header('Location: login.php');
			exit();
		}
?>
<!doctype html>
<html lang="en">
<head>
		<meta charset="UTF-8">
		<meta name="viewport"
		   content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
		<meta http-equiv="X-UA-Compatible" content="ie=edge">
		<title>Login With Google</title>
		<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vslU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">
</head>
<style>body {
  background-image: url('images/logo1.jpg');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: cover;
}</style>
<body><div class="container" style="margin-top: 100px">
		<div class="row">
			<div class="col-md-3">
				<img style="width: 80%;" src="<?php echo $_SESSION['picture'] ?>">
			</div>

			<div class="col-md-9">
				<table class="table table-hover table-bordered">
					<tbody><tr><td>ID</td>
		<td><?php echo $_SESSION['id'] ?></td>
		</tr><tr>	<td>First Name</td>
		<td><?php echo $_SESSION['givenName'] ?></td>
</tr><tr><td>Last Name</td>
		<td><?php echo $_SESSION['familyName'] ?></td>
		</tr><tr><td>Email</td>
		<td><?php echo $_SESSION['email'] ?></td></tr>
</tbody>
</table>
</div>
</div>
</div>
</body>
</html>
```

- Display required data inside my application.


File Edit View History Bookmarks Tools Help
Options
Gmail
Login With Google
localhost/SoftwareSec2/index.php
Most Visited
Getting Started
Activate Windows
Go to PC settings to activate Windows.
2:50 PM
5/10/2020

ID	101334005457465012797
First Name	ms20907266
Last Name	Gulawita I.M
Email	ms20907266@my.sliit.lk

8. Finally creating logout.php file as follows

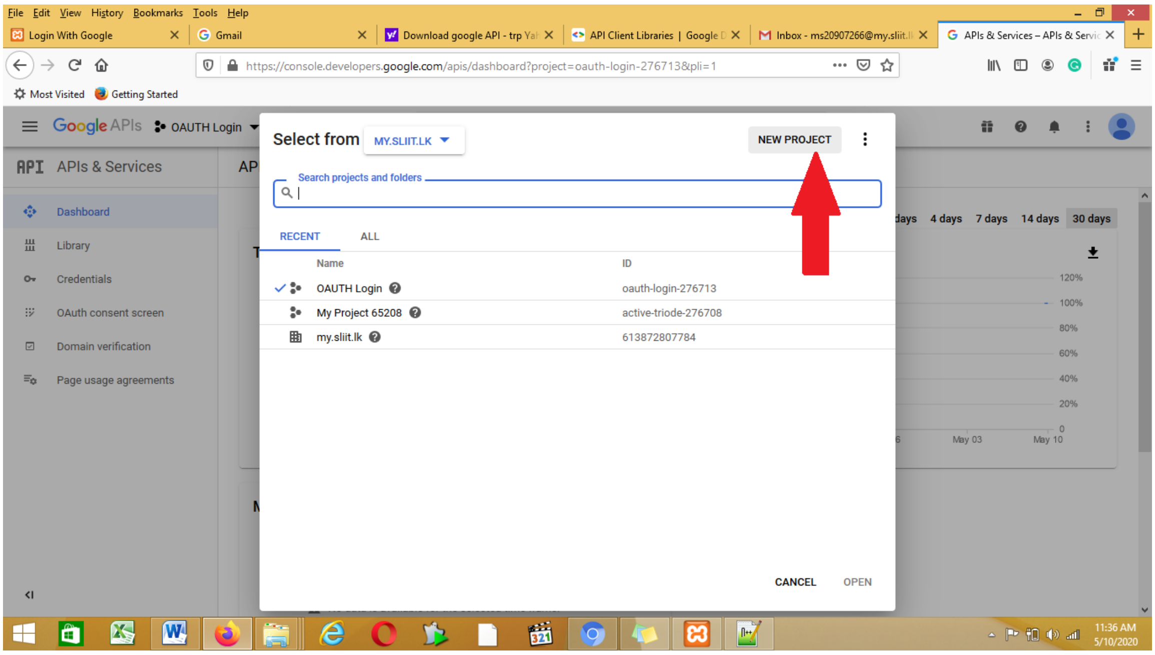
```
<?php
        require_once "config.php";
        unset($_SESSION['access_token']);
        $gClient->revokeToken();
        session_destroy();
        header('Location: login.php');
        exit();
?>
```

Appendix

Following are the steps followed for Gogle API

To use Google identity platform with OAuth, it requires to create new project in Google API console. (https://console.developers.google.com/)

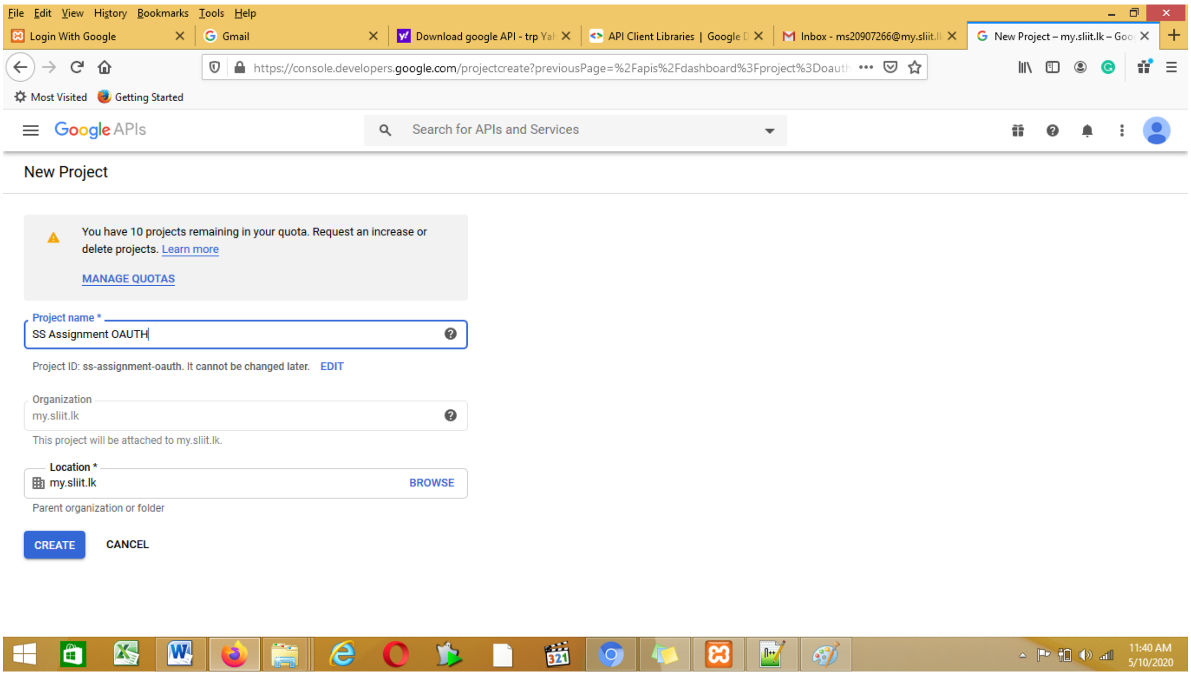
- Create new Project



File Edit View History Bookmarks Tools Help
Login With Google
Gmail
Download google API - trp Yal
API Client Libraries | Google D
Inbox - ms20907266@my.sliit.l
APIs & Services – APIs & Servic
https://console.developers.google.com/apis/dashboard?project=oauth-login-276713&pli=1
Most Visited
Getting Started
Google APIs
OAUTH Login
APIs & Services
Dashboard
Library
Credentials
OAuth consent screen
Domain verification
Page usage agreements
Select from
MY.SLIIT.LK
NEW PROJECT
Search projects and folders
RECENT
ALL
days 4 days 7 days 14 days 30 days
120%
100%
80%
60%
40%
20%
0
May 03
May 10
CANCEL
OPEN
11:36 AM
5/10/2020

Name	ID
✓ OAUTH Login	oauth-login-276713
My Project 65208	active-triode-276708
my.sliit.lk	613872807784

Create new project in Google API console.



File Edit View History Bookmarks Tools Help
Login With Google
Gmail
Download google API - trp Yal
API Client Libraries | Google D
Inbox - ms20907266@my.sliit.l
New Project – my.sliit.lk – Goo
https://console.developers.google.com/projectcreate?previousPage=%2Fapis%2Fdashboard%3Fproject%3Doauth
Most Visited
Getting Started
Google APIs
Search for APIs and Services
BROWSE
CREATE
CANCEL
11:40 AM
5/10/2020

New Project

You have 10 projects remaining in your quota. Request an increase or delete projects. Learn more

MANAGE QUOTAS

Project name *: SS Assignment OAUTH

Project ID: ss-assignment-oauth. It cannot be changed later. EDIT

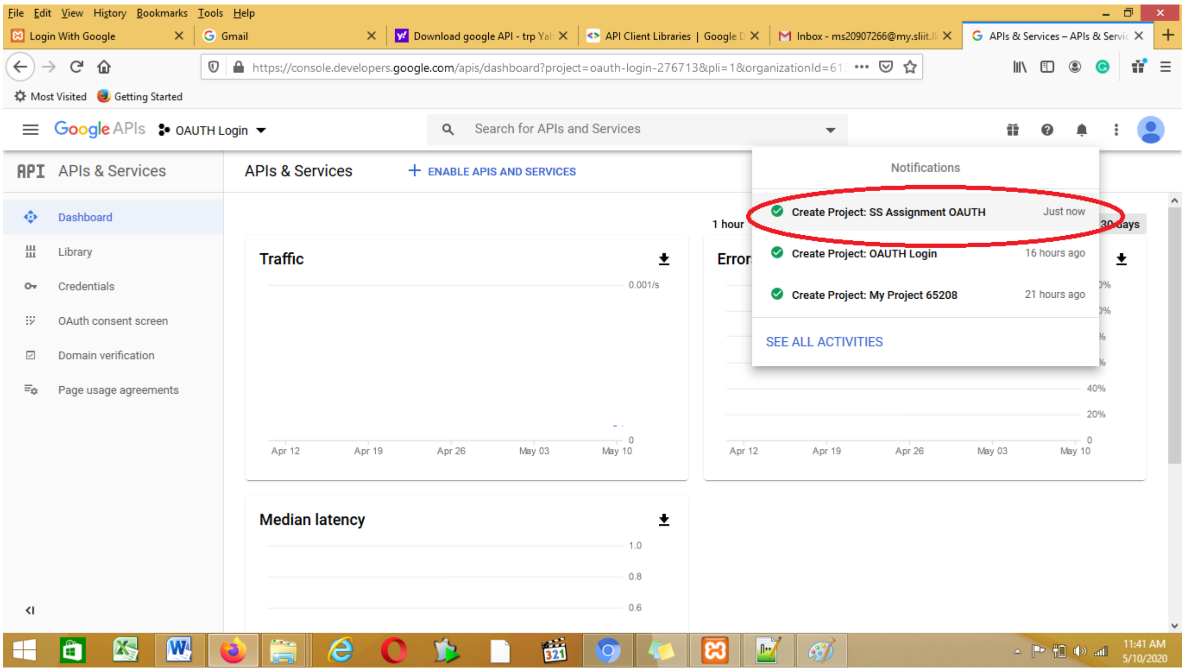
Organization: my.sliit.lk

This project will be attached to my.sliit.lk.

Location *: my.sliit.lk

Parent organization or folder

Create new project as “SS Assignment OAUTH”

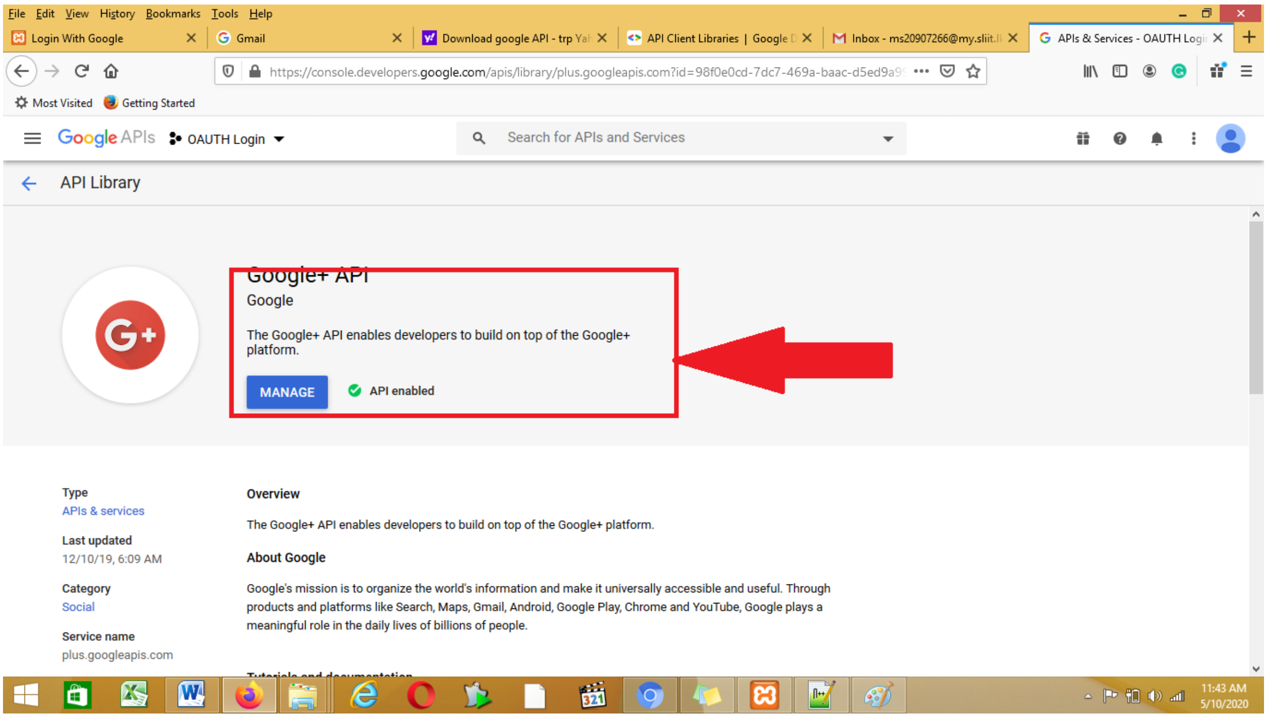


Create Project: SS Assignment OAUTH Just now

Create Project: OAUTH Login 16 hours ago

Create Project: My Project 65208 21 hours ago

SS Assignment OAUTH new project is created


File Edit View History Bookmarks Tools Help
Login With Google
Gmail
Download google API - trp Yal
API Client Libraries | Google D
Inbox - ms20907266@my.sliit.l
APIs & Services - OAUTH Logi
https://console.developers.google.com/apis/library/plus.googleapis.com?id=98f0e0cd-7dc7-469a-baac-d5ed9a9
Most Visited
Getting Started
Google APIs
OAUTH Login
Search for APIs and Services
API Library
MANAGE
11:43 AM
5/10/2020

Google+ API

Google

The Google+ API enables developers to build on top of the Google+ platform.

API enabled

Type
APIs & services

Last updated
12/10/19, 6:09 AM

Category
Social

Service name
plus.googleapis.com

Overview

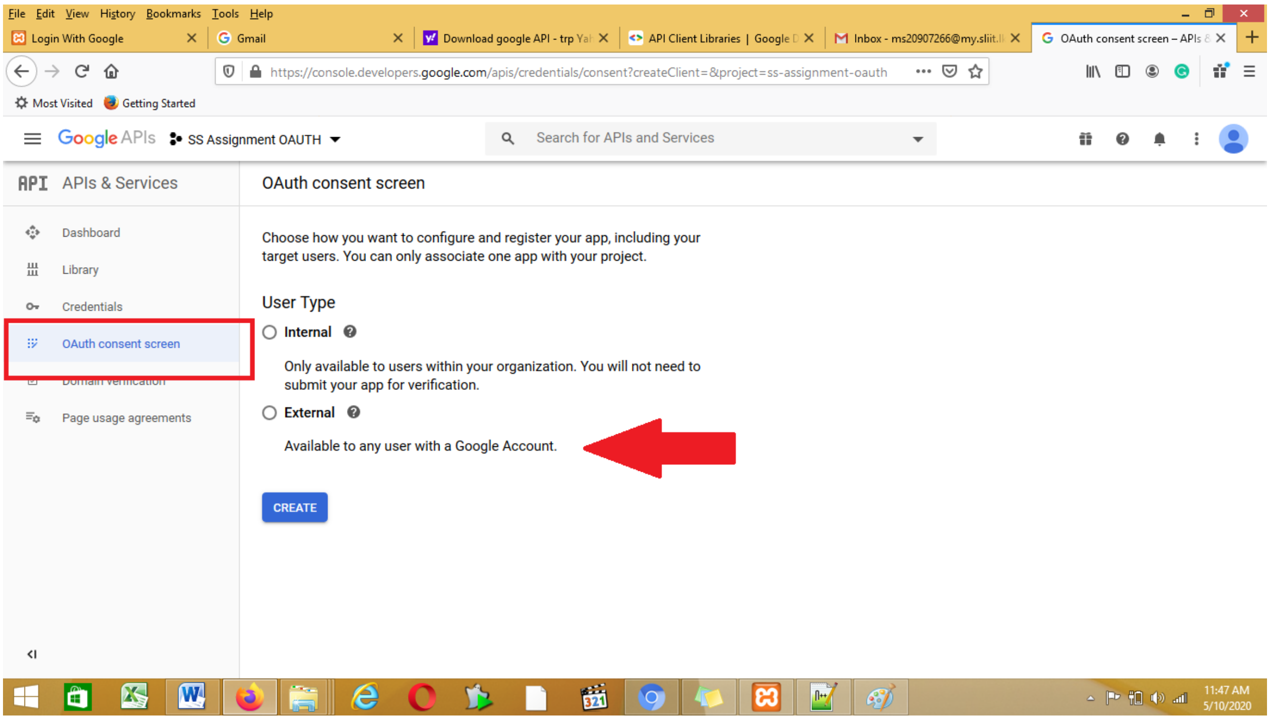
The Google+ API enables developers to build on top of the Google+ platform.

About Google

Google's mission is to organize the world's information and make it universally accessible and useful. Through products and platforms like Search, Maps, Gmail, Android, Google Play, Chrome and YouTube, Google plays a meaningful role in the daily lives of billions of people.

Tutorials and documentation

Go to the API Library and Enable Google+ API



File Edit View History Bookmarks Tools Help
Login With Google
Gmail
Download google API - trp Yal
API Client Libraries | Google D
Inbox - ms20907266@my.sliit.l
OAuth consent screen – APIs &
https://console.developers.google.com/apis/credentials/consent?createClient=&project=ss-assignment-oauth
Most Visited
Getting Started
Google APIs
SS Assignment OAUTH
Search for APIs and Services
API APIs & Services
Dashboard
Library
Credentials
OAuth consent screen
Domain verification
Page usage agreements
11:47 AM
5/10/2020

OAuth consent screen

Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.

User Type

○ Internal

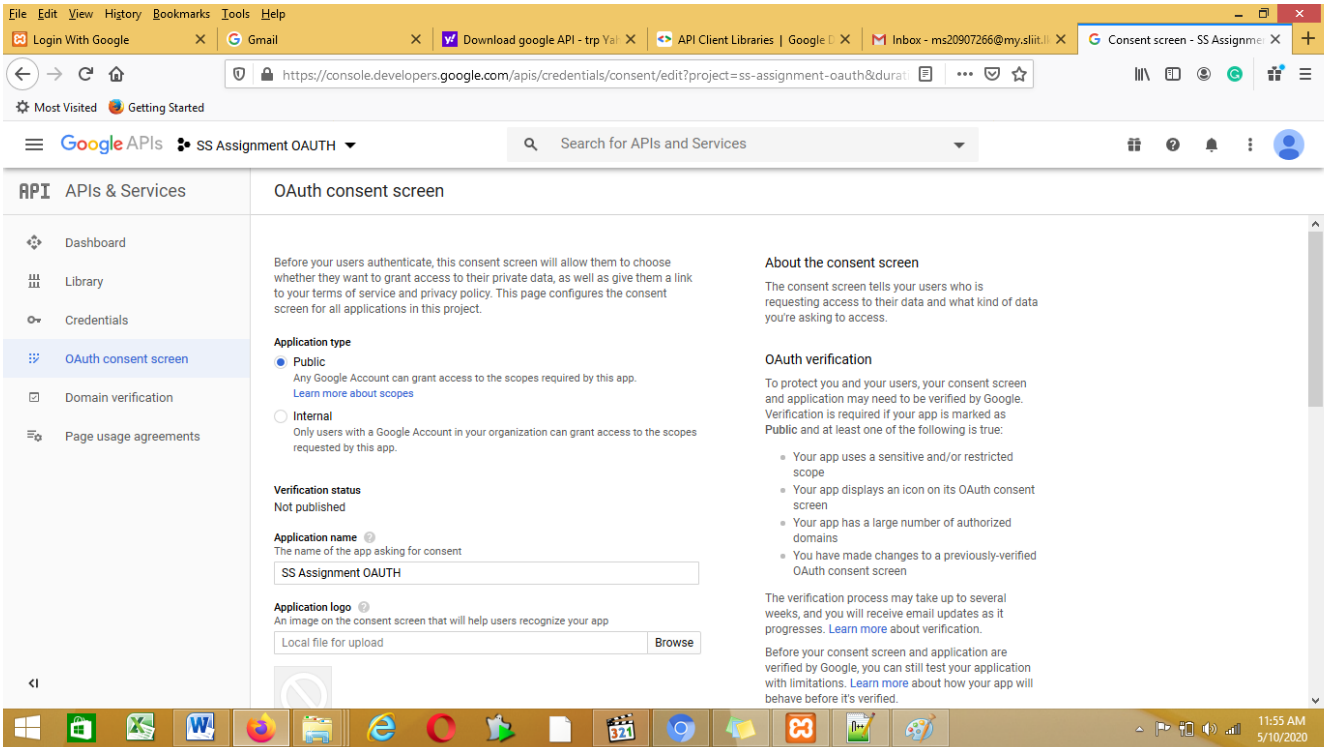
Only available to users within your organization. You will not need to submit your app for verification.

○ External

Available to any user with a Google Account.

CREATE

OAuth consent screen – Selected User Type as External



File Edit View History Bookmarks Tools Help
Login With Google
Gmail
Download google API - trp Yal
API Client Libraries | Google D
Inbox - ms20907266@my.sliit.l
Consent screen - SS Assignme
https://console.developers.google.com/apis/credentials/consent/edit?project=ss-assignment-oauth&durat
Most Visited
Getting Started
Google APIs
SS Assignment OAUTH
Search for APIs and Services
API APIs & Services
Dashboard
Library
Credentials
OAuth consent screen
Domain verification
Page usage agreements
11:55 AM
5/10/2020

OAuth consent screen

Before your users authenticate, this consent screen will allow them to choose whether they want to grant access to their private data, as well as give them a link to your terms of service and privacy policy. This page configures the consent screen for all applications in this project.

Application type

● Public
Any Google Account can grant access to the scopes required by this app.
Learn more about scopes

○ Internal
Only users with a Google Account in your organization can grant access to the scopes requested by this app.

Verification status
Not published

Application name
The name of the app asking for consent
SS Assignment OAUTH

Application logo
An image on the consent screen that will help users recognize your app
Local file for upload | Browse

About the consent screen

The consent screen tells your users who is requesting access to their data and what kind of data you're asking to access.

OAuth verification

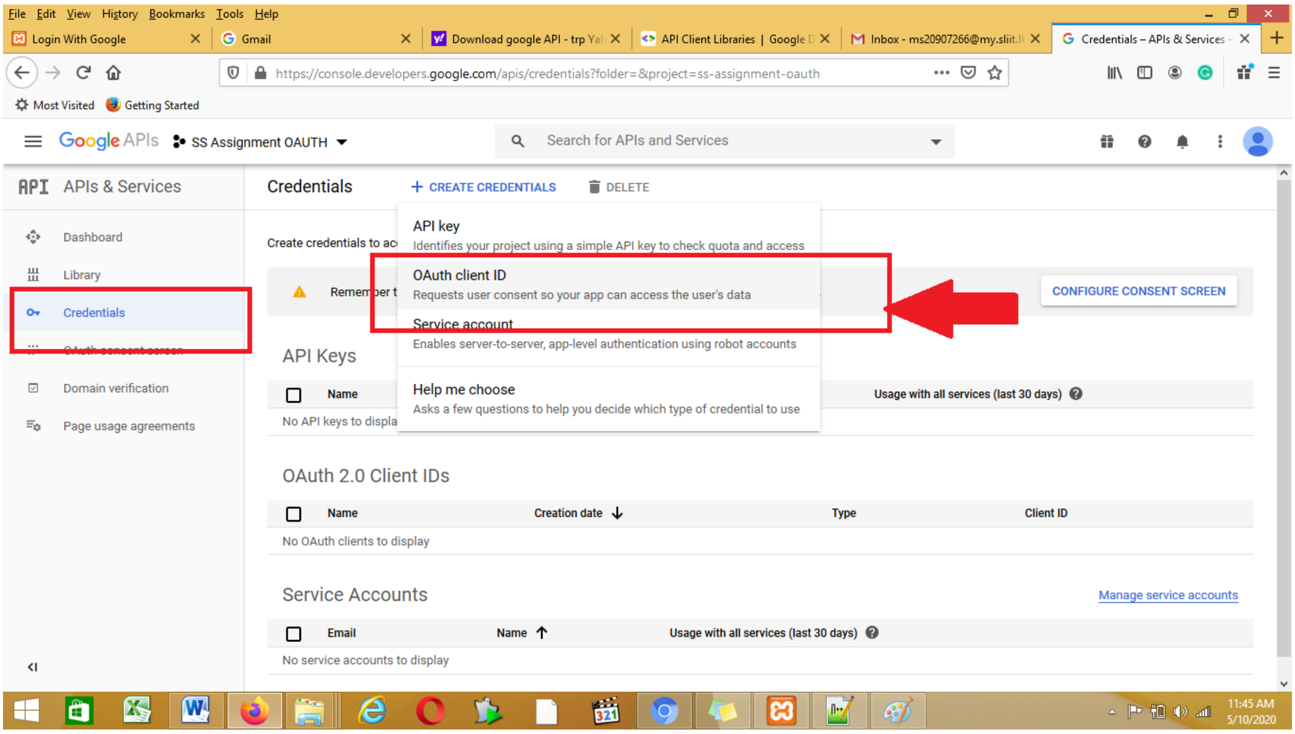
To protect you and your users, your consent screen and application may need to be verified by Google. Verification is required if your app is marked as **Public** and at least one of the following is true:

- Your app uses a sensitive and/or restricted scope
- Your app displays an icon on its OAuth consent screen
- Your app has a large number of authorized domains
- You have made changes to a previously-verified OAuth consent screen

The verification process may take up to several weeks, and you will receive email updates as it progresses. Learn more about verification.

Before your consent screen and application are verified by Google, you can still test your application with limitations. Learn more about how your app will behave before it's verified.

OAuth consent screen – Given required details



https://console.developers.google.com/apis/credentials?folder=&project=ss-assignment-oauth

Google APIs · SS Assignment OAUTH

APIs & Services

- Dashboard
- Library
- Credentials
- OAuth consent screen
- Domain verification
- Page usage agreements

Credentials   + CREATE CREDENTIALS   DELETE

API key
Identifies your project using a simple API key to check quota and access

OAuth client ID
Requests user consent so your app can access the user's data

Service account
Enables server-to-server, app-level authentication using robot accounts

Help me choose
Asks a few questions to help you decide which type of credential to use

Create credentials to ac…

Remember t…   CONFIGURE CONSENT SCREEN

API Keys

□	Name	Usage with all services (last 30 days)
No API keys to displa…		

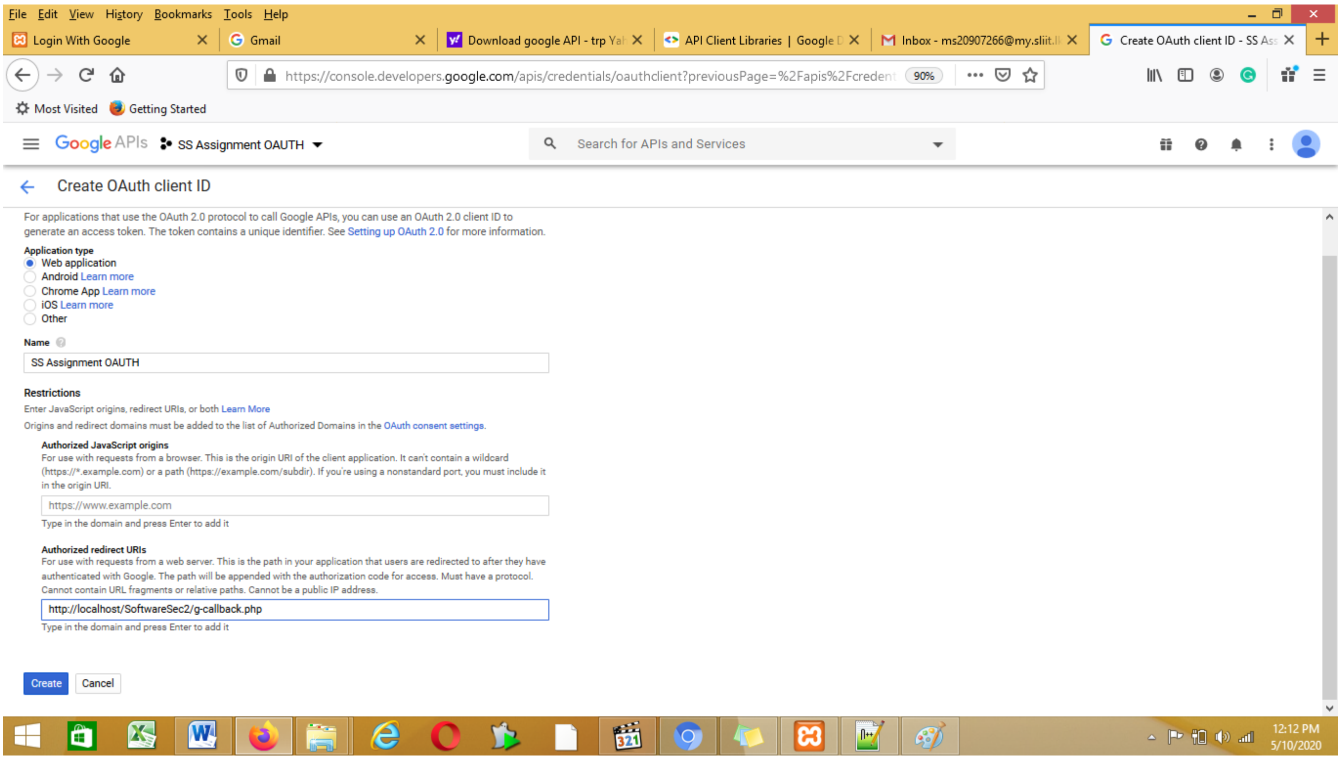
OAuth 2.0 Client IDs

□	Name	Creation date	Type	Client ID
No OAuth clients to display				

Service Accounts   Manage service accounts

□	Email	Name	Usage with all services (last 30 days)
No service accounts to display			

Go to credentials and Create credentioals with OAuth Cliend ID



https://console.developers.google.com/apis/credentials/oauthclient?previousPage=%2Fapis%2Fcreden…

Google APIs · SS Assignment OAUTH

Create OAuth client ID

For applications that use the OAuth 2.0 protocol to call Google APIs, you can use an OAuth 2.0 client ID to generate an access token. The token contains a unique identifier. See Setting up OAuth 2.0 for more information.

Application type
- ● Web application
- ○ Android Learn more
- ○ Chrome App Learn more
- ○ iOS Learn more
- ○ Other

Name
SS Assignment OAUTH

Restrictions

Enter JavaScript origins, redirect URIs, or both Learn More

Origins and redirect domains must be added to the list of Authorized Domains in the OAuth consent settings.

Authorized JavaScript origins
For use with requests from a browser. This is the origin URI of the client application. It can't contain a wildcard (https://*.example.com) or a path (https://example.com/subdir). If you're using a nonstandard port, you must include it in the origin URI.

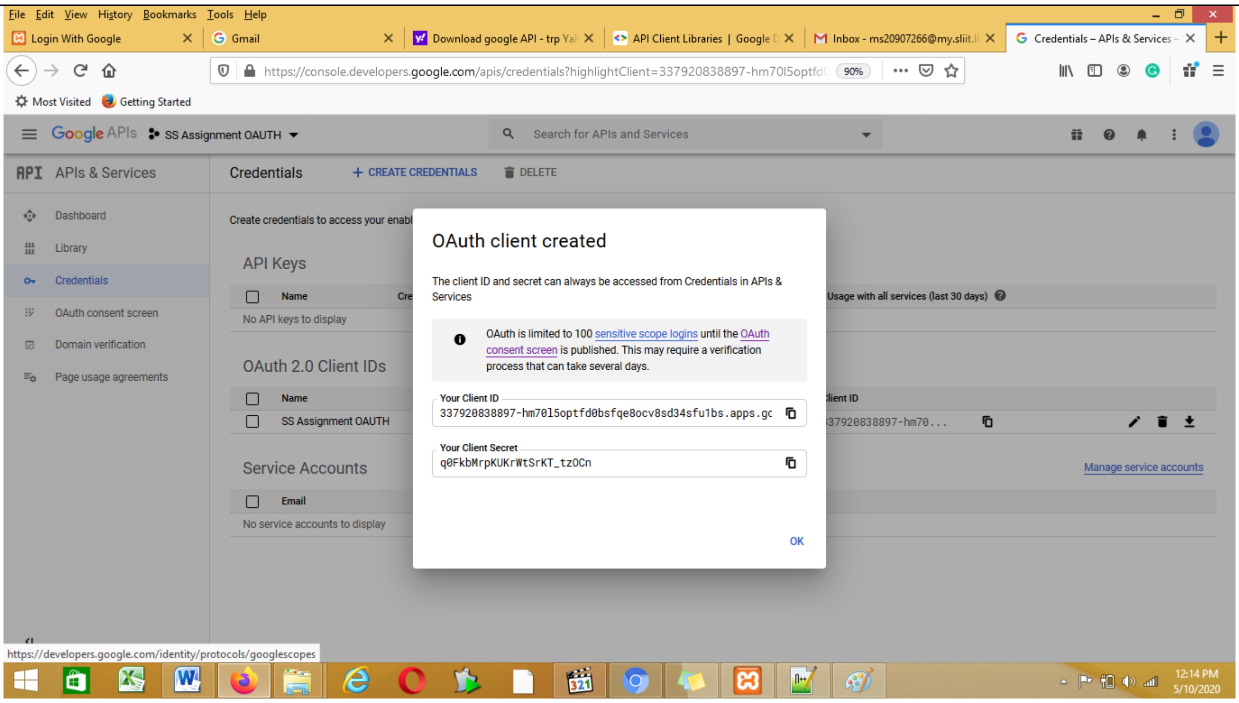
https://www.example.com
Type in the domain and press Enter to add it

Authorized redirect URIs
For use with requests from a web server. This is the path in your application that users are redirected to after they have authenticated with Google. The path will be appended with the authorization code for access. Must have a protocol. Cannot contain URL fragments or relative paths. Cannot be a public IP address.

http://localhost/SoftwareSec2/g-callback.php
Type in the domain and press Enter to add it

Create   Cancel

OAuth credentials – Given required details



OAuth client created

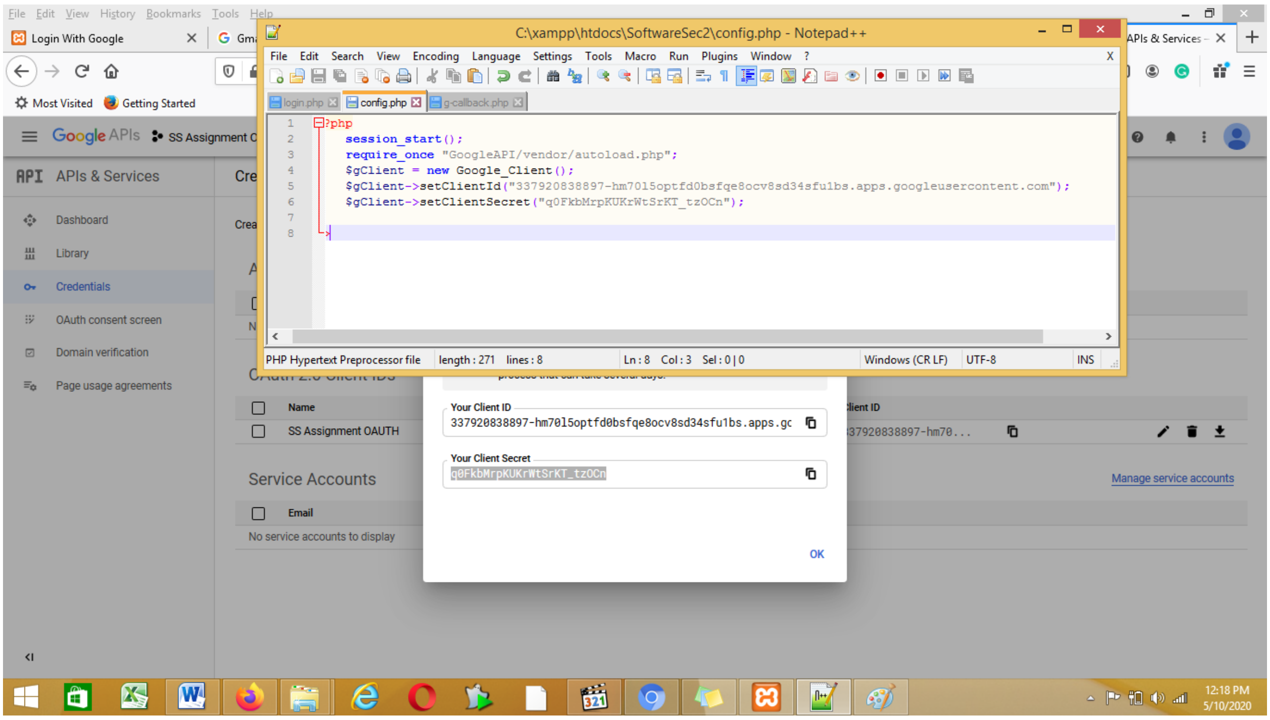
The client ID and secret can always be accessed from Credentials in APIs & Services

OAuth is limited to 100 sensitive scope logins until the OAuth consent screen is published. This may require a verification process that can take several days.

Your Client ID: 337920838897-hm7015optfd0bsfqe8ocv8sd34sfu1bs.apps.gc

Your Client Secret: q0FkbMrpKUKrWtSrKT_tzOCn

Get My Client ID and Client Secret



C:\xampp\htdocs\SoftwareSec2\config.php - Notepad++

```
?php
    session_start();
    require_once "GoogleAPI/vendor/autoload.php";
    $gClient = new Google_Client();
    $gClient->setClientId("337920838897-hm7015optfd0bsfqe8ocv8sd34sfu1bs.apps.googleusercontent.com");
    $gClient->setClientSecret("q0FkbMrpKUKrWtSrKT_tzOCn");

>
```

Added Client ID and Client Secret in to config.php

Source Codes

Login.php

```
<?php
  require_once "config.php";
if (isset($_SESSION['access_token'])) {
                        header('Location: index.php');
                        exit();
            }


            $loginURL = $gClient->createAuthUrl();
?>

<!doctype html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport"
        content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>Login With Google</title>
   <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-
/Y6pD6FV/Vv2HJnA6t+vslU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">
</head>
<style>
body {
  background-image: url('images/logo1.jpg');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: cover;
}
</style>
<body>
   <div class="container" style="margin-top: 100px">
      <div class="row justify-content-center">
         <div class="col-md-6 col-offset-3" align="center">

            <img src="images/logo2.png"><br><br>

            <form >
               <input placeholder="Email..." name="email" class="form-control"><br>
               <input type="password" placeholder="Password..." name="password" class="form-control"><br>
               <input type="submit" value="Log In" class="btn btn-primary"><br><br>
               <input type="button" onclick="window.location = '<?php echo $loginURL ?>';" value="Log In With Google" class="btn btn-
danger">
            </form></div>
      </div> </div>
</body>
</html>
```

Config.php

```
<?php
            session_start();
            require_once "GoogleAPI/vendor/autoload.php";
            $gClient = new Google_Client();
            $gClient->setClientId("337920838897-hm70l5optfd0bsfqe8ocv8sd34sfu1bs.apps.googleusercontent.com");
            $gClient->setClientSecret("q0FkbMrpKUKrWtSrKT_tzOCn");
            $gClient->setApplicationName("SS Assignment OAUTH");
            $gClient->setRedirectUri("http://localhost/SoftwareSec2/g-callback.php");
            $gClient->addScope("https://www.googleapis.com/auth/plus.login https://www.googleapis.com/auth/userinfo.email");
?>
```

g-callback.php

```
<?php
        require_once "config.php";

        if (isset($_SESSION['access_token']))
                $gClient->setAccessToken($_SESSION['access_token']);
        else if (isset($_GET['code'])) {
                $token = $gClient->fetchAccessTokenWithAuthCode($_GET['code']);
                $_SESSION['access_token'] = $token;
        } else {
                header('Location: login.php');
                exit();
        }

        $oAuth = new Google_Service_Oauth2($gClient);
        $userData = $oAuth->userinfo_v2_me->get();

        $_SESSION['id'] = $userData['id'];
        $_SESSION['email'] = $userData['email'];
        $_SESSION['gender'] = $userData['gender'];
        $_SESSION['picture'] = $userData['picture'];
        $_SESSION['familyName'] = $userData['familyName'];
        $_SESSION['givenName'] = $userData['givenName'];

        header('Location: index.php');
        exit();
?>
```

index.php

```
<?php
        session_start();

        if (!isset($_SESSION['access_token'])) {
                header('Location: login.php');
                exit();
        }
?>
<!doctype html>
<html lang="en">
<head>
        <meta charset="UTF-8">
        <meta name="viewport"
            content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
        <meta http-equiv="X-UA-Compatible" content="ie=edge">
        <title>Login With Google</title>
        <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css"
integrity="sha384-/Y6pD6FV/Vv2HJnA6t+vslU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M"
crossorigin="anonymous">
</head>
<style>
body {
  background-image: url('images/logo1.jpg');
  background-repeat: no-repeat;
  background-attachment: fixed;
  background-size: cover;
}
</style>
```

```
<body>
<div class="container" style="margin-top: 100px">
        <div class="row">
                <div class="col-md-3">
                        <img style="width: 80%;" src="<?php echo $_SESSION['picture'] ?>">
                </div>

                <div class="col-md-9">
                        <table class="table table-hover table-bordered">
                                <tbody>
                                        <tr>
                                                <td>ID</td>
                                                <td><?php echo $_SESSION['id'] ?></td>
                                        </tr>
                                        <tr>
                                                <td>First Name</td>
                                                <td><?php echo $_SESSION['givenName'] ?></td>
                                        </tr>
                                        <tr>
                                                <td>Last Name</td>
                                                <td><?php echo $_SESSION['familyName'] ?></td>
                                        </tr>
                                        <tr>
                                                <td>Email</td>
                                                <td><?php echo $_SESSION['email'] ?></td>
                                        </tr>
                                </tbody></table></div>
        </div>
</div>
</body>
</html>
```

logout.php

```
<?php
        require_once "config.php";
        unset($_SESSION['access_token']);
        $gClient->revokeToken();
        session_destroy();
        header('Location: login.php');
        exit();
?>
```