**Google Login Configuration**

The hassle of managing different accounts for different sites specifically, when we have several passwords for different services and a website asks us to create yet another account on their site.

To handle with this, we could offer a single sign-on feature to allow visitors to use their existing credentials to open an account on our site. In this users can login by using their existing Google account. This is a more convenient way for new users to register with a third-party site instead of signing up for a new account with yet another username and password.

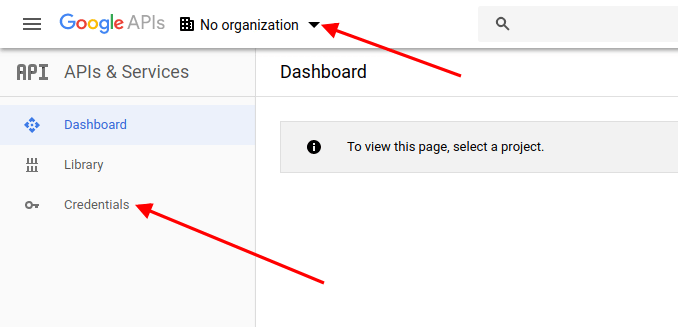
This document describes how to complete a basic Google Login configuration.

1. Create authorization credentials
2. Load the Google platform library
3. Specify our app’s client ID
4. Add a Google Login button
5. Get profile information

**We will go through the basic setup which is required to configure Google Login with our web application.**

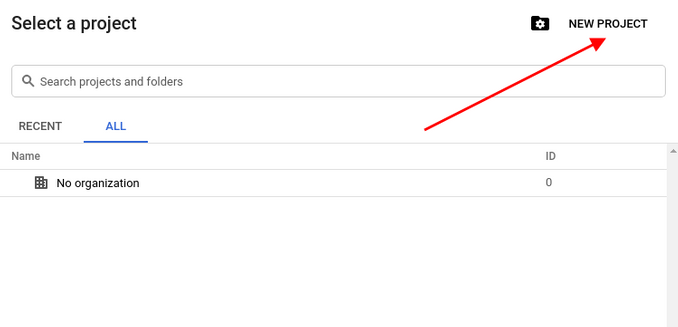
Firstly, we need to create an application with Google which will allow us to register our site with Google. It allows us to set up basic information about our website and a couple of technical details as well.

Once we were logged in with Google, open the **Google Developers Console** (<https://console.cloud.google.com>). That should open up the **Google Dashboard page**.

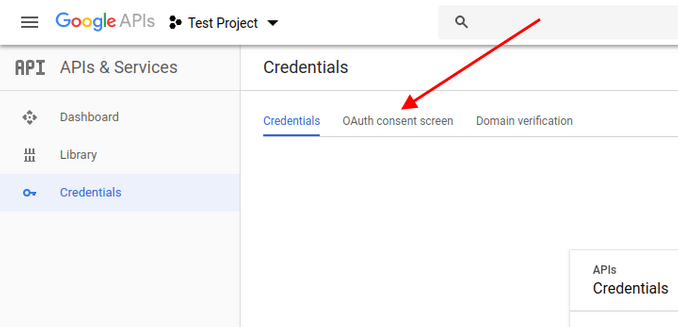


In that page from the top-left menu, click on the **Select a project link**. That should open up a popup.

Next we have to click on the **New Project** link and it will ask us to enter the **Project Name** and other details.

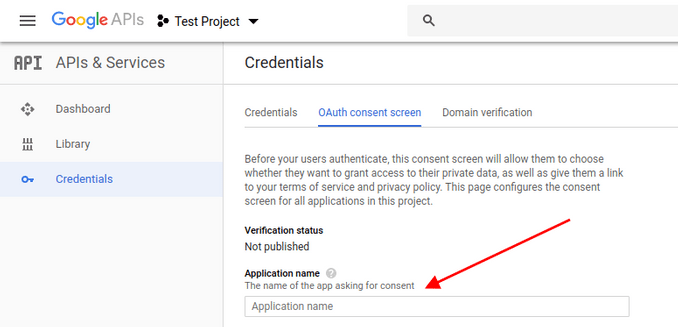


Click on the **Create** button to save our new project. We will be redirected to the Dashboard page. Click on the **Credentials** from the left sidebar, and go to the **OAuth consent screen** tab.

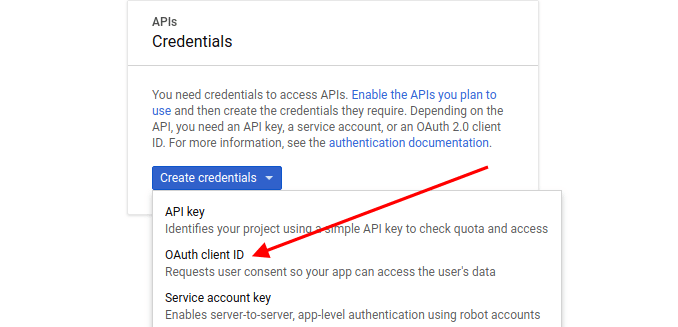


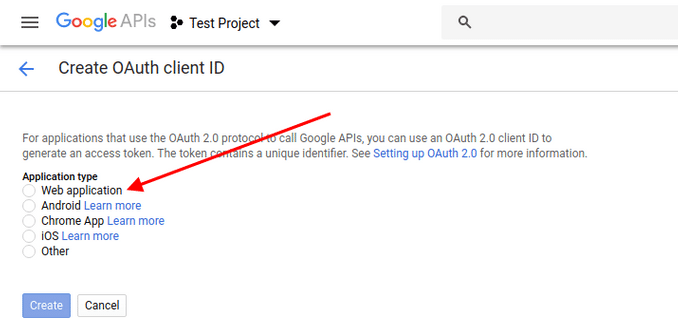
On this tab we need to enter the details about our application, like the application name and few other details. Fill in the necessary details and save them.

Next, click on Credentials box under the Credentials tab.



Click **Client credentials>OAuth client ID** to create a new set of credentials for our application. That should present us with a screen that asks us to choose an appropriate application type. It is because a client ID is used to identify a single app to Google’s OAuth servers. If our app runs on multiple platforms, each will need its own client ID. See **Settings up OAuth 2.0** for more information.



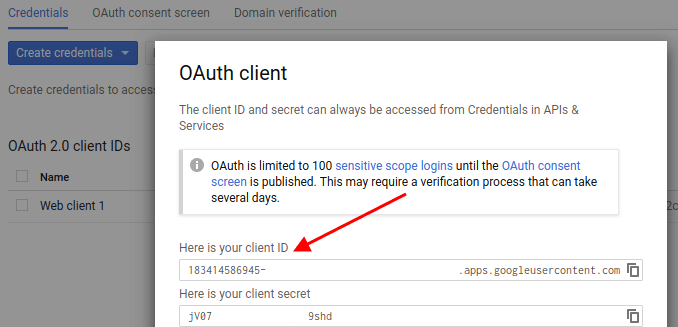


Choose an application type.

Next click on the **Create** button. It will be asked us to provide a few more details about our application.

Enter all the details and save it. After that we need to set the **Redirect URI** as per our application settings. It is the URL where the user will be redirected after login.

At this point, we have created the **Google OAuth2 client application**, and now we should be able to use this application to integrate Google login.



Lastly we have to note down or copy the **Client ID** and **Client Secret** values that will be required during the application configuration on our end.

**Load the Google platform library**

We must include the Google platform library on our web pages that integrate Google Login.

**Specify our app’s client ID**

Specify the client Id that we created for our app in the Google Developers Console.

**Add a Google Login button**

We can add a button by writing a few lines of code that configures itself to have the appropriate text, logo for the login state of the user

**Get profile information**

This Google login function returns Google user basic profile information such as username email and the profile image.