

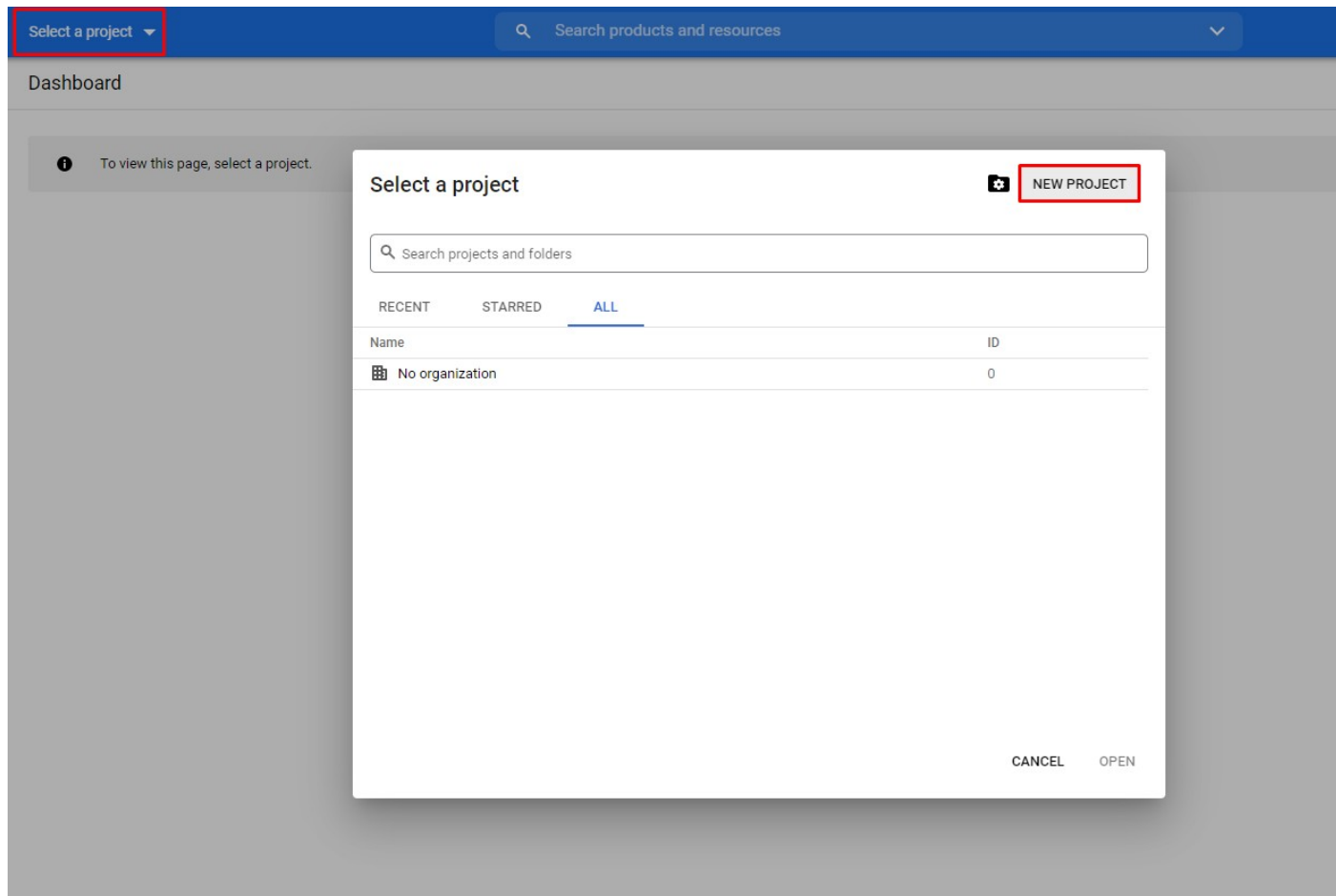
# Google Calendar Integration

## Register a Google Application and enable Calendar API

You need a Google Application so that you can get API keys through which you can make API calls. If you have an existing Google Application you can use that, just make sure that you enable Calendar API. If you don't have an existing Google Application follow the below steps to get one

## Step 1: Go to [Google API Console](#)

Create a project by clicking "**Select a project**" (at the top), and then clicking on the "**New Project**" button in the dialog box top right. In the next screen enter your project name and click the **Create** button.



## Step 2: Enable Calendar API

Click the **Library** tab on the left. Search for "**Calendar API**" and enable it. By enabling "**Calendar API**", your Google application can get access to the user's Calendar.

---



### Google Calendar API

Google Enterprise API

Integrate with Google Calendar using the Calendar API.

ENABLE

TRY THIS API [↗](#)



## Step 3: Setup OAuth Consent Screen

Setup your app info, follow all steps, fill out the mandatory fields and save it.

API	APIs & Services	OAuth consent screen
	<ul style="list-style-type: none"><li>Dashboard</li><li>Library</li><li>Credentials</li><li><b>OAuth consent screen</b></li><li>Domain verification</li><li>Page usage agreements</li></ul>	<p>Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.</p> <p><b>User Type</b></p> <p><input type="radio"/> Internal ?</p> <p>Only available to users within your organization. You will not need to submit your app for verification. <a href="#">Learn more about user type</a></p> <p><input checked="" type="radio"/> External ?</p> <p>Available to any test user with a Google Account. Your app will start in testing mode and will only be available to users you add to the list of test users. Once your app is ready to push to production, you may need to verify your app. <a href="#">Learn more about user type</a></p> <p><a href="#">CREATE</a></p>

## Step 4: Setup Credentials

Now click on the **"Credentials"** tab from left, click on **"Create credentials"** Choose **"OAuth Client ID"** as the type

Google Cloud Platform		My Calendar Project	Search project
API	APIs & Services	Credentials	<a href="#">+ CREATE CREDENTIALS</a> <a href="#">DELETE</a>
	<ul style="list-style-type: none"><li>Dashboard</li><li>Library</li><li><b>Credentials</b></li><li>OAuth consent screen</li><li>Domain verification</li><li>Page usage agreements</li></ul>	<p>Create credentials to access your project's APIs</p> <p><b>API Keys</b></p> <ul style="list-style-type: none"><li><input type="checkbox"/> Name</li><li><input type="checkbox"/> API key 1</li></ul> <p><b>OAuth 2.0 Client IDs</b></p> <ul style="list-style-type: none"><li><input type="checkbox"/> Name</li></ul> <p>No OAuth clients to display</p>	<p>API key Identifies your project using a simple API key to check quota and access</p> <p><b>OAuth client ID</b> Requests user consent so your app can access the user's data</p> <p>Service account Enables server-to-server, app-level authentication using robot accounts</p> <p>Help me choose Asks a few questions to help you decide which type of credential to use</p>

Add a redirect url in the section **Authorised redirect URIs**. This url should point to your redirect url script.

Google Cloud Platform My Calendar Project

API APIs & Services

Dashboard

Library

Credentials

OAuth consent screen

Domain verification

Page usage agreements

Create OAuth client ID

A client ID is used to identify a single app to Google's OAuth servers. If your app runs on multiple platforms, each will need its own client ID. See [Setting up OAuth 2.0](#) for more information. [Learn more](#) about OAuth client types.

Application type \*  
Web application

Name \*  
Web client 1

The name of your OAuth 2.0 client. This name is only used to identify the client in the console and will not be shown to end users.

The domains of the URIs you add below will be automatically added to your [OAuth consent screen](#) as [authorized domains](#).

Authorized JavaScript origins ?

For use with requests from a browser

+ ADD URI

Authorized redirect URIs ?

For use with requests from a web server

Go to admin panel > settings > calendar settings and copy your own generated link and paste here

URIs \*  
Paste your link here

+ ADD URI

CREATE CANCEL

Aoxio

Dashboard

Settings

Website Settings

Payment Settings

License

Change Password

Language

Plans

Website Settings

Appearance

Preferences

Charts

Calendar Settings

Twilio SMS Settings

Email Settings

Google Calendar

Client Id  
926711531504-lhamb9e7bbpfptvb5t88d8jghqalhGOO.a

Client Secret  
GkLThGkP55gKbcJwS7agg

Google Callback URL  
http://localhost/aoxio/gc/auth/oauth

Copy your own generated link

On **success** you will get the App Client **ID** and **Secret**

## OAuth client created

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

**i** OAuth access is restricted to the [test users](#) listed on your [OAuth consent screen](#)

Your Client ID —  
393484700987-m20cv0ah5ofivcjpag6po6rkvp0ihq. .apps.g

Your Client Secret —  
GOCSPX-80n-JVzrG2VcxHTIyFNIUxvg

⬇️ DOWNLOAD JSON

OK

Copy this **Client ID & Secret** and save on your site **settings > calendar settings**

⚙️ Website Settings

🔧 Appearance

⚙️ Preferences

📊 Charts

📅 **Calendar Settings**

💬 Twilio SMS Settings

@ Email Settings

% Subscription Tax

🔒 reCaptcha V2 Settings

🌐 Social Settings

Google Calendar

Client Id

926711531504-lhamb9e7bbpftvb5t88d8jghq00.apps.googleusercontent.com

Client Secret

GkLThGkP55gKbcJwS72aC

Google Callback URL

http://localhost/aoxio/gc/auth/oauth

Save Changes

Note: If your site is ready for production then you can publish your app for live mode.