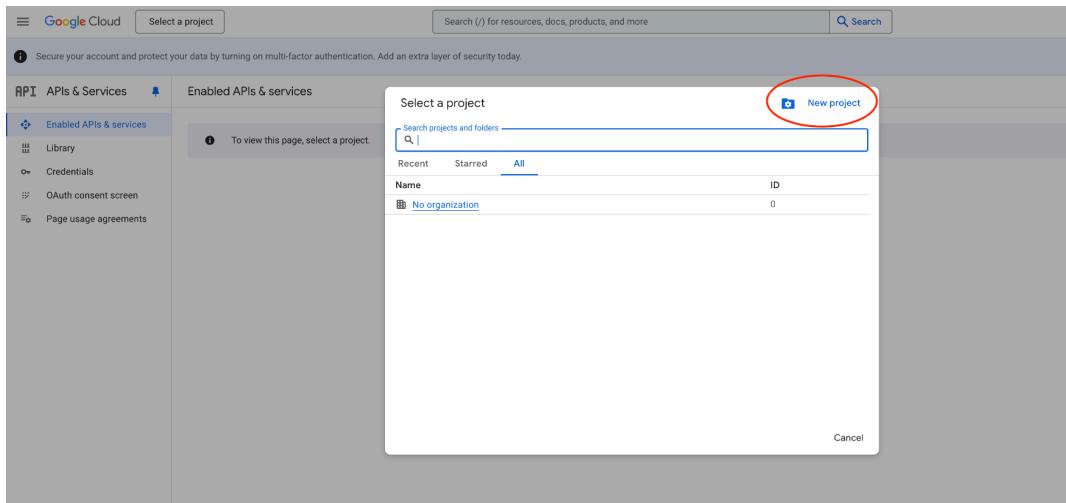


Google Calendar API Integration

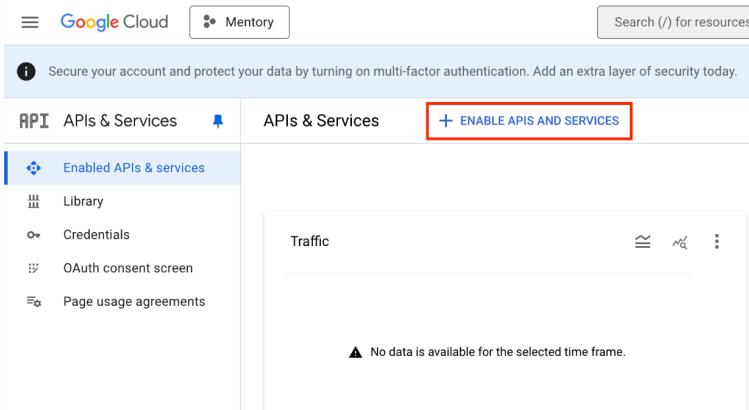
Step 1: Set Up a Google Cloud Project

1. **Go to Google Cloud Console:** <https://console.cloud.google.com/>.
2. **Create a New Project:**
 - Click the project dropdown at the top of the page and select **New Project**.
 - Enter a project name and optional organization details.
 - Click **Create**.



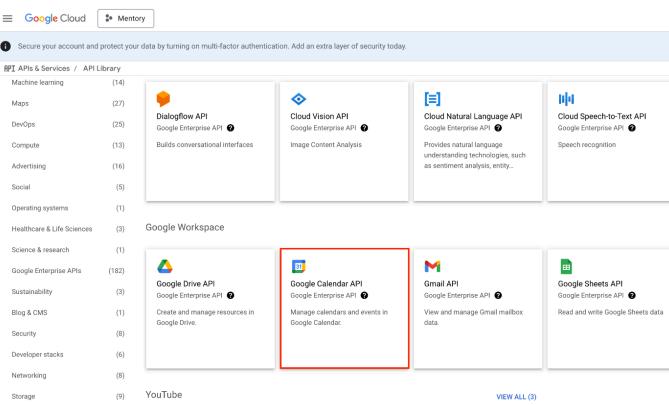
Step 2: Enable Google Calendar API

1. In the Google Cloud Console, go to the **APIs & Services** dashboard.
2. Click **+ Enable APIs and Services** at the top.



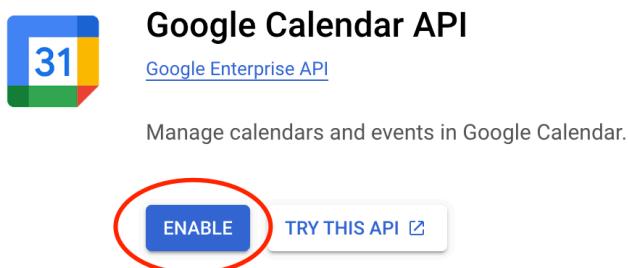
The screenshot shows the Google Cloud APIs & Services dashboard. At the top, there is a navigation bar with 'Google Cloud' and 'Mentory' tabs, and a search bar. Below the navigation bar, a message encourages enabling multi-factor authentication. The main area is divided into two sections: 'Enabled APIs & services' on the left and 'APIs & Services' on the right. The 'Enabled APIs & services' section contains links for 'Library', 'Credentials', 'OAuth consent screen', and 'Page usage agreements'. The 'APIs & Services' section has a 'Traffic' chart and a message stating 'No data is available for the selected time frame.' A red box highlights the '+ ENABLE APIs AND SERVICES' button at the top right of the main section.

3. Search for **Google Calendar API** in the library.



The screenshot shows the Google Cloud API Library. The left sidebar lists categories like Machine learning, Maps, DevOps, Compute, Advertising, Social, Operating systems, Healthcare & Life Sciences, Science & research, Google Enterprise APIs, Sustainability, Blog & CMS, Security, Developer stacks, Networking, and Storage. The main area shows cards for various APIs, including Dialogflow API, Cloud Vision API, Cloud Natural Language API, Cloud Speech-to-Text API, Google Drive API, Google Calendar API (which is highlighted with a red box), Gmail API, and Google Sheets API. A red box also highlights the 'Google Calendar API' card.

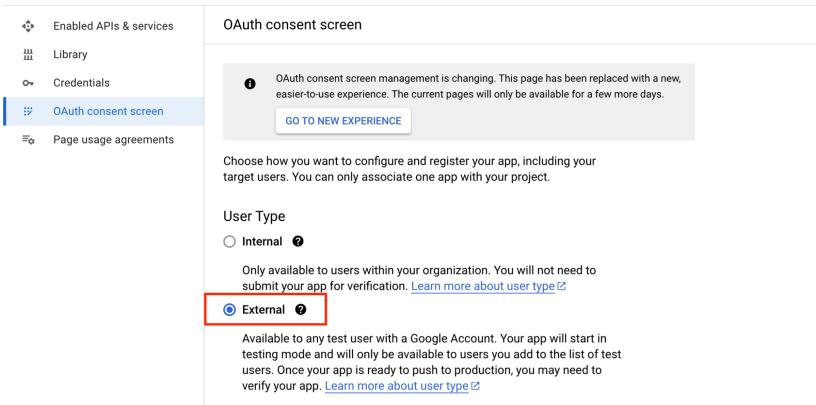
4. Select **Google Calendar API** from the results and click **Enable**.



The screenshot shows the Google Calendar API detail page. At the top, there is a logo with the number '31' and the text 'Google Calendar API' and 'Google Enterprise API'. Below the logo, a description says 'Manage calendars and events in Google Calendar.' At the bottom, there are two buttons: 'ENABLE' (which is circled in red) and 'TRY THIS API'.

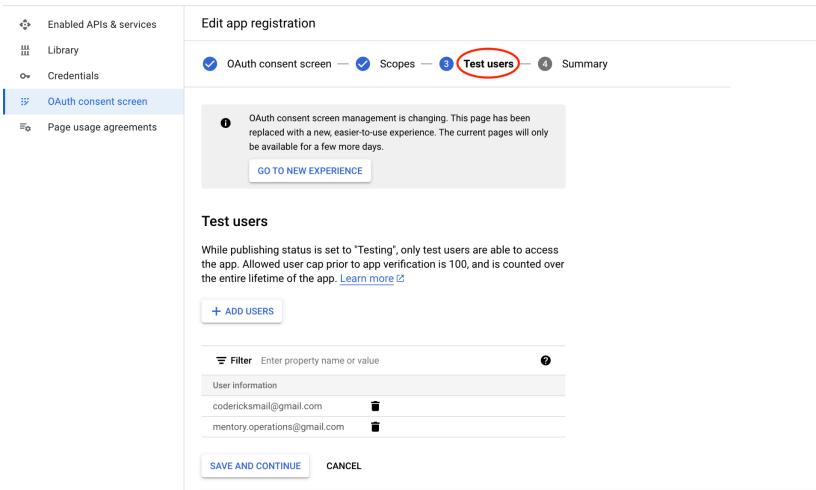
Step 3: Configure OAuth Consent Screen

1. Go to **APIs & Services > OAuth consent screen**.
2. Choose **External** (if you want it available to others) or **Internal** (for users in your organization only). Click **Create**.



The screenshot shows the 'OAuth consent screen' configuration page. On the left, a sidebar lists 'Enabled APIs & services', 'Library', 'Credentials', and 'OAuth consent screen' (which is selected and highlighted with a blue border). Below the sidebar, the main content area is titled 'OAuth consent screen'. A message box at the top states: 'OAuth consent screen management is changing. This page has been replaced with a new, easier-to-use experience. The current pages will only be available for a few more days.' with a 'GO TO NEW EXPERIENCE' button. The main content area is titled 'Choose how you want to configure and register your app, including your target users. You can only associate one app with your project.' Under 'User Type', there are two options: 'Internal' (unchecked) and 'External' (checked and highlighted with a red box). A note below says: 'Only available to users within your organization. You will not need to submit your app for verification. [Learn more about user type](#)' and '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. [Learn more about user type](#)'.

3. Fill in the necessary details: **App name**, **User support email** & **Developer email**
4. Under **Scopes**, click **Add or Remove Scopes**, and select the required scopes (this is optional).
5. Test Users: Add some test email addresses to be used for testing the Calendar API in testing mode.

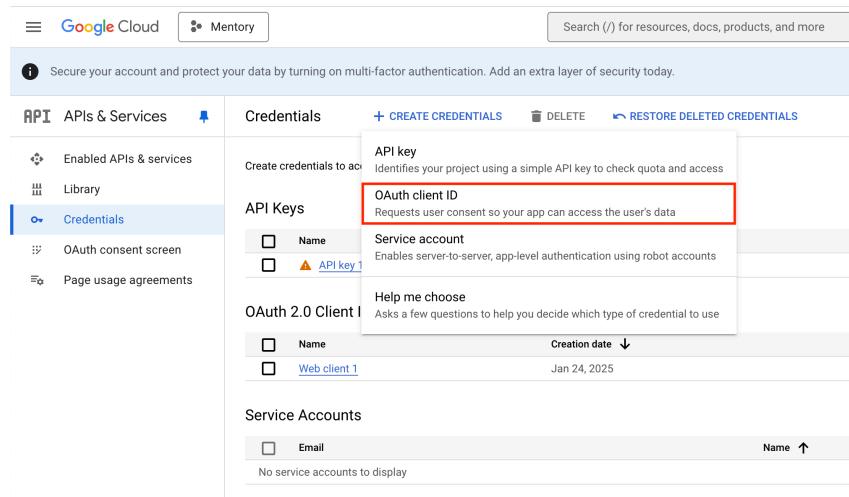


The screenshot shows the 'Edit app registration' page. The sidebar on the left is identical to the previous screenshot. The main content area is titled 'Edit app registration' and shows the navigation tabs: 'OAuth consent screen' (selected and highlighted with a blue border), 'Scopes', 'Test users' (which is highlighted with a red circle), and 'Summary'. A message box at the top is identical to the one in the previous screenshot. Below the tabs, the 'Test users' section is titled 'Test users'. It contains a note: 'While publishing status is set to "Testing", only test users are able to access the app. Allowed user cap prior to app verification is 100, and is counted over the entire lifetime of the app. [Learn more](#)'. Below this is a 'User information' table with two rows: 'codericks@gmail.com' and 'mentory.operations@gmail.com'. At the bottom are 'SAVE AND CONTINUE' and 'CANCEL' buttons.

6. **Save** your changes.

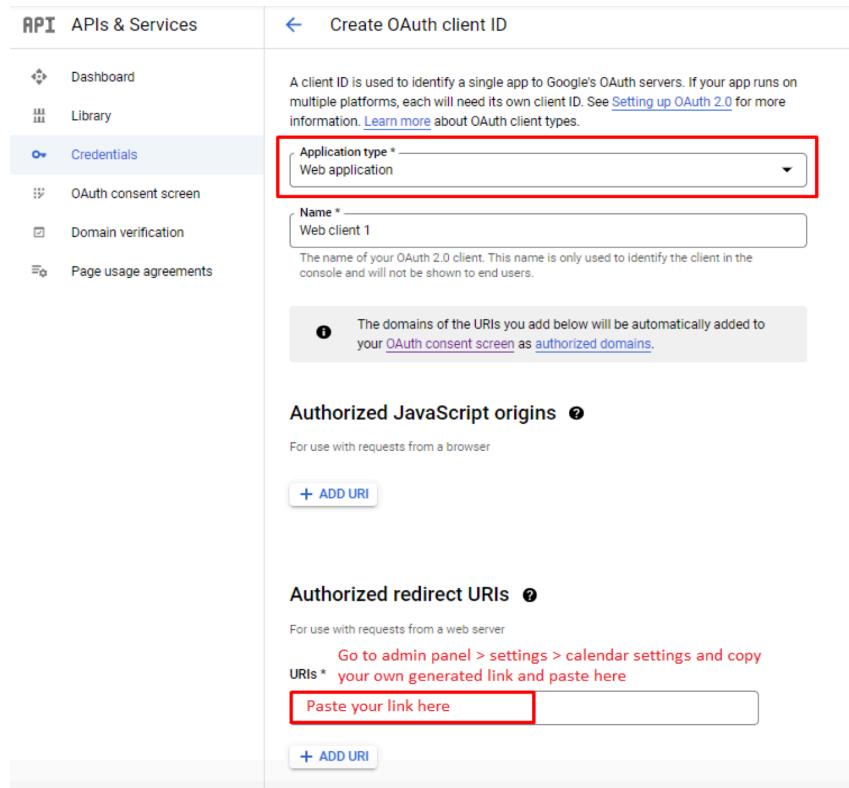
Step 4: Create Credentials

- A. Go to **APIs & Services > Credentials**.
- B. Click **+ Create Credentials** and choose **OAuth Client ID**



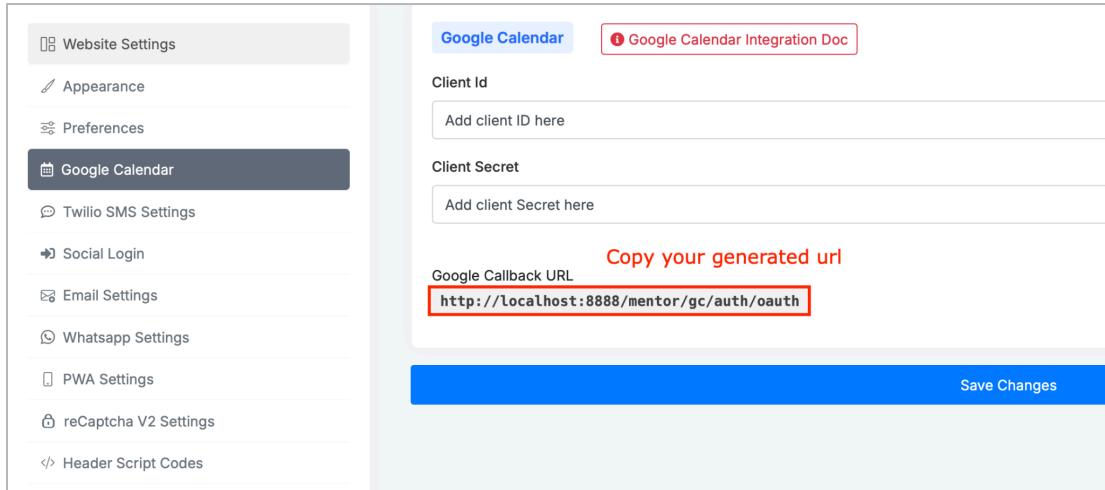
The screenshot shows the Google Cloud API & Services Credentials page. The 'OAuth client ID' option is highlighted with a red box. The page includes sections for API keys, OAuth 2.0 Client ID, and Service Accounts, with a 'Help me choose' section for deciding between them.

- C. Choose **Web application** as the application type. Add the redirect URL generated in your Admin Panel under **Settings > Calendar Settings**, and paste it into the **Authorized Redirect URIs** field.

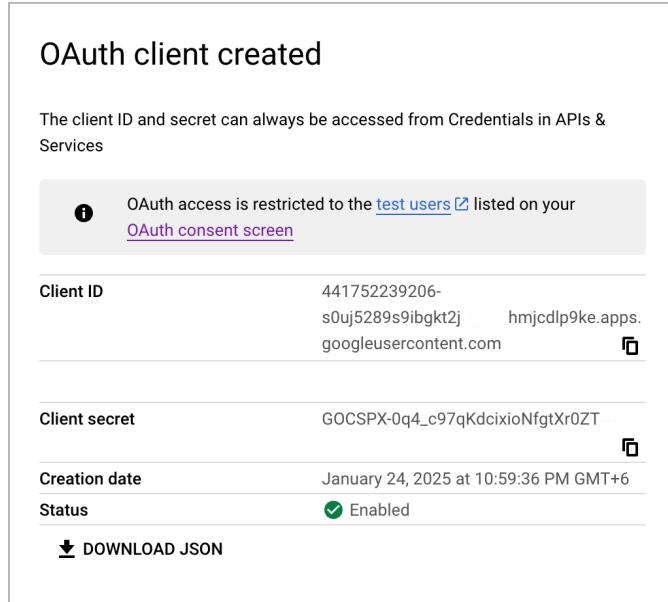


The screenshot shows the 'Create OAuth client ID' form. The 'Application type' dropdown is set to 'Web application' and is highlighted with a red box. The 'Name' field contains 'Web client 1'. Below the form, there is a note about authorized domains and a section for 'Authorized redirect URIs' with a 'Paste your link here' input field highlighted with a red box.

Here's an example callback url from our local server (**Note:** Do not use this example callback URL; use the callback URL generated for your live site based on your domain).



D. Click **Create**, then copy and save the **Client ID** and **Client Secret** to your **admin panel > settings > google calendar settings** and save the updates.



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