

---

## Google Developer Console Account Access

Ex Libris requires access to your Google Developer Console account in order for us to release the campusM apps on your behalf to the Google Play store.

---

## Create a New Account

Follow the steps on the Google Developer pages to set up a new account: [https://support.google.com/googleplay/android-developer/answer/6112435?hl=en&ref\\_topic=3450769](https://support.google.com/googleplay/android-developer/answer/6112435?hl=en&ref_topic=3450769). Once the account has been created, provide Ex Libris with access as described below.

---

### Note

You may not need to create a new developer account if your institutions already have an existing official developer account. If this is the case, you can skip this step.

---

---

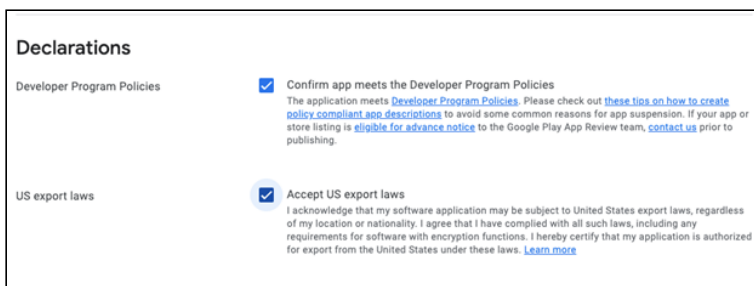
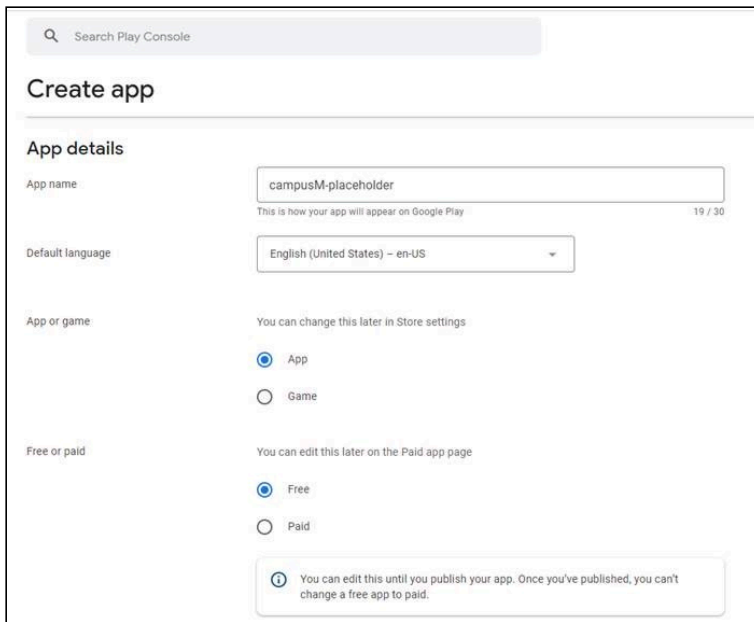
## Create a New Placeholder App

In the Google Play Console, create a new Placeholder App for campusM.

1. Visit <https://play.google.com/apps/publish> and log in to your institution's account.
2. Select **Create app**.



3. Configure the parameters as follows:
  - **App name:** campusM-placeholder
  - **App or game:** App
  - **Free or paid:** Free
  - Select all options in the **Declarations** section.



4. Select **Create App**.

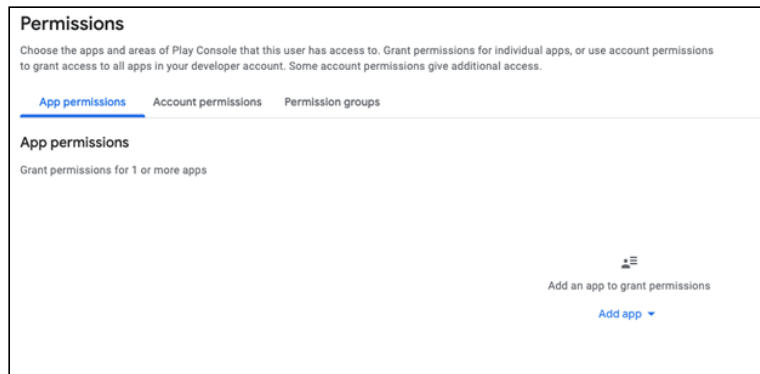
## Grant Ex Libris Access to Your Google Play Developer Console

The campusM team requires sufficient access to your Google Play Developer Console, via which your app can be managed. Below are the steps required to achieve this:

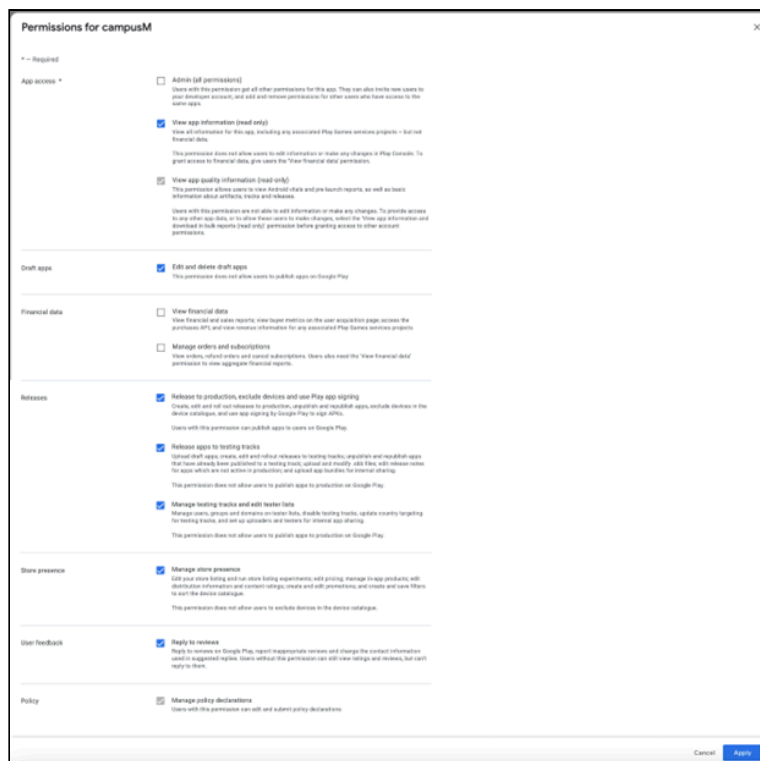
1. Visit <https://play.google.com/apps/publish> and log in to your institution's account.
2. Search for "IAM" and then select the **IAM** option.



3. Leave the **Set access expiry date** checkbox clear.
4. Under **Permissions**, select **Add app**.



5. Choose the campus-placeholder app.
6. Select **Apply**. The permissions page displays:



7. Set the permissions according to the image above.
8. Select **Apply**.
9. Select **Invite User**.

Access is granted to the Google Play developer console.

For questions about the required permissions, see our [related FAQ](#).

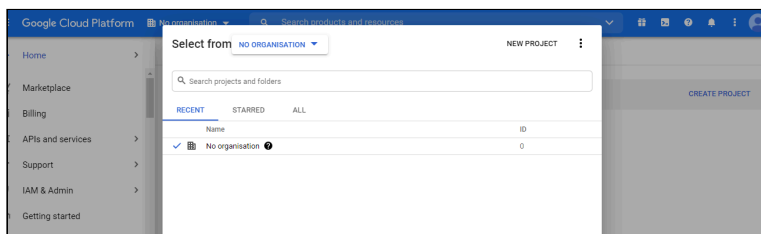
To find out more about the Google Play Console, see the [Google Play Console Help](#).

## Google Cloud Project

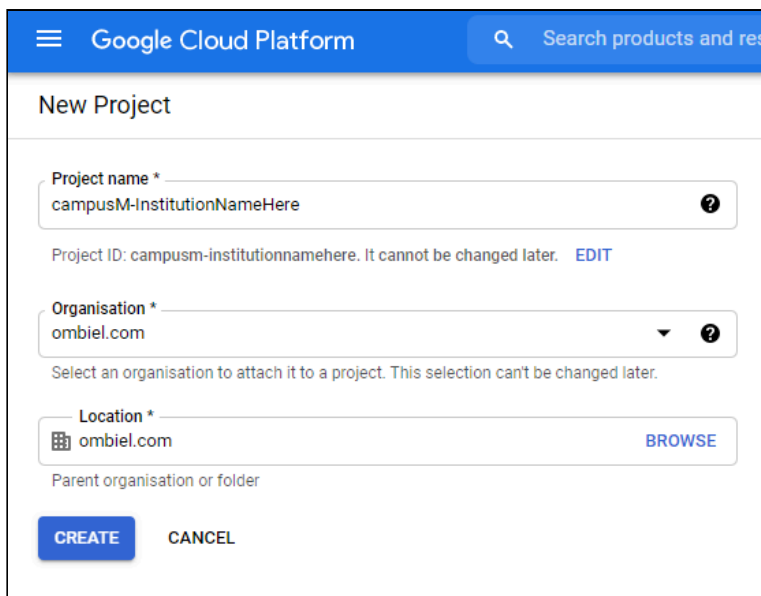
Sign into the Cloud Console at [Cloud Console Sign-In](#).

If you do not already have a Google Cloud Project, you must create a new project before granting Ex Libris access, as follows:

1. Navigate to **Select a project** in the header of the page, and in the resulting modal, select **NEW PROJECT**.



2. Enter a **Project name** e.g. *campusM-InstitutionName* and **CREATE**. Include the institution name or name abbreviation within the project name.

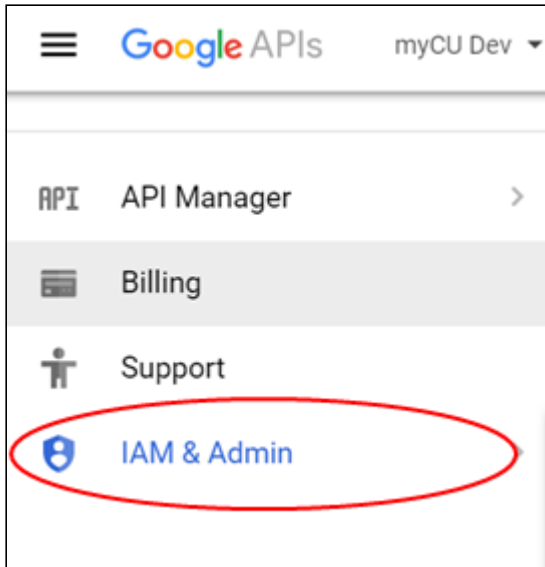


---

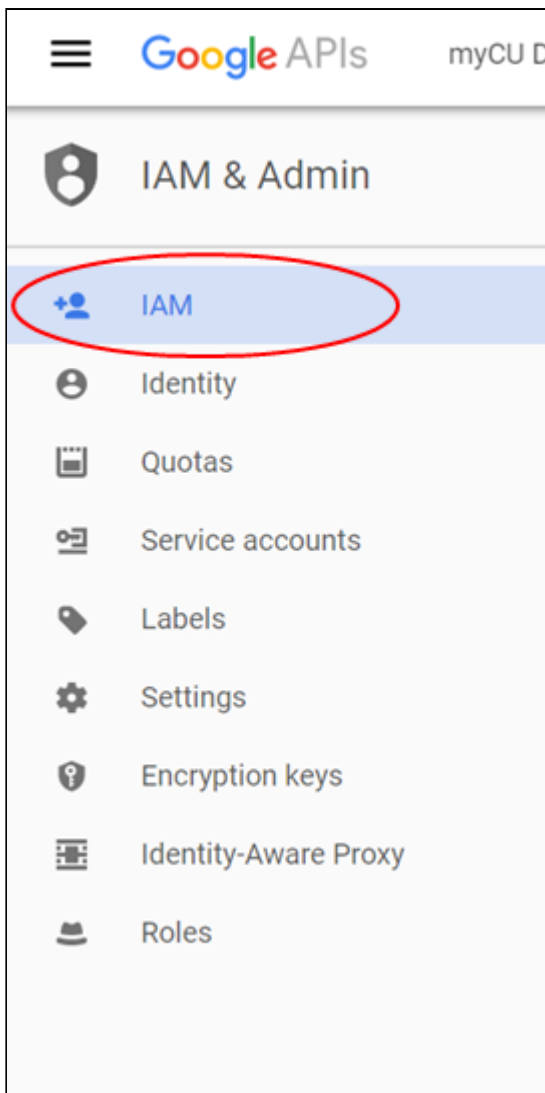
## Grant Ex Libris Access to Your Google Cloud Project

The campusM team requires access to your Google Cloud project. To do this, perform the following steps:

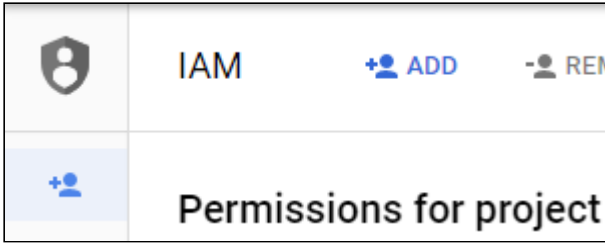
1. Visit <https://console.developers.google.com/apis/dashboard> and log into your institution's account.
2. Click to expand the menu, and then select **IAM & Admin**.



3. Navigate to IAM.



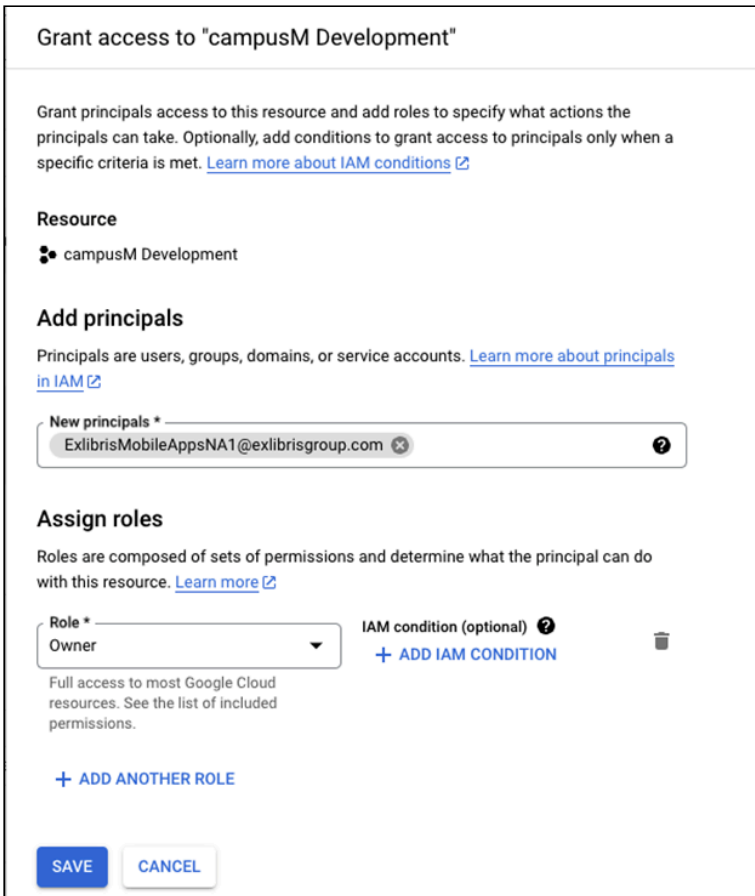
4. Select **Grant Access**.



5. In the **Add Principals** field, enter the **campusM** email address for your region:

Region	campusM Store Authorized User (Email)
APAC	<a href="mailto:ExlibrisMobileAppsAPAC1@exlibrisgroup.com">ExlibrisMobileAppsAPAC1@exlibrisgroup.com</a>
EMEA	<a href="mailto:ExlibrisMobileAppsEMEA1@exlibrisgroup.com">ExlibrisMobileAppsEMEA1@exlibrisgroup.com</a>
NA	<a href="mailto:ExlibrisMobileAppsNA1@exlibrisgroup.com">ExlibrisMobileAppsNA1@exlibrisgroup.com</a>

6. Configure the role to be **Project > Owner**.



7. Select **Save**.

Find out more about the Google Play Console by following this link: <https://support.google.com/googleplay/android-developer>.

---

## Why Does Ex Libris Require Access to Your Google Cloud Project?

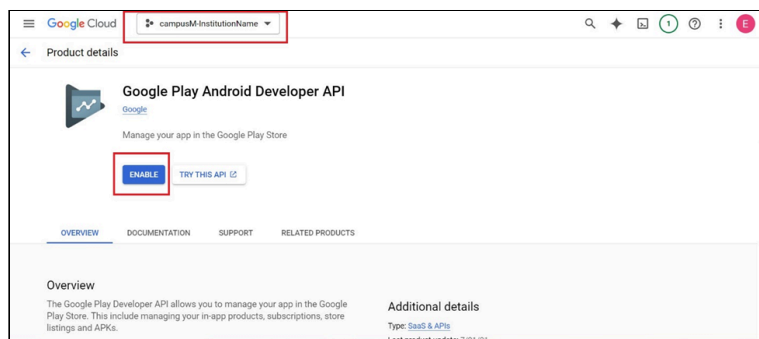
- Google Maps - Our native Android app integrates with the Google Maps API which requires access to Google Cloud.
- Push notifications - Our native Android app requires access to Google Cloud to enable GCM (Google Cloud messaging).

---

## Providing Ex Libris with Your Google Developer Service Account

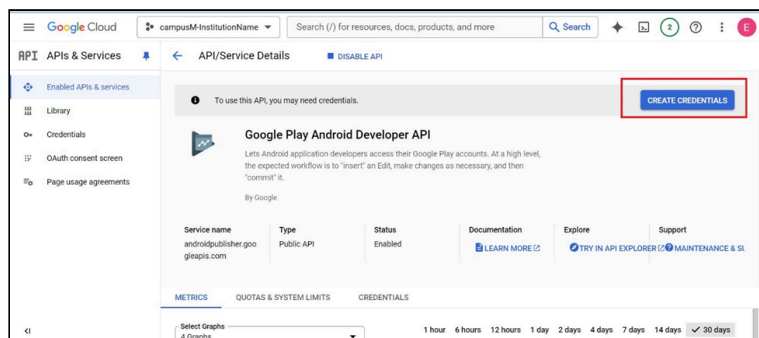
The Service Account is different from your Google Developer Account. It is created via the Google Cloud console, when logged in to your Google Developer Account.

1. Log in to your Google Cloud Console (<https://console.cloud.google.com>).
2. Enable the Google Play Android Developer API for the cloud project you've just created.



You are automatically redirected to the API/Service Details page.

3. Select the **Create Credentials** button on the far right.



4. In the form that appears, under **Select an API**, select **Google Play Android Developer API**, and then select **Application data** as your data preference.

**1 Credential Type**

**Which API are you using?**  
Different APIs use different auth platforms and some credentials can be restricted to only call certain APIs.

Select an API \*  
Google Play Android Developer API

**What data will you be accessing? \***  
Different credentials are required to authorize access depending on the type of data that you request. [Learn more](#)

User data  
Data belonging to a Google user, like their email address or age. User consent required. [This will create an OAuth client](#)

**Application data**  
Data belonging to your own application, such as your app's Cloud Firestore backend. This will create a service account.

**NEXT**

**2 Your Credentials**

**DONE** CANCEL

5. Select **Next**. The **Create service account** form opens.
6. Enter your preferred **Display Name** for the Service Account and add a brief description.

**1 Service account details**

Service account name  
campusM

Display name for this service account

Service account ID \*  
campusm

Email address: campusm@campusm-institutionname-432613.iam.gserviceaccount.com

Service account description  
campusM

Describe what this service account will do

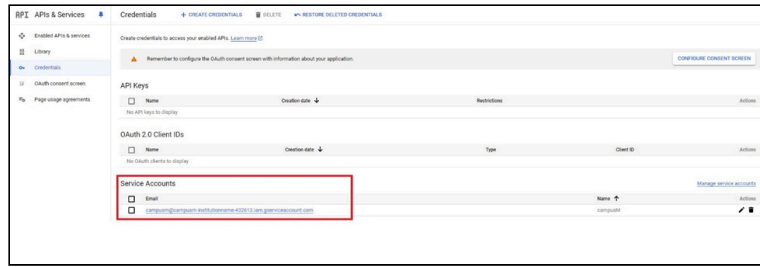
**CREATE AND CONTINUE**

**2 Grant this service account access to project (optional)**

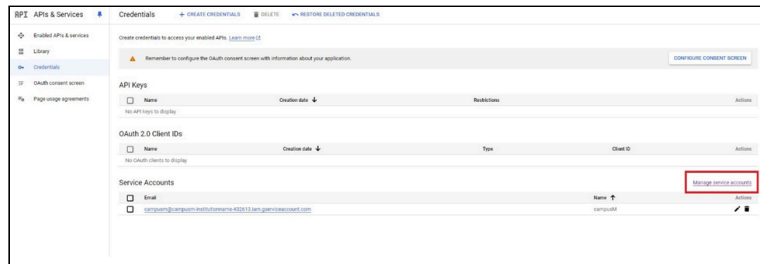
**3 Grant users access to this service account (optional)**

**DONE** CANCEL

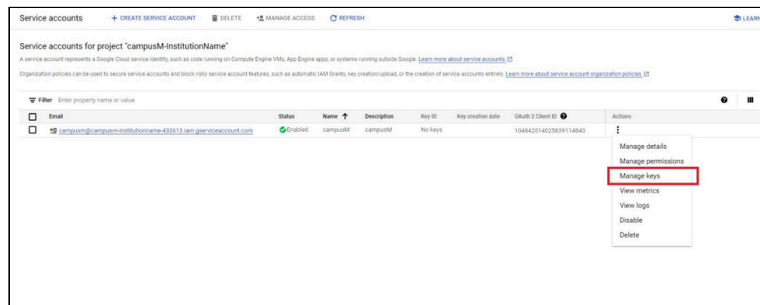
7. Select **Create and Continue**.
8. In the second step, assign a role to your account by clicking on the **Select a Role** dropdown menu. Add the following permissions:
  - **Firebase Admin**
  - **API Keys Admin**
  - **Service Usage Admin**
  - **Owner**
9. The IAM Condition and third step are not required, so you can skip them by clicking **Continue** and then **Done**. This brings you back to the main screen.
10. In the left panel, select the **Credentials** menu. You should now see your new Service Account listed, identified by its unique email address.



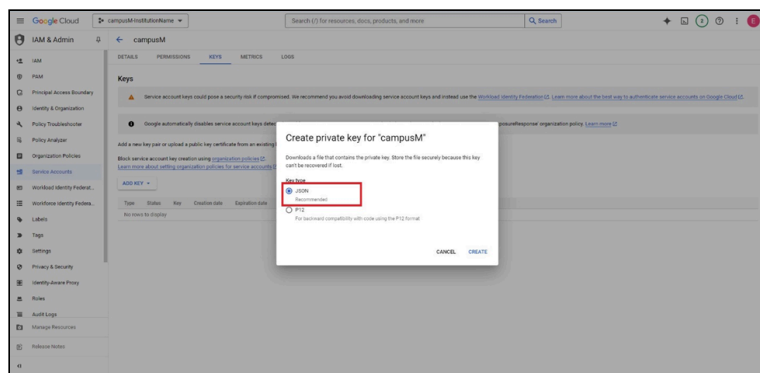
11. Copy the Service Account email for a later step.
12. In the Service Accounts section, select **Manage Service Accounts**.



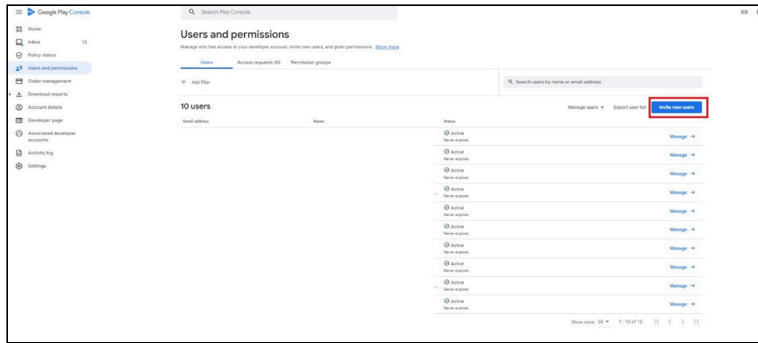
13. When the context menu appears, select **Manage Keys**.



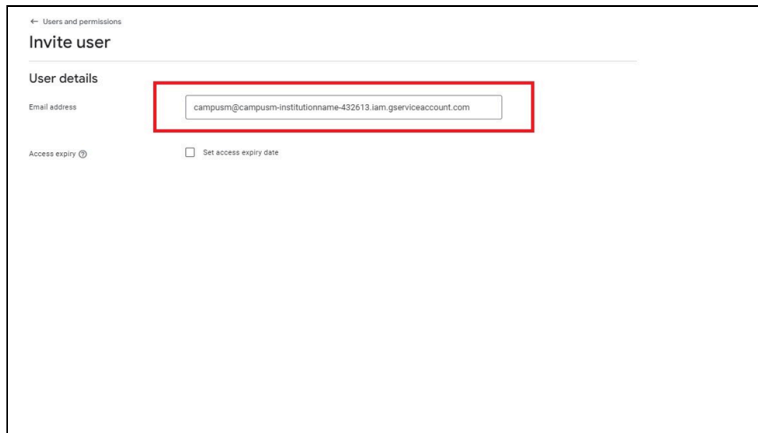
14. In the next window, select the **Add Key** button and choose **Create New Key** from the context menu.
15. In the popup window, select the recommended **JSON** key format and then select **Create**.



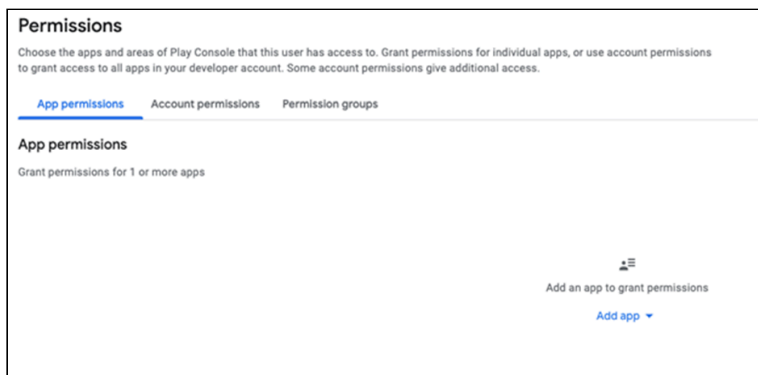
16. Download the JSON file and provide it to campusM.
17. Go to the Google Play Console (<https://play.google.com/console>). On the **Users and Permissions** page, select **Invite New Users**.



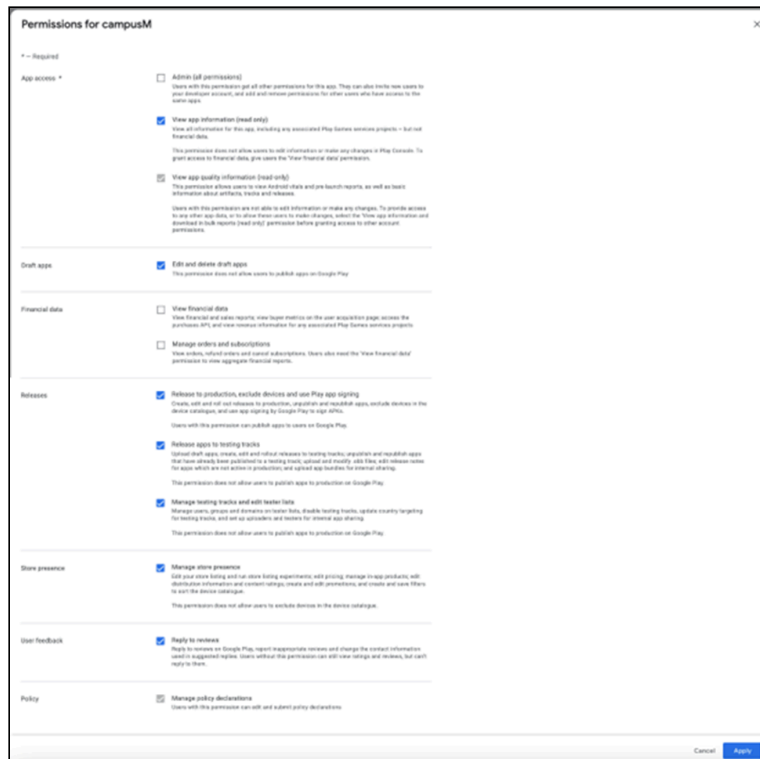
- Paste the Service Account email address you copied earlier into the provided space. Leave the **Set access expiry date** checkbox unchecked.



- Under **Permissions**, select **Add app**.



- Choose the **campus-placeholder** app.
- Select **Apply**. The permissions page displays:



22. Set the permissions according to the image above.
23. Select **Apply**.
24. Select **Invite User**.

Congratulations! You've successfully completed the process. Your Service Account is now active and operational. Don't forget to provide campusM with the JSON file you downloaded.

## App Store Listing – Containing or Displaying Ads

On the Google Play Store, Android users will sometimes identify that customer's apps have the label "Contains Ads." campusM never shows advertisements within the product, and have no intention of doing so. However, this is necessitated by Google's policy towards third party content and what it defines as an advertisement.

In areas wherein Google lacks formal documentation, conventions are formed around what Google will reject people for. We have had messages from Google with content that can be summarized as follows:

*By January 11, 2016, you must sign in to the Play Developer Console and declare whether your apps contain ads (such as ads delivered through third party ad networks, display ads, native ads, and/or banner ads). After this date, the ads declaration will be required to make any updates to your apps. Please note that misrepresenting ad presence violates Google Play Developer Program Policies and may lead to suspension.*

Here are a few examples from Google of when you should answer "yes" to having ads for purposes of the "Ads" label:

- Native ads: My app includes ads natively, indistinguishable from other content (e.g. sponsored articles, ads within a feed, etc).
- House ads: My app renders a small banner to promote my other apps.
- Banner and interstitial: My app integrates with an Ad SDK to show banners and/or interstitials.

This includes things such as Sports events promotion, promoting other Institutional products or portals, links to third party

content such as the Olympics, etc. As such, by default, we include "contains ads."

Removing this label risks not only having an app pulled from the store, but having all of the apps on your account pulled when the account is suspended on Google Play. As such, we strongly recommend that customers not remove this or request Ex Libris to do so for them.

---

## Billing Requirement for Maps API

Google are rebranding their mapping technologies and adjusting the pricing structure for their use. As a result, campusM customers will need enable billing with a credit card against their Google Developer account by the 11th June 2018.

Google's pricing table (<https://cloud.google.com/maps-platform/pricing/sheet/>) makes it clear that campusM customers will not be charged for the mobile maps that are used as part of our native apps. So there is no expectation that you will be required to pay anything additional to Google for campusM as a result of their new pricing structure. However, they are explicit about the need for billing against the developer account to keep it in good standing and enable the app to be available and up-gradable through the Play Store.

Google have provided a transition tool that will guide you through adding the required billing details to your account: <https://mapsplatformtransition.withgoogle.com/>

You will need your developer account login details and a payment method.

Note that all of the campusM mobile apps have a Google Maps Platform API key.

To find out more about the changes Google are making to their mapping platform please visit: <https://cloud.google.com/maps-platform/user-guide/>.