Google+ Platform
- Quickstart
- PhotoHunt Sample
- API Reference

- Features

- Android
  - Getting started
  - Sign in
  - Share
  - +1 button
  - Over-the-air installs
  - App activities
  - People and profiles
  - Google Play services
- iOS
- Websites
- Hangout Apps

Downloads
Best Practices
Branding Guidelines

- Community & Support

Release Notes
Google APIs Console

Developer Policies
Terms of Service

Manage app activities in Google

After the user signs in using the Google+ Sign-In button, you can write app activities to Google on behalf of your user with the moments methods. App activities reflect a variety of user actions including check-ins, reviews, commenting on articles, and more. Users control the visibility of their app activities on Google using circles. Your app use the type of activity that best reflects your content or the user's action.

> **Note:** The moment methods do not write directly to a user's Google+ stream. They instead write to a user's profile, and are not necessarily viewable by others depending on the user's preferred sharing settings.

Before you begin

You must create a Google APIs Console project and initialize the PlusClient object.

Write an app activity to Google

When a user performs an action within your app, your app can write an app activity to the authenticated user's Google account using the `PlusClient.writeMoment` method. The target of your app activity might be a comment, a document, a song, or something else that the user might interact with in your app.

Writing an app activity with a target URL

When the target of your app activity has a URL associated with it, you will provide that URL as part of the `Moment` object and when the activity is written to Google, a snippet is generated from that target URL. The target URL should have a schema.org type associated with the page that matches the moment type that you specify.

1. Construct a `Moment` object representing a review activity.

```
ItemScope rating = new ItemScope.Builder()
    .setType("http://schema.org/Rating")
    .setRatingValue("100")
    .setBestRating("100")
    .setWorstRating("0")
    .build();

ItemScope result = new ItemScope.Builder()
    .setType("http://schema.org/Review")
    .setName("A Humble Review of Widget")
    .setUrl("https://developers.google.com/+/web/snippet/examples/review")
    .setDescription("It is amazingly effective")
    .setReviewRating(rating)
    .build();

ItemScope target = new ItemScope.Builder()
    .setUrl(targetUrl)
    .build();

Moment moment = new Moment.Builder()
    .setType("http://schemas.google.com/ReviewActivity")
    .setTarget(target)
    .setResult(result)
    .build();
```

2. Use the `PlusClient` object to write a moment to the current user's account. This must be done after the

PlusClient is connected, usually in the onConnected handler.

```
if (mPlusClient.isConnected()) {
    mPlusClient.writeMoment(moment);
}
```

Google may obtain an image by crawling your target URL for moments that do not specify an image field. All content associated with your target URL that Google obtains and uses in connection with a moment is considered content that you submit to Google under Section 5 of the Google APIs Terms of Service.

Writing an app activity without a target URL

When the target of your app activity does not have a publically accessible URL associated with it, you can specify the fields for populating the snippet. For mobile apps that do not have an associated web platform, this scenario is common.

1. Construct a Moment object representing a listen activity.

```
ItemScope target = new ItemScope.Builder()
    .setId("myuniqueidforthissong")
    .setName("When Johnny Comes Marching Home")
    .setDescription("A song about missing one's family members fighting in the Americ
    .setImage("http://example.com/images/albumThumb.png")
    .setType("http://schema.org/MusicRecording")
    .build();

Moment moment = new Moment.Builder()
    .setType("http://schemas.google.com/ListenActivity")
    .setTarget(target)
    .build();
```

   For a full list of fields that you can specify for the target of the app activity, see the ItemScope.Builder() methods.

2. Use the PlusClient object to write a moment to the current user's account. This must be done after the PlusClient is connected, usually in the onConnected handler.

```
if (mPlusClient.isConnected()) {
    mPlusClient.writeMoment(moment);
}
```

The PlusClient.writeMoment method is a native API binding around the moments.insert method.

List app activities

Use the PlusClient.loadMoments method to make an asynchronous request for the user's moments written by your app. The MomentBuffer offers a cursor-like interface for traversing through the loaded moments. You must invoke the MomentBuffer.close method when you are done loading data from the MomentBuffer.

The MomentBuffer.copyMoment method returns a new Moment object based on the buffer's current position.

```
mPlusClient.loadMoments(new PlusClient.OnMomentsLoadedListener() {
    @Override
    public void onMomentsLoaded(ConnectionResult status, MomentBuffer momentBuffer,
            String nextPageToken, String updated) {
        if (status.getErrorCode() == ConnectionResult.SUCCESS) {
            try {
                Log.d(TAG, "Moment Name: " + momentBuffer.get(i).getTarget().getName();
            } finally {
                momentBuffer.close();
            }
        } else {
            Log.e(TAG, "Error when loading moments: " + status.getErrorCode());
        }
    }
});
```

The PlusClient.loadMoment method is a native API binding around the moments.list method.

Remove an app activity from Google

Use the `PlusClient.removeMoment` method to remove a moment from Google on behalf of the signed-in user. This method requires that you pass an identifier for the moment to be removed, which is available by calling `Moment.getId()`.

```
if (mPlusClient.isConnected() && moment != null) {
    mPlusClient.removeMoment(moment.getId());
}
```

The `PlusClient.removeMoment` method is a native API binding around the moments.remove method.

Next steps

- Create interactive posts
- Fetch people information

Last updated April 29, 2013.