## PLAY BILLING

**Tổng quan về hệ thống thanh toán của Google Play**

Hệ thống thanh toán của Google Play là dịch vụ cho phép bạn bán các sản phẩm và nội dung kỹ thuật số trong ứng dụng Android của mình.

Bạn có thể sử dụng hệ thống thanh toán của Google Play để bán các loại nội dung kỹ thuật số sau:

* Sản phẩm một lần: Sản phẩm một lần là nội dung mà người dùng có thể mua với một khoản phí duy nhất, không định kỳ cho hình thức thanh toán của người dùng.

Sản phẩm một lần có thể tiêu hao hoặc không tiêu hao:

* + Sản phẩm tiêu hao là sản phẩm mà người dùng tiêu thụ để nhận nội dung trong ứng dụng, chẳng hạn như đơn vị tiền tệ trong trò chơi. Khi người dùng tiêu thụ sản phẩm, ứng dụng của bạn phân phối nội dung được liên kết và người dùng có thể mua lại mặt hàng.
  + Sản phẩm không tiêu hao là sản phẩm chỉ được mua một lần để cung cấp lợi ích vĩnh viễn. Ví dụ bao gồm nâng cấp cao cấp và gói cấp độ.
* [Đăng ký](https://developer.android.com/google/play/billing/subs): Đăng ký là một sản phẩm cung cấp quyền truy cập vào nội dung trên cơ sở định kỳ. Gói đăng ký tự động gia hạn cho đến khi bị hủy. Ví dụ về đăng ký bao gồm quyền truy cập vào các tạp chí trực tuyến và dịch vụ phát nhạc trực tuyến.

Google Play Console mang lại rất nhiều sự linh hoạt khi tạo sản phẩm đăng ký. Ví dụ: bạn có thể đặt thời hạn thanh toán, cung cấp bản dùng thử miễn phí, cung cấp giá giới thiệu, cung cấp thời gian gia hạn khi thanh toán không thành công và cho phép người dùng tạm dừng đăng ký của họ thay thế cho việc hủy. Để biết thêm thông tin và danh sách đầy đủ các tính năng đăng ký, hãy xem Triển [khai các tính năng dành riêng cho đăng ký](https://developer.android.com/google/play/billing/subs).

# Thuật ngữ liên quan đến thanh toán trên Google Play

Chủ đề này liệt kê và xác định các thuật ngữ và cụm từ chính có liên quan đến hệ thống thanh toán của Google Play.

## Kiểu nội dung

Bạn có thể sử dụng Google Play để bán các loại nội dung kỹ thuật số sau:

* Sản phẩm một lần: Sản phẩm một lần là nội dung mà người dùng có thể mua với một khoản phí duy nhất, không định kỳ cho hình thức thanh toán của người dùng. Ví dụ về các sản phẩm một lần bao gồm các cấp độ trò chơi bổ sung, hộp loot cao cấp và các tệp phương tiện. Sản phẩm một lần là tiêu hao hoặc không tiêu hao:
  + Sản phẩm tiêuhao: Một sản phẩm tiêu hao là sản phẩm mà người dùng tiêu thụ để nhận nội dung trong trò chơi. Khi người dùng tiêu thụ sản phẩm của bạn, bạn phân phối các lợi ích hoặc tác dụng liên quan trong một quá trình được gọi là cung cấp. Ví dụ: người dùng có thể mua một sản phẩm tiền tệ trong trò chơi, chẳng hạn như một chồng tiền xu. Khi người dùng tiêu thụ sản phẩm, ứng dụng của bạn sau đó phân phối một số lượng tiền cố định cho người dùng và sản phẩm tiền xu đã mua không còn nữa. Các sản phẩm tiêu hao có thể được mua nhiều lần.
  + Sản phẩm không tiêuhao: Sản phẩm không tiêu hao là sản phẩm được mua một lần và mang lại lợi ích vĩnh viễn. Sau khi mua, các sản phẩm này được liên kết vĩnh viễn với Tài khoản Google của người dùng. Ví dụ về các sản phẩm không tiêu hao bao gồm nâng cấp cao cấp và gói cấp độ.

[Google Play Console đề cập đến](https://play.google.com/console/) các sản phẩm một lần là sản phẩm được quảnlý và Thư viện thanh toán của Google Play gọi chúng là sản phẩm "INAPP".

* [Đăng ký](https://developer.android.com/google/play/billing/subs): Đăng ký là một sản phẩm cung cấp quyền truy cập vào nội dung trên cơ sở định kỳ. Khi thời gian đăng ký kết thúc, đăng ký sẽ tự động gia hạn và người dùng phải chịu thêm một khoản phí cho hình thức thanh toán của họ. Đăng ký được gia hạn vô thời hạn cho đến khi chúng bị hủy. Ví dụ về đăng ký bao gồm quyền truy cập vào các tạp chí trực tuyến và dịch vụ phát nhạc trực tuyến.

Google Play Console mang lại rất nhiều sự linh hoạt khi tạo sản phẩm đăng ký. Ví dụ: bạn có thể đặt thời hạn thanh toán, cung cấp bản dùng thử miễn phí, cung cấp giá giới thiệu, cung cấp thời gian gia hạn khi thanh toán không thành công và cho phép người dùng tạm dừng đăng ký của họ thay thế cho việc hủy. Để biết thêm thông tin và danh sách đầy đủ các tính năng đăng ký, hãy xem Triển [khai các tính năng dành riêng cho đăng ký](https://developer.android.com/google/play/billing/subs).

Bạn có thể cung cấp nhiều gói đăng ký trong cùng một ứng dụng. Tuy nhiên, lưu ý rằng tất cả các gói đăng ký phải được định giá trong phạm vi giá được chấp nhận và không thể công bố đăng ký.

Thư viện thanh toán của Google Play gọi gói đăng ký là "SUBS".

## Công nghệ và khái niệm

Phần này liệt kê và mô tả các công nghệ và khái niệm cấp cao mà bạn có thể gặp phải khi tích hợp hệ thống thanh toán của Google Play vào ứng dụng của mình. Hãy chắc chắn làm quen với từng mục trong phần này trước khi tiếp tục.

### Công nghệ

* [Google Play](https://play.google.com/): Google Play là cửa hàng trực tuyến nơi người dùng có thể tải xuống ứng dụng và các sản phẩm kỹ thuật số khác.
* [Google Play Console](https://play.google.com/console): Google Play Console cung cấp giao diện nơi bạn có thể phát hành ứng dụng lên Google Play. Google Play Console cũng hiển thị thông tin chi tiết về ứng dụng của bạn, bao gồm mọi sản phẩm hoặc nội dung bạn bán bằng Google Play.
* [Google API Console](https://console.developers.google.com/): Google API Console quản lý API phụ trợ, chẳng hạn như API nhà phát triển trên Google Play.
* [Thư viện thanh toán của Google Play](https://developer.android.com/google/play/billing/integrate): Thư viện thanh toán trên Google Play là API mà bạn có thể sử dụng để tích hợp hệ thống thanh toán của Google Play vào ứng dụng của mình.
* [API nhà phát triển trên Google Play](https://developers.google.com/android-publisher/): API nhà phát triển trên Google Play là API REST mà bạn có thể sử dụng để xử lý các tác vụ xuất bản và quản lý ứng dụng theo lập trình.
* [Cloud Pub/Sub](https://cloud.google.com/pubsub/): Cloud Pub/Sub là một dịch vụ nhắn tin thời gian thực được quản lý toàn phần cho phép bạn gửi và nhận tin nhắn giữa các ứng dụng độc lập. Google Play sử dụng Cloud Pub/Sub để cung cấp thông báo dành cho nhà phát triển theo thời gian thực. Để sử [dụng Cloud Pub/Sub](https://cloud.google.com/pubsub/), bạn phải có một dự ántrên Google Cloud [Platform (GCP)](https://cloud.google.com/) với Cloud Pub/Sub API được bật. Nếu bạn không quen thuộc với GCP và Cloud Pub/Sub, hãy xem hướng dẫn Bắt đầu [nhanh](https://cloud.google.com/pubsub/docs/quickstart-console).
* [Thông báo dành cho nhà phát triển theo thời](https://developer.android.com/google/play/billing/getting-ready#configure-rtdn)gian thực: Thông báo dành cho nhà phát triển theo thời gian thực cho phép bạn theo dõi các thay đổi trạng thái đối với đăng ký do Google Play quản lý trong thời gian thực bằng cách [tận dụng Cloud Pub / Sub](https://cloud.google.com/pubsub).
* Máy chủ phụ trợ bảo mật : Là một phần của việc tích hợp hệ thống thanh toán của Google Play vào ứng dụng của bạn, chúng tôi khuyên bạn nên sử dụng máy chủ phụ trợ an toàn để triển khai các tác vụ liên quan đến thanhtoán như xác minh mua hàng, tính năng dành riêng cho đăng ký và xử lý thông báo của nhà phát triển theo thời gian thực.
* Ứng dụng Cửa hàng Google Play: Ứng dụng Cửa hàng Google Play quản lý tất cả các hoạt động liên quan đến Google Play. Tất cả các yêu cầu do ứng dụng của bạn thực hiện đều do ứng dụng Cửa hàng Google Play xử lý.

### Khái niệm

* Luồng: Luồng hiển thị các bước điển hình liên quan đến tác vụ liên quan đến thanh toán. Ví dụ: quy trình mua hàng trình bày các bước liên quan khi người dùng mua sản phẩm của bạn. Luồng đăng ký có thể cho thấy cách chuyển đổi đăng ký giữa các tiểu bang.
* Quyềnlợi: Khi người dùng mua một sản phẩm trong ứng dụng, họ sẽ được hưởng sản phẩm đó trong ứng dụng của bạn. Đối với các sản phẩm một lần, điều này có nghĩa là người dùng bây giờ sẽ có quyền truy cập vĩnh viễn vào sản phẩm. Đối với đăng ký, điều này có nghĩa là người dùng nên có quyền truy cập trong khi đăng ký đang hoạt động.
* Mã sảnphẩm: Mã sản phẩm là ID của một loại sản phẩm cụ thể.
* Mã thông báomua hàng : Mã thông báo mua hàng là một chuỗi đại diện cho quyền lợi của người mua đối với một sản phẩm trên Google Play. Nó chỉ ra rằng một người dùng Google đã trả tiền cho một sản phẩm cụ thể.
* ID đơnhàng : ID đơn hàng là chuỗi đại diện cho giao dịch tài chính trên Google Play. ID đơn hàng được tạo mỗi khi giao dịch tài chính xảy ra. Chuỗi này được bao gồm trong biên lai được gửi qua email cho người mua. Bạn có thể sử dụng MÃ đơn hàng để quản lý khoản hoàn tiền trong mục Quản lý đơn hàng của Google Play Console. ID đơn hàng cũng được sử dụng trong báo cáo bán hàng và thanh toán.

# Chuẩn bị sẵn sàng

Chủ đề này liệt kê và mô tả các bước thiết lập bạn cần thực hiện trước khi có thể bán sản phẩm trong ứng dụng của mình. Ở cấp độ cao, thiết lập này bao gồm tạo tài khoản nhà phát triển, tạo và định cấu hình sản phẩm bạn muốn bán cũng như bật và định cấu hình API mà bạn sử dụng để bán và quản lý sản phẩm. Chủ đề này cũng mô tả cách định cấu hình thông báo của nhà phát triển theo thời gian thực sẽ được thông báo bất cứ khi nào trạng thái của sản phẩm thay đổi.

## Thiết lập tài khoản nhà phát triển trên Google Play

Để xuất bản ứng dụng và trò chơi của bạn trên Google Play, hãy [sử dụng Google Play Console](https://developer.android.com/distribute/console). Bạn cũng sử dụng Google Play Console để quản lý các sản phẩm và tùy chọn cài đặt liên quan đến thanh toán.

Để truy cập Vào Google Play Console, bạn cần thiết [lập Tài khoản nhà phát triển trên Google Play](https://support.google.com/googleplay/android-developer/answer/6112435).

Để bán ứng dụng phải trả phí và giao dịch mua hàng trong ứng dụng trên Google Play, bạn cũng phải thiết lập hồ sơ trong [Trung tâm thanh toán Google](https://pay.google.com/) rồi liên kết hồ sơ đó với tài khoản nhà phát triển trên Google Play của mình. Để tìm hiểu cách liên kết hồ sơ của bạn với tài khoản của bạn hoặc để tìm hiểu cách kiểm tra xem bạn đã có tài khoản và hồ sơ được liên kết chưa, hãy xem Liên kết tài khoản nhà phát triển trên Google Play với hồ sơ thanh toán của [bạn](https://support.google.com/googleplay/android-developer/answer/3092739).

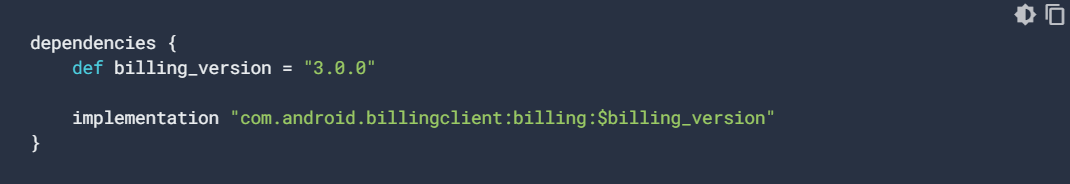
## Bật các tính năng liên quan đến thanh toán trong Google Play Console

Sau khi thiết lập tài khoản nhà phát triển, bạn phải xuất bản phiên bản ứng dụng bao gồm Thư viện thanh toán của Google Play. Bước này là bắt buộc để bật các tính năng liên quan đến thanh toán trong Google Play Console, chẳng hạn như định cấu hình các sản phẩm bạn muốn bán.

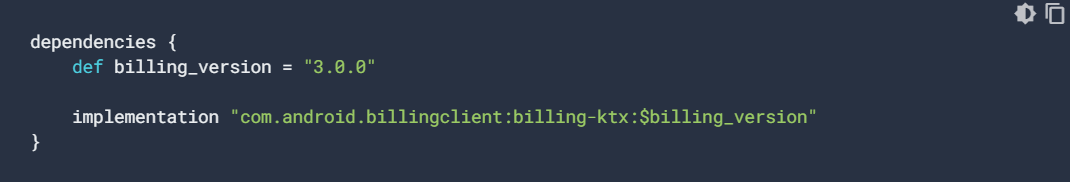
### Thêm phụ thuộc thư viện

Để tích hợp hệ thống thanh toán của Google Play, trước tiên hãy thêm phụ thuộc vào Thư viện thanh toán của Google Play trong ứng dụng của bạn. Thư viện này cung cấp quyền truy cập vào API Android kết nối bạn với Google Play. Từ đó, bạn có thể truy cập thông tin mua hàng, truy vấn để cập nhật về giao dịch mua, nhắc người dùng mua hàng mới, v.v.

Thư viện thanh toán Google Play có sẵn [từ kho lưu trữ Maven của Google](https://developer.android.com/studio/build/dependencies#google-maven). Thêm sự phụ thuộc vào tệp build.gradle của ứng dụng như được hiển thị:



Nếu bạn đang sử dụng Kotlin, mô-đun KTX của Thư viện thanh toán Play chứa tiện ích mở rộng Kotlin và hỗ trợ coroutines cho phép bạn viết Kotlin thành ngữ khi sử dụng hệ thống thanh toán của Google Play. Để bao gồm các tiện ích mở rộng này trong dự án của bạn, hãy thêm phụ thuộc sau vào tệp build.gradle của ứng dụng như được hiển thị:



### Tải ứng dụng lên

Sau khi bạn đã thêm thư viện vào ứng dụng, hãy xây dựng và phát hành ứng dụng của bạn. Đối với bước này, tạo [ứng dụng của bạn và](https://support.google.com/googleplay/android-developer/answer/113469) sau đó phát hành lên bất kỳ bản nhạc nào, bao gồm cả rãnh [kiểm tra nội bộ](https://support.google.com/googleplay/android-developer/answer/3131213).

## Tạo và định cấu hình sản phẩm

Sau khi bật tính năng Lập hóa đơn trên Google Play cho ứng dụng, bạn cần định cấu hình sản phẩm để bán.

Các bước tạo sản phẩm và đăng ký một lần tương tự nhau. Đối với mỗi sản phẩm, bạn cần cung cấp id sản phẩm duy nhất, tiêu đề, mô tả và thông tin giá cả. Đăng ký có thêm thông tin cần thiết, chẳng hạn như thời gian gia hạn, liệu bạn có đang cung cấp bản dùng thử miễn phí hay không và liệu đăng ký có ưu đãi giới thiệu hay không.

Google Play Console cung cấp giao diện web mà bạn có thể sử dụng để quản lý sản phẩm của mình.

* Để tạo và định cấu hình sản phẩm một lần, hãy xem [Tạo sản phẩm được quản lý.](https://support.google.com/googleplay/android-developer/answer/1153481) Xin lưu ý rằng Google Play Console đề cập đến các sản phẩm một lần là sản phẩm được quản lý.
* Để tạo và cấu hình đăng ký, hãy [xem Tạo thuê bao](https://support.google.com/googleplay/android-developer/answer/140504?hl=en&ref_topic=3452890).

Thay thế cho giao diện web, bạn cũng có thể quản lý sản phẩm của mình bằng cách sử [dụng tài nguyên](https://developers.google.com/android-publisher/api-ref/rest/v3/inappproducts) REST không áp dụng trong API nhà phát triển của Google Play.

## Định cấu hình API nhà phát triển trên Google Play

API dành cho nhà phát triển trên Google Play là API máy chủ đến máy chủ bổ sung cho Thư viện thanh toán của Google Play trên Android. API này cung cấp chức năng không có sẵn trong Thư viện thanh toán của Google Play, chẳng hạn như xác minh an toàn các giao dịch mua và hoàn tiền cho người dùng của bạn.

Là một phần của việc tích hợp hệ thống thanh toán của Google Play vào ứng dụng của mình, bạn phải định cấu hình quyền truy cập vào API nhà phát triển trên Google Play thông qua Google Play Console. Để biết hướng dẫn, hãy [xem Bắt đầu với API nhà phát triển Trên Google Play](https://developers.google.com/android-publisher/getting_started).

Sau khi bạn đã định cấu hình quyền truy cập vào API nhà phát triển trên Google Play, hãy đảm bảo rằng bạn đã cấp quyền Xem dữ liệu tài chính cần thiết để truy cập chức năng liên quan đến thanh toán. Để biết thông tin về các phương pháp hay nhất, cùng với thông tin thêm về cách định cấu hình quyền, hãy xem [Thêm người dùng tài khoản nhà phát triển và quản lý quyền](https://support.google.com/googleplay/android-developer/answer/2528691).

## Định cấu hình thông báo dành cho nhà phát triển theo thời gian thực

Thông báo dành cho nhà phát triển theo thời gian thực (RTDN) là một cơ chế để nhận thông báo từ Google bất cứ khi nào có thay đổi về quyền lợi của người dùng trong ứng [dụng của bạn.](https://cloud.google.com/pubsub/docs/reference/libraries)RTDN tận dụng [việc sử dụng Google Cloud Pub / Sub](https://cloud.google.com/pubsub/docs/overview), cho phép bạn nhận dữ liệu được đẩyđến URL mà bạn đặt hoặc được thăm dò bằng thư viện máy khách. Những thông báo này cho phép bạn phản ứng ngay lập tức với các thay đổi về trạng thái đăng ký, tránh sự cần thiết phải thăm dò ý kiến API nhà phát triển của Google Play. Xin lưu ý rằng việc sử dụng API nhà phát triển trên Google Play không hiệu quả có thể dẫn đến các hạn chế về hạn ngạch API.

[Cloud Pub/Sub là](https://cloud.google.com/pubsub/) một dịch vụ nhắn tin thời gian thực được quản lý toàn phần mà bạn có thể sử dụng để gửi và nhận tin nhắn giữa các ứng dụng độc lập. Google Play sử dụng Cloud Pub/Sub để xuất bản thông báo đẩy về các chủ đề mà bạn đăng ký.

Để nhận thông báo, bạn cần tạo một máy chủ phụ trợ để tiêu thụ các tin nhắn được gửi đến chủ đề của bạn. Sau đó, máy chủ của bạn có thể tiêu thụ các thư này bằng cách trả lời yêu cầu HTTPS đến điểm cuối đã đăng ký hoặc bằng cách sử [dụng Thư viện Máy khách Cloud Pub/Sub](https://cloud.google.com/pubsub/docs/reference/libraries). Các thư viện này có sẵn bằng nhiều ngôn ngữ khác nhau. Bạn cũng có thể tìm thêm thông tin trong phần Tạo đăng ký [Pub/Sub](https://developer.android.com/google/play/billing/getting-ready#create-sub) trong chủ đề này.

### Xác định giá cả và hạn ngạch

Để biết chi tiết về giá cả và hạn ngạch, tham khảo [giá cả](https://cloud.google.com/pubsub/pricing) và [hạn ngạch](https://cloud.google.com/pubsub/quotas).

#### Ước tính mức sử dụng dữ liệu

Phần dữ liệu của thông báo đăng ký là khoảng 1KB dữ liệu cho mỗi yêu cầu. Mỗi lần xuất bản và kéo yêu cầu một yêu cầu riêng biệt hoặc khoảng 2KB dữ liệu cho mỗi thông báo. Số lượng thông báo mỗi tháng phụ thuộc vào chu kỳ thanh toán và hành vi của người dùng. Bạn nên mong đợi ít nhất một thông báo cho mỗi người dùng trong chu kỳ thanh toán.

### Thiết lập Cloud Pub/Sub

Để bật thông báo dành cho nhà phát triển theo thời gian thực, trước tiên bạn phải thiết lập Cloud Pub/Sub bằng dự án Google Cloud Platform (GCP) của riêng mình, sau đó bật thông báo cho ứng dụng của mình.

Để sử dụng Cloud Pub/Sub, bạn phải bật dự án GCP với Cloud Pub/Sub API. Nếu bạn không quen thuộc với GCP và Cloud Pub/Sub, hãy xem hướng [dẫn Bắt đầu nhanh](https://cloud.google.com/pubsub/docs/quickstart-console).

### Tạo chủ đề

Để bắt đầu nhận thông báo, bạn phải tạo [chủ](https://cloud.google.com/pubsub/architecture#the_basics_of_a_publishsubscribe_service) đề mà Google Play sẽ xuất bản thông báo. Để tạo chủ đề, hãy làm theo hướng dẫn trong Tạo chủ [đề](https://cloud.google.com/pubsub/docs/quickstart-console#create_a_topic).

### Tạo thuê bao Pub/Sub

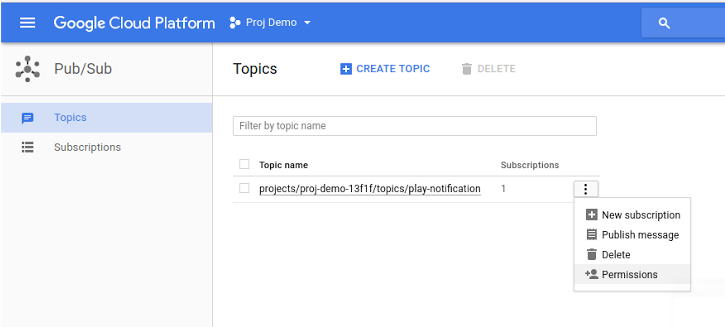
Để nhận thư được xuất bản thành một chủ đề, bạn phải tạo đăng ký Pub/Sub cho chủ đề đó. Để tạo đăng ký Pub/Sub, hãy làm như sau:

1. Đọc Hướng [dẫn đăng ký Cloud Pub/Sub để](https://cloud.google.com/pubsub/subscriber) xác định xem nên định cấu hình đăng ký dưới dạng đăng ký đẩy hoặc đăng ký kéo.
   * Đăng ký đẩy cho phép Cloud Pub / Sub gửi thông báo đến backend an toàn của bạn bằng cách phát hành các yêu cầu HTTPS.
   * Đăng ký kéo yêu cầu máy chủ backend an toàn của bạn bắt đầu các yêu cầu đến máy chủ Cloud Pub / Sub để truy xuất thư.
2. Làm theo hướng dẫn [trong Thêm đăng ký](https://cloud.google.com/pubsub/docs/quickstart-console#add_a_subscription) để tạo đăng ký.

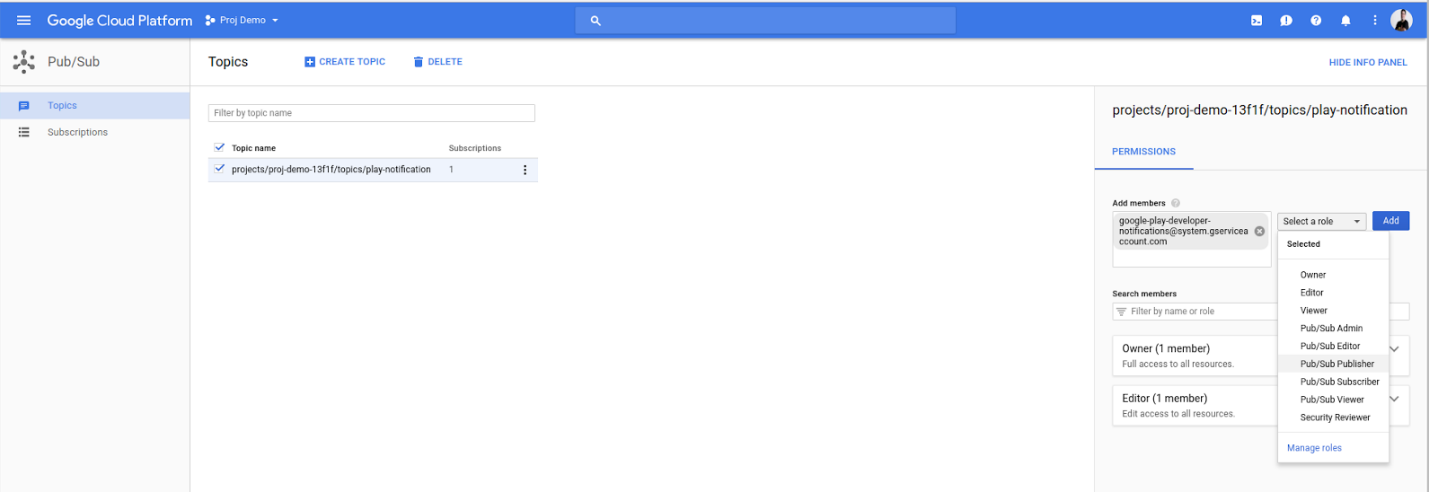
### Cấp quyền xuất bản về chủ đề của bạn

[Cloud Pub/Sub yêu cầu](https://cloud.google.com/pubsub/) bạn cấp đặc quyền cho Google Play để xuất bản thông báo cho chủ đề của mình.

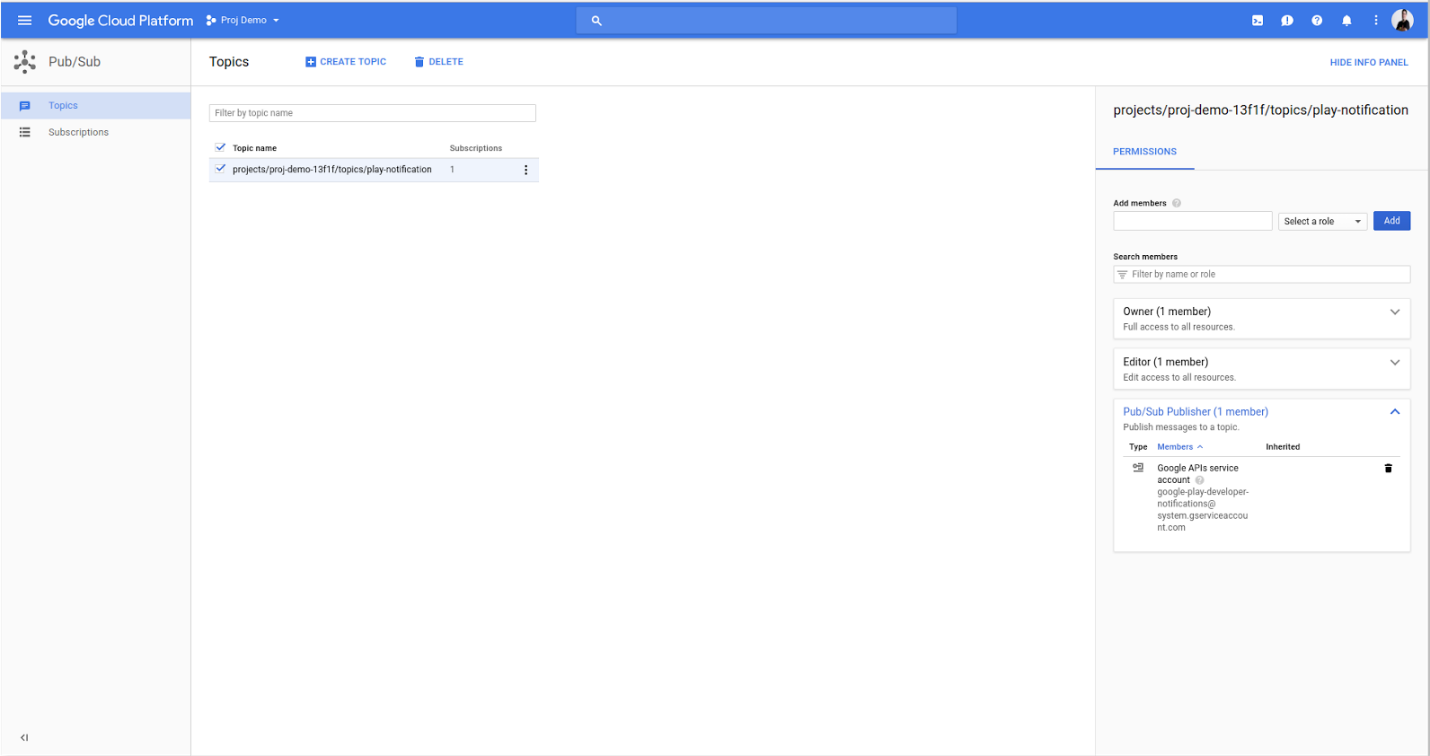
1. Mở [Google Cloud Console](https://console.cloud.google.com/).
2. Chọn dự án của bạn, rồi bấm Pub/Sub trong dẫn hướng bên trái.
3. Tìm chủ đề của bạn và mở chi tiết quyền.



1. Thêm tài khoản dịch vụ google-play-developer-notifications@system.gserviceaccount.com và cấp cho nó vai trò của Pub/Sub Publisher.

Biểu đồ 2. Thêm tài khoản dịch vụ Của Google Play làm nhà xuất bản Pub/Sub.

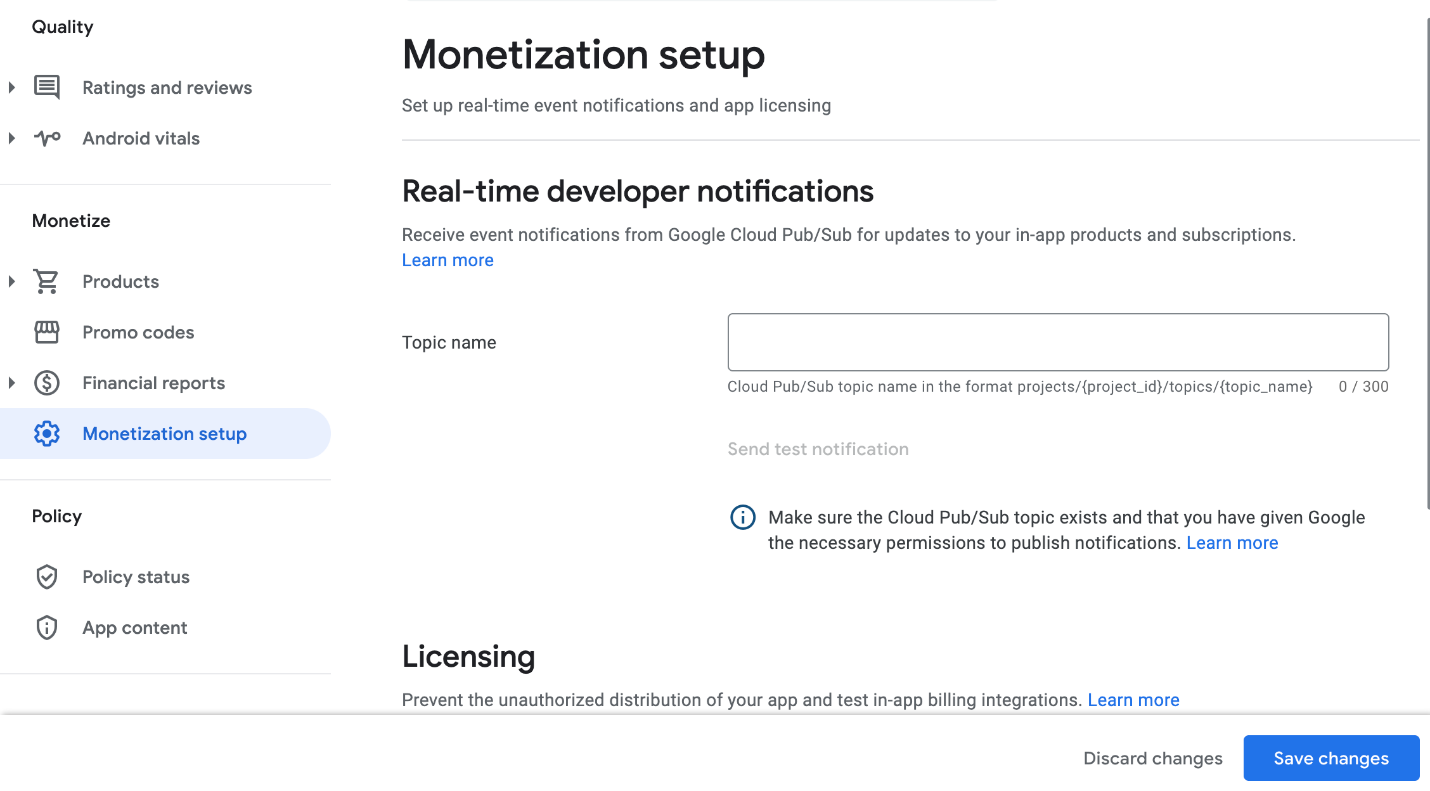
1. Bấm Lưu để hoàn tất thiết lập chủ đề.

Biểu đồ 3. Một chủ đề được cấu hình.

### Bật thông báo dành cho nhà phát triển theo thời gian thực cho ứng dụng của bạn

Để bật thông báo dành cho nhà phát triển theo thời gian thực cho ứng dụng của bạn, hãy làm như sau:

1. Mở [Google Play Console](https://play.google.com/console/).
2. Chọn ứng dụng của bạn.
3. Chuyển đến phần Kiếm tiền > thiết lập kiếm tiền.
4. Cuộn đến phần Thông báo dành cho nhà phát triển theo thời gian thực ở đầu trang.

Biểu đồ 4. Phần thông báo dành cho nhà phát triển theo thời gian thực.

1. Trong trường Tên chủ đề, nhập tên chủ đề Cloud Pub/Sub đầy đủ mà bạn đã cấu hình trước đó. Tên chủ đề phải ở định dạng của các dự án/{project\_id}/topics/{topic\_name} trong đó project\_id là mã định danh duy nhất cho dự án của bạn và topic\_name là tên của chủ đề được tạo trước đó.
2. Nhấp vào Gửi thư kiểm tra để gửi thư kiểm tra. Thực hiện xuất bản thử nghiệm giúp đảm bảo rằng mọi thứ được thiết lập và cấu hình đúng. Nếu xuất bản thử nghiệm thành công, một thông báo được hiển thị cho biết xuất bản thử nghiệm đã thành công. Nếu bạn có người đăng ký cho chủ đề này, nó sẽ nhận được thông báo kiểm tra.

Nếu xuất bản không thành công, lỗi được hiển thị. Đảm bảo rằng tên chủ đề là chính xác và tài khoản dịch vụ google-play-developer-notifications@system.gserviceaccount.com có quyền truy cập Pub/Sub Publisher vào chủ đề.

1. Bấm Lưu thay đổi.

### Xác minh cấu hình của bạn

Để nhận thông báo của nhà phát triển theo thời gian thực, bạn nên tạo một máy chủ backend an toàn để tiêu thụ các tin nhắn được gửi đến chủ đề Cloud Pub / Sub của bạn.

Bạn có thể kiểm tra cấu hình của mình bằng cách sử dụng nút Gửi tin nhắn kiểm tra trong Google Play Console như được mô tả trong phần trước. Nếu bạn chưa định cấu hình máy chủ phụ trợ để tiêu thụ thông báo, bạn có thể sử dụng công cụ dòng lệnh gcloud để xác minh cấu hình. Để biết hướng dẫn xử lý tin nhắn bằng gcloud, [hãy xem Kéo tin nhắn từ đăng ký](https://cloud.google.com/pubsub/docs/quickstart-console#pull_the_message_from_the_subscription).

# Tích hợp Thư viện thanh toán của Google Play vào ứng dụng của bạn

Chủ đề này mô tả cách tích hợp Thư viện thanh toán của Google Play vào ứng dụng của bạn để bắt đầu bán sản phẩm. Trước khi đọc chủ đề này, hãy đảm bảo bạn đã thiết lập cấu hình Google Play trước bằng cách làm theo các bước trong [Chuẩn bị sẵn sàng](https://developer.android.com/google/play/billing/getting-ready).

Chủ đề này bao gồm các ví dụ mã dựa trên các ứng dụng mẫu chính thức trên GitHub. Xem các [tài nguyên bổ](https://developer.android.com/google/play/billing/additional-resources#samples) sung để biết danh sách đầy đủ các ứng dụng mẫu và các tài nguyên khác mà bạn có thể sử dụng trong khi tích hợp.

## Tuổi thọ của một giao dịch mua

Dưới đây là quy trình mua hàng điển hình cho giao dịch mua một lần hoặc đăng ký.

* Cho người dùng thấy những gì họ có thể mua.
* Khởi chạy luồng mua hàng để người dùng chấp nhận mua hàng.
* Xác minh giao dịch mua trên máy chủ của bạn.
* Cung cấp nội dung cho người dùng và xác nhận việc phân phối nội dung. Tùy chọn, đánh dấu mặt hàng là đã tiêu thụ để người dùng có thể mua lại mặt hàng.

Đăng ký tự động gia hạn cho đến khi chúng bị hủy. Đăng ký có thể đi qua các trạng thái sau:

* Đang hoạt động: Người dùng đang ở trạng bộ tốt và có quyền truy cập vào đăng ký.
* Đã hủy: Người dùng đã hủy nhưng vẫn có quyền truy cập cho đến khi hết hạn.
* Trong thời gian gia hạn: Người dùng gặp sự cố thanh toán nhưng vẫn có quyền truy cập trong khi Google đang thử lại phương thức thanh toán.
* Tạm dừng: Người dùng gặp sự cố thanh toán và không còn quyền truy cập trong khi Google đang thử lại phương thức thanh toán.
* Tạm dừng: Người dùng đã tạm dừng quyền truy cập của họ và không có quyền truy cập cho đến khi họ tiếp tục.
* Đã hết hạn: Người dùng đã hủy và mất quyền truy cập vào đăng ký. Người dùng được coi là khuấy khi hết hạn.

## Mã thông báo mua hàng và ID đơn hàng

Google Play theo dõi các sản phẩm và giao dịch bằng mã thông báo mua hàng và ID đơn hàng.

* Mã thông báo mua hàng là chuỗi đại diện cho quyền lợi của người mua đối với sản phẩm trên Google Play. Điều này cho thấy rằng người dùng Google có quyền sử dụng một sản phẩm cụ thể được đại diện bởi SKU. Bạn có thể sử dụng mã thông báo mua hàng với API nhà phát triển trên Google Play.
* ID đơn hàng là chuỗi đại diện cho giao dịch tài chính trên Google Play. Chuỗi này được bao gồm trong biên lai được gửi qua email cho người mua. Bạn có thể sử dụng ID đơn hàng để quản lý tiền hoàn lại trong báo cáo doanh số và thanh toán đã sử dụng.

ID lệnh được tạo mỗi khi giao dịch tài chính xảy ra. Mã thông báo mua hàng chỉ được tạo khi người dùng hoàn tất quy trình mua hàng.

* Đối với các sản phẩm một lần, mỗi giao dịch mua tạo ra một mã thông báo mua hàng mới. Hầu hết các giao dịch mua cũng tạo ID đơn hàng mới. Ngoại lệ cho điều này là khi người dùng không bị tính bất kỳ khoản tiền nào, như được mô tả trong [Mã khuyến mãi](https://developer.android.com/google/play/billing/promo).
* Đối với đăng ký, giao dịch mua ban đầu sẽ tạo mã thông báo mua hàng và ID đơn hàng. Đối với mỗi kỳ thanh toán liên tục, mã thông báo mua hàng vẫn giữ nguyên và ID đơn hàng mới được phát hành. Nâng cấp, hạ cấp, thay thế và đăng ký lại đều tạo mã thông báo mua hàng và ID đơn hàng mới.

Đối với đăng ký, hãy lưu ý những điều sau:

* Nâng cấp đăng ký, hạ cấp và các luồng mua đăng ký khác tạo ra mã thông báo mua hàng phải thay thế mã thông báo mua trước đó. Bạn phải làm mất hiệu lực mã thông báo mua hàng xuất hiện trong trường LinkedPurchaseToken của API nhà phát triển Google Play. Để biết thêm thông tin, hãy [xem Thực hiện linkedPurchaseToken chính xác để ngăn chặn đăng ký trùng lặp](https://medium.com/androiddevelopers/implementing-linkedpurchasetoken-correctly-to-prevent-duplicate-subscriptions-82dfbf7167da).
* Số đơn đặt hàng để gia hạn đăng ký chứa một số nguyên bổ sung đại diện cho một phiên bản gia hạn cụ thể. Ví dụ: ID đơn hàng đăng ký ban đầu có thể là GPA.1234-5678-9012-34567 với ID đơn hàng tiếp theo là GPA.1234-5678-9012-34567.. 0 (gia hạn đầu tiên), GPA.1234-5678-9012-34567.. 1 (gia hạn thứ hai), v.v.

## Xử lý Lỗi

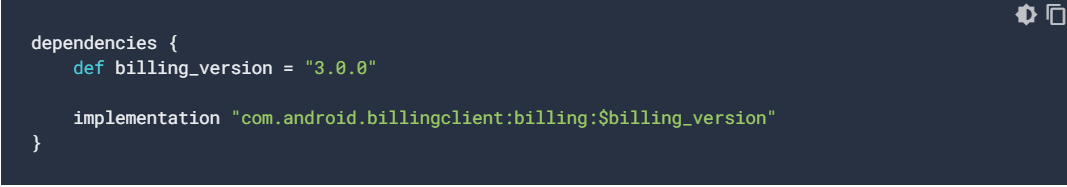
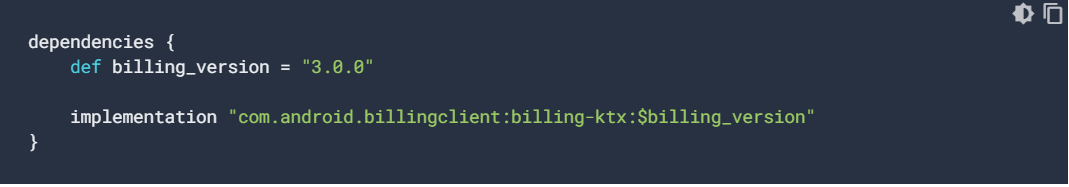
Thư viện thanh toán Google Play trả về lỗi dưới dạng [BillingResult](https://developer.android.com/reference/com/android/billingclient/api/BillingResult). BillingResult chứa [BillingResponseCode](https://developer.android.com/reference/com/android/billingclient/api/BillingClient.BillingResponseCode), phân loại cáclỗi có thể liên quan đến thanh toán mà ứng dụng của bạn có thể gặp phải. Ví dụ: nếu bạn nhận được mã [SERVICE\_DISCONNECTED,](https://developer.android.com/reference/com/android/billingclient/api/BillingClient.BillingResponseCode#service_disconnected) ứng dụng của bạn nên khởi tạo lại kết nối với Google Play. Ngoài ra, BillingResult chứa thông báo [gỡ lỗi , rất hữu](https://developer.android.com/reference/com/android/billingclient/api/BillingResult#getDebugMessage())ích trong quá trình phát triển để chẩn đoán lỗi.

## Kết nối với Google Play

Bước đầu tiên để tích hợp với hệ thống thanh toán của Google Play là thêm thư viện vào ứng dụng của bạn và khởi tạo kết nối.

## Thêm phụ thuộc Vào Thư viện thanh toán của Google Play

Thêm phụ thuộc Vào Thư viện thanh toán của Google Play vào tệp build.gradle của ứng dụng như được hiển thị:

Nếu bạn đang sử dụng Kotlin, mô-đun KTX của Thư viện thanh toán Play chứa tiện ích mở rộng Kotlin và hỗ trợ coroutines cho phép bạn viết Kotlin thành ngữ khi sử dụng Thư viện thanh toán của Google Play. Để bao gồm các tiện ích mở rộng này trong dự án của bạn, hãy thêm phần khởi tạo BillingClient sau đây

Sau khi thêm phụ thuộc vào Thư viện thanh toán của Google Play, bạn cần khởi tạo phiên [bản BillingClient.](https://developer.android.com/reference/com/android/billingclient/api/BillingClient)  BillingClient là giao diện chính để giao tiếp giữa Thư viện thanh toán của Google Play và phần còn lại của ứng dụng. BillingClient cung cấp các phương thức tiện lợi, cả đồng bộ và không đồng bộ, cho nhiều hoạt động thanh toán phổ biến.

Để tạo BillingClient, hãy sử [dụng newBuilder()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#newBuilder(android.content.Context)). Để nhận thông tin cập nhật về giao dịch mua, bạn cũng phải [gọi setListener(),](https://developer.android.com/reference/com/android/billingclient/api/BillingClient.Builder#setListener(com.android.billingclient.api.PurchasesUpdatedListener))chuyển tham chiếu [đến PurchasesUpdatedListener](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener). Trình nghe này nhận được thông tin cập nhật cho tất cả các giao dịch mua trong ứng dụng của bạn.

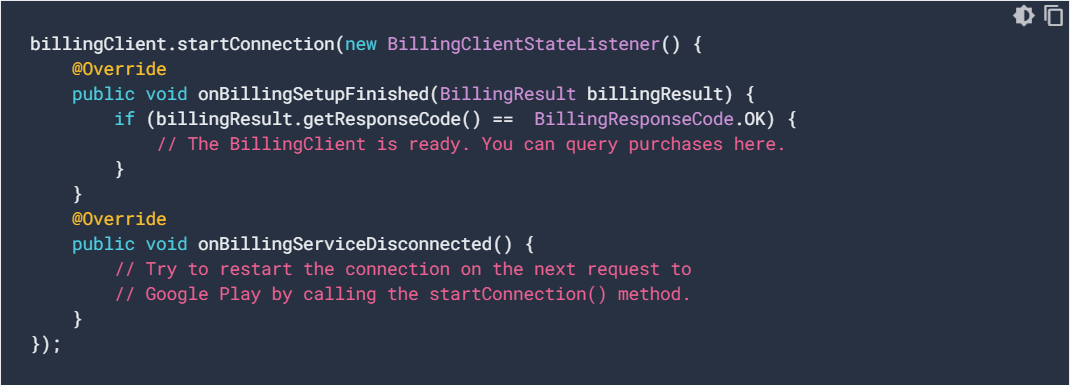
## Thiết lập kết nối với Google Play

Sau khi tạo BillingClient, bạn cần thiết lập kết nối với Google Play.

Để kết nối với Google Play, hãy [gọi startConnection()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#startConnection(com.android.billingclient.api.BillingClientStateListener)). Quá trình kết nối không đồng bộ và bạn phải thực hiện [BillingClientStateListener để](https://developer.android.com/reference/com/android/billingclient/api/BillingClientStateListener) nhận được gọi lại sau khi thiết lập máy khách hoàn tất và sẵn sàng thực hiện thêm yêu cầu.

Bạn cũng phải triển khai logic thử lại để xử lý các kết nối bị mất với Google Play. Để thực hiện logic thử lại, hãy ghi [đè phương thức gọi lại onBillingServiceDisconnected()](https://developer.android.com/reference/com/android/billingclient/api/BillingClientStateListener#onBillingServiceDisconnected()) và đảm bảo rằng BillingClient gọi phương thức [startConnection() để](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#startConnection(com.android.billingclient.api.BillingClientStateListener)) kết nối lại với Google Play trước khi thực hiện thêm yêu cầu.

Ví dụ sau đây trình bày cách bắt đầu kết nối và kiểm tra xem kết nối đã sẵn sàng để sử dụng:

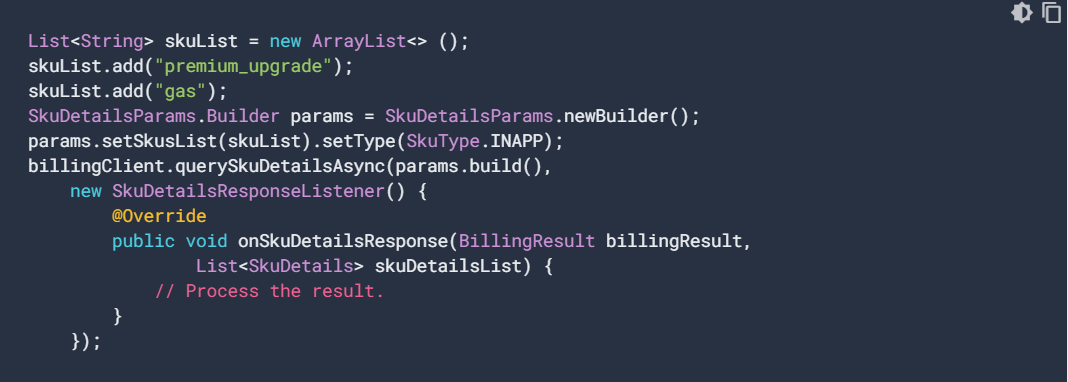


## Hiển thị sản phẩm có sẵn để mua

Sau khi thiết lập kết nối với Google Play, bạn đã sẵn sàng truy vấn các sản phẩm có sẵn và hiển thị chúng cho người dùng của mình. Để truy vấn Google Play để biết chi tiết sản phẩm trong ứng dụng, [hãy truy vấn cuộc gọiSkuDetailsAsync()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querySkuDetailsAsync(com.android.billingclient.api.SkuDetailsParams,%20com.android.billingclient.api.SkuDetailsResponseListener)). Truy vấn chi tiết SKU là một bước quan trọng trước khi hiển thị sản phẩm cho người dùng, vì nó trả về thông tin sản phẩm được bản địa hóa. Đối với gói đăng ký, hãy đảm bảo hiển thị sản phẩm của bạn tuân thủ tất cả các chính sách của [Play.](https://play.google.com/about/monetization-ads/subscriptions)

Khi gọi querySkuDetailsAsync(), hãy chuyển một [phiên bản của SkuDetailsParams](https://developer.android.com/reference/com/android/billingclient/api/SkuDetailsParams) chỉ định danh sách các chuỗi ID sản phẩm được tạo trong Google Play Console cùng với SkuType. SkuType có thể là SkuType.INAPP cho các sản phẩm một lần hoặc SkuType.SUBS cho đăng ký.

Để xử lý kết quả của thao tác không đồng bộ, bạn cũng phải chỉ định một listener thực [hiện giao diện SkuDetailsResponseListener.](https://developer.android.com/reference/com/android/billingclient/api/SkuDetailsResponseListener) Sau đó, bạn có [thể ghi đè lên onSkuDetailsResponse()](https://developer.android.com/reference/com/android/billingclient/api/SkuDetailsResponseListener#onSkuDetailsResponse(com.android.billingclient.api.SkuDetails.SkuDetailsResult)), thông báo cho người nghe khi truy vấn kết thúc, như trong ví dụ sau:

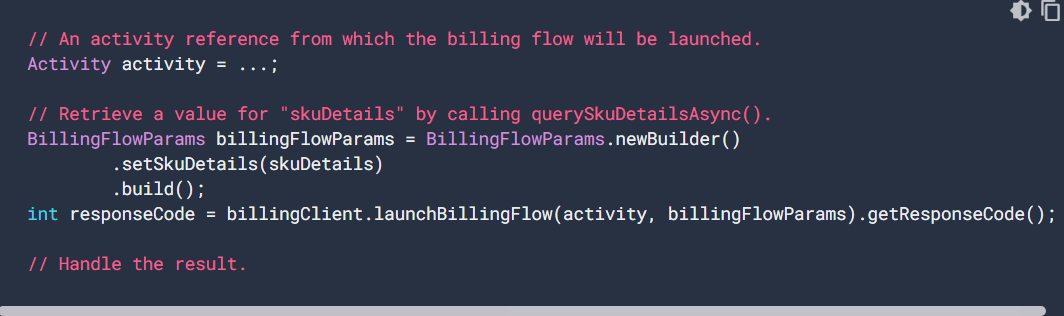
Thư viện thanh toán của Google Play lưu trữ kết quả truy vấn trong Danh sách [các đối tượng SkuDetails.](https://developer.android.com/reference/com/android/billingclient/api/SkuDetails) Sau đó, bạn có thể gọi nhiều phương thức khác nhau trên mỗi đối tượng SkuDetails trong danh sách để xem thông tin liên quan về một sản phẩm trong ứng dụng, chẳng hạn như giá hoặc mô tả của nó. Để xem thông tin chi tiết sản phẩm có sẵn, hãy xem danh sách các phương thức trong lớp [SkuDetails.](https://developer.android.com/reference/com/android/billingclient/api/SkuDetails)

Trước khi chào bán mặt hàng, hãy kiểm tra xem người dùng chưa sở hữu mặt hàng chưa. Nếu người dùng có vật tư tiêu hao vẫn còn trong thư viện mặt hàng của họ, họ phải tiêu thụ mặt hàng trước khi họ có thể mua lại.

Trước khi cung cấp đăng ký, hãy xác minh rằng người dùng chưa đăng ký.

## Khởi chạy luồng mua hàng

Để bắt đầu yêu cầu mua hàng từ ứng dụng của bạn, hãy [gọi phương thức launchBillingFlow()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#launchbillingflow) từ luồng chính của ứng dụng. Phương pháp này có tham chiếu đến đối tượng [BillingFlowParams](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams) có chứa đối tượng [SkuDetails có](https://developer.android.com/reference/com/android/billingclient/api/SkuDetails) liên quan thu được từ việc gọi [querySkuDetailsAsync()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#queryskudetailsasync). Để tạo đối tượng BillingFlowParams, hãy sử dụng [lớp BillingFlowParams.Builder.](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams.Builder)

Phương thức launchBillingFlow() trả về một trong một số mã phản hồi được liệt [kê trong BillingClient.BillingResponseCode](https://developer.android.com/reference/com/android/billingclient/api/BillingClient.BillingResponseCode). Hãy chắc chắn kiểm tra kết quả này để đảm bảo không có lỗi khởi chạy luồng mua hàng. BillingResponseCode của OK cho biết khởi chạy thành công.

Trong một cuộc gọi thành công để khởi chạyBillingFlow(), hệ thống sẽ hiển thị màn hình mua hàng của Google Play. Hình 1 hiển thị màn hình mua hàng cho đăng ký:

the google play purchase screen shows a subscription that is
            available for purchaseBiểu đồ 1. Màn hình mua hàng trên Google Play hiển thị gói đăng ký có sẵn để mua.

Google Play kêu gọiPurchasesUpdated() cung cấp kết quả của hoạt động mua hàng cho người nghe thực hiện giao diện PurchasesUpdatedListener. Nghe được chỉ định bằng phương thức setListener() khi bạn khởi [tạo máy khách của mình](https://developer.android.com/google/play/billing/integrate#initialize).

Bạn phải thực hiện onPurchasesUpdated() để xử lý các mã phản hồi có thể. Ví dụ sau đây cho thấy cách ghi đè lên onPurchasesUpdated():

Giao dịch mua thành công tạo ra màn hình thành công mua hàng trên Google Play tương tự như hình 2.

Biểu đồ 2. Màn hình thành công mua hàng của Google Play.

Giao dịch mua thành công cũng tạo ra mã thông báo mua hàng, đây là mã định danh duy nhất đại diện cho người dùng và ID sản phẩm cho sản phẩm trong ứng dụng mà họ đã mua. Ứng dụng của bạn có thể lưu trữ mã thông báo mua hàng cục bộ, mặc dù chúng tôi khuyên bạn nên chuyển mã thông báo đến máy chủ backend an toàn của mình, nơi sau đó bạn có thể xác minh giao dịch mua và bảo vệ chống gian lận. Quá trình này được mô tả thêm trong phần sau.

Người dùng cũng được gửi qua email biên lai giao dịch có chứa ID đơn hàng hoặc ID duy nhất của giao dịch. Người dùng nhận được email có ID đơn hàng duy nhất cho mỗi lần mua sản phẩm một lần, đồng thời cho giao dịch mua đăng ký ban đầu và gia hạn tự động định kỳ tiếp theo. Bạn có thể sử dụng Mã đơn hàng để quản lý khoản hoàn tiền trong Google Play Console.

## Xử lý giao dịch mua

Sau khi người dùng hoàn tất giao dịch mua, ứng dụng của bạn cần xử lý giao dịch mua đó. Trong hầu hết các trường hợp, ứng dụng của bạn được thông báo về các giao [dịch mua thông qua PurchasesUpdatedListener](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener).

Ứng dụng của bạn nên xử lý giao dịch mua theo cách sau:

1. Xác minh giao dịch mua.
2. Cung cấp nội dung cho người dùng và xác nhận việc phân phối nội dung. Tùy chọn, đánh dấu mặt hàng là đã tiêu thụ để người dùng có thể mua lại mặt hàng.

Để xác minh giao dịch mua, trước tiên hãy kiểm tra xem trạng [thái mua đã](https://developer.android.com/reference/com/android/billingclient/api/Purchase#getpurchasestate) mua [.](https://developer.android.com/reference/com/android/billingclient/api/Purchase.PurchaseState) Nếu giao dịch mua đang chờ xử lý, thì bạn nên xử lý giao dịch mua như được mô tả trong Xử lý các giao dịch đang chờ xử [lý](https://developer.android.com/google/play/billing/integrate#pending). Đối với các giao dịch mua nhận được từ [onPurchaseUpdated()](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener#onPurchasesUpdated(com.android.billingclient.api.BillingResult,%20java.util.List%3Ccom.android.billingclient.api.Purchase%3E)) hoặc [queryPurchases](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases), bạn nên xác minh thêm việc mua hàng để đảm bảo tính hợp pháp trước khi ứng dụng của bạn cấp quyền lợi. Để tìm hiểu cách xác minh đúng cách giao dịch mua, hãy xem Xác [minh giao dịch mua trước khi cấp quyền](https://developer.android.com/google/play/billing/security#verify)lợi .

Sau khi bạn đã xác minh giao dịch mua, ứng dụng của bạn đã sẵn sàng cấp quyền cho người dùng. Sau khi cấp quyền lợi, ứng dụng của bạn phải xác nhận giao dịch mua. Xác nhận này thông báo cho Google Play rằng bạn đã cấp quyền mua hàng.

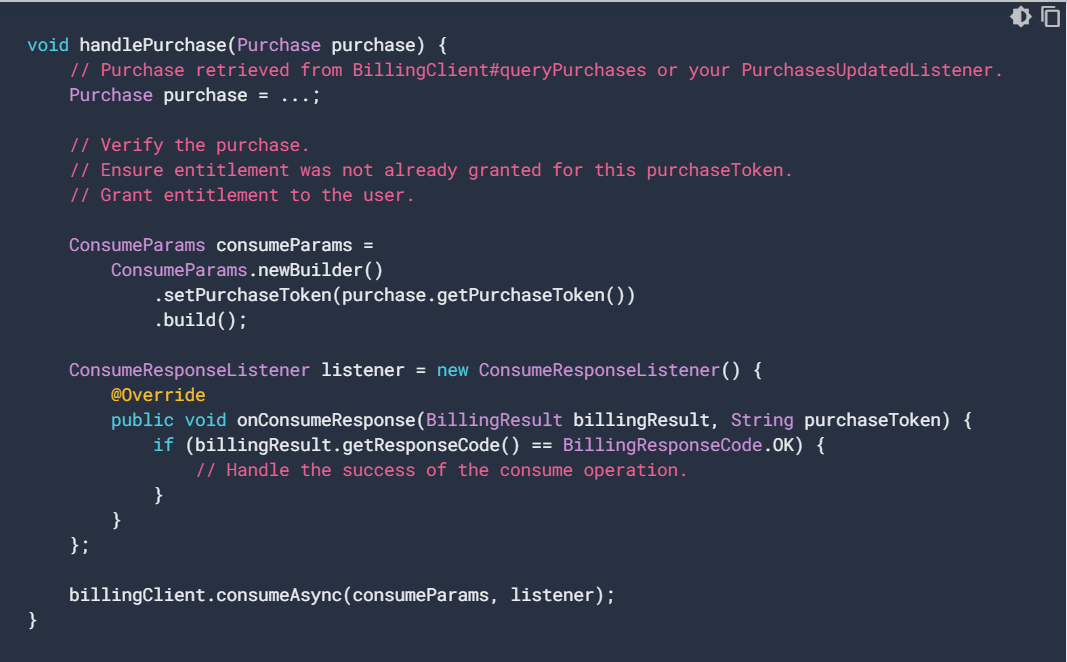
Lưu ý: Nếu bạn không xác nhận giao dịch mua trong vòng ba ngày, người dùng sẽ tự động nhận được tiền hoàn lại và Google Play sẽ thu hồi giao dịch mua. Lưu ý: Bạn không cần xác nhận khi sử dụng phiên bản Thư viện thanh toán của Google Play trước 2.0.

Quá trình cấp quyền lợi và thừa nhận việc mua phụ thuộc vào việc mua hàng là không tiêu hao, tiêu hao hoặc đăng ký.

Đối với hàng tiêu dùng, phương [thức consumeAsync()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#consumeasync) đáp ứng yêu cầu xác nhận và chỉ ra rằng ứng dụng của bạn đã cấp quyền cho người dùng. Phương pháp này cũng cho phép ứng dụng của bạn cung cấp sản phẩm một lần để mua lại.

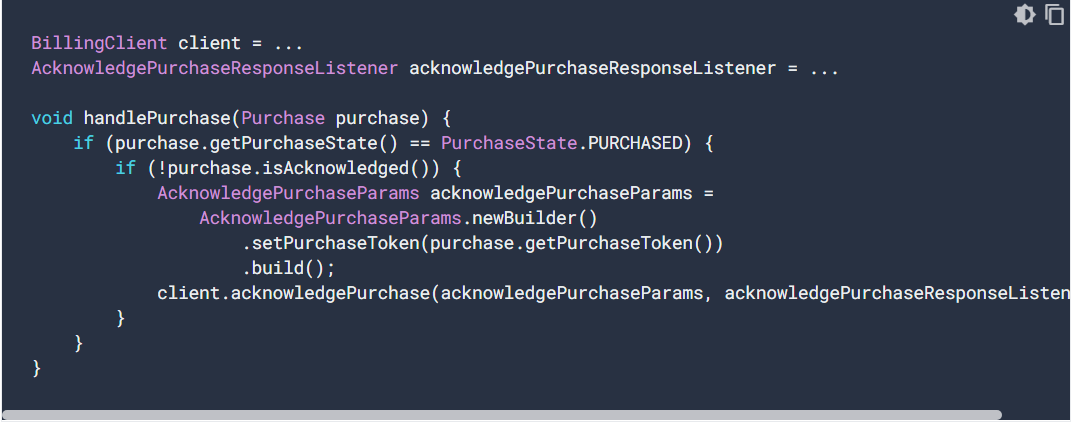
Để chỉ ra rằng một sản phẩm một lần đã được tiêu thụ, hãy gọi consumeAsync() và bao gồm mã thông báo mua hàng mà Google Play sẽ cung cấp để mua lại. Bạn cũng phải vượt qua một đối tượng thực hiện [giao diện ConsumeResponseListener.](https://developer.android.com/reference/com/android/billingclient/api/ConsumeResponseListener) Đối tượng này xử lý kết quả của thao tác tiêu thụ. Bạn có thể ghi đè phương thức [onConsumeResponse()](https://developer.android.com/reference/com/android/billingclient/api/ConsumeResponseListener#onConsumeResponse(com.android.billingclient.api.BillingResult,%20java.lang.String)) mà Thư viện thanh toán của Google Play gọi khi thao tác hoàn tất.

Ví dụ sau minh họa việc tiêu thụ sản phẩm bằng mã thông báo mua hàng được liên kết:

Lưu ý: Vì yêu cầu tiêu thụ đôi khi có thể không thành công, bạn phải kiểm tra máy chủ phụ trợ an toàn của mình để đảm bảo rằng mỗi mã thông báo mua hàng chưa được sử dụng để ứng dụng của bạn không cấp quyền truy cập nhiều lần cho cùng một giao dịch mua. Ngoài ra, ứng dụng của bạn có thể đợi cho đến khi bạn nhận được phản hồi về mức tiêu thụ thành công từ Google Play trước khi cấp quyền. Nếu bạn chọn giữ lại giao dịch mua từ người dùng cho đến khi Google Play gửi phản hồi tiêu thụ thành công, bạn phải rất cẩn thận để không mất dấu việc mua hàng sau yêu cầu tiêu thụ.

Để xác nhận các giao dịch mua không tiêu hao, hãy [sử dụng BillingClient.acknowledgePurchase() từ](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#acknowledgepurchase) Thư viện thanh toán [hoặc Product.Purchases.Acknowledge](https://developers.google.com/android-publisher/api-ref/purchases/products/acknowledge) từ API nhà phát triển của Google Play. Trước khi xác nhận giao dịch mua, ứng dụng của bạn nên kiểm tra xem ứng dụng đó đã được xác nhận chưa bằng cách sử dụng phương thức [isAcknowledged()](https://developer.android.com/reference/com/android/billingclient/api/Purchase#isacknowledged) trong Thư viện thanh toán của Google Play hoặc trường [acknowledgementState](https://developers.google.com/android-publisher/api-ref/purchases/products) trong API nhà phát triển của Google.

Ví dụ sau đây cho biết cách xác nhận giao dịch mua bằng Thư viện thanh toán của Google Play:

Đăng ký được xử lý tương tự như không tiêu hao. Bạn có thể xác nhận đăng ký bằng [BillingClient.acknowledgePurchase() từ](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#acknowledgepurchase) Thư viện thanh toán của Google Play hoặc [Purchases.Subscriptions.Acknowledge](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/acknowledge) từ API nhà phát triển của Google Play. Tất cả các giao dịch mua đăng ký ban đầu cần phải được thừa nhận. Gia hạn đăng ký không cần phải được thừa nhận. Để biết thêm thông tin về thời điểm đăng ký cần được xác nhận, hãy xem chủ [đề Bán đăng](https://developer.android.com/google/play/billing/subs) ký.

## Tìm nạp mục mua

Nghe các bản cập nhật mua hàng bằng [PurchasesUpdatedListener là không đủ](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener) để đảm bảo ứng dụng của bạn xử lý tất cả các giao dịch mua. Có thể ứng dụng của bạn có thể không biết tất cả các giao dịch mua mà người dùng đã thực hiện. Dưới đây là một số tình huống mà ứng dụng của bạn có thể mất dấu hoặc không biết về giao dịch mua:

* Sự cố mạng trong quá trình mua hàng: Người dùng mua hàng thành công và nhận được xác nhận từ Google, nhưng thiết bị của họ bị mất kết nối mạng trước khi thiết bị của họ nhận được thông báo về giao dịch mua thông qua PurchasesUpdatedListener.
* Nhiều thiết bị: Người dùng mua một mặt hàng trên một thiết bị và sau đó mong đợi nhìn thấy mặt hàng khi họ chuyển đổi thiết bị.
* Xử lý các giao dịch mua được thực hiện bên ngoài ứng dụng của bạn: Bạn có thể thực hiện một số giao dịch mua, chẳng hạn như đổi quà khuyến mãi bên ngoài ứng dụng của mình.

Để xử lý các tình huống này, hãy đảm bảo rằng ứng dụng của [bạn gọi BillingClient.queryPurchases() bằng](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases) các phương thức onResume() và onCreate() của bạn để đảm bảo rằng tất cả các giao dịch mua được xử lý thành công như được mô tả trong quá trình xử lý giao dịch [mua](https://developer.android.com/google/play/billing/integrate#process).

## Xử lý các giao dịch mua được thực hiện bên ngoài ứng dụng của bạn

Một số giao dịch mua, chẳng hạn như đổi quà khuyến mãi, có thể xảy ra bên ngoài ứng dụng của bạn. Khi người dùng mua hàng bên ngoài ứng dụng của bạn, họ mong đợi ứng dụng của bạn hiển thị tin nhắn trong ứng dụng hoặc sử dụng một số loại cơ chế thông báo để cho người dùng biết rằng ứng dụng đã nhận và xử lý giao dịch mua chính xác. Một số cơ chế chấp nhận được là:

* Hiển thị cửa sổ bật lên trong ứng dụng.
* Gửi tin nhắn đến hộp thông báo trong ứng dụng và nêu rõ rằng có một tin nhắn mới trong hộp thông báo trong ứng dụng.
* Sử dụng thông báo hệ điều hành.

Lưu ý rằng ứng dụng của bạn có thể ở bất kỳ trạng thái nào khi ứng dụng của bạn nhận ra giao dịch mua. Ứng dụng của bạn thậm chí có thể không được cài đặt khi mua hàng được thực hiện. Người dùng dự kiến sẽ nhận được giao dịch mua của họ khi họ tiếp tục ứng dụng, bất kể trạng thái của ứng dụng.

Bạn phải phát hiện các giao dịch mua bất kể trạng thái mà ứng dụng là khi mua hàng được thực hiện. Tuy nhiên, có một số trường hợp ngoại lệ mà có thể chấp nhận được khi không thông báo ngay cho người dùng rằng mục đã được nhận. Ví dụ:

* Trong phần hành động của trò chơi, trong đó việc hiển thị thông báo có thể làm người dùng mất tập trung. Trong trường hợp này, bạn phải thông báo cho người dùng sau khi phần hành động kết thúc.
* Trong các đoạn cắt cảnh, trong đó việc hiển thị một thông báo có thể làm người dùng mất tập trung. Trong trường hợp này, bạn phải thông báo cho người dùng sau khi đoạn cắt cảnh kết thúc.
* Trong hướng dẫn ban đầu và các phần thiết lập người dùng của trò chơi. Chúng tôi khuyên bạn nên thông báo cho người dùng mới về phần thưởng ngay sau khi họ mở trò chơi hoặc trong quá trình thiết lập người dùng ban đầu. Tuy nhiên, có thể chấp nhận đợi cho đến khi chuỗi trò chơi chính có sẵn để thông báo cho người dùng.

Luôn ghi nhớ người dùng khi quyết định thời điểm và cách thông báo cho người dùng về các giao dịch mua được thực hiện bên ngoài ứng dụng của bạn. Bất cứ khi nào người dùng không nhận được thông báo ngay lập tức, họ có thể bị nhầm lẫn và có thể ngừng sử dụng ứng dụng của bạn, liên hệ với bộ phận hỗ trợ người dùng hoặc phàn nàn về ứng dụng đó trên phương tiện truyền thông xã hội.

## Xử lý các giao dịch đang chờ xử lý

Lưu ý: Các giao dịch đang chờ xử lý chỉ có sẵn trong Thư viện thanh toán Google Play phiên bản 2.0 trở lên. Lưu ý: Các hình thức thanh toán bổ sung không có sẵn để mua gói đăng ký.

Google Play hỗ trợ các giao dịch đang chờ xử lý hoặc các giao dịch yêu cầu một hoặc nhiều bước bổ sung giữa thời điểm người dùng bắt đầu mua hàng và thời điểm phương thức thanh toán cho giao dịch mua được xử lý. Ứng dụng của bạn không được cấp quyền cho các loại giao dịch mua này cho đến khi Google thông báo cho bạn rằng phương thức thanh toán của người dùng đã được tính phí thành công.

Ví dụ: người dùng có thể tạo giao dịch mua mục đang chờ xử lý trong ứng dụng bằng cách chọn tiền mặt làm hình thức thanh toán. Sau đó, người dùng có thể chọn một cửa hàng thực tế nơi họ sẽ hoàn thành giao dịch và nhận mã thông qua cả thông báo và email. Khi người dùng đến cửa hàng thực, họ có thể đổi mã bằng nhân viên thu ngân và thanh toán bằng tiền mặt. Sau đó, Google thông báo cho cả bạn và người dùng rằng tiền mặt đã được nhận. Sau đó, ứng dụng của bạn có thể cấp quyền cho người dùng.

Để bật giao dịch mua đang chờ xử lý, [hãy gọi enablePendingPurchases() như một](https://developer.android.com/reference/com/android/billingclient/api/BillingClient.Builder#enablependingpurchases) phần của việc khởi tạo ứng dụng của bạn.

Khi ứng dụng của bạn nhận được giao dịch mua mới, [thông qua PurchasesUpdatedListener của bạn](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener) hoặc là kết quả của việc gọi [queryPurchases(),](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases)hãysử dụng phương [thức getPurchaseState()](https://developer.android.com/reference/com/android/billingclient/api/Purchase#getpurchasestate) để xác định xem trạng thái mua hàng là ĐÃ MUA hay ĐANG CHỜ XỬ LÝ. Lưu ý rằng bạn chỉ nên cấp quyền lợi khi tiểu bang được MUA. Nếu ứng dụng của bạn đang chạy khi người dùng hoàn tất giao dịch mua, [PurchasesUpdatedListener của bạn](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener) sẽ được gọi lại và PurchaseState hiện đã ĐƯỢC MUA. Tại thời điểm này, ứng dụng của bạn có thể xử lý giao dịch mua bằng phương pháp tiêu chuẩn để xử lý giao dịch mua một [lần](https://developer.android.com/google/play/billing/integrate#process). Ứng dụng của bạn cũng nên gọi queryPurchases() trong các phương thức onResume() và onCreate() của ứng dụng để xử lý các giao dịch mua đã chuyển sang trạng thái ĐÃ MUA trong khi ứng dụng của bạn không chạy.

Lưu ý: Bạn chỉ nên xác nhận mua hàng khi tiểu bang được MUA. Bạn không thể xác nhận trong khi giao dịch mua đang chờ xử lý. Thời gian xác nhận ba ngày chỉ bắt đầu khi trạng thái mua chuyển từ 'ĐANG CHỜ XỬ LÝ' sang 'ĐÃ MUA'.

Ứng dụng của bạn cũng có thể sử [dụng thông báo dành cho nhà phát triển thời](https://developer.android.com/google/play/billing/getting-ready#configure-rtdn) gian thực với các giao dịch mua đang chờ xử lý bằng cách nghe OneTimeProductNotifications. Khi giao dịch mua chuyển từ ĐANG CHỜ XỬ LÝ sang ĐÃ MUA, ứng dụng của bạn sẽ nhận được ONE\_TIME\_PRODUCT\_PURCHASED báo. Nếu giao dịch mua bị hủy, ứng dụng của bạn sẽ nhận được ONE\_TIME\_PRODUCT\_CANCELED báo. Điều này có thể xảy ra nếu khách hàng của bạn không hoàn tất thanh toán trong khung thời gian yêu cầu. Khi nhận được các thông báo này, bạn có thể sử dụng API nhà phát triển của Google Play, bao gồm trạng thái ĐANG CHỜ xử lý cho [Purchases.products](https://developers.google.com/android-publisher/api-ref/purchases/products).

# Sell subscriptions

Chủ đề này mô tả cách xử lý các sự kiện vòng đời đăng ký, chẳng hạn như gia hạn và hết hạn. Nó cũng mô tả các tính năng đăng ký bổ sung như cung cấp các chương trình khuyến mãi và cho phép người dùng của bạn quản lý đăng ký của riêng họ.

Trước khi đọc chủ đề này, hãy nhớ đã đọc [Tích hợp Thư viện thanh toán google Play](https://developer.android.com/google/play/billing/integrate) vào ứng dụng của mình để biết hướng dẫn chung về cách bán và quản lý sản phẩm trong ứng dụng của bạn.

Nếu bạn chưa định cấu hình sản phẩm đăng ký cho ứng dụng của mình, hãy xem [Tạo và định cấu hình sản phẩm.](https://developer.android.com/google/play/billing/getting-ready#products)

## Xử lý vòng đời đăng ký

Đăng ký có thể trải qua nhiều thay đổi trạng thái khác nhau trong [suốt vòng đời của](https://developer.android.com/google/play/billing/integrate#life)nó và ứng dụng của bạn cần phản hồi từng thayđổi. Để kiểm tra trạng thái của người đăng ký, ứng dụng của bạn có thể truy vấn bằng [BillingClient.queryPurchases() trong](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases) Thư viện thanh toán của Google Play hoặc [Purchases.subscriptions:get](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/get) trong API nhà phát triển trên Google Play.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | BillingClient.queryPurchases() | | Purchases.subscriptions:nhận | | | |
| Bang | Được trả lại? | [isAutoĐể từ chối](https://developer.android.com/reference/com/android/billingclient/api/Purchase#isautorenewing) | Được trả lại? | hết hạnTimeMillis | paymentState | tự động Từ chối |
| Hoạt động | Có | Đúng | Có | Tương lai | 1 (Đã nhận thanh toán) | Đúng |
| Bị hủy bỏ | Có | Sai | Có | Tương lai | 1 (Đã nhận thanh toán) | Sai |
| Trong thời gian ân hạn | Có | Đúng | Có | Tương lai | 0 (Đang chờ thanh toán) | Đúng |
| Đang chờ | Không | N/a | Có | Quá khứ | 0 (Đang chờ thanh toán) | Đúng |
| Tạm dừng | Không | N/a | Có | Quá khứ | 1 (Đã nhận thanh toán) | Đúng |
| Hết hạn | Không | N/a | Có | Quá khứ | 1 (Đã nhận thanh toán) | Sai |

Lưu ý: Mã thông báo mua hàng có hiệu lực từ khi đăng ký đăng ký cho đến 60 ngày sau khi hết hạn. Sau ngày này, mã thông báo mua hàng không còn hợp lệ để gọi API nhà phát triển trên Google Play.

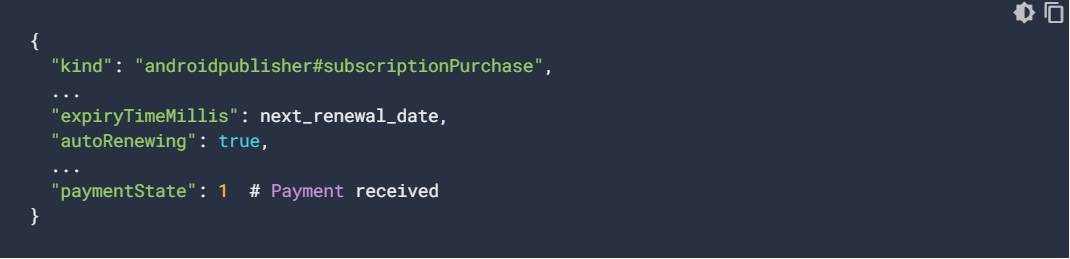
Nếu ứng dụng của bạn lưu trữ trạng thái đăng ký trên máy chủ phụ trợ an toàn, ứng dụng của bạn nên nghe các thay đổi trạng thái bằng [thông báo của nhà phát triển theo thời gian](https://developer.android.com/google/play/billing/getting-ready#configure-rtdn) thực để đảm bảo trạng thái được đồng bộ hóa. Thông báo [Đăng ký được gửi](https://developer.android.com/google/play/billing/rtdn-reference#sub) cho các sự kiện ảnh hưởng đến trạng thái đăng ký như gia hạn và hủy bỏ. Bạn cần gọi API nhà phát triển sau khi nhận được thông báo của nhà phát triển theo thời gian thực để có được trạng thái hoàn chỉnh và cập nhật trạng thái backend của riêng bạn. Những thông báo này chỉ cho bạn biết rằng trạng thái đăng ký đã thay đổi. Họ không cung cấp cho bạn thông tin đầy đủ về trạng thái đăng ký tổng thể.

Lưu ý: Do các hạn chế về hạn ngạch, bạn không nên kiểm tra trạng thái bằng cách thăm dò API nhà phát triển trên Google Play định kỳ thay vì tận dụng thông báo của nhà phát triển theo thời gian thực.

Ứng dụng của bạn cần xử lý các thay đổi trạng thái được mô tả trong các phần sau.

### Đăng ký mới

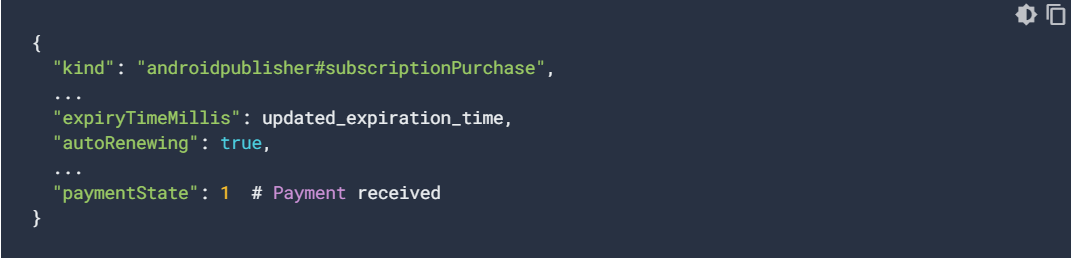
Hãy chắc chắn làm theo các khuyến nghị của chúng tôi để [xử lý các giao dịch mua mới](https://developer.android.com/google/play/billing/integrate#process). Khi đăng ký được mua, đăng ký sau đó được trả lại bởi [BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases)và thông báo SUBSCRIPTION\_PURCHASED được gửi. Khi bạn nhận được thông báo này, ứng dụng của bạn nên truy vấn API nhà phát triển Google Play để có trạng thái đăng ký mới [nhất](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions). Tài nguyên đăng ký trông tương tự như sau:

{Gia hạn  


Lưu ý: Gia hạn đăng ký không cần phải được xác nhận.

Nếu đăng ký gia hạn thành công, đăng ký tiếp tục được trả lại [bởi BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases).

Một SUBSCRIPTION\_RENEWED báo cũng được gửi khi đăng ký gia hạn. Ứng dụng của bạn phải đảm bảo người dùng vẫn được hưởng đăng ký và sau đó cập nhật trạng thái đăng ký với hết hạn mớiTimeMillis được cung cấp trong [tài nguyên đăng ký](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) được trả về từ API nhà phát triển của Google Play. Tài nguyên đăng ký trông tương tự như sau:

{Hết hạn

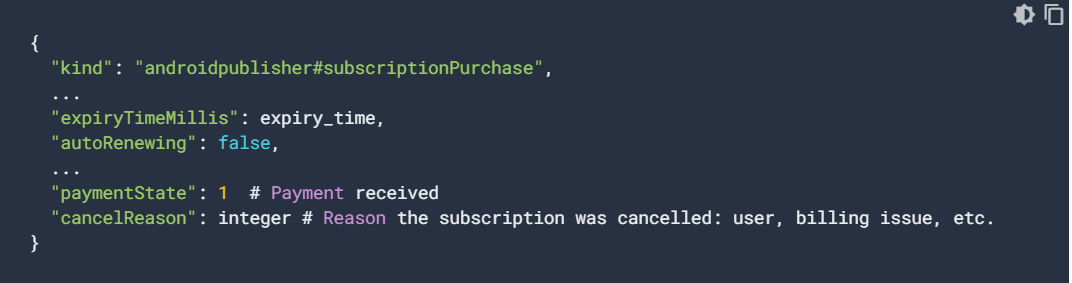
Khi đăng ký hết hạn, đăng ký không còn được trả về [trong BillingClient.queryPurchases() và](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases) người dùng sẽ mất quyền truy cập vào đăng ký.

Thông báo SUBSCRIPTION\_EXPIRED cũng được gửi khi đăng ký hết hạn. Khi bạn nhận được thông báo này, ứng dụng của bạn nên truy vấn API nhà phát triển Google Play để có trạng thái đăng [ký mới nhất](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions). Tài nguyên đăng ký trông tương tự như sau:

{Hủy bỏ  


Người dùng có thể tự nguyện hủy đăng ký khỏi Cửa hàng Play hoặc tự động hủy đăng ký nếu họ không khôi phục sau khi bị [giữ tài khoản](https://developer.android.com/google/play/billing/subscriptions#account-hold). Khi người dùng hủy đăng ký, họ sẽ giữ quyền truy cập vào nội dung cho đến khi kết thúc chu kỳ thanh toán hiện tại. Khi chu kỳ thanh toán kết thúc, quyền truy cập sẽ bị thu hồi.

Khi đăng ký bị hủy nhưng chưa hết [hạn](https://developer.android.com/google/play/billing/subscriptions#expirations), nó được trả về từ [BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases). Hủy đăng ký sẽ kích hoạt thông báo SUBSCRIPTION\_CANCELED bạn. Khi bạn nhận được thông báo này, tài [nguyên đăng ký được](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) trả về từ API nhà phát triển của Google Play chứa autoRenewing = false và expiryTimeMillis chứa ngày người dùng sẽ mất quyền truy cập vào đăng ký. Nếu hết hạnTimeMillis trong quá khứ, thì người dùng sẽ mất quyền lợi ngay lập tức. Nếu không, người dùng nên giữ quyền lợi cho đến khi hết hạn. Tài nguyên đăng ký trông tương tự như sau:

{Ứng dụng bạn có thể xem cancelReason trong tài nguyên đăng   
[ký được](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) trả về từ API nhà phát triển trên Google Play để tìm hiểu lý do hủy đăng ký (ví dụ: khách hàng đã hủy hoặc gặp vấn đề về thanh toán). Nếu người dùng hủy đăng ký, ứng dụng của bạn có thể xem trường cancelSurveyResult để tìm hiểu lý do người dùng hủy đăng ký.

Ứng dụng của bạn có thể muốn hiển thị thông báo trong ứng dụng thông báo cho người dùng rằng gói đăng ký của họ đã bị hủy, chẳng hạn như "Gói đăng ký của bạn sẽ hết hạn vào \_somedate. Ứng dụng của bạn cũng có [thể liên kết sâu đến Cửa hàng Google Play để cho](https://developer.android.com/google/play/billing/subscriptions#deep-link) phép người dùng khôi phục [gói đăng ký của họ](https://developer.android.com/google/play/billing/subscriptions#restore).

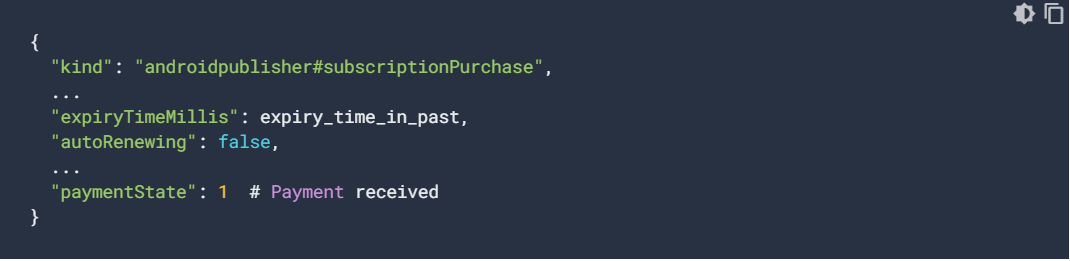
Nếu bạn hiển thị thông báo này, bạn cũng nên cung cấp cho người dùng khả năng loại bỏ vĩnh viễn thư.

Cũng lưu ý rằng thông báo hủy có thể khiến người dùng thất vọng, đặc biệt là người dùng đã hủy đăng ký theo cách thủ công thay vì hủy đăng ký của họ vì khoản thanh toán của họ đã lỗi thời. Bạn có thể chọn không thông báo cho người dùng đã hủy đăng ký theo cách thủ công.

Cảnh báo: Không xóa gói đăng ký khỏi Google Play trong khi bất kỳ người dùng nào vẫn có quyền sử dụng nội dung đó. Xóa nội dung mà người dùng có quyền bị phạt.

### Thu hồi

Đăng ký có thể bị thu hồi từ người dùng vì nhiều lý do, bao gồm cả ứng dụng của bạn thu hồi đăng ký bằng [cách sử dụng Purchases.subscriptions:revoke](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/revoke) hoặc giao dịch mua bị tính phí lại. Trong trường hợp này, ứng dụng của bạn nên thu hồi quyền lợi từ người dùng ngay lập tức. Đăng ký bị thu hồi không còn được trả lại từ [BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases). Thông báo SUBSCRIPTION\_REVOKED cũng được gửi khi điều này xảy ra. Khi bạn nhận được thông báo này, tài [nguyên đăng ký được](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) trả về từ API nhà phát triển của Google Play chứa autoRenewing = false và expiryTimeMillis chứa ngày người dùng sẽ mất quyền truy cập vào đăng ký. Tài nguyên đăng ký trông tương tự như sau:

{  


### Giữ tài khoản

Giữ tài khoản là trạng thái đăng ký bắt đầu khi hình thức thanh toán của người dùng không thành công và bất kỳ thời gian gia hạn liên [quan nào đã kết thúc](https://developer.android.com/google/play/billing/subscriptions#grace) mà không có giải pháp thanh toán. Khi đăng ký bị tạm dừng, bạn nên chặn quyền truy cập vào nội dung hoặc dịch vụ của mình. Thời gian giữ tài khoản kéo dài tối đa 30 ngày.

Trong quá trình giữ tài khoản, đăng ký không được [trả về bởi BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases).

Trong quá trình giữ tài khoản, bạn nên [xử lý bất](https://developer.android.com/google/play/billing/subscriptions#cancel)kỳ [hủy](https://developer.android.com/google/play/billing/subscriptions#restore), khôi phục hoặc mua lại đăng ký của bạn khicần thiết.

Lưu ý: Nếu người dùng sửa phương thức thanh toán của họ và khôi phục đăng ký của họ, mã thông báo mua hàng giống hệt với trước khi tài khoản của người dùng bắt đầu. Tuy nhiên, có thể người dùng đã lấy lại quyền truy cập vào nội dung đã đăng ký bằng cách mua lại đăng ký trong thời gian giữ. Trong trường hợp này, giá trị mã thông báo mua mới được trả về để đại diện cho phiên bản đăng ký mới này.

Khi người dùng tham gia vào việc giữ tài khoản, bạn nên tận dụng thông báo của nhà phát triển theo thời gian thực để thông báo cho người dùng của bạn tại sao quyền truy cập vào đăng ký bị tạm ngưng. Trong ứng dụng của mình, bạn nên cung cấp thông báo hướng dẫn về cách khắc phục phương thức thanh toán của họ và lấy lại quyền truy cập vào gói đăng ký. Tin nhắn của bạn phải bao [gồm liên kết đến cài đặt gói đăng ký Google Play](https://developer.android.com/google/play/billing/subscriptions#deep-link) để họ có thể sửa phương thức thanh toán của mình. Ví dụ: bạn có thể sử dụng thư tương tự như sau:

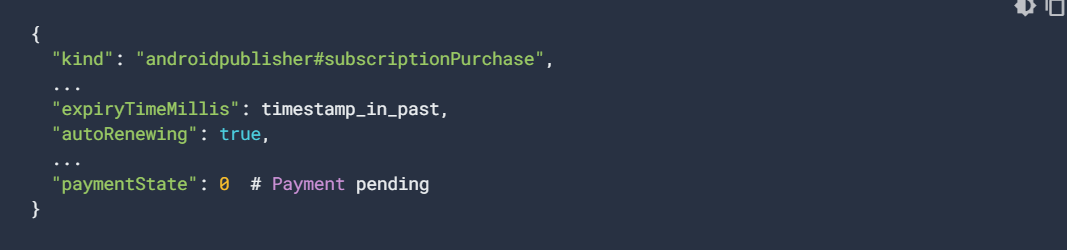
"Có một vấn đề với đăng ký của bạn. Nhấp vào đây để truy cập cài đặt đăng kýGoogle Play để sửa phương thức thanh toán của bạn."

Nếu người dùng của bạn có thể truy cập nội dung đăng ký bên ngoài ứng dụng của bạn, bạn có thể muốn gửi thông báo đẩy hoặc email cho người dùng để cho họ biết rằng đăng ký của họ không còn hoạt động nữa.

Nếu khách hàng có thể khắc phục sự cố thanh toán, bạn có thể hiển thị thông báo trong ứng dụng thông báo cho người dùng khi gói đăng ký của họ được khôi phục. Ví dụ: bạn có thể sử dụng thư tương tự như sau:

"Hình thức thanh toán của bạn đã được cập nhật và đăng ký của bạn đã được khôi phục."

Với [thông báo của nhà phát triển theo thời](https://developer.android.com/google/play/billing/getting-ready#configure-rtdn)gianthực, bạn sẽ nhận được SUBSCRIPTION\_ON\_HOLD báo khi đăng ký vào trạng khoản tạm giữ. Gọi API nhà [phát triển Google Play từ](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/get) máy chủ phụ trợ bảo mật của bạn để truy xuất thông tin đăng ký mới. Trong quá trình giữ tài khoản, expiryTimeMillis của tài [nguyên đăng](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) ký được đặt thành dấu thời gian trước đó và paymentState được đặt thành 0:

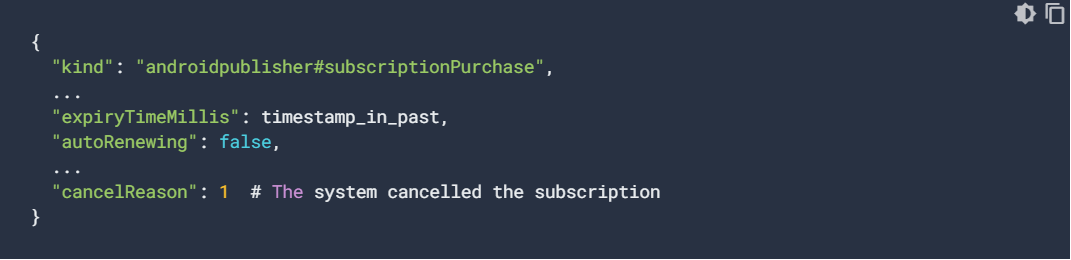
{  


Khi người dùng sửa phương thức thanh toán của họ, đăng ký trở về trạng thái hoạt động và sau đó bạn phải khôi phục quyền truy cập vào nội dung đã đăng ký.

Nếu ứng dụng của bạn chỉ dựa vào [queryPurchases() để xác](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases) định xem người dùng có được quyền đăng ký hay không, thì ứng dụng của bạn sẽ tự động xử lý đăng ký khôi phục sau khi giữ tài khoản.

Nếu ứng dụng của bạn đồng bộ hóa trạng thái đăng ký với backend, ứng dụng của bạn nên nghe thông báo SUBSCRIPTION\_RECOVERED được thông báo khi đăng ký được khôi phục và người dùng sẽ lấy lại quyền truy cập. Nếu bạn truy vấn đăng ký sau khi nhận được thông báo này, expiryTimeMillis được đặt thành dấu thời gian trong tương lai và paymentState là 1:

{Nếu người dùng không sửa phương thức thanh toán của họ trước khi kết thúc thời gian giữ tài khoản, bạn sẽ nhận được thông báo SUBSCRIPTION\_CANCELED nhà phát triển theo thời gian thực. Để biết hướng dẫn xử lý hủy, hãy [xem hủy đăng ký](https://developer.android.com/google/play/billing/subscriptions#cancel). Khi bạn truy vấn đăng ký đã bị hủy theo cách này, hết hạn trả vềTimeMillis được đặt thành dấu thời gian quá khứ và cancelReason có giá trị 1:

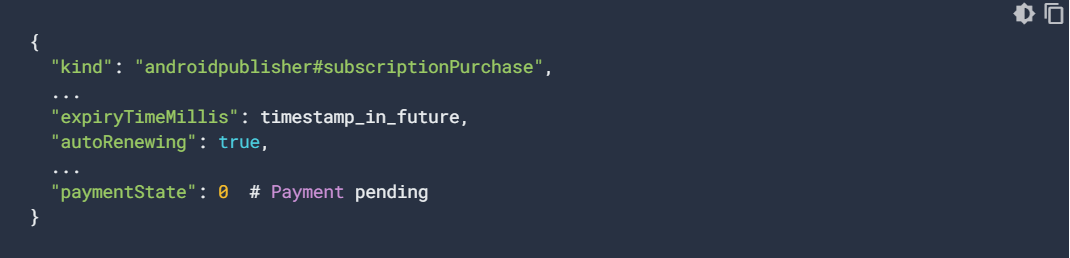
{Thời gian ân hạn  


Lưu ý: Theo mặc định, tất cả các gói đăng ký đều đã bật thời gian gia hạn. Bạn có thể tắt thời gian gia hạn cho SKU trong Google Play Console.

Nếu thời gian gia hạn được bật, đăng ký sẽ bước vào thời gian gia hạn nếu có vấn đề về thanh toán vào cuối chu kỳ thanh toán. Trong thời gian này, người dùng vẫn có quyền truy cập vào gói đăng ký trong khi Google Play cố gắng gia hạn đăng ký. Bạn có thể chỉ định thời gian gia hạn từ cài đặt sản phẩm trong ứng dụng trong Google Play Console.

Nếu ứng dụng của bạn chỉ dựa vào [queryPurchases() để kiểm tra](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases) xem người dùng có được hưởng đăng ký hay không, thì ứng dụng của bạn sẽ tự động xử lý thời gian gia hạn, vì queryPurchases() tiếp tục trả lại các giao dịch mua bị hủy trước ngày hết hạn.

Nếu ứng dụng của bạn đồng bộ hóa trạng thái đăng ký với backend, ứng dụng của bạn nên nghe thông báo của nhà phát triển thời gian thực SUBSCRIPTION\_IN\_GRACE\_PERIOD để được thông báo khi người dùng bước vào thời gian gia hạn. Trong khi người dùng đang trong thời gian gia hạn, tài [nguyên đăng ký](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) chứa autoRenewing = true và paymentState = 0 (tức là đang chờ xử lý).

Lưu ý: Google Play tự động gia hạn giá trị ExpiryTimeMillis cho đến khi hết thời gian gia hạn.

Khi người dùng bước vào thời gian gia hạn, bạn sẽ hiển thị thông báo trong ứng dụng cho người dùng biết cách khắc phục phương thức thanh toán của họ. Nếu không, người dùng mất quyền truy cập vào đăng ký khi thời gian gia hạn kết thúc. Thông báo này có [thể liên kết sâu đến Cửa hàng Google Play để](https://developer.android.com/google/play/billing/subscriptions#deep-link) giúp người dùng quản lý gói đăng ký của họ.

Ngay sau khi người dùng sửa phương thức thanh toán của họ, đăng ký sẽ gia hạn và ứng dụng của bạn có thể xử lý gia hạn như được mô tả [trong gia hạn](https://developer.android.com/google/play/billing/subscriptions#renewals).

Nếu người dùng không sửa phương thức thanh toán của họ trong thời gian gia hạn, đăng ký vào tài [khoản giữ](https://developer.android.com/google/play/billing/subscriptions#account-hold).

### Đăng ký bị tạm dừng

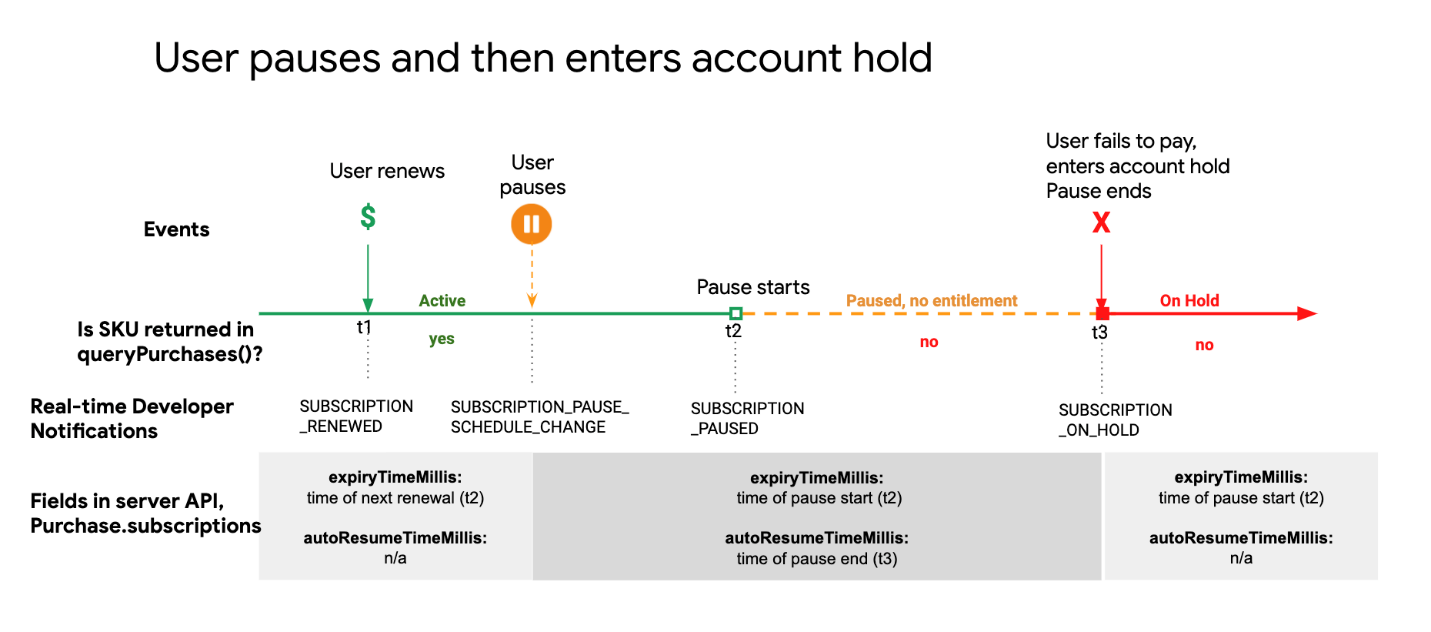
Lưu ý: Tất cả các gói đăng ký đã tạm dừng được bật theo mặc định. Bạn có thể tắt tính năng tạm dừng [từ Google Play Console](https://support.google.com/googleplay/android-developer/answer/140504#pauses).

Bạn có thể ngăn chặn sự khuấy động tự nguyện bằng cách cho phép người dùng tạm dừng đăng ký của họ. Khi bạn bật tính năng tạm dừng, người dùng có thể chọn tạm dừng đăng ký của họ trong khoảng thời gian từ một tuần đến ba tháng, tùy thuộc vào khoảng thời gian định kỳ. Sau khi bật, tùy chọn tạm dừng sẽ xuất hiện cả trong trung tâm đăng ký và trong quy trình hủy. Lưu ý rằng bạn không thể tạm dừng đăng ký hàng năm và giới hạn tạm dừng là một tuần và ba tháng có thể thay đổi bất cứ lúc nào.

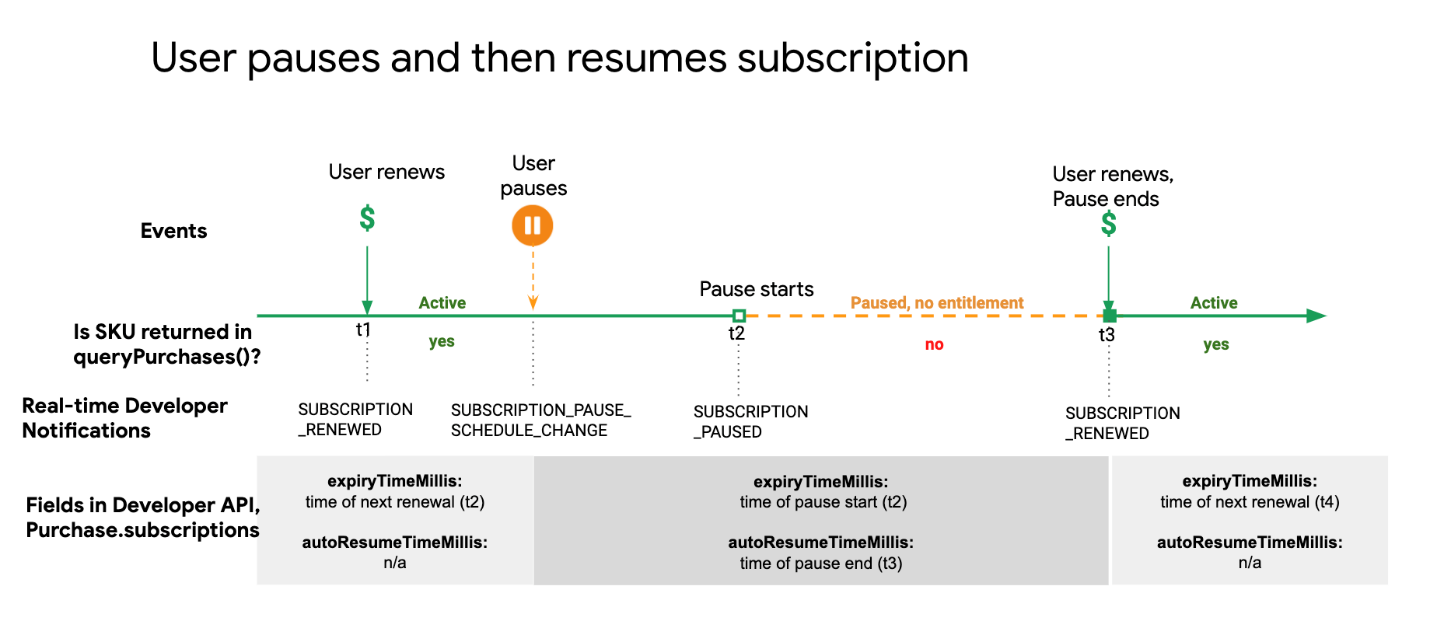
Để cho phép người dùng tạm dừng đăng ký của họ, hãy làm như sau:

1. Đăng nhập vào [Google Play Console](https://play.google.com/console).
2. Chọn ứng dụng của bạn, sau đó đi tới Lưu trữ sự hiện diện > sản phẩm trong ứng dụng > đăng ký.
3. Mở rộng phần Thiết đặt gói đăng ký.
4. Chọn Activate Pause (Kích hoạt tạm dừng).

Tạm dừng đăng ký chỉ có hiệu lực sau khi thời hạn thanh toán hiện tại kết thúc. Trong khi đăng ký bị tạm dừng, người dùng không có quyền truy cập vào đăng ký. Vào cuối thời gian tạm dừng, gói đăng ký sẽ tiếp tục và Google cố gắng gia hạn đăng ký. Nếu sơ yếu lý lịch thành công, đăng ký sẽ hoạt động trở lại. Nếu sơ yếu lý lịch không thành công do sự cố thanh toán, người dùng vào trạng thái giữ tài khoản, như trong hình 1:

Biểu đồ 1. Người dùng tạm dừng đăng ký của họ và sau đó nhập giữ tài khoản.

Người dùng cũng có thể chọn tiếp tục đăng ký theo cách thủ công bất cứ lúc nào trong khoảng thời gian tạm dừng, như trong hình 2. Khi người dùng tiếp tục theo cách thủ công, ngày thanh toán sẽ thay đổi thành ngày tiếp tục thủ công.

Biểu đồ 2. Người dùng tạm dừng và sau đó tiếp tục đăng ký của họ.

Khi đăng ký của người dùng bị tạm dừng, đăng ký không được trả về bởi [queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases). Nếu đăng ký được tiếp tục, đăng ký sau đó được trả về bởi queryPurchases().

Nếu ứng dụng của bạn đồng bộ hóa trạng thái đăng ký với máy chủ phụ trợ an toàn, ứng dụng của bạn nên nghe thông báo của nhà phát triển thời gian thực để duy trì trạng thái. Những thông báo này cũng cho phép bạn thông báo cho người dùng trong ứng dụng của mình rằng họ đã tạm dừng đăng ký và không có quyền truy cập vào đăng ký đó. Bạn cũng nên cung cấp một cách để người dùng tiếp tục đăng ký theo cách thủ công bằng cách sử dụng [liên kết sâu đến Google Play](https://developer.android.com/google/play/billing/subscriptions#deep-link).

Thông báo SUBSCRIPTION\_PAUSE\_SCHEDULE\_CHANGED nhà phát triển theo thời gian thực được gửi khi người dùng của bạn bắt đầu tạm dừng đăng ký của họ. Tại thời điểm này, người dùng nên tiếp tục truy cập vào đăng ký của họ và [tài nguyên đăng](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) ký chứa autoRenewing = true, paymentState = 1 (Đã nhận thanh toán) và các giá trị trong tương lai cho hết hạnTimeMillis và autoResumeTimeMillis.

Thông báo SUBSCRIPTION\_PAUSED nhà phát triển theo thời gian thực được gửi khi tạm dừng có hiệu lực. Tại thời điểm này, người dùng sẽ mất quyền truy cập vào đăng ký của họ và [tài nguyên đăng](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) ký chứa autoRenewing = true và paymentState = 0 (đang chờ xử lý), giá trị trong tương lai cho autoResumeTimeMillis và giá trị quá khứ cho expiryTimeMillis.

Thông báo SUBSCRIPTION\_RENEWED nhà phát triển thời gian thực được gửi đi nếu đăng ký được tiếp tục tự động vào cuối khoảng thời gian tạm dừng hoặc nếu người dùng chọn tiếp tục đăng ký theo cách thủ công. Điều này nên được xử lý như được mô tả [trong gia hạn](https://developer.android.com/google/play/billing/subscriptions#renewals).

Thông SUBSCRIPTION\_ON\_HOLD nhà phát triển thời gian thực được gửi nếu có lỗi thanh toán trong khi cố gắng tiếp tục đăng ký. Điều này nên được xử lý như được mô tả trong [phần giữ tài khoản](https://developer.android.com/google/play/billing/subscriptions#account-hold).

### Phục hồi

Lưu ý: Hỗ trợ khôi phục đăng ký là bắt buộc đối với tất cả các nhà phát triển.

Gói đăng ký bị hủy vẫn hiển thị trong ứng dụng Cửa hàng Play cho đến ngày hết hạn. Người dùng có thể khôi phục gói đăng ký đã hủy trước khi hết hạn bằng cách nhấp vào Đăng ký lại (trước đó là Khôi phục) trong phần Gói đăng ký trong ứng dụng Cửa hàng Google Play.

subscriptions section in the google play store app showing a
            cancelled subscription with a resubscribe buttonBiểu đồ 3. Tài khoản > đăng ký trong ứng dụng Cửa hàng Google Play hiển thị gói đăng ký bị hủy bằng nút Đăng ký lại.

Nếu ứng dụng của bạn chỉ dựa vào [queryPurchases() để xác](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases) định xem người dùng có được quyền đăng ký hay không, thì ứng dụng của bạn sẽ tự động xử lý các khôi phục, vì queryPurchases() tiếp tục trả lại các giao dịch mua bị hủy trước ngày hết hạn. Đăng ký được khôi phục tiếp tục gia hạn như thể nó không bị hủy bỏ.

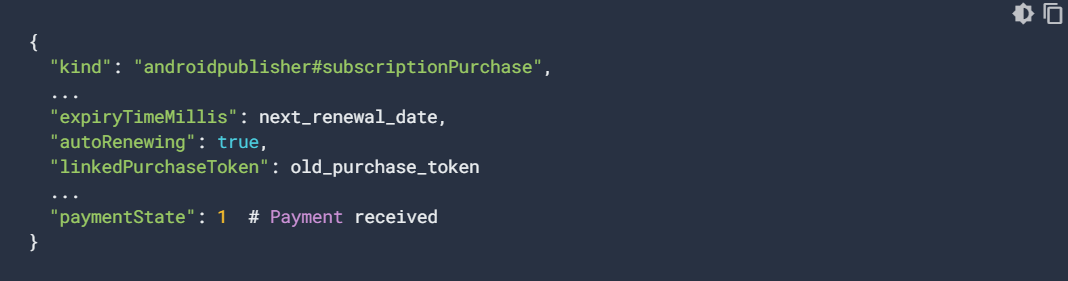
Nếu ứng dụng của bạn đồng bộ hóa trạng thái đăng ký với backend, ứng dụng của bạn sẽ nghe thông báo SUBSCRIPTION\_RESTARTED nhà phát triển theo thời gian thực. Sau khi nhận được, ứng dụng của bạn có thể trả lời thông báo, ghi lại rằng đăng ký hiện được đặt để gia hạn và ngừng hiển thị thông điệp khôi phục trong ứng dụng của bạn. Tài nguyên đăng ký trông tương tự như sau:

Lưu ý: Đăng ký được khôi phục sử dụng cùng một purchaseToken từ khi đăng ký bị hủy. Tất cả các trường hủy được xóa khỏi tài nguyên.

### Nâng cấp, hạ cấp và từ chức

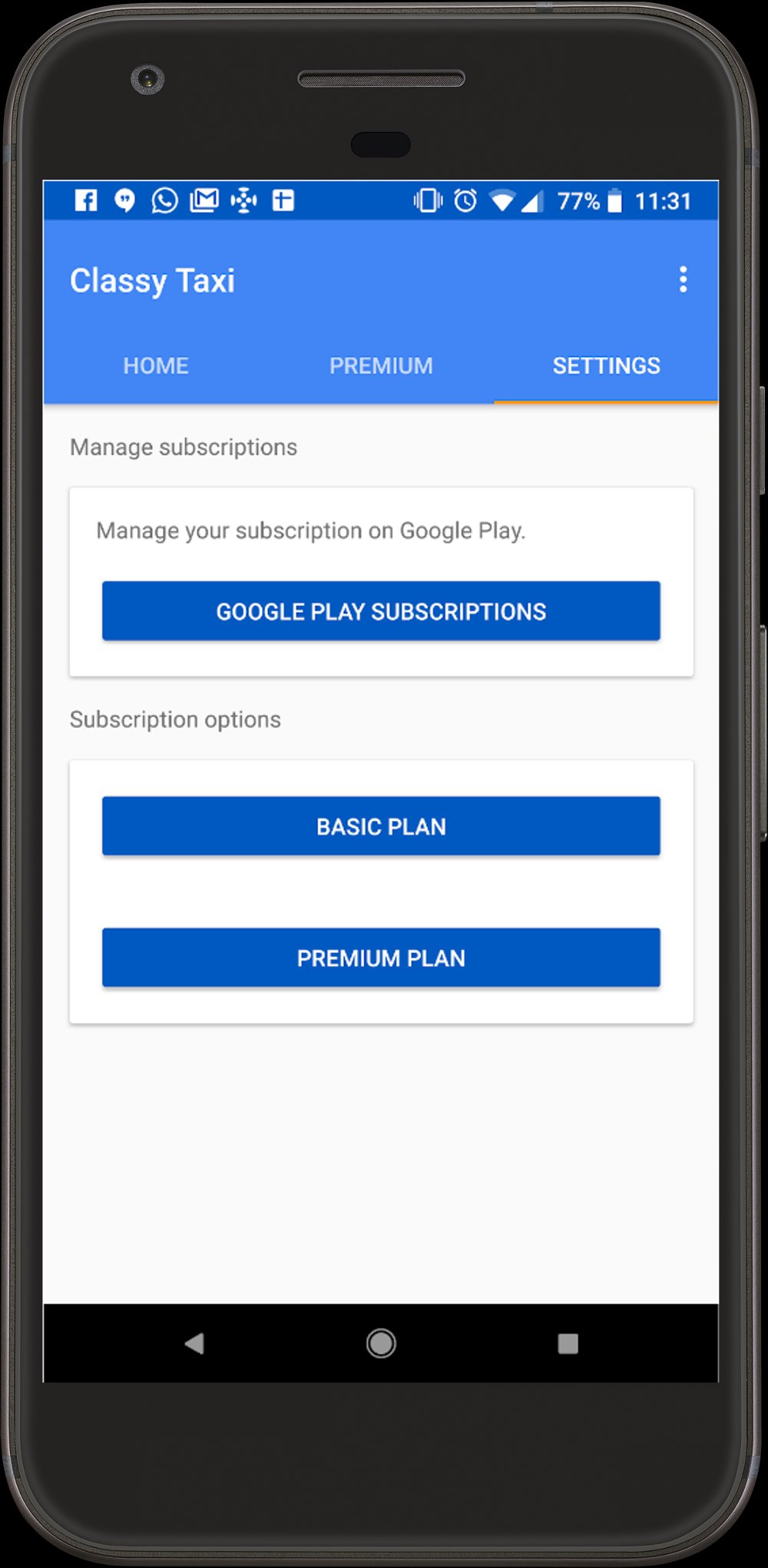
Khi người dùng [nâng](https://developer.android.com/google/play/billing/subs#change)cấp , [hạ](https://developer.android.com/google/play/billing/subs#change)cấp hoặc đăng [ký lại](https://developer.android.com/google/play/billing/subs#resubscribe), đăng ký cũ bị vô hiệu hóa vàđăng ký [mới được](https://developer.android.com/google/play/billing/subscriptions#new) tạo bằng mã thông báo mua mới.

Ngoài ra, tài nguyên [đăng ký được](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) trả về từ API nhà phát triển trên Google Play sẽ chứa api Nhà phát triển Được liên kếtpurchaseToken cho biết giao dịch mua cũ mà người dùng đã nâng cấp, hạ cấp hoặc đăng ký lại. Bạn có thể sử dụng linkedPurchaseToken để tra cứu đăng ký cũ và xác định tài khoản người dùng hiện có để bạn có thể liên kết giao dịch mua mới với cùng một tài khoản. Bạn cũng nên sử dụng API nhà phát triển của Google Play để xác nhận giao dịch mua để [giảm](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/acknowledge) ma sát người dùng. Tài nguyên đăng ký trông tương tự như sau:

{  


## Sử dụng liên kết sâu để cho phép người dùng quản lý đăng ký

Là nhà phát triển, bạn phải giúp khách hàng dễ dàng quản lý đăng ký của họ. Ứng dụng của bạn phải bao gồm liên kết trên màn hình cài đặt hoặc tùy chọn cho phép người dùng quản lý gói đăng ký của họ. Một ví dụ về liên kết này được hiển thị trong hình 4.



Biểu đồ 4. Nút Đăng ký trên Google Play là ví dụ về liên kết "Quản lý gói đăng ký".

Trong trình xử lý nhấp chuột của liên kết này, hãy thêm logic để xác định xem người dùng có bất kỳ đăng ký không hết hạn nào cho ứng dụng của bạn hay không, trong đó hết hạnTimeMillis trong tương lai hoặc autoRenewing được đặt thành true.

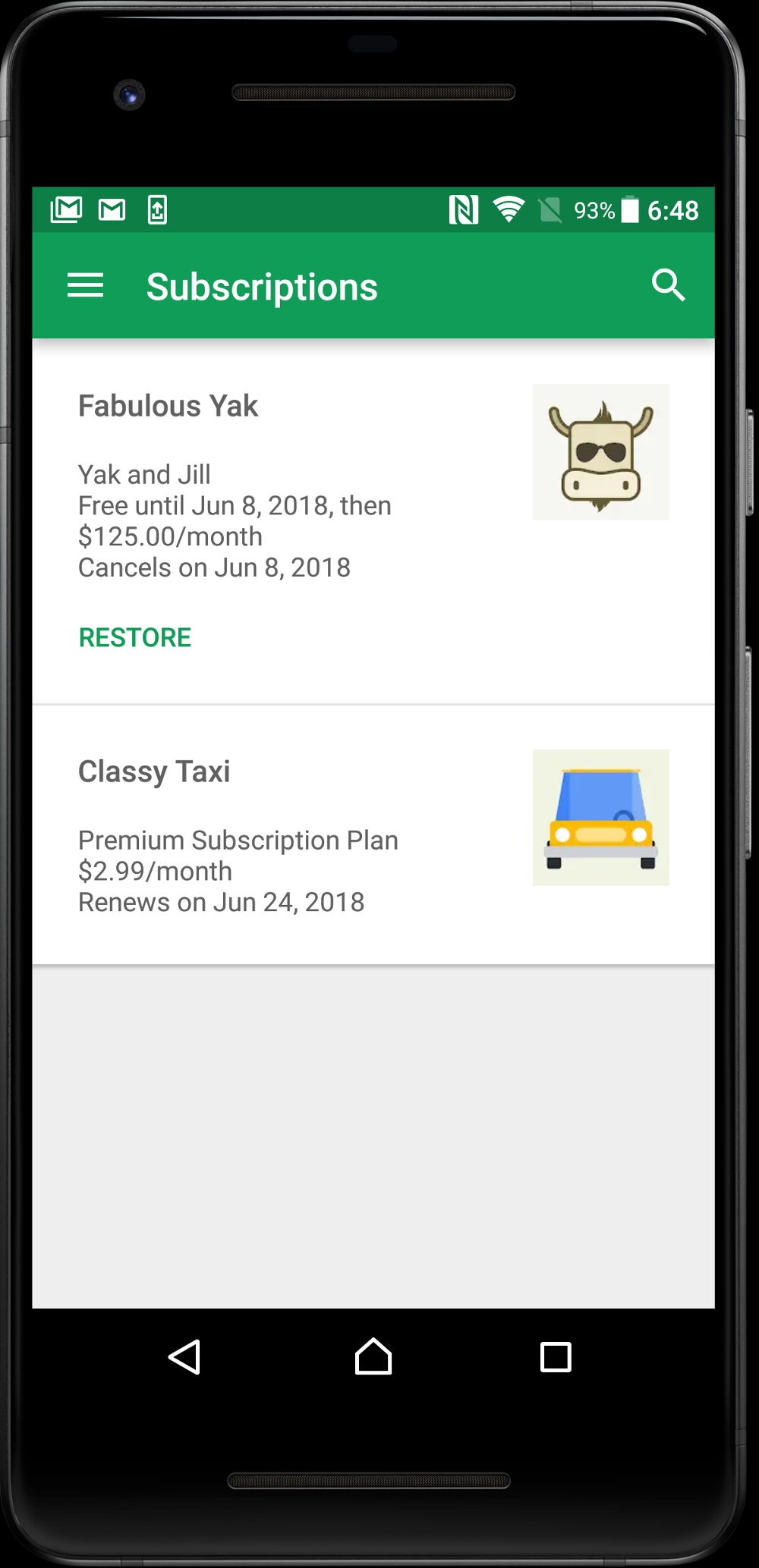
SKU của mỗi gói đăng ký khớp với ID sản phẩm mà bạn đã chỉ định khi tạo trong Play Console. Để xác định SKU theo lập trình cho đăng ký hiện có, hãy truy vấn phụ trợ của ứng dụng để biết danh sách các gói đăng ký được liên kết với một người dùng cụ thể.

Nếu người dùng có gói đăng ký không hết hạn, bạn có thể chuyển hướng họ đến URL tương tự như sau, thay thế "your-sub-product-id" và "your-app-package" bằng ID đăng ký và thông tin gói ứng dụng:

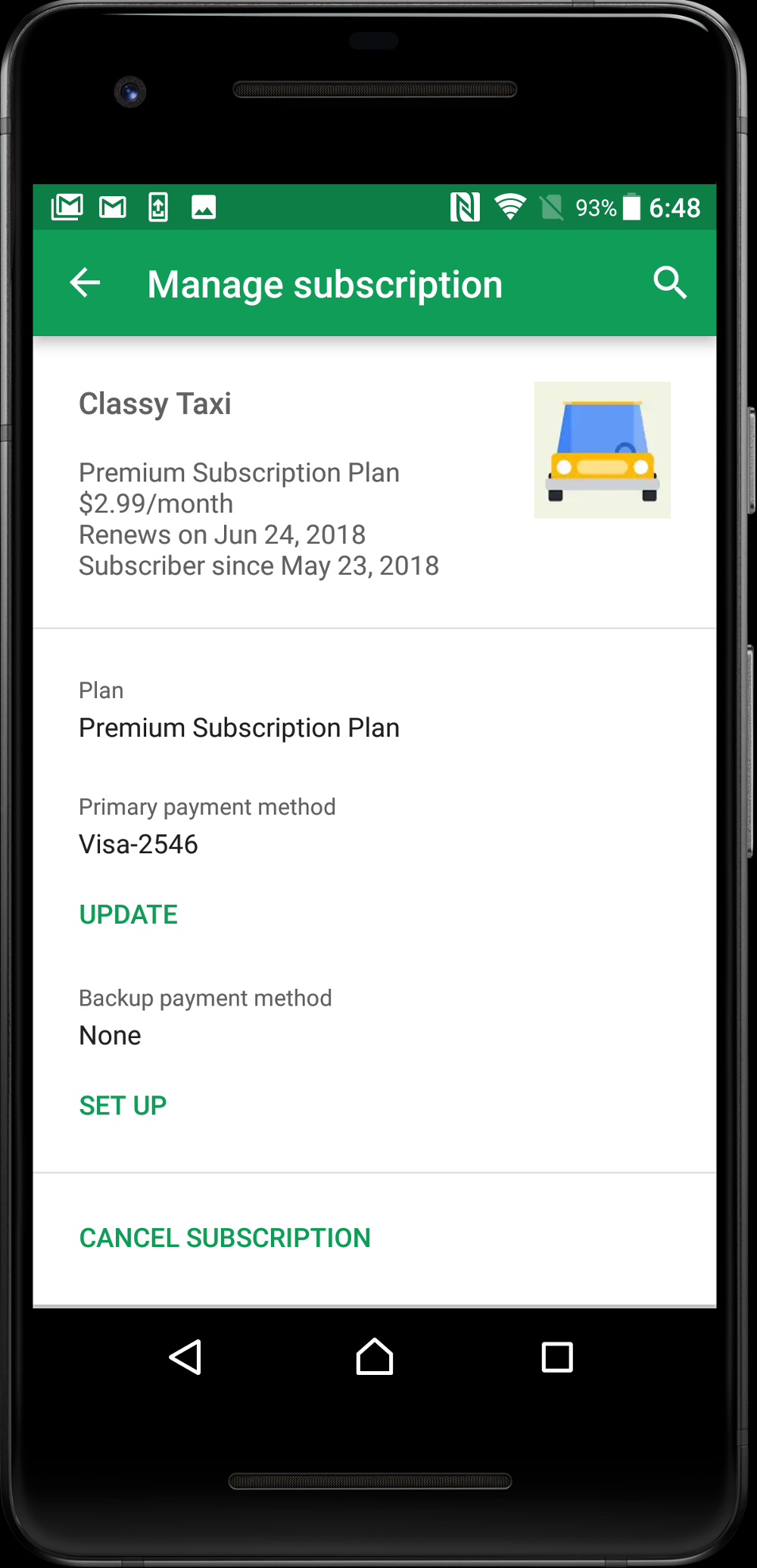
https://play.google.com/store/account/subscriptions?sku=your-sub-product-id&package=your-app-package

Nếu người dùng không có bất kỳ đăng ký không hết hạn nào trong ứng dụng của bạn, hãy sử dụng URL sau để hướng họ đến trang hiển thị tất cả các gói đăng ký khác của họ, như trong hình 5 và 6:

<https://play.google.com/store/account/subscriptions>



Biểu đồ 5. Màn hình đăng ký Cửa hàng Play hiển thị trạng thái cho tất cả đăng ký của người dùng.

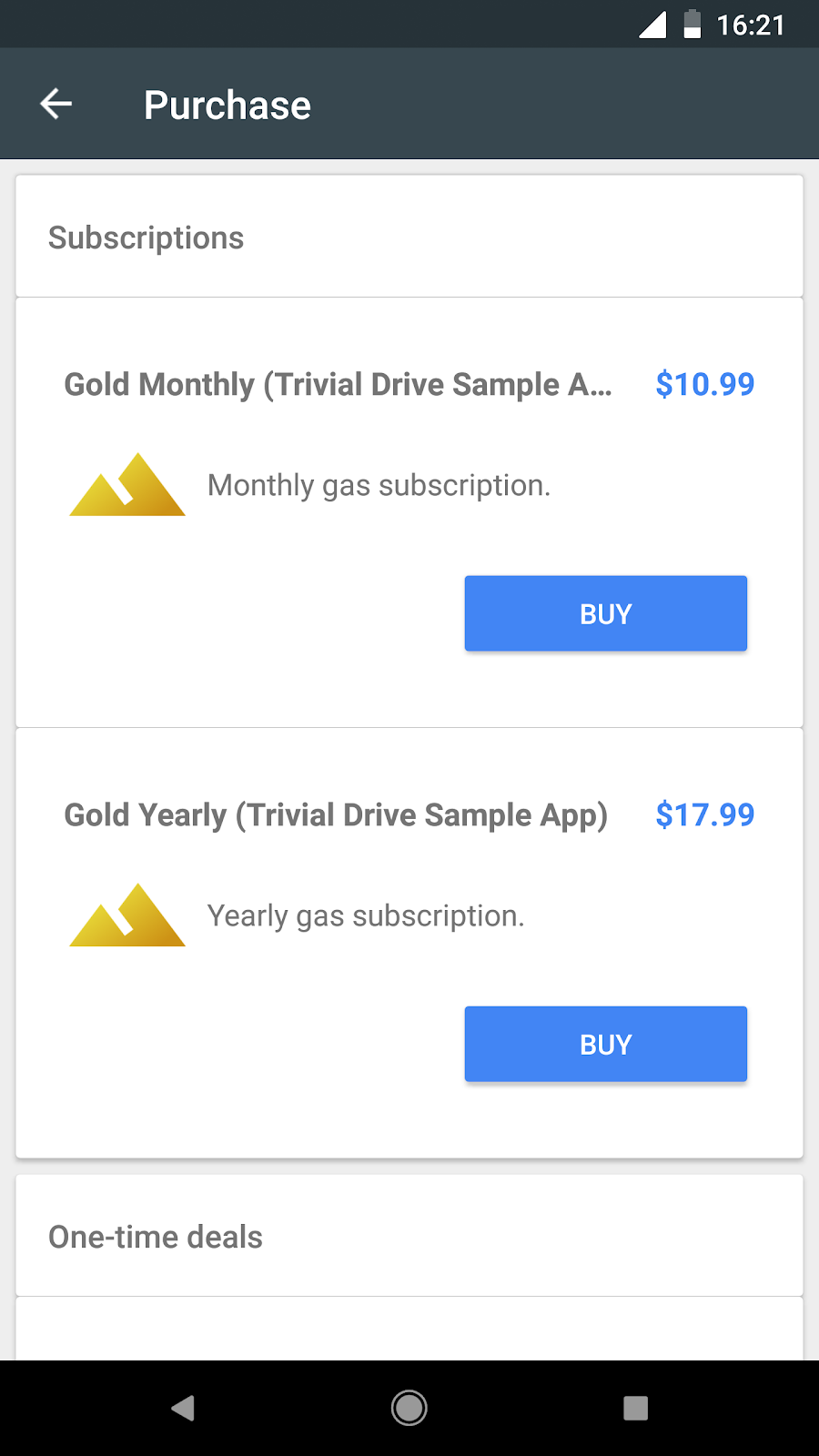


Biểu đồ 6. Nhấn vào đăng ký để xem thêm chi tiết.

Bạn có thể tìm thấy mã ví dụ cho logic liên kết đăng ký [trong ứng dụng mẫu Classy Taxi](https://github.com/android/play-billing-samples/blob/52f85ddb6acc019b6b405ca68956d698ca96d640/ClassyTaxiJava/app/src/main/java/com/example/android/classytaxijava/ui/MainActivity.java#L119-L136).

## Cho phép người dùng nâng cấp, hạ cấp hoặc thay đổi đăng ký của họ

Bạn có thể cung cấp cho người dùng các bậc đăng ký khác nhau, chẳng hạn như bậc cơ sở và bậc cao cấp. Hình 7 hiển thị màn hình cung cấp hai bậc đăng ký:

Biểu đồ 7. Ứng dụng này có hai bậc đăng ký.

Người dùng sẽ có thể truy cập màn hình tương tự như hình 7 để nâng cấp hoặc hạ cấp đăng ký. Khi nâng cấp hoặc hạ cấp đăng ký, bạn có thể đặt chế độ proration hoặc cách thay đổi ảnh hưởng đến người đăng ký của bạn. Bảng sau liệt kê các chế độ proration có sẵn:

|  |  |
| --- | --- |
| Chế độ proration | Mô tả |
| IMMEDIATE\_WITH\_TIME\_PRORATION | Đăng ký được nâng cấp hoặc hạ cấp ngay lập tức. Bất kỳ thời gian nào còn lại đều được điều chỉnh dựa trên chênh lệch giá và được ghi có vào đăng ký mới bằng cách đẩy ngày thanh toán tiếp theo. Đây là hành vi mặc định. |
| IMMEDIATE\_AND\_CHARGE\_PRORATED\_PRICE | Đăng ký được nâng cấp ngay lập tức và chu kỳ thanh toán vẫn giữ nguyên. Chênh lệch giá cho khoảng thời gian còn lại sau đó được tính cho người dùng. |
| IMMEDIATE\_WITHOUT\_PRORATION | Đăng ký được nâng cấp hoặc hạ cấp ngay lập tức và giá mới được tính khi đăng ký gia hạn. Chu kỳ thanh toán vẫn giữ nguyên. |
| Hoãn | Đăng ký chỉ được nâng cấp hoặc hạ cấp khi đăng ký gia hạn. |

### Ví dụ về proration

Để hiểu cách mỗi chế độ proration hoạt động, hãy xem xét tình huống sau:

Samwise có đăng ký nội dung trực tuyến từ ứng dụng Country Gardener. Anh hiện có đăng ký hàng tháng cho phiên bản Cấp 1 của nội dung, chỉ dành cho văn bản. Đăng ký này tiêu tốn của anh ta 2 đô la mỗi tháng và nó sẽ gia hạn vào ngày đầu tiên của tháng.

Vào ngày 15 tháng 4, Samwise đã chọn nâng cấp lên phiên bản hàng năm của đăng ký Cấp 2, bao gồm cập nhật video và chi phí 36 đô la mỗi năm.

Khi nâng cấp đăng ký, nhà phát triển chọn chế độ proration. Danh sách sau đây mô tả cách mỗi chế độ proration ảnh hưởng đến đăng ký của Samwise:

IMMEDIATE\_WITH\_TIME\_PRORATION

Đăng ký Tier 1 của Samwise kết thúc ngay lập tức. Vì anh ấy đã trả tiền cho cả tháng (1-30 tháng 4) nhưng được nâng cấp nửa thời gian đăng ký, nửa tháng đăng ký ($ 1) được áp dụng cho đăng ký mới của anh ấy. Tuy nhiên, vì đăng ký mới đó có giá 36 đô la mỗi năm, số dư tín dụng 1 đô la chỉ trả trong 10 ngày (ngày 16-25 tháng 4), vì vậy vào ngày 26 tháng 36, anh ta bị tính phí 36 đô la cho đăng ký mới và 36 đô la khác vào ngày 26 tháng 4 hàng năm sau đó.

IMMEDIATE\_AND\_CHARGE\_PRORATED\_PRICE

Chế độ này có thể được sử dụng vì giá đăng ký Cấp 2 cho mỗi đơn vị thời gian ($ 36 / năm = $ 3 / tháng) lớn hơn giá đăng ký Cấp 1 cho mỗi đơn vị thời gian ($ 2 / tháng). Đăng ký Tier 1 của Samwise ngay lập tức kết thúc. Vì anh ta đã trả tiền cho cả tháng nhưng chỉ sử dụng một nửa số tiền đó, nửa tháng đăng ký (1 đô la) được áp dụng cho đăng ký mới của anh ta. Tuy nhiên, vì đăng ký mới đó có giá $ 36 / năm, 15 ngày còn lại có giá $ 1.50, vì vậy anh ta bị tính phí chênh lệch 0,50 đô la cho đăng ký mới và 35,50 đô la khác vào ngày 1 tháng 5. Sau đó, Samwise bị tính phí 36 đô la mỗi năm sau đó.

IMMEDIATE\_WITHOUT\_PRORATION

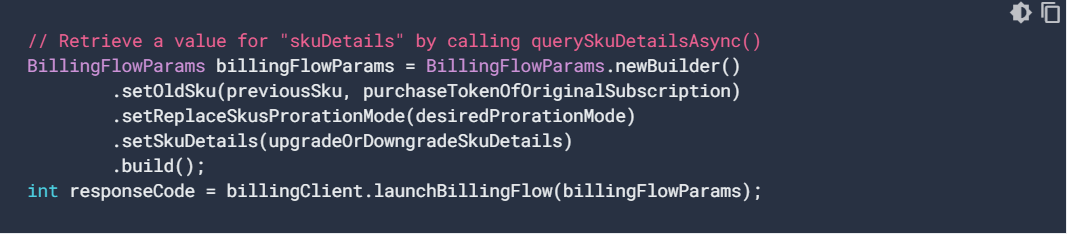
Đăng ký Tier 1 của Samwise ngay lập tức được nâng cấp lên Tier 2 mà không phải trả thêm phí và vào ngày 1 tháng 5, anh ta bị tính phí 36 đô la cho hạng đăng ký mới và 36 đô la khác vào ngày 1 tháng 5 mỗi năm sau đó.

Hoãn

Đăng ký Tier 1 của Samwise tiếp tục cho đến khi hết hạn vào ngày 30 tháng 4. Vào ngày 1 tháng 5, đăng ký Tier 2 có hiệu lực và Samwise bị tính phí 36 đô la cho cấp đăng ký mới của mình.

Khi chọn chế độ proration, hãy chắc chắn xem lại các khuyến nghị [proration của chúng tôi](https://developer.android.com/google/play/billing/subscriptions#proration-recommendations).

Ứng dụng của bạn có thể cung cấp cho người dùng bản nâng cấp hoặc hạ cấp bằng cách sử dụng các bước tương tự [như khi khởi chạy luồng mua hàng](https://developer.android.com/google/play/billing/integrate#launch). Tuy nhiên, khi nâng cấp hoặc hạ cấp, bạn cần cung cấp chi tiết cho đăng ký hiện tại, đăng ký trong tương lai (nâng cấp hoặc hạ cấp) và chế độ proration để sử dụng, như trong ví dụ sau:

Đối với các chế độ proration thay thế ngay lập tức, ứng dụng của bạn nhận được giao dịch mua [mới trong PurchasesUpdatedListener .](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener) Việc mua cũng có sẵn trong [BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases). Khi bạn nhận được mã thông báo mua hàng, hãy làm theo quy trình xác minh tương [tự](https://developer.android.com/google/play/billing/integrate#process) như với việc xác minh mã thông báo mua hàng mới. Đảm bảo xác nhận các giao dịch mua này với [BillingClient.acknowledgePurchase()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#acknowledgepurchase) từ Thư viện thanh toán của Google Play hoặc [Purchases.subscriptions:acknowledge](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/acknowledge) từ API nhà phát triển của Google Play.

API nhà phát triển Google Play trả về linkedPurchaseToken trong tài nguyên [đăng ký](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions). Hãy chắc chắn làm mất hiệu lực mã thông báo được cung cấp trong linkedPurchaseToken để đảm bảo rằng mã thông báo cũ không được sử dụng để có quyền truy cập vào các dịch vụ của bạn. Xem Nâng [cấp, hạ cấp và từ chức để biết thông tin về](https://developer.android.com/google/play/billing/subscriptions#upgrade-downgrade) cách xử lý nâng cấp và hạ cấp giao dịch mua.

Đối với chế độ thay thế hoãn lại, ứng dụng của bạn nhận được cuộc gọi [đến PurchasesUpdatedListener của bạn](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener) với danh sách giao dịch mua trống và trạng thái xem việc nâng cấp hoặc hạ cấp có thành công hay không. Cho đến khi việc thay thế có hiệu lực, [BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases) tiếp tục trả lại giao dịch mua cho gói đăng ký ban đầu. Khi gói mới có hiệu lực, queryPurchases() trả về dữ liệu mua cho đăng ký mới và thông báo SUBSCRIPTION\_RENEWED được gửi đến máy chủ backend an toàn của bạn. Đối với các thay thế hoãn lại, chúng tôi khuyên bạn nên lắng nghe thông báo này và xác nhận việc mua hàng bằng [Purchases.subscriptions:acknowledge](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/acknowledge). LinkedPurchaseToken trong tài nguyên đăng ký có thể được sử dụng để xác định người dùng nào trong backend đăng ký của bạn, nếu có, sẽ được cập nhật với quyền lợi mới. Ứng dụng của bạn không nên dựa vào người dùng mở ứng dụng và thừa nhận thông qua [BillingClient.acknowledgePurchase(),](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#acknowledgepurchase)vì người dùng có thể không mở ứng dụng trong vòng ba ngàykể từ khi thay đổi gói có hiệu lực.

### Nâng cấp với ưu đãi giá dùng thử hoặc giới thiệu miễn phí

[Cài đặt đủ](https://developer.android.com/google/play/billing/subscriptions#free-trial) điều kiện dùng thử miễn phí được áp dụng khi người dùng nâng cấp hoặc hạ cấp. Bạn có thể điều chỉnh các tùy chọn cài đặt đủ điều kiện dùng thử miễn phí trong Google Play Console.

Lưu ý những điều sau:

* Nếu người dùng chỉ có thể nhận được một bản dùng thử miễn phí trên tất cả các đăng ký có sẵn trong ứng dụng của bạn, gói mà người dùng đang thay đổi sẽ không có giá dùng thử hoặc giới thiệu miễn phí.
* Nếu bạn cung cấp một bản dùng thử miễn phí cho mỗi sản phẩm đăng ký, gói mà người dùng đang thay đổi có thể có giá dùng thử hoặc giới thiệu miễn phí.

Bảng sau mô tả hành vi của từng chế độ proration nếu cả gói mới và cũ đều có bản dùng thử miễn phí và người dùng đang nâng cấp trong thời gian dùng thử miễn phí:

|  |  |  |
| --- | --- | --- |
|  | Một bản dùng thử miễn phí cho mỗi ứng dụng | Một bản dùng thử miễn phí cho mỗi sản phẩm đăng ký |
| IMMEDIATE\_WITH\_TIME\_PRORATION | Người dùng mất bản dùng thử miễn phí ngay lập tức. Thời gian dùng thử miễn phí còn lại được chuyển đổi thành thời gian miễn phí tương đương của bậc mới dựa trên chênh lệch giá. | Người dùng mất bản dùng thử miễn phí trước đó nhưng ngay lập tức bắt đầu dùng thử miễn phí mới. Ngoài ra, thời gian dùng thử miễn phí còn lại của hạng cũ được chuyển đổi thành thời gian miễn phí tương đương của hạng mới và được thêm vào bản dùng thử miễn phí mới. |
| IMMEDIATE\_AND\_CHARGE\_PRORATED\_PRICE | Người dùng mất bản dùng thử miễn phí ngay lập tức. Chênh lệch giá cho khoảng thời gian còn lại sau đó được tính cho người dùng. Ngày thanh toán tiếp theo vẫn không thay đổi.  Lưu ý: Tùy chọn này chỉ khả dụng để nâng cấp đăng ký, trong đó giá trên mỗi đơn vị thời gian tăng lên. | |
| IMMEDIATE\_WITHOUT\_PRORATION | Người dùng ngay lập tức được nâng cấp lên bậc mới. Người dùng giữ quyền truy cập dùng thử miễn phí vào bậc mới cho đến khi thời hạn thanh toán trước đó kết thúc. | |
| Hoãn | Người dùng giữ quyền truy cập dùng thử miễn phí vào đăng ký cũ cho đến ngày thanh toán tiếp theo. | |

Để hiểu cách chuyển tiếp dùng thử miễn phí hoạt động trong trường hợp mặc định của một bản dùng thử miễn phí cho mỗi ứng dụng, hãy xem xét trường hợp sau:

Maria có đăng ký nội dung trực tuyến từ ứng dụng Country Gardener. Cô hiện có đăng ký hàng tháng cho phiên bản Cấp 1 của nội dung, chỉ dành cho văn bản. Đăng ký này tiêu tốn của cô ấy $ 10 / tháng và cô ấy đã đăng ký vào ngày 1 tháng 4. Cô đang tận hưởng bản dùng thử miễn phí 30 ngày với tư cách là người đăng ký lần đầu tiên, điều đó có nghĩa là khoản thanh toán đầu tiên của cô đến hạn vào ngày 1 tháng 5.

Vào ngày 15 tháng 4, Maria chọn nâng cấp lên đăng ký Tier 2, bao gồm cập nhật video và chi phí $ 20 / tháng. Đăng ký thứ hai này cũng có bản dùng thử 30 ngày.

Danh sách sau đây mô tả cách chuyển tiếp dùng thử miễn phí cho mỗi chế độ proration:

* IMMEDIATE\_WITH\_TIME\_PRORATION - Maria được nâng cấp lên Tier 2 ngay lập tức. Kể từ khi Maria nâng cấp giữa thời gian đăng ký, một nửa số đăng ký của tháng (15 ngày trị giá 10 đô la / tháng) được áp dụng cho đăng ký mới của cô. Tuy nhiên, vì đăng ký mới đó có giá $ 20 / tháng, số dư 15 ngày chỉ trả trong 7,5 ngày. Maria không đủ điều kiện để dùng thử miễn phí khác cho Cấp 2, vì vậy bắt đầu từ ngày 22 tháng 20, cô ấy bị tính phí 20 đô la mỗi tháng.
* IMMEDIATE\_AND\_CHARGE\_PRORATED\_PRICE - Chế độ này có thể được sử dụng vì giá đăng ký Cấp 2 trên mỗi đơn vị thời gian ($ 20 / tháng) lớn hơn giá đăng ký Cấp 1 cho mỗi đơn vị thời gian ($ 10 / tháng). Đăng ký Tier 1 của Maria ngay lập tức được nâng cấp lên Tier 2 và cô mất bản dùng thử miễn phí. Vì ngày thanh toán tiếp theo của Maria là ngày 1 tháng 5, cô bị tính phí 10 đô la hôm nay để trang trải nửa thứ 2 của tháng 4, sau đó bắt đầu từ ngày 1 tháng 20, cô bị tính phí 20 đô la mỗi tháng.
* IMMEDIATE\_WITHOUT\_PRORATION - Đăng ký Cấp 1 của Maria ngay lập tức được nâng cấp lên Tier 2. Maria giữ bản dùng thử miễn phí cho đến ngày 30 tháng 4 và hiện có quyền truy cập vào nội dung Cấp 2. Bắt đầu từ ngày 1 tháng 20, cô bị tính phí 20 đô la mỗi tháng.
* HOÃN LẠI - Đăng ký Cấp 1 của Maria tiếp tục cho đến khi khoản thanh toán tiếp theo đến hạn vào ngày 1 tháng 5. Vào ngày 1 tháng 5, đăng ký Tier 2 có hiệu lực và Maria bị tính phí 20 đô la vào ngày đầu tiên của mỗi tháng.

Danh sách sau mô tả hành vi chuyển tiếp nếu nhà phát triển thay vào đó cho phép một bản dùng thử miễn phí cho mỗi đăng ký:

* IMMEDIATE\_WITH\_TIME\_PRORATION - Maria được nâng cấp lên Tier 2 ngay lập tức. Kể từ khi Maria nâng cấp giữa thời gian đăng ký, một nửa số đăng ký của tháng (15 ngày trị giá 10 đô la / tháng) được áp dụng cho đăng ký mới của cô. Tuy nhiên, vì đăng ký mới đó có giá $ 20 / tháng, số dư 15 ngày chỉ trả trong 7,5 ngày. Maria đủ điều kiện để dùng thử miễn phí khác lên Cấp 2, vì vậy cô không bị tính phí thêm 37,5 ngày nữa. Bắt đầu từ ngày 22 tháng 20, cô bị tính phí 20 đô la mỗi tháng.
* IMMEDIATE\_AND\_CHARGE\_PRORATED\_PRICE - Chế độ này có thể được sử dụng vì giá đăng ký Cấp 2 trên mỗi đơn vị thời gian ($ 20 / tháng) lớn hơn giá đăng ký Cấp 1 cho mỗi đơn vị thời gian ($ 10 / tháng). Đăng ký Tier 1 của Maria ngay lập tức được nâng cấp lên Tier 2 và cô mất bản dùng thử miễn phí. Vì ngày thanh toán tiếp theo của Maria là ngày 1 tháng 5, cô bị tính phí 10 đô la hôm nay để trang trải nửa thứ 2 của tháng 4, sau đó bắt đầu từ ngày 1 tháng 20, cô bị tính phí 20 đô la mỗi tháng.
* IMMEDIATE\_WITHOUT\_PRORATION - Đăng ký Cấp 1 của Maria ngay lập tức được nâng cấp lên Tier 2. Maria giữ bản dùng thử miễn phí cho đến ngày 30 tháng 4 và hiện có quyền truy cập vào Cấp 2.
* HOÃN LẠI - Đăng ký Cấp 1 của Maria tiếp tục cho đến khi khoản thanh toán tiếp theo đến hạn vào ngày 1 tháng 5. Vào ngày 1 tháng 5, đăng ký Tier 2 có hiệu lực và Maria bị tính phí 20 đô la vào ngày đầu tiên của mỗi tháng.

### Khuyến nghị proration

Bảng sau đây hiển thị các tình huống chứng thực diferrent cùng với những gì chúng tôi đề xuất cho mỗi tình huống:

|  |  |  |
| --- | --- | --- |
| Kịch bản | Chế độ proration được đề xuất | Kết quả |
| Nâng cấp lên hạng đắt hơn | IMMEDIATE\_AND\_CHARGE\_PRORATED\_PRICE | Người dùng nhận được quyền truy cập ngay lập tức trong khi vẫn giữ cùng thời hạn thanh toán. |
| Hạ cấp xuống một tầng ít tốn kém hơn | Hoãn | Người dùng đã thanh toán cho bậc đắt hơn, vì vậy họ sẽ giữ quyền truy cập cho đến ngày thanh toán tiếp theo. |
| Thay đổi thời gian định kỳ trên cùng một bậc (ví dụ: hàng tháng sang hàng năm) | Hoãn | Người dùng sẽ thanh toán giá định kỳ mới vào ngày thanh toán tiếp theo. |
| Nâng cấp trong khi dùng thử miễn phí, giữ bản dùng thử | IMMEDIATE\_WITHOUT\_PRORATION | Người dùng giữ quyền truy cập dùng thử miễn phí, nhưng nâng cấp lên cấp cao hơn cho phần còn lại của bản dùng thử. |
| Nâng cấp trong khi dùng thử miễn phí - kết thúc quyền truy cập vào bản dùng thử miễn phí | IMMEDIATE\_AND\_CHARGE\_PRORATED\_PRICE | Người dùng nhận được quyền truy cập vào tầng mới ngay lập tức nhưng không còn bản dùng thử miễn phí. |

## Winback sau khi khuấy

Sau khi người dùng đã hủy gói đăng ký của họ, bạn có thể thử giành lại chúng trong ứng dụng của mình hoặc thông qua cửa hàng Play. Bảng sau mô tả các tình huống đăng ký khác nhau cùng với các hành động winback liên quan và yêu cầu ứng dụng.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Trước khi hết hạn đăng ký | | Sau khi hết hạn đăng ký | |
|  | Trong ứng dụng | Trong Cửa hàng Play | Trong ứng dụng | Trong Cửa hàng Play |
| Tính năng winback | Đăng ký trong ứng dụng | Khôi phục | Đăng ký trong ứng dụng | Đăng ký lại |
| Người dùng trải qua quy trình thanh toán | Có | Không | Có | Có |
| Đăng ký người dùng vẫn được liên kết với cùng một SKU | Người dùng có thể đăng ký SKU giống nhau hoặc khác nhau | Có | Người dùng có thể đăng ký SKU giống nhau hoặc khác nhau | Có |
| Tạo mã thông báo mua hàng mới | Có | Không | Có | Có |
| Được bật theo mặc định | Không | Có, hỗ trợ cần thiết cho tất cả các nhà phát triển | Không | Ứng dụng không có Thư viện thanh toán 2.0 trở lên: Không  Ứng dụng có Thư viện thanh toán 2.0 trở lên: Có. Nhà phát triển có thể chọn không tham gia trong Console. |
| Khi người dùng bị tính phí | Nếu sử dụng cùng một SKU: kết thúc kỳ thanh toán hiện tại.  Nếu sử dụng SKU khác nhau: phụ thuộc vào chế độ proration. | Kết thúc kỳ thanh toán hiện tại | Ngay lập tức | Ngay lập tức |
| Yêu cầu thực hiện | Cung cấp giao diện người dùng đăng ký lại trong ứng dụng của bạn | Phát hiện thay đổi trong trạng thái đăng ký  Liên kết sâu đến Cửa hàng Play | Cung cấp giao diện người dùng đăng ký lại trong ứng dụng của bạn | Xử lý các giao dịch mua ngoài ứng dụng |

### Trước khi hết hạn đăng ký - trong ứng dụng

Đối với các gói đăng ký đã bị hủy nhưng chưa hết hạn, bạn có thể cho phép người đăng ký khôi phục gói đăng ký của họ trong ứng dụng của mình bằng cách áp dụng quy trình mua sản phẩm trong ứng dụng giống như đối với người đăng ký mới. Đảm bảo giao diện người dùng của bạn phản ánh rằng người dùng có đăng ký hiện có. Ví dụ: bạn có thể muốn hiển thị ngày hết hạn hiện tại của người dùng và giá định kỳ bằng nút Kích hoạt lại.

Hầu hết thời gian, bạn sẽ muốn cung cấp cho người dùng cùng một mức giá và SKU mà họ đã đăng ký, như sau:

* Bắt đầu mua đăng ký mới với cùng một SKU.
* Đăng ký mới thay thế đăng ký cũ và gia hạn vào cùng ngày hết hạn. Đăng ký cũ ngay lập tức được đánh dấu là đã hết hạn.
* Ví dụ: Achilles có đăng ký Ứng dụng Example Music và đăng ký sẽ hết hạn vào ngày 1 tháng 8. Vào ngày 10 tháng 7, anh đăng ký lại đăng ký một tháng với cùng một mức giá mỗi tháng. Đăng ký mới được tính theo tỷ giá với tín dụng còn lại, ngay lập tức hoạt động và vẫn gia hạn vào ngày 1 tháng 8.

Nếu muốn cung cấp một mức giá khác — ví dụ: bản dùng thử miễn phí mới hoặc giảm giá winback — thay vào đó, bạn có thể cung cấp một SKU khác cho người dùng:

* Bắt đầu nâng [cấp hoặc hạ cấp](https://developer.android.com/google/play/billing/subscriptions#change) với SKU khác nhau bằng cách sử dụng chế độ proration IMMEDIATE\_WITHOUT\_PRORATION.
* Đăng ký mới thay thế đăng ký cũ và gia hạn vào cùng ngày hết hạn. Người dùng bị tính giá của SKU mới, bao gồm mọi giá giới thiệu, vào ngày hết hạn ban đầu.
* Ví dụ: Achilles có đăng ký Ứng dụng Example Music và đăng ký sẽ hết hạn vào ngày 1 tháng 8. Vào ngày 10 tháng 7, anh đăng ký lại đăng ký hàng năm với giá giới thiệu. Đăng ký mới ngay lập tức hoạt động và người dùng bị tính giá giới thiệu vào ngày 1 tháng 8.
* Nếu bạn quyết định đưa giá dùng thử hoặc giới thiệu miễn phí vào SKU winback, hãy đảm bảo rằng người dùng đủ điều kiện bằng cách bỏ chọn hộp Cho phép dùng thử miễn phí cho mỗi ứng dụng trong Google Play Console, hạn chế người dùng nhận được một bản dùng thử miễn phí cho mỗi ứng dụng.

Khi bạn nhận được mã thông báo mua [hàng, hãy xử lý giao](https://developer.android.com/google/play/billing/integrate#process) dịch mua giống như khi bạn đăng ký mới. Ngoài ra, API nhà phát triển Google Play trả về api LinkedPurchaseToken trong tài nguyên đăng ký. Hãy chắc chắn làm [mất hiệu lực mã thông báo](https://developer.android.com/google/play/billing/subs#upgrade-downgrade) được cung cấp trong linkedPurchaseToken để đảm bảo rằng mã thông báo cũ không được sử dụng để có quyền truy cập vào các dịch vụ của bạn.

### Trước khi hết hạn đăng ký - trong Cửa hàng Play

Lưu ý: Hỗ trợ khôi phục là bắt buộc đối với tất cả các nhà phát triển.

Mặc dù gói đăng ký bị hủy nhưng vẫn hoạt động, người dùng có thể khôi phục đăng ký trong trung tâm đăng ký Google Play bằng cách nhấp vào Đăng ký lại (trước đó là Khôi phục). Điều này giữ nguyên đăng ký và mã thông báo mua.

phần đăng ký trong ứng dụng cửa hàng google play hiển thị
            đăng ký bị hủy bằng nút đăng ký lại

Biểu đồ 8. Tài khoản > đăng ký trong ứng dụng Cửa hàng Google Play hiển thị gói đăng ký bị hủy bằng nút Đăng ký lại.

Để biết thêm thông tin về cách khôi phục đăng ký, [hãy xem Khôi phục](https://developer.android.com/google/play/billing/subscriptions#restore).

### Sau khi hết hạn đăng ký - trong ứng dụng

Bạn có thể cho phép người đăng ký đã hết hạn đăng ký lại trong ứng dụng của mình bằng cách áp dụng quy trình mua sản phẩm trong ứng dụng giống như đối với người đăng ký mới. Lưu ý những điều sau:

* Để cung cấp cho người dùng ưu đãi giảm giá, bạn có thể muốn cung cấp ID sản phẩm với giá đặc biệt cho gói đăng ký của mình, còn được gọi là SKU winback. Bạn có thể cung cấp ưu đãi trong ứng dụng của mình hoặc bạn có thể thông báo cho người dùng về ưu đãi bên ngoài ứng dụng, chẳng hạn như trong email.
* Để bắt đầu đăng ký winback, hãy khởi chạy quy trình mua hàng trong ứng dụng Android bằng Thư viện thanh toán của Google Play. Đây là quá trình tương tự như với đăng ký mới, nhưng bạn có thể xác định SKU có sẵn cho người dùng.
* Nếu bạn quyết định đưa giá dùng thử hoặc giới thiệu miễn phí vào SKU winback, hãy đảm bảo rằng người dùng đủ điều kiện bằng cách bỏ chọn hộp Cho phép dùng thử miễn phí cho mỗi ứng dụng trong Google Play Console, hạn chế người dùng nhận được một bản dùng thử miễn phí cho mỗi ứng dụng.
* Nếu người dùng đăng ký lại cùng một SKU, họ không còn đủ điều kiện nhận bản dùng thử miễn phí hoặc giá giới thiệu. Đảm bảo rằng giao diện người dùng của bạn phản ánh điều này.

Khi bạn nhận được mã thông báo mua [hàng, hãy xử lý giao](https://developer.android.com/google/play/billing/integrate#process) dịch mua giống như khi bạn đăng ký mới. Bạn sẽ không nhận được linkedPurchaseToken trong tài nguyên đăng ký.

### Sau khi hết hạn đăng ký - trong Cửa hàng Play

Lưu ý: Bạn chỉ có thể đăng ký lại từ Cửa hàng Play cho các ứng dụng có phiên bản Thư viện thanh toán phiên bản 2.0 trở lên. Lưu ý: Tất cả SKU đăng ký cho các ứng dụng đủ điều kiện đã đăng ký lại theo mặc định. Bạn có thể chọn không tham gia bất cứ lúc nào bằng cách điều chỉnh cài đặt SKU thông [qua Google Play Console](https://support.google.com/googleplay/android-developer/answer/140504).

Nếu được bật, người dùng có thể đăng ký lại cùng một SKU trong tối đa một năm sau khi hết hạn bằng cách nhấp vào Đăng ký lại trong trung tâm đăng ký Google Play. Điều này tạo ra một đăng ký mới và mua mã thông báo.

phần đăng ký trong ứng dụng cửa hàng google play hiển thị
            đăng ký đã hủy và hết hạn với đăng ký lại và xóa
            Nút

Biểu đồ 9. Tài khoản > Đăng ký trong ứng dụng Cửa hàng Google Play hiển thị gói đăng ký đã hủy và hết hạn bằng nút Đăng ký lại và Xóa.

Gửi lại được coi là giao dịch mua ngoài ứng dụng, vì vậy hãy đảm bảo tuân thủ các phương pháp hay nhất để xử lý các [giao dịch mua được thực hiện từ bên ngoài ứng dụng của bạn](https://developer.android.com/google/play/billing/integrate#ooap).

## Quản lý khách hàng

Sử dụng thông báo dành cho nhà phát triển theo thời gian thực, bạn có thể phát hiện trong thời gian thực khi người dùng quyết định hủy. Khi người dùng hủy, nhưng trước khi gói đăng ký của họ hết hạn, bạn có thể gửi cho họ thông báo đẩy hoặc tin nhắn trong ứng dụng để yêu cầu họ đăng ký lại. Để biết thêm thông tin, hãy [xem Winback sau khi khuấy](https://developer.android.com/google/play/billing/subscriptions#winback).

## Dùng thử miễn phí và giá giới thiệu

Ứng dụng của bạn có thể cung cấp bản dùng thử miễn phí và giá giới thiệu để khuyến khích khách hàng dùng thử ứng dụng của bạn.

### Dùng thử miễn phí

Bạn có thể cho phép người dùng dùng thử đăng ký trước khi thanh toán. Bản dùng thử miễn phí chạy trong một khoảng thời gian mà bạn chỉ định và tự động chuyển đổi thành đăng ký đầy đủ sau khi thời gian dùng thử miễn phí kết thúc. Các đăng ký được chuyển đổi này sử dụng cùng khoảng thời gian và giá như các đăng ký thông thường. Bạn có thể thiết lập bản dùng thử miễn phí cho bất kỳ loại đăng ký nào.

Để bắt đầu dùng thử miễn phí, người dùng hoàn tất quy trình tiêu chuẩn để mua gói đăng ký trên Google Play. Nếu người dùng đủ điều kiện dùng thử miễn phí, họ sẽ không bị tính phí trong suốt thời gian dùng thử miễn phí và họ được thông báo qua email rằng đăng ký bao gồm thời gian dùng thử miễn phí. Google Play ghi lại giao dịch là 0,00 USD và gói đăng ký được đánh dấu là đã mua trong thời gian dùng thử miễn phí hoặc cho đến khi bị hủy.

Google Play xác minh rằng người dùng có phương thức thanh toán hợp lệ trước khi bắt đầu dùng thử miễn phí. Một số người dùng có thể thấy xác minh này là tạm dừng hoặc tính phí trên phương thức thanh toán của họ. Việc giữ hoặc tính phí này là tạm thời và sau đó được đảo ngược hoặc hoàn lại tiền.

Sau khi thời gian dùng thử kết thúc, phương thức thanh toán của người dùng sẽ được tính cho toàn bộ số tiền đăng ký.

Nếu người dùng hủy đăng ký bất cứ lúc nào trong thời gian dùng thử miễn phí, đăng ký vẫn hoạt động cho đến khi kết thúc bản dùng thử và họ không bị tính phí khi thời gian dùng thử miễn phí kết thúc.

Theo mặc định, người dùng chỉ có thể nhận được một bản dùng thử miễn phí trên tất cả các đăng ký có sẵn trong ứng dụng của bạn. Nếu muốn cung cấp một bản dùng thử miễn phí cho mỗi sản phẩm đăng ký, bạn có thể thay đổi tùy chọn cài đặt tương ứng trong Google Play Console.

Để thêm bản dùng thử miễn phí vào gói đăng ký, hãy xem hướng [dẫn dùng thử miễn phí](https://support.google.com/googleplay/android-developer/answer/140504) trong Trung tâm trợ giúp của Google Play.

[Các sự kiện vòng đời đăng](https://developer.android.com/google/play/billing/subscriptions#lifecycle) ký cho đăng ký với bản dùng thử miễn phí hoạt động giống như đăng ký mà không cần dùng thử miễn phí. Sự khác biệt duy nhất là thời gian gia hạn có thể khác với khoảng thời gian gia hạn đăng ký thông thường.

Khi [hiển thị cho người dùng của bạn các sản phẩm có sẵn để](https://developer.android.com/google/play/billing/integrate#show-products)bán, bạn có thể sử dụng phương pháp [SkuDetails.getFreeTrialPeriod() để hiển](https://developer.android.com/reference/com/android/billingclient/api/SkuDetails#getfreetrialperiod) thị rằng bản dùng thử miễn phí có sẵn cho một sản phẩm.

Lưu ý: SkuDetails.getFreeTrialPeriod() trả về thời gian dùng thử miễn phí được cấu hình trong Google Play Console. Có thể người dùng đã sử dụng bản dùng thử miễn phí. Trong trường hợp này, Google Play thông báo cho người dùng rằng họ không đủ điều kiện dùng thử miễn phí khác.

Nếu đăng ký được mua với bản dùng thử miễn phí, đăng ký [được trả](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) về từ API nhà phát triển của Google Play có paymentState = 2 (dùng thử miễn phí). Nếu đăng ký gia hạn thành công, paymentState chuyển sang 1 (thanh toán nhận được).

### Giá giới thiệu

Tương tự như bản dùng thử miễn phí, ứng dụng của bạn có thể cung cấp giá giới thiệu để khuyến khích người dùng dùng thử ứng dụng của bạn. Để tìm hiểu thêm về giá giới thiệu và cách bật tính năng này, hãy xem [Tạo đăng ký](https://support.google.com/googleplay/android-developer/answer/140504).

[Các sự kiện vòng](https://developer.android.com/google/play/billing/subscriptions#lifecycle) đời đăng ký cho đăng ký với giá giới thiệu hoạt động giống như đăng ký mà không cần giá giới thiệu. Sự khác biệt duy nhất là thời gian gia hạn có thể khác với khoảng thời gian gia hạn đăng ký thông thường.

Khi [hiển thị cho người dùng của bạn các sản phẩm có sẵn](https://developer.android.com/google/play/billing/integrate#show-products)đểbán, bạn có thể sử dụng [SkuDetails.getIntroductoryPrice() và các phương](https://developer.android.com/reference/com/android/billingclient/api/SkuDetails#getintroductoryprice) pháp liên quan để hiển thị cho người dùng của bạn rằng giá giới thiệu có sẵn cho đăng ký.

Lưu ý: Các phương pháp định giá giới thiệu trong SkuDetails trả về giá như được định cấu hình trong Google Play Console. Có thể người dùng đã sử dụng ưu đãi giá giới thiệu cho SKU đó. Trong trường hợp này, Google Play thông báo cho người dùng rằng họ không đủ điều kiện nhận lại giá giới thiệu.

## Quảng bá gói đăng ký của bạn

Bạn có thể tạo mã khuyến mãi để cung cấp cho người dùng được chọn bản dùng thử miễn phí mở rộng cho SKU đăng ký hiện có. Để tìm hiểu thêm, hãy xem [Mã khuyến mãi](https://developer.android.com/google/play/billing/promo).

## Hủy, hoàn tiền hoặc thu hồi

Bạn có thể sử dụng API nhà [phát triển trên Google Play](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) [để hủy](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/cancel), [hoàn](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/refund)tiềnhoặc thu [hồi](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/revoke) gói đăng ký. Chức năng này cũng có sẵn trong Google Play [Console](https://support.google.com/googleplay/android-developer/answer/2741495).

* Hủy: Người dùng có thể hủy gói đăng ký trên Google Play. Bạn cũng có thể cung cấp tùy chọn để người dùng hủy trong ứng dụng hoặc trên trang web của mình. Ứng dụng của bạn sẽ xử lý các trường hợp hủy này như được mô [tả trong Thu hồi](https://developer.android.com/google/play/billing/subscriptions#revoke).
* Hoàn tiền: Khi bạn hoàn tiền, người dùng có thể tiếp tục sử dụng gói đăng ký. Bạn có thể sử dụng tính năng hoàn tiền nếu có lỗi kỹ thuật khiến người dùng không thể truy cập vào sản phẩm của bạn, nhưng lỗi đã được giải quyết. Xin lưu ý rằng để hoàn tiền nhiều hơn khoản thanh toán gần đây nhất hoặc nếu muốn hoàn tiền một phần, bạn phải sử dụng Google Play Console.
* Thu hồi: Khi bạn thu hồi, người dùng ngay lập tức mất quyền truy cập vào đăng ký. Điều này có thể được sử dụng nếu, ví dụ, có một lỗi kỹ thuật ngăn người dùng truy cập vào sản phẩm của bạn và người dùng không muốn tiếp tục sử dụng sản phẩm. Ứng dụng của bạn sẽ xử lý các trường hợp hủy này như được mô [tả trong Thu hồi](https://developer.android.com/google/play/billing/subscriptions#revoke).

Bảng sau đây minh họa sự khác biệt giữa hủy, hoàn tiền và thu hồi.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Dừng gia hạn | Hoàn tiền | Thu hồi quyền truy cập |
| [Hủy bỏ](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/cancel) | Có | Không | Không |
| [Hoàn](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/refund) | Không | Có | Không |
| [Thu hồi](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/revoke) | Có | Có | Có |

## Trì hoãn thanh toán cho người đăng ký

Bạn có thể tạm ứng ngày thanh toán tiếp theo cho người đăng ký bằng [cách sử dụng Purchases.subscriptions:defer](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions/defer) từ API nhà phát triển trên Google Play. Trong thời gian trì hoãn, người dùng đã đăng ký nội dung của bạn với toàn quyền truy cập nhưng không bị tính phí. Ngày gia hạn đăng ký được cập nhật để phản ánh ngày mới.

Thanh toán trả chậm cho phép bạn làm như sau:

* Cung cấp cho người dùng quyền truy cập miễn phí như một ưu đãi đặc biệt, chẳng hạn như tặng một tuần miễn phí để mua phim.
* Cung cấp quyền truy cập miễn phí cho khách hàng như một cử chỉ thiện chí.

Thanh toán có thể được hoãn lại ít nhất là một ngày và miễn là một năm cho mỗi lệnh gọi API. Để trì hoãn việc thanh toán hơn nữa, bạn có thể gọi lại API trước khi ngày thanh toán mới đến.

Ví dụ: Darcy có đăng ký hàng tháng cho nội dung trực tuyến cho ứng dụng Fishing Quarterly. Cô thường được lập hóa đơn £ 1.25 vào ngày đầu tiên của mỗi tháng. Vào tháng 3, cô đã tham gia một cuộc khảo sát trực tuyến cho nhà xuất bản ứng dụng. Nhà xuất bản thưởng cho cô sáu tuần miễn phí bằng cách trì hoãn khoản thanh toán tiếp theo cho đến ngày 15 tháng 5, tức là sáu tuần sau ngày thanh toán dự kiến trước đó của cô là ngày 1 tháng 4. Darcy không bị tính phí vào tháng 4 hoặc đầu tháng 5 và vẫn có quyền truy cập vào nội dung. Vào ngày 15 tháng 5, cô bị tính phí đăng ký £ 1.25 bình thường trong tháng. Ngày gia hạn tiếp theo của cô bây giờ là ngày 15 tháng XNUMX.

Khi trì hoãn, bạn có thể muốn thông báo cho người dùng qua email hoặc trong ứng dụng để thông báo cho họ rằng ngày thanh toán của họ đã thay đổi.

## Thay đổi giá đăng ký

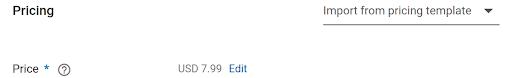
Cảnh báo: Bạn không nên thay đổi giá đăng [ký với gói đăng ký của Google.](https://developers.google.com/news/subscribe)

Google Play cho phép bạn bán các sản phẩm kỹ thuật số của mình trên khắp thế giới cho nhiều người dùng ở các địa phương khác nhau với các loại tiền tệ và cân nhắc về giá cả khác nhau. Đôi khi, do chi phí khu vực hoặc biến động tiền tệ, bạn có thể quyết định cần thay đổi giá đăng ký của mình. Nếu bạn sẵn sàng giữ người đăng ký hiện tại ở mức giá hiện tại, bạn có thể tạo SKU mới với giá cập nhật và cung cấp cho người đăng ký mới.

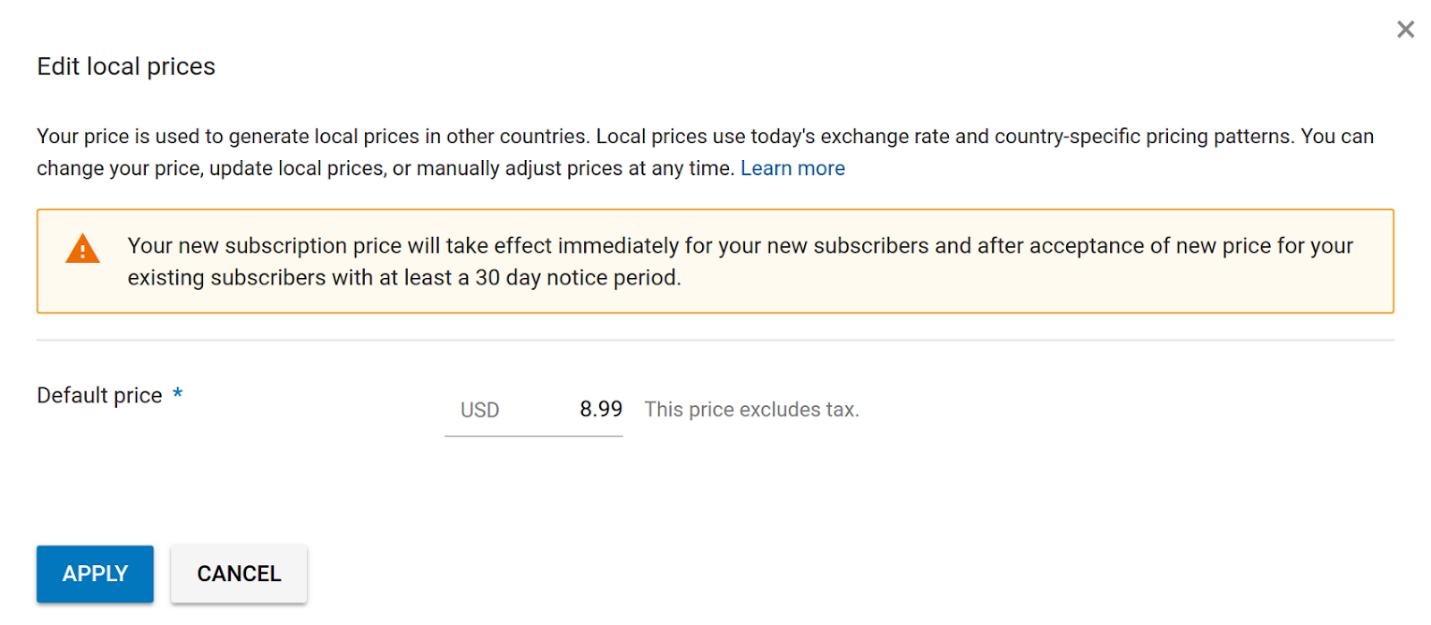
Nếu bạn không thể tiếp tục hỗ trợ người đăng ký với mức giá được cung cấp khi họ đăng ký đăng ký của bạn và sẵn sàng hủy tất cả đăng ký cho người dùng quyết định không chấp nhận giá mới, bạn có thể thực thi thay đổi giá bắt buộc. Nếu người dùng không đồng ý với giá mới, đăng ký của họ sẽ bị hủy. Để tối đa hóa mức giá chọn tham gia và khuyến khích người dùng thực hiện hành động, ứng dụng của bạn sẽ hiển thị thông điệp cho người dùng về thay đổi giá sắp tới. Để biết thêm thông tin về thông điệp thay đổi giá, hãy xem [Truyền đạt thay đổi giá của bạn.](https://developer.android.com/google/play/billing/subscriptions#price-change-communicate)

Để thay đổi giá của đăng ký, hãy làm như sau:

1. Đăng nhập vào [Google Play Console](https://play.google.com/console/).
2. Tìm ứng dụng chứa gói đăng ký có giá bạn muốn thay đổi.
3. Chọn Store presence > sản phẩm Trong ứng dụng, rồi mở tab Subscriptions (Đăng ký).
4. Chọn liên kết Chỉnh sửa bên cạnh giá bạn muốn thay đổi, như trong hình 10:

Biểu đồ 10. Chỉnh sửa trường giá đăng ký.

1. Nhập giá đăng ký mới.
2. Khi bạn thay đổi giá, một hộp thoại xác nhận sẽ xuất hiện, như trong hình 11.

Biểu đồ 11. Cảnh báo xuất hiện khi bạn chỉnh sửa giá đăng ký.

Hộp thoại này giải thích cách thay đổi giá ảnh hưởng đến người đăng ký mới và hiện tại. Lưu ý những điều sau:

* + Bạn không thể đảo ngược thay đổi giá sau khi bạn xác nhận.
  + Google Play bắt đầu thông báo cho người dùng về thay đổi giá bắt đầu từ bảy ngày sau khi xác nhận.
  + Khi tăng giá đăng ký, người dùng phải chấp thuận thay đổi giá trong vòng 30 ngày hoặc đăng ký của họ sẽ tự động bị hủy.
  + Giảm giá đăng ký được áp dụng tự động cho người đăng ký hiện tại vào ngày gia hạn tiếp theo của họ. Bất kỳ ai đăng ký sau khi giảm giá sẽ tự động nhận được giá giảm.

Bấm Áp dụng để tiếp tục.

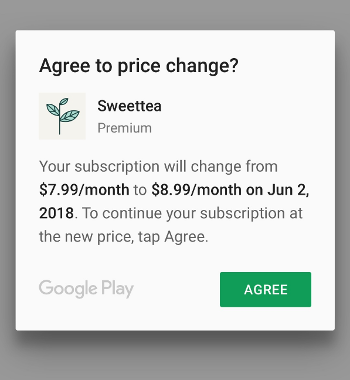
1. Hộp thoại xác nhận xuất hiện một lần nữa mô tả khi giá đăng ký mới có hiệu lực.

confirmation dialog that shows when the subscription price
           change takes effectBiểu đồ 12. Hộp thoại xác nhận hiển thị khi thay đổi giá đăng ký có hiệu lực.

### Truyền đạt thay đổi giá của bạn cho người dùng

Bạn nên thông báo cho người đăng ký hiện tại của mình bất cứ khi nào bạn thực hiện thay đổi giá, đặc biệt nếu giá đang tăng.

Khi tăng giá đăng ký, bạn có ít nhất bảy ngày để thông báo cho người đăng ký hiện tại về thay đổi giá trước khi Google Play có thể bắt đầu thông báo cho họ. Khi Google Play thông báo cho người dùng, nó sẽ làm như vậy bằng cách hiển thị một hộp thoại tương tự như hộp thoại được hiển thị trong hình 13. Hộp thoại này hiển thị giá cũ, giá mới và ngày giá mới có hiệu lực.

Lưu ý: Khi giảm giá đăng ký, người dùng không cần phải đồng ý với thay đổi và không được hiển thị hộp thoại thay đổi giá. Biểu đồ 13. Hộp thoại chung thông báo cho người dùng về thay đổi giá đăng ký.

Trong ứng dụng của mình, bạn có thể hiển thị hộp thoại này theo những cách sau:

* [Khởi chạy quy trình xác nhận thay đổi giá](https://developer.android.com/google/play/billing/subscriptions#price-change-launch) khi người dùng mở ứng dụng của bạn.
* Gửi người dùng đến trung tâm đăng ký để xem lại thay đổi giá bằng liên [kết sâu](https://developer.android.com/google/play/billing/subscriptions#deep-link).

Ứng dụng của bạn có thể xác định xem người dùng có thay đổi giá đang chờ xử lý bằng cách xem trường priceChange trong tài nguyên đăng [ký .](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) Nếu có, trường này cho biết đăng ký có thay đổi giá và trường priceChange.state cho biết liệu thay đổi giá đã được chấp nhận hay vẫn chưa xuất hiện.

### Khởi chạy quy trình xác nhận thay đổi giá

Để hiển thị hộp thoại thay đổi giá của Google Play khi ứng dụng của bạn khởi [chạy, hãy sử dụng BillingClient.launchPriceChangeConfirmationFlow()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#launchpricechangeconfirmationflow).

Trước khi hiển thị hộp thoại thay đổi giá, bạn cũng có thể trình bày thông báo hoặc hộp thoại của riêng mình để giải thích lý do đằng sau sự thay đổi giá. Nếu bạn tạo và hiển thị loại tin nhắn tùy chỉnh này, người dùng của bạn có nhiều khả năng gia hạn đăng ký của họ ở mức giá mới.

Dưới đây là ví dụ về việc khởi chạy quy trình xác nhận thay đổi giá:

### Xử lý xác nhận của người dùng về sự thay đổi giá

Nếu người dùng chấp nhận giá mới của đăng ký của bạn, đăng ký sẽ gia hạn và có thể được xử lý giống như bất kỳ gia hạn nào [khác](https://developer.android.com/google/play/billing/subscriptions#renewals). Ngoài ra, bạn nhận được thông SUBSCRIPTION\_PRICE\_CHANGE\_CONFIRMED nhà phát triển thời gian thực.

### Xử lý khi thay đổi giá bị từ chối

Nếu người dùng chưa chấp nhận thay đổi giá của bạn vào thời điểm đăng ký cũ của họ hết hạn, họ sẽ hủy đăng ký và bạn nhận được thông báo SUBSCRIPTION\_CANCELED bạn. Sự kiện này có thể được xử lý như được mô tả [trong Hủy](https://developer.android.com/google/play/billing/subscriptions#cancel)bỏ .

### Thay đổi giá ngẫu nhiên

Nếu bạn đã vô tình thay đổi giá của gói đăng ký, hãy đảo ngược thay đổi ngay lập tức. Miễn là giá được hoàn nguyên trong vòng bảy ngày, người đăng ký hiện tại sẽ không được thông báo về việc thay đổi giá ngẫu nhiên. Lưu ý rằng người đăng ký mới có thể nhận được giá ngẫu nhiên trong thời gian giữa thay đổi giá đầu tiên và đảo ngược.

### Thực hiện hai thay đổi giá liên tiếp

Bạn nên đảm bảo rằng bạn chỉ thực hiện một thay đổi giá tại một thời điểm. Thay đổi giá không được khuyến khích cho mục đích thử nghiệm.

Nếu bạn thay đổi giá đăng ký hai lần trong khoảng thời gian bảy ngày, người dùng chỉ cần đồng ý với thay đổi giá mới nhất.

Nếu giá thay đổi cách nhau hơn bảy ngày, các thay đổi về giá sẽ không hủy bỏ lẫn nhau. Thay vào đó, người dùng phải đồng ý với thay đổi giá đầu tiên và có một chu kỳ gia hạn ở mức giá đầu tiên. Sau khi chu kỳ gia hạn kết thúc, sau đó bạn có thể bắt đầu tính giá thứ hai. Gia hạn sau đó được tính giá thứ hai mỗi tháng.

# Mã khuyến mãi

Khuyến mãi hoặc mã khuyến mãi cho phép bạn cung cấp sản phẩm hoặc bản dùng thử một lần cho đăng ký miễn phí cho một số lượng người dùng hạn chế. Người dùng nhập mã khuyến mãi, trong ứng dụng của bạn hoặc trong ứng dụng Cửa hàng Google Play và nhận bản dùng thử mặt hàng hoặc gói đăng ký miễn phí.

Trong Play Console, bạn có thể tạo các loại mã khuyến mãi sau:

* Mã sử dụng một lần: Đây là những mã duy nhất được tạo tự động mà người dùng chỉ có thể đổi một lần. Người dùng đổi các mã này trực tiếp từ Cửa hàng Play hoặc từ bên trong ứng dụng của bạn.
* Mã tùy chỉnh: Bạn có thể chỉ định mã tùy chỉnh có thể được đổi nhiều lần đến giới hạn được xác định trước. Mã tùy chỉnh chỉ có sẵn cho đăng ký và chỉ có thể được đổi bởi những người dùng chưa đăng ký trước đó.

Bạn có thể sử dụng mã khuyến mãi theo nhiều cách để tương tác sáng tạo với người dùng, bao gồm:

* Bạn có thể phân phối thẻ có mã khuyến mãi tại một sự kiện và người dùng sẽ nhập mã khuyến mãi của họ để mở khóa một mặt hàng đặc biệt trong trò chơi.
* Bạn có thể cung cấp mã cho nhân viên để họ có thể chia sẻ chúng với bạn bè và gia đình của họ.
* Bạn có thể gửi mã khuyến mãi cho những người mua ứng dụng của bạn trong một khoảng thời gian nhất định.

Đối với đăng ký, hãy lưu ý những điều sau:

* Mã khuyến mãi cung cấp bản dùng thử miễn phí cho đăng ký và không đăng ký miễn phí.
* Chương trình khuyến mãi không mở rộng các bản dùng thử miễn phí khác. Nếu đăng ký bắt đầu bằng bản dùng thử miễn phí mà không cần mã khuyến mãi, chương trình khuyến mãi sẽ ghi đè thời lượng dùng thử miễn phí ban đầu.
* Người dùng không đủ điều kiện nhận bản dùng thử miễn phí không khuyến mãi — ví dụ: người dùng đã mua đăng ký trong quá khứ vẫn đủ điều kiện dùng thử miễn phí thông qua chương trình khuyến mãi.

Người dùng có thể đổi mã khuyến mãi trong Cửa hàng Google Play cho đến ngày kết thúc chương trình khuyến mãi mà bạn chỉ định trong Play Console. Chương trình khuyến mãi có thể kéo dài đến một năm.

Trước khi thực hiện chương trình khuyến mãi, hãy chắc chắn xem xét những điều sau:

* Đối với sản phẩm dùng một lần, bạn có thể tạo tối đa 500 mã khuyến mãi mỗi quý trên tất cả các sản phẩm được quản lý trong ứng dụng. Bạn có thể sử dụng các kết hợp mã khuyến mãi khác nhau bao gồm một hoặc nhiều sản phẩm một lần. Ví dụ bao gồm:
  + 500 mã khuyến mãi cho một sản phẩm một lần.
  + 100 mã khuyến mãi, mỗi mã cho năm sản phẩm một lần khác nhau.
* Đối với chương trình khuyến mãi đăng ký, bạn có thể tạo tối đa 10.000 mã sử dụng một lần mỗi quý cho mỗi sản phẩm đăng ký. Giới hạn này không được tính vào giới hạn mã khuyến mãi cho sản phẩm một lần.
* Khi tạo chương trình khuyến mãi đăng ký bằng mã tùy chỉnh, bạn có thể chọn giới hạn đổi thưởng từ 2.000 đến 99.999.

Lưu ý: Sau khi tạo chương trình khuyến mãi, bạn không thể thay đổi số lượng mã khuyến mãi trong chương trình khuyến mãi đó và không thể chuyển các mã khuyến mãi đó sang loại khác. Lưu ý: Nếu không sử dụng tất cả mã khuyến mãi trong một quý, bạn sẽ mất quyền truy cập vào chúng. Mã không sử dụng không được chuyển sang quý tiếp theo.

## Tạo và quản lý chương trình khuyến mãi

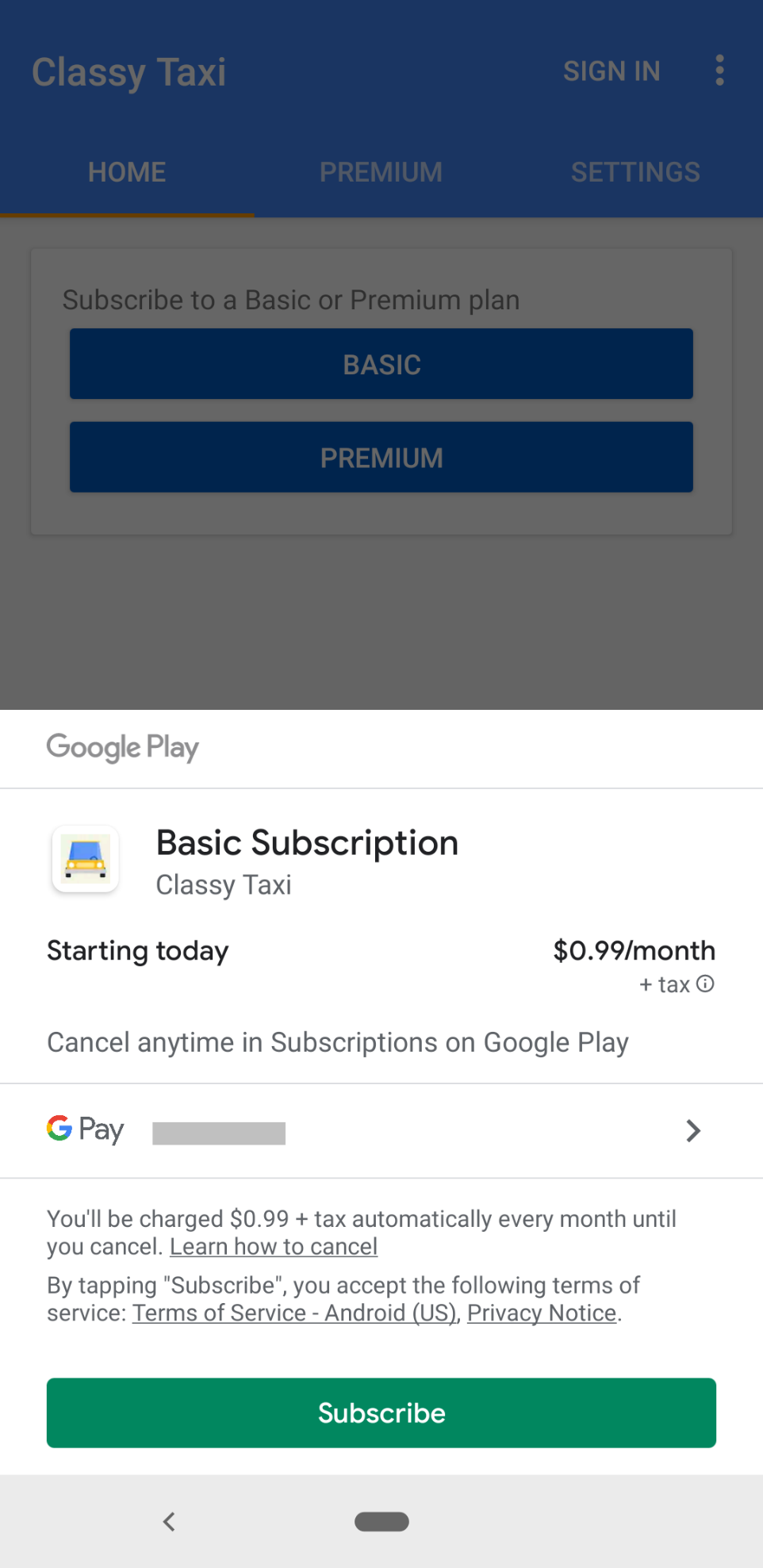
Để tìm hiểu cách thiết lập và quản lý chương trình khuyến mãi, hãy xem [Tạo chương trình khuyến mãi.](https://support.google.com/googleplay/android-developer/answer/6321495)

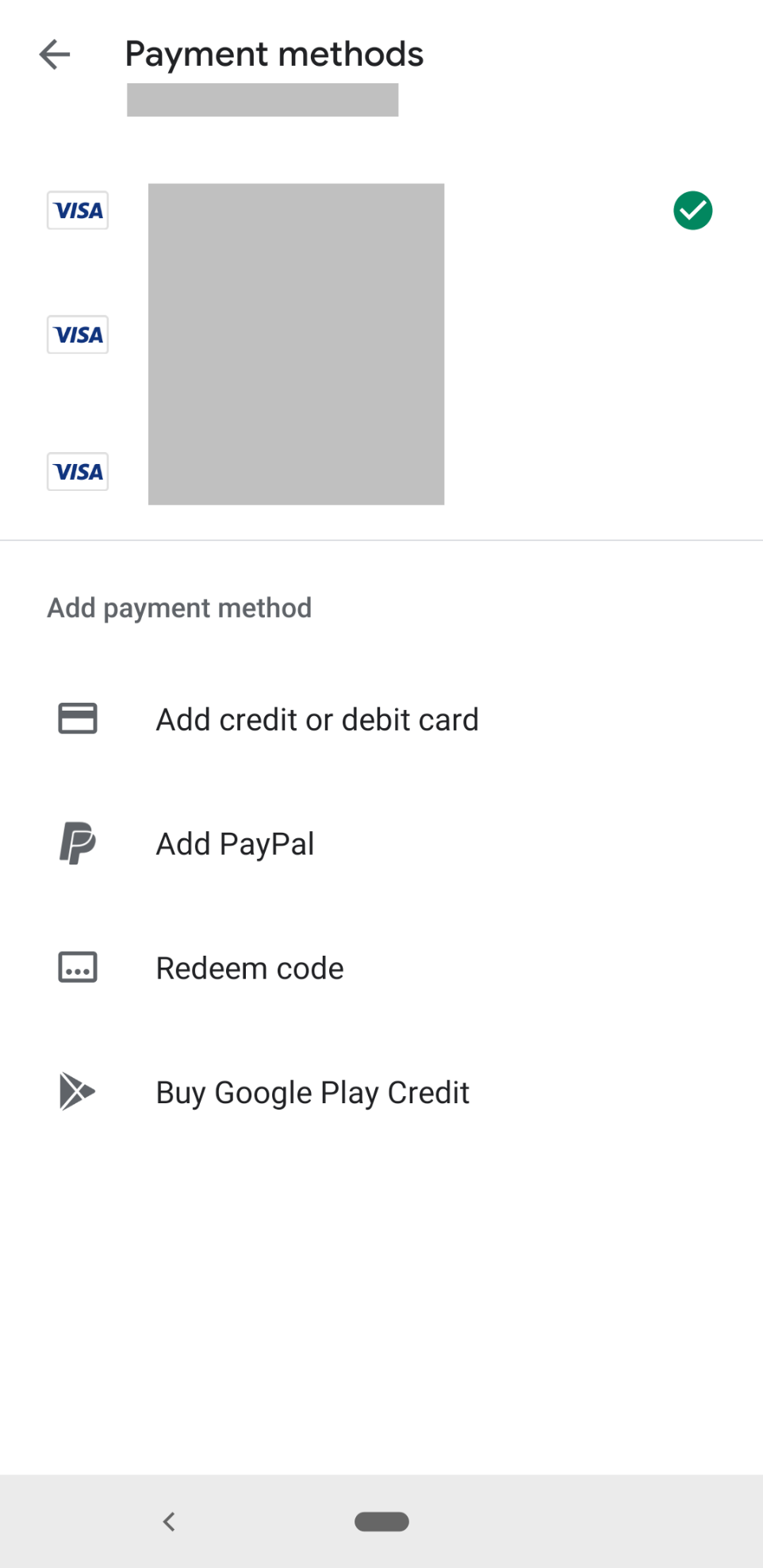
## Luồng đổi thưởng của người dùng

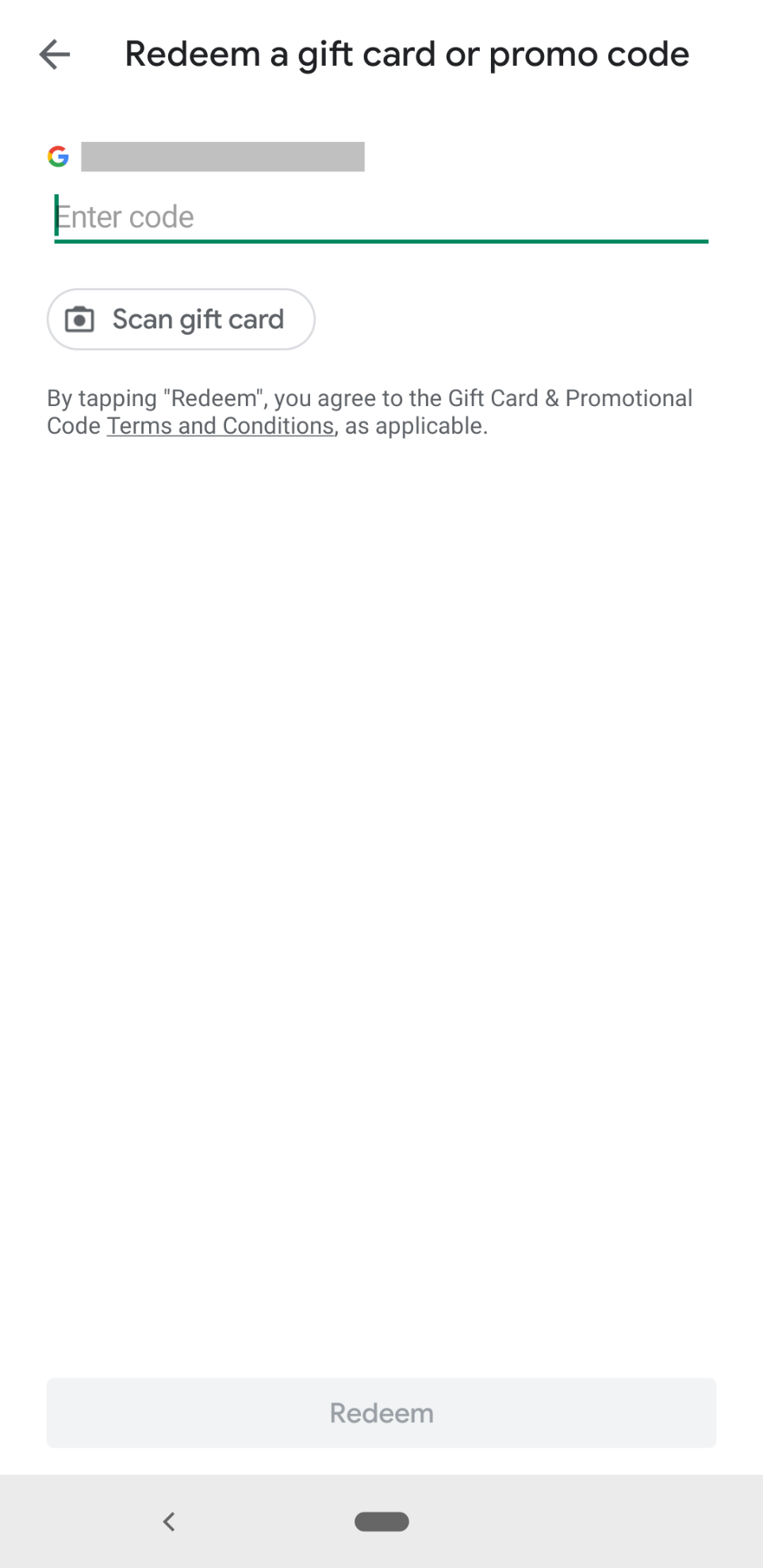
Khi người dùng có mã khuyến mãi, họ có thể đổi mã đó theo một trong các cách sau:

* Trong ứng dụng của bạn: Người dùng có thể bắt đầu mua hàng bằng cách nhấp vào mũi tên xuống bên cạnh hình thức thanh toán trong màn hình mua hàng trên Google Play rồi nhấp vào liên kết Đổi để nhập mã.
* Trong cửa hàng Play: Người dùng có thể nhập mã theo cách thủ công vào Cửa hàng Google Play bằng cách nhấp vào menu điều hướng bên trái của Cửa hàng Play và nhấn vào Đổi mã.
  + Người dùng cũng có thể bỏ qua mục nhập mã thủ công bằng cách theo [liên kết](https://developer.android.com/google/play/billing/promo#deep-link) sâu đến Cửa hàng Google Play.

Ví dụ: hình 1 hiển thị màn hình mua hàng cho gói đăng ký. Để nhập mã khuyến mãi, hãy chạm vào mũi tên bên cạnh phương thức thanh toán hiện tại để hiển thị màn hình Phương thức thanh toán, như trong hình 2. Tiếp theo, chạm vào Đổi mã để chuyển đến màn hình Đổi thẻ quà tặng hoặc mã khuyến mãi, như trong hình 3. Sau đó, bạn có thể nhập mã khuyến mãi trên màn hình này và nhấn vào Đổi để hoàn tất.



Biểu đồ 1. Hộp thoại mua đăng ký.

Biểu đồ 2. Màn hình liệt kê các phương thức thanh toán để mua hàng trong ứng dụng.

Biểu đồ 3. Màn hình mã khuyến mãi.

Đối với mã khuyến mãi đăng ký, hãy lưu ý những điều sau:

* Mã tùy chỉnh chỉ có thể được đổi từ bên trong ứng dụng của bạn, trong khi mã một lần có thể được đổi thông qua cả ứng dụng và cửa hàng Play.
* Sau khi người dùng đổi mã, họ vẫn cần mua đăng ký với mã được áp dụng. Một hình thức thanh toán hợp lệ là bắt buộc đối với đăng ký và đăng ký sẽ tự động gia hạn vào cuối thời gian dùng thử miễn phí mã khuyến mãi.
* Nếu sử dụng Thư viện thanh toán phiên bản 2.0 trở lên, người dùng đổi mã một lần từ cửa hàng Play sẽ ngay lập tức được yêu cầu mua đăng ký thông qua cửa hàng Play. Đây là giao dịch mua ngoài ứng dụng, vì vậy hãy chắc chắn rằng ứng dụng của bạn có thể xử lý các [giao dịch mua này một cách duyên dáng](https://developer.android.com/google/play/billing/integrate#ooap).
  + Nếu ứng dụng của bạn chưa hỗ trợ Thư viện thanh toán 2.0, người dùng phải tải xuống ứng dụng của bạn, điều hướng đến đăng ký chính xác, sau đó mua đăng ký từ bên trong ứng dụng của bạn để sử dụng mã khuyến mãi.

## Thực hiện mã khuyến mãi

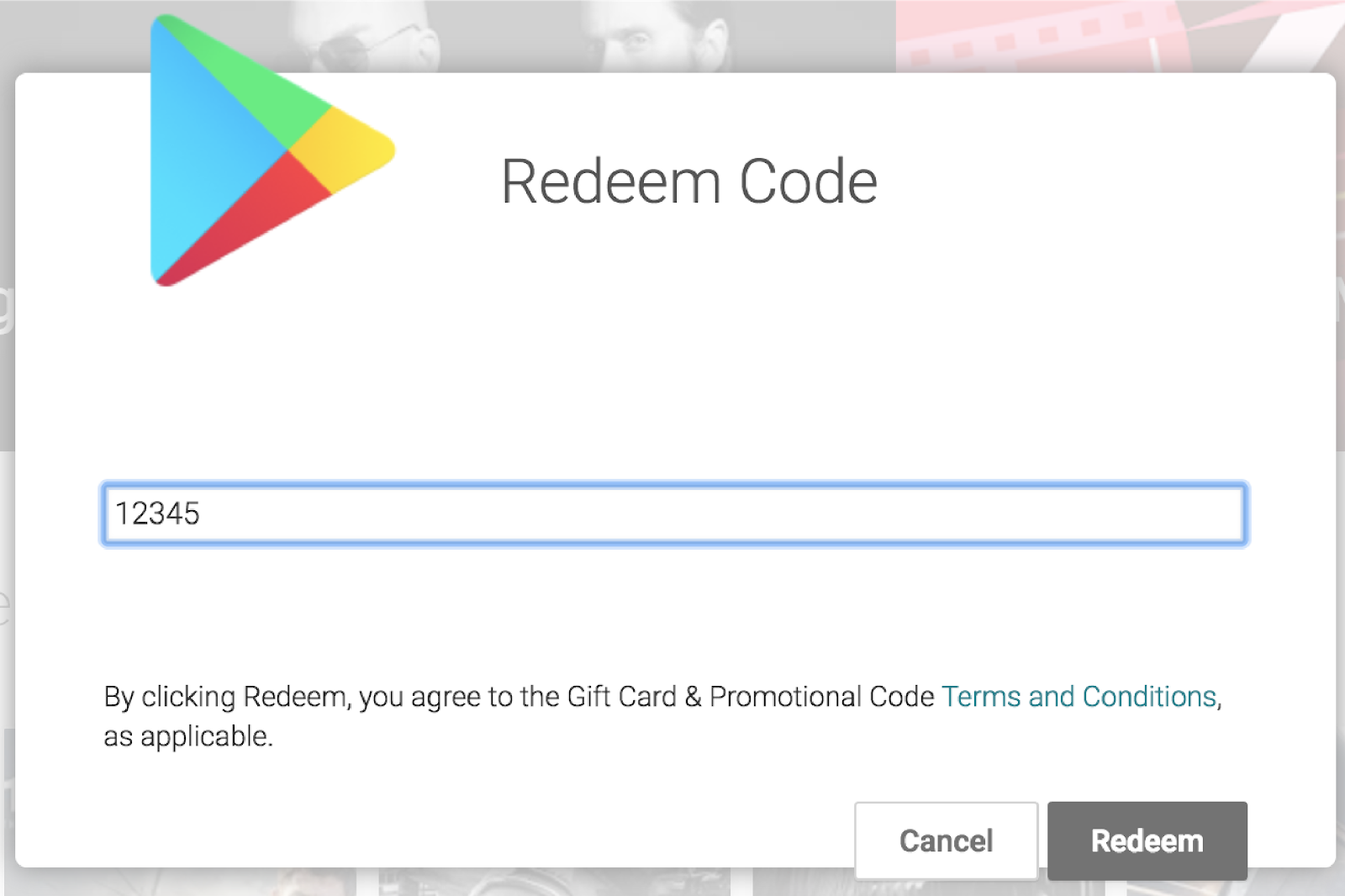
Để đảm bảo ứng dụng của bạn đã sẵn sàng xử lý mã khuyến mãi, ứng dụng của bạn cần xử lý đúng cách các giao dịch đổi lại xảy ra bên ngoài ứng dụng của bạn. Để tìm hiểu thêm, hãy xem [Xử lý giao dịch mua](https://developer.android.com/google/play/billing/integrate#process), Tìm nạp [mục mua](https://developer.android.com/google/play/billing/integrate#fetch)và Xửlý [giao dịch mua được thực hiện bên ngoài ứng dụng của](https://developer.android.com/google/play/billing/integrate#ooap) bạn trong [Tích hợp Thư viện thanh toán Google Play vào ứng dụng của bạn](https://developer.android.com/google/play/billing/integrate).

## Liên kết sâu

Bạn cũng có thể chia sẻ mã khuyến mãi bằng cách tạo URL gửi người dùng đến Cửa hàng Google Play và tự động điền vào trường Nhập mã. Sử dụng định dạng sau cho URL mã khuyến mãi:

https://play.google.com/redeem?code=promo\_code

Hình 4 hiển thị hộp thoại Đổi Mã của ứng dụng Google Play:

Biểu đồ 4. Hộp thoại Đổi Mã của ứng dụng Google Play.

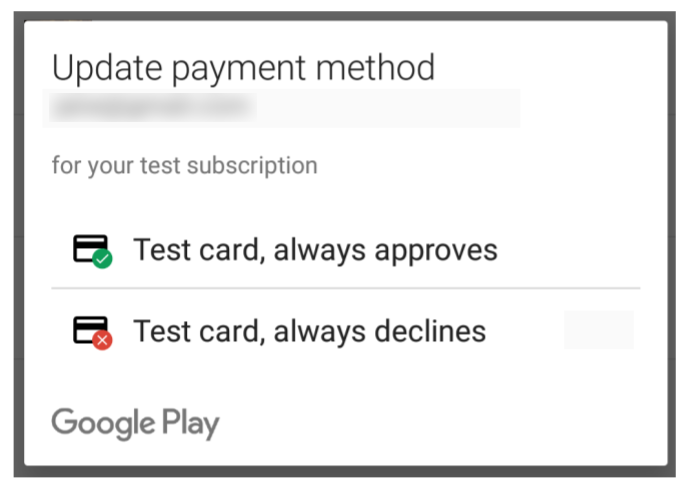
Sau khi người dùng nhấn Đổi, nếu phiên bản mới nhất của ứng dụng được cài đặt, Cửa hàng Google Play sẽ nhắc người dùng mở ứng dụng. Nếu không, Cửa hàng Google Play sẽ nhắc người dùng cập nhật hoặc tải xuống ứng dụng của bạn.

# Test your Google Play Billing Library integration

Bạn nên thử nghiệm tích hợp của mình trong suốt quá trình phát triển. Để kiểm tra trong giai đoạn phát triển, chúng tôi khuyên bạn nên tận dụng các trình kiểm tra giấy phép để chạy qua các kịch bản được mô tả trong chủ đề này. Để định cấu hình người kiểm tra giấy phép, hãy [xem Kiểm tra thanh toán trong ứng dụng bằng cấp phép ứng dụng](https://support.google.com/googleplay/android-developer/answer/6062777).

Sử dụng người kiểm tra giấy phép cung cấp các lợi ích sau:

* Thông thường, Thư viện thanh toán của Google Play bị chặn đối với các ứng dụng không được ký và tải lên Google Play. Người kiểm tra giấy phép có thể bỏ qua kiểm tra này, có nghĩa là bạn có thể tải ứng dụng bên cạnh để thử nghiệm, ngay cả đối với các ứng dụng sử dụng bản dựng gỡ lỗi có chữ ký gỡ lỗi mà không cần tải lên phiên bản ứng dụng mới. Xin lưu ý rằng tên gói phải khớp với tên gói của ứng dụng được định cấu hình cho Google Play và Tài khoản Google phải là người kiểm tra giấy phép cho tài khoản Google Play Console.
* Người kiểm tra giấy phép có quyền truy cập vào các phương thức thanh toán thử nghiệm tránh tính phí cho người thử nghiệm tiền thật để mua hàng. Bạn cũng có thể sử dụng phương thức thanh toán thử nghiệm để mô phỏng một số tình huống nhất định, chẳng hạn như khi thanh toán bị từ chối. Hình 1 cho thấy các hình thức thanh toán thử nghiệm này khi chúng xuất hiện trong luồng mua hàng.
* Người kiểm tra giấy phép có thể [nhanh chóng kiểm tra các tính năng đăng ký](https://developer.android.com/google/play/billing/test#subs).

Biểu đồ 1. Người kiểm tra giấy phép có quyền truy cập vào các phương thức thanh toán thử nghiệm.

Dưới đây là một số chi tiết bổ sung về quá trình mua thử:

* Mua thử nghiệm sử dụng cùng một luồng mua ứng dụng được sử dụng bởi các giao dịch mua thực tế.
* Thuế không được tính cho các giao dịch mua thử nghiệm.
* Google Play cho biết giao dịch mua thử bằng cách hiển thị thông báo ở giữa hộp thoại mua hàng.

Bạn có thể xác nhận tài khoản đang mua hàng bằng cách mở rộng hộp thoại mua hàng. Lưu ý những điều sau:

* Tài khoản thử nghiệm phải nằm trên thiết bị Android của người thử nghiệm.
* Nếu thiết bị có nhiều tài khoản, giao dịch mua được thực hiện bằng tài khoản đã tải ứng dụng xuống.
* Nếu không có tài khoản nào đã tải xuống ứng dụng, giao dịch mua được thực hiện bằng tài khoản đầu tiên.

Trước khi phân phối ứng dụng, bạn có thể sử dụng các bản thử nghiệm của Google Play [để thực](https://support.google.com/googleplay/android-developer/answer/3131213) hiện xác thực bổ sung. Ví dụ: bạn có thể tận dụng các bản thử nghiệm để nhóm QA của bạn đủ điều kiện phát hành mới.

Với các bản thử nghiệm, người dùng có thể cài đặt ứng dụng của bạn từ Google Play và thử nghiệm phiên bản ứng dụng chưa được công khai. Người dùng có thể mua hàng thực bằng bất kỳ phương thức thanh toán nào của họ trong Google Play.

Lưu ý: Người dùng mua hàng trong các bản thử nghiệm dẫn đến phí thực tế cho tài khoản người dùng trừ khi người dùng cũng là người kiểm tra giấy phép.

Để kiểm tra tích hợp Thư viện thanh toán Google Play bằng các bản thử nghiệm, hãy làm như sau:

1. Phát hành ứng dụng của bạn lên [rãnh thử nghiệm](https://support.google.com/googleplay/android-developer/answer/3131213). Xin lưu ý rằng sau khi bạn phát hành ứng dụng lên rãnh thử nghiệm, có thể mất vài giờ để ứng dụng có sẵn cho người thử nghiệm.
2. Đảm bảo mỗi người [thử nghiệm chọn tham gia thử nghiệm ứng dụng của bạn](https://support.google.com/googleplay/android-developer/answer/3131213). Trên URL chọn tham gia thử nghiệm, người thử nghiệm sẽ thấy lời giải thích về ý nghĩa của việc trở thành người thử nghiệm cùng với liên kết đến việc chọn tham gia.

Bạn có thể kiểm tra tích hợp của mình trên bất kỳ thiết bị phần cứng chạy Android 1.6 trở lên nào. Phiên bản mới nhất của ứng dụng Google Play phải được cài đặt trên thiết bị. Để biết thông tin chung về cách thiết lập thiết bị để sử dụng trong việc phát triển các ứng dụng Android, hãy xem Sử [dụng thiết bị phần cứng](https://developer.android.com/tools/device).

Lưu ý: Mặc dù người kiểm tra giấy phép được khuyến nghị phát triển và thử nghiệm, nhưng hãy đảm bảo bạn cũng kiểm tra ứng dụng của mình bằng tài khoản người thử nghiệm không có giấy phép, đôi khi hoặc khi thực hiện các thay đổi lớn. Thử nghiệm không phải giấy phép giúp đảm bảo rằng ứng dụng của bạn không dựa vào việc kiểm tra logic cụ thể, chẳng hạn như thời gian gia hạn. Lưu ý: Người dùng trong rãnh thử nghiệm cũng có thể là người kiểm tra giấy phép cho ứng dụng của bạn.

## Thử nghiệm sản phẩm một lần

### Thử nghiệm các sản phẩm tiêu hao

Khi thử nghiệm các sản phẩm tiêu hao, chúng tôi khuyên bạn nên thử nghiệm nhiều tình huống khác nhau, bao gồm:

* Mua thành công trong đó người dùng nhận được một mặt hàng. Với người kiểm tra giấy phép, bạn có thể sử dụng công cụ Kiểm tra, luôn phê duyệt phương thức thanh toán.
* Giao dịch mua mà phương thức thanh toán không bị tính phí và người dùng không nên nhận mặt hàng. Với trình kiểm tra giấy phép, bạn có thể sử dụng công cụ Kiểm tra, luôn từ chối phương thức thanh toán.
* Đảm bảo các mặt hàng có thể được mua nhiều lần.

Bạn cũng nên xác minh rằng các giao dịch mua được xác nhận đúng như được mô tả trong xử [lý mua hàng](https://developer.android.com/google/play/billing/integrate#process). Đối với giao dịch mua từ người kiểm tra giấy phép, giao dịch mua sẽ được hoàn lại sau 3 phút nếu ứng dụng của bạn không xác nhận giao dịch mua và bạn sẽ nhận được email về việc hủy. Bạn cũng có thể kiểm tra tab Đơn hàng trong Google Play Console để xem đơn hàng có được hoàn tiền sau 3 phút không.

### Kiểm tra các sản phẩm không tiêu hao

Vật tư tiêu hao phải được kiểm tra giống như vật tư tiêu hao, nhưng bạn nên xác minh một mặt hàng không thể được mua lại trong ứng dụng của mình. Hãy chắc chắn xác minh xác nhận mua hàng cho cả hàng tiêu dùng và hàng tiêu dùng (nếu có) vì logic để xử lý mỗi loại mua hàng khác nhau.

Lưu ý: Để thực hiện nhiều giao dịch mua thử nghiệm cho cùng một sản phẩm không tiêu hao, bạn có thể [hoàn tiền và thu hồi giao dịch mua bằng Google Play Console](https://support.google.com/googleplay/android-developer/answer/2741495).

## Kiểm tra các tính năng dành riêng cho đăng ký

Luồng mua cho các sản phẩm và đăng ký một lần tương tự nhau, nhưng đăng ký có các kịch bản bổ sung, chẳng hạn như gia hạn đăng ký thành công hoặc bị từ chối. Để gia hạn thử nghiệm, bạn có thể sử dụng công cụ Kiểm tra, luôn phê duyệt và Công cụ kiểm tra, luôn từ chối các phương thức thanh toán có sẵn cho người thử nghiệm giấy phép, như trong hình 1. Sử dụng các công cụ thanh toán này để kiểm tra các tình huống ngoài kịch bản đăng ký thành công.

Similiar cho các sản phẩm một lần, bạn cũng nên xác minh rằng mua hàng được thừa nhận đúng như được mô tả trong xử [lý mua hàng](https://developer.android.com/google/play/billing/integrate#process). Đối với giao dịch mua từ người kiểm tra giấy phép, giao dịch mua sẽ được hoàn lại sau 3 phút nếu ứng dụng của bạn không xác nhận giao dịch mua và bạn sẽ nhận được email về việc hủy. Bạn cũng có thể kiểm tra tab Đơn hàng trong Google Play Console để xem đơn hàng có được hoàn tiền sau 3 phút không.

### Thời gian gia hạn

Đăng ký thử nghiệm gia hạn nhanh hơn đăng ký thực tế và đăng ký thử nghiệm có thể gia hạn tối đa sáu lần.

Bảng sau liệt kê thời gian gia hạn thử nghiệm cho đăng ký trong các khoảng thời gian khác nhau. Những thời điểm này là gần đúng. Bạn có thể thấy các biến thể nhỏ trong thời gian chính xác của một sự kiện. Để bù đắp cho biến thể, hãy gọi API để xem trạng thái hiện tại sau mỗi ngày hết hạn đăng ký.

|  |  |
| --- | --- |
| Thời gian đăng ký sản xuất | Thử nghiệm gia hạn đăng ký |
| 1 tuần | 5 phút |
| 1 tháng | 5 phút |
| 3 tháng | 10 phút |
| 6 tháng | 15 phút |
| 1 năm | 30 phút |

Các tính năng đăng ký dựa trên thời gian như bản dùng thử miễn phí cũng được rút ngắn để thử nghiệm. Bảng sau đây xác định khoảng thời gian thử nghiệm được liên kết với các tính năng đăng ký dựa trên thời gian:

|  |  |
| --- | --- |
| Tính năng | Thời gian thử nghiệm |
| Xác nhận mua hàng | 5 phút |
| Dùng thử miễn phí | 3 phút |
| Kỳ giá giới thiệu | Giống như thời gian thử nghiệm đăng ký |
| Thời gian ân hạn (cả 3 và 7 ngày) | 5 phút |
| Giữ tài khoản | 10 phút |
| Tạm dừng (1 tháng) | 5 phút |
| Tạm dừng (2 tháng) | 10 phút |
| Tạm dừng (3 tháng) | 15 phút |

### Trường hợp thử nghiệm

Mở rộng phần sau đây bằng cách nhấp vào Hiển thị /Ẩn để hiển thị các tình huống thử nghiệm bạn nên sử dụng để xác minh tích hợp đăng ký của mình.

HIỆN/ẨN

## Kiểm tra mã khuyến mãi

Bạn có thể sử dụng Google Play Console để [tạo mã cho thử nghiệm của riêng mình](https://support.google.com/googleplay/android-developer/answer/6321495). Lưu ý rằng bạn chỉ có thể tạo 500 mã khuyến mãi mỗi quý trên tất cả các sản phẩm được quản lý trong ứng dụng.

Bạn nên kiểm tra các tình huống đổi mã khuyến mãi sau:

* Khi mã khuyến mãi được nhập vào hộp thoại mua hàng đã được khởi chạy trong ứng dụng của bạn.
* Khi mã khuyến mãi được đổi trong ứng dụng Cửa hàng Google Play.
* Khi mã khuyến mãi được đổi tại https://play.google.com/store [sử](https://play.google.com/store) dụng nút Đổi trong điều hướng bên trái.

Trong những trường hợp này, bạn nên kiểm tra mã đổi theo nhiều cách nhất có thể. Chúng tôi khuyên bạn nên thực hiện tối thiểu các xét nghiệm sau:

* Đổi quà trước khi cài đặt ứng dụng.
* Đổi quà trong khi ứng dụng đang chạy ở nền trước. Lưu ý rằng đối với thử nghiệm này, bạn cần một thiết bị khác để thử nghiệm bằng ứng dụng Cửa hàng Google Play. Đảm bảo kiểm tra quy đổi từ các màn hình khác nhau trong ứng dụng của bạn.
* Đổi quà [với chế độ nhiều cửa](https://developer.android.com/guide/topics/ui/multi-window)sổ , nơi cả ứng dụng của bạn và ứng dụng Cửa hàng Google Play đang được hiển thị cùng mộtlúc.

Đối với mỗi thử nghiệm, hãy đảm bảo rằng mục được phát hiện chính xác và người dùng được thông báo.

# Chống gian lận và lạm dụng

Khi ứng dụng của bạn trở nên phổ biến, ứng dụng cũng có thể thu hút sự chú ý không mong muốn của những người dùng độc hại có thể muốn lạm dụng ứng dụng của bạn. Chủ đề này mô tả các đề xuất mà bạn nên sử dụng để giúp ngăn chặn các cuộc tấn công này vào tích hợp thanh toán và giảm tác động của hành vi lạm dụng trong ứng dụng của bạn.

## Di chuyển logic nhạy cảm sang backend của bạn

Nhiều như thiết kế ứng dụng của bạn cho phép, hãy di chuyển dữ liệu nhạy cảm và logic đến máy chủ phụ trợ mà bạn kiểm soát. Bạn càng có nhiều dữ liệu và logic trong một thiết bị frontend, bạn càng dễ bị sửa đổi hoặc giả mạo.

Ví dụ, một trò chơi cờ vua trực tuyến nên xác thực tất cả các động thái trong backend thay vì tin tưởng rằng frontend luôn gửi các động thái pháp lý.

Hơn nữa, nếu bạn tìm thấy lỗ hổng hoặc vấn đề bảo mật, tùy thuộc vào thiết kế hệ thống của bạn, có thể dễ dàng gỡ lỗi, khắc phục và tung ra các bản cập nhật trên backend thay vì frontend.

## Xác minh giao dịch mua trước khi cấp quyền lợi

Một trường hợp đặc biệt của dữ liệu nhạy cảm và logic cần được xử lý trong backend là xác minh mua hàng. Sau khi người dùng đã mua hàng, bạn nên làm như sau:

1. Gửi giao dịch mua tương ứngToken đến backend của bạn. Điều này có nghĩa là bạn nên duy trì hồ sơ của tất cả các giá trị purchaseToken cho tất cả các giao dịch mua.
2. Xác minh rằng giá trị purchaseToken cho giao dịch mua hiện tại không khớp với bất kỳ giá trị purchaseToken nào trước đó. purchaseToken là duy nhất trên toàn cầu, vì vậy bạn có thể sử dụng giá trị này một cách an toàn làm khóa chính trong cơ sở dữ liệu của mình.
3. Sử dụng [Purchases.products:get](https://developers.google.com/android-publisher/api-ref/purchases/products) hoặc [Purchases.subscriptions:get](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) endpoints trong API nhà phát triển trên Google Play để xác minh với Google rằng giao dịch mua là hợp pháp.
4. Nếu việc mua hàng là hợp pháp và chưa được sử dụng trong quá khứ, sau đó bạn có thể cấp quyền truy cập một cách an toàn cho mục hoặc đăng ký trong ứng dụng.
5. Đối với đăng ký, [khi linkedPurchaseToken được](https://developers.google.com/android-publisher/api-ref/purchases/subscriptions) đặt trong Purchases.subscriptions:get, bạn cũng nên xóa linkedPurchaseToken khỏi cơ sở dữ liệu của mình và thu hồi quyền được cấp cho linkedPurchaseToken để đảm bảo rằng nhiều người dùng không được hưởng cùng một giao dịch mua.

Lưu ý: Không sử dụng orderId để kiểm tra các giao dịch mua trùng lặp hoặc làm khóa chính trong cơ sở dữ liệu của bạn, vì không phải tất cả các giao dịch mua đều được đảm bảo tạo orderId. Đặc biệt, các giao dịch mua được thực hiện bằng mã khuyến mãi không tạo ra một orderId.

## Bảo vệ nội dung đã mở khóa của bạn

Để ngăn người dùng độc hại phân phối lại nội dung đã mở khóa của bạn, đừng đóng gói nội dung đó vào tệp APK của bạn. Thay vào đó, hãy thực hiện một trong các tác vụ sau:

* Sử dụng dịch vụ theo thời gian thực để phân phối nội dung của bạn, chẳng hạn như nguồn cấp dữ liệu nội dung. Phân phối nội dung thông qua dịch vụ theo thời gian thực cũng cho phép bạn giữ cho nội dung của mình luôn mới mẻ.
* Sử dụng máy chủ từ xa để phân phối nội dung của bạn.

Khi bạn phân phối nội dung từ máy chủ từ xa hoặc dịch vụ thời gian thực, bạn có thể lưu trữ nội dung đã mở khóa trong bộ nhớ thiết bị hoặc lưu trữ nội dung đó trên thẻ SD của thiết bị. Nếu bạn lưu trữ nội dung trên thẻ SD, hãy đảm bảo mã hóa nội dung và sử dụng khóa mã hóa dành riêng cho thiết bị.

## Phát hiện và xử lý các giao dịch mua bị vô hiệu hóa

Giao dịch mua bị hủy là các giao dịch mua đã bị hủy, thu hồi hoặc bị tính phí lại. Nếu giao dịch mua bị hủy trước đó đã cấp các mặt hàng trong ứng dụng hoặc nội dung khác cho người dùng, bạn có thể [sử dụng API Giao dịch mua bị vô hiệu hóa](https://developers.google.com/android-publisher/voided-purchases) để có được lý do giao dịch mua bị vô hiệu hóa cùng với bất kỳ nội dung liên quan nào mà bạn có thể vuốt lại.

Lưu ý: Các giao dịch mua bị vô hiệu hóa mà không có bất kỳ nội dung vuốt ngược nào được hiển thị bởi API Mua hàng bị vô hiệu hóa.

Việc mua các mặt hàng và gói đăng ký trong ứng dụng có thể bị hủy vì nhiều lý do, bao gồm:

* Người dùng, nhà phát triển hoặc Google đã hủy giao dịch mua. Đối với đăng ký, lưu ý rằng điều này đề cập đến việc hủy bỏ việc mua đăng ký, [thay vì tự hủy đăng ký](https://support.google.com/googleplay/answer/7018481?co=GENIE.Platform%3DAndroid&hl=en).
* Một giao dịch mua được tính phí lại.
* Nhà phát triển ứng dụng hủy hoặc hoàn tiền đơn đặt hàng của người dùng và kiểm tra tùy chọn "thu hồi" trong bảng điều khiển.

Dựa trên lý do mua hàng bị vô hiệu hóa và tính đến dữ liệu hành vi của người dùng trước đó, bạn có thể quyết định một quá trình hành động. Chúng tôi khuyên bạn nên thực hiện một hoặc nhiều cách sau:

* Thực hiện móng vuốt: Khi việc mua hàng bị vô hiệu hóa, bạn có thể vuốt lại các mặt hàng chưa sử dụng như thể chúng chưa bao giờ được mua. Ví dụ: nếu việc mua tiền tệ trong trò chơi bị vô hiệu hóa, bạn có thể vuốt lại đơn vị tiền tệ đã được cấp cho người dùng. Trong trường hợp người dùng đã chi tiêu đơn vị tiền tệ, hãy cân nhắc đặt số dư tiền tệ thành âm và hạn chế hoạt động ứng dụng và giao dịch mua trong tương lai cho đến khi số dư tiền tệ dương.
* Triển khai nhiều cuộc đình công: Cân nhắc thực hiện các hành động ít quyết liệt hơn đối với người phạm tội lần đầu, chẳng hạn như hiển thị cảnh báo trong ứng dụng. Đối với người tái phạm, hãy xem xét các biện pháp nghiêm trọng hơn.
* Tạm thời vô hiệu hóa giao dịch mua: Tương tự như triển khai nhiều cảnh cáo, hãy cân nhắc vô hiệu hóa giao dịch mua cho người dùng có giao dịch mua bị vô hiệu hóa cho đến khi bạn có thể điều tra kỹ lưỡng hơn lý do tại sao giao dịch mua bị vô hiệu hóa.
* Tạm thời hoặc vĩnh viễn không cho phép truy cập vào ứng dụng của bạn: Đối với các trường hợp cực đoan có hoạt động độc hại lặp đi lặp lại, hãy cân nhắc không cho phép truy cập vào ứng dụng của bạn, tạm thời hoặc vĩnh viễn.
* Thực hiện cuộc gọi thường xuyên đến API Mua hàng bị vô hiệu hóa: Khi bạn phát hiện một hoặc nhiều giao dịch mua bị vô hiệu hóa, hãy cân nhắc thực hiện các cuộc gọi thường xuyên hơn đến API Mua hàng bị vô hiệu hóa để vuốt lại giao dịch mua trước khi người dùng có thể sử dụng chúng. Bạn có thể tìm thêm thông tin về hạn ngạch API Mua hàng bị vô hiệu hóa trong [tài liệu API Giao dịch mua bị vô hiệu](https://developers.google.com/android-publisher/voided-purchases#quotas).

## Giúp Google phát hiện gian lận trước khi xảy ra

Một số loại gian lận có liên quan đến những người dùng độc hại tạo nhiều tài khoản Google và trong ứng dụng để ẩn hoạt động của họ.

Sử dụng [các phương pháp setObfuscatedAccountId](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams.Builder#setObfuscatedAccountId(java.lang.String)) và [setObfuscatedProfileId](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams.Builder#setobfuscatedprofileid) trong trình tạo cho [BillingFlowParams](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams.Builder) để giúp Google ánh xạ tài khoản Google với tài khoản trong ứng dụng.

Google sử dụng dữ liệu này để phát hiện hành vi đáng ngờ và chặn một số loại giao dịch gian lận trước khi hoàn tất.

## Thực hiện hành động chống lại vi phạm nhãn hiệu và bản quyền

Nếu bạn đang sử dụng máy chủ từ xa để phân phối hoặc quản lý nội dung, hãy yêu cầu ứng dụng của bạn xác minh trạng thái mua của nội dung đã mở khóa bất cứ khi nào người dùng truy cập nội dung. Điều này cho phép bạn thu hồi việc sử dụng khi cần thiết và giảm thiểu vi phạm bản quyền. Nếu bạn thấy nội dung của mình được phân phối lại trên Google Play, hãy nhớ hành động nhanh chóng và dứt khoát. Để biết thêm chi tiết, hãy xem [trang Câu hỏi thường gặp về bản](https://support.google.com/legal/answer/4558836?hl=en) quyền trong Trung tâm trợ giúp bản quyền.

# AIDL to Google Play Billing Library migration guide

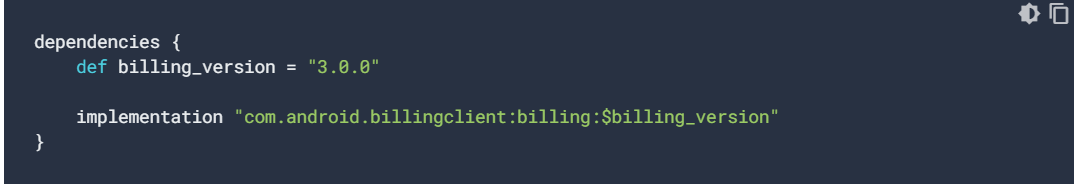
Chủ đề này mô tả cách di chuyển ra khỏi tích hợp thanh toán sử dụng Ngôn [ngữ định nghĩa giao diện Android (AIDL)](https://developer.android.com/google/play/billing/api). Việc truy cập vào hệ thống thanh toán của Google Play bằng AIDL sẽ không còn được dùng nữa và tất cả các tiện ích tích hợp phải sử dụng Thư viện thanh toán của Google Play trong tương lai.

**Lưu ý:** Chủ đề này nêu bật những thay đổi quan trọng nhất cần thiết để di chuyển. Để biết các bước tích hợp chi tiết, hãy xem các liên kết được tham chiếu trong trang này cùng với đoạn mã trong các mẫu dành riêng cho thanh [toán](https://github.com/android/play-billing-samples) được lưu trữ trên GitHub.

## Các bước di chuyển

### Nhập Thư viện thanh toán của Google Play

Trước tiên, hãy thêm phụ thuộc vào Thư viện thanh toán của Google Play. Nếu bạn đang sử dụng Gradle, bạn có thể thêm nội dung sau vào tệp build.gradle của ứng dụng:

Bạn có thể xóa bất kỳ mã "keo" nào, chẳng hạn [như IabHelper](https://github.com/android/play-billing-samples/blob/7a94c6905a9c125518354c216b5c3094fde47ce1/TrivialDrive/app/src/main/java/com/example/android/trivialdrivesample/util/IabHelper.java), mà bạn có thể đã sao chép từ mã tham chiếu trước đó. Chức năng được cung cấp bởi IabHelper hiện là một phần của Thư viện thanh toán của Google Play.

### Xóa quyền com.android.vending.BILLING

Thư viện thanh toán của Google Play nhúng quyền com.android.vending.BILLING bên trong bản kê khai. Không còn cần thiết phải thêm rõ ràng quyền này vào bản kê khai của ứng dụng.

### Kết nối với Thanh toán trên Google Play

[BillingClient](https://developer.android.com/reference/com/android/billingclient/api/BillingClient) của Thư viện thanh toán Google Play xử lý việc quản lý kết nối cho bạn. Để di chuyển, hãy thực hiện các thay đổi sau đối với ứng dụng của bạn:

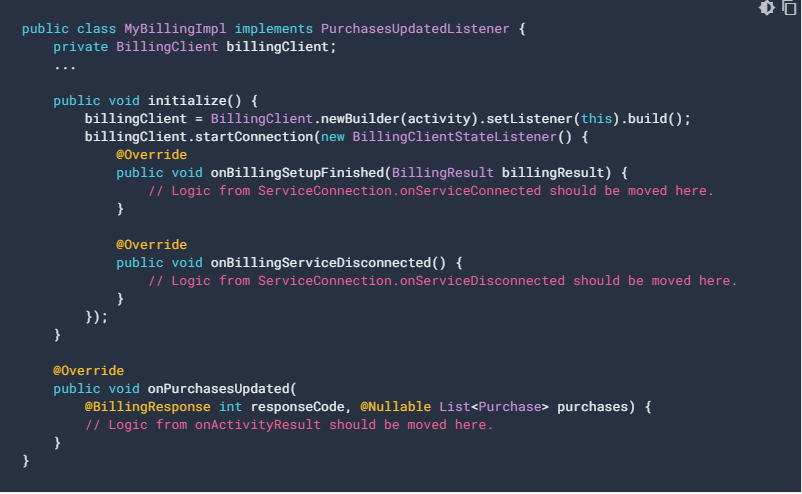
* Tạo một phiên bản của BillingClient.
* Triển khai [BillingClientStateListener để nhận](https://developer.android.com/reference/com/android/billingclient/api/BillingClientStateListener) gọi lại về trạng thái dịch vụ.
* Gọi [startConnection() trên](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#startConnection(com.android.billingclient.api.BillingClientStateListener)) phiên bản BillingClient của bạn.
* Xóa [mã onActivityResult()](https://developer.android.com/reference/android/app/Activity#onActivityResult(int,%20int,%20android.content.Intent)) liên quan đến mua hàng trong ứng dụng và chuyển sang [PurchasesUpdatedListener](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener).

Các ví dụ sau đây cho thấy ứng dụng của bạn có thể trông như thế nào trước và sau khi thực hiện những thay đổi sau:

**Trước**

}

**Sau**

}

### Mua hàng

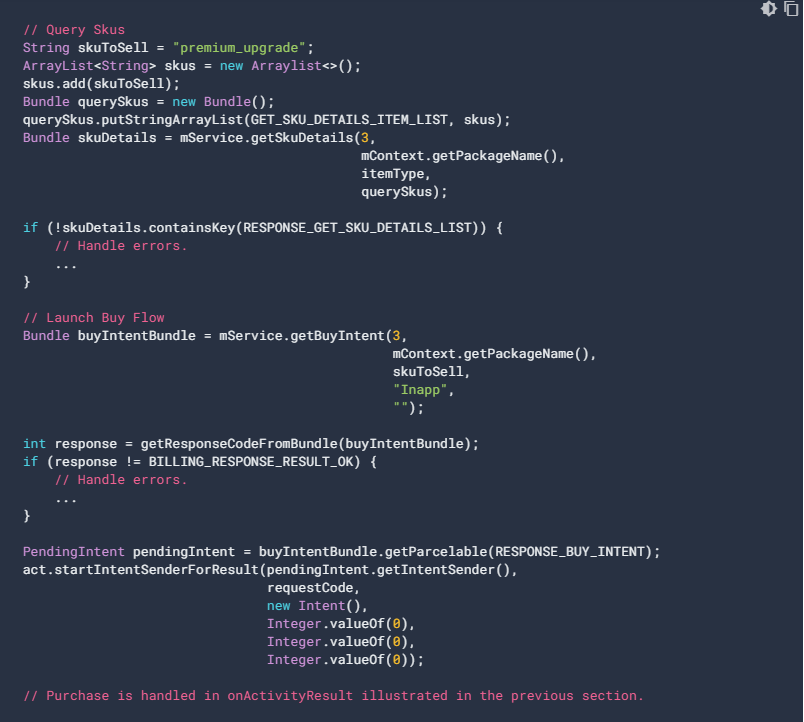
**Lưu ý:** Bạn cần truy vấn chi tiết SKU cho phiên bản 2.0 trở lên của Thư viện thanh toán của Google Play.

Để khởi chạy hộp thoại mua hàng, hãy làm như sau:

* Chuyển đổi chi tiết SKU của bạn [Gói thành SkuDetailsParams](https://developer.android.com/reference/com/android/billingclient/api/SkuDetailsParams).
* Chuyển cuộc [gọi mService.getSkuDetails()](https://developer.android.com/google/play/billing/billing_reference#getSkuDetails) để thay vào đó sử dụng [BillingClient.querySkuDetailsAsync()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querySkuDetailsAsync(com.android.billingclient.api.SkuDetailsParams,%20com.android.billingclient.api.SkuDetailsResponseListener))
* Chuyển đổi gói ý định mua hàng của bạn [thành đối tượng BillingFlowParams.](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams)
* Chuyển cuộc [gọi mService.getBuyIntent()](https://developer.android.com/google/play/billing/billing_reference#getBuyIntent) để thay vào đó sử dụng [BillingClient.launchBillingFlow()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#launchBillingFlow(android.app.Activity,%20com.android.billingclient.api.BillingFlowParams)).
* Xóa bất kỳ mã liên quan đến mua hàng trong [ứng dụng từ onActivityResult()](https://developer.android.com/reference/android/app/Activity#onActivityResult(int,%20int,%20android.content.Intent))và di chuyển mã này vào [PurchasesUpdatedListener](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener).

Các ví dụ sau đây cho thấy ứng dụng của bạn có thể trông như thế nào trước và sau khi thực hiện những thay đổi sau:

**Trước**



**Sau**



### Tiêu thụ giao dịch mua

Để tiêu thụ giao dịch mua bằng Thư viện thanh toán của Google Play, hãy làm như sau:

* Thay vì gọi [consumePurchase(),](https://developer.android.com/google/play/billing/billing_reference#consumePurchase)gọi [BillingClient.consumeAsync()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#consumeAsync(com.android.billingclient.api.ConsumeParams,%20com.android.billingclient.api.ConsumeResponseListener)).
* Thực [hiện ConsumeResponseListener](https://developer.android.com/reference/com/android/billingclient/api/ConsumeResponseListener).

Các ví dụ sau đây cho thấy ứng dụng của bạn có thể trông như thế nào trước và sau khi thực hiện những thay đổi sau:

**Trước**

}

**Sau**

### Xác nhận giao dịch mua

Bắt đầu với phiên bản 2.0 của Thư viện thanh toán Google Play, ứng dụng của bạn phải tiêu thụ hoặc [xác nhận tất cả các giao dịch mua](https://developer.android.com/google/play/billing/billing_library_releases_notes#2_0_acknowledge).

Nếu bạn không tiêu thụ hoặc xác nhận giao dịch mua trong vòng ba ngày, Google sẽ tự động thu hồi giao dịch mua và hoàn tiền cho người dùng. Để biết thêm thông tin, hãy [xem Xác nhận mua hàng](https://developer.android.com/google/play/billing/billing_library_overview#acknowledge).

### Nhận biết các giao dịch mua ngoài ứng dụng

Để di chuyển xử lý mua hàng ngoài ứng dụng sang Thư viện thanh toán của Google Play, hãy làm như sau:

* Đảm bảo rằng ứng dụng của [bạn gọi BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#querypurchases) trong cuộc gọi lại onResume() của ứng dụng.
* Loại bỏ bộ thu phát sóng com.android.vending.billing.PURCHASES\_UPDATED và di chuyển mã gọi lại tương ứng vào [PurchasesUpdatedListener của bạn](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener).

Trước đây, khi tích hợp Google Play Billing với AIDL, ứng dụng của bạn sẽ cần đăng ký người nghe để nhận com.android.vending.billing.PURCHASES\_UPDATED ý định xử lý các giao dịch mua được thực hiện bên ngoài ứng dụng của bạn.

Với Thư viện thanh toán của Google Play, ứng dụng của bạn phải luôn [gọi queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#queryPurchases(java.lang.String)) [trong cuộc gọi lại onResume()](https://developer.android.com/guide/components/activities/activity-lifecycle#onresume) của ứng dụng là bước đầu tiên để đảm bảo rằng tất cả các giao dịch mua được thực hiện trong khi ứng dụng của bạn không chạy đều được nhận dạng. Trong khi ứng dụng của bạn đang chạy, thư viện sẽ tự động lắng nghe các giao dịch mua ngoài ứng dụng, thông báo cho bạn thông qua [PurchasesUpdatedListener](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener).

### Xử lý các giao dịch đang chờ xử lý

Bắt đầu với phiên bản 2.0 của Thư viện thanh toán google Play, ứng dụng của bạn phải *xử lý các giao dịch đang chờ xử* lý cần hành động bổ sung sau khi mua trước khi cấp quyền. Ví dụ: người dùng có thể chọn mua sản phẩm trong ứng dụng của bạn tại cửa hàng thực bằng tiền mặt. Điều này có nghĩa là giao dịch được hoàn tất bên ngoài ứng dụng của bạn. Trong trường hợp này, bạn nên cấp quyền chỉ sau khi người dùng đã hoàn thành giao dịch.

Để biết thêm thông tin, hãy xem [Hỗ trợ giao dịch đang chờ xử lý](https://developer.android.com/google/play/billing/billing_library_overview#pending).

### Tải trọng dành cho nhà phát triển

Tải trọng của nhà phát triển trong lịch sử đã được sử dụng cho các mục đích khác nhau, bao gồm ngăn chặn gian lận và quy kết mua hàng cho đúng người dùng. Giờ đây, Thư viện thanh toán của Google Play hỗ trợ các trường hợp sử dụng này, chúng tôi đã không còn được dùng nữa, tải trọng dành cho nhà phát triển bắt đầu bằng phiên bản 2.2 của Thư viện thanh toán của Google Play. Để biết thêm thông tin, hãy xem [Tải trọng dành cho nhà phát triển](https://developer.android.com/google/play/billing/developer-payload).

### Thông báo lỗi chi tiết

Bắt đầu với phiên bản 2.0 của Thư viện thanh toán Google Play, tất cả các lỗi đều có thông báo liên quan đến gỡ lỗi tương ứng. Những tin nhắn này có thể được lấy [bằng cách gọi BillingResult.getDebugMessage()](https://developer.android.com/reference/com/android/billingclient/api/BillingResult#getDebugMessage()).

# Developer Payload

Tải trọng của nhà phát triển trong lịch sử đã được sử dụng cho các mục đích khác nhau, bao gồm ngăn chặn gian lận và quy kết mua hàng cho đúng người dùng. Với phiên bản 2.2 trở lên của Thư viện thanh toán google Play, các trường hợp sử dụng dự định trước đây dựa vào tải trọng của nhà phát triển hiện được hỗ trợ đầy đủ ở các phần khác của thư viện.

Với hỗ trợ này tại chỗ, chúng tôi đã không còn là tải trọng của nhà phát triển, bắt đầu với phiên bản 2.2 của Thư viện thanh toán Google Play. Các phương pháp liên quan đến tải trọng dành cho nhà phát triển đã không còn được dùng nữa trong phiên bản 2.2 và đã bị xóa trong phiên bản 3.0. Xin lưu ý rằng ứng dụng của bạn có thể tiếp tục truy xuất tải trọng dành cho nhà phát triển cho các giao dịch mua được thực hiện bằng cách sử dụng các phiên bản trước của thư viện hoặc AIDL.

Để biết danh sách chi tiết các thay đổi, hãy xem [ghi chú phát hành Thư viện thanh toán 2.2](https://developer.android.com/google/play/billing/release-notes#2-2)của Google Playvà ghi chú phát hành Thư viện thanh toán Google [Play 3.0](https://developer.android.com/google/play/billing/release-notes#3-0).

**Lưu ý:** Nếu bạn có một trường hợp sử dụng không được đề cập dưới đây, vui lòng gửi [lỗi](https://issuetracker.google.com/issues/new?component=311487&template=1014446).

## Xác minh giao dịch mua hàng

Để đảm bảo giao dịch mua là xác thực và không bị giả mạo hoặc phát lại, Google khuyên bạn nên sử dụng mã thông báo mua hàng (thu được [từ phương thức getPurchaseToken()](https://developer.android.com/reference/com/android/billingclient/api/Purchase#getpurchasetoken) trong đối [tượng Mua](https://developer.android.com/reference/com/android/billingclient/api/Purchase) hàng) cùng với API Nhà phát triển trên Google Play để xác minh giao dịch mua là xác thực. Để biết thêm thông tin, hãy xem Chống [gian lận và lạm dụng](https://developer.android.com/google/play/billing/security).

## Phân bổ mua hàng

Nhiều ứng dụng, đặc biệt là trò chơi, cần đảm bảo rằng giao dịch mua được quy cho chính xác nhân vật / hình đại diện trong trò chơi hoặc hồ sơ người dùng trong ứng dụng đã bắt đầu mua hàng. Bắt đầu với Google Play Billing Library 2.2, ứng dụng của bạn có thể chuyển số nhận dạng hồ sơ và tài khoản bị xáo trộn cho Google khi khởi chạy hộp thoại mua hàng và yêu cầu họ trả lại khi truy xuất giao dịch mua.

Sử dụng các [tham số setObfuscatedAccountId()](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams.Builder#setObfuscatedAccountId(java.lang.String)) và [setObfuscatedProfileId()](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams.Builder#setobfuscatedprofileid) [trong BillingFlowParams và](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams) truy xuất chúng bằng phương [thức getAccountIdentifiers()](https://developer.android.com/reference/com/android/billingclient/api/Purchase#getAccountIdentifiers()) trong đối tượng [Mua.](https://developer.android.com/reference/com/android/billingclient/api/Purchase)

**Lưu ý:** Mua hàng được thực hiện bằng cách sử dụng các phiên bản trước của thư viện bằng cách sử dụng [setAccountId()](https://developer.android.com/reference/com/android/billingclient/api/BillingFlowParams.Builder#setAccountId(java.lang.String)) (đổi tên thành setObfuscatedAccountId()) không được trả về bởi getAccountIdentifiers().

## Liên kết siêu dữ liệu với giao dịch mua

Google khuyên bạn nên lưu trữ siêu dữ liệu về giao dịch mua trên máy chủ phụ trợ an toàn mà bạn duy trì. Siêu dữ liệu mua hàng này phải được liên kết với mã thông báo mua hàng thu được bằng [cách sử dụng phương pháp getPurchaseToken](https://developer.android.com/reference/com/android/billingclient/api/Purchase#getPurchaseToken()) trong đối [tượng Mua.](https://developer.android.com/reference/com/android/billingclient/api/Purchase) Dữ liệu này có thể được duy trì bằng cách chuyển mã thông báo mua hàng và siêu dữ liệu vào backend của bạn [khi PurchasesUpdatedListener của bạn](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener) được gọi sau khi mua thành công.

Để đảm bảo siêu dữ liệu được liên kết trong trường hợp gián đoạn luồng mua hàng, Google khuyên bạn nên lưu trữ siêu dữ liệu trên máy chủ phụ trợ của bạn trước khi khởi chạy hộp thoại mua hàng và liên kết siêu dữ liệu đó với ID tài khoản của người dùng, SKU đang được mua và dấu thời gian hiện tại.

Nếu luồng mua hàng bị gián đoạn trước khi [PurchasesUpdatedListener của bạn](https://developer.android.com/reference/com/android/billingclient/api/PurchasesUpdatedListener) được gọi, ứng dụng của bạn sẽ khám phá giao dịch mua sau khi ứng dụng của bạn tiếp tục và [gọi BillingClient.queryPurchases()](https://developer.android.com/reference/com/android/billingclient/api/BillingClient#queryPurchases(java.lang.String)). Sau đó, bạn có thể gửi các giá trị được truy xuất từ [getPurchaseTime()](https://developer.android.com/reference/com/android/billingclient/api/Purchase#getPurchaseTime())của đối tượngMua, [getSku()](https://developer.android.com/reference/com/android/billingclient/api/Purchase#getsku)và [getPurchaseToken() đến](https://developer.android.com/reference/com/android/billingclient/api/Purchase#getpurchasetoken) máy chủ backend của bạn để tra cứu siêu dữ liệu, liên kết siêu dữ liệu với mã thông báo mua hàng và tiếp tục xử lý giao dịch mua.

# Sử dụng Thư viện thanh toán của Google Play với Unity

Plugin Google Play Billing mở rộng các dịch vụ và tài sản tích hợp của Unity để mua hàng trong ứng dụng, được gọi [là Unity IAP](https://docs.unity3d.com/Manual/UnityIAP.html), để cung cấp cho tròchơi của bạn tất cả các tính năng mới nhất của Thư viện thanh toán Google Play. Hướng dẫn này giải thích cách thiết lập dự án của bạn để sử dụng plugin. Hướng dẫn này cũng mô tả cách triển khai các tính năng của Thư viện thanh toán trong trò chơi của bạn trong Unity.

### Thiết lập plugin Thanh toán trên Google Play

Để thiết lập plugin, hãy hoàn thành các bước trong mỗi phần được liên kết sau:

1. [Bật lớp trừu tượng Unity IAP](https://developer.android.com/google/play/billing/unity#abstraction-layer).
2. [Tải xuống và nhập plugin](https://developer.android.com/google/play/billing/unity#import-plugin).
3. [Cấu hình cài đặt xây dựng của plugin](https://developer.android.com/google/play/billing/unity#plugin-build-settings).
4. [Bật plugin](https://developer.android.com/google/play/billing/unity#enable-plugin).

#### Bật lớp trừu tượng Unity IAP

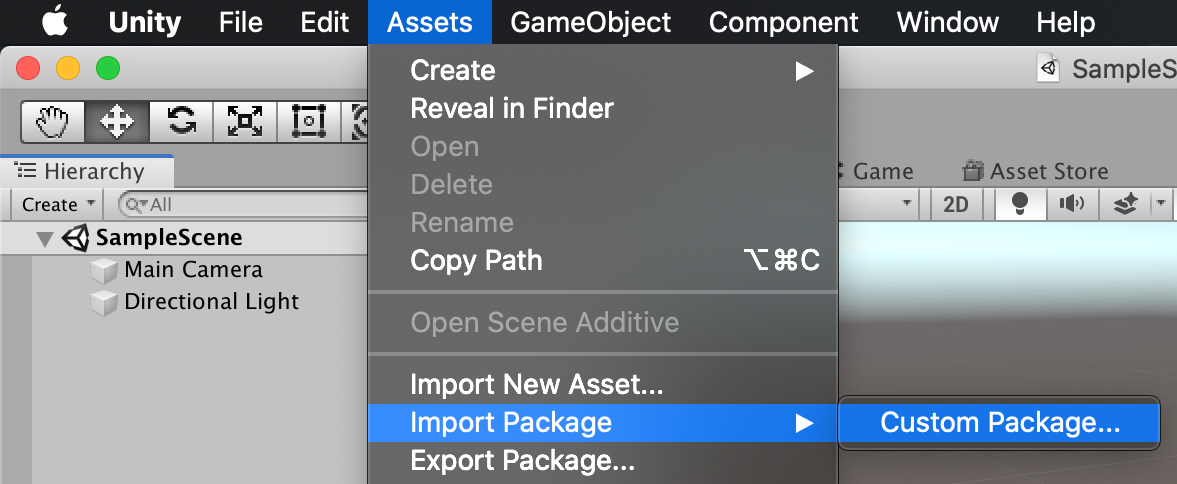
Plugin Google Play Billing được xây dựng trên một layer trừu tượng đi kèm với Unity IAP, vì vậy bạn cần bật layer trừu tượng này trước khi tải xuống và nhập plugin. Để kích hoạt lớp trừu tượng Unity IAP, hãy làm như sau:

1. Hoàn thành tất cả các bước trong hướng dẫn Unity sau: [Thiết lập dự án của bạn cho Unity Services](https://docs.unity3d.com/Manual/SettingUpProjectServices.html).
2. Hoàn thành tất cả các bước trong hướng dẫn Unity sau: [Bật dịch vụ Unity IAP](https://docs.unity3d.com/Manual/UnityIAPSettingUp.html).

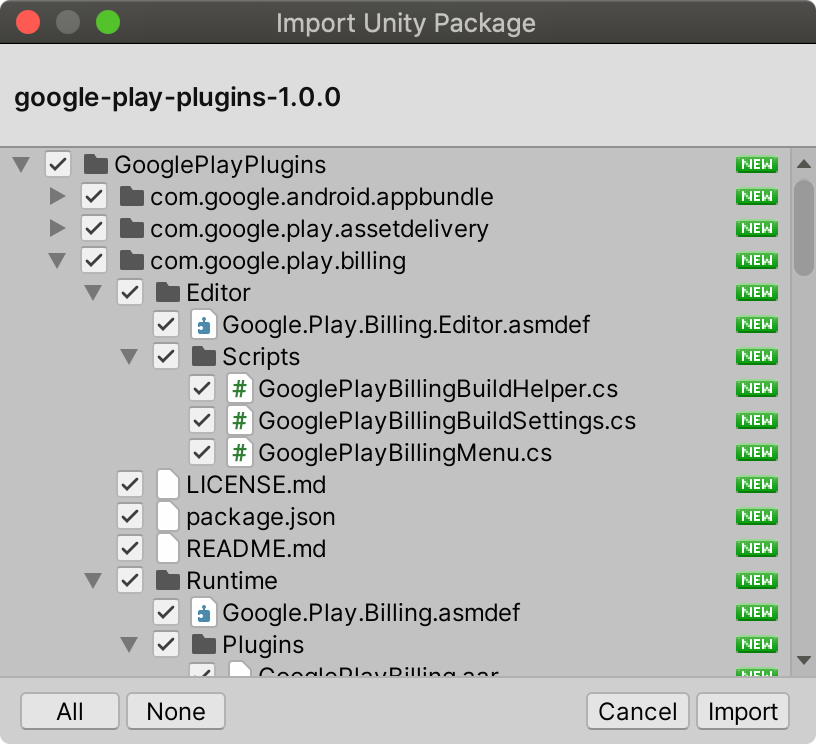
#### Tải xuống và nhập plugin

Plugin được vận chuyển dưới dạng gói Unity ở định dạng .unitypackage. Để tải xuống và nhập plugin, hãy làm theo các bước sau:

1. Tải xuống bản phát hành mới nhất của Google Play Plugins for Unity từ trang phát hành của [kho lưu trữ trên GitHub](https://github.com/google/play-unity-plugins/releases).
2. Từ thanh menu Unity, nhấp vào **Tài sản > Nhập Gói > Gói Tùy chỉnh**.



1. Xác định vị trí bạn đã tải xuống tệp .unitypackage và chọn tệp đó.
2. Trong hộp thoại **Nhập Gói Thống nhất,** hãy để tất cả tài sản được chọn và bấm **Nhập**.

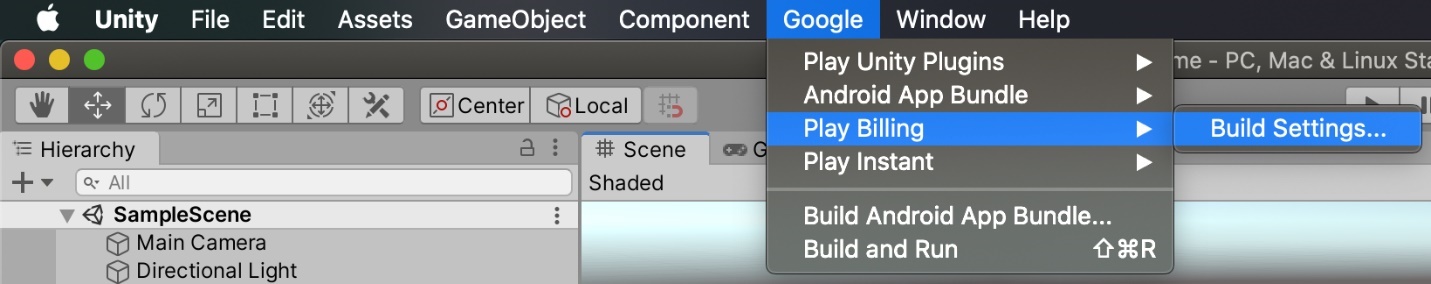


Sau khi nhập gói, một thư mục mới có **tên GooglePlayPlugins** (ở thư mục gốc của thư mục Tài sản) sẽ được thêm vào tài sản của dự án. Thư mục này chứa tất cả các tài sản Thư viện Thanh toán cho plugin.

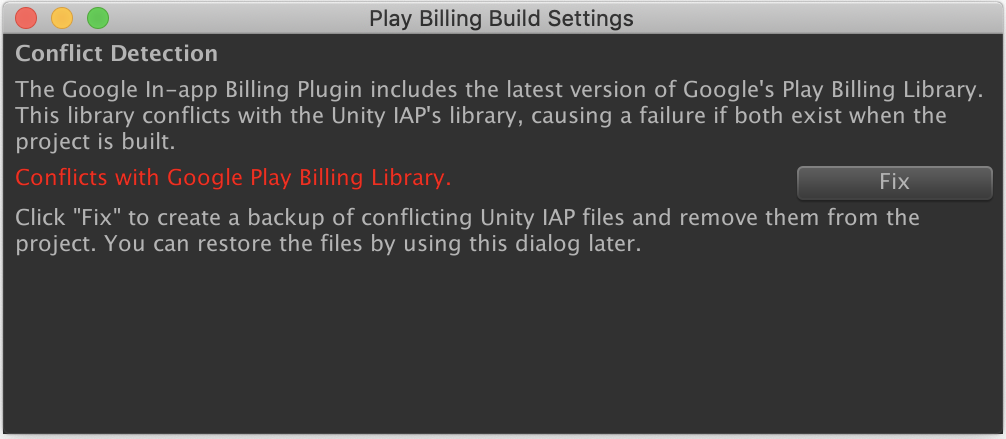
#### Cấu hình thiết đặt bản dựng

Bởi vì plugin mở rộng Unity IAP, Unity sẽ gặp xung đột và không xây dựng APK Android trừ khi một số phụ thuộc cũ hơn, chồng chéo trong Unity IAP bị xóa khỏi bản dựng. Plugin cung cấp một cách tự động để loại bỏ các thư viện xung đột khỏi dự án của bạn. Để giải quyết các xung đột này, hãy làm theo các bước sau:

1. Từ thanh menu Unity, chọn Google > **Thanh toán > cài đặt bản dựng**.

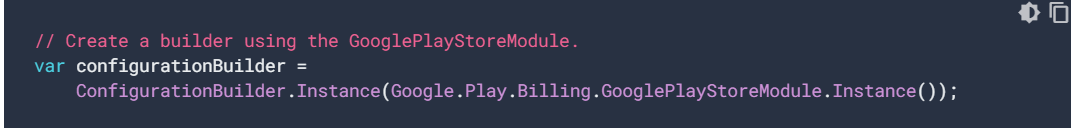


1. Trong cửa sổ Phát Thiết đặt Bản dựng Thanh toán, bấm **Sửa chữa**. Điều này giải quyết xung đột và di chuyển các tệp IAP Unity xung đột vào thư mục sao lưu. Sau khi bạn bấm **Sửa**chữa , nút thay đổithành Khôi **phục**, bạn có thể bấm để khôi phục các tệp gốc, xung đột.

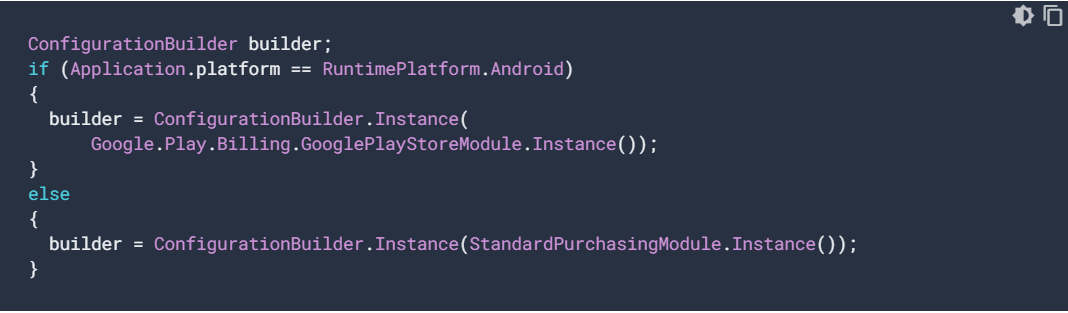


#### Bật plugin

Để kích hoạt plugin, hãy thay thế việc triển khai Google Play của Unity IAP bằng plugin Google Play Billing. Ví dụ: khi sử dụng [Unity IAP Purchaser Script](https://learn.unity.com/tutorial/unity-iap#5c7f8528edbc2a002053b46e), bạn sẽ thay đổi StandardPurchaseModule được chuyển vào trình tạo IAP để sử dụng Google.Play.Billing.GooglePlayStoreModule:

**Lưu ý:** Để tránh mọi xung đột tiềm ẩn với việc triển khai IAP tiêu chuẩn của Unity, bạn nên chỉ định đầy đủ không gian tên bằng cách bao gồm Google.Play.Billing bất cứ khi nào mã của bạn đề cập đến các lớp công khai trong plugin Google Play Billing. Tất cả các ví dụ mã trong hướng dẫn này sử dụng phương pháp này.

Nếu trò chơi của bạn sử dụng cùng một Tập lệnh người mua cho nhiều nền tảng, thì bạn nên thêm kiểm tra nền tảng để đảm bảo rằng Unity sẽ tiếp tục sử dụng giải pháp IAP của riêng mình cho các nền tảng khác:

Nếu xuất bản trò chơi của mình trên các cửa hàng ứng dụng Android khác bên cạnh Cửa hàng Google Play, bạn chỉ nên thay thế triển khai Unity IAP mặc định khi chọn Cửa hàng Google Play:

## Triển khai các tính năng của Thư viện thanh toán trên Google Play trong trò chơi của bạn

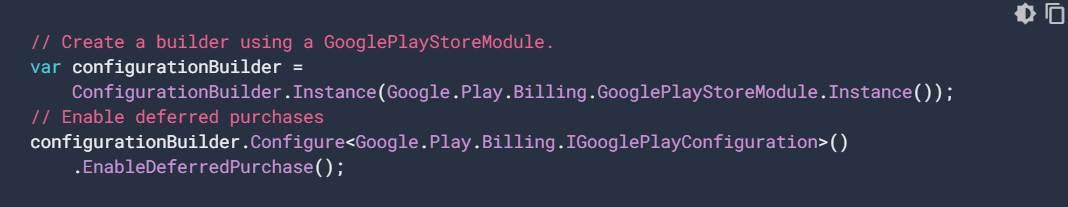
Plugin Google Play Billing mở rộng các dịch vụ Unity IAP, vì vậy bạn có thể sử dụng cùng một API Unity để quản lý quy trình mua hàng phổ biến. Lưu ý rằng có một số [thay đổi nhỏ đối với hành vi API](https://developer.android.com/google/play/billing/unity#behavior-changes) do sự khác biệt giữa Thư viện thanh toán và triển khai IAP tiêu chuẩn của Unity cho các cửa hàng ứng dụng khác. Nếu bạn mới sử dụng API Unity IAP, hãy xem phần "Tạo tập lệnh mua hàng" trong hướng [dẫn Unity IAP để](https://learn.unity.com/tutorial/unity-iap#5c7f8528edbc2a002053b46e) biết ví dụ về cách triển khai các luồng mua cơ bản.

Thư viện thanh toán cũng bao gồm một số tính năng duy nhất cho cửa hàng Google Play. Bạn có thể truy cập các tính năng này thông qua giao diện mở rộng. Phần còn lại của phần này mô tả cách triển khai các tính năng Thư viện thanh toán độc đáo này trong trò chơi của bạn.

### Cho phép mua hàng trả chậm

Google Play hỗ trợ mua hàng trả chậm — còn được gọi [là trasacation đang](https://developer.android.com/google/play/billing/billing_library_overview#pending) chờ xử lý hoặc giao dịch mua đang chờ xử lý — nơi người dùng có thể tạo giao dịch mua và hoàn tất giao dịch mua sau này bằng tiền mặt trong cửa hàng.

Để cho phép mua hàng hoãn lại, hãy sử dụng trình tạo IAP của bạn để sửa đổi cấu hình mô-đun của bạn bằng cách gọi phương thức EnableDeferredPurchase() :



Tiếp theo, triển khai gọi lại giao dịch mua hàng trả chậm bằng tiện ích của Cửa hàng Play:

### Chuyển ID tài khoản bị xáo trộn vào Google Play

Bạn có thể chuyển ID tài khoản người dùng bị xáo trộn sang Google Play để tạo điều kiện phát hiện lạm dụng, chẳng hạn như phát hiện xem nhiều thiết bị có đang mua hàng trên cùng một tài khoản trong một khoảng thời gian ngắn không.

**Lưu ý:** ID tài khoản này cũng được trả lại với dữ liệu mua hàng. Nếu đang sử dụng tải trọng dành cho nhà phát triển để xác định người dùng, bạn có thể sử dụng ID tài khoản này thay thế.

Để truyền ID tài khoản bị xáo trộn, hãy gọi phương thức SetObfuscatedAccountId() từ API tiện ích mở rộng:

### 

### Chuyển ID hồ sơ bị xáo trộn sang Google Play

Bạn có thể chuyển ID hồ sơ bị xáo trộn cho Google Play để tạo điều kiện phát hiện gian lận, chẳng hạn như phát hiện xem nhiều thiết bị có đang mua hàng trên cùng một tài khoản trong một khoảng thời gian ngắn không. Điều này tương tự [như vượt qua ID tài khoản người dùng bị xáo trộn](https://developer.android.com/google/play/billing/unity#obfuscated-account-ids). Trong cả hai trường hợp, ID đại diện cho một người dùng, nhưng ID hồ sơ cho phép bạn xác định duy nhất một người dùng trên nhiều hồ sơ mà họ có trong một ứng dụng. Sau khi gửi ID hồ sơ bị xáo trộn đến Google Play, bạn có thể truy xuất ID đó sau này trong biên lai mua hàng.

Để vượt qua id hồ sơ bị xáo trộn, hãy sử dụng trình tạo IAP của bạn để sửa đổi cấu hình mô-đun của bạn bằng cách gọi phương thức SetObfuscatedProfileId() :

### Xác nhận thay đổi giá cho gói đăng ký

Google Play cho phép bạn [thay đổi giá của đăng ký đang hoạt động](https://developer.android.com/google/play/billing/billing_subscriptions#change-price-sub). Người dùng trò chơi của bạn phải xác nhận bất kỳ thay đổi giá nào trước khi thay đổi có thể có hiệu lực. Để nhắc người dùng xác nhận thay đổi giá cho đăng ký của họ, hãy gọi phương thức ConfirmSubscriptionPriceChange(:

## Thay đổi hành vi API Unity

Khi bạn đang sử dụng plugin Google Play Billing, hầu hết các API hoạt động giống như cách triển khai IAP tiêu chuẩn của Unity cho các cửa hàng ứng dụng khác. Tuy nhiên, có một số trường hợp API sẽ hoạt động khác nhau. Phần này mô tả những khác biệt về hành vi này.

### Tải trọng dành cho nhà phát triển không được hỗ trợ

Tải trọng dành cho nhà phát triển không được chấp nhận của Google Play và đang thay thế nó bằng các lựa chọn thay thế có ý nghĩa và theo ngữ cảnh hơn. Vì lý do này, tải trọng dành cho nhà phát triển không được hỗ trợ. Để biết thêm thông tin về các lựa chọn thay thế, hãy xem trang giới thiệu về [Tải trọng dành cho nhà phát triển](https://developer.android.com/google/play/billing/developer-payload).

Bạn có thể tiếp tục sử dụng các giao diện tương tự được xác định bởi triển khai IAP tiêu chuẩn của Unity cho các cửa hàng ứng dụng khác, bao gồm IStoreControllor. Khi bạn bắt đầu mua hàng, bạn vẫn có thể sử dụng IStoreControllor và gọi phương thức InitiatePurchase().

Tuy nhiên, mọi tải trọng mà bạn chuyển vào sẽ không có hiệu lực (sẽ không xuất hiện trong biên lai cuối cùng).

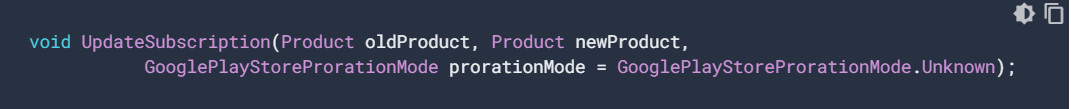
### SubscriptionManager không được hỗ trợ

Unity IAP cung cấp [lớp SubscriptionManager](https://docs.unity3d.com/Manual/UnityIAPSubscriptionProducts.html) để quản lý đăng ký. Vì việc triển khai IAP tiêu chuẩn của Unity trong lớp này sử dụng tải trọng dành cho nhà phát triển, lớp này không được hỗ trợ. Bạn vẫn có thể tạo lớp này, nhưng bạn có thể nhận được dữ liệu không đáng tin cậy khi sử dụng bất kỳ phương thức getter nào của lớp.

### UpdateSubscription có những thay đổi API nhỏ

Plugin Thanh toán Google Play không hỗ trợ sử dụng các phương thức SubscriptionManager.UpdateSubscription() và SubscriptionManager.UpdateSubscriptionInGooglePlayStore() để nâng cấp và hạ cấp đăng ký của bạn. Nếu trò chơi của bạn gọi các phương pháp này, GooglePlayStoreUnsupportedException sẽ được ném.

Thư viện thanh toán cung cấp một API thay thế để sử dụng thay cho các phương thức này. Để nâng cấp hoặc hạ cấp đăng ký, hãy gọi phương thức UpdateSubscription() bằng chế độ proration:

Bạn có thể kết thúc cuộc gọi phương thức này bằng kiểm tra nền tảng hoặc trong một khối bắt khi GooglePlayStoreUnsupportedeException bị bắt.

Để biết thêm thông tin và ví dụ về cách sử dụng chế độ proration, hãy [xem Đặt chế độ proration](https://developer.android.com/google/play/billing/billing_subscriptions#prorate).