Handling plan changes

In this article

Upgrading or downgrading a GitHub Marketplace app triggers the `marketplace_purchase` event webhook with the `changed` action, which kicks off the upgrade or downgrade flow.

> **Note:** This article applies to publishing GitHub Apps in GitHub Marketplace only. For more information about publishing GitHub Actions in GitHub Marketplace, see "Publishing actions in GitHub Marketplace."

For more information about upgrading and downgrading as it relates to billing, see "Using the GitHub Marketplace API in your app."

Step 1. Pricing plan change event 🔗

GitHub send the `marketplace_purchase` webhook with the `changed` action to your app, when a customer makes any of these changes to their GitHub Marketplace order:

- Upgrades to a more expensive pricing plan or downgrades to a lower priced plan.
- Adds or removes seats to their existing plan.
- Changes the billing cycle.

GitHub will send the webhook when the change takes effect. For example, when a customer downgrades a plan, GitHub sends the webhook at the end of the customer's billing cycle. GitHub sends a webhook to your app immediately when a customer upgrades their plan to allow them access to the new service right away. If a customer switches from a monthly to a yearly billing cycle, it's considered an upgrade. See "Billing customers" to learn more about what actions are considered an upgrade or downgrade.

Read the `effective_date` , `marketplace_purchase` , and `previous_marketplace_purchase` from the `marketplace_purchase` webhook to update the plan's start date and make changes to the customer's billing cycle and pricing plan. See "Webhook events for the GitHub Marketplace API" for an example of the `marketplace_purchase` event payload.

If your app offers free trials, you'll receive the `marketplace_purchase` webhook with the `changed` action when the free trial expires. If the customer's free trial expires, upgrade the customer to the paid version of the free-trial plan.

Step 2. Updating customer accounts 🔗

You'll need to update the customer's account information to reflect the billing cycle and pricing plan changes the customer made to their GitHub Marketplace order. Display

upgrades to the pricing plan, `seat_count` (for per-unit pricing plans), and billing cycle on your Marketplace app's website or your app's UI when you receive the `changed` action webhook.

When a customer downgrades a plan, it's recommended to review whether a customer has exceeded their plan limits and engage with them directly in your UI or by reaching out to them by phone or email.

To encourage people to upgrade you can display an upgrade URL in your app's UI. See "About upgrade URLs" for more details.

> **Note:** We recommend performing a periodic synchronization using `GET` `/marketplace_listing/plans/:id/accounts` to ensure your app has the correct plan, billing cycle information, and unit count (for per-unit pricing) for each account.

Failed upgrade payments 🔗

In the case where a customer upgrades their plan and the payment fails, GitHub reverts their GitHub Marketplace subscription to its previous state. GitHub also sends an email to the customer to inform them of the failure and allow them to re-attempt their purchase. You will receive a webhook with the `changed` action requesting you to revert to the previous plan.

About upgrade URLs 🔗

You can redirect users from your app's UI to upgrade on GitHub using an upgrade URL:

```
https://www.github.com/marketplace/<LISTING_NAME>/upgrade/<LISTING_PLAN_NUMBER>/<CUS
```

For example, if you notice that a customer is on a 5 person plan and needs to move to a 10 person plan, you could display a button in your app's UI that says "Here's how to upgrade" or show a banner with a link to the upgrade URL. The upgrade URL takes the customer to your listing plan's upgrade confirmation page.

Use the `LISTING_PLAN_NUMBER` for the plan the customer would like to purchase. When you create new pricing plans they receive a `LISTING_PLAN_NUMBER`, which is unique to each plan across your listing, and a `LISTING_PLAN_ID`, which is unique to each plan in the GitHub Marketplace. You can find these numbers when you List plans, which identifies your listing's pricing plans. Use the `LISTING_PLAN_ID` and the "Apps" endpoint to get the `CUSTOMER_ACCOUNT_ID`.

> **Note:** If your customer upgrades to additional units (such as seats), you can still send them to the appropriate plan for their purchase, but we are unable to support `unit_count` parameters at this time.