Cart API and Session Lifecycle

Punchout Documentation

Guide for Magento administrators, technical project owners, and implementation teams.

LAST UPDATED
2026-09-26

SOURCE
Open this article online

EXTENSION
Magento 2 Extension Punchout

Cart API and Session Lifecycle

After OCI login or cXML setup, the returned token can drive a browser shopping session, the cart API, or both. Every cart call is bound to the Magento quote stored on that PunchOut context.

Open the storefront session

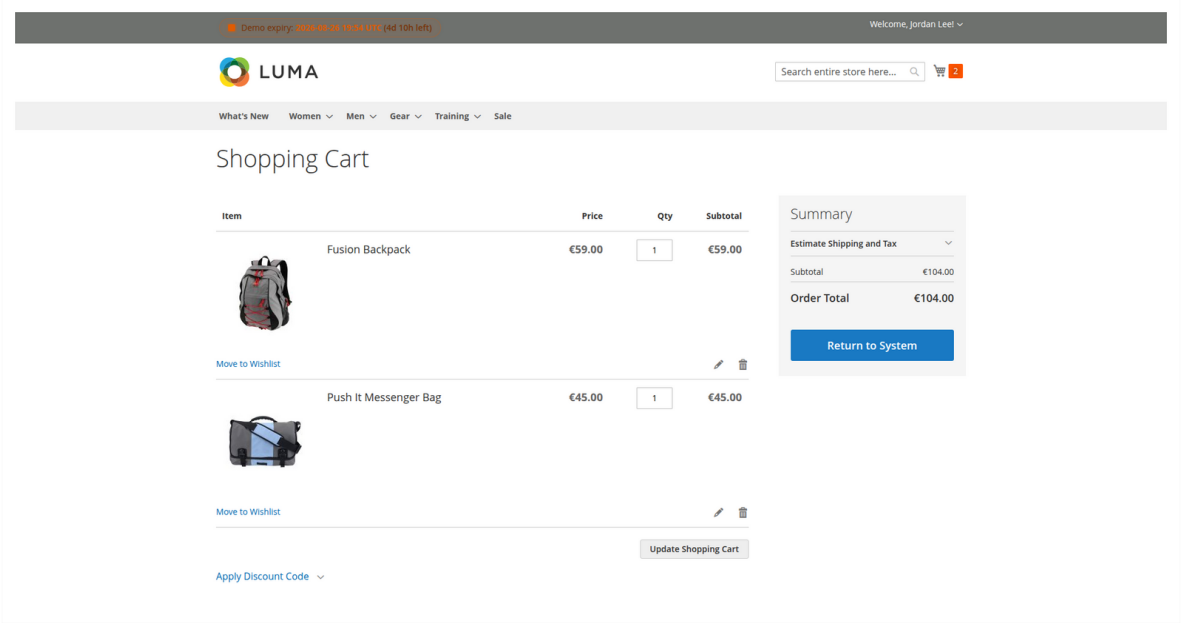
Send the browser to:

```
/punchout/session/activate?token=<token>&target_url=/catalogsearch/result/?q=chair
```

The activation route validates the token, stores it in the Magento session and applies the customer context resolved during setup. Depending on the buyer profile, this can be a guest, a fixed customer, an account resolved from the request, or a customer created through JIT provisioning. `target_url` may be relative or use the same storefront host; an external host is discarded.

PunchOut creates its own active quote. It never reuses the mapped customer's ordinary checkout quote. On every cart load, Magento checks that the quote is active, belongs to the selected Store view and has the same customer association as the context. A quote from another store or customer, or a quote that has already been submitted, is rejected. The context's currency and conversion rate are restored before totals are collected.

During the active session, the cart shows the configured return action instead of the normal checkout action.



Tokenized cart endpoints

Send the token in the request body, query parameters or `X-Punchout-Token` header.

Endpoint	Method	Required business input
`/punchout/cart/add`	POST	`sku`, positive `qty`; or an `items` array
`/punchout/cart/update`	POST	`item_id` or `sku`, plus fields to change; or an `items` array
`/punchout/cart/remove`	POST	`item_id`, `sku`, `item_ids`, `skus` or `items`
`/punchout/cart/get`	GET or POST	token only

Example add request:

```
{
  "token": "<session-token>",
  "items": [
    {"sku": "24-WG085", "qty": 4}
  ]
}
```

Magento remains the pricing authority by default. A `price` value is applied only when **Allow inbound cart item prices** is enabled for the context store. Accepted custom prices remain locked during later total recollection.

A successful `/punchout/cart/get` response uses HTTP 200, sets `success` to `true`, and returns the current quote summary under `cart`. Keep the token and request signature out of screenshots, exported traces and support tickets. A normal storefront cart image does not prove that the tokenized API contract was called.

Return the cart

The storefront action `/punchout/session/back` attaches the active Magento quote to the context and prepares the mapped payload.

- OCI returns an HTML form that posts indexed `NEW_ITEM-*` fields to the hook URL.
- cXML returns an HTML form that posts the generated `PunchOutOrderMessage` as `cXML-urlencoded`.

Direct integrations can call `/punchout/oci/transfer` or `/punchout/cxml/transfer` and receive the generated HTML or XML payload inside a JSON response instead.

Status and expiry

A new context starts in `AUTHENTICATED`. A successful direct transfer changes it to `TRANSFERRED`. Logout changes it to `EXPIRED`; time expiry also prevents further browser or cart API use. Expired contexts are removed by cron every 15 minutes according to the cleanup rules. A transferred context cannot be used again, even when its token has not reached its time limit.

The cart API follows the same operation policy as the storefront. An `inspect` context may be read but cannot update or remove lines. `DETAIL` does not initialise a normal shopping quote. `DETAILADD` accepts only the resolved product and closes the one-item handoff after Add to Cart.

The configured lifetime must cover browsing and cart return. A later, independently authenticated cXML `OrderRequest` can still use its retained BuyerCookie snapshot after browser-session expiry, provided the snapshot has not expired. Configure snapshot retention separately for the approval window.