

CASE STUDY #5

Cart Integrity When Stickers Were Removed or Made Unavailable by Admin

Tadoo -- Checkout Validation and Cart State Consistency

Project	Tadoo -- Tattoo and Sticker Marketplace
Area	Cart Management, Checkout Validation, Admin-User Interaction
Difficulty	Medium-High -- Async availability changes, silent failures, user experience
Outcome	No checkout failures from unavailable items; users informed early with clear recovery path

Background

Tadoo admin panel gave administrators the ability to unpublish individual sticker listings or deactivate entire categories at any time. This was a necessary operational capability. Admins needed to pull designs that were no longer available, temporarily delist items during restocking, or remove content that needed updating. The problem was that none of these admin actions had any awareness of what users currently had in their active carts.

The Problem

Problem 1 -- Silent Checkout Failure for Unavailable Items

When a user added a sticker to their cart and then the admin unpublished that sticker before the user checked out, the cart still showed the item as available. The user would proceed through the checkout flow, select their payment gateway, and submit the order. Only at the final order creation step would the system check availability and reject the transaction. The user saw a generic error with no explanation of which item was the problem or what to do next.

The Worst Case

A user with 6 items in their cart, one of which had been unpublished by an admin, would complete the entire checkout flow including payment gateway authentication, only to hit an error at the final step. No payment was processed, but the error screen gave no indication of which item caused the failure or that their payment method had not been charged.

Problem 2 -- Cart Showed Unavailable Items With No Visual Indication

The cart page fetched item data at load time. If the user had opened their cart before an admin unpublished a sticker and then not refreshed the page, the cart continued to display the item as available with a price and an add to cart button. Nothing in the UI indicated the item status had changed. The cart was showing the user something they could not actually purchase.

Problem 3 -- Category Deactivation Silently Affected Multiple Cart Items

When an admin deactivated an entire category, every sticker in that category became unavailable simultaneously. A user with multiple items from that category in their cart had all of them become unpurchasable at once. The system had no way to notify the user, and the cart continued to display and price all of those items as available until the user attempted checkout and hit the rejection.

The Solution

1. Cart Availability Validation on Cart Page Load

Every time the cart page loaded, the system performed a lightweight availability check against all items currently in the cart. Items that had been unpublished or whose categories had been deactivated were flagged in the response. The cart UI rendered these items with a clear No longer available label, a muted appearance, and a remove button. The checkout button remained disabled as long as any unavailable item was in the cart. The user was told exactly which items were the problem before they wasted any time in the checkout flow.

2. Pre-Checkout Availability Re-Validation

Even with the cart-load check in place, a sticker could be unpublished in the time between the user loading their cart and clicking the checkout button. A second availability check ran at the moment the user initiated checkout, before any gateway was contacted. If any item had become unavailable in that window, the user was returned to the cart page with the problematic items flagged. No gateway interaction happened for an order that could not complete.

3. Soft Unpublish With Cart Grace Period

Rather than making items immediately unavailable the moment an admin clicked unpublish, a soft unpublish system was introduced. When an admin unpublished a sticker, it was flagged as pending removal for a 15-minute grace period. During this window, users who already had the item in their cart could still complete their purchase. New users could not add the item and it did not appear in browse results. After 15 minutes, the item became fully unavailable and the cart validation flagged it for any remaining users who had not yet checked out.

4. Automated User Notification on Cart Item Removal

When the cart page flagged an item as unavailable, logged-in users received an in-app notification informing them that a saved item was no longer available. The notification linked directly to the cart so the user could remove the item and continue. Users who had the item in both their cart and their wishlist had it removed from the cart but kept on the wishlist, since wishlisted items carry no purchase intent and the user might want to know when the item becomes available again.

Problem	Solution
Unavailable items cause silent checkout failure	Pre-checkout re-validation before any gateway contact
Cart shows unavailable items as purchasable	Availability check on every cart page load with visual flagging
Category deactivation affects multiple cart items	Category-level availability propagated to all stickers in cart check
Immediate unpublish disrupts in-progress checkouts	15-minute soft unpublish grace period for cart holders
User not informed when cart item becomes unavailable	In-app notification with direct link to cart on item status change

Outcome

After shipping the cart validation system, checkout failure rates from unavailable items dropped to zero. Users who had items become unavailable were informed at the cart page rather than at the checkout error screen. The soft unpublish grace period meant that in-progress purchases were almost never disrupted by admin catalog changes. Admins gained confidence that they could manage the catalog without breaking active user sessions, and users trusted that what their cart showed was accurate.

Key Takeaway

A cart is a promise to the user that these items can be purchased. When the system cannot keep that promise, it needs to say so as early and as clearly as possible, not at the moment the user is expecting a payment confirmation. Cart validation is not a checkout concern. It is a cart concern. The earlier the system surfaces availability problems, the less damage they do to user trust.

-- Ehsan