bluemint

nShift Checkout for Magento 2

User Guide

Module Name: Bluemint_NshiftCheckout
Version: 1.3.0
Magento Compatibility: Magento 2.4.x
PHP Compatibility: 8.1 – 8.3

Purpose of this Guide

This document explains how to install, configure, and operate the Bluemint nShift Checkout module for Magento 2.

It is intended for:

- Store administrators
- E-commerce managers
- Technical support teams
- Magento agencies and developers

The guide covers:

- Backend configuration
- Frontend checkout behaviour
- Order and shipment lifecycle
- nShift partial shipments
- Book & Print integrations (Delivery / Transsmart)

Table of Contents

1) Overview
2) Key Concepts & Architecture
3) System Requirements
4) Installation
5) Widget Setup (Mandatory)
6) Backend Configuration
    - General Settings
    - API Configuration
    - Book & Print Platforms
    - Book & Print Failure Handling
    - Sender Information
    - Package & Dimensions
    - Widget Configuration
    - Carrier Configuration
7) Customer Checkout Flow
8) Order & Shipment Lifecycle
9) Book & Print Platforms

1. Overview

1.1 What Is nShift Checkout?

nShift Checkout is a delivery selection solution that allows customers to choose from **multiple carriers, delivery services, and pickup points** directly during checkout.

The **Bluemint nShift Checkout module** integrates Magento 2 with the **nShift Checkout API**, enabling:

- Real-time delivery options at checkout
- Embedded nShift delivery widget
- Storage of selected delivery option on quote and order
- Creation of **partial shipments** in nShift
- Automatic **Book & Print** when a Magento shipment is created
- Support for **Delivery (Unifaun)** and **Transsmart** Book and Print system
- Multi-store and B2B support
- Supports Additional Custom Parameters

1.2 High-Level Flow

Cart → Address → nShift Session
↓
Checkout Widget → Option Selection
↓
Order Placed → Partial Shipment (cron)
↓
Magento Shipment → Book & Print → Labels & Tracking

1.3 Key Features

Feature	Description
nShift Checkout API	Session, delivery options, partial shipments
Embedded Widget	Customer selects delivery option & pickup point
Quote & Order Persistence	Selected option stored permanently
Partial Shipment Cron	Background creation of nShift partial shipments
Book & Print	Automatic shipment & labels on shipment creation
Multi-Store	Per store view configuration
B2B Support	Customer group–based B2B handling
Exclusive Mode	Hide all other shipping methods
Additional custom Parameters	Enables us to pass any quote and customer parameters from Magento to nShift which can be used to create rules for showing different prices, e.g. free shipping for orders with a grand total greater than 1000 SEK

2. Key Concepts & Terminology

nShift Session

A temporary checkout session created with nShift that contains:

- Destination address
- Cart value
- Weight and dimensions
- Language and currency

Used to fetch delivery options.

Delivery Option

A specific carrier service returned by nShift (e.g. Home delivery 2–3 days, Pickup at PostNord), including:

- Price
- Estimated delivery time
- Optional pickup points

Partial Shipment

A reservation created in nShift after order placement, confirming which option the customer selected.

⚠️ Book & Print cannot run without a partial shipment.

Book & Print

The final shipment creation step:

- Runs when a Magento shipment is created
- Calls Delivery or Transsmart API
- Generates labels and tracking numbers
- Attaches labels to the Magento shipment

3. System Requirements

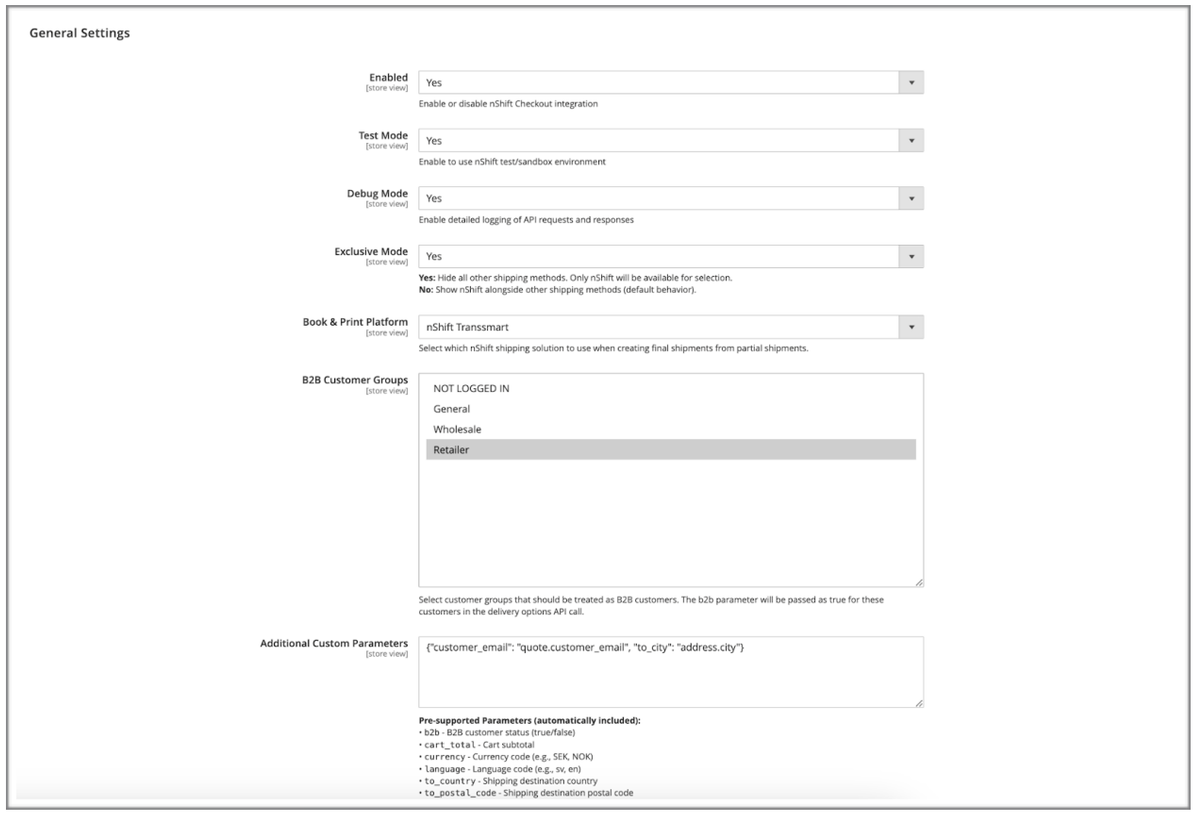
- Magento: 2.4.x
- PHP: 8.1, 8.2, or 8.3
- nShift Account: Active Checkout account
- Credentials Required:
  - Client ID
  - Client Secret
  - Connection ID
- Widget Files: Official nShift Checkout JS & CSS

4. Backend Configuration

All settings are located under:

Stores → Configuration → Bluemint → nShift Checkout

4.1 General Settings

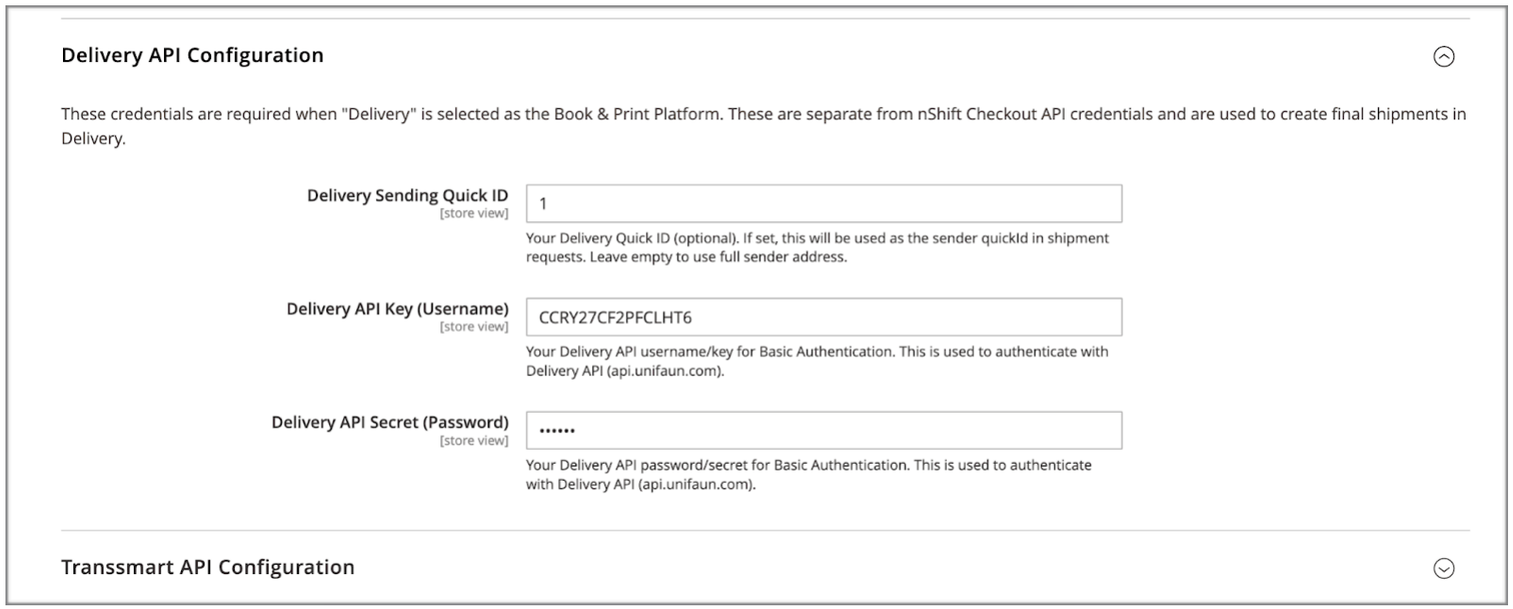


Field	Description
Enabled	Enables or disables the entire module
Test Mode	Officially uses nShift sandbox environment CSS
Debug Mode	Enables detailed logging
Exclusive Mode	Hides all non-nShift shipping methods
Book & Print Platform	Delivery / Transsmart / Ship
B2B Customer Groups	Sends `b2b=true` for selected groups , can be used to make rule for different price for shipping for different customer groups
B2B Customer Groups	Advanced API parameters

4.2 API Configuration

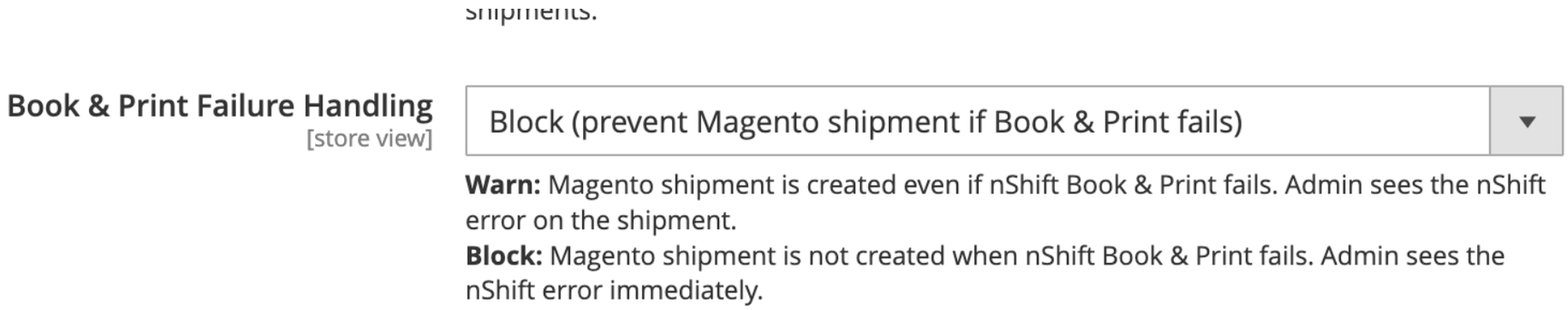
Field	Description
Client ID	nShift API client ID
Client Secret	Encrypted API secret
Connection ID	Identifies carrier setup in nShift
Checkout ID	Optional (only if provided by nShift)
API Timeout	Request timeout in seconds

4.3 Book & Print – Delivery (Unifaun)



Field	Description
API Key	Delivery username
API Secret	Delivery password
Sending Quick ID	Optional sender identifier

4.4 Book & Print Failure Handling



Path: Stores → Configuration → Bluemint → nShift Checkout → General

Default: Warn

When an admin creates a Magento shipment for an nShift order, the module calls nShift Book & Print (Delivery or Transsmart, depending on Book & Print Platform) to book the final shipment and generate labels. This setting controls what happens if that call fails.

Book & Print runs for orders that:

- Use an nShift shipping method (nshiftcheckout_*)
- Have the module enabled
- Have a configured Book & Print handler for that store

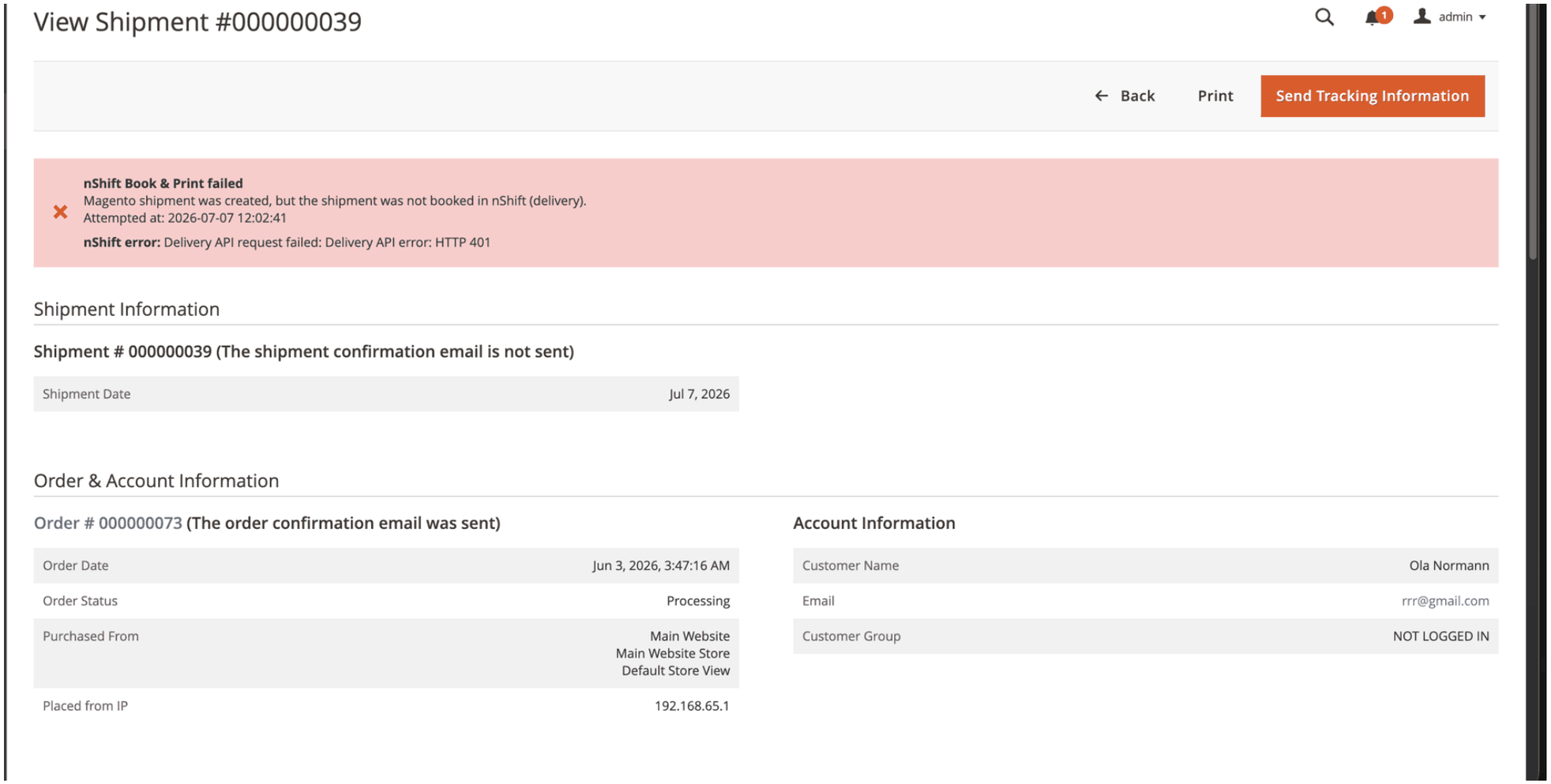
If no nShift partial shipment exists yet, the module tries to create one on demand (same logic as the cron job) before booking. Failures during that step are handled the same way as Book & Print failures.

Warn (default)

Label in admin: Warn (create Magento shipment, show admin error)

Magento shipment creation continues even if Book & Print fails.

What the admin sees an immediate error message at the top of the admin page:

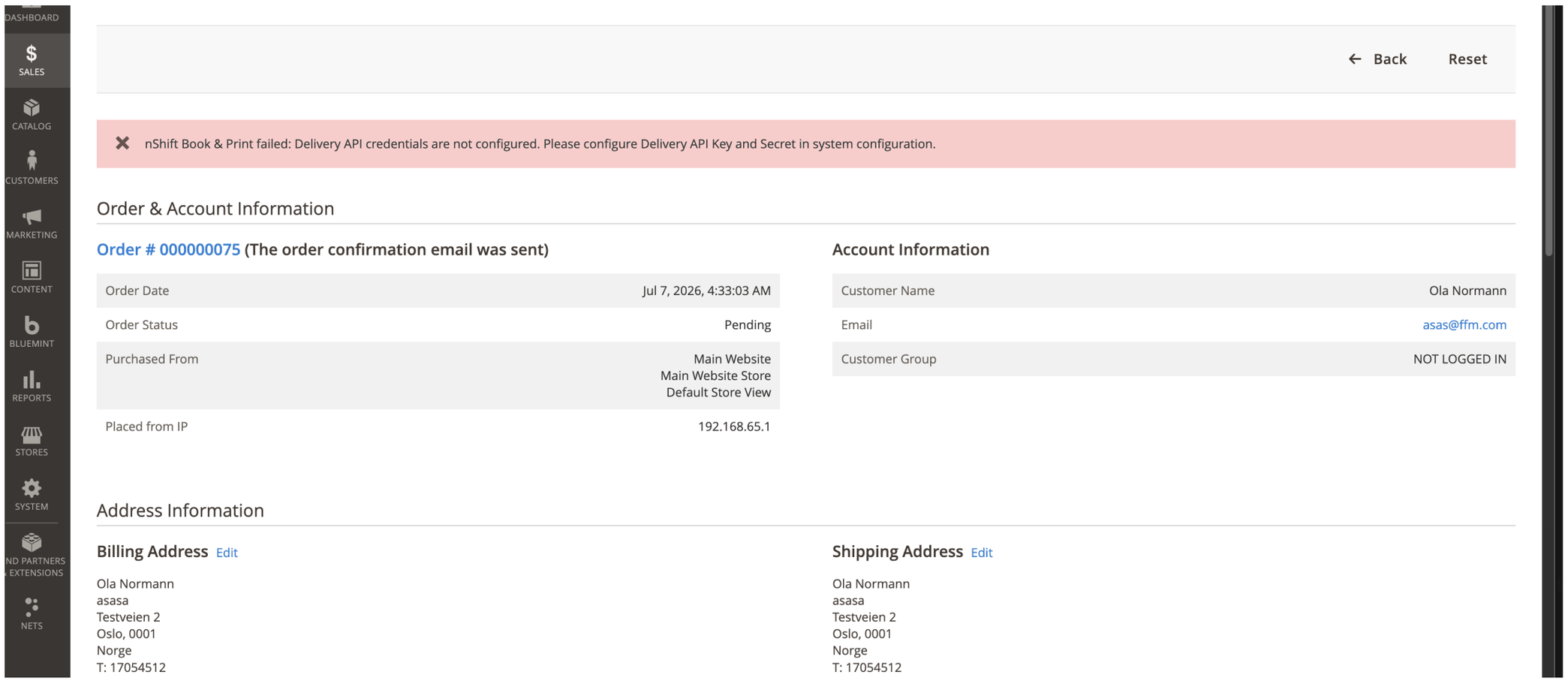


Block

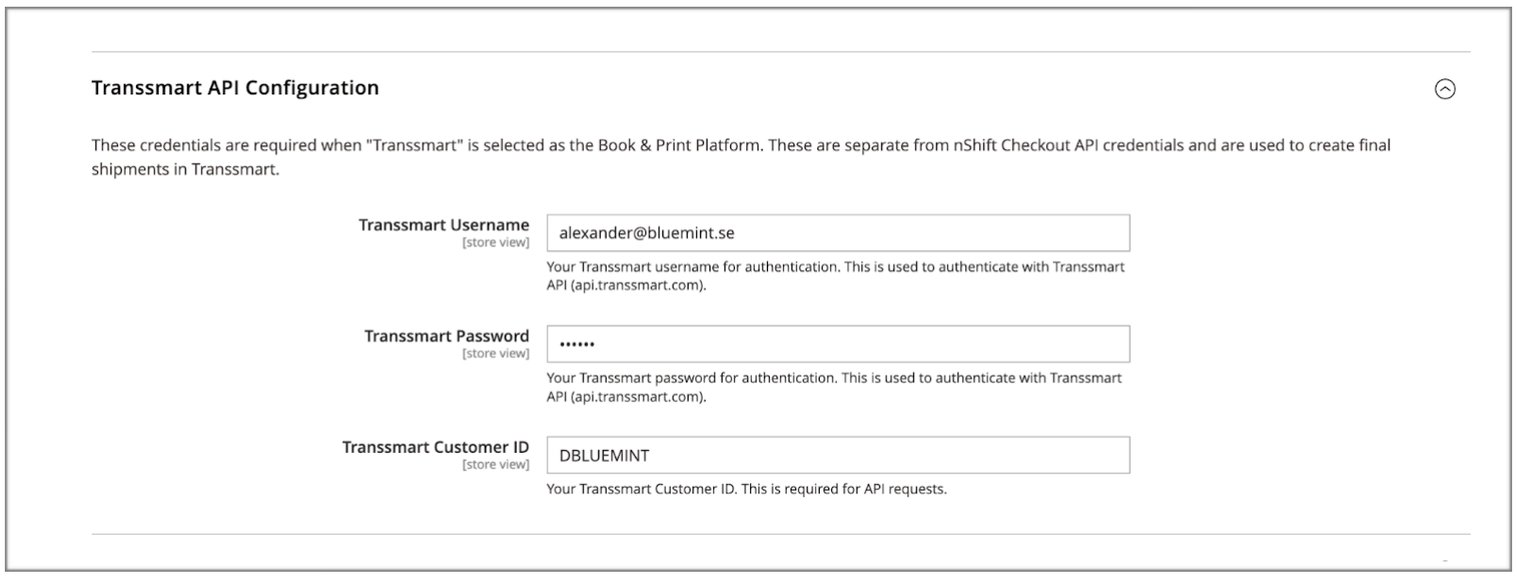
Label in admin: Block (prevent Magento shipment if Book & Print fails)

Magento shipment creation is stopped if Book & Print fails. No shipment record is saved.

What the admin sees Immediate error when clicking Submit Shipment:

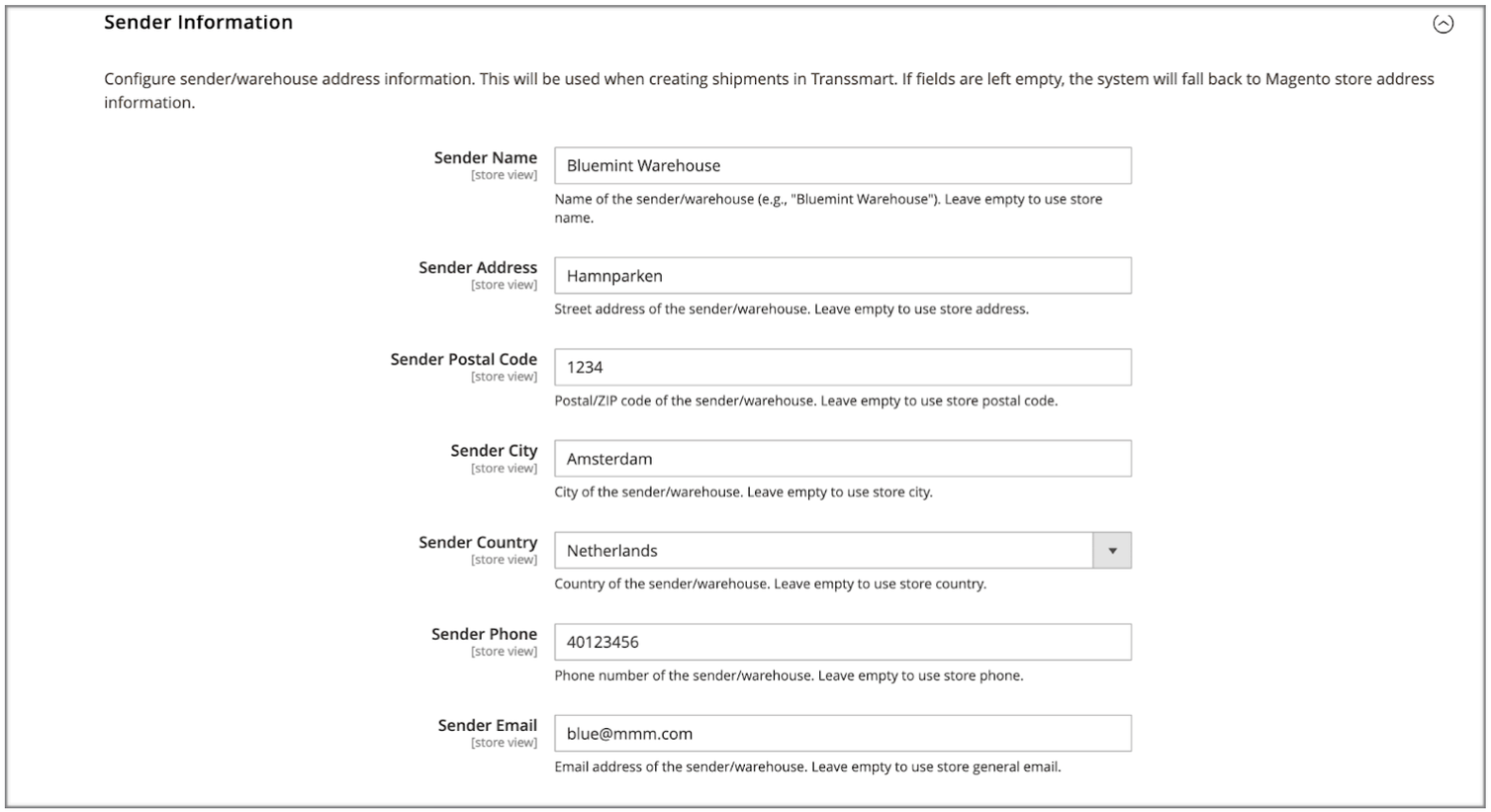


4.5 Book & Print – Transsmart



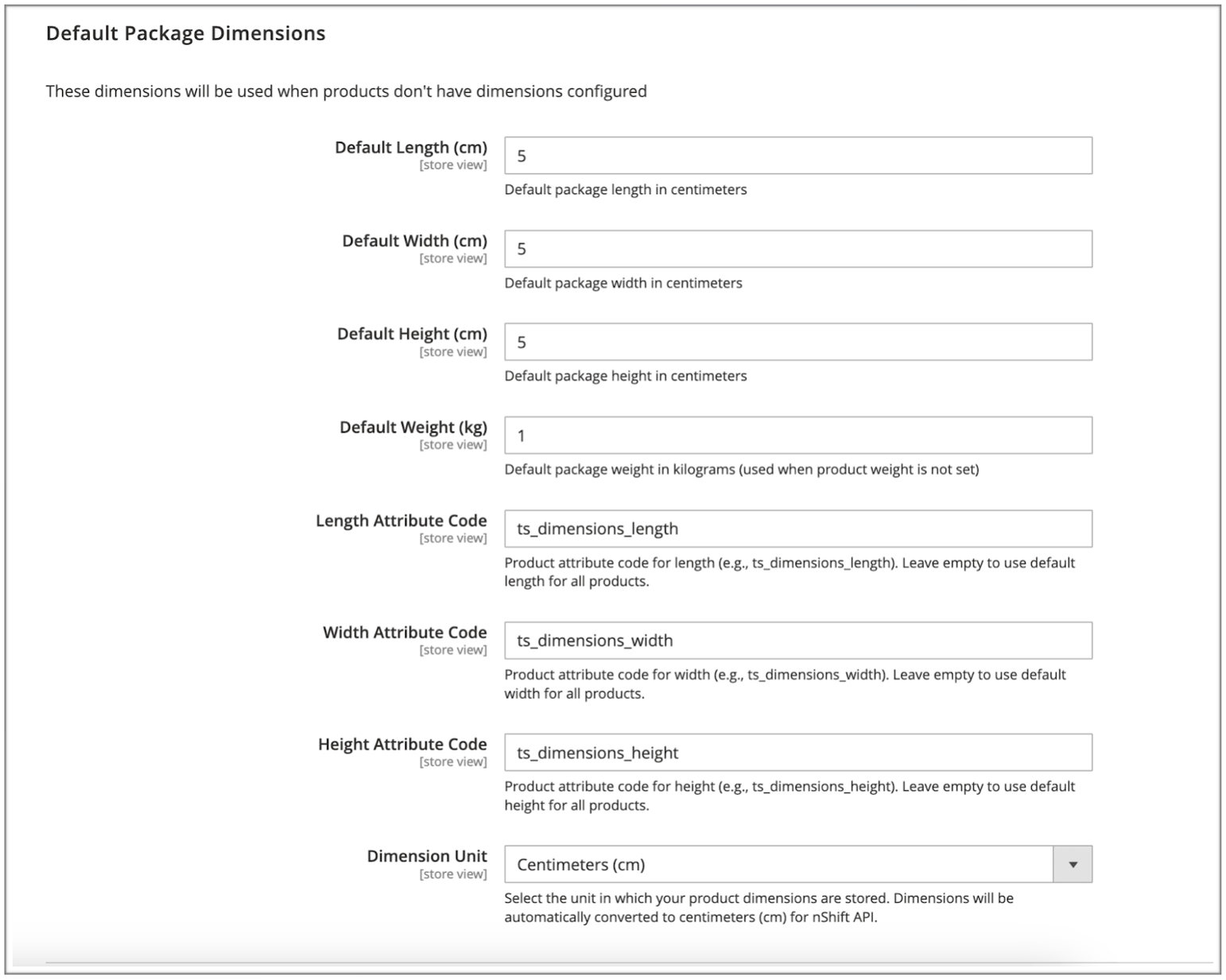
Field	Description
Username	API login
Password	API password
Customer ID	Required Transsmart ID

4.6 Sender Information (Transsmart)



Used as the **ship-from address** if provided; otherwise store address is used.

4.7 Package & Dimensions



nShift needs weight and dimensions to calculate options and create partial shipments. These settings apply when products do not have weight or dimensions.

Default Length, Width, Height (cm) and Default Weight (kg)

What it does: If a product has no dimensions (or no dimension attributes), the module uses these **default values** in cm and kg when building the session and partial shipment payload. This avoids missing or zero values that could cause errors or wrong pricing.

When to use: Set realistic defaults for a typical parcel (e.g. 30×20×15 cm, 1 kg). Use product attributes where possible and defaults as fallback.

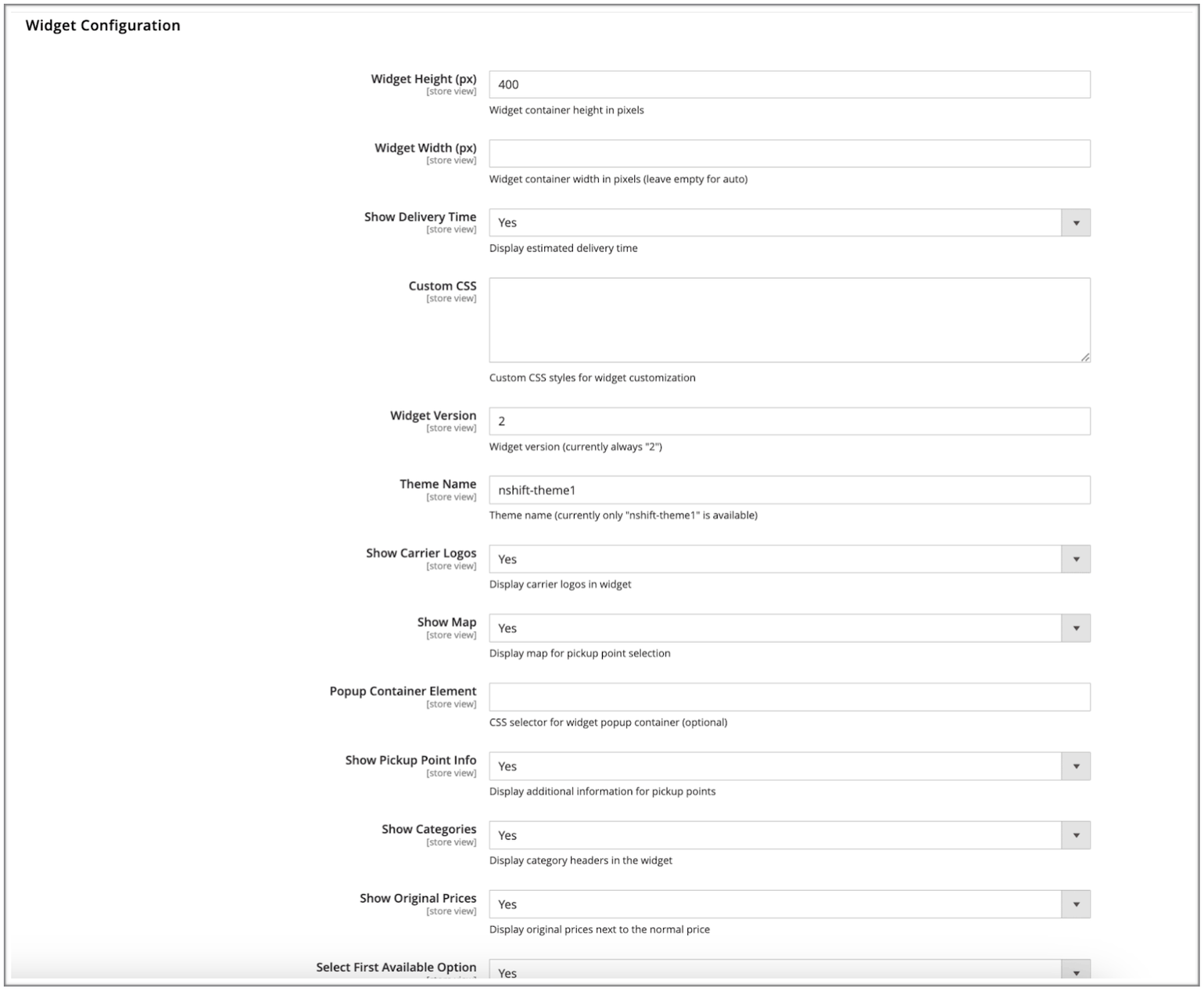
Length/Width/Height Attribute Code

- What it does: If your products have custom attributes for length, width, and height (e.g. `ts_dimensions_length`), enter the attribute codes here. The module will read them from the product and use them (after unit conversion) instead of default dimensions.
- When to use: Use when you maintain dimensions per product. Leave all three empty to use default dimensions for every product.

Dimension Unit

- What it does: Defines the unit in which product dimensions are stored (e.g. cm, mm, inches). The module converts values to centimetres before sending to nShift.
- When to use: Set to the unit you use in your catalog. Wrong unit gives wrong dimensions and possibly wrong carrier options.

4.8 Widget Configuration



These options control how the nShift checkout widget looks and behaves on the shipping step. They are passed to the widget when it is initialised via the nShift widget API. Settings are grouped into: container styling (height, width, custom CSS), widget API settings (version, theme name, themeOverride options), and responsive breakpoints.

Widget Height and Widget Width (px)

What it does: Height sets the container element's height in pixels so the delivery options list has enough vertical space. If not set, the container uses a default minimum height (200px). Width can be set for a fixed container width, or left empty for "auto" (the container then adapts to the column width of your checkout layout).

When to use: Set height if options are cut off or if there's too much empty space. Typical values: 400–600px for desktop, 300–400px for mobile. Set width only if you need a fixed width for layout consistency; otherwise leave empty for responsive behavior.

Technical note: These are applied to the container element via inline CSS (`style.height`, `style.width`). They are not part of the nShift widget API; they control the Magento container that wraps the widget.

Show Delivery Time

What it does: When Yes, the widget displays the estimated delivery date or time range for each delivery option (e.g. "2–3 business days", "Delivery by 15:00"). This setting is passed to the widget as `themeOverride.showDeliveryTime`.

When to use: Usually set to Yes so customers see delivery timing. Set to No only if you want a minimal widget without delivery time information.

Custom CSS

What it does: Allows you to add your own CSS rules to style the widget container or override nShift's default styles. The CSS is injected into the page `<head>` as a `<style>` tag with ID `nshift-custom-css`. You can target the widget container (`#nshift-widget-container`) or any nShift widget classes.

Example:

css

```
#nshift-widget-container {
   border: 1px solid #e0e0e0;
   border-radius: 4px;
   padding: 15px;
}
.nshift-checkout-widget {
   font-family: 'Your Font', sans-serif;
}
```

Technical note: This CSS is not part of the nShift widget API; it's injected separately for your own styling.

Widget Version

What it does: The nShift widget version (currently "2"). This is passed to `nShiftCheckoutWidget.createWidget()` as `widgetVersion`. The widget version determines which API and features are available.

When to use: Set to "2" (the current version). Do not change unless nShift releases a new widget version and you upgrade. The widget file you download from nShift Portal must match this version.

Theme Name

What it does: The nShift theme identifier (e.g. "nshift-theme1"). This is passed to the widget as `theme`. The theme controls the widget's visual style, layout, and available customization options.

When to use: Set according to the nShift theme you are using (check nShift documentation or Portal). Currently "nshift-theme1" is the standard theme. Do not change unless you switch to a different nShift theme.

Show Carrier Logos

What it does: When Yes, the widget displays carrier logos next to each delivery option (e.g. PostNord logo, DHL logo). This makes it easier for customers to identify carriers visually. Passed to widget as `themeOverride.showLogos`.

When to use: Usually Yes for better visual recognition. Set to No for a text-only, minimal widget.

Show Map

What it does: When Yes, the widget displays an interactive map that customers can use to select a pickup point by location. The map shows pickup points (parcel shops, lockers) and lets customers click to choose one. Passed to widget as `themeOverride.showMap`.

When to use: Enable if you offer pickup points and want customers to choose by location on a map. Disable if you only offer home delivery or if you want a simpler interface without a map.

Popup Container Element

What it does: Optional CSS selector for the widget's popup container. Some nShift widget features (e.g. map popup, pickup point selection popup) may need to render in a specific container element. If set, the widget uses this element; otherwise it uses a default (e.g. `#opc-shipping_method`).

When to use: Leave empty unless your theme layout or nShift documentation specifically requires a different popup container. Most installations can leave this empty.

Show Pickup Point Info

What it does: When Yes, the widget shows additional information for pickup points when they are selected or expanded (e.g. address, opening hours, phone number, directions). Passed to widget as `themeOverride.showPickupPointInfo`.

When to use: Enable if you want customers to see detailed pickup point information. Disable for a simpler display with just pickup point names.

Show Categories

What it does: When Yes, the widget displays category headers in the delivery options list (e.g. "Home Delivery", "Pickup Points", "Express"). This groups options by type. Passed to widget as `themeOverride.showCategories`.

When to use: Enable for better organization when you have many options across different categories. Disable for a flat list without grouping.

Show Original Prices

What it does: When Yes, the widget can display original/comparison prices next to the normal price if your nShift setup provides them (e.g. a strikethrough original price next to a discounted price). Passed to widget as `themeOverride.showOriginalPrices`.

When to use: Enable if your nShift connection provides original prices and you want to show price comparisons. Disable if you don't use original prices or want a simpler price display.

Select First Available Option

What it does: When Yes, the widget automatically selects the first available delivery option when the list loads. The customer can still change the selection, but this avoids the "no option selected" state that might block checkout progress.

When to use: Enable to improve UX and avoid customers forgetting to select an option. Disable if you prefer customers to explicitly choose (e.g. for legal/compliance reasons).

Number of Always Visible Text Lines

What it does: Controls how many lines of text (1–4) are visible in the widget before any option is selected. This affects the initial layout and "above the fold" content. Higher values show more placeholder text; lower values show less. Passed to widget as `themeOverride.numberOfAlwaysVisibleTextLines`.

When to use: Adjust for best appearance on desktop and mobile. Default is 1. Increase if you want more initial content visible; decrease for a more compact initial state.

Number of Initially Visible Options

What it does: Controls how many delivery options (e.g. 1, 2, 3) are shown initially before the customer expands the list to see more. Options beyond this number are hidden behind a "Show more" or expand button. Passed to widget as `themeOverride.numberOfInitiallyVisibleOptions`.

When to use: Set to 1 for a compact initial view (customer sees one option and expands for more). Set to 2–3 if you want to show multiple options immediately. Higher values (4+) show most or all options without expansion.

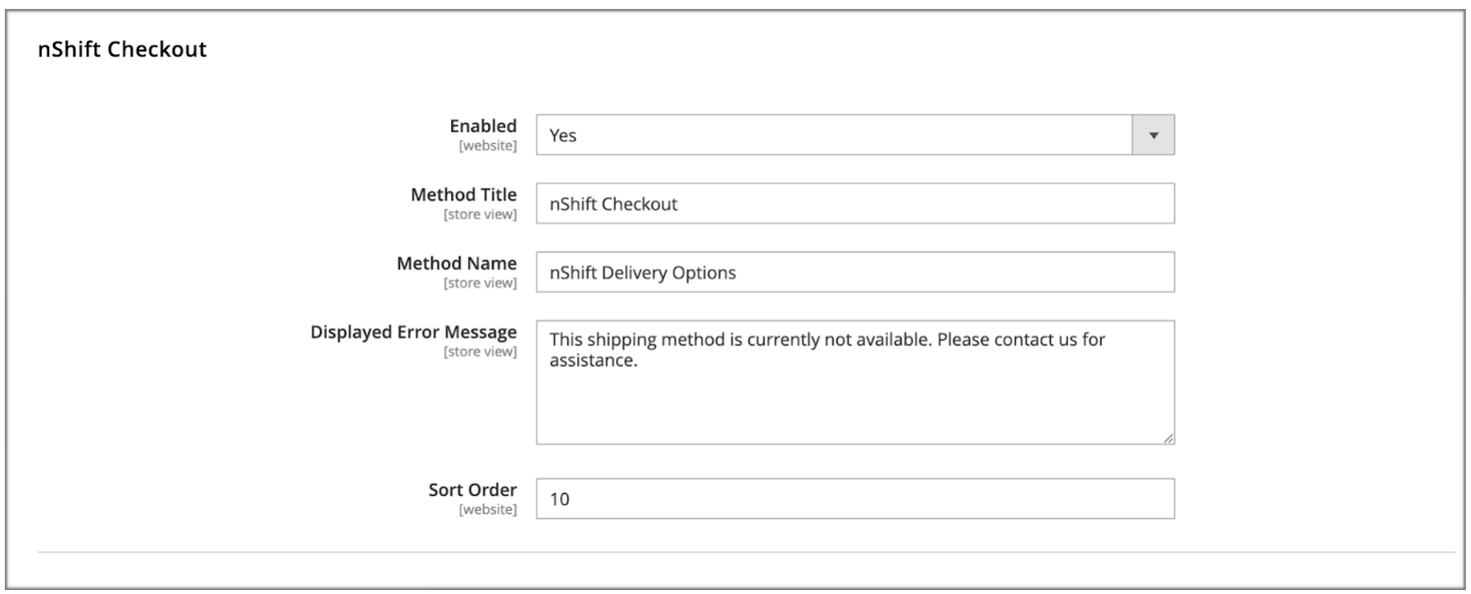
Width Breakpoint – Narrow and Ultra Narrow (px)

What it does: Defines responsive breakpoints for when the widget switches to narrower layout modes. Narrow (default: 400px) and Ultra Narrow (default: 300px) are pixel widths below which the widget adapts its layout (e.g. stacks elements vertically, reduces padding, hides some details). Passed to widget as `widthBreakpoints.narrow` and `widthBreakpoints.ultra-narrow`.

When to use: Adjust if your checkout column is narrower or wider than typical. Defaults (400px and 300px) work for most themes. Set narrower breakpoints if your column is very narrow; set wider if you have a wide checkout column.

Summary: All widget settings are passed to the nShift widget API when it's created, except for Height, Width, and Custom CSS which are applied to the container element for your own styling. The widget uses these settings to control its appearance, behavior, and responsive layout.

4.9 Shipping Method Configuration



Path: Stores > Configuration > Sales > Shipping Methods > nShift Checkout.

This is the standard Magento carrier block. It controls whether the nShift method is active and how it appears in the list of shipping methods.

Enabled — What it does: Enables or disables the nShift Checkout carrier in the shipping methods list. When No, nShift does not appear at checkout even if the module is enabled under Bluemint > nShift Checkout. When to use: Set Yes when you want to offer nShift; No to hide it for a store view without disabling the whole module.

Method Title — What it does: The main label the customer sees (e.g. “nShift Delivery Options” or “Choose delivery”). The actual option title and price (e.g. “Home delivery 2–3 days – 49 SEK”) come from the selected nShift option and are stored on the quote/order.

Method Name — Internal name; can be shown in some themes or emails. Can match the title or be shorter.

Displayed Error Message — What it does: If the nShift method fails (e.g. session could not be created, API error), this message can be shown instead of a generic error. When to use: Enter a short, clear message (e.g. “Delivery options are temporarily unavailable. Please try again or contact us.”).

Sort Order — What it does: Numeric order among shipping methods; lower numbers appear first. When to use: Set so nShift appears where you want (e.g. 10 for nShift, 20 for flat rate, 30 for pickup).

After changing any configuration, click Save Config and flush cache if you use full-page or config cache.

7. Customer Checkout Flow

This section explains in detail what happens from cart to shipment, so you can support customers and troubleshoot.

7.1 Cart and shipping step

The customer adds products to the cart and goes to checkout. On the shipping step, they must enter or select a shipping address (country, postcode, street, city, etc.). The address is required because nShift uses it to calculate which carriers and options are available and at what price. If the address is invalid or not supported by your nShift connection, the widget may show no options or an error.

7.2 Choosing the nShift shipping method

The list of shipping methods shows your configured carriers (e.g. Flat Rate, Free Shipping, nShift Checkout). When the customer selects nShift Checkout (or whatever you set as Method Title):

1. Session creation — The store calls the nShift API to create a session for this quote and address. The request includes: currency, language, total weight and volume (from cart and default dimensions), total price, receiver (name, address, postcode, country, phone, email), and packages (items with dimensions/weight). nShift returns a session ID and optionally a checkout configuration ID. The session is stored in the database and linked to the quote.

2. Widget loads — The checkout widget (the nShift JavaScript you installed) loads and is initialised with this session ID and your Widget Configuration (theme, language, height, show pickup points, etc.). The widget then requests delivery options from nShift (or uses the session to show them). nShift returns a list of options: carrier name, service name, price, estimated delivery time, and optionally pickup points and a map.

3. Customer sees options — The customer sees the list (e.g. “Home delivery 2–3 days – 49 SEK”, “Pickup at PostNord – 29 SEK”) and can expand pickup points or the map if enabled.

4. Any method configured in nShift portal will be visible in the iframe directly delivered as configured.

7.3 Selecting an option and placing the order

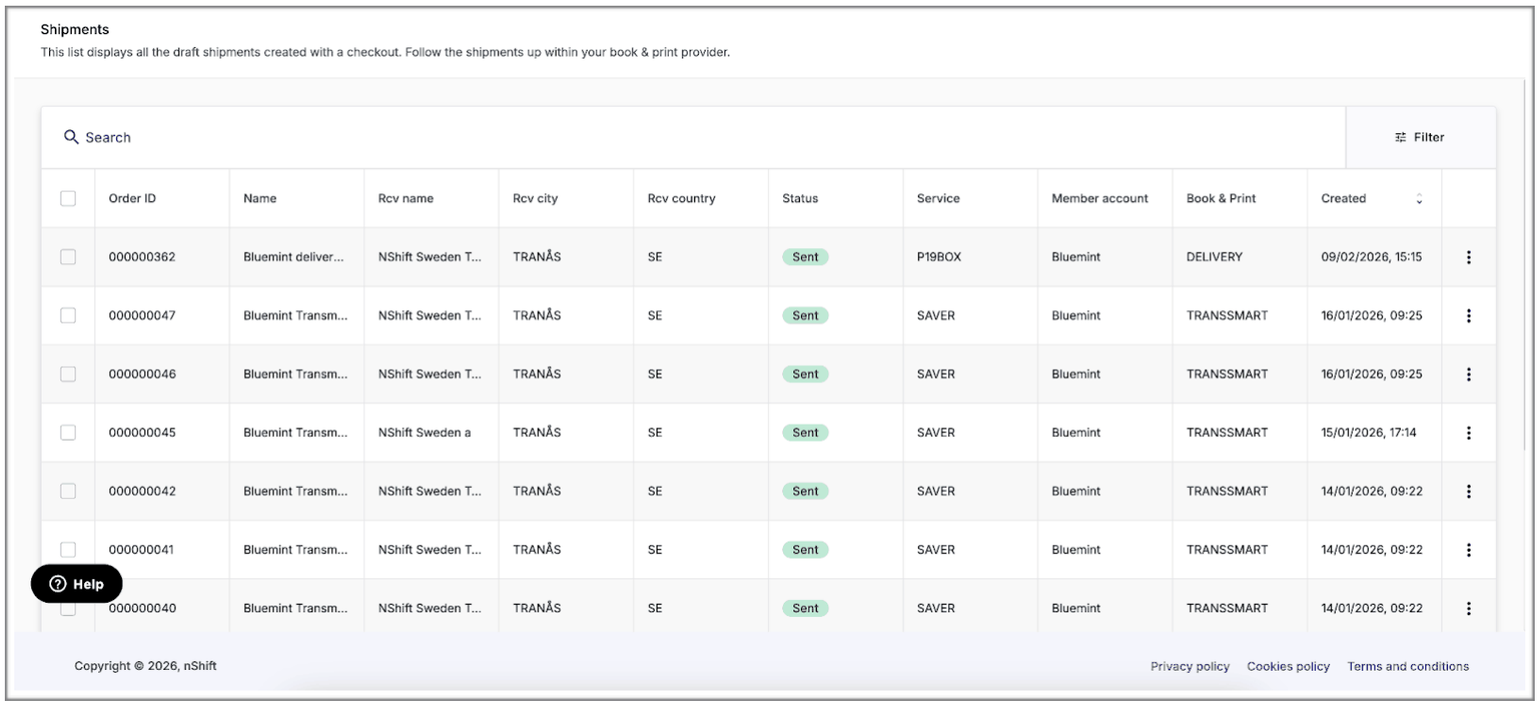
When the customer clicks one option (and optionally selects a pickup point):

- The widget sends the selected option back to Magento: delivery title, price, option ID, carrier product ID, carrier ID, pickup point ID/name, and the full result object (JSON). This is sent in the shipping address’s extension attributes when the customer proceeds (e.g. when they go to the next step or save shipping information).

- A Magento plugin reads these extension attributes and saves them on the quote (e.g. `nshift_delivery_title`, `nshift_delivery_price`, `nshift_option_id`, `nshift_result`). The quote now “remembers” the selection. The shipping method title and price shown in the order summary come from this stored data.

- When the customer clicks Place Order, the order is created. An observer runs and copies all nShift fields from the quote to the order. So the order stores: delivery title, price, option ID, carrier product ID, carrier ID, pickup point ID/name, and the full `nshift_result` JSON. The order’s shipping description and price will show the chosen option. The partial shipment is not created at this moment by default; it is created by the cron job (see below) or by your custom flow if you have one.

7.4 Partial shipment creation (background)



After the order is placed, nShift must be told “this order chose this option” so it can create a partial shipment. The partial shipment is a record in nShift that links the order to the chosen service; your Book & Print platform will use it later to create the real shipment and labels.

- The module includes a cron job named `bluemint_nshift_create_partial_shipments` that runs every minute. It finds orders that: use nShift as shipping method, have `nshift_result` (the full widget result) saved, and do not yet have a partial shipment ID. For each such order (up to 50 per run), it calls the nShift API to create the partial shipment and then saves the returned partial shipment ID on the order.

- So within a short time after placement, the order will have `nshift_partial_shipment_id` set. Until this is set, Book & Print cannot run when you create a Magento shipment (the module will log a warning and skip).

7.5 Creating the shipment and Book & Print

When you (or your process) create a shipment in Magento (Admin: open order → Ship → Submit Shipment, or via API):

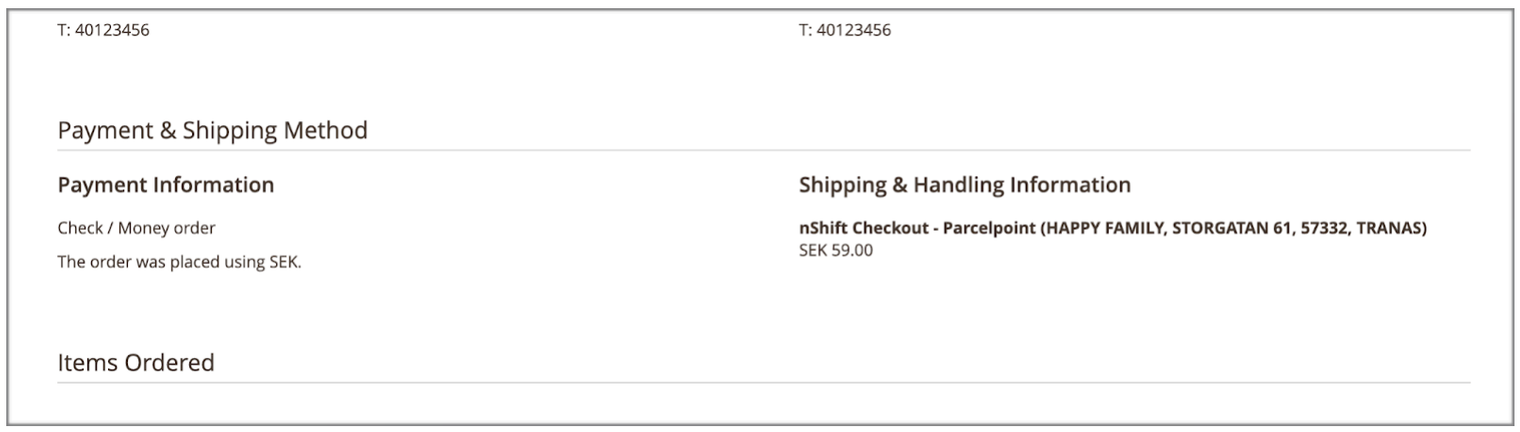
- Magento creates the shipment record and fires the event “shipment saved”.

- The module’s observer runs. It checks: Is the module enabled? Is this order’s shipping method nShift? Does the order have a partial shipment ID? Is a Book & Print platform configured? If all yes, it calls the corresponding handler (Delivery, Ship, or Transsmart).

- The handler uses the order data (address, items, weight), the chosen service (from `nshift_carrier_product_id` / option ID), and the partial shipment ID to call the platform’s API. The platform creates the final shipment and returns label PDFs and tracking number. The module attaches the label(s) to the Magento shipment, adds the tracking number as a track, and saves any platform-specific data. If the API call fails (e.g. wrong credentials), the Magento shipment is still created; the error is logged so you can fix configuration.

8. Order & Shipment Handling

8.1 What Is Stored on the Order (in detail)



Every nShift order stores the following. Understanding these helps when debugging or integrating with other systems.

- nshift_delivery_title — The human-readable title of the option the customer chose (e.g. “Home delivery – 2–3 days”, “Pickup at PostNord”). Shown in the order’s shipping description and in emails. Comes from the widget selection.

- nshift_delivery_price — The price of the chosen option (e.g. 49.00). This is the amount charged for shipping for this order. Stored on the order so it never changes even if nShift prices change later.

- nshift_option_id — The nShift option identifier for the chosen service. Required when creating the partial shipment and when Book & Print talks to the platform (e.g. Delivery uses it to know which service to book).

- nshift_carrier_product_id — The carrier product or service ID (e.g. a Delivery API service code like “P19”). Used by Book & Print (especially Delivery) to create the shipment with the correct product. If missing, some implementations fall back to a default (e.g. “FREE”).

- nshift_carrier_id — The nShift carrier ID. Useful for reporting or for platforms that need it.

- nshift_pickup_point_id and nshift_pickup_point_name — If the customer chose a pickup point (parcel shop), these store its ID and display name. Used when creating the partial shipment and when creating the final shipment (e.g. Delivery can set the “agent” to this pickup point).

- nshift_result — The full JSON result returned by the widget when the customer selected an option. It contains sessionId, optionId, and all other data nShift sent. Required for creating the partial shipment (the cron and any manual flow read sessionId and option data from here). Do not delete or truncate this field.

- nshift_partial_shipment_id — Set after the module (or cron) has successfully called the nShift API to create the partial shipment. This ID is the link between the order and nShift’s record of “this order chose this option”. Required for Book & Print: when you create a Magento shipment, the observer sends this ID to your Book & Print platform so it can create the final shipment and labels. If this is empty, Book & Print will not run (and will log a warning).

8.2 Creating a Shipment (step by step)

1. In Admin go to Sales > Orders and open the order.
2. Click Ship (or “Create Shipment”).
3. In the shipment form you can adjust which items to ship and quantities. You can also add a tracking number manually if needed (e.g. if Book & Print did not return one). For nShift orders, the tracking number and label are usually added automatically by Book & Print after you submit.
4. Click Submit Shipment.

What happens in the background:

- Magento saves the shipment. The observer listens for “shipment saved”. It checks that the order used nShift and has `nshift_partial_shipment_id` set. It then calls the handler for your configured Book & Print platform (Delivery, Ship, or Transsmart) with: the order, the Magento shipment, and the partial shipment ID.
- Delivery: The handler builds a shipment request (sender, receiver, parcels, service ID from `nshift_carrier_product_id`, optional pickup point) and calls the Delivery API. It then fetches the label PDF(s) and attaches them to the Magento shipment, adds the tracking number as a track, and saves Delivery shipment/print IDs on the shipment. You can then print labels from the shipment in Admin.
- Transsmart: Same idea using Transsmart credentials and sender information from config.
- Ship: The module currently has a placeholder; no real API calls. Labels are not created until the Ship integration is implemented.

If Book & Print fails (e.g. wrong API credentials, invalid address, or API error), the Magento shipment is still created so your workflow is not blocked. The error is logged (check your nShift log and Debug Mode). Fix the configuration or address and, if needed, create a new shipment or use a manual process to get labels.

8.3 Cron: Partial Shipments

- Job name: `bluemint_nshift_create_partial_shipments`
- Schedule: Every minute (`* * * * *` in crontab).
- What it does:

1. If the module is disabled, it exits.
2. It loads a collection of orders that: have `shipping_method` like `nshiftcheckout%`, have `nshift_result` not empty, and have `nshift_partial_shipment_id` empty (or null). It limits to 50 orders per run.
3. For each order it calls the ShipmentService to create the partial shipment: it parses `nshift_result` to get sessionId and option data, builds the payload (orderId, sessionId, optionId, receiver, packages, etc.), and POSTs to the nShift shipments API. On success it saves the returned partial shipment ID on the order.
4. If one order fails (e.g. API error), it logs the error and continues with the next order.

When to check: If orders never get a partial shipment ID, ensure Magento cron is running (e.g. Stores > Configuration > Advanced > System > Cron, or your server's crontab running `bin/magento cron:run`). Enable Debug Mode and check the nShift log for "Creating partial shipment" and any error messages. Ensure `nshift_result` on the order contains a valid sessionId and that the nShift API is reachable from your server.

9. Book & Print Platforms

The Book & Print Platform setting determines which external system creates the final shipment and labels when you click “Ship” in Magento. Each platform has its own API and credentials; the module only calls the one you select.

9.1 Delivery (Unifaun)

What it is: The Delivery (Unifaun) API is used to create shipments and retrieve shipping labels (e.g. PDF). Many Nordic carriers are integrated with Delivery. The module sends the order’s receiver address, weight, chosen service (from `nshift_carrier_product_id`), and optional pickup point (agent). Delivery returns shipment ID, tracking number (shipmentNo), and print IDs; the module then fetches each label PDF and attaches it to the Magento shipment.

Required configuration: In Delivery API Configuration (visible when Book & Print Platform = Delivery) you must set Delivery API Key (Username) and Delivery API Secret (Password).

These are your Basic Auth credentials for the Delivery API (e.g. api.unifaun.com). Get them from your Delivery/Unifaun account. Delivery Sending Quick ID is optional: if you have a pre-configured sender “Quick ID” in Delivery, set it here so the module uses it as the sender; otherwise the module may use a default or full address.

Typical flow: When you create a Magento shipment, the module calls Delivery’s prepared-shipments endpoint (using order increment ID as prepareId), then fetches label PDFs from the prints endpoint and attaches them to the shipment. The tracking number is added as a track on the shipment.

9.2 Ship (nShift Ship)

What it is: nShift Ship is another platform that can create shipments and labels. The module has a placeholder implementation: when you select Ship as Book & Print Platform, the observer will call the Ship handler, but the current code does not perform real API calls. No labels or tracking are created until the Ship integration is implemented by your developer.
Required configuration: None for now. When Ship is implemented, a “Ship API” configuration section may be added (similar to Delivery and Transsmart).

9.3 Transsmart

What it is: The Transsmart API is used to create shipments and labels for carriers connected to Transsmart. The module sends order and shipment data (receiver, sender, service, optional pickup point) and receives labels and tracking. Sender information can come from the Sender Information configuration block or from the Magento store address.

Required configuration: In Transsmart API Configuration (visible when Book & Print Platform = Transsmart) you must set Transsmart Username, Transsmart Password, and Transsmart Customer ID. Get them from your Transsmart account. In Sender Information you can optionally set the sender/warehouse name, address, postal code, city, country, phone, and email. If left empty, the module falls back to the store address. Fill these if your shipments are sent from a specific warehouse or if Transsmart requires a complete sender.

Typical flow: When you create a Magento shipment, the module calls the Transsmart API with the order data and partial shipment ID, creates the shipment in Transsmart, and attaches returned labels and tracking to the Magento shipment.

Only the configuration section for the selected platform is shown in Stores > Configuration > Bluemint > nShift Checkout. Switch Book & Print Platform to Delivery, Ship, or Transsmart to see and fill the relevant credential and sender fields.

Default Package Code

This setting defines the package type code sent to nShift Delivery or Transsmart on every parcel during Book & Print (e.g. BX for Box).

Some carriers require it , Kaukokiito is a common example. Without a valid code, Book & Print can fail with errors like “Mandatory value missing: Parcel[0].packageCode”.

How to choose a value

- Pick one code that works across all carriers you use. nShift typically recommends a generic value such as BX (Box) or PK (Package).
- To look up valid codes: log in to app.unifaun.com → Help → Code Lists → Package Codes - [Your Carrier] → Show. Enter the code ID exactly as shown (e.g. BX, PK, EP, KLL).
- Leave empty if your carriers do not require a package code. When empty, the field is omitted from the API request so carriers that do not need it are unaffected.

Validate button

Click Validate to check the entered code against the live Delivery API partner list. The result appears inline. The partners list is cached for 24 hours.

Note: This applies to both platforms, Default Package Code (Delivery) and Default Package Type (Transsmart). Configure only the section that matches your Book & Print Platform setting.

bluemint

Thank you