

KiwiCommerce | Magento 2 Search

User Guide & Configuration Documentation

This guide documents the functionality present in the supplied Kiwi Search extension package. It intentionally covers only implemented functionality and does not include future/premium roadmap items.

1. Overview

- Custom Magento 2 product search opened from the storefront search field.
- AJAX-based search requests are sent to the Kiwi Search controller and results are rendered without a full page reload.
- Searches product name and SKU.
- Displays product name, SKU, price, product image and product URL.
- Supports category, color and price range filtering.
- Category filters show matching product counts.
- Price filter provides the minimum and maximum price available for the current search.
- The extension maintains a dedicated `kiwi_search_product` table populated from CSV data.
- A scheduled CSV generation/import flow keeps the search table synchronized.

2. Installation

- Copy the module to `app/code/Kiwi/Search`.
- Enable the module: `php bin/magento module:enable Kiwi_Search`
- Run: `php bin/magento setup:upgrade`
- Run: `php bin/magento setup:di:compile`
- Run: `php bin/magento setup:static-content:deploy`
- Run: `php bin/magento cache:flush`

3. Admin Configuration

Navigate to Stores → Configuration → Catalog → Kiwi Search.

General Configuration fields

Setting	Purpose
Enable Kiwi Search	Enable or disable the storefront search extension.
CSV File Path	Relative path from Magento root. Default: <code>var/import/kiwi_search_products.csv</code>
Minimum Characters	Minimum number of characters required before keyword results are requested.
Result Limit	Maximum number of result items returned/displayed.
CSV Sync Time	24-hour time used for the configured CSV synchronization schedule, e.g. 05:00 or 14:30.

Configuration Scope

- The configuration fields are available at Default Config, Website and Store View scope.
- Enablement, CSV path, minimum characters, result limit and sync time are read from Magento configuration.
- After changing configuration, flush Magento cache before validating the storefront.

4. Storefront Search

The extension attaches to the Magento storefront search input. When the customer focuses the search field, the custom Kiwi Search panel opens.

AJAX request

GET /kiwi/search/index/?q=vitamin

- The JavaScript uses the browser Fetch API to call the search endpoint.
- The request includes the keyword and any selected filters.
- The JSON response contains an items array and a filters object.
- The product grid is rebuilt from the returned JSON data.
- A short debounce is used while typing to reduce unnecessary requests.
- Escape, the backdrop, or the Close button closes the search panel.

5. Search Matching

- Keyword matching is performed against product_name.
- SKU is also searched.
- The two conditions are combined using OR.
- The configured Minimum Characters value controls when keyword search is allowed.
- When no keyword is entered, the extension returns default products.

6. Category Filter

Category filters are generated dynamically from active direct children of the current store root category.

Example request: ?q=vitamin&category;[]=3

- The selected category ID is passed as category[].
- Products are matched through catalog_product_entity and catalog_category_product.
- Multiple category IDs are supported.
- Category counts are calculated from active matching search products.
- Categories with zero matching products are not displayed.

7. Price Range Filter

The price filter is available from the price information returned for the current query.

- The API returns the current minimum and maximum active product price.
- The UI provides minimum and maximum price controls.
- The customer clicks Apply to activate the selected range.
- price_min applies a lower price boundary.
- price_max applies an upper price boundary.
- Price filtering can be combined with keyword and category filters.
- Empty price values do not restrict the search.

Example

/kiwi/search/index/?q=vitamin&price;_min=5&price;_max=10

8. Color Filter

- Color is read from the search product data.

- Available colors are grouped and returned with counts.
- Color values are submitted using color[].
- Multiple colors can be selected.
- If no color data exists for the current query, the Color section is empty.

9. Search Response

A successful response contains:

Response field	Description
items	Matching product records.
items.product_name	Product display name.
items.sku	Product SKU.
items.price	Product price.
items.url	Product URL.
items.image	Product image URL.
filters.categories	Matching category names, IDs and counts.
filters.colors	Available color values and counts.
filters.price	Minimum and maximum price for the current keyword.

10. CSV Search Data

The extension exports Magento product data to the configured CSV file and imports that data into the `kiwi_search_product` database table.

- CSV columns generated: `product_name`, `sku`, `price`, `url`, `image`, `color`.
- Only enabled Magento products are exported.
- Products that are not visible on the frontend are excluded from export.
- The importer validates required fields: `product_name`, `sku`, `price`, `url` and `image`.
- The importer normalizes product and image URLs.
- The search table is truncated before a CSV import and then populated from the CSV records.
- Each imported record is marked active.

11. Cron Synchronization

- `kiwi_search_generate_csv` runs the CSV product export.
- `kiwi_search_sync_products` imports the CSV into `kiwi_search_product`.
- Both jobs use Magento's default cron group.
- The configured CSV Sync Time controls the schedule configuration used by the module.
- If the module is disabled, the product synchronization job exits without importing.

12. Database Structure

- Main table: `kiwi_search_product`.
- Unique SKU constraint prevents duplicate SKU records.
- Stored fields include SKU, product name, price, URL, image, color, active status and timestamps.
- Category membership is resolved against Magento's catalog product/category tables during search/filter operations.

13. User Guide – Typical Usage

Step 1: Click/focus the Magento search box.

Step 2: Enter a keyword such as vitamin.

Step 3: Review matching products.

Step 4: Optionally select a category.

Step 5: Optionally select a color when available.

Step 6: Select a price range and click Apply.

Step 7: Click a product card to open the product page.

14. Functional Test Checklist

Test	Expected result
Keyword search	Matching products are returned through AJAX.
SKU search	Products matching the SKU are returned.
Category selection	Only products assigned to selected category are returned.
Price range	Only products inside the selected range are returned.
Color selection	Only products matching selected color values are returned when color data exists.
Combined filters	Keyword and selected filters are applied together.
No keyword	Default products and available filters are returned.
Minimum character rule	Search does not run below configured minimum.
Disabled extension	Search endpoint returns no products/filters and storefront UI is disabled.