AI Product Recommendation for Magento 2

Extension Specification

This Magento 2 extension provides personalized product recommendations by tracking customer behaviour and leveraging AI-driven suggestions. It records category views, product views, search keywords, cart items to build a profile for each customer or guest. To utilize the extension's features on the storefront, the merchant must set up an OpenAI account and generate an API key to configure in the Magento backend. The cost of usage will vary depending on the OpenAI model selected for product recommendations.

When a user visits the site, the extension retrieves relevant products based on their viewed categories and recent searches, filtering out carted, wishlisted, out-of-stock, and child configurable products. It then selects set of relevant items and sends them, along with the user's search and purchase history, to the OpenAI API, which returns a set of personalized product recommendations. These recommendations are stored and can be rendered on any frontend page.

To maintain efficiency, the extension ensures that no duplicate requests are made if recommendations already exist. A daily cron job clears outdated recommendations if new user activity is detected and removes visitor data older than specific number of days. Admins can configure limits for stored data, manage API credentials, and enable/disable the extension via the backend. This ensures a cost-effective, AI-powered personalized shopping experience that boosts engagement and conversions.

Features

- **Adds smart product suggestions based on customer behavior.**
- **Provides setup option for merchants to connect with OpenAI for recommendations.**
- **Displays personalized products on storefront.**
- **Excludes items already in cart, wishlist, out of stock, or product variants.**
- **Cleans old data daily to keep the store fast and updated.**

Admin Panel Configuration

Navigate to **Admin > Stores > Configuration > Pits > AI Product Recommendation**.

Figure 1 – AI Product Recommendation Settings – General Settings

Figure 1 displays the admin general settings configurations of the extension, The **General Settings** encompass the following fields

- **Module Enable** – Enable / Disable module.
- **Open AI Api Key** – Create API key from Open AI for models.
- **Open AI API Endpoint** – Always set Chat Completions API (https://api.openai.com/v1/chat/completions) as endpoint to integrate OpenAI models with extension.
- **Maximum Search Terms Limit For User** – Set limit for search terms to be tracked for each user.

- **Maximum Category Limit For User** – Set limit for categories to be tracked for each user.
- **Maximum Products Limit For User** – Set Limit for products to be tracked for each user.
- **Products Passed To OpenAI API** – Set the limit for filtered available products which needs to be passed to OpenAI API as input. (It is suggested to pass 50-250 products to obtain best results)
- **Products To Be Recommended By OpenAI API** – Set the limit for filtered products which is suggested by AI (It is suggested to pass between 5 to 25 best results).
- **Minimum Customer Data Limit** – Minimum data required for the customers to fetch recommendations from Open AI.
- **OpenAI Model** – 6 Open AI models are provided as options to choose from.
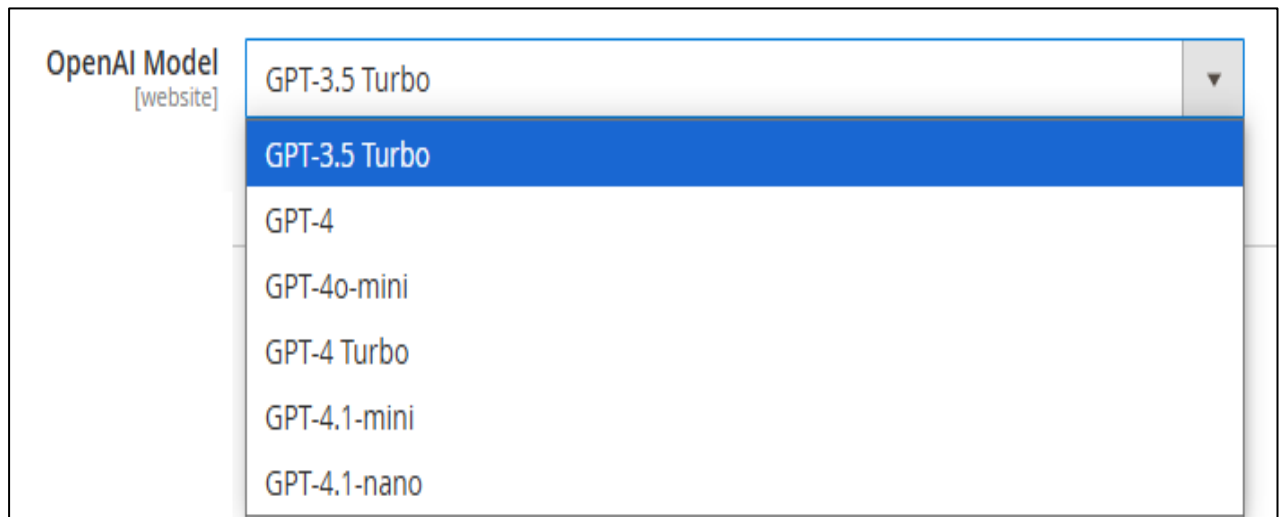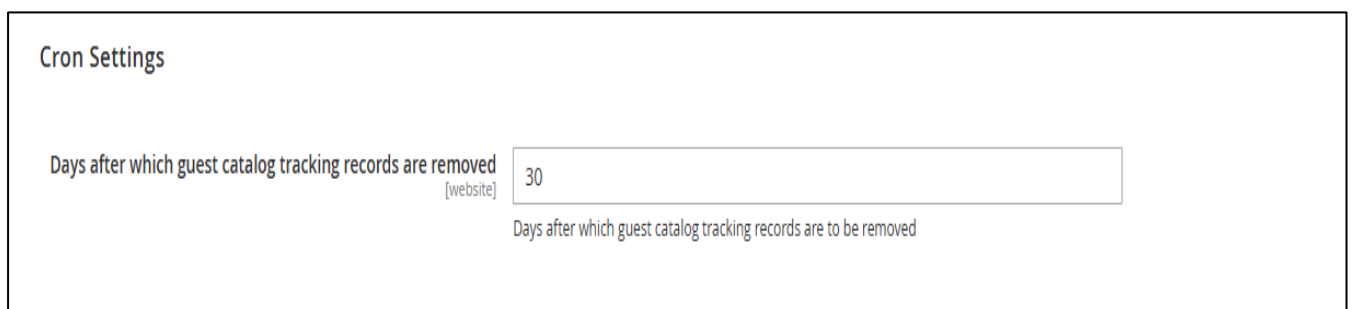

Figure 2 – List of OpenAI models

Figure 3 – AI Product Recommendation Settings – Cron Settings

Figure 3 displays the admin cron configurations of the extension, The **Cron settings** *encompass the following fields*

- **Days after which guest tracking records are removed** - The tracked records of guest users are removed automatically after certain days.

Workflow

Creation of Open AI API Keys

Admin is required to create an Open AI API Key to obtain api keys.

Navigate to https://platform.openai.com/api-keys after creation of Open AI account. And admin can create a key which needs to be saved in Magento admin configuration so users can make use of Open AI models.
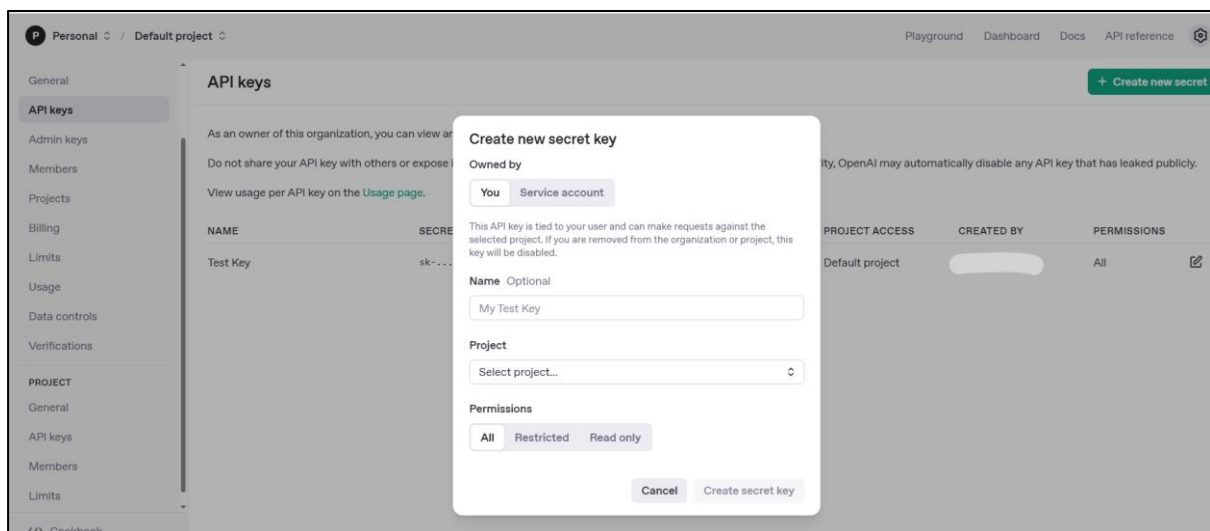

Figure 4 – Open AI API Key Creation

Tracking of Catalog Views

When a user visits a category or product page, the visit is recorded in a custom table. If the admin sets a store limit for categories or products in the backend, only that number of category and product view records will be stored per user.

Tracking of Search Terms

When a user searches a term, the search is also recorded in a custom table. If the admin sets a store limit for search terms in the backend, only that number of search

term records will be stored per user.

To ensure the recommendations are meaningful and personalized, it filters out products that the user has already added to their cart or wishlist, as well as those that are currently out of stock or are child items of configurable products.

Backend

Open AI model token optimization

The backend configurations

'**Maximum Search Terms Limit For User**', '**Maximum Category Limit For User**', '**Maximum Product Limit For User**', '**Products Passed To OpenAI API**', '**Minimum Customer Data Limit**' and '**Products To Be Recommended By Open API**' are designed to optimize the performance and cost-efficiency of the OpenAI model integration.

Important: As the values of following backend configuration fields are increased:

Maximum Search Terms Limit For User

Maximum Category Limit For User

Maximum Product Limit For User

Products Passed To OpenAI API

Minimum Customer Data Limit

Products To Be Recommended By Open API,

The OpenAI token consumption increases, and accuracy improves as well

---

It is recommended to set the **Products Passed To OpenAI API** between **50 and 200** to minimize input token usage, which helps in reducing API costs. Similarly, the **Products To Be Recommended By Open API** should ideally be set between **5 and 25** to allow OpenAI to return the most relevant product recommendations while keeping output token usage and API costs low.

Figure 5 – Admin configuration fields to optimise token usage and better performance of Open AI models.

Creation of AI Product Recommendation Widget

Navigate to **Content** > **Elements** > **Widgets** > **Click on Add Widget.**

Figure 6 – Create a new widget named ' **AI Product Recommendations'**

Create a widget with type '**AI Product Recommendations**' to render recommended products from OpenAI APIs.

Figure 7 – Created widget '**AI Product Recommendations'**

A widget has been added in the backend, which can be configured and placed on any page to render dynamic content on the frontend.

Figure 8 – Setting storefront properties for widget

Figure 9 – Setting options for widget

The frontend label and information for the widget are displayed on the frontend as shown in Figure 10. The number of items to display in the widget, which is set to '5' as shown in Figure 9, is also reflected on the frontend accordingly.
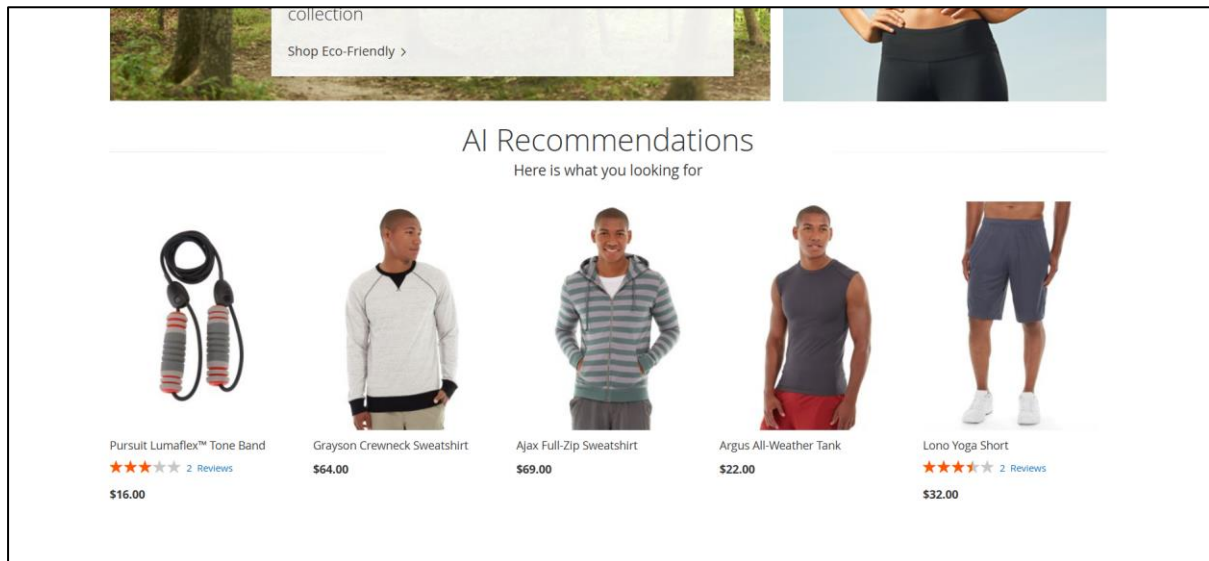
Figure 10 – Recommendation Widget

Figure 11 – Setting layout updates for widget

In the widget settings, we can define the layout location where the recommended products should be displayed on the frontend. In the Figure-11, the widget is configured to appear on the CMS Home Page within the main content container. It is shown in main content container of CMS Home Page as shown in Figure-12.
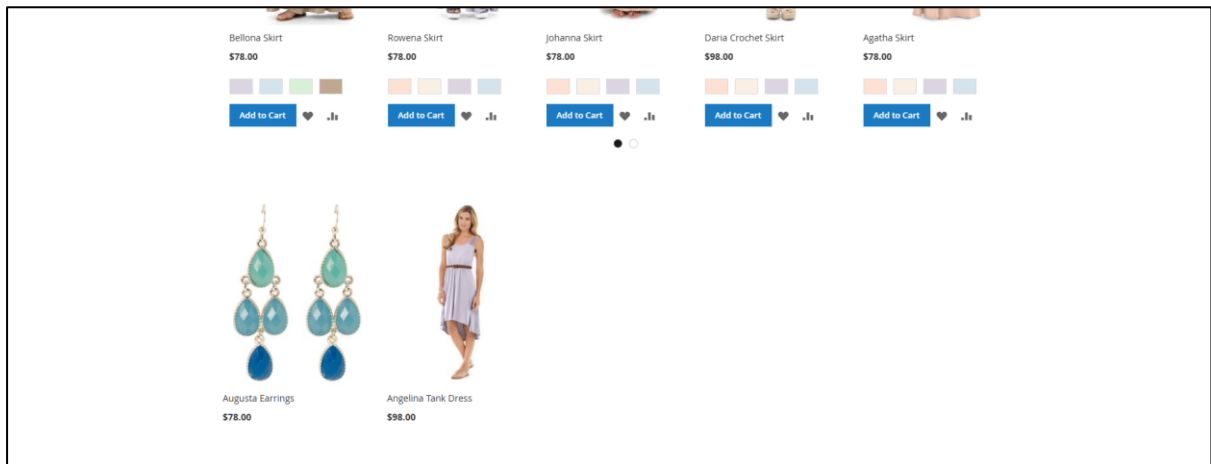
Figure 12 - AI Product Recommendations Widget in CMS Home page main content container.

Figure 13 - AI Product Recommendations Widget in Shopping Cart main content top.

The widget is populated based on the recommended products obtained from AI for each customer. The OpenAI API is called for a user when user navigates to a certain number of product/category pages and searches performed. The OpenAI API is called, and the products are recommended by it based on a structured prompt passed to AI API.

Adding Widget in Blocks and Pages

Admin can also add widget to existing pages and blocks.

1. In the following example, an existing block 'Home Page Block' is considered from **Content > Elements > Blocks**.

Figure 14 - Editing existing block.

2. Click on 'Edit with Page Builder' and click on 'Edit' of HTML code

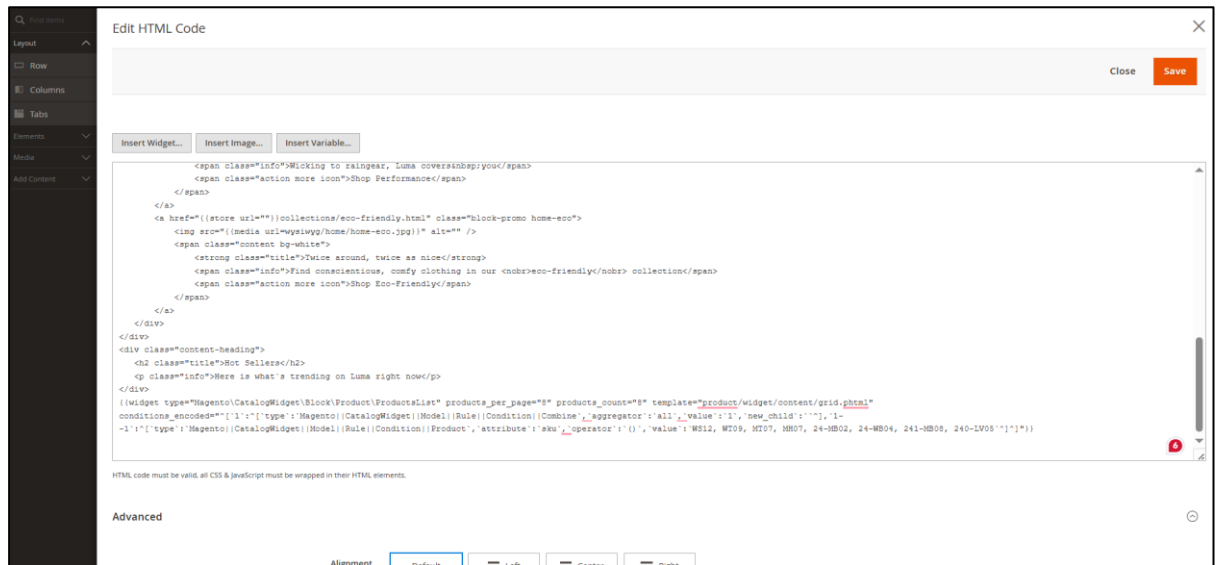Figure 15 – Edit HTML code

3. Click on Insert Widget.

Figure 16 – Insert widget to block

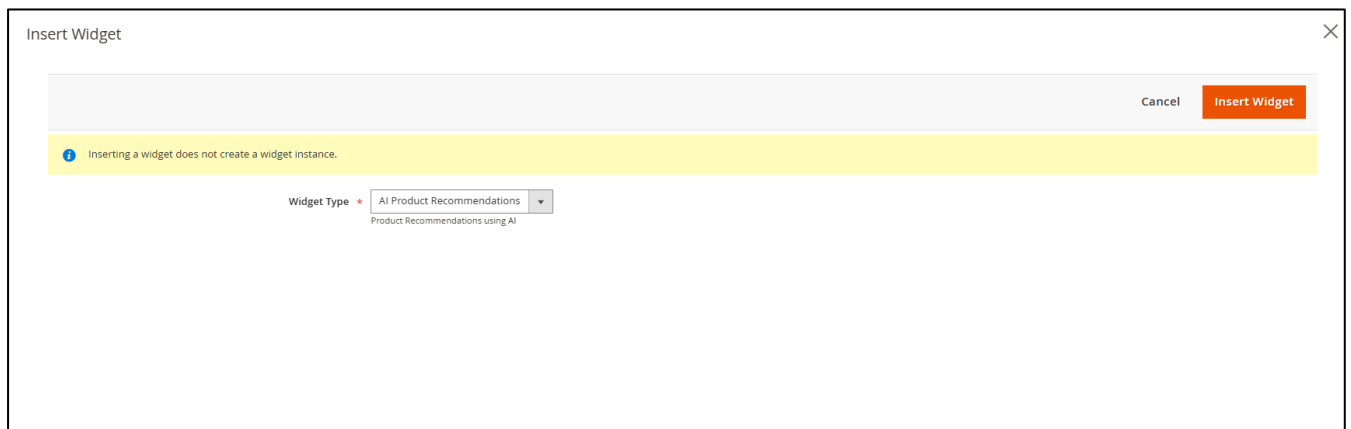4. Choose widget type as 'AI Product Recommendations'

Figure 17 – Choose Widget type as AI Product Recommendations

5. The widget is inserted into the block with class 'Pits\AIProductRecommendation\Block\Widget\RecommendedProducts' and click on 'Save'
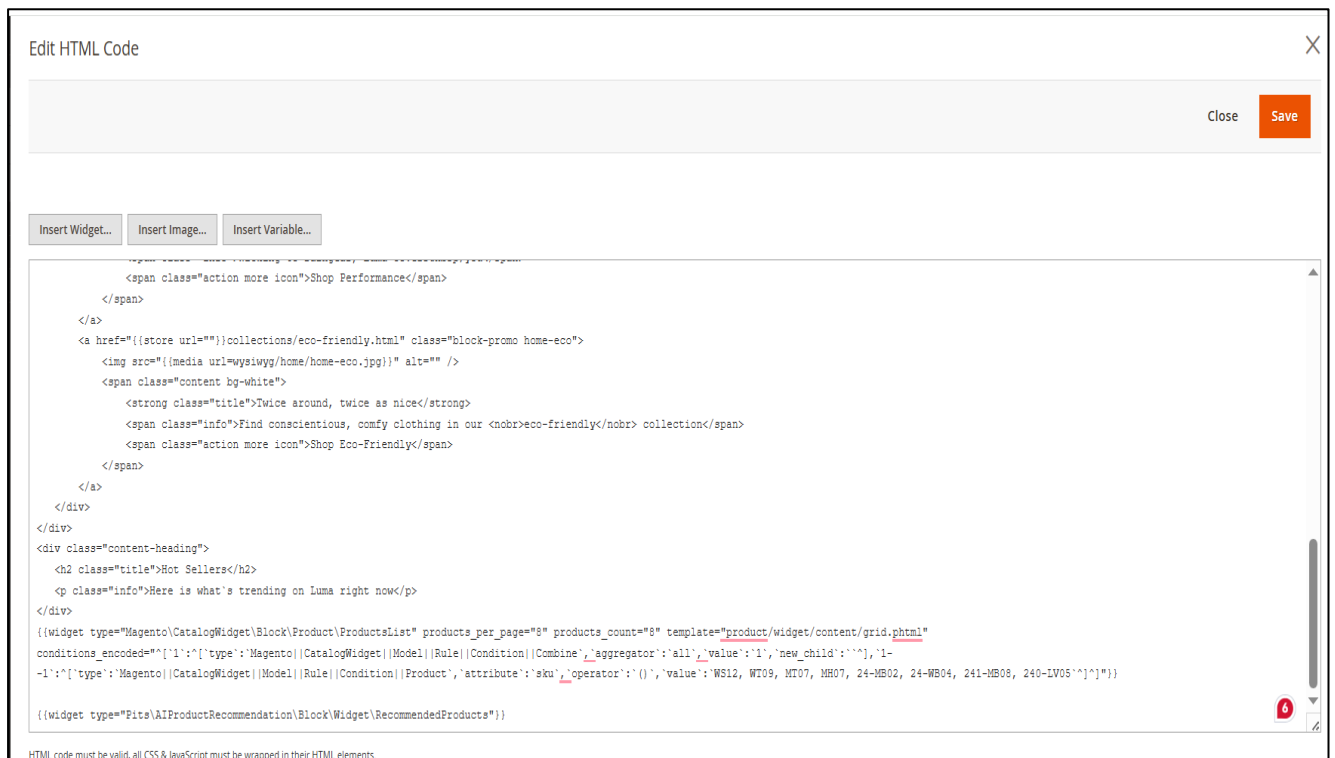
Figure 18 – Recommendation Widget is added

Deletion of guest records.

The cron is setup to remove the guest tracking items stored in the database after a certain period to remove unwanted storage in database. Figure-3 shows the field set in backend to remove the tracked records of guest users automatically after certain days.

Technical Requirements / Compatible with:

- Magento Open Source 2.4.x

Supported Languages

- English, German, French, Italian

Installation Steps

To install the AI Product Recommendation extension, follow the steps below.

1. Download the extension.
2. Access your web server directories and unzip and upload the content of the zip file to app/code directory.
3. Run the below commands on terminal
   - php bin/magento setup:di:compile
   - php bin/magento setup:upgrade
   - php bin/magento setup:static-content:deploy -f
   - php bin/magento cache:clean

Change Log / Release Notes Version:

Version: 1.0.0: June 2025

• Initial version

Support

 If you have questions, use our contact form at webshopextension.com or email at support@webshopextension.com