Rentman 4G WooCommerce Plugin
Developer Manual, Version 4.9.0
Olaf Kampers, 20-12-2017

About this document

This document is intended for developers who want to extend the plugin or use the Rentman API to create their own. It contains some additional documentation about the code of the plugin.

rentman.php

- This is the main file of the plugin. It contains code for the display of the menu page and contains functions for logging in, checking common errors and parsing the JSON data.
- **Functions**:
- *check_compatibility*: Checks the PHP time limit and checks whether images can be uploaded to the server.
- *check_github_updater*: Checks if the GitHub Updater plugin has been activated to search for automatic updates.
- *completedata:* Checks if a username, account and password were filled in.
- *do_request*: Main request function that is used for all API requests. Uses cURL to send a message to the server.
- *error_info:* Displays additional error information about a request.
- *get_dates*: Receives the selected rental period dates from the state.
- *login_user*: Main function for the user login. Returns a token that is used in all other requests with this user.
- *menu_display*: Contains the HTML code for the display of the plugin menu and calls the right functions when certain buttons have been pressed.
- *parseResponse*: Changes the key names of the items in the response. The order of the columns is not guaranteed to always be the same, so additional key names are required.
- *receive_endpoint*: Receives the latest endpoint of the Rentman API.
- *register_settings*: Registers specific Wordpress settings and their sanitization callback.
- *register_submenu*: Adds the Rentman page to the WooCommerce menu.
- *translatePlugin*: Loads the *rentalshop* text domain containing the translations for the plugin.

rentman_project.php

- Contains functions that use order data from WooCommerce to generate a new project in Rentman.
- **Functions**:
- *add_dates_to_email*: Adds the rental period data to the order data in the confirmation email.
- *add_project*: Receives the API endpoint and the user token. After that, it sends a 'create new project' request.
- *add_rental_data*: Adds the rental period data to the post meta of the order.
- *adjust_checkout*: Adds extra fields for the external reference, shipping phone number and shipping email to the checkout screen.
- *display_dates_in_order:* Displays the rental period dates on the order details screen.
- *get_material_array*: Generates an array containing all products that were imported from Rentman in the order.
- *planmaterial_array:* Uses the array containing all Rentman products to create a 'Planningmateriaal' array in the right request format.

rentman_requests.php

- Contains all functions that create API request objects
- **Functions**:
- *available_request:* Sets up the 'is product x available' request in the right format, ready to be encoded in JSON.
- *receive_btwcode_request*: Sets up the 'receive linked Btwcode' request in the right format, ready to be encoded in JSON.
- *rentRequest*: Builds a specific project request when one or more rentable products have been purchased.
- *saleRequest*: Builds a specific project request when only non-rentable products have been purchased. (No rental period has to be specified in this case).
- *setup_check_request:* Sets up the 'check if contact exists' request in the right format, ready to be encoded in JSON.
- *setup_discount_request*: Sets up the 'receive discount' request in the right format, ready to be encoded in JSON.
- *setup_file_request*: Sets up the 'receive all attached files' request in the right format, ready to be encoded in JSON.
- *setup_folder_request*: Sets up the 'receive all folders' request in the right format, ready to be encoded in JSON.
- *setup_import_request*: Sets up the receive all products' request in the right format, ready to be encoded in JSON.
- *setup_location_request:* Sets up the 'check if location exists' request in the right format, ready to be encoded in JSON.

- *setup_login_request*: Sets up the login request in the right format, ready to be encoded in JSON.
- *setup_newlocation_request:* Sets up the 'create a new location' request in the right format, ready to be encoded in JSON.
- *setup_newproject_request*: Sets up the new project request in the right format, ready to be encoded in JSON.
- *setup_newuser_request:* Sets up the 'create a new contact' request in the right format, ready to be encoded in JSON.
- *setup_staffel_request*: Sets up the 'receive staffel' request in the right format, ready to be encoded in JSON.
- *setup_staffelgroup_request*: Sets up the 'receive staffel group' request in the right format, ready to be encoded in JSON.

rentman_user.php
- Contains functions for the import and export of contact and location data.
- **Functions**:
- *export_users*: This function is called when a new order has been placed. It first checks whether the order contains any products that were imported from Rentman. Next up, it checks whether the customer and the location that were specified in the order already exist in Rentman by using the (email) address. A new contact and/or location is created in Rentman if they do not exist yet. The IDs of the contact and the location are passed to the *add_project()* function.

product_availability.php
- Contains all functions for the availability check of rentable products.
- **Functions**:
- *add_custom_field*: This function first checks whether the current product is a rentable product. After that, it checks if a rental period has already been specified for a product in the shopping cart. If a rental period has not been specified yet, two date fields are created where the customer can select the rental dates for his/her order.
- *add_date_checkout*: Checks whether there are any rentable products in the shopping cart. If there are, two date fields containing the specified rental period are added to the checkout screen. The *update_dates()* function is called if any of the dates is changed.
- *add_to_cart_template*: Creates a product page template for rentable products.
- *check_available*: Does the actual availability request for rentable products and uses the response to generate the right message (i.e. not available / maybe available / available).
- *init_datepickers*: Registers and localizes the script for the datepickers.

- *set_functions:* Registers and localizes the availability script used on the product page of rentable products.
- *update_amount:* Applies the *check_available()* function on products in the cart where the amount field has been updated.
- *update_dates*: Applies the availability check on all items in the cart when the rental dates have been updated.
- *woo_custom_cart_button_text*: Changes the text of the 'add to cart' button to 'Reserve' for rentable products.

product_categories.php
- Contains all functions for the creation and deletion of product categories.
- **Functions**:
- *add_category*: Checks if a certain category already exists and creates it if it does not exist.
- *arrange_folders*: Receives the name, ID and parent of every product category in the response and adds them to a new array.
- *displayChildren*: Recursive function that checks the item count of the child folders and deletes empty ones.
- *remove_empty_categories*: Removes all empty/unused product categories that originate from Rentman.
- *set_parents*: Sets the right parent to the product category.

product_import.php
- Contains all functions regarding the import of products from Rentman.
- **Functions**:
- *add_rentable_product*: Makes the *rentable* product type selectable.
- *array_to_product*: Imports five products from the product array, starting from the received index.
- *check_updated*: Compares the 'last modified' date in Rentman with the 'last modified' date of the product in Wordpress. The old product is deleted if it has been updated in Rentman.
- *compare_and_delete*: Deletes all products imported from Rentman that aren't available in Rentman anymore.
- *convert_items*: Checks if a product already exists or if it has been updated and arranges the product data from the API response in a new array.
- *delete_by_index*: Deletes up to 15 products from a given array.
- *get_product_list*: Creates a list containing all products that were imported from Rentman.
- *import_product*: Adds a product to Wordpress using 'wp_insert_post'. After that it updates the post meta and attaches all media files to the post.

- *import_products*: Main function for the import. It receives all products, attached media files and product categories from Rentman and adds the data to different arrays. After that, it registers and localizes a script that handles the actual import. Finally, it removes all empty categories from Wordpress.
- *list_of_ids:* When given a list of products, returns a list containing their IDs.
- *register_rental_product_type*: Registers the 'rentable' product type in WooCommerce.
- *rentman_ids*: Returns a list of all IDs of products imported from Rentman
- *reset_rentman*: Registers and localizes a script that deletes all Rentman products and categories from Wordpress.
- *wp_exist_post_by_title*: Checks whether a product already exists by using a database query.

product_media.php
- Contains main functions for attachment of media files to posts
- **Functions**:
- *attach_media*: Uploads images to the server and attaches them to a product.
- *get_files*: Returns a list of image file URL's for every product
- *update_images*: Updates all product images using a script

product_prices.php
- Contains all functions regarding the price calculation, discounts and fees of Rentman products in Wordpress.
- **Functions**:
- *apply_customer_discount*: Applies specific customer discounts from Rentman to the products in WooCommerce. First it checks whether the logged in user exists as a Contact in Rentman. After that it calculates the total discount for all products in the shopping cart.
- *apply_staffel*: Loads the 'staffelgroup' of each product in the cart from Rentman and determines the 'staffel' for each product based on the total number of days in the rental period and the staffelgroup. The staffels of each product are then used to calculate the total fee. (Sale products do not get an additional fee.)
- *calculate_discount*: Function for the actual discount calculation using the data received from Rentman. The total price is multiplied by -1 to subtract the discount from the cart totals.
- *calculate_fee*: Function for the actual staffel fee calculation. The price of the product is multiplied by the staffel. The difference between the new price and the original price is then added as a fee.
- *get_all_discounts*: Generates an array that links products to their respective discount. The array is used during the project export.

- *get_staffel*: Does the actual 'receive staffel' request for a certain staffel group and rental period.
- *get_staffelgroup*: Does the actual 'receive staffel group' request for a certain product.
- *get_staffels*: Generates an array that links products to their respective daily fee multipliers. The array is used during the project export.
- *my_text_strings*: Changes the 'subtotal' text strings in the shopping cart.
- *show_discount*: Contains the code for the display of the Rentman customer discount and the taxes on the product page of Rentman products.

admin_available.js
- Contains functions for the dynamic availability check of rentable products.
- **Functions**:
- *attachFunction*: Attaches the *quickCheck()* function to the 'amount' field on the product page.
- *quickCheck*: Receives the starting date, ending date and amount from the product page and receives the availability of the product using an API request.

admin_datepickers.js
- Contains functions for the datepickers.
- **Functions**:
- *ajax_post_date*: Sends a message to PHP to check if the products in the cart are available in the updated rental period.

admin_delete.js
- Contains functions for the deletion of Rentman products.
- **Functions**:
- *applyAjax*: Recursive function that sends product indices to PHP until the whole array (containing products that have to be deleted) has been covered.
- *removeFolders*: Called after the *applyAjax()* function has finished. Sends a message to PHP to remove all empty product categories from WooCommerce.

admin_images.js
- Contains functions for the update of attached images.
- **Functions**:
- *applyAjax*: Recursive function that sends image indices to PHP until the whole array (containing images that have to be uploaded) has been covered.

admin_import.js

- Contains functions for the import of Rentman products.
- **Functions**:
- *applyAjax*: Recursive function that sends product indices to PHP until the whole array (containing products that have to be imported) has been covered.
- *removeFolders*: Called after the *applyAjax()* function has finished. Sends a message to PHP to remove all empty product categories from WooCommerce.