



# Shopware

*A practical userguide to integrate Shopware e-commerce and Stanley/Stella API*

## Prerequisites

This guide will explain how to connect a Shopware 5.4.5 webshop with Stanley/Stella API Framework to periodically synchronize the product catalogue.

The guide assumes that you are familiar with Shopware and that you already have your Shopware running.

It also requires a little bit of development knowledge in order to transform the CSV files.

In this guide, we will illustrate the integration in the PHP development environment but this could also be realized in any other modern development environment.

Please note that the PHP code provided in this document is for illustration purpose only.

Further technical explanation on the API can be found at <https://api.stanleystella.com>

The demo website build with these instructions can be found at <https://shopware.stanleystella.com>

## Table of Contents

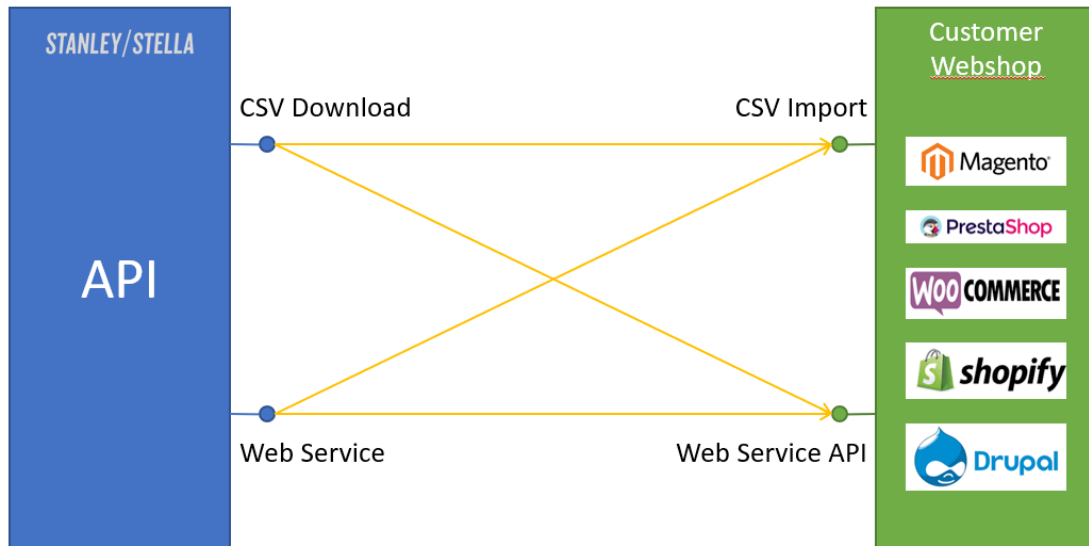
Prerequisites .....	2
Integration scenario Stanley/Stella API → Shopware .....	4
Connect to Stanley/Stella Product API in PHP .....	4
Import CSV's in Shopware.....	5
Generating the right CSV import files .....	7
Generating the Product CSV import file .....	7
Generating the Pictures CSV import file .....	8
PHP Code.....	8
Products CSV file generation .....	8
Pictures CSV file generation.....	13

## Integration scenario Stanley/Stella API → Shopware

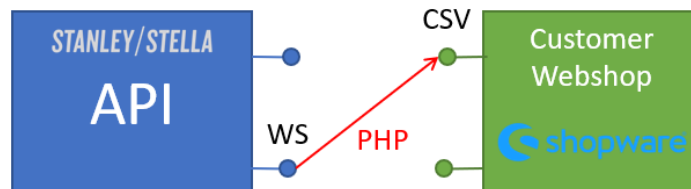
Like a lot of ecommerce solutions, Shopware provides a way to import product information:

- Manual import via CSV file

At Stanley/Stella, our API provides 2 capabilities.



In this scenario, we chose to use the REST API on Stanley/Stella side and the CSV import on Shopware side.



The integration logic in the middle has been developed in PHP.

## Connect to Stanley/Stella Product API in PHP

The first step of the integration is to call the Stanley/Stella Product REST API from the PHP code.

We use the Products V2 API which already groups the variants per style.

For simplicity, we only download the product data in 1 language (English).

See the following PHP code as inspiration.

```
function getAllSTSTProductsV2($language) {
    $url = "https://api.stanleystella.com/webrequest/productsV2/get_json";
    $jsonData = array(
        'jsonrpc' => '2.0',
        'method' => 'call',
        'params' => array(
            'db_name' => "production_api",
            'password' => "YOUR_PASSWORD",
```

```

        'user' => "YOUR_LOGIN",
        "LanguageCode" => $language,
    ),
    'id' => 0
);

$ch = curl_init($url);
$jsonDataEncoded = json_encode($jsonData);

curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $jsonDataEncoded);
curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));
$result = curl_exec($ch);

$jsonDataDecoded = json_decode(json_decode($result)->result, true);
curl_close($ch);

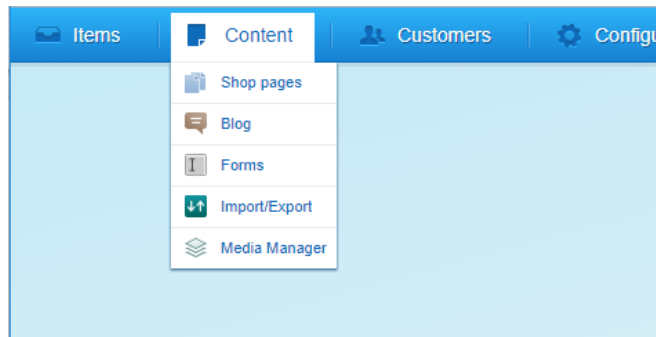
return $jsonDataDecoded;
}

```

## Import CSV's in Shopware

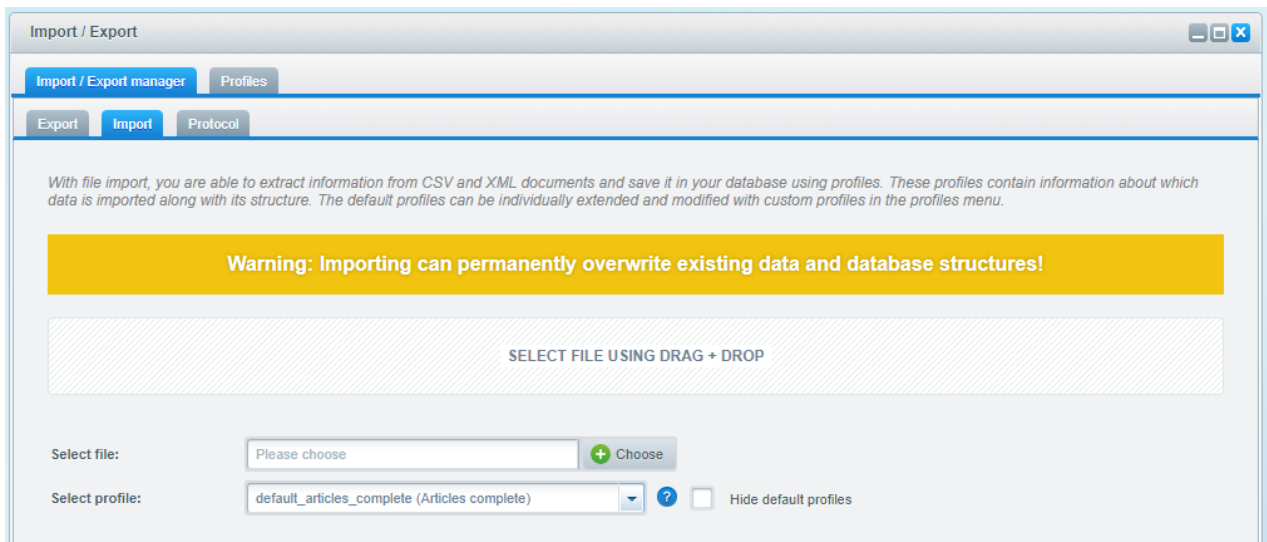
Shopware is a beautiful e-commerce platform, with a very well architecture. The structure of the Stanley/Stella collection fits perfectly in the structure of products inside Shopware.

Shopware provides many different CSV file formats for import and export. It is possible to download sample CSV files. You will find these formats in the menu "Content", under "Import/Export"

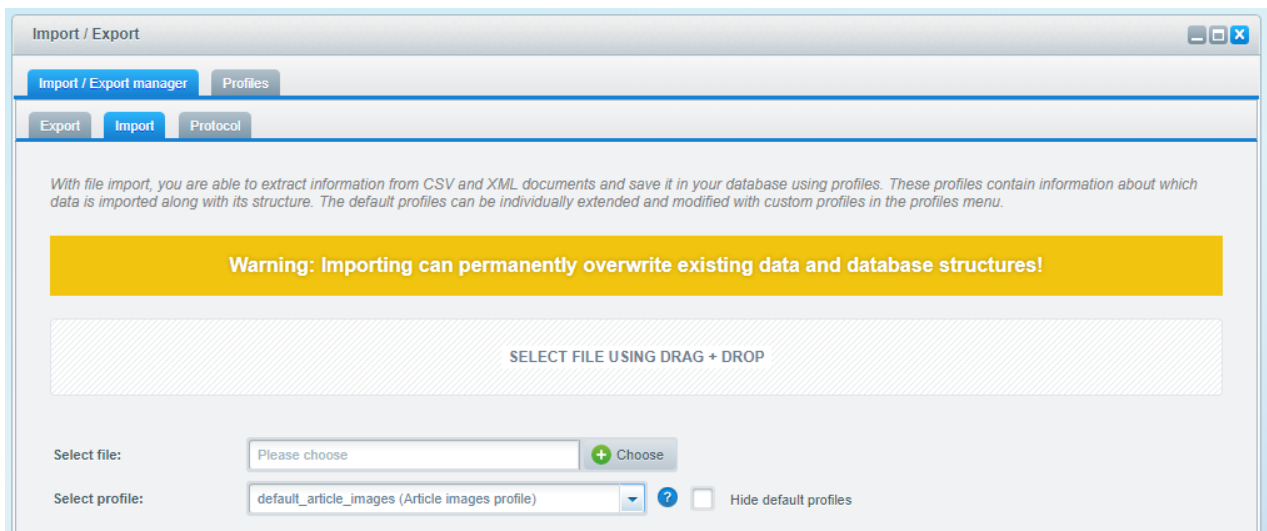


We use 2 specific formats:

- To import the products (Styles) and the combinations (Color + Size Variants), we use "default\_articles\_complete"



- To import the pictures and attach them to the right variants, we use “default\_article\_images”

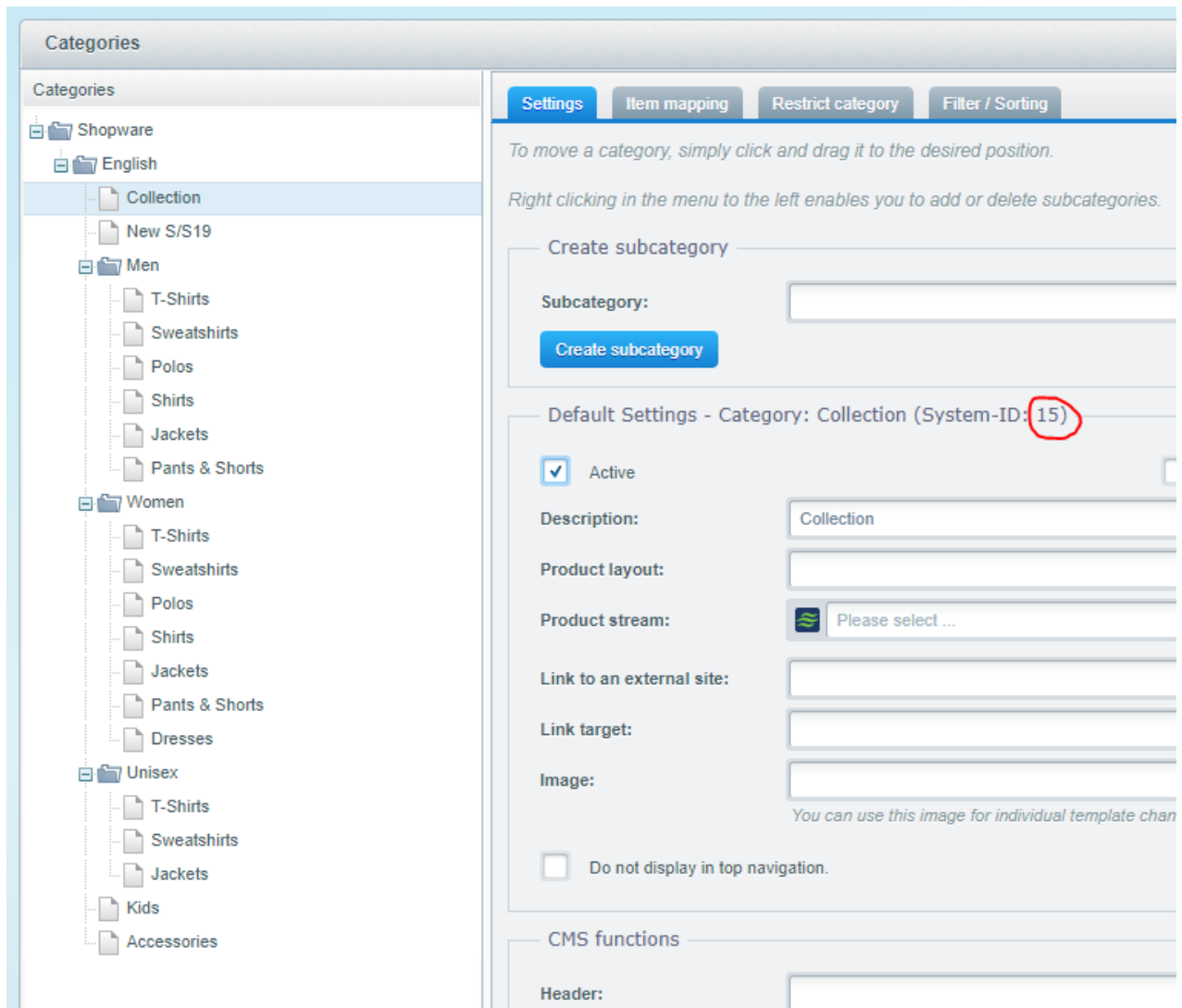


Note: There is also a CSV format to import pictures via URL's. This one did not work in our setup. But the “default\_article\_images” worked perfectly with URL's. The pictures were downloaded from the URL and stored in Shopware in several ratio's.

Shopware will not create the categories on the fly, during product import. It will create the colors and size attribute values on the fly, though. This means that we have to define all categories first.

Unfortunately, in the CSV import, Shopware uses the internal id's of these categories. This means that we will have to map the Stanley/Stella categories to the internal category id's of Shopware.

See the screenshot hereunder to see the structure of categories we defined and where to find their internal id's.



Shopware will automatically display a “NEW” logo on the products that were created less than 5 days ago (5 days is a parameter you can configure in the Shopware settings). Shopware compares the current date with the creation date of the product. And via the CSV import, it is possible to change this creation date. We will use this functionality to make sure the “NEW” logo stays on the product as long as the product is marked as new in the Stanley/Stella API.

Regarding the pictures, Shopware allows to define a sort order and a main picture and to attach the pictures to the right variant (in our case the right colour). This is a perfect behaviour and makes it easy to automate the update of the pictures exactly like defined in the Stanley/Stella API's.

## Generating the right CSV import files

### Generating the Product CSV import file

The Product CSV import file contains both lines for the styles as for the variants. So, we first call the Stanley/Stella Product API V2, with grouped variants per styles. We then loop on all styles.

- For each style, we start by preparing the data we will need : stylecode, stylename, publication status, gender, categorycode, short and long descriptions, price of the first variant, fitting, neckline, etc.
- For the categories, we add the mapping logic between the Stanley/Stella categories and the Shopware categories and their internal id's.
- We replace all “\n” of the long description with its HTML equivalent <br/>.
- And we add the line for the style in the CSV.
- Note that we don't specify the main picture for the style. This will be done in the Pictures CSV import.

We then loop on all variants of the style:

- For each variant, we start by preparing the data we will need: sku ref, color code, size code, stock, price, publication status, weight, composition, certifications, etc.
- And we add a line in the CSV for each variant

See the example of PHP code in the annexes for more details.

## Generating the Pictures CSV import file

The Pictures CSV file is easier than the product import file because it contains less columns. But it requires a little bit more complex logic because the sort order of the pictures is very important.

- We first retrieve the list of pictures using the Stanley/Stella Product Pictures API.
- We then group these pictures by style and then color. We then identify their type.
- We then retrieve the list of products using the Stanley/Stella Products V2 API. We then loop on the styles.
- For each style, we loop on all pictures that we received from the API for this style and we select some specific pictures (if they exist) in this specific order: SFM0, SFM5, PFM0, PBM0, SBM0, SBM5.
- We then add a line in the Pictures CSV for each selected picture, with the specific sort order.
- If the picture is the main picture of the style, we set the main flag to 1 in the CSV. Else it is a 2.

And that's it.

The import function of Shopware will do all the rest of the work.

## PHP Code

See the following PHP code as inspiration for generating the CSV files.

### Products CSV file generation

```
<?php
ini_set("memory_limit","128M");
ini_set("max_execution_time",9000);
ini_set('default_charset', 'utf-8');
header('Content-Type: text');
```



```

// Shopware categories
// Change these internal id's to your own categories
$categories = array();
$categories["All"] = 15;
$categories["Women"] = array();
$categories["Women"]["id"] = 14;
$categories["Women"]["TEES"] = 29;
$categories["Women"]["SHIRT"] = 32;
$categories["Women"]["SWEATS"] = 30;
$categories["Women"]["DRESSES"] = 35;
$categories["Women"]["POLO"] = 31;
$categories["Women"]["JACKETS"] = 33;
$categories["Women"]["PANTS"] = 34;
$categories["Men"] = array();
$categories["Men"]["id"] = 13;
$categories["Men"]["TEES"] = 20;
$categories["Men"]["SHIRT"] = 22;
$categories["Men"]["SWEATS"] = 23;
$categories["Men"]["POLO"] = 25;
$categories["Men"]["JACKETS"] = 26;
$categories["Men"]["PANTS"] = 28;
$categories["Unisex"] = array();
$categories["Unisex"]["id"] = 17;
$categories["Unisex"]["TEES"] = 36;
$categories["Unisex"]["SWEATS"] = 37;
$categories["Unisex"]["JACKETS"] = 38;
$categories["Unisex"]["ACCESSORY"] = 19;
$categories["Unisex"]["SHOES"] = 19;
$categories["Kids"] = array();
$categories["Kids"]["id"] = 18;
$categories["NEW"] = 16;

$languagecode = 'en_US';
$stpm = getAllSTSTProductsV2($languagecode);

$csvlines = array();
// CSV Header
$csvlines[] = addShopwareCSVHeader();

foreach ($stpm as $stylecode => $style) {
    $stylename = $style["StyleName"];
    $stylecode = $style["StyleCode"];
    // Categories
    $gender = $style["Gender"];
    $category = $style["CategoryCode"];
    // Add collection category id
    $categorystring = $categories["All"];
    // Add gender category id
    if ($gender=="Boys" || $gender=="Girls") {
        $gender = "Kids";
    }
    $categorystring.="|";
    $categorystring.= $categories[$gender]["id"];
    // Add type category id for all but kids
    if ($gender!="Kids") {
        $categorystring.="|";
        $categorystring.= $categories[$gender][$category];
    }
    // Add Men + Women if unisex
    if ($gender=="Unisex") {
        $categorystring.="|";
        $categorystring.= $categories["Men"]["id"];
        $categorystring.="|";
        $categorystring.= $categories["Women"]["id"];
    }
    // Add newstyle category id
    if ($style["Variants"][0]["NewStyle"]) {
        $categorystring.="|";
        $categorystring.= $categories["NEW"];
    }
    // Descriptions
    $shortdescription = $style["ShortDescription"];
    $longdescription = str_replace("\n", "<br/>", $style["LongDescription"]);
    // Price : take first variant price
    $firstvariant = reset($style["Variants"]);

```

```

$price = $firstvariant["Price<10 EUR"];
// Tags
$fit = $style["Fit"];
$neckline = $style["Neckline"];
$sleeves = $style["Sleeve"];
$weight = "";
$composition = "";
$construction = "";
$finish = "";
$certifications = array();
// Status
if ($style["StylePublished"]) {
    $published = 1;
} else {
    $published = 0;
}
// Main picture
if (isset($style["MainPicture"][0]["HTMLPath"])) {
    $imageurl = $style["MainPicture"][0]["HTMLPath"];
} else {
    $imageurl = "";
}

// Add CSV line for the style
// $imageurl -> pictures will be done separately
$csvlines[] = addShopwareCSVStyleLine($stylecode, "", $stylename, $price, $published, 0,
$shortdescription, $longdescription, "", "", $categorystring, "", $fit, $sleeves, $neckline,
$weight, $composition, $construction, $finish, $certifications);

foreach ($style["Variants"] as $key => $variant) {
    $b2bskuref = $variant["B2BSKUREF"];
    $colorcode = $variant["ColorCode"];
    $sizecode = $variant["SizeCodeNavigation"];
    $size = $variant["SizeCode"];
    $color = $variant["ColorCode"];
    $stock = $variant["Stock"];
    $price = $variant["Price<10 EUR"];
    if ($variant["Published"]) {
        $published = 1;
    } else {
        $published = 0;
    }
    // Tags
    $weight = $variant["Weight"];
    $composition = $variant["CompositionList"];
    $construction = $variant["ConstructionList"];
    $finish = $variant["FinishList"];
    $certifications = array();
    $certifications["GOTS"] = $variant["GOTS"];
    $certifications["OCS100"] = $variant["OCS100"];
    $certifications["OCSBlended"] = $variant["OCSBlended"];
    $certifications["Ecotex"] = $variant["Ecotex"];
    $certifications["Fairwear"] = $variant["Fairwear"];

    $csvlines[] = addShopwareCSVStyleLine($stylecode, $b2bskuref, $stylename, $price,
$published, $stock, $shortdescription, $longdescription, $size, $color, $categorystring, "", $fit,
$sleeves, $neckline, $weight, $composition, $construction, $finish, $certifications);
}

}

// Write to CSV
$fp = fopen('shopware_product_import.csv', 'w');
foreach ($csvlines as $key => $csvline) {
    fputcsv($fp, $csvline, ";");
}
fclose($fp);

function addShopwareCSVHeader() {
    $csvrow = array();
    $csvrow[] = "ordernumber";
    $csvrow[] = "mainnumber";
    $csvrow[] = "name";
}

```

```

$csvrow[] = "additionalText";
$csvrow[] = "supplier";
$csvrow[] = "tax";
$csvrow[] = "price_EK";
$csvrow[] = "pseudoprice_EK";
$csvrow[] = "baseprice_EK";
$csvrow[] = "price_H";
$csvrow[] = "pseudoprice_H";
$csvrow[] = "baseprice_H";
$csvrow[] = "active";
$csvrow[] = "instock";
$csvrow[] = "stockmin";
$csvrow[] = "description";
$csvrow[] = "description_long";
$csvrow[] = "shippingtime";
$csvrow[] = "added";
$csvrow[] = "changed";
$csvrow[] = "releasedate";
$csvrow[] = "shippingfree";
$csvrow[] = "topseller";
$csvrow[] = "keywords";
$csvrow[] = "minpurchase";
$csvrow[] = "purchasesteps";
$csvrow[] = "maxpurchase";
$csvrow[] = "purchaseunit";
$csvrow[] = "referenceunit";
$csvrow[] = "packunit";
$csvrow[] = "unitID";
$csvrow[] = "pricegroupID";
$csvrow[] = "pricegroupActive";
$csvrow[] = "laststock";
$csvrow[] = "suppliernumber";
$csvrow[] = "weight";
$csvrow[] = "width";
$csvrow[] = "height";
$csvrow[] = "length";
$csvrow[] = "ean";
$csvrow[] = "similar";
$csvrow[] = "configuratorsetID";
$csvrow[] = "configuratoretype";
$csvrow[] = "configuratorOptions";
$csvrow[] = "categories";
$csvrow[] = "propertyGroupName";
$csvrow[] = "propertyValueName";
$csvrow[] = "accessory";
$csvrow[] = "imageUrl";
$csvrow[] = "main";
$csvrow[] = "attr1";
$csvrow[] = "attr2";
$csvrow[] = "attr3";
$csvrow[] = "purchasePrice";
$csvrow[] = "metatitle";

```

```

return $csvrow;

```

```

}

```

```

function addShopwareCSVStyleLine($stylecode, $b2bskuref, $stylename, $price, $active, $instock,
$shortdescription, $longdescription, $sizename, $colorname, $categories, $imageurl, $fit,
$slleeves, $neckline, $weight, $composition, $construction, $finish, $certifications) {
    $csvrow = array();
    if ($b2bskuref!="") {
        $csvrow[] = $b2bskuref;
    } else {
        $csvrow[] = $stylecode;
    }
    $csvrow[] = $stylecode;
    $csvrow[] = $stylename;
    $csvrow[] = "";
    $csvrow[] = "Stanley/Stella";
    $csvrow[] = 21;
    $csvrow[] = $price;
    $csvrow[] = 0;
    $csvrow[] = "";
    $csvrow[] = "";
    $csvrow[] = "";
    $csvrow[] = "";
    $csvrow[] = "";
}

```

```

$csvrow[] = $active;
$csvrow[] = $instock;
$csvrow[] = 0;
$csvrow[] = $shortdescription;
$csvrow[] = $longdescription;
$csvrow[] = ""; // Shipping time
$csvrow[] = date("y-m-d");
$csvrow[] = date("y-m-d H:i:s");
$csvrow[] = "";
$csvrow[] = 0;
$csvrow[] = 0;
$csvrow[] = "";
$csvrow[] = 1;
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = 0;
$csvrow[] = 0; // laststock
if ($b2bskuref!="") {
    $csvrow[] = $b2bskuref;
} else {
    $csvrow[] = $stylecode;
}
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
if ($b2bskuref!="") {
    $csvrow[] = 8; // ConfiguratorsetID
    $csvrow[] = 2; // 0 = standard 1 = selection 2 = pictures
    $options = "StStSizes:".$$size."|StStColors:".$$colorname;
    $csvrow[] = $options;
} else {
    $csvrow[] = ""; // ConfiguratorsetID
    $csvrow[] = "";
    $csvrow[] = "";
}
$csvrow[] = $categories;
$csvrow[] = "Clothing"; // Properties
$propertiesvalues = "Fit:".$$fit."|Sleeves:".$$sleeves."|Neckline:".$$neckline;
if ($weight!="") $propertiesvalues .= "|Weight:".$$weight;
if ($composition!="") $propertiesvalues .= "|Composition:".$$composition;
if ($construction!="") $propertiesvalues .= "|Construction:".$$construction;
if ($finish!="") $propertiesvalues .= "|Finish:".$$finish;
foreach ($certifications as $certif => $certifboolean) {
    if ($certifboolean) $propertiesvalues.="|Certifications:".$$certif;
}
$csvrow[] = $propertiesvalues; // Properties values
$csvrow[] = "";
if ($imageurl!="") {
    $csvrow[] = $imageurl;
    $csvrow[] = 1;
} else {
    $csvrow[] = "";
    $csvrow[] = "";
}
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = "";
$csvrow[] = 0;
$csvrow[] = $stylename;

return $csvrow;
}

function getAllSTSTProductsV2($language) {
    $url = "https://api.stanleystella.com/webrequest/productsV2/get_json";
    $jsonData = array(
        'jsonrpc' => '2.0',
        'method' => 'call',
    );
}

```

```

        'params' => array(
            'db_name' => "production_api",
            'password' => "YOUR_PASSWORD",
            'user' => "YOUR_LOGIN",
            "LanguageCode" => $language,
        ),
        'id' => 0
    );

    $ch = curl_init($url);
    $jsonDataEncoded = json_encode($jsonData);

    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
    curl_setopt($ch, CURLOPT_POST, 1);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $jsonDataEncoded);
    curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));
    $result = curl_exec($ch);

    $jsonDataDecoded = json_decode(json_decode($result)->result, true);
    curl_close($ch);

    return $jsonDataDecoded;
}

```

## Pictures CSV file generation

```

<?php
ini_set("memory_limit","128M");
ini_set("max_execution_time",9000);
ini_set('default_charset', 'utf-8');
header('Content-Type: text');

// 1. Get all product images
$stpm = getAllSTSTProductImages();

$pics = array();
foreach ($stpm as $key => $pic) {
    // Group by color
    // Select only SFM0, SFM5, SBM0, SB5, PFM0, PBM0
    // And add position
    // Get main indication from the Product API
    $stylecode = $pic["StyleCode"];
    $colorcode = $pic["ColorCode"];
    $stype = substr($pic["FName"],0,4);
    $imageurl = $pic["HTMLPath"];
    if (!isset($pics[$stylecode])) $pics[$stylecode] = array();
    if (!isset($pics[$stylecode][$colorcode])) $pics[$stylecode][$colorcode] = array();

    $pics[$stylecode][$colorcode][$stype] = $imageurl;
}

$csvlines = array();
// CSV Header
$csvlines[] = addShopwareCSVHeader();

// 2. Get all products, to identify the main picture
$languagecode = 'en_US';
$ap = getAllSTSTProductsV2($languagecode);
foreach ($ap as $key => $style) {
    if ($style["StylePublished"]) {
        if (isset($style["MainPicture"][0]["HTMLPath"])) {
            $stylecode = $style["StyleCode"];
            foreach ($pics[$stylecode] as $colorcode => $pp) {
                if (isset($pp["SFM0"])) {
                    if ($style["MainPicture"][0]["HTMLPath"]===$pp["SFM0"]) {

```



```

$csvrow[] = "relations";

return $csvrow;
}

function addShopwareCSVStyleLine($stylecode, $imageurl, $main, $position, $relations) {
    $csvrow = array();
    $csvrow[] = $stylecode;
    $csvrow[] = $stylecode;
    $csvrow[] = $imageurl;
    $csvrow[] = $main;
    $csvrow[] = "";
    $csvrow[] = $position;
    $csvrow[] = "";
    $csvrow[] = "";
    $csvrow[] = $relations;

    return $csvrow;
}

function getAllSTSTProductImages() {
    $url = "https://api.stanleystella.com/webrequest/products_images/get_json";
    $jsonData = array(
        'jsonrpc' => '2.0',
        'method' => 'call',
        'params' => array(
            'db_name' => "production_api",
            'password' => "YOUR_PASSWORD",
            'user' => "YOUR_LOGIN"
        ),
        'id' => 0
    );

    $ch = curl_init($url);
    $jsonDataEncoded = json_encode($jsonData);

    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
    curl_setopt($ch, CURLOPT_POST, 1);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $jsonDataEncoded);
    curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));
    $result = curl_exec($ch);

    $jsonDataDecoded = json_decode(json_decode($result)->result, true);
    curl_close($ch);

    return $jsonDataDecoded;
}

function getAllSTSTProductsV2($language) {
    $url = "https://api.stanleystella.com/webrequest/productsV2/get_json";
    $jsonData = array(
        'jsonrpc' => '2.0',
        'method' => 'call',
        'params' => array(
            'db_name' => "production_api",
            'password' => "YOUR_PASSWORD",
            'user' => "YOUR_LOGIN",
            "LanguageCode" => $language,
        ),
        'id' => 0
    );

    $ch = curl_init($url);
    $jsonDataEncoded = json_encode($jsonData);

    curl_setopt($ch, CURLOPT_FOLLOWLOCATION, true);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
    curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, 0);
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, 0);
    curl_setopt($ch, CURLOPT_POST, 1);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $jsonDataEncoded);
    curl_setopt($ch, CURLOPT_HTTPHEADER, array('Content-Type: application/json'));

```

```
$result = curl_exec($ch);  
  
$jsonDataDecoded = json_decode(json_decode($result)->result, true);  
curl_close($ch);  
  
return $jsonDataDecoded;  
}
```