

How to Create a Basic Dropshipping WordPress plugin for WooCommerce

<https://orbisius.com/blog/create-basic-dropshipping-plugin-woocommerce-p6388>



Test WordPress site deployed in seconds? Yes, it's possible. Visit [qSandbox](#) for more info.

What is Dropshipping?

Dropshipping is a business that allows you to sell products that you don't own or hold an inventory for. The order is placed on your site and then the order info is forwarded to the actual manufacturer to process the order and ship it.

In this business model you set your own prices for the products and you earn a profit by keeping a margin on every product.

Because we don't know what the actual retailer would be, we'll prepare the ordered products and be ready to call and send that information to.

Here's a basic WordPress plugin that hooks into WooCommerce checkout process and prepares the ordered products to send them to another service for shipment.

Because each shipping service is different it's not possible to have a universal solution.

We can add different addons to support a new service.

The very first step is to hook into the proper woocommerce action so your code can be notified when an order has been placed.

The best candidate is status = processing

woocommerce_order_status_processing

Here's a video that explains how the plugin works.

The code below will hook into the action above and when called will loop through all the ordered products except the downloadable and virtual ones. If you want you can turn off the check and have all the ordered products be sent to the external service.

It will add a note to the order to say shipped.

You will need to customize the code to fit your needs. Some APIs will accept key value pairs, JSON or XML.

Possible Improvement

Also it's good to add logging so you know what the API has returned and you can troubleshoot in case of an error.

Prepare the data in the format that your dropshipping provider is expecting in.

Prefix order id with the environment e.g. dev_, stg_ or _live. That way there won't be an order id conflict when when testing the plugin on different sites.

If you need this plugin customized feel free to [contact us](#)

```
<?php
```

```
/*
```

```
Plugin Name: Orbisius Dropshipping Basic Plugin Tutorial
```

```
Plugin URI: https://orbisius.com/6388
```

```
Description: This tutorial provides an example how to hook into WooCommerce checkout pro
```

Version: 1.0.0
Author: Orbisius
Author URI: <https://orbisius.com>
*/

```
$orbisius_dropshipping_obj = orbisius_tutorial_basic_dropshipping::get_instance();
```

```
// Processing — Payment received (paid) and stock has been reduced; order is awaiting fulfillment
```

```
// All product orders require processing, except those that only contain products which are bo
```

```
add_action( 'woocommerce_order_status_processing', array( $orbisius_dropshipping_obj, 'pre
```

```
/**
```

```
*
```

```
*/
```

```
class orbisius_tutorial_basic_dropshipping {
```

```
    private $user_agent = '';
```

```
    /**
```

```
    * @param int $order_id
```

```
    * @return orbisius_tutorial_basic_dropshipping_result
```

```
    */
```

```
    public function prepare_for_send( $order_id ) {
```

```
        try {
```

```
            $res_obj = new orbisius_tutorial_basic_dropshipping_result();
```

```
            if ( empty( $order_id ) ) {
```

```
                throw new exception( 'Missing order id' );
```

```
            }
```

```
            $order = wc_get_order( $order_id );
```

```
            if ( empty( $order ) ) {
```

```
                throw new exception( 'Invalid order id' );
```

```
            }
```

```
            // $customer_obj = $order->get_user();
```

```
            $billing_email = $order->get_billing_email();
```

```
            $phone         = $order->get_shipping_phone();
```

```
            $phone         = preg_replace( '#\s*(x|ext).*\s*$', '', $phone ); // rm ext if any
```

```

    $company = $order->get_shipping_company();
    $city    = $order->get_shipping_city();
    $state   = $order->get_shipping_state();
    $postal_code = $order->get_shipping_postcode();
    $address = $order->get_shipping_address_1();
    $address2 = $order->get_shipping_address_2();
    $country = $order->get_shipping_country();

    $shipping_data = [
        'email'    => $billing_email,
        'phone'    => $phone,
        'address'  => $address,
        'address2' => $address2,
        'city'     => $city,
        'state'    => $state,
        'postal_code' => $postal_code,
        'company'  => $company,
        'country'  => $country,
        'first_name' => $order->get_shipping_first_name(),
        'last_name' => $order->get_shipping_last_name(),
    ];

    // Using date created because the order may be in processing state.
    $order_date = $order->get_date_created();
    $order_date_iso_fmt = date( 'c', strtotime( $order_date ) ); // 2021-11-07T00:00:00

    $order_data = [
        'order_id'    => $order_id,
        'email'       => $billing_email,
        'phone'       => $phone,
        'address'     => $address,
        'address2'    => $address2,
        'city'        => $city,
        'state'       => $state,
        'postal_code' => $postal_code,
        'company'     => $company,
        'country'     => $country,
        'products'    => [],
        'shipping_data' => $shipping_data,
        'order_date'  => $order_date,
        'order_date_fmt' => $order_date_iso_fmt,
    ];

```

```
    ];
```

```
// https://stackoverflow.com/questions/50779953/how-to-get-the-product-sku-from-order-it
```

```
foreach ( $order->get_items() as $item ) {  
    $product = wc_get_product( $item->get_product_id() );
```

```
// Skip these.
```

```
if ( $product->is_downloadable() || $product->is_virtual() ) {  
    continue;  
}
```

```
$item_sku = $product->get_sku();
```

```
$product_rec = [  
    'id' => $item->get_product_id(),  
    'product_name' => $item->get_name(),  
    'sku' => $item_sku,  
    'qty' => $item->get_quantity(),  
];
```

```
$order_data['products'][] = $product_rec;  
}
```

```
$dropshipping_api_url = 'https://example.com/api/';  
$api_params = [  
    'user' => 'user123',  
    'pass' => 'pass123',  
];
```

```
$request = wp_remote_post(  
    $dropshipping_api_url,  
    array(  
        'body' => $api_params,  
        'user-agent' => $this->user_agent,  
    )  
);
```

```
if ( is_wp_error( $request ) || wp_remote_retrieve_response_code( $request ) != 200 ) {  
    throw new Exception( "Error: " . ( is_object( $request ) ? $request->get_error_message() : '' ) );  
}
```

```

    $note = ' Shipped ';
    $order->add_order_note( $note );
    $res_obj->status = true;
} catch ( Exception $e ) {
    $res_obj->msg = $e->getMessage();
}

```

```

return $res_obj;
}

```

```

private function __construct() {
}

```

```

/**
 * Singleton pattern i.e. we have only one instance of this obj
 * This method is cool that it allow inheritance and makes the method work with all children
 * @staticvar static $instance
 * @return static
 */

```

```

public static function get_instance() {
    static $instance = null;

```

```

    if ( is_null( $instance ) ) {
        $instance = new static();
    }

```

```

    return $instance;
}
}

```

```

/**
 *
 */
class orbisius_tutorial_basic_dropshipping_result {
    public $id = "";
    public $code = "";
    public $msg = "";
    public $data = [];
    public $status = false;
}

```

Test WordPress site deployed in seconds? Yes, it's possible. Visit [qSandbox](#) for more info.

Disclaimer: The content in this post is for educational purposes only. Always remember to take a backup before doing any of the suggested steps just to be on the safe side.

Referral Note: When you purchase through an referral link (if any) on our site, we earn a commission.