

# STORE CONNECTION FLOW



# Contents

## 1. API Integration Strategy

Introduction	3
Authentication	5

## 2. Connection Flows

Hosted Carts	7
Open-Source Platforms	8

## 3. Tips and Usability Hygiene

Improve Your UX Design	10
Best Authentication Practices	11

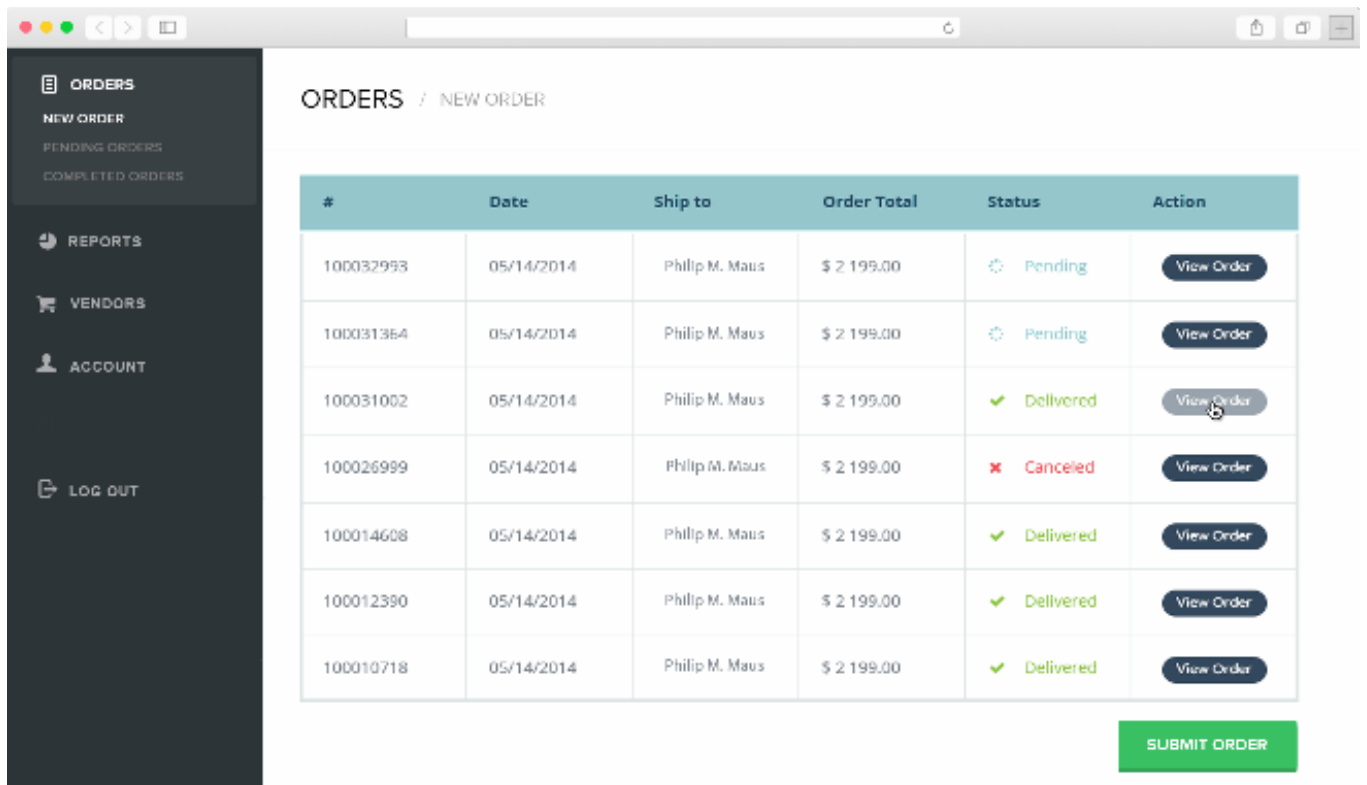
## 4. About API2Cart

12

# API Integration Strategy

## Introduction

Before you start, let's use the mock up below as an example of an order and shipping management application. The admin panel of this application includes common tabs such as Dashboard, Settings, Billing, etc. We will use this example within the pdf to show how a UI/UX may differentiate depending on a shopping cart and the way you want to integrate with them.



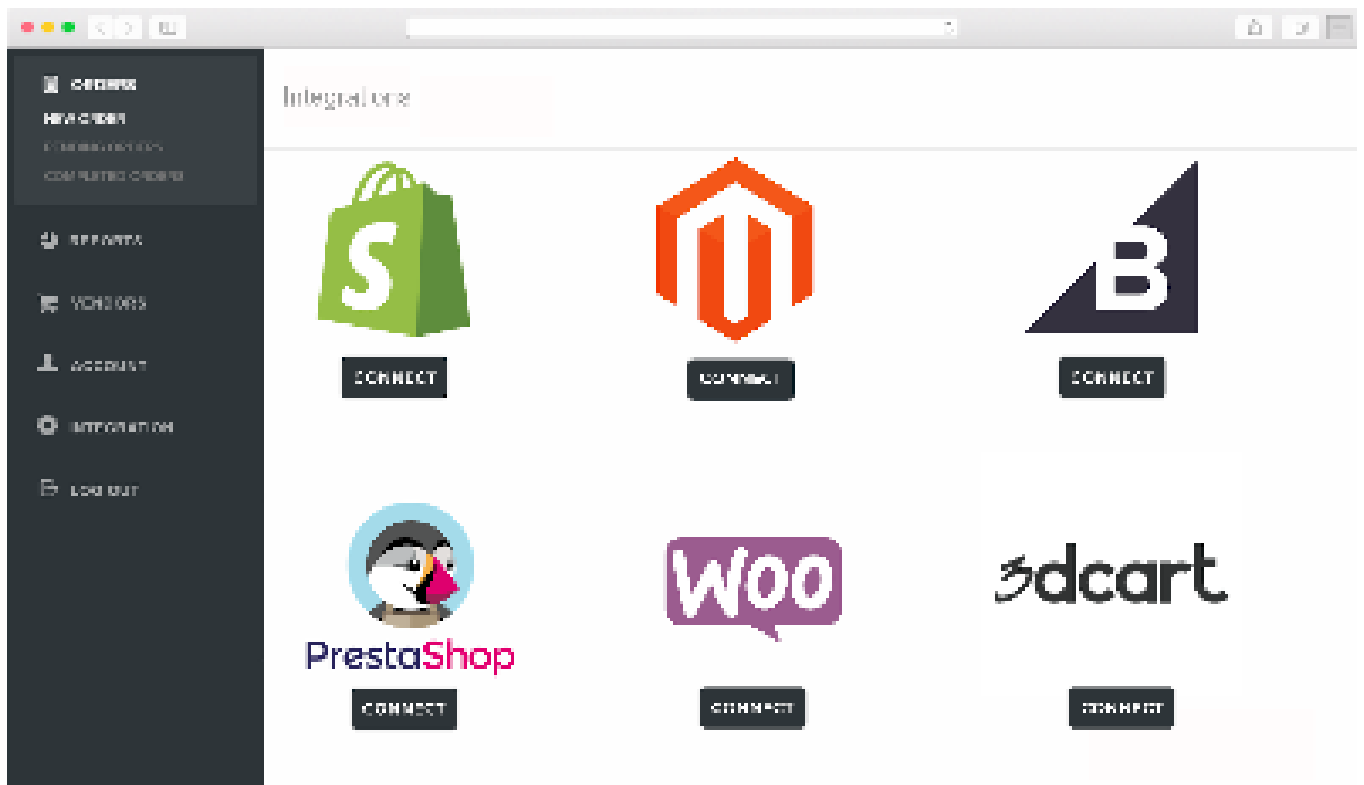
The screenshot shows a web application interface for managing orders. On the left is a dark sidebar with navigation options: ORDERS (with sub-items: NEW ORDER, PENDING ORDERS, COMPLETED ORDERS), REPORTS, VENDORS, ACCOUNT, and LOG OUT. The main content area is titled 'ORDERS / NEW ORDER' and contains a table of orders. Each row in the table has a unique order ID, a date (all 05/14/2014), a ship-to name (Philip M. Maus), an order total (\$ 2 199.00), a status (Pending, Delivered, or Canceled), and a 'View Order' button. A green 'SUBMIT ORDER' button is located at the bottom right of the table.

#	Date	Ship to	Order Total	Status	Action
100032993	05/14/2014	Philip M. Maus	\$ 2 199.00	Pending	View Order
100031354	05/14/2014	Philip M. Maus	\$ 2 199.00	Pending	View Order
100031002	05/14/2014	Philip M. Maus	\$ 2 199.00	Delivered	View Order
100026999	05/14/2014	Philip M. Maus	\$ 2 199.00	Canceled	View Order
100014608	05/14/2014	Philip M. Maus	\$ 2 199.00	Delivered	View Order
100012390	05/14/2014	Philip M. Maus	\$ 2 199.00	Delivered	View Order
100010718	05/14/2014	Philip M. Maus	\$ 2 199.00	Delivered	View Order

While this paper can showcase the setup of new integrations, it can also be used for optimizing existing integrations.

We suggest that you consider these steps below when planning on integrating with shopping carts:

- Determine the best UI options within your application
- Determine the types of authentication that your app will use
- Check the possibility of creating plugins for connecting web stores with your application



**Note:** When they click the "Integration" tab, they can see the list of all supported platforms.

# Authentication

Since shopping carts use different authentication mechanisms, they require different connection workflows between your app and shopping carts.

There are 2 authentication mechanisms you can use to connect an online store to your application:

## AUTHENTICATION MECHANISMS

**Basic access authentication** — A widely used protocol for simple username/password authentication. This type of mechanism provides no confidentiality protection for the transmitted credentials

**OAuth** — An open standard for authorization, commonly used as a way for users to authorize websites or applications to access their information on other websites but without giving them the passwords.



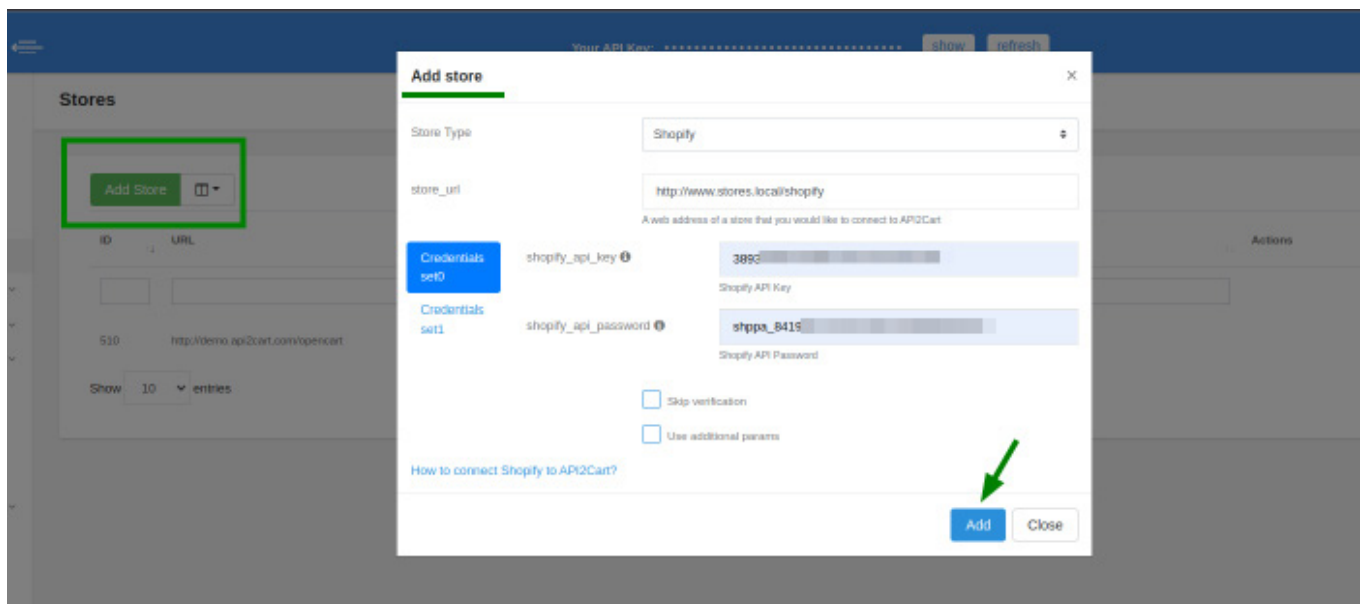
# Connection Flows

There are two ways of adding a store:

- manually
- via API call

## Manually

To connect store **manually** you will need to fill out the following basic fields in your API2Cart account: *store URL, store type, API key/Token or store key*. Credentials depend on the chosen store type. See the full list of requirements for different platforms [here](#).



You can use this way while you are testing the integration. For the production, we recommend to do it automatically via API call.

## Via API call

The second way is **by executing [account.cart.add](#) method** with the same credentials described above. You can send a request via Postman or visit our [Interactive Docs](#).

Also, the process of connecting a store **depends on a type of eCommerce platform**:

- based on a **hosted platform** (Shopify, BigCommerce, Etsy, eBay, Amazon)
- based on an **open-source platform** (Magento, WooCommerce, Prestashop)

See the list of hosted and open-source platforms [here](#).

# Hosted Cart

You can add a store based on **the hosted platform** in two ways:

- Basic authentication method
- OAuth authentication

## Basic authentication method

Let's see how the connection with the **Basic Authentication** method will work:

1. A web store owner logs in to your application account.
2. Goes to the Integration page and chooses the needed platform.
3. Fills in required fields like API path and API key (please note that fields depending on the shopping cart).
4. Clicks the Connect button.
5. The connection between the web store and your system is established.

## OAuth authentication

**OAuth authentication** does not require username/password details but wants access tokens instead.

First of all, you need to create your app and place it on the App Marketplace of the needed platform. Your customers will be able to install this application in their store directly from the marketplace. In its turn, the store owner needs to approve the application to get an access token.

Finally, you use this access token in [account.cart.add](#) method to connect the store.

# Open-Source Platform

To connect stores that are based on **open-source shopping platforms** like Magento, WooCommerce, PrestaShop, and CS-Cart, and others we use the Connection Bridge file.

The connection Bridge file is being used to retrieve information from your shopping cart via API and should be installed into the server root folder.

There are two ways you can install Bridge file on the store:

- manually
- automatically via plugins

## Manually

**Manually** download the Bridge file from your API2Cart account and provide it to your customers. Next, they will need to place it into the root folder of the store.

If needed, a bridge file can be stored in any folder on the server, but in that case, you'll need to use parameters *bridge\_url* (the full http url for a bridge) and *store\_root* (absolute server path to the store root folder).

If the bridge folder is contained in the store root folder, then you don't need to specify *store\_root* parameter, only *bridge\_url*.

After uploading the bridge, use [account.cart.add](#) method.

Here is the example of request:

```
https://api.api2cart.com/v1.1/account.cart.add.json?api_key=XXXXXXXXXX  
&cart_id=Opencart14 &store_url=http://mystore.com &store_key=XXXXXXXXXX
```



## Automatically via Plugins

The **API2Cart plugins** are intended to make it easier to add stores based on open-source shopping carts like Magento, WooCommerce, PrestaShop, and CS-Cart. They automatically install a connection bridge to the root folder of the store thus excluding the inconveniences that the process usually entails.

You can download needed plugins from your API2Cart account. The plugins are organized blank and white label, which means that you can modify them with your logos and additional specified functionality to present as a native code module. You can also list and promote them on relevant directories and marketplaces.

How the API2Cart plugins work:

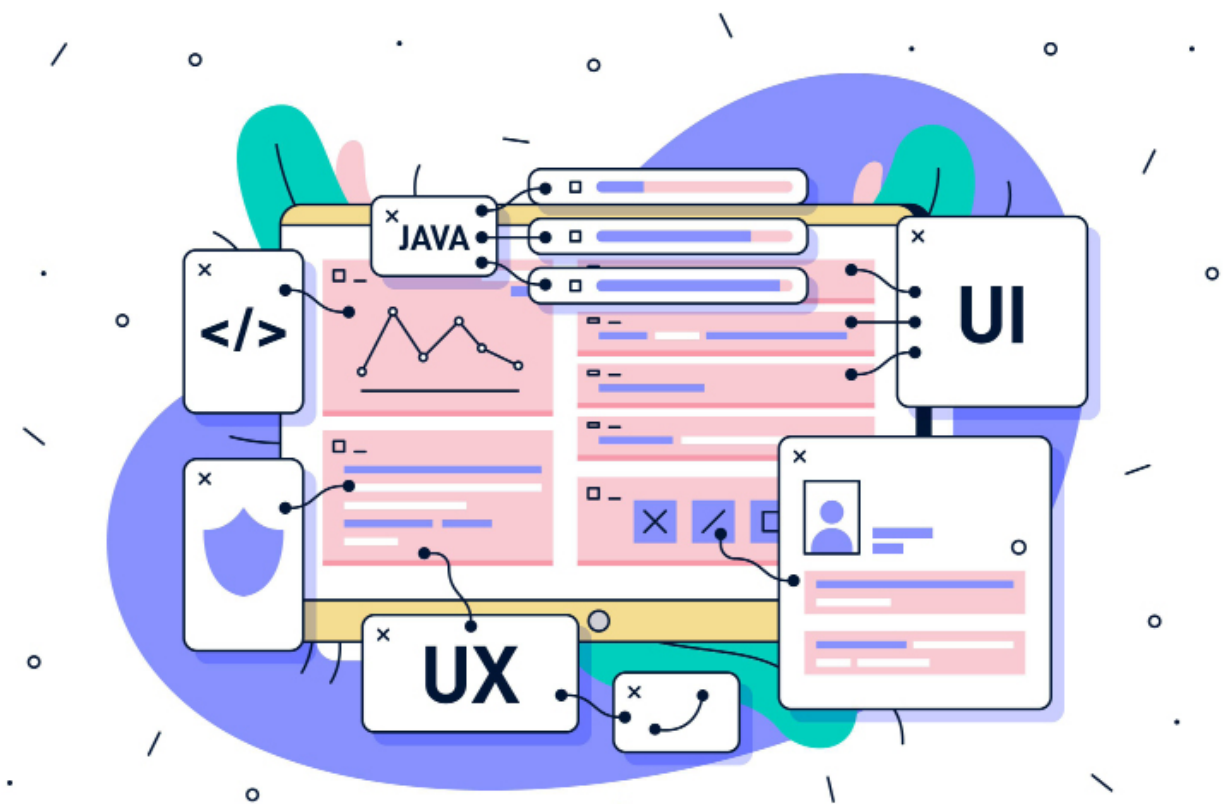
1. A store owner installs the plugin.
2. Press **Connect** and the system automatically generates a store key.
3. The store owner passes the key to the software vendor.
4. The vendor system identifies the user account and sends an *account.cart.add* API call to API2Cart.
5. The store is connected.

# Tips and Usability Hygiene

## Improve Your UX Design

This section focuses on usability hygiene and the basics, which help to make the process of connecting the web store with your application more clear.

1. Use the newest authentication methods if possible.
2. Add a short description of what the integration with a shopping cart does so that users could understand what this connection can do for them. For example, "auto import tracking numbers and order info from Shopify store every 4 hours", etc.
3. Add a short tutorial on how to connect a web store with your application. For example, a "How to setup" video or "Step-by-step guide" could be a great option.
4. Think about "How to..." FAQs or knowledge base pages. For instance, "How to install the Magento plugin", "How to connect my Shopify store", etc.
5. Link to some relevant articles to cover the most important questions.
6. When they enter incorrect access credentials, make sure that they receive a warning message or a pop-up window that tells them about an error.



# The Best Authentication Practices

To optimize your integration experience and conduce a flawless UX for your customers, we recommend that you should do the following:

1. Connect hosted platforms (Shopify, Bigcommerce) using OAuth apps.
2. Connect all other hosted platforms using the Basic Authentication method.
3. Connect the most popular open-source platforms (Magento, WooCommerce, PrestaShop) using our ready-to-use blank plugins.
4. Connect all other (or less important for you) open-source platforms asking your clients to download and install a bridge file themselves.

Following these recommendations, you will surely provide the best user and integration experience for your clients. If you have more questions about the best integration practices, feel free to [contact us](#).

TRY IT FOR FREE



# About API2Cart

**API2Cart** provides a unified API to integrate with multiple eCommerce platforms at once. It allows to connect with stores based on any of the supported platforms and work with all the necessary store data for further processing. Integration via the unified shopping cart data interface helps to optimize business logic, save time and resources that separate integrations would take, and invest them in other business issues.

API2Cart offers 100+ API methods to let you operate and manipulate shopping cart store data as you need it for business. It ensures secure data transfer with the help of an SSL certificate and 32-symbol API key. To make your integration progress easier, the service provides detailed documentation with code samples and interactive examples. Plus, full tech support will help you solve any issues.

DOCUMENTATION

SUPPORTED PLATFORMS

## Contact us

WEBSITE: <https://api2cart.com/>

EMAIL: [manager@api2cart.com](mailto:manager@api2cart.com)

PHONE: 1-800-224-0976 (USA&Canada Toll-Free)



**API2CART**  
unified shopping cart data interface