



A Guide to Liv-ex Software Development Kit (SDK)

Document revision: 1.0
Date of Issue: 9 May 2018
Date of revision:

Contents

1. Overview	3
2. What you can do with the Liv-ex SDK	3
3. The Liv-ex SDK components	3
4. Getting started with the Liv-ex SDK	3
4.1 Obtain access key from Liv-ex	4
4.2 Check prerequisites	4
4.3 Configure the SDK script in your webpage	4
5. Example	5
6. Appendix	8

1. Overview

The Liv-ex Software Development Kit (SDK) is a convenient way to integrate Liv-ex services within your website and web-based applications, without the need for any back-end programming.

This guide explains what you can do with the Liv-ex SDK, its main components, and will show you how to get started to configure the SDK from an example to view the Liv-ex Exchange on your webpage. Further documents explaining each of the Liv-ex SDK in more details are available.

2. What you can do with the Liv-ex SDK

With the Liv-ex SDK, you can write JavaScript applications to programmatically interact with Liv-ex services. The Liv-ex SDK provides a rich set of client-side functionality that allows you to do the following from your website:

- View activity from the Liv-ex Exchange
- Search for wines
- Get active markets of wines on the Liv-ex Exchange.
- Place bids and offers on wines

3. The Liv-ex SDK components

The Liv-ex SDK supports client-side JavaScript to integrate into a web page. The SDK also includes several third-party libraries (Appendix A), which are included for functionality such as cross-domain communication and other features.

The Liv-ex SDK is divided into two areas:

- The **Core SDK** allows you to easily interact with Liv-ex APIs. It facilitates the return of data in JSON format so that you can build your own UI easily around it.
- The **UI SDK** includes Liv-ex UI components so that you can provide rich and engaging material to by simply integrating and configuring it into your web page.

Currently the following Core methods and UI SDKs are available for:

- **The Market**, that allows you to display a Liv-ex Exchange grid based on recent market activities on the Liv-ex platform.
- **Wine search**, which allows you to integrate wine search into your application with pricing in the result of the search.
- **The Liv-ex grid**, gives flexibility to the end developer to configure a range of Liv-ex API services to form a Liv-ex grid.

4. Getting started with the Liv-ex SDK

Getting started with the Liv-ex SDK is easy:

1. Obtain Access key from Liv-ex.
2. Check prerequisites
3. Configure SDK script in your application.

Once you have familiarized yourself with the information in this guide, you will be ready to use the Liv-ex SDK to interact with the Liv-ex platform.

4.1 Obtain access key from Liv-ex

You should be an active member of Liv-ex. Please ask your Account manager for the access key. You will get Client key and Client secret along with API access rights which are used to authenticate/authorise members to use Liv-ex services from the SDK.

4.2 Check prerequisites

There are no specific tools and tech required. You should be familiar with the basics of HTML and JavaScript.

Compatible browser

We've tested the Liv-ex SDK with the latest versions of the following browsers:

- Google Chrome
- Mozilla Firefox
- Apple Safari
- Microsoft Internet Explorer 11
- Microsoft Edge

JSON support

When running in a web browser, the SDK's only dependency is JSON support. If your site supports older browsers, you might need to include JSON handling for compatibility. To learn more, see the [JSON website](#).

4.3 Configure the SDK script in your webpage

The Liv-ex SDK doesn't have any standalone files that need to be downloaded or installed and hosted locally with your project, instead you simply need to include a short piece of regular JavaScript in your HTML that will asynchronously load the SDK into your pages. In order to use functionality available in the SDK, you'll need to include that library in your webpage. It is a script file that is pulled in real-time from the Liv-ex location. The async load means that it does not block the loading of other elements of your page.

The libraries are available at <https://app.liv-ex.com/sdk/>. The Liv-ex SDK library files are classified for the different Liv-ex SDK methods, which can be referred to from the SDK's specific documentation.

The example below is one of the uses of the Liv-ex SDK showing how to use the UI SDK library to add a visual component to your web page.

5. Example

The example below shows you how to integrate one of the Liv-ex Grids into your web page using the SDK UI component.

The following snippet of code will give the basic version of the SDK where the options are set to their most common defaults. You should insert it directly after the opening `<body>` tag on each page you want to load it on:

```
<div id="container"></div>
<script src="http://use.fontawesome.com/981042b3e9.js"></script>
<script src="https://app.liv-ex.com/sdk/livex.ui.the-liv-ex-grid.js"></script>
<script>
  var TheLivexGrid = LIVX.UI.TheLivexGrid;
  TheLivexGrid.connect('8cd6b4be-828e-4c44-b903-7e3b6b9d3f56', 'xxxxxxx');
  TheLivexGrid.draw(document.getElementById('container'), {
    lwinSource:'bestBidOffer',
    priceSource:'activeMarket',
    currency:'GBP',
    limit:25,
  });
</script>
```

This code will load and initialize the SDK. You must replace the values in the Connect method with the Client key and Client secret provided by Liv-ex for access.

Understanding above code example

Create a `<div>` element with id attribute inside your page where you want to include the visual output of the component. Id value should be unique across the page. You need to use this id attribute value while configuring the SDK code.

```
<div id="container"></div>
```

Import the SDK library file from the Liv-ex location.

```
<script src="https://app.liv-ex.com/sdk/livex.ui.the-liv-ex-grid.js"></script>
```

Configuring and making request to Liv-ex services via UI SDK code is a 3 step process. You need to write code inside the `<script>` element.

Step 1

Create a local object of the UI component.

```
var TheLivexGrid = LIVX.UI.TheLivexGrid;
```

Step 2

Initialize the local object with `Connect` method by passing Client key and Client secret provided by us.

```
TheLivexGrid.connect('<CLIENT_KEY>', '<CLIENT_SECRET>');
```

Step 3

Call the `draw` method with the required parameters. The first parameter of the method is the Element which you created in step 1 and the second parameter in this example is talking about feed detail (feed created on liv-ex data-feed portal) that you want to display prices for wines from the Liv-ex market place.

```
TheLivexGrid.draw(document.getElementById('container'), {
  lwinSource:<lwinsourceservicename>,
  priceSource:<pricingsourceservicename>,
  currency:<currency>,
  limit: <Number of rows>,
});
```

Finally the output will look as below for desktop and mobile resolutions for the above example code.

Wine	Bid	Offer	Last Trade	Market
Pontet Canet, 2010 > 12x75, SIB	1,590 1 Make Offer	1,650 3 Place Bid	1,550 19/03/18	1,590
Lafite Rothschild, 2012 > 12x75, Special - Condition ⓘ @	5,529 — Make Offer	5,529 2 Place Bid	4,510 14/11/17	4,750
Montrose, 2010 > 12x75, SIB	— — Make Offer	1,704 1 Place Bid	1,700 21/03/18	2,000
Mouton Rothschild, 2014 > 12x75, SIB	3,500 1 Make Offer	— — Place Bid	4,000 23/03/18	3,584
Cheval Blanc, 2005 > 12x75, SIB	5,800 2 Make Offer	5,904 3 Place Bid	5,900 22/03/18	6,600
Latour, 2005 > 12x75, SIB	1,380 2 Make Offer	6,536 1 Place Bid	6,430 01/03/18	6,900
Lafite Rothschild, 2011 > 12x75, SIB	3,800 2 Make Offer	4,000 1 Place Bid	4,570 07/03/18	4,800

Pontet Canet, 2010 >			
12x75, SIB			
Bid	Offer	Last Trade	Market
1,590 1 Make Offer	1,650 3 Place Bid	1,550 19/03/18	1,590

Lafite Rothschild, 2012 >			
12x75, Special - Condition ⓘ ⓘ			
Bid	Offer	Last Trade	Market
5,529 — Make Offer	5,529 2 Place Bid	4,510 14/11/17	4,750

You will find further details about each of the Liv-ex SDKs, their usage and how to customize your implementation in further documents.

6. Appendix

Appendix A:

The following is a list of third party libraries used by Liv-ex SDK for JavaScript:

- react: 15.3.0
- redux: ^3.7.2
- bulma: ^0.6.1
- aws4: ^1.6.0
- babel-polyfill: ^6.26.0
- font-awesome: ^4.7.0
- i18next: ^8.4.3
- moment: ^2.18.1
- eslint: 3.2.2
- jasmine: ^2.2.1
- karma: ^0.13.2
- webpack: ^3.5.5