



THE FINE WINE MARKET

## LWIN: LWIN View API v1

Document Revision 1.0  
Date of Issue: 09 March 2020  
Date of revision: 20 January 2021

Nick Palmer

Product Lead

## Table of Contents

<b>1. Purpose .....</b>	<b>3</b>
<b>2. Glossary of Terms .....</b>	<b>3</b>
<b>3. Technical Standards .....</b>	<b>3</b>
<b>4. Request Header .....</b>	<b>3</b>
<b>5. Listing .....</b>	<b>5</b>
5.1 LWIN View service (POST method).....	5
<b>6. Response Codes.....</b>	<b>16</b>
6.1 Request validation error codes .....	16
6.2 HTTP Status codes .....	16

## 1. Purpose

To provide the API end point information and examples of the web services available for LWIN View.

## 2. Glossary of Terms

Term	Meaning
<b>LWIN</b>	LWIN - the Liv-ex Wine Identification Number – serves as a universal wine identifier for the wine trade. LWIN is a unique seven to eighteen-digit numerical code that can be used to quickly and accurately identify a product. LWIN allows wine companies to keep their preferred naming system, while introducing a new universal code. This document refers to LWIN7 (7-digit, wine) and LWIN11 (11-digit, wine + vintage) codes only.
<b>Combine</b>	When two LWINs for the same wine have been erroneously created codes are "combined". This preserves the overall data of the product. The dominant LWIN is known as the "leader", while the LWIN that has been merged away is known as the "follower".

## 3. Technical Standards

- Permitted users will be issued with a unique token (CLIENT\_KEY) and password (CLIENT\_SECRET) combination to control the access for all the web services covered under Exchange Integration.
- The web services will consume and produce both XML and JSON. The user can provide the content type in the request header. If the user does not provide any information, then the default content type will be JSON.
- The project will support ISO 8601.
- The project will only support HTTPS protocol for client and server communications.
- The API will support the following methods:
  - POST for create operation
- Pretty printing for output readability only is supported if required
- Compression for bandwidth savings are used
- Authentication mechanism will be custom based on CLIENT\_KEY and CLIENT\_SECRET
- The APIs will be accessible at <https://api.liv-ex.com/> followed by their specific base URIs

## 4. Request Header

This information will be used to authenticate valid access to the REST API. Each user will have to provide the following information in the request header.

### Param

Name	Mandatory	Description
CLIENT_KEY	Y	A valid merchant GUID which will be unique for each user.
CLIENT_SECRET	Y	Password/Secret for the users CLIENT_KEY.

ACCEPT	Y	Accept header is a way for a client to specify the media type of the response content it is expecting. The values for the content type will be application/json or application/xml.  If no/ invalid content type is found in the request, then JSON format will be used by default.
CONTENT-TYPE	Y for POST requests	Content-type is a way to specify the media type of request being sent from the client to the server. The values for the content type will be application/json or application/xml.  If no/ invalid content type is found in the request, then JSON format will be used by default.

e.g.

CLIENT\_KEY: 94B5CC70-BC3D-49C3-B636-C3C7552E543D

CLIENT\_SECRET: merchantpasswd

ACCEPT: application/json

CONTENT-TYPE: application/json

#### Invalid header JSON response

```
{
  "status": "Unauthorized",
  "statusCode": "401",
  "message": "Request was unsuccessful",
  "livexCode": "R000"
  "apiInfo": {
    "version": "1.0",
    "timestamp": 1518524979121,
    "provider": "Liv-ex"
  }
}
```

#### Invalid header XML response

```
<Response>
  <Status>Unauthorized</Status>
  <HttpCode>401</Code>
  <Message>Request was unsuccessful.</Message>
  <LivexCode>R001</LivexCode>
  <ApiInfo>
    <Version>1.0</Version>
    <Timestamp>2019-11-04T11:12:30</Timestamp>
    <Provider>Liv-ex</Provider>
  </ApiInfo>
</Response>
```

## 5. Listing

### 5.1 LWIN View service (POST method)

#### Description

This service will be used to view the metadata associated with a given LWIN code. It supports querying at both the LWIN7 (wine) and LWIN11 (vintage specific) level.

#### Base URI

[lwin/view/v1/lwinView](#)

#### URI Pagination Parameters

Name	Mandatory	Description
?limit	N default = 50	Values: Any value between '1' and '50' <b>Type:</b> integer
?offset	N default = 1	<b>Type:</b> alphanumeric

Example base URI including pagination: [lwin/view/v1/lwinView?limit=25&offset=1](#)

#### Request Parameters

Name	Mandatory	Description
lwin	Y	LWIN7 or LWIN11 code to be queried. Only one value permitted per request <b>Type:</b> integer (7/11)
includeVintageListing	N Default = false	Active for LWIN7 request only. When set to true returns all LWIN11 metadata for valid vintage values <b>Type:</b> Boolean true/false

#### Sample Request Body

##### JSON

```
{
  "lwin": "1234567",
  "includeVintageListing": false
}
```

##### XML

```
<lwinView>
  <lwin>123467</lwin>
  <includeVintageListing>false</includeVintageListing>
</lwinView>
```

#### Sample Response Body

The LWIN View service returns all the metadata associated with an LWIN code. Metadata can be queried at the wine (LWIN7) and vintage specific (LWIN11) level.

By setting the “includeVintageListing” attribute to true when calling an LWIN7 code, the service will return the entire listing of vintage specific metadata for a given wine (e.g. all historic namings for the wine named “Bahans Haut Brion”). Note that this type of request may return 100+ items depending on the LWIN code requested.

**Full vintage listings for deleted LWIN7s**

It is not possible to request a full listing of LWIN11 metadata for LWINs that have been deleted. The API service will return a null response with a totalResults count of zero.

**Response pagination parameters**

Name	Description
totalResults	A count of the number of records (distinct LWIN events) contained in the response <b>Type:</b> integer
limit	The number of records (distinct LWIN events) contained in this specific response payload <b>Type:</b> integer
offset	The position in the dataset of this set of records. Values start at 1. <b>Type:</b> integer

**Response parameters**

Name	Description
lwin	LWIN7 or LWIN11 code depending on the type of the query <b>Type:</b> integer(7 11)
producerTitle	Title of producer or owner of wine <b>Type:</b> alphanumeric
producerName	Producer or owner of wine <b>Type:</b> alphanumeric
wine	Name of wine (brand and/or grape and/or technical term) <b>Type:</b> alphanumeric
country	Country of origin <b>Type:</b> alphanumeric
region	Region of origin (where applicable, specific to local laws) <b>Type:</b> alphanumeric
subRegion	Sub-region of origin (where applicable, specific to local laws) <b>Type:</b> alphanumeric
site	Site within sub-region (where applicable, specific to local laws) <b>Type:</b> alphanumeric
parcel	Parcel within site (where applicable, specific to local laws) <b>Type:</b> alphanumeric
colour	Colour of LWIN product <b>Type:</b> alphanumeric

	<b>Values:</b> "white", "red", "rose", null
type	LWIN beverage type <b>Type:</b> alphanumeric
subType	Subcategory of LWIN type <b>Type:</b> alphanumeric
designation	Officially assigned status (specific to local laws) <b>Type:</b> alphanumeric
classification	Officially declared quality level (where applicable, specific to local laws) <b>Type:</b> alphanumeric
vintageConfiguration	Vintage pattern of the wine (e.g. single vintage only, production ended) Not returned for LWIN11 <b>Type:</b> alphanumeric <b>Values:</b> "sequential", "nonSequential", "singleVintageOnly"
vintageValues	<b>Type:</b> array, integer(4)
firstVintage	The first vintage of the LWIN7. Not returned for LWIN11 <b>Type:</b> integer(4)
finalVintage	The final vintage of the LWIN7. Not returned for LWIN11 <b>Type:</b> integer(4)
childOf	Parent wine LWIN7. This should be used to map associated wines to one another. This is a separate relationship to LWIN merge references Not returned for LWIN11 <b>Type:</b> integer (7)
displayNameType	One out of 8 display name concatenation formulas. Assigned by LWIN administrator <b>Type:</b> alphanumeric
displayName	<b>Type:</b> alphanumeric
status	<b>Type:</b> alphanumeric <b>Values:</b> "live", "deleted", "combined"
combineReference	The new leader LWIN for the LWIN that has been combined <b>Type:</b> integer(7)
dateCreated	<b>Type:</b> datetime / epoch
lastUpdateDate	<b>Type:</b> datetime / epoch

### JSON Response

The response is sent per request. The examples below show responses to an LWIN7 request, LWIN11 request and an LWIN7 full vintage listing request.

LWIN7 response	LWIN11 response
<pre>{   "status": "OK",   "httpCode": "200",   "message": "Request completed successfully",</pre>	<pre>{   "status": "OK",   "httpCode": "200",   "message": "Request completed successfully",</pre>

<pre> "internalErrorCode": "R001", "apiInfo": {   "version": "1.0",   "timestamp": 1589550146943,   "provider": "Liv-ex" }, "pageInfo": {   "totalResults": 1,   "limit": 50,   "offset": 1 }, "lwinView": [   {     "lwin": "1002425",     "producerTitle": null,     "producerName": "John Glaetzer",     "wine": "John's Blend Cabernet Sauvignon N°41",     "country": "Australia",     "region": "South Australia",     "subRegion": "Barossa Valley",     "site": null,     "parcel": null,     "colour": "Red",     "type": "Wine",     "subType": "Still",     "designation": "GI",     "classification": null,     "vintageConfiguration": "sequential",     "vintageValues": [       "2015",       "2014",       "2013",       "2012",       "2010",       "2009",       "2008",       "2007",       "2006",       "2005",       "2004",       "2003",       "2002",       "2001",       "2000",       "1999",       "1996",       "1995"     ],     "firstVintage": null,     "finalVintage": null,     "childOf": null,     "displayNameType": "Type 4",     "displayName": "John Glaetzer, John's Blend Cabernet Sauvignon N°41, Barossa Valley",     "status": "live",     "combineReference": null,     "dateCreated": 1570836860000,     "lastUpdateDate": 1589381521000   } ], "errors": null </pre>	<pre> "internalErrorCode": "R001", "apiInfo": {   "version": "1.0",   "timestamp": 1589550285067,   "provider": "Liv-ex" }, "pageInfo": {   "totalResults": 1,   "limit": 50,   "offset": 1 }, "lwinView": [   {     "lwin": "10024252009",     "producerTitle": null,     "producerName": "John Glaetzer",     "wine": "John's Blend Cabernet Sauvignon N°36",     "country": "Australia",     "region": "South Australia",     "subRegion": "Barossa Valley",     "site": null,     "parcel": null,     "colour": "Red",     "type": "Wine",     "subType": "Still",     "designation": "GI",     "classification": null,     "vintageConfiguration": null,     "vintageValues": [       "2009"     ],     "displayNameType": "Type 4",     "displayName": "John Glaetzer, John's Blend Cabernet Sauvignon N°36, Barossa Valley",     "status": "live",     "combineReference": null,     "dateCreated": 1570836860000,     "lastUpdateDate": 1587801307000   } ], "errors": null </pre>
---	--



```
}
}
```

**LWIN7 with includeVintageListing = true**

```
{
  "status": "OK",
  "statusCode": "200",
  "message": "Request completed successfully",
  "internalErrorCode": "R001",
  "apiInfo": {
    "version": "1.0",
    "timestamp": 1589550351346,
    "provider": "Liv-ex"
  },
  "pageInfo": {
    "totalResults": 18,
    "limit": 50,
    "offset": 1
  },
  "lwinView": [
    {
      "lwin": "10024252015",
      "producerTitle": null,
      "producerName": "John Glaetzer",
      "wine": "John's Blend Cabernet Sauvignon N°41",
      "country": "Australia",
      "region": "South Australia",
      "subRegion": "Barossa Valley",
      "site": null,
      "parcel": null,
      "colour": "Red",
      "type": "Wine",
      "subType": "Still",
      "designation": "GI",
      "classification": null,
      "vintageConfiguration": null,
      "vintageValues": [
        "2015"
      ],
      "displayNameType": "Type 4",
      "displayName": "John Glaetzer, John's Blend Cabernet Sauvignon N°41,
Barossa Valley",
      "status": "live",
      "combineReference": null,
      "dateCreated": 1589381489000,
      "lastUpdateDate": 1589381521000
    },
    {
      "lwin": "10024252014",
      "producerTitle": null,
      "producerName": "John Glaetzer",
      "wine": "John's Blend Cabernet Sauvignon N°40",
      "country": "Australia",
      "region": "South Australia",
      "subRegion": "Barossa Valley",
      "site": null,
      "parcel": null,
      "colour": "Red",
      "type": "Wine",
      "subType": "Still",
      "designation": "GI",
      "classification": null,
      "vintageConfiguration": null,
      "vintageValues": [
        "2014"
      ],
    }
  ]
}
```

```

        "displayNameType": "Type 4",
        "displayName": "John Glaetzer, John's Blend Cabernet Sauvignon N°40,
Barossa Valley",
        "status": "live",
        "combineReference": null,
        "dateCreated": 1570836860000,
        "lastUpdateDate": 1587801307000
    },
    {
        "lwin": "10024252013",
        "producerTitle": null,
        "producerName": "John Glaetzer",
        "wine": "John's Blend Cabernet Sauvignon N°39",
        "country": "Australia",
        "region": "South Australia",
        "subRegion": "Barossa Valley",
        "site": null,
        "parcel": null,
        "colour": "Red",
        "type": "Wine",
        "subType": "Still",
        "designation": "GI",
        "classification": null,
        "vintageConfiguration": null,
        "vintageValues": [
            "2013"
        ],
        "displayNameType": "Type 4",
        "displayName": "John Glaetzer, John's Blend Cabernet Sauvignon N°39,
Barossa Valley",
        "status": "live",
        "combineReference": null,
        "dateCreated": 1570836860000,
        "lastUpdateDate": 1587801307000
    },
    {
        "lwin": "10024252012",
        "producerTitle": null,
        "producerName": "John Glaetzer",
        "wine": "John's Blend Cabernet Sauvignon N°38",
        "country": "Australia",
        "region": "South Australia",
        "subRegion": "Barossa Valley",
        "site": null,
        "parcel": null,
        "colour": "Red",
        "type": "Wine",
        "subType": "Still",
        "designation": "GI",
        "classification": null,
        "vintageConfiguration": null,
        "vintageValues": [
            "2012"
        ],
        "displayNameType": "Type 4",
        "displayName": "John Glaetzer, John's Blend Cabernet Sauvignon N°38,
Barossa Valley",
        "status": "live",
        "combineReference": null,
        "dateCreated": 1570836860000,
        "lastUpdateDate": 1587801307000
    }
    ],
    "errors": null
}

```

**Invalid JSON response**

```
{
  "status": "OK",
  "statusCode": "200",
  "message": "Request completed successfully",
  "internalErrorCode": "R001",
  "apiInfo": {
    "version": "1.0",
    "timestamp": 1611152023388,
    "provider": "Liv-ex"
  },
  "pageInfo": {
    "totalResults": 0,
    "limit": 50,
    "offset": 1
  },
  "lwinView": {
    "lwin": "1012781198",
    "includeVintageListing": "false"
  },
  "errors": {
    "error": [
      {
        "code": "L002",
        "message": "Incorrect LWIN: 1012781198."
      }
    ]
  }
}
```

**XML Response**

The response is sent per request. The examples below show responses to an LWIN7 request, LWIN11 request and an LWIN7 full vintage listing request.

LWIN7 response	LWIN11 response
<pre>xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt; &lt;/lwinViewResponse&gt; &lt;?xml version="1.0" encoding="UTF-8" standalone="yes"?&gt; &lt;lwinViewResponse&gt;   &lt;Status&gt;OK&lt;/Status&gt;   &lt;HttpCode&gt;200&lt;/HttpCode&gt;   &lt;Message&gt;Request completed successfully&lt;/Message&gt;   &lt;InternalErrorCode&gt;R001&lt;/InternalErrorCode&gt;   &lt;ApiInfo&gt;     &lt;Version&gt;1.0&lt;/Version&gt;     &lt;Timestamp&gt;2020-05- 15T13:50:36.829Z&lt;/Timestamp&gt;     &lt;Provider&gt;Liv-ex&lt;/Provider&gt;   &lt;/ApiInfo&gt;   &lt;pageInfo&gt;     &lt;totalResults&gt;1&lt;/totalResults&gt;     &lt;limit&gt;50&lt;/limit&gt;     &lt;offset&gt;1&lt;/offset&gt;   &lt;/pageInfo&gt;   &lt;lwinView&gt;     &lt;view&gt;       &lt;lwin&gt;1002425&lt;/lwin&gt;</pre>	<pre>&lt;?xml version="1.0" encoding="UTF-8" standalone="yes"?&gt; &lt;lwinViewResponse&gt;   &lt;Status&gt;OK&lt;/Status&gt;   &lt;HttpCode&gt;200&lt;/HttpCode&gt;   &lt;Message&gt;Request completed successfully&lt;/Message&gt;    &lt;InternalErrorCode&gt;R001&lt;/InternalErro rCode&gt;   &lt;ApiInfo&gt;     &lt;Version&gt;1.0&lt;/Version&gt;     &lt;Timestamp&gt;2020-05- 15T13:52:26.295Z&lt;/Timestamp&gt;     &lt;Provider&gt;Liv-ex&lt;/Provider&gt;   &lt;/ApiInfo&gt;   &lt;pageInfo&gt;    &lt;totalResults&gt;1&lt;/totalResults&gt;   &lt;limit&gt;50&lt;/limit&gt;   &lt;offset&gt;1&lt;/offset&gt; &lt;/pageInfo&gt; &lt;lwinView&gt;   &lt;view&gt;     &lt;lwin&gt;10024252009&lt;/lwin&gt;</pre>

<pre> &lt;producerTitle xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt;   &lt;producerName&gt;John   Glaetzer&lt;/producerName&gt;   &lt;wine&gt;John's Blend Cabernet   Sauvignon N°41&lt;/wine&gt;   &lt;country&gt;Australia&lt;/country&gt;   &lt;region&gt;South Australia&lt;/region&gt;   &lt;subRegion&gt;Barossa   Valley&lt;/subRegion&gt;   &lt;site xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt;   &lt;parcel xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt;     &lt;colour&gt;Red&lt;/colour&gt;     &lt;type&gt;Wine&lt;/type&gt;     &lt;subType&gt;Still&lt;/subType&gt;     &lt;designation&gt;GI&lt;/designation&gt;     &lt;classification xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt; &lt;/classification&gt; &lt;vintageConfiguration&gt;sequential&lt;/vintageConfig uration&gt;   &lt;vintageValues&gt;     &lt;vintage&gt;2015&lt;/vintage&gt;     &lt;vintage&gt;2014&lt;/vintage&gt;     &lt;vintage&gt;2013&lt;/vintage&gt;     &lt;vintage&gt;2012&lt;/vintage&gt;     &lt;vintage&gt;2010&lt;/vintage&gt;     &lt;vintage&gt;2009&lt;/vintage&gt;     &lt;vintage&gt;2008&lt;/vintage&gt;     &lt;vintage&gt;2007&lt;/vintage&gt;     &lt;vintage&gt;2006&lt;/vintage&gt;     &lt;vintage&gt;2005&lt;/vintage&gt;     &lt;vintage&gt;2004&lt;/vintage&gt;     &lt;vintage&gt;2003&lt;/vintage&gt;     &lt;vintage&gt;2002&lt;/vintage&gt;     &lt;vintage&gt;2001&lt;/vintage&gt;     &lt;vintage&gt;2000&lt;/vintage&gt;     &lt;vintage&gt;1999&lt;/vintage&gt;     &lt;vintage&gt;1996&lt;/vintage&gt;     &lt;vintage&gt;1995&lt;/vintage&gt;   &lt;/vintageValues&gt;   &lt;firstVintage xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt;   &lt;finalVintage xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt;   &lt;childOf xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt;     &lt;displayNameType&gt;Type   4&lt;/displayNameType&gt;     &lt;displayName&gt;John Glaetzer, John's   Blend Cabernet Sauvignon N°41, Barossa   Valley&lt;/displayName&gt;     &lt;status&gt;live&lt;/status&gt;     &lt;combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt; </pre>	<pre> &lt;producerTitle xmlns:xsi="http://www.w3.org/2001/XML Schema-instance" xsi:nil="true"/&gt;   &lt;producerName&gt;John   Glaetzer&lt;/producerName&gt;   &lt;wine&gt;John's Blend   Cabernet Sauvignon N°36&lt;/wine&gt;   &lt;country&gt;Australia&lt;/country&gt;   &lt;region&gt;South   Australia&lt;/region&gt;   &lt;subRegion&gt;Barossa   Valley&lt;/subRegion&gt;   &lt;site xmlns:xsi="http://www.w3.org/2001/XML Schema-instance" xsi:nil="true"/&gt;   &lt;parcel xmlns:xsi="http://www.w3.org/2001/XML Schema-instance" xsi:nil="true"/&gt;     &lt;colour&gt;Red&lt;/colour&gt;     &lt;type&gt;Wine&lt;/type&gt;     &lt;subType&gt;Still&lt;/subType&gt; &lt;/subType&gt;     &lt;designation&gt;GI&lt;/designation&gt;     &lt;classification xmlns:xsi="http://www.w3.org/2001/XML Schema-instance" xsi:nil="true"/&gt; &lt;/classification&gt;   &lt;vintageValues&gt;     &lt;vintage&gt;2009&lt;/vintage&gt;   &lt;/vintageValues&gt;   &lt;displayNameType&gt;Type   4&lt;/displayNameType&gt;   &lt;displayName&gt;John   Glaetzer, John's Blend Cabernet   Sauvignon N°36, Barossa   Valley&lt;/displayName&gt;   &lt;status&gt;live&lt;/status&gt;   &lt;combineReference xmlns:xsi="http://www.w3.org/2001/XML Schema-instance" xsi:nil="true"/&gt;   &lt;dateCreated&gt;2019-10-   11T23:34:20Z&lt;/dateCreated&gt;   &lt;lastUpdateDate&gt;2020-04-   25T07:55:07Z&lt;/lastUpdateDate&gt; &lt;/view&gt; &lt;/lwinView&gt; &lt;errors xmlns:xsi="http://www.w3.org/2001/XML Schema-instance" xsi:nil="true"/&gt; &lt;/lwinViewResponse&gt; </pre>
--	---

<pre> &lt;dateCreated&gt;2019-10- 11T23:34:20Z&lt;/dateCreated&gt; &lt;lastUpdateDate&gt;2020-05- 13T14:52:01Z&lt;/lastUpdateDate&gt; &lt;/view&gt; &lt;/lwinView&gt; &lt;errors&gt; xmlns:xsi="http://www.w3.org/2001/XMLSchema- instance" xsi:nil="true"/&gt; &lt;/lwinViewResponse&gt; </pre>	
--	--

<p><b>LWIN7 with includeVintageListing = true</b></p> <pre> &lt;?xml version="1.0" encoding="UTF-8" standalone="yes"?&gt; &lt;lwinViewResponse&gt;   &lt;Status&gt;OK&lt;/Status&gt;   &lt;HttpCode&gt;200&lt;/HttpCode&gt;   &lt;Message&gt;Request completed successfully&lt;/Message&gt;   &lt;InternalErrorCode&gt;R001&lt;/InternalErrorCode&gt;   &lt;ApiInfo&gt;     &lt;Version&gt;1.0&lt;/Version&gt;     &lt;Timestamp&gt;2020-05-15T13:53:28.117Z&lt;/Timestamp&gt;     &lt;Provider&gt;Liv-ex&lt;/Provider&gt;   &lt;/ApiInfo&gt;   &lt;pageInfo&gt;     &lt;totalResults&gt;18&lt;/totalResults&gt;     &lt;limit&gt;50&lt;/limit&gt;     &lt;offset&gt;1&lt;/offset&gt;   &lt;/pageInfo&gt;   &lt;lwinView&gt;     &lt;view&gt;       &lt;lwin&gt;10024252015&lt;/lwin&gt;       &lt;producerTitle xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/&gt;       &lt;producerName&gt;John Glaetzer&lt;/producerName&gt;       &lt;wine&gt;John's Blend Cabernet Sauvignon N°41&lt;/wine&gt;       &lt;country&gt;Australia&lt;/country&gt;       &lt;region&gt;South Australia&lt;/region&gt;       &lt;subRegion&gt;Barossa Valley&lt;/subRegion&gt;       &lt;site xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/&gt;       &lt;parcel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/&gt;       &lt;colour&gt;Red&lt;/colour&gt;       &lt;type&gt;Wine&lt;/type&gt;       &lt;subType&gt;Still&lt;/subType&gt;       &lt;designation&gt;GI&lt;/designation&gt;       &lt;classification xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/&gt;       &lt;vintageValues&gt;         &lt;vintage&gt;2015&lt;/vintage&gt;       &lt;/vintageValues&gt;       &lt;displayNameType&gt;Type 4&lt;/displayNameType&gt;       &lt;displayName&gt;John Glaetzer, John's Blend Cabernet Sauvignon N°41, Barossa Valley&lt;/displayName&gt;       &lt;status&gt;live&lt;/status&gt;       &lt;combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/&gt;       &lt;dateCreated&gt;2020-05-13T14:51:29Z&lt;/dateCreated&gt;       &lt;lastUpdateDate&gt;2020-05-13T14:52:01Z&lt;/lastUpdateDate&gt;     &lt;/view&gt;     &lt;view&gt;       &lt;lwin&gt;10024252014&lt;/lwin&gt;       &lt;producerTitle xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/&gt;       &lt;producerName&gt;John Glaetzer&lt;/producerName&gt; </pre>
---

```

        <wine>John's Blend Cabernet Sauvignon N°40</wine>
        <country>Australia</country>
        <region>South Australia</region>
        <subRegion>Barossa Valley</subRegion>
        <site xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <parcel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <colour>Red</colour>
        <type>Wine</type>
        <subType>Still</subType>
        <designation>GI</designation>
        <classification xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <vintageValues>
            <vintage>2014</vintage>
        </vintageValues>
        <displayNameType>Type 4</displayNameType>
        <displayName>John Glaetzer, John's Blend Cabernet Sauvignon N°40, Barossa
Valley</displayName>
        <status>live</status>
        <combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <dateCreated>2019-10-11T23:34:20Z</dateCreated>
        <lastUpdateDate>2020-04-25T07:55:07Z</lastUpdateDate>
    </view>
    <view>
        <lwin>10024252013</lwin>
        <producerTitle xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <producerName>John Glaetzer</producerName>
        <wine>John's Blend Cabernet Sauvignon N°39</wine>
        <country>Australia</country>
        <region>South Australia</region>
        <subRegion>Barossa Valley</subRegion>
        <site xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <parcel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <colour>Red</colour>
        <type>Wine</type>
        <subType>Still</subType>
        <designation>GI</designation>
        <classification xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <vintageValues>
            <vintage>2013</vintage>
        </vintageValues>
        <displayNameType>Type 4</displayNameType>
        <displayName>John Glaetzer, John's Blend Cabernet Sauvignon N°39, Barossa
Valley</displayName>
        <status>live</status>
        <combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <dateCreated>2019-10-11T23:34:20Z</dateCreated>
        <lastUpdateDate>2020-04-25T07:55:07Z</lastUpdateDate>
    </view>
    <view>
        <lwin>10024252012</lwin>
        <producerTitle xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <producerName>John Glaetzer</producerName>
        <wine>John's Blend Cabernet Sauvignon N°38</wine>
        <country>Australia</country>
        <region>South Australia</region>
        <subRegion>Barossa Valley</subRegion>
    
```

```

        <site xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <parcel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <colour>Red</colour>
        <type>Wine</type>
        <subType>Still</subType>
        <designation>GI</designation>
        <classification xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <vintageValues>
            <vintage>2012</vintage>
        </vintageValues>
        <displayNameType>Type 4</displayNameType>
        <displayName>John Glaetzer, John's Blend Cabernet Sauvignon N°38, Barossa
Valley</displayName>
        <status>live</status>
        <combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <dateCreated>2019-10-11T23:34:20Z</dateCreated>
        <lastUpdateDate>2020-04-25T07:55:07Z</lastUpdateDate>
    </view>
</lwinView>
    <errors xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/>
</lwinViewResponse>

```

#### Invalid XML Response

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<lwinview>
    <Status>OK</Status>
    <HttpCode>200</HttpCode>
    <Message>Request completed successfully</Message>
    <InternalErrorCode>R001</InternalErrorCode>
    <ApiInfo>
        <Version>1.0</Version>
        <Timestamp>2021-01-20T14:15:44.364Z</Timestamp>
        <Provider>Liv-ex</Provider>
    </ApiInfo>
    <pageInfo>
        <totalResults>0</totalResults>
        <limit>50</limit>
        <offset>1</offset>
    </pageInfo>
    <errors>
        <error>
            <code>L002</code>
            <message>Incorrect LWIN: 123467.</message>
        </error>
    </errors>
    <lwinView>
        <lwin>123467</lwin>
        <includeVintageListing>false</includeVintageListing>
    </lwinView>
</lwinview>

```

## 6. Response Codes

This section describes the response codes that will be returned by the service.

Code	Message
R000	Request was unsuccessful
R001	Request completed successfully
R002	Request partially completed

### 6.1 Request validation error codes

Code	Message
L001	Mandatory field [%s] missing
L002	Incorrect LWIN: [%]
L007	Invalid LWIN7 [%] and vintage combination
L028	Invalid includeVintageListing value. Possible values are 'true' or 'false'.

### 6.2 HTTP Status codes

HTTP defines a bunch of meaningful status codes that can be returned from our API. These can be leveraged to help API consumers route their responses accordingly.

Code	Message
200 OK	Response to a successful GET, POST, PUT, DELETE. Can also be used for a POST that doesn't result in a creation.
201 Created	Response to a POST that results in a creation.
202 Accepted	The request has been accepted and will be processed later. It is a classic answer to asynchronous calls (for better UX or performances).
204 No Content	Response to a successful request that won't be returning a body (like a DELETE request)
400 Bad Request	The request is malformed, such as if the body does not parse
401 Unauthorized	When no and/or invalid authentication details are provided. Can also be used to trigger an auth popup if API is used from a browser
403 Forbidden	When authentication succeeded but authenticated user doesn't have access to the resource
404 Not Found	When a non-existent resource is requested



405 Method Not Allowed	When an HTTP method is being requested that isn't allowed for the authenticated user
406 Not Acceptable	Nothing matches the Accept-* Header of the request. As an example, you ask for an XML formatted resource, but it is only available as JSON.
409 Conflict	Indicates one or more supplied parameters are triggering a validation error. A relevant TR code should be returned in the response.
410 Gone	Indicates that the resource at this end point is no longer available. Useful as a blanket response for old API versions
415 Unsupported Media Type	If incorrect content type was provided as part of the request
422 Unprocessable Entity	Used for validation errors. Should be used if the server cannot process the entity, e.g. if an image cannot be formatted or mandatory fields are missing in the payload.
429 Too Many Requests	When a request is rejected due to rate limiting
500 Internal Server Error	The general catch-all error when the server-side throws an exception. The request may be correct, but an execution problem has been encountered at our end.