



THE FINE WINE MARKET

LWIN: Change Since API v1

Document Revision 1.2
Date of Issue: 20 January 2020
Date of revision: 27 July 2022

Daria Ershova

Product Manager

Table of Contents

1. Purpose	3
2. Glossary of Terms	3
3. Technical Standards	3
4. Request Header	3
5. API Listing	5
5.1 LWIN Change Since service (POST method)	5
6. Response Codes	15
6.1 Request validation error codes.....	15
6.2 HTTP Status codes.....	15

1. Purpose

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

2. Glossary of Terms

Term	Meaning
LWIN	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.
Combine	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. Please note that the API expects the 4 headers as listed within this documentation and submitting a request with additional headers may lead to errors and/or failed responses.

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>2022-11-04T11:12:30</Timestamp>
    <Provider>Liv-ex</Provider>
  </ApiInfo>
</Response>
```

5. API Listing

5.1 LWIN Change Since service (POST method)

Description

This service will be used to request all changes in the LWIN dataset over a given time period. LWINs are regularly created, edited, combined or deleted. The service lists these changes in a structured manner.

Base URI

[lwin/changeSince/v1/lwinChangeSince](#)

URI Pagination Parameters

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

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

Request Parameters

Name	Mandatory	Description
timeframe	Y	The time window of LWIN change data to return. Type: alphanumeric Values: 1 hour, 12hour, 24hour, 1week, 1month, or a YYYY-MM-DD date input Limit: 3 months

Sample Request Body

JSON

<pre>{ "timeframe": "1hour" }</pre>
<pre>{ "timeframe": "2022-06-30" }</pre>

XML

<pre><lwinChangeSince> <timeframe>1hour</timeframe> </lwinChangeSince></pre>
<pre><lwinChangeSince> <timeframe>2022-06-30</timeframe> </lwinChangeSince></pre>

Sample Response Body

The LWIN Change Since service returns a list of all LWIN events that have taken place within the requested timeframe period. There are 7 possible event types that can be returned:

<i>LWIN event</i>	<i>Description</i>
LWIN7 creation	New LWIN7 code, all LWIN7 metadata
LWIN7 update	Specific LWIN7 code, all LWIN7 metadata
LWIN7 deletion	Specific LWIN7 code only
LWIN11 creation	New LWIN11 code, all LWIN11 metadata
LWIN11 update	Specific LWIN11 code, all LWIN11 metadata
LWIN11 deletion	Specific LWIN11 code only
LWIN7 combine	"Leader" LWIN7 and "follower" LWIN7

Response pagination parameters

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

Response parameters

Name	Description
lwin	LWIN7 or LWIN11 code depending on type of event Type: alphanumeric
changeType	The category of change: "lwin7Creation", "lwin11Creation", "lwin7Update", "lwin11Update", "lwin7Deletion", "lwin11Deletion", "lwin7Combine" Type: alphanumeric
changeDate	The timestamp of when the event took place Type: datetime / epoch
combineReference	The new leader LWIN for the LWIN that has been combined Type: alphanumeric
producerTitle	Title of producer or owner of wine Type: alphanumeric
producerName	Producer or owner of wine Type: alphanumeric

wine	Name of wine (brand and/or grape and/or technical term) Type: alphanumeric
country	Country of origin Type: alphanumeric
region	Region of origin (where applicable, specific to local laws) Type: alphanumeric
subRegion	Sub-region of origin (where applicable, specific to local laws) Type: alphanumeric
site	Site within sub-region (where applicable, specific to local laws) Type: alphanumeric
parcel	Parcel within site (where applicable, specific to local laws) Type: alphanumeric
colour	Colour of LWIN product Type: alphanumeric Values: "white", "red", "rose", null
type	LWIN beverage type Type: alphanumeric
subType	Subcategory of LWIN type Type: alphanumeric
designation	Officially assigned status (specific to local laws) Type: alphanumeric
classification	Officially declared quality level (where applicable, specific to local laws) Type: alphanumeric
vintageConfiguration	Vintage pattern of the wine (e.g. single vintage only, production ended) Should be null for LWIN11 Type: alphanumeric Values: "sequential", "nonSequential", "singleVintageOnly"
vintageValues	Type: alphanumeric
firstVintage	The first vintage of the LWIN7 Should be null for LWIN11 Type: alphanumeric
finalVintage	The final vintage of the LWIN7 Should be null for LWIN11 Type: alphanumeric
childOf	Parent wine LWIN7. This should be used to map associated wines to one another. This is a separate relationship to LWIN merge references Should be null for LWIN11 Type: alphanumeric
displayNameType	One out of 8 display name concatenation formulas. Assigned by LWIN administrator Type: alphanumeric
displayName	Type: alphanumeric
status	Type: alphanumeric Values: "live", "deleted", "combined"

requestReference	Reference can be provided with "lwin7Creation", "lwin11Creation" metadata if LWIN 7/11 was created for a specific request. Type: integer (11)
dateCreated	Type: datetime / epoch
lastUpdateDate	Type: datetime / epoch

JSON Response

The response is sent per request. The example below shows 5 events – LWIN7 deletion, LWIN7 update, LWIN11 update, LWIN7 creation, LWIN11 creation.

```

"status": "OK",
"httpCode": "200",
"message": "Request completed successfully",
"internalErrorCode": "R001",
"apiInfo": {
  "version": "1.0",
  "timestamp": 1580118867307,
  "provider": "Liv-ex"
},
"pageInfo": {
  "totalResults": 3352,
  "limit": 50,
  "offset": 1
},
"lwinChangeSince": [
  {
    "lwin": "2548074",
    "changeType": "lwin7Deletion",
    "changeDate": 1579530323000,
    "combineReference": null,
    "metaData": null
  },
  {
    "lwin": "2548234",
    "changeType": "lwin7Update",
    "changeDate": 1579525119000,
    "combineReference": null,
    "metaData": {
      "producerTitle": null,
      "producerName": "test5756",
      "wine": "test",
      "country": "Australia",
      "region": "South Australia",
      "subRegion": "Langhe",
      "site": "xyz",
      "parcel": null,
      "colour": "black",
      "type": "Spirit",
      "subType": "Brandy",
      "designation": "Bodegas",
      "classification": "Bodegas",
      "vintageConfiguration": "singleVintageOnly",
      "vintageValues": [
        "2006"
      ],
      "firstVintage": null,
      "finalVintage": null,
      "childOf": null,
      "displayNameType": "Type 5",
    }
  }
]

```



```

        "displayName": "test5756, xyz, test, Bodegas, Langhe, South Australia",
        "status": "live",
        "requestReference": null,
        "dateCreated": 1579525088000,
        "lastUpdateDate": 1579525119000
    }
},
{
    "lwin": "25482342006",
    "changeType": "lwin11Update",
    "changeDate": 1579525119000,
    "combineReference": null,
    "metaData": {
        "producerTitle": null,
        "producerName": "test5756",
        "wine": "test",
        "country": "Australia",
        "region": "South Australia",
        "subRegion": "Langhe",
        "site": "xyz",
        "parcel": null,
        "colour": "black",
        "type": "Spirit",
        "subType": "Brandy",
        "designation": "Bodegas",
        "classification": "Bodegas",
        "vintageConfiguration": null,
        "vintageValues": [
            "2006"
        ],
        "firstVintage": null,
        "finalVintage": null,
        "childOf": null,
        "displayNameType": "Type 5",
        "displayName": "test5756, xyz, test, Bodegas, Langhe, South Australia",
        "status": "live",
        "requestReference": null,
        "dateCreated": 1579525088000,
        "lastUpdateDate": 1579525119000
    }
},
{
    "lwin": "2548221",
    "changeType": "lwin7Creation",
    "changeDate": 1579522380000,
    "combineReference": null,
    "metaData": {
        "producerTitle": "Bodegas",
        "producerName": "Test",
        "wine": "be3u655556668333",
        "country": "Casa",
        "region": "Unknown",
        "subRegion": "Mount",
        "site": "City",
        "parcel": "Myparcel",
        "colour": "black",
        "type": "Spirit",
        "subType": "Brandy",
        "designation": "Bodegas",
        "classification": "asdf",
        "vintageConfiguration": "sequential",
        "vintageValues": [
            "2018"
        ],
        "firstVintage": "2017",
        "finalVintage": "2018",
    }
}

```

```

        "childOf": null,
        "displayNameType": "Type 1",
        "displayName": "be3u655556668333, Bodegas, Test, asdf, Mount",
        "status": "live",
        "requestReference": null,
        "dateCreated": 1579522380000,
        "lastUpdateDate": 1579522380000
    },
    {
        "lwin": "25482212018",
        "changeType": "lwin11Creation",
        "changeDate": 1579522380000,
        "combineReference": null,
        "metaData": {
            "producerTitle": "Bodegas",
            "producerName": "Test",
            "wine": "be3u655556668333",
            "country": "Casa",
            "region": "Unknown",
            "subRegion": "Mount",
            "site": "City",
            "parcel": "Myparcel",
            "colour": "black",
            "type": "Spirit",
            "subType": "Brandy",
            "designation": "Bodegas",
            "classification": "asdf",
            "vintageConfiguration": "sequential",
            "vintageValues": [
                "2018"
            ],
            "firstVintage": null,
            "finalVintage": null,
            "childOf": null,
            "displayNameType": "Type 1",
            "displayName": "be3u655556668333, Bodegas, Test, asdf, Mount",
            "status": "live",
            "requestReference": null,
            "dateCreated": 1579522380000,
            "lastUpdateDate": 1579522380000
        }
    },
    "errors": null
}

```

Invalid JSON response

```

{
    "status": "OK",
    "statusCode": "200",
    "message": "Request completed successfully",
    "internalErrorCode": "R001",
    "apiInfo": {
        "version": "1.0",
        "timestamp": 1611151765336,
        "provider": "Liv-ex"
    },
    "pageInfo": {
        "totalResults": 0,
        "limit": 50,
        "offset": 1
    },
    "lwinChangeSince": {
        "timeframe": "1hou"
    },
}

```

```

"errors": {
  "error": [
    {
      "code": "L021",
      "message": "Invalid timeframe: 1hou. Possible values are '1hour',
'12hour', '24hour', '1week', '1month', or a 'YYYY-MM-DD' date input."
    }
  ]
}
}

```

XML Response

The response is sent per request. The example below shows 5 events – LWIN7 deletion, LWIN7 update, LWIN11 update, LWIN7 creation, LWIN11 creation.

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<lwinChangeSinceResponse>
  <Status>OK</Status>
  <HttpCode>200</HttpCode>
  <Message>Request completed successfully</Message>
  <InternalErrorCode>R001</InternalErrorCode>
  <ApiInfo>
    <Version>1.0</Version>
    <Timestamp>2020-01-27T09:55:20.167Z</Timestamp>
    <Provider>Liv-ex</Provider>
  </ApiInfo>
  <pageInfo>
    <totalResults>3352</totalResults>
    <limit>50</limit>
    <offset>1</offset>
  </pageInfo>
  <lwinChangeSince>
    <lwinChange>
      <lwin>2548074</lwin>
      <changeType>lwin7Deletion</changeType>
      <changeDate>2020-01-20T14:25:23Z</changeDate>
      <combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
      <metaData xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
    </lwinChange>
    <lwinChange>
      <lwin>2548234</lwin>
      <changeType>lwin7Update</changeType>
      <changeDate>2020-01-20T12:58:39Z</changeDate>
      <combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
      <metaData>
        <producerTitle xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <producerName>test5756</producerName>
        <wine>test</wine>
        <country>Australia</country>
        <region>South Australia</region>
        <subRegion>Langhe</subRegion>
        <site>xyz</site>
        <parcel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <colour>black</colour>
        <type>Spirit</type>
        <subType>Brandy</subType>
        <designation>Bodegas</designation>
        <classification>Bodegas</classification>
        <vintageConfiguration>singleVintageOnly</vintageConfiguration>

```

```

        <vintageValues>
            <vintage>2006</vintage>
        </vintageValues>
        <firstVintage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <finalVintage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <childOf xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <displayNameType>Type 5</displayNameType>
        <displayName>test5756, xyz, test, Bodegas, Langhe, South
Australia</displayName>
        <status>live</status>
        <requestReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <dateCreated>2020-01-20T12:58:08Z</dateCreated>
        <lastUpdateDate>2020-01-20T12:58:39Z</lastUpdateDate>
        </metaData>
    </lwinChange>
    <lwinChange>
        <lwin>25482342006</lwin>
        <changeType>lwin11Update</changeType>
        <changeDate>2020-01-20T12:58:39Z</changeDate>
        <combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <metaData>
            <producerTitle xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
            <producerName>test5756</producerName>
            <wine>test</wine>
            <country>Australia</country>
            <region>South Australia</region>
            <subRegion>Langhe</subRegion>
            <site>xyz</site>
            <parcel xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
            <colour>black</colour>
            <type>Spirit</type>
            <subType>Brandy</subType>
            <designation>Bodegas</designation>
            <classification>Bodegas</classification>
            <vintageConfiguration xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:nil="true"/>
            <vintageValues>
                <vintage>2006</vintage>
            </vintageValues>
            <firstVintage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
            <finalVintage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
            <childOf xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
            <displayNameType>Type 5</displayNameType>
            <displayName>test5756, xyz, test, Bodegas, Langhe, South
Australia</displayName>
            <status>live</status>
            <requestReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
            <dateCreated>2020-01-20T12:58:08Z</dateCreated>
            <lastUpdateDate>2020-01-20T12:58:39Z</lastUpdateDate>
            </metaData>
        </lwinChange>
    </lwinChange>
        <lwin>2548221</lwin>
        <changeType>lwin7Creation</changeType>
        <changeDate>2020-01-20T12:13:00Z</changeDate>
    
```

```

    <combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
    <metaData>
      <producerTitle>Bodegas</producerTitle>
      <producerName>Test</producerName>
      <wine>be3u655556668333</wine>
      <country>Casa</country>
      <region>Unknown</region>
      <subRegion>Mount</subRegion>
      <site>City</site>
      <parcel>Myparcel</parcel>
      <colour>black</colour>
      <type>Spirit</type>
      <subType>Brandy</subType>
      <designation>Bodegas</designation>
      <classification>asdf</classification>
      <vintageConfiguration>sequential</vintageConfiguration>
      <vintageValues>
        <vintage>2018</vintage>
      </vintageValues>
      <firstVintage>2017</firstVintage>
      <finalVintage>2018</finalVintage>
      <childOf xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
      <displayNameType>Type 1</displayNameType>
      <displayName>be3u655556668333, Bodegas, Test, asdf, Mount</displayName>
      <status>live</status>
      <requestReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
      <dateCreated>2020-01-20T12:13:00Z</dateCreated>
      <lastUpdateDate>2020-01-20T12:13:00Z</lastUpdateDate>
    </metaData>
  </lwinChange>
  <lwinChange>
    <lwin>25482212018</lwin>
    <changeType>lwin11Creation</changeType>
    <changeDate>2020-01-20T12:13:00Z</changeDate>
    <combineReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
    <metaData>
      <producerTitle>Bodegas</producerTitle>
      <producerName>Test</producerName>
      <wine>be3u655556668333</wine>
      <country>Casa</country>
      <region>Unknown</region>
      <subRegion>Mount</subRegion>
      <site>City</site>
      <parcel>Myparcel</parcel>
      <colour>black</colour>
      <type>Spirit</type>
      <subType>Brandy</subType>
      <designation>Bodegas</designation>
      <classification>asdf</classification>
      <vintageConfiguration xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:nil="true"/>
      <vintageValues>
        <vintage>2018</vintage>
      </vintageValues>
      <firstVintage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
      <finalVintage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
      <childOf xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
      <displayNameType>Type 1</displayNameType>
      <displayName>be3u655556668333, Bodegas, Test, asdf, Mount</displayName>

```

```
        <status>live</status>
        <requestReference xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
        <dateCreated>2020-01-20T12:13:00Z</dateCreated>
        <lastUpdateDate>2020-01-20T12:13:00Z</lastUpdateDate>
        </metaData>
    </lwinChange>
</lwinChangeSince>
    <errors xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/>
</lwinChangeSinceResponse>
```

Invalid XML Response

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<lwinChangeSinceResponse>
  <Status>OK</Status>
  <HttpCode>200</HttpCode>
  <Message>Request completed successfully</Message>
  <InternalErrorCode>R001</InternalErrorCode>
  <ApiInfo>
    <Version>1.0</Version>
    <Timestamp>2021-01-20T14:11:04.481Z</Timestamp>
    <Provider>Liv-ex</Provider>
  </ApiInfo>
  <pageInfo>
    <totalResults>0</totalResults>
    <limit>50</limit>
    <offset>1</offset>
  </pageInfo>
  <errors>
    <error>
      <code>L021</code>
      <message>Invalid timeframe: 24. Possible values are '1hour', '12hour',
'24hour', '1week', '1month', or a 'YYYY-MM-DD' date input.</message>
    </error>
  </errors>
  <lwinChangeSince>
    <timeframe>24</timeframe>
  </lwinChangeSince>
</lwinChangeSinceResponse>
```

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
L021	Invalid timeframe: [%s]. Possible values are '1hour', '12hour', '24hour', '1week', '1month', or a 'YYYY-MM-DD' date input.

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.