# Internal Error Responses

A main internal error message is defined, with JSON structure with the following fields:

* A unique error "code", which can be identified and traced for more details. It must be human readable; therefore, it must not be a numeric code, but an alphanumeric one. It is recommended to follow the following pattern: "/[A-Z]{3}[0-9]\*/g", where the first part of the code must be a string indicating the type of error or equipment that must manage it, and where the second part is a numerical value. In turn, to achieve a better location of the error, you can parameterize the value or field that is causing it ('%1'), and include it in the response.
* A clear error “reason”. For security these error messages must not contain internal data values.
* A detailed description of "message"

A JSON error structure is proposed below:

|  |
| --- |
| {  "code": "ERR001",  "reason": "Missing mandatory field",  "message": "Please provide an Authorization header"  } |

In error handling different cases must be considered, even at the functional level that it is possible to modify the error message returned to the API consumer. For this error handling there are two possible alternatives listed below:

* Error handling done with custom policies in the API admin tool.
* Error management performed in a service queried by API.

The essential requirements to consider would be:

* Error handling should be centralized in a single place, regardless of whether they are APIs that are proxies to end services or APIs that invoke an integrator service on the ESB.
* Customization of the generated error based on the error content returned by the final core service should be contemplated.
* Latency should be minimized in its management.

Below we detail a classification table of error codes – HTTP codes – Description:

**Client Errors**

**syntax Exceptions**

| **Exception** | **Type** | **Text** | **Description** |
| --- | --- | --- | --- |
| SVC1000 | syntax | "Missing mandatory parameter: %1" %1 - Element/Attribute name | API Request without mandatory field |
| SVC1001 | syntax | "Invalid parameter: %1" %1 - Element/Attribute name | API Request with an element not conforming to Swagger definitions or to a list of allowed Query Parameters. |
| SVC0003 | syntax | "Invalid parameter value: %1. Possible values are: %2 " %1- Element/Attribute name %2 - List of valid values | API Request with an element or attribute value not conforming to Swagger definitions or to a list of allowed Query Parameter values, because the element or attribute value belongs to an enumerated list of possible values. |
| SVC0004 | syntax |  | Requested API port or resource does not exist |
| SVC1003 | syntax | Requested HTTP Method Operation does not exist | Requested Operation does not exist |
| SVC1023 | syntax | "Parser Error: JSON content not well formed" | The body of a REST request is not correctly formed, i.e.: the JSON body is not well formed |
| SVC1013 | syntax | "%1 Operation is not allowed: %2" %1 - Operation %2 - Details | Operation syntax is right, but it does not fulfil the conditions (permissions,...) of the scenario or service |
| SVC0005 | syntax | "Value %1 specified in field %2 is a duplicate" %1 - Value %2 - Field | Value specified in a request message is invalid because it already exists. |

**Service Exceptions**

| **Exception** | **Type** | **Text** | **Description** |
| --- | --- | --- | --- |
| SVC0001 | service | "Generic Client Error: %1" %1 - additional info | API Generic wildcard fault response |
| SVC1004 | service | "Requested version API is deprecated. Use %1" %1 - Supported versions of API | API error response for requests over a deprecated version the API. |
| SVC1006 | service | "Resource %1 does not exist" %1 Resource Identifier (e.g.: UserId) | Reference to a resource identifier which does not exist in the collection/repository referred (i.e.: unknown header value or query parameter value ) |
| SVC1011 | service | "Invalid %1 length. Length should be less than %2 characters" %1 - Parameter %2 - Parameter Maximum | Request is indicating a parameter which overcomes length limit established by the service |
| SEC1004 | service | "Invalid Token" | Request has not successfully passed security validations due to wrong Token |
| SEC1003 | service | "Rate Limit" | User has made a lot of request in a time period. (RATE LIMIT) |

#### **Server Errors**

| **Exception** | **Type** | **Text** | **Description** |
| --- | --- | --- | --- |
| SVR1000 | server | "Generic Server Error: %1" %1 - Details | There was a problem in the Service Providers network that prevented to carry out the request |
| SVR1003 | server | "Requested Operation is not implemented: %1" %1 - HTTP method | There was a problem in the Service Providers network that prevented to carry out the request |
| SVR1006 | server | "Service temporarily unavailable: system overloaded" | There was a problem in Server side, unable to handle the request due to a temporary overloading or maintenance of the server |
| SVR1008 | server | "Timeout processing request[:%s]" [:%s] - is optional %s - a string with more details about the timeout cause | There was a timeout in the server side while processing the request. |

***Note: When no login has been performed or no authentication has been assigned, a non-descriptive generic error will always be returned in all cases, a SEC1004 401 Invalid Token is returned, whatever the reason.***