



# User Guide Version 1.16

---

**API Version 1.016**

**Last Update on 2013, February 18th**

## VERSION CONTROL

Version	Date	Description
1.16	05/02/2013	New method <b>GetOperationByID</b> included which allows you to get an operation by its id.
1.15	18/12/2012	Original document including two new methods, <b>AddNewRequest_V2</b> , <b>CheckRequest_V2</b> which basically work like their previous version, but including specific information about your business.

## INTRODUCTION

UboxServer **Api Version 1. 16** allows your applications to communicate directly with the UboxServer. You can now order unlock codes from your website , without having to log in on UboxServer.net.

You can integrate the API in your front office, like an unlocking website, to place directly your customer's orders on the UboxServer, as well as in your back office, to request a code verification.

The API develop by Micorsoft.NET WebService, It is XML-based, platform Independent, and can be interfaced on any OS in any programming language.

## CREATING MY API KEY

To create your API Key you have to create your account in [www.uboxserver.net](http://www.uboxserver.net)

After the register process, you have to send an email to the website administrator ([info@uboxserver.net](mailto:info@uboxserver.net)) and he will reply you with your API Key.

This Key, just as your UboxServer password, must be kept secret.

For security purposes, you can regenerate it just requesting to the website administrator and he will generate a new API KEY and will reply you with your new API KEY.

Please note that your old key will instantly become inactive!

## CREATING MY SIGN

Once you have your API Key, you should generate your sign to validate yourself in the UboxServer API.

Sign = SHA512(user+password+API KEY);

Example:

User : "Jhon"

Password : "Locke"

API KEY: "234234-3223423-676778-5656"

**Sign** = SHA512("JhonLocke234234-3223423-676778-5656");

The result for this example is:

**Sign**="948914afc66ddadb2e7f70b80c9ecd07f5b4d7aa181ae50bb0ba3487027cd1b07cdd90c93c1fb0d2022033076d54019d58b45f82149c2f25fee2033312a0a228"

The Sign is a required parameter of some methods of the API, and please note that the generation of the sign is case sensitive, so check it before generate.

**Obviously if you change your API KEY you have to regenerate your sign to communicate with the UboxServer API.**

## TECHNICAL INFORMATION

The API is located at <http://codes.ath.cx:8099/UBoxAPI.asmx>

You can send your requeries by HTTP GET or HTTP POST method.

All exchanges with the API are encoded in UTF-8.

More information please refer to Example (Coming soon...).

To get a formal definition of the service, please check the service descriptor at '<http://codes.ath.cx:8099/UBoxAPI.asmx?WSDL>'.

## XML STREAM FORMAT

The XML stream always conforms to the template:

```
<?xml version="1.0" encoding = "UTF-8" ?>
```

```
<response>
```

```
    <haserrors> </haserrors>
```

```
    <error>
```

```
    ...
```

```
    </error>
```

```
    <result>
```

```
    ...
```

```
    </result>
```

```
</response>
```

### Very Important Question

If the **<haserrors>** tag has Value = *"true"* the tag **<error>** will contain a tag called **<message>** that will contain a full description of the error, and the **<result>** tag will be empty.

If the **<haserrors>** tag has Value = *"false"* the tag **<error>** will be empty and the **<result>** tag will contain the result of the request.

All the tags follows the **standard HtmlEncode** to allow special characters due to the error logs include this kind of characters.

There is no parameters on XML tags, such as **<Tag Parameter = "Value">**

You will only encounter simple <Tags>

A tag can contain either a value, the specific data will be explained ahead for each method.

**<Tag> Value </Tag>**

Or contain other tags:

**<Tag>**

**<SubTag>**

**...**

**</SubTag>**

**</Tag>**

But not both:

**<Tag>**

**Value**

**<SubTag>**

**...**

## AVAILABLE FUNCTIONS

All functions are encapsulated into a class named UboxApi

- If you are using Visual Studio and you want to create a WebReference you have to instantiate an object of the type UboxApi .
- In any other case you can directly access by the url.

## API SERVICES

In the UboxServer API, the services are named Actions.

Here is an exhaustive list of the actions:

Action	Description
GetUserInfo	Returns the information of the UboxServer account.
GetBrands	Returns a list of the UboxServer brands.
GetModels	Returns a list of the UboxServer models.
GetOperations	Returns a list of the UboxServer operations.
GetAPIServicesVersion	Returns the version of available services.
GetAPIVersion	Returns the version of API interface.
AddNewRequest	Order an unlock code by IMEI or an external account.
TryCancelRequest	Try to cancel an existing request
CheckRequest	Check the state of an existing request.
GetAllRequestsIds	Get all your Request's Ids.
GetNetWorks	Returns a list of the NetWorks associated to a Operation.
AddNewRequest_V2	Order an unlock code by IMEI or an external account including specific information about your business.
CheckRequest_V2	Check the state of an existing request including specific information about your business.
GetOperationByID	Returns an UboxServer operation by its id.

The actions are detailed in the following pages.

## GetUserInfo

- DESCRIPTION:
  - Returns the information of the UboxServer account.
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
sign	String	Yes	User sign.

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>  
  
  <haserrors>false</haserrors>  
  
  <error />  
  
  <result>  
  
    <userInfo>  
  
      <id>898</id>  
  
      <name>John</name>  
  
      <surname>Locke </surname />  
  
      <email>john999@yahoo.es</email>  
  
        <phone>677238823</phone />  
  
      <credits>1.000,000</credits>  
  
    </userInfo>  
  
  </result>  
  
</response>
```

- This is an example with errors. Real data will be different.

```
<response>
```

```

<haserrors>true</haserrors>

<error >
    <message> An error... </message>

</error>

</result>

</response>

```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	Long	haserrors = FALSE	The Id of the account.
name	String	haserrors = FALSE	The name of the account.
surname	String	haserrors = FALSE	The surname of the account.
email	String	haserrors = FALSE	The email address of the account.
phone	String	haserrors = FALSE	The phone of the account.
credits	Double	haserrors = FALSE	Credits currently available on the account. Decimal separator: , Thousand separator: ‘ ’





- DESCRIPTION:
  - Returns a list of the UboxServer brands.
  
- PARAMETERS
  - Do not have.
  
- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <brand>
      <id>28</id>
      <name>BRAND01</name>
      <description>BRAND01 Desc</description>
    </brand>
    <brand>
      <id>29</id>
      <name>BRAND02</name>
      <description>BRAND02 Desc</description>
    </brand>
  </result>
</response>
```

- o This is an example with errors. Real data will be different.

```

<response>
  <haserrors>true</haserrors>
  <error >
    <message> An error... </message>
  </error>
</result>
</response>

```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	Long	haserrors = FALSE	The Id of the brand.
name	String	haserrors = FALSE	The name of brand.
description	String	haserrors = FALSE	The description of the brand.

- DESCRIPTION:
  - Returns a list of the UboxServer models.

- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
sign	String	Yes	User sign.

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <model>
      <id>6</id>
      <name>Model 009</name>
      <description>Model 009 Desc</description />
      <brandId>28</brandId>
    </model>
    <model>
      <id>8</id>
      <name>Model 045</name>
      <description> Model 045 Desc </description />
      <brandId>28</brandId>
    </model>
  </result>
</response>
```

- o This is an example with errors. Real data will be different.

```

<response>
  <haserrors>true</haserrors>
  <error >
  <message> An error...</message>
  </error
  </result>
</response>

```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	Long	haserrors = FALSE	The Id of the model.
name	String	haserrors = FALSE	The name of model.
description	String	haserrors = FALSE	The description of the model.
brandId	Long	haserrors = FALSE	The Id of the brand.

## GetOperations

- DESCRIPTION:
  - Returns a list of the UboxServer operations.
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
sign	String	Yes	User sign.

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

<response>

<haserrors>**false**</haserrors>

<error />

<result>

<operation>

<id>**2**</id>

<name>**OPER\_001**</name>

<modelId>**6**</modelId>

<credits>**15**</credits>

<description> **OPER\_001DESC**</description>

<resolutionTime>**0 to 5 Minutes**</resolutionTime>

<minResolution>**0**</minResolution>

<maxResolution>**5**</maxResolution>

<periodResolution>**Minutes**</periodResolution>

<infoextraRequired>**true**</infoextraRequired>

<netRequired> **true** </netRequired>

</operation>

<operation>.....</operation>

</result>

</response>

This is an example with errors. Real data will be different.

<response>

<haserrors>true</haserrors>

<error >

<message> An error...</message>

</error

</result>

</response>

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	Long	haserrors = FALSE	The Id of the operation.
name	String	haserrors = FALSE	The name of operation.
modelId	Long	haserrors = FALSE	The name of the model
credits	Double	haserrors = FALSE	Cost in credits of the operation Decimal separator: , Thousand separator: ‘,
description	String	haserrors = FALSE	The description of the model.
resolutionTime	String	haserrors = FALSE	A description of the required time to solve a request of the operation
minResolution	int	haserrors = FALSE	Min time of resolution
maxResolution	int	haserrors = FALSE	Max time of resolution
periodResolution	String	haserrors = FALSE	Resolution period. It can be one of these values: Undefined/Minutes/Hours/ Days/Week
infoextraRequired	String	haserrors = FALSE	It indicates if infoExtra Field is required when adding new IMEI. It can be one of these values: true/false
netRequired	String	haserrors = FALSE	It indicates if netRequired is required when adding new IMEI. It can be one of these values: true/false

## *GetAPIServicesVersion*

- DESCRIPTION:
  - Brands, Models and Operations are frequently changed (price, availability, name, etc). This method provides the way to know the current version. You can check this value to know if your data version is up to date.
- PARAMETERS
  - Do not have.
- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

<response>

<haserrors>false</haserrors>

<error />

<result>

<servicesVersion>API000000002076</servicesVersion>

</result>

</response>

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
servicesVersion	String	haserrors = FALSE	The actual version.

## AddNewRequest

- DESCRIPTION:
  - Order an unlock code by IMEI or an external account.
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
operationId	Long	Yes	The Id of the ordered operation.
imei	String	Only if order by imei	The ordered IMEI.(Must be valid IMEI). If you are ordering an external account, the IMEI must be empty string.
infoExtra	String	No	
notes	String	No	
sign	String	Yes	User sign.
netId	Long	Yes	The Id of the ordered operation, <b>-1</b> if the operation do not require Network.

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <petition>
      <id>S0034132312388PPF</id>
      <name>PETITION 001 OF NEW BRAND<name>
      <brand>BRAND002</brand>
      <model>MODEL0034</model>
      <requestDate>2010/07/13 12:07:48</requestDate>
```



```
<infoExtra>API REQUEST INFO</infoExtra>

<notes>API REQUEST NOTES</notes>

<imei>999999999888882</imei>

<state>Resolved</state>

<result>999999999888882</result>

<description />

<isCancellationLocked>True</isCancellationLocked>

<maxCancellationDate>2010/07/17 12:07:48

</maxCancellationDate>

<credits>20,000</credits>

<creditReturned>false</creditReturned>

</petition>

</result>

</response>
```

- This is an example with errors. Real data will be different.

```
<response>

  <haserrors>true</haserrors>

  <error >

    <message> An error...</message>

  </error>

</result>

</response>
```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	String	haserrors = FALSE	The Id of the request.
name	String	haserrors = FALSE	The name of request.
brand	String	haserrors = FALSE	The name of the brand requested.
model	String	haserrors = FALSE	The name of the model requested.
request Date	String	haserrors = FALSE	The date of the request with the format: YYYY/MM/DD HH:MM:SS
infoExtra	String	haserrors = FALSE	The info extra of the request.
notes	String	haserrors = FALSE	The notes of the request.
imei	String	haserrors = FALSE	The IMEI of the request.
state	Enum	haserrors = FALSE	The current state of the request can be: Pending Erroneous Cancelled Resolved Undefined
result	String	haserrors=FALSE	The result of the request
description	String	haserrors = FALSE	The description of the request. It contains info about error/Cancellation if needed.
isCancellationLocked	Bool	haserrors = FALSE	A Boolean value that explains if the user can cancel the request. False: The user can cancel the request. True: The user can not cancel the request.
maxCancellationDate	String	haserrors = FALSE	The top date to user cancellation with the format: YYYY/MM/DD HH:MM:SS
credits	Double	haserrors = FALSE	Cost in credits of the request Decimal separator: , Thousand separator: ‘ ’
creditReturned	Bool	haserrors = FALSE	It indicates if credit has been returned for canceled/erroneous requests. It can be ‘true’ or ‘false’.

## TryCancelRequest

- DESCRIPTION:
  - Try to cancel an existing request, if the parameters are correctly introduced and is possible cancel the petition, the state will be “Cancelled” and the credits will be returned and showed in the *credits* tag.
  - In any other case, an error will be returned.
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
requestId	Long	Yes	The Id of the request that you want cancel.
sign	String	Yes	User sign.

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <petition>
      <id>S0034132312388PPF</id>
      <name>PETITION 001 OF NEW BRAND</name>
      <brand>BRAND002</brand>
      <model>MODEL0034</model>
      <requestDate>2010/07/13 12:07:48</requestDate>
      <infoExtra>API REQUEST INFO</infoExtra>
      <notes>API REQUEST NOTES</notes>
    </petition>
  </result>
  <imei>999999999888882</imei>
  <state>Resolved</state>
  <result>999999999888882</result>
```

```
<description />
<isCancellationLocked>True</isCancellationLocked>
<maxCancellationDate>2010/07/17
12:07:48</maxCancellationDate>
<credits>20,000</credits>
<creditReturned>true</creditReturned>
</petition>
</result>
</response>
```

- o This is an example with errors. Real data will be different.

```
<response>
  <haserrors>true</haserrors>
  <error >
    <message> An error...</message>
  </error>
</result>
</response>
```

▪ FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	String	haserrors = FALSE	The Id of the request.
name	String	haserrors = FALSE	The name of request.
brand	String	haserrors = FALSE	The name of the brand requested.
model	String	haserrors = FALSE	The name of the model requested.
request Date	String	haserrors = FALSE	The date of the request with the format: YYYY/MM/DD HH:MM:SS
infoExtra	String	haserrors = FALSE	The info extra of the request.
notes	String	haserrors = FALSE	The notes of the request.
imei	String	haserrors = FALSE	The IMEI of the request.
state	Enum	haserrors = FALSE	The current state of the request can be: Pending Erroneous Cancelled Resolved Undefined
result	String	haserrors=FALSE	The result of the request
description	String	haserrors = FALSE	The description of the request It contains info about error/Cancellation if needed.
isCancellationLocked	Bool	haserrors = FALSE	A Boolean value that explains if the user can cancel the request. False: The user can cancel the request. True: The user can not cancel the request.
maxCancellationDate	String	haserrors = FALSE	The top date to user cancellation with the format: YYYY/MM/DD HH:MM:SS
credits	Double	haserrors = FALSE	Cost in credits of the request Decimal separator: , Thousand separator: ‘ ’
creditReturned	Bool	haserrors = FALSE	It indicates if credit has been returned for canceled/erroneous requests. It can be ‘true’ or ‘false’.

## CheckRequest

- DESCRIPTION:
  - Check the state of an existing request.
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
requestId	Long	Yes	The Id of the request that you want check.
sign	String	Yes	User sign.

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <petition>
      <id>S0034132312388PPF</id>
      <name>PETITION 001 OF NEW BRAND</name>
      <brand>BRAND002</brand>
      <model>MODEL0034</model>
      <requestDate>2010/07/13 12:07:48</requestDate>
      <infoExtra>API REQUEST INFO</infoExtra>
      <notes>API REQUEST NOTES</notes>
      <imei>999999999888882</imei>
      <state>Error</state>
    </result />
    <description>Not Found.</description>
    <isCancellationLocked>True</isCancellationLocked>
```

```
<maxCancellationDate>2010/07/17
12:07:48</maxCancellationDate>
<credits>20,000</credits>
<creditReturned>true</creditReturned>
</petition>
</result>
</response>
```

- This is an example with errors. Real data will be different.

```
<response>
  <haserrors>true</haserrors>
  <error >
    <message> An error...</message>
  </error>
</result>
</response>
```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	String	haserrors = FALSE	The Id of the request.
name	String	haserrors = FALSE	The name of request.
brand	String	haserrors = FALSE	The name of the brand requested.
model	String	haserrors = FALSE	The name of the model requested.
request Date	String	haserrors = FALSE	The date of the request with the format: YYYY/MM/DD HH:MM:SS
infoExtra	String	haserrors = FALSE	The info extra of the request.
notes	String	haserrors = FALSE	The notes of the request.
imei	String	haserrors = FALSE	The IMEI of the request.
state	Enum	haserrors = FALSE	The current state of the request can be: Pending Erroneous Cancelled Resolved Undefined
result	String	haserrors=FALSE	The result of the request
description	String	haserrors = FALSE	The description of the request. It contains info about error/Cancellation if needed.
isCancellationLocked	Bool	haserrors = FALSE	A Boolean value that explains if the user can cancel the request. False: The user can cancel the request. True: The user can not cancel the request.
maxCancellationDate	String	haserrors = FALSE	The top date to user cancellation with the format: YYYY/MM/DD HH:MM:SS
credits	Double	haserrors = FALSE	Cost in credits of the request Decimal separator: , Thousand separator: ‘ ’
creditReturned	Bool	haserrors = FALSE	It indicates if credit has been returned for canceled/erroneous requests. It can be ‘true’ or ‘false’.



## *GetAllRequestsIds*

- DESCRIPTION:
  - Returns a list of all the ids of your petitions.
  
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
sign	String	Yes	User sign.

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```

<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <petition>
      <id>API00000000010B</id>
    </petition>
    <petition>
      <id>API00000000010C</id>
    </petition>
    ....
  </result>
</response>

```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	Long	haserrors = FALSE	The Id of the request.

- DESCRIPTION:
  - Returns a list of the NetWorks associated to a Operation.
- PARAMETERS

Parameter	Type	Required	Description
idOperation	long	Yes	The Id of the operation .

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```

<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    < network>
      <id>2099</id>
      <name>Net001</name>
      <aux>AuxiliarData</aux>
    </ network>
    < network>
      ...
    </ network>
    .....
  </result>
</response>

```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	Long	haserrors = FALSE	The Id of the network.
name	String	haserrors =FALSE	The name of the network
aux	String	haserrors =FALSE	Additional data of the network.

## AddNewRequest\_V2

- DESCRIPTION:
  - Order an unlock code by IMEI or an external account.
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
operationId	Long	Yes	The Id of the ordered operation.
imei	String	Only if order by imei	The ordered IMEI.(Must be valid IMEI). If you are ordering an external account, the IMEI must be empty string.
infoExtra	String	No	
notes	String	No	
sign	String	Yes	User sign.
netId	Long	Yes	The Id of the ordered operation, <b>-1</b> if the operation do not require Network.
customData	String	Yes	A custom string that represents specific information of your business. (Max length = 25 characters)

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <petition>
      <id>S0034132312388PPF</id>
      <name>PETITION 001 OF NEW BRAND<name>
      <brand>BRAND002</brand>
      <model>MODEL0034</model>
      <requestDate>2010/07/13 12:07:48</requestDate>
```

```
<infoExtra>API REQUEST INFO</infoExtra>

<notes>API REQUEST NOTES</notes>

<imei>999999999888882</imei>

<state>Resolved</state>

<result>999999999888882</result>

<description />

<isCancellationLocked>True</isCancellationLocked>

<maxCancellationDate>2010/07/17 12:07:48

</maxCancellationDate>

<credits>20,000</credits>

<creditReturned>false</creditReturned>

<customData>M001</ customData >

</petition>

</result>

</response>
```

- This is an example with errors. Real data will be different.

```
<response>

  <haserrors>true</haserrors>

  <error >

    <message> An error...</message>

  </error>

</result>

</response>
```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	String	haserrors = FALSE	The Id of the request.
name	String	haserrors = FALSE	The name of request.
brand	String	haserrors = FALSE	The name of the brand requested.
model	String	haserrors = FALSE	The name of the model requested.
request Date	String	haserrors = FALSE	The date of the request with the format: YYYY/MM/DD HH:MM:SS
infoExtra	String	haserrors = FALSE	The info extra of the request.
notes	String	haserrors = FALSE	The notes of the request.
imei	String	haserrors = FALSE	The IMEI of the request.
state	Enum	haserrors = FALSE	The current state of the request can be: Pending Erroneous Cancelled Resolved Undefined
result	String	haserrors=FALSE	The result of the request
description	String	haserrors = FALSE	The description of the request. It contains info about error/Cancellation if needed.
isCancellationLocked	Bool	haserrors = FALSE	A Boolean value that explains if the user can cancel the request. False: The user can cancel the request. True: The user can not cancel the request.
maxCancellationDate	String	haserrors = FALSE	The top date to user cancellation with the format: YYYY/MM/DD HH:MM:SS
credits	Double	haserrors = FALSE	Cost in credits of the request Decimal separator: , Thousand separator: ‘ ’
creditReturned	Bool	haserrors = FALSE	It indicates if credit has been returned for canceled/erroneous requests. It can be ‘true’ or ‘false’.
customData	String	haserrors=False	It has the same information that the parameter 'customData'.

## CheckRequest\_V2

- DESCRIPTION:
  - Check the state of an existing request.
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
requestId	Long	Yes	The Id of the request that you want check.
sign	String	Yes	User sign.

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <petition>
      <id>S0034132312388PPF</id>
      <name>PETITION 001 OF NEW BRAND</name>
      <brand>BRAND002</brand>
      <model>MODEL0034</model>
      <requestDate>2010/07/13 12:07:48</requestDate>
      <infoExtra>API REQUEST INFO</infoExtra>
      <notes>API REQUEST NOTES</notes>
      <imei>999999999888882</imei>
      <state>Error</state>
      <description>Not Found.</description>
      <isCancellationLocked>True</isCancellationLocked>
      <maxCancellationDate>2010/07/17
12:07:48</maxCancellationDate>
```

```

<credits>20,000</credits>

<creditReturned>true</creditReturned>

<customData>M001</ customData >

</petition>

</result>

</response>

```

- o This is an example with errors. Real data will be different.

```

<response>

  <haserrors>true</haserrors>

  <error >

    <message> An error...</message>

  </error

</result>

</response>

```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	String	haserrors = FALSE	The Id of the request.
name	String	haserrors = FALSE	The name of request.
brand	String	haserrors = FALSE	The name of the brand requested.
model	String	haserrors = FALSE	The name of the model requested.
request Date	String	haserrors = FALSE	The date of the request with the format: YYYY/MM/DD HH:MM:SS
infoExtra	String	haserrors = FALSE	The info extra of the request.
notes	String	haserrors = FALSE	The notes of the request.
imei	String	haserrors = FALSE	The IMEI of the request.

state	Enum	haserrors = FALSE	The current state of the request can be: Pending Erroneous Cancelled Resolved Undefined
result	String	haserrors=FALSE	The result of the request
description	String	haserrors = FALSE	The description of the request. It contains info about error/Cancelation if needed.
isCancellationLocked	Bool	haserrors = FALSE	A Boolean value that explains if the user can cancel the request. False: The user can cancel the request. True: The user can not cancel the request.
maxCancellationDate	String	haserrors = FALSE	The top date to user cancellation with the format: YYYY/MM/DD HH:MM:SS
credits	Double	haserrors = FALSE	Cost in credits of the request Decimal separator: ',' Thousand separator: ''
creditReturned	Bool	haserrors = FALSE	It indicates if credit has been returned for canceled/erroneous requests. It can be 'true' or 'false'.
customData	String	haserrors=FALSE	A custom string that represents specific information of your business. (Max length = 25 characters)



## *GetOperationByID*

- DESCRIPTION:
  - Returns an UboxServer operation by its id.
- PARAMETERS

Parameter	Type	Required	Description
user	String	Yes	User name of the account.
sign	String	Yes	User sign.
idOperation	long	Yes	Id of the operation

- XML STREAM FORMAT
  - This is an example without errors. Real data will be different.

```
<response>
  <haserrors>false</haserrors>
  <error />
  <result>
    <operation>
      <id>2</id>
      <name>OPER_001</name>
      <modelId>6</modelId>
      <credits>15</credits>
      <description> OPER_001DESC</description>
      <resolutionTime>0 to 5 Minutes</resolutionTime>
      <minResolution>0</minResolution>
      <maxResolution>5</maxResolution>
      <periodResolution>Minutes</periodResolution>
      <infoextraRequired>true</infoextraRequired>
      <netRequired> true </netRequired>
    </operation>
  </result>
</response>
```

This is an example with errors. Real data will be different.

```

<response>
  <haserrors>true</haserrors>
  <error >
    <message> An error...</message>
  </error>
</result>
</response>

```

- FIELD DESCRIPTIONS

Parameter	Type	Visibility	Description
haserrors	Bool	Yes	A Boolean value that explains if there are any error.
message	String	haserrors = TRUE	A description of the error.
id	Long	haserrors = FALSE	The Id of the operation.
name	String	haserrors = FALSE	The name of operation.
modelId	Long	haserrors = FALSE	The name of the model
credits	Double	haserrors = FALSE	Cost in credits of the operation Decimal separator: , Thousand separator: ‘
description	String	haserrors = FALSE	The description of the model.
resolutionTime	String	haserrors = FALSE	A description of the required time to solve a request of the operation
minResolution	int	haserrors = FALSE	Min time of resolution
maxResolution	int	haserrors = FALSE	Max time of resolution
periodResolution	String	haserrors = FALSE	Resolution period. It can be one of these values: <i>Undefined/Minutes/Hours/Days/Week</i>
infoextraRequired	String	haserrors = FALSE	It indicates if infoExtra Field is required when adding new IMEI. It can be one of these values: <i>true/false</i>
netRequired	String	haserrors = FALSE	It indicates if netRequired is required when adding new IMEI. It can be one of these values: <i>true/false</i>

## [Appendix A]

---

### *Integrating PHP.*

The API PHP libraries for connect with UboxServer can be downloaded from the address [http://uboxserver.net/support/UboxAPI\\_Php\\_Libraries.rar](http://uboxserver.net/support/UboxAPI_Php_Libraries.rar).

This .rar file contains two folders, the API library and an usage example. The PHP API library is a PHP class built to connect with the UboxServer SOAP Web Service. It uses the PHP library 'nusoap', which is available at <http://sourceforge.net/projects/nusoap/>. Nusoap is included in the file, but you can download a recent release from the official site and replace files.

The example folder contains a few PHP files included for showing the usage of the API file. Those examples can help you to do the SOAP API calls. The processing of the API response is not included. Each user must to develop the code for processing the API response according with its business logic.

## [Appendix B]

---

### *DHRU FUSION is ready.*

Now you can get our services through DHRU FUSION.

API is free update, but if your customer need your API (UBOXSERVER.NET) now, so they just need to pay for upgrade service.(20\$).

All new users and with next version upgrade all will get free, so just call your client to contact our support and ask for your API now.

Your site added in our API list on main site too <http://dhru.com/cms/fusion/Features>

Also there is no need any manual, just go settings> API settings> and active your API, see this:

## [Appendix C]

---

### *ERROR MESSAGES.*

Here you have a table with all the possible error messages detailed.

#01#000	Unhandled exception while getting the user info. Contact support.
#01#001	IP has been locked.
#01#002	Invalid user.
#01#003	IP has been locked.
#01#004	Incorrect user.
#01#005	User locked.
#01#006	User registration not allowed.
#01#007	Incorrect Key.
#02#000	Unhandled exception while getting brands. Contact support.
#03#000	Unhandled exception while getting models. Contact support.
#04#000	Unhandled exception while getting operations. Contact support.
#04#001	Unhandled exception while getting operations. Contact support.
#04#002	Invalid user.
#04#003	Invalid sign.
#04#004	Invalid login params.
#04#005	Invalid user.
#04#006	User refused. Wrong authentication.
#05#000	Unhandled exception while getting networks. Contact support.
#05#001	Invalid Operation Id.
#06#000	Unhandled exception while adding new request. Contact support.
#06#001	Invalid User.
#06#002	Invalid OperationId.
#06#003	Invalid Sign.
#06#004	IP has been locked. Please, contact support.
#06#005	Invalid User.
#06#006	Invalid OperationId.
#06#007	Operation not enabled. Only for VIP Users.
#06#008	Forbidden operation. You have exceeded the amount of IMEIS allowed for a month.
#06#009	Forbidden operation. You have exceeded the amount of IMEIS allowed for a month.
#06#010	Forbidden operation. Insuficient credits.
#06#011	Invalid IMEI.

#06#012	Invalid Tool id.
#06#013	Invalid Model id.
#06#014	Invalid Brand id.
#06#015	Invalid Network.
#06#016	IMEI Out of range.
#06#017	IMEI Out of range.
#06#018	Invalid Provider ID. Type the Digits exactly as appear at the back of your device
#06#019	Invalid Provider ID. Type the Digits exactly as appear at the back of your device
#06#020	Info Extra Required. Read Operation description to get more info.
#06#021	Invalid User.
#06#022	IP has been locked. Please, contact support.
#06#023	Invalid User.
#06#024	Invalid Request ID. Not Found.
#06#025	Invalid User.
#07#000	Unhandled exception while checking request. Contact support.
#07#001	Invalid User.
#07#002	Invalid Request ID. Empty RequestID.
#07#003	Invalid Sign.
#07#004	Invalid Request ID. Invalid ID.
#07#005	Invalid Request ID. RequestID Not Found.
#07#006	IP has been locked. Please, contact support.
#07#007	Invalid User.
#07#008	Invalid Request ID. Not Found.
#07#009	Invalid User.
#08#000	Unhandled exception while trying to cancel request. Contact support.
#08#001	Invalid User.
#08#002	Invalid Request ID.
#08#003	Invalid Sign.
#08#004	Invalid Request ID.
#08#005	Invalid Request ID.
#08#006	IP has been locked. Please, contact support.
#08#007	Invalid User.
#08#008	Invalid cancellation.
#08#009	Cancellation not allowed.
#08#010	Cancellation not allowed.
#08#011	Cancellation not allowed.
#08#012	Invalid cancellation.
#08#013	Error While Getting Request.
#08#014	Invalid User.
#09#000	Unhandled exception while getting all requests IDs by user. Contact support.
#09#001	Invalid User.

#09#002	Invalid Sign.
#09#003	IP has been locked. Please, contact support.
#09#004	Invalid User.
#09#005	Invalid User.
#10#000	Unhandled exception while getting services version. Contact support.
#11#000	Unhandled exception while adding new request V2. Contact support.
#11#001	Invalid User.
#11#002	Invalid OperationId.
#11#003	Invalid Sign.
#12#000	Unhandled exception while checking request V2. Contact support.
#12#001	Invalid User.
#12#002	Invalid Request ID. Empty RequestID.
#12#003	Invalid Sign.
#12#004	Invalid Request ID. Invalid ID.
#12#005	Invalid Request ID. RequestID Not Found.
#13#000	Unhandled SoapException while Getting the operation by ID. Contact support.
#13#001	Unhandled exception while Getting the operation by ID. Contact support.
#13#002	Invalid User.
#13#003	Invalid Sign.
#13#004	Invalid Operation Id.
#13#005	Invalid Operation.
#13#006	Invalid Login Params.
#13#007	Invalid User.
#13#008	Invalid Operation.
#13#009	The user is not allowed to use this operation.
#13#010	User refused. Wrong authentication.