**Description of REST ISP WebAPI services**

The REST ISP WebAPI services system enables applications external to the ISP to perform various queries on the ISP database as well as various functions with files, directories and communication facilities.

All services are divided into groups, depending on the object they are aimed at. For example, the webusers group combines services which are designed to work with a list of web users. These services include a service for entering a new user, a service for searching all users, a service for searching the data of a particular user, etc.

**Service groups**

1. Service functions (4)
2. WebUsers mobile app user list services (11)
3. Services for working with the list of mobile app users' devices WebUsersDevices (11)
4. Client list services for Clients (7)
5. Services for working with the Countries list (1)
6. Currencies list services (1)
7. Services for working with a list of banks (1)
8. Specials list services (1)
9. Services for working with Brunches client industry list (1)
10. ClientDocFiles document list services (2)
11. Services for handling the list of investment accounts of Contracts(1) clients
12. Services for the ClientsBankAccounts list (3)
13. Services for working with a list of client request types ClientsRequestsTypes
14. Services for handling ClientsRequests (4)
15. Options list services (1)
16. **Service functions**

The serverjobs group of services combines services that perform a variety of service functions.

* 1. **Email service for sending an email**  
       
     http://10.0.4.7:8091/serverjobs/SendEMail/{'email': 'abc@def. com', 'body': 'Hello, .... '}  
       
     Here instead of '[abc@def. com](mailto:abc@def.com)' you should specify the e-mail address you want the message to be sent to, and   
     instead of 'Hello, ....' you should specify the text of the message itself

After sending the message, the service returns a value of 1.

For example:  
[http://10.0.4.7:8091/serverjobs/SendEMail/{'email':'aaa@bbb.gr','body](http://10.0.4.7:8091/serverjobs/SendEMail/%7b'email':'aaa@bbb.gr','body)': 'Your email Verification Code is 123456'}

**1.2 Service for sending a standard e-mail message**  
  
The following service is used to send an e-mail message with a defined standard text by the system server  
  
http://10.0.4.7:8091/serverjobs/SendEMail46/{'recipient\_email': 'abc@def.com', 'request\_action' : '6', 'request\_id': '123', 'att': 'document. pdf', 'person\_name': ' Κ\*\*\*\*\*Σ Π\*\*\*\*\*\*\*\*\*\*Σ ', 'afm': '123456789', 'delay': '30'}  
  
where

recipient\_email is the email address the message should be sent to, e.g. here email abc@def. com. This attribute is mandatory  
request\_action is the number of one of the standard messages. Their list is given below. This attribute is mandatory

request\_id - request ID received after the request has been sent to the server. This attribute is mandatory only if request\_action=1 or 2 or 3  
att- name of the file attached to this message. This attribute is optional  
person\_name - last name of the person. Can be coded with asterisks. This attribute is optional  
afm - a person's ΑΜ. This attribute is optional.  
delay - the delay time in seconds before an e-mail is sent right after the files have been uploaded  
The list of standard messages:  
1. Ενημέρωση αιτήματος - message about successful sending of the client's request. The format of the message is as follows

{'recipient\_email': 'it@hellasfin. gr', 'request\_action': '1', 'request\_id': '3', 'person\_name': 'Βασιλειοσ Βενετικιδησ'}

where

recipient\_email is the e-mail address the message should be sent to,  request\_action = 1

request\_id - request ID received after the request has been sent to the server

2. Ολοκλήρωση αιτήματος - message about the successful execution of the customer order3  
. Απόρριψη αιτήματος - message about cancellation of the customer order  
4. Δημιουργία συνθηματικού - message about successful change of password  
5. Επιτυχημένη εγγραφή - Αποδοχή όρων - message on successful registration6  
. Αποστολή πρόσκλησης στον master - message to the master inviting the client with the name and AFM   
7. Παραλαβή πρόσκλησης στον πελάτη - message to the client about the invitation to contact from the person with the given name and AFM8  
. Διαγραφή συνδέσεις στον πελάτη - message to the client to remove communication with him9  
. Διαγραφή συνδέσεις στον master - message to the master to remove communication with the specified client10  
. Απόρρριψη πρόσκλησης στον πελάτη - message to the client about the cancellation of their communication invitation11  
. Απόρρριψη πρόσκλησης στον master - telling the master to cancel the client's communication invitation12  
. Δημιουργία συνδέσεις στον πελάτη - message to the client to establish communication with him13  
. Δημιουργία συνδέσεις στον master - message to the master to contact the client  
14. Λήξη πρόσκλησης στον master - message to master about expiration of the waiting period of 30 days  
15. Λήξη πρόσκλησης στον πελάτη - message to the client about the expiration of the waiting period of 30 days  
16. Ακύρωση πρόσκλησης στον master - message to master about cancellation of communication with client  
17. Ακύρωση πρόσκλησης στον πελάτη - message to the client to annul communication with the specified person   
18. Ακύρωση αιτήματος στον χρήστη - message to the user who created the request that he himself has cancelled the request while this request has been sent but not yet executed by Alkis  
19. Ακύρωση αιτήματος in HellasFin - message that the user has cancelled the request that he has already sent, but it has not yet been executed

20. Έχετε ένα νέο αίτημα in HellasFin - message about a new client application

{'recipient\_email': 'it@hellasfin.gr', 'request\_action': '20', 'person\_name': 'Βασιλης Βενετικιδης' }

After sending the message, the service returns a value of 1.

* 1. **Service for sending SMS messages**  
       
     http://10.0.4.7:8091/serverjobs/SendSMS/{'mobile': 'AAAAAA', 'c': 'BBBBB'}  
       
     Here instead of 'AAAAAA' you should specify the mobile phone number to which **the message should be sent,**and the text of the message itself instead of BBBBB

For example  
:http://10.0.4.7:8091/serverjobs/SendSMS/{'mobile': XXXXXXXXXXXX, 'message': 'Your mobile Verification Code is 123456'}

* 1. **Service for uploading a file to the server**  
       
     http://10.0.4.7:8091/serverjobs/UploadFile/{'file\_name': 'alfa. pdf', 'client\_id': '356', 'doc\_type': '27', 'status' : '1' }  
       
     this service uploads file alfa.pdf, which is document type 27 with status equal to 1, to the client folder with ID = 356

1. **Services for working with the user list of the WebUsers mobile app** 
   1. **Service for registering a new user and their mobile device in the system database**

The service is called by POST using the command

http://10.0.4.7:8091/webusers/сгеатеПри service is called, data about the new mobile user is sent to the server in JSON format with the following structure:

int Client\_ID   
string Surname  
 string Firstname  
 string AFM  
 string email   
 string mobile\_phone   
 string password   
 int WU\_ID  
 string Manufacturer   
string Brand   
string Model string   
Board  
 string Hardware   
 string Unique\_ID  
 string ScreenResolution   
 string ScreenDensity   
 string Host   
 string Version   
 string API\_level   
 string Build\_ID   
 string Build\_Time   
 string Fingerprint   
 string PhoneType   
 string NetworkCountryISO   
 string NetworkOperatorName   
 string DeviceId   
 string DeviceSoftwareVersion   
 string SimCountryIso   
 string SimOperatorName   
 string SimSerialNumber   
 string Imei   
 string Meid   
 string MmsUAProfUrl   
 string MmsUserAgent   
 string SubscriberId   
 string TypeAllocationCode  
 string OS  
 string Video  
 int Status   
datetime DateIns  
  
After the request has been processed, the service returns the full set of data about the web user with the structure shown above with the new values of Client\_ID and WU\_ID.

* 1. **Service for registering a new user**The service is called by POST using the command

http://10.0.4.7:8091/webusers/addwebuserПри service is called, data about the new mobile user is sent to the server in JSON format with the following structure:

int Client\_ID   
string Surname  
 string Firstname  
 string AFM  
 string email   
 string mobile\_phone   
 string password

* 1. **A service to determine a web user's ID by any combination of email, mobile, afm, dob, client\_id.**

The service is called using the GET method with the command

http://10.0.4.7:8091/webusers/getwebuser\_id/{'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME', 'client\_id': '1234' }

The parameters email, mobile, afm, dob, client\_id can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter could be omitted.  
  
After the request is fulfilled, the service returns the ID of the found web user.   
If web user with specified parameters is not found 0.   
If several web users with specified parameters   
are found, ID of the last user is returned, i.e. maximal value of found ID   
Using only one parameter dob is technically possible, but it is correct to use it in combination with other parameter, for example afm.

For example,  
http://10.0.4.7:8091/webusers/GetWebUser\_ID/{'email': 'gundartsev@gmail.com'} - Return web user ID by email

http://10.0.4.7:8091/webusers/getwebuser\_id/{'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX'}-returning web user ID by mobile and afm

* 1. **A service to determine a complete set of web user data by any combination of email, mobile, afm, dob, client\_id.**

The service is called using the GET method with the command

http://10.0.4.7:8091/webusers/getwebuser\_data/{'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME', 'client\_id': '1234' }

The parameters email, mobile, afm, dob, client\_id can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter can be omitted.  
  
After the request has been executed, the service returns the complete data set about the web user, with the structure shown below.   
If a web user with the specified parameters is not found, an empty dataset is returned.   
If more than one web user with the specified parameters   
is found, the data set of the last user is returned, i.e. the set with the maximum ID value

Structure of the returned dataset  
id: 1\_Email: ["aaa@bbb.](mailto:vitaly@otenet.gr) gr\_Mobile: "XXXXXXXXXX"\_Password: "2"\_AFM0: "123456789"  
surname: "XXXXXXXXXXXXXX"  
firstname: "XXXXXXXXXX"  
surnameFather: ""  
firstnameFather: ""  
surnameMother: ""  
firstnameMother: ""

doB: "1980-01-01T00:00:00"  
bornPlace: ""  
adt: "AE 176152"  
expireDate: ""  
police: ""  
doy: ""  
afm: "XXXXXXXXX"  
address: ""  
city: ""  
zip: ""  
country\_ID: 0  
tel: ""  
mobile: "XXXXXXXXXX"  
eMail: [aaa@bbb.gr](mailto:vitaly@otenet.gr)  
client\_ID: 11884  
currentStep: 4  
termsAgreement: "TermsAgreement\_202011275616.pdf"  
status: 1  
dateIns: "0001-01-01T00:00:00"

For example,   
http://10.0.4.7:8091/webusers/GetWebUser\_Data/{'mobile': 'XXXXXXXXXXXX'} - return the full set of user web user data by mobile

* 1. **Webuser password change service**

The service is called by POST using the command

http://10.0.4.7:8091/webusers/updatepasswordWhen the service is called, the server receives   
 int wu\_id - webuser ID string password - new password

* 1. **Webuser status change service**

The service is called by POST using the command

http://10.0.4.7:8091/webusers/updatestatusWhen the service is called, the server receives   
 int id - webuser ID int status - new status

On completion the service will return 1 if the change has taken place, 0 if it has not, for example because there is no WebUsers record with the specified ID

* 1. **Service of linking webuser with client.**Service establishes connection between webuser with WU\_ID and client with Client\_ID.   
     Service is called by GET  
      method with command

http://10.0.4.7:8091/webusers/MatchWUC/{'WU\_ID': '1', 'Client\_ID': '234'}

On completion the service returns WU\_ID

* 1. **Registration status retention service**

The service is called by POST using the command

http://10.0.4.7:8091/webusersstates/savedataWhen the service is invoked, the following is sent to the server   
 int wu\_id - webuser ID int status - record status, if not specified, defaults to 0.  
 string email - email, if not specified, then by default it is "  
" string mobile - mobile, if not specified, then by default it is ""   
Returned value: 1 - success, 0 - fail

* 1. **Registration status service**

The service is called using the GET method with the command

http://10.0.4.7:8091/webusersstates/getstatus/123where 123 is the webuser ID

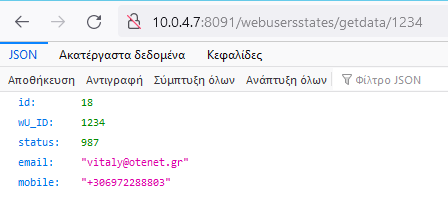
After the request has been processed, the service returns the value status

* 1. **Service to determine all registration status data**

The service is called using the GET method with the command

http://10.0.4.7:8091/webusersstates/getdata/123where 123 is the webuser ID

After the request has been processed, the service returns the registration status in the following format



**2.11 Webuser removal service**

TEMPORARY service - available for the development period

The service deletes the webuser by its ID.  
 The service is called using the GET method with the command

[http://10.0.4.7:8091/webusers/delete?](http://10.0.4.7:8091/webusers/delete?id=) id=123

On completion, the service returns the ID of the deleted record

1. **Services for working with the list of user devices of the mobile app WebUsersDevices**
   1. **Service for new mobile device registration in the system databaseCall**   
      the service by POST   
      method using <http://10.0.4.7:8091/webusersdevices/create> command  
        
      When the service is called, data about new mobile device is transmitted to server in JSON-format with following structure:

WU\_ID int,  
 Manufacturer nvarchar(100),  
 Brand nvarchar(100),  
 Model nvarchar(100),  
 Board nvarchar(100),  
 Hardware nvarchar(100),  
 Unique\_ID nvarchar(100),  
 ScreenResolution nvarchar(100),  
 ScreenDensity nvarchar(100),  
 Host nvarchar(100),  
 Version nvarchar(100),  
 API\_level nvarchar(100),  
 Build\_ID nvarchar(100),  
 Build\_Time nvarchar(100),  
 Fingerprint nvarchar(100),  
 PhoneType nvarchar(100),  
 NetworkCountryISO nvarchar(100),  
 NetworkOperatorName nvarchar(100),  
 DeviceId nvarchar(100),  
 DeviceSoftwareVersion nvarchar(100),  
 SimCountryIso nvarchar(100),  
 SimOperatorName nvarchar(100),  
 SimSerialNumber nvarchar(100),  
 Imei nvarchar(100),  
 Meid nvarchar(100),  
 MmsUAProfUrl nvarchar(100),  
 MmsUserAgent nvarchar(100),  
 SubscriberId nvarchar(100),  
 TypeAllocationCode nvarchar(100),  
 OS nvarchar(100),  
 Video nvarchar(100),  
 Status smallint  
  
Service returns ID of new mobile device upon completion

* 1. **Service to identify a client's device ID by any combination of that client's ID, email, mobile, afm, dob.**

The service is called using the GET method with the command

[http://10.0.4.7:8091/webusersdevices/getwebusersdevices\_id/{'id](http://10.0.4.7:8091/webclientsdevices/getwebusersdevices_id/%7b'id)': '123', 'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME' }

The parameters ID, email, mobile, afm, dob can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter can be omitted  
  
After the request has been processed the service returns the ID of the found device.   
If no device with specified client parameters is found, 0 is returned.  
If several devices with specified client parameters are found, ID of last device   
is returned, i.e. maximal value of found ID   
Using only one parameter dob is technically possible, but it is correct to use it in combination with other parameter, for example afm.

For example,  
http://10.0.4.7:8091/webusersdevices/GetWebUsersDevices\_ID /{'id': 1234} - Returns the device ID by the ID of the client the device belongs to

http://10.0.4.7:8091/ webusersdevices/GetWebUsersDevices\_ID /{'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX'}-return the device ID by mobile and afm of the client that this device belongs to

* 1. **A service to identify a client's complete device data set by any combination of that client's ID, email, mobile, afm, dob.**

The service is called using the GET method with the command

[http://10.0.4.7:8091/webusersdevices/getwebusersdevices\_data/{'id](http://10.0.4.7:8091/webclientsdevices/getwebusersdevices_data/%7b'id)': '123', 'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME' }

The parameters ID, email, mobile, afm, dob can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter could be omitted.  
  
After the request has been processed the service returns the full data set of the devices found.   
If no device with specified client parameters was found, the service returns 0.   
If several devices with specified client parameters were found, the list of devices with all their characteristics is returned.   
Using only one parameter dob is technically possible, but it is correct to use it in combination with another parameter, e.g. afm.

For example,  
http://10.0.4.7:8091/webusersdevices/GetWebUsersDevices\_Data /{'id': 1234} - Returns the device characteristics by the client ID of the client the device belongs to

http://10.0.4.7:8091/ webusersdevices/GetWebUsersDevices\_Data /{'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX'}- Returns the device characteristics of the mobile and afm client to which this device belongs  
  
Example

A picture containing text

Description automatically generated

* 1. **Service to identify a client's device ID list by any combination of that client's ID, email, mobile, afm, dob.**

The service is called using the GET method with the command

[http://10.0.4.7:8091/webusersdevices/getwebusersdeviceslist\_id/{'id](http://10.0.4.7:8091/webclientsdevices/getwebusersdeviceslist_id/%7b'id)': '123', 'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME' }

The parameters ID, email, mobile, afm, dob can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter can be omitted  
  
After the request has been processed the service returns the list of found device IDs.   
If no devices with specified client parameters were found, the empty dataset is returned.   
Using only one parameter dob is technically possible, but it is correct to use it in combination with another parameter, e.g. afm.

For example,  
http://10.0.4.7:8091/webusersdevices/GetWebUsersDevicesList\_ID /{'id': 1234} - Returns the device ID list by the client ID of the device

http://10.0.4.7:8091/ webusersdevices/GetWebUsersDevicesList\_ID /{'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX'}- Returns the device ID list by mobile and afm of the client that device belongs to

* 1. **A service to define a list of complete customer device datasets using any combination of that customer's ID, email, mobile, afm, dob.**

The service is called using the GET method with the command

[http://10.0.4.7:8091/webusersdevices/getwebusersdeviceslist\_data/{'id](http://10.0.4.7:8091/webclientsdevices/getwebusersdeviceslist_data/%7b'id)': '123', 'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME' }

The parameters ID, email, mobile, afm, dob can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter may be omitted.  
  
After the request has been processed the service returns the list of the full datasets of the devices found.   
If no devices with specified client parameters were found, the empty set is returned.   
Using only one parameter dob is technically possible, but it is correct to use it in combination with another parameter, e.g. afm.

For example,  
http://10.0.4.7:8091/webusersdevices/GetWebUsersDevicesList\_Data /{'id': 1234} - returns the list of device characteristics by the client ID of the device

http://10.0.4.7:8091/ webusersdevices/GetWebUsersDevicesList\_Data /{'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX'}- returns a list of device characteristics by mobile and afm of the client to which this device belongs

* 1. **Service for determining a list of complete device datasets by their unique\_ID**

The service is called using the GET method with the command

http://10.0.4.7:8091/webusersdevices/GetList\_Unique\_ID/{'unique\_ID': '4d691c8f74c87626af'}

After the request has been processed the service returns a list of complete datasets of found devices.   
If no devices with the specified client parameters are found, an empty set is returned.  
 **Graphical user interface, text, application, email

Description automatically generated**

* 1. **Service to determine the status of a device by its unique\_ID**

The service is called using the GET method with the command

[http://10.0.4.7:8091/webusersdevices/GetDeviceStatus/{'unique\_ID': '9182nb0k2751qa8hl45'}](http://10.0.4.7:8091/webusersdevices/GetDeviceStatus/%7b'unique_ID':%20‘9182nb0k2751qa8hl45'%7d)

where unique\_ID is value of unique\_ID parameter.   
  
After request processing service returns status of found device:   
1 - device is active,   
2 - device is blocked,   
0 - device status could not be determined. For example, device with specified unique\_ID is not found, etc.  
  
As the unique\_ID   
parameter may not be unique, the service determines the status of the first found device with the specified unique\_ID. All devices with the same unique\_ID should have the same status if there are no previous errors

* 1. **Service for changing any device data by device ID**

The service is called by POST using the command

[h ttp://10.0.4.7:8091/webusersdevices/update](http://10.0.4.7:8091/webusersdevices/update)

When the service is called, the device id, i.e. WebUsersDevices. ID as well as data to be changed, in JSON format with the following structure:  
 {'id':123 , 'wu\_id': 274, 'Manufacturer': 'Apple Inc', 'Brand' : 'Apple Inc ', ....}

where .... values of any data about this device. All possible data are listed in point 3.1

* 1. **Service for changing the WebUser ID (WU\_ID) value.**

The service is called by POST using the command

http://10.0.4.7:8091/webusersdevices/updatewu\_idWhen the service is called to the server, the   
 int id   
 int wu\_id,  
where id is the ID of the device whose status is to be changed wu\_id is the new value of the WebUser ID

* 1. **Mobile device status change service**

This service allows you to change the status of a mobile device by its ID value and by or by the value of Unique\_ID field. Only the value of one of these parameters should be specified. If both parameters values are specified by mistake during server call, Unique\_ID field value will be ignored.  
As the device ID is unique, the execution of this service may result in status changes in only one record of the WebUsersDevices table when the ID value is specified.  
Since Unique\_ID field value is not unique, if Unique\_ID value is specified, execution of this service may result in status change in all WebUsersDevices table entries with such Unique\_ID.

The service is called by POST using the command

http://10.0.4.7:8091/webusersdevices/updatestatusWhen the service is called to the server, the   
 string id,

string unique\_id,  
 int status,  
where id is the device ID field

unique\_id- the Unique\_ID field,

status - 1 - device active, 2 - device locked

* 1. **Device removal service.**

The service is called by POST using the command

http://10.0.4.7:8091/webusersdevices/deleteWhen the service is called, the server receives   
 int id - ID of record to be deleted

1. **Services for working with a company's client list Clients**
   1. **Customer ID service for any combination of email, mobile, afm, dob.**

The service is called using the GET method with the command

http://10.0.4.7:8091/clients/getclient\_id/{'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME' }

The parameters email, mobile, afm, dob can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter can be omitted  
  
After request processing service returns the ID of found client.   
If client with specified parameters is not found 0.   
If several clients with specified parameters are found, ID of last client   
is returned, i.e. maximal value of found ID.   
Using only one parameter dob is technically possible, but it is correct to use it in combination with other parameter, for example afm.

For example,  
http://10.0.4.7:8091/clients/GetClient\_ID/{'email': 'gundartsev@gmail.com'} - Return customer ID by email

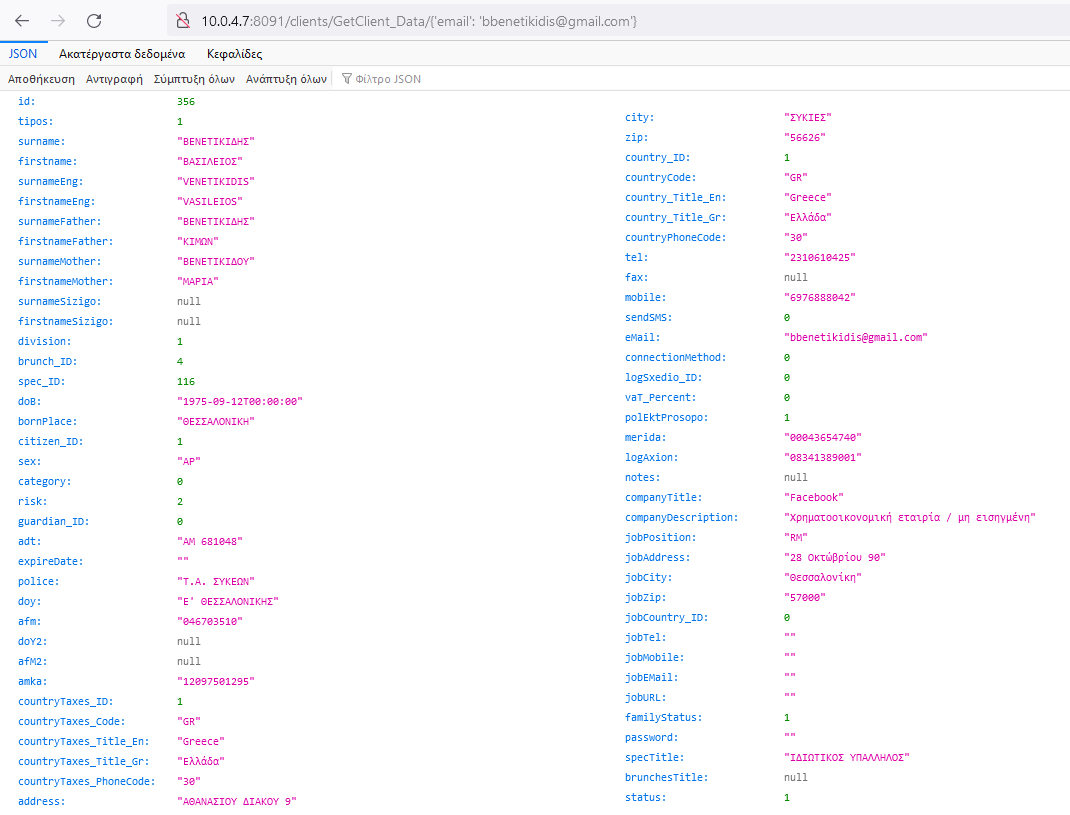
http://10.0.4.7:8091/client/getclient\_id/{'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX'}-returning the client ID by mobile and afm

* 1. **Service to determine the full customer data set by any combination of email, mobile, afm, dob.**

The service is called using the GET method with the command

http://10.0.4.7:8091/clients/getclient\_data/{'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME' }

The parameters email, mobile, afm, dob can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter can be omitted  
  
After the request has been executed the service returns the full client   
data set with the structure shown below.   
If a client with the specified parameters is not found, the service returns an empty dataset.   
If several clients with the specified parameters   
are found, the data set of the last client, i.e. the data set with the maximum ID value is returned

Structure of the returned data set  


For example,   
http://10.0.4.7:8091/webusers/GetWebUser\_Data/{'mobile': '6971234567'} - return the complete web user data set by mobile device number 6971234567 .

* 1. **A service to determine the full set of data from a client list by any combination of email, mobile, afm, dob.**

The service is called using the GET method with the command

http://10.0.4.7:8091/clients/getclient\_list/{'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME' }

The parameters email, mobile, afm, dob can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter can be omitted  
  
After the request has been executed the service returns the full set of client list data with the structure as shown below.   
If no clients with specified parameters are found, the empty dataset is returned.   
If several clients with the specified parameters are found, the service returns the list of all found clients with their data

The structure of the returned data set consists of several data groups with the same structure. The number of data groups depends on the number of clients found.

For example,   
http://10.0.4.7:8091/clients/getclient\_list/{'email': 'ceo@hellasfin.gr'} - Return full dataset from the list of clients with the specified email

The resulting data set consists of 3 groups of data. Group one is customer data with ID = 277, group two is customer data with ID = 3227, group three is customer data with ID = 9085

Customer type is set in the tipos field: 1 - natural person, 2 - legal person

* 1. **A service for entering communication between two clients.**

The service is called by POST using the command

<http://10.0.4.7:8091/clients/AddClients_Clients>

When the service is called, the IDs of both clients are transmitted to the server in JSON format with the following structure:

Client\_ID int,  
 Client2\_ID int,  
 Status int (0 - not confirmed, 1 - confirmed) ,   
DateIns smalldatetime

where Client\_ID is the ID of the client making the invitation, Client2\_ID is the ID of the client being invited, Status = 0,  
 DateIns is the date this entry was entered in the database

* 1. **A service for changing the communication status between two clients.**

The service is called by POST using the command

<http://10.0.4.7:8091/clients/UpdateStatusClients_Clients>

When the service is called, the following is sent to the server   
 int id - record ID int Status - new status

* 1. **Service to define a list of all links for a specified client with ID**

The service is called using the GET method with the command

[h ttp://10.0.4.7:8091/clients/GetClients\_Clients/id=269](http://10.0.4.7:8091/clients/GetClients_Clients/id=269)  
  
where ID is Client\_ID of client for which the list of connections is requested.  
  
After the request has been processed, the service returns data on all found relations, indicating Client\_ID, Client\_NameYu and their AFM and e-mail

* 1. **A service to remove communication between clients.**

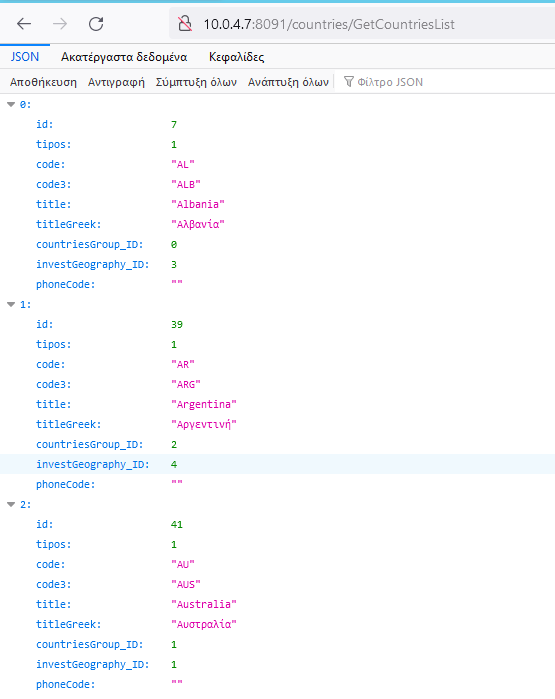
The service is called by POST using the command

http://10.0.4.7:8091/clients/DeleteClients\_ClientsWhen the service is called, the server receives   
 int id - ID of the connection to be deleted

1. **Services for working with the Countries list**
   1. **Country list detection service**

The service is called using the GET method with the command

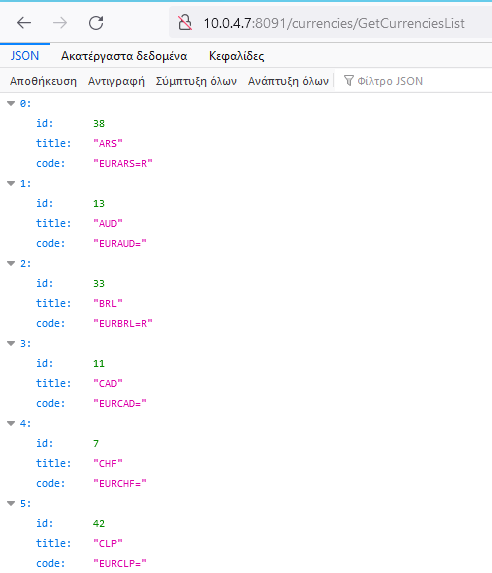
http://10.0.4.7:8091/countries/GetCountriesList

After the request has been processed, the service returns a list of all the countries:  
  


1. **Currencies list services**
   1. **Currency list service**

The service is called using the GET method with the command

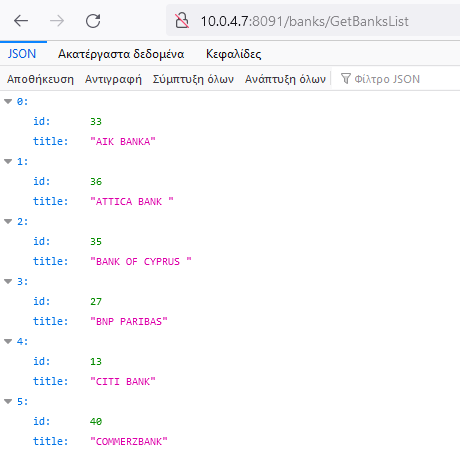
http://10.0.4.7:8091/currencies/GetCurrenciesList

After the request has been processed, the service returns a list of all currencies:  
  


1. **Services for working with a list of banks**
   1. **Bank list identification service**

The service is called using the GET method with the command

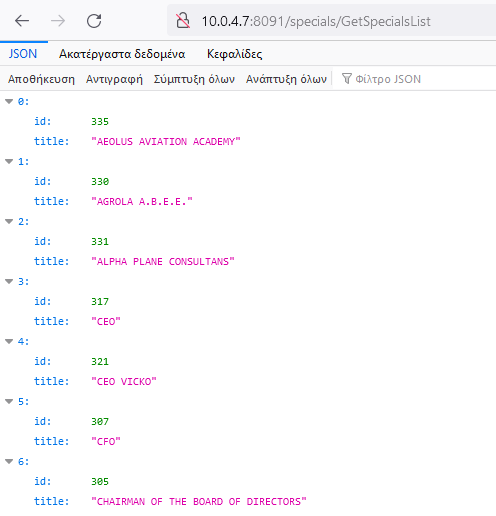
http://10.0.4.7:8091/banks/GetBanksList

After the request has been processed, the service returns a list of all banks:  
 ****

1. **Services for working with the list of specials**
   1. **Service for identifying a list of professions**

The service is called using the GET method with the command

http://10.0.4.7:8091/specials/GetSpecialsList

The service returns a list of all professions after the query has been processed:  
  
****

1. **Services for working with Brunches client industry list**
   1. **Service for identifying a list of sectors**

The service is called using the GET method with the command

[h ttp://10.0.4.7:8091/brunches/GetBrunchesList](http://10.0.4.7:8091/brunches/GetBrunchesList)  
  
After the request has been processed, the service returns a list of all industries:

**Graphical user interface, text, application

Description automatically generated**

**10. Services for working with ClientsDocFiles document list  
  
10.1 Service of document list definition by any combination of client\_id, contract\_id, precontract\_id, doctypes\_id parameters.**

The service is called using the GET method with the command  
  
http://10.0.4.7:8091/clientsdocfiles/GetClientsDocFilesList/{'client\_id': '356', 'contract\_id': '0', 'precontract\_id': '0', 'doctypes\_id': '3924' }

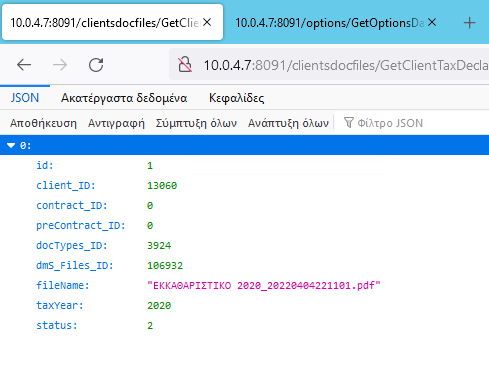
where doctypes\_id is the document type code. For example, 3924 is Εκκαθαριστικό  
  
Parameters client\_id, contract\_id, precontract\_id, doctypes\_id can be specified in any order and in any combination. At least one parameter is mandatory.   
Any parameter can be omitted.  
  
After query processing service returns list of found documents with structure shown below. **Graphical user interface, text, application

Description automatically generated**

**10.2 Service for determining the list of ΕΚΚΑΘΑΡΙΣΤΙΚΟ by client\_id, doctypes\_id**

The service is called using the GET method with the command  
  
http://10.0.4.7:8091/clientsdocfiles/GetClientTaxDeclarationsList/{'client\_id': '13060', 'doctypes\_id': '3924', 'start\_year': '2016', 'last\_year': '2021' }  
  
where client\_id is the client ID doctypes\_id is the document type code. For example, 3924 is Εκκαθαριστικό start\_year - starting year last\_year - ending year

After the request has been processed, the service returns a list of found files with the structure shown below.



1. **Services for dealing with the list of Contracts' clients' investment accounts**

**11.1 Service for defining the list of investment accounts**The service is called by the GET method using the command

http://10.0.4.7:8091/webusers/getwebuser\_id/{'email': 'gundatsev@gmail. ru', 'mobile': 'XXXXXXXXXXXX', 'afm': 'XXXXXXXXXXX', 'dob': 'YYYY-MM-DD TIME', 'client\_id': '1234' }

The parameters email, mobile, afm, dob, client\_id can be specified in any order and in any combination. At least one parameter must be specified.   
Any parameter may be omitted

1. **ClientsBankAccounts services for the list of clients' bank accounts**
   1. **Service for identifying a customer's bank account list by its Client\_ID**

The service is called using the GET method with the command

http://10.0.4.7:8091/ClientsBankAccounts/GetClientBankAccountsList\_Data/{'client\_id':'356}  
  
where client\_id is the client ID of the system.

In response, the system returns a list of all active bank accounts of this customer with their basic data:  
ID - account ID in the database  
tipos - 0 - individual, 1 - collective

Bank\_ID - Bank ID

Bank\_Title - name of the bank

Client\_ID -

AccNumber - bank account number

Currency - currency of the bank account

**Graphical user interface, text, application, email

Description automatically generated**

* 1. **Service for entering (adding) a new bank account**The service is called by POST method using the command

[http://10.0.4.7:8091/ClientsBankAccounts](http://10.0.4.7:8091/ClientsBankAccounts%20/addrecord)  
 /adreadWhen the service is called, data about the new mobile user is sent to the server in JSON format with the following structure:

int Client\_ID - ID of the customer to whom this bank account belongs

int Bank\_ID - ID of the bank where this bank account is opened

string Currency - currency of the bank account

After processing, the service returns the new bank account ID

* 1. **Bank account status change service**

The service changes the status of a bank account by its ID.  
 The service is called by POST using the command

[http://10.0.4.7:8091/ ClientsBankAccounts /EditRecord](http://10.0.4.7:8091/%20ClientsBankAccounts%20/)  
  
When the service is called, data about the new mobile user is transmitted to the server in JSON format with the following structure:

int ID - ID of this bank account record

int Status - new status of the bank account.   
 A value of 0 means the account is cancelled, 1 means the account is active

The service returns the ID of this bank account record when successfully completed. If the status has not been changed for any reason, 0 is returned

**13. Services for working with a list of client request types ClientsRequestsTypes**HF clients can send different types of requests:

|  |  |  |  |
| --- | --- | --- | --- |
| Type of request | Name of request | DocType1\_ID | DocType2\_ID |
| 1 | Αίτημα αλλαγής ταυτότητας | 2 | 0 |
| 2 | Αίτημα αλλαγής κινητού τηλεφώνου | 9418 | 9328 |
| 3 | Αίτημα αλλαγής σταθερού τηλεφώνου | 9418 | 9328 |
| 4 | Αίτημα αλλαγής email επικοινωνίας | 9431 | 0 |
| 5 | Αίτημα αλλαγής διεύθυνσης κατοικίας | 0 | 9328 |
| 6 | Αίτημα αλλαγής ΑΦΜ | 1 | 0 |
| 7 | Αίτημα προσθήκης εκκαθαριστικού | 3924 | 0 |
| 8 | Αίτημα αλλαγής επαγγέλματος | 4 | 0 |
| 9 | Αίτημα αλλαγής xώρας φορολόγησης | 1 | 0 |
| 10 | Αίτημα αλλαγής ειδικής κατηγορίας προσώπου | 0 | 0 |
| 11 | Αίτημα προσθήκης νέου τραζεικού λογαριασμού | 5 | 0 |
| 12 | Αίτημα προσθήκης νέου συνδεδεμένου ανήλικου προσώπου | 9432 | 9432 |
| 13 | Αίτημα αλλαγής W8 ΒΕΝ | 58 | 0 |
| 14 | Αίτημα αλλαγής Διαβατηρίου | 9267 | 0 |
| 15 | Αίτημα αλλαγής Χρημηματιστηριακής Μερίδας | 0 | 0 |
| 16 | Αίτημα αλλαγής Λογαριασμόυ Αξιών | 0 | 0 |
| 17 | Αίτημα αλλαγής ΑΜΚΑΑ | 9419 | 0 |
| 18 |  |  |  |
| 19 | Άνοιγμα Νέου Επενδυτικού Λογαριασμού | 0 | 0 |

1. **Services for working with client request lists ClientsRequests**

**14.1 Service for entering a new request from a client**The service is called by POST using the command

http://10.0.4.7:8091/clientsrequests/AddClientsRequests  
  
When the service is called, data about the new request is transmitted to the server in JSON format with the following structure:

**Client\_ID** int - client\_id  
**Tipos** smallint - type of request (the list is given above)  
**Aktion** smallint - action (0 - insertion of a new request, 1 - modification of the request, 2 - deletion)  
**Source\_ID** int - who entered the request: 1 - the client, 2 - our employee   
**Description** nvarchar (1000) - contents of the request. It is different for each request type. The detailed description is given below  
**Warning** nvarchar(500) - the message that will be written by our employee that checks the request in case of correct request input  
**DateIns** smalldatetime - date of request entering  
**DateClose** smalldatetime - date when the request is closed (executed)  
**User\_ID** int - ID of our employee that entered the request in the case Source\_ID = 2. If the request was submitted by the client himself/herself, i.e. Source\_ID = 2, then User\_ID = 0  
**Status** smallint - status of the request:   
1 - the request is under preparation (draft)   
2 - the request is sent to responsible persons (Compliance, BackOffice)  
3 - the request was checked and executed (closed)  
4 - the request was checked and rejected, as incorrectly entered. The comment of the checker is stored in the field Warning0   
- the request is cancelled, deleted for the client

If the query is entered successfully, the service returns the ID of the new query when it is finished, otherwise it returns 0.  
**Description**   
field Description. Description   
field is intended to describe data of all types of queries.   
It is a string field, containing data in JSON format. A string consists of several elements, each with ~ at the end.

Basically, MobiApp will indicate in the Description field the set of file-documents that the user downloads. But there are client requests where the Description field should contain other data.   
Sets of document files can contain a variable number of files separated by^ . Each file is described by two attributes: file name and file type.   
For example, {'file\_name' : 'pay\_1.pdf','file\_type' : 25}^{'file\_name' : 'bill\_2.pdf','file\_type' : 83}^ describes two files pay\_1.pdf and bill\_2.pdf. The first one has document type 25 and the second one has document type 83. The first file is described as   
{'file\_name' : 'pay\_1.pdf','file\_type' : 25}^ and   
the second file is described as   
{'file\_name' : 'bill\_2.pdf','file\_type' : 83}^

In addition to file-document sets, the Description field may contain an action attribute indicating the type of action, for example,   
 {'action ' : '1'}~where   
0-enter new value, 1-change, 2-delete,  
  
as well as an answer to the question whether the user is the owner of the mobile or home   
{'owner' : '0'}~0   
means no, 1 means yes

The contents of the Description field for each specific query type:  
  
- tavtotita number replacement  
{'file\_name' : 'tavtotita\_17-12-2021\_15-36-47.png','file\_type' : 2}^{'file\_name' : 'tavtotita\_17-12-2021\_15-36-47.png','file\_type' : 2}^~

- mobile phone number replacement  
{'action' : '0'}~ {'owner' : '0'}~{'new\_number' : '6971234567'}~{'file\_name' : 'mobile\_1.pdf','file\_type' : 9418}^{'file\_name' : 'mobile\_2.pdf','file\_type' : 52}^~ {'file\_name' : 'file3.pdf','file\_type' : 9418}^{'videochat' : '0'}~

- replacement landline number{  
'owner' : '0'}~ {'new\_number' : '2310234567'}~ {'file\_name' : 'tel\_1.pdf','file\_type' : 9418}^{'file\_name' : 'tel\_2.pdf','file\_type' : 31}^~ {'file\_name' : 'file3.pdf','file\_type' : 9418}^~

- email replacement   
 {'new\_email' : 'newemail@yahoo.com'}~{'videochat' : '0'}~

-home address change  
 {'owner' : '0'}~{'file\_name' : 'add\_20211221141836.pdf','file\_type' : 4}^{'file\_name' : 'add\_20211221141924.pdf','file\_type' : 4}~{'file\_name' : 'yd\_20211221141958.png','file\_type' : 7}^{'file\_name' : 'yd\_20211221142041.png','file\_type' : 7}^~

- ΑΦΜ replacement  
 {'file\_name' : 'afm\_17-12-2021\_15-36-47.png','file\_type' : 1}^{'file\_name' : 'afm\_17-12-2021\_15-36-47.png','file\_type' : 1}^~

- input ΕΚΚΑΘΑΡΙΣΤΙΚΟ  
 {'tax\_declaration' : '0'}~{ ΕΚΚΑΘΑΡΙΣΤΙΚΟ 2018.pdf^ΕΚΚΚΑΘΑΡΙΤΙΚΟ 2019.pdf^}~{ 'file\_name' : 'abc.pdf', 'file\_type' : '7'}^{'file\_name' : 'def.pdf', 'file\_type' : '8'}^~

- specialty replacement  
 {'new\_spec\_id' : '3'}~ {'file\_name' : 'spec\_17-12-2021\_15-36-47.png','file\_type' : 4}^{'file\_name' : 'spec\_17-12-2021\_15-36-47.png','file\_type' : 4}^~

- tax country replacement  
 {'new\_country\_id' : '25'}~ {'file\_name' : 'tax\_17-12-2021\_15-36-47.png','file\_type' : 1}^{'file\_name' : 'tax\_17-12-2021\_15-36-47.png','file\_type' : 1}^~

- replace special client category  
{'cat' : '0','file\_name' : ''}~{'file\_name' : 'dilosi.pdf'}~{'cat' : '1','file\_name' : 'abc.pdf'}~{'cat' : '2','file\_name' : 'def.pdf'}~ where cat is the selected category number   
file\_name is the name of the corresponding file  
  
- replace the passport number   
 {'file\_name' : 'psp\_17-12-2021\_15-36-47.png','file\_type' : 9267}^{'file\_name' : 'psp\_17-12-2021\_15-36-47.png','file\_type' : 9267}^~

- Μερίδα replacement  
 {'file\_name' : 'psp\_17-12-2021\_15-36-47.png','file\_type' : 0}^{'file\_name' : 'psp\_17-12-2021\_15-36-47.png','file\_type' : 0}^~

- Λογαριασμόυ Αξιών  
 {'file\_name' : 'psp\_17-12-2021\_15-36-47.png','file\_type' : 0}^{'file\_name' : 'psp\_17-12-2021\_15-36-47.png','file\_type' : 0}^~

- replace ΑΜΚΑ  
 {'file\_name' : 'psp\_17-12-2021\_15-36-47.png','file\_type' : 9419}^{'file\_name' : 'psp\_17-12-2021\_15-36-47.png','file\_type' : 9419}^~

- entering a new bank account   
 {'file\_name' : 'ba\_17-12-2021\_15-36-47.png','file\_type' : 0}^{'file\_name' : 'ba\_17-12-2021\_15-36-47.png','file\_type' : 0}^~

- Change of bank account status   
{'action' : '2'}~{id' : '156', 'status' : '0'}~

- entering (adding) an adult HellasFin client to the list ΣΥΝΔΕΔΔΕΜΕΝ ΠΡΟΣΩΠΠΑ   
{'client\_id' : '356'}~

- entering (adding) a child to the ΣΥΝΝΔΕΔΔΕΜΕΕΑΝ ΠΡΟΣΩΠΑ involves loading different documents. In general, the format of the entry will be as follows: {'surname' : 'DELLAS'}~{'firstname' : 'NIKOS'}~{'dob' : '15/04/2009'}~{'file\_name' : file1.pdf','file\_type' : 9432}^{'file\_name' : file2.pdf','file\_type' : 2}^{'file\_name' : file3.pdf','file\_type' : 3968}^{'file\_name' : file4.pdf','file\_type' : 55}^{'file\_name' : file5.pdf','file\_type' : 1}~

Where 9432 - permission from the father, 9432 - permission from the mother, 3968 - tautota, 55 - birth certificate, 1 - document from the AFM

------------------Требуют   
уточнения---------------------13 Αίτημα αλλαγής W8 ΒΕΝ

-------------------------------------------------------------------

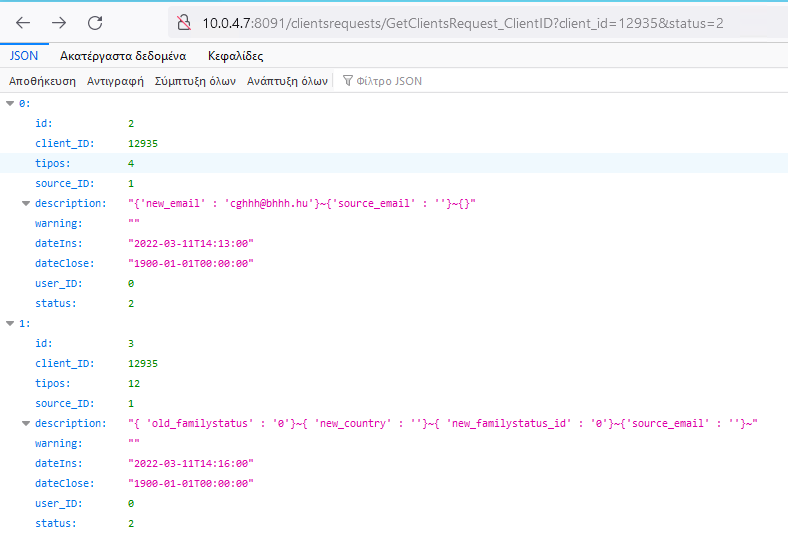
**14.2 Service to determine (read) request data from client by request ID**The service is called by the GET method using the command

[http://10.0.4.7:8091/ clientsrequests /getclientsrequest\_data](http://10.0.4.7:8091/%20clientsrequests%20/get)? id=1234  
where id is request record ID  
  
After the request has been processed, the service returns all data about the request with the specified ID



**14.3 Service to determine (read) request data by Client\_ID, i.e. request author's ID**Service is called by GET method using command

[http://10.0.4.7:8091/ clientsrequests /GetClientsRequest\_ClientID](http://10.0.4.7:8091/%20clientsrequests%20/)?client\_id=4469&status=2  
where client\_id is ID of client - author of request. This parameter is mandatory. status - request status. The values of this parameter are given in section 14.1.   
This parameter is optional.   
After request processing service returns data sets about client's requests with specified client\_ID. If status is also specified, for example, status=2, only requests with specified status are included in the result set.



**14.4 Service to change the status of a request from a client**The service is called by POST using the command

[http://10.0.4.7:8091/ clientsrequests/updatestatus](http://10.0.4.7:8091/%20clientsrequests/updatestatus)When the service is called, the server receives   
 int id   
 int status,  
where id is request record ID which status should be changed  
 status - request status. Possible values for this status are described in 14.1.  
Only statuses 2 and 0 are set   
in MobiApp. status 2 is set when a new request is entered, i.e. in the AddClientsRequests service, and status 0 when it is cancelled (deleted).

1. **Services for working with the Options list**
   1. **System parameter list service**

The service is called using the GET method with the command

<http://10.0.4.7:8091/options/GetOptionsData>

After the request has been processed, the service returns the list of parameters shown below

