**IdealQ**

**REST API Specification Doc**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 1.0 | 30-Jul-2016 | Adrian Ionita | Initial draft |

Table of Contents

[1. POST login 3](#_Toc457692382)

[Request 3](#_Toc457692383)

[Response 3](#_Toc457692384)

[2. GET ClientInfo 4](#_Toc457692385)

[Request 4](#_Toc457692386)

[Response 4](#_Toc457692387)

[4. POST Deschidere client nou 4](#_Toc457692388)

[Request 4](#_Toc457692389)

[Response 5](#_Toc457692390)

[Conventions 6](#_Toc457692391)

[Status Codes 6](#_Toc457692392)

## 

Methods

## 1. POST login

Authenticate the user with the system and obtain the authKey

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** | http://desktop-jo3h85m:7101/IdeallQ-IdeallQViewController-context-root/IveloResources/iveloBoRest/login |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| FORM\_PARAM  FORM\_PARAM  FORM\_PARAM | CallingSystem  username  password | string  string  string |

**api\_key**

api\_key trebuie trimis la toate requesturile. Reprezinta codul aplicatiei apelante

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | {  "authKey": <auth\_key>,  “clientId”: <client\_id>  }  auth\_key (**string**) - all further API calls must have this key in header |
| 403 | {"error":"API key is missing."} |
| 400 | {"error":"Please provide username."} |
| 400 | {"error":"Please provide password."} |
| 401 | {"error":"Invalid API key."} |
| 401 | {"error":"Incorrect username or password."} |
| 500 | {"error":"Something went wrong. Please try again later."} |

## 2. GET ClientInfo

Returneaza informatiile de baza despre un anumit client

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **GET** |  |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| URL\_PARAM | <client\_id> | number |

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | * **ClientJSONObject - va contine informatiile de baza despre un client** |
| 401 | {"error":"Invalid API key."} |
| 500 | {"error":"Something went wrong. Please try again later."} |

## 3. POST Deschidere client nou

## Request

|  |  |
| --- | --- |
| **Method** | **URL** |
| **POST** |  |

|  |  |  |
| --- | --- | --- |
| **Type** | **Params** | **Values** |
| FORM\_PARAM  FORM\_PARAM  FORM\_PARAM  FORM\_PARAM  FORM\_PARAM  FORM\_PARAM  FORM\_PARAM | <username>  <password>  <CallingSystem>  <categorieClient>  <nume>  <prenume>  <telefon> | string  string  string  string  string  string  string |

## Response

|  |  |
| --- | --- |
| **Status** | **Response** |
| 200 | **<OpeningClientJSONObject>** |
| 400 | {"error":"Please provide client\_id."} |
| 400 | {"error":"Invalid client\_id."} |
| 401 | {"error":"Invalid Auth key."} |
| 500 | {"error":"Something went wrong. Please try again later."} |

**OpeningClientJSONObject** field specification

authKey String – reprezinta codul de autentificare necesar la apelul altor servicii post

clientId Number – reprezinta id-ul clientului necesar pentru diferite servicii get sau post

Glossary

## Conventions

* **Client** - Client application.
* **Status** - HTTP status code of response.
* All the possible responses are listed under ‘Responses’ for each method. Only one of them is issued per request server.
* All response are in JSON format.
* All request parameters are mandatory unless explicitly marked as [optional]
* The type of values accepted for a *request* parameter are shown the the values column like this [**10**|<any number>] .The | symbol means *OR*. If the parameter is [optional], the default value is shown in blue bold text, as **10** is written in [**10**|<any number>].

## Status Codes

All status codes are standard HTTP status codes. The below ones are used in this API.

2XX - Success of some kind

4XX - Error occurred in client’s part

5XX - Error occurred in server’s part

|  |  |
| --- | --- |
| **Status Code** | **Description** |
| 200 | OK |
| 201 | Created |
| 202 | Accepted (Request accepted, and queued for execution) |
| 400 | Bad request |
| 401 | Authentication failure |
| 403 | Forbidden |
| 404 | Resource not found |
| 405 | Method Not Allowed |
| 409 | Conflict |
| 412 | Precondition Failed |
| 413 | Request Entity Too Large |
| 500 | Internal Server Error |
| 501 | Not Implemented |
| 503 | Service Unavailable |