Control de documentación

En este apartado se encuentra el historial de versiones y cambios de este documento.

**Datos generales**

|  |  |
| --- | --- |
| Autor(es): | Alfredo Bolio |
| Autorizador(es): | Alfredo Bolio |
| Fecha: | 3/13/2020 |

**Historial de versiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Estado** | **Responsable** |
| 1.0.0 | 3/13/2020 | R | Alfredo Bolio |
|  |  |  |  |
|  |  |  |  |

Estado: [R]edacción, [E]spera, [A]probado.

**Historial de cambios**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Cambios** | **Responsable** |
| 1.0.0 | 3/13/2020 | Primera versión | Alfredo Bolio |
|  |  |  |  |
|  |  |  |  |

**ABOUT THE API**  
This is version 1.0 of WEB II API. Below you will find a current list of the available methods.

The Applications request are made through the link **https://127.0.0.1/1.0** followed by the method and its required variables.

All methods require the sending of this keys on the header:

|  |  |  |
| --- | --- | --- |
| Key | Value | Required |
| x-api-key | **String** | Required |
| Content-Type | **Application/json** | Required |

All methods require the sending of a JSON (application/json) input defined for each one.

Client methods require an additional user key

|  |  |  |
| --- | --- | --- |
| Key | Value | Required |
| user-api-key | **String** | Required |

**ERROR RESPONSE**

All methods can return and error on the result field in case of and invalid request or missing required fields. In case of an invalid key it always return a 401 http response. In other cases it will return a 400 http response.

## Responsesapplication/json

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 400  401 | |  |  |  |  | | --- | --- | --- | --- | | result | String (error) |  | Required | |

**CLIENT LOGIN**

**POST** /client/login  
Validate the credentials of a user, for access to the application.

**Url:** https://127.0.0.1/1.0/client/login

Authentication

API Key

Request Body application/json

|  |  |  |  |
| --- | --- | --- | --- |
| Object |  |  |  |
| user | String | Client Email or cellphone | Required |
| password | String (MD5) | Client password in MD5 | Required |

## Responsesapplication/json

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 200 | |  |  |  |  | | --- | --- | --- | --- | | result | String |  |  | | client | **Object** |  |  | | user\_id | int |  | Optional | | user\_api\_key | String | API KEY of the client |  | |

## Request Exampleapplication/json

{

"user": "alfred@hotmail.com",

"password": "893ejrf20jdc13851f940f8df710c4c7"

}

## Response Exampleapplication/json **200**

{

"result": "sucess"

"client": {

"user\_id": 1915,

"user\_api\_key": “jeYeriey2389018shdjksui12794uisdjkh1!werwe31”

}

}

## **400**

{

"error": "Invalid credentials"

}

**CLIENT CREATE PIN**

**POST** /client/list  
Create a PIN (access code) for the client

**Url:** https://127.0.0.1/1.0/client/movies

Authentication

API Key

USER API Key

Request Body application/json

|  |  |  |  |
| --- | --- | --- | --- |
| Object |  |  |  |
| user | String | Client Email | Required |
| password | String (MD5) | Client password in MD5 | Required |

## Responsesapplication/json

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 200 | |  |  |  |  | | --- | --- | --- | --- | | result | String(success) |  | Optional | | movies | Object |  | Optional | | id | String |  | Optional | | title | String |  | Optional | | releaseDate | String |  | Optional | | imageU.0rl | Url |  | Optional | | description | String |  | Optional | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 400 | |  |  |  |  | | --- | --- | --- | --- | | result | String(error) |  | Optional | |

## Request Exampleapplication/json

{

"user": "prueba@hotmail.com",

"password": "893ejrf20jdc4c7"

}

## Response Exampleapplication/json **200**

{

"result": "The pin was set correctly"

}

## **400**

{

"result": "error"

}