**WebAPI documentation**

**Login**

Request: <address:port>/api

body: {

“requestType”: “login” (string),

“login”: (string),

“password”: (string)

}

Response:

body: {

“result”: “success”/”failure” (string),

“operationResultMessage”: (string) (debug info),

“cookie”: (string) (random 256 character string)

}

**Get user**

Request: <address:port>/api

body: {

“requestType”: “getUser” (string),

“cookie”: (string),

“userLogin”: (string)

}

Response:

body: {

“result”: “success”/”failure” (string),

“operationResultMessage”: (string) (debug info),

“data”: { “login”: (string), “username” (string), “role”: (string) }

}

**Edit/Add user**

Request: <address:port>/api

body: {

“requestType”: “editUser”/”addUser” (string),

“cookie”: (string),

“userLogin”: (string) (“None” if adding new),

“data”: { “login”: (string), “username” (string), “password”: (string), “role”: (string) }

}

Response:

body: {

“result”: “success”/”failure” (string),

“operationResultMessage”: (string) (debug info)

}

**Delete user**

Request: <address:port>/api

body: {

“requestType”: “deleteUser” (string),

“cookie”: (string),

“userLogin”: (string)

}

Response:

body: {

“result”: “success”/”failure” (string),

“operationResultMessage”: (string) (debug info)

}

**Search users**

Request: <address:port>/api

body: {

“requestType”: “searchUsers” (string),

“cookie”: (string),

“searchString”: (string),

“roles”: [](string) (should be empty if no filters)

}

Response:

body: {

“result”: “success”/”failure” (string),

“operationResultMessage”: (string) (debug info),

“data”: [{ “login”: (string), “username” (string), “role”: (string) }, <...>]

}

**Get roles/categories**

Request: <address:port>/api

body: {

“requestType”: “getRoles”/”getCategories” (string),

“cookie”: (string)

}

Response:

body: {

“result”: “success”/”failure” (string),

“operationResultMessage”: (string) (debug info),

“data”: [](string)

}

**Search articles**

Request: <address:port>/api

body: {

“requestType”: “searchArticles” (string),

“cookie”: (string),

“searchString”: (string),

“categories”: [](string) (should be empty if no filters)

}

Response:

body: {

“result”: “success”/”failure” (string)

“operationResultMessage”: (string) (debug info)

“data”: [{

“ID”: (string),

“category”: (string),

“type” (string),

“name”: (string),

“born”: (string),

“died”: (string),

“nationality”: (string),

“knownFor”: (string),

“notableWork”: (string),

“about”: (string),

“year”: (string),

“medium”: (string),

“dimensions”: (string),

“location”: (string),

“designedByDeveloper”: (string) }, <...>]

}

If any of fields in article shouldn’t contain data, they will have value set to string with word “None” instead.

**Get article**

Request: <address:port>/api

body: {

“requestType”: “getArticle” (string),

“cookie”: (string),

“articleID”: (string)

}

Response:

body: {

“result”: “success”/”failure” (string),

“operationResultMessage”: (string) (debug info),

“data”: {

“ID”: (string),

“category”: (string),

“type” (string),

“name”: (string),

“born”: (string),

“died”: (string),

“nationality”: (string),

“knownFor”: (string),

“notableWork”: (string),

“about”: (string),

“year”: (string),

“medium”: (string),

“dimensions”: (string),

“location”: (string),

“designedByDeveloper”: (string) }

}

If any of fields in article shouldn’t contain data, they will have value set to string with word “None” instead.

**Edit/Add article**

Request: <address:port>/api

body: {

“requestType”: “editArticle”/”addArticle” (string),

“cookie”: (string),

“articleID”: (string) (“None” if adding new),

“data”: {

“category”: (string),

“type” (string),

“name”: (string),

“born”: (string),

“died”: (string),

“nationality”: (string),

“knownFor”: (string),

“notableWork”: (string),

“about”: (string),

“year”: (string),

“medium”: (string),

“dimensions”: (string),

“location”: (string),

“designedByDeveloper”: (string) }

}

If any of fields in article shouldn’t contain data, they will have value set to string with word “None” instead.

Response:

body: {

“result”: “success”/”failure” (string)

“operationResultMessage”: (string) (debug info)

}

**Delete article**

Request: <address:port>/api

body: {

“requestType”: “deleteArticle” (string),

“cookie”: (string),

“article”: (string)

}

Response:

body: {

“result”: “success”/”failure” (string),

“operationResultMessage”: (string) (debug info)

}

**Test connection**

Request: <address:port>/api

body: {

“requestType”: “test” (string),

}

Response:

body: {

“result”: “success” (string)

}

**Check if cookie is still valid**

Request: <address:port>/api

body: {

“requestType”: “cookieTest” (string),

“cookie”: (string)

}

Response:

body: {

“result”: “success”/”failure” (string)

}