Credentials Api Documentation

Toci

August 21, 2015

Contents

1 Introduction

2 Save data for resource operation.

3 Retrieve data for resource operation.

4 Retrieve data list for resource operation.

5 Delete data for resource operation.

6 Update data for resource operation.

**1 Indtroduction**

Resource: /models/Project models/Projects

Resource destination: This resource allows to Load/Save/Update and Delete projects in databae, with required scope. Project data are stored encrypted in the database, so only it can be only

read having the right credential (scope).

Error Handling: The error handling is unified in the entire API, and the error list is created bitwise.

We use following error codes:

0 - success

1 - IdMissing

2 - WrongId

4 - DataMissing

8 - PasswordMissing

16 - NameMissing

32 - WrongData

Error message example:

Cannot save the project., Project name exist.

**2. Save data for resource operation.**

Resource: /Save/Project

Resource destination: This resource allows to Save project in

database, encrypting the project data.

Example usage:

POST /Save/Project

Request body:

projectName - string, is a name of project

projectAuthor - string, is a name of author

projectData - string, is a data, which will be encrypted

scopeName - string, is a name of scope, which will allow to delete, update, load this project

Example JSON request structure:

{"projectName":"Name",

"projectAuthor":"Me",

"projectData":"Data",

"scopeName":"scope"}

Response formats: JSON

Example JSON Response structure:

{"code":"0",

"message":"Saved!"}

Response structure description:

code - the error code specified in the "Error Handling" section

message - response message from api

Remark:

All body parameters required.

**3. Retrieve data for resource operation.**

Resource: /Load/Project

Resource destination: This resource allows to load project from database, decrypting the project data, for projectName and required scope.

Example usage:

POST /Load/Project

Request body:

projectName - string, is a name of project

scopesList - string, list of strings separated by decimal

Example JSON request structure:

{"projectName":"Name",

"scopesList":"scope, scope2"}

Response formats: JSON

Example JSON Response structure:

{"code":"0",

"message":"Loaded!",

"data":{"Result":

{"scopeName":"scope",

"projectId":"8",

"projectName":"Name",

"projectData":"Data",

"projectAuthor":"Me",

"modificationdate":"21.08.2015 09:59:43"}}}

Response message description:

code - the error code specified in the "Error Handling" section

message - response message from api

data/result - the requested content

Remark:

All body parameters required.

**4. Retrieve data list for resource operation.**

Resource: /Load/Projects

Resource destination: This resource allows to load all projects from

database, decrypting the project data for required scopes.

Example usage:

POST /Load/Projects

Request body:

scopesList - string, list of strings separated by decimal

Example JSON request structure:

{"scopesList":"scope, scope2"}

Response formats: JSON

Example JSON Response structure:

{"code":"0",

"message":"Loaded!",

"data":{"Result":

[{"scopeName":"scope",

"projectId":"8",

"projectName":"Name",

"projectData":"Data",

"projectAuthor":"Me",

"modificationdate":"21.08.2015 09:59:43"},

{"scopeName":"scope2",

"projectId":"10",

"projectName":"2Name",

"projectData":"Data2",

"projectAuthor":"Me2",

"modificationdate":"21.08.2015 12:34:53"}]}}

Response message description:

code - the error code specified in the "Error Handling" section

message - response message from api

data/result - the requested content

Remark:

All body parameters required.

**5. Delete data for resource operation.**

Resource: /Delete/Project

Resource destination: This resource allows to delete project from

database, for projectName and required scope.

Example usage:

POST /Delete/Project

Request body:

projectName - string, is a name of project

scopesList - string, list of strings separated by decimal

Example JSON request structure:

{"projectName":"Name",

"scopesList":"scope, scope2"}

Response formats: JSON

Example JSON Response structure:

{"code":"0",

"message":"Deleted!"}

Response message description:

code - the error code specified in the "Error Handling" section

message - response message from api

Remark:

All body parameters required.

**6 Update data for resource operation.**

Resource: /Update/Project

Resource destination: This resource allows to update project from

database, for required projectId and scope.

Example usage:

POST /Delete/Project

Request body:

projectId - int, project Id

projectName - string, is a name of project

scopesList - string, list of strings separated by decimal

projectData - string, is a data, which will be encrypted

projectAuthor - string, is a name of author

scopeName -new scope name

Example JSON request structure:

{"projectId":"10",

"scopesList":"scope, scope2",

"projectData":"Data2"}

Response formats: JSON

Example JSON Response structure:

{"code":"0",

"message":"Updated!"}

Response message description:

code - the error code specified in the "Error Handling" section

message - response message from api

Remarks:

projectId and scopesList are required for request body parameters