

# Documentation bloc Mahara iENA

Documentation du bloc connecteur de groupe MOODLE MAHARA du projet iENA.

Ce document explique comment paramétrer le plugin MOODLE permettant de connecter un cours MOODLE à un groupe MAHARA.

Ce bloc permet à l'enseignant et aux étudiants de se rendre en un clic d'un espace de cours MOODLE à un espace de groupe sur MAHARA. Il ne nécessite pas de paramétrage pour l'enseignant.

Il exploite pour ce faire l'API de MAHARA pour créer le groupe, inscrire les utilisateurs sur MAHARA au besoin et les inscrire dans le groupe lié au cours. C'est l'objet de cette documentation.

Cependant, notez bien qu'il est nécessaire que les deux plateformes soient liées au préalable pour créer un compte utilisateur depuis MOODLE sur MAHARA au besoin et connecter l'utilisateur automatiquement sur MAHARA.

*Ce plugin a été développé par la Société Softia pour l'Université de Lorraine dans le cadre du projet AMI 2016 soutenu par la MIPNES.*

**Documentation valable pour la version 2019021901 du plugin.** D'après la documentation initiale de Softia.



UNIVERSITÉ  
DE LORRAINE



## Table des matières

<b>1</b>	<b>PREREQUIS SUR MAHARA .....</b>	<b>2</b>
1.1	CREATION D'UN UTILISATEUR POUR GENERER UN TOKEN .....	2
1.2	DONNER LES DROITS.....	3
1.3	ACTIVER L'API .....	4
1.4	CREATION DU SERVICE .....	5
1.5	ATTRIBUTION DES FONCTIONS AU WEBSERVICE .....	6
1.6	GENERER UN TOKEN MANUELLEMENT POUR UN UTILISATEUR : .....	7
1.7	TESTER L'API .....	8
<b>2</b>	<b>UTILISATION DU PLUGIN SUR MOODLE .....</b>	<b>8</b>
2.1	AUTRES INFORMATIONS .....	8

# 1 PREREQUIS SUR MAHARA

Il est indispensable que les deux plateformes soient liées à l'avance dans l'administration des serveurs du réseau MOODLE.

## 1.1 CREATION D'UN UTILISATEUR POUR GENERER UN TOKEN

Il faut créer un utilisateur sur l'institution de votre choix. Attention l'institution va déterminer les groupes et utilisateurs que vous pouvez "voir".

Dans notre exemple, nous avons au préalable créé une institution "MaharaMoodle" afin de mettre en place mahoodle.

**1 - User creation method**

☒ Create new user from scratch

First name \*  
apimoodle

Last name \*  
apimoodle

Email \*  
api@gmail.com

OR

☐ Upload Leap2A file [Choisir un fichier](#) Aucun fichier choisi

**2 - Basic details**

Username \*  
apimoodle

Password \*  
.....

Site staff ☒ No

Site administrator ☒ No

Institution \*  
MaharaMoodle: internal

File quota: (Megabytes)  
50

Institution administrator ☒ No

**3 - Create**

General account options

[Create user](#)

They will receive an email informing them of their new account details. On first login, they will be forced to change their password.

Choix de l'institution

## 1.2 DONNER LES DROITS

Une fois notre utilisateur créé, il faut lui donner les droits administrateurs sur son institution. Pour ce faire, il rendez-vous sur la page "Institution administrators".

Il faut ensuite ajouter l'utilisateur "apimoodle" dans le groupe "Current administrators".

The screenshot shows the Mahara Administration interface. At the top, there's a header with the Mahara logo and 'Administration' text. A search bar labeled 'Search users' is present. Below the header, the page title is 'Institution administrators' with a building icon, and the institution name 'MaharaMoodle' is displayed on the right. A descriptive text states: 'Here you can choose which users are administrators for the institution. The current administrators are listed in the "Current administrators" field, and potential administrators are shown in the "Institution members" field.'

The main content area contains two lists of users:

- Admin users:** A search bar and a 'Search' button.
- Institution members:** A list of users including 'Admin Utilisateur (admin1)', 'tool generator (tool)', and 'apimoodle apimoodle (apimoodle)'. The 'apimoodle' user is highlighted in blue.
- Current administrators:** A list of users including 'mahoodle! Camille (mahoodle!)'. The title 'Current administrators' is highlighted in blue.

Arrows indicate the workflow: one arrow points from the highlighted 'apimoodle' user in the 'Institution members' list to the right-pointing arrow button between the lists; another arrow points from the 'Submit' button at the bottom to the left-pointing arrow button between the lists.

### 1.3 ACTIVER L'API

Pour activer l'api, il faut se rendre dans la page : "Web services configuration" (URL : /webservice/admin/index.php). Puis, cochez les cases comme l'écran ci-dessous :

**mahara** Administration

Search users

## Web services configuration

Here you can set up the varying web services rules and enable or disable them.

**Enable web service functionality**

**Web service requester master switch**

Allow outgoing web service requests: ☒ Yes **Web services enabled**

**Web service provider master switch**

Accept incoming web service requests: ☒ Yes **Web services enabled**

Enable or disable protocols supported as a web services provider:

Protocol

SOAP:	<input checked="" type="checkbox"/> Yes
XML-RPC:	<input checked="" type="checkbox"/> Yes
REST:	<input checked="" type="checkbox"/> Yes
OAuth:	<input checked="" type="checkbox"/> Yes

Networking certificates

Manage service groups

Manage service access tokens

Manage service users

## 1.4 CREATION DU SERVICE

Il faut créer un service. Un service est un groupe de fonctions que l'on peut utiliser avec notre API. Saisissez le nom du service en bas et validez.

Manage service groups

Build lists of functions into service groups that can be allocated to users authorised for execution.

SERVICE	COMPONENT	ENABLED	AUTHORISED USERS ONLY	USER TOKEN ACCESS	FUNCTIONS	
moodletest (moodletestshort)	(Custom)	✓	✗	✓	mahara_autologin_redirect, mahara_group_create_groups, mahara_group_delete_groups, mahara_group_get_groups, mahara_group_get_groups_by_id, mahara_group_update_group_members, mahara_group_update_groups, mahara_institution_add_members, mahara_institution_decline_members, mahara_institution_get_members, mahara_institution_get_requests, mahara_institution_invite_members, mahara_institution_remove_members, mahara_submission_get_views_for_user, mahara_submission_release_submitted_view, mahara_submission_submit_view_for_assessment, mahara_user_create_users, mahara_user_delete_users, mahara_user_get_all_favourites, mahara_user_get_context, mahara_user_get_extended_context, mahara_user_get_favourites, mahara_user_get_my_user, mahara_user_get_online_users, mahara_user_get_users, mahara_user_get_users_by_id, mahara_user_update_favourites, mahara_user_update_users, module_lti_launch, module_mobileapi_get_blogs, module_mobileapi_get_folders, module_mobileapi_get_notifications, module_mobileapi_get_tags, module_mobileapi_get_user_profile, module_mobileapi_get_user_profileicon, module_mobileapi_sync, module_mobileapi_upload_blog_post, module_mobileapi_upload_file	<a>Edit</a> <a>Delete</a>
softiaCam (webservice)	(Custom)	✓	✓	✗	mahara_user_get_users_by_id, mahara_user_create_users, mahara_user_delete_users, mahara_user_get_all_favourites, mahara_user_get_context, mahara_user_get_extended_context, mahara_user_get_favourites, mahara_user_get_my_user, mahara_user_get_online_users, mahara_user_get_users, mahara_user_update_favourites, mahara_user_update_users	<a>Edit</a> <a>Delete</a>
LTI integration (maharaiti)	module/lti	✓	✗	✗	module_lti_launch	<a>Edit</a>
Mahara Mobile API (maharamobile)	module/mobileapi	✓	✗	✓	module_mobileapi_sync, module_mobileapi_get_blogs, module_mobileapi_get_folders, module_mobileapi_get_notifications, module_mobileapi_get_tags, module_mobileapi_get_user_profile, module_mobileapi_get_user_profileicon, module_mobileapi_upload_file, module_mobileapi_upload_blog_post	<a>Edit</a>

Add

## 1.5 ATTRIBUTION DES FONCTIONS AU WEBSERVICE

Une fois créé, on attribue des fonctions au service.

Il faut avoir la même configuration que sur les captures d'écran ci-dessus.

Service group: moodleapi

Fields marked by '\*' are required.

**Service group: moodleapi**

Name \*

A human-readable name for this service group.

Short name

A machine-readable name for this service group. This is the name that will be used if an external service needs to refer to this service group.

**Enable or disable the service**

Service ☒ Yes

Authorised users only ☐ No

User token access ☒ Yes

Ensuite les fonctions à activer à minima sont les suivantes :

- mahara\_group\_create\_groups
- mahara\_group\_get\_groups
- mahara\_group\_delete\_groups
- mahara\_group\_update\_group\_members
- mahara\_user\_get\_my\_user
- mahara\_user\_get\_users
- mahara\_user\_get\_users\_by\_id

Résultat final :

Manage service groups					
Build lists of functions into service groups that can be allocated to users authorised for execution.					
SERVICE	COMPONENT	ENABLED	AUTHORISED USERS ONLY	USER TOKEN ACCESS	FUNCTIONS
moodleapi (moodleapi)	(Custom)	✓	✗	✓	mahara_group_create_groups, mahara_group_delete_groups, mahara_group_get_groups, mahara_group_get_groups_by_id, mahara_group_update_group_members, mahara_group_update_groups, mahara_user_get_my_user, mahara_user_get_users, mahara_user_get_users_by_id

## 1.6 GENERER UN TOKEN MANUELLEMENT POUR UN UTILISATEUR :

Dans cette étape, on génère le token pour l'utilisateur et le webservice précédemment créés.

Il faut se rendre dans l'onglet "Manage Service access tokens" URL (</webservice/admin/index.php>).



Tapez le *username* de l'utilisateur et cliquez sur "Generate token".

Username:

**Generate token**

On a maintenant le token pour l'utilisateur, il faut maintenant choisir le bon type de service que nous avons créé précédemment, dans notre cas "**moodleapi**" et sauvegarder.

### Service access tokens

**Token "285831e9aa14223bfbd7057d9897d07e"**

Institution: MaharaMoodle

Username: apimoodle

Service: moodleapi

Enabled: Yes

Authorised users only: No

Functions: mahara\_group\_create\_groups, mahara\_group\_delete\_groups, mahara\_group\_get\_groups, mahara\_group\_get\_groups\_by\_id, mahara\_group\_update\_group\_members, mahara\_group\_update\_groups, mahara\_user\_get\_my\_user, mahara\_user\_get\_users, mahara\_user\_get\_users\_by\_id

Enable web services security (XML-RPC Only): No

## 1.7 TESTER L'API

Afin de tester l'api vous pouvez utiliser cette requête CURL

```
curl -U nomUtilisateur -H 'Host: ADRESSE.DE.MAHARA'
'https://ADRESSE.DE.MAHARA/webbservice/rest/server.php?alt=json&wsfunction=
mahara_user_get_my_user&wstoken=LE_TOKEN_UTILISATEUR_GENERE'
```

## 2 UTILISATION DU PLUGIN SUR MOODLE

Lors de l'installation du plugin dans Moodle vous devez modifier les paramètres. Attention il faut taper en minuscule le nom de votre institution.

Moodle Softia

[Tableau de bord](#) / [Administration du site](#) / [Plugins](#) / [Blocs](#) / [Mahara iena Block](#)

### Mahara iena Block

Base URL mahara <small>base_mahara</small>	<input type="text" value="http://192.168.10.65/"/>	Défaut : http://192.168.10.65/
Token <small>wstoken</small>	<input type="text" value="dccba94369f67fe577d7d0ed911535a1"/>	Défaut : 4b166d2e71428228cb610d50c442c0d6
Institution <small>institution_mahara</small>	<input type="text" value="maharamoodle"/>	Défaut : maharamoodle

[Enregistrer les modifications](#)

Une fois le bloc ajouté au cours vous avez trois boutons :

- Créer un groupe dans MAHARA : celui-ci permet de créer le groupe dans MAHARA
- Créer plusieurs groupes dans MAHARA: celui-ci permet de créer un groupe dans MAHARA pour chaque groupe MOODLE dans le cours
- Accéder au groupe MAHARA: celui-ci est "grisé" tant qu'aucun groupe MAHARA n'est lié au cours

Pour créer un groupe dans MAHARA et le lier au cours, il suffit de cliquer sur l'un des deux premiers boutons. Attention, le choix de créer un groupe MAHARA pour tout le cours MOODLE ou un groupe MAHARA pour chaque groupe d'utilisateur dans le cours MOODLE est définitif.

Maintenant le bouton "Accéder au groupe MAHARA" est accessible.

### 2.1 AUTRES INFORMATIONS

Vous pouvez définir le nom de la plateforme MAHARA dans l'administration. Par exemple, si vous la renommez « ePortfolio » l'utilisateur verra s'afficher « Aller à mon groupe sur *ePortfolio* » au lieu de « Aller à mon groupe sur *MAHARA* ».