

# eThink



## Creating Your First Web Service Integration From Core APIs



# The REST-like Moodle WebServices API

- REST-like not RESTful
  - All calls are POST
  - Return 200 HTTP statuses, even on failure
  - Functions are parameters instead of URIs
- Core WebService calls are highly focused on Mobile App functionality

# Getting started

- Define the integration
  - User Management
  - Courses
  - Enrollments
  - Completion??

# Investigate


- API Documentation
  - Site administration -> Plugins -> Web services -> API Documentation

`core_user_add_user_device` ▶  
`core_user_add_user_private_files` ▶  
`core_user_agree_site_policy` ▶  
`core_user_create_users` ▶  
`core_user_delete_users` ▶  
`core_user_get_course_user_profiles` ▶  
`core_user_get_private_files_info` ▶  
`core_user_get_user_preferences` ▶  
`core_user_get_users` ▶  
`core_user_get_users_by_field` ▶  
`core_user_remove_user_device` ▶  
`core_user_set_user_preferences` ▶  
`core_user_update_picture` ▶  
`core_user_update_user_preferences` ▶  
`core_user_update_users` ▶  
`core_user_view_user_list` ▶  
`core_user_view_user_profile` ▶

# Create


- External Services
  - Site administration -> Plugins -> Web services -> External services



External service



Name  00\_MyWebService



Short name 00\_MyWebService


Enabled

Authorized users only 

Can download files  

Can upload files  

Required capability   **No required capability**

Search 

[Show less...](#)

**Add service** Cancel

# Add Functions

- Functions
  - Site administration -> Plugins -> Web services -> External services -> Functions

00\_MyWebService

## ▼ Add functions

Name



× core\_user\_create\_users:Create users.


× core\_user\_get\_users:search for users matching the parameters

× core\_user\_update\_users:Update users.

core\_user\_create ▼

Add functions

Cancel

There are required fields in this form marked  .

# Create Your WebService User

- Add a new user
  - Site administration -> Users -> Accounts -> Add a new user

## General

Username



wsuser

Choose an authentication  
method



Web services authentication

Suspended account

Generate password and notify user

The password must have at least 8 characters, at least 1 digit(s), at least 1 lower case letter(s), at least 1 upper case letter(s), at least 1 non-alphanumeric character(s) such as \*, -, or #

New password



[Click to enter text](#)

Force password change

First name



00\_MyWebService

Last name



User

Email address



jesse.safran@ethinkeducation.com

# Add Your User to Your WebService

- Authorized users
  - Site administration -> Plugins -> Web services -> External services -> Authorized users

**Authorized users (1)**  
00\_MyWebService User

**Too many users (134) to show**  
Please use the search

◀ Add

▶ Remove

Search  Clear

Search options ▶

Search  Clear

Change settings for the authorized users

[00\\_MyWebService User, jesse.safran2@ethinkeducation.com](#)

Missing capabilities: moodle/user:create, moodle/user:viewdetails, moodle/user:viewhiddendetails, moodle/course:useremail, moodle/user:update



# Create a Token

- Create Token
    - Site administration -> Plugins -> Web services -> Manage tokens
- Create token

## Token

User ! **00\_MyWebService User**

Search

Service ! **00\_MyWebService**

IP restriction

Valid until     Enable

## Manage tokens

Any token can be deleted, though you can only view tokens that you created.

Token	First name <sup>▲</sup> / Last name	Service
11114db6e24ec54598d18686ff103485	00_MyWebService User	00_MyWebService

# Test your work

- Using Postman

The screenshot shows a Postman REST client interface. At the top, the request method is set to POST and the URL is `https://academy.ethinksites.com/webservice/rest/server.php?wsfunction=core_user_get_users`. The 'Body' tab is selected, and the request body is configured as 'form-data'. The body contains four key-value pairs, all checked:

KEY	VALUE	DESCRIPTION
wstoken	11114db6e24ec54598d18686ff103485	
moodlewsrestformat	json	
criteria[0][key]	username	
criteria[0][value]	jesse.safran	

Below the body configuration, the response is displayed in the 'Body' tab. The status is 200 OK, the time taken is 504 ms, and the size is 904 B. The response is shown in JSON format:

```
1 {
2   "users": [
3     {
4       "id": 88,
5       "username": "jesse.safran",
6       "firstname": "Jesse",
7       "lastname": "Safran",
8       "fullname": "Jesse Safran",
9       "department": "",
10      "firstaccess": 0,
11      "lastaccess": 0,
```

# Expanding your work

- Using The Demo Mobile App + Google Chrome Network Inspector
  - <https://mobileapp.moodledemo.net/>

The screenshot displays a mobile application interface on the left and a Chrome Network Inspector on the right. The app's header is orange and contains the text 'JesseEmailTesting'. Below the header, there is a list of participants: 'Amy Call' (last access: 11 days ago), 'Vanessa Royce', 'Jesse Safran', and 'JesseSafran TestUser' (last access: a few seconds ago). The 'Participants' link is circled in red. The Network Inspector shows a list of network requests. The selected request is a REST API call to 'academy.ethinksites.com/webservice/pluginfile.php' with the following details:

- Request Headers (6):**
  - expires: Thu, 01 Jan 1970 00:00:00 GMT
  - pragma: no-cache
  - server: Apache/2.4.39 (Ubuntu)
  - status: 200
- Query String Parameters:**
  - moodlewsrestformat: json
  - wsfunction: core\_enrol\_get\_enrolled\_users
- Form Data:**
  - courseid: 71
  - options[0][name]: limitfrom
  - options[0][value]: 0
  - options[1][name]: limitnumber
  - options[1][value]: 50
  - options[2][name]: sortby
  - options[2][value]: siteorder
  - moodlewssettingfilter: true
  - moodlewssettingfileurl: true
  - wsfunction: core\_enrol\_get\_enrolled\_users

# eThink



[www.ethinkeducation.com](http://www.ethinkeducation.com)

