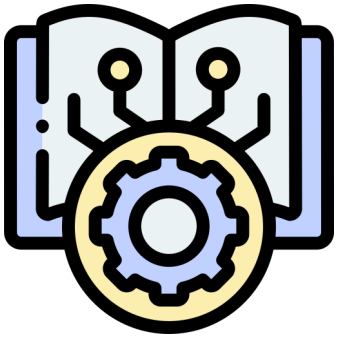
X5-GON Moodle Plugin



X5Moodle

Version 1.3 (2021-02-28)

X5GON-Nantes University

The **X5Moodle** is an easy to install Moodle plugin enabling the integration of different X5-GON project features within any type Moodle LMS.

1. How to read this document

Users primarily interested in installing the X5Moodle should go to the section with that name. Users interested in how the plugin works and what data it is collecting should read the section 3. Users who want to know more about the X5-GON project should consult the website or read section 2.
In order to ensure readability we have coloured in red text corresponding to Moodle terms and in blue those for the new plugin.

2. X5-GON Project

X5-GON is an EU H2020 project aiming to create a solution which will help users/students to find what they need not just in OER[1] repositories, but across all open educational resources on the web.
This solution will adapt to the user's needs and learn how to make ongoing customized recommendations and suggestions through a truly interactive and impactful learning experience.
This new AI-driven platform will deliver OER content from everywhere, tailored to the students' needs at the right time and place. To do so it will use the following solutions to accomplish this goal:

- **Aggregation**: It will gather relevant content in one place, from the project case studies as well as from external providers and other preferred resources.
- **Curation**: AI and machine learning will be key to curate relevant and contextual content and external students at the right time and point of need.
- **Personalization**: It will make increasingly personalized recommendations for learning content to suit students' needs, based on the analysis of relevant factors.
- **Creation**: Large, small and medium-sized universities have tacit knowledge that can be unlocked and re-used. This approach will allow any organization to release and build their own content libraries quickly and conveniently to share with the world and vice versa.

3. Functionalities

The plugin ensures three main functionalities:

3.1 X5Moodle, a new Moodle activity type

The X5-Moodle is a Moodle activity plugin implemented based on 2 key ideas: 1) providing AI-based smart tools for delivering open resources as a support during the course session, and 2) evolving the tools through the usage-based approaches rather than focusing on the user data. The aim of this new type of activity is to provide the teachers and students the best learning experience. Using the OER collection and the AI tools developed within X5GON, a new learning activity is designed following the Moodle specification and workflow, as well as the new-fashioned spirit of using AI to support learning. The X5-Moodle uses:

- The X5GON Search Engine to find more OERs,
- The X5-Recommend item-based system available via the Learning Analytics Machine API to provide recommended OERs based on the resources usage history.

The plugin can be used through 2 different perspectives: the teacher and the students.

From the teachers point of view, the plugin is mainly a configuration interface in which they can choose between 3 functionalities that can be seen by the students: The X5GON Search Engine, the X5-Recommend, and the X5-Playlist provided by the X5Learn (see Section 10). The teacher sets up the required configuration of the specific functionality, such as the Search Engine initialization, playlist upload, and other. Figure 1 shows the Moodle interface where the teacher configures the plugin functionalities.

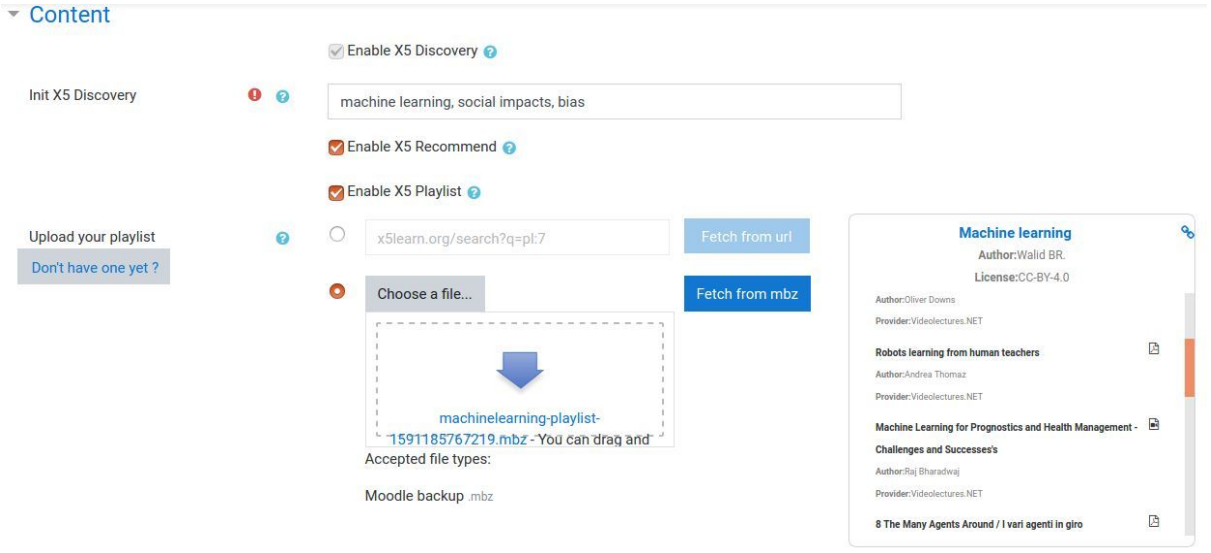


Fig. 1: TheMoodle plugin interface allowing a teacher to choose the X5GON activity, e.g. X5GON Search engine, the X5-Recommend and the X5-Playlist

Once the plugin is instantiated inside the Moodle course, the students can see the plugin functionalities configured by the teacher. The X5GON Search Engine allows the students to search for new OER resources. In addition, they are able to see a list of the most frequent

search queries performed by the course participants. Figure 2 shows an example of the configured X5GON Search Engine view in the Moodle.

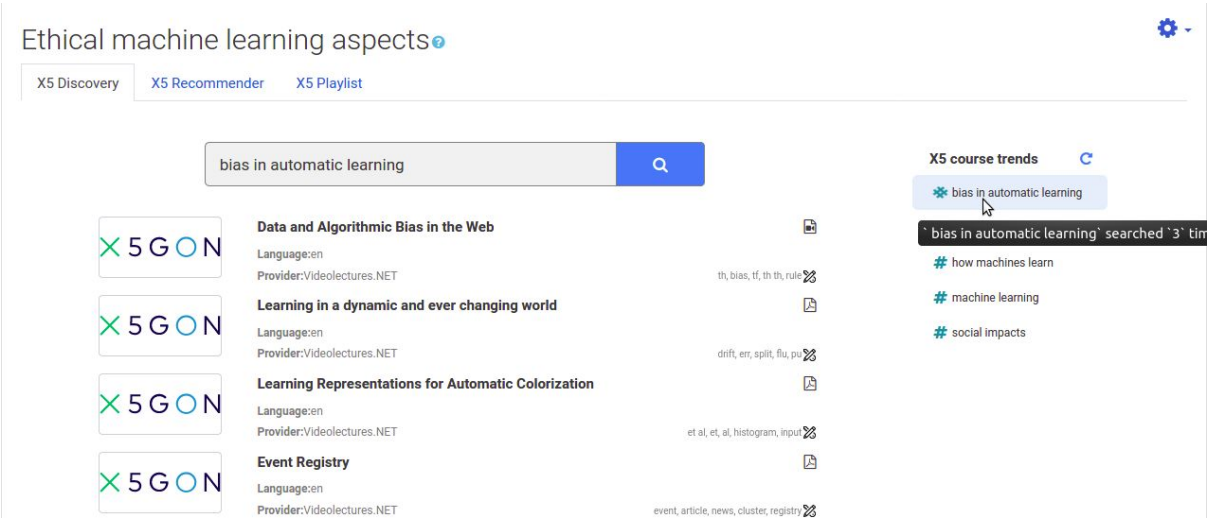


Fig. 2: The X5GON Search Engine functionality in the X5-Moodle. With it, a student can search through OERs as well as see the most popular search queries.

The X5-Playlist functionality allows the teacher to embed a playlist that was created through the X5Learn service. The published playlists can be embedded into the Moodle course via the plugin, making it available to the students. Figure 11shows the embedded playlist in the Moodle course.

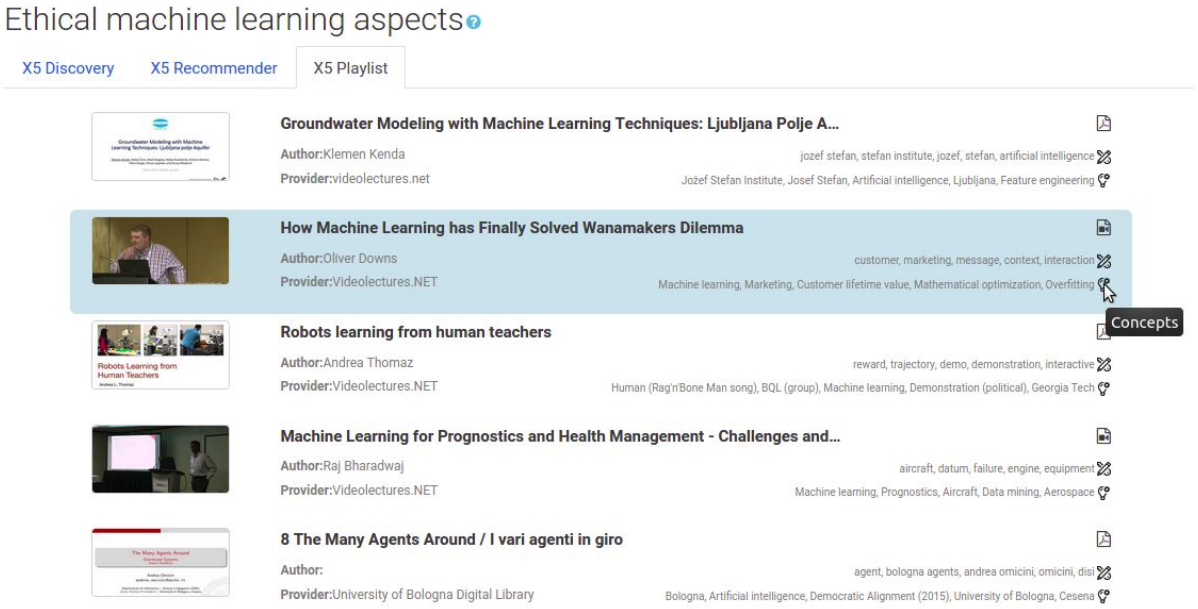


Fig. 3: The X5-Playlist functionality embedded in the Moodle course. The example shows the teachers playlist created using the X5Learn service.

The X5-Recommend functionality provides OER recommendations based on the students current work. It also provides viewing statistics, e.g. how many times an OER was viewed by the students' fellows, as well as additional OER metadata. Figure 12shows an example of the OER recommendation list.

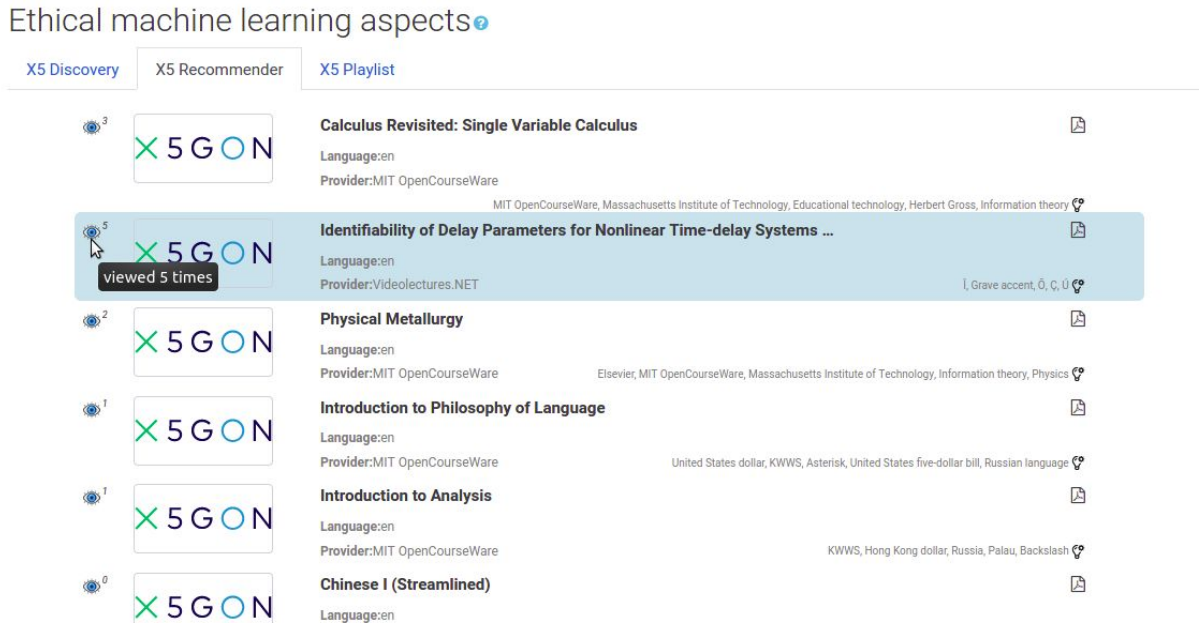


Fig. 4: An example of the integrated X5-Recommend list.It shows the recommended OER resources as well as the viewing statistics of the student and its fellows.

With the objective of improving the learning experience of both the teacher and students, the X5-Moodle is designed to offer several features that hopefully will lead us to achieving our objective. These features include:

- OER based search engine through the rich X5GON OER network.
- Usage-focused functionalities, instead of tracking the users.
- X5-Trends, the most popular search queries based on the viewers' history.
- Extra information about the OERs, including the history of views, the OER metadata, and the enriched information such as keywords, concepts, difficulty score, and other.
- X5-Recommend. Built as a linear combination of the recommendations from the item-based X5GON recommendation system and the popularity of the viewed OERs in the Moodle course. The recommendations are systematically evolving (with popular and new resources) with the students' exploration as well as the Moodle class shifting their attention to new OERs.

3.2 X5-GON connect-service integration

The Second role of the plugin is to make easy the control of the ***integration of X5-GON JS library*** by managing the call and the activation (execution) of the connect-service responsible for notifying X5-GON server about an OER access dynamically in Moodle suitable pages.

3.1.1. OER Criteria

The connect-service is activated on the courses pages and their sub-pages[2] and notifies X5-GON about any access taking place on one of its pages (for more details about what information is sent by the connect-service, please refer to the X5-GON connect-service documentation). The activation will take place when the following conditions are met:

- ***Course category***: attribute course.category should be equal to "OER" or another administrator defined value.
- ***Course visibility***[3]: attribute course.visibility should be equal to "1" or another administrator defined value.
- ***Course enrolment type***[4]: course should have at least an enrol type equal to "guest" + "attribute enrol.password should be empty", or another administrator defined value.

3.1.2. The X5GON privacy policy

The user privacy was an ultimate objective since the design and implementation moment of X5Moodle. That's why once installed, an X5GON privacy policy is installed as well in the Moodle installation (especially with the versions having a site policy handler technology). That way, when the setting of Policies(tool_policy) is enabled in Moodle, no data will be sent before the user agrees on the X5GON privacy policy which is aiming to allow sending some OER meta-data to X5GON in order to improve the AI algorithms, with the total respect of the GDPR rules.

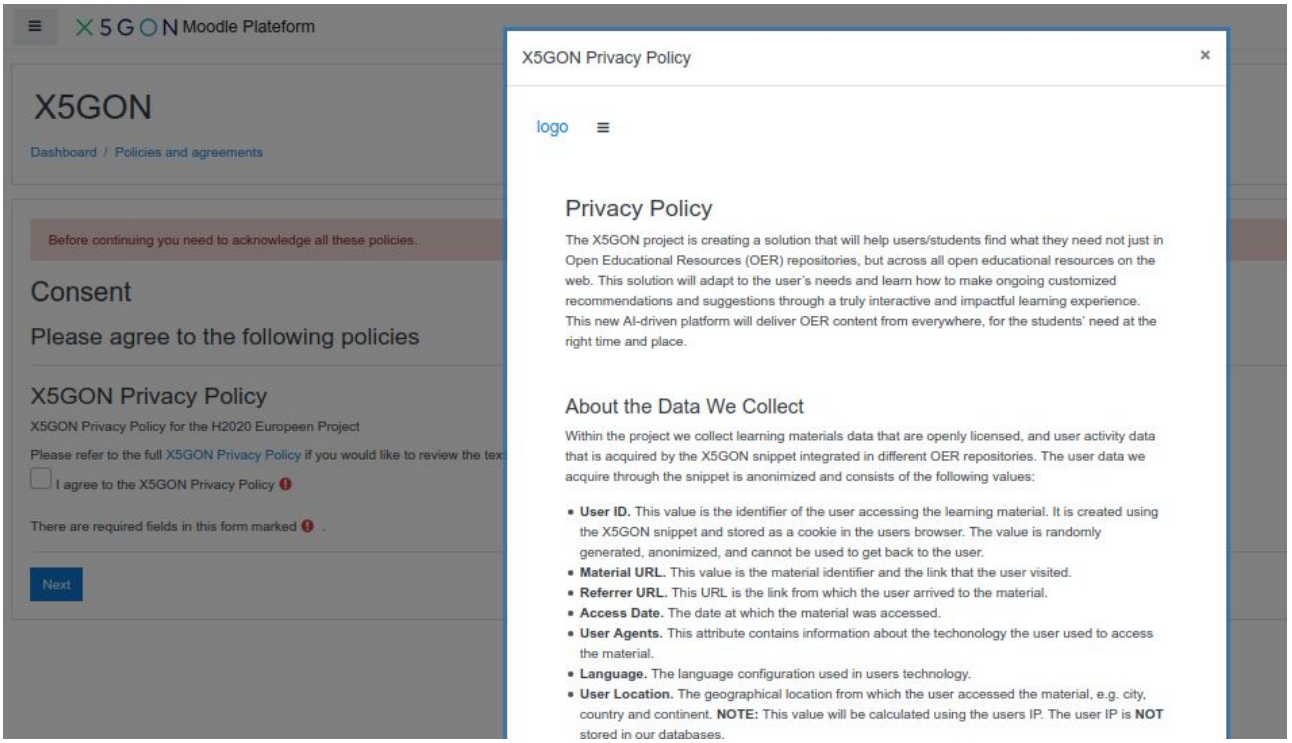


Fig. 5: The X5GON privacy policy shown to a new Moodle user.

3.3 Moodle Data API webservice

The second role of the plugin is to enable the ***communication between Moodle and external parties*** (in our case the X5-GON server) using a webservice, while respecting all security precautions : protocol, tokens, restrictions on services and permissions...
This part of the plugin will be complementary to the coonect-service work. Moodle is a complete and well architectured LMS: all urls shown in its pages referencing the different resources/courses are coded with internal IDs. And since the meta-data from the resources are needed by X5-GON, this plugin will be in charge of answering requests received from the X5-GON server to fetch meta-data about an OER or an OER course.

3.2.1. OER Criteria

At this initial step, the data API web-service will answer the requests concerning resources or courses if the following conditions are met:

- ***Course category***: attribute course.category should be equal to "OER" or another administrator defined value.
- ***Course visibility***[3]: attribute course.visibility should be equal to "1" or another administrator defined value.
- ***Course enrolment type***[4]: course should have at least an enrol type equal to "guest" + attribute enrol.password should be empty, or another administrator defined value.

- ***Course module visibility***[5]: attribute module.visibility should be equal to "1" or another administrator defined value.
- ***Course module availability***[6]: attribute module.availability should be empty or another administrator defined value.
- ***File license***[7]: attribute file.license should be any one of the Creative Commons (CC) licenses.

3.2.2. How to use the Data API

The Data API is a webservice. It is in charge of answering to the external requests demanding specific services passed as parameters. The requested services should be one of the followings:

- *oerinfos*: Returns the meta-data of a specific Moodle module if it fits the OER criteria. For modules of type **Resource**, the response will also contain all information related to their attached final files (media, files, ...).
  Below is an example of a response about a Moodle module with ID=22:

```
stdClass Object
(
    [id] => 22
    [course] => 2
    [added] => 1527272984
    [modname] => resource
    [moduleurl] => http://localhost:8080/moodle/mod/resource/view.php?id=22
    [course_module_instance] => stdClass Object
        (
            [name] => follio
            [timemodified] => 1529091130
        )

    [course_module_file] => stdClass Object
        (
            [filename] => courseofme.mp4
            [filesize] => 91989114
            [mimetype] => video/mp4
            [author] => Admin User
            [license] => cc
            [timecreated] => 1527272974
            [timemodified] => 1529091128
            [fileurl] => http://localhost:8080/moodle/pluginfile.php/44/mod_resource/content/0/courseofme.mp4
        )

    [course_module_file_licence] => stdClass Object
        (
            [shortname] => cc
            [fullname] => Creative Commons
            [source] => http://creativecommons.org/licenses/by/3.0/
            [version] => 2010033100
        )

    [course_gen_infos] => stdClass Object
        (
            [id] => 2
            [fullname] => Machine Learning For Beginners
            [shortname] => Machine Learning
            [summary] => <p>Initial resources for Machine Learning.<br></p>
            [lang] => en
            [timecreated] => 1525089382
            [timemodified] => 1542208982
            [courseurl] => http://localhost:8080/moodle/course/view.php?id=2
        )

    [course_cat_infos] => stdClass Object
        (
            [id] => 2
            [name] => OER
            [description] => <p>Test Category for OERs.</p>
            [parent] => 0
            [coursecount] => 1
        )

)
```

Fig. 6: Example of *oerinfos* service.

- *courseoers*: Returns specific course meta-data and related OER Moodle modules.
  Below is an example of a response about a course with ID=2:

```
stdClass Object
(
    [course_gen_infos] => stdClass Object
        (
            [id] => 2
            [fullname] => Machine Learning For Beginners
            [shortname] => Machine Learning
            [summary] => <p>Initial resources for Machine Learning.<br></p>
            [lang] => en
            [timecreated] => 1525089382
            [timemodified] => 1542208982
            [courseurl] => http://localhost:8080/moodle/course/view.php?id=2
        )

    [course_cat_infos] => stdClass Object
        (
            [id] => 2
            [name] => OER
            [description] => <p>Test Category for OERs.</p>
            [parent] => 0
            [coursecount] => 1
        )

    [course_modules] => stdClass Object
        (

            [6] => stdClass Object
                (
                    [id] => 6
                    [course] => 2
                    [added] => 1525447021
                    [modname] => resource
                    [moduleurl] => http://localhost:8080/moodle/mod/resource/view.php?id=6
                    [course_module_instance] => stdClass Object
                        (
                            [name] => Part5
                            [timemodified] => 1525447021
                        )

                    [course_module_file] => stdClass Object
                        (
                            [filename] => X5GON Project Management Board Meeting - Minutes (1).docx
                            [filesize] => 20118
                            [mimetype] => application/vnd.openxmlformats-officedocument.wordprocessingml.document
                            [author] => Admin User
                            [license] => cc
                            [timecreated] => 1525447016
                            [timemodified] => 1525447021
                            [fileurl] => http://localhost:8080/moodle/pluginfile.php/28/mod_resource/content/0/X5GON Project Management Board Meeting - Minutes (1).docx
                        )

                    [course_module_file_licence] => stdClass Object
                        (
                            [shortname] => cc
                            [fullname] => Creative Commons
                            [source] => http://creativecommons.org/licenses/by/3.0/
                            [version] => 2010033100
                        )

                )


            [22] => stdClass Object
                (
                    [id] => 22
                    [course] => 2
                    [added] => 1527272984
                    [modname] => resource
                    [moduleurl] => http://localhost:8080/moodle/mod/resource/view.php?id=22
                    [course_module_instance] => stdClass Object
                        (
                            [name] => follio
                            [timemodified] => 1529091130
                        )

                    [course_module_file] => stdClass Object
                        (
                            [filename] => courseofme.mp4
                            [filesize] => 91989114
                            [mimetype] => video/mp4
                            [author] => Admin User
                            [license] => cc
                            [timecreated] => 1527272974
                            [timemodified] => 1529091128
                            [fileurl] => http://localhost:8080/moodle/pluginfile.php/44/mod_resource/content/0/courseofme.mp4
                        )

                    [course_module_file_licence] => stdClass Object
                        (
                            [shortname] => cc
                            [fullname] => Creative Commons
                            [source] => http://creativecommons.org/licenses/by/3.0/
                            [version] => 2010033100
                        )

                )

            [3] => stdClass Object
                (
                    [id] => 3
                    [course] => 2
                    [added] => 1525092192
                    [modname] => url
                    [moduleurl] => http://localhost:8080/moodle/mod/url/view.php?id=3
                    [course_module_instance] => stdClass Object
                        (
                            [name] => External URL resource
                            [timemodified] => 1542035558
                        )

                    [course_module_file] =>
                    [course_module_file_licence] =>
                )

        )
)
```


Fig. 7: Example of *courseoers* service.

- *oerslist*: Returns the meta-data of all OER Moodle modules which can be found in Moodle classified by course.
  Below is an example of this type of service:


```
stdClass Object
(
    [2] => stdClass Object
        (
            [course_gen_infos] => stdClass Object
                (
                    [id] => 2
                    [fullname] => Machine Learning For Beginners
                    [shortname] => Machine Learning
                    [summary] => <p>Initial resources for Machine Learning.<br></p>
                    [lang] => en
                    [timecreated] => 1525089382
                    [timemodified] => 1542208982
                    [courseurl] => http://localhost:8080/moodle/course/view.php?id=2
                )

            [course_cat_infos] => stdClass Object
                (
                    [id] => 2
                    [name] => OER
                    [description] => <p>Test Category for OERs.</p>
                    [parent] => 0
                    [coursecount] => 1
                )

            [course_modules] => stdClass Object
                (
                    [25] => stdClass Object
                        (
                            [id] => 25
                            [course] => 2
                            [added] => 1529589842
                            [modname] => assign
                            [moduleurl] => http://localhost:8080/moodle/mod/assign/view.php?id=25
                            [course_module_instance] => stdClass Object
                                (
                                    [name] => Assign-Test
                                    [timemodified] => 1538142327
                                )

                            [course_module_file] =>
                            [course_module_file_licence] =>
                        )

                )

        )
    [3] => stdClass Object
        (
            [course_gen_infos] => stdClass Object
                (
                    [id] => 3
                    [fullname] => Deep Learning Level 1
                    [shortname] => DLearningLvl1
                    [summary] => <p>Level one in Deep learning</p>
                    [lang] =>
                    [timecreated] => 1528124285
                    [timemodified] => 1529573738
                    [courseurl] => http://localhost:8080/moodle/course/view.php?id=3
                )

            [course_cat_infos] => stdClass Object
                (
                    [id] => 1
                    [name] => Miscellaneous
                    [description] =>
                    [parent] => 0
                    [coursecount] => 1
                )

            [course_modules] => stdClass Object
                (
                    [23] => stdClass Object
                        (
                            [id] => 23
                            [course] => 3
                            [added] => 1528124285
                            [modname] => forum
                            [moduleurl] => http://localhost:8080/moodle/mod/forum/view.php?id=23
                            [course_module_instance] => stdClass Object
                                (
                                    [name] => Announcements
                                    [timemodified] => 1528124285
                                )

                            [course_module_file] =>
                            [course_module_file_licence] =>
                        )

                    [33] => stdClass Object
                        (
                            [id] => 33
                            [course] => 3
                            [added] => 1542190664
                            [modname] => resource
                            [moduleurl] => http://localhost:8080/moodle/mod/resource/view.php?id=33
                            [course_module_instance] => stdClass Object
                                (
                                    [name] => Topic1-introduction
                                    [timemodified] => 1542190664
                                )

                            [course_module_file] => stdClass Object
                                (
                                    [filename] => transparents_fourier.pdf
                                    [filesize] => 316275
                                    [mimetype] => application/pdf
                                    [author] => Admin User
                                    [license] => cc
                                    [timecreated] => 1542190658
                                    [timemodified] => 1542190664
                                    [fileurl] => http://localhost:8080/moodle/pluginfile.php/58/mod_resource/content/0/transparents_fourier.pdf
                                )

                            [course_module_file_licence] => stdClass Object
                                (
                                    [shortname] => cc
                                    [fullname] => Creative Commons
                                    [source] => http://creativecommons.org/licenses/by/3.0/
                                    [version] => 2010033100
                                )

                        )

                )

        )

)
```


Fig. 8: Example of *oerslist* service.

Remark: All the above services respect the OER criteria explained previously:

- for the course: category, visibility and enrolment type.
- for the module: visibility and availability.
- for the final file (resource): CC license.

Technically and practically, those services are reached using an ***XMLRPC request*** from outside Moodle through specifying:

1. ***MethodName*:** The main service name needed to be used when trying to reach the explained services above.
   MethodName must be specified in the request: **mod_xfgon_oermetadata_requests**
2. ***Params*:** The parameters array which must be passed to call a specific service from the list above. The array must contain two elements like **[element_1, element_2]** where:
   element_1: Service name to be called .
   For example: ***'courseoers'*.**
   element_2: Moodle module.id[8] OR course.id[9] needed to fetch the meta-data.
   For example: ***20147*** (as module.id) OR ***2147*** (as course.id)**.**

4. How the X5Moodle Data API works

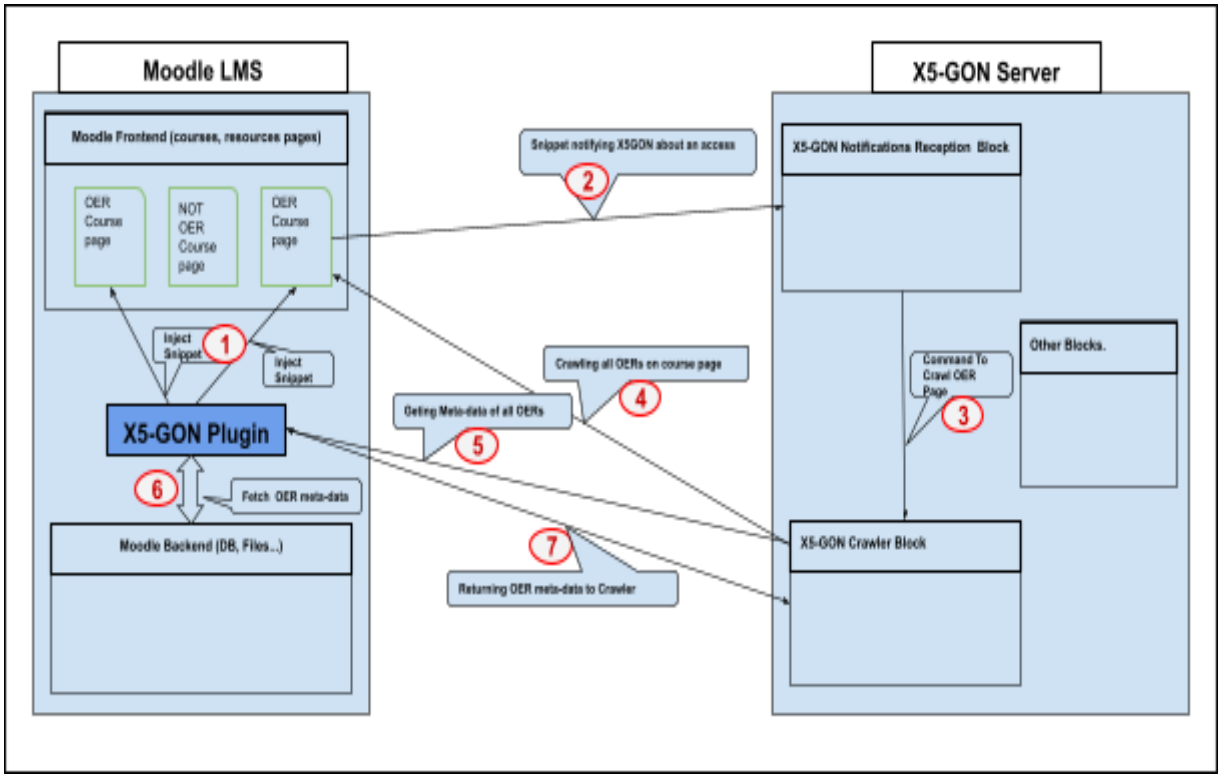


Fig. 9: X5moodle, the Data API - How it's working.

XMoodle will play the role of the principal intermediate between the Moodle LMS and the X5-GON server. In summary, this is the process:

1. *The X5-GON connect-service* is activated on the suitable Moodle courses pages when the OER criteria are respected.
2. *The X5-GON server* is notified about any access occurring in any of these courses' pages.
3. *The X5-GON crawler* is launched on the page which is the origin of notification.
4. *The X5-GON crawler* crawls the *course page*, origin of notification and fetches all related urls which are potential OER resources.
5. *The X5-GON crawler* communicates with the X5-GON Data API (the second part of X5Moodle) to obtain the relevant meta-data about the collected resources.
6. *The X5-GON Data API* fetches the OER meta-data by a direct communication with Moodle.
7. *The X5-GON crawler* returns the OER meta-data to the X5-GON server in JSON format.

5. How to install

5.1. Installation

It is easy to install the *X5Moodle plugin*. Briefly, the plugin will be functional once the next steps have been followed:

1. Fork the last version of X5Moodle from Github or Gitlab.
2. Remove the .git folder from the forked colne, then compress it and rename it on the "mod_xfgon_moodle39_2021022301.zip" format (2021022301 is the last version). This step is to be done, if you get the plugin from a git repository and to make sure that Moodle will detect automatically the plugin type.
3. Using the Install Plugins wizard of Moodle, install it from "Plugins / Install plugins" under your Site administration panel of your Moodle administration GUI (as an activity plugin , if not automatically detected).
   Moodle will detect the new plugin (by verifying notifications section), then will ask you to install it.
   Alternatively, you can import the plugin folder in the mod folder of Moodle installation, if you have enough permissions for that.
4. Proceed with the plugin installation.

At this step and if the plugin is enabled, the activity part of the plugin can be used. The following steps will run you through configuring the Data API part.

5. Enable Web service under "Plugins / Web services".

6. Enable XML-RPC Protocol from Manage Protocols under "Plugins / Web services".
7. Create a user (system user, or a user with limited permissions) with at least the ability to use the XML-RPC Protocol.
8. Create an external service under "Plugins / Web services / External Services" associated to the X5Moodle plugin newly installed, **if it is not automatically created**.
9. Generate a Token associated to the external service (newly created related to X5-GON plugin) and the user (newly created). Here, you can precise a number of restrictions on how to use the webservice: IP, validity durations, users…
   The generated token is meant to be used by the external parties when they make their xmlrpc requests to Moodle.
10. Communicate the generated Token to X5-GON server admin with a secure way.
11. Configure OER categories field in plugin settings page.
12. Do the **first** save changes in the plugin settings page to store plugin settings in Moodle DB for the first time.

5.2. Plugin settings page

This section explains the main functionalities of the plugin settings page and the different settings that admin can control in order to better characterize OERs within his LMS.

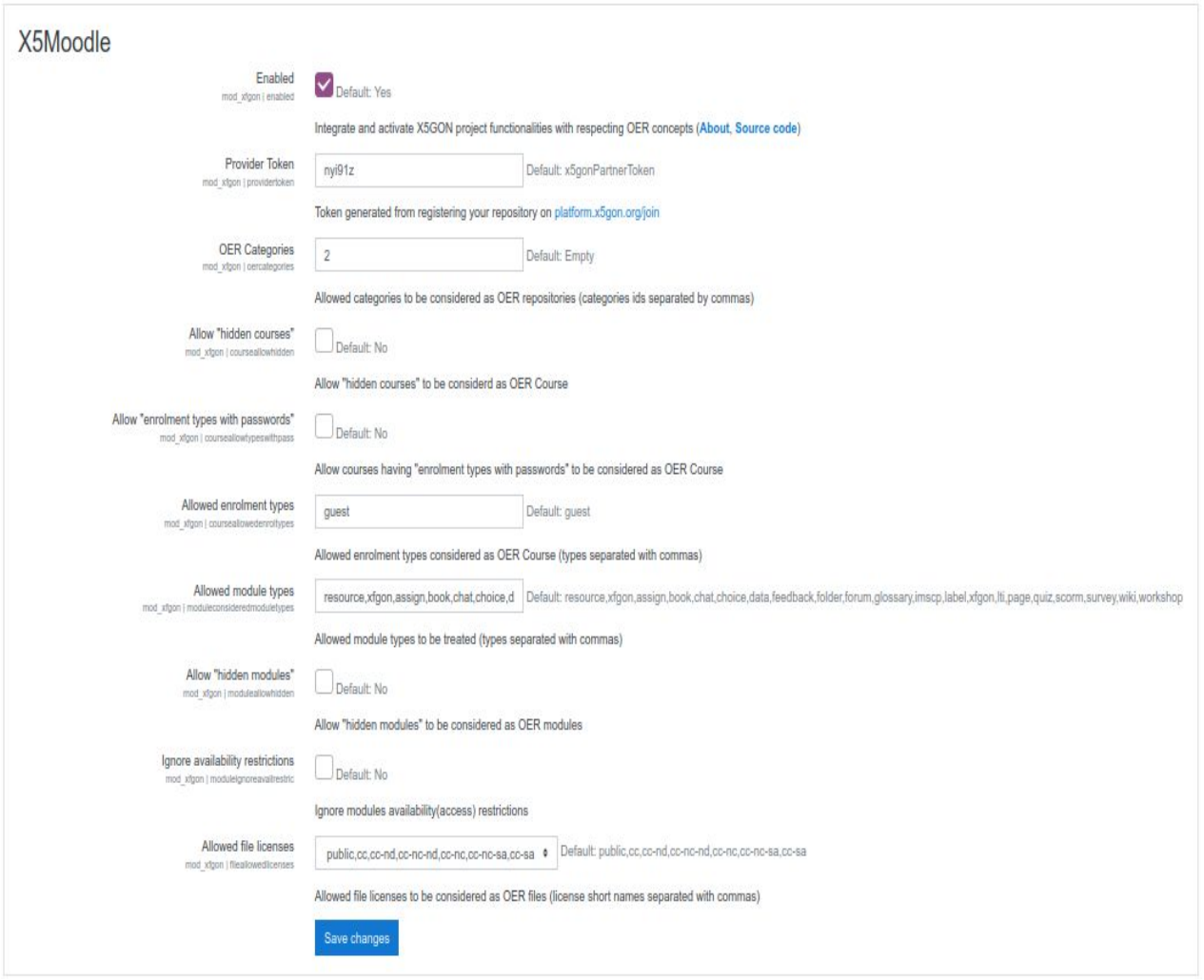


Fig. 10: X5-GON plugin settings page.

Enabled: Activate/Deactivate the plugin.
OER Categories: List of categories IDs (separated with commas) that meant to be containing OERs.
Allow "hidden courses": Consider hidden courses also as OER courses.
Allow "enrolment types with passwords": Consider courses with enrolment types using a password as OER courses.
Allowed enrolment types: List of allowed enrolment types (separated with commas) for which the courses are considered as OER courses.
Allowed module types: List of allowed Moodle module types which are treated by the plugin.
Allow "hidden modules": Consider hidden modules also as OER modules.
Ignore availability restrictions: Consider modules with availability restrictions also as OER modules.
Allowed licenses: List of licenses (licenses short names separated by commas).

6. Technical specifications

This plugin version has been developed and tested on Moodle version **6/7/8/9** which is based on ***PHP* 7**. So, it is optimized for the mentioned versions and it is theoretically compatible with more recent ones.

7. Future developments

7.1. Known limitations of the actual version

For the first part of the X5Moodle (connect-service integration), the main limitations are:

- The connect-service is not activated in the Categories pages[10]. This will be done in the next versions.
- The connect-service is not able to collect rich traces about user activities. It will send only the information about an access happening in a *course page* or a *course related pages*.

For the second part of the X5Moodle (Data API webservice), the main limitations are:

- Data API is treating only Moodle modules type **Resource** (especially **File** type). That means that for this particular type of module the response is more detailed: the plugin will fetch all meta-informations about them and their attached final files (media, files, pdf..). But for the other types, only general meta-data is returned (module type, name, description...).
- Resources embedded in sections, courses, categories descriptions and in general in any section description are ignored for this version.

7.2. Future developments

The next versions will improve plugin functionalities mainly at the level of treating the different Moodle modules types as much as possible and enrich access information sent by the connect-service to X5-GON server. This could be summarized by :

- Treating the use cases of embedded resources in the different Moodle description sections (in category, course, modules, ...)
- More control for admin to better control the different fonctionnalites of the plugin: enable/disable, set of allowed categories, set of OER criterias, restrictions of users ....

7.3. Good to know about Moodle treated cases

We specify here the cases considered by the X5-GON plugin. Moodle accepts many module types and each of these has its different way to include OERs (final resources: files, media files...).

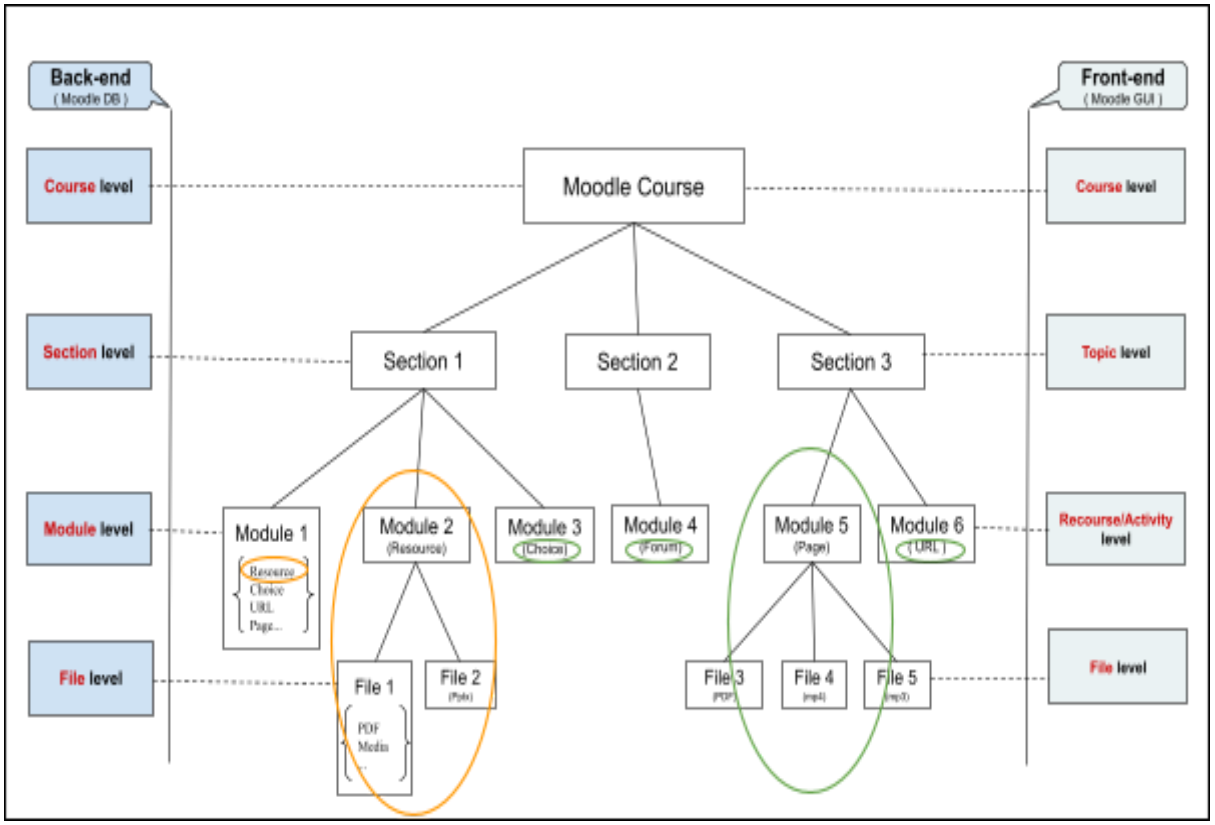


Fig. 11: Moodle course hierarchy components (Back-end side).

The Fig. 10 shows how a Moodle course is composed and the different important levels in Moodle course structure.
Final files (resources) can be included in one of the different levels of the hierarchy shown in Fig. 10:

- **Course level**: final files can be included in the course description.
- **Section level**: final files can be included in the section description.
- **Module level**: final files can be included in the module description or content.
- **File level**: this is the final level where final files are stored as final resources.

For this current version:

- **Resource** Moodle module type (equivalent to "File Resource" in Moodle GUI) is the only one well treated and detailed (highlighted with orange color in Fig. 10). So by requesting the meta-data of an OER for this type of module, X5Moodle returns its general information as well as the meta-data of its attached final files such as: author, license, creation time….
- ***The Other*** Moodle module types such as Choice, URL, Page… are less treated (highlighted with green color in Fig. 10). For these module types, X5Moodle returns only their general information such as: name, creation time... .

- Embedded files embedded in any type of description section aren't treated: the same for category, course, section or module level (their corresponding description sections).

Lexicon

1.OER

Open Educational Resources.

2.sub-pages

Or Course related pages are Moodle pages which can be accessed while visiting a specific course: such as the pages of different types of modules that can compose a course (Quiz page, media player page...).
Example: Choice/Page/Media Resource Moodle modules within a course are always opened in a standalone web page when you try to access it.

3.Course visibility

Moodle internal status for courses which concerns courses that are in preparation and not ready yet to be accessed by users. By default, X5-GON plugin considers only ready and visible courses. This can be extended through the plugin configurations by setting:

Allow "hidden courses".

4.Course enrolment type

Moodle enrolment types to courses. Mainly, concerns how the course author wants users/learners to access his course (as guest/self, with password, with authentication...). By default, X5-GON plugin considers courses which have "guest + empty password" as enrolment type (type not needing neither passwords nor authentications: public courses) . This can be extended through the plugin configurations by setting :

Allow "enrolment types with passwords".

Allowed enrolment types.

5.Course module visibility

Moodle module visibility in the course page in the front of Moodle users: this is computed basing on both *section* (course section) and *module* visibilities. By default, X5-GON plugin considers only the visible Moodle modules. This can be extended through the plugin configurations by setting:

Allow "hidden modules".

6.Course module availability

Moodle module availability in the course page: this is some kind of rule to limit access to specific module. This is computed based on both section and module availabilities. As

restriction types, there are:
Date, User profile: considered as restrictions touching directly module content.
Grade, Activity completion: considered as restrictions touching more the user activity profile.
By default, X5-GON plugin considers modules with any of the above restrictions as not OER. This can be extended through the plugin configurations by setting:

Ignore availability restrictions.

7.File license

License attribute is one of the meta-data of the final resources (files, media, ...) included in Moodle modules of type *Resource*. By default, X5-GON will only index final resources having one of CC (Creative Commons) licenses. This can be extended through the plugin configurations by setting :

Allowed licenses.

8.Moodle module ID

Modules Ids are the ids exposed in the courses pages, exactly within the Urls pointing to different modules in any course page.
Example: http://moodleinstallation.com/mod/resource/view.php?id=212769

9.Moodle course ID

Courses Ids are the ids found in the Urls pointing to any course.
Example: http://moodleinstallation.com/course/view.php?id=29932

10.Category page

Are the web page of a specific category. In Moodle, a category can englobe one or more courses.
Example: http://moodleinstallation.com/course/index.php?categoryid=916

Figures

Fig.1

The Moodle plugin interface allowing a teacher to choose the X5GON activities: e.g. X5GON Search engine, the X5-Recommend and the X5-Playlist.

Fig.2

The X5GON Search Engine functionality in the X5-Moodle. With it, a student can search through OERs as well as see the most popular search queries.

Fig.3

The X5-Playlist functionality embedded in the Moodle course. The example shows the teachers playlist created using the X5Learn service.

Fig.4

An example of the integrated X5-Recommend list.It shows the recommended OER resources as well as the viewing statistics of the student and its fellows.

Fig.5

The X5GON privacy policy shown to a new Moodle user.

Fig.6

Example of *oerinfos* service: Example of *oerinfos* service output.

Fig.7

Example of *courseoers* service: Example of *courseoers* service output.

Fig.8

Example of *oerslist* service: Example of *oerslist* service output.

Fig.9

X5Moodle - How it's working: The plugin features in the interactions with Moodle and X5-GON server.

Fig.10

X5-GON plugin settings page: Plugin configurations page within Moodle LMS.

Fig.11

Moodle course hierarchy components (Back-end side): Treated cases by X5-GON plugin in the front of the global architecture of a Moodle course.