***Front***

Summary (one page)

Index

Introduction.

Objectives (incluyendo descripción del problema, estudio de alternativas y

metodología empleada).

Technological description (puede incluir especificación, diseño e

implementación).

Conclusions (incluyendo los logros principales alcanzados y posibles

trabajos futuros).

Bibliography

Appendix

**Summary**

***This document contains the description and development process of the Moodle Web Services Python Library. This library has been made to provide some external functions to facilitate the development of applications that interact with Moodle and to exploit this Moodle functionality.***

***Throughout this document we will see the Moodles requeriments, its main characteristics, its web services plug-in, the REST protocol sentences to interact with it and the python code to do it and the diverse functionality provided by this python library.  
  
In first place, we will see our starting Moodle's development point, what are his characteristics and what we can do with them.***

***After that we will analyze what kind of programs usually use Moodle like an external aplications and haw they interact with it to work.***

***Later, we will explain this python library solution to develop external applications which interact with Moodle.***

***Finally we see the development process of this library,*** ***his API and some examples of its use.  
  
Moodle has become a popular teaching support system all around the world. This and the increase of mobile capacities to interact with many systems in the web have made appear new applications for Moodle.***

***Because of this, Moodle developers decided to include some additional functionality to allow external applications work with Moodle since version 2.0.***

***These functions are implemented like a Moodle plug-in, and its functionality initially was limited.***

***Later, the list of available functions were extended on version 2.5, so you will be able to do many things that Moodle can do but with this external service.***

***However, the use of this Moodle functionality require some advanced knowledges about Moodle functionality and REST or other applications data exchange protocols.***

***This project tries to give some tools to configure Moodle easily to get this functionality with a python library to abstract at users of*** ***the Moodle communication process.***

***What is Moodle***

***Moodle is an open source code system used for teaching and learning purposes. It allows to create on line dynamic web pages, create courses and manage it with the web interface. This made it popular between teachers around the world as a tool to provide resources to students and support teaching. It needs a data base system and a web server to work properly. Moodle's users are classified by roles. One user is able to have different roles at the same time, and different roles depending on the Moodle's part that he is. For example, one user can have the 'teacher' role in one course and the 'non editing teacher' role in another course. These roles gives to users different capabilities on the system. Depending on this capabilities, users will be able to make some changes and consults to the system or not. By default, a user without role don't have any capability on the system, so they can't do anything. The roles are what give the users the chance to do things in Moodle. There are some roles predefined in Moodle:***

* ***Manger***
* ***Course creator***
* ***Teacher***
* ***Non-editing teacher***
* ***Student***
* ***Guest***
* ***Authenticated user***
* ***Authenticated user on front page***

***These roles can be assigned to work on these contexts:***

* System
* User
* Category
* Course
* Activity module
* Block

Categories contains courses, and courses contains different type of activities separated in blocks.

[***https://moodle.org/about/***](https://moodle.org/about/)

***Moodle requirements:***

* ***PHP***
* ***Database system: MySQL***
* ***Web server: apache***

Moodle Web Services:

Moodle web services are additional Moodle modules that let you to use Moodle with other protocols than http such as AMF, SOAP, REST and XML-RPC. This services has been tough for interacting Moodle with other applications.

***Creating an external application using Moodle via Web Services***

* ***Moodle requeriments:  
  REST protocol will be used in this application.***

*Using web services:* [*http://docs.moodle.org/24/en/Using\_web\_services*](http://docs.moodle.org/24/en/Using_web_services)

*To Allow users to create and see their own security keys(token) you must assign a system role with this user capabilities:*

* + *users* [*moodle/webservice:createtoken*](http://docs.moodle.org/24/en/Capabilities/moodle/webservice:createtoken) *(if not, the administrator must create the token manually for each user for the service)*
  + [*webservice/rest:use*](http://docs.moodle.org/24/en/Capabilities/webservice/rest:use)
  + *service required capability (depends of the service, check Settings-> Site administration-> Plugins-> Web services-> Manage services->****Edit)***
  + ***required capabilities for the service functions.***
  + ***Also you must enable the web service function documentation, in Settings-> Site administration-> Plugins-> Web services-> Manage protocols***
* ***Get web services token:***[***http://docs.moodle.org/dev/Creating\_a\_web\_service\_client***](http://docs.moodle.org/dev/Creating_a_web_service_client)

***Service short name info:*** [***https://moodle.org/mod/forum/discuss.php?d=197187***](https://moodle.org/mod/forum/discuss.php?d=197187)

* *Get JSON answers to REST requests:* [*https://moodle.org/mod/forum/discuss.php?d=204469*](https://moodle.org/mod/forum/discuss.php?d=204469)

*In the request you must include the parameter "moodlewsrestformat=json"  
example: /webservice/rest/server.php?wstoken=1cfc5fsd5a6fa75dfa&wsfunction=core\_webservice\_get\_site\_info&moodlewsrestformat=json*