



Virtual Portal for Interaction and ICT Training for People with Disabilities

Integrated ViPi platform

Outcome No.		13	
Workpackage No.	4	Workpackage Title	Deployment of ViPi interactive community portal and e-learning platform
Authors		Kosmas Petridis, Nikolaos Kakardakos (HYPERTECH), George Milis (EuroCy), Karel Van Isacker, Bernard Van Isacker (PhoenixKM)	
Status (F: final; D: draft; RD: revised draft):		F	
File Name:		ViPi-D13-Integrated ViPi Platform-v2.0.docx	

The ViPi KA3 LLL project (511792-LLP-1-2010-1-GR-KA3-KA3NW) has been partially funded under the Lifelong Learning program, subprogramme KA3 ICT. This publication reflects the views only of the author(s), and the Commission cannot be held responsible for any use which may be made of the information contained therein.



Version History table

Version no.	Date	Dates and comments
0.1	13-03-2012	First draft version that contains the adopted technologies, installation procedure and open development issues of the integrated ViPi platform (Hypertech)
0.2	25-05-2012	Updated version that contains the full functionality of ViPi portal (Hypertech)
1.0	28-05-2012	Final revised version (PhoenixKM)
2.0	20-12-2013	Additional revised version taking into account the modifications performed during testing & piloting (i.e. WP6 & WP7 implementation)(Hypertech, EuroCy)

Table of Contents

Version History table	2
1 Introduction.....	5
2 ViPi Platform Aim & Objectives	6
3 ViPi Portal Functionality	8
3.1 Registering with ViPi.....	8
3.2 Profile Annotation	9
3.3 Browsing through ViPi Portal	10
3.4 The ViPi Learning Object Repository	12
3.5 Contributing to ViPi / Submitting New Content.....	13
3.6 Semantically-enabled Search	18
3.6.1 Quick search	18
3.6.2 Guided Search	18
3.7 Connection with Popular Social Networks	20
3.8 Printing Pages/Posts.....	20
3.9 Using the ViPi Online Training Environment	21
4 Adopted Technologies & Installation Steps	22
4.1 Installation of Xampp	22
4.2 Installation & Deployment of WordPress.....	22
4.2.1 Enabling Permalinks	23
4.2.2 XML-RPC Services Export & Deployment	23
4.3 Installation & Deployment of the BuddyPress Plug-in	24
4.3.1 BuddyPress XML-RPC Services Export & Testing.....	26
4.4 Installation & Deployment of ATutor	26
4.5 Installation of Selected WordPress Plug-ins.....	28
4.5.1 WP-Post Ratings Plug-in	28
4.5.2 Multilingual Support.....	28
4.5.3 Facebook, Twitter & Google+ Social Widgets	29
4.5.4 Duplicator	30
4.5.5 Access Keys.....	31
4.5.6 Acronyms.....	31
4.5.7 All In One SEO Pack.....	32

4.5.8	Print me!.....	32
4.5.9	BuddyPress Moderation	33
4.5.10	Q and A – FAQ	33
4.5.11	Fast Secure Contact Form.....	34
4.5.12	Contact Form DB	35
4.5.13	Blog-in-Blog	36
4.5.14	Require Post Category	36
4.5.15	Search by Category.....	37
4.5.16	Advanced Browser Check	37
4.5.17	Disable Admin Bar	37
4.5.18	Force Post Title	38
4.5.19	Image Widget	38
4.5.20	WP User FrontEnd	38
4.5.21	Theme My Login	39
4.5.22	Remove Dashboard Access.....	39
4.5.23	Role Scoper	40
4.5.24	WP Accessibility.....	40
5	New Components Implemented inside ViPi.....	41
5.1	WordPress & ATutor Seamless Access	41
5.2	ViPi Semantic Content Management (VSCM) plug-in	41
5.2.1	Ontology Management	41
5.2.2	Administration Component.....	42
5.2.3	Annotation of Learning Objects (LOs)	44
5.2.4	Annotation of User Profiles	44
5.2.5	Quick and Guided Search	44
5.2.6	XML-RPC Interface.....	44
5.2.7	Maintenance & Exploitation.....	47
	Annex I: ViPi Semantic Content Management (VSCM) Module.....	48
	Annex II: ViPi Semantic Content Management (VSCM) Ontology	49
	Annex III: Evaluation of Accessibility Plug-ins	50

1 Introduction

The main purpose of the deliverable “D13-Integrated ViPi Platform” is to describe the functionality, the adopted technologies, as well as, the components used to launch the ViPi platform. Chapter 2 describes the ViPi portal aims & objectives. Chapter 3 presents the ViPi portal with all the integrated functionality. Chapter 4 describes the adopted technologies & the necessary installation steps that need be followed in order to deploy the integrated platform from scratch, whereas a solution is also provided to migrate an existing ViPi platform installation as well as backup the platform periodically. Chapter 5 is dedicated to the description of the new components & WordPress plug-ins that were developed during the WP4 development.

Finally, Annex I includes the initial high-level design approach for the ViPi Semantic Content Management (VSCM) module, Annex II presents the corresponding VSCM ontology implemented, while Annex III describes the extensive evaluation of accessibility plug-ins performed during the ViPi platform implementation.

2 ViPi Platform Aim & Objectives

A majority of Europe's people with disabilities are unemployed (ANED figures state over 60%). There are many reasons for this. Many people with disabilities have advanced skills but suffer significant barriers to employment. This is an important issue which must be addressed. For others, skill acquisition must be facilitated. ViPi aims to support and facilitate the acquisition of basic ICT skills for those people with disabilities who lack them.

ViPi aims to provide a "one-stop-shop" interactive portal & learning environment for ICT skills by delivering a comprehensive multilingual portal, with:

- an embedded multilingual social community (for VET centres, people with disabilities, ICT training centres, etc.), using the latest social media, facilitating interactive information sharing, interoperability, and collaboration, with access to all;
- a fully accessible (WCAG 2.0) multilingual Web 2.0 enabled online learning environment for ICT for people with disabilities and their trainers, with an interactive and extensive repository of interoperable SCORM compliant learning objects (LOs) that focus on basic ICT literacy to support the acquisition of core skills by people with disabilities, in order to facilitate their entry into or sustain their employment in, the regular labour market; LOs will be enriched with intuitive mobile Java and Flash Lite based mobile and internet/PC based educational/serious games;

The integrated ViPi platform, which is described in this deliverable, will equally act as a "one-stop-shop" for trainer organizations to find and contribute LOs that they can integrate in existing learning environments and practices. Using Open Source Software, ViPi platform extensions can be easily integrated. The ViPi platform will thus be able to support a fully accessible and Open Source based pan-European learning network and community, bringing together key stakeholders and gatekeepers (VET, target groups, umbrella organizations), while offering a vast set of reusable (PC and Mobile) LOs, supported by Web 2.0 social services.

ViPi project envisages fulfilling the gap of accessible and flexible training, designed to meet the specific needs of people with disabilities, as well as the trainers themselves. ViPi project develops an interactive online platform where people with disabilities can access a wide variety of ICT training courses and serious games for acquiring ICT skills, while trainers are able to upload and download specific learning objects, using a semantically enriched environment to improve the searchability, thus increasing the relevance of the results obtained.

This will allow people with disabilities to grasp these core skills and become able to enter or sustain their employment in the regular labour market. The platform will be enriched with intuitive and accessible mobile and Internet/PC based educational/serious games. Additionally, also a mobile Android based social application linked with ViPi platform is also deployed.

The ViPi integrated platform is currently available in Dutch, English, Greek and Lithuanian languages.

It is obvious from the above objectives, that the ViPi platform adopts efficiently the concepts and technologies of Web 2.0, that is, it moves beyond the traditional one-way interaction of the Internet, providing collaborative and social networking facilities. The importance of this fact is twofold: i) The two-way interaction capabilities of the Web 2.0 are employed also for the benefit of people with disabilities, increasing their quality of life and communication means, ii) The social networking technologies are employed in the training/education domain, showing the path to efficient technology adoption in domains other than the pure leisure communication.

Moving one step further, the ViPi project also explores the adoption of the Web 3.0 concepts. That is, the environment of the ViPi platform is semantically enriched, thus improve the search-ability and relevance of the results obtained. The ViPi platform is enhanced with semantic annotation and discovery of content, thus moving beyond the basic keyword search functionality and adopting at the extent possible the value of the Web 3.0.

The following sections provide further insight and detail the functionality and design considerations of the integrated ViPi platform.

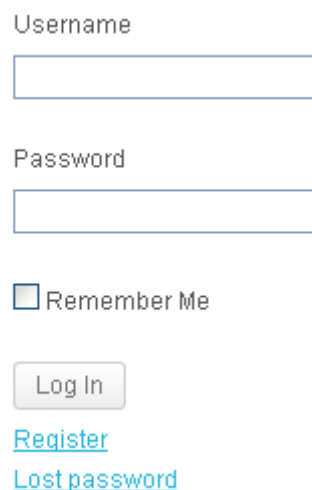
3 ViPi Portal Functionality

ViPi portal is accessible through the following Web address:

<http://vipi-skills.eu/> or <http://vipi-skills.eu/portal/>

3.1 Registering with ViPi

The registration of the ViPi portal is essential for every user that wants to have access to the ViPi content. This can be performed through the main page of the portal:



Username

Password

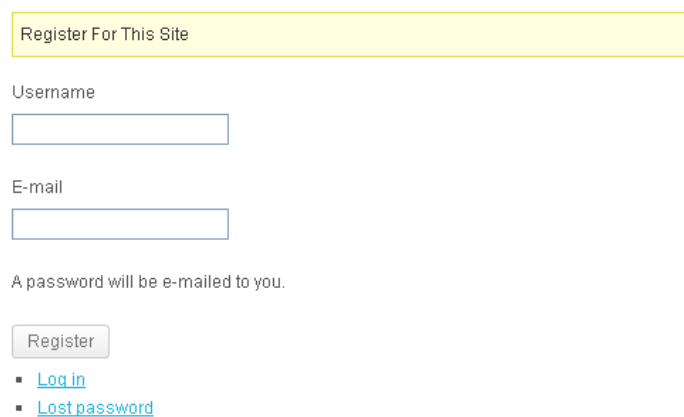
☐ Remember Me

[Register](#)
[Lost password](#)

Figure 1: The registration and login panel of ViPi portal

If the “Register” option is selected, then the user is redirected to the following page, where he/she is prompted to enter all required data. More specifically, username and email address are required for registering into ViPi portal.

Register



Register For This Site

Username

E-mail

A password will be e-mailed to you.

- [Log in](#)
- [Lost password](#)

Figure 2: User fills in all required registration data

Once submitted, the user has to activate the registration by following the instructions that have been sent to his/her email.

3.2 Profile Annotation

One essential part of the registration process, is to provide a detailed profile with option to indicate interested fields that match user preferences. Interested fields are concepts (classes or instances) from the ViPi ontology that will be presented in a following section of the document. This option is available under the Profile page of a registered user.

The profile annotation of each user could be performed by dragging and dropping the desired and relevant concepts from left to the right. Once concepts are dropped into the “User annotations” tree, user profile is automatically annotated with the selected concept. The annotation process can be performed also by selecting a node and pressing the key “d” on the keyboard, while selecting a node on the right tree and pressing the key “del” results on removing the selected annotation. The specific procedure will later allow the personalization of the navigation experience of the user within the ViPi platform.

My profile description and preferences

- Please, drag appropriate terms from the left to the right, to describe your profile and preferences.
- Select a node and press "d" on your keyboard to annotate. Select a node and press "del" on your keyboard to delete an annotation.
- Select a term under the "Role(s) in the platform" node to request for upgraded role in the portal ("Trainee" is the default option).

Available selections	My profile description and preferences
<ul style="list-style-type: none"> <input type="checkbox"/> ICT skills <input type="checkbox"/> Type(s) of impairment or limitation <input type="checkbox"/> Role(s) in the platform <input type="checkbox"/> Assistive Technologies Used/Required <input type="checkbox"/> Type(s) of content <input type="checkbox"/> Language(s) of content <input type="checkbox"/> Payment method(s) <input type="checkbox"/> Accreditation types <input type="checkbox"/> Accessibility levels <input type="checkbox"/> Age group(s) <ul style="list-style-type: none"> <input type="checkbox"/> Adult (> 16 years old) <input type="checkbox"/> Child (0 to 10 years old) <input type="checkbox"/> Teenage (10 to 16 years old) 	<ul style="list-style-type: none"> <input type="checkbox"/> My profile description and preferences <ul style="list-style-type: none"> <input type="checkbox"/> English <input type="checkbox"/> Greek

Figure 3: User profile annotation

It should be noted that if a user selects a term under the “Role(s) in the platform” option (“Content Provider”, “Trainer” or “Trainee”), it is assumed as a request for upgraded role in the portal (“Trainee” is the default option).

3.3 Browsing through ViPi Portal

Once user visits the portal, the following main web page appears.

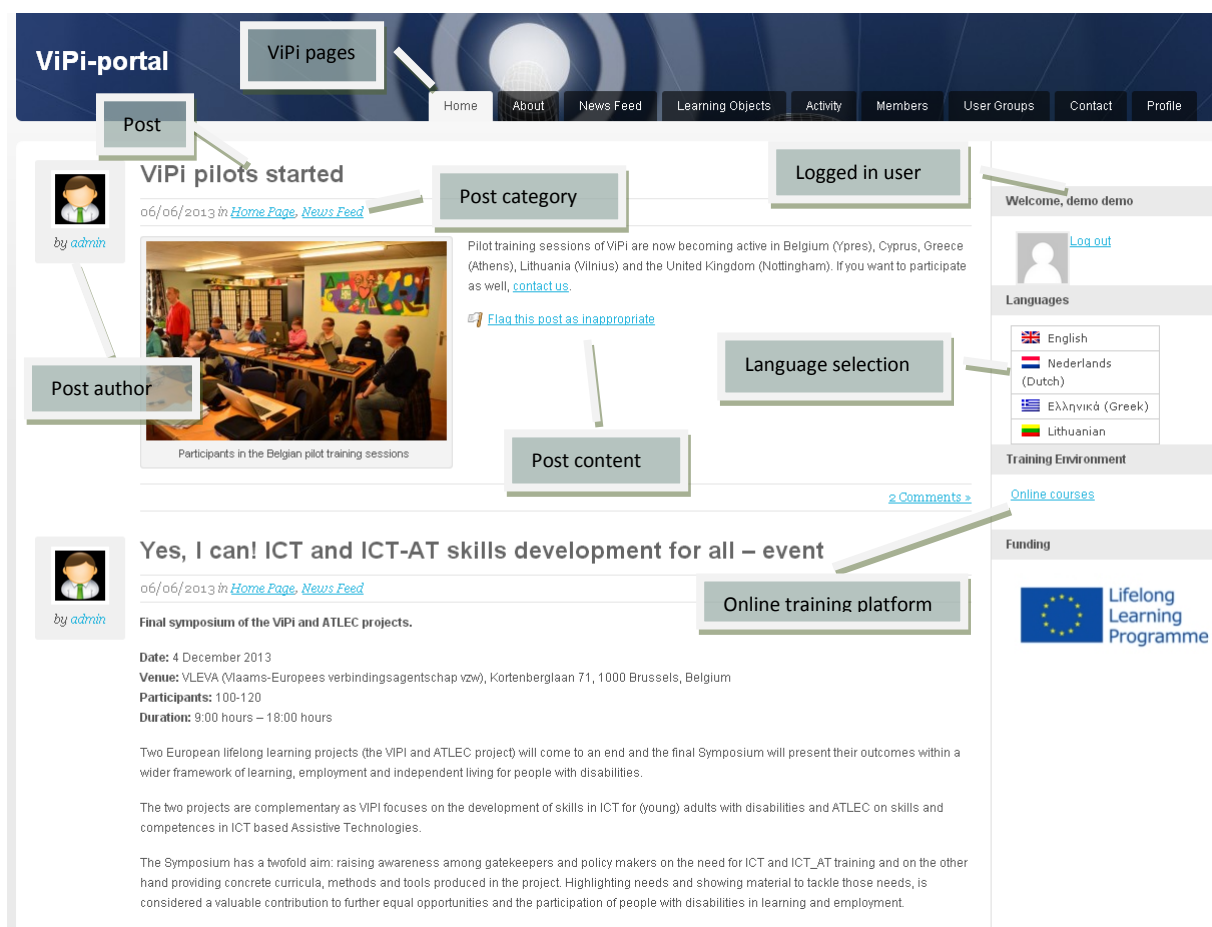


Figure 4: ViPi portal main page

The following sections appear in the main page of ViPi portal:

- **ViPi pages:** ViPi pages provide a significant functionality for accessing main content of the portal. "Home page" is the default page that appears when a user logs in the portal and displays the most recent posts that are submitted to the platform by all users under the "Home page" and "News Feed" categories. "About" page gives a brief overview of the platform, while "News Feed" provides posts that are submitted to the "News Feed" category. "Learning Objects" page lists all posts that are under the category "Learning Object" and compose the main content of ViPi platform. This page also contains a built in search functionality on top of the page for searching content within only Learning Objects. "Activity" page provides an overview of recent user's activity on the portal, "Members" an overview of all registered users to the platform and "User Groups" an overview of users groups that are created within ViPi platform. "Contact" page provides a form to communicate with the portal administrator, while the "Profile" page provides the functionality to edit the logged in user's profile. Finally, the "New Learning Object", which is visible only to certain users (authorised



Education and Culture DG

Lifelong Learning Programme

ViPi Project 511792-LLP-1-2010-1-GR-KA3-KA3NW

with “Content provider” role), allows the submission of new Learning Objects to the portal. A sample of these pages is displayed below.

The collage displays six screenshots of the ViPi-portal interface:

- ViPi pilots started:** Announces pilot training sessions in Belgium (Greece, Cyprus, Greece, Lithuania, Lithuania) and the United Kingdom (Netherlands). It includes a photo of participants and a link to a video.
- Yes, I can! ICT and ICT-AT skills development for all – event:** Announces a final symposium of the ViPi and ATLE projects, dated 4 December 2013.
- ViPi symposium agenda:** Details the agenda for the symposium, including the date (4 December 2013), venue (L'Eclat, European Parliament), location (Brussels), and topics (ICT and ICT-AT skills development).
- Learning Objects:** Shows a search interface for Learning Objects, with filters for ICT skills, type of impairment, and content type.
- Members Directory:** Displays a list of members, including their names, roles, and contact information.
- Groups Directory:** Displays a list of groups, including their names, descriptions, and contact information.

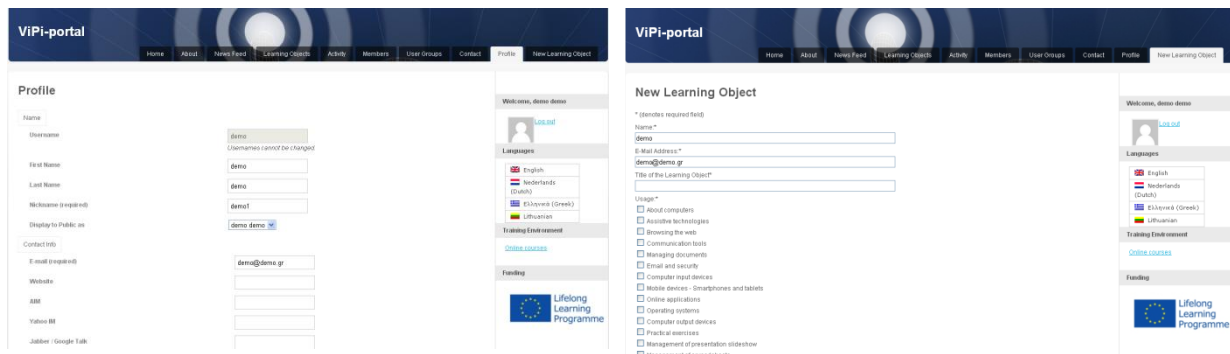


Figure 5: ViPi portal overview of pages

- **Language Selection:** Language selection enables multilingualism in ViPi portal by displaying all content in the selected language. Available languages that are supported are English, Dutch, Greek and Lithuanian. When a language is selected, e.g. Greek, then all portal content is displayed in Greek.

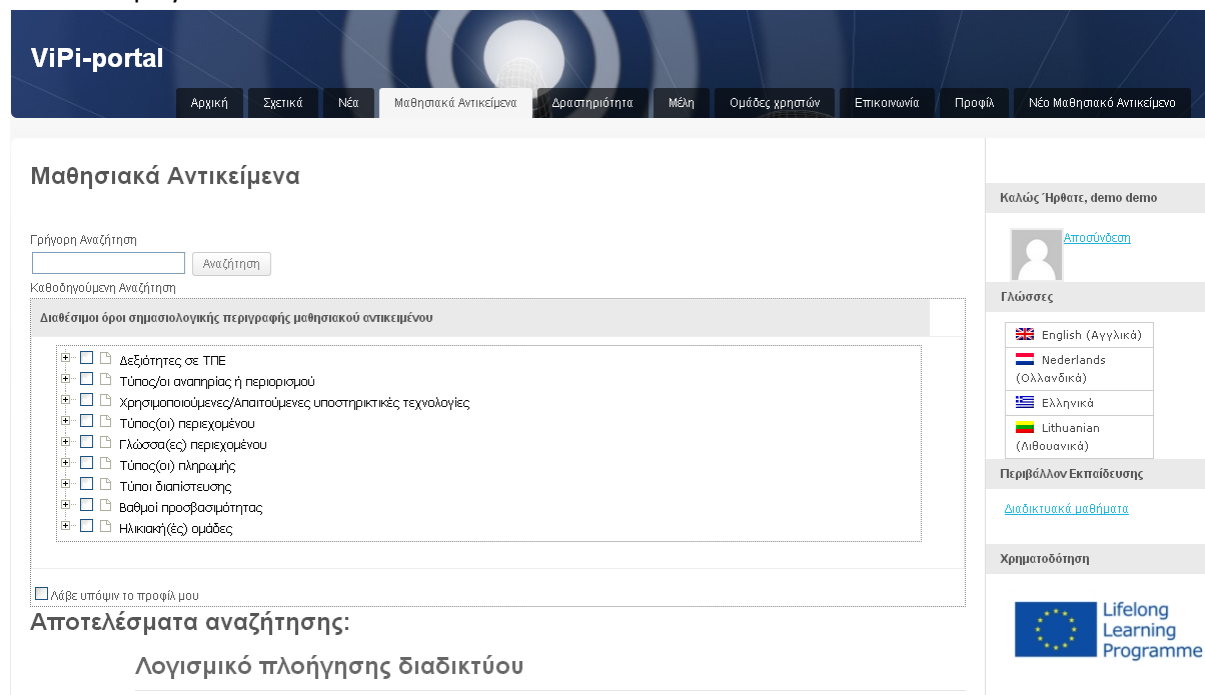


Figure 6: Example of Learning Objects page in Greek

- **Post:** All posts that are under the category “Home page” or “News Feed” are displayed in the main page in descending order according to the date they were submitted. Each post is consisted of a title, the content of post and its comments if available. Additionally, the category of the post appears after the date that it was submitted.

3.4 The ViPi Learning Object Repository

An important part of the ViPi portal is the Learning Object (LO) repository that can be accessed through a dedicated Web page provided as an item in the central menu of the portal’s homepage.

The repository is also accessible through ViPi’s mobile application (Android based). This repository strives to become a one-stop-shop of learning content created by trainers and other content

providers, for the training of persons with disabilities, and providing trainers and other stakeholders with learning content in a variety of formats (URLs, courses, serious games, mobile games, etc.). Initially the repository contains fully accessible training material for basic ICT skills, which was the application area of the ViPi project, however, the repository is implemented with built-in scalability to host content from any other area in the future and of any accessibility level.

Learning Objects is the terminology used in the ViPi portal to describe any piece of content that can be considered as content/material useful for the training of people. Therefore, LOs can span from structured courses, to small pieces of textual information, to multimedia files, to links to other resources, to serious/educational and other games, etc.

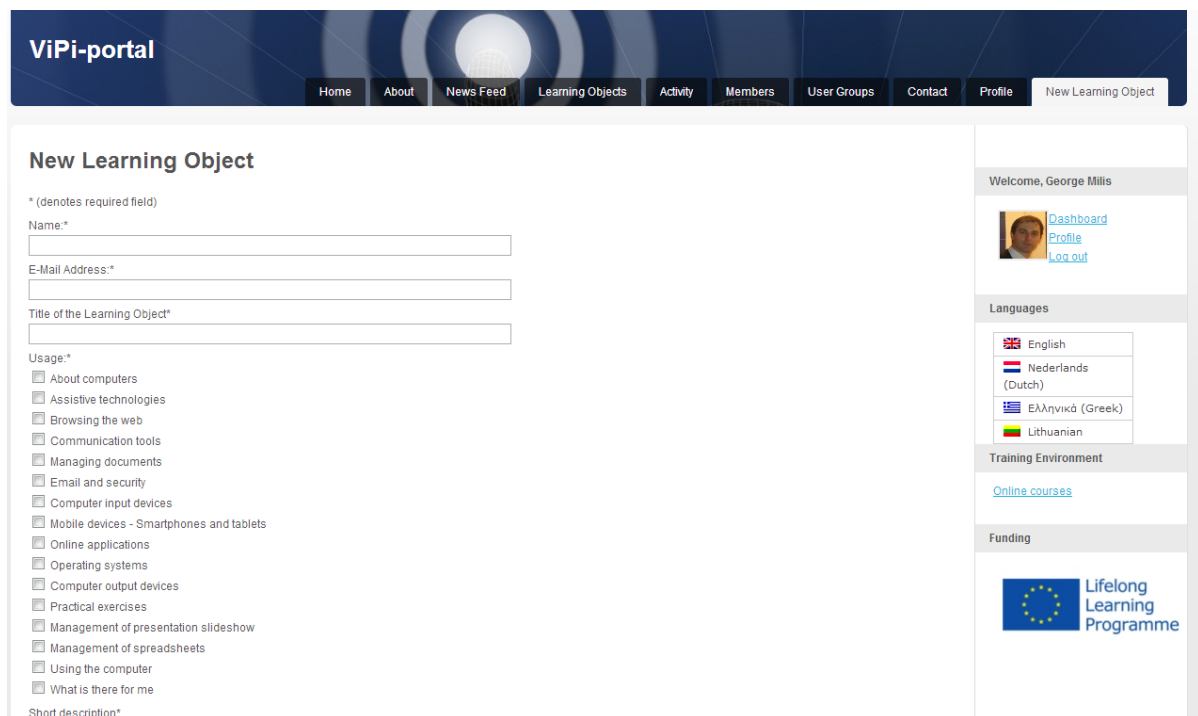
The repository is implemented as multilingual, that is, the content can be provided in different and in more than one languages, as long as the content provider performed and uploaded the required translations.

The LOs are a specific category of content in the ViPi portal and, therefore, dedicated search functionality is available to the interested users to locate the content they look for. Details on how to use the search capabilities are given in a later section. LOs are stored in the repository as individual and well separated entities, with title, content and other metadata that support their discoverability. Furthermore, they are all linked to specific categories of skills (currently only basic ICT skills), as well as, to specific types of disabilities, devices, etc. The LOs are expected to be used by trainers in their teaching either using the ViPi curriculum or using their own curriculums and practices, but also by trainees that need to find material on specific topics and suggest it to their trainers or read on their own, possibly while taking the courses through the ViPi training environment.

Overall, the ViPi LOs repository is the hosting place for a vast number of training content entities/material and offers an easy-to-use search mechanism to facilitate the efficient discovery of what the user is exactly looking for.

3.5 Contributing to ViPi / Submitting New Content

Submission of new Learning Objects to be published in the ViPi portal, can be performed through the main menu of the portal. The relevant functionality becomes available with a link in the main menu, as soon as a user with sufficient access rights (administrators and content providers) logs-in the portal. Figure 7 presents this functionality.



ViPi-portal

Home About News Feed Learning Objects Activity Members User Groups Contact Profile New Learning Object

New Learning Object

*(denotes required field)

Name:*

E-Mail Address:*


Title of the Learning Object*

Usage:*





- ☐ About computers
- ☐ Assistive technologies
- ☐ Browsing the web
- ☐ Communication tools
- ☐ Managing documents
- ☐ Email and security
- ☐ Computer input devices
- ☐ Mobile devices - Smartphones and tablets
- ☐ Online applications
- ☐ Operating systems
- ☐ Computer output devices
- ☐ Practical exercises
- ☐ Management of presentation slideshow
- ☐ Management of spreadsheets
- ☐ Using the computer
- ☐ What is there for me

Short description*

Welcome, George Mills

 [Dashboard](#)
[Profile](#)
[Log out](#)

Languages

	English
	Netherlands (Dutch)
	Ελληνικά (Greek)
	Lithuanian

Training Environment

[Online courses](#)

Funding


 Lifelong Learning Programme

Figure 7: Submission of new Learning Objects

Once the Content Provider user has selected to submit a new Learning Object, a page is opened which contains a form that has to be filled in. First of all, the title of the new object needs to be entered in the preferred language. Then the ICT skills with which the learning object is associated, need to be indicated, followed by a short description of the learning object and detailed explanation of its operation/usage. Finally, the user is asked to give information about the target user groups, the operating systems on which the learning object can operate, as well as cost information and links to download or attached file.

The submission of a learning object does not initiate its immediate publication. A relevant email is sent to the Administrators' group (depending on the language of the post) and the latter will check the learning object, validate its applicability and publish it through the dedicated functionality of the ViPi Portal Content Management System.

The following paragraphs describe the publication process of the Learning Objects, as well as of other content type. First the title of the entity post can be given in the selected languages, as shown in Figure 8.



 Add New Post

Figure 8: New post title in the selected language

The language of the post can be defined by the selection box of Figure 9

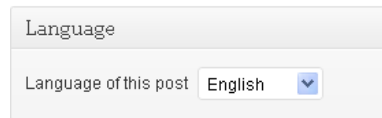


Figure 9: Language selection of new post

The next step is to provide the main content of the post. The form in Figure 10 is used, where user should fill in the content.

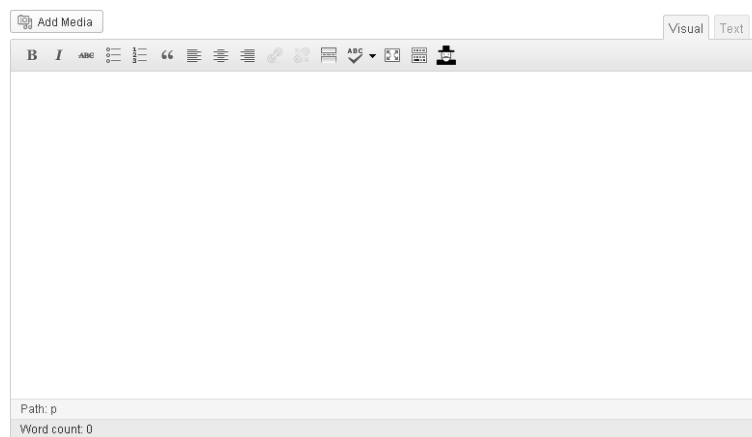


Figure 10: New post content in the four languages

Each content submission form includes text formatting options, as well as buttons for uploading images, files, videos etc. Presentation of HTML code is also supported by selecting the appropriate tab.

The next step is to select the category of the post. All available categories are listed in the box shown in Figure 11.

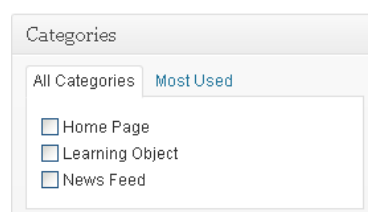


Figure 11: Category selection when submitting a new post

Category selection is mandatory, so if no category is selected, and the user tries to submit the post for review, then the following popup message will appear.

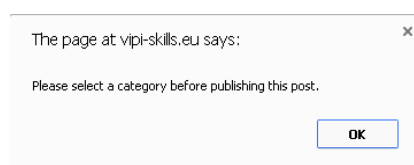


Figure 12: Category selection alert window

The selected category should reflect the nature of the post. For example, if the “News Feed” category is selected, then the post will be automatically displayed under the “News Feed” page and should contain news information related to the ViPi community. Accordingly, if the “Learning Object” category is selected then the content of this post should contain a new Learning Object, and this will be again automatically displayed under the corresponding ViPi portal page. Multiple categories can be selected inheriting the properties of each of the selected categories.

When the “Learning Object” category is selected, the following meta-box appears at the bottom of the page.

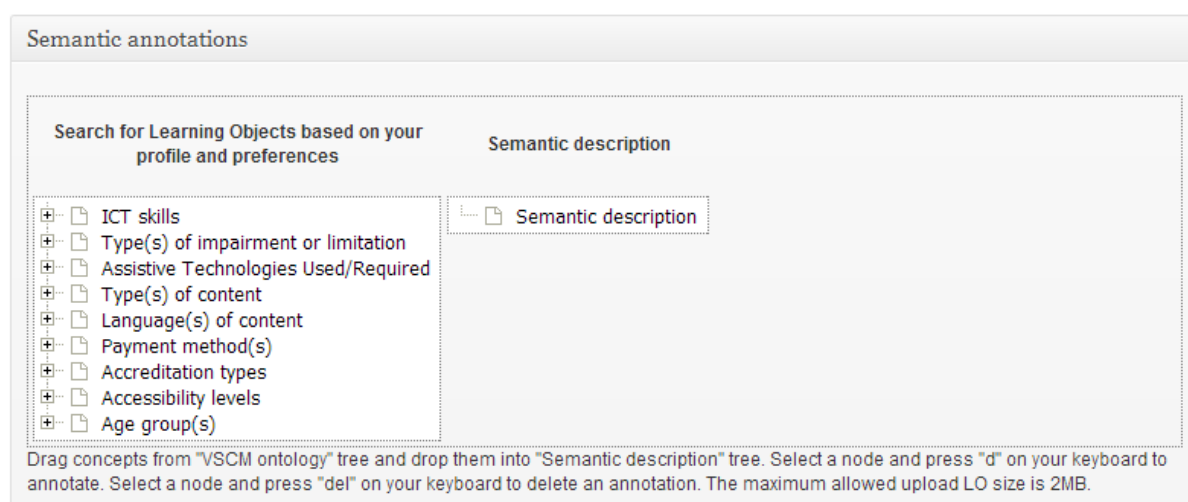


Figure 13: LO semantic annotations meta-box

This meta-box provides the ability to semantically annotate the post using the ViPi ontology supported by the ViPi Semantic Content Management (VSCM) module, which is presented in a later section. As shown in Figure 13, two trees are displayed: the VSCM ontology and the Semantic annotations tree. The VSCM ontology tree, displays the ontology concepts (classes and individuals) which are defined as associated with the annotation of learning objects. That is, in technical terms, the ontology object property

[“http://www.eurocyinnovations.com/neologism/vscm#forLOAnnotation”](http://www.eurocyinnovations.com/neologism/vscm#forLOAnnotation)

is set to “YES”. The tree is fully expandable, while the annotation is performed by dragging concepts/objects from the left tree and dropping them to the right one, as shown in Figure 14. The main objective is to describe the LO by specifically defining its characteristics. Selection of a leaf node, means that the LO has exactly that characteristic. For example, selection of “ECDL” means that the LO is suitable for that accreditation level. Also, selection of the “Creating and saving documents with Libre Office” means that the LO addresses exactly that ICT skill. On the other hand, selection of a class-node (that is, a node which has child-nodes), means that the LO has ALL (with AND) child characteristics. For example, selection of the “Web browsing software” means that the LO covers all topics under that node. Also, selection of the “Language” means that the LO content is available in ALL languages defined. Finally, selection of multiple leaf-nodes and/or multiple class-nodes is always treated with AND.

It should be noted that, instead of the real ontology concept name, a user-friendly text is displayed for each concept to assist the annotation procedure, translated into the user's selected language (current supported languages are English, Dutch, Greek and Lithuanian).

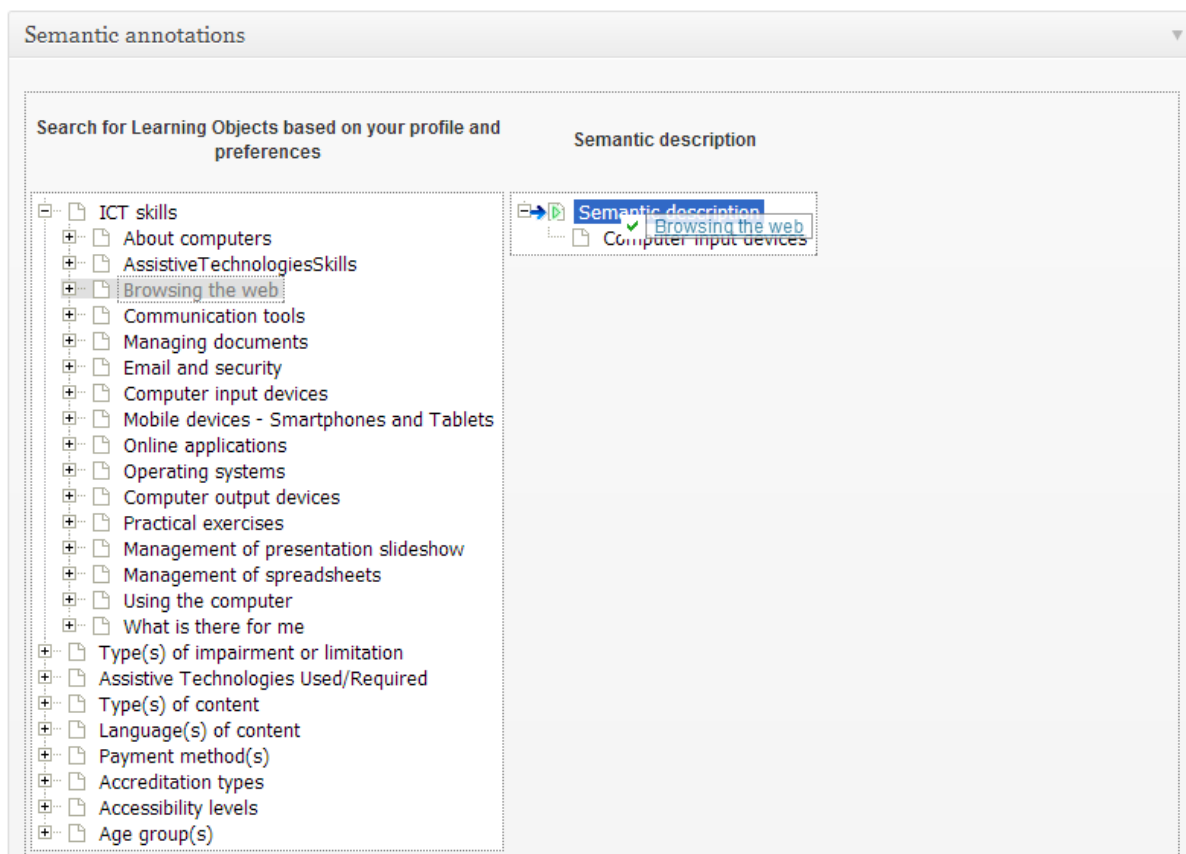


Figure 14: Drag and Drop functionality for annotating learning objects

Once a concept is dropped into the “Semantic annotations” tree, then the post is automatically annotated (associated) with this concept. It is noted here, that the annotations need to be provided in all languages of the post. The translations are handled internally by the module's ontology.

As soon as the creation of the learning-object is completed (including its annotation), the administrator-user can preview the post or save it for later usage. Once submitted, the status of the post is set to “Final”, and it will be available through ViPi portal to all public and registered users.

As already explained, the ViPi portal is the interface to the ViPi Learning Objects (LO) repository. The LO repository is expected to be enriched by training content to be contributed by persons external to the ViPi team. However, in order to maintain a high quality for the content in the repository, all potential training content providers are expected to follow the registration procedure offered on the ViPi portal. To further ensure the quality of the content, only approved and registered (on the ViPi platform) trainers are allowed to take the role of “Content Provider”.

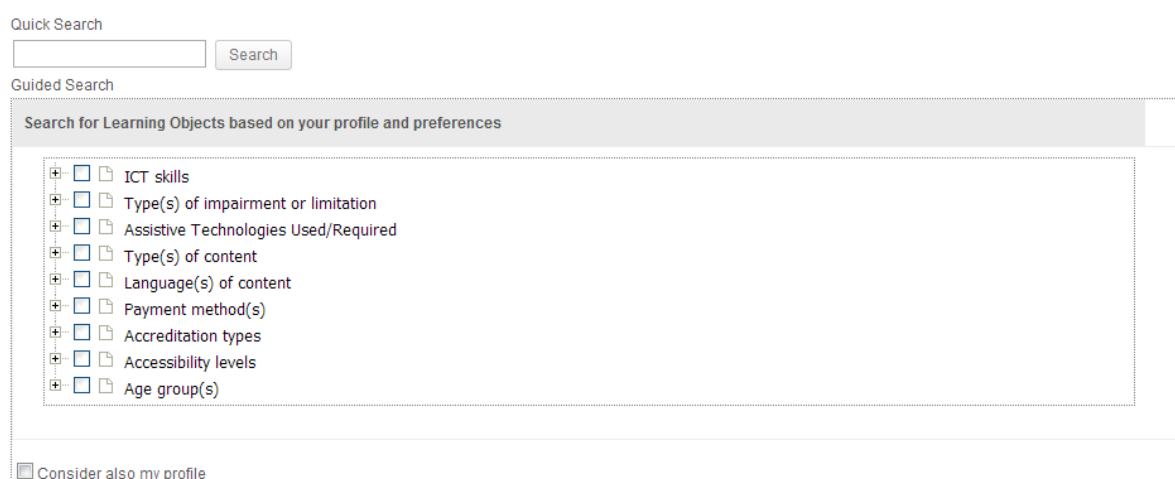
3.6 Semantically-enabled Search

A dedicated search functionality is integrated and available through the “Learning Objects” page of the ViPi Portal. The two searching options, Quick and Guided, are presented in the next paragraphs.

3.6.1 Quick search

The Quick Search box can be found immediately under the title “Learning Objects” in the relevant page of the Portal, as shown in Figure 15. This offers the standard search functionality within content classified under the Learning Objects category. The user enters the desired keywords in the text box provided, and by clicking the “Search” button, the search engine retrieves all learning objects that have the given keywords/phrases in their title and/or content. The results are displayed in the same page, immediately under the Search Results sub-title. It is understood, that the language used in the given text will affect the returned results, since the search is performed through the content without any extra embedded intelligence.

Learning Objects



Quick Search

Guided Search

Search for Learning Objects based on your profile and preferences

- ☐ ☐ ICT skills
- ☐ ☐ Type(s) of impairment or limitation
- ☐ ☐ Assistive Technologies Used/Required
- ☐ ☐ Type(s) of content
- ☐ ☐ Language(s) of content
- ☐ ☐ Payment method(s)
- ☐ ☐ Accreditation types
- ☐ ☐ Accessibility levels
- ☐ ☐ Age group(s)

☐ Consider also my profile

Figure 15: The LO search options

3.6.2 Guided Search

The Guided Search component is also visible in Figure 15. This search option provides a widget that allows navigation through the ViPi ontology.

By selecting classes or instances of the ontology tree, the user forms his/her preferences for Learning Objects. Selections are performed by clicking in the empty boxes on the left of each concept/instance. At each moment, the learning objects that match the current selections are displayed at the bottom. The titles of the learning objects are actually links to dedicated pages for the whole LO content. The VSCM ontology is always presented in the language defined by the user and all translations and mappings to other languages are performed automatically with no inconsistency.

The guided search mechanism in the current version of the VSCM plug-in operates with the following behaviour:

Anybody can use this functionality. If logged-in the user has the extra option of considering also his/her profile (see at the bottom of Figure 15. When selecting leaf-nodes and/or class-nodes manually to perform guided search, the user actually extends or defines his/her profile. That is, he/she describes his/her profile and preferences and the engine will then decide which LOs need to be shown. The same policy applies as in the user profile annotation case. Selecting (or having selected in the profile) a leaf-node means that any LO suitable with that specific characteristic/preference will be returned. E.g. selecting "Lithuanian" means that the LOs that are available in Lithuanian will be returned (Note: the language here does not refer to the language of the post in the ViPi portal, but rather to the language in which the actual content of the LO is provided). Also, selecting "Google Chrome" means that the LOs that are described as addressing that specific ICT skill, will be returned. Obviously, it is fine if an LO is described with more terms than the selected ones. Selecting (or having selected in the profile) a class-node means that any LO that is described with that class-node or with ANY of the child-nodes, will be returned. What is meant here is that the matching is performed with OR for selections within a single top-level class-node. For example, in ICT Skills, the user specifies preference/interest and therefore the child-nodes can be treated with OR (e.g. I am interested in LOs addressing Google Chrome but also in LOs addressing Mozilla Firefox). Selecting multiple leaf-nodes and/or multiple class-nodes, will return LOs that match the selections with OR (for nodes within the same top-level node) and with AND (for nodes across top-level nodes). For example, selecting "Greek", "English" and "WCAG2.0" will return the LOs that are annotated with ("Greek" OR "English") AND ("WCAG2.0"). Another example, selecting "Web browsing software", "Email and security", "Creating and saving documents with MS Word", "Dutch", "Audio" and "Text", will return the LOs that are annotated with (("Chrome", OR "Mozilla" OR "Lynx" OR "IE") OR ("Email services and security" OR "Using the email services")) AND "Dutch" AND ("Audio" OR "Text"). If there exist LOs that match the profile/search-terms at <100% degree, then these are returned but marked as Relevant Search results. In the above example, an LO not defining "audio" or "text" specifically, will be returned as relevant and not 100% result. Same for an LO not defining any language. But if an LO defines English but not Dutch, then it should not be returned at all, as it specifically defines a not match.

An important part of the guided search functionality is the previous annotation of the profile of the user. The user profile creation is addressed in the relevant section, however, we further describe here the meaning of that annotation. As explained, having a pre-annotated profile is expected to help users avoid any manual selection of multiple standard preferences each time they want to find learning objects that match their standard preferences. The annotation of user profiles can be made either by the administrator or by the user himself/herself. This is available in the Profile page, which is accessible after logging-in. The main objective is to define the characteristics that describe the profile and preferences of the user. Selection of a leaf-node means that the user has that characteristic/preference. For example, selection of "Filling-in forms" means that the user has that specific preference in ICT skills. Also, selection of "Spatial ability limitation" means that the user has that specific characteristic. On the other hand, selection of a class-node means that the user has ALL child characteristics/preferences. For example, selection of the "Communication impairment" means

that the user has limited reading ability but also limited speech ability, etc. Also, selection of the “Language” means that the user can understand ALL defined languages. Multiple selection of leaf-nodes or class-nodes is treated with AND. That is, the user has all selected and implied characteristics/preferences.

The results of quick and guided search are presented below the ontology search tree, and are distinguished into “Search results” and “Relevant results”. If the total number of results are more than 10, then a selection “Load more results” appears, which appends the next 10 results etc.

3.7 Connection with Popular Social Networks

Users could publish ViPi platform posts into three widely known social networks: Facebook, Twitter and Google+. In every post page, the following buttons appear:



Figure 16: Connection with social networks

When pressing one of the social network buttons, a login form appears as popup window where the user should enter his/her own credentials. If user has already signed in the same browser session, then the sign-in popup message is skipped and the post appears in the corresponding social network page.

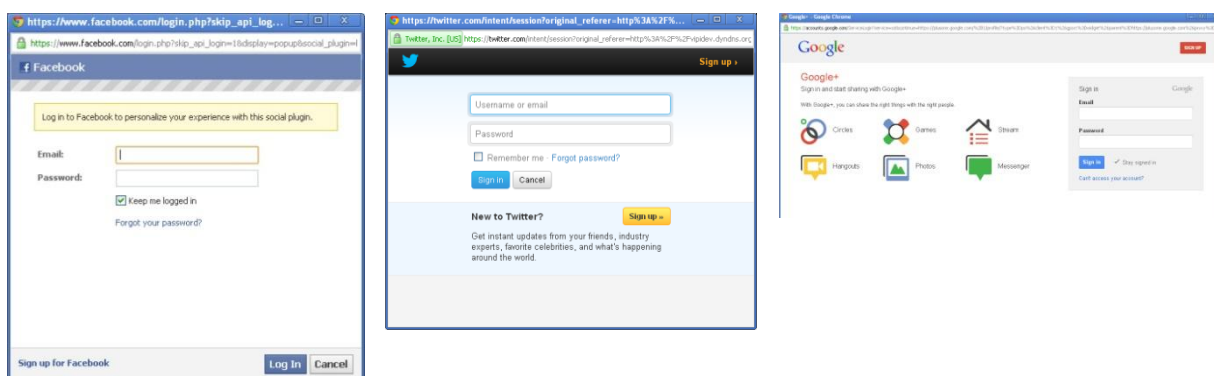


Figure 17: Social networks sign in pop-ups

3.8 Printing Pages/Posts

Each page or post can be printed by pressing the button appearing at the bottom of each page/post.



Figure 18: Printing functionality

Once the print button is pressed, the following pop-up window appears where the user can preview the content that will be printed.



Figure 19: Printing pop-up window

3.9 Using the ViPi Online Training Environment

The core training environment is offered through the ViPi platform, supported by ATutor:

<http://vipi-skills.eu/ATutor/>

This learning management system was designed with accessibility as a priority. A wide range of features ensure assistive technology users can participate fully in learner, instructor, and administrative activities.

Once a user is logged into the ViPi platform, s/he can access also the training environment.

The different roles a user can have in the learning platform are the following:

- Trainees
- Trainers
- Content Providers
- Administrators (ViPi staff)

4 Adopted Technologies & Installation Steps

The following subsections present the components used and the necessary steps that need to be performed in order to properly install and deploy the integrated ViPi platform, along with addressing the issues that may be faced during the installation of these components/plugin-ins.

It is noted that in the following text the word “localhost” should be replaced with the IP of the server where the main back-end platforms (WordPress, ATutor etc.) are going to be installed.

4.1 Installation of Xampp

Xampp is an easy to install **Apache** distribution containing the **MySQL** database management system, the **PHP** web scripting language and the **Perl** programming language. This distribution is used to host the backend, WordPress and ATutor, platforms, as well as the backend repository (MySQL database). The latest (currently 1.7.7) version of Xampp can be downloaded from the URL:

<http://www.apachefriends.org/en/Xampp.html>

Installation of Xaamp can be performed easily by following the wizard steps. Once finished, the MySQL and Apache Servers must be set up and running. It should be noted that Xaamp is only a recommendation due to its user friendly interface, thus any other recent MySQL or Apache implementations can be adopted.

4.2 Installation & Deployment of WordPress

Before proceeding with the installation of WordPress, a new database that will host the WordPress platform shall be created within the MySQL database system, e.g. “WordPress”, coupled with a database user, e.g. “WordPressUser”.

The next step is to download the latest version -currently 3.5.1 is used- of **WordPress** from the URL:

<http://WordPress.org/download/>

In order to install WordPress, the downloaded WordPress compressed file shall be uncompressed into the “**htdocs**” folder of Xampp installation (or into the appropriate folder according to the apache web server used). For setting up the WordPress instance, the file “**wp-config-sample.php**” has to be renamed to “**wp-config.php**” and the following entries have to be edited by using a text editor:

Editing Steps (wp-config.php)
define('DB_NAME', 'database_name_here');
define('DB_USER', 'username_here');
define('DB_PASSWORD', 'password_here');

Table 1: Editing of wp-config.php for first time installation of the WordPress

where, adopting the example details given above, the text 'database_name_here' should be replaced with the name of the database WordPress will be installed (e.g. 'WordPress'), the 'username_here'

the user name that has full access to '*database_name_here*' (e.g. '*WordPressuser*') and the '*password_here*' with the password of the MySQL user.

Finally, in order to create the necessary tables in the database of WordPress, the entries of the URL: <http://localhost/WordPress/wp-admin/install.php> should be properly filled in. Once installation is completed, the portal is ready for usage and configuration.

4.2.1 Enabling Permalinks

Permalinks are the permanent URLs to the individual content entities of WordPress (posts, pages, categories, etc). A permalink is what another webpage will use to link to a page (or section) in the subject WordPress instance, or how a link could be sent in an e-mail message. Thus, it is very essential during the installation procedure to enable permalinks, under **Settings->Permalinks**.

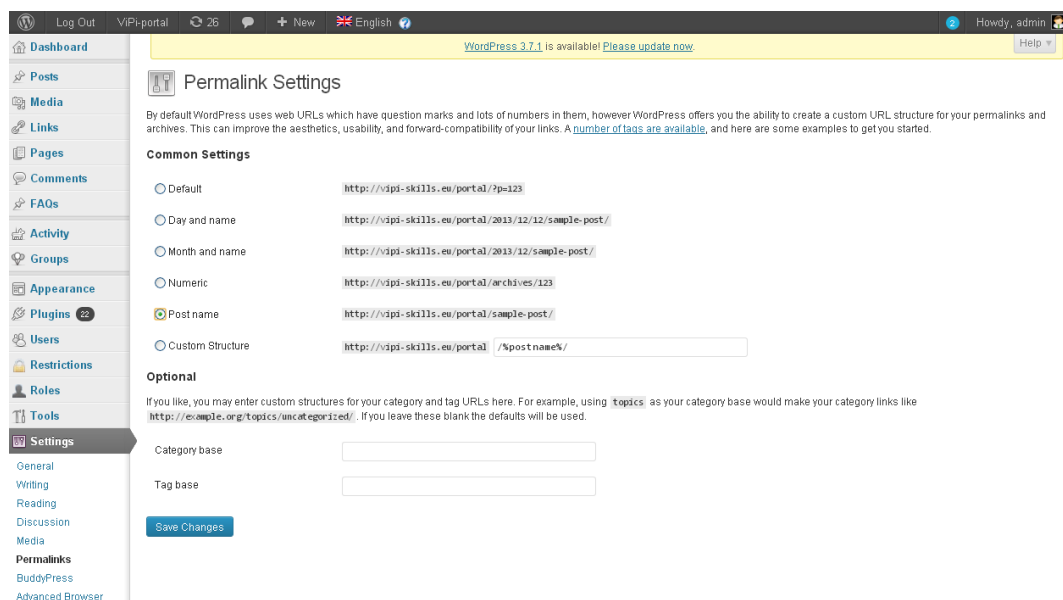


Figure 20: Enabling permalinks during the first time installation of the ViPi platform

The recommended ViPi portal permalink structure is the **“Post name”** field in the above form.

When a user submits a post, its permalink is displayed in the editor and could be used as a reference by other users.



Figure 21: Permalinks in a post message

4.2.2 XML-RPC Services Export & Deployment

One of the core functionalities supported by the WordPress platform is the exposing of several functionalities with XML-RPC services. In order to have several services exposed, the remote

publishing feature should be enabled. In that way, the existing XML-RPC interfaces will be exposed whereas new interfaces could be implemented, allowing for publishing or accessing the ViPi platform data remotely (e.g. from a mobile device).

XML-RPC interfaces could be enabled via the administrator view in the following menu: **Settings->Writing->Remote Publishing**.

The XML-RPC methods currently supported by WordPress could be found in the URL that follows:

http://codex.wordpress.org/XML-RPC_wp

A variety of XML-RPC clients are available (most of them distributed as open source) so far for publishing remotely content in a WordPress installation. These include clients for PC/Mac terminals and mobile phones (Android, iPhone, Blackberry, Symbian, palmOS, J2ME). A list of clients are presented in

http://codex.wordpress.org/Weblog_Client

whereas, the **WordPress for Android** open source project was used to test the WordPress for Android mobile devices:

<http://Android.wordpress.org/>

This library was tested successfully to ensure the native WordPress & BuddyPress XML-RPC services deployment. Finally, it should be noted that the XML-RPC interface has been extended by the ViPi Semantic Content Management (VSCM) plug-in developed within WP4 (see section 5.2.5), in order to expose additional functions to the mobile and other clients via the web and especially fit into the ViPi mobile application needs.

4.3 Installation & Deployment of the BuddyPress Plug-in

BuddyPress is a WordPress plug-in that offers social-networking functionality. More specifically, it supports creating profiles, posting messages, making connections, creating and interacting in groups, etc. The plug-in can be installed through the WordPress dashboard, as illustrated in Figure 22.

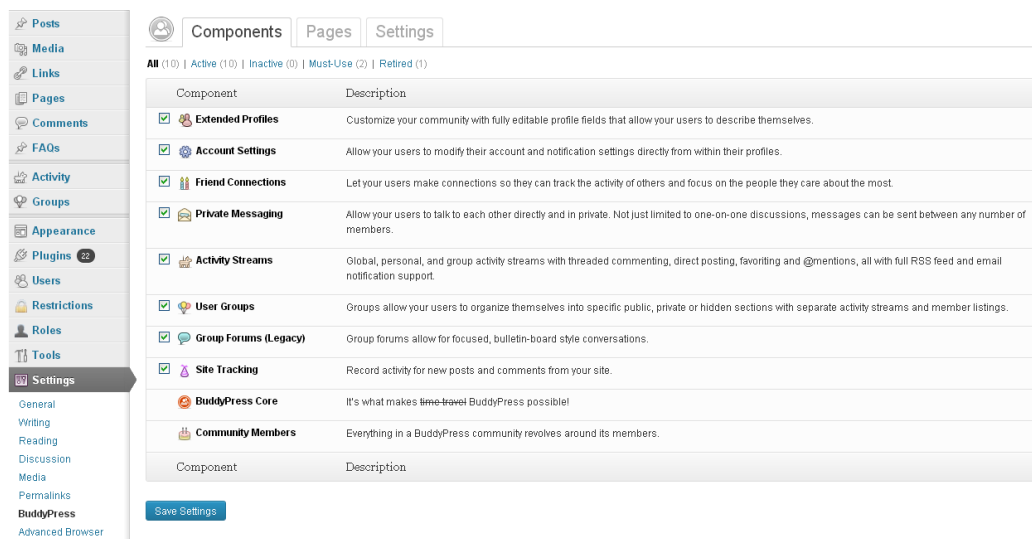


Figure 22: WordPress with BuddyPress as part of the ViPi platform installation procedure

In order to take advantage of the plug-in's full functionality, a BuddyPress compatible theme should be loaded. For this reason the BuddyPress default theme is recommended, but additional themes can be tested and activated as required. The major functionalities that have been installed and tested are the following:

- Fully customizable profile for each user of the ViPi platform

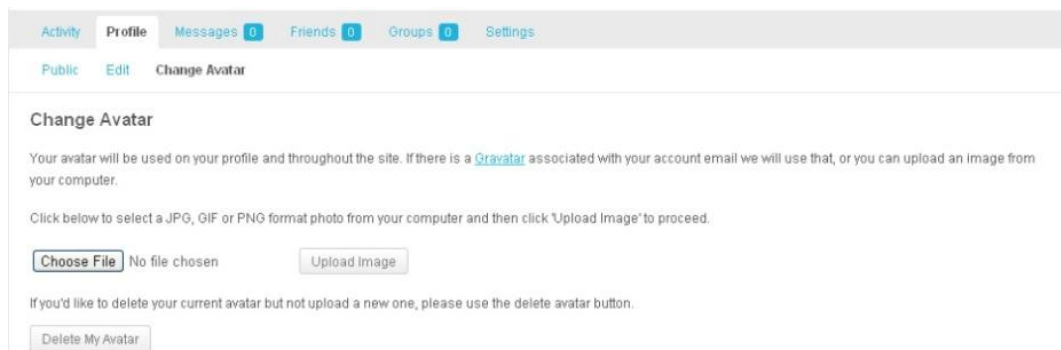


Figure 23: Customization of a registered user, fully synchronized with the ViPi WordPress database

- Private messaging between page users



Figure 24: Functionality of sending/receiving private messages within the ViPi platform

- Friends connections and Group activities

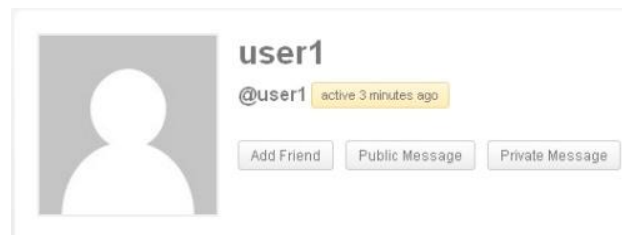


Figure 25: Adding friends and group-based activities within the social networking section of the ViPi platform

It should be noted that **BuddyPress user accounts** are fully synchronized with the **WordPress accounts**. BuddyPress uses its own tables within WordPress database for storing its data, e.g. **wp_bp_groups** table. During the installation procedure, no overlaps or integration issues are expected to be encountered.

4.3.1 BuddyPress XML-RPC Services Export & Testing

The BuddyPress plug-in provides a comprehensive XML-RPC interface, through the **BuddyPress XMLRPC – Receiver** plug-in. This can be downloaded from the following URL:

<http://wordpress.org/extend/plugins/buddypress-xmlrpc-receiver/>

The following functions are currently supported:

- ***bp.verifyConnection***: checks if connection works
- ***bp.getNotifications***: returns notifications like new messages, new friends, followers etc.
- ***bp.updateProfileStatus***: updates the status of a profile
- ***bp.getActivity***: gets various activity stream items
- ***bp.updateExternalBlogPostStatus***: sends an activity stream update field under blogs
- ***bp.deleteExternalBlogPostStatus***: deletes the activity update related to an already posted activity record
- ***bp.getMyFriends***: returns the list of friends
- ***bp.getMyFollowers***: returns the list of followers
- ***bp.getMyFollowing***: returns the list of following
- ***bp.getMyGroups***: returns a list of groups

As already mentioned, the BuddyPress XML-RPC services have been enabled and successfully tested/validated with desktop and mobile clients (see section 4.2.2).

4.4 Installation & Deployment of ATutor

The **ATutor** Learning management tool can be downloaded from the following URL:

<http://ATutor.ca/ATutor/download.php>

The downloaded compressed file should be extracted into the **“htdocs”** folder of Xampp. By pointing a web browser to the URL:

<http://localhost/ATutor>

the user is prompted to follow the procedure to install ATutor, as illustrated in Figure 26 to Figure 28:

ATutor 2.0.3 Installation

ATutor does not appear to be installed. [Continue on to the installation.](#)

Figure 26: First time installation of the ATutor learning platform

ATutor 2.0.3 Installation 

Welcome to the ATutor Installation
 This process will step you through your ATutor installation or upgrade.
 During this process be sure not to use your browser's *Refresh* or *Reload* feature as it may complicate the installation process.
 Before you continue you may want to review the [ATutor handbook](#) for more detailed instructions.

Requirements
 Please review the requirements below before proceeding.

File Integrity	Detected	Status
Case Sensitivity	Ignored	✓
PHP Options		
PHP 4.3.0+	5.3.8	✓
xml	Enabled	✓
mbstring	Enabled	✓
mysql	Enabled	✓
safe_mode = Off	Off	✓
file_uploads = On	On	✓
gd	On	✓
JPEG Support	On	✓
upload_max_filesize >= 2 MB	128M	✓
post_max_size >= 8 MB	8M	✓
session.save_path	0	✓
session.save_path	Directory Writable	✓
curl	Disabled	⚠
include_path	Enabled	⚠
Mail configuration	Disabled	⚠
MySQL Options		
MySQL 4.1.10+	Detected	✓
Found Version mysqlnd 5.0.8-dev - 20102224 - \$Revision: 310735 \$		

[New Installation >>](#) [Install](#)

Figure 27: Successful installation of ATutor as part of the ViPi platform

Following the instructions provided to the installation wizard, the user has to provide specific details concerning the ATutor database storage:

*Database Hostname: Hostname of the database server. Default: localhost	<input type="text" value="localhost"/>
*Database Port: The port to the database server. Default: 3306	<input type="text" value="3306"/>
*Database Username: The username to the database server.	<input type="text" value="wordpress"/>
*Database Password: The password to the database server.	<input type="text" value="wordpress"/>
*Database Name: The name of the database to use. It will be created if it does not exist. Default: atutor	<input type="text" value="atutor"/>
? Table Prefix: The prefix to add to table names to avoid conflicts with existing tables. Default: AT_	<input type="text" value="AT_"/>

Figure 28: Information details for the ATutor backend installation (database installation)

By providing the necessary details for the administrator and the user, the installation is completed successfully.

It was decided to set the installation folder of ATutor inside the WordPress installation folder, to allow the **Duplicator** plug-in, presented later in section 4.5.4, perform backup to ATutor files also. For

the same reason, it is also recommended to use the WordPress database to install the ATutor database (Figure 28).

4.5 Installation of Selected WordPress Plug-ins

The current section provides information about the various available WordPress plug-ins that have been installed and deployed in the ViPi platform to extend its functionality, improve the user experience and increase accessibility of the available features. All the plug-ins included, have been installed and tested and the results are described below.

4.5.1 WP-Post Ratings Plug-in

This plug-in is used for rating posts, pages or comments.

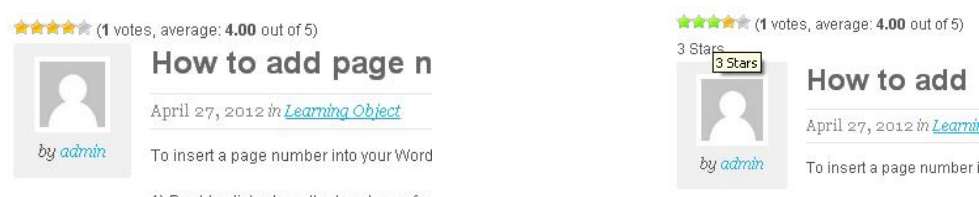


Figure 29: Rating plug-in installed as part of the ViPi platform

The average rating along with the total number of votes is displayed above the post title. To rate a post, the user has to move the mouse above the stars displayed and click on the desired star rating. The only limitations imposed, is that rating is only allowed to posts that are under the category “Learning Objects” and individual users can rate each LO one time only.

Use within the ViPi platform:

- Rating of LOs by users (people with disabilities and/or trainers). This will enable other users to choose specific LOs over others covering the same or similar subject.

4.5.2 Multilingual Support

Multilinguality in ViPi portal is ensured with the use of the **WPML** plug-in. It supports a variety of languages as can be seen in Figure 30 .

Site Languages

These languages are enabled for this site:

English (default)
Dutch
Greek
Lithuanian

<input type="checkbox"/> Chinese (Simplified)	<input type="checkbox"/> Chinese (Traditional)	<input type="checkbox"/> Danish	<input checked="" type="checkbox"/> Dutch	<input checked="" type="checkbox"/> English	<input type="checkbox"/> French
<input type="checkbox"/> German	<input type="checkbox"/> Italian	<input type="checkbox"/> Japanese	<input type="checkbox"/> Russian	<input type="checkbox"/> Spanish	<input type="checkbox"/> Albanian
<input type="checkbox"/> Arabic	<input type="checkbox"/> Armenian	<input type="checkbox"/> Basque	<input type="checkbox"/> Bosnian	<input type="checkbox"/> Bulgarian	<input type="checkbox"/> Catalan
<input type="checkbox"/> Croatian	<input type="checkbox"/> Czech	<input type="checkbox"/> Esperanto	<input type="checkbox"/> Estonian	<input type="checkbox"/> Finnish	<input checked="" type="checkbox"/> Greek
<input type="checkbox"/> Hebrew	<input type="checkbox"/> Hindi	<input type="checkbox"/> Hungarian	<input type="checkbox"/> Icelandic	<input type="checkbox"/> Indonesian	<input type="checkbox"/> Irish
<input type="checkbox"/> Korean	<input type="checkbox"/> Kurdish	<input type="checkbox"/> Latin	<input type="checkbox"/> Latvian	<input checked="" type="checkbox"/> Lithuanian	<input type="checkbox"/> Macedonian
<input type="checkbox"/> Malay	<input type="checkbox"/> Maltese	<input type="checkbox"/> Moldavian	<input type="checkbox"/> Mongolian	<input type="checkbox"/> Nepali	<input type="checkbox"/> Norwegian Bokmål
<input type="checkbox"/> Persian	<input type="checkbox"/> Polish	<input type="checkbox"/> Portuguese (Brazil)	<input type="checkbox"/> Portuguese (Portugal)	<input type="checkbox"/> Punjabi	<input type="checkbox"/> Quechua
<input type="checkbox"/> Romanian	<input type="checkbox"/> Serbian	<input type="checkbox"/> Slovak	<input type="checkbox"/> Slovenian	<input type="checkbox"/> Somali	<input type="checkbox"/> Swedish
<input type="checkbox"/> Tamil	<input type="checkbox"/> Thai	<input type="checkbox"/> Turkish	<input type="checkbox"/> Ukrainian	<input type="checkbox"/> Urdu	<input type="checkbox"/> Uzbek
<input type="checkbox"/> Vietnamese	<input type="checkbox"/> Welsh	<input type="checkbox"/> Yiddish	<input type="checkbox"/> Zulu		

Figure 30: Management of supported languages within the WPML plug-in

Since ViPi portal main languages are English, Dutch, Greek and Lithuanian, some actions have to be performed for configuring this plug-in successfully. English language is by default enabled, while Dutch, Greek and Lithuanian languages can be easily enabled by the menu provided in the above figure.

When the plug-in is installed and properly configured, any post can be written in the available languages and published accordingly. It should be noted that the plug-in does not translate the content automatically, but it is up to the content providers to fill-in the translations needed.


The plug-in is coupled with a widget that enables the end-users to choose the language to be displayed.

Use within the ViPi platform:

- Provide full support of all ViPi main languages, that is, Dutch, English, Greek, and Lithuanian. The initial content of the platform will be translated and populated in advance in all 4 languages. However, it will be the content providers' responsibility to translate any LOs (or other content entities) to be published at later stage.

4.5.3 Facebook, Twitter & Google+ Social Widgets

This plug-in allows for publishing content entities to social networks like **Facebook**, **Twitter** and **Google+**. Publishing content to social networks is allowed only for learning objects.



by admin

Web Browsing Software

21/09/2012 in [Learning Object](#)

★★★★★ 1 vote, 4.00 avg. rating (78% score)

A course on Web browsing software, offered online at ViPi training platform.

You can register for free at the [ViPi portal](#) and access the course at <http://vipi-skills.eu/ATutor/admin/index.php>

Characteristics of this LO

Is suitable for the following age groups: Teenage (10 to 16 years old), Child (0 to 10 years old), Adult (> 16 years old)




Is of type: Course

Addresses the ICT Skills: Web browsing software, Basics for browsers

Targets users with the following impairment type or limitation: Type(s) of impairment or limitation

Is available for payment method: Free

Is available in languages: Dutch, English, Greek

 0
  0
  0


 [Flag this post as inappropriate](#)
 Also available in: [Greek](#)

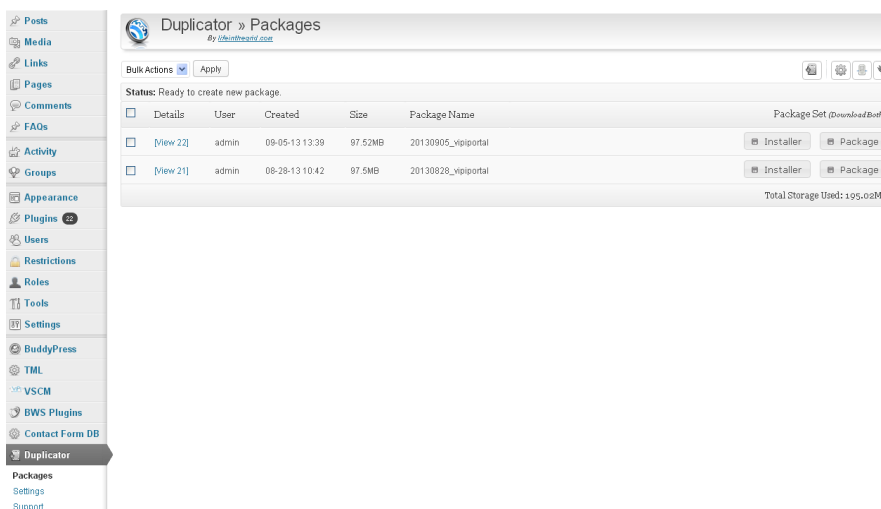
Figure 31: Social network integration in posts

Use within the ViPi platform:

- Publish specific type of content directly on selected social networks upon publication in the ViPi platform.

4.5.4 Duplicator

This plug-in is useful for creating backup and migrating the WordPress installation into another location. Backups can be performed on demand easily from the **Duplicator** menu in the administrator's Dashboard.



Details	User	Created	Size	Package Name	Package Set (Download Both)
[View 22]	admin	09-05-13 13:39	97.52MB	20130905_vipiportal	Installer Package
[View 21]	admin	08-28-13 10:42	97.5MB	20130828_vipiportal	Installer Package

Total Storage Used: 195.02MB

Figure 32: ViPi portal backup settings page

The backup process produces a **.zip** file within the **wp-content/backups** folder. In order to migrate the whole platform into another server, the compressed file must be unzipped into a proper folder. Additionally, the data of the MySQL database must be inserted. This can be performed with a database management tool (like **phpMyAdmin**). The sql file created within this zip, must be imported to enable the database contain all the information of the previous host. Note that **utf-8** should be selected as character set of the sql file during the database import.

If the new database is migrated into a new one with different name, then the appropriate fields in **wp-config.php** should be modified accordingly.

Use within the ViPi platform:

- Supports the portability of the installed back-end components (WordPress, BuddyPress, ATutor etc.).

4.5.5 Access Keys

This plug-in allows adding Access Keys to Category and Page navigation menus to make the website far more accessible. The Access Keys plug-in can be set up from **Posts->Access Keys** menu.

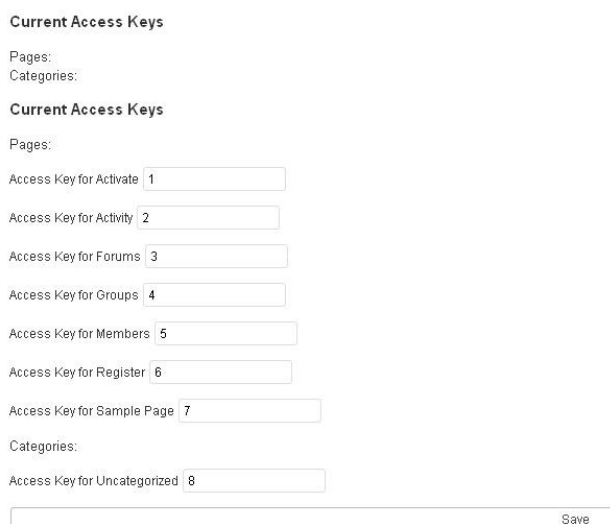


Figure 33: Access Key support for pages contained in the fully integrated ViPi platform

The administrator should assign a number to each access key as shown in Figure 33. Then by pressing Alt+Shift+ a number, the corresponding page will open, according to administrator's settings. The full list of keyboard key combinations for the most popular web browsers can be found here:

<http://anthologyoi.com/WordPress/plugin-ins/WordPress-access-keys.html>

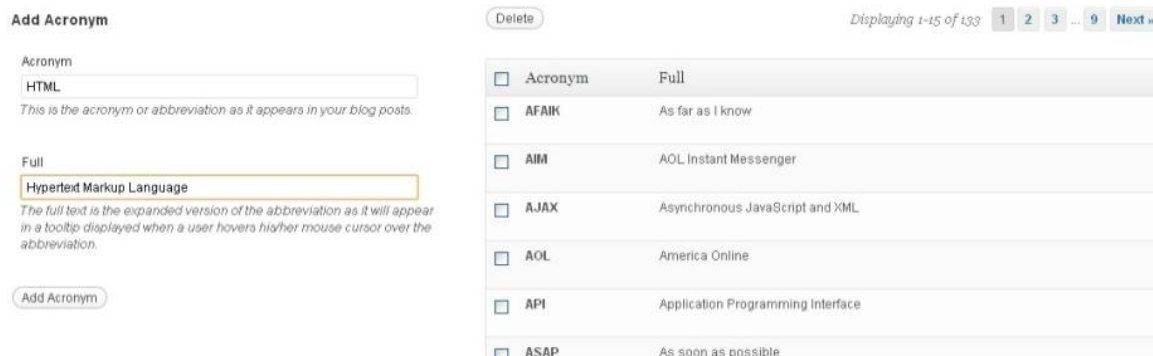
Use within the ViPi platform:

- The plug-in can be used to activate access keys for the main menu items and other main items of the ViPi portal. Its extended use must be avoided in order not to introduce unnecessary complexity for the end-users.

4.5.6 Acronyms

This WordPress plug-in allows maintaining a list of acronyms from within the WordPress management interface. The instances of defined acronyms are automatically replaced in content entities and comments with the full HTML acronym tag, e.g.

`<acronym title="Hypertext Markup Language">HTML</acronym>.`



Add Acronym

Acronym
HTML
This is the acronym or abbreviation as it appears in your blog posts.

Full
Hypertext Markup Language
The full text is the expanded version of the abbreviation as it will appear in a tooltip displayed when a user hovers his/her mouse cursor over the abbreviation.

Add Acronym

Table:

Acronym	Full
<input type="checkbox"/> AFAIK	As far as I know
<input type="checkbox"/> AIM	AOL Instant Messenger
<input type="checkbox"/> AJAX	Asynchronous JavaScript and XML
<input type="checkbox"/> AOL	America Online
<input type="checkbox"/> API	Application Programming Interface
<input type="checkbox"/> ASAP	As soon as possible

Displaying 1-15 of 133 | 1 2 3 ... 9 Next »

Figure 34: Using the acronym plug-in as part of the deployed ViPi platform

Then, if a content entity includes the acronym, a popup will be displayed showing the full text, when a user moves the mouse over the abbreviation.

Use within the ViPi platform:

- The plug-in can be used to enable acronyms for key words/phrases to be used within the ViPi platform. This will improve the user's navigation experience.
- The content providers that create content in specific languages, need to check whether acronyms already exist, and if not, create the necessary ones to use in their content entities. It is noted that the translations of acronyms in different languages, should be treated as separate individual acronyms.

4.5.7 All In One SEO Pack

This plug-in optimizes the WordPress blog for Search Engines (Search Engine Optimization).

Use within the ViPi platform:

- Although the ViPi portal does not have any pure marketing flavour, its appearance in results of search engines is essential to improve its popularity. That is, the users will not necessarily remember the URL to visit the ViPi platform but can use meaningful keywords to discover the portal through search engines.

4.5.8 Print me!

This plug-in adds an icon for allowing printing of a page or post and its comments. This icon is displayed at the bottom of each page and post and popup windows appear when selected to provide the aforementioned functionality.



Figure 35: Adding printing functionality to the ViPi platform

Use within the ViPi platform:

- The print option is enabled for all pages of the ViPi platform.

4.5.9 BuddyPress Moderation

This plug-in adds buttons for reporting posts as inappropriate and gives a convenient way to moderators to view reports. When installed, the following button will appear under each post:



Figure 36: Moderation of posts

All the reports are summarized in the moderation page under BuddyPress menu.

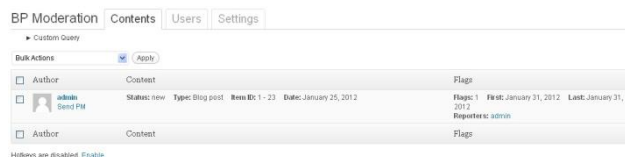


Figure 37: List of reported posts

Use within the ViPi platform:

- The plug-in was activated to provide a way of administering inappropriate content in the web page.

4.5.10 Q and A – FAQ

It allows insertion of FAQs into posts. This can be performed from the dashboard menu of WordPress.



Figure 38: Adding a new FAQ

It also allows the categorization of FAQs, so that the administrator could add the list of FAQs contained in a category to a post.



Figure 39: Managing FAQ categories and FAQs in a post

The insertion of a FAQ in a post could be achieved by using the shortcode **[qa]** into a post, or **[qa cat="nameOfCategory"]** in order to display FAQs of a specific category.

Use within the ViPi platform:

- The plug-in was activated within the ViPi platform to provide useful help to users by answering their most common questions through a dedicated portal page. The questions and answers will be grouped in proper categories to improve the end-user navigation experience.

4.5.11 Fast Secure Contact Form

This plug-in allows the administrator to easily create and add contact forms to WordPress. The contact form allows users to send emails to a site's administrator, and also send a meeting request to talk over phone or video. An administration panel is present, where the webmaster can create and preview unlimited forms. In order to embed a contact form within a page or post, the creator needs to use the shortcode **[si-contact-form form='NumberOfForm']**, where **NumberOfForm** is the id of the form defined in the plug-in's settings page.

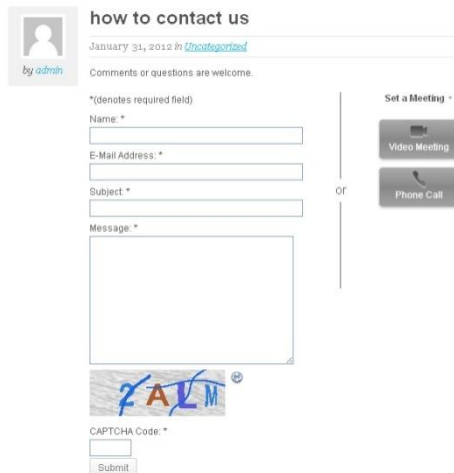



Figure 40: Example of a contact form in a post.

Use within the ViPi platform:

- The plug-in was activated and specific contact form was created to provide an easy interface for end-users to contact the site's administrator and any other responsible person that will be deemed necessary. Additionally, the plug-in is providing a Learning Object submission form for Content Providers, which is then forwarded to the portal Administrator for validation and approval.

4.5.12 Contact Form DB

This plug-in stores all content submitted through the Fast Secure Contact forms of the site in the internal WordPress database, thus ensuring that no submission will be lost. The plug-in is only visible through the administration panel, which provides also the functionality to export the submitted data into various formats.



Submitted	from_name	from_email	ex_field1	ex_field2	ex_field3	ex_field4	ex_field5	ex_field6
2013-11-12 13:31:14 +02:00	Andy Burton	andy.burton@ntu.ac.uk	Skills Workshop ICT training resources	* About computers * Browsing the web * Communicate	Links to ICT Training resources	http://www.skillsworkshop.org/ict	Communicate impairment * Hearing impairment *	* Microsoft
2013-06-14 14:11:24 +03:00	andy	andy.burton@ntu.ac.uk	BBC Accessibility Guide	* About computers * Assistive technologies * Computer input	A site presenting accessible information to help	A mine of information to help when a user fits into any of the following categories: I can't see very well I am blind I can't hear very well	Communicate impairment * Hearing impairment *	* Microsoft * Microsoft * Microsoft
2013-06-14 14:03:26 +03:00	Andy Burton	andy.burton@ntu.ac.uk	Data and Information (BBC KS3 Bitesize)	* About computers * Browsing the web * Managing data and	An English Language course on Managing information	Contents of the course... Course revision material, activities and tests to learn about Data and Information. Sub topics include: Managing information Saving your work	Communicate impairment * Hearing impairment *	* Microsoft * Microsoft * Microsoft
2013-05-10 14:00:02 +03:00	andy	andy.burton@ntu.ac.uk	test2	* About computers	test2	test2 more	Upper limb impairment	* Microsoft
2013-05-10 14:00:02 +03:00	andy	andy.burton@ntu.ac.uk	Fly Swat Game	* About computers * Computer input and devices * Using	A game to experience practice	Fly swat has been developed to meet some of the ICT teaching needs for users with multiple and profound disabilities. The ICT syllabus for people with these needs is broken down into a set of assessable skills for students who are not yet achieving the	Learning difficulties * Mobility limitation	* Microsoft * Microsoft

Figure 41: The FS Contact form DB administration panel

Use within the ViPi platform:

- The plug-in is used for storing Learning Object submissions into the internal portal database.

4.5.13 Blog-in-Blog

This plug-in shows posts from a WordPress category on a page using shortcodes. For example, the “Learning Objects” page displays all posts under the category “Learning Object”, by using the following shortcode:

[blog_in_blog category_slug='lo' num=5]

where “lo” is the slug of the category “Learning object”.

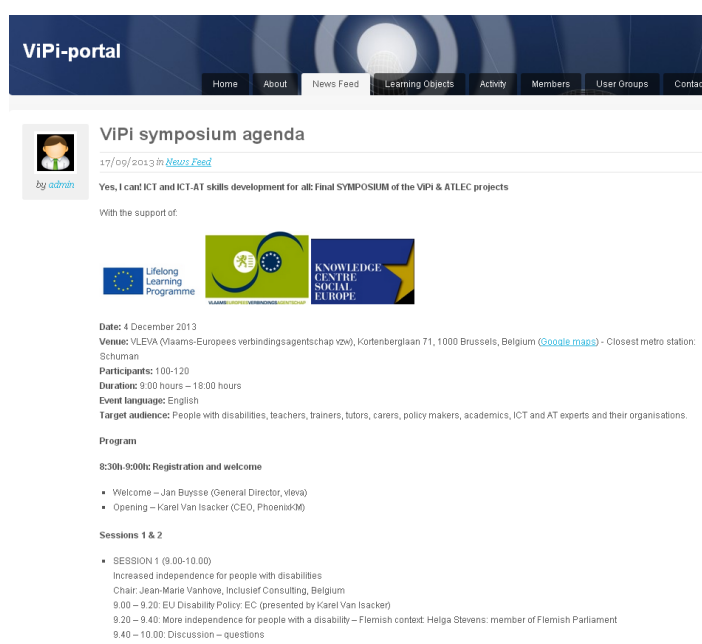


Figure 42: Example of a page that consists of specific category posts

Use within the ViPi platform:

- The plug-in is used to display posts of specific category within a WordPress page.

4.5.14 Require Post Category

This plug-in ensures that every submitted post has an annotated category. If no category is selected when publishing a new post, then the following message will appear, prohibiting user to upload post without selecting the category where it belongs.

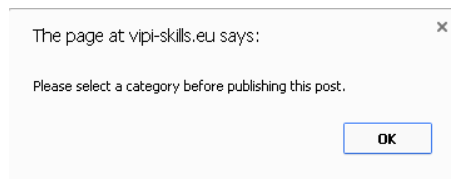


Figure 43: Empty category selection popup when publishing posts

Use within the ViPi platform:

- The plug-in is used to prohibit publishing posts that are not contained in any category.

4.5.15 Search by Category

This plug-in provides the ability to search in posts of a specific category.

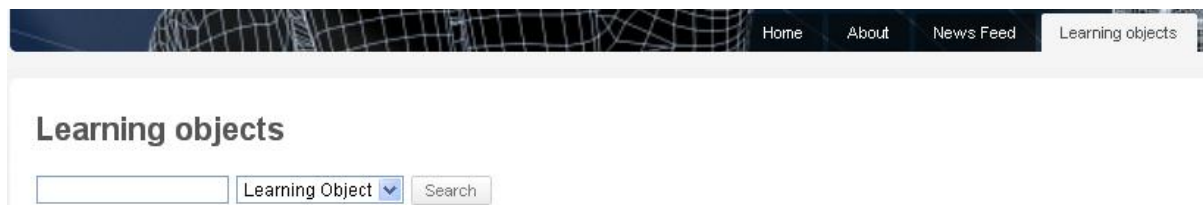


Figure 44: Searching of learning objects in a page

Use within the ViPi platform:

- The plug-in was activated to provide the search functionality within the “Learning Objects” category posts.

4.5.16 Advanced Browser Check

This plug-in checks the site visitor’s browser, and displays a proper message if the browser version is not supported by the ViPi portal.

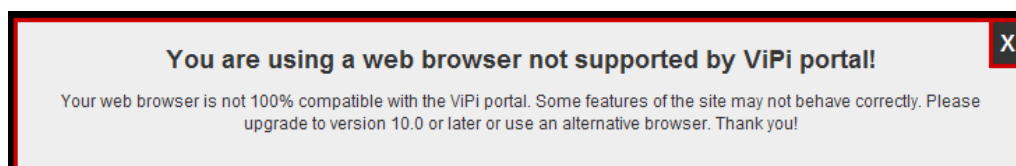


Figure 45: Popup message for unsupported browser version

Use within the ViPi platform:

- The plug-in was activated to inform Internet Explorer users that ViPi platform does not support versions earlier than 10.0.

4.5.17 Disable Admin Bar

This plug-in removes the administrator bar of WordPress.

Use within the ViPi platform:

- The plug-in is used for removing the administrator bar of ViPi portal.

4.5.18 Force Post Title

This plug-in forces WordPress users to assign a title to a post before publishing it.



Figure 46: The popup message that appears when a post is submitted with an empty title

Use within the ViPi platform:

- The plug-in is preventing posts to be submitted with empty title in ViPi portal.

4.5.19 Image Widget

This plug-in provides the functionality of adding image widgets to a WordPress site.



Figure 47: An example of Image Widget

Use within the ViPi platform:

- The plug-in is used for the display of the Funding widget.

4.5.20 WP User FrontEnd

This plug-in provides the functionality of editing the user profile from WordPress front-end. For this reason, a dedicated page was created, where the logged in user may alter the corresponding profile settings.

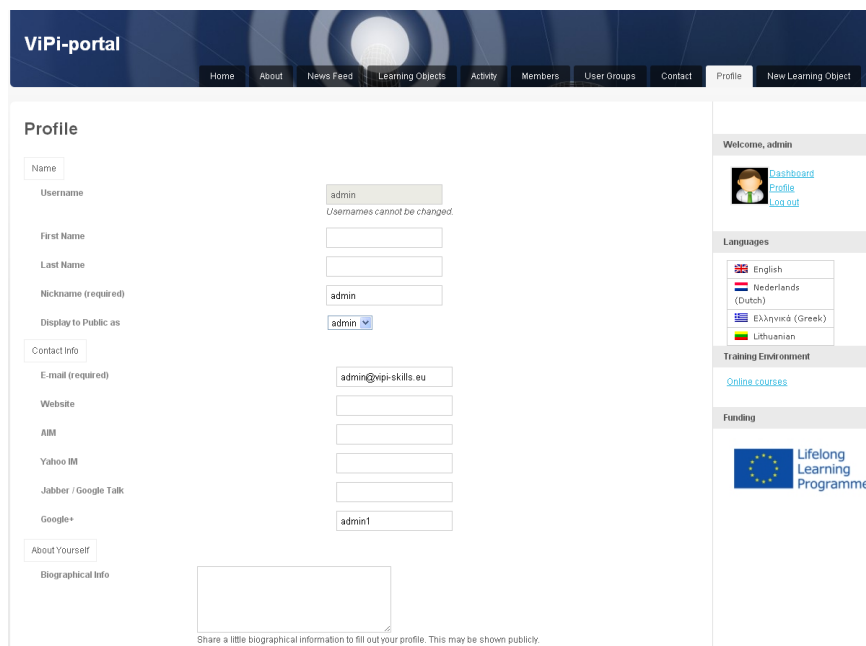


Figure 48: The front-end user profile editor

Use within the ViPi platform:

- The plug-in is used for editing the profile through the ViPi front-end, thus providing a unified and friendly interface to the portal user.

4.5.21 Theme My Login

This plug-in provides a front-end solution for user login, logout, register and forgot password functionalities.

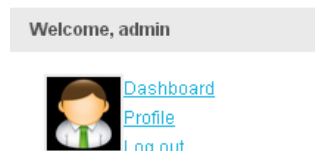


Figure 49: Theme My Login widget

Use within the ViPi platform:

- The plug-in provides user account interaction through the ViPi front-end.

4.5.22 Remove Dashboard Access

This plug-in disables the dashboard access of certain users based on their role.

Use within the ViPi platform:

- The plug-in is used for prohibiting the dashboard (WordPress back-end) access to all users except Administrator.

4.5.23 Role Scoper

This plug-in provides restrictions and scopes over the WordPress user roles.

Use within the ViPi platform:

- The plug-in is used for assigning restrictions on certain pages for “Trainer” and “Trainee” user roles.

4.5.24 WP Accessibility

This plug-in provides options to improve accessibility of a WordPress site, including title attributes.

WP Accessibility: Settings

Remove Title Attributes

Remove title attributes from:

- ☒ Nav menus
- ☒ Page lists
- ☒ Category lists
- ☒ Archive links
- ☒ Tag clouds
- ☒ Category links
- ☒ Post edit links
- ☒ Edit comment links

[Update Title Attribute Settings](#)

Add Skiplinks

Configure Skiplinks

- ☐ Enable Skiplinks
- ☐ Skiplinks always visible

Skip to Content link target (ID of your main content container)

Skip to Navigation link target (ID of your main navigation container)

Site Map link target (URL for your site map)

Add your own link (link or container ID) Link text for your link

Styles for Skiplinks when they have focus

left: 1em; top: 1em; background: #B3D9FA; color: #000; padding: 5px; border: 1px solid #357; box-shadow: 2px 2px #777; border-radius: 5px;

Styles for Skiplinks without focus (not currently visible)

[Update Skiplink Settings](#)

Figure 50: An example of accessibility settings

Use within the ViPi platform:

- The plug-in improves accessibility of the ViPi portal to impaired people.

5 New Components Implemented inside ViPi

This section describes the components that have been designed and implemented in the framework of the project, to be part of the fully integrated ViPi platform. Any WordPress plug-ins that are going to be developed inside ViPi, are planned to be submitted to the **WordPress Plug-in Repository** as contribution of the ViPi consortium to the community.

The plug-ins will initially clarify the WordPress version(s) that they support. The strategy to be adopted by the ViPi consortium for the continuous support and maintenance of the plug-ins and the compatibility with newer WordPress versions, will be described in the exploitation plan of the ViPi project. In any case, since the plug-ins are made available to the WordPress community, support of newer versions can be offered by the WordPress community as well.

5.1 WordPress & ATutor Seamless Access

This implementation aims to support the necessary **single-sign-on** capability among the main ViPi platform components (WordPress, BuddyPress & ATutor). Once a user is logged in the ViPi platform through e.g. the WordPress portal, he/she is then able to access the associated social networking content (BuddyPress front-end) and also navigate in the ATutor eLearning environment without having to login again.

The single-sign-on implementation takes advantage of the unified access to the WordPress & ATutor database. Then, by overriding the login feature of the ViPi portal (WordPress back-end), the login to the eLearning environment (ATutor back-end) is performed in a transparent way and vice versa. The same applies also for logging-out, where a **single-sign-out** functionality has been implemented.

5.2 ViPi Semantic Content Management (VSCM) plug-in

The **ViPi Semantic Content Management (VSCM)** is a new WordPress plug-in implemented in the framework of the ViPi project, that enables the semantic annotation and discovery of LOs and user profiles within the ViPi platform, allows the implementation of semantic relations among these and advances the quality of navigation and search experience in the available LOs.

The plug-in implements the VSCM module approach adopted inside ViPi and especially during the WP4 development, which is described in detail in section 0. Additionally, the VSCM plug-in takes advantage of the corresponding VSCM ontology created inside WP4 that is also detailed in section 0.

The VSCM plug-in consists of the components presented in the following sections.

5.2.1 Ontology Management

The VSCM plug-in utilises the **Powl** framework and its underlying **RDF-API for PHP** for managing persistent ontologies. These frameworks can be found at the following addresses:

<http://sourceforge.net/projects/powl/>

<http://www4.wiwiss.fu-berlin.de/bizer/rdfapi/>

Both frameworks provide persistent management of ontologies in MySQL databases. For this reason, once the VSCM plug-in is installed for the first time, the following tables are created automatically in the WordPress database table:

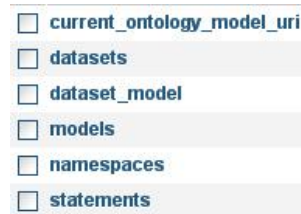


Figure 51: MySQL tables for ontology management

These tables are used for storing the ontology models, namespaces, datasets and RDF triples that ontology is transformed to. It should be noted that these APIs have been properly adjusted to support UTF-8 text for storing the Greek and Lithuanian related ontology concepts. Additionally, querying within ontology is performed with the build in mechanism provided from these frameworks.

5.2.2 Administration Component

This component adds administration functionality within the WordPress backend platform, as shown in Figure 52.

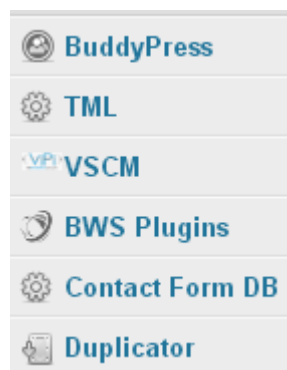


Figure 52: Plug-in preview within ViPi administrative view

The administration options are displayed in the VSCM menu of the administration panel that provides the necessary VSCM ontology management functionality

VSCM

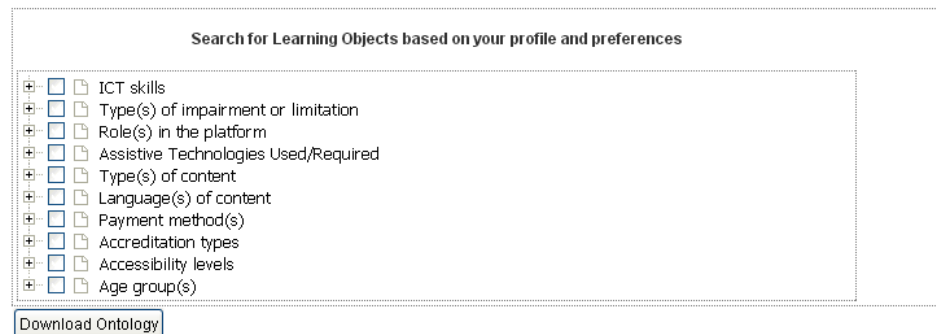


Figure 53: Ontology management in the administrator dashboard

When the plug-in is first time installed, the default ViPi ontology is automatically uploaded to the ontology repository. Thus, if the administrator navigates through the VSCM menu panel, he/she will be able to view all ontology classes/instances and its annotated Learning Objects. It should be noted that, instead of the real ontology concept name, a user-friendly text is displayed for each concept to assist the navigation procedure, translated into the user's selected language (current supported languages are English, Dutch, Greek and Lithuanian).

By moving the mouse cursor over a class/instance, the full URI of the class appears as a tool tip.

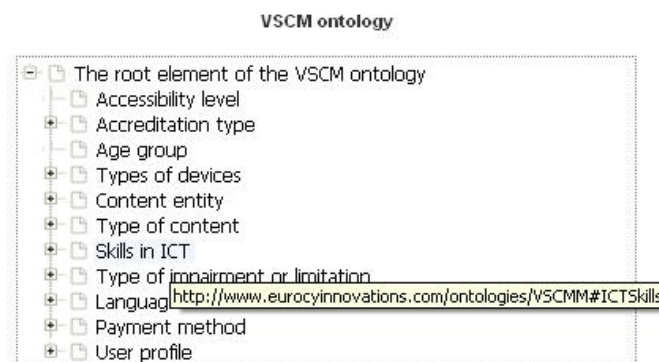
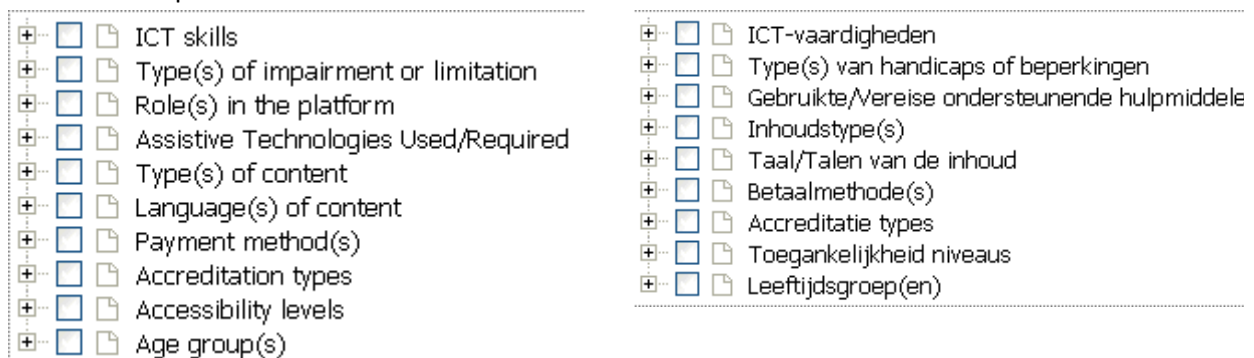


Figure 54: The default ViPi ontology tree structure and tooltip functionality

Switching to different languages can be performed by the default language selection of the administrator panel.



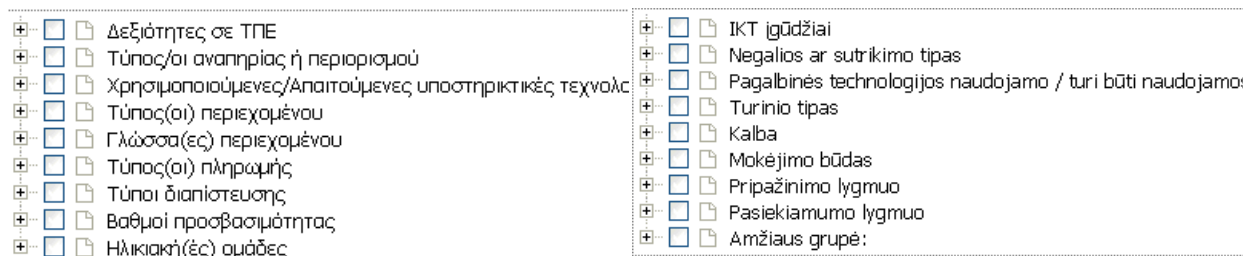


Figure 55: The default ViPi ontology in English, Dutch, Greek and Lithuanian languages respectively

Moreover, the button “**Download Ontology**” downloads the whole ontology from the database into a RDF file, named “**VSCM_Ontology.rdf**”.

5.2.3 Annotation of Learning Objects (LOs)

Annotation of LOs is an essential part of the ViPi portal and can be performed during the publish of a new post belonging to the “Learning Object” category.

For this reason, the metabox containing the LO annotation functionality appears only when the “Learning Object” category is selected in the appropriate field. More details can be found in section 3.5.

5.2.4 Annotation of User Profiles

Annotation of user profiles can be performed for each user registered in the ViPi platform. For this reason, a new entry on the WordPress profile page has been added, named “**Semantic profile annotation**”. Further details on the annotation procedure are presented in sections 3.2 and 3.6.2.

5.2.5 Quick and Guided Search

The quick and guided search component provides a widget that allows navigation through the ontology. The component functionality was presented in details in section 3.6.

5.2.6 XML-RPC Interface

The VSCM plug-in also exports its functionality, by extending the WordPress XML-RPC interface, to enable access to ViPi portal content by the ViPi mobile application. For this purpose, the following functions have been developed & deployed.

XML-RPC Function	<i>String</i> vscm.getOntology (<i>boolean forLOAnnotation</i> , <i>boolean forUserAnnotation</i> , <i>String lang</i> , <i>boolean showLO</i> , <i>String username</i> , <i>String password</i>)
Functionality	Returns the VSCM ontology together with the annotations (classes and instances) of the posted LOs in JSON format.
Arguments	<p>forLOAnnotation: A <i>Boolean</i> parameter indicating whether the returned ontology will include the classes/instances where the property value</p> <p>http://www.eurocyinnovations.com/ontologies/VSCM#forLOAnnotation</p>



	<p>is set to <i>true</i> (Actually includes the classes/instances used for the annotation of the LOs).</p> <p>forUserAnnotation: A <i>Boolean</i> parameter indicating whether the returned ontology will include the classes/instances where the property value</p> <p>http://www.eurocyinnovations.com/ontologies/VSCM#forUserAnnotation</p> <p>is set to <i>true</i> (Actually includes the classes/instances used for the annotation of user profiles).</p> <p>lang: The language in which the ontology will be returned. Currently, available options are: “en” for English, “nl” for Dutch, “el” for Greek and “lt” for Lithuanian.</p> <p>showLO: A <i>Boolean</i> parameter indicating whether the returned <i>String</i> should contain the annotated LOs or not.</p> <p>username: The username of the user that invokes the function. It is used for authentication purposes.</p> <p>password: The password of the user that invokes the function. It is used for authentication purposes.</p>
Return Value	<p>A <i>String</i> that contains the desired ontology in JSON format. A sample of the returned JSON format text follows:</p> <pre>[{ "title": "Skills in ICT", "uri": "http://www.eurocyinnovations.com/ontologies/VSCM#ICTskills", "LOs": "", "children": [{ "title": "About computers", "uri": "http://www.eurocyinnovations.com/ontologies/VSCM#AboutComputers", "LOs": "134, 214", ... </pre> <p>In the above example, the <i>showLO</i> argument was set to <i>true</i> to also return the annotations of the posted LOs. This sample indicates that the http://www.eurocyinnovations.com/ontologies/VSCM#AboutComputers class was used to annotate the LO posts with IDs 134 & 214. In cases where <i>showLO</i> is set to <i>false</i> the “LOs”: ... ” part is omitted in the returned result.</p>

XML-RPC Function	<i>String</i> vscm.searchLO (<i>String</i> typeOfSearch, <i>String</i> userSelection, <i>boolean</i> considerProfile, <i>String</i> lang, <i>String</i> username, <i>String</i> password)
Functionality	Performs search in the LO pages.
Parameters	<p>typeOfSearch: A <i>String</i> including the type of search. Currently, available options are "guided" and "quick".</p> <p>userSelection: In "quick" search, it includes the inserted keywords. In "guided" search, it contains the selected ontology concepts separated by commas. For instance, if the user selects to search for</p> <p>http://www.eurocyinnovations.com/ontologies/VSCMM#LearningObject</p> <p>and</p> <p>http://www.eurocyinnovations.com/ontologies/VSCMM#WCAG2.0</p> <p>then the <i>String</i> should be "LearningObject,WCAG2.0" (the "http://www.eurocyinnovations.com/ontologies/VSCMM#" prefix is omitted).</p> <p>Please note that if <i>userSelection</i> needs to be an empty <i>String</i>, then the "=" character should be used as parameter value to invoke the function properly.</p> <p>considerProfile: <i>true</i> or <i>false</i> according to user's selection to consider their profile or not.</p> <p>lang: The language in which the search of LOs will be performed. Currently, available options are: "en" for English, "nl" for Dutch, "el" for Greek and "lt" for Lithuanian.</p> <p>username: The username of the user that invokes the function. It is used for authentication purposes.</p> <p>password: The password of the user that invokes the function. It is used for authentication purposes.</p>
Return Value	<p>A list of <i>String</i> values, containing the IDs of the LO posts matching the query provided.</p> <p>In "guided" search results, the <i>String</i> values that include the "==" prefix refer to relevant results (<100% matching).</p>

Table 2: XML-RPC interface of the VSCM plug-in

5.2.7 Maintenance & Exploitation

As was perceived in the previous sections, during the developments and customizations performed in the ViPi portal, the added services and functionality has been developed in the form of additional plug-ins, such as to eliminate (or at least minimise) the need for code modifications of the core WordPress platform and/or of third-party plug-ins. This not only benefits the ViPi developments by allowing portability and integration with several side services and functionalities, but also serves as input to the WordPress community and increases the exploitability.

A critical component of the VSCM plug-in developed as part of the ViPi project, is the VSCM ontology (see section 0). This is the structured knowledge model used by the plug-in, and subsequently by the ViPi platform, to semantically describe the users' profiles, as well as the learning objects and then be able to perform automatic reasoning and retrieve semantically-rich matches in the Learning Objects Guided Search functionality. The version of the plug-in that is released together with the ViPi platform and that has been developed as part of the ViPi project, has a built-in ontology, fully able to support the current platform implementations and knowledge modelling requirements. Therefore, an update of this ontology is not foreseen any time very soon.

After the end of the ViPi project, the source code for this plug-in, together with the rest of the ViPi platform components, will be released as open source software, under an appropriate licensing scheme. This will allow the ViPi developers to cooperate with other developers and build a community around this plug-in. This community of developers is expected to undertake the future updates of the plug-in, under the leadership of Hypertech S.A and G.M EuroCy Innovations Ltd, the initial developers of the plug-in on behalf of the ViPi consortium. The plug-in is expected to evolve, not only to support new functionality required (or desired) by the ViPi platform, but also to become more generic and exploitable in other application domains. After all, it offers a semantic matching and searching engine, the content of which depends on the ontology included.

The above show that one way or another, the ontology bound to the VSCM plug-in will need to be updated. This process will be facilitated by a dedicated back-end administration functionality implemented and integrated within the VSCM plug-in, with a wide range of functionalities such as the following, in order to increase its possibilities of reaching a wider audience and become popular and useful to the WordPress community (see section 0 for more details):

- Capability to maintain several ontologies, always defining one of them as the default (currently used)
- Option to upload new ontology, replace or delete an existing one
- Avoid any loss of information with evolution of the ontology, after changing the default ontology or performing structural changes, i.e. provide the administrator with the ability to perform a mapping between old and new ontology concepts. The mapping will be mandatory and facilitated by a dedicated tool. Therefore, the VSCM plug-in is expected to also provide an interface through which the administrator will upload the mapping and run the process to update the stored annotations accordingly.

It should be noted however, that these functionalities are already performed by the technical team of ViPi manually, to support the ontology updates deemed necessary at the project duration.

Annex I: ViPi Semantic Content Management (VSCM) Module

As already mentioned the VSCM module enables the semantic annotation and discovery of LOs and user profiles within the ViPi platform and allows the implementation of semantic relations among these and advances the quality of navigation in the available LOs. The detailed VSCM design approach initially suggested inside ViPi and especially during the WP4 development, which actually lead to the creation of the VSCM plug-in (see section 5.2), is described in a dedicated document delivered as an attachment to the current deliverable.

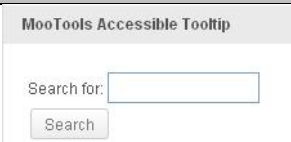
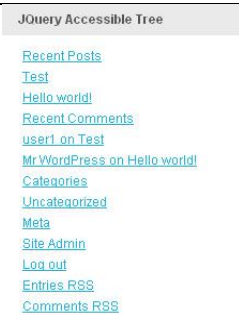

Annex II: ViPi Semantic Content Management (VSCM) Ontology

The design of the VSCM module led to the creation of the corresponding VSCM ontology created inside WP4, which is also presented in a dedicated document delivered as an attachment to the current document.

Annex III: Evaluation of Accessibility Plug-ins




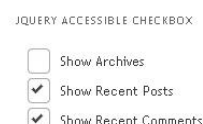
As part of the development procedure, several plug-ins enabling accessibility and that were available in the WordPress plug-in Repository have been tested (mostly those developed within **Aegis & Accessible** FP7-ICT projects). The main outcome of this evaluation is that most of the plug-ins are outdated and efforts should be made to get them work properly within the integrated ViPi platform. The evaluation results are provided below.




In order to ensure necessity and proper functionality, extensive tests will be carried out during “WP6: Testing and Community building” & “WP7: Piloting and Community building” implementation that would lead to the final selection of the plug-ins to be included in the integrated ViPi platform.

Plug-in Name	Description / URL	Screenshot	Comments / issues
MooTools Accessible Tooltip	An extension of the default search plug-in that uses the MooTools WAI-ARIA enabled tooltip (http://wordpress.org/extend/plugins/mootools-accessible-tooltip/)		-
JQuery Accessible Tree	A tree of a blog's recent posts, recent comments, categories and meta-info, that uses the JQuery WAI-ARIA enabled tree (http://wordpress.org/extend/plugins/jquery-accessible-tree/)		The layout is not displayed in a Tree like structure
JQuery Accessible Tooltip	An extension of the default search plug-in that uses the JQuery WAI-ARIA enabled tooltip (http://wordpress.org/extend/plugins/jquery-accessible-tooltip/)		-

Plug-in Name	Description / URL	Screenshot	Comments / issues
JQuery Accessible Tabs	A tab of the blog's recent posts, recent comments and archives that uses the JQuery WAI-ARIA enabled tab (http://wordpress.org/extend/plugins/jquery-accessible-tabs/)		The widget of this plug-in should be placed in the first footer widget area
JQuery Accessible Slider	A slider controlling the number of the blog's recent posts, that uses the JQuery WAI-ARIA enabled accordion and slider (http://wordpress.org/extend/plugins/jquery-accessible-slider/)		This plug-in does not work properly
JQuery Accessible Progressbar	A progress-bar showing up when the button group controlling the number of a blog's recent posts, recent comments and archives shown in an accordion is activated. It uses the JQuery WAI-ARIA enabled progress bar, accordion and button. (http://wordpress.org/extend/plugins/jquery-accessible-progressbar/)		This plug-in does not work properly
JQuery Accessible Menu	A menu bar of the blog's recent posts, recent comments, categories and meta. It uses the JQuery WAI-ARIA enabled menu bar. It's used in a wide widget area, like the "First Footer Widget Area" of the "Twenty Ten" theme (http://wordpress.org/extend/plugins/jquery-accessible-menu/)		Some visual improvements could be applied. It does not work on Google Chrome and auto complete feature does not

Plug-in Name	Description / URL	Screenshot	Comments / issues
			work properly
JQuery Accessible Dialog	<p>A dialog to add new tweets in a blog. It uses the JQuery WAI-ARIA enabled dialog.</p> <p>(http://wordpress.org/extend/plugins/jquery-accessible-dialog/)</p>		-
JQuery Accessible Carousel	<p>A carousel of the images in folder "wp-content/plugin-ins/JQueryAccessibleCarousel/lib/images" that uses the JQuery WAI-ARIA enabled carousel. All images (.jpg or .png or .gif) in the "wp-content/plugin-ins/JQueryAccessibleCarousel/lib/images" folder are first resized to 150x150 px in order to be used by the plug-in. Each image file name is used as the image's alt text (underscores "_" are replaced by spaces " ")</p> <p>(http://wordpress.org/extend/plugins/jquery-accessible-carousel/)</p>		This plug-in works smoothly but some visual adjustments have to be done in order to be compatible with the page theme
JQuery Accessible Autocomplete	<p>An extension to the default search plug-in that uses the JQuery WAI-ARIA enabled autocomplete</p> <p>(http://wordpress.org/extend/plugins/jquery-accessible-autocomplete/)</p>		-

Plug-in Name	Description / URL	Screenshot	Comments / issues
Fluid Accessible Sorting Grid	A sorting grid of the blog's recent posts, recent comments, categories and meta that uses the Fluid WAI-ARIA enabled sorting list (http://wordpress.org/extend/plugins/fluid-accessible-sorting-grid/)		The widget of this plug-in should be placed in the first footer widget area
Fluid Accessible Sorting List	A sorting list of the blog's recent posts, recent comments, categories and meta and uses the Fluid WAI-ARIA enabled sorting list (http://wordpress.org/extend/plugins/fluid-accessible-sorting-list/)		-
MooTools Accessible Tabpanel	A panel of tabs of the blog's recent posts, recent comments and archives that uses the MooTools WAI-ARIA enabled tabpanel (http://wordpress.org/extend/plugins/mootools-accessible-tabpanel/)		The widget of this plug-in should be placed in the first footer widget area
JQuery Accessible Checkbox	A checkbox controlling the number of the blog's recent posts, recent comments and archives shown in an accordion that uses the JQuery WAI-ARIA enabled accordion and checkbox (http://wordpress.org/extend/plugins/jquery-accessible-checkbox/)		This plug-in does not work properly

Plug-in Name	Description / URL	Screenshot	Comments / issues
MooTools Accessible Accordion	An accordion of the blog's recent posts, comments and archives that uses the MooTools WAI-ARIA enabled accordion (http://wordpress.org/extend/plugins/mootools-accessible-accordion/)		-
Fluid Accessible Progressbar	A progress-bar showing up when the button group controlling the number of the blog's recent posts, comments and categories shown in an accordion is activated and uses the Fluid WAI-ARIA enabled progress bar and the JQuery WAI-ARIA enabled accordion and button (http://wordpress.org/extend/plugins/fluid-accessible-progressbar/)		The widget of this plug-in should be placed in the first footer widget area
Fluid Accessible Sorting Portlet	(http://wordpress.org/extend/plugins/fluid-accessible-sorting-portlet/)	-	This plug-in could not be activated within WordPress (not valid header)
JQuery Accessible Accordion	An accordion of the blog's recent posts, comments and archives that uses the JQuery WAI-ARIA enabled accordion (http://wordpress.org/extend/plugins/jquery-accessible-accordion/)		-




Plug-in Name	Description / URL	Screenshot	Comments / issues
JQuery Accessible Button	WAI-ARIA enabled button plug-in for WordPress (http://wordpress.org/extend/plugins/jquery-accessible-button/)		This plug-in does not work properly

Table 3: Accessibility plug-ins tested inside the integrated ViPi platform