From LMS to PLE: a Step Forward through OpenSocial Apps in Moodle

Evgeny Bogdanov[1], Carsten Ullrich[2], Erik Isaksson[3],
Matthias Palmer[3], and Denis Gillet[1]

[1] Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland
evgeny.bogdanov@epfl.ch, denis.gillet@epfl.ch
[2] Shanghai Jiao Tong University, 200030 Shanghai, China
ullrich_c@sjtu.edu.cn
[3] Uppsala University, Box 256, 751 05 Uppsala, Sweden
erikis@kth.se, matthias@kth.se

Abstract. Bringing flexibility and extensibility into Learning Management Systems) is crucial because it gives teachers and students a free choice of technologies and educational materials they want to use for their courses. The paper presents a solution via enabling widgets (OpenSocial apps) within Moodle. Our first Moodle plug-in allows teachers to freely choose a set of tools they want to use in their courses though students can not change widgets proposed by teachers. This environment was evaluated with students within several courses. Even though the environment was perceived as useful by students, they still lacked their own personalization. We describe how the future plug-in tackles this problem.

Keywords: widgets, learning management systems, personal learning environments, flexible education

1 Introduction

In the confrontation of the two worlds with Learning Management Systems (LMSs) on one pole and Personal Learning Environments (PLEs) on the other one a compromise is needed. Both have their pros and cons. LMSs are controlled and managed by universities, they are widespread and can be found in almost every university, students and teachers are used to them. The main critics to LMSs come from lifelong learning perspective. First, LMSs are not flexible to be personalized by learners themselves, impose a specific learning process and an environment on students, and, second, they are disconnected from the Internet cloud of information [9, 5]. These limitations gave birth to PLEs, where learners are in the full control of their learning process and can construct their learning environments themselves by aggregating tools and content required for their specific tasks [7]. However, a PLE also has several disadvantages. First, it requires a rather steep learning curve and strong self-motivation, because students need to understand how a learning process works before they start to take a full advantage of the provided flexibility or even use it [6]. The second reality is that LMSs

are popular in universities, students are used to them and they are reluctant to learn (migrate to) new environments. This paper argues that instead of offering PLEs as an alternative environment to LMSs [8, 4] they should be seen as a complementary technology that would augment the current capabilities of LMSs by providing more flexibility and personalization to their users, which would shrink the gap between the two competing worlds.

The paper discusses how the main PLE components can be brought into LMS via widgets that are portable Web applications implemented with HTML, CSS and JavaScript. Several different standards exist for widgets. We adopted OpenSocial apps specification for our work[4], however other standards can be used in a similar manner (both OpenSocial apps and widgets are used interchangeably in the paper). Our main goal was to bring the benefits of PLE to students and teachers but, at the same time, decrease as much as possible the amount of new things/environments they will have to learn or interact with - a requirement requested by several teachers. With widgets, students and teachers become flexible in personalizing their environments and can bring much more functionality into LMSs than is available there by default. For this task, we developed plug-ins for a popular LMS - Moodle.

The rest of the paper is organized as follows. In Section 2 we introduce in more details the Moodle platform, describe the limitations it has and show how our solution tackles these limitations. In Sections 3, 4 we describe how the plug-in was used in the university courses and how students perceived it. Section 5 highlights the possible improvements of the existing plug-in and introduces a new plug-in that encompasses these improvements. Section 6 provides an overview of related work and Section 7 concludes the paper.

2   Moodle Plug-in Description

Moodle is a popular LMS to manage courses that is the de-facto standard among many Educational Institutions. It is a plug-in based PHP application that can be extended by installing additional modules. These modules have to be installed on a Moodle server by a system administrator. The Moodle view, as shown to students and teachers, consists of a main center area and a rather narrow right column with blocks (Fig. 3). The center area contains main course resources, such as a wiki page, a forum, a lesson, a quiz, etc. The right block contains some helper plug-ins that a teacher can add to every center page, e.g., a calendar, upcoming events, latest news, a recent activity, etc. These are to extend and enrich the functionality of the main center page.

The Moodle flexibility and adaptability is achieved via visual themes and server side plug-ins, thus an intervention of system administrators is required every time a change should be done. Teachers and students are not involved in the process of the customization. Teachers, for example, can not add or remove plug-ins on their own. Differently from Moodle plug-ins, widgets are client-side

---

[4] http://docs.opensocial.org/display/OSD/Specs

applications that can be added to a system by skipping server side installation, which makes them easy to add.

We have two parts of the OpenSocial plug-in for Moodle. The first one adds a new module to Moodle, which is similar to the standard pages module[5]. Once it is installed to Moodle, a teacher can add a "Widget space" to the course, specify a set of widgets for it, and choose whether 1, 2 or 3 column view should be used for widgets display (Fig. 1). The resulting outcome (as displayed to students) is the page with widgets shown in the iGoogle similar fashion, where students can work with several widgets simultaneously (Fig. 2). The second part of the plug-in adds a new block to Moodle[6]. The teacher can add widgets to the right column for already existing in Moodle wiki pages, lessons, forums, etc. (Fig. 3).

Fig. 1: A teacher creates a space with widgets for a course

One of the main benefits of this plug-in is the big pool of available widgets that can be used by teachers. Thus, once the OpenSocial plug-in is installed in Moodle, a teacher can achieve the needed functionality without bothering system administrators with server-side plug-ins installation. The plug-in enhances the flexibility in choosing the resources and tools according to the course specifics: teachers can easily add and remove widgets needed for a course, develop their own ones, etc. Widgets can be found in the existing widget repositories (iGoogle Directory, ROLE Widget Store, etc.). Teachers can re-use existing educational resources and learning objects from external websites. Depending on the desired integration level, teachers can either use an iFrame widget that simply integrates a website URL or develop their own widgets that provide a deeper integration.

From the implementation perspective, the plug-in consists of two main parts. The first part is an engine that renders OpenSocial apps on a page. This engine is a JAVA-based Apache Shindig[7] which represents a reference open-source im-

---

[5] https://github.com/vohtaski/shindig-moodle-mod
[6] https://github.com/vohtaski/shindig-moodle-block
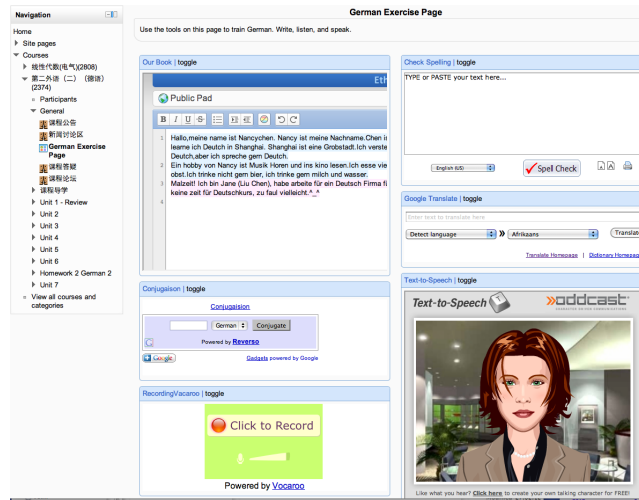[7] http://shindig.apache.org

Fig. 2: Widgets as displayed within Moodle

plementation of the OpenSocial specification. The second part is a PHP module that is responsible for a configuration of a page with widgets, adding and removing them to/from the page and gluing Moodle with Shindig engine. OpenSocial API provides the standardized way to retrieve and exchange information between different Moodle installations and other social networks, which improves data portability and interoperability. More precisely, widgets can query Moodle for data via Shindig engine: they can retrieve the currently logged in user, the current course, its participants as well as save and get arbitrary data. The privacy and security are managed via Shindig engine and it is in the full control of university administrators. However, a widget installed within a course runs on behalf of the teacher who added it and can retrieve/update information that teachers can normally do in their courses. Thus, teachers are responsible for checking the trustfulness of a widget, before adding it into their environments. The ability to retrieve a course information and its participants is achieved via OpenSocial Space extension[8] that allows widgets to adapt to the specific context of the course (contextual widgets). For example, a wiki widget can save data for a course and restrict access to only people engaged in this course. The same wiki widget will behave differently being added to another course: it will have a different wiki history and a different list of participants.

3 Usage at the Online College of Shanghai Jiao Tong University

The Moodle plug-in has been used at the distance university of Shanghai Jiao Tong University (SJTU School of Continuing Education, SOCE). SOCE students

---

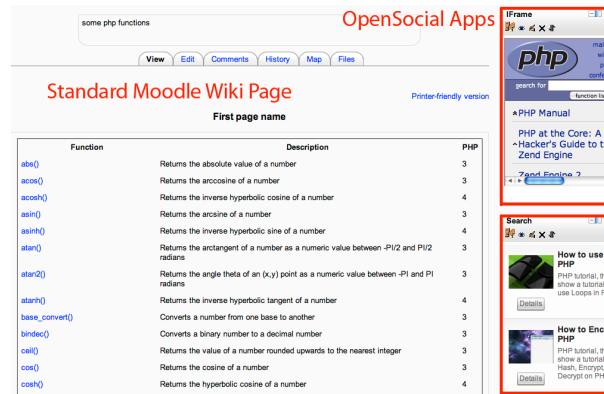[8] http://docs.opensocial.org/display/OSD/Space+Proposal

Fig. 3: Widgets as blocks on the right

are adult learners who study for an associate or bachelor degree [6]. The college implements blended learning, i.e., students can come to classrooms in person to attend live lectures or watch the lectures live through the Web. All lectures are recorded and available for subsequent non-live view. Teaching and learning follows a traditional pattern and is very teacher-centric, with most students watching the lectures rather passively. Within the ROLE project[9], we investigated how to use existing technologies and tools to provide a larger amount of opportunities for interaction and creation. For instance, tools like Voice Recorders and Text-to-Speech allow foreign language students to practice their pronunciation by recording themselves and comparing their speech to the "original" one. Other tools, such as collaborative text editors enable students to work on joint texts in an easier manner than by using forums. A large percentage of the widgets used in SOCE are existing Web pages that train very specific domain knowledge, such as the usage of German articles and French verbs, or visualize data structures such as linked lists.

The Moodle plug-in has been used at SOCE since August 2011 to add ROLE technology via widgets to a number of courses in the domain of foreign languages as well as computer science. The precise courses are "Business English", "English Newspaper Reading", "Data Structures", "German" and "French". Fig. 2 contains screenshots of a Moodle page. In the lectures "Business English" and "English Newspaper Reading" the teacher used the ROLE approach in several ways. Firstly, and in a similar manner to the other lectures, by converting existing resources into widgets suitable for use in training students for various aspects of business English, such as writing CVs, business emails, etc. Secondly, in order to make the course more realistic, the teachers used a role play scenario in which students set up fictitious companies and products. Students were then instructed to create a web page for their company and a slide set presenting their products.

---

[9] http://www.role-project.eu

The resources were uploaded in Moodle and students used a rating widget to assess resources authored by their peers.

In the German and Computer Science courses, the teachers used the PLE (during the courses we referred to the plug-in within Moodle as PLE and we will use this naming in the paper) in order to provide additional exercises, and during the semester also to offer training opportunities, but ostensibly the PLE was utilised by the students mostly for exam preparation. The teachers converted existing web resources consisting of exercises training various aspects of German grammar and visualizations of data structure algorithms to aid the students understand and enhance their learning opportunities.

The usage of ROLE technology allows to extend the widgets and embedded tools with functionality helpful for the overall learning process. For instance, the users' interaction with the tools is captured and used in a visualization which allows teacher and students to see how often they interacted with the resources and to compare their activities to those of their peers. Without ROLE technology, these activities would have been much more difficult to implement in Moodle. While the integration of external exercises is possible with link lists, this approach does not allow to collect interaction data about how often the students actually used the exercises.

4   Evaluation

The current section describes the results of a questionnaire that was conducted with students who were using the described plug-in in the courses at SJTU university. Our main goals were to find out whether students see a value in using PLE components brought via the OpenSocial apps and whether they look for more flexibility, i.e., they can manage widgets on their own.

20 students responded to the questionnaire with approximately half of the students coming from the French course and the other half from the German course. In general, students perceived the PLE to be useful for their learning tasks (Fig. 4). They found it to be helpful to learn in an independent manner, to accomplish work more effectively and they would like to use it in the future.

The second answer we wanted to get was whether students are looking for more personalization and flexibility. The questionnaire showed that students feel comfortable in organizing their own environment for learning by customizing the list of widgets offered by teachers, assembling their own sets of widgets for their learning tasks and to search for existing sets of tools (Fig. 4). In Fig. 5 one can see how students rated the tools that were offered by teachers in the PLE. Even though the majority of widgets were useful for students, a number of them (Listen to your pronunciation, Record yourself, Spell check, Activity visualization) were not highly appreciated by several students. This indirectly confirms the fact that students were looking for more flexibility: they would prefer not to have some widgets at all or probably replace them with other alternatives. It seems as the functionality to remove/hide some widgets on the page or to replace them would be appreciated by some students.
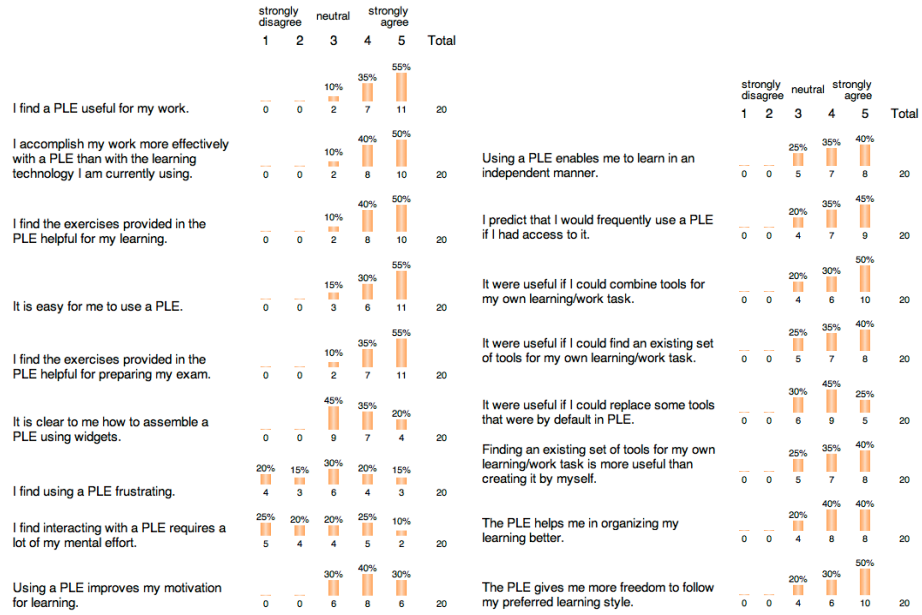
	strongly disagree		neutral		strongly agree	
	1	2	3	4	5	Total
I find a PLE useful for my work.	0	0	10% 2	35% 7	55% 11	20
I accomplish my work more effectively with a PLE than with the learning technology I am currently using.	0	0	10% 2	40% 8	50% 10	20
I find the exercises provided in the PLE helpful for my learning.	0	0	10% 2	40% 8	50% 10	20
It is easy for me to use a PLE.	0	0	15% 3	30% 6	55% 11	20
I find the exercises provided in the PLE helpful for preparing my exam.	0	0	10% 2	35% 7	55% 11	20
It is clear to me how to assemble a PLE using widgets.	0	0	45% 9	35% 7	20% 4	20
I find using a PLE frustrating.	20% 4	15% 3	30% 6	20% 4	15% 3	20
I find interacting with a PLE requires a lot of my mental effort.	25% 5	20% 4	20% 4	25% 5	10% 2	20
Using a PLE improves my motivation for learning.	0	0	30% 6	40% 8	30% 6	20

	strongly disagree		neutral		strongly agree	
	1	2	3	4	5	Total
Using a PLE enables me to learn in an independent manner.	0	0	25% 5	35% 7	40% 8	20
I predict that I would frequently use a PLE if I had access to it.	0	0	20% 4	35% 7	45% 9	20
It were useful if I could combine tools for my own learning/work task.	0	0	20% 4	30% 6	50% 10	20
It were useful if I could find an existing set of tools for my own learning/work task.	0	0	25% 5	35% 7	40% 8	20
It were useful if I could replace some tools that were by default in PLE.	0	0	30% 6	45% 9	25% 5	20
Finding an existing set of tools for my own learning/work task is more useful than creating it by myself.	0	0	25% 5	35% 7	40% 8	20
The PLE helps me in organizing my learning better.	0	0	20% 4	40% 8	40% 8	20
The PLE gives me more freedom to follow my preferred learning style.	0	0	20% 4	30% 6	50% 10	20

Fig. 4: Questionnaire results

	low	average		high		
	1	2	3	4	5	Total
Collaborative text editor	0	5% 1	15% 3	50% 10	30% 6	20
Translator	0	0	10% 2	50% 10	40% 8	20
Business dictionary	0	0	15% 3	40% 8	45% 9	20
Text-to-speech	0	5% 1	10% 2	40% 8	45% 9	20
Listen to your pronunciation	10% 2	0	15% 3	42% 8	31% 6	19

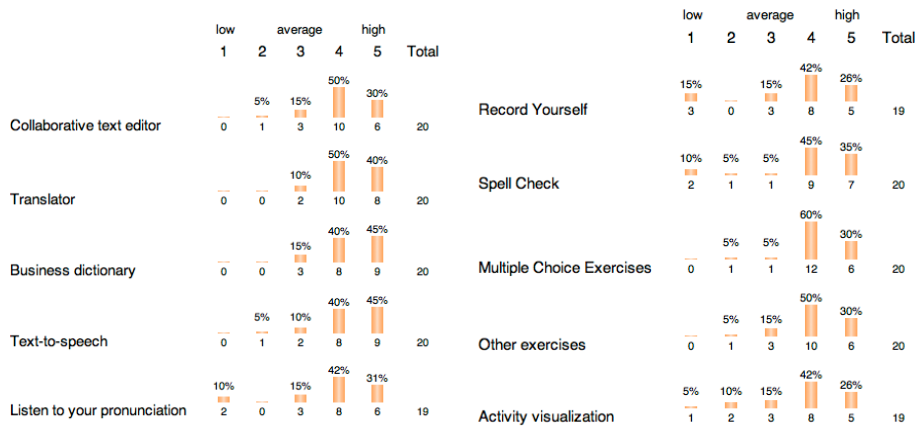|	low	average		high		
	1	2	3	4	5	Total
Record Yourself	15% 3	0	15% 3	42% 8	26% 5	19
Spell Check	10% 2	5% 1	5% 1	45% 9	35% 7	20
Multiple Choice Exercises	0	5% 1	5% 1	60% 12	30% 6	20
Other exercises	0	5% 1	15% 3	50% 10	30% 6	20
Activity visualization	5% 1	10% 2	15% 3	42% 8	26% 5	19

Fig. 5: Tools as rated by students on a scale from 1 (not useful) to 5 (very useful)

5 Future Steps

The plug-in described earlier in this paper takes several steps forward in turning Moodle into a PLE. We will now describe a future plug-in that brings in an actual PLE into Moodle, from a PLE installation running side-by-side with Moodle, while maintaining integration with Moodle through its Web services (Fig. 6).

The embedded PLE has improved personal aspects. Students can add widgets (from an integrated widget store) alongside those chosen by the teacher, which is difficult to enable in the existing plug-in because Moodle requires additional access rights, which students normally do not have. Students can also change the preferences of any widget, configuring it to their needs, overriding the teacher's preferences. Furthermore, widgets can be rearranged and resized.

A dashboard is added to the bottom of the Moodle page and contains widgets that are chosen by the student without teacher's involvement. The dashboard is available independently of the course, which means that students can add widgets that they can access in every course. Outside of Moodle, the dashboard can be accessed on any Web page by means of a bookmarklet (i.e., a Web browser bookmark or favorite that executes a small piece of JavaScript on the current Web page; in our case, the script loads the dashboard). This provides a ubiquitos (cross-organizational) access to the PLE for students which is an important part of lifelong learning [9].

This new plug-in does not integrate with Moodle database. Instead, the plug-in retrieves a token that it passes on to the PLE, allowing the PLE to access Moodle Web services [3] on the authenticated user's behalf. In this way, the embedded PLE is able to access information such as course metadata and participants, and make that information available to widgets. The PLE augments this with support for widgets to store data within the context of a course, similar to the App Data and Media Items of OpenSocial, and other functionality that would not be available from Moodle alone, such as real-time communication. The interface provided to widgets is generic rather than Moodle-specific, and it is possible to implement similar integration that offers the same widget interface also for other LMSs.

As the PLE installation is actually running side-by-side with Moodle, it is possible to access it without Moodle. While it is desirable to keep everything within the same Website (which the embedding of the PLE achieves), it is in some cases better to access the PLE separately. The UI overhead of Moodle page layout is avoided, leaving more screen real estate for the widgets. Additionally, it is possible that the standalone PLE (in this case without the full access rights to the course) could be used as a means to attract students to the course, by giving a taste of what the course is about.

Finally, there are some technological aspects that should make this kind of integration a cleaner solution, although being more complex than integration directly with Moodle database. The more loosely-coupled integration via Moodle provided interfaces is likely (but not certainly) to offer better isolation to changes in Moodle as new versions become available. More importantly, it may be easier to convince administrators to allow the installation if it only integrates

with Moodle through its provided interfaces. As this simultaneously reduces the dependency on Moodle, it could also be the first step in leaving the world of LMSes completely, making a PLE-only solution possible.
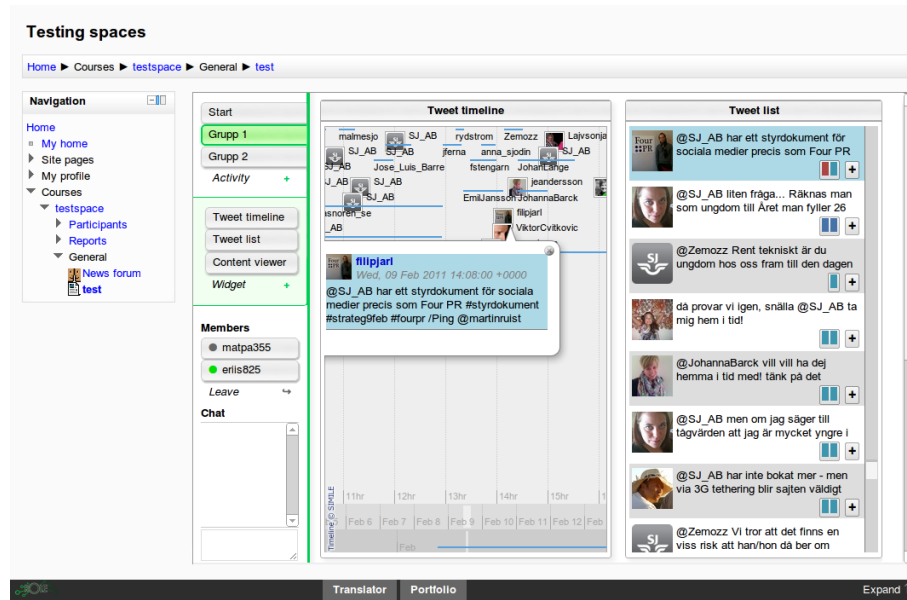

Fig. 6: A mock-up of the embedded PLE plug-in for the future step

6 Related Work

The earlier work of the authors [6] investigated the usage of a PLE in a similar setting (a French course at SOCE), but with significant differences. First, the PLE was implemented in an external system (Liferay[10]) which was not integrated into the school LMS, thus introducing an additional layer of complexity due to the different user interface and additional log in. Also, the used ROLE technology was still in a very early stage. The paper [1] investigates the applicability of social media platforms within an academic context. More specifically, a social media platform Graasp[11] was used to serve as a stand-alone PLE to augment and enrich the offer of learning tools provided by universities. Through the notion of a space it allowed learners to aggregate and organize both institutional and external resources within different contexts and to conduct learning activities in these contexts. It should be noted that widgets usage is not limited to language

---
[10] http://liferay.com
[11] http://graasp.epfl.ch

learning. Widgets approach is used for teaching chemistry, programming, etc. For example, widgets are used to conduct remote control labs [2].

7   Conclusion

The paper presented an approach to extend the existing LMS Moodle with a plug-in enabling OpenSocial apps to run within Moodle. Introducing these apps allows to bring PLE functionalities to LMS, namely, flexibility in managing tools used by people for their learning goals and aggregation of external resources from the Internet. The first version of the plug-in allows teachers to freely choose a set of tools for their courses. The plug-in was used within several courses by students and the questionnaire showed that students find this environment useful, however they still look for more personalization where they can manage apps themselves. The new plug-in that is currently under development is planned to tackle this problem. Thus, our future plans are to finalize the new plug-in, introduce it to students within the courses and evaluate it.

Acknowledgments

References

1. E. Bogdanov, F. Limpens, N. Li, S. El Helou, C. Salzmann, and D. Gillet. A Social Media Platform in Higher Education. 2012. accepted to Educon Conference.
2. E. Bogdanov, C. Salzmann, and D. Gillet. Widget-Based Approach for Remote Control Labs. *9th IFAC Symposium on Advances in Control Education*, 2012.
3. M. A. Conde, D. A. G. Aguilar, A. Pozo de Dios, and F. J. G. Penalvo. Moodle 2.0 web services layer and its new application contexts. 73:110–116, 2010.
4. F. Moedritscher, G. Neumann, V. M. Garcia-Barrios, and F. Wild. A web application mashup approach for eLearning. *OpenACS and LRN Conference*, 2008.
5. C. Severance, J. Hardin, and A. Whyte. The coming functionality mash-up in Personal Learning Environments. *Interactive Learning Environments*, 2008.
6. C. Ullrich, R. Shen, and D. Gillet. Not Yet Ready for Everyone: An Experience Report about a Personal Learning Environment for Language Learning. *Advances in WebBased Learning–ICWL 2010*, pages 269–278, 2010.
7. M. Van Harmelen. Personal Learning Environments. *Sixth IEEE International Conference on Advanced Learning Technologies ICALT06*, 16(1):815–816, 2006.
8. M. Van Harmelen. Design trajectories: four experiments in PLE implementation. *Interactive Learning Environments*, 16(1):35–46, 2008.
9. S. Wilson, O. Liber, M. Johnson, P. Beauvoir, P. Sharples, and C. Milligan. Personal Learning Environments: Challenging the dominant design of educational systems. *Journal of eLearning and Knowledge Society*, 2(1):173–182, 2007.