Student-oriented planning of e-learning contents for Moodle

Valentina Caputi[a], Antonio Garrido[b]

[a]Dipartimento di Fisica, Università della Calabria

via Pietro Bucci, 87036 Arcavacata di Rende (CS), Italy. Tel: +39 0984 4911, fax: + 39 0984 49 4401.

E-mail: valentina.caputi@fis.unical.it.

[b]Departamento de Sistemas Informáticos y Computación, Universitat Politècnica de València

Camino de Vera, s/n 46022 Valencia, Spain. Tel: +34 963877007, fax: +34 963877359.

E-mail: agarridot@dsic.upv.es.

ABSTRACT

We present a way to automatically plan student-oriented learning contents in Moodle. Rather than offering the same contents for all students, we provide personalized contents according to the students' background and learning objectives. Although curriculum personalization can be faced in several ways, we focus on Artificial Intelligence (AI) planning as a very useful formalism for mapping actions, i.e. learning contents, in terms of preconditions (precedence relationships) and causal effects to find plans, i.e. learning paths that best fit the needs of each student. A key feature is that the learning path is generated and shown in Moodle in a seamless way for both the teacher and student, respectively. We also include some experimental results to demonstrate the scalability and viability of our approach.

Keywords: E-learning; personalization of learning paths; Moodle; intelligent planning; Artificial Intelligence.

1. Introduction

E-learning is increasingly widespread in the educational world by taking advantage of information, computing and telecommunication technology, together with a wide range of electronic multimedia uses. The validity of online assessment methods has already been demonstrated in (Hewson, 2012). Furthermore, the application of multimedia tools have a great impact on education, training and, in general, on curricula considerations. These tools support (and facilitate) learning, and their usage within e-learning makes the learning process friendly to students, who interact with teachers in a better way than in traditional classroom teaching (Martín-Blas & Serrano, 2009). In fact, e-learning permits us to remove the barriers of time and space, which are characteristic of traditional teaching worldwide, because the access to a course is now possible by a simple connection to Internet. In addition, e-learning makes it possible better monitor the learning progress of the students. This is very valuable for students and teachers because they can realize students' learning state in a very easy way.

1.1. Learning Management Systems

E-learning requires two kinds of activities: communication activities (e-mail, forums, conferences, on-line blogs, etc.), and exploration activities (mainly navigation of contents). These activities usually take place on a LMS (Learning Management System). A LMS is a platform for administrating, documenting and delivering e-learning contents, which offers the enrolled students a vast number of courses with highly customizable capabilities. Many of these platforms, such as Moodle, Sakai, Docebo, Atutor, Ilias, .LRN, etc., are increasingly being used in schools and universities as a powerful support and improvement for teaching activities. Although LMSs are a fraction of educational ecosystems where different platforms (LMSs, e-portfolios, assessment systems, curricula management systems, etc.) live together and collectively support e-learning, the great risk here is not to exploit LMSs up to their full potential. On the contrary, LMSs are traditionally used simply as mere "repositories" of learning contents. For the best use of these contents, it is fundamental not to consider them in an isolated way (and, consequently, not to consider a LMS just as a simple database), but as part of a much larger system in which contents are aggregated for the construction of courses that can be fully personalized. Intuitively, the underlying idea is to build student-oriented learning paths by combining appropriate learning contents, where a learning path is a set of activities that a student needs to perform to achieve a certain level of knowledge.

It is important to note that each student has his/her own characteristics (profile, learning style, prior background and learning objectives). These individual traits are very useful to provide each student the most adequate learning path to attain his/her learning outcomes (Garrido & Onaindia, 2013; Papanikolaoum, Grigoriadou, Magoulas, & Kornilakis, 2002). In other words, it is not enough to plan a general learning path for all students but to personalize as much as possible each learning path. Therefore, what is essential for a LMS is, first, to identify a specific learning path for each student, and second, to provide the maximum possible autonomy to him/her. Thus, learning paths should be student-oriented, and planned to meet the individual characteristics of each student.

1.2. Motivation

We motivate the necessity of personalization by using a simple example. Let us imagine that two students, Paul and Kate, enroll on an Italian course. The course consists of three sequential modules (corresponding to three different learning levels): "Elementary module", "Intermediate module" and "Advanced module"; and it is possible for a student to take the entire course or just a part of it. Let us suppose that Paul has sufficient knowledge of Italian and only wants to improve his grammar. Kate, however, has already a good level of Italian but wants to speak more fluently. Certainly, it makes no sense to design the same learning path for both students. It is necessary to plan for Paul a path that only includes the “Intermediate module” and for Kate a path that includes the “Advanced module”. Starting from these considerations, it is necessary to find (and to put it into practice within a LMS) the best learning path so that each student achieves his/her learning objective, starting from his/her initial characteristics.

Although the sequence of the Italian course’s activities may seem unique, we consider the portions of the course assigned to Paul and Kate as two different learning paths. More generally, we can consider a course where the sequence of activities may be (or not) unique, but this does not necessarily mean the sequence of learning activities is the same because different students can skip parts of the course and take different learning paths according to their specific needs. For example, in a course composed by *n* activities, we can have learning paths of *n, n-1, n-2… 1* activities, and in different orderings. And the personalization can be even more flexible. If two activities achieve the same learning outcome (e.g. by means of a multimedia document and by reading a paper, respectively), one student could take the former and another student the latter. In other words, it is possible to find learning paths that involve, for example, the same number of learning activities but in a different sequence (in line with the course’s constraints of causality) or different sets of activities, depending on the specific learning outcomes and students’ profiles/learning styles.

Consequently, we need planning to select the best sequence of learning activities (and in the right order), from the entire set of activities defined by the teacher, to satisfy each student’s learning goals. It is necessary to plan the steps to reach one or more goals because the steps cannot be a simple, arbitrary sequence of learning activities but what the student needs to do/learn in an adequate causal ordering. Also, although a student can tick some parts that she/he already knows, we still need planning. Perhaps in a long sequence of activities the student has a background on some parts, but this does not mean that we do not need to plan the remaining part of the sequence. In other words, the planner needs to plan the remaining part of the path to satisfy all the learning goals, and this can be significantly different from one student to another.

Additionally, a good planning activity should be accompanied by a good monitoring activity of the learning paths. In fact, though a student is following a certain learning path, that path could eventually need to change, because of discrepancies between expected and real results, updates on the learning objectives, etc., and a re-planning of the path, in part or whole, may be necessary.

1.3. Related work

The need for systems that automatically build student-oriented learning paths by combining appropriate learning contents has become more and more intense in the last years (Baylari & Montazer, 2009; Chen, 2008; Garrido & Onaindia, 2013; Kontopoulos, Vrakas, Kokkoras, Bassiliades, & Vlahavas, 2008; Papanikolaoum, Grigoriadou, Magoulas, & Kornilakis, 2002). Generally speaking, literature abounds with works to exploit techniques on nearly all aspects of e-learning.

There are a variety of studies that face the problem of curriculum personalization in different ways, without focusing on a specific LMS. For example, (Dorça, Lima, Fernandes, & Lopes, 2013) show three different strategies to automatically detect and exactly adjust students' learning styles, by taking into account students' performance. In another approach, (Thyagharajan & Nayak, 2007) suggest to address the automatic selection and integration of adequate learning materials for a student by using Web services based on student's features as initial knowledge, objectives, preferences, etc. More generally, (Thyagharajan & Anbumani, 2009) propose a model to help teachers build an interactive courseware, without being experts in multimedia programming and Web technologies, to get the adaptive presentation of multimedia elements through streaming to the students by considering their specific needs.

(Laurillard, Charlton, Craft, Dimakopoulos, Ljubojevic, Magoulas, Masterman, Pujadas, Whitley, & Whittlestone, 2013) highlight that the use of digital technology in teaching is not always optimized and suggest the Learning Design Support Environment project as a way to enable the teachers to develop and test their learning ideas in terms of effective learning design. (Chang & Ke, 2013; Chang, Hsieh, & Li, 2010; Tan, Shen, & Wang, 2012) apply a genetic algorithm approach to customize and personalize course generation. The results of these works are promising but their application to standard LMSs can be difficult.

From a perspective based on the design, analysis and scoring of tests, the personalization of e-learning systems has been approached by using the Item Response Theory (PEL-IRT) which, by considering the difficulty of the learning materials to be provided and the ability of the students, finds personalized learning paths (Chen, Lee, & Chen, 2005). Another work based on the students' results of pre-tests, has led to a genetic-based customized e-learning system which conducts to a personalized curriculum sequencing (Chen, 2008). Also, a real-time assessment of students' productivity and interest in learning by using a Recommender System has been considered in (Kaklauskas et al., 2013). Other authors combine a personalized multi-agent e-learning system based on item response theory with artificial neural networks and soft computing methods (Baylari & Montazer, 2009; Brusilovsky & Vassileva, 2003; Idris, Yusof, & Saad, 2009).

Like in our case, several works use AI methods in order to identify student-oriented learning contents. In particular, the prediction of the students' behavior to help in the decision-making teaching procedures in open and distance education has been considered by using Bayesian networks (Xenos, 2003). Such a work takes into consideration general students' behavior without focusing on specific learning contents. On the other hand, similarly to our idea, intelligent planning has been used for learning paths' personalization (Kontopoulos, Vrakas, Kokkoras, Bassiliades, & Vlahavas, 2008). That work focuses on creating a new planning ontology from the e-learning information and use standard planners to solve the problem. On the contrary, we do not create any new ontology, but we perform a knowledge engineering-based mapping from Moodle (Module Object-Oriented Dynamic Learning Environment) to standard PDDL (Planning Domain Definition Language) to make our compilation ready for any of the PDDL planners that are publicly available.

Moodle has been considered by previous works such as (Romero, Ventura, & García, 2008), which used data mining techniques in order to improve the course management (i.e. statistics, clustering, classification, visualization, etc.), without focusing on a real-time planning activity. Additionally, some other papers such as (Martín-Blas & Serrano, 2009) just focus on Moodle's characteristics and consider this platform as a valid tool in order to perform learning/teaching activities. That kind of work is oriented to a specific course but does not focus on the possibility of a learning path's planning activity in real-time.

In the line proposed by (Garrido, Fernández, Morales, Onaindía, Borrajo, & Castillo, 2013), there are tools that use IMS structures such as SCORM or Learning Design in order to get the personalization. But this means to make important changes from the Moodle's point of view. On the contrary, the idea that underpins our paper is to integrate the intelligent planning techniques within Moodle by making minimal changes to create a real time learning paths' customization based on the specific students' characteristics in relation with determined learning contents.

1.4. Objectives of the paper

This paper builds on the general work of (Garrido & Onaindia, 2013) and extends the results presented in (Caputi & Garrido, 2013) to offer now a thorough design, development, implementation and testing of intelligent personalization in Moodle. Particularly, in this paper the personalization of an e-learning path is faced from the point of view of AI planning through the automated compilation of e-learning models. We have fully adapted the knowledge engineering planning mapping introduced in (Garrido & Onaindia, 2013) to be directly used in Moodle, while trying to minimize the modifications in Moodle. It is important to highlight that our general idea of applying planning to e-learning personalization does not depend on any specific LMS. But when implementing it on top of a particular LMS, some specific technical issues are necessary to face and solve, which means that eventually there will be some LMS dependent changes. Moodle is a platform that includes a constructivist and social constructionist approach to education, emphasizing that students (and not just teachers) contribute to the educational experience. Consequently, Moodle facilitates the interaction among students in real time by permitting the exchange of views and sharing of knowledge and difficulties while taking the courses.

We detail here an automated way to bridge the gap between the model of (e-learning) course implemented in Moodle and the planning model for supporting contents personalization, which means the generation of student-oriented learning paths. To our knowledge, there are three features that show essential to derive the greatest possible learning benefits: i) a transparent way to translate from the Moodle's insights to the planning ones, and vice versa; ii) a seamless procedure to run an intelligent planner to personalize as much as possible each learning path, depending on each particular student; and iii) a simple way to monitor the progress of the students in their learning paths and the possibility to re-plan to adapt them to new scenarios. The thorough explanation of these features is the main goal of this paper, in which we also provide some experimental results to evaluate the scalability and feasibility of our work.

2. Planning in the context of e-learning

Most of human activities involve some kind of planning of tasks to reach an objective. According to Cambridge dictionary, planning is "the activity of thinking about and deciding what you are going to do or how you are going to do something". Therefore, intuitively, planning is about taking decisions on what is the most adequate action to be done in every moment. From a more technical point of view, intelligent planning involves the representation of actions and world models, reasoning about the effects of actions, and techniques for efficiently searching the space of possible plans. In other words, given a domain of possible actions, intelligent planning selects a subset of them (e.g. a plan where actions are ordered according to their causal-effect relationships) that, after their execution, allow us to reach an objective state starting from an initial state (Ghallab, Nau, & Traverso, 2004).

2.1. PDDL, a Planning Domain Definition Language

Planning technology has witnessed incredible advances in the last decades. State-of-the-art planning algorithms deal with problems with hundreds (and even thousands) of actions in a few minutes. In order to unify the definition of planning problems and promote an interchangeable use of planners, a standard Planning Domain Definition Language, PDDL, was agreed by the planning community (Ghallab, Nau, & Traverso, 2004).

The implicit formalism behind PDDL is the separation of the domain data, to describe a family of similar problems and enhance reutilization, from the problem data, thus requiring two plain text files. First, the domain file contains all the actions that could be applied. The semantics of each action is described in terms of: i) a name that, grounded with the values of the optional parameters, acts as a unique identifier; ii) an optional duration to model problems where actions have different duration -otherwise all durations are considered unitary; iii) a set of preconditions that must hold before the action execution, i.e. causal precedents; and iv) a set of effects that are asserted once the action is executed. Second, the problem file contains the initial state of the world, the goals that need to be achieved by using the actions of the domain and, optionally, a metric to be optimized such as makespan, number of actions, cost, etc. A planner takes these two files and returns a plan, as a set of ordered actions, which allows us to reach the objectives starting from the initial state in an optimal or suboptimal way.

2.2. Planning vs. e-learning

Metaphorically speaking, the personalization of e-learning paths is analogous to the execution of a planning process. The main elements of e-learning are: i) the background and student's preferences, ii) the learning outcomes to achieve, iii) the learning contents adapted to the student's profile, iv) the ordering relationships, and v) the specific learning path for each student. Through an e-learning to PDDL mapping, which will be detailed later, these elements can match, respectively, with the next planning elements: i) the initial state, ii) the problem goals, iii) the actions, iv) the causal links, and v) the solution plan.

The optimization process that planning offers is also very interesting, because students and teachers often prefer a quality learning path in terms of time, resources usage and/or cost, and not yet another path.

2.3. A simple PDDL example

Let us revisit the Italian course considered in our motivation example and let us imagine that the "Elementary module" is required for the execution of the "Intermediate module". Let us imagine that the minimum time that a student must spend in "Intermediate module" is 5 hours (300 minutes). In a very general way, that is, without deepening into the specifics of individual activities that make up the entire course, the "Intermediate module" can be represented in a PDDL domain a simple action (executable by a given parameterized student), with its duration, preconditions and effects, as shown in Fig. 1.

Let us assume that Paul and Kate need to take, respectively, the "Intermediate module" and the "Advanced module". Imagine, in fact, that Paul has already taken the "Elementary module" whereas Kate has already taken the "Intermediate module". We can represent this information as the initial state, the goals and metric to be optimized (total-time that stands for makespan) of a PDDL problem, as shown in Fig. 2.

```
(:durative action intermediate
 :parameters (?s - student)
 :duration (= ?duration 300)
 :condition (at start (and
          (not (intermediate_done ?s))
          (elementary_done ?s)))
:effect (
          (at end (intermediate_done?s))))
```

Fig. 1. The representation of the "Intermediate module" as a PDDL action of the domain.

```
(:init
          (elementary_done Student_Paul)
          (intermediate_done Student_Kate))
(:goal (and
          (intermediate_done Student_Paul)
          (advanced_done Student_Kate)))
(:metric minimize (total-time))
```

Fig. 2. The PDDL problem for Paul and Kate.

3. Personalization of learning paths and application to Moodle platform

The personalization of learning paths involves the development of different activities, as explained in the following paragraph.

3.1. General overview

As shown in Fig. 3, once chosen the LMS platform on which to focus (Moodle in our case), the personalization of learning paths requires developing a number of activities. First of all we need to carry out a mapping of the different modules present in the platform. This activity includes the understanding of the relationships between the different modules and the study of the way in which each student can enter the platform information about his/her background and his/her learning objectives. The next step consists in building a course by using the most appropriate resources which Moodle offers. Once structured the course, it is necessary that students who take it fill into the platform information about their own initial states and learning goals. At this point we can proceed with the translation of the relationships between the course's activities into actions of a PDDL domain, while the information about students' initial states and learning goals has to be translated into a PDDL problem. PDDL domain and problem can be used by any standard planner, in order to generate a plan, or a set of learning paths, one for each student enrolled in the course. By using the tools available in the platform it is necessary to ensure that each student only visualizes and takes the portion of the course present in his/her own learning path. Finally, it is required to develop a monitoring activity that takes into account all the changes that can occur in the performance of each learning path and the possible variations of the students' goals, in order to eventually re-plan the paths.

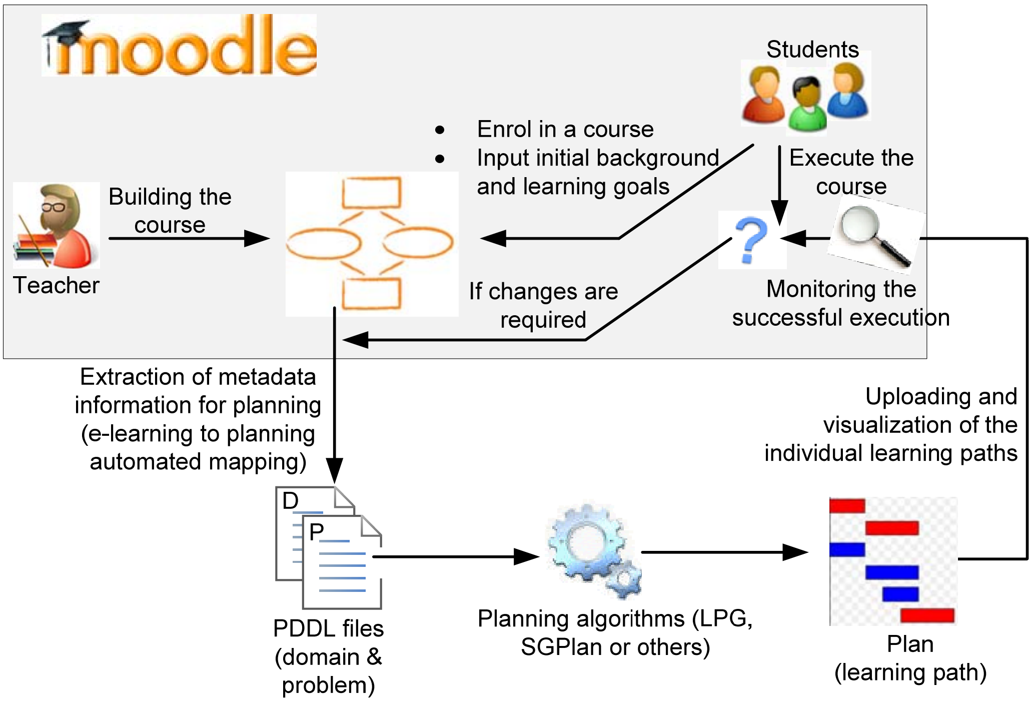

Fig. 3. Activities required for personalized learning paths.

3.2. Moodle's description

We have decided to use Moodle (http://www.moodle.org/), a Learning Management System implemented as a free, open-source PHP Web Application, to offer and conduct online learning contents (Fig. 4). There are many reasons to use Moodle. For example, it is a platform easy to be used (Moodle is very "user friendly" although, like in all computing platforms, it is required some prior IT knowledge) and, if necessary, it results easy to modify. Moodle works on all systems that support PHP, such as Windows, Linux and MacOS and can use databases in different formats such as Oracle, PostgreSQL and MySQL. Moodle is used all over the world in different universities, schools and companies, with excellent credibility, increased by the fact that there is a forum (https://moodle.org/forums/) that connects users and developers all around the world to share resources and ideas, support new users, etc.

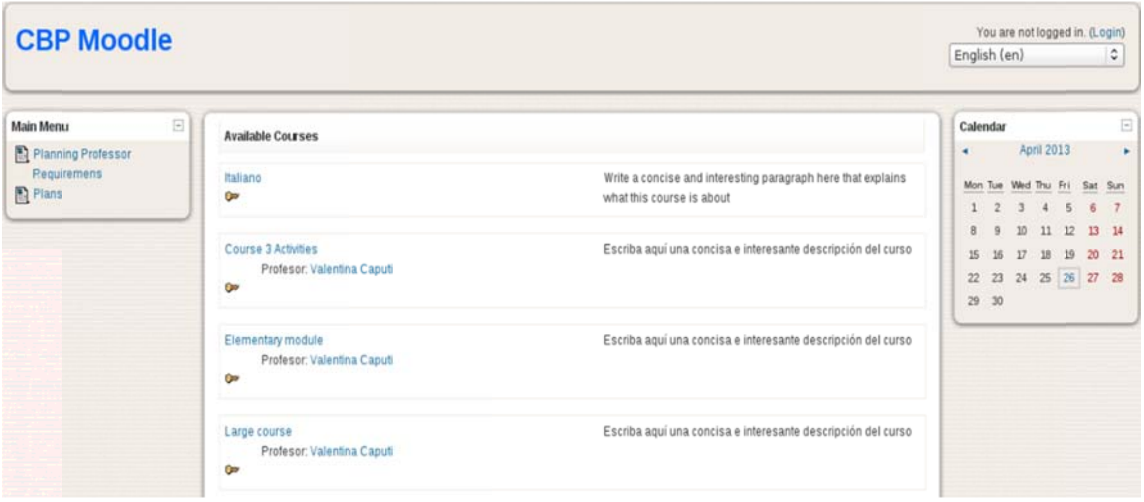

Fig. 4. An example of Moodle's main page containing three courses.

3.3. Mapping: from planning to Moodle

We carry out a mapping by considering the elements of the platform and how they can be adapted to a planning activity. In particular, in Table 1 we can observe the mapping of the elements generally used in learning paths' planning and the elements available in the platform. Note that this mapping relies on a general translation from e-learning to planning (Garrido, Fernández, Morales, Onaindía, Borrajo, & Castillo, 2013), but it now needs to be slightly adapted to use the Moodle terminology.

Table 1
Mapping between planning terms and Moodle terms

Planning in general	Planning in Moodle
Course	Course
Tasks	Activities: lessons, chat, forum, wiki, etc.
Initial background	Student's profile (very limited)
Learning goals	Student's profile (very limited)

As Table 1 shows, there are no substantial differences as regards to the general definition of a course (and the tasks that are part of it) in planning terms, and a course in Moodle. The main elements necessary to build a course in Moodle are activities and resources. While the activities (chat, forum, wiki, database, lesson, etc.) are considered as the main way to interact with the students, resources (page text, Web page, link to a Web page, label etc.) are used to transmit additional information to activities. Hence, the various contents of a general course can be translated into the different activities available in Moodle.

We found some limitations with the definition of the students' initial states and learning goals (Caputi & Garrido, 2013), and the definition of students' learning styles. As we will explain in more detail in the following paragraphs, a student can just input into the platform general information but it is not expected that s/he inputs detailed information about his/her background and learning goals, which results fundamental in learning path's planning. This limitation involves an important drawback, that is, a lack of information for the subsequent phase of definition of the PDDL problem. On the other hand, although dealing with learning styles and students' profiles is totally possible in planning (Garrido, Fernández, Morales, Onaindía, Borrajo, & Castillo, 2013), we do not consider this because it requires: i) taking into account both the type of each learning task and the opinion of an expert to assess the students' learning styles, and ii) map types of learning styles with types of tasks to decide which combinations are better. To our knowledge, this information cannot be easily included/modeled in Moodle. Consequently, although planning allows for more personalization, we are constrained by the limitations of Moodle's modules, which do not include the possibility to define this kind of information.

3.4. Building a course by using Moodle's lessons

Our work on Moodle begins with the definition of a course. For example, we can organize the Italian's "Elementary module" as shown in Fig. 5. The module is composed by 10 lessons: we decided to only use the lesson because it facilitates the definition of relationships between activities and it is very simply to use for both students and teachers.

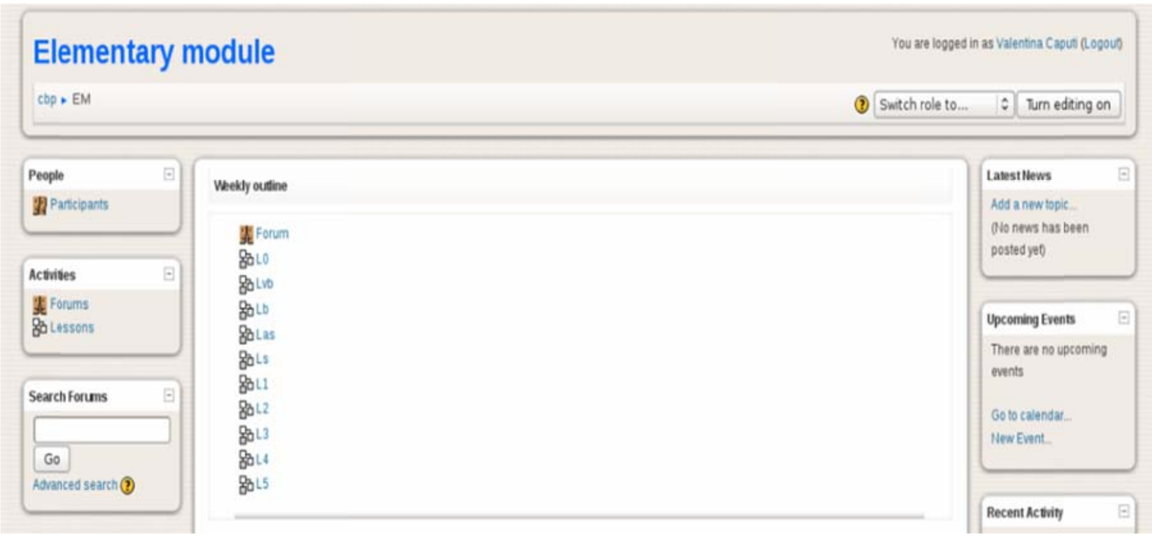

Fig. 5. "Elementary module" in Moodle.

In general terms, a lesson in Moodle is composed of a sequential series of pages. The total number of pages depends, in general, on the content which is necessary to provide. The teacher can, at any moment, add or remove pages, depending on the specific educational requirements. At the end of each page there is a question with a number of possible answers. Teachers can set the answer mode in several ways: true/false, multiple-answer, numeric answer, etc. If the student answers correctly, then s/he will be able to continue with the next page in the lesson, or otherwise s/he will have to repeat the entire lesson or the single page (depending on how it is originally defined by the teacher). The student may not be obliged to complete the entire lesson in one session, but s/he can access it several times (each time the system will show him/her the point where s/he left it). At the end of the lesson, the teacher can optionally associate a final grade of completion, i.e. a percentage obtained by considering the contribution of the resolution of the individual pages. If nothing is specified by the teacher, then the percentage of completion by default is 100%.

3.4.1. Course enrollment

In order to make sure that every student can perform the learning path most appropriate to his/her needs, it is essential to know his/her background and his/her learning goals before starting of the course. Unfortunately, Moodle (as well as other LMSs) is not originally designed to provide custom content to the different students enrolled in a course. For this reason, when a student has to input into the platform his/her personal information (see Fig. 6), there is not a field in which s/he can express what is, for example, his/her own previous knowledge with respect to a certain topic and what is the learning level that s/he wants to achieve. Rather, s/he has only the ability to enter fairly generic information such as e-mail address, preferred language, etc.

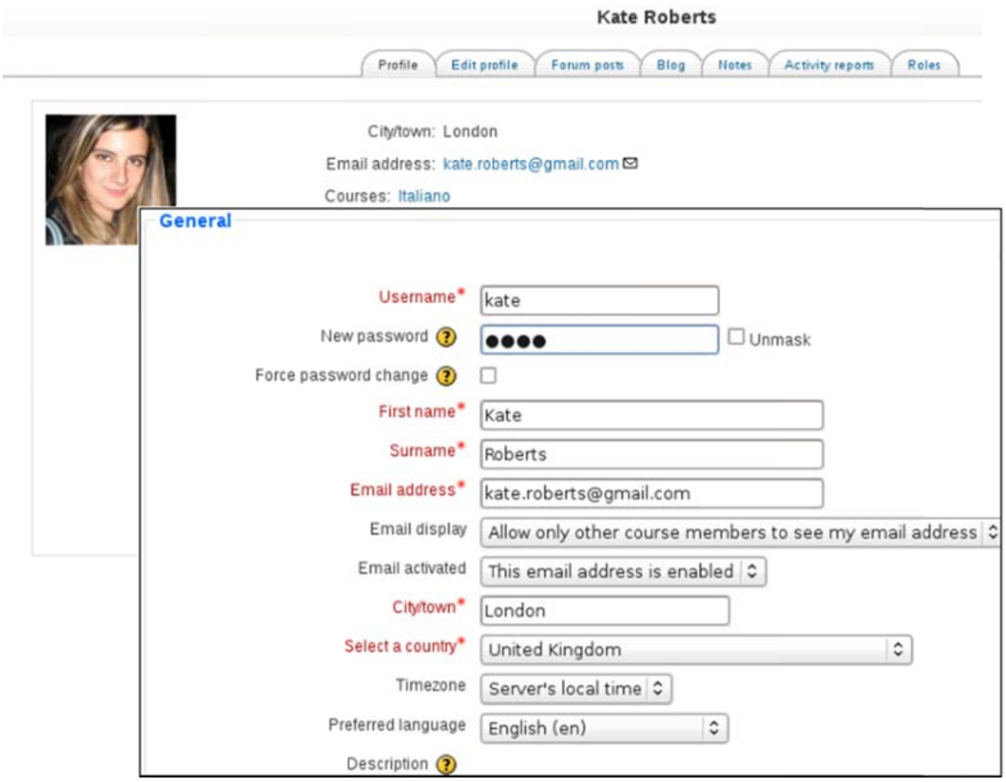

Fig. 6. Kate's profile in Moodle.

We find a way to define the initial state and goals of each student by creating a fictitious lesson, named initial questionnaire or L0 for short (as shown in Fig. 5), to be carried out by the students before the effective start of the course. Essentially, it is a dummy lesson that consists of two question pages, the first to define all the possible initial states and the second to input the possible learning goals. This is a simple way to define the initial background and learning goals of each student by using the standard functionality of Moodle.

3.4.2. Lessons' relationships within the course

When defining a lesson in Moodle, the teacher can define two kinds of relationships with other activities: "dependency" and "activity link". The first is a binding link while the second is considered as a simple suggestion. Let us clarify this concept with an example. Let us imagine that the teacher is structuring the lesson L5 of the "Elementary module". Suppose that L5 cannot be executed before the completion of the lesson Las. The teacher can define into the platform the dependence of L5 from Las (see Fig. 7). Let us imagine also that the teacher wants to suggest to the students the lesson Ls to execute after the completion of L5. This should be achieved by setting Ls in the field "activity link" of the L5's configuration page, as shown in Fig. 7.

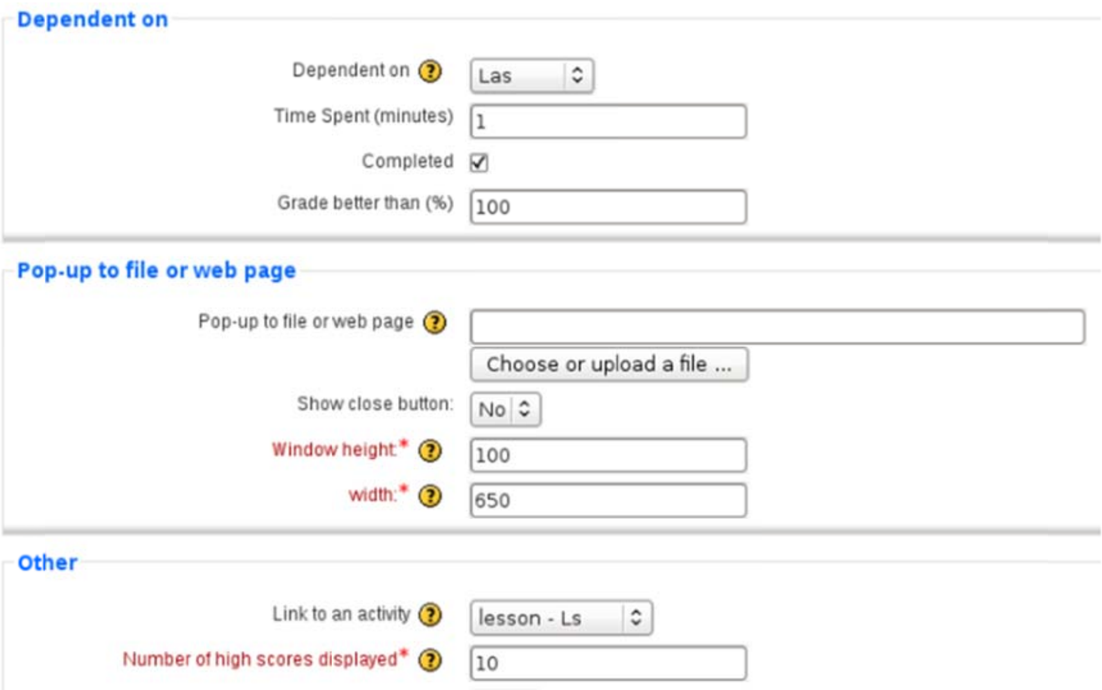

Fig. 7. Definition of the relationships for the lesson L5 of the "Elementary module".

The "activity link" and the "dependency" help us create only one-to-one relationships between lessons. From the perspective of planning it is very limiting because it requires the student to perform a single possible path within the course (Caputi & Garrido, 2013). Consequently, to ensure the possibility to define multiple relationships within a course, we insert the concept of fictitious lesson (Lf). Unlike a real lesson (Lr), it represents the achievement of a certain learning level (or learning state) on which depends the performance of other lessons and that can be reached by taking one or more alternative lessons.

By considering the previous observations, the "Elementary module" can be composed of an initial questionnaire (L0), five real lessons (L1, L2, L3, L4, L5) and four possible learning states, "very bad", "bad", "almost sufficient" and "sufficient", to which we associate the fictitious lessons Lvb, Lb, Las, Ls respectively (see Fig. 8). Each learning state is also part of L0's multi-choice question pages to define the initial state and

learning goals of each student. So, it is important the consistency between the nomenclatures used in the definition of L0 and in the rest of the course. The dependencies between lessons are represented as continuous lines while the suggestions as broken lines. Fig. 8 also shows the duration of each lesson. In particular, we have to assign a minimum execution time for each lesson, so we suppose that fictitious lessons have duration of 1 minute (null times are not allowed in Moodle).
Therefore, the result that we want to achieve in our system is that once a teacher defines a course each student (by simply completing an initial questionnaire) can already get a learning plan suited to his/her specific needs.

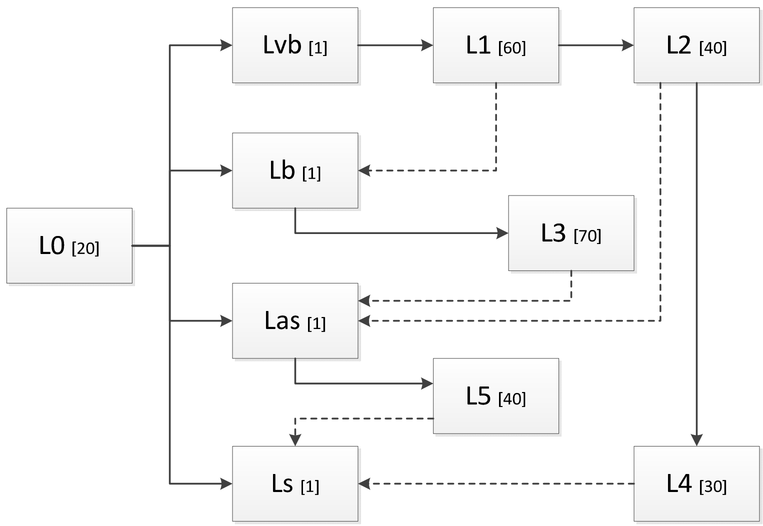

Fig. 8. Structure of "Elementary module". Durations in minutes between brackets, "activity links" as broken lines and "dependencies" as continuous lines.

3.5. Compilation of a PDDL document from the database of Moodle

When designing a personalized learning path we consider both the information about the course and each student who takes it. Course's information has to be translated into PDDL actions, in order to define a PDDL domain. On the other hand, a PDDL problem is generated from the initial states and goals of each student. The PDDL domain does not depend on a specific PDDL problem, but describes a family of similar problems.
In order to generate a correct learning path for each student, it is necessary the proper structuring of the course in terms of "dependency" links and "activity links" between lessons. In particular, it is necessary to establish these relationships so as to have the right analogies with the PDDL domain to be generated.
As already mentioned, a course consists of a set of real lessons $Lr=Lr_1,\ldots, Lr_n$, a set of fictitious lessons $Lf=Lf_1,\ldots, Lf_n$ and an initial questionnaire (that we named L0), that is a particular fictitious lesson. The constraints to be respected when structuring a course are the following:

- L0 has not "dependency" constraints or "activity links" to any lesson;

- Each Lr_i can depend on a real or fictitious lesson. If Lr_i allows us to reach a certain learning state, it is necessary to set in the Lr_i's configuration page an "activity link" that leads to the fictitious lesson representative that state;
- Each Lf_j has only a "dependency" on L0 and does not have "activity links".

The features listed above become preconditions and effects for actions in the PDDL domain, as shown in Table 2.

Table 2
Mapping: from Moodle's lessons to PDDL's actions for a given student ?s

<table>
<tr><td colspan="4">Moodle lesson</td><td colspan="2">PDDL action</td></tr>
<tr><td colspan="2">Kind</td><td>dependency</td><td>activity link</td><td>preconditions</td><td>effects</td></tr>
<tr><td colspan="2">L0</td><td>-</td><td>-</td><td>not (L0_done ?s)</td><td>(L0_done ?s)</td></tr>
<tr><td rowspan="2">Lr_i</td><td>(allows to reach a Lf_j)</td><td>L0
(Lr_h, $h \neq i$
or
Lf_k $k \neq j$)</td><td>to Lf_j</td><td>not (Lr_i _done ?s)
(L0_done ?s)
((Lr_h _done ?s)
or
(Lf_k _done ?s))</td><td>(Lr_i _done ?s)
and
(Lf_j _done ?s)</td></tr>
<tr><td>(does not allow to reach a Lf_j)</td><td>L0
(Lr_h, $h \neq i$
or
Lf_k $k \neq j$)</td><td>-</td><td>not (Lr_i _done ?s)
(L0_done ?s)
((Lr_h _done ?s)
or
(Lf_k _done ?s))</td><td>(Lr_i _done ?s)</td></tr>
<tr><td colspan="2">Lf_j</td><td>L0</td><td>-</td><td>not (Lf_j _done ?s)
(L0_done ?s)</td><td>(Lf_j _done ?s)</td></tr>
</table>

In order to clarify these concepts, Fig. 9 shows a real action and a fictitious action in the PDDL domain (which includes parameters, durations, preconditions and effects) of the "Elementary module" structured in Fig. 8.

After the domain is generated, we need to compile a PDDL problem file with the information about the students. In particular, the choices made by the students when executing L0 will be the initial state and goals of the PDDL problem. Let us imagine that four new students, Mark, Laura, David and Polly just took the "Elementary module" and the results arising from the L0's execution are as shown in Table 3. For instance, Mark has a "very bad" initial knowledge and wants to achieve the "almost sufficient" state. L0's results can be translated into the initial states and goals of the PDDL problem, as shown in Fig. 10.

```
(:durative-action L2
:parameters (?s - student)
:duration (= ?duration 40)
:condition (at start (and
        (not (L2_done ?s))
        (L1_done ?s)))
:effect (and
        (at end (L2_done ?s))
        (at end (Las_done ?s))))
```

```
(:durative-action Lvb
:parameters (?s - student)
:duration (= ?duration 1)
:condition (at start (and
        (not (Lvb_done ?s))
        (L0_done ?s)))
:effect ((at end (Lvb_done ?s))))
```

Fig. 9. A real lesson (on the left) and a fictitious lesson (on the right) in the PDDL domain of the "Elementary module".

Table 3
Initial states and goals of the students enrolled in the "Elementary module"

Student	Initial states	Goals
Mark	Lvb	Las
Laura	Lb, L2	Ls
David	Lb, L1, L2	L4, Ls
Polly	Lvb	L5, Ls

```
(:init
(Lvb_done Student_Mark)
(L2_done Student_Laura)
(Lb_done Student_Laura)
(Lb_done Student_David)
(L1_done Student_David)
(L2_done Student_David)
(Lvb_done Student_Polly))
```

```
(:goal (and
(Las_done Student_Mark)
(Ls_done Student_Laura)
(Ls_done Student_David)
(L4_done Student_David)
(Ls_done Student_Polly)
(L5_done Student_Polly))))
```

Fig. 10. Problem PDDL for the "Elementary module".

3.6. Plan generation and visualization

Once created the PDDL domain and problem, it is necessary to use them in order to generate a plan that contains a learning path for each student enrolled in the course. PDDL is a standard planning language and it is supported by many state-of-the-art planners. In consequence, there are many planners that can be used here. In our implementation, we have chosen LPG (http://zeus.ing.unibs.it/lpg/) because it is publicly available and shows a good tradeoff between running time and quality of solutions. But it is important to note that we can use other planners without further modifications. We can see the resulting plan in Fig. 11, where we have omitted the representation of L0, which is obviously taken by every student when s/he defines his/her initial state (background) and learning goals.

In order to create a more efficient visualization of the course, it is necessary that each student only visualizes in the platform the lessons that are included into his/her learning path. In general, if we want to associate a lesson to a specific cluster of students in the platform we have to create a "group", comprehending the set of selected students and a "grouping" associated to the particular lesson. This is a Moodle weird feature. Thus, it is necessary to create an association between the "group" and the "grouping". Generally speaking, a "group" can contain one or more students and a "grouping" can include one or more groups.

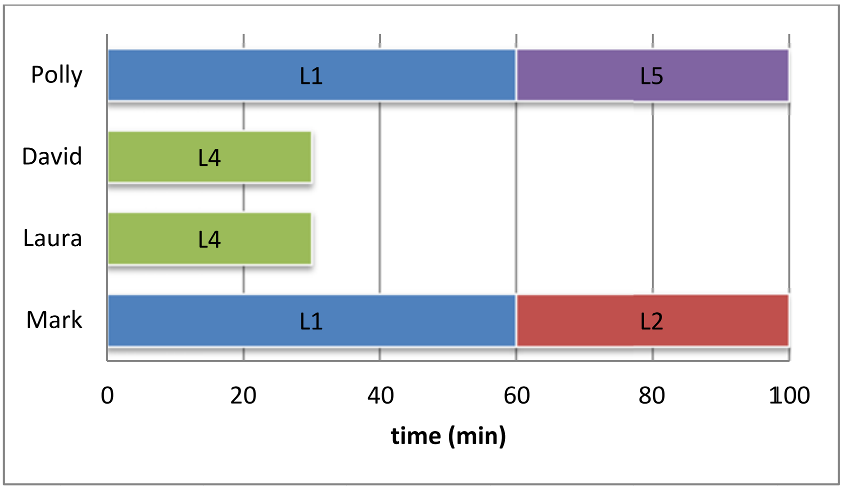

Fig. 11. Resulting learning paths for the "Elementary module".

In order to clarify this general concept, let us imagine that we want to assign to each student enrolled in the "Elementary module" only the lessons that appear in his/her plan (as shown in Fig. 11). The steps to be performed are as follows (see also Fig. 12):

1) We create a "group" for each student (for simplicity, we name each group as the user id of the student in the platform);

2) We create a "grouping" for each lesson;

3) We create the associations between the "groups" and the "groupings" depending on the results of the plan (see Fig. 11). In particular, we associate the "groupings" of the lessons L1 and L2 to Polly's and Mark's "groups", the "grouping" of the lesson L4 to David's and Laura's "groups" and the "grouping" of the Lesson L5 to Polly's "group".

At this point, what we expect is that each student only visualizes the lessons included in his/her own learning path, but a similar result was not obtained by simply using the platform as it was initially conceived. In fact, Moodle (like others LMSs) was initially designed to provide the same material (activities and resources) to all the students enrolled in a course. We found here some limitations concerning the ability to create personalized visualizations within a course (Caputi & Garrido, 2013). We had to do a modification in the code in order to achieve the correct visibility. In particular, as Fig. 13 shows, we make visible for each student only his/her specific learning path within the "Elementary module".

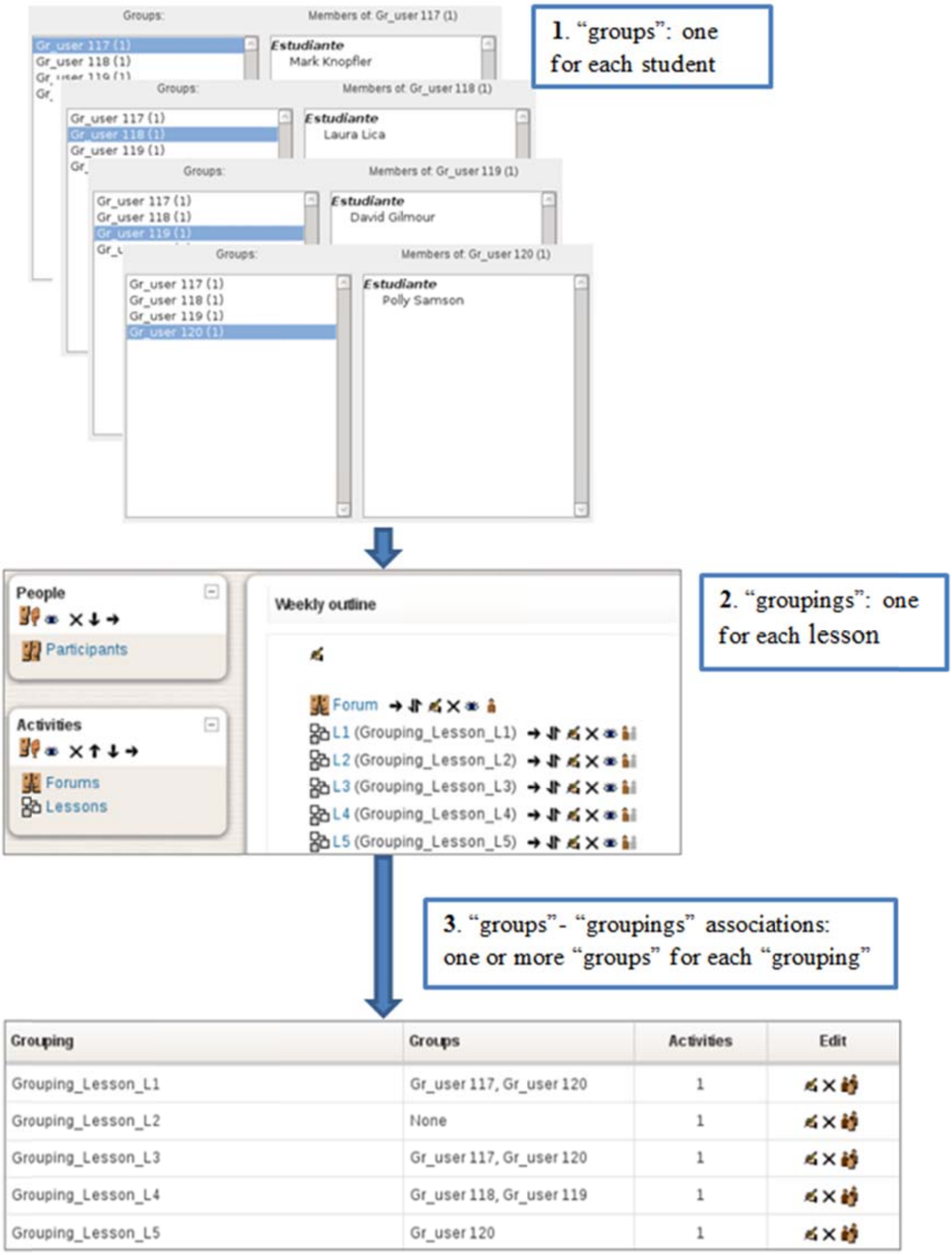

Fig. 12. How to associate activities to students in the "Elementary module" in Moodle.

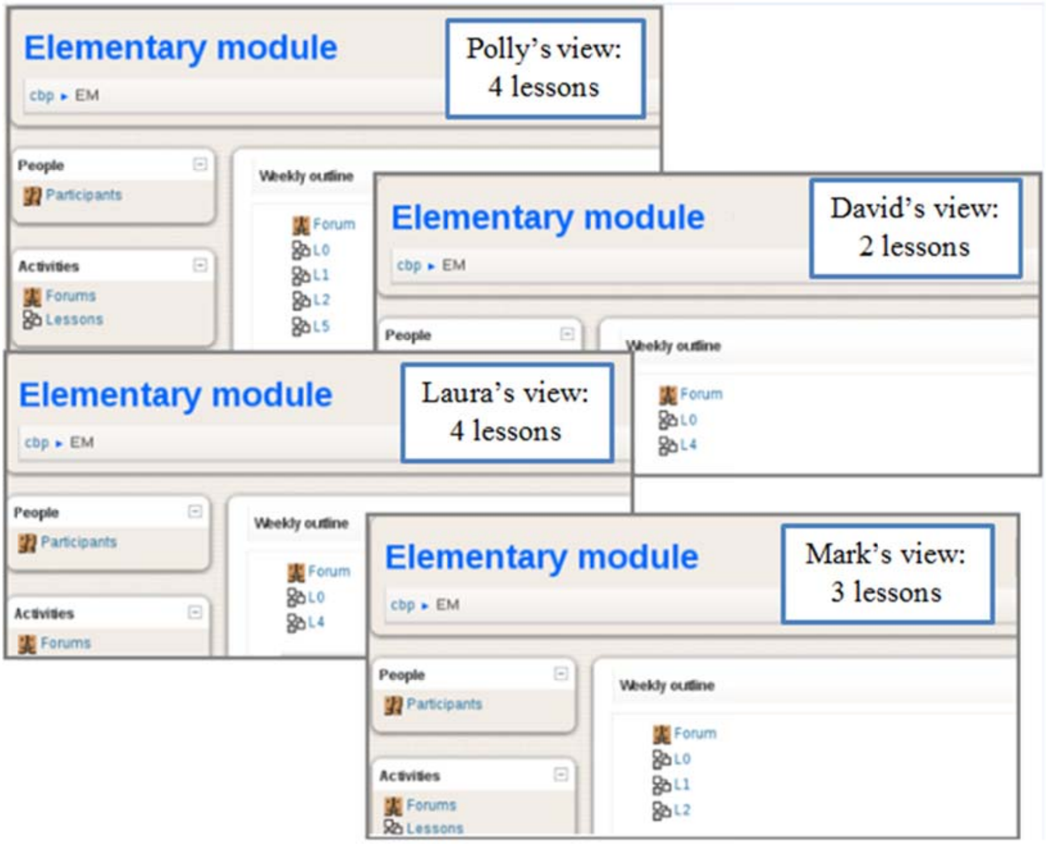

Fig. 13. Personalized visualizations of the "Elementary module" in Moodle.

4. Monitoring

Once the plan and all its content, i.e. a set of lessons and their relationships, are correctly shown to each student, it needs to be executed. By considering our initial knowledge in terms of the students' initial states and goals, we know that the learning path that we offer to each student is the most appropriate to his/her needs. But we cannot be sure that it is executable in its entirety. In fact, it is possible that at some time the expected results do not correspond to the real results achieved by the student (because a discrepancy appears). For example, let us consider Polly's learning path within the "Elementary module". Before executing L2, she has to spend 60 minutes in completing L1. Imagine that she has already spent 60 minutes in performing L1 without terminating it. A similar problem can occur for a variety of reasons: lack of appropriate equipment, loss of time due to external factors, error in the self-assessment of the initial state, etc. In the specific Polly's case, it is possible, for example, that she changed her learning goals over time. Thanks to how we designed the system, Polly (and any student in general) can express a change of the initial state and/or learning goals by simply performing one more time L0.

Thus, together with the planning activity, it is necessary to develop a monitoring activity that allows us to assess if the student's progress remains in line with his/her expected path. The monitoring activity, which is included in our system by simply invoking again the planner whenever it is deemed necessary, is indispensable to eventually (partially or completely) re-plan students' learning paths. Every time we invoke the planner, the lessons already carried out by the student and any changes in his/her learning goals are considered respectively as part of the initial state and goals in a new PDDL problem. This problem, together with the PDDL domain (which can also change if the teacher modifies the course structure), is considered by the planner to generate a new student-oriented plan, which is shown to the student.

5. Experiments

We have first decided to perform a quantitative evaluation to measure the validity and scalability of the system by creating fictitious courses and students. Consequently, we carried out tests on three courses of different sizes: up to 9, 13 and 40 lessons named, "Small", "Medium" and "Large", respectively. We also created 500 fictitious students to whom we randomly assigned their initial states and learning objectives (in terms of real and fictitious lessons), depending on the specific course in question: in particular, up to 6 goals for the "Small" course, up to 10 goals for the "Medium" course and up to 12 goals for the "Big" course. After the automated generation of the PDDL domains+problems, we used two different standard planners to assess the viability of the solving process by current planning technology. In particular, we use LPG (http://zeus.ing.unibs.it/lpg/) and SGPlan (http://www.sgplan.com/) because they have traditionally shown very effective in the International Planning Competitions (IPCs, http://ipc.icaps-conference.org/). We run all our experiments on an Intel Core i5 CPU (dual core 2.27 GHz processor) with 4 GB of RAM. All experiments were censored after 900 seconds.

Fig. 14 and Table 4 show, respectively, the total number of solved plans and the percentage of solved problems by the two planners for the three courses. As can be seen, SGPlan is more effective than LPG in solving problems. Specifically, LPG has some limitations when dealing with courses with more than 100-150 students, which is indeed a promising number. Furthermore, SGPlan shows a very scalable behavior and has little problems in finding plans for all students in the three courses. This demonstrates that current planning technology is sufficient to solve the personalization planning problems we create in our approach.

On the other hand, Fig. 15 shows the average time to solve plans depending on the total number of students. The plots show that LPG is very fast in the problems it manages to solve; LPG takes less than 5 seconds in finding plans for the "Small" and "Medium" courses, even for up to 500 students. SGPlan takes more time, but it also solves more problems. Big courses with 500 students are solved by SGPlan in less than 15 minutes, which is an excellent result. Clearly, personalization of learning paths does not usually need to involve such a high number of students because independent paths can be generated for much smaller groups of students, which means having many different problems but with no more than 20-50 students each. This means our experiments significantly exceed the usual requirements and, therefore, we prove that planning technology can successfully cope with very demanding courses.

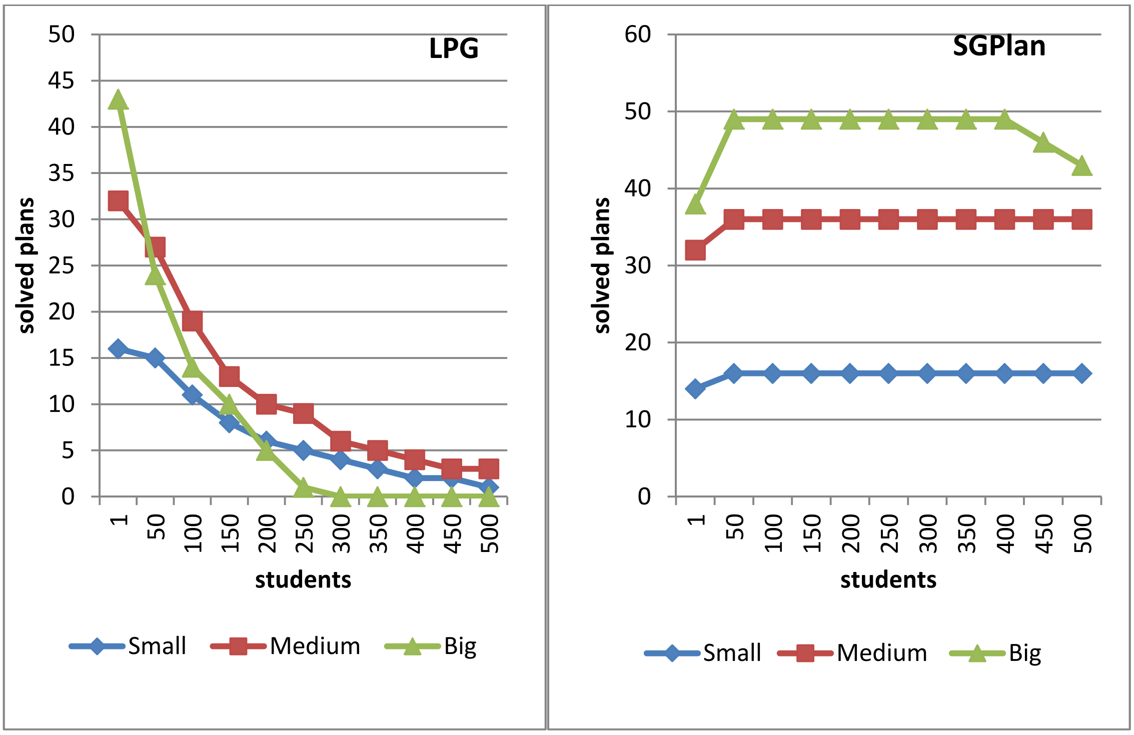

Fig. 14. Number of solved plans by LPG and SGPlan.

Table 4
Percentage of problems solved by LPG and SGPlan for the three courses

Courses	# generated problems	LPG	SGPlan
Small	176	41.48 %	98.85 %
Medium	396	34.08 %	98.99 %
Big	539	19.11 %	98.32 %
Total	1111	27.98 %	98.65 %

Our second experiment focuses on a qualitative evaluation for planning e-learning contents. We have performed such an experimental evaluation by means of opinion questionnaires answered by a group of 10 teachers and 10 students to assess the consistency of the planned contents with respect to the course objectives, the quality of learning paths, their size/duration and their adequacy to the particular profiles. We structured a questionnaire for a qualitative evaluation of an AI course, as shown in Table 5. In particular, the questionnaire was divided into 3 blocks concerning, respectively, the course contents and structure; the teachers' opinion on the learning contents, i.e. the elements used to define the course; and the students' opinion on the course. Each question had 5 possible answers: Very little, Little, Neutral, Much and Very much.

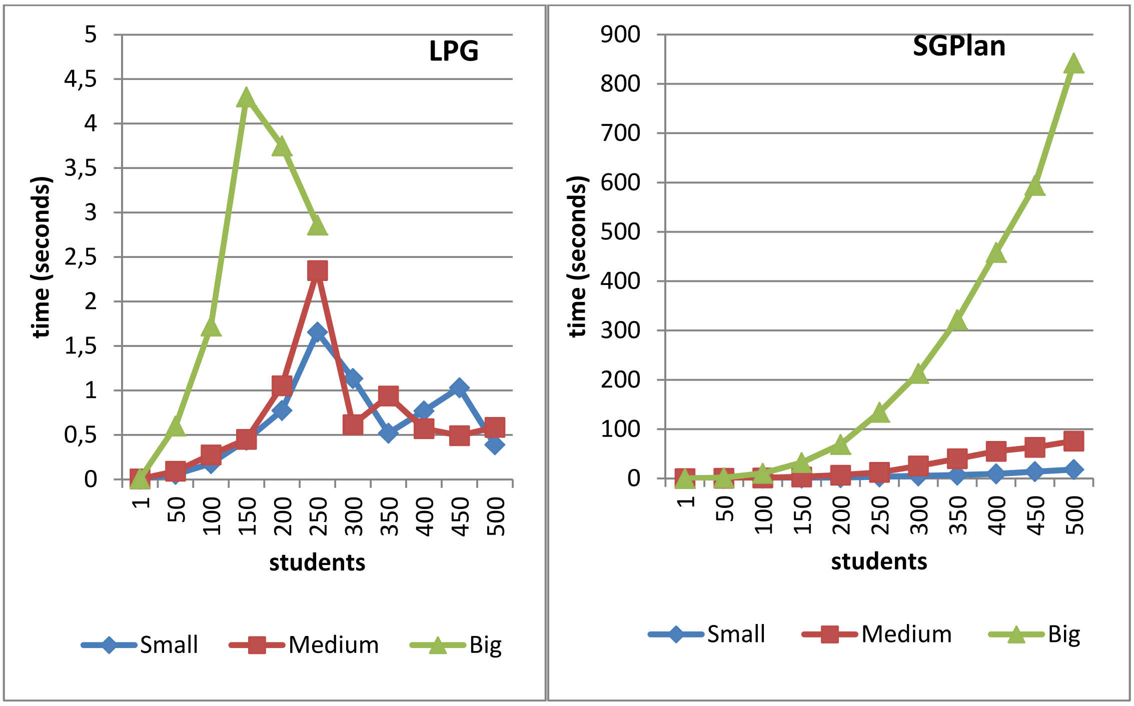

Fig. 15. Average time (in seconds) to find plans by using LPG and SGPlan.

Table 5
Questionnaire for a qualitative evaluation of an AI course

Block	Questions
1. Course contents and structure	Q1. Is the sequence of contents consistent with the objectives of the course? Q2. Is the size (number of learning contents) of the course appropriate? Q3. Is the duration of the course appropriate? Q4. Do you think the learning path and contents are adapted to the student's profile?
2. Teachers' opinion on the learning contents	Q1. How much experience do you need to deal with these learning contents? Q2. How much planning background is necessary? Q3. Do you consider this approach useful? Q4. Would you recommend this approach to other lecturers?
3. Students' opinion on the entire course	Q1. Do you find e-learning as a positive and motivating experience versus traditional teaching? Q2. To which extent did the course fit your needs and constraints? Q3. Would you suggest some changes in the course structure? Q4. Would you recommend this approach to other students?

Fig. 16 shows the summary results for each block of questions. It is possible to observe that teachers very much agree with the paths in terms of their form, size and adaptation to the students. However, some teachers recognize that it is hard to evaluate how learning paths fit to individual profiles. In short, teachers appreciate that kind of personalization, but in some cases they cannot reasonably answer why. On the other side, students find the experience highly positive because they feel the learning path is very student-oriented, and not the same path for everybody. It also helps learning in a more personalized way, which subsequently could improve the learning process and, eventually, the final scores. All in all, that contents personalization is highly appreciated and both teachers and students believe this approach is viable and very promising.

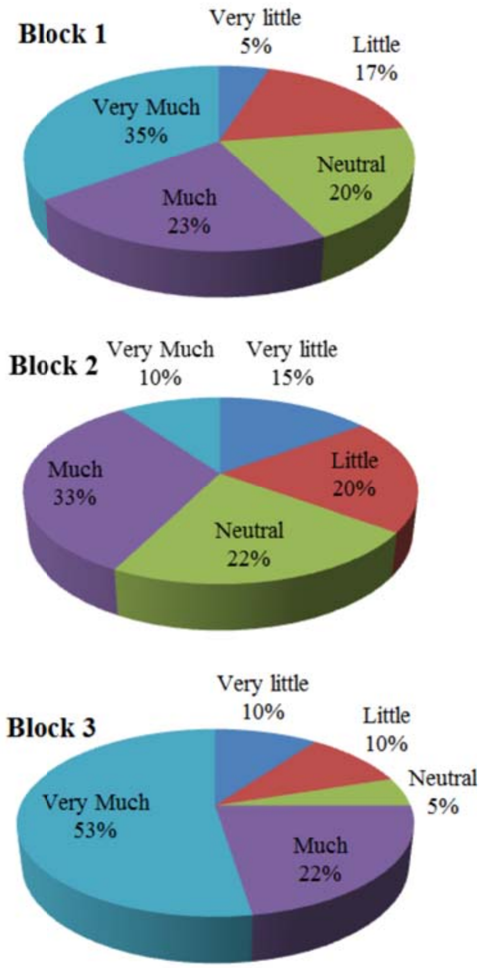

Fig. 16. Percentages of given answers for our qualitative evaluation

As a summary, it is important to highlight that these tests require more teachers and students to be fully conclusive. However, testing our approach to a larger extent is somewhat difficult as it requires the strong collaboration of teachers, students and the correct definition of complete courses in Moodle which, unfortunately, are not always freely available. Additionally, we have found that some teachers are reluctant to participate in this type of entirely student-oriented planning approach because it means to change their *inertia* in their way of teaching: they prefer using their experience as human planners rather than using automated intelligent planners. As part of our ongoing work, we are currently designing and implementing a Moodle course on Physics that deals with a larger number of teachers and students.

6. Conclusions

In this paper we have faced the learning paths' customization from an AI planning perspective to a LMS. The core about using planning technology by compiling a PDDL model based on the course definition (activities and their relationships) and students' features (profiles and background) is independent from the LMS used, and it is applicable to any LMS. In our work we adapt this idea to Moodle. Moodle is a LMS that allows us to manage and to deliver courses' material in a simple and functional way.

In order to get the maximum benefit from Moodle, we provide the design and a way to monitor student-oriented learning paths (according to students' initial background and learning goals) to offer the best contents to the adequate person. In particular, by using a standard planner we generate a plan, i.e. a learning path for each student, and we monitor the plan's execution by simply invoking the planner as often as necessary.

We had to solve some implementation limitations in Moodle because this platform, as well as other LMSs, is not originally designed to provide students with personalized contents. In particular, we have faced problems concerning the impossibility of creating complex relationships between courses' activities and the scarcity of information about the students' profiles (background and learning goals) insertable into the platform. Another limitation that we had to solve concerns the impossibility to create separate course's views only related to the content of specific learning paths. Consequently, there are some technical issues that are specific and, in some sense, fully Moodle dependent, such as the way the learning activities and "activity links" (i.e. the precedence relationships) are modeled, or the access to the particular database schemata of Moodle, which is different from other LMS.

In order to be able to adapt the planning activity to the tools that Moodle provides we have developed a number of tasks that contribute with:

1) A knowledge engineering mapping of lessons in Moodle to actions of a PDDL domain. Also, we have created some dummy lessons to model students' profiles (in terms of initial background and learning goals) to be translated into initial states and goals of a PDDL problem.

2) A PDDL standard model to be used by a PDDL-compliant planner. This permits us to easily generate customized learning paths in Moodle.

3) A seamless integration to show only the adequate contents to each student in Moodle.

4) A monitoring activity, indispensable to eventually re-plan students' learning paths.

5) Proof of the scalability of the system. Our tests have shown that for a reasonable time it is possible to find plans that include learning paths even for a large number of students and for a large number of lessons.

As part of our ongoing work, we are working on the learning paths' customization in Moodle by implementing additional real courses to be taken by a large number of students. The main advantage of our system relies on its flexible design, so that it can be adapted in a straightforward way to any type of educational content and, therefore, be used by a wide variety of users. All in all, the idea of generating a PDDL model from e-learning aspects, using planning and giving feedback to the LMS to facilitate individual learning paths to the students is generic, and it can be implemented on top of any LMS by making some technical adaptation depending on the specific LMS's features.

Finally, there are two lines open for future work. First, although the main objective of our current approach is to manage the different curricula by simply using a LMS, we want to investigate the possibility to extend our idea to be included, for example, into a curricula management system, which integrates other tools to support e-learning. Second, we want to analyze the possibility to allocate tasks in time if the presence of some practical lessons becomes necessary (teachers giving face-to-face lessons in classrooms, exams in real labs, or any type of task that requires the use of shared and limited resources). This possibility can be easily extended to the planning approach by including intelligent techniques on scheduling reasoning, though it would require more work for the contextualization in Moodle.

Acknowledgements

Valentina Caputi thanks the European Commission, the European Social Fund, and the Regione Calabria for financial support of her PhD fellowship and funding for her stay in Valencia.

This paper is co-funded with support from the European Commission, the European Social Fund and the Regione Calabria. The paper was also partially funded by the Consolider AT project CSD2007-0022 INGENIO 2010 of the Spanish Ministry of Science and Innovation, the MICINN project TIN2011-27652-C03-01 and the Valencian Prometeo project II/2013/019.

The European Commission and the Regione Calabria disclaim any responsibility for the use that may be made of the information contained in this publication.

References

Al-Ajlan, A., & Zedan, H. (2008). Why Moodle. 12th IEEE International Workshop on *Future Trends of Distributed Computing Systems*.

Baylari, A., & Montazer, Gh. A. (2009). Design a personalized e-learning system based on item response theory and artificial neural network approach. *Expert Systems with Applications*, 36(4), 8013-8021.

Brusilovsky, P., & Vassileva, J. (2003). Course sequencing techniques for large-scale Web-based education. *International Journal Continuing Engineering Education and Lifelong Learning*, 13(1/2), 75-94.

Caputi, V., & Garrido, A. (2013). Experiences on using Intelligent Planning for curriculum personalization in Moodle. EDULEARN13 Proceedings, 168-176.

Chang, T. Y., & Ke, Y. R. (2013). A personalized e-course composition based on a genetic algorithm with forcing legality in an adaptive learning system. *Journal of Network and Computer Applications*, 36(1), 533-542.

Chang, Y. C., Hsieh, S. M., & Li, J. W. (2010). Application of the genetic algorithm in customization personalized e-course. *In System Science and Engineering (ICSSE), 2010 International Conference* on (pp. 221-226). IEEE.

Chen, C.M. (2008). Intelligent Web-based learning system with personalized learning path guidance. *Computers & Education*, 51(2), 787-814.

Chen, C.M., Lee, H.M., & Chen, Y. H. (2005). Personalized e-learning system using Item Response Theory. *Computers & Education*, 44(3), 237-255.

Dorça, F.A., Lima, L.V., Fernandes, M.A., & Lopes, C.R. (2013). Comparing strategies for modeling students learning styles through reinforcement learning in adaptive and intelligent educational systems: An experimental analysis. *Expert Systems with Applications*, 40(6), 2092-2101.

Garrido, A., Fernández, S., Morales, L., Onaindía, E., Borrajo, D., & Castillo, L. (2013). On the automatic compilation of e-learning models to planning. *The Knowledge Engineering Review*, 28(2), 121-136.

Garrido, A., & Onaindia, E. (2013). Assembling learning objects for personalized learning. An AI planning perspective. *IEEE Intelligent Systems*, 28(2), 64-73.

Ghallab, M., Nau, D., & Traverso, P. (2004). *Automated Planning. Theory and Practice*, Ed. Morgan Kaufmann.

Hewson, C. (2012). Can online course-based assessment methods be fair and equitable? Relationships between students' preferences and performance within online and offline assessments. *Journal of Computer Assisted Learning*, 28(5), 488-498.

http://www.moodle.org/.

http://docs.moodle.org/.

http://docs.moodle.org/all/es/Lecciones/.

http://zeus.ing.unibs.it/lpg/.

http://www.sgplan.com/.

Idris, N., Yusof, N., & Saad, P. (2009). Adaptive course sequencing for personalization of learning path using neural network. *Int. J. of Advances in Soft Computing and its Applications*, 1(1), 49-61.

Kaklauskas, A., et al. (2013). Recommender System to Analyze Student's Academic Performance. *Expert Systems with Applications*. http://dx.doi.org/10.1016/j.eswa.2013.05.034

Kontopoulos, E., Vrakas, D., Kokkoras, F., Bassiliades, N., & Vlahavas, I. (2008). An ontology-based planning system for e-course generation. *Expert Systems with Applications*, 35(1-2), 398-406.

Laurillard, D., Charlton, P., Craft, B., Dimakopoulos, D., Ljubojevic, D., Magoulas, G., Masterman, E., Pujadas, R., Whitley, E.A., & Whittlestone, K. (2013), A constructionist learning environment for teachers to model learning designs. *Journal of Computer Assisted Learning*, 29: 15-30. doi: 10.1111/j.1365-2729.2011.00458.x

Martín-Blas, T., & Serrano-Fernández, A. (2009). The role of new technologies in the learning process: Moodle as a teaching tool in Physics. *Computers & Education*, 52, 35-44.

Papanikolaoum, K.A., Grigoriadou, M., Magoulas, G.D., & Kornilakis, H. (2002). Towards new forms of knowledge communication: The adaptive dimensions of a Web-based learning environment. *Computers & Education*, 39, 333-360.

Romero, C., Ventura, S., & García, E. (2008). Data mining in course management systems: Moodle case study and tutorial. *Computers & Education*, 51(1), 368-384.

Tan, X. H., Shen, R. M., & Wang, Y. (2012). Personalized course generation and evolution based on genetic algorithms. *Journal of Zhejiang University SCIENCE C*, 13(12), 909-917.

Thyagharajan, K. K., & Anbumani, B. (2009). An Enhanced Authoring and Context-Aware Presentation Model for Web-based Education. *International Journal of Computer Science, Systems Engineering and Information Technology*, 2(2), 191-195.

Thyagharajan, K. K., & Nayak, R. (2007). Adaptive content creation for personalized e-learning using Web services. *Journal of Applied Sciences Research*, 3(9), 828-836.

Xenos, M. (2004). Prediction and assessment of student behaviour in open and distance education in computers using Bayesian networks. *Computers & Education*, 43(4), 345-359.