New Operating Tools in the Easy Learning On-line Platform

Radu Rădescu, Adrian Șișu, Raul Tudor

Polytechnic University of Bucharest,
Applied Electronics and Information Engineering Dept.
1-3, Iuliu Maniu Blvd., Sector 6, ROMANIA
E-mail: rradescu@atm.neuro.pub.ro

Abstract

The present paper deals with the new tools introduced in the Easy Learning platform. The first layer of this architecture implemented by the Symfony tool is the model layer, based exclusively on an abstract version of the database and on the users’ access to data and recordings. Following this model, the database was conceived and designed as the core of the whole project. The interaction tutor-student using the database is considered in all its statistical aspects. A conclusive example is given in the case of the testing module, emphasizing the advantages introduced for all the actors involved in the e-learning process.

Keywords: eLearning platform, Symfony architecture, Modules design

Introduction: A Short History of the Easy-Learning Platform

In 2004, the first version of the Easy-Learning platform emerged. It was a new interaction method between a tutor and his students, even though it was limited to managing the laboratory class book. Thus, every student was able to see all obtained grade points, throughout the activity.

By default, this first version was somewhat limited. During the last five years, the platform suffered multiple adjustments, taking into consideration the always-emerging necessities, having good but also bad repercussions. The very first restructuring gave birth to the three known interfaces: Administrator, Tutor and Student. This structure helped limiting tasks to every type of user. The changed proved to be logical and well received.

The Administrator interface took up some of the tutor’s tasks, such like creating series and groups of students, along with populating them.

The Tutor interface gave up the administrative tasks, making life easier for every tutor (class structure, class books, timetables became the main attributes of a tutor).

The Student interface aside from viewing class books and timetables received the main role of sustaining on-line tests, taking laboratory classes etc. This way the student had permanent access to the platform’s eLearning content.

Disadvantages of the Existing Platform And the Saturation Point

As any other software product out there, Easy-Learning has a life cycle, starting with the concept and ending with the saturation point. Now, the main drawback is the lack of
homogeneity. Its backbone may not be the most suited one, so that in many cases it may lose its purpose: helping. This is due to the numerous patches made by different programmers (mostly students). It has come to a very strenuous code, any error being almost impossible to debug.

The main database cannot handle all the requests, having a much to fork, non-unitary structure. This might be proved through the power of example, because the most harmed modules are the ones on which it is acted from multiple interfaces.

The logic behind the testing module proved not to be the most suited one, on a long term. A test is based on the questions inserted by a tutor, selecting a given number of them. However, there is a limit, not imposed, so that the questions that have overcome this limit would never be taken into consideration for a test. This is also due to the overcharge of the database, with it's too many unstable connections.

Other examples are the class book and classes modules. Any class has a structure composed by the grade points of all the afferent activities. However, for the old platform, this structure was only configurable once. Any edit on it during the academic year would turn all the activities inside out. Editing a structure is necessary because of the grade points variations or any other time factor.

In addition, all the formulas that were calculating grades aggravated a tutor’s activity, being uneven. Teaching activities had different formulas, using different percents. The tutor had to remember if for the activity X the grade was composed of points of percent of the final grade. Therefore, students viewed mistakes and so there was place for confusions.

Problems may continue with promoting series, groups and students from one year to another, this being impossible with the old platform. The tutor had to recreate them after every academic year. A counter was a solution but it was not implemented. The new Easy-Learning platform, along with its new tools, tries to solve all these problems and even more, to extend it’s functionality using the model of renowned universities.

**New Modules of the Tutor Interface**

The tutor interface is the main relationship between a tutor and the platform. Using it may keep a strong link with the students.

It provides options for editing personal data of the authenticated tutor along with classes, timetables, documents and announcements. The main attributes of this interface are concentrated in the class book like, tests, questions and categories modules. A more detailed description for them is available next.

These modules were a challenge for the Symfony framework’s architecture. Its main system, admin generators for every table of the database proved to be insufficient. Therefore, we needed to manually build them using forms, validations, actions, editing and templates. The base classes of the framework and the tree-like architecture of the project took the main stage.

**The Class Book Module**

Through this module, the tutor may manage grade points for every one of his students. As it may be seen in Figure 1, the list action allows viewing a complete list of grade points
for every existing student in the database. The form contains, along with the actual list, an active filter only if there are records. If there is not any available evidence, the filter will not be listed.

![Available Class Book Lists](image1)

*Fig. 1. Available Class Book Lists*

Available actions are: edit (📝) and delete (🗑️) which apply to the current record. In order to prevent accidents, the delete action is provided with a confirmation from the user. The *Create new evidence* button redirects the user to a new form, where new class books can be made (see Figure 2). At this moment, the user can return to the list or proceed with adding new evidence.

![Creating A New Evidence](image2)

*Fig. 2. Creating A New Evidence*

The form contains a combo-box populated with the list of students, edit-boxes for every grade point and filters by class and student. Here comes the first innovation of this module: the tutor will insert, for every type of activity, regardless of it’s share, a score of maximum 100 points. This score is taken by the general formula, based on the actual
shares of each activity and converted to a final score of maximum 100 points too. All this
formulae and conversions are invisible to the user to ease up the task. In the student
interface, he/she will see the real score detailed for every activity and as a final grade.

The Laboratory Class Book Module
This module was one of the most difficult to implement (along with the test module),
using the most complex tools available. Here, the tutor will be able to manage all the
grade points of laboratory-like activities for his students.
The action list allows viewing all available scores, and filtering them by discipline.
The same rule with the active/inactive filter will apply. In addition, the same actions are
available, as seen in Figure 3: edit (_edit) and delete (_delete). The Create button will redirect
the user, as in the previous case to the proper form. There are a few steps to be followed.
The first one is choosing a class and hitting the add button. A new form will open (Figure
3), the one for the actual creation of the evidence.

![Fig. 3. Creating A New Laboratory Evidence](image)

Due to the database query, posted in actionClass, the logged in tutor, will only have
access to his classes and students, thus properly limiting activities. This selection applies
to all Tutor interface modules.

Through this form, a student might be chosen so that he will have evidence attributed.
With a simple click on the attendance check box, the required attendance ill is added.
Along with listing the right date and time, these are automatically taken from the
system’s clock, with the help of a JavaScript. Annulling the attendance requires just
another click. If the selected laboratory had a test attached, the tutor may grade it using the combo-box, which has scores between 0 and 100. This score will be part of the laboratory grade formula, based on the appropriate share of the laboratory tests activity.

The last section of the form, other grades, contains two combo-boxes with scores also between 0 and 100 for the essay and final test activities. The same formula and shares principles apply. The final test is the most complex because when a student finishes it, after grading and finalizing it, the final grade will automatically fill the right field, so that the user is exempt of writing it by hand. For a quicker search of the students in the lower side of the form, there is a filter by groups.

Tests/Questionnaires, Questions and Questions Categories modules

These modules will be treated together because they are interdependent, question categories being the base of the ahead thought theory for building tests.

In order to describe better the mechanics of these modules we will refer to the crowds’ theory. Suppose a laboratory final test with 20 questions. The activity was structured on multiple sessions, every session having it’s own laboratory platform. In order to include the entire class subject, the test must contain questions from every platform. The diagram in Figure 4 represents graphically this procedure.

Same as for the other modules, there is a list action for every available test of the authorized tutor, sorted by classes. This time, the actions column is far more complex. It contains the usual edit (แก้ไข) and delete (ลบ) buttons, but also ones for listing questions (รายละเอียด) and grading the test (ติเตาะว์).

The Add test form becomes available after acting on the Add test/questionnaire button. This builds the skeleton of a test, to be modeled later using the earlier presented procedure. Multiple variables must be selected: class, type, name, length and number of questions. It should be mentioned that a test becomes available for a student only if the active status is selected. After setting the details, the user may create question categories.
The categories module is specially designed for this purpose. The name of a category is attributed to an indicator number used in allocating questions.

When categories have been added, the user may return to the test module where he can build questions for the test, sorted by his own categories. This may be done with a click on the List questions button (Fig. 5)

![Fig. 5. Question List For The Current Test](image)

The Create a new question form uses tools that were the hardest to implement and develop, due to the complexity and multiple variable cases. Thus, the user will run through a series of steps, one dependent of the previous. He will select the type of the question from a combo-box. This type may be: text, unique answer, and multiple answers. When this is finished, clicking on the button Choose will open a new form, specific for the selected type. Supposing this was Multiple answers, the next step is selecting the number of options and the category.

The last step is creating a statement and possible answers. In addition, the tutor will check the right answer(s) in order to grade it later (see Figure 6).

![Fig. 6. Forming A Statement And The Possible Answer](image)

Now the question can be saved and the procedure will resume in the same manner for every question. When the test was populated with all the wanted questions, it will become
available to the student that will sustain it (to be talked about later on). When he will have finished, it will be available for grading by the tutor. This is what the grade button (اظهار نقاط) is for.

The grade test form contains all the questions for the specific test along with the answers provided by the student. A score will be selected for every question, between 0 and 100, with a 10 points step, so that every question is graded percentage-like, the final grade containing all the scores. Once all the scores will have been finalized, the laboratory grade will be set in the grades evidence module. This time too, conversions are invisible, so that the final score is actually a grade (see Figure 7).

Fig. 7. Finalize Grading A Test

Conclusions
The Easy-Learning platform may be considered a great way of real-time distance interaction, between tutor and student. In order to describe better this whole process, based on the diagram below, we will use the types of access to the database for the two types of users. The 90% true case is:

- the tutor will access the database by update, insert, delete, select actions, so mostly adding and deleting records that the student may use.
- the student will use ONLY select-like actions to view records. He will almost never alter the records structure because permissions do not allow him to.

The above rule is 90% valid as there is one exception: the tests module. Here, the 2-way interaction contains acting on the database by both ends. There are six steps building the procedure for sustaining a test, from its creation to listing the obtained grade:
1. The tutor builds a new test inserting it in the database;
2. The student sees a new test and opens it in order to take it;
3. The student takes the test adding results in the database;
4. The tutor obtains the answers in order to grade them;
5. The tutor grades the test and inserts the grade;
6. The student sees his grade.

Taking into account the above description it is clear that the student editing access is extremely restricted. Even in the test case. For the evidence modules, the objective was to ease up as much as possible the work of a tutor, along with the student understanding the grade standards and detailed scores. This way, a tutor will grade of maximum 100% for every activity, regardless of importance or share. In addition, graded activities may be changed at any point in time, a missing aspect of the old Easy-Learning platform. The advantage also applies to the student, as presented throughout the article. In the end, it is a known fact that the purpose of Easy-Learning is to help and not burden.

REFERENCES