An Online Dynamic Examination System (ODES) based on Open Source Software Tools

George F. Fragulis*, Lazaros Lazaridis*,
Maria Papatsimouli* and Ioannis A. Skordas*
*Laboratory of Robotics, Embedded & Integrated Systems
Department Of Electrical and Computer Engineering
University of Western Macedonia, Kozani, Hellas

I. Application description & Documentation

Our proposed plugin application named (ODES) is developed for the Wordpress platform. Through this specific plugin the roles of the teacher and the student are created. The teacher inserts either multiple choice or long answer questions (essays) and can create exams. In each exam the teacher defines only the importance (weight/score) that each question will have (including negative values if needed).

The exam generation is based on a mixing of long answer and multiple choice questions which are selected in order to be displayed according the specification that entered by the teacher . The selection of the exam questions is done randomly. In the case of multiple choice questions, both questions and answers of each one are randomly displayed for different users. Multiple choice questions are automatically corrected by the system. On the other hand, there are long answer questions which are also displayed randomly for different users but the final score is given by the teacher himself.

By the time the exam is completed, the teacher is asked to check the long answer questions (if any) so as to give the grades for them. The final exam score(grades) appears automatically in the system for all students that take the exam with time stamp for each one of them.

A. Plugin installation and activation

Like most Wordpress functions, the plugin installation does not require any special knowledge on programming as the only thing that is required is the connection to the control panel and then the plugin is inserted (Plugins -> Add new). In this case all the available plugins can be viewed and the selected one is loaded. The downloading

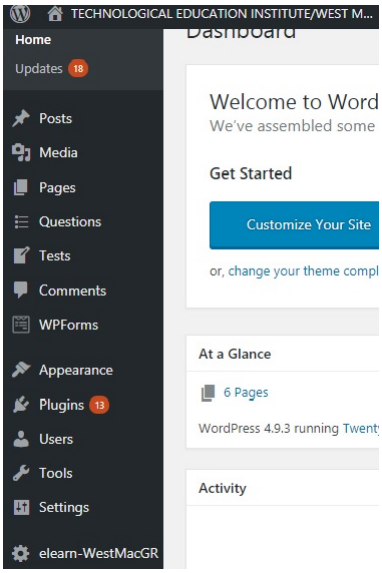

Fig. 1. Application options have been added in the menu

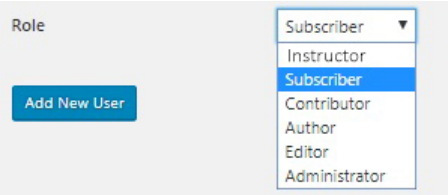

Fig. 2. Adding the Instructor user entity

procedure can be done with the use of any file transfer program. The uploading of plugin can take place in a zip form in the plugin's folder (/wp-content/plugins/) and it will be loaded automatically by Wordpress. Finally, the activation must be done manually.

Initially in the administration menu (Figure 1), there are three options the "Questions", "Tests" and "elearn-WestMacGR".

- A new user entity can be created with the name "instructor" (Figure 2) that has administrator rights.
- In the database, a new table is added with the name wp_elearn_results in which the students' answers will be stored (see Figure 3).

B. Application settings

The application enables us to select in which page of our website we want all the exams to be displayed. From the popup menu we can select one of the pages that exist in the Wordpress and is defined as exam page (Figure 4).

The page settings can be found in file settings.php via Wordpress function add_menu_page

C. Questions

Questions are an custom post type which is defined in cpt-erotisi.php file by making use of register_post_type Wordpress function. Even though questions are posted they cannot be seen by anybody because during their definition the *public setting* is turned to false. They are only available through the control panel.

Every question belongs to one or more question categories that are also defined by cpt-erotisi.php file and it is a classification that is defined through register_taxonomy Wordpress function.

By pressing the menu "Questions" all the stored questions are displayed in a table where the "Title", "Type", "Category" and "entry date" of each question are viewed (Figure 5).

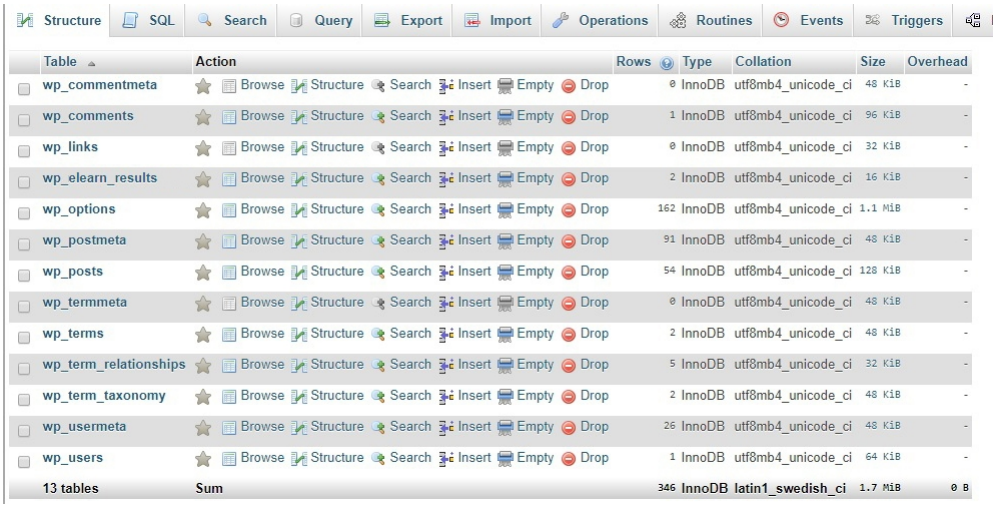

Fig. 3. The table wp_elearn_results of the database

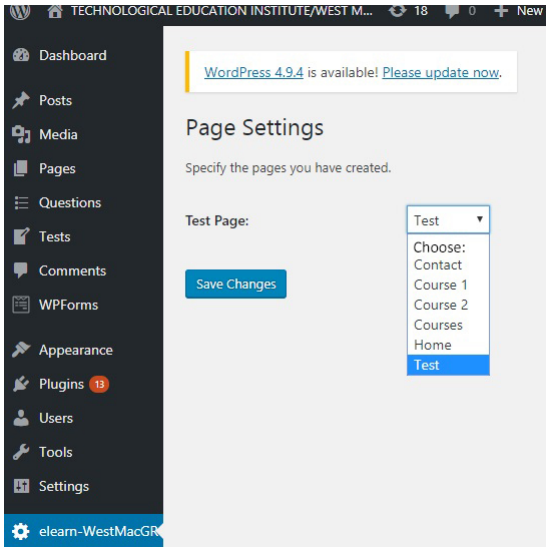

Fig. 4. Application settings

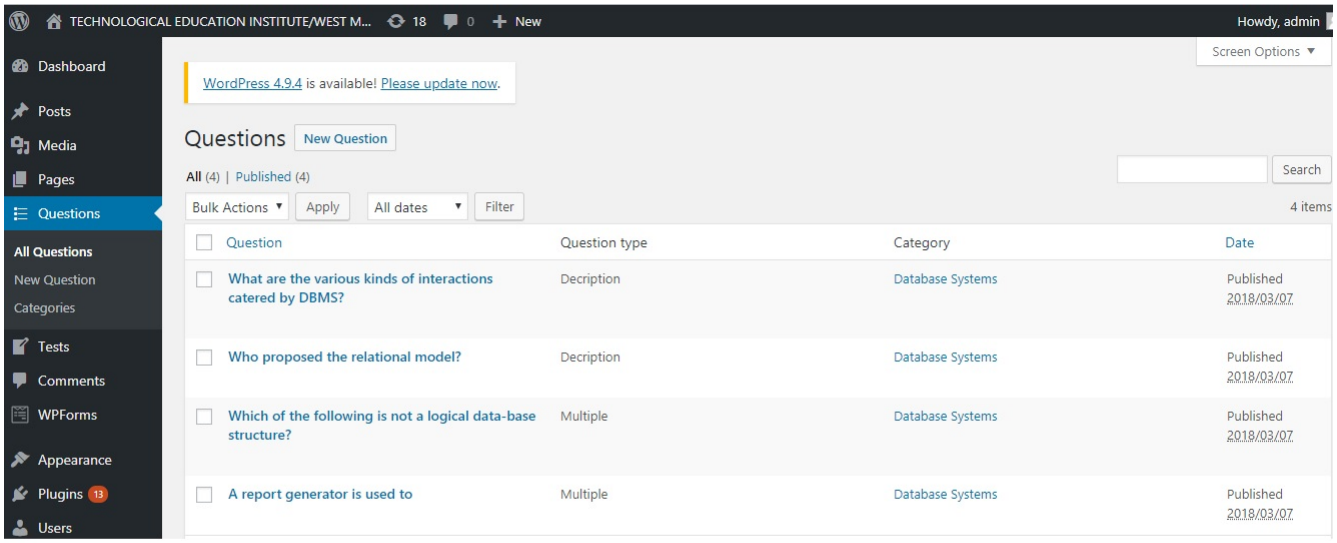

Fig. 5. Custom post type questions

1) Question adding: In order to add a new question we need to select the submenu “New Question”. Then a form will be displayed for completion (Figure 6). This form is almost the same with the standard form that exists in the Wordpress in order to post an article or a page.

In the form there are the following fields :

- Title
- Question description
- Question options
- Categories
- Publication

In the Title field we complete the main question that the student must answer.

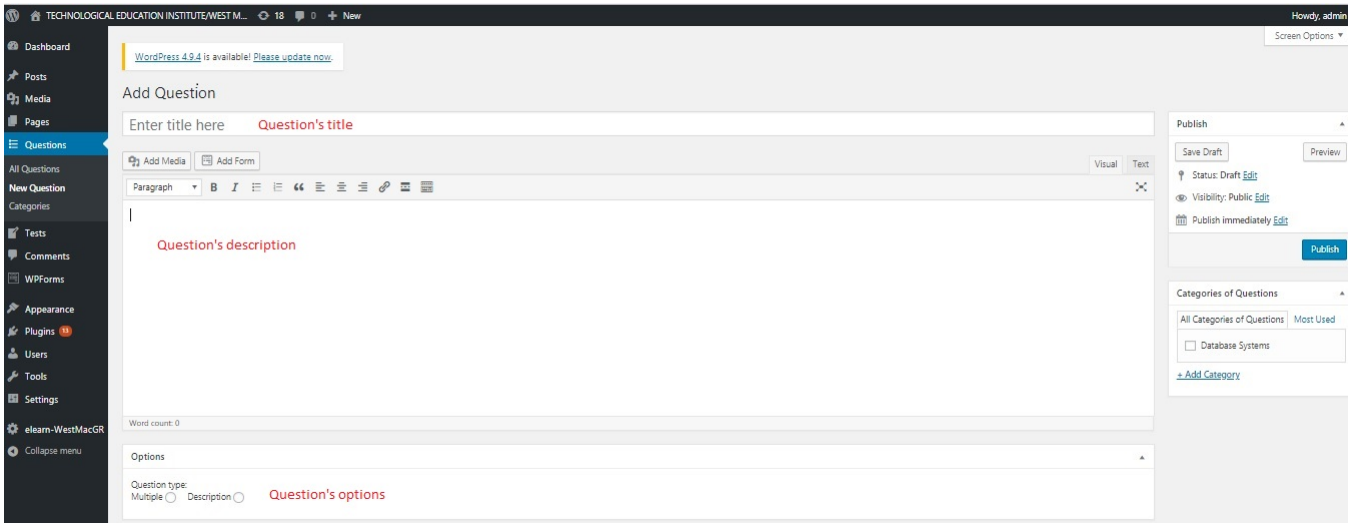

Fig. 6. New question adding

In the Description field the general instructions of the question are optional. This field can accept formatted text as well as multimedia files (images, video, etc.).

In the Categories field (Figure 7) we can select in which of the existed categories or

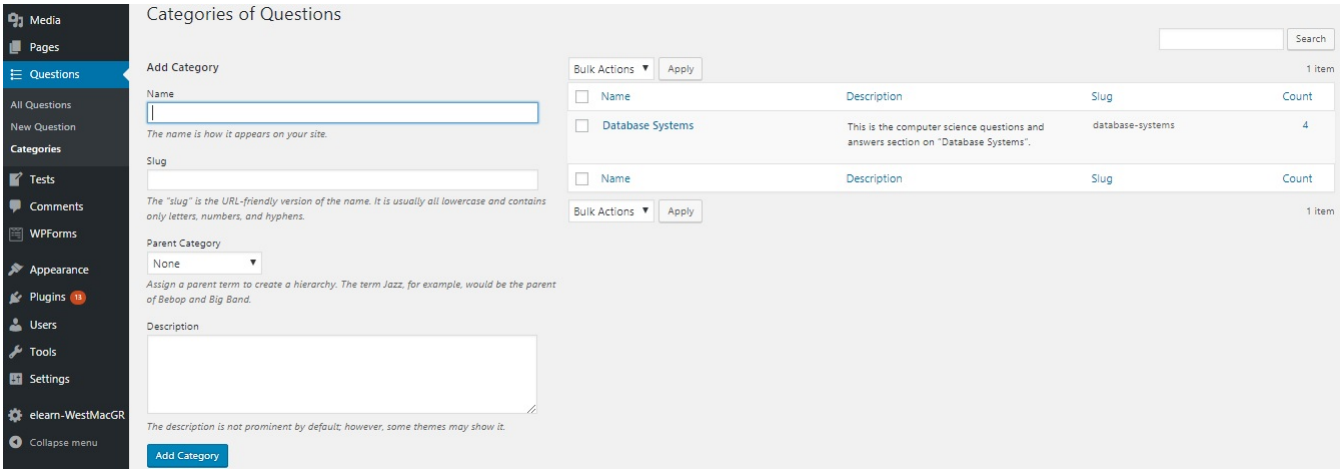

Fig. 7. Question category adding

sub-categories the question belongs. In the case we want to add a new category we must press the link "+ Add category". In the first field we enter the title of the category, in the popup menu – parent category – we select if the category will be a sub-category or not and we press the button "Add category"as in Figure 8.

In the "Publish" field we can publish the question or store it in order to change its settings later.

Finally, in the "Options" field we define several question elements that unlock depending on the type of question that we want to add.

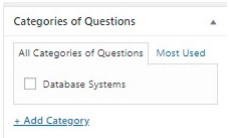

Fig. 8. Quick question category add

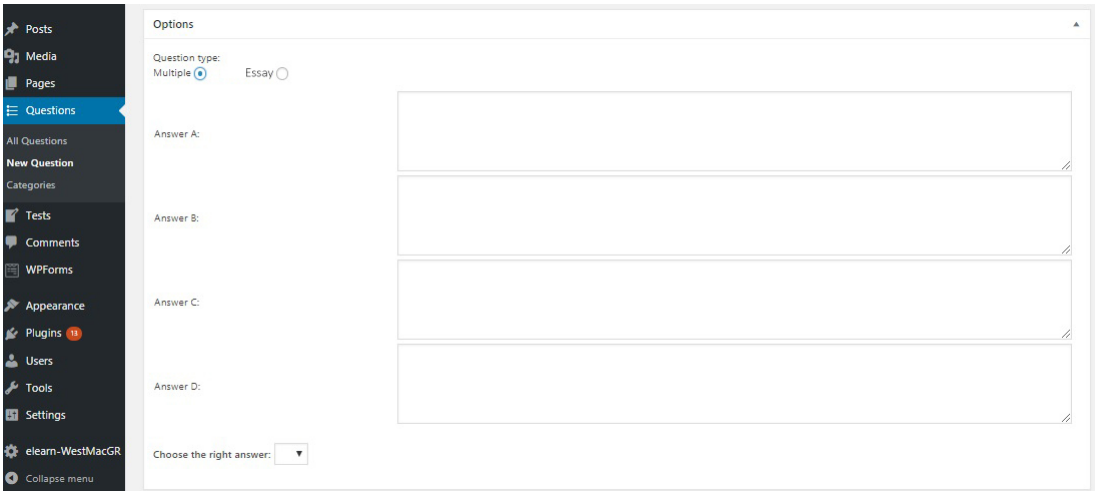

Fig. 9. Multiple choice questions options

2) Question options: Initially in the "Option" field we' re required to select if the question is multiple choice or long answer. If we select "Description" then some necessary extra options are displayed automatically. The swapping is done via jQuery code dynamically.

As shown in Figure 9, if we select multiple choice questions in the "Question type" we have to define the four possible answers and then which one is the correct. Finally, we can store or publish the question.

3) Possible Errors during publication: In case the question publication takes place without having chosen its type, then the system warns us as shown in Figure 10.

4) Question categories: In the third sub-menu of the menu "Questions", there is the question category management page (Figure 11). We can see all added categories and from the left page side we can add new ones. Also, we can edit each category separately (Figure 12).

D. Exams

Exams are, like questions, an adapted post type and are defined in the cpt-diagonisma.php file. They are accessible to administrator/teacher user group in contrast with the questions and have their own properties.

By pressing the menu "Tests" all the stored exams are displayed in a table where the Title and Entry date are viewed. Furthermore, the sub-menus "All Tests", "New Test" are viewed as well (Figure 13).

1) Adding Exams: In order to add a new exam we just need to press the sub-menu "New test"(Figure 14).
In the new exam page the following fields can be completed:

- Title
- Permanent link (it is displayed after the draft save)
- Description

Fig. 10. Warning for not choosing question type

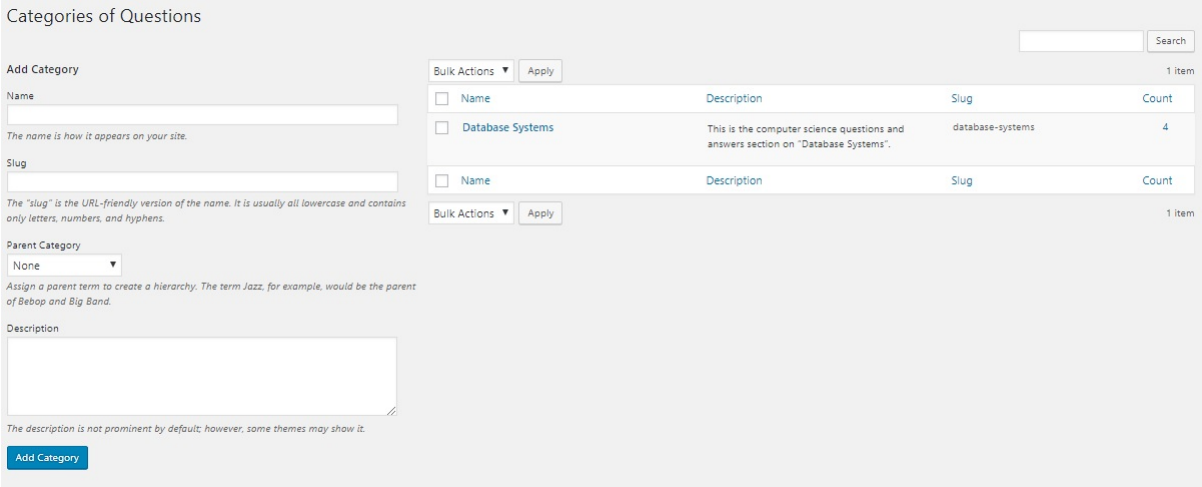

Fig. 11. Management question category page

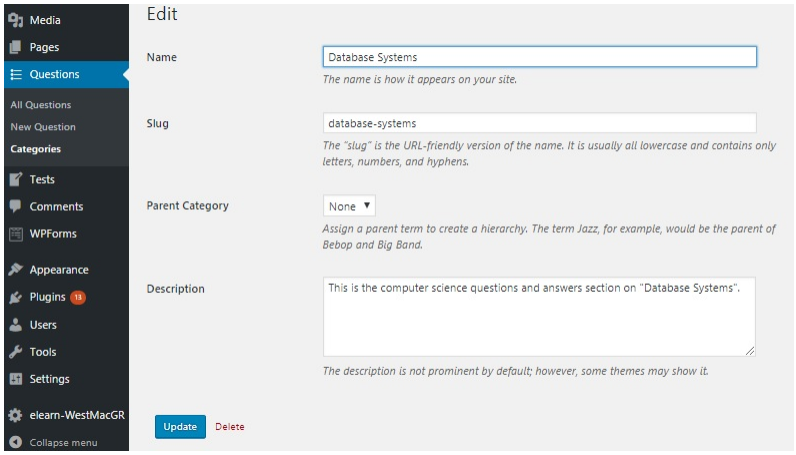

Fig. 12. Question category editing

- Options
- Publish

In the Title field we complete the title that we want the exam to have

In the field Permanent link we can define which the unique URL is going to be. The options are also depended on the Wordpress permanent link settings.

In the Description field we can write general instructions concerning the exam. This field can accept formatted text as well as multimedia (images, video, etc.). This field can be blank.

In the Publish field we can publish or save the exam as a draft and edit it later without being published yet.

Fig. 13. Custom post type exam

Fig. 14. New exam addition

Finally, in the Options field there are options concerning how the exam is going to manage the questions.

2) Exam options: Initially in the exam options the teacher defines from which question category the selection will take place. In the popup menu all the existing categories are displayed automatically. Then we complete how many multiple choice and long answer questions will be selected. Next to each selection there are the corresponded question importance fields (Figure 15). The importance is about the question type and there is an equivalent allocation on each one of them. Then, reduction to the unit by the system is done and the scoring will be displayed with a maximum of 10 or 100 depending on what we have chosen on the popup menu "Maximum Rating". Finally, we select if there will be a question randomization. If "No" is selected then the system will display the same questions in the same order to every student.

3) Running the Student exam: As shown in the following question, depending on the options that were previously made we can see all the available exams. The exam page is defined in the templates/page-diagonismata.php file.

By pressing on one of the exams (Figure 16) the student is redirected to his/her their exam page which is defined by tempates/single-diagonisma.php file.

Initially the student is asked to provide his personal details. But the system has already selected the questions which will be displayed in him and will store them along with his personal details. This happens because we don't want the student to view the selected questions and to prevent him from making any unwanted actions (e.g. to close the browser tab and revisit the exam hoping for "easier" questions).

When the user presses the exam "Start" button his/her personal details are stored in the database and specifically in the the wp_elearn_results table (Figure 17). The entry time, exam status and selected questions are also stored in a array in order to be stored in a table field. Furthermore, the HTML symbol " "' has been override by putting the "\" symbol. This technique prevents any unwanted malicious action.

Then the student is redirected in a new web page that has all the questions that have been selected randomly and is asked to answer them (Figures 18-19). In the case of multiple choice questions the possible answer order is random as well .

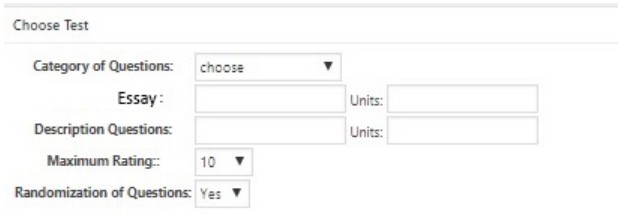

Fig. 15. Exam options

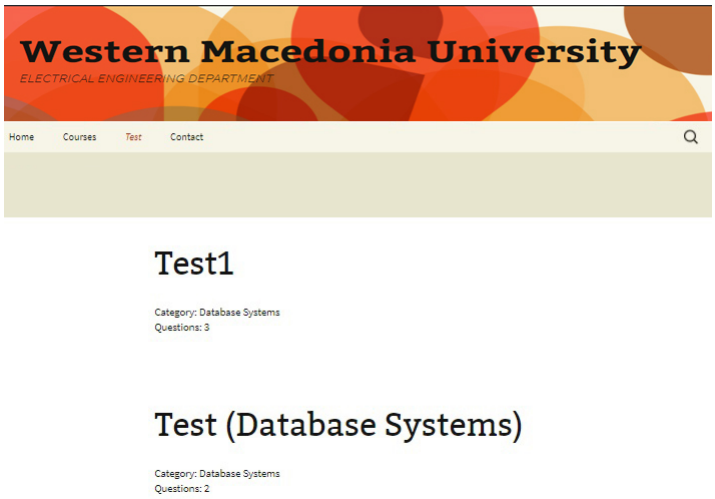

Fig. 16. Exam page

The student, after reading and answer the questions must press the "Finalize" button in order the exam to be stored in the database and then the teacher will be able to check it (Figure 20).

Before the final answers is sent, there is a final validation message (Figure 21) warning the student that he/she cannot be able to change the answers if the exam is submitted.

The answers have been stored in the database and also the status has been changed from "open" to "finalized"(Figure 22).

E. Answer correction by the teacher

The teacher can view the stored answers from the menu "elearn-WestMacGR" → "Tests". In the templates/admin-results.php file, there is a table with all the answered exams. In the table the exam title, students' personal details, exam status, grades and entry date are shown. By pressing the exam title, the teacher can correct and put grades on.

As shown in Figure 23, in the templates/admin-results.php file the submitted answers have already been corrected and the teacher is asked to check the long answer questions.

After the teacher inserts the grade in the long answer questions in the list with the stored exams the grade is displayed for the specific exam and the status changes to "Checked"(Figure 24).

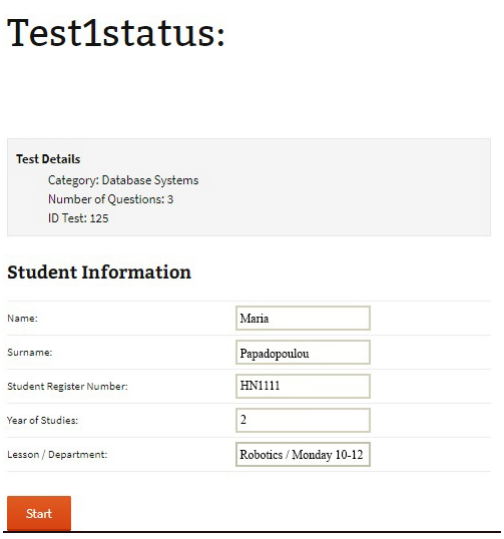

Fig. 17. Registering student details for the exam

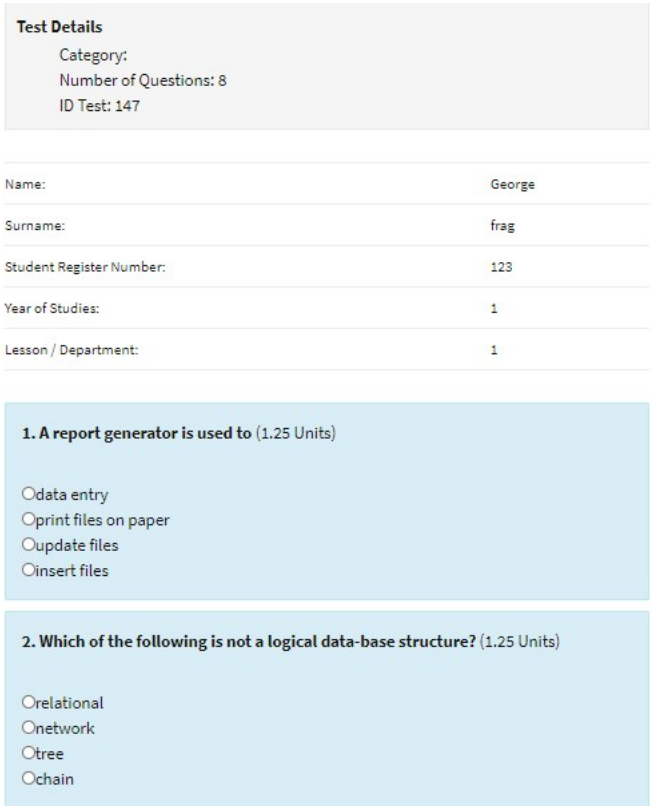

Fig. 18. Multiple choice questions

In Figure 25 the admin can see all the stored exams using phpMyAdmin application.

II. WordPress plugin Table architecture

As it shown in Figure 26, the table of the plugin is named wp_elearn_results. In this table are saved the test results of the students

The following fields are displayed in the table:

Result_id: The primary key of the result table. (Type: Integer, with a maximum length of 9 digits)

-Diagonisma_id: A field in which are stored the identifiers of the tests. (Type: integer, with a maximum length of 11 digits)

-First_name: A field in which is stored student's name. (Type: Varchar, with a maximum length of 50 digits)

-Second_name: A field in which is stored the last name of the student. (Type: Varchar, with a maximum length of 50 digits)

-Am: A field in which is stored student's identifier. (Type: Varchar, with a maximum length of 10 digits)

-Etos_Spoudon: A field in which is stored the year of the studies of the student. (Type: Varchar, with a maximum length of 20 digits)

-Tmima: A field in which is stored the department of studies of the student. (Type: Varchar, with a maximum length of 100 digits)

-Time_submitted: A field in which is stored the date and the time of the completed test. (Type: Date and time with the following form YYYY-MM-DD HH:MM:SS)

-Status: A filed in which is stored test's status which can be Open, Checked or Finalized. (Type: Varchar, with the maximum length of 100 digits)

-Answers: A field in which are stored the answers of the test. (Type: Text, with a maximum length of 65535 digits)

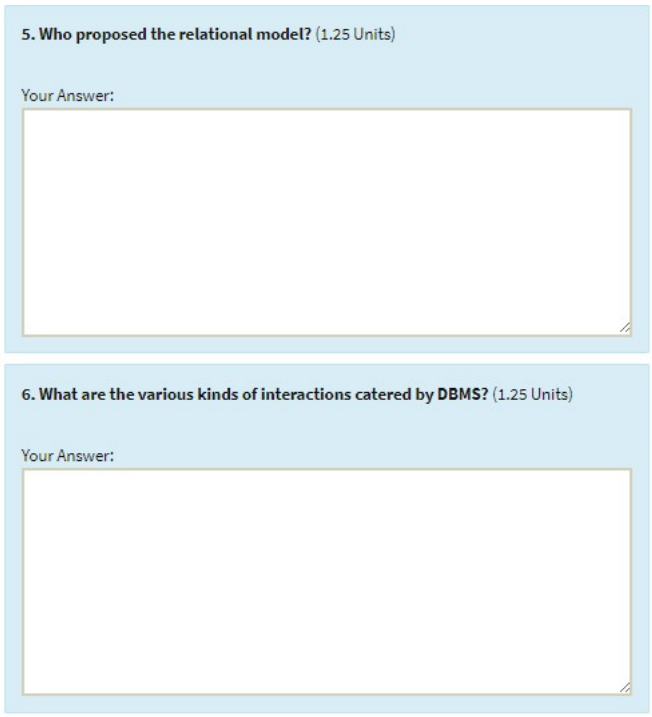

Fig. 19. Long answer/essay form question

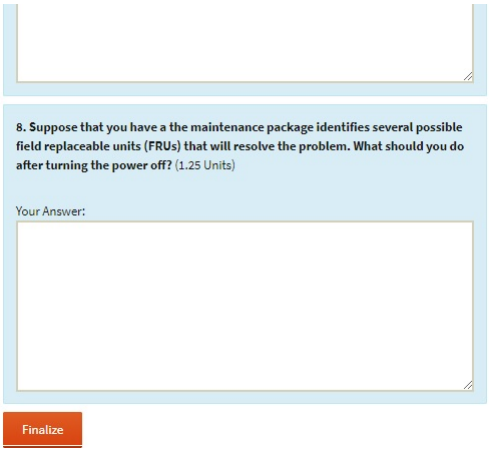

Fig. 20. Finalize button

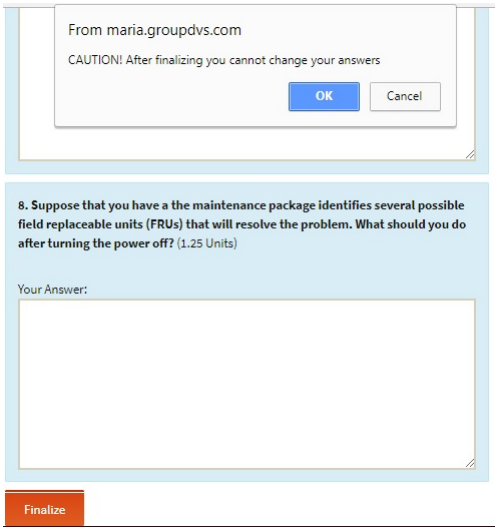

Fig. 21. Warning message before test submmited

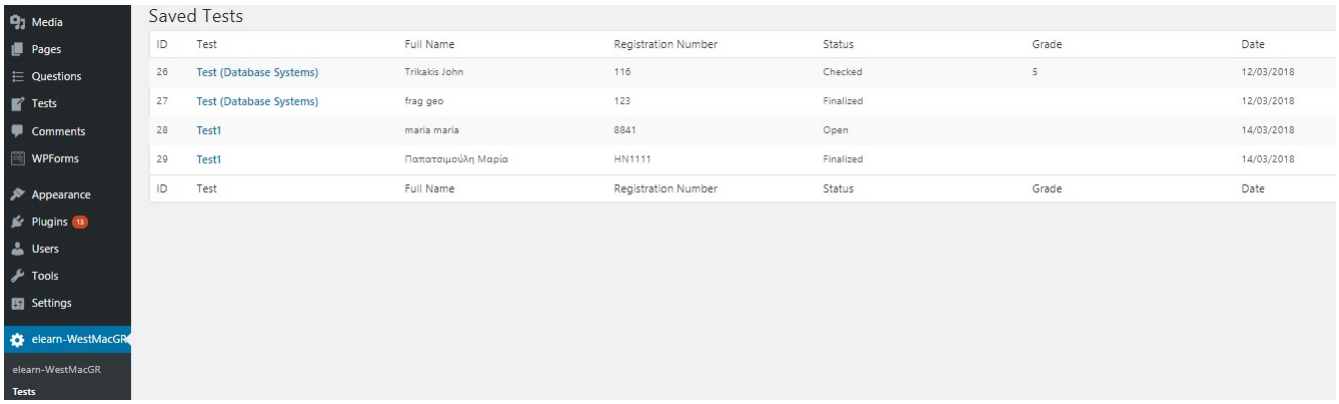

Fig. 22. Status change from open to finalized

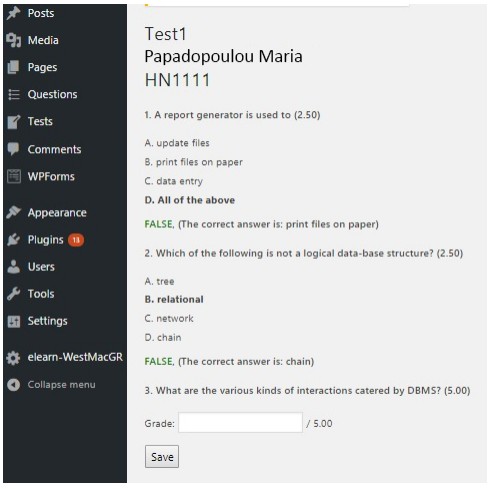

Fig. 23. Exam check and score finalization

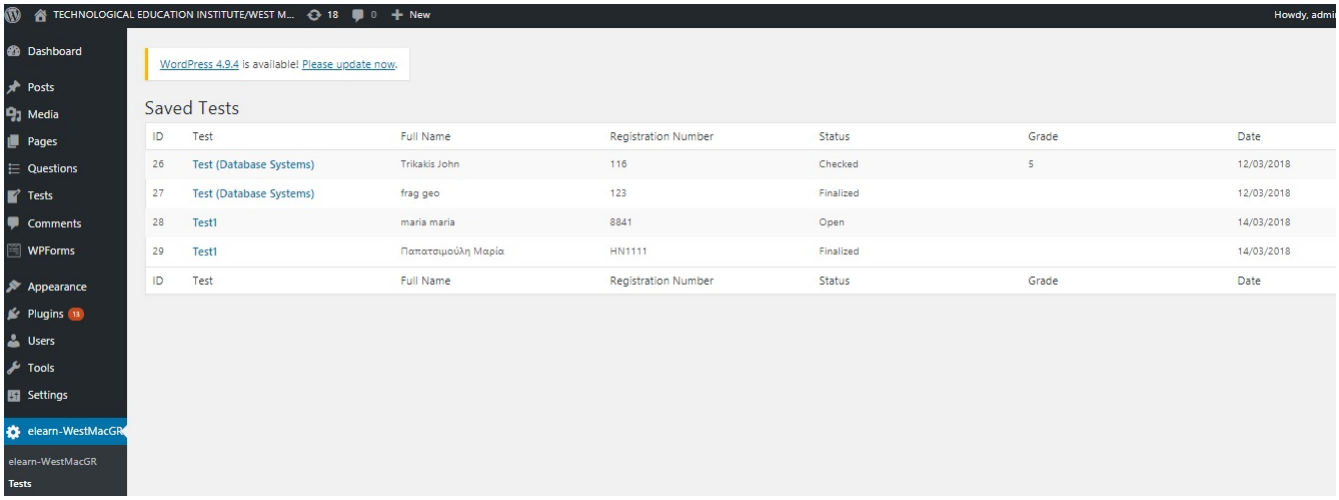

Fig. 24. Final Grades display

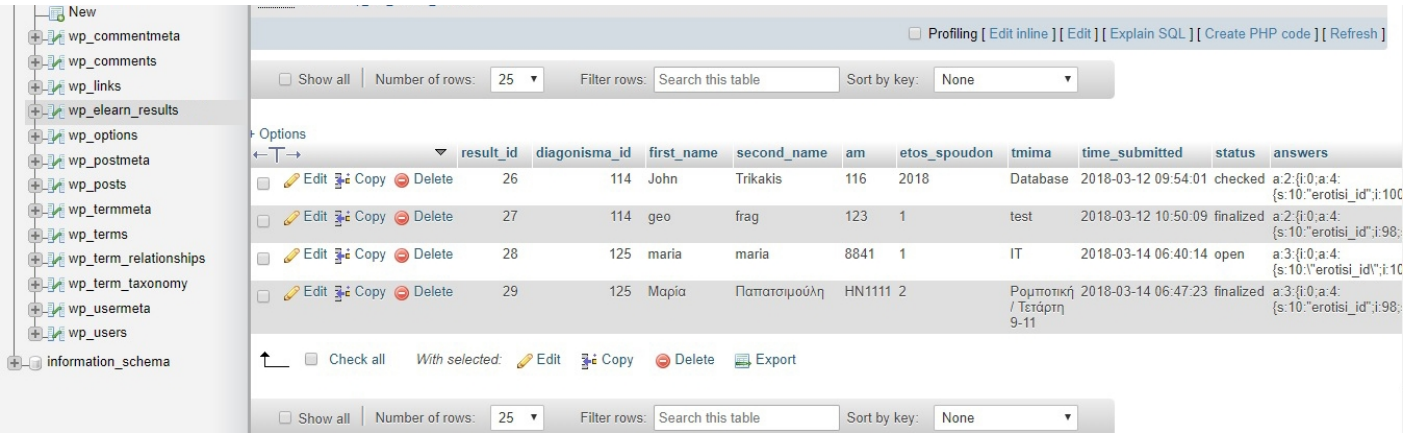

Fig. 25. Stored exams management

wp_elearn_results

Name	Type
result_id	mediumint(9)
diagonisma_id	int(11)
first_name	varchar(50)
second_name	varchar(50)
am	varchar(10)
etos_spoudon	varchar(20)
tmima	varchar(100)
time_submitted	datetime
status	varchar(100)
answers	text

Fig. 26. Plugin's table in which the results of student's tests are stored