

## WEB TARGET EDUCATIONAL INTERACTIVE EXERCISES CREATOR

by  
**Ioan Vlasin**

**Abstract.** In this paper I want to present powerful software for interactive educative exercises creation - Hot Potatoes and extensions for this. The University of Victoria Humanities Computing offer Hot Potatoes free to the educational community, under certain conditions. The purpose of the Hot Potatoes is to enable the user to create interactive Web-based teaching exercises that can be delivered to any Internet-connected computer equipped with a browser. The exercises use HTML and JavaScript to implement their interactivity, but the user do NOT need to know anything about these languages in order to use the programs. Can be created more than six exercise-type, fully customizable. The results can be sent by e-mail or saved with a program or CGI script. In web pages the user can insert images, links, sounds, objects. Using java applets we can create a complex interactive educational environment, for example assisted virtual experiment on physics. Because of its full source configuration of web pages, can create new types of tests, like the adaptive tests. The advantage is that all interactive features and questions of the test are available offline. This suite is a flexible and very good tool for teachers.

### 1. GENERAL PRESENTATION

This suite includes five basic programs:

- The JQuiz program creates question-based quizzes. Questions can be of four different types, including multiple-choice and short-answer. Specific feedback can be provided both for right answers and predicted wrong answers or distracters. In short-answer questions, the student's guess is intelligently parsed and helpful feedback to show what part of a guess is right and what part is wrong. The student can ask for a hint in the form of a "free letter" from the answer.
- The JCloze program creates gap-fill exercises. Unlimited correct answers can be specified for each gap, and the student can ask for a hint and see a letter of the correct answer. A specific clue can also be included for each gap. Automatic scoring is also included. The program allows gapping of selected words, or the automatic gapping of every nth word in a text.
- The JCross program creates crossword puzzles that can be completed. You can use a grid of virtually any size. A hint button allows the student to request a free letter if help is needed.
- The JMix program creates jumbled-sentence exercises. Unlimited correct answers can be specified, based on the words and punctuation in the base sentence, and a hint button prompts the student with the next correct word or segment of the sentence if needed.

- The JMatch program creates matching or ordering exercises. A list of fixed items appears on the left (these can be pictures or text), with jumbled items on the right. This can be used for matching vocabulary to pictures or translations, or for ordering sentences to form a sequence or a conversation.

The exercises are standard Web pages using XHTML 1.1 code for display, and JavaScript for interactivity. But you don't need to know anything about XHTML or JavaScript to use the programs. Only enter your data (texts, questions, answers etc.) and the programs will create the Web pages. However, the programs are designed so that almost every aspect of the pages can be customized, so if the creator of exercises knows HTML or JavaScript code, he can make almost any change he wants on the way the exercises work or on the format of the Web pages.

The most important quality of exercises is the high interactivity; the user can have in each moment information and feedback for his actions. This can make the exercises very interesting and the education attractive. For example the next exercise (Fig.1) the user can find the correct answer letter by letter, but this will decrease the score.

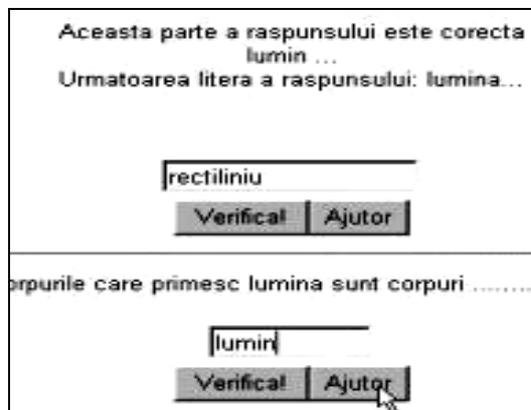


Fig. 1. A short answer exercise with help

## 2. FEATURES

All of the Hot Potatoes programs give the option to include a reading text in a separate frame next to your quiz.

In order to insert images, sounds, videos or links into the Web pages is possible to type HTML tags directly into most of the Hot Potatoes text boxes, or use one of the built-in functions to help.

The XHTML 1.1 specification includes a sophisticated system for embedding "objects" into the middle of a Web page. This is the most useful for including embedded "players" for media files. In applications of this suite can use this system to

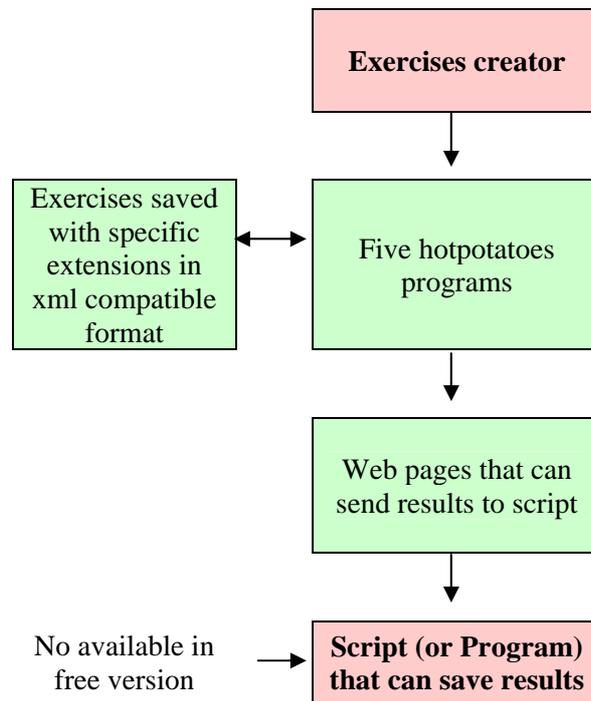
include (for example) a Flash movie or a RealMedia player on a Web page, by using the Insert / Media Object menu command.

Windows 2000 and XP support Unicode (a standard system for encoding characters from almost all human languages in such a way that languages can be mixed easily in the same document, and displayed reliably without the need for special fonts) natively, and Hot Potatoes version 6 makes use of this to enable the exercises creators to create exercises in virtually any language (or a mixture of languages).

To limit the time we can add a timer to exercises by specifying this in the Timer tab of the Configuration screen.

### 3. MAKING AN EXERCISE

A schematic representation of programs functionality can be this:



There are three main steps to create an exercise:

#### 3.1 Entering data

Entering data (questions, answers, reading text, images, links, and so on) is very simple because the interface of each application is intuitive and have button on the toolbar for main activities.



Fig. 2 Form (partial) for question input with Jquiz

This application can manage the question: insert, clone, move the questions in test, delete, and shuffle order of questions or answers (Fig.3).

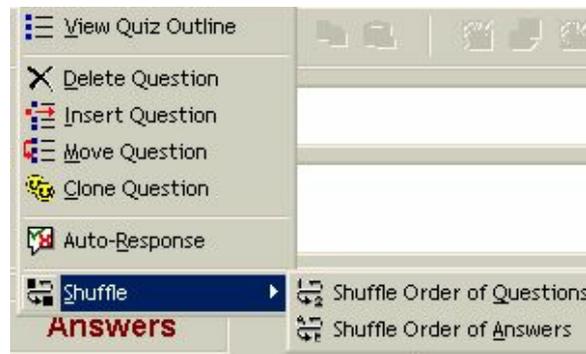


Fig. 3. Options of thing with the questions tests

All the information's of tests is saved in xml format, and is simple to work with this, like with text documents.

### 3.2 Configuring the output

The exercises created by Hot Potatoes programs use a variety of buttons and prompt to interface with the user. While the data for exercises (questions, answers etc.) will change from exercise to exercise, such things as button captions, prompts, and explanations will not change so often. These are therefore stored in the form of a configuration file. Each program provides an interface to the configuration file. These will bring up a screen and giving access to all the configuration data relevant to the application.

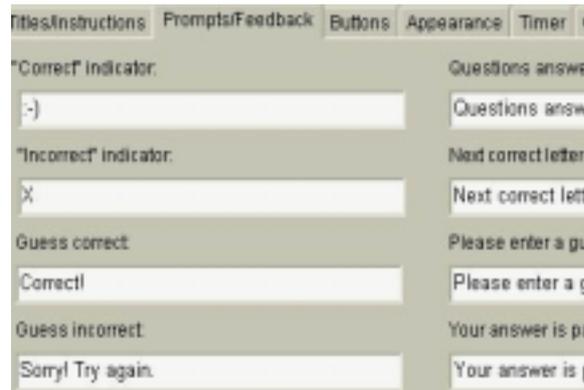


Fig.4. Detail of a Web page configuration

One of the most powerful features in *Hot Potatoes* is the ability to edit directly the source files (or templates) that the program uses to generate the Web pages (usually kept in the **source** subfolder of the main *Hot Potatoes* folder). These files are text files, and can be changed.

If the test creator is familiar with HTML and JavaScript, he may want to make he's own changes to the templates used to create the Web pages. In previous versions (3,4,5) of hotpotatoes applications a function enables the creator of exercises to edit directly the source of templates. In version 6 the templates can be modified only if the user edits the file of source.

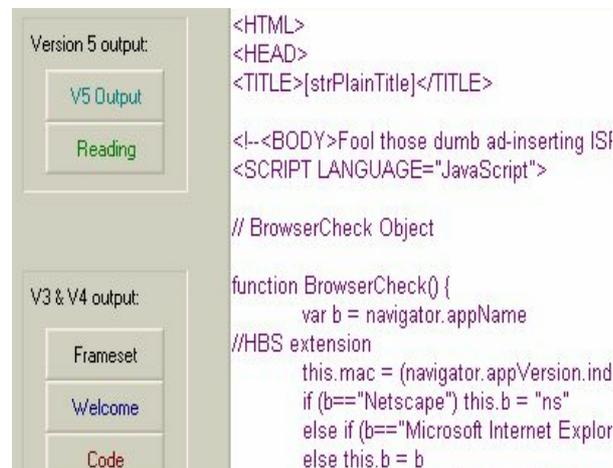


Fig.4 Edit source code of templates

For the creators that do not want much particularization of his exercises, the multiples options for configurations are enough.

### 3.3. *Creating the Web pages*

Creating the Web pages (compiling exercises into HTML pages) is the simplest part of using *Hot Potatoes*. Use the Create Web Page commands, accessible through the File menu or the toolbar, specified a file name, and the Web page will be created.

## 3. SEND SCORES

From the Web pages, the Internet browser cannot receive command to save the information. Only the user can do this. This is the reason why the exercises created with this suite don't save results. But all can send results to a program or a CGI script that can save results into a file or send by email. The exercises creator can configure this using the CGI panel of the configuration screen. The free version of this suite doesn't have a program or a CGI script to save scores.

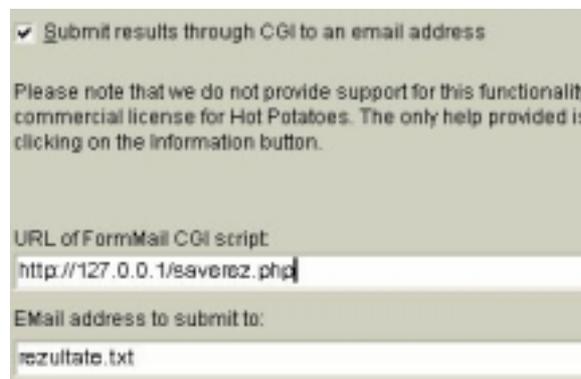


Fig. 5. Enter program or script that receives the scores.

A program or a CGI script can be created or find on the Internet. To use CGI script or a program a web server such as Apache is required, this must send results from the web page to the script.

## 4. EXTENSIONS

In this section I present some examples, possible extensions that can make this suite more useful.

#### 4. 1. Save results with detailed score

To do this first I create a program in C language that can save results in a text file. With source changes of templates (Java Script support), the exercises can send detail of score.



Fig. 6. Detailed score saved and returned to user

A PHP script can do this if the web server has a support for PHP. A very simple PHP script can be the next:

```
<?php
echo "<html><title>Rezultate</title>";
echo "<body>";
$form_fields= array_keys($_HTTP_POST_VARS);
for ($i = 0; $i < sizeof($form_fields); $i++) {
    $thisField = $form_fields[$i];
    $thisValue = $_HTTP_POST_VARS[$thisField];
    $rez.="<b><i>$thisValue.</i>";
}
echo "<h3><center>Rezultate test</center><b>";
echo $_HTTP_POST_VARS[realname]. "</b> la testul <b><i>";
echo $_HTTP_POST_VARS[Exercise];
echo "</b></i> ai obtinut scorul " . $_HTTP_POST_VARS[Score];
echo "<br><br>Testul a avut loc de la:<br>" . $_HTTP_POST_VARS[Start_Time];
echo "<br> pina la:<br> " . $_HTTP_POST_VARS[End_Time];
echo "<br><br>Multumim!";
$rez.="<br></body></html>";
echo "<br></body></html>";
if ($fileName == "") {
    $fileName="rezult.txt";
```

```
}  
if (file_exists($fileName)){  
$fileHandle = fopen($fileName, "a");  
} else {  
$fileHandle = fopen($fileName, "w");  
}  
if (!$fileHandle) {  
return 0;  
}  
fwrite ($fileHandle, $rez);  
fwrite($fileHandle, "\n");  
fclose($fileHandle);  
?>
```

In field Email address you can write the file name where the results will be saved.

#### 4.2. A new type of test (exercise)

Modifying the templates source is caned obtained a new type of test. The next type has radio buttons and counts the number of each type of answer.

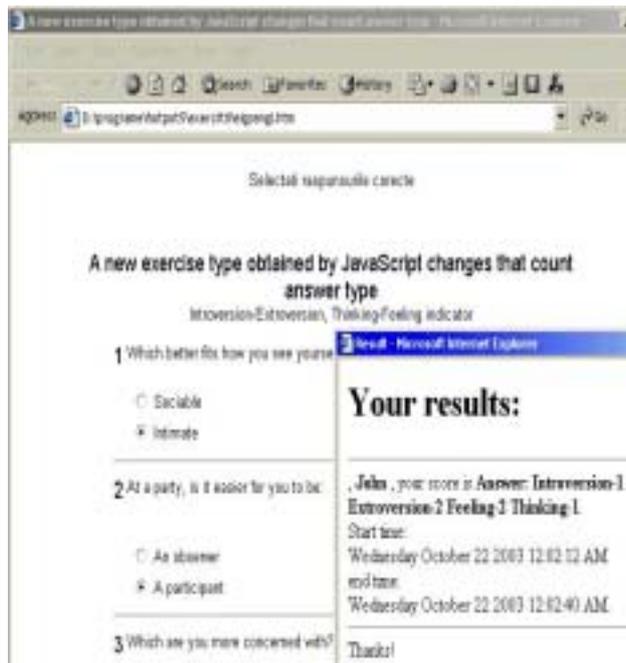


Fig.7.The new test type that counts the answers type

#### 4.3. Building a complex educational environment

A powerful, but an undocumented way to extend the ability of exercises is to insert java applet. Virtual assisted physics experiments can be created using java applet find on Internet and indications, questions in exercises. An example can be seen in Fig. 8.

#### 4.4. Creating adaptive tests

The source of templates can be modified to create adaptive tests. The advantage is that all features and questions are available offline.

A possible diagram for a basic adaptive test (tutorial) with 30 (for example) question /level can be view in Fig. 9. A test can contain more than 2 total level, for example 4 level of questions, with an entry point at 3 level to obtain 2 level /test.

Direction of question order (ordered or random in a specified level) can be:

- a) if the score of current level is  $>$  min\_acceptable – next question in this level (or next level if test have multiple level)
- b) if the score of current level is  $<$  min\_acceptable – back, go to previous level

With HotPotatoes is possible to create adaptive tests from standard interface because each question can have a weight = level.

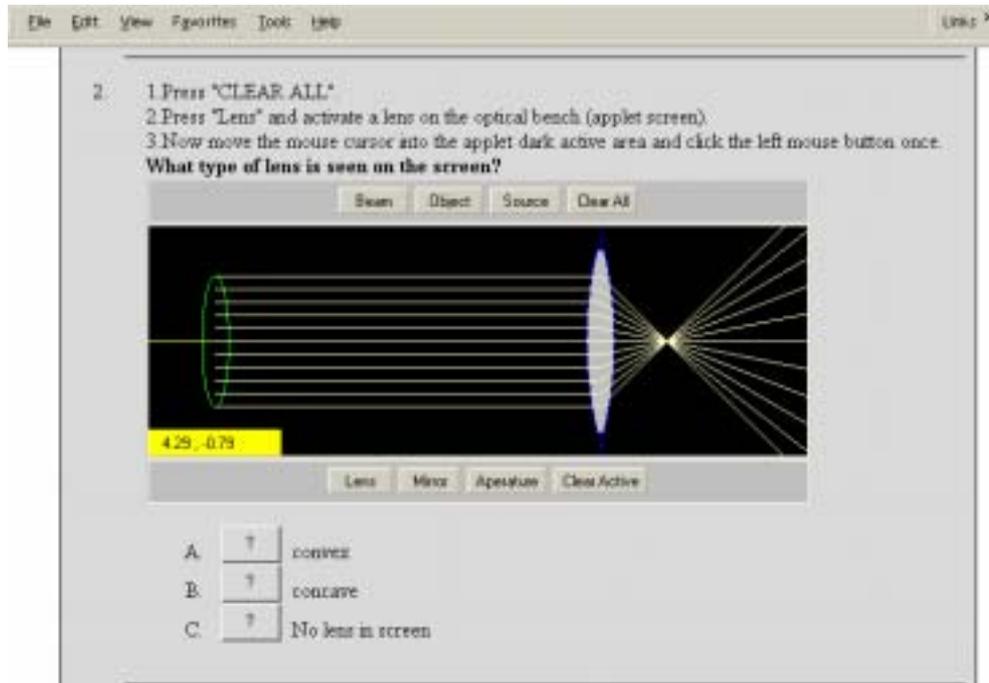


Fig. 8. Complex educational environment with Java Applets - Optics virtual assisted experiment

<i>Level</i>	<i>Questions</i>	1 LEVEL /TEST		2 LEVELS/ TEST	
I	1	<b>TEST 1</b> Basic knowledge	<b>TEST 2</b>	<b>TEST 1</b>	<b>TEST 2</b>
	2				
	3				
	....				
	29 30				
II	31	<b>TEST 3</b>	Entry point of test 2		
	....				
	59 60				
III	61	Entry point of test 3	<b>TEST 4</b>	<b>TEST 3</b>	Entry point of test 2
	...				
	89 90				
IV	91	<b>TEST 5</b>	Entry point of test 4		
	...				
	119 120				
V	121	Entry point of test 5	<b>TEST 6</b>	Entry point of test 3	<b>TEST 4</b>
	....				
	149 150				
VI	151	<b>TEST 7</b>	Entry point of test 6		
	...				
	179 180				
VII	181	Entry point of test 7	<b>TEST 8 ...</b>		Entry point of test 4
	....				
	210				

Fig.9. Possible adaptive tests with 1 or 2 levels/test

### References

- \*\*\* Hot Potatoes documentation
- \*\*\* JavaScript by example
- \*\*\* PHP Manual