Erasmus+

Project funded by: **Erasmus+ / Key Action 2 - Cooperation for innovation and the exchange of good practices, Strategic Partnerships for school education (European Commission, EACEA)**

WORLD OF PHYSICS

TECHNICAL GUIDE

Creating Interactive Quiz Activities

1. Introduction

Quiz activities are a great way to assess the student's learning outcomes.
In WOP, we used Quizzes in many cases in the form of series of multiple-choice questions.

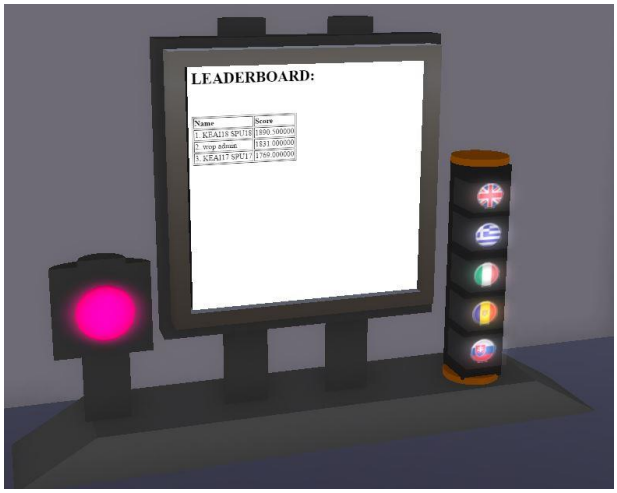
For the content of the quiz you need to come up with the question and the available options and also the feedback provided for each answer.

You can repeat the question until the student finds the correct answer or allow them to answer all questions and provide the results at the end.

To grade the student's performance, you can count the number of mistakes and/or the time it took them to complete the quiz successfully.

A nice idea is to integrate the quiz within a learning scenario. Completing the quiz successfully can provide the student with a prize or a specific object that they use in another activity. For example, we have used multiple Quizzes in one of the learning scenarios, each one awarding a Puzzle Piece that allows the student (or a group of students) to complete a Puzzle and view a relevant video.

In our implementation we have buttons to change the language of the quiz before taking it and we have a scoreboard that displays the names of the avatars that have complete the quiz along with their ranking.



2. Technical Details

First you need a linked set with the individual parts of the quiz. Besides the screen that will display the results that will be the Root object of the Linked Set, make sure you include the buttons that you want the user to be able to click to interact (Change language, start the quiz etc). After you link the object (select the screen prim last to make it the root of the set), you can put scripts in each part of it to make it interactive.

For the scripts in the button prims just use a 'touch_start' event and use 'llMessageLinked' to send messages to the root.

For the script of the root object (screen), use a 'link_message' event to wait for messages from the child objects (buttons), such as messages to change the language.

The quiz script uses 'llDialog' to show dialogue messages to the user and prompt them to select the correct message. It waits for messages (listen event) when the user selects an answer and validates if the answer was correct to decide what the feedback and the next question should be. It also waits for messages from the buttons (link_message event).

The quiz script can also keep track how much time it took for the student to answer all questions and how many mistakes they made and stores the data in list variables.

It also generates a text response in html format that it projects as a page in one of the planes of the root object, using 'llSetPrimMediaParams' to change the 'PRIM_MEDIA_CURRENT_URL'. parameter

You can find the scripts we use and more instructions in the "Scripts Section" here: http://aigroup.ceid.upatras.gr/wop-oer/scripts.html

3. References/Links

http://aigroup.ceid.upatras.gr/wop-oer/scripts.html

http://wiki.secondlife.com/wiki/LSL_Portal

http://wiki.secondlife.com/wiki/Touch_start

http://wiki.secondlife.com/wiki/LlMessageLinked

http://wiki.secondlife.com/wiki/Link_message

http://wiki.secondlife.com/wiki/LlSetPrimMediaParams

http://wiki.secondlife.com/wiki/LlDialog