

# Documentation



## PROJECT MODULE FILE (IT DEPARTAMENT)

---

GROUP (DAM2 | DAW2 | ASIX2 | SMX2A | SMX2B)

---

DAW2

TITLE

---

Web application languages for ONCE

GROUP MEMBERS (The students' names that will develop the project)

---

Sergio Baena López

DESCRIPTION (Write in no more than 15 text lines the objectives of the project)

---

The purpose of the project is to create an educational web application languages can also be accessed visually impaired children (also who do not have these problems). To do that, my application will incorporate sounds (a person speaking) along its execution describe any event that is happening inside it and a navigation through keys. Therefore, this application to the be adapted to children with visual deficiency does not require any type of screen reader for use. Not so, we must point out that this application can also use any other child in the way they have always done, as also support normal navigation by mouse and without any sound. The usefulness of this software is to make learning more fun for children. Here children will have exercises of count number of syllables, monosyllabic words to find, to categorize by type of word, ie, acute plain or antepenultimate, etc. Definitely, with this application we provide a learning tool for any child, of any.

## MATERIALS REQUIRED AND TECHNICAL SPECIFICATIONS OF THE PROJECT

---

For this project the following requirements:

Programming Languages:

- XHTML (also HTML5)
- CSS
- JavaScript
- JQuery
- PHP
- Ajax
- SQL

Software:

- MySQL
- Apache
- PHP interpreter
- Linux
- Komodo Edit (IDE)
- JAWS (screen reader)
- Recorder

Materials:

- Computer

- External hard disk

## FUNCTIONAL REQUIREMENTS

(Draft version: it is a first version to specify all functionalities and their results, as well as the profile users who may run each functionality)

---

An example of functional requirements:

**RF1. THE MANAGEMENT OF CARDS** (short description of the functionality and priority level: high, medium, low)

**RF1.1 NEW ENTRY CARD** (short description of the functionality and priority level: high, medium, low)

**RF1.2 CANCEL CARD** (short description of the functionality and priority level: high, medium, low)

**RF1.3 MODIFY DATA CARD** (short description of the functionality and priority level: high, medium, low)

**RF1.4 CONSULT DATA CARD** (short description of the functionality and priority level: high, medium, low)

**RF2. ...**

**(NOTE: as a result of your initial interviews with the clients, you may use whatever you consider necessary in order to clarify what the client precisely wants: diagram, graphic, description, etc)**

We will have three users for this application, namely, students, teachers and webmasters.

### RF1. MANAGING USER PROFILE

**RF1.1 REGISTERING A STUDENT.** Register a new student in the system so he/she can use it as such. Priority: high

**RF1.2 REGISTERING A TEACHER.** Register a new teacher in the system so he/she can use it as such. Priority: high

**RF1.3 VIEW A STUDENT.** Students can view their profile at any time. Also the teacher and webmaster can see its. Priority: high

**RF1.4 VIEW A TEACHER.** Teachers can view their profile at any time. Also webmaster can see its. Priority: high

**RF1.5 VIEW A WEBMASTER.** Webmaster can view their profile at any time. Priority:

**RF1.6 MODIFY A STUDENT.** The student can at any time modify some data on his/her user profile. Also the webmaster can. Priority: medium

**RF1.7 MODIFY A TEACHER.** The teacher can at any time modify some data on his/her user profile. Also the webmaster can. Priority: medium

**RF1.8 MODIFY A WEBMASTER.** The webmaster can at any time modify some data on his/her user profile. Priority: low

**RF1.9 DELETE A STUDENT.** The student may at any time unsubscribe from the system and no longer have access to games and other things. Also the webmaster can. Priority: low

**RF1.10 DELETE A TEACHER.** The teacher at any time may unsubscribe to the system and no longer have access to the activities of their associated students and other things. Also the webmaster can. Priority: low

**RF1.11 DELETE A WEBMASTER.** The web administrator at any time may unsubscribe to no longer the system and be disable to manage the site using this utility. Priority: low

**RF2. GAME COUNT NUMBER OF SYLLABLES.** Set of 10 words by game. A word is displayed on the screen and there will be time to respond. It is scored as follows: 10 points for the first correctly, 1 point for the second correctly and 0 points for not doing it correctly. Priority: high

**RF3. GAME TO FIND MONOSYLLABIC WORDS.** Game is a list of some monosyllabic words that the user has to find what they are. Response time will be. Scores will correctly 10 points completely, if you do not select the proportional part will be discounted based on the number of monosyllables beech (no decimals accepted in the ratings). 10 lists for departure. Priority: high

**RF4. GAME OF CATALOGING WORDS BY ACUTE, PLAIN OR ANTEPENULTIMATE.** Set of 10 words per game. For each screen will display a word and should be indicated if is acute, plain or antepenultimate at a given time. Scores will be 10 points for the first correctly, 1 point for correctly the second and 0 points for no right or finish time. Priority: high

RF5. GAME TO INDICATE WHETHER THAT WORD HAS AN ACCENT OR NOT. Set of 10 words per game. For each screen presented with a word without accent and should be indicated in a particular time if have an accent or not. 6 points will be given for correctly and no points for failure or time completion. Priority: high

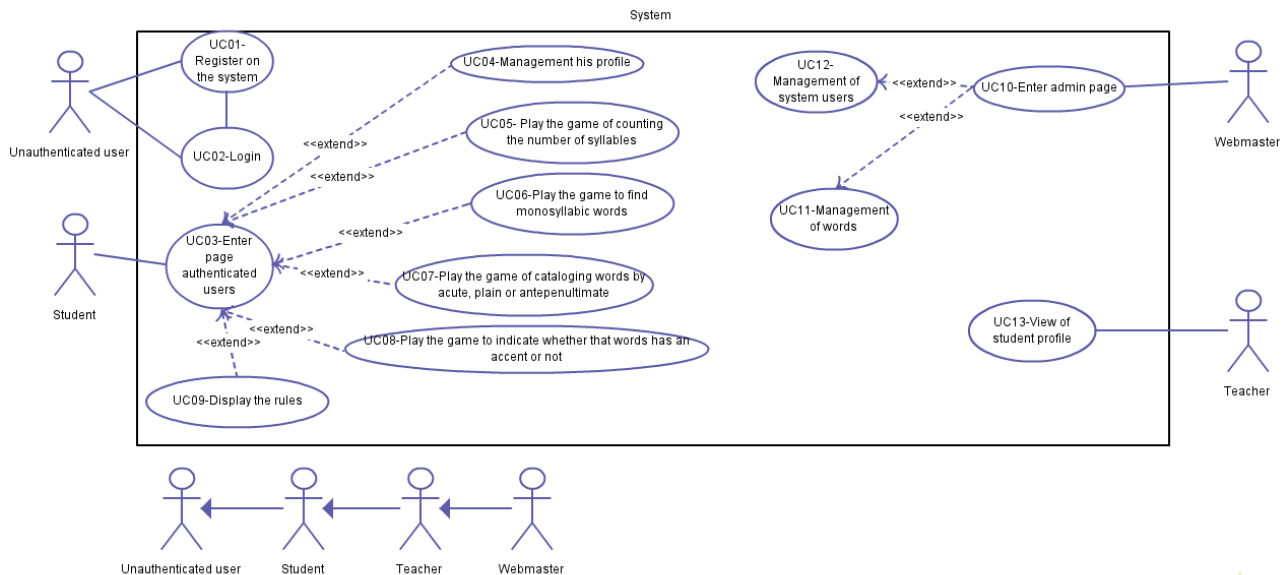
RF6. MANAGEMENT OF WORDS. Add new word with its characteristics (number of syllables, if monosyllabic or not, if is acute ..., etc), list words, modify, delete. Priority: low

RF7. LIST ALL TEACHERS. Lists all teachers currently registered in the system. Priority: low

RF8. LIST ALL STUDENTS. Lists all students currently registered in the system. Priority: low

RF9. LIST ALL WEBMASTERS. Lists all webmasters currently registered in the system. Priority: low

RF10 - DISPLAY INFORMATION ABOUT THE RULES OF THE GAMES. Priority: high



[online diagramming & design] [creately.com](https://creately.com)

Code	UC01
Name	Register on the system
Author	Sergio Baena López
Date	2014-05-07
Description	Register as a new user in the system
Actors	Unauthenticated user
Preconditions	Nothing
Normal flow	<ol style="list-style-type: none"> <li>1. The actor requesting registration option.</li> <li>2. The system displays a registration form.</li> <li>3. The actor fill out the form by typing in text fields: the name, surnames, school, selected from a select the province, city, username, password, confirmation of the password. We must also say that the actor has above all a select whose options are student and teacher. <ol style="list-style-type: none"> <li>3.1. The actor clicks the student option <ol style="list-style-type: none"> <li>3.1.1. The system extends the form.</li> <li>3.1.2. The actor entered the course and date of birth.</li> </ol> </li> <li>3.2. The actor clicks the teacher option <ol style="list-style-type: none"> <li>3.2.1. The system extends the form.</li> <li>3.2.2. The actor entered the courses separated by commas</li> </ol> </li> </ol> </li> <li>4. The actor clicks the <i>Register</i> button.</li> <li>5. The system validates all fields.</li> <li>6. The system makes an insert on the table <i>User</i>, <i>Student</i> or <i>Teacher</i> and the <i>Ranking</i>.</li> <li>7. The system displays on the screen <i>Registro satisfactorio</i>.</li> <li>8. The system redirects the actor to login page.</li> </ol>
Alternative flow	5.1. The system finds some incorrect data. Displays an error message telling the actor the error and marking the field in red.
Postconditions	The actor is registered in the system (present in the table <i>User</i> , <i>Student</i> or <i>Teacher</i> and <i>Ranking</i> ).

Code	UC02
Name	Login
Author	Sergio Baena López
Date	2014-05-07
Description	An user authentication system.
Actors	Unauthenticated user
Preconditions	The actor has to be registered in the system.
Normal flow	<ol style="list-style-type: none"> <li>1. The actor requested the option to log in.</li> <li>2. The system displays the form of logging</li> <li>3. The actor enters his username and password.</li> <li>4. The actor clicks the button "Entrar".</li> <li>5. The system verifies that the user is present in the DB.</li> <li>6. The system to open the session</li> <li>7. The system will redirect to the page authenticated users.</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>5.1. The system detects that the user is not present in the DB.</li> <li>5.2. The system shows the actor an error message: <i>Nombre de usuario y/o contraseña no válidas.</i></li> <li>5.3. The system deletes the content entered in the form.</li> </ol>
Postconditions	The session of the actor has to opened.

Code	UC03
Name	Enter page authenticated users
Author	Sergio Baena López
Date	2014-05-07
Description	The authenticated actor could view and run any functionality present on the page of authenticated users.
Actors	Student
Preconditions	The user session must be open.
Normal flow	1. The system show page authenticated users. 2. The actor performs the functionality he wants
Alternative flow	Nothing
Postconditions	The actor has executed a functionality



Code	UC04
Name	Management his profile
Author	Sergio Baena López
Date	2014-05-07
Description	The actor can view, modify and delete his profile.
Actors	Student
Preconditions	The user session must be open.
Normal flow	<p>1. The actor requested the option to manage their profile</p> <p>2. The system has five forms, filled. The first contains the attributes of the student (table <i>User</i> and <i>Student</i>), the second contains the attributes of the student ranking of a specific game (read only fields), the third in another and so, successively. In addition, the bottom, we have two buttons, namely <i>Guardar</i> and <i>Eliminar cuenta</i>.</p> <p>2.1. The actor displays his profile data</p> <p>2.2. The actor modifies any of his profile data (first form)</p> <p>2.2.1. The actor clicks the <i>Guardar</i> button.</p> <p>2.2.2. The system validates the data.</p> <p>2.2.3. The system updates the active user (stored in the session).</p> <p>2.2.4. The system displays the screen <i>Tu perfil ha sido modificado correctamente</i>.</p> <p>2.3. The actor clicks the <i>Eliminar cuenta</i> button.</p> <p>2.3.1. The system launches a <i>confirm</i> to the actor asking if he is sure he wants to delete his account.</p> <p>2.3.2. The system makes a <i>DELETE</i> table <i>User</i>, <i>Student</i>. And also, last four <i>DELETE</i>'s on the table <i>Ranking</i> eliminating the rankings associated actor.</p> <p>2.3.3. The system close the user session.</p> <p>2.3.4. The system shows the actor the message <i>Tu cuenta ha sido eliminada correctamente</i> (alert).</p> <p>2.3.5. The system redirects the web to the home page.</p>
Alternative flow	<p>2.2.2.1. The system discovers that some data is incorrect.</p> <p>2.2.2.2. The system displays an error message indicating the errors found.</p> <p>2.2.2.3. The system puts the wrong fields in red.</p> <p>2.3.1.1. The actor has indicated a "no" on the "confirm".</p>
Postconditions	If the actor has not just looked at his profile, the user stored in the session must have been modified

Code	UC05
Name	Play the game of counting the number of syllables
Author	Sergio Baena López
Date	2014-05-08
Description	The actor plays the game of counting the number of syllables
Actors	Student
Preconditions	The user session must be open.
Normal flow	<p>1. The actor requested the option to play this game.</p> <p>2. The system shows, first, a text describes the game (<i>short_description</i>), second game scores (<i>punctuation_at_the_first_attempt</i> and <i>punctuation_at_the_second_attempt</i>) and, finally, two buttons, namely, <i>Jugar ahora</i> and <i>Ver reglas</i>.</p> <p>2.1. The actor clicks the <i>Jugar ahora</i> button.</p> <p>2.1.1. The system changes the contents by a word (from the DB), the time remaining to the actor, and the number of attempts.</p> <p>2.1.2. The actor press the number of syllables believes there within time.</p> <p>2.1.3. The system indicates that the actor has correctly the word.</p> <p>2.1.4. The system displayed again a new word, and thus successively.</p> <p>2.1.5. The system displays the end of the game, statistics of this game and statistics every game. The bottom, there will be a button to play again.</p> <p>2.2. The actor clicks the <i>Ver reglas</i> button.</p> <p>2.2.1. The system shows the actor rules of the game together with a button <i>Jugar ahora</i>.</p>
Alternative flow	<p>2.1.1.1. The actor does not press a key on the time his has.</p> <p>2.1.1.2. The actor lost an attempt or loses this word. If lost, a new word appears.</p> <p>2.1.2.1. The system indicates that it is wrong and adds one more try.</p> <p>2.1.2.2. The system indicates that the actor has pressed a key that is not a number.</p>
Postconditions	The ranking stored in the session has been updated.

Code	UC06
Name	Play the game to find monosyllabic words
Author	Sergio Baena López
Date	2014-05-08
Description	The actor plays the game to find monosyllabic words
Actors	Student
Preconditions	The user session must be open.
Normal flow	<p>1. The actor requested the option to play this game.</p> <p>2. The system shows, first, a text describes the game (<i>short_description</i>), second game scores (<i>punctuation_at_the_first_attempt</i> and, finally, two buttons, namely, <i>Jugar ahora</i> and <i>Ver reglas</i>.</p> <p>2.1. The actor clicks the <i>Jugar ahora</i> button.</p> <p>2.1.1. The system displays a list of words with some monosyllabic with a checkbox next to her, then the time remaining and, finally, the <i>Aceptar</i> button.</p> <p>2.1.2. The actor checks the words believes appropriate.</p> <p>2.1.3. The actor clicks the <i>Aceptar</i> within the proper time.</p> <p>2.1.4. The system indicates that your choice is correct.</p> <p>2.1.5. The system displayed again a new list of words, and thus successively.</p> <p>2.1.6. The system displays the end of the game, statistics of this game and statistics every game. The bottom, there will be a button to play again.</p> <p>2.2. The actor clicks the <i>Ver reglas</i> button.</p> <p>2.2.1. The system shows the actor rules of the game together with a button <i>Jugar ahora</i>.</p>
Alternative flow	<p>2.1.2.1. The actor did not click the <i>Aceptar</i> button and the time is already past.</p> <p>2.1.2.2. The system shown any successful word</p> <p>2.1.2.3. The system again displays a list of words and thus successively.</p> <p>2.1.3.1. The system indicates that the selected words do not correspond to 100%.</p> <p>2.1.3.2. The system again displays a list of words and thus successively.</p>
Postconditions	The ranking stored in the session has been updated.

Code	UC07
Name	Play the game of cataloging words by acute, plain or antepenultimate.
Author	Sergio Baena López
Date	2014-05-08
Description	The actor plays he game of cataloging words by acute, plain or antepenultimate.
Actors	Student
Preconditions	The user session must be open.
Normal flow	<p>1. The actor requested the option to play this game.</p> <p>2. The system shows, first, a text describes the game (<i>short_description</i>), second game scores (<i>punctuation_at_the_first_attempt</i> and <i>punctuation_at_the_second_attempt</i>) and, finally, two buttons, namely, <i>Jugar ahora</i> and <i>Ver reglas</i>.</p> <p>2.1. The actor clicks the <i>Jugar ahora</i> button.</p> <p>2.1.1. The system displays a word, three options (acute, flat and antepenultimate) along with a radiobutton, the remaining time and the number of attempts.</p> <p>2.1.2. The actor checked an option within the correct time interval.</p> <p>2.1.3. The system verifies that it is a valid response.</p> <p>2.1.4. The system reports that the answer is correct.</p> <p>2.2.5. The system displayed again a new word, and thus successively.</p> <p>2.1.6. The system displays the end of the game, statistics of this game and statistics every game. The bottom, there will be a button to play again.</p> <p>2.2. The actor clicks the <i>Ver reglas</i> button.</p> <p>2.2.1. The system shows the actor rules of the game together with a button <i>Jugar ahora</i>.</p>
Alternative flow	<p>2.1.1.1. The actor does not check any option on all the time.</p> <p>2.1.1.2. The system tells the actor that the time is over and it increases a try.</p> <p>2.1.1.2.1. Step 2.1.1 runs if more attempts are not possible.</p> <p>2.1.1.2.2. The system deselects the radiobutton, put back a new time and indicates that now is the second attempt. All this, in case a possible retry.</p> <p>2.1.1.2.2.1. It runs from step 2.1.1.</p> <p>2.1.3.1. The system indicates that it is not the right answer.</p> <p>2.1.3.2. The system increases a try.</p> <p>2.1.3.3. It runs what it contains step 2.1.1.2</p>
Postconditions	The ranking stored in the session has been updated.

Code	UC08
Name	Play the game to indicate whether that words has an accent or not
Author	Sergio Baena López
Date	2014-05-08
Description	The actor plays the game to indicate whether that words has an accent or not
Actors	Student
Preconditions	The user session must be open.
Normal flow	<p>1. The actor requested the option to play this game.</p> <p>2. The system shows, first, a text describes the game (<i>short_description</i>), second game scores (<i>punctuation_at_the_first_attempt</i>) and, finally, two buttons, namely, <i>Jugar ahora</i> and <i>Ver reglas</i>.</p> <p>2.1. The actor clicks the <i>Jugar ahora</i> button.</p> <p>2.1.1. The system displays a word, three options (<i>Tiene acento</i>, <i>No tiene acento</i>) along with a radiobutton, the remaining time and the number of attempts.</p> <p>2.1.2. The actor checked an option within the correct time interval.</p> <p>2.1.3. The system verifies that it is a valid response.</p> <p>2.1.4. The system reports that the answer is correct.</p> <p>2.2.5. The system displayed again a new word, and thus successively.</p> <p>2.1.6. The system displays the end of the game, statistics of this game and statistics every game. The bottom, there will be a button to play again.</p> <p>2.2. The actor clicks the <i>Ver reglas</i> button.</p> <p>2.2.1. The system shows the actor rules of the game together with a button <i>Jugar ahora</i>.</p>
Alternative flow	<p>2.1.1.1. The actor does not check any option on all the time.</p> <p>2.1.1.2. The system tells the actor that the time is over</p> <p>2.1.1.3. Step 2.1.1 runs</p> <p>2.1.3.1. The system indicates that it is not the right answer.</p> <p>2.1.3.2. It runs step 2.1.1</p>
Postconditions	The ranking stored in the session has been updated.

Code	UC09
Name	Display the rules
Author	Sergio Baena López
Date	2014-05-08
Description	The rules of a game is shown.
Actors	Student
Preconditions	The user session must be open.
Normal flow	<ol style="list-style-type: none"> <li>1. The actor requested the option to see the rules of a particular game.</li> <li>2. The system displays on screen rules of the game in particular, and a button to play this game. <ol style="list-style-type: none"> <li>2.1. The actor clicks on the <i>Jugar ahora</i>. <ol style="list-style-type: none"> <li>2.1.1. Runs the use case specific game.</li> </ol> </li> <li>2.2. The actor did not click anything.</li> </ol> </li> </ol>
Alternative flow	
Postconditions	Nothing

Code	UC10
Name	Enter admin page
Author	Sergio Baena López
Date	2014-05-08
Description	Page links to all the functionalities of the administrator.
Actors	Webmaster
Preconditions	The user session must be open.
Normal flow	<ol style="list-style-type: none"> <li>1. The actor requested the option to enter the site administrator.</li> <li>2. The system directs the actor to the administrator page.</li> <li>3. The actor displays the page.</li> <li>4. The actor executed some of the functionality of the administrator.</li> </ol>
Alternative flow	Nothing
Postconditions	An administrator functionality is running.

Code	UC11
Name	Management of words
Author	Sergio Baena López
Date	2014-05-08
Description	The actor can add, view, modify and delete a word. He can also delete all the words.
Actors	Webmaster
Preconditions	The user session must be open.
Normal flow	<p>1. The actor has a menu <i>Management of words</i> from which hang from it options to <i>Añadir</i>, <i>Buscar</i> and <i>Eliminar todo</i>.</p> <p>1.1. The actor clicks on <i>Añadir</i></p> <p>1.1.1. The system displays a form with three fields, namely, <i>palabra</i>, <i>número de sílabas</i> and <i>categoría</i>.</p> <p>1.1.2. The actor introduces the new word, the number of syllables and category (select).</p> <p>1.1.3. The actor clicks the button <i>Añadir</i></p> <p>1.1.4. The system validates the data.</p> <p>1.1.5. The system makes an INSERT with the new word.</p> <p>1.1.6. The system displays the message: <i>La palabra se ha añadido correctamente</i>.</p> <p>1.1.7. The form is reset.</p> <p>1.2. The actor clicks on <i>Buscar</i>:</p> <p>1.2.1. The system displays a form with two fields, namely, <i>Criterio de búsqueda</i> and <i>Valor</i>.</p> <p>1.2.2. The actor selects the criteria (id, <i>valor</i>, <i>número de sílabas</i>, <i>categoría</i>) and writes the value to search.</p> <p>1.2.3. The actor clicks the <i>Buscar</i> button</p> <p>1.2.4. The system goes to the table <i>word</i> and does a LIKE.</p> <p>1.2.5. The system displays a list of search results with this format: <i>&lt;within result criteria&gt; &lt;link to see&gt; &lt;link to modify&gt; &lt;link to remove&gt;</i>.</p> <p>1.2.5.1. The actor clicks on <i>Ver</i>.</p> <p>1.2.5.1.1. The system displays a read-only form identical to add. The information it contains are those of the selected word.</p> <p>1.2.5.2. The actor clicks on <i>Modificar</i>.</p> <p>1.2.5.2.1. The system displays a form identical to add. The information it contains are those of the selected word.</p> <p>1.2.5.2.2. The actor makes some changes.</p> <p>1.2.5.2.3. The actor clicks <i>Modificar</i>.</p> <p>1.2.5.2.4. The system validates the data.</p> <p>1.2.5.2.5. The system makes an UPDATE to the table <i>word</i>.</p> <p>1.2.5.2.6. The system displays a message: <i>La palabra se ha modificado correctamente</i>.</p> <p>1.2.5.3. The actor clicks on <i>Eliminar</i>.</p> <p>1.2.5.4. The system launches a <i>confirm</i> asking if you are sure you want to delete.</p>

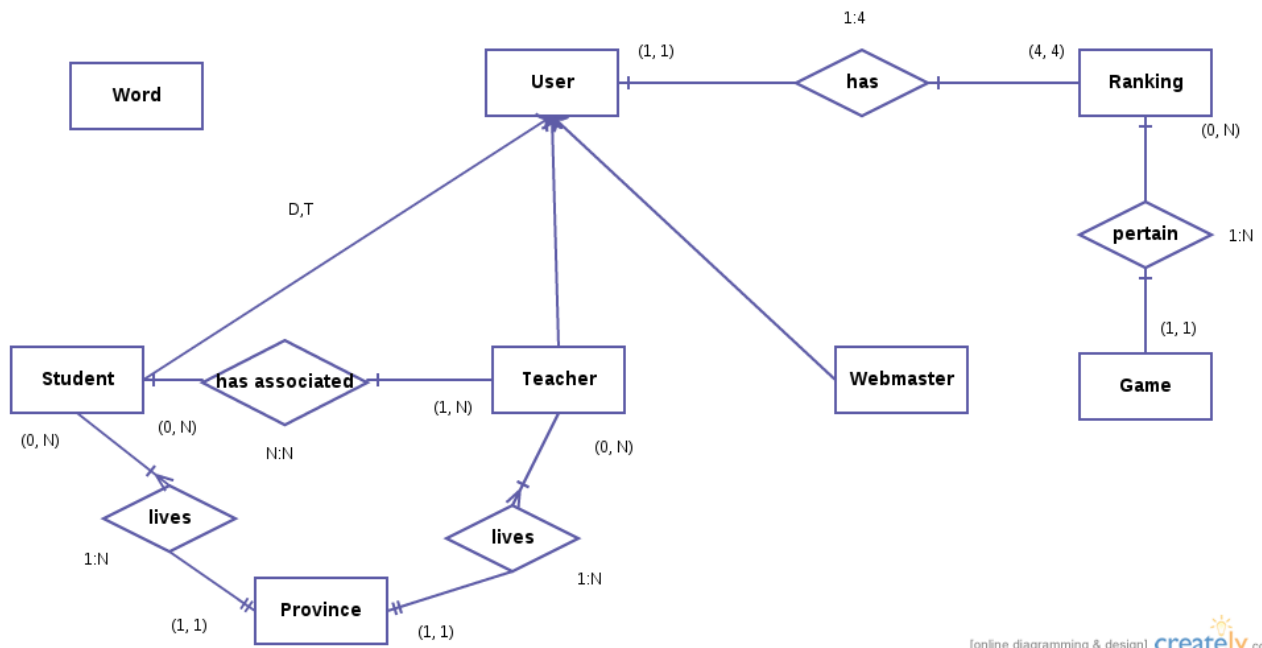


	<p>1.2.5.5. The system does a <i>DELETE</i> on the table <i>word</i>.</p> <p>1.2.5.6. The system displays a message: <i>La palabra se ha eliminado correctamente</i>.</p> <p>1.2.5.7. The system deletes the entry for that word in the search result.</p> <p>1.3. The actor clicks on <i>Eliminar todo</i>.</p> <p>1.3.1. The system launches a <i>confirm</i> asking if he is sure he wants to delete all.</p> <p>1.3.2. The actor indicates <i>yes</i>.</p> <p>1.3.3. The system does a <i>DELETE</i> all records in the table <i>word</i>.</p> <p>1.3.4. The system displays a message: <i>Todas las palabras se han eliminado correctamente</i>.</p>
Alternative flow	<p>1.1.4.1. The system detects any wrong data.</p> <p>1.1.4.2. The system displays an error message indicating the wrong fields.</p> <p>1.1.4.3. The system marks the wrong fields in red.</p> <p>1.2.4.1. The system detects that there is no match.</p> <p>1.2.4.2. The system will then display a message: <i>No hay coincidencias</i>.</p> <p>1.2.4.3. The system resets the form.</p> <p>1.2.5.2.4.1. Step <i>1.1.4.1</i> is executed</p> <p>1.2.5.4.1. The actor says he is not sure.</p> <p>1.3.1.1. The actor indicates <i>no</i>.</p>
Postconditions	The database was updated (if the actor has only seen the word not this is true).

Code	UC12
Name	Management of system users
Author	Sergio Baena López
Date	2014-05-09
Description	The actor can add, view, modify and delete an user. He can also delete all the users.
Actors	Webmaster
Preconditions	The user session must be open.
Normal flow	<p>1. The actor has a menu <i>Management of users</i> from which hang from it options to <i>Añadir</i>, <i>Buscar</i> and <i>Eliminar todo</i>.</p> <p>1.1. The actor clicks on <i>Añadir</i></p> <p>1.1.1. The system displays an identical form to register.</p> <p>1.1.2. The actor introduces the username, password (two times), name, surnames, school, city and more data. Also select what type of user the actor wants to add.</p> <p>1.1.3. The actor clicks the button <i>Añadir</i></p> <p>1.1.4. The system validates the data.</p> <p>1.1.5. The system makes some INSERT's for the new user.</p> <p>1.1.6. The system displays the message: <i>El usuario se ha añadido correctamente</i>.</p> <p>1.1.7. The form is reset.</p> <p>1.2. The actor clicks on <i>Buscar</i>:</p> <p>1.2.1. The system displays a form with two fields, namely, <i>Criterio de búsqueda</i> and <i>Valor</i>.</p> <p>1.2.2. The actor selects the criteria (id, <i>nombre</i>, <i>apellidos</i>, etc) and writes the value to search.</p> <p>1.2.3. The actor clicks the <i>Buscar</i> button</p> <p>1.2.4. The system goes to the DB and does a LIKE.</p> <p>1.2.5. The system displays a list of search results with this format:  &lt;within result criteria&gt; &lt;link to see&gt; &lt;link to modify&gt; &lt;link to remove&gt;.</p> <p>1.2.5.1. The actor clicks on <i>Ver</i>.</p> <p>1.2.5.1.1. The system displays a read-only form identical to add. The information it contains are those of the selected user.</p> <p>1.2.5.2. The actor clicks on <i>Modificar</i>.</p> <p>1.2.5.2.1. The system displays a form identical to add. The information it contains are those of the selected user.</p> <p>1.2.5.2.2. The actor makes some changes.</p> <p>1.2.5.2.3. The actor clicks <i>Modificar</i>.</p> <p>1.2.5.2.4. The system validates the data.</p> <p>1.2.5.2.5. The system makes an UPDATE to the DB.</p> <p>1.2.5.2.6. The system displays a message: <i>El usuario se ha modificado correctamente</i>.</p> <p>1.2.5.3. The actor clicks on <i>Eliminar</i>.</p> <p>1.2.5.4. The system launches a <i>confirm</i> asking if you are sure you want to delete.</p>

	<p>1.2.5.5. The system does a <i>DELETE</i> on the DB.</p> <p>1.2.5.6. The system displays a message: <i>El usuario se ha eliminado correctamente.</i></p> <p>1.2.5.7. The system deletes the entry for that user in the search result.</p> <p>1.3. The actor clicks on <i>Eliminar todo</i>.</p> <p>1.3.1. The system launches a <i>confirm</i> asking if he is sure he wants to delete all.</p> <p>1.3.2. The actor indicates <i>yes</i>.</p> <p>1.3.3. The system does a <i>DELETE</i> all records in the table <i>User, Student, Teacher or Webmaster</i>.</p> <p>1.3.4. The system displays a message: <i>Todos los usuarios se han eliminado correctamente.</i></p>
Alternative flow	<p>1.1.4.1. The system detects any wrong data.</p> <p>1.1.4.2. The system displays an error message indicating the wrong fields.</p> <p>1.1.4.3. The system marks the wrong fields in red.</p> <p>1.2.4.1. The system detects that there is no match.</p> <p>1.2.4.2. The system will then display a message: <i>No hay coincidencias.</i></p> <p>1.2.4.3. The system resets the form.</p> <p>1.2.5.2.4.1. Step <i>1.1.4.1.</i> is executed</p> <p>1.2.5.4.1. The actor says he is not sure.</p> <p>1.3.1.1. The actor indicates <i>no</i>.</p>
Postconditions	The database was updated (if the actor has only seen the user not this is true).

Code	UC13
Name	View of student profile
Author	Sergio Baena López
Date	2014-05-09
Description	View of student profile
Actors	Teacher
Preconditions	The user session must be open.
Normal flow	<ol style="list-style-type: none"> <li>1. The actor requested the option to see the profile of the student.</li> <li>2. The system displays a search form associated to its students. This form has two fields, namely, <i>Criterio de búsqueda</i> and <i>Valor</i>.</li> <li>3. The actor selects the criteria (id, <i>nombre</i>, <i>apellidos</i>, etc) and writes the value to search.</li> <li>4. The actor clicks the <i>Buscar</i> button</li> <li>5. The system goes to the DB and does a LIKE.</li> <li>6. The system displays a list of search results with this format: <i>&lt;within result criteria&gt; &lt;link to see&gt;</i></li> <li>7. The actor clicks on <i>Ver</i> link of a student.</li> <li>8. The system displays a read-only form selected student profile (same style as specified in the <i>UC04</i>).</li> </ol>
Alternative flow	<ol style="list-style-type: none"> <li>5.1. The system detects that there is no match.</li> <li>5.2. The system will then display a message: <i>No hay coincidencias</i>.</li> <li>5.3. The system resets the form.</li> </ol>
Postconditions	Nothing



[online diagramming & design] [creately.com](https://creately.com)

User			
Name	Type	Default values	Possibles values
Id	INT(6)	-	-
Username	VARCHAR(10)	-	-
Password	VARCHAR(200)	-	-
Name	VARCHAR(50)	-	-
Surnames	VARCHAR(100)	-	-

Student			
Name	Type	Default values	Possibles values
School	VARCHAR(50)	-	-
city	VARCHAR(50)	-	-
Course (NL)	VARCHAR(2)	-	-
date_of_birth	DATE	-	-

Teacher			
Name	Type	Default values	Possibles values
School	VARCHAR(50)	-	-
city	VARCHAR(50)	-	-
Courses (NL, NL, ...)	VARCHAR(50)	-	-

Webmaster			
Name	Type	Default values	Possibles values
Role	VARCHAR(20)	-	-
Description_of_their_role	VARCHAR(100)	-	-

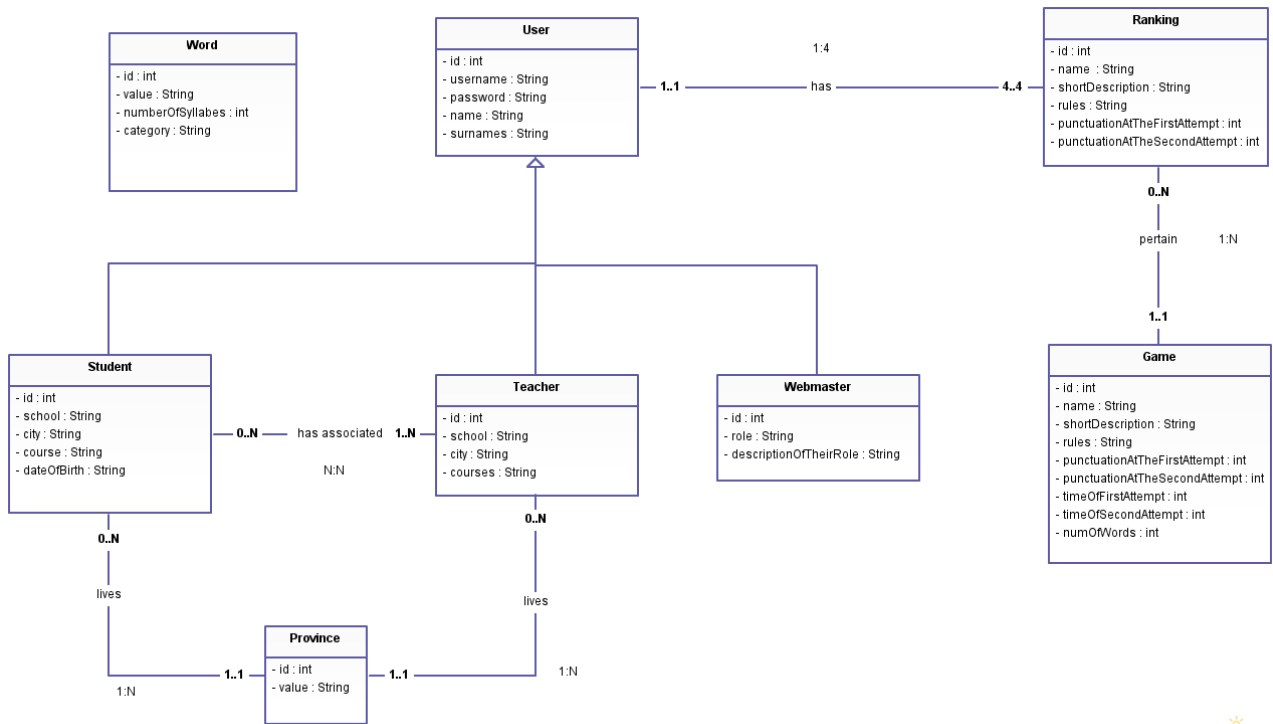
Ranking			
Name	Type	Default values	Possibles values
Id	INT(6)	-	-
points	INT(20)	0	-
number_of_hits	INT(10)	0	-
number_of_failures	INT(10)	0	-
number_of_attempts	INT(10)	0	-

Game			
Name	Type	Default values	Possibles values
Id	INT(6)	-	-
Name	VARCHAR(50)	-	-
Short_description	VARCHAR(100)	-	-
Rules	VARCHAR(200)	-	-
Punctuation_at_the_first_attempt	INT(2)	-	-
Punctuation_at_the_second_attempt	INT(2)	-	-
Time_of_first_attempt	INT(2)	-	-
Time_of_second_attempt	INT(2)	-	-
num_of_words	INT(2)	-	-

Word			
Name	Type	Default values	Possibles values
Id	INT(6)	-	-
Value	VARCHAR(20)	-	-
number_of_syllables	INT(1)	-	-
category	VARCHAR(16)	-	'aguda', 'llana', 'esdr&uacute;jula'

Province			
Name	Type	Default values	Possibles values

Id	INT(6)	-	-
Value	VARCHAR(20)	-	-





# TEST CASES

Name	TC01 - Register on the system
Creator	Sergio Baena López
Version	1.0
Description / purpose	Register as a new user in the system
Actor	Unauthenticated user
Test data	Name [valid, invalid], surnames [valid, invalid], school [valid, invalid], city [valid, invalid], username [valid, invalid], password [valid, invalid], course [valid, invalid], date of birth [valid, invalid], courses [valid, invalid]
Precondition	Nothing

Condition	Valid classes	Invalid classes
Name	Field containing only letters and spaces (1)	<ul style="list-style-type: none"> <li>* Empty field (2)</li> <li>* Contains only spaces (3)</li> <li>* Contain any character that is not letter or space. (4)</li> </ul>
Surnames	Field containing only letters and spaces (5)	<ul style="list-style-type: none"> <li>* Empty field (6)</li> <li>* Contains only spaces (7)</li> <li>* Contain any character that is not letter or space. (8)</li> </ul>
School	Not empty string neither contains only spaces. (9)	<ul style="list-style-type: none"> <li>* Empty string (10)</li> <li>* Contains only spaces (11)</li> </ul>
City	Not empty string neither contains only spaces (12)	<ul style="list-style-type: none"> <li>* Empty string (13)</li> <li>* Contains only spaces (14)</li> </ul>
Username	First character must be a letter of the English alphabet (uppercase or lowercase), and the following characters may be as a the first character or contain a number or point a dash or underscore. The minimum length of 4 characters. (15)	<ul style="list-style-type: none"> <li>* We did not start by letter of the English alphabet. (16)</li> <li>* The second character or one of the following is not a letter of the English alphabet, a number, a dot, a dash or underscore. (17)</li> <li>* Length less than 4. (18)</li> </ul>
Password	Length greater than 6, and match the confirmation password. (19)	<ul style="list-style-type: none"> <li>* Smaller than 7 length.(20)</li> <li>* The password does not match the confirmation password.(21)</li> </ul>
Course	Number from 1-6 and a letter of the English alphabet. (22)	<ul style="list-style-type: none"> <li>* The number is out of range. (23)</li> <li>* The next character after the number is not a letter of the English alphabet. (24)</li> <li>* Where a number is expected to enter otherwise. (25)</li> <li>* Different length to 2. (26)</li> </ul>
Date of birth	Last date with <i>DD/MM/YYYY</i> format.(27)	<ul style="list-style-type: none"> <li>* Format is not followed.(28)</li> <li>* Impossible date.(29)</li> <li>* Today's Date.(30)</li> <li>* Future date.(31)</li> </ul>
Courses	Right courses separated by comma and space. (32)	<ul style="list-style-type: none"> <li>* Any course is incorrect.(33)</li> <li>* Format is not followed.(34)</li> <li>* Empty string(35)</li> </ul>

Name	TC02 - Login
Creator	Sergio Baena López
Version	1.0
Description / purpose	An user authentication system.
Actor	Unauthenticated user
Test data	Username and password [valid, invalid]
Precondition	The actor has to be registered in the system.

Condition	Valid classes	Invalid classes
Username and password	The combination of user name (valid username) and password (valid password) exist in the database. (1)	* The combination of user name and password does not exist in the database. (2) * Is invalid username (3) * Is invalid password (4)

Management his profile is identical to the register.

Name	TC03 - Management of words
Creator	Sergio Baena López
Version	1.0
Description / purpose	The actor can add, view, modify and delete a word. He can also delete all the words.
Actor	Webmaster
Test data	Word[valid, invalid], number of syllables [valid, invalid] and category [valid, invalid]
Precondition	The user session must be open.

Condition	Valid classes	Invalid classes
Word	Only letters.(1)	* Contains something that is not letter.(2) * Empty string.(3)
Number of syllables	Integer between 0 and 10. (4)	* Integer out of range.(5) * There is an decimal number. (6) * It is something other than a number.(7)
Category	Valores admitidos: 'Aguda', 'Llana' o 'Esdrújula'(8)	No es un valor admitido.(9)

Sprint 12 – 16th May									
Id	Description	Owner	Importance	Estimate (hours)	Date start	Date end	Real time	Cost (hours)	Status
1	Add sounds to register page and key capture	Sergio	High	6	12/05/14	14/05/14	10		DONE
2	Create interface of login page	Sergio	High	1	15/05/14	15/04/14			DONE
3	Create functionality of login page (JS)	Sergio	High	2	15/04/14	15/04/14	3		DONE
4	Create functionality of login page (PHP)	Sergio	High	2	16/04/14	16/04/14	2		DONE



Sprint 12 – 16th May									
Id	Description	Owner	Importance	Estimate (hours)	Date start	Date end	Real time	Cost (hours)	Status
1	Add sounds to register page and key capture	Sergio	High	6	12/05/14	14/05/14	10		DONE
2	Create interface of login page	Sergio	High	1	15/05/14	15/04/14			DONE
3	Create functionality of login page (JS)	Sergio	High	2	15/04/14	15/04/14	3		DONE
4	Create functionality of login page (PHP)	Sergio	High	2	16/04/14	16/04/14	2		DONE

Sprint 26 – 30th May – 04th May									
Id	Description	Owner	Importance	Estimate (hours)	Date start	Date end	Real time	Cost (hours)	Status
1	Create interface of the game	Sergio	Low	1	26/05/14	26/05/14	2		DONE
2	Create functionality of game	Sergio	High	4	26/05/14	26/05/14	9		DONE
3	Add sounds to game and key capture	Sergio	High	4	27/05/14	27/05/14	6		DONE
4	Create interface of the end game	Sergio	Low	1	28/05/14	28/05/14	1		DONE
5	Create functionality of end game (JS)	Sergio	High	3	28/05/14	28/05/14	3		DONE
6	Create functionality of end game (PHP)	Sergio	High	2	29/05/14	29/05/14	2		DONE
7	Add sounds to end game	Sergio	High	3	29/05/14	30/05/14	5		DONE

	page and key capture								
--	----------------------	--	--	--	--	--	--	--	--

Sprint 02 – 06th April – 04th May									
Id	Description	Owner	Importance	Estimate (hours)	Date start	Date end	Real time	Cost (hours)	Status
1	Create functionality of logout	Sergio	Low	5	02/06/14	03/06/14	6		DONE
2	Add sounds to logout	Sergio	High	2	04/06/14	04/06/14	3		DONE

# Problems and solutions

- **Sounds:** We have had problems with the sounds because it's something that we have barely touched.
  - One problem is that I was wanting to stop all sounds from the web and try it threw an exception. A loop that ran through a list of sounds and call its "pause()" method and then set your "currentTime" attribute to 0 was not correct. The reason is that sound behaves asynchronously. The solution was to capture an event, "onpause" to only set the "currentTime" attribute when the sound degas paused.
  - Another problem I had with the "duration" attribute. When used, it did not work. The reason, very similar to the above. I had to capture an event to avoid trying to get the duration before he had time to load.
- **Ajax:** Another problem was with Ajax. I went several times the error 500, which gives no hint of what was the cause of the error. The solution to resolve the server failure without going blind was to use the Apache log.
- **HTML entities.** The database contains words coded HTML, to avoid encoding problems. I had problems when I wanted to pass a string with HTML entities to sounds. The "espeak" command did not do well. The solution was to use a JavaScript library "encoder.js" to decode the string so that the command can interpret well.

# Final conclusions

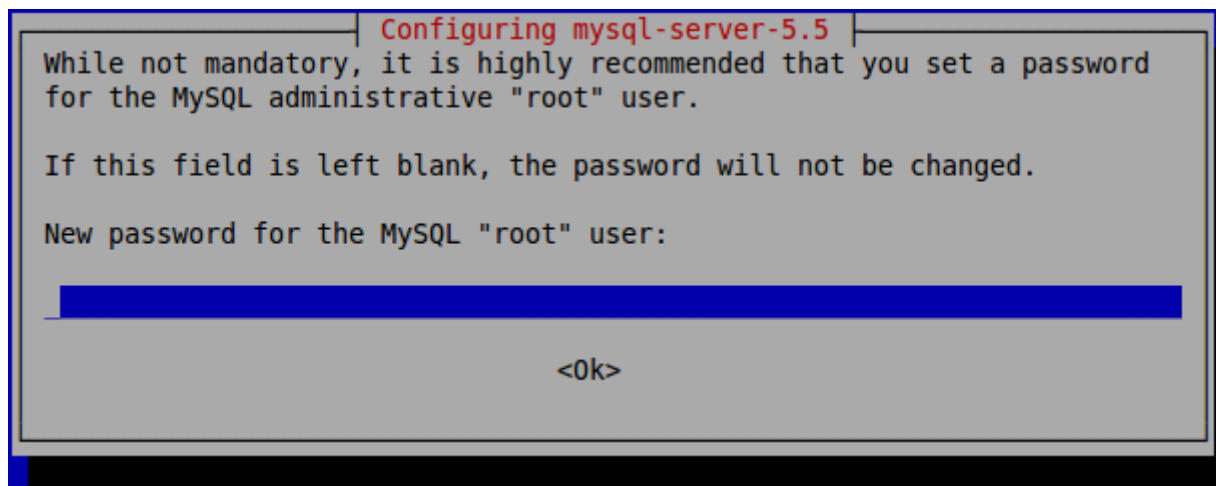
I liked to have learned a way to make a page also accessible to the blind. Good to know make this pages also, apart from the usual. It has also been nice to start learning HTML5, which is the future of web programming. The "audio" tag is a very powerful and interesting tag and have already started to use it's fine. Also one of the first experiences in making a real project, which among other things, is knowing how to combine several languages properly for a big objective.

In conclusion, there has been a great experience.

1) Primero de todo, descargamos e instalamos el paquete *apache2* con el comando *sudo apt- get install apache2*.

2) A continuación, descargamos e instalamos el paquete *php5* con el comando *sudo apt- get install php5*.

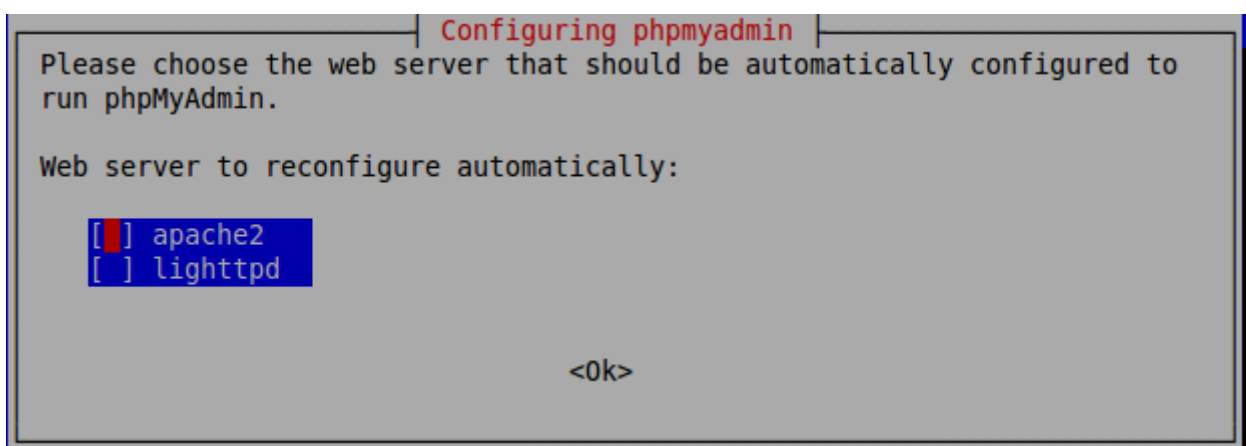
3) Para instalar *mysql*, tenemos que descargar e instalar los siguientes paquetes, a saber, *mysql-server*, *mysql-client*. En primer lugar, descargamos e instalamos el paquete *mysql-server* con la orden: *sudo apt-get install mysql-server*. Durante la instalación, la terminal nos solicita que introduzcamos una contraseña para el usuario *root* de MySQL:



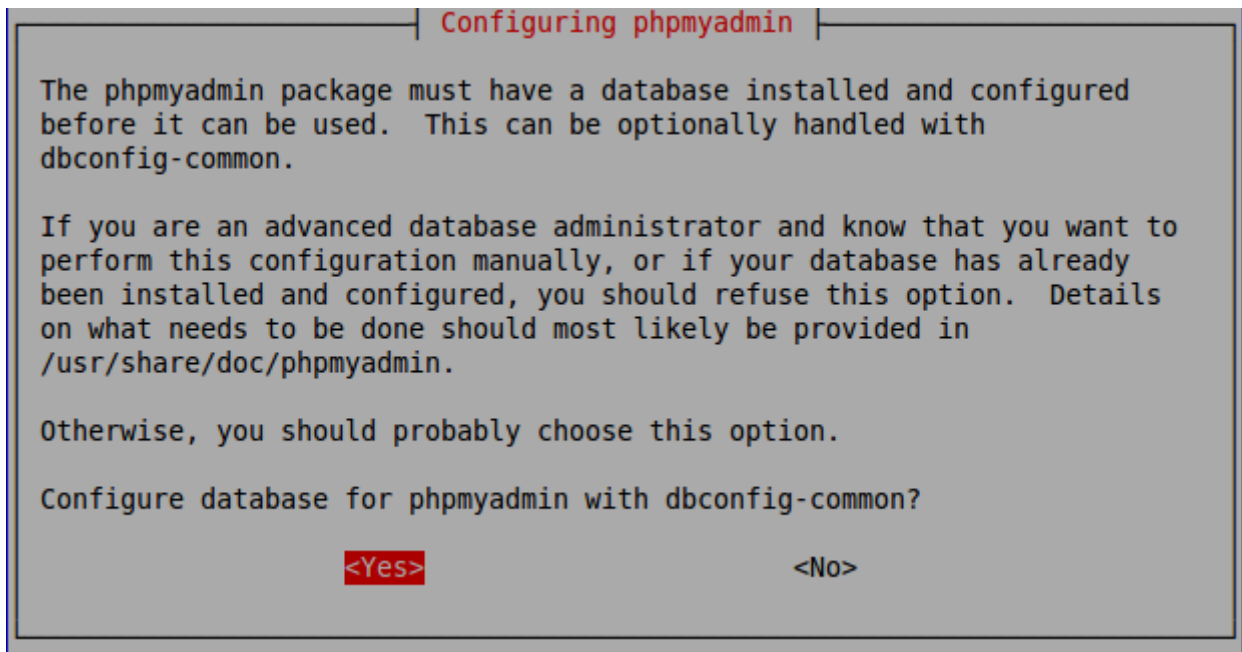
La introducimos, la confirmamos en otra ventana similar y dejamos que la shell termine de instalar y configurar el paquete.

4) Instalamos el siguiente paquete de MySQL, *mysql-client*, con la sentencia *sudo apt-get install mysql-client*.

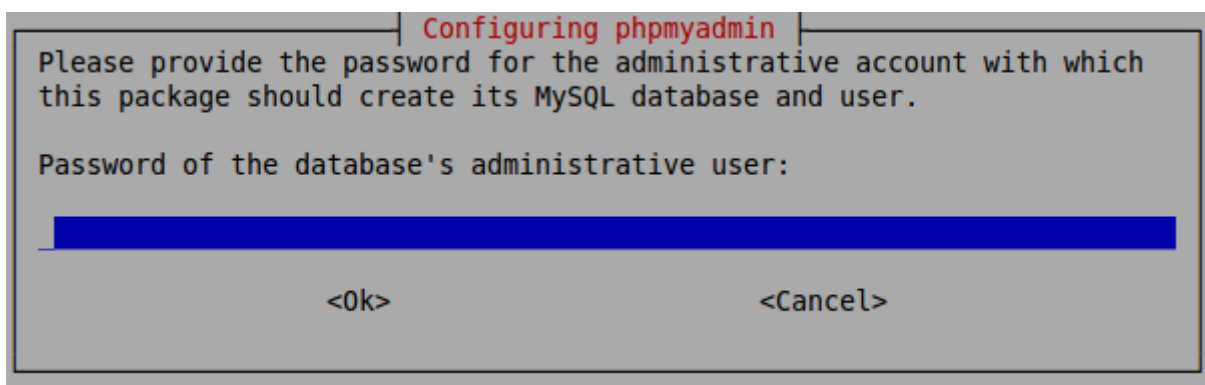
5) Ahora, toca instalar el paquete *phpmyadmin*. Introducimos en la consola *sudo apt-get install phpmyadmin*. En medio del proceso de instalación, nos saldrá este ventana:



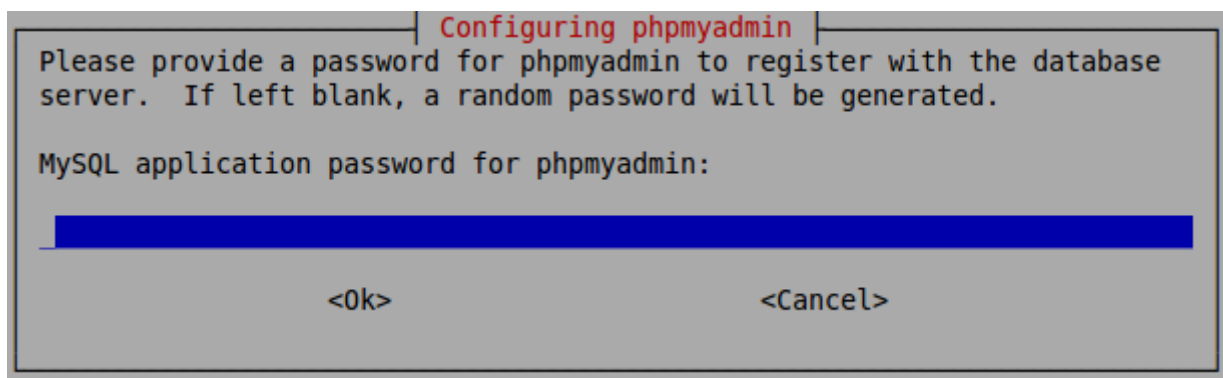
Esta ventana nos pregunta qué servidor web queremos reconfigurar para ejecutar el software *phpMyAdmin*. Seleccionamos el servidor *apache2* con la barra espaciadora y con el tabulador hacemos clic en *<Ok>*. Después de un pequeño rato, la terminal vuelve a lanzar una nueva ventana para nosotros:



Nos pregunta aquí si queremos que se nos configure la base de datos para poder ser usada por *phpmyadmin* de forma automática por el paquete *dbconfig-common* o si preferimos nosotros ser lo que lo hagamos. Seleccionamos *Yes* para que se haga automáticamente. Después de ello, casi al instante, vuelve a salir otra ventana:



Introducimos una contraseña para el usuario administrador de la base de datos. Y, después, se lanza esta ventana:



Introducimos una contraseña para la aplicación MySQL para phpmyadmin. A continuación, y después de confirmarla, se continúa ejecutando APT hasta que se finaliza la instalación de

*phpmyadmin.*

6) Instalamos el convertidor de cadenas a sonidos (espeak) . Ejecutamos el comando *sudo apt-get install espeak*

7) Instalamos ahora el conversor de formato de audios *lame*. Ejecutamos el comando *sudo apt-get install lame*

Ya hemos acabado de instalar todo lo necesario y, en teoría, tendría que funcionar todo. Vamos a comprobarlo.

- 1) Comprobamos que el servidor Apache funciona, para ello, escribimos en el navegador la ruta *localhost* y tendría que salir la página índice de nuestro servidor. Y, como podemos ver, se muestra lo que debería mostrarse:

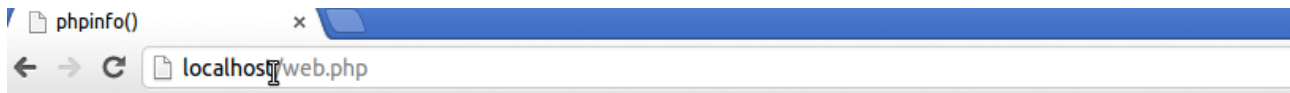



## It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

- 2) Comprobamos que el intérprete de PHP funcione y este asociado con Apache. Esto lo podemos hacer ejecutando un código PHP:



<b>PHP Version 5.4.9-4ubuntu2.3</b>		
<b>System</b>	Linux lub13 3.8.0-19-generic #29-Ubuntu SMP Wed Apr 17 18:19:42 UTC 2013 i686	
<b>Build Date</b>	Sep 4 2013 19:16:03	
<b>Server API</b>	Apache 2.0 Handler	
<b>Virtual Directory Support</b>	disabled	
<b>Configuration File (php.ini) Path</b>	/etc/php5/apache2	
<b>Loaded Configuration File</b>	/etc/php5/apache2/php.ini	
<b>Scan this dir for additional .ini files</b>	/etc/php5/apache2/conf.d	
<b>Additional .ini files parsed</b>	/etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-gd.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysql.ini, /etc/php5/apache2/conf.d/mcrypt.ini	
<b>PHP API</b>	20100412	
<b>PHP Extension</b>	20100525	

- 3) Por último, comprobamos que MySQL y phpMyAdmin funciona y este asociado a Apache:





Welcome to phpMyAdmin

Language

English

Log in

Username:

Password:

Go

phpMyAdmin



(Recent tables) ...

information\_schema  
mysql  
performance\_schema  
phpmyadmin  
test

localhost

Databases SQL Status Users Export Import Settings More

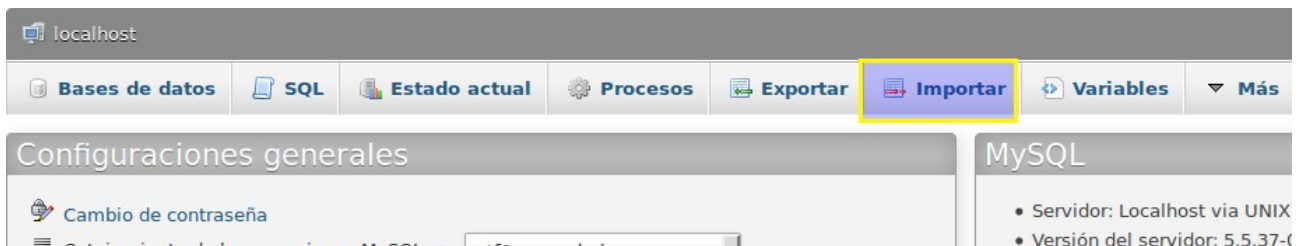
Users having access to "test"

User	Host	Type	Privileges	Grant	Action
Any	%	database-specific	SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, REFERENCES, INDEX, ALTER, CREATE TEMPORARY TABLES, LOCK TABLES, CREATE VIEW, EVENT, TRIGGER, SHOW VIEW, CREATE ROUTINE	No	Edit Privileges
debian-sys-maint	localhost	global	ALL PRIVILEGES	Yes	Edit Privileges
root	127.0.0.1	global	ALL PRIVILEGES	Yes	Edit Privileges
root	:::1	global	ALL PRIVILEGES	Yes	Edit Privileges
root	localhost	global	ALL PRIVILEGES	Yes	Edit Privileges
root	lub13	global	ALL PRIVILEGES	Yes	Edit Privileges

New

Para empezar a utilizar mi aplicación web, ¿qué necesitamos hacer, una vez está todo instalado?






- 1) Obtén el directorio contenedor de toda mi aplicación web.
- 2) Lo mueves en la raíz del servidor web. Si está configurado por defecto, hay que dirigirse a `/var/www/`
- 3) Inicias el servicio de *apache2* en el caso que no esté ya iniciado mediante el comando `sudo service apache2 start`.
- 4) Inicias el servicio de *mysql* en el caso que no esté ya iniciado mediante el comando `sudo service mysql start`.
- 5) Nos dirigimos a *phpmyadmin*. Donde pone *Importar* nos dirigimos para importar los ficheros SQL. Primero el *create\_database.sql* y luego el *create\_tables.sql*. Importante, obviamente, cuando importamos el segundo script, hay que, previamente, estar dentro de la base de datos que acaba de crear el script *create\_database.sql* (*letters\_game\_db*).



6) Por último, abrimos un navegador web (versión que soporte la etiqueta *audio* de HTML5) y escribimos la dirección <http://localhost/lettersGame>.

## Browser Support

The numbers in the table specifies the first browser version that fully supports the element.

Element					
<audio>	4.0	9.0	3.5	4.0	10.5

# Innovation

*My application has innovative use of multimedia on the web without using the outdated "Adobe Flash Player". We made use of the "audio" tag of HTML5.*