

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

**Diseño e implementación de una infraestructura de
gamificación para aplicaciones educativas**

**M^a Ángeles Cano Rubio
Tutor: Xavier Alamán Roldán**

FEBRERO 2017

Diseño e implementación de una infraestructura de gamificación para aplicaciones educativas

AUTOR: M^a Ángeles Cano Rubio
TUTOR: Xavier Alamán Roldán

Grupo de Herramientas Interactivas Avanzadas
Dpto. Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Febrero de 2017

Resumen (castellano)

La gamificación es un elemento cada vez más presente en nuestro día a día, muchas grandes compañías han implantado sistemas basados en gamificación para lograr una mayor colaboración con sus empleados. Aplicar estos principios a la educación consigue grandes aportes tanto para los alumnos como para los profesores, obteniendo un mejor ambiente en las aulas, una mayor cohesión entre los alumnos y un gran aumento en el rendimiento académico.

En este trabajo se explica el desarrollo de una infraestructura basada en gamificación orientada para entornos educativos, explicando las tecnologías empleadas y su funcionamiento. Se propone un sistema fácil de usar para los profesores que es altamente modificable pero que a la vez está listo para usar sin realizar cambios. Para la parte de los alumnos se ofrece una interfaz muy atractiva, con elementos que captan su atención que hagan que desee usar todos los días la aplicación. El sistema ha sido desarrollado en PHP, CSS y JavaScript.

Cómo parte complementaria a la web se implementó un módulo externo que se encarga de realizar la conexión entre OpenSim y la base de datos de la aplicación web. Este módulo de comunicación está desarrollado en LSL para la parte de OpenSim y en PHP en la parte que se encarga de recibir y transmitir la información.

Por último, se realizó una batería de pruebas tanto para la parte web como para el módulo de comunicación y en ambos casos se obtuvieron resultados satisfactorios. El sistema está pendiente de implantación para los próximos dos meses.

Abstract (English)

Gamification is an element that is more present in our lives day by day, very huge companies have applied gamification into their systems to get and keep their employees collaboration. Applying these points into education would give great profits to both teachers and students, gaining better behaviour in the classroom, stronger cohesion between students and a great increase in their academic performance.

In this Bachelor Thesis is explained the performance of an infrastructure based on gamification oriented to academic environments, explaining the used technologies and its behaviour. It is proposed a user-friendly system for teachers that is highly modifiable but at the same time is ready to use without making changes. For the students part a very attractive interface is offered, with elements that capture their attention that make them want to use the application every day. The system has been developed in PHP, CSS and JavaScript.

As a counterpart of the web, an external module was implemented that is responsible for making the connection between OpenSim and the database of the web application. This communication module has been developed in LSL for OpenSim and in PHP for the part that exchanges information.

Finally, a battery of tests was carried out both for the web part and for the communication module and in both cases satisfactory results were obtained. The system is pending implementation for the next two months.

Palabras clave (castellano)

Gamificación, gamificación en educación, web, mundos virtuales

Keywords (inglés)

Gamification, gamification in education, web, virtual environments

Agradecimientos

Quiero dar las gracias a mis padres y mi abuela por estar apoyándome durante estos años de estudio. También gracias a mis amigos por estar ahí siempre que les he necesitado.

Por último, gracias a Xavier Alamán y María José Lasala por ofrecerme la oportunidad de realizar este trabajo y que se lleve a cabo su implantación.

INDICE DE CONTENIDOS

1	Introducción.....	1
1.1	Motivación.....	1
1.2	Objetivos.....	1
1.3	Organización de la memoria.....	2
2	Estado del arte	3
2.1	Gamificación	3
2.1.1	Gamificación en la educación.....	4
2.2	Mundos virtuales en la educación	5
2.2.1	OpenSim	6
2.3	Conclusiones.....	6
3	Requerimientos y Diseño	7
3.1	Requerimientos.....	7
3.1.1	Sistema usado en el aula.....	8
3.1.1.1	Profesores	8
3.1.1.2	Alumnos	10
3.2	Diseño.....	11
3.2.1	Base de Datos	11
3.2.2	Sistema usado en el aula.....	13
3.2.3	Módulo de comunicación	15
4	Desarrollo	16
4.1	Página web.....	16
4.2	Módulo de comunicación	20
5	Integración, pruebas y resultados	23
6	Conclusiones y trabajo futuro.....	24
6.1	Conclusiones.....	24
6.2	Trabajo futuro	24
	Referencias	25
	Glosario	27
	Anexos.....	I
A	Configuración de Apache y PHP para el funcionamiento de la web.....	I
B	Conexión de OpenSim con una base de datos externa. Módulo de comunicación. V	
C	Casos de uso de la pantalla de habilidades de los alumnos.....	- 1 -

INDICE DE FIGURAS

FIGURA 2-1.1:	MAPA BÚSQUEDAS DE “GAMIFICATION”	3
FIGURA 2-2.2:	GRÁFICO BÚSQUEDAS DE “GAMIFICATION”.....	4
FIGURA 2-3.3:	GRÁFICO BÚSQUEDAS DE “WHAT IS GAMIFICATION”	4
FIGURA 2-4.4:	GRÁFICO BÚSQUEDAS DE “GAMIFICATION EDUCATION”.....	5

FIGURA 2-5.5: PRESENTACIÓN DE SLOODLE EN OPENSIM.....	6
FIGURA 3-1.1.1.1: PANTALLA EINES (HERRAMIENTAS) DEL PROFESOR.....	18
FIGURA 3-2.1.2.1: PANTALLA PRINCIPAL DE ALUMNO.....	18
FIGURA 3-3.1.2.1: PANTALLA SCRIPTORIUM	19
FIGURA 3-4.1.2.1: PANTALLA DE VISOR DE GRUPOS	19
FIGURA 3-5.1.2.1: PANTALLA PRINCIPAL DE UN ALUMNO EN LA CÁRCEL	19

INDICE DE TABLAS

TABLA 4-2: EJEMPLOS DE COMANDOS DEL TÓTEM MÁSTER.....	18
---	----

1 Introducción

1.1 Motivación

La gamificación cada vez está más presente en nuestro entorno, gran parte de las compañías que nos rodean han implantado la gamificación para conseguir un mayor compromiso con sus empleados. La gamificación consiste en aplicar técnicas y metodologías de juegos a otras actividades, en el caso de este documento, la enseñanza. Un estudio de Project Tomorrow for Speak Up reveló un cambio en el ámbito educativo, en 2010 un 47% de profesores usaba videos online en su clase; esta cifra creció a un 68% en 2015. [1]

Un claro ejemplo de gamificación en el aula es Classcraft, un entorno en el que se favorece el trabajo en equipo con recompensas para poder customizar tu avatar a tu gusto. Las implantaciones de estos sistemas en las aulas han tenido respuestas muy positivas tanto por parte de los alumnos como de los profesores, generando un entorno más favorable para el aprendizaje. La compañía Merck Sharp & Dohme (MSD) se convirtió en la primera compañía mundial en implantar cursos basados en gamificación. Los resultados de este curso de 5 años y 760 empleados fueron excelentes: las habilidades que cubrían el curso mejoraron en un 25% y la productividad de los empleados subió un 8% [2]

Otra herramienta también muy utilizada en los entornos educativos son los mundos virtuales. En este TFG hablaremos sobre todo de OpenSim, una plataforma que adopta los protocolos de Second Life y que además es de código libre. Permite una gran variedad de opciones ya que a través de scripts podemos llegar a modificar nuestros objetos de tal modo que se conviertan en elementos útiles para nuestra clase, como por ejemplo un proyector que muestre contenidos que se vieron en clase o un pupitre en el que sentarse para rellenar un cuestionario que luego será evaluable.

El objetivo general de este proyecto de fin de grado es proponer una herramienta que tenga los puntos positivos de la gamificación en el aula y los mundos virtuales ya que actualmente no hay ninguna aplicación que abarque estos dos entornos.

1.2 Objetivos

El objetivo es presentar VirtualCraft, un sistema que para sus usuarios sea una interfaz atractiva, útil y sencilla de usar y que aúna las dos partes descritas anteriormente. Se compone de dos partes diferenciadas, una para usar durante las clases y otra para el acceso al mundo virtual en horario no lectivo. Se ha establecido un sistema de roles y puntuaciones personalizados de modo que los avances realizados en la parte usada en clase tengan sus repercusiones (tanto buenas como malas) en el mundo virtual.

El ámbito de este trabajo de fin de grado se encuentra en el desarrollo de la herramienta usada en clase y del módulo de comunicación que se encarga de conectar ambas partes.

1.3 Organización de la memoria

La memoria consta de seis capítulos y tres anexos:

- **Capítulo 1: Introducción**

Explicación de la motivación y los objetivos del proyecto, además se expone una breve descripción de la organización del documento.

- **Capítulo 2: Estado del arte**

Descripción del estado de las tecnologías usadas en el proyecto actualmente, con sus ventajas e inconvenientes.

- **Capítulo 3: Diseño**

Exposición de los aspectos técnicos del proyecto explicando la conexión entre módulos.

- **Capítulo 4: Desarrollo**

Análisis de las herramientas usadas en el desarrollo del proyecto y descripción de su funcionamiento. Se incorporarán imágenes para detallarlo.

- **Capítulo 5: Integración pruebas y resultados**

Resumen de las pruebas realizadas y sus resultados

- **Capítulo 6: Conclusiones y trabajo futuro**

Conclusiones finales y estimaciones de mejoras futuras.

- **Anexo 1: Conexión de OpenSim con una base de datos externa**

2 Estado del arte

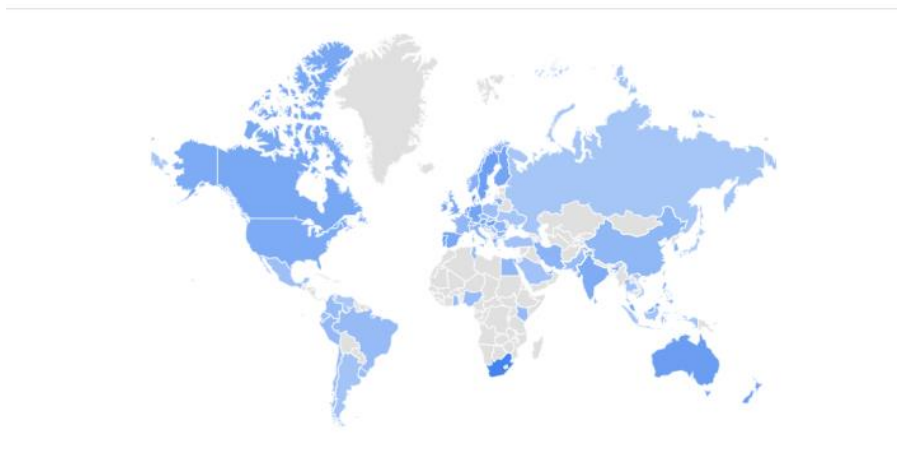
2.1 Gamificación

Gabe Zichermann, uno de los primeros gurús de la gamificación y coautor del libro *Gamification by Design* explicó: *la gamificación es el proceso de comprometer a la gente cambiando su comportamiento con mecánicas de un juego. Es aplicar lo divertido de los juegos y aplicarlo a situaciones que pueden no ser tan divertidas.* [3]

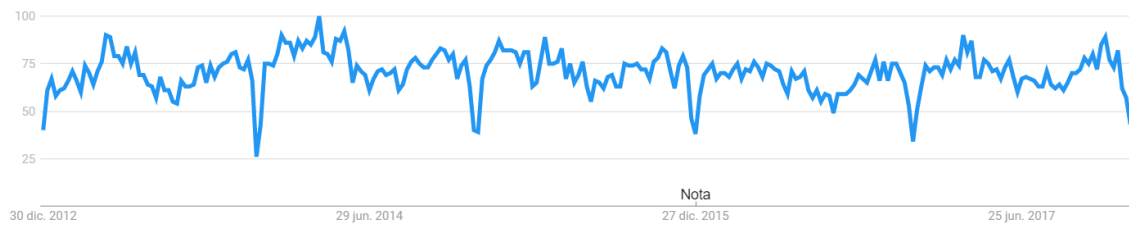
Aunque la gamificación está asociada principalmente a la tecnología, su esencia se basa en la psicología, Gabe Zichermann ha repetido en varias ocasiones que *la gamificación es un 75% psicología y un 25% tecnología*

La gamificación se basa en unos principios básicos que son recompensas, progreso y diversión. Estos tres factores son clave y se auto complementan entre sí para así tener un sistema que logre el compromiso de los usuarios y su motivación.

El crecimiento de la gamificación ha sido muy pronunciado estos últimos años, se espera que crezca \$5,500 billones para 2018 y que sobrepase los \$10 billones para 2020 [4] También es muy ilustrativo ver la cantidad de búsquedas que reciben el término “gamification” o la frase “what is gamification”. Esto queda reflejado a continuación en las figuras 2-1.1, 2-1.2 y 2-1.3.



**Figura 2-1.1: Mapa búsquedas de “gamification”,
<https://trends.google.com/trends/explore?q=gamification>**



**Figura 2-1.2: Gráfico búsquedas de “gamification”,
<https://trends.google.com/trends/explore?q=gamification>**



**Figura 2-1.3: Gráfico búsquedas de “what is gamification”,
<https://trends.google.com/trends/explore?q=what%20is%20gamification>**

2.1.1 Gamificación en la educación

La gamificación usa nuestro deseo por recompensas y competiciones para cambiar los hábitos de las personas. La gamificación en la educación pretende aplicar aspectos presentes en los videojuegos para lograr la atención de los estudiantes, pero no pretende ser un juego de aprendizaje en sí. Gracias al esfuerzo del estudiante para progresar en el juego ganará puntos y subirá de nivel, pero estos puntos se ganarán si el estudiante realiza sus tareas y estudios necesarios.

Ananth Pai era un profesor de Minnesota que reemplazó el plan de estudios estándar con un plan de estudios basados en videojuegos usando unas Nintendo DS y videojuegos para aprender lengua y matemáticas. En un periodo de 18 semanas pasó de un nivel de tercer grado a un nivel de 4.5 en matemáticas y lengua, además preguntando a los estudiantes sobre su experiencia en las clases dijeron que *aprender era divertido* y que *aprender se hacía en grupo* [5]

De acuerdo al estudio de Traci Sitzmann los juegos se convierten en una gran herramienta de enseñanza por diferentes motivos: [6]

- **Engancha**
Aumenta la retención de los contenidos aprendidos en un 90%
- **Genera confianza**

Al ser un videojuego los alumnos suelen estar acostumbrados a este tipo de plataformas y la confianza en uno mismo aumenta en un 20% en comparación a usar otros métodos.

- **Aumenta el rendimiento**

Los videojuegos motivan al estudiante, aumentando su rendimiento hasta en un 20%

- **Basados en la práctica**

El estudiante aprende gracias a la práctica constante.

Además, cabe destacar la cantidad de búsquedas que recibe “gamification education” que queda reflejado en el gráfico mostrado en la figura 2-1.4



**Figura 2-1.4: Gráfico búsquedas de “gamification education”,
<https://trends.google.com/trends/explore?q=gamification%20education>**

Por último, es importante mencionar un sistema de gamificación en educación llamado Classcraft, que se diferencia de otros sistemas de gamificación ya que está pensado para que los alumnos lo usen diariamente durante las horas lectivas, mientras que los otros están pensados para que los alumnos los usen en su tiempo libre. Además, es importante su mención ya que es un claro ejemplo en el que se han obtenido resultados que demuestran no sólo que la gamificación ayuda al rendimiento académico de los alumnos, sino a la relación entre ellos llegando a reducir el bullying [7].

2.2 Mundos virtuales en la educación

Los mundos virtuales son herramientas colaborativas, muchos estudios corroboran que el aprendizaje y el trabajo colaborativos son más eficientes. Ackermann dijo en su estudio que la interactividad es la clave para el aprendizaje y que un gran número de diseñadores de software y educadores concuerdan en que la experiencia se obtiene con la interactividad y, en definitiva, la experiencia es conocimiento [8]. Las experiencias de los mundos virtuales son muy beneficiosas para los alumnos ya que aprenden cosas de primera mano y además pueden realizarse experiencias que en la vida real podrían llegar a ser peligrosas o muy costosas. A continuación, se expone un ejemplo de herramienta de creación de mundos virtuales, OpenSim, que ha sido usada en numerosas ocasiones en entornos académicos.

2.2.1 OpenSim

OpenSim surge de la contracción de Open Simulator, es un servidor 3D de código abierto que permite crear ambientes virtuales. Para ejecutar correctamente los visores se debe disponer, como mínimo, 1 GB de RAM y una tarjeta gráfica con 256 MB (DDR2 o DDR3) de memoria de vídeo dedicada. Puede descargarse en <http://opensimulator.org/wiki/Download>

OpenSim ha sido usado en entornos académicos, para su uso se creó la herramienta Sloodle, un proyecto de código abierto que une la enseñanza en web junto con la interacción del mundo virtual. Gracias a esta unión entre las dos plataformas su interacción resulta muy sencilla. Al subir contenido a Sloodle, por ejemplo, una presentación en Power Point, podríamos tener un proyector en OpenSim que cumpliera esa funcionalidad y poder ver el contenido desde el mundo virtual junto a todos los alumnos. La imagen (figura 2-2.5), mostrada a continuación es un ejemplo de ello.



Figura 2-3.5: Presentación de Sloodle en OpenSim, <https://elearningindustry.com/who-moved-my-sloodle>

Sin embargo, se descartó usar Sloodle como interfaz web para este proyecto por la poca personalización que ofrece Moodle y por usar características que no son necesarias en el momento de la implantación.

2.3 Conclusiones

Tal y cómo se ha explicado en los puntos 2.1 y 2.2 la gamificación y los mundos virtuales son herramientas muy útiles y logran resultados muy positivos en el ámbito educativo. Sin embargo, no se encuentran sistemas que aprovechen estos dos elementos, es decir, un sistema basado en gamificación y que a la vez esté conectado a un mundo virtual que los alumnos puedan usar. Este es el punto que abarca este trabajo de fin de grado, proponiendo la unión de ambas herramientas para conseguir un sistema que tenga los beneficios de ambas partes.

3 Requerimientos y Diseño

3.1 Requerimientos

Los requerimientos de la aplicación se obtuvieron en varias reuniones, presenciales y online con la usuaria María José Lasala que va a ser una de las profesoras que se encargue de implantar el sistema en su instituto. La aplicación está formada por dos partes diferenciadas: una herramienta basada en gamificación para usarla durante las clases y un mundo virtual que los alumnos usarán después de las clases. Para la unión de ambas partes se creó un módulo de comunicación. El ámbito de este trabajo de fin de grado abarca el desarrollo de la página web y el módulo de comunicación; por otro lado la parte del mundo virtual ha sido desarrollada por Andrea Díez Barroso en su trabajo llamado *Diseño e implementación de un sistema educativo basado en mundos virtuales*. A continuación se detallan los requisitos de todo el sistema:

- La herramienta usada en clase ha de ser usable por dispositivos móviles.
- Todos los alumnos que accedan a usar la aplicación deben disponer de un ordenador portátil o de sobremesa para poder acceder al mundo virtual, si no la experiencia de la aplicación no sería completa.
- Los alumnos estarán organizados en grupos de 4 o 5 personas. Está pensado que no hayan más de 7 grupos en una misma clase.
- El profesor se encarga de crear los grupos y de asignar los roles de los usuarios.
- La aplicación define tres tipos de roles para los alumnos, que son, abad, caballero y conde. Los tres roles deben estar equilibrados dentro de un grupo. Cada uno de los roles tiene 6 niveles.
- La aplicación define 5 tipos de puntuaciones que los alumnos podrán adquirir o perder:
 - Puntos de vida (PV), en caso de llegar a 0 entrarían a la cárcel y sólo el profesor puede sacarlos de ahí. Si un alumno está en la cárcel no podrá acceder al mundo virtual.
 - Puntos de oficio (PO).
 - Puntos de impuestos (PD).
 - Puntos de superación (PP).
 - Puntos de oro (FO).
- Todos los alumnos son creados con 20 puntos de oro inicialmente, con estos puntos pueden comprar objetos. Con el uso de la parte web pueden obtener más puntos de oro, al subir de nivel desbloquearan objetos más llamativos.
- Cada grupo posee una parcela circular dentro del mundo virtual en la que podrán situar los objetos que compren. Cuando todos los alumnos de un grupo suban de nivel la parcela crecerá en tamaño. Ningún alumno puede colocar objetos en una parcela que no sea de su grupo.
- En el mundo virtual existen tres tiendas en las que los alumnos pueden comprar objetos, una para cada tipo de avatar. En cada una de las tiendas se vigilará si el alumno tiene el nivel y los puntos de oro necesarios y si el tipo de avatar es correcto antes de realizar alguna compra.

3.1.1 Sistema usado en el aula

Se presenta un sistema pensado para la existencia de varios institutos, con varios profesores trabajando, cada uno de ellos con varias clases con alumnos. La aplicación viene preparada con una serie de parámetros preconfigurados: puntuaciones iniciales de los avatares, acciones para premiar o penalizar el comportamiento de los alumnos, habilidades para los alumnos o condenas en caso de que un alumno se quede sin puntos. Esto hace que esté lista para usarse, pero el profesor puede cambiarlos a gusto propio.

Se definen tres tipos de avatar, condes, caballeros y abades, pensados para distintos tipos de alumnos, de modo que las características de cada rol sean mejor adaptables a un alumno inquieto, trabajador o inteligente. Cada uno de los roles tiene 6 niveles, todos ellos parten con unos puntos iniciales y podrán adquirir más o perderlos en el transcurso de las clases.

En función del nivel, tipo de avatar y sus puntos los alumnos podrán desbloquear y usar habilidades (también llamadas privilegios). Las habilidades son poderes que podrán comprar y usar si tienen puntos durante su experiencia en el aula. Las habilidades son distintas para cada tipo de avatar y ofrecen recompensas más atractivas según sube el nivel. Ejemplos de habilidades podrían ser 5 minutos más en el examen para el alumno que compra la habilidad o que el grupo del alumno que compra la habilidad entregue el próximo trabajo un día más tarde.

En relación con los niveles y habilidades, se establecen cinco tipos de puntos:

- Puntos de vida (PV) que muestran la evolución en clase, en caso de llegar a 0 entrarían a la cárcel y sólo el profesor puede sacarlos de ahí.
- Puntos de oficio (PO) que permiten comprar habilidades.
- Puntos de impuestos (PD) que permiten usar las habilidades compradas.
- Puntos de superación (PP) que permiten subir de nivel.
- Puntos de oro(FO) que permiten comprar objetos en el mundo virtual.

El sistema tiene dos tipos de usuarios que se definen a continuación:

3.1.1.1 Profesores

Existen tres tipos de profesores en cuanto a los permisos que tienen dentro de la aplicación.

3.1.1.1.1 Profesor sin permisos

Un profesor sin permisos especiales puede:

- Crear varias clases sin límite de alumnos.
- Borrar clases, esto implica borrar los alumnos pertenecientes a esa clase.
- Añadir alumnos a una clase ya creada.
- Borrar alumnos de una clase.
- Crear un nuevo grupo para una clase.
- Añadir y quitar alumnos de un grupo.
- Dentro de una clase un profesor puede:

- Dar o quitar puntos gracias a las acciones definidas en el sistema o con las acciones creadas por él mismo.
- Listar todos los alumnos de la clase viendo su tipo, puntuaciones, nivel y si están o no en la cárcel.
- Ver las habilidades ejercidas por un alumno.
- Publicar contenidos de interés para la clase.
- Usar las herramientas del profesor, que son:
 - Elegir un alumno al azar.
 - Elegir un grupo al azar.
 - Elegir dos alumnos al azar.
 - Usar el cronómetro.
 - Botón configurable para que redirija a una web que el profesor quiera.
- Modificar las imágenes de escudo y fondo de grupo las veces que desee.
- Listar los profesores dentro de su mismo instituto viendo sus permisos.
- Listar los alumnos dentro de sus clases viendo sus puntuaciones, grupo y contraseña.
- Elegir su imagen de avatar.
- Cambiar contraseñas de alumnos o la suya propia.
- Configurar el sistema, esto implica:
 - Añadir nuevas condenas, éstas estarán presentes únicamente en sus clases.
 - Listar todas las condenas que tiene activadas.
 - Habilitar y deshabilitar las condenas que vienen por defecto en el sistema.
 - Borrar las condenas introducidas por el mismo.
 - Añadir nuevas acciones, éstas estarán presentes únicamente en sus clases.
 - Listar todas las acciones que tiene activadas.
 - Habilitar y deshabilitar las acciones que vienen por defecto en el sistema.
 - Borrar las acciones introducidas por el mismo.
 - Añadir nuevas puntuaciones iniciales para los diferentes tipos de avatar, éstas estarán presentes únicamente en sus clases.
 - Listar todas las puntuaciones que tiene activadas.
 - Habilitar y deshabilitar las puntuaciones que vienen por defecto en el sistema.
 - Borrar las puntuaciones introducidas por el mismo.
 - Crear nuevas habilidades.
 - Listar las habilidades activas.
 - Añadir imágenes al Sistema. Pueden ser de avatares, escudos, fondos de grupo, habilidades y profesores.
 - Borrar imágenes.
 - Cambiar los parámetros de actualizaciones diarias que tiene la aplicación.
 - Cambiar las direcciones definidas por los botones de herramientas.

3.1.1.1.2 Admin

Un profesor con permisos de Admin puede:

- Realizar todas las acciones descritas en un profesor sin permisos.
- Añadir un nuevo profesor a su instituto, con permisos de admin o no.

3.1.1.1.3 SuperAdmin

Un profesor con permisos de SuperAdmin puede:

- Listar todos los profesores existentes en la aplicación, agrupados por institutos, viendo sus permisos, correo y contraseñas.
- Listar todos los alumnos agrupados por clases, mostrando sus puntos, grupo, correo y contraseña.
- Listar todos los institutos junto con el profesor de contacto asociado al instituto.
- Añadir un nuevo instituto asignándole un profesor de contacto, éste profesor creado puede tener permisos de Admin o SuperAdmin o directamente no tener permisos.
- Cambiar contraseña de un profesor o un alumno.
- Cambiar su imagen de avatar.

3.1.1.2 Alumnos

Un alumno puede:

- Ver sus puntuaciones, nombre de equipo, nombre de la clase, escudo, fondo y compañeros de grupo en pantalla.
- Ver los contenidos publicados por el profesor y responderlos.
- Ver los grupos existentes dentro de la clase, con opción a ver sus integrantes, sus puntuaciones, el tipo de avatar y si están o no en la cárcel.
- Acceder a la pantalla de habilidades, que se verán en color las que están disponibles, opacas las que no están compradas pero sí disponibles y en blanco y negro las que no están disponibles por falta de nivel o porque no están desbloqueadas las habilidades necesarias. Se encuentra una descripción más detallada en el anexo C.
- En caso de que el alumno esté en la cárcel, sus puntuaciones no estarán visibles y los enlaces a habilidades, vista de grupos y contenidos estarán deshabilitados.
- Comprar objetos en el mundo virtual con los puntos de oro obtenidos en el sistema usado en el aula.

3.2 Diseño

A continuación se muestra un esquema general de todo el sistema en la figura 3.2-1.

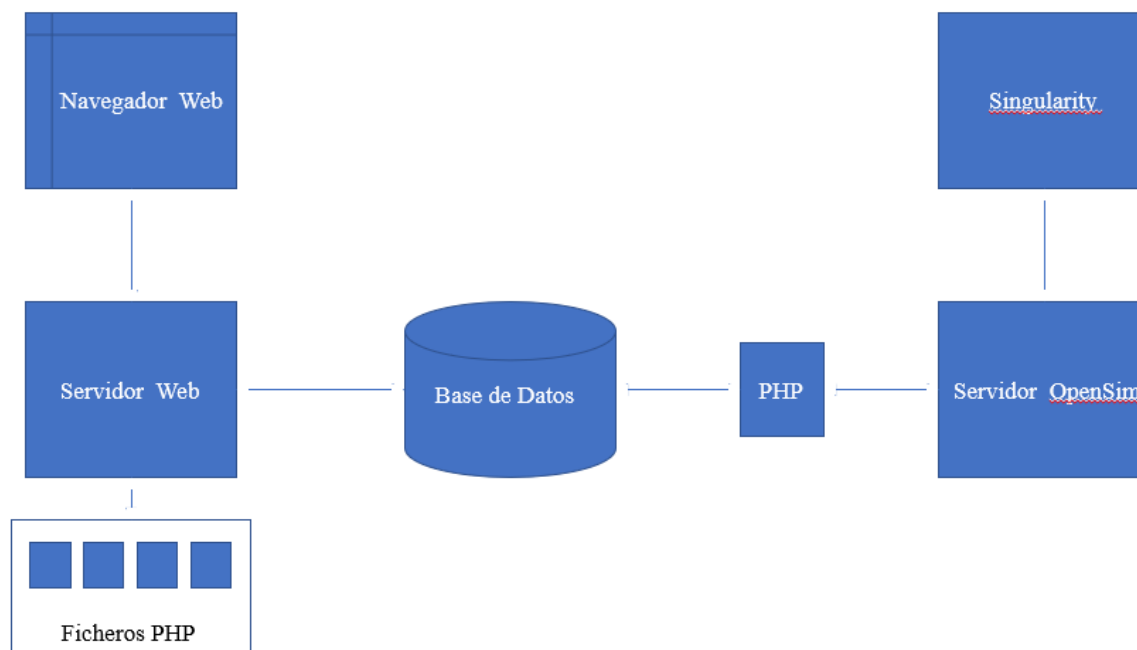


Figura 3-2.1: Esquema del sistema general

El sistema usado en el aula se implementó como una página web, de esta forma nos asegurábamos de que el sistema se vería correctamente independientemente del sistema operativo del dispositivo con el que se visualizase. Además, así no es necesario pedirle al usuario instalar ninguna aplicación, únicamente es necesario acceso a Internet.

Así en la figura 3-2.1 vemos cómo a través de un navegador web accedemos a la Web, que está alojada en nuestro servidor y se realizan peticiones a nuestra base de datos. Por otro lado, la conexión al servidor de OpenSim se realiza a través de la aplicación Singularity (enlace de descarga en <http://www.singularityviewer.org/downloads>). Finalmente, el módulo de comunicación entre la parte usada en clase y el mundo virtual es ese fichero PHP que está entre la base de datos y el servidor de OpenSim, el funcionamiento de esta parte viene más detallado en el punto 3.2.3

3.2.1 Base de Datos

A continuación, se presenta un esquema con las tablas de la base de datos que se diseñaron para el funcionamiento de la aplicación.

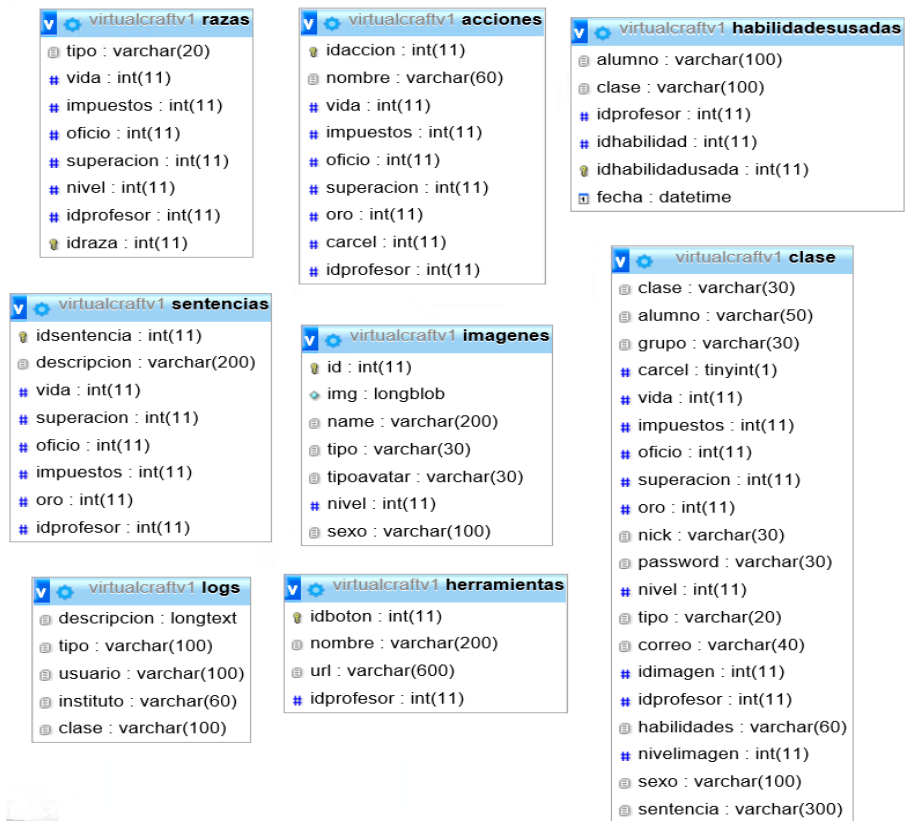


Figura 3-2.3a: Esquema de las tablas de la base de datos (1)



Figura 3-2.3b: Esquema de las tablas de la base de datos (2)

Una explicación de las tablas mostradas en las figuras 3-2.3a y 3-2.3b se muestra a continuación:

- Tabla logs: tabla que sirve para registrar todo lo que ocurre en la aplicación, desde la creación de una clase a la habilidad usada por un alumno.
- Tabla sentencias: tabla que recoge las sentencias incluidas por defecto en la aplicación y las añadidas por cada profesor.
- Tabla razas: tabla que recoge las puntuaciones de los distintos tipos de avatar para los distintos niveles.
- Tabla acciones: tabla que guarda las acciones incluidas por defecto en la aplicación y las añadidas por cada profesor.
- Tabla imágenes que contiene todas las imágenes con las que trabaja la aplicación.
- Tabla herramientas que almacena la configuración dada por cada profesor para el botón de herramientas.
- Tabla habilidadesusadas que guarda las habilidades usadas con clase, fecha y hora por cada alumno.
- Tabla clase que guarda la información de los alumnos como sus puntuaciones, sus habilidades, si está o no en la cárcel, la clase y el profesor de la clase
- Tabla profesores que almacena toda la información relacionada con los profesores.
- Tabla institutos que contiene todos los institutos.
- Tabla contenidos en la que se recogen todas las tareas dadas por los profesores para los alumnos.
- Tabla habilidades que guarda las habilidades por defecto de la aplicación.
- Tabla habilidadesmod que almacena las habilidades modificadas por cada profesor.
- Tabla grupos que guarda los grupos creados por cada profesor.

3.2.2 Sistema usado en el aula

En las figuras 3-2.4, 3-2.5 y 3-2.6 se muestra el diseño del sistema usado en el aula, a través del login accederemos cómo alumno o profesor. En la figura 3-2.4 vemos las principales funcionalidades de un alumno y en las figuras 3-2.5 y 3-2.6 veremos las funcionalidades del profesor.

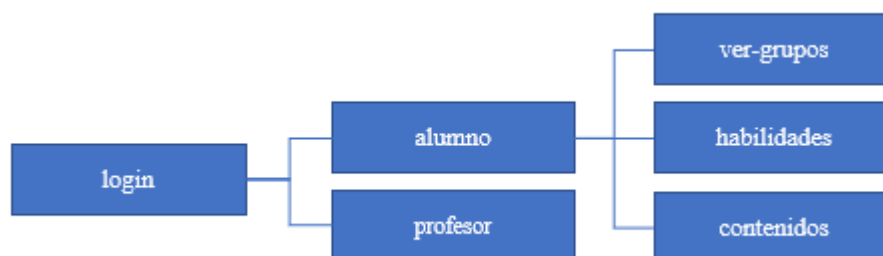


Figura 3-2.4: Esquema del sistema usado en el aula (1)

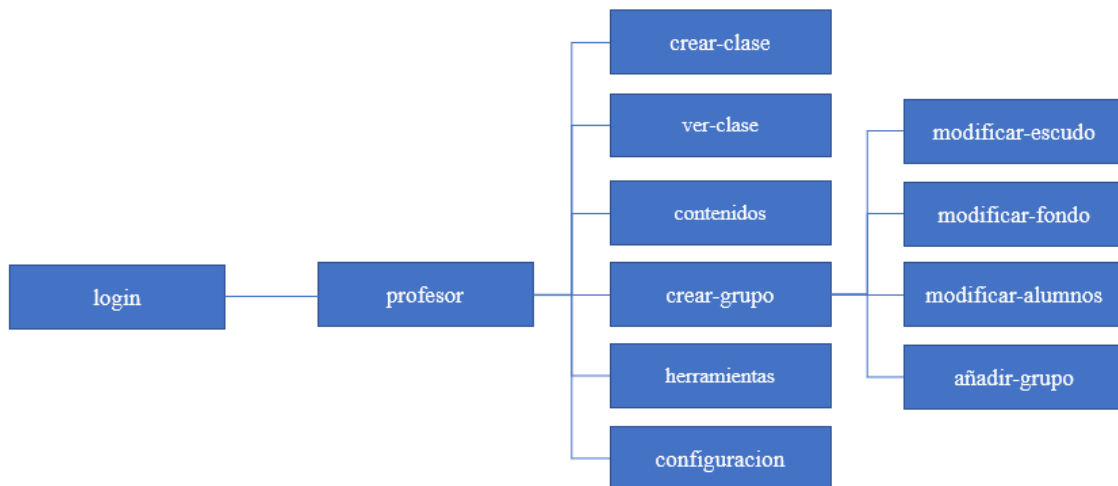


Figura 3-2.5: Esquema del sistema usado en el aula (2)

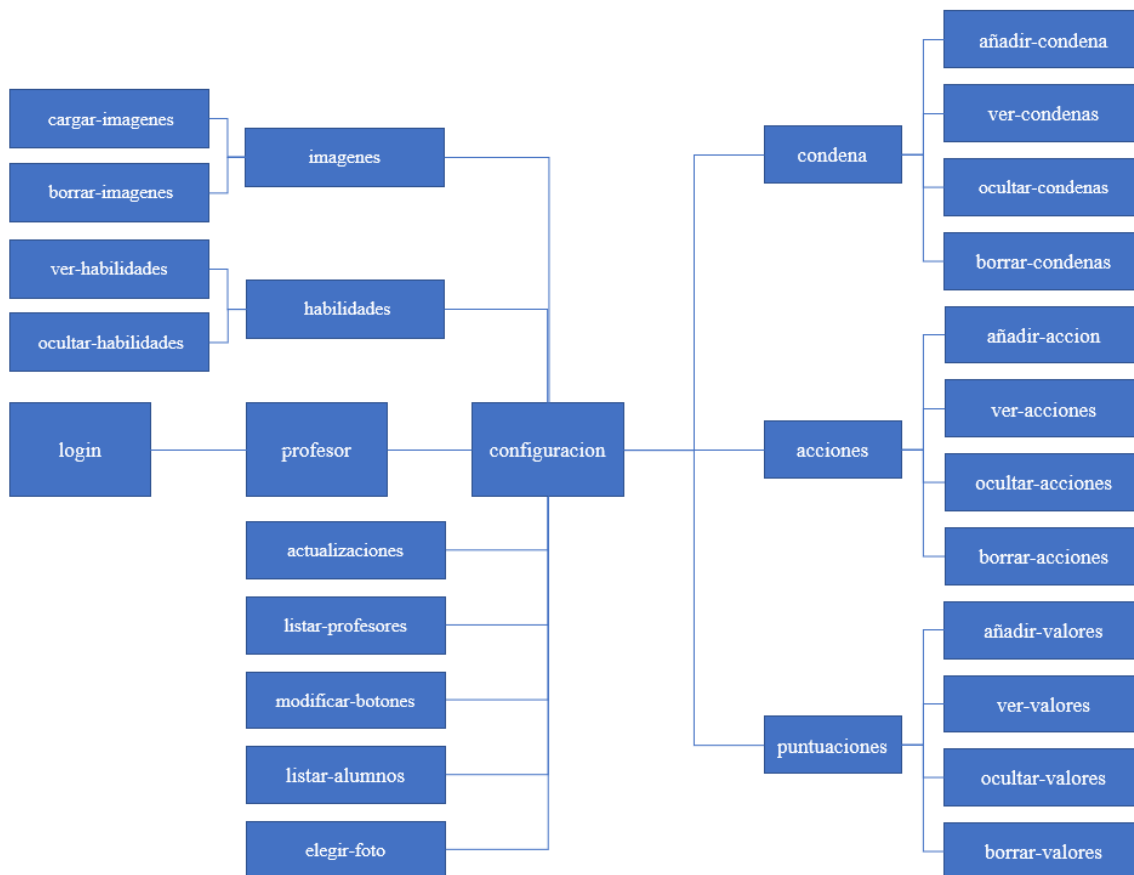


Figura 3-2.6: Esquema del sistema usado en el aula (3)

3.2.3 Módulo de comunicación

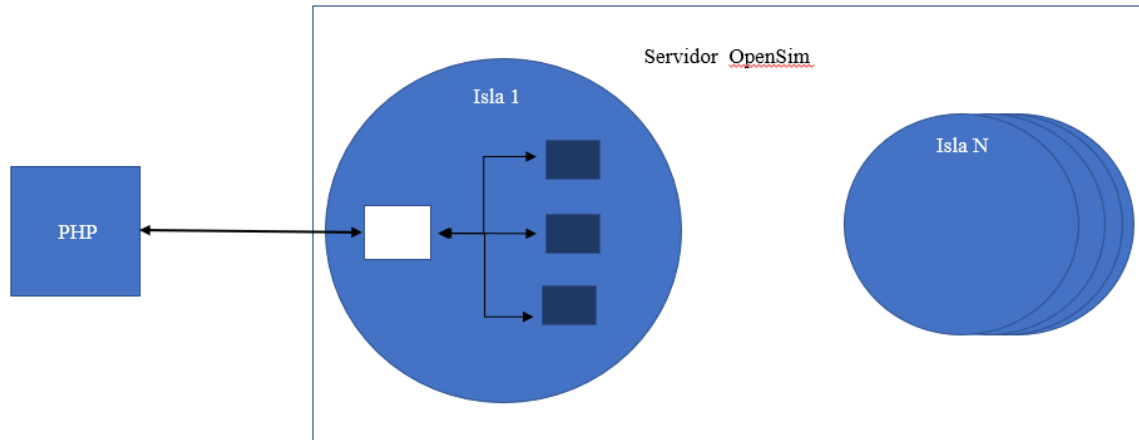


Figura 3-2.7: Esquema del sistema general

Tal y cómo se muestra en la figura 3-2.7 en cada isla de OpenSim tenemos objetos que están en comunicación por situaciones mencionadas anteriormente, tales cómo evitar que un alumno de un grupo ajeno coloque objetos en una parcela o gestionar las compras en las tiendas. Toda esta información es obtenida por un objeto global que se encarga de realizar la petición a la base de datos: tal y cómo se ve en la figura 3-2.7, la comunicación entre las tiendas y las parcelas pasa a otro objeto, y ese objeto se comunica con un PHP que a su vez se comunica con la base de datos.

Este trabajo de fin de grado abarca hasta la implementación de ese objeto comunicador entre las parcelas y las tiendas. La comunicación se realiza a través de una serie de comandos definidos, que obtienen la información necesaria de la base de datos y posteriormente dicha información se redirige al objeto que inició la petición. A continuación, se listan las funcionalidades implementadas en el protocolo de comandos:

- Grupo de un alumno.
- Grupo de varios alumnos.
- Obtener todas las puntuaciones de los alumnos.
- Obtener los puntos de oro.
- Aumentar los puntos de oro de un usuario.
- Obtener los puntos de vida.
- Obtener los puntos de impuestos.
- Obtener los puntos de oficio.
- Obtener los puntos de superación.
- Obtener el nivel.
- Obtener la clase de varios alumnos.
- Obtener el tipo de avatar.

- Saber si un alumno está en la cárcel o no.
- Obtener toda la información de un usuario.
- Obtener toda la información de los usuarios de un grupo.
- Obtener el nivel, oro, tipo de avatar y si está o no en la cárcel de un alumno.
- Obtener el profesor de un alumno.
- Obtener el nivel, oro, tipo de avatar y si está o no en la cárcel de un alumno. Y realizar una compra si se cumplen todas las condiciones necesarias.

4 Desarrollo

Los lenguajes principales usados en este proyecto han sido HTML, CSS, PHP, LSL y JavaScript. Como herramienta que aunaba los principales aspectos para el proyecto se ha usado WampServer en la versión 3.0.6 ya que ofrece en un único paquete de instalación PHP, Apache2 y MySQL, el enlace de descarga se encuentra en: <http://www.wampserver.com/en/>

El proyecto VirtualCraft viene definido por dos partes claramente diferenciadas, por un lado, existe la parte web desarrollada con PHP y JavaScript con su correspondiente base de datos (MySQL) a la que los alumnos accederán diariamente durante las clases para poder disfrutar de todas las funcionalidades que ofrece la aplicación. Por otra, parte existe un módulo de comunicación que sirve de intermediario entre la base de datos de la web y OpenSim.

4.1 Página web

Para el funcionamiento de la página web se implementaron un total de 99 ficheros. Se pueden clasificar en 11 ficheros para alumnos y 86 para profesores, cada uno de ellos con su fichero CSS correspondiente. Se muestran también los ficheros que se encargan del funcionamiento de un temporizador y la pantalla para que un alumno seleccione su imagen de avatar cada vez que suba de nivel. Estas dos últimas funcionalidades fueron descartadas en un segundo análisis de requisitos que tuvo lugar en diciembre, pero se optó por dejar el código en su directorio para que puedan usarse en caso de que se quiera volver a usar estas funcionalidades.

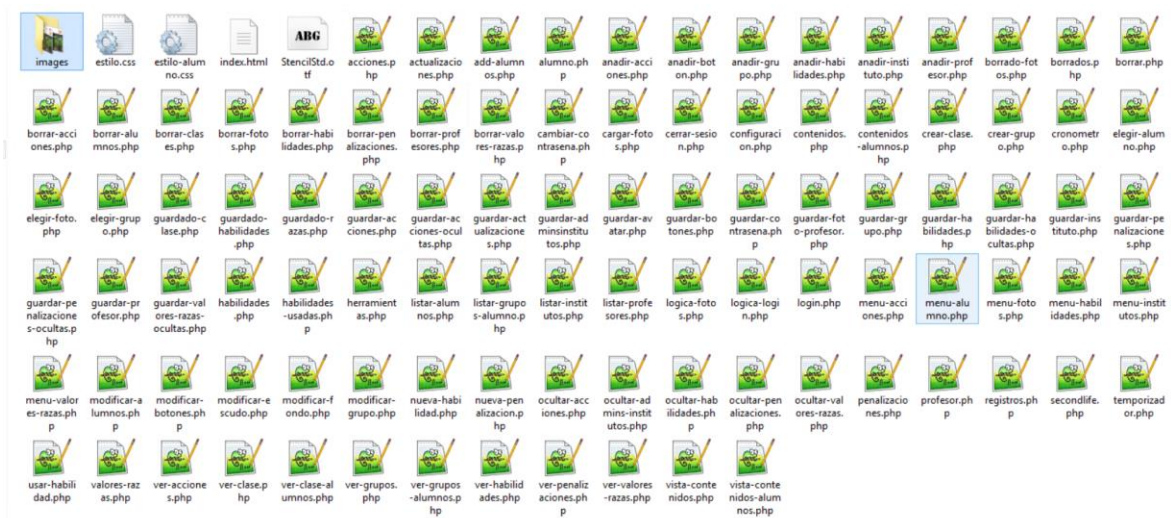


Figura 4-1.1: Ficheros desarrollados

Además, en la carpeta podemos ver un fichero de fuente llamada StencilSTD, en la versión de escritorio se muestra el texto perfectamente, pero a la hora de la visualización con dispositivos móviles no se veía la fuente con el aspecto adecuado, por ello fue necesario descargar el fichero y hacer la llamada como se muestra a continuación:

```

@font-face {
  font-family: MyStencil;
  src: url(StencilStd.ott);
}

.center3 {
  color:black;
  font-family: MyStencil;
  font-size: 46px;
  position: absolute;
  padding-left: 0%;
  top: 10%;
  width: 100%;
  text-align: center;
}

```

Figura 4-1.2: Uso de una fuente externa con CSS

El enlace de descarga es el siguiente: <http://www.fontpalace.com/font-download/StencilStd/>

En la carpeta de images se encuentran todas las imágenes usadas en el proyecto como pueden ser los fondos, escudos, herramientas del profesor y todas las imágenes para conseguir el aspecto mostrado en la parte de alumno.

Para completar toda la parte de desarrollo de la página web se deben mencionar un trigger que hace que el alumno entre automáticamente en la mazmorra si sus puntos de vida son menores o iguales a cero, y un evento que se encarga de las actualizaciones recurrentes cada 24 horas de las puntuaciones de los alumnos. A continuación se muestra el código de estas funcionalidades.

```

CREATE TRIGGER `triggercarcel` BEFORE UPDATE ON `clase`
FOR EACH ROW BEGIN
if new.vida <= 0 then
    set new.carcel = true;
END IF;
END

```

```

CREATE DEFINER=`root`@`localhost` EVENT `Actualizaciones_18` ON SCHEDULE
EVERY 24 HOUR STARTS '2018-01-09 22:59:17' ON COMPLETION NOT PRESERVE
ENABLE DO UPDATE clase SET vida = vida + 17, oficio = oficio + 18, superacion =
superacion + 19, impuestos = impuestos + 17, oro = oro + 18 WHERE idprofesor = 18

```

A continuaci3n se a~adan algunas im~agenes que muestran el aspecto visual de la parte web:



Figura 4-1.3: Pantalla Eines (herramientas) del profesor.



Figura 4-1.4: Pantalla principal de alumno

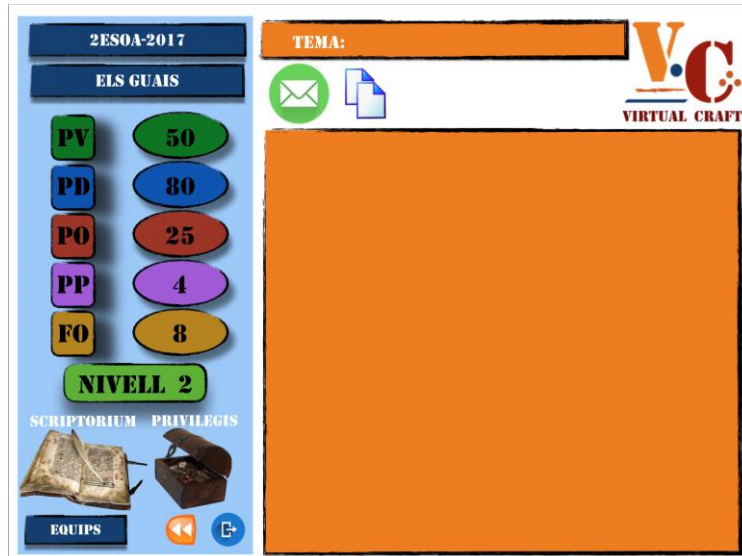


Figura 4-1.5: Pantalla Scriptorium



Figura 4-1.6: Pantalla de visor de grupos

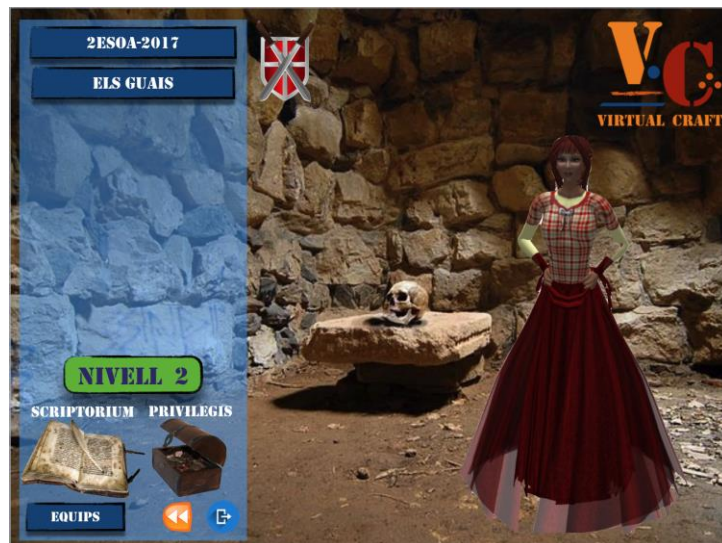


Figura 4-1.7: Pantalla principal de un alumno en la mazmorra

4.2 Módulo de comunicación

Para el desarrollo del objeto que establece la comunicación entre el módulo PHP y las tiendas y las parcelas de OpenSim se implementó un script en LSL que realiza peticiones HTTP al servidor. El fichero PHP se encarga de parsear los comandos y mandar una respuesta apropiada. En función del comando que reciba hará las consultas necesarias a la base de datos de la web y devolverá los resultados. El objeto que se encarga de la comunicación entre el fichero PHP y los objetos de OpenSim es llamado Tótem Máster.

La sintaxis general de los comandos y de las respuestas en notación Backus Normal Form (BNF) es:

<NombreComando> ::= GRP | GRPN | PNT | PNT_Oro | PNT_Vida | PNT_Oficio | PNT_Superacion | PNT_Impuestos | LVL | CLSN | ROL | JAIL | SHOP | ALLU | ALLG | TCH | UPD_Oro | VENTA

<NumEntidades> ::= [-] 1,2,...N

<ArgComando> ::= NombreAlumno | NombreGrupo

<Comando> ::= <NombreComando> <NumEntidades> <ArgComando>

<Dato> ::= vida | oficio | superacion | impuestos | oro | carcel | nivel | grupo | tipo

<DatosAlumno> ::= <Dato>:<NumEntidades> | <Dato> : <NumEntidades> <DatosAlumno>

<ListaDatosAlumno> ::= alumno:NombreAlumno <DatosAlumno> | alumno:NombreAlumno <DatosAlumno> <ListaDatosAlumno>

<Valor> ::= NombreAlumno | NombreGrupo | <NumEntidades> | <DatosAlumno> | ERR

<Respuesta> ::= <Comando> “;” <Valor>

A continuación se muestra una tabla que muestra el comando enviado por el Tótem Máster al fichero PHP y la respuesta que se envía al Tótem Máster una vez realizada la consulta. En concreto hay un comando para cada una de las acciones definidas en el apartado 3.2.3

Acción del comando	Ejemplos de comandos enviados	Ejemplos de respuestas recibidas
Grupo de un alumno.	GRP 2 MariaAngeles_Cano Alumno_2	GRP 1 MariaAngeles_Cano Alumno_2; Grupo1Prueba Grupo1Prueba
Alumnos de un grupo.	GRPN 1 Grupo1Prueba	GRPN 1 Grupo1Prueba; MariaAngeles Cano Alumno 2
Obtener todas las puntuaciones de los alumnos.	PNT 1 MariaAngeles_Cano	PNT 1 MariaAngeles_Cano ; vida:40-oficio:25-superacion:10-impuestos:40-oro:500
Obtener los puntos de oro.	PNT_Oro 2 MariaAngeles_Cano Alumno_2	PNT_Oro 2 MariaAngeles_Cano ; 500 500

Obtener los puntos de vida.	PNT_Vida 1 MariaAngeles_Cano	PNT_Vida 1 MariaAngeles_Cano ; 40
Obtener los puntos de oficio.	PNT_Oficio 1 MariaAngeles_Cano	PNT_Oficio 1 MariaAngeles_Cano ; 25
Obtener los puntos de superación.	PNT_Superacion 1 MariaAngeles_Cano	PNT_Superacion 1 MariaAngeles_Cano ; 10
Obtener los puntos de impuestos.	PNT_Impuestos 1 MariaAngeles_Cano	PNT_Impuestos 1 MariaAngeles_Cano ; 40
Obtener el nivel.	LVL 1 MariaAngeles_Cano	LVL 1 MariaAngeles_Cano ; 1
Obtener la clase de varios alumnos.	CLSN 1 MariaAngeles_Cano	CLSN 1 MariaAngeles_Cano ; ClasePrueba
Obtener el tipo de avatar.	ROL 1 MariaAngeles_Cano	ROL 1 MariaAngeles_Cano ; conde
Saber si un alumno está en la cárcel o no.	JAIL 1 MariaAngeles_Cano	JAIL 1 MariaAngeles_Cano ; 0
Obtener el nivel, oro, tipo de avatar y si está o no en la cárcel de un alumno.	SHOP 1 MariaAngeles_Cano	SHOP 1 MariaAngeles_Cano ; nivel:1-carcel:0-oro:500-tipo:conde
Obtener toda la información de un usuario.	ALLU 1 MariaAngeles_Cano	ALLU 1 MariaAngeles_Cano ; vida:40-oficio:25-superacion:10-impuestos:40-oro:500-carcel:0-nivel:1-tipo:conde-grupo:Grupo1Prueba
Obtener toda la información de los usuarios de un grupo.	ALLG 1 Grupo1Prueba	ALLG 1 Grupo1Prueba ; alumno: MariaAngeles_Cano-vida:40-oficio:25-superacion:10-impuestos:40-oro:500-carcel:0-nivel:1-tipo:conde alumno: Alumno_2- vida:40-oficio:25-superacion:10-impuestos:40-oro:500-carcel:0-nivel:1-tipo:conde
Obtener el profesor de un alumno.	TCH 1 MariaAngeles_Cano	TCH 1 MariaAngeles_Cano ; Xavier Alaman
Aumentar los puntos de oro de un usuario.	UPD_Oro 1 MariaAngeles_Cano 100	UPD_Oro 1 MariaAngeles_Cano 100 ; OK
Obtener el nivel, oro, tipo de avatar y si está o	VENTA 1 NombreAlumno caballero 1 5 objeto	VENTA 1 MariaAngeles_Cano caballero 1 5 ; OK MariaAngeles_Cano objeto

no en la cárcel de un alumno. Y realizar una compra si se cumplen todas las condiciones necesarias.		
---	--	--

Tabla 4-2: Ejemplos de comandos del Tótem Máster

Se sigue la misma estructura para todos los comandos, primero se envía el nombre del comando, luego un número que indica el número de alumnos a los que se le aplicará el comando y posteriormente el nombre del alumno (o del grupo dependiendo del comando usado). El nombre del alumno irá normalmente separado por espacios en blanco, pero antes de enviar el comando al Tótem Máster se le hace el tratamiento necesario para que de “MariaAngeles Cano” pase a “MariaAngeles_Cano”. Todos los casos documentados arriba son casos de prueba exitosos, en caso de fallo se sigue la misma estructura que en caso de éxito, devolviendo el comando seguido de “;” y ERR, por ejemplo:

ROL 1 MariaAngeles_Cano ; ERR.

A continuación se muestra el código que envía uno de los comandos mostrados en la tabla mostrada en la página anterior:

```

string nombreobjeto= "nombreobjeto";
default
{
    state_entry(){
        llListen(17345, "", "", "");
    }
    touch_start(integer number)
    {
        llRegionSay(17345, "GRP 1 MariaAngeles_Cano");
        llOwnerSay("Mensaje primer cubo: GRP 1 MariaAngeles_Cano");
    }

    listen(integer channel, string name, key id, string message)
    {
        //llOwnerSay("Canal recibido:"+channel);
        llOwnerSay("Mensaje recibido del cubo 2:"+message);
    }
}

```

Figura 5.1: Envío de un comando al módulo de comunicación

5 Integración, pruebas y resultados

Se han realizado dos rondas de pruebas a la parte web que vienen redactadas en otro documento. Tras una primera prueba en la que surgieron algunos problemas, realizando unas leves modificaciones el sistema se dio por estable y seguro, además se realizaron pruebas unitarias. Está pensada la implantación del sistema a lo largo de enero y/o febrero con la profesora María José Lasala. La profesora ha estado altamente vinculada en el desarrollo de la web indicando los elementos que encuentra más atractivos para la implantación de sistemas de gamificación en el aula ya que ella ya ha realizado experiencias con estos sistemas y sus resultados fueron excelentes.

A continuación, se muestra una captura en la figura 4-2.1 de la batería de pruebas para probar todos los comandos usados para la comunicación entre la base de datos y el módulo de comunicación. Para más información acerca de la implementación de este módulo ver el anexo B.

```
string nombreobjeto= "nombreobjeto";

default
{
    state_entry(){
        llListen(17345, "", "", "");
    }
    touch_start(integer number)
    {
        llRegionSay(17345, "GRP 1 MariaAngeles_Cano");
        llRegionSay(17345, "GRPN 1 Grupo1Prueba");
        llRegionSay(17345, "PNT 1 MariaAngeles_Cano");
        llRegionSay(17345, "PNT_Oro 1 MariaAngeles_Cano");
        llRegionSay(17345, "PNT_Vida 1 MariaAngeles_Cano");
        llRegionSay(17345, "PNT_Oficio 1 MariaAngeles_Cano");
        llRegionSay(17345, "PNT_Superacion 1 MariaAngeles_Cano");
        llRegionSay(17345, "PNT_Impuestos 1 MariaAngeles_Cano");
        llRegionSay(17345, "LVL 1 MariaAngeles_Cano");
        llRegionSay(17345, "CLSN 1 MariaAngeles_Cano");
        llRegionSay(17345, "ROL 1 MariaAngeles_Cano");
        llRegionSay(17345, "JAIL 1 MariaAngeles_Cano");
        llRegionSay(17345, "SHOP 1 MariaAngeles_Cano");
        llRegionSay(17345, "ALLU 1 MariaAngeles_Cano");
        llRegionSay(17345, "ALLG 1 Grupo1Prueba");
        llRegionSay(17345, "TCH 1 MariaAngeles_Cano");
        llRegionSay(17345, "UPD Oro 1 MariaAngeles_Cano 100");
        llRegionSay(17345, "VENTA 1 MariaAngeles_Cano conde 1 5 "+ nombreobjeto);
    }
    listen(integer channel, string name, key id, string message)
    {
        //llOwnerSay("Canal recibido:"+channel);
        llOwnerSay("Mensaje recibido del Totem Master:"+message);
    }
}
```

Figura 5.1: Batería de pruebas del Tótem Máster

6 Conclusiones y trabajo futuro

6.1 Conclusiones

El propósito de este Trabajo de Fin de Grado era crear un sistema de gamificación para el aula que fuera atractivo para los alumnos y fácil de usar para los profesores. El sistema se desarrolló en formato Web para asegurar la compatibilidad con cualquier dispositivo y además evitar al usuario tener que realizar alguna instalación en sus dispositivos.

Se han cumplido todos los requisitos que se definieron en un primer momento y se han añadido funcionalidades que en la revisión del primer sistema estable se vieron necesarias. La implantación del sistema está a la espera de terminar con el diseño de algunas de las imágenes de la web, cómo por ejemplo las de las habilidades. Tras enseñar la maquetación de las ventanas a los alumnos se recibieron respuestas positivas y están deseando empezar a usarlo.

El desarrollo de este trabajo de fin de grado ha ayudado a reforzar todos los conocimientos adquiridos de PHP y CSS durante la carrera, así cómo aprender LSL y buscar y ajustar los parámetros de la configuración del servidor para tener el comportamiento deseado.

6.2 Trabajo futuro

El sistema está pensado para que su crecimiento no se vea limitado. Ya está pensado para que trabajen varios profesores de distintos institutos a la vez y la aplicación está provista de elementos colocados por defecto para que el profesor pueda usarla sin tener que hacer modificaciones.

Cómo posible mejora futura está pensado que haya un sistema de diferenciación del idioma y redirigir al usuario a la parte web en castellano o catalán según lo desee. Actualmente la aplicación dirige a la web en catalán ya que era un requisito fundamental el que la Web estuviese en catalán para poder implantarse. Además, para realizar esta diferenciación por idioma serían necesarias nuevas imágenes con los textos en castellano.

Además, otra mejora pensada en relación al Tótem Máster es la implementación de un comando en el que se envíe y guarde en la base de datos todo lo que los alumnos escriban en el mundo virtual.

Referencias

- [1] Speak up, Project Tomorrow for Speak Up, 2016, http://www.tomorrow.org/speakup/pdfs/Speak%20Up%20May%202015_PR1.pdf
- [2] CASE STUDY MSD: A Game-Based Learning Method, 2012, <https://www.game-learn.com/gamification-example-successful-story-merck/>
- [3] Gabe Zichermann, Christopher Cunningham, Gamification by Design, 2011, https://books.google.es/books/about/Gamification_by_Design.html?id=Hw9X1miVMMwC&redir_esc=y
- [4] Global Knowledge, <https://www.globalknowledge.com/ca-en/content/articles/the-growth-of-gamification/>
- [5] Ananth Pai, <https://ed.ted.com/on/djRl8dab>
- [6] Traci Sitzmann, A META-ANALYTIC EXAMINATION OF THE INSTRUCTIONAL EFFECTIVENESS OF COMPUTER-BASED SIMULATION GAMES, 2011, <http://onlinelibrary.wiley.com/doi/10.1111/j.1744-6570.2011.01190.x/abstract>
- [7] <https://www.businesswire.com/news/home/20171002005164/en/Classroom-Study-Reveals-New-Strategies-Reduce-Bullying>
- [8] Ackermann, E. (1994). Direct and Mediated Experience: Their Role in Learning. In LEWIS, R. et MENDELSON, P., (eds.), *Lessons from Learning*. North-Holland, Amsterdam.

Glosario

API	Application Programming Interface
CSS	Cascading Stylesheets
PHP	Hypertext Preprocessor
LSL	Linden Scripting Language
HTTP	Hypertext Transfer Protocol
BNF	Backus Normal Form

Anexos

A Configuración de Apache y PHP para el funcionamiento de la web.

En primer lugar, es posible que no nos lleguen las conexiones al servidor, pero accediendo desde el ordenador en el que se ejecuta el servidor accediendo con localhost o 127.0.0.1 vemos que el servidor está ejecutándose, esto puede deberse a que el cortafuegos no permite que las conexiones entren por el puerto que tiene definido Apache por defecto que es el 80. Para cambiarlo el necesario abrir el fichero de configuración llamado httpd.conf.

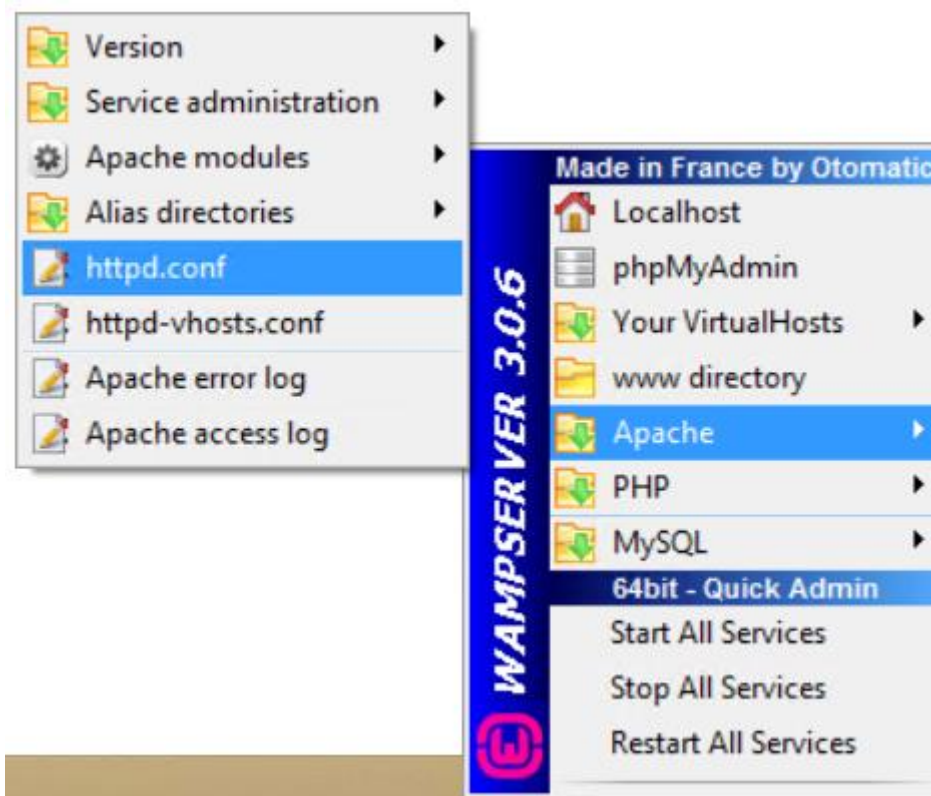


Figura 0.1.1: Localizar fichero httpd.conf

Una vez abierto el fichero observamos que contiene las siguientes líneas

```
#  
# Change this to Listen on specific IP addresses as shown below to  
# prevent Apache from glomming onto all bound IP addresses.  
#  
#Listen 12.34.56.78:80  
Listen 0.0.0.0:80  
Listen [::0]:80
```

Para añadirle la opción de aceptar tráfico por otro puerto, por ejemplo, el 8000, es necesario añadir las siguientes líneas:


```
Listen 0.0.0.0:8000
```

```
Listen [::0]:8000
```

También puede darse el caso que desde dispositivos dentro de una misma red se permita la conexión, pero con dispositivos fuera de esa red las conexiones se vean restringidas. Nuevamente volveremos a tocar el fichero httpd.conf y localizaremos las siguientes líneas:

```
#  
# DocumentRoot: The directory out of which you will serve your  
# documents. By default, all requests are taken from this directory, but  
# symbolic links and aliases may be used to point to other locations.  
#  
DocumentRoot "${INSTALL_DIR}/www"  
<Directory "${INSTALL_DIR}/www/">  
    Options +Indexes +FollowSymLinks +Multiviews  
    AllowOverride all  
    Require local  
</Directory>
```

Para permitir que cualquier dispositivo pueda acceder a nuestro servidor desde cualquier red es necesario cambiar la línea de *Require local* por *Require all granted*

Por último, en la página web se permite que los profesores suban fotos para la aplicación para escudos, fondos, avatares y profesores. En una de las subidas de fotos se encontró un error inesperado ya que de 20 fotos que habían sido seleccionadas para subirlas a la aplicación sólo permitió subir 18, con las otras dos la web lanzó una excepción diciendo que el tamaño de imagen era 0, sin embargo, el tamaño de imagen indicaba 2.3MB.

Tras una breve investigación se llegó a la conclusión de que el problema venía en los parámetros de configuración de PHP ya que se establece un tamaño máximo de archivos de subida, para cambiar esto debemos cambiar el archivo php.ini:

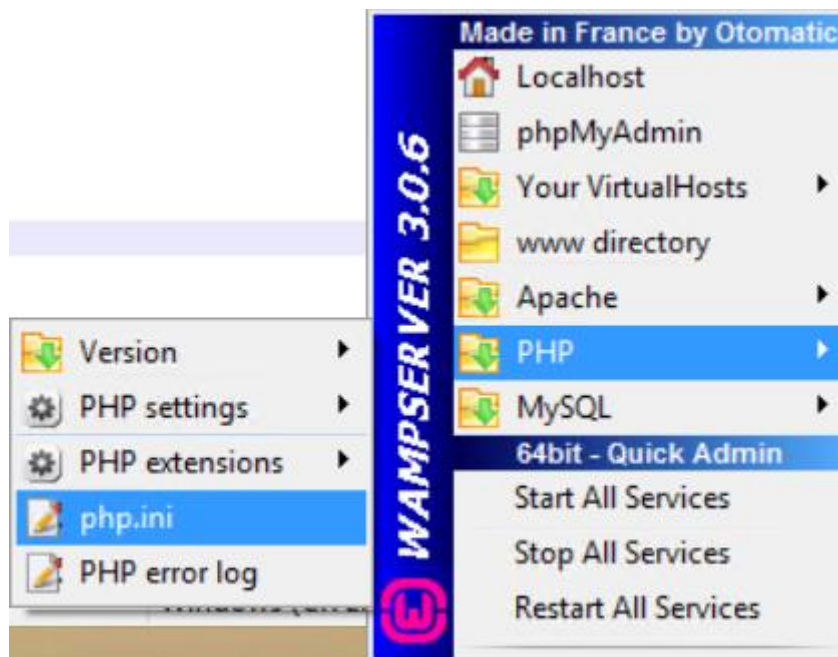


Figura 0.1.2: Localizar fichero php.ini

Es necesario localizar estas líneas:

```
; Maximum allowed size for uploaded files.
; http://php.net/upload-max-filesize
upload_max_filesize = 20M
```

```
; Maximum size of POST data that PHP will accept.
; Its value may be 0 to disable the limit. It is ignored if POST data reading
; is disabled through enable_post_data_reading.
; http://php.net/post-max-size
post_max_size = 20M
```

Simplemente es necesario cambiar los valores de las variables `upload_max_filesize` y `post_max_size` por los valores que necesitemos, teniendo en cuenta que `post_max_size` debe ser mayor o igual que `upload_max_filesize`.

Por último, se debe recordar que para que todos estos cambios se vean reflejados es necesario parar todos los servicios y volver a arrancarlos.

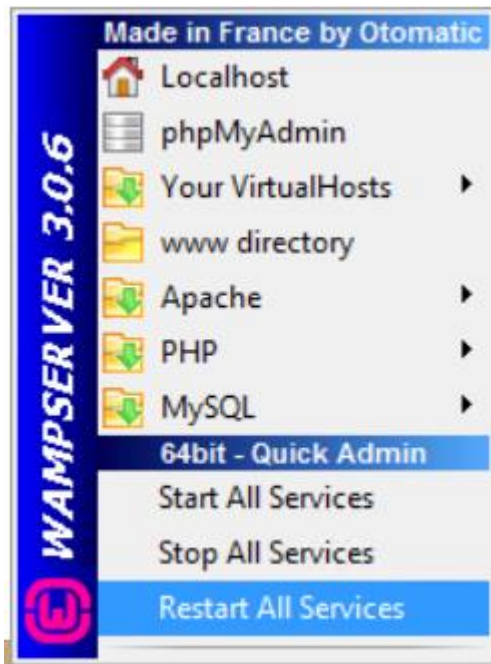


Figura 0.1.3: Cómo reiniciar el servidor

B Conexión de OpenSim con una base de datos externa. Módulo de comunicación.

El método más sencillo para poder acceder a la información de nuestra base de datos desde el mundo virtual es mediante conexiones HTTP de modo que un fichero PHP acceda a la base de datos y nos devuelva los resultados que necesitamos. A continuación, se muestra un ejemplo paso a paso de la configuración del proceso.

En primer lugar, nos crearemos un objeto en OpenSim y le introduciremos el siguiente script:

```
key clave;
default
{
  state_entry(){} touch_start(integer number)
  {
    clave=llHTTPRequest("http://xxxx.xxxx.xxxx.xxxx/directorio/fichero.php?lista=mensaje",
    [HTTP_METHOD,"GET"],"");
  }
  http_response(key id, integer status, list metaData, string body)
  {
    if(id==clave){ llOwnerSay("Body :'+ body);
    }
  }
}
}
```

Además en nuestro directorio de apache debemos tener el fichero PHP que acceda a nuestra base de datos con algo de este estilo:

```
<?php
if(isset($_GET['lista'])){

  $db=new PDO("mysql:dbname=nombre; host=localhost","usr","");
  $stmt = $db->prepare('consulta');
  $stmt->execute();
  $result=$stmt->fetchAll();
  foreach($result as $arrays3) {
    $aux2.=$arrays3["grupo"]." ";
  }
  $ret=$_POST['lista']." ;'.$aux2;
  echo $ret;
}
?>
```

Es importante tener el método `http_response` en el script de OpenSim ya que imprimiendo el body devuelto de la conexión podremos ver si ésta se ha realizado con éxito. En caso de recibir algún error y sabemos que no hay error en el fichero de PHP se recomienda pegar la URL que estamos mandando en algún navegador (“`http://xxxx.xxxx.xxxx.xxxx/directorio/fichero.php?lista=mensaje`”) y observar si realiza el funcionamiento correctamente.

Si el error no está en la dirección es probable que nuestros directorios no tengan permiso de acceso o edición por la aplicación de OpenSim. Para cambiar los permisos de las carpetas desde Windows nos vamos al directorio principal que contiene todos los ficheros que tenemos compartidos en apache, pulsamos con el botón derecho y abrimos propiedades.

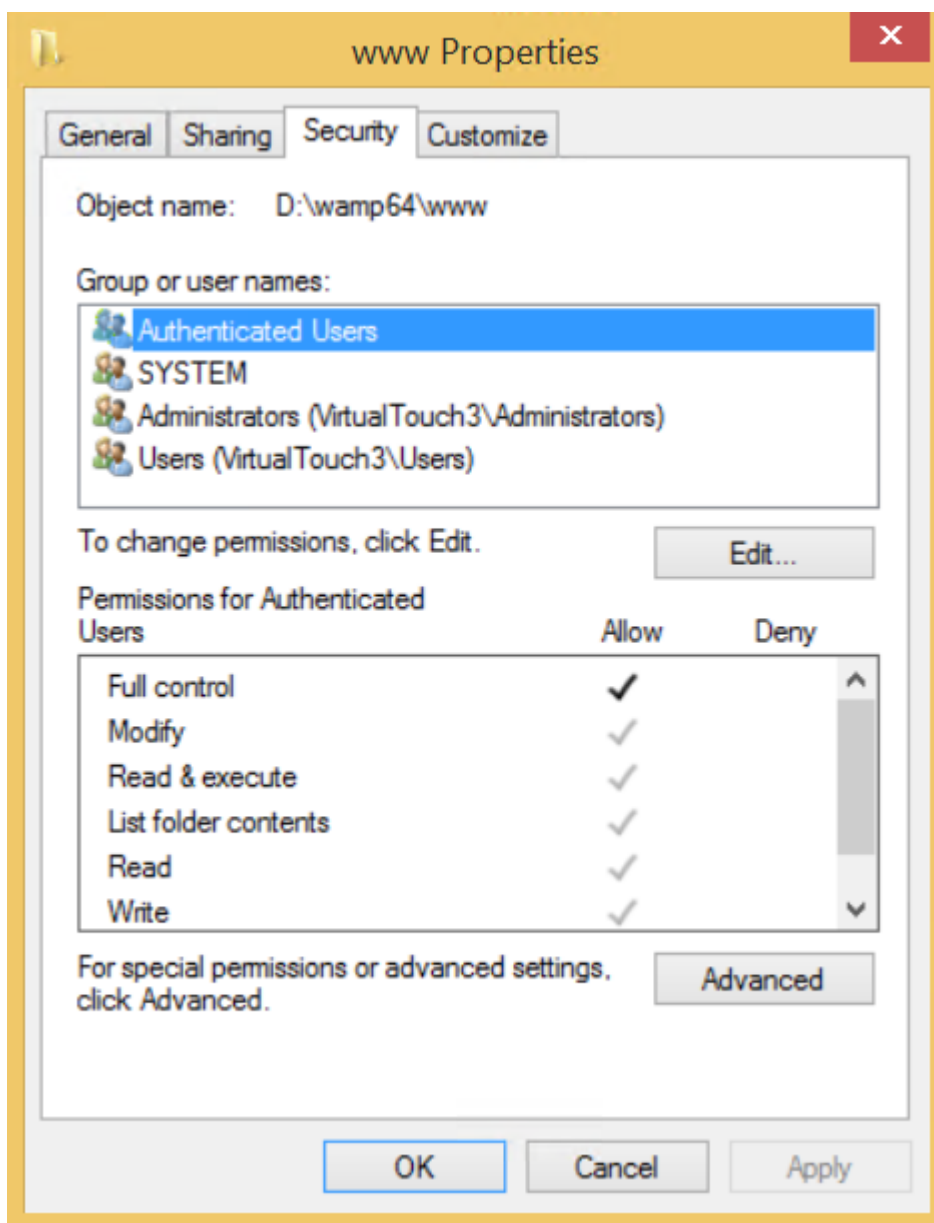


Figura 0.2.1: Cambiar permisos de una carpeta en Windows (1)

Después en Editar de la pestaña Seguridad seleccionamos todos los nombres de grupos y usuarios y vamos asignándoles permisos a cada uno, guardamos y volvemos a probar la conexión desde OpenSim.

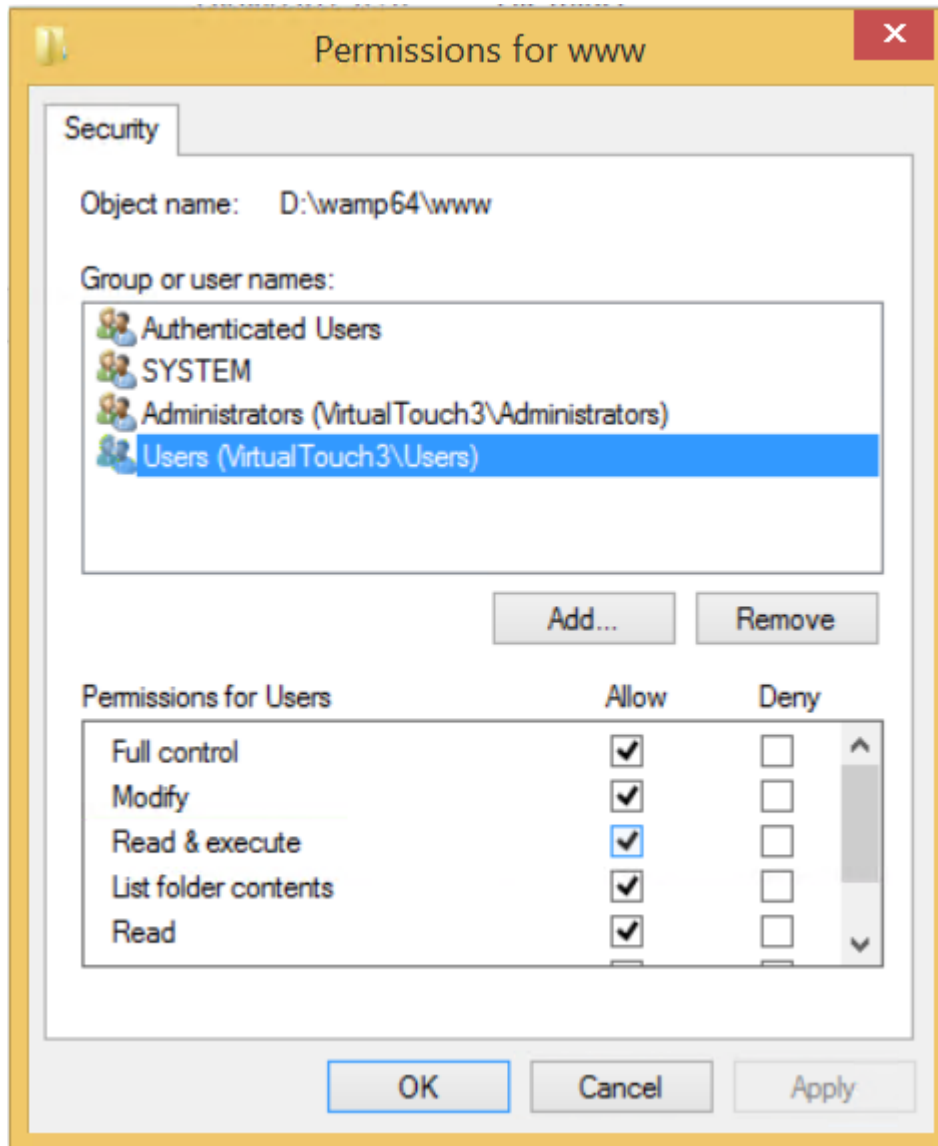


Figura 0.2.2: Cambiar permisos de una carpeta en Windows (2)

Puede existir aún otra posible fuente de error con la conexión, se debería cambiar la dirección a una dirección distinta de 127.0.0.1 o localhost puesto que a veces puede llevar a error, si no es el caso, también es posible que el cortafuegos nos esté cortando la conexión por el puerto 80. En este caso se recomienda habilitar las escuchas de Apache por otro puerto además del 80, por ejemplo, el 8080, quedando la dirección así (“http://xxxx.xxxx.xxxx.xxxx:8080/directorio/fichero.php?lista=mensaje”) Los pasos para configurar Apache se encuentran en el anexo A.

Por último, también se adjunta otro código de script para OpenSim en el que los parámetros que mandamos para la conexión se pasan por variables:

```

key clave;
string message="mensaje";
default
{
    state_entry(){
        touch_start(integer number)
        {
            clave=llHTTPRequest(http://xxx.xxx.xxx.xxx/directorio/fichero.php,
[HTTP_METHOD, "POST", HTTP_MIMETYPE, "application/x-www-form-
urlencoded"],"lista="+message
        }
        http_response(key id, integer status, list metaData, string body)
        {
            if(id==clave){
                llOwnerSay("Body :"+ body);
            }
        }
    }
}

```

Con este script debemos tener en cuenta que ahora en el fichero php se usará POST en lugar de GET.

Por último en la figura 0.2.3 se muestra el script del objeto definitivo que se comunica con el fichero PHP:

```

key clave;
integer canal;
string aux;
integer tienda=17345;
integer tienda2=17344;
integer tienda3=17343;
integer totem1=17350;
integer totem2=17351;
integer totem3=17352;
integer totem4=17353;
integer totem5=17354;
integer totem6=17355;
default{
state entry(){
    llListen(tienda,"","", ""); //tienda
    llListen(tienda2,"","", ""); //tienda
    llListen(tienda3,"","", ""); //tienda
    llListen(totem1,"","", ""); //totem1
    llListen(totem2,"","", ""); //totem2
    llListen(totem3,"","", ""); //totem3
    llListen(totem4,"","", ""); //totem4
    llListen(totem5,"","", ""); //totem5
    llListen(totem6,"","", ""); //totem6
}
listen(integer channel, string name, key id, string message){
    canal=channel;
    clave=llHTTPRequest("http://mundov.ii.uam.es/secondlife.php", [HTTP_METHOD, "POST",
        HTTP_MIMETYPE, "application/x-www-form-urlencoded"], "lista="+message);
}
http_response(key id, integer status, list metaData, string body){
    if(id==clave){
        llOwnerSay("Body :"+ body);
        llRegionSay(canal,body);
    }
}
}
}

```

Figura 0.2.3: Tótem Máster

Vemos que sigue la misma estructura que los ejemplos mencionados anteriormente, además se le han añadido varios canales de escucha, uno para cada grupo y otros tres canales de escucha, uno para cada tienda. Ahora se muestra una captura de uno de los comandos implementados en el fichero de comunicación:


```

<?php
$aux2=" ";
if(isset($_POST["lista"])){

$array=explode(" ",$_POST["lista"]);
$db=new PDO("mysql:dbname=virtualcraftv1; host=localhost","root","");

if(strcmp($array[0],"GRP")==0){
    $cont=intval($array[1]);
    $i=0;
    for($i=0;$i<$cont;$i++){
        $stmt = $db->prepare('SELECT grupo FROM clase WHERE nick LIKE :alumno');
        if (!$stmt) {
            echo "\nPDO::errorInfo():\n";
            print_r($dbh->errorInfo());
        }
        $string=explode("_",$array[$i+2]);
        $alumno = implode(" ", $string);

        $stmt->bindParam(':alumno', $alumno, PDO::PARAM_STR);

        $stmt->execute();
        $result=$stmt->fetchAll();
        if(count($result)==0){
            $aux2.=" ERR ";
        }
        foreach($result as $arrays3) {
            $aux2.=$arrays3["grupo"]." ";
        }
    }
    $ret=$_POST["lista"]." ".$aux2;
    // $ret=$aux . " ".$aux2;
    echo $ret;
}
}

```

Figura 0.2.4: Parseo de un comando del módulo de comunicación (1)

C Casos de uso de la pantalla de habilidades de los alumnos.

A continuación se van a mostrar las posibles combinaciones que pueden darse en la pantalla de habilidades, en la primera imagen se muestra el aspecto de la aplicación para un alumno que ha alcanzado el nivel 6 y tiene todas las habilidades compradas.



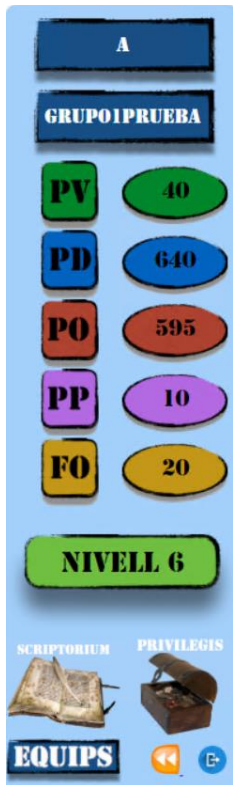
Figura 0.3.1: Habilidades. Todas compradas

En la siguiente imagen vemos el caso de un alumno que no tiene ninguna habilidad comprada y es de nivel 2 por lo que sólo están disponibles las habilidades de la columna 1 y las de la columna 2 se muestran opacas. Las habilidades que no estén disponibles por nivel o porque no se tienen los padres desbloqueados aparecerán en blanco y negro y las habilidades que estén no hayan sido compradas pero si disponibles estarán opacas.



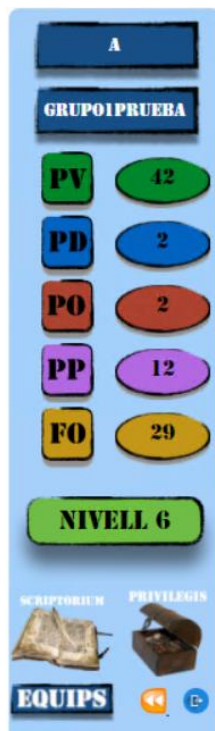
Figura 0.3.2: Habilidades. Ninguna comprada

Al pulsar sobre una habilidad que no esté disponible se mostrará un aviso por pantalla al alumno indicándoselo, si el alumno puede comprar la habilidad porque su nivel lo permite y tiene comprados todos los padres de esa habilidad, se le mostrará un mensaje por pantalla indicando cuanto cuesta comprarla. También se mostrarán mensajes indicando cuanto cuesta usar una habilidad. Estas dos situaciones de compra de habilidades o uso de ellas acaban las dos en dos posibles situaciones, o bien el alumno tiene puntos suficientes y al comprar o usar se le restan, o bien no tiene puntos suficientes. Estas dos situaciones vienen reflejadas en las siguientes dos imágenes que se muestran:



Guardado completado

Figura 0.3.3: Habilidades. No tiene puntos para comprar o usar



No tienes puntos suficientes

Figura 0.3.4: Habilidades. Se usa o se compra la habilidad

Según el alumno va comprando se irán poniendo disponibles las habilidades cuyos padres están desbloqueados, vease cómo ejemplo que al comprar la habilidad Màgia la habilidad Ressucitació no pierde la opacidad.



Figura 0.3.5: Habilidades. Ejemplo de compra, visible el bloqueo por los padres.

Por último para completar el caso anterior se muestra el caso en el que todos los padres estén comprados para una determinada habilidad, vemos ahora como Ressucitació pierde la opacidad y ya está lista para ser comprada.



Figura 0.3.6: Habilidades. Ejemplo de compra, se quita el bloqueo por los padres.

A continuación se adjunta un ejemplo de código que cumple las funcionalidades descritas anteriormente:

```

$stmt2 = $db->prepare('SELECT * FROM habilidades WHERE idhabilidad =:idhab');
$stmt2->bindParam(':idhab', $newstring, PDO::PARAM_INT);
$stmt2->execute();
$result2=$stmt2->fetchAll();
foreach($result2 as $arrays45){
    if (in_array($idfoto, $auxblog) ) {
        $stmt2 = $db->prepare('SELECT * FROM habilidadesmod WHERE idprofesor = :id and idhabilidadmod = :idhab');
        $stmt2->bindParam(':id', $profesor, PDO::PARAM_INT);
        $stmt2->bindParam(':idhab', $newstring, PDO::PARAM_INT);
        $stmt2->execute();
        $result2=$stmt2->fetchAll();
        foreach($result2 as $arrays53){
            $oficio=$arrays53["oficio"];
            $superacion=$arrays53["superacion"];
            $vida=$arrays53["vida"];
            $impuestos=$arrays53["impuestos"];
            $oro=$arrays53["oro"];
            $descripcion=$arrays53["descripcion"];
        }
    }else{
        $oficio=$arrays45["oficio"];
        $superacion=$arrays45["superacion"];
        $vida=$arrays45["vida"];
        $impuestos=$arrays45["impuestos"];
        $oro=$arrays45["oro"];
        $descripcion=$arrays45["descripcion"];
    }
}

```

Figura 0.3.7: Habilidades. Fragmento de código 1.

En la figura 0.3.7 se hace una consulta a la base de datos para coger el id de la habilidad que se va a mostrar, si la habilidad ha sido modificada por el profesor se cogerán los textos que el haya introducido nuevos, si no se cogen los textos por defecto de la habilidad.

```

if($arrays45['nivel']<=$nivel){
    $padres = explode("-", $arrays45["padres"]);
    $flag = 0;
    foreach ($padres as $p){
        if (in_array($p, $string) && $p!=0){
            $flag++;
        }
    }
    if (in_array($newstring, $string)){
        $mensaje="Aquest Privilegi et permet: ".$descripcion ." Vols portar a terme el teu Privilegi? Et costarà ". $impuestos ." PO";
        <div onclick=" if (confirm('<?php echo $mensaje; ?>')){ javascript:location.href='user-habilidad.php?id=<?php echo $arrays45["idhabilidad"]>'}">
        <?php echo <input type="image" name="habilidad".$arrays45["idhabilidad"]." src="data:image/jpeg;base64, ".base64_encode( $arrays4["img"] )." style="width:944"/>?>
        </div>
    }else{
        <?php <div onclick="alert('<?php echo $mensaje; ?>');">
        if ($flag == count($padres)){
            $mensaje="Aquest Privilegi encara no està adquirit. Comprer-lo et costarà: ". $oficio ." PO. Vols comprar-lo ara? Quan adquireixis aquest Privilegi podràs: ".$descripcion;
            $popup="return confirm('".$mensaje."');";
            echo <form id="user-habilidad".$arrays45["idhabilidad"]." action="guardar-habilidades.php" method="post" onclick="'.$popup.'">
            echo <input type="image" name="habilidad".$arrays45["idhabilidad"]." src="data:image/jpeg;base64, ".base64_encode( $arrays4["img"] )." style="width:944"/>?>
            echo </form>;
        }else{
            $mensaje="Aquest Privilegi no està encara disponible. Quan adquireixis aquest Privilegi podràs: ".$descripcion ;?>
            <div onclick="alert('<?php echo $mensaje; ?>');">
            <?php echo ?>
            </div>
        }
    }
}
}
else{
    $mensaje="Aquest Privilegi no està encara disponible. Quan adquireixis aquest Privilegi podràs: ".$descripcion ;?>
    <div onclick="alert('<?php echo $mensaje; ?>');">
    <?php echo ?>
    </div>
    <?php
}
}

```

Figura 0.3.8: Habilidades. Fragmento de código 2.

En el siguiente fragmento de código mostrado en la figura 0.3.8 se comprueba que el alumno tenga el nivel suficiente, si no lo tiene se mostrará en blanco y negro. Si lo tiene, pero no tiene a los padres saldrá opaca y si tiene el nivel suficiente saldrá en color.