

# Blexer-med

**Blender Exergames medical platform**

**Plataforma web médica y base de datos para el entorno  
de videojuegos terapéuticos “Blexer”**

Memoria técnica

Registro software

UNIVERSIDAD POLITÉCNICA DE MADRID



Autores: Mónica Jiménez Ramos, Martina Eckert

Fecha: Mayo de 2018

## Contenido

1. Descripción del programa .....	2
1.1. Objetivos .....	2
1.2. Realización.....	2
2. El lenguaje de programación.....	3
3. El entorno operativo .....	4
4. Listado de ficheros .....	6
5. Diagramas.....	6

## 1. Descripción del programa

### 1.1. Objetivos

La plataforma web “Blexer-med” forma parte del proyecto “Blexer” (Blender Exergames) y tiene como objetivo ser una herramienta de ayuda para médicos y fisioterapeutas que pretenden incorporar en la terapia o rehabilitación de sus pacientes videojuegos para realizar ejercicios físicos. Mediante el acceso por web, el profesional tendrá la posibilidad de elegir un juego adecuado, configurar los ejercicios incluidos en él y ajustar la dificultad. Además tiene la posibilidad de ver los resultados de cada jugador dado el alta en el sistema.

Los videojuegos a incluir dentro de este entorno se controlan por movimientos que se capturan mediante una cámara Kinect de Microsoft. Pueden ser de géneros diferentes, al gusto del paciente, en el momento del registro funciona con un juego tipo aventuras llamado “Phiby’s Adventures” en el cual el jugador usa diferentes movimientos de brazos y tronco para dirigir a un personaje por un ambiente de naturaleza, donde debe superar diferentes obstáculos.

### 1.2. Realización

La plataforma web, como herramienta de trabajo para los especialistas, debe consistir en una interfaz gráfica amigable para que terapeutas puedan realizar configuraciones de ejercicios y visualizar sus resultados y, por consiguiente, debe contar también con una base de datos robusta. Así, sumando los elementos mencionados, el entorno de videojuegos Blexer estaría constituido por los dispositivos (Kinect, *smartphone*,...), los videojuegos, el *middleware*, la plataforma web, y la base de datos. Con esto se lograría el desacoplamiento de las partes de forma que se podría modificar o incluso sustituir alguna de ellas sin afectar sustancialmente implementación de las demás.

La base de datos elegida para este proyecto es MySQL por ser, sobre todo, muy fácil de aprender a utilizar, pero también por tener una versión de libre distribución, por la cantidad de documentación que hay al respecto, por ser escalable y sencilla de modificar, y por ser multiplataforma (da soporte varios sistemas operativos como Windows, Linux y Mac), que puede ser útil en un futuro.

Por otro lado, se ha determinado que la plataforma web se basará en un Web Service al que podrá acceder tanto el terapeuta desde su navegador como el middleware CHIRO (también desarrollado bajo el proyecto Blexer) desde el PC del paciente. Este Web Service estará implementado en Java, el lenguaje orientado a objetos por excelencia. Esta elección se fundamenta en la disponibilidad de librerías de Java para desarrollo Web así como para acceso a bases de datos SQL (*Structured Query Language*).

Se ha implementado una interfaz gráfica para uso de terapeutas, accesible desde el navegador del personal médico, que a su vez accede un Servicio Web instalado en un servidor Apache Tomcat. Además se ha hecho uso de una base de datos MySQL para el almacenamiento de información personal de profesionales y pacientes, historial de configuraciones y resultados de los ejercicios.

Se han diseñado tres roles para los usuarios de la plataforma: terapeuta, administradores de diferentes centros médicos y un superadministrador de la plataforma, con distintos permisos cada uno, y se han utilizado algunas técnicas de encriptado para securizar las contraseñas, con el propósito de convertirla en segura y confidencial, otro de los objetivos iniciales.

Con el fin de potenciar la escalabilidad de la plataforma, se ha estructurado la base de datos de manera que un juego consta de uno o más ejercicios, con la intención de que se puedan añadir, modificar o borrar si es necesario, permitiendo adaptar los juegos y ejercicios disponibles al desarrollo de la investigación global.

## 2. El lenguaje de programación

Para almacenar la información de usuarios, pacientes, configuraciones, resultados de los juegos, etc., se ha utilizado una base de datos **MySQL** que reside en un servidor MySQL. Se ha utilizado la versión *Community Edition* de MySQL por ser de uso libre y estar disponible para alrededor de 20 plataformas y sistemas operativos como Linux, Mac y Windows. Tiene implementados conectores para diferentes lenguajes, y en este proyecto se usa como conector **JDBC** (*Java Database Connectivity*), una API (*Application Programming Interface*) de Java para acceder desde el servidor Web a la base de datos MySQL.

Para el desarrollo Web, se ha utilizado la versión **HTML5** (*Hypertext Markup Language*). Como lenguaje de programación se ha elegido JavaScript, porque al ejecutarse en *front-end* (lado del cliente) se ahorra tiempo de petición/respuesta al servidor, por ejemplo a la hora de validar formularios, y la respuesta al usuario es prácticamente inmediata. A través de JavaScript, el HTML – a pesar de ser estático– puede ser accedido y modificado y así tener una interfaz dinámica.

Las capas de presentación y negocio se han desarrollado en el entorno de desarrollo **Eclipse JEE Luna** para Java. La capa de presentación está implementada en un conjunto de documentos HTML, JS, imágenes, hojas de estilo en la carpeta *WebContent* del proyecto de Eclipse. Tiene componentes de presentación (HTML y CSS) y componentes de procesado (JavaScript). La Capa de Presentación y la Capa de Negocio se comunican entre sí mediante peticiones HTTP que se realizan usando funciones de jQuery para solicitar/enviar un recurso a la Capa de Negocio. Dichas funciones devuelven texto plano a interpretar por la capa de Presentación y mostrar al usuario de forma amigable. La Capa de Negocio está programada en lenguaje **Java**. Para que la Capa de Negocio sea accesible a través de HTTP se ha implementado un servicio web **RESTful** (*Representational State Transfer*). La interfaz

elegida a través de la cual se accede al servicio web es la API **JAX-RS** (*Java API for RESTful Web Services*).

### 3. El entorno operativo

De cara al proyecto Blexer se puede dividir las comunicaciones en tres partes:

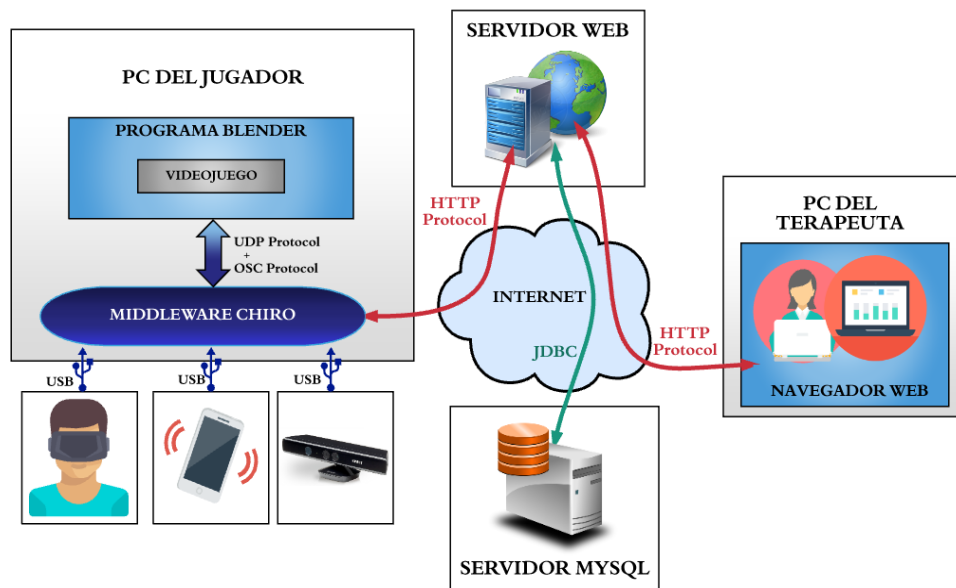
- Interacción de usuarios con el servidor Web para acceder a la plataforma.
- Comunicación del middleware CHIRO con la plataforma para recibir configuraciones de juegos y enviar resultados.
- Comunicación del Servidor con la Base de datos para obtener la información que el Servidor responde a CHIRO o al usuario.

Las tres comunicaciones, representadas en la Figura 1, se basan en el modelo Cliente/Servidor. Éste es un sistema distribuido en el que el cliente solicita un recurso al servidor y éste le responde.

Por un lado, el navegador (Cliente Web) del especialista carga la aplicación (Servidor Web) mediante peticiones/respuestas HTTP. Éste es un protocolo para la transferencia de texto plano.

Por otro lado, CHIRO además de todo lo que hacía en el punto de partida se comportará como Cliente Web para solicitar a la aplicación la configuración de los ejercicios y, del mismo modo, enviar los resultados.

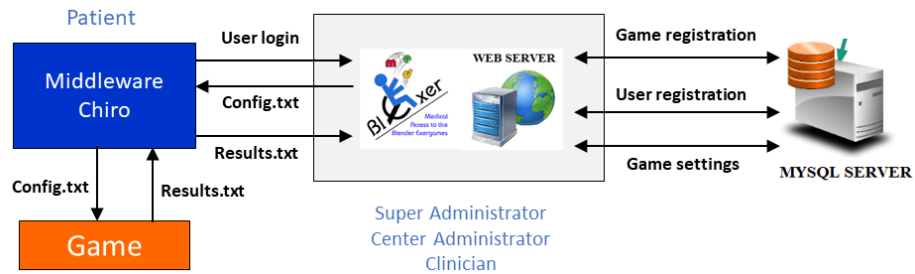
En ambos casos, cuando el Servidor Web recibe una petición, como puede ser el inicio de sesión del terapeuta o cuando CHIRO solicita un recurso a la plataforma, se accede desde la plataforma Web (Cliente MySQL) a la base de datos MySQL (Servidor MySQL) en forma de petición/respuesta.



**Figura 1. Estructura del entorno y comunicaciones**

Al iniciar CHIRO, lo primero es que pide la contraseña y el usuario se identifica. Con estos datos, CHIRO se conecta a la plataforma web, que se alberga en el servidor del CITSEM, y recibe los datos

de configuración (Figura 2), los cuales guarda en el disco local. El usuario ejecuta el juego y éste lee los datos de configuración. Durante el juego, CHIRO se encarga a transmitir los datos de los sensores conectados. El videojuego almacena los resultados de los ejercicios (puntuaciones, tiempos etc.) en el disco local durante el juego y al terminar este (cerrando CHIRO), éste los envía a la plataforma web desde la cual se almacenan en la base de datos. De esta manera, el terapeuta puede supervisar el avance del jugador y, en su caso, reajustar la dificultad de los ejercicios.

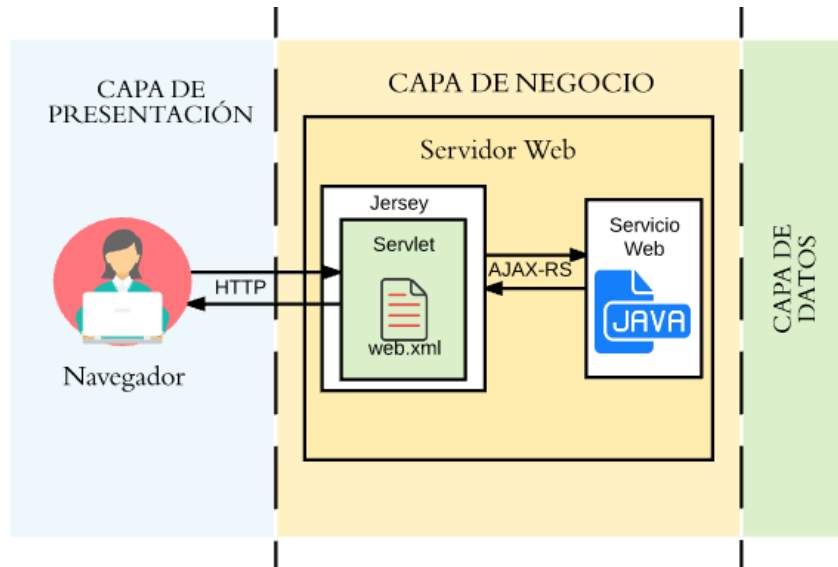


**Figura 2. Intercambio de información entre CHIRO, el juego y la plataforma web**

Para Blexer-med, el dominio es “blexer-med.citsem.upm.es”, el puerto es el de Tomcat (8080), el nombre de aplicación “blexer-med”, y el patrón de url del servlet dado en *web.xml* es “/ws/\*”. Así, el servlet interceptará las peticiones dirigidas a

<http://blexer-med.citsem.upm.es:8080/blexer-med/ws/>.

En la Figura 3 se presenta un esquema de los componentes de la Capa de Negocio que permiten el acceso al servicio Web desde la Capa de Presentación.



**Figura 3. Esquema de acceso a Capa de Negocio**

## 4. Listado de ficheros

Workspace Blexer.zip

## 5. Diagramas

Como se trata de una plataforma web con base de datos, en vez de un diagrama de flujo, se presentan la relación entre las clases de la capa de negocio (figura 5) y la estructura de la base de datos (figura 5).

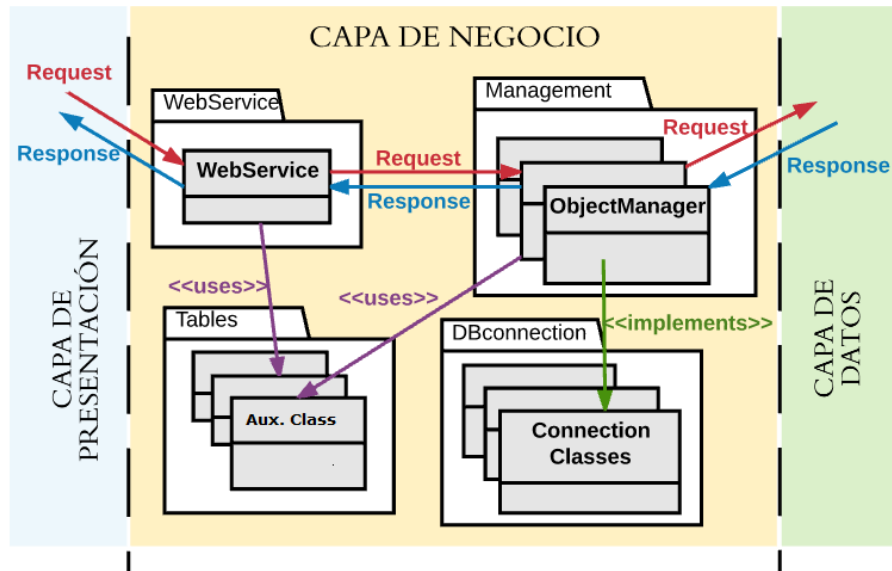


Figura 4. Relación entre las clases que componen la Capa de Negocio de Blexer-Med

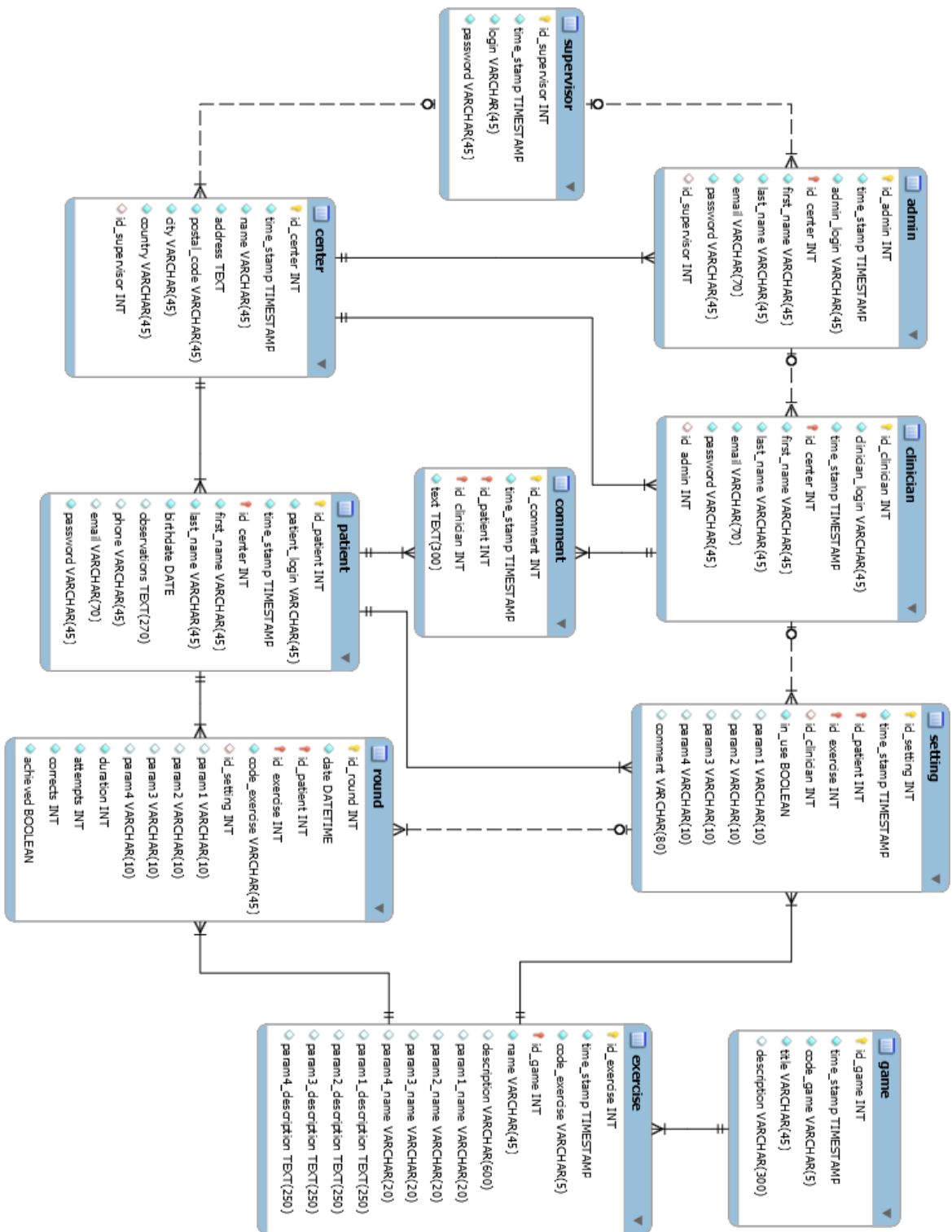


Figura 5. Modelo de la base de datos de Blexer-med