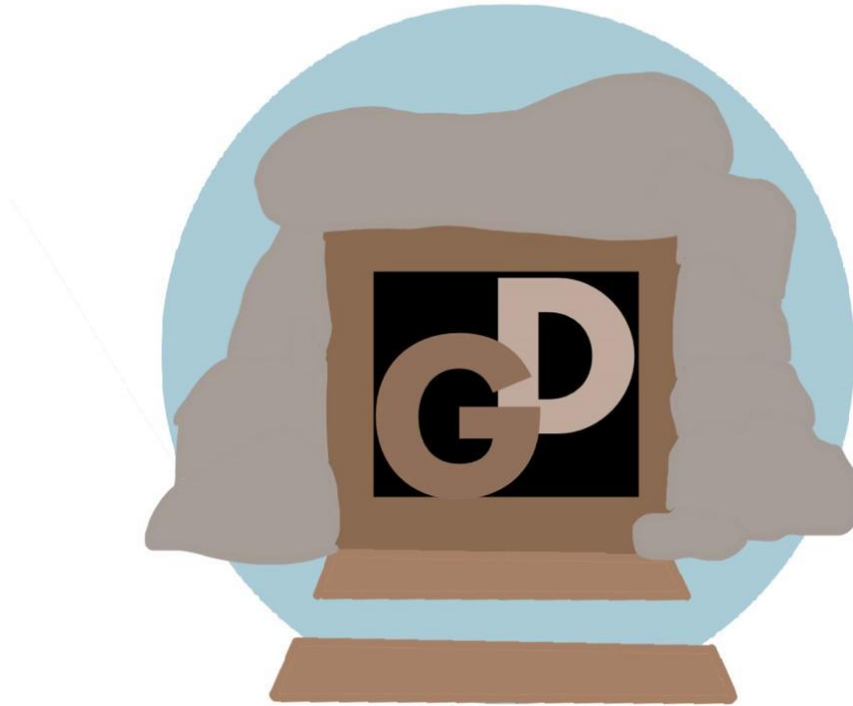


Memoria Práctica 3



Miembros:

- David Cruza Sesmero
- Arturo Lopez Fernandez
- Camilo Andres D'isidoro
- Antonio Bonillo Saiz
- Luis Manuel Perez Cuadrado
- Jesús Abajo Magro

Índice

1. Descripción del proyecto
2. Objetivos a alcanzar en esta práctica
3. Base de datos: Estructura y funcionamiento
4. Archivos implementados, explicación de sus funcionalidades
5. Tabla de gestión del proyecto

1. Descripción del proyecto

Nuestra página web se centrará sobre todo en un comparador de videojuegos; en dicho comparador estableceremos distintas funcionalidades donde podamos implementar a varios actores (usuario registrado, no registrado, administrador, administrador de foro, etc...), todo esto girará en torno a una base de datos donde se gestione las preferencias de cada usuario y según su rol asignado cada uno de los usuarios tendrá funcionalidades distintas; a su vez implementaremos funciones donde distarán de la idea principal tales como venta de segunda mano, foro, calendario de salida de noticias con noticiero interno, chat “en línea”, etc...

2. Objetivos a alcanzar

En esta práctica los objetivos a alcanzar son los siguientes:

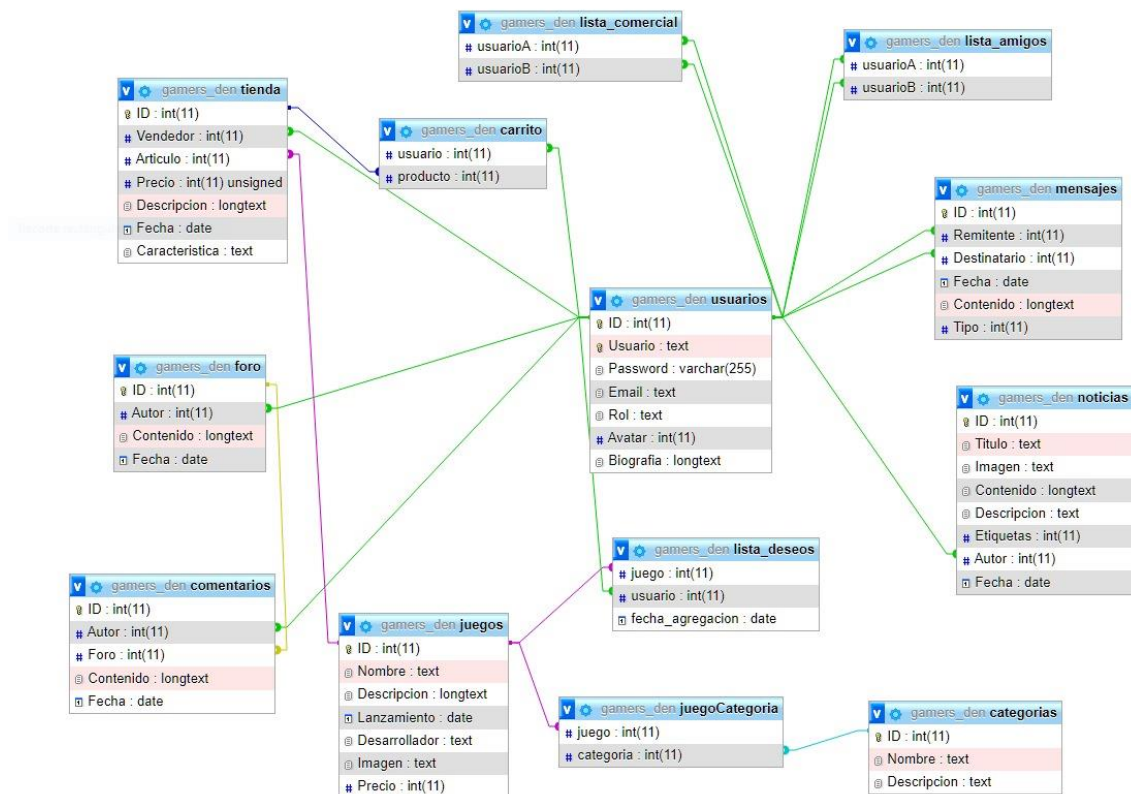
Práctica 2:

- Detallar la base de datos y su estructura
- Implementar tres funcionalidades: Perfil y chat “en línea”
- Gestión de inicio de sesión y registro de los usuarios
- Implementar distintos roles para los usuarios
- Creación de una memoria para detallar la práctica y nuestro trabajo

Práctica 3:

- Reformulación de la base de datos y arreglar errores con hash contraseña
- Implementar funcionalidades de tienda y de noticias
- Acabar con la función de chat e implementación de la misma en tienda
- Desarrollar las CSS implementadas en la P2
- Creación de una memoria para detallar la práctica y nuestro trabajo

3. Base de datos: Estructura y funcionamiento



En nuestra base de datos para esta etapa del proyecto contamos con 6 tablas, las cuales se detalla su funcionamiento a continuación:

- **Usuarios:** Esta tabla es el componente principal de la base de datos actualmente. Cada usuario contará con un Identificador en forma de número autoincremental, un nombre de usuario, una contraseña cifrada con un algoritmo SHA, un correo electrónico, un Rol (En el formulario de registro al público cargará por defecto el valor Usuario, y el administrador podrá de dar alta roles privilegiados desde su plataforma de acceso), un número identificador para el avatar (Dispone de 20 posibles opciones para elegir un avatar, con números adicionales para identificar las fotos de perfil de los roles privilegiados), y una Biografía, donde cada usuario podrá personalizar su perfil con una frase o información sobre este.
- **Juegos:** La tabla de juegos es otro componente importante de la BD, aunque actualmente no tendrá aplicación en esta etapa del proyecto. En esta guardaremos la información relativa a los videojuegos que haya en el mercado, como su nombre, desarrolladora, fecha de lanzamiento, descripción, y el precio oficial lanzado por la desarrolladora para este juego. La tabla de juegos está directamente vinculada a la tabla de Tienda, detallada a continuación.

- **Tienda:** Dentro de la tienda contaremos con un ID de la publicación de la tienda, así como un ID que relacione al Vendedor con un ID de la tabla Usuarios, y un ID de producto relacionado a la tabla de Juegos, y el precio que se haya establecido para ese juego.
- **Mensajes:** Esta tabla es bastante importante para la funcionalidad del chat. En esta contamos con un ID único para cada mensaje, y relacionamos con la tabla Usuarios para obtener el ID del remitente y el destinatario. También contamos con un campo para el texto del mensaje, y una fecha de envío que será por defecto un Current Timestamp.
- **Foro:** Esta tabla es fundamental para el funcionamiento del foro y sus comentarios. En esta tendremos un ID único para cada entrada del foro, un ID del autor relacionado a la tabla usuarios, el contenido del foro y la fecha en la que fue creado.
- **Comentarios:** Dentro de esta tabla guardaremos los comentarios que se hayan hecho dentro de una entrada del foro. Para ello tendremos, además de un ID para cada comentario, el ID del autor de ese comentario, y el ID de la entrada del foro al que va relacionado, el contenido de dicho comentario, y la fecha en la que fue realizado.
- **Lista Amigos:** Aquí tendremos una tabla de relación Muchos a Muchos para darle funcionalidad a la lista de amigos (Un usuario A puede tener muchos amigos B, y a su vez, puede ser amigo de muchos usuarios B).
- **Lista Deseos:** En esta tabla tendremos la información relativa a la lista de deseos de los usuarios, siendo de relación muchos a muchos (Un usuario puede tener muchos juegos en su lista de deseos, y un juego puede estar en muchas listas de deseos), incluyendo también la fecha en que fue añadido el juego a esa lista.
- **Noticias:** En este apartado tendremos los atributos necesarios para recoger la información referente a las noticias que se ofrecen dentro de nuestra página web. Los distintos atributos que podemos ver los mostramos en el dibujo.
- **Carrito:** Al usuario cliente se le asignará un producto que haya acordado con un vendedor.
- **Lista comercial:** con esta lista gestionamos los vendedores y clientes que han llegado a un acuerdo por un producto.

Esta Base de datos está sujeta a cambios y modificaciones a lo largo del desarrollo del proyecto, pudiendo implementarse más funcionalidades o eliminar algunas para adaptarlo al producto final

1. Archivos implementados, explicación de sus funcionalidades

/css

- **Estilos.css:** Estilos aplicados a la página web

Includes/mysql

- Aquí encontramos las tablas referentes a la base de datos, tanto los modelos como tablas rellenas con datos
- **centro.php:** Mostrará el contenido principal de la página, habiendo un contenido público, tanto para usuarios registrados como no registrados, y un contenido restringido solamente para usuarios registrados
- **usuarios.php:** Es el encargado de comprobar si hay un usuario logeado o no, así como de realizar las operaciones de inicio y cierre de sesión.
- **usuariosbd.php:** En este archivo básicamente establecemos la clase usuario con sus distintos atributos a su vez tenemos las diferentes funciones que nos busca el id del user, nos devuelve distintos parámetros (contra, user, etc...).

/includes/vistas/común

- **pie.php:** Breve descripción de la página web.
- **sidebar.php:** Contiene la barra de navegación de la página web, en la que de momento tiene un botón para el Home, para el perfil del usuario y para el chat.
- **cabecera.php:** Muestra el logo de nuestra tienda con un menú para poder navegar por la página, un apartado para iniciar sesión o registrarse en el caso de que no estés registrado y el logo.

/includes/vistas/plantillas

- **plantilla.php:** Es donde vertemos nuestra variable de \$contenidoPrincipal para que se vayan mostrando las funcionalidades

/includes

- **Aplicación.php:** En este archivo gestionamos la clase aplicación donde nos encontramos principalmente las funcionalidades propias para iniciar, cerrar, controlar si está iniciada o establecer una conexión con la bd.

- **Categoría.php:** Se establece los atributos propios a la clase de categoría, estableciendo el constructo propio de la clase y sus distintos getters; a parte de ello las distintas funcionalidades propias con `editarCategoría`, `buscarporId`, `borrarCategoría`, etc...
- **Formulario.php:** Clase padre del resto de formularios donde tenemos las funciones de gestión, creación, envío, procesamiento y generación de formularios junto con los atributos principales de los mismos.
- **FormularioAmigos.php:** Este archivo hereda de la clase `Formulario`, en este archivo realizamos un formulario para que el usuario pueda añadir amigos que se encuentren dentro de la base de datos.
- **FormularioaddListaDeseos.php:** Se establece un formulario en el que principalmente se asocia un usuario con el id de un videojuego para que este sea agregado a una lista de "deseos" donde se refleje que x usuario tiene un interés en x juego.
- **FormularioCambiarAvatar.php:** Establecemos un formulario para que le usuario pueda proporcionar otra imagen para que represente su avatar. Esto mismo nos encontramos con **CambiarBio.php**
- **FormularioBusquedaNoticia.php:** esta clase hereda de `FormularioGestionManual`, en dicho archivo realizamos una búsqueda de una noticia mediante una palabra clave. Esto también se da con **BusquedaJuegos.php** y **BuscarProductos.php**.
- **FormulariosCreación:** Mediante el uso de formularios que extienden de `Formulario` gestionamos distintos formularios asociados a **Negociación, Categoría, Juegos, Noticia y Producto**.
- **FormulariosEdición:** Mediante el uso de formularios el usuario determinado por su rol podrá editar las clases asociadas a **Negociación, Categoría, Juegos, Noticia y Producto**.
- **FormularioEliminarAmigos.php:** Formulario para eliminar amigos, utilizamos las funciones correspondientes a usuarios como búsqueda de usuario en la bd y después de comprobar si son amigos dichos usuarios se procede a eliminarlo de la base de datos como amigo. Nos encontramos el mismo caso en los

siguientes archivos de eliminación: **Categoría, Deseos, Juego, Noticia y Producto.**

- **FormularioGestionManual.php:** esta clase extiende de formulario y lo utilizamos para que nos devuelva después del envío de un formulario una clase de resultadoGestionFormulario.
- **FormularioLogin.php:** Formulario que comprueba si los datos introducidos de un usuario coinciden con los que hay en la base de datos y por tanto te permite hacer login en la app.
- **FormularioMandaMensajes.php:** Establece un formulario para enviar mensajes donde antes de nada checkea si el destinatario de dicho mensaje es amigo con el que envía el mensaje a su vez se utilizan clases correspondientes a Usuario como addMensaje donde ya todo esto va a formar parte del chat. Cabe resaltar que este archivo se divide en dos ámbitos: Amigo y Vendedor, según si se trata de un chat normal con un amigo o establecemos una negociación con un vendedor.
- **FormularioRegistro.php:** Establecemos un formulario de registro donde le pedimos al usuario que nos proporcione un nombre de usuario, un email y una contraseña la cuál se le añadirá una doble verificación para que no haya errores a la hora de introducir la contraseña.
- **Mensaje.php:** Establece la clase mensaje proporcionando las distintas funcionalidades que hacen que se retorne los distintos atributos necesarios de la clase.
- **Noticia.php:** Aquí tenemos la clase noticia donde gestionamos los atributos de la misma así como mostramos las distintas funciones que vamos a utilizar para con ella; tales como: subeNoticia, cargarNoticia, buscaNoticia, cargarCaracterísticas y los distintos getters que necesitamos para recoger los atributos más directos de la misma.
- **Producto.php:** clase que especifica los atributos propios de producto, esta clase es muy parecida a videojuegos únicamente que esa clase se utiliza para referenciar los videojuegos que tiene un usuario en su perfil y esta se utiliza

para ver todos los videojuegos disponibles para los usuarios, así como funciones para subir productos a la tienda o enseñar por características.

- **ResultadoGestionFormulario.php:** Aquí tenemos los resultados propios de la gestión de los formularios como si se ha enviado con éxito, si ha habido errores a la hora de procesarlo o los resultados aparentes que saca dichos formularios.
- **Usuario.php:** En la clase usuario gestionamos los atributos principales del usuario, desde sus datos a los roles que puede tener el usuario.
- **Videojuegos.php:** Clase que sirve para mostrar los atributos de los videojuegos de lo cuales dispone el usuario en su perfil, tiene un matiz distinto a productos porque estos son los que tiene referenciado en sus listas.
- **Config.php:** Tiene los parámetros de configuración básicos para el funcionamiento de la página, como el charset, el inicio de una sesión, y la conexión a la base de datos.

./.

- **addAmigo.php:** Nos proporciona un formulario que proviene de la clase formularioAmigos.
- **Bienvenido.php:** Establecemos una página de bienvenida mediante un pequeño script que a la vez muestra nuestro logo.
- **buscarNoticia.php:** En este archivo utilizamos el formulario 'FormularioBusquedaNoticia' donde buscamos una noticia mediante una palabra clave.
- **buscarJuego.php:** Inicia la gestión del formulario búsqueda de videojuegos.
- **buscarProducto.php:** Inicia la gestión del formulario búsqueda de productos.

- **chat.php:** En este archivo se gestiona el chat desde su aspecto a como gestionamos con el set interval() que emula que el chat es en línea puesto que establecemos un refresco.
- **Chat_aux.php:** En este archivo lo que hacemos es volcar lo que hemos recogido en el archivo de log en la base de datos en el campo mensajes a su vez mostramos el chat particular del usuario que este utilizando dicha función en ese momento junto con el avatar y nombre del destinatario.
- **Chat_amigo.php:** genera un chat mediante formularioMandaMensajesAmigo reconoce la lista de amigos del usuario selecciona el id del usuario amigo y le proporciona los mensajes correspondientes a su chat.
- **Chat_negocio.php:** mediante las funciones dadas por chat auxiliar, establecemos esta vez un chat con un vendedor para negociar sobre un producto.
- **crearX.php:** En cada uno de estos documentos: Categoría, Productos, Noticia y Juego creamos instancias de cada una de estos objetos los cuales se gestionan con los distintos formularios de creación que dispone.
- **editarX.php:** Formularios de edición de las categorías anteriormente mencionadas.
- **Juego_particular.php:** presenta las funciones necesarias para la gestión de los juegos particulares del usuario así como la gestión y funcionalidad de los distintos botones como edición o creación.
- **eliminarAmigo.php:** Utilización de 'FormularioEliminarAmigos'; básicamente hacemos un delete de las tablas correspondientes básicamente de lista de amigos
- **cambioAvatar.php:** formulario para cambiar el avatar del usuario.
- **cambioBio.php:** formulario para cambiar la biografía del usuario.
- **inbox.php:** generación de funciones y contenido para chat particular tales como generarAvatar o generarChat.
- **index.php:** página principal de Gamers Deen.
- **mostrarBusquedaNoticias.php:** capturamos en la variable contenido principal el formulario de búsqueda de noticia así como mostramos las noticias que hay disponibles en ese momento en la base de datos.

- **mostrarBuscarJuego.php:** capturamos en la variable contenido principal el formulario de búsqueda de juego así como mostramos los juegos que hay disponibles en ese momento en la base de datos.
- **NotFound.php:** Es un archivo donde se muestra una imagen de un gato para denotar que esa página está en construcción o todavía no se ha diseñado.
- **Noticias_concretas.php:** Búsqueda de noticia individual, esto se hace mediante una función llamada buscaNoticia que se encuentra dentro de Noticia.php.
- **Noticias_principal.php:** aquí mostramos las noticias principales divididas por tags o las características que disponen las mismas.
- **Noticias_aux.php:** Se muestra mediante la función de generaContenidoPrincipal el contenedor de noticias junto con las distintas secciones de noticias (Nueva, destacada, popular), función de búsqueda y un cuadro de noticias donde las establecemos.
- **Tienda.php:** Nuestra tienda donde se exponen los productos ("Videojuegos") mediante una disposición de rejilla y a su vez las imágenes de los juegos y sus propios nombres tienen un enlace que deben redirigirte a tienda_particular.php.
- **Tienda_particular.php:** es parecido al chat particular, una vez has seleccionado un producto te lleva a una página donde se muestran los datos de dicho producto así como proporcionar un botón que te permita establecer un chat con el vendedor.
- **registro.php:** Muestra la interfaz para poder registrarse.
- **login.php:** Presenta la funcionalidad necesaria para logear a un usuario con control de errores.
- **logout.php:** Funcionalidad necesaria para hacer el logout de un usuario de nuestra app.
- **Perfil.php:** Aquí se mostrará toda la información del usuario que haya iniciado sesión, así como su lista de amigos, su foto de perfil, y lista de deseos.

- **Perfil_aux.php:** Se establecen distintas funcionalidades útiles para la generación del perfil del usuario como su lista de amigos, su foto de perfil, y lista de deseos, la gestión del contenido principal donde se muestra dichos resultados.

5. Tabla de gestión del proyecto

Aquí mostraremos de una manera rápida la asignación de los miembros del equipo a las distintas funcionalidades propuestas; para ver la “carga de trabajo” de cada alumno esta se mostrará en Github mediante la pestaña *Insights/Contributors*, cabe resaltar que muchas veces como hacemos por parejas una determinada tarea una persona es la encargada de subir dicho código, así que el “esfuerzo” será detallado por cada miembro del grupo con las encuestas que se proporcionan el día de la entrega de cada práctica.

Todo el contenido del proyecto ha sido revisado por todo el conjunto de miembros.

El enlace a nuestra carpeta de Github se muestra en el pie de página.

Práctica 3

Desplegar aplicación en VPS	<ul style="list-style-type: none">• Camilo Andres D'isidoro
Tienda funcional	<ul style="list-style-type: none">• David Cruza Sesmero• Arturo López Fernández• Camilo Andrés D'isidoro
Tienda particular	<ul style="list-style-type: none">• David Cruza Sesmero• Arturo López Fernández• Camilo Andrés D'isidoro
Chat e implementación del mismo	<ul style="list-style-type: none">• David Cruza Sesmero• Arturo López Fernández
Noticias	<ul style="list-style-type: none">• Antonio Bonillo Saiz• Jesús Abajo Magro• Luis Manuel Pérez Cuadrado
Noticias particulares	<ul style="list-style-type: none">• Antonio Bonillo Saiz• Jesús Abajo Magro

	<ul style="list-style-type: none">• Luis Manuel Pérez Cuadrado
Estilos CSS mejoras	<ul style="list-style-type: none">• Arturo López Fernández• Antonio Bonillo Saiz• David Cruza Sesmero (Solo rejilla)
Memoria práctica 3	<ul style="list-style-type: none">• Camilo Andrés D'isidoro• David Cruza Sesmero
Implementar/Arreglar funciones perfil usuario	<ul style="list-style-type: none">• Arturo López Fernández• Antonio Bonillo Saiz

Práctica 2

Layout funcional	<ul style="list-style-type: none">• Luis Manuel Perez Cuadrado• Arturo Lopez Fernandez
Base de datos	<ul style="list-style-type: none">• Camilo Andres D'isidoro• David Cruza Sesmero
Procesamiento login usuario	<ul style="list-style-type: none">• David Cruza Sesmero• Camilo Andres D'isidoro
Registro	<ul style="list-style-type: none">• David Cruza Sesmero• Camilo Andres D'isidoro
Chat "en línea"	<ul style="list-style-type: none">• David Cruza Sesmero• Arturo Lopez Fernandez
Creación Perfil Usuario	<ul style="list-style-type: none">• Antonio Bonillo Saiz• Jesús Abajo Magro
Creación y desarrollo de la memoria	<ul style="list-style-type: none">• David Cruza Sesmero• Camilo Andres D'isidoro