



ParticulAres

ARQUITECTURA Y PROTOTIPO FUNCIONAL

Miguel Ángel Castillo Moreno

Fernando Cortés Sancho

Jorge García Cerros

Arturo Aguirre Calvo

Raúl Vindel Loeches

José Morcuende Sierra

Índice

Introducción.....	3
Descripción.....	3
Scripts.....	3
Clases.....	7
Estructura de la base de datos.....	12
Instrucciones de instalación.....	18

Introducción

Este es un sitio web para que profesores y alumnos se organicen a la hora de dar y recibir clases particulares.

Los alumnos podrán encontrar al profesor perfecto aplicando filtros por materia, nivel, horario, localización y valoración, a la vez que se organizan con el calendario que ofrecemos. Mientras, los profesores pueden darse a conocer y organizar sus clases a su gusto. De esta forma acabamos con la forma rudimentaria de las clases particulares como los carteles en los postes y números de teléfono poco fiables.

Descripción

Como tutor, podrás registrarte por un módico precio, subir asignaturas que quieres impartir, el nivel, la zona mediante un pequeño mapa y a qué horas. Además, podrás adecuar tu perfil dentro de la web según tus gustos para intentar atraer estudiantes, incluyendo fotos, videos, valoraciones de otros alumnos, etc.

Como alumno, podrás buscar por varios filtros como materia y nivel, para encontrar al profesor que mejor te convenga. Puedes puntuar y comentar profesores, chatear con ellos, enviarles archivos... Además, puedes anotar las clases a las que te suscribas en un cómodo calendario.

El administrador puede borrar tanto usuarios como clases, sin tener que saber nada sobre base de datos o de cómo funciona la aplicación.

Scripts

- ad.php

Se trata del script encargado de consultar en la base de datos la información sobre un anuncio de un profesor y mostrar esa información correctamente.

- myAds.php

Muestra los anuncios que has creado. Solo se tiene acceso si eres profesor.

- chat.php

Encargado de consultar en la base de datos y mostrar la conversación entre un profesor y un alumno. Se verán primero los mensajes más recientes hasta un máximo determinado. Si se quiere ver más mensajes habrá un botón de "ver más".

- createAd.php

Se encarga de procesar el formulario con toda la información sobre un anuncio que quiere publicar el profesor.

- editMyProfile.php

Se encarga de procesar el formulario que permite a un usuario modificar su información personal.

- header.php

El header realiza la funcionalidad de buscador (salvo en la página principal que el buscador será un elemento más importante), por lo que procesa la información del buscador y consulta en la base de datos en función de esa búsqueda.

Además, tendrá que distinguir si se trata de un usuario con la sesión ya iniciada o no, para mostrar los botones con las acciones correctas para cada momento.

- footer.php

Se encarga de mostrar información en la parte inferior de la página, como enlaces a "Sobre nosotros" o las redes sociales.

- index.php

Como hemos dicho anteriormente, contendrá el buscador, por lo que procesa su información y consulta en la base de datos.

Debajo del buscador aparece una sección de novedades en la que se carga de la base de datos los últimos anuncios de los profesores.

- login.php

Procesa la información recogida por el formulario y realiza la consulta requerida en la base de datos para comprobar que ese usuario es correcto.

- logout.php

Elimina la información del usuario actual, borrando las variables de la sesión y destruyendo dicha sesión.

- messages.php

Consulta en la base de datos y muestra todas las conversaciones que tiene el usuario.

- novedades.php

Muestra los 3 mejores anuncios de forma acortada, ordenados por puntuación media.

- payment.php

Página en la que se pondría el procesamiento del pago si existiese, aunque no se utiliza ahora mismo.

- showProfile.php

Debe distinguir si se trata del perfil propio o del perfil de otro usuario para, en este último, redirigir a showMyProfile.
Consulta en la base de datos la información sobre el usuario.

- showMyProfile.php

Muestra la información del usuario logueado y permite modificarla.

- search.php

Procesa la información sobre la búsqueda, realiza la consulta adecuada a la base de datos y muestra el resultado.

- signIn.php

Procesa la información del formulario, comprueba que es válida y añade el usuario a la base de datos.

- myCalendar.php

Muestra el calendario del usuario, junto con las clases a las que estás suscrito.

- download.php

Descarga un archivo de mensaje. Comprobará que tengas permiso para descargártelo.

- admin.php

Pequeño panel de admin, que permite borrar usuarios y anuncios.

Clases

- Advertisement

Contendrá la lógica necesaria para realizar operaciones sobre los anuncios de los profesores.

Está compuesta por las funciones:

- creaAnuncio(\$id_profesor, \$materia, \$nivel, \$idioma, \$descripcion, \$zona, \$precio, \$puntuacion_media, \$num_puntuaciones)
- borraAnuncio(\$id_anuncio)
- getAnuncios()
- getAnunciosByAuthor(\$id_profesor)
- getBestAnuncios()
- getAnuncio(\$id_anuncio)
- buscaAnuncio(\$materia, \$nivel)
- getNiveles()
- Constructora y getters

- User

Contiene la lógica necesaria para realizar operaciones sobre los usuarios.

Está compuesta por las funciones:

- login(\$nombreUsuario, \$password)
- buscaUsuario(\$nombreUsuario)
- buscaUsuarioPorId(\$id)
- crea(\$array)
- hashPassword(\$password)
- guarda(\$usuario)
- inserta(\$user)
- actualiza(\$user)
- getDemasUsuarios(\$idUsuario)
- puntuarUsuario(\$idUsuario, \$puntuacion, \$idPuntuado)
- yaHaPuntuado(\$idPuntua, \$idPuntuado)
- setPhoto(\$file)
- delete()
- Constructora y getters.

- Message

Contiene la lógica necesaria para realizar operaciones sobre los mensajes.

Contiene las funciones:

- historialMensajes(\$idEmisor, \$idReceptor)
- searchById(\$id)
- ultimosMensajes(\$idEmisor)
- crea(\$idEmisor, \$idReceptor, \$texto)
- inserta(\$mensaje)
- borra(\$idmensaje)
- setFile(\$file)
- Constructora, getters y setters.

- Application

Es la clase que se encarga de conectarse a la base de datos correctamente.

Utilizamos el patrón “singleton”. Contiene las funciones:

- getSingleton()
- init(\$bdDatosConexion)
- shutdown()
- compruebaInstanciaIniciada()
- conexionBD()
- Constructora y otros.

- Chat

Contiene la lógica necesaria para realizar operaciones sobre chats. Está formada por las funciones:

- Constructora
- getChats(\$user_id)
- getAllChats()

- Comments

Contiene la lógica necesaria para realizar operaciones sobre los comentarios en los perfiles de usuarios. Contiene las funciones:

- historialComentarios(\$idReceptor)
- historialComentariosPorEmisor(\$idEmisor)

- borrarComentario(\$idComentario)
- crea(\$idEmisor, \$idReceptor, \$texto)
- inserta(\$mensaje)
- Constructora, getters y setters.

- Events

Contiene la lógica necesaria para el calendario y las suscripciones.

Contiene las funciones:

- jsonSerialize()
- creaEvento(\$id_clase, \$fecha_ini, \$hora_ini, \$fecha_fin, \$dia, \$intervalo, \$duracion)
- borraEvento(\$id_evento)
- getEventos(\$id_alumno)
- getHorarios(\$id_clase)
- addToCalendar(\$id_alumno, \$id_fecha_clase)
- getSuscripciones(\$id_alumno)
- Constructora, getters y setters.

- Form

Es la clase base para la gestión de formularios. Es abstracta, por lo que cada formulario tiene que redefinir las funciones de “generaCamposFormulario” y “procesaFormulario”. Contiene las funciones:

- gestiona()
- generaCamposFormulario(\$datosIniciales)
- procesaFormulario(\$datos)
- formularioEnviado(\$params)
- generaFormulario(\$errores, \$datos)
- generalListaErrores(\$errores)
- Constructora

- FormularioLogin

Extiende a la clase Form y procesa la información de los formularios del Login.

- FormularioRegistro

Extiende la case Form y procesa la información recibida de los formularios de SignIn.

- FormularioEditProfiles

Extiende la case Form y procesa la información necesaria para editar el perfil de un usuario.

- FormularioMessages

Extiende la clase Form y genera el campo para introducir el texto de un nuevo mensaje y se encarga de realizar las comprobaciones e inserción del mensaje.

- FormularioBusqueda

Extiende la clase Form y tiene los campos necesarios para realizar búsquedas sobre clases.

- FormularioEnviarPuntuacion

Extiende la clase Form y los campos necesarios para enviar la puntuación sobre un usuario, y la inserta.

- FormularioComentarios

Extiende la clase Form y genera el campo para introducir el texto de un nuevo comentario en los usuarios, y se encarga de realizar las comprobaciones e inserción del comentario en el perfil.

- FormularioAd

Extiende la clase Form y procesa la información de los formularios de crear clases.

- FormularioPayment

Extiende la clase Form y procesa la información de los formularios de Payment.

Estructura de la base de datos

Vamos a proceder a detallar la estructura de la base de datos. Iremos tabla a tabla explicando cada atributo, relaciones con otras tablas y qué scripts consultan o modifican las mismas.

Usuario

En esta tabla se recogerán los datos principales de todos los usuarios de nuestra aplicación.

id

Identificador único de todos los usuarios de la base de datos.

username

Nombre único con el que se identificarán los usuarios al entrar a la aplicación. No puede ser nulo.

pass

Contraseña con la que accederán los usuarios a la aplicación. No puede ser nulo y debe tener una longitud mínima de 15 caracteres. No puede ser nulo.

email

Dirección de correo electrónico que será la vía de comunicación de ParticulAres con los usuarios. Por ejemplo para restablecer la contraseña, avisos de pagos, etc. No puede ser nulo.

movil

Teléfono móvil del usuario, es opcional.

nombre

Nombre del usuario. No puede ser nulo.

apellidos

Apellidos del usuario. No puede ser nulo.

edad

Edad del usuario. Es opcional.

es_admin

Booleano que señala si tiene permisos de administrador o no.

es_profesor

Campo para indicar si el usuario es además profesor o no.

descripción

Breve descripción pública del usuario para que los demás tengan un poco más de información sobre él. Es opcional.

puntuacion_media

Puntuación media entre 1 y 5 de todas las valoraciones que otros usuarios le han dado.

num_puntuaciones

Número de valoraciones que otros usuarios le han dado a este.

Scripts de modificación y consulta

User

Clases

En esta tabla se irán guardando detalladamente las clases que los profesores publiquen. Todo el que visite la página podrá ver las clases publicadas más recientes.

Id

Identificador único de las clases.

id_profesor

Id del profesor que imparte la clase. Referencia de la tabla Usuario. No puede ser nulo.

materia

Materia que se imparte en la clase. No puede ser nulo.

nivel

Nivel de la clase. Primaria, Secundaria, Universidad, etc. No puede ser nulo.

idioma

Idioma en el que se imparte la clase. No puede ser nulo.

zona

Zona más o menos aproximada (según las preferencias del profesor) donde se imparte la clase. Es opcional.

descripcion

Descripcion que el usuario desee sobre su clase, o cualquier información que desee añadir.

precio

Precio por hora de la clase. A criterio del profesor. No puede ser nulo.

fecha

Fecha de creación de la clase

Scripts de modificación y consulta

Advertisement

Fecha clases

En esta tabla se guardarán las fechas y las horas de una determinada clase.

id_clase

Identificador de la clase de la cual se va a registrar un horario. Referencia de la tabla Clases.

No puede ser nulo.

fecha_ini

Fecha de inicio de la clase

hora_ini

Hora de inicio de la clase.

fecha_fin

Fechas en las que se terminan las clases

Dia

Dia de la semana en la que se realiza la clase

Intervalo

Número de días entre clase y clase

Duracion

Duración de cada clase

Scripts de modificación y consulta

Events

Suscripciones

En esta tabla se guardarán las suscripciones de un alumno a una determinada clase publicada por un profesor.

id_alumno

Identificador del alumno.

id_fecha_clase

Identificador del horario de la clase. Referencia de la tabla fecha_clases.

Scripts de modificación y consulta

Events

Comentarios usuarios

Esta tabla es simiular a la anterior pero con los usuarios en si. Se guardarán todos los comentarios generales que reciban cada usuario.

id

Identificador del mensaje. No puede ser nulo.

id_emisor

Identificador del emisor del mensaje. Referencia de la tabla Usuarios. No puede ser nulo.

id_receptor

Identificador del receptor del mensaje. Referencia de la tabla Usuarios. No puede ser nulo.

texto

Texto en si del comentario.

Scripts de modificación y consulta

Comments

Profesor_datos

En esta tabla se guardan los datos sensibles de los usuarios que tienen cuenta de profesor.

id

Identificador del profesor. Referencia de la tabla Usuario. No puede ser nulo.

IBAN

Número de cuenta bancaria para realizar los pagos.

Scripts de modificación y consulta

User

Mensajes

En esta tabla se guardarán los mensajes privados entre dos usuarios.

id

Identificador del mensaje. No puede ser nulo.

id_emisor

Identificador del emisor del mensaje. Referencia de la tabla Usuarios. No puede ser nulo.

id_receptor

Identificador del receptor del mensaje. Referencia de la tabla Usuarios. No puede ser nulo.

texto

Texto en si del mensaje.

fecha

Fecha en la que se envía el mensaje. No puede ser nulo.

Scripts de modificación y consulta

Message

Puntuaciones

Esta tabla se utiliza para mantener si un usuario ya ha puntuado a otro ya, y no pueda puntuar 2 veces.

id_puntua

id del usuario que puntúa.

id_puntuado

id del usuario al que se puntúa

Scripts de modificación y consulta

User

Instrucciones de instalación

Solo hace falta añadir todos los archivos a htdocs e importar la base de datos dentro de la carpeta mysql. Las credenciales con los que accede son: Host: "particulares-bbdd", usuario: "root", contraseña: "". De todas maneras, esto se puede cambiar rápidamente en el archivo include/config.php.