**Nombre de la tabla:** usuarios

**Enlace: http://212.142.193.210:16001/mikels/ejercicio\_include/registro.php**

Proceso de registro:

registro.php

Empezando desde la primera instrucción en el main, comprobamos si existe la variable “registro” que viene de la petición de la función de “registro2” (inc\_config.php->registro2). Si la variable es false hacemos un echo diciendo que el nombre o el correo están en uso; si no, es que la sentencia sql del insert no se ha completado.

Si la función de petición es POST: Guardamos en variables lo que nos llegue del formulario mediante POST. El nombre y el apellido lo filtramos en la función filter(inc\_config.php->filter), el email, contraseña, login/usuario, fecha de nacimiento y teléfono los filtramos y lo que devuelva la función “filter” lo pasamos como argumentos en la función “preg\_match” para cada campo del formulario. La función “preg\_match” devuelve true si el string proporcionado coincide con el “pattern”. Si alguno devuelve false guardamos en variables los datos proporcionados para mostrarlos en el formulario, damos valor true a la variable “$verformulario” y a la variable ”$mensaje” le damos un string diciendo que algún dato no está bien introducido; y si ninguno da false damos el valor false a “$verformulario”.

Si “$verformulario” es true vemos el formulario con las variables del formulario si tienen algún valor; En caso contrario llamamos a la función “registro2” pasandole los datos que nos han introducido como argumentos.

inc\_config.php

En la función “registro2” empezamos con un try catch capturando la excepciones de “PDO”. Hacemos la conexión a la base de datos dando la información de la base de datos al método “mysqli\_connect” de la clase “mysqli”. Invocamos a la función “bin2hex” para que nos devuelva un string hexadecimal de lo que le pasamos mediante la función “random\_bytes” para crear el token para la confirmación de usuario. A continuación guardamos la fecha del servidor. Creamos las consultas sql para seleccionar los usuarios con el correo y con el login introducidos y las pasamos en el método “query” del objeto de la clase “mysqli”, después guardamos el resultado de las consultas en las variables “$result” y “$result2”. Si alguno de los dos devuelve alguna línea le redirigimos a “registro.php” diciendo que el correo o el usuario están en uso, si no procedemos a la inserción.

Guardamos el insert en una variable y si al llamar al método “query” con el insert nos devuelve true le redirigimos a “registro\_enviado.php” con el correo proporcionado y el token que hemos creado. Si no, le llevamos a “registro.php” diciendo que ha ocurrido un error en la inserción.

registro\_enviado.php

Guardamos en “$mensaje” una cadena de texto de un html que será el email de confirmación con un link a “registro\_confirmar.php” pasando el token por GET.

Si existe la variable email invocamos la función “mail\_phpMailer” (inc\_config.php->mail\_phpMailer) dandole por argumento el mensaje anteriormente creado y el email que nos llega por GET. Si el resultado de esa llamada es true informamos que se ha enviado un correo electrónico.

registro\_confirmar.php

El link que le enviamos al usuario por email le redireccionará a esta página. Creamos la conexión a la base de datos como hemos hecho anteriormente. A continuación ejecutamos un select del usuario que tenga el token que nos llega del enlace desde el correo electrónico, si devuelve alguna línea guardamos en la variable “$valor” el resultado que nos devuelve “fetch\_assoc” que llamamos sobre las líneas que nos devuelve el método “query”. El método “fetch\_assoc” nos devuelve un array asociativo, guardamos el valor del id para después ejecutar un Update sobre la tabla usuarios para cambiar el estado del usuario con ese id.