



Programación Avanzada
Ingeniería Informática en Sistemas de Información - Curso 2019/2020
ENSEÑANZAS PRÁCTICAS Y DE DESARROLLO
Prueba de evaluación 2ª convocatoria: PHP (Parte I de II)

Importante: El resultado de esta prueba consistirá en un proyecto de NetBeans que será comprimido en formato ZIP conteniendo la aplicación. Emplee las credenciales por defecto de XAMPP para la conexión con la base de datos. El nombre del fichero tendrá un formato específico dictado por el nombre de cada alumno. Por ejemplo, para un alumno llamado "José María Núñez Pérez" el fichero se nombrará como NunyezPerezJM.zip. Obsérvese que las tildes son ignoradas y las eñes sustituidas. No se aceptará ningún envío que no cumpla con las anteriores especificaciones de formato y nombrado. **Las rutas de los ficheros empleados serán relativas, a fin de que las resoluciones a los ejercicios y problemas puedan ser examinadas en cualquier equipo. Cualquier entrega que no cumpla las reglas de nombrado, el formato de compresión del archivo o el contenido de los archivos del mismo, será penalizada con 2 puntos sobre 10 por cada incumplimiento. Pasado el límite de entrega se aceptará el envío del trabajo, con una penalización de 2 puntos sobre 10 de la calificación por cada minuto o fracción de retraso a partir de tercer minuto. No es necesario que incluya el script de la base de datos en el envío.**

Objetivos

- Demostrar los conocimientos adquiridos en el desarrollo de aplicaciones empleando PHP.

Descripción del ejercicio propuesto

Dada la crisis sanitaria que está transcurriendo en estos momentos, se ha decidido desarrollar un sistema informático que permita obtener información sobre las pruebas de COVID-19 realizadas a pacientes de toda la ciudad, de forma que se pueda obtener información acerca de cuántas pruebas se han realizado y los resultados de las mismas. Para ello, se debe realizar una aplicación web que use PHP como lenguaje de programación y MariaDB como base de datos.

Un usuario perteneciente a un hospital determinado puede logarse en el sistema y visualizar todas las pruebas de COVID-19 realizadas en su hospital, mostrándose el nombre del paciente, su DNI, la localidad a la que pertenece, la fecha en la que se le realizó la prueba y el resultado, siendo posible los resultados positivos o negativos. Además, el usuario identificado podrá registrar nuevas pruebas realizadas en el hospital.

Desde el punto de vista técnico, cuando proceda, las entradas del usuario deben ser validadas y saneadas para evitar inyecciones SQL, así como cumplir con determinados requisitos y formatos de la aplicación. Se utilizará una variable de sesión para controlar que el usuario está autenticado en todas las páginas de la aplicación web. Ser permitirá cerrar la sesión desde cualquiera de ellas. Se guardará una *cookie* que contendrá el nombre de usuario que se ha autenticado para que éste aparezca relleno automáticamente en la página de login en próximas ocasiones.

Encontrará en el material adicional la base de datos exportada (con datos de ejemplo), así como las vistas que se han de generar, desde la que podrá extraer el código HTML a emplear (use dicho código para no perder tiempo en esto).

Actividades a realizar

Para desarrollar el sistema, complete las siguientes tareas:

1. **[2 puntos]** Implementar la funcionalidad de login partiendo de la vista *index.html* proporcionada en el material adicional. El envío de este formulario deberá procesarse en la misma página, de forma que si las credenciales del usuario son correctas, se establecerá una variable de sesión que indique si el usuario está identificado o no (utilice un valor booleano) y se muestre una página llamada *gestionCitas.php*, cuyo contenido se describirá mas adelante (**0,5 puntos**). Si las credenciales de usuario no son correctas, se deberá regresar a la página de login e informar al usuario mediante un mensaje (**0,5 puntos**). Además, tenga en cuenta que si accede a la página de login con un usuario ya identificado, se deberá redirigir automáticamente a la página *gestionCitas.php* (**0,5 puntos**). Por último, establezca una *cookie* con una validez de 15 días que se renueve cada vez que el usuario intente iniciar sesión de tal forma que cuando se acceda a la página de login (o bien por error en las credenciales o por hacer logout en la aplicación), aparezca el nombre de usuario prerrelleno (**0,5 puntos**).



2. **[4,75 puntos]** Implemente la página `gestionCitas.php` (basada en `gestionCitas.html` proporcionada en el material adicional). Esta página está compuesta por:
 1. Un mensaje de bienvenida, que variará en función del usuario logado y el nombre del hospital al que pertenece (**0,5 puntos**).
 2. Un botón de logout que al pulsar sobre él, elimine la información de usuario almacenada en la sesión y regrese a la página de inicio (**0,25 puntos**).
 3. Una tabla debidamente formateada que muestre todas las pruebas realizadas en el hospital al que pertenece el usuario logado. Al obtener la información para mostrar la tabla, tenga en cuenta las relaciones existentes en la base de datos, por lo que deberá realizar las operaciones necesarias para obtener el nombre (**1,75 puntos**).
 4. A la derecha de cada registro de la tabla deberá aparecer un botón que permita eliminar la prueba sobre la que se haga click (**1 punto**). Una vez que se elimine la prueba, deberá mostrarse de nuevo la página `gestionCitas.php` con la tabla actualizada (**0,25 puntos**).
 5. Un botón llamado "registrar prueba", que al pulsar sobre él vaya a la página `altaPrueba.php`, cuyo contenido se describe más adelante (**0,25 puntos**).
 6. Asegúrese de que no se puede acceder a la página `gestionCitas.php` ni `altaPrueba.php` si el usuario no está identificado previamente (por ejemplo, escribiendo la ruta completa en el navegador) (**0,75 puntos**).
3. **[1 punto]** Implemente la página `altaPrueba.php` de tal forma que se muestre un formulario que permita dar de alta una nueva prueba. Para ello, se deberá implementar un formulario (similar al que puede encontrar en el material adicional), de tal forma que al enviar el formulario se procese en la misma página PHP. Este procesamiento consistirá en comprobar y validar los datos del formulario (descrito en el siguiente apartado), así como registrar (en caso de que toda la validación sea satisfactoria) una nueva cita en la base de datos. Tenga en cuenta que al registrar una nueva cita, el identificador del hospital debe estar fijado al mismo hospital al que pertenece la persona que está dando de alta la nueva prueba.
4. **[2,25 puntos]** Asegúrese de sanear y validar todos los campos de formulario de tal forma que:
 1. En el formulario de login:
 1. Los campos de usuario y contraseña deben existir y estar rellenos (**0,25 puntos**).
 2. Campo de usuario y contraseña no pueden contener los caracteres `# : ; ! " *` (**0,25 puntos**).
 3. El campo contraseña contiene al menos un número (**0,25 puntos**).
 2. En el formulario de alta prueba:
 1. El campo `nombre_paciente`, `dni_paciente`, `localidad_paciente`, `fecha_prueba`, `resultado_prueba` y `hospital_asociado` deben existir y estar rellenos (**0,25 puntos**).
 2. El campo `nombre_paciente` solamente debe contener caracteres alfabéticos (**0,25 puntos**).
 3. El campo `dni_paciente` deberá estar compuesto por 8 dígitos y una letra mayúscula (**0,25 puntos**).
 4. El campo `localidad_paciente` deberá comenzar con una letra mayúscula y deberá contener al menos 5 caracteres (**0,25 puntos**).
 5. El campo `fecha_prueba` deberá indicarse con el formato DD-MM-YYYY. Puede utilizar para ello la siguiente expresión regular: (**0,25 puntos**)

`^([0-2][0-9]|(3)[0-1])(\(((0)[0-9])|((1)[0-2]))(\-)\d{4}$`

Tenga en cuenta que para registrar un campo tipo Date en la BD necesitará formatear dicha salida a Y-m-d, pudiendo utilizar la función `date` para crear la nueva fecha formateada, por ejemplo:

```
date('Y-m-d', strtotime($fecha));
```

6. El resultado de la prueba deberá ser 0 en caso negativo y 1 en caso afirmativo (**0,25 puntos**).



Material suministrado

Como material adicional dispondrá del fichero "Material Adicional.zip" que contendrá:

- Base de datos COVID19.sql: Script de creación de la base de datos. Existen hospitales, usuarios y pruebas ya creadas.
- Carpeta Vistas, donde se encuentran las diferentes páginas HTML que podrá usar de base para el desarrollo de los formularios.

Datos de la prueba

Autor del documento: José F. Torres (Junio 2020).

Revisiones del documento

1. Carlos D. Barranco (Junio 2020)