Nombre y Apellido: Delfina Morello

11086 - Programación en Ambiente Web - UNLU
Primer Parcial 2020

Entrega: archivo PDF por mail a la dirección paw@unlu.edu.ar antes del viernes 1 de mayo a las 23.59.59, es decir cuenta con 36hs para realizarlo. Además del envío por mail del PDF se solicita que suban también dicho archivo PDF a un repositorio propio y envíen dicha dirección en mail aparte o por whatsapp/telegram para garantizar recepción a tiempo utilizando dos vías independientes y con timestamp.
Metodología: el examen es individual y si bien puede utilizar libros e Internet, el examen debe ser autocontenido y con respuestas «propias», no con URLs hacia material externo. Puede incluir esquemas o lo que considere necesario para ilustrar sus respuestas.
Nota Importante: En ninguno de los 10 puntos se solicita código. Responda cada punto considerando el escenario más real/auténtico que pueda imaginar y explicite cada una de sus asunciones.

Imagine una aplicación web "portal de noticias" y responda las siguientes consignas:

1. ¿Por qué las sesiones pueden guardar mucha más información que las cookies? ¿Qué almacenaría para esta app en cookies y/o sesiones?

Una sesión nos permite guardar información de los usuarios cuando estos cambian de página dentro de un mismo sitio web. Los datos asociados a ella se almacenan del lado servidor.
Las sesiones suelen ir acompañadas de cookies, que es un archivo pequeño que se almacena del lado del cliente y que forman parte de la cabecera HTTP (se setean en los headers de la petición).

Las cookies pueden guardar cualquier información que el servidor considere importante.
Debido a que su tamaño es limitado, cuando se tiene mucha información asociada a un usuario, lo que podemos hacer es, guardar toda esa información en el servidor y en la cookie almacenar solo el id para permitirle al servidor saber a qué sesión de usuario hace referencia.

Las sesiones pueden guardar mucha más información ya que al encontrarse del lado del servidor no se ve tan limitadas en cuanto al espacio de almacenamiento, además de encontrarse menos vulnerable a ataques.

En las cookies almacenaría el id de la sesión, la última vez(fecha) que el usuario visitó el portal, los enlaces en los cuales entró, el contenido en los que está interesado, el idioma de las consultas.

2. ¿Qué ventajas ofrece el uso de Virtualhost en el contexto de servidores Web (en gral y en particular para esta app)?

Como ventaja general el uso de Virtualhost permite ejecutar más de un sitio web en un mismo servidor físico. Pueden estar basados en direcciones IP o en nombres.

Ventajas en particular para esta app:

-Mayor tolerancia a fallos: podemos tener réplicas de la máquina virtual y en caso de que se produzca algún fallo poder reemplazarla rápidamente.

-Más escalabilidad, fácil aumento de recursos: permite el agregado de nuevos contenidos con facilidad, pero hay que tener en cuenta que la máquina “real” en la que se está ejecutando dicho virtualhost tiene que estar preparada para soportar esa cantidad de contenido.

3. Defina con sus palabras la diferencia principal entre contenido estático y dinámico.

Contenido estático: generalmente es un archivo html y que ya existe en el servidor, el cual lo presenta “tal como está” debido a que su contenido es siempre el mismo y no varía en el tiempo.

Contenido dinámico: Este tipo de contenido se genera en tiempo de ejecución del lado del servidor a partir de la información ingresada por el usuario y apunta a personalizar la información mostrada, es decir que su contenido puede variar para cada usuario.

4. ¿Cómo aplicaría el modelo MVC para el diseño de esta app?. No necesita escribir código alguno, sino argumentar conceptualmente como separaría la lógica de la app en estos tres elementos.

Modelo: Define las clases y los métodos que comunican con la base de datos, es el responsable de acceder a la capa de almacenamiento de datos.
También es aquí donde se realizan las validaciones de los datos ingresados por el usuario.

Controlador: Es el responsable de recibir las peticiones de los usuarios y solicitarle los datos al modelo. Es el que gestiona la información que circula entre la vista y el modelo.

Vista: Permite la interacción del usuario con el sistema. Es el responsable de recibir los datos del modelo y presentarlos al usuario, ej vista de solo algunas noticias del portal.

5. a) ¿Por qué es posible afirmar que PDO mejora la seguridad en la capa de base de datos de una app PHP?

PDO nos brinda una capa de abstracción de acceso a los datos, sin importar la base de datos que se esté utilizando.

Es posible afirmar que mejora la seguridad en la capa de base de datos de una app porque está compuesto por prepare statement (declaraciones preparadas), que son como un tipo de plantilla que ofrecen una defensa muy importante para evitar la inyección de código SQL.

b) ¿Qué otras cuestiones debemos tener en cuenta en la capa de base de datos en el sentido de la seguridad?

-Nunca confiar en los datos que recibo del usuario, por lo cual debo validar toda entrada que provenga de él.
-Aplicar SALT a la información sensible, ej password, antes de almacenarla en la base de datos.

6. La app muestra signos de "envejecimiento" en cuanto al diseño, tanto usuarios finales como redactores del portal lo informan a diario. ¿Qué ideas se le ocurren al respecto?

El diseño se podría mejorar si le incorporamos:

-Aquellas prácticas que otros sitios web realizan, como por ej la ubicación de los elementos, información de contacto (que en la mayoría de los sitios se encuentra al final de la página).

-Que sea responsive, es decir, que se adapte a diferentes dispositivos (tablets, celulares, etc) y a sus respectivas resoluciones.

-Que su contenido sea actual, preciso, necesario y varíe regularmente.

-Que sea navegable, es decir, que le permita al usuario saber en qué parte del sitio está ubicado.

-Que sea consistente con su estilo (ej color de letra, tipo de letra, color fondo) en todo el sitio web.

-Diseñar pensando en la estructura que los navegadores utilizan para posicionar una página web.

7. Se le informa al equipo de desarrollo que las nuevas funcionalidades están repercutiendo negativamente en la performance de esta app web en el ambiente productivo, no así en el ambiente de testing (QA).
DevOps informa que existe últimamente mucha carga a nivel de bases de datos. ¿Qué se le ocurre hacer en su rol de Desarrollador Web?

Para disminuir la carga en la base de datos podría controlar el acceso de los diferentes usuarios al sitio en base a los roles que cumplan en el mismo y también permitirles que realicen consultas solo sobre ciertas tablas de la base de datos.

Por ejemplo, a un usuario visitante de la página solo le otorgaría permisos de lectura sobre el portal y el acceso a tablas de datos específicas, mientras que a los administradores del mismo tendrían permisos de lectura y escritura sobre todas las tablas.

Otra manera de evitar la sobrecarga en la base, es implementar una base de datos distribuida en donde podemos repartir la carga mejorando así la velocidad de acceso a los datos y la tolerancia a fallos.

También se podría considerar el uso de sesiones para ahorrar la consulta de datos de usuario (ej nombre, mail) que no cambian a lo largo de la estadía dentro del sitio para no sobrecargar aún más a la base de datos.

8. Imagine ahora que el "portal de noticias" debe considerar tener un "paywall" (ciertos contenidos se vuelven pagos) y por ende almacenará tarjetas de débito / crédito de los clientes.
a) ¿Cuáles son las implicancias de seguridad de esta nueva funcionalidad?

-Utilizar https para que el envío de contraseñas sea más seguro.
-Encriptar y guardar la información del lado del servidor.
-Tener un backup de los datos almacenados en el servidor.
-No almacenar información sensible en texto plano.

b) ¿Cómo implementaría algún límite sobre la cantidad de noticias que puede ver un usuario que no paga, e.g. puede ver sólo 10 artículos por mes calendario?

El límite sobre la cantidad de noticias que puede ver un usuario que no paga lo podría llevar de la siguiente manera:

Llevaría un contador en una cookie sobre la cantidad de noticias que el usuario va visitando, y cuando llegue al límite (10 visitas), mostrarle un pop-up informándole que para seguir viendo más debe pagar o también podría agregarle publicidades dificultándole la visión de las noticias. Este usuario no podrá seguir viendo artículos hasta que pague o bien hasta que la cookie expire, un mes después de haber sido seteada, volviendo a reiniciar el contador de visitas.

9. Se requiere implementar un buscador de noticias dentro de esta app. Explique qué responsabilidades tiene cada capa de la aplicación en la resolución de la búsqueda. ¿Qué método HTTP le parece el más adecuado para implementar esto? ¿Qué problemas observa?

El método más adecuado para la implementación de la búsqueda es el GET, ya que es utilizado para realizar consultas al servidor. Este método cachea la petición para evitar volver a realizar la consulta al servidor nuevamente durante un tiempo determinado.

Modelo: Contiene los métodos necesarios para realizar la búsqueda y obtener los resultados de las mismas. Ej buscar palabras parecidas, enviar resultados encontrados, recuperar datos.

Controlador: le pide al modelo que le devuelva las noticias que coincidan con la petición del cliente, enviar a la vista una cierta cantidad de resultados encontrados.

Vista: Ingresa datos a buscar, muestra los resultados que obtuvo de dicha búsqueda.

El problema que encuentro en la búsqueda de noticias es que, se encuentra vulnerable a inyección sql que podría ingresar el usuario a través del input de datos del buscador.

10. Se requiere que la experiencia del sitio sea uniforme en versiones de Chrome/Firefox/IE de hasta 3 años atrás. ¿Cómo puede cumplir con dicho requisito? ¿Qué estrategias adoptaría desde el punto de vista del diseño e implementación?

Para cumplir con dicho requisito podríamos utilizar elementos que sean compatibles en distintos navegadores y estandarizar la apariencia de los sitios.

Las estrategias que adoptaría son:

-Hojas de estilo específicas: Cada navegador debe tener su propia hoja de estilo, debido a que poseen diferentes características, permitiéndonos realizar tareas de mantenimiento y actualización de contenido fácilmente.

-Validación: Escribir el código css y html siguiendo estándares para que muestro sitio web sea compatible con el mayor número de navegadores posible.

-Resetear hojas de estilo: El objetivo de esta estrategia es reducir las variadas interpretaciones de los diferentes navegadores, para lo cual se eliminarán todos los estilos que tengan aplicados los navegadores por defecto.

-Usar medidas relativas, para que pueda adaptarse a todo tipo de dispositivos y pantallas.