

| | | | |
|-------------------|--|---------------------|------------|
| Estudios | Grado Superior en Desarrollo de Aplicaciones Multiplataforma | | |
| Asignatura | Desarrollo de Interfaces | | |
| Profesor | Santiago Moreno Morales | | |
| Evaluación | 2ª Web | Fecha limite | 15/03/2019 |
| Alumno | | DNI | |

Practica 2

Entrega

- La entrega de todos los archivos se realizara mediante la intranet de la escuela. **Solo y exclusivamente por este medio.**
- Solo se puede subir **un archivo por alumno y practica**, así que asegúrate que esta todo correcto varias veces, antes de realizar la carga de dicho archivo.
- Asegúrate cual es la fecha y hora limite para realizar la carga del archivo, y da un margen de tiempo para evitar problemas de ultima hora.
- Asegúrate que todo lo relacionado con la web se instala correctamente, haz varias pruebas de instalación en diferentes equipos.
- La prueba de validación se realizara primero instalando la web, BD, y todo lo necesario para el correcto funcionamiento. Y luego se realizara dicha prueba.
- Si por algún motivo no se puede instalar correctamente la web(falta de algún archivo, algún formato equivocado, etc..) **no se procederá a la corrección de dicha practica**, con lo que ello conlleva.

Puntuación

La valoración de esta practica dependerá de la calidad, tanto de la documentación, como de la programación, y del diseño:

- En la parte de la documentación se valorara:
 - Tiene que ser lo más correcta y completa posible, pero sin abusar del texto de relleno.
 - No se puntuara por páginas rellenas, si no por calidad de la información contenida en ella.
 - Que este bien estructurada y clara.
 - Que aporte todo lo necesario, para la buena comprensión de lo que allí se explica.
- En la parte de la codificación, se tendrá en cuenta:
 - La forma de utilizar la orientación a objetos, los métodos, los objetos, las herencias, etc...
 - La agrupación y organización mediante archivos, carpetas, clases, etc... del código.
 - Penalizara, la duplicidad de código, y la mala utilización de la orientación a objetos.

- En la parte de diseño:
 - Se tendrá muy en cuenta, el UX, y que la web sea usable y comprensible para el publico objetivo.
 - Se puntuara mejor, cuanto mas atractiva y cuidada sea la estética.
 - Que se utilice en mayor medida, las ultimas versiones de los lenguajes utilizados, como por ejemplo html5 en vez de html4, etc... teniendo en cuenta, la compatibilidad con las ultimas versiones, **en la mayoría de los navegadores.**

La documentación del programa

- 1 Debes de realizar uno o varios documentos, donde realizar la documentación del proyecto.
- 2 Debes incluir en la documentación, todos los datos que son relevantes:
 - 2.1 Explicación de la web, y su funcionalidad.
 - 2.2 Etapas y tareas.
 - 2.3 Mapa/árbol web.
 - 2.4 Bocetos de las interfaces.
 - 2.5 Explicación de usabilidad y estructura de las interfaces.
 - 2.6 Explicación de las funciones (código) mas relevantes.
 - 2.7 Esquema de la base de datos.
 - 2.8 Pruebas realizadas.
 - 2.9 Problemas y soluciones.
- 3 Tendrás que ir evolucionando la documentación, conforme evolucione el proyecto, de tal manera que:
 - 3.1 Realizar la definición y evolución de las tareas en las etapas antes documentadas.
 - 3.2 Tendrás que documentar tus fuentes a la hora de recopilar código, diseños y demás componentes del proyecto.

Lo que hay que desarrollar.

- 4 Tendrás que crear una pagina web:
 - 4.1 La temática es de tú elección, si no se te ocurre nada, consúltalo al profesor, y te facilitara una.

4.2 Debe interactuar con los usuarios.

4.3 Estéticamente tiene que ser agradable a la vista y ser usable.

Lo que ha de tener la pagina.

5 Tiene que disponer de las funciones básicas de una web, así como:

5.1 Una pagina de bienvenida o índice para la recepción del usuario.

5.2 Debería de disponer de una galería de imágenes, vídeos, o una zona de comentarios.

5.3 Un formulario de contacto.

5.4 Un menú para desplazarse por la web.

5.5 Manual de usuario o FAQ.

5.6 Unos estilos para dar forma al html.

5.7 Validaciones para los campos de formulario.

6 También debe de disponer de unas funcionalidades avanzadas, como:

6.1 Una base de datos, donde almacenar usuarios para login, o direcciones a imágenes/vídeos, o comentarios, etc...

6.2 Debe de disponer de un login, ya sea para usuarios, o para administración.

6.3 Tiene que haber una parte privada, o tienen que haber unas funcionalidades, donde solo se podrá acceder o utilizar, estando correctamente logueado.

6.4 Debe de tener medidas de seguridad, para evitar los ataques mas comunes.

Items obligatorios

7 Al facilitar los archivos en el momento de la entrega, debe facilitar todo lo necesario para poder instalar y que funcione **todo** en la web.

8 No todo el código tiene que ser tuyo, puedes basarte en ejemplos que puedas encontrar en internet, **aunque no sera dado por bueno el código exacto de dicho ejemplo, o con pequeñas modificaciones para hacerlo pasar por vuestro**, de ahí la necesidad de documentar las fases del desarrollo y las fuentes de las que se recopila.

9 Tienes que utilizar:

9.1 Html5.

9.2 Css3.

9.3 JavaScript.

9.4 Php.

9.5 MySQL..

9.6 Y otras tecnologías que consideres oportunas(jquery, ajax, etc..).

10 Tiene que haber una base de datos relacional **MySQL**, donde se ejecutaran las consultas y modificaciones necesarias. Se tiene que realizar de forma local y se tendrá que adjuntar dicha BD junto al proyecto incluyendo algunos datos con las pruebas realizadas, los usuarios y contraseñas que en ella hubiesen.

11 La interfaz gráfica deberá de contener al menos los siguientes componentes:

11.1 Todos los botones, inputs, etc... Necesarios para que se pueda interactuar de forma correcta.

11.2 Una zona de menú desde donde desplazarse por la web.

11.3 Una parte de formulario con todos los campos necesarios para que este completo, y que funcione correctamente.

11.4 Una galería con algún tipo de efecto al interactuar con ella.

11.5 Manual de usuario o FAQ.

11.6 Partes privadas donde solo acceder si se esta correctamente logueado.

11.7 Al cargar datos de la BD, no se puede permitir que la pagina se extienda hasta el infinito, en dicho caso se tiene que limitar.

12 En la parte del código:

12.1 El código tiene que tener un buen nivel de documentación (comentarios). Tienen que estar comentados detallando para que sirven los métodos y variables más importantes, así como cuantos valores necesita dicho método, de que tipo son los valores que necesita, si hay sobrecarga de ese método, y que devuelve.

- 12.2 El código tiene que tener una estructura lógica y optimizada(por lo menos en las ultimas versiones de la web), en tantos archivos como sea necesario y lógico, y tiene que estar correctamente tabulado.
- 12.3 Se tienen que realizar métodos independientes, con herencias(si lo permite), duplicando lo mínimo que se pueda el código.

Elementos opcionales para desarrollar

13 Para mejorar la nota de la practica se pueden realizar unos añadidos al proyecto:

- 13.1 Documentar en que versión realices cada una de las etapas o de sus tareas. Pudiendo añadir tareas que no existían en versiones anteriores.
- 13.2 Que el diseño sea responsive o adaptativo.
- 13.3 Que la pagina tenga estilos claramente visibles y aplicados a través de toda la web.
- 13.4 Al cargar datos de la BD, para que no se extienda hasta el infinito, se puede tanto limitar(de forma obligatoria), como crear una paginación(puntuación extra según el trabajo realizado).
- 13.5 Crear componentes gráficos en tiempo de ejecución.
- 13.6 Que se utilicen formularios para rellenar los campos en la base de datos. Como por **ejemplo**, rellenar los items y sus descripciones de una galería, creando dicho formulario para subir la imagen al servidor, guardar su localización en la BD, tener un campo de la descripción, que lo guarde también en la BD, algún campo de posición en la galería, y que luego cargue dicha galería de forma automática, en el orden de posición indicado.

Dudas a: santiagomor.edib@gmail.com