



Proyecto FP Dual INDRA

Alumnos 1º curso DAM/DAW



Índice

1.- Planteamiento del proyecto	2
2.- Tareas a realizar	4
2.1.- Lenguaje de marcas.	4
2.2.- Bases de datos.	4
2.3.- Programación.....	5
2.4.- Sistemas informáticos.....	6
2.5.- Entornos de desarrollo.....	6

1.- Planteamiento del proyecto

Una propuesta de ejercicio para la práctica, por ejemplo, “**Portal para la gestión de eventos sostenibles**”. Los estudiantes deberán desarrollar un pequeño portal web que permita a los usuarios gestionar eventos relacionados con la sostenibilidad, tales como conferencias, talleres, o actividades ecológicas.

Se ha entrevistado al CEO de la empresa y nos ha comentado:

Responsable de la empresa: “A ver, joven, yo no sé mucho de esto de los ordenadores, pero le cuento lo que necesito para el portal que vamos a usar para organizar los eventos. Mire, lo primero es que tenemos estos **eventos**, ¿vale? Como conferencias, talleres o actividades relacionadas con la sostenibilidad. En algún sitio tiene que guardarse toda la información de estos eventos: **el nombre, cuándo van a ser, cuánto duran, dónde serán, esas cosas. ¿Me sigue?**

Luego están los **usuarios**, porque claro, la gente que quiera participar en esos eventos tendrá que registrarse en el portal. Aquí necesitamos guardar sus **nombres**, su **correo electrónico y una contraseña**, supongo. Algo sencillo, que les permita entrar y apuntarse a los eventos.

Ah, **también está la parte de los organizadores. No se olviden de ellos, porque muchas veces no somos nosotros los que organizamos el evento, sino otras empresas o instituciones. Y necesitamos saber quién está detrás de cada evento, con su nombre y su información de contacto, por si hay que llamarlos o mandarles algún correo.**

Por supuesto, si la gente se apunta a los eventos, habrá que llevar la cuenta de eso. No podemos olvidarnos de quién ha decidido participar. Supongo que esto será como hacer una lista con los nombres de los asistentes para cada evento. Eso lo necesitamos.

Consulta

Otra cosa que se me ocurre es que estos eventos pueden ser de distintos **tipos**, ¿no? Algunos son **talleres**, otros son **conferencias...** entonces, **me imagino que estaría bien tener algún tipo de clasificación, para que la gente pueda buscar por categorías, ¿no? Que si uno quiere solo talleres, le salga solo eso.**

Esto se puede hacer con un select

Y finalmente, la parte más importante: las **inscripciones**. Me imagino que esto es lo que conecta todo, ¿verdad? Porque al final, lo que realmente queremos es saber quién se ha inscrito a qué evento. Sería como tener un registro de cada **usuario y el evento** al que se ha apuntado.

Entonces, yo no sé cómo se hace todo esto técnicamente, pero supongo que hay que poner todo esto en algún sitio, bien organizado, para que luego el sistema funcione bien y no haya errores.”

Responsable de la empresa: “A ver, lo que quiero es que todo funcione de forma automática, sin que tengamos que estar encima de todo. Usted me ha mencionado eso de ‘clases’ y ‘objetos’, y aunque no sé exactamente cómo funciona, me imagino que

Eventos
tendra la
clave de
usuarios
como
FK y
también
tendra
la clave
de
organiza
ción
como
FK, Se
creara
una
tabla
con los
participa
ntes
que hay
en cada
evento

cada parte de nuestro sistema tiene que comportarse como si fuera una pieza independiente, ¿no?

Por ejemplo, empecemos por los **eventos**. Cada evento tiene que tener sus propios datos, como el nombre, la fecha, la ubicación donde tendrá lugar el evento (online, dirección ...), etc. Además, necesitamos que los **usuarios** puedan apuntarse al evento. Imagínese que tenemos una ficha para cada evento, y en esa ficha hay un botón que permite registrar a los participantes. También debería haber una opción para cancelar el evento si, por lo que sea, no se puede realizar.

Luego, están los **usuarios**. Cada usuario que se registre en el portal necesita tener su perfil, con su nombre y su correo, claro. Pero además de eso, el sistema debe permitir que el usuario se inscriba en un evento. Supongo que, cuando un usuario se apunta a un evento, se debe guardar esa información, como si el usuario tuviera una lista de eventos a los que está apuntado. También deberían poder cancelar su inscripción si cambian de idea.

Por otro lado, tenemos a los **organizadores**. Estas son las **personas o empresas que crean los eventos**. Entonces, ellos tienen que poder entrar al sistema, crear nuevos eventos y, si es necesario, modificarlos. Me imagino que para cada organizador también habrá una especie de perfil con sus datos y los eventos que ha creado.

Ahora, **aparte de estos tres grupos**, hay más cosas que necesitamos. Por ejemplo, los eventos deben estar clasificados por categorías, como le mencioné antes, y esa clasificación debería ser parte del sistema también. Es decir, cuando alguien cree un evento, el sistema debería preguntarle a qué categoría pertenece ese evento, si es una conferencia, un taller o lo que sea. Esa información tiene que estar ahí para que después los usuarios puedan buscar eventos de una categoría específica.

También tenemos las **inscripciones**, que es lo que conecta a los usuarios con los eventos. Cada vez que un usuario se inscribe a un evento, se crea un vínculo entre ellos. No sé cómo lo hacen ustedes, pero el sistema tiene que asegurarse de que eso esté bien registrado.

Por último, imagino que todas estas piezas tienen que estar bien organizadas y que se comuniquen entre sí. Usted me mencionó algo sobre proteger la información y hacer que ciertas acciones solo puedan realizarlas las personas adecuadas, como los organizadores editando eventos o los usuarios inscribiéndose. También habló de que a veces una cosa puede comportarse como otra, como los eventos online y los presenciales, que aunque son diferentes, deben funcionar de manera similar.

Resumiendo, lo que necesitamos es algo así como **seis tipos de 'cosas'** que gestionen cada parte del portal. No sé cómo lo harán ustedes con eso de las clases, pero necesitamos que cada cosa haga lo suyo: **los eventos, los usuarios, los organizadores, las categorías, las inscripciones y las ubicaciones**. Todo debe funcionar bien y estar conectado, pero sin que tengamos que hacer todo manualmente.”

Se realizará un despliegue de la página web desarrollada con anterioridad.

Esto se realiza con getter y setter

Para ello, deberemos realizar una instalación de Windows 10 en una máquina virtual. Tendremos que revisar los requisitos hardware necesarios para su instalación. Es conveniente instalar la edición PRO.

Deberemos también crear un servidor web a nivel local en el sistema operativo Windows 10. Para ello, utilizaremos Xampp. La captura final que deberá aparecer como colofón final de esta parte deberá ser mostrar la página web desarrollada en apartados anteriores.

2.- Tareas a realizar

2.1.- Lenguaje de marcas.

HTML y CSS: Los alumnos deben desarrollar una página web con tres secciones:

Página principal: Presentación del portal y un carrusel de eventos destacados.

Página de listado de eventos: Una tabla/listado con todos los eventos, incluyendo el nombre, la fecha, y una breve descripción.

Página de detalle del evento: Cada evento en el listado lleva a una página de detalle con información más completa (ubicación, organizadores, imagen del evento, etc.).

Requisitos:

Uso adecuado de etiquetas semánticas HTML5.

Diseño en CSS.

Todas las páginas web serán estáticas.

2.2.- Bases de datos.

Se deberá de realizar el diagrama entidad-relación con una herramienta software (a elegir entre Miro, Dia, Yed, Draw.io ...) acompañado de la tabla de descripción de entidades que se adjunta a continuación y el diseño físico de la base de datos en un script SQL con al menos 6 tablas.

Se debe aportar información detallada de cada una de las tablas, rellenando para cada una de ellas una tabla como la siguiente:

6

ENTIDAD: _____									
Atributo	Descripción	Dominio	Tipo/longitud			Opcional?		Univaluado?	
			Num	Alf	Otro	S	N	S	N
Observaciones y restricciones de la entidad:									

2.3.-Programación

El objetivo principal de este ejercicio es que apliques los conceptos fundamentales de la programación que estás aprendiendo en tu centro. Implementar una solución que siga los principios de la **programación estructurada**, el **uso modular** para organizar tu código, y la **manipulación de vectores (arreglos)** para gestionar los datos de manera eficiente.

Lenguaje de programación:

El desarrollo se realizará en el lenguaje que estás utilizando en tu formación actual.

Programación estructurada:

Se espera que sigas el paradigma de la programación estructurada, utilizando bloques lógicos y evitando estructuras complejas o desorganizadas.

Modularidad:

La solución deberá dividirse en **funciones** o **procedimientos** que aborden tareas específicas del programa.

Uso de vectores:

Los vectores o arreglos deben emplearse para almacenar y manipular conjuntos de datos.

2.4.- Sistemas informáticos

Realizar una instalación de Windows 10 en una máquina virtual.

Crear un servidor web a nivel local en el sistema operativo Windows 10. Para ello, utilizaremos Xampp. Deben realizarse una serie de capturas para realizar unas evidencias del proceso tanto de instalación como del resultado de la parte desplegada con el Xampp.

Propuesta de evidencias sugeridas:

- Captura de la configuración de la máquina virtual.
- Captura de las pantallas de instalación de Windows 10 (selección de idioma, particiones y configuración inicial).

2.5.- Entornos de desarrollo

Una vez finalizada la implementación del ejercicio, se requiere que todo el código desarrollado, junto con la documentación complementaria, sea **subido a un repositorio personal en GitHub**. Este paso es fundamental para practicar la gestión de proyectos en un sistema de control de versiones y para consolidar las habilidades relacionadas con el uso de Git.

1. Creación del repositorio en GitHub
2. Uso de Ramas. Deberás estructurar el desarrollo del proyecto utilizando **ramas** en Git para reflejar las diferentes etapas del trabajo
3. Subir el grafo de ramas