

Memoria de Decisiones del Proyecto

Diseño y Planificación (La base de todo)



Antes de comenzar con el código, durante la fase del **plan de negocio**, se ha definido toda la **guía de estilos**, incluyendo colores, tipografías y logo. No queríamos que el diseño fuera improvisado, así que se tomaron decisiones basándonos en la psicología del color, estudio del cliente ideal y un plan de marketing con estudio del mercado en ese nicho en concreto.

Creamos unos **wireframes** a mano para visualizar la estructura. Luego, lo trasladamos a **Figma** para crear un **mockup** de alta fidelidad. De este modo, al llegar a la programación, ya sabía exactamente cómo debería lucir la web.

Backend y Base de Datos (PHP/MySQL): Se ha configurado la lógica siguiendo el **modelo MVC** que aprendimos en clase de **DWES**. Esto ayudó a no mezclar el HTML con el PHP, manteniendo todo más organizado.

Seguridad: Para el login, no guardar las contraseñas directamente; se utilizó **password_hash**. También se ha añadido **htmlspecialchars** en los inputs para evitar problemas con ataques XSS. BD y

Errores: Se conectó con **PDO** porque es más profesional y seguro contra inyecciones SQL. Usando bloques **try-catch** para que, si algo falla, la aplicación no "explote" y pueda comunicar lo que está sucediendo sin mostrar datos extraños al usuario.

Frontend y "Toque" Móvil Para que la web sea verdaderamente responsive y no tardar mucho con el CSS, aprovechamos **Bootstrap**.

Código limpio: Al separar el **header** y el **footer** en una carpeta /shared. Así, si se realiza un cambio de un enlace del menú, se actualiza en todas las páginas a la vez, lo que ahorra mucho trabajo.

Interactividad: Incorporado un poco de **JS** para que la web sea más ágil y no dependa únicamente de recargar las páginas.

Estrategia PWA: Para aprovechar todo el código que ya se había hecho en PHP, configuramos la app como una **PWA (Web Progresiva)**. La idea es que se sienta como una app nativa usando **Flutter y las APIs de Google (Firebase)**, pero reutilizando la base que ya estaba desarrollada. Es una forma de tener una app híbrida sin necesidad de programar todo de nuevo.

Entorno de trabajo Todo el desarrollo se ha realizado en la máquina virtual Linux de Isard. Esto ha permitido lidiar con los permisos de carpetas y la configuración real de **Apache**, que es como se trabaja en un entorno de producción real.

Entrega y Despliegue (GitHub y MySQL)

Para que el proyecto se pueda evaluar sin líos, todo el código se ha subido en forma pública a **GitHub**. La idea es que sea "descargar y listo":

1. **Código Fuente:** Está todo el árbol de directorios con el MVC, los assets y la carpeta /shared. Solo hay que clonar el repo en cualquier localhost (XAMPP, Laragon o directamente en una VM Linux como la de Isard).
2. **Base de Datos (Dump):** He incluido un archivo .sql (un dump) con toda la estructura de tablas y algunos datos de prueba. Para que funcione, solo hay que importar ese archivo desde PHPMyAdmin o por terminal.
3. **Configuración:** He dejado el archivo de conexión preparado para que solo haya que cambiar los credenciales del config.php y que la app conecte con la base de datos local al momento.