

Proyecto Integrado



Antonella Racconto
2º DAW Aula Bilingüe

Resumen

El siguiente proyecto consiste en la creación de una página web para una empresa dedicada al mundo del pilates. La página tiene el objetivo de informar al cliente de los servicios que ofrece la empresa, además de llevar un control de horarios, pagos, clases y usuarios.

The following projects consists of the creation of a web page for a company dedicated to the world of Pilates. The page has the objective of informing the client of the services offered by the company, in addition to carrying out a control of schedules, payments, classes and users.

Tabla de contenido

1.	Justificación	3
1.1.	Características generales	3
1.2.	Aspectos a cubrir y los que no se van a tratar	3
2.	Tecnología empleada	4
3.	Requerimientos hardware y software	4
3.1.	Hardware	4
3.2.	Software	4
4.	Análisis y diseño	5
4.1.	Diagrama de casos de uso	5
4.2.	Diagrama de clases.....	6
4.3.	Base de datos relacional	7
5.	Implementación	8
6.	Relación detallada de cada uno de los ficheros.	11
7.	Evaluación y prueba	12
8.	Manual de estilos	13
8.1.	Sketch.....	13
8.2.	Criterios de accesibilidad	15
8.3.	Criterios de usabilidad.....	16
8.4.	Fuentes principales y secundarias	16
8.5.	Tamaños principales.....	17
8.6.	Mapa de colores.	17
8.7.	Dispositivos y vistas para las que se ha diseñado el proyecto.....	18
9.	Mejoras posibles y aportaciones.....	20
10.	Bibliografía	20

1. Justificación

1.1. Características generales

La página creada contiene las siguientes características:

- **Objetivo.** Informar al cliente la variedad de servicios que ofrece la empresa con una interfaz intuitiva y dinámica.
- **Funcionalidad.** El alta de usuarios la realiza el administrador de la empresa y el cliente realiza el login con el objetivo de ver su perfil. El administrador posee todos los permisos para gestionar la página web, como, la creación de horarios y la gestión de pagos.
- **Público.** La web está orientada a todo aquel usuario que quiera informarse sobre el método pilates y al usuario dado de alta en la empresa como cliente.

1.2. Aspectos a cubrir y los que no se van a tratar

En el siguiente apartado describiré aquellos aspectos que se han logrado cubrir a lo largo del proyecto y aquellos que no.

Aspectos a cubrir:

- **Patrón de diseño MVC.** Para implementar interfaces de usuario, datos y lógica de control.
- **Diseño.** Se trata de una interfaz dinámica e interactiva, con adaptación de distintos dispositivos.
- **Subir a producción los archivos.** El proyecto ha sido subido al hosting de Hostinger.

Aspectos que no se van a tratar:

- **Información confidencial de los usuarios.**
- **La funcionalidad de algunas páginas de la página web,** como pueden ser de contactar con la empresa.
- **Tratamiento de las imágenes.** Al ser la mayoría de imágenes extraídas de la base de datos, no se ha logrado realizar la adaptación de la resolución a los diferentes dispositivos.

2. Tecnología empleada

La tecnología empleada en el proyecto han sido las estudiadas y aprendidas por el alumno a lo largo del curso.

- PHP. Se ha elegido como lenguaje en entorno servidor ya que es la herramienta más conocida por el alumno. También ofrece una estructura previamente hecha de cómo organizar un proyecto.
Se había planteado la opción de utilizar frameworks como symphony o laravel, ya que ambos son de tipo MVC y tienen un buen rendimiento. Sin embargo, como se ha mencionado el alumno cuenta con una estructura previamente hecha de MVC que es el que utilizará para el proyecto.
- JavaScript. Se ha elegido como lenguaje en entorno cliente por su gran sencillez, integración con HTML y CSS, y librerías. Y, puesto que el alumno tiene experiencia con este lenguaje y utiliza algunas librerías donde se implementa este lenguaje, JavaScript ha sido el lenguaje elegido.
- HTML y CSS. Son los lenguajes utilizados para dar interactividad y estilo a la página web. Se ha utilizado SASS, ya que proporciona características y funcionalidades adicionales que no están presentes en CSS tradicional, como las variables, los mixins y los anidamientos.

3. Requerimientos hardware y software

3.1. Hardware

- Ordenador. Se necesitará un ordenador con suficiente capacidad de almacenamiento para ejecutar el entorno de desarrollo y manejar los recursos necesarios para el proyecto.
- Conexión a Internet. Se necesitará una conexión a Internet estable y de alta velocidad.

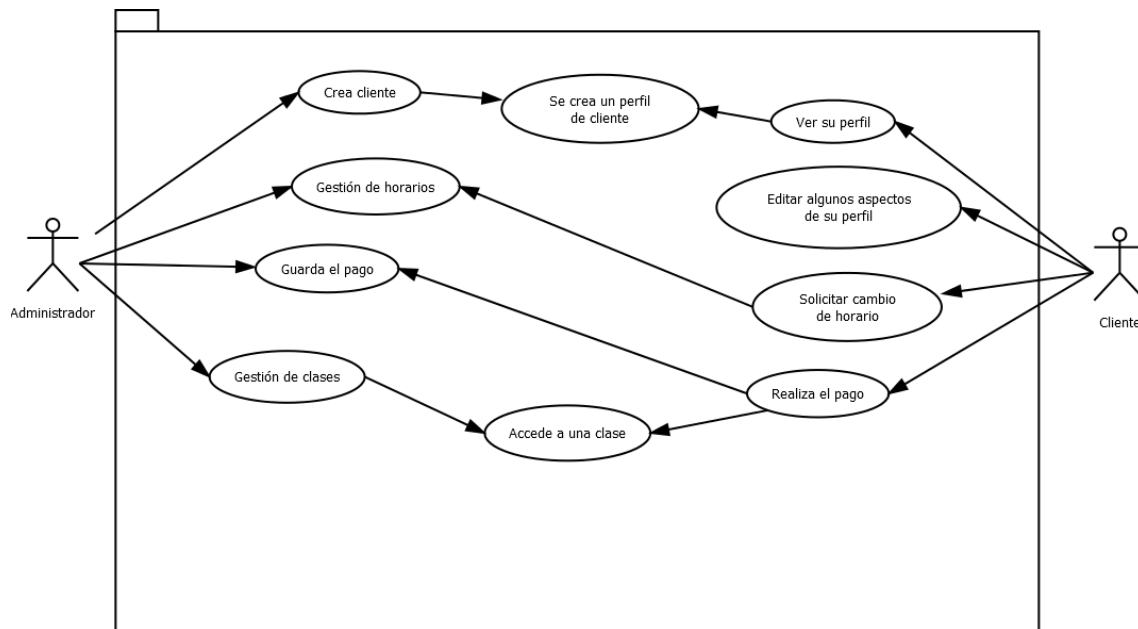
3.2. Software

- Sistema operativo. Se debe tener un sistema operativo compatible con la tecnología utilizada en el proyecto.
- Entorno de desarrollo integrado. Es una herramienta necesaria para el lado del servidor, ya que permite escribir, depurar y gestionar el código de manera eficiente. En este caso se ha utilizado Visual Studio Code.

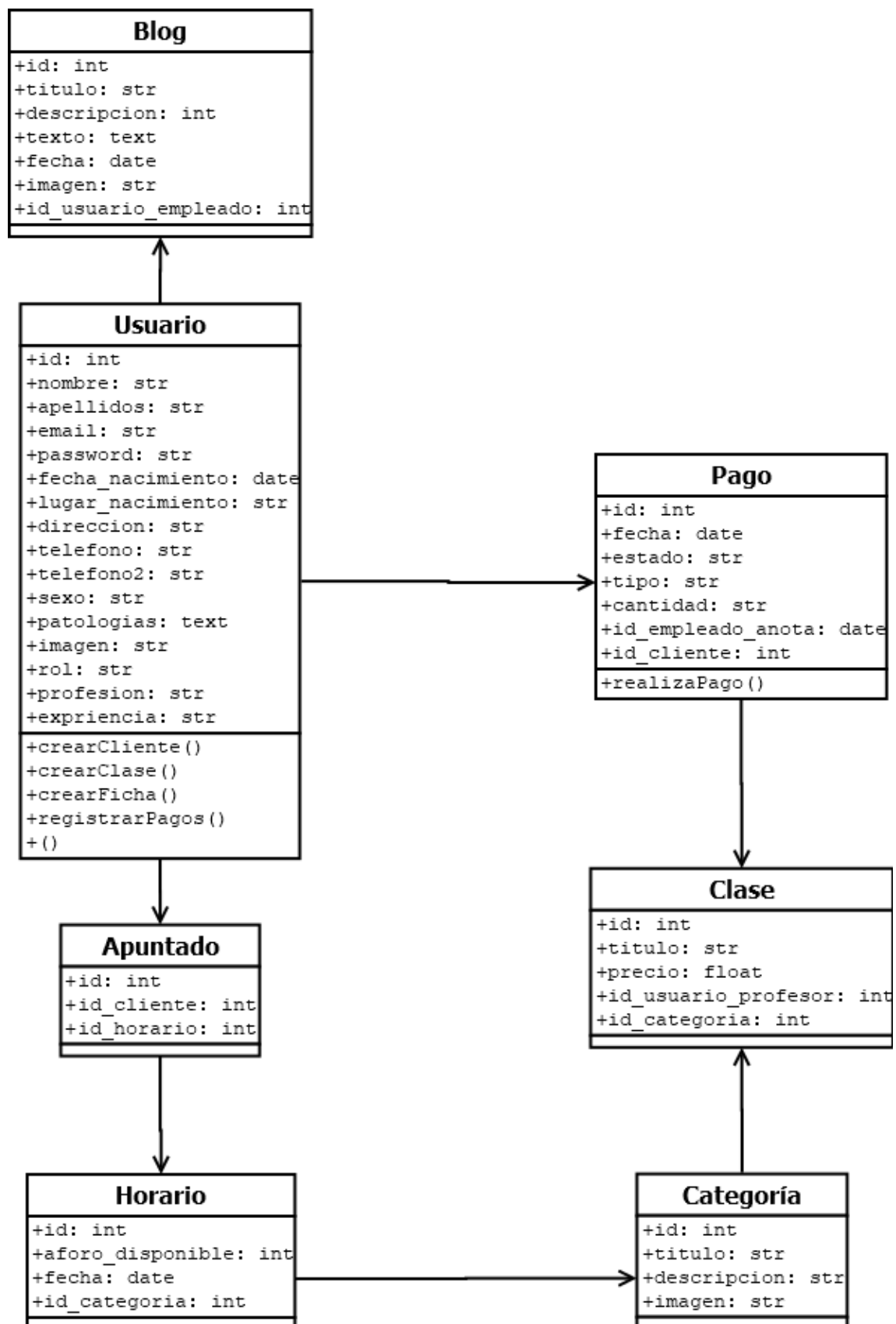
- Navegadores web. Es necesario tener un navegador instalado para probar el funcionamiento de la web. La página web creada solo ha sido probada en Google Chrome.
- Tecnologías específicas del proyecto. Por ejemplo, el servidor web utilizado ha sido Apache y para la base de datos se ha utilizado MySQL.
- Herramienta de control de versiones. Para llevar un seguimiento de los cambios realizados en el código se ha utilizado la herramienta Git.

4. Análisis y diseño

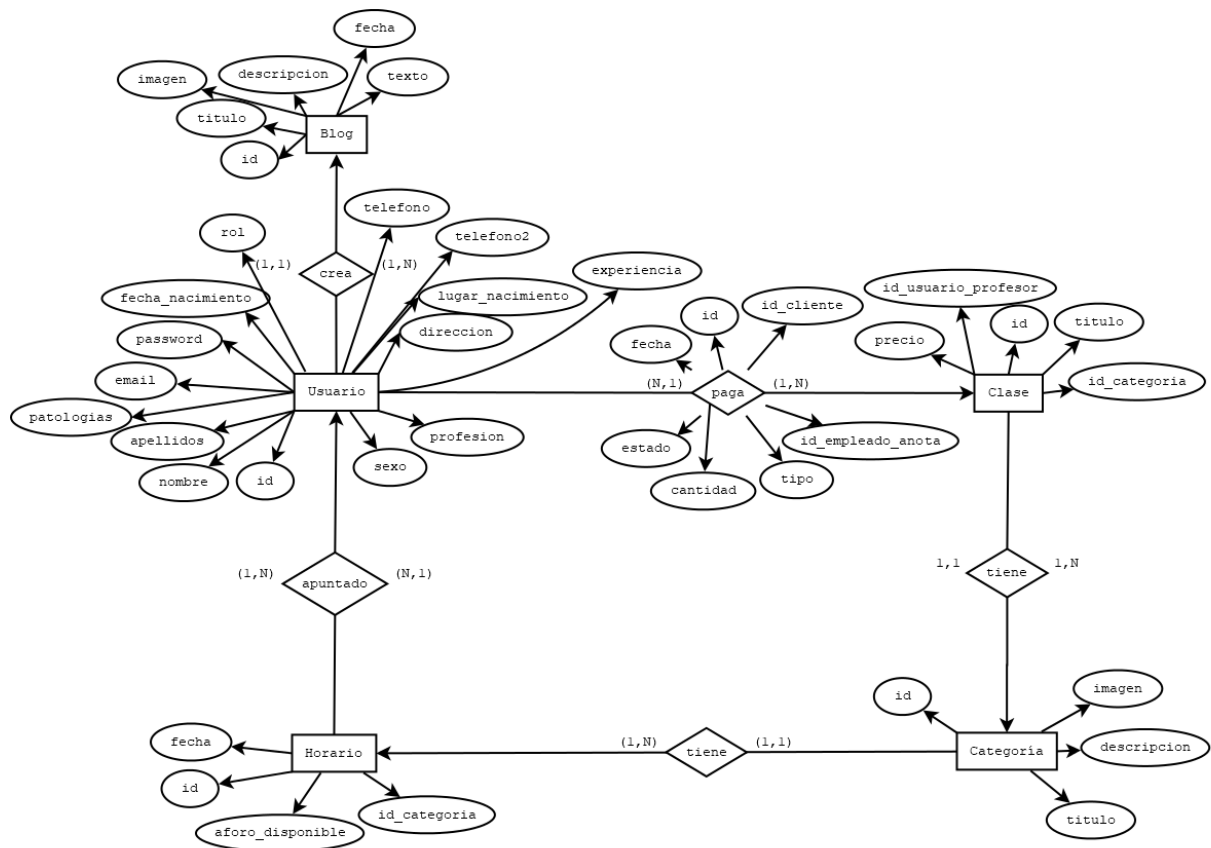
4.1. Diagrama de casos de uso



4.2. Diagrama de clases



4.3. Base de datos relacional



5. Implementación

- Hojas de estilos. Para definir la apariencia y el diseño visual de un sitio web se ha utilizado CSS. En este proyecto se han utilizado las siguientes hojas de estilo, que se han importado en el archivo estilo.scss:

```
@import "abstract/mixin";
@import "abstract/variables";

@import "layout/body";
@import "layout/nav";
@import "layout/header";
@import "layout/footer";
@import "layout/main";
@import "layout/crud";
@import "layout/sobreNosotros";
@import "layout/paginasSecundarias";
@import "layout/verUsuario";

@import "componentes/formularios";
@import "componentes/botones";
```

- Plantillas. Para definir la estructura de la página web se ha utilizado en la mayoría de las páginas la etiqueta <div> con una serie de clases para diferenciar de otras maquetaciones, pero muchas comparten la misma plantilla. Por ejemplo, en los ficheros gestion.php, se utiliza la misma maquetación.

```
<main>
  <!-- HEADER CONTENIDO -->
  <div class="header-submenu">
    <div class="overlay">
      <h1>Gestión de Horario</h1>
    </div>
  </div>

  <!-- MAIN CONTENIDO -->
  <div class="main-contenido">
    <div class="crud">
      <div class="añadir">
        <a class="add" href="<?=$_ENV['base_url'] ?>horario/crear">Añadir +</a>
      </div>

      <table id="tabla" class="display responsive nowrap" style="width:100%">
        <thead>
          <tr>
            <th>ID</th>
            <th>Aforo disponible</th>
            <th>Día y hora</th>
            <th>Categoría</th>
            <th>Opciones</th>
          </tr>
        </thead>
        <tbody>
          <?php $horarios = Horario::obtenerHorario(); ?>
          <?php while ($horario = $horarios->fetch(PDO::FETCH_OBJ)) : ?>
```

- Formularios. Los formularios que se han implementado en el proyecto son los del registro, login, inserción de datos y edición. Por ejemplo, en la siguiente imagen tenemos un formulario para crear un nuevo horario. La mayoría de formularios siguen el mismo estilo.



A screenshot of a web form for creating a new schedule. The form is set against a dark purple background. It contains three input fields: 'Aforo disponible:' with a white text input box, 'Día y hora:' with a date and time picker showing 'dd/mm/aaaa --:--' and a calendar icon, and 'Categoría:' with a dropdown menu showing 'Selecciona una categoría' and a downward arrow. Below these fields are two buttons: an orange 'Crear' button and a black 'Volver' button.

- Funciones de envío de datos. Para el envío de datos se ha utilizado el método post y get, y funciones de javascript, como pueden ser buscador de las tablas de gestión que está incluido en la librería DataTable
- Conexión. La conexión se ha realizado con la extensión PDO, utilizando las variables del fichero .env.

```
DB_HOST = localhost
DB_USER = root
DB_PASS =
DB_DATABASE = proyectozesari
```

```

namespace Lib;
use PDO;

class BaseDatos extends PDO {

    private string $servidor;
    private string $usuario;
    private string $pass;
    private string $base_datos;
    private string $tipo_de_base = 'mysql';

    function __construct(){
        $this->servidor = $_ENV['DB_HOST'];
        $this->usuario = $_ENV['DB_USER'];
        $this->pass = $_ENV['DB_PASS'];
        $this->base_datos = $_ENV['DB_DATABASE'];
        try{
            $opciones = array(
                PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES utf8",
                PDO::MYSQL_ATTR_FOUND_ROWS => true
            );
            parent::__construct("{$_this->tipo_de_base}:dbname={$_this->base_datos};host={$_this->servidor}", $_this->usuario, $_this->pass, $opciones);
        }catch(PDOException $e){
            echo "Ha surgido un error y no se puede conectar a la base de datos. DETALLE: ". $e->getMessage();
            exit;
        }
    }
}

```

- Consultas. En el proyecto se han utilizado, en la mayoría de funciones, consultas preparadas.

```

public function borrar($id) {
    /**
     * Borra un blog según el id que se le pasa
     * Si se ha borrado devuelve true y si no devuelve false
     */
    $sql = "DELETE FROM blogs WHERE id = :id";
    $resul = $this->db->prepare($sql);
    $resul->bindParam(':id', $id, PDO::PARAM_STR);
    try{
        $resul->execute();
        $result = true;
    }catch(PDOException $err){
        $result = false;
    }

    return $result;
}

```

6. Relación detallada de cada uno de los ficheros.

En el siguiente apartado se describirán los ficheros de las vistas. Hay que destacar que algunos ficheros se repiten en algunas carpetas, como gestion, crear, editar, etc., y la única diferencia entre ellos es que son distintos datos.

- index.php. Muestra la página principal. Sus enlaces redirigen a los ficheros: nav-sobreNostros.php y nav-contacto.php.
- header.php. Muestra el encabezado de la página principal. Redirige a todos los enlaces de la página.
- footer.php. Muestra el pie de página.
- ver.php. Muestra un determinado elemento (un usuario, una categoría, un blog, etc.) gracias al id que se le manda al controlador correspondiente. Se utiliza el método get.
- registro.php. Con el método get se muestra el formulario de registro del usuario y con el método post se le envían todos los datos necesarios para que se registre el usuario.
- login.php. Con el método get se muestra el formulario de login del usuario y con el método post se le envían todos los datos necesarios para que el usuario realice el login. El login redirige al index.php.
- perfil.php. Muestra el perfil de un determinado usuario.
- gestion.php. Muestra la tabla de gestión de un determinado elemento. En este caso, este fichero se encuentra en los directorios: usuario, pago, horario, clase, categoría y blog.
 - Añadir. Redirige al fichero crear.php, donde hay un formulario para insertar los datos.
 - Borrar. Elimina la fila seleccionada por el método get, pasándole el parámetro id.
 - Editar. Redirige al fichero editar.php, donde se editan los datos pasando por el método get, el parámetro id.

En gestión de usuarios hay tres opciones diferentes:

- Ver. Redirige al archivo ver.php, donde se observa la ficha del usuario. En este archivo se puede eliminar el usuario.
- Pagar. Redirige al archivo pagoUsuario.php, al que se le pasa el id del usuario.
- Apuntar. Redirige al archivo apuntar.php al que se le pasa el id del usuario.

7. Evaluación y prueba

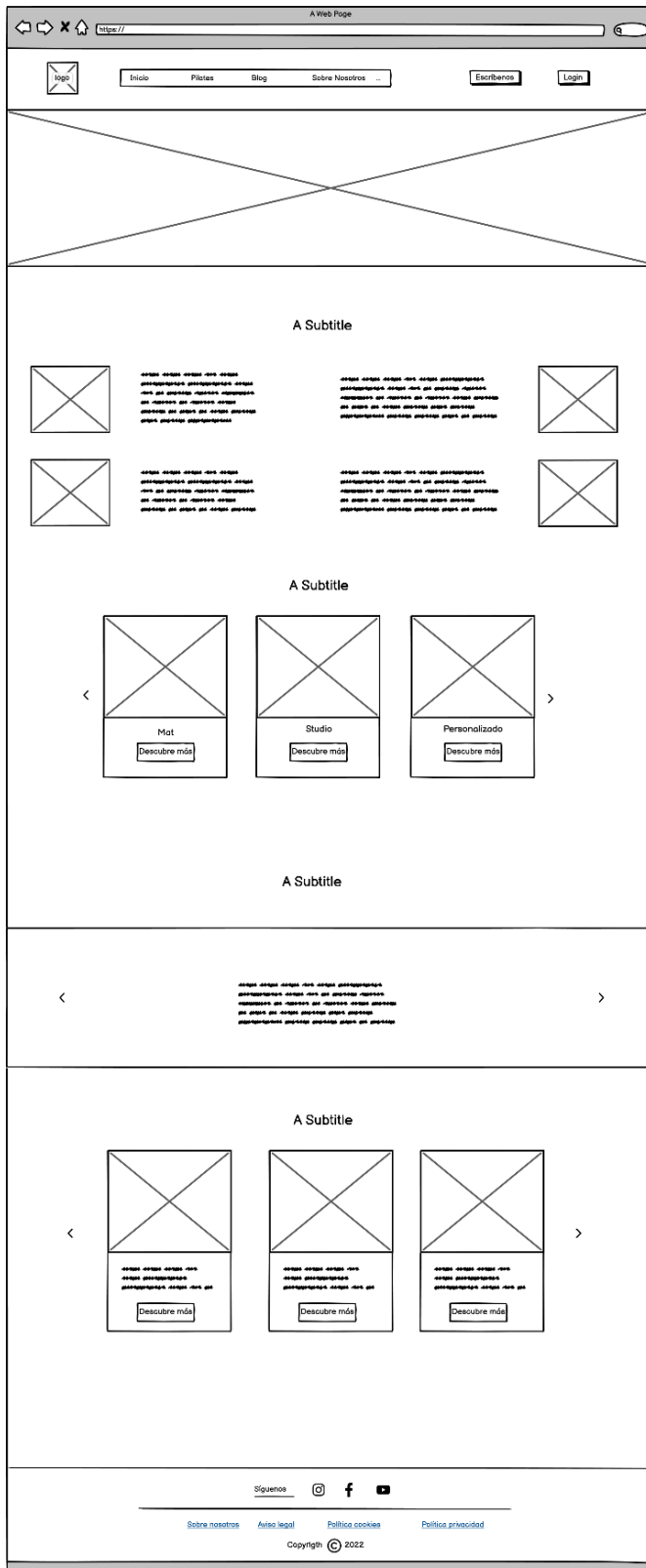
El testeo se ha realizado con la verificación de campos y la realimentación al usuario. Algunos ejemplos serían:

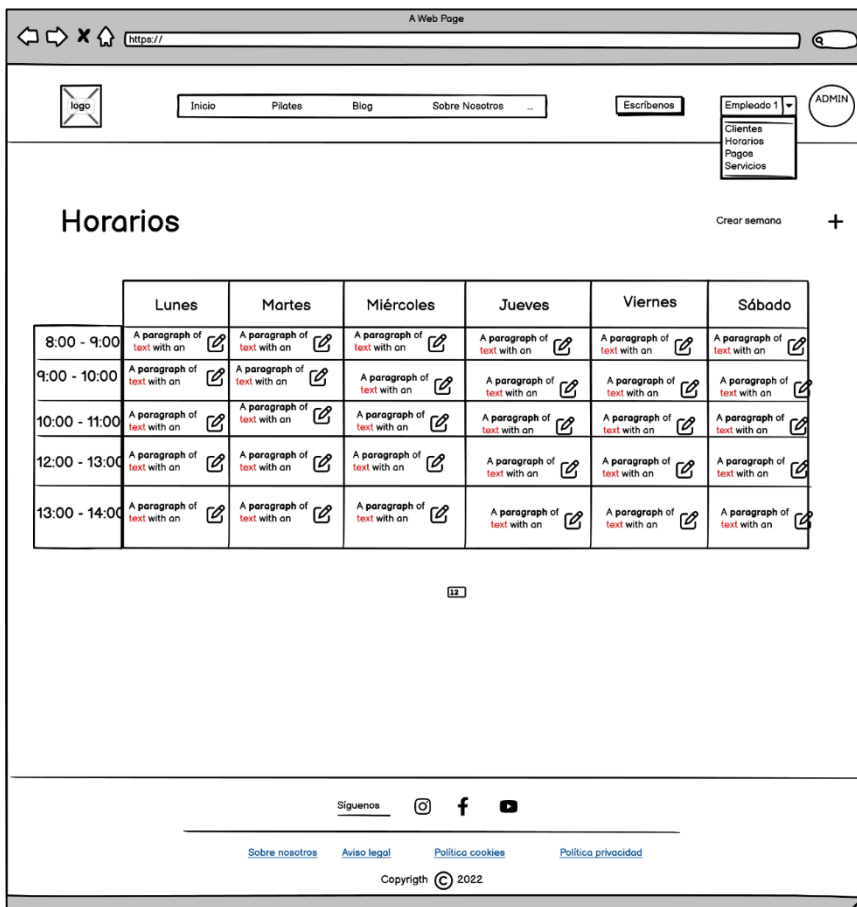
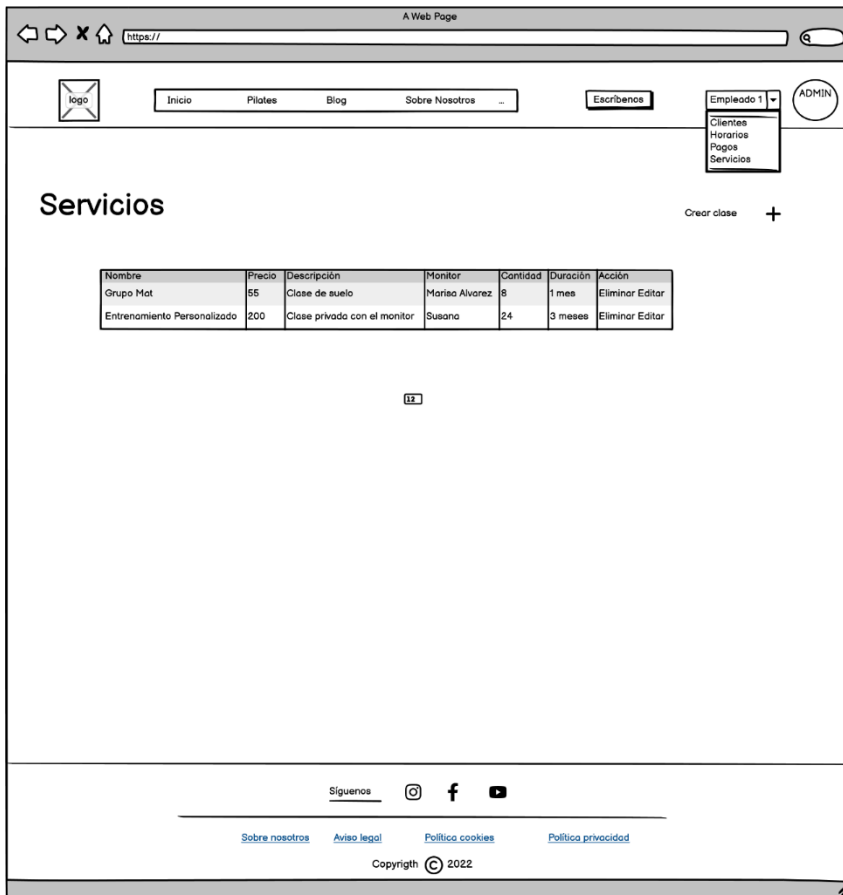
- Los datos obligatorios en los formularios deben ir con required. De esta forma, solo quedan el resto de validaciones en el lado del servidor.
- Validación de campos en la creación y edición de un usuario. Donde los nombres y apellidos deben ser en mayúscula. Si no lo son, devuelve un mensaje de error, especificando el error en cuestión.
- En la creación de clases, etc. Donde hay un select o el valor de un input debe ser numérico se valida, devolviendo el error en cuestión.
- A la hora de apuntar a un cliente en un horario, si la clase está llena, aparece dicho error.

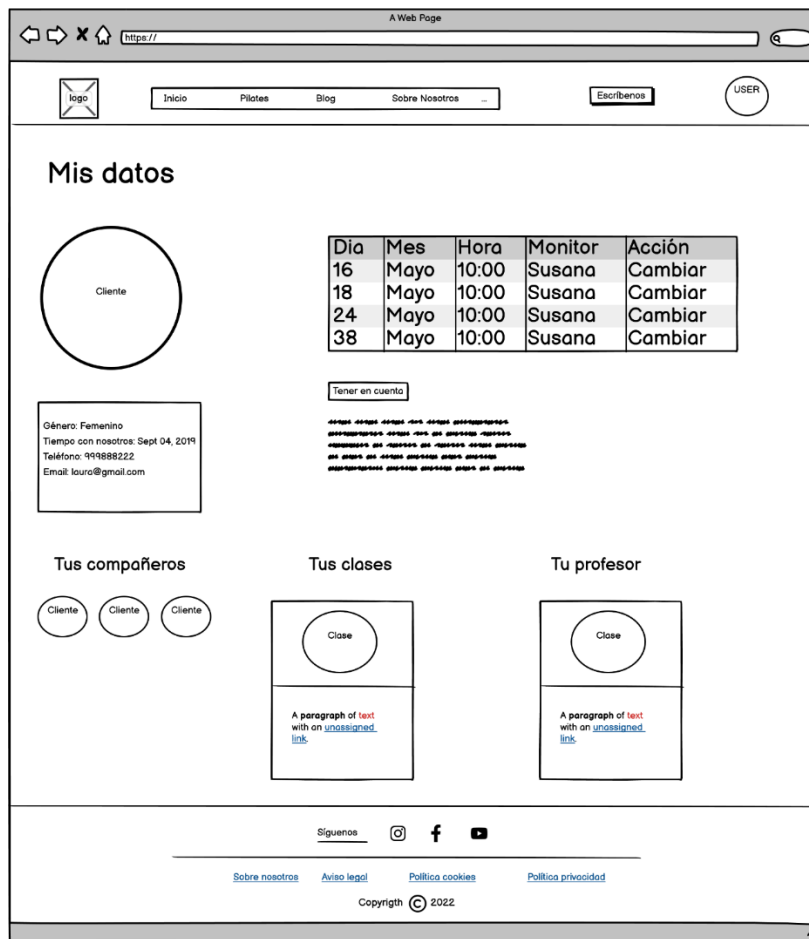
8. Manual de estilos

8.1. Sketch

Los siguientes sketches corresponden a la página principal, un ejemplo de una de las tablas de gestión, los horarios y el perfil de usuario.







8.2. Criterios de accesibilidad

Los criterios de accesibilidad que se han seguido en la página web realizada son:

- Perceptibilidad. Se proporcionan alternativas textuales como imágenes, se utilizan colores y contrastes adecuados para garantizar que el contenido sea legible.
- Comprensibilidad. Se utiliza un lenguaje claro y sencillo en el contenido.

A continuación, se ha realizado la comprobación de accesibilidad de la página web, donde la mayoría de errores son a raíz del contraste de algunos títulos. Sin embargo, se ha considerado que el contraste es el correcto.

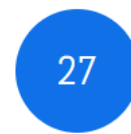
URL: https://empresazesari.com/
Fecha de escaneo: 2023-06-15T18:21:14.195Z
Resultados totales: 41



errores



Advertencias



Avisos

8.3. Criterios de usabilidad

- Facilidad de aprendizaje. Los elementos de la interfaz deben ser intuitivos y comprensibles.
- Retroalimentación y respuesta del sistema. El sistema proporciona una retroalimentación clara y oportuna a las acciones de los usuarios, como, por ejemplo, cuando se crea una clase, aparece un aviso.

8.4. Fuentes principales y secundarias

En la página solo se ha implementado un tipo de fuente y es “**Open Sans**”.

Regular 400

ZESARI

8.5. Tamaños principales

Los tamaños principales que se han utilizado en la mayoría de fuentes son las siguientes:





```
$letraPeque: 15px;  
$letraMediana: 20px;  
$letraH1secundario:50px;  
$letraTitulo: 4vmin;
```

8.6. Mapa de colores.

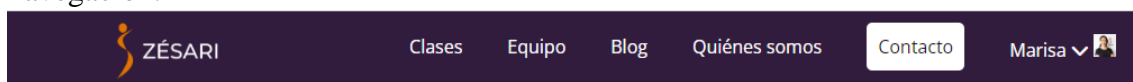
El mapa de colores ha sido elegido obteniendo la paleta de colores del logo de la empresa.



```
$paleta1: #321c3d;  
$paleta2: #84778c;  
$paleta3: #D96704;  
$paleta4: #8C471F;
```

Color	RGB	Hexadecimal
	(58,28,61)	#321c3d
	(132,119,140)	#84778c
	(217,103,4)	#d96704
	(140,71,31)	#8c471f

- Paleta1. Se utiliza en los fondos de la página. Por ejemplo, el fondo del menú de navegación.



- Paleta2. Se ha utilizado para los fondos de los formularios. Por ejemplo, en el formulario para crear la clase:

Titulo:

Precio:

Cantidad:

Profesor:

Categoria:

- Paleta3. Se ha utilizado para algunos títulos y botones.

Leer Más

¿Por qué elegir Zésari?

Estas son algunas de nuestras razones

- Paleta4. Este color no se ha utilizado por el momento.

8.7. Dispositivos y vistas para las que se ha diseñado el proyecto

El proyecto ha sido diseñado para adaptarse a los dispositivos móviles, tablets, portátiles y ordenadores de mayor resolución. En la siguiente imagen se observan los diferentes puntos de ruptura para los distintos dispositivos.

```
$breakpoints: (  
  grid-tablet: 600px,  
  movil-max: 767px,  
  tablet-min: 768px,  
  nav-menu: 1279px,  
  portatil-l: 1440px,  
  ordenador-4k: 2000px,  
);  
  
@mixin max($breakpoint) {  
  /*Es la resolución maxima de un dispositivo*/  
  @media only screen and (max-width: map-get($breakpoints, $breakpoint)) {  
    @content  
  }  
};  
  
@mixin between($breakpointMin, $breakpointMax) {  
  /*Es la resolución entre dos medidas*/  
  @media only screen and (min-width: map-get($breakpoints, $breakpointMin)) and (max-width: map-get($breakpoints, $breakpointMax)) {  
    @content  
  }  
};  
  
@mixin min($breakpoint) {  
  /*Resolución mínimas para los dispositivos*/  
  @media only screen and (min-width: map-get($breakpoints, $breakpoint)) {  
    @content  
  }  
};
```

Algunas páginas donde se utilizan estas funciones son el menú y en los display grid.

- `max(movil-max)`. En el siguiente ejemplo se utiliza esta función para el menú responsive.



- `between(tablet-min, nav-menu)`. En el siguiente ejemplo se utiliza esta función para el display grid de los principios.



Pasión por el pilates

En Zesari intentamos transmitir la pasión del pilates en ti.



Ambiente único y exclusivo

Nuestro centro cuenta con un lugar excelente para practicar cualquier ejercicio.



Gran equipo y magníficas instalaciones

Grandes profesionales y las instalaciones muy amplias luminoso un lugar que invita al bienestar.



Clases adaptadas a ti

Nuestros centro procura ofrecerte la clase que más se adapte a ti.



Pasión por el pilates

En Zesari intentamos transmitir la pasión del pilates en ti.



Ambiente único y exclusivo

Nuestro centro cuenta con un lugar excelente para practicar cualquier ejercicio.



Gran equipo y magníficas instalaciones

Grandes profesionales y las instalaciones muy amplias luminoso un lugar que invita al bienestar.



Clases adaptadas a ti

Nuestros centro procura ofrecerte la clase que más se adapte a ti.

9. Mejoras posibles y aportaciones

- Mejorar la interfaz de la página web. Hay ciertas páginas en las que se necesitan mejorar los estilos como, por ejemplo, los márgenes, los anchos y ciertos tamaños de las fuentes.
- Mejorar el perfil de usuario. En el sketch se observa el objetivo final de la página, para que se mostrara toda la información necesaria para cliente. Esto se solucionaría realizando las funciones oportunas.
- Mejorar la funcionalidad. Algunas funcionalidades que deberían realizarse son:
 - En la página de contacto falta realizar el envío de mail a la empresa.
 - En el envío de mail del registro del cliente falta realizar el envío con el token y la confirmación de que el cliente ha cambiado la contraseña de su cuenta.
 - La gestión de los horarios y los pagos. La idea final era que el cliente pudiera pagar una clase, se le apuntara a un horario y ese horario se reflejara en un calendario en la página Horario. Sin embargo, por cuestiones de tiempo y prioridades no se consiguió. Esto haría más dinámica la página y cómoda de usar para el administrador.
 - Faltan ciertas validaciones en la edición de datos del usuario, como la contraseña y la longitud del móvil.
 - En la tabla de gestión de usuario debería haber un botón de desapuntar.
- La librería ckeditor se debe mejorar para que se puedan insertar títulos y más imágenes.
- Considero que la base de datos podría mejorarse, ya que las tablas de horarios y pagos fueron añadidos en último momento para que la página funcionara, pero, por ejemplo, la relación de horarios y categorías, debería ser entre horarios y clases. Este apartado queda como posible sugerencia.

10. Bibliografía

- <https://fonts.google.com/>
- <https://colorkit.co/color-palette-generator/321c3d-84778C-D96704-8C471F/>
- <https://fontawesome.com/search>
- <https://www.php.net/manual/es/intro-what-is.php>
- <https://datatables.net/>
- <https://ckeditor.com/>