

Proyecto 1

Página Web Fun Coruña

Iván Lado Canosa

Abril 2023

Índice

1. Introducción	4
1.1. Glosario	4
1.2. Requisitos funcionales	4
2. Inventario de contenido	5
3. Arquitectura de la Información	6
4. Diseño de interfaces	7
4.1. Prototipo manual	7
4.2. Esqueleto digital	7
4.3. Diseño final	13
4.4. Consideraciones generales para el diseño	14
4.4.1. Paleta de colores	14
4.4.2. Fondos	14
4.4.3. Fuente	15
4.4.4. Responsividad	15
4.5. Mapa de navegación	15
4.6. Casos de uso	15
4.6.1. CU-1: Ver los eventos de que hay en oferta clasificados por el tipo de evento.	16
4.6.2. CU-2: Reservar entradas para un evento	16
5. Storyboard	18
6. Estructura de ficheros	19
7. HTML: mapa de etiquetas	20
8. CSS: mapa de etiquetas	21
9. JavaScript	23
9.0.1. Fichero /js/faq.js	23
9.0.2. Fichero /js/form.js	24
9.0.3. Fichero /js/tarifas.js	25
9.0.4. Fichero /js/slideEventos.js	26
9.1. Métodos de acceso al DOM	27
9.2. Respuesta a eventos	28
9.2.1. Evento click	28
9.2.2. Evento submit	28
9.2.3. Evento keyDown	29
9.3. Objetos JQuery	29
9.4. Uso de JS6	29
9.5. Uso de expresiones regulares	30
9.6. Carga contenido en formato JSON	30
9.7. Carga contenido en formato XML	32

A. Anexo I: Diseños finales.	34
B. Anexo II: HTML mapas de etiquetas	40
C. Anexo III: CSS mapas de etiquetas	45

1. Introducción

En este proyecto se ha analizado, diseñado y desarrollado la página web para la empresa ficticia Fun Coruña, una empresa que se dedica a la gestión y promoción de eventos que tengan lugar en la zona geográfica de la ciudad de A Coruña.

1.1. Glosario

- Evento: Cualquier acto o suceso deportivo, cultural, social o musical a la que asista un cierto público a presenciarlo.
- Tipos de eventos: Deportivo, cultural, social o musical.
- Cliente: Cualquier persona que use los servicios de Fun Coruña. Puede ser de dos tipos: empresas o usuarios (aquellos no son empresas, sino particulares).
- FAQs: Preguntas frecuentes.
- Plan: Tipo de tarifa. Pueden existir distintos planes a distintos precios, cada uno ofertando servicios diferentes.

1.2. Requisitos funcionales

Se han identificado los siguientes requisitos funcionales para la página web:

- RE_01: Mostrar información de carácter general sobre la empresa, como una breve presentación, o la historia de la misma.
- RE_02: Mostrar información de los servicios prestados a empresas y usuarios.
- RE_03: Mostrar preguntas frecuentes tanto ambos tipos de clientes: empresas y usuarios.
- RE_04: Mostrar distintas tarifas de los distintos planes ofertados a clientes, así como los servicios que se incluyen con cada tipo de plan.
- RE_05: Mostrar eventos a usuarios, clasificados según su tipo.
- RE_06: Reservar entradas.
- RE_07: Mostrar la opiniones de clientes.

2. Inventario de contenido

En la Figura 1 se muestra un Inventario de Contenido, en el que se recoge y organiza toda la información (temas centrales o ítems) que se mostrarán en la página web, con el objetivo de garantizar que todo el contenido necesario está presente y que no hay redundancias o contenido superfluo.

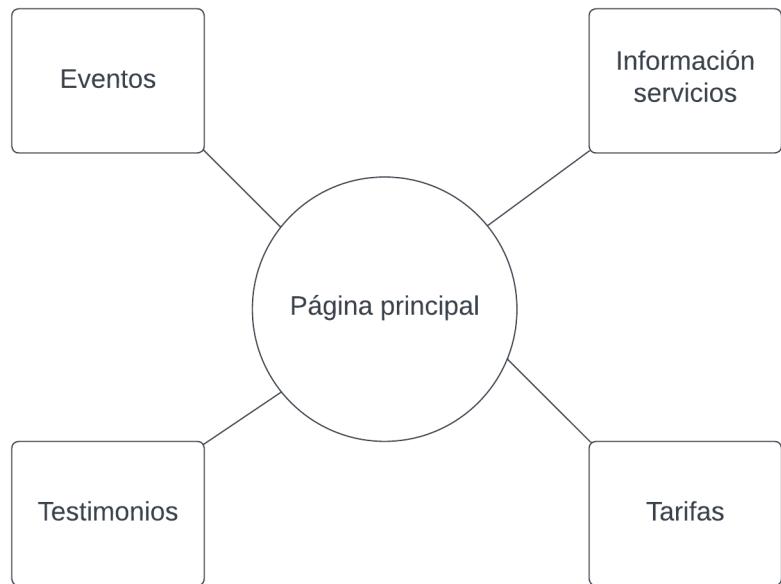


Figura 1: Inventario de contenido

3. Arquitectura de la Información

La diferente información y funcionalidades que debe cumplir la página web se estructurará en diferentes páginas. Una vez establecidos los ítems centrales de la página web, es necesario establecer la arquitectura de la información (Figura 2), para establecer las relaciones entre los distintos elementos.

La página web se accederá a través de la página principal, en la que se mostrarán todos los eventos. Además, esta contendrá una barra de navegación a través de la cual se podrá acceder al resto de páginas. Este menú se mostrará en todas las páginas, para que sea posible acceder a todas las páginas desde cualquier página.

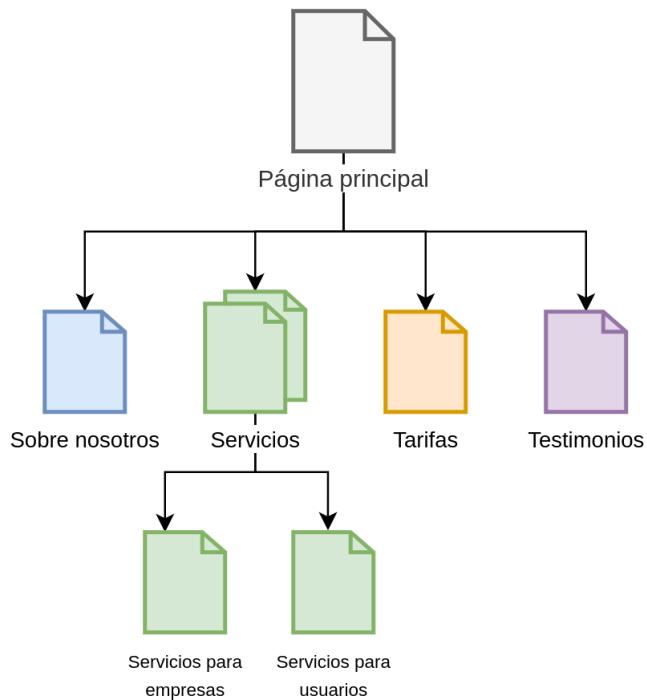


Figura 2: Arquitectura de la información

4. Diseño de interfaces

4.1. Prototipo manual

En la Figura 3 se muestra un primer prototipo realizado a mano alzada, en el que se trata de plasmar las ideas básicas del diseño de la página web, sin entrar en detalles.

El prototipo se realizó a partir de la Arquitectura de la Información de la sección anterior. Por lo tanto, hay un total de 6 páginas: Página principal, Sobre nosotros, Servicios para empresas, Servicios para usuarios, Tarifas y Testimonios.

Además, en la Figura 3 se muestra boceto a mayores, la del formulario. Esta no se cuenta como página, ya que el documento HTML que la implementará será el mismo que la página principal. El formulario será visible en el momento de pulsar el botón de reservar, y se mostrará por encima de la página principal.

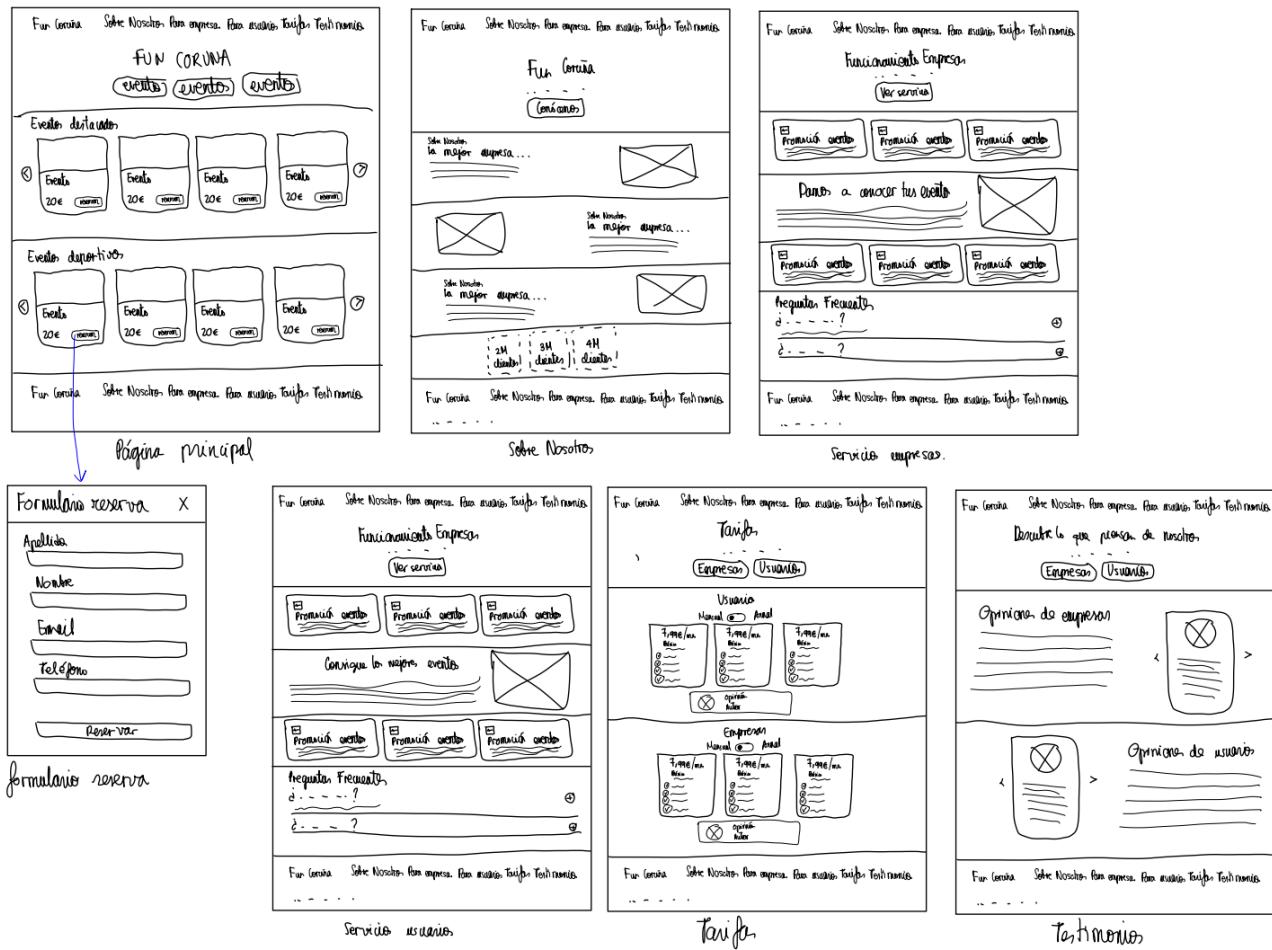


Figura 3: Prototipo manual hecho a mano

4.2. Esqueleto digital

Se ha realizado un wireframe (esqueleto) para cada página, con el objetivo de tener una primera idea de cómo distribuir los elementos dentro de las diferentes páginas. Así, se establecen las

proporciones que tendrán los distintos elementos en el diseño final.

Una característica común a todas las páginas es que la barra de navegación se encuentra en la parte superior, y el footer en la más inferior. En todas ellas, tanto la barra de navegación como el footer son el mismo, y centrados con respecto al eje vertical. Otro aspecto de diseño común, es que todos los contenedores (excepto la barra de navegación) se han centrado, tanto vertical como horizontalmente.

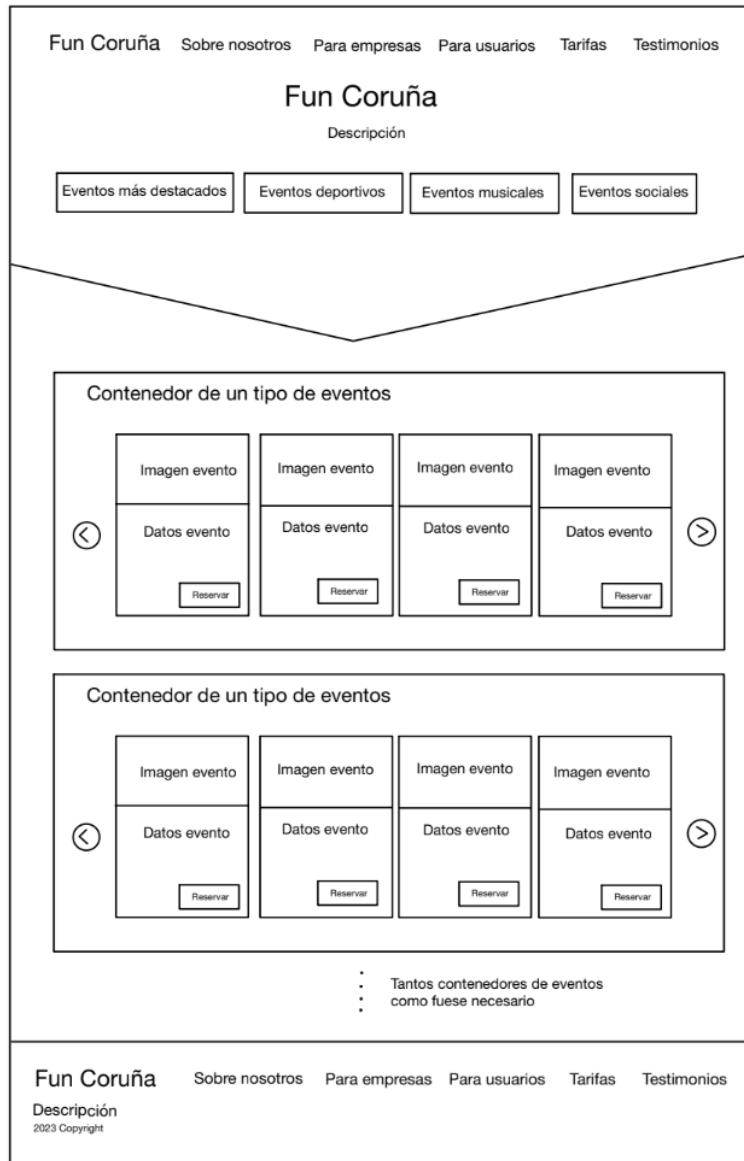


Figura 4: Wireframe página principal

En la Figure 4 se muestra del wireframe de la página principal. Podemos dividir su diseño en dos partes:

- La portada, en la parte superior. Donde se muestra la barra de navegación, el título de la página, y 4 botones alineados verticalmente en el centro de la portada.

- El contenido de la página, con cajas de eventos. Cada evento estará dividido en dos mitades: en la superior se mostrará una imagen representativa del evento, y una inferior información del mismo.

En la Figura 5 se muestra el wireframe del formulario de la reserva entradas. Como se ha dicho anteriormente, este formulario aparece cuando en la pantalla principal se clica en el botón de reservar de un evento. En ese momento, el formulario de muestra por encima del contenido de la página principal. Este formulario centra con respecto a la ventana del navegador.

El wireframe muestra un cuadro rectangular con un título "Formulario de reserva" en la parte superior izquierda y un icono de cierre ("X") en la parte superior derecha. Dentro del cuadro, hay seis secciones, cada una con la palabra "Dato" seguida de un cuadro rectangular vacío para introducir información. La última sección tiene el botón "Reservar" en su interior.

Figura 5: Wireframe formulario reserva eventos

En la Figura 6 se muestra el wireframe de la página sobre nosotros. Esta página se caracteriza por una portada en la parte superior que ocupa toda la pantalla (a diferencia del resto de páginas). Con respecto al contenido de la página, este se organiza en bloques, dividido horizontalmente en dos mitades: en una se muestra algún tipo de información, mientras que en la otra una foto. El lado en el que se encuentra la foto se va alternando.

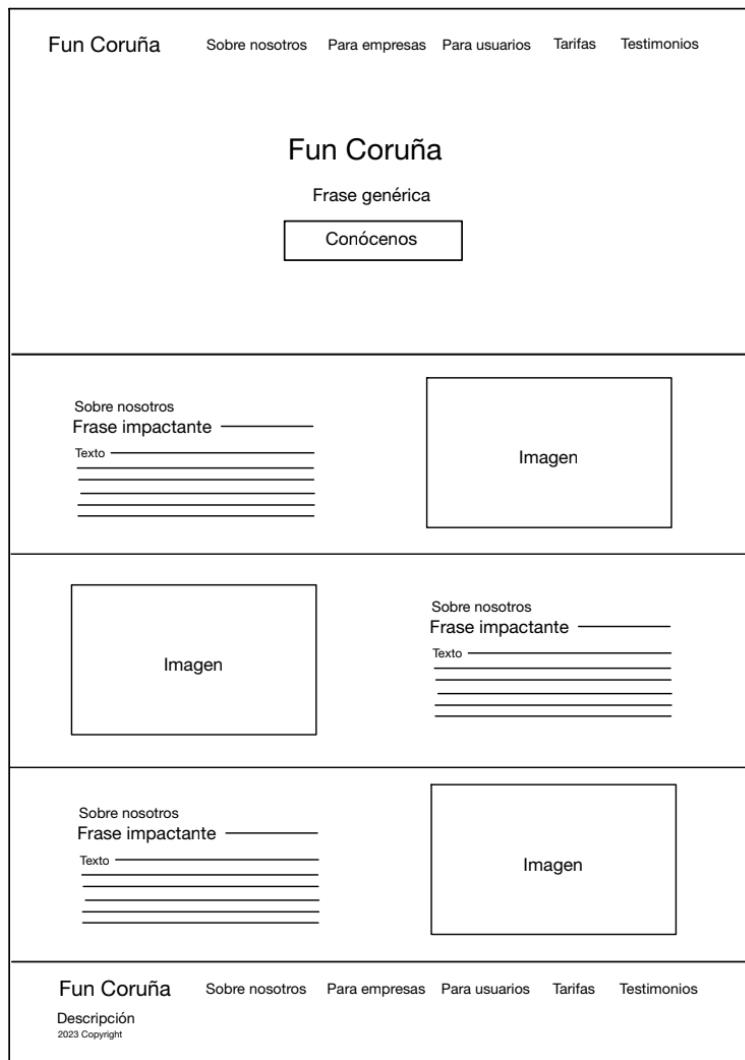


Figura 6: Wireframe página Sobre Nosotros

En la Figura 7 se muestra el wireframe de las páginas de servicios, el cual es el mismo para las empresas como para los usuarios. En ella, se mostrarán distintas características de los servicios ofertados por la empresa, así como las preguntas frecuentes (FAQs) en la parte inferior. Las FAQs se mostrarán sin respuesta por defecto, de tal forma que para que se despliegue la respuesta de una pregunta haya que clicar en ella.

Las secciones que muestran características de los servicios se despliegan secuencialmente. La primera y la tercera son tres cajas del mismo tamaño que se disponen en la misma fila, mientras que la segunda divide en dos columnas la información en formato de texto y la imagen.

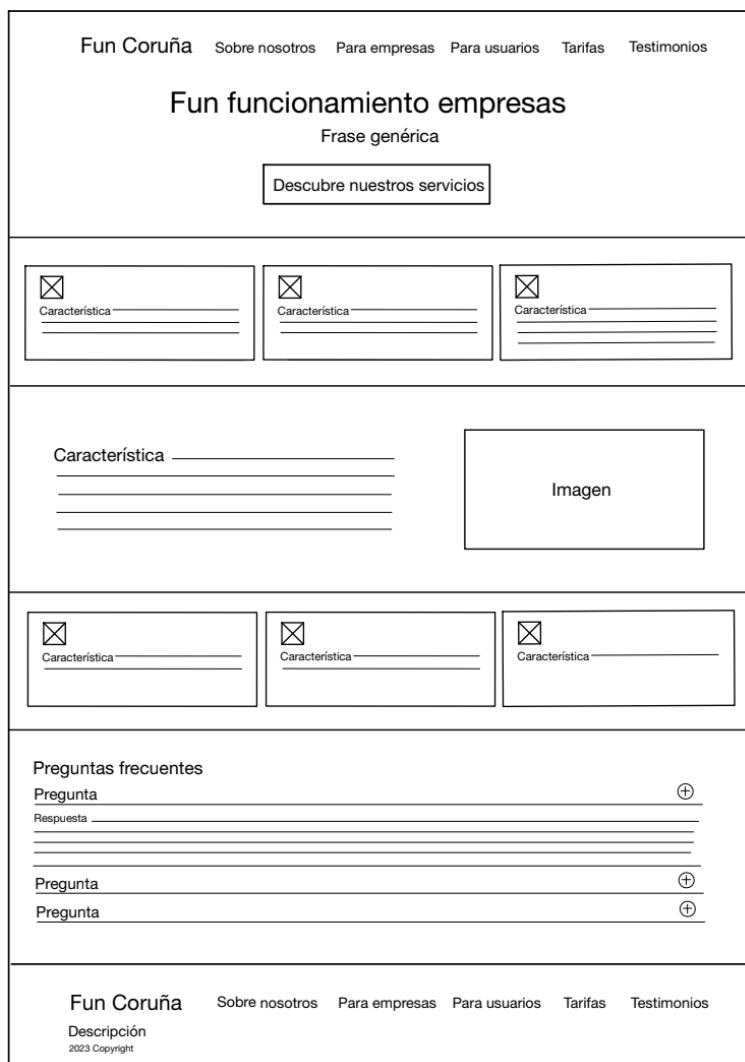


Figura 7: Wireframe páginas de servicios

En la Figura 8 se muestra el wireframe de la página de las tarifas. Hay dos secciones iguales, una para las tarifas de empresas y otra para la de usuarios. Los planes se muestran como 3 cajas del mismo tamaño centradas con respecto al eje vertical.

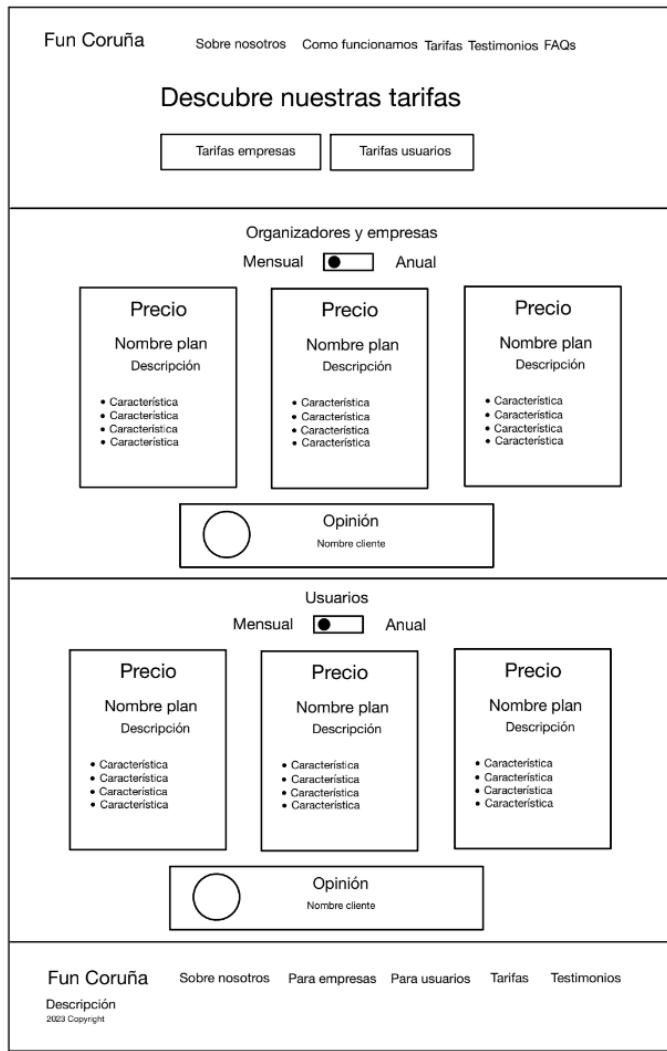


Figura 8: Wireframe tarifas

En la Figura 9 se muestra el wireframe de los testimonios. Hay dos secciones, una para las opiniones de las empresas, y otra para las de los usuarios. Estas son simétricas con respecto al eje vertical, y están divididas en dos lados: en uno muestran una opinión destacada, mientras que en el otro un “carousel” de opiniones.

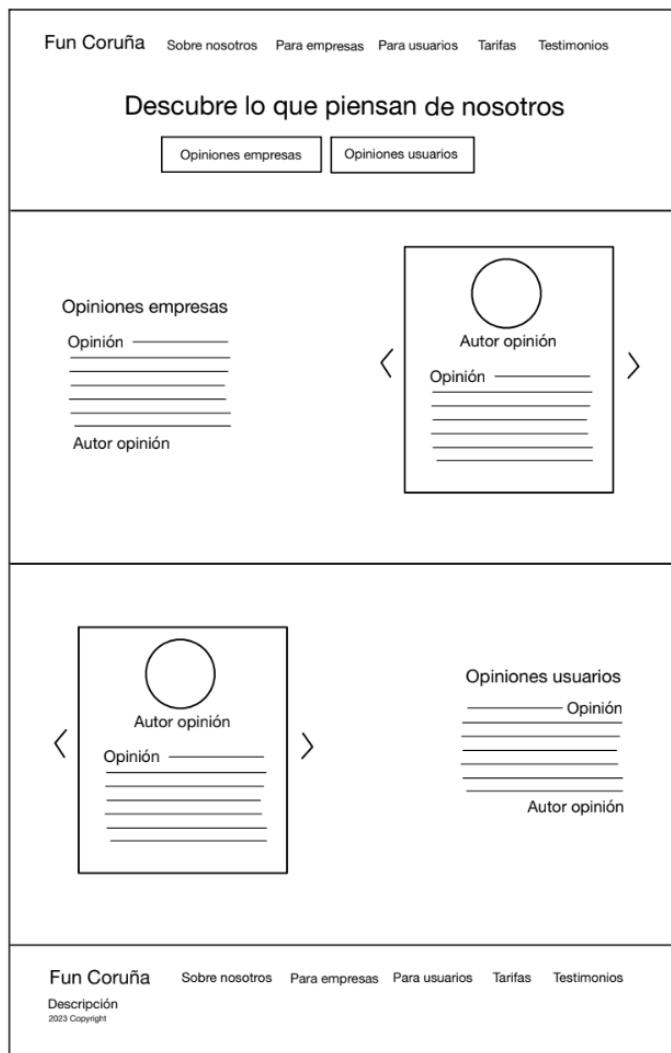


Figura 9: Wireframe tarifas

4.3. Diseño final

Los diseños finales se muestran en el Anexo I:

- Página principal, Figura 28.
- Página principal, formulario de reserva, Figura 29.
- Página “Sobre nosotros”, Figura 30.
- Página “Para empresas”, análoga, a “Para usuarios” Figura 31.
- Página “Tarifas”, Figura 32.
- Página “Testimonios”, Figura 33.

4.4. Consideraciones generales para el diseño

4.4.1. Paleta de colores

Se ha escogido la paleta de colores con el apoyo de Colormind, el cual permite generar una paleta de 5 colores mediante una IA, pudiendo el usuario fijar algunos colores. Si el usuario no fija ningún color, la herramienta sugiere otros.

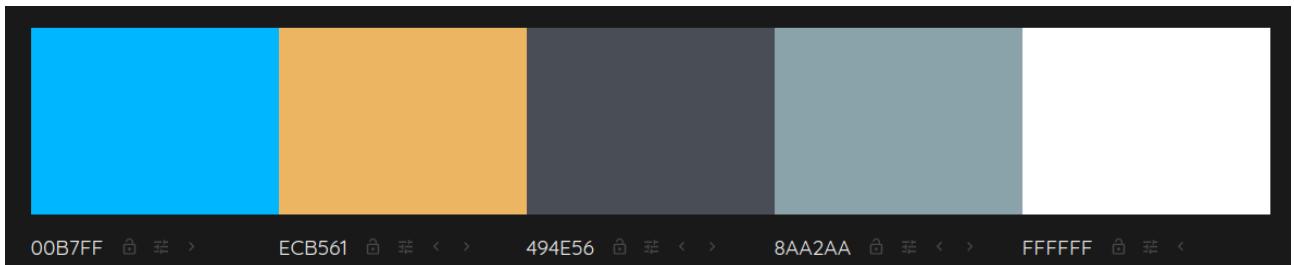


Figura 10: Paleta de colores

En la Figura 10 se muestran los colores elegidos para la paleta del sitio web. A la hora de generarla, se escogió el color azul como el principal de la página web, ya que es el más representativo y simbólico de la ciudad de A Coruña (de hecho el logo del Ayuntamiento es del mismo color). Además, debido al carácter social de los eventos, se buscó un color más “social”, por lo que se eligió el amarillo.

4.4.2. Fondos

En todas las páginas del sitio web se usaron dos fondos:

- Fondo gris, el tercero de la Figura 10.
- Fondo blanco. Para evitar los fondos completamente blancos, se eligió un color muy próximo al blanco con puntos amarillos o azules. Aunque es cierto que hay veces que no se aprecian los puntos, igualmente cumple el efecto óptico de que no parezca un fondo completamente blanco. En la Figura 11 se puede ver un ejemplo de fondo.



Figura 11: Fondo

4.4.3. Fuente

La fuente utilizada ha sido “Poppins” de Google Fonts, y es sin serif. Para los distintos tamaños de fuente, se ha usado la unidad relativa rem. Además, se ha usado la negrita en diferentes partes de la página web.

4.4.4. Responsividad

Se ha procurado que todas las páginas tengan un cierto grado de responsividad, garantizando que como mínimo, el contenido sea legible en todo tipo de dispositivos. Posteriormente, en la sección de CSS, se tratará un poco más este tema.

4.5. Mapa de navegación

Desde todas las páginas se pueden acceder a las mismas páginas, ya que todas tienen la misma barra de navegación, la cual se muestra en la Figura 12



Figura 12: Barra de navegación

De izquierda a derecha, se puede acceder a:

1. Fun Coruña: La página principal, donde se encuentran los eventos. En ella está también el formulario de reserva de entradas.
2. Sobre nosotros, en la que se encuentra información genérica sobre la empresa.
3. Para empresas, en la que se encuentra información sobre los servicios que ofrece Fun Coruña a clientes de tipo empresa.
4. Para usuarios, en la que se encuentra información sobre los servicios que ofrece Fun Coruña a clientes de tipo usuario.
5. Tarifas, en la que se encuentra información las tarifas para los dos tipos de clientes.
6. Testimonios, en la que se encuentran opiniones de antiguos clientes.

Debido a que desde todas las páginas se puede acceder a todas las páginas, se ha omitido la realización explícita de un mapa de navegación, que simplemente con la barra de navegación ya se ve toda posible navegación.

4.6. Casos de uso

EL diagrama de casos de uso de la Figura 13 muestra todas las operaciones que puede realizar un cliente en la página web.

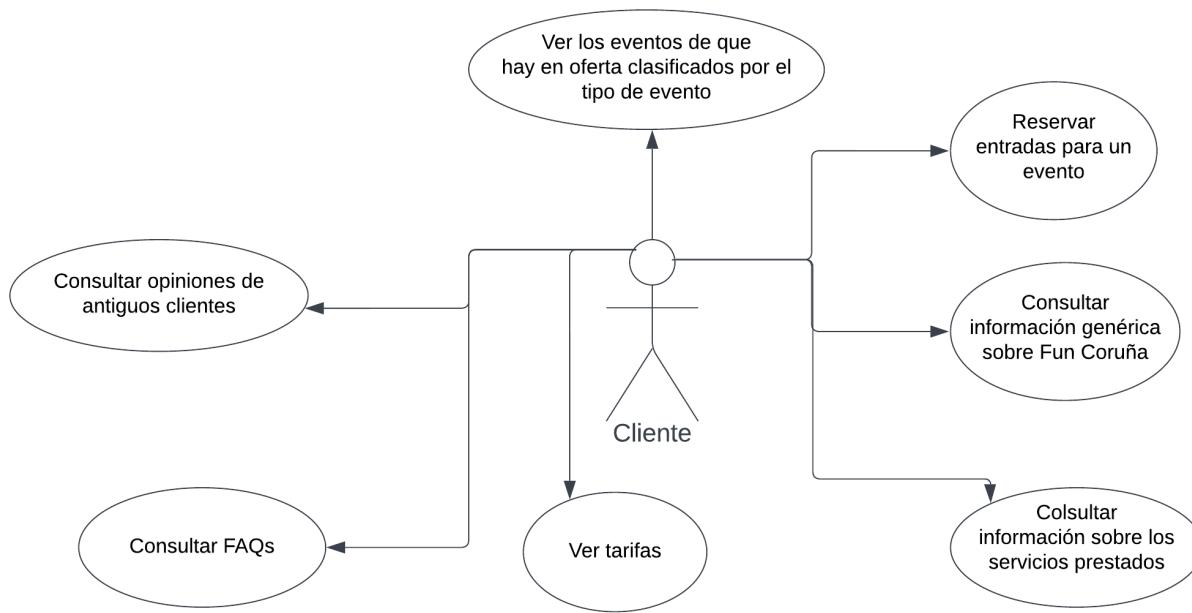


Figura 13: Diagrama de casos de uso

Como se puede ver, todos los casos de uso son de consulta de información excepto el de la reserva de entradas. Por lo tanto solo se detallarán dos, uno que servirá de ejemplo y modelo para todos los de consulta, y otro para la reserva de entradas.

4.6.1. CU-1: Ver los eventos de que hay en oferta clasificados por el tipo de evento.

- Descripción: Los clientes consultan los eventos que Fun Coruña oferta, de tal forma que aparecen agrupados por tipo de evento.
- Precondición: El cliente ha accedido al sitio web.
- Secuencia normal:
 1. Si el cliente se encuentra en la página principal, ya se encuentra en esa misma página la información que desea consultar. Si se encuentra en otra página distinta de la principal, selecciona en la barra de navegación “Fun Coruña”.
 2. El cliente recibe la página principal con los eventos.
- Postcondición: El cliente ha obtenido la página con los eventos.

4.6.2. CU-2: Reservar entradas para un evento

- Descripción: Los clientes puede reservar hasta un máximo de 10 entradas por cada reserva.
- Precondición: El cliente ha accedido al sitio web.
- Secuencia normal:

1. Si el cliente no se encuentra en la página principal, selecciona en la barra de navegación “Fun Coruña”.
 2. El cliente recibe la página principal con los eventos.
 3. El cliente se dirige al evento que le interesa y clica el botón de reservar.
 4. El cliente cubre el formulario emergente, llenando todos los campos obligatorios.
 5. El cliente clica el botón de reservar.
- Postcondición: El cliente ha reservado las entradas.

5. Storyboard

Como se ha dicho anteriormente, casi todos los casos de uso son de consultar información. En estos casos, el storyboard estaría formado por una única imagen (en la que se indica qué elemento de la cabecera hay que clicar). Por lo tanto, no se han incluido este tipo de storyboards.

El único caso de uso que no es tan simple, es el de la reserva de entradas, (CU-2). Ver Figura 14

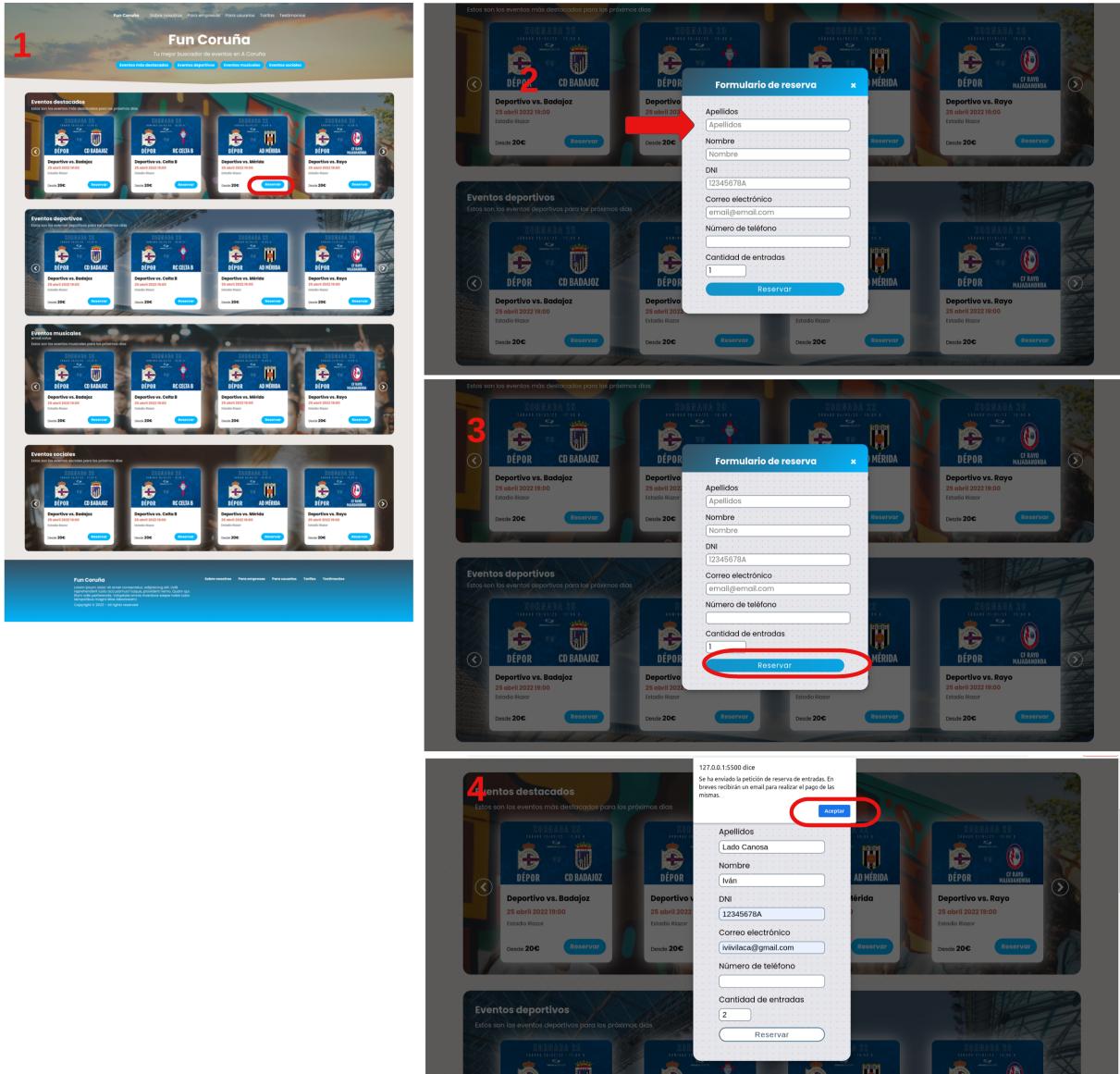


Figura 14: Storyboard CU-2: Reserva de entradas

6. Estructura de ficheros

En la Figura 15 se muestra un diagrama de la estructura de ficheros.

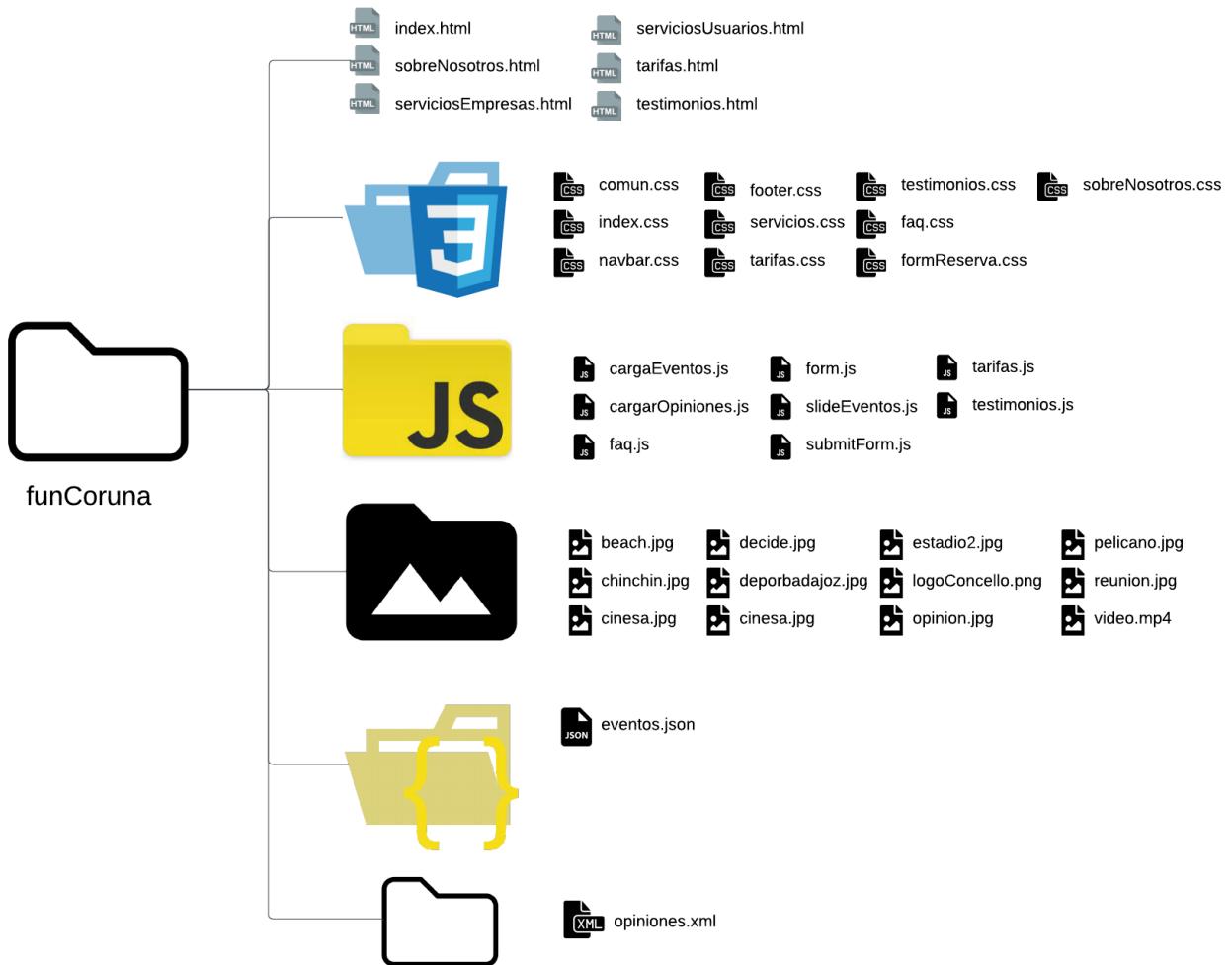


Figura 15: Mapa estructura de ficheros

7. HTML: mapa de etiquetas

Con respecto al código HTML, se ha usado procurado las etiquetas de HTML 5, las cuales incluyen etiquetas para la organización funcional. Así, en todas las páginas se ha usado la etiqueta nav para la barra de navegación y footer para el “footer”. Además, se han tenido en cuenta una serie de buenas prácticas:

- Uso de etiquetas de bloque semánticas, en vez de la etiqueta div.
- No usar etiquetas de bloque que agrupen un solo bloque.
- Se ha tenido en cuenta la universalidad de la página web:
 - Se han usado etiquetas meta informar sobre la codificación de la página web.
 - Se ha usado el atributo alt en las imágenes.
 - Se han usado etiquetas meta con palabras clave así como con una descripción del sitio web, con el objetivo de que la página web, en caso de ser pública en Internet, tuviese más facilidades para ganar visitas.
 - Al no usarse de manera explícita la etiqueta meta para incluir directivas para robots de búsqueda, se toma la directiva por defecto, “all”, la cual no impone restricciones en la indexación ni en la presentación de contenido.

El mapa de etiquetas de la página principal se muestra en la Figura 34 (en el anexo II). Cabe destacar:

- Uso de la etiqueta section para cada grupo de eventos del mismo tipo.
- Uso de la etiqueta article para cada evento.
- Uso de la etiqueta h1 para el título de la página y las sucesivas etiquetas h para subtítulos.
- Uso de las etiquetas form, label e input para formularios.

El resto de páginas comparten estas mismas características:

- Página “Sobre Nosotros” (Figura 35). En ella se usa la etiqueta video (propia de HTML 5) para incluir un vídeo.
- Página “Para Empresas” (Figura 36).
- Página “Para Tarifas” (Figura 37).
- Página “Para Testimonios” (Figura 38).

8. CSS: mapa de etiquetas

Con respecto al código CSS, se ha procurado no crear un número excesivo de clases o identificadores. En caso de que un etiqueta HTML fuese semántica (p. ej. header, article) se procuraba, siempre que fuese conveniente, usar un selector CSS por el tipo de etiqueta, evitando crear clases. A continuación se exponen diferentes consideraciones:

- Con el objetivo de facilitar la programación de características comunes, se ha creado el fichero /css/comun.css. En él hay diferentes variables (p. ej. colores), reglas para el botón azul que se usa en todas las páginas, o la regla para el fondo blanco.
- La barra de navegación y el footer, se reutilizaron en todas las páginas, por lo que se creó un fichero CSS separado para cada uno de ellos, y poder reutilizarlos.
- Se ha usado gradientes de color en muchas ocasiones, sobre todo para bordes de elementos, como en la página de tarifas por ejemplo.
- Para los texto sobre las imágenes, para hacer que el texto sea legible, se aplicaba un gradiente gris (con una cierta opacidad) a la imagen de fondo.
- Se ha tenido en cuenta la responsividad de casi absolutamente todas las páginas. La estrategia más usada ha sido convertir los display flex por fila en display flex por columna cuando la página se hacía más pequeña. En la Figura 16 se puede ver el código CSS asociado a la responsividad de la barra de navegación:

```
58  /* Header responsive */
59  @media(max-width: 900px) {
60
61    header {
62      justify-content: start;
63      gap: 0.25rem;
64    }
65
66    nav {
67      flex-direction: column;
68      gap: 0.5rem;
69    }
70
71    .nav-items {
72      flex-direction: column;
73      padding: 1.25rem;
74      gap: 0.5rem;
75      backdrop-filter: blur(10px);
76      border-radius: 20px;
77    }
78
79    .nav-items a:hover{
80      font-weight: bold;
81    }
82
83    .logo-nav {
84      font-size: 2rem;
85    }
86
87 }
```

Figura 16: CSS: Responsividad barra de navegación

- Se ha usado el efecto hover en algunas ocasiones, por ejemplo para el botón azul que se usa múltiples veces. Cuando el ratón se pone encima, cambia de color:

```

93   .btn-std:hover {
94     background-color: var(--amarillo);
95   }

```

Figura 17: CSS: Hover botón azul

El mapa de etiquetas CSS de la página principal se puede ver en la Figura 39. Lo más destacable de esta página, es el formulario, el cual aparece por encima de la página principal, quedando fijo. Para ello se usa CSS y JS. Con respecto al CSS (fichero /css/formReserva.css:

- Para el formulario se usa position:fixed, para que se queda en una posición fija en la pantalla aunque se haga scroll.
- Para el fondo se usa background-color: rgba(0, 0, 0, 0.6). Así, adquiere un color grisáceo con cierta opacidad.
- Se usa una animación de movimiento de 1 segundo, que hace que se desplace hacia la izquierda, Figura 18.

```

39  @-webkit-keyframes animacionInicio {
40    from {
41      left: -100px;
42      opacity: 0
43    }
44    to {
45      left: 0;
46      opacity: 1
47    }

```

Figura 18: CSS: Animación form página principal

El resto de página, no tiene ninguna característica CSS distintas a las ya comentadas. Se pueden ver sus diagramas de cajas en el Anexo III

- Página “Sobre Nosotros” (Figura 35).
- Página “Para Empresas” (Figura 36).
- Página “Para Tarifas” (Figura 37).
- Página “Para Testimonios” (Figura 38).

9. JavaScript

Para la realización de la página web, se ha usado Java Script en diferentes partes de la página web:

- /js/cargaEventos.js: Carga los eventos desde un fichero JSON mediante AJAX usando el método XMLHttpRequest. Lee los eventos en el fichero /json/eventos.json, y los inserta en el documento HTML index.html. Para ello hace uso de métodos JQuery.
- /js/cargarOpiniones.js: De un fichero XML, mediante AJAX, y usando el método load de JQuery, inserta la opinión destacada de los dos tipos de clientes (empresas y usuarios) en el documento HTML testimonios.html usando métodos de JQuery.
- /js/faq.js: Dota de dinamismo a las preguntas frecuentes de los documentos serviciosEmpresas.html y serviciosUsuarios.html. Hace que se despliegue o encarte la respuesta a una pregunta frecuente. Para ello, usa métodos de acceso al DOM.
- /js/form.js: Afecta a la página principal, y hace que cuando se clica el botón de reservar, se muestre el formulario. Usa métodos de acceso al DOM.
- /js/slidesEventos.js: Afecta a la página principal, al documento index.html. Mediante métodos de JQuery, permite cambiar los 4 eventos de la página principal (slider).
- /js/submitForm.js: Afecta a la página principal, al documento index.html, al formulario. Comprueba, mediante expresiones regulares que el correo electrónico sea correcto.
- /js/tarifas.js: Afecta al documento de tarifas.html. Mediante métodos de acceso al DOM, dota de dinamismo a la página haciendo que se pueda escoger el tipo de tarifas: mensuales o anuales.
- /js/testimonios.js: Mediante métodos de JQuery, se implementa el slider de las opiniones en la página de testimonios.

A continuación se hablará sobre los ficheros js más destacables.

9.0.1. Fichero /js/faq.js

El fichero JavaScript de la Figura 21 dota de dinamismo a las preguntas frecuentes de los documentos serviciosEmpresas.html y serviciosUsuarios.html. Hace que se despliegue o encarte la respuesta a una pregunta frecuente. Para ello, usa los siguientes métodos de acceso al DOM:

```

1 // Preguntas
2 const preguntas = document.querySelectorAll('.accordion button');
3
4 // Se añade el evento a todas las preguntas
5 preguntas.forEach((item) => item.addEventListener('click', toggleAccordion));
6
7 function toggleAccordion() {
8     const itemToggle = this.getAttribute('aria-expanded');
9     // Se cierran o encartan el resto de preguntas
10    for (item of preguntas) {
11        item.setAttribute('aria-expanded', 'false');
12    }
13    // Si no está expandida, se expande o abre la respuesta la pregunta actual
14    if (itemToggle == 'false') {
15        this.setAttribute('aria-expanded', 'true');
16    }
17}

```

Figura 19: Fichero JS de las FAQs. Usa métodos de acceso al DOM.

- `querySelectorAll()`, en la línea 1, para seleccionar todos aquellos elementos del DOM según el selector pasado como parámetro. Devuelve un `NodeList`.
- `getAttribute()`, en la línea 8, para obtener el valor del atributo “`aria-expanded`”, y así ver si la respuesta de la FAQ está desplegada o no.
- `setAttribute()`, en las líneas 11 y 14, modificar el atributo “`aria-expanded`” y hacer que la respuesta se contraiga o expanda.

9.0.2. Fichero /js/form.js

Afecta a la página principal, y hace que cuando se clica el botón de reservar, se muestre el formulario. (Ver Figura 20. En el código se ve como se añade un evento para que cuando se clique el botón de cerrar, o se cliente fuera del formulario, este se cierre. Además, cuando se clica el botón de reservar de la página principal, se muestra el formulario.

```

1 let formulario = document.getElementById("form-reserva");
2 let btnCerrar = document.getElementsByClassName("close")[0];
3
4 // Si se clica el btn de reservar se muestra el formulario
5 function reservar(btn) {
6     formulario.style.display = "flex";
7 }
8
9 //Si se clica el btn de cerrar se cierra el formulario
10 btnCerrar.onclick = function () {
11     formulario.style.display = "none";
12 }
13
14 // Si se clica fuera del formulario, se cierra el formulario
15 window.onclick = function (event) {
16     if (event.target == formulario) {
17         formulario.style.display = "none";
18     }
19 }

```

Figura 20: Fichero JS formulario página principal

- Método getElementById para seleccionar el formulario.
- Método getElementsByClassName para seleccionar el botón de cerrar. Como este método puede devolver varios elementos, cogemos el primero (sabemos que solo hay uno).

9.0.3. Fichero /js/tarifas.js

El fichero JavaScript de la Figura 21 dota de dinamismo a las tarifas del documento HTML tarifas.html. Así, a través de un botón (input de tipo checkbox en HTML), se pueden mostrar los precios mensuales o anuales. En las primeras líneas de este fichero, se obtienen los botones, y se les añade un listener para el evento click. También se obtienen los textos que aparecen a la derecha del precio poniendo “mensual” o “anual”. Con respecto al método cambioModalidad, este se encarga de cambiar dichos textos, así como de cambiar el precio. Se han usado los siguientes métodos de acceso al DOM:

```

1 // Botones de mensual anual
2 const btnEmpresas = document.getElementById("btn-mensual-anual-empresas");
3 const btnUsuarios = document.getElementById("btn-mensual-anual-usuarios");
4
5 // Se añaden eventos
6 btnEmpresas.addEventListener("click", cambiarModalidadPagoEmpresas);
7 btnUsuarios.addEventListener("click", cambiarModalidadPagoUsuarios);
8
9 // Textos de año/mes en las tarifas
10 const textosMensualAnualEmpresas = document.querySelectorAll("#planes-organizadores .modalidad-pago");
11 const textosMensualAnualUsuarios = document.querySelectorAll("#planes-usuarios .modalidad-pago");
12
13 // Cte para los precios
14 const cte = 9;
15
16 function cambiarModalidadPagoEmpresas() {
17     cambioModalidad(textosMensualAnualEmpresas, btnEmpresas);
18 }
19
20 function cambiarModalidadPagoUsuarios() {
21     cambioModalidad(textosMensualAnualUsuarios, btnUsuarios);
22 }
23
24 function cambioModalidad(textos, btn){
25     if (btn.checked == true) {
26         //Anual
27         for (e of textos) {
28             e.innerText = "/año";
29             const precio = parseInt(e.parentNode.firstChild.textContent);
30             e.parentNode.firstChild.textContent = Math.trunc(precio * cte);
31         }
32     } else {
33         // Mensual
34         for (e of textos) {
35             e.innerText = "/mes";
36             const precio = parseInt(e.parentNode.firstChild.textContent);
37             e.parentNode.firstChild.textContent = Math.trunc(precio / cte);
38         }
39     }
40 }

```

Figura 21: Fichero JS de las FAQs. Usa métodos de acceso al DOM.

- Método getElementById para seleccionar los botones por su id (uno para las tarifas de empresas y otra para las de usuarios).
- Método querySelectorAll para seleccionar los textos de las tarifas que ponen “mensual” o “anual” a la derecha del precio.
- Propiedad parentNode, para obtener el nodo padre del nodo actual.
- Propiedad firstChild, para obtener el hijo padre del nodo actual.

9.0.4. Fichero /js/slideEventos.js

Afecta a la página principal, e implementa el slider de los eventos (Figure 22). Lo más destacable del código, es el uso de JQuery para ocultar y mostrar eventos. La limitación de esta implementación, es que el slider no es circular, sino que los eventos permanecen fijos en el DOM,

entonces, si por ejemplo hay 7 eventos en el DOM, y se están mostrando los 4 últimos, al clicar la flecha derecha, se mostrarán, en este orden, 1, 5, 6, 7.

```

1  $(".prev, .next").on("click", function (e) {
2    |  cambiarContenido(e);
3  );
4
5  // Devuelve el nuevo indice
6  function siguienteIndex(i, n, sentido) {
7    let nuevoValor = i + sentido;
8    nuevoValor = nuevoValor < 0 ? n-1 : nuevoValor;
9    return nuevoValor % n;
10 }
11
12 function cambiarContenido(e) {
13   // Se recatan los contenedores y elementos correspondientes a los eventos
14   const target = e.target;
15   const container = $(target).parent().find(".caja_eventos")[0];
16   const eventos = $(container).children();
17   // Datos relacionados con el indice
18   const sentido = parseInt($(target).attr("class") == "prev" ? -1 : 1);
19   let i = parseInt($(target).parent().find(".indexActual").text());
20   const n = eventos.length;
21   // Se actualiza el indice actual
22   i = siguienteIndex(i, n, sentido);
23   $(target).parent().find(".indexActual").text(i);
24   // Se ocultan eventos
25   eventos.each(function () {
26     |  $(this).hide();
27   })
28   // Array auxiliar para ver conseguir los indices de los elementos a mostrar
29   const mostrarSet = Array.from({ length: 4 }, (_, j) => siguienteIndex(j + i - 1, n, 1));
30   console.log(mostrarSet);
31   // Se muestran los eventos oportunos
32   eventos.each(function (index) {
33     |  if (mostrarSet.includes(index)) $(this).show();
34   });
35
36 }

```

Figura 22: Fichero JS slider página principal

- Método getElementById para seleccionar el formulario.
- Método getElementsByClassName el botón de cerra. Como este método puede devolver varios elementos, cogemos el primero (sabemos que solo hay uno).

9.1. Métodos de acceso al DOM

La práctica pedía un mínimo de 5 métodos diferentes de acceso al DOM. Estos se han incluido en diferentes ficheros js:

- Método querySelectorAll, por ejemplo en el fichero/js/faq.js, para obtener todas las preguntas frecuentes. Devuelve un NodeList.
- Método getAttribute, en el fichero/js/faq.js, para obtener el valor del atributo “aria-expanded”, y así ver si la respuesta de la FAQ está desplegada o no.
- Método setAttribute, en el fichero/js/faq.js, modificar el atributo “aria-expanded” y hacer que la respuesta se contraiga o expanda.

- Método getElementById, por ejemplo en el fichero /js/tarifas.js, para seleccionar los botones de las tarifas por su id (uno para las tarifas de empresas y otra para las de usuarios).
- Método getElementsByClassName, en el fichero /js/form.js para seleccionar el botón de cerrar.
- Propiedad parentNode, en el fichero /js/faq.js, para obtener el nodo padre del nodo actual.
- Propiedad firstChild, en el fichero /js/faq.js, para obtener el hijo padre del nodo actual.
- Método querySelector, en el fichero /js/submit.from, para obtener el campo de entrada del email en el formulario del documento HTML index.html. Si hubiese varios elementos que cumpliese la regla del selector, solo devolvería el primero, a diferencia de querySelectorAll.

9.2. Respuesta a eventos

La práctica exige responder, al menos, a 3 eventos diferentes.

9.2.1. Evento click

Se responde a él en numerosas ocasiones, por ejemplo en la Figura 21 del fichero /js/tarifas.js. En las líneas 6 y 7 se ve como se añade un listener a los botones, para que cambiar del precio de mensual a anual y viceversa.

9.2.2. Evento submit

Se responde a él en el formulario de reserva de entradas, ver Figura 23. Cuando se hace submit, se comprueba la expresión regular del correo electrónico, y si es correcta, se finaliza la reserva. Por contra, si es incorrecta, se muestra un mensaje de error.

```

1 const email = document.querySelector('input[name="email"]');
2
3 // Expresión regular correo
4 const regexEmail = /^[a-zA-Z0-9._!#$%&^*+=?^`{|}~-]+@[a-zA-Z0-9-]+(?:\.[a-zA-Z0-9-]+)*$/i;
5
6 // Listener eventos
7 document.getElementById("form-reserva").addEventListener("submit", submitReserva);
8 document.addEventListener("keydown", (event) => {
9   if (event.key === "Escape") {
10     btnCerrar.click();
11   }
12 });
13
14 function submitReserva(e) {
15   if (regexEmail.test(email.value)) {
16     alert("Se ha enviado la petición de reserva de entradas. En breves recibirán un email para realizar el pago de las mismas.");
17   } else {
18     alert("El email introducido no es correcto. Introduzca uno correcto");
19     e.preventDefault();
20   }
21 }
22
23 function exitForm() {
24   btnCerrar.click();
25 }
```

Figura 23: Fichero JS /js/submitForm.js. Hay una expresión regular, y responde a eventos.

9.2.3. Evento keyDown

Se responde a él en el formulario de reserva de entradas, ver Figura 23. Cuando se pulsa la tecla de escape del teclado, se cierra el formulario. Como dato destacable, este evento no es equivalente a keyPress, ya que este último, además de estar obsoleto, no responde a teclas como escape, alt o shift, sino que solo a aquellas que producen un carácter alfanumérico o signos de puntuación.

9.3. Objetos JQuery

Se ha usado la librería JQuery en múltiples ocasiones a lo largo de la práctica, una de ellas, en el fichero /js/cargaEventos.js.

```
const container = $('#' + idHTML + ' .caja_eventos');
container.append(code);
const eventosHide = $(container).children();
eventosHide.filter(':gt(' + (numberVisibleItems-1)+)').hide();
```

Figura 24: Carga de eventos desde un fichero JSON. Se usan, además, métodos JQuery.

El código de la Figura 24, es una pequeña parte del mostrado en la Figura 26, en la sección de JSON. Como se puede ver en el código de la Figura 24, se usa las funciones JQuery \$(), append y children para el acceso y modificación del DOM. En la última línea se ocultan todos los nodos excepto los 4 primeros. Para ello, se usan:

- gt: Selecciona los nodos cuyo índice sean mayor que un cierto número.
- filter: Selecciona los nodos que cumplen la condición ':gt(' + (numberVisibleItems-1)+')'
- hide: Oculta un elemento.

9.4. Uso de JS6

Se han usado características propias de JavaScript 6 en diversas ocasiones:

- Declaración de variables: let y const.
- Uso de bucles for in y of.
- Funciones Arrow

En el fichero /js/slideEventos, se ha usado la función Arrow de la Figura 25.

```
(_, j) => siguienteIndex(j + i - 1, n, 1)
```

Figura 25: Función Arrow

9.5. Uso de expresiones regulares

Se han utilizado expresiones regulares para la comprobación de los datos introducidos en el formulario del archivo index.html. Dicho formulario se usa para reservar entradas de un evento. Hay dos formas de comprobar expresiones regulares:

- Mediante el atributo pattern de HTML. Mediante este método se ha comprobado la entrada de los apellidos, del nombre, y el DNI.
- Mediante JS con funciones test y exec. Se ha usado para el correo electrónico, y su código del fichero /js/submit.js se puede ver en la Figura 23. Este código hace que cuando se clique el botón de reservar, se comprueba si el correo electrónico introducido es correcto, comprobando si el correo introducido por el usuario cumple la expresión regular adecuada. Para eso se ha usado el método test. Si es correcto, se muestra un diálogo con un mensaje de aceptación, y se cierra el formulario. Por contra, si es incorrecto, se muestra un mensaje de error, pero no se cierra el formulario.

9.6. Carga contenido en formato JSON

En la página principal, se ha optado por cargar la información de los eventos (nombre del evento, fecha, lugar, precio y dirección de la imagen) a partir de un archivo en formato JSON mediante XMLHttpRequest. Hay dos motivos principales por los que es conveniente cargar estos datos de un fichero:

- Se reduce notablemente el código HTML generado, dando lugar a código más mantenible, y aumentando la escalabilidad del mismo. Es importante notar, que si no cargásemos los datos de esta forma, tendríamos, en el caso de tener 10 eventos, 10 bloques de código exactamente HTML iguales.
- Los eventos son información que pueden variar dinámicamente.

El código correspondiente a esta carga de datos está en la ruta /js/cargaEventos.js.

```

1 const fileName = "eventos.json";
2 const filePath = "http://127.0.0.1:5500/json/" + fileName;
3 const numberVisibleItems = 4;
4
5 function cargarEventos() {
6     let xhttp = new XMLHttpRequest();
7     xhttp.onload = function () {
8         if (this.readyState == 4 && this.status == 200) {
9             let json = JSON.parse(this.responseText);
10            cargaEventos("eventos-destacados",json.eventosDestacados);
11            cargaEventos("eventos-deportivos",json.eventosDeportivos);
12            cargaEventos("eventos-musicales",json.eventosMusicales);
13            cargaEventos("eventos-sociales",json.eventosSociales);
14        }
15    }
16    xhttp.open("GET", filePath);
17    xhttp.send();
18 }
19
20 function cargaEventos(idHTML, eventos){
21     let code = '';
22     for (e of eventos) {
23         code += '<article class="evento"> <div class="container_imagen_evento"> </div><div class="co';
26         code += e.nombreEvento;
27         code += '</h3><p class="fecha-evento"> ';
28         code += e.fecha;
29         code += '</p><p class="lugar-evento"> ';
30         code += e.lugar;
31         code += '</p><div class="footer-evento"><p class="container-precio"> Desde <span cla';
32         code += e.precio;
33         code += '</span></p><p class="btn-std" onclick="reservar(this)"> Reservar</p></div>';
34     }
35     const container = $('#' + idHTML + ' .caja_eventos');
36     container.append(code);
37     const eventosHide = $(container).children();
38     eventosHide.filter(':gt(' + ( numberVisibleItems-1)+')').hide();
39 }
40
41 cargarEventos();

```

Figura 26: Carga de eventos desde un fichero JSON mediante XMLHttpRequest. Se usan, además, métodos JQuery.

El código JSON correspondiente se muestra en la Figura 26. En él la función cargarEventos utiliza el objeto XMLHttpRequest de AJAX para obtener los datos del fichero haciendo una petición de tipo get mediante el método open. Con no se le ha añadido un tercer parámetro, se realiza de manera asíncrona por defecto. Se usa el método send para enviar la petición al servidor.

Se define el atributo onload del objeto XMLHttpRequest como una función callback que será ejecutada cuando la petición sea recibida. Esta función parsea el documento HTML, e invoca a la función auxiliar cargaEventos tantas veces como tipo de eventos tenemos en la página principal (destacados, deportivos, sociales, y musicales).

La función auxiliar cargaEventos se encarga de generar una cadena de texto con el código HTML

necesario, añadiéndole los datos de cada evento del fichero JSON. Mediante métodos JQuery se añade este nuevo código al DOM. Con respecto al uso de métodos JQuery, se comentará en su sección correspondiente.

9.7. Carga contenido en formato XML

En la página de testimonios (testimonios.html), se ha decidido cargar el comentario principal tanto de la sección de clientes como de usuarios desde un fichero XML mediante JQuery. Para ello, como se ve en la Figura 27, se invoca la función load con el selector adecuado en función de la información que queramos cargar. En este caso, cargaremos el nombre del cliente, y su comentario, tanto para la sección de usuarios como de empresas.

```
1  const fileName = "opiniones.xml";
2  const filePath = "http://127.0.0.1:5500/xml/" + fileName;
3
4  $(document).ready(function(){
5    // Empresas
6    $("#opiniones-empresas .presentacion p").load(filePath + " " + "empresa nombre");
7    $("#opiniones-empresas blockquote").load(filePath + " " + "empresa opinion");
8    // Usuarios
9    $("#opiniones-usuarios .presentacion p").load(filePath + " " + "usuario nombre");
10   $("#opiniones-usuarios blockquote").load(filePath + " " + "usuario opinion");
11 })
```

Figura 27: Carga de eventos desde un fichero XML. Mediante JQuery.

A. Anexo I: Diseños finales.

The screenshot displays the final design of the Fun Coruña website's homepage. The top navigation bar includes links to "Fun Coruña", "Sobre nosotros", "Para empresas", "Para usuarios", "Tarifas", and "Testimonios". Below the header, the title "Fun Coruña" is prominently displayed, followed by the tagline "Tu mejor buscador de eventos en A Coruña". The main content area is organized into four sections:

- Eventos destacados**: Shows four cards for matches between Dépor and other teams: Badajoz, Celta B, Mérida, and Rayo. Each card includes the date (25 abril 2022), time (19:00), stadium (Estadio Riazor), price (Desde 20€), and a "Reservar" button.
- Eventos deportivos**: Shows four cards for the same matches as the first section, with identical details.
- Eventos musicales**: Shows four cards for the same matches as the first section, with identical details.
- Eventos sociales**: Shows four cards for the same matches as the first section, with identical details.

At the bottom of the page, there is a footer section with the "Fun Coruña" logo, a brief legal notice about copyright, and links to "Sobre nosotros", "Para empresas", "Para usuarios", "Tarifas", and "Testimonios".

Figura 28: Diseño final página principal

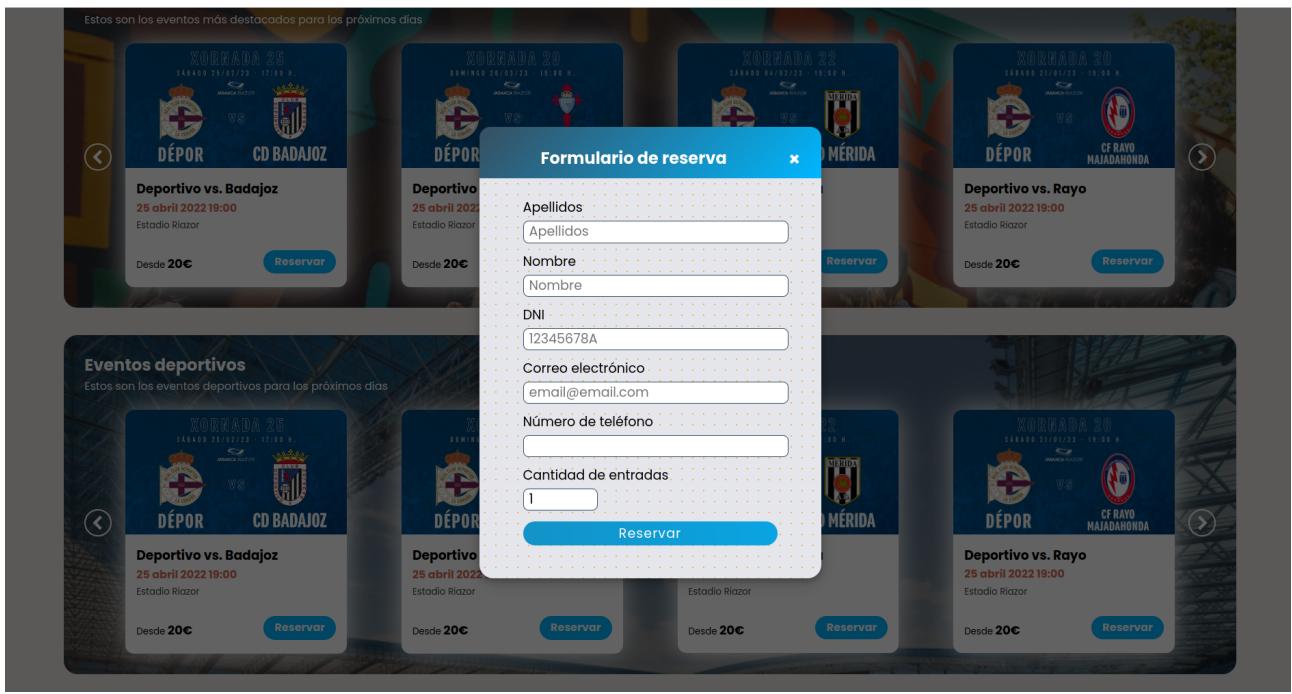


Figura 29: Diseño final formulario de la página principal

Fun Coruña Sobre nosotros Para empresas Para usuarios Tarifas Galería Testimonios

Fun Coruña

Lorem ipsum dolor sit amet consectetur adipisciing elit. Magni, totam suscipit? Sunt, exercitationem dignissimos voluptas sint voluptatum minus facilis pariatur praesentium voluptatibus quod erit suscipit tempore dolorum nobis consequatur cupiditate.

[Conócenos](#)

Sobre nosotros

Transformamos tus sueños en eventos inolvidables.

Somos una empresa especializada en la promoción, creación y gestión de eventos sociales, culturales, musicales y deportivos en la ciudad de A Coruña. Contamos con un equipo altamente capacitado y una amplia red de contactos en la industria para asegurarnos de que cada evento sea un éxito.

Historia

Desde 2015

Lorem ipsum dolor, sit amet consectetur adipisciing elit. Quis enim esse laudantium quam voluptates sed soepe distinctio a expedita aperiam. Rerum, ut. Eum, neque tenetur nisi possimus sequi corrupti alias.

Sobre nosotros

Transformamos tus sueños en eventos inolvidables.

Somos una empresa especializada en la promoción, creación y gestión de eventos sociales, culturales, musicales y deportivos en la ciudad de A Coruña. Contamos con un equipo altamente capacitado y una amplia red de contactos en la industria para asegurarnos de que cada evento sea un éxito.

2.4M
Clientes satisfechos

15.2k
Eventos promocionados

9
Premios a nivel nacional

Fun Coruña

Lorem ipsum dolor sit amet consectetur, adipisciing elit. Odit reprehenderit iusto accusamus? Eaque, provident nemo. Quam qui illum odio preferendis. Voluptate omnis inventore soepe nobis iusto temporibus magni illos laboriosam!

Copyright © 2023 - All rights reserved

Figura 30: Diseño final sobre nosotros

Fun Coruña Sobre nosotros Para empresas Para usuarios Tarifas Testimonios

Funcionamiento empresas

Descubre los servicios que podemos prestar a tu empresa

[Descubre nuestros servicios](#)

Damos a conocer tus eventos

Fun Coruña se encarga de dar a conocer y promocionar tus eventos, de tal manera que llega al mayor número de clientes potenciales.

Damos a conocer tus eventos

Fun Coruña se encarga de dar a conocer y promocionar tus eventos, de tal manera que llega al mayor número de clientes potenciales.

Damos a conocer tus eventos

Fun Coruña se encarga de dar a conocer y promocionar tus eventos, de tal manera que llega al mayor número de clientes potenciales.

Damos a conocer tus eventos

Fun Coruña se encarga de dar a conocer y promocionar tus eventos, de tal manera que llega al mayor número de clientes potenciales.

Damos a conocer tus eventos

Fun Coruña se encarga de dar a conocer y promocionar tus eventos, de tal manera que llega al mayor número de clientes potenciales.

Damos a conocer tus eventos

Fun Coruña se encarga de dar a conocer y promocionar tus eventos, de tal manera que llega al mayor número de clientes potenciales.

Damos a conocer tus eventos

Fun Coruña se encarga de dar a conocer y promocionar tus eventos, de tal manera que llega al mayor número de clientes potenciales.

Preguntas frecuentes

¿Qué tipos de eventos puede organizar Fun Coruña? (+) (−)

¿Cómo puedo contratar los servicios de Fun Coruña? (+) (−)

Puedes ponerte en contacto con nosotros a través de nuestro sitio web o por teléfono. Uno de nuestros representantes se pondrá en contacto contigo para discutir los detalles de tu evento y elaborar un presupuesto personalizado. (+) (−)

¿Qué servicios ofrece Fun Coruña? (+) (−)

Fun Coruña

LinkedIn | Facebook | Instagram | YouTube

Sobre nosotros Para empresas Para usuarios Tarifas Testimonios

Copyright © 2023 - All rights reserved

Figura 31: Diseño final servicios para empresas (análogo servicios para usuarios)

Fun Coruña Sobre nosotros Para empresas Para usuarios Tarifas Testimonios

Descubre nuestras tarifas

Planes empresas **Planes usuarios**

Organizadores y empresas

Mensual Anual

Tarifa	Precio	Planes
Básico	99.99€ /mes	Planes empresas
Pro	179.99€ /mes	Planes empresas
Premium	329.99€ /mes	Planes empresas

Básico
Perfecto para empresas que buscan un servicio económico pero aún así de alta calidad.

- ✓ Promoción en las redes sociales.
- ✓ Asistencia en la venta de entradas y registro de participantes.
- ✓ Un equipo de especialistas en eventos que trabajará con la empresa para crear y gestionar el evento.

Pro
Para empresas que desean un mayor nivel de atención y personalización.

- ✓ Promoción en las redes sociales.
- ✓ Asistencia en la venta de entradas y registro de participantes.
- ✓ Un equipo de especialistas en eventos que trabajará con la empresa para crear y gestionar el evento.
- ✓ Promoción adicional en redes sociales y marketing por correo electrónico dirigido a nuestra base de datos de contactos.
- ✓ Análisis post-evento y un informe de evaluación.

Premium
A single license for solo designers, developers and freelancers.

- ✓ Promoción en las redes sociales.
- ✓ Asistencia en la venta de entradas y registro de participantes.
- ✓ Un equipo de especialistas en eventos que trabajará con la empresa para crear y gestionar el evento.
- ✓ Promoción adicional en redes sociales y marketing por correo electrónico dirigido a nuestra base de datos de contactos.
- ✓ Análisis post-evento y un informe de evaluación.
- ✓ Publicidad en medios impresos y digitales seleccionados.
- ✓ Cobertura fotográfica y videográfica premium del evento.

PELICANO
Sala Pelicano

Fun Coruña ha mejorado la expansión de mi empresa.

Usuarios

Mensual Anual

Tarifa	Precio	Planes
Básico	7.99€ /mes	Planes usuarios
Pro	15.99€ /mes	Planes usuarios
Premium	25.99€ /mes	Planes usuarios

Básico
Perfecto para usuarios que buscan un servicio económico pero aún así de alta calidad.

- ✓ Notificaciones sobre eventos relevantes.
- ✓ Asistencia al cliente básica para preguntas y problemas técnicos.
- ✓ Acceso prioritario a eventos públicos y gratuitos en la ciudad.

Pro
Para usuarios que desean un mayor nivel de atención y personalización.

- ✓ Notificaciones sobre eventos relevantes.
- ✓ Asistencia al cliente básica para preguntas y problemas técnicos.
- ✓ Acceso prioritario a eventos públicos y gratuitos en la ciudad.
- ✓ Acceso exclusivo a pre-ventas y eventos con descuento.
- ✓ Recomendaciones personalizadas de eventos y actividades en función del historial de reservas.

Premium
A single license for solo designers, developers and freelancers.

- ✓ Notificaciones sobre eventos relevantes.
- ✓ Asistencia al cliente básica para preguntas y problemas técnicos.
- ✓ Acceso prioritario a eventos públicos y gratuitos en la ciudad.
- ✓ Acceso exclusivo a pre-ventas y eventos con descuento.
- ✓ Recomendaciones personalizadas de eventos y actividades en función del historial de reservas.
- ✓ Eventos privados y exclusivos.

Inés

Fun Coruña me ha hecho descubrir muchos eventos, y me ha facilitado la reserva de entradas.

Fun Coruña
Lorem ipsum dolor sit amet consectetur, adipisicing elit. Odit reprehenderit iusto accusamus? Eaque, provident nemo. Quam qui illum odio perferendis. Voluptate omnis inventore saepe nobis iusto temporibus magni alias laboriosam!
Copyright © 2023 - All rights reserved

Figura 32: Diseño final tarifas

Figura 33: Diseño final testimonios

B. Anexo II: HTML mapas de etiquetas

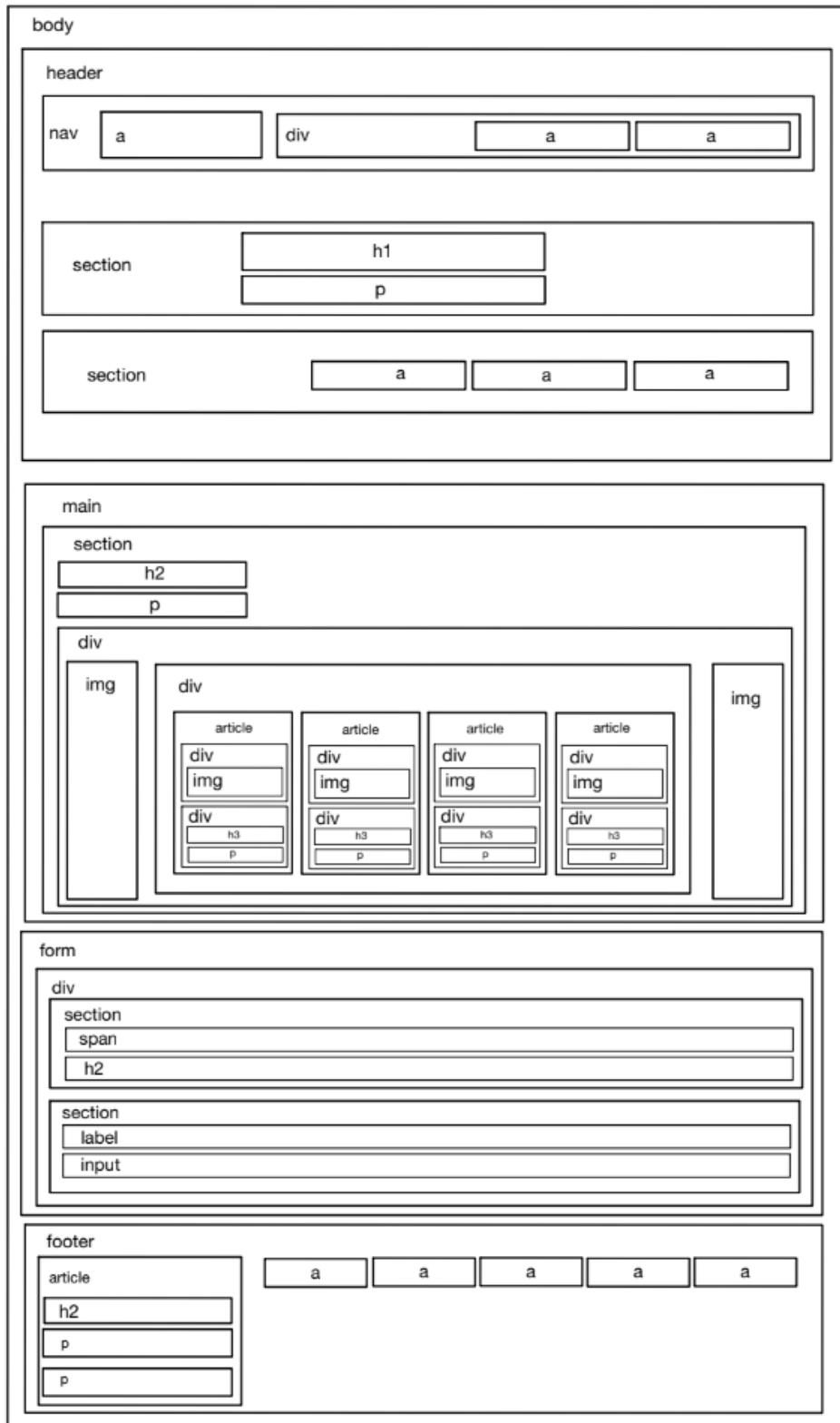


Figura 34: HTML: Mapa etiquetas página principal (incluye formulario)

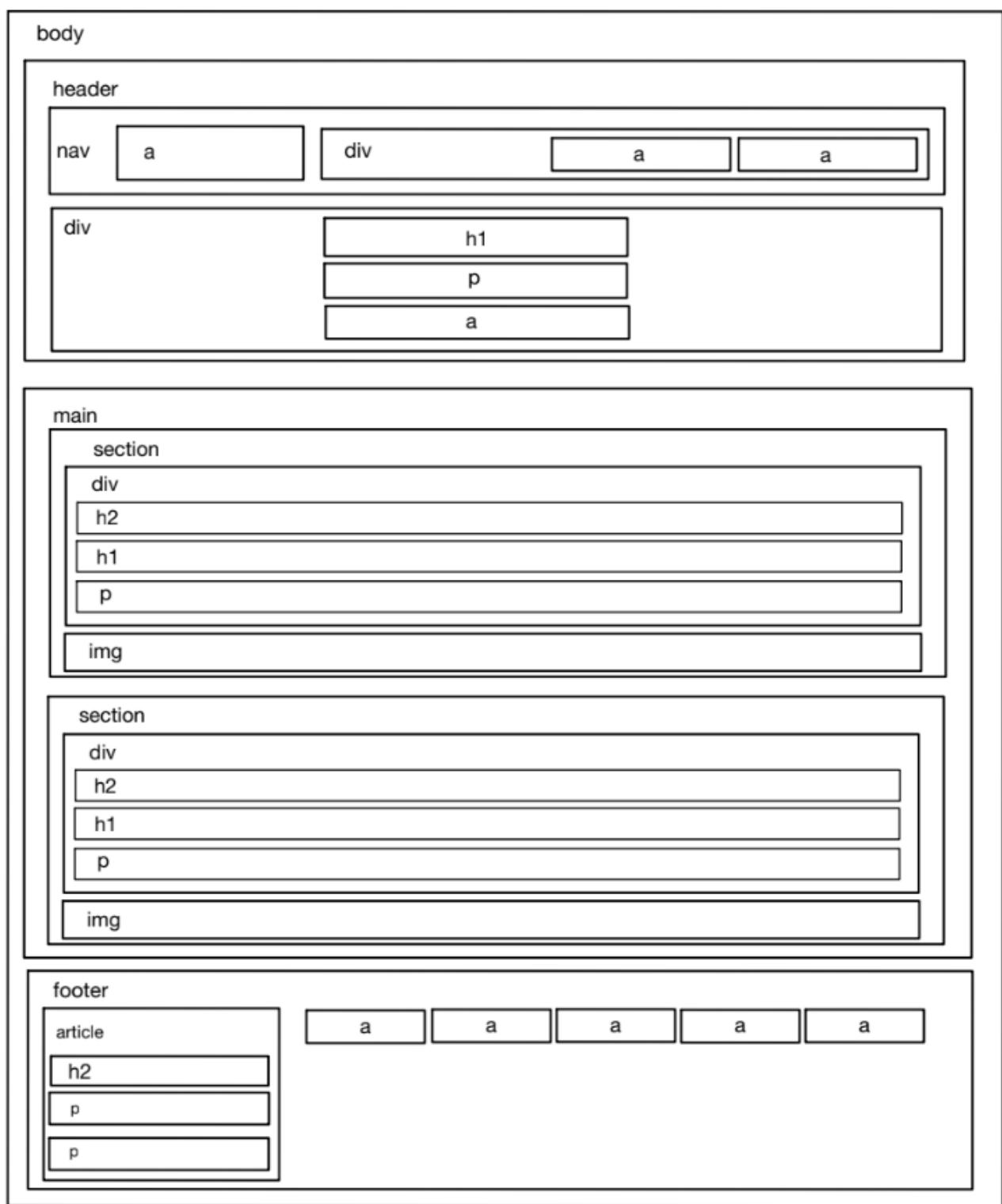


Figura 35: HTML: Mapa etiquetas página “Sobre nosotros”

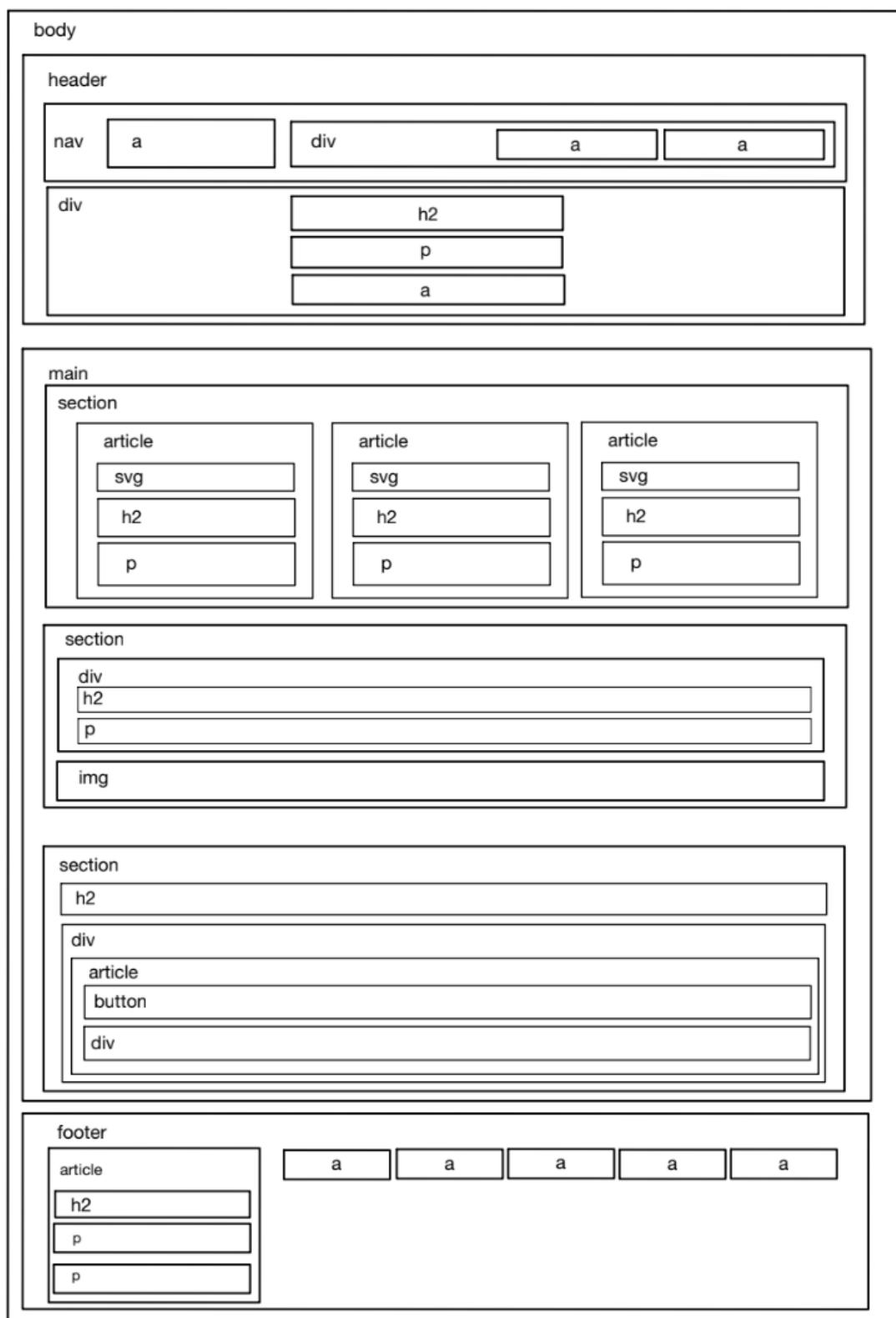
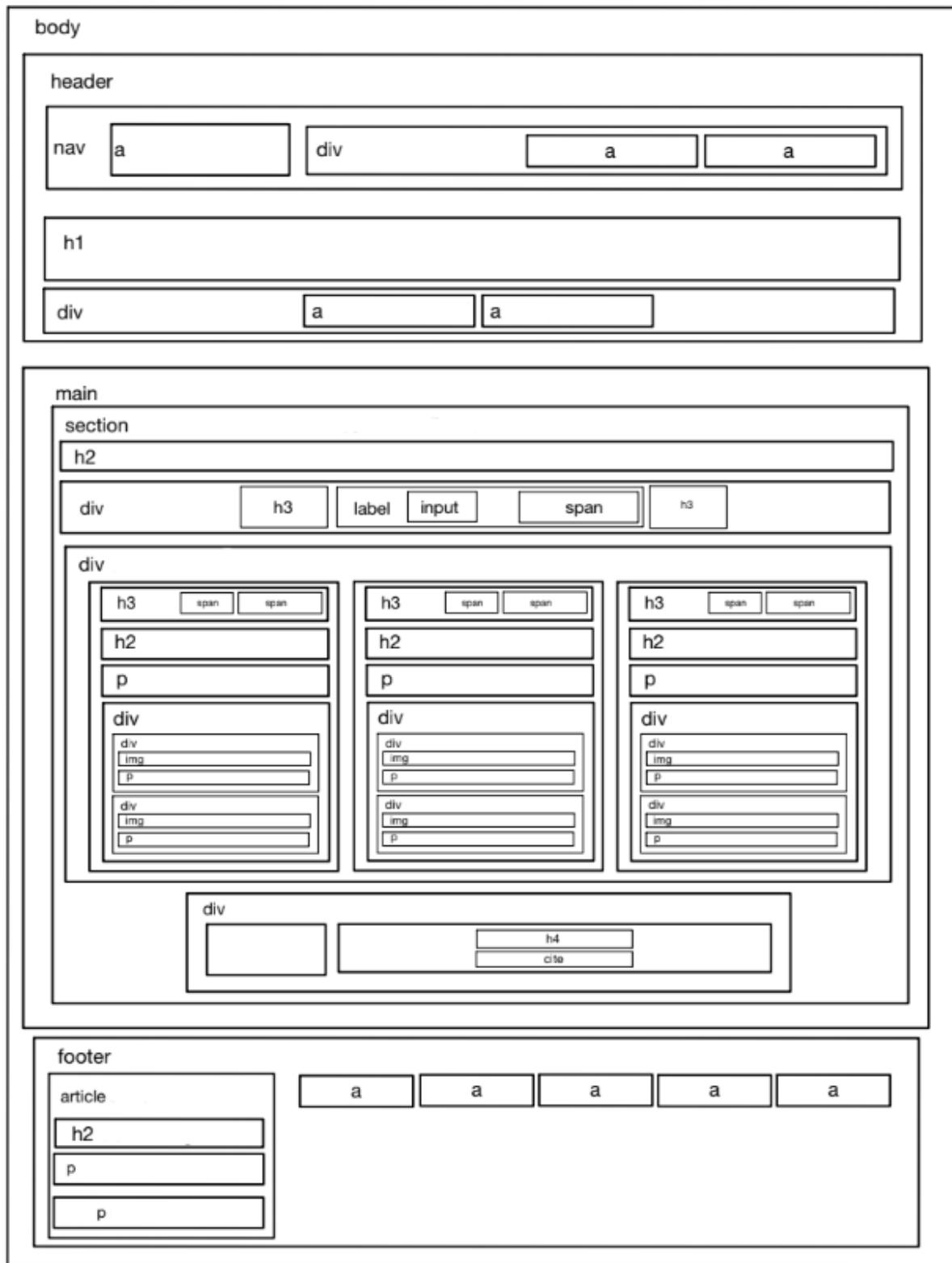


Figura 36: HTML: Mapa etiquetas página “Para empresas” (análogo para usuarios)



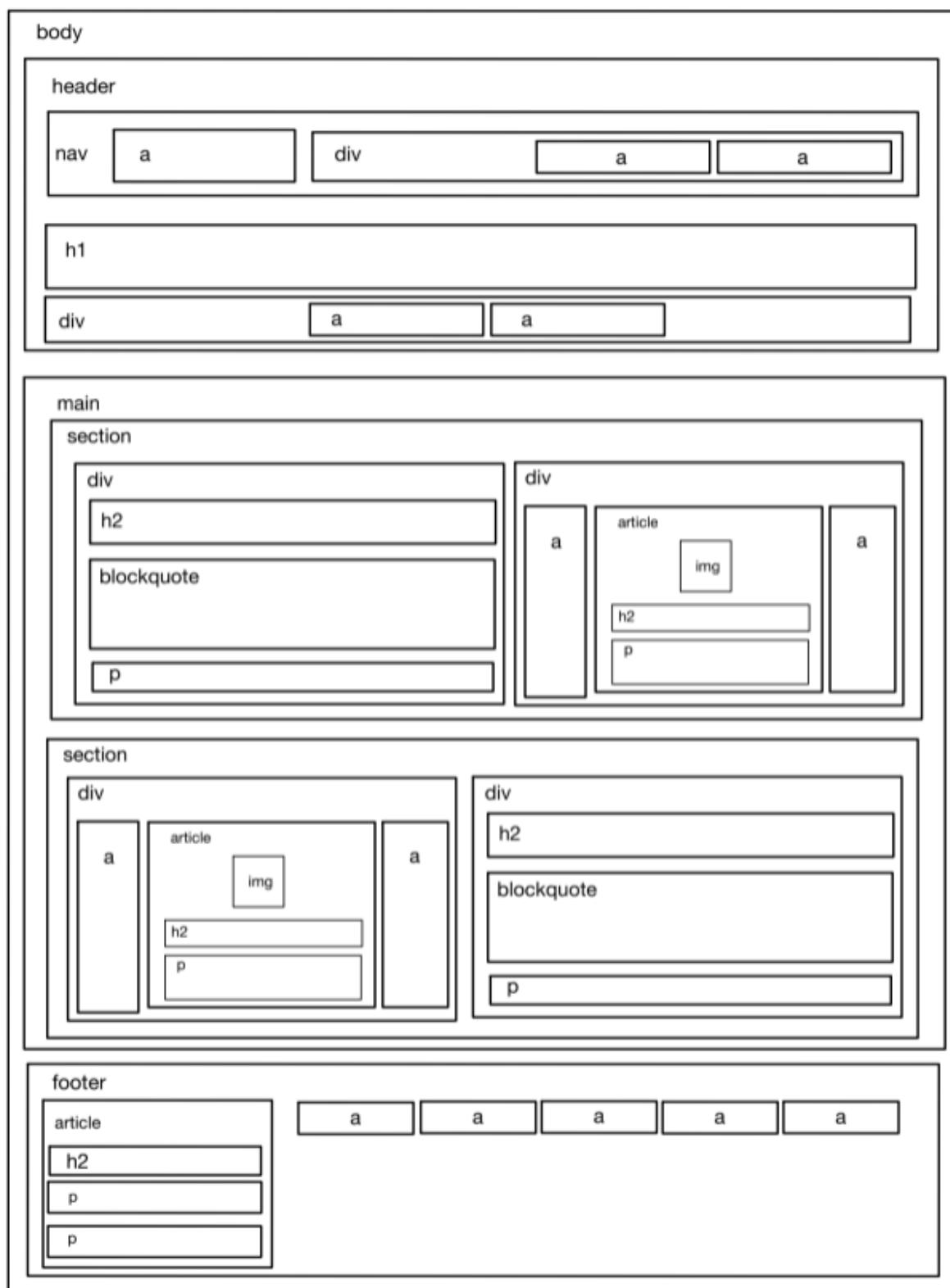


Figura 38: HTML: Mapa etiquetas página “Testimonios”

C. Anexo III: CSS mapas de etiquetas

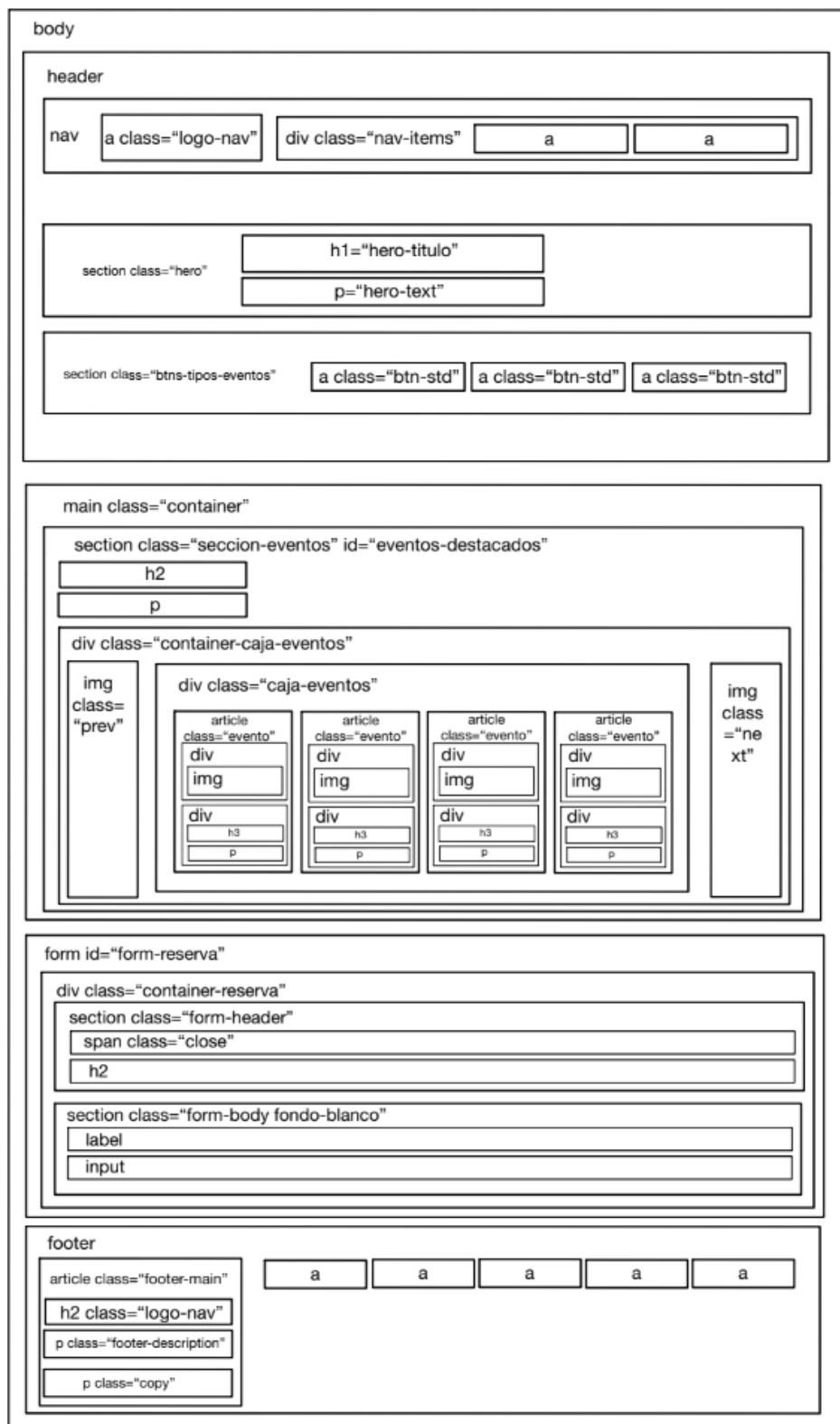


Figura 39: CSS: Mapa etiquetas página principal (incluye formulario)

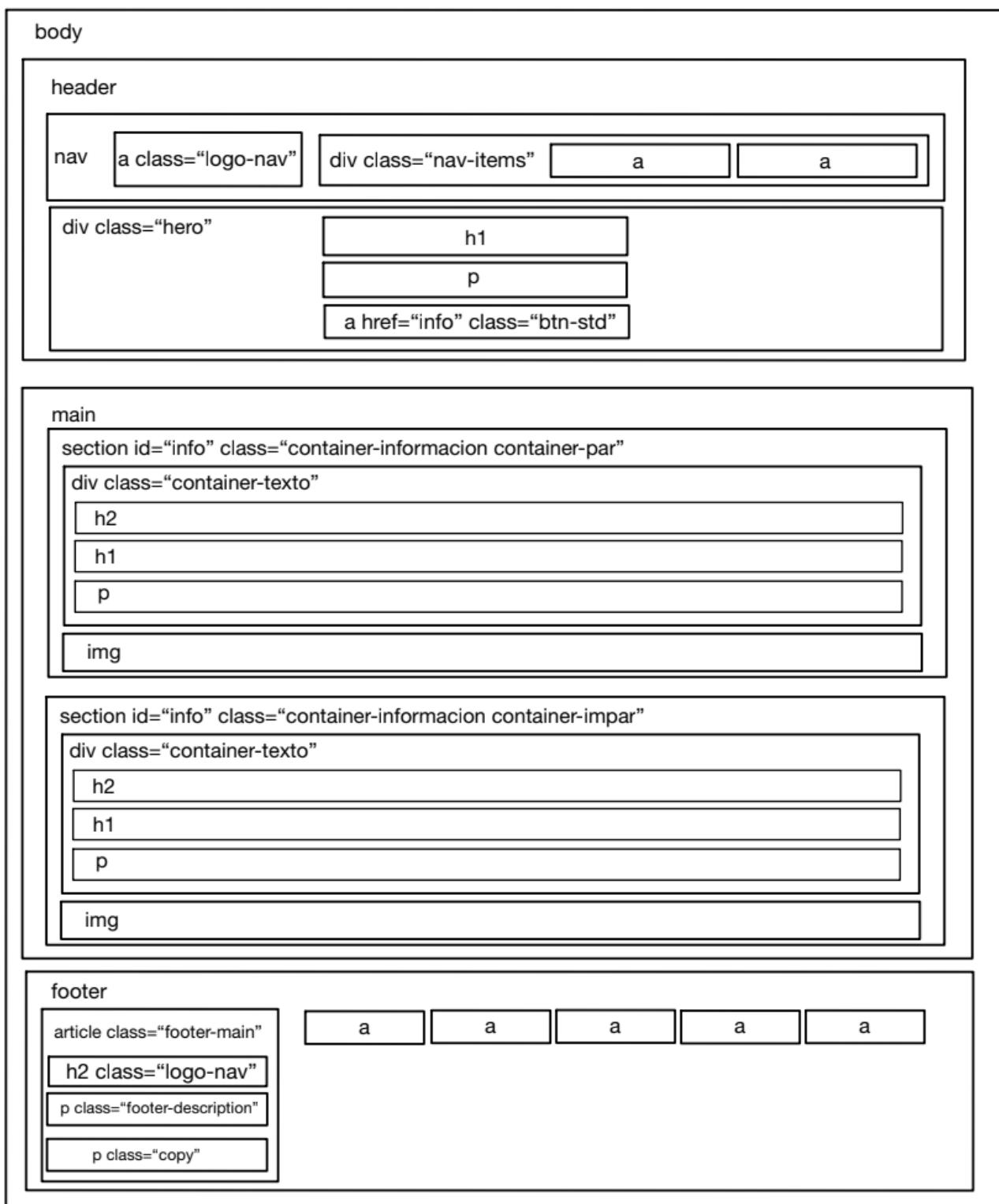


Figura 40: CSS: Mapa etiquetas página “Sobre nosotros”

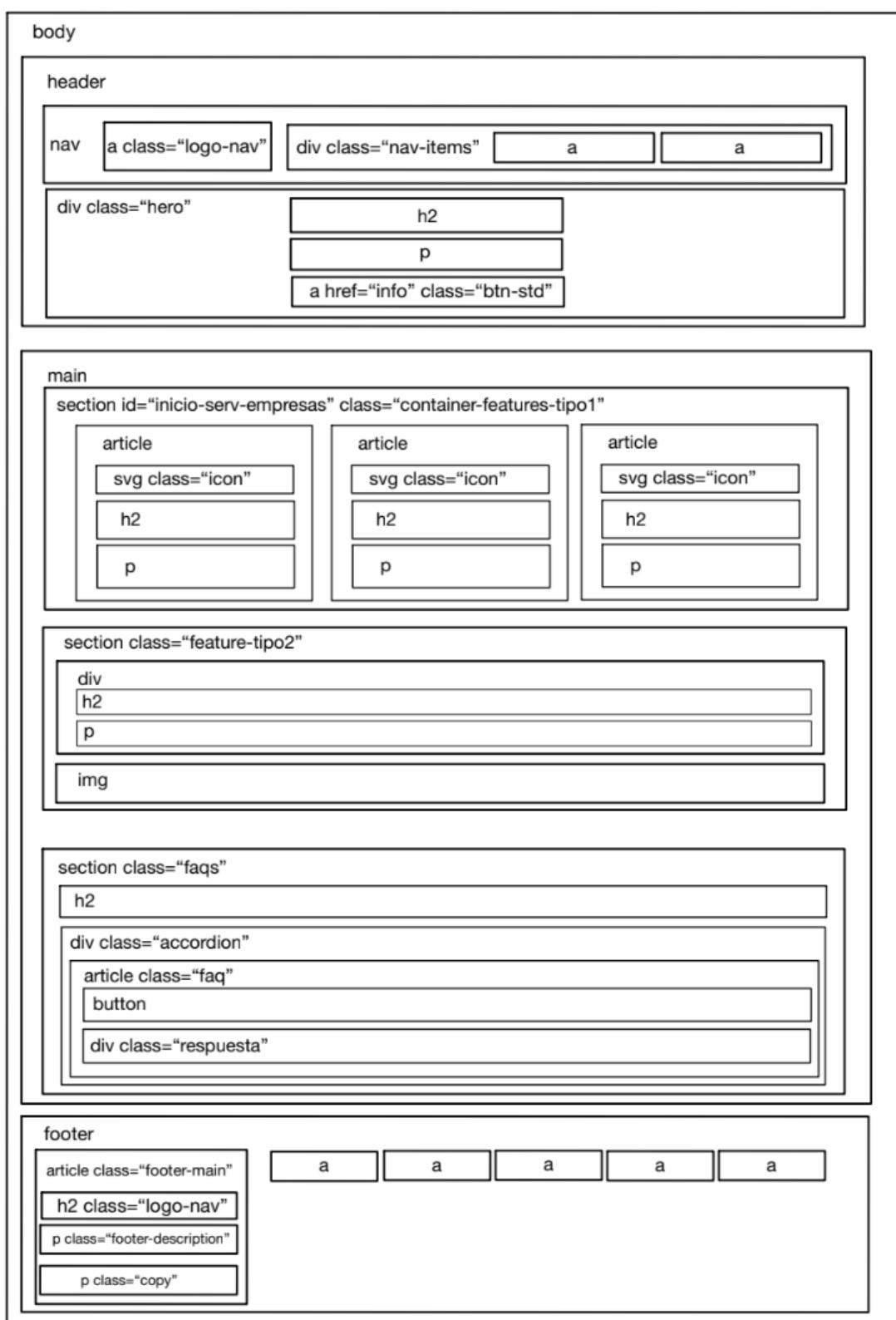


Figura 41: CSS: Mapa etiquetas página “Para empresas” (análogo para usuarios)

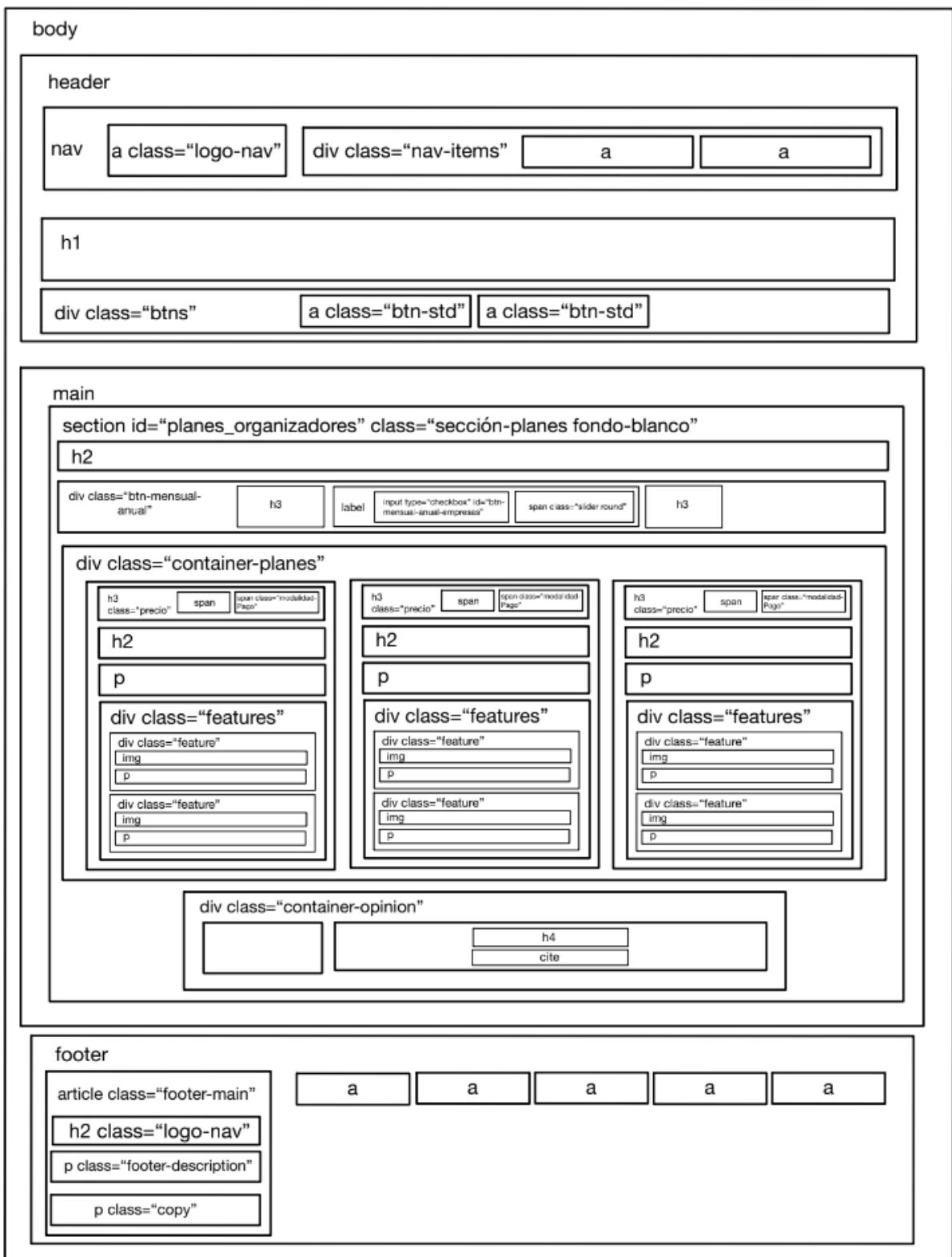


Figura 42: CSS: Mapa etiquetas página “tarifas”

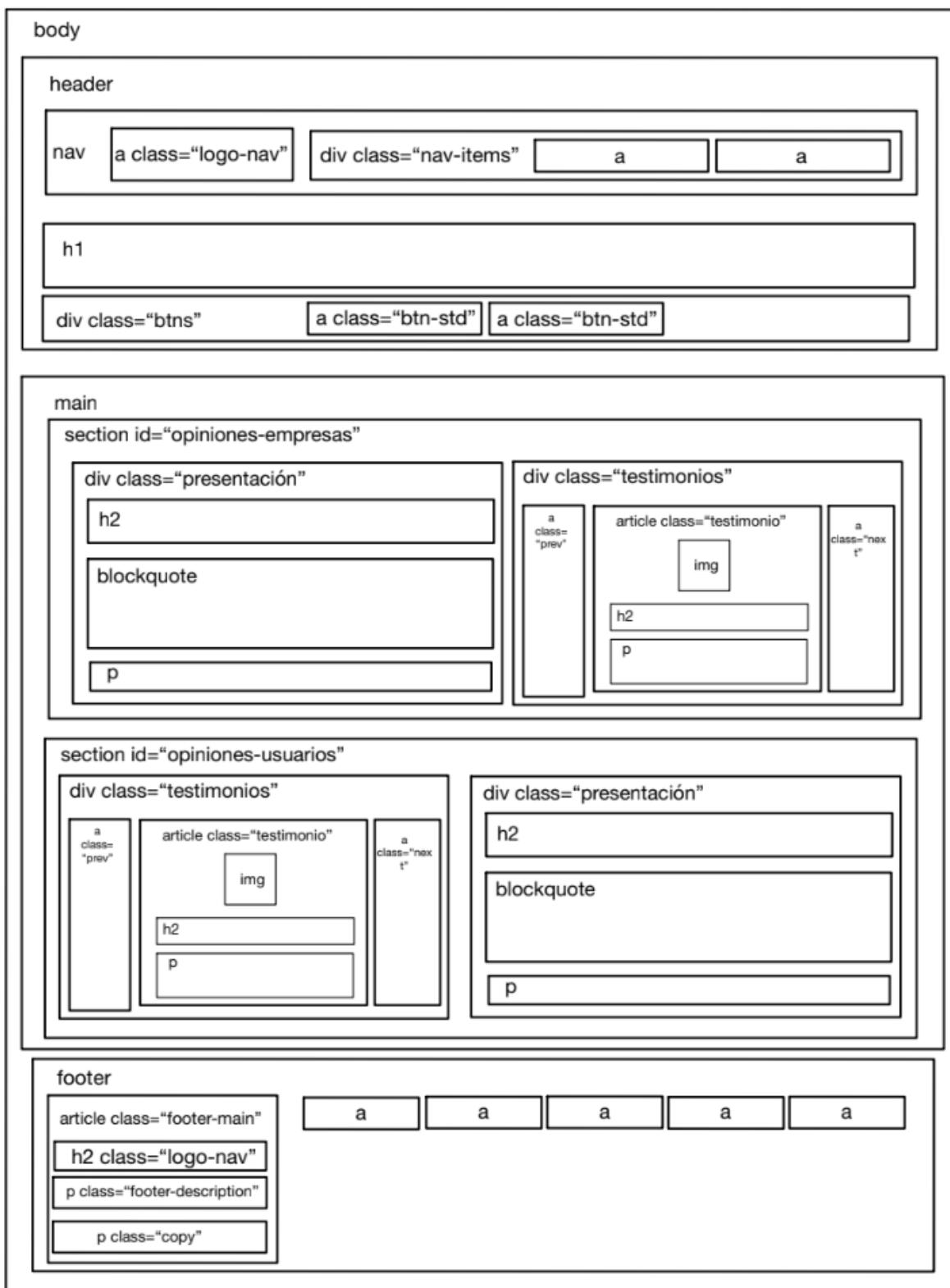


Figura 43: CSS: Mapa etiquetas página “Testimonios”