PAC 1: M4.258 – Eines HTML i CSS II



Enric Miralles i Selvi

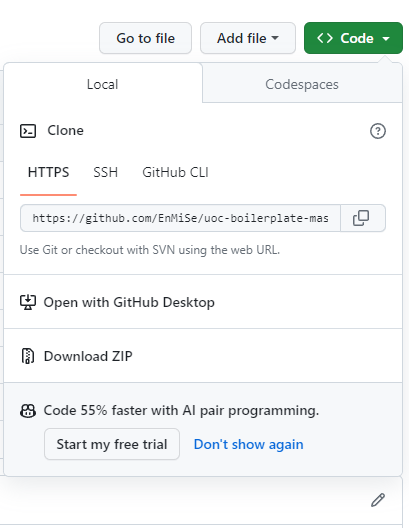
06/12/2023

Link web: <https://eineshtmlcss2pac1emiralles.netlify.app/>

Link repositori: <https://github.com/EnMiSe/uoc-boilerplate-master_emiralles_PAC2>

1. **Procés de desenvolupament**
   1. **Clonació repositori**

Per començar amb la pràctica s’ha clonat el repositori de Github amb el boilerplate de la UOC al pc mitjançant la opció que ofereix GitHub Web. En aquest cas s’ha triat la opció “Download ZIP” que hi ha a la secció “<> Code” de la pròpia pàgina de Github.



 A continuació, s’ha guardat i descomprimit la carpeta a un directori del dispositiu i s’ha creat un nou repositori mitjançant Github Desktop. Amb un commit inicial, s’ha pujat a Github els fitxers i components del nou projecte.

* 1. **Dependències**

S’instal·la node per mitjà de la comanda “npm install” amb la consola de Windows, un cop situats a la ruta amb la carpeta principal del repositori. Per a l’execució del lloc web en un servidor de desenvolupament”es tira la comanda “npm run dev”. Dita comanda, executa el script situat al fitxer package.json i crearà/modificarà carpetes com “dist” i “parcel” a mesura que s’avanci en el projecte.

Per altra banda, emprant un enllaç situat dins del head dels documents html, es permet a font-awesome poder importar icones al projecte.

* 1. **Contingut web**

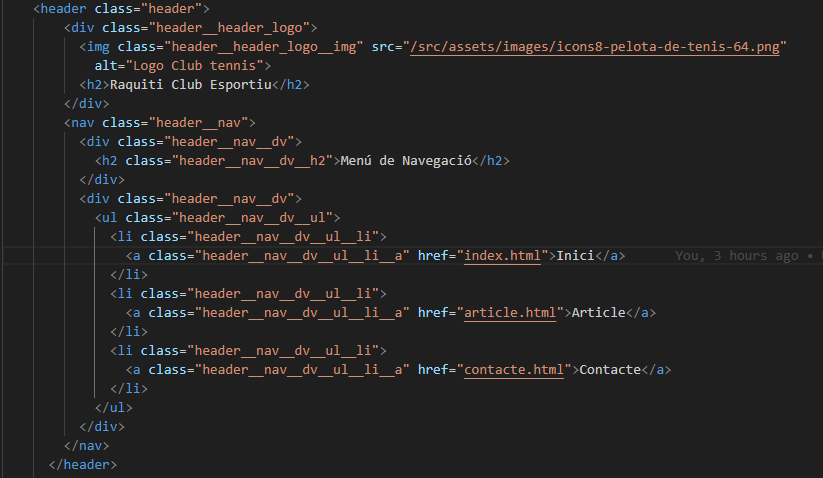
La idea principal es fer un lloc web promocional d’un club de tenis creat per un grup d’amics. L’objectiu es mostrar l’empresa als usuaris i incentivar-los a participar als tornejos i activitats que duu a terme la entitat.

Es crea 4 pàgines on dividir tot el contingut. La primera correspon a la portada que serveix com a presentació. S’ha implementat elements visuals i poc contingut en text amb un menú de navegació per a permetre a l’usuari redirigir-se entre les diferents pàgines. A la pàgina de participants es mostra un llistat dels participants que s’han apuntat al torneig de tenis. Seguidament, a la pàgina article s’ha creat una entrevista amb el conseller d’esports on es promociona els tornejos i s’explica una mica l’experiència viscuda en un dels tornejos anteriors. Finalment, a la pàgina de contactes hi ha un formulari on els usuaris poden introduir les seves dades personals i enviar un missatge per a demanar més informació o per a inscriure’s al torneig.

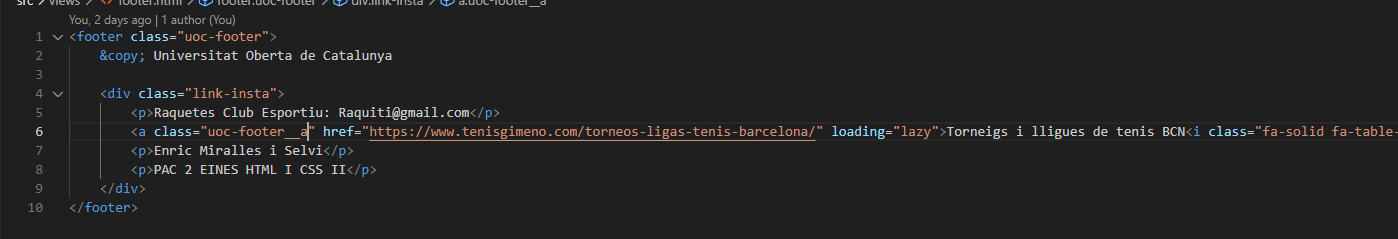
El text que compren la web s’han generat a través d’una IA per tal d’estalviar temps.

En general, totes les pàgines contenen un “header” i un “footer” compartit.

A la secció del “header” se situa la icona del club de tenis, a més del nom de la entitat i un menú de navegació per a navegar per les diferents pàgines del lloc web.

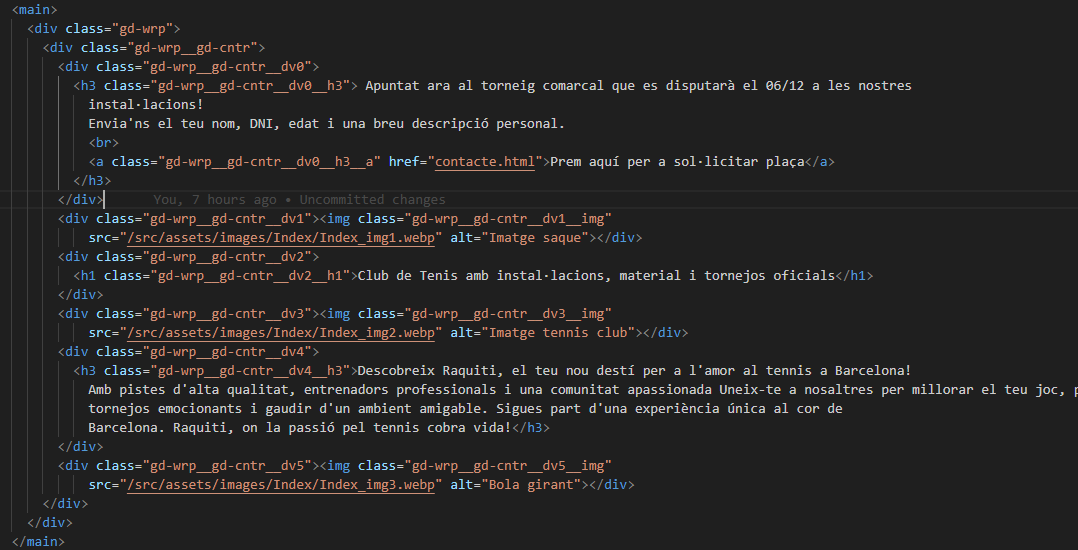


A la secció del “footer” situat en un altre fitxer .html, s’ha afegit el meu nom i el de l’assignatura a més d’una icona extreta de “font awesome”. Per afegir aquesta dependència s’ha enllaçat el document HTML amb un link “cdnjs”. Tambér s’hi ha afegit la icona dins de l’eqtiqueta <a> amb un enllaç a una pàgina externa per a consultar els tornejos vigents.



* **Portada**

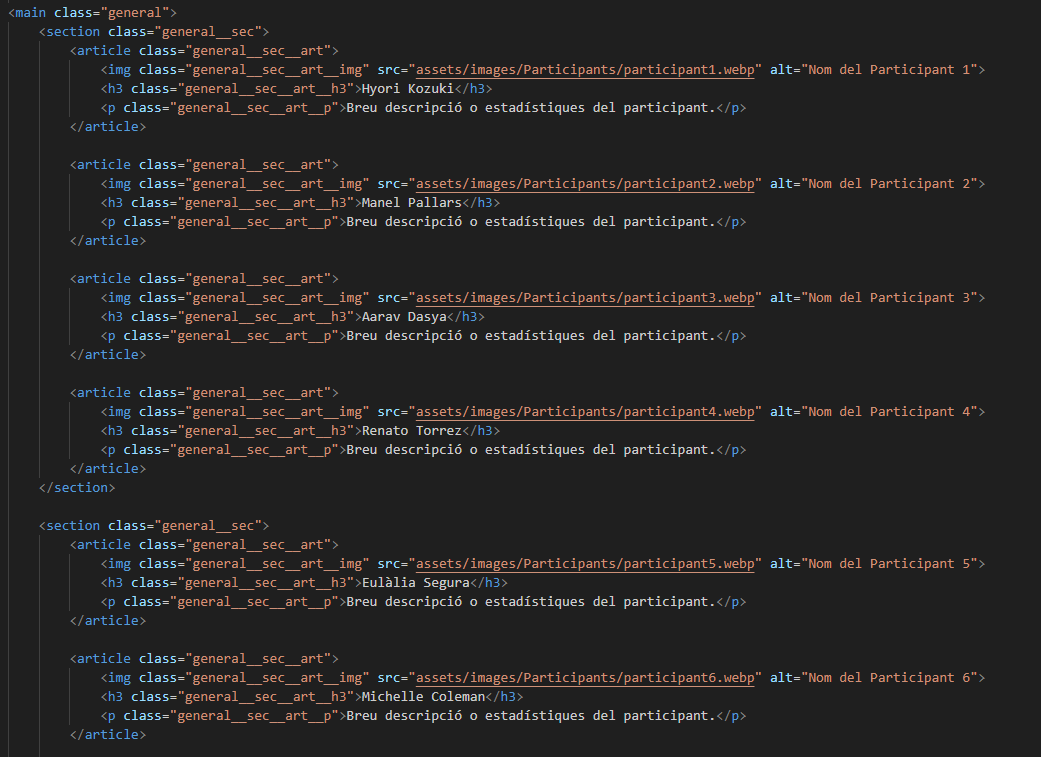
Pàgina maquetada amb el Grid de CSS amb l’objectiu de crear una divisió per caselles dels diferents elements a disposar en aquesta pàgina. El Grid consta de 9 caselles en les quals hi ha tres imatges en diagonal que ocupen la major part de la web per a captar l'atenció de l'usuari. Al centre, se situa el “logo” del club de tennis. A la resta de caselles s’hi reparteix el Títol de la web, així com el menú de navegació i un breu text referent a la lliga de tenis.  Es proporciona també una casella anunciant el següent torneig amb un enllaç per a indicar a l’usuari on ha de sol·licitar plaça.



Per als canvis de format a mòbil i tablet, els elements del grid es distribueixen per adaptar-se a la pàgina segons la resolució del dispositiu de l’usuari. També, com s’explica més endavant a la secció de disseny i estils, s’ha donat classes i ids als elements de la pàgina seguint la nomenclatura BEM.

* **Pàgina interior 1: Participants**

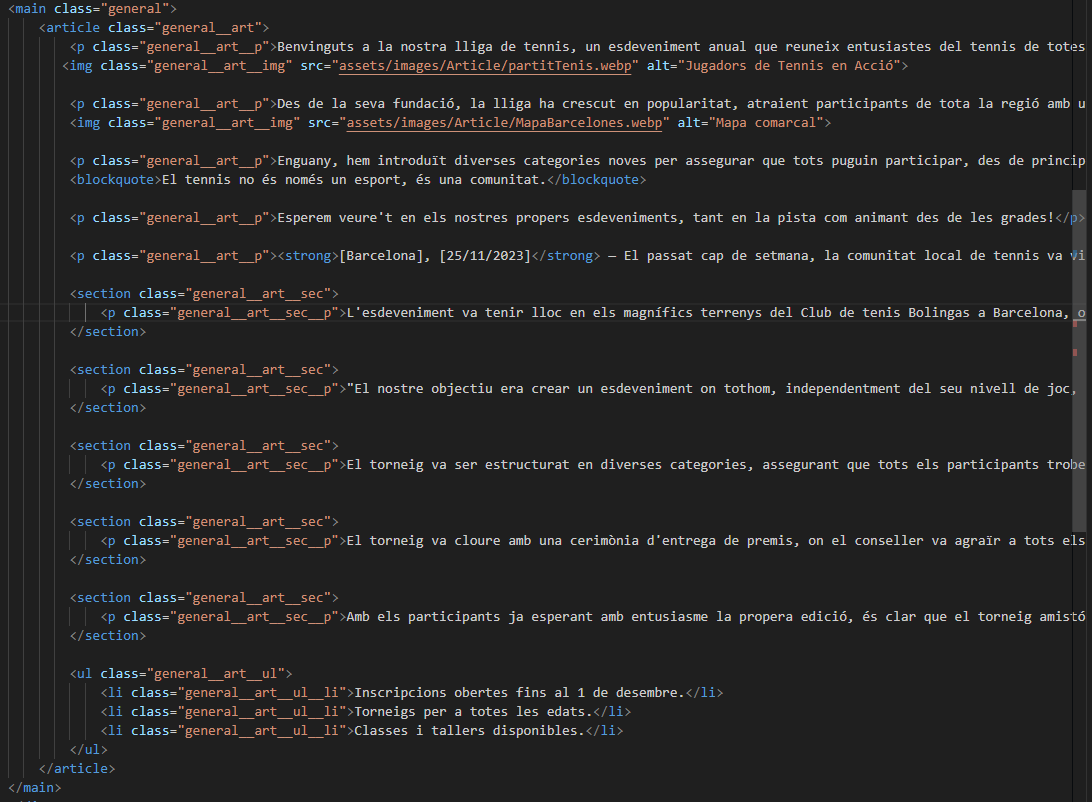
Pàgina centrada en mostrar un llistat dels participants del torneig de tenis. Conté dues seccions <section> que engloben articles <article> amb informació sobre diversos participants. Cada article inclou una imatge del participant, un títol amb el seu nom i un paràgraf de descripció o estadístiques.



La pàgina està estructurada de manera clara amb l'ús de classes BEM (Block Element Modifier) per a l'estilització CSS, permetent una organització i manteniment eficients dels estils. La pàgina també està dissenyada per ser responsiva, adaptant-se a diferents mides de pantalla.

* **Pàgina interior 2: Article**

Pàgina centrada en un article. Conté un <article> amb diversos paràgrafs <p> que descriuen l'esdeveniment, el club i altres detalls relacionats amb el tennis. També inclou imatges relacionades amb el tennis i un mapa, així com un <blockquote> per destacar una cita sobre el tennis. Al final de l'article, hi ha una llista amb ítems <li> que proporcionen informació addicional sobre el club i els esdeveniments, com dates d'inscripció i disponibilitat de classes.



La pàgina utilitza classes BEM per a l'estilització CSS i està estructurada per ser visualment atractiva i informativa, dirigida a persones interessades en participar o assistir a esdeveniments de tennis. La inclusió d'imatges i l'ús d'elements de llista afegeixen a la claredat i l'atractiu visual de la pàgina.

* **Pàgina interior 3: Contacte**

Pàgina web estàndard que inclou una capçalera, un menú de navegació, un formulari de contacte i un peu de pàgina. El contingut principal consta d’una secció que conté un formulari de contacte (<form>). El formulari inclou camps per a nom, edat, comentari, correu electrònic i una casella de selecció per apuntar-se a un torneig. Cada camp del formulari està etiquetat amb <label> i té els corresponents inputs.

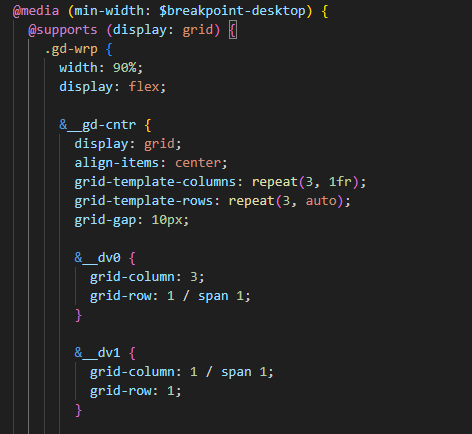


S’ha utilitzat BEM per a la creació de les classes html i els estils CSS.

* 1. **Entorn de desenvolupament**
* **Ús de Grid i @supports**

Referent a aquest apartat, s’ha utilitzat grid i @suports a la pàgina de la portada “index.html”.

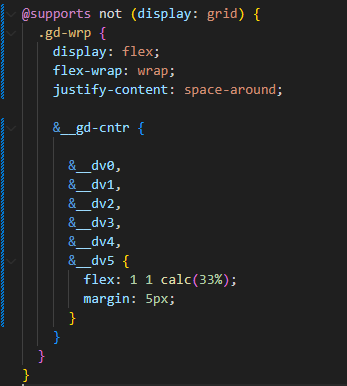
Pel Grid s’ha definit una graella CSS dins de la classe “.gd-wrp\_\_gd-cntr”. Aquesta està configurada amb tres columnes i tres files. Es fa servir grid-template-columns: repeat(3, 1fr) per crear tres columnes amb amplades iguals i flexibles. De manera similar, grid-template-rows: repeat(3, auto) estableix tres files amb alçades que s'adapten automàticament al contingut dels elements. Amb grid-gap: 10px afegeix un espaiat de 10 píxels entre les cel·les del grid, tant horitzontalment com verticalment, aportant una separació visual.



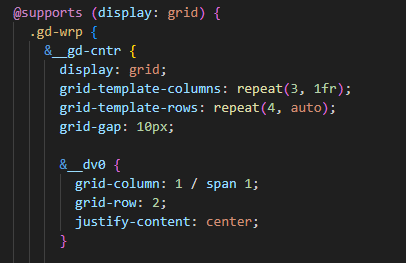
Diverses subclasses com &\_\_dv0, &\_\_dv1, &\_\_dv2, etc., s'utilitzen per posicionar específicament els elements. Per exemple, &\_\_dv0 { grid-column: 3; grid-row: 1 / span 1; } col·loca aquest element a la tercera columna i la primera fila de la graella

S’ha emprat @supports per a oferir una versió alternativa en cas de que el navegador de l’usuari no suporti CSS Grid.

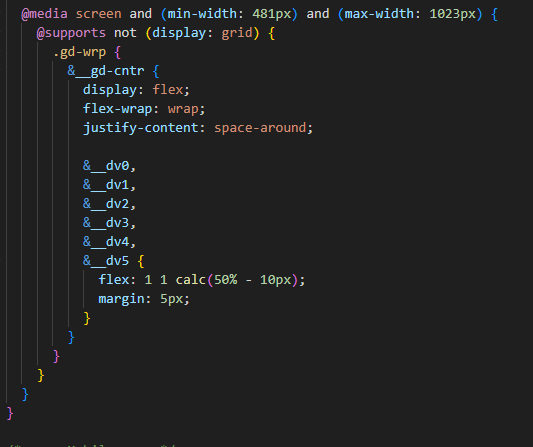
Pel format Desktop s’utilitza uns estils de reserva per organitzar els elements en fileres i columnes de manera similar a com ho faria Grid. El valor flex: 1 1 calc (33%); intenta imitar el comportament de tres columnes per fila, amb un espaiat de 5px entre els elements.



Per a tablets, se li afegeix una nova fila al grid per a permetre als elements tenir un major espai per a acomodar-se.



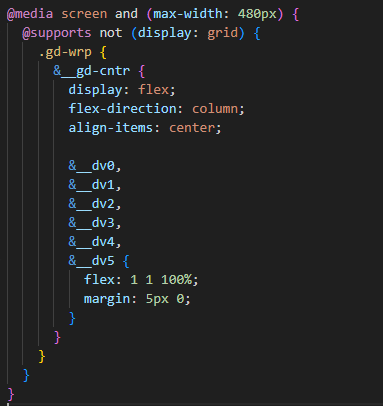
En cas de que el navegador no suporti grid, cada element ocupa la meitat de l'amplada disponible (50% - 10px), permetent dos elements per fila.



Finalment, per a dispositius mòbils, el grid s’adapta a la estreta pantalla dels dispositius disposant-se en una sola columna amb 9 files.



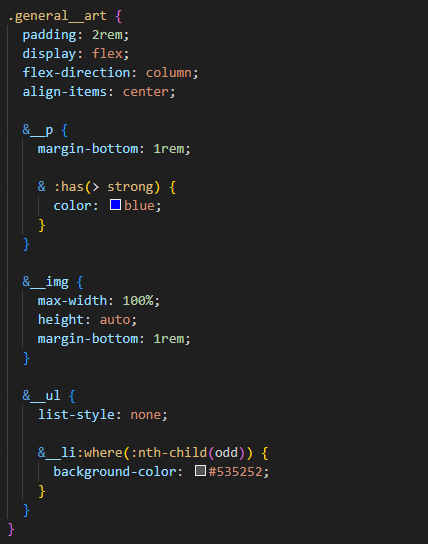
En cas de navegador que no suporti grid, els elements s'organitzen en una sola columna (flex-direction: column), amb cada element ocupant l'amplada completa disponible de manera similar a com ho faria el grid.



* **Pseudoclasses funcionals i capes de cascada**

S’ha utilitzat pseudoclasses a la pàgina de l’article. En aquest cas es fa servir dues pseudoclasses funcionals: :has() i :where().

En el cas de la classe :has() s'utilitza en la regla & :has(> strong) { color: blue; } dins de .general\_\_art &\_\_p. Aquesta pseudoclasse selecciona tots els elements <p> que contenen directament un element <strong> com a fill (> denota un fill directe). Si un element <p> compleix aquesta condició, el color del seu text canviarà a blau.



Per a la classe :where() s’ 'utilitza en &\_\_li:where(:nth-child(odd)) { background-color: #535252; } dins de .general\_\_art &\_\_ul. Aquesta pseudoclasse permet seleccionar tots els elements <li> imparells dins d'una llista (:nth-child(odd)), aplicant un fons de color #535252. L'avantatge d'utilitzar :where() aquí és que els estils definits poden ser fàcilment sobreescriure's per altres regles més específiques sense necessitat d'utilitzar !important.

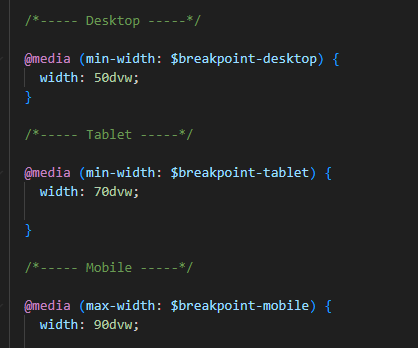
* **Container quèries**
* **Pàgina de contingut lliure:**

A la pàgina de “contacte”, s’ha utilitzat les unitats següents:

Unitats dvw (Dynamic Viewport Width):

Aquestes unitats es basen en l'amplada de la finestra de visualització (viewport). En el codi, s'utilitzen per definir l'amplada dels elements en funció de l'amplada de la pantalla.

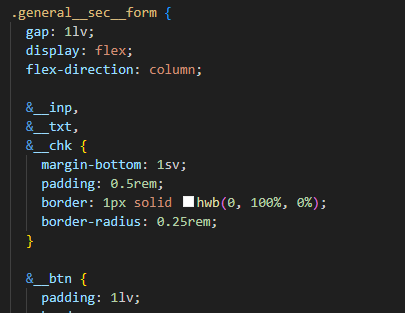
Exemple: .general\_\_sec { width: 80dvw; } significa que l'amplada de l'element .general\_\_sec serà el 80% de l'amplada de la finestra de visualització. Aquest valor canvia segons els punts de trencament (breakpoints) per a dispositius d'escriptori, tablets i mòbils.



Unitats lv (Length Viewport):

Aquestes unitats estan relacionades amb les dimensions de la finestra de visualització.

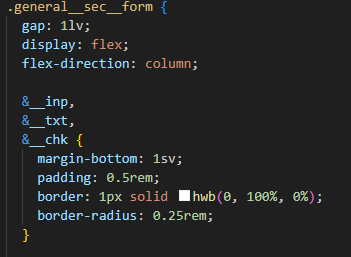
En el codi, la regla .general\_\_sec\_\_form { gap: 1lv; } el valor de gap està basat en una proporció de la dimensió del viewport.



Unitats sv (Scaling Viewport):

Aquesta unitat està vinculada a les dimensions del viewport.

El codi té margin-bottom: 1sv; dins de .general\_\_sec\_\_form &\_\_inp, &\_\_txt, &\_\_chk, que mostra un marge inferior basat en una proporció del viewport.



* 1. **Compilació**

1. **Metodologia i guia d’estils**
   1. **Metodologia aplicada**

Al fitxer \_home.scss també s’ha implementat tots els estils referents a colors, disposició dels elements a la web, així com els estils de les imatges i adaptació de la pàgina per als diferents formats de dispositius.

Al haver-hi molts elements a adaptar a la pàgina s’ha seguit la metodologia Desktop first per tal d’assegurar-se de que dits elements es disposen correctament. Per al format mòbil només es necessari donar un format on tot es disposi de manera vertical en na columna, ja sigui amb grid o flex-box.

El codi SCSS proporcionat conté diverses seccions d'estil per a una pàgina web, utilitzant la metodologia BEM per a la nomenclatura de classes. A continuació es detallen els components clau i les seccions:

Variables: Es defineix les variables per a punts de trencament (breakpoints) en diferents amplades de pantalla per a dispositius mòbils, tablets i escriptoris.

Estils Globals: S’estableix box-sizing: border-box per a tots els elements i defineix estils base per a l'etiqueta body, incloent tipografia, color de fons i mides.

Capçalera (header): Es configura estils per a la capçalera de la pàgina, incloent el logo i la barra de navegació. S’utilitza Flexbox per alinear elements i defineix estils per a enllaços de navegació.

Media Queries: Es defineix estils específics per a diferents mides de pantalla, especialment per a la classe .uoc-footer en dispositius d'escriptori.

Secció #INDEX: Conté estils per a la classe .gd-wrp i els seus descendents, incloent configuracions per a un disseny grid en escriptoris, tablets i mòbils.

Mitjançant @media, s’ha determinat l’us del Grid per a formats de gran resolució mentre que en formats més petits, s’han disposat els elements de manera vertical.

Secció #PARTICIPANTS: Estil per a una secció amb articles (art), incloent configuracions de flexbox i estils per a imatges.

Secció #ARTICLE: Inclou estils per a articles (art) amb paràgrafs i imatges, amb ajustos específics de text basats en la mida de la pantalla. Utilitza pseudoclasses funcionals com :has() i :where().

Secció #CONTACTE: Estableix un mixin form-styles per a estilitzar formularis i l'aplica a la classe .general\_\_sec\_\_form. Inclou estils responsius basats en la mida de la pantalla i estils per a botons.

Peu de Pàgina (footer): Estil per al peu de pàgina amb una imatge de fons gradient i estils de text.

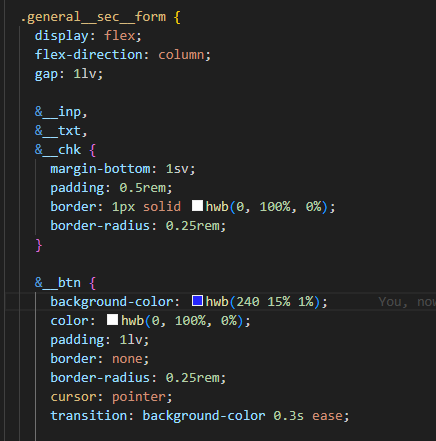
S’ha fet un intensiu de SCSS per a organitzar i modularitzar els estils, utilitzant mixins, variables i media queries per a assegurar la responsivitat i la coherència visual en diferents dispositius. Les pseudoclasses :has() i :where() són utilitzades per a aplicar els estils condicionals.

* 1. **Disseny i Estil aplicat (Afegir fotos resultants de la web)**

Per al disseny del lloc web s’ha escollit un fons negre que ressalti l’interior on es mostra el contingut rellevant. El color de la font (blanc), s’ha escollit per a crear un contrast amb el fons. Com el logotip seleccionat per a la web comparteix els colors vermell i blau, s’ha decidit fer un gradient entre ambdós colors per al footer. D’aquesta manera, la pàgina no queda nomes en blanc i negre i dona un toc de vivesa al conjunt.

El codi SCSS s’ha organitzat seguint l’estructura BEM ja implementada al codi html i de manera que paràmetres que donen estils referents a mides de contenidors, margins i paddings, es troben en primer lloc. Tot seguit, hi ha els colors referents a aquests contenidors i al seu contingut. En últim lloc, s’ha posat els paràmetres que defineixen les funcionalitats com flex-box i grid.

Finalment, per als colors referents al mòdul 3 s’ha decidit utilitzar el format HWB a la pàgina de contacte, al botó de submit del formulari. Els colors escollits son similars als de la paleta ja proposada i semblants als del logotip. Un blau com a color base que es torna a vermell quan hi passem per sobre el cursor.



* 1. **Configuració Stylelint (Afegir fotos)**

1. **Deployment a Netlify (Afegir fotos)**

Primerament es selecciona la opció de “New site from Git” per a seleccionar el repositori de Git el qual es vol fer el Deployment.

Un cop seleccionat, es selecciona la branca main del projecte i s’estableix les comandes (en aquest cas “npm run build”) per a compilar el codi juntament amb el directori de la publicació. Finalment, amb “Deploy site” es realitza el procés de publicació de la web.

1. **Referencies**

**Header:**

* **Logo tenis:** <https://iconos8.es/icons>

**Index:**

* **Index\_Img 1:** https://weezevent.com/es/blog/organizar-torneo-dia-mundial-del-tenis/
* **Index\_Img 2:** <https://www.freepik.es/vector-premium/tennis-logo-template-vector-deporte-activo-campeonato-torneos-tenis_25881866.htm>
* **Index\_Img 3:** <https://pixabay.com/es/photos/tenis-pelota-de-tenis-vez-pelota-1381230/>

**Participants:**

* **Generador de persones:** https://this-person-does-not-exist.com/es

**Article:**

* **Generador d’imatges:** https://gencraft.com/
* **Mapa Barcelonès:** https://www.enciclopedia.cat/gran-enciclopedia-catalana/el-barcelones