

Documentació projecte amb tailwind



Hassan Shah

MP09 - Disseny Interfícies Web

UF1 - CSS i Tailwind

UF2 - Elements multimèdia: creació i integració

13-12-23

Índex

1. Introducció.....	3
2. Tecnologies utilitzades.....	3
3. Disseny inicial del portfolio.....	3
4. Procés de desenvolupament.....	4
4.1. Colors i Fonts.....	4
4.2. Components reutilitzats.....	4
4.3. Flex.....	5
4.4. Selectors de tailwind.....	5
4.5. Responsive.....	5
4.6. Group.....	5
5. Optimització d'imatges.....	6
5.1. Compresió i format.....	6
5.2. Optimització amb Sharp.....	7
6. Webgrafia.....	9

1. Introducció

He decidit fer un portfolio personal el qual inclou la pàgina d'inici, llistat dels projectes, el detall de cada projecte i finalment la pàgina de la webgrafia. He fet servir un disseny peculiar que es caracteritza per tenir un espai entre cada component de la pàgina i que tenen unes cantonades rodones.

2. Tecnologies utilitzades

Les tecnologies utilitzades per a la realització del projecte són:

- Visual Studio Code, per fer el codi de la web.
- Tailwind CSS
- Parcel, com a gestor dels diferents paquets i deploy de la pàgina.
- Github

3. Disseny inicial del portfolio

Per començar veiem la pàgina de l'inici d'on hi ha un botó per veure el llistat de tots els projectes que està en una altra pàgina i per últim una pàgina individual per cada projecte.

Prototip fet amb figma:



4. Procés de desenvolupament

4.1. Colors i Fonts

Per triar els colors he fet servir la web de <https://coolors.co/palettes/trending> i he acabat combinant dues paletes de colors ja que cap m'acaba de convencer.

Els colors escollits són els següents:

```
colors: {
  transparent: 'transparent',
  current: 'currentColor',
  'white': '#ffffff',
  'principal': '#E9EDC9',
  'secondary': '#292929',
  'highlight': {
    500: '#D4A373',
    800: '#c98541',
  },
},
```

La font l'he deixat per defecte que és la 'Noto Sans'.

4.2. Components reutilitzats

Hi ha components que s'han utilitzat en més d'un lloc i són completament iguals com poden ser els botons.

El disseny dels botons és senzill i fa ús de l'efecte hover per fer-lo visualment més atractiu i a l'hora de posar el ratolí a sobre:



Al posar-te a sobre del botó el color el més fosc i el ratolí es torna un punter. Les classes de tailwind utilitzades són les següents:

```
■ hover:text-white ■ hover:bg-highlight-800 duration-500"
```

```
class="px-8 py-3 ■ bg-highlight-500 ■ text-white rounded-2xl
```

4.3. Flex

He fet ús del flex en tot el projecte per aconseguir el disseny dels blocs proposats al disseny.

També he fet molt ús del flex per justificar elements tant horitzontalment com verticalment i no per estructurar elements com en altres casos.

4.4. Selectors de tailwind

He fet servir els següents selectors desde un element pare:

```
[&>div>a]:border [&>div>a]:px-2 [&>div>a]:py-1
```

serveix per aplicar els mateixos estils a tant fills com tingui i en aquest cas aquests estils els estic aplicant a totes les etiquetes `<a>` que estiguin dintre d'un div partint de l'element pare i es fan servir per no repetir el mateix codi per cada un dels elements.

4.5. Responsive

Com ja sabem el tailwind és mobile first, per tant ho he respectat i he començat el disseny per mòbils i l'he anat adaptant per pantalles més grans fins arribar a l'escriptori.

He fet servir molts dels selectors com sm, md, lg ...

També he fet servir els selectors per rangs com per exemple (només afecten pel rang seleccionat):

md:max-lg:

4.6. Group

La utilitat de group serveix per agrupar elements com ja diu el propi nom i pots jugar de moltes maneres amb elll.

En aquest projecte l'he fet servir per crear un efecte hover a la pàgina d'índex, quan passa el ratolí per sobre del div dels projectes apareixen tots els elements de forma suau.

És fa servir de la següent forma:

```
<div class="group" □ bg-secondary ■ text-white w-1/3 text-center py-5">  
  <span class="text-[2vh] group-hover:opacity-100">TPV / POS</span>  
  <a class="group-hover:opacity-100 □ group-hover:text-highlight-500 rounded-md  
</div>
```

Al pare se li posa la classe group i després als fills els hi hem de dir que volem que facin al fer hover a l'element pare, que en aquest cas per defecte tenen la opacitat a 0 i al fer el hover la poso a 100 i apareixen els elements

5. Optimització d'imatges

5.1. Compressió i format

Imatges de la pàgines del llistat de projectes per la vista de tablets cap amunt

TPV -

Original: 469 Kb (jpg)

Optimitzada: 42 Kb (webp)

Collsacampus -

Original: 27 Kb (jpg)

Optimitzada: 12 Kb (webp)

USPG -

Original: 4 Kb (png)

Optimitzada: No he aconseguit optimitzar-la més del que ja està

Logo fet amb Inkscape: 872 B (png)

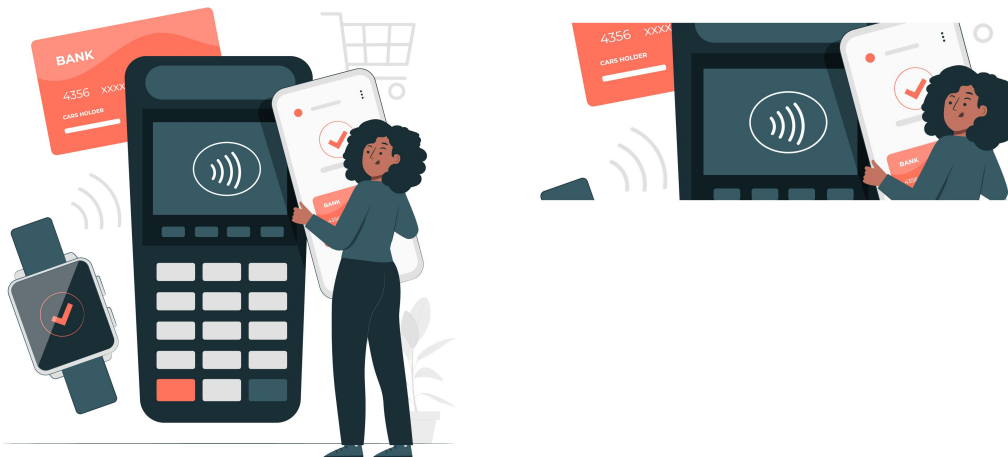
Imatge de la pàgina principal

Original: 411 Kb (jpg)

Optimitzada: 127 Kb (webp)

He retallat les imatges en la pàgina del projectes perquè siguin d'una resolució diferent per a dispositius amb pantalles petites. He triat el format webp ja que és el format ideal per les imatges per webs ja que redueixen molt el pes de les mateixes i no perden molta qualitat.

Per exemple la imatge del projecte de 'TPV' per a diferents dispositius:



Animació -

He afegit una animació de pulsació al botó de descarregar el curriculum.

POts veure l'animació al següent enllaç:

<https://www.tailwindcss-animated.com/configurator.html?animation=pulse&count=infinite&duration=3500>

Clip path -

He fet la imatge meva completament rodona fent servir el clip-path de la següent manera:

```
.profile-img{  
  clip-path: circle(50%);  
}
```

5.2. Optimització amb Sharp

Podem optimitzar imatges fent servir el sharp i automatitzar el procés amb parcel.

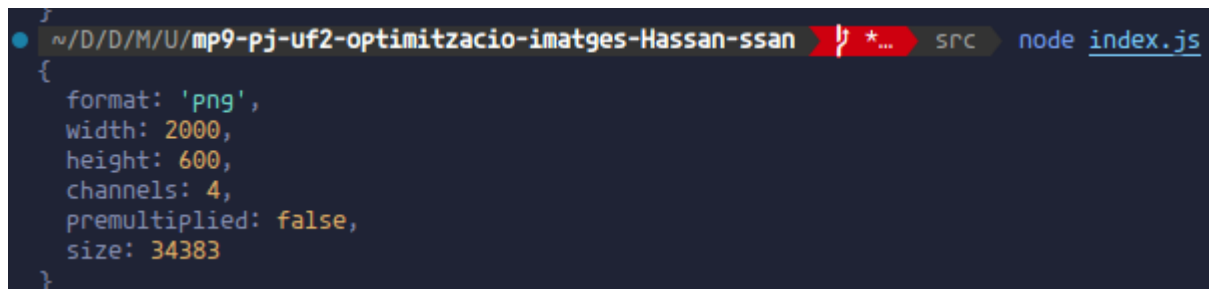
Per fer-lo servir l'hem d'instalar amb 'npm install sharp' i ja el podem fer servir, el codi js és el següent:

```
const sharp = require("sharp");  
  
sharp("../img/tpv-mb.png")  
  .png({ quality: 75, compressionLevel: 7 })  
  .toFile("tpv-mb.webp", (err, info) => {  
    if (err) {  
      console.error(err);  
    } else {  
      console.log(info);  
    }  
  });
```

Li hem de donar la imatge d'entrada i el nom i format de la sortida i si executem això amb node podem veure que s'ha perdut molt pes sense perdre gaire qualitat:

Per original - 293 Kb (png)

Comprimida amb sharp - 33 Kb (webp)

A screenshot of a code editor window with a dark theme. The top bar shows the file path '~ / D / D / M / U / mp9-pj-uf2-optimitzacio-imatges-Hassan-ssan' followed by a red error icon and 'src'. The editor displays a JavaScript object with the following properties: format: 'png', width: 2000, height: 600, channels: 4, premultiplied: false, and size: 34383. The code is enclosed in curly braces.

```
{
  format: 'png',
  width: 2000,
  height: 600,
  channels: 4,
  premultiplied: false,
  size: 34383
}
```

Video amb u tutorial de com utilitzar sharp:

<https://youtu.be/D-3aWAOqPd8>

6. Webgrafia

<https://tailwindcss.com/> - Tailwind

<https://color.adobe.com/es/explore> - Paleta de colors per orientar-me

www.figma.com - Figma per al disseny

<https://www.freepik.es/> - Imatges

<https://www.tailwindcss-animated.com/configurator.html> - Animació de tailwind

<http://www.gimp.org/es/> - Programa per adeitar i optimitzar imatges

<https://www.npmjs.com/package/sharp> - Per optimitzar imatges amb js

<https://inkscape.org/es/> - Software per crear el logo de la web