

Disseny responsive

Joel Vizcaíno
MP615 - Disseny d'interfícies

Índex

1. Resultat d'aprenentatge i criteris d'avaluació
2. Què és un disseny responsive?
3. Què són els media queries , unitats flexibles, flexbox/grid?
4. Resolucions
5. Passos per fer un disseny responsive

Resultat d'aprenentatge i criteris d'avaluació

2. Crea interfícies web homogènies definint i aplicant estils.

Criteris d'avaluació.

2.1 Reconeix les possibilitats de modificar les etiquetes HTML.

2.2 Defineix estils de manera directa.

2.3 Defineix i associa estils globals a fulls externs.

2.4 Defineix fulls d'estils alternatius.

2.5 Redefineix estils.

2.6 Identifica les diferents propietats de cada element.

2.7 Crea classes d'estils.

2.8 Utilitza eines de validació de fulls d'estils.

2.9 Analitza i utilitza tecnologies i frameworks per crear interfícies web amb un disseny responsive.

2.10 Analitza i utilitza preprocessadors d'estils per traduir estils comuns a un codi estàndard i recognoscible pels navegadors.

2. Què és un disseny responsive?

Un disseny **responsive** és una tècnica de desenvolupament web que **adapta automàticament el format** i la disposició **dels continguts** d'un lloc web **a qualsevol** mida de **pantalla i/o dispositiu**, tenint en compte els més importants avui dia: mòbils, tauletes i pantalla d'ordinador. Actua en temps real, és a dir, no s'ha de recarregar la pàgina per veure les adaptacions de la pàgina a les resolucions de pantalla.

Per fer un disseny responsive és molt important utilitzar:

- Elements *media queries*
- Unitats flexibles i *flexbox/grid*

Beneficis principals:

- Millora l'experiència d'usuari i SEO (Google prioritza pàgines responsives).
- Redueix costos de manteniment (només una pàgina per diversos dispositius).

3. Què són els media queries , unitats flexibles, flexbox/grid?

Media queries	Unitats flexibles	Flexbox/grid
• Regles CSS. • Estils condicionals segons característiques del dispositiu (amplada, alçada i orientació). • Disseny mobile first es dissenya en petit i després mitjançant media queries s'adapta a pantalles grans.	• Mides relatives que s'adapten a la pantalla. • Substitueix pixels fixos, ja que aquests no s'adapten. • Per exemple, quan utilitzem: width: 50% • S'utilitzen ("%", "rem" i/o "em").	• Flexbox: crea contenidors unidimensionals (fila o columna), que distribueix espai i alineen elements automàticament. • Molt utilitzat per menús, cards, seccions, etc. • Grid: gestiona posicions bidireccionals (files i columnes), ideal per layouts complexos. • Defineix espaiats (gap) i posiciona elements (justify-items/align items).

4. Resolucions

- **Mòbils petits:** 320px (mòbils petits com iPhone 4 (molt antic))
- **Mòbils mitjans:** 360px i 375px (mides freqüents en Android i iPhone 6, 7, 8)
- **Tablets verticals:** 768px (iPads i tablets comuns en orientació vertical i mòbils actuals)
- **Tablets horitzontals i petites pantalles d'ordinador portàtil:** 1024px (especialment tablets en horitzontal i portàtils petits).
- **Pantalles portàtils mitjanes:** 1366px (resolució típica d'ordinadors portàtils).
- **Pantalles d'escriptori grans:** 1920px (pantalles Full HD comunes d'ordinadors d'escriptori).

```
@media (min-width: 320px) { /* mòbil petit */ }  
@media (min-width: 360px) { /* mòbil mitjà */ }  
@media (min-width: 768px) { /* tablet vertical */ }  
@media (min-width: 1024px) { /* tablet horitzontal / portàtil petit */ }  
@media (min-width: 1366px) { /* portàtil mitjà */ }  
@media (min-width: 1920px) { /* escriptori gran */ }
```

5. Passos per fer un disseny responsive

1. Anàlisi i preparació de l'HTML:

- Verificar que la pàgina tingui una estructura semàntica clara (elements **<header>**, **<nav>**, **<main>**, **<section>**, **<footer>**).
- Assignar classes útils i descriptives per poder aplicar estils CSS amb facilitat i evitar la dependència de selectors massa generals.

2. Configuració del viewport:

- Incloure la meta etiqueta: **<meta name="viewport" content="width=device-width, initial-scale=1.0">** dins del **<head>** per assegurar que el navegador ajusti la visualització conforme a l'amplada del dispositiu.

3. Estils base (Mobile First)

- Començar el CSS apostant per la disposició i estil per dispositius mòbils (amb pantalles petites).
- Emprar unitats relatives com **%**, **em**, **rem** i **vw** en lloc d'unitats fixes com **px** per adaptar-se millor a diferents pantalles.
- Utilitzar **display: flex** o **display: grid** per facilitar la disposició flexible i adaptativa dels continguts.
- Aplicar **box-sizing: border-box**; per facilitar el control de mides tenint en compte el padding i border dins de l'amplada total.

5. Passos per fer un disseny responsive

4. Media queries per adaptar-se a pantalles més grans:

- Primer aplicar a **min-width: 768px** per tablets, generalment canviant la direcció dels elements de vertical a horitzontal, ampliant marges i espais.
- Després aplicar a **min-width: 1024px** per desktops, implementant una disposició en graella o multi-columnes, fent el contingut més ample i afegint millores visuals i d'espaiat.

5. Proves iteratives:

- És fonamental provar la pàgina a diferents mides i dispositius, ajustant marges, mides i estructura perquè la usabilitat i llegibilitat siguin òptimes en totes les condicions.