

## PAC 2 – DESENVOLUPAMENT DE WEB

M4.257 – EINES HTML I CSS

JORDI CRUZ LLADÓ

GITHUB: <https://github.com/stratdi/M4.257-PAC2>

NETLIFY: <https://rainbow-blini-dd77f7.netlify.app/>

## Taula de contingut

<b><i>Tècniques responsive utilitzades .....</i></b>	<b><i>2</i></b>
Flexbox.....	2
Grid Layout.....	2
Imatges .....	2
Resolution switching.....	2
Direcció d'art.....	3
Ús de Parcel .....	3
<b><i>Animació d'elements CSS .....</i></b>	<b><i>3</i></b>
Menú de navegació .....	3
Títol de la pàgina 'presentació' .....	4
Primera part (0%).....	4
Segona part (70%).....	4
Mantenim el farcit i la traça sense canvis. ....	4
Tercera part (80%).....	4
Quarta part (100%) .....	4
<b><i>Ús de clip-path.....</i></b>	<b><i>5</i></b>
<b><i>Estandards i qualitat de codi.....</i></b>	<b><i>5</i></b>
Portada .....	6
Categoria .....	7
Personatges.....	8
Temporades.....	9
Presentació .....	10
Enllaços .....	11

## Recursos gràfics: Justificació dels formats utilitats.

Pel logotip hem triat el format PNG per dos motius: per la transparència i perquè, al ser el logotip, tingui la millor qualitat d'imatge possible.

Per les imatges del contingut hem tirat el format JPEG, gràcies a que permet comprimir la informació sense renunciar gaire a la qualitat i reduir la mida dels arxius. També, per compatibilitat amb altres navegadors i per emprar menys la CPU (ja que sinó haguéssim triat el format WebP.)

Per la imatge de la portada, on aplicarem clip-path, hem triat una imatge en WebP per la seva adaptabilitat a les animacions.

## Tècniques responsive utilitzades

### Flexbox

Per fer un menú de navegació adaptatiu hem utilitzat el model Flexbox, ja que ens permet reorganitzar de manera fluida els ítems de navegació per diferents amplades de pantalles. En amples de, com a mínim 769px, el menú es mostrarà amb els ítems un vora l'altre. En amples menors, el menú es reagrupa de forma que els ítems queden apil·lats un amb l'altre. Aquest comportament ho aconseguim amb les Media Query.

El footer també el fem de manera semblant amb Flexbox.

Pels ítems de la classe 'card', que s'utilitza a gran part de la web, també hem utilitzat Flexbox però de diferent manera que al menú. Quan l'ample té com a mínim 769px, apareixerà una imatge a l'esquerra i un text a la dreta. Per amplades menors, la imatge es posa damunt el text i agafa tot l'ample del contenidor.

### Grid Layout

A les pàgines dins 'Categoria', tenim un estil anomenat 'content-with-related' que utilitzam per tenir a l'esquerra el contingut propi de la pàgina i, a la dreta, contingut relacionat. Gràcies a les Media Query, quan la pantalla és major a 1201px, es mostra així. Però, per pantalles amb menor mida, el contingut de la dreta baixa avall del tot i la part esquerra ocupa tot l'ample de la pantalla.

### Imatges

Les imatges també poden ser responsive, com hem vist al mòdul 3, i en aquesta PAC hem emprat les següents tècniques:

#### Resolution switching

Aquesta tècnica l'hem emprada al logotip de la pàgina (tant del menú com del footer), i l'hem emprada tant en grandària com en dpis, agafant com a breakpoint la mida de la pantalla de 768px.

També l'hem emprat per la imatge de la bandera de la pàgina inicial, però en aquest cas només l'hem emprat per dpis.

En el cas de les imatges de les pàgines ‘categoria’, ‘personatges’ i ‘temporades’, l’hem emprat només en grandària.

#### Direcció d’art

Aquesta tècnica l’hem emprat a la pàgina ‘presentació’, per la imatge inicial després del títol. Aquí, per pantalles amb mides iguals o superiors a 769px es carregarà una imatge amb format rectangular. Per mides inferiors, es carregarà una imatge quadrada.

#### Ús de Parcel

Una de les avantatges d’utilitzar Parcel és que podem automatitzar la tècnica de Resolucion Switching sense haver de preparar cada imatge per separat. Parcel ens permet afegir uns paràmetres a la URI de cada imatge que ens permet redimensionar una imatge de manera automàtica. En generar la carpeta dist/, Parcel generarà aquestes imatges per nosaltres.

## Animació d’elements CSS

Hem animat diferents elements utilitzant CSS:

- El menú de navegació
- El títol de la pàgina a la pàgina de ‘presentació’.

#### Menú de navegació

La idea que teníem era animar el menú de navegació quan l’usuari es posa damunt un ítem amb el ratolí (hover). Se’ns ha acudit fer una animació clàssica però vistosa, que consisteix en assignar un color per cada ítem a la part inferior i, en fer hover, que aquest color vagi creixent fins acabar d’ocupar tot el color de l’ítem.

Per aconseguir això, hem utilitzat el següent CSS:

A cada ítem li hem assignat una classe i, dins aquesta, la propietat

```
background: linear-gradient(var(--color-menu-bg), var(--color-menu-bg) 50%, #3498db 50%, #3498db);
```

D’aquesta manera definim un fons gradient, però exagerant la transició de manera que sembla que la meitat superior és d’un color i la inferior d’un altre.

Amb la següent propietat ampliam el color superior per fer que ocupi tot (i, d’aquesta manera, amagui l’altre color)

```
background-size: 100% 200%;
```

Després, de forma genèrica per cada ítem del menú, indicam la transició amb aquesta propietat:

```
transition: background .3s;
```

I, finalment, al seu hover fem que el segon color desplaci el color inferior:

```
background-position: 100% 100%;
```

D’aquesta manera, al fer hover d’un ítem, fa la transició de manera elegant i suau.

### Títol de la pàgina ‘presentació’

En aquest cas, hem introduït una imatge SVG amb el nom de la pàgina web, ja que era un dels requeriments d'aquesta pràctica. Com que aquesta imatge és un gràfic vectorial, hem aprofitat per fer una animació vistosa amb CSS.

La idea és que les lletres apareguin primer amb la silueta de les lletres per, finalment, quedar pintades completament. Aquest comportament es fa cíclicament, és a dir, no quedarà fixa una vegada feta, sinó que una vegada colorejades les lletres es tornaran a desfer pel seu contorn.

Per aconseguir això, el primer de tot, creem un keyframe a CSS amb l'animació. L'anomenarem ‘stroke’.

```
@keyframes stroke {
  0% {
    fill: rgba(72, 138, 20, 0);
    stroke: rgba(54, 95, 160, 1);
    stroke-dashoffset: 25%;
    stroke-dasharray: 0 50%;
    stroke-width: 2;
  }

  70% {
    fill: rgba(72, 138, 20, 0);
    stroke: rgba(54, 95, 160, 1);
  }

  80% {
    fill: rgba(72, 138, 20, 0);
    stroke: rgba(54, 95, 160, 1);
    stroke-width: 3;
  }

  100% {
    fill: rgba(72, 138, 204, 1);
    stroke: rgba(54, 95, 160, 0);
    stroke-dashoffset: -25%;
    stroke-dasharray: 50% 0;
    stroke-width: 0;
  }
}
```

#### Primera part (0%)

- **fill**: estableix el farcit de la transparent.
- **stroke**: estableix el color de la traça.
- **stroke-dashoffset**: iniciat en 25% de la longitud de la traça.
- **stroke-dasharray**: 0% 50%, indicant que inicialment no hi ha una traça visible i s'estableix un espai igual al 50% de la longitud del traç.
- **stroke-width**: l'amplada de la traça.

#### Segona part (70%)

Mantenim el farcit i la traça sense canvis.

#### Tercera part (80%)

Ampliam l'amplada de la traça.

#### Quarta part (100%)

- **fill**: fem visible el farcit.
- **stroke**: estableix la traça transparent.
- **stroke-dashoffset**: -25%, fent que sembli que el traç s'esvaeix.
- **stroke-dasharray**: invertim la traça, contribuint a l'efecte d'esvaniment.
- **stroke-width**: l'amplada del traç es redueix a 0, el que completa la transició del traç des de la seva aparença inicial fins a desaparèixer completament.

## Ús de clip-path

Per satisfer el requeriment d'usar clip-path a la PAC, a la pàgina inicial hem ubicat una imatge en format WebP amb la bandera icònica de la tripulació dels barret de palla.

Per no fer els típics retalls i formes, hem realitzat com un retall de la bandera en forma poligonal.



Per fer aquesta forma, hem definit 34 punts al polígon, de manera que retallam el voltant de la imatge, cream com un rectangle poligonal, retallam l'interior però respectant la imatge de la calavera, generant una forma octogonal.

## Estandards i qualitat de codi

La pàgina ha de complir les recomanacions de la WCAG 2.1 AA. Per això, se'ns indica utilitzar eines com AChecker per analitzar la nostra pàgina web.

Com que AChecker es troba caiguda, hem decidit utilitzar dos validadors com a plugins del navegador web:

- ☐ [WAVE](#) (Web Accessibility Evaluation Tool). Avalua fins a WCAG 2.2.
- ☐ [Axe DevTools](#). Avalua fins a WCAG 2.1 (en la versió gratuïta).

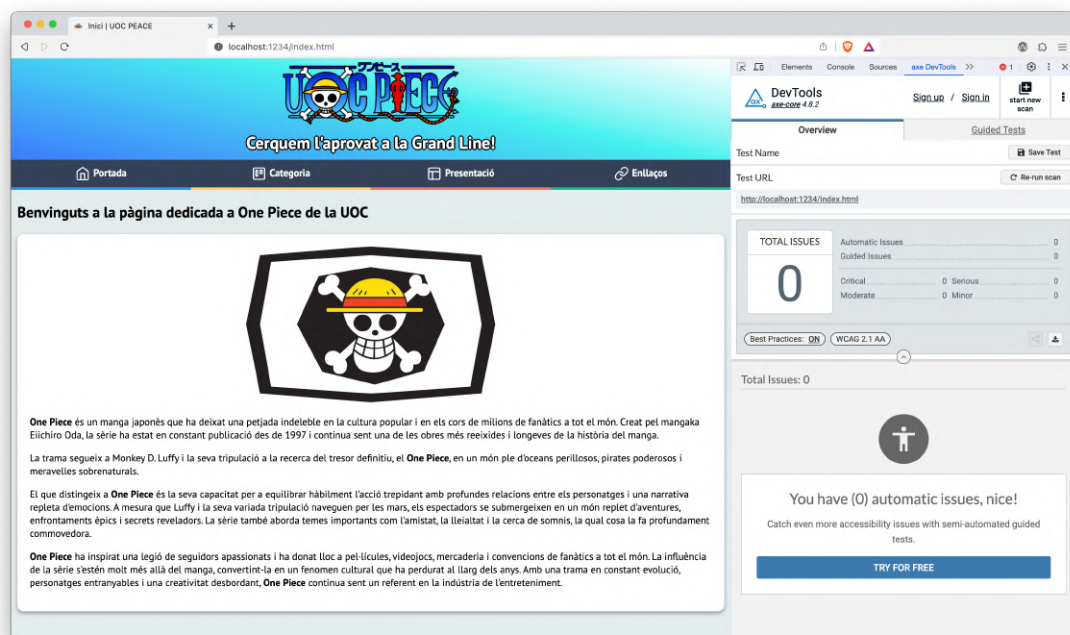
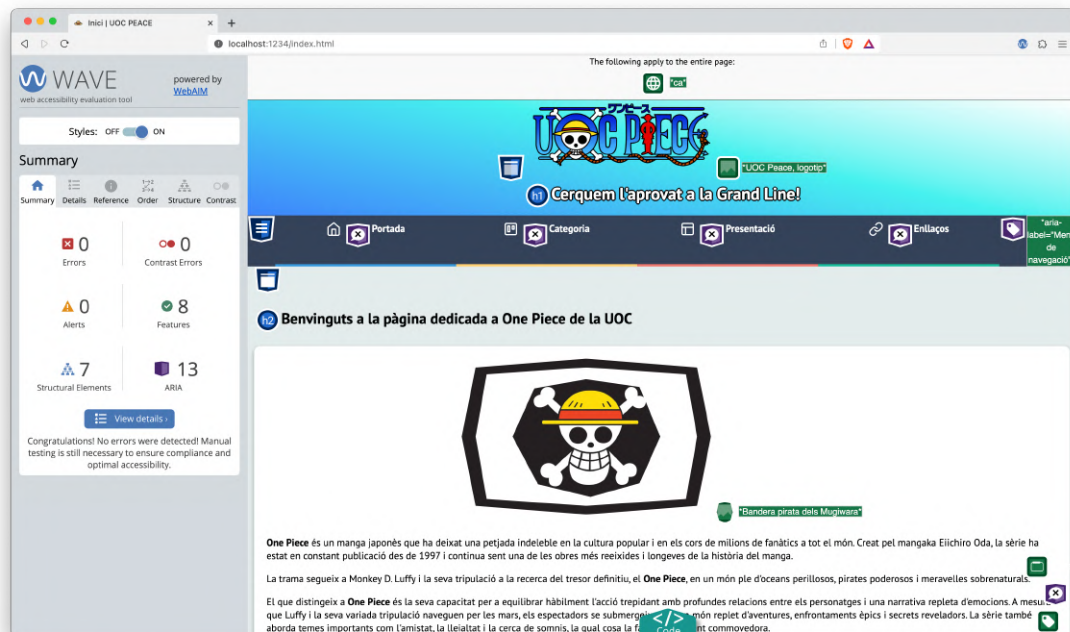
Executant aquestes eines hem vist que hi havia petites cosetes per modificar. Les alertes o errors reportats han estat els següents:

- ☐ Al footer, tenim elements d'input sense un fieldset i legend. La correcció és clara: hem introduir els elements i donat estil amb CSS per integrar-ho a la web.
- ☐ A la pàgina d'enllaços, les fonts les visualitzàvem a 3rem. WAVE ha reportat que sembla un possible 'heading', ja que texts amb tanta mida sembla que han de ser títols. Hem reduït la mida a menys de 20px i solucionat.
- ☐ Als vídeos de YouTube, ens recomana que ens assegurem que hi ha subtítols del contingut que estiguin sincronitzats amb la imatge. Aquesta no hem pogut fer res, ja que és un contingut aliè a la pàgina web.

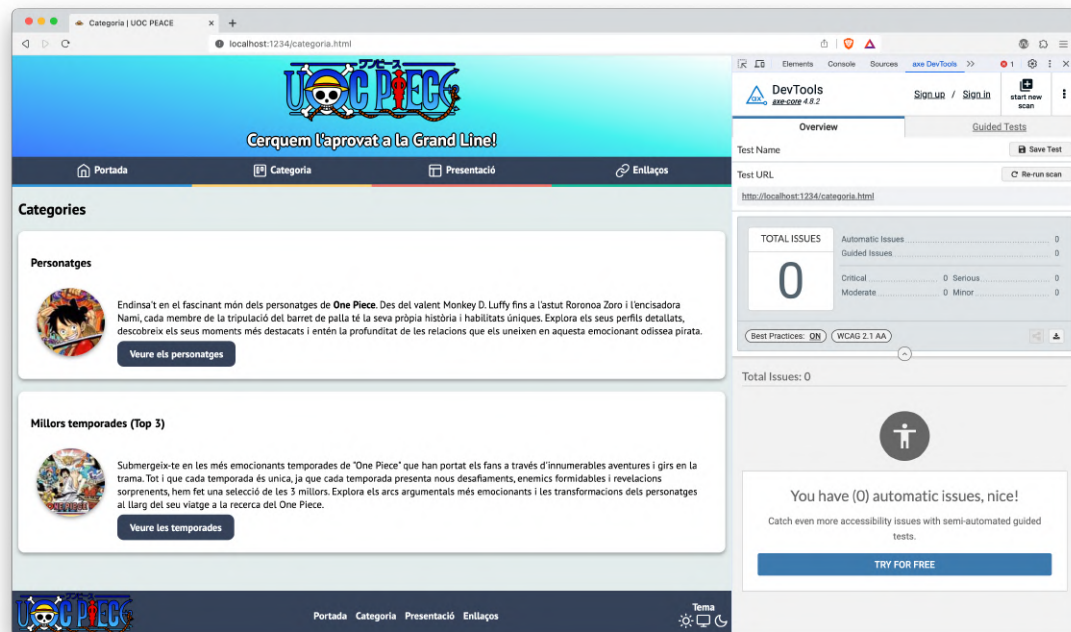
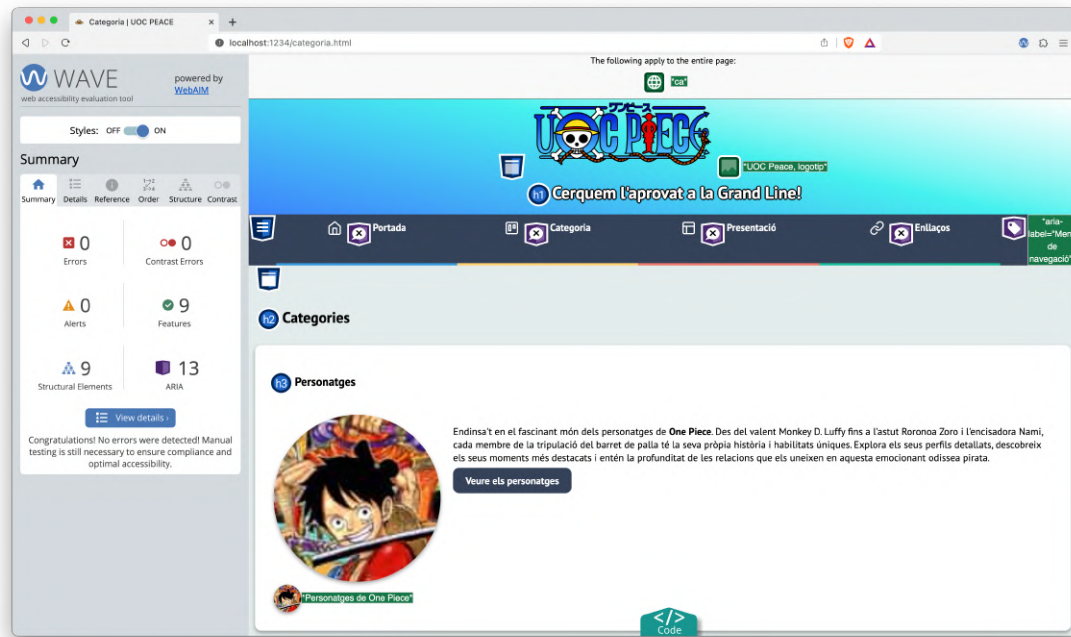
A continuació, adjuntam captures de cada pàgina amb el seu test:



## Portada

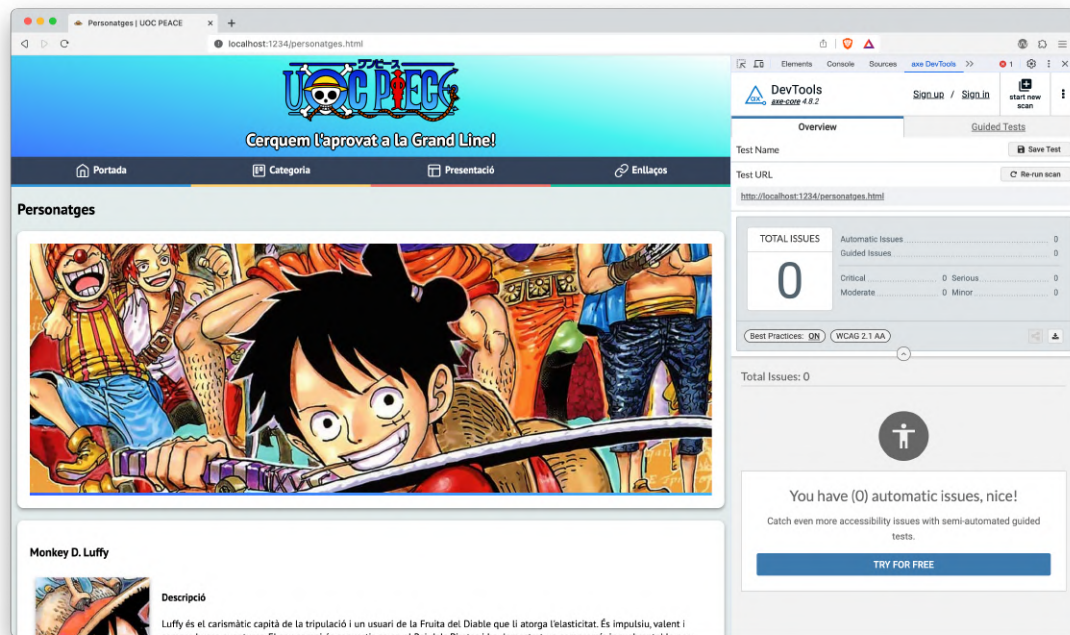
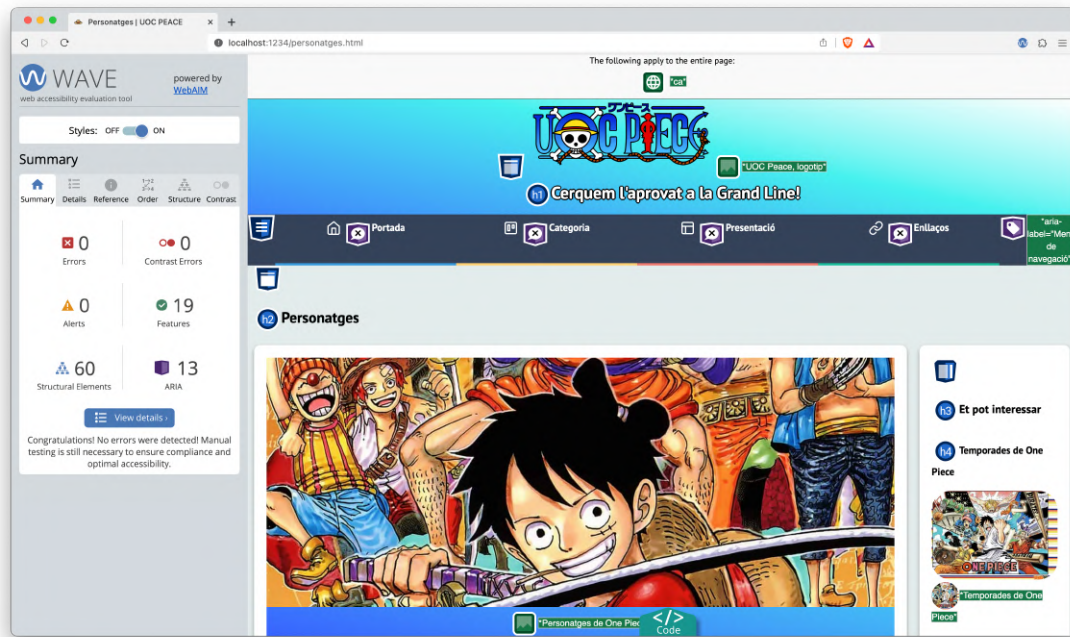


## Categoria

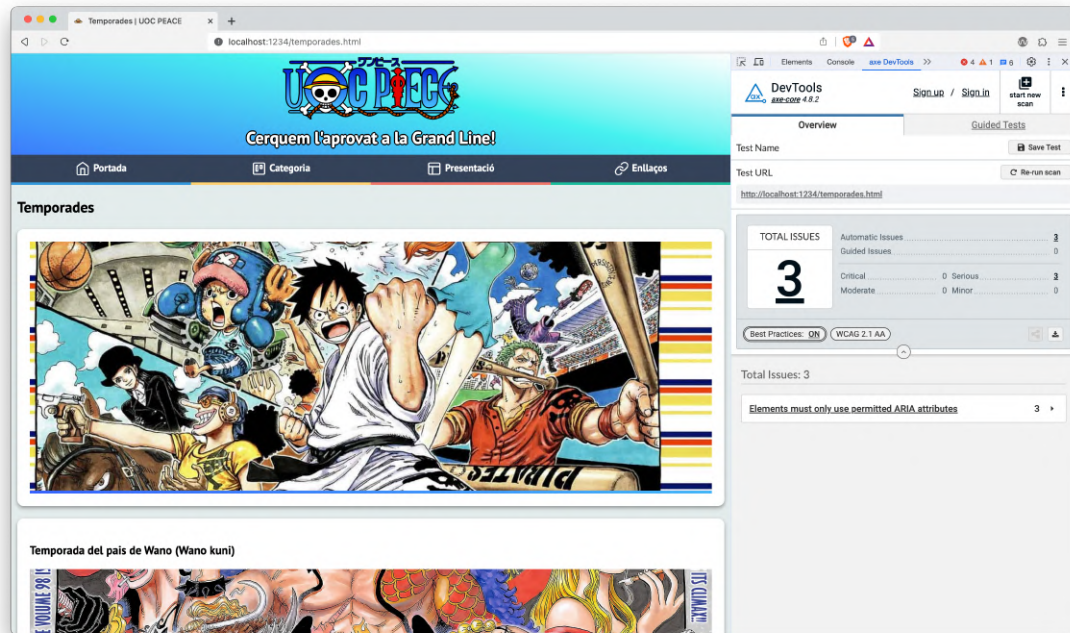
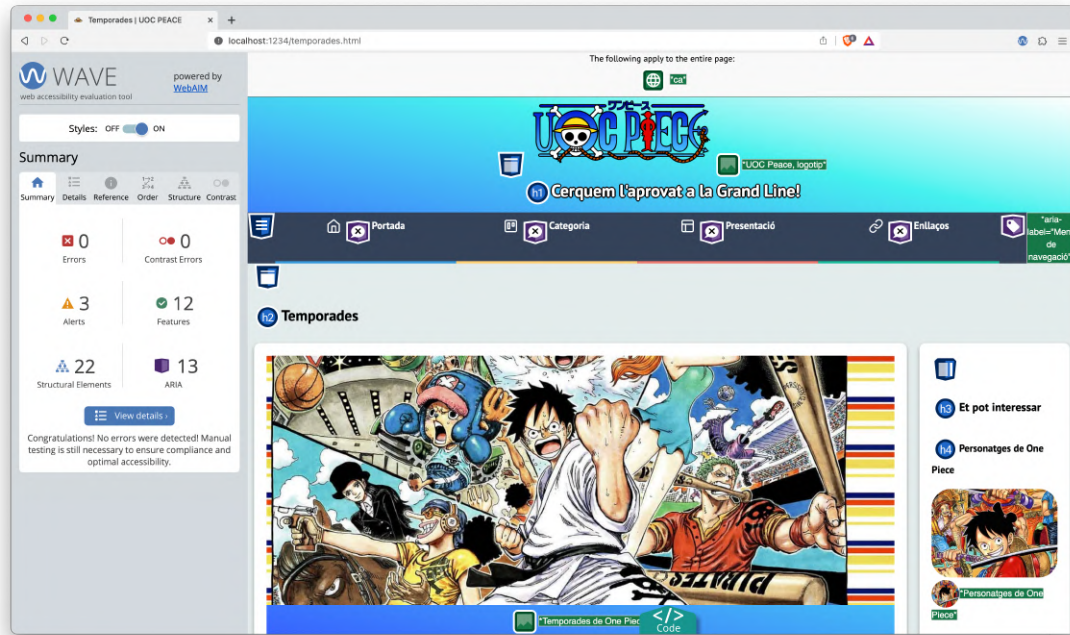




## Personatges



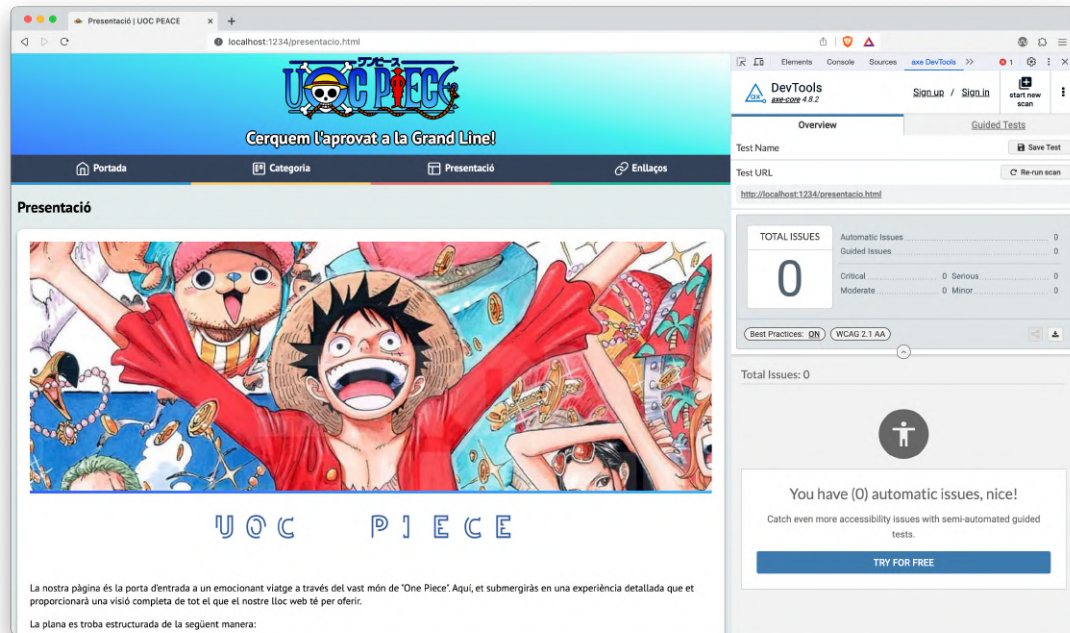
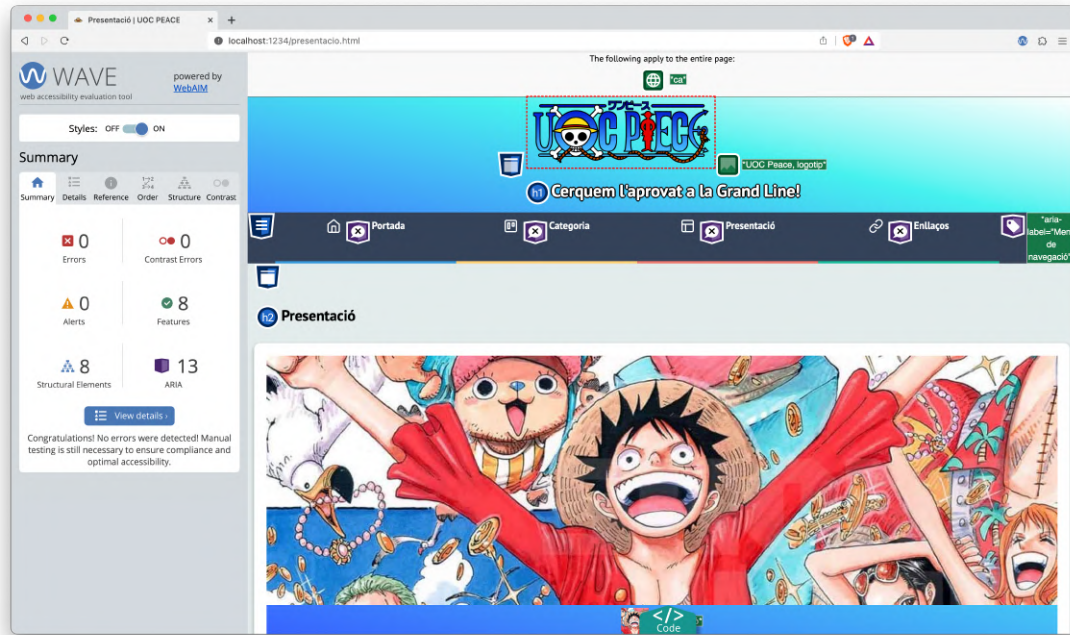
## Temporades



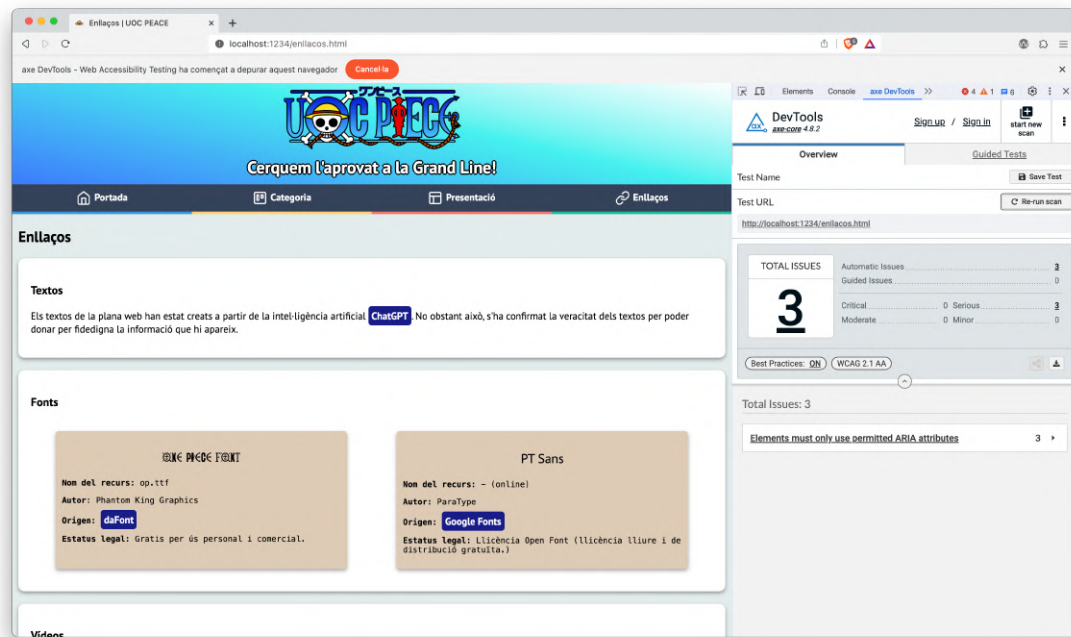
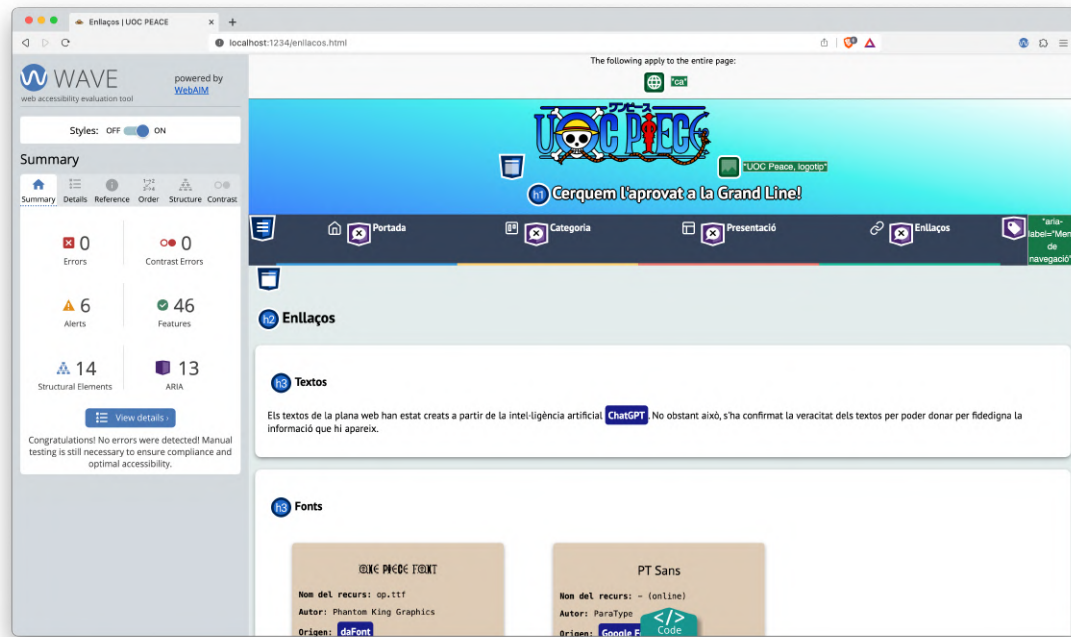
3 alert/issues pels vídeos de YouTube.



## Presentació



## Enllaços



WAVE detecta 6 vídeos de YouTube, ja que per cada vídeo l'enquestam i també posam l'enllaç.

Axe DevTools detecta 3 vídeos de Youtube (els enquestats).