

# Resum

Aquest treball final de grau té com a objectiu investigar l'accessibilitat web per a persones amb discapacitat visual, amb especial atenció a la ceguesa total, i desenvolupar una eina web pensada per acompañar i facilitar la tasca dels desenvolupadors en la creació d'aplicacions accessibles. Actualment, l'aplicació de bones pràctiques d'accessibilitat pot ser un procés extens i poc intuïtiu, la qual cosa sovint en dificulta la incorporació en projectes reals. L'eina desenvolupada ofereix una recollida interactiva i didàctica de recomanacions i guies pràctiques, complementades amb exemples concrets implementats en tecnologies estàndard com HTML i CSS. Amb això, es pretén afavorir una millor comprensió de les tècniques d'accessibilitat i fomentar-ne la integració de manera senzilla i directa, contribuint així a la construcció d'una web més inclusiva i universal.

**Paraules clau:** Accessibilitat web, Discapacitat visual, Ceguesa, Desenvolupament web, Bones pràctiques, HTML, CSS, WAI-ARIA, Inclusió digital, Eines interactives

---

# Resumen

Este trabajo de fin de grado tiene como objetivo investigar la accesibilidad web para personas con discapacidad visual, con especial atención a la ceguera total, y desarrollar una herramienta web concebida para acompañar y facilitar la labor de los desarrolladores en la creación de aplicaciones accesibles. En la actualidad, la aplicación de buenas prácticas de accesibilidad puede convertirse en un proceso extenso y poco intuitivo, lo que a menudo dificulta su incorporación en proyectos reales. La herramienta desarrollada ofrece un recopilatorio interactivo y didáctico de recomendaciones y guías prácticas, complementadas con ejemplos concretos implementados en tecnologías estándar como HTML y CSS. Con ello, se pretende favorecer una mejor comprensión de las técnicas de accesibilidad y fomentar su integración de manera sencilla y directa, contribuyendo así a la construcción de una web más inclusiva y universal.

**Palabras clave:** Accessibilitat web, Discapacitat visual, Ceguesa, Desenvolupament web, Bones pràctiques, HTML, CSS, WAI-ARIA, Inclusió digital, Eines interactives

---

# Abstract

This final degree project aims to research web accessibility for people with visual impairments, with special attention to total blindness, and to develop a web tool designed to support and facilitate the work of developers in creating accessible applications. At present, the application of accessibility best practices can become a lengthy and unintuitive process, which often hinders their adoption in real projects. The developed tool provides an interactive and educational collection of recommendations and practical guidelines, complemented by concrete examples implemented with standard technologies such as HTML and CSS. The ultimate goal is to promote a better understanding of accessibility techniques and

to encourage their straightforward integration, thus contributing to the development of a more inclusive and universal web.

**Key words:** Accessibilitat web, Discapacitat visual, Ceguesa, Desenvolupament web, Bones pràctiques, HTML, CSS, WAI-ARIA, Inclusió digital, Eines interactives

---

# Índex

---

<b>Índex</b>	<b>v</b>
<b>Índex de figures</b>	<b>ix</b>
<b>Índex de taules</b>	<b>x</b>
<hr/>	
<b>1 Introducció</b>	<b>1</b>
1.1 Motivació . . . . .	2
1.2 Evolució històrica de l'accessibilitat web . . . . .	3
1.3 Objectius . . . . .	5
1.3.1 Objectius generals . . . . .	5
1.3.2 Objectius específics . . . . .	5
1.3.3 Objectius complementaris . . . . .	5
1.4 Metodologia per assolir els objectius . . . . .	6
1.4.1 Anàlisi i revisió bibliogràfica . . . . .	6
1.4.2 Definició dels requisits . . . . .	6
1.4.3 Disseny i prototipatge . . . . .	6
1.4.4 Implementació iterativa . . . . .	6
1.4.5 Documentació i difusió . . . . .	7
1.5 Estructura de la memòria . . . . .	7
<b>2 Estat de l'art</b>	<b>9</b>
2.1 Marc normatiu i estàndards internacionals . . . . .	9
2.2 Eines d'avaluació d'accessibilitat . . . . .	9
2.3 Tecnologies d'assistència: lectors de pantalla . . . . .	10
2.4 Casos reals i impacte de la inaccessibilitat . . . . .	10
2.5 Limitacions de les solicions actuals . . . . .	10
2.5.1 Comparació mala praxis vs bona praxis . . . . .	11
2.6 Ús de la intel·ligència artificial en accessibilitat . . . . .	13
<b>3 Accessibilitat web</b>	<b>15</b>
3.1 Estàndards d'accessibilitat . . . . .	15
3.1.1 WAI . . . . .	15
3.1.2 WAI-ARIA . . . . .	17
3.1.3 Us de Wai-Aria . . . . .	18
3.1.4 Millora de l'accessibilitat en aplicacions web complexes . . . . .	18
3.1.5 Principis bàsics abans d'utilitzar ARIA . . . . .	20
3.1.6 Exemples d'ús de WAI-ARIA . . . . .	21
3.1.7 Errors freqüents i com evitar-los . . . . .	25
3.2 Bones pràctiques d'accessibilitat web . . . . .	27
3.2.1 Etiquetatge semàntic i etiquetes descriptives (WCAG 2.4.6) . . . . .	27
3.2.2 Navegació per teclat (WCAG 2.1.1) . . . . .	28
3.2.3 Contrast de color (WCAG 1.4.3) . . . . .	28
3.2.4 Indicador de focus visible (WCAG 2.4.7) . . . . .	28

3.2.5	Text alternatiu per a contingut no textual (WCAG 1.1.1) . . . . .	28
3.3	Lectors de pantalla . . . . .	29
3.3.1	Com els lectors de pantalla interpreten el DOM . . . . .	29
3.3.2	Voice Over . . . . .	31
3.3.3	Ús de VoiceOver . . . . .	34
3.3.4	JAWS . . . . .	36
<b>4</b>	<b>Anàlisi del problema</b>	<b>37</b>
4.1	Recursos i dificultats dels desenvolupadors en l'accessibilitat web . . . . .	37
4.2	Com utilitzen les persones no vidents els ordinadors i els lectors de pantalla . . . . .	38
4.3	Avaluació de l'accessibilitat web . . . . .	39
4.3.1	WAVE . . . . .	39
4.3.2	Lighthouse . . . . .	40
4.3.3	Exemple d'ús de Google Lighthouse . . . . .	40
4.3.4	Exemple d'ús de WAVE (Web Accessibility Evaluation Tool)	41
4.4	Requisits de l'aplicació . . . . .	42
4.4.1	Requisits funcionals . . . . .	42
4.4.2	Requisits no funcionals . . . . .	43
4.4.3	Traçabilitat (resum) . . . . .	43
<b>5</b>	<b>Disseny de la solució i Implementació</b>	<b>45</b>
5.1	Funcionament de l'aplicació i estructura . . . . .	45
5.2	Vista de l'aplicació . . . . .	47
5.3	Decisions de disseny . . . . .	51
5.3.1	Tipografia i llegibilitat . . . . .	51
5.3.2	Estructura i jerarquia visual . . . . .	52
5.3.3	Tooltip inicial i canvi a menú desplegable . . . . .	52
5.3.4	Consistència i reconeixement . . . . .	52
5.4	Metodologia de desenvolupament . . . . .	53
5.4.1	Fases del procés . . . . .	53
5.4.2	Tecnologies utilitzades . . . . .	53
5.5	Tecnologies emprades . . . . .	53
5.5.1	HTML5 . . . . .	54
5.5.2	CSS3 . . . . .	54
5.5.3	JavaScript . . . . .	54
5.5.4	WAI-ARIA . . . . .	54
5.5.5	Eines de validació . . . . .	54
5.5.6	Entorn de desenvolupament . . . . .	54
5.5.7	Redacció de la memòria . . . . .	55
<b>6</b>	<b>Conclusions i Treball Futur</b>	<b>57</b>
6.1	Conclusions generals . . . . .	57
6.1.1	Aportacions principals . . . . .	58
6.1.2	Limitacions . . . . .	58
6.1.3	Lliçons apreses . . . . .	58
6.1.4	Relació amb les assignatures del grau . . . . .	59
6.2	Treball futur . . . . .	60
6.2.1	Accessibilitat per a ceguesa parcial (baixa visió) . . . . .	60
6.2.2	Mode d'alt contrast . . . . .	60
6.2.3	Paletes de colors per a daltonisme . . . . .	60

6.2.4 Tipografies recomanades per a facilitar la visió . . . . .	61
6.2.5 Internacionalització i canvi d'idioma . . . . .	61
6.3 Tancament . . . . .	61
<b>Annex. Objectius de Desenvolupament Sostenible (ODS)</b>	<b>64</b>
<b>Bibliografia</b>	<b>67</b>

# Índex de figures

---

1.1	Imatge de la evolució WCAG [1]	4
2.1	Web amb un html correcte	11
2.2	codi Web amb un html correcte	12
2.3	Web amb un html incorrecte	12
2.4	codi Web amb un html incorrecte	13
3.1	Aria-describedby exemple	19
3.2	Lectura de botó 1	21
3.3	Imatge utilitzant alt	22
3.4	Imatge utilitzant alt quan no existeix	22
3.5	Imatge utilitzant aria-label	23
3.6	Imatge utilitzant aria-label quan no existeix	23
3.7	aria-label	24
3.8	Formulari utilitzant aria-labelledby	25
3.9	Recuadre VoiceOver	31
3.10	voiceOver h1	32
3.11	voiceOver ul	32
3.12	Taula accessible vista amb VoiceOver	32
3.13	Botó en logotipo utilitzant aria-label	34
4.1	Google Lighthouse en el Pais.com	40
4.2	Web del projecte analitzada amb Lighthouse	41
4.3	Web del projecte analitzada amb WAVE	42
5.1	Diagrama de navegació	46
5.2	Pàgina de inici	47
5.3	Pàgina de inici	48
5.4	Element del formulari	48
5.5	Desplegable tancat	49
5.6	Desplegable obert	49
5.7	Botó de informació	50
5.8	Pàgina d'exemples WAI-ARIA	50
5.9	Pàgina de les eines de validació	51
5.10	Codi amb tooltip	52

# Índex de taules

3.1	Combinacions de teclat per a VoiceOver . . . . .	34
3.2	Llegir per seccions concretes . . . . .	35
3.3	Combinacions de tecles per navegar amb JAWS . . . . .	36
6.1	Grau de relació del treball amb els Objectius de Desenvolupament Sostenible (ODS) . . . . .	64

---

# CAPÍTOL 1

## Introducció

---

L'accessibilitat web és un pilar fonamental per garantir que totes les persones, independentment de les seues capacitats físiques, sensorials o cognitives, puguen accedir, comprendre i interactuar amb la informació i els serveis disponibles en línia. En un context en què la digitalització abasta àmbits tan diversos com l'educació, la sanitat, l'administració pública i el comerç electrònic, assegurar la inclusió digital esdevé una responsabilitat compartida entre desenvolupadors, dissenyadors, empreses i institucions públiques.

El concepte d'accessibilitat digital no és nou: ja des de finals dels anys noranta, el World Wide Web Consortium (W3C) va establir les primeres *Web Content Accessibility Guidelines* (WCAG), amb l'objectiu de proporcionar un marc comú que orientara el desenvolupament web cap a la inclusivitat. Amb el temps, aquestes guies han evolucionat fins a l'actual versió 2.1, incorporant criteris que abasten noves tecnologies, dispositius i formes d'interacció[2].

Malgrat aquests avanços, les estadístiques mostren que una part significativa de les pàgines web no compleixen encara amb els estàndards mínims d'accessibilitat [3], [4]. Això implica que milions de persones amb discapacitats visuals, auditives, motrius o cognitives es troben amb barreres que dificulten o impedeixen l'ús de serveis bàsics. Aquesta realitat no només vulnera drets fonamentals reconeguts per organismes internacionals, sinó que també limita el potencial econòmic i social de la transformació digital.

En el cas concret de les persones amb ceguesa total, les barreres digitals poden ser especialment restrictives. L'eficàcia de tecnologies d'assistència com els lectors de pantalla depèn, en gran mesura, de la qualitat del codi i del compliment d'estàndards semàntics. Una etiqueta mal utilitzada, una imatge sense text alternatiu o una estructura de navegació confusa poden transformar una experiència d'usuari acceptable en una experiència frustrant o, directament, inaccessible [5].

És en aquest context que es planteja aquest Treball Final de Grau (TFG), amb la voluntat d'aportar una eina pràctica, clara i immediata que facilite la implementació de bones pràctiques d'accessibilitat web. L'objectiu no és substituir altres eines o metodologies existents, sinó complementar-les amb una solució que puga ser utilitzada tant en entorns professionals com formatius, amb un enfocament que prioritze la rapidesa de consulta, la simplicitat d'ús i la fiabilitat de la informació.

Aquest projecte, per tant, no només s'inscriu en el marc tècnic del desenvolupament web, sinó també en el compromís ètic i social de contribuir a una web més inclusiva i accessible per a tothom.

## 1.1 Motivació

---

En l'actualitat, desenvolupar pàgines web accessibles per a persones no vidents continua sent un repte considerable per a molts professionals del sector. Tot i que existeixen guies i estàndards consolidats, com les WCAG, el procés d'aplicar bones pràctiques no sempre és directe ni senzill. El desenvolupament accessible requereix un coneixement específic i, sovint, una inversió addicional de temps i recursos que moltes empreses no estan disposades o no poden assumir.

Aquesta situació genera un problema real: nombroses pàgines web no compleixen els criteris bàsics d'accessibilitat, la qual cosa provoca que persones amb discapacitat visual total o parcial es troben amb barreres digitals que limiten el seu dret a la informació, la comunicació i la participació en igualtat de condicions. La manca d'accessibilitat no és només una qüestió tècnica, sinó també un problema social i ètic.

Com a estudiant d'enginyeria informàtica, he pogut comprovar tant en entorns acadèmics com en projectes reals que les eines disponibles per millorar l'accessibilitat soLEN dividir-se en dos extrems: o bé són molt bàsiques i poc flexibles, o bé són molt potents però requereixen un alt nivell de coneixement tècnic per a ser aproveitades. A més, moltes depenen de la connexió a Internet o de serveis externs, fet que en determinats entorns corporatius o de desenvolupament ràpid pot resultar poc pràctic, com podrien ser les d'Intel·ligència Artificial, les quals en entorns corporatius solent tindre alguna limitació o directament no estar disponibles per temes de confidencialitat.

La motivació d'aquest treball naix de la necessitat de crear una solució intermèdia, que combine la simplicitat d'ús amb la solidesa de la informació proporcionada. Volia desenvolupar una eina que servira tant per a guiar desenvolupadors novells en la correcta implementació de bones pràctiques, com per a proporcionar una consulta ràpida a professionals que necessiten recordar o verificar aspectes concrets de l'accessibilitat HTML.

També he comprovat que, tot i els avanços de la intel·ligència artificial en l'anàlisi i correcció de contingut web, no sempre és viable ni eficient dependre d'aquesta per a tasques ràpides o entorns amb restriccions d'accés. Disposar d'una eina estàtica, clara i immediata pot resultar d'utilitat.

Per tot això, aquest treball final de grau pretén aportar un recurs pràctic i immediat que, a més de complir amb les recomanacions internacionals, tinga un enfocament pedagògic, fomentant la conscienciació sobre la importància de l'accessibilitat web i facilitant la seua aplicació des del primer moment del desenvolupament.

## 1.2 Evolució històrica de l'accessibilitat web

---

L'accessibilitat web, entesa com la capacitat d'un lloc web o aplicació per ser utilitzat per qualsevol persona independentment de les seues capacitats físiques o sensorials, és un concepte que ha evolucionat en paral·lel al desenvolupament d'Internet.

### Els inicis: una web per a tothom

Als anys noranta, amb la popularització de la World Wide Web, la filosofia original proposada per Tim Berners-Lee ja defensava que “el poder de la web està en la seu universalitat”[6]. Tot i aquesta visió, la manca d'estàndards clars provocava que cada desenvolupador implementara els seus propis criteris d'usabilitat, la qual cosa derivava en experiències molt desiguals per a usuaris amb discapacitat.

### Primers estàndards: WCAG 1.0

El 1999, el *World Wide Web Consortium* (W3C), a través de la seva *Web Accessibility Initiative* (WAI), va publicar la primera versió de les **Web Content Accessibility Guidelines** (WCAG 1.0). Aquest document establia 14 directrius bàsiques centrades en aspectes com l'ús d'alternatives textuais, la claredat del contingut i la separació entre presentació i estructura.

### Consolidació i ampliació: WCAG 2.0

El 2008 es va publicar la versió 2.0, que reorganitzava els criteris en quatre principis fonamentals: perceptible, operable, comprensible i robust. Aquesta versió va introduir el concepte de “nivells de conformitat” (A, AA i AAA) i va establir les bases per a una evaluació més objectiva i mesurable.

### Adaptació a noves tecnologies: WCAG 2.1

Amb l'extensió de l'ús de dispositius mòbils i tecnologies tàctils, el 2018 es va publicar WCAG 2.1, afegint criteris específics per a millorar l'experiència en pantalles reduïdes, suports tàctils i per a col·lectius amb discapacitat cognitiva o baixa visió.

### Tendències actuals i futur

En l'actualitat, s'està treballant en la versió 2.2 i en les futures WCAG 3.0, que incorporaran un model més flexible per avaluar l'accessibilitat i que aspiren a cobrir un espectre encara més ampli de situacions i dispositius. Aquestes evolucions responen a la necessitat constant d'adaptar-se a una web cada vegada més interactiva, multimèdia i personalitzada.[7]

A la figura ?? es veu aquesta evolució.

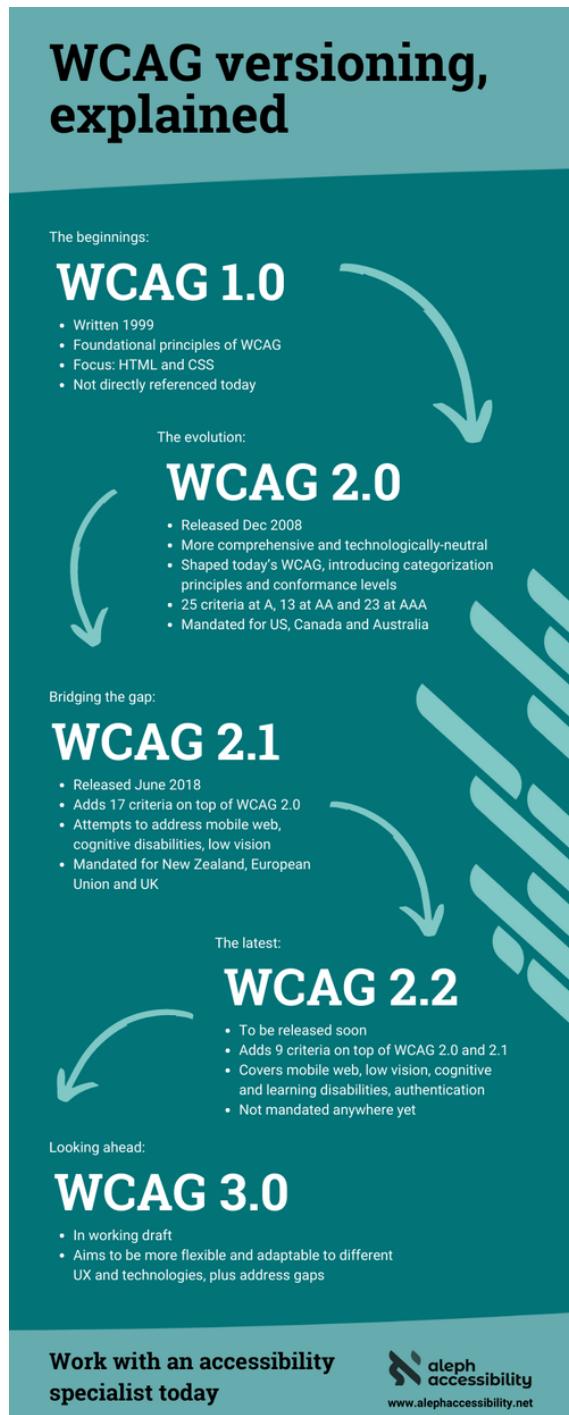


Figura 1.1: Imatge de la evolució WCAG [1]

## Impacte social i legal

Aquesta evolució no ha estat només tècnica, sinó també social i jurídica. A mesura que els serveis digitals s'han convertit en essencials per a la vida quotidiana, molts països han incorporat l'accessibilitat web a les seues lleis i regulacions, estableint terminis i sancions per al seu incompliment. Això reforça la idea que l'accessibilitat no és opcional, sinó una obligació legal i un compromís ètic.[5, 8, 7, 9, 10]

## 1.3 Objectius

---

L'objectiu principal d'aquest treball final de grau és desenvolupar una aplicació web senzilla, estàtica i fàcilment consultable que permeta guiar els desenvolupadors en la creació de llocs web accessibles per a persones amb discapacitat visual, especialment aquelles amb ceguesa total. L'aplicació es concep com un recurs pràctic i immediat que recopile bones pràctiques d'accessibilitat i les presente de manera clara i estructurada.

### 1.3.1. Objectius generals

- Contribuir a la millora de la inclusió digital a través d'una eina que facilite la implementació de criteris d'accessibilitat web.
- Promoure la conscienciació dels desenvolupadors sobre la importància de l'accessibilitat com a part inherent del procés de disseny i desenvolupament.
- Oferir un recurs que puga ser utilitzat tant en entorns tant acadèmics com professionals, servint com a material de formació i consulta.

### 1.3.2. Objectius específics

- Recopilar i sistematitzar bones pràctiques d'accessibilitat basades en els estàndards WCAG 2.1 i 2.2, així com en recomanacions de la WAI.
- Implementar exemples pràctics amb diferents etiquetes HTML que mostren el seu ús correcte i la seua adaptació per a persones usuàries de lectors de pantalla.
- Incorporar funcionalitats interactives, com ara botons d'informació i eines per copiar fragments de codi, que faciliten l'aprenentatge i l'aplicació immediata.
- Assegurar que el disseny visual de l'aplicació compleix els criteris de contrast i llegibilitat, d'acord amb les directrius d'accessibilitat.
- Complementar l'aplicació amb altres eines d'avaluació disponibles (WAVE, axe, Lighthouse) per tal comprovar l'accessibilitat dels espais web.

### 1.3.3. Objectius complementaris

- Difondre la importància de l'accessibilitat com a compromís ètic i social, més enllà del compliment legal.
- Plantejar la possibilitat de futures ampliacions, com ara la incorporació de suport multilingüe o la integració amb eines automàtiques d'anàlisi o ampliar l'accessibilitat per a ceguesa parcial.

- Establir una base documental i metodològica que puga servir de referència per a treballs futurs en l'àmbit de l'accessibilitat web.

## 1.4 Metodologia per assolir els objectius

---

Per aconseguir els objectius definits en aquest treball, s'ha adoptat una metodologia basada en el desenvolupament iteratiu i en la combinació d'anàlisi documental, implementació pràctica i validació mitjançant proves amb usuaris. Aquesta metodologia ha permés avançar de manera estructurada, garantint que cada objectiu quedara degudament cobert.

### 1.4.1. Anàlisi i revisió bibliogràfica

En una primera fase s'ha dut a terme una revisió exhaustiva de la literatura existent, incloent-hi els estàndards WCAG [11], la normativa vigent a nivell europeu [12] i estatal [13], així com articles i guies elaborades per organitzacions com el W3C [14] i entitats especialitzades en accessibilitat digital [15, 16]. Aquesta revisió ha servit com a base per identificar les bones pràctiques que posteriorment s'han incorporat a l'aplicació.

### 1.4.2. Definició dels requisits

A partir de l'anàlisi previ, s'han definit els requisits funcionals i no funcionals de l'aplicació. Entre els funcionals destaca la necessitat d'ofrir exemples pràctics d'etiquetes HTML accessibles i d'incloure informació addicional en format didàctic. Entre els no funcionals, s'ha establert la simplicitat d'ús, la claredat visual i el compliment dels criteris d'accessibilitat com a elements essencials.

### 1.4.3. Disseny i prototipatge

La següent fase ha consistit en l'elaboració d'esbossos i prototips de la interfície, valorant diferents opcions de presentació i navegació. S'han tingut en compte aspectes com la jerarquia visual, la disposició dels elements i la compatibilitat amb els lectors de pantalla. Aquesta fase ha permés refinjar les decisions de disseny abans de la implementació final.

### 1.4.4. Implementació iterativa

L'aplicació s'ha desenvolupat mitjançant un procés iteratiu, afegint progressivament funcionalitats i validant-ne el comportament. La implementació s'ha realitzat utilitzant HTML5, CSS3 i JavaScript pur, sense dependències externes, per tal de garantir la màxima lleugeresa i compatibilitat. Cada iteració ha inclòs proves d'accessibilitat amb eines automàtiques i amb lectors de pantalla.

### 1.4.5. Documentació i difusió

Finalment, s'ha elaborat la documentació corresponent, incloent-hi aquesta memòria, amb l'objectiu de descriure no només el procés tècnic seguit, sinó també les reflexions ètiques i socials que han motivat el projecte. La documentació pretén ser també un recurs útil per a futurs investigadors i professionals interessats en l'accessibilitat web.

## 1.5 Estructura de la memòria

---

La memòria s'organitza en sis capítols que segueixen una progressió lògica, des de la motivació inicial fins a les conclusions i perspectives de futur. A continuació es descriu el contingut de cadascun:

- **Introducció:** Presenta el context del treball, la motivació i els objectius plantejats, així com la metodologia emprada per al seu desenvolupament.
- **Estat de l'art:** Ofereix una revisió de les eines i recursos existents relacionats amb l'accessibilitat web, posant de manifest les seues funcionalitats, limitacions i l'espai d'oportunitat que justifica la creació de l'eina desenvolupada.
- **Accessibilitat web:** Descriu els fonaments teòrics i normatius en matèria d'accessibilitat, incloent les WCAG, les especificacions WAI-ARIA i altres iniciatives rellevants a escala internacional i europea.
- **Anàlisi del problema:** Examina les dificultats concretes que enfronten les persones amb discapacitat visual a l'hora de navegar per Internet, així com els reptes que això planteja als desenvolupadors web i els recursos de què disposen actualment.
- **Disseny de la solució i Implementació:** Descriu l'eina web desenvolupada en el marc d'aquest treball, incloent les decisions de disseny, les tecnologies emprades, l'estructura de les pàgines i els exemples pràctics implementats. També s'inclouen captures i fragments de codi per a facilitar-ne la comprensió.
- **Conclusions i Treball Futur:** Recull les conclusions generals del projecte, destacant-ne les aportacions i limitacions. A més, es proposen línies de treball futur que permetrien ampliar l'abast i la utilitat de l'eina.

Aquesta estructura garanteix un recorregut clar i coherent: primer es presenta el marc teòric i l'anàlisi del context, després la solució proposada i, finalment, les conclusions i possibles línies de continuïtat.

---

## CAPÍTOL 2

# Estat de l'art

---

L'accessibilitat web ha evolucionat notablement en les darreres dècades, passant de ser un aspecte poc considerat a constituir un requisit fonamental en el desenvolupament de llocs i aplicacions. Aquesta evolució ha estat impulsada per diversos factors: la conscienciació social sobre la inclusió digital, la creació d'estàndards internacionals i la implementació de legislacions que obliguen a complir criteris específics d'accessibilitat.

### 2.1 Marc normatiu i estàndards internacionals

---

Els estàndards de referència són les **Web Content Accessibility Guidelines** (WCAG), actualment en la versió 2.1, elaborades pel W3C. Aquestes guies estableixen criteris per a garantir que el contingut web siga perceptible, operable, comprensible i robust. A Europa, la norma harmonitzada **EN 301 549** defineix els requisits d'accessibilitat per a productes i serveis TIC, incloent aplicacions web, i és d'aplicació obligatòria per a organismes del sector públic i, en determinats casos, per a entitats privades.

A Espanya, la **Llei 11/2007** d'accés electrònic dels ciutadans als serveis públics i el **Reial decret 1112/2018** [17, 18] estableixen l'obligació de complir amb aquests requisits. Aquest marc normatiu implica que l'accessibilitat no és únicament una qüestió ètica o de bones pràctiques, sinó un deure legal amb implicacions jurídiques i econòmiques.

### 2.2 Eines d'avaluació d'accessibilitat

---

Diverses eines han estat desenvolupades per a facilitar la detecció de problemes d'accessibilitat uns exemples són:

- **WAVE**: ofereix una anàlisi visual de la pàgina i assenyala errors, advertències i característiques d'accessibilitat. Tot i ser intuïtiva, requereix certa interpretació per part de l'usuari.
- **axe DevTools**: extensió molt completa utilitzada en entorns professionals, amb informes detallats però una corba d'aprenentatge elevada.

- **Lighthouse**: integrat en Google Chrome, proporciona puntuacions d'accésibilitat juntament amb altres mètriques com rendiment i SEO.
- **Tenon.io i Siteimprove**: solucions comercials amb informes exhaustius i capacitat de seguiment continu.

Aquestes eines, tot i ser útils, sovint presenten informació en formats poc accessibles per a desenvolupadors amb poca experiència o en situacions on la ràpidesa és clau.[8, 9]

## 2.3 Tecnologies d'assistència: lectors de pantalla

---

Els lectors de pantalla són eines essencials per a persones amb ceguesa total. Programes com **NVDA** (Windows), **VoiceOver** (macOS i iOS) o **JAWS** interpreten el contingut i el presenten en forma d'àudio o en línies Braille. La seua eficàcia depèn directament de la qualitat del codi HTML i de l'ús correcte d'etiquetes semàntiques i atributs ARIA.

Els estudis mostren que una gran part de les dificultats experimentades pels usuaris de lectors de pantalla són degudes a etiquetes mal definides, imatges sense text alternatiu i estructures no semàntiques.[19, 20]

## 2.4 Casos reals i impacte de la inaccessibilitat

---

Hi ha nombrosos exemples de llocs web que han patit conseqüències per no complir amb els estàndards d'accessibilitat. El cas de *Domino's Pizza* als Estats Units (2019) [21] és el perfecte exemple: l'empresa va ser demandada per un usuari cec que no podia realitzar comandes a través de la seua web i aplicació mòbil. El Tribunal Suprem va rebutjar el recurs de l'empresa, establint un precedent sobre la responsabilitat legal en matèria d'accessibilitat.

A nivell europeu, diverses institucions públiques han hagut d'afrontar sancions o processos de millora accelerada per no complir amb la EN 301 549. Aquestes situacions demostren que la inaccessibilitat no només afecta la inclusió social, sinó també la reputació corporativa i pot comportar costos econòmics significatius [22].

## 2.5 Limitacions de les solucions actuals

---

Tot i els avanços, les solucions existents sovint requereixen:

- Coneixements tècnics previs per a complir amb els estàndards.
- Temps per a aprendre la documentació oficial i aplicar-la.
- Utilització de la intel·ligència artificial

Aquesta realitat crea un espai per a eines que, com la proposta en aquest treball, proporcionen informació immediata, estructurada i fiable sobre bones pràctiques, sense dependències externes i amb un enfocament pedagògic.

### 2.5.1. Comparació mala praxis vs bona praxis

Tota aplicació web s'estructura mitjançant jerarquies semàntiques i diferenciació de elements html. Si no es respecten les normes del llenguatge html, les aplicacions web son tremendament complicades de comprendre i no ofereixen una experiència d'usuari optima. Per il·lustrar-ho, s'han posat dos exemples: en les figures 2.1 i 2.3 es veu com per a un usuari sense discapacitat visual, el contingut de la pagina web és identic en ambdos casos. Però com es veu a les figures 2.2 i 2.4, el codi és different, en aquest cas per a les persones amb discapacitat visual l'opció de la web 2.4 resultaria molt complicat d'entendre o directament impossible mitjançant el lector de pantalla.

#### Títol de la pàgina web ben feta

Paragraf numero 1 lorem ipsum - .àsfjgas

segon Paragraf

1. ací tenim
2. una llista
3. ordenada

#### Aci tenim un subtítol

altre paràgraf Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

#### Segon subtítol

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

**Figura 2.1:** Web amb un html correcte

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Pagina ben feta</title>
</head>
<body>
    <h1>Títol h1 de la pàgina web </h1>

    <p>Paragraf numero lorem ipsum -..àsfjgas</p>

    <p>segon Paragraf</p>

    <ol>
        <li>aci tenim</li>
        <li>una llista</li>
        <li>ordenada</li>
    </ol>

    <h2>Aci tenim un subtítol</h2>

    <p>
        altre paragraf Lorem Ipsum is simply dummy text of the
        printing and typesetting industry. Lorem Ipsum has been
        the industry's standard dummy text ever since the 1500s,
        when an unknown printer took a galley of type and scrambled
        it to make a type specimen book. It has survived not only five
        centuries, but also the leap into electronic typesetting, remaining e
        ssentially unchanged. It was popularised in the 1960s with the release
        of Letraset sheets containing Lorem Ipsum passages, and more recently
        with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.
    </p>

    <h2>Segón subtítol</h2>

```

**Figura 2.2:** codi Web amb un html correcte

## Títol de la pàgina web mal feta

Paragraf numero 1 lorem ipsum -..àsfjgas

segon Paragraf

1. aci tenim

2. una llista

3. ordenada

### Aci tenim un subtítol

altre paragraf Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining e ssentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

### Segón subtítol

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, r remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets

**Figura 2.3:** Web amb un html incorrecte

```

<span style="font-size: 2.5em">Titol de la pàgina web mal feta</span> <br /><br />
Paràgraf numero 1 lorem ipsum -..àsfjgas
<br /><br />
segon Paràgraf
<br /><br />
1. ací tenim
<br /><br />
2. una llista
<br /><br />
3. ordenada
<br /><br />
<span style="font-size: 2em">Aci tenim un subtítol</span>
<br /><br />
altre paràgraf Lorem Ipsum is simply dummy text of the
printing and typesetting industry. Lorem Ipsum has been
the industry's standard dummy text ever since the 1500s,
when an unknown printer took a galley of type and scrambled
it to make a type specimen book. It has survived not only five
centuries, but also the leap into electronic typesetting, remaining e
ssentially unchanged. It was popularised in the 1960s with the release
of Letraset sheets containing Lorem Ipsum passages, and more recently
with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.
<br /><br />
<span style="font-size: 2em">Segon subtítol</span>
<br /><br />
Lorem Ipsum is simply dummy text of the printing and typesetting industry.
Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when
an unknown printer took a galley of type and scrambled it to make a type specimen book.
It has survived not only five centuries, but also the leap into electronic typesetting, r
emaining essentially unchanged. It was popularised in the 1960s with the release of Letraset
sheets containing Lorem I
psum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

```

**Figura 2.4:** codi Web amb un html incorrecte

Aquest es un clar exemple de perquè es fonamental utilitzar un perfecte html a l'hora de desenvolupar i no utilitzar solucions de codi o d'estil que tindrien visualment el mateix efecte però no a nivell estructural.

## 2.6 Ús de la intel·ligència artificial en accessibilitat

En els darrers anys, la **intel·ligència artificial (IA)** s'ha consolidat com una de les eines més potents i utilitzades en l'àmbit del desenvolupament web i, especialment, en la revisió de criteris d'accessibilitat. Les solucions basades en IA permeten analitzar grans quantitats de codi i contingut en qüestió de segons, oferint suggeriments immediats per corregir errors habituals com la manca de text alternatiu, l'ús incorrecte de rols ARIA o problemes de contrast de color [23]. Aquesta rapidesa i capacitat d'automatització converteixen la IA en una alternativa còmoda i eficient per als desenvolupadors, que poden integrar-la directament en els seus fluxos de treball i obtenir resultats immediats sense necessitat de realitzar una auditoria manual completa.

Per contra, aquesta apparent comoditat presenta diverses **limitacions**.

En primer lloc, les eines d'IA solen funcionar com a caixa negra, és a dir, proporcionen resultats sense explicar sempre el raonament o la normativa concreta en què es basen. Això pot generar dependència excessiva i una falta de comprensió real per part dels desenvolupadors sobre els principis d'accessibilitat [24].

En segon lloc, a nivell corporatiu, l'ús d'IA pot veure's restringit per motius legals o de seguretat: moltes empreses no poden enviar codi font ni continguts interns a plataformes externes per qüestions de confidencialitat i protecció de dades [25]. En aquests casos, la IA deixa de ser una solució viable i obliga a recórrer a alternatives més tradicionals, com per exemple el recurs objectiu d'aquest treball.

A més, cal assenyalar que la IA presenta limitacions qualitatives. Tot i que detecta errors evidents, sovint no és capaç d'avaluar la qualitat semàntica d'una pàgina ni l'experiència subjectiva d'un usuari amb discapacitat.

Això significa que, encara que una pàgina passe els filtres automàtics, no necessàriament oferirà una experiència òptima per a tots els usuaris [26].

Per tant, la IA és útil com a eina de suport, però no substitueix ni la necessitat de coneixements tècnics per part dels desenvolupadors ni la validació manual amb lectors de pantalla i eines d'avaluació experta.

En aquest context, la creació d'aplicacions i guies estàtiques d'accessibilitat continua sent de gran valor. A diferència de la IA, aquestes eines proporcionen informació clara, immediata i no dependent de tercers, permetent als desenvolupadors consultar directament bones pràctiques i adaptar-les al seu projecte sense riscos de privacitat ni dependència tecnològica.

---

# CAPÍTOL 3

## Accessibilitat web

---

Aquest apartat aprofundeix en els conceptes fonamentals relacionats amb l'accessibilitat web, explicant-ne els principis, les tècniques i els criteris que han de guiar el desenvolupament d'aplicacions digitals inclusives. S'analitza com les persones amb diferents tipus de discapacitat interactuen amb la web i quines barreres poden trobar, posant especial atenció als lectors de pantalla i a altres tecnologies d'assistència. També es ressalta la importància social i legal de garantir l'accessibilitat en entorns digitals.

### 3.1 Estàndards d'accessibilitat

---

Aquest treball es basa en la documentació oficial publicada pel *World Wide Web Consortium* (W3C), que constitueix l'autoritat de referència en matèria d'estàndards per al desenvolupament web. A més, cal assenyalar que als Estats Units existeix la *Section 508* de la Rehabilitation Act, i que a Europa directives com la 2016/2102 i l'*European Accessibility Act* també regulen l'accessibilitat web. Totes aquestes normatives, es basen en les pautes de la *Web Accessibility Initiative* (WAI) del W3C.

El **World Wide Web Consortium (W3C)** és un consorci internacional que té com a missió desenvolupar i promoure estàndards oberts per al World Wide Web [27]. La seua creació va respondre a la necessitat de garantir una major compatibilitat i interoperabilitat entre els diferents actors de la indústria tecnològica a l'hora d'adoptar noves especificacions web. Abans de la seu existència, ja es patien problemes de fragmentació derivats de les versions d'HTML pròpies de cada venedor, la qual cosa generava dificultats per assegurar una experiència d'ús homogènia. Tot i els esforços del consorci, cal destacar que aquests problemes de compatibilitat encara persisteixen en determinades situacions [28]. Cal dir que aquests problemes no s'han solucionat per complet[29].

#### 3.1.1. WAI

El compromís del *World Wide Web Consortium* (W3C) de portar la Web al seu màxim potencial inclou garantir un alt grau d'usabilitat per a les persones amb discapacitats. La **Iniciativa d'Accessibilitat Web (WAI)** és una iniciativa del W3C

que desenvolupa estratègies, estàndards i recursos de suport per tal que la Web siga accessible per a tothom [30, 31].

El treball de la WAI es desenvolupa a través del procés de consens del W3C, que involucra múltiples parts interessades en l'accessibilitat web. Entre aquestes s'inclouen empreses de la indústria, organitzacions de persones amb discapacitat, administracions públiques, institucions acadèmiques i grups de recerca especialitzats en accessibilitat digital.

L'objectiu del W3C és que la Web i les tecnologies relacionades siguin accessibles de manera que totes les persones, independentment de les seues capacitats, puguen participar en igualtat de condicions en l'entorn digital. Açò implica diversos aspectes fonamentals:

- L'harmonització global dels estàndards i directrius d'accessibilitat.
- La inclusió de requisits d'accessibilitat en totes les especificacions tècniques.
- El suport a l'accessibilitat en les eines que creen i distribueixen contingut digital.
- L'adquisició de coneixements rellevants d'accessibilitat per part de totes les persones implicades en el desenvolupament i gestió de productes digitals.
- La consideració de l'accessibilitat des del disseny inicial de noves tecnologies, productes i continguts, és a dir, que siguin "nascuts accessibles".

## **Objectius d'alt nivell de la WAI**

La WAI defineix un conjunt d'objectius estratègics que orienten el seu treball [32]:

### **1. Especificacions i tecnologies accessibles:**

- Assegurar que les especificacions del W3C contemplen l'accessibilitat.
- Proporcionar orientació sobre accessibilitat en altres especificacions i tecnologies.

### **2. Normes internacionals d'accessibilitat:**

- Facilitar la cooperació internacional per afavorir l'adopció global de les normes d'accessibilitat.
- Desenvolupar especificacions tècniques de suport, com WAI-ARIA i les assignacions d'API d'accessibilitat.
- Elaborar estàndards i directrius específiques per a eines i continguts digitals.
- Proporcionar guies addicionals, com la documentació complementària de les WCAG.

### **3. Suport a la implementació i formació:**

- Promoure i facilitar les traduccions oficials dels estàndards i recursos d'accessibilitat.

- Proporcionar documentació que facilite la comprensió i aplicació pràctica de les normes.
- Elaborar recursos de conscienciació i educació, com els materials disponibles a WAI Resources.

#### 4. Investigació en accessibilitat:

- Impulsar la recerca internacional per establir una base sòlida en els estàndards d'accessibilitat.
- Realitzar tasques d'*horizon scanning* per anticipar necessitats futures.

### 3.1.2. WAI-ARIA

En el desenvolupament d'aplicacions web completes no sempre és suficient utilitzar únicament etiquetes HTML estàndard, ja que normalment les pàgines no són estàtiques. Per aquest motiu és necessari disposar de recursos addicionals que permeten als lectors de pantalla reconèixer i descriure els elements de manera correcta. D'aquesta necessitat sorgeix la iniciativa **WAI-ARIA**.

La **Accessible Rich Internet Applications (WAI-ARIA)** és una especificació del W3C que defineix mecanismes per fer que el contingut i les aplicacions web siguin més accessibles per a les persones amb discapacitats [33].

Aquesta iniciativa resulta especialment rellevant per al contingut dinàmic i per als controls avançats de la interfície d'usuari desenvolupats amb HTML, JavaScript i tecnologies relacionades. Es a dir, pràcticament per a tot allò que vaja més enllà d'una pàgina web amb capçaleres, paràgrafs i alguna imatge.

Sense WAI-ARIA, determinades funcionalitats dels llocs web no serien accessibles per a alguns col·lectius, com ara les persones que depenen de lectors de pantalla o aquelles que no poden utilitzar un ratolí, en aquest treball ens centrem sols amb aquelles que depenen dels lectors de pantalla.

Així, WAI-ARIA se centra en tot allò que va més enllà de l'HTML bàsic, establint mecanismes perquè les tecnologies d'assistència puguen interpretar correctament la interfície. Mitjançant l'ús de WAI-ARIA, els desenvolupadors poden garantir que les aplicacions web avançades siguin accessibles i usables per un major nombre de persones.

L'objectiu principal és assegurar que funcionalitats avançades, com ara menús desplegables, formularis i tots els seus elements, pestanyes, controls lliscants (*sliders*) o diàlegs, resulten comprensibles i operables per a persones amb discapacitat.

Per aconseguir-ho, l'ús d'atributs ARIA (com `role` o `aria-*`) permet proporcionar informació semàntica sobre el rol, l'estat i les propietats dels elements, ja siguin estàtics o dinàmics. Sense aquests atributs, el navegador no pot deduir per si mateix aquestes característiques, deixant part de la interfície invisible o inoperable per a la tecnologia d'assistència.

## Definició i propòsit de WAI-ARIA

WAI-ARIA és una recomanació oficial del W3C publicada al juny de 2023 [34]. Proporciona un conjunt de rols i atributs que poden aplicar-se als elements HTML per enriquir-ne la semàntica. Per exemple, permet identificar regions de la pàgina (com ara “navegació”, “capçalera” o “secció”), especificar tipus d’elements (botons, menús, caselles de selecció) i declarar estats dinàmics (ítems marcats, deshabilitats, expandits, etc.). Sense aquests atributs, moltes d’aquestes funcionalitats podrien resultar invisibles per a les tecnologies d’assistència si es feren servir exclusivament els recursos natius de l’HTML.

### 3.1.3. Us de Wai-Aria

Les directrius de Wai-Aria indiquen que hi ha que utilitzar-la només quan siga estrictament necessaria. Es a dir, si es pot utilitzar HTML de forma nativa amb atributs propis del HTML es millor utilitzar-los que utilitzar ARIA, aquesta es la primera regla d’ús d’Aria [35].

Per exemple, si s’ha dutilitzar un botó es millor utilitzar l’element <button> natiu de HTML en lloc de <div role="button"> que seria l’opció de ARIA. Utilitzar ARIA més del estrictament necessari pot crear problemes cap al lector de pantalla com missatges contradictoris rols superflus o directament erronis.

En paraules de la guia oficial del W3C, «cap ARIA és millor que un ARIA dolent»[36]. Per tant sols hi ha que utilitzar Aria quan no existisca altra alternativa nativamente en HTML.

Un exemple d’aquesta situació és l’atribut aria-live. no té cap equivalent natiu i s’utilitza per anunciar canvis en regions dinàmiques, amb valors com polite o assertive [37].

WAI-ARIA (W3C) proporciona rols, propietats i estats que ajuden els lectors de pantalla a comprendre components avançats.

Per tal de esclarir millor la utilitat de Aria anem a vore un us típic de les propietats de ARIA es per a cambiar el text que llig el lector de pantalla per defecte.

### 3.1.4. Millora de l’accessibilitat en aplicacions web complexes

El conjunt d’atributs i rols definits per WAI-ARIA resulta especialment útil quan es desenvolupen aplicacions web amb interaccions complexes que van més enllà de l’ús d’etiquetes HTML bàsiques.

Gràcies a ARIA, és possible dividir un lloc web en regions lògiques i en components (widgets) que poden ser interpretats correctament per les tecnologies assistives. D’aquesta manera, les persones amb discapacitat visual poden entendre l’estructura d’una pàgina i moure’s amb agilitat a través dels menús dels lectors de pantalla o utilitzant únicament el teclat.

Per exemple, ARIA permet definir «landmarks» com banner, main o navigation, la qual cosa facilita una navegació ràpida entre seccions [38].

Un altre àmbit on ARIA aporta un gran valor és en la creació de widgets personalitzats. Per exemple, un element definit com:

```
<ul role="menubar">
  <li role="menuitem">Opció 1</li>
  <li role="menuitem">Opció 2</li>
</ul>
```

Serà interpretat pels lectors de pantalla com un menú, encara que es tracte d'una llista no semàntica en HTML (La qual cosa es estructuralment incorrecta, però es un exemple util per a il·lustrar les possibilitats d'aria). De manera semblant, els atributs aria-live i aria-busy permeten declarar regions amb actualitzacions dinàmiques, de manera que els canvis siguin anunciats automàticament per la tecnologia d'assistència sense necessitat de recarregar la pàgina. Això és especialment útil en contextos com aplicacions de missatgeria, notícies en temps real o panells d'estat [38].

També és possible descriure controls personalitzats amb estats canviants. Per exemple:

```
<div role="button" aria-pressed="true">Reproduir</div>
```

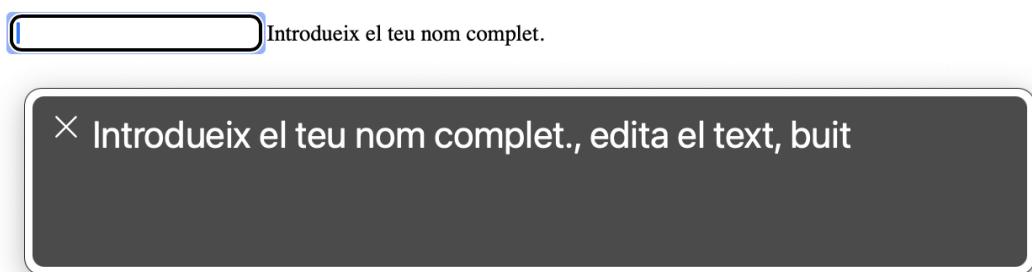
En aquest cas el botó 'Reproduir' serà tractat per VoiceOver com un botó i, a més, anunciarà el seu estat actual («activat» o «desactivat»).

Altres atributs com aria-checked, aria-expanded o aria-selected ofereixen informació clau per a entendre l'estat d'un component interactiu. Així, l'usuari rep un feedback constant i precís de les seues accions.

A més, l'atribut aria-describedby permet associar a un element un text descriptiu addicional. D'aquesta manera, el lector de pantalla llig primer el nom accessible (accessible name) i tot seguit la descripció vinculada, millorant la comprensió de camps de formulari o botons amb funcions no evidents. Per exemple:

```
<input type="text" aria-describedby="ajuda-nom">
<small id="ajuda-nom">Introduceix el teu nom complet.</small>
```

En aquest cas el quan es faça focus sobre el input type text es llig el accesible name de l'etiqueta . Com es veu en la figura 3.1



**Figura 3.1:** Aria-describedby exemple

Cal destacar que WAI-ARIA no substitueix l'ús d'HTML semàntic, sinó que l'amplia. Els encapçalaments, les llistes, els enllaços i l'atribut alt en les imatges continuen sent recursos natius essencials per a l'accessibilitat. Els rols i atributs d'ARIA només s'han d'aplicar quan la semàntica pròpia de l'HTML no és suficient, ja que un mal ús pot generar informació contradictòria o confusa per als usuaris de lectors de pantalla [35].

En aquest sentit, resulta fonamental recordar la màxima recollida en la guia APG del W3C: «*cap ARIA és millor que un ARIA dolent*» [36].

### 3.1.5. Principis bàsics abans d'utilitzar ARIA

Abans de recórrer a WAI-ARIA, és fonamental conéixer i aprofitar al màxim les capacitats pròpies d'HTML. Les directrius oficials del W3C estableixen que ARIA només s'ha d'utilitzar quan no hi ha cap alternativa nativa en HTML [35]. A continuació es presenten els tres principis bàsics que tot desenvolupador ha de seguir:

1. **Semàntica nativa primer.** Sempre s'han de prioritzar els elements i atributs HTML natius, ja que ofereixen millor suport per part dels navegadors, els lectors de pantalla i les eines d'indexació. Alguns exemples són:
  - <label> + for per a etiquetar correctament els camps de formulari.
  - Atributs com required, disabled, readonly, placeholder, pattern, o bé tipus d'entrada específics com type="email", type="tel", type="url", type="number", etc.
  - <fieldset> + <legend> per a agrupar camps relacionats.
  - <form> amb un títol visible, associat a un id, en lloc d'utilitzar directament aria-labelledby.

*Regla d'or:* “Cap ARIA és millor que un ARIA dolent” [36].

2. **Accessible Name i Description.** Cada control d'interfície ha de tindre un *nom accessible*, que és el text que les tecnologies d'assistència (com els lectors de pantalla) llegeixen per identificar-lo. L'ordre de prioritat és el següent:
  - aria-labelledby (*té la màxima prioritat*).
  - Si no existeix, aria-label.
  - En defecte, el contingut d'una etiqueta <label> associada o text intern visible.

Addicionalment, es poden proporcionar *descripcions accessibles* mitjançant aria-describedby, especialment útil per a instruccions contextuales o missatges d'error.

3. **Evitar redefinir rols natius.** No s'ha d'emprar l'atribut role per a redefinir un element HTML que té una semàntica pròpia. Per exemple, no és correcte utilitzar role="textbox" en un <input type="text>, ja que això pot provocar inconsistències i confusions en els lectors de pantalla.

### 3.1.6. Exemples d'us de WAI-ARIA

Si tenim un element que és un botó:

```
<button> Aquest es el text que es va a llegir </button>
```

En aquest cas el lector de pantalla llegirà les paraules "Aquest es el text que es va a llegir" com es veu en la figura 3.2

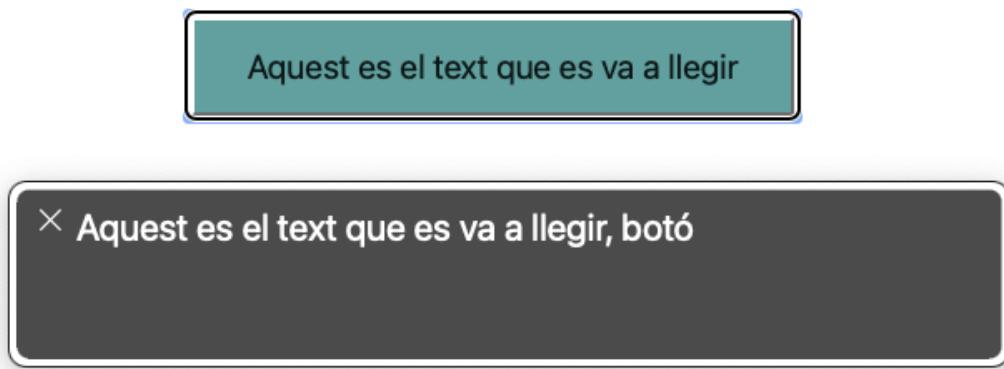


Figura 3.2: Lectura de botó 1

Allò que llig el lector de pantalla es coneix com a *accessible name* i per exemple en els botons és allò que apareix entre les etiquetes de button.

Però hi ha vegades que el que es mostra en el botó no és suficientment descriptiu per a entendre el que vol expressar el botó simplement llegint-ho, o què passaria si el botó fos un emoji o una imatge?

Per a solucionar aquest problema el que hem de fer és canviar el *accessible name* per defecte, és a dir, fer que el lector de pantalla llisca un text diferent al que es mostra de forma visual per pantalla.

Html no proporciona de forma nativa les eines per a fer-ho, per tant hem recurrir a atributs ARIA.

Es important tindre en compte que en el cas de les imatges, aquests tenen un atribut concret per a proporcionar accessibilitat el "alt". L'efecte a nivell d'usuari quan s'utilitza la propietat alt o aria-label és exactament el mateix:

Si escrivim fent us de alt:

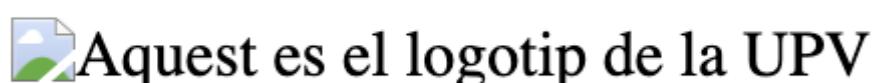
```

```



**Figura 3.3:** Imatge utilitzant alt

Quan no existeix la imatge i hem utilitzat la propietat alt es mostra el que es veu en la figura 3.4



**Figura 3.4:** Imatge utilitzant alt quan no existeix

Si escrivim fent ús de aria-label:

```

```



**Figura 3.5:** Imatge utilitzant aria-label

Quan no existeix la imatge i hem utilitzat la propietat alt es mostra el que es veu en la figura 3.6



**Figura 3.6:** Imatge utilitzant aria-label quan no existeix

Com es pot comprovar en les figures 3.3 i 3.5 la percepció per part de l'usuari evident és quasi igual, llevat del cas en el qual la figura no ha carregat correctament com es veu en les figures 3.4 i 3.6

Este és un clar exemple de perquè sempre que es puga s'ha de intentar no utilitzar les propietats d'Aria, ademés del comentat amb anterioritat els navegadors, motors de renderització i lectors de pantalla tenen suport total per a alt.

En canvi, aria-label en <img> no està tan àmpliament suportat, i pot ser ignorat en alguns contextos o navegadors antics. També es important tindre en compte el SEO si es vol que la pàgina es posicione correctament, i en aquest cas els motors de cerca si que tenen en compte els atributs natius de html, es a dir alt, i no els de aria-label.

Tenint l'anteriorment esmentat en compte, hi ha diverses formes de canviar els noms i aquest és l'ordre de prioritats en les quals ho llig el lector de pantalla: Ordre de prioritat dels noms accessibles

Els lectors de pantalla prioritzem així (de més a menys):

1. aria-labelledby
2. aria-label
3. <label> vinculat a <input>
4. Contingut visible textual

Aquest ordre es clau per tal de fer un bon ús de els atributs aria [39] d'aquesta forma el quan es desenvolupa una aplicació web podem canviar el que el lector de pantalla diu per a completar la informació.

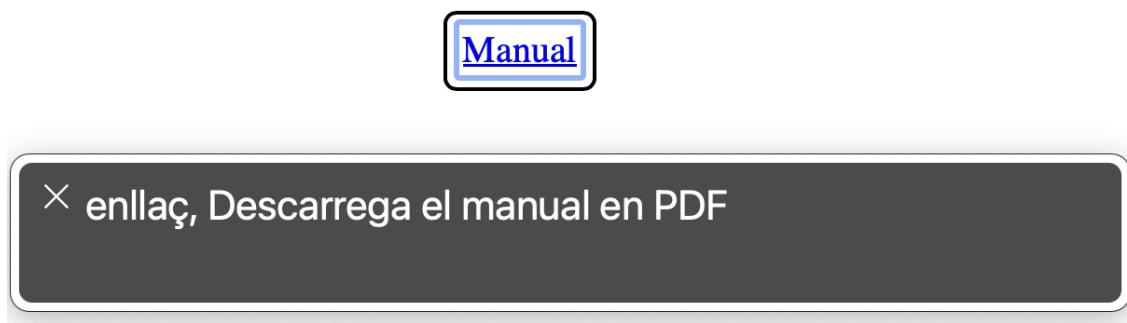
A continuació s'explica com s'utilitzaria cadascun d'aquests atributs:

### **aria-label**

Aria-label és l'atribut bàsic de aria, aquest ens permet canviar directament el nom que llegirà el lector de pantalla quan fem focus a un element.

```
<a href="pdf/manual.pdf" aria-label="Descarrega el manual en PDF"> Manual</a>
```

En aquest cas, si no hi hagués atribut aria-label, el lector llegiria manual, però com tenim aquest atribut, llig directament "Descarrega el manual en PDF". Com ens veu en la figura 3.7



**Figura 3.7:** aria-label

### **aria-labelledby**

L'atribut aria-labelledby assigna un nom accessible a un element mitjançant la referència a un altre element HTML que ja té un text visible. A diferència de aria-label (que escrius tu mateix el text), aria-labelledby apunta a un o més id d'elements HTML que ja contenen el contingut desitjat.

```
<h2 id="titol-formulari">Formulari de contacte</h2>

<form aria-labelledby="titol-formulari">
  <label for="email">Correu electrònic:</label>
  <input type="email" id="email" name="email" required>
</form>
```

En aquest cas utilitzem el accessible name natiu de `<h2>` per a el formulari, sense necessitat de tornar-lo a escriure i podent utilitzar-lo en altres parts del fitxer sense necessitat de reescriure codi, com si fos una variable. en la figura 3.8 es veu la informació que obtendria la persona no vident.

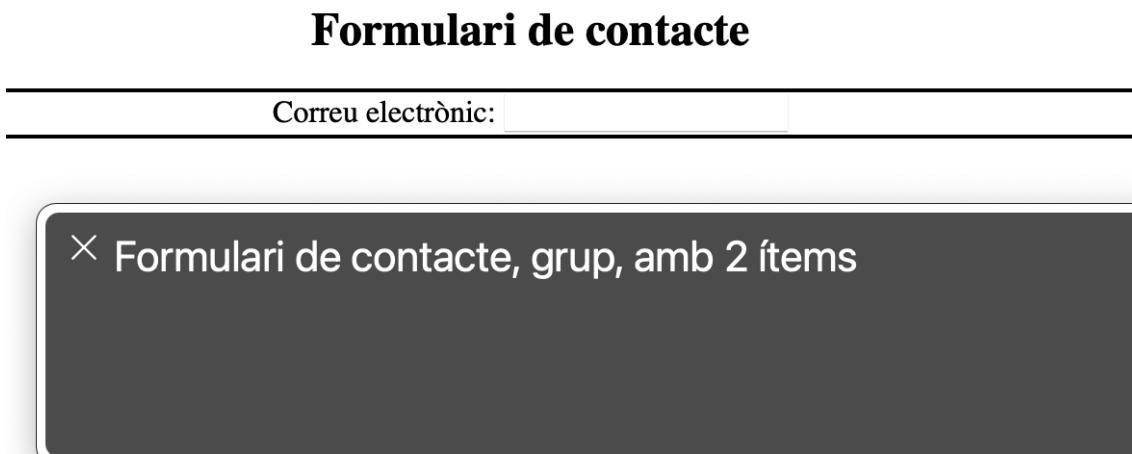


Figura 3.8: Formulari utilitzant aria-labelledby

### 3.1.7. Errors freqüents i com evitar-los

En el desenvolupament de formularis accessibles és habitual trobar certs errors que comprometen l'experiència d'usuari de les persones que utilitzen lectors de pantalla o altres tecnologies d'assistència. A continuació es descriuen els més comuns i la manera correcta d'evitar-los.

- Duplicació de noms accessibles.** Un error molt recurrent és combinar una etiqueta `<label>` amb un atribut `aria-label` que conté un text diferent. Això provoca que el lector de pantalla anuncie dues denominacions contradictòries per al mateix control.

#### Incorrecte:

```
<label for="nom">Nom</label>
<input id="nom" type="text" aria-label="Escriu el teu nom complet">
```

En aquest cas, un lector de pantalla podria llegir tant “Nom” com “Escriu el teu nom complet”, creant confusió.

**Correcte:**

```
<label for="nom">Escriu el teu nom complet</label>
<input id="nom" type="text">
```

La solució és utilitzar només un mecanisme i, sempre que siga possible, prioritzar les etiquetes HTML natives.

2. **Marcar camps requerits de manera incorrecta.** Encara hui és freqüent trobar formularis on els camps obligatoris només estan indicats amb color o amb JavaScript, fent-los inaccessible. La bona pràctica és emprar l'atribut natiu `required`, que és comprés i anunciat automàticament pels navegadors. En casos de components personalitzats, s'afegeix `aria-required="true"`.

**Incorrecte:**

```
<label for="email">Correu electrònic (obligatori)</label>
<input id="email" type="email">
<script>
    // Marca l'error només amb CSS
    document.getElementById('email').style.border = "2px solid red";
</script>
```

**Correcte:**

```
<label for="email">Correu electrònic</label>
<input id="email" type="email" required>
```

3. **Ocultar contingut de manera inadequada.** Hi ha diferència entre amagar un element només visualment i llevar-lo també de l'accessibilitat.

**Incorrecte:**

```
<p style="display:none;">Aquest text mai serà llegit pel lector</p>
```

**Correcte (ocultar només visualment, però accessible):**

```
<p class="visually-hidden">Instruccions només per
a usuaris de lector de pantalla</p>
```

S'oculta amb la classe de css: `visually-hidden` per tant es sols un canvi visual però no estructural.

4. **Interaccions no anunciades.** Quan es produeixen canvis dinàmics (p. ex., un missatge d'error després d'enviar el formulari), si no s'utilitzen mecanismes com aria-live, el lector de pantalla no informa a l'usuari.

**Incorrecte:**

```
<span id="error-msg">El camp és obligatori</span>
```

**Correcte:**

```
<span id="error-msg" aria-live="polite">El camp és obligatori</span>
```

Ara el lector de pantalla anunciarà el missatge en el moment en què aparega.

5. **Ús de rols contradictoris.** Assignar un role="presentation" a un element interactiu fa que perda la seua semàntica i siga invisible per a la tecnologia assistiva.

**Incorrecte:**

```
<button role="presentation">Enviar</button>
```

**Correcte:**

```
<button>Enviar</button>
```

Els rols "presentation" o "none" només han de col·locar-se en elements decoratius, mai en elements interactius.

## **3.2 Bones pràctiques d'accessibilitat web**

---

Les Directrius d'Accessibilitat per al Contingut Web (WCAG 2.1 i 2.2) [40, 41] estableixen un conjunt de criteris que garanteixen que les pàgines web siguin utilitzables pel màxim nombre de persones, incloses aquelles amb discapacitat visual, objecte d'aquest treball. A continuació expliquem els criteris més rellevants.

### **3.2.1. Etiquetatge semàntic i etiquetes descriptives (WCAG 2.4.6)**

L'ús correcte dels elements semàntics d'HTML és fonamental. Els encapçalaments <h1>–<h6> han de reflectir l'estructura jeràrquica del document i no utilitzar-se únicament amb finalitats estètiques. De la mateixa manera, els camps de formulari han d'estar associats amb un <label> descriptiu que explique clarament la seua funció.

```
<h1>Guia d'accessibilitat</h1>
<h2>Formulari de contacte</h2>
<label for="email">Correu electrònic</label>
<input id="email" type="email" required>
```

Un etiquetatge correcte permet que els lectors de pantalla anuncien el context i la finalitat de cada element, millorant la comprensió de la pàgina [35].

### 3.2.2. Navegació per teclat (WCAG 2.1.1)

Tota la funcionalitat interactiva ha de ser operable amb el teclat. Açò significa que enllaços, botons, menús desplegables i formularis han de ser accessibles sense ratolí. Els usuaris amb discapacitat visual o motriu sovint utilitzen la tecla *Tab* per desplaçar-se entre elements.

```
<button type="submit">Enviar</button>
<a href="ajuda.html">Ajuda</a>
```

Si s'utilitzen components personalitzats en JavaScript, cal garantir que responden a esdeveniments de teclat (*keydown*, *keyup*) i no sols a clics amb ratolí [40].

### 3.2.3. Contrast de color (WCAG 1.4.3)

El contrast entre el text i el fons ha de ser com a mínim de 4.5:1 per al nivell AA, i de 7:1 per al nivell AAA. Això assegura la llegibilitat per a persones amb baixa visió o daltonisme[42, 43].

Eines com WAVE, Lighthouse o contrast-checkers del W3C permeten validar de manera automàtica aquesta relació [44].

### 3.2.4. Indicador de focus visible (WCAG 2.4.7)

Quan un usuari navega amb el teclat, és imprescindible saber quin element té el focus en cada moment. Els navegadors mostren per defecte un contorn (*outline*), però molts dissenys l'eliminen per motius estètics, deixant la pàgina inusable amb teclat per a aquelles persones amb visió reduïda.

Aquest criteri és essencial per a garantir que les persones que naveguen sense ratolí puguen localitzar-se dins la interfície.

### 3.2.5. Text alternatiu per a contingut no textual (WCAG 1.1.1)

Qualsevol imatge, gràfic o element multimèdia ha de disposar d'una alternativa textual equivalent. Les imatges decoratives poden marcar-se amb un *alt=""*, mentre que les informatives han de tindre un text alternatiu que descriga el seu significat.

```
<!-- Imatge informativa -->

<!-- Imatge decorativa -->

```

Els vídeos han d'ofrir transcripcions i subtítols, i els gràfics han de comptar amb descripcions alternatives accessibles [41].

## 3.3 Lectors de pantalla

---

Un lector de pantalla és un programari d'assistència que transforma en veu —o en braille, quan es disposa del dispositiu corresponent— la informació que es mostra en pantalla. A partir de la interfície d'accessibilitat del sistema o del navegador, el lector construeix i consulta l'anomenat *arbre d'accessibilitat*, una representació estructurada del document que inclou rols, noms accessibles i propietats dels elements. D'aquesta manera, les persones cegues o amb baixa visió poden accedir al contingut i a la funcionalitat de la pàgina.

En el procés de lectura, el programari recorre el document HTML i interpreta els elements semàntics (encapçalaments, paràgrafs, llistes, taules, enllaços, botons, etc.) per generar una experiència auditiva coherent. Perquè aquesta experiència siga satisfactòria, resulta fonamental que el codi HTML estiga correctament estructurat i que, només quan siga imprescindible, s'empren atributs WAI-ARIA per a completar la semàntica [45].

### 3.3.1. Com els lectors de pantalla interpreten el DOM

Abans de continuar amb l'explicació de lectors de pantalla concrets és important saber quin es el **flux d'informació** que segueix una pàgina web fins a convertir-se en *nodes accessibles* consumibles per un lector de pantalla, i com aquesta informació es reproduceix en forma de so.

El navegador construeix el **DOM** a partir de l'HTML, deriva un **arbre d'accessibilitat** amb rols, noms i estats, l'exposa mitjançant l'API de plataforma (per exemple: *AXAPI* en macOS, *UIA* en Windows, *AT-SPI* en Linux), i el lector de pantalla en fa la presentació i la navegació a l'usuari.

#### Flux d'informació: del DOM a la síntesi de veu

1. **Marcació semàntica (HTML)**: El desenvolupador d'una aplicació utilitza etiquetes natives (`<button>`, `<nav>`, `<table>`) i, només si cal, atributs WAI-ARIA per aportar semàntica i estat.
2. **DOM → Arbre d'accessibilitat**: El navegador crea un arbre específic per a les AT (*assistive technologies*), derivat del DOM, que conté únicament la informació rellevant (rol, *accessible name*, estat, relacions).

3. **Exposició per API de plataforma:** Els nodes accessibles es publiquen per l'API pròpia del sistema (*AXAPI/UIA/AT-SPI*) amb propietats normalitzades segons els *Accessibility API Mappings* de W3C.
4. **Consum pel lector de pantalla:** el lector consulta l'API, construeix la seu vista estructural (enllaços, encapçalamens, formularis, regions), i anuncia el *nom accessible*, el rol i l'estat; l'usuari interactua amb teclat o gestos i rep resposta via veu o braille.

## L'arbre d'accessibilitat

L'**arbre d'accessibilitat** és una representació simplificada i semànticament significativa del DOM: pot *excloure* nodes sense valor semàntic i *incluir* elements que el DOM visual amaga darrere de components en. És, per tant, un subconjunt orientat a les tecnologies d'assistència. Entre els factors que influeixen en la seua construcció destaquen:

- La semàntica nativa d'HTML i els mapejats normalitzats (*HTML-AAM*) [?].
- El suport de rols i estats ARIA segons *Core-AAM*.
- L'efecte de CSS i atributs de presentació: `display:none` i `visibility:hidden` eliminan el node de l'arbre accessible; `opacity:0` no el lleva; `aria-hidden="true"` l'exclou de l'arbre sense amagar-lo visualment.

## Nom i descripció accessibles (*Accessible Name/Description*)

El **nom accessible** és la cadena que el lector anuncia com a etiqueta de l'element, i es calcula seguint un algoritme estandarditzat (*Accessible Name and Description Computation*). L'ordre de preferència inclou:

1. `aria-labelledby` (si referencia text vàlid).
2. `aria-label`.
3. Fonts natives de nom segons el tipus (p. ex., `<label>` associada a `<input>`, text interior de `<button>`, `alt` en `<img>`, `caption` en `<table>`).
4. `title` (quan procedisca).
5. Contingut textual residual segons regles de l'algoritme.

La **descripció accessible** (*accessible description*) amplia o contextualitza el nom, sovint via `aria-describedby` o contingut associat.

## Principis pràctics per al desenvolupament

De la cadena *DOM* → *Arbre accessible* → *API* → *Lector* se'n deriven bones pràctiques clau:

1. **Prioritza HTML nadiu:** usa etiquetes i atributs estàndard; utilitza ARIA només quan HTML no cobrisca el cas, i mai per contradir semàntica nativa.
2. **Garanteix un nom accessible** per a tot element interactiu (enllaços, botons, camps, *dialog*, etc.). Prefereix etiquetes visibles (`<label>`, `<button>`, `<legend>`, `<caption>`).
3. **Ves amb compte en amagar contingut:** `display:none/visibility:hidden` el trauen de l'arbre accessible; `aria-hidden="true"` l'amaga d'AT però no visualment; per a «*només per a lector*» empra patrons *visually hidden* provats.
4. **Comprova l'arbre accessible** amb les eines del navegador (pestanya *Accessibility* de DevTools) i valida amb WAVE i Lighthouse com s'explica en la secció corresponent de l'aplicació.

En aquest treball donat que el dispositiu amb el que es farà el desenvolupament de l'artefacte software i amb el que es fan les proves té el sistema operatiu de MacOS es centrarà l'explicació en el lector de pantalla propi del sistema que és VoiceOver.

### 3.3.2. Voice Over

VoiceOver obté la informació del contingut web a través de l'API d'accessibilitat del navegador, que li proporciona un arbre d'accessibilitat on cada element rep rols i noms accessibles derivats del seu contingut i atributs. La qualitat d'aquesta estructura és determinant per a l'accessibilitat efectiva de la pàgina.

A més de la locució, VoiceOver ofereix un suport visual que mostra en un recuadre gris el text que s'està llegint, la qual cosa és útil tant per a persones videntes que volen aprendre el funcionament del lector com per a documentar el comportament, aquesta opció es la que utilitzarem en el treball per entendre el funcionament del voiceover. El recuadre es mostra a la figura 3.9.

× VoiceOver activat Safari, Pagina ben feta, finestra, Pagina ben feta, contingut web, té el focus del teclat

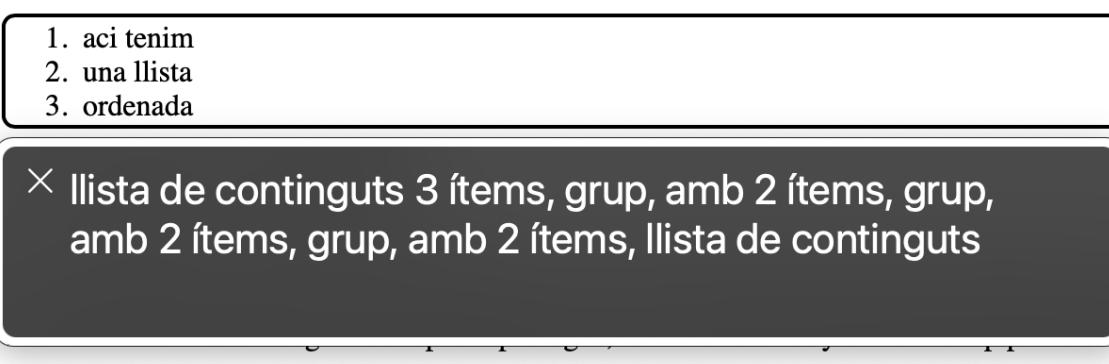
Figura 3.9: Recuadre VoiceOver

- Els encapçalaments HTML `<h1>–<h6>` reben automàticament el rol *heading*; VoiceOver els anuncia com “encapçalament nivell 1”, “nivell 2”, etc. [45]. Vegeu la figura 3.10.



**Figura 3.10:** voiceOver h1

- Els paràgrafs <p> es llegeixen com a text corrent.
- Les llistes <ul>/<ol> s'anuncien com a llistes, amb el nombre d'elements; vegeu la figura 3.11.

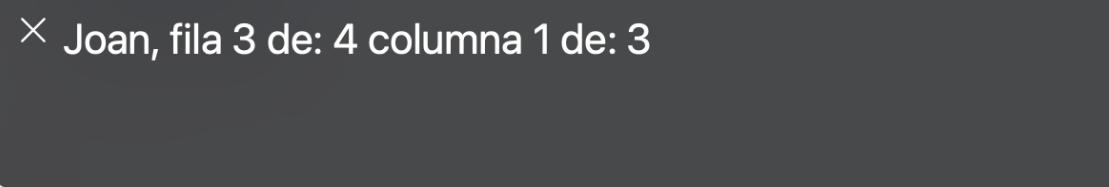


**Figura 3.11:** voiceOver ul

- Les taules, quan tenen capçaleres i cel·les etiquetades adequadament, es poden recórrer amb informació de fila i columna. Vegeu la figura 3.12

## Taula accessible

Resultats de l'enquesta		
Nom	Valoració	Comentaris
Anna	8	Bon servei
Joan	9	Molt professional
Marc	7	Podria millorar



**Figura 3.12:** Taula accessible vista amb VoiceOver

- Les etiquetes semàntiques <nav>, <header>, <main>, <aside>, <footer> defineixen *landmarks* de pàgina que VoiceOver incorpora al seu rotor, facilitant salts ràpids entre "navigation", "main", "complementary", etc.

El rotor de Landmarks de VoiceOver és una eina que permet a l'usuari accedir ràpidament a les diferents regions principals d'una pàgina web. Aquestes regions es defineixen mitjançant etiquetes semàntiques d'HTML5 (com `<header>`, `<main>`, `<nav>`, `<footer>`) o amb rols ARIA equivalents (`role="banner"`, `role="navigation"`, `role="main"`, etc.). Quan l'usuari activa el rotor i selecciona la categoria “Landmarks”, VoiceOver mostra una llista d'aquestes seccions, de manera que és possible saltar directament a la part desitjada de la pàgina sense haver de recórrer-la element per element.

### Què és l'*accessible name*

L'*accessible name* d'un element és el text que el lector de pantalla anuncia com a nom del control. Per defecte, en un botó:

```
<button>Accepta</button>
```

El nom accessible és el text “Accepta”. Si es necessita un nom diferent del contingut visible, cal recórrer a atributs ARIA, ja que HTML no proporciona un mecanisme natiu per substituir aquest comportament (vegeu les figures 3.2 i 3.7) [46].

De manera similar, tal com s'illustra en les figures 3.3 i 3.5, una imatge amb:

```

```

Fa que VoiceOver llija “Logotip de l'empresa”, perquè l'atribut `alt` actua com a nom accessible per defecte. En aquest cas, utilitzar `aria-label` és una mala pràctica, ja que HTML ja ofereix el mecanisme adequat; vegeu la figura 3.6 [46].

### Formularis

En formularis, el text d'una etiqueta `<label>` associada a un `<input>` proporciona el nom accessible del camp. Quan no hi ha text visible, s'ha de definir amb `aria-label` o `aria-labelledby` (referenciant un altre element amb text). Aquestes tècniques s'han mostrat a les figures 3.7 i 3.8.

Per exemple, si un botó només conté una icona, es pot proporcionar un nom accessible amb `aria-label="Obrir menú"`, de manera que VoiceOver anuncie “Obrir menú”. En aquest cas, l'ús d'ARIA és pertinent perquè no existeix cap text visible que puga actuar com a etiqueta. A la figura 3.13 es mostra un cas pràctic:

```
<button aria-label="Envia el formulari">(icona)</button>
```



- ✗ Enviar el formulari, botó

**Figura 3.13:** Botó en logotipo utilitzant aria-label

### 3.3.3. Ús de VoiceOver

VoiceOver utilitza combinacions de tecles per facilitar la navegació i la interacció amb els continguts web.

#### Activació o desactivació de VoiceOver

- **Drecera ràpida:** *Command + F5*.
- **Configuració:** Preferències del Sistema → Accessibilitat → VoiceOver → Activar.

#### Tecles bàsiques

- La “tecla VO” correspon a *Control + Option*.

Combinació de tecles	Funció
<b>VO + fletxes (→ o ←)</b>	Navega pels elements d’una pàgina (avançar/retrocedir)
<b>Ctrl + Option + barra espaiadora</b>	Activa o selecciona l’element actual
<b>Ctrl + Option + U (Rotor)</b>	Obri el rotor per seleccionar tipus d’elements
<b>Ctrl + Option + A</b>	Llig tot el contingut visible de la pàgina actual
<b>Control</b>	Pausa la veu

**Taula 3.1:** Combinacions de teclat per a VoiceOver

### Navegar per tipus d'elements (Rotor)

El rotor permet filtrar i recórrer ràpidament enllaços, botons, encapçalaments, formularis, paràgrafs, taules, *landmarks*, etc.

**Obrir el rotor:** VO + U.

**Una vegada obert:** fletxes ←/→ per canviar la categoria; fletxes ↑/↓ per recórrer els elements dins de la categoria.

### Llegir per seccions concretes

Combinació de tecles	Funció
Llegir el títol de la pàgina	VO + Shift + T
Navegar per encapçalaments (h1, h2...)	Rotor o VO + Command + H
Navegar per paràgrafs	VO + Command + P
Navegar per formularis	VO + Command + J
Navegar per taules	VO + Command + T

Taula 3.2: Llegir per seccions concretes

### Interactuar amb formularis

Per escriure o interactuar amb <input>, <select> o <textarea>:

1. Desplaçar-se fins al formulari amb VO + fletxes.
2. **Entrar al mode d'interacció:** VO + Shift + ↓.
3. Omplir el camp.
4. **Eixir del mode:** VO + Shift + ↑.

### Altres configuracions

- **Anunciar el tipus d'element:** Preferències → VoiceOver → Web → “Anunciar tipus d'element”. Aquesta opció fa que, a més del nom accessible, s'indique el tipus (botó, enllaç, encapçalament...), la qual cosa facilita la ubicació en la navegació i ajuda en la verificació del marcatge semàntic durant el desenvolupament.
- Emprar **pàgines ben estructurades semànticament** (<header>, <main>, <nav>, <p>, <h1>, etc.).
- Ajustar la **velocitat de lectura** i la **veu** segons preferències personals [47].

### 3.3.4. JAWS

JAWS (*Job Access With Speech*) és un lector de pantalla per a Windows desenvolupat per Freedom Scientific. Orientat a persones amb discapacitat visual, proporciona lectura i navegació per textos, enllaços, imatges (amb alternatives textuales), formularis, botons i altres elements interactius, amb un conjunt ampli de dreceres de teclat per recórrer i operar la interfície.

A continuació es mostren algunes combinacions de tecles habituals:

Combinació de tecles	Funció
<b>Insert + F7</b>	Mostra llista d'enllaços
<b>Insert + F6</b>	Mostra llista d'encapçalaments
<b>H o Shift + H</b>	Navega pels encapçalaments (avançar/retrocedir)
<b>F o Shift + F</b>	Navega pels camps de formularis (avançar/retrocedir)
<b>T o Shift + T</b>	Navega per taules (avançar/retrocedir)
<b>Ctrl</b>	Pausa la lectura actual
<b>Insert + fletxes</b>	Navega per caràcters, paraules o línies de text

**Taula 3.3:** Combinacions de tecles per navegar amb JAWS

---

# CAPÍTOL 4

## Anàlisi del problema

---

En aquest capítol es descriu el problema concret que es pretén resoldre: la falta d'eines senzilles i didàctiques que permeten als desenvolupadors aplicar bones pràctiques d'accessibilitat des del disseny inicial d'un lloc web.

Es presenten les necessitats detectades en el context actual, les limitacions de les solucions existents i les raons per les quals cal plantejar una eina alternativa que combine interactivitat, claredat i rigor tècnic.

---

### 4.1 Recursos i dificultats dels desenvolupadors en l'accessibilitat web

---

Actualment, els desenvolupadors web disposen d'un ampli nombre de normatives, guies i eines que tenen com a objectiu facilitar la creació de llocs web accessibles.

En l'àmbit internacional, les *Web Content Accessibility Guidelines* (WCAG), elaborades pel *World Wide Web Consortium* (W3C) a través de la *Web Accessibility Initiative* (WAI), constitueixen la referència principal.

De manera complementària, les especificacions *Accessible Rich Internet Applications* (ARIA) ofereixen atributs i rols que permeten enriquir semànticament els elements HTML.

A nivell legal, cal nombrar la *Section 508* de la *Rehabilitation Act* als Estats Units, així com la Directiva (UE) 2016/2102 i l'*European Accessibility Act* (2019/882), que estableixen obligacions concretes per al sector públic i privat, respectivament, en matèria d'accessibilitat digital.

A més, existeixen nombroses eines d'avaluació automàtica que permeten comprovar el grau de compliment d'aquests estàndards, com ara WAVE, AXE o Lighthouse. Aquestes eines resulten útils per identificar errors, però presenten limitacions: se centren en la detecció de problemes i no ofereixen una guia didàctica per a la seuva prevenció o correcció. Es a dir, son útils quan la web ja està creada però no tant mentre es crea.

Tot i la disponibilitat d'aquest conjunt de recursos, la realitat mostra que molts llocs web continuen sent poc accessibles.

Les causes principals són la complexitat i extensió de les guies oficials, la falta de formació específica en accessibilitat, la percepció que aquesta suposa un sobrecost en el desenvolupament i, sobretot, el fet que normalment es planteja com una etapa d'auditoria final i no com una part mes durant l'etapa desenvolupament.

Aquesta situació deixa clar la necessitat d'eines complementàries, de caràcter pràctic i didàctic, que faciliten l'aprenentatge i l'aplicació de bones pràctiques d'accessibilitat per part dels desenvolupadors. Es en aquest punt on es centra aquest treball.

## 4.2 Com utilitzen les persones no vidents els ordinadors i els lectors de pantalla

---

En l'actualitat, l'accés a la informació digital és pràcticament una necessitat quotidiana. Tanmateix, per a les persones no vidents aquesta realitat presenta una barrera d'entrada molt elevada. El fet de no poder veure la pantalla, i per tant no poder fer ús del ratolí com a perifèric principal d'interacció, dificulta enormement l'accés als recursos digitals.

Per superar aquest obstacle, les persones cegues utilitzen una combinació del teclat amb un lector de pantalla, un programari d'assistència que interpreta el contingut visual i el transforma en veu sintetitzada o en caràcters braille (quan es disposa del maquinari adequat). Entre els lectors de pantalla més coneguts trobem:

- NVDA (Windows, gratuït),
- JAWS (Windows, comercial),
- VoiceOver (macOS/iOS, integrat),
- TalkBack (Android, integrat).

És important remarcar que aquestes eines no lleixeixen la pàgina web com ho faria una persona vident, sinó que processen l'*estructura semàntica* del codi HTML. Això implica que els desenvolupadors han d'emprar correctament les etiquetes i atributs disponibles, de manera que el lector de pantalla puga interpretar-los amb sentit. En cas contrari, el resultat pot ser una navegació confusa, tediosa o fins i tot impossible.

Per exemple, si un menú de navegació no està ben definit semànticament, en lloc d'anunciar “Menú: apartat 1..., apartat 2...”, el lector podria limitar-se a llegir una sèrie de paraules inconnexes sense cap estructura.

A nivell tècnic, fer una pàgina accessible no és una tasca complexa en termes de programació. No cal recórrer a llenguatges nous, ni a frameworks específics, ni a tecnologies estranyes. En essència, es tracta de fer un ús correcte de les etiquetes HTML i, quan siga necessari, complementar-les amb atributs ARIA. La dificultat real radica en identificar quins atributs són adequats en cada cas, ja que la documentació disponible sol ser extensa, poc clara i difícil de consultar per a desenvolupadors que s'inicien en aquest camp.

Amb l'objectiu de guiar aquesta pràctica, es van crear les *Web Content Accessibility Guidelines* (WCAG), elaborades pel W3C. Aquest estàndard estableix principis i bones pràctiques que garanteixen que el contingut siga:

- **Perceptible**,
- **Operable**,
- **Comprehensible**,
- **Robust**.

Quan aquests criteris es compleixen, els lectors de pantalla poden navegar d'una manera lògica i ordenada.

Existeixen també eines com **Lighthouse**, **WAVE** o **axe DevTools**, que permeten analitzar el nivell d'accessibilitat d'una pàgina. Tot i això, cap d'aquestes solucions resol per si sola tots els problemes ni substitueix un desenvolupament correcte d'origen.

A més, la creació d'una pàgina inicialment accessible sol ser una tasca farragosa. Aquesta dificultat es deu, principalment, a la falta de recursos senzills i directes per als desenvolupadors. Quan aquests volen adaptar una web o crear-la des de zero, sovint es veuen obligats a revisar la documentació oficial de les WCAG, que resulta molt extensa i poc pràctica en el dia a dia del desenvolupament.

Aquest treball pretén donar resposta a aquesta problemàtica mitjançant el desenvolupament d'una eina senzilla i interactiva. L'objectiu és oferir fragments de codi amb els atributs HTML i ARIA necessaris perquè els desenvolupadors puguen aplicar-los directament mentre creen o adapten les seues aplicacions web, assegurant així un accés més just i inclusiu per a totes les persones.

## 4.3 Avaluació de l'accessibilitat web

---

Tot i que aquest treball se centra principalment en l'aprenentatge i la creació d'una aplicació web accessible des de zero, també és fonamental conèixer quines opcions tenim quan ja disposem d'una aplicació i volem comprovar en quin grau és accessible i quins aspectes poden ser millorats. O com comprovar el grau d'accessibilitat que hem aconseguit en la nostra aplicació després d'utilitzar l'eina d'aquest treball.

### 4.3.1. WAVE

WAVE és una de les eines més conegudes per a l'avaluació de l'accessibilitat. Permet detectar errors i advertències directament sobre la pàgina web, mostrant icones i informació contextual. Tot i això, la interpretació dels resultats requereix un cert nivell de coneixement tècnic. A més, la seua execució implica carregar la pàgina en un entorn extern o bé utilitzar una extensió, la qual cosa pot alentir el procés en entorns de desenvolupament ràpid.

### 4.3.2. Lighthouse

Integrada en Google Chrome, Lighthouse permet avaluar aspectes clau com l'accessibilitat, el rendiment, el SEO i bones pràctiques de desenvolupament. El seu principal avantatge és la gratuïtat i la facilitat d'accés, però la informació que ofereix és generalista i no cobreix de manera exhaustiva situacions específiques, com l'experiència de persones amb ceguesa total.

### 4.3.3. Exemple d'ús de Google Lighthouse

Lighthouse és una eina de codi obert desenvolupada per Google que s'integra directament en les Developer Tools del navegador Chrome. Permet generar informes d'auditories automàtiques sobre diversos aspectes d'un lloc web, incloent-hi accessibilitat, rendiment, SEO, bones pràctiques i PWA (Progressive Web Apps).

**Com funciona** Per realitzar una auditoria d'accessibilitat amb Lighthouse, cal utilitzar el navegador Chrome i obrir la pestanya d'inspecció (F12):

1. A la barra superior del panell, seleccionar la pestanya "Lighthouse".
2. Marcar la casella d'"Accessibilitat" i prémer "Generate report".

Lighthouse executa una sèrie de tests automàtics mitjançant la llibreria axe-core (la mateixa que fa servir l'eina axe de Deque Systems) i genera un informe interactiu amb una puntuació sobre 100 punts.

Podem veure un exemple en les figures 4.1 i 4.2, on s'aprecia la diferència de puntuació entre una web professional i la web del projecte.

The figure shows a screenshot of the Google Lighthouse audit interface for the website [elpais.com](https://elpais.com). The audit score is 86. The interface includes a navigation bar with developer tools like Elements, Console, Fuentes, Red, Rendimiento, and Lighthouse. The Lighthouse tab is selected. Below the navigation is a timestamp (17:06:34 - elpais.com). The main content area displays audit results under the heading "Algunos problemas han afectado a la ejecución de Lighthouse:" (Some problems have affected the execution of Lighthouse:).

- Puede que algunos datos almacenados hayan afectado al rendimiento de carga en esta ubicación: IndexedDB. Haz una auditoría de esta página en una ventana de Incógnito para evitar que esos recursos afecten a tus puntuaciones.

Below this, there is a section titled "Accesibilidad" (Accessibility) with a sub-section "NOMBRES Y ETIQUETAS" (Names and Labels). It highlights a problem: "Los botones no tienen nombres accesibles" (Buttons do not have accessible names). It provides a recommendation: "Estas comprobaciones permiten identificar oportunidades para mejorar la accesibilidad de tu aplicación web. La detección automática solo puede detectar un subconjunto de problemas y no garantiza la accesibilidad de tu aplicación web, por lo que también te recomendamos que hagas pruebas manuales." (These checks help identify opportunities to improve your application's accessibility. Automatic detection only finds a subset of problems and does not guarantee the accessibility of your application, so we also recommend manual testing.)

At the bottom, there is a note: "A continuación se indican consejos para mejorar la semántica de los controles de tu aplicación. Estos consejos pueden" (The following are recommendations to improve the semantic controls of your application. These recommendations can).

Figura 4.1: Google Lighthouse en el Pais.com

**Figura 4.2:** Web del projecte analitzada amb Lighthouse

Com s'observa a la figura 4.1, aquesta eina proporciona:

- Una puntuació global sobre 100.
- Llistat d'elements amb problemes: per exemple, imatges sense alt, botons sense nom accessible, falta de contrast,ús incorrecte d'encapçalaments, etc.
- Suggeriments i bones pràctiques: cada problema inclou una descripció i un enllaç a la documentació oficial per entendre'l i solucionar-lo [48].

Tot i ser una bona eina per detectar omissions puntuals, Lighthouse té limitacions importants:

- Només detecta errors automàtics: no pot validar criteris subjectius com l'ordre lògic de lectura, la claredat del contingut o la comprensió del llenguatge.
- No substitueix proves manuals: és necessari complementar-lo amb tests reals amb lectors de pantalla i navegació per teclat.

#### 4.3.4. Exemple d'ús de WAVE (Web Accessibility Evaluation Tool)

WAVE és una eina desenvolupada per WebAIM (Web Accessibility In Mind) que permet avaluar visualment l'accessibilitat d'una pàgina web. A diferència de Lighthouse, que genera un informe estructurat, WAVE superposa icones i anotacions directament sobre la pàgina analitzada per indicar errors, alertes i característiques d'accessibilitat.

Es pot utilitzar des del lloc web oficial o afegint una extensió al navegador (Chrome o Firefox). Aquesta última és la més pràctica en entorns de desenvolupament.

Una vegada analitzada la pàgina, WAVE mostra:

- **Errors** (en roig): per exemple, imatges sense text alternatiu, formularis sense etiquetes, elements sense contrast adequat.
- **Alertes** (en groc): problemes potencials com textos molt petits, estructures poc clares oús incorrecte de llistes.
- **Característiques** (en verd): bones pràctiques implementades correctament.
- **Informació estructural**: encapsalaments jeràrquics, regions (<main>, <nav>, etc.), enllaços i llistes.



**Figura 4.3:** Web del projecte analitzada amb WAVE

## 4.4 Requisits de l'aplicació

L'eina es definix a partir de les barreres identificades i del marc WCAG 2.1–2.2/WAI. Es prioritza un frontend estàtic, didàctic i ràpid.

### 4.4.1. Requisits funcionals

- **RF1.** Visualitzar bones pràctiques basades en WCAG/WAI.
- **RF2.** Organització per temes (encapsalaments, imatges, taules, formularis, navegació, multimèdia, ARIA) amb estructura homogènia.
- **RF3.** Exemples de codi accessibles (HTML i ARIA només quan cal).

- **RF4.** Botó per copiar el codi en un clic, operable amb teclat.
- **RF5.** Ajuda contextual amb <details>/<summary> o diàleg accessible.
- **RF6.** Guia breu per validar amb WAVE i Lighthouse.
- **RF7.** Enllaços directes a eines i documentació oficial.
- **RF8.** Patrons WAI-ARIA només quan l'HTML no és suficient (*native first*).
- **RF9.** Operable íntegrament amb teclat i amb focus visible.
- **RF10.** Noms accessibles i descripcions en controls i formularis.
- **RF11.** (*Futur*) Cerca/filtre per etiqueta o criteri WCAG.

#### 4.4.2. Requisits no funcionals

- **RNF1.** Usabilitat i coherència visual (patrons repetibles).
- **RNF2.** Estructura clara: jerarquia d'encapçalaments i menú lateral.
- **RNF3.** Rendiment: càrrega < 2 s; Lighthouse (Performance)  $\geq 90$ .
- **RNF4.** Accessibilitat WCAG 2.2 AA (contrast, reflow, teclat, focus, errors).
- **RNF5.** Disseny responsiu sense pèrdua d'informació.
- **RNF6.** Compatibilitat amb lectors de pantalla (VoiceOver/NVDA).
- **RNF7.** Mantenibilitat: HTML semàntic, CSS modular, JS separat.
- **RNF8.** Escalabilitat de contingut sense canvis globals.
- **RNF9.** Dependències mínimes: funcionament en local/hosting estàtic.
- **RNF10.** Privacitat: sense recollida de dades; demos sense enviament real.
- **RNF11.** (*Futur*) Internacionalització (ES/EN).
- **RNF12.** SEO bàsic (<title>, <meta description>); Lighthouse (SEO)  $\geq 90$ .

#### 4.4.3. Traçabilitat (resum)

- Barreres detectades → RF1, RF3, RF8–RF10; RNF4, RNF6.
- Objectiu docent → RF2–RF7; RNF1–RNF2, RNF7–RNF8.
- Arquitectura estàtica → RNF3, RNF7, RNF9.
- Validació amb eines → RF6–RF7; RNF3–RNF4, RNF12.

---

---

# CAPÍTOL 5

## Disseny de la solució i Implementació

---

En aquest capítol es presenta la solució desenvolupada en el marc d'aquest treball de fi de grau. L'objectiu principal ha sigut crear una eina web senzilla, interactiva i didàctica que ajude als desenvolupadors a entendre i aplicar les bones pràctiques d'accessibilitat en el disseny de pàgines web.

A diferència d'altres eines d'avaluació ja existents, la proposta d'aquesta aplicació no es limita a detectar errors de manera automàtica, sinó que ofereix exemples concrets d'etiquetes HTML amb els seus atributs corresponents, explicacions detallades del perquè de cada decisió i bones pràctiques associades. L'objectiu és facilitar l'aprenentatge i proporcionar un recurs pràctic que es puga utilitzar durant el procés de desenvolupament, i no només al final com a etapa d'auditoria.

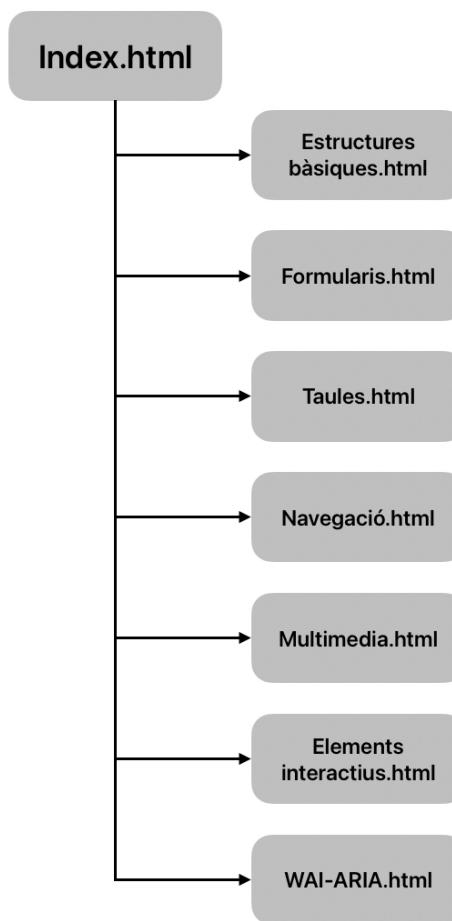
Per aconseguir-ho, l'eina integra una col·lecció d'elements HTML representatius (com encapçalaments, imatges, taules o formularis) acompanyats de les seues respectives recomanacions d'accessibilitat. A més, ofereix la possibilitat de visualitzar i copiar directament el codi font, la qual cosa redueix el temps d'implementació i minimitza el risc d'errors.

---

### 5.1 Funcionament de l'aplicació i estructura

---

L'estructura de pàgines de l'aplicació és molt senzilla i s'ha dissenyat amb l'objectiu de garantir claredat i rapidesa en la navegació. La jerarquia es pot observar en el diagrama de navegació de la figura 5.1.



**Figura 5.1:** Diagrama de navegació

Aquesta simplicitat ha sigut clau en el procés de desenvolupament, ja que l'objectiu no és crear una aplicació sobrecarregada amb animacions, transicions visuals ni un complexe sistema de pàgines. Sinó un recurs pràctic, funcional i ràpid per al públic al qual va dirigida: els desenvolupadors web.

L'aplicació s'ha concebut pensant en dos perfils d'usuari principals.

El primer correspon a aquells que s'endinsen per primera vegada en el món de l'accessibilitat web i que, per tant, necessiten una introducció bàsica. Per a ells, la pàgina d'inici ofereix una explicació introductòria i un marc conceptual general sobre accessibilitat.

El segon perfil són els usuaris que ja estan familiaritzats amb aquestes pràctiques o que han utilitzat l'aplicació prèviament. En aquest cas, es prioritza l'agilitat i per això es proporcionen funcionalitats com el botó de còpia ràpida del codi, que permet inserir immediatament fragments en el seu projecte.

En navegar per les diferents seccions, l'usuari pot trobar exemples de formularis, menús, taules, botons i altres elements habituals en el desenvolupament web. Aquests s'han agrupat per tipus comú per tal de facilitar trobar-los ràpid

al usuari. Cada element està acompañat d'un bloc desplegable que mostra el codi base accessible. Aquest codi està pensat per a ser reutilitzat i adaptat: el desenvolupador pot modificar el contingut visual segons les seues necessitats, però ha de mantindre l'estructura i els atributs proposats, que són els que garanteixen l'accessibilitat.

D'aquesta manera, l'aplicació no busca automatitzar el procés d'accessibilitat, sinó formar i guiar el desenvolupador, proporcionant-li tant els fragments de codi com les explicacions necessàries per entendre el perquè de cada atribut i cada decisió. A més, cada element incorpora un botó d'informació que desplega una explicació detallada dels atributs HTML i ARIA utilitzats, permetent comprendre com i per què contribueixen a fer l'etiqueta accessible.

En resum, la filosofia de l'eina combina simplicitat i rigor: d'una banda, redueix la complexitat tècnica oferint recursos immediats i reutilitzables; de l'altra, fomenta l'aprenentatge progressiu perquè els desenvolupadors incorporen bones pràctiques d'accessibilitat en el seu flux de treball habitual.

## 5.2 Vista de l'aplicació

### Pàgina de inici

En la figura 5.3 es mostra la pagina de benvinguda de l'aplicació, en aquesta s'explica el funcionament de la pròpia aplicació, quin es el seu objectiu i una introducció als conceptes generals de l'accessibilitat.

The screenshot shows the main interface of the application. On the left, there's a sidebar with a blue header containing the word 'Inici'. Below it is a list of categories: Estructures bàsiques, Formularis, Taules, Navegació, Multimèdia, Elements interactius, WAI-ARIA, and Ferramentes de validació. The main content area has a white header with the title 'Guia interactiva d'elements HTML accessibles'. Below the header is a large text box containing the following text:

Aquesta aplicació web està pensada com una eina d'ajuda per a desenvolupadors i desenvolupadores que volen fer les seues aplicacions web més accessibles per a persones amb discapacitat visual.

Aplicar criteris d'accessibilitat si no es té experiència pot resultar un procés confús i complex. Aquesta guia pretén simplificar aquesta tasca, oferint exemples pràctics i interaccions de tots els elements HTML, explicant com adaptar-los perquè siguin accessibles.

En navegar per les diferents seccions de la web, trobaràs elements com formularis, menús, taules, botons i altres, cadascun d'ells mostrat amb el seu codi corresponent ja preparat per ser accessible. Passant el ratolí per damunt de cada exemple, apareixerà un recuadre (tooltip) amb el codi HTML que pots copiar i adaptar al teu projecte. La idea és modificar el contingut de cada etiqueta però sense tocar la seua estructura ni els seus atributs, que són els que fan accessible la etiqueta.

Aquesta guia no automàtica l'accessibilitat, sinó que et dona les eines i coneixements necessaris per a implementar-la correctament, entenent què cal fer i per què.

**Recorda:** una web accessible no sols ajuda a les persones amb discapacitat visual, sinó que millora l'experiència d'usuari per a tothom.

Below this text box is another section titled 'Conceptes generals de l'accessibilitat' with the following content:

Fer una aplicació web accessible no depèn només del disseny visual, sinó també de l'estructura del codi. La base fonamental és utilitzar HTML semàntic, ja que aquest permet als navegadors i als lectors de pantalla interpretar correctament la jerarquia i la funció dels diferents elements.

Per exemple, etiquetes com <header>, <main>, <nav>, <section> o <footer> defineixen àrees clares dins d'una pàgina, mentre que <h1> fins a <h6> estableixen els diferents nivells d'encapçalaments.

Es a dir tot i que es pot fer un botó amb etiquetes com div, a o span i una combinació de css i JavaScript i a experiència d'usuari seria igual, una persona no vident que utilitz un lector de pantalla no podria reconeixer correctament que està davant d'un botó.

Per tant les bases serien les següents:

- Us correcte de l'HTML semàntic (<header>, <main>, <nav>, <section> o <footer>)

Figura 5.2: Pàgina de inici

En la part final de la pagina inicial estàn els enllaços cap a la documentació

The screenshot shows a web application's homepage. On the left, a sidebar menu lists various topics: Inici, Estructures bàsiques, Formularis, Taules, Navegació, Multimèdia, Elements interactius, WAI-ARIA, and Ferramentes de validació. The 'Inici' item is highlighted with a blue background. The main content area contains several sections:

- Component d'integració avançada**: A section about WAI-ARIA attributes, mentioning aria-label, aria-labelledby, aria-hidden, and aria-role.
- Bibliografia**: Links to the W3C Web and examples from dequeuniversity.
- Footer**: Copyright information: © 2025 Nom de la teua empresa.

Figura 5.3: Pàgina de inici

## Estructures bàsiques

En la figura 5.4 es veu com es distribueixen els diferents elements HTML al llarg de les diverses pàgines de l'aplicació

The screenshot shows a web application's form creation page. On the left, a sidebar menu lists the same topics as Figure 5.3, with 'Formularis i interacció' highlighted. The main content area contains three sections:

- Creació del formulari**: A section with a link to the initial code of the form.
- Introducció de text bàsica: <input> amb errors accessibles**: A form with fields for Name, Surname, and Email. It includes a note: "No compartirem el teu correu amb ningú." and a link to the accessible input code.
- Exemples de tipus d'<input> avançats**: A section with examples of advanced input types: Date of birth (22/08/2025), Appointment time (12:30), and Quantity.

Figura 5.4: Element del formulari

### Desplegable i botó de copiar

Com es veu en les imatges 5.5 i 5.6 cada element de l'aplicació conté un desplegable, aquest quan està obert conté el codi html per tal que el desenvolupador puga copiar-lo directament s'ha desenvolupat un botó anomenat copia codi, que el que fa es copiar solament el codi que es veu en el desplegable amb el format optim per a inserir-lo directament en la seua aplicació.



Figura 5.5: Desplegable tancat



Figura 5.6: Desplegable obert

### Botó de informació

El botó de informació es situa a la dreta del títol de cada element, aquest obri un quadre informatiu on s'expliquen les bones pràctiques d'accessibilitat associades a l'etiqueta. Com es veu en la figura 5.7

The screenshot shows a dark-themed web interface. On the left is a sidebar with navigation links: Inici, Estructures bàsiques, Formularis, **Taules**, Navegació, Multimèdia, Elements interactius, WAI-ARIA, and Ferramentes de validació. The main content area has two sections: 'Taula accessible' and 'Footer'. A tooltip is overlaid on the 'Footer' section. The tooltip title is 'Informació d'accessibilitat'. The content explains the importance of using the <caption> tag for tables and the <thead> and <tbody> tags for semantic separation. It also mentions the <aria-describedby> attribute for providing context to screen readers. A 'Tancar' button is at the bottom of the tooltip.

**Figura 5.7:** Botó de informació

## Explicació i exemples d'ARIA

Donat que els atributs i conceptes d'ARIA son fonamentals per al correcte desenvolupament d'aplicacions accessibles s'ha afegit una pàgina on es donen exemples concrets dels atributs d'aria i s'expliquen detalladament com es veu en la figura

The screenshot shows a sidebar with the same navigation as Figure 5.7. The main content area contains three examples of WAI-ARIA attributes:

- Què és WAI-ARIA i quan convé usar-la**: A general introduction to WAI-ARIA, stating it's a set of roles, states, and properties that complement HTML to communicate better with assistive technologies. It mentions the 'primer, HTML natiu' norma d'or.
- aria-label — Nom accessible explícit**: Explains how to provide an accessible name for controls that don't have visible text. It includes a code example for a button:
 

```
<button type="button" aria-label="Obrir menú principal">
        <span aria-hidden="true">≡</span>
      </button>
```
- aria-labelledby — Nom referenciat a text visible**: Shows how to reference a visible text label for an element. It includes a code example for a menu item:
 

```
Menú principal
```
- aria-describedby — Descripció/ajuda complementària**: Explains how to provide additional descriptive text. It includes a code example for an input field:
 

```
Correu electrònic: No compartirem mai el teu correu.
```

**Figura 5.8:** Pàgina d'exemples WAI-ARIA

## Eines de validació

Com s'ha comentat en el punt anterior hi ha diverses eines per tal de avaluar l'accessibilitat de l'aplicació com que aquestes també formen part de el flux de treball del desenvolupament, com es veu en la figura 5.9 s'ha afegit una pàgina on s'explica el funcionament d'aquestes i es recomana el seu ús.

Aquesta secció et permet conèixer i utilitzar ferramentes de validació per comprovar si la teua pàgina web és accessible. Les eines WAVE i Lighthouse analitzen el codi i la interfície, detecten possibles errors i ofereixen recomanacions per millorar-ne l'accessibilitat.

**WAVE**

WAVE és una eina gratuïta de WebAIM que permet analitzar qualsevol pàgina web i detectar problemes d'accessibilitat. Funciona com una extensió del navegador (Chrome i Firefox) o bé mitjançant el seu servei web.

Quan s'excuta, WAVE mostra la pàgina analitzada amb una sèrie d'ícones i marques visuals que indiquen:

- **Errors:** problemes greus d'accessibilitat (per exemple, imatges sense text alternatiu o encapçalaments mal estructurats).
- **Avisos:** possibles millors recomanades, que no sempre són errors.
- **Característiques:** elements correctes que contribueixen a l'accessibilitat (per exemple, presència d'etiquetes de formulari associades).

L'avantage de WAVE és que mostra els errors directament sobre la mateixa pàgina, facilitant-ne la identificació i correcció. És ideal per a validacions ràpides i concretes.

[Enllaç a l'estensió de WAVE](#)

**Lighthouse**

Lighthouse és una eina de Google integrada en el navegador Chrome DevTools, que permet realitzar una anàlisi completa d'una pàgina web. Genera un informe detallat que avaluва quatre dimensions principals:

- Performance
- Accessibility
- Best practices
- SEO

Per a l'accessibilitat, Lighthouse comprova automàticament aspectes com: contrast de colors, ús correcte d'etiquetes, estructura d'encapçalaments, presència de noms accessibles en botons i enllaços, entre altres criteris. L'informe ofereix una puntuació de 0 a 100, acompanyada d'una llista d'errors i suggeriments per millorar la pàgina. És especialment útil per a una evaluació global i per a comprovar l'eficiència del lloc en diferents dimensions.

[Enllaç a l'estensió de Lighthouse](#)

Figura 5.9: Pàgina de les eines de validació

## 5.3 Decisions de disseny

Durant el procés de desenvolupament de l'aplicació, he considerat fonamental prendre decisions de disseny que no només respongueren a criteris estètics, sinó també a requeriments d'accessibilitat, usabilitat i coherència tècnica. Aquestes decisions han estat guiades pels principis de les WCAG 2.1, així com per la meua experiència prèvia en desenvolupament web i els resultats preliminars de proves amb lectors de pantalla.

### 5.3.1. Tipografia i llegibilitat

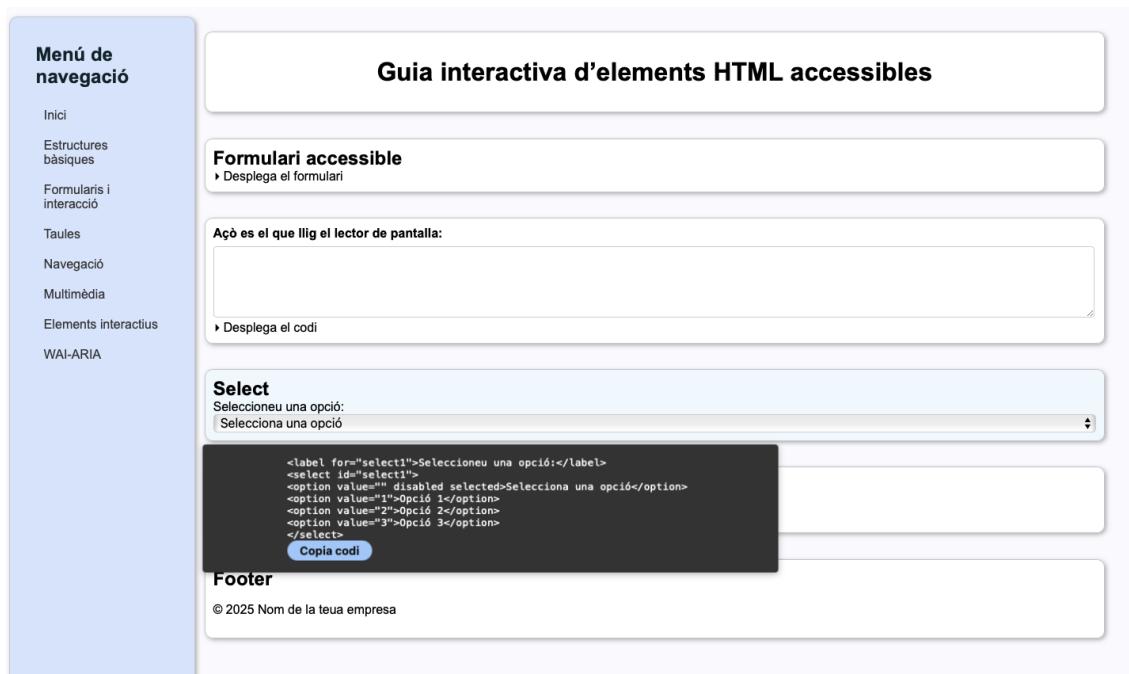
La selecció tipogràfica s'ha orientat a garantir una lectura clara i còmoda, especialment per a persones amb baixa visió. S'ha optat per una font sense serifa d'alta llegibilitat i amb un bon rendiment en diferents resolucions i pantalles. S'ha evitat l'ús de tipografies decoratives que, tot i ser atractives visualment, poden dificultar la lectura en contextos d'accessibilitat.

### 5.3.2. Estructura i jerarquia visual

L'organització del contingut segueix una jerarquia clara amb encapçalaments definits (h1, h2, h3) que no només milloren l'experiència visual, sinó que també permeten a les tecnologies d'assistència, com els lectors de pantalla, interpretar correctament l'ordre i rellevància de la informació. Respectant aquelles bones pràctiques d'accessibilitat que la pròpia aplicació mostra. Es a dir, es una aplicació totalment accessible que explica com fer aplicacions totalment accessibles.

### 5.3.3. Tooltip inicial i canvi a menú desplegable

Inicialment per a mostrar el codi de cada element es va optar per una opció flotant, anomenada tooltip en css, en la qual cada vegada que es pasava el cursor per damunt de un element apareixia el seu codi i el botó pertinent de copia de codi com es veu en la figura 5.10



**Figura 5.10:** Codi amb tooltip

Aquesta opció va ser descartada per l'actual mostrada en la figura 5.6 ja que generava molt de soroll visual alhora de navegar per la interficie fent degradant molt així l'experiència d'usuari.

### 5.3.4. Consistència i reconeixement

S'ha buscado mantenir un estil visual uniforme en tota l'aplicació, amb patrons de disseny coherents que faciliten el reconeixement dels elements. Prioritzant al màxim la velocitat alhora de buscar un element i copiar el seu codi per als usuaris experimentats amb l'aplicació. Per exemple posant el botó de copiar codi el mes prop possible del botó de desplegar per tal que es tinga que moure el ratolí el menys possible

## 5.4 Metodologia de desenvolupament

---

El desenvolupament de l'aplicació s'ha dut a terme seguint una metodologia iterativa i incremental, amb l'objectiu de permetre una avaluació i millora constants durant totes les fases del projecte. Aquesta estratègia ha garantit que cada component implementat responguera a requisits específics d'accessibilitat i que les decisions tècniques pogueren validar-se amb proves reals.

### 5.4.1. Fases del procés

El procés de desenvolupament s'ha estructurat en les següents fases principals:

1. **Anàlisi de requisits:** Recollida d'informació sobre necessitats d'accessibilitat per a persones amb ceguesa total, revisió de normatives WCAG 2.1 i estudi de bones pràctiques existents.
2. **Disseny conceptual:** Elaboració d'esbossos inicials de la interfície i definició de l'estructura HTML necessària per incloure la major varietat possible d'elements web.
3. **Implementació inicial:** Creació de la maqueta funcional amb HTML i CSS, prioritzant la semàntica i el compliment dels criteris d'accessibilitat.
4. **Proves preliminars:** Verificació de la compatibilitat amb diversos lectors de pantalla (NVDA, VoiceOver) i ajust de l'estructura segons els resultats.
5. **Integració de funcionalitats:** Afegit del botó d'informació, seccions desplegables amb `<details>` i botó de còpia de codi.
6. **Optimització i proves finals:** Ajust del disseny visual, comprovació de contrastos i proves d'ús amb usuaris reals i simulats.

### 5.4.2. Tecnologies utilitzades

L'aplicació s'ha desenvolupat amb HTML5, CSS3 i JavaScript pur, evitant dependències externes innecessàries per garantir la lleugeresa i compatibilitat màxima. Les proves d'accessibilitat s'han realitzat amb eines com WAVE, axe DevTools i Lighthouse, així com mitjançant la interacció directa amb lectors de pantalla.

## 5.5 Tecnologies emprades

---

En el desenvolupament d'aquest treball s'han emprat diverses tecnologies i eines que han facilitat tant la construcció de l'aplicació web com l'elaboració de la memòria. A continuació es descriuen les més rellevants i la seua funció dins del projecte.

### 5.5.1. HTML5

El llenguatge de marques **HTML5** ha sigut la base fonamental de l'aplicació. S'ha utilitzat per a definir l'estructura semàntica de les pàgines i implementar els diferents elements (encapçalaments, taules, formularis, multimèdia, etc.). L'èmfasi ha estat en l'ús correcte de l'HTML semàntic com a pilar de l'accessibilitat, afegint atributs i rols ARIA només quan ha sigut necessari.

### 5.5.2. CSS3

Els fulls d'estil en cascada **CSS3** han permés definir l'aparença visual de l'aplicació. A través de classes pròpies i variables de disseny s'ha aconseguit una presentació clara i coherent, a més d'un estil visual modern. També s'han utilitzat propietats com `flexbox` i `grid` per a l'organització de continguts, i s'han definit estats de focus i interacció accessibles.

### 5.5.3. JavaScript

El llenguatge **JavaScript** ha sigut utilitzat per a afegir interactivitat a l'aplicació. Concretament, s'ha implementat el botó de còpia de codi, el desplegament de *tooltips* i la gestió d'alguns elements dinàmics. Tot i que l'aplicació manté un caràcter majoritàriament estàtic, aquestes funcionalitats aporten una experiència d'ús més pràctica i immediata.

### 5.5.4. WAI-ARIA

Les especificacions de **WAI-ARIA** (*Accessible Rich Internet Applications*) s'han emprat de manera puntual per a complementar el marcat HTML quan ha sigut necessari millorar el nom accessible, establir rols o indicar estats. L'ús d'ARIA s'ha limitat a aquells casos en què l'HTML per si sol no oferia suficient informació als lectors de pantalla.

### 5.5.5. Eines de validació

S'han utilitzat diverses eines per a comprovar la qualitat i l'accessibilitat de l'aplicació:

- **WAVE**: per a l'anàlisi automàtica de contrastos, estructura de la pàgina i errors comuns d'accessibilitat.
- **Lighthouse**: per a l'avaluació integrada dins de Chrome, que proporciona mètriques d'accessibilitat junt amb rendiment i bones pràctiques.

### 5.5.6. Entorn de desenvolupament

El desenvolupament s'ha dut a terme principalment amb:

- **Visual Studio Code**, com a editor de codi principal.
- **XAMPP**, per a la configuració d'un entorn local amb servidor web durant la fase de proves.

### 5.5.7. Redacció de la memòria

Per a la redacció d'aquesta memòria s'ha utilitzat **LaTeX**, gestionat mitjançant la plataforma en línia **Overleaf**. Aquesta elecció ha permés mantindre una estructura organitzada, una bibliografia coherent i un format acadèmic de qualitat.

---

# CAPÍTOL 6

## Conclusions i Treball Futur

---

### 6.1 Conclusions generals

---

Aquest treball ha tingut com a objectiu principal el disseny i desenvolupament d'una eina web senzilla, estàtica i didàctica que permeta als desenvolupadors comprendre i aplicar bones pràctiques d'accessibilitat en el disseny d'interfícies web per a persones amb discapacitat visual.

Tot i que l'aplicació resultant no presenta una gran complexitat a nivell de codi, el valor real del projecte radica en el procés de recerca, anàlisi i síntesi d'informació sobre accessibilitat web, així com en l'estudi detallat de com traslladar aquestes recomanacions a exemples pràctics i comprensibles, estalviant tot aquest treball a futurs desenvolupadors.

La dedicació principal, per tant, s'ha centrat a entendre en profunditat les necessitats dels usuaris, el funcionament dels lectors de pantalla i els criteris normatius del W3C, per a després transformar aquest coneixement en una eina senzilla, pedagògica accessible i útil.

Des d'un punt de vista tècnic i metodològic, la solució:

- Integra una col·lecció representativa d'elements HTML (encapçalaments, imatges, taules, formularis, navegació, multimèdia i components interactius), el seu pertinent codi i descripcions que expliquen el perquè de cada decisió, l'ús d'atributs, el nom accessible i l'impacte en lectors de pantalla.
- Prioritza l'ús d'HTML semàntic, reservant WAI-ARIA per a casos on és estrictament necessari.
- Ofereix una experiència d'aprenentatge i posada en pràctica immediata mitjançant botons de còpia de codi i estructures repetibles (element-header, botó d'informació amb data-info, bloc <details> amb codi), que acceleren la transferència a projectes reals.
- Acompanya la part pràctica amb criteris d'avaluació mitjançant eines com WAVE i Lighthouse, de manera que l'usuari aprèn també a verificar els resultats una vegada desenvolupada l'aplicació.

### 6.1.1. Aportacions principals

Les aportacions més rellevants d'aquesta memòria es poden sintetitzar en quatre punts:

1. **Orientació pedagògica:** L'eina no és únicament un catàleg; és un itinerari formatiu amb exemples, justificacions i bones pràctiques que faciliten la comprensió i l'aplicació immediata.
2. **Enfocament centrat en l'usuari amb discapacitat visual:** s'ha atès, de manera prioritària, la compatibilitat amb lectors de pantalla i el nom accessible dels elements.
3. **Qualitat del marcat i coherència visual:** la combinació d'HTML semàntic, patrons d'interacció coherents i estilització acurada crea una base sòlida i replicable.
4. **Verificació i criteris:** s'ha incorporat l'ús de validadors i mètriques d'accésibilitat, fent èmfasi en criteris objectius (p. ex., contrast i estructura).

### 6.1.2. Limitacions

Aquesta aplicació pot créixer d'una forma pràcticament il·limitada, ja que aquest treball s'ha centrat solament en la ceguesa total, per tant possibles ampliacions per a l'aplicació podrien ser:

- **Ampliació de la discapacitat visual:** Queden per aprofundir escenaris de baixa visió (ceguera parcial), sensibilitat al contrast i anomalies de percepció cromàtica com daltonisme i derivats.
- **Internacionalització:** L'aplicació actualment no incorpora un mecanisme complet de canvi d'idioma ni localització de continguts i atributs. amb un element de cerca.
- **Personalització d'experiència:** La configurabilitat (temes d'alt contrast, tipografies, espaiat, paletes) encara no està exposada com a preferències d'usuari persistents.
- **Avaluació amb usuaris:** tot i la validació amb eines automàtiques, aquesta aplicació no compta amb la validació de persones amb ceguersa ni d'altres desenvolupadors.

### 6.1.3. Lliçons apreses

El desenvolupament ha evidenciat que:

- L'HTML semàntic és el pilar de l'accessibilitat; ARIA ha de ser complementària i precisa.

- La documentació d'exemples amb “per què” i “quan no” és tan important com el codi en si, donat que de res val que un desenvolupador copie el codi si no sap quins son els atributs que ha de tindre en compte.
- La coherència dels patrons d’interacció (estructura repetible, botons, *tool-tips*, <details>) millora l’aprenentatge i la comoditat d’us.

#### 6.1.4. Relació amb les assignatures del grau

El treball es troba clarament alineat amb diverses assignatures del Grau en Enginyeria Informàtica de la UPV, especialment aquelles que aborden usabilitat, desenvolupament web i qualitat del programari. A continuació es detalla la contribució de cadascuna a les competències posades en pràctica en aquest TFG.

##### Eix principal (directament relacionades)

- **Interfícies Persona–Computador (IPC).** Fonaments d’usabilitat, disseny centrat en l’usuari, navegació per teclat, gestió del focus i criteris d’accessibilitat. La guia i els exemples del TFG apliquen aquests principis a patrons HTML/ARIA i validacions amb lector de pantalla.
- **Enginyeria del Programari.** Enginyeria de requisits (RF/RNF), traçabilitat, documentació, proves i qualitat.
- **Desenvolupament Web.** Ús d’HTML semàntic, CSS i JavaScript lleuger; patrons WAI–ARIA només quan l’HTML no és suficient (*native first*). L’aplicació il·lustra bones pràctiques amb exemples funcionals i codi copiable.
- **Tecnologies de Sistemes d’Informació en la Xarxa / Sistemes i Serveis en Xarxa.** Publicació en entorns estàtics, bones pràctiques de càrrega i cache, i integració d’eines d’auditoria web (WAVE, Lighthouse) que el TFG utilitza per verificar la qualitat.

##### Eix de qualitat, ètica i gestió

- **Deontologia i Professionalitat.** Inclusió i disseny universal com a deure professional; l’accessibilitat s’emmarca com a requisit no funcional essencial.
- **Gestió de Projectes.** Planificació, fites, memòria de treball.

##### Assignatures de suport metodològic

- **Estructura de Dades i Algorismes / Programació.** Rigor, claredat i modularitat en el codi d’exemple; components repetibles i fàcilment ampliables.
- **Desenvolupament Centrat en l’Usuari (opt.).** Iteració amb heurístiques d’accessibilitat i comprovació amb lectors de pantalla com a *pseudo-tests* d’usuari.

## 6.2 Treball futur

---

A continuació es defineix una fulla de ruta d'evolució de l'aplicació, centrada en ampliar l'abast cap a baixa visió, personalització, internacionalització i una avaliació més robusta amb usuaris. Es presenta organitzada per àrees, amb objectius, criteris d'acceptació i consideracions d'implementació.

### 6.2.1. Accessibilitat per a ceguesa parcial (baixa visió)

**Objectiu:** oferir ajustos visuals que milloren la llegibilitat i la percepció de l'es-structura per a persones amb baixa visió.

**Línies d'acció:**

- Escala tipogràfica i espaiat: control de mida de font, alçària de línia i espaiat entre paràgrafs i lletres; previsualització en temps real.
- Guia visual de focus i lectura: reforç d'estats de focus amb contorns visibles i persistents; possibilitat d'activar un marcador de línia (subratllat o ressaltat suau).
- Reducció de soroll visual: opcions per ocultar elements no essencials, mini-mitzant distraccions i millorant la figura-fons.

### 6.2.2. Mode d'alt contrast

**Objectiu:** Incorporar un tema d'alt contrast i respectar les preferències del sistema operatiu.

**Línies d'acció:**

- Tema alt contrast: definició d'un conjunt de variables CSS amb relacions de contrast  $\geq 7:1$ .
- Preferències de sistema: suport de @media (prefers-contrast: more) i compatibilitat amb High Contrast Mode.
- Focus i estat: contorns i indicadors d'estat sempre visibles i no dependents del color únicament.

### 6.2.3. Paletes de colors per a daltonisme

**Objectiu:** proporcionar una pàgina específica amb paletes accessibles i eines de comprovació per a diferents tipus de daltonisme.

**Línies d'acció:**

- Galeria de paletes: col·lecció amb variació de lluminositat i no només de to, amb etiquetes d'aptitud.

- Simulació i anàlisi: demostració visual i càlcul automàtic de contrast.
- Evitar codificar només amb color: patrons alternatius (tramat, icones, text) per a informació crítica.

#### 6.2.4. Tipografies recomanades per a facilitar la visió

**Objectiu:** orientar el desenvolupador sobre l'elecció de tipografies que milloren la llegibilitat.

**Línies d'acció:**

- Criteris tipogràfics: diferència clara entre glifs similars, traç net i formes obertes.
- Catàleg curat: fonts com Inter, Source Sans, Noto Sans, Lexend, Atkinson Hyperlegible.
- Paràmetres d'ajust: mòduls per provar mida, espaiat i alçària de línia.

#### 6.2.5. Internacionalització i canvi d'idioma

**Objectiu:** fer l'aplicació multilingüe i accessible.

**Línies d'acció:**

- Estratègia d'i18n: paquets de cadenes (.json) i ús correcte de lang.
- Comutador d'idioma accessible: component amb nom accessible i persistència.
- SEO i navegadors: ús de hreflang i evitar traduir codi amb translate="no".

### 6.3 Tancament

---

El treball realitzat demostra que és viable construir una eina formativa que, des de l'HTML semàntic i patrons d'interacció consistents, facilite l'adopció de bones pràctiques d'accessibilitat. L'evolució natural del projecte passa per ampliar el focus cap a baixa visió i personalització (alt contrast, paletes per a daltonisme, tipografies), incorporar internacionalització i establir un cicle de qualitat amb proves automàtiques i validació amb persones usuàries. Això no sols reforçarà l'impacte pedagògic de l'eina, sinó que la convertirà en un recurs pràctic encara més útil per a equips de desenvolupament que busquen incorporar l'accessibilitat en origen i no com a etapa final d'auditoria.

# Annex. Objectius de Desenvolupament Sostenible (ODS)

## **Grau de relació del treball amb els Objectius de Desenvolupament Sostenible**

Objectiu de Desenvolupament Sostenible	Grau de relació
Fi de la pobresa	No procedeix
Fam zero	No procedeix
Salut i benestar	Baix
Educació de qualitat	Alt
Igualtat de gènere	No procedeix
Aigua neta i sanejament	No procedeix
Energia assequible i no contaminant	No procedeix
Treball digne i creixement econòmic	Mitjà
Indústria, innovació i infraestructures	Alt
Reducció de les desigualtats	Alt
Ciutats i comunitats sostenibles	Mitjà
Producció i consum responsables	Baix
Acció pel clima	No procedeix
Vida submarina	No procedeix
Vida d'ecosistemes terrestres	No procedeix
Pau, justícia i institucions sòlides	Baix
Aliances per aconseguir objectius	Mitjà

**Taula 6.1:** Grau de relació del treball amb els Objectius de Desenvolupament Sostenible (ODS)

## Reflexió sobre la relació del TFG amb els ODS

Aquest Treball de Fi de Grau s'ha centrat en la investigació i el desenvolupament d'una eina web destinada a facilitar la creació d'aplicacions accessibles per a persones amb discapacitat visual. Tot i que a primera vista pot semblar una aportació tècnica circumscrita a l'àmbit de la informàtica, la seua incidència real és més amplia i es relaciona directament amb diversos Objectius de Desenvolupament Sostenible (ODS).

En primer lloc, la relació més evident es dona amb l'**ODS 4: Educació de qualitat**. L'aplicació desenvolupada ofereix recursos i exemples pràctics perquè els futurs desenvolupadors web aprenguen a integrar l'accessibilitat des de les fases inicials dels seus projectes. En aquest sentit, contribueix a millorar la formació en competències digitals clau i fomenta un aprenentatge inclusiu que garanteix que les tecnologies puguen ser utilitzades per totes les persones. La plataforma no sols és una guia didàctica, sinó també una eina de sensibilització sobre la importància de crear un entorn digital sense barreres.

Estretament relacionat amb això trobem l'**ODS 10: Reducció de les desigualtats**. Les persones amb discapacitat visual es troben sovint en situació de desvantatge a l'hora d'accendir a continguts digitals. En promoure bones pràctiques de disseny accessible, el treball contribueix directament a reduir aquesta bretxa i a garantir la igualtat d'oportunitats en l'accés a la informació i als serveis en línia. D'aquesta manera, la solució desenvolupada té un impacte social clar, ja que fomenta una societat més justa i inclusiva.

L'**ODS 9: Indústria, innovació i infraestructures** també es veu reflectit en el treball. L'aplicació apostà per la innovació en l'àmbit del desenvolupament web, posant l'èmfasi no només en la funcionalitat o l'estètica, sinó també en la responsabilitat social de la tecnologia. Impulsar la construcció d'infraestructures digitals accessibles és clau per a una economia i una societat que depenen cada vegada més d'internet i dels serveis en línia.

D'altra banda, encara que de manera més indirecta, el treball pot relacionar-se amb l'**ODS 8: Treball digne i creixement econòmic**, en la mesura que fomenta la formació de professionals més qualificats i conscients de la importància de desenvolupar productes accessibles, la qual cosa pot traduir-se en millors oportunitats laborals i en empreses més competitives.

En menor mesura, el projecte té connexió amb l'**ODS 11: Ciutats i comunitats sostenibles**, ja que les ciutats del segle XXI no poden ser considerades plenament sostenibles si els seus serveis digitals no són accessibles per a tota la ciutadania. També amb l'**ODS 17: Aliances per aconseguir els objectius**, donat que la difusió de bones pràctiques en accessibilitat és un repte global que requereix la col·laboració d'institucions educatives, empreses tecnològiques i administracions públiques.

Finalment, encara que en grau baix, el projecte es pot vincular a l'**ODS 3: Salut i benestar** i a l'**ODS 16: Pau, justícia i institucions sòlides**. L'accessibilitat digital té un impacte positiu en el benestar de les persones, en particular aquelles amb discapacitat, i reforça la idea d'institucions i serveis més justos, transparents i accessibles.

En conclusió, aquest TFG s’alinea de manera clara i destacada amb els ODS 4, 9 i 10, i de manera complementària amb altres objectius com el 5, 8, 11 i 17. La seuva aportació principal és demostrar que la tecnologia no sols ha de ser eficient, sinó també inclusiva i compromesa amb la societat. D’aquesta manera, es posa de manifest que fins i tot projectes que aparentment tenen un abast reduït, com una aplicació web educativa, poden contribuir de forma significativa als reptes globals establerts per l’Agenda 2030.

# Bibliografia

---

- [1] Aleph Accessibility. Wcag versioning, explained. <https://www.alephaccessibility.net/resources/infographic-wcag-versioning-explained>, 2023. Infografia que resumeix les versions 1.0, 2.0, 2.1, 2.2 i la futura 3.0. Consulta: agost 2025.
- [2] Wikipedia. Web accessibility. [https://en.wikipedia.org/wiki/Web\\_accessibility](https://en.wikipedia.org/wiki/Web_accessibility), 2025. Tipus d'avaluació i conceptes clau en accessibilitat web. Consulta: agost 2025.
- [3] WebAIM. The webaim million - 2025. <https://webaim.org/projects/million/>, 2025. Estudi anual que analitza 1 milió de pàgines d'inici, indicant que el 94,8% contenen errors d'accessibilitat detectables segons WCAG 2. Consulta: agost 2025.
- [4] Digital Trust Index. The state of digital accessibility in europe 2025. <https://www.digitaltrustindex.eu/>, 2025. Estudi que revela que el 92,97% de les pàgines d'inici europees no compleixen els requisits d'accessibilitat. Consulta: agost 2025.
- [5] W3C Web Accessibility Initiative. Web content accessibility guidelines (wcag) overview. <https://www.w3.org/WAI/standards-guidelines/wcag/>, 2023. Història i evolució de les WCAG des de 1.0 fins 2.2. Consulta: agost 2025.
- [6] Tim Berners-Lee. The world wide web: A very short personal history, 1998. Accés el 30 d'agost de 2025.
- [7] Uservision. The web content accessibility guidelines: Past, present, future. <https://uservision.co.uk/thoughts/the-web-content-accessibility-guidelines-past-present-future/>, 2024. Nous criteris WCAG 2.1 i 2.2, i tendències futures. Consulta: agost 2025.
- [8] Accessible Web. Wcag version history. <https://accessibleweb.com/wcag/wcag-version-history/>, 2024. Cronologia de versions WCAG i nivells de conformitat. Consulta: agost 2025.
- [9] Deque Systems. What is wcag 2.1: History and overview. <https://www.deque.com/blog/what-is-wcag-2-1-history/>, 2018. Evolució i canvis introduïts a WCAG 2.1. Consulta: agost 2025.

- [10] 3Play Media. Understanding wcag guidelines. <https://www.3playmedia.com/blog/understanding-wcag-guidelines/>, 2024. Explicació dels principis POUR de WCAG 2.0 i posteriors. Consulta: agost 2025.
- [11] W3C Web Accessibility Initiative. Web content accessibility guidelines (wcag) overview. <https://www.w3.org/WAI/standards-guidelines/wcag/>, 2023. Resum oficial de les WCAG i la seua evolució històrica. Consulta: agost 2025.
- [12] ETSI. EN 301 549: Accessibility requirements for ict products and services. <https://www.etsi.org/standards#EN%20301%20549>, 2023. Norma europea que estableix requisits d'accessibilitat TIC. Consulta: agost 2025.
- [13] Gobierno de España. Real decreto 1112/2018, de accesibilidad de los sitios web y aplicaciones para dispositivos móviles del sector público. <https://www.boe.es/buscar/act.php?id=BOE-A-2018-12699>, 2018. Normativa española sobre accessibilitat digital al sector públic. Consulta: agost 2025.
- [14] W3C Web Accessibility Initiative. Introduction to web accessibility. <https://www.w3.org/WAI/fundamentals/accessibility-intro/>, 2023. Introducció als conceptes bàsics d'accessibilitat web. Consulta: agost 2025.
- [15] WebAIM. Introduction to web accessibility. <https://webaim.org/intro/>, 2024. Guia introductòria sobre els principis i pràctiques d'accessibilitat web. Consulta: agost 2025.
- [16] Deque Systems. Deque university - accessibility training. <https://dequeu niversity.com/>, 2025. Recursos formatius i guies sobre bones pràctiques d'accessibilitat. Consulta: agost 2025.
- [17] Wikipedia. Web content accessibility guidelines. [https://en.wikipedia.org/wiki/Web\\_Content\\_Accessibility\\_Guidelines](https://en.wikipedia.org/wiki/Web_Content_Accessibility_Guidelines), 2025. Versió 2.0 acceptada com a ISO/IEC 40500:2012 i integració en EN 301 549. Consulta: agost 2025.
- [18] European Commission. European accessibility act. [https://en.wikipedia.org/wiki/European\\_Accessibility\\_Act](https://en.wikipedia.org/wiki/European_Accessibility_Act), 2019. Directiva europea per garantir l'accessibilitat de productes i serveis digitals. Consulta: agost 2025.
- [19] Jonathan Lazar, Andrea Allen, Julie Kleinman, and Chris Malarkey. What frustrates screen reader users on the web: A study of 100 blind users. *International Journal of Human-Computer Interaction*, 22(3):247–269, 2007.
- [20] The Guardian. Denying access doesn't pay. <https://www.theguardian.com/technology/2004/apr/15/internet.epublic>, 2004. Reportatge que indica que el 81
- [21] Justia Law. Robles v. domino's pizza, llc — court of appeals decision. <https://law.justia.com/cases/federal/appellate-courts/ca9/17-55504/17-55504-2019-01-15.html>, 2019. Sentència del Novè Circuit sobre aplicació del ADA a webs i apps. Consulta: agost 2025.

- [22] Level Access. Top penalties for eaa non-compliance. <https://www.levelaccess.com/blog/penalties-for-eaa-non-compliance/>, 2025. Itàlia: multes fins al 5
- [23] Laura Gómez and Pablo Hernández. Artificial intelligence for web accessibility: Opportunities and challenges. *ACM Transactions on Accessible Computing*, 16(4):1–22, 2023.
- [24] Cynthia Bennett and Os Keyes. The challenges of using ai for accessibility. *AI & Society*, 35(3):563–574, 2020.
- [25] World Economic Forum. Global ai governance: Risks and recommendations. <https://www.weforum.org/reports/global-ai-governance>, 2022. Informe sobre riscos corporatius en l'ús de la intel·ligència artificial. Consulta: agost 2025.
- [26] Wei Zhang and Lin Wu. Limits of automated accessibility testing: Ai and the user experience. In *Proceedings of the 23rd International ACM SIGACCESS Conference on Computers and Accessibility*, pages 45–56, 2021.
- [27] W3C. About w3c. <https://www.w3.org/about/>, 2024. Pàgina oficial del W3C on es descriu la seua missió i objectius. Consulta: agost 2025.
- [28] W3C. Html compatibility and interoperability issues. <https://www.w3.org/TR/html/compatibility/>, 2023. Document que recull problemes històrics i actuals de compatibilitat entre navegadors i versions d'HTML. Consulta: agost 2025.
- [29] World Wide Web Consortium. World wide web consortium — viquipèdia, l'enciclopèdia lliure. [https://ca.wikipedia.org/wiki/World\\_Wide\\_Web\\_Consortium](https://ca.wikipedia.org/wiki/World_Wide_Web_Consortium), 2025. Consulta: maig 2025.
- [30] Universitat d'Alacant. Què és wai — accessibilitat web. <https://accesibilidadweb.dlsi.ua.es/?menu=que-es-wai>, 2025. Consulta: maig 2025.
- [31] World Wide Web Consortium. About wai — web accessibility initiative. <https://www.w3.org/WAI/about/>, 2025. Consulta: maig 2025.
- [32] World Wide Web Consortium. Missió de la iniciativa per a l'accessibilitat web (wai). <https://www.w3.org/WAI/about/#mission>, 2025. Consulta: maig 2025.
- [33] World Wide Web Consortium. Wai-aria: Accessible rich internet applications. <https://www.w3.org/WAI/standards-guidelines/aria/>, 2025. Consulta: maig 2025.
- [34] World Wide Web Consortium. Wai-aria: Accessible rich internet applications. <https://www.w3.org/WAI/standards-guidelines/aria/>, 2023. WAI-ARIA esdevingué Recomanació del W3C el 6 de juny de 2023. Consulta: maig 2025.
- [35] World Wide Web Consortium. Using wai-aria in html. <https://www.w3.org/TR/using-aria/>, 2024. Inclou la recomanació: "If you can use a native HTML element... then do so". Consulta: maig 2025.

- [36] World Wide Web Consortium. Wai-aria authoring practices guide: Read me first. <https://www.w3.org/WAI/ARIA/apg/practices/read-me-first/>, 2024. Explica que els rols, estats i propietats ARIA no canvien el comportament ni l'estil visual dels elements HTML. Consulta: maig 2025.
- [37] Bureau of Internet Accessibility. Should my website use wai-aria for accessibility? <https://www.boia.org/blog/should-my-website-use-wai-aria-for-accessibility>, 2024. L'article assenyala que WAI-ARIA pot ser essencial per a certs components que no poden ser fets accessibles amb HTML natiu. Consulta: maig 2025.
- [38] Mozilla Developer Network. Aria: Roles reference. <https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Reference/Roles>, 2025. Documentació sobre els rols ARIA i la seu aplicació semàntica. Consulta: juny 2025.
- [39] World Wide Web Consortium. Wai-aria authoring practices guide: Accessible names and descriptions. <https://www.w3.org/WAI/ARIA/apg/practices/names-and-descriptions/>, 2024. Guia sobre com utilitzar noms i descripcions accessibles amb ARIA. Consulta: juliol 2025.
- [40] W3C. Web content accessibility guidelines (wcag) 2.1, 2018. Accés: agost 2025.
- [41] W3C. Web content accessibility guidelines (wcag) 2.2, 2023. Accés: agost 2025.
- [42] W3C. Web content accessibility guidelines (wcag) 2.1 – success criterion 1.4.3: Contrast (minimum), 2018. Accés: agost 2025.
- [43] WebAIM. Contrast checker, 2022. Accés: agost 2025.
- [44] WebAIM. Contrast and color accessibility, 2022. Accés: agost 2025.
- [45] Mozilla Developer Network. Accessibility html — learn web development. [https://developer.mozilla.org/en-US/docs/Learn\\_web\\_development/CORE/Accessibility/HTML](https://developer.mozilla.org/en-US/docs/Learn_web_development/CORE/Accessibility/HTML), 2025. Guia sobre com utilitzar elements HTML per millorar l'accessibilitat abans d'usar ARIA. Consulta: juny 2025.
- [46] Mozilla Developer Network. aria-label (aria attribute) — mdn web docs. <https://developer.mozilla.org/en-US/docs/Web/Accessibility/ARIA/Reference/Attributes/aria-label>, 2025. Documentació detallada sobre l'atribut aria-label per fer contingut més accessible. Consulta: juni 2025.
- [47] WebAIM. Using voiceover to evaluate web accessibility. <https://webaim.org/articles/voiceover/>, 2021. Guia sobre com usar VoiceOver a Mac per testar l'estructura i la navegació d'un lloc web. Consulta: juliol 2025.
- [48] Deque University. Buttons must have discernible text — axerule4.1 / 4.10. <https://dequeuniversity.com/rules/axe/4.1/button-name>, 2025. Explica que els botons han de tenir text discernible o atributs ARIA com aria-label. Consulta: juliol 2025.