
Cuaderno de diseño y prototipaje de una propuesta de mejora de una interfaz

PID_00278936

Jonathan Chacón
Carla Leal

Tiempo mínimo de dedicación recomendado: 3 horas



Universitat
Oberta
de Catalunya

Jonathan Chacón**Carla Leal**

El encargo y la creación de este recurso de aprendizaje UOC han sido coordinados por la profesora: Paloma González Díaz

La revisión de este recurso de aprendizaje UOC ha sido coordinada por la profesora: Susana Acedo Nadal

Segunda edición: septiembre 2022
© de esta edición, Fundació Universitat Oberta de Catalunya (FUOC)
Av. Tibidabo, 39-43, 08035 Barcelona
Autoría: Jonathan Chacón, Carla Leal
Producción: FUOC



Los textos e imágenes publicados en esta obra están sujetos –excepto que se indique lo contrario– a una licencia Creative Commons de tipo Reconocimiento-Compartir igual (BY-SA) v.3.0. Se puede modificar la obra, reproducirla, distribuirla o comunicarla públicamente siempre que se cite el autor y la fuente (Fundació per a la Universitat Oberta de Catalunya), y siempre que la obra derivada quede sujeta a la misma licencia que la obra original. La licencia completa se puede consultar en: <http://creativecommons.org/licenses/by-sa/3.0/es/legalcode.es>

Índice

Introducción.....	5
1. Reto 1: Investigación: exploración del escenario que hay que trabajar (PEC).....	9
1.1. Definir los perfiles de usuario	9
1.2. Investigación de la competencia	13
1.3. Aspectos básicos de la interacción persona-ordenador	16
1.4. Conclusiones	17
2. Reto 2: Definición: síntesis de la propuesta.....	18
2.1. Diagrama de flujo	19
2.2. Test con usuarios	21
2.2.1. Definición del alcance de la evaluación	21
2.2.2. Definición del perfil de los participantes y captación ...	22
2.2.3. Definición del guion de las sesiones	22
2.2.4. Realización de las sesiones	23
2.2.5. Análisis de resultados y elaboración de las conclusiones	23
2.3. Propuesta de soluciones	24
2.4. Conclusiones	26
3. Reto 3: Diseño universal y accesibilidad (PEC 2).....	27
3.1. Diseño universal	27
3.2. Accesibilidad	33
3.2.1. Estándares de la accesibilidad web	34
3.3. Niveles de accesibilidad A, AA y AAA	41
3.3.1. Herramientas para la evaluación de la accesibilidad web	42
3.4. Conclusiones	43
4. Reto 4: Generación: implementación de un prototipo de baja fidelidad.....	44
4.1. <i>User journey</i> con <i>happy path</i>	44
4.2. Herramientas de prototipaje	45
4.3. Conceptos fundamentales del IPO	47
4.4. Conclusiones	48
5. Reto 5: Evaluación: test y valoración del prototipo.....	49
5.1. Test con usuario	49
5.2. Evaluar la accesibilidad de nuestra propuesta	50
5.3. Elaboración del informe final	50

5.3.1.	Portada del documento	50
5.3.2.	Índice y resumen	51
5.3.3.	Introducción	51
5.3.4.	Estructura	51
5.3.5.	Conclusiones	52
5.3.6.	Referencias	52
5.4.	Conclusiones	53
6.	Conclusiones. ¿Qué hemos aprendido?.....	54

Introducción

¿Qué es la interacción? La Real Academia Española (RAE) define la **interacción** como

«La acción que se ejerce recíprocamente entre dos o más objetos, personas, agentes, fuerzas, funciones, etc.»

Dentro de los ámbitos de esta definición en esta asignatura nos centraremos en la interacción entre las personas y la tecnología. No pondremos el foco de atención en el estudio de las personas, ni en el estudio de la tecnología, sino que nos centraremos en estudiar cómo diseñar este puente entre las dos.

Cuando hablamos de diseño de interacción, la Fundación sobre el Diseño de Interacción (<https://www.interaction-design.org>) lo define como lo que pasa entre los usuarios y los productos. Pero de productos hay de muchos tipos. Entonces, ¿en cuáles nos centraremos? Dentro del marco de esta asignatura nos centraremos en productos del ámbito tecnológico, y más concretamente en los que hacen de puente entre estos productos tecnológicos o *software* y la persona: las **interfaces de usuario de páginas web o apps**.

Pero, ¿qué es una interfaz? Citando nuevamente a la RAE, esta define **interfaz** como

«la conexión, física o lógica, entre un ordenador y un usuario [...]»

Cuando hablamos de diseñar la interfaz, entendemos que es diseñar la cara pública del *software*, lo que verá el usuario y manipulará para realizar acciones concretas.

Por ejemplo, el portal para comprar la entrada a un concierto, o la parte que vemos en una app para hacer reservas de hoteles o vuelos.

En este contexto, podemos decir que en esta asignatura nos centraremos en la parte del diseño de interacción que diseña interfaces que sean comprensibles para los usuarios y les permitan completar sus objetivos de la mejor forma posible.

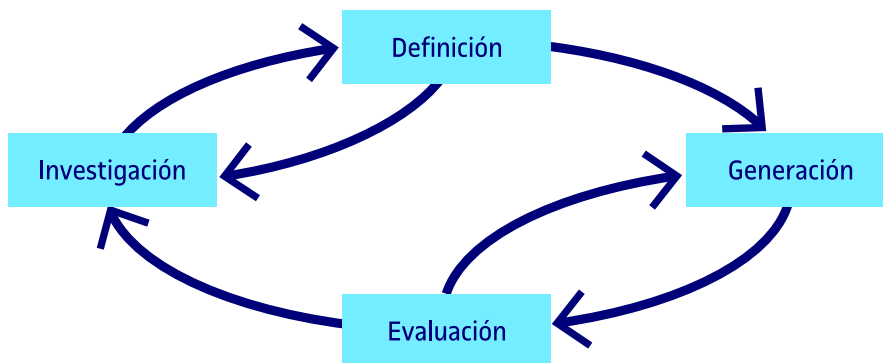
Entonces, el objetivo principal de la asignatura Diseño de Interacción y Diseño de Interfaces será proporcionar los conocimientos y las competencias necesarios para que podamos ser capaces de diseñar y evaluar una interfaz.

La **metodología** para el diseño de interfaces que aplicaremos dentro de la asignatura seguirá las siguientes cuatro etapas:

- investigación,
- definición,
- generación y
- evaluación.

En un proyecto real, realizaríamos diferentes iteraciones en estas cuatro fases para completar un producto tecnológico. Además, es normal volver a etapas anteriores para definir nuevamente conceptos a medida que el proyecto va avanzando. Podríamos tener un caso real donde se ha hecho una investigación, y con los resultados de esta se ha hecho una definición. Esta nueva definición ha dado pie a llevar a cabo una nueva investigación de aspectos que no se habían detectado anteriormente. Redefinir nuevamente los requerimientos u objetivos, generar una propuesta, evaluar la propuesta, volver a generar una segunda propuesta con los resultados de la evaluación y evaluar nuevamente. Y acabar marcando unos nuevos aspectos clave que darían pie a investigar nuevamente. Esta metodología también se puede aplicar a diferentes niveles de granularidad. Podemos entender un grano grande como la idea general de toda la interfaz, y un grano fino como un aspecto muy específico, como podría ser la mejor manera de diseñar un botón para que el usuario lo entienda, lo pueda usar y reciba *feedback* para saber que ha hecho clic (ver Figura 1).

Figura 1. Metodología para el diseño de interfaces basada en cuatro etapas



En esta asignatura concreta hemos fijado el alcance en solo una iteración concreta para cada una de las diferentes etapas. Hemos relacionado las actividades con las etapas de la metodología de la forma siguiente:

1) Investigación

- PEC 1 - Exploración del escenario a trabajar.

2) Definición

- 1.ª parte de la práctica - Síntesis de la propuesta.

3) Generación

- PEC 2 - Diseño universal y accesibilidad.
- 2.ª parte de la práctica - Implementación de un prototipo de alto nivel.

4) Evaluación

- 3.ª parte de la práctica - Test y valoración del prototipo con usuarios.

Definir una interfaz desde cero supone mucho de tiempo y recursos, y quedaría fuera del alcance de la asignatura. Por lo tanto, en esta asignatura os facilitaremos el primer día un caso de estudio concreto, que constará de un conjunto interfaces y unos requisitos. Como si recibiéramos el encargo de un cliente real. A partir de este caso de estudio, seguiremos las diferentes etapas de la metodología para que os familiaricéis con esta y con algunas de las herramientas concretas de cada etapa.

Caso de estudio de ejemplo

¿Qué aplicación usáis para escuchar música? Hoy en día nos encontramos que hay mucha diversidad de aplicaciones para escuchar música con diferentes opciones y posibilidades. Imaginemos que tenemos como objetivo mejorar la interacción y la experiencia del usuario a la hora de escuchar música. Este será nuestro objetivo a lo largo de la práctica.

Para empezar analizaremos las siguientes interfaces de estos tipos de servicio:

- **Spotify:** <https://www.spotify.com/es>.
- **Deezer:** <https://www.deezer.com/es/>.
- **Google Play Music:** <https://play.google.com/music/>.
- Podéis usar de ejemplo otras apps u otros webs que usáis para escuchar música.

El objetivo que hay que lograr al final de la práctica es mejorar la interacción con el servicio seleccionado, cumpliendo los cinco requisitos fundamentales:

- 1) Poder crear listas de canciones.
- 2) Organizar las canciones en categorías.
- 3) Tiene que haber un buscador que permita buscar por tipo de música, categoría o palabras clave.
- 4) El usuario tiene que poder compartir su lista de música.

5) El usuario tiene que poder ver recomendaciones de canciones similares a las que está escuchando.

Estos cinco requisitos constituyen el alcance inicial de la práctica para toda la asignatura.

1. Reto 1: Investigación: exploración del escenario que hay que trabajar (PEC)

Tomando como punto de partida el caso de estudio planteado en este primer reto, tenemos los tres objetivos principales siguientes:

- 1) Entender qué tipo de usuario usa nuestro servicio interactivo.
- 2) Explorar herramientas existentes similares para entender qué se está haciendo y cómo.
- 3) Analizar cómo se están aplicando los aspectos básicos de la interacción persona-ordenador en el caso dado.

Si empezamos a diseñar para todo el mundo, acabaremos diseñando para ningún usuario en particular. Por eso es muy importante empezar el proyecto definiendo cuál será el perfil de usuario que usará nuestra interfaz. El segundo objetivo consistirá en analizar las diferentes interfaces que nos plantea el caso de estudio. Las tomamos como si fueran la competencia y las analizaremos para ver como resuelven estas interfaces los requisitos planteados en el caso de estudio. Finalmente, el tercer objetivo, el más teórico, consiste en aprender los principios básicos de la interacción persona-ordenador. Tenemos que saber de qué recursos disponemos para diseñar nuestra propuesta, y estos principios son clave.

1.1. Definir los perfiles de usuario

Todo proyecto de diseño tiene que tener claro cuáles serán los usuarios que usarán su interfaz para entender sus necesidades particulares. Definir estos perfiles de usuario, además, es una tarea importante para que todo el equipo tenga presente para quien estamos trabajando.

Definimos los perfiles de usuario al principio pero los podemos ir ampliando y perfilando a medida que avanza el proyecto. En determinados puntos, nos puede ser muy útil para tomar decisiones sobre el diseño, y justificar la toma de decisiones para el perfil elegido.

Para construir estos perfiles podemos usar el método definido en el *design toolkit*:

https://aprenentatge.recursos.uoc.edu/design_toolkit/es/pdf/perfil_de_usuario.pdf.

Algunos aspectos para crear este perfil de usuario son ponerle un nombre, unos comportamientos, una demografía y fijar cuáles son sus necesidades y objetivos.

- Podemos empezar inventando uno **nombre y unos apellidos** y coger una **imagen** que nos sirva de referencia. Podéis coger fotografías sin derechos de autor o usar herramientas como <https://thispersondoesnotexist.com/>, que crea imágenes aleatorias por ordenador de personas imaginarias.
- Una vez tenemos el nombre y la fotografía, aprovechamos para crear la **demografía**. Pensad en una franja de edad que usaría nuestra interfaz, el estado civil, a qué se dedica, si vive con familia, dónde vive, entre otros factores. Todos estos datos hacen que el usuario parezca más real y facilite ponerse en su piel. Después definimos una serie de comportamientos: qué les gusta, cómo usan la tecnología, cómo pasan el día a día, qué horarios tienen, etc.
- Finalmente, cerramos definiendo unas **necesidades y unos objetivos** para usar estas interfaces.

A continuación, os facilitamos algunos ejemplos (ver Figura 2):

Nota

Normalmente, definirías los perfiles de usuario con nuestro equipo, para este trabajo podemos hacer una propuesta y explicarla a familiares y amigos para iterar el perfil de usuario.

Figura 2. Perfiles de usuario

Pablo García Pérez	Comportamientos
	Muestra interés por las TIC. Utiliza frecuentemente ordenadores, <i>smartphones</i> y tabletas. Trabaja la mayor parte del día. Tiene un horario laboral que tiene que compaginar con su pareja para recoger a su hijo.
Demografía	Necesidades y objetivos
30 a 40 años (38 años) Casado con Lluís (también trabaja) Trabaja en el ámbito de las comunicaciones Padre adoptivo de un hijo de 4 años Viven en un piso de alquiler en Barcelona	Crear listas de reproducción: música para él, listas para su hijo, etc. Seguir a gente con gustos parecidos.
Anna Bosch Baldrich	Comportamientos
	Se pasa todo el día con el <i>smartphone</i> . Utiliza con mucha frecuencia (prácticamente de manera constante internet: redes sociales), búsqueda de ofertas y compras en línea. Cierta independencia económica.
Demografía	Necesidades y objetivos
15 a 24 años (21 años) Soltera Estudiante universitaria Le gustan los animales (tiene un gato) Vive en un piso compartido de estudiantes en Barcelona	Necesita poder consultar las últimas tendencias. Necesita ver qué escuchan sus amigos.

Fotos supuestas de las personas Pablo García Pérez (arriba) y Anna Bosch Baldrich (abajo). Las imágenes se han generado usando una web que crea caras a partir de fotos reales de personas. Combina diferentes factores y crea una cara: [this person does not exist: https://thispersondoesnotexist.com/](https://thispersondoesnotexist.com/).

Una vez tengamos estos perfiles de usuario definidos es el momento de presentarlos a los amigos, compañeros de clase, compañeros de piso, familia, etc. Explicaremos el caso de estudio en el que estamos trabajando y por qué hemos definido estos perfiles, o por qué creemos que estos perfiles usan estas interfaces. Esto te hará ver si te has dejado algún aspecto o no has considerado algo que ellos sí que han tenido en cuenta. Aprovecha para iterar los que tienes, y crear alguno de más, si crees que te has olvidado de un perfil clave.

- Algunos aspectos que hay que considerar serían tener representantes de **diferentes franjas de edad**. No es lo mismo niños que adolescentes o universitarios y trabajadores.
- También hemos de tener presente la **perspectiva de género** y considerar el mismo número de perfiles de hombres que de mujeres. No se comportan ni actúan de la misma forma y, por lo tanto, deben tener representación dentro del diseño de nuestro producto o estaremos excluyendo a la mitad de la población que utilizará nuestra interfaz.

Podéis consultar el censo por edades y sexo para haceros una idea de los diferentes volúmenes de población usando las herramientas que proporciona el Instituto Nacional de Estadística (ver Figura 3) para entender el volumen del censo en España: <https://www.ine.es/>.

Figura 3. Ejemplo de consulta de censo en el Instituto Nacional de Estadística

The screenshot shows the INEbase web application interface. At the top, there's a search bar and navigation links like 'Censo Electoral', 'Sede electrónica', and 'Compartir'. Below the header, the main content area is titled 'Principales series de población desde 1998' and 'Comunidades Autónomas'. It includes a sub-header 'Población por comunidades, edad (grupos quinquenales), Españoles/Extranjeros, Sexo y Año'. The interface features several dropdown menus for selecting 'Comunidades', 'Edad (grupos quinquenales)', 'Españoles/Extranjeros', 'Sexo', and 'Año'. There are also buttons for 'Consultar selección' and 'Consultar todo'. A gear icon is visible on the right side of the interface.

Por ejemplo, imaginad que queremos hacernos una idea del volumen de usuarios en la comunidad autónoma de Cataluña de los perfiles que hemos definido: Pablo y Anna. El perfil de usuario Pablo tiene 38 años y hemos definido el perfil entre 30 y 40 años, mientras que Anna tiene 21, y es de nuestro perfil de entre 15 y 24 años. Si miramos en el INE el volumen de usuarios que estamos considerando tenemos los siguientes datos (ver Figura 4):

Figura 4. Ejemplo de datos de la consulta en el censo del Instituto Nacional de Estadística

Tabla

Gráfico

Mapa

	Hombres 2019
CATALUÑA	
30-34 años	
Espaníoles	165.618 ¹
35-39 años	
Espaníoles	213.769 ¹

Volumen de usuarios en Cataluña representado por el perfil Pablo

Tabla

Gráfico

Mapa

	Mujeres 2019
CATALUÑA	
15-19 años	
Espaníoles	162.377 ¹
20-24 años	
Espaníoles	143.710 ¹

Volumen de usuarios en Cataluña representado por el perfil Anna

Una vez trabajados y entendida la importancia de los perfiles de usuarios, continuaremos analizando la competencia o las interfaces similares.

1.2. Investigación de la competencia

El caso de estudio nos facilita unas interfaces que podemos considerar nuestra competencia. Las analizaremos para entender cómo lo hacen, qué hacen bien y qué aspectos podemos mejorar. Para desarrollar esta tarea aplicaremos el método llamado **Benchmarking**.

En este método definimos una serie de criterios para analizar todas las interfaces comprobando si:

- cumplen el criterio,
- cumplen parcialmente el criterio o
- no cumplen el criterio.

De esta forma acabaremos con una *checklist* de cada interfaz y podríamos llegar a elegir cuál se adecua más a nuestras necesidades. Además, aprovechamos esta exploración para anotar todos estos aspectos que encontramos que son positivos y aspectos que queremos evitar porque ya vemos que se podrían mejorar.

Para definir los criterios del *benchmarking*, partimos de los requisitos fundamentales del caso de estudio. En el ejemplo que trabajamos en este cuaderno teníamos los siguientes:

- 1) Poder crear listas de canciones.
- 2) Organizar las canciones en categorías.
- 3) Tiene que haber un buscador que permita buscar por tipo de música, categoría o palabras clave.
- 4) El usuario tiene que poder compartir su lista de música.
- 5) El usuario tiene que poder ver recomendaciones de canciones similares a las que está escuchando.

Para cada requisito desplegamos los criterios relacionados que pensamos que pueden ser interesantes de explorar. Por ejemplo, hemos visto que el requisito fundamental «compartir una lista de canciones» nos interesa y lo queremos trabajar con detenimiento.

Así pues, partiendo de este criterio, lo dividimos en cuatro criterios concretos:

- Permite compartir una lista de canciones con un amigo externo a la aplicación.

Enlace recomendado

Podéis encontrar toda la información sobre el método Benchmarking en el *design toolkit* de la UOC:

https://aprenentatge.recursos.uoc.edu/design_toolkit/es/pdf/benchmarking.pdf.

- Permite compartir una lista de canciones con usuarios de la aplicación.
- Permite seguir a otras listas de canciones ya existentes.
- Ofrece información adicional a las listas de canciones (seguidores, otra información...).

Por lo tanto, hemos detallado aspectos concretos relacionados con compartir que queremos trabajar en profundidad, puesto que creemos que compartir con amigos, estén o no en la plataforma, es importante, queremos poder seguir a listas de música de otra gente y tener información en tiempo real del éxito de las listas.

Una vez tenemos los criterios definidos, creamos una tabla como la que tienes a continuación, y empezaremos el análisis de las diferentes interfaces. En el documento de entrega es importante que describas cada aplicación con el nombre y una captura para recordar rápidamente cuál era. Después presentaremos la tabla parecida al ejemplo que os pasamos a continuación, donde hemos marcado si las interfaces permiten o no el criterio definido.

No obstante, a veces hay criterios que no serán «sí» o «no», y que quizás pueden hacerse parcialmente. En estos casos lo podemos especificar. En nuestro ejemplo tenemos marcas verdes para decir que «sí», cruces rojas para decir que «no», y podríamos incluir unos círculos naranjas, por ejemplo, para decir que lo cumplen parcialmente (ver Figura 5). También en estos casos podríamos incluir una columna para añadir descripciones y explicar cuál es la parte que hacen.

Figura 5. Ejemplo de cómo hacer una tabla para analizar los criterios del *benchmarking* en múltiples plataformas

CRITERIOS DE ANÁLISIS	Spotify	Deezer	Amazon Music
Organización por categorías <i>Existen categorías establecidas y organiza las canciones según estas.</i>			
Creador por tipo de música, palabra, categoría... <i>Dispone de un buscador para poder buscar las canciones por palabra clave, un tipo de música concreto, categoría...</i>			
Creación de lista de canciones <i>Posibilidad de crear listas de canciones y dar un nombre a esta lista.</i> <i>Ofrece otras interacciones con la lista de canciones: añadir imágenes, comentarios...</i>	 	 	
Compartir una lista de canciones <i>Permite compartir una lista de canciones con alguien externo a la aplicación.</i> <i>Permite compartir una lista de canciones con usuarios de la aplicación.</i> <i>Permite seguir a otras listas de canciones ya existentes.</i> <i>Ofrece información adicional a las listas de canciones (seguidores, otra información...).</i>	 	 	
Recomendaciones de canciones similares <i>Ofrece recomendaciones de canciones similares a las que se están escuchando.</i> <i>Posibilidad de hacer nosotros recomendaciones.</i>	 	 	
Descarga de canciones y escucharlas offline <i>Permite descargar las canciones para poder escucharlas sin necesidad de conexión a Internet.</i>			

Fuente: tabla realizada por la estudiante Montserrat Castrillo.

Una vez hemos acabado la tabla, mientras hacemos el análisis, es importante, como ya hemos comentado anteriormente, anotar los aspectos positivos y las cosas para mejorar. Puesto que acabaremos el *benchmarking* con una lista de *insights* o aspectos clave que hayas detectado. Recuerda, aspectos que ves que funcionan muy bien y quieres imitar, y aspectos que ves que no funcionan y quieres evitar.

Por ejemplo (ver Tabla 1):

Tabla 1

<i>Insights positivos</i>	<i>Insights para evitar</i>
Amplia distribución por categorías: canciones, artistas, álbumes, <i>playlists</i> , podcasts y videos, episodios de podcast, perfiles y géneros.	Para poder hacer descargas de las canciones y las listas hace falta que tengas una cuenta premium (previo pago).
Buscador por tipo de música, palabra, categoría... muy potente para encontrar rápidamente todo lo que nos puede ofrecer la aplicación en relación con una palabra clave: nombre del artista, nombre de la canción, estilo de música, etc.	No ofrece la posibilidad de compartir directamente en Facebook, Messenger, Twitter y otras redes sociales.
El usuario o usuaria puede hacer recomendaciones marcando el «me gusta» en diferentes momentos (icono de un coro).	La opción de compartir una canción no es tan intuitiva como con los álbumes o las listas de canciones.

Al acabar el *benchmarking*, podemos tener una interfaz que tiene más criterios completos que las otras, entonces podemos coger esta como plantilla para nuestro desarrollo. Si veis que hay unas cuantas con el mismo número de criterios cumplidos, podéis consultar la lista de *insights* para ver cuál puede dar más juego para mejorar.

Este *benchmarking* nos ha abierto una serie de posibilidades para trabajar durante la asignatura. Podemos trabajar en él para incluir mejoras que permitan completar criterios que no se han cumplido o se han cumplido parcialmente. Podemos trabajar para mejorar *insights* que hemos visto en esta interfaz que se pueden evitar, y actuar para evitarlos. Y finalmente, si creemos que podemos mejorar alguno de los aspectos que ya funcionan para hacerlos más eficientes, también los podemos trabajar.

1.3. Aspectos básicos de la interacción persona-ordenador

Aquí queremos presentar los conceptos fundamentales de la interacción persona-ordenador:

- metáfora,
- *affordance*,
- visibilidad,
- retroalimentación,
- restricción y
- modelo mental.

Nota

Recordad que podemos encontrar información sobre estos conceptos en el *design toolkit* de la UOC.

A continuación os presento los diferentes conceptos con una descripción breve y el enlace para encontrar todos los detalles de cada uno (ver Tabla 2).

Tabla 2

Concepto fundamental	Descripción abreviada y enlace de consulta
Metáfora	Usar una analogía para referirse a otro concepto de forma figurativa. Por ejemplo, los iconos de cualquier interfaz. Una hoja para denotar un documento, la metáfora del escritorio de Windows, o llamar ventana a la ventana del navegador. Más información aquí: http://design-toolkit.recursos.uoc.edu/es/metafora/ .
<i>Affordance</i>	Entenderemos como <i>affordance</i> a aquellas propiedades de nuestra interfaz que hacen que se autoexpliquen o nos permitan deducir su funcionamiento. Por ejemplo, un botón normalmente tiene un sombreado para dar la sensación que es un elemento 3D que puedes pulsar. Más información aquí: http://design-toolkit.recursos.uoc.edu/es/affordance/ .
Visibilidad	Es la propiedad que define que los elementos fundamentales de nuestro diseño sean fácilmente percibidos o visibles por el usuario. Más información aquí: http://design-toolkit.recursos.uoc.edu/es/visibilidad/ .

Concepto fundamental	Descripción abreviada y enlace de consulta
Retroacción	Hemos de trabajar nuestra interfaz para que el usuario sea consciente de que las acciones que pide se están ejecutando y el estado actual de estas. Pueden ser elementos textuales, lumínicos o incluso sonoros. Como el círculo dando vueltas mientras estamos esperando, el ruido al hacer clic, etc. Más información aquí: http://design-toolkit.recursos.uoc.edu/es/retroaccion/ .
Restricción	Este concepto marca que tenemos que restringir las opciones que puede hacer el usuario para evitar equivocaciones y facilitar la elección de opciones. En una calculadora si pulsas el signo «más» y después el signo «menos», automáticamente quita el «más», puesto que no tiene sentido sumar y restar a la vez. Más información aquí: http://design-toolkit.recursos.uoc.edu/es/restriccion/ .
Modelo mental	Entendemos como modelo mental a aquellas propiedades que por similitud a otras que ya hemos usado podremos intuir cómo funcionarán. El usuario se puede hacer un modelo mental de cómo funciona una cafetera que ve por primera vez porque ya ha usado otras similares. Más información aquí: http://design-toolkit.recursos.uoc.edu/es/modelo-mental/ .

Una vez hayamos estudiado los diferentes conceptos, tendremos que hacer pantallazos de la interfaz para poner ejemplos de dónde se aplican estos. Esto nos ayudará a entenderlos y a comprenderlos de forma que los tengamos presentes cuando tengamos que hacer nuestras propias propuestas de diseño.

1.4. Conclusiones

En este primer reto hemos explorado los perfiles de usuario que usarán tu futura propuesta de interfaz. Hemos analizado la competencia para ver cómo lo hacen, y hemos aprendido lo que hacen bien y lo que tenemos que evitar. Además, hemos podido detectar una serie de puntos para mejorar que nos servirán para hacer propuestas de diseño que los mejoren. Finalmente, hemos podido explorar los conceptos fundamentales de la interacción persona-ordenador para tenerlos presentes en nuestras propuestas de diseño de interacción o de interfaces.

2. Reto 2: Definición: síntesis de la propuesta

En el primer reto, hemos creado unos perfiles de usuario para tu interfaz del caso de estudio. Hemos explorado la competencia, hemos visto si hay criterios que podemos mejorar y aspectos clave de una serie de interfaces que hemos analizado. Además, los *insights* o aspectos clave te han permitido ver qué elementos funcionan muy bien y aspectos que podemos mejorar. Finalmente, hemos aprendido los conceptos fundamentales de la interacción persona-ordenador: metáfora, *affordance*, visibilidad, restricción, retroacción y modelo mental.

Hecha la investigación, el siguiente paso será la definición. Por lo tanto, los objetivos que trabajaremos en este segundo reto son:

- 1) Definir el flujo de los acontecimientos principales para encontrar aspectos que podemos mejorar o clarificar.
- 2) Hacer un test de usuarios con posibles *stakeholders* implicados para ver necesidades.
- 3) Elaborar los primeros borradores de posibles soluciones.

El diagrama de flujo te permitirá analizar cosas como el número de clics y los pasos para hacer acciones determinadas. Esto nos permite ver si hay pasos repetitivos y si podemos reducir el número de pasos para completar una determinada tarea, por ejemplo.

El test con usuarios nos permitirá ver cómo usuarios de los perfiles que hemos definido usan la interfaz en primera persona. Podemos ver entre otras cosas en qué puntos hay problemas o qué funciona muy bien.

Para acabar, si sintetizamos los resultados de estos dos primeros objetivos podemos empezar a dibujar a mano alzada una posible distribución de pantallas y estilos de interacción.

2.1. Diagrama de flujo

El diagrama de flujo es un método que nos permite estructurar la secuencia lógica de acciones para completar una tarea dada.

Es un método muy útil para ver el tiempo que tardamos en hacer una tarea, número de clics que tenemos que hacer, pasos que hacemos para completar una acción y pantallas por las que pasamos. Nos permite comunicar el proceso que se tiene que seguir para completar una tarea. Finalmente, también nos puede ser útil para detectar posibles errores dentro del flujo de nuestra interfaz, si llegamos a lugares donde los usuarios no saben cómo continuar, etc.

Enlace recomendado

Podéis encontrar la explicación completa sobre el método del diagrama de flujo en la guía pertinente del *design toolkit*:

<http://design-toolkit.recursos.uoc.edu/es/guia/diagrama-de-flujo/>.

En un diagrama de flujo utilizamos básicamente cuatro elementos:

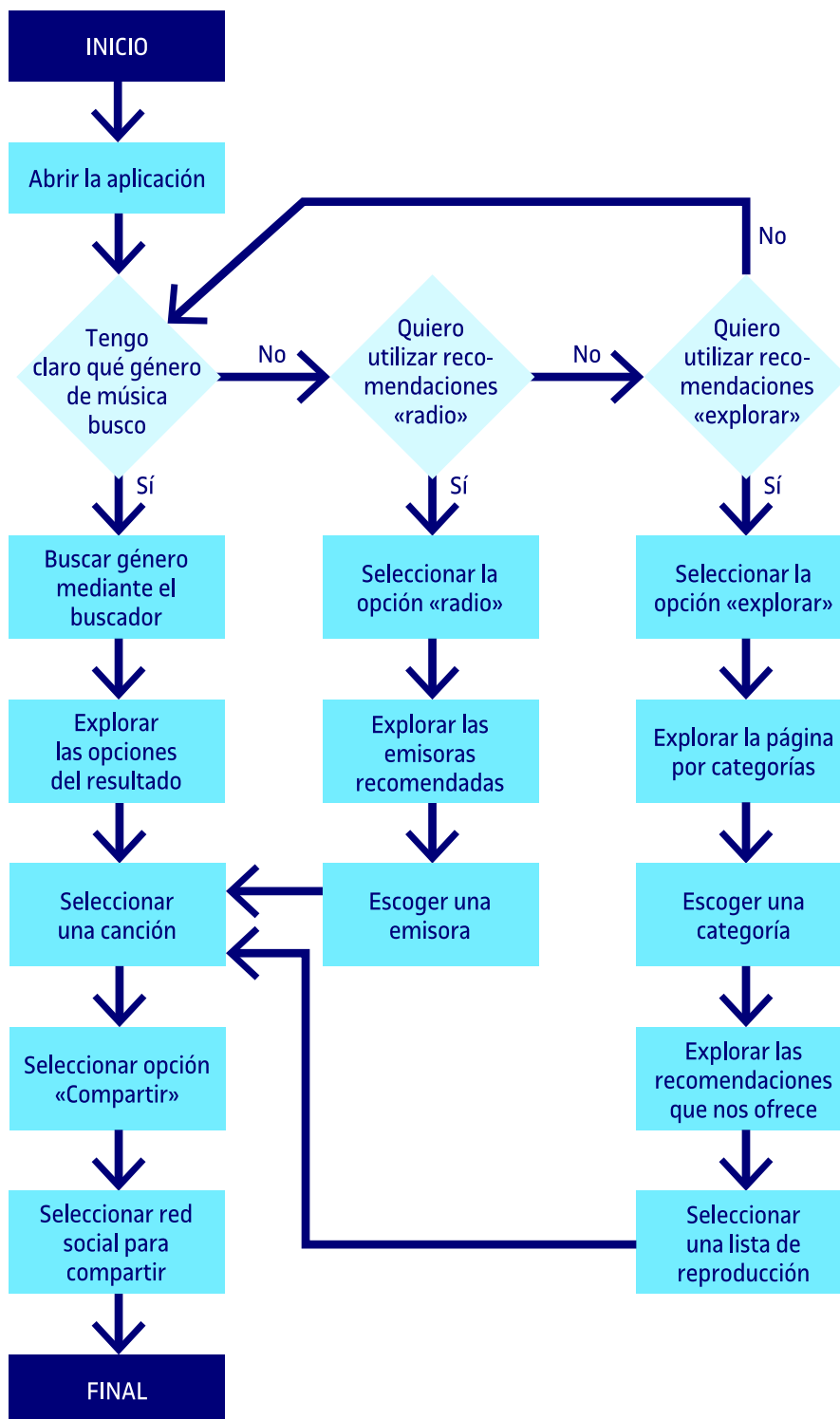
- unos círculos para denotar el inicio y el final del flujo,
- unos cuadrados o rectángulos para definir acciones sencillas,
- rombos para hacer preguntas de «sí» o «no», tipos: ¿es correcta la contraseña?, y definir qué acción tenemos que hacer si es correcta y qué acción tenemos que hacer si no lo es;
- y finalmente las flechas para relacionar estos elementos.

Siguiendo con el caso de estudio de ejemplo del cuaderno, imagina que la tarea fuera:

«compartir la canción 'Yellow' de Coldplay en la red social Facebook».

El diagrama de flujo que construiremos sería como el que tenemos a continuación (ver figura 6):

Figura 6. Diagrama de flujo de ejemplo sobre la tarea: compartir una canción en las redes sociales



En este flujo, empezamos abriendo la interfaz, elegimos el género, exploramos las opciones y seleccionamos la canción «Yellow» de Coldplay, seleccionamos «compartir», y «Facebook» para publicarla y acabamos el flujo con el recuadro rojo de fin. No obstante, si en vez de la canción «Yellow» fuera una canción cualquiera, vemos que tenemos otras formas de compartir, bien por la parte de radio o bien por la parte de explorar de la interfaz dada. Por lo tanto, podemos

ampliar el flujo principal con estas. Esto es lo que conocemos como ramas o caminos alternativos al flujo principal. Con este trabajo podemos ver qué aspectos se pueden trabajar, simplificar, agrupar y mejorar.

Analizando este diagrama de flujo vemos que el camino más corto para hacerlo son seis pasos, si queremos que compartir sea una cosa superrápida, nos tendríamos que preguntar: ¿Puedo acortar estos pasos de alguna forma? Por ejemplo, podemos poner la opción de compartir directamente junto a cada resultado de nuestra búsqueda «Yellow», de forma que no sea necesario entrar en la canción concreta.

2.2. Test con usuarios

Con el diagrama de flujo, nos hemos puesto el sombrero de usuario y hemos podido analizar de primera mano el paso a paso. Como diseñadores, hemos analizado lo que está pasando para hacer una acción. Ahora, con el test de usuarios pasaremos a ser observadores de los usuarios usando la interfaz para tomar nota y detectar puntos que funcionan y puntos que no. Siempre buscando qué podemos mejorar.

2.2.1. Definición del alcance de la evaluación

El primer punto que trabajaremos será definir cuál será el alcance de la evaluación. ¿Cuál es el objetivo? En este caso el objetivo será evaluar elementos relacionados con los cinco requisitos del cliente para encontrar aspectos que podamos mejorar:

- 1) Poder crear listas de canciones.
- 2) Organizar las canciones en categorías.
- 3) Tiene que haber un buscador que permita buscar por tipo de música, categoría o palabras clave.
- 4) El usuario tiene que poder compartir su lista de música.
- 5) El usuario tiene que poder ver recomendaciones de canciones similares a las que está escuchando.

Por lo tanto, lo primero que hay que hacer es concretar qué aspectos de la lista trabajaremos (esto puede venir dado por el enunciado de la práctica).

Tenemos que considerar también los resultados del *benchmarking*, puesto que si detectamos aspectos para mejorar relacionados con la parte de compartir entonces centraremos el test para trabajar bien esta parte y ver qué podemos mejorar.

Finalmente, también tendremos que decidir si la evaluación es presencial o remota, puesto que los mecanismos serán ligeramente diferentes. En caso de elegir la parte remota, tenemos que encontrar la forma de ver al usuario, puesto que podemos aprender mucho de sus reacciones, lenguaje corporal y la pantalla que usa. Para ver, entre otras, en qué partes duda o dónde deja el ratón.

2.2.2. Definición del perfil de los participantes y captación

El segundo punto es la definición del perfil de los participantes. En esta práctica lo hemos hecho en la primera parte. Por lo tanto, en este punto del test de usuario listaremos a qué personas de nuestro entorno podemos captar para que entren dentro de los perfiles de usuario que hemos elegido. También tendremos que definir el número de personas de cada perfil que captaremos para el test de usuario, normalmente intentaremos que haya el mismo número de personas de cada perfil.

En los test de usuario es importante trabajar la cuestión de género y seleccionar una representación equilibrada de personas para los diferentes géneros, de forma que evitemos, por ejemplo, tener solo hombres de un perfil dado.

Recordemos que es un momento excelente para enriquecer nuestros perfiles de usuario. Por lo tanto, recoged la información de los usuarios que entrevistemos siguiendo los mismos campos que tiene nuestro perfil de usuario. Y, posteriormente, iteraremos los perfiles con la información proporcionada para enriquecerlos si fuera necesario.

2.2.3. Definición del guion de las sesiones

Captar personas para hacer los test y conseguir su tiempo es complicado. Por eso, se tiene que planificar muy bien la sesión para no perder la oportunidad que hemos conseguido. Por lo tanto, en este apartado tenemos que pensar muy bien cómo preparamos los siguientes apartados:

- **Presentación.** En este punto presentamos cómo los evaluaremos, qué dispositivo usarán para probar la interfaz y cuál es la finalidad del estudio.
- **Cuestionario demográfico.** Se tiene que recoger siempre la información demográfica de los participantes, para contrastar que la captación se haya hecho correctamente y sea un usuario que encaja con los perfiles. Además, siempre nos puede servir para enriquecer las tablas del perfil de usuario. Finalmente, también se usará por temas estadísticos. Aquí puedes preguntar temas referentes a la edad, el género, la ocupación, el nivel de experiencia con la tecnología y alguna pregunta más que podamos necesitar.
- **Formulario de consentimiento.** Relacionado con la ley de protección de datos, tenemos que comunicar al usuario qué datos utilizaremos, cómo los utilizaremos y hasta cuando los tendremos. Además, tenemos que infor-

marlo de si pensamos grabar la sesión y nos lo tiene que autorizar. De no hacerlo, no podremos grabar su sesión.

- **Tareas del test.** Aquí presentaremos al usuario las tareas que queremos que haga con la interfaz. Siguiendo el caso de estudio presentado en este cuaderno, podríamos hacer que el usuario buscase una canción concreta de Coldplay y la compartiera en una red social como puede ser Facebook (mismo ejemplo que hemos hecho en el diagrama de flujo).
- **Cuestionario final.** Una vez el usuario haya hecho las tareas y haya acabado, podemos hacerle una pequeña entrevista. Es importante pensar qué preguntas podemos usar. Enfoquemos siempre estos temas concretos que queremos mejorar. Por ejemplo, si hemos notado que la parte de comparación no acaba de funcionar, podemos centrar parte de las preguntas en este tema.
- **Retribución.** En casos reales se recompensaría el tiempo del usuario con algún tipo de obsequio u otras opciones.

2.2.4. Realización de las sesiones

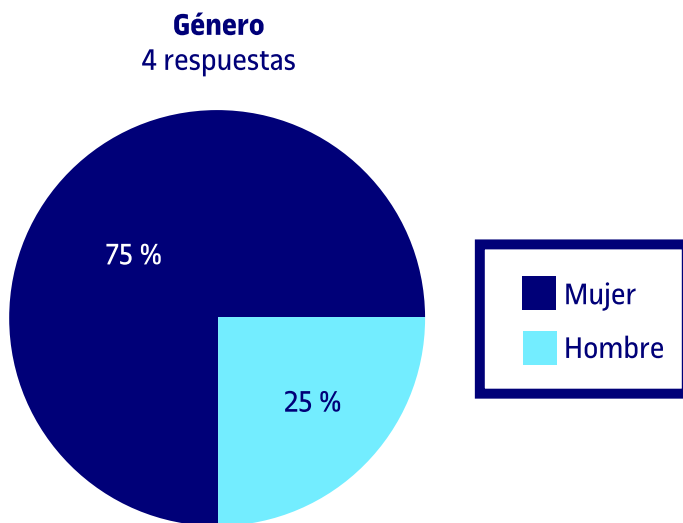
En la medida de lo que sea posible, intentaremos no intervenir una vez el usuario empieza a hacer las tareas que le hemos pedido. De esta forma intentaremos que sea lo más natural posible. Pero, sí que podemos observar al usuario, tomar notas y pedir que vaya verbalizando los pensamientos que va teniendo en voz alta. Si vemos que hace algo que no esperamos, también podríamos hacer preguntas para que nos explique por qué ha procedido así. Si nos ha dado autorización, también podemos grabar las sesiones para poder repasarlas después.

2.2.5. Análisis de resultados y elaboración de las conclusiones

Durante las sesiones, recogeremos dos tipos de datos: por un lado, tendremos datos cuantitativos como la edad, número de tareas completadas con éxito, tiempo que ha tardado en hacerlo, número de clics o número de errores que ha cometido; y por el otro lado, tenemos los datos cualitativos: anotaciones, comentarios, problemas que hemos observado, respuestas abiertas en el test, entre otros.

Ejemplo de dato cuantitativo (ver Figura 7):

Figura 7. Ejemplo de representación de un dato cuantitativo como el género del entrevistado



Ejemplo de datos cualitativos:

Participante número 1

- Se observa que la persona tiene habilidades tecnológicas limitadas.
- No valora positivamente tanta información a la vez en una misma pantalla. Aunque ve positiva la cantidad de información que puede encontrar, puede confundir tanto volumen en una misma pantalla.
- Le cuesta encontrar las opciones de añadir a la lista y compartir porque le cuesta encontrar estas opciones detrás del icono de los tres puntos. Lo valora negativamente. Esto nos demuestra la importancia de la parte más visual de la aplicación.

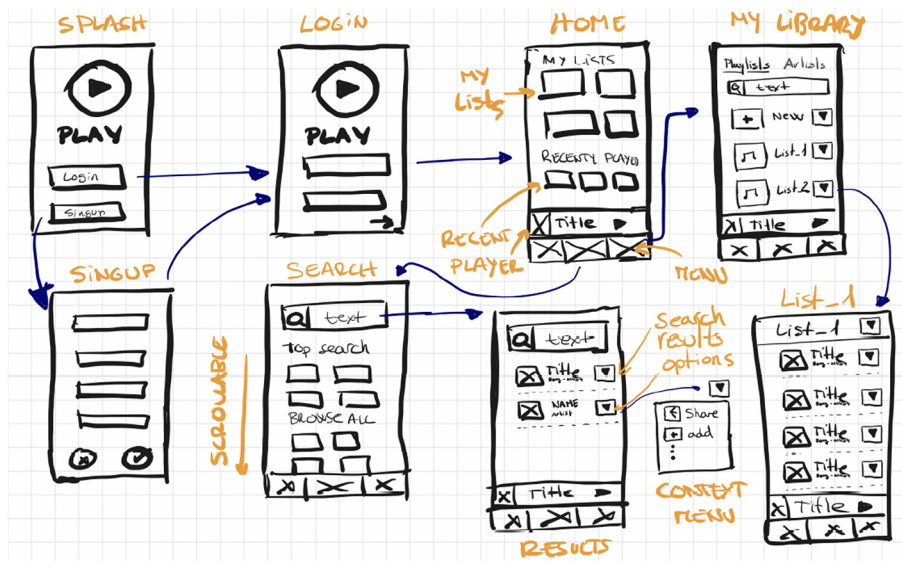
En este reto, hemos usado dos métodos diferentes para analizar la interfaz. El diagrama de flujo, como un método personal para analizar el paso a paso para realizar una tarea, y el test de usuario para ver cómo usan la interfaz. Con los datos recogidos utilizando los dos métodos ya tendríamos que poder definir dos o tres aspectos que queremos mejorar.

Una vez claros los aspectos que hay que mejorar, el siguiente paso será hacer propuestas de solución, tal como describimos en el siguiente apartado.

2.3. Propuesta de soluciones

Con los aspectos extraídos de los diagramas de flujo y del test de usuario queremos sintetizar diferentes **propuestas de mejora**. Para hacerlo podemos hacer directamente esbozos a mano alzada que nos permitan comunicar nuestras ideas, como veis en el siguiente dibujo (ver Figura 8):

Figura 8. Ejemplo de esbozo de las múltiples pantallas que puede contener una interfaz



Llegados aquí, es importante hacer un pequeño esbozo de las diferentes capturas y cómo queremos redistribuir los elementos, o qué elementos queremos añadir para hacer nuestra propuesta de mejora. Recordad aplicar aquí los conceptos fundamentales de la interacción persona-ordenador: metáforas, *affordance*, visibilidad, retroacción, restricción y el modelo mental. Además, tenéis que pensar qué estilos de interacción queréis aplicar.

Hay diversos **estilos de interacción**:

- indicadores de órdenes (típica consola de mandos),
- menús de selección o navegación,
- formularios,
- diálogo basado en lenguaje natural y
- manipulación directa.

No tengáis miedo y haced borradores, no tienen que ser cuidadosos, tienen que servir solo para comunicar cuántas páginas diferentes os imagináis, cómo van cambiando los elementos de la página, o bien qué elementos ponéis en según qué pantallas. Podéis probar diferentes combinaciones para compartirlas con el equipo de trabajo (o familiares y amigos), en vuestro caso, para ver si los entienden y lo ven igual.

Observación

Seguramente los estilos que más usaremos serán los menús de selección y los formularios, pero cada día es más fácil encontrar interfaces que trabajan con voz o manipulación directa.

2.4. Conclusiones

Así pues, en este segundo reto, hemos explorado el diagrama de flujo de una interfaz para ver el número de pasos que necesitamos para completar acciones. Esto nos ha permitido ver, entre otros aspectos, si necesitamos muchos pasos para completar acciones. A partir de los aspectos que detectamos en el *benchmarking* del primer reto, y de aspectos que hemos podido detectar en el diagrama de flujo, hemos trabajado un test de usuario enfocado a ver cómo perciben los usuarios los aspectos que queremos mejorar, y ver si es un problema para ellos. Finalmente, a partir de los resultados que hemos obtenido, hemos tenido claros qué aspectos queríamos mejorar y hemos hecho propuestas de cómo podíamos reestructurar las interfaces.

3. Reto 3: Diseño universal y accesibilidad (PEC 2)

En el primer reto, hemos definido los perfiles de usuario, hemos analizado la competencia y hemos explorado los principios fundamentales de la interacción persona-ordenador.

En el segundo reto, hemos usado dos métodos para analizar una interfaz. En el diagrama de flujo, hemos definido nosotros el paso a paso necesario para completar algunas tareas. En la segunda tarea, nos hemos servido de usuarios reales para ver el uso en directo y detectar posibles errores o cosas que se pueden mejorar.

En este tercer reto, los objetivos que trabajaremos son los siguientes:

- 1) Explorar qué es el diseño universal.
- 2) Entender qué es la accesibilidad y cómo la podemos adaptar a nuestro diseño.

3.1. Diseño universal

Ronald L. Mace acuñó el término *diseño universal* para referirse al tipo de diseño que cualquier persona puede entender y usar, independientemente de la edad, habilidad o estado. El diseño universal define siete principios que se tienen que cumplir para que una interfaz se considere universal:

Principio 1: igualdad de uso. El diseño de la interfaz lo tiene que poder usar cualquier persona independientemente de las capacidades que tenga. Tenemos que evitar segregar cualquier grupo de personas por sus restricciones personales o por las maneras de uso del dispositivo (ved figura 1).

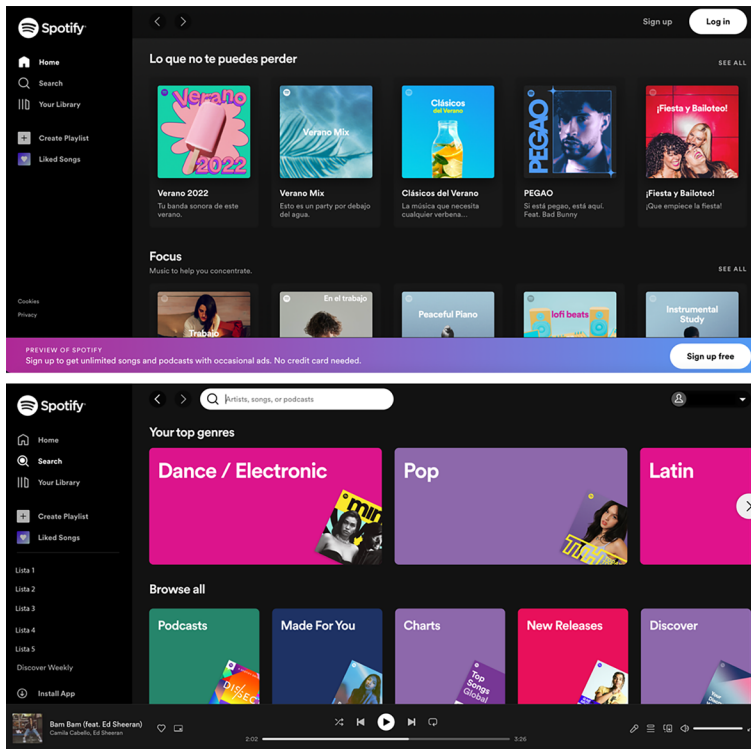
Pautas

- Que proporcione las mismas maneras de uso para todos los usuarios: idénticas cuando sea posible y equivalentes cuando no lo sea.
- Que evite segregar o estigmatizar a cualquier usuario.
- Las características de privacidad, garantía y seguridad tienen que estar igualmente disponibles para todos los usuarios.
- Que el diseño sea atractivo para todos los usuarios.

Enlace de interés

Más información sobre diseño universal en: <http://design-toolkit.uoc.edu/es/accesibilidad/>

Figura 1. Ejemplo del principio de igualdad de uso en Spotify



Misma ubicación de elementos tanto de contenido como de navegación y tanto para los usuarios que han efectuado el Login como para los que no.

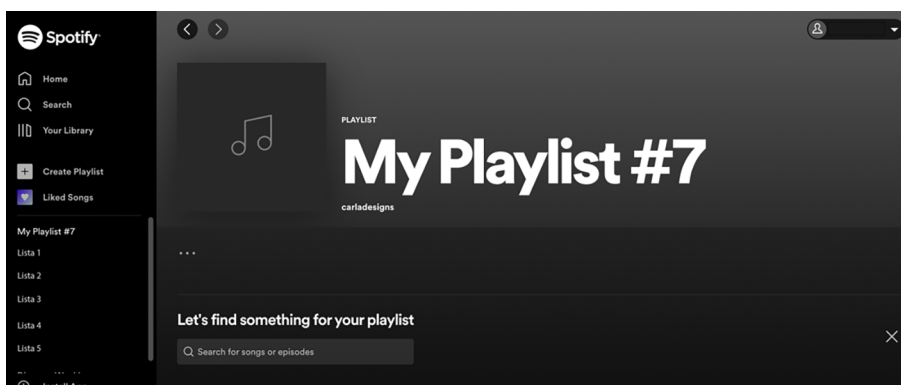
Principio 2: flexibilidad. El diseño de la interfaz se tiene que poder adaptar a un amplio rango de preferencias y habilidades.

Tenemos que asegurarnos de proporcionar diferentes opciones en los métodos de uso y facilitar la adaptabilidad de acceso al usuario (ved figura 2).

Pautas

- Que ofrezca posibilidades de elección en los métodos de uso.
- Que se pueda acceder y se pueda usar tanto con la mano derecha como con la izquierda.
- Que facilite la exactitud y la precisión al usuario.
- Que se adapte al ritmo de uso del usuario.

Figura 2. Ejemplo del principio de flexibilidad en Spotify



Creación de listas de reproducción para adaptar el acceso a las canciones, según las preferencias del usuario.

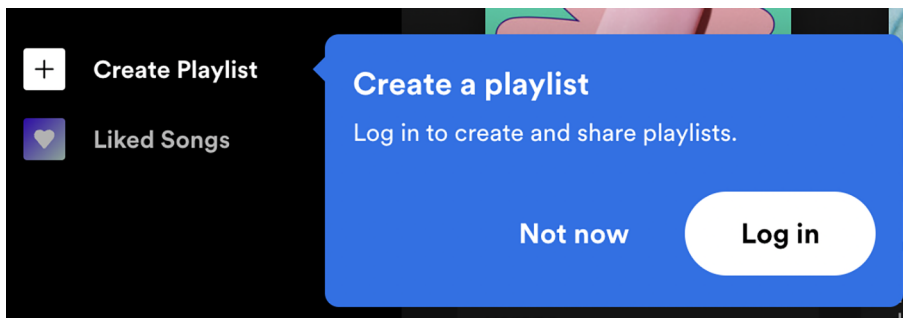
Principio 3: simple e intuitivo. El diseño de la interfaz tiene que ser fácil de comprender independientemente de la experiencia, los conocimientos, las habilidades o el nivel de concentración de los usuarios.

Se tiene que conseguir que los diseños sean intuitivos para el usuario y, cuanto más simple sea el diseño, más probabilidades habrá de que los usuarios logren su objetivo (ved figura 3).

Pautas

- Que elimine la complejidad innecesaria.
- Que sea consistente con las expectativas e intuición del usuario.
- Que se adapte a un amplio rango de habilidades lingüísticas.
- Que presente la información de manera consistente en relación con su importancia.
- Que proporcione avisos eficaces y métodos de respuesta durante y después de la finalización de cada tarea.

Figura 3. Ejemplo del principio de diseño simple e intuitivo en Spotify



Proporción de instrucciones para la elaboración de determinadas acciones a fin de que el usuario conozca el uso de la herramienta y sus posibilidades, y así pueda lograr sus objetivos.

Principio 4: información fácil de percibir. El diseño de la interfaz tiene que permitir la interacción/comunicación con el usuario, independientemente de las condiciones ambientales o de las capacidades sensoriales que tenga.

Este principio está muy relacionado con el primero, de uso equitativo. Se tienen que usar diferentes maneras de presentar la información esencial para que sea fácil de entender.

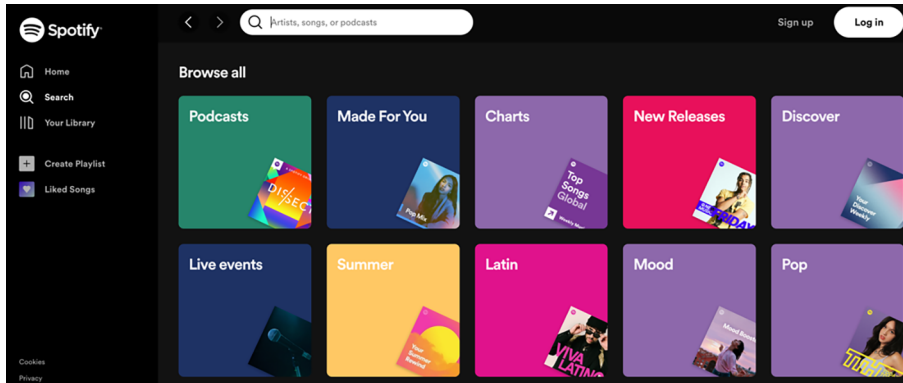
Además, nos tenemos que asegurar de que hay compatibilidad con las técnicas o dispositivos que usan las personas con limitaciones sensoriales, como por ejemplo las que tienen algún tipo de discapacidad visual (ved figura 4).

Pautas

- Que use diferentes métodos para presentar de manera redundante la información esencial (gráfica, verbal o táctilmente).
- Que proporcione suficiente contraste entre la información esencial y su alrededor.
- Que se amplíe la legibilidad de la información esencial.

- Que diferencie los elementos en formas que se puedan describir (por ejemplo, que sea sencillo proporcionar instrucciones o direcciones).
- Que proporcione compatibilidad con variedad de técnicas o dispositivos empleados por personas con limitaciones sensoriales.

Figura 4. Ejemplo del principio de información fácil de percibir en Spotify



Géneros musicales clasificados por tipología, con una fácil distinción entre sí.

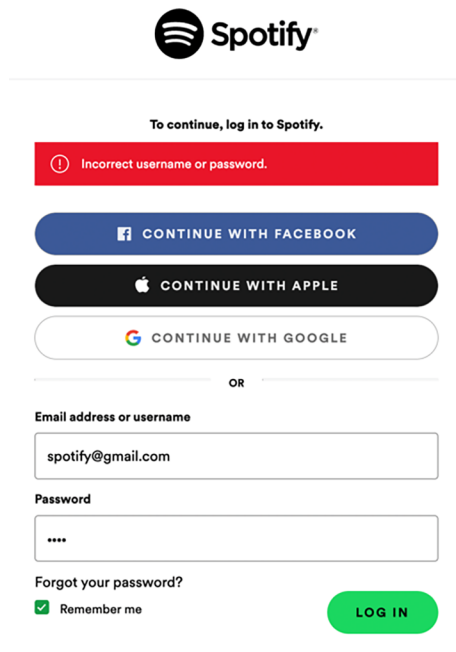
Principio 5: tolerante a errores. El diseño de la interfaz tiene que minimizar los peligros y las consecuencias de las acciones no intencionadas, como por ejemplo hacer clic accidentalmente en una zona de la pantalla.

Tenemos que organizar los elementos para minimizar estos errores y proporcionar advertencias en caso de que pasen, para que se pueda solucionar el problema sin fricciones (ved figura 5).

Pautas

- Que disponga de elementos para minimizar los riesgos y errores: visibilizar los elementos más utilizados y accesibles, y eliminar los elementos peligrosos que puedan conducir al error.
- Que proporcione advertencias sobre peligros y errores.
- Que proporcione características seguras de interrupción.
- Que reduzca acciones inconscientes en tareas que requieren vigilancia.

Figura 5. Ejemplo del principio de diseño tolerante a errores en Spotify



The image shows the Spotify login interface. At the top is the Spotify logo. Below it, a message says "To continue, log in to Spotify." A red error banner displays a warning icon and the text "Incorrect username or password." Below the banner are three social login buttons: "CONTINUE WITH FACEBOOK", "CONTINUE WITH APPLE", and "CONTINUE WITH GOOGLE". A horizontal line with "OR" in the center separates these from the standard login fields. The "Email address or username" field contains "spotify@gmail.com". The "Password" field is masked with four dots. Below the password field is a link for "Forgot your password?". At the bottom left is a "Remember me" checkbox, which is checked. At the bottom right is a green "LOG IN" button.

Indicadores de error a la hora de introducir los valores que forman los campos de Login.

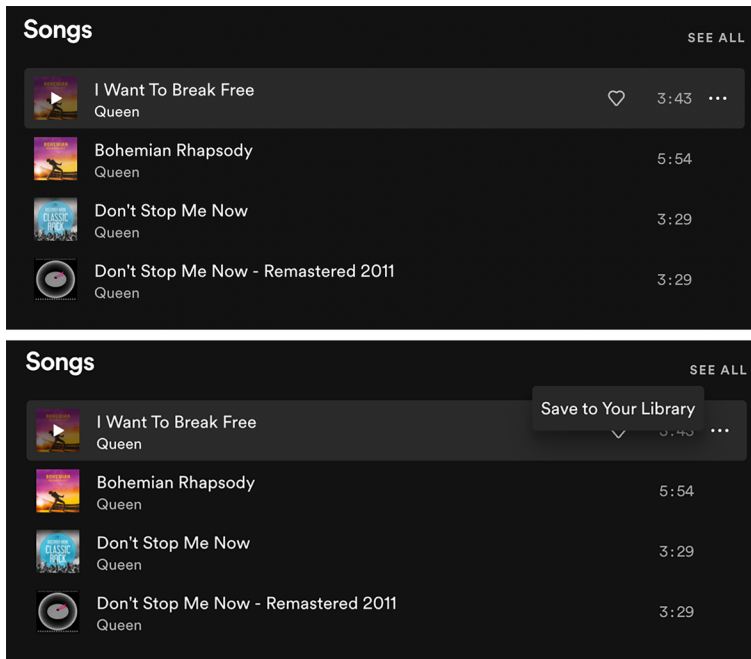
Principio 6: escaso esfuerzo físico. El diseño de la interfaz tiene que permitir el uso con tan poco esfuerzo como sea posible y que se pueda usar de manera eficiente y cómoda, sin fatigar al usuario.

Se tiene que tener en cuenta tanto el cansancio físico como el de carga cognitiva, así como la carga percibida en las tareas (ved figura 6).

Pautas

- Que permita que el usuario mantenga una posición corporal neutra.
- Que use de manera razonable las fuerzas necesarias para operar.
- Que minimice las acciones repetitivas.
- Que minimice el esfuerzo físico continuado.

Figura 6. Ejemplo del principio de escaso esfuerzo físico en Spotify



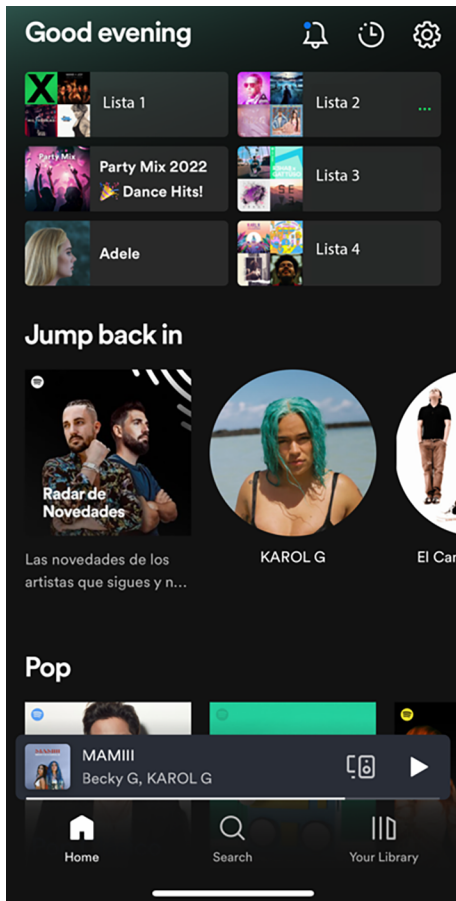
Posibilidad de guardar una canción de manera rápida en un solo clic, mediante la selección del icono del «corazón».

Principio 7: dimensiones apropiadas. El diseño de la interfaz tiene que tener unas medidas apropiadas y un espacio adecuado entre elementos, para que el usuario pueda usar el producto sin problemas, independientemente de las características o la movilidad que tenga (ved figura 7).

Pautas

- Que proporcione una línea de visión clara hacia los elementos importantes para cualquier contexto de uso del usuario.
- Que el acceso a cualquier funcionalidad y componente sea accesible para cualquier contexto de uso del usuario.
- Que se adapte a las variaciones de la medida de la mano y de sujeción.

Figura 7. Ejemplo del principio de dimensiones apropiadas en Spotify



Adaptación de los elementos de navegación y contenido para dispositivos móviles, para asegurar un uso correcto para las dimensiones y gesticulación necesaria del propio dispositivo.

Los siete principios del diseño universal nos ayudarán a crear mejores experiencias para todo el mundo, habrá que tener en cuenta la accesibilidad y la inclusividad, y los tendremos que tener presentes e incluirlos en todos los procesos de diseño.

Por lo tanto, el objetivo de una buena interfaz tendría que ser seguir estos principios y que cualquier persona la pudiera usar sin necesidad de adaptarla o crear un diseño especializado para usuarios concretos.

3.2. Accesibilidad

Todos estos principios tienen como objetivo conseguir que el rango más amplio posible de usuarios puedan usar una interfaz que les sea accesible.

Dentro del ámbito web, hay un consorcio que dicta las normativas actuales para trabajar el diseño universal y se conoce como **world wide web consortium** o **W3C**, abreviado. Este consorcio fue el encargado de definir un marco para guiar a cualquier desarrollador web en la creación de webs y de interfaces accesibles. Para conseguir este hito desarrollaron la *Guía de accesibilidad para el contenido web* o *Web content accessibility guidelines* (**WCAG**, en inglés).

Enlace de interés

Podéis encontrar toda la información punto por punto en su página web sobre la iniciativa de accesibilidad web: <https://www.w3.org/WAI/>

3.2.1. Estándares de la accesibilidad web

Los estándares de la accesibilidad web se centran en diferentes componentes que funcionan juntos, entre los cuales, están:

- **Contenido web.** Se refiere a cualquier parte de un sitio web, como por ejemplo texto, imágenes, formularios y multimedia, así como el código de marcaje, *scripts*, aplicaciones, entre otros.
- **Agentes de usuario.** Son las herramientas de software que las personas usan para acceder al contenido web, incluyen navegadores de escritorio, navegadores por voz, navegadores de teléfono móvil, reproductores multimedia o *plugins*.
- **Herramientas de autor.** Se refiere a los software o servicios que las personas usan para crear contenido web, incluyendo editores de código, herramientas de conversión de documentos, sistemas de gestión de contenido, blogs, *scripts* de base de datos y otras herramientas.

Enlace de interés

Podéis encontrar información sobre estrategias, estándares y materiales para la web accesible para personas con discapacidades en: <https://www.w3.org/WAI/fundamentals/accessibility-principles/es>

En este módulo nos focalizaremos en los estándares relacionados con el contenido web, que se agrupa alrededor de cuatro conceptos clave:

- 1) perceptible
- 2) operable
- 3) comprensible
- 4) robusto

Para cada uno de estos conceptos clave se plantean una serie de hitos que trabajar y que el diseñador tiene que tener en cuenta.

Perceptible

Cuando hablemos de *perceptible*, trabajaremos en temas como por ejemplo alternativas al texto (contenidos no textuales); como el hecho de trabajar con multimedia (audio y vídeos); formas de hacer el diseño adaptable, y cómo hacerlo distinguible (colores, contraste, medidas, etc.).

Dentro de este principio, encontramos los requisitos siguientes:

1) Alternativas de texto para contenido no textual. Son equivalentes para el contenido no textual.

Ejemplos

- Equivalentes cortos para imágenes, incluyendo iconos, botones y gráficos.
- Descripción de los datos representados en gráficos, diagramas e ilustraciones.
- Breves descripciones de contenido no textual como por ejemplo archivos de audio y vídeo.
- Etiquetas para controles de formulario, entrada de datos y otros componentes de la interfaz de usuario.

Las alternativas textuales transmiten el propósito o función de una imagen para proporcionar una experiencia de usuario equivalente. Por ejemplo, una alternativa textual apropiada para un botón de busca sería «buscar» en lugar de «lupa».

Las alternativas textuales se pueden representar de varias maneras. Por ejemplo, pueden ser leídas en voz alta por personas que no pueden ver la pantalla y por las que tienen dificultades de lectura, aplicadas a medida de texto personalizado o en dispositivos braille.

Las alternativas textuales sirven como etiquetas para controles y funcionalidades que ayudan a la navegación mediante el teclado o por reconocimiento de voz (entrada por voz). También actúan como etiquetas para identificar audios, vídeos y archivos en otros formatos, así como aplicaciones integradas como parte de un sitio web.

2) Subtítulos y otras alternativas para multimedia. Son alternativas para las personas que no pueden escuchar audio o ver vídeo.

Ejemplos

- Transcripciones textuales y subtítulos para contenido de audio, como grabaciones de una entrevista de radio.
- Audio descripciones, que son narraciones que describen los detalles más importantes, en un vídeo.
- Interpretación en el lenguaje de signos de contenido de audio, incluyendo experiencias auditivas relevantes.

Las transcripciones de texto bien escritas que contienen una correcta secuencia de información auditiva o visual proporcionan un nivel básico de accesibilidad y facilitan la producción de subtítulos y descripciones de audio.

3) Contenido que se puede presentar de diferentes maneras. Para que los usuarios sean capaces de cambiar la presentación del contenido, es necesario que:

- Los encabezamientos, listas, tablas, campos de formulario y estructuras de contenido usen el marcaje apropiado.
- El orden de la información o la secuencia de instrucciones sean independientes de la presentación.
- Los navegadores y tecnologías de apoyo faciliten configuraciones para personalizar la presentación.

El cumplimiento de este requisito permite que el contenido se lea correctamente en voz alta, se amplíe o se adapte para satisfacer necesidades y preferencias de diferentes personas. Por ejemplo, que se pueda presentar usando combinaciones de color personalizadas, medida de texto u otro estilo, para facilitar la lectura.

Este requisito también facilita otras maneras de adaptación, incluyendo la generación automática de esquemas de página y resúmenes para ayudar a obtener una visión general y a centrarse en partes concretas con más facilidad.

4) Contenido más fácil de ver y escuchar. El contenido distinguible es más fácil de ver y leer.

Ejemplos

- El color no se usa como única manera de transmitir información o identificar el contenido.
- Las combinaciones predeterminadas de color de primer plano y fondo proporcionan suficiente contraste.
- Cuando los usuarios redimensionen el texto hasta un 44 % o cambian el espaciado del texto, no se pierde información.
- Reflujo de texto en ventanas pequeñas (*viewpoints*) y en el momento cuando los usuarios hacen el texto más grande.
- Las imágenes de texto se redimensionan, se sustituyen por texto real o se evitan siempre que sea posible.
- Los usuarios pueden poner en pausa, parar o ajustar el volumen del audio que se reproduce en un sitio web.
- El audio de fondo es bajo o se puede silenciar para evitar interferencias o distracciones.

El cumplimiento de este requisito ayuda a separar el primer plano del fondo para hacer que la información importante sea más distinguible. Esto incluye consideraciones tanto para las personas que no usan tecnologías de apoyo como para las que sí y pueden percibir interferencias de contenido de audio o visual, del segundo plano.

Por ejemplo, muchas personas con daltonismo no usan ninguna herramienta en particular y confían en un diseño adecuado que proporcione suficiente contraste de color entre el texto y el fondo que lo rodea. Para otras personas, el audio que se reproduce automáticamente, podría interferir con el sistema de «texto a voz» o con dispositivos de ayuda auditiva (ALD).

Operable

Cuando hablemos de *operable*, trabajaremos temas como por ejemplo tipos de teclados, temas de tiempos de respuesta, de navegación o modos de interactuar (ratones, punteros, gestos, etc.).

Dentro de este principio, se encuentran los requisitos siguientes:

1) Funcionalidad disponible desde el teclado. Muchas personas no usan el ratón y dependen del teclado para interactuar con la web. Este hecho requiere acceso a todas las funciones mediante el teclado, incluyendo los controles de formulario, entrada de datos y otros componentes de interfaz de usuario.

Ejemplos

- Toda la funcionalidad que está disponible con el ratón, también lo está con el teclado.
- Los navegadores web, herramientas de autor y otras herramientas facilitan apoyo para el teclado.

El cumplimiento de este requisito ayuda a los usuarios de teclado, incluyendo a quienes usan teclados alternativos, como por ejemplo los de diseño ergonómico, teclados en pantalla o dispositivos de conmutación.

También ayuda a las personas que utilizan reconocimiento de voz (entrada por voz) para interactuar con sitios web y dictar texto mediante la interfaz del teclado.

2) Los usuarios tienen suficiente tiempo para leer y usar el contenido. Algunas personas necesitan más tiempo que otras para leer y usar el contenido. Por ejemplo, hay quien requiere más tiempo para escribir texto, comprender instrucciones o incluso completar tareas de un sitio web.

Ejemplos

Entre los ejemplos de cómo se puede proporcionar suficiente tiempo, se incluyen mecanismos como:

- Parar, alargar o ajustar los límites de tiempos, excepto cuando sean necesarios.
- Interrumpir, parar u ocultar movimiento o desplazamiento del contenido.
- Posponer o suprimir interrupciones, excepto cuando sean necesarias.
- Reautenticar cuando la sesión expire, sin la pérdida de datos.

3) El contenido no causa convulsiones ni reacciones físicas. El contenido que parpadea a ciertas velocidades o siguiendo determinados patrones puede causar reacciones fotosensibles, lo que incluye convulsiones. Lo ideal es evitar el contenido parpadeando o solo usarlo de forma que no cause ningún tipo de riesgo.

También las animaciones y el contenido en movimiento pueden provocar incomodidad y reacciones físicas.

Ejemplos

Algunos ejemplos de cómo evitar convulsiones o reacciones físicas son:

- No incluir contenido que parpadea a ciertas velocidades y siguiendo determinados patrones.
- Advertir a los usuarios antes de mostrar contenido parpadeando y proporcionar alternativas.
- Facilitar mecanismos para desconectar las animaciones, excepto cuando sean esenciales.

4) Los usuarios pueden navegar fácilmente, encontrar contenido y determinar en qué punto están. El contenido bien organizado ayuda a los usuarios a orientarse y navegar eficazmente.

Ejemplos

- Páginas que tienen títulos claros y están organizados usando cabeceras de sección de manera descriptiva.
- Hay más de un camino para encontrar las páginas relevantes en un conjunto de páginas web.
- Los usuarios son informados sobre en qué punto están dentro de un conjunto de páginas relacionadas.
- Hay maneras de evitar los bloques de contenido que se repiten en varias páginas.
- El foco del teclado es visible y el orden del foco sigue una secuencia significativa.
- El propósito de un enlace es evidente, incluso cuando se ve por sí mismo.

Cumplir con este requisito ayuda a las personas a navegar por las páginas web de diferentes maneras, dependiendo de las necesidades particulares y las preferencias.

Por ejemplo, mientras que algunas personas confían en estructuras jerárquicas de navegación, como las barras de menú para encontrar páginas web específicas, otros confían en las funciones de búsqueda de los sitios web.

Algunas personas pueden ver el contenido, mientras que otras lo pueden escuchar o incluso ver y escuchar a la vez. Algunas personas pueden usar el contenido solo con el ratón o teclado, mientras que otras se pueden servir de ambas opciones.

5) Los usuarios pueden utilizar diferentes modalidades de entrada, además del teclado. Las modalidades de entrada, más allá del teclado, como la activación táctil, el reconocimiento por voz (entrada por voz) y los gestos, hacen que el contenido sea más fácil de usar para muchas personas. A pesar de que no todas las entradas pueden utilizar cada una de estas modalidades o en la misma medida. Las consideraciones particulares del diseño pueden maximizar el beneficio de cada una.

Ejemplos

- Gestos que requieran destreza o movimientos hasta que tienen alternativas que ya no necesitan una habilidad avanzada.
- Componentes que están diseñados para evitar la activación accidental, por ejemplo, que facilitan la función de «deshacer».
- Etiquetas que corresponden con nombres en el código, lo que apoya a la activación por voz.
- Funcionalidad que se activa con movimiento o también mediante componentes de la interfaz del usuario.
- Botones, enlaces y otros componentes activos que tienen la medida suficiente para facilitar la activación por teclado.

Cumplir con este requisito hace que el contenido sea más fácil de usar para cualquier persona con un amplio rango de habilidades y dispositivos. Esto incluye el contenido empleado en teléfonos móviles, tabletas y terminales de autoservicio como las máquinas de tickets.

Comprensible

Cuando hablemos de *comprensible*, trabajaremos temas del lenguaje, como por ejemplo focalizar la atención del usuario, asistir en casos de error (mensajes, etiquetas, ayuda, etc.).

Dentro de este principio, están los requisitos siguientes:

1) El texto es legible y comprensible. Los autores de contenido tienen que asegurar que el contenido textual es legible y comprensible por la más amplia audiencia, incluso cuando es leído por herramientas de texto a voz.

Ejemplos

Algunos ejemplos de lo que se puede hacer:

- Identificar el idioma primario de la página web.
- Identificar el idioma de fragmentos de texto, frases u otras partes de la página web.
- Proporcionar definiciones por palabras no comunes, frases o abreviaturas inusuales.
- Usar el lenguaje más claro y simple posible o proporcionar versiones simplificadas.

El cumplimiento de este requisito ayuda al software, incluyendo las tecnologías de apoyo, a procesar correctamente el contenido del texto. Por ejemplo, este requisito ayuda al software a leer el contenido en voz alta, a elaborar resúmenes de páginas y proporcionar definiciones para palabras inusuales, como los argots técnicos.

También ayuda a las personas que tienen dificultad para entender oraciones más complejas, frases o vocabulario. En particular, ayuda a las personas con diferentes tipos de discapacidad cognitiva.

2) El contenido aparece y opera de manera predecible. Muchas personas confían en interfaces de usuario predecibles y se desorientan o se distraen con aportaciones o complementos no consistentes.

Ejemplos

Algunos ejemplos para hacer el contenido más predecible serían:

- Mecanismos de navegación que se repiten en múltiples páginas y que aparecen siempre en el mismo lugar.
- Componentes de la interfaz de usuario que se repiten en las páginas web y que tienen las mismas etiquetas.
- Cambios significativos en la página web que no se hacen efectivos sin el consentimiento del usuario.

El cumplimiento de este requisito ayuda a las personas a aprender rápidamente la funcionalidad y los mecanismos de navegación que se ofrecen en un sitio web, así como a emplearlos según sus necesidades y preferencias específicas.

Por ejemplo, algunas personas asignan teclas de acceso directo personalizadas a funciones que usan a menudo para mejorar la navegación mediante el teclado. Otros memorizan los pasos para llegar a determinadas páginas o para completar procesos en un sitio web. Ambos tipos de personas dependen de que la funcionalidad sea predecible y consistente.

3) Asistencia a los usuarios para prevenir y corregir errores. Los formularios y otros elementos de interacción pueden ser confusos o dificultar el uso para muchas personas, como resultado, serán más propensas a cometer errores.

Ejemplos

Algunos ejemplos de ayuda para prevenir y corregir errores serían:

- Instrucciones descriptivas, mensajes de error y sugerencias de corrección.
- Ayuda contextual para funcionalidades e interacciones más complejas.
- Opción de revisar, corregir o revertir errores si es necesario.

El cumplimiento de este requisito ayuda a las personas que no ven o no escuchan el contenido, o que quizás no reconocen las relaciones implícitas, las secuencias y otros indicios visuales.

También ayuda a las personas que no entienden la funcionalidad, están desorientadas o confusas, o que olvidan o cometen errores usando formularios y elementos de interacción, por cualquier razón.

Robusto

Cuando hablemos de *robusto* trabajaremos temas de compatibilidad. Un diseño robusto puede ser interpretado por un amplio número de agentes y asistentes tecnológicos.

El requisito de este principio es que **el contenido es compatible con las herramientas de usuario actuales y futuras**. El contenido robusto es compatible con diferentes navegadores, tecnologías de apoyo y otros agentes de usuario.

Ejemplos

Entre los ejemplos de cómo se puede conseguir, se incluyen:

- Asegurarse de que el marcaje puede interpretarse de manera confiable, por ejemplo, asegurando que es válido.
- Proporcionar un nombre, papel y valor para los componentes no estándares de la interfaz del usuario.

El cumplimiento de este requisito ayuda a maximizar la compatibilidad con los agentes de usuario actuales y futuros, incluyendo las tecnologías de apoyo.

En particular, permite que las tecnologías de apoyo puedan procesar el contenido de manera confiable y presentarlo de diferentes maneras. Se incluyen en este punto los botones no estándares, campos de formulario y otros controles.

3.3. Niveles de accesibilidad A, AA y AAA

Dentro de cada uno de los cuatro conceptos clave de accesibilidad, encontraremos que hay diferentes criterios descritos. Para cada criterio podemos comprobar si nuestro diseño los cumple o no. En la guía de aplicación, encontramos que hay tres niveles de conformidad de accesibilidad:

- 1) **Nivel A**. Es el nivel más bajo y lo obtendremos si nuestra interfaz cumple todos los criterios definidos en los cuatro conceptos clave de accesibilidad.
- 2) **Nivel AA**. Conseguiremos este nivel si cumplimos satisfactoriamente todos los criterios definidos como nivel A y todos los definidos como nivel AA.

Enlace de interés

Puedes ampliar la información sobre este tema en:
<https://www.w3.org/TR/WCAG21/>

3) Nivel AAA. Es el nivel más alto y solo se obtendrá si cumplimos satisfactoriamente todos los criterios definidos como A, AA y AAA. No obstante, no se recomienda aplicar siempre este alto nivel, puesto que para algunos contenidos es muy difícil satisfacer todos los criterios AAA.

Una vez la parte teórica ha quedado clara, analizaremos la interfaz que queremos mejorar y veremos qué criterios satisface de uno de los conceptos clave. Dentro del enunciado de la práctica te pondremos qué concepto o conceptos clave de los cuatro posibles tienes que trabajar.

Ejemplo

Si nos piden trabajar el concepto fundamental de *accesibilidad perceptible*, iremos a la página web <https://www.w3.org/tr/wcag21/> y dentro de *perceivable* (*perceptible* en inglés) miraremos todos los criterios (*level A*) definidos. En este caso, tendremos un total de nueve: *non-text content*, *audio-only and video-only*, *captions*, *audio description or media alternative*, *info and relationships*, *meaningful sequence*, *sensory characteristics*, *use of color* y *audio control*. Para cada uno de ellos tendremos que mirar si se cumplen o no los criterios siguiendo las explicaciones que encontraréis.

Imaginemos que queremos ver si satisfacemos el criterio *non-text content*. Tendríamos que mirar si todo el contenido que no sea textual tiene un texto alternativo que sirve para un propósito equivalente. Esto se puede mirar dentro del código HTML si hay la etiqueta: `<img alt="text">` con el texto descriptivo de la imagen. Este tipo de texto lo leen los software para personas ciegas, por ejemplo, y así pueden hacerse una imagen mental de lo que hay en la fotografía, el vídeo o el audio.

Hacerlo manualmente para una propuesta de diseño es mucho trabajo, pero podemos usar herramientas para llevar a cabo la evaluación de la accesibilidad web de manera automatizada.

3.3.1. Herramientas para la evaluación de la accesibilidad web

Hay una gran cantidad de herramientas relacionadas con la evaluación automatizada de la accesibilidad con diferentes propósitos y tecnologías de evaluación. A continuación se exponen algunos ejemplos.

1) Navegadores. A continuación se presentan algunas extensiones y *plugins* que se focalizan en la evaluación de la accesibilidad.

- **Firefox:** WAVE, web accessibility testing, equal access accessibility checker
- **Edge:** accessibility insights for web, devTools-web accessibility testing, eye-able-ayuda visual
- **Chrome:** LERA, siteimprove accessibility checker, ARCO toolkit, accessible web helper

Enlace de interés

Puedes consultar más herramientas para la evaluación de la accesibilidad web en: <http://accesibilidadweb.dlsi.ua.es/?menu=herramientas> y <https://www.w3.org/WAI/ER/tools/>

2) Sitios web. A continuación se presentan algunos sitios web que permiten hacer evaluaciones de accesibilidad web.

- TAW
- WAVE
- Run FAE

La evaluación de la accesibilidad web tiene por objetivo analizar, estudiar y validar diferentes aspectos de los sitios web, con el propósito de diseñar experiencias para diferentes tipos de usuario y verificar las pautas y directrices establecidas por la W3C.

Como hemos visto, la evaluación de la accesibilidad web se puede hacer de manera automática o manual. Por eso se usan varias herramientas de revisión, para obtener una evaluación completa y fiable.

Ahora que tenemos clara la existencia de herramientas que permiten la evaluación de la accesibilidad web de manera automatizada, sigue las **instrucciones del tablón** para completar la evaluación de la accesibilidad propuesta en este reto.

Una vez hecha la evaluación, teniendo en cuenta los conceptos clave de accesibilidad, haz una revisión de la propuesta que has dibujado en el reto 2 y haz anotaciones sobre qué mejoras tendrías que aplicar para cumplir el nivel A de accesibilidad. Puedes volver a dibujar la propuesta que habías hecho para el reto 2.

3.4. Conclusiones

En este tercer reto hemos aprendido conceptos fundamentales para hacer buenos diseños a fin de que cualquier persona pueda entender y usar, independientemente de la edad, habilidad o estado. Hemos aprendido que hay cuatro conceptos clave para trabajar la accesibilidad que agrupan todos los principios del diseño universal. Hemos visto que hay tres niveles diferentes de accesibilidad y ahora sabemos cuáles son los imprescindibles para poder tener un diseño accesible de nivel A.

4. Reto 4: Generación: implementación de un prototipo de baja fidelidad

En el primer reto, hemos definido los perfiles de usuario, hemos analizado la competencia y hemos explorado los principios fundamentales de la interacción persona-ordenador.

En el segundo reto, hemos utilizado dos métodos para analizar una interfaz. En el diagrama de flujo hemos definido nosotros el paso a paso necesario para completar algunas tareas. En la segunda, hemos utilizado usuarios reales para ver el uso en directo y detectar posibles errores o cosas para mejorar.

En el tercer reto, hemos trabajado el diseño universal, la accesibilidad y hemos refinado la propuesta de mejora para cumplir el nivel A de la WCAG.

Este cuarto reto dentro de la etapa de generación tiene los objetivos siguientes:

- 1) La generación de un prototipo de alto nivel
- 2) Dibujar un *user journey* siguiendo la aproximación del *happy path*
- 3) Comprobar cómo estamos aplicando los conceptos fundamentales del IPO

Partiremos de dibujar un tipo concreto de *user journey* partiendo de los borradores que hicimos en el reto 2 e iteramos en el reto 3. A partir de este *user journey*, prototiparemos los elementos necesarios para comunicar o enseñar nuestra propuesta. Finalmente, repasaremos cómo estamos aplicando los conceptos fundamentales del IPO: metáfora, *affordance*, restricción, visibilidad, retroalimentación y modelos mentales.

4.1. *User journey* con *happy path*

Desarrollar un *mockup* o prototipo de toda la interfaz puede ser muy costoso. Por lo tanto, dentro de este cuaderno desarrollaremos solo el espacio necesario para cumplir un par de tareas.

Siguiendo el ejemplo musical que estamos trabajando, imaginad que queremos compartir un *user journey* y solo queremos prototipar esta parte para comunicar al cliente. El primer paso es definir los pasos que queremos hacer. Por lo tanto, cogemos nuestro perfil de usuario y simulamos que lo estamos usando mediante la elaboración de un *user journey*. Más concretamente, apli-

caremos un *happy path*, un tipo *user journey* que plantea solo el escenario exitoso, no hay que hacer registros, no hay errores, el paso a paso normal que haríamos con el cliente para enseñar el aspecto que hemos trabajado.

En el primer ejemplo que tenemos a continuación, la usuaria Yolanda coge el *smartphone* y entra en la aplicación, clicla el icono de buscar, busca «REM». Una vez tiene la lista de canciones, selecciona la canción y la compartimos en Facebook. Tan pronto como tenemos definidas las acciones, podemos tener claro cuántas pantallas tenemos que prototipar para cumplir esta tarea. Por lo tanto, podemos acompañar a cada paso con una pantalla (ver Figura 10).

Figura 10. Ejemplo de un *user journey* para la tarea de compartir una lista de canciones a otras personas

USER JOURNEY B (43 años)	TAREA: COMPARTIR UNA LISTA DE CANCIONES CON OTRAS PERSONAS			
	DESCUBRIMIENTO	CONSIDERACIÓN	USO	FIDELIZACIÓN
Objetivo	Compartir mi lista de canciones con mis amigos/as	Valorar las diferentes opciones que tengo para compartir una lista	Hacer pública mi lista con otras personas	Actualizar mi lista con nuevas canciones
Touchpoints	Seleccionar la lista de reproducción que quiero	Ver las diferentes formas de compartir mi lista	Compartir mi lista de canciones con mis amigos/as	Enriquecer mi lista con aportaciones de canciones por parte de otros
Acciones	1. Escoger de entre mis listas aquellas que quiero compartir.	1. Buscar dónde está la opción para poder compartir una lista. 2. Ver que tengo diferentes opciones para compartir: redes sociales, URL, enlace... 3. Valorar cuál es la mejor opción según a quien quiera compartir.	1. Entrar en la lista de canciones que quiero compartir. 2. Escoger una opción de compartir y hacerlo con una persona concreta. 3. Repetir esta acción con cada persona que quiera compartir.	1. Entrar en la lista de canciones que quiero. 2. Marcar la opción de lista colaborativa.

Fuente: ejemplo modificado de la entrega de la estudiante Montserrat Castrillo.

4.2. Herramientas de prototipaje

Podemos elaborar diferentes tipos de prototipado, de hecho, el que iniciamos en el reto 2 estaría dentro de la categoría de *sketching* (dibujo rápido para empezar a pensar y a colocar las cosas). Después, tenemos el prototipado de baja fidelidad (o *wireframe*), construido normalmente con las ideas extraídas de la *sketch* y, finalmente, se acabaría con un prototipado de alta fidelidad más parecido al producto final.

Una vez tenemos claro el paso a paso que tenemos que hacer, por qué pantallas pasamos y qué opciones seleccionamos (recordad que teníamos un borrador de esto en el reto 2 y que lo habíamos mejorado en el reto 3 para hacerlo accesible). Procedemos a prototipar las diferentes pantallas que hemos trabajado en el borrador.

Para elaborar el prototipo de baja fidelidad, podemos usar herramientas como Justinmind (ver Figura 11):

Enlace recomendado

Sobre prototipado, podéis obtener más información y referentes en <http://design-toolkit.recursos.uoc.edu/es/guia/prototipado/>.

Figura 11. Ejemplo de prototipo realizado con la herramienta Justinmind



En esta herramienta podemos definir diferentes *screens* (pantallas) y para cada pantalla poner en esta los elementos necesarios (texto, botones, etc.). Nos permite conectar las diferentes pantallas y simular el proceso que queremos probar / comunicar. Os dejo a continuación una pantalla que se ha definido para una propuesta de prototipo dentro del caso de estudio de este cuaderno. En la primera pantalla vemos el aspecto y en la segunda podemos ver las zonas de interacción. Los apartados que hemos programado para ir a otras pantallas dentro de Justinmind (ver Figuras 12 y 13).

Figura 12. Ejemplo de los apartados trabajados y de cómo Justinmind visualiza las partes interactivas

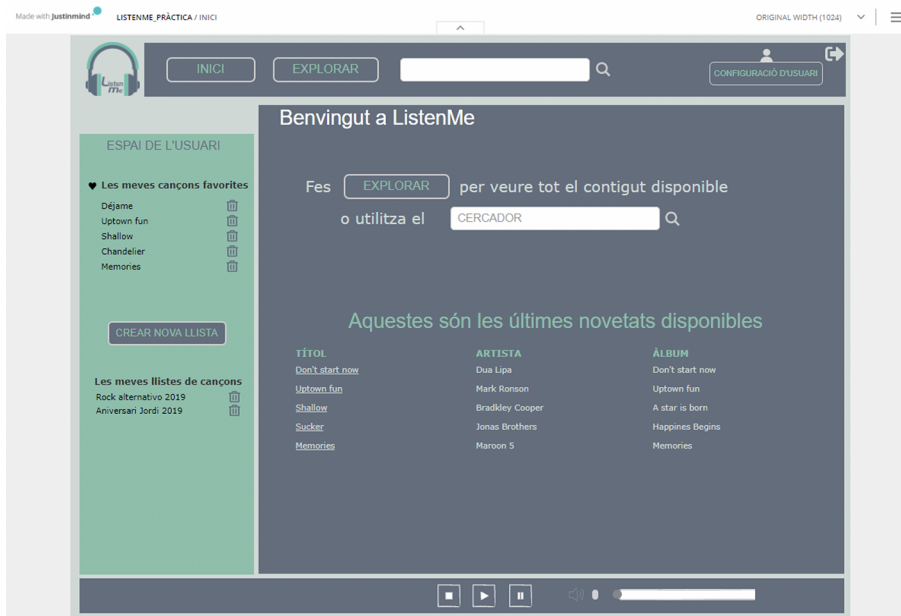
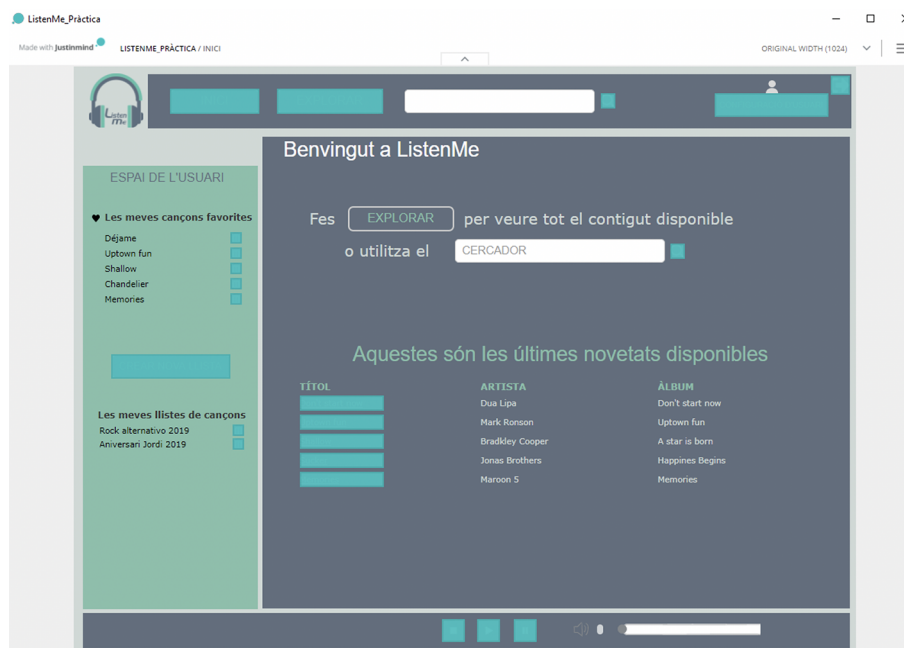


Figura 13. Ejemplo de los apartados trabajados y de cómo Justinmind visualiza las partes interactivas



4.3. Conceptos fundamentales del IPO

Una vez tenemos la primera iteración del prototipo es un buen momento para repasar cómo estamos usando los conceptos fundamentales de la interacción persona-ordenador dentro de nuestra interfaz. ¿Qué uso hacemos de la metáfora, visibilidad, restricción, retroacción, *affordance* y modelo mental?

También es un buen momento para entender qué es la arquitectura de la información.

De forma breve, entendemos por arquitectura de la información la disciplina que se dedica a organizar y etiquetar la información de nuestra interfaz. Por lo tanto, es una cosa que hemos trabajado sin saberlo dentro de los prototipados, puesto que hemos trabajado cómo organizar la información en las diferentes pantallas para que fuera accesible y útil para el usuario. Por lo tanto, es un buen momento para reflexionar si la navegación es adecuada, si los iconos que estamos usando son los adecuados, si los títulos de secciones están bien y son reconocibles.

Enlace recomendado

Podéis encontrar más información sobre qué es la arquitectura de la información aquí: <http://design-toolkit.uoc.edu/es/guia/arquitectura-de-la-informacion/>.

4.4. Conclusiones

En este cuarto reto, hemos aprendido a hacer un *user journey* para ver cuáles son las pantallas que usa el usuario. Es una buena herramienta para comunicar nuestro paso a paso dentro de la interfaz. Hemos aprendido a trabajar una herramienta para hacer prototipados de baja y de alta fidelidad generando un prototipo de baja fidelidad con las pantallas necesarias para poder evaluar posteriormente. Finalmente, hemos repasado cómo hemos integrado los conceptos fundamentales del IPO dentro de nuestro diseño. Y hemos hecho una pequeña introducción a la arquitectura de la información para aprender la importancia de los cimientos del diseño y tener una buena estructura de la información.

5. Reto 5: Evaluación: test y valoración del prototipo

En el primer reto, hemos definido los perfiles de usuario, hemos analizado la competencia y hemos explorado los principios fundamentales de la interacción persona-ordenador.

En el segundo reto, hemos utilizado dos métodos para analizar una interfaz. En el diagrama de flujo hemos definido nosotros el paso a paso necesario para completar algunas tareas. En la segunda, hemos utilizado usuarios reales para ver el uso en directo y detectar posibles errores o cosas para mejorar.

En el tercer reto, hemos trabajado el diseño universal, la accesibilidad, y hemos refinado la propuesta de mejora para cumplir el nivel A de la WCAG.

En el cuarto reto, hemos implementado un prototipo para completar unas tareas dadas.

Finalmente, en este quinto y último reto, los objetivos son:

- 1) Evaluar nuestro prototipo con usuarios.
- 2) Evaluar la accesibilidad de nuestro prototipo.
- 3) Elaborar un informe sobre el trabajo realizado.

Nuevamente, haremos un test de usuarios para evaluar nuestro prototipo y ver si los estos perciben mejoras. Aplicaremos las guías de la WCAG para saber si conseguimos el nivel A de accesibilidad. Y finalmente elaboraremos un informe sobre el trabajo realizado y los resultados de las evaluaciones, para ver si hemos conseguido una mejora de la funcionalidad. Detallaremos los aspectos que hemos mejorado y cómo los hemos mejorado. Y en el caso que no hayamos conseguido una mejora notable, detallaremos qué aspectos tendríamos que modificar para una teórica segunda iteración.

5.1. Test con usuario

Ha llegado el momento de hacer la evaluación de nuestra propuesta y ver si hemos podido mejorar algunos aspectos comparándolo con el primer test de usuarios que hicimos.

Es el momento para repasar el cuestionario, las tareas que pedimos y el guion del test de usuario. Podemos hacer pequeños cambios para corregir algunos errores que detectamos mientras lo confeccionábamos, como añadir o quitar

alguna pregunta que ahora no tiene sentido y repetir el proceso. Es muy importante centrarlo en los aspectos que hemos implementado en el prototipo. Recordemos trabajar datos tanto cualitativos como cuantitativos. Con los datos cuantitativos podemos mirar, por ejemplo, si hemos reducido el tiempo que necesitan los usuarios para completar las tareas. Con los datos cualitativos podemos anotar comentarios para ver, por ejemplo, si dudan menos a la hora de elegir las opciones.

5.2. Evaluar la accesibilidad de nuestra propuesta

¿Has podido aplicar los principios de accesibilidad? Es el momento de hacer la evaluación de accesibilidad de tu prototipo. Revisa los criterios y mira cuáles cumples y cuáles no. Es el momento de darle un nivel a tu prototipo, aprovecha para hacer anotaciones de cosas que hay que considerar para conseguir como mínimo el nivel A en próximas iteraciones del prototipado.

5.3. Elaboración del informe final

Es el momento de empaquetar todo el trabajo hecho en un documento. Explica brevemente el estado inicial, las mejoras prototipadas y si hemos conseguido o no mejorar la experiencia de usuario. Trabaja la información que obtengas para hacer una reflexión. Si no hemos conseguido una mejora sustancial, no sufras, explica lo que has conseguido y entonces reflexiona sobre qué cambios propones para la siguiente iteración.

5.3.1. Portada del documento

El título tiene que ser conciso y preciso en el punto que queremos comunicar. Por ejemplo, si finalmente hemos mejorado la parte de compartición en un sistema de música, pues podemos poner esto directamente:

«Estudio para la mejora de la compartición de canciones en una interfaz de reproducción de música en línea».

Si nos hemos enfocado más en proponer una nueva visualización de algunos aspectos para clarificar, podemos tener esto:

«Mejora de la visualización de los aspectos de compartición de una interfaz de reproducción de música en línea».

Podéis hacer propuestas para ver si se entiende el trabajo general que habéis hecho con el título a compañeros o con el mismo tutor.

La página del título llevará también el logotipo de la UOC en el centro con el nombre de la asignatura y el curso académico.

Por ejemplo: logotipo, Diseño de Interacción y Diseño de Interficies 2020-21, Semestre 1.

En la misma página, también incluiremos el nombre del estudiante y el consultor de la asignatura en la parte inferior derecha del documento.

5.3.2. Índice y resumen

La segunda página contendrá el índice del documento. Usaremos hasta un máximo de tres niveles de títulos. Aprovecho para comentar que nunca pondremos en el documento un título enganchado a un subtítulo. Siempre haremos como mínimo una explicación de qué contendrá aquella sección (ver Tabla 3).

Tabla 3

Incorrecto	Correcto
1. Introducción 1.1. Contexto 1.1.1. Visualización etc.	1. Introducción En este apartado presentaremos el contexto de actuación, la metodología seguida, etc. 1.1. Contexto Texto que describa como mínimo de qué hablaremos aquí. 1.1.1. Visualización

En la tercera página, haremos un breve resumen del documento que vendrá en dos idiomas: inglés y catalán o castellano, dependiendo de vuestra preferencia. El resto del documento lo podéis elaborar en el idioma que os sintáis más cómodos: catalán, castellano o inglés.

Dentro del resumen pondremos:

- Los objetivos que queremos trabajar de forma general.
- Una descripción de los problemas que hemos detectado.
- Una descripción de la propuesta de solución que hacemos.
- Un resumen de las conclusiones a las que hemos llegado.

5.3.3. Introducción

Dentro de la introducción presentaremos el caso de estudio del que partimos. Para los objetivos de la asignatura, podéis coger los objetivos de cada reto y explicarlos con vuestras palabras. Explicaremos por qué es importante trabajar en estas mejoras. Finalmente, acabaremos explicando cómo hemos estructurado el resto del documento.

5.3.4. Estructura

El documento tendrá tres capítulos principales:

1) El primero será el del contexto, donde describiremos la interfaz que hemos seleccionado para mejorar y los perfiles de usuario finales que hemos realizado. Aquí explicaremos el análisis que hemos hecho (podemos adaptar los textos de las entregas que ya hemos hecho) y finalmente qué problemáticas hemos encontrado al aplicar los diferentes métodos que hemos aprendido.

2) El segundo será la mejora propuesta. Aquí la describiremos ayudándonos de los diferentes prototipos que hemos implementado. Podéis usar el borrador para explicar la navegación general y las diferentes pantallas, y el prototipo de bajo nivel para enseñar cómo han quedado las pantallas que hemos trabajado para mejorar.

3) El tercer capítulo será la parte de análisis y comparación de resultados. Aquí podemos comparar el test que hicimos en el reto 2, con los resultados de analizar nuestro prototipo de bajo nivel del reto 5.

También aprovecharemos para incluir aquí cómo hemos utilizado los conceptos fundamentales del IPO en nuestra propuesta comentando qué metáforas usamos, la *affordance*, la visibilidad, la retroacción y si hemos aprovechado algún modelo meta. Para cerrar este tercer capítulo, comentaremos el nivel que hemos conseguido de accesibilidad, o qué aspectos tendrán que tener.

5.3.5. Conclusiones

Las conclusiones tienen que resumir lo que hemos conseguido respecto a los objetivos que planteamos al principio. Por lo tanto, podemos mirar el objetivo de la asignatura y qué hemos conseguido.

Entenderemos también como objetivos las propuestas de mejora que hacemos y hasta qué punto hemos conseguido completarlas. También es el punto para comentar las limitaciones que tiene nuestra propuesta y hacer las propuestas pertinentes para trabajar en el futuro.

5.3.6. Referencias

Hay diferentes formas de citar y de poner referencias: IEEE, APA, etc. Para este informe seguiremos estos criterios dependiendo del tipo de material que haya que citar:

Libros:

Autor (apellido, inicial del nombre), año de publicación, título del libro, lugar dónde se publicó y quién lo publicó.

Solo un capítulo del libro:

Autor (apellido, inicial del nombre), año de publicación, título del capítulo, título del libro, lugar dónde se publicó y quién lo publicó, páginas que comprende el capítulo.

Revistas:

Autor (apellido, inicial del nombre), año de publicación, título del artículo, título de la revista, número de la revista y páginas que comprende el artículo.

Webs:

Autor (apellido, inicial del nombre) [si sabemos quién ha hecho aquel material], fecha de la última actualización del material [si la encontramos], título del sitio web, URL, fecha que accedimos a esta.

5.4. Conclusiones

En este quinto y último reto, hemos evaluado nuestro prototipo para ver si hemos conseguido o no mejoras sustanciales. Mediante el test de usuarios hemos valorado cualitativa y cuantitativamente la mejora. Mediante las guías de accesibilidad del WCAG hemos podido valorar si estamos haciendo el diseño accesible o no, anotando las mejoras y los cambios para que no se pierdan en próximas iteraciones del prototipo. Y, finalmente, hemos hecho un documento como es debido para presentar nuestro trabajo.

6. Conclusiones. ¿Qué hemos aprendido?

Nos han dado un caso de estudio sobre un conjunto de interfaces y unos requerimientos para trabajar.

En el primer reto, hemos definido los perfiles de usuario, hemos analizado la competencia y hemos explorado los principios fundamentales de la interacción persona-ordenador. Esto nos ha servido para ponernos en la piel de los usuarios que usen la interfaz. Es muy importante tener claro para quién diseñamos. Hemos visto cómo lo hace la competencia. Es clave entender qué hace bien la gente para imitarlo y evitar los errores que ya han hecho los competidores. Y hemos aprendido los conceptos fundamentales de la interacción persona-ordenador. Hemos entendido la importancia de elegir metáforas adecuadas que puedan entender los usuarios, trabajar el *affordance* para que sea intuitivo de usar, trabajar la visibilidad para que los diferentes elementos sean perceptibles para el usuario, trabajar la restricción para evitar que los usuarios puedan cometer errores, y hemos pensado qué modelos mentales tienen nuestros usuarios para aprovechar lo que ya saben hacer.

En el segundo reto, hemos utilizado dos métodos para analizar una interfaz. En el primero, nosotros hemos hecho directamente el paso a paso de algunas tareas mediante un diagrama de flujo. Este tipo de método nos permite analizar los pasos que hace un usuario y ver por qué pantallas tiene que pasar para completar una acción. En el segundo método aplicado, hemos aprendido a trabajar con usuarios reales para ver sus puntos de vista, muchas veces nos han sorprendido haciendo cosas que no esperábamos, o aportando puntos de vista que no teníamos presentes. Esto nos ha permitido ver aspectos que podemos mejorar de la interfaz que seleccionamos en el primer reto. Nos ha permitido detectar una serie de temáticas que serían interesantes de analizar. Son los puntos para trabajar en los siguientes retos.

En el tercer reto, hemos trabajado el diseño universal, la accesibilidad y hemos refinado la propuesta de mejora para cumplir el nivel A de la WCAG. Este reto ha supuesto abrir el punto de visión, considerar aspectos que seguramente no estábamos teniendo en cuenta. Hemos entendido los cuatro conceptos clave de accesibilidad: perceptible, operable, comprensible y robustez. Hemos visto cómo funcionan las guías de accesibilidad de la WCAG, cómo define criterios para cada concepto clave de accesibilidad, y cómo tienes que trabajar tu diseño para conformar los niveles A, AA o AAA.

En el cuarto reto, hemos implementado un prototipo para completar unas tareas dadas. Hemos definido un *happy path* que tenemos que seguir para completar una acción. A partir de este *happy path*, hemos prototipado una serie de pantallas usando una herramienta de prototipaje como puede ser Justinmind

y hemos conseguido una primera propuesta de *wireframe*, un pequeño prototipo de baja fidelidad, que nos permite comunicar nuestras ideas al cliente y al equipo de trabajo.

Finalmente, en el quinto y último reto, hemos evaluado nuestro *wireframe* con usuarios reales y con las guías de la WCAG para tener resultados tangibles y compararlos con el estado inicial. Hemos conseguido un nivel de accesibilidad A o hemos definido los aspectos que tendremos que tener presentes en el desarrollo de las siguientes iteraciones del prototipo para cumplirlas. Y hemos elaborado un informe sobre el trabajo realizado para justificar los cambios que hemos hecho. Finalmente, hemos concluido este último reto con una reflexión crítica sobre el trabajo hecho.

