

Material Imprimible

Diseño UX/UI

Módulo 4

**Contenidos:**

- UX Writing
- Prototipado
- Principios heurísticos
- Evaluación heurística
- MVP
- Early adopters

## UX Writing

Uno de los elementos fundamentales a la hora de realizar nuestro diseño, teniendo en cuenta la experiencia de los usuarios, es la manera en la que nos vamos a comunicar con las personas. Por eso, es que nos vamos a centrar ahora en el *UX writing*, que es la práctica por la que determinaremos de qué manera le hablaremos a nuestros usuarios. Es decir, qué palabras utilizaremos y cómo las vamos a distribuir a lo largo de nuestro producto digital. Tengamos presente que se tratará de nuestra voz dentro del producto digital.

Para comprender de qué manera redactaremos nuestro contenido, pensemos un instante en nuestra experiencia cuando visitamos páginas web. ¿Nos detenemos en cada una de las palabras o buscamos directamente la información que necesitamos? Por ejemplo, si entramos en un sitio de compras porque necesitamos un producto específico, ¿leemos toda la página de inicio o nos fijamos rápidamente en qué sección se encuentra ese producto? Lo más habitual es que vayamos directamente a lo que estamos buscando.

Por lo tanto, como tenemos poco tiempo de atención de nuestros usuarios, es importante que podamos redactar de una manera clara lo que queremos comunicarles para que puedan sacarle el mayor provecho a nuestro producto y que se queden con nosotros. Como ya dijimos en varias oportunidades, si las personas entran a nuestro sitio y no encuentran la información que buscaban, lo van a abandonar porque no lo van a considerar útil.

Recordemos al “padre de la usabilidad”, Jakob Nielsen. A través de los estudios que realizó, detectó que los usuarios no leen en profundidad, sino que ojean de manera superficial las distintas páginas. Incluso, el desarrollador de software Steve Krug, en su libro *No me hagas pensar*, afirmó que “*los usuarios navegan a los tumbos, dándose golpes contra una y otra cosa hasta que encuentran lo que quieren*”.

Esto sucede porque las personas no quieren perder mucho tiempo buscando lo que necesitan en la página web que están visitando. Por el ritmo de vida actual, siempre están apurados. Y, como el avance de la tecnología y de los estudios que se realizaron a lo largo de los años, logró acelerar todos estos procesos, nos acostumbramos a tener la información de forma inmediata y ya no queremos dedicarle más tiempo que el necesario.

Además, descubrimos que no hace falta leer todo para cumplir con nuestro objetivo. Basta con que encontremos rápidamente algún botón que diga “buscar”, o “ingresar”, para saber qué tenemos que hacer sin recorrer todo el sitio.

Por este motivo es que tendremos que armar una estrategia para la redacción de nuestro contenido, de manera que podamos definir cuáles son las palabras que vamos a usar y cómo las vamos a distribuir a lo largo de la página para que los usuarios las encuentren rápidamente. Entonces, los discursos deben, indefectiblemente, ser concisos, objetivos y escaneables.

Cuando decimos que nuestro discurso debe ser conciso, nos referimos a que no tenemos que dar muchas vueltas para decir lo que queremos comunicar. Por el contrario, debemos ir directamente al grano. Luego, por objetivos nos referimos a que le debemos dar al usuario lo que necesita, que lo vamos a tener determinado por todas las instancias de investigación del proceso de UX. Y, por último, que sea escaneable implica que los usuarios puedan reconocer rápidamente cuáles son los puntos importantes para encontrar la información que están buscando.

Tengamos en cuenta que, a lo que estamos apuntando es a que nuestro discurso sea claro, conciso y útil, no sólo para que puedan encontrar la información, sino también para que puedan recuperarse de los errores que podrían llegar a cometer.

¿Cuántas veces nos pasó que nos equivocáramos al momento de escribir nuestra contraseña? Si la página no nos permitiera reponernos de ese error o no nos indicara el motivo por el cuál no estamos pudiendo ingresar a nuestra cuenta, no vamos a entender qué pasó. Por eso, es tan importante que seamos claros en la redacción y decir, por ejemplo, “contraseña incorrecta”, en lugar de simplemente indicar que “se produjo un error”, así como darnos la posibilidad de intentarlo de nuevo, o de recuperar la clave, como pueden ver en el ejemplo del material complementario.

Conozcamos, ahora, cómo podemos hacer para que nuestro contenido esté bien expresado en una página web. Si estamos volcando información muy extensa, es importante que las oraciones sean cortas y que se dividan en párrafos que no sean muy largos. De hecho, deberían tener, aproximadamente, 4 oraciones. De esta manera, lograremos que la lectura sea más ágil y fluida.

Pero, además, es fundamental que coloquemos subtítulos. Si nosotros estamos buscando información específica, con los subtítulos la vamos a encontrar mucho más

---

rápido, que si tuviéramos que leer todo el texto. Por eso, también otra posibilidad es que usemos negrita para resaltar palabras clave.

Por último, podemos acudir a la utilización de viñetas o enumeraciones. Así, logramos generar un corte en el bloque de texto. De esta manera, los usuarios seguramente se detengan en esos puntos y, si les interesa ese contenido, continuarán leyendo. De lo contrario, lo más probable es que lo pasen por alto.

Teniendo en cuenta todas estas pautas, será más fácil que cumplamos con el objetivo de que el contenido sea escaneable. Consideremos que, si la página está compuesta por demasiada información, con párrafos interminables y sin ningún corte, los usuarios difícilmente lean todo y, cuando revisen rápidamente si les sirve ese contenido, no van a encontrar lo que están buscando.

Esto está relacionado con el patrón de lectura que detectó Jakob Nielsen cuando realizó una prueba llamada *eyetracking study*, que se puede traducir como estudio de seguimiento ocular. Consiste en utilizar una luz especial que crea un reflejo en los ojos de la persona y permite determinar qué movimiento realiza con la mirada y en qué puntos se detiene.

Una de las conclusiones a las que llegó fue que, en las páginas de contenido informativo, las personas tienen un patrón de lectura en F. Es decir, en el mundo occidental, el escaneo se realiza leyendo de izquierda a derecha los primeros párrafos. Luego, se baja con la mirada hasta el siguiente punto que se lee, nuevamente, de forma horizontal cubriendo un área más corta que en la primera parte. Y, finalmente se continúa bajando la vista, realizando un movimiento vertical.

Este patrón de lectura, indica cómo los usuarios van buscando la información sin detenerse en todo el texto, sino en puntos clave. Por eso, como se presta mayor atención en la primera parte, es que es imprescindible que redactemos siguiendo la estructura de pirámide invertida, que implica poner la información más importante al principio, continuar en el cuerpo del texto con el contenido de apoyo y dejar los detalles para el final.

Ahora bien, Nielsen explica que el patrón en forma de F se utiliza para la lectura de las áreas de la página web que tengan mayor cantidad de contenido. Sin embargo, es necesario tener en cuenta otras posibilidades de redacción.

Veamos las que propuso:

- Por un lado, el patrón de torta de capas, que implica el escaneo de títulos y subtítulos, omitiendo el resto del contenido.
- Otra posibilidad es el patrón manchado. En este caso, directamente se omite el texto y se buscan enlaces, dígitos o conjuntos de palabras que estén escritas con un formato distintivo. Por ejemplo, cuando buscamos un domicilio o número de teléfono.
- Luego, el patrón de marcado, que se utiliza mayormente en dispositivos móviles, e implica mantener la mirada en un punto fijo mientras se desliza el dedo por la página.
- Con el patrón de omisión, las personas omiten las primeras palabras de una línea.
- Y, por último y menos frecuente, el patrón de compromiso, que implica la lectura de, prácticamente, todo el contenido de la página.

Por lo tanto, si tenemos en consideración todas las pautas que mencionamos anteriormente, es muy probable que logremos cumplir con el objetivo de que nuestro diseño de redacción sea escaneable para los distintos patrones de lectura.

## Prototipado

Un prototipo es un boceto de nuestro producto. De esta manera, podremos observar, de forma tangible, cuál sería el resultado final y nos permitirá hacer pruebas para evaluar si logramos cumplir con el objetivo que habíamos establecido en las etapas previas de la experiencia del usuario.

El beneficio principal que implica esta práctica, es que vamos a poder validar nuestras hipótesis antes de gastar tiempo y dinero en el desarrollo final. Sin embargo, recordemos que esto depende de la metodología que utilicemos, porque si nos estamos centrando en el modelo en cascada, no va a haber prototipo, directamente se va a realizar el producto cuando lleguemos al final del proceso. En cambio, las metodologías ágiles permiten que realicemos estos bocetos en cada una de las instancias de iteración, para ir observando los resultados parciales.

De esta manera, nos permite “fallar rápido y barato”, porque, como se imaginarán, el desarrollo del producto es la instancia más costosa de todo el proceso. Entonces, si lo realizamos y detectamos fallas, tendremos que volver a comenzar, perdiendo todo el tiempo y el dinero invertido en la creación.

Mientras que, la realización de prototipos, nos da la posibilidad de ir evaluando cada uno de los componentes de nuestro producto sin gastar mucha plata. Recordemos que una forma de prototipar es usando hojas de papel escritas a mano. Por lo tanto, el costo es prácticamente nulo y poder prevenir los distintos errores que puedan surgir nos va a resultar un ahorro económico muy grande, porque el producto va a llegar a la instancia de desarrollo listo para salir al mercado.

Además, implica nuestra primera aproximación al producto concreto. Ya no vamos a estar viendo los userflows o las arquitecturas de información que tenemos que interpretar para imaginar cuál será el resultado final. Con el boceto, tenemos la posibilidad de analizar los colores, las imágenes y los textos que formarán parte de nuestro diseño.

Cabe destacar que los prototipos están conformados de wireframes, que recordemos que son las representaciones visuales de la interfaz. Entonces, nos indican cuál será el contenido de nuestro producto, cómo se va a articular y en qué sectores se va a ubicar cada elemento. También nos muestra los botones que usaremos, qué tamaño tendrán las imágenes y qué secciones contendrá cada una de las pantallas.

Ahora bien, para realizar nuestro boceto, debemos atravesar diferentes instancias: comenzaremos por el sketch, luego el wireframe propiamente dicho, continuaremos con el mockup y finalizaremos en el develop.

El prototipo sketch se ubica al comienzo del proceso porque implica aquellos diseños realizados a mano. En este caso, no vamos a ahondar mucho en detalles, no es necesario que haya texto o colores, ni que determinemos qué formas tendrán los botones y las figuras. Lo que nos interesa en este punto es la estructura del diseño.

Por lo tanto, es la etapa más económica y rápida de realizar. Como nos estamos centrando en los elementos estructurales, nos sirve para volcar distintas ideas y evaluarlas para ver cuál resulta más acorde a nuestros objetivos. La única contra que tienen es que son estáticos. Es decir, no podemos apretar un botón y ver a dónde nos lleva. Igualmente, nos puede dar una idea aproximada si realizamos los dibujos de las distintas pantallas y,

---

cuando “seleccionamos” alguna opción en el papel, lo cambiamos por el de la pantalla a la que me estaría dirigiendo.

Una vez que realizamos el sketch, podemos avanzar hacia la elaboración del wireframe. En esta instancia, sí vamos a digitalizar nuestro boceto por lo que será mucho más interactivo y, por lo tanto, más funcional para evaluar cómo se podrá utilizar nuestro producto.

Estos bocetos se pueden realizar en fidelidad baja o media. Nos referimos a su contenido visual e informativo. Por eso es que hay tres tipos de fidelidades posibles: la visual, la de contenido y la de funcionalidad. Antes de continuar con la descripción del wireframe, veamos qué las diferencia.

La fidelidad visual refiere a los gráficos que componen nuestro prototipo. Mientras más baja sea la fidelidad, menor contenido de imágenes tendrá. Pero, a medida que nos vamos acercando a la fidelidad alta, más parecido será su diseño al de nuestro producto final. Luego, la fidelidad de contenido tiene que ver con las palabras escritas. Y, la fidelidad de funcionalidad hace referencia al nivel de interacción que tendrá o no nuestro boceto.

Muy bien, habiendo aclarado eso, si nuestro wireframe se realiza en fidelidad baja, será muy parecido al sketch: es decir, no tendrá mucho texto y se caracterizará por ser una representación visual de la estructura. En cambio, si se realiza en fidelidad media, ya nos mostrará botones con texto, íconos, estructuras más estandarizadas, y se pueden crear las interacciones o simplemente elaborar un userflow con wireframes, como vimos anteriormente.

Luego, la siguiente fase es la de mockup. En este caso ya será una representación fiel de la realidad, porque se realiza en alta fidelidad. Entonces, no sólo va a tener los botones y los textos, sino también los colores, las imágenes y un nivel de interacción completo.

Ahora bien, como se trata de la última instancia antes del proceso de desarrollo, será nuestra última oportunidad de realizar las modificaciones necesarias. Es por este motivo que necesitamos que el boceto quede idéntico a cómo queremos que sea nuestro producto en la realidad.

Finalmente, llegamos a la fase de develop, que significa desarrollo. Como su nombre lo indica, será el momento en el que los programadores harán las codificaciones en base al mockup que les entregamos. Por eso es fundamental que en esa instancia haya sido

completamente validada la interfaz, que sea funcional y con todo el diseño que necesitamos. Tengamos en cuenta que los desarrolladores no se ocupan de diseñar, por lo que van a hacer una copia fiel de lo que nosotros les entreguemos.

Seguramente se estén preguntando si es necesario realizar todas estas etapas, y la respuesta es que no. Si lo que estamos haciendo es un rediseño o una actualización de un producto ya existente, no tendría sentido que comencemos por el sketch, podemos acudir directamente a los wireframes o a los mockups, dependiendo de las necesidades.

En cambio, si se trata de un producto nuevo, seguramente sí tengamos que atravesar cada una de estas instancias, para poder verificar y modificar los distintos elementos de nuestro diseño. Lo importante es que podamos utilizar las fases que nos permitan llegar más rápidamente a nuestra solución.

Para ello, tenemos que tener en cuenta las fidelidades a las que hicimos referencia, porque a medida que nos acercamos a la etapa de mockup la fidelidad va a ser cada vez más alta y por lo tanto nuestro diseño va a ser mucho más concreto a nivel visual.

### Principios heurísticos

Es momento de que conozcamos qué tenemos que tener en cuenta para el desarrollo de la interfaz de nuestro producto. Para eso, vamos a conocer los principios heurísticos que elaboró Jakob Nielsen para que los usuarios puedan comprender cómo utilizar el producto que estamos desarrollando.

Recordemos que la interfaz es el medio por el cual las personas van a interactuar con el diseño, por lo que tiene que resultar sumamente clara para que el usuario lo pueda usar. Es por este motivo que Nielsen creó estos 10 principios fundamentales para cumplir con el objetivo de la usabilidad.

Se tratan de:

- La visibilidad del estado del sistema
- La consistencia entre el sistema y el mundo real
- El control y la libertad para el usuario
- La consistencia y los estándares
- La prevención de errores

- La minimización del uso de la memoria
- La flexibilidad y la eficiencia de uso
- El diseño minimalista
- El poder reconocer, diagnosticar y recuperarse de los errores
- Y la ayuda y la documentación

La **visibilidad del estado del sistema** se trata de que el sistema nos informe qué es lo que está haciendo. ¿Se acuerdan que, anteriormente, cuando decíamos que la computadora estaba “pensando”, el cursor se convertía en un reloj de arena para pedirnos que esperemos? Bueno, es eso mismo. El sistema nos debe dar algún tipo de indicación visible para que podamos saber si se realizó una tarea, si ya se completó, etc. De esta manera, los usuarios conocerán cuál es el resultado de sus interacciones y podrán determinar cuáles serán sus próximos a seguir.

Para analizar con un ejemplo lo que podríamos hacer para cumplir con esta norma, podemos tomar a Whatsapp. Para enterarnos de que nuestro mensaje le llegó al destinatario, podemos ver dos tildes grises, que, si se convierten en azules, nos está informando que el receptor ya lo leyó. Mientras que, si el mensaje queda con una tilde gris sola, eso nos estaría indicando que el mensaje aún no fue recibido por el destinatario. Esta es una manera muy sencilla de que el sistema nos informe qué hizo con la orden que nosotros le dimos.

Por lo tanto, la recomendación que realiza Nielsen es, más allá de comunicar cuál es el estado del sistema, que no se tomen acciones drásticas que tengan consecuencias para los usuarios, sin informárselos. Su importancia radica en que *“las interacciones predecibles generan confianza tanto en el producto como en la marca”*.

El siguiente principio es el de la **consistencia entre el sistema y el mundo real**. Se trata de utilizar el lenguaje de los usuarios, es decir, usar palabras y frases conocidas por las personas y no términos propios del sistema. Pero, además, se pueden incorporar imágenes que representen la cotidianidad de los usuarios. Recordemos el ejemplo que habíamos visto del *switch*. Cuando las personas asocian los gráficos de un dispositivo con su vida diaria, es mucho más sencillo que puedan comprender cómo se usan.

Un ejemplo muy concreto de esto es la famosa “papelera de reciclaje” que tenemos en las computadoras. Su ícono representa un tacho de basura que, si está vacío, indica que no eliminamos nada. Pero, si está lleno, con el dibujo de papeles abollados adentro, nos

damos cuenta de que contiene los archivos que decidimos eliminar. Y, al igual que un tacho de basura físico, si los quisiéramos recuperar, los podemos sacar de ahí. Pero, en el momento que la vaciamos, ya no vamos a poder acceder a ellos.

Pasemos ahora al **control y la libertad** para el usuario. Esto implica que las personas tengan la libertad para moverse libremente por la interfaz. Por lo tanto, es importante tener la posibilidad de cancelar acciones que estén en progreso o tener una especie de “salida de emergencia” para poder volver para atrás si ingresamos en un sector que no era el que queríamos, sin tener que pasar por un proceso prolongado.

Además, es fundamental poder deshacer cualquier acción que realicemos, porque esto les brinda a los usuarios una sensación de libertad y confianza. Bien sabemos que hay algunas que son irreversibles, pero, para poder cumplir con esta ley, el sistema nos debe informar que esa acción no se podrá deshacer, de manera que sea el usuario el que determine qué decisión tomar.

Un ejemplo muy claro es cuando queremos eliminar un mail en algunos proveedores. No solo el sistema nos pide que confirmemos si realmente queremos hacer eso, sino que tenemos unos segundos para deshacer esa acción. Por eso, Nielsen sugiere incorporar una forma clara de suspender un proceso, como un botón que diga “cancelar”, y que sea muy visible la opción de volver para atrás.

Luego, la **consistencia y estándares** se trata de que los usuarios no necesiten preguntarse si las palabras que encuentran, por ejemplo, en una aplicación, significan lo mismo que en otras. Es decir, que si entran en Youtube entiendan que el botón de “buscar” cumple la misma funcionalidad que el de “buscar” en MercadoLibre, porque las experiencias que tienen los usuarios con los distintos productos establecen sus expectativas.

Es por eso que se utilizan convenciones en el diseño de la interfaz, para que se relacione con los modelos mentales que tienen los usuarios. De esta manera, van a poder utilizar nuestro producto de manera intuitiva.

Otro ejemplo que podemos mencionar en este caso, es el ícono del carrito de compras. Los usuarios comprenden que en ese lugar se van a encontrar los artículos que está adquiriendo en el sitio, entonces sería un error diseñar una interfaz que contenga ese ícono pero que cumpla otra función completamente diferente.

Por eso, Nielsen sugiere que haya una coherencia tanto interna como externa. Por interna se refiere a que, dentro de un mismo producto, o de una familia de ellos, como podría ser el paquete Office, se mantenga la consistencia entre las funcionalidades de sus botones. Y luego, por coherencia externa, explica la necesidad de mantener las convenciones de la industria.

Analicemos ahora la **prevención de errores**. Este punto se relaciona con tratar de evitar que los usuarios cometan errores, porque, de acuerdo con Nielsen *“los buenos mensajes de error son importantes, pero los mejores diseños evitan cuidadosamente que ocurran problemas en primer lugar”*. Pero, por supuesto, no hay nada que sea infalible, por lo que, en caso de ocurran estos errores, se les debe poder indicar de forma clara cómo los pueden subsanar.

Ahora bien, hay dos formas de cometer equivocaciones: los deslices y los errores propiamente dichos. Los deslices implican equivocaciones que cometen los usuarios de forma inconsciente, por no estar prestando la debida atención. Un ejemplo muy sencillo es cuando intentamos enviar un mail y nos olvidamos de escribir el asunto. En ese momento, el sistema nos va a hacer la advertencia de que eso está ocurriendo para que nosotros decidamos si continuamos o no. Lo mismo ocurre cuando escribimos mal una palabra y nos la subraya en rojo.

Pero otra posibilidad son los errores conscientes que se dan por el desajuste que se produce entre el modelo mental del usuario y el diseño. Esto es lo que tenemos que evitar. De cualquier manera, es importante que en este punto podamos darles las advertencias adecuadas a los usuarios y, por supuesto, la opción de deshacer.

**Minimizar el uso de la memoria** se trata de que los usuarios puedan reconocer, en lugar de tener que recordar. Es decir, que el sistema logre minimizar la información que un usuario tendría que recordar, mostrándola con objetos, acciones u opciones porque, de acuerdo con Nielsen, los seres humanos tienen una memoria limitada a corto plazo.

Las opciones que ofrece para poder cumplir con este objetivo son que permitamos que las personas reconozcan la información que se encuentra en la interfaz, de manera que no tengan que recordarla. Y, también, debemos tratar de reducir la cantidad de contenido que los usuarios deben memorizar. Para poder llevarlo a cabo, es necesario que nuestro diseño cuente con procesos que eviten la utilización de la memoria del usuario, como historiales o sugerencias.

Por eso, es que un ejemplo es, en los sitios de compras, tener un historial al que podamos acceder. De esta manera, si quisiéramos comprarle algo al mismo vendedor, no necesitaremos recordar cómo se llamaba, sino que el sistema nos lo va a mostrar. Otra posibilidad es la función de autocompletar, por la que el sistema nos ofrece sugerencias de cómo terminar la frase que estamos escribiendo.

La **flexibilidad y eficiencia de uso** implica que tanto los usuarios experimentados, como los nuevos, puedan acceder al sistema de manera eficaz. Una forma de garantizarlo, es con los aceleradores, que permiten acceder más rápidamente a las distintas funciones. Uno de los más conocidos es el atajo “ctrl+c” para copiar o “ctrl+v” para pegar.

Sin embargo, para aquellos usuarios que no sepan cómo utilizar estos aceleradores, el sistema debe presentar la opción de cortar y pegar de la forma más larga, seleccionando las opciones en el menú de inicio. Entonces, será apto en ambos casos. De esta forma, las personas podrán elegir el método que mejor les resulte.

Muy bien, pasemos ahora al **diseño minimalista**. En este caso, su nombre nos da una idea bastante concreta de lo que expresa este principio: se trata de que la interfaz no esté sobrecargada de elementos que dificulten su comprensión y funcionalidad. Por eso, debemos procurar que figure únicamente la información necesaria para la tarea que tenemos que completar, sin ornamentos, texturas o tipografías complejas, de manera que los elementos innecesarios no distraigan a los usuarios de la información principal.

Si son usuarios de *Gmail*, seguramente hayan notado que la pantalla de inicio de sesión es muy simple: tiene un fondo blanco, el botón principal, el de “acceder”, se destaca por sobre los demás elementos, y solamente podemos realizar acciones vinculadas al inicio de sesión. En esa pantalla no vamos a poder hacer ninguna otra actividad más que esa, lo que logra que sea muy clara su funcionalidad.

El anteúltimo principio que tenemos que analizar es el de **reconocer, diagnosticar y recuperarse de los errores**. Es similar al punto de prevención de errores, pero en este caso se trata más bien del mensaje que le tenemos que transmitir a los usuarios para que se repongan de estas equivocaciones. Para este caso sirve el mismo ejemplo que pusimos cuando analizamos el UX writing con la contraseña incorrecta. Es necesario que el sistema indique dónde se produjo el error y qué opciones nos brinda para poder resolverlo.

Otro ejemplo es cuando ingresamos a un link que, por algún motivo, fue deshabilitado. Si nosotros entramos ahí y solamente aparece un mensaje que dice “error” no nos va a quedar claro si es un problema constante de la página, que estaba caída en un momento determinado o que nos funciona mal internet. En cambio, si sale un mensaje que dice “esta página no existe”, al menos vamos a saber que, no importa cuántas veces le demos clic a ese link, nunca podremos ingresar allí.

Por lo tanto, para poder cumplir con este principio, es necesario que utilicemos un lenguaje claro y que le brinde al usuario toda la información precisa para comprender por qué motivo se produjo el error. En este punto es crucial que evitemos la jerga técnica, porque si aparece el clásico cartel de “error 404”, no entenderemos qué ocurrió.

Además, como dijimos, el sistema nos debe ofrecer opciones para reponernos de esta situación, por más que sea simplemente volver a la página de inicio. De esta manera, el usuario va a poder tener el control de la situación. Y, también, incluir tratamientos visuales que contribuyan al reconocimiento de los errores por parte de los usuarios.

Finalmente, llegamos a la **ayuda y documentación**. Si bien el producto debería ser lo suficientemente claro como para no necesitar instrucciones, no podemos omitir una sección de ayuda, que contenga preguntas frecuentes y una guía paso a paso para realizar determinadas acciones.

Lo importante es que toda esta información sea accesible para el usuario, esto quiere decir que la pueda encontrar fácilmente, porque también puede ocurrir que haya productos más complejos que no se puedan utilizar de manera intuitiva, como podría ser el caso de un *software* de facturación. Por lo tanto, la documentación debe ser concisa, fácil de buscar, y relacionada con las tareas que deben completar los usuarios.

### Evaluación heurística

Como habrán podido ver, cumplir con todos estos principios nos permitirá que nuestro producto sea lo más utilizable posible. Para corroborar que hayamos cumplido con todos, podemos realizar una evaluación heurística.

Mediante esta técnica, que implica hacer pruebas de usabilidad sin usuarios, vamos a poder determinar si todos estos principios fueron cumplidos. Pero para realizarla, no

puede participar una única persona, porque podría pasar que no encuentre todos los errores de usabilidad que podría haber en la interfaz.

En base a las investigaciones realizadas por Nielsen, es que sugiere que sean cinco personas las que realicen esta prueba, que deberán hacerla de manera individual. Para llevarla a cabo tendremos que armar una planilla en la que pondremos cada una de las leyes y, mediante el testeado del prototipo, verificaremos cuáles fueron incluidas y cuáles no.

Por lo tanto, podemos realizar esta prueba en cualquier momento, siempre que contemos con un prototipo funcional. De lo contrario, no se puede realizar. Lo que sí es aconsejable es hacerla antes de las pruebas de usabilidad, en las que los usuarios reales prueban el producto, porque podrían surgir estos errores que son menores y nos perderíamos de información valiosa que nos podrían aportar los usuarios por no poder completar las tareas que les proponemos.

Tengamos en cuenta que esta evaluación implica determinar la calidad de uso de nuestra interfaz, por lo que vamos a poder detectar estos primeros errores. Luego, iteraremos el diseño del prototipado, y una vez que hayamos concluido que todas estas leyes fueron cumplidas, estaremos en condiciones de presentarle el prototipo a los usuarios reales para que realicen las pruebas de usabilidad.

## **MVP**

Para finalizar, nos vamos a introducir en el concepto de MVP, que son las siglas en inglés de minimum viable product, o producto mínimo viable. Se trata de una metodología por la cual vamos a crear el producto con las funcionalidades mínimas, de manera que podamos sacarlo al mercado para que los usuarios lo utilicen y, de esa manera, validarlo rápidamente.

De acuerdo con el ingeniero en sistemas Diego Lopez Castan, *“esta filosofía tiene como valor principal el proceso de aprendizaje. Lo que se busca es crear prototipos rápidos para comprobar si nuestro cliente ve valor en nuestro producto”*, y agrega que *“por otro lado conseguimos feedback de nuestros usuarios que nos van a permitir mejorar nuestro producto digital o cambiar el rumbo del mismo”*.

Entonces, cuando decimos “mínimo viable” estamos hablando de que, para lanzar nuestro producto al mercado, su funcionalidad principal tiene que estar completa. Para eso, tenemos que establecer cuál es la tarea que tienen que poder realizar los usuarios esencialmente.

Veamos un ejemplo: como ya hemos comentado, la función principal por la que surgió Whatsapp era la de enviar mensajes a un costo muy bajo. Entonces, si le quitamos las funcionalidades que aparecieron después, como los mensajes de voz, las videollamadas, los gifs y los stickers, todavía se podrían enviar esos mensajes de texto. Por lo tanto, su función principal es realizable.

En eso nos tenemos que centrar cuando creamos un MVP, porque si lanzamos al mercado nuestro producto y los usuarios pueden realizar la tarea principal, ya vamos a poder empezar a validarlo. Más adelante le vamos a ir incorporando las demás funciones, pero al menos ya tenemos algo para comenzar a testear.

Entonces, los usuarios lo van a probar y a nosotros nos sirve para recabar toda la información que nos brinden mediante los reportes de uso, de eficiencia y de eficacia, así como las reseñas que nos dejan las personas. De esta forma, podremos detectar qué mejoras podemos hacer, volvemos al principio del proceso de Design Thinking e iteramos el prototipado.

Obviamente, antes de lanzar nuestro MVP al mercado, vamos a realizar las validaciones y evaluaciones pertinentes. Pero la información más exacta que podemos recibir es aquella que nos dan nuestros usuarios reales. Por eso es que este proceso es tan importante, porque nuestro producto va a entrar en contacto con la realidad de la experiencia de los usuarios.

Por lo tanto, de acuerdo con Lopez Castan, los beneficios de realizar un MVP son:

- *“Ayudar al aprendizaje del problema*
- *Validar el producto antes de crearlo en su totalidad*
- *Probar productos con recursos mínimos*
- *Reducir el desperdicio de recursos*
- *Aumentar la velocidad de salida a producción del producto”.*

Esta metodología va ser muy utilizada por aquellas empresas que recién están empezando. El consultor de marketing digital, Iván de Souza, explica que *“las startups*

*son las que más utilizan concepto de MVP, pues generalmente tienen poco dinero y una idea central de gran novedad en el mercado”.*

El desarrollo de un MVP no solamente nos va a permitir comprobar si nuestro producto funciona como lo esperábamos, sino que también es una instancia importantísima para corroborar si realmente cumple con las necesidades de los usuarios. De esta manera, podremos continuar perfeccionándolo o directamente cambiar el foco de nuestra propuesta.

La mayoría de las grandes empresas lanzan estos MVP sobre los cuales van realizando la incorporación de nuevas funcionalidades. De hecho, los invitamos a fijarse las aplicaciones que tengan instaladas en sus celulares. Seguramente vean en la descripción de producto, el número de versión, que generalmente se ve representado de la siguiente manera: 1.12.25, etc.

Todos esos números representan la cantidad de versiones que salieron al mercado. Es decir que se trata de estos MVP con todas las iteraciones que se les realizaron para incorporarles características nuevas que se fueran adaptando a las nuevas necesidades de los usuarios.

Ahora bien, dijimos que lanzar un MVP implicaba mantener solamente la funcionalidad principal del producto. Pero, también es cierto que el mercado se modificó sustancialmente. Entonces, si nuestro producto es completamente nuevo, vamos a poder cumplir con esa premisa. En cambio, si lo que estamos desarrollando tiene alguna competencia directa, para poder mantenernos activos vamos a tener que sumar funcionalidades que seguramente no sean las principales.

Vamos a ejemplificarlo para explicarlo mejor. Actualmente, hay diversas aplicaciones de vehículos de traslado. Si nuestro producto va a cumplir ese mismo objetivo, podríamos decir que su función principal sería la de poder solicitar un auto, indicándole, como mínimo por dónde nos tiene que pasar a buscar. Ahora bien, si no le incorporamos algunas características más, como el destino final para que nos muestre el proyecto que realizará, o si no podemos pagar con tarjeta, seguramente nuestros usuarios piensen que nuestra aplicación no tiene lo que ellos necesitan y no la utilicen.

Por eso es que, en algunos casos, el MVP va a presentar algunas funcionalidades que, a simple vista, no son las principales. Esto se debe a que va a tener que cumplir con ciertos

requerimientos mínimos a los que los usuarios ya están acostumbrados a utilizar con la competencia, para lograr mantenerse en el mercado.

Estas decisiones generalmente dependen del cliente que contrata a los diseñadores para desarrollar su producto, porque es él el que evalúa cuáles son las necesidades del negocio y qué requisitos mínimos debe cumplir para que sea satisfactoria su inserción en el mercado.

Pero la desventaja que tiene esta situación, es que va a demorar el lanzamiento del producto, porque va a requerir de mayor cantidad de etapas de diseño, lo que nos va a demorar, también, las validaciones que pretendemos realizar. Por lo tanto, en la medida que sea posible, debemos centrarnos en las funciones principales para que nuestro producto pueda salir al mercado cuanto antes, siempre asegurándonos de que esté completo.

Por otro lado, es importante que, una vez que salió al mercado y que las personas comenzaron a utilizarlo, no nos demoremos mucho en evaluar cuáles son los aspectos que debemos modificar o incorporar, en base al feedback que recibimos de nuestros usuarios. Iván de Souza explica que, *“si este tiempo es demasiado extenso, puedes perder el timing de la solución o regalarla a otros empresarios”*.

Esto quiere decir que, si “nos dormimos en los laureles” y no lanzamos rápidamente una nueva versión mejorada de nuestro MVP, puede ocurrir que otras empresas tomen nuestra idea, la perfeccionen y la saquen ellos al mercado. Por lo tanto, una vez que comenzamos con este proceso, tenemos que estar atentos y ser expeditivos a la hora de incorporar soluciones para la satisfacción de los usuarios.

### **Early adopters**

Un último tema que vamos a estudiar en relación con el MVP es que nos va a permitir conocer a nuestros usuarios. Tengamos en cuenta que cuando hicimos nuestras investigaciones para comenzar con el diseño, determinamos los arquetipos de persona y las validamos. Entonces, cuando lancemos nuestro producto al mercado, vamos a poder constatar quiénes son las primeras personas que acceden a él y si coinciden con esos arquetipos.

Los primeros usuarios de un producto son llamados early adopters. De acuerdo con el sitio web Destinonegocio.com, *“los early adopters son los primeros clientes porque tienen problemas específicos y están a la búsqueda de productos que puedan solucionarlos”*. Además, aclaran que estas personas no están buscando la perfección del producto, sino que, por el contrario, suelen compartir sus opiniones sobre cómo se podría mejorar.

Algunos ejemplos de este tipo de usuarios, son aquellas personas fanáticas de la marca Apple, que se caracterizan por adquirir sus productos aun si están en fase de prueba. Y, luego, podemos mencionar también aquellos productos que se lanzan teniendo en cuenta a un público específico, que resulta ser su early adopter, como fue el caso de Tik Tok.

Esta aplicación fue pensada para ser utilizada por adolescentes, por eso su funcionalidad principal de creación de videos atrajo mayormente a ese público, así como sus actualizaciones posteriores. Sin embargo, esto no quiere decir que solamente esas personas vayan a usar la aplicación, porque, de hecho, muchos adultos se sumaron posteriormente. Pero lo importante es que sí fue creada para ellas y por eso fueron sus early adopters.

Tener en consideración a estos primeros usuarios nos puede brindar importantes beneficios. En Destinonegocio.com explican que *“el sociólogo norteamericano Everett Rogers tiene un gráfico que señala como se divide el mercado. Él decía que los primeros clientes que tiene una empresa antes de convertirse en masiva son los early adopters y los innovadores, -que son aquellos pocos clientes que siempre quieren estar a la vanguardia-, una vez que estos han validado un producto los productos tiene una curva de crecimiento y son adoptados por una mayoría de pragmáticos, luego son adoptados por la mayoría conservadora, para finalmente capturar a los escépticos”*.

Por eso es que toda la información que nos puedan brindar, nos va a servir para realizar las mejoras necesarias de manera que podamos atraer, posteriormente, a un mayor número de personas. Porque, además, como explica el cofundador de NNGroup, Donald Norman, los early adopters implican un pequeño porcentaje del mercado. La mayoría está conformada por los pragmáticos y los conservadores, quienes son más proclives a acceder a estos nuevos productos una vez que hayan pasado por diversos procesos de mejora.

Es por este motivo que son tan importantes estos primeros usuarios, porque según Norman, ellos *“pasarán por alto la inestabilidad, la dificultad de uso, la apariencia poco elegante”*. Entonces, su experiencia con nuestro producto nos va a dar un primer *feedback*. Pero luego, *“el conjunto de clientes cambia radicalmente (...) los consumidores quieren eficiencia, placer y conveniencia. Esto requiere una forma de desarrollo de producto muy diferente a la que se puede utilizar en las primeras etapas de una tecnología”*.

Y agrega que *“la gran mayoría de los clientes son usuarios tardíos. Esperan hasta que la tecnología haya demostrado su eficacia y luego insisten en la conveniencia, la buena experiencia del usuario y el valor”*. Por eso es que es tan importante poder realizar todas las validaciones posibles, para que podamos satisfacer las necesidades de la mayoría de las personas, ofreciendo un producto de calidad.