EL PROCESO CREATIVO:

Los pasos del proceso creativo:

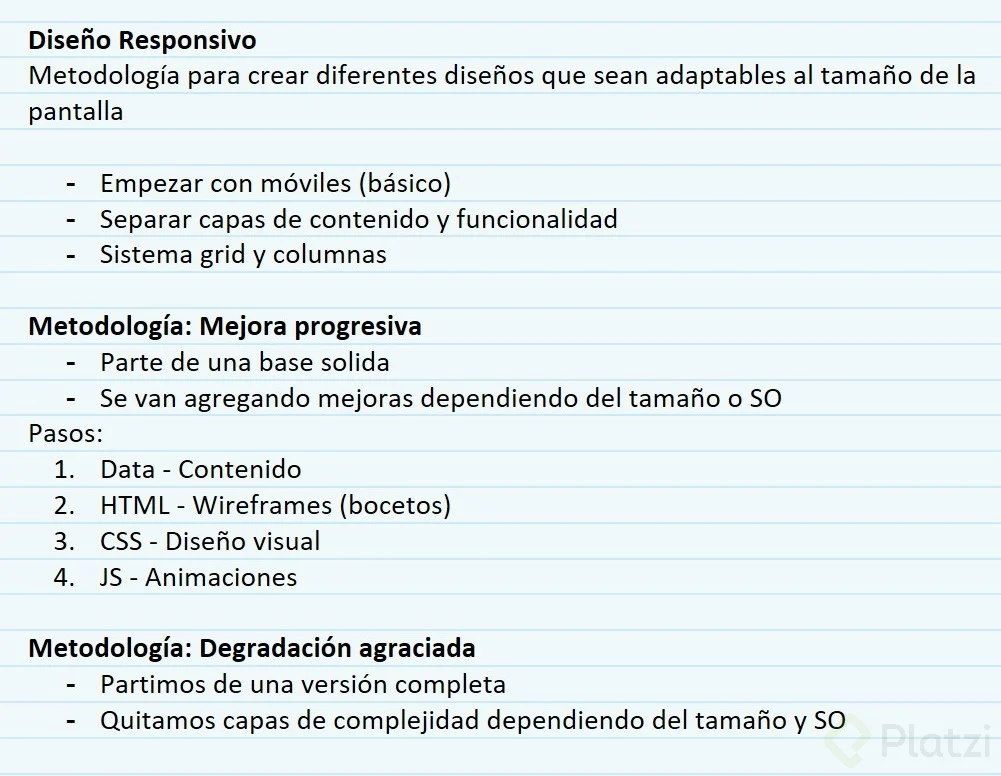
1. **Preparación**: Investigar, recopilar. Investigamos y recopilamos toda la información relativa a un problema o algo que queramos solucionar.
2. **Incubación**: Experimentar, sintetizar. Buscamos diferentes soluciones. Miramos como alguien ya lo soluciono.
3. **Iluminación**: Idear, imaginar. Aquí es donde se nos ocurre las ideas. Ideamos e imaginamos soluciones.
4. **Evaluación**: Criticar, replantear. Evaluamos si estas soluciones son prácticas, son viables. Vemos los fallos y si es necesario, replanteamos las soluciones.
5. **Implementación**: Construir, trabajar. Construimos la idea hasta el producto final. “La creatividad es algo que se desarrolla”.

Conceptos:

1. **Balance**: la posición de cada elemento está dada por su peso visual. Puede ser:

a. Simétrico  
b. Asimétrico

1. **Contraste**: Señalar un concepto marcado por la diferencia entre 2 elementos distintos (claro obscuro, grande pequeño).
2. **Alineación**: Crear rutas visuales.
3. **Proximidad**: Agrupar elementos.
4. **Repetición**: Concepto de marca.
5. **Espacio**: Como hacerle uso.



TIPS PARA LA ACCESIBILIDAD Y DISEÑO:

Una parte importante del diseño es la accesibilidad. La accesibilidad es la posibilidad de que cualquier persona pueda acceder a la web, no importa la incapacidad que tenga. Esta accesibilidad es obligatoria en todas la Webs.

-. Usar los encabezados correctos en cada lugar.

-. Usar fuentes legibles y visibles fácilmente.

-. Usar el contraste de colores adecuadamente.

-. Garantizar que los colores no sean el único código para relacionar el contenido.

-. Diseña teniendo en cuenta los eventos de la página. No quitar el focus o el active, y que estos sean diferentes.

-. Añade títulos descriptivos a los campos de un formulario.

-. Los ALT y descripciones en la fotos y videos es vital.

-. Evitar que las animaciones bloqueen el contenido.

BRIEF Y REQUERIMIENTOS TECNICOS

El primer elemento que debemos tener claro para construir una aplicación es el BRIEF.

¿Qué es el BRIEF? Es la hoja de ruta para empezar a diseñar. Es lo que se habló en los pasos del proceso creativo: investigar y recopilar toda la información relativa con el proyecto, con el problema a solucionar. Es donde apuntamos esos datos que hemos recopilado.

Las secciones más comunes son:

• Descripción del cliente o producto.

• Objetivos o retos.

• El target o audiencia al que va destinado ese producto o servicio.

• La competencia que puede haber.

• Como será la distribución.

Es importante que este documento sea bonito, este organizado y podamos ver bien la jerarquía de contenidos.

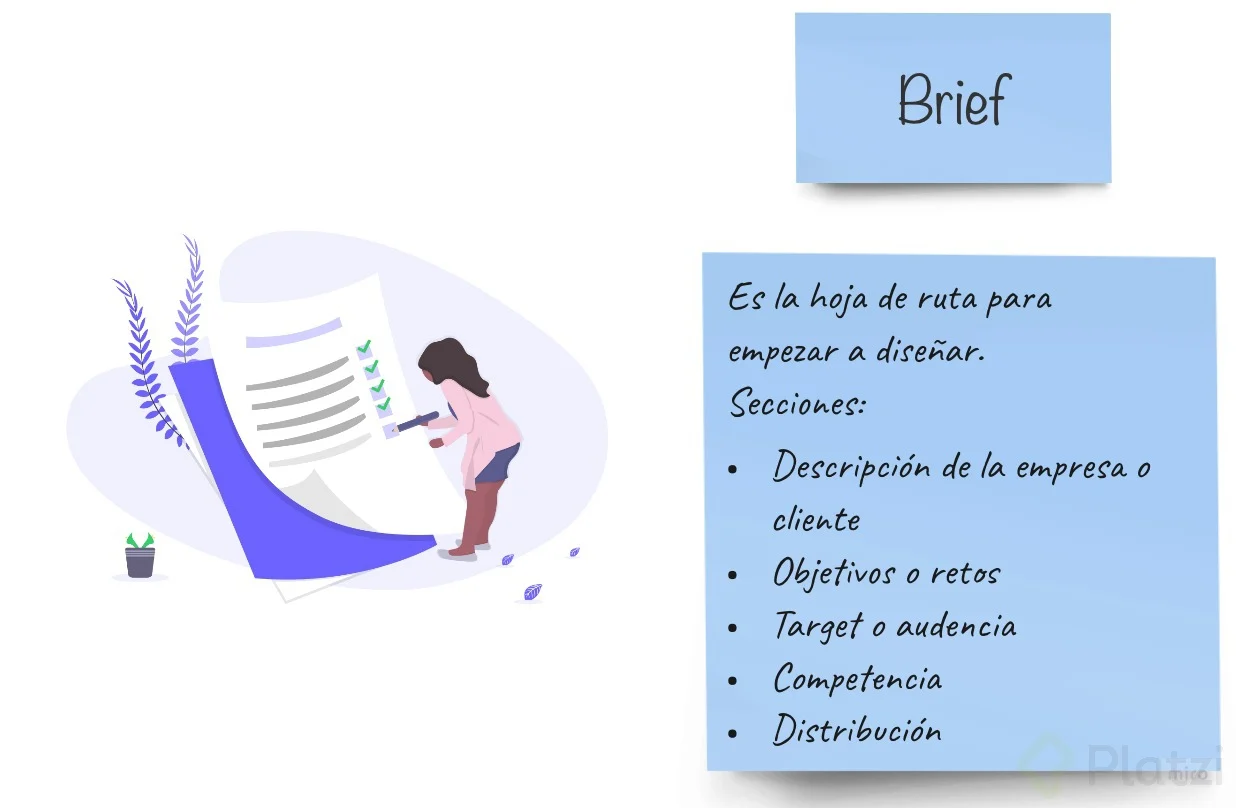
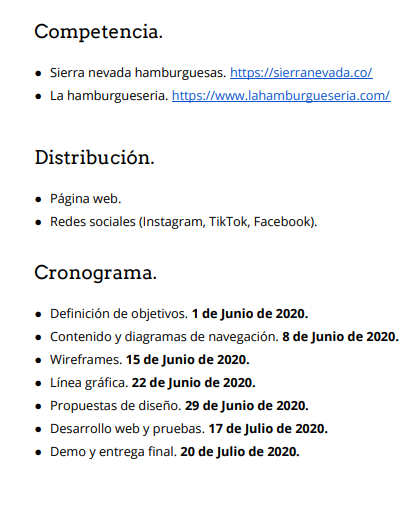
Es importante que cualquier persona que lea el BRIEF sepa que está leyendo.

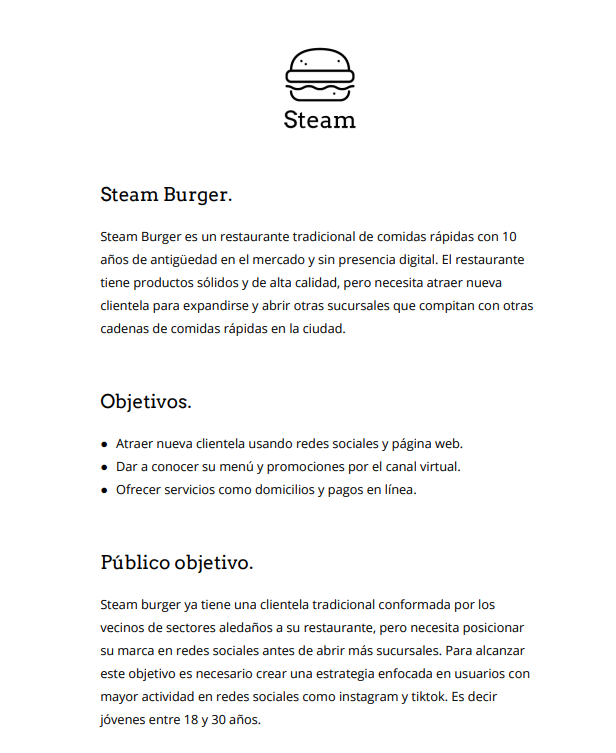
Los objetivos es la parte más importante, porque de ahí vamos a partir para crear nuestro producto.

También es bueno tener un cronograma donde vamos a poner las fechas de las entregas.

<https://platzi.com/blog/tipos-de-brief/>

EJEMPLO:



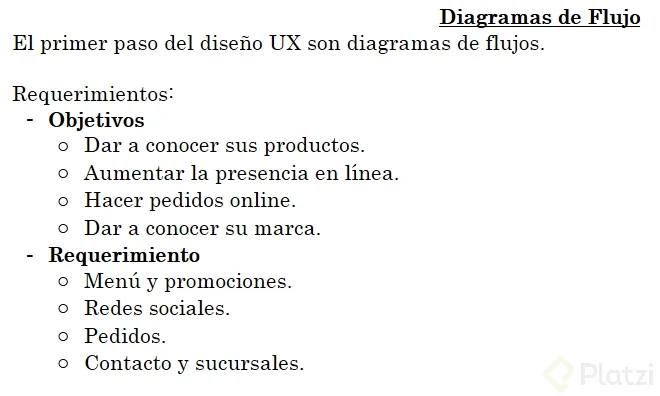


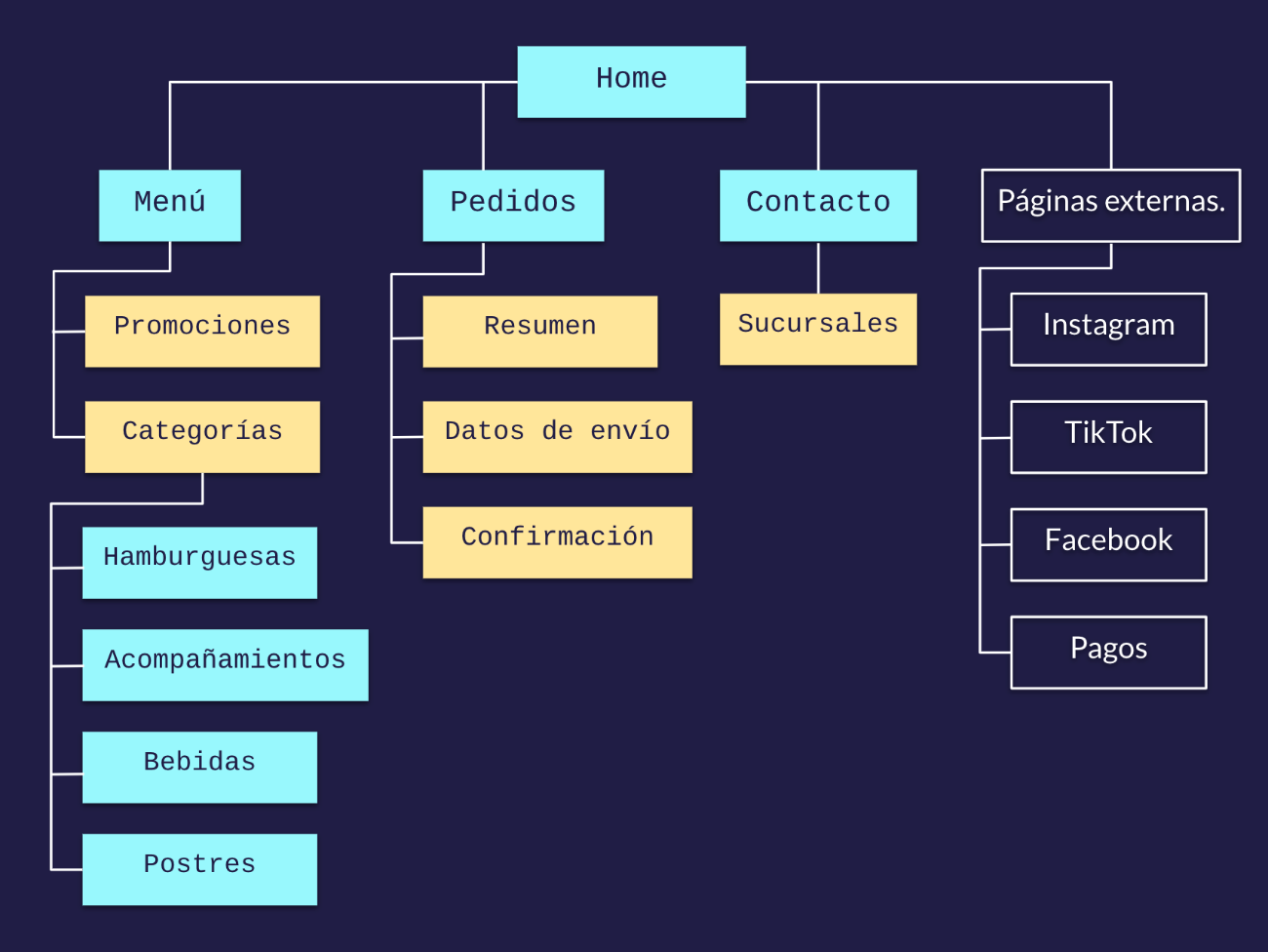
**Definición de diseño UX**

1. **Investigación -->** *Recopilar información para conocer que ocupan los usuarios principalmente. Es bueno apoyarse de aplicaciones ya existentes para crear un buen diseño UX.*
2. **Análisis** --> *Una vez recopilada la información se analiza y se obtienen los puntos importantes que debemos tener en cuenta al momento de crear el diseño.*
3. **Diseño** --> *Se crean prototipos o sketches para visualizar el resultado.*
4. **Pruebas de usuario** --> *Por lo general se llevan a cabo en los sketches para poder realizar ajustes antes de llevar el diseño a código.*

**DIAGRAMAS DE FLUJO**

*Primer paso de nuestro diseño UX son los diagramas de flujo. Pasamos los elementos de nuestro BRIEF a elementos tangibles. El primer diagrama se llama SITEMAP Y es un mapa completo de nuestro sitio. Aquí definimos cuales son las secciones principales, las secciones secundarias y las páginas externas a nuestro sitio. El siguiente es el USER FLOW. Estos nos permiten hacer un diagrama de todos los pasos que hace un usuario para completar una tarea. Con este flujo podemos saber cuántas pantallas componentes tenemos para realizar una acción de usuario. No hay un número definido de pantallas o componentes que vamos a diseñar para un trabajo en particular.*

**





**Sitemap**  
Un mapa del sitio es un diagrama jerárquico que permite visualizar la estructura de un sitio web. Se utiliza para definir la taxonomía del sitio agrupando los contenidos relacionados. Los mapas del sitio identifican la estructura del sitio y, hasta cierto punto, los tipos de páginas. Los mapas del sitio nos muestran qué va dónde y cómo están conectadas ciertas páginas.

**User Flows**  
Los flujos de usuario detallados, o a veces llamados flujos de tareas, se sumergen en los detalles para entender todos los pasos y decisiones que un usuario necesita tomar para pasar por su idea, característica o producto. Esto nos ayuda a comprender realmente todas las pantallas y estados que hay que diseñar y tener en cuenta en el proceso.

WIREFRAMES

¿Qué es un WIREFRAMES? Es el plano de nuestra aplicación. En ellos hacemos nuestros bocetos en papel de todos los componentes y pantallas que nos salieron en los flujos que hemos creado antes.

Los WIREFRAMES dibujados a mano se consideran de baja calidad, pero con ellos podemos empezar a hacer pruebas de usuarios.

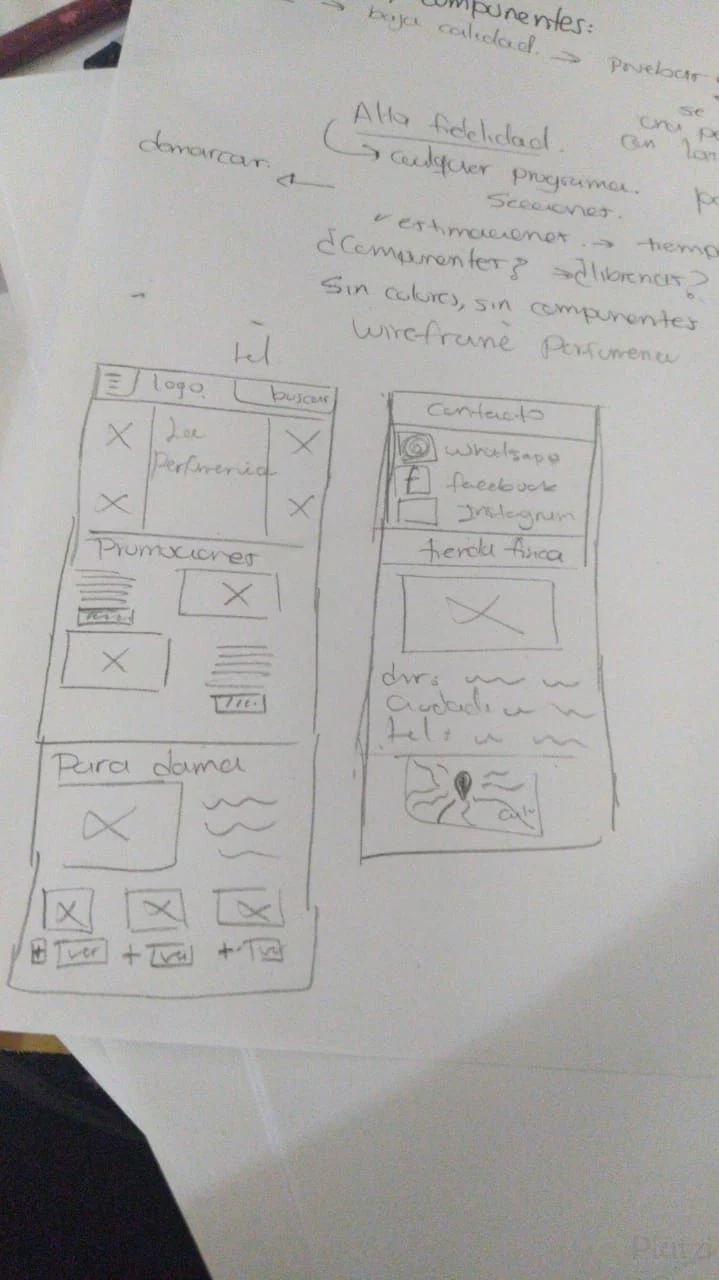
Podemos hacer estas pruebas de muchas formas. Lo importante es ver la usabilidad de los componentes y pantallas, si son muy complejos para los usuarios finales.

Los WIREFRAMES de alta fidelidad son los creados con programas de diseño.

Esta parte es importante en diseño y también en desarrollo, porque en esta etapa los desarrolladores van a ver el tamaño de la aplicación final, y poder hacer estimaciones, además de decidir con que herramientas lo va a programar.

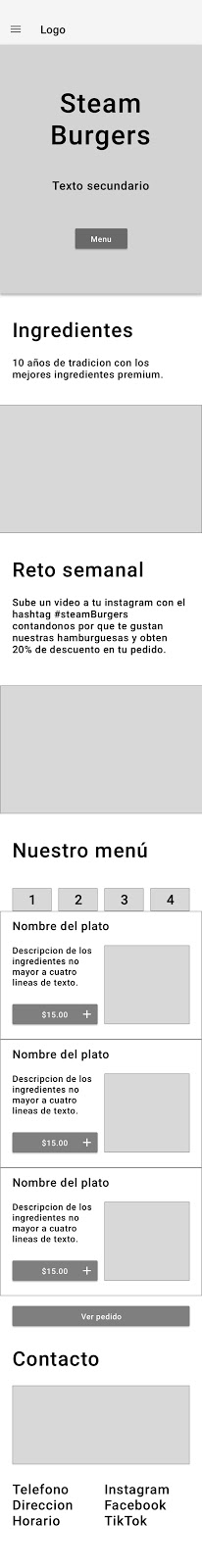
También es importante que lo vea el cliente para que se haga una idea de que secciones va a tener su producto.

En esta etapa no se usan las paletas de colores. Solo es un boceto en blanco y negro o grises.



**Ejemplo tienda de hamburguesas:**

**Mobile:**



**Desktop:**



DEFINICION DE DISEÑO UI (Diseño de interfaz del usuario)

Es el momento de ponerle color a los bocetos.

El diseño UI es el diseño que se enfoca en la capa de estilos visuales que va por encima de la estructura de contenido.

Se diferencia del UX en que este último se enfoca en la investigación, prototipado y arquitectura de la información.

De que se encarga cada diseño.

**UX:**

o Diseño de interacción.

o Prototipado.

o Arquitectura de la información

o Investigación y pruebas de usuario

**UI:**

o Diseño visual.

o Colores.

o Layouts.

o Tipografías.

Cuando trabajamos con los dos diseños al mismo tiempo garantizamos que nuestra aplicación va a ser bonita, agradable y va a ser funcional para todos los usuarios. No debemos quitarle importancia a ninguno de los dos.





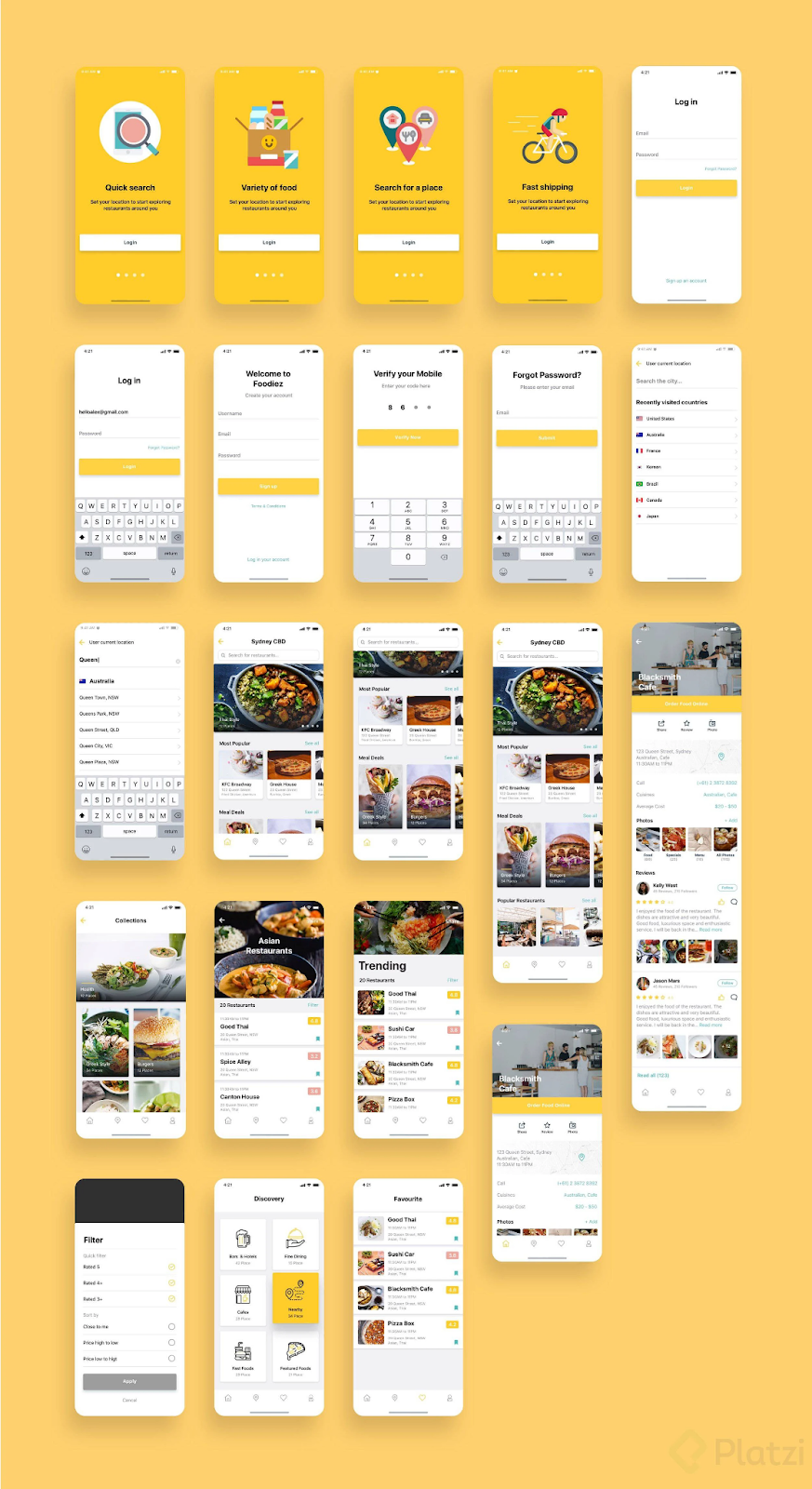
**Moodboard**:

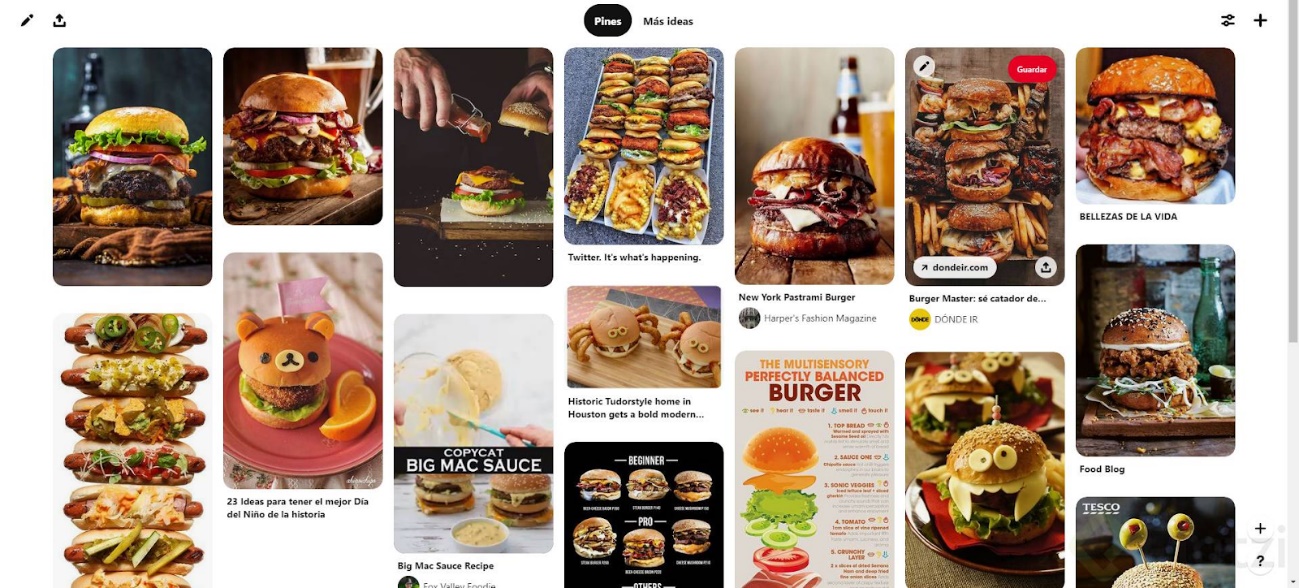
Su traducción literal es mapa de inspiración, Un moodboard es una colección de referencias visuales, una herramienta creativa que consiste en una visualización rápida de imágenes y palabras en un mismo soporte.

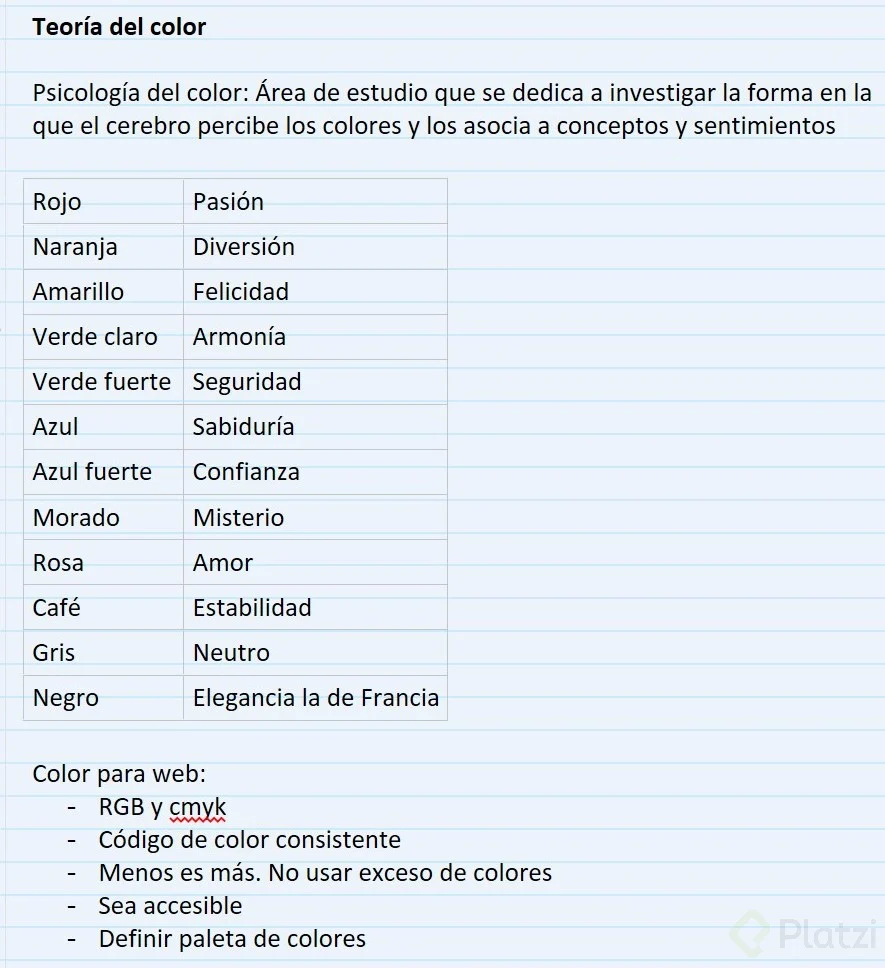
Estas imágenes y palabras pueden ser tanto físicas, con recortes de revistas o páginas impresas, como en formato digital. Un ejemplo de moodboard digital muy reconocible podría ser un muro de Pinterest, donde bajo un tema común se aglutinan imágenes y palabras que nos gustan.

espero les guste mi definición, es la más específica para ustedes.

EJEMPLOS DE MODBOARD:







**Paletas de colores:**

[**https://colors.muz.li/**](https://colors.muz.li/)

PALETA DE COLOR.  
Para construir una paleta de color, tenemos que tener claro que tipo de colores existen:  
• Colores primarios: Amarillo, azul y rojo.  
• Colores secundarios: Son los que resultan de la combinación de los colores primarios.  
• Colores terciarios: Son los colores intermedios que hay entre un color primario y un color secundario.  
Tipos de paletas:  
• Combinación monocromática: Diferentes tonos de un mismo color. (diferente opacidad).  
• Combinación análoga: Resulta de combinar un color primario con uno secundario y uno terciario que estén seguidos en el circulo cromático.  
• Combinación complementaria: Consiste en combinar un color primario con un color secundario que no estén seguidos en el circulo cromático. Estos colores están opuestos en el circulo.  
• Combinación triádica: Se trata de combinar tres colores que estén en el circulo cromático de forma que forman un triángulo dentro del círculo.  
• Combinación tétrada: Combinamos 4 colores elegidos al formar un rectángulo dentro del círculo cromático. Se eligen dos primarios y dos secundarios.

TIPOGRAFÍAS.  
Otra de las cosas esenciales en una aplicación es el tipo de letra.  
Tips para el uso eficiente de tipografías:  
• No uses demasiados tipos de letras.  
• Usa fuentes estándar.  
• Limita la cantidad de texto. No son libros. Mucho texto echa a los usuarios de la página.  
• Selecciona tipos de textos legibles en diferentes tamaños.  
• Mantén las líneas separadas unas de otras. Los altos de línea que sean adecuados.  
• Que tengan suficiente contraste el texto con el fondo.  
• Evitar las animaciones intermitentes en los textos.  
Reglas de combinación de tipografía:  
Las fuentes tienen personalidad.  
• Serif: tradicional, sofisticada, confiable, práctica, formal… Se puede combinar con Sans Serif, scripts, display… y se suele usar en logos, textos de párrafos, títulos, impresos.  
• Sans Serif: Moderna, limpia, humanista, geométrica, universal. Se combina con serif, script y slab serif. Sus usos son: logos, textos de párrafos, títulos y textos pequeños.  
• Script o fuente cursiva: Elegante, clásica, formal, sofisticada y estilizada. Combina con serif y sans serif. Se usa en logos, títulos e invitaciones. También textos cortos que queramos destacar.



LAYOUT Y SISTEMA DE GRILLAS.  
¿Qué es una grilla?  
Es un sistema de columnas creado por nosotros de acuerdo a las necesidades para ajustar nuestros elementos y componentes.  
La mejor herramienta de CSS para crear una grilla es Grid.  
Sass es un pre-procesador de CSS te ayuda a escribir CSS de una manera más rápida y más fácil.  
Mixin es una clase que tiene dentro del pre-procesador, que te ayuda a manejar mejor los Breakpoints.  
Tenemos que configurar nuestros BreakPoints (Tipos de pantalla).  
• $xs: 360px. Para móviles pequeños.  
• $s: 440px. Para móviles con la pantalla más grande.  
• $m: 768px. Para tablets.  
• $l: 1280px. Para Ordenadores pantalla normal.  
• $xl: 1440px. Monitores de alta calidad.

Podemos usar un Mixin para manejar los distintos breakpoints más fácilmente.  
Después vamos a configurar nuestras variables dependiendo de los breakpoints.  
–columns. Es el número de columnas que vamos a poner.  
–column-gap es el espacio entre las columnas.  
Creamos después una clase Grid para configurar cada uno de los elementos contenedores donde vamos a incluir nuestros componentes.  
Finalmente le añadimos un display grid, que nos permite que toda esta configuración surta efecto. Y definimos un grip-column-gap para la distancia entre columnas y un grid-template-column que nos permite tener esta estructura.  
Para definir el tamaño de un componente, se pone grid-column: 1/5, donde uno es la columna donde empieza, y 5 es la columna donde termina.

SISTEMAS DE COMPONENTES UI  
Una de las metodologías que podemos usar es separar la funcionalidad por componentes.  
Un sistema de componentes es un conjunto de componentes que vamos a usar par crear nuestra aplicación. Esos componentes funcionan como una librería.  
Puedes usar cualquier tecnología que prefieras. Lo importante es encapsular las funcionalidades en componentes. Cada componente tendrá su propio css y funcionalidad dentro de su propia función.  
Así podremos importar esas características a cualquier pantalla de nuestra aplicación, reutilizando el código y haciendo más eficiente nuestra aplicación.  
En la clase de Wireframe creamos unos bocetos de unos componentes, que luego dividimos en unidades de funcionalidad.  
La idea es crear a partir del wireframe todos los componentes y funcionalidades, y después le añadimos la capa visual.

**Imágenes para Web**

* JPG: Imágenes con degradados (ligero pero sin tanta calidad)
* PNG: Fondo transparente
* SVG: Vectores, usados en iconos y animaciones
* GIF: En movimiento, es pesado

**Como elegir:**

* Aporten al contenido
* Dirigido al cliente
* Consistente con la paleta de colores
* 👀 Licencias

**Rendimiento y accesibilidad**

* Evitar texto en las imágenes
* Exportar al tamaño del contenedor final
* Lazy loading: Cargando dinámico
* Texto alternativo

GRAFICOS EN MOVIMIENTO PARA WEB  
Formatos de gráficos en movimiento:  
• CSS animado: Indicado para animaciones sencillas y transiciones.  
• SVG animado: Indicado para animaciones de elementos vectoriales.  
• JS (Canvas, WebGL): Indicado para animaciones complejas como animaciones de datos o 3D. Hay librerías en JS que sirven para este tipo de animaciones.  
• Videos: Indicado para filmaciones o animaciones de alta complejidad y corta duración. Siempre preguntarnos: ¿realmente necesito este video? Pues son pesados y ralentizan la carga del sitio.  
¿Cómo elegir gráficos en movimiento?  
• Elegir animaciones que aporten al contenido. No sobrecargar con muchas animaciones.  
• Procurar que no se reproduzcan automáticamente y si lo hacen, que no tengan sonido.  
• Evita que tus animaciones tengan flashes. Los brillos continuos molestan y pueden ser dañinas.  
• Si tus animaciones aportan contenido, añade subtítulos o transcripciones para que las pueda leer el lector de pantalla.  
• Evita que las animaciones bloqueen la lectura básica del contenido. Es molesto una animación a la hora de leer, además de que el lector de pantallas no podrá acceder a ese contenido.  
• Recuerda que las animaciones y los videos ralentizan la carga de la página. Se aplica nuevamente el menos es más.

Algunas librerías de animaciones para CSS

* [Hover](https://ianlunn.github.io/Hover/)
* [CSShake](https://elrumordelaluz.github.io/csshake/)
* [Anijs](https://anijs.github.io/)
* [CSS Wand](https://www.csswand.dev/?ref=producthunt)
* [Animate Components](https://animate-components.surge.sh/)
* [Keyframes](https://keyframes.app/)
* [AniCollection](https://anicollection.github.io/#/)
* [SpinKit Loader](https://tobiasahlin.com/spinkit/)
* [Animejs](https://animejs.com/)
* [Animte CSS](https://animate.style/)

### Paso para el Diseño

* Paso 1: Creamos nuestro Brief -> Resumen de plan
* Paso 2: Creamos nuestro site Map -> Mapa del sitio de solución al brief
* Paso 3: Creamos nuestro wireframe -> diseño en blanco y negro en Desk y Tablet
* Paso 4: Creamos nuestro sketch -> diseño con tonos de color e iconografías en Desk y Tablet
* Paso 5: Definir componentes para evitar retrabajo
* Paso 6: No olvides mostrar tu trabajo y tener mente abierta para los comentarios y criticas todo comentario es bueno para mejorar, claro obviando los comentarios tóxicos.