*1er desafío complementario*

*Lograr que todo el proyecto sea totalmente responsive utilizando grids para el layout, flexbox para los componentes y box modeling para terminar de acomodar los elementos. Es necesaria la utilización de media queries.*

Componentes:

1. Estructura básica de la Web
2. Estilo básico de la Web

*Estructura de la web*

* **Formato:** Archivos HTML
* **Objetivo del desafío:**
  + **Maquetar la web:** El estudiante deberá, en base a la estructura visual que desea plantear con las nuevas herramientas de Flexbox y Grids, agregar y/o reestructurar los elementos de su HTML.
  + **Estructuración del contenido:** El estudiante deberá llevar su contenido a la estructura HTML haciendo uso de los tags que corresponden para el contenido a insertar.

**Incluir:**

* **Maquetado de la web:** Las etiquetas deben ser en base al contenido que incluirá y su futura estructuración visual desde el estilo.
* **Páginas:** agregar más contenido a sus páginas utilizando las etiquetas correspondientes.

**Formato:** Archivo CSS

* **Objetivo del desafío:**
  + El estudiante deberá aplicar lo aprendido en box-modelling, flex, y grids para estructurar el contenido de su web armando layouts en las que distribuye el contenido.

**Incluir:**

* **Estructuración visual del contenido:** usar reglas de CSS para crear layouts y distribuir los elementos de su sitio web en base al diseño planteado originalmente. Debe valerse de flex o box-modeling para estructuras unidimensionales, o grid para bidimensionales. Deben incluirse media queries para generar un diseño responsive.

*RÚBRICAS DE EVALUACIÓN*

*1er desafío complementario*

| **Estructura de la web** | | | |
| --- | --- | --- | --- |
| **Criterios** | **Bajo**  Falta más profundización. Es confuso. | **Correcto**  Acorde pero con errores puntuales. | **Óptimo**  Es claro y pertinente. |
| Código prolijo | * El uso de tabulaciones y/o nuevas líneas marca de forma correcta la jerarquía de padre/hijo pero hay inconsistencias; ya sea que es es errático, usa diferentes niveles de tabulaciones, usa diferentes tamaños de tabulaciones, o los elementos no cierran en el mismo nivel en el que empiezan. | * Tabulaciones correctas y ordenadas, denotando jerarquía entre los elementos. * No hay saltos erráticos y hay un tamaño consistente de tabulación. | * La estructura es clara por el uso de tabulaciones y líneas nuevas consistentes en tamaño y cantidad. * Las etiquetas comienzan y terminan en la misma columna. |
| Comentarios  <!-- --> | * Se usan demasiados comentarios (seguramente por la falta de tags semánticas). | * Los comentarios se usan de forma pragmática para saber dónde comienzan y/o terminan secciones. * Comenta demasiado código para uso futuro o porque ya no lo va a usar. | * Usa los comentarios de forma pragmática y pertinente para documentar secciones de su HTML/CSS. |
| Tags HTML | * Nesting acorde en general. * Errores mínimos a la hora de escribir tags. * Tags semánticas usadas pero con problemas para comprender cual sirve para cada caso. | * Buen nesting, usándolo con el fin de jerarquizar y estructurar los elementos. * Los tags y atributos están bien escritos. * Crea tags que envuelven a otros innecesariamente, ya sea porque no cumplen ninguna función o no se usan. | * El HTML no contiene errores en los atributos y tags. * El nesting es óptimo. Uso correcto de tags semánticos y estructuración de la página desde el HTML. |
| Estilo en el HTML | * Hojas de estilo bien linkeadas y con todo el CSS. * Usa demasiadas clases innecesariamente por aún no saber cómo reutilizarlas u organizarlas. * Clases de CSS declaradas en el HTML pero no aparecen en las hojas de estilo. | * Hojas de estilo bien linkeadas. * Las hojas de estilo tienen todo el CSS. * Uso de clases CSS eficiente y óptimo. Re-utilizan clases para no re-definir reglas de CSS para tags con estilos iguales (como colores, fondos, etc). | * Uso de clases eficiente en los atributos. * Nombres de clases consistentes, usa BEM. |
| Funcionalidad | * La web tiene enlaces rotos. * Las fotos no se cargan por errores en la ruta. * No hay enlaces para navegar por las diferentes páginas. * El usuario queda atrapado al no tener como volver a la home por falta de enlaces. * Se usan rutas absolutas para archivos de la web, incluso haciendo uso del protocolo file:// | * La web tiene enlaces, que representan secciones que van a desarrollarse. * Las fotos están bien cargadas pero son de relleno (no tienen un propósito o funcionan como “placeholder”). * Hay enlaces que llevan a diferentes páginas y tiene como volver a la home. * Se usan rutas relativas para los archivos de la web. | * Los enlaces a las secciones funcionan correctamente. * Las páginas están correctamente interconectadas y el usuario puede navegar entre ellas. * Las imágenes están bien cargadas y el contenido es pertinente a la web (no simulado o placeholder). * El enlace a la home es el logo. * Las rutas relativas son correctas. |
| Contenido | * El contenido tiene errores ortográficos. * Hay contenido que no corresponde a la página donde está ubicado. * El contenido está desorganizado. | * El contenido está bien escrito y no presenta faltas ortográficas. * Hay poco contenido o está incompleto en algunas de las páginas. * La organización del contenido es equitativa pero mal distribuida a lo largo de la página. * Usa textos que no aportan relevancia a su sitio. (lorem) | * Las páginas tienen una cantidad contenido apropiado y está bien distribuida. * El contenido no tiene faltas ortográficas o gramaticales. * El contenido es apropiado a la sección en la que está ubicado. |

| Código de la estructura visual o layout | * Se usa posición absoluta o relativa para armar el layout visual de los contenidos. * Se usa float para armar columnas de contenido. * Se usan tablas para armar layouts. * Uso de br para separar elementos en vez de usar reglas de spacing. | * Logra el layout de web haciendo uso de reglas CSS ineficientes. * Usa flex para layouts multidimensionales complejos que se favorecen del uso de grids. * Usa grids para layouts unidimensionales simples que se favorecen del uso de flex. | * Usa reglas apropiadas para armar los diferentes tipos de layout. * El uso de tablas es solo para mostrar contenido estructurado. * El uso de flex y grid es apropiado para el tipo de layout armado. * Uso de br apropiado, separando los párrafos del texto. |
| --- | --- | --- | --- |
| Diseño de la estructura visual o layout | * El diseño del layout de la web no es consistente a lo largo de las páginas. * Elementos de misma jerarquía son inconsistentes página a página. * El estilo visual de los elementos web cambia mucho página a página. * La ubicación de elementos de navegación cambia de lugar. * El layout de la web no es intuitivo o de fácil navegación * El estilo visual de algunos elementos web es el default. | * El diseño del layout de la web es consistente a lo largo de algunas páginas pero no en todas. * El layout del sitio web es navegable. * Los elementos de la misma jerarquía, son consistentes a lo largo de las diferentes páginas. * El estilo visual de los elementos web se mantiene consistente a lo largo de las páginas. | * El diseño del layout es consistente página a página. * Los estilos definidos para los elementos se mantienen consistentes a lo largo de las páginas. * La interfaz web planteada por el layout es intuitiva y navegable. |
| Diseño web atractivo | * Elementos fuera que no pertenecen a sección sueltos a lo largo de la página. * Uso de colores chillantes o con mucho contraste entre sí. * El texto no es legible. * La página ocupa todo el ancho del navegador lo que dificulta la lectura. * No hay una paleta de colores. | * Uso correcto de colores pero el texto no es legible. * La paleta de colores varía a lo largo de las páginas. * Hay un elemento contenedor pero es demasiado grande o demasiado chico. * El elemento contenedor no está centrado o alineado a nada. * Los elementos respetan la paleta pero varían en diseño página a página. | * El contraste entre los colores es apropiado. * Hay una paleta de colores y se respeta a lo largo de las páginas del sitio web. * El texto es legible. * Hay un elemento contenedor que evita que el contenido vaya hasta los bordes de la pantalla y está alineado. |