



Bottega DevCamp

FULL STACK DEVELOPER

CHECKPOINT 2: EJERCICIOS TEÓRICOS

Fecha: 13 de febrero de 2025

Autor: Domínguez Becerril, Jorge

DNI: 72607195J

Índice

| | |
|---|----------|
| 1. Cuestiones teóricas | 1 |
| 1.1. Nombra cuatro valores que se pueden usar con la propiedad justify-content. | 1 |
| 1.2. ¿Qué es un objetivo primordial? | 3 |
| 1.3. ¿Qué son los wireframes de baja fidelidad? | 4 |
| 1.4. ¿Qué es Git y para qué sirve? | 5 |
| 1.5. ¿Qué es un mapa del sitio? | 6 |
| 1.6. Nombra tres valores que se pueden usar con flex-direction, incluido el valor predeterminado que usa flexbox cuando se crea. | 7 |

1. Cuestiones teóricas

1.1. Nombra cuatro valores que se pueden usar con la propiedad justify-content.

Algunos de los valores que se pueden usar con la propiedad justify-content son los siguientes:

- start
- end
- center
- space-between
- space-around

El valor start hace que los items flex estén alineados en el borde de la izquierda de la pantalla, tal y como se puede ver en la figura 1.



Figura 1: Ejemplo de justify-content para el valor start.

El valor end hace que los items flex estén alineados en el borde de la derecha de la pantalla, tal y como se puede ver en la figura 2.



Figura 2: Ejemplo de justify-content para el valor end.

El valor center hace que los items flex estén alineados en el centro de la pantalla, tal y como se puede ver en la figura 3



Figura 3: Ejemplo de justify-content para el valor center.

El valor space-between hace que los items flex estén distribuidos uniformemente. Los items flex están equiespaciados, con el primer item alineado con el borde de la izquierda de la pantalla y el último item alineado con el borde derecho de la pantalla. En la figura 4 se puede ver un ejemplo de este comportamiento.



Figura 4: Ejemplo de justify-content para el valor space-between.

Por último, el valor space-around hace que los items flex estén distribuidos uniformemente. El espaciado entre cada par de items es el mismo. El espaciado antes del primer item y después del último es la mitad del espaciado entre cada par de items. En la figura 5 se puede ver un ejemplo de este comportamiento.



Figura 5: Ejemplo de justify-content para el valor space-around.

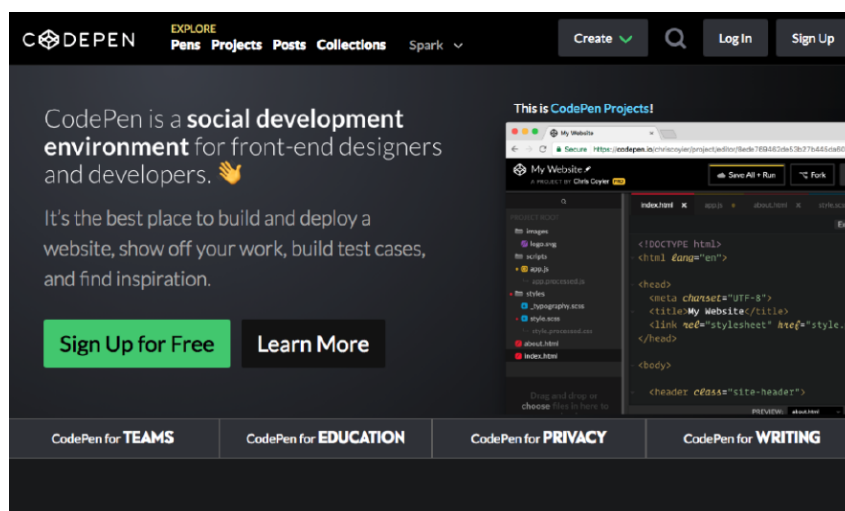
1.2. ¿Qué es un objetivo primordial?

El objetivo primordial de un proyecto responde al propósito principal que se busca lograr con su desarrollo y funcionamiento. Por ejemplo, en los años 2000 era complicado consumir contenido digital como series y películas. Entonces surgió Netflix, acercando a la gente una infinidad de series y películas por un precio competente. Netflix se creó con el propósito de cubrir esa carencia de los consumidores.

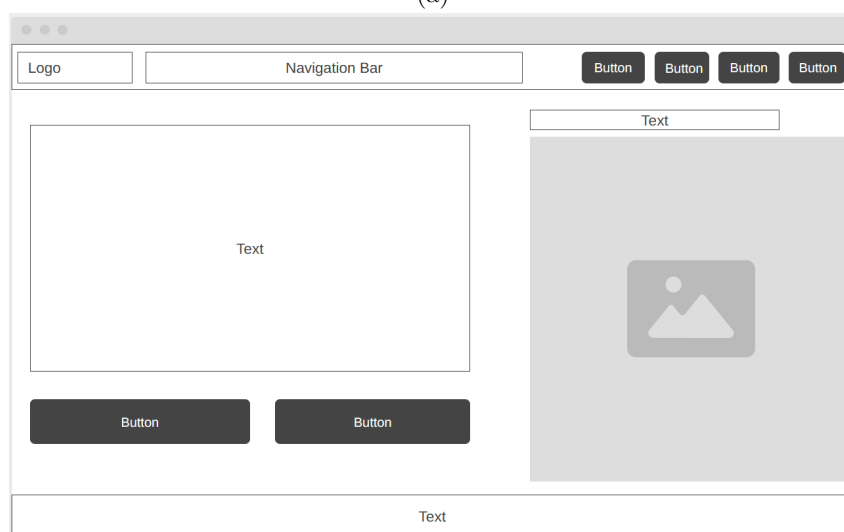
Resulta vital conocer el objetivo primordial de un proyecto. De esta manera, cada característica o funcionalidad se puede desarrollar pensando en este objetivo último, así como rechazar aquellas ideas que no guardan relación con este.

1.3. ¿Qué son los wireframes de baja fidelidad?

Un wireframe de baja fidelidad es un plano del diseño de una página web o aplicación muy rudimentario. En él se muestra la estructura básica del sitio web prescindiendo de los detalles cósméticos. A modo de ejemplo, en las figuras 6(a) y 6(b) se muestra un sitio web y su wireframe de baja fidelidad.



(a)



(b)

Figura 6: Sitio web de Codepen (a) y su correspondiente wireframe de baja fidelidad (b).

Como se puede comprobar, el logo y la imagen de la página de inicio se han sustituido por cajas (placeholder). Los botones también se han sustituido por cajas en las que pone Button y cuyo color no se corresponde con el color final. El texto presente en el sitio web tampoco aparece en el wireframe, sino solo una caja en la que pone Text. Con esto se pretende hacer un boceto rápido con los principales elementos que aparecerán en la página web y, de esta forma, empezar a trabajar en el código lo antes posible.

1.4. ¿Qué es Git y para qué sirve?

Git es un sistema de control de versiones distribuido de código abierto que permite gestionar el código de un proyecto de manera eficiente.

Algunas de sus características son las siguientes:

- **Control de versiones:** el sistema Git guarda un historial de los cambios realizados en los archivos. Esto permite que, en caso de que una nueva característica o funcionalidad esté dando problemas, se pueda retroceder a una versión anterior del proyecto.
- **Distribuido:** esto significa que cuando un desarrollador quiere implementar una nueva característica o funcionalidad clona el proyecto para trabajar en una copia local. De esta manera, diferentes desarrolladores pueden estar trabajando para implementar nuevas funcionalidades sin que los cambios que cada uno de ellos realiza en el código cree problemas a los demás desarrolladores.
- **Código abierto:** Git es un sistema de código abierto, lo que significa que su uso es completamente gratuito. Esto también presenta la ventaja de que hay una gran comunidad detrás del sistema y cualquier duda que le surja a algún desarrollador seguramente ya haya sido respondida por otra persona.

1.5. ¿Qué es un mapa del sitio?

Un mapa del sitio es un documento en el que se organiza y presenta la estructura de un sitio web, indicando las páginas más importantes y su interrelación. Tiene que contener todas las historias de usuario. En la figura 7 se muestra un mapa del sitio de un sitio web concebido para escuchar música y mensajear.

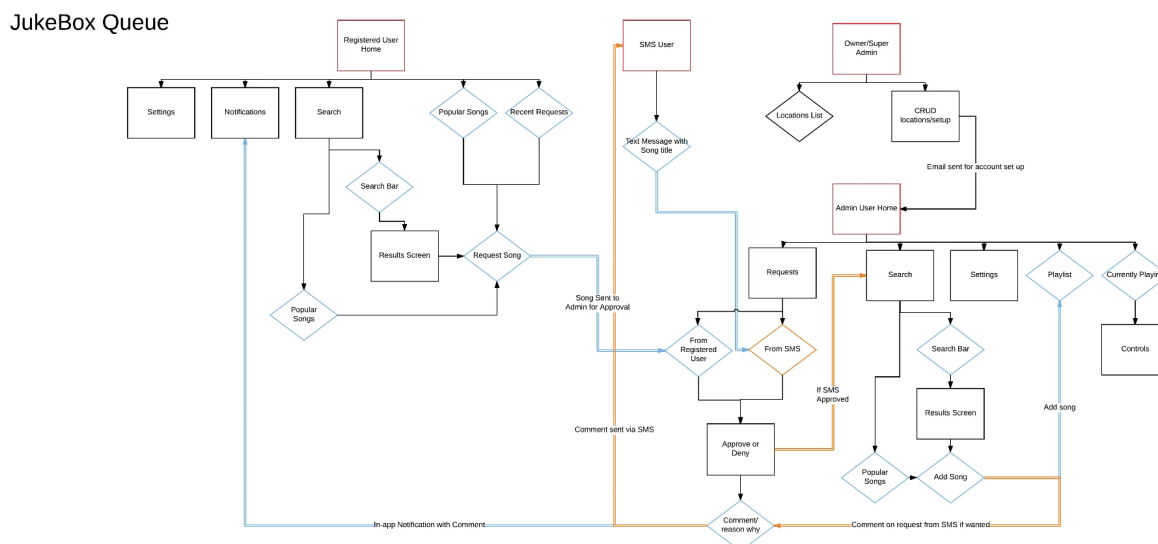


Figura 7: Sitemap de un sitio web concebido para escuchar música y mensajear.

Como se puede ver, cada una de las acciones conduce a otra acción o página web. Esto permite tener una visión global de todas las páginas que se deben crear y las acciones que se pueden realizar desde cada una de ellas, cubriendo todas las historias de usuario del proyecto.

1.6. Nombra tres valores que se pueden usar con flex-direction, incluido el valor predeterminado que usa flexbox cuando se crea.

Algunos de los valores que se pueden usar con la propiedad flex-direction son los siguientes:

- row
- row-reverse
- column

El valor row hace que los items flex estén distribuidos horizontalmente, tal y como se puede ver en la figura 8. Es el valor por defecto en los contenedores flex.



Figura 8: Ejemplo de flex-direction para el valor row.

El valor row-reverse hace que los items flex estén distribuidos horizontalmente, como ocurre con el valor row. Sin embargo, a diferencia de este, el orden de los items se invierte, tal y como se puede ver en la figura 9.



Figura 9: Ejemplo de flex-direction para el valor row-reverse.

Por último, el valor column hace que los items flex estén distribuidos verticalmente, tal y como se puede ver en la figura 10.

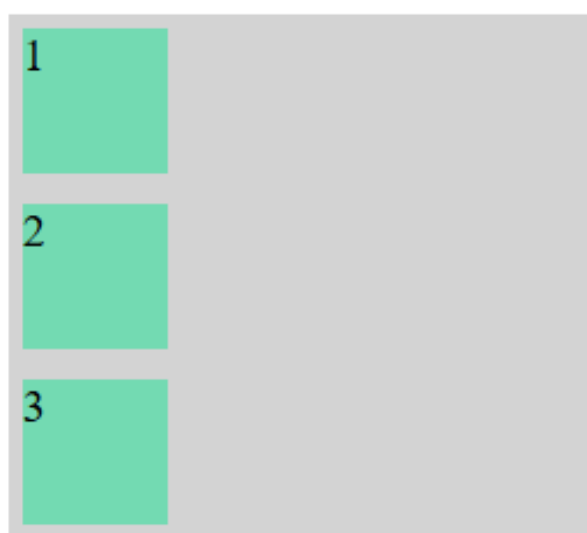


Figura 10: Ejemplo de flex-direction para el valor column.