



CLASE #2

Frontend, Backend y anatomía de una página web

FRONTEND

Si el desarrollo fuese construir un restaurante, los frontend serían responsables de **construir la parte que es visible, de la atención a los clientes y de coordinar con la cocina.**

Principales funciones:

1. Tomar los pedidos de los clientes
2. Pasar los pedidos a la cocina.
3. Recibir los pedidos de la cocina.
4. Servir los pedidos a los clientes.



Principales desafíos:

Adaptarse a todo los tipos clientes.
Atender rápidamente a los clientes.
Lograr que los **clientes realicen pedidos**.

Se realizan pedidos a la cocina

La cocina entrega los pedidos

BACKEND

Si el desarrollo fuese construir un restaurante, los backend serían responsables de **construir la cocina, preparar los pedidos y dejarlos listos para que el front los sirva al cliente**

Principales funciones:

1. Tomar los pedidos pasado por front.
2. Tomar los ingredientes
3. Preparar los pedidos.
4. Entregar los pedidos.



Principal desafíos:

Preparar de forma **segura**.
Prepara **varios pedidos a la vez**.
Disminuir el **tiempo de preparación**.

FRONTEND

Se encarga de escribir el código para **hacer visible el sitio**, que permite la **interacción con los usuarios** y de la **comunicación con el Backend**.

Principales funciones:

1. Tomar los pedidos de los usuarios.
2. Pasar los pedidos (como datos) al Backend.
3. Tomar los pedidos (datos) desde el backend.
4. Mostrar (visualizar) los datos a los clientes.

Principales desafíos:

Adaptarse a los distintos navegadores.

Desplegar rápidamente el sitio (Performance)

Lograr que los **clientes realicen pedidos** (usabilidad).

BACKEND

Se encarga de escribir el código que **crea o utiliza los servicios que dan respuesta a los pedidos** y de la **comunicación con el frontend**.

Principales funciones:

1. Tomar los pedidos (datos) del frontend.
2. Acceder a los datos o servicios para realizar los pedidos.
3. Realizar los pedidos.
4. Entregar los pedidos (datos) al frontend

Principal desafío:

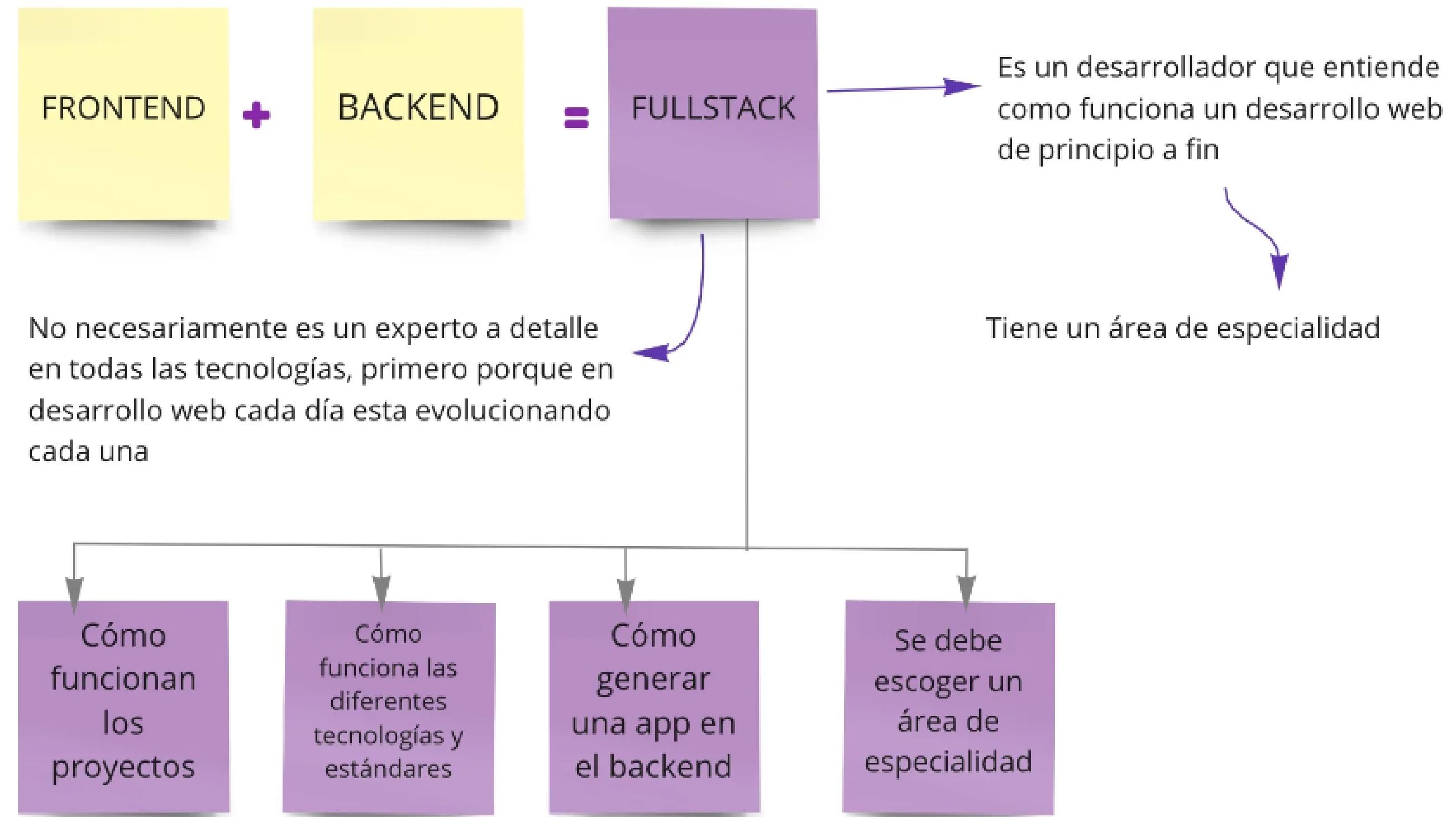
Garantizar la **seguridad del sitio**.

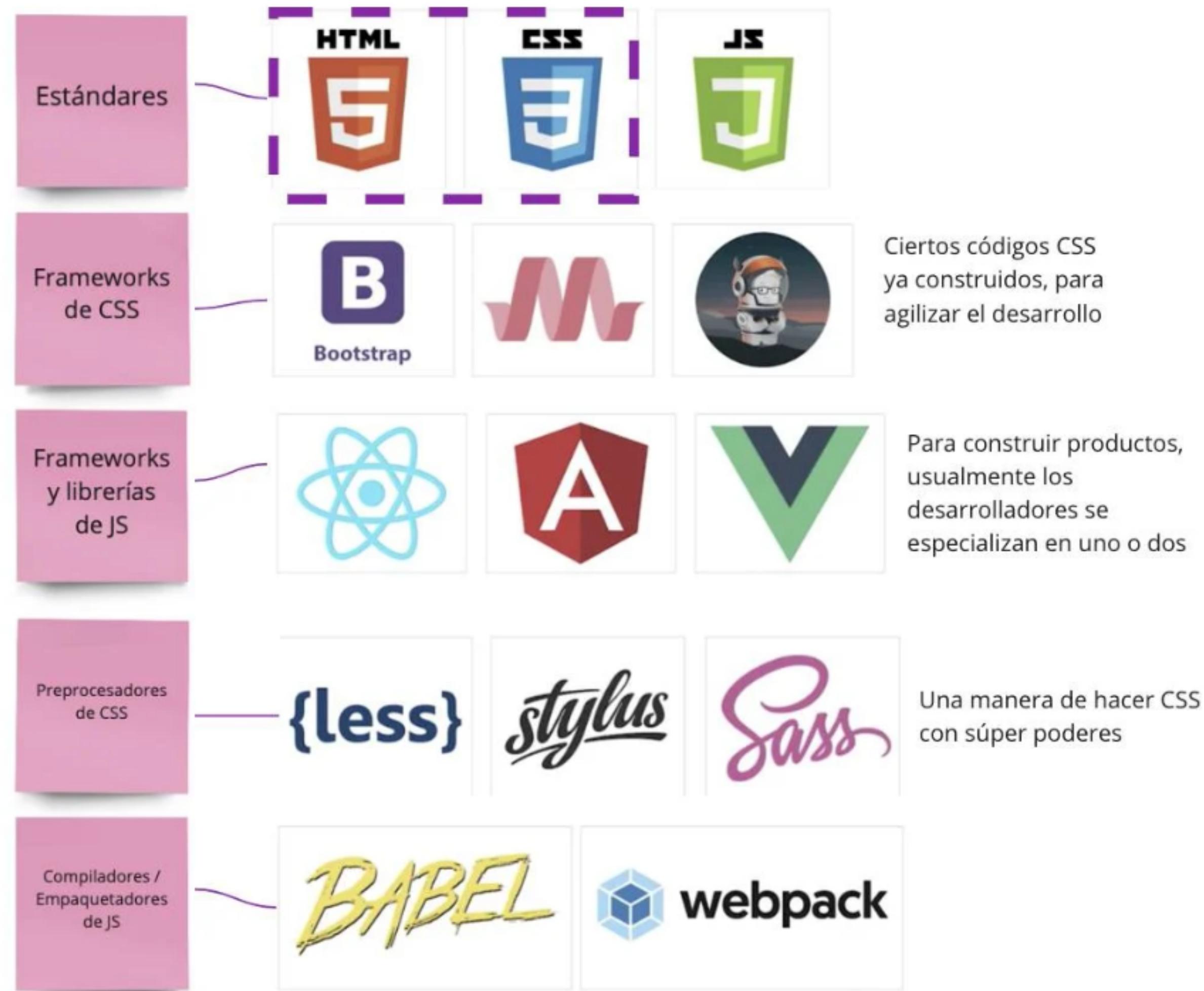
Atender y resolver **varios pedidos a la vez**.

Optimizar los **tiempo de respuesta (performance)**.



FULLSTACK







```
1 <div id="container">
2 <header>
3   <h1>Basic Structure Elements of an HTML Document</h1>
4 </header>
5 <dl>
6   <dt>&lt;!doctype html&gt;</dt>
7   <dd>The document type declaration is not really a part
     render the mark-up</dd>
8   <dt>&lt;html&gt;</dt>
9   <dd>The html tag is the container for all the rest of
10    <dt class="second-level">&lt;head&gt;</dt>
11    <dd>The head section contains the title, meta tags, ...
12    <dt class="third-level">&lt;title&gt;</dt>
13    <dd>The title tag is required and provides a title for
     <ul>
14      <li>Defines the title in the browser bar</li>
15      <li>provides a name for the page when bookmarked</li>
16      <li>Displays the name of the page in search results</li>
17    </ul>
18  </dd>
19  <dt class="second-level">&lt;body&gt;</dt>
20  <dd>The body is where all the content of a web page
21    <ul>
22      <li>The body tag is the main content area of a web page.
        It can contain text, images, styles, and any other non-...
23    </ul>
24  </dd>
25 </dl>
```

Frameworks

Infraestructura

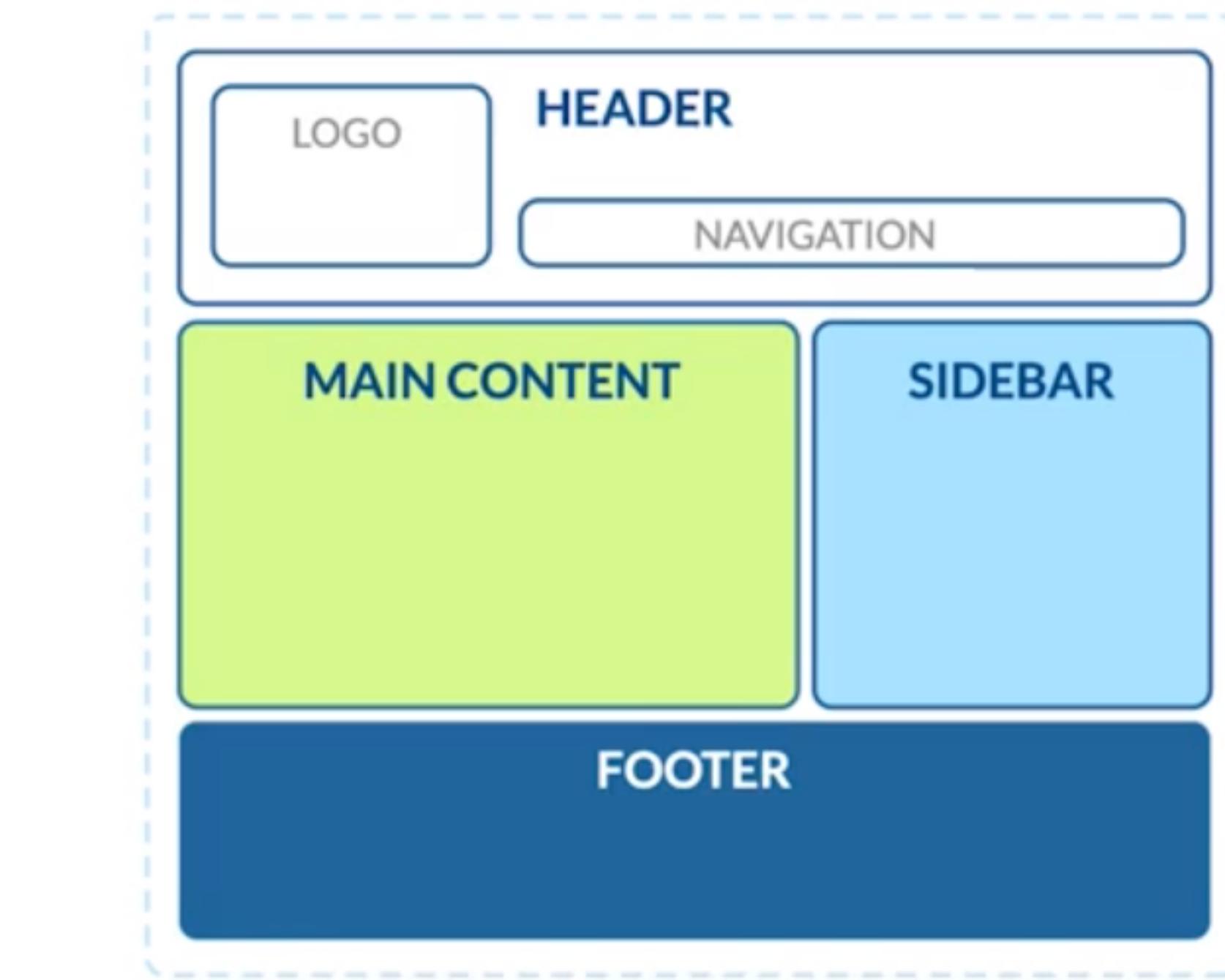
A diferencia del FRONT-END no tiene estándares, tiene más tecnologías y diferentes lenguajes de programación, no es necesario que sepas todos, para ser back-end puedes serlo con un solo lenguaje pero es un plus cuando sabes más de uno

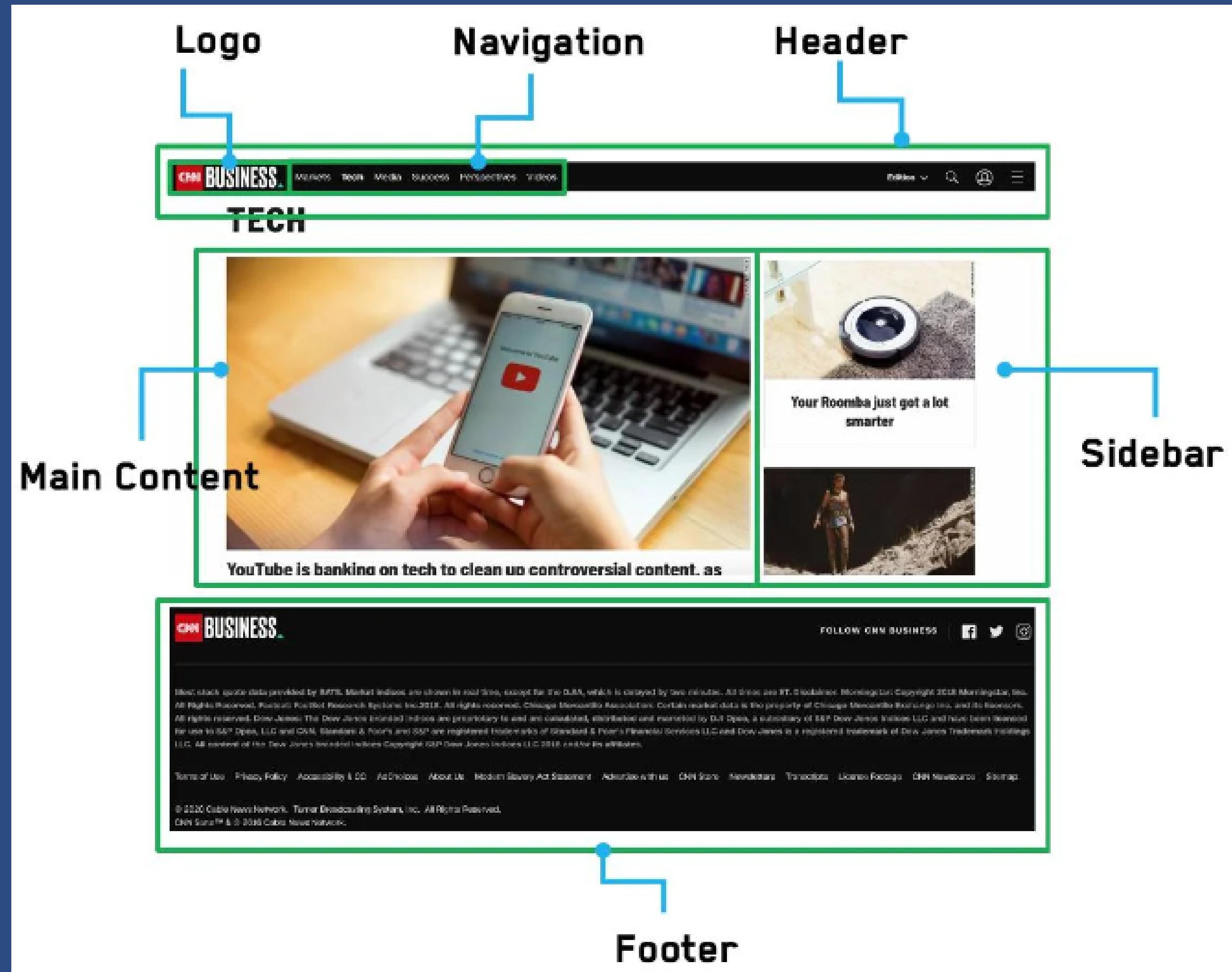
Dónde va a subir el desarrollo de su app, el BACK-END debe entender cómo funciona y como se conecta con el FRONT-END

Ayudan a trabajar de manera más rápida ciertas especificaciones según el lenguaje utilizado



CONTAINER







Anatomía de una Etiqueta HTML

Elemento

