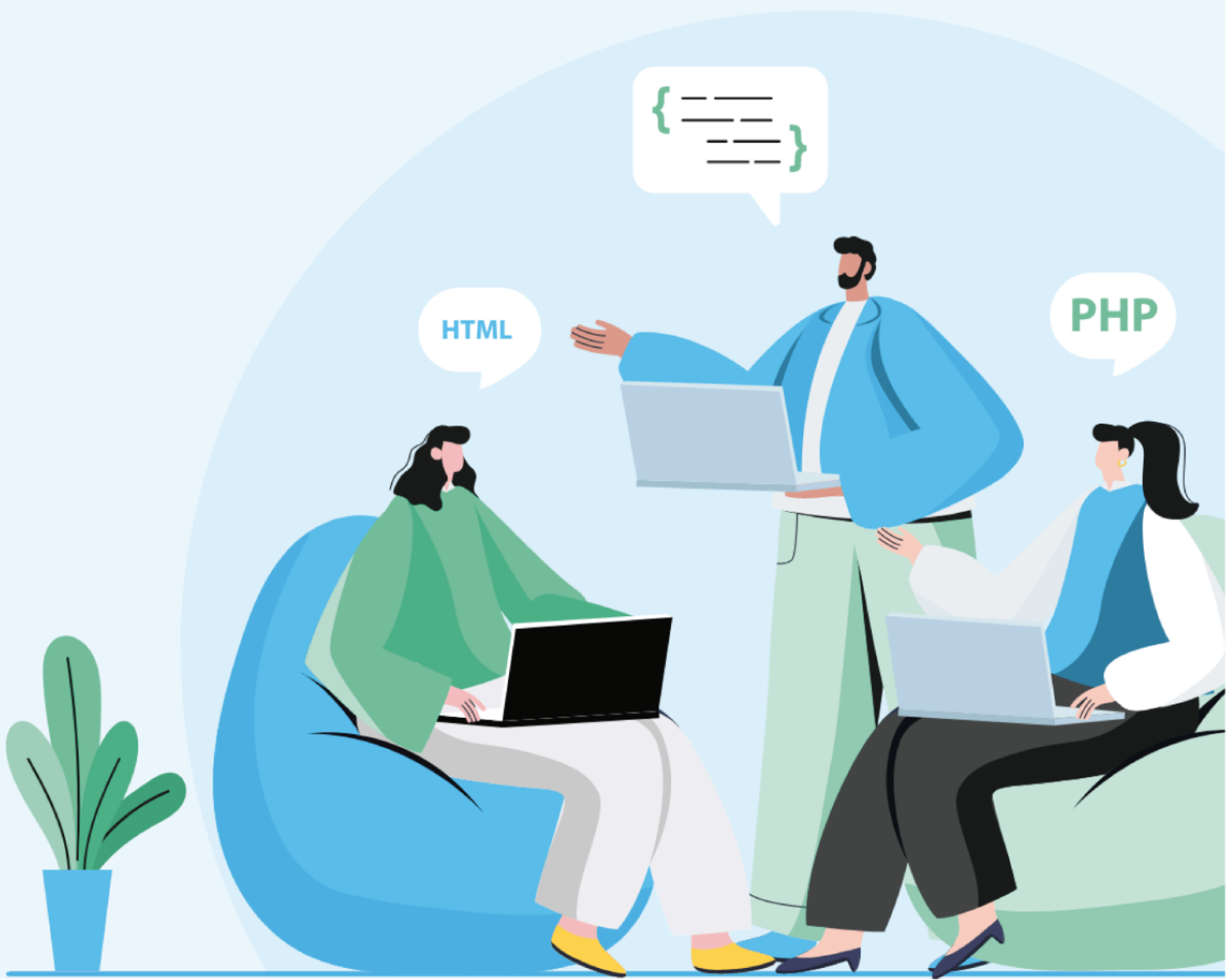




Skill Up Lab - React

Unidad 3





Unidad 3: Armado de la estructura básica

Antes de adentrarse en esta unidad

Para que puedas aprovechar mucho más el contenido presente en esta unidad, te recomendamos que antes de avanzar tengas muy en claro lo siguiente:

- cómo hacer llamados a una API con React y la librería AXIOS
- el motivo por el cual necesitamos de un Token dentro de la aplicación y el lugar donde almacenamos el mismo
- el mecanismo de validación de datos de un formulario
- cómo enviar datos de un formulario al endpoint de una API
- cómo se implementa la librería React Router Dom dentro de una aplicación de React

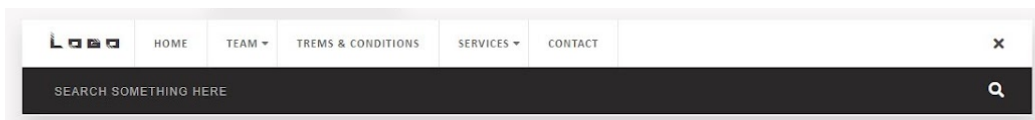
Objetivos de la unidad

Con el estudio de esta unidad buscamos que:

- Puedas armar el componente Cabecera en donde dispondrás todo lo referente a la barra de navegación de la aplicación.
- Elabores un componente PieDePagina en donde deberás disponer los créditos y los elementos claves que deben estar presentes en el footer de una aplicación
- Integres los componentes anteriores con los demás componentes y permitas que hagan parte del flujo de toda la aplicación
- Implementes una librería de CSS para comenzar a gestionar los estilos visuales de la aplicación



Clase 1: Armado del Encabezado y barra de navegación



¿Para qué nos servirá este componente?

Este componente tiene como finalidad contener una barra de navegación que esté presente a lo largo de toda la aplicación. A su vez, servirá facilitar el ingreso a las distintas rutas que han sido configuradas y en las cuales podremos ver la información esperada.

¿Por qué es importante contar con una barra de navegación?

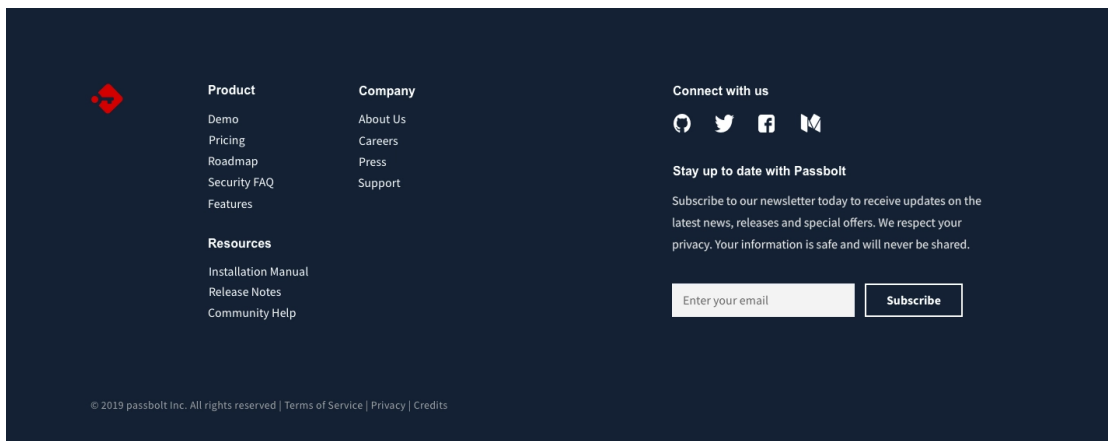
Las barras de navegación en las aplicaciones web son algo muy común, ya que como lo mencionamos anteriormente, le permiten al visitante de la página, acceder a cualquier ruta que esté disponible. Además, al implementar una barra de navegación no solamente estaremos aprendiendo cómo generar la estructura a través de JSX sino que también entenderemos cómo funciona el sistema de ruteo dentro de una SPA (single page application).

¿El encabezado y la barra de navegación pueden tener lógica condicional?

Entendiendo que la lógica condicional nos va a permitir mostrar o no algunas porciones de contenido y que particularmente estamos hablando de la implementación de algunos IF/ELSE, la respuesta sería: claro que sí. Esto se debe a que particularmente ese será uno de los objetivos de esta unidad, pues de esa manera podrás aprender a implementar partes de la barra de navegación que solamente estén presentes si el visitante de la aplicación se ha autenticado correctamente.



Clase 2: Armado del pie de página



¿Es necesario que toda aplicación posea un pie de página?

No hay nada escrito al respecto y bien esto podría tener una respuesta bastante extensa. Sin embargo, para darte una respuesta más breve, la sugerencia será que siempre implementes un pie de página, así sea bastante sencillo, pues esta parte de contenido le podrá dar un "cierre oficial" a los demás bloques de contenido presentes dentro del documento. Además, si lo piensas un poco más, puedes aprovechar este espacio para disponer allí información no tan relevante, pero que sí estaría bueno tener presente constantemente.

¿Puedo poner una barra de navegación en el pie de página?

Por supuesto, cualquier información que le aporte valor al visitante del sitio puede ir dentro de este apartado de contenido, pues aquí podrás dejar disponibles aquellos accesos directos a las secciones más relevantes de tu aplicación, los enlaces a las redes sociales o cosas por el estilo que le aporten más información a tu sitio web.

¿El pie de página se desarrolla como cualquier otro componente?

Totalmente. Pues de esta manera podrás tener de manera aislada todo su contenido y de ser necesario en un futuro, podrás modificar el mismo, sin afectar a otros apartados de la aplicación.



Clase 3: Integración de encabezado y la barra de navegación



¿Cómo se integra este componente con el resto de la aplicación?

De la manera más sencilla posible, pues no debemos dejar de pensar que este es un componente más como cualquier otro. Y para implementarlo lo deberemos tratar como tal.

¿Por qué nos conviene hacerlo como un componente externo en vez de implementarlo directamente dentro del componente principal de la aplicación?

Siempre que pensemos en que un apartado de código de la aplicación deberá estar presente a lo largo de la misma, es un buen indicador de que esto deberá ser tratado como un componente aparte, pues de esta manera podremos procurar un eficiente desarrollo del componente, pensando incluso en su futura escalabilidad.

La NO integración de este componente, ¿pone en riesgo el resto de la aplicación?

Realmente no, pero siendo sensatos, la mejor manera de permitir que las personas que usan nuestra aplicación puedan acceder a cualquier parte de



la misma, es haciendo uso justamente de un encabezado que contenga una llamativa y contundente barra de navegación.

Clase 4: Integración de la librería de CSS



¿Qué es una librería de CSS?

Quizás puede sonar un poco tonta la pregunta, pero recuerda: no hay pregunta tonta, lo tonto es no preguntar.

Una librería de CSS es una serie de archivos que ya vienen preparados para ser usados directamente, sin pensar en escribir siquiera una sola línea de código CSS.

Existen varias librerías de reconocida fama en el mercado. Quizás te suenen los nombres de Bootstrap, Tailwind o Bulma ¿no? Pues bien, estamos seguros que en algún momento de tu vida, te habrás encontrado con alguna de ellas.

¿Cual es la ventaja de usar una librería de CSS?

La principal ventaja es la rapidez con la que podemos construir nuestra aplicación, pensando particularmente siempre en el código funcional, más no en el visual. Además, gracias a las librerías de CSS es que podemos implementar en muchas ocasiones efectos súper llamativos que de otra manera nos hubiese costado bastante esfuerzo llegar a implementar.



No me gustan las librerías de CSS ¿puedo implementar mi propio código CSS dentro de React?

Claro que sí. Y no necesitas ser un experto para hacerlo. Es más, a lo largo de esta clase te vamos a mostrar como puedes, además de usar una librería, implementar tu propio CSS dentro de los componentes de React. Para que de esa manera pueda ponerle un poco de tu impronta a los estilos visuales de la aplicación

Enlaces y documentación

- [Bootstrap CSS](#): documentación oficial de esta gran librería de CSS. Estamos seguros que ya la conoces, pero nunca está de más contar con el enlace a la mano.
- [Tailwind CSS](#): documentación oficial de esta genial librería que le viene comiendo los talones a Bootstrap. Si bien su uso difiere un poco de esta última, Tailwind se viene consolidando como una de las mejores librerías actualmente en el mundo del desarrollo de aplicaciones web.
- [Sass - Preprocesador de CSS](#): si eres fanático del CSS y quieres llevar su potencial mucho más allá, no puedes dejar de leer esta documentación oficial, pues aquí vas a encontrar todas aquellas cosas que hacen de este preprocesador una de las mejores herramientas al momento de escribir CSS por tu cuenta.
- [Styled Components](#): si lo tuyo es escribir todo a la "React Way" esta librería de NPM para trabajar con componentes de estilo es lo mejor que podrás encontrar, pues implementa toda la lógica de un componente de React, con la virtud de poner implementar estilos CSS de una manera mucho más lógica y funcional.

