

[Entregas](#) / Master Front End VIII - Laboratorio - Módulo 4 - Frameworks - Vue

Master Front End VIII - Laboratorio - Módulo 4 - Frameworks - Vue

Laboratorio Vue.js

El objetivo de este ejercicio es el que te familiarizes con VueJS.

Deberás partir del repositorio en la carpeta `./00_start` en el proyecto <https://github.com/Lemoncode/vuejs-exercise>

Este proyecto contiene el desarrollo de nuestro módulo de Vue.js.

Pasos para ejecutarlo

- Descarga el proyecto.
- Instala las dependencias.

```
npm install
```

- Arranca la aplicación y comprueba que funciona correctamente.

```
npm start
```

Sistema de Validaciones

Implementa un sistema de validaciones para el formulario de edición de recetas, de forma que no permita guardar una receta si no están completos los campos necesarios:

- Nombre de la receta.
- Al menos un ingrediente en la receta.
- Descripción de la receta.

Mejora del Layout de la aplicación

Utiliza imágenes en servidores gratuitos de internet o almacénalas en local en el repositorio para mostrar una imagen de la receta.

Pistas:

- Depende de la aproximación que quieras hacer, tal vez debas hacer uso de `require` en las url dinámicas. En las presentaciones había un ejemplo de cómo manejar imágenes dinámicas mediante rutas dinámicas.

Crea un componente Snackbar (Toast) y sustituye los mensajes de la aplicación

Vuetify tiene un componente [Snackbar](#) que realiza esta función. Deberás sacarlo a común y utilizarlo cuando sea necesario.

Pistas:

- Puedes hacerlo mediante la directiva `slot` o pasar las propiedades necesarias como `props`.

Actualiza la lista de recetas

Puedes utilizar el componente `v-data-table` de [Vuetify](#)

Desafíos para subir nota

CRUD

Crea la funcionalidad necesaria para eliminar y crear nuevas recetas desde la lista de recetas.

Componente Logout y rutas privadas

- Persiste el usuario en el `local storage`.
- Muestra su nombre en la toolbar de la aplicación.
- Crear un componente de Logout (servirá un botón en la toolbar).
- Configura el enrutador para que admita unas rutas públicas y privadas.
- Actualiza el router de modo que solicite estar logueado para poder entrar a rutas distintas del `root` y `login`.

Pistas:

- Una aproximación para realizar esto en el router:

`./src/router.ts`

```
...  
export const router = new Router({  
  routes,  
});  
  
// beforeEach es una guarda de Vue-Router que SIEMPRE debe devolver la invocaci3n  
router.beforeEach((to, from, next) => {  
  // 1. Definir rutas p3blicas  
  // 2. Definir rutas que necesitan autorizaci3n  
  // 3. Saber si el usuario est3 autenticado  
  
  // 4. Retornar next("/login") si NO est3 autenticado o next() si SÍ est3 autent  
});
```

Entrega:

Enlaces repositorios

GUARDAR



© Lemoncode 2021 - Todos los derechos reservados

