

Introducción

En este proyecto se unifica todo lo aprendido hasta ahora. Los alumnos desarrollarán un e-commerce donde los usuarios se registrarán, logearán , verán productos y crearán pedidos utilizando la API del e-commerce que crearon en el 1º Proyecto de backend.



Requisitos

Una vez analizadas las necesidades del proyecto, se espera que el alumno desarrolle un frontend que sea capaz de lo siguiente:

- Registro de usuarios.
- Login de usuarios.
- Que se puedan ver los productos y añadir al carrito de compra
- Que se pueda crear pedidos
- Que en tu perfil puedas ver tus datos
- Logout

Requisitos imprescindibles del proyecto:

- Uso de ramas con git, cuando se termine el proyecto deberán quedar dos ramas la master o main y la develop.
- Presentación de **README excelente**.

1.1.E-commerce(tienda online)

Componentes mínimos:

- ☐ Register
- ☐ Login
- ☐ Home
- ☐ Products
 - Product
- ☐ Cart
- ☐ Perfil. Vista perfil con los datos del usuario logeado y sus pedidos
- ☐ Header
- ☐ Footer

Implementa React Router en tu página

- ☐ /home. Home de la app
- ☐ /login
- ☐ /register
- ☐ /profile.
- Utilizar Context
- Readme Excelente
- Uso de SASS
- Importante el diseño

Reglas:

- Los componentes no podrán sobrepasar las 400 líneas de código.
- Las funciones no deberán sobrepasar las 75 líneas de código.

Para poder hacer peticiones a vuestra API teneis que instalar el módulo de CORS:

1. En el repo del backend: npm i cors
2. En el index.js:
 - a. `const cors = require("cors")`
 - b. `app.use(cors())`

1.2. Extras

- Buscador de productos
- Implementación de filtros, por ejemplo que se pueda filtrar un producto por precio
- Vista Admin donde se puede hacer el CRUD de productos (solamente si tienes el role Admin)
- El usuario con rol Admin tenga una opción en el nav que le lleva a la vista Admin
- CRUD de los productos
- Añadir o quitar un producto de favoritos
- Que el usuario pueda añadir reviews a un producto
- Que solo puedas editar y eliminar las reviews que tu creas.
- El usuario puede subir fotos en las reviews o cambiar su foto de perfil.
- El usuario puede dar likes a las reviews de los productos.
- El usuario en su perfil puede ver sus pedidos
- Implementación de guards
- Que sea responsive

1.3. Entregables

- Subir el proyecto a un repositorio en github
- Dejar el enlace del repositorio adjuntado en Classroom