

**Time: 10 days**

**Team: 2 or 3**

**GitHub - each task has a branch**

**Bonus: Deploy on Github pages or Vercel**

**<https://vercel.com/docs/deployments/overview>**

### **Idea 1:**

Building an eCommerce project using React:

Requirements:

Bonus: USE MaterialUI instead of Bootstrap

DON'T USE JavaScript ALERT

List → Pagination

Delete → Confirmation

Form → Validation

Localization → React Localization

#### 1. User Authentication:

- User registration and login. (save data in local storage)
- Validation (on registration and login)
- if the user needs to add products to his cart he must **LoggedIN**, if he didn't log in he can browse

#### 2. Admin Panel:

- If the User is an admin he can add, edit, or delete the products → **API (post, put/patch, delete)**

#### 3. Product Management:

- call API to list products.
- add pagination to the list
- Product details, including images, descriptions, and prices. (call API to view product)

#### 4. Shopping Cart:

- Add, remove, and update items in the cart.
- Calculate cart totals and taxes.
- before remove must have a confirmation message

#### 5. Wishlist and Favorites: (using redux)

- Allow users to create and manage wishlists.
- Save favorite products for later

## 6. Search and Filtering:

- Search products by name
- Filter the products by price and category

## 7. Responsive Design:

- Ensure the application is accessible and usable on various devices and screen sizes.(small, medium, large)

## Idea 2:

Building an Elearning project using React:

Requirements:

USE MaterialUI instead of Bootstrap

DON'T USE JavaScript ALERT

List → Pagination

Delete → Confirmation

Form → Validation

Localization → React Localization

### 1. User Authentication:

- User registration and login. (save data in local storage)
- Validation (on registration and login)
- if the user needs to join courses he must LoggedIN, if he didn't log in he can browse

### 2. Admin Panel:

- If the User is an admin he can add, edit, or delete the courses → API (post, put/patch, delete)

### 3. Courses Management:

- call API to list courses.
- add pagination to the list
- Course details, including images, descriptions, and prices. (call API to view courses)

### 4. Dashboard For User:

- List All Courses that he joined
- He can Unjoind Courses.
- before Unjoind must have a confirmation message

### 5. Wishlist and Favorites: (using redux)

- Allow users to create and manage wishlists.
- Save favorite Courses for later

### 6. Search and Filtering:

- Search courses by name

- Filter the courses by price and category

#### 7. Responsive Design:

- Ensure the application is accessible and usable on various devices and screen sizes.(small, medium, large)