Frontend Developer Assignment – Product Dashboard

Objective

Build a product management interface using **React** that connects to a public API to display, manage, and filter a list of products.

Core Requirements

Fetch & Display Products

- Fetch product data from:
  `https://mock-data-josw.onrender.com/products`
- Display the products with the following fields:
  - `name`, `price`, `category`, `rating`

Example Product Response:

```
{
  "id": 99,
  "name": "Veggie Burger Patties",
  "description": "Delicious veggie burger patties for grilling or frying.",
  "price": 5.99,
  "category": "Food - Meat Substitutes",
  "rating": 4.3
}
```

CRUD Operations

Use the endpoints below for product management:

```
GET     https://mock-data-josw.onrender.com/products
POST    https://mock-data-josw.onrender.com/products
PUT     https://mock-data-josw.onrender.com/products/:id
DELETE  https://mock-data-josw.onrender.com/products/:id
```

- Create new products
- Edit existing products
- Delete products

Additional Features

★ **Favorites**

- Allow users to **mark/unmark** products as favorites
- Persist favorites **across sessions** using `localStorage` or libraries like `redux-persist`

Search & Filter

- Search products by **name**
- Filter products by **category** (from: `https://mock-data-josw.onrender.com/categories`)
- Sort products by **price** (Low → High, High → Low)

Modal Details

- Clicking on a product row should open a **modal** showing:
  - name, price, description, category

Pagination

- Implement pagination to manage large product lists

Toast Notifications

- Show toast messages for actions like:
  - Successful creation, update, or deletion of a product
  - Errors during API calls

Optional (Bonus)

- Use a **state management** library like **Redux**, **Zustand**, or **Context API**
- Responsive design for mobile and desktop
- Show **loading indicators** and **error messages** during API calls

---

Tech Stack

- **Required**: React
- **Optional**: Redux, Tailwind CSS, Material UI, Shadcn UI, Zustand, etc.

Deadline

Submit your completed project within **7 days** of receiving this brief.

Submission

- Submit a **GitHub repository link**
- Submit the **hosted project link** (e.g., Render deployment)
- Include a `README.md` file with:
  - Setup instructions (how to run it locally)
  - Any assumptions or decisions made
  - Navigation flow (e.g., "Click Add Product to create a new entry...")
  - What you'd improve with more time