PROBATION TASK 3: Dynamic Product Gallery with Category Filtering & Live Search

**Task Overview**

Develop an interactive, animated e-commerce product showcase website using **React**. The project will emphasize smooth animations, interactivity, and efficient data fetching and state management using **Redux**. Ensure compatibility with modern web browsers, and regularly commit to a GitHub repository to document development progress.

**Project Guidelines**

1. **Project Scope**: Build a visually engaging, animated website with React, featuring product listings fetched from a JSON API, customizable with category selections and search functionality.
2. **Data Integration**:

* Fetch product and category data from the [DummyJSON API](https://dummyjson.com/docs) and display all available categories.
* Implement single-selection functionality for categories, allowing users to filter products based on the selected category.
* If no category is selected, show products from all categories.

1. **Pagination**:

* Implement server-side pagination while fetching products, displaying 10 products per batch.
* Dynamically fetch and display the next batch as the user scrolls, until all products are retrieved (e.g., 1–10, 11–20).

1. **Search Functionality**:

* Enable users to search for products by name or description.
* Ensure selected categories and search queries are stored as query parameters, allowing direct link sharing with filtered results.

1. **System Architecture**:

* Structure files logically with React functional components, CSS, and JavaScript for animations.
* Utilize **Redux** to store and manage product and category data, search input, and selected category state.

1. **Component Interaction & Animations**:

* Link animations to user actions like scrolling, selecting categories, and inputting search queries.
* Ensure animations are implemented using only CSS and JavaScript.

**Technical Requirements**

1. Use only **React functional components**.
2. Store selected categories and search input as **query parameters** for shareable URLs.
3. Ensure clean UI and UX for an optimized browsing experience.

**Notes:**

Include any limitations in App.js as comments.

This approach will ensure a performant, interactive experience while meeting the functional and technical requirements using React and Redux.

**Project Duration**

- **Deadline- 2 NOV (5PM)** , adhering to all requirements.

**Submission Guidelines**

**1. GitHub Repository:**

- Create a repository with regular commits documenting progress.

- Use meaningful commit messages.

**2. Final Submission:**

- Include source code files and a README with instructions and deployment steps.

**3. Deployment:**

- Deploy the website, ensuring it’s accessible and browser-compatible.

- Include the live link in the README.

**Important Notes**

- Originality: All work must be unique and authentic.

- Browser Compatibility: Verify deployment on modern web browsers.

- GitHub Commits: Regular commits are required and evaluated.

**Resources**

<https://youtube.com/playlist?list=PLC3y8-rFHvwgg3vaYJgHGnModB54rxOk3&feature=shared>

<https://youtube.com/playlist?list=PLu0W_9lII9agx66oZnT6IyhcMIbUMNMdt&feature=shared>

Good luck with your project! Ensure originality, authenticity, and creativity throughout.