Key Workflow of Next.js Project

1. User Registration:

- User navigates to the registration page.
- User enters registration details (e.g., name, email, password).
- User submits the registration form.
- The frontend sends a POST request to the backend API to register the user.
- The backend API validates the input data.
- If valid, the backend API sends a request to Sanity to store the user data.
- Sanity stores the user data in the database.
- The backend API receives a confirmation from Sanity.
- The backend API sends a confirmation email to the user.
- The user receives the confirmation email.

2. Product Browsing:

- User navigates to the product browsing page.
- The frontend sends a GET request to the Sanity API to fetch product categories.
- Sanity API retrieves the product categories data.
- The frontend receives the product categories data.
- The frontend displays the product categories to the user.
- User selects a product category.
- The frontend sends a GET request to the Sanity API to fetch products within the selected category.
- Sanity API retrieves the product data.
- The frontend receives the product data.
- The frontend displays the products to the user.

3. Order Placement:

- User adds products to the cart.
- User navigates to the checkout page.
- User enters shipping and payment details.
- User submits the order.
- The frontend sends a POST request to the backend API to place the order.
- The backend API validates the input data.
- If valid, the backend API processes the payment (if applicable).
- The backend API sends a request to Sanity to store the order details.
- Sanity stores the order details in the database.
- The backend API receives a confirmation from Sanity.
- The backend API sends an order confirmation email to the user.
- The user receives the order confirmation email.

4. Shipment Tracking:

- User navigates to the shipment tracking page.

- The frontend sends a GET request to a third-party API to fetch the order status.
- The third-party API retrieves the shipment status data.
- The frontend receives the shipment status data.
- The frontend displays the shipment status to the user.
- (Optional) The user can refresh the shipment status to get updates.