

# Workflows For General E-Commerce Project On Next.js

## 1. User Registration:

- ❖ **Action:** User signs up via the registration form.
- ❖ **Workflow:**
  - User inputs details in the form.
  - Data is sent to the Next.js API.
  - Data is stored in Sanity CMS.
  - A confirmation email is sent to the user.

## 2. User Login:

- ❖ **Action:** User logs in via the login form.
- ❖ **Workflow:**
  - User inputs email and password.
  - Authentication is handled via a custom Next.js API or a third-party service.
  - Upon successful login, the user is redirected to the dashboard or homepage.

## 3. Product Browsing:

- ❖ **Action:** User browses products on the shop page.
- ❖ **Workflow:**
  - User selects a category or searches for products.
  - The frontend sends a request to the Sanity CMS API.
  - Product data is fetched and displayed dynamically on the frontend.

## 4. Product Details Viewing:

- ❖ **Action:** User clicks on a product to view details.
- ❖ **Workflow:**
  - The frontend fetches detailed product data using a Sanity CMS API.
  - Dynamic routes in Next.js display the product information.

## 5. Adding to Cart:

- ❖ **Action:** User adds items to their shopping cart.
- ❖ **Workflow:**
  - Product details (name, price, quantity, etc.) are stored in Local Storage.
  - The cart UI is updated to reflect the changes.
  -

## 6. Order Placement:

- ❖ **Action:** User proceeds to checkout.
- ❖ **Workflow:**
  - User fills in shipping and payment details.
  - Data is sent to the Next.js API.
  - Order details are stored in Sanity CMS.
  - Payment is processed through a payment gateway.
  - User receives an order confirmation email.

## 7. Shipment Tracking:

- ❖ **Action:** User tracks their order status.
- ❖ **Workflow:**
  - The frontend fetches tracking information from a third-party shipment API.
  - Real-time status updates are displayed to the user on the order tracking page.

## 8. Blog Browsing:

- ❖ **Action:** User explores blog posts.
- ❖ **Workflow:**

- The frontend sends a request to the Sanity CMS API.
- Blog data is fetched and displayed dynamically.
- Users can view individual blog posts through dynamic routes.

## 9. Contact Form Submission:

- ❖ **Action:** User submits the contact form.
- ❖ **Workflow:**
  - User inputs their message and details.
  - Data is sent to a Next.js API route.
  - An email is sent to the admin using a third-party service.
  - The user receives a confirmation of their message.

## 10.Favorite Management:

- ❖ **Action:** User adds or removes products from their favorites.
- ❖ **Workflow:**
  - Favorite product IDs are stored in Local Storage.
  - The favorites page fetches details for the stored IDs from Sanity CMS and displays them.
  -

## 11.Admin Panel (Optional):

- ❖ **Action:** Admin manages content through Sanity CMS.
  - ❖ **Workflow:**
    - Admin logs into the Sanity Studio.
    - Products, blogs, and other content are updated in real time.
    - Changes are reflected dynamically on the frontend.
-