

Technical Planning Outcomes Checklist

Technical Plan

1. **Tech Stack:**
 - Frontend: **Next.js**
 - Backend: **Sanity.io** (for content management)
 - Database: Sanity's structured content with GROQ queries.
 - Deployment: Vercel for hosting.
2. **Core Features:**
 - User Authentication (optional for buyers and sellers).
 - Product Listings (categories, prices, images).
 - Search & Filters (based on price, category, location).
 - Cart and Checkout.
 - Seller Dashboard for managing inventory.
3. **Integrations:**
 - Payment Gateway (Stripe, PayPal).
 - Image Optimizer (Sanity and Next.js Image Component).

Workflow

1. **Planning Phase:**
 - Define features (MVP: Product listing, search, and cart).
 - Set timelines for design, development, and testing.
2. **Development Phase:**
 - Set up Sanity schema.
 - Connect Sanity with Next.js frontend using APIs.
 - Build UI components (e.g., ProductCard, Filters, Cart).
 - Implement routing using Next.js dynamic routes.
3. **Testing Phase:**
 - Verify API responses and edge cases.
 - Test responsiveness and performance.
4. **Deployment:**
 - Deploy backend (Sanity Studio).
 - Deploy frontend on Vercel.

API Requirements

1. **Product Management APIs:**
 - GET: Fetch product listings.

- POST: Add a new product.
- PUT: Update product details.
- DELETE: Remove a product.
- 2. **Search and Filter APIs:**
 - Query by category, price, and availability.
 - Text search for product names/descriptions.
- 3. **Cart and Checkout APIs:**
 - POST: Add to cart.
 - DELETE: Remove from cart.
 - POST: Process payment.

Sanity Schemas

Product

- title (string)
- description (text)
- price (number)
- category (reference → Category)
- image (image)
- available (boolean)

Category

- name (string)
- description (text)

User

- username (string)
- email (string)
- isSeller (boolean)