Technical Foundation for SNKRS.io

System Architecture

1. **Frontend (Next.js):** A responsive interface that supports seamless browsing and personalization.
2. **Sanity CMS:** Manages customizable products, customer profiles, and order history.
3. **Third-Party APIs:**
   - Payment Gateway (Stripe/PayPal/Local): Handles "Try Now, Pay Later" and installment plans.
   - AR Visualization API: Supports virtual try-on features.
   - Logistics API: Tracks deliveries and returns.

Key Workflows

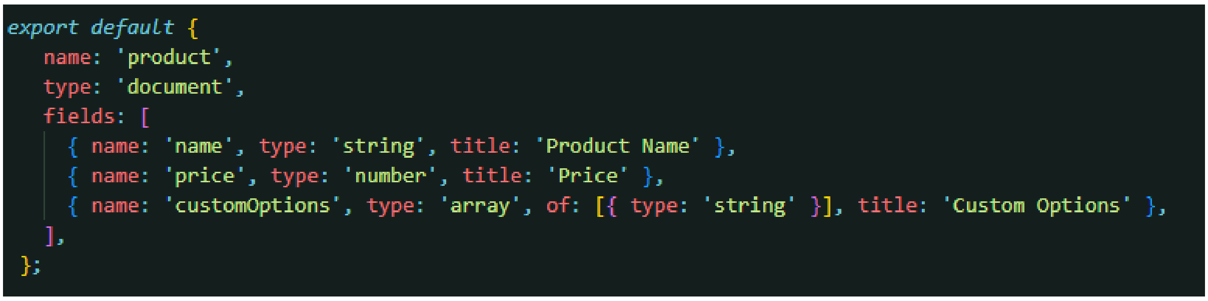
1. **User Registration:**
   - Input user data -> Validate -> Store in Sanity CMS.
2. **Customization Process:**
   - User selects style -> Custom options dynamically fetched from Sanity -> Product preview generated.
3. **Order Placement:**
   - Products added to the cart -> API validates payment -> Order stored in Sanity CMS.
4. **AR Try-On (Future):**
   - User accesses AR feature -> API fetches product dimensions -> Virtual preview displayed.

API Requirements

- **Endpoint:** `/custom-products`
  - **Method:** GET
  - **Description:** Fetch all customizable options for selected products.
  - **Response Example:** `{ "id": 101, "color": ["red", "blue"], "size": [6, 7, 8] }`
- **Endpoint:** `/order-status`
  - **Method:** GET
  - **Description:** Fetch delivery status.
  - **Response Example:** `{ "orderId": 123, "status": "In Transit" }`

Sanity Schema Example

- **Product Schema:**



```
export default {
  name: 'product',
  type: 'document',
  fields: [
    { name: 'name', type: 'string', title: 'Product Name' },
    { name: 'price', type: 'number', title: 'Price' },
    { name: 'customOptions', type: 'array', of: [{ type: 'string' }], title: 'Custom Options' },
  ],
};
```