

1. Define Technical Requirements

a. Frontend:

- User Friendly UI/UX for browsing products dynamically
- Responsive design for mobile, tablet and laptop screens.
- **Pages:** Home – About – Products – Product Details – Cart – Checkout – Order Confirmation – Contact us – FAQs
- **Tools:**
 - Vs Code (for writing program)
 - Git-hub (Version control)
 - Vercel (Deployment)
- **Framework:** Next.js
- **Styling:** CSS / Tailwind CSS / React icons

b. Backend:

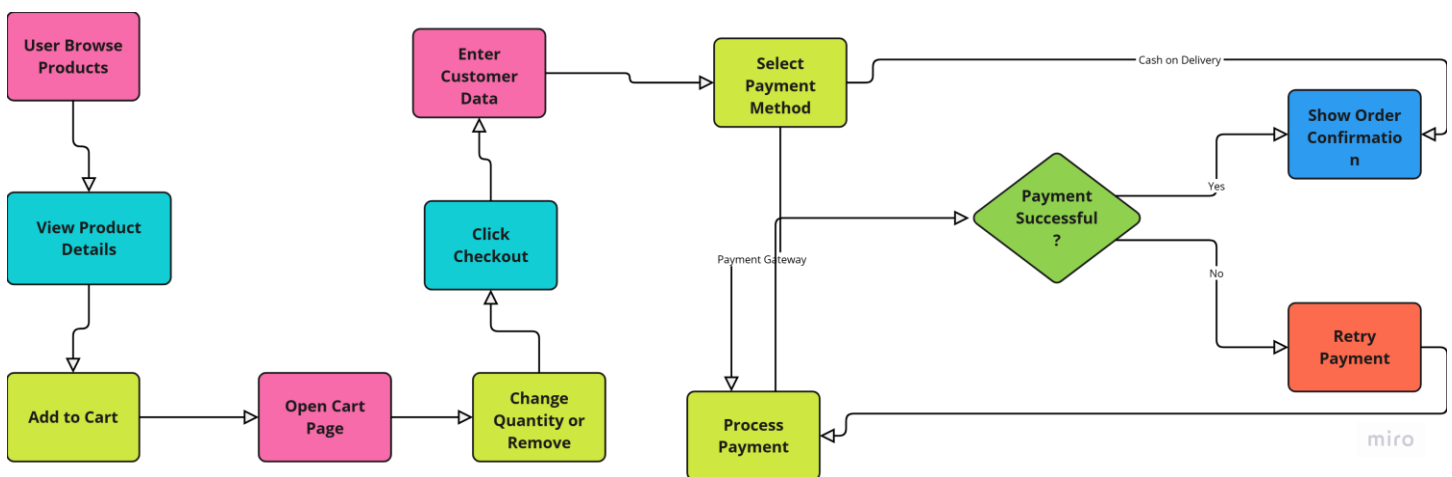
- **Sanity CMS:** Products, Orders and Customer's data Management

c. Third party APIs:

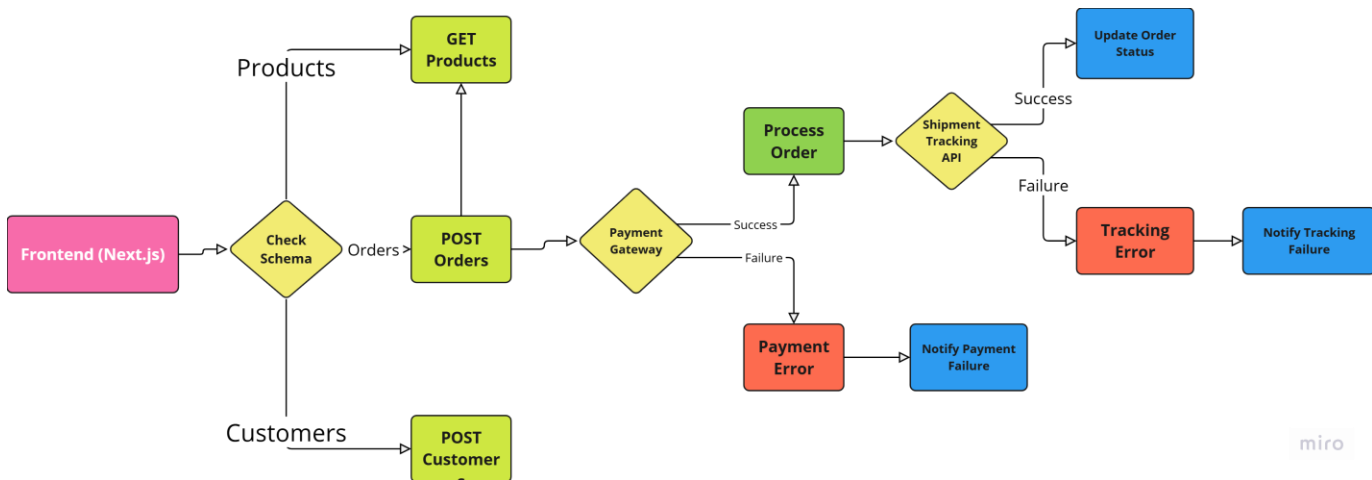
- **Shipment:** Leopard / TSC
- **Payment gateway:** easypaisa / jazzcash

2. Define System Architecture

a. Buying process (WorkFlow):



b. Technical Architecture:



miro

3. Workflow:

- User browse products and check products detail.
- User add product to cart by clicking on cart button.
- Product is added in the cart, user can open cart page to see items.
- User can change product quantity or remove product from cart.
- After selection user click on checkout button to start order process. Checkout page opens up and show a form to add customer data including (name, email, address, phone, payment method) and shows total amount of order.
- User add data, set payment method, if selected (Cash on delivery), order confirmation message is shown, if selected any payment gateway (jazzcash, easypaisa etc...) it leads to payment process in that gateway. Which is connected through API.
- After successful payment, shows order confirmation message.

4. Plan API requirements

- Endpoint name: /products
 - **Method:** GET
 - **Description:** Fetch all available products from Sanity
 - **Response:** Product details (ID, name, price, stock, image, color, description).
- Endpoint name: /customers
 - **Method:** POST
 - **Description:** Get customer's data and add it in sanity
 - **Response:** customer id, name, email, phone, address, order history
- Endpoint name: /orders
 - **Method:** POST
 - **Description:** Create a new order in Sanity.
 - **Response:** Customer info, order date, product details, total amount, payment status.
- Endpoint name: /shipment
 - **Method:** GET
 - **Description:** Track order status via third-party API
 - **Response:** Shipment ID, order ID, status, expected delivery date, carrier, tracking number

- e. **Endpoint name:** /payments
 - **Method:** GET
 - **Description:** collect payment from multiple payment gateways or pay cash on delivery
 - **Response:** payment id, order id, customer id, payment method, status, transaction id, payment gateway

5. Sanity Data Schema for all endpoints:

a. Products:

export default

```
{ name: 'product', type: 'document', fields: [
  { name: 'id', type: 'number', title: 'Product Id' },
  { name: 'name', type: 'string', title: 'Product Name' },
  { name: 'price', type: 'number', title: 'Price' },
  { name: 'stock', type: 'number', title: 'Stock Level' },
  { name: 'color', type: 'string', title: 'Color Options' },
  { name: 'description', type: 'string', title: 'Description' },
]
```

b. Customers:

export default

```
{ name: 'customer', type: 'document', fields: [
  { name: 'id', type: 'number', title: 'Customer Id' },
  { name: 'name', type: 'string', title: 'Customer Name' },
  { name: 'email', type: 'string', title: 'Email' },
  { name: 'phone', type: 'number', title: 'Phone Number' },
  { name: 'address', type: 'string', title: 'Address' },
]
```

c. Orders:

export default

```
{ name: 'order', type: 'document', fields: [
  { name: 'id', type: 'number', title: 'Order Id' },
  { name: 'customerId', type: 'number', title: 'Customer Id' },
  { name: 'date', type: 'date', title: 'Order Date' },
  { name: 'amount', type: 'number', title: 'Total Amount' },
  { name: 'productID', type: 'number', title: 'Product Id' },
  { name: 'address', type: 'string', title: 'Delivery Address' },
]
```

d. Shipment:

export default

```
{ name: 'shipment', type: 'document', fields: [
  { name: 'id', type: 'number', title: 'Shipment Id' },
  { name: 'orderId', type: 'number', title: 'Order Id' },
  { name: 'status', type: 'string', title: 'Order Status' },
  { name: 'date', type: 'Date', title: 'Delivery Date' }
]
```

```
{ name: 'customerId', type: 'number', title: 'Customer Id'},  
{ name: 'tracking', type: 'number', title: 'Tracking Number'},  
{ name: 'address', type: 'string', title: 'Delivery Address'},  
]
```

e. Payment:

export default

```
{ name: 'payment', type: 'document', fields: [  
  { name: 'id', type: 'number', title: 'payment Id' },  
  { name: 'customerId', type: 'number', title: 'Customer Id' },  
  { name: 'date', type: 'date', title: 'Payment Date' },  
  { name: 'amount', type: 'number', title: 'Paid Amount' },  
  { name: 'status', type: 'string', title: 'Payment Status'},  
  { name: 'method', type: 'string[]', title: 'Payment Method'},  
  { name: 'gateway', type: 'string', title: 'Payment Gateway'},  
]
```