**E - Commerce Website Architecture**

Establishing the technical groundwork for a furniture e-commerce site with Next.js 14 requires careful planning to guarantee scalability, optimal performance, and long-term maintainability.

Key factors such as efficient architecture, seamless user experience, and robust backend integration must be considered to support growth and ensure reliability.

## KEY FEATURES

1. Frontend Development
2. Backend Development
3. Payment Gateways
4. Authentication
5. Deployment
6. Shiping and Delivery
7. User Experience [UX] Features

**TECHNICAL REQUIREMENTS**

## FRONTEND

1. App Routing

Defines navigation paths for different pages like Home, Products, and Contact.

2. Dynamic Routing

Generates URLs dynamically, e.g., /product/:id for product-specific pages.

3. Styling (Responsive Design)

: Use Tailwind CSS, ShadCN ,Framer Motion for responsive designandspeed.

4. Dynamic Rendering

Renders content dynamically based on user interactions or data from APIs.

5. State Management

Manages user data (e.g., cart, filters) using tools like Redux or Context API.

6. Image Optimization

Improves performance with responsive, lazy-loaded, and optimized images.

7. Search and Filtering

Allows users to search and filter products efficiently.

# BACKEND:

1. Content Management

Allows admins to create, update, and manage furniture items, categories, and product details.

2. Structured Content

Stores data in a structured and flexible way using GROQ queries for easy retrieval.

3. Real-Time Updates

Provides real-time updates to frontend when content changes in Sanity.

4. API Integration

Sanity delivers content via its automatically generated, scalable APIs for easy frontend consumption.

5. Image Management

Optimizes and delivers images with built-in support for transformations and responsive formats.

6. vast services

Handles large-scale content efficiently, suitable for growing furniture websites.

7. Hosting and Deployment

Sanity Studio (backend dashboard) is hosted on the web, while its data is accessible via APIs.

# Third-Party API Integration

1. Shipment Tracking APIs

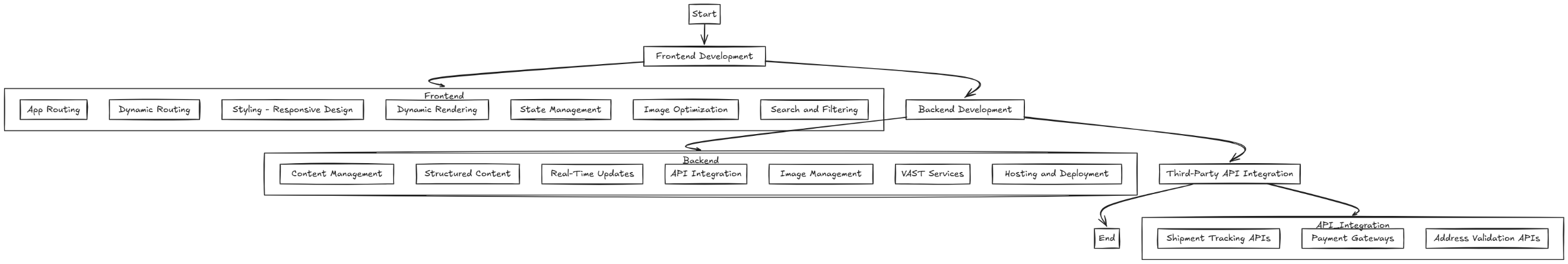
Integrates services like Shippo, shipengine for real-time shipment tracking and delivery updates.

2. Payment Gateways

Uses APIs like Stripe, PayPal, to handle secure online payments and transactions.

3.Mock APIs

**DESIGN SYSTEM ARCHITECTURE**

SHHHHHHHHHHHHHHHHHHHHHHHHDSKJ

DSSJJJJJJJJJJJJ

DDDDDDDDDDDD

**Sanity Schema For E-Commerce Furniture Website.**

# ***Product Schema :***

export default {

name: 'product’,

title: ‘Product’,

type: 'document',

fields: [

{

Name:’title’,

title: 'title',

type: 'string',

},

{

name: 'slug',

title: 'Slug',

type: 'slug',

options : {

source: 'title',

maxLength: 96,

}

},

{

name: 'description',

title: 'Description',

type: 'text',

},

{

name: 'price',

title: 'Price',

type: 'number',

},

{

name: 'category',

title: 'Category',

type: 'reference',

to : [{ type : 'category' }],

},

{

name: 'images',

title: 'Images',

type: 'array',

of : [{ type : 'image' }]

},

{

name: 'stock',

title: 'Stock',

type: 'number',

},

{

name: 'featured',

title: 'Featured',

type: 'boolean',

},

]

};

# *Category Schema :*

export default {

name: 'category',

title: 'Category',

type: 'document',

fields: [

{

name: 'title',

title: 'title',

type: 'string',

},

{

name: 'slug',

title: 'Slug',

type: 'slug',

options : {

source: 'title',

maxLength: 96,

}

},

{

name: 'description',

title: 'Description',

type: 'text',

},

]

};

# *User Schema :*

export default {

name: 'user',

title: 'User',

type: 'document',

fields: [

{

name: 'name',

title: 'Name',

type: 'string',

},

{

name: 'email',

title: 'Email',

type: 'string',

},

{

name: 'address',

title: 'Address',

type: 'text',

},

{

name: 'orders',

title: 'Orders',

type: 'array',

of: [{ type: 'reference', to: [{ type: 'order' }] }]

},

],

};

# *Order Schema :*

export default {

name: 'order',

title: 'Order',

type: 'document',

fields: [

{

name: 'orderId',

title: 'Order ID',

type: 'string',

},

{

name: 'user',

title: 'User',

type: 'reference',

to: [{ type: 'user' }],

},

{ name: 'products',

title: 'Products',

type: 'array',

of: [

{

type: 'object',

fields: [

{

name: 'product',

title: 'Product',

type: 'reference',

to: [{ type: 'product' }],

},

{

name: 'quantity',

title: 'Quantity',

type: 'number',

},

],

},

],

},

{

name: 'totalAmount',

title: 'Total Amount',

type: 'number',

},

{

name: 'status', title: 'Status', type: 'string', options: {

list: [ { title: 'Pending', value: 'pending'},

{ title: 'Shipped', value: 'shipped' },

{ title: 'Delivered', value: 'delivered'},

{ title: 'Cancelled', value: 'cancelled'},

],

},

},

{

name: 'createdAt',

title: 'Created At',

type: 'datetime',

},

],

};

# *Zone Schema :*

export default {

name: 'zone',

title: 'Zone',

type: 'document',

fields: [

{

name: 'name',

title: 'Zone Name',

type: 'string',

description: 'The name of the delivery zone (e.g., North America, Europe).',

},

{

name: 'regions',

title: 'Regions or Countries',

type: 'array',

of: [{ type: 'string' }],

description: 'List of regions or countries within this zone.',

},

{

name: 'shippingCost',

title: 'Shipping Cost',

type: 'number',

description: 'Base shipping cost for this zone.',

},

{

name: 'deliveryTime',

title: 'Estimated Delivery Time',

type: 'string',

description: 'Estimated delivery time for this zone (e.g., 3-7 business days).',

},

{

name: 'notes',

title: 'Additional Notes',

type: 'text',

description: 'Any specific restrictions or additional details about this zone.',

},

],

};

# *Shipment Schema :*

export default {

name: 'shipment',

title: 'Shipment',

type: 'document',

fields: [

{

name: 'shipmentID',

title: 'Shipment ID',

type: 'string',

description: 'Unique identifier for the shipment.',

},

{

name: 'order',

title: 'Order',

type: 'reference',

to: [{ type: 'order' }],

description: 'The order associated with this shipment.',

},

{

name: 'customerDetails',

title: 'Customer Details',

type: 'object',

fields: [

{ name: 'name', title: 'Customer Name', type: 'string' },

{ name: 'address', title: 'Shipping Address', type: 'text' },

{ name: 'contact', title: 'Contact Number', type: 'string' },

],

},

{

name: 'carrier',

title: 'Carrier',

type: 'string',

description: 'The shipping service provider (e.g., FedEx, DHL)’,

},

{

name: 'tracking',

title: 'Tracking Info',

type: 'object',

fields: [

{ name: 'trackingNumber', title: 'Tracking Number', type: 'string' },

{ name: 'trackingURL', title: 'Tracking URL', type: 'url' },

],

},

{

name: 'status',

title: 'Shipment Status',

type: 'string',

options: {

list: [

{ title: 'Pending', value: 'pending' },

{ title: 'Shipped', value: 'shipped' },

{ title: 'In Transit', value: 'inTransit' },

{ title: 'Delivered', value: 'delivered' },

{ title: 'Cancelled', value: 'cancelled' },

],

},

},

{

name: 'deliveryDate',

title: 'Delivery Dates',

type: 'object',

fields: [

{

name: 'estimated',

title: 'Estimated Delivery Date',

type: 'datetime'

},

{

name: 'actual',

title: 'Actual Delivery Date',

type: 'datetime'

},

],

},

{

name: 'zone',

title: 'Shipping Zone',

type: 'reference',

to: [{ type: 'zone' }],

description: 'The zone where the shipment is being delivered.',

},

{

name: 'notes',

title: 'Notes',

type: 'text',

description: 'Additional remarks or delivery instructions.',

},

],

};

**API Endpoints of the Avion E-Commerce furniture website**

Module : User Management

1. Endpoint

POST/api/users/register

Action

Register a new user

Flow

Frontend API SanityCMS Response to frontend

2. Endpoint

POST/api/users/login

Action

Authenticate user and return token

Flow

Frontend API Authentication service Token Response to frontend

Module : Product Management

1.Endpoint

GET/api/products

Action

Authenticate user and return token

Flow

Frontend API Authentication service Token Response to frontend

2.Endpoint

GET/api/products/ : id

Action

Fetch specific product details

Flow

Frontend API SanityCMS Response to frontend

Module : Cart Management

1.Endpoint

POST/api/cart

Action

Add a product to cart Flow

Flow

Frontend API Sanity CMS Response to frontend

2.Endpoint

GET/api/cart:userId

Action

Get cart details

Flow

Frontend API Sanity CMS Response

Module : Order Management

1.Endpoint

POST/api/orders

Action

Place an order

Flow

Frontend API SanityCMS Response to frontend

2.Endpoint

GET /api /orders/ : orderId

Action

Get order details

Flow Frontend API SanityCMS Response to frontend

Module : Payment Management

1.Endpoint

POST/api/payment

Action

Process payment

Flow

Frontend API SanityCMS Response to frontend

Module : Zone Management

1.Endpoint

GET /api /zones

Action

Get delivery zones and shipping info

Flow

Frontend API SanityCMS Response to frontend

Module : Shipment Management

1.Endpoint

GET /api/shipment/:shipmentId

Action

Track shipment status

*THE END*