Marketplace Technical Foundation - FURINO

Day 2 Activities: Transitioning to Technical Plannin

Technical Roadmap

This document outlines the system architecture for transitioning into technical planning. The architecture is structured to support a scalable, responsive, and user-friendly e-commerce platform. The system is divided into three key layers:

Front-end Architecture

Technology Stack:

The frontend leverages Tailwind CSS and Next.js to create a modern, responsive, and interactive user interface.

Core Features:

- Intuitive design for smooth product browsing.
- Optimized for both mobile and desktop devices.
- Includes key pages: Home, Product Listing, Product Details, Cart, Checkout, and Order Confirmation.

Responsibilities:

- Rendering dynamic UI components based on user actions and API responses.
- Maintaining high performance and accessibility standards.

Back-end Architecture

Technology Stack:

The backend is powered by Sanity CMS, serving as the primary data management system.

Core Features:

- Centralized handling of product information, customer details, and order data.
- Custom schema design aligned with the business requirements defined in the planning phase.

Responsibilities:

- Efficient and organization of all critical data.
- Providing APIs to enable seamless data exchange with the frontend.

Third-Party Integrations

APIs:

The system incorporates third-party services for enhanced functionality:

- Shipment tracking integration for real-time updates.
- Secure payment gateway integration for processing transactions.
- Additional backend services essential for a complete e-commerce experience.

Core Features:

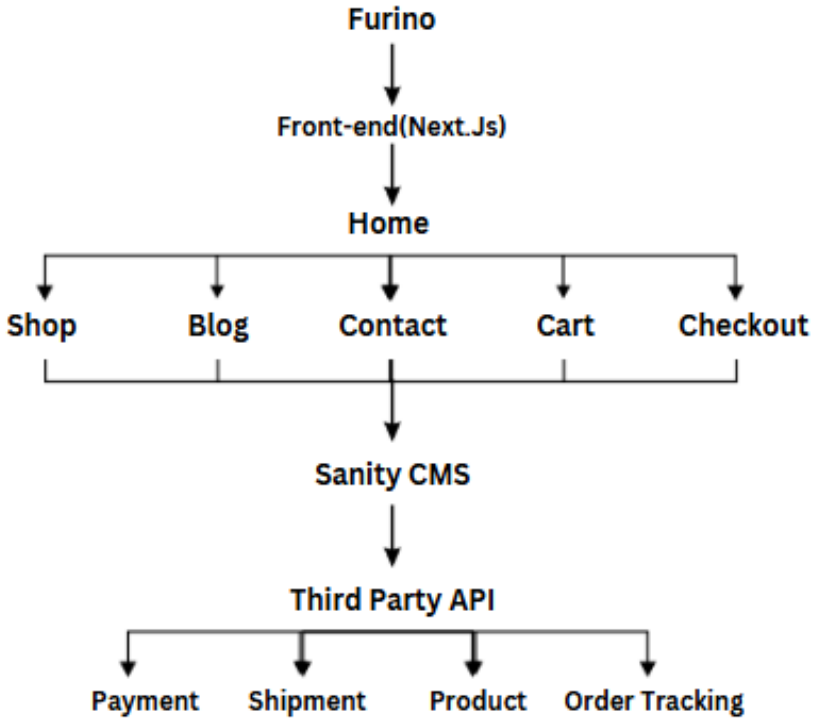
- Smooth data synchronization across the frontend, backend, and external services.
- Ensuring robust security and reliability in all operations.

Responsibilities:

- Enabling backend operations to run efficiently.
- Delivering the necessary data for the frontend to operate seamlessly.

Data Schema

System Architecture



Role of Components (Brief Overview)

Furino:

The platform's core brand and framework, creating the first impression and overall structure for the user experience.

Frontend (Next.js):

- Technology: Built with Next.js for a fast, responsive, and interactive user interface.
- Pages and Features:
  - *Home:* Main landing page guiding users to key sections.
  - *Shop:* Displays products for browsing, filtering, and selection.
  - *Blog:* Shares articles and updates to boost engagement.
  - *Contact:* Allows users to send inquiries or feedback.
  - *Cart:* Summarizes selected items for review.
  - *Checkout:* Handles payments, shipping, and order confirmation.

Sanity CMS:

- Manages backend data like product details, blog posts, user profiles, and orders.
- Acts as the central hub for organizing platform data.

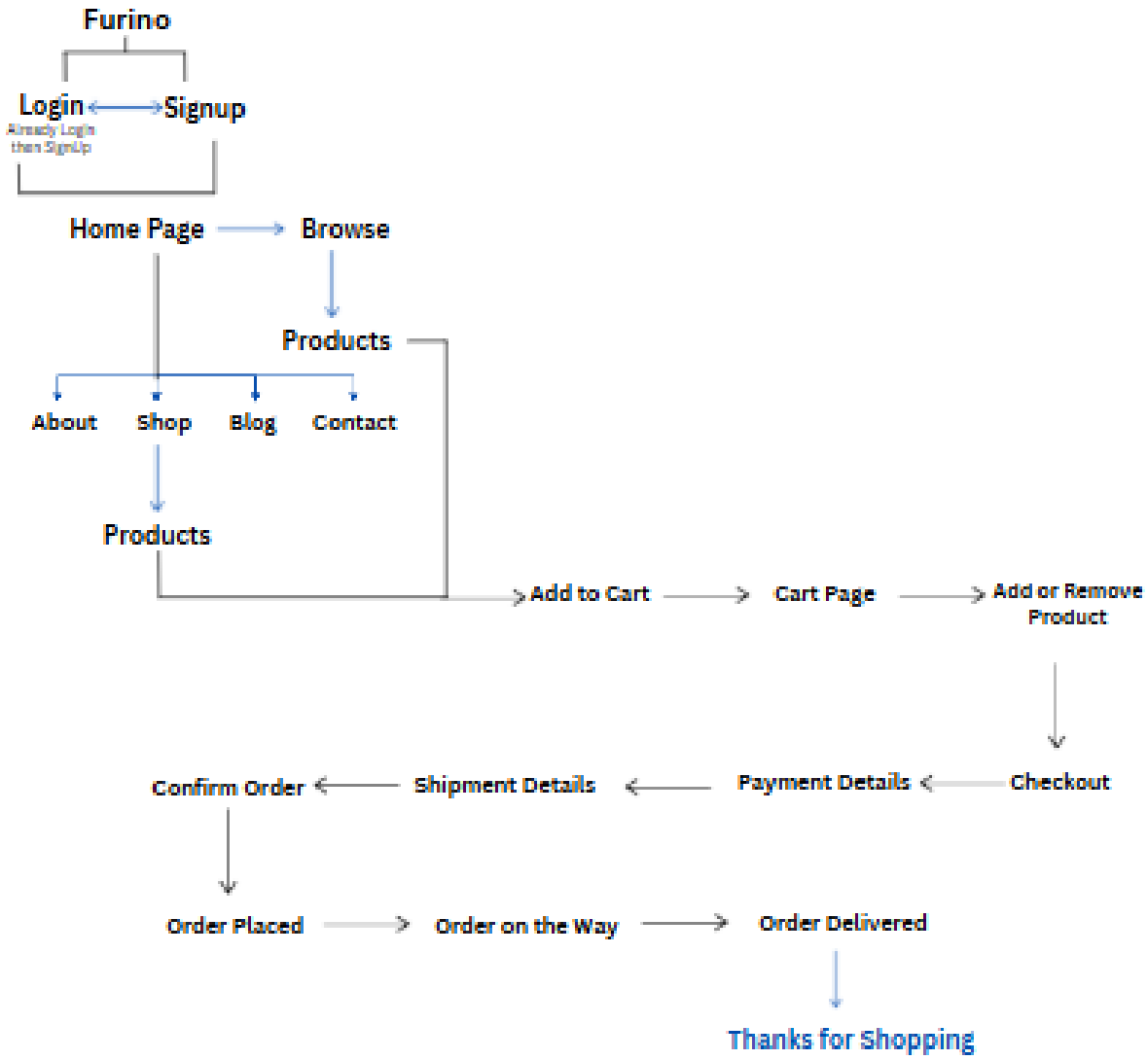
Third-Party API Integrations:

- Payment Gateways: Secure and smooth transaction processing.
- Shipping Management: Tracks and coordinates deliveries.
- Product Syncing: Ensures up-to-date product data across systems.
- Order Tracking: Provides real-time updates on order status.

API Specification

Endpoint	Method	Purpose	Response
/api/products	GET	Fetch all products	Product list
/api/products/{id}	GET	Fetch product details	ID, name, price, image
/api/cart	GET	Retrieve cart items	Products, Quantity, Total Bill
/api/shipment	POST	Add shipping details	Shipment ID, order ID, status, expected delivery date
/api/payment	POST	Process payment	Card details, Order ID, Price
/api/orders	GET	Get user order history	Customer info, product details, payment status

Workflow Diagram



By Maida Murtaza