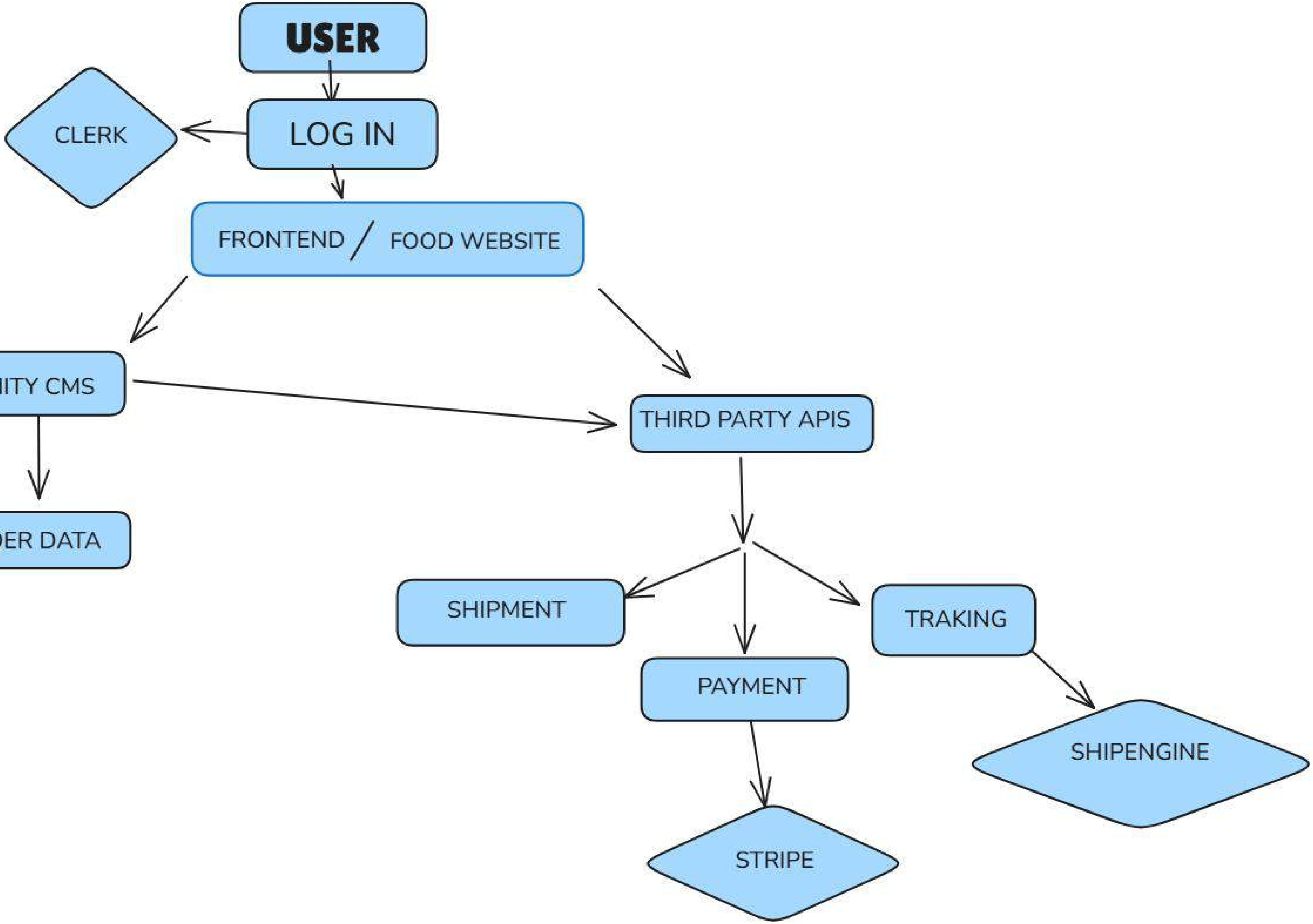

USER
CLERK
LOG IN
FRONTEND / FOOD WEBSITE
SANITY CMS
THIRD PARTY APIS
ORDER DATA
SHIPMENT
TRAKING
PAYMENT
SHIPENGINE
STRIPE

DAY 2

Date

MARKETPLACE TECHNICAL FOUNDATION PRESENTATION

* Introduction:

This document outlines the technical foundation for your food buisness marketplace. The focus is on defining the system architecture workflows, API requirements and Sanity CMS schema.

* Technical requirements:

* Responsive design:

Ensure the marketplace works seamlessly on mobile and desktop devices.

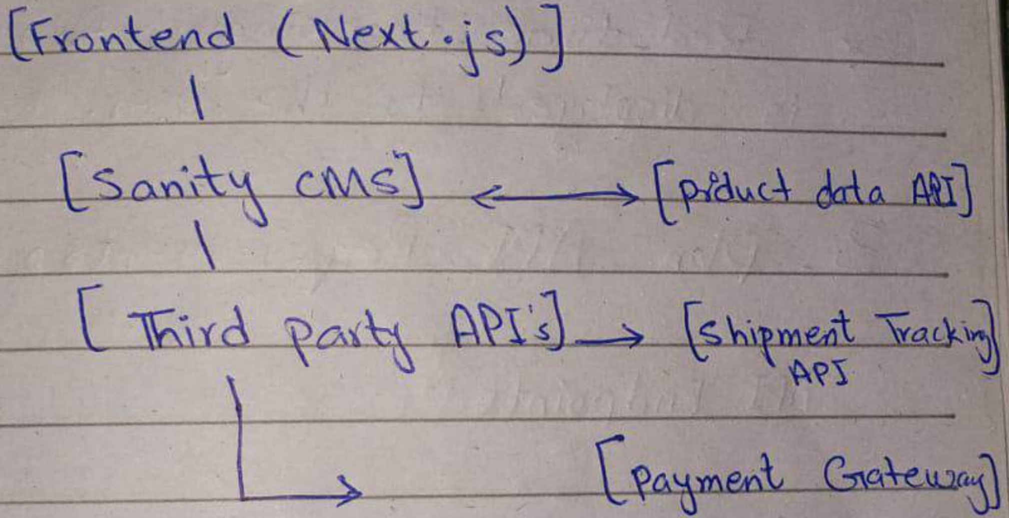
* Essential pages:

- Home
- Menu
- Blog
- About
- Shop
- products/id
- Sign in
- Sign up

- Contact
- Checkout
- Order Confirmation

* Backend reqluirements:
use Sanity CMS for managing product data, Customer details, order records.

2. Design System Architecture



* Workflow description:

*- user visits the frontend (Next.js) Frontend requests product data via the Sanity CMS API.

* Products are dynamically displayed to the user.

2. Order placement:

* user adds product to cart and

proceed to checkout.

* Order details are sent to Sanity CMS for record - keeping.
* payment is processed through the payment gateway.

3 - Shipment Tracking:

* order status is updated using the shipment Tracking API.
* Real_time tracking information is displayed to the user.

3- Plan API Requirements

API Endpoints

Endpoint	METHOD	Description	Responce
/shop	GET	Fetch all product details	{id, name, Price ,stock, image etc}
/orders	POST	Create a new order	{order Id, status: "success"}
/shipment	GET	Fetch order shipment status	{shipment Id, Status, delivery-date}
/ Payment	POST	Payment method	{ Payment Id , status : "Proffed" }
/ product/id	GET	Fetch single product details	{id ,name , price , stock image etc}

Sanity CMS Schemas

* <u>Product Schema:</u>

```
export default {
    name :'product',
    type : 'document',
    fields : [
      { name : 'name',
        type : 'string',
        title : 'product Name'
     },
       { name : 'price',
         type : 'number',
         title : 'price'
       },
       {
         name : 'Stock',
         type : 'number',
         title : 'Stock level'.
        },
        { name : 'image',
           type : 'image',
           title : 'product Image'
       }
         ]
          };
```

* Order Schema:

```
export default {
        name : 'order',
        type : 'document',
        fields : [
          { name : 'customer',
            type : 'reference',
              to : [
                { type : 'customer' }],
             title : 'Customer'
            },
         { name : 'products',
            type : 'array',
              of : [
                { type : 'reference',
              to : [
                   { type : 'product' }
         ]}],
         title : 'products' },
       { name : 'totalAmount',
         type : 'number',
         title : 'Total Amount' },
      { name : 'status',
        type : 'string',
        title : 'Order Status }
  },]
```