- **NAME : HAFSA SULTAN**
- **ROLL NUMBER : 453493**

PLANNING THE TECHNICAL FOUNDATION

OVERVIEW :

Build an e-commerce website where users can brouse products , add them to a cart and make purchases securely.

Buyers (customers) & Sellers (who manage and list products).

TECH STACKS :

- FRONTEND : Next.Js , React , CSS
- Backend: Sanity (CMS)
- Database : Mngo DB
- Payment : Paypal
- Shipment Trackings : shiptingine

SYSTEM COMPONENTS :

1. **User Interation :**
   - User login/ Sign up , credentials.
   - Database: For storing data securely with coded passwords.

2. **Content Management SYSTEM (CMS):**
   - The Frontend would fetch the data trom Sanity & this from APIs.
   - Results: The data will be fet ched the user can browses the product.

3. Order Managemnt :

- User places an order it will show them details of that and the total products added before.
- After its confirmation it will store securely on Sanity And shipment move order details to shipment API's for further porocess, tracking, delivery & status.

4. Cart :

- Users are allowed to add items, update quantities and remove.
- Frontend will create cart with options to view, edit or delete itens.

5. Payments :

- The option available like (PayPal, EasyPaisa) or can be selected for COD
- API will handle these transaction. Order will receive only after payment clearation.

6. Shipment Tracking :

- Ship Engine API used to for the tracking.
- User will notify with updates of order status

7. Delivery Zone :

- Forontend will show the status if delivered to the receiver.
- Assigned driver has to delivered with the correct ID's & info.

API ENDPOINTS :

1. Get Products :

Get / api / Products : Fetches list of available product.

GET / api / Products/ [id] : (Products/[id]: Fetcher detail of a single product byrd.

2. User Management :

Post / api / auth / register : register a new users.

Post /api / auth / login : authenticates existing user.

Posi / api / auth / profile : Fetch user profile.

3. Cart :

GET / api / card : Fetches the current user cart.

GET / api / cart / product : Add product to the cart.

4. Order :

POST / api / orders : Place order & process.

GET / api / orders / [ID] : Fetches an order by order ID.

5. Payment :

P051 / api / payment / create : creates payment.

6. Shipment :

POST / api / shipment : Create a new shipment.

Get / api / shipments / Tracks : Track shipment status

DATA SCHEME FOR SANITY :

FOR PRODUCTS :

```
export default {
  name: 'product',
  title: 'Product',
  type: 'document',
  fields: [
    {
      name: 'title',
      title: 'Product Title',
      type: 'string',
      description: 'The title of the product',
    },

    {
      name: 'description',
      title: 'Description',
      type: 'text',
      description: 'A detailed description of the product',
    },
    {
      name: 'price',
      title: 'Price',
      type: 'number',
      description: 'Price of the product in USD',
    },
    {
      name: 'image',
      title: 'Product Image',
      type: 'image',
      description: 'Image of the product',


    },

    {
      name: 'stock',
      title: 'Stock Quantity',
      type: 'number',
      description: 'The quantity of the product available in stock',
    },
  ],
};
```

FOR CUSTOMERS :

```
export default {
  name: 'customer',
  title: 'Customer',
  type: 'document',
  fields: [
    {
      name: 'firstName',
      title: 'First Name',
      type: 'string',

    },
    {
      name: 'lastName',
      title: 'Last Name',
      type: 'string',

    },
    {
      name: 'email',
      title: 'Email',
   type: 'string',


  },
  {
   name: 'phone',
   title: 'Phone',
   type: 'string',


  },
  {
   name: 'address',
   title: 'Address',
   type: 'text',
   description: 'The shipping address of the customer',
  },
  {
   name: 'orders',
   title: 'Orders',
   type: 'array',


   description: 'A list of orders placed by the customer',
  },
 ],
};
```

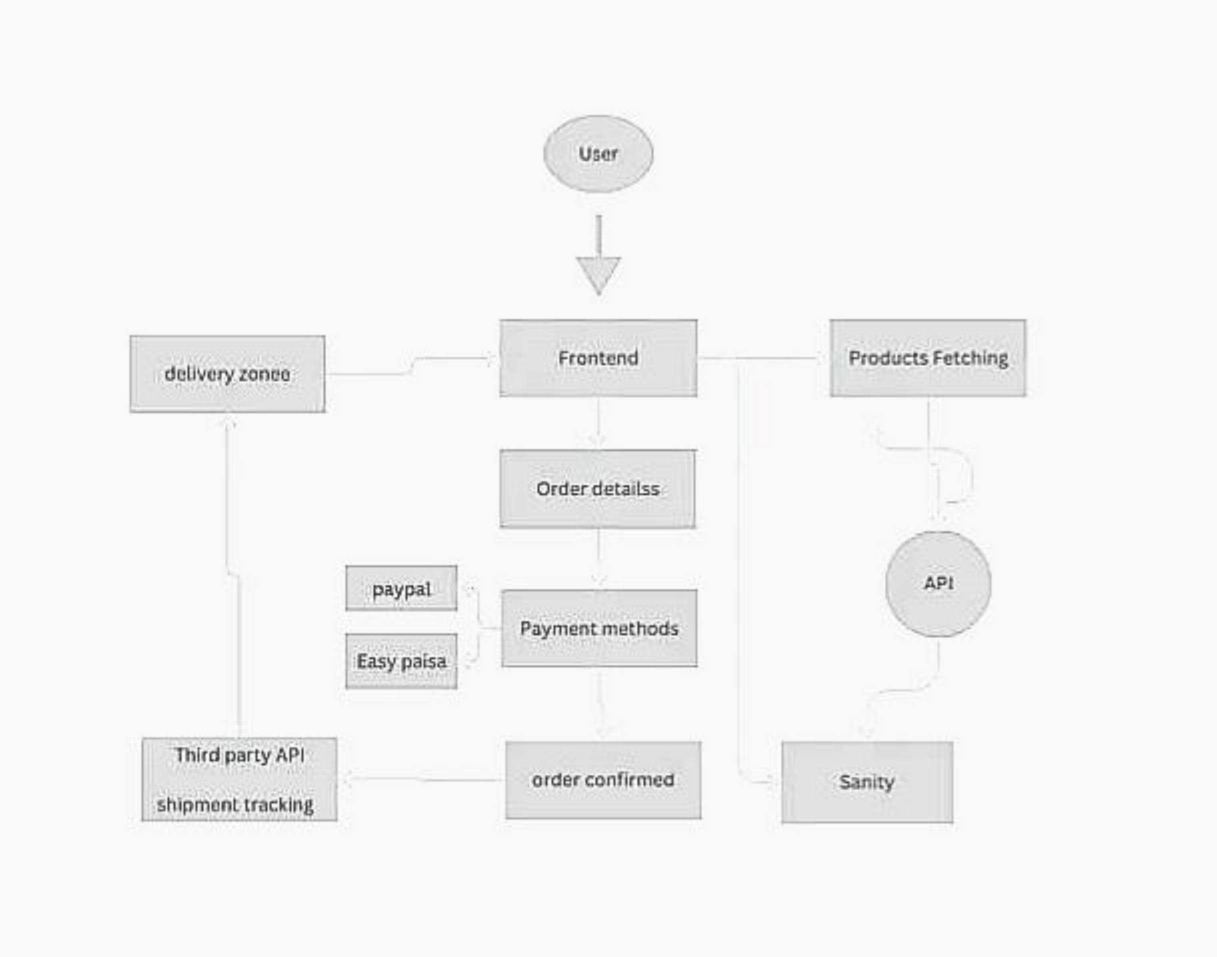
FOR ORDERS :

```
export default {
  name: 'order',
  title: 'Order',
  type: 'document',
  fields: [
    {
      name: 'orderNumber',
      title: 'Order Number',
      type: 'string',
      description: 'Unique identifier for the order',
    },
    {
      name: 'customer',
      title: 'Customer',
      type: 'reference',

    },
    {
      name: 'products',
      title: 'Products',
      type: 'array',

      description: 'List of products in the order',
    },
    {
      name: 'totalPrice',
      title: 'Total Price',
      type: 'number',
      description: 'The total price for the order',
    },
    {
      name: 'status',
      title: 'Order Status',
      type: 'string',
      options: {
        list: ['Pending', 'Shipped', 'Delivered', 'Cancelled'],
      },
      description: 'Current status of the order',
    },
    {
      name: 'shippingAddress',
      title: 'Shipping Address',
      type: 'text',
      description: 'The address to which the order should be shipped',
    },
    {
      name: 'orderDate',
      title: 'Order Date',
      type: 'datetime',
      description: 'The date the order was placed',
    },
  ],
};
```

DIAGRAM REPRESNTATION :


User
Frontend
delivery zonee
Products Fetching
Order detailss
API
paypal
Payment methods
Easy paisa
Third party API
shipment tracking
order confirmed
Sanity