MARKET PLACE TECHNICAL FOUNDATION

Day2

GOVERNOR SINDH INITIATIVE

Governor Sindh Initiative for Artificial Intelligence, Web 3.0 & Metaverse

Designed BY ENGR SIR ABDUL REHMAN ANSARI

FRONTEND WEB DEVELOPER/HTML/CSS/JAVA/NEXT.JS/REACT

Step:1

Technology Stack

- **Framework:** Next.js
  A React-based framework that enables server-side rendering (SSR) and static site generation (SSG), perfect for building high-performance and SEO-friendly applications.
- **UI Library:** shadcn/UI
  Provides pre-built, customizable UI components that work seamlessly with Tailwind CSS.
- **Styling Framework:** Tailwind CSS
  A utility-first CSS framework for fast and responsive design.

Step:2

Pages to Create

Home Page

- Features: Display promotional banners, featured products, and categories.
- Libraries/Features to Use:
- shadcn/ui components like Card, Button, and Carousel.
- Tailwind CSS for responsive design.

Step:2

Steps to create page

Product Listing Page

- shadcn/ui components like Select, Input, and Grid.
- Use dynamic routes (pages/products/[category].tsx) for category-based listing.

Product Details Page

- shadcn/ui components like Modal, Carousel, and Tabs.
- Image optimization using Next.js <Image> component.

Cart Page

- shadcn/ui components like Table and Badge.
- Persist cart data in localStorage or context.

Checkout Page

- shadcn/ui components like Form and Input.
- Integrate a payment gateway like **Stripe**.

Rental Listing Page

- Dynamic routes (pages/rentals/[id].tsx) for individual helicopter details.
- shadcn/ui components like Card and Filter.

Rental Details Page

- shadcn/ui components like Modal and Accordion.

Rental Cart Page

- shadcn/ui components for form handling and summary display.

Sanity CMS

Step:3

Schema Design for Data Management

Product Schema

example

```
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: 'description', type: 'text', title: 'Description' },
    { name: 'image', type: 'image', title: 'Product Image' }
  ]
```

Step:3

Schema Design for Data Management

Rental Schema

example

```
export default {
  name: 'rental',
  type: 'document',
  fields: [
    { name: 'name', type: 'string', title: ‘Gift Name' },
    { name: 'rentalPrice', type: 'number', title: 'Rental Price per Hour' },
    { name: 'availability', type: 'boolean', title: 'Availability' },
    { name: 'description', type: 'text', title: 'Description' },
    { name: 'image', type: 'image', title: ' Image' },
    { name: 'rentalTerms', type: 'text', title: 'Rental Terms' }
  ]
};
```

Step:3

Schema Design for Data Management

Customer Schema

example

```
export default {
  name: 'customer',
  type: 'document',
  fields: [
    { name: 'name', type: 'string', title: 'Customer Name' },
    { name: 'email', type: 'string', title: 'Email Address' },
    { name: 'address', type: 'string', title: 'Shipping Address' }
  ]
};
```

Step:3

Schema Design for Data Management

Order Schema

example

```
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: 'total', type: 'number', title: 'Total Amount' },
    { name: 'status', type: 'string', title: 'Order Status' }
  ]
};
```

Step:3

Schema Design for Data Management

Rental Order Schema

example

```
export default {
  name: 'rentalOrder',
  type: 'document',
  fields: [
    { name: 'customer', type: 'reference', to: [{ type: 'customer' }], title: 'Customer' },
    { name: ‘gift', type: 'reference', to: [{ type: 'rental' }], title: ‘Gift' },
    { name: 'rentalDuration', type: 'number', title: 'Rental Duration (Hours)' },
    { name: 'total', type: 'number', title: 'Total Amount' },
    { name: 'status', type: 'string', title: 'Rental Order Status' }
  ]
};
```

Step:4

Third-Party APIs

Requirements

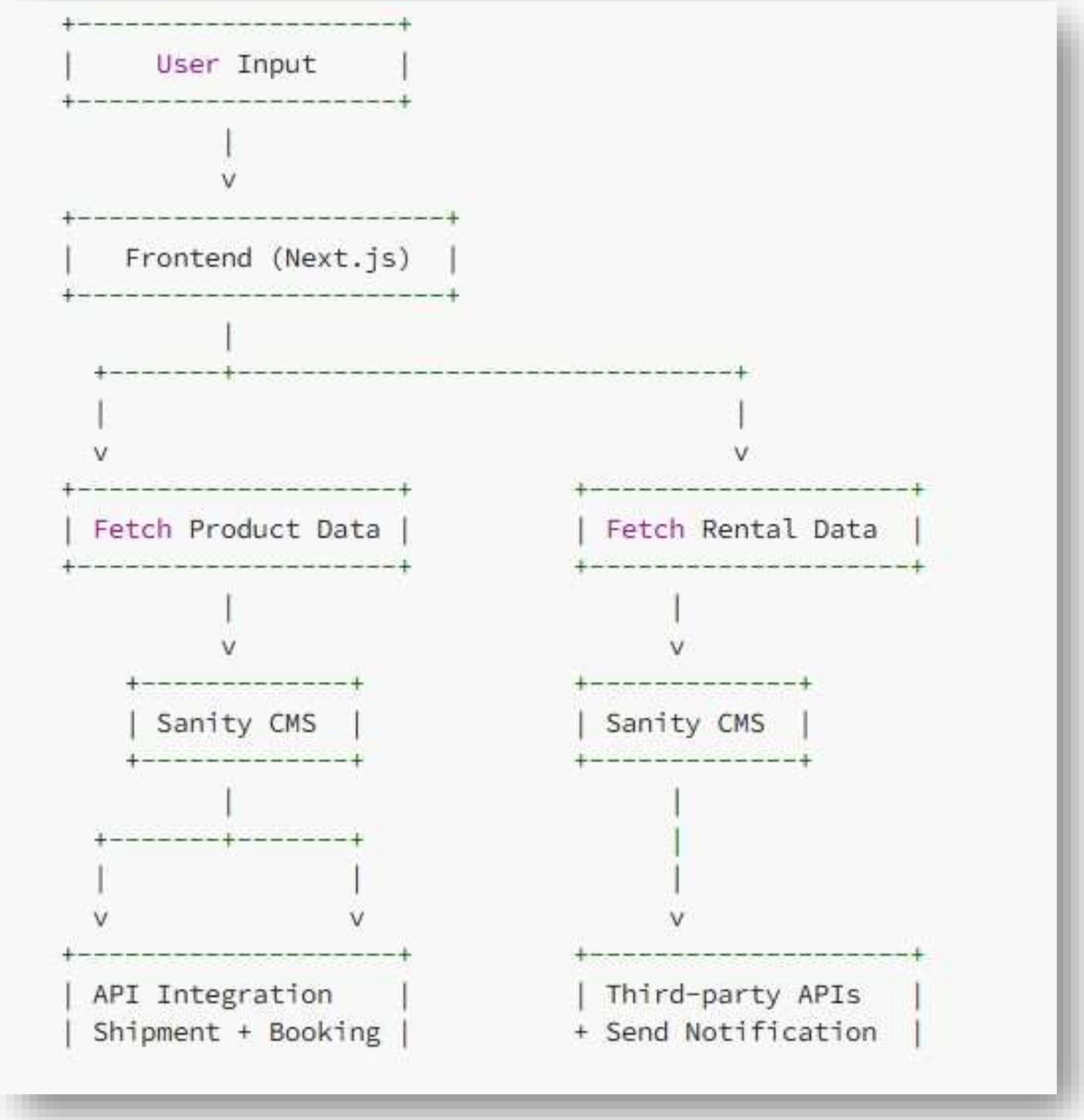
example

- **Payment Gateway API**: For secure transaction processing.
- **Shipment Tracking API**: To provide real-time shipment updates.
- **Email Notification API**: For sending order confirmations.
- **Booking Availability API**: To check gift rental availability.

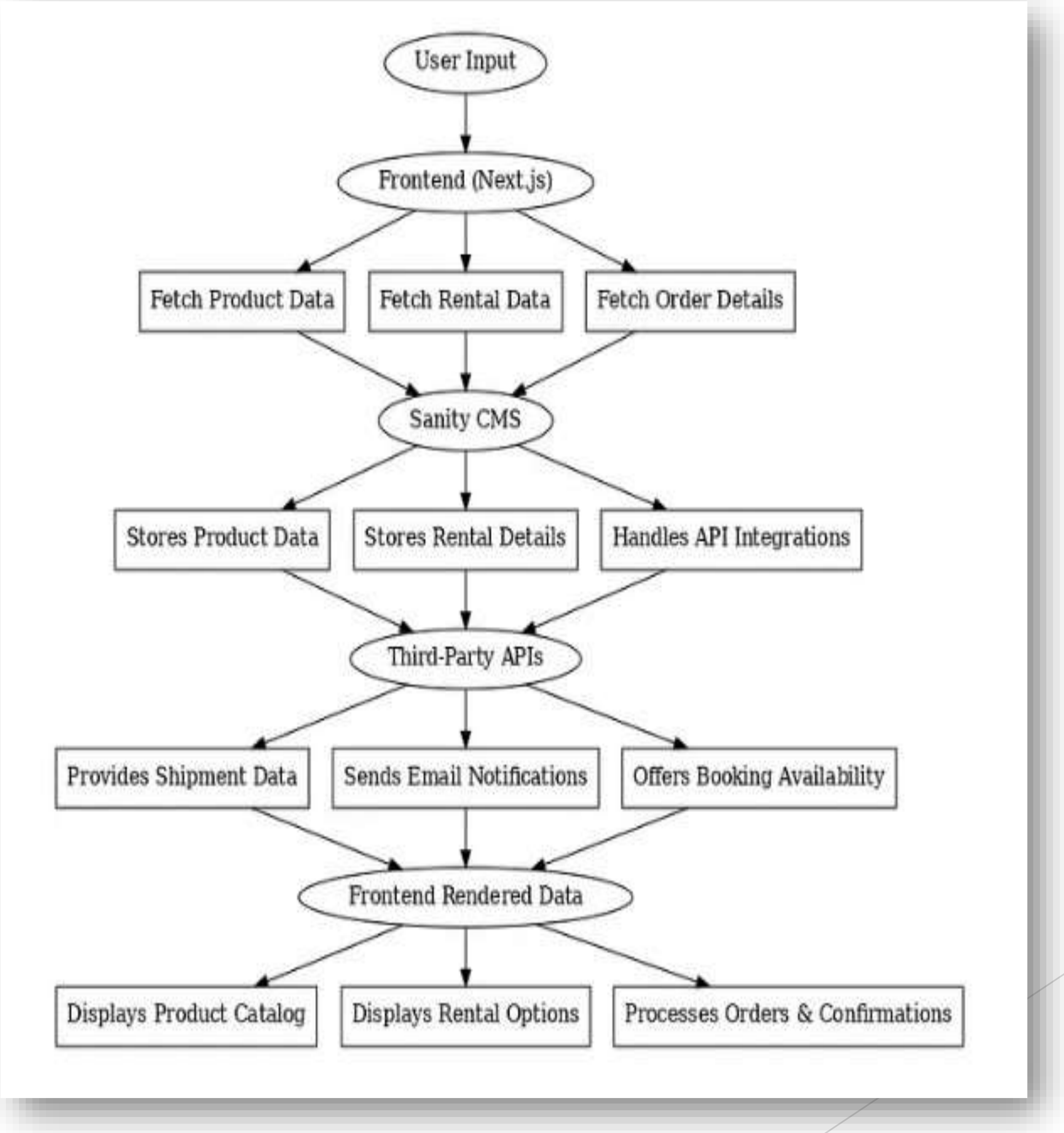
Step:5

System Architecture Design

Diagram



```
+--------------------+
|     User Input     |
+--------------------+
          |
          v
+-----------------------+
|   Frontend (Next.js)  |
+-----------------------+
          |
  +-------+-------------------------------+
  |                                       |
  v                                       v
+--------------------+          +--------------------+
| Fetch Product Data |          | Fetch Rental Data  |
+--------------------+          +--------------------+
          |                           |
          v                           v
    +-------------+             +-------------+
    | Sanity CMS  |             | Sanity CMS  |
    +-------------+             +-------------+
          |                           |
  +-------+-------+                   |
  |               |                   |
  v               v                   v
+--------------------+          +--------------------+
| API Integration    |          | Third-party APIs   |
| Shipment + Booking |          + Send Notification  |
```


User Input
Frontend (Next.js)
Fetch Product Data
Fetch Rental Data
Fetch Order Details
Sanity CMS
Stores Product Data
Stores Rental Details
Handles API Integrations
Third-Party APIs
Provides Shipment Data
Sends Email Notifications
Offers Booking Availability
Frontend Rendered Data
Displays Product Catalog
Displays Rental Options
Processes Orders & Confirmations

Step:6

API Endpoints Plan

Planning

Route

/products **(GET)**

- **Description**: Fetch all product details.
- **Response Example:**

GO

Examples

```
 [
 {
   "id": "1",
   "name": “gift item",
   "price": 20.0,
   "stock": 50
 }
]
```

To be continue

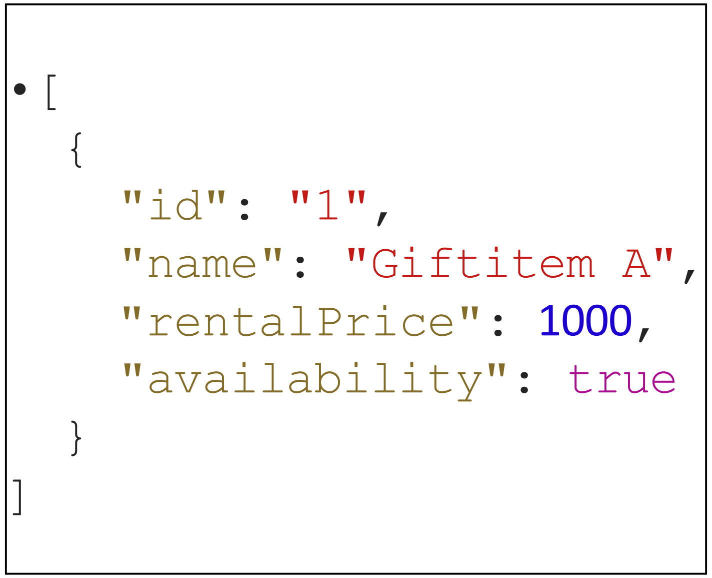
Route

/rentals **(GET)**

- **Description**: Fetch all available rental gift items.
- **Response Example:**

GO

Examples



```
• [
  {
    "id": "1",
    "name": "Giftitem A",
    "rentalPrice": 1000,
    "availability": true
  }
]
```

To be continue

Route

/orders **(POST)**

- **Description**: Save new order details.
- **Request Example:**

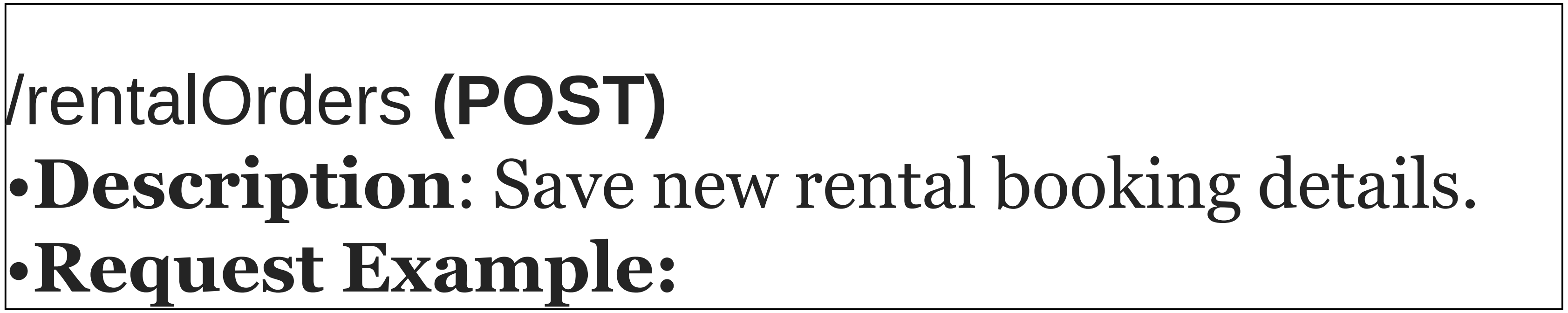
GO

Examples

```
{
  "customerId": “786",
  "products": ["1", "2"],
  "total":90.0,
}
```

To be continue

Route



/rentalOrders **(POST)**

- **Description**: Save new rental booking details.
- **Request Example:**

GO

Examples

```
{
  "orderId": "123",
  "status": "In Transit"
}
{
  "customerId": “786",
  “giftitemId": "1",
  "rentalDuration": 5,
  "total": 2500
}
```

To be continue

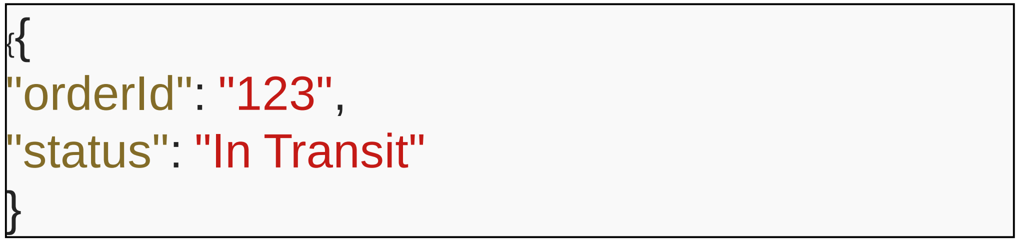
Route

/shipment **(GET)**

- **Description**: Fetch the shipment status for an order.
- **Response Example:**

GO

Examples



```
{
"orderId": "123",
"status": "In Transit"
}
```

To be continue

Thank you so much For watching my repo

Contact:+923100273708(official)

https://github.com/rehman2017

abdulrehman201717@gmail.com
abdulrehman20172024@gmail.com
abdulrehman2024@outlook.com

Connected with

Engr Sir Abdul Rehman Ansari(official web developer working at the Academy School near Zia Uddin hospital main campus)

Special Thanx To

Allah Hafiz

Take care of yourselves