

Thursday
Day: 16-Jan-2025

Date: _____

Hackathon Marketplace Builder

Day 2

How does my marketplace functions?

My marketplace's functionality is divided into 3 parts

1. The frontend where the user visits and views the product and place orders and interacts with the website. The frontend is Next.js styled with Tailwind.

2. The backend where the orders and product data is stored from where the data is given to the frontend for where the user sees the data.

3. The APIs who deliver the data to backend from frontend and also provide data from backend to frontend.

1. The frontend:-

This is where the user visits the website the first page is the homepage which is the starting point from where they will navigate to their desired page through links which will be displayed on the navbar or the read more. The homepage has a section which has promotions of the product.

There is a link to a page named shop which acts as a product listing page which shows product image name and price from where user can click any product they want and that product's data will be fetched dynamically. There is an add to cart button which adds the product.

Day: _____

Date: _____

to the cart. Then the user visit the cart page from where they can checkout. For checkout they will visit the checkout page when they click checkout on api step's in to save and create their orders on the backend. For all this to work user must sign up to the website. When the user sign's up through email they will get an confirmation which is an authentication for the user once they sign up their data will be stored in sanity

2. The backend.

Now the backend works. The product and order and customers all have an dedicated schema with all the necessary details required for the user to place the order. The backend of my market place is on sanity an headless cms.

How the backend work when the user clicks on shop the backend receives an request to display all the product so it complies. When the user click's a product the backend again receives an request for an individual. now the backend check if it have the product if yes then it send's the data if not then it throw's an error e.g "product not found".

Now the last part the creation of a new user and it's order's. When the user give's it detail the backend check's if all the information needed is given then it creates an user and then it creates the order.

Day: _____

Date: _____

all the process is done through API
endpoint Get and Post

Get endpoint work's and fetch's and display
the requested product. I will explain this in the
json file. So see that for explanation.

Day: _____

Date: _____

Frontend Workplace

Homepage:-

User opens the website and lands on the homepage.



Go To Shop:-

User navigates to the shop section of the website



Choose The Desired Product:-

User select a product they wish to buy.



Product Details:-

Product details are retrieved from the backend and displayed to the user



Select Product:-

User selects the product and adds it to their cart



Place Order :-

User places their order and provide payment and shipping information.



Order Created:-

Order is created and a confirmation is sent to the user.



Track Order:-

User can track their order using a tracking ID

Day: _____

Date: _____

Close Website:-

User closes the website.

Backend Workflow:-

Receive Product Selection Request:-

Backend receives a request from the frontend to retrieve product detail



Retrieve Product Details from Database:-

Backend retrieves the product details from database if available, if not it shows error.



Validate Product Availability & Pricing:-

Backend validates the products availability and pricing.



Update Shopping Cart & Session:-

Backend updates the shopping cart session data.



Receive Order Placement Request:-

Backend receives a request from the frontend to place an order.



Validate Order Information & Payment Details:-

Backend validates the order information and payment details.

Day: _____

Date: _____



Process Payment + Update Order Status:-

Backend processes the payment
+ update the order status.



Generate Order Confirmation + Tracking ID:-

Backend generates an order
confirmation and tracking ID.



Send Order Confirmation Email:-

Backend sends an order confirmation
email to the customer



Update Inventory + Order History:-

Backend updates the inventory
+ order history.