

Day 3 - API Integration Report - [Your Marketplace Name]

Objective

On Day 3, the primary goal is to integrate APIs and migrate data into Sanity CMS to build a functional marketplace backend. Students will learn how to utilize APIs for referencing data, migrate existing data into Sanity CMS, and ensure compatibility with marketplace templates. This practical approach simulates real-world scenarios, preparing students to handle diverse client requirements, such as integrating headless APIs or migrating data from popular eCommerce platforms.

Submission Requirements

Submit a detailed report documenting the following:

- **API Integration Process:** Steps, methods, and tools used to integrate APIs into the marketplace backend.
- **Schema Modifications:** Adjustments made to Sanity schemas to support API data or ensure compatibility with templates.
- **Data Migration Steps and Tools Used:** Describe the migration process, the tools utilized, and how the data was transferred into Sanity CMS while ensuring accuracy and consistency

```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. All rights reserved.

D:\day3\day03>npx create sanity@latest
  ■ create ■

Welcome to create: a delightful repository templating engine.

Learn more about create on:
  https://create.bingo

Running with mode --migrate using the template:
  create-sanity@latest

x Could not load create-sanity@latest: The default export of create-sanity@latest should be a Template.
- The default export of create-sanity@latest should be a Template.

■ You are logged in as hassanjaffer043@gmail.com using Google
■ Fetching existing projects

? Create a new project or select an existing one Create new project
? Your project name: dayap1
Your content will be stored in a dataset that can be public or private, depending on
whether you want to query your content with or without authentication.
The default dataset configuration has a public dataset named "production".
? Use the default dataset configuration? Yes
■ Creating dataset
? Would you like to add configuration files for a Sanity project in this Next.js folder? Yes
? Do you want to use TypeScript? Yes
? Would you like an embedded Sanity Studio? Yes
? What route do you want to use for the Studio? /studio
? Select project template to use Clean project with no predefined schema types
? Would you like to add the project ID and dataset to your .env.local file? Yes
Added http://localhost:3000 to CORS origins
Running 'npm install --legacy-peer-deps --save @sanity/vision@3 sanity@3 @sanity/image-url@1 styled-components@6'
npm warn deprecated @sanity/block-tools@3.70.0: Renamed - use '@portabletext/block-tools' instead. '@sanity/block-tools' will no longer receive updates.

added 906 packages, and audited 1274 packages in 5m

245 packages are looking for funding
  run 'npm fund' for details

found 0 vulnerabilities

added 16 packages, and audited 1290 packages in 19s

245 packages are looking for funding
  run 'npm fund' for details

found 0 vulnerabilities

Success! Your Sanity configuration files has been added to this project

D:\day3\day03>_

```

```

run 'npm fund' for details

found 0 vulnerabilities

Success! Your Sanity configuration files has been added to this project

D:\day3\day03>

```




gadgets

PLAN
Growth Trial

STATUS
Active

PROJECT ID
sl21hvcz

- Getting started
- Overview
- Members
- Studios
- Datasets
- Access
- Activity
- Usage
- Plan
- API
- Settings

Usage

View more →

407 / 1m API CDN Requests	35 / 250k API Requests
1.9 MB / 100 GB Assets	7.8 MB / 100 GB Bandwidth
1 / 2 Datasets	38 / 10k Documents
1 / 20 User seats	

Project members

View all →

Invite your first team member

+ Invite project members

Activity

View more →

Feb 8 at 1:26 PM
CORS origin <https://gadgets-hackathon.vercel.app> was added to the project

Feb 7 at 8:59 PM
CORS origin <https://gadgets-phi.vercel.app> was added to the project

Feb 7 at 4:50 PM



gadgets

PLAN
Growth Trial

STATUS
Active

PROJECT ID
sl21hvcz

- Getting started
- Overview
- Members
- Studios
- Datasets
- Access
- Activity
- Usage
- Plan
- API
- Settings

- Webhooks
- CORS origins
- Tokens

GROQ-powered webhooks

HTTP callbacks to a given URL triggered by changes in your content lake

Learn more about webhooks

+ Create webhook

0 of 2 webhooks
(2 included in plan)

Get more webhooks

There are no GROQ-powered webhooks in this project
Maybe try creating a new webhook?

CORS origins

Hosts that can connect to the project API.

+ Add CORS origin

ORIGIN	CREDENTIALS	CREATED
https://www.sanity.io/manage/personal/project/sl21hvcz/api#tokens		

Getting startedOverviewMembersStudiosDatasetsAccessActivityUsagePlanAPISettings

Webhooks

CORS origins

Tokens

CORS origins

Hosts that can connect to the project API.

ORIGIN	CREDENTIALS	CREATED	
https://gadgets-hackathon.vercel.app	Allowed	43 minutes	
https://gadgets-phi.vercel.app	Allowed	17 hours	
http://localhost:3000	Allowed	21 hours	
...			

+ Add CORS origin

Tokens

Tokens are used to authenticate apps and scripts to access project data.

+ Add API token

src

pages

product

_app.tsx

_document.tsx

cables.tsx

cars.tsx

charging.tsx

checkout.tsx

dropdown.tsx

earbuds.tsx

gaming.tsx

hero.tsx

index.tsx

laptop.tsx

login.tsx

phonecases.tsx

saletx

signup.tsx

smartwatches.tsx

sanity

lib

schemaTypes

attentionBarts

cables.ts

charging.ts

ear-buds.ts

gaming-home.ts

gaming.ts

hero.ts

index.ts

kitchen.ts

laptops.ts

phone-cases.ts

products.ts

salets

smart-watches.ts

trending.ts

envts

structure.ts

styles

emulocal

eslintconfig

gitignore

next-env.d.ts

next.config.mjs

package-lock.json

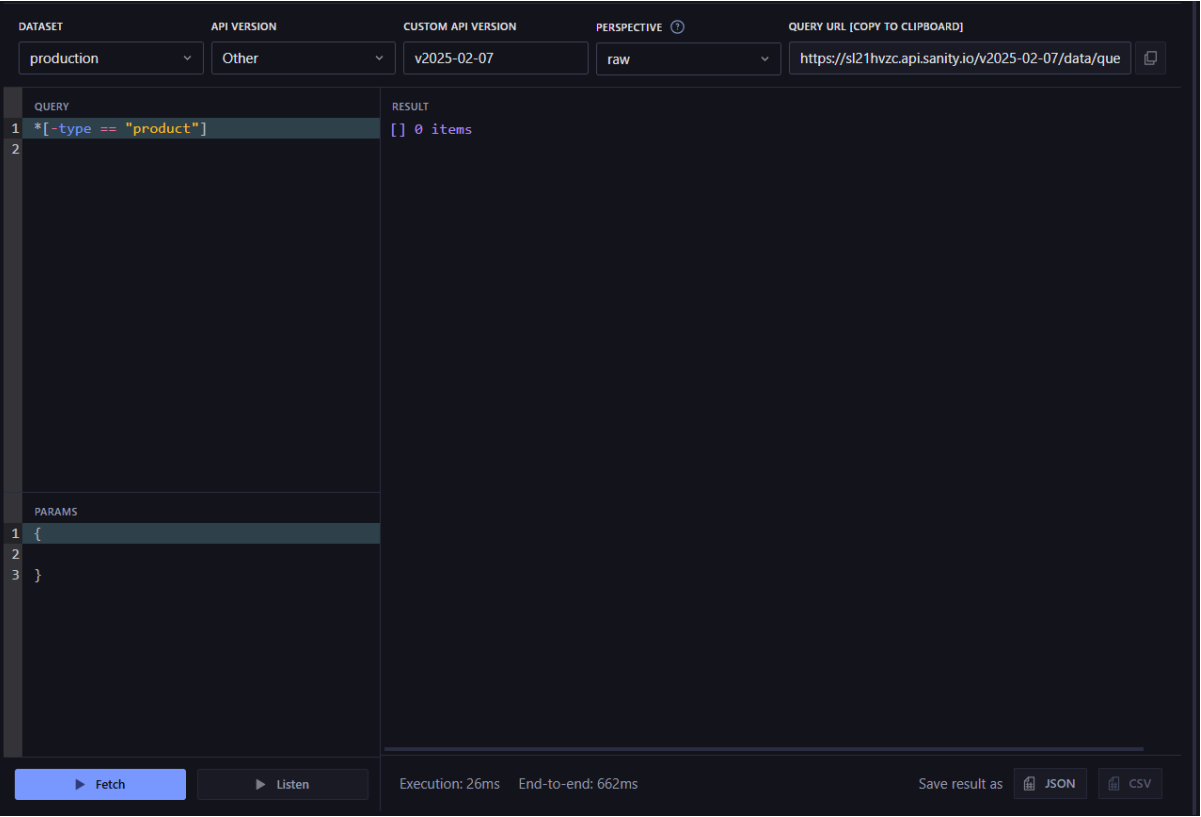
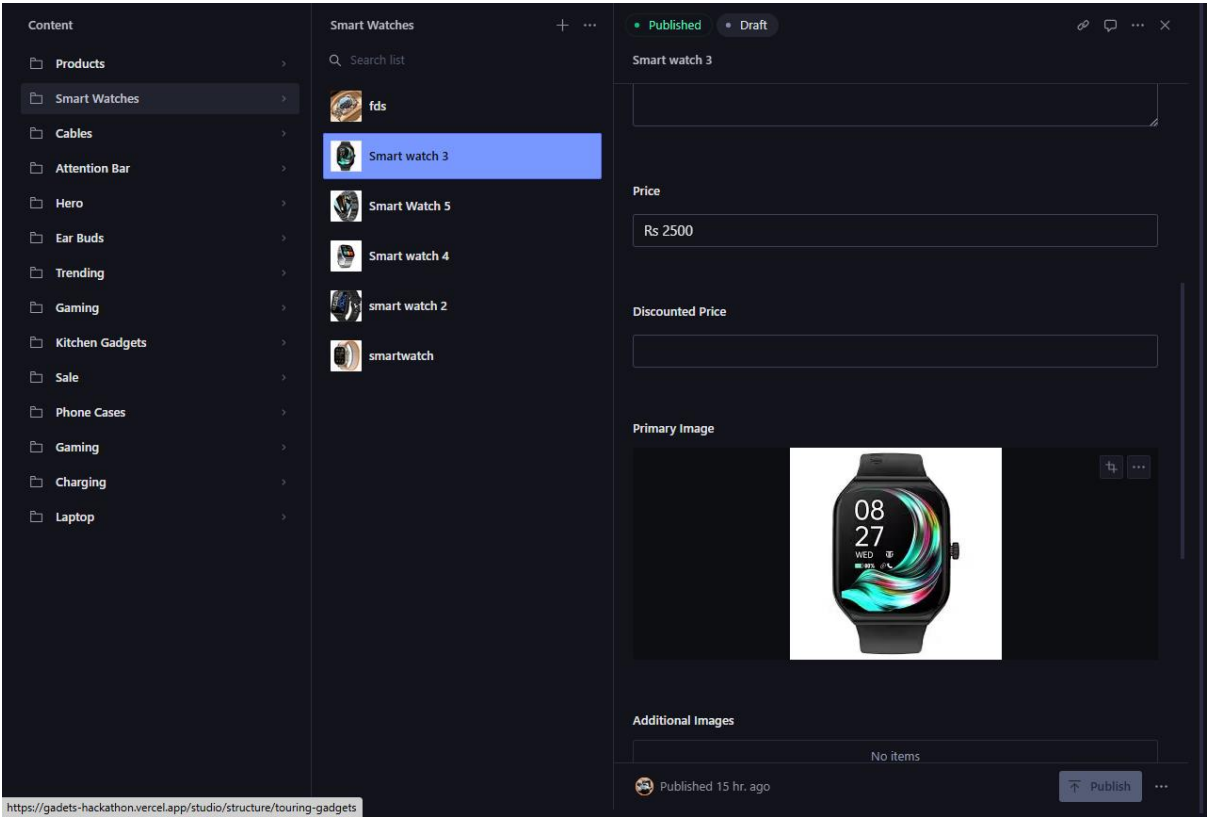
OUTLINE

TIMELINE

```
import { type SchemaTypeDefinition } from 'sanity';

export const product: SchemaTypeDefinition = {
  name: 'product',
  title: 'Products',
  type: 'document',
  fields: [
    {
      name: 'name',
      title: 'Name',
      type: 'string',
    },
    {
      name: 'description',
      title: 'Description',
      type: 'text',
    },
    {
      name: 'price',
      title: 'Price',
      type: 'string',
    },
    {
      name: 'discountedPrice',
      title: 'Discounted Price',
      type: 'number',
    },
    {
      name: 'image',
      title: 'Primary Image',
      type: 'image',
      options: { hotspot: true },
    },
    {
      name: 'images',
      title: 'Additional Images',
      type: 'array',
      of: [{ type: 'image' }],
      options: { hotspot: true },
    },
    {
      name: 'slug',
      title: 'Slug',
      type: 'slug',
      options: { source: 'name', maxLength: 96 },
    },
    {
      name: 'inStock',
      title: 'In Stock',
      type: 'boolean',
      description: 'Check this box if the product is available in stock.',
    },
  ],
};
```


Manual Data Entry:



Product List

Smart Gadget



Xomi Ball
Rs 6999

Add to Cart



3 in 1 (Charger)
Rs 2500

Add to Cart



Out Door Camera
Rs 16000

Add to Cart



Mobile control RGB Bulb
Rs1500

Add to Cart



Electric scooty
RS 11000

Add to Cart

Bluetooth Gadgets



Bluetooth Speaker 5
Rs 6500

Add to Cart



Bluetooth Airbuds
Rs 2500

Add to Cart



Bluetooth Speaker 2
Rs 2500

Add to Cart



Air buds 2
Rs 3500

Add to Cart



Bluetooth Speaker
Rs 4500

Add to Cart

Document Title: "Day 4 - Dynamic Frontend Components – E-Corner"

E-Corner – Your Ultimate Destination for Bike Accessories

E-Corner is a premier online store offering a wide range of high-quality Electronic Gadget and accessories at competitive prices. Whether you're upgrading your home with premium components or looking for essential gear, we provide everything from performance electric shaving kit, Sound systems, and Laptop and RGB lighting solutions. At E-Corner, we are committed to delivering superior quality, affordability, and a seamless shopping experience for all Electronic enthusiasts.

Steps for Building Key Features of an E-commerce Platform

1. Product Listing
 - Design a database schema for products (name, price, description, category, images).
 - Develop an API to fetch products from the database.
 - Create a frontend UI to display products in a grid or list format.
 - Implement pagination, sorting, and filtering to improve product navigation.
2. Product Detail Page
 - Set up a dynamic product detail route (e.g., /product/:id).
 - Fetch and display product-specific details such as name, price, description, images, and customer reviews.
 - Add an "Add to Cart" button and display stock availability.
3. Search Bar
 - Implement a search input field on the homepage or header.
 - Provide real-time search suggestions with debounce for a smoother user experience.
 - Fetch matching products using a search API with query-based filtering.
 - Display search results dynamically as the user types.
4. Admin Panel

- Build an admin authentication system for secure access.
- Develop CRUD functionalities for managing products, categories, and users.
- Implement an order management system to track purchases.
- Add dashboard analytics for sales performance and insights.

5. Cart System

- Create a cart model in the database or use local storage for guest users.
- Allow users to add, remove, and update the quantity of items in their cart.
- Display the cart total, applicable taxes, and checkout options.
- Develop a checkout page with integration for payment gateways

6. User Profile


- Implement user authentication with features such as login, registration, and token-based authentication.
- Develop a profile page where users can view their order history, saved addresses, and account details.
- Provide functionality for users to update profile information and manage their orders.

7. Category Management System

- Create category models in the database (e.g., Bike Parts, Riding Gear, Accessories).
- Develop an API to retrieve categories dynamically from the database.
- Design a category-based filtering system to enhance product browsing and organization.
- Integrate category-based navigation into the UI for a more intuitive shopping experience


Product Listing

Smart Gadget




Xomi Ball
Rs 6999

Add to Cart




3 in 1 (Charger)
Rs 2500

Add to Cart




Out Door Camera
Rs 16000

Add to Cart



Mobile control RGB Bulb
Rs1500


Add to Cart



Electric scooty
RS 11000


Add to Cart

Bluetooth Gadgets




Bluetooth Speaker 5
Rs 6500

Add to Cart




Bluetooth Airbuds
Rs 2500

Add to Cart




Bluetooth Speaker 2
Rs 2500

Add to Cart



Air buds 2
Rs 3500

Add to Cart




Bluetooth Speaker
Rs 4500

Add to Cart

<https://gadgets-hackathon.vercel.app/product/null>

Product Details:


0123-45678
customer.ecorner@gmail.com
Welcome To E-Corner



Shopping Bag 0

Home
Sale
Charging
Laptop
Cables
Smart Watches
Earbuds
Gaming
Phone Cases

Xomi Ball







Dual EQ mode: Designed with two equalizer modes normal & deep bass mode, so that you can choose your music/ video listening experience the way you want it to be. Signal-to-noise ratio : 70dB Delightful sound experience delivered by the uniquely designed powerful 16W speaker that makes your music as lively as it can get. While looking classic & trendy, the portable compact design comes with convenient loop strip makes it easy to bring the beat wherever you go. Built-In Microphone with Voice Assistant Support for you to enjoy a more convenient, hands-free calling experience. You can also access your favorite voice assistants from your speaker. The IPX7 waterproof rating ensures that you don't miss out on listening to your favorite music or podcast anywhere. This speaker could be the star of your pool parties, rain dance parties and your swimming sessions.

Rs 6999

Out of Stock

Out of Stock

ECorner

Product

Stocks
Futures & Options
Mutual Funds
Fixed deposits

Company


About
Products
Pricing
Careers
Press & Media

Support

Contact
Support Portals
List Of Charges
Downloads & Resources
Videos

Menu:

0123-45678
customer.ecorner@gmail.com
Welcome To E-Corner




Shopping Bag 0

Home
Sale
Charging
Laptop
Cables
Smart Watches
Earbuds
Gaming
Phone Cases

Search Bar:


[Home](#) [Sale](#) [Charging](#) [Laptop](#) [Cables](#) [Smart Watches](#) [Earbuds](#) [Gaming](#) [Phone Cases](#)

Charging



1000 W 8 Ports Charging Device
RS 11000

Add to Cart




Multiple Charging Devices
Rs 6999

Add to Cart

Cart:


[Home](#) [Sale](#) [Charging](#) [Laptop](#) [Cables](#) [Smart Watches](#) [Earbuds](#) [Gaming](#) [Phone Cases](#)

Smart Gadget




Xomi Ball
Rs 6999

Add to Cart




3 in 1 (Charger)
Rs 2500

Add to Cart




Out Door Camera
Rs 16000

Add to Cart



Mobile control RGB Bulb
Rs1500

Add to Cart



Electric scooty
RS 11000

Add to Cart

Your Cart



Bluetooth AirPods

Rs NaN

- 1 +

Remove

Total: Rs NaN

Checkout

```
1 // File: src/pages/cart.tsx
2
3 import Image from 'next/image';
4 import Header from '../app/components/header';
5 import Footer from '../app/components/footer';
6 import { useCart } from '@app/contexts/cartcontext';
7 import Link from 'next/link';
8
9 const Cart: React.FC = () => {
10   const { cartItems, removeFromCart, updateQuantity, getTotalPrice } = useCart();
11
12   // Increase the quantity for the given item
13   const handleIncrease = (id: string) => {
14     updateQuantity(id, 'increase');
15   };
16
17   // Decrease the quantity for the given item
18   const handleDecrease = (id: string) => {
19     updateQuantity(id, 'decrease');
20   };
21
22   return (
23     <div>
24       <Header />
25       <div className="container mx-auto px-4 py-8">
26         <h1 className="text-3xl font-bold text-center mb-8">Your Cart</h1>
27         {cartItems.length === 0 ? (
28           <p className="text-center text-gray-600">Your cart is empty.</p>
29         ) : (
30           <div className="grid grid-cols-1 gap-6">
31             {cartItems.map((item, index) => (
32               <div
33                 key={index}
34                 className="flex justify-between items-center border border-gray-200 rounded-lg p-4 shadow-lg bg-white"
35               >
36                 <div className="flex items-center">
37                   <Image
38                     src={item.imageUrl}
39                     alt={item.name}
40                     width={64}
41                     height={64}
42                     className="object-cover rounded-lg mr-4"
43                   />
44                   <div>
45                     <h2 className="text-lg font-semibold">{item.name}</h2>
46                     <p className="text-gray-600">
47                       Rs {
48                         {typeof item.price === 'number'
49                           ? item.price.toFixed(2)
50                           : Number(item.price).toFixed(2)}
51                       }
52                     </p>
53                     <div className="flex items-center mt-2">
54                       <button
55                         onClick={() => handleDecrease(item.id)}
56                         className="bg-gray-200 text-gray-800 py-1 px-2 rounded hover:bg-gray-300"
57                         disabled={item.quantity <= 1}
58                       />
59                     </div>
60                   </div>
61                 </div>
62               )
63             )}
64           </div>
65         )}
66       </div>
67     </div>
68   );
69 }
```

```

1 // file: src/app/contexts/cartcontext.tsx
2
3 import React, {
4   createContext,
5   useContext,
6   useState,
7   useEffect,
8   ReactNode,
9 } from 'react';
10
11 export interface CartItem {
12   id: string; // Unique identifier for the product (using slug in our case)
13   name: string;
14   price: number;
15   discountedPrice?: number | null;
16   imageUrl: string;
17   slug: string;
18   quantity: number;
19 }
20
21 interface CartContextType {
22   cartItems: CartItem[];
23   addToCart: (item: CartItem) => void;
24   removeFromCart: (id: string) => void;
25   updateQuantity: (id: string, action: 'increase' | 'decrease') => void;
26   clearCart: () => void;
27   getTotalPrice: () => number;
28 }
29
30 const CartContext = createContext<CartContextType | undefined>(undefined);
31
32 export const CartProvider = ({ children }: { children: ReactNode }) => {
33   const [cartItems, setCartItems] = useState<CartItem[]>([]);
34
35   // Load cart items from localStorage when the component mounts
36   useEffect(() => {
37     const storedCart = localStorage.getItem('cartItems');
38     if (storedCart) {
39       setCartItems(JSON.parse(storedCart));
40     }
41   }, []);
42
43   // Save cart items to localStorage whenever they change
44   useEffect(() => {
45     localStorage.setItem('cartItems', JSON.stringify(cartItems));
46   }, [cartItems]);
47
48   const addToCart = (item: CartItem) => {
49     setCartItems((prevItems) => {
50       // Check if the item is already in the cart using its unique id
51       const existingItem = prevItems.find((cartItem) => cartItem.id === item.id);
52       if (existingItem) {
53         // Increase the quantity if it already exists
54         return prevItems.map((cartItem) =>
55           cartItem.id === item.id
56             ? { ...cartItem, quantity: cartItem.quantity + 1 }
57             : cartItem
58         );
59       } else {
60         // Add new item to the cart
61         return [...prevItems, item];
62       }
63     });
64   };
65
66   const removeFromCart = (id: string) => {
67     setCartItems((prevItems) => prevItems.filter((item) => item.id !== id));
68   };
69
70   const updateQuantity = (id: string, action: 'increase' | 'decrease') => {
71     setCartItems((prevItems) =>
72       prevItems.map((item) => {
73         if (item.id === id) {
74           const newQuantity = action === 'increase' ? item.quantity + 1 : item.quantity - 1;
75           return { ...item, quantity: newQuantity };
76         }
77         return item;
78       })
79     );
80   };
81
82   const clearCart = () => {
83     setCartItems([]);
84   };
85
86   const getTotalPrice = () => {
87     const total = cartItems.reduce((sum, item) => sum + item.price * item.quantity, 0);
88     return total;
89   };
90
91   return (
92     <CartContext.Provider value={{
93       cartItems,
94       addToCart,
95       removeFromCart,
96       updateQuantity,
97       clearCart,
98       getTotalPrice,
99     }}>
100       {children}
101     </CartContext.Provider>
102   );
103 };

```

User Profile:



Search products...

Search

Shopping Bag 0

Create an Account

Full Name

Email Address

Password

Confirm Password

Sign Up

Already have an account? [Log In](#)

ECorner



Product

Stocks
Futures & Options
Mutual Funds
Fixed deposits

Company

About
Products
Pricing
Careers
Press & Media

Support

Contact
Support Portals
List Of Charges
Downloads & Resources
Videos

© 2025 All rights reserved by ecorner



Search products...

Search

Shopping Bag 0

Log In

Email Address

Password

Log In


Don't have an account? [Sign Up](#)

Document Title: "Day 5 - Testing and Backend Refinement - [E-Corner]"

Functional Testing:

0123-45678

customer.ecorner@gmail.com



gadets-hackathon.vercel.app says

3 in 1 (Charger) has been added to your cart.


OK

Shopping Bag

0


HomeSaleChargingLaptopCablesSmart WatchesEarbudsGamingPhone Cases

Smart Gadget




Xomi Ball
Rs 6999

Add to Cart




3 in 1 (Charger)
Rs 2500

Add to Cart




Out Door Camera
Rs 16000

Add to Cart



Mobile control RGB Bulb
Rs1500

Add to Cart




Electric scooty
RS 11000

Add to Cart

0123-45678

customer.ecorner@gmail.com

Welcome To E-Corner




Search products...

Search

Shopping Bag

0

Your Cart



3 in 1 (Charger)
Rs NaN

-

1

+

Remove


Total: Rs NaN

Checkout

0123-45678

customer.ecomer@gmail.com

Welcome To E-Corner



Search products...

Shopping Bag 0

Checkout

Order Summary


Product 1 x 2	\$40
Product 2 x 1	\$50
Total:	\$90

Shipping & Payment

0123-45678

customer.ecomer@gmail.com

Welcome To E-Corner



Search products...

Shopping Bag 0

Checkout

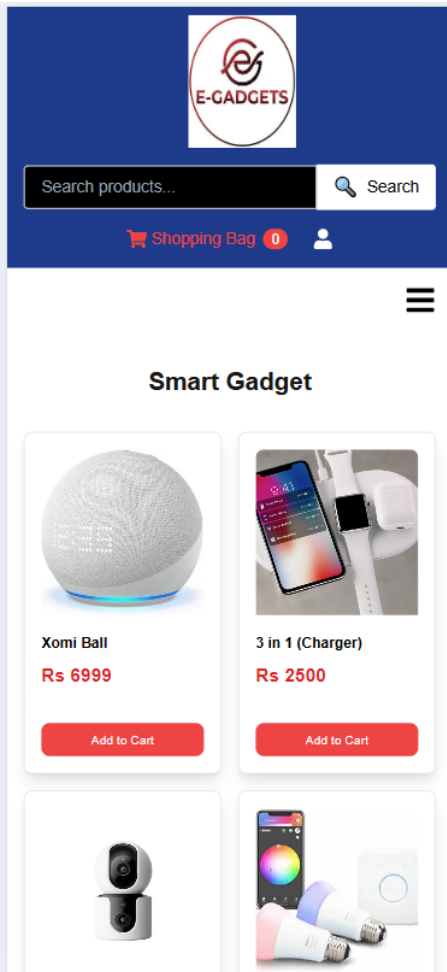
Order Placed Successfully!

Thank you for your purchase, SFADFI

Your tracking number is: [TRK-459108259](#)

You will receive an email confirmation shortly.

Responsiveness:



```

<head> </head>
<body class="antialiased" cz-
shortcut-listen="true">
<div id="__next" > </div>
</div>
<script id="__NEXT_DATA__"
type="application/json"> </script>
<next-route-announcer> </next-route-announcer>
<script src="/next/static/chunks/pages/signup-3d60c3b4ea68f175.js"></script>
<script src="/next/static/chunks/5862-05ae07e8c608f024.js"></script>
<script src="/next/static/chunks/5219-a44d1bac6957c3b6.js"></script>
<script src="/next/static/chunks/pages/index-4551f517c54f618d.js"></script>
<script src="/next/static/chunks/pages/gaming-f3a20d2f4a93cf62.js"></script>
<script src="/next/static/chunks/pages/phonecases-0971928f04-043d4c"></script>

```

html body.antialiased div#__next

Styles Computed Layout >>

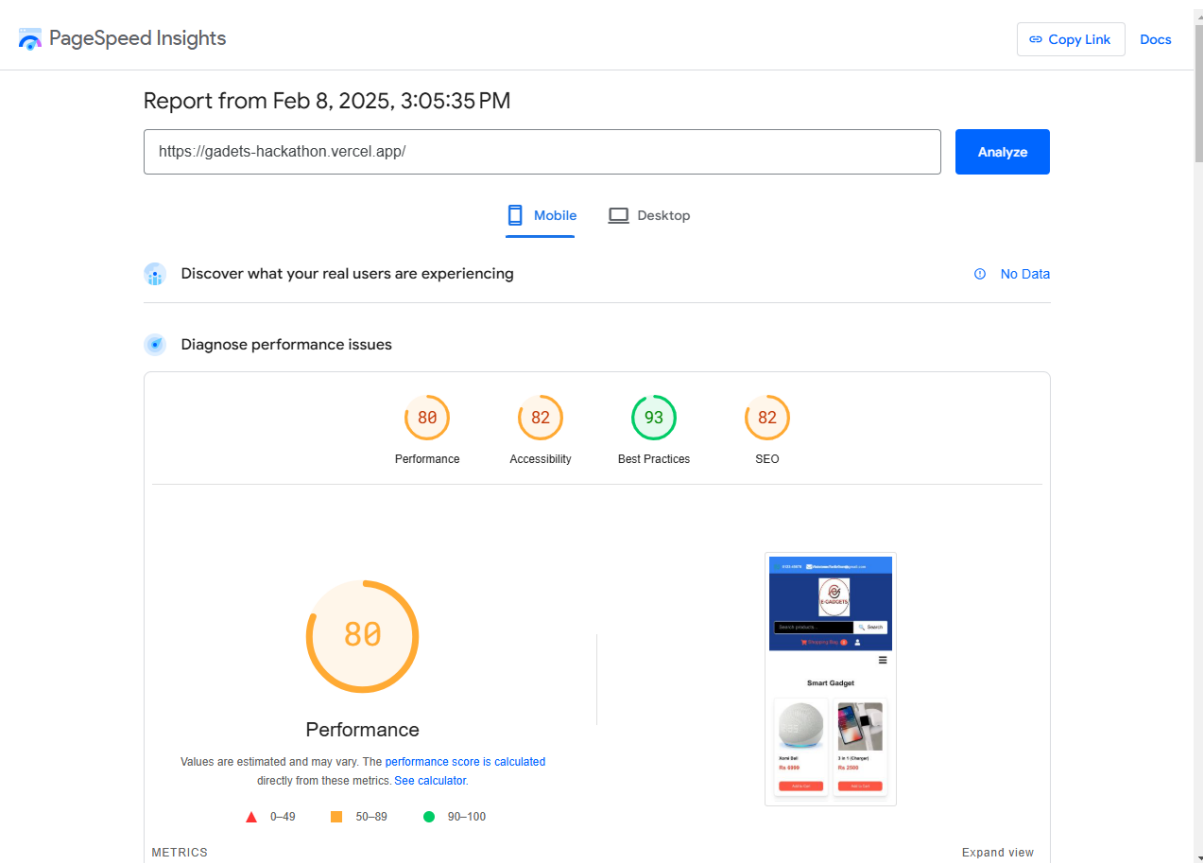
Filter show .cls + -

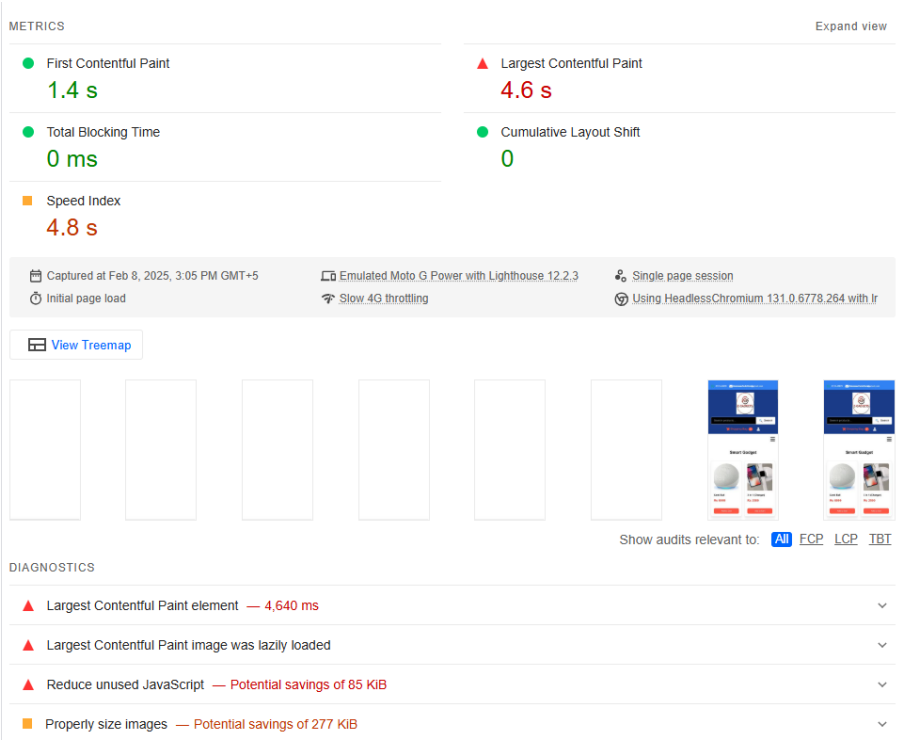
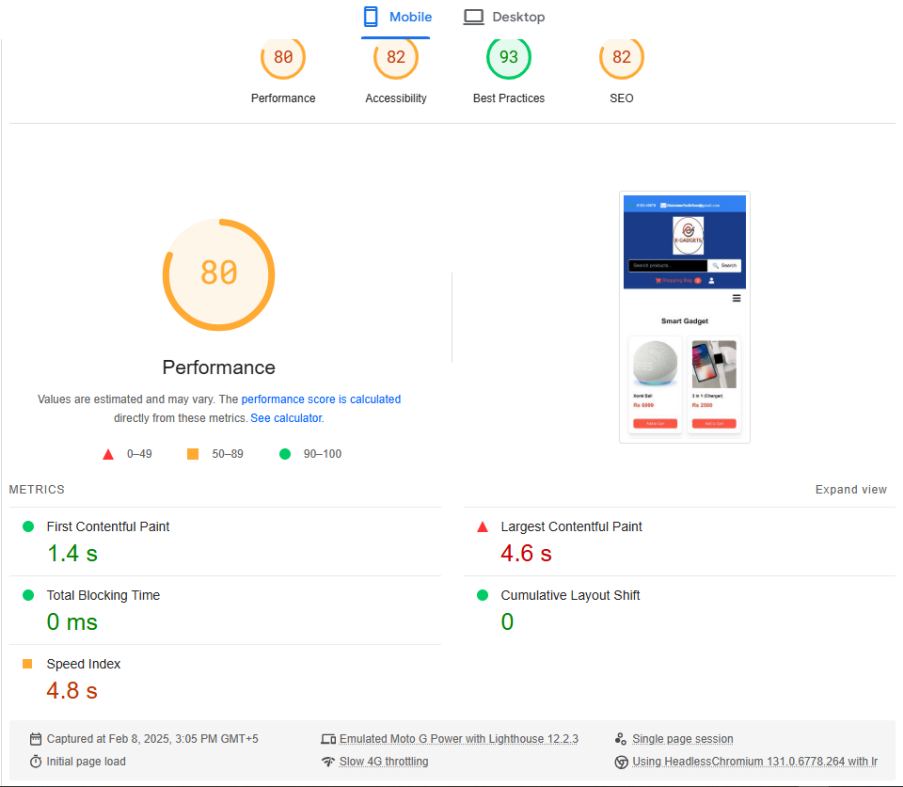
```

element.style {
}
*,
:after, :before {
  box-sizing: border-box;
  border: 0 solid #e5e7eb;
}
*,
:after, :before {
  --tw-border-spacing-x: 0;
  --tw-border-spacing-y: 0;
  --tw-translate-x: 0;
  --tw-translate-y: 0;
  --tw-rotate: 0;
  --tw-skew-x: 0;
  --tw-skew-y: 0;

```


									cefully
4	TC003	Check cart functionality	Add product to cart > Verify cart contents	Cart updates with added product	Cart updates as expected	Passed	High	-	Works as expected
5	TC004	Ensure responsiveness on mobile	Resize browser window > Check layout	Layout adjusts properly to screen size	Responsive layout working as intended	Passed	Medium	-	Test successful





DIAGNOSTICS	
▲ Largest Contentful Paint element — 4,640 ms	▼
▲ Largest Contentful Paint image was lazily loaded	▼
▲ Reduce unused JavaScript — Potential savings of 85 KiB	▼
■ Properly size images — Potential savings of 277 KiB	▼
○ Initial server response time was short — Root document took 320 ms	▼
○ User Timing marks and measures — 8 user timings	▼
○ Avoids enormous network payloads — Total size was 622 KiB	▼
○ Avoids an excessive DOM size — 304 elements	▼
○ JavaScript execution time — 0.1 s	▼
○ Minimizes main-thread work — 0.4 s	▼
○ Minimize third-party usage — Third-party code blocked the main thread for 0 ms	▼
More information about the performance of your application. These numbers don't directly affect the Performance score.	
PASSED AUDITS (25)	Show

 Mobile
  Desktop



Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.

NAMES AND LABELS

- ▲ Buttons do not have an accessible name ▼
- ▲ Document doesn't have a `<title>` element ▼

These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users of assistive technology, like a screen reader.

CONTRAST

- ▲ Background and foreground colors do not have a sufficient contrast ratio. ▼

These are opportunities to improve the legibility of your content.

TABLES AND LISTS

- ▲ Lists do not contain only `` elements and script supporting elements (`<script>` and `<template>`). ▼

Mobile

Desktop

TABLES AND LISTS

▲ Lists do not contain only elements and script supporting elements (<script> and <template>).

These are opportunities to improve the experience of reading tabular or list data using assistive technology, like a screen reader.

ADDITIONAL ITEMS TO MANUALLY CHECK (10)

Show

These items address areas which an automated testing tool cannot cover. Learn more in our guide on [conducting an accessibility review](#).

PASSED AUDITS (20)

Show

NOT APPLICABLE (33)

Show

93

Best Practices

USER EXPERIENCE

▲ Displays images with incorrect aspect ratio

Mobile

Desktop

82

SEO

These checks ensure that your page is following basic search engine optimization advice. There are many additional factors Lighthouse does not score here that may affect your search ranking, including performance on [Core Web Vitals](#). [Learn more about Google Search Essentials](#).

CONTENT BEST PRACTICES

▲ Document doesn't have a <title> element

▲ Document does not have a meta description

Format your HTML in a way that enables crawlers to better understand your app's content.

ADDITIONAL ITEMS TO MANUALLY CHECK (1)

Show

Run these additional validators on your site to check additional SEO best practices.

PASSED AUDITS (6)

Show

NOT APPLICABLE (2)

Show