

Testing, Error Handling and Backend Integration

The Day 5 is all about testing the document, error handling and backend integration. The tasks I have did in Day5.

1.Functionality Testing:

I ensure that all:

- ☐ The features and functionality have implemented properly.
- ☐ All the products have listed correctly with their dynamic data.
- ☐ All the data have stored in Sanity cms.



Timber Craft

Introducing TimberCraft—a collection that celebrates the timeless beauty of wood craftsmanship and the art of nature's finest materials. Inspired by t



AddToCart



Bright Space

Welcome to BrightSpace—a collection designed to infuse your home with light, energy, and vibrancy. Inspired by the power of natural light and open spa



AddToCart

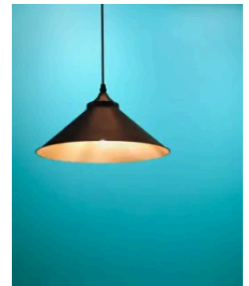


Vase Set

Elevate your home decor with the timeless beauty of the Vase Set. Designed to complement any interior style, this carefully curated set of vases bring



AddToCart



The Lucky Lamp

Introducing The Lucky Lamp—a unique blend of charm, elegance, and positive energy designed to illuminate your home and bring good fortune into your li



AddToCart

Syltherine



Introducing Syltherine – the ultimate fusion of elegance and power. Crafted for those who demand exceptional performance, Syltherine is a premium product designed to elevate your experience to the next level. Whether you're seeking innovation, beauty, or unmatched durability, Syltherine offers all this and more. This cutting-edge solution stands at the intersection of advanced technology and refined aesthetics. Its sleek design, combined with carefully selected materials, promises not only a visually stunning appearance but also long-lasting reliability. Whether you are using Syltherine for professional tasks or personal indulgence, it delivers with precision, speed, and unmatched performance. Perfectly balanced to meet the needs of modern users, Syltherine is intuitive and easy to use, providing a seamless experience from start to finish. It boasts a variety of features to ensure that you stay ahead of the curve, whether it's smart integration, energy efficiency, or superior functionality. Experience the future of performance with Syltherine—where luxury meets innovation, and every detail is engineered for perfection. Make Syltherine your go-to choice for an unparalleled experience today. Key Features: Superior performance and precision Sleek, modern design Built for durability and long-lasting use Easy integration and seamless user experience Ideal for both professionals and personal use Elevate your standards with Syltherine—where every moment is enhanced by innovation.

Price: \$200

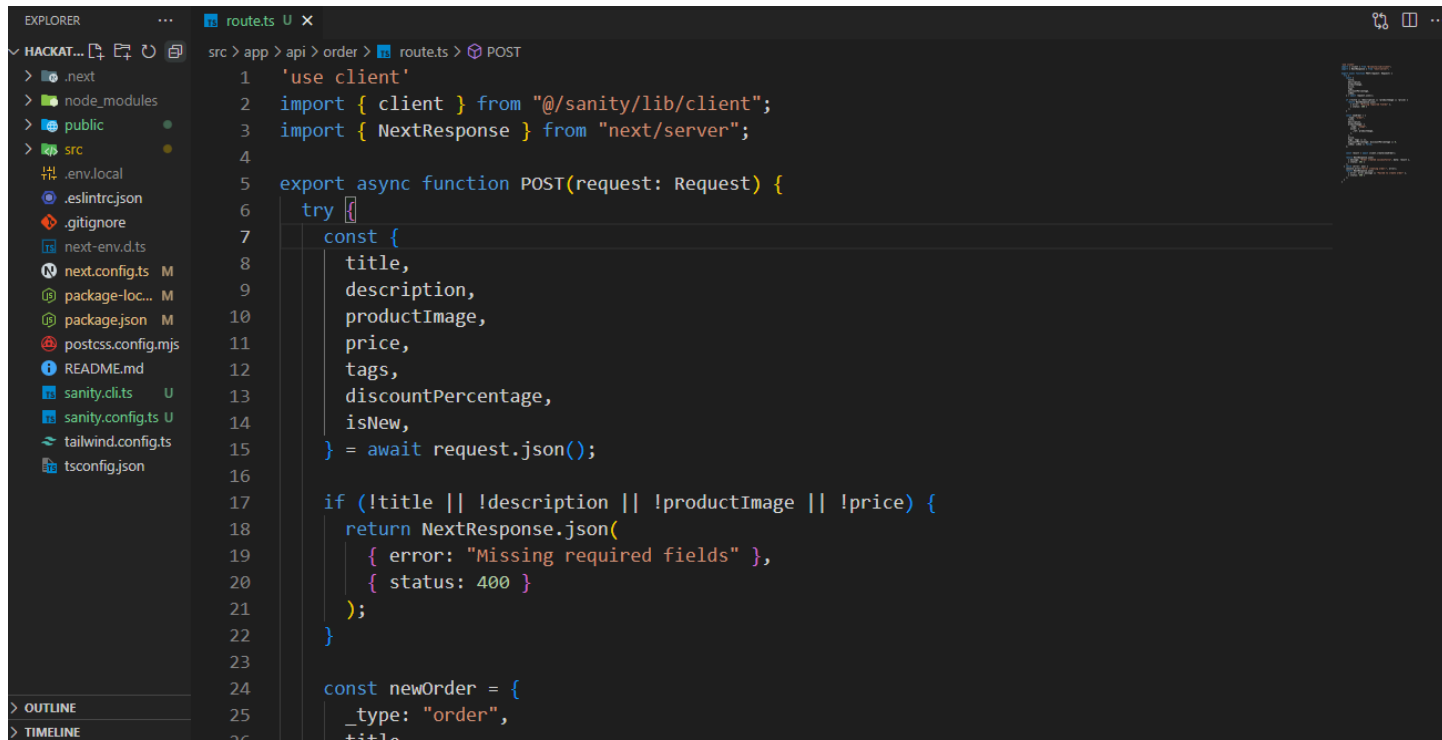
living fancy elegance desgin

[Shop Now](#)

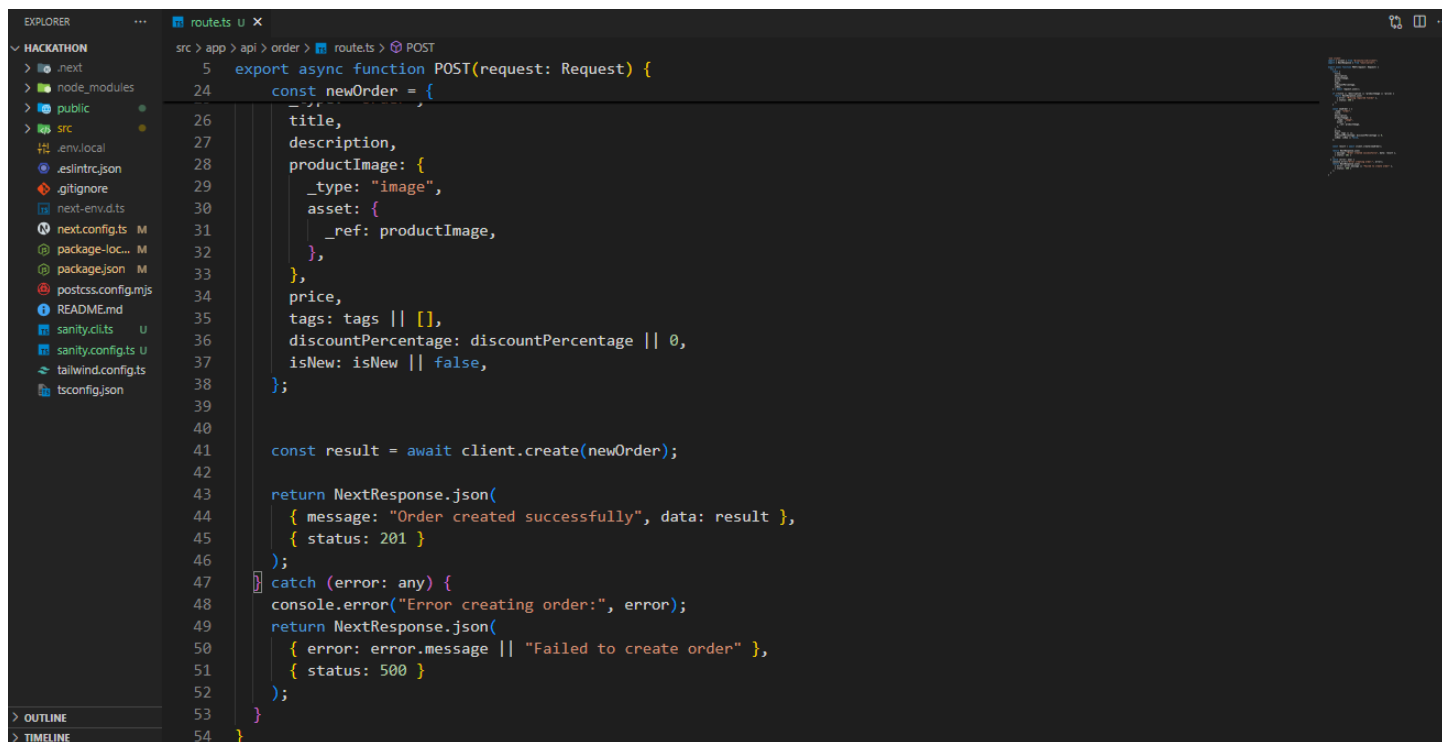
2. Error Handling:

- Implemented error testing, product details , product listing and add to cart functionality.
- React will notify with errors and success message.

```
1 import { client } from "@sanity/lib/client";
2 import { NextResponse } from "next/server";
3
4 export async function GET(request: Request) {
5   try {
6     const query = `*[_type == "product"]{
7       _id,
8       title,
9       description,
10      productImage{
11        asset->{
12          _id,
13          url
14        }
15      },
16      price,
17      tags,
18      discountPercentage,
19      isNew
20    }`;
21     const data = await client.fetch(query);
22
23     // Return a JSON response
24     return NextResponse.json(data);
25   } catch (error: any) {
26     console.error(error); // Log the error
27     return NextResponse.json({ error: error.message }, { status: 500 });
28   }
29 }
```



```
1 'use client'
2 import { client } from "@sanity/lib/client";
3 import { NextResponse } from "next/server";
4
5 export async function POST(request: Request) {
6   try {
7     const {
8       title,
9       description,
10      productImage,
11      price,
12      tags,
13      discountPercentage,
14      isNew,
15    } = await request.json();
16
17    if (!title || !description || !productImage || !price) {
18      return NextResponse.json(
19        { error: "Missing required fields" },
20        { status: 400 }
21      );
22    }
23
24    const newOrder = {
25      _type: "order",
26      title,
```

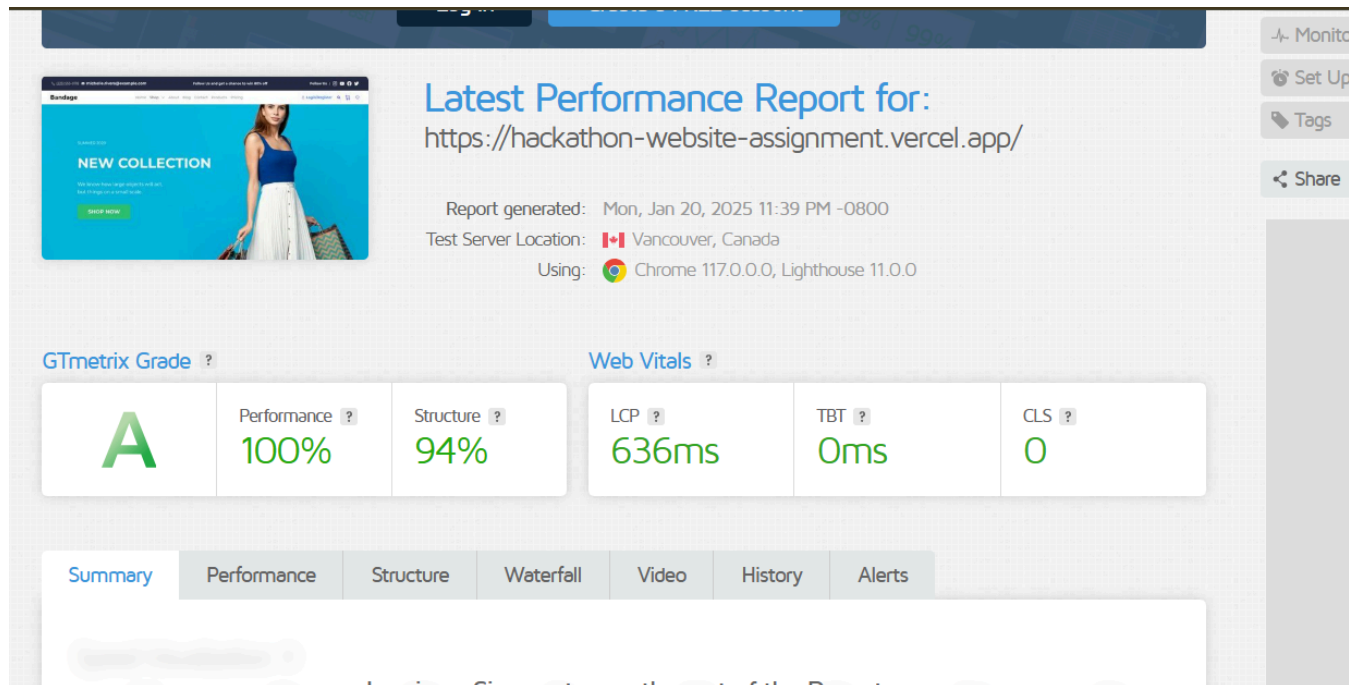


```
5 export async function POST(request: Request) {
24   const newOrder = {
26     title,
27     description,
28     productImage: {
29       _type: "image",
30       asset: {
31         _ref: productImage,
32       },
33     },
34     price,
35     tags: tags || [],
36     discountPercentage: discountPercentage || 0,
37     isNew: isNew || false,
38   };
39
40
41   const result = await client.create(newOrder);
42
43   return NextResponse.json(
44     { message: "Order created successfully", data: result },
45     { status: 201 }
46   );
47 } catch (error: any) {
48   console.error("Error creating order:", error);
49   return NextResponse.json(
50     { error: error.message || "Failed to create order" },
51     { status: 500 }
52   );
53 }
54 }
```

3. Performance Testing:

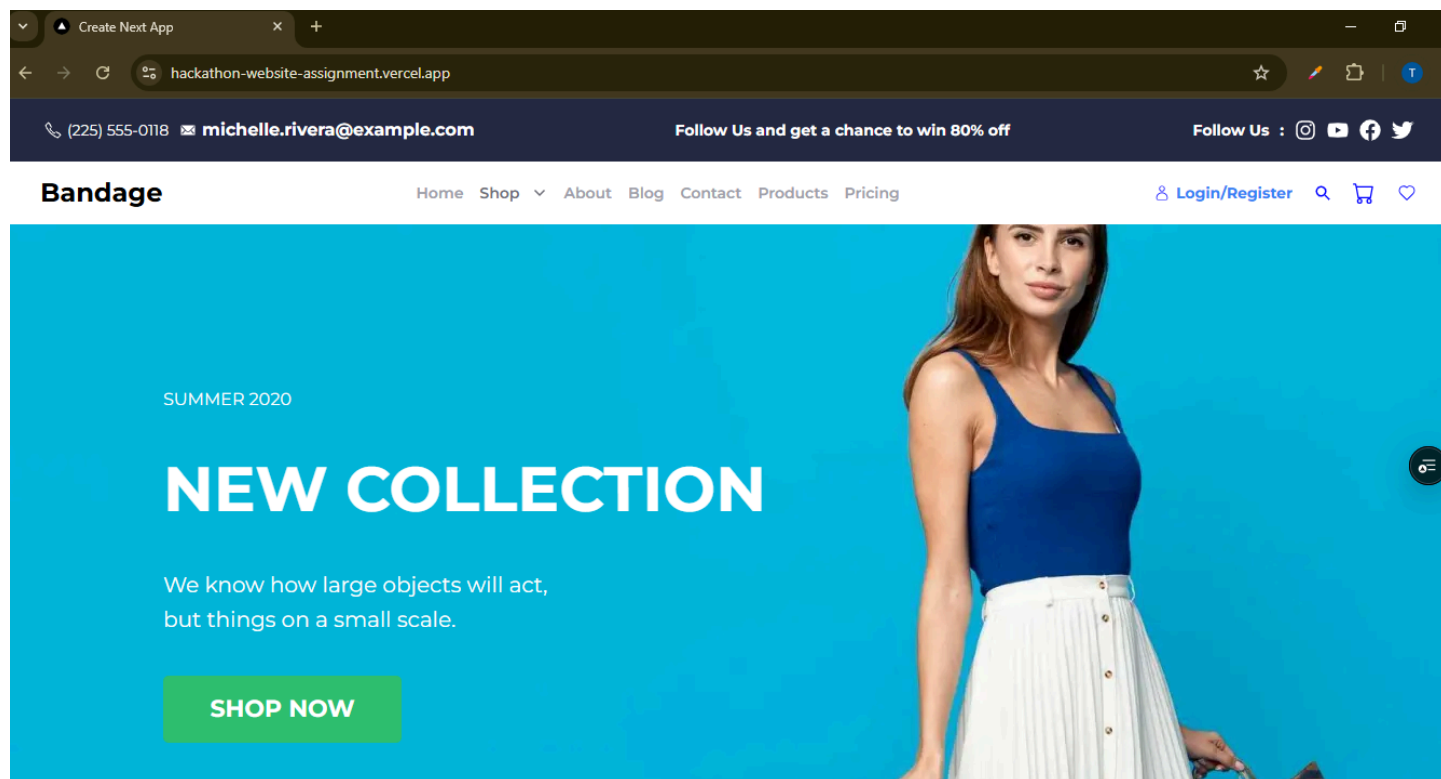
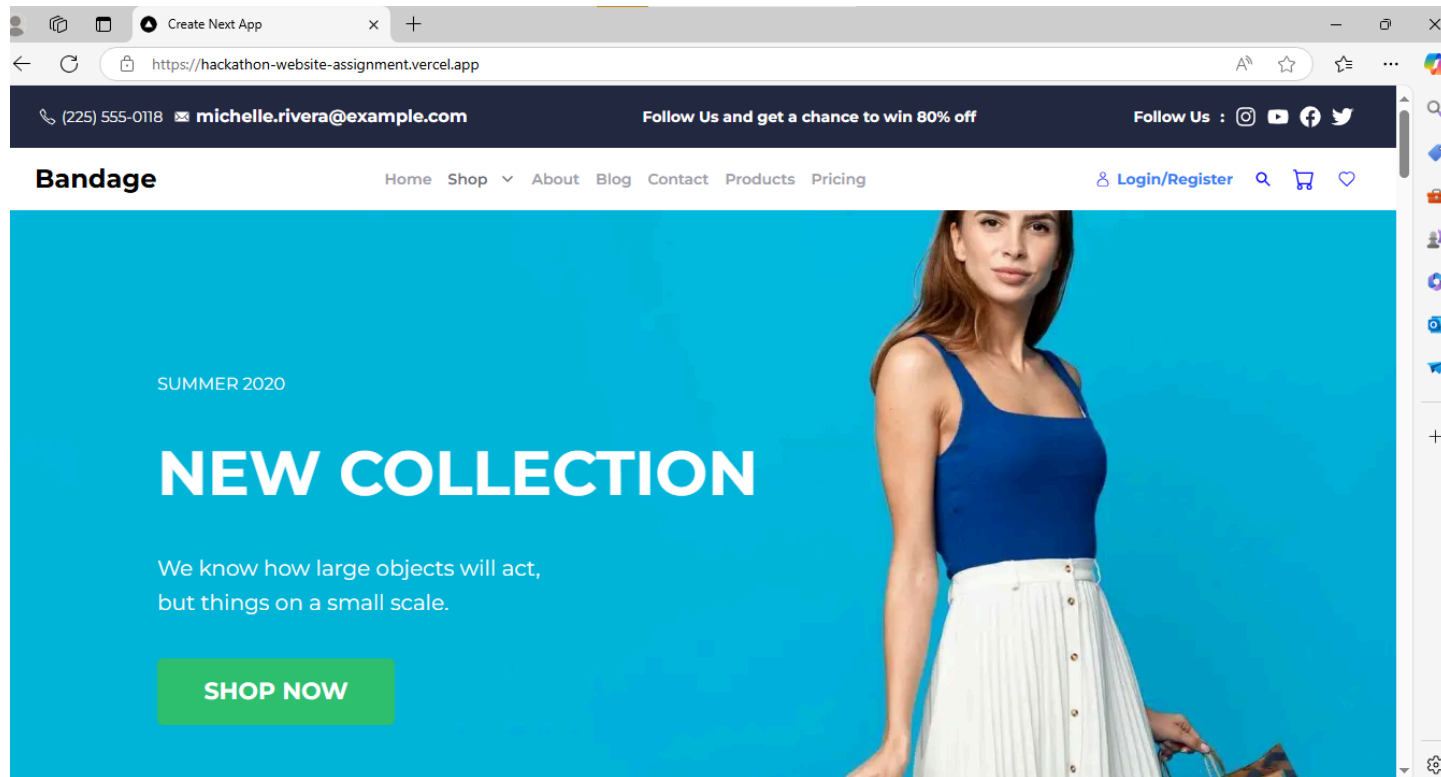
- Then , I have tested my website on gtmetrix.com.

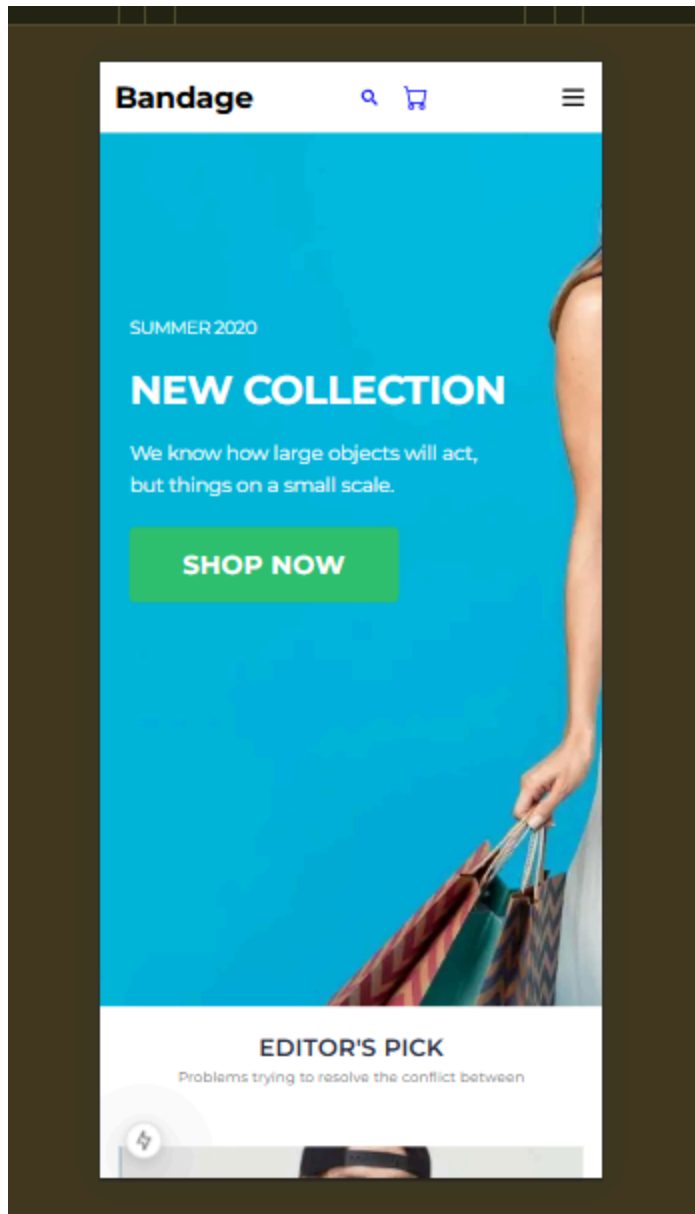
Here are the result:



4. Cross Browser and Device Testing:

- Tested website on google chrome, microsoft edge.
- To check its responsiveness on mobile screen, laptop and tab.





5. Security Testing:

When user supposed to add any product to cart then the user must have to fill personal information like name, email, address and contact number.

6. User Acceptance Testing:

Collected feedback from non tech users.

7.Documentation Update:

Updated documentation for changes in Schema.

Product Schema:

```
import { defineType } from "sanity"

export const product = defineType({
  name: "product",
  title: "Product",
  type: "document",
  fields: [
    {
      name: "title",
      title: "Title",
      validation: (rule) => rule.required(),
      type: "string"
    },
    {
      name: "description",
      type: "text",
      validation: (rule) => rule.required(),
      title: "Description",
    },
    {
      name: "productImage",
      type: "image",
      validation: (rule) => rule.required(),
      title: "Product Image"
    },
    {
      name: "price",
      type: "number",
```

```

        validation: (rule) => rule.required(),
        title: "Price",
      },
      {
        name: "tags",
        type: "array",
        title: "Tags",
        of: [{ type: "string" }]
      },
      {
        name: "discountPercentage",
        type: "number",
        title: "Discount Percentage",
      },
      {
        name: "isNew",
        type: "boolean",
        title: "New Badge",
      }
    ]
  })

```

Order Schema:

```

import { defineType } from "sanity"

export const order = defineType({
  name: "order",
  title: "Order",
  type: "document",
  fields: [
    {
      name: "title",

```



```
        title: "Title",
        validation: (rule) => rule.required(),
        type: "string"
    },
    {
        name: "description",
        type: "text",
        validation: (rule) => rule.required(),
        title: "Description",
    },
    {
        name: "productImage",
        type: "image",
        validation: (rule) => rule.required(),
        title: "Product Image"
    },
    {
        name: "price",
        type: "number",
        validation: (rule) => rule.required(),
        title: "Price",
    },
    {
        name: "tags",
        type: "array",
        title: "Tags",
        of: [{ type: "string" }]
    },
    {
        name: "discountPercentage",
        type: "number",
        title: "Discount Percentage",
```

```
    },  
    {  
      name: "isNew",  
      type: "boolean",  
      title: "New Badge",  
    }  
  ]  
}))
```

8. Conclusion:

This has enhanced features and functionality of website.