

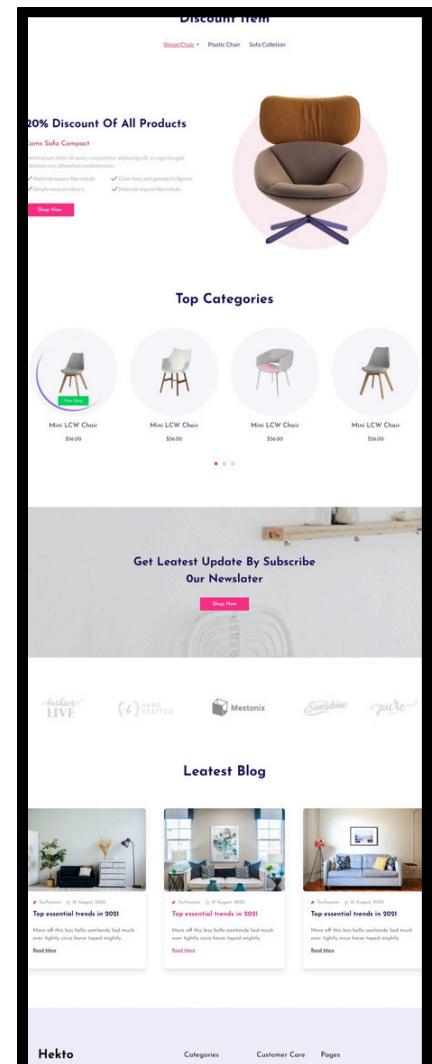
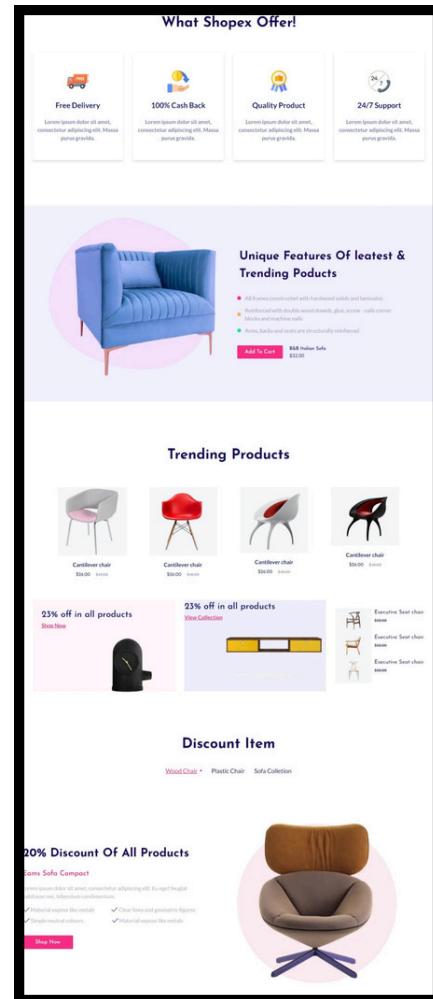
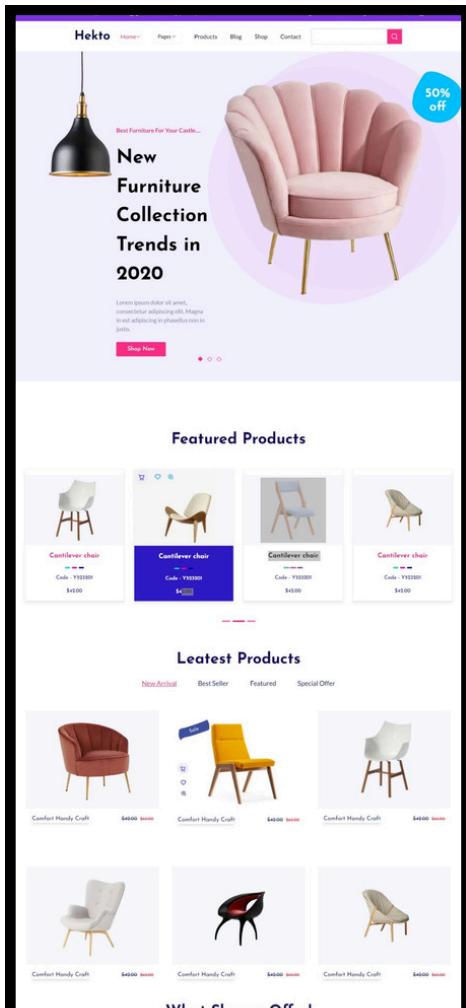
Day 4 - Dynamic Frontend Components - Haketo

1. Functional Deliverables:

- Screenshots or screen recordings showcasing:
 - The product listing page with dynamic data. ✓
 - Individual product detail pages with accurate routing and data rendering. ✓
 - Working category filters, search bar, and pagination. (working on it)
 - Any additional features implemented, such as related products. ✓

Screenshots are available in the **comparison folder** inside the document folder.

Before



After

The screenshot shows a user interface for a furniture store named 'Hekto'. At the top, there's a purple header bar with a user email ('mhhasanul@gmail.com'), a phone number ('(123)4567 890'), language ('English'), currency ('USD'), and a login/wishlist/cart section. Below the header, the main navigation menu includes 'Home', 'Pages', 'Products', 'Shop', and 'Contact'. A search bar is on the right. The main content area features a large banner with the text 'Product details' and a backlink to 'Home - Pages - Product details'. To the right of the banner, a large blue hand-drawn annotation reads 'Dynamic Router'. Below the banner, a product card for a 'Luxury Flower Shell Sofa Chair' is displayed, featuring a pink scalloped sofa chair, its price (\$2500.00), original price (\$3750.00), a 50% discount, and a stock level of 9. There are also 'Add to Cart' and 'share' buttons.

This section displays three separate screenshots of the Shopex website:

- Home Page:** Shows a large image of a pink sofa chair with a '50% off' discount. Text on the left says 'New Furniture Collection Trends in 2025'. Below it, a 'Featured Products' section shows three chairs: 'Uchiwa Quilted Lounge Chair' (yellow), 'Repos ThirtyNine Guest Chair' (yellow), and 'Uchiwa Quilted Lounge Chair' (grey). A 'Leatest Products' section follows, featuring a grid of furniture items.
- Offers Page:** Titled 'What Shopex Offer!', it highlights four service icons: 'Free Delivery', '100% Cash Back', 'Quality Product', and '24/7 Support'. Below these are sections for 'Unique Features Of leatest & Trending Products' (showing a blue armchair) and 'Trending Products' (showing four modern chairs).
- Product Details Page:** Similar to the Hekto example, it shows a large image of a blue armchair, its price (\$848), original price (\$1200), a 33% discount, and a stock level of 5. It also includes 'Add To Cart' and 'share' buttons. This page also features a 'Discount Item' section with a large image of a brown and grey armchair.

<https://muhammedsuhaihackathon2.vercel.app>

Featured Products

Nunc in ★★★
\$2000 ~~\$2000~~ ★★★★★
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magna in est adipiscin
in phasellus non in justo.

Uchiwa Quilted Lounge Chair ★★★★★
\$1600 ~~\$1600~~ ★★★★★
Code - 0e0ztuj

Yellow Chair ★★★★★
\$1300 ~~\$1300~~ ★★★★★
Rapson Thirty-Nine Guest Chair

<https://muhammedsuhaihackathon2.vercel.app/0b78c2a0-65d0-4e06-ac6c-0d7dd7bcd18>

Featured Products

Nunc in ★★★
\$2000 ~~\$2000~~ ★★★★★
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magna in est adipiscin
in phasellus non in justo.

Uchiwa Quilted Lounge Chair ★★★★★
\$1600 ~~\$1600~~ ★★★★★
Code - 0e0ztuj

Rapson Thirty-Nine Guest Chair ★★★★★
\$1300 ~~\$1300~~ ★★★★★
Code - 0e0ztuj

All products are dynamic

<https://muhammedsuhaihackathon2.vercel.app/0b78c2a0-65d0-4e06-ac6c-0d7dd7bcd18>

Nunc in ★★★
\$2000 ~~\$2000~~ ★★★★★
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magna in est adipiscin
in phasellus non in justo.

Curabitur lectus ★★★
\$540 ~~\$540~~ ★★★★★
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magna in est adipiscin
in phasellus non in justo.

Vitae facilisis ★★★
\$540 ~~\$540~~ ★★★★★
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magna in est adipiscin
in phasellus non in justo.

In nulla ★★★
\$2300 ~~\$2300~~ ★★★★★
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magna in est adipiscin
in phasellus non in justo.

Vel sem ★★★
\$1900 ~~\$1900~~ ★★★★★
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Magna in est adipiscin
in phasellus non in justo.

<https://muhammedsuhaihackathon2.vercel.app>

Shop Grid page

Home · Pages · Shop Grid page

Ecommerce Accessories & Fashion items

About 9,620 results (0.62 seconds)

Nunc in ★★★
\$2000.00 ~~\$2000.00~~ ★★★★★

Uchiwa Quilted Lounge Chair ★★★★★
\$1600.00 ~~\$1600.00~~ ★★★★★

Hans Wegner Style Three-Legged Shell Chair ★★★
\$990.00 ~~\$990.00~~ ★★★★★

Rapson Thirty-Nine Guest Chair ★★★
\$1300.00 ~~\$1300.00~~ ★★★★★

All dynamic

2. Code Deliverables:

Code snippets for key components:

1. ProductCard

```
export default async function Home() {
  <>
    <h1 className="mb-[48px] mt-[129px] text-center text-[42px] font-bold text-[#1A0B5B]">
      Featured Products
    </h1>
    <>
      <Carousel className="mx-auto">
        <CarouselContent>
          {data.map((product, index) => (
            <CarouselItem
              key={product._id}
              style={{ background: `url(${product.image}) no-repeat center / cover` }}>
              <div>
                <h2>${product.title}</h2>
                <p>${product.description}</p>
                <div>
                  ${product.price} ₪
                </div>
              </div>
            </CarouselItem>
          ))}
        </CarouselContent>
      </Carousel>
    </>
  </>
}
```

```
<CarouselItem
  key={product._id}
  className="flex flex-col items-stretch p-4 transition duration-300 ease-in-out [& hover:bg-[#f599c15b] hover:shadow-2xl md:basis-1/2 lg:basis-1/3 xl:basis-1/4 ${index === 0 ? "animate-nudge" : ""}]">
  <Link href={`/ ${product._id}`}>
    <div className="flex h-full items-center justify-center">
      <Card className="flex h-full flex-col rounded-none">
        <CardContent className="flex h-full flex-col items-center justify-center">
          /* head of card with img */
          <div className="flex items-center justify-center [& bg-[#F6F7FB]">
            <img
              src={urlFor(product.image.asset.url())}
              alt={product.name}
              width={201}
              height={201}
              loading="lazy"
              className="h-56 w-full object-cover"
            />
          </div>
        </CardContent>
        <span className="text-center text-lg font-bold [& text-[#FB2E86]">
          {product.name}
        </span>
      </Card>
    </div>
  </Link>
</CarouselItem>
```

2. ProductList

```
File Edit Selection View Go Run ... < > my-app 08 □ - □ x
```

```
page.tsx app page.tsx ...List M
```

```
app > List > page.tsx > ShopList
```

```
29  async function ShopList() {  
114  </div>  
112  /* list Section */  
113  <div className="flex flex-col gap-8" >  
114  <dataLP.map((product) => (  
115  <Link  
116  href={`/Grid/${product._id}`}  
117  key={product._id}  
118  >  
119  <div key={product._id} className="flex  
flex-col md:flex-row px-[4vh] md:px-0  
items-center justify-center gap-8">  
120  <div>  
121  <img  
122  src={urlFor(product.image.asset.url())}  
123  alt={product.name}  
124  width={313.63}  
125  height={217.56}  
126  className="h-[217.56px] w-[313.63px]  
rounded-[10px] object-cover"  
127  />  
128  </div>  
129  <div className="px-[4vh] md:px-0" >  
130  <div className="flex items-center gap-5  
text-[19px] font-semibold text-[#111C85]">  
131  {product.name}  
132  
```

```
29  async function ShopList() {  
114  <dataLP.map((product) => (  
144  <span className="ml-1 text-xs text-[#fb2e2e] line-through">  
145  $  
146  {  
147  parseFloat(product.price) *  
148  (1 + product.  
discountPercentage / 100)  
149  ).toFixed(2)}  
150  </span>  
151  </span> " "  
152    
154  </div>  
155  <p  
156  className="md:w-[591px] text-base  
font-normal text-[#9295AA]">  
157  {product.description}  
158  </p>  
159    
166  </div>  
167  </div>  
168  </Link>  
169  </div>  
170  </div>
```

```
x main* Launchpad 0 0 0 0 62 0 0 0 106 AWS: AWS Builder ID Amazon Q Ln 87, Col 16 Spaces: 2 UTF-8 LF () TypeScript JSX Go Live Prettier 23°C 7:47 PM
```

3. SearchBar

(will work on it soon).

Scripts or logic for API integration and dynamic routing:

```
import Footer from "@/components/Footer";
import Header from "@/components/Header";
import Mi from "@/components/Mi";
import Nvbr from "@/components/Navbar";
import { client } from "@/sanity/lib/client";
import { urlFor } from "@/sanity/lib/image";
import Image from "next/image";
import Link from "next/link";

interface Product {
  _id: string;
  name: string;
  image: { asset: { url: string } } | null;
  price: string;
  description: string;
  discountPercentage: number;
  isFeaturedProduct: boolean;
  stockLevel: number;
  category: "Chair" | "Sofa";
}

interface Params {
  params: {
    id: string;
  };
}

async function page({ params }: Params) {
  const resp: Product = await client.fetch(
    `*[_type = "product" && _id = ${params.id}]{
      name,
      image,
      price,
      description,
      discountPercentage,
      isFeaturedProduct,
      stockLevel,
      category
    }`,
    { id: params.id },
  );

  const similarProducts: Product[] = await client.fetch(
    `*[_type = "product" && category = ${resp.category} && _id != ${params.id}]{
      _id,
      name,
      image,
      price,
      discountPercentage
    }`,
    { category: resp.category, id: params.id },
  );
}

return (
  <div>
    <Header />
    <Nvbr />
    <Breadcrumb pageName="Product details" />
    <div className="container mx-auto my-12 px-4">
      {/* Product Section */}
      <div className="flex flex-col gap-6 md:flex-row">
        {/* Left: Product Image */}
        <div className="flex-1">
          <img
            src={resp.image ? urlFor(resp.image).url() : "/placeholder.png"}
            alt={resp.name}
            className="size-[325px] rounded-lg object-cover shadow-md"
          />
        </div>
        {/* mid: Product Details */}
        <div className="flex-1">
          <h1 className="mb-4 text-2xl font-bold text-gray-800">
            {resp.name}
          </h1>
          <p className="mb-2 text-xl text-gray-700">
            Price:{" "}
            <span className="font-bold text-[#FB2E86]">
              ${parseFloat(resp.price).toFixed(2)}
            </span>
          </p>
          {resp.discountPercentage > 0 && (
            <p className="text-md text-gray-600 line-through">
              Original Price: ${parseFloat(resp.price).toFixed(2)}
            </p>
          )}
        </div>
      </div>
    </div>
  </div>
)
```



```
<button className="bg-gray-200 rounded-lg flex px-6 mx-auto h-[39px] w-full font-medium duration-300 hover:scale-105 focus:outline-none focus:ring-1">
  Buy Now
</button>
</div>
</div>
/* Right: Buttons */
<div className="flex flex-1 flex-row items-center gap-4 md:flex-col">
  <button className="flex items-center gap-2 rounded-lg bg-[#FB2E86] px-4 py-2 text-white transition hover:bg-[#e02174]">
    Add to
    <Image
      width={20}
      height={20}
      src="/uil_heart-alt.png"
      alt="Wishlist"
      className="h-5 w-5"
    />
  </button>
  <button className="flex items-center gap-2 rounded-lg bg-[#FB2E86] px-4 py-2 text-white transition hover:bg-[#e02174]">
    <Image
      width={20}
      height={20}
      src="/share.png"
      alt="Wishlist"
      className="h-5 w-5"
    />
    share
  </button>
</div>
</div>

/* Product Details Section */
<div className="mt-12 rounded-lg bg-gray-100 p-4">
  <h2 className="mb-4 text-xl font-bold text-gray-800">
    Product Details
  </h2>
  <p className="text-gray-700">{resp.description}</p>
</div>

/* Similar Products Section */
<div className="mt-12">
  <h2 className="mb-6 text-xl font-bold text-gray-800">Similar Products</h2>
  <div className="grid grid-cols-1 gap-6 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4">
    {similarProducts.slice(0, 4).map((product) => (
      <div
        key={product._id}
        className="flex flex-col items-center rounded-lg bg-white p-4 shadow-md transition hover:shadow-xl"
      >
        <img
          src={
            product.image
              ? urlFor(product.image).url()
              : "/placeholder.png"
          }
          alt={product.name}
          className="h-48 w-full rounded-t-lg object-cover"
        />
        <h3 className="mt-4 text-lg font-bold text-gray-800">
          {product.name}
        </h3>
        <p className="text-gray-700">
          Price: ${parseFloat(product.price).toFixed(2)}
        </p>
        {product.discountPercentage > 0 && (
          <p className="text-sm text-gray-500 line-through">
            $
            {
              parseFloat(product.price) *
              (1 + product.discountPercentage / 100)
            ).toFixed(2)
          </p>
        )}
      <Link
        href={`/ ${product._id}`}
        className="mt-4 rounded-lg bg-[#FB2E86] px-4 py-2 text-white transition hover:bg-[#e02174]"
      >
        View Details
      </Link>
    </div>
  )));
</div>
</div>
</div>
<Footer />
<Mi />
</div>
);
}

export default page;
here
```

Completed Tasks from Day 2:

- ★ Integrated a carousel from ShadCN ✓
- ★ Added additional tags in the schema, such as "Featured," "Latest Products," "Trending," etc., along with extra details ✓
- ★ Added more products ✓
- ★ Prepared necessary data for frontend functionality ✓

Page Name	Static	Dynamic
Home	✓	✓
Listing Product	✓	✓
Product Details	✓	✓
Cart	✓	Needs work
Checkout	✓	Needs work
order	✓	Needs work
order complete	✓	No dynamic additions needed

Day 4 Progress:

- ★ Successfully rendered and displayed data on the UI without issues.
- ★ Attempted to create a reusable component for dynamic routes, but it required the page to be client-side. Eventually, abandoned the approach and handled it manually.