



# Day\_4\_Building\_Dynamic\_Frontend\_ Components\_for\_Your\_Marketplace.

Presented by Hadiqa Gohar

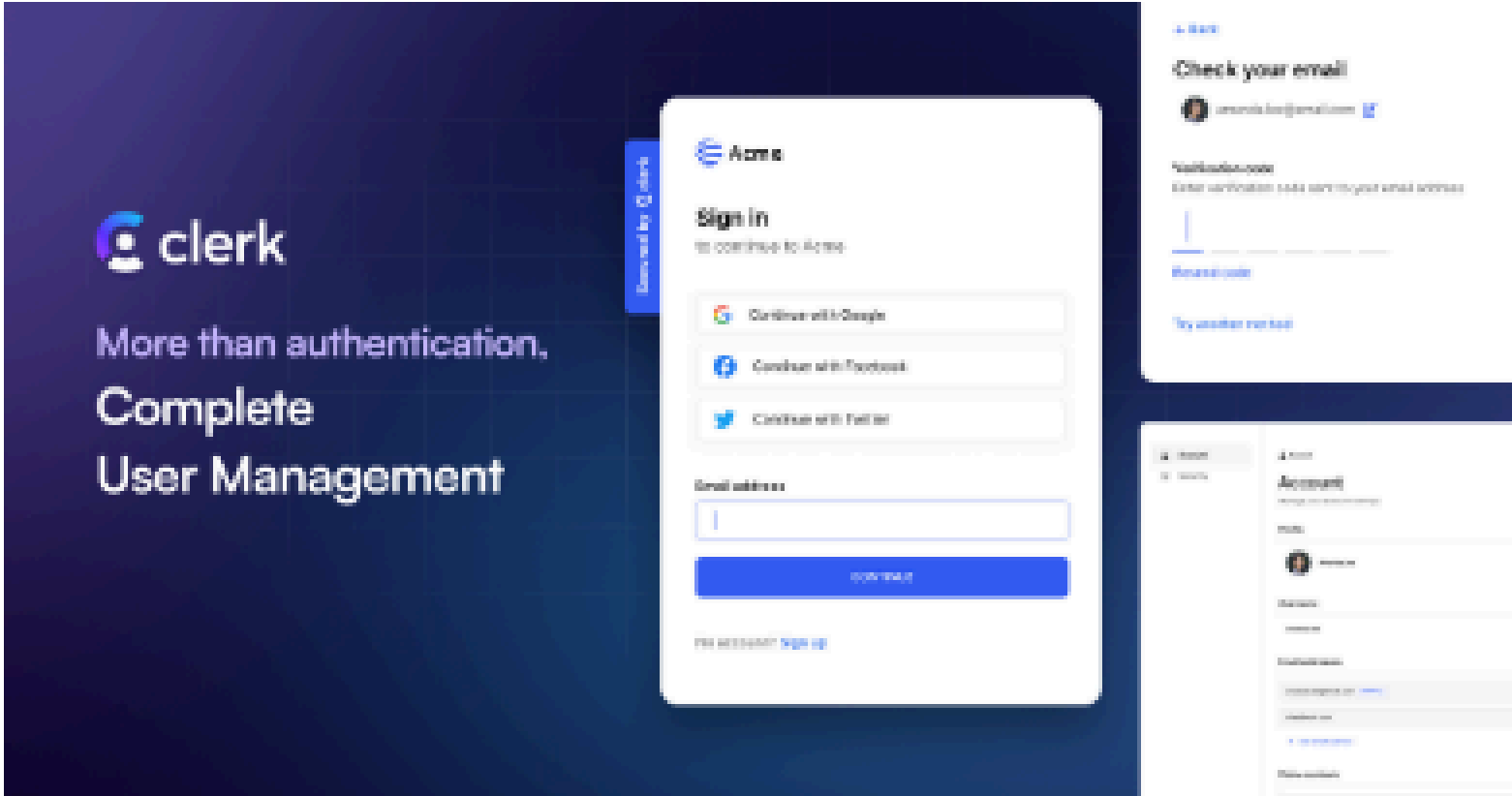


# Day 4 - Dynamic Frontend Components for E-commerce Website

On Day 4, I focused on designing and developing dynamic frontend components to enhance the functionality and user experience of my e-commerce website. The primary features I implemented include:

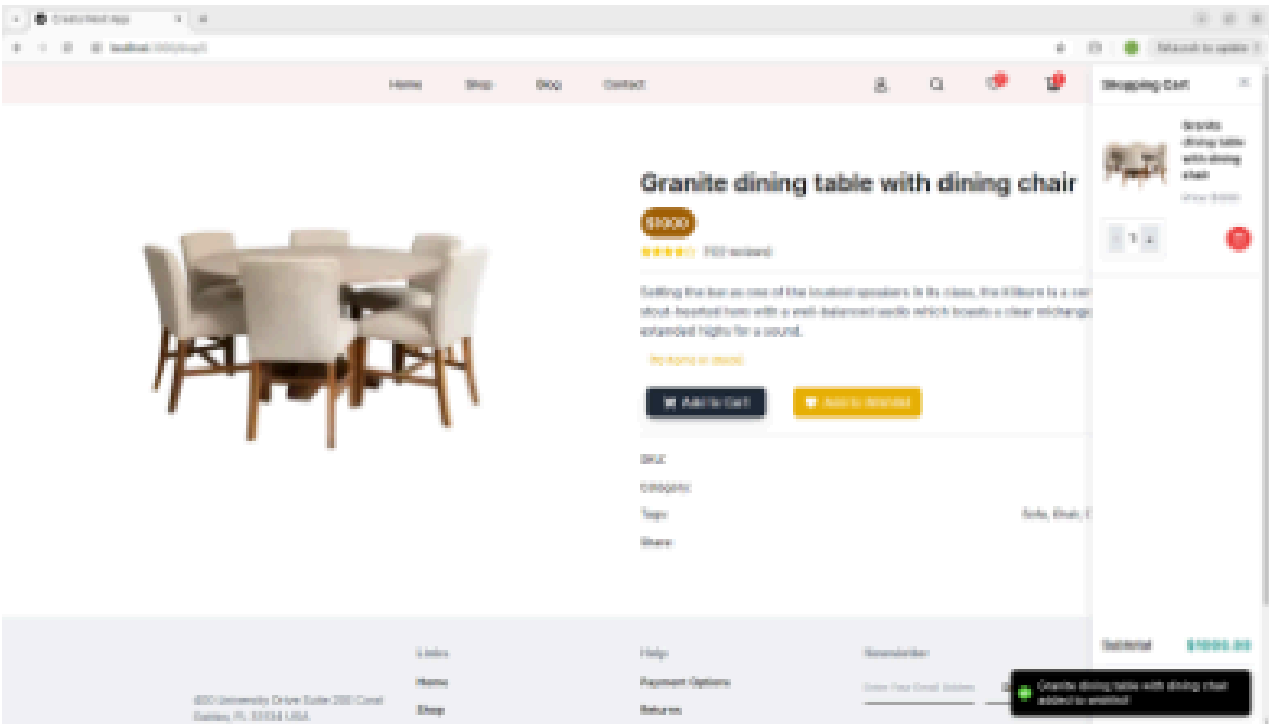
## Clark Auth Integration:

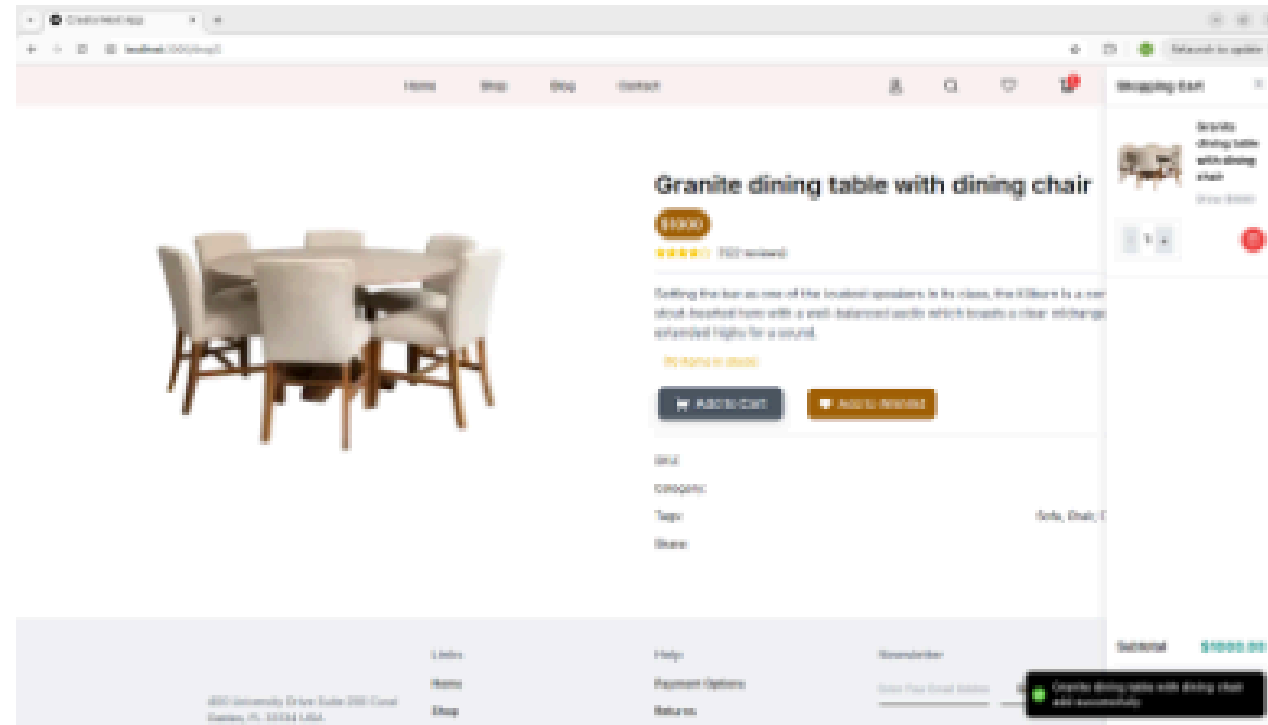
Initially, I implemented Clerk Authentication on my website to ensure that user data, including passwords, orders, and other sensitive information, remains secure. Clerk Auth provides robust security features and simplifies user management, making it an excellent choice for safeguarding sensitive operations on the platform."



## Features:

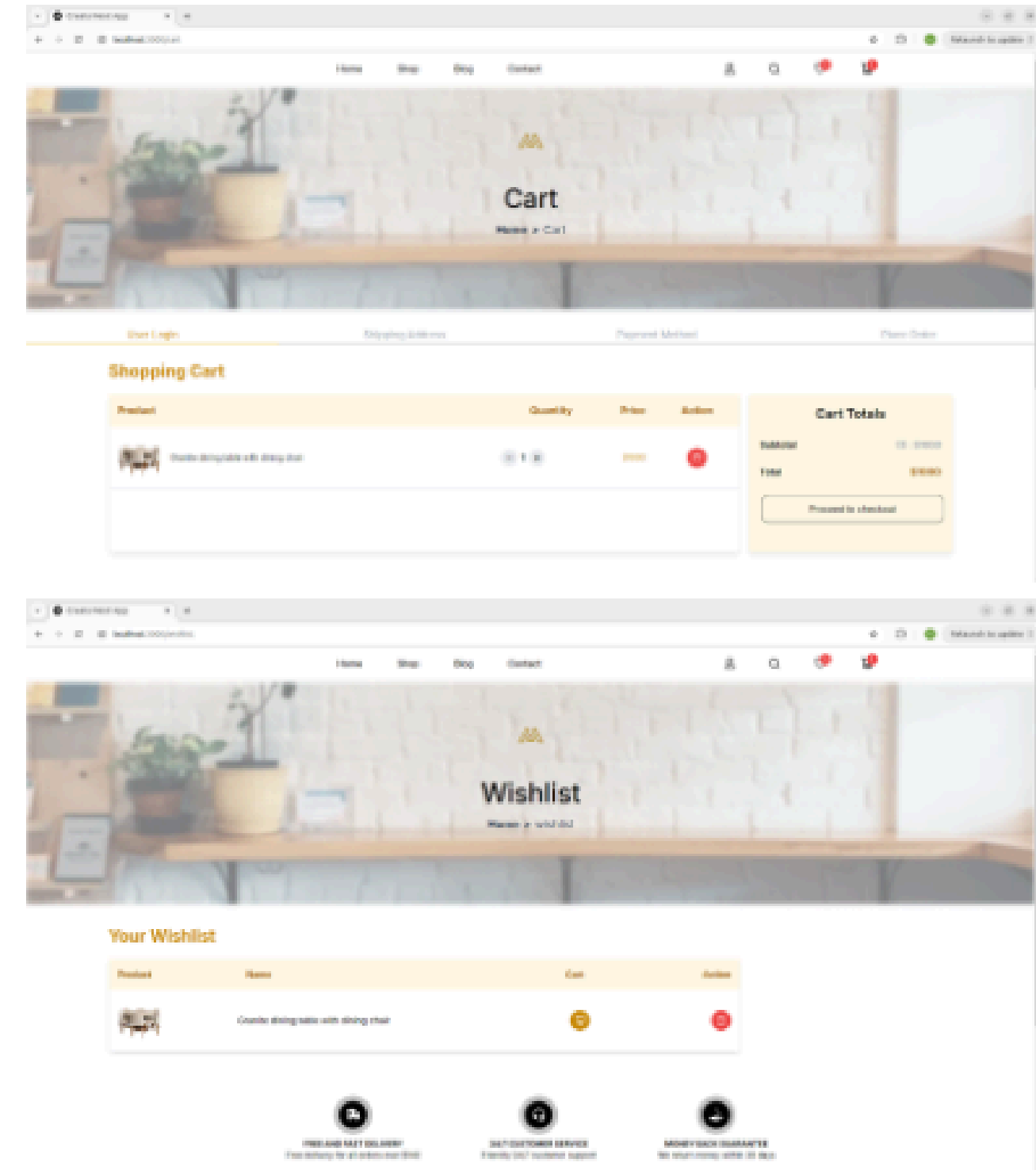
I have integrated features like Add to Cart, Wishlist, and a Cart Sidebar into my website to enhance user experience. Users can easily add products to their cart or wishlist, with functionality to increase, decrease, add, or delete products as needed. This ensures a seamless and interactive shopping experience for the users.





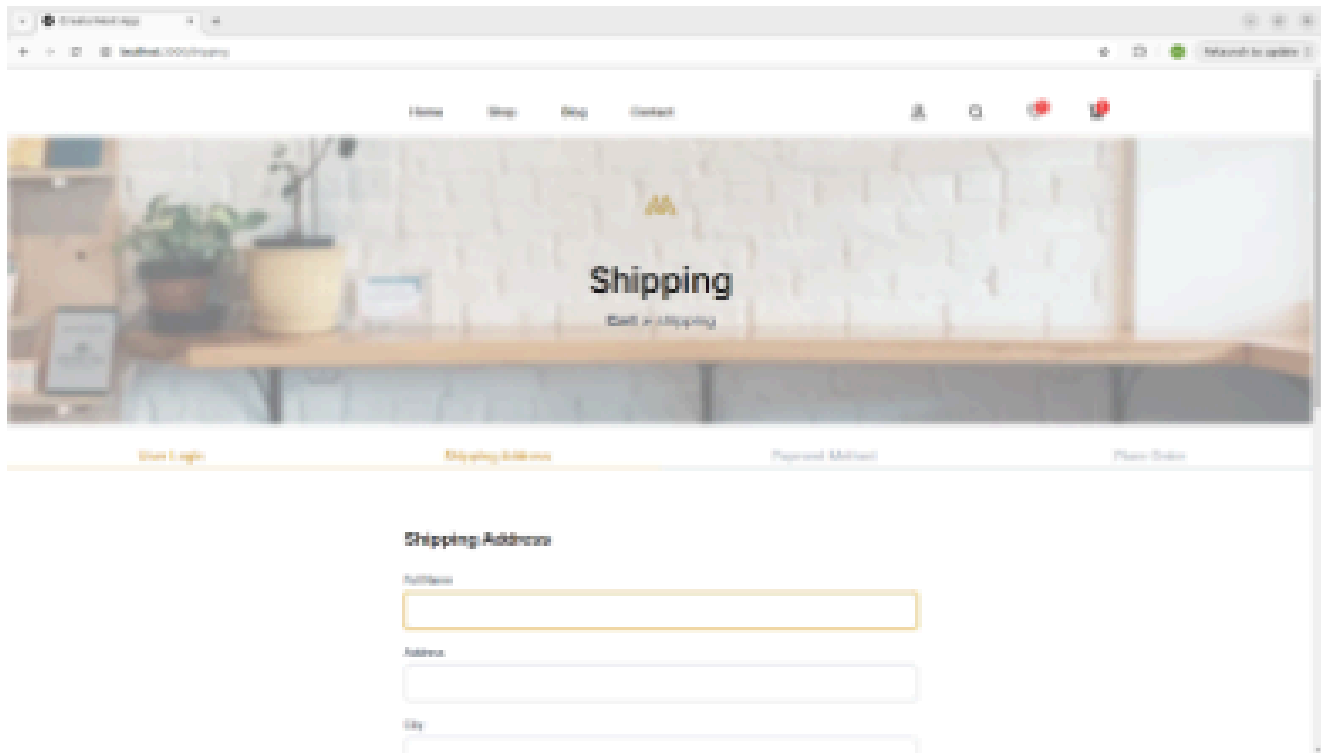
## Cart and Wishlist Page :

I created dedicated pages for Add to Cart and Wishlist on my website. These pages allow users to view and manage their selected items conveniently, providing options to adjust quantities, remove products, or move items between the cart and wishlist. This setup enhances user interaction and streamlines the shopping process.

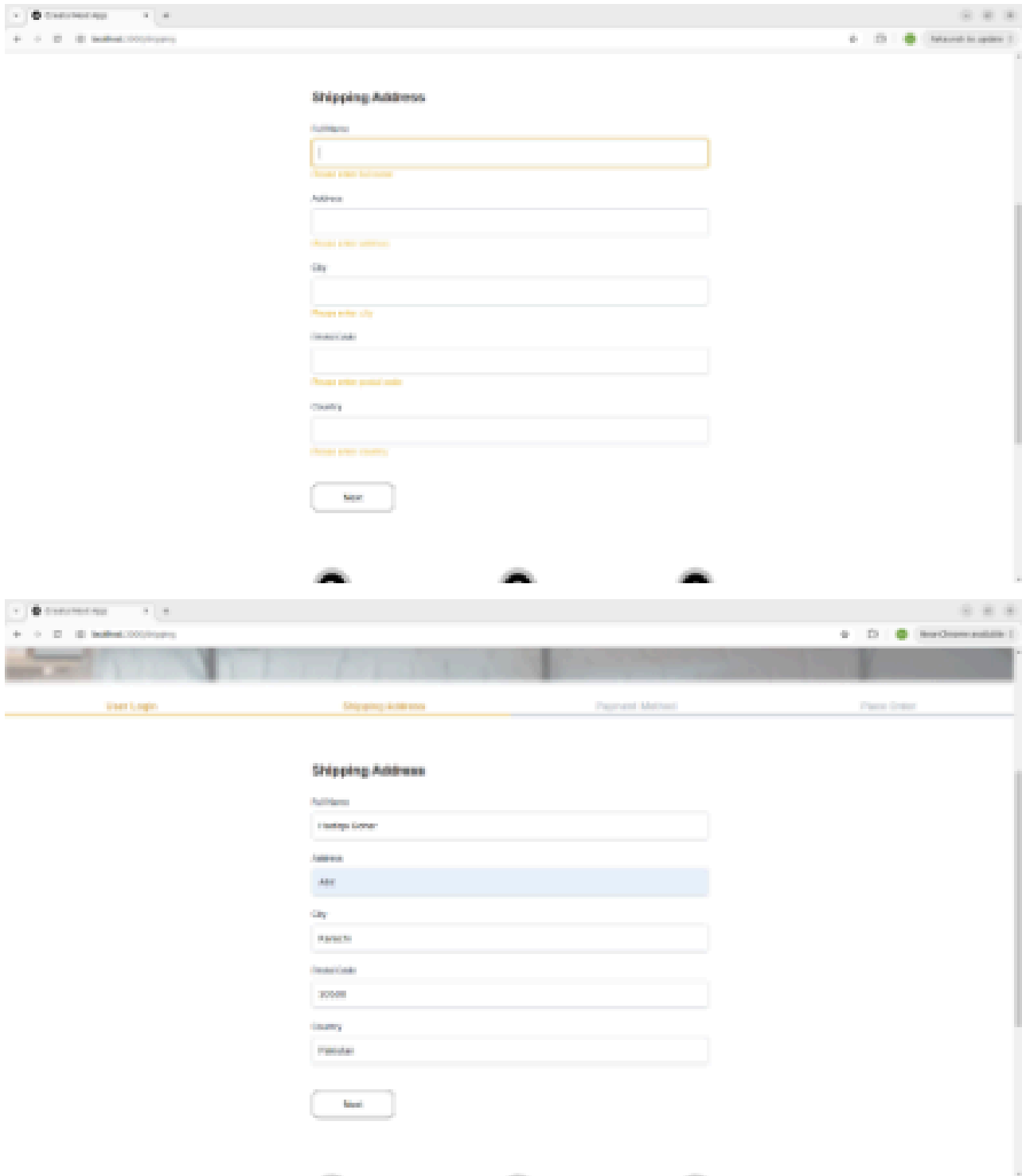


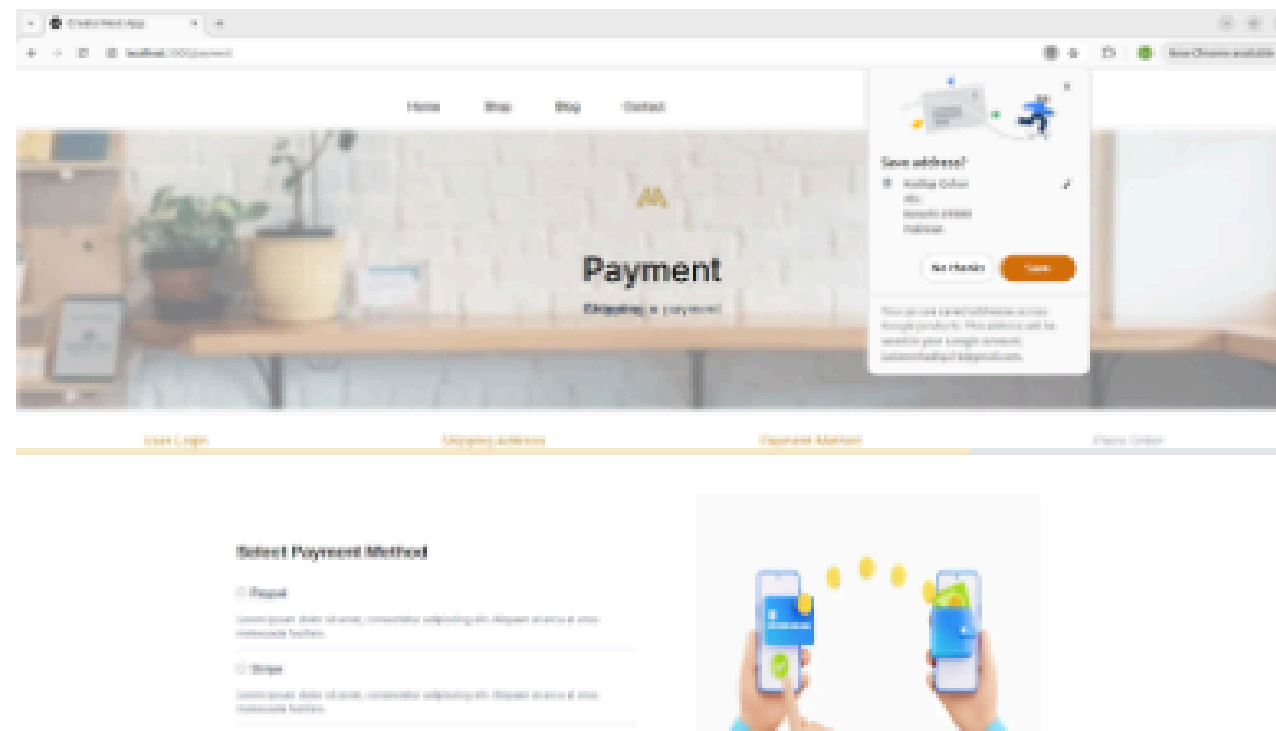
## Functionality:

I have implemented shipping and checkout functionality on my website. This allows users to seamlessly proceed from adding items to their cart or wishlist to completing their purchase. The checkout process includes options for selecting shipping methods, entering delivery details, and finalizing payments, ensuring a smooth and efficient shopping experience.



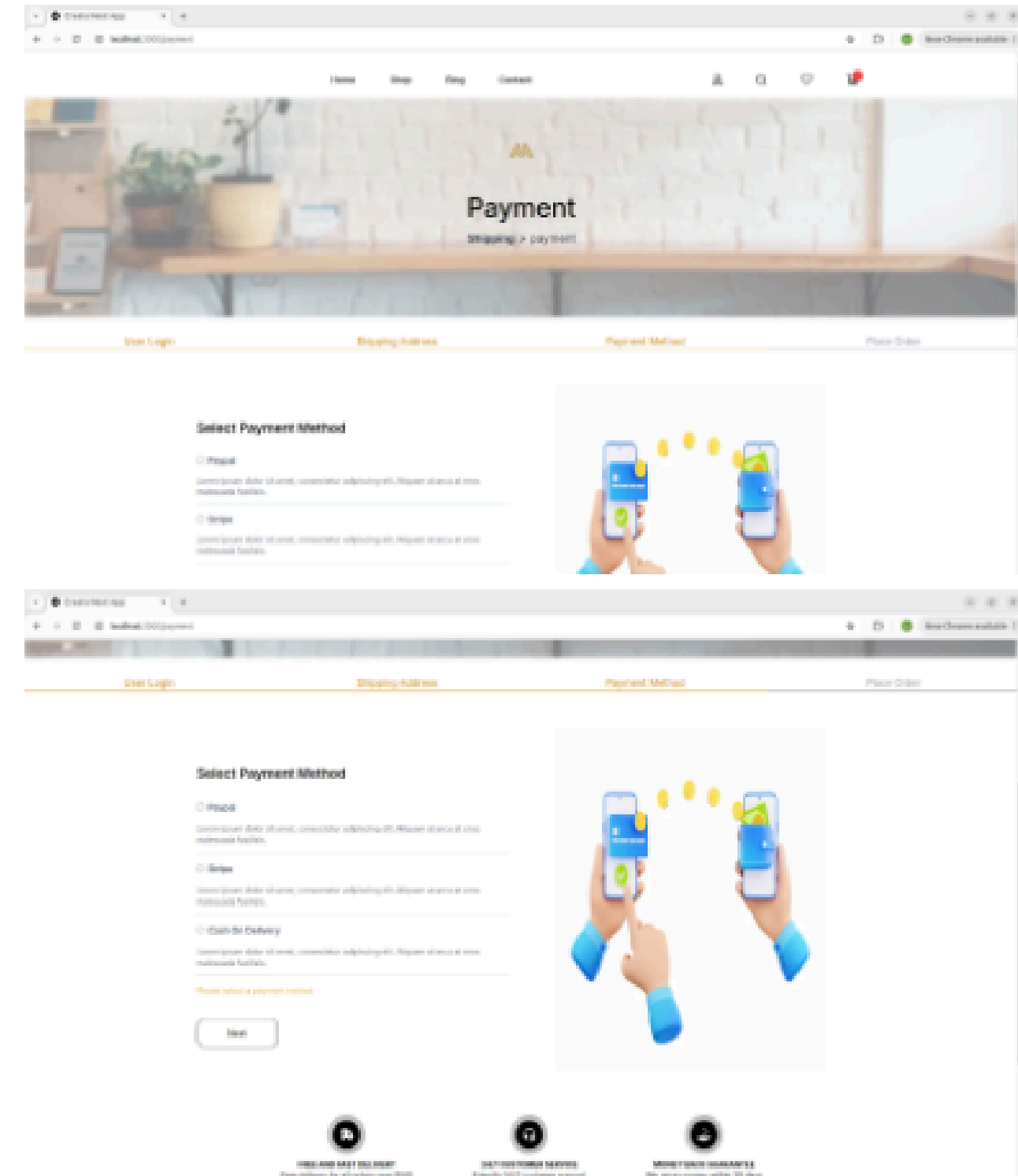
I have also integrated alert functionalities to improve the user experience during the checkout process. These alerts notify users to provide essential details such as their name, address, and other required information, ensuring that all necessary fields are completed before proceeding. This helps to prevent errors and ensures a seamless checkout experience.

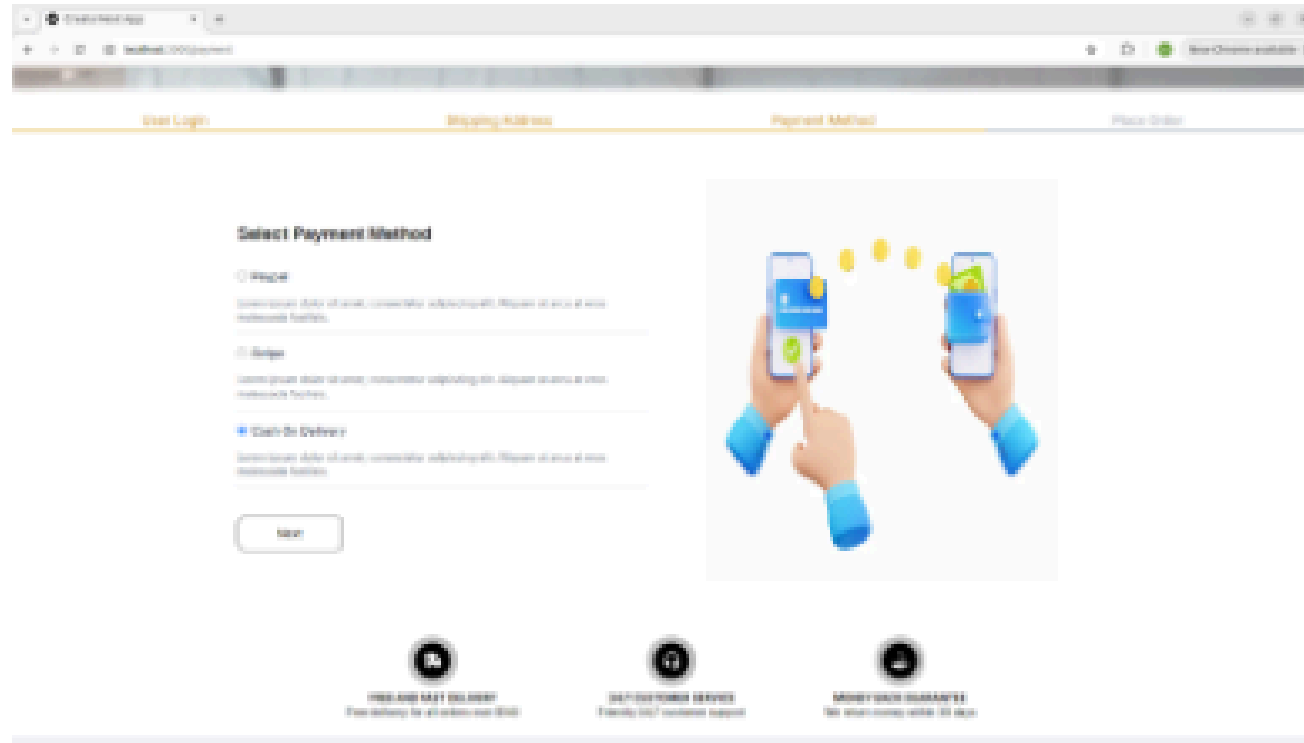




### Payment Integration:

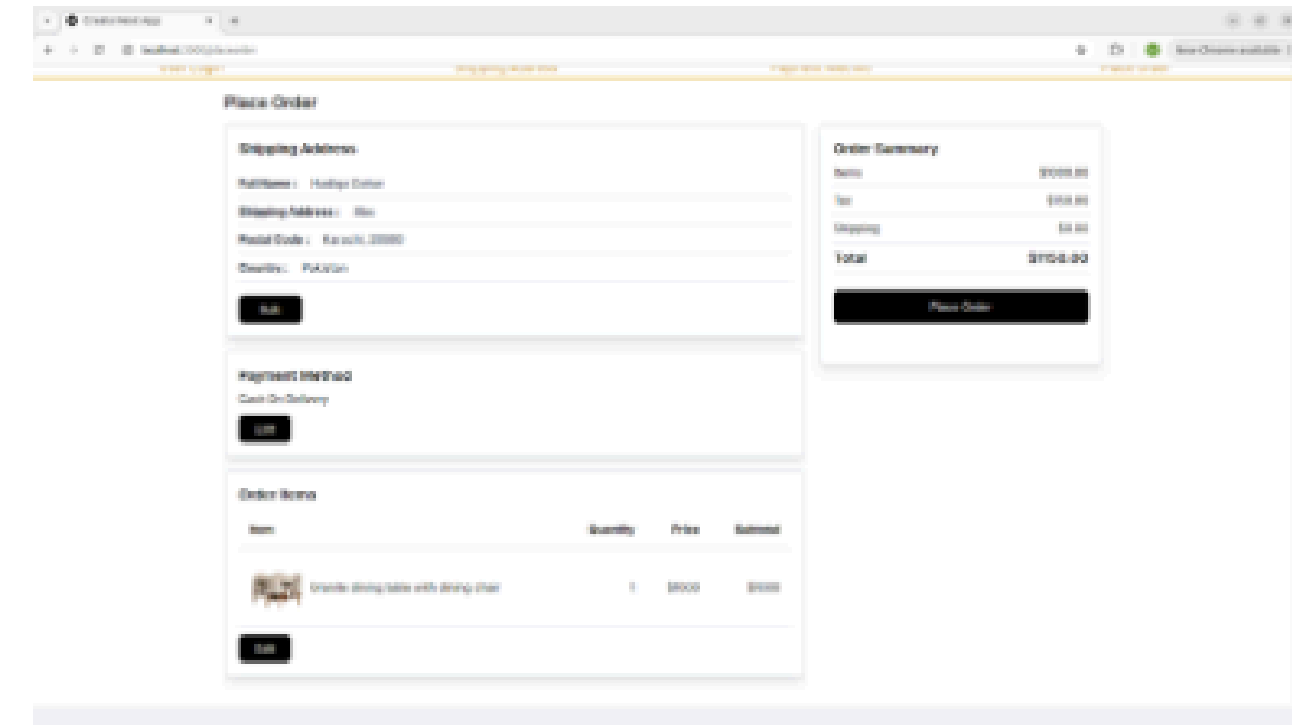
I have added a payment method selection feature with checkboxes, including options like Stripe, PayPal, and Cash on Delivery. Users must select a payment method before proceeding to the order placement page. This ensures that the payment preference is confirmed, streamlining the checkout process and avoiding incomplete transactions.





## Order Placement Page:

The last page, 'Order Placement,' displays all relevant details of the user's order, including the list of products, shipping address, payment method, and a summary of the total amount. It also shows the shipping cost and payment details. Each section has an 'Edit' button, allowing users to modify their information if needed before finalizing the order. This ensures users can double-check all details, including the total amount and payment method, and make any necessary changes before placing the order.



## Sanity Schema Creation:

Designed and implemented schemas in Sanity CMS to manage dynamic product data effectively, including fields for product details, images, pricing, and categories

```
1 const productSchema = {
2   name: 'product',
3   type: 'document',
4   title: 'Product',
5   fields: [
6     {
7       name: 'id',
8       type: 'string',
9       title: 'Product ID',
10      description: 'Unique identifier for the product',
11    },
12    {
13      name: 'name',
14      type: 'string',
15      title: 'Product Name',
16    },
17    {
18      name: 'description',
19      type: 'string',
20      title: 'Description',
21    },
22    {
23      name: 'price',
24      type: 'number',
25      title: 'Product Price',
26    },
27    {
28      name: 'rating',
29      type: 'number',
30      title: 'Rating',
31      description: 'Rating of the product',
32    },
33    {
34      name: 'ratingCount',
35      type: 'number',
36      title: 'Rating Count',
37      description: 'Number of ratings',
38    },
39    {
40      name: 'countInStock',
41      type: 'number',
42      title: 'Quantity',
43      description: 'Quantity of the product',
44    },
45    {
46      name: 'sku',
47      type: 'string',
48      title: 'SKU',
49    },
50    {
51      name: 'tags',
52      type: 'string',
53      title: 'Tags',
54    },
55  ]
56 }
```