**MINOR PROJECT REPORT**

Subject code - BCA 204

On

**E-Commerce Website**

**Submitted in partial fulfilment of the requirement of Bachelors of Computer Applications (BCA)**

**Guru Gobind Singh Indraprastha University, Delhi**



**Session 2020-23**

(FONT STYLE- TNR BOLD/FONT SIZE 14)

Under the Guidance of Submitted by

Designation

BCA-V Sem

Enrollment No.



JIMS ENGINEERING MANAGEMENT TECHNICAL CAMPUS

48/4 Knowledge Park III, Greater Noida-201306 (U.P.)

**DECLARATION**

I hereby declare that this Minor Project Report titled “E-commerce Website” submitted by me to JEMTEC, Greater Noida is a bonafide work undertaken by me and has not been submitted to any other University or Institution for the award of any degree diploma / certificate or published any time before.

(Signature of the Student) Date: / / 2023

Name: Harsh

Enroll. No.: 00725502020

**BONAFIDE CERTIFICATE**

This is to certify that as per best of my belief the project entitled **“E-Commerce Website”** is the bonafide research work carried out by **Harsh** student of BCA, JEMTEC, Greater Noida, in partial fulfilment of the requirements for the Minor Project Report of the Degree of Bachelor of Computer Application.

He / She has worked under my guidance.

I wish him/her a success in all his/her future career endeavours.

Project Guide

Name:

Designation: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Signature with Date

**ACKNOWLEDGEMENT**

I offer my sincere thanks and humble regards to JEMTEC, Greater Noida for imparting us very valuable professional training in BCA.

I pay my gratitude and sincere regards to Dr./Ms…………., my project Guide for giving me the cream of his knowledge. I am thankful to him/her as he/she has been a constant source of advice, motivation and inspiration. I am also thankful to him/her for giving his suggestions and encouragement throughout the project work.

I take the opportunity to express my gratitude and thanks to our computer Lab staff and library staff for providing me opportunity to utilize their resources for the completion of the project.

I am also thankful to my family and friends for constantly motivating me to complete the project and providing me an environment, which enhanced my knowledge.

Date:  / / 2023

Name: Harsh

Enroll. No.: 00725502020

Course: BCA (Vth-Sem)

(Signature of the Student)

|  |  |  |  |
| --- | --- | --- | --- |
| **S.no** | **Topic** | **Page no.** | **Teacher Signature** |
| **1** | **Chapter-1: Introduction** |  |  |
| **2** | **Chapter-2: System Analysis** |  |  |
| **3** | **Chapter-3: System Design** |  |  |
| **4** | **Chapter-4: Testing** |  |  |
| **5** | **Chapter-5: System implementation** |  |  |
| **6** | **Conclusion** |  |  |
| **7** | **References** |  |  |

**CHAPTER-1**

**INTRODUCTION**

Aim of this project is to create a website using HTML, CSS and JAVASCRIPT that bring offline market into online shopping experience. E-commerce websites have become increasingly popular in recent years, with more and more consumers turning to the internet for their shopping needs. E-commerce website such as Amazon, Flipkart, Messo etc. These website offers a wide range of products, including clothing, electronics, home goods, and more. Customers can easily browse and purchase items from the comfort of their own homes, making for a convenient and hassle-free shopping experience but when we talk about Pricing. They are not up to the mark there are a lot of parties involve that increase the price of the product. So keep this in mind we found bazaar.

Bazaar differentiates itself from other e-commerce websites by offering competitive prices and excellent customer service. The website also has a user-friendly interface and a secure checkout process, which helps to build trust with customers and encourage repeat business. In addition, bazaar offers various promotions and discounts throughout the year, which helps to drive traffic to the website and increase sales.

The e-commerce industry as a whole has seen significant growth in recent years. In 2020, global e-commerce sales reached $4.28 trillion, and this number is expected to continue to rise in the coming years. This growth can be attributed to the convenience and accessibility of e-commerce websites, as well as the expansion of internet access and the increasing prevalence of mobile devices.

**CHAPTER-2**

**SYSTEM ANALYSIS**

**2.1. Study of the Problem**

The Current shopping System is critical toset up online shops, customers to browse through the shops, and a system administrator to approve and reject requests for new shops and maintain lists of shop categories. This is a small scale project for Online shopping System. The basic idea is that the candidates can buy product from anywhere during any time by using their card number and password provided to them. The database will maintain the product details information. Customer can view their product details using the card details. This Online shopping system involves only with users.

**Customer role:**

* The customers can login to the System. He/she can view his/her product details and buy their product. The customer can just view the information whereas he/she could not make changes in the database.

**2.2. Objectives**

The purpose of this project is to define the requirements of Online shopping system. This supplementary specification lists the requirements that are not readily captured in the use case model. Supplementary specification and the use case model capture a complete set of requirement of the system.

**2.3. Requirements**

The supplementary specification applies to online shopping system. This specification defines the non-functional requirement of the system such as:

**Functionality:**

* Since it stand alone application, one or more user may use it at a time.

**Hardware Requirements**

* Intel core, windows xp/7/vista/8/10
* 320MB ram or above

**Software Requirements**

* Windows XP-10
* Front end -:Visual studio
* Back end-: SANITY AND STRIPE

**Performance:**

The performance depends on hardware specification **CHAPTER-3**

**SYSTEM DESIGN**

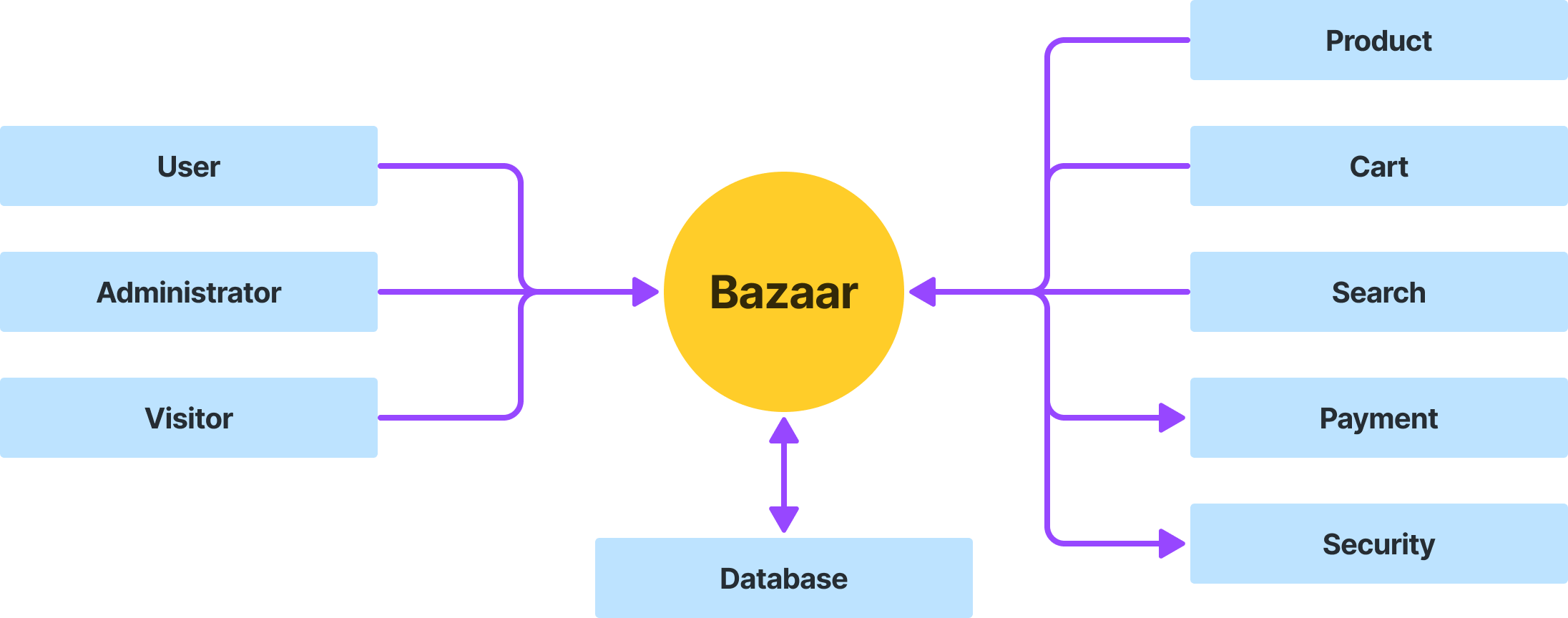
**OVERVIEW:**

System design is a solution, a "HOW TO APPROACH" to the creation of a new system. This important phase is composed of several steps. It provides the understanding and procedural details to implement the system. Design goes through a logical and physical stage of the progress. Logical design reviews the present physical system, prepares input/output specifications, makes audit security and control specifications, detailed implementation plans, and prepares the logical design walkthrough. The physical design makes out the details of the physical system, plans the system implementation and specifies any new hardware and software products.

**DATABASE DESIGN:**

The collection of data is usually referred to as the database. The database contains the information about one particular enterprise. Database system of data involves both the definitions of structures for the storage of information, processing and mechanism for the manipulation of information. In addition, the database system provides for the safety of information stored in the database despite system crashes or attempts of unauthorized access.

**DFD:**



**CHAPTER-4**

**TESTING**

Testing Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

A strategy for software testing integrates software test case design methods into a well- planned series of steps that result in the successful construction of software. Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively applied to both strategic to both large and small-scale systems.

**4.1Objective of Testing**

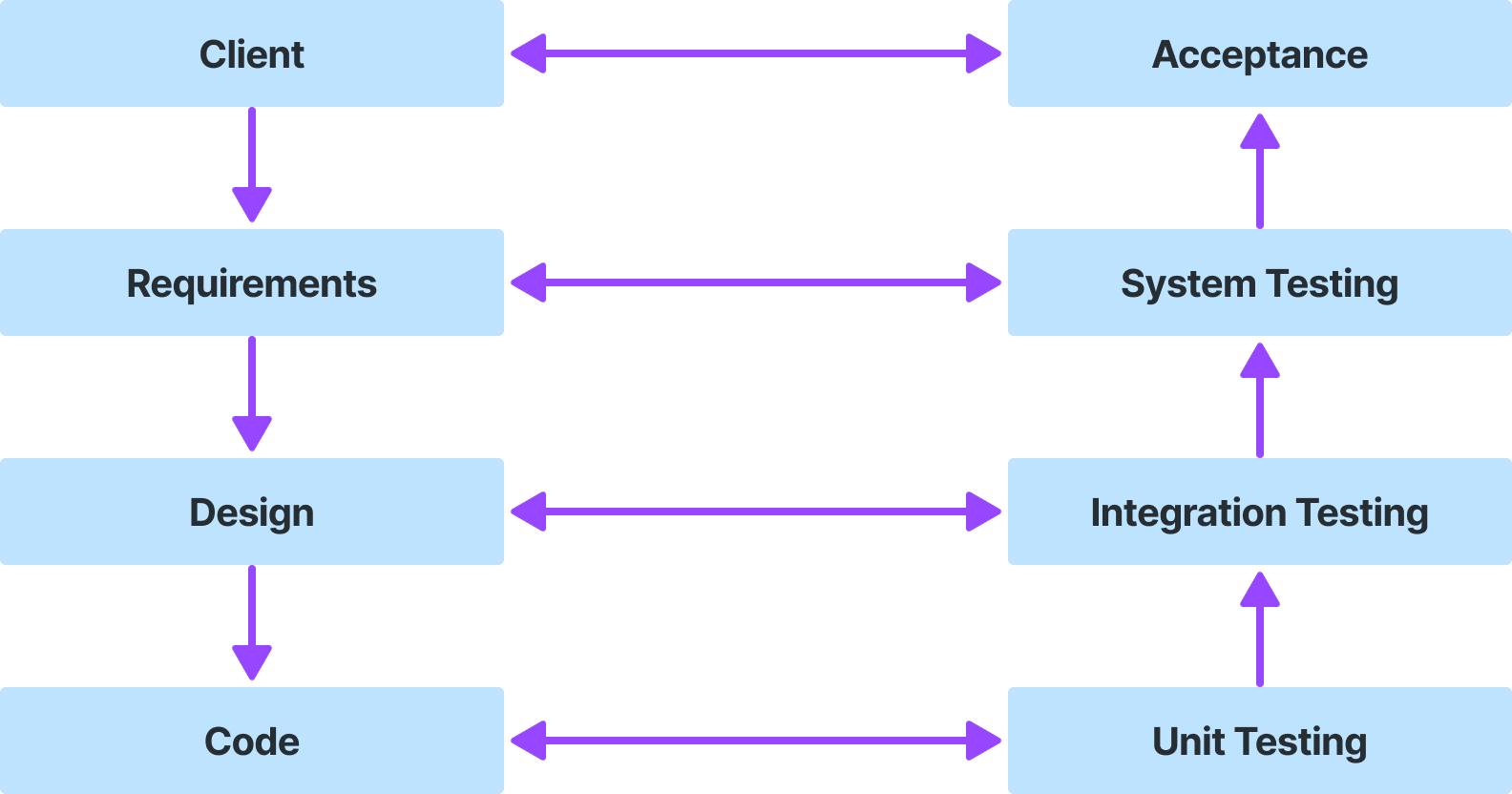
A good testing case is one that has high probability of finding as yet undiscovered errors. Testing is a process of executing a program with the intent of finding errors in it .A successful test is one that uncovers an as yet undiscovered error .The aim of testing is to determine that a program works by showing that it has no errors.

**4.2. Steps in Testing**

* Test planning: The testing process begins with defining what needs to be tested, and how much needs to be tested.
* Test case design: Based on the test plan, test cases or scenarios need to be identified.
* Test data creation: Once detailed test cases are available, actual test data needs to be created or generated, with at least one piece of data for each case.
* Context set up: To perform an actual test, the contextual data scenario required for the test needs to be set up such as populating appropriate tables, setting up communication channels etc. Note that actual test could be any among the kinds of testing listed in the previous section. This completes the preparatory phase of software testing.
* Test execution: The actual test itself needs to be performed by executing the application program (or part thereof) over the test data.
* Test verification: The results of executing the test need to be verified.
* Test logging and follow-up: Activities such as logging the test results, performing any data/metric collection, fixing problems if any and measuring the effectiveness of the test cases/ data need to be performed.
* Test management: Managing all the above activities, tracking and directing the testing process, deciding when to stop testing are activities to be done throughout the above activities.
* Test maintenance: As application requirements, design, implementation change, the test plan/cases/data/metrics etc. will also have to change accordingly. These activities also need to be performed and managed.

**4.3 Levels of Testing**

* Testing is usually relying on to detect the faults on each phase, in addition to the faults introduced during the coding phase itself. Due to this, different levels of testing are used in the testing process, each level of testing aims to test different aspect of a system.



**4.3.1 Unit Testing**

In the bottom of coding technique, each module is tested individually. Firstly the module is tested with some test data that covers all the possible paths and then the actual data was fed to check for results

**4.3.2 System Testing**

Testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing falls within the scope of black box testing, and as such, should require no knowledge of the inner design of the code or logic.

**4.3.3 Integration Testing**

After all the modules are ready and duly tested, these have to be integrated into the application. This integrated application was again tested first with the test data and then with the actual data.

**4.3.4. Acceptance Testing**

Acceptance testing was done after the implementation of the system. The acceptance testing will check if the system works correctly in the user environment and if the entire user specified functionalities are present. It also tests if the system adheres to the company policies and quality standard. The HR Database system was tested and accepted by Sierra Atlantic after the acceptance testing.

**4.4. Testing Strategy For Online Shopping**

Online shopping is a very vast hence it was important that exhaustive testing is done to provide best possible results .It consolidated low level tests to verify small source code segments as well as high level tests that validated the major system functions against the requirements.

**CHAPTER-5**

**IMPLEMENTATION**

**bazaar\schemas\banner.js**

export default {

  name: 'banner',

  title: 'Banner',

  type: 'document',

  fields: [

      {

          name: 'image',

          title: 'Image',

          type: 'image',

          options: {

              hotspot: true,

          },

      },

      {

        name: 'image2',

        title: 'Image2',

        type: 'image',

        options: {

            hotspot: true,

        },

    },

      {

          name: 'buttonText',

          title: 'ButtonText',

          type: 'string',

      },

      {

          name: 'product',

          title: 'Product',

          type: 'string',

      },

      {

          name: 'desc',

          title: 'Desc',

          type: 'string',

      },

      {

          name: 'smallText',

          title: 'SmallText',

          type: 'string',

      },

      {

          name: 'midText',

          title: 'MidText',

          type: 'string',

      },

      {

        name: 'midText2',

        title: 'MidText2',

        type: 'string',

    },

      {

          name: 'largeText1',

          title: 'LargeText1',

          type: 'string',

      },

      {

          name: 'largeText2',

          title: 'LargeText2',

          type: 'string',

      },

      {

          name: 'discount',

          title: 'Discount',

          type: 'string',

      },

      {

          name: 'saleTime',

          title: 'SaleTime',

          type: 'string',

      },

  ],

};

**bazaar\schemas\index.js**

import product from './product';

import banner from './banner';

import women from './women';

import men from './men';

export const schemaTypes = [product, banner, women, men]

**bazaar\schemas\men.js**

export default {

    name: 'men',

    title: 'men',

    type: 'document',

    fields: [

      {

        name: 'image',

        title: 'Image',

        type: 'array',

        of: [{ type: 'image' }],

        options: {

          hotspot: true,

        }

      },

      {

        name: 'name',

        title: 'Name',

        type: 'string',

      },

      {

        name: 'slug',

        title: 'Slug',

        type: 'slug',

        options: {

          source: 'name',

          maxLength: 90,

        }

      },

      {

        name: 'price',

        title: 'Price',

        type: 'number',

      },

      {

        name: 'details',

        title: 'Details',

        type: 'string',

      },

      {

        name: 'Shop',

        title: 'Shop',

        type: 'string',

      }

    ]

  }

**bazaar\schemas\product.js**

export default {

    name: 'product',

    title: 'Product',

    type: 'document',

    fields: [

      {

        name: 'image',

        title: 'Image',

        type: 'array',

        of: [{ type: 'image' }],

        options: {

          hotspot: true,

        }

      },

      {

        name: 'name',

        title: 'Name',

        type: 'string',

      },

      {

        name: 'slug',

        title: 'Slug',

        type: 'slug',

        options: {

          source: 'name',

          maxLength: 90,

        }

      },

      {

        name: 'price',

        title: 'Price',

        type: 'number',

      },

      {

        name: 'details',

        title: 'Details',

        type: 'string',

      },

      {

        name: 'Shop',

        title: 'Shop',

        type: 'string',

      }

    ]

  }

**bazaar\schemas\women.js**

export default {

    name: 'women',

    title: 'women',

    type: 'document',

    fields: [

      {

        name: 'image',

        title: 'Image',

        type: 'array',

        of: [{ type: 'image' }],

        options: {

          hotspot: true,

        }

      },

      {

        name: 'name',

        title: 'Name',

        type: 'string',

      },

      {

        name: 'slug',

        title: 'Slug',

        type: 'slug',

        options: {

          source: 'name',

          maxLength: 90,

        }

      },

      {

        name: 'price',

        title: 'Price',

        type: 'number',

      },

      {

        name: 'details',

        title: 'Details',

        type: 'string',

      },

      {

        name: 'Shop',

        title: 'Shop',

        type: 'string',

      }

    ]

  }

**components\About.jsx**

import React from 'react'

const About = () => {

  return (

    <div>

      <div className='about' >

      <h1>HI! MY NAME IS HARSH </h1>

      <h1>AND THIS IS MY MINIOR PROJECT. </h1>

      </div>

    </div>

  )

}

export default About

**components\Cart.jsx**

import React, { useRef } from 'react';

import Link from 'next/link';

import { AiOutlineMinus, AiOutlinePlus, AiOutlineLeft, AiOutlineShopping } from 'react-icons/ai';

import { TiDeleteOutline } from 'react-icons/ti';

import toast from 'react-hot-toast';

import { useStateContext } from '../context/StateContext';

import { urlFor } from '../lib/client';

import getStripe from '../lib/getStripe';

const Cart = () => {

  const cartRef = useRef();

  const { totalPrice, totalQuantities, cartItems, setShowCart, toggleCartItemQuanitity, onRemove } = useStateContext();

  const handleCheckout = async () => {

    const stripe = await getStripe();

    const response = await fetch('/api/stripe', {

      method: 'POST',

      headers: {

        'Content-Type': 'application/json',

      },

      body: JSON.stringify(cartItems),

    });

    if(response.statusCode === 500) return;

    const data = await response.json();

    toast.loading('Redirecting...');

    stripe.redirectToCheckout({ sessionId: data.id });

  }

  return (

    <div className="cart-wrapper" ref={cartRef}>

      <div className="cart-container">

        <button

        type="button"

        className="cart-heading"

        onClick={() => setShowCart(false)}>

          <AiOutlineLeft />

          <span className="heading">Your Cart</span>

          <span className="cart-num-items">({totalQuantities} items)</span>

        </button>

        {cartItems.length < 1 && (

          <div className="empty-cart">

            <AiOutlineShopping size={150} />

            <h3>Your shopping bag is empty</h3>

            <Link href="/">

              <button

                type="button"

                onClick={() => setShowCart(false)}

                className="btn"

              >

                Continue Shopping

              </button>

            </Link>

          </div>

        )}

        <div className="product-container">

          {cartItems.length >= 1 && cartItems.map((*item*) => (

            <div className="product" key={item.\_id}>

              <img src={urlFor(item?.image[0])} className="cart-product-image" />

              <div className="item-desc">

                <div className="flex top">

                  <h5>{item.name}</h5>

                  <h4>₹{item.price}</h4>

                </div>

                <div className="flex bottom">

                  <div>

                  <p className="quantity-desc">

                    <span className="minus" onClick={() => toggleCartItemQuanitity(item.\_id, 'dec') }>

                    <AiOutlineMinus />

                    </span>

                    <span className="num" onClick="">{item.quantity}</span>

                    <span className="plus" onClick={() => toggleCartItemQuanitity(item.\_id, 'inc') }><AiOutlinePlus /></span>

                  </p>

                  </div>

                  <button

                    type="button"

                    className="remove-item"

                    onClick={() => onRemove(item)}

                  >

                    <TiDeleteOutline />

                  </button>

                </div>

              </div>

            </div>

          ))}

        </div>

        {cartItems.length >= 1 && (

          <div className="cart-bottom">

            <div className="total">

              <h3>Subtotal:</h3>

              <h3>₹{totalPrice}</h3>

            </div>

            <div className="btn-container">

              <button type="button" className="btn" onClick={handleCheckout}>

                Pay with Stripe

              </button>

            </div>

          </div>

        )}

      </div>

    </div>

  )

}

export default Cart

**components\Contact.jsx**

import React from 'react'

const Contact = () => {

  return (

    <div>

          <div class="background">

        <div class="container">

          <div class="screen">

            <div class="screen-header">

              <div class="screen-header-left">

                <div class="screen-header-button close"></div>

                <div class="screen-header-button maximize"></div>

                <div class="screen-header-button minimize"></div>

              </div>

              <div class="screen-header-right">

                <div class="screen-header-ellipsis"></div>

                <div class="screen-header-ellipsis"></div>

                <div class="screen-header-ellipsis"></div>

              </div>

            </div>

            <div class="screen-body">

              <div class="screen-body-item left">

                <div class="app-title">

                  <span>CONTACT</span>

                  <span>US</span>

                </div>

                <div class="app-contact">CONTACT INFO : +91 999 5555 999</div>

              </div>

              <div class="screen-body-item">

                <div class="app-form">

                <form name="submit-to-google-sheet">

                  <div class="app-form-group">

                    <input type="text" class="app-form-control" name="name" placeholder="Name" required></input>

                    <input type="email" class="app-form-control" name="Email" placeholder="Enter Your Email" required></input>

                    <input type="number" class="app-form-control" name="number" placeholder="Contact No" required></input>

                    <input type="text" class="app-form-control" name="message" placeholder="Message" required></input>

                    <div class="button-container">

                        <button class="app-form-button">Submit</button></div>

                  </div>

                </form>

                </div>

              </div>

            </div>

          </div>

        </div>

      </div>

    </div>

  )

}

export default Contact

**components\Footer.jsx**

import React from 'react';

import { AiFillFacebook, AiFillInstagram, AiOutlineTwitter} from 'react-icons/ai';

const Footer = () => {

  return (

    <div className="footer-container">

      <section class="footer">

        <div class="footer-box">

            <h3>BAZAAR</h3>

            <p>Our online store is the perfect place to shop for unique, one-of-a-kind clothing items that you won't find anywhere else.

            </p>

        </div>

        <div class="footer-box">

            <h3>Support</h3>

            <li><a href="#">Products</a></li>

            <li><a href="#">Help</a></li>

            <li><a href="#">Return Policy</a></li>

            <li><a href="#">Terms of Use</a></li>

        </div>

        <div class="footer-box">

            <h3>View Guides</h3>

            <li><a href="#">Payments</a></li>

            <li><a href="#">Blogs</a></li>

            <li><a href="#">Our Partners</a></li>

        </div>

        <div class="footer-box contact">

            <h3>Contact</h3>

            <li><a href="#">250 New Delhi, India</a></li>

            <li><a href="#">+91 999 5555 999</a></li>

            <li><a href="#">Bazaar@shop.in</a></li>

        </div>

    </section>

      <p>2023 Bazaar All rights reserverd</p>

      <p className="icons">

        <AiFillInstagram />

        <AiOutlineTwitter />

        <AiFillFacebook />

      </p>

    </div>

  )

}

export default Footer

**components\FooterBanner.jsx**

import React from 'react';

import Link from 'next/link';

import { urlFor } from '../lib/client';

const FooterBanner = ({ footerBanner: { *discount*, *largeText1*, *largeText2*, *saleTime*, *smallText*, *midText*, *midText2*, *desc*, *product*, *buttonText*, *image2* } }) => {

  return (

    <div className="footer-banner-container">

      <div className="banner-desc">

        <div className="right">

          <p>{smallText}</p>

          <h3>{midText}</h3>

          <h3>{midText2}</h3>

          <p>{desc}</p>

          <Link href="/">

            <button type="button">{buttonText}</button>

          </Link>

        </div>

        <div className="left">

          <p>{discount}</p>

          <h3>{largeText1}</h3>

          <h3>{largeText2}</h3>

          <p>{saleTime}</p>

        </div>

        <img

          src={urlFor(image2)} className="footer-banner-image"

        />

      </div>

    </div>

  )

}

export default FooterBanner

**components\HeroBanner.jsx**

import React from 'react';

import Link from 'next/link';

import { urlFor } from '../lib/client';

const HeroBanner = ({ *heroBanner* }) => {

  return (

    <div className="hero-banner-container">

      <div >

        <p className="beats-solo">{heroBanner.smallText}</p>

        <h3 className= "heroBanner-midText">{heroBanner.midText}{heroBanner.midText2}</h3>

        <h1>{heroBanner.largeText1}</h1>

        <img src={urlFor(heroBanner.image)} alt="" className="hero-banner-image" />

        <div className="button1">

          <Link href="/">

            <button type="button" >{heroBanner.buttonText}</button>

          </Link>

          <div className="desc">

            <p>{heroBanner.desc}</p>

          </div>

        </div>

      </div>

    </div>

  )

}

export default HeroBanner

**components\index.js**

export {default as Footer} from './Footer';

export {default as Layout} from './Layout';

export {default as Navbar} from './Navbar';

export {default as Product} from './Product';

export {default as HeroBanner} from './HeroBanner';

export {default as FooterBanner} from './FooterBanner';

export {default as Cart} from './Cart';

export {default as Women} from './Women';

export {default as Review} from './Review';

export {default as Men} from './Men';

export {default as Contact} from './Contact';

export {default as About} from './About';

**components\Layout.jsx**

import React from 'react';

import Head from 'next/head';

import Navbar from './Navbar';

import Footer from './Footer';

const Layout = ({ *children* }) => {

  return (

    <div className="layout">

      <Head>

        <title>Bazaar.in</title>

      </Head>

      <header>

        <Navbar />

      </header>

      <main className="main-container">

        {children}

      </main>

      <footer>

        <Footer />

      </footer>

    </div>

  )

}

export default Layout

**components\Men.jsx**

import React from 'react';

import Link from 'next/link';

import { urlFor } from '../lib/client';

const Men = ({ men: { *image*, *name*, *slug*, *price*, *Shop* } }) => {

  return (

    <div>

      <Link href={`/men/${slug.current}`}>

        <div className="product-card">

          <img

            src={urlFor(image && image[0])}

            width={250}

            height={250}

            className="product-image"

          />

          <p className="product-name">{name}</p>

          <div className="product-desc">

          <p className="product-price">₹{price}</p>

          <p className="product-price">{Shop}</p>

          </div>

        </div>

      </Link>

    </div>

  )

}

export default Men

**components\Navbar.jsx**

import React from 'react';

import Link from 'next/link';

import { AiOutlineSearch, AiOutlineShopping } from 'react-icons/ai'

import { Cart } from './';

import { useStateContext} from '../context/StateContext';

const Navbar = () => {

  const { showCart, setShowCart, totalQuantities } = useStateContext();

  return (

    <div className="navbar-container">

      <p className="logo">

      <div className='icon-logo'></div><Link href="/">Bazaar</Link>

      </p>

                <ul id="navbar">

                    <li><Link href="/"><a>Home</a></Link></li>

                    <li><Link href="/women"><a>Women</a></Link></li>

                    <li><Link href="/men"><a>Men</a></Link></li>

                    <li><Link href="/about"><a>About Me</a></Link></li>

                    <li><Link href="/contact"><a>Contact Us</a></Link></li>

                </ul>

      <div>

      <button type="button" className="search-icon">

        <AiOutlineSearch />

      </button>

      <button type="button" className="cart-icon" onClick={() => setShowCart(true)}>

        <AiOutlineShopping />

        <span className="cart-item-qty">{totalQuantities}</span>

      </button>

      </div>

      {showCart && <Cart />}

    </div>

  )

}

export default Navbar

**components\Product.jsx**

import React from 'react';

import Link from 'next/link';

import { urlFor } from '../lib/client';

const Product = ({ product: { *image*, *name*, *slug*, *price*, *Shop* } }) => {

  return (

    <div>

      <Link href={`/product/${slug.current}`}>

        <div className="product-card">

          <img

            src={urlFor(image && image[0])}

            width={250}

            height={250}

            className="product-image"

          />

          <p className="product-name">{name}</p>

          <div className="product-desc">

          <p className="product-price">₹{price}</p>

          <p className="product-price">{Shop}</p>

          </div>

        </div>

      </Link>

    </div>

  )

}

export default Product

**components\Review.jsx**

import React from 'react';

import { AiOutlineMinus, AiOutlinePlus, AiFillStar, AiOutlineStar } from 'react-icons/ai';

const Review = ({ *review* }) => {

    return (

      <div>

            <selection className="reviews" id="reviews">

        <h2 className="review-heading">Our Customer's</h2>

        <div className="review-container">

            <div className="box">

            <h2>Priya Singh</h2>

                <div className="stars">

                <AiFillStar />

              <AiFillStar />

              <AiFillStar />

              <AiFillStar />

              <AiOutlineStar />

                </div>

                <p>"I recently purchased a product from <b>Bazaar</b> and I am extremely satisfied with my experience. The website was easy to navigate and I was able to find exactly what I was looking for."</p>

            </div>

            <div className="box">

            <h2>Anuj Kumar</h2>

                <div className="stars">

                <AiFillStar />

              <AiFillStar />

              <AiFillStar />

              <AiFillStar />

              <AiOutlineStar />

                </div>

                <p>"My experience with <b>Bazaar</b> was excellent and I would highly recommend it to others. The product I received was of high quality and I am very happy with my purchase. Thank you for the great service!"</p>

            </div>

            <div className="box">

            <h2>Varun Sharma</h2>

                <div className="stars">

                <AiFillStar />

              <AiFillStar />

              <AiFillStar />

              <AiFillStar />

              <AiOutlineStar />

                </div>

                <p>"I am extremely satisfied with my experience. The product I ordered was exactly as described and arrived promptly. Overall, I had a great shopping experience and would highly recommend <b>Bazaar</b> to others."</p>

            </div>

        </div>

    </selection>

    <section class="newsletter">

        <h2>Subscribe For Special Discounts</h2>

        <form name="submit-to-google-sheet">

            <input type="email" name="Email" placeholder="Enter Your Email..." required></input>

            <button type="submit" class="email-btn">Subscribe</button>

        </form>

    </section>

      </div>

    )

  }

export default Review

**components\Women.jsx**

import React from 'react';

import Link from 'next/link';

import { urlFor } from '../lib/client';

const Women = ({ women: { *image*, *name*, *slug*, *price*, *Shop* } }) => {

  return (

    <div>

      <Link href={`/women/${slug.current}`}>

        <div className="product-card">

          <img

            src={urlFor(image && image[0])}

            width={250}

            height={250}

            className="product-image"

          />

          <p className="product-name">{name}</p>

          <div className="product-desc">

          <p className="product-price">₹{price}</p>

          <p className="product-price">{Shop}</p>

          </div>

        </div>

      </Link>

    </div>

  )

}

export default Women

**context\StateContext.js**

import React, { createContext, useContext, useState, useEffect } from 'react';

import { toast } from 'react-hot-toast';

const Context = createContext();

export const StateContext = ({ *children* }) => {

  const [showCart, setShowCart] = useState(false);

  const [cartItems, setCartItems] = useState([]);

  const [totalPrice, setTotalPrice] = useState(0);

  const [totalQuantities, setTotalQuantities] = useState(0);

  const [qty, setQty] = useState(1);

  let foundProduct;

  let index;

  const onAdd = (*product*, *quantity*) => {

    const checkProductInCart = cartItems.find((*item*) => item.\_id === product.\_id);

    setTotalPrice((*prevTotalPrice*) => prevTotalPrice + product.price \* quantity);

    setTotalQuantities((*prevTotalQuantities*) => prevTotalQuantities + quantity);

    if(checkProductInCart) {

      const updatedCartItems = cartItems.map((*cartProduct*) => {

        if(cartProduct.\_id === product.\_id) return {

          ...cartProduct,

          quantity: cartProduct.quantity + quantity

        }

      })

      setCartItems(updatedCartItems);

    } else {

      product.quantity = quantity;

      setCartItems([...cartItems, { ...product }]);

    }

    toast.success(`${qty} ${product.name} added to the cart.`);

  }

  const onRemove = (*product*) => {

    foundProduct = cartItems.find((*item*) => item.\_id === product.\_id);

    const newCartItems = cartItems.filter((*item*) => item.\_id !== product.\_id);

    setTotalPrice((*prevTotalPrice*) => prevTotalPrice -foundProduct.price \* foundProduct.quantity);

    setTotalQuantities(*prevTotalQuantities* => prevTotalQuantities - foundProduct.quantity);

    setCartItems(newCartItems);

  }

  const toggleCartItemQuanitity = (*id*, *value*) => {

    foundProduct = cartItems.find((*item*) => item.\_id === id)

    index = cartItems.findIndex((*product*) => product.\_id === id);

    const newCartItems = cartItems.filter((*item*) => item.\_id !== id)

    if(value === 'inc') {

      setCartItems([...newCartItems, { ...foundProduct, quantity: foundProduct.quantity + 1 } ]);

      setTotalPrice((*prevTotalPrice*) => prevTotalPrice + foundProduct.price)

      setTotalQuantities(*prevTotalQuantities* => prevTotalQuantities + 1)

    } else if(value === 'dec') {

      if (foundProduct.quantity > 1) {

        setCartItems([...newCartItems, { ...foundProduct, quantity: foundProduct.quantity - 1 } ]);

        setTotalPrice((*prevTotalPrice*) => prevTotalPrice - foundProduct.price)

        setTotalQuantities(*prevTotalQuantities* => prevTotalQuantities - 1)

      }

    }

  }

  const incQty = () => {

    setQty((*prevQty*) => prevQty + 1);

  }

  const decQty = () => {

    setQty((*prevQty*) => {

      if(prevQty - 1 < 1) return 1;

      return prevQty - 1;

    });

  }

  return (

    <Context.Provider

      value={{

        showCart,

        setShowCart,

        cartItems,

        totalPrice,

        totalQuantities,

        qty,

        incQty,

        decQty,

        onAdd,

        toggleCartItemQuanitity,

        onRemove,

        setCartItems,

        setTotalPrice,

        setTotalQuantities

      }}

    >

      {children}

    </Context.Provider>

  )

}

export const useStateContext = () => useContext(Context);

**lib\client.js**

import sanityClient from '@sanity/client';

import imageUrlBuilder from '@sanity/image-url';

export const client = sanityClient({

    projectId: 'gf4co1qf',

    dataset:'production',

    apiVersion: '2023-01-06',

    useCdn: true,

    token: process.env.NEXT\_PUBLIC\_SANITY\_TOKEN

});

const builder = imageUrlBuilder(client);

export const urlFor = (*source*) => builder.image(source);

**lib\getStripe.js**

import { loadStripe } from '@stripe/stripe-js';

let stripePromise;

const getStripe = () => {

  if(!stripePromise) {

    stripePromise = loadStripe(process.env.NEXT\_PUBLIC\_STRIPE\_PUBLISHABLE\_KEY);

  }

  return stripePromise;

}

export default getStripe;

**lib\utils.js**

import confetti from 'canvas-confetti';

export const runFireworks = () => {

  var duration = 5 \* 1000;

  var animationEnd = Date.now() + duration;

  var defaults = { startVelocity: 30, spread: 360, ticks: 60, zIndex: 0 };

  function randomInRange(*min*, *max*) {

    return Math.random() \* (max - min) + min;

  }

  var interval = setInterval(function() {

    var timeLeft = animationEnd - Date.now();

    if (timeLeft <= 0) {

      return clearInterval(interval);

    }

    var particleCount = 50 \* (timeLeft / duration);

    // since particles fall down, start a bit higher than random

    confetti(Object.assign({}, defaults, { particleCount, origin: { x: randomInRange(0.1, 0.3), y: Math.random() - 0.2 } }));

    confetti(Object.assign({}, defaults, { particleCount, origin: { x: randomInRange(0.7, 0.9), y: Math.random() - 0.2 } }));

  }, 250);

}

**pages\about**

import React from 'react'

import { About } from '../../components';

const index = () => {

  return (

    <div>

      <About />

    </div>

  )

}

export default index

**pages\api**

import Stripe from 'stripe';

const stripe = new Stripe(process.env.NEXT\_PUBLIC\_STRIPE\_SECRET\_KEY);

export default async function handler(*req*, *res*) {

  if (req.method === 'POST') {

    try {

      const params = {

        submit\_type: 'pay',

        mode: 'payment',

        payment\_method\_types: ['card'],

        billing\_address\_collection: 'auto',

        shipping\_options: [

          { shipping\_rate: 'shr\_1MNBUeSAYbgfZjlXCU25qT5r' },

        ],

        line\_items: req.body.map((*item*) => {

          const img = item.image[0].asset.\_ref;

          const newImage = img.replace('image-', 'https://cdn.sanity.io/images/gf4co1qf/production/').replace('-webp', '.webp');

          return {

            price\_data: {

              currency: 'INR',

              product\_data: {

                name: item.name,

                images: [newImage],

              },

              unit\_amount: item.price \* 100,

            },

            adjustable\_quantity: {

              enabled:true,

              minimum: 1,

            },

            quantity: item.quantity

          }

        }),

        success\_url: `${req.headers.origin}/success`,

        cancel\_url: `${req.headers.origin}/canceled`,

      }

      // Create Checkout Sessions from body params.

      const session = await stripe.checkout.sessions.create(params);

      res.status(200).json(session);

    } catch (err) {

      res.status(err.statusCode || 500).json(err.message);

    }

  } else {

    res.setHeader('Allow', 'POST');

    res.status(405).end('Method Not Allowed');

  }

}

**pages\contact**

import React from 'react'

import { Contact } from '../../components';

const index = () => {

  return (

    <div>

      <Contact />

    </div>

  )

}

export default index

**pages\men**

import React from 'react';

import { client } from '../../lib/client';

import { Men } from '../../components';

const Home = ({ *mens* }) => (

    <div>

      <div className="products-heading">

        <h2>Men's Collection</h2>

        <p>Express Yourself By Your Clothes!!</p>

      </div>

      <div className="products-container">

      {mens?.map((*men*) => <Men key={Men.\_id} men={men} />)}

      </div>

    </div>

  );

  export const getServerSideProps = async () => {

    const query = '\*[\_type == "men"]';

    const mens = await client.fetch(query);

    return {

      props: { mens }

    }

  }

  export default Home;

**pages\product**

import React, { useState } from 'react';

import { AiOutlineMinus, AiOutlinePlus, AiFillStar, AiOutlineStar } from 'react-icons/ai';

import { client, urlFor } from '../../lib/client';

import { Product } from '../../components';

import { useStateContext } from '../../context/StateContext';

const ProductDetails = ({ *product*, *products* }) => {

  const { image, name, details, price } = product;

  const [index, setIndex] = useState(0);

  const { decQty, incQty, qty, onAdd, setShowCart } = useStateContext();

  const handleBuyNow = () => {

    onAdd(product, qty);

    setShowCart(true);

  }

  return (

    <div>

      <div className="product-detail-container">

        <div>

          <div className="image-container">

            <img src={urlFor(image && image[index])} className="product-detail-image" />

          </div>

          <div className="small-images-container">

            {image?.map((*item*, *i*) => (

              <img

                key={i}

                src={urlFor(item)}

                className={i === index ? 'small-image selected-image' : 'small-image'}

                onMouseEnter={() => setIndex(i)}

              />

            ))}

          </div>

        </div>

        <div className="product-detail-desc">

          <h1>{name}</h1>

          <div className="stars">

            <div>

              <AiFillStar />

              <AiFillStar />

              <AiFillStar />

              <AiFillStar />

              <AiOutlineStar />

            </div>

            <p>

              (20)

            </p>

          </div>

          <h3>Details: </h3>

          <p>{details}</p>

          <p className="price">₹{price}</p>

          <div className="quantity">

            <h3>Quantity:</h3>

            <p className="quantity-desc">

              <span className="minus" onClick={decQty}><AiOutlineMinus /></span>

              <span className="num">{qty}</span>

              <span className="plus" onClick={incQty}><AiOutlinePlus /></span>

            </p>

          </div>

          <div className="buttons">

            <button type="button" className="add-to-cart" onClick={() => onAdd(product, qty)}>Add to Cart</button>

            <button type="button" className="buy-now" onClick={handleBuyNow}>Buy Now</button>

          </div>

        </div>

      </div>

      <div className="maylike-products-wrapper">

          <h2>You may also like</h2>

          <div className="marquee">

            <div className="maylike-products-container track">

              {products.map((*item*) => (

                <Product key={item.\_id} product={item} />

              ))}

            </div>

          </div>

      </div>

    </div>

  )

}

export const getStaticPaths = async () => {

  const query = `\*[\_type == "product"] {

    slug {

      current

    }

  }

  `;

  const products = await client.fetch(query);

  const paths = products.map((*product*) => ({

    params: {

      slug: product.slug.current

    }

  }));

  return {

    paths,

    fallback: 'blocking'

  }

}

export const getStaticProps = async ({ params: { *slug* }}) => {

  const query = `\*[\_type == "product" && slug.current == '${slug}'][0]`;

  const productsQuery = '\*[\_type == "product"]'

  const product = await client.fetch(query);

  const products = await client.fetch(productsQuery);

  console.log(product);

  return {

    props: { products, product }

  }

}

export default ProductDetails

**pages\women**

import React from 'react';

import { client } from '../../lib/client';

import { Women } from '../../components';

const Home = ({ *womens* }) => (

    <div>

      <div className="products-heading">

        <h2>Women Collection</h2>

        <p>Express Yourself By Your Clothes!!</p>

      </div>

      <div className="products-container">

        {womens?.map((*women*) => <Women key={Women.\_id} women={women} />)}

      </div>

    </div>

  );

  export const getServerSideProps = async () => {

    const query = '\*[\_type == "women"]';

    const womens = await client.fetch(query);

    return {

      props: { womens }

    }

  }

  export default Home;

**pages\\_app.js**

import React from 'react';

import { Toaster } from 'react-hot-toast';

import { Layout } from '../components';

import '../styles/globals.css';

import '../styles/contact.css';

import { StateContext } from '../context/StateContext';

function MyApp({ *Component*, *pageProps* }) {

  return (

    <StateContext>

      <Layout>

        <Toaster />

        <Component {...pageProps} />

      </Layout>

    </StateContext>

  )

}

export default MyApp

**pages\\_document.js**

import { Html, Head, Main, NextScript } from 'next/document'

export default function Document() {

  return (

    <Html lang="en">

      <Head />

      <body>

        <Main />

        <NextScript />

      </body>

    </Html>

  )

}

**pages\index.js**

import React from 'react';

import { client } from '../lib/client';

import { Product, FooterBanner, HeroBanner, Review } from '../components';

const Home = ({ *products*, *bannerData*, *review* }) => (

  <div>

    <HeroBanner heroBanner={bannerData.length && bannerData[0]}  />

    <div className="products-heading">

      <h2>Best Selling Products</h2>

      <p>These are the Best Selling Products of 2022</p>

    </div>

    <div className="products-container">

      {products?.map((*product*) => <Product key={product.\_id} product={product} />)}

    </div>

    <FooterBanner footerBanner={bannerData && bannerData[0]} />

    <Review review={review} />

  </div>

);

export const getServerSideProps = async () => {

  const query = '\*[\_type == "product"]';

  const products = await client.fetch(query);

  const bannerQuery = '\*[\_type == "banner"]';

  const bannerData = await client.fetch(bannerQuery);

  return {

    props: { products, bannerData }

  }

}

export default Home;

**pages\success.js**

import React, { useState, useEffect } from 'react';

import Link from 'next/link';

import { BsBagCheckFill } from 'react-icons/bs';

import { useStateContext } from '../context/StateContext';

import { runFireworks } from '../lib/utils';

const Success = () => {

  const { setCartItems, setTotalPrice, setTotalQuantities } = useStateContext();

  useEffect(() => {

    localStorage.clear();

    setCartItems([]);

    setTotalPrice(0);

    setTotalQuantities(0);

    runFireworks();

  }, []);

  return (

    <div className="success-wrapper">

      <div className="success">

        <p className="icon">

          <BsBagCheckFill />

        </p>

        <h2>Thank you for your order!</h2>

        <p className="email-msg">Check your email inbox for the receipt.</p>

        <p className="description">

          If you have any questions, please email

          <a className="email" href="mailto:Harshchhindre9718@gmail.com">

            order@example.com

          </a>

        </p>

        <Link href="/">

          <button type="button" width="300px" className="btn">

            Continue Shopping

          </button>

        </Link>

      </div>

    </div>

  )

}

export default Success

**styles\contact.css**

@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600;700&display=swap');

\*{

    font-family: 'Poppins', sans-serif;

    margin: 0;

    padding: 0;

    scroll-behavior: smooth;

    scroll-padding-top: 2rem;

    box-sizing: border-box;

    list-style: none;

    text-decoration: none;

}\*,

\*:before, \*:after {

    box-sizing: border-box;

    -webkit-font-smoothing: antialiased;

    -moz-osx-font-smoothing: grayscale;

  }

  body {

    background: linear-gradient(to right, #fff 0%, #fff 100%);

    font-size: 12px;

  }

  .background {

    display: flex;

    min-height: 100vh;

    font-weight: 600;

  }

  .container {

    flex: 0 1 700px;

    margin: auto;

    padding: 10px;

  }

  .screen {

    position: relative;

    background: #3e3e3e;

    border-radius: 15px;

  }

  .screen:after {

    content: '';

    display: block;

    position: absolute;

    top: 0;

    left: 20px;

    right: 20px;

    bottom: 0;

    border-radius: 15px;

    box-shadow: 0 20px 40px rgba(0, 0, 0, .4);

    z-index: -1;

  }

  .screen-header {

    display: flex;

    align-items: center;

    padding: 10px 20px;

    background: #4d4d4f;

    border-top-left-radius: 15px;

    border-top-right-radius: 15px;

  }

  .screen-header-left {

    margin-right: auto;

  }

  .screen-header-button {

    display: inline-block;

    width: 8px;

    height: 8px;

    margin-right: 3px;

    border-radius: 8px;

    background: white;

  }

  .screen-header-button.close {

    background: #ed1c6f;

  }

  .screen-header-button.maximize {

    background: #e8e925;

  }

  .screen-header-button.minimize {

    background: #74c54f;

  }

  .screen-header-right {

    display: flex;

  }

  .screen-header-ellipsis {

    width: 3px;

    height: 3px;

    margin-left: 2px;

    border-radius: 8px;

    background: rgb(196, 196, 196);

  }

  .screen-body {

    display: flex;

  }

  .screen-body-item {

    flex: 1;

    padding: 50px;

  }

  .screen-body-item.left {

    display: flex;

    flex-direction: column;

  }

  .app-title {

    display: flex;

    flex-direction: column;

    position: relative;

    color: #fda702;

    font-size: 26px;

  }

  .app-title:after {

    content: '';

    display: block;

    position: absolute;

    left: 0;

    bottom: -10px;

    width: 25px;

    height: 4px;

    background: #fda702;

  }

  .app-contact {

    margin-top: auto;

    font-size: 0.765rem;

    color: #888;

  }

  .app-form-group {

    margin-bottom: 15px;

  }

  .app-form-group.message {

    margin-top: 40px;

  }

  .app-form-group.buttons {

    margin-bottom: 0;

    text-align: right;

  }

  .app-form-control {

    width: 100%;

    padding: 10px 0;

    background: none;

    border: none;

    border-bottom: 1px solid #666;

    color: #ddd;

    font-size: 1rem;

    font-weight: 500;

    outline: none;

    transition: border-color .5s;

  }

  .app-form-control::placeholder {

    color: #666;

  }

  .app-form-control:focus {

    border-bottom-color: #ddd;

  }

  .button-container{

    background: rgb(82, 82, 82);

    margin: 1.2rem 0rem ;

    padding: 10px 20px;

    text-align: center;

    border-radius: 0.5rem;

  }

  .app-form-button {

    background: none;

    border: none;

    color: #fff;

    font-size: 1rem;

    cursor: pointer;

    outline: none;

    font-weight: 600;

    transition: 500ms;

  }

  .app-form-button:hover {

    color: #fda702;

  }

  @media screen and (max-width: 520px) {

    .screen-body {

      flex-direction: column;

    }

    .screen-body-item.left {

      margin-bottom: 30px;

    }

    .app-title {

      flex-direction: row;

    }

    .app-title span {

      margin-right: 12px;

    }

    .app-title:after {

      display: none;

    }

  }

  @media screen and (max-width: 600px) {

    .screen-body {

      padding: 40px;

    }

    .screen-body-item {

      padding: 0;

    }

  }

**styles\globals.css**

@import url('https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600;700&display=swap');

html,

body,

\* {

  padding: 0;

  margin: 0;

  font-family: 'Poppins', sans-serif;

  box-sizing: border-box;

  margin: 0;

  padding: 0;

  scroll-behavior: smooth;

  scroll-padding-top: 2rem;

  box-sizing: border-box;

  list-style: none;

  text-decoration: none;

}

:root{

  --main-color:#fda702;

  --second-color:#363636;

  --text-color: #02171d;

  --third-color:rgb(247, 247, 247);

  --bg-color: #fff;

  --box-shadow: 2px 2px 10px 4px rgb(14 55 54 / 15%);

}

::-webkit-scrollbar {

  width: 0px;

}

a {

  color: inherit;

  text-decoration: none;

  padding: 0 10px;

}

.main-container{

  padding: 5rem 10rem 0rem;

margin: auto;

width: 100%;

}

.navbar-container{

  width: 100%;

  display: flex;

  justify-content: space-between;

  padding: 1rem 10rem;

  position: fixed;

  top: 0;

  left: 0;

  background: white;

  box-shadow: 0 1px 4px hsl(0 4% 15% / 10%);

  z-index: 100;

}

#navbar{

  display: flex;

  align-items: center;

  font-size: 1rem;

  font-weight: 500;

}

#navbar a{

  transition: 0.5s;

}

#navbar a:hover{

  font-size: 1.1rem;

  color: var(--main-color);

}

.marquee-text{

  font-size: 29px;

  font-weight: 600;

  margin: 60px 0px;

  color: #f02d34;

}

.marquee {

  position: relative;

  height: 400px;

  width: 100%;

  overflow-x: hidden;

}

.track {

  position: absolute;

  white-space: nowrap;

  will-change: transform;

  animation: marquee 15s linear infinite;

  width: 180%;

}

.track:hover {

  animation-play-state: paused;

}

@keyframes *marquee* {

  from { transform: translateX(0); }

  to { transform: translateX(-50%); }

}

 span.text-red {

    -webkit-text-stroke: 1px #f02d34;

    margin-left: 6px;

}

.logo{

  color: var(--main-color);

    font-size: 18px;

    display: flex;

    /\* justify-content: center; \*/

    align-items: center;

}

.logo a{

  font-size: 1.4rem;

  font-weight: 800;

}

.icon-logo{

  font-size: 1.5rem;

}

.search-icon{

  font-size: 25px;

  padding: 0px 10px;

  color: black;

  cursor: pointer;

  position: relative;

  transition: transform .4s ease;

  border: none;

  background-color: transparent;

}

.search-icon:hover{

  color: var(--main-color);

  transform: scale(1.1,1.1);

}

.cart-icon{

  font-size: 25px;

  color: black;

  cursor: pointer;

  position: relative;

  transition: transform .4s ease;

  border: none;

  background-color: transparent;

}

.cart-icon:hover{

  color: var(--main-color);

  transform: scale(1.1,1.1);

}

.cart-item-qty{

  position: absolute;

  right: -8px;

  font-size: 12px;

  color: #eee;

  background-color: var(--second-color);

  width: 18px;

  height: 18px;

  border-radius: 50%;

  text-align: center;

  font-weight: 600;

}

.about{

  display: grid;

  margin: 16rem 0rem;

  text-align: center;

}

.about h1{

  text-transform: lowercase ;

}

.products-container{

  display: flex;

  flex-wrap: wrap;

  justify-content: center;

  gap: 15px;

  margin-top: 20px;

  width: 100%;

}

.product-card{

 cursor: pointer;

  transform: scale(0.9, 0.9);

  transition: transform 0.5s ease;

 color: #324d67;

}

.product-card:hover{

  transform: scale(1,1)

}

.product-image{

  border-radius: 15px;

 background-color: #ebebeb;

  transform: scale(1, 1);

  transition: transform 0.5s ease;

}

.product-name{

  font-size: 18px;

  font-weight: 800;

  color: black;

}

.product-price{

  font-size: 16px;

  font-weight: 600;

  margin-top: 6px;

  color: black;

}

.hero-banner-container{

  padding: 100px 60px;

    display: flex;

    align-items: center;

    background-color: #fda702;

    border-radius: 15px;

    position: relative;

    height: 500px;

    line-height: 0.9;

    width: 100%;

}

.hero-banner-container .beats-solo{

  font-size: 20px;

  font-weight: 500;

  margin-bottom: 10px;

  padding-left: 3.1rem;

}

.hero-banner-container button{

   border-radius: 15px;

  padding: 10px 20px;

  background-color: #f02d34;

  color: white;

  border: none;

  /\* margin-top: 30px; \*/

  font-size: 18px;

  font-weight: 500;

  cursor: pointer;

  z-index:10000 !important;

}

.hero-banner-container h3{

  font-size: 4rem;

  margin-top: 4px;

}

.hero-banner-container h1{

  color: white;

  font-size: 10em;

  margin-left: -20px;

  text-transform: uppercase;

  padding: 0.5rem 3.7rem;

  font-weight: 900;

}

.hero-banner-image{

  position: absolute;

  top: 10%;

  right: 6%;

  height: 400px;

}

.heroBanner-midText{

  padding: 0 3rem;

    font-weight: 800;

}

.button1{

  padding: 0rem 3.1rem

}

.desc{

position: absolute;

right: 10%;

bottom: 5%;

width: 300px;

line-height: 1.3;

display: flex;

flex-direction: column;

 color: #324d67;

}

.desc p{

  color: #5f5f5f;

  font-weight: 100;

text-align: end;

}

.desc h5{

  margin-bottom: 12px;

  font-weight: 700;

  font-size: 16px;

  /\* color: black; \*/

  align-self: flex-end;

}

.products-heading{

  text-align: center;

  margin: 5rem 0rem 3rem;

 color: var(--text-color);

}

.products-heading h2{

  font-size: 40px;

  font-weight: 800;

}

.products-heading p{

  font-size: 16px;

  font-weight: 200;

}

.footer-banner-container{

  padding: 100px 40px;

  background-color: var(--main-color);

  border-radius: 15px;

  position: relative;

  height: 400px;

  line-height: 1;

  color: white;

  width: 100%;

  margin-top: 8rem;

}

.banner-desc{

  display: flex ;

  justify-content: space-between;

}

.banner-desc button{

  border-radius: 15px;

  padding: 10px 16px;

  background-color: white;

  color: red;

  border: none;

  margin-top: 20px;

  font-size: 18px;

  font-weight: 500;

  cursor: pointer;

}

.banner-desc .left h3{

  font-weight: 900;

  font-size: 80px;

  margin-left: 25px;

}

.banner-desc .left p{

  margin: 2px 18px;

    padding-left: 0.5rem;

    font-size: 1.1rem;

    font-weight: 400;

}

.footer-banner-image{

  position: absolute;

  top: -35%;

  left: 8%;

  top: -25%;

  left: 26%;

}

.banner-desc .right{

  line-height: 1;

  margin-left: 3rem;

}

.banner-desc .right h3{

  font-weight: 800;

  font-size: 60px;

}

.banner-desc .right p{

  font-size: 18px;

}

.banner-desc .right .company-desc{

  font-size: 14px;

  font-weight: 300;

}

.cart-wrapper{

  width: 100vw;

  background: rgba(0, 0, 0, 0.5);

  position: fixed;

  right: 0;

  top: 0;

  z-index: 100;

  /\* will-change: transform; \*/

  transition: all 1s ease-in-out;

}

.cart-container{

  height: 100vh;

  width: 600px;

  background-color: white;

  float: right;

  padding: 40px 10px;

  position: relative;

}

.footer-container{

  color: #324d67;

  /\* text-align: center; \*/

  margin-top: 20px;

  padding: 30px 10px;

  /\* font-weight: 700; \*/

  display: flex;

  flex-direction: column;

  align-items: center;

  gap: 10px;

  justify-content: center;

}

.footer-container .icons{

  font-size: 30px;

  display: flex;

  gap: 10px;

}

 .cart-heading{

  display: flex;

  align-items: center;

  font-size: 18px;

  font-weight: 500;

  cursor: pointer;

  gap: 2px;

  margin-left: 10px;

  border: none;

  background-color: transparent;

}

.cart-heading .heading{

  margin-left: 10px;

}

.cart-num-items{

  margin-left: 10px;

  color: #f02d34;

}

.empty-cart{

  margin:40px;

  text-align:center;

}

.empty-cart h3{

  font-weight: 600;

  font-size: 20px;

}

.cancel{

  cursor: pointer;

}

.product-container{

  margin-top: 15px;

  overflow: auto;

max-height: 70vh;

padding: 20px 10px;

}

.product{

  display: flex;

 gap: 30px;

  padding: 20px;

}

.product .cart-product-image{

  width:180px ;

  height: 150px;

    border-radius: 15px;

 background-color: #ebebeb;

}

.product-desc{

  display: flex;

  justify-content: space-between;

}

.item-desc .flex{

  display: flex;

  justify-content: space-between;

  width: 350px;

   color: #324d67;

}

.item-desc .bottom{

  margin-top: 60px;

}

.flex h5{

  font-size: 24px;

}

.flex h4{

  font-size: 20px;

}

.total{

  display: flex;

  justify-content: space-between;

}

.total h3{

  font-size: 22px;

}

.remove-item{

  font-size: 24px;

  color: #f02d34;

  cursor: pointer;

  background: transparent;

  border: none;

}

.cart-bottom{

  position: absolute;

  bottom: 12px;

  right: 5px;

width: 100%;

padding: 30px 65px;

}

.btn-container{

  width: 400px;

  margin: auto;

}

.btn{

  width: 100%;

  max-width: 400px;

  padding: 10px 12px;

  border-radius: 15px;

  border: none;

  font-size: 20px;

  margin-top: 10px;

  margin-top: 40px;

  text-transform: uppercase;

  background-color: #f02d34;

  color: #fff;

  cursor: pointer;

    transform: scale(0.9, 0.9);

  transition: transform 0.5s ease;

}

.btn:hover{

  transform: scale(1,1);

}

.product-detail-container{

  display: flex;

 gap: 40px;

 margin: 40px;

 margin-top: 60px;

 color: #324d67;

}

.product-detail-image{

  border-radius: 15px;

   background-color: #ebebeb;

  width: 400px;

  height: 400px;

  cursor: pointer;

  transition: .3s ease-in-out;

}

.product-detail-image:hover{

  background-color: var(--main-color);

}

.small-images-container{

  display: flex;

  gap: 10px;

  margin-top: 20px;

}

.small-image{

  border-radius: 8px;

  background-color: #ebebeb;

  width: 70px;

  height: 70px;

  cursor: pointer;

}

.reviews{

  margin-top: 10rem;

}

.review-container{

  display: grid;

  grid-template-columns: repeat(auto-fit, minmax(220px, auto));

  gap: 1.5rem;

  margin-top: 2rem ;

  margin-bottom: 2rem;

  padding: 0px 100px;

}

.stars{

  margin-top: 1rem;

  font-size: 1rem;

  color: var(--main-color);

}

.review-container .box p{

  font-size: 0.938rem;

  font-style: italic;

  margin: 0.5rem 0.5rem;

}

.review-container .box p:hover{

  color: var(--main-color);

}

.review-container .box h2{

  margin-top: 1rem;

  font-size: 1.5rem;

  font-weight: 600;

}

.review-container .box{

  padding: 10px;

  border: 2px solid var(--text-color);

  border-radius: 0.5rem;

  text-align: center;

  box-shadow: var(--box-shadow);

  transition: 0.5s;

}

.review-container .box img{

  width: 70px;

  height: 70px;

  object-fit: cover;

  border-radius: 50%;

}

.review-container .box:hover{

  background: var(--second-color);

  color: var(--main-color);

}

.newsletter{

  display: flex;

  flex-direction: column;

  align-items: center;

  justify-content: center;

  text-align: center;

  row-gap: 2rem;

  margin-top: 7rem;

}

.newsletter h2{

  font-size: 1.8rem;

  text-transform: uppercase;

}

.newsletter form{

  background: var(--bg-color);

  padding: 10px;

  border-radius: 0.5rem;

  box-shadow: var(--box-shadow);

}

.newsletter form input{

  width: 200px;

  border: none;

  outline: none;

  background: transparent;

}

.newsletter form .email-btn{

  background: var(--main-color);

  border: none;

  padding: 8px 14px;

  color: var(--bg-color);

  text-transform: uppercase;

  font-size: 1rem;

  font-weight: 500;

  max-width: 115px;

  border-radius: 0.5rem;

  transition: 0.5s;

}

.newsletter form .email-btn:hover{

  color: var(--second-color);

}

.footer{

  display: grid;

  grid-template-columns: repeat(auto-fit, minmax(150px, auto));

  padding: 5rem 8rem 0rem;

  gap: 1.5rem;

  background: #fff;

  margin-bottom: 40px;

}

.footer-box a{

  margin-bottom: 10px;

  font-size: 1rem;

  color: var(--text-color);

  transition: 0.6s;

}

.footer-box a:hover{

  color: var(--main-color);

}

.footer-box p{

  font-size: 0.938rem;

  margin-bottom: 10px;

  padding: 0 10px;

}

.footer-box span{

  margin-bottom: 10px;

  font-size: 1rem;

  color: var(--text-color);

}

.social{

  display: flex;

  align-items: center;

  column-gap: 0.5rem;

}

.social a .bx{

  font-size: 24px;

  color: var(--text-color);

  padding: 10px;

  background: var(--third-color);

  border-radius: 5rem;

  transition: 0.5s;

}

.social a .bx:hover{

  background: var(--main-color);

  color: var(--bg-color);

}

.footer-box h3{

  font-weight: 600;

  margin-bottom: 10px;

  color: var(--text-color);

  padding: 0 10px;

}

.contact{

  display: flex;

  flex-direction: column;

}

.contact span{

  display: flex;

  align-items: center;

}

.contact i{

  font-size: 20px;

  margin-right: 1rem;

}

.product-detail-desc h3{

  margin-top: 10px;

}

.product-detail-desc p{

  margin-top: 10px;

}

.reviews p{

  color: #324d67;

  margin-top: 0px;

}

.product-detail-desc .price{

  font-weight: 700 ;

  font-size: 26px;

  margin-top: 30px;

}

.price .old-price, .product-price .old-price, .price .old-price{

  color: gray;

  text-decoration: line-through;

}

.product-detail-desc .quantity{

  display: flex;

  gap: 20px;

  margin-top: 10px ;

  align-items: center;

}

.product-detail-desc .buttons{

  display: flex;

  gap: 30px;

}

.buttons .add-to-cart{

  padding: 10px 20px;

  border: 1px solid var(--text-color) ;

  margin-top: 40px;

  font-size: 18px;

  font-weight: 500;

  background-color: white;

  color: var(--text-color);

  cursor: pointer;

  width: 200px;

   transform: scale(1, 1);

  transition: transform 0.5s ease;

}

.buttons .add-to-cart:hover{

  transform:scale(1.1,1.1)

}

.buttons .buy-now{

  width: 200px;

  padding: 10px 20px;

  background-color: var(--text-color);

  color: white;

  border: none;

  margin-top: 40px;

  font-size: 18px;

  font-weight: 500;

  cursor: pointer;

   transform: scale(1, 1);

  transition: transform 0.5s ease;

}

.buttons .buy-now:hover{

  transform:scale(1.1,1.1)

}

.quantity-desc{

  border: 1px solid black;

padding: 5px;

}

.quantity-desc span{

  font-size: 16px;

  padding: 6px 12px;

  cursor: pointer;

}

.quantity-desc .minus{

  color: var(--text-color);

}

.quantity-desc .num{

  border-right: 1px solid black;

  border-left: 1px solid black;

  font-size: 20px;

}

.quantity-desc .plus{

  color: var(--text-color);

}

.maylike-products-wrapper{

  margin-top: 120px;

}

.maylike-products-wrapper h2{

  text-align: center;

  margin: 50px;

  color: #324d67;

  font-size: 28px;

}

.maylike-products-container{

  display: flex;

  justify-content: center;

  gap: 15px;

  margin-top: 20px;

}

.max-qty{

  font-weight: 500;

  color: #f02d34;

}

.success-wrapper, .cancel-wrapper{

  background-color: white;

  min-height: 60vh;

}

.success, .cancel{

  width: 1000px;

margin: auto;

margin-top: 160px;

  background-color: #dcdcdc;

  padding: 50px;

  border-radius: 15px;

  display: flex;

  justify-content: center;

  align-items: center;

  flex-direction: column;

}

.success .icon {

   color: green;

   font-size: 40px;

}

.success h2{

  text-transform: capitalize;

  margin-top: 15px 0px;

  font-weight: 900;

  font-size: 40px;

  color:#324d67;

}

.success .email-msg{

   font-size: 16px;

  font-weight: 600;

  text-align: center;

}

.cancel p{

  font-size: 20px;

  font-weight: 600;

}

.success .description{

  font-size: 16px;

  font-weight: 600;

  text-align: center;

  margin: 10px;

  margin-top: 30px;

}

.success .description .email{

  margin-left: 5px;

  color: #f02d34;

}

.product-max-qty{

  margin-top: 10px;

}

.review-heading{

  display: flex;

  margin-top: 6rem;

  margin-bottom: 3rem;

  text-transform: uppercase;

  font-size: 1.8rem;

  color: black;

  justify-content: center;

}

@media screen and (max-width:800px) {

  .hero-banner-container{

    height: 560px;

  }

  .hero-banner-image{

    width: 77%;

    height: 62%;

    top: -2%;

    right: -6%;

  }

  .footer-banner-container{

    height: 560px;

    margin-top: 80px;

  }

  .footer-banner-image{

    width: 77%;

    left: 30%;

    top: 6%;

    height: 56%

  }

  .banner-desc .left h3{

    font-weight: 900;

    font-size: 50px;

    margin-left: 5px;

  }

  .banner-desc .left p{

    margin:18px;

  }

  .banner-desc .right h3{

    font-size: 45px;

  }

  .banner-desc .right p{

    font-size: 18px;

  }

  .banner-desc .right .company-desc{

    font-size: 14px;

  }

  .banner-desc{

    flex-wrap: wrap;

    gap: 20px;

  }

  .hero-banner-container{

    line-height: 1.3;

  }

  .hero-banner-container h1{

    font-size: 50px;

  }

 .hero-banner-container h3{

    font-size: 40px;

  }

  .hero-banner-container button{

    margin-top: 90px;

    z-index: 10000;

  }

  .desc{

   bottom: 60px;

  }

  .product-detail-container{

    flex-wrap: wrap;

    }

  .product-detail-container .product-detail-image{

    width: 350px;

    height: 350px;

  }

  .cart-container{

    width: 415px;

  padding: 4px;

  }

  .cart-heading{

    margin-top: 35px;

  }

  .product-container{

    margin-top: 10px;

  }

  .review-container{

    display: flex;

    align-items: center;

  }

  .product{

    padding: 20px 5px;

  }

  .product .cart-product-image{

    width: 25%;

    height: 25%;

  }

  .buttons .add-to-cart{

    width: 150px;

  }

  .buttons .buy-now{

    width: 150px;

  }

  .product-detail-container{

    margin: 20px;

  }

.item-desc .flex{

  width: 200px;

}

.top{

  flex-wrap: wrap;

  gap: 10px;

}

.item-desc .bottom{

  margin-top: 30px;

}

.flex h5{

  font-size: 16px;

   color: #324d67;

}

.flex h4{

  font-size: 16px;

   color: black;

}

.cart-bottom{

  padding: 30px;

}

.total h3{

  font-size: 20px;

}

.track {

  animation: marquee 10s linear infinite;

  width: 550%;

}

.success-wrapper, .cancel-wrapper{

  min-height: 69vh;

}

.success, .cancel {

   width: 370px;

  margin-top: 100px;

  padding: 20px;

}

.success{

  height: 350px;

}

.success h2{

  font-size: 17px;

}

.btn-container{

  width: 300px;

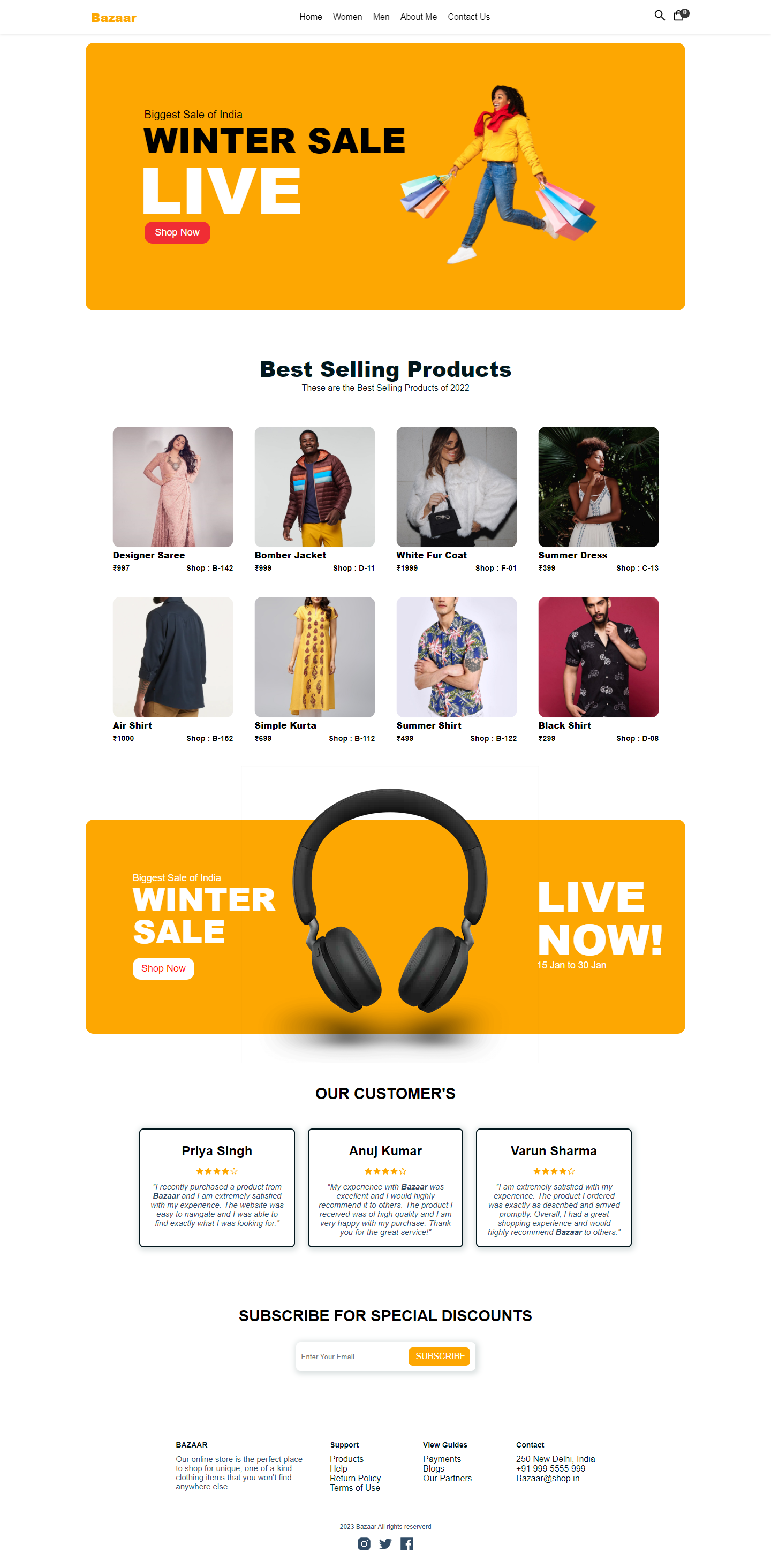
  margin: auto;

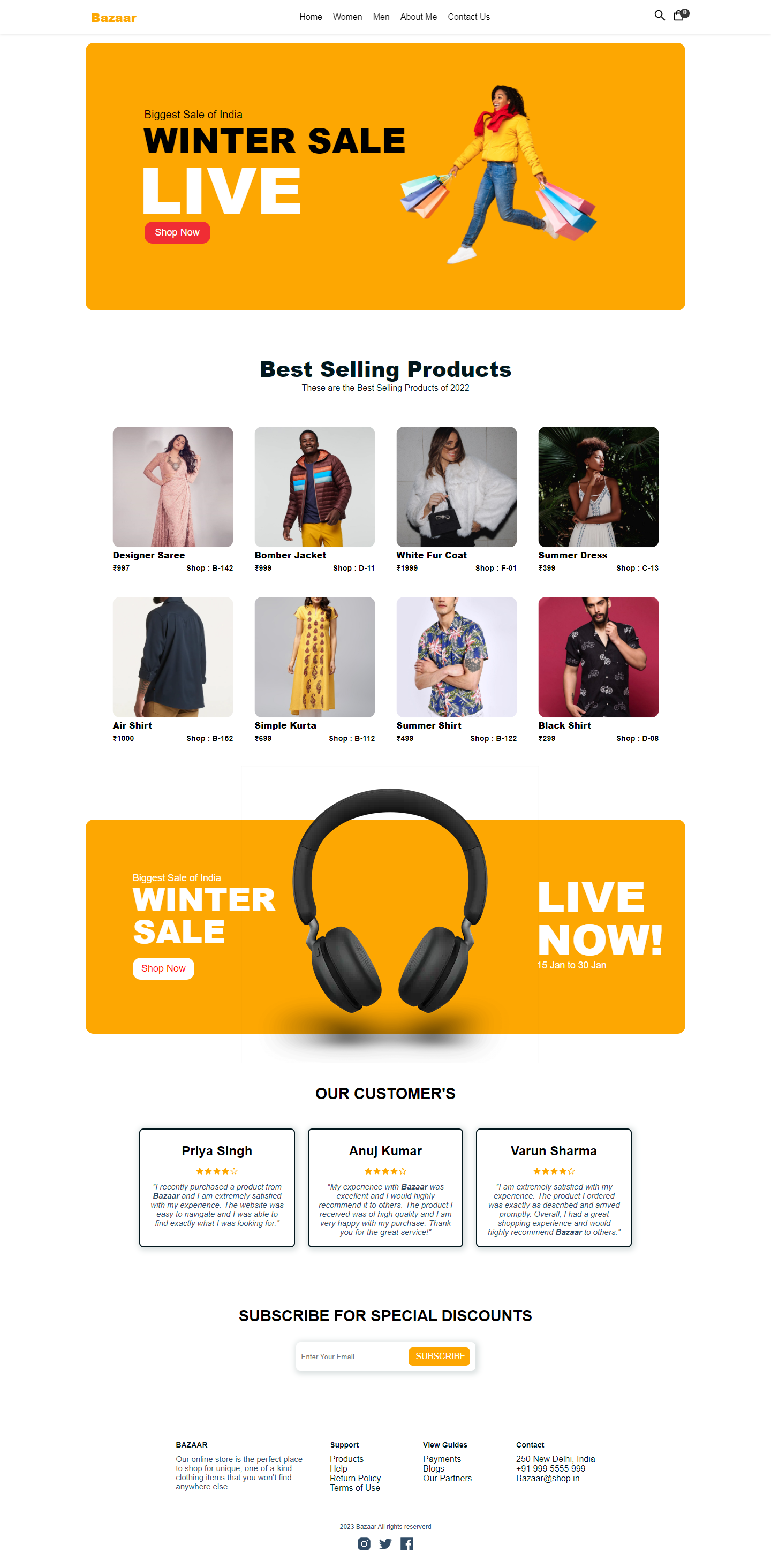
}

}

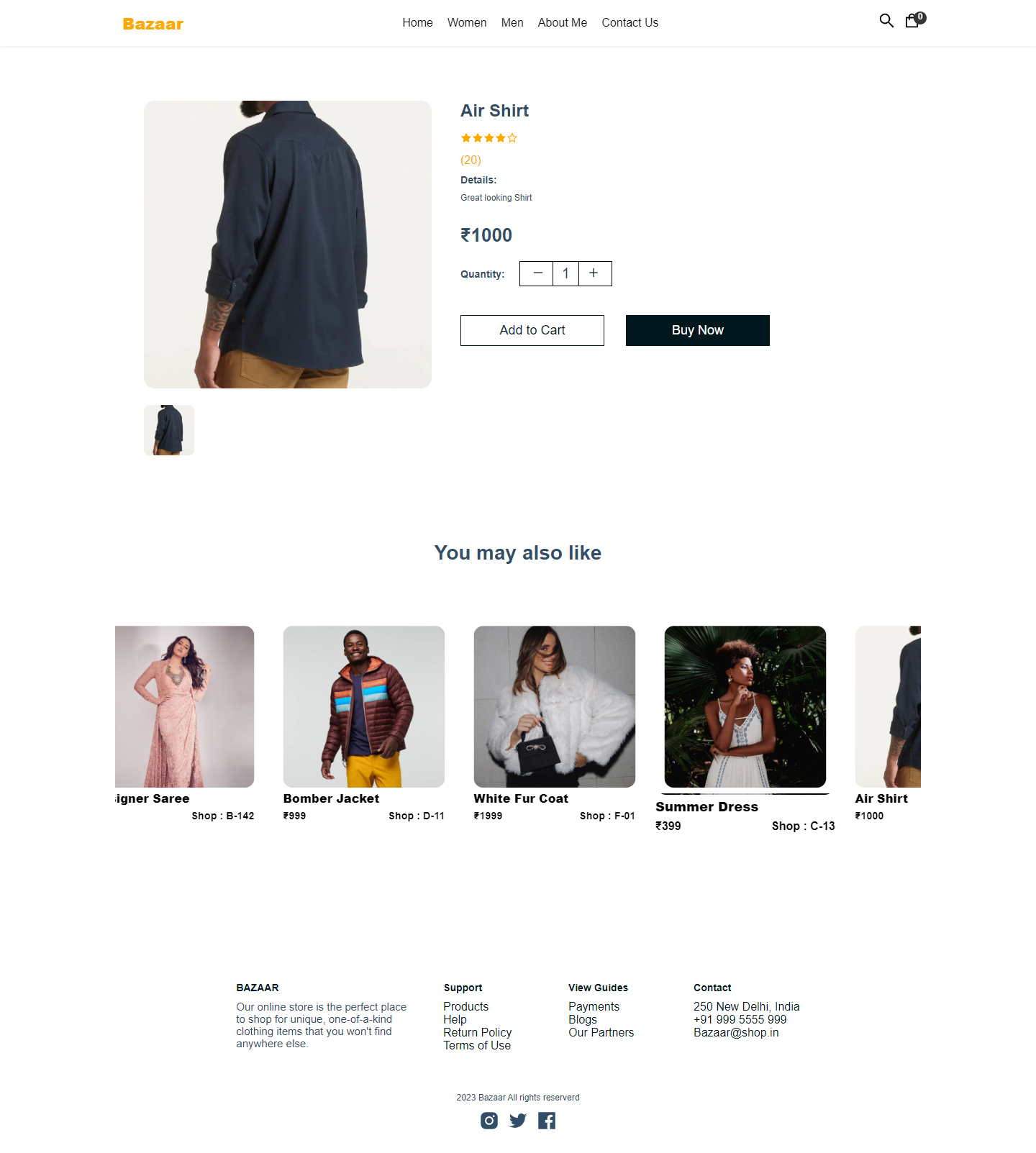
**SCREENSHOTS**

**HOME PAGE**

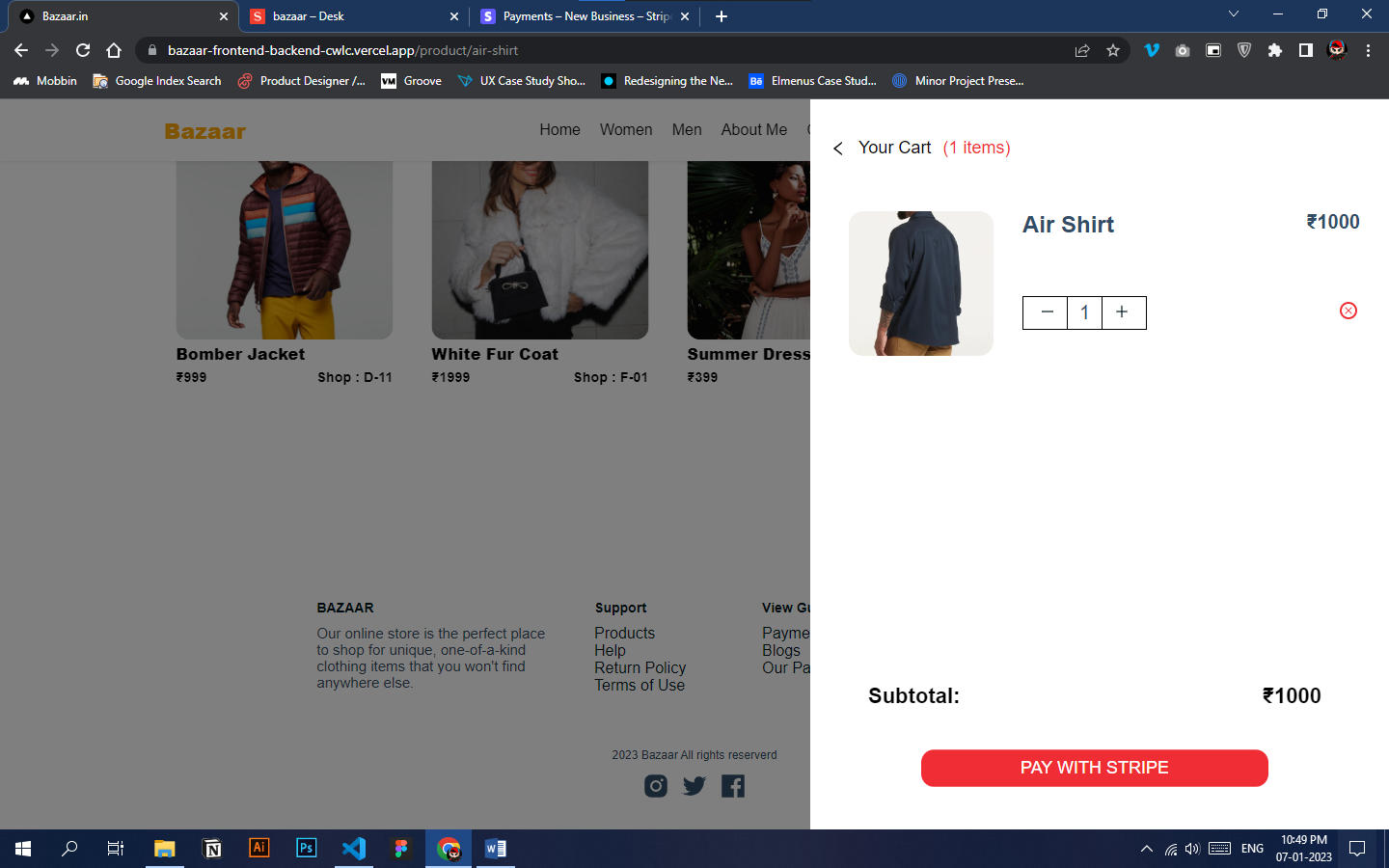
****

****

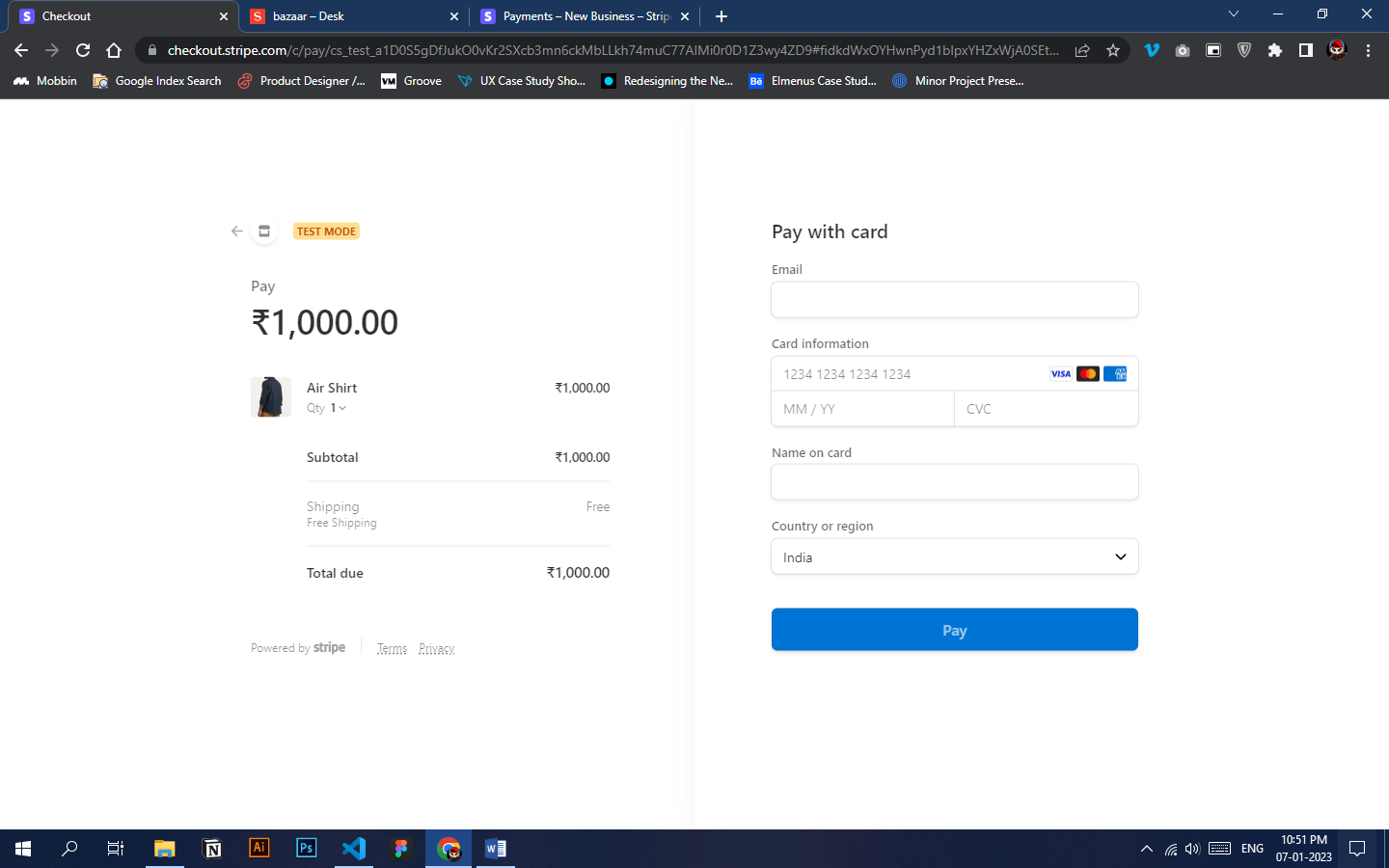
**PRODUCT PAGE**

****

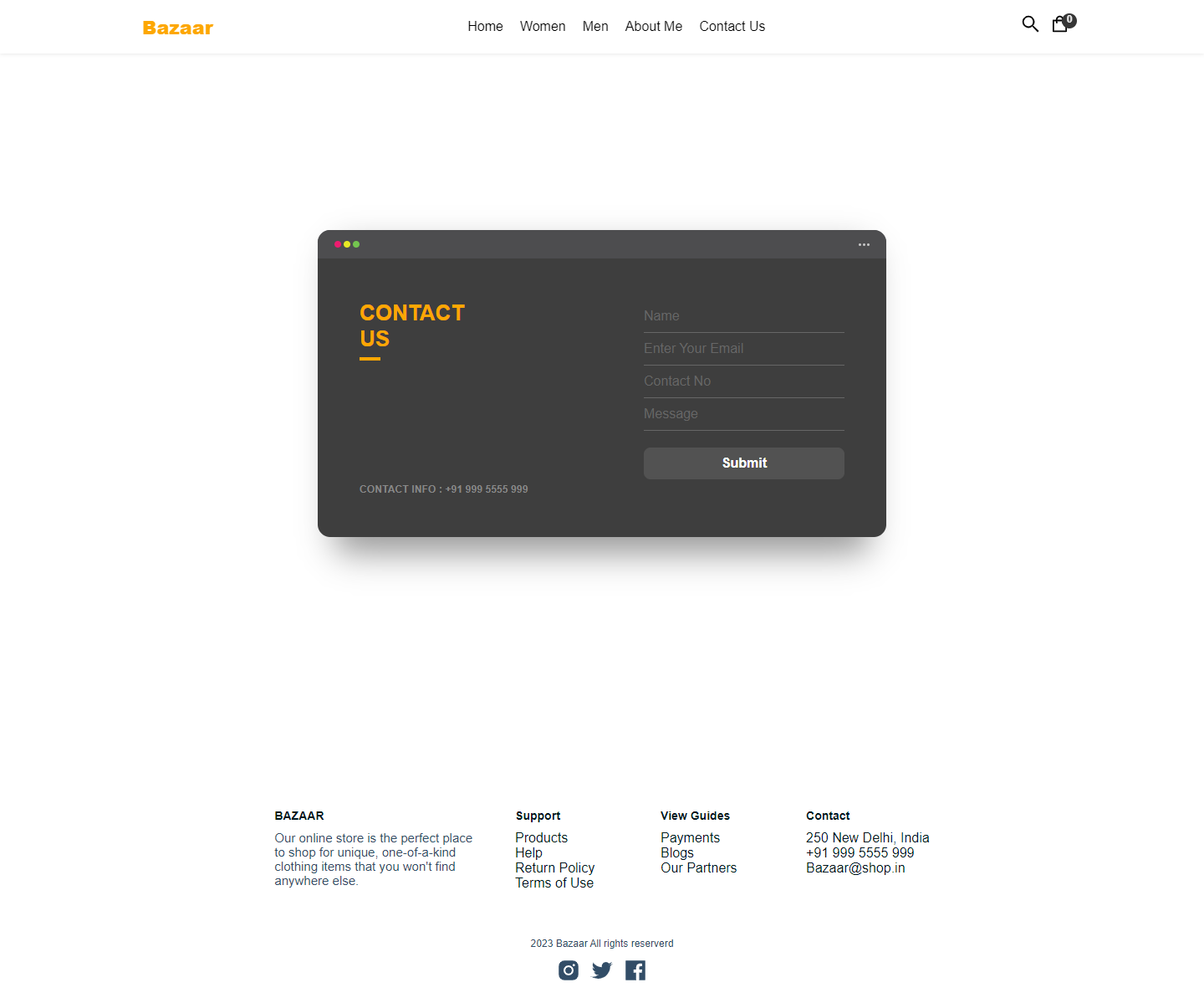
**CART PAGE**

****

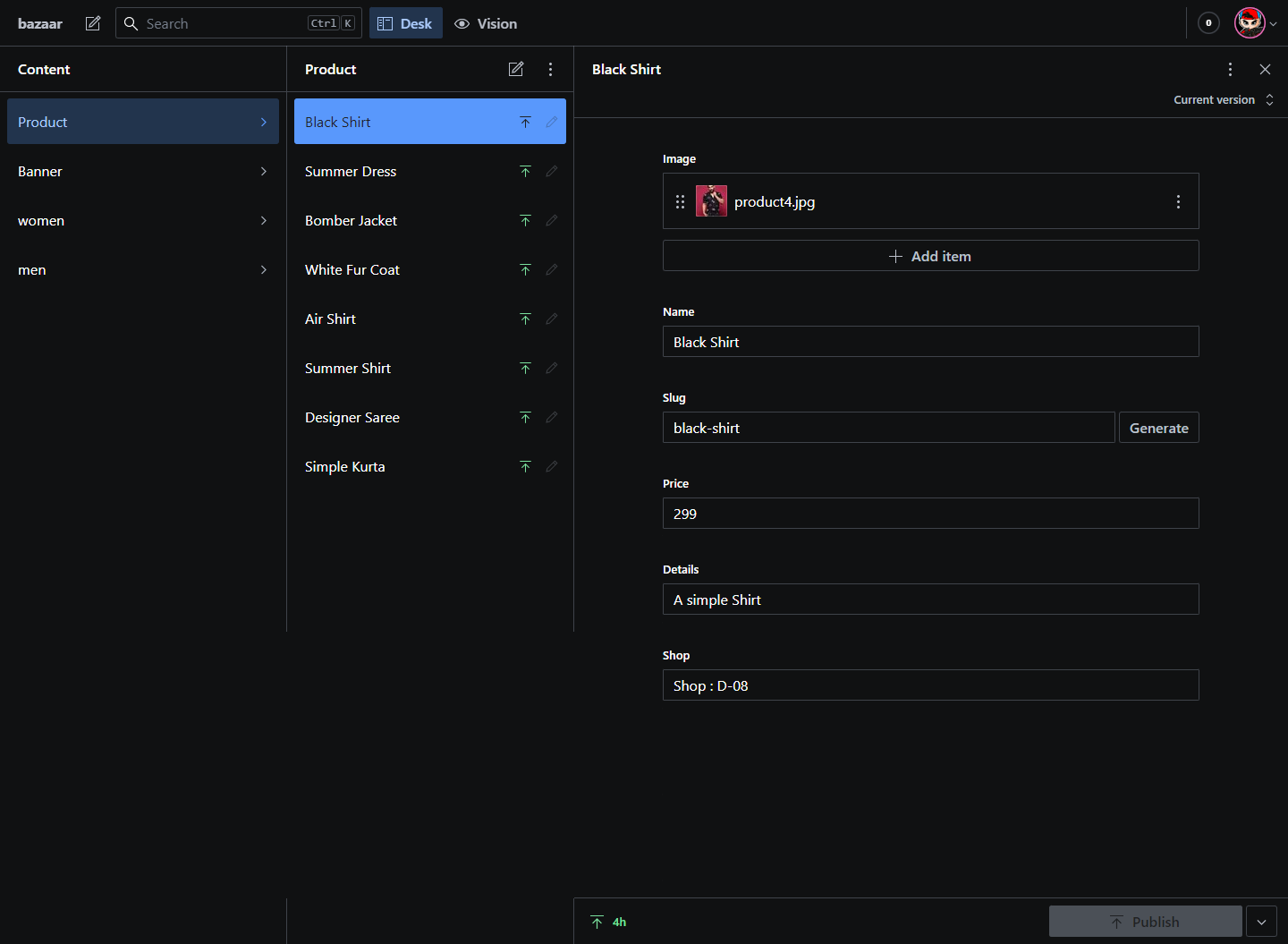
**PAYMENT PAGE**

****

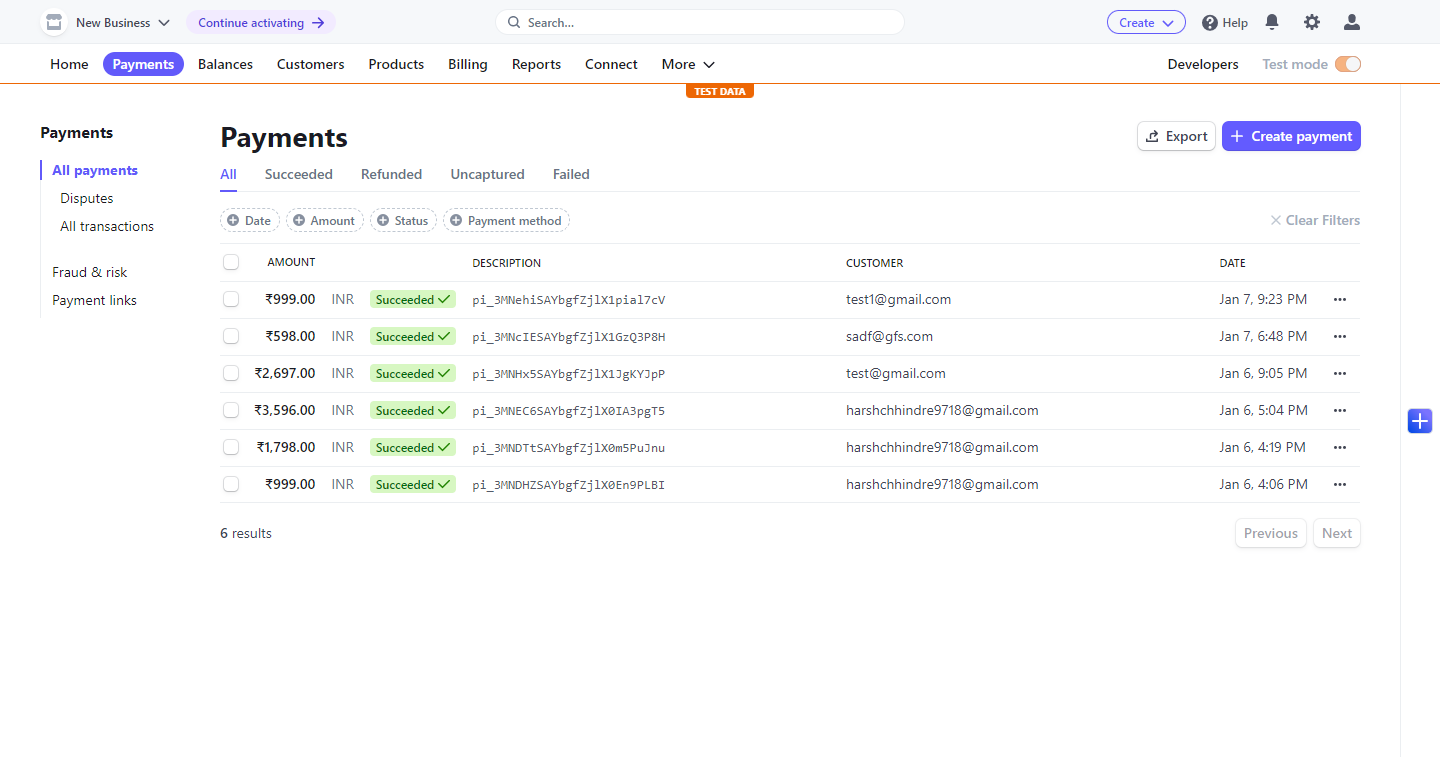
**CONTACT US PAGE**

****

**PRODUCT BACKEND PAGE**

****

**PAYMENT BACKEND PAGE**

****

**SUMMARY & CONCLUSION**

**Summary**

An online store is a virtual store on the Internet where customers can browse the catalogue and select products of interest. The selected items may be collected in a shopping cart. At checkout time, the items in the shopping cart will be presented as an order. At that time, more information will be needed to complete the transaction. Usually, the customer will be asked to fill or select a billing address, a shipping address, a shipping option, and payment information such as credit card number. An e- mail notification is sent to the customer as soon as the order is placed.

**Future Scope of the Project**

"On-Line Shopping" is a web-based project which is made for remote-shopping or shopping through Internet. As the technology is being advanced the way of life is changing accordance. Now a day’s we can place the order for anything from our home. There is no need to go the shop of the things we want. The order can be placed online through Internet. The payment, the confirmation of purchasing, we Cando everything we want. Now we can think that how the days have been changed with time. People had to stand in rows to wait there terms to buy a particular thing from a popular shop. But what is happening now a day's we can extremely surprise that those things can be available on the door-step in few hours In future we will try to make this website which work so flexible and beneficial for the customer and also try to make

Smooth service. Recommendation system

Improvement /Update

Important updates -:

\* We have updated the website by adding a new mode of transactions i.e. multiple card

Payments (visa, American express, credit/debit card)

\*There is shipping address added with the payment page which was not in previous update.

\* A new feedback form is being provided to the customer for to give feedback whenever

They fell dissatisfaction among any products or page.

\*New products are being added to the page,so that customers can choose wide range to buy.

\* An Admin is been added for the page whose role will be to add, remove and check the customer status.

\*New ads are being added so customer can surf another page for more products variety.

**Conclusions**

Before online shopping there was only shopping in store but as the world has got more technology, shopping online has become more popular. I think that as the time goes on soon everyone will shop online, but I think shops will still stay in business because without shops people can't visually see what their buying and people may want their item that exact time when they see it and only by actually going shopping you can have the item you want at the exact same time you purchase it so that's why I think that online shopping will never take over actual shops on the high street. They could improve online shopping by making the delivery time a lot quicker or allowing people to make a delivery time exactly when they want it. They could also improve online shopping's security levels, with this they could gain a lot more customers.

**REFERENCES**

**1) https://msdn.microsoft.com/en-us/library/dn762121(v=vs.120).aspx**

**2) https://www.w3schools.com/**

**3) https://www.tutorialspoint.com/csharp/**

**4) https://www.w3schools.com/asp/**

**5) https://www.w3schools.com/howto/how to css checkout form.asp**