

“E-Commerce Website Using MERN Tech Stack”

Submitted in partial fulfillment of the requirements of

the degree

B.Tech. (Computer Engineering)

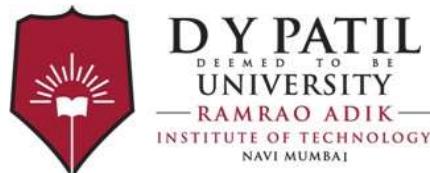
By

Deeptanshu Lal Roll No. 22CE1285

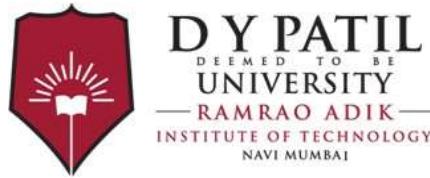
Krish Pradeshi Roll No. 22CE1232

Supervisor

Dr. Shital Patil



Department of Computer Engineering
Ramrao Adik Institute of Technology,
Sector 7, Nerul, Navi Mumbai
(Under the ambit of D. Y. Patil Deemed to be University)
April 2024



Ramrao Adik Institute of Technology
(Under the ambit of D. Y. Patil Deemed to be University)
Dr. D. Y. Patil Vidyanaagar, Sector 7, Nerul, Navi Mumbai 400 706.

Certificate

This is to certify that, the Mini Project- II entitled

“E-Commerce Product Catalog”
is a bonafide work done by,

Deeptanshu Lal Roll No. 22CE1285
Krish Pradeshi Roll No. 22CE1232

and is submitted in the partial fulfillment of the requirement for the degree of

B. Tech. in Computer Engineering

to the
D. Y. Patil Deemed to be University.

Supervisor

(Dr. Shital Patil)

Dr. Siuli Das
Mini Project Coordinator

Dr. A. V. Vidhate
Head of Department

Dr. Mukesh Patil
Principal

Approval Sheet

This Mini Project - II entitled "**E-Commerce Website Using MERN Tech Stack**" by
Deeptanshu Lal Roll No. 22CE1285, Krish Pradeshi Roll No. 22CE1232 is approved
in the partial fulfillment of the requirement for the degree of **B. Tech. in Computer
Engineering.**

Examiners

1.....
(Internal Examiner Name & Sign)

2.....
(External Examiner name & Sign)

Date:

Place:

Abstract

In the digital era, e-commerce platforms have revolutionized the way businesses present and market their products to consumers worldwide. The e-commerce catalog serves as a pivotal tool in this landscape, providing a structured, visual, and interactive representation of available products and services. The primary objective of an e-commerce catalog is to offer users a seamless and intuitive browsing experience, enabling them to explore products, gather information, and make informed purchase decisions. It amalgamates product details such as images, descriptions, specifications, pricing, and reviews into a cohesive and accessible format. The catalog's design emphasizes user-friendliness, ensuring that customers can effortlessly navigate through categories, filter results, and access relevant information.

Contents

1 Introduction

 1.1 Overview

 1.2 Motivation

 1.3 Problem Statement and Objectives

2 Literature Survey

 2.1 Survey of Existing System

 2.2 Limitations of Existing System

3 Proposed System

 3.1 Problem Statement

 3.2 Proposed Methodology / Techniques

 3.3 System Design

 3.4 Details of Hardware and Software Requirements

4 Results and Discussion

 4.1 Implementation Details

 4.2 Project Outcomes

5 Conclusion and Future Work

- References
- Appendix
 - Weekly progress report.
- Acknowledgement

Chapter 1

Introduction

In the realm of digital commerce, the e-commerce catalogue emerges as a cornerstone, defining the online shopping journey for global consumers. Serving as a digital storefront, an e-commerce catalogue meticulously organizes and showcases a curated selection of products and services, enriching the browsing experience with accessibility and visual allure. This introduction delves into the essence, evolution, and technological significance of e-commerce catalogues, particularly within the context of our mini-project.

Within our technical exploration, e-commerce catalogues play a pivotal role in orchestrating user interactions, shaping backend functionalities, and driving frontend engagements. Through a lens of technology, these catalogues serve as dynamic databases, efficiently managing product information, categories, and user preferences. Leveraging frameworks like the MERN (MongoDB, Express.js, React.js, Node.js) stack, our mini-project encapsulates the essence of modern web development, intertwining data management, server-side scripting, and client-side rendering to construct a robust e-commerce catalogue system.

Motivation

The core drive for selecting the MERN Tech Stack and embarking on this project was to gain hands-on experience with real-world website development and understand the intricate workings involved. It's been a journey to grasp the depth of effort required in creating a fully functional platform. This endeavor serves as an invaluable opportunity to immerse oneself in the nuances of real-world application, gaining insights into the intricate processes involved. Through this journey, I aim to not only expand my technical skill set but also cultivate a deeper understanding of the holistic framework that underpins modern web development.

Objectives:

The primary objectives include:

- I. **Strengthening Technical Proficiency:** The primary objective of selecting the MERN (MongoDB, Express.js, React.js, Node.js) Tech Stack for this project is to foster a deeper understanding of contemporary web development practices. By immersing ourselves in the intricacies of these technologies, we aim to enhance our technical proficiency and gain practical insights into the complexities of building a fully functional e-commerce platform.

- II. **Exploring Software Architecture:** By delving into the architecture of a MERN-based application, we seek to gain a comprehensive understanding of its structural components and their interrelationships. This includes exploring concepts such as the Model-View-Controller (MVC) pattern, RESTful API design, state management in React, and database integration with MongoDB.

- III. **Building Brand Identity and Brand Aesthetics:** To communicate the brand's values, identity, and value proposition through consistent branding elements, compelling visuals, and engaging content that resonate with the target audience.

Chapter 2

Literature Survey

In the MERN Stack, the amalgamation of MongoDB, Express.js, Node.js, and React.js forms a robust framework that aids developers in addressing real-life problems and daily development needs. MERN stack, an acronym for MongoDB, Express, React, and Node, is widely favored by developers due to its ease, speed, and smoothness in the development process.

- I. **MongoDB:** MongoDB, a leading NoSQL database, stores data in documents called collections, offering an alternative to relational databases. Its use of BSON format, an extension of JSON, enhances performance, scalability, and reliability. MongoDB's strong query processes, easy accessibility, customizable schema, and security make it more widely used than SQL databases.
- II. **Express:** Express.js, an essential Node.js package, facilitates the creation of server-side web applications. Express allows the development of customized, reliable, and scalable servers to connect with the frontend. It aids in setting up middleware and API routes for data transmission between frontend and backend.
- III. **React:** React.js, a popular frontend JavaScript library, simplifies the creation of web applications through its component-based architecture. Component reusability across different pages or projects saves time and resources. React's JSX enables the creation of creative interfaces by integrating HTML directly into JavaScript. JSX facilitates the sequential execution of functions and expressions within curly braces.
- IV. **Node:** NodeJS enables JavaScript execution outside the browser, making it suitable for server-side applications. Node's asynchronous nature and event-driven processing enhance performance. Its built-in package manager, npm, simplifies package installation with a single command. Node is widely adopted by large organizations for real-time server applications. [1 Arora, Karishma & Vaishnavi, & Nagpal, Jai. (2023). Implementation of MERN: A Stack of Technologies to Design Effective Web Based Freelancing Applications. International Journal of Scientific Research in Computer Science, Engineering and Information Technology. 23-32. 10.32628/CSEIT23902104.]

E-commerce, also known as electronic commerce, involves buying and selling products over the internet, along with the transfer of money and data to complete transactions. Initially less popular, the rise in mobile phone usage spurred widespread interest in online shopping, leading to its popularity.

There are four main types of e-commerce business models:

1. B2C (Business to Consumer): Businesses sell products directly to end-users over the internet.
2. B2B (Business to Business): Transactions occur between large companies, organizations, and businesses, constituting a significant portion of e-commerce.
3. C2B (Consumer to Business): Individuals sell goods or services to companies, often through online platforms or electronic mediums. Examples include freelancing, where consumers set their prices for work assignments.
4. C2C (Consumer to Consumer): This model enables consumers to exchange goods directly with each other, facilitated by online platforms. Transactions may involve charging fees to motivate buyers and sellers. [2 Nagothu Diwakar Naidu, Pentapati Adarsh, Sabharinadh Reddy, Gumpula Raju, Uppu Sai Kiran, & Vikash Sharma. "E-Commerce Web Application by Using MERN Technology." International Journal for Modern Trends in Science and Technology 7, 1–5 (2021).]

Limitation of Existing System

Product Information Accuracy: Maintaining accurate and up-to-date product information is challenging. Prices, stock availability, product descriptions, and images need regular updates, and discrepancies can lead to customer dissatisfaction.

Scalability Issues: As a business grows and adds more products to its catalog, managing and organizing them can become complex. This can affect the performance and user experience of the e-commerce platform.

Product Representation: Representing physical products accurately through digital means is challenging. Customers can't touch or feel products, so providing detailed descriptions, images, and videos becomes crucial. However, conveying all product details effectively can be challenging.

Mobile Responsiveness: As more customers shop using mobile devices, ensuring that the e-commerce catalog is optimized for various screen sizes and devices is crucial. However, managing a responsive catalog that provides a consistent user experience across devices can be challenging.

Cost of Maintenance: Regularly updating, optimizing, and managing an e-commerce catalog requires resources, including manpower and technology. The ongoing cost of maintaining a comprehensive and user-friendly catalog can be significant for businesses.

[3 Nagothu Diwakar Naidu, Pentapati Adarsh, Sabharinadh Reddy, Gumpula Raju, Uppu Sai Kiran, & Vikash Sharma. "E-Commerce Web Application by Using MERN Technology." International Journal for Modern Trends in Science and Technology 7, 1–5 (2021).]

Chapter 3

3.1 Problem Statement

Project aims to develop a user-friendly e-commerce platform with organized project structure, backend functionality for product and user management, and frontend features including shopping cart functionality, product search, and responsive design. Deployment and documentation will ensure accessibility and reliability of the platform.

- I. **Backend Setup:** Develop models and routes for product and user management functionalities.
- II. **Frontend Development:** Build React components for essential features like product listings and shopping cart.
- III. **Shopping Cart:** Develop functionality for adding/removing items and displaying total price.
- IV. **Payment Portal:** Enable payment methods using stripe or something similar and implement filters for categories and price range.
- V. **Deployment:** Deploy backend to a server (e.g., AWS) and frontend to a static hosting service (e.g., Vercel).

3.2 System Design

Frontend Module:

- User Authentication (User Page, Register, Login)
- Navigation (Contact, Nav Links)
- Product Display (Project Views, Product Cards, Search)
- Shopping Cart Management (Cart, Total, Checkout)
- Submodules interact seamlessly to provide a cohesive user experience.

Backend Module:

- Database Module (MongoDB)
 - Stores user details and product information.
 - NoSQL database for flexible data storage and schema evolution.
- API Module
 - Handles CRUD operations for products.
 - Integrates with third-party services like Stripe for payment processing.

Communication:

- Frontend and Backend communicate through the API module.
- Frontend sends HTTP requests to the API.
- API processes requests, interacts with the Database Module, and returns responses.

Technologies and Frameworks:

- Frontend: React or Angular
- Backend: Node.js with Express
- Stripe integration follows API documentation and best practices for secure payment handling and PCI compliance.

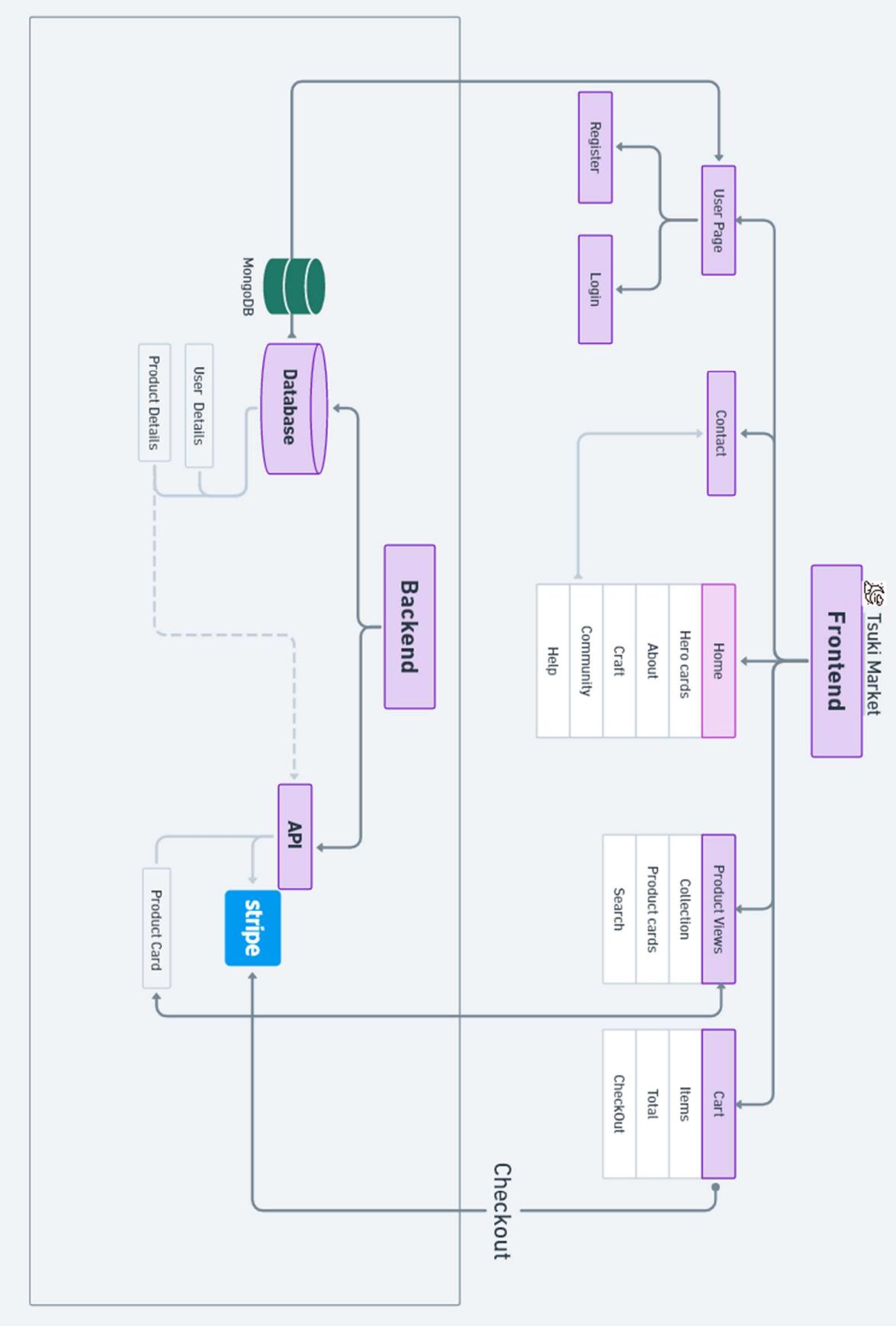


Fig- 3.1 System Design

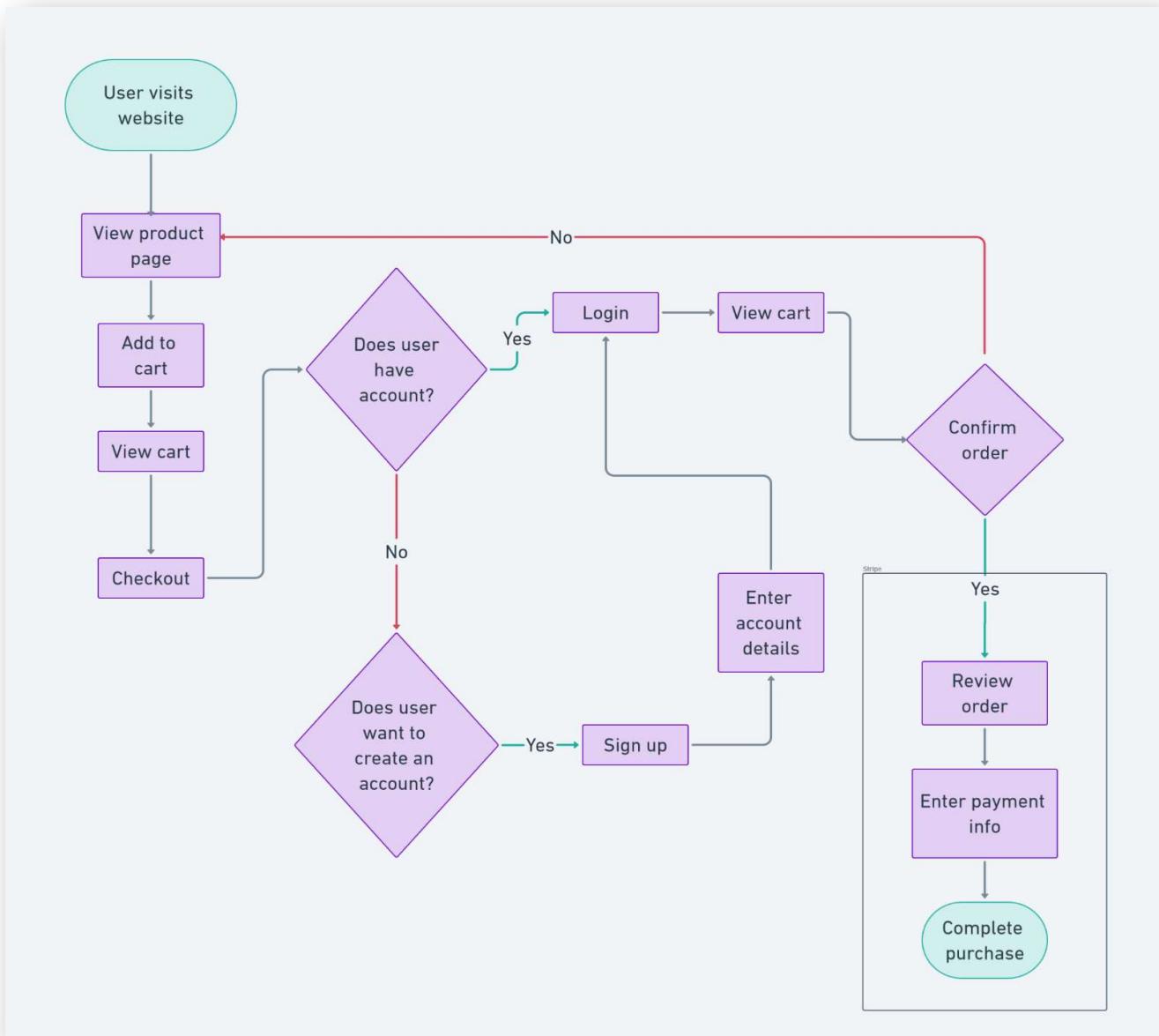


Fig - 3.2 User Flow

3.3 Details of Hardware and Software Requirements

For Software:

1. **React JS:** React.js is a popular JavaScript library developed by Facebook for building user interfaces, emphasizing component-based architecture. It enables developers to create interactive and dynamic web applications with efficient data binding and state management.
2. **Node JS:** Node.js is a runtime environment built on Chrome's V8 JavaScript engine, enabling developers to execute JavaScript code outside of a web browser. It's commonly used for building scalable and high-performance server-side applications.
3. **ExpressJS:** Express.js is a minimalist web application framework for Node.js, providing a robust set of features for building web servers and APIs. It simplifies the process of handling HTTP requests, routing, middleware integration, and template rendering.
4. **MongoDB:** MongoDB is a popular NoSQL database management system, known for its flexibility, scalability, and ease of use. It stores data in a flexible, JSON-like format, making it well-suited for handling unstructured or semi-structured data in modern web applications.
5. **NPM (Node Package Manager):** React relies on Node.js and its package manager, npm, for managing dependencies and running build scripts.

For Hardware:

Operating System:

- Windows: 7 (32/64 bit) or higher, preferably 10 or 11
- macOS: 10.10 or higher
- Linux: Ubuntu 16 or higher

Hardware:

- **Processor:** A modern dual-core or better processor is recommended.
- **RAM:** 4GB of RAM is the minimum, but 8GB or more is ideal for a smoother experience, especially with larger projects.
- **Storage:** At least 10GB of available storage space is needed for React itself and its associated tools and dependencies.

Other Requirements:

- **Web browser:** A modern web browser like Chrome, Firefox, Edge, or Safari is essential for viewing and testing React applications.
- **Code editor:** A code editor or IDE (Integrated Development Environment) is necessary for writing and editing React code. Popular choices include Visual Studio Code, Atom, Sublime Text, and WebStorm.

Recommended Specifications for Optimal Performance:

- **Processor:** Intel Core i5 or equivalent
- **RAM:** 8GB or more
- **Storage:** SSD for faster loading times
- **Display:** Full HD (1920x1080) resolution or higher

Chapter 4

Results and Discussion

Tsuki serves as an online platform for businesses to buy products. It facilitates transactions through secure payment gateways, enabling users to browse, select, and purchase items conveniently. Features like product catalogs, search functionality, customer reviews, and personalized recommendations enhance user experience and drive sales. Effective inventory management ensures accurate product availability information and timely order fulfillment. Integration with logistics and shipping partners streamlines delivery processes, ensuring prompt and reliable product delivery.

Tsuki has revolutionized the way businesses operate, enabling online transactions, global reach, and seamless customer experiences. Its growth is fueled by technological advancements, changing consumer preferences, and the convenience of shopping anytime, anywhere. However, challenges such as security concerns, competition, logistics, and maintaining customer trust require strategic planning and adaptatio

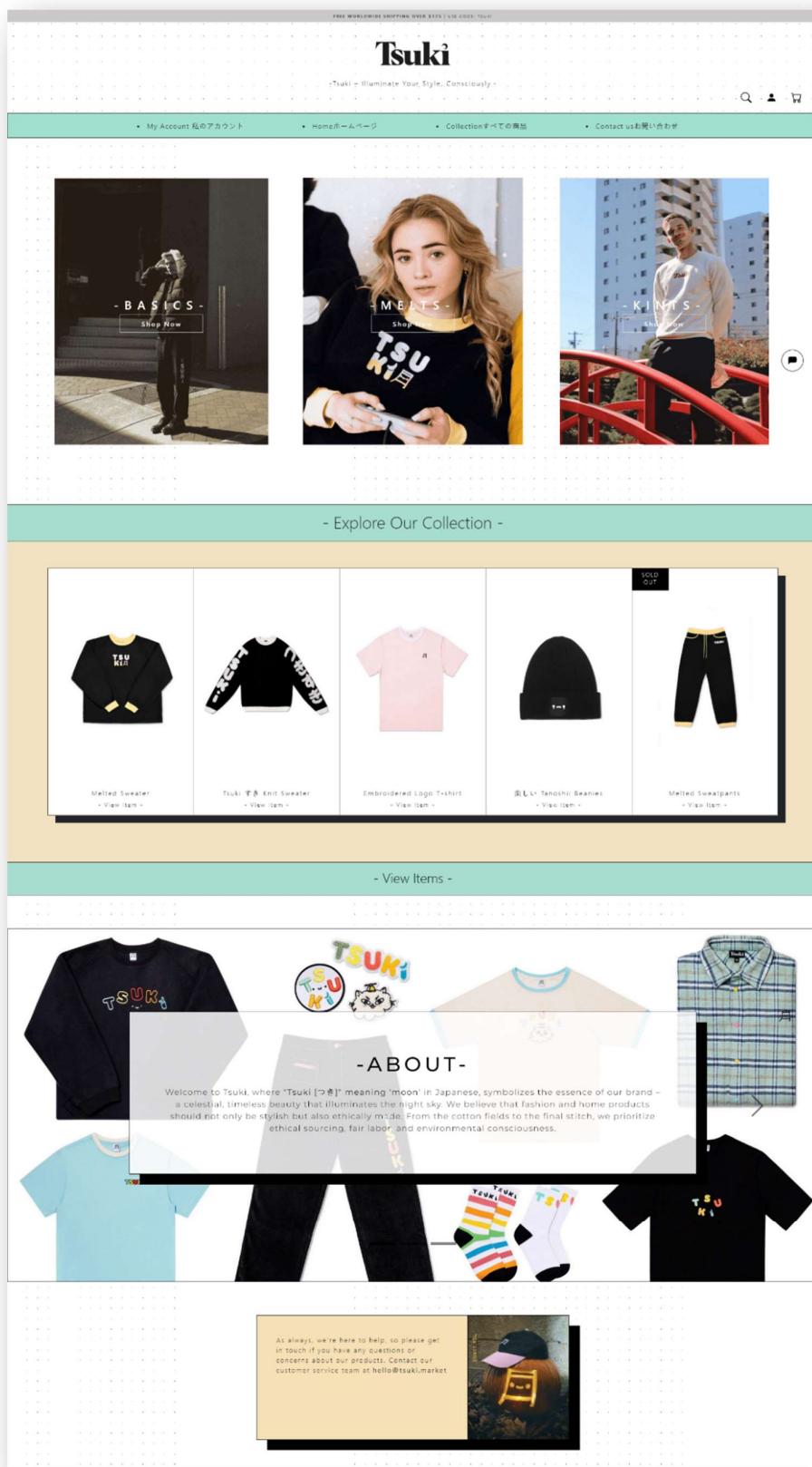


Fig-4.1 Home Page

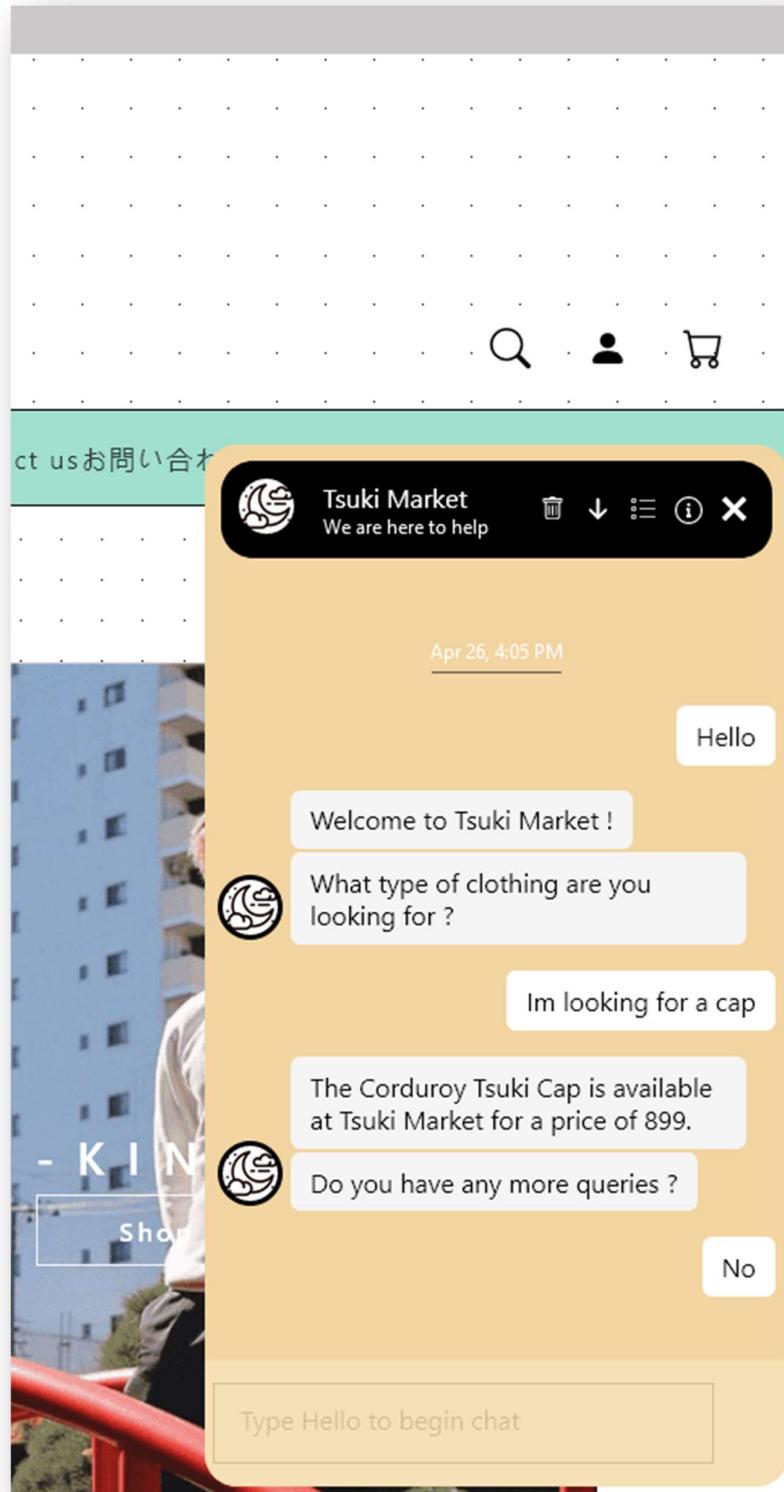


Fig-4.2 AI Chat Bot

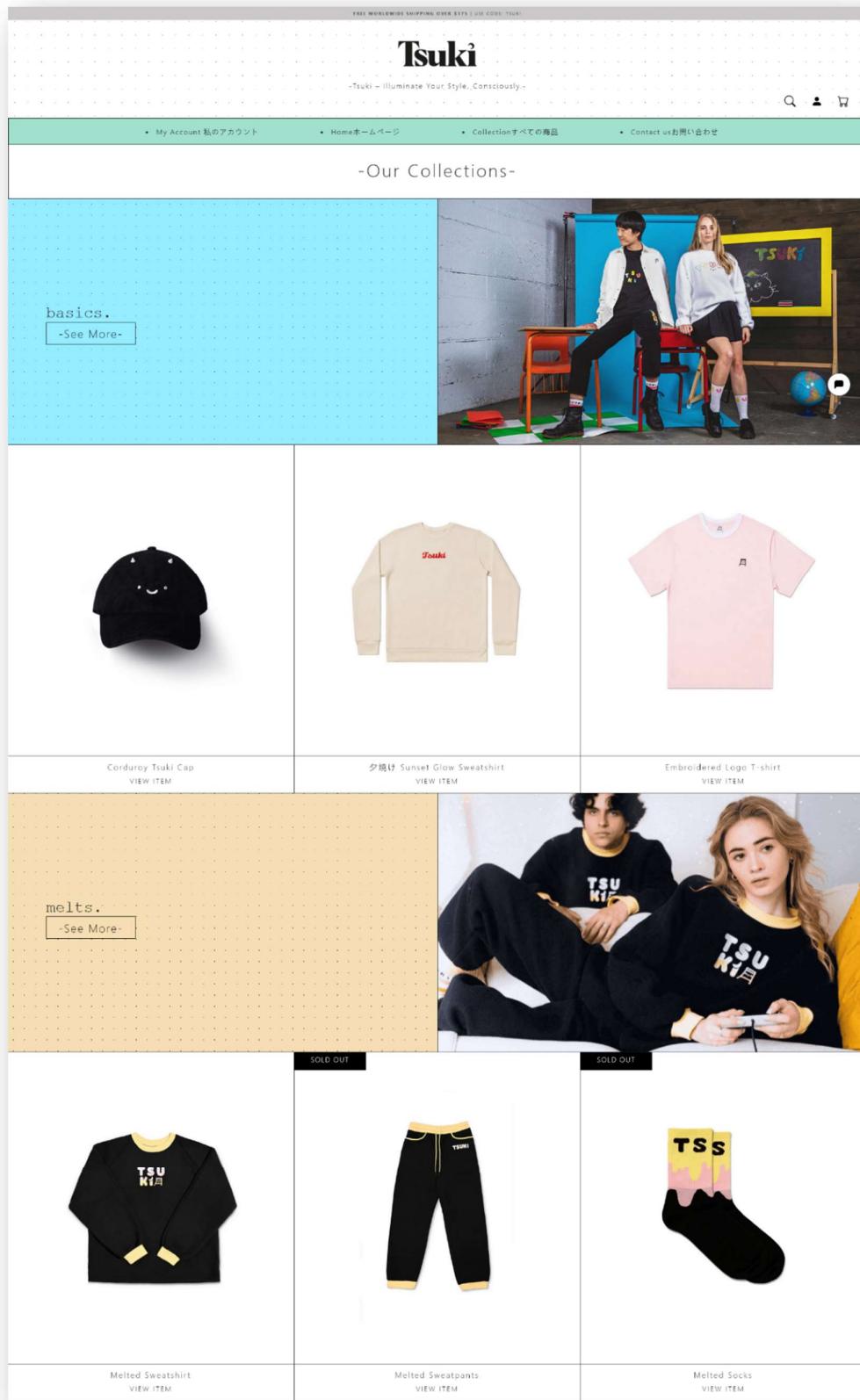


Fig-4.3 Collection Page

FREE WORLDWIDE SHIPPING OVER \$175 | USE CODE: TSUKI

Tsuki

-Tsuki - Illuminate Your Style, Consciously.-

- My Account 私のアカウント
- Homeホームページ
- Collectionすべての商品
- Contact usお問い合わせ

-BASICS-

		
<p>Embroidered Logo T-shirt - View Item -</p>	<p>Corduroy Tsuki Cap - View Item -</p>	<p>Tsuki Socks - View Item -</p>
		
<p>夕焼け Sunset Glow Sweatshirt - View Item -</p>	<p>ストライプ Basic Scarf - View Item -</p>	

- BACK TO COLLECTION -

- HAPPY SHOPPING ハッピー・ショッピング -

NEWSLETTER

Subscribe to be the first to hear about our latest collections, offers and news about the brand.

laldeep@133.com

SOCIALS

[Instagram](#) [Pinterest](#) [Twitter](#) [Facebook](#)

© Tsuki Market 2017 © Made with love by Deepanshu ❤

Fig-4.4 Category Page

FREE WORLDWIDE SHIPPING OVER \$175 | USE CODE: TSUKI

Tsuki

-Tsuki - Illuminate Your Style, Consciously.-

- My Account 私のアカウント
- Home ホームページ
- Collectionすべての商品
- Contact us お問い合わせ

- ITEM DETAILS -



Tsuki すき Knit Sweater

₹11999/-

In Stock

Selected Size:

Select Size ▾

-
1
+

[Add to Cart](#)

This jacquard knit combines everything you could want in a sweater: softness, warmth, and style. Made with fluffy double layered yarn, creating white specks throughout the knit. With TSUKI on one sleeve and すき (general translation = tsuki love) on the other, this jumper is a cosy essential!

DETAILS:

- Embroidery details on front and back, 100% cotton.
- Care Instructions:** Hand wash with soap and water, air dry, do not put in washer/dryer.
- 80% acrylic, 20% nylon; made with extra fluffy yarn.
- Wash at 30 degrees celsius, cool iron, dry flat, do not bleach or dry clean.
- Made in the UK as a fully fashioned flat knit, producing a nearly waste free product.

- Explore similar items -



SOLD OUT

Melted Sweater
[View Item](#)



Melted Sweatpants
[View Item](#)



楽しい Tanoshi Beanie
[View Item](#)



Embroidered Logo T-shirt
[View Item](#)



Tsuki すき Knit Sweater
[View Item](#)

- Back to Collection -

Fig-4.5 Product page

- ITEM DETAILS -

Embroidered Logo T-shirt

₹1899/-

In Stock

Selected Size:

Your new favourite t-shirt. Super soft and versatile, this tee is available in three classic Tsuki colour combinations. Marzia is wearing a size small and Felix is wearing a size medium.

DETAILS:

- Embroidery details on front and back, 100% cotton.
- **Care Instructions:** Hand wash with soap and water, air dry, do not put in washer/dryer.
- 80% acrylic, 20% nylon; made with extra fluffy yarn.
- Wash at 30 degrees celsius, cool iron, dry flat, do not bleach or dry clean.
- Made in the UK as a fully fashioned flat knit, producing a nearly waste free product.

Fig- 4.6 Product Page

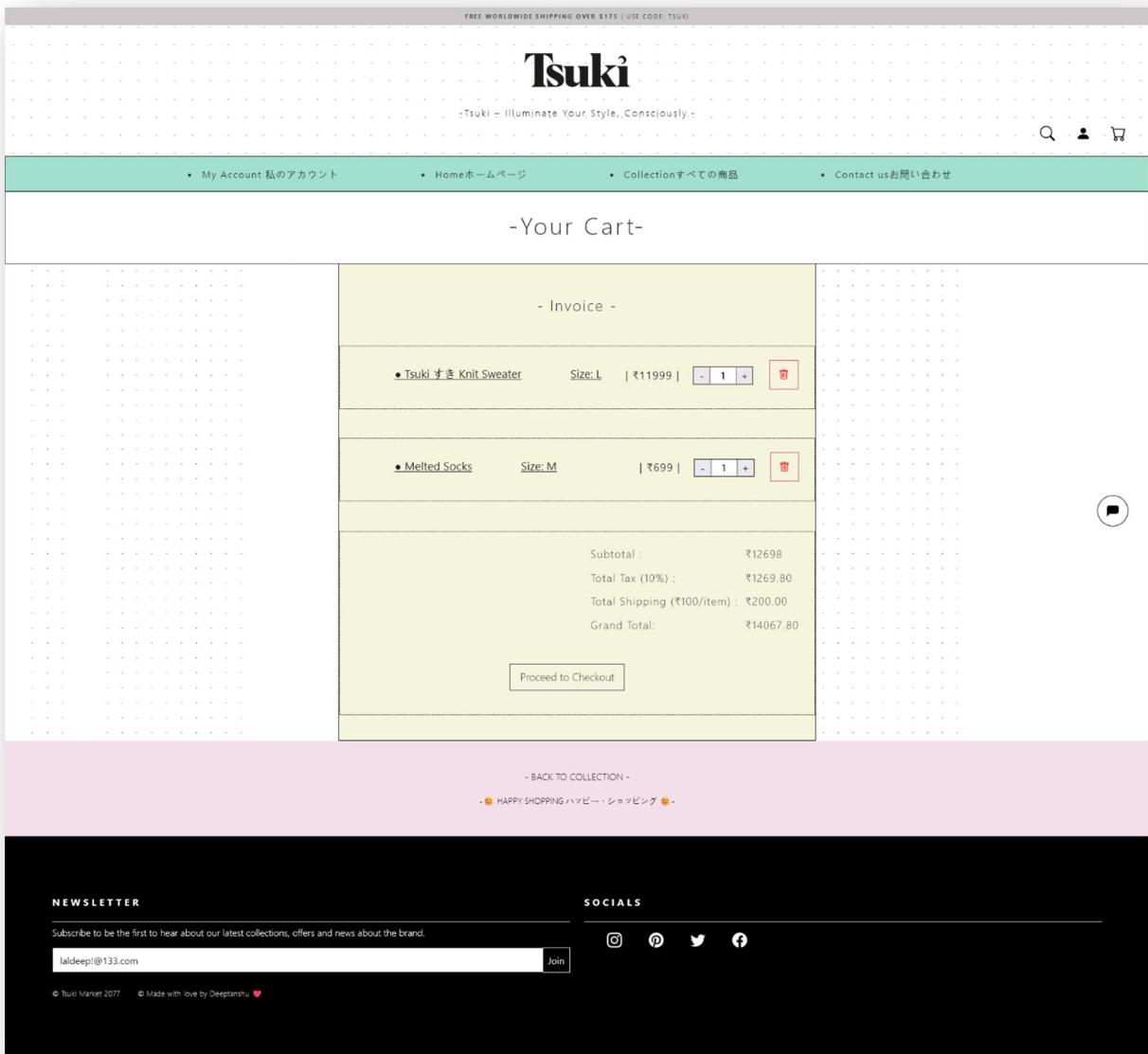


Fig-4.7 Cart page

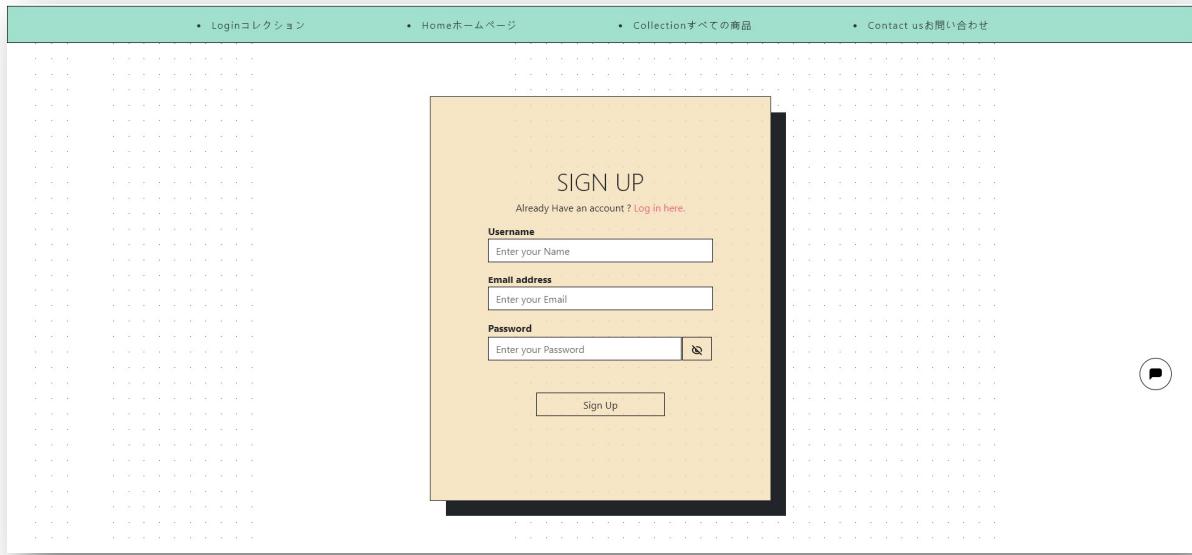
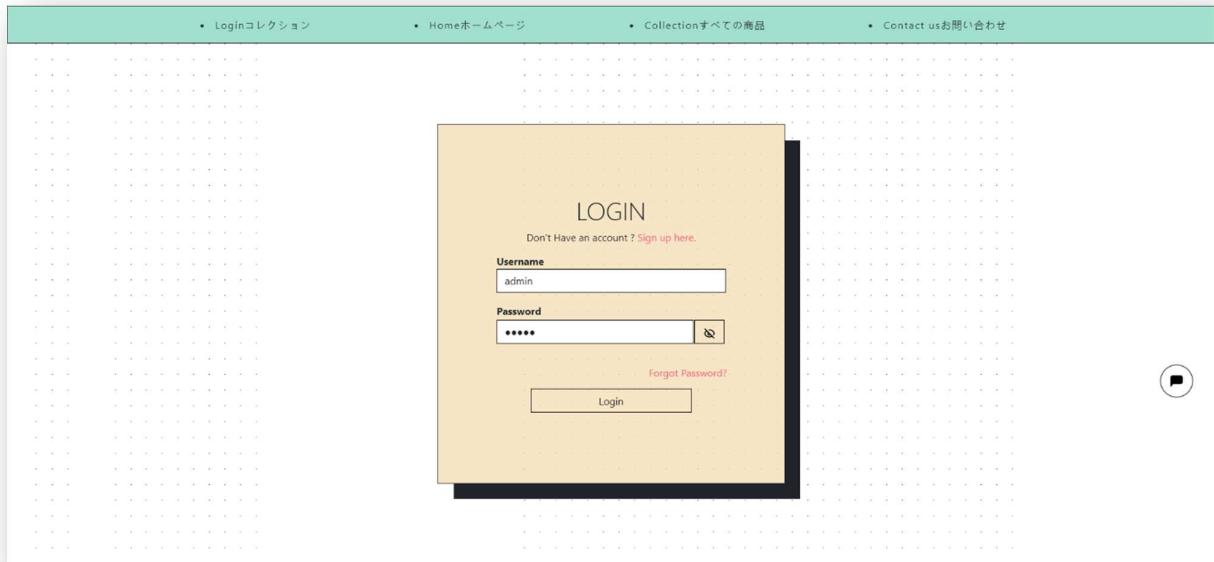


Fig-4.8/4.9 Login /Sign -Up Page

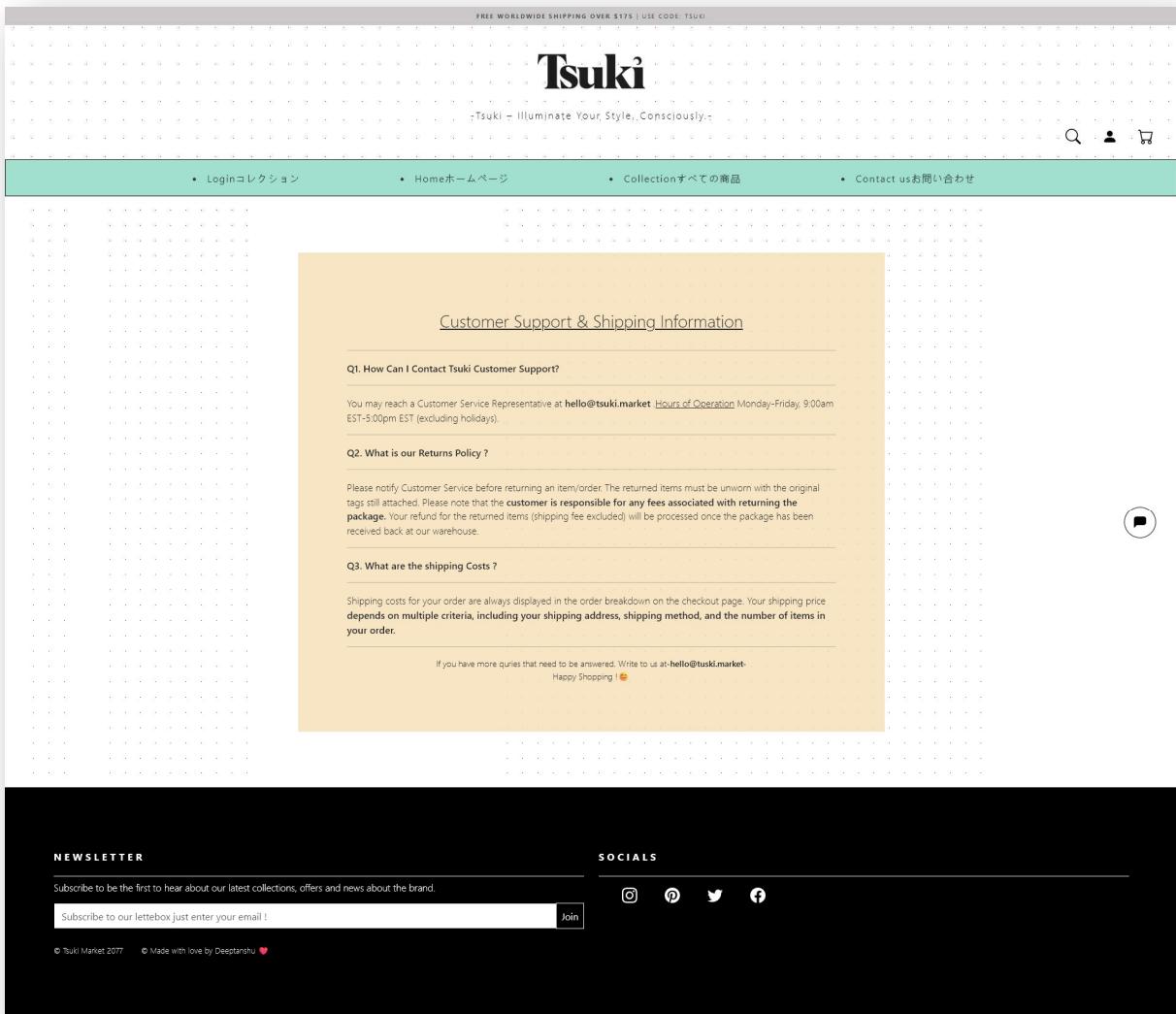


Fig-4.10 Contact page

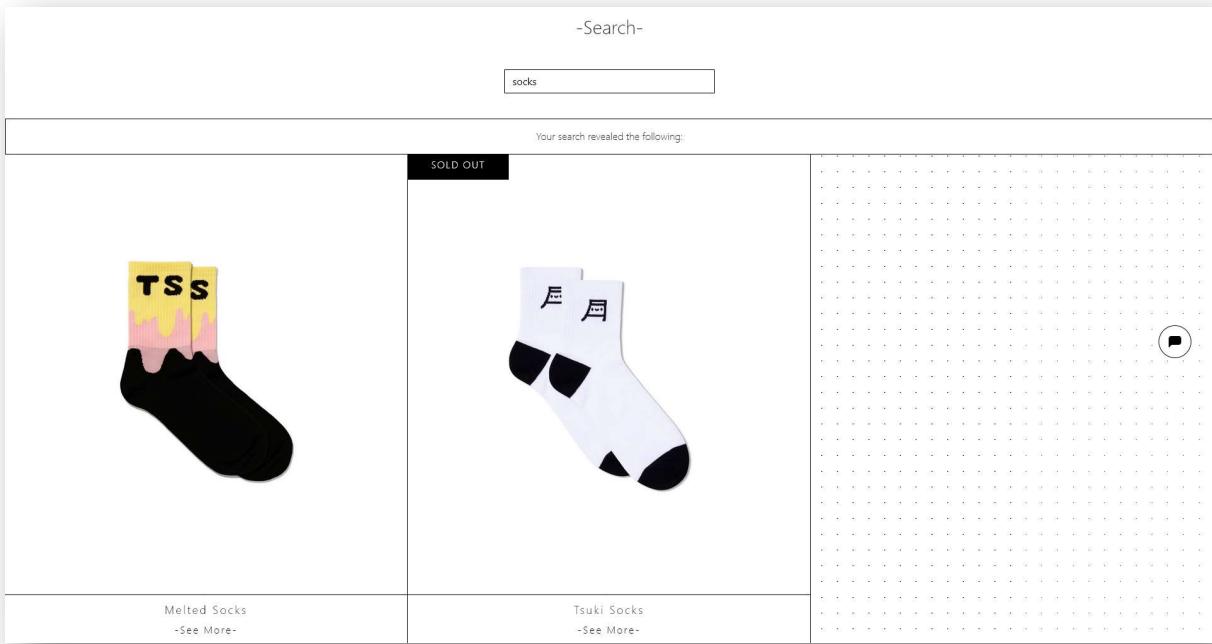


Fig-4.11 Search Page

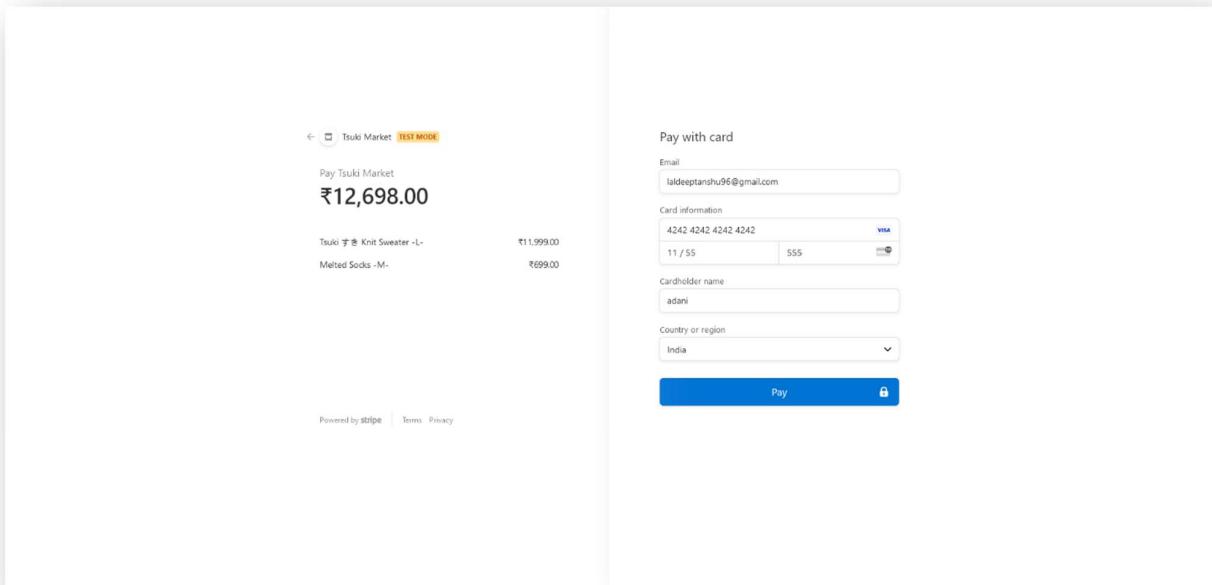


Fig-4.12 Payment Page (Stripe)



Fig-4.13 Successful Transaction Page



Fig – 4.14 Cancelled Transaction Page

FREE WORLDWIDE SHIPPING OVER \$175 | USE CODE: TSUKI

Tsuki

-Tsuki - Illuminate Your Style, Consciously.-

My Account 私のアカウント Homeホームページ Collectionすべての商品 Contact usお問い合わせ

- Order History -

RECEIPT:

Order ID: 662b5b4af9af73f9989155da
Order Placed on: 3/21/2024, 1:14:10 AM
Order Placed by: admin
Order Status: Paid

Items:

Corduroy Tsuki Cap -XL- - | x1 | - Price: 899
Corduroy Tsuki Cap -M- - | x1 | - Price: 899

Total: ₹1798

[View Invoice](#)

RECEIPT:

Order ID: 662bb564eb3343744b0940dd
Order Placed on: 4/26/2024, 4:15:56 PM
Order Placed by: admin
Order Status: Paid

Items:

Tsuki すき Knit Sweater -L- - | x1 | - Price: 11999
Melted Socks -M- - | x1 | - Price: 699

Total: ₹12698

[View Invoice](#)

RECEIPT:

Order ID: 662b8619eb3343744b0940de
Order Placed on: 4/26/2024, 4:16:49 PM
Order Placed by: admin
Order Status: Paid

Items:

Tsuki すき Knit Sweater -L- - | x1 | - Price: 11999
Melted Socks -M- - | x1 | - Price: 699

Total: ₹12698

[View Invoice](#)

Fig – 4.15 Order History

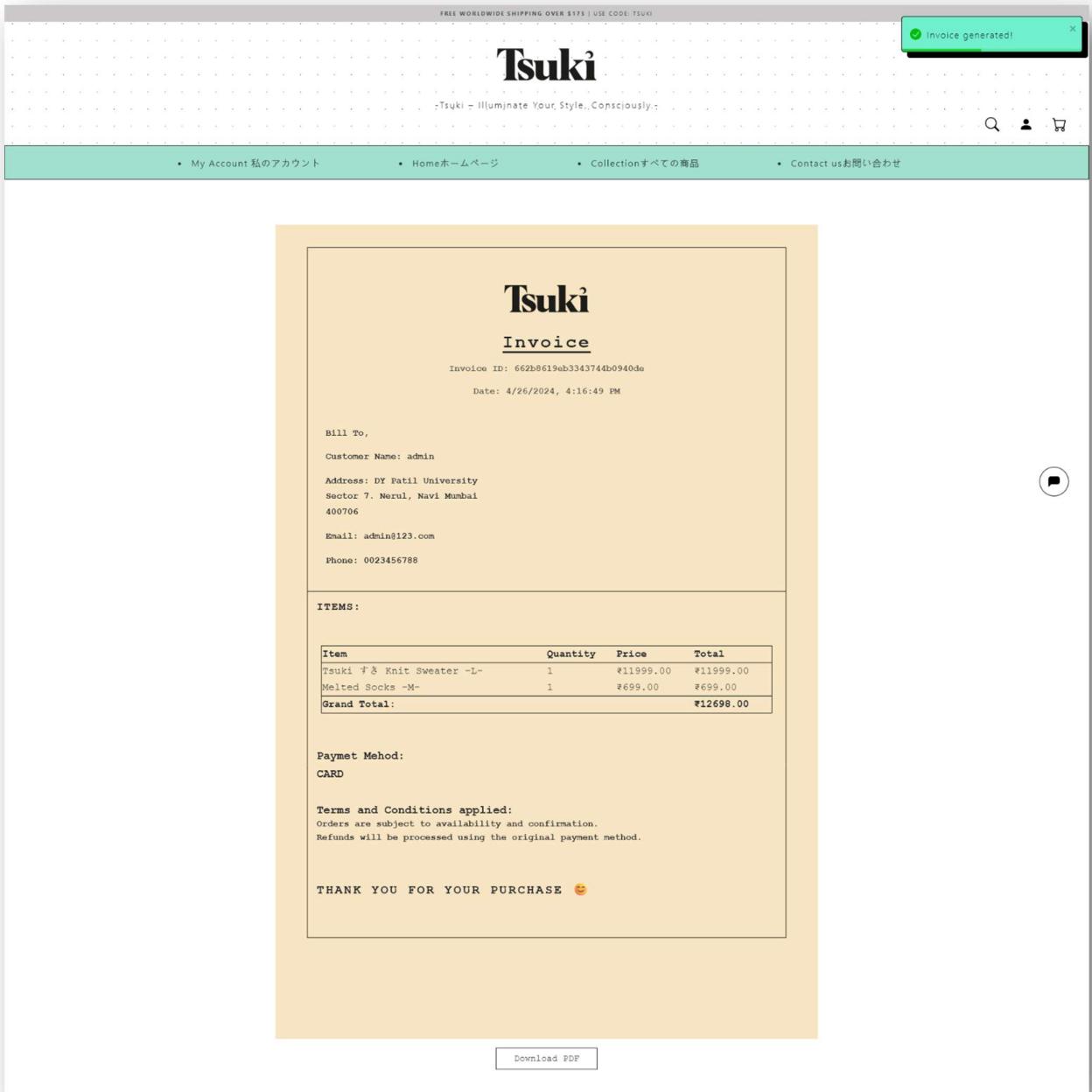


Fig – 4.16 Invoice Page

The screenshot shows the GitHub repository page for 'laughing-computer'. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings. Below the navigation is a search bar and a 'Code' dropdown menu. The main content area displays a list of files and their details:

File	Description	Last Commit
Deeptanshuu Create README.md	broooo	1 hour ago
Client	size selector backend logic	4 hours ago
Doc	stripe checkout metadata	3 hours ago
Server	Create README.md	6 minutes ago
README.md		

Below the file list is a 'README' section with a 'Project Setup' checklist:

- Organize project structure: Create separate directories for frontend and backend.
- Create backend models for products and users (if authentication is implemented).
- Implement backend routes for product listing, user authentication, cart management, and checkout.
- Add middleware for handling CORS, parsing JSON, etc.
- Implement CRUD operations for products and users.
- Secure the application using authentication middleware (optional).

Under 'Frontend Development':

- Create React components for product listings, shopping cart, authentication, search, and filters.
- Implement React Router for navigation between pages.

On the right side, there are sections for About, Releases, Packages, Deployments, and Languages. The Languages section shows a bar chart with the following data:

Language	Percentage
JavaScript	75.4%
HTML	23.2%
CSS	2.0%
Dockerfile	0.1%

Fig-4.17 GitHub page

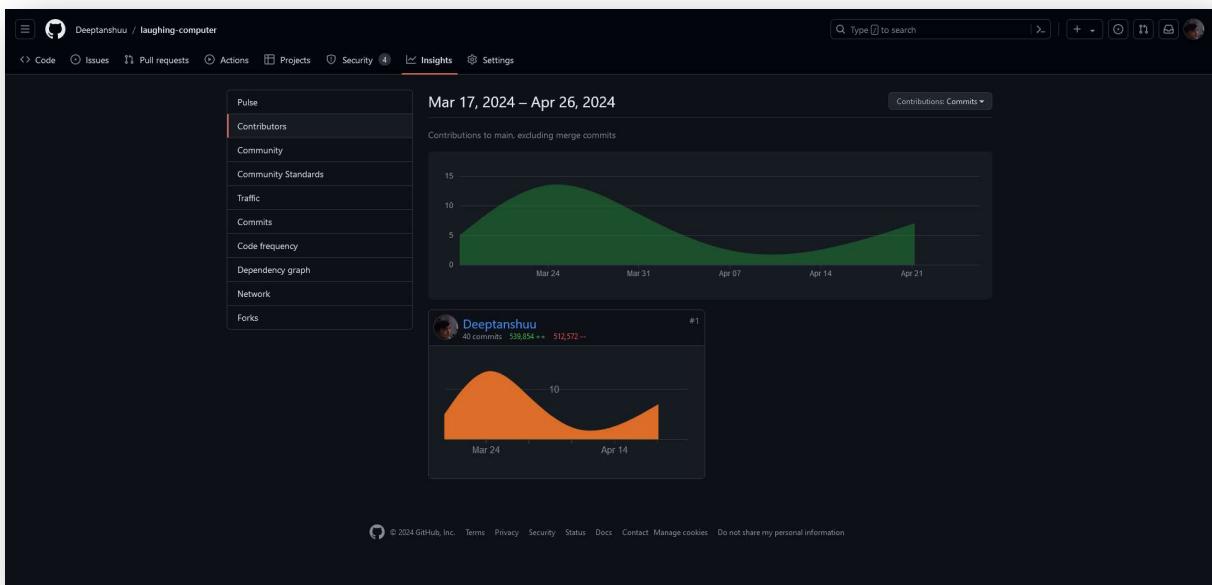


Fig-4.18 GitHub Contribution page

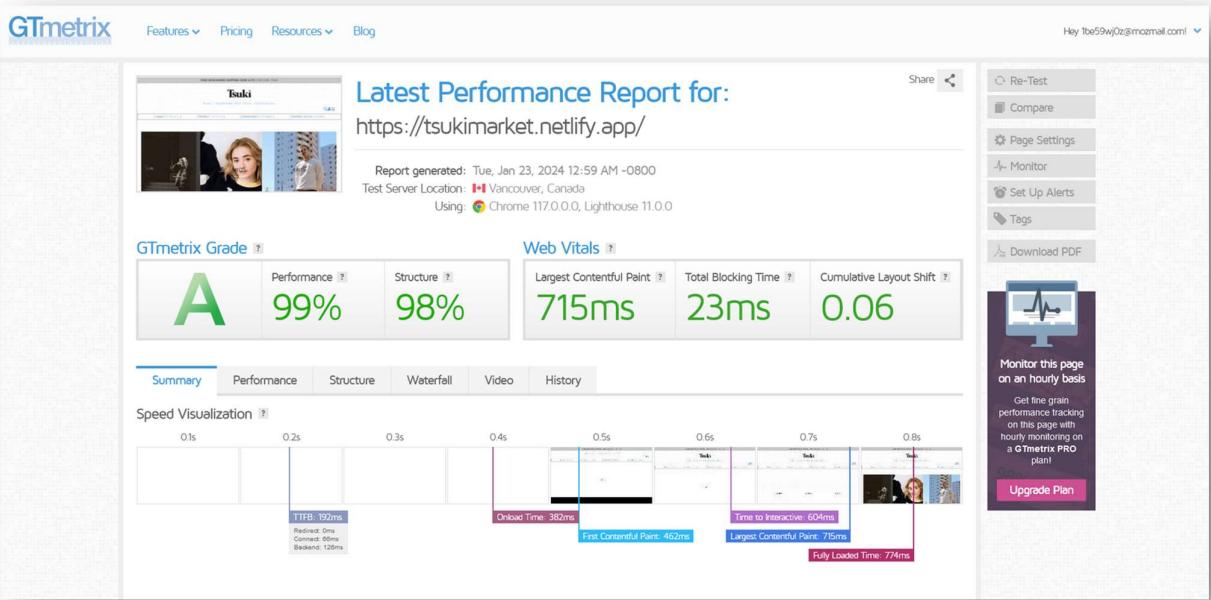


Fig-4.19 GTmetrix page
(indication of performance of the website)

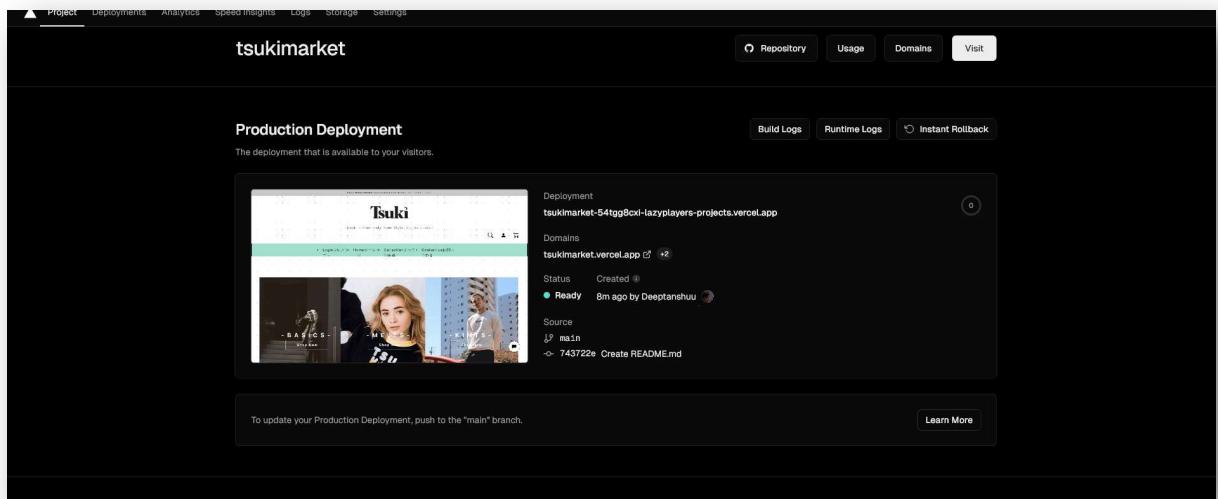


Fig- 4.20 Vercel Deployment Page
(Platform used to deploy the website)

Chapter 5

Conclusion and Future work

In conclusion, the development of our e-commerce website project has provided invaluable insights into the integration of modern web technologies and methodologies to create a dynamic and user-centric platform. By leveraging tools such as React.js, Node.js, Express.js, and MongoDB, we have constructed a robust system capable of delivering a seamless shopping experience to users.

Through the implementation of responsive design principles, we have ensured that our e-commerce website is accessible across a variety of devices and screen sizes, catering to the diverse needs of our user base. This adaptability enhances user engagement and satisfaction, contributing to a positive overall shopping experience.

Furthermore, we can consider integrating additional features and functionalities into our e-commerce catalog, such as user authentication and authorization, product recommendations based on user behavior, and social media integration for enhanced sharing and engagement. These enhancements would enrich the overall user experience and contribute to the continued growth and success of our platform.

References

- I. Kirti Bhandge, Tejas Shinde, Dheeraj Ingale, Neeraj Solanki, and Reshma Totare. "A Proposed System for Touchpad Based Food Ordering System Using Android Application." International Journal of Advanced Research in Computer Science Technology (IJARCST), 2015.
- II. Varsha Chavan, Priya Jadhav, Snehal Korade, and Priyanka Teli. "Implementing Customizable Online Food Ordering System Using Web Based Application." International Journal of Innovative Science Engineering Technology (IJISET), 2015.
- III. Resham Shinde, Priyanka Thakare, Neha Dhomne, and Sushmita Sarkar. "Design and Implementation of Digital dining in Restaurants using Android." International Journal of Advance Research in Computer Science and Management Studies, 2014.
- IV. Ashutosh Bhargave, Niranjan Jadhav, Apurva Joshi, Prachi Oke, and S. R Lahane. "Digital Ordering System for Restaurant Using Android." International Journal of Scientific and Research Publications, 2013.
- V. K. Khairunnisa, J. Ayob, Mohd. Helmy A. Wahab, M. Erdi Ayob, M. Izwan Ayob, and M. Afif Ayob. "The Application of Wireless Food Ordering System." MASAUM Journal of Computing, 2009.
- VI. Patel Krishna, Patel Palak, Raj Nirali, and Patel Lalit. "Automated Food Ordering System." International Journal of Engineering Research and Development (IJERD), 2015.
- VII. Mayur D. Jakhete and Piyush C. Mankar. "Implementation of Smart Restaurant with e-menu Card." International Journal of Computer Applications, 2015.
- VIII. A. Bharadi. "Intelligent e-Restaurant using Android OS Mumbai." 2013.
- IX. M. Rajesh. "E-Restaurant: Online Restaurant Management." International Journal & Magazine of Engineering, 2015.
- X. T. Raibagi, A. Vishwakarma, J. Naik, R. Chaudhari, and G. Kalme. "Orderista - AI-based Food Ordering Application." 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), pp. 34-37, 2021.
- XI. "<https://www.mongodb.com>". [Online]. Available.

ACKNOWLEDGEMENT

We take this opportunity to express my profound gratitude and deep regards to my guide **Dr. Shital Patil** for her exemplary guidance, monitoring and constant encouragement throughout the completion of this report. We are truly grateful to his/her efforts to improve my understanding towards various concepts and technical skills required in our project. The blessing, help and guidance given by her time to time shall carry us a long way in the journey of life on which we are about to embark.

We take this privilege to express my sincere thanks to **Dr. Mukesh D. Patil Principal, RAIT, D. Y. Patil deemed to be University** for providing the much necessary facilities. We are also thankful to **Dr. A. V. Vidhate**, Head of Department of Computer Engineering, **Mrs. Siuli Das**, Mini Project Co-ordinator, for their generous support.

Last but not the least we would also like to thank all those who have directly or indirectly helped us in completion of this project report.

Deeptanshu Lal

Krish Pradeshi



D Y PATIL UNIVERSITY
RAMRAO ADIK INSTITUTE OF TECHNOLOGY, NAVI MUMBAI

SE Mini-Project-Weekly Project Performance Report Even Sem 2023-2024
Department of Computer Engineering

Project Title: _____

Group No: _____

Name of Students 1: Name of Students 2: Name of Students 3: Name of Students 4:

Week No.	Topics to be Covered	Progress Status	Student 1 Sign	Progress Status	Student 2 Sign	Progress Status	Student 3 Sign	Progress Status	Student 4 Sign	Suggestions if any
1.	Proposed Methodology									
2.	System Design									
3.	Design of proposed system									
4.	Implementation Details Module wise									
5.	Implementation Details Module wise									
6.	Full Implementation Details									
7.	Result Analysis									
8.	Report Writing									
9.	Report Writing									
10.	Conclusion / Future Work									

A: Satisfactory B: Average C: Needs Improvement

Project Guide Name and Sign