FINAL YEAR PROJECT REPORT

MERN STACK WEB DEVELOPMENT PROJECT ON E-COMMERCE WEBSITE "MULTIMEDIA MART"

BY

MD. ASIF KARIM
ID: 191-40-580

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Multimedia and Creative Technology

Supervised By

Apurba Ghosh
Assistant Professor
Department of MCT
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

AUGUST 2023

APPROVAL

This Project titled **"MERN STACK WEB DEVELOPMENT PROJECT ON E-COMMERCE WEBSITE 'MULTIMEDIA MART'"**, submitted by **MD. Asif Karim** to the Department of Multimedia and Creative Technology, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Multimedia and Creative Technology and approved as to its style and contents. The presentation has been held on **19th August, 2023**.

BOARD OF EXAMINERS

Md. Salah Uddin **Chairman**
Assistant Professor & Head
Department of Multimedia & Creative Technology
Faculty of Science & Information Technology
Daffodil International University

Dr. Shaikh Muhammad Allayer **Internal Examiner**
Professor
Department of Multimedia & Creative Technology
Faculty of Science & Information Technology
Daffodil International University

Arif Ahmed **Internal Examiner**
Associate Professor
Department of Multimedia & Creative Technology
Faculty of Science & Information Technology
Daffodil International University

Dr. Mohammad Zahidur Rahman **External Examiner**
Professor
Department of Computer Science & Engineering
Jahangirnagar University

DECLARATION

I hereby declare that, this project has been done by me under the supervision of **Apurba Ghosh, Assistant Professor, Department of MCT** Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Apurba Ghosh
Assistant Professor
Department of MCT
Daffodil International University

Submitted by:

Asif

MD. Asif Karim
ID: 191-40-580
Department of MCT
Daffodil International University

ACKNOWLEDGEMENT

First, I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project successfully.

I am really grateful and wish our profound our indebtedness to **Apurba Ghosh**, Assistant Professor, Department of MCT Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “Web Development” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

I would like to thank **Md Salah Uddin**, **Dr. Shaikh Muhammad Allayear**, **Mr. Arif Ahmed**, **Mr. Md. Samaun Hasan** and my entire course mates in Daffodil International University, who took part in this discussion while completing the course work.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

ABSTRACT

Multimedia Mart is a dynamic and user-friendly MERN (MongoDB, Express.js, React.js, Node.js) stack website designed to cater to the needs of students and individuals seeking multimedia related products. The objective of this project is to provide a convenient platform for users to purchase and sell a wide range of multimedia products.

React.js was used for the frontend, Node.js was used for the backend server-side logic, Express.js was used for the creation of the API, and MongoDB was used for storage throughout the development process, hence the MERN stack.

The "Multimedia Mart" project's overarching goal is to supply students with a convenient and thorough online storefront where they can browse and buy a wide variety of products. The website offers a wide variety of products, meeting the needs of a large audience, thanks to its support for both independent vendors and traditional stores.

This defense project will explain the goals, objectives, features, implementation & the work process of the website called “Multimedia Mart”.

TABLE OF CONTENTS

CONTENTS	PAGE
Acknowledgement	iii
Abstract	iv
Table of Contents	v-vi
List of Figures	vii-viii
CHAPTER	
CHAPTER 1: INTRODUCTION	**1-2**
1.1 Introduction	1
1.2 Objective	2
1.3 Inspiration	2
CHAPTER 2: RESEARCH AND ANALYSIS	**3-13**
2.1 Understanding MERN Stack	3
2.2 How MERN stack works?	7
2.3 Multimedia E-commerce Landscape	8
2.4 Related Websites	11
CHAPTER 3: SOFTWARES USED TO IMPLEMENT MY PROJECT	**14-19**
3.1 Visual Studio Code	14
3.2 Node.js	15

3.3 Express.js 15

3.4 MongoDB 16

3.5 React 16

3.6 Figma 17

3.7 GitHub 18

3.8 Chrome Devtools 18

3.9 Dependencies 19

CHAPTER 4: DEVELOPMENT AND WORKFLOW 20-46

4.1 Introduction 20

4.2 Key Features and Functionality 20

4.3 Workflow 34

CHAPTER 5: FINDINGS & OBERSERVATIONS 47-48

CHAPTER 6: CONCLUSION 49-50

REFERENCES 51-52

LIST OF FIGURES

FIGURES	PAGE NO
Figure 2.1.1: MongoDB Structure	3
Figure 2.1.3: ReactJs Project Structure	5
Figure 2.1.4: NodeJS Architecture	6
Figure 2.2: Architecture of MERN stack	7
Figure 2.3.1: Projected E-Commerce Growth	8
Figure 2.3.4: “Multimedia Mart” Product Categories	10
Figure 2.4.1: BestBuy Website	11
Figure 2.4.2: eBay Website	12
Figure 2.4.3: Star Tech Website	13
Figure 3.1: Visual Studio Code Interface	14
Figure 3.2: Node.js Logo	15
Figure 3.3: Express.js code	15
Figure 3.4: MongoDB Logo	16
Figure 3.5: React Logo	17
Figure 3.6: Figma Logo	17
Figure 3.7: GitHub Logo	18
Figure 4.2.1.a: Register Page	21
Figure 4.2.1.b: Login Page	21
Figure 4.2.1.c: Verification email	22
Figure 4.2.2.a: Create Product	22
Figure 4.2.2.b: All Orders	23
Figure 4.2.2.c: Seller Profile	23
Figure 4.2.3: Responsive Design	24
Figure 4.2.4.a: Best Deals	25
Figure 4.2.4.b: Featured Products	25
Figure 4.2.5.a: User Dashboard	26
Figure 4.2.5.b: Seller Dashboard	27

Figure 4.2.5.c: Admin Dashboard 27
Figure 4.2.6: Search Bar 28
Figure 4.2.7: Wishlist & Cart 29
Figure 4.2.8.a: All Messages 30
Figure 4.2.8.b: Live Conversation 30
Figure 4.2.9.a: Product Rating 31
Figure 4.2.9.b: Product Reviews 31
Figure 4.2.10.a: Shipping Option 32
Figure 4.2.10.b: Payment Option 33
Figure 4.2.10.c: Order Confirmation 33
Figure 4.3.1: Roles Interaction 34
Figure 4.3.2: Website Wireframes 35
Figure 4.3.3: MERN Stack 36
Figure 4.3.4.a: CSS Styles 37
Figure 4.3.4.b: React Components Setup 38
Figure 4.3.4.c: React Component 38
Figure 4.3.4.d: Routes 39
Figure 4.3.4.e: State Management 40
Figure 4.3.4.f: Integration of API 41
Figure 4.3.5.a: Server Setup 42
Figure 4.3.5.b: Connecting MongoDB 43
Figure 4.3.5.c: API Endpoints Creation 44
Figure 4.3.5.d: Using Multer to Upload 44
Figure 4.3.5.e: User Authentication and Authorization 45
Figure 4.3.5.f: Setting Up Socket.io 46

CHAPTER 1
INTRODUCTION

1.1 Introduction

In today's world, e-commerce websites have emerged as a vital part of growing companies, and web development has emerged as an essential step in the process of developing these websites. Web development is the process of designing and developing websites, as well as optimizing their functionality and performance to provide users with an enjoyable experience and protect financial transactions [1].

I developed “Multimedia Mart” as a very user-friendly website that will provide a seamless experience for both buyers and sellers. To ensure secure access, login and signup pages are implemented for both users and sellers. Account activation is a mandatory step, requiring users to click on an activation link sent to their email addresses, thereby verifying their accounts and ensuring a reliable user base. The website incorporates strategic sections such as the "Best Selling" page, showcasing the most popular products, and the "Featured Products" section, which highlights the latest additions to the marketplace. These sections serve as valuable tools for users to explore trending items and stay updated with the latest offerings.

To cater to the specific needs of sellers and users, dedicated dashboards are provided, offering a personalized and efficient management system. Additionally, an admin role is implemented to oversee and maintain the smooth functioning of the website, ensuring proper moderation and resolving any potential issues that may arise. A critical aspect of any e-commerce platform is the payment options it offers. "Multimedia Mart" supports two payment methods: credit card payments and cash on delivery. This flexibility provides users with convenient and secure payment choices during the checkout process.

1.2 Objective

My main goal in creating the "Multimedia Mart" website is to serve as a platform for selling multimedia-related products, with a specific focus on satisfying the needs of students. Aside from providing as a platform for selling multimedia products, another primary goal of developing the "Multimedia Mart" website is to demonstrate my talents and expertise in web development. By creating this website from the ground up, I hope to demonstrate my skills to design and develop a fully functional and user-friendly website that can meet the needs of a certain target audience.

1.3 Inspiration

My inspiration for taking on this project began when our beloved Head of Department, Md. Salah Uddin sir, enrolled in a Web Engineering course. As a result, my interest in web building grew, as did my ambition to demonstrate my abilities in this field. The idea of creating a website from scratch up, merging numerous technologies, and executing sophisticated functionality piqued my interest. I wanted to show off my web developing abilities.

The concept for "Multimedia Mart" evolved from my love of technology, understanding of student needs, and enthusiasm for web development. This project provided an excellent opportunity to apply my creativity, problem-solving skills, and technical knowledge to a real-world application that would assist me in the future.

CHAPTER 2

RESEARCH AND ANALYSIS

2.1 Understanding MERN Stack

MERN stack is essential in 2023 and in high demand because of its wide application in the web development community. It is used for its ease and dynamic nature [2]. In all addition, the MERN stack is an all-inclusive solution that includes front-end as well as back-end development. The MERN stack includes of four crucial technologies that are developed on top of JavaScript. This is how they look:

2.1.1 MongoDB (Database)

If an app stores any kind of data, like a social platform that keeps track of user accounts, content, comments, likes, and other similar information, it needs to use a database. There are a lot of different databases to choose from, but MongoDB has been chosen as the best database for the MERN stack [3]. MongoDB is a NoSQL database that gives web application developers the ability to handle the databases of their online apps. Using the MERN stack, we can write documents that are made up of key pairs, like JSON [4].

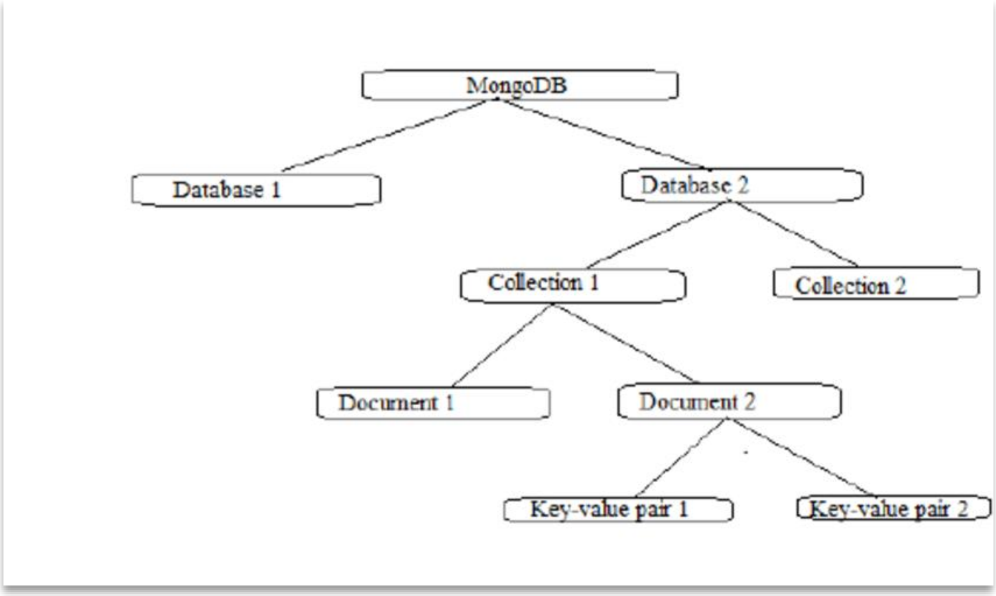


Figure 2.1.1: MongoDB Structure

There are many advantages of using MongoDB as database, they are –

- This framework can be built up efficiently, and it can handle document models.
- MongoDB offers a huge and active community as well as a good ecosystem of tools, libraries, and resources.
- MongoDB has very good flexible schema that makes it possible to manage dynamic data structures.
- MongoDB provides support for a wide variety of coding languages and platforms, which leads to an increase in the database's adaptability and the number of ways it may be integrated.
- MongoDB has built-in support for cloud environments, which makes it easy handle databases in cloud platforms like Amazon Web Services [5], Microsoft Azure, and Google Cloud Platform. [6].

2.1.2 ExpressJS (Backend)

Express.js is a widely used web application framework for Node.js. It helps to create web applications and APIs easier by being simple, flexible, versatile, and having a number of other features [7]. In addition, the ExpressJS framework offers support for a wide variety of middlewares, which helps keep code clean and makes it less complicated to build. Advantages of using ExpressJS are –

- **Simplicity**: Express.js gives enough structure to construct web apps without restricting code organization. This makes it appropriate for small applications to large-scale systems.
- **Routing**: Express's flexible routing system lets users specify the behavior for various routes. Different HTTP methods (GET, POST, PUT, DELETE, etc.) can each have their own defined routes.
- **Performance**: Its lightweight design allows it to handle requests more quickly than other frameworks.

2.1.3 React (Frontend)

React is a JavaScript library that can be used to make dynamic frontend web apps. The library lets JavaScript writers build complex interfaces with simple parts that connect the application to the backend server and show them as HTML. The main benefit of using React is being able to handle stateful, data-driven applications with less code [8].

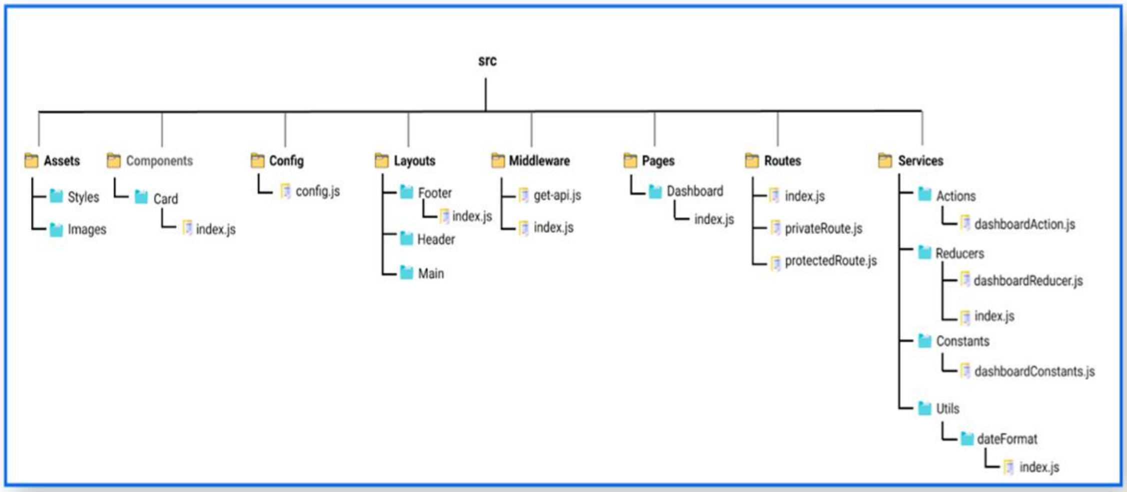


Figure 2.1.3: ReactJs Project Structure

React has had a phenomenon impact on web developing landscape and become an important part in modern web development. Here are some advantages of using react-

- **Better speed**: React.js uses a virtual DOM to improve performance, which means that complex UI components load faster and the user experience is smoother.
- **Server-side rendering:** Support for server-side rendering in React.js boosts search engine optimization and decreases user wait times.
- **State Management:** Redux is a state management library for React, which focuses on the UI layer. These libraries simplify application state management and data flow.
- **Huge User-Base:** Many large websites and services, including as Facebook, Instagram, Airbnb, and more, use React. Its scalability and dependability are demonstrated by the fact that it is used by popular applications and websites.

2.1.4 NodeJS (Runtime Environment)

Node.js is an effective runtime environment for running JavaScript applications outside of a web browser. It brings the JavaScript language to the server-side, which lets developers build flexible, high-performance, event-based apps [9]. In summary, we require Node.js to use Express.js, which communicates with our front-end.

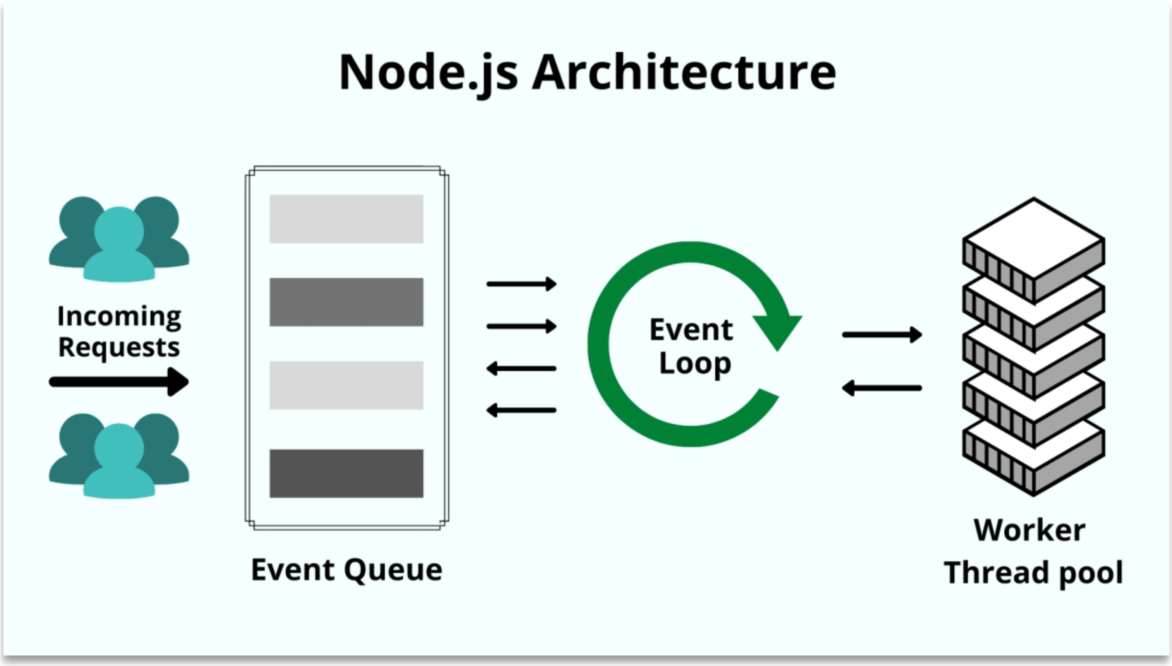


Figure 2.1.4: NodeJS Architecture

The advantages of using NodeJS are –

- **Versatility:** Node.js is a lightweight and scalable runtime environment for developing JavaScript-based server-side applications.
- **Fast Execution:** Because it is based on Chrome's engine, the process of executing code in it is quite quick.
- **NPM (Node Package Manager**): The powerful package manager that comes with Node.js is called NPM. It has a large collection of open-source tools and modules that are easy to add to apps.

- **Real-Time Applications:** Node.js is great for making real-time apps that need clients and servers to talk to each other right away. With platforms like Socket.io, it's possible to build real-time features like chat apps, live messages, and tools for working together

2.2 How MERN stack works?

The MERN stack's structure is outlined in the diagram below. We can see that it is built from a front end, a back end and a database. This is known as a complete stack architecture.

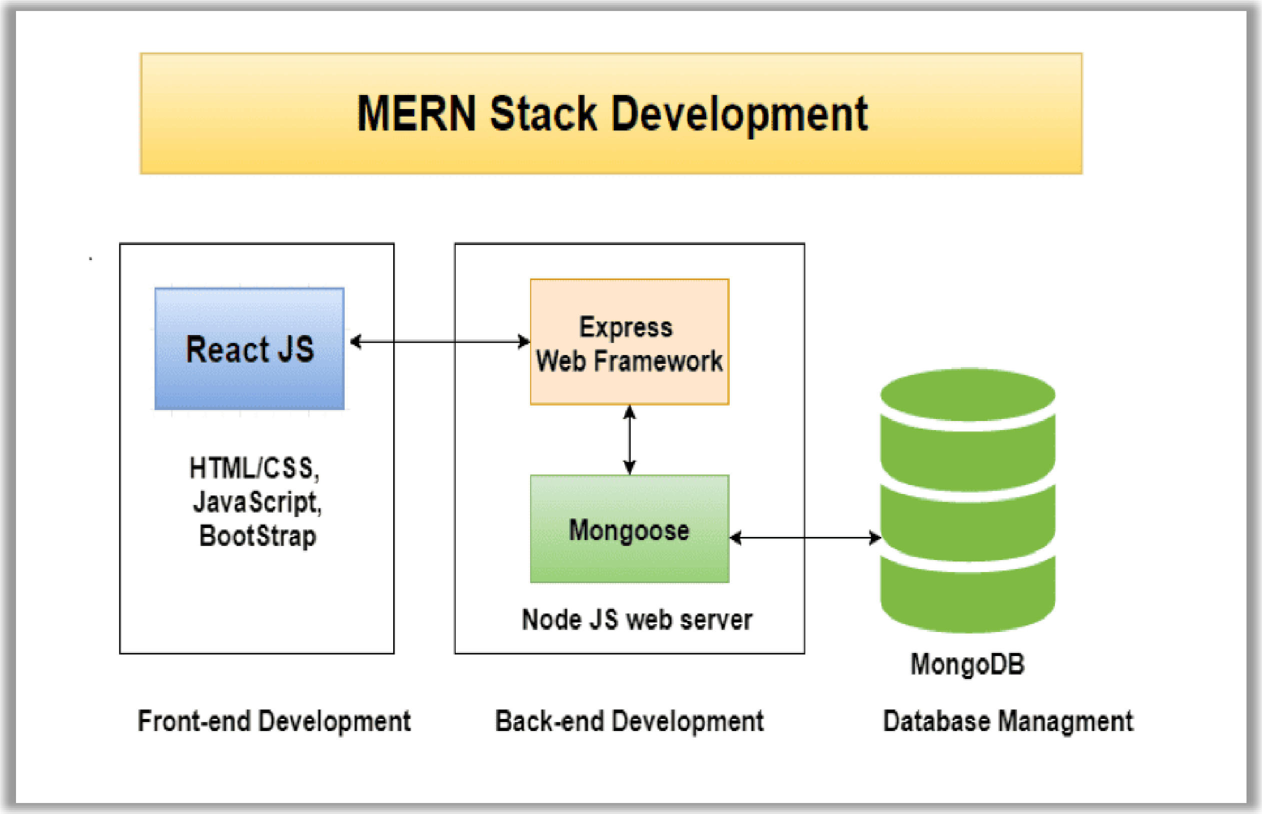


Figure 2.2: Architecture of MERN stack

The graphic shows how MongoDB, NodeJS, and Express are used to create the backend of websites. This implies that these frameworks handle numerous related tasks, including as database administration, scripting, HTML documentation, and HTTP requests [10].

Finally, the React framework's job here is to handle HTTP requests. We can use Http calls to set up dynamic data downloads without having to reload the page. It's because to this that the final product is a lot quicker than the typical front-end software. To put it simply, MERN is a powerful technology platform. The MERN stack's four components complement one other, making it more comprehensive.

2.3 Multimedia E-commerce Landscape

The world of multimedia e-commerce is a growing and changing part of the e-commerce business as a whole. It is all about buying and selling multimedia products online, like electronics, gadgets, accessories etc. This digital revolution has changed a lot about how people use multimedia products. As a result, an e-commerce site like "Multimedia Mart" is very important for meeting the many needs of today's consumers.

2.3.1 Growth and Significance

Over the last decade multimedia e-commerce has evolved very quickly. With the rise of high-speed internet connections, mobile and other digital devices, customers now have easy access to a wide range of multimedia products. Growing use of digital media for entertainment, learning, and work has increased online demand for multimedia products.

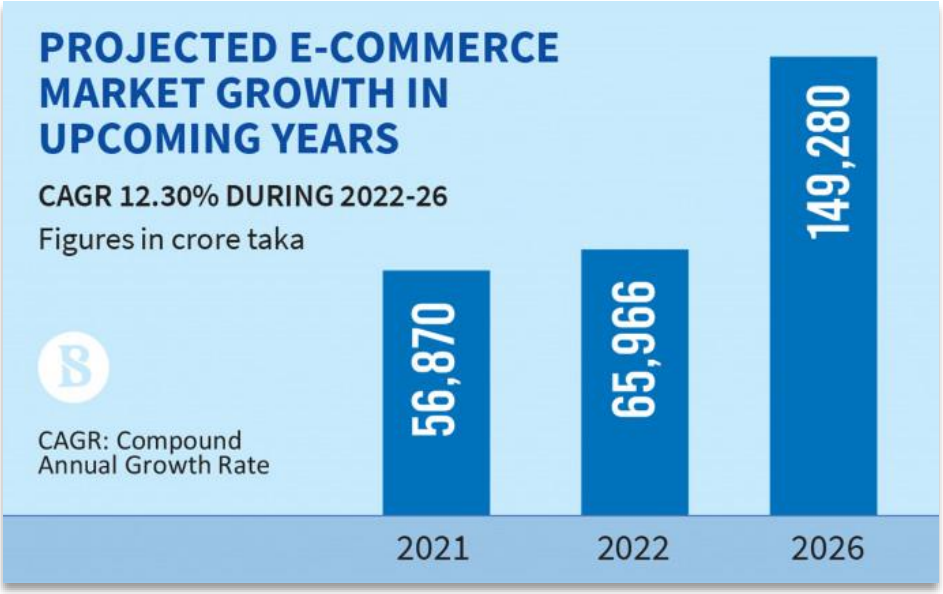


Figure 2.3.1: Projected E-Commerce Growth

As we can see, A recent market study shows that the e-commerce business in Bangladesh is projected to grow by 17.61% per year, with the goal of reaching Tk149,966 crore in 2026 [11].

2.3.2 Challenges

Multimedia related e-commerce platform have a lot of promise, but it also has some problems that need to be fixed for it to be successful in the long run. These problems affect both well-known platforms and new platforms like "Multimedia Mart," so it's important to understand them and put together effective strategies. These challenges are-

Fierce Competition: Platforms from industry giants to new entrepreneurs compete for consumers' attention. Newer startups struggle because established companies have brand recognition and a huge client base. Unique products and services, great user experiences, and innovative marketing methods are needed to separate the "Multimedia Mart" from other competitors [12].

Security Concerns: Online purchases and personal data sharing have raised data security and privacy concerns. Data leaks, identity theft, and cyberattacks can damage user’s trust. "Multimedia Mart" must prioritize security, develop secure payment methods, and reassure users of its data protection.

Technological Changes: The fast-paced evolution of technology has advantages and disadvantages. Constant technological innovation continually shifts customer preferences and the competitive landscape. To remain competitive, "Multimedia Mart" must adapt to these changes by keeping its platform up-to-date and in step with users' ever-evolving demands.

Counterfeit Products: Fake products are a major issue in e-commerce, particularly multimedia. These items harm user experience and the platform's reputation. "Multimedia Mart" needs strict quality control to check product authenticity via its platform. Review and feedback processes can help discover and resolve counterfeit merchandise.

2.3.3 Target Audience

The demographic of multimedia e-commerce customers has expanded beyond traditional boundaries. The primary target audience for multimedia e-commerce platforms like "Multimedia Mart" includes students, tech-savvy individuals, and regular users of digital media. Students, in particular rely on multimedia related products for educational purposes. The platform's focus on this audience aligns well with their needs for affordable, high-quality multimedia products.

2.3.4 Product Categories

Various Multimedia e-commerce platforms offer a wide range of product categories, such as electronic gadgets, digital contents, softwares etc. As you can see there are mix between hardware and software related products, I chose to select multimedia related hardware products on my “Multimedia Mart” e-commerce site. The reason behind is, in a competitive e-commerce landscape, offering hardware products can help differentiate my platform from others that primarily focus on digital content or software. This uniqueness could attract customers seeking a wide-ranging shopping destination.

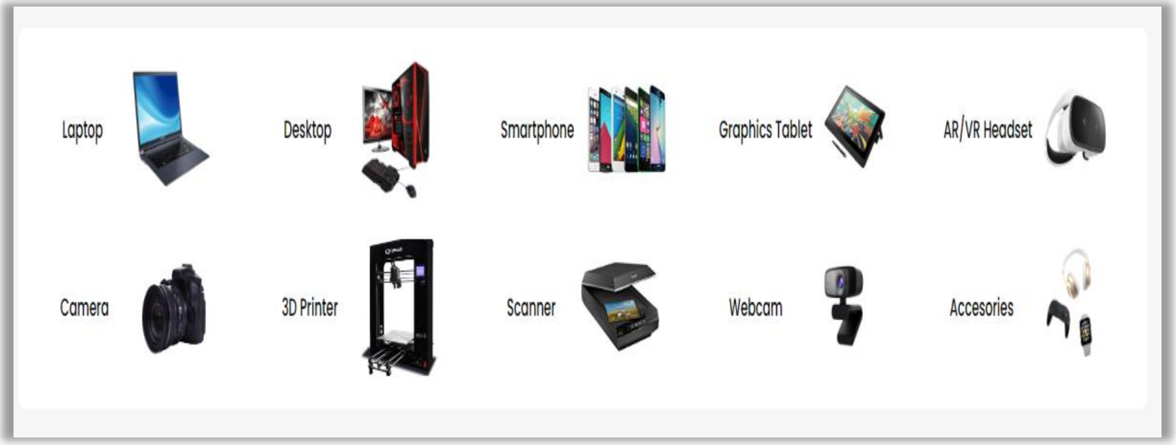


Figure 2.3.4: “Multimedia Mart” Product Categories

By doing an in-depth analysis of the multimedia e-commerce landscape, "Multimedia Mart" has the ability to position itself strategically in order acquire an important part of the market while simultaneously satisfying the requirements of its target audience.

2.4 Related Websites

Here are some successful websites that I’ve taken inspiration from in order to create a compelling and competitive platform like "Multimedia Mart."

2.4.1 BestBuy

BestBuy is a major electronics retailer in the United States offering a wide range of multimedia products. Best Buy is ranked no.3 as a computer electronics and technology ecommerce website with an estimated figure of 192.70 million monthly visits [13].

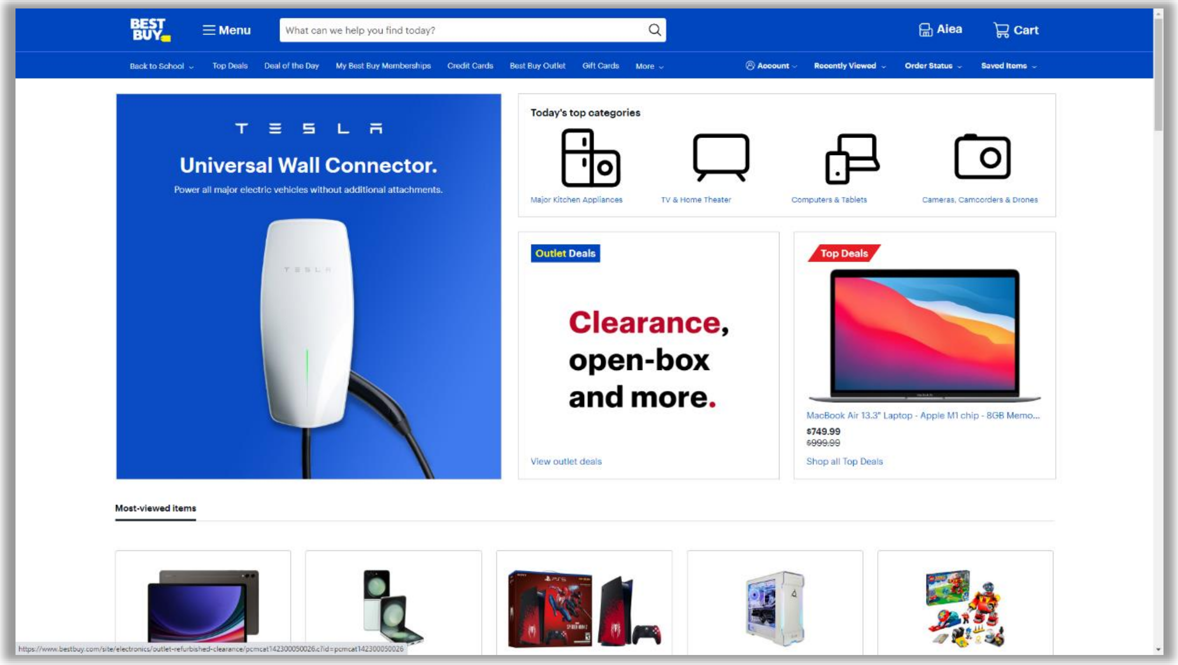


Figure 2.4.1: BestBuy Website

Features:

- An interface that is easy to use and focuses on tech and multimedia.
- An easy way to sign up and log in.
- Sections for best-selling and featured goods that are easy to find.
- Accounts for users to keep track of orders.
- Credit cards and PayPal are both safe ways to pay.

- Detailed product specifications and customer reviews.

2.4.2 eBay

eBay is an American international ecommerce company. Worldwide, they've delivered millions of items. It is one of our top ecommerce websites since they sell electronics, art & collectibles, accessories, sporting items and more.

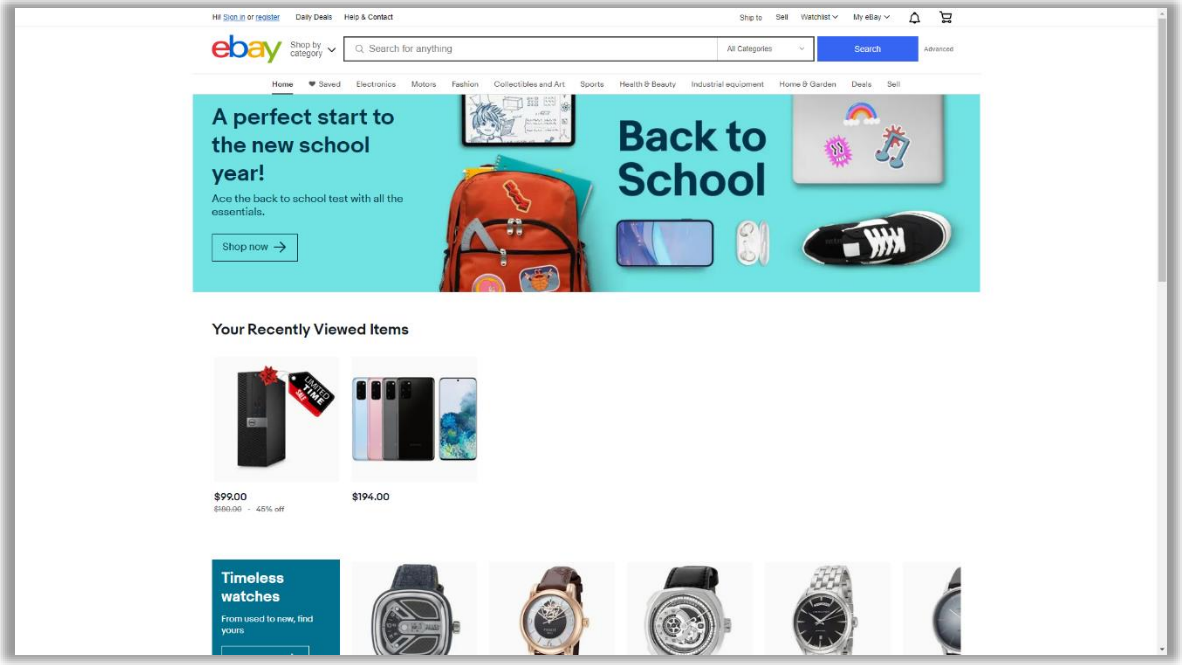


Figure 2.4.2: eBay Website

Features:

- User-friendly platform with a wide-ranging product catalog.
- Efficient account creation process.
- Auction-style listings and fixed-price options for sellers.
- Sections for featured and trending products.
- Opportunities for individual sellers and established brands to participate.
- Buyer and seller dashboards for transaction management.
- Secure payment options including PayPal and credit cards.

2.4.3 Star Tech

Star Tech is a well-known technology-focused e-commerce platform in Bangladesh offering a wide range of electronics and tech-related products.

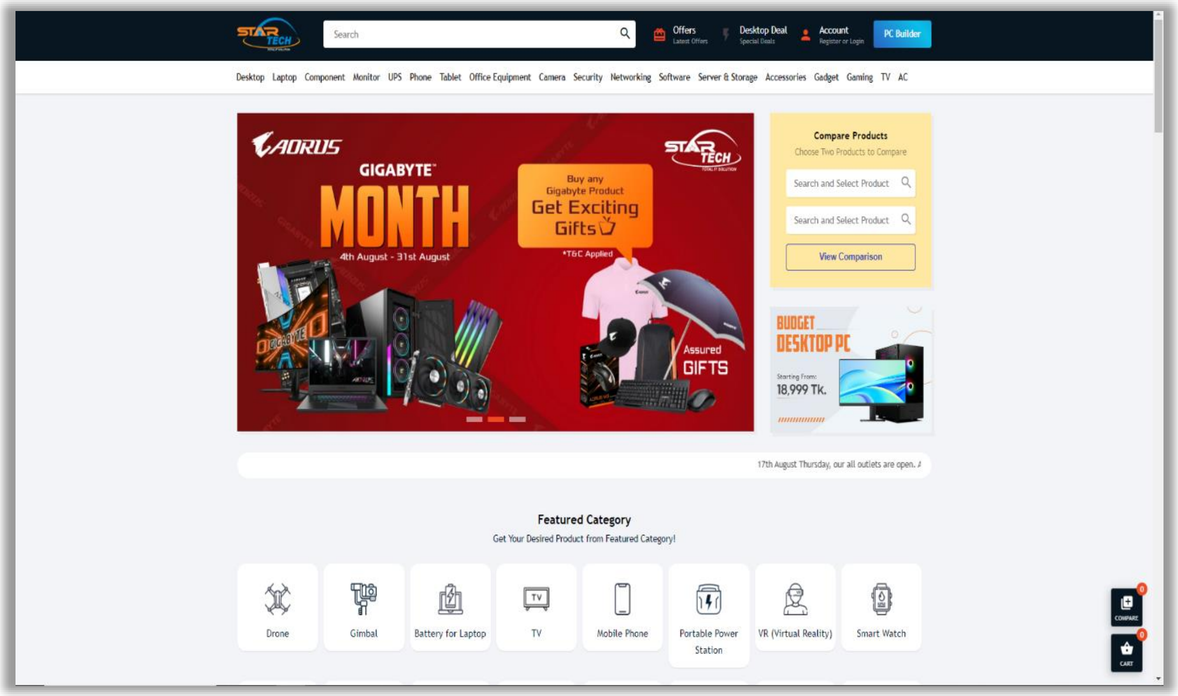


Figure 2.4.3: Star Tech Website

Features:

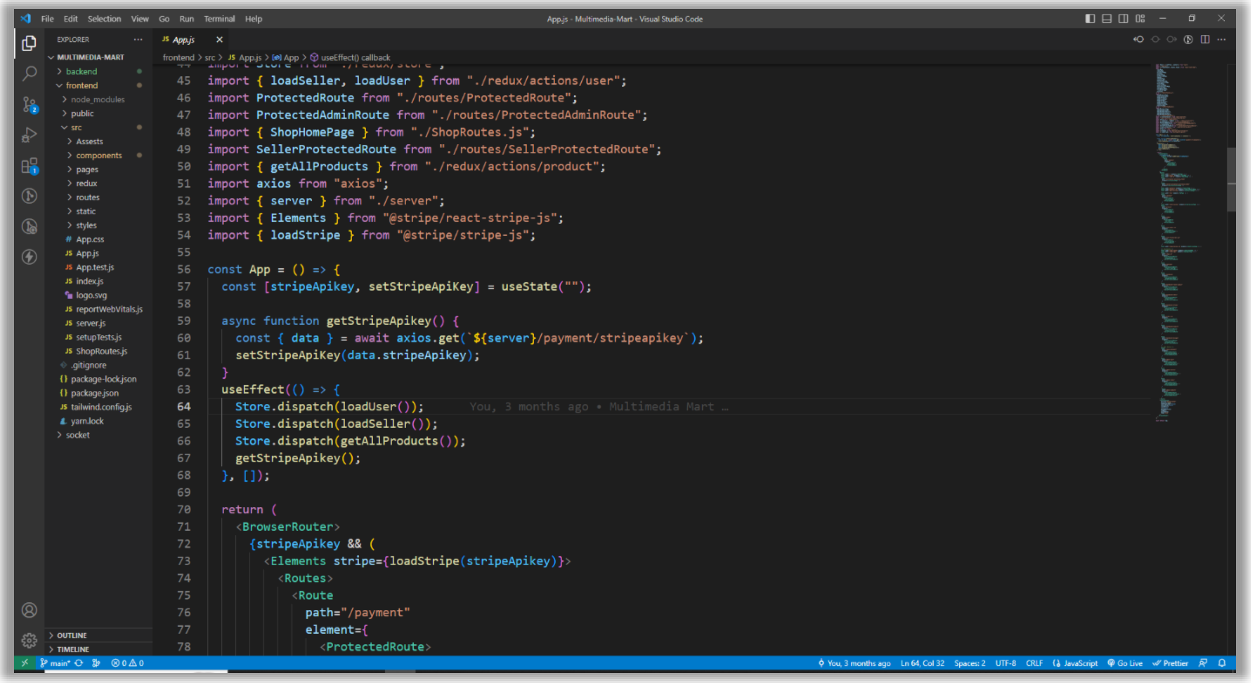
- Search and filter options for easy navigation.
- Add to cart, checkout functionality & Order tracking and history.
- Well Responsive web design for different screen size and devices.
- It has a custom PC building option.
- Huge variety of electronic and multimedia related products.
- Secure payment options including Bkash, Nogod, Rocket and credit cards.

CHAPTER 3

SOFTWARES USED TO IMPLEMENT MY PROJECT

3.1 Visual Studio Code

Visual Studio Code (VS Code) is a popular text editor and Integrated Development Environment (IDE) among developers for creating web applications, including those created with the MERN framework. It provides a robust and feature-rich environment for writing, revising, and debugging code quickly. Developers can concentrate on their code thanks to VS Code's user-friendly interface and clean architecture. It provides features such as code highlighting, autocompletion, and code excerpts, which can accelerate the development process and reduce coding errors [14].



```
import Store from "./redux/store";
import { loadSeller, loadUser } from "./redux/actions/user";
import ProtectedRoute from "./routes/ProtectedRoute";
import ProtectedAdminRoute from "./routes/ProtectedAdminRoute";
import { ShopHomePage } from "./ShopRoutes.js";
import SellerProtectedRoute from "./routes/SellerProtectedRoute";
import { getAllProducts } from "./redux/actions/product";
import axios from "axios";
import { server } from "./server";
import { Elements } from "@stripe/react-stripe-js";
import { loadStripe } from "@stripe/stripe-js";

const App = () => {
  const [stripeApikey, setStripeApiKey] = useState("");

  async function getStripeApikey() {
    const { data } = await axios.get(`${server}/payment/stripeapikey`);
    setStripeApiKey(data.stripeApikey);
  }
  useEffect(() => {
    Store.dispatch(loadUser());       You, 3 months ago • Multimedia Mart …
    Store.dispatch(loadSeller());
    Store.dispatch(getAllProducts());
    getStripeApikey();
  }, []);

  return (
    <BrowserRouter>
      {stripeApikey && (
        <Elements stripe={loadStripe(stripeApikey)}>
          <Routes>
            <Route
              path="/payment"
              element={
                <ProtectedRoute>
```

Figure 3.1: Visual Studio Code Interface

3.2 Node.js

Node.js is a JavaScript runtime that enables server-side execution of JavaScript code outside the browser. It facilitates the development of server-side applications using JavaScript and offers a variety of packages and modules from the npm (Node Package Manager) ecosystem.

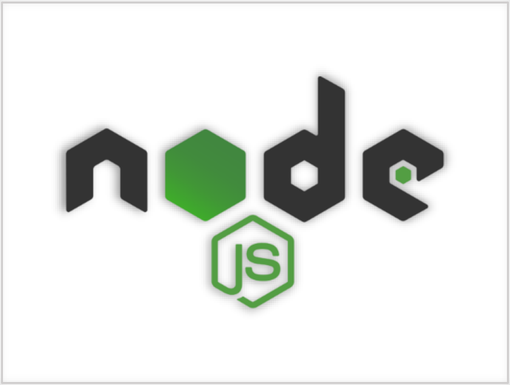


Figure 3.2: Node.js Logo

3.3 Express.js

Express.js, a web application framework built on Node.js, is widely used. Through the provision of routing, middleware, and other utilities, it facilitates the development of web servers and APIs. Express.js is utilized for the management of server-side logic, the definition of routes, and the communication with databases.

Figure 3.3: Express.js Logo

3.4 MongoDB

MongoDB, a NoSQL document-oriented database, is frequently utilized in web application development, including MERN stack applications. It's appropriate for a variety of applications due to its flexible and scalable data storage and retrieval. Complex data associations can be handled efficiently using MongoDB's flexible data model. It allows embedded documents and arrays for nesting and hierarchical data representation. In e-commerce applications that handle product catalogs and user reviews, this flexibility is useful.

Figure 3.4: MongoDB Logo

3.5 React

React is a powerful JavaScript library for building dynamic and reusable user interfaces. Its component-based architecture, virtual DOM, one-way data flow, and extensive ecosystem of tools and libraries make it a popular choice for web developers working with the MERN stack. React's focus on performance, maintainability, and scalability contributes to creating robust and efficient web applications. One of the best things about React JS is that it lets us reuse components. Developers don't have to write different codes for the same features, which saves them time. Also, if we change something in one part, it won't have any effect on the rest of the program.

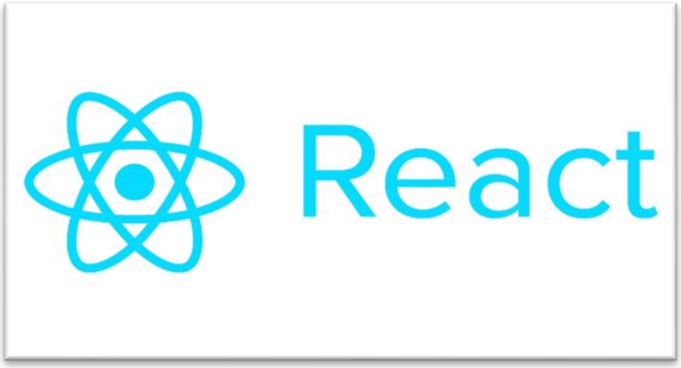


Figure 3.5: React Logo

3.6 Figma

Figma was used as this project’s UI/UX tool since it allows to create a user-friendly website. Figma is a low-cost, user-friendly, and adaptable tool for creating wireframes, prototypes, and user interfaces. It has unique design tools that give web developers an interesting experience. It can be used for a variety of jobs, such as vector drawing, user interface design, app design, and prototyping. The world's leading companies, including Airbnb, Dropbox, and Instagram all utilize this prototype tool [15].

Figure 3.6: Figma Logo

3.7 GitHub

GitHub is a widely-used platform for storing and discussing programming projects' source code. It's a platform used by numerous businesses and non-profits to improve project management and teamwork among their programming team. As of October 2020, it has more than 60 million new repositories and over 56 million developers, making it the most popular source code site [16].

Figure 3.7: GitHub Logo

3.8 Chrome Devtools

Chrome DevTools is a complete collection of tools that enables developers to build, debug, and optimize websites more effectively, which ultimately leads to improved user experiences and more effective web apps. It's an all-inclusive set of tools for analyzing page performance, debugging code, and making changes. Among the many options are renaming the class, adding a new attribute, and tinkering with the existing ones' values.

3.9 Dependencies

1. material-ui
2. redux-toolkit
3. tailwind CSS
4. axios
5. country-state-city
6. stripe
7. react
8. react-router-dom
9. react-icons
10. react-lottie
11. react-redux
12. react-toastify
13. socket.io-client
14. cookie-parser
15. cors
16. dotenv
17. express
18. jsonwebtoken
19. mongoose
20. multer
21. nodemailer
22. nodemon

CHAPTER 4
DEVELOPMENT AND WORKFLOW

4.1 Introduction

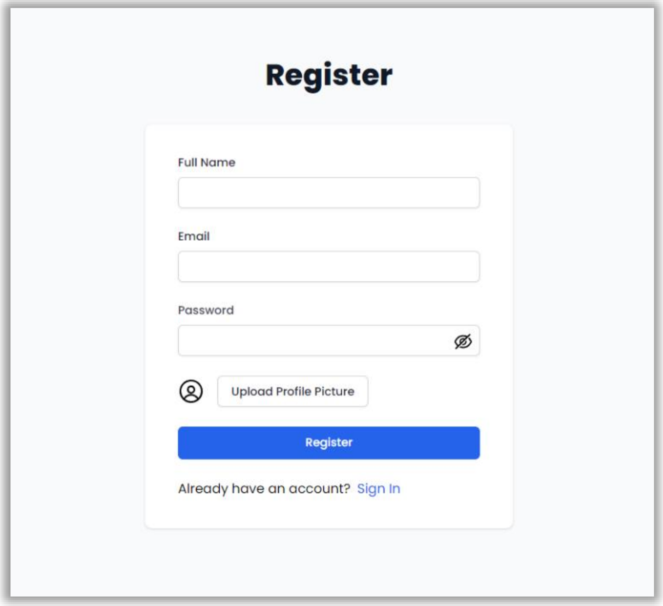
The development of "Multimedia Mart," a user-friendly MERN stack e-commerce platform, aimed to innovate the way multimedia products are traded online. The idea and creation of "Multimedia Mart" were meant to make it a central place for people who are involved in multimedia, students, and people looking for a wide range of multimedia related products. The project went through a long development process with many important steps. Each of these steps added to the platform's general functionality and user experience. This part is an in-depth look at the process of making this website, including the architectural choices, workflow and the most important features and functions.

4.2 Key Features and Functionality

Through the combination of MERN stack technology, this platform managed to develop a wide variety of features and functionalities that addressed the varying requirements of its users:

4.2.1 User Authentication and Verification

User authentication is a key part of any online platform. It makes sure that users can access the platform safely and keeps user data safe. "Multimedia Mart" set up a strong security system that included safe ways to register and log in. User information, like passwords, was protected to make it safer. The security of user accounts was improved by adding a verification email system, which made sure that only real users could log in. After signing up, users got an email with an activation link.



Register

Full Name

Email

Password

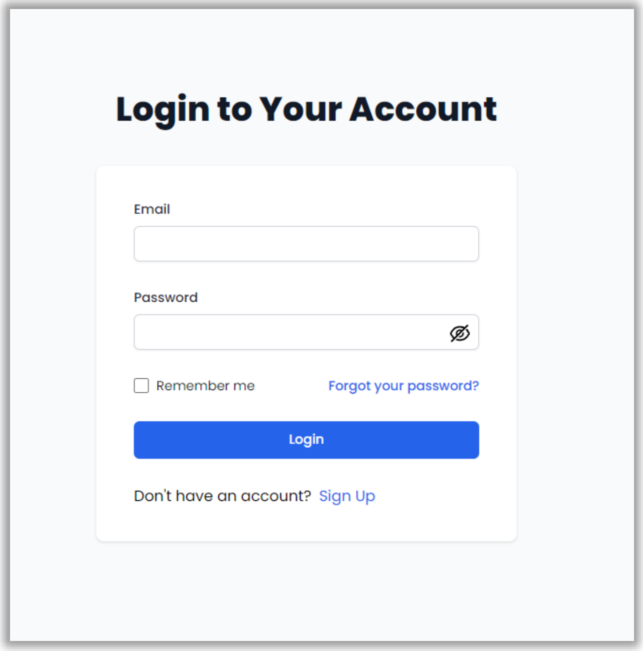
Upload Profile Picture

Register

Already have an account? Sign In

Figure 4.2.1.a: Register Page

To go through a protected route, user must have an id. So, the page will redirect to register page as the image above and if the user has already an id, he should click on ‘sign in’ and it’ll redirect to login page as the image below.



Login to Your Account

Email

Password

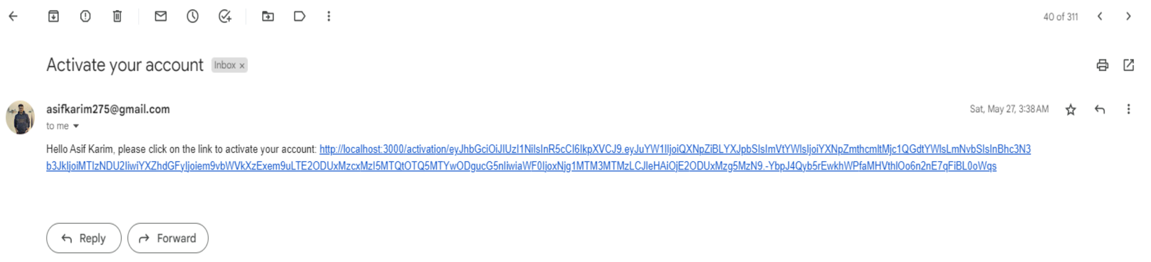
□ Remember me

Forgot your password?

Login

Don't have an account? Sign Up

Figure 4.2.1.b: Login Page



Activate your account

asifkarim275@gmail.com

to me

Sat, May 27, 3:38 AM

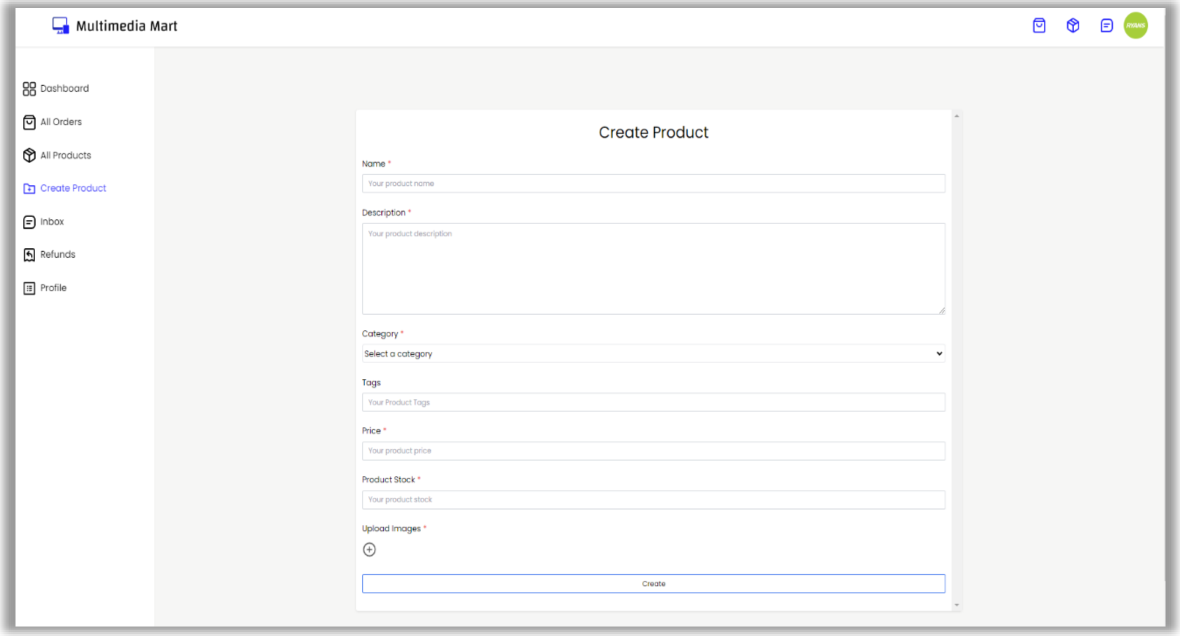
Hello Asif Karim, please click on the link to activate your account: http://localhost:3000/activation/eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJuYW1lIjoiQXNpZiBLYXJpbSIsImVtYWlsIjoiYXNpZmthcmltMjc1QGdtYWlsLmNvbSIsInBhc3N3b3JkIjoiMTIzNDU2IiwiYXZhdGFyIjoiem9vbWVkXzExem9uLTE2ODUxMzcxMzI5MTQtOTQ5MTYwODgucG5nIiwiaWF0IjoxNjg1MTM3MTMzLCJleHAiOjE2ODUxMzg5MzN9.-YbpJ4Qyb5rEwkhWPfaMHVthlOo6n2nE7qFiBL0oWqs

Figure 4.2.1.c: Verification email

User will get a verification email via Gmail and if they click the link their id will be automatically activated.

4.2.2 Seller Integration & Product Management

Sellers, whether individuals or established shops have the tools to manage their products through dedicated dashboards. The 'Create Product' feature empowers sellers to provide product details description, including name, category, price, product stock, tags and high-quality images. This dashboard serves as a hub for tracking sales, monitoring pending orders and overview earnings.



Create Product

Name *

Your product name

Description *

Your product description

Category *

Select a category

Tags

Your Product Tags

Price *

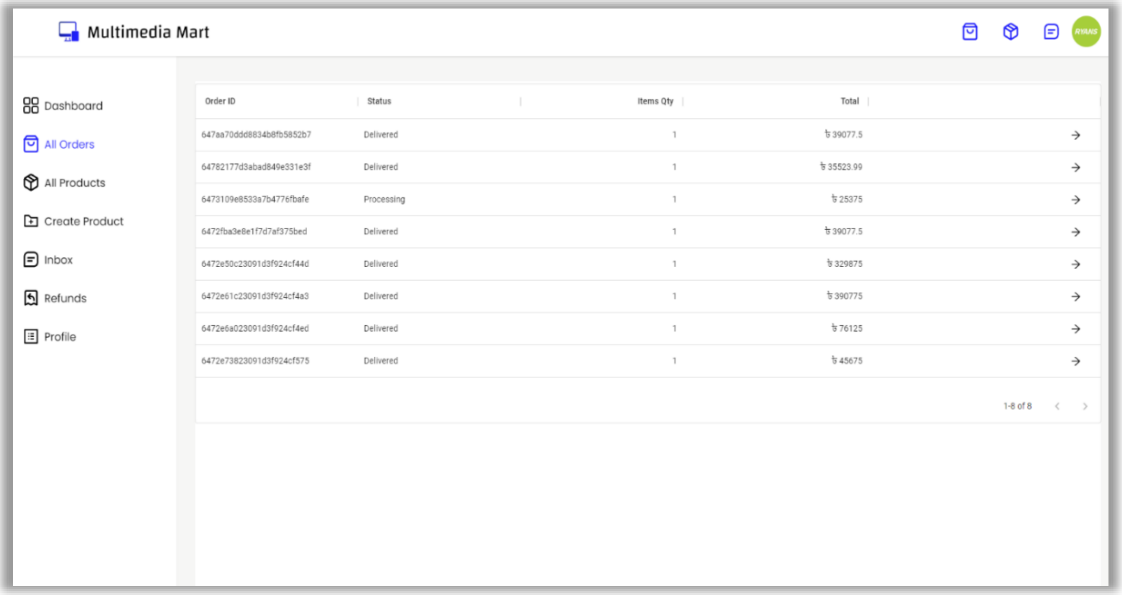
Your product price

Product Stock *

Your product stock

Upload Images *

Figure 4.2.2.a: Create Product

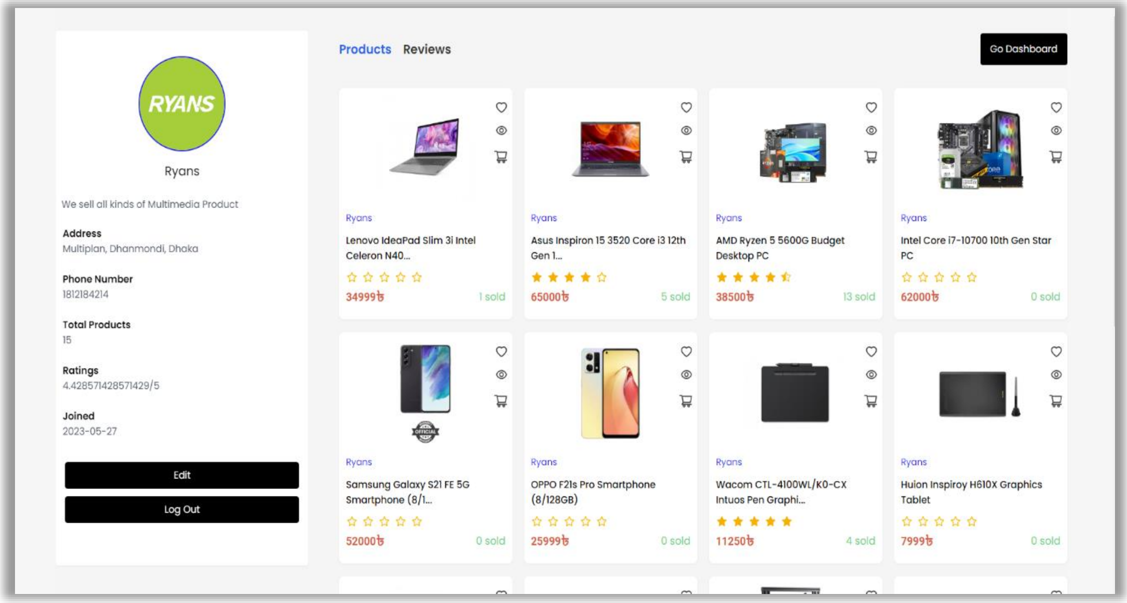


Order ID	Status	Items Qty	Total
647aa70ddd8834b8fb5852b7	Delivered	1	৳ 39077.5
64782177d3abad849e331e3f	Delivered	1	৳ 35523.99
6473109e8533a7b4776fbafe	Processing	1	৳ 25375
6472fba3e8e1f7d7af375bed	Delivered	1	৳ 39077.5
6472e50c23091d3f924cf44d	Delivered	1	৳ 329875
6472e61c23091d3f924cf4a3	Delivered	1	৳ 390775
6472e6a023091d3f924cf4ed	Delivered	1	৳ 76125
6472e73823091d3f924cf575	Delivered	1	৳ 45675

1-8 of 8

Figure 4.2.2.b: All Orders

In the image above, we can see that the seller has access to all of the orders he has received through the seller dashboard. In the image below, we can see that a seller has a specific section of his or her profile that is dedicated to them.



Ryans

We sell all kinds of Multimedia Product

Address
Multiplan, Dhanmondi, Dhaka

Phone Number
1812184214

Total Products
15

Ratings
4.428571428571429/5

Joined
2023-05-27

Ryans
Lenovo IdeaPad Slim 3i Intel Celeron N40...
34999৳ 1 sold

Ryans
Asus Inspiron 15 3520 Core i3 12th Gen 1...
65000৳ 5 sold

Ryans
AMD Ryzen 5 5600G Budget Desktop PC
38500৳ 13 sold

Ryans
Intel Core i7-10700 10th Gen Star PC
62000৳ 0 sold

Ryans
Samsung Galaxy S21 FE 5G Smartphone (8/1...
52000৳ 0 sold

Ryans
OPPO F21s Pro Smartphone (8/128GB)
25999৳ 0 sold

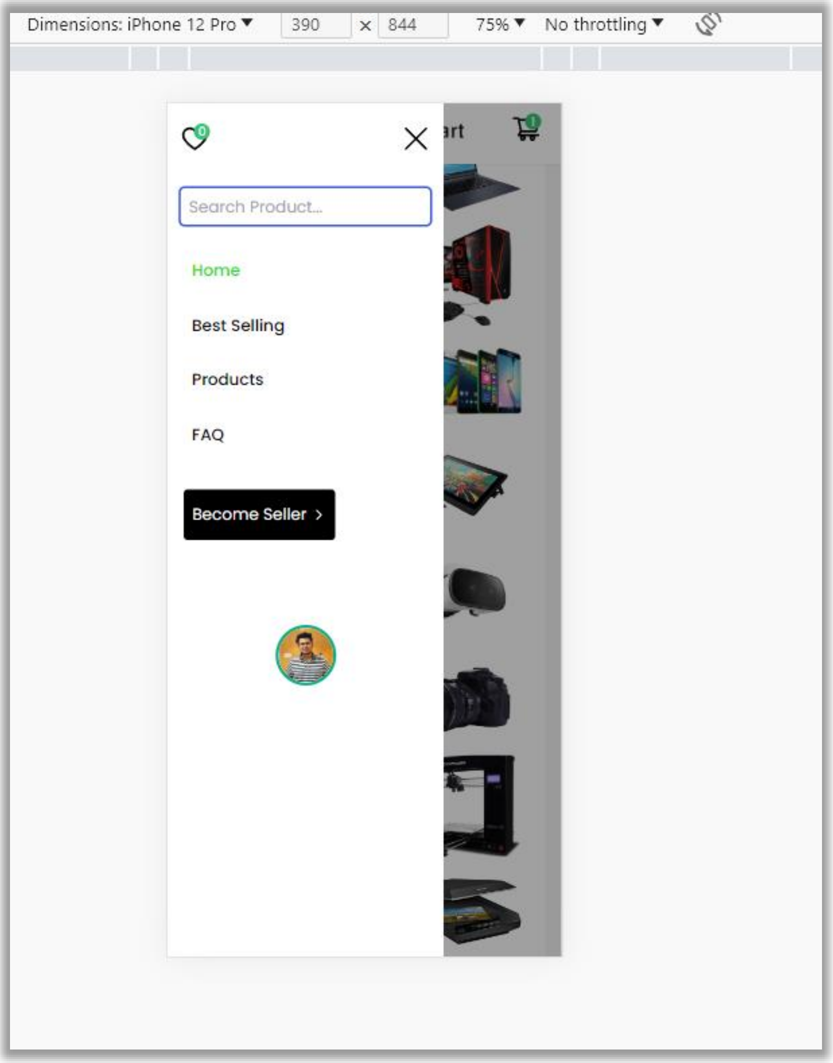
Ryans
Wacom CTL-4100WL/K0-CX Intuos Pen Graphi...
11250৳ 4 sold

Ryans
Huion Inspiroy H610X Graphics Tablet
7999৳ 0 sold

Figure 4.2.2.c: Seller Profile

4.2.3 Full Responsive

Users are able to visit the “Multimedia Mart” through desktop computers, laptops, tablets or smartphones thanks to the site's responsive design. Which provided a consistent and very accessible platform for all of these devices. The platform's overall authenticity and accessibility were both improved as a result of the dedication to responsive design, which not only led to an increase in user satisfaction but also brought it into alignment with modern web design standards.

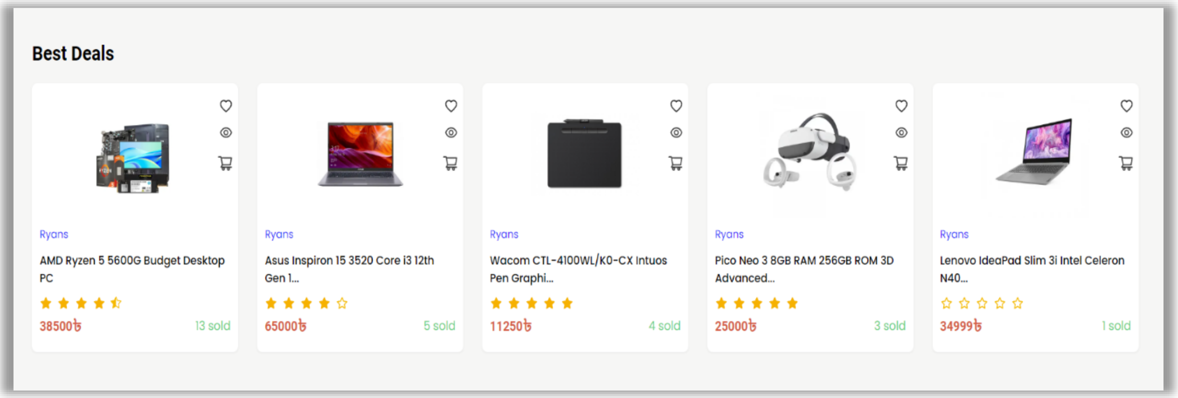


Dimensions: iPhone 12 Pro ▾ 390 × 844 75% ▾ No throttling ▾

Figure 4.2.3: Responsive Design

4.2.4 Best Selling and Featured Products

“Multimedia Mart” has the "Best Selling" and "Featured Products" section that helps to attract more users and is very useful to both sellers and buyers. The "Best Selling" section highlighted products that got a lot of sells, so users can see what is popular. This information is very helpful for both people who wants to buy or sell because it shows them how the market is changing and what people likes.



Best Deals

Ryans
AMD Ryzen 5 5600G Budget Desktop PC
★★★★⯪
38500৳ 13 sold

Ryans
Asus Inspiron 15 3520 Core i3 12th Gen 1...
★★★★☆
65000৳ 5 sold

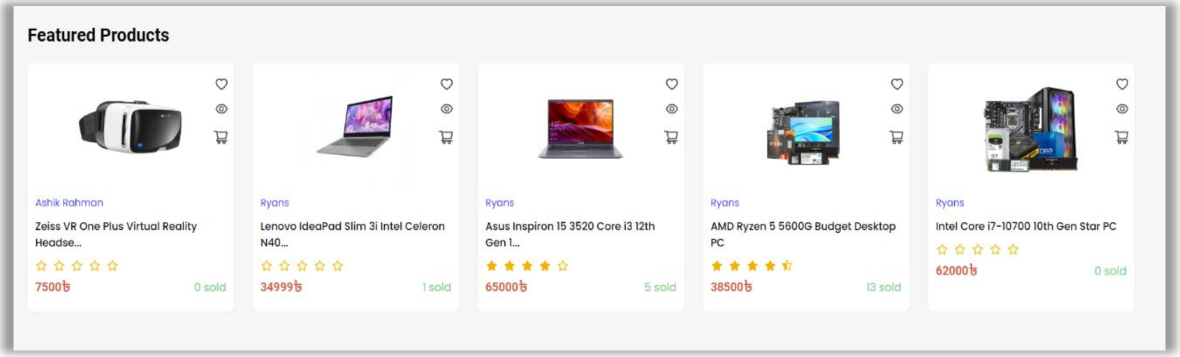
Ryans
Wacom CTL-4100WL/K0-CX Intuos Pen Graphi...
★★★★★
11250৳ 4 sold

Ryans
Pico Neo 3 8GB RAM 256GB ROM 3D Advanced...
★★★★★
25000৳ 3 sold

Ryans
Lenovo IdeaPad Slim 3i Intel Celeron N40...
☆☆☆☆☆
34999৳ 1 sold

Figure 4.2.4.a: Best Deals

The "Featured Products" section displays off the newest additions to the site, making users excited and helping them find new things. This feature makes people want to come back and keep their interest.



Featured Products

Ashik Rahman
Zeiss VR One Plus Virtual Reality Headse...
☆☆☆☆☆
7500৳ 0 sold

Ryans
Lenovo IdeaPad Slim 3i Intel Celeron N40...
☆☆☆☆☆
34999৳ 1 sold

Ryans
Asus Inspiron 15 3520 Core i3 12th Gen 1...
★★★★☆
65000৳ 5 sold

Ryans
AMD Ryzen 5 5600G Budget Desktop PC
★★★★⯪
38500৳ 13 sold

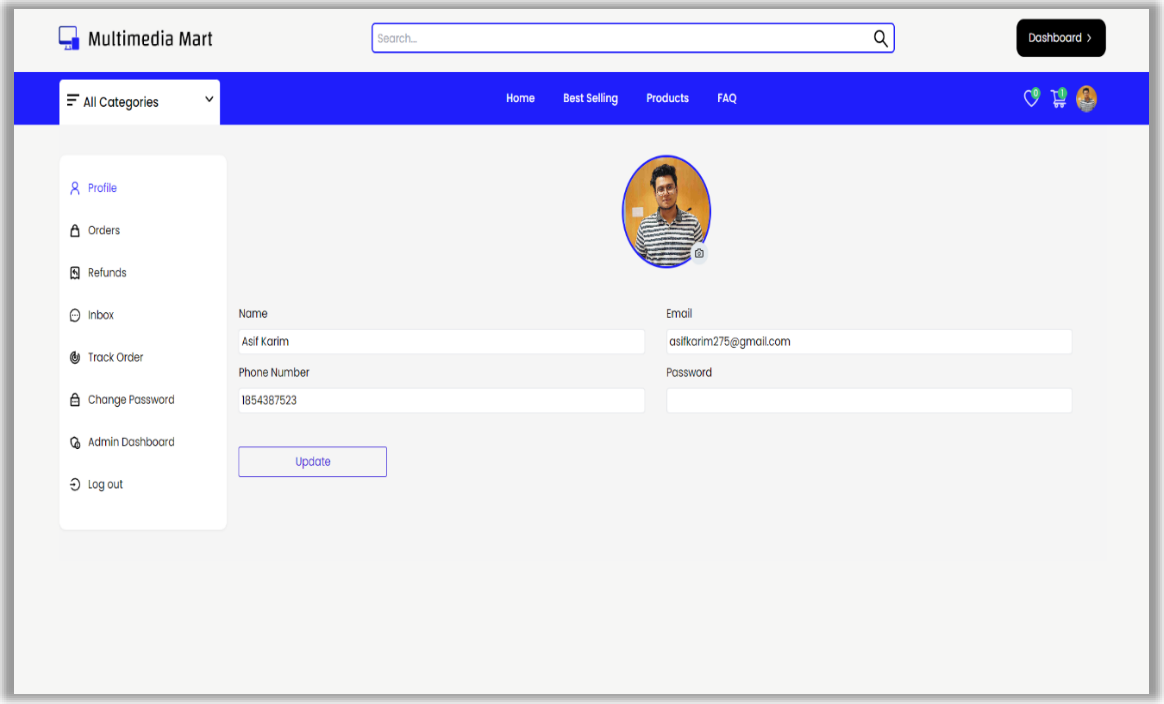
Ryans
Intel Core i7-10700 10th Gen Star PC
☆☆☆☆☆
62000৳ 0 sold

Figure 4.2.4.b: Featured Products

4.2.5 User, Seller and Admin Dashboards

User, seller, and admin all have their own dashboards, which make the user experience unique for every role. The user dashboard let buyers manage their orders, refunds, talk to sellers and track the progress of their product deliveries.

Through their own dashboards, sellers have full control over all of their activities. This dashboard let sellers keep an eye on their sales, track orders, manage their products, talk to buyers, handle refunds, and make their profiles look real and interesting.



Name: Asif Karim

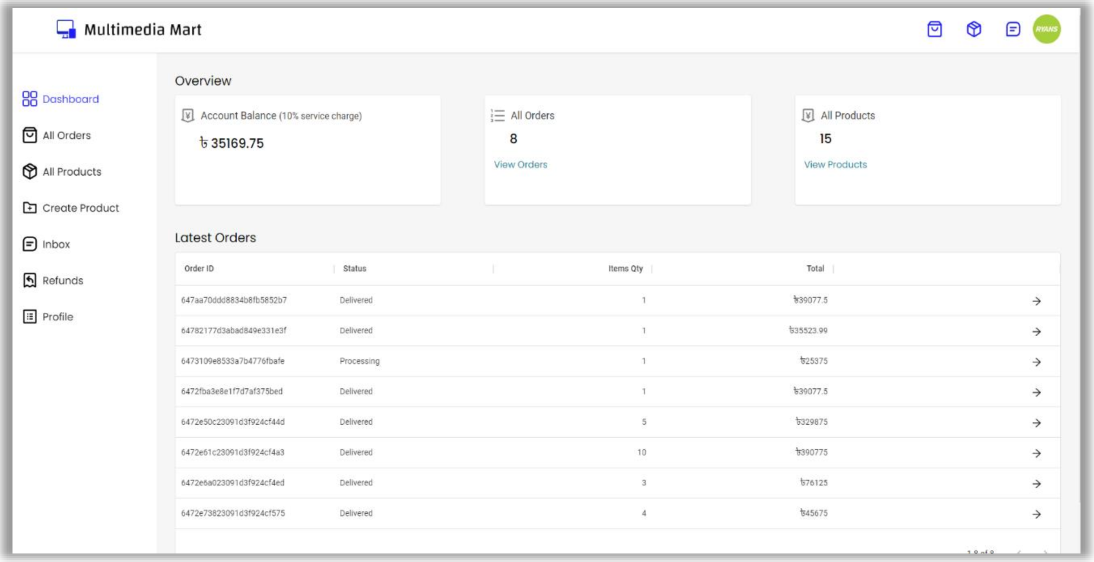
Email: asifkarim275@gmail.com

Phone Number: 1854387523

Password:

Figure 4.2.5.a: User Dashboard

A user can see their orders and the update of their order through user dashboard. To get the recent update there is a section called ‘track order’ section which helps to give the update to user. Users can also chat with the sellers through inbox section.



Overview

Account Balance (10% service charge)
৳ 35169.75

All Orders
8
View Orders

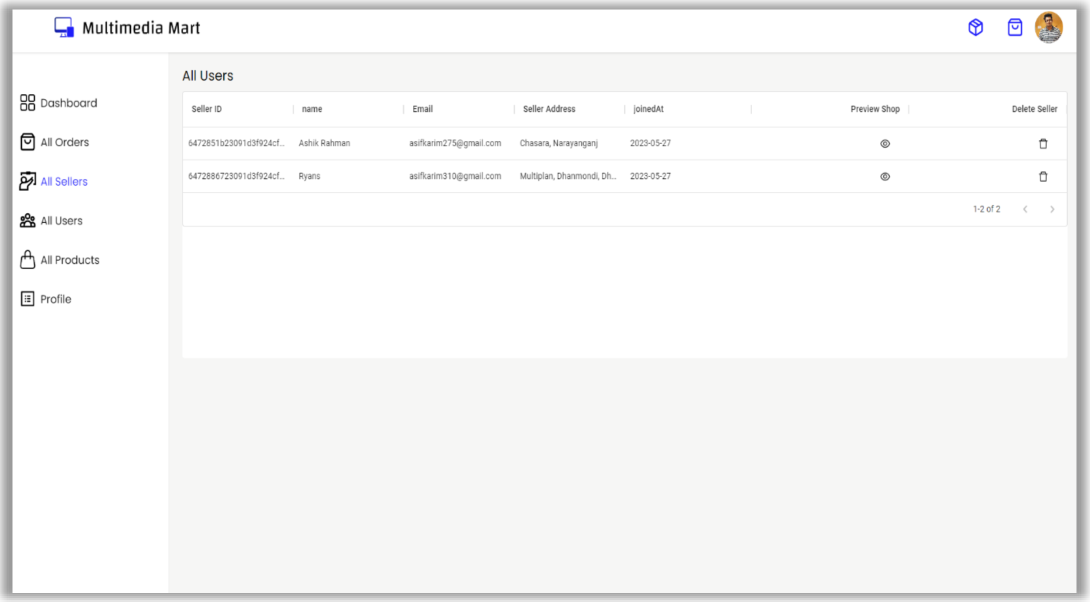
All Products
15
View Products

Latest Orders

Order ID	Status	Items Qty	Total	
647aa70ddd8834b8fb5852b7	Delivered	1	৳39077.5	→
64782177d3abad849e331e3f	Delivered	1	৳35523.99	→
6473109e8533a7b4776fbafe	Processing	1	৳25375	→
6472fba3e8e1f7d7af375bed	Delivered	1	৳39077.5	→
6472e50c23091d3f924cf44d	Delivered	5	৳329875	→
6472e61c23091d3f924cf4a3	Delivered	10	৳390775	→
6472e6a023091d3f924cf4ed	Delivered	3	৳76125	→
6472e73823091d3f924cf575	Delivered	4	৳45675	→

Figure 4.2.5.b: Seller Dashboard

Admins have full control over the site through the admin dashboard. They can handle user accounts, keep an eye on products, keep track of sales and earnings and moderate content made by users. To maintain and secure the platform admins can delete any product, users or sellers.



All Users

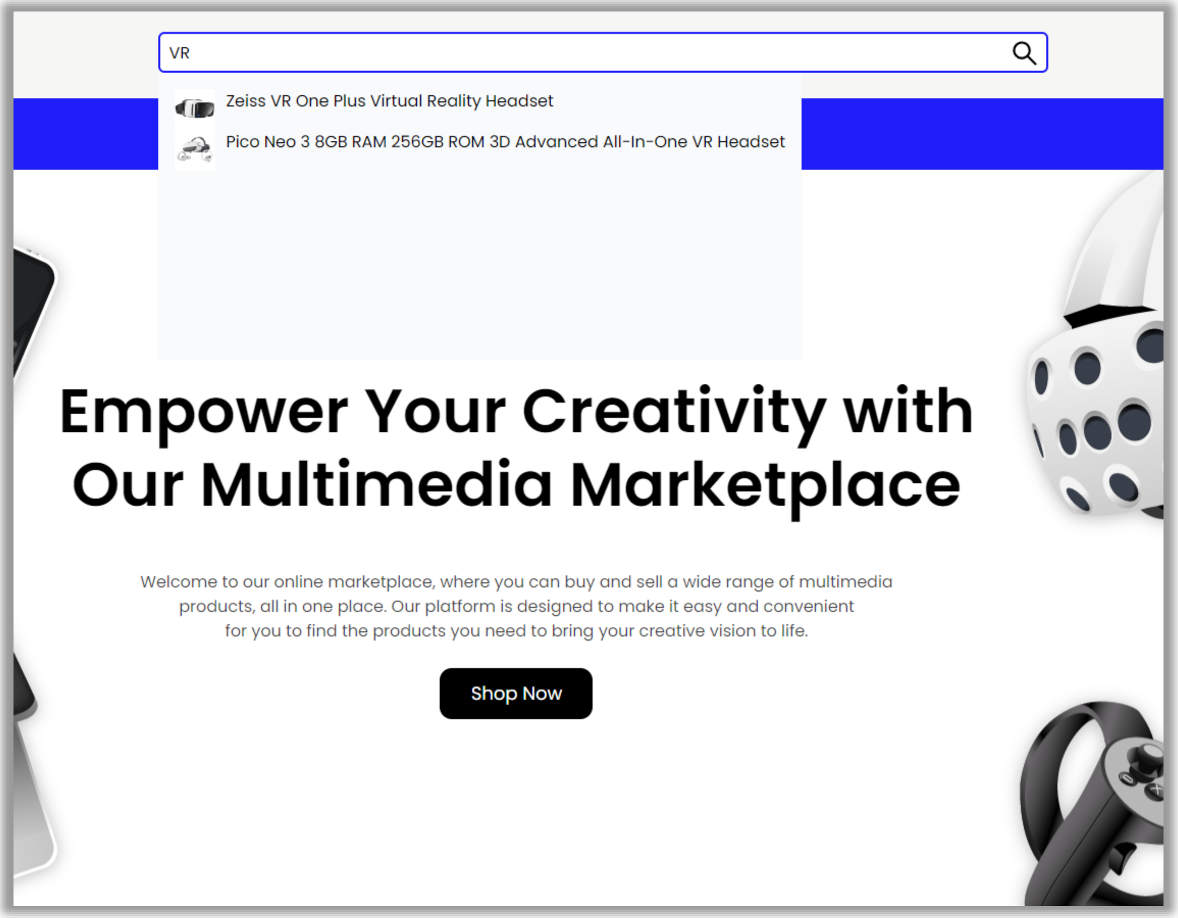
Seller ID	name	Email	Seller Address	joinedAt	Preview Shop	Delete Seller
6472851b23091d3f924cf...	Ashik Rahman	asifkarim275@gmail.com	Chasara, Narayanganj	2023-05-27	👁	🗑
6472886723091d3f924cf...	Ryans	asifkarim310@gmail.com	Multiplan, Dhanmondi, Dh...	2023-05-27	👁	🗑

1-2 of 2

Figure 4.2.5.c: Admin Dashboard

2.4.6 Search Bar

A solid search bar at the top of the website makes it easy to find products effortlessly using various tags, which made the user experience easier. The search feature used a flexible method that let users find products by using different criteria, like keywords and categories. This search system makes it easier for users to find the products they were looking for, which improves their experience and get them more involved.



VR

Zeiss VR One Plus Virtual Reality Headset

Pico Neo 3 8GB RAM 256GB ROM 3D Advanced All-In-One VR Headset

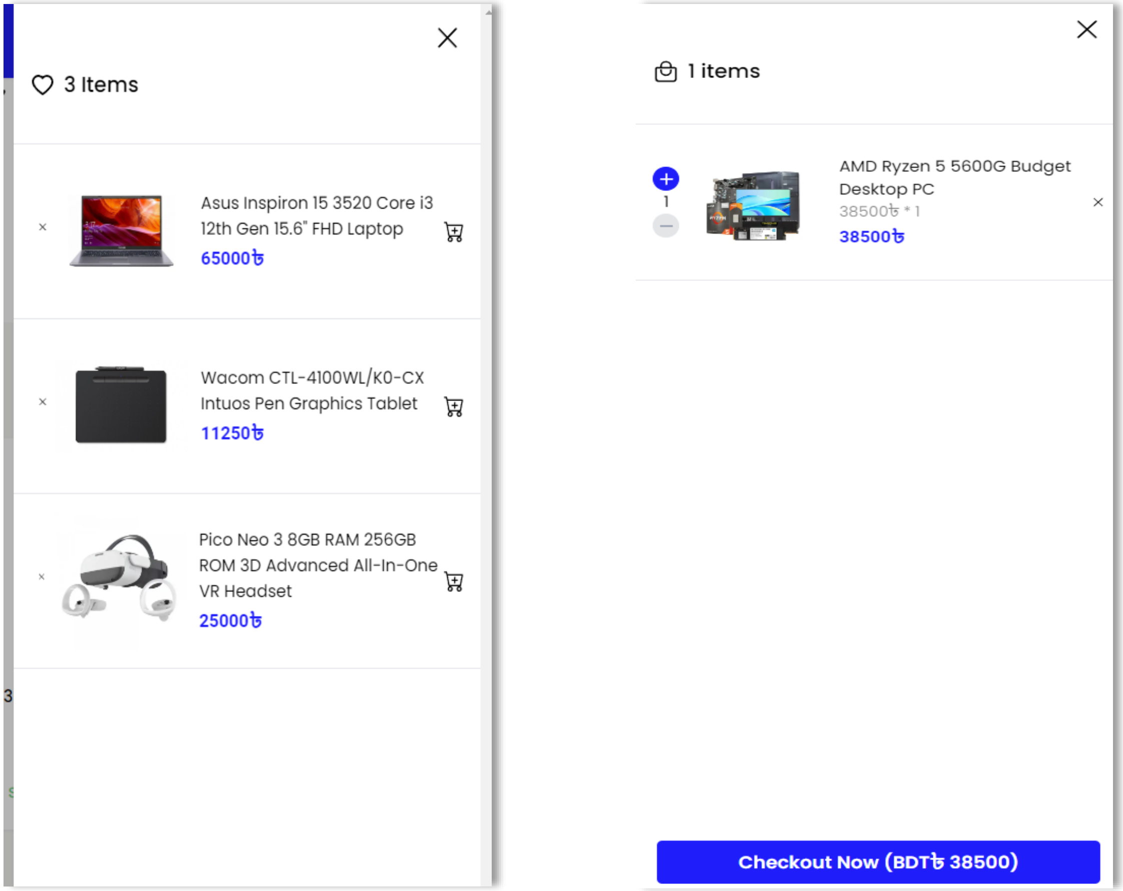
Empower Your Creativity with Our Multimedia Marketplace

Welcome to our online marketplace, where you can buy and sell a wide range of multimedia products, all in one place. Our platform is designed to make it easy and convenient for you to find the products you need to bring your creative vision to life.

Figure 4.2.6: Search Bar

4.2.7 Wishlist and Cart

Adding a "Wishlist" option makes the user's experience even more unique and easy. Buyers can make a list of products they want to buy, which makes it easier for them to make a choice afterwards. The "Add to Cart" feature makes shopping easy by letting customers put together a list of products before going to the checkout page.



♡ 3 Items

Asus Inspiron 15 3520 Core i3 12th Gen 15.6" FHD Laptop
65000৳

Wacom CTL-4100WL/K0-CX Intuos Pen Graphics Tablet
11250৳

Pico Neo 3 8GB RAM 256GB ROM 3D Advanced All-In-One VR Headset
25000৳

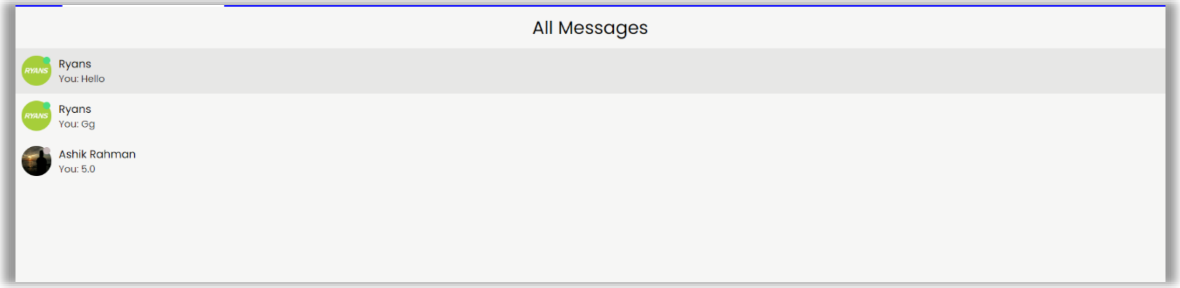
1 items

AMD Ryzen 5 5600G Budget Desktop PC
38500৳ * 1
38500৳

Figure 4.2.7: Wishlist & Cart

4.2.8 Live Chat

The addition of a live chat function improved the level of engagement of users and real-time assistance provided by the “Multimedia Mart” platform. Any user can send message and have a conversion with the seller. Thus, this pre-purchase talk and a post-purchase discussion between buyers and sellers can reduce concerns and boost confidence accordingly.



All Messages

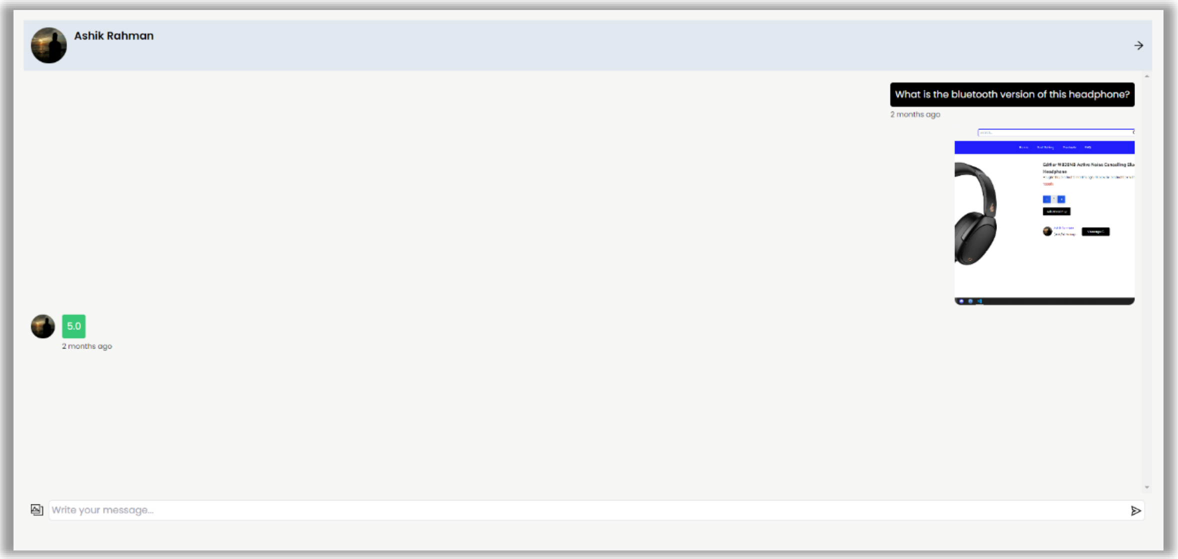
Ryans
You: Hello

Ryans
You: Gg

Ashik Rahman
You: 5.0

Figure 4.2.8.a: All Messages

As the image above, a user can see all their messages from various users in “All Messages” section. Below, we can see one to one conversation between users. They can send image files in their inbox thus helping the process so much easier.



Ashik Rahman

What is the bluetooth version of this headphone?
2 months ago

5.0
2 months ago

Write your message...

Figure 4.2.8.b: Live Conversation

4.2.9 Product Reviews and Ratings

User reviews and ratings can be a way to show that a product is authentic. This feature makes products more obvious, shows potential buyers more about it, and help sellers to get more successful.

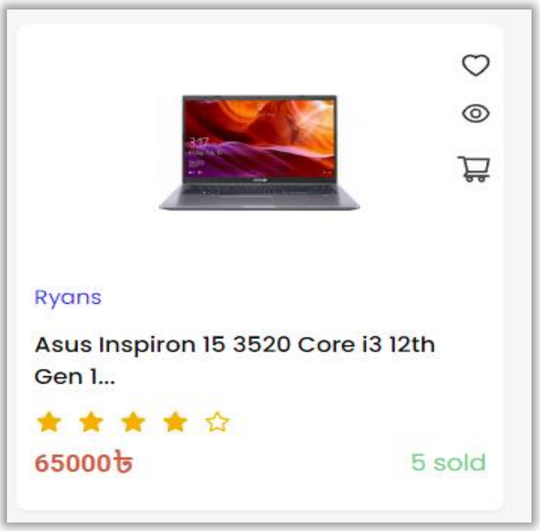
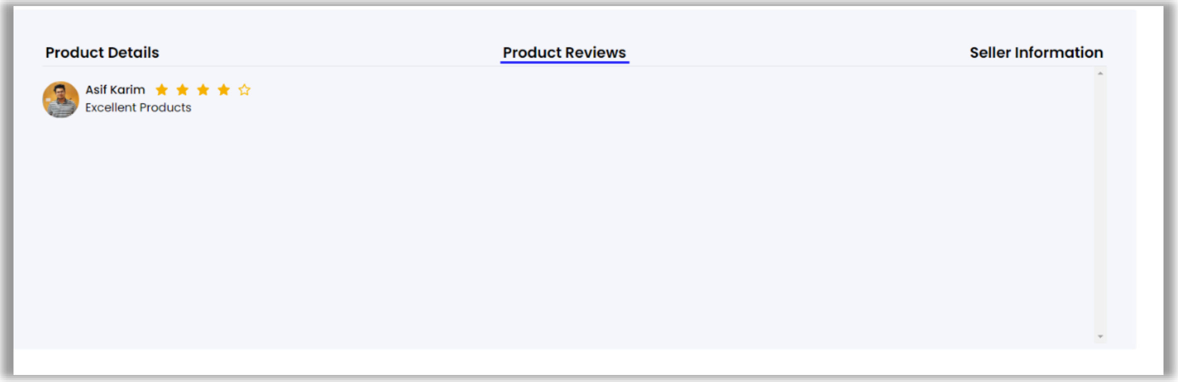


Figure 4.2.9.a: Product Rating

After buying, the user’s work continues with the chance to review and give ratings to the items they purchased. This not only helps potential buyers to make smart choices, but it also makes the platform feel more like a community. Reviews and ratings that are based on real experiences can help sellers improve and build trust among users.

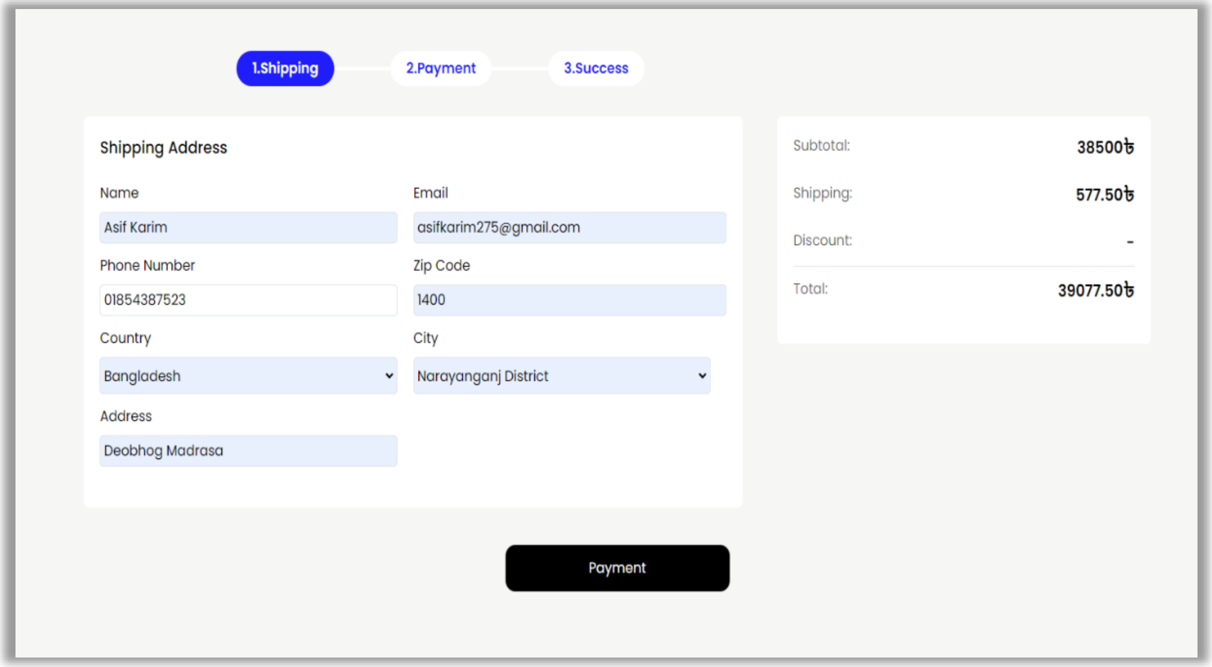


Asif Karim ★★★★☆
Excellent Products

Figure 4.2.9.b: Product Reviews

4.2.10 Payment Systems

In order to provide services to a larger audience “Multimedia Mart” has two different payment options. Credit card payments enabled safe and quick online purchases for those used to using digital payment methods. Cash on delivery, on the other hand provided to users who would rather pay when they receive the products. The platform's accessibility and diversity are enhanced by this double approach to payment methods, which also provided to user’s preferences and saved their time.



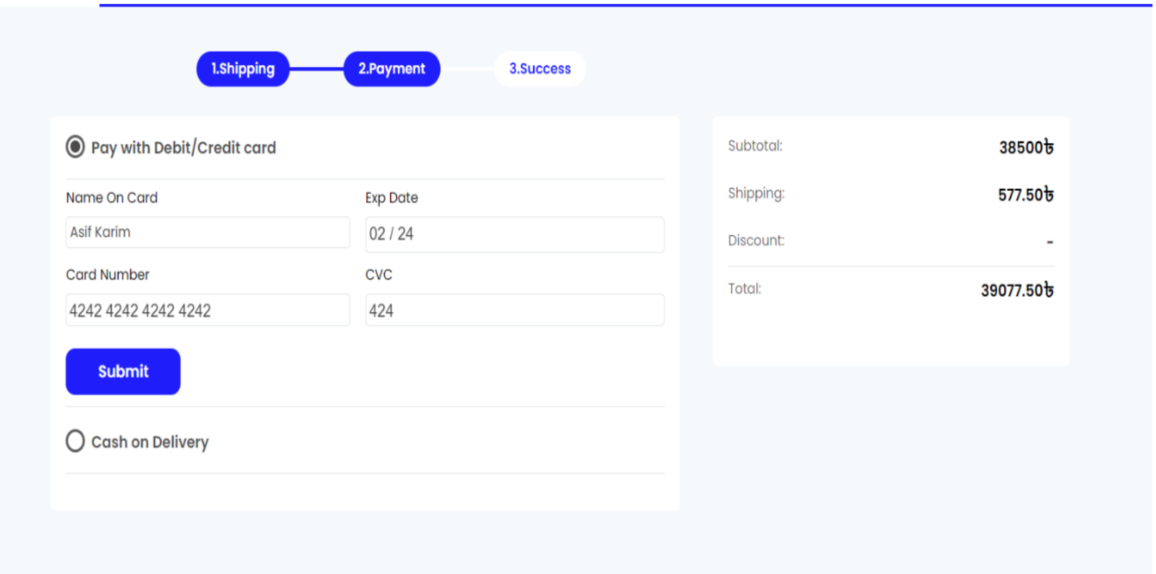
Shipping Address

Name	Email
Asif Karim	asifkarim275@gmail.com
Phone Number	Zip Code
01854387523	1400
Country	City
Bangladesh	Narayanganj District
Address	
Deobhog Madrasa	

Subtotal:	38500৳
Shipping:	577.50৳
Discount:	-
Total:	39077.50৳

Figure 4.2.10.a: Shipping Option

This is the first step of the payment sections. A customer should input their correct information to advance to the next step of payment section.



● Pay with Debit/Credit card

Name On Card: Asif Karim

Exp Date: 02 / 24

Card Number: 4242 4242 4242 4242

CVC: 424

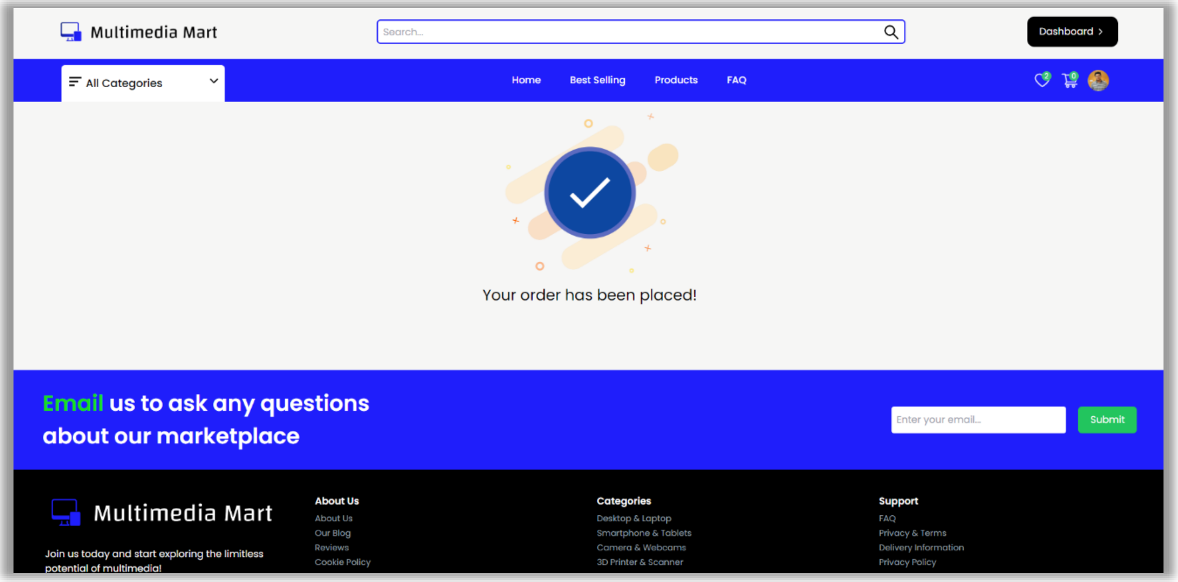
Submit

○ Cash on Delivery

Subtotal:	38500৳
Shipping:	577.50৳
Discount:	-
Total:	39077.50৳

Figure 4.2.10.b: Payment Option

As we can see image above, the customer can choose between two payment options. After submitting their correct credentials, the customer can click the submit button and it will redirect to success page as the below, which will give the confirmation if their order.



Your order has been placed!

Figure 4.2.10.c: Order Confirmation

4.3 Workflow

4.3.1 Planning

The journey began with an extensive phase of project planning and requirements gathering. I took initiative to define the project's scope, objectives, target audience, and key features. The goal was to create a comprehensive list of functionalities, user interactions, and technical specifications that would serve as a blueprint for the development process.

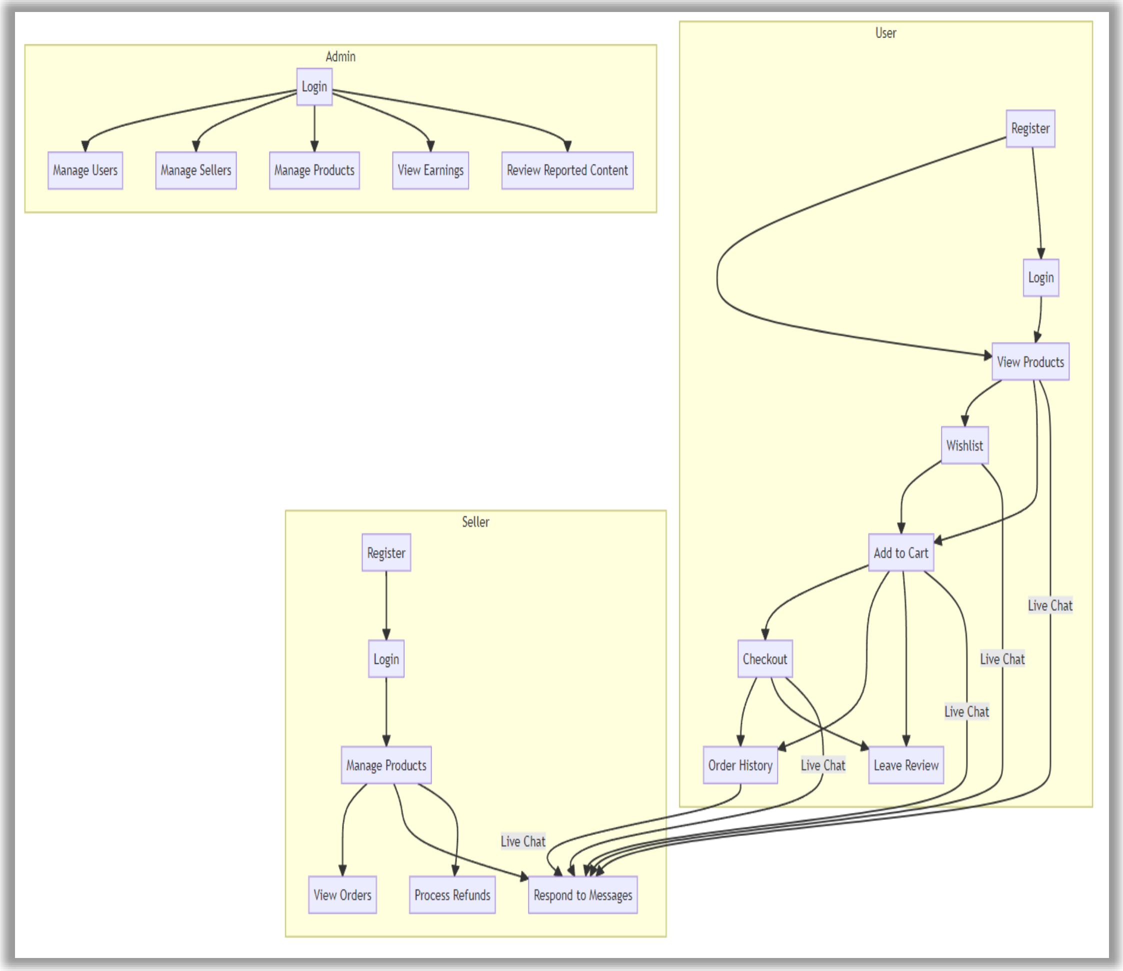


Figure 4.3.1: Roles Interaction

This diagram represents the flow of actions and interactions between different roles of this website including users, sellers, and administrators.

4.3.2 Design

The design part started once it was clear what were the requirements. UI (user interface) and UX (user experience) were very important, because the success of the platform depended on how easy it was to use and how nice it looked. I used figma to create wireframes to visualize the layout, navigation and overall look of the website. These wireframes acted as a foundation for the frontend development.

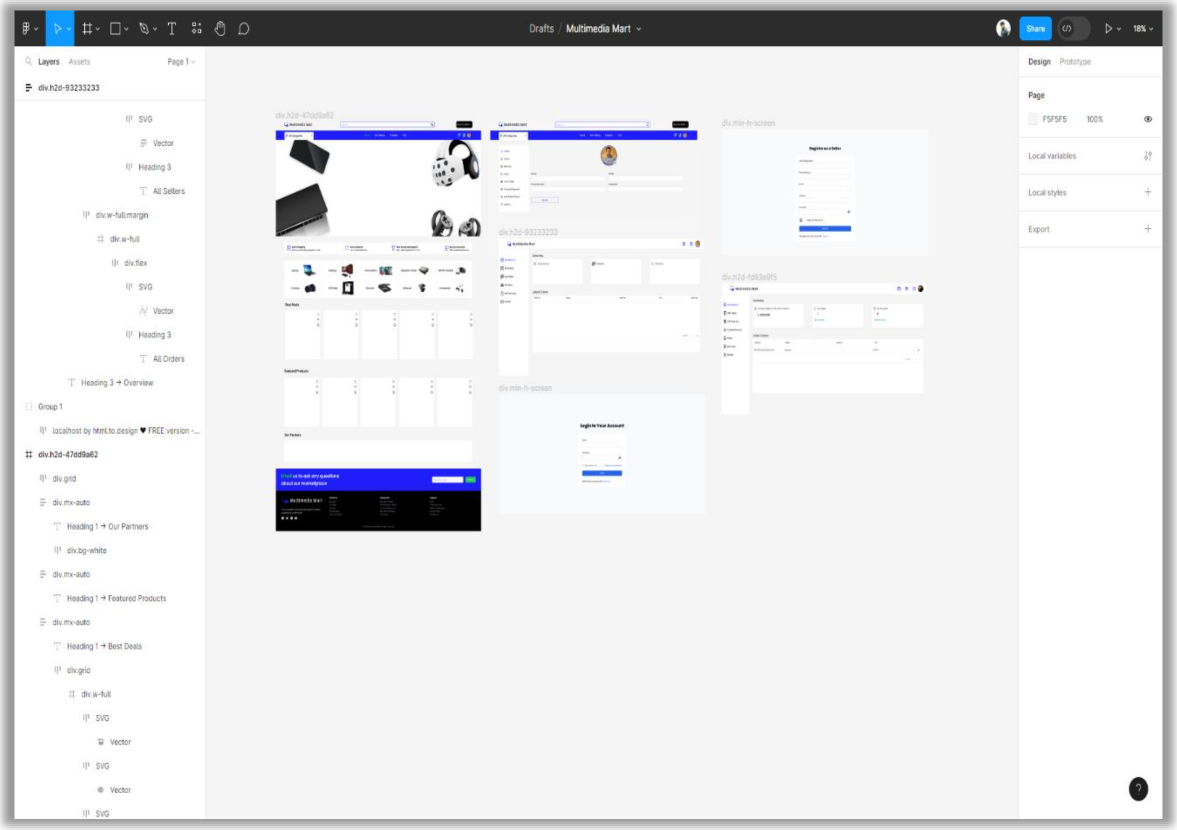


Figure 4.3.2: Website Wireframes

4.3.3 Technology Selection

Based on the project's requirements, the MERN stack (MongoDB, Express.js, React, and Node.js) was chosen as the technology stack.

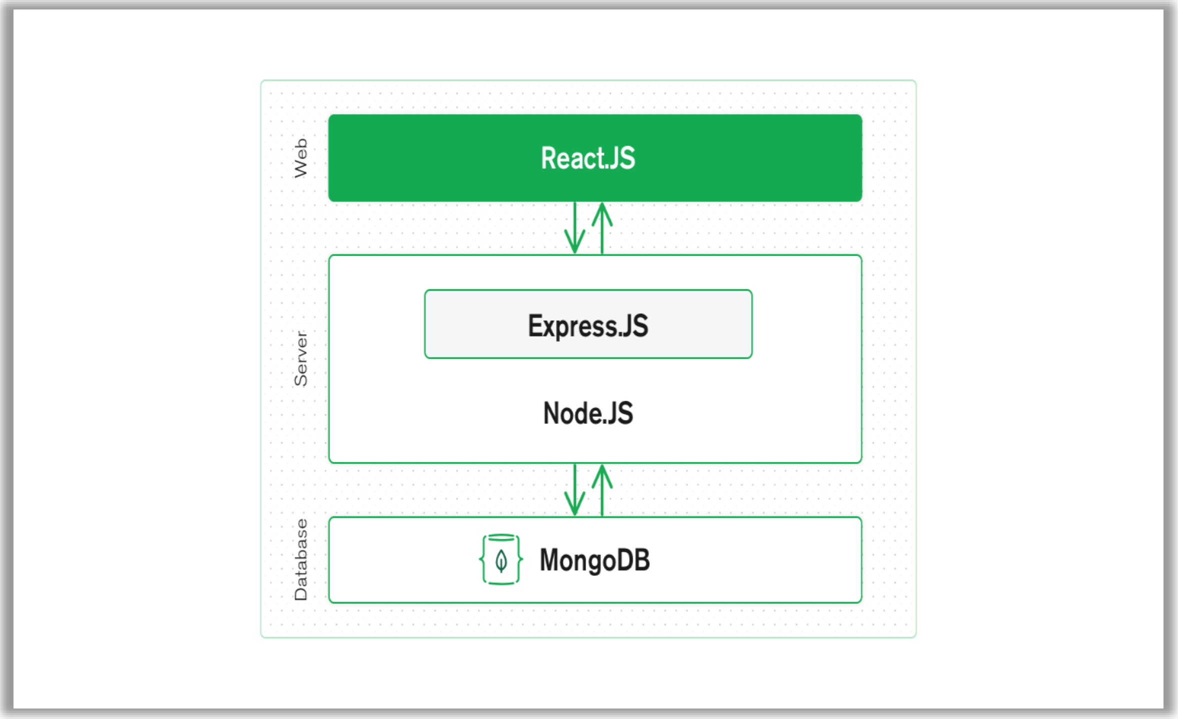


Figure 4.3.3: MERN Stack

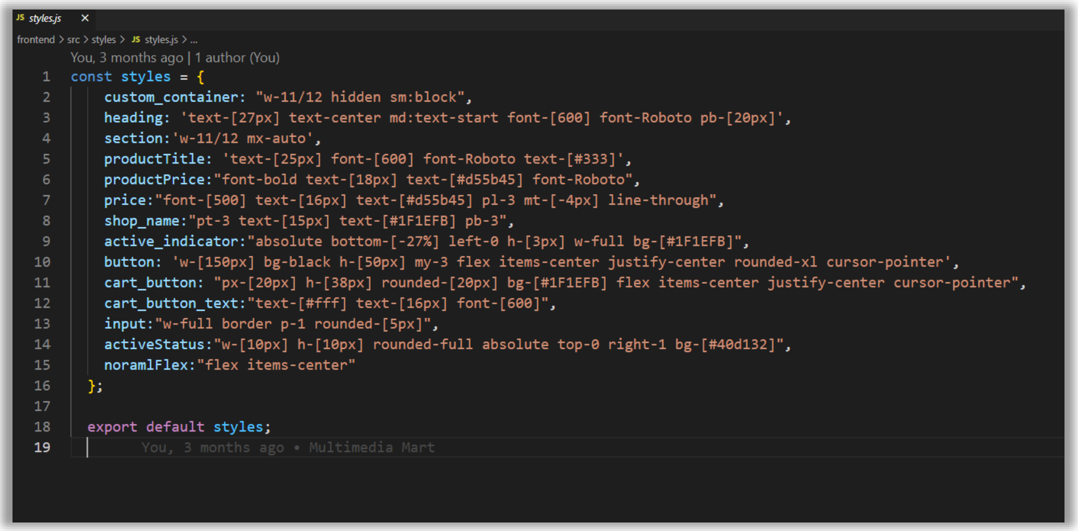
The choice to use MERN stack was based on the need to make an efficient, flexible, and interactive website. MongoDB was used as the database to store data about products, users, and transactions. Express.js made it easy to set up the server, the routes, and the API. React was chosen because of its flexible user interface, which makes interactions smooth. Node.js ran the backend server and handled requests, user identification, and real-time features.

4.3.4 Frontend Development

The frontend development process of the "Multimedia Mart" was a key part of how users interacted with the platform and how it worked as a whole. React is a well-known library written in JavaScript for making user interfaces. It is used by many businesses developers to make websites that are dynamic and interacting. Here's the break down the of frontend development process of the "Multimedia Mart" website using various key points:

1. **UI/UX Design Implementation**

   This is done by using a combination of HTML and CSS to organize and style the elements. This includes putting things like headers, navigation choices, product cards, buttons, forms, and more in a way that makes them easy to use and shows information clearly.



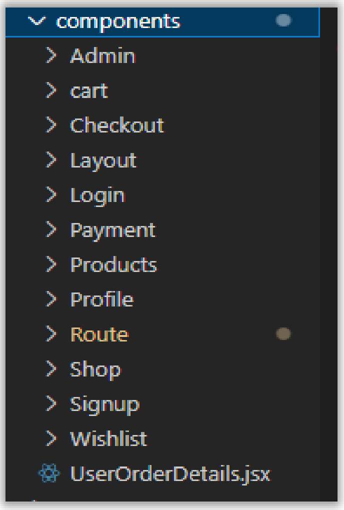
styles.js
frontend > src > styles > styles.js > ...
You, 3 months ago | 1 author (You)

```
const styles = {
    custom_container: "w-11/12 hidden sm:block",
    heading: 'text-[27px] text-center md:text-start font-[600] font-Roboto pb-[20px]',
    section:'w-11/12 mx-auto',
    productTitle: 'text-[25px] font-[600] font-Roboto text-[#333]',
    productPrice:"font-bold text-[18px] text-[#d55b45] font-Roboto",
    price:"font-[500] text-[16px] text-[#d55b45] pl-3 mt-[-4px] line-through",
    shop_name:"pt-3 text-[15px] text-[#1F1EFB] pb-3",
    active_indicator:"absolute bottom-[-27%] left-0 h-[3px] w-full bg-[#1F1EFB]",
    button: 'w-[150px] bg-black h-[50px] my-3 flex items-center justify-center rounded-xl cursor-pointer',
    cart_button: "px-[20px] h-[38px] rounded-[20px] bg-[#1F1EFB] flex items-center justify-center cursor-pointer",
    cart_button_text:"text-[#fff] text-[16px] font-[600]",
    input:"w-full border p-1 rounded-[5px]",
    activeStatus:"w-[10px] h-[10px] rounded-full absolute top-0 right-1 bg-[#40d132]",
    noramlFlex:"flex items-center"
  };

  export default styles;
        You, 3 months ago • Multimedia Mart
```

Figure 4.3.4.a: CSS Styles

2. **React Component Creation**

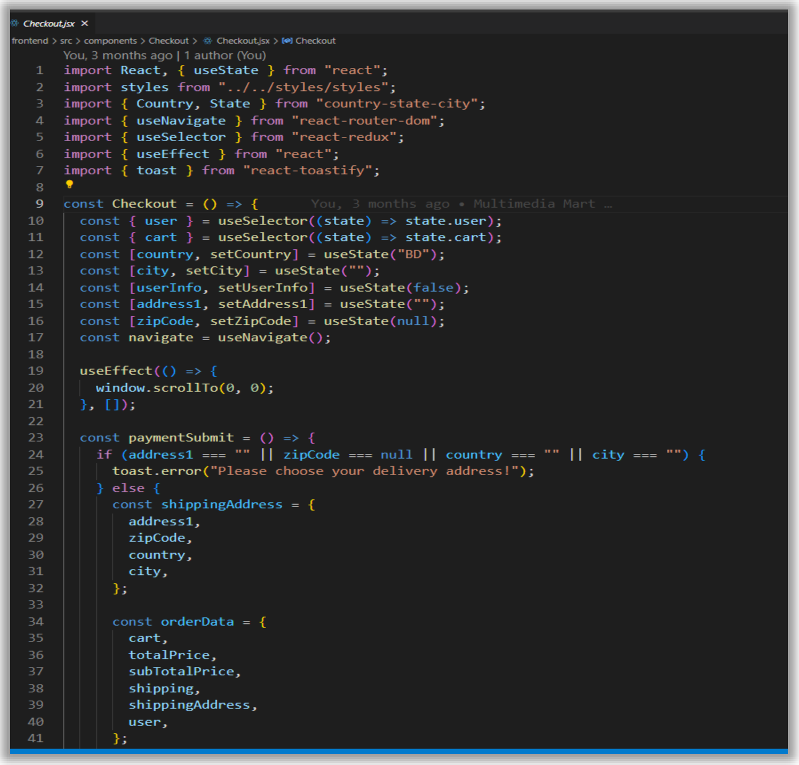
   Frontend development revolves around React components, which form the "Multimedia Mart" website's user interface. Organizing and managing website elements is easy with React's component-based architecture. Headers, product cards, and checkout forms are contained in each component.



components
- Admin
- cart
- Checkout
- Layout
- Login
- Payment
- Products
- Profile
- Route
- Shop
- Signup
- Wishlist
- UserOrderDetails.jsx

Figure 4.3.4.b.: React Components Setup

In the image above we can see basic setup of all the components of “Multimedia Mart” website. This is a very essential structure for frontend development. It makes coding easy and saves time immensely. And in the below I’ve given an example of a react component and how it looks.



Checkout.jsx ×

frontend > src > components > Checkout > Checkout.jsx > Checkout

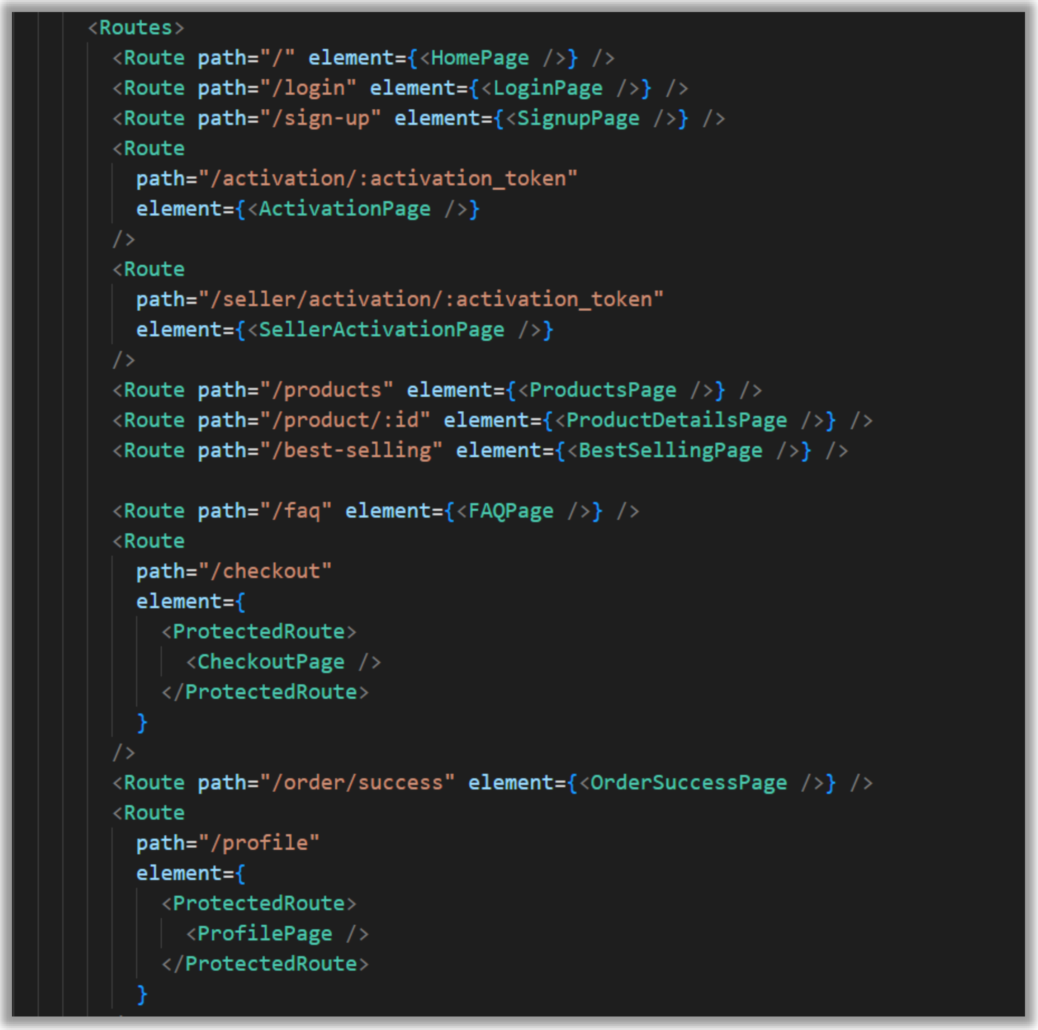
You, 3 months ago | 1 author (You)

```
import React, { useState } from "react";
import styles from "../../styles/styles";
import { Country, State } from "country-state-city";
import { useNavigate } from "react-router-dom";
import { useSelector } from "react-redux";
import { useEffect } from "react";
import { toast } from "react-toastify";

const Checkout = () => {        You, 3 months ago • Multimedia Mart …
  const { user } = useSelector((state) => state.user);
  const { cart } = useSelector((state) => state.cart);
  const [country, setCountry] = useState("BD");
  const [city, setCity] = useState("");
  const [userInfo, setUserInfo] = useState(false);
  const [address1, setAddress1] = useState("");
  const [zipCode, setZipCode] = useState(null);
  const navigate = useNavigate();

  useEffect(() => {
    window.scrollTo(0, 0);
  }, []);

  const paymentSubmit = () => {
    if (address1 === "" || zipCode === null || country === "" || city === "") {
      toast.error("Please choose your delivery address!");
    } else {
      const shippingAddress = {
        address1,
        zipCode,
        country,
        city,
      };

      const orderData = {
        cart,
        totalPrice,
        subTotalPrice,
        shipping,
        shippingAddress,
        user,
      };
```

Figure 4.3.4.c.: React Component

3. Routing and Navigation

React Router is a popular tool for handling client-side routing in React apps. It is used to handle navigation between different parts of the site without having to reload the whole page. By using routes, users can move easily from one part of a website to another, but the site still feels like a standard multi-page application but in reality, it is a single page application.

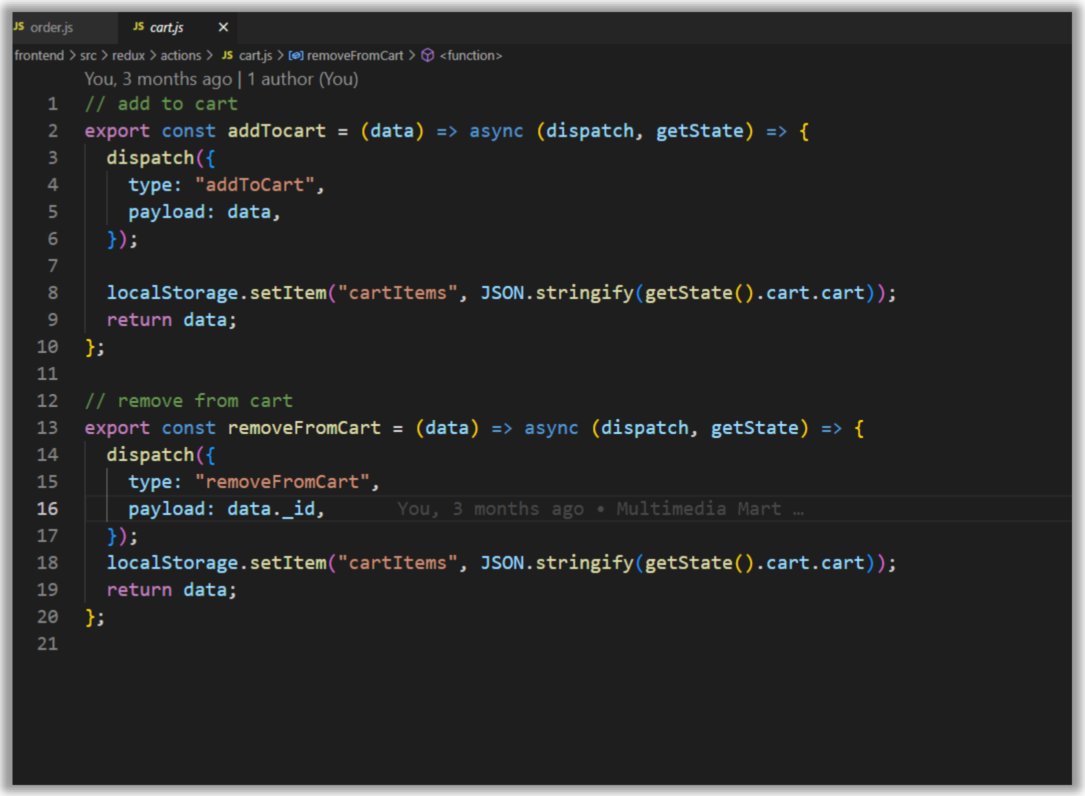


```
<Routes>
  <Route path="/" element={<HomePage />} />
  <Route path="/login" element={<LoginPage />} />
  <Route path="/sign-up" element={<SignupPage />} />
  <Route
    path="/activation/:activation_token"
    element={<ActivationPage />}
  />
  <Route
    path="/seller/activation/:activation_token"
    element={<SellerActivationPage />}
  />
  <Route path="/products" element={<ProductsPage />} />
  <Route path="/product/:id" element={<ProductDetailsPage />} />
  <Route path="/best-selling" element={<BestSellingPage />} />

  <Route path="/faq" element={<FAQPage />} />
  <Route
    path="/checkout"
    element={
      <ProtectedRoute>
        <CheckoutPage />
      </ProtectedRoute>
    }
  />
  <Route path="/order/success" element={<OrderSuccessPage />} />
  <Route
    path="/profile"
    element={
      <ProtectedRoute>
        <ProfilePage />
      </ProtectedRoute>
    }
```

Figure 4.3.4.d.: Routes

4. State Management

State is a term for dynamic data that can change as a user interacts with a site, such as a user's login state, the items in their shopping carts, and their wishlists. Redux was used for the process of maintaining and sharing this data throughout the application.



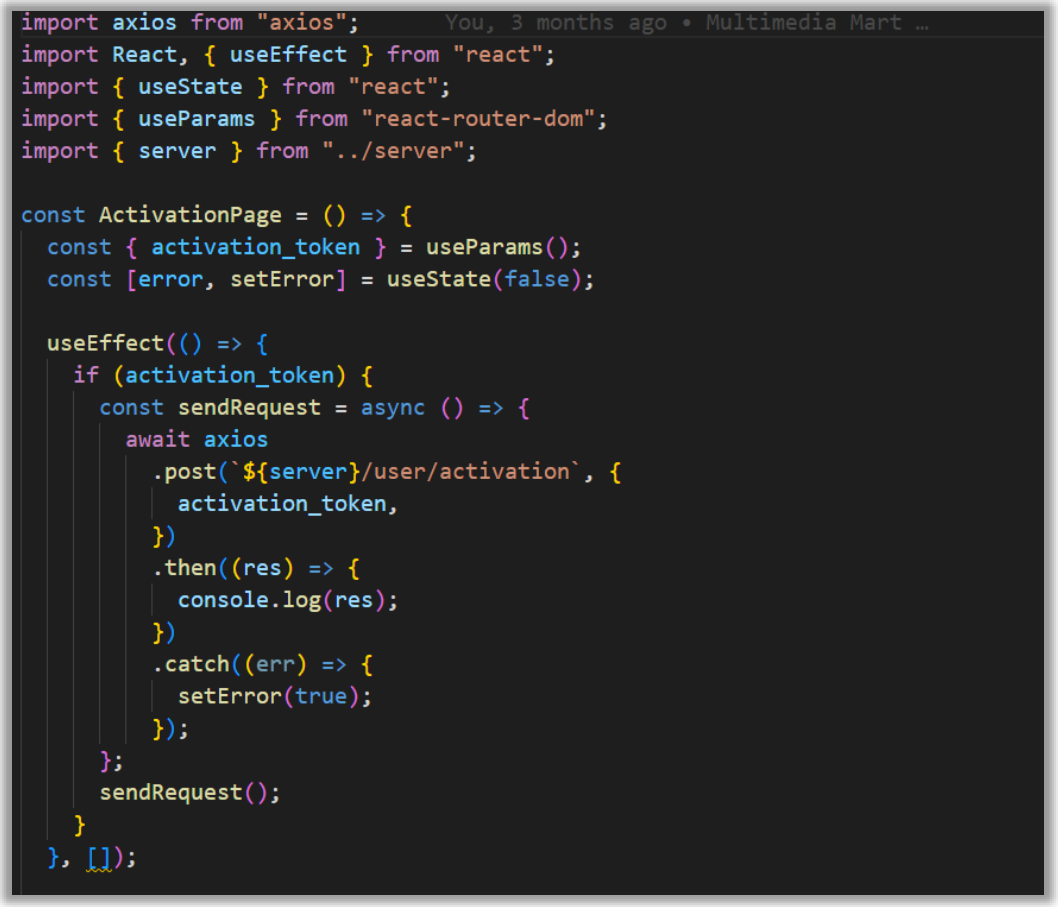
order.js
cart.js ×
frontend > src > redux > actions > cart.js > removeFromCart > <function>
You, 3 months ago | 1 author (You)
You, 3 months ago • Multimedia Mart …

```
// add to cart
export const addTocart = (data) => async (dispatch, getState) => {
  dispatch({
    type: "addToCart",
    payload: data,
  });

  localStorage.setItem("cartItems", JSON.stringify(getState().cart.cart));
  return data;
};

// remove from cart
export const removeFromCart = (data) => async (dispatch, getState) => {
  dispatch({
    type: "removeFromCart",
    payload: data._id,
  });
  localStorage.setItem("cartItems", JSON.stringify(getState().cart.cart));
  return data;
};
```

Figure 4.3.4.e.: State Management

5. Integration of API

The frontend gets info from the backend server using asynchronous HTTP calls and tools like Axios. API integration includes getting different kinds of data, like information about products, user identities, order details, and more. When a person goes to the site, product descriptions are pulled from the database on the backend and shown on the frontend.



```
import axios from "axios";
import React, { useEffect } from "react";
import { useState } from "react";
import { useParams } from "react-router-dom";
import { server } from "../server";

const ActivationPage = () => {
  const { activation_token } = useParams();
  const [error, setError] = useState(false);

  useEffect(() => {
    if (activation_token) {
      const sendRequest = async () => {
        await axios
          .post(`${server}/user/activation`, {
            activation_token,
          })
          .then((res) => {
            console.log(res);
          })
          .catch((err) => {
            setError(true);
          });
      };
      sendRequest();
    }
  }, []);
```

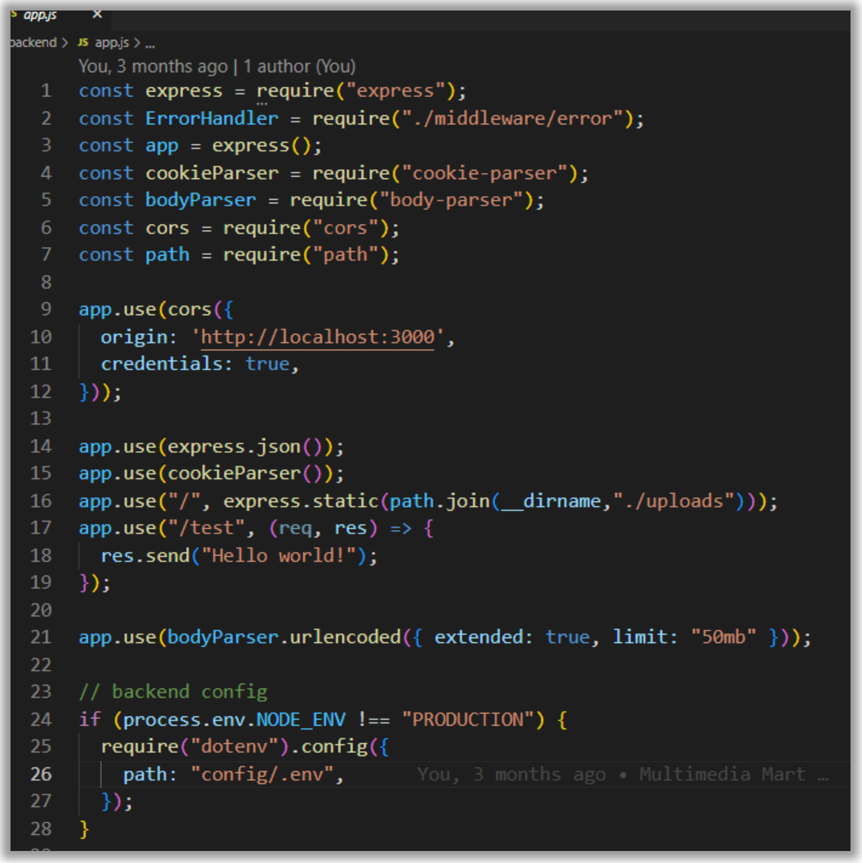
Figure 4.3.4.f.: Integration of API

4.3.5 Backend Development

The backend is the backbone of the platform, responsible for things like user authentication, data storage and retrieval, API endpoints, and component interactions. Data such as user profiles, product information, orders, and reviews might be easily stored in MongoDB. To protect user accounts and financial activities, authentication measures such as JWT-based user verification were deployed. Here's the break down the of backend development process of the "Multimedia Mart" website using various key points:

1. **Server Setup**

   Backend server setup started using Node.js and Express.js, creating a foundation for handling incoming HTTP requests and responses.

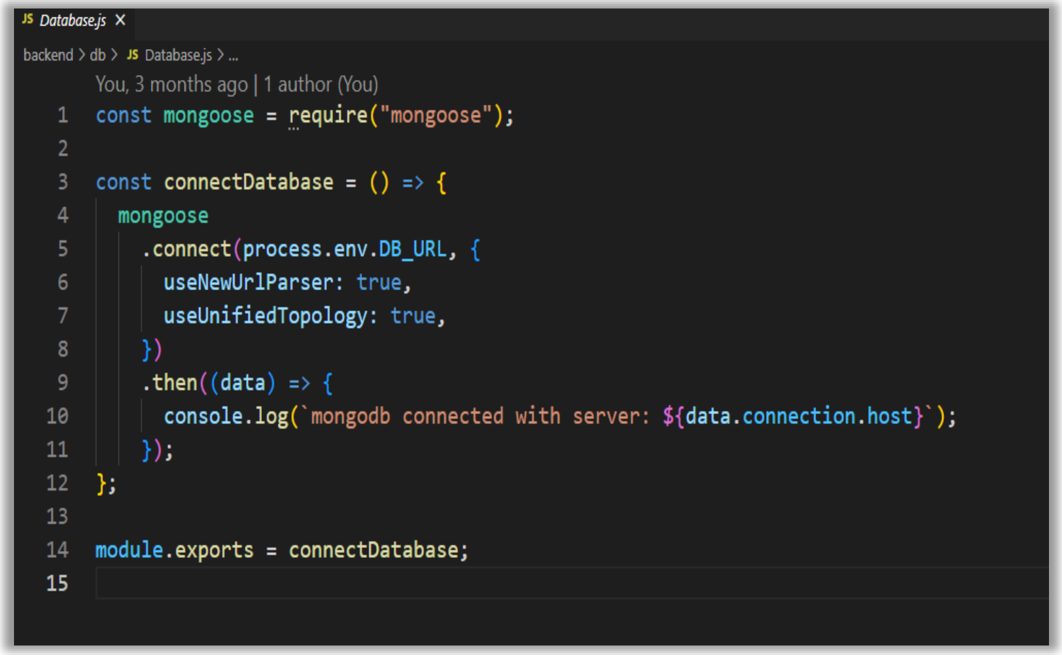

app.js ×
backend > JS app.js > ...
You, 3 months ago | 1 author (You)
You, 3 months ago • Multimedia Mart ...

```javascript
const express = require("express");
const ErrorHandler = require("./middleware/error");
const app = express();
const cookieParser = require("cookie-parser");
const bodyParser = require("body-parser");
const cors = require("cors");
const path = require("path");

app.use(cors({
  origin: 'http://localhost:3000',
  credentials: true,
}));

app.use(express.json());
app.use(cookieParser());
app.use("/", express.static(path.join(__dirname,"./uploads")));
app.use("/test", (req, res) => {
  res.send("Hello world!");
});

app.use(bodyParser.urlencoded({ extended: true, limit: "50mb" }));

// backend config
if (process.env.NODE_ENV !== "PRODUCTION") {
  require("dotenv").config({
    path: "config/.env",
  });
}
```

Figure 4.3.5.a: Server Setup

2. **Database Integration**

MongoDB is selected as the database system to store flexible data related to products, users, orders, reviews and more. To help the program communicate with the MongoDB backend, I used Mongoose, an Object Data Modeling (ODM) module.

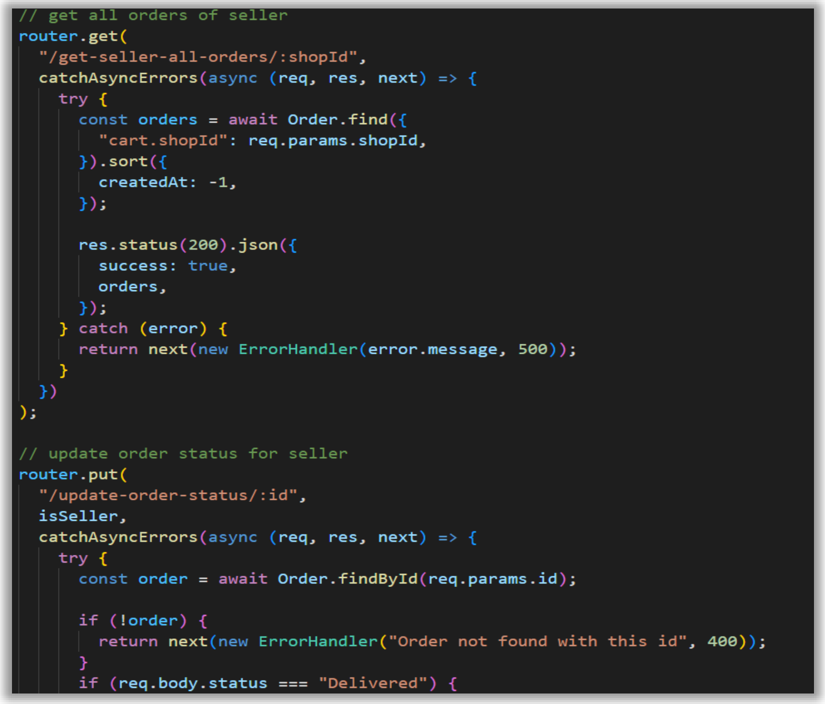

JS Database.js ×
backend > db > JS Database.js > ...
You, 3 months ago | 1 author (You)

```javascript
const mongoose = require("mongoose");

const connectDatabase = () => {
  mongoose
    .connect(process.env.DB_URL, {
      useNewUrlParser: true,
      useUnifiedTopology: true,
    })
    .then((data) => {
      console.log(`mongodb connected with server: ${data.connection.host}`);
    });
};

module.exports = connectDatabase;

```

Figure 4.3.5.b: Connecting MongoDB

3. **API Endpoints Creation**

"Multimedia Mart" website's frontend and backend server are connected by API endpoints. They set up the paths through which the frontend can ask the backend to do certain things or get data. Each API route has a unique URL path and an HTTP method (GET, POST, PUT, or DELETE) that relates to the action.

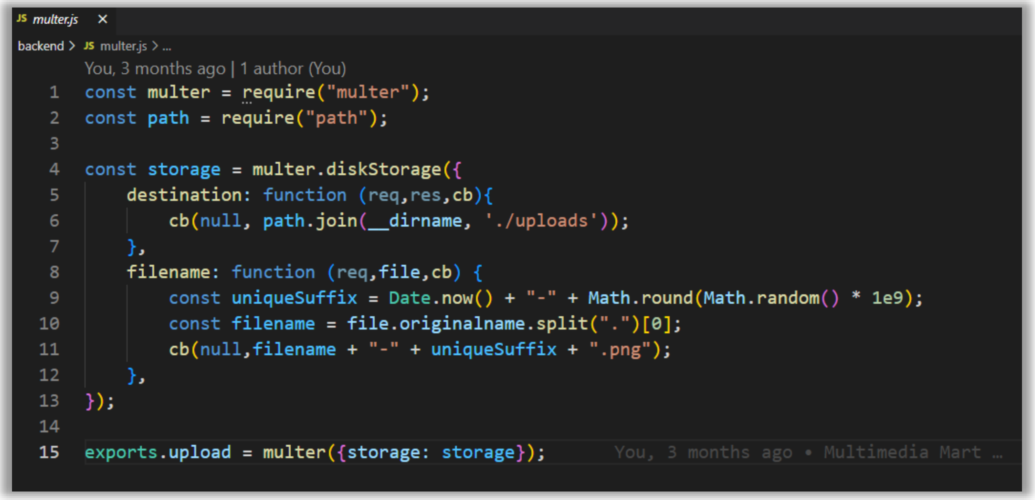


```
// get all orders of seller
router.get(
  "/get-seller-all-orders/:shopId",
  catchAsyncErrors(async (req, res, next) => {
    try {
      const orders = await Order.find({
        "cart.shopId": req.params.shopId,
      }).sort({
        createdAt: -1,
      });

      res.status(200).json({
        success: true,
        orders,
      });
    } catch (error) {
      return next(new ErrorHandler(error.message, 500));
    }
  })
);

// update order status for seller
router.put(
  "/update-order-status/:id",
  isSeller,
  catchAsyncErrors(async (req, res, next) => {
    try {
      const order = await Order.findById(req.params.id);

      if (!order) {
        return next(new ErrorHandler("Order not found with this id", 400));
      }
      if (req.body.status === "Delivered") {
```

Figure 4.3.5.c: API Endpoints Creation

4. File Upload and Storage

Multer, which is middleware for handling file uploads, was used to handle the uploading and saving of product pictures. MongoDB database stores file paths or references so that posted files can be linked to their products or users.


JS multer.js ×
backend > JS multer.js > ...
You, 3 months ago | 1 author (You)

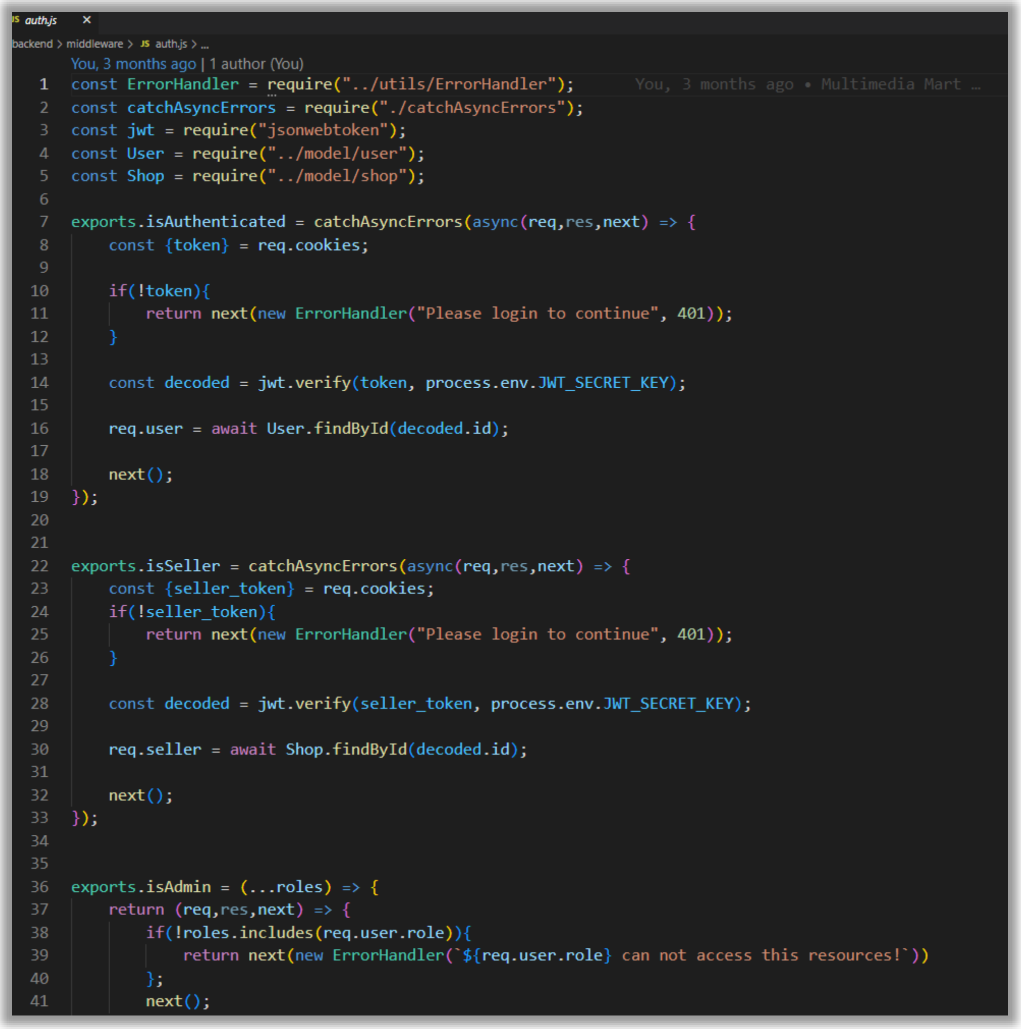
```
const multer = require("multer");
const path = require("path");

const storage = multer.diskStorage({
    destination: function (req,res,cb){
        cb(null, path.join(__dirname, './uploads'));
    },
    filename: function (req,file,cb) {
        const uniqueSuffix = Date.now() + "-" + Math.round(Math.random() * 1e9);
        const filename = file.originalname.split(".")[0];
        cb(null,filename + "-" + uniqueSuffix + ".png");
    },
});

exports.upload = multer({storage: storage});
```

You, 3 months ago • Multimedia Mart ...

Figure 4.3.5.d: Using Multer to Upload

5. User Authentication and Authorization

User authentication is done with JWT (JSON Web Tokens) so that user sessions can be managed safely and logins are safe. User roles and permissions are set up to give users the right access rights, such as letting sellers handle their products or users see their orders.

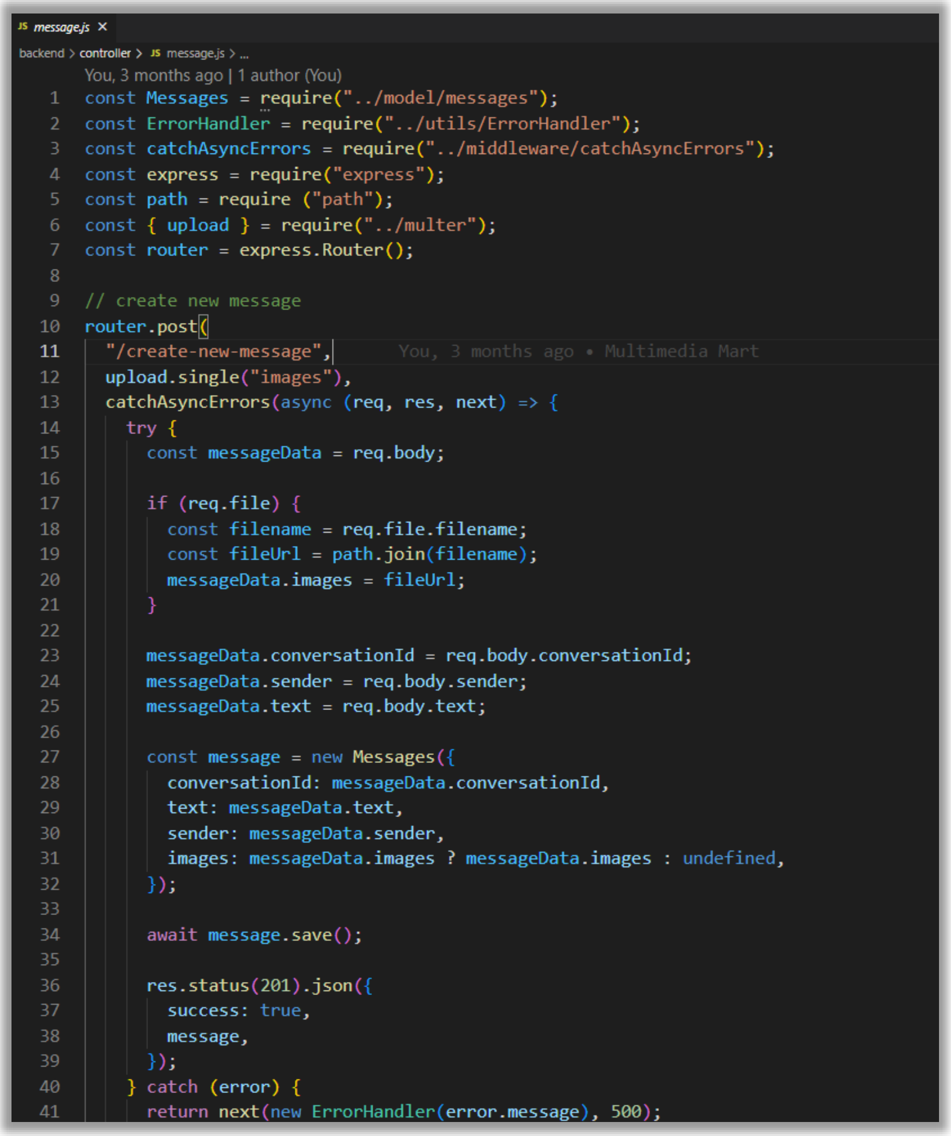

auth.js ×
backend > middleware > auth.js > ...
You, 3 months ago | 1 author (You)
You, 3 months ago • Multimedia Mart ...

```
const ErrorHandler = require("../utils/ErrorHandler");
const catchAsyncErrors = require("./catchAsyncErrors");
const jwt = require("jsonwebtoken");
const User = require("../model/user");
const Shop = require("../model/shop");

exports.isAuthenticated = catchAsyncErrors(async(req,res,next) => {
    const {token} = req.cookies;

    if(!token){
        return next(new ErrorHandler("Please login to continue", 401));
    }

    const decoded = jwt.verify(token, process.env.JWT_SECRET_KEY);

    req.user = await User.findById(decoded.id);

    next();
});


exports.isSeller = catchAsyncErrors(async(req,res,next) => {
    const {seller_token} = req.cookies;
    if(!seller_token){
        return next(new ErrorHandler("Please login to continue", 401));
    }

    const decoded = jwt.verify(seller_token, process.env.JWT_SECRET_KEY);

    req.seller = await Shop.findById(decoded.id);

    next();
});


exports.isAdmin = (...roles) => {
    return (req,res,next) => {
        if(!roles.includes(req.user.role)){
            return next(new ErrorHandler(`${req.user.role} can not access this resources!`))
        };
        next();
```

Figure 4.3.5.e: User Authentication and Authorization

6. Real-time Communication

The backend of the "Multimedia Mart" website has an important feature, that is real-time communication. This makes it easier for users to get involved and gives them engaging ways to talk to each other. I used Socket.io in the backend to connect the frontend to the server through WebSocket.


message.js ×
backend > controller > message.js > ...
You, 3 months ago | 1 author (You)
You, 3 months ago • Multimedia Mart

```javascript
const Messages = require("../model/messages");
const ErrorHandler = require("../utils/ErrorHandler");
const catchAsyncErrors = require("../middleware/catchAsyncErrors");
const express = require("express");
const path = require ("path");
const { upload } = require("../multer");
const router = express.Router();

// create new message
router.post(
  "/create-new-message",
  upload.single("images"),
  catchAsyncErrors(async (req, res, next) => {
    try {
      const messageData = req.body;

      if (req.file) {
        const filename = req.file.filename;
        const fileUrl = path.join(filename);
        messageData.images = fileUrl;
      }

      messageData.conversationId = req.body.conversationId;
      messageData.sender = req.body.sender;
      messageData.text = req.body.text;

      const message = new Messages({
        conversationId: messageData.conversationId,
        text: messageData.text,
        sender: messageData.sender,
        images: messageData.images ? messageData.images : undefined,
      });

      await message.save();

      res.status(201).json({
        success: true,
        message,
      });
    } catch (error) {
      return next(new ErrorHandler(error.message), 500);
```

Figure 4.3.5.f: Setting Up Socket.io

CHAPTER 5
FINDINGS & OBERSERVATIONS

Completing this project has been one of the biggest challenges I've faced in my academic journey. Developing "Multimedia Mart" website as a solo project was far from easy. Throughout the development phase, I encountered numerous difficulties, particularly when it came to developing the login and registration pages with the essential JWT (JSON Web Token) authentication feature. It was also challenging to develop the live chat feature. Implementing a real-time chat feature demanded the integration of messaging functionality within the website, considering user privacy and data protection.
Integrating two different payment options was also a challenge. Designing and developing distinct dashboards for users, sellers, and administrators required thoughtful UX/UI design. Ensuring that each type of user had access to relevant information and features was a complex task & I learned so much from that. Last but not least one of the primary challenges was ensuring the security and authenticity of user accounts. Implementing a verification system via email confirmation proved to be a critical step in preventing fraud registrations.

In this project I tried to covered the gaps that many e-commerce platforms have been missing in previous or existing time. For example, many e-commerce platforms tend to focus on a limited set of product categories, which may exclude various markets or fail to cater to the diverse needs of consumers. "Multimedia Mart" recognized the importance of offering a comprehensive range of multimedia-related products. It includes categories such as laptops, desktops, smartphones, graphics tablets, AR/VR headsets, cameras, 3D printers, scanners, webcams, and accessories. This wide range of offerings distinguishes "Multimedia Mart" from competitors by serving as a one-stop destination for individuals interested in multimedia technology. Traditionally, some e-commerce platforms have favored established businesses, making it challenging for individual sellers to participate in the marketplace. "Multimedia Mart" took a more inclusive approach by welcoming both individual sellers and well-known shops. This inclusivity encourages a diverse and dynamic marketplace where users can choose from products offered by various sellers. By providing a platform for individual sellers, "Multimedia Mart" fosters competition and innovation, resulting in a wider variety of products and competitive pricing. Also, one notable gap in many e-commerce platforms is the lack of direct, real-time communication between buyers and sellers. "Multimedia Mart" addressed this gap by implementing a live chat feature. This feature allows customers to engage with sellers, ask questions, and seek clarification about products before making a purchase. Moreover, it fosters trust and transparency in the buying process, as users can receive immediate responses to their inquiries.

Although "Multimedia Mart" represents a significant achievement, there is always room for improvement. Such as, enhancing recommendation and data sorting. By continuing to better the product recommendation and sorting algorithms can lead to more personalized user experiences and increased sales. Adding more payment options, such as digital wallets and cryptocurrencies, can cater to a wider range of customer preferences. Also expanding the platform's reach to international markets can lead to increased revenue and a larger user base.

Looking forward, my plans for the website are far from complete. I aspire to introduce additional features such as AR/VR product viewing [17], vouchers, events, and AI-Powered Customer Support in the future. While I recognize that the outcome might have been even better had I worked on this project as part of a group, I'm not disheartened by the results I've achieved. My initial vision was to create a multimedia-oriented website capable of relating with the features of modern e-commerce platforms. I believe this project demonstrates that I've succeeded in realizing that vision & through this project, I hope to further establish my capabilities as a MERN stack developer.

CHAPTER 6
CONCLUSION

The development of "Multimedia Mart," a MERN stack website dedicated to selling multimedia-related products, has been a significant achievement in my academic career. During the planning, creation, and development of "Multimedia Mart," several key features were added to make sure that users, from students to professionals, had a smooth and enriching experience. This makes "Multimedia Mart" a versatile and user-centric platform that caters to a wide range of people and their multimedia needs.

One of the primary goals of the project was to create a thriving marketplace for multimedia products, and "Multimedia Mart" has certainly achieved this objective. The website excels in meeting the requirements of both customers and sellers with its thoughtful and user-focused design. To ensure the safety and trustworthiness of the platform, robust user authentication and seller verification processes were implemented. This not only protects the interests of users but also creates an open and secure environment for transactions. "Multimedia Mart" stands out not only for its security measures but also for the diversity of products it offers. Sellers have the freedom to list both used and new multimedia items, making the platform accessible to consumers with various preferences and budgets. This extensive product selection caters to a broad audience, from students looking for affordable equipment to professionals seeking high-end multimedia tools.

To enhance the user experience further, the website includes sections like "Best Sellers" and "Featured Products." These sections showcase the most popular and newest products, engaging customers and aiding them in making informed purchase decisions. The responsive design of the website ensures that it functions seamlessly on any device, making it convenient for users to shop on the go. Security and trust were paramount in the development of "Multimedia Mart." The inclusion of email-based account activation adds an extra layer of security and builds trust among users and sellers. Additionally, features such as wishlists, a review and ratings system, and a smart search function help consumer navigate the platform efficiently and discover products that align with their needs and preferences. Communication is a vital aspect of any online marketplace, and "Multimedia Mart" recognized this by incorporating a live chat feature. This real-time communication tool enables immediate interaction between buyers and sellers, addressing queries and concerns promptly. Such a feature fosters a sense of community and engagement within the platform. Administrative control is another crucial facet of any successful online marketplace. "Multimedia Mart" has a robust admin dashboard that provides oversight on earnings, products, users, and sellers. Admins also have the authority to remove any inappropriate content, ensuring that the platform maintains its integrity and reputation. This level of control ensures that the platform remains a safe and reliable space for all users.

In conclusion, "Multimedia Mart" represents a significant milestone in my academic journey and my journey in web development. It has been a project that has taught me the value of perseverance and problem-solving. It has ignited my passion for continuous improvement and innovation in the realm of web development. The journey of creating this platform has been a learning experience, not just in terms of technical skills but also in understanding the importance of user-centered design, security, and the complexities of running an online marketplace. "Multimedia Mart" is not just a website; it's a testament to the possibilities of what can be achieved through dedication and creativity in the world of web development.

REFERNCES

1. Kedah, Z. (2023). Use of e-commerce in the world of business. Startupreneur Business Digital (SABDA Journal), 2(1), 51-60.

2. Nguyen, H. (2020). End-to-end E-commerce web application, a modern approach using MERN stack.

3. Bawane, M., Gawande, I., Joshi, V., Nikam, R., & Bachwani, S. A. (2022). A Review on Technologies used in MERN stack. Int J Res Appl Sci Eng Technol, 10(1), 479-488.

4. Membrey, P., Plugge, E., Hawkins, D., & Hawkins, T. (2010). The definitive guide to MongoDB: the noSQL database for cloud and desktop computing. springer.

5. Ward, M. (2013). NoSQL database in the cloud: MongoDB on AWS. Amazon Web Services.

6. Shashi, A. (2023). Data Storage in Google Cloud. In Designing Applications for Google Cloud Platform: Create and Deploy Applications Using Java (pp. 73-117). Berkeley, CA: Apress.

7. Archer, R. (2016). Express. js: Web App Development With Node. js Framework.

8. Gackenheimer, C., & Gackenheimer, C. (2015). What is react?. introduction to react, 1-20.

9. Heller, Martin. "What is Node. js? The JavaScript runtime explained." InfoWorld (2017).

10. [Last accessed on: 15August,2023] https://www.imaginarycloud.com/blog/what-is-mern-stack-and-how-does-it-work/

11. Akter, Rowshan. "E-commerce and its Contribution in Business in Bangladesh." (2023).

12. Santos, V., Augusto, T., Vieira, J., Bacalhau, L., Sousa, B. M., & Pontes, D. (2023). E-Commerce: Issues, Opportunities, Challenges, and Trends. Promoting Organizational Performance Through 5G and Agile Marketing, 224-244.

13. [Last accessed on: 17August, 2023] https://www.cloudways.com/blog/top-ecommerce-websites/

14. [Last accessed on: 18August, 2023] https://code.visualstudio.com/learns

15. [Last accessed on: 11August, 2023] https://www.makeuseof.com/what-is-figma-used-for/

16. Mazrae, P. R., Decan, A., Mens, T., & Wessel, M. (2023). A Preliminary Study of GitHub Actions Workflow Changes.

17. Baltierra, S. (2023, January). Virtual Reality and Augmented Reality Applied to E-Commerce: A Literature Review. In Human-Computer Interaction: 8th Iberoamerican Workshop, HCI-COLLAB 2022, Havana, Cuba, October 13–15, 2022, Revised Selected Papers (p. 201). Springer Nature.