PROJECT REPORT

On

ECO CONSCIOUS – Eco Friendly Website for Shopping

Submitted in partial fulfilment of the requirement for the
Course BEE (22CS026) of

COMPUTER SCIENCE AND ENGINEERING
B.E. Batch-2022

in

Dec - 2024

Under the Guidance of
Ms. Meenakshi

Submitted By
Jashanjit Kaur
2210990440
Jasjeet Kaur
2210990442
Kamaljeet Kaur
2210990470
Kanchan Yadav
2210990471
Kashika
2210990493

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
CHITKARA UNIVERSITY
PUNJAB

CERTIFICATE

This is to be certified that the project entitled "**ECO CONSCIOUS - Eco Friendly Website for Shopping**" has been submitted for the Bachelor of Computer Science Engineering at Chitkara University, Punjab during the academic semester July 2024- December 2024 is a Bonafide piece of project work carried out by "Jashanjit Kaur 2210990440, Jasjeet Kaur 2210990442, Kamaljeet Kaur 2210990470, Kanchan Yadav 2210990471, Kashika 2210990493" towards the partial fulfilment for the award of the course Integrated Project (CS 203) under the guidance of Ms. Meenakshi and supervision.

Sign. of Project Guide:

Ms. Meenakshi

CANDIDATE'S DECLARATION

We, **Jashanjit Kaur 2210990440, Jasjeet Kaur 2210990442, Kamaljeet Kaur 2210990470, Kanchan Yadav 2210990471, Kashika 2210990493 of the Group-10**, B.E.-2022 of the Chitkara University, Punjab hereby declare that the Integrated Project Report entitled **"ECO CONSCIOUS - Eco Friendly Website for Shopping"** is an original work and data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other Institute for the award of any other course.

Sign. of Student 1	**Sign. of Student 2**	**Sign. of Student 3**	**Sign. of Student 4**	**Sign. of Student 5**
Jashanjit Kaur	Jasjeet Kaur	Kanchan Yadav	Kamaljeet Kaur	Kashika
2210990440	2210990442	2210990470	2210990471	2210990493

Place: Chitkara University
Date: 11 December, 2024

ACKNOWLEDGEMENT

It is our pleasure to be indebted to various people, who directly or indirectly contributed in the development of this work and who influenced my thinking, behaviour and acts during the course of study.

We express our sincere gratitude to all for providing me an opportunity to undergo Integrated Project as the part of the curriculum.

We are thankful to **Ms. Meenakshi** and **Mr. Shiv Mohan** for her support, cooperation, and motivation provided to us during the training for constant inspiration, presence and blessings.

Lastly, we would like to thank the almighty and our parents for their moral support and friends with whom we shared our day-to day experience and received lots of suggestions that improve our quality of work.

Jashanjit Kaur	Jasjeet Kaur	Kanchan Yadav	Kamaljeet Kaur	Kashika
2210990440	2210990442	2210990470	2210990471	2210990493

The report consists of following chapters:

1. **Abstract/keywords……6**
2. **Introduction ……..7**
    - 2.1 **Background**
    - 2.2 **Problem Statement**
3. **Software and Hardware Requirement Specification…….9**
    - 3.1 **Methods**
    - 3.2 **Software requirements**
    - 3.3 **Hardware Requirements**
4. **Database Analysing, design and implementation……11**
5. **GUI Constructing (Project Snapshots) ……12**
6. **Code-Implementation and Database Connections……...15**
7. **Limitations……...18**
8. **Conclusion……19**
9. **Future Scope……20**
10. **Contributions of all Teammates……..21**
11. **References ……23**

1. Abstract/Keywords

Eco-Conscious is a groundbreaking eco-commerce platform crafted to transform online shopping into a sustainable, environmentally conscious activity. As the global community grapples with the pressing challenges of climate change and environmental degradation, **Eco-Conscious** offers a practical and impactful solution. By enabling users to make **informed shopping decisions**, the platform seamlessly bridges convenience with **eco-responsibility**, encouraging consumers to reduce their **ecological footprint** through simple changes in their **purchasing behavior**.

This innovative platform harnesses the power of the **MERN stack (MongoDB, Express.js, React.js, Node.js)** to deliver a scalable, **robust backend** and an intuitive, **user-centric frontend**. Database management is powered by **MongoDB Atlas and MongoDB Compass**, ensuring seamless data handling. **Deployment** is facilitated through **Render and Vercel**, while development and testing are conducted in **Visual studio Code**. To ensure seamless **collaboration and version control**, the team utilized **GitHub** throughout the project lifecycle.

At the core of **Eco-Conscious** lies the **Eco-Score**, a dynamic metric that evaluates products based on key environmental factors such as carbon **footprint, material sourcing, and recyclability**. Each product page prominently displays its eco-score, empowering users to understand the impact of their choices. The platform features advanced search and filter functionalities, allowing users to **explore products by categories**, **price range, and eco-friendly criteria**. These features not only provide transparency but also simplify the process of discovering sustainable alternatives, ensuring that environmentally friendly options are accessible to everyone.

Eco-Conscious incorporates an array of features to enhance user convenience and personalization. A secure **Signup/Login system with email verification**, powered by **Nodemailer**, ensures user **data protection**. The platform also includes a **Wishlist for saving favourite items**, a **Cart** for streamlined shopping, and an **Order History section for tracking past purchases**. Upon placing an order, users receive instant **email notifications**, confirming their purchase and improving engagement.

The platform is designed with **sustainability** at its core, offering tools to make eco-conscious shopping decisions effortless. The filter functionality is particularly noteworthy, enabling users to sort products based on environmental criteria, such as low carbon footprint, sustainable materials. For those seeking to transition to greener lifestyles, **Eco-Conscious** provides **alternative product suggestions**, **presenting better options with higher eco-scores**.

Each product page is enriched with detailed information, including standard attributes like price and availability, alongside sustainability metrics and eco-friendly certifications. The **dynamic search and filter systems** further enhance the shopping experience, while the platform's **responsive design** ensures seamless access across devices.

2. Introduction

2.1 Background

As global concerns about environmental sustainability intensify, individuals, corporations, and governments are increasingly recognizing the need to adopt eco-friendly practices. Industries such as energy, transportation, and agriculture have begun implementing measures to minimize their environmental impact. However, the retail and e-commerce sectors, despite their exponential growth, lag in providing sustainable solutions to consumers. The ease and convenience of online shopping often mask the environmental damage caused by unsustainable sourcing, packaging, and logistics.

Consumers today are more environmentally aware than ever before. A growing segment of shoppers actively seeks eco-friendly alternatives to reduce their ecological footprint. However, traditional e-commerce platforms offer little support for this growing demand, often failing to provide transparency about product origins, material sourcing, or environmental impact. This provides transparency about product origins, material sourcing, or environmental impact. This creates a significant gap in the market for platforms designed to prioritize sustainability and promote conscious consumerism.

Eco-Conscious emerges as a response to this gap, offering a dedicated platform where users can shop for sustainable products with confidence. Unlike conventional e-commerce sites, Eco-Conscious integrates tools to evaluate and display the environmental impact of products. Using features such as **eco-scores**, **environmental filters**, and **alternative product suggestions**, the platform aims to transform the online shopping experience into an opportunity for responsible consumption.

The project reflects a growing trend toward merging technology with purpose. Leveraging technologies such as **Node.js**, **Express.js**, **MongoDB**, and **React.js**, Eco-Conscious is a scalable, user-centric platform that balances convenience with environmental responsibility. By offering a robust backend and an intuitive frontend, the platform ensures users can make informed decisions without compromising on efficiency. With features such as personalized user profiles, detailed product pages, and secure authentication, Eco-Conscious is more than an e-commerce site—it is a movement toward sustainable living.

The vision behind Eco-Conscious extends beyond simply selling eco-friendly products; it aims to create a movement that shifts consumer behaviour towards sustainability. By incorporating environmental criteria into product discovery and providing users with eco-friendly alternatives, the platform encourages users to make more conscious purchasing decisions. It aligns with the growing trend of ethical consumerism, where individuals increasingly expect brands to align with their environmental values.

2.2 Problem Statement

The rapid growth of e-commerce has undoubtedly transformed the retail industry, offering unparalleled convenience and accessibility. However, this convenience often comes at a significant environmental cost. Many products available on conventional platforms are mass-produced, poorly sourced, and lack transparency regarding their ecological impact. This makes it challenging for consumers to assess the sustainability of their purchases and contributes to a cycle of wasteful consumption.

Traditional e-commerce platforms fail to prioritize environmental criteria, leaving a gap for users who wish to shop responsibly. For example:

- Consumers lack clear indicators of a product's environmental impact, such as its carbon footprint or recyclability.
- The absence of tools to filter and sort products based on eco-friendly attributes makes it difficult to identify sustainable options.
- There is limited or no guidance on finding greener alternatives to conventional products.

Moreover, the lack of integration between user preferences and environmental considerations further complicates the decision-making process. Existing platforms do not adequately address the need for personalized suggestions based on sustainability metrics, leaving eco-conscious consumers underserved.

Eco-Conscious seeks to address these challenges by providing a comprehensive solution. It bridges the gap between convenience and responsibility by offering features such as:

- **Eco-Scoring:** An intuitive metric to evaluate the environmental impact of products.
- **Environmental Filters:** Advanced filtering options based on sustainability factors like carbon footprint and recyclability.
- **Alternative Product Recommendations:** Suggestions for greener alternatives with better eco-scores.

This approach ensures that users can make well-informed choices without sacrificing the ease and efficiency of online shopping. By addressing these pain points, Eco-Conscious aspires to lead the transformation toward a more sustainable e-commerce ecosystem. Its mission is to empower consumers with the tools and knowledge needed to reduce their ecological footprint, fostering a future where shopping aligns with the values of sustainability and environmental responsibility.

By addressing these challenges, Eco-Conscious not only promotes sustainable shopping but also fosters a culture of environmental responsibility. The platform aims to redefine e-commerce by making sustainability a core aspect of the online shopping experience. It seeks to inspire both consumers and businesses to prioritize the planet, creating a ripple effect that extends beyond individual purchases.

3. Software and Hardware Requirement Specification

3.1 Methods

To develop and deploy the **Eco-Conscious** platform, the following methods and practices were employed to ensure an efficient, scalable, and user-friendly application:

Development Process

- **Agile Methodology**: Iterative development cycles with continuous feedback allowed for incremental improvements and flexibility to incorporate user requirements.
- **Modular Development**: Breaking the application into reusable, independent modules for easier scaling and feature integration.

Backend Development

- **Node.js**: Utilized for its non-blocking, event-driven architecture to handle concurrent requests efficiently.
- **Express.js**: Employed as the backend framework for routing, middleware integration, and API creation.
- **Nodemailer**: Implemented for sending email notifications during sign-up and order confirmations.
- **RESTful APIs**: Designed for seamless communication between the frontend and backend.

Database Management

- **MongoDB Atlas**: Hosted database for secure and scalable data management.
- **MongoDB Compass**: Used for local testing and visual database interaction.

Authentication and Security

- **JWT (JSON Web Tokens)**: Ensured secure user authentication for login and session management.
- **Email Verification**: Integrated during user registration to validate accounts.

Frontend Development

- **React.js**: Built a responsive and dynamic user interface using component-based architecture.

Deployment

- **Render**: Deployed the backend services for reliable API hosting.
- **Vercel**: Hosted the React.js frontend for fast and scalable user access.

Tools and Collaboration

- **GitHub**: Used for version control and team collaboration.
- **VS Code**: Adopted as the primary IDE for writing, editing, and debugging code

3.2 Software Requirements

- **Node.js (version 14.x or higher)**: This is essential for running the backend server. Node.js enables the execution of JavaScript on the server side, supporting asynchronous I/O operations and ensuring the backend is responsive and scalable.

- **MongoDB (version 4.x or higher)**: As a NoSQL database, MongoDB stores all data related to products, user profiles, and transactions.

- **Express.js (latest stable version)**: The web application framework used in the backend to manage routing, middleware, and API requests.

- **React.js (latest stable version)**: React.js is used on the frontend for building dynamic user interfaces.

- **JWT (for secure authentication)**: JWT is used to manage user authentication and session management. Ensuring that the latest version of the JWT library is used helps prevent security vulnerabilities and enhances the efficiency of the authentication process.

- **Axios**: Axios is used for handling HTTP requests from the frontend. It simplifies the communication between the client and server, ensuring that data is fetched and displayed in real-time.

- **Git**: As a version control system, Git helps manage and track changes to the codebase. This ensures collaboration is smooth and that all changes are well-documented.

- **Visual Studio Code**: A lightweight, powerful code editor with features like IntelliSense, debugging, Git integration, and support for multiple programming languages, providing a streamlined development environment for both front-end and back-end projects.

3.3 Hardware Requirements

- **Processor**: A processor equivalent to an **Intel Core i5** or higher is recommended for smooth development and testing performance. A fast processor ensures that the development environment runs efficiently and that complex processes like compiling and testing are completed without delays.

- **RAM**: A minimum of **8 GB of RAM** is required for handling the development environment, running multiple applications simultaneously (such as the code editor, browser, and server), and ensuring that the platform performs well during testing.

- **Storage**: At least **100 GB of free storage space** is necessary to store project files, databases, dependencies, and backups.

- **Internet Connection**: A stable internet connection is required to access online resources, download necessary dependencies, test APIs, and deploy the application to cloud services

4. Database Analysing, Design, and Implementation

The Eco Conscious platform uses a custom-designed database implemented in MongoDB. As a NoSQL database, MongoDB was selected for its flexibility, scalability, and ability to handle large datasets efficiently, making it ideal for storing eco-friendly product information, user data, and order transactions.

4.1 Database Analysis

The database design focused on optimizing storage and retrieval of five core data collections:

- Products: Attributes like name, price, category, eco score, and availability.
- Users: User profile.
- Order: order history.
- Wishlist: favourite products.
- Cart: items for later purchase

4.2 Database Design

The database schema was structured using a document-oriented model, with the following main collections:

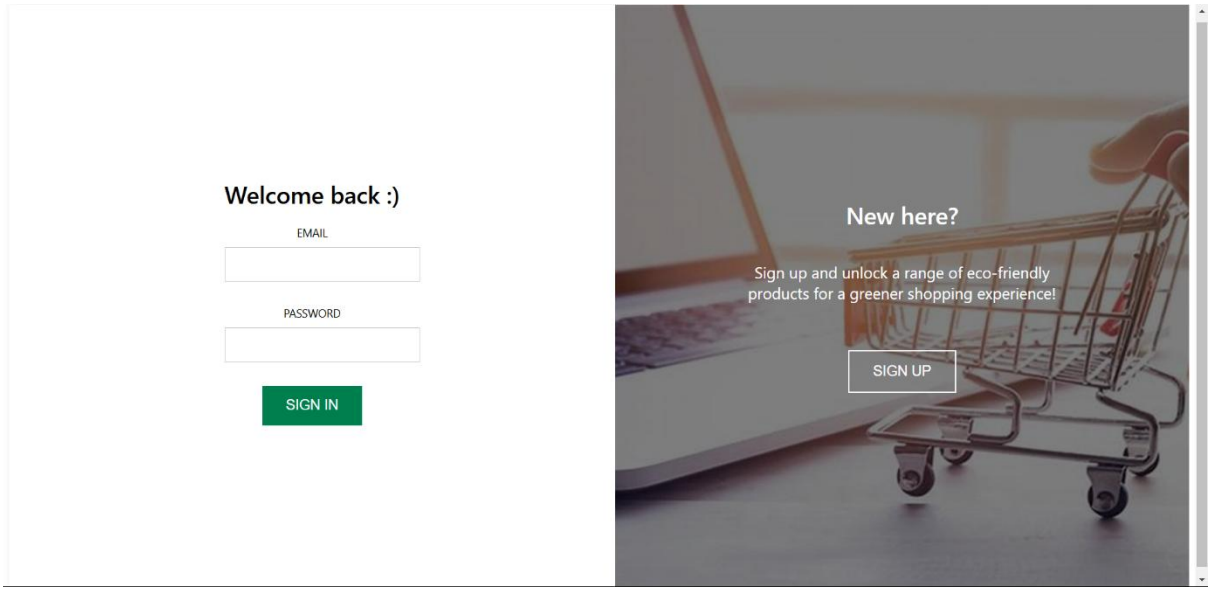
4.3 Database Implementation

- Mongoose: an ODM for MongoDB, was used for defining schemas and performing CRUD operations:
- Connection: The database connects to MongoDB via Mongoose, using MongoDB Atlas.
- Models: Mongoose models define schema for each collection, enforcing validation (e.g., Product, User models).
- CRUD Operations: Implemented for creating new products, updating stock, and managing user orders.

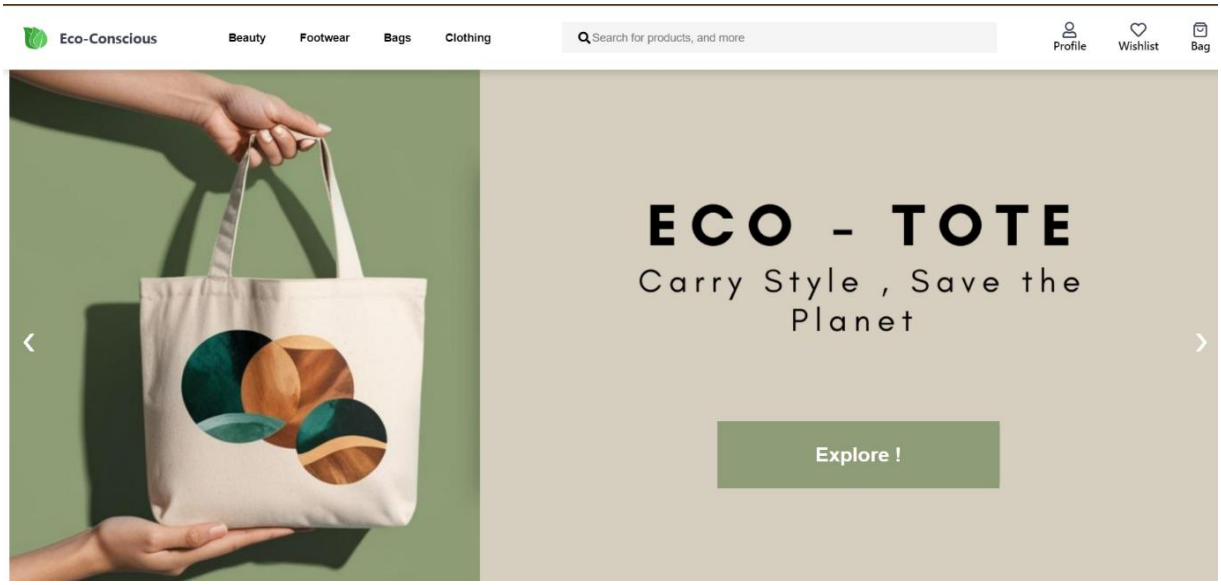
5. GUI Constructing (Project Snapshots)

These are the snapshots of our projects.

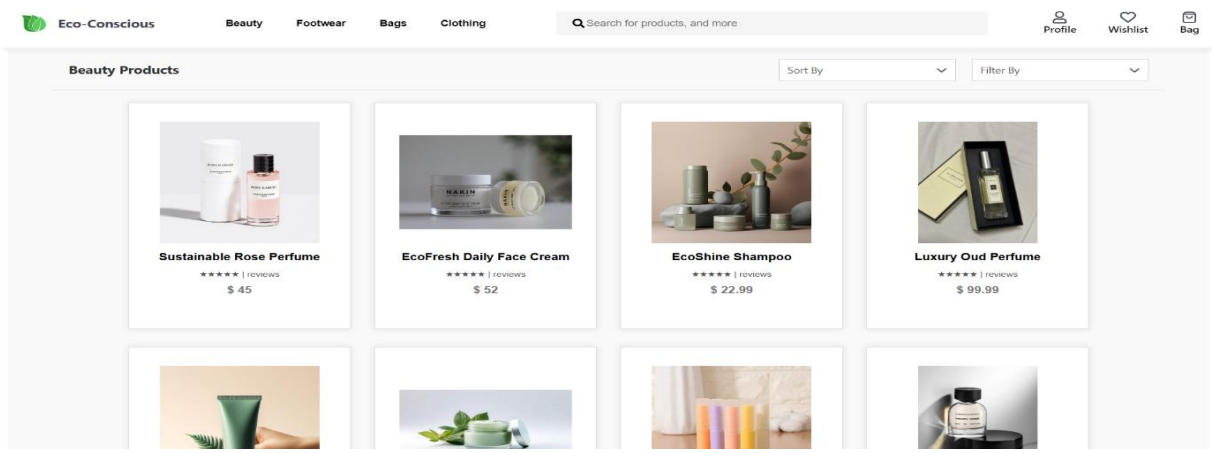
SignIn/SignUp Page -



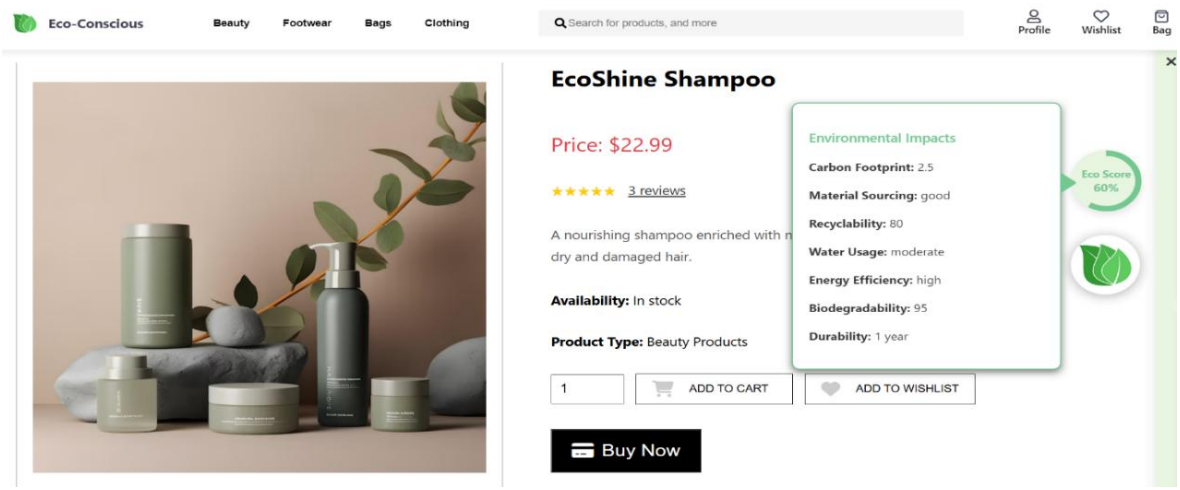
Home Page -



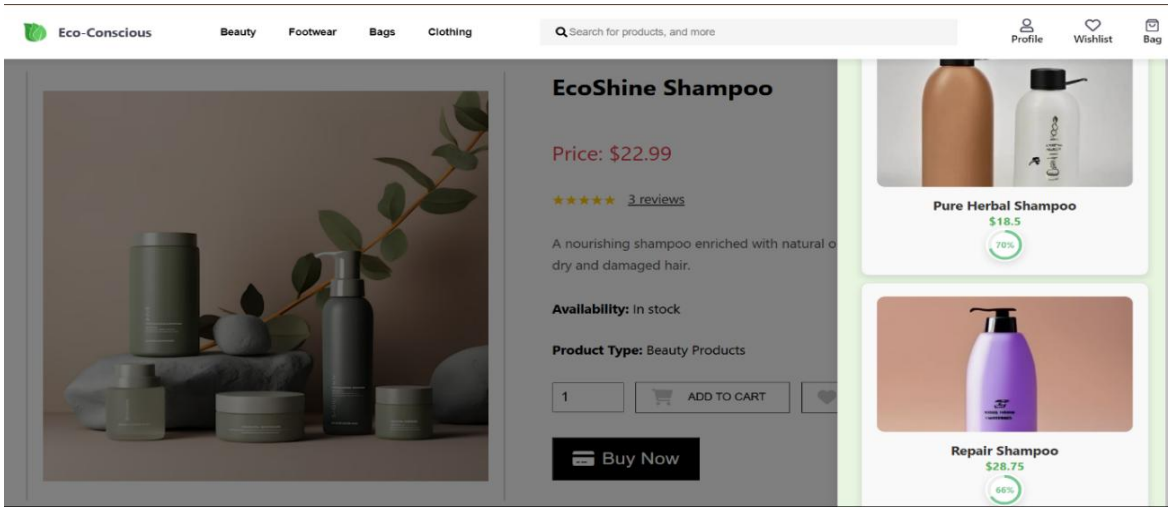
Product List -



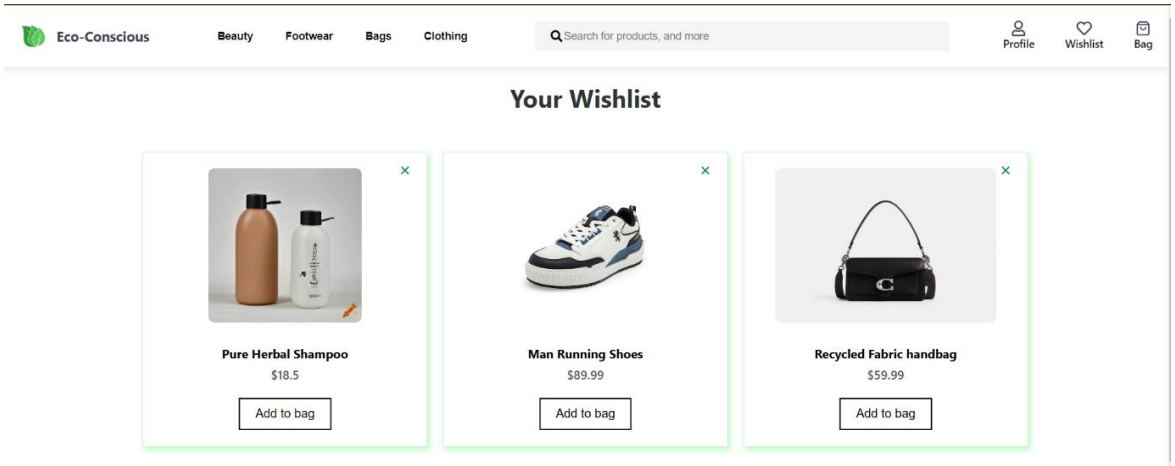
Product Details -



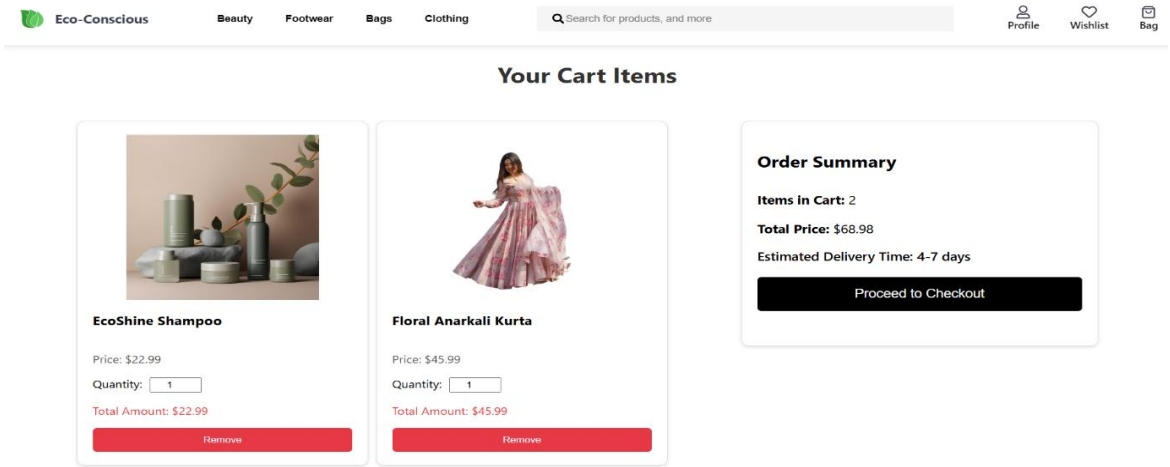
Alternatives -



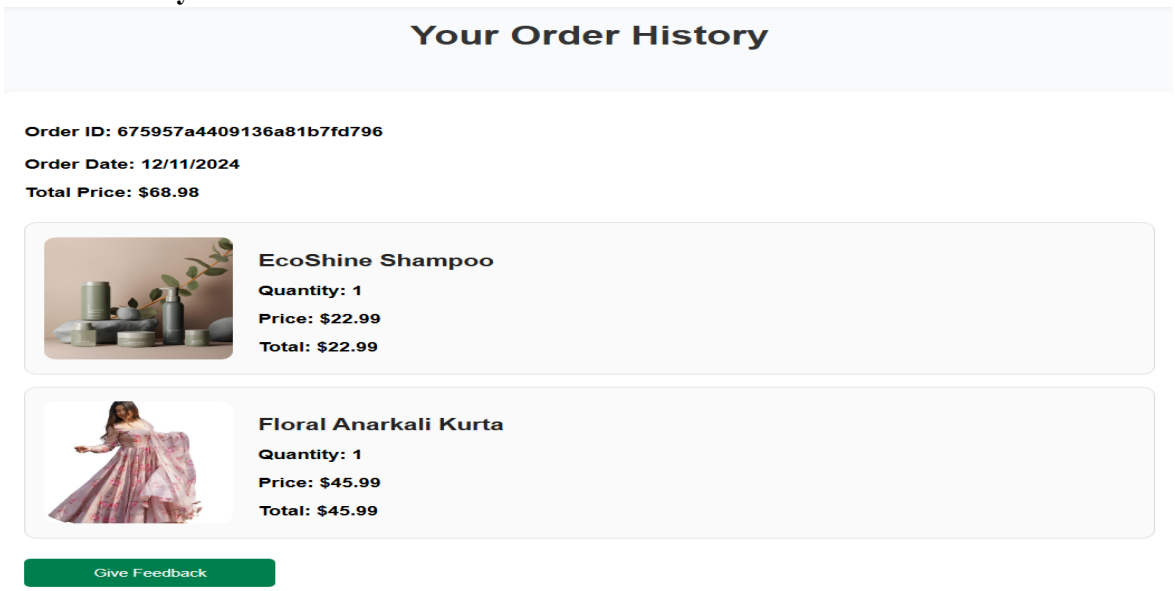
Wishlist -



Cart -

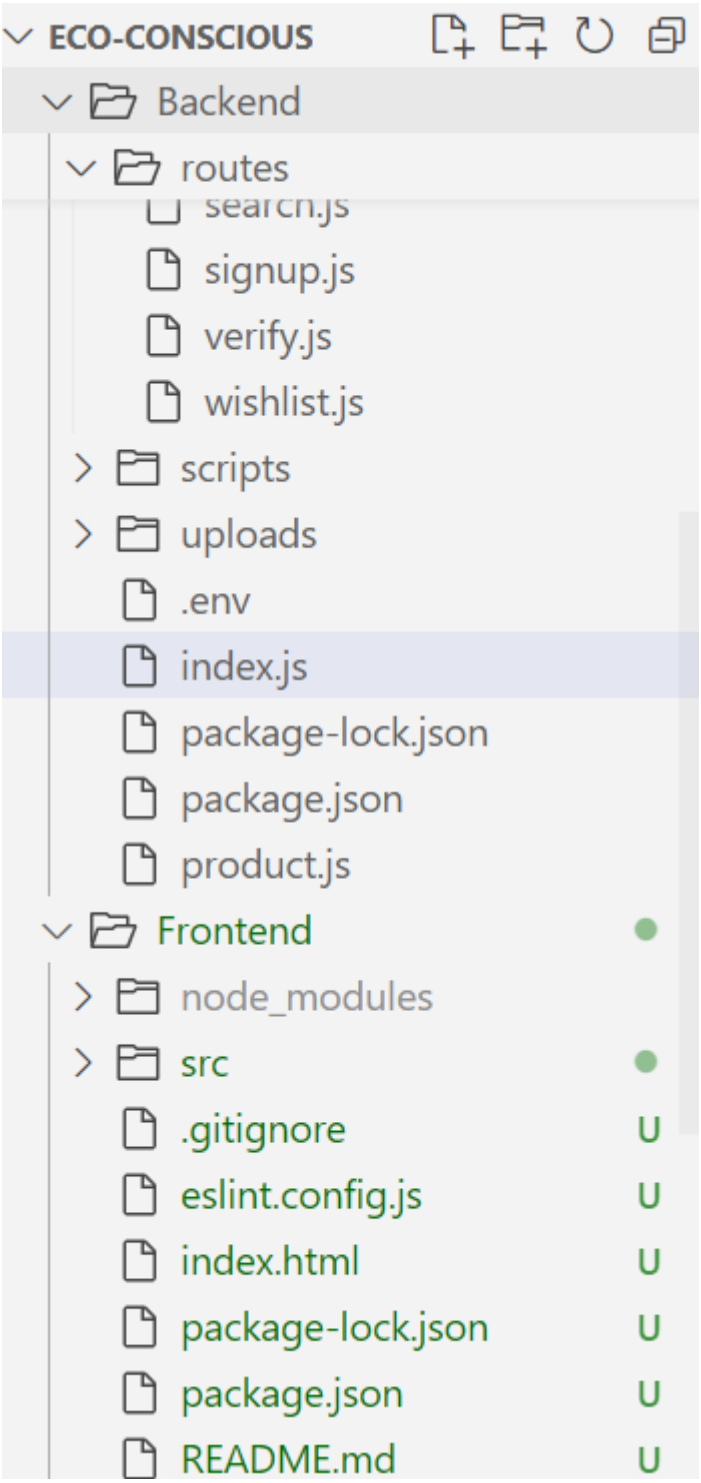


Order History –



6. Code-Implementation and Database Connections

Frontend & Backend File hierarchy:



Index.js file code:

index.js

Backend > index.js > ...

```
const express = require("express");
const mongoose = require("mongoose");
const bodyParser = require("body-parser");
const cors = require("cors");
const dotenv = require("dotenv");

// Import Routes
const signupRouter = require("./routes/signup");
const loginRouter = require("./routes/login");
const profileRouter = require("./routes/profile");
const productsRouter = require("./routes/products");
const editRouter = require("./routes/edit");
const deleteRouter = require("./routes/delete");
const wishlistRouter = require("./routes/wishlist");
const cartRouter = require("./routes/cart");
const orderRoutes = require("./routes/order");
const orderhistoryRoutes = require("./routes/orderhistory");
const bestProductRouter = require("./routes/bestProduct");
const verifyRouter = require("./routes/verify");

const errorHandler = require("./Middlewares/errorHandler");
const searchRouter = require("./routes/search");
const alternativeRouter = require("./routes/alternative");
const authenticateToken = require("./Middlewares/tokenAuthentication");
const feedbackRouter = require("./routes/feedback");

dotenv.config();

const app = express();
const port = process.env.PORT || 3000;

// MongoDB Connection using URI from .env file
mongoose
  .connect(process.env.MONGO_URL, {
    useNewUrlParser: true,
    useUnifiedTopology: true,
  })
  .then(() => console.log("Connected to MongoDB"))
  .catch((error) => console.error("MongoDB Connection Error:", error.message));
```

```
41  // Middleware
42  app.use(
43    cors({
44      // Replace this with your actual frontend URL on Vercel
45      origin: process.env.FRONTEND_URL, // Vercel frontend URL
46      methods: ["GET", "POST", "PUT", "DELETE"],
47      credentials: true,
48    })
49  );
50  app.use(bodyParser.urlencoded({ extended: true }));
51  app.use(bodyParser.json());
52  app.use(express.static("public"));
53  app.use("/uploads", express.static("uploads"));
54
55  // Routes
56  app.get("/", (req, res) => {
57    res.send("Welcome to the Eco-Conscious API");
58  });
59  app.use("/signup", signupRouter);
60  app.use("/login", loginRouter);
61  app.use("/api/profile", authenticateToken, profileRouter);
62  app.use("/api/products", authenticateToken, productsRouter);
63  app.use("/api/edit", authenticateToken, editRouter);
64  app.use("/api/delete", authenticateToken, deleteRouter);
65  app.use("/api/wishlist", authenticateToken, wishlistRouter);
66  app.use("/api/cart", authenticateToken, cartRouter);
67  app.use("/api/order", authenticateToken, orderRoutes);
68  app.use("/api/search", searchRouter);
69  app.use("/api/alternatives", alternativeRouter);
70  app.use("/api/order-history", authenticateToken, orderhistoryRoutes);
71  app.use("/api/bestproduct", authenticateToken, bestProductRouter);
72  app.use("/api/feedback", feedbackRouter);
73  app.use("/verify", verifyRouter);
74
75  // Error handling middleware
76  app.use(errorHandler);
77
78  // Update the port binding to use the dynamic environment port.
79  app.listen(process.env.PORT || 3000, () =>
80    console.log(`Server running at http://localhost:${process.env.PORT || 3000}`)
81  );
82
```

7. Limitations

Despite the innovative features and strong functionality of Eco-Conscious, there are several limitations that the platform faces, which may impact its ability to fully meet all user expectations and provide the optimal experience. These limitations include the following:

1. Difficulty in Sourcing Reliable Eco-Friendly Alternatives

A key feature of the platform is its ability to recommend **alternative eco-friendly products** with better eco-scores. However, sourcing these alternatives can be difficult, especially in product categories where eco-friendly options are scarce or less developed. While many industries are gradually introducing more sustainable products, there are still significant gaps in availability across different sectors. For example, in some markets, eco-friendly alternatives to electronics or certain everyday products may not be widely available or accessible. As Eco-Conscious grows, it will need to continually seek out new partnerships with sustainable product suppliers and manufacturers to ensure that users have access to a broader range of alternatives.

2. Challenges in Achieving Profitability While Maintaining Sustainability

Like many other businesses, Eco-Conscious faces the challenge of balancing profitability with its commitment to sustainability. While it is focused on promoting environmentally friendly products, the platform also needs to remain financially viable to sustain its operations and grow. The costs associated with ensuring product sustainability, vetting suppliers, and maintaining a high-quality user experience may limit the platform's profitability in the short term. Moreover, some eco-friendly products can be more expensive to produce or source, making it difficult to offer competitive pricing while still maintaining the platform’s sustainability goals. Striking the right balance between social responsibility and financial sustainability is a challenge that Eco-Conscious will need to navigate as it continues to grow.

8. Conclusion

Eco-Conscious is a pioneering eco-friendly e-commerce platform designed to empower consumers to make sustainable and informed shopping choices effortlessly. With growing environmental awareness, consumers are seeking ways to reduce their ecological footprint, but traditional platforms often lack transparency on product sustainability. Eco-Conscious bridges this gap by integrating features like eco-scoring, environmental filters, and alternative product suggestions, allowing users to make conscious decisions without sacrificing convenience.

At the core of Eco-Conscious is a commitment to sustainability. Each product is assessed based on environmental factors like carbon footprint, material sourcing, recyclability, and overall sustainability, presented as an eco-score. This feature, along with advanced filters, enables users to search for products based on specific sustainability criteria. Additionally, personalized suggestions for greener alternatives are provided based on users' behavior, encouraging them to opt for more sustainable choices.

The platform is powered by modern technologies such as MongoDB, Express.js, React.js, and Node.js, ensuring a scalable and secure experience. MongoDB handles flexible, unstructured data, while Express.js supports seamless API routing, React.js creates an interactive frontend, and Node.js ensures efficient server-side performance. Together, these technologies provide a smooth and responsive shopping experience while enabling Eco-Conscious to grow with increasing demand.

Beyond offering eco-friendly products, Eco-Conscious aims to drive a shift toward sustainable consumer behavior. The platform educates users about ethical consumerism and the environmental impact of their purchases through detailed product profiles and sustainability certifications. Future educational resources, such as interactive articles, videos, and webinars, will further engage users and deepen their understanding of sustainable living.

With a focus on continuous growth, innovation, and collaboration, Eco-Conscious is helping shape a future where every purchase contributes to a healthier planet. By aligning with the rising trend of ethical consumerism, Eco-Conscious is creating a movement that combines convenience with environmental responsibility, inspiring both consumers and businesses to join in building a more sustainable world.

As Eco-Conscious continues to evolve, the platform's ultimate goal remains clear: to redefine shopping with purpose and create a global community of consumers who are passionate about sustainability. Through innovation, collaboration, and a commitment to continuous improvement, Eco-Conscious is poised to play a pivotal role in shaping a future where every purchase contributes to a healthier, greener planet.

By aligning itself with the growing trend of ethical consumerism, Eco-Conscious is not just meeting the demand for eco-friendly products but is also driving a shift in the way people think about shopping. As more and more consumers prioritize sustainability, Eco-Conscious is paving the way for a new era of responsible consumption, where convenience and environmental responsibility go hand in hand.

9. Future Scope

The future of Eco-Conscious is bright, with immense potential for growth and innovation in the eco-commerce space. As the world increasingly turns toward sustainable living, the platform is uniquely positioned to lead the way in promoting responsible consumerism and making a positive environmental impact. The future scope of Eco-Conscious is driven by several exciting opportunities for expansion and enhancement.

Expanding the Product Range

- Eco-Conscious aims to broaden its catalogue to include a wider variety of eco-friendly products across diverse categories.
- By partnering with environmentally conscious brands and vendors, the platform will make sustainable options more accessible, allowing users to shop responsibly for all their needs.

Global Reach and Local Sourcing

- Expanding globally by establishing local partnerships and improving shipping logistics is a key priority.
- This approach will reduce the carbon footprint associated with long-distance transportation and ensure that sustainable products are accessible worldwide.

Integration with Third-Party Platforms

- Collaborating with other e-commerce platforms can increase visibility and allow users to make sustainable choices across multiple platforms.
- Future plans include integration with apps and tools that track carbon footprints, waste production, and energy consumption, enriching the user experience and promoting sustainability.

Educational Content and Awareness

- Adding interactive sustainability resources such as articles, videos, and webinars will help educate users about eco-friendly practices.
- By becoming a hub for sustainability knowledge, Eco-Conscious can empower users to make more informed purchasing decisions.

Building a Global Community

- Fostering a community of environmentally conscious consumers, brands, and suppliers is a long-term vision.
- Through collaboration and innovation, the platform aims to inspire collective action toward a more sustainable future.

Technological Advancements

- Leveraging cutting-edge technology to improve user experience, optimize logistics, and enhance environmental scoring algorithms.
- Continuous development will ensure that Eco-Conscious remains a leader in eco-friendly e-commerce.

10. Contributions of all Teammates

10.1 Jashanjit Kaur (2210990440)

1. **Add to Cart:** Implemented functionality to allow users to add, update, and remove products from their cart.
2. **Place Order:** Developed a seamless order placement system that processes purchases efficiently, ensuring accurate order details, and real-time updates to the database.
3. **Cart Management**: Enabled retrieval and management of user cart details from the database, allowing users to add, update, and remove items efficiently.
4. **UI Design and Implementation:** Designed a responsive and user-friendly interface with interactive elements for an enhanced user experience.
5. **Order Page:** Created a detailed order page displaying product information, quantities, prices, and order status.
6. **Order History**: Implemented a feature to fetch and display users' past orders in an organized and accessible manner.

10.2 Jasjeet Kaur (2210990442)

1. **Signup and Login:** Developed secure user authentication using JWT for signup and login functionality.
2. **Email Verification during Signup**: Integrated an email verification process during signup to ensure user authenticity by sending a confirmation link.
3. **Delete Account:** Implemented a feature allowing users to delete their accounts with proper verification.
4. **User Profile Management:** Enabled retrieval and management of user profile details from the database.
5. **Edit user details:** Developed functionality for users to update and edit their profile information securely.
6. **Logout Functionality:** Implemented a secure logout feature to invalidate user sessions and ensure data privacy.
7. **Responsive Layout**: Developed a responsive UI that adapts seamlessly across different devices and screen sizes.

10.3 Kamaljeet Kaur (2210990470)

1. **Search Functionality:** Developed a feature to allow users to search for products efficiently based on keywords.
2. **Top Picks:** Implemented a section showcasing popular and recommended products for users.
3. **Eco-Friendly Products:** Highlighted and categorized eco-friendly products to align with the platform's theme.
4. **UI Development**: Designed and implemented a user-friendly interface with intuitive navigation and visually appealing layouts to enhance the overall user experience.

10.4 Kanchan Yadav (22210990471)

1. **Database Setup:** Designed the database schema and established secure connections to manage product and user data efficiently.
2. **Eco-Score Algorithm:** Implemented an algorithm to calculate eco-scores for products based on defined environmental criteria.
3. **Product Data Retrieval:** Created APIs to retrieve and optimize product details for efficient data access.
4. **Product Profile Pages:** Designed individual product profile pages showcasing detailed information and eco-scores.
5. **Environmental Criteria Display:** Enabled dynamic display of environmental criteria for each product.
6. **Alternative Product Suggestions:** Implemented alternative product suggestions based on eco-friendly attributes.
7. **UI Design and Implementation:** Designed a user-friendly interface with responsive layouts and interactive elements to enhance user experience.
8. **Deployment:** Successfully deployed the application on a cloud platform, ensuring accessibility and scalability.

10.5 Kashika (2210990493)

1. **Wishlist:** Added functionality for users to save and manage their favourite products.
2. **Order Now:** Implemented a seamless "Order Now" feature for quick and efficient purchases.
3. **Filtering:** Developed dynamic filtering options to allow users to narrow down product selection based on specific criteria.
4. **Sorting:** Implemented sorting functionality for users to arrange products by price, popularity, or eco-score.
5. **Product Categories:** Organized products into categories to enhance navigation and improve user experience.
6. **Interactive Elements**: Incorporated interactive UI components like buttons, forms, and hover effects to improve user engagement.
7. **Order Verification:** Implemented an order verification process by sending a confirmation email with order details and a tracking link upon successful purchase.

11. References

Repository Link: https://github.com/ydv-kanchan/Eco-Conscious
For contributions:

- Fork the repository
- Create a new branch
- Commit your changes
- Push and create a pull request

Deployment Links:
Render: https://eco-conscious-z418.onrender.com
Vercel: https://eco-conscious.vercel.app/

Documentations and other links:

- reactjs: https://reactjs.org/docs/getting-started.html
- nodejs: https://nodejs.org/en/docs/
- expressjs: https://expressjs.com/en/starter/installing.html
- mongodb: https://www.mongodb.com/docs/
- mongoosejs: https://mongoosejs.com/docs/
- mongodb atlas: https://cloud.mongodb.com/