ICT ACADEMY OF KERALA

Summer Internship Report

Full Stack Application Development with ReactJS

DIYA EL FADHIL P H

Universal Engineering College

25.06.2024

EXECUTIVE SUMMARY

The project focused on developing a E-commerce website using React for the frontend and MongoDB and Express.js for the backend. The website aims to provide a user-friendly interface for customers to browse, purchase products, and manage their profiles, while also allowing administrators to manage products and users. Key activities included creating various pages for user interaction and admin functionalities. The project successfully achieved its objectives, resulting in a functional and interactive E-commerce platform.

INTRODUCTION

Project Overview : This project involved creating an E-commerce website to facilitate the buying and selling of products online. It aimed to provide a seamless shopping experience for users and efficient management capabilities for administrators.

Background Information : E-commerce platforms are essential in today's digital age, enabling businesses to reach a wider audience and providing consumers with convenience and variety.

Existing Studies : Previous studies and existing E-commerce platforms highlighted the importance of a user-friendly interface, secure transactions, and robust admin functionalities.

Context : The project was developed to address the growing need for online shopping solutions, incorporating modern web development technologies.

OBJECTIVES

1. Develop a user-friendly interface : Create web pages for users to browse and purchase products.

2. Implement secure authentication : Ensure secure login and signup processes for both users and administrators.

3. Enable efficient admin management : Provide admin functionalities to manage products and users effectively.

SCOPE AND DELIVERABLES

Scope : The project included developing the frontend using React, setting up the backend with Express and MongoDB, and ensuring seamless interaction between the two.

Tasks and Activities :

- Develop a homepage displaying newly added products.
- Create a product page listing all available products.
- Implement a profile page showing user details like name, email, location, and mobile number.
- Design login and signup pages for both users and admins.
- Create an admin page for adding and removing products and managing users.
- Develop a cart page for users to manage their selected products.

Deliverables :

- Fully functional e-commerce website.
- User authentication and profile management.
- Admin dashboard for product and user management.

METHODOLOGY

Approach : Agile allows for flexible development and continuous improvements through regular feedback.

Techniques and Tools :

- Frontend : React for building dynamic and responsive user interfaces.
- Backend : Express for setting up the server and handling API requests.
- Database : MongoDB for storing user and product information.
- Authentication : JWT (JSON Web Tokens) for secure user authentication.
- Justification : These technologies were chosen for their efficiency, scalability, and compatibility with modern web development practices.

PROJECT ACTIVITIES

Homepage Development : Displaying newly added products using React components and fetching data from the backend.

Product Page Creation : Listing all products with detailed information and enabling users to add products to the cart.

Profile Page Implementation : Fetching and displaying user details with edit functionality.

Authentication Setup : Implementing login and signup pages with JWT-based authentication.

Admin Page Development : Providing functionalities for admins to add, remove products, and manage user accounts.

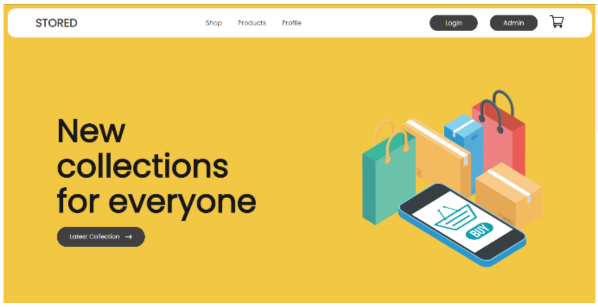
Cart Management : Allowing users to add and remove products from their cart.

RESULTS & FINDINGS

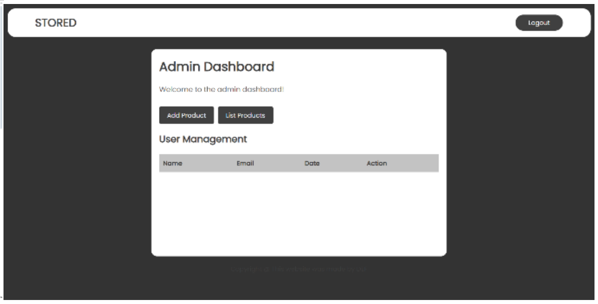
Key Results : The E-commerce website was successfully developed with all intended functionalities.

Insights :

- Improved user experience through a responsive and intuitive interface.
- Secure authentication ensured safe and reliable user interactions.
- Efficient admin management streamlined product and user handling.



a. Home Page



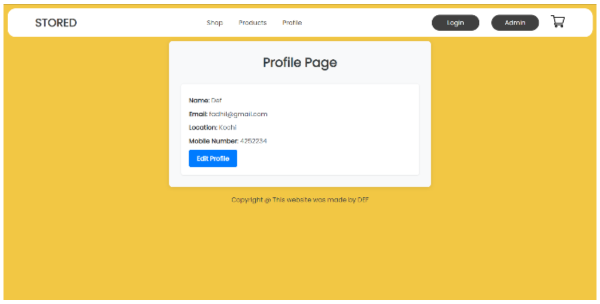
Admin Dashboard

Welcome to the admin dashboard!

User Management

Name	Email	Date	Action

b. Admin Dashboard

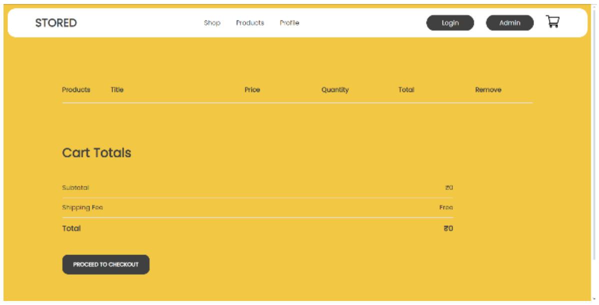


Profile Page

Name: Def
Email: fadhil@gmail.com
Location: Kochi
Mobile Number: 4252234

Copyright @ This website was made by DEF

c. Profile Page



Products	Title	Price	Quantity	Total	Remove

Cart Totals

Subtotal	₹0
Shipping Fee	Free
Total	₹0

d. Cart

CONCLUSION

Summary : The project met all objectives, resulting in a fully functional E-commerce website with comprehensive user and admin functionalities.

Reflections : The project enhanced skills in React, Express, and MongoDB, and provided valuable experience in developing a full-stack web application.

APPENDIX

- Frontend

Index.js

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import './index.css';
import App from './App';
import ShopContextProvider from './Context/ShopContext';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
    <ShopContextProvider>
      <App />
    </ShopContextProvider>
);
```

NavBar Code :

```
import React, { useContext, useRef, useState } from 'react';
import './Navbar.css';
import { Link } from 'react-router-dom';
import cart_icon from '../Assets/cart_icon.png';
import { ShopContext } from '../../Context/ShopContext';
import { AdminContext } from '../../Context/AdminContext';
import nav_dropdown from '../Assets/nav_dropdown.png';
```

```
const Navbar = () => {
  const [menu, setMenu] = useState('shop');
  const { getTotalCartItems } = useContext(ShopContext);
  const { isAdmin, logout } = useContext(AdminContext);

  const menuRef = useRef();

  const dropdown_toggle = (e) => {
    menuRef.current.classList.toggle('nav-menu-visible');
    e.target.classList.toggle('open');
  };

  const handleLogout = () => {
    logout();
    window.location.replace('/');
  };

  return (
    <div className='nav'>
      <Link to='/' onClick={() => { setMenu('shop'); }} style={{ textDecoration: 'none' }} className="nav-logo">
        <p>STORED</p>
      </Link>
      <img onClick={dropdown_toggle} className='nav-dropdown' src={nav_dropdown} alt="" />
      <ul ref={menuRef} className="nav-menu">
        {!isAdmin && (
          <>
            <li onClick={() => { setMenu('shop'); }}><Link to='/' style={{ textDecoration: 'none' }}>Shop</Link></li>
            <li onClick={() => { setMenu('mens'); }}><Link to='/products' style={{ textDecoration: 'none' }}>Products</Link></li>
            <li onClick={() => { setMenu('profile'); }}><Link to='/profile' style={{ textDecoration: 'none' }}>Profile</Link></li>
          </>
        )}
      </ul>
      <div className="nav-login-cart">
        {!isAdmin ? (
          <>
            <Link to='/login' style={{ textDecoration: 'none' }}><button>Login</button></Link>
            <button><Link to="/admin/login" className="admin-login-btn">Admin</Link></button>
          </>
        ) : (
          <button onClick={handleLogout}>Logout</button>
        )}
        {!isAdmin && (
        <Link to="/cart"><img src={cart_icon} alt="cart" /></Link>
        )}
      </div>
    </div>
  );
};

export default Navbar;
```

- Backend

```

// Add Product
app.post("/addproduct", async (req, res) => {
  let products = await Product.find({});
  let id = products.length > 0 ? products.slice(-1)[0].id + 1 : 1;
  const product = new Product({
    id: id,
    name: req.body.name,
    description: req.body.description,
    image: req.body.image,
    category: req.body.category,
    new_price: req.body.new_price,
    old_price: req.body.old_price
  });
  await product.save();
  res.json({ success: true, name: req.body.name });
});

// Remove Product
app.post("/removeproduct", async (req, res) => {
  await Product.findOneAndDelete({ id: req.body.id });
  res.json({ success: true, name: req.body.name });
});
```