FULL STACK DEVELOPMENT MINI PROJECT DOC

Project Title

Supply Chain Management with Admin Dashboard

Team Member

Siddhesh Dumre (1032222342)

Introduction - Supply Chain Management (SCM) is crucial for e-commerce, where efficiency and transparency are vital. A MERN stack-based SCM system ensures smooth product flow from suppliers to customers, minimizing delays and enhancing satisfaction. The system's integrated dashboard provides real-time insights for monitoring and optimizing the supply chain**.**

Background

Businesses need a robust SCM platform that offers real-time visibility and is both user- and vendor-friendly. The MERN stack, with MongoDB, Express.js, React.js, and Node.js, offers the flexibility and efficiency required for a scalable, dynamic SCM system.

Problem Statement

Businesses often struggle with poor visibility, order delays, and inventory management across multiple locations. A MERN stack-based SCM solution addresses these issues by offering a centralized, real-time dashboard for better decision-making and enhanced supply chain transparency.

Proposed System

The proposed SCM system with an integrated dashboard is designed to offer a comprehensive view of the entire supply chain, from procurement to delivery. The system will include features such as:

- **Inventory Management:** Real-time tracking of inventory levels across multiple warehouses.
- **Order Management:** Streamlined order processing with visibility into order status and fulfillment timelines.
- **Supplier Management:** Detailed tracking of supplier performance, including delivery times, quality, and costs.

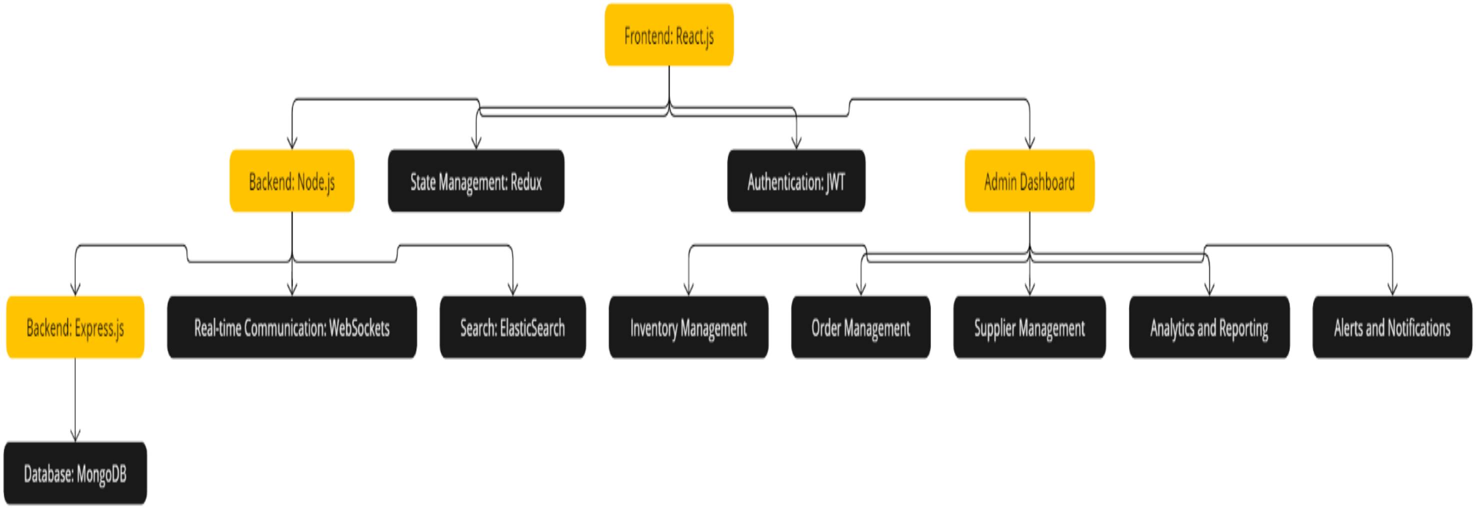
- **Analytics and Reporting:** In-depth analysis of supply chain metrics, with customizable reports and dashboards.
- **Alerts and Notifications:** Instant alerts for critical events such as low stock levels, delayed shipments, or order discrepancies.

Technology Stack

- **Frontend:** React.js, HTML, CSS, JavaScript
- **Backend:** Node.js, Express.js
- **Database:** MongoDB
- **Others:**
    - Redux for state management
    - JWT for authentication
    - WebSockets for real-time communication
    - ElasticSearch for advanced search capabilities and real-time data analysis

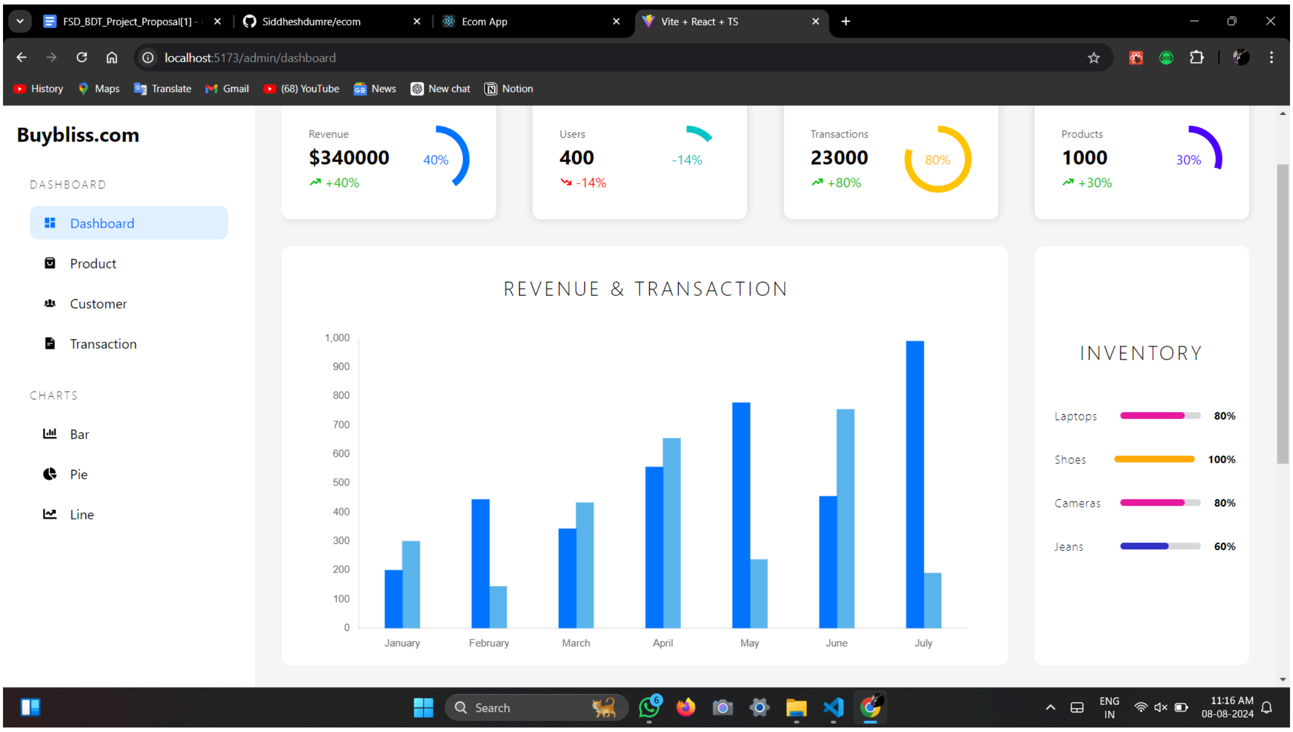
Additional Technologies

- **WebSockets:** For real-time communication, such as live chat support or instant notifications about order updates.
- **ElasticSearch:** For advanced search capabilities and real-time data analysis, providing users with faster and more relevant search results.



Proposed temp frontend and backend code

ADMIN DASHBOARD

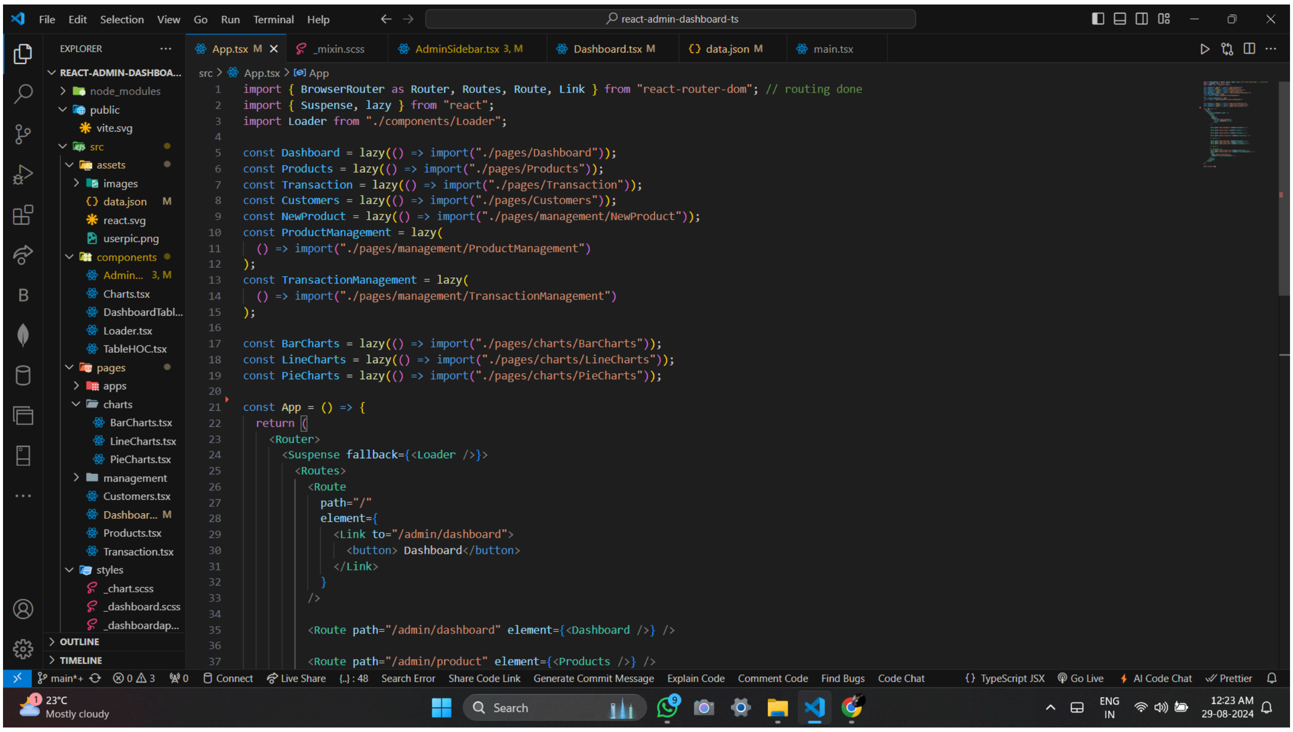


Revenue
$340000 40%
+40%

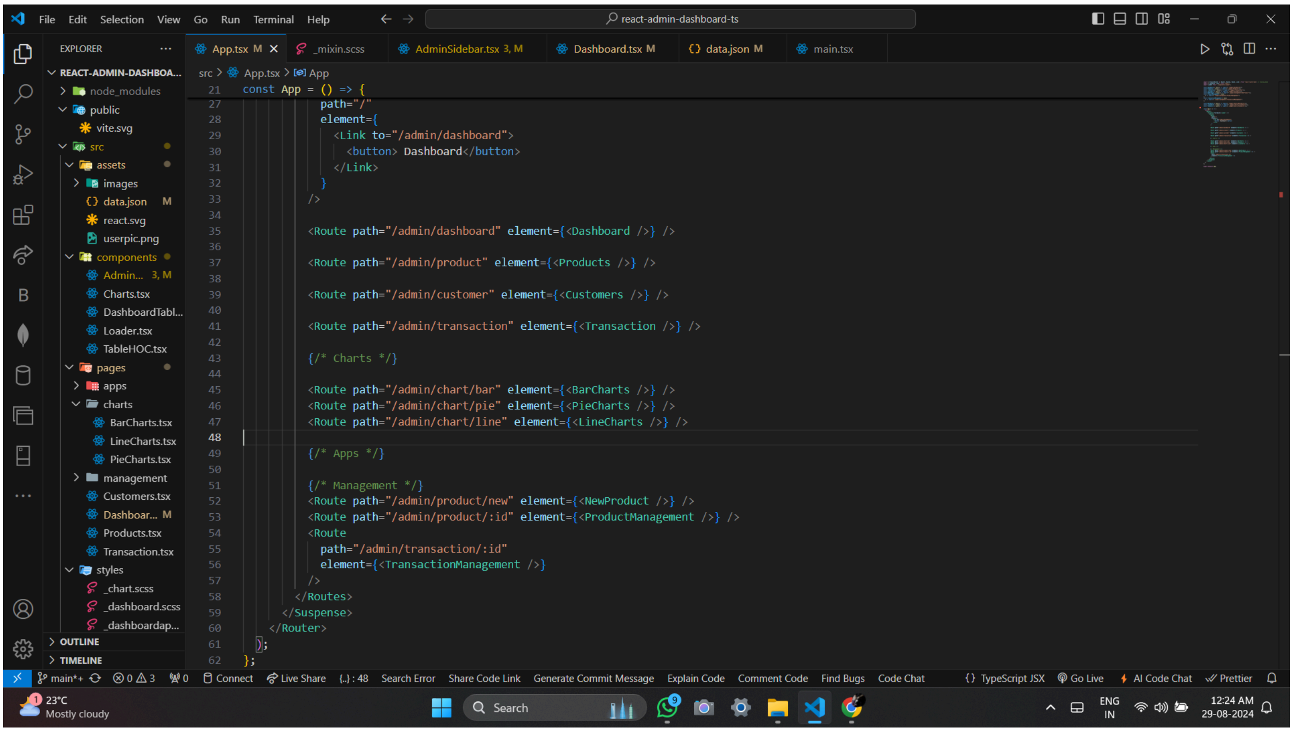
Users
400 -14%
-14%

Transactions
23000 80%
+80%

Products
1000 30%
+30%



```tsx
import { BrowserRouter as Router, Routes, Route, Link } from "react-router-dom"; // routing done
import { Suspense, lazy } from "react";
import Loader from "./components/Loader";

const Dashboard = lazy(() => import("./pages/Dashboard"));
const Products = lazy(() => import("./pages/Products"));
const Transaction = lazy(() => import("./pages/Transaction"));
const Customers = lazy(() => import("./pages/Customers"));
const NewProduct = lazy(() => import("./pages/management/NewProduct"));
const ProductManagement = lazy(
  () => import("./pages/management/ProductManagement")
);
const TransactionManagement = lazy(
  () => import("./pages/management/TransactionManagement")
);

const BarCharts = lazy(() => import("./pages/charts/BarCharts"));
const LineCharts = lazy(() => import("./pages/charts/LineCharts"));
const PieCharts = lazy(() => import("./pages/charts/PieCharts"));

const App = () => {
  return (
    <Router>
      <Suspense fallback={<Loader />}>
        <Routes>
          <Route
            path="/"
            element={
              <Link to="/admin/dashboard">
                <button> Dashboard</button>
              </Link>
            }
          />

          <Route path="/admin/dashboard" element={<Dashboard />} />

          <Route path="/admin/product" element={<Products />} />
```



```tsx
const App = () => {
            path="/"
            element={
              <Link to="/admin/dashboard">
                <button> Dashboard</button>
              </Link>
            }
          />

          <Route path="/admin/dashboard" element={<Dashboard />} />

          <Route path="/admin/product" element={<Products />} />

          <Route path="/admin/customer" element={<Customers />} />

          <Route path="/admin/transaction" element={<Transaction />} />

          {/* Charts */}

          <Route path="/admin/chart/bar" element={<BarCharts />} />
          <Route path="/admin/chart/pie" element={<PieCharts />} />
          <Route path="/admin/chart/line" element={<LineCharts />} />

          {/* Apps */}

          {/* Management */}
          <Route path="/admin/product/new" element={<NewProduct />} />
          <Route path="/admin/product/:id" element={<ProductManagement />} />
          <Route
            path="/admin/transaction/:id"
            element={<TransactionManagement />}
          />
        </Routes>
      </Suspense>
    </Router>
  );
};
```