SRI KRISHNA COLLEGE OF TECHNOLOGY
An Autonomous Institution | Accredited by NAAC with 'A' Grade
Affiliated to Anna University | Approved by AICTE
KOVAIPUDUR, COIMBATORE 641042

LOGISTICS MANAGEMENT SYSTEM

Software Design Pattern

A PROJECT REPORT

Submitted by

JENISHA ANGEL B (727822TUAD025)
SREELEKA M S (727822TUAD051)
PAVITHRA E (727822TUAD035)
RAKESH KUMAR V S (727822TUAD040)

in partial fulfillment for the award of the degree

Of

BACHELOR OF TECHNOLOGY

IN

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

JULY – 2024

LOGISTICS MANAGEMENT SYSTEM

INTRODUCTION:

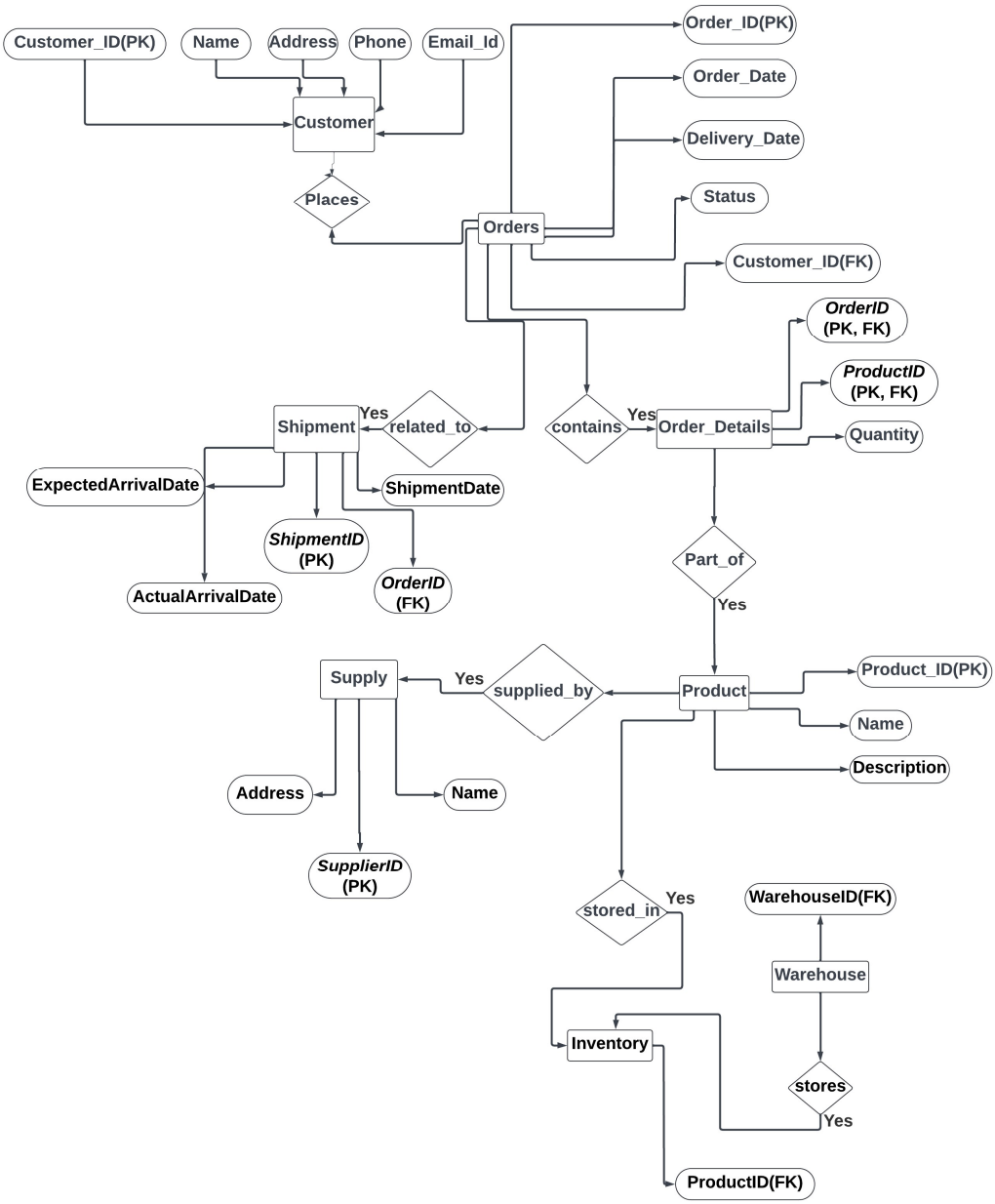
In the modern business landscape, effective logistics management is crucial for maintaining a competitive advantage and ensuring customer satisfaction. TranspoMaster, an advanced, user-friendly platform, is designed to streamline logistics operations, enhance efficiency, and provide comprehensive oversight of supply chain activities.

The primary objective of TranspoMaster is to integrate various logistics functions, such as transport services, inventory management, supplier coordination, vendor management, and financial reporting, into a cohesive system. This integration facilitates real-time data access, improved decision-making, and efficient resource allocation, ultimately leading to reduced operational costs and enhanced service delivery.

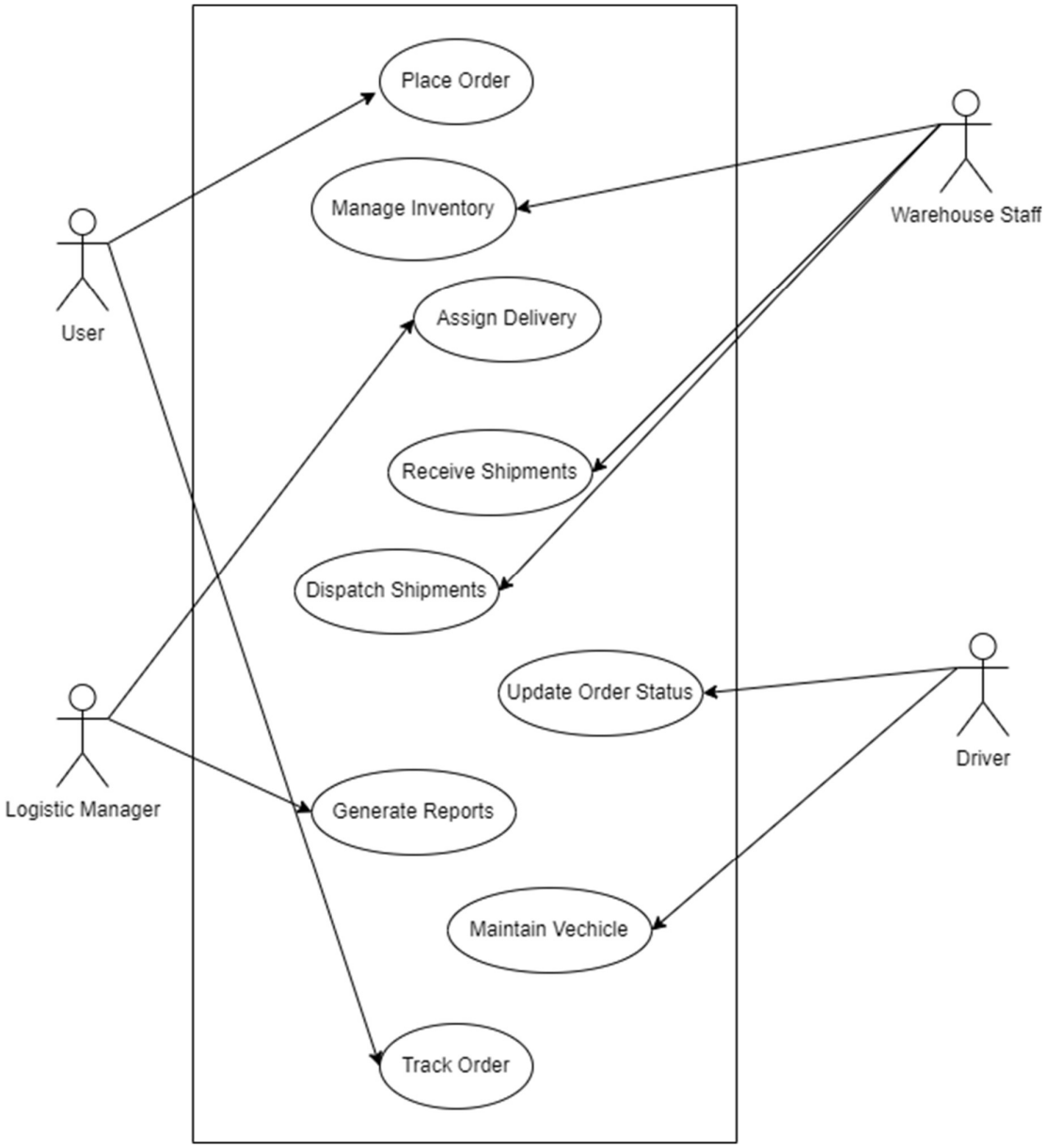
OBJECTIVE:

The objective of TranspoMaster is to enhance the efficiency, transparency, and collaboration within logistics and supply chain operations. This platform aims to optimize transport services, streamline inventory management, and improve coordination with suppliers and vendors. By providing real-time tracking, comprehensive financial reporting, and advanced analytics, TranspoMaster ensures operational excellence and informed decision-making. Additionally, it leverages automation and AI to increase agility and responsiveness, supporting business growth and scalability. Ultimately, TranspoMaster emphasizes the importance of effective communication and cooperation, fostering stronger connections among stakeholders and driving competitive advantage.

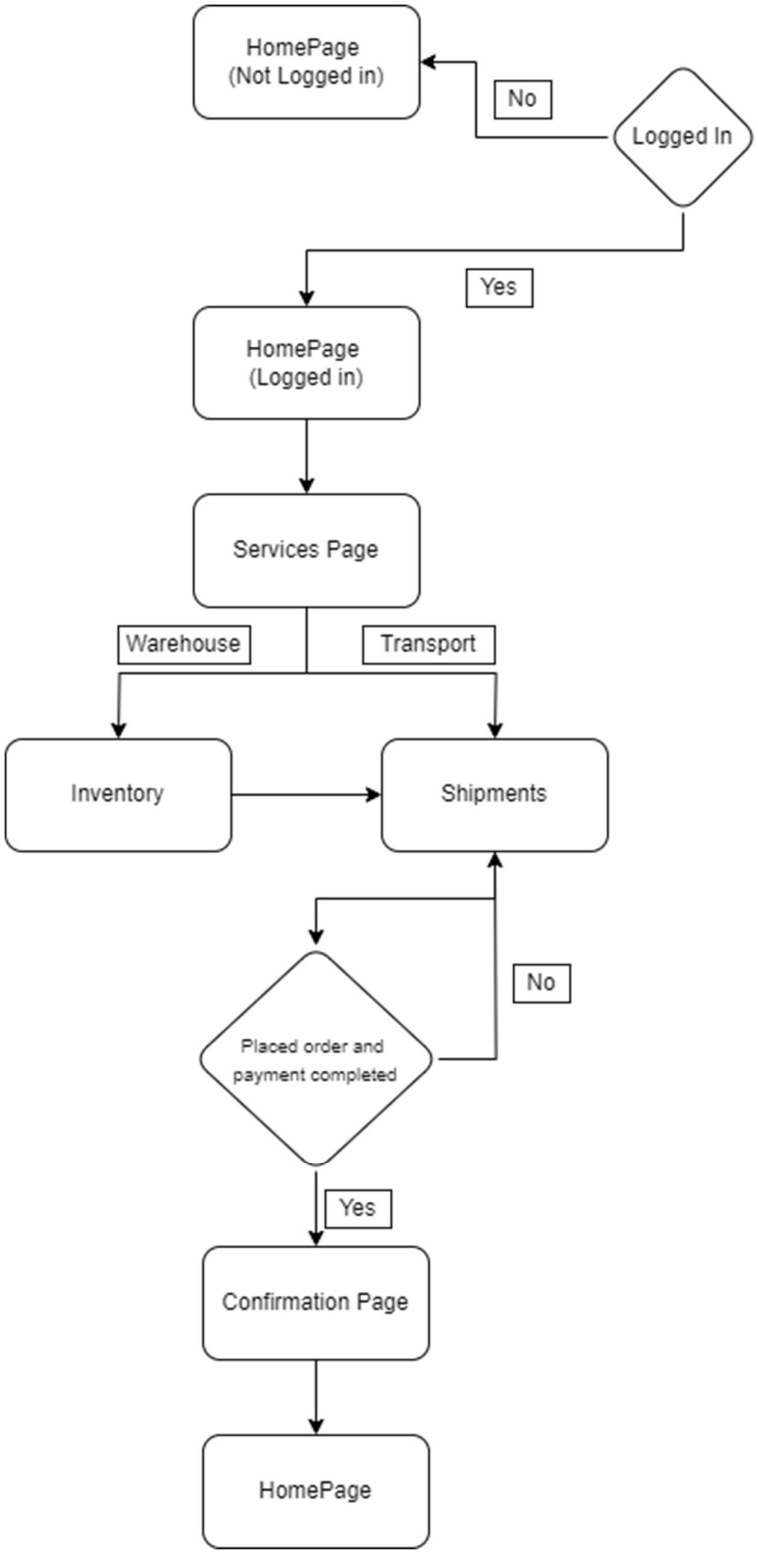
ER DIAGRAM:


Customer_ID(PK)
Name
Address
Phone
Email_Id
Customer
Places
Order_ID(PK)
Order_Date
Delivery_Date
Status
Orders
Customer_ID(FK)
OrderID
(PK, FK)
ProductID
(PK, FK)
Quantity
Shipment
Yes
related_to
contains
Yes
Order_Details
ExpectedArrivalDate
ShipmentDate
ShipmentID
(PK)
OrderID
(FK)
ActualArrivalDate
Part_of
Yes
Supply
Yes
supplied_by
Product
Product_ID(PK)
Name
Description
Address
Name
SupplierID
(PK)
stored_in
Yes
WarehouseID(FK)
Warehouse
Inventory
stores
Yes
ProductID(FK)

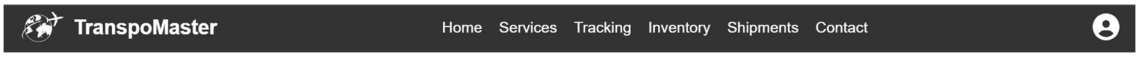
USE-CASE DIAGRAM:



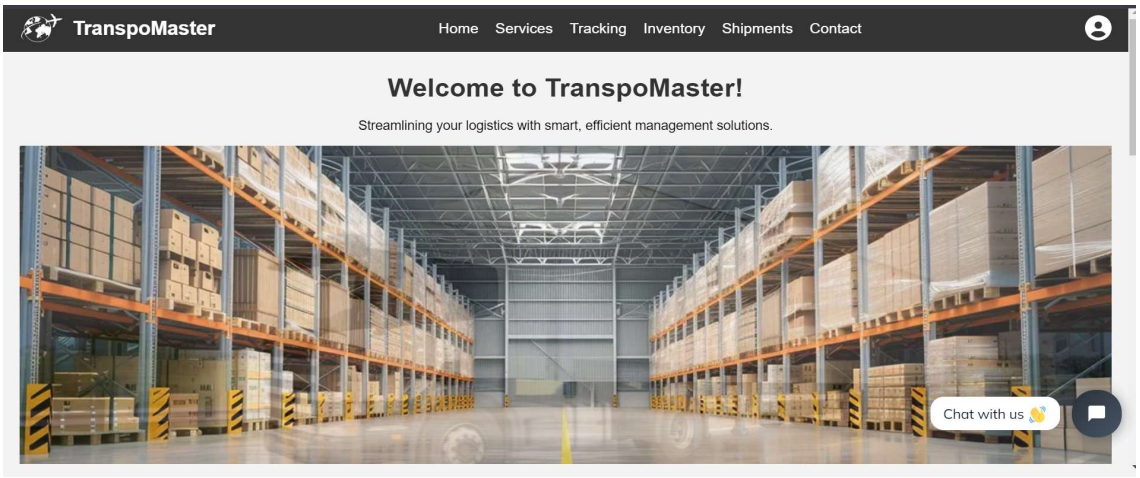
USER FLOW CHART:

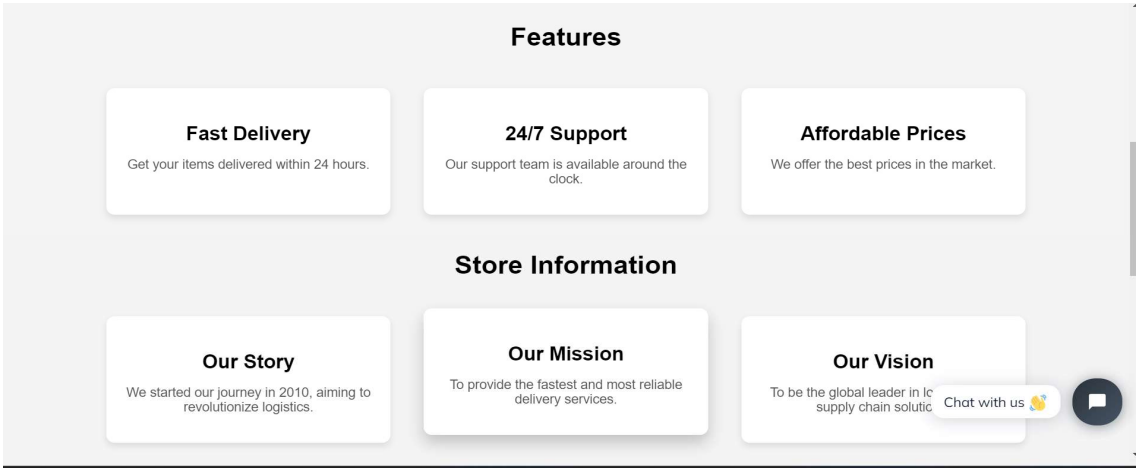


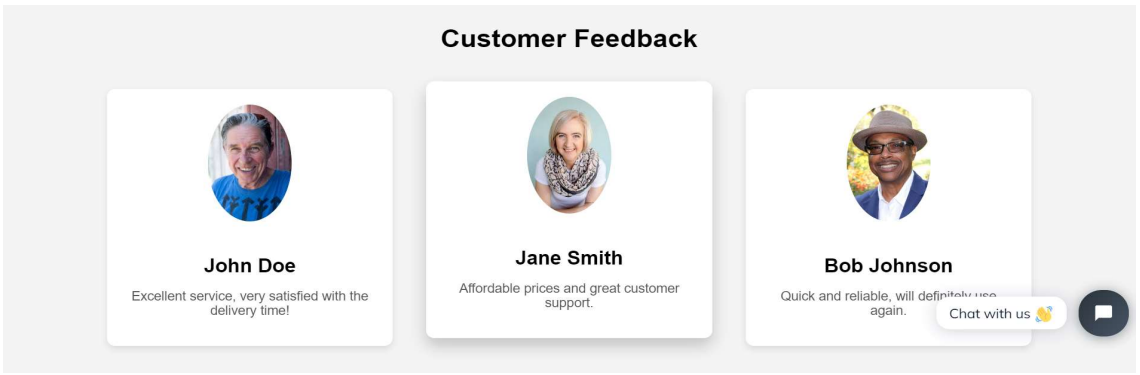
USER MODULES:



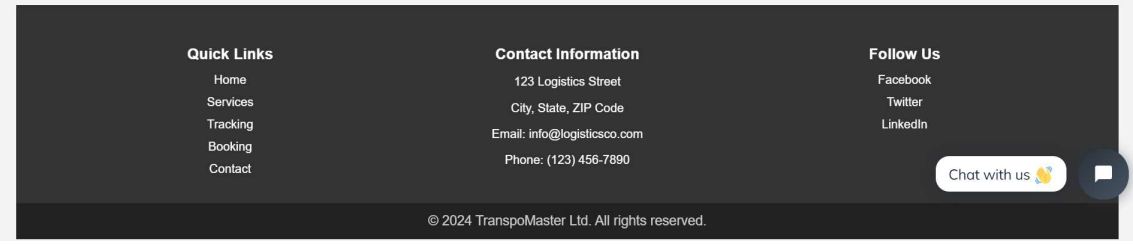
User Navigation Bar



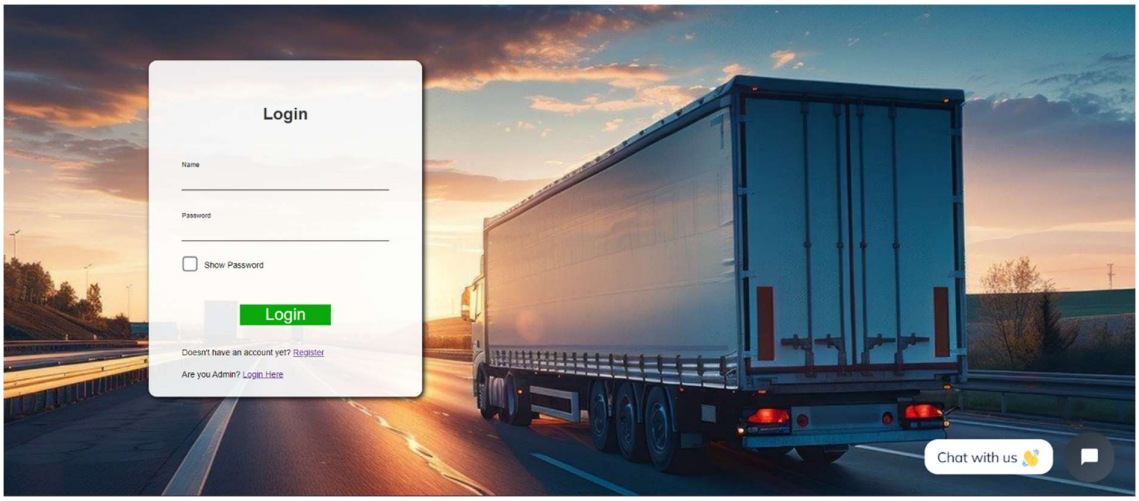




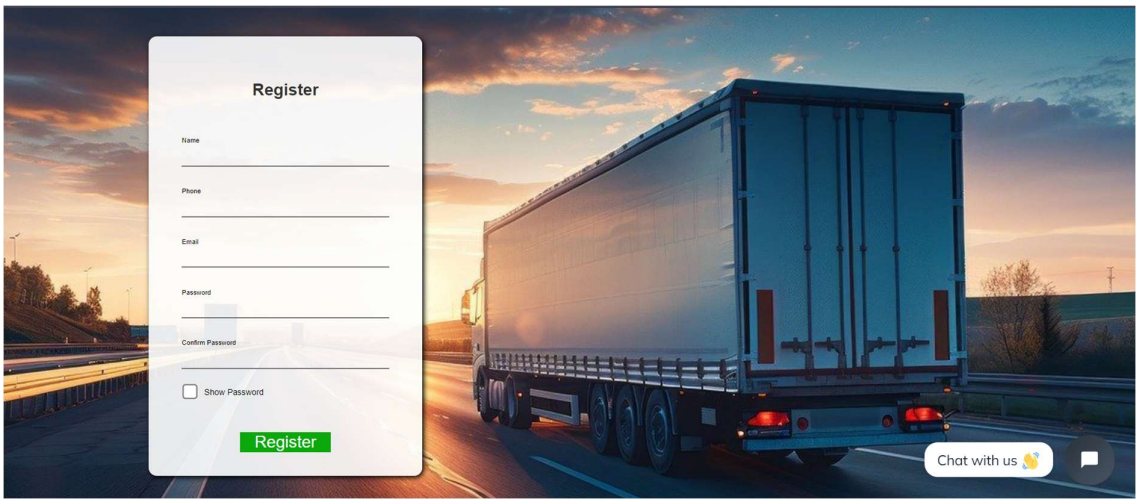
Homepage



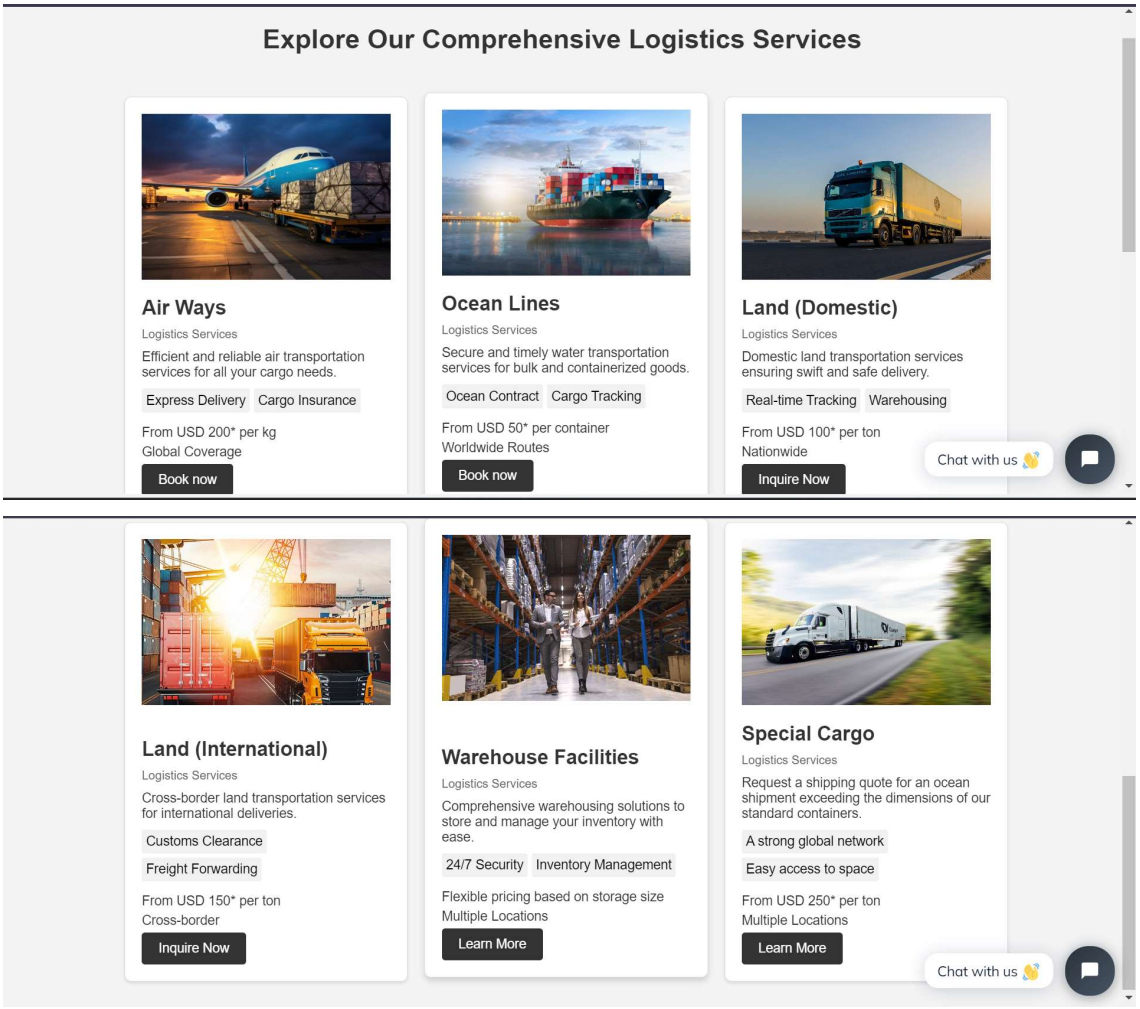
Footer



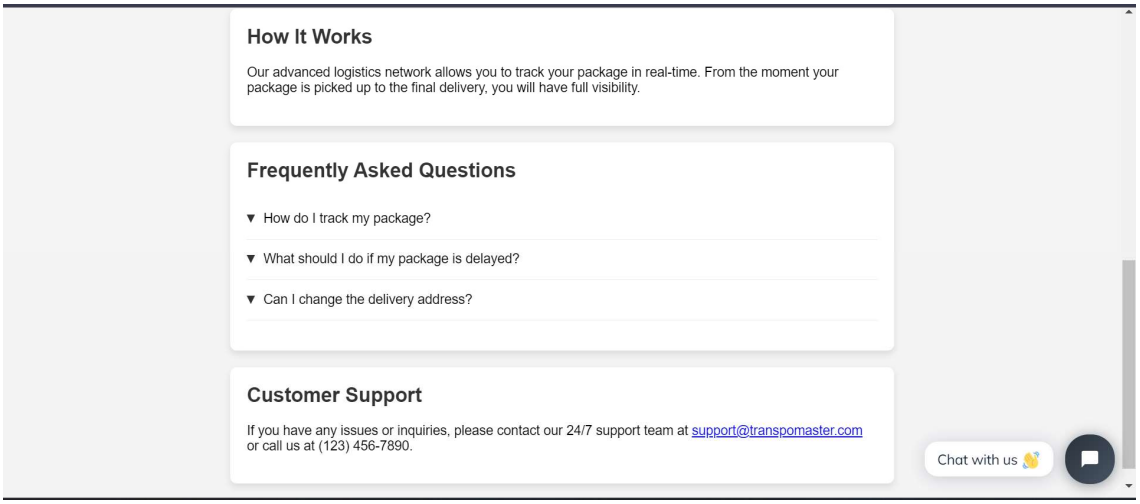
Login Page



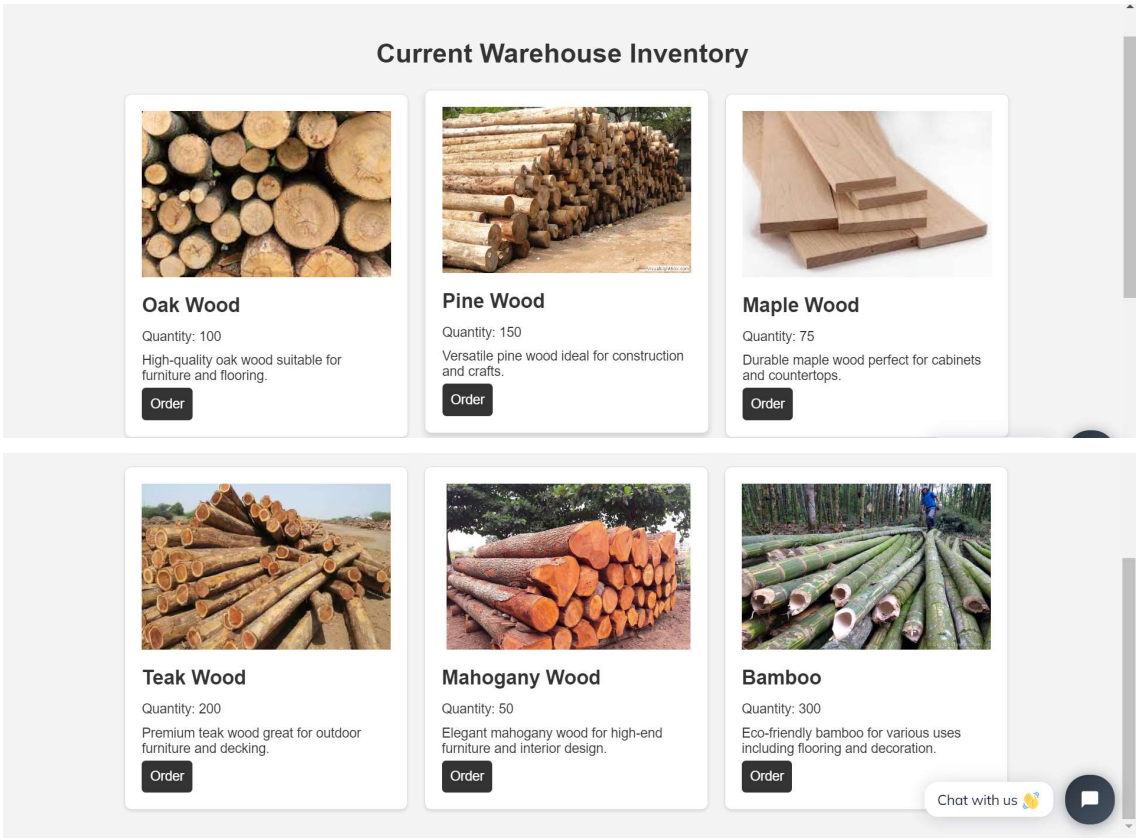
Register Page



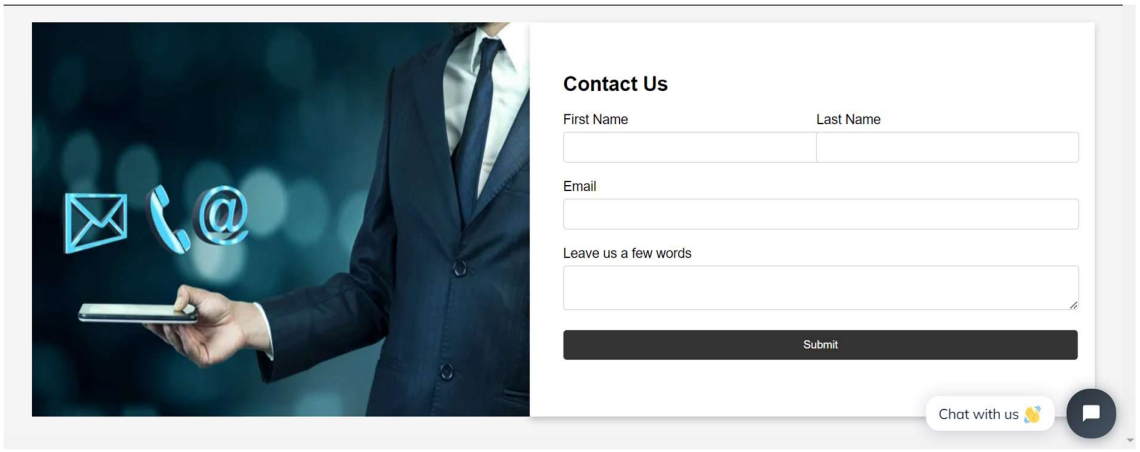
Services Page



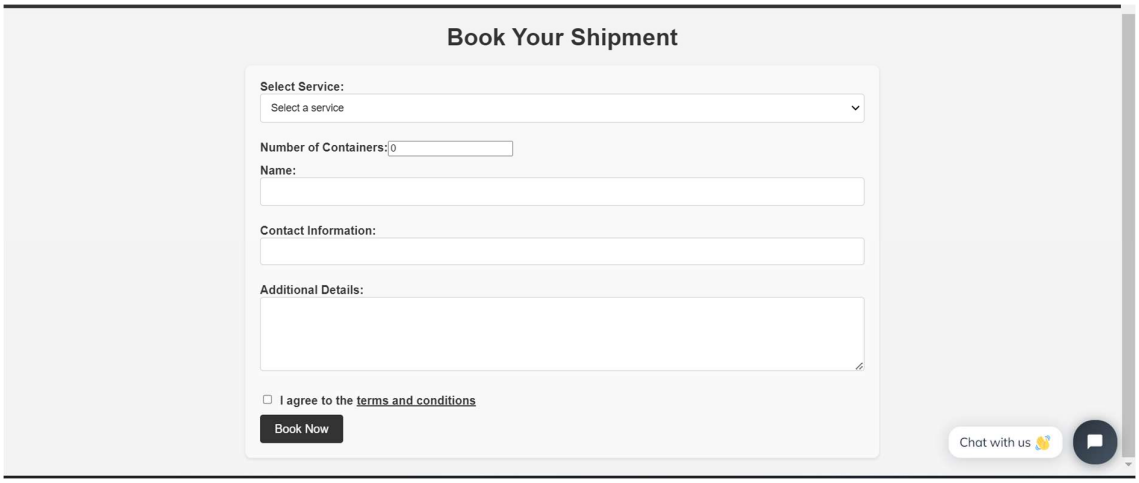
Tracking Page



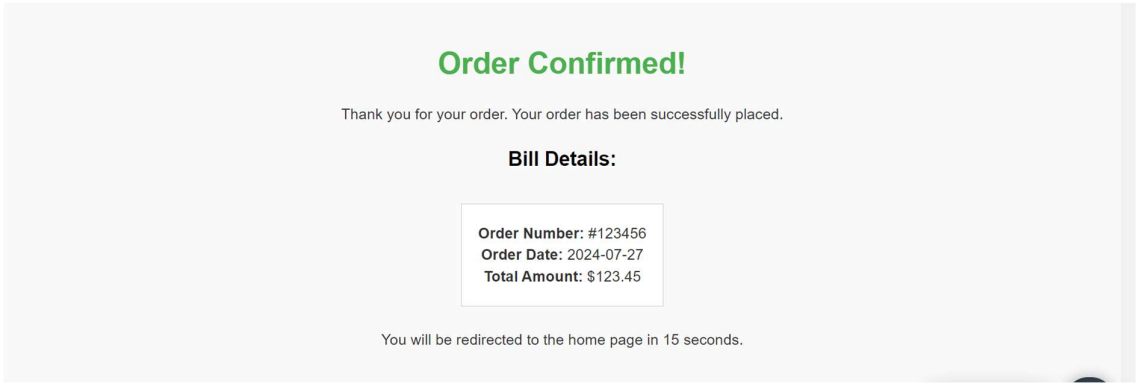
Inventory Page



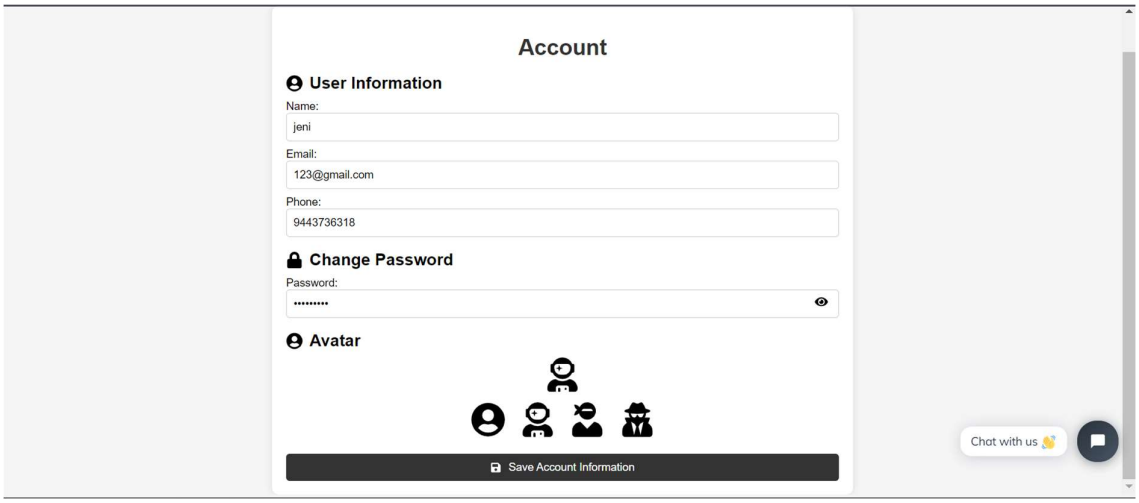
ContactUs Page



Shipments Page



Order Confirmation Page



Account Page

CODING:

login.jsx

```
import React, { useState } from "react";
import { Link } from "react-router-dom";
import '../../Assets/CSS/style.css'

function Login() {
    const [loginName, setLoginName] = useState('');
    const [loginPassword, setLoginPassword] = useState('');
    const [loginNameErr, setLoginNameErr] = useState(false);
    const [loginPasswordErr, setPasswordErr] = useState(false);
    const [incorrectErr, setIncorrectErr] = useState(false);
    const [showPassword, setShowPassword] = useState(false);

    function Loginvalidation() {
        if (loginName.trim().length !== 0) {
            setLoginNameErr(false);
        } else {
            setLoginNameErr(true);
        }

        if (loginPassword.trim().length !== 0) {
            setPasswordErr(false);
        } else {
            setPasswordErr(true);
        }
        const register = JSON.parse(localStorage.getItem('user'));
```

```
        if (register && (register.name !== loginName || register.password !== loginPassword)) {
          setIncorrectErr(true);
          alert("User Not Found");
        } else if (register) {
          setIncorrectErr(false);
          window.location.href = '/';
        }
      }
      return (
        <div className="login-body">
          <div className="login-main">
            <h1>Login</h1>
              {incorrectErr && <small style={{ color: 'red', textAlign: 'center' }}>Enter the correct username and password</small>}
            <br />
            <p>Name</p>
                <input type='text' value={loginName} onChange={(e) => { setLoginName(e.target.value) }}></input>
             {loginNameErr && <small style={{ color: '#d3521d' }}>Please enter the Username</small>}
            <br />
            <p>Password</p>
                    <input type={showPassword ? 'text' : 'password'} value={loginPassword} onChange={(e) => { setLoginPassword(e.target.value) }}></input>
              {loginPasswordErr && <small style={{ color: '#d3521d' }}>Please enter the password</small>}
            <br />
```

```
            <label className="checkbox-container">
                <input type="checkbox" checked={showPassword} onChange={()
=> setShowPassword(!showPassword)} />
                <span className="checkmark"></span>
                Show Password
            </label>
            <br/>
            <button onClick={Loginvalidation}>Login</button>
             <p style={{ fontSize: '10px' }}>Doesn't have an account yet? <Link
to={'/register'}>Register</Link></p>
                    <p style={{ fontSize: '10px' }}>Are you Admin? <Link
to={'/adminlogin'}>Login Here</Link></p>
          </div>
       </div>
   );
}

export default Login;
```

register.jsx

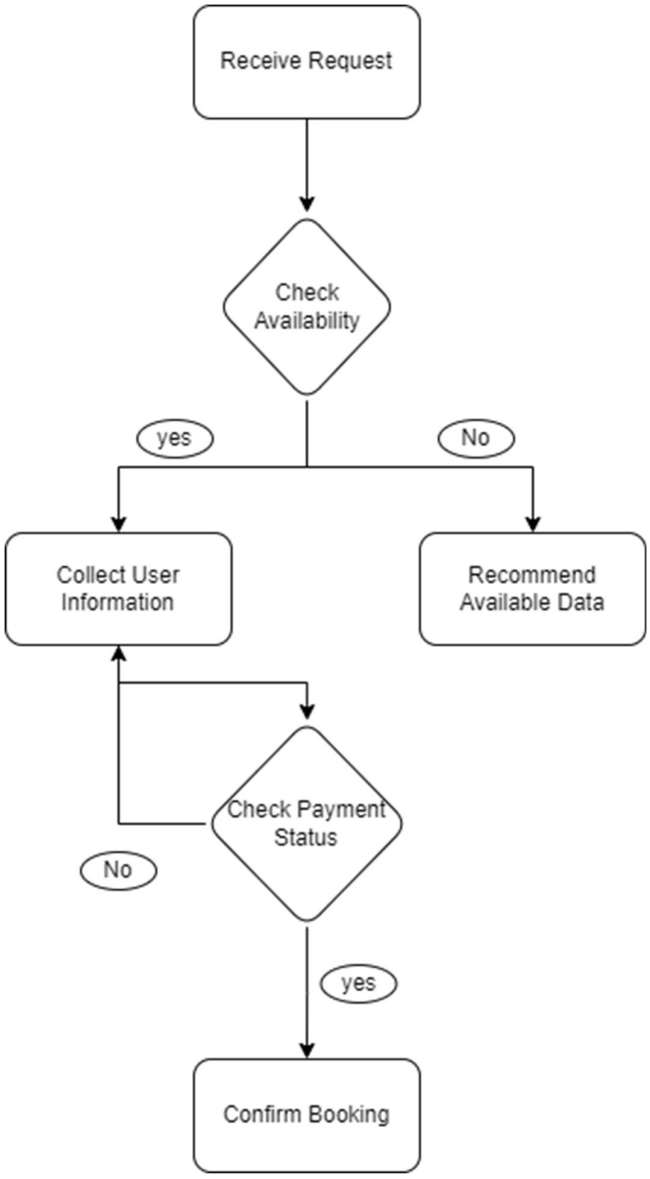
```
import React, { useState } from "react";
import '../../Assets/CSS/style.css'

function Register() {
   const [username, setUsername] = useState('');
   const [email, setEmail] = useState('');
   const [password, setPassword] = useState('');
   const [confirmPassword, setConfirmPassword] = useState('');
```

```
  const [number, setNumber] = useState('');
  const [nameErr, setNameErr] = useState(false);
  const [showPassword, setShowPassword] = useState(false);

  function registration() {
      if ((username.trim().length === 0) || (password.trim().length === 0) ||
(email.trim().length === 0)) {
        setNameErr(true);
      } else if (!email.includes('@') || !email.includes('.') || !email.includes('com'))
{
        alert('Please enter a valid email address');
      } else if (password.length < 8) {
        alert('Please enter a password with more than eight characters');
      } else if (password !== confirmPassword) {
        alert('Passwords do not match');
      } else if (number.length !== 10) {
        alert('Check your phone number');
      } else {
        setNameErr(false);
          const user = { username: username, email: email, number: number,
password: password };
        console.log(user);
          localStorage.setItem('user', JSON.stringify({ name: username, email:
email, number: number, password: password }));
        window.location.href = '/login';
      }
  }
  return (
    <div className="register-body">
```

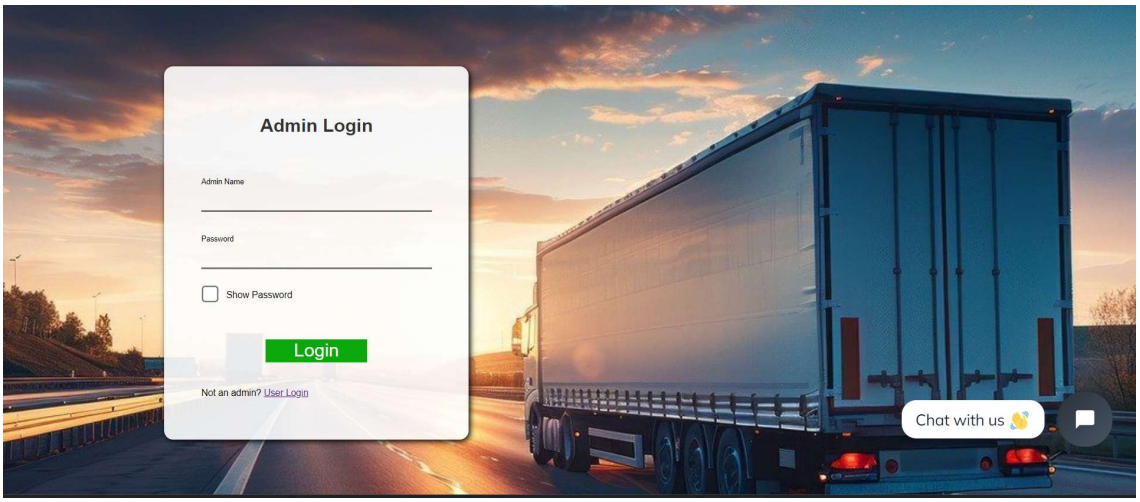
```
<div className="register-main">
    <h1>Register</h1>
    {nameErr && <p className="errP">Please fill every input field</p>}
    <br />
    <p>Name</p>
        <input type='text' value={username} onChange={(e) => { setUsername(e.target.value) }}></input>
    <br />
    <p>Phone</p>
        <input type='text' value={number} onChange={(e) => { setNumber(e.target.value) }}></input>
    <br />
    <p>Email</p>
        <input type='text' value={email} onChange={(e) => { setEmail(e.target.value) }}></input>
    <br />
    <p>Password</p>
     <input type={showPassword ? 'text' : 'password'} value={password} onChange={(e) => { setPassword(e.target.value) }}></input>
    <br />
    <p>Confirm Password</p>
            <input type={showPassword ? 'text' : 'password'} value={confirmPassword} onChange={(e) => { setConfirmPassword(e.target.value) }}></input>
    <br />
    <label className="checkbox-container">
       <input type="checkbox" checked={showPassword} onChange={() => setShowPassword(!showPassword)} />
        <span className="checkmark"></span>
```

```
                Show Password
            </label><br />
            <button onClick={registration}>Register</button>
          </div>
        </div>
    );
}

export default Register;
```

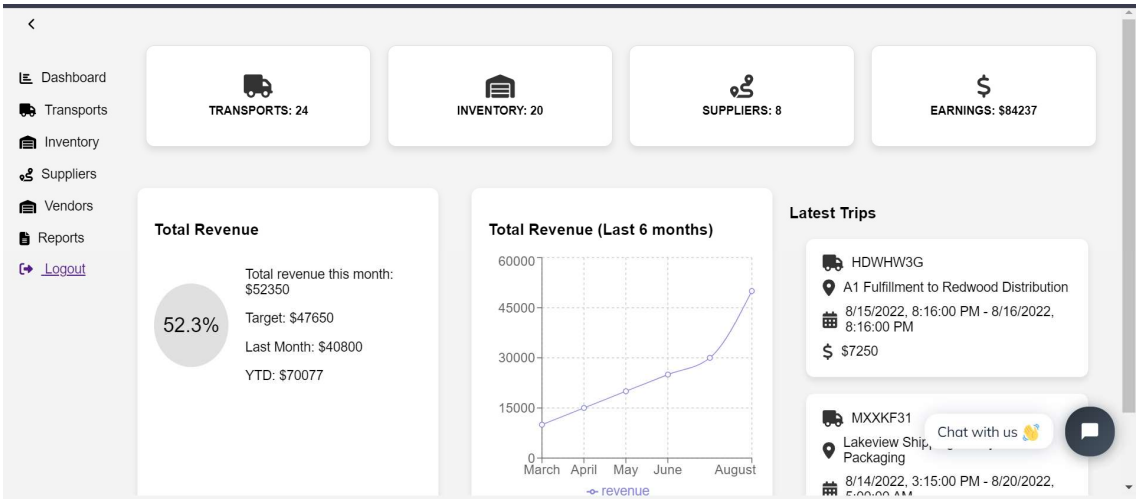
ADMIN FLOW CHART:



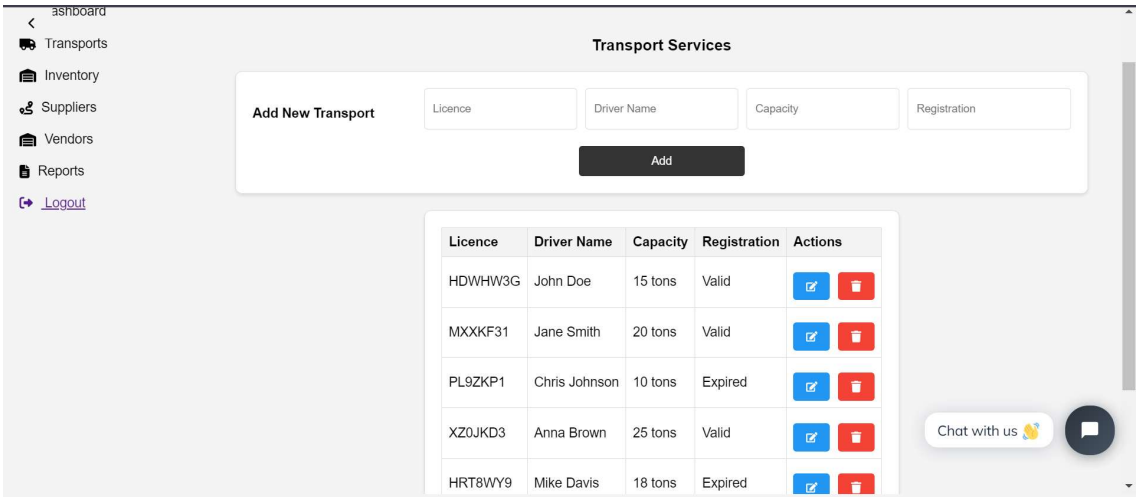
ADMIN MODULES:



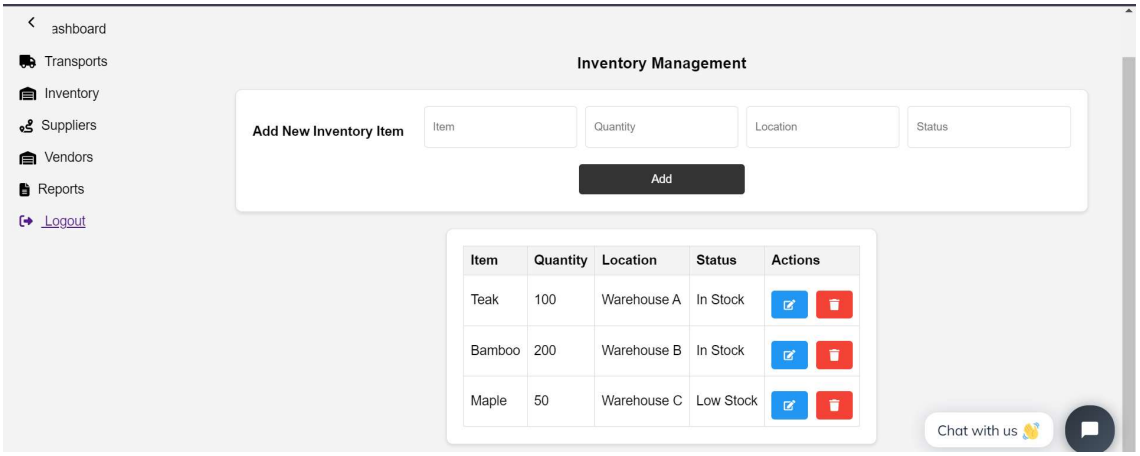
Admin LoginPage



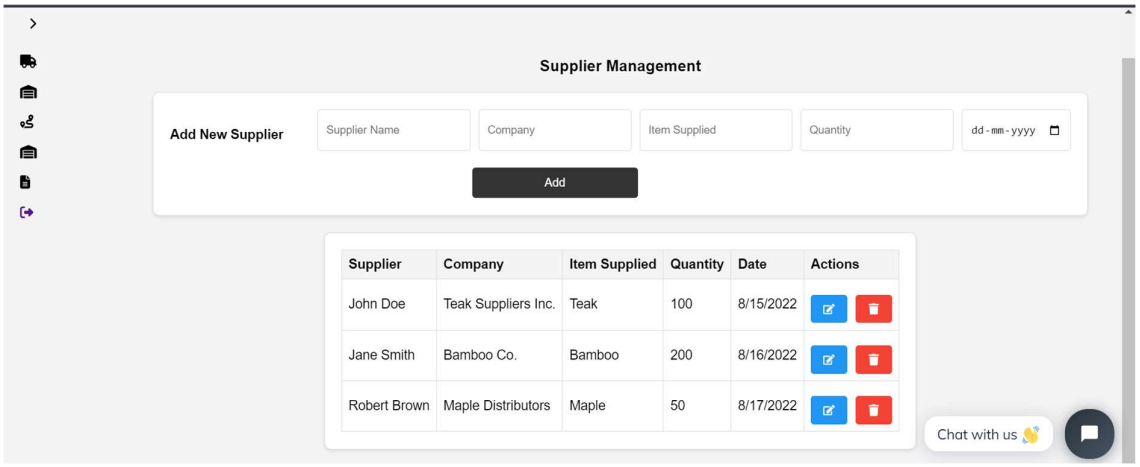
Admin Dashboard



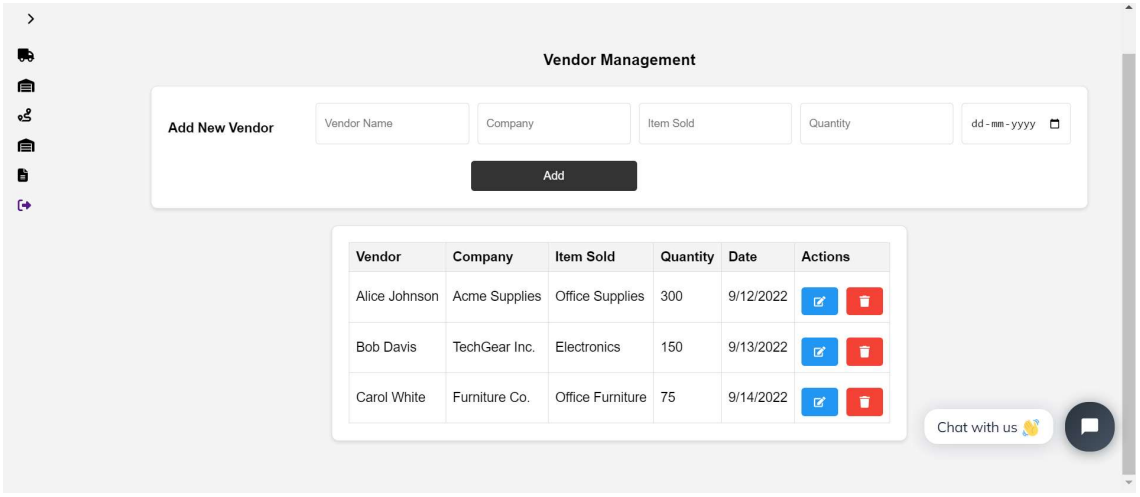
Transport Services



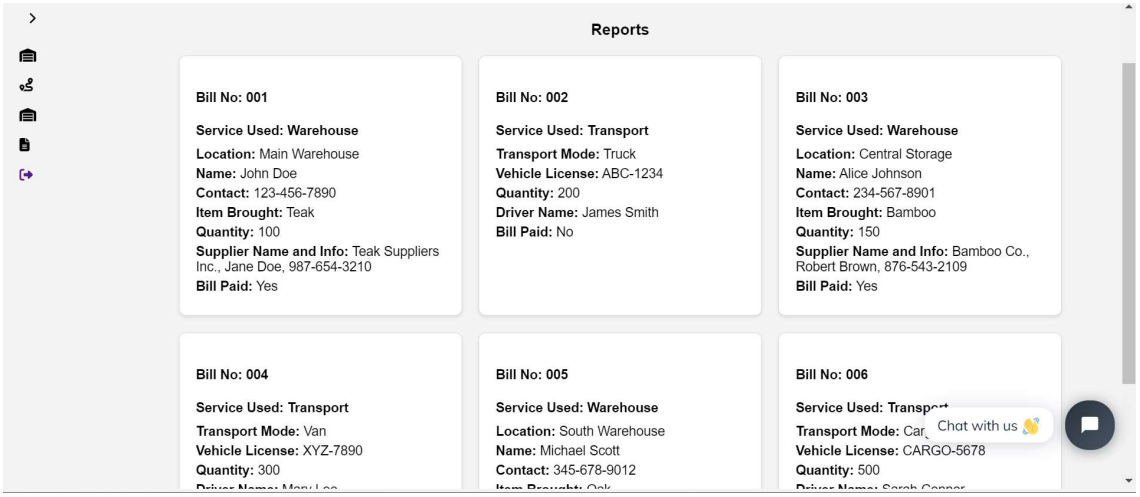
Inventory Management



Supplier Management



Vendor Management



Reports Management

CODING:

AdminLogin.jsx

```
import React, { useState } from "react";
import { Link } from "react-router-dom";
import '../../Assets/CSS/style.css';

function AdminLogin() {
   const [adminName, setAdminName] = useState('');
   const [adminPassword, setAdminPassword] = useState('');
   const [adminNameErr, setAdminNameErr] = useState(false);
   const [adminPasswordErr, setAdminPasswordErr] = useState(false);
   const [incorrectErr, setIncorrectErr] = useState(false);
   const [showPassword, setShowPassword] = useState(false);

   const adminCredentials = {
      name: 'admin',
      password: 'admin123'
   };

   function AdminLoginValidation() {
      if (adminName.trim().length !== 0) {
         setAdminNameErr(false);
      } else {
         setAdminNameErr(true);
      }

      if (adminPassword.trim().length !== 0) {
```

```
        setAdminPasswordErr(false);
      } else {
        setAdminPasswordErr(true);
      }

      if (adminName === adminCredentials.name && adminPassword === adminCredentials.password) {
        setIncorrectErr(false);
        window.location.href = '/admin-dashboard';
      } else {
        setIncorrectErr(true);
        alert("Invalid Admin Credentials");
      }
    }

    return (
      <div className="login-body">
        <div className="login-main">
          <h1>Admin Login</h1>
            {incorrectErr && <small style={{ color: 'red', textAlign: 'center' }}>Invalid Admin Credentials</small>}
          <br />
          <p>Admin Name</p>
              <input type='text' value={adminName} onChange={(e) => { setAdminName(e.target.value) }}></input>
          {adminNameErr && <small style={{ color: '#d3521d' }}>Please enter the Admin Name</small>}
          <br />
          <p>Password</p>
```

```
                        <input type={showPassword ? 'text' : 'password'} value={adminPassword} onChange={(e) => { setAdminPassword(e.target.value) }}></input>
                {adminPasswordErr && <small style={{ color: '#d3521d' }}>Please enter the Password</small>}
            <br />
            <label className="checkbox-container">
                <input type="checkbox" checked={showPassword} onChange={() => setShowPassword(!showPassword)} />
                <span className="checkmark"></span>
                Show Password
            </label>
            <br/>
            <button onClick={AdminLoginValidation}>Login</button>
                        <p style={{ fontSize: '10px' }}>Not an admin? <Link to={'/login'}>User Login</Link></p>
        </div>
    </div>
  );
}

export default AdminLogin;
```

AdminDashboard.jsx

```
import React, { useState } from 'react';
import TripCard from '../../Components/TripCard';
import '../../Assets/CSS/dashboard.css';
```

```
import TransportServices from './TransportServices';
import { FontAwesomeIcon } from '@fortawesome/react-fontawesome';
import { faTruck, faRoute, faWarehouse, faDollarSign, faCalendarAlt, faChartBar, faFileAlt, faCog, faSignOutAlt, faChevronLeft, faChevronRight } from '@fortawesome/free-solid-svg-icons';
import { LineChart, Line, XAxis, YAxis, CartesianGrid, Tooltip, Legend, ResponsiveContainer } from 'recharts';
import InventoryManagement from './inventory';
import SupplierManagement from './supplier';
import VendorManagement from './vendor';
import Reports from './reports';
import Settings from './settings';
import { Link } from 'react-router-dom';

const AdminDashboard = () => {
  const [isSidebarOpen, setIsSidebarOpen] = useState(true);
   const [activeSection, setActiveSection] = useState('dashboard'); // Default to 'dashboard'

  const toggleSidebar = () => {
    setIsSidebarOpen(!isSidebarOpen);
  };

  const handleSidebarItemClick = (section) => {
    setActiveSection(section);
  };

  return (
    <div className="dashboard">
```

```
<Sidebar
  isOpen={isSidebarOpen}
    onSidebarItemClick={handleSidebarItemClick} // Pass click handler to Sidebar
/>
<div className={`main-content ${isSidebarOpen ? 'sidebar-open' : 'sidebar-closed'}`}>
  <div className="top-bar">
    <button className="sidebar-toggle" onClick={toggleSidebar}>
          <FontAwesomeIcon icon={isSidebarOpen ? faChevronLeft : faChevronRight} />
    </button>
    {activeSection === 'dashboard' && (
     <>
      <div className="header-item">
       <FontAwesomeIcon icon={faTruck} />
       <span>TRANSPORTS: 24</span>
      </div>
      <div className="header-item">
       <FontAwesomeIcon icon={faWarehouse} />
       <span>INVENTORY: 20</span>
      </div>
      <div className="header-item">
       <FontAwesomeIcon icon={faRoute} />
       <span>SUPPLIERS: 8</span>
      </div>
      <div className="header-item">
       <FontAwesomeIcon icon={faDollarSign} />
       <span>EARNINGS: $84237</span>
```

```
            </div>
          </>
        )}
      </div>
      <div className='stats-revenue'>
        {activeSection === 'dashboard' && (
          <>
            <Stats />
            <RevenueChart />
            <LatestTrips />
          </>
        )}
        {activeSection === 'transports' && <TransportServices />}
        {activeSection === 'inventory' && <InventoryManagement />}
        {activeSection === 'suppliers' && <SupplierManagement />}
        {activeSection === 'vendors' && <VendorManagement />}
        {activeSection === 'reports' && <Reports />}
        {/* {activeSection === 'calendar' && <Calendar />} */}

      </div>
    </div>
  </div>
 );
};

const Sidebar = ({ isOpen, onSidebarItemClick }) => (
  <div className={`sidebar ${isOpen ? 'open' : 'closed'}`}>
    <ul>
      <li onClick={() => onSidebarItemClick('dashboard')}>
```

```
  <FontAwesomeIcon icon={faChartBar} /> {isOpen && 'Dashboard'}
 </li>
 <li onClick={() => onSidebarItemClick('transports')}>
  <FontAwesomeIcon icon={faTruck} /> {isOpen && 'Transports'}
 </li>
 <li onClick={() => onSidebarItemClick('inventory')}>
  <FontAwesomeIcon icon={faWarehouse} /> {isOpen && 'Inventory'}
 </li>
 <li onClick={() => onSidebarItemClick('suppliers')}>
  <FontAwesomeIcon icon={faRoute} /> {isOpen && 'Suppliers'}
 </li>
 <li onClick={() => onSidebarItemClick('vendors')}>
  <FontAwesomeIcon icon={faWarehouse} /> {isOpen && 'Vendors'}
 </li>
 <li onClick={() => onSidebarItemClick('reports')}>
  <FontAwesomeIcon icon={faFileAlt} /> {isOpen && 'Reports'}
 </li>
 {/* <li onClick={() => onSidebarItemClick('settings')}>
  <FontAwesomeIcon icon={faCog} /> {isOpen && 'Settings'}
 </li> */}
 {/* <li onClick={() => onSidebarItemClick('calendar')}>
  <FontAwesomeIcon icon={faCalendarAlt} /> {isOpen && 'Calendar'}
 </li> */}
 <li>
  <Link to="/login">
   <FontAwesomeIcon icon={faSignOutAlt} /> {isOpen && 'Logout'}
  </Link>
 </li>
</ul>
```

```
  </div>
);

const Stats = () => (
  <div className="stats">
    <div className="stat">
      <h3>Total Revenue</h3>
      <div className="revenue">
        <div className="revenue-circle">52.3%</div>
        <div className="revenue-details">
          <p>Total revenue this month: $52350</p>
          <p>Target: $47650</p>
          <p>Last Month: $40800</p>
          <p>YTD: $70077</p>
        </div>
      </div>
    </div>
  </div>
);

const data = [
  { name: 'March', revenue: 10000 },
  { name: 'April', revenue: 15000 },
  { name: 'May', revenue: 20000 },
  { name: 'June', revenue: 25000 },
  { name: 'July', revenue: 30000 },
  { name: 'August', revenue: 50000 },
];
```

```
const RevenueChart = () => (
  <div className="revenue-chart">
    <h3>Total Revenue (Last 6 months)</h3>
    <ResponsiveContainer width="100%" height={300}>
      <LineChart data={data}>
        <CartesianGrid strokeDasharray="3 3" />
        <XAxis dataKey="name" />
        <YAxis />
        <Tooltip />
        <Legend />
                <Line type="monotone" dataKey="revenue" stroke="#8884d8"
activeDot={{ r: 8 }} />
      </LineChart>
    </ResponsiveContainer>
  </div>
);
const trips = [
    {
      truck: 'HDWHW3G',
      origin: 'A1 Fulfillment',
      destination: 'Redwood Distribution',
      startDate: '8/15/2022, 8:16:00 PM',
      endDate: '8/16/2022, 8:16:00 PM',
      earnings: 7250,
    },
    {
      truck: 'MXXKF31',
      origin: 'Lakeview Shipping',
      destination: 'Keystone Packaging',
```

```
      startDate: '8/14/2022, 3:15:00 PM',
      endDate: '8/20/2022, 5:00:00 AM',
      earnings: 13000,
    },
  ];

const LatestTrips = () => (
    <div className="latest-trips">
    <h3>Latest Trips</h3>
    <div className="trip-cards">
      {trips.map((trip, index) => (
        <TripCard
          key={index}
          truck={trip.truck}
          origin={trip.origin}
          destination={trip.destination}
          startDate={trip.startDate}
          endDate={trip.endDate}
          earnings={trip.earnings}
        />
      ))}
    </div>
  </div>
);

export default AdminDashboard;
```

App.js

```
import React from 'react';
import { BrowserRouter as Router, Route, Routes } from 'react-router-dom';
import Home from './Pages/User/home';
import Login from "./Pages/User/login";
import Register from "./Pages/User/register";
import Services from './Pages/User/services';
import Tracking from './Pages/User/tracking';
import Contact from './Pages/User/contact';
import Shipments from './Pages/User/shipments';
import ConfirmationPage from './Pages/User/confirmation';
import Inventory from './Pages/User/inventory';
import Terms from './Components/terms';
import AccountPage from './Pages/User/account';
import AdminLogin from './Pages/Admin/AdminLogin';
import AdminDashboard from './Pages/Admin/AdminDashboard';

function App() {
  return (
    <Router>
      <div className="App">
        <Routes>
          <Route path="/" element={<Home />} />
          <Route path="/login" element={<Login />} />
          <Route path="/register" element={<Register />} />
          <Route path="/services" element={<Services />} />
          <Route path="/tracking" element={<Tracking />} />
```

```
          <Route path="/shipments" element={<Shipments />} />
          <Route path="/confirmation" element={<ConfirmationPage />} />
          <Route path="/account" element={<AccountPage />} />
          <Route path="/inventory" element={<Inventory />} />
          <Route path="/terms" element={<Terms />} />
          <Route path="/contact" element={<Contact />} />
          <Route path="/adminlogin" element={<AdminLogin />} />
          <Route path="/admin-dashboard" element={<AdminDashboard/>} />
        </Routes>
      </div>
    </Router>
  );
}

export default App;
```

CONCLUSION:

The TranspoMaster project concludes with significant advancements in logistics and supply chain operations, achieving marked improvements in efficiency, transparency, and collaboration. By optimizing transport services, streamlining inventory management, and enhancing coordination with suppliers and vendors, TranspoMaster has established itself as an invaluable tool for operational excellence. The platform’s integration of real-time tracking, comprehensive financial reporting, and advanced analytics has empowered businesses to make informed decisions with confidence. This holistic approach not only minimizes operational costs but also enhances service delivery, ensuring that businesses can meet their customers' demands promptly and effectively. Moreover, the centralized system has simplified complex logistics processes, providing a seamless experience for users and reducing the likelihood of errors or delays.

Furthermore, the utilization of automation and AI has significantly increased agility and responsiveness within logistics operations, supporting substantial business growth and scalability. TranspoMaster’s innovative features, such as predictive analytics and automated workflow management, have enabled businesses to anticipate market trends and adapt quickly to changes, thereby maintaining a competitive edge. The platform's emphasis on fostering effective communication and cooperation among stakeholders has strengthened industry connections, promoting a culture of collaboration and shared success. This commitment to building robust relationships and driving continuous improvement has not only enhanced operational performance but also solidified TranspoMaster's role as a key driver of innovation in the logistics sector. By positioning businesses for sustained success in an increasingly dynamic market, TranspoMaster ensures they are well-equipped to navigate future challenges and capitalize on emerging opportunities.