Plotline

NAME – JONAL SUTHAR

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY,KTR

JS2019@SRMIST.EDU.IN

PLOTLINE PROJECT DESCRIPTION

In this project, I have developed a Node.js server that handles billing operations for a company, allowing users to perform various billing-related tasks through RESTful APIs. The server is designed to manage different scenarios that can arise during billing processes, providing a seamless experience for users.

Features and Functionality:

User Account Management:

Users can create their accounts using the dedicated API endpoint.

Product and Service Information:

The server provides an API endpoint to fetch information about all available products and services, along with their respective prices.

Cart Management:

Users can add or remove items (products or services) to/from their cart

The cart can be cleared entirely.

Tax Calculation for Products:(Tax calculation code is Implemented at Client/cart_reducer.js)

Tax calculation for products is based on specified rules and price ranges.

Tax PA is applied when the product price falls between 1000 and 5000, with a tax percentage of 12% of the price.

Tax PB is applied when the product price is above 5000, with a tax percentage of 18% of the price.

Tax PC is applied uniformly to all products with a flat tax amount of 200.

Tax Calculation for Services:

Similar to products, tax calculation for services follows rules and price ranges.

Tax SA is applied when the service price falls between 1000 and 8000, with a tax percentage of 10% of the price.

Tax SB is applied when the service price is above 8000, with a tax percentage of 15% of the price.

Tax SC is applied uniformly to all services with a flat tax amount of 100.

Individual Tax Calculation:

Tax is applied individually to each product or service in the cart, ensuring accurate billing.

View Total Bill:

The total bill includes the price, quantity, and tax for each item, as well as the cumulative value of all selected items.

Order Confirmation:

Users can confirm their order using the dedicated API endpoint after reviewing their total bill.

Technology Stack:

Frontend: React.js

Building the user interface and handling user interactions.

Displaying product and service information, cart contents ,order history, and total bill.

Backend: Node.js with Express

Creating RESTful APIs to manage user authentication, products,order history, and order confirmation.

Integrating JWT tokens for secure user sessions.

Database: MongoDB

Storing user accounts, product and service information, cart data, and order history.

Security:

JWT (JSON Web Tokens) for secure authentication and authorization.

bcrypt for password hashing and encryption**.**

Conclusion:

By implementing this Node.js server, the company can efficiently manage its billing operations, offering users the ability to interact with their carts, view accurate and transparent pricing details, calculate taxes for both products and services, and finally confirm their orders. The use of REST APIs ensures a modular and scalable architecture, making it easier to extend and enhance functionality in the future.