VIGA – BACKEND INTERN

NAME: R. PRANATHI

TASK: BACKEND DEVELOPMENT FOR A FOOD DELIVERY APP

The goal of this project is to developing a REST API backend for a food delivery application using Node.js. The primary focus of this project is to implement a dynamic pricing module that calculates the total cost of food delivery based on various factors. The API should provide endpoints for user authentication, food ordering, and pricing calculation.

FEATURES

- **User Authentication**: Implement user authentication endpoints to allow users to register, login, and manage their accounts securely.
- **Food Ordering**: Create endpoints to handle food ordering functionality, including listing available restaurants, menu items, and placing orders.
- **Dynamic Pricing Module**: Develop a module or function to calculate the total cost of food delivery. The pricing should be dynamic and take into account factors such as distance, time of day, surge pricing, and any other relevant parameters.
- **API Documentation**: Provide clear documentation for all API endpoints, including input parameters, expected responses, and any authentication requirements.
- **Testing:** Thoroughly test the API endpoints using tools like Postman or curl to ensure they function correctly and handle edge cases gracefully.

Implementation Guidelines

- Use Express.js to create the REST API endpoints and handle incoming requests.
- Organize your project into logical folders such as routes, controllers, models, etc., to maintain a clean code structure.
- Implement error handling and input validation to ensure the API is robust and secure.
- Follow best practices for asynchronous programming to handle database operations and other asynchronous tasks efficiently.
- Consider scalability and performance optimizations, especially for the pricing module, which may need to handle high volumes of requests.

Deliverables

- Fully functional REST API backend with endpoints for user authentication, food ordering, and pricing calculation.
- Clear and comprehensive API documentation detailing each endpoint and its usage.
- Test cases and results demonstrating the reliability and correctness of the API.
- Well-structured codebase with appropriate comments and documentation to aid understanding and future maintenance.

STEPS

To create a REST API backend for a food delivery app with a dynamic pricing module in Node.js, you'll need to follow these general steps:

1. Setup Node.js environment: Make sure Node.js is installed on your system.

2. Initialize a new Node.js project: Use npm or yarn to initialize a new project and manage dependencies.

```
npm init
```

3. Install necessary packages: Install packages such as Express.js for creating the API endpoints and any other packages you might need for database interactions, authentication, etc.

```
npm install express
```

4. Create the folder structure: Organize your project files into logical folders such as routes, controllers, models, etc.
5. Define API endpoints: Create endpoints for various functionalities including user authentication, food ordering, and dynamic pricing calculation.
6. Implement dynamic pricing module: Create a module or function to calculate the total cost of food delivery based on various factors such as distance, time of day, surge pricing, etc. You can define the pricing logic based on your specific requirements.
7. Handle requests and response: Implement request handling logic in your Express.js routes to handle incoming requests, process data, and send appropriate responses.
8. Test your API: Use tools like Postman or curl to test your API endpoints and ensure they behave as expected.

CODE

javascript

```
// Import necessary modules
const express = require('express');
const app = express();

// Define routes
app.get('/calculate-price', (req, res) => {
   // Extract parameters from request (e.g., distance, time, etc.)
```

```
    const { distance, time, items } = req.query;

    // Calculate total cost based on pricing logic
    const totalPrice = calculatePrice(distance, time, items);

    // Send response
    res.json({ totalPrice });
});

// Pricing calculation function
function calculatePrice(distance, time, items) {
    // Add your pricing logic here
    // Example: totalPrice = basePrice + distanceCharge + timeCharge +
itemCharge
    return totalPrice;
}

// Start the server
const port = process.env.PORT || 3000;
app.listen(port, () => {
    console.log(`Server is running on port ${port}`);
});
```

CONCLUSION

In conclusion, this project involves developing a REST API backend for a food delivery application using Node.js, with a primary focus on implementing a dynamic pricing module. The backend will include features such as user authentication, food ordering functionality, and dynamic pricing calculation based on various factors like distance, time of day, and surge pricing. The implementation will follow best practices in terms of code organization, error handling, input validation, and documentation. Upon completion, the project will deliver a fully functional backend with comprehensive API documentation, thorough testing, and a well-structured codebase.