```
// server.js
const express = require('express');
const mongoose = require('mongoose');
const cors = require('cors');
const app = express();
app.use(express.json());
app.use(cors());
// MongoDB connection
mongoose.connect('mongodb://localhost:27017/food-ordering', {
useNewUrlParser: true,
useUnifiedTopology: true,
}).then(() => console.log("Connected to MongoDB"))
.catch((err) => console.error("Error connecting to MongoDB:", err));

// Menu Schema
const menuSchema = new mongoose.Schema({
name: String,
price: Number,
});
const Menu = mongoose.model('Menu', menuSchema);
// Order Schema
const orderSchema = new mongoose.Schema({
items: [{ name: String, price: Number }],
total: Number,
});
const Order = mongoose.model('Order', orderSchema);
//controller
const Customer = require('../models/Customer');
const { getCoordinates } = require("../helpers/utils/getCoordinates");
const User = require("../models/User");
async function createCustomer(req, res) {
```

```
    const coordinates = await getCoordinates();
    console.log(req.user.id);
    console.log(req.user._id);
    const customer = new Customer({
        user_id: req.user.id,
        location: {
            latitude: coordinates.latitude,
            longitude: coordinates.longitude
        }
    });
    await customer.save();
    return res.status(201).json({ success: true, message : "New Customer Created" , customer: customer },);
}

async function updateCustomerDetails(req, res) {
    const updatedCustomer = await Customer.findOneAndUpdate(
        { user_id: req.user.id },
        { $set: req.body },
        { new: true }
    );
    if (!updatedCustomer) {
        return res.status(404).json({success : false, error : "Customer not found"});
    }
    res.status(202).json({ success: true, message: "Customer Update Successfully", updatedCustomer : updatedCustomer });
}

async function deleteCustomer(req, res) {
    const deletedCustomer = await Customer.findOneAndDelete({ user_id: req.user.id });
    if (!deletedCustomer) {
        return res.status(404).json({success : false, error: "Customer not found" });
```

```
    }
    return res.status(200).json({ success: true, customer: deletedCustomer });
}

module.exports = { createCustomer, updateCustomerDetails, deleteCustomer }

// Routes
const express = require("express");
const router = express.Router();
const asyncErrorHandler = require("../middlewares/asyncErrorHandler");
const isLoggedIn = require("../middlewares/isLoggedIn");
const customerController = require("../controllers/customerController");

// # new customer
router.post('/create', isLoggedIn ,asyncErrorHandler(customerController.createCustomer));
// # update customer
router.put('/update', isLoggedIn, verifyRole("Customer"),
asyncErrorHandler(customerController.updateCustomerDetails));
// # delete customer
router.delete('/delete', isLoggedIn, verifyRole("Customer"),
asyncErrorHandler(customerController.deleteCustomer));

module.exports = router;
app.get('/menu', async (req, res) => {
const menu = await Menu.find();
res.json(menu);
});
app.post('/menu', async (req, res) => {
const newItem = new Menu(req.body);
await newItem.save();
res.json(newItem);
});
```

```
app.post('/order', async (req, res) => {
const newOrder = new Order(req.body);
await newOrder.save();
res.json(newOrder);
});
app.get('/orders', async (req, res) => {
const orders = await Order.find();
res.json(orders);
});
// Start the server
const PORT = 5000;
app.listen(PORT, () => console.log(Server running on http://localhost:${PORT}));
```