Create a Food Ordering System API

Objective
The objective of this assignment is to create a set of RESTful APIs for a food ordering system. This will involve creating models, repositories, services, and controllers.

Requirements
1. User Management
- Users can register, login, and view their profile.
- Users can be either customers or restaurant owners.

2. Restaurant Management
- Restaurant owners can create, update, and delete their restaurants.
- Each restaurant will have a list of food items.

3. Food Item Management
- Restaurant owners can add, update, and delete food items in their restaurant.
- Each food item will have a name, description, price, and availability status.

4. Order Management
- Customers can place orders from a restaurant.
- Customers can view their order history.
- Restaurant owners can view and update the status of orders.

Models
1. **User**
   - `id`
   - `username`
   - `password`
   - `email`
   - `role` (CUSTOMER, OWNER)

2. **Restaurant**
   - `id`
   - `ownerId`
   - `name`
   - `address`
   - `phone`
   - `foodItems` (List of FoodItem)

3. **FoodItem**
   - `id`
   - `restaurantId`
   - `name`

   - `description`
   - `price`
   - `availability` (boolean)

4. **Order**
   - `id`
   - `customerId`
   - `restaurantId`
   - `foodItems` (List of FoodItem)
   - `totalPrice`
   - `status` (PENDING, IN_PROGRESS, COMPLETED)

Repositories
1. **UserRepository**
   - `findByUsername(String username)`
   - `findByEmail(String email)`

2. **RestaurantRepository**
   - `findByOwnerId(String ownerId)`

3. **FoodItemRepository**
   - `findByRestaurantId(String restaurantId)`

4. **OrderRepository**
   - `findByCustomerId(String customerId)`
   - `findByRestaurantId(String restaurantId)`

Services
1. **UserService**
   - `register(User user)`
   - `login(String username, String password)`
   - `getUserProfile(String userId)`

2. **RestaurantService**
   - `createRestaurant(Restaurant restaurant)`
   - `updateRestaurant(String restaurantId, Restaurant restaurant)`
   - `deleteRestaurant(String restaurantId)`
   - `getRestaurantsByOwnerId(String ownerId)`

3. **FoodItemService**
   - `addFoodItem(String restaurantId, FoodItem foodItem)`
   - `updateFoodItem(String foodItemId, FoodItem foodItem)`
   - `deleteFoodItem(String foodItemId)`
   - `getFoodItemsByRestaurantId(String restaurantId)`

4. **OrderService**
   - `placeOrder(Order order)`
   - `getOrdersByCustomerId(String customerId)`
   - `getOrdersByRestaurantId(String restaurantId)`
   - `updateOrderStatus(String orderId, String status)`

Controllers
1. **UserController**
Register a new user
Login a user
Get user profile

2. **RestaurantController**
Create a new restaurant
Update a restaurant
Delete a restaurant
Get restaurants by owner

3. **FoodItemController**
Add a food item
Update a food item
Delete a food item
Get food items by restaurant

4. **OrderController**
Place a new order
Get orders by customer
Get orders by restaurant
Update order status

Submission Guidelines
1. Create a GitHub repository for your project.
2. Ensure your code is well-organized and commented.
3. Provide a README file with instructions on how to run your application.
4. Submit the link to your GitHub repository.

Grading Criteria
- **Code Quality**: 30%
- **API Functionality**: 40%
- **Documentation**: 20%
- **Project Structure**: 10%

Good luck, and happy coding! If you have any questions, feel free to reach out for assistance.