# Meiza Backend Full API Guide

This document describes all backend routes, their purpose, required body/params, and how to call them from React.

## 1. Auth

• POST /auth/register – Create a user and return JWT.  
Body: { name, email, password, roles }  
Returns token + user info.

• POST /auth/login – Login and return JWT.  
Body: { email, password }  
Returns token + user info.

• GET /auth/me – Return logged-in user info (requires Authorization header).

## 2. Categories

• POST /categories/addCategory – Create a new category.  
Body: { name, img }  
Returns created category.

• GET /categories – List all categories.

• GET /categories/:id – Get one category by ID.

• PUT /categories/:id – Replace category fields.

• DELETE /categories/:id – Delete a category if not in use by products.

## 3. Products

• POST /products/addNewProduct – Add product with options.  
Body: { name, desc, category, options: [ { name, price, quantity, img, sale } ] }

• GET /products – List all products with populated category.

• GET /products/:id – Get product by ID.

• PUT /products/:id – Update/replace product fields.

• DELETE /products/:id – Delete product.

## 4. Users (Admin only)

Requires Authorization: Bearer <JWT> for admin role.

• POST /users/newUser – Create new user.  
Body: { name, email, password, roles }

• GET /users – List all users.

• GET /users/:id – Get single user by ID.

• PUT /users/:id – Replace user fields (password rehash if changed).

• PATCH /users/:id – Partial update.  
• DELETE /users/:id – Delete user.

## 5. Cart

Requires Authorization: Bearer <JWT>.  
Cart is per user.

• GET /cart – Get current user's cart.

• POST /cart/items – Add item to cart.  
Body: { productId, optionId, quantity }

• PATCH /cart/items/:itemId – Update quantity.  
Body: { quantity }

• DELETE /cart/items/:itemId – Remove item from cart.

• DELETE /cart – Clear entire cart.

## 6. Orders

Requires Authorization: Bearer <JWT>.

• POST /orders/checkout – Create an order from current cart and decrement stock.  
Body: { shipping, shippingPrice, paymentMethod }

• GET /orders/my – Get logged-in user’s orders.

• GET /orders/:id – Get single order (only owner or admin).

• GET /orders – Admin only: list all orders.

• PATCH /orders/:id/status – Admin only: update order status.  
Body: { status: 'shipped' }

## 7. Required Headers

For protected routes:  
Authorization: Bearer <token>  
Content-Type: application/json

## 8. Example React usage

import axios from 'axios';  
const api = axios.create({ baseURL: 'https://thefogbank.online' });  
api.defaults.headers.common.Authorization = `Bearer ${token}`;  
  
// Login  
const { data } = await api.post('/auth/login', { email, password });  
  
// Get products  
const { data: products } = await api.get('/products');  
  
// Add to cart  
await api.post('/cart/items', { productId, optionId, quantity: 2 });  
  
// Checkout  
await api.post('/orders/checkout', { shipping, shippingPrice: 25, paymentMethod: 'cod' });  
  
// Admin update order  
await api.patch(`/orders/${orderId}/status`, { status: 'shipped' });