Product Model

const mongoose = require('mongoose');

const productSchema = new mongoose.Schema({

name: { type: String, required: true },

category: { type: String, required: true },

price: { type: Number, required: true },

imageUrl: { type: String },

});

module.exports = mongoose.model('Product', productSchema);

User Model

const mongoose = require('mongoose');

const bcrypt = require('bcrypt');

const userSchema = new mongoose.Schema({

name: { type: String, required: true },

email: { type: String, required: true, unique: true },

password: { type: String, required: true },

address: { type: String },

isAdmin: { type: Boolean, default: false },

}, { timestamps: true });

// Hash password before saving

userSchema.pre('save', async function(next) {

if (!this.isModified('password')) return next();

this.password = await bcrypt.hash(this.password, 10);

next();

});

module.exports = mongoose.model('User', userSchema);

Order Model

const mongoose = require('mongoose');

const orderSchema = new mongoose.Schema({

user: { type: mongoose.Schema.Types.ObjectId, ref: 'User' },

items: [

{

product: { type: mongoose.Schema.Types.ObjectId, ref: 'Product' },

quantity: Number,

price: Number,

}

],

total: { type: Number },

status: { type: String, default: 'Pending' },

paymentStatus: { type: String, default: 'Pending' },

}, { timestamps: true });

module.exports = mongoose.model('Order', orderSchema);