Here's a clean and scalable **backend folder structure** for your **PERN Stack Stationery E-commerce Website**, designed to support:

- Role-based access (`admin`, `user`)
- Auth (JWT: access & refresh tokens)
- Product, Cart, Order, and Return Management
- PostgreSQL with a query builder or ORM (e.g., **Sequelize**, **Prisma**, or **Knex** — adaptable)

📁 Backend Folder Structure

```
backend/
├── config/
│   ├── db.js                # PostgreSQL DB connection
│   ├── jwt.js               # JWT secret keys & config
│   └── dotenv.js            # Load environment variables
│
├── controllers/
│   ├── auth.controller.js   # Register, Login, Logout, Refresh
│   ├── user.controller.js   # User data, profile, admin-user mgmt
│   ├── product.controller.js# CRUD for products
│   ├── cart.controller.js   # Cart management
│   ├── order.controller.js  # Place, view, update orders
│   ├── return.controller.js # Return request and status mgmt
│
├── middleware/
│   ├── auth.middleware.js   # JWT token verify middleware
│   ├── role.middleware.js   # Role-based access control
│   ├── validate.middleware.js # Input validation (if needed)
│
├── models/
│   ├── index.js             # Init all models
│   ├── user.model.js
│   ├── product.model.js
│   ├── cart.model.js
│   ├── order.model.js
│   ├── orderItem.model.js
│   ├── return.model.js
│
├── routes/
│   ├── auth.routes.js
│   ├── user.routes.js
│   ├── product.routes.js
│   ├── cart.routes.js
│   ├── order.routes.js
│   ├── return.routes.js
│   └── index.js             # Combine all routes
│
├── services/
│   ├── auth.service.js
│   ├── user.service.js
│   ├── product.service.js
│   ├── cart.service.js
│   ├── order.service.js
│   ├── return.service.js
│
├── utils/
│   ├── token.utils.js       # Create, verify tokens
│   ├── password.utils.js    # Hash/compare passwords
│   ├── response.utils.js    # Format standard responses
│
├── validations/
│   ├── auth.validation.js
│   ├── product.validation.js
│   ├── cart.validation.js
│   ├── order.validation.js
│   ├── return.validation.js
│
├── .env                     # Environment variables
├── app.js                   # Express app config
├── server.js                # App entry point
└── package.json
```