Technical Assignment: Node.js Backend

Context

We are building a Node.js backend that fronts a React Native app. Shopify is the source of truth for catalog and inventory. The backend should expose: a catalog endpoint, a checkout endpoint, and a simple OAuth-based authentication system for email login.

Requirements

- Node.js + TypeScript
- Use Express or Fastify
- Data store can be in-memory
- Use provided environment variables for secrets
- Implement endpoints: GET /catalog, POST /checkout, POST /auth/login, POST /auth/token
- Validation and error handling required
- Basic logging and rate limiting required
- Minimal tests (unit + integration)
- Provide README with instructions

Endpoints

- GET /catalog – Returns active products with pagination and transformations.
- POST /checkout – Validates items, calculates amount, creates a mock payment intent, and updates inventory.
- POST /auth/login – Accepts an email, returns a mock OAuth authorization code.
- POST /auth/token – Exchanges authorization code for an access token (JWT or mock token).

Sample Data

products.json

```
[
  {
    "id": "prod_1",
    "title": "T-Shirt Classic",
    "status": "active",
    "price": 19.99,
    "inventory": 24,
    "variants": [
      { "id": "var_1_s", "title": "S" },
      { "id": "var_1_m", "title": "M" }
    ]
  },
  {
    "id": "prod_2",
    "title": "Hoodie Premium",
    "status": "active",
    "price": 49.5,
    "inventory": 5,
    "variants": [
      { "id": "var_2_m", "title": "M" },
      { "id": "var_2_l", "title": "L" }
    ]
  },
  {
    "id": "prod_3",
    "title": "Cap",
    "status": "draft",
    "price": 15,
    "inventory": 100,
    "variants": []
  }
]
```

Deliverables

- GitHub repo or zip archive
- README with setup and run instructions
- Minimal tests included
- Short notes on assumptions and trade-offs