Frontend Team APIs

1. Authentication & Users (All Roles)

- `POST /api/auth/signup` → Register new user

- `POST /api/auth/login` → Login user (returns JWT token)

- `POST /api/auth/logout` → Logout user

- `GET /api/auth/me` → Get current user profile (requires login)

Admin-only:

- `GET /api/users` → List all users

- `PATCH /api/users/:id` → Update user (role, status, etc.)

- `DELETE /api/users/:id` → Delete user

---

2. Books (Buyer + Seller + Admin)

- `GET /api/books` → Get list of books (with filters: `?search=`, `?category=`, `?sort=`)

- `GET /api/books/:id` → Get details of a specific book

Seller-only:

- `POST /api/books` → Add new book to inventory

- `PATCH /api/books/:id` → Update book details

- `DELETE /api/books/:id` → Remove a book from inventory

Admin:

- Can also **approve/reject** books if moderation is required.

---

3. Catalog & Wishlist (Buyer)

- `GET /api/catalog` → Browse catalog (paginated, filterable)

- `POST /api/wishlist` → Add book to wishlist

- `GET /api/wishlist` → Get user's wishlist

- `DELETE /api/wishlist/:bookId` → Remove book from wishlist

---

4. Cart & Checkout (Buyer)

- `GET /api/cart` → Get items in user's cart

- `POST /api/cart` → Add book to cart `{ bookId, qty }`

- `PATCH /api/cart/:itemId` → Update cart item qty

- `DELETE /api/cart/:itemId` → Remove item from cart

Checkout

- `POST /api/checkout` → Start checkout process (returns payment link/intent)

- `POST /api/orders` → Place order after payment success

- `GET /api/orders/:id` → Get order details

- `GET /api/orders` → Get user's past orders

---

5. Orders (Seller + Admin)

Seller

- `GET /api/seller/orders` → Get orders for seller's books

- `PATCH /api/seller/orders/:id/status` → Update order status (Pending → Shipped → Delivered → Returned)

- `POST /api/seller/orders/:id/return` → Process return

Admin

- `GET /api/admin/orders` → All orders across platform

---

6. Sales & Reports (Seller + Admin)

Seller

- `GET /api/seller/reports/sales` → View monthly/weekly sales

- `GET /api/seller/reports/inventory` → View stock & inventory

Admin

- `GET /api/admin/reports/sales` → Platform-wide sales analytics

- `GET /api/admin/reports/users` → User activity report

---

7. Customer Management (Seller + Admin)

Seller

- `GET /api/seller/customers` → View customers who bought from seller

- `GET /api/seller/customers/:id` → View profile of customer

- `POST /api/seller/customers/:id/message` → Send message to customer

Admin

- `GET /api/admin/customers` → View all platform customers

---

8. Account Settings (All Roles)

- `PATCH /api/account/profile` → Update profile info

- `PATCH /api/account/password` → Change password

- `PATCH /api/account/payment` → Update payment preferences (for seller payouts)

---

9. Support & Help

- `POST /api/support/ticket` → Create support ticket

- `GET /api/support/ticket/:id` → View ticket status

- `GET /api/faq` → View FAQ

Data Models

User

- `id` (string/UUID)

- `name` (string)

- `email` (string, unique)

- `passwordHash` (string, not exposed to frontend)

- `role` (enum: "buyer", "seller", "admin")

- `createdAt`, `updatedAt`

Book

- `id` (string/UUID)

- `title` (string)

- `author` (string)

- `price` (number)

- `description` (string)

- `coverImageUrl` (string)

- `sellerId` (foreign key → User)

- `createdAt`, `updatedAt`

Order

- `id` (string/UUID)

- `buyerId` (foreign key → User)

- `bookId` (foreign key → Book)

- `status` (enum: "pending", "paid", "shipped", "delivered")

- `createdAt`, `updatedAt`