

listing resource

Listing Resource:

a) Use Case: Seller creates a new listing for a textbook

- Resource(s): listings
- HTTP Method: POST
- Request URL: <http://localhost:8080/api/listings>
- Request Body:

```
{
  "seller_id": "674b1234567890abcdef1234",
  "title": "Introduction to Algorithms (3rd Edition)",
  "description": "Excellent condition textbook, barely used. No highlights or
notes.",
  "price": 45.00,
  "category": "Textbooks",
  "condition": "Like New",
  "status": "Published"
}
```
- Response:

```
{
  "id": "674b5678901234abcdef5678",
  "seller_id": "674b1234567890abcdef1234",
  "title": "Introduction to Algorithms (3rd Edition)",
  "description": "Excellent condition textbook, barely used. No highlights or
notes.",
  "price": 45.0,
  "category": "Textbooks",
  "condition": "Like New",
  "status": "Published",
  "created_at": "2025-11-18T10:30:00.123456",
  "updated_at": "2025-11-18T10:30:00.123456"
}
```

b) Use Case: Buyer browses all available listings

- Resource(s): listings
- HTTP Method: GET
- Request URL: http://localhost:8080/api/listings?page=1&per_page=10
- Request Body: None
- Response:

```
{
  "listings": [
    {
      "id": "674b5678901234abcdef5678",
      "seller_id": "674b1234567890abcdef1234",
      "title": "Introduction to Algorithms (3rd Edition)",
      "description": "Excellent condition textbook, barely used. No highlights or
notes.",
      "price": 45.0,
      "category": "Textbooks",
      "condition": "Like New",
      "status": "Published",
      "created_at": "2025-11-18T10:30:00.123456",
      "updated_at": "2025-11-18T10:30:00.123456"
    }
  ],
  "page": 1,
  "per_page": 10,
  "total": 1,
  "total_pages": 1
}
```

c) Use Case: Buyer filters listings by category (Electronics)

- Resource(s): listings
- HTTP Method: GET
- Request URL:
http://localhost:8080/api/listings?category=Electronics&page=1&per_page=10
- Request Body: None
- Response:

```
{
  "listings": [],
  "page": 1,
  "per_page": 10,
  "total": 0,
  "total_pages": 0
}
```

d) Use Case: Buyer sorts listings by price (ascending)

- Resource(s): listings
- HTTP Method: GET
- Request URL:
http://localhost:8080/api/listings?sort_by=price&page=1&per_page=10

- Request Body: None
- Response:

```
{
  "listings": [
    {
      "id": "674b5678901234abcdef5678",
      "seller_id": "674b1234567890abcdef1234",
      "title": "Introduction to Algorithms (3rd Edition)",
      "description": "Excellent condition textbook, barely used. No highlights or
notes.",
      "price": 45.0,
      "category": "Textbooks",
      "condition": "Like New",
      "status": "Published",
      "created_at": "2025-11-18T10:30:00.123456",
      "updated_at": "2025-11-18T10:30:00.123456"
    }
  ],
  "page": 1,
  "per_page": 10,
  "total": 1,
  "total_pages": 1
}
```

e) Use Case: Buyer views details of a specific listing

- Resource(s): listings
- HTTP Method: GET
- Request URL:

<http://localhost:8080/api/listings/674b5678901234abcdef5678>

- Request Body: None
- Response:

```
{
  "id": "674b5678901234abcdef5678",
  "seller_id": "674b1234567890abcdef1234",
  "title": "Introduction to Algorithms (3rd Edition)",
  "description": "Excellent condition textbook, barely used. No highlights or
notes.",
  "price": 45.0,
  "category": "Textbooks",
  "condition": "Like New",
  "status": "Published",
  "created_at": "2025-11-18T10:30:00.123456",
  "updated_at": "2025-11-18T10:30:00.123456"
}
```

f) Use Case: Seller updates the price and description of their listing

- Resource(s): listings
- HTTP Method: PATCH
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678>
- Request Body:

```
{  
  "price": 40.00,  
  "description": "Excellent condition textbook, barely used. No highlights or  
notes. Price reduced!"  
}
```
- Response:

```
{  
  "id": "674b5678901234abcdef5678",  
  "seller_id": "674b1234567890abcdef1234",  
  "title": "Introduction to Algorithms (3rd Edition)",  
  "description": "Excellent condition textbook, barely used. No highlights or  
notes. Price reduced!",  
  "price": 40.0,  
  "category": "Textbooks",  
  "condition": "Like New",  
  "status": "Published",  
  "created_at": "2025-11-18T10:30:00.123456",  
  "updated_at": "2025-11-18T10:35:00.789012"  
}
```

g) Use Case: Seller deletes their listing

- Resource(s): listings
- HTTP Method: DELETE
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678>
- Request Body: None
- Response:

```
{  
  "message": "Listing deleted successfully"  
}
```


h) Use Case: Seller adds a photo to their listing

- Resource(s): listings, photos
- HTTP Method: POST
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/photos>
- Request Body:

```
{  
  "photo_url": "https://example.com/images/textbook-front.jpg",  
  "display_order": 0  
}
```
- Response:

```
{  
  "id": "674b9999888877776666aaaa",  
  "listing_id": "674b5678901234abcdef5678",  
  "photo_url": "https://example.com/images/textbook-front.jpg",  
  "uploaded_at": "2025-11-18T10:40:00.123456",  
  "display_order": 0  
}
```

i) Use Case: Buyer views all photos of a listing

- Resource(s): listings, photos
- HTTP Method: GET
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/photos>
- Request Body: None
- Response:

```
{
  "photos": [
    {
      "id": "674b9999888877776666aaaa",
      "listing_id": "674b5678901234abcdef5678",
      "photo_url": "https://example.com/images/textbook-front.jpg",
      "uploaded_at": "2025-11-18T10:40:00.123456",
      "display_order": 0
    }
  ]
}
```

j) Use Case: Seller changes the order of photos

- Resource(s): listings, photos
- HTTP Method: PATCH
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/photos/674b9999888877776666aaaa>
- Request Body:

```
{
  "display_order": 1
}
```
- Response:

```
{
  "id": "674b9999888877776666aaaa",
  "listing_id": "674b5678901234abcdef5678",
  "photo_url": "https://example.com/images/textbook-front.jpg",
  "uploaded_at": "2025-11-18T10:40:00.123456",
  "display_order": 1
}
```

k) Use Case: Seller removes a photo from their listing

- Resource(s): listings, photos
- HTTP Method: DELETE
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/photos/674b9999888877776666aaaa>
- Request Body: None
- Response:

```
{  
  "message": "Photo deleted successfully"  
}
```

I) Use Case: Buyer contacts seller about a listing

- Resource(s): listings, threads
- HTTP Method: POST
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/threads>
- Request Body:

```
{  
  "buyer_id": "674b9876543210fedcba9876"  
}
```
- Response:

```
{  
  "id": "674baaaa999988887777bbbb",  
  "listing_id": "674b5678901234abcdef5678",  
  "buyer_id": "674b9876543210fedcba9876",  
  "seller_id": "674b1234567890abcdef1234",  
  "created_at": "2025-11-18T10:45:00.123456",  
  "last_activity_time": "2025-11-18T10:45:00.123456"  
}
```

m) Use Case: Seller views all chat threads for their listing

- Resource(s): listings, threads
- HTTP Method: GET
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/threads>

• Request Body: None

• Response:

```
{
  "threads": [
    {
      "id": "674baaaa999988887777bbbb",
      "listing_id": "674b5678901234abcdef5678",
      "buyer_id": "674b9876543210fedcba9876",
      "seller_id": "674b1234567890abcdef1234",
      "created_at": "2025-11-18T10:45:00.123456",
      "last_activity_time": "2025-11-18T10:45:00.123456"
    }
  ]
}
```

n) Use Case: Buyer or seller views a specific chat thread

- Resource(s): listings, threads
- HTTP Method: GET
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/threads/674baaaa999988887777bbbb>
- Request Body: None
- Response:

```
{
  "id": "674baaaa999988887777bbbb",
  "listing_id": "674b5678901234abcdef5678",
  "buyer_id": "674b9876543210fedcba9876",
  "seller_id": "674b1234567890abcdef1234",
  "created_at": "2025-11-18T10:45:00.123456",
  "last_activity_time": "2025-11-18T10:45:00.123456"
}
```

o) Use Case: Update last activity time when new message is sent

- Resource(s): listings, threads
- HTTP Method: PATCH
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/threads/674baaaa999988887777bbbb>
- Request Body:
{
- Response:
{
 "id": "674baaaa999988887777bbbb",
 "listing_id": "674b5678901234abcdef5678",
 "buyer_id": "674b9876543210fedcba9876",
 "seller_id": "674b1234567890abcdef1234",
 "created_at": "2025-11-18T10:45:00.123456",
 "last_activity_time": "2025-11-18T10:50:00.789012"
}

p) Use Case: User views all their chat threads

- Resource(s): threads
- HTTP Method: GET
- Request URL:
http://localhost:8080/api/threads?user_id=674b9876543210fedcba9876&page=1&per_page=10
- Request Body: None
- Response:

```
{
  "threads": [
    {
      "id": "674baaaa999988887777bbbb",
      "listing_id": "674b5678901234abcdef5678",
      "buyer_id": "674b9876543210fedcba9876",
      "seller_id": "674b1234567890abcdef1234",
      "created_at": "2025-11-18T10:45:00.123456",
      "last_activity_time": "2025-11-18T10:50:00.789012"
    }
  ],
  "page": 1,
  "per_page": 10,
  "total": 1
}
```

q) Use Case: User views a specific thread directly

- Resource(s): threads
- HTTP Method: GET
- Request URL:
- <http://localhost:8080/api/threads/674baaaa999988887777bbbb>
- Request Body: None
- Response:

```
{
  "id": "674baaaa999988887777bbbb",
  "listing_id": "674b5678901234abcdef5678",
  "buyer_id": "674b9876543210fedcba9876",
  "seller_id": "674b1234567890abcdef1234",
  "created_at": "2025-11-18T10:45:00.123456",
  "last_activity_time": "2025-11-18T10:50:00.789012"
}
```

r) Use Case: Attempt to create listing without required field

- Resource(s): listings
- HTTP Method: POST
- Request URL:
<http://localhost:8080/api/listings>
- Request Body:

```
{  
  "seller_id": "674b1234567890abcdef1234",  
  "title": "Test Item"  
}
```
- Response:

```
{  
  "error": "Missing required field: price"  
}
```

s) Use Case: Attempt to retrieve a listing that doesn't exist

- Resource(s): listings
- HTTP Method: GET
- Request URL:
<http://localhost:8080/api/listings/000000000000000000000000>
- Request Body: None
- Response:

```
{  
  "error": "Listing not found"  
}
```

t) Use Case: Buyer attempts to create another thread for same listing

- Resource(s): listings, threads
- HTTP Method: POST
- Request URL:
<http://localhost:8080/api/listings/674b5678901234abcdef5678/threads>
- Request Body:

```
{
  "buyer_id": "674b9876543210fedcba9876"
}
```
- Response:

```
{
  "error": "Thread already exists",
  "thread": {
    "id": "674baaaa999988887777bbbb",
    "listing_id": "674b5678901234abcdef5678",
    "buyer_id": "674b9876543210fedcba9876",
    "seller_id": "674b1234567890abcdef1234",
    "created_at": "2025-11-18T10:45:00.123456",
    "last_activity_time": "2025-11-18T10:50:00.789012"
  }
}
```

thread resource & review resource

Thread Resource & Review Resource:

1) Use Case: Buyer initiates a conversation with a seller about a listing

- Resource(s): threads
- HTTP Method: POST
- Request URL:
<http://localhost:8080/api/listings/692157e5c803ec7283549476/threads>
- Request Body:

```
{
  "buyer_id": "692157c469eeb4e203aaee3f"
}
```
- Response:

```
{
  "id": "69215858c4570cd6742bd49b",
  "listing_id": "692157e5c803ec7283549476",
  "buyer_id": "692157c469eeb4e203aaee3f",
  "seller_id": "692157c469eeb4e203aaee3e",
  "created_at": "2025-11-22T06:29:44.758258",
  "last_activity_time": "2025-11-22T06:29:44.758265"
}
```

2) Use Case: Seller views all chat threads for their listing

- Resource(s): threads
- HTTP Method: GET
- Request URL:
<http://localhost:8080/api/listings/692157e5c803ec7283549476/threads>
- Request Body: None
- Response:

```
{
  "threads": [
    {
      "id": "69215858c4570cd6742bd49b",
      "listing_id": "692157e5c803ec7283549476",
      "buyer_id": "692157c469eeb4e203aaee3f",
      "seller_id": "692157c469eeb4e203aaee3e",
      "created_at": "2025-11-22T06:29:44.758000",
      "last_activity_time": "2025-11-22T06:29:44.758000"
    }
  ]
}
```


3) Use Case: Retrieve details of a specific conversation

- Resource(s): threads
- HTTP Method: GET
- Request URL:
<http://localhost:8080/api/listings/692157e5c803ec7283549476/threads/69215858c4570cd6742bd49b>
- Request Body: None
- Response:

```
{
  "id": "69215858c4570cd6742bd49b",
  "listing_id": "692157e5c803ec7283549476",
  "buyer_id": "692157c469eeb4e203aaee3f",
  "seller_id": "692157c469eeb4e203aaee3e",
  "created_at": "2025-11-22T06:29:44.758000",
  "last_activity_time": "2025-11-22T06:29:44.758000"
}
```

4) Use Case: Update thread activity time when messages are sent

- Resource(s): threads
- HTTP Method: PATCH
- Request URL:
<http://localhost:8080/api/listings/692157e5c803ec7283549476/threads/69215858c4570cd6742bd49b>
- Request Body:

```
{
```
- Response:

```
{  
  "id": "69215858c4570cd6742bd49b",  
  "listing_id": "692157e5c803ec7283549476",  
  "buyer_id": "692157c469eeb4e203aaee3f",  
  "seller_id": "692157c469eeb4e203aaee3e",  
  "created_at": "2025-11-22T06:29:44.758000",  
  "last_activity_time": "2025-11-22T06:29:44.874604"  
}
```

5) Use Case: User views all their conversations (as buyer or seller)

- Resource(s): threads
- HTTP Method: GET
- Request URL:
http://localhost:8080/api/threads?user_id=692157c469eeb4e203aaee3f
- Request Body: None
- Response:

```
{
  "threads": [
    {
      "id": "69215858c4570cd6742bd49b",
      "listing_id": "692157e5c803ec7283549476",
      "buyer_id": "692157c469eeb4e203aaee3f",
      "seller_id": "692157c469eeb4e203aaee3e",
      "created_at": "2025-11-22T06:29:44.758000",
      "last_activity_time": "2025-11-22T06:29:44.874000"
    }
  ],
  "page": 1,
  "per_page": 10,
  "total": 1
}
```

6) Use Case: Direct access to a thread without listing context

- Resource(s): threads
- HTTP Method: GET
- Request URL: <http://localhost:8080/api/threads/69215858c4570cd6742bd49b>
- Request Body: None
- Response:

```
{
  "id": "69215858c4570cd6742bd49b",
  "listing_id": "692157e5c803ec7283549476",
  "buyer_id": "692157c469eeb4e203aaee3f",
  "seller_id": "692157c469eeb4e203aaee3e",
  "created_at": "2025-11-22T06:29:44.758000",
  "last_activity_time": "2025-11-22T06:29:44.874000"
}
```

7) Use Case: Buyer leaves feedback for a seller after a transaction

- Resource(s): reviews
- HTTP Method: POST
- Request URL: <http://localhost:8080/api/reviews>
- Request Body:

```
{
  "listing_id": "692157e5c803ec7283549476",
  "reviewer_id": "692157c469eeb4e203aeee3f",
  "reviewee_id": "692157c469eeb4e203aeee3e",
  "rating": 5,
  "comment": "Great seller! Item was exactly as described."
}
```
- Response:

```
{
  "id": "69215858c4570cd6742bd49c",
  "listing_id": "692157e5c803ec7283549476",
  "reviewer_id": "692157c469eeb4e203aeee3f",
  "reviewee_id": "692157c469eeb4e203aeee3e",
  "rating": 5,
  "comment": "Great seller! Item was exactly as described.",
  "helpful_count": 0,
  "created_at": "2025-11-22T06:29:44.940825",
  "updated_at": "2025-11-22T06:29:44.940828",
  "is_flagged": false,
  "moderation_status": "Pending"
}
```

8) Use Case: Admin views all reviews in the system for moderation

- Resource(s): reviews
- HTTP Method: GET
- Request URL: <http://localhost:8080/api/reviews>
- Request Body: None
- Response:

```
{
  "reviews": [
    {
      "id": "69215858c4570cd6742bd49c",
      "listing_id": "692157e5c803ec7283549476",
      "reviewer_id": "692157c469eeb4e203aeee3f",
      "reviewee_id": "692157c469eeb4e203aeee3e",
      "rating": 5,
      "comment": "Great seller! Item was exactly as described.",
      "helpful_count": 0,
      "created_at": "2025-11-22T06:29:44.940000",
      "updated_at": "2025-11-22T06:29:44.940000",
      "is_flagged": false,
      "moderation_status": "Pending"
    }
  ],
  "page": 1,
  "per_page": 10,
  "total": 1,
  "total_pages": 1
}
```

9) Use Case: View details of a particular review

- Resource(s): reviews
- HTTP Method: GET
- Request URL: <http://localhost:8080/api/reviews/69215858c4570cd6742bd49c>
- Request Body: None
- Response:

```
{
  "id": "69215858c4570cd6742bd49c",
  "listing_id": "692157e5c803ec7283549476",
  "reviewer_id": "692157c469eeb4e203aeee3f",
  "reviewee_id": "692157c469eeb4e203aeee3e",
  "rating": 5,
  "comment": "Great seller! Item was exactly as described.",
  "helpful_count": 0,
  "created_at": "2025-11-22T06:29:44.940000",
  "updated_at": "2025-11-22T06:29:44.940000",
  "is_flagged": false,
  "moderation_status": "Pending"
}
```

10) Use Case: Admin approves review and users mark it as helpful

- Resource(s): reviews
- HTTP Method: PATCH
- Request URL: <http://localhost:8080/api/reviews/69215858c4570cd6742bd49c>
- Request Body:

```
{
  "moderation_status": "Approved",
  "helpful_count": 1
}
```
- Response:

```
{
  "id": "69215858c4570cd6742bd49c",
  "listing_id": "692157e5c803ec7283549476",
  "reviewer_id": "692157c469eeb4e203aeee3f",
  "reviewee_id": "692157c469eeb4e203aeee3e",
  "rating": 5,
  "comment": "Great seller! Item was exactly as described.",
  "helpful_count": 1,
  "created_at": "2025-11-22T06:29:44.940000",
  "updated_at": "2025-11-22T06:29:45.046874",
  "is_flagged": false,
  "moderation_status": "Approved"
}
```


11) Use Case: View all approved reviews for a specific listing

- Resource(s): reviews, listings
- HTTP Method: GET
- Request URL:
<http://localhost:8080/api/listings/692157e5c803ec7283549476/reviews>
- Request Body: None
- Response:

```
{
  "reviews": [
    {
      "id": "69215858c4570cd6742bd49c",
      "listing_id": "692157e5c803ec7283549476",
      "reviewer_id": "692157c469eeb4e203aeee3f",
      "reviewee_id": "692157c469eeb4e203aeee3e",
      "rating": 5,
      "comment": "Great seller! Item was exactly as described.",
      "helpful_count": 1,
      "created_at": "2025-11-22T06:29:44.940000",
      "updated_at": "2025-11-22T06:29:45.046000",
      "is_flagged": false,
      "moderation_status": "Approved"
    }
  ]
}
```

12) Use Case: View a user's reputation (all reviews received)

- Resource(s): reviews, users
- HTTP Method: GET
- Request URL:
<http://localhost:8080/api/users/692157c469eeb4e203aeee3e/reviews>
- Request Body: None
- Response:

```
{
  "reviews": [
    {
      "id": "69215858c4570cd6742bd49c",
      "listing_id": "692157e5c803ec7283549476",
      "reviewer_id": "692157c469eeb4e203aeee3f",
      "reviewee_id": "692157c469eeb4e203aeee3e",
      "rating": 5,
      "comment": "Great seller! Item was exactly as described.",
      "helpful_count": 1,
      "created_at": "2025-11-22T06:29:44.940000",
      "updated_at": "2025-11-22T06:29:45.046000",
      "is_flagged": false,
      "moderation_status": "Approved"
    }
  ],
  "total_reviews": 1,
  "average_rating": 5.0
}
```

13) Use Case: Admin removes inappropriate review (soft delete)

- Resource(s): reviews
- HTTP Method: DELETE
- Request URL: <http://localhost:8080/api/reviews/69215858c4570cd6742bd49c>
- Request Body: None
- Response:

```
{  
  "message": "Review deleted successfully"  
}
```

Authentication & Access Control

Authentication & Access Control:

a) Use Case: Create a new user account (Registration)

- Resource(s): users
- HTTP Method: POST
- Request URL: <http://localhost:8080/api/users>
- Request Body:

```
{
  "name": "Alice Smith",
  "email": "alice@university.edu",
  "password": "alice123",
  "role": "Seller"
}
```
- Response:

```
{
  "message": "User created successfully",
  "user": {
    "id": "692a1f49194999909c33d64a",
    "name": "Alice Smith",
    "email": "alice@university.edu",
    "role": "Seller",
    "verified": false,
    "created_at": "2025-11-28T22:16:41.326931"
  }
}
```

b) Use Case: User login - creates a JWT session token

- Resource(s): sessions
- HTTP Method: POST
- Request URL: <http://localhost:8080/api/sessions>
- Request Body:

```
{  
  "email": "alice@university.edu",  
  "password": "alice123"  
}
```

- Response:

```
{  
  "token":  
    "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyX2lkIjoiaWkyYTFmNDkxOTQ5OTk5MDIjMzNkNjRhliwiZXhwljoxNzY0NDU1MjY5LCJpYXQiOiE3NjQzNjg4NjIjLjCIQel4jya53gwKLu0EE5O54upEg3db6FRsUJXNkBhL0",  
  "user": {  
    "id": "692a1f49194999909c33d64a",  
    "name": "Alice Smith",  
    "email": "alice@university.edu",  
    "role": "Seller",  
    "verified": false,  
    "created_at": "2025-11-28T22:16:41.326000"  
  },  
  "message": "Login successful"  
}
```

c) Use Case: User logout - invalidates the current session

- Resource(s): sessions
- HTTP Method: DELETE
- Request URL: <http://localhost:8080/api/sessions>
- Request Body: None
- Response:

```
{  
  "message": "Logout successful. Please remove the token from client  
storage."  
}
```

d) Use Case: Try to create listing without authentication token

- Resource(s): listings
- HTTP Method: POST
- Request URL: <http://localhost:8080/api/listings>
- Request Body:

```
{  
  "title": "Test Listing",  
  "description": "This should fail",  
  "price": 10.00,  
  "category": "Textbooks",  
  "condition": "Good"  
}
```

- Response:

```
{  
  "error": "Authentication required"  
}
```


e) Use Case: Try to create listing as buyer (wrong role)

- Resource(s): listings
- HTTP Method: POST
- Request URL: `http://localhost:8080/api/listings`
- Request Body:

```
{  
  "title": "Test Listing from Buyer",  
  "description": "This should fail",  
  "price": 10.00,  
  "category": "Textbooks",  
  "condition": "Good"  
}
```

- Response:

```
{  
  "error": "User is not authorized to sell"  
}
```

f) Use Case: Create a new listing as authenticated seller

- Resource(s): listings
- HTTP Method: POST
- Request URL: `http://localhost:8080/api/listings`
- Request Body:

```
{
  "title": "Calculus Textbook - 8th Edition",
  "description": "Barely used calculus textbook in excellent condition",
  "price": 45.00,
  "category": "Textbooks",
  "condition": "Like New",
  "status": "Published"
}
```
- Response:

```
{
  "id": "692a226a194999909c33d64c",
  "seller_id": "692a1f49194999909c33d64a",
  "title": "Calculus Textbook - 8th Edition",
  "description": "Barely used calculus textbook in excellent condition",
  "price": 45.0,
  "category": "Textbooks",
  "condition": "Like New",
  "status": "Published",
  "created_at": "2025-11-28T22:30:02.351730",
  "updated_at": "2025-11-28T22:30:02.351736"
}
```

g) Use Case: Update a listing as the owner

- Resource(s): listings
- HTTP Method: PATCH
- Request URL: <http://localhost:8080/api/listings/692a226a194999909c33d64c>
- Request Body:

```
{
  "price": 40.00,
  "description": "Barely used calculus textbook in excellent condition -
Price reduced!"
}
```
- Response:

```
{
  "id": "692a226a194999909c33d64c",
  "seller_id": "692a1f49194999909c33d64a",
  "title": "Calculus Textbook - 8th Edition",
  "description": "Barely used calculus textbook in excellent condition -
Price reduced!",
  "price": 40.0,
  "category": "Textbooks",
  "condition": "Like New",
  "status": "Published",
  "created_at": "2025-11-28T22:30:02.351730",
  "updated_at": "2025-11-28T22:35:15.123456"
}
```

h) Use Case: Add a photo to a listing

- Resource(s): photos
- HTTP Method: POST
- Request URL:
<http://localhost:8080/api/listings/692a226a194999909c33d64c/photos>
- Request Body:

```
{  
  "photo_url": "https://example.com/photos/calculus-book-cover.jpg",  
  "display_order": 0  
}
```
- Response:

```
{  
  "id": "692a2345194999909c33d64d",  
  "listing_id": "692a226a194999909c33d64c",  
  "photo_url": "https://example.com/photos/calculus-book-cover.jpg",  
  "uploaded_at": "2025-11-28T22:40:12.456789",  
  "display_order": 0  
}
```

i) Use Case: Start a chat thread with seller

- Resource(s): threads
- HTTP Method: POST
- Request URL:
<http://localhost:8080/api/listings/692a226a194999909c33d64c/threads>
- Request Body: None
- Response:

```
{
  "id": "692a2456194999909c33d64e",
  "listing_id": "692a226a194999909c33d64c",
  "buyer_id": "692a2129194999909c33d64b",
  "seller_id": "692a1f49194999909c33d64a",
  "created_at": "2025-11-28T22:45:30.789012",
  "last_activity_time": "2025-11-28T22:45:30.789012"
}
```

j) Use Case: Get a specific thread as participant

- Resource(s): threads
- HTTP Method: GET
- Request URL: <http://localhost:8080/api/threads/692a2456194999909c33d64e>
- Request Body: None
- Response:

```
{
  "id": "692a2456194999909c33d64e",
  "listing_id": "692a226a194999909c33d64c",
  "buyer_id": "692a2129194999909c33d64b",
  "seller_id": "692a1f49194999909c33d64a",
  "created_at": "2025-11-28T22:45:30.789012",
  "last_activity_time": "2025-11-28T22:45:30.789012"
}
```

k) Use Case: Get all threads for current authenticated user

- Resource(s): threads
- HTTP Method: GET
- Request URL: http://localhost:8080/api/threads?page=1&per_page=10
- Request Body: None
- Response:

```
{
  "threads": [
    {
      "id": "692a2456194999909c33d64e",
      "listing_id": "692a226a194999909c33d64c",
      "buyer_id": "692a2129194999909c33d64b",
      "seller_id": "692a1f49194999909c33d64a",
      "created_at": "2025-11-28T22:45:30.789012",
      "last_activity_time": "2025-11-28T22:45:30.789012"
    }
  ],
  "page": 1,
  "per_page": 10,
  "total": 1
}
```

I) Use Case: Create a review after transaction

- Resource(s): reviews
- HTTP Method: POST
- Request URL: <http://localhost:8080/api/reviews>
- Request Body:

```
{
  "listing_id": "692a226a194999909c33d64c",
  "reviewee_id": "692a1f49194999909c33d64a",
  "rating": 5,
  "comment": "Excellent seller! Book was in perfect condition as
described."
}
```
- Response:

```
{
  "id": "692a2567194999909c33d64f",
  "listing_id": "692a226a194999909c33d64c",
  "reviewer_id": "692a2129194999909c33d64b",
  "reviewee_id": "692a1f49194999909c33d64a",
  "rating": 5,
  "comment": "Excellent seller! Book was in perfect condition as
described.",
  "helpful_count": 0,
  "created_at": "2025-11-28T22:50:45.234567",
  "updated_at": "2025-11-28T22:50:45.234567",
  "is_flagged": false,
  "moderation_status": "Pending"
}
```