

- 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](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](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](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": "674b999988887776666aaaa",  
    "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": "674b999988887776666aaaa",  
      "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/674b99998888777>

76666aaaa

- 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/674baaaa99998887777bbbb>
- 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/674baaaa99998887777bbbb>
- 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](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"  
    }  
}
```