**Bước 1: Truy Cập Swagger UI**

* **Orchestrator Service**: http://localhost:8080/swagger-ui.html

**Bước 2: Test Start Attempt SAGA**

1. **Tìm endpoint**: POST /saga/start-attempt
2. **Click "Try it out"**
3. **Nhập Request Body**:

{

  "assignmentId": 1

}

1. **Click "Execute"**

**Bước 3: Quan Sát Response**

Bây giờ bạn sẽ thấy response đầy đủ:

{

  "attemptId": 1001,

  "status": "PENDING"

}

**Bước 4: Test Submit Attempt SAGA**

1. **Sử dụng attemptId** từ bước trước
2. **Tìm endpoint**: POST /saga/submit-attempt
3. **Nhập Request Body**:

{

  "attemptId": 1001,

  "answers": [

    {

      "questionId": 1,

      "selectedOption": "correct"

    },

    {

      "questionId": 2,

**🔍 Kiểm Tra Logs:**

Trong các terminal, bạn sẽ thấy chi tiết SAGA flow:

* **Orchestrator Service**: Logs về SAGA coordination
* **Assignment Service**: Logs về eligibility check và reservation
* **Attempt Service**: Logs về attempt creation và management

**📊 Kiểm Tra Database:**

Sau khi test xong, kiểm tra database:

*-- Assignment Database*

USE assignment\_db;

SELECT \* FROM quiz\_assignment;

SELECT \* FROM assignment\_attempt\_counter;

*-- Attempt Database*

USE attempt\_db;

SELECT \* FROM quiz\_attempt;

SELECT \* FROM user\_answer;

Bây giờ bạn có thể demo SAGA Pattern hoàn chỉnh với response đầy đủ cả attemptId và status!  
  
  
Perfect! Database đã được update đúng. Bây giờ hãy test các scenarios khác:

**2. Test Idempotency - Cùng Assignment Nhiều Lần:**

Bạn có thể test lại assignment ID = 2 (còn 2 attempts nữa):**Trong Swagger UI Orchestrator Service:**

1. **Test Start Attempt** với {"assignmentId": 2} lần nữa
2. **Quan sát:** Sẽ tạo attempt mới với ID khác
3. **Kiểm tra database:** used sẽ tăng lên 2

**3. Test Compensation Logic - Tạo Lỗi:**

**Scenario A: Stop Assignment Service**

1. **Stop Assignment Service** (Ctrl+C trong terminal)
2. **Test Start Attempt** với {"assignmentId": 3}
3. **Quan sát logs:** Sẽ thấy compensation logic
4. **Restart Assignment Service**

**Scenario B: Test Assignment Hết Hạn**

1. **Test với Assignment ID = 1** (đã hết hạn)
2. **Quan sát:** Sẽ bị lỗi "Assignment has closed"

**4. Test Edge Cases:**

**Test với Assignment ID không tồn tại:**

{

  "assignmentId": 999

}

**Test với Answers không hợp lệ:**

{

  "attemptId": 3,

  "answers": []

}

**🎯 Hiểu Rõ SAGA Pattern từ Logs:**

**SAGA Benefits:**

* ✅ **Atomicity**: Tất cả steps thành công hoặc rollback
* ✅ **Consistency**: Data luôn consistent across services
* ✅ **Isolation**: Mỗi SAGA độc lập với nhau
* ✅ **Durability**: Kết quả được persist vào database

**Compensation Logic:**

Khi có lỗi, SAGA sẽ tự động:

1. **Cancel Attempt** (nếu đã tạo)
2. **Release Assignment** (nếu đã reserve)

**Idempotency:**

* Cùng idempotencyKey → Cùng kết quả
* Hỗ trợ retry an toàn
* Không tạo duplicate data

**🚀 Bước Tiếp Theo - Test Compensation:**

Hãy test compensation logic bằng cách:

1. **Mở Swagger UI**: http://localhost:8080/swagger-ui.html
2. **Test Start Attempt** với {"assignmentId": 3}
3. **Trong khi đang process**, **stop Assignment Service**
4. **Quan sát logs** để thấy compensation
5. **Restart Assignment Service** và test lại

Bạn muốn test scenario nào tiếp theo?