

# Xử lý debugger khi agent ra bug

## 1. Sử dụng lệnh `/orchestrate` (Khuyến nghị cho lỗi phức tạp)

Lệnh này sẽ kích hoạt chế độ điều phối, giúp hai Agent thảo luận và hỗ trợ cho nhau.

```
“ Prompt: /orchestrate @wso2-mi-specialist @debugger "API SyncUC83API.xml đang trả về lỗi 500 khi gọi backend. Hãy phân tra log MI, đối chiếu với references/UC-83.md để tìm nguyên nhân và sửa code."
```

## 2. Prompt trực tiếp gọi cả hai Agent

Cách này nhanh gọn khi bạn đã biết lỗi nằm ở đâu.

```
“ Prompt: @wso2-mi-specialist @debugger Trong file SyncUC83API.xml, tôi thấy PayloadFactory đang map sai giá trị requestId từ request. Hãy kiểm tra lại logic và sửa dựa trên UC-83.md."
```

## 3. Prompt khi lỗi không rõ ràng (Troubleshooting)

Khi bạn chỉ thấy lỗi mà không biết tại sao.

```
“ Prompt: @debugger @wso2-mi-specialist Tôi gửi request POST lên /api/v1/sync/uc-83 nhưng bị treo timeout. Hãy kiểm tra cấu hình Endpoint trong file XML và hướng dẫn tôi cách debug
```

# Tại sao sự kết hợp này lại hiệu quả?

- **@debugger**: Sẽ sử dụng kỹ năng `systematic-debugging` để đi qua 4 bước: (1) Thu thập bằng chứng, (2) Đưa ra giả thuyết, (3) Thử nghiệm, (4) Xác minh. Nó sẽ yêu cầu xem log MI (`carbon.log`) hoặc chạy các lệnh kiểm tra kết nối.
- **@wso2-mi-specialist**: Sẽ đảm bảo các giải pháp sửa lỗi mà `@debugger` đưa ra là đúng cú pháp XML của WSO2, sử dụng đúng Mediator và Scope (default, transport, axis2...).

**Mẹo nhỏ:** Để việc debug hiệu quả nhất, bạn hãy copy một đoạn log lỗi từ MI vào chat cùng với Prompt.

## NOTE

Tôi vẫn đang chờ bạn xác nhận **implementation\_plan.md** cho `UC-83`. Nếu bạn thấy kế hoạch đã ổn, hãy phản hồi "Proceed" hoặc "Xác nhận" để tôi bắt đầu tạo code XML đầu tiên cho bạn nhé!

Phiên bản #1

Được tạo 2026-03-05 03:36:26 UTC bởi Nam Đặng

Được cập nhật 2026-03-05 03:36:57 UTC bởi Nam Đặng