Event

-Event được phân phối bởi các module khi có 1 số hành động nhất định được kích hoạt. Ngoài các event riêng của nó, ứng dụng cho phép tự tạo event có thể được phân phối trong mã nguồn. Khi 1 event được phân phối, nó có thể chuyển dữ liệu cho bất kì observers nào được config để lắng nghe event đó

-Event có thể được phân phối bằng cách sử dụng class [Magento\Framework\Event\ManagerInterface](https://github.com/magento/magento2/blob/2.4/lib/internal/Magento/Framework/Event/ManagerInterface.php). Class này có thể được lấy thông qua dependency injection bằng cách định nghĩa dependency trong constructor

-Để dispatch 1 event, sử dụng dispatch function của event manager class và cung cấp cho nó tên của event mà bạn muốn dispatch cùng với 1 array data mà bạn muốn cung cấp cho observers

A screen shot of a computer program

AI-generated content may be incorrect.

-Custome events có thể được dispatch bằng cách pass 1 event name duy nhất cho event manager khi bạn gọi hàm dispatch. Event name được tham chiếu trong file events.xml của module, nơi mà bạn xác định observers nào sẽ phản ứng lại event đó

-Ví dụ, tạo event tên my\_module\_event\_after bằng cách khai báo file MyCompany/MyModule/etc/events.xml với nội dung như sau:

<config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="urn:magento:framework:Event/etc/events.xsd">

<event name="my\_module\_event\_after">

<observer name="my\_module\_event\_after\_observer" instance="MyCompany\MyModule\Observer\MyEvent"/>

</event>

</config>

-Thông thường, file events.xml sẽ được đặt bên dưới folder Module\_name/etc. Observers tương tác với events ở đây sẽ lắng nghe events 1 cách toàn cục. File events.xml cũng có thể được đặt dưới các folder Module\_name/etc/frontend hoặc Module\_name/etc/adminhtml để cấu hình các observers chỉ lắng nghe các sự kiện trong các vùng cụ thể

| **Area** | **File location** |
| --- | --- |
| global | <module-dir>/etc/events.xml |
| adminhtml | <module-dir>/etc/adminhtml/events.xml |
| crontab | <module-dir>/etc/crontab/events.xml |
| frontend | <module-dir>/etc/frontend/events.xml |
| graphql | <module-dir>/etc/graphql/events.xml |
| webapi\_rest | <module-dir>/etc/webapi\_rest/events.xml |
| webapi\_soap | <module-dir>/etc/webapi\_soap/events.xml |

Observers

-Observers là 1 loai class có thể ảnh hưởng tới các hành vi chung, performance hoặc thay đổi business logic. Observers được thực thi bất cứ khi nào event đang được nó lắng nghe được dispatch bởi event manager

-Để tạo 1 observers. Bạn phải đặt class file bên dưới folder Module\_name/Observer. Observers class của bạn nên implement [Magento\Framework\Event\ObserverInterface](https://github.com/magento/magento2/blob/2.4/lib/internal/Magento/Framework/Event/ObserverInterface.php) và định nghĩa hàm execute() của nó

A screenshot of a computer program

AI-generated content may be incorrect.

Một trong số những tính năng mạnh mẽ của observers đó là chúng có thể sử dụng các parameter được pass thông qua event khi nó được dispatch

A screen shot of a computer program

AI-generated content may be incorrect.

Observers có thể được cấu hình để lắng nghe các sự kiện hiện có trong file events.xml.

Observer xml element có các thuộc tính sau:

+name (required) – Tên của observer cho việc định nghĩa event

+instance(required) – Tên đầy đủ của class của observer

+disabled – xác định xem observer này có hoạt động hay không, giá trị mặc định là false

+shared – xác định lifestyle của class, giá trị mặc định là true

<?xml version="1.0"?>

<config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="urn:magento:framework:Event/etc/events.xsd">

<event name="my\_module\_event\_before">

<observer name="myObserverName" instance="MyCompany\MyModule\Observer\MyObserver" />

</event>

<event name="my\_module\_event\_after">

<observer name="myObserverName" instance="MyCompany\MyModule\Observer\AnotherObserver" />

</event>

</config>

-Trong ví dụ trên, ta gán observer MyObserver cho custome event my\_module\_event\_before và AnotherObserver cho my\_module\_event\_after

-Tên của các observers phải là duy nhất cho mỗi event. Điều này nghĩa là bạn không thể có 2 observers với cùng tên trong cùng 1 event definition.

-Nếu khai báo 2 observers trùng tên trong cùng 1 event definition thì ứng dụng sẽ hợp nhất 2 observers này lại và hoạt động theo load order, tức là module được định nghĩa sau trong file app/etc/config.php sẽ được sử dụng còn observer của module đầu tiên sẽ bị ghi đè

\*Các lưu ý khi sử dụng observer

-Cố gắng giữ cho observer **ngắn gọn và hiệu quả**, tránh thực hiện các phép tính phức tạp nếu có thể.

-Không nên chứa business logic trong observer

-Đăng ký observer trong phạm vi (scope) phù hợp

-Tránh vòng lặp sự kiện (cyclical event loop)

-Không phụ thuộc vào thứ tự gọi observer