Introduction to events for developers

Events are a method of notifying different parts of Aero of tasks that have either been completed or need completing right after something has occurred. Events are only fired when a specified listener has been assigned to it and appropriately notified.

The events are always declared first in any **Service Provider**, while the listeners are wrapped within the event class. This is because we might need a number of listeners for a given event, each responsible for carrying out different tasks.

See our section on events for more information.

Example of an event:

```
<?php

namespace Aero\Account\Events;

use Aero\Account\Models\Customer;
use Aero\Events\ManagedEvent;

class CustomerUpdated extends ManagedEvent
{
    public $customer;
    public $customerName;
    public $customerEmail;

    public static $variables = [
        'customer',
        'customerName',
        'customerEmail',
    ];

    public function __construct(Customer $customer)
    {
        $this->customer = $customer;
```

```
        $this->customerName = $customer->name;
        $this->customerEmail = $customer->email;
    }

    public function getNotifiable()
    {
        return $this->customer;
    }
}
```

Revision #1
Created 2026-09-02 13:28:13 UTC by Michael Price
Updated 2026-09-02 13:28:13 UTC by Michael Price