How do I complete the transaction in my payment driver?

Once the payment gateway has handled the customer inputting their secure details (such as card information), they will most likely be redirected back to the retailer's store. On doing so, the `complete` method is called on the payment driver, allowing for the transaction to be validated and capturing of the payment.

The `complete` method should return a `Aero\Payment\Responses\PaymentResponse` instance.

```
public function complete()
{
    $reference = $this->request->input('reference');

    // don't continue if a reference isn't provided
    abort_unless($reference, 404);

    $response = new \Aero\Payment\Responses\PaymentResponse();

    // the order total
    $amount = $this->order->total_rounded;

    // obtain the transaction from the 3rd party API
    $request = AcmePaymentsInc::getTransaction($reference);

    // validate if the currency matches
    if ($request->currency !== $this->order->currency_code) {
        $response->setError('Currency does not match.');

        return $response;
    }

    // validate if the amount matches
```

```
if ($request->amount !== $this->order->total_rounded) {
    $response->setError('Total amount does not match.');

    return $response;
}

// validate if the transaction status is "authorized"
if ($request->status !== AcmePaymentsInc::STATUS_AUTHORIZED) {
    $response->setError('The transaction was not authorized.');

    return $response;
}

// create the payment record in the aero database
$payment = $this->order->payments()->updateOrCreate([
    'reference' => $reference,
], [
    'id' => (string) \Illuminate\Support\Str::uuid(),
    'payment_method_id' => $this->method->getKey(),
    'state' => \Aero\Payment\Models\Payment::AUTHORIZED,
    'amount' => $amount,
    'currency_code' => $this->order->currency->code,
    'exchange_rate' => $this->order->currency->exchange_rate,
    'merchant_mode' => $this->getMerchantMode(),
]);

// request to capture the transaction using the 3rd party API
$capture = AcmePaymentsInc::captureTransaction($reference, $amount);

// check if the transaction status is "captured"
if ($capture->status !== AcmePaymentsInc::STATUS_CAPTURED) {
    // update the status of the aero payment to "failed"
    $payment->update([
        'state' => \Aero\Payment\Models\Payment::FAILED,
    ]);

    // set an error on the response
    $response->setError('The transaction failed to be captured.');

    return $response;
}
```

```
    // internally capture the amount for the payment
    $payment->capture([
        'amount' => $amount,
    ]);

    // mark the response as successful
    $response->setSuccessful(true);

    return $response;
}
```

Revision #1
Created 2026-09-02 13:28:40 UTC by Michael Price
Updated 2026-09-02 13:28:40 UTC by Michael Price