

# Callback Integration

- [Notifications via Callback URL](#)

# Notifications via Callback URL

Frontpayment will notify your system of the payment status via the `callback.callbackUrl` you provided in the initial request payload. This allows you to update the order status in your system accordingly.

The callback URL will include the following parameters as query strings:

## Available Parameters

Parameter	Description
<code>orderId</code>	The unique identifier for the order. (for Reservation, Regular Order and Subscription)
<code>status</code>	The current status of the order (e.g., <code>PAID</code> , <code>INVOICED</code> , <code>RESEVRED</code> , <code>CAPTURED</code> , <code>CHARGED</code> ).
<code>createdAt</code>	The timestamp when the order was created.
<code>paymentMethod</code>	The payment method user has selected.
<code>amount</code>	The order amount
<code>timestamp</code>	The current timestamp when the callback is sent.
<code>checksum</code>	A SHA256 hash for integrity verification.

**Example Callback URL:** `https://your-callback-url.com/callback?orderId=ODR123&status=success&paymentMethod=Visa&amount=100&createdAt=1755764131&timestamp=1755764131&checksum=abcdef123456...`

## Checksum Verification

To ensure the integrity and authenticity of the callback, you **must** verify the `checksum` included in the query string of any callback url. The checksum is generated using the following formula:

```
hash('sha256', $routeParams . $secretKey)
```

Where:

- `orderId` is the value of the `orderId` parameter from the callback URL.

- `status` is the value of the `status` parameter from the callback URL.
- `createdAt` is the value of the `createdAt` parameter from the callback URL.
- `amount` is the value of total order amount from the callback URL.
- `paymentMethod` is the `paymentMethod` used. example: visa, mastercard etc.
- `timestamp` timestamp is the current timestamp
- `secretKey` will be given by frontpayment.

### Example Verification (Conceptual):

```
// In your callback handler
$getParameters = $_GET;

//Or manually read each query string
$receivedOrderUuid = $getParameters['orderUuid'];
$receivedStatus = $getParameters['status'];
$receivedCreatedAt = $getParameters['createdAt'];
$receivedAmount = $getParameters['amount'];
$paymentMethod = $getParameters['paymentMethod'];
$receivedTimestamp = $getParameters['timestamp']; // You might also want to log/check this for freshness
$receivedChecksum = $getParameters['checksum'];

$secretKey = ""; // Given by Front Payment;

// Construct the string used to calculate the checksum
$concatenatedValues = "";
foreach($getParameters as $key => $value) {
    // Except checksum parameter
    if ($key == 'checksum') { continue; }
    $concatenatedValues .= $value;
}

$hashedKey = hash('sha256', $concatenatedValues . $secretKey);

if (!hash_equals($hashedKey, $receivedChecksum)) {
    return "Checksum verification failed.";
}

// Checksum is valid, process the callback data
// e.g., update order status in your database
return "Callback successfully processed.";
```

By verifying the checksum, you can confirm that the callback data has not been altered during transmission, enhancing the security of your integration.