

Resend Payment Link

The **Resend Payment Link** endpoint allows your system to resend the payment link associated with an existing order to a customer. This is useful when the original link was lost, not received, expired, or needs to be sent again.

You can resend the link via:

- **SMS** (requiring `countryCode` and `msisdn`)
- **Email** (requiring a valid `email` address)

Importantly, the SMS or email does **not** need to match the contact details already stored with the customer profile. The payment link can be sent to a **new phone number** or a **different email address**, making it flexible for scenarios where the customer wants to use an alternative contact method.

This ensures that the customer can always receive the payment link, even if their original phone number or email is unavailable.

You will find details about the endpoint URL, authentication, request structure, validation rules, and sample responses below.

Endpoint

```
POST https://demo-api.frontpayment.no/api/v1/connect/orders/resend/{{ORDER_UUID}}
```

Authorization

Include a **Bearer Token** in the `Authorization` header. You can obtain this token from **Front Payment**.

Example:

```
Authorization: Bearer YOUR_FRONTPAYMENT_BEARER_TOKEN
```

Request Payload

Send the following parameters as a JSON object in the request body:

```
{
  "countryCode": +47,
  "msisdn": "xxxxxxx",
  "email": "example-email@email.com"
}
```

Validation Rules

Make sure your request meets the following requirements:

Field	Type	Description
countryCode	string	Conditional Required Country code for the customer's phone number (e.g., "+47").
msisdn	string	Conditional Required Mobile Subscriber MSISDN Number (phone number). If you want to resend order payment link via customer phone number.
email	email	Conditional Required Customer's email address. If you want to resend order payment link via email.

Response

A successful request will return a `202` status with the following JSON payload:

```
{
  "status_code": 202,
  "status_message": "OK",
  "message": "resentOrderSuccessfully",
  "is_data": true,
  "data": null
}
```

API returns a `404` error, it means requested order with `ORDER_UUID` could not be found in our system.

```
{
  "status_code": 404,
  "status_message": "Not Found",
}
```

```
"message": "orderNotFound",
"is_error": false,
"errors": null
}
```

API returns a `417` error, it means requested payload is not valid.

```
{
  "status_code": 417,
  "status_message": "Client Error",
  "message": "payloadValidationErrors",
  "is_error": true,
  "errors": "Array"
}
```

API return a `400` error, it means your requested order is already `PAID` or `CANCELLED`.

```
{
  "status_code": 400,
  "status_message": "Conflict of Business Logic",
  "message": "orderStatusAlreadyPaid",
  "is_error": false,
  "errors": null
}
```

```
{
  "status_code": 400,
  "status_message": "Conflict of Business Logic",
  "message": "orderStatusAlreadyCancelled",
  "is_error": false,
  "errors": null
}
```

API returns a `510` error, it means something failed on the server side

```
{
  "status_code": 510,
  "status_message": "Execution Exception Occurred",
  "message": "Something Went Wrong",
  "is_error": true,
}
```

```
"errors": "Array"  
}
```

Revision #21

Created 8 January 2024 17:17:54 by Admin

Updated 29 January 2026 10:09:55 by Nayamot Ullah