/api-docs

Explore

DummyPay API

0.0.0 OAS 3.0

/api-docs

API documentation for DummyPay

Servers

http://localhost:8080

payment-controller

GET `/api/payment/{paymentId}` Get payment session

Returns the payment session view for the user containing amount, currency, status, merchantName and timestamps.

Parameters

Try it out

Name	Description

paymentId * required
`integer($int64)`
`(path)`
	Identifier of the payment session
	paymentId

Responses

Code	Description	Links
200	Payment session retrieved successfully	*No links*

Code **Description** **Links**

Media type

application/json

Controls `Accept` header.

Example Value Schema

```
{
  "sessionId": 0,
  "amount": 0,
  "currency": "string",
  "status": "OPEN",
  "merchantName": "string",
  "createdAt": "2025-12-11T21:46:45.092Z",
  "closedAt": "2025-12-11T21:46:45.092Z"
}
```

404 Payment not found *No links*

Media type

application/json

Example Value Schema

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.092Z"
}
```

500 Internal server error *No links*

Media type

application/json

Example Value Schema

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.093Z"
}
```

POST `/api/payment` Create a new payment session

Creates a new payment session. Required fields: amount (number), currency (string), merchant (string), updateWebhook (string). The updateWebhook is where DummyPay will POST webhook notifications when the payment status changes (see the pay/close endpoints for payload details).

Parameters

Try it out

No parameters

Request body required

application/json

New payment session payload

Example Value Schema

```
{
  "amount": 1099,
  "currency": "USD",
  "merchant": "acme-store",
  "updateWebhook": "https://merchant.example.com/webhooks/payment-update"
}
```

Responses

Code **Description** **Links**

200 Payment session created successfully *No links*

Media type

application/json

Controls `Accept` header.

Example Value Schema

```
{
  "sessionId": 123
}
```

400 Validation error in the request payload *No links*

Media type

application/json

Example Value Schema

```
{
  "errors": {
    "updateWebhook": "updateWebhook is required"
  }
}
```

500 Unexpected error *No links*

Code	Description	Links
	Media type application/json **Example Value** Schema	

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.095Z"
}
```

PUT `/api/payment/{paymentId}/pay` Mark payment as paid

Set a payment's status to 'paid'. On success, DummyPay issues a webhook POST to the 'updateWebhook' provided when the session was created. Webhook details: method=POST, Content-Type=application/json, payload schema: {"sessionId": number, "status": "PAID"}. Example webhook payload: {"sessionId":123,"status":"PAID"}.

Parameters

Try it out

Name	Description
paymentId * required `integer($int64)` *(path)*	paymentId

Responses

Code	Description	Links
200	Payment marked as paid Media type application/json Controls `Accept` header. **Example Value** Schema	*No links*

```
"string"
```

Code **Description** **Links**

400 Invalid request or invalid payment state *No links*

Media type

application/json

Example Value Schema

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.096Z"
}
```

404 Payment not found *No links*

Media type

application/json

Example Value Schema

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.097Z"
}
```

500 Internal server error *No links*

Media type

application/json

Example Value Schema

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.098Z"
}
```

PUT `/api/payment/{paymentId}/close` Mark payment as closed

Set a payment's status to 'closed'. On success, DummyPay issues a webhook POST to the 'updateWebhook' provided when the session was created. Webhook details: method=POST, Content-

Type=application/json, payload schema: {"sessionId": number, "status": "CLOSED"}. Example webhook payload: {"sessionId":123,"status":"CLOSED"}.

Parameters

Try it out

Name	Description
paymentId * required `integer($int64)` *(path)*	paymentId

Responses

Code	Description	Links
200	Payment marked as closed	*No links*

Media type

application/json

Controls `Accept` header.

Example Value Schema

```
"string"
```

Code	Description	Links
400	Invalid request	*No links*

Media type

application/json

Example Value Schema

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.099Z"
}
```

Code	Description	Links
404	Payment not found	*No links*

Media type

application/json

Code **Description** **Links**

Example Value Schema

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.099Z"
}
```

500 Internal server error *No links*

Media type

application/json

Example Value Schema

```
{
  "status": 0,
  "message": "string",
  "path": "string",
  "errorCode": "string",
  "timestamp": "2025-12-11T21:46:45.100Z"
}
```

payment-page-controller

GET `/payment/{paymentId}` Payment page

Renders the payment page for the given paymentId and attaches PaymentSessionUserViewDTO to the model. Lets user preform payment or canceling.

Parameters Try it out

Name	Description
paymentId * required `integer($int64)` *(path)*	paymentId

Responses

Code **Description** **Links**

200 Payment page (HTML) rendered successfully; model contains PaymentSessionUserViewDTO. *No links*

Media type

application/json

Controls `Accept` header.

Example Value Schema

```
{
  "sessionId": 0,
  "amount": 0,
  "currency": "string",
  "status": "OPEN",
  "merchantName": "string",
  "createdAt": "2025-12-11T21:46:45.101Z",
  "closedAt": "2025-12-11T21:46:45.101Z"
}
```

404 Payment not found *No links*

Media type

/

Example Value Schema

```
string
```

500 Internal server error *No links*

Media type

/

Example Value Schema

```
string
```

Schemas

```
ErrorResponse    {
    status
                             integer($int32)
    message
                             string
    path
                             string
    errorCode
                             string
    timestamp
                             string($date-time)
}
```

```
NewPaymentSessionDTO    {
    amount*
                             integer($int32)
                         example: 1099
                         Amount in the smallest currency unit

    currency*
                             string
                         example: USD
                         ISO currency code

    merchant*
                             string
                         example: acme-store
                         Merchant name

    updateWebhook*
                             string
                         example: https://merchant.example.com/webhooks/payment-update
                         Webhook URL for payment status updates

}
```

```
CreatedSessionResponseDTO    {
    description:         Response returned when a payment session is created

    sessionId
                             integer($int64)
                         example: 123
                         Identifier of the created session

}
```

```
ValidationErrorResponseDTO  {
    description:          Validation errors keyed by field name

    errors
                              {
                              description:          Map of field names to validation error
                                                    messages

                              < * >:
                                                        string
                                                    example: {"updateWebhook":"updateWebhook
                                                    is required"}
                                                    Map of field names to validation error
                                                    messages

                          }
                          example: OrderedMap { "updateWebhook": "updateWebhook is required"
                          }
}



PaymentSessionUserViewDTO  {
    sessionId
                              integer($int64)
    amount
                              integer($int32)
    currency
                              string
    status
                              string
                          Enum:

                              [ OPEN, PAID, CLOSED ]
    merchantName
                              string
    createdAt
                              string($date-time)
    closedAt
                              string($date-time)
}
```