

Fluffy Doodle - because payments should be easy

1.0.0

[Base URL: localhost:8080/v1/api]

This is a prototype transaction/payment restful webservice. If using editor.swagger.io there's obviously CORS issues when interacting with this directly but swagger-ui running locally will allow interaction with the API. I've included this file under swagger/

[Peter Turner - Website](#)

[Send email to Peter Turner](#)

Schemes

HTTP

API Payment handling



payments



POST /payments Create payment

Minimal payment object represented here. Full object in model/payment.go

Parameters

Try it out

Name

Description

body * required

Payment object that needs creating

(body)

Example Value Model

```
{
  "type": "Payment",
  "id": "09a8fe0d-e239-4aff-8098-7923eadd0444",
  "version": 0,
  "organisation_id": "12345678-1234-1234-1234-123456789abc"
}
```

Parameter content type

application/json

Responses

Response content type

application/json

Code

Description

201

Successfully created

400

Invalid json

405

Invalid payment type (has to be Payment for this demo)

PUT **/payments** Update an existing payment or create a new payment

Minimal payment object represented here. Full object in model/payment.go

Parameters

Try it out

Name

Description

body * required

(body)

Payment object that needs updating/creating

Example Value Model

```
{
  "type": "Payment",
  "id": "09a8fe0d-e239-4aff-8098-7923eadd0444",
  "version": 0,
  "organisation_id": "12345678-1234-1234-1234-123456789abc"
}
```

Parameter content type

application/json

Responses

Response content type

application/json

Code

Description

200

Payment found based on ID and updated

201

Successfully created

400

Invalid json

405

Invalid payment type (has to be Payment for this demo)

GET **/payments** List all payments

List all payments in the database (no pagination :))

Parameters

Try it out

No parameters

Responses

Response content type

application/json

Code

Description

| Code | Description |
|---|---------------------------|
| 200 | <i>Successfully found</i> |
| <div>Example Value Model</div> <pre>{ "type": "Payment", "id": "09a8fe0d-e239-4aff-8098-7923eadd0444", "version": 0, "organisation_id": "12345678-1234-1234-1234-123456789abc" }</pre> | |

GET

/payments/{paymentID}

Find payment

Returns a single payment

Parameters

Try it out

| Name | Description |
|--|-------------------------|
| paymentID ★ required integer(\$int64) <i>(path)</i> | ID of payment to return |

Responses

Response content type

application/json

| Code | Description |
|---|---------------------------|
| 200 | <i>Successfully found</i> |
| <div><div>Example Value</div><div>Model</div><pre>{ "type": "Payment", "id": "09a8fe0d-e239-4aff-8098-7923eadd0444", "version": 0, "organisation_id": "12345678-1234-1234-1234-123456789abc"}</pre></div> | |
| 404 | <i>Payment not found</i> |

DELETE

/payments/{paymentID}

Delete a payment

Parameters

Try it out

| Name | Description |
|--|----------------------|
| <div>paymentID ★ required</div> <div>integer(\$int64)</div> <div>(path)</div> | Payment id to delete |

| Responses | | Response content type |
|-----------|--------------------------------------|-----------------------|
| | | application/json |
| Code | Description | |
| 204 | <i>No content - deletion success</i> | |
| 404 | <i>Payment not found</i> | |

| Models | | ↙ |
|-----------------|---|---|
| Payment | { | ↶ |
| type | string | |
| | Enum: | |
| | [Payment, Holiday, Car, Insurance] | |
| id | string(\$string) | |
| | example: 09a8fe0d-e239-4aff-8098-7923eadd0444 | |
| version | integer(\$int32) | |
| organisation_id | string(\$string) | |
| | example: 12345678-1234-1234-1234-123456789abc | |
| | } | |