Credit Card API - Access Guide

Version: 0.0.1-SNAPSHOT

Table of Contents

Overview

Credit Card Api is a RestFul application built with Spring Boot and Embedded MongoDB It's based on Level 3 of the famous Richardson Maturity Model. This is one of the most discussed subjects about API design. This source code is hosted in https://github.com/jonyfs/credit-card-api.

Endpoints

Environment	Url
dev	http://localhost:8080/api
test	http://credit-card-api.testserver.com/api
production	https://creditcardapi.herokuapp.com/api

HTTP verbs

RESTful notes tries to adhere as closely as possible to standard HTTP and REST conventions in its use of HTTP verbs.

Verb	Usage
`GET`	Used to retrieve a resource
`POST`	Used to create a new resource

HTTP status codes

RESTful notes tries to adhere as closely as possible to standard HTTP and REST conventions in its use of HTTP status codes.

Status code	Usage
`200 OK`	The request completed successfully
`201 Created`	A new resource has been created successfully. The resource's URI is available from the response's `Location` header
`204 No Content`	An update to an existing resource has been applied successfully

Status code	Usage
`400 Bad Request`	The request was malformed. The response body will include an error providing further information
`422 Unprocessable Entity`	The request was well-formed but was unable to be followed due to semantic errors.
`404 Not Found`	The requested resource did not exist

Hypermedia

RESTful Notes uses hypermedia and resources include links to other resources in their responses. Responses are in Hypertext Application Language (HAL) format. Links can be found benath the `_links` key. Users of the API should not created URIs themselves, instead they should use the above-described links to navigate from resource to resource.

Resources

Index

The index provides the entry point into the service.

Accessing the index

A `GET` request is used to access the index

Example response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 33

0.0.1-SNAPSHOT - 2015-12-23 14:56
```

Payment

The payment resource provides the entry point to do a payment and get a payment.

Accessing the payment resource

version 1 (v1)

```
There is no documentation for this version
```

version 2 (v2)

```
There is no documentation for this version
```

version 3 (v3)

How get a Payment info

Path Parameters

Table 1. /api/v3/payments/{id}

Parameter	Description
id	The Credit Card Transaction ID.

Request

```
GET /api/v3/payments/567ab63f036465a9cbc74826 HTTP/1.1
Content-Type: application/json
Accept-Encoding: gzip
Host: localhost
```

Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 447

{
  "cardType" : "VISA",
  "cardNumber" : "4716651077977392",
  "expirationDate" : 1895969040000,
  "store" : {
    "name" : "Walmart"
  },
  "products" : [ {
    "name" : "Moto X 2015",
    "price" : 149.27
  }, {
    "name" : "Sandisk Cruzer 32GB Flash Drive",
    "price" : 9.99
  } ],
  "_links" : {
    "self" : {
      "href" : "http://localhost:8080/api/v3/payments/567ab63f036465a9cbc74826"
    }
  },
  "id" : "567ab63f036465a9cbc74826"
}
```

Response Fields

Path	Type	Description
id	String	The Credit Card Transaction ID.
cardType	String	Credit Card Type
cardNumber	String	Credit Card Number
expirationDate	Object	Credit Card Expiration Date
store	Array	Store
products	Array	Products Array
_links	Object	HATEOAS links

How send a Payment info to validate and confirm the payment transaction

When the transaction is commited the service return a transaction id

Request Fields

Path	Type	Description
id	String	The Credit Card Transaction ID.
cardType	String	Credit Card Type
cardNumber	String	Credit Card Number
expirationDate	Object	Credit Card Expiration Date
store	Array	Store
products	Array	Products Array

Request

```
POST /api/v3/payments HTTP/1.1
Content-Type: application/json
Accept-Encoding: gzip
Host: localhost
Content-Length: 304

{
  "cardType" : "VISA",
  "cardNumber" : "4716651077977392",
  "expirationDate" : 1895969040000,
  "store" : {
    "name" : "Walmart"
  },
  "products" : [ {
    "name" : "Moto X 2015",
    "price" : 149.27
  }, {
    "name" : "Sandisk Cruzer 32GB Flash Drive",
    "price" : 9.99
  } ],
  "id" : null
}
```

Response

```
HTTP/1.1 201 Created
Content-Type: application/json
Content-Length: 165

{
  "content" : "567ab63f036465a9cbc74827",
  "_links" : {
    "self" : {
      "href" : "http://localhost:8080/api/v3/payments/567ab63f036465a9cbc74827"
    }
  }
}
```

Payment data validation exception. This occurs with invalid payment paramters

Request

```
POST /api/v3/payments HTTP/1.1
Content-Type: application/json
Accept-Encoding: gzip
Host: localhost
Content-Length: 127

{
  "cardType" : null,
  "cardNumber" : null,
  "expirationDate" : null,
  "store" : null,
  "products" : null,
  "id" : null
}
```

Response

```
HTTP/1.1 422 Unprocessable Entity
Content-Type: application/json
Content-Length: 603

{
  "code" : "151cf57e88c",
  "exception" : "InvalidRequestException",
  "message" : "Invalid Payment",
  "errors" : [ {
    "code" : "NotNull",
    "message" : "may not be null",
    "parameter" : "cardNumber"
  }, {
    "code" : "NotNull",
    "message" : "may not be null",
    "parameter" : "cardType"
  }, {
    "code" : "NotNull",
    "message" : "may not be null",
    "parameter" : "expirationDate"
  }, {
    "code" : "NotNull",
    "message" : "may not be null",
    "parameter" : "store"
  }, {
    "code" : "NotEmpty",
    "message" : "may not be empty",
    "parameter" : "products"
  } ]
}
```