

Documentation Utilisateur Ynov Pizza

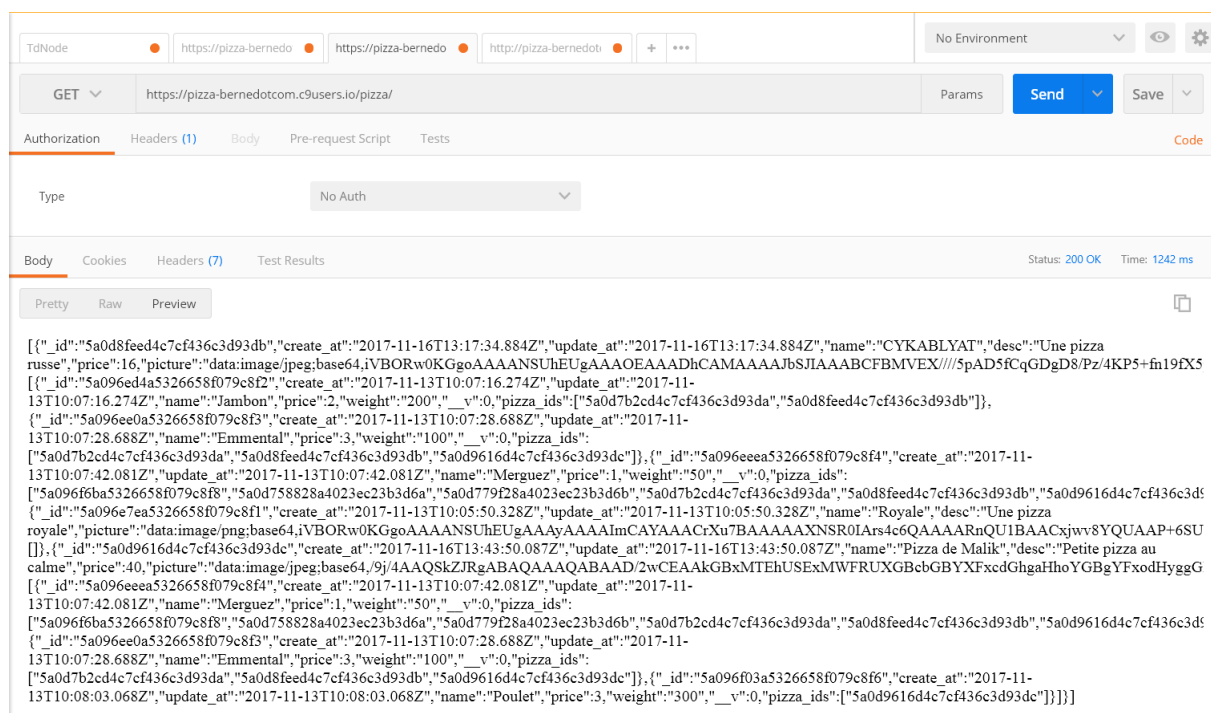
Introduction

Ce document à pour but de présenter la documentation utilisateur du projet Ynov Pizza développé sous NodeJs.

Il présentera les différentes routes proposées par l'API, ce qu'elles retournent comme données et surtout le moyen d'y accéder.

Test de l'API :

Pour tester l'API vous pouvez utiliser des utilitaires de type [Postman](#) afin de simuler des requêtes HTTP.



Exemple d'utilisation de Postman avec une requête GET

Routes Pizza

Voici les différentes routes concernant les pizzas proposées par l'API.

Récupérer toutes les pizzas :

<https://ide.c9.io/bernedotcom/pizza>

Récupérer une pizza grâce à son identifiant :

<https://ide.c9.io/bernedotcom/pizza/:idPizza>

Récupérer une pizza grâce à son nom :

[https://ide.c9.io/bernedotcom/pizza /name/:name](https://ide.c9.io/bernedotcom/pizza/name/:name)

Récupérer une pizza grâce à son prix :

<https://ide.c9.io/bernedotcom/pizza /price/:price>

Récupérer une pizza grâce à sa date de modification :

<https://ide.c9.io/bernedotcom/pizza /price/:price>

Récupérer une pizza grâce à son prix :

https://ide.c9.io/bernedotcom/pizza /update /:update_at

Récupérer une pizza grâce à l'un de ses ingrédients :

https://ide.c9.io/bernedotcom/pizza /ingredient/:ingredient_id

Ajouter une pizza nouvelle pizza :

The screenshot shows a REST client interface with a POST request to `http://pizza-bernedotcom.c9users.io/pizza`. The request body is a JSON object representing a pizza. The interface includes tabs for Authorization, Headers (1), Body, Pre-request Script, and Tests. The Body tab is selected, and the content type is set to `JSON (application/json)`. The request body is a JSON object with the following structure:

```
1 { "name": "Test",
2   "desc": "Une pizza de test",
3   "picture": " ",
4   "price": 15,
5   "ingredient_ids": [ "", "5a096e56a5326658f079c8ef", "5a096ef9a5326658f079c8f5", "5a096eeaa5326658f079c8f4" ] }
```

Exemple de création d'une pizza à l'aide d'une requête http de type postS

Mettre à jour une pizza existante :

PUT ▾	http://pizza-bernedotcom.c9users.io/pizza/5a0d8feed4c7cf436c3d93db	Params	Send ▾
Authorization	Headers (1)	Body ●	Pre-request Script Tests
<input type="radio"/> form-data <input type="radio"/> x-www-form-urlencoded <input checked="" type="radio"/> raw <input type="radio"/> binary JSON (application/json) ▾			
1 ▾	{ 2 "_id": "5a0d8feed4c7cf436c3d93db", 3 "name": "Test", 4 "desc": "Une pizza de test", 5 "picture": " ", 6 "price": 15, "ingredient_ids": ["", "5a096e56a5326658f079c8ef", "5a096ef9a5326658f079c8f5", "5a096eeea5326658f079c8f4"] }		

Exemple de mise à jour d'une pizza à l'aide d'une requete http de type put

Supprimer une pizza existante :

DELETE ▾	http://pizza-bernedotcom.c9users.io/pizza/5a0d8feed4c7cf436c3d93db	Params	Send ▾
Authorization	Headers (1)	Body ●	Pre-request Script Tests
Type	No Auth ▾		

Exemple de suppression d'une pizza à l'aide d'une requête http de type delete

Routes Ingrédient :

Voici les différentes routes concernant les ingrédients proposées par l'API.

Récupérer tous les ingrédients :

<https://ide.c9.io/bernedotcom/ingredient>

Récupérer un ingrédient grâce à son identifiant :

https://ide.c9.io/bernedotcom/ingredient/:ingredient_id

Récupérer un ingrédient grâce à son nom :

<https://ide.c9.io/bernedotcom/ingredient/name/:name>

Récupérer un ingrédient grâce à son prix :

<https://ide.c9.io/bernedotcom/ingredient/price/:price>

Récupérer un ingrédient grâce à son poids :

<https://ide.c9.io/bernedotcom/ingredient/weight/:weight>

Récupérer un ingrédient grâce à une pizza qui le contient :

<https://ide.c9.io/bernedotcom/ingredient/pizzald/:pizzald>

Créer un nouvel ingrédient :

The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** <http://pizza-bernedotcom.c9users.io/ingredient>
- Body:** A JSON object:

```
{ 1: { 2:   name: "Poulet", 3:   price: 2, 4:   weight: "100" 5: }}
```
- Headers:** The 'Body' tab is selected, showing the raw JSON body.
- Buttons:** 'Send' and a dropdown arrow are visible.

Exemple de création d'un nouvel ingrédient à l'aide d'une requête http de type post

Modification d'un ingrédient existant :

The screenshot shows a REST client interface with the following details:

- Method:** PUT
- URL:** <http://pizza-bernedotcom.c9users.io/ingredient/5a096ed4a5326658f079c8f2>
- Body:** A JSON object:

```
{ 1: { 2:   name: "Poulet", 3:   price: 3, 4:   weight: "200" 5: }}
```
- Headers:** The 'Body' tab is selected, showing the raw JSON body.
- Buttons:** 'Send' and a dropdown arrow are visible.

Exemple de modification d'un ingrédient à l'aide d'une requête http de type put

Suppression d'un ingrédient existant :

DELETE ▾	http://pizza-bernedotcom.c9users.io/ingredient/5a096ed4a5326658f079c8f2	Params	Send ▾	
Authorization	Headers (1)	Body ●	Pre-request Script	Tests
Type	No Auth ▾			

Suppression d'un ingrédient à l'aide d'une requête http de type delete