2025/11/04 22:17 1/9 Service Rest

# **Service Rest**

Open database API

## -- Connexion

Base URL: users/

URL	Méthode	Paramètres	Rôle	Réponses possibles
check	GET		Vérifie qu'un utilisateur est connecté	<pre>{token : ddd34,connected: true} {connected: false}</pre>
connect	POST	mail, password	Effectue la connection avec les paramètres postés	<pre>{token : ddd34,connected: true} {connected: false}</pre>
disconnect	GET		Déconnecte (ferme la session) de l'utilisateur actif	{connected: false}
exists/:mail	GET		Vérifie que l'email existe dans la base de données	<pre>{exists: true ,mail: "admin@local.fr"} {exists: false ,mail: "jcheron@local.fr"}</pre>
add	POST	mail, password	Ajoute l'utilisateur avec les paramètres postés	<pre>{token : ddd34,inserted: true} {inserted: false}</pre>
check/:mail/:password	GET		Pour tests : effectue une connexion avec les paramètres passé	<pre>{token : ddd34,connected: true} {connected: false}</pre>

## -- Brasseurs

## -- Liste des brasseurs

URL	Méthode	<b>Paramètres</b>	Rôle
/breweries	GET		Retourne tous les brasseurs

## **Exemple de réponse :**

URL: /breweries

```
[
    "id":"1",
    "name":"Anchor",
    "url":"http:\/\/www.anchorbrewing.com\/",
    "photo":"",
    "thumbnail":"",
    "created_at":"2015-03-06 19:11:39",
    "updated_at":"2015-03-06 21:53:12"
},
```

 $update: \\ 2019/08/31 \\ slam 4: rich client: angular js: project: open beer database: rest http://slam wiki2.kobject.net/slam 4/rich client/angular js/project/open beer database/rest? rev=1425859309$ 

```
{
        "id":"6",
        "name": "Dogfish Head",
        "url": "http:\/\/www.dogfish.com\/",
        "photo":"",
        "thumbnail":"",
        "created at": "2015-03-07 02:19:22",
        "updated_at":"2015-03-07 02:19:22"
    },
]
```

#### **Erreurs possibles:**

Code	Code Raison		Réponse					
404	Aucune brasserie trouvée dans la BDD	{code:404,	message:	"Aucune	brasserie	trouvée	."}	

## -- Brasseur correspondant à un id

URL	Méthode	Paramètres	Rôle
/breweries/:id	GET		Retourne le brasseur dont l'id est passé en paramètre

#### **Exemple de réponse :**

URL:/breweries/7

```
{
    "id":"7",
    "name": "Stone Brewing Co.",
    "url": "http:\/\/www.stonebrew.com\/",
    "photo":"",
    "thumbnail":"",
    "created_at":"2015-03-07 02:19:22",
    "updated_at":"2015-03-07 02:19:22"
}
```

#### **Erreurs possibles:**

Code	Raison	Réponse	
404	Aucune brasserie trouvée dans la BDD	{"message":"Ooops! Le brasseur 7 est	
404	correspondant à l'id passé en paramètre	<pre>introuvable","code":404}</pre>	

## -- ajout d'un brasseur

URL	Méthode	<b>Paramètres</b>	data	Rôle
				Ajoute le
/breweries	POST	token	{"name":"Fischer","url":"http://www.fischer.com"}	brasseur
				posté

## **Exemple de réponse :**

**URL**: /breweries

2025/11/04 22:17 3/9 Service Rest

Méthode: POST

```
{
    "data":{
        "id":"32",
        "name": "Fischer",
        "url": "http:\/\/www.fischer.com",
        "photo":null,
        "thumbnail":null,
        "created at": "2015-03-08 19:47:47",
        "updated at":"2015-03-08 19:47:47"
    },
    "message":{
        "type":"success",
        "content": "'Fischer' a \u00e9t\u00e9 correctement ajout\u00e9e dans les
brasseries."
    }
}
```

#### **Erreurs possibles:**

Cod	Raison	Réponse		
401	Utilisateur non connecté (token invalide ou absent et force non utilisé)	{"message":"Vous n'avez pas les droits pour ajouter une brasserie","code":401}		
409	Brasseur dont le nom existe déjà dans la base	{"message":"Impossible d'ajouter 'Fischer' dans la base de données","code":409}		

#### -- Modification d'un brasseur

URL	Méthode	<b>Paramètres</b>	data	Rôle
/breweries/:id	PUT	ITΛVΔN	₹"name":"Fischer	modifie le brasseur posté

## **Exemple de réponse :**

URL : /breweries/32 Méthode : PUT

```
{
   "data":{
        "id":"32",
        "name":"Fischer modifié",
        "url":"http:\/\/www.fischer.com",
        "photo":null,
        "thumbnail":null,
        "created_at":"2015-03-08 19:47:47",
        "updated_at":"2015-03-08 19:47:47"
},
   "message":{
        "type":"success",
        "content":"'Fischer' a \u000e9t\u00e9 correctement modifi\u00e9e dans les
```

```
brasseries."
}
```

#### **Erreurs possibles:**

Code	Raison	Réponse					
401		<pre>{"message":"Vous n'avez pas les droits pour modifier une prasserie","code":401}</pre>					
4114		{"message":"Suppression : La brasserie 'Fischer' n'existe plus dans la base de données","code":409}					
иna		{"message":"Impossible de modifier 'Fischer' dans la base de données","code":409}					

## -- Suppression d'un brasseur

URL	Méthode	Paramètres	data	Rôle
/breweries/:id	DELETE	token		supprime le brasseur dont l'id est passée dans l'URL

#### **Exemple de réponse :**

URL : /breweries/32 Méthode : DELETE

```
{
    "data":{
        "id":"32",
        "name":"Fischer",
        "url":"http:\/\/www.fischer.com",
        "photo":null,
        "thumbnail":null,
        "created_at":"2015-03-08 19:47:47",
        "updated_at":"2015-03-08 19:47:47"
    },
    "message":{
        "type":"success",
        "content":"'Fischer' a \u00e9t\u00e9 correctement supprim\u00e9e de
l'ensemble des brasseries."
    }
}
```

## **Erreurs possibles:**

Code	Raison	Réponse			
	Utilisateur non connecté (token invalide ou absent et force non utilisé)	{"message":"Vous n'avez pas les droits pour supprimer une brasserie","code":401}			
	Le brasseur n'existe plus dans la base	{"message":"Suppression : La brasserie 'Fischer' n'existe plus dans la base de données","code":409}			
409	Brasseur ayant des bières associées	{"message":"Impossible de supprimer 'Fischer' dans la base de données","code":409}			

2025/11/04 22:17 5/9 Service Rest

## -- Bières

#### -- Liste des bières

URL	Méthode	<b>Paramètres</b>	Rôle
/beers	GET		Retourne toutes les bières

#### **Exemple de réponse :**

URL:/beers

```
[
    {
        "id":"1",
        "name": "Strawberry Harvest",
        "description": "Strawberry Harvest Lager is a wheat beer ...",
        "abv":"4.2",
        "photo":null,
        "created_at":"2015-03-07 00:43:23",
        "updated_at":"2015-03-07 00:43:23",
        "idBrewery":"1"
    },
    {
        "id":"2",
        "name": "Brooklyn Detonation Ale",
        "description": "For about eight years now...",
        "abv":"bda",
        "photo":null,
        "created_at": "2015-03-08 21:06:46",
        "updated_at":"2015-03-08 21:06:46",
        "idBrewery":"12"
    },
]
```

## **Erreurs possibles:**

C	ode	Raison	Réponse				
4	04	Aucune bière trouvée dans la BDD	{code:404,	message:	"Aucune	bière	trouvée."}

## -- Bière correspondant à un id

URL	Méthode	<b>Paramètres</b>	Rôle
/beers/:id	GET		Retourne la bière dont l'id est passé en paramètre

#### **Exemple de réponse :**

URL:/beers/7

```
{
    "id":"1",
```

 $\frac{\text{upuate:}}{2019/08/31} s lam 4: rich client: angular js: project: open beer database: rest \ http://slam wiki2.kobject.net/slam 4/rich client/angular js/project/open beer database/rest? rev=1425859309$ 

```
"name": "Strawberry Harvest",
    "description": "Strawberry Harvest Lager is a wheat beer ...",
    "abv": "4.2",
    "photo":null,
    "created_at":"2015-03-07 00:43:23",
    "updated_at":"2015-03-07 00:43:23",
    "idBrewery": "1"
}
```

#### **Erreurs possibles:**

Cod	e Raison	Réponse
404		{"message":"0oops! La bière 7 est
	correspondant à l'id passé en paramètre	introuvable","code":404}

#### -- Liste des bières d'une brasserie

URL	Méthode	<b>Paramètres</b>	Rôle
/beers/brewery/:id	CET		Retourne les bières correspondant à la brasserie dont l'id est passé
/beers/brewery/.iu	GEI		en paramètre

#### **Exemple de réponse :**

URL: /beers/brewery/12

```
[
    {
        "id":"2",
        "name": "Brooklyn Detonation Ale",
        "description": "For about eight years now, we've brewed...",
        "abv": "bda",
        "photo":null,
        "created at": "2015-03-08 21:06:46",
        "updated at": "2015-03-08 21:06:46",
        "idBrewery":"12"
    },{
        "id":"3",
        "name": "Brooklyn Ama Bruna",
        "description": "Brewed by Amarcord Brewery in the small medieval town of
Apecchio, Italy...",
        "abv":"bab",
        "photo":null,
        "created_at":"2015-03-08 21:06:46",
        "updated at": "2015-03-08 21:06:46",
        "idBrewery": "12"
    }
```

## **Erreurs possibles:**

2025/11/04 22:17 7/9 Service Rest

Co	ode Raison	ode Raison	Réponse
40	Aucune bière trouvée dans la BDD correspondant correspondant à la brasserie dont l'id passé en paramètre	<b>04</b> correspo	{"message":"Ooops! La brasserie 12 ne dispose d'aucune bière","code":404}

## -- ajout d'une bière

URL	Méthode	<b>Paramètres</b>	data	Rôle
/beers	POST	token	R. DameFiccher	Ajoute la bière postée

## **Exemple de réponse :**

URL : /beers Méthode : POST

```
{
    "data":{
        "id":"3",
        "name": "Brooklyn Ama Bruna",
        "description": "Brewed by Amarcord Brewery in the small medieval town of
Apecchio...",
        "abv": "bab",
        "photo":null,
        "created_at":"2015-03-08 21:06:46",
        "updated_at": "2015-03-08 21:06:46",
        "idBrewery":"12"
    },
    "message":{
        "type": "success",
        "content":"'Brooklyn Ama Bruna' a \u00e9t\u00e9 correctement ajout\u00e9e
dans les bières."
    }
}
```

#### **Erreurs possibles:**

Code	Raison	Réponse
401	Utilisateur non connecté (token invalide ou absent et force non utilisé)	{"message":"Vous n'avez pas les droits pour ajouter une bière","code":401}
ZHU	Bière dont le nom existe déjà dans la base	{"message":"Impossible d'ajouter 'Fischer' dans la base de données","code":409}

## -- Modification d'une bière

URL	Méthode	<b>Paramètres</b>	data	Rôle
/beers/:id	PUT	token	modif", "description": "Brewed by Amarcord Brewery in the small medieval town of	modifie la bière postée

#### **Exemple de réponse :**

URL:/beers/3

Méthode: PUT

```
{
    "data":{
        "id":"3",
        "name": "Brooklyn Ama Bruna modif",
        "description": "Brewed by Amarcord Brewery in the small medieval town of
Apecchio...",
        "abv": "bab",
        "photo":null,
        "created at":"2015-03-08 21:06:46",
        "updated_at": "2015-03-08 21:06:46",
        "idBrewery":"12"
    },
    "message":{
        "type": "success",
        "content": "'Brooklyn Ama Bruna modif' a \u00e9t\u00e9 correctement
modifi\u00e9e dans les brasseries."
}
```

#### **Erreurs possibles:**

Code	Raison	Réponse
1		{"message":"Vous n'avez pas les droits pour modifier une bière","code":401}
404		{"message":"Suppression : La brasserie 'Brooklyn Ama Bruna' n'existe plus dans la base de données","code":409}
MAC I		{"message":"Impossible de modifier 'Brooklyn Ama Bruna' dans la base de données","code":409}

## -- Suppression d'une bière

URL	Méthode	Paramètres	data	Rôle
/beers/:id	DELETE	token		supprime la bière dont l'id est passée dans l'URL

#### **Exemple de réponse :**

URL : /beers/3 Méthode : DELETE

```
{
    "data":{
        "id":"3",
        "name":"Brooklyn Ama Bruna",
        "description":"Brewed by Amarcord Brewery in the small medieval town of
Apecchio...",
        "abv":"bab",
        "photo":null,
        "created_at":"2015-03-08 21:06:46",
```

2025/11/04 22:17 9/9 Service Rest

```
"updated_at":"2015-03-08 21:06:46",
    "idBrewery":"12"
    },
    "message":{
        "type":"success",
        "content":"'Brooklyn Ama Bruna' a \u00e9t\u00e9 correctement supprim\u00e9e
de l'ensemble des bières."
    }
}
```

## **Erreurs possibles:**

Code	Raison	Réponse
401		{"message":"Vous n'avez pas les droits pour supprimer une bière","code":401}
4114		{"message":"Suppression : La bière 'Brooklyn Ama Bruna' n'existe plus dans la base de données","code":409}
409	Aucune raison	{"message":"Impossible de supprimer 'Brooklyn Ama Bruna' dans la base de données","code":409}

From

http://slamwiki2.kobject.net/ - SlamWiki 2.1

Permanent link:

http://slamwiki2.kobject.net/slam4/richclient/angularjs/project/openbeerdatabase/rest?rev=1425859309

Last update: 2019/08/31 14:40

