

Documentation API

ACQUART Quentin, DIMEGLIO Nicolas, ROMANO Sébastien
& SIXDENIER Alexandre

Table des matières :

I.	Services Connexion :	1
a.	POST /auth/verify/	1
b.	POST /auth/signin/	2
II.	Service de gestion des comptes	3
c.	POST /account/	3
d.	PUT /account/:email	4
e.	GET /role/	5
f.	GET /city/	5
g.	GET /city/:id	5
III.	Services Restaurant	6
h.	GET /restaurant/	6
i.	GET Restaurant/:id	6
j.	PUT Restaurant/:id	7
k.	DELETE /restaurant/:id	7
l.	POST /restaurant/	8
IV.	Service de commande	9
m.	GET /orders/	9
n.	GET /orders/:id	10
o.	GET /orders/res/:id_restaurant	11
p.	PUT /orders/:id	11
q.	GET /orders/	12
V.	Service de livraison	13
r.	GET /delivery/	13
s.	GET /delivery/:id	13
t.	GET /delivery/del/:id	14
u.	POST /delivery/	14

I. Services Connexion :a. **POST** /auth/verify/

Vérification du token

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Réponses :	
200	Token is Valid or Token is renew + Token Bearer XXX
404	Valid token : Data Not Found
500	Bad Token : Token is not valid

b. **POST** /auth/signin/

Se connecter à l'application

Corps de requête :

{ email : string, password : string }

Réponses :	
200	{ idUser: number, firstnameUser : string, lastnameUser : string, phone : string, nameUser : string, siretUser: number, addressUser : string, pointsUser : string, isSponsorUser : boolean, isSponsorisedUser : boolean, role_idRole : string, cpCityID : number }, Token : Bearer XXXX
404	Error missing parameters
404	Data not found

II. Service de gestion des comptes

c. **POST** /account/

Créer un nouveau compte utilisateur

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Corps de requête :

```
{
  firstnameUser : string,
  lastnameUser : string,
  email : string,
  phone : string,
  nameUser : string,
  siretUser: number,
  addressUser : string,
  mpUser : string,
  pointsUser : string,
  isSponsorUser : boolean,
  isSponsorisedUser : boolean,
  role_idRole : string,
  cpCityID : number
}
```

Réponses :	
201	User creation done
500	Internal error

d. **PUT** /account/:email

Modifier un compte utilisateur

Paramètres :

Token	Envoyer le token dans le header de la requête
Email	Email du compte que l'on veut modifier

Corps de requête :

```
{
  firstnameUser : string,
  lastnameUser : string,
  phone : string,
  nameUser : string,
  addressUser : string,
  mpUser : string,
  pointsUser : string,
  isSponsorUser : boolean,
  isSponsorisedUser : boolean,
  role_idRole : string,
  cpCityID : number
}
```

Réponses :

200	<pre>{ firstnameUser : string, lastnameUser : string, phone : string, nameUser : string, addressUser : string, mpUser : string, pointsUser : string, isSponsorUser : boolean, isSponsorisedUser : boolean, role_idRole : string, cpCityID : number }</pre> With update
400	Error missing parameters
404	Data not found

e. GET /role/

Récupérer la liste des rôles

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Réponses :

200	{ roles : string }
404	Data not found

f. GET /city/

Récupérer la liste des villes

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Réponses :

200	{ cpCityid: number, nameCity : string }
404	Data not found

g. GET /city/:id

Récupérer une villes

Paramètres :

CpCityid	Le numéro de la ville (code postal)
----------	-------------------------------------

Réponses :

200	{ cpCityid: number, nameCity : string }
404	Data not found

III. Services Restaurant

h. GET /restaurant/

Récupérer tous les restaurants

Pas de Paramètres.

Réponses	
200	<pre>"means_of_payment_accepted": [{ "Cards": "string" }], "schedules": "string", "estimated_average_delivery": "string", "_id": "string", "menus": { ... } "articles": {...}</pre>
404	Data not found

i. GET Restaurant/:id

Récupérer les informations sur un restaurant spécifique.

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Réponses :	
200	<pre>"means_of_payment_accepted": [{ "Cards": "string" }], "schedules": "string", "estimated_average_delivery": "string", "_id": "string", "menus": { ... } "articles": {...}</pre>
400	Error missing parameters
404	Data not found

j. **PUT** Restaurant/:id

Modifier les informations d'un restaurant

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Corps de requête :

```
"means_of_payment_accepted": [  
  {  
    "Cards": "string"  
  }  
],  
"schedules": "string",  
"estimated_average_delivery": "string",  
"_id": "string",  
"menus": { ... }  
"articles": {...}  
}
```

Réponses :	
200	OK
400	False + Error
404	Data not found

k. **DELETE** /restaurant/:id

Supprimer un restaurant

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Réponses :	
200	OK
400	False + Error
404	Data not found

I. **POST** /restaurant/

Rajouter des restaurants

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Corps de requête :

```
"means_of_payment_accepted": [  
  {  
    "Cards": "string"  
  }  
],  
"schedules": "string",  
"estimated_average_delivery": "string",  
"_id": "string",  
"menus": { ... }  
"articles": {...}  
}
```

Réponses :	
200	OK
400	False + Error
404	Data not found

IV. Service de commandem. **GET** /orders/

Récupérer toutes les commandes de tous les utilisateurs.

Paramètres :

Token	Envoyer le token dans le header de la requête
User_id	Id de l'utilisateur

Réponses :

200	{ order_Number: String, user_id: String, restaurant_id: String, orders_Details: { }, order_payment: { }, order_status: String, orderDate: String }
400	False + Error
404	Data not found

n. GET /orders/:id

Récupérer les commandes d'un utilisateur.

Paramètres :

Token	Envoyer le token dans le header de la requête
User_id	Id de l'utilisateur

Réponses :

200	{ order_Number: String, user_id: String, restaurant_id: String, orders_Details: { }, order_payment: { }, order_status: String, orderDate: String }
400	False + Error
404	Data not found

o. GET /orders/res/:id_restaurant

Récupérer toutes les commandes d'un restaurant

Paramètres :

Token	Envoyer le token dans le header de la requête
id_restaurant	Id du restaurant

Réponses :

200	{ order_Number: String, user_id: String, restaurant_id: String, orders_Details: { }, order_payment: { }, order_status: String, orderDate: String }
400	False + Error
404	Data not found

p. PUT /orders/:id

Récupérer une commande spécifique

Paramètres :

Token	Envoyer le token dans le header de la requête
order_id	Id de la commande

Corps de requête :

```
{  
  order_Number: String,  
  user_id: String,  
  restaurant_id: String,  
  orders_Details: {  
  },  
  order_payment: {  
  },  
  order_status: String,  
  orderDate: String  
}
```

Réponses :

200	{ order_Number: String, user_id: String, restaurant_id: String, orders_Details: { }, order_payment: { }, order_status: String, orderDate: String }
400	False + Error
404	Data not found

q. GET /orders/

Récupérer toutes les commandes

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Réponses :

200	{ order_Number: String, user_id: String, restaurant_id: String, orders_Details: { }, order_payment: { }, order_status: String, orderDate: String }
400	False + Error
404	Data not found

V. Service de livraisonr. GET /delivery/

Récupérer toutes les livraisons

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Réponses :

200	{ delivery_id : String, devlivery_adress: String, devlivery_time: String, order_Number: String, devlivery_id: String, }
400	False + Error
404	Data not found

s. GET /delivery/:id

Récupérer une livraison spécifique

Paramètres :

Token	Envoyer le token dans le header de la requête
delivery_id	L'identifiant de la commande

Réponses :

200	{ delivery_id : String, devlivery_adress: String, devlivery_time: String, order_Number: String, devlivery_id: String, }
400	False + Error
404	Data not found

t. GET /delivery/del/:id

Récupérer toutes les livraisons d'un certain restaurant

Paramètres :

Token	Envoyer le token dans le header de la requête
delivery_id	L'identifiant de la commande

Réponses :

200	{ delivery_id : String, devlivery_adress: String, devlivery_time: String, order_Number: String, devlivery_id: String, }
400	False + Error
404	Data not found

u. POST /delivery/

Créer une nouvelle commande

Paramètres :

Token	Envoyer le token dans le header de la requête
-------	-----------------------------------------------

Corps de requête :

```
{
  delivery_id : String,
  devlivery_adress: String,
  devlivery_time: String,
  order_Number: String,
  devlivery_id: String,
}
```

200	Delete Delivery
400	False + Error
404	Data not found