

API - url

Les agences

Url de base : api/agency

/create

POST

Fonctionnalités : *Créer une nouvelle agence*

Paramètres :

- Un objet agence

```
{
  "id": 0,
  "type": "string",
  "address": "string",
  "idMotherAgency": 0,
  "phoneNum": "string",
  "city": "string",
  "name": "string",
  "bankLink": "string",
  "bankName": "string",
  "rib": "string",
  "status": true}
```

Sortie:

- Renvoie l'agence créée
- Si erreur renvoie un code d'erreur



/edit

PUT

Fonctionnalités : *Modifier les paramètres d'une agence*

Paramètres :

- Un objet agence

```
{
  "id": 0,
  "type": "string",
  "address": "string",
  "idMotherAgency": 0,
  "phoneNum": "string",
  "city": "string",
  "name": "string",
  "bankLink": "string",
  "bankName": "string",
  "rib": "string",
  "status": true}
```

Sortie:

- Renvoie l'agence modifiée
- Si erreur renvoie un code d'erreur



/view

GET

Fonctionnalités : *Visualiser les données d'une agence*

Paramètres :

- id : l'identifiant de l'agence à visualiser

Sortie:

- Renvoie l'objet agence
- Si erreur renvoie un code d'erreur



/delete

DELTE

Fonctionnalités : *Supprimer une agence*

Paramètres :

- id : l'identifiant de l'agence à supprimer

Sortie:

- Renvoie les données l'agence supprimée
- Si erreur renvoie un code d'erreur



/vehicle

GET

Fonctionnalités : *Visualiser la liste des véhicules d'une agence*

Paramètres :

- id : l'identifiant de l'agence ou de la sous-agence

Sortie:

- Renvoie une liste de véhicule
- Si erreur renvoie un code d'erreur



/list

GET

Fonctionnalités : *Obtenir la liste des sous agences pour une agence*

Paramètres :

- id : l'identifiant de l'agence ou de la sous-agence

Sortie:

- Renvoie une liste d'agences
- Si erreur renvoie un code d'erreur



/transactions

GET

Fonctionnalités : *Obtenir la liste des transactions effectuées par une agence sur une période donnée*

Paramètres :

- id : l'identifiant de l'agence ou de la sous-agence
- start_date : date de début de la période
- end_date : date de la fin de la période

Sortie:

- Renvoie une liste de transactions
- Si erreur renvoie un code d'erreur



/rents

GET

Fonctionnalités : *Obtenir la liste des location effectuées par une agence sur une période donnée*

Paramètres :

- id : l'identifiant de l'agence ou de la sous-agence
- start_date : date de début de la période
- end_date : date de la fin de la période

Sortie:

- Renvoie une liste de locations
- Si erreur renvoie un code d'erreur



/getAllAgencies

GET

Fonctionnalités : *Obtenir la liste de toutes les agences inscrites dans la base de données*

Sortie:

- Renvoie la liste des agences
- Si erreur renvoie un code d'erreur



/activate

PUT

Fonctionnalités : *Réactiver une agence qui a été supprimée*

Paramètres :

- id : l'identifiant de l'agence ou de la sous-agence

Sortie:

- Renvoie l'agence réactivée
- Si erreur renvoie un code d'erreur



Les users

Url de base : api/user

/create

POST

Fonctionnalités : *Créer un nouvel utilisateur*

Paramètres :

- un objet user :

```
{ "id": 0,  
  "userName": "string",  
  "userFirstName": "string",  
  "role": "string",  
  "mailAddress": "string",  
  "login": "string",  
  "phoneNum": "string",  
  "idAgency": 0,  
  "token": "string",  
  "tokenExpiration": "2017-12-17T11:29:24.571Z",  
  "status": true }
```

Sortie:

- Renvoie l'objet user créé
- Si erreur renvoie un code d'erreur



/edit

PUT

Fonctionnalités : *Modifie les données d'un utilisateur*

Paramètres :

- un objet user :

```
{ "id": 0,  
  "userName": "string",  
  "userFirstName": "string",  
  "role": "string",  
  "mailAddress": "string",  
  "login": "string",  
  "phoneNum": "string",  
  "idAgency": 0,  
  "token": "string",  
  "tokenExpiration": "2017-12-17T11:29:24.571Z",  
  "status": true }
```

Sortie:

- Renvoie l'objet user modifié
- Si erreur renvoie un code d'erreur



/view

GET

Fonctionnalités : *Visualiser les données d'un utilisateur*

Paramètres :

- id : l'identifiant de l'utilisateur

Sortie:

- Renvoie l'objet user
- Si erreur renvoie un code d'erreur



/disable

DELTE

Fonctionnalités : *Supprimer un utilisateur*

Paramètres :

- id : l'identifiant de l'utilisateur à supprimer

Sortie:

- Renvoie l'objet user
- Si erreur renvoie un code d'erreur



/authenticate

GET

Fonctionnalités : *Permettre à une utilisateur de s'identifier*

Paramètres :

- login : le nom de connexion de l'utilisateur
- password : le mot de passe de l'utilisateur

Sortie:

- Renvoie l'objet user de l'utilisateur connectée
- Si erreur renvoie un code d'erreur



/logout

GET

Fonctionnalités : *Permettre à un utilisateur de se déconnecter*

Paramètres : aucun paramètre

Sortie:

- Si erreur renvoie un code d'erreur



/reativate

PUT

Fonctionnalités : *Réactiver un compte utilisateur supprimé*

Paramètres :

- id : l'identifiant de l'utilisateur à réactiver

Sortie:

- Renvoie les données de l'utilisateur
- Si erreur renvoie un code d'erreur



/editpassword

PUT

Fonctionnalités : *Modifier le mot de passe d'un utilisateur*

Paramètres :

- idUser : l'identifiant de l'utilisateur
- oldPass : ancien mot de passe
- newPass : nouveau mot de passe

Sortie:

- Renvoie les données de l'utilisateur
- Si erreur renvoie un code d'erreur



/getAll

GET

Fonctionnalités : *Obtenir tous les utilisateur enregistré dans la bdd*

Paramètres : aucun paramètre

Sortie:

- Renvoie la liste des utilisateurs
- Si erreur renvoie un code d'erreur



/transactions

GET

Fonctionnalités : *Obtenir les transactions d'un utilisateur*

Paramètres :

- id : identifiant de l'utilisateur

Sortie:

- Renvoie la liste des transactions de l'utilisateur
- Si erreur renvoie un code d'erreur



Les véhicules

Url de base : api/vehicle

/create

POST


Fonctionnalités : Créer un véhicule

Paramètres :

- un objet véhicule

```
{ "id": 0,  
  "type": 0,  
  "brand": "string",  
  "price": 0,  
  "insurance": "string",  
  "idAgency": 0,  
  "status": true,  
  "characteristicList": [] }
```

Sortie:

- Renvoie le véhicule créé
 - Si erreur renvoie un code d'erreur
- 

/addCharac


POST

Fonctionnalités : Ajouter une caractéristique d'un véhicule

Paramètres :

- idVehicle : identifiant du véhicule
- idCharacteristic : identifiant de la caractéristique à ajouter
- un objet caractéristique : { "valueCharacteristic": "string", }

Sortie:

- Renvoie la caractéristique ajoutée
 - Si erreur renvoie un code d'erreur
- 

/list


GET

Fonctionnalités : Obtenir la liste des caractéristiques possibles

Paramètres :

- idType(optionel) : identifiant du type de véhicule

Sortie:

- Renvoie la liste de toutes les caractéristiques de la bdd ou des caractéristiques d'un type de véhicule si idType non nul
 - Si erreur renvoie un code d'erreur
- 

/view

GET

Fonctionnalités : *Visualiser un véhicule*

Paramètres :

- id : identifiant du véhicule

Sortie:

- Renvoie le véhicule
- Si erreur renvoie un code d'erreur



/edit

PUT

Fonctionnalités : *Modifier un véhicule*

Paramètres :

- un objet véhicule
{
 "id": 0,
 "type": 0,
 "brand": "string",
 "price": 0,
 "insurance": "string",
 "idAgency": 0,
 "status": true,
 "characteristicList": []
}

Sortie:

- Renvoie le véhicule modifié
- Si erreur renvoie un code d'erreur



/editCharac

PUT

Fonctionnalités : *Modifier une caractéristique d'un véhicule*

Paramètres :

- idVehicle : identifiant du véhicule à modifier
- idCharacteristic : identifiant de la caractéristique à modifier
- un objet caractéristique : {"valueCharacteristic": "string",}

Sortie:

- Renvoie la caractéristique modifiée
- Si erreur renvoie un code d'erreur



/delete

DELTE

Fonctionnalités : *Effacer un véhicule*

Paramètres :

- id : identifiant du véhicule à supprimer

Sortie:

- Renvoie le véhicule supprimé
- Si erreur renvoie un code d'erreur



/search

GET

Fonctionnalités : *Obtenir une liste de véhicule disponible sur une période de temps*

Paramètres :

- startDate : date début periode
- endDate : date de fin de periode

Sortie:

- Renvoie une liste de véhicule
- Si erreur renvoie un code d'erreur



/dispo

GET

Fonctionnalités : *Obtenir une liste des disponibilités d'un véhicule*

Paramètres :

- id : identifiant du véhicule

Sortie:

- Renvoie une liste de disponibilité
- Si erreur renvoie un code d'erreur



/type

GET

Fonctionnalités : *Obtenir les types possibles de véhicule dans la bdd*

Paramètres :

Sortie:

- Renvoie une liste de type de véhicule
- Si erreur renvoie un code d'erreur



/rent

GET

Fonctionnalités : *Obtenir une liste des location pour un véhicule*

Paramètres :

- id : id du véhicule

Sortie:

- Renvoie une liste de location
- Si erreur renvoie un code d'erreur



Transactions

Url de base : api/transaction

/rent

GET

Fonctionnalités : *Obtenir une transaction*

Paramètres :

- id : id d'une transaction

Sortie:

- Renvoie une transaction
- Si erreur renvoie un code d'erreur

