



API REST CLEVERMULTIMEDIAS SPECIFICATIONS TECHNIQUES

Table des matières

1. Introduction.....	3
2. Cadre d'utilisation des Web Services	3
2.1. Architecture	3
2.2. Parties prenantes.....	3
2.3. Echange de données	4
3. FONCTIONS disponibles au niveau des Web Services Rest	5
3.1. Obtenir les informations du profil	5
3.2. Créer un utilisateur (à venir).....	7
3.3. Modifier un utilisateur.....	9
3.4. Rechercher un utilisateur.....	10
3.5. Créer un contact.....	10
3.6. Modifier un contact	12
3.7. Rechercher un contact	14
3.8. Supprimer un contact.....	15
3.9. Lister tous les contacts	16
3.10. Lister les contacts d'un utilisateur (<i>Manager uniquement</i>)	18
3.11. Effectuer une campagne de messages (Push)	20
3.12. Effectuer une campagne de mails (Push)	22
3.13. Effectuer une campagne de messages avec critères (Push)	25
3.14. Effectuer une campagne de mails avec critères (Push)	27
3.15. Effectuer une campagne de messages à tous les contacts (Push)	30
3.16. Effectuer une campagne de mails à tous les contacts (Push)	32
3.17. Supprimer une campagne de messages différée (Push)	35
3.18. Supprimer une campagne de mails différée (Push)	36
3.19. Obtenir les informations d'une campagne de messages (Push)	37
3.20. Obtenir les informations d'une campagne mails (Push).....	39
3.21. Obtenir les acquittements d'une campagne de messages (Push)	40
3.22. Obtenir les acquittements d'une campagne mails (Push).....	42
3.23. Obtenir les MO's d'une campagne de messages (Push)	44
3.24. Lister les champs additionnels (à venir).....	45
3.25. Lister les messages prédéfinis (à venir).....	46

1. INTRODUCTION

La société Clever Technologies dispose d'une solution en ligne permettant de faire des envois de sur différents médias en masse. Elle est disponible à l'adresse <https://clevermultimedias.clever-is.fr>.

Des Web services de type ReST sont implémentés pour interfacer certaines fonctionnalités disponibles au sein de l'application. Ils aux clients de s'intégrer depuis la plus grande variété de systèmes. Notre API de webservices permet de faire des appels authentifiés à partir d'applications existantes ou d'outils tiers.

L'objectif de ce document est de présenter :

- ❖ le cadre d'utilisation de ces Web Services.
- ❖ les fonctionnalités publiées.

2. CADRE D'UTILISATION DES WEB SERVICES

2.1. Architecture

Les clients effectuent des requêtes HTTP standard sur un canal SSL.

2.2. Parties prenantes

On trouve du côté serveur :

Les Webservices publiés dans les serveurs de la société CLEVER Technologies.

On trouve du côté client :

Les programmes développés par des tiers, et dont la finalité est de consommer les Webservices publiés par CLEVER Technologies. L'envoi de requête doit être effectué en envoyant des commandes HTTP standard.

- Ressources

Les clients font des requêtes à l'encontre des ressources des WS. Le format de l'URL est :

`http://endpoint/api/resource [? Query_parameters]`

Exemple : <http://webservicesmultimedias.clever-is.fr/api/pushs>

- Opérations

- GET : récupère une liste de ressources ou les détails d'une ressource spécifique.
- POST : crée une nouvelle instance de ressource avec les paramètres spécifiés.
- PUT : met à jour une ressource existante avec les paramètres spécifiés.
- DELETE : supprime une ressource avec les paramètres spécifiés.

2.3. Echange de données

- Content-type

Les données échangées sont au format JSON. Le client doit fournir dans le header de sa requête l'entête suivant : **Content-type : application/json**

- Authentification

L'authentification est de type basic HTTP Header. L'identifiant et le mot de passe sont encodés en utilisant l'algorithme base64 de la façon suivante : **base_64(identifiant:mot_de_passe)** . Puis au niveau du Header, l'entête suivant : **Authorization : base_64(identifiant:mot_de_passe)** doit être fourni.

- Codes de réponse

Les Webservices répondent avec des codes de réponse HTTP standard appropriés au résultat de la demande. Bien que la signification exacte du code varie en fonction de la demande, les règles sont les suivantes :

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée et des détails sur la réponse peuvent être trouvés dans le corps de la réponse.
201	Created	La requête est traitée avec succès et un objet a été créé sur le serveur
202	Accepted	L'opération demandée a été acceptée. Vous pouvez interroger pour vérifier sur l'avancement de la demande.
204	No Content	L'opération demandée a réussi et il n'y a pas de corps de réponse.
400	Bad Request	Les paramètres de la requête sont incorrects.
402	Payment Required	Le crédit du compte est insuffisant. Paiement requis pour accéder à la ressource.
401	Unauthorized	L'authentification est incorrecte.
403	Forbidden	La demande n'a pas été autorisée.
404	Not Found	La ressource demandée n'existe pas.
405	Method Not Allowed	L'opération demandée n'est pas permise.
409	Conflict	La requête entre en conflit avec une autre requête. Des informations sur les raisons de ce conflit doivent être indiquées en retour.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.
501	Not Implemented	Le serveur HTTP ne supporte pas la fonctionnalité nécessaire pour traiter la requête.
503	Service Unavailable	Le serveur HTTP est actuellement incapable de traiter la requête en raison d'une surcharge temporaire ou d'une opération de maintenance.

- Entités de réponse

Toutes les méthodes GET répondent avec le JSON de la ou des ressource(s) demandée(s).

Les méthodes POST peuvent répondre avec un code de réponse 201 Created ou 202 Accepted Selon que la création a été achevée immédiatement ou si un processus asynchrone est lancé pour l'opération. Si la ressource a été créée immédiatement, les Webservices fourniront une réponse JSON avec un contenu qui inclut l'ID unique de la nouvelle ressource. Sinon une réponse sans contenu.

Les méthodes PUT et DELETE répondent en général avec 204 No Content. Quand il s'agit d'opération on à longue durée de vie, dans ces cas, le PUT répondra avec un 202 Accepted.

3. FONCTIONS DISPONIBLES AU NIVEAU DES WEB SERVICES REST

3.1. Obtenir les informations du profil

- Requête :

- **URL:** <http://webservicesmultimedias.clever-is.fr/api/account/me>
- **Commande :** GET
- **Paramètres :** Vide

- Retour :

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

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

- Request:** Method: GET, URL: <http://webservicesmultimedias.clever-is.fr/api/account/me>
- Headers:** Authorization: Basic Y2VnZW50ZDZldiZGltMjAxOQ==, Content-Type: application/json
- Body:** Request Body (empty)

- Exemple :

- o **Corps de requête :** Objet datas JSON
L'opération demandée de corps de requête.

- **Corps de la réponse (Body) en cas de code 200 :**

```
{
  "account": {
    "name": "Clever",
    "lastname": "Tec",
    "email": "support@clever.fr",
    "telephone": "+33160536053",
    "credit": 999733,
    "date_expiration": "2033-02-10 09:49:21",
    "company": {
      "company_name": "Clever Technologies - IS",
      "address": "52 rue d'emeraïnvill",
      "country": "France",
      "city": "Croissy",
      "zip_code": 77183
    }
  }
}
```

account : Objet retourné.

name : nom du contact du compte.

lastname : prénom du contact du compte.

email : email du contact du compte.

telephone : numéro de téléphone du contact du compte.

credit : nombre d'unités restantes du compte.

date_expiration : date d'expiration du compte.

company : informations de société du compte.

Method: GET URL: imediass.clever-is.fr/api/accounts/me

Headers: Authorization: Basic Y2x0dmVydGV... Content-Type: application/json

Body: Request Body

[+] Response

Response Headers Response Body (Raw) Response Body (Highlight) Response Body (Preview)

```

1. {
2.   "account":
3.   {
4.     "name": "Clever",
5.     "lastname": "Tec",
6.     "email": "support@clever.fr",
7.     "tocontrol": 0,
8.     "telephone": "+33160536053",
9.     "credit": 999733,
10.    "date_expiration": "2033-02-10 09:49:21",
11.  }
12.  "company":
13.  {
14.    "company_name": "Clever Technologies - IS",
15.    "address": "52 rue d'emeraïnvill",
16.    "country": "France",
17.    "city": "Croissy",
18.    "zip_code": 77183
19.  }
20. }
21. }
```

3.2. Créer un utilisateur (à venir)

- Requête :

- URL: <http://webservicesmultimedias.clever-is.fr/api/users>
- Commande : POST
- Paramètres :

Nom paramètre	Obligatoire	Commentaire
<i>name</i>	<i>oui</i>	
<i>lastname</i>	<i>oui</i>	
<i>telephone</i>	<i>oui</i>	<i>format international à respecter +336xxxxxxxx</i>
<i>email</i>	<i>oui</i>	
<i>username</i>	<i>oui</i>	
<i>password</i>	<i>oui</i>	<i>Le mot de passe doit contenir au moins 6 caractères dont une lettre capitale et un chiffre</i>
credit	non	Par défaut 0, l'utilisateur ne peut effectuer de campagne sans crédit
company_name	non	
<i>address</i>	<i>oui</i>	
<i>zipCode</i>	<i>oui</i>	
<i>city</i>	<i>oui</i>	
<i>country</i>	<i>oui</i>	<i>Se référer à la liste pays fournie dans l'API</i>

- Retour :

Code	Message	Signification
201	Created	L'utilisateur a été créé avec succès.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
409	Conflit	L'utilisateur existe, username déjà pris.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON

```
{
  "datas":
  {
    "username": "useryb",
    "password": "A12345678e",
    "account":
    {
      "name": "clever",
      "lastname": "tec",
      "email": "support@clever.fr",
      "telephone": "+33160536053",
      "company":
      {
        "address": "Croissy",
        "company_name": "IS"
      }
    }
  }
}
```

- **Corps de la réponse (Body) en cas de code 201 :**

```
{
  "user":
  {
    "username": "useryb",
    "password": "A12345678e",
    "account":
    {
      "name": "clever",
      "lastname": "tec",
      "email": "support@clever.fr",
      "telephone": "+33160536053",
      "company":
      {
        "address": "Croissy",
        "company_name": "IS"
      }
    }
  }
}
```


contact : Objet retourné.

id : identifiant unique du contact créé.

name : nom du contact.

lastName : nom du contact.

gsmPrimary : numéro de téléphone mobile principal du contact.

email : email du contact.

3.3. Modifier un utilisateur

- Requête :

- **URL**: http://webservicesmultimedias.clever-is.fr/api/users/{id_user}

- **Commande** : PUT

- **Paramètres** :

Nom paramètre	Obligatoire	Commentaire
<i>id_user</i>	<i>oui</i>	<i>doit être un entier (Paramètre d'URL)</i>
<i>name</i>	<i>oui</i>	
<i>lastname</i>	<i>oui</i>	
<i>telephone</i>	<i>oui</i>	<i>format international à respecter +336xxxxxxxx</i>
<i>email</i>	<i>oui</i>	
<i>username</i>	<i>oui</i>	
<i>password</i>	<i>oui</i>	<i>Le mot de passe doit contenir au moins 6 caractères dont une lettre capitale et un chiffre</i>
credit	non	Par défaut 0, l'utilisateur ne peut effectuer de campagne sans crédit
company_name	non	
<i>address</i>	<i>oui</i>	
<i>zipCode</i>	<i>oui</i>	
<i>city</i>	<i>oui</i>	
<i>country</i>	<i>oui</i>	<i>Se référer à la liste pays fournie dans l'API</i>

- Retour :

Code	Message	Signification
204	No Content	La requête s'est correctement exécutée. Pas de corps de réponse.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
409	Conflit	L'utilisateur existe, username déjà pris.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

3.4. Rechercher un utilisateur

- Requête :

- **URL:** `http://webservicesmultimedias.clever-is.fr/api/users/{id_user}`
- **Commande :** GET
- **Paramètres :** Vide (Paramètre d'URL id_user)

- Retour :

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
404	Not Found	La ressource demandée n'existe pas.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

3.5. Créer un contact

- Requête :

- **URL:** `http://webservicesmultimedias.clever-is.fr/api/contacts`
- **Commande :** POST
- **Paramètres :**

Nom paramètre	Obligatoire	Commentaire
<i>name</i>	<i>oui</i>	<i>pas de caractères spéciaux</i>
<i>lastname</i>	<i>oui</i>	<i>pas de caractères spéciaux</i>
<i>gsm_primary</i>	<i>oui</i>	<i>format international à respecter +336xxxxxxxx</i>
<i>gsm_secondary</i>	non	
<i>landline_primary</i>	non	
<i>landline_secondary</i>	non	
<i>fax_primary</i>	non	
<i>fax_secondary</i>	non	
<i>email</i>	non	
<i>zipCode</i>	non	
<i>city</i>	non	
<i>street</i>	non	
<i>streetNumber</i>	non	
<i>duplicates</i>	non	Par défaut 0, duplicates=1 pour autoriser les doublons

- Retour :

Code	Message	Signification
201	Created	Le contact a été créé avec succès.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
409	Conflit	Le contact existe, le paramètre duplicates doit être activé.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON

```
{
  "datas" :
  {
    "name": "contact",
    "lastName": "contact lastname",
    "gsmPrimary": "+33600000001",
    "email" : "contact@email.com"
  }
}
```

- **Corps de la réponse (Body) en cas de code 201 :**

```
{
  "contact":
  {
    "id": "155576",
    "name": "contact",
    "lastName": "contact lastname",
    "gsmPrimary": "+3361000000",
    "email" : "contact@email.com"
  }
}
```

contact : Objet retourné.

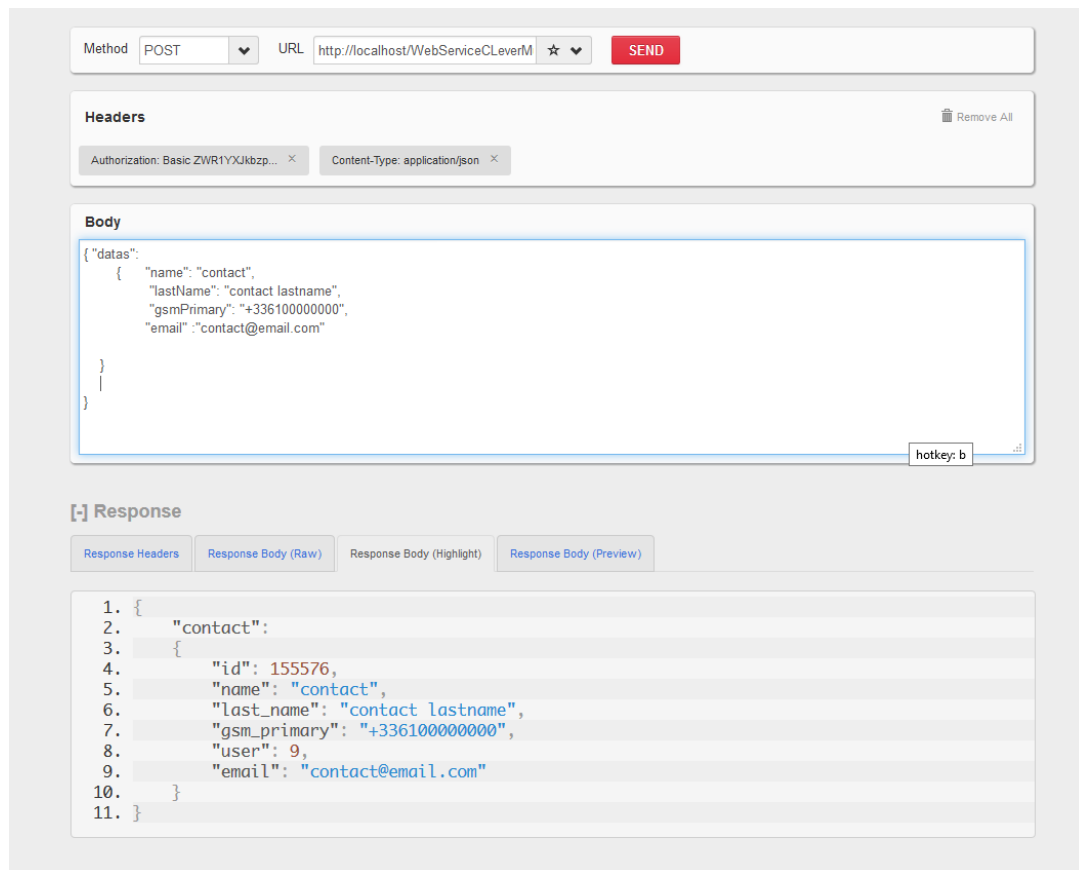
id : identifiant unique du contact créé.

name : nom du contact.

lastName : nom du contact.

gsmPrimary : numéro de téléphone mobile principal du contact.

email : email du contact.



3.6. Modifier un contact

- Requête :

- **URL:** http://webservicesmultimedias.clever-is.fr/api/contacts/{id_contact}
- **Commande :** PUT
- **Paramètres :**

Nom paramètre	Obligatoire	Commentaire
<i>id_contact</i>	<i>oui</i>	<i>doit être un entier (Paramètre d'URL)</i>
<i>name</i>	<i>oui</i>	<i>pas de caractères spéciaux</i>
<i>lastname</i>	<i>oui</i>	<i>pas de caractères spéciaux</i>
<i>gsm_primary</i>	<i>oui</i>	<i>format international +336xxxxxxxx</i>
<i>gsm_secondary</i>	non	
<i>landline_primary</i>	non	
<i>landline_secondary</i>	non	
<i>fax_primary</i>	non	
<i>fax_secondary</i>	non	
<i>email</i>	non	
<i>zipCode</i>	non	
<i>city</i>	non	
<i>street</i>	non	
<i>streetNumber</i>	non	
<i>duplicates</i>	non	Par défaut 0, duplicates=1 pour autoriser les doublons

- Retour :

Code	Message	Signification
204	No Content	La requête s'est correctement exécutée. Pas de corps de réponse.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
409	Conflit	Le contact existe, le paramètre duplicates doit être activé.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON

```
{
  "datas" :
  {
    "name": "contact",
    "lastName": "contact lastname",
    "gsmPrimary": "+33600000002",
    "email" : "contact@email.com"
  }
}
```

- **Corps de la réponse (Body) en cas de code 204 :**

L'opération demandée a réussi et il n'y a pas de corps de réponse.

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

- Method:** PUT
- URL:** `https://didas.clever-is.fr/api/contacts/214917`
- Headers:**
 - Authorization: Basic Y2xldmVydGV...
 - Content-Type: application/json
- Body:**

```
{
  "datas" :
  {
    "name": "contact",
    "lastName": "contact lastname",
    "gsmPrimary": "+33600000002",
    "email" : "contact@email.com"
  }
}
```
- [-] Response:**
 - Response Headers
 - Response Body (Raw)
 - Response Body (Highlight)
 - Response Body (Preview)
- Response Details:**
 - Status Code : 204 No Content
 - Cache-Control : no-cache
 - Connection : close
 - Content-Type : text/html; charset=UTF-8
 - Date : Wed, 20 Sep 2017 14:45:13 GMT
 - Server : Apache/2.4.23 (Win64) OpenSSL/1.0.2j PHP/5.6.25
 - X-Powered-By : PHP/5.6.25

3.7. Rechercher un contact

- **URL**: `http://webservicesmultimedias.clever-is.fr/api/contacts/{id_contact}`
- **Commande** : GET
- **Paramètres** : Vide (Paramètre d'URL `id_contact`)

- **Retour** :

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
404	Not Found	La ressource demandée n'existe pas.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple** :

- **Corps de requête** : Objet datas JSON
L'opération demandée de corps de requête.
- **Corps de la réponse (Body) en cas de code 200** :

```
{
  "contact":
    {
      "id": "214917",
      "name": "contact",
      "lastName": "contact lastname",
      "gsmPrimary": "+3361000002",
      "email": "contact@email.com"
    }
}
```

contact : Objet retourné.

id : identifiant unique du contact créé.

name : nom du contact.

lastName : nom du contact.

gsmPrimary : numéro de téléphone mobile principal du contact.

email : email du contact.

Method
URL

Headers

Authorization: Basic Y2xldmVydGV...
Content-Type: application/json

Body

[+] Response

```

1. {
2.   "contact":
3.   {
4.     "id": 214917,
5.     "name": "contact",
6.     "last_name": "contact lastname",
7.     "gsm_primary": "600000002",
8.     "email": "contact@email.com"
9.   }
10. }

```

3.8. Supprimer un contact

- **URL**: `http://webservicesmultimedias.clever-is.fr/api/contacts/{id_contact}`
- **Commande** : DELETE
- **Paramètres** : Vide (Paramètre d'URL id_contact)
- **Retour** :

Code	Message	Signification
204	No Content	La requête s'est correctement exécutée. Pas de corps de réponse.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
404	Not Found	La ressource demandée n'existe pas.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple** :

- **Corps de requête** : Objet datas JSON
L'opération demandée de corps de requête.
- **Corps de la réponse (Body) en cas de code 204** :
L'opération demandée a réussi et il n'y a pas de corps de réponse.

Method: DELETE URL: sdias.clever-is.fr/api/contacts/214917 SEND

Headers Remove All

Authorization: Basic Y2xidmVydGVV... Content-Type: application/json

Body

Request Body

[+] Response

Response Headers Response Body (Raw) Response Body (Highlight) Response Body (Preview)

```

1. Status Code      : 204 No Content
2. Cache-Control    : no-cache
3. Connection       : close
4. Content-Type     : text/html; charset=UTF-8
5. Date             : Wed, 20 Sep 2017 15:27:41 GMT
6. Server           : Apache/2.4.23 (Win64) OpenSSL/1.0.2j PHP/5.6.25
7. X-Powered-By     : PHP/5.6.25
  
```

3.9. Lister tous les contacts

- **URL:** http://webservicesmultimedias.clever-is.fr/api/contactlists
- **Commande :** GET
- **Paramètres :** Vide
- **Retour :**

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple :**
 - o **Corps de requête :** Objet datas JSON
L'opération demandée de corps de requête.

- **Corps de la réponse (Body) en cas de code 200 :**

```
{
  "contact": [
    {
      "id": 214905,
      "name": "Clever",
      "last_name": "ALGnew3",
      "gsm_primary": "771100899",
      "gsm_secondary": "",
      "landline_primary": "",
      "landline_secondary": "",
      "email": "cleverimap@gmail.com",
      "fax_primary": "",
      "fax_secondary": "",
      "zip_code": "",
      "city": "Torcy",
      "street": "rue de paris",
      "street_number": "4",
      "longitude": "2.6566156",
      "latitude": "48.8511663"
    },
    {
      "id": 214906,
      "name": "contact",
      "last_name": "contact lastname",
      "gsm_primary": "600000001",
      "email": "contact@email.com"
    }
  ]
}
```

contact : Objet retourné.

id : identifiant unique du contact créé.

name : nom du contact.

lastName : nom du contact.

gsmPrimary : numéro de téléphone mobile principal du contact.

email : email du contact.

gsm_secondary : numéro de téléphone mobile secondaire du contact.

landline_primary : numéro de téléphone fixe principal du contact.

landline_secondary : numéro de téléphone fixe secondaire du contact.

fax_primary : numéro de téléphone fax principal du contact.

fax_secondary : numéro de téléphone fax secondaire du contact.

zip_code : code postal adresse du contact.

city : ville adresse du contact.

street : rue adresse du contact.

street_number : numéro rue adresse du contact.

longitude : longitude adresse du contact.

latitude : latitude adresse du contact.

The screenshot shows an API client interface. At the top, the Method is set to GET and the URL is 'http://multimedias.clever-is.fr/api/contactlists'. There is a 'SEND' button. Below this, the Headers section shows 'Authorization: Basic Y2xidmVydGV...' and 'Content-Type: application/json'. The Body section is empty. The Response section shows a JSON array with one contact object. The contact has the following fields: id (214905), name (Clever), last_name (ALGnew3), gsm_primary (771100899), gsm_secondary (empty), landline_primary (empty), landline_secondary (empty), email (cleverimap@gmail.com), fax_primary (empty), fax_secondary (empty), zip_code (empty), city (Torcy), street (rue de paris), street_number (4), longitude (2.6566156), and latitude (48.8511663).

```

1. {
2.   "contact":
3.   [
4.     {
5.       "id": 214905,
6.       "name": "Clever",
7.       "last_name": "ALGnew3",
8.       "gsm_primary": "771100899",
9.       "gsm_secondary": "",
10.      "landline_primary": "",
11.      "landline_secondary": "",
12.      "email": "cleverimap@gmail.com",
13.      "fax_primary": "",
14.      "fax_secondary": "",
15.      "zip_code": "",
16.      "city": "Torcy",
17.      "street": "rue de paris",
18.      "street_number": "4",
19.      "longitude": "2.6566156",
20.      "latitude": "48.8511663"
21.    },
22.    {
23.      "id": 214906

```

3.10. Lister les contacts d'un utilisateur (*Manager uniquement*)

- **URL:** `http://webservicesmultimedias.clever-is.fr/api/contactlists/{login}`
- **Commande :** GET
- **Paramètres :** Vide (Paramètre d'URL login)
- **Retour :**

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON
L'opération demandée de corps de requête.
- **Corps de la réponse (Body) en cas de code 200 :**

```
{
  "contact": [
    {
      "id": 214905,
      "name": "Clever",
      "last_name": "ALGnew3",
      "gsm_primary": "771100899",
      "gsm_secondary": "",
      "landline_primary": "",
      "landline_secondary": "",
      "email": "cleverimap@gmail.com",
      "fax_primary": "",
      "fax_secondary": "",
      "zip_code": "",
      "city": "Torcy",
      "street": "rue de paris",
      "street_number": "4",
      "longitude": "2.6566156",
      "latitude": "48.8511663"
    },
    {
      "id": 214906,
      "name": "contact",
      "last_name": "contact lastname",
      "gsm_primary": "600000001",
      "email": "contact@email.com"
    }
  ]
}
```

contact : Objet retourné.

id : identifiant unique du contact créé.

name : nom du contact.

lastName : nom du contact.

gsmPrimary : numéro de téléphone mobile principal du contact.

email : email du contact.

gsm_secondary : numéro de téléphone mobile secondaire du contact.

landline_primary : numéro de téléphone fixe principal du contact.

landline_secondary : numéro de téléphone fixe secondaire du contact.

fax_primary : numéro de téléphone fax principal du contact.

fax_secondary : numéro de téléphone fax secondaire du contact.

zip_code : code postal adresse du contact.

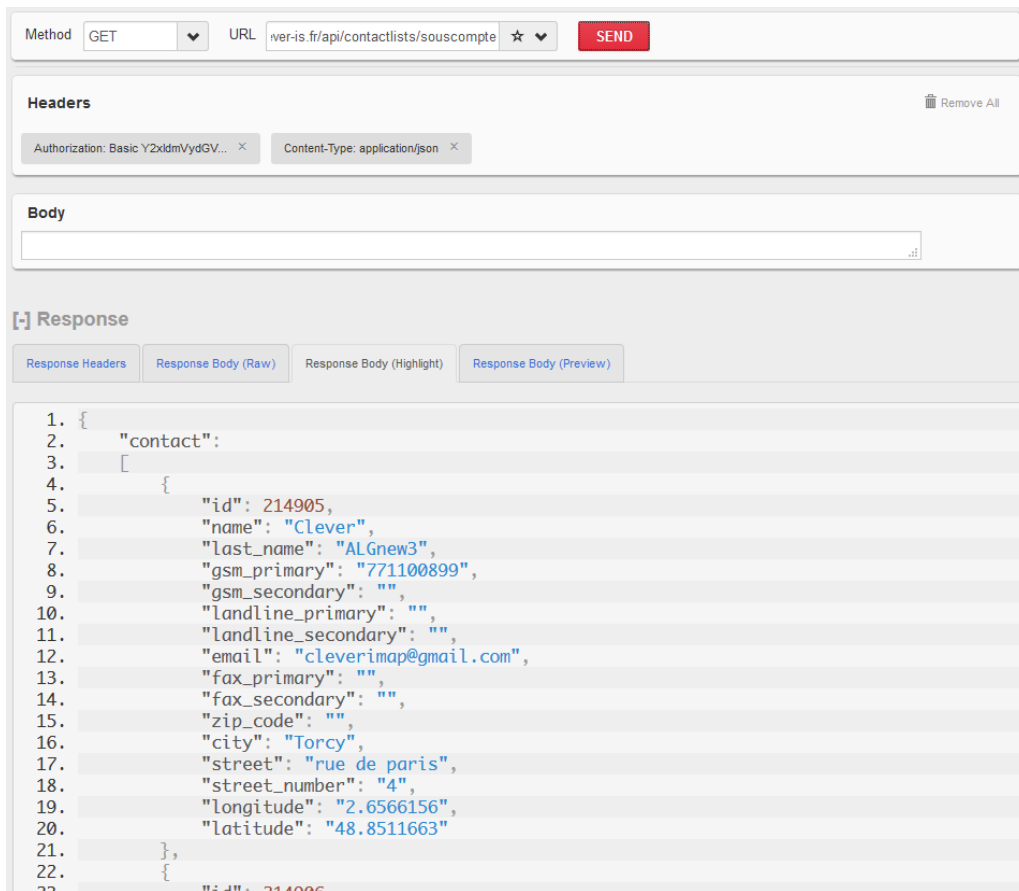
city : ville adresse du contact.

street : rue adresse du contact.

street_number : numéro rue adresse du contact.

longitude : longitude adresse du contact.

latitude : latitude adresse du contact.



3.11.Effectuer une campagne de messages (Push)

- **URL:** <http://webservicesmultimedias.clever-is.fr/api/pushs>
- **Commande :** POST
- **Paramètres :**

Nom paramètre	Obligatoire	Commentaire
number_list	<i>oui</i>	Tableau de numéros format international +336xxxxxxx
text	<i>oui</i>	Message à envoyer, longueur maximale 1000 caractères
encodage	<i>oui</i>	3 pour le Latin et 4 pour l'UCS2
short_code	non	11 caractères maximum
hd_send	non	Format Y-m-d H:i:s
filename	non	Extension : wav ou mp3 Caractères interdits (<, >, :, ", /, \, , ?, *)
type_push	non	Par défaut : sms, sinon : tts ou voice
options	non	F0 pour les messages Flash F1 pour le mode normal
duplicates	non	Par défaut 0, duplicates=1 pour autoriser les doublons

- Retour :

Code	Message	Signification
201	Created	La campagne (Push) est créée avec succès.
400	Bad Request	Les paramètres de la requête sont incorrects.
402	Payment Required	Le crédit du compte est insuffisant. Paiement requis.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON

```
{
  "datas": {
    {
      "text": "send push test",
      "number_list": "+33600000001;+33600000002",
      "type_push": "sms",
      "encodage": 3,
      "options": "F1",
      "duplicates": 0
    }
  }
}
```

- **Corps de la réponse (Body) en cas de code 201 :**

```
{
  "push": {
    "reference": "Push20171012124247",
    "nb_destination": 2,
    "number_list": "+33600000001;+33600000002",
    "encodage": 3,
    "text": "test envoi du message",
    "hd_send": "2017-10-12 12:42:47",
    "filename": "noFile",
    "type_push": "sms",
    "unit": 14,
    "nb_sms": 1,
    "duplicates": 0
  }
}
```

push : objet retourné.

reference : référence (Id) de la campagne (push) de messages.

nb_destination : nombre de destinataires dans la campagne.

number_list : liste des destinataires de la campagne.

text : message envoyé dans la campagne.

encodage : encodage du texte de la campagne.

hd_send : date d'envoi de la campagne.

filename : nom du fichier envoyé dans la campagne (cas de fichier wave).

type_push : type de la campagne.

unit : coût de la campagne en unités.

nb_sms : nombre de SMS pour le message de la campagne.

duplicates : coût de la campagne en unités.

Method: POST URL: /api/pushs

Headers: Authorization: Basic Y2xldmVydGV... Content-Type: application/json

Body:

```
{
  "datas": {
    "text": "test envoi du message",
    "number_list": "+33600000001;+33600000002",
    "type_push": "sms",
    "encodage": 3,
    "options": "F1",
    "duplicates": 0
  }
}
```

[-] Response

Response Headers Response Body (Raw) Response Body (Highlight) Response Body (Preview)

```
1. {
2.   "push":
3.     {
4.       "reference": "Push20171012124247",
5.       "nb_destination": 2,
6.       "number_list": "+33600000001;+33600000002",
7.       "encodage": 3,
8.       "text": "test envoi du message",
9.       "hd_send": "2017-10-12 12:42:47 ",
10.      "filename": "noFile",
11.      "type_push": "sms",
12.      "unit": 14,
13.      "nb_sms": 1,
14.      "duplicates": 0
15.    }
16.  }
17. }
```

3.12.Effectuer une campagne de mails (Push)

- URL: <http://webservicesmultimedias.clever-is.fr/api/pushsmails>
- Commande : POST
- Paramètres :

Nom paramètre	Obligatoire	Commentaire
<i>email_list</i>	<i>oui</i>	Tableau de mails
<i>text</i>	<i>oui</i>	
<i>email_subject</i>	<i>oui</i>	
type_mail	non	Par défaut : brut sinon html
hd_send	non	Format Y-m-d H:i:s
filename	non	Extension : pdf
type_push	non	mail
open_tag	non	Par défaut 0 sinon 1
duplicates	non	Par défaut 0, duplicates=1 pour autoriser les doublons

- Retour :

Code	Message	Signification
201	Created	La campagne (Push) est créée avec succès.
400	Bad Request	Les paramètres de la requête sont incorrects.
402	Payment Required	Le crédit du compte est insuffisant. Paiement requis.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON

```
{
  "datas":
  {
    "text": "Send Push Mail test",
    "email_list": "cleverimap@gmail.com;cleverimap@gmail.com",
    "email_subject": "push mail",
    "open_tag": 0,
    "type_mail": "html",
    "duplicates": 1
  }
}
```

- **Corps de la réponse (Body) en cas de code 201 :**

```
{
  "push": {
    "reference": "Push20171012153053",
    "email_list": "cleverimap@gmail.com;cleverimap@gmail.com",
    "text": "Send Push Mail",
    "email_subject": "push mail",
    "hd_send": "2017-10-12 15:30:53",
    "filename": "noFile",
    "unit": 4,
    "nb_mail": 2,
    "open_tag": 0,
    "type_mail": "html",
    "duplicates": 1
  }
}
```

push : objet retourné.

reference : référence (Id) de la campagne (push) de messages.

email_list : liste des destinataires de la campagne.

text : message envoyé dans la campagne.

email_subject : sujet du mail de la campagne.

hd_send : date d'envoi de la campagne.

filename : nom du fichier envoyé dans la campagne (cas de pièce jointe).

type_mail : type du mail de la campagne.

unit : coût de la campagne en unités.

nb_mail : nombre de destinataires dans la campagne.

open_tag : tag pour tracker l'ouverture des mails de la campagne.

duplicates : coût de la campagne en unités.

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

- Method:** POST
- URL:** /ltimedias.clever-is.fr/api/pushmails
- Headers:**
 - Authorization: Basic Y2xldmVydGV...
 - Content-Type: application/json
- Body:**

```
{
  "datas": {
    "text": "Send Push Mail",
    "email_list": "cleverimap@gmail.com;cleverimap@gmail.com",
    "email_subject": "push mail",
    "open_tag": 0,
    "duplicates": 1,
    "type_mail": "html"
  }
}
```
- Response:**
 - Response Headers
 - Response Body (Raw)
 - Response Body (Highlight) - Selected
 - Response Body (Preview)

The response body (highlighted) is a JSON object:

```
1. {
2.   "push":
3.     {
4.       "reference": "Push20171012153053",
5.       "email_list": "cleverimap@gmail.com;cleverimap@gmail.com",
6.       "text": "Send Push Mail",
7.       "email_subject": "push mail",
8.       "hd_send": "2017-10-12 15:30:53",
9.       "filename": "noFile",
10.      "unit": 4,
11.      "nb_mail": 2,
12.      "open_tag": 0,
13.      "type_mail": "html",
14.      "duplicates": 1
15.    }
16.  }
17. }
```


3.13.Effectuer une campagne de messages avec critères (Push)

- **URL:** <http://webservicesmultimedias.clever-is.fr/api/pushscriterias>
- **Commande :** POST
- **Paramètres :**

Nom paramètre	Obligatoire	Commentaire
number_list	<i>oui</i>	Tableau de numéros format international +336xxxxxxxx
text	<i>oui</i>	
encodage	<i>oui</i>	3 pour le Latin et 4 pour l'UCS2
criteria	<i>oui</i>	$\sum field=value$ avec des opérateurs logiques ($field='value'$ AND $fieldl='value'$...)
short_code	non	11 caractères maximum
hd_send	non	Format Y-m-d H:i:s
filename	non	Extension : wav ou mp3 Caractères interdits (<, >, :, ", /, \, , ?, *)
type_push	non	Par défaut : text, sinon : tts ou voice
options	non	F0 pour les messages Flash F1 pour le mode normal
gsm_routage_mo	non	numéro format international +336xxxxxxxx
mail_routage_mo	non	
all_number		Par défaut 0, sinon 1 pour envoi sur tous les numéros du contact
cascade		Par défaut 0, sinon 1 pour envoi en cascade sur les numéros secondaires
time_cascade		Temps de cascade en minute
predefined_message	non	Par défaut 0, sinon 1, dans ce cas, le champ text prendra le titre du message prédéfini à envoyer.
duplicates	non	Par défaut 0, duplicates=1 pour autoriser les doublons

- Retour :

Code	Message	Signification
201	Created	La campagne (Push) est créée avec succès.
400	Bad Request	Les paramètres de la requête sont incorrects.
402	Payment Required	Le crédit du compte est insuffisant. Paiement requis.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- o **Corps de requête :** Objet datas JSON

```
{
  "datas":
  {
    "text": "send push criteria test",
    "criteria": "diffusion=on and BRINGUP=oui",
    "type_push": "sms",
    "encodage": 3,
    "options": "F1",
    "duplicates": 0
  }
}
```

○ **Corps de la réponse (Body) en cas de code 201 :**

```
{
  "push": {
    "reference": " Push201710121707081059df851c0e82a0484665235b4cd228f",
    "nb_destination": 1,
    "number_list": "+33600000001",
    "encodage": 3,
    "text": " send push criteria test",
    "hd_send": "2017-10-12 17:07:08",
    "filename": "noFile",
    "type_push": "sms",
    "unit": 7,
    "nb_sms": 1,
    "duplicates": 0
  }
}
```

push : objet retourné.

reference : référence (Id) de la campagne (push) de messages.

nb_destination : nombre de destinataires dans la campagne.

number_list : liste des destinataires de la campagne.

text : message envoyé dans la campagne.

encodage : encodage du texte de la campagne.

hd_send : date d'envoi de la campagne.

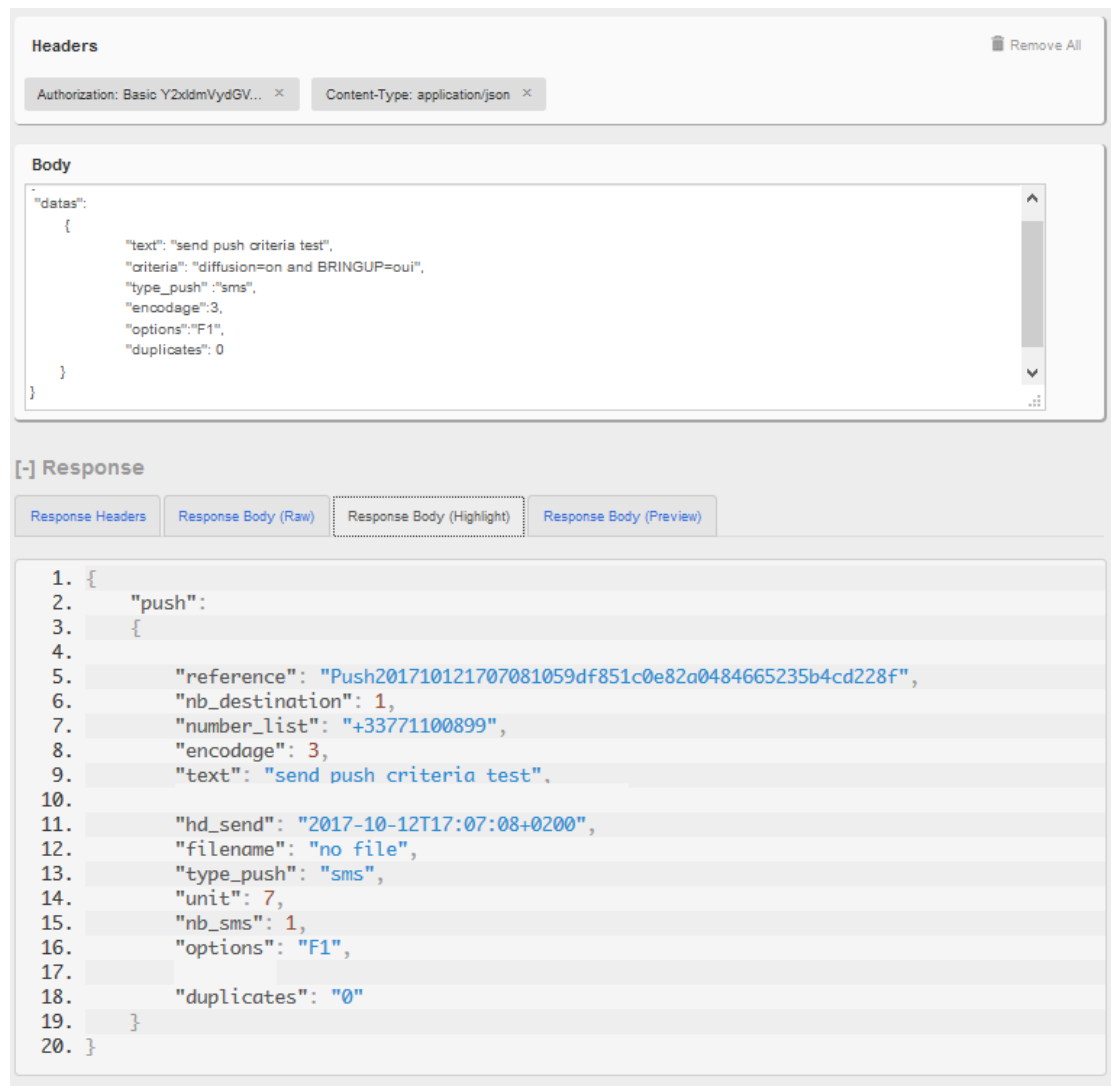
filename : nom du fichier envoyé dans la campagne (cas de fichier wave).

type_push : type de la campagne.

unit : coût de la campagne en unités.

nb_sms : nombre de SMS pour le message de la campagne.

duplicates : coût de la campagne en unités.



3.14. Effectuer une campagne de mails avec critères (Push)

- **URL:** <http://webservicesmultimedias.clever-is.fr/api/pushsmailcriterias>
- **Commande :** POST
- **Paramètres :**

Nom paramètre	Obligatoire	Commentaire
<i>email_list</i>	<i>oui</i>	Tableau de mails
<i>text</i>	<i>oui</i>	
<i>email_subject</i>	<i>oui</i>	
<i>criteria</i>	<i>oui</i>	$\sum field=$ value avec des opérateurs logiques ($field='value'$ AND $fieldl='value'$...)
type_mail	non	Par défaut : brut sinon html
hd_send	non	Format Y-m-d H:i:s
filename	non	Extension : pdf
type_push	non	mail
open_tag	non	Par défaut 0 sinon 1
predefined_message	non	Par défaut 0, sinon 1, dans ce cas, le champ text prendra le titre du message prédéfini à envoyer.
duplicates	non	Par défaut 0, duplicates=1 pour autoriser les doublons

- Retour :

Code	Message	Signification
201	Created	La campagne (Push) est créée avec succès.
400	Bad Request	Les paramètres de la requête sont incorrects.
402	Payment Required	Le crédit du compte est insuffisant. Paiement requis.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON

```
{
  "datas":
  {
    "text": "send push criteria test",
    "criteria": "diffusion=on and BRINGUP=oui",
    "email_subject": "push mail",
    "open_tag": 0,
    "type_mail": "html",
    "duplicates": 1
  }
}
```

- **Corps de la réponse (Body) en cas de code 201 :**

```
{
  "push": {
    "reference": "Push20171012173014",
    "email_list": "test@dns.fr",
    "status": 2,
    "text": "send push mail criteria test",
    "email_subject": "push mail",
    "hd_send": "2017-10-12 17:30:14",
    "type_push": "mail",
    "unit": 2,
    "nb_mail": 1,
    "open_tag": 0,
    "type_mail": "html",
    "duplicates": 1
  }
}
```

push : objet retourné.

reference : référence (Id) de la campagne (push) de messages.

email_list : liste des destinataires de la campagne.

text : message envoyé dans la campagne.

email_subject : sujet du mail de la campagne.

hd_send : date d'envoi de la campagne.

filename : nom du fichier envoyé dans la campagne (cas de pièce jointe).

type_mail : type du mail de la campagne.

unit : coût de la campagne en unités.

nb_mail : nombre de destinataires dans la campagne.

open_tag : tag pour tracker l'ouverture des mails de la campagne.

duplicates : coût de la campagne en unités.

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

- Method:** POST
- URL:** `ias.clever-is.fr/api/pushmailcriteria`
- Headers:**
 - Authorization: Basic Y2xldmVydGV...
 - Content-Type: application/json
- Body:**

```
{
  "datas": {
    "text": "send push criteria test",
    "criteria": "diffusion=on and BRINGUP=oui",
    "email_subject": "push mail",
    "open_tag": 0,
    "duplicates": 1,
    "type_mail": "html"
  }
}
```
- Response:**

```
1. {
2.   "push":
3.   {
4.     "reference": "Push20171012173014",
5.     "email_list": "cleverimap@gmail.com ",
6.     "statut": 0,
7.     "text": "send push criteria test",
8.     "email_subject": "push mail",
9.     "hd_send": "2017-10-12 17:30:14 ",
10.    "type_push": "mail",
11.    "unit": 2,
12.    "nb_mail": 1,
13.    "open_tag": 0,
14.    "type_mail": "html",
15.    "duplicates": 1
16.  }
17. }
```

3.15.Effectuer une campagne de messages à tous les contacts (Push)

- **URL:** <http://webservicesmultimedias.clever-is.fr/api/pushsbroadcasts>
- **Commande :** POST
- **Paramètres :**

Nom paramètre	Obligatoire	Commentaire
number_list	<i>oui</i>	Tableau de numéros format international +336xxxxxxxx
text	<i>oui</i>	
encodage	<i>oui</i>	3 pour le Latin et 4 pour l'UCS2
short_code	non	11 caractères maximum
hd_send	non	Format Y-m-d H:i:s
filename	non	Extension : wav ou mp3 Caractères interdits (<, >, :, ", /, \, , ?, *)
type_push	non	Par défaut : text, sinon : tts ou voice
options	non	F0 pour les messages Flash F1 pour le mode normal
gsm_routage_mo	non	numéro format international +336xxxxxxxx
mail_routage_mo	non	
all_number		Par défaut 0, sinon 1 pour envoi sur tous les numéros du contact
cascade		Par défaut 0, sinon 1 pour envoi en cascade sur les numéros secondaires
time_cascade		Temps de cascade en minute
predefined_message	non	Par défaut 0, sinon 1, dans ce cas, le champ text prendra le titre du message prédéfini à envoyer.
duplicates	non	Par défaut 0, duplicates=1 pour autoriser les doublons

- **Retour :**

Code	Message	Signification
201	Created	La campagne (Push) est créée avec succès.
400	Bad Request	Les paramètres de la requête sont incorrects.
402	Payment Required	Le crédit du compte est insuffisant. Paiement requis.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple :**

- **Corps de requête :** Objet datas JSON

```
{
  "datas":
  {
    "text": "send push all contacts test",
    "type_push": "sms",
    "encodage": 3,
    "options": "F1",
    "gsm_routage_mo": "+336000000012",
    "mail_routage_mo": "test@clever.fr",
    "duplicates": 0
  }
}
```

○ **Corps de la réponse (Body) en cas de code 201 :**

```
{
  "push": {
    "reference": "Push201710121805301059df92caac8d629486258997453f8d0",
    "nb_destination": 0,
    "number_list":
"+33651028452;+33771100899;+33699593839;+33600000001;+33770304614;+33151028452",
    "encodage": 3,
    "text": "send push all contacts test",
    "date_entry": "2017-10-12T18:05:30+0200",
    "hd_send": "2017-10-12T18:05:30+0200",
    "type_push": "sms",
    "unit": 44,
    "nb_sms": 1,
    "options": "F1",
    "way": 1,
    "duplicates": "0",
    "mail_routage_mo": "ybelasli@cleverinterne.fr",
    "gsm_routage_mo": "+33651028452"
  }
}
```

push : objet retourné.

reference : référence (Id) de la campagne (push) de messages.

nb_destination : nombre de destinataires dans la campagne.

number_list : liste des destinataires de la campagne.

text : message envoyé dans la campagne.

encodage : encodage du texte de la campagne.

hd_send : date d'envoi de la campagne.

filename : nom du fichier envoyé dans la campagne (cas de fichier wave).

type_push : type de la campagne.

unit : coût de la campagne en unités.

nb_sms : nombre de SMS pour le message de la campagne.

duplicates : coût de la campagne en unités.

Method **POST** URL **sdias.clever-is.fr/api/pushsbroadcasts** **SEND**

Headers Remove All

Authorization: Basic Y2xldmVydGV... Content-Type: application/json

Body

```
{
  "datas":
  {
    "text": "send push all contacts test",
    "type_push": "sms",
    "encodage": 3,
    "options": "F1",
    "gsm_routage_mo": "+336000000012",
    "mail_routage_mo": "test@clever.fr",
    "duplicates": 0
  }
}
```

[+] Response

Response Headers Response Body (Raw) Response Body (Highlight) Response Body (Preview)

```
1. {
2.   "push":
3.     {
4.       "id": 676,
5.       "reference": "Push201710121805301059df92caac8d629486258997453f8d0",
6.       "nb_destination": 0,
7.       "number_list": "+33651028452;+33771100899;+33699593839;+33600000001;+33770304614;+3
8.       3151028452",
9.       "encodage": 3,
10.      "text": "send push all contacts test",
11.      "date_entry": "2017-10-12T18:05:30+0200",
12.      "hd_send": "2017-10-12T18:05:30+0200",
13.      "type_push": "sms",
14.      "unit": 44,
15.      "nb_sms": 1,
16.      "options": "F1",
17.      "way": 1,
18.      "duplicates": "0",
19.      "mail_routage_mo": "ybelasli@cleverinterne.fr",
20.      "gsm_routage_mo": "+33651028452"
21.    }
22.  }
```

3.16. Effectuer une campagne de mails à tous les contacts (Push)

- **URL:** <http://webservicesmultimedias.clever-is.fr/api/pushsmailbroadcasts>
- **Commande :** POST
- **Paramètres :**

Nom paramètre	Obligatoire	Commentaire
<i>email_list</i>	<i>oui</i>	<i>Tableau de mails</i>
<i>text</i>	<i>oui</i>	
<i>email_subject</i>	<i>oui</i>	
type_mail	non	Par défaut : brut sinon html
hd_send	non	Format Y-m-d H:i:s
filename	non	Extension : pdf
type_push	non	mail
open_tag	non	Par défaut 0 sinon 1
predefined_message	non	Par défaut 0, sinon 1, dans ce cas, le champ text prendra le titre du message prédéfini à envoyer.
duplicates	non	Par défaut 0, duplicates=1 pour autoriser les doublons

- Retour :

Code	Message	Signification
201	Created	La campagne (Push) est créée avec succès.
400	Bad Request	Les paramètres de la requête sont incorrects.
402	Payment Required	Le crédit du compte est insuffisant. Paiement requis.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON

```
{
  "datas":
  {
    "text": "Send Push mail for all contacts",
    "email_subject": "push mail",
    "open_tag": 0,
    "duplicates": 1,
    "type_mail": "html"
  }
}
```

- **Corps de la réponse (Body) en cas de code 201 :**

```
{
  "push": {
    "reference": "Push20171012174641",
    "email_list": "test@clever.fr,cleverimap@gmail.com ",
    "text": "Send Push mail for all contacts",
    "email_subject": "push mail",
    "hd_send": "2017-10-12 17:46:41",
    "type_push": "mail",
    "unit": 4,
    "nb_mail": 2,
    "open_tag": 0,
    "type_mail": "html",
    "duplicates": 1
  }
}
```

push : objet retourné.

reference : référence (Id) de la campagne (push) de messages.

email_list : liste des destinataires de la campagne.

text : message envoyé dans la campagne.

email_subject : sujet du mail de la campagne.

hd_send : date d'envoi de la campagne.

filename : nom du fichier envoyé dans la campagne (cas de pièce jointe).

type_mail : type du mail de la campagne.

unit : coût de la campagne en unités.

nb_mail : nombre de destinataires dans la campagne.

open_tag : tag pour tracker l'ouverture des mails de la campagne.

duplicates : coût de la campagne en unités.

Method: POST URL: clever-is.fr/api/pushmailbroadcasts SEND

Headers: Authorization: Basic Y2xldmVydGV... Content-Type: application/json Remove All

Body:

```
{
  "datas": [
    {
      "text": "Send Push mail for all contacts",
      "email_subject": "push mail",
      "open_tag": 0,
      "duplicates": 1,
      "type_mail": "html"
    }
  ]
}
```

[+] Response

Response Headers Response Body (Raw) Response Body (Highlight) Response Body (Preview)

```
1. {
2.   "push":
3.   {
4.     "id": 675,
5.     "reference": "Push20171012174641",
6.     "email_list": "cleverimap@gmail.com,support@clever.fr",
7.
8.     "text": "Send Push mail for all contacts",
9.     "email_subject": "push mail",
10.
11.     "hd_send": "2017-10-12 17:46:41",
12.
13.     "type_push": "mail",
14.     "unit": 14,
15.     "nb_mail": 7,
16.     "open_tag": 0,
17.
18.     "type_mail": "html",
19.     "duplicates": 1
20.   }
21. }
```

3.17. Supprimer une campagne de messages différée (Push)

- **URL:** `http://webservicesmultimedias.clever-is.fr/api/pushs/{referencePush}`
- **Commande :** DELETE
- **Paramètres :** Vide (Paramètre d'URL `referencePush`)
- **Retour :**

Code	Message	Signification
204	No Content	La requête s'est correctement exécutée. Pas de corps de réponse.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
404	Not Found	La ressource demandée n'existe pas.
405	Method Not Allowed	L'opération demandée n'est pas permise.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple :**
 - **Corps de requête :** Objet datas JSON
L'opération demandée de corps de requête.
 - **Corps de la réponse (Body) en cas de code 204 :**
L'opération demandée a réussi et il n'y a pas de corps de réponse.

The screenshot shows an API client interface with the following details:

- Method:** DELETE
- URL:** `s.fr/api/pushs/Push20171013131152`
- Headers:**
 - Authorization: Basic Y2xldmVydGV...
 - Content-Type: application/json
- Body:** Request Body (empty)
- Response:**
 - Status Code: 204 No Content
 - Cache-Control: no-cache
 - Connection: Keep-Alive
 - Content-Type: text/html; charset=UTF-8
 - Date: Fri, 13 Oct 2017 11:24:56 GMT
 - Keep-Alive: timeout=5, max=100
 - Server: Apache/2.4.23 (Win64) OpenSSL/1.0.2j PHP/5.6.25
 - X-Powered-By: PHP/5.6.25

3.18. Supprimer une campagne de mails différée (Push)

- **URL:** `http://webservicesmultimedias.clever-is.fr/api/pushsmails/{referencePush}`
- **Commande :** DELETE
- **Paramètres :** Vide (Paramètre d'URL referencePush)

- **Retour :**

Code	Message	Signification
204	No Content	La requête s'est correctement exécutée. Pas de corps de réponse.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
404	Not Found	La ressource demandée n'existe pas.
405	Method Not Allowed	L'opération demandée n'est pas permise.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple :**

- **Corps de requête :** Objet datas JSON
L'opération demandée de corps de requête.
- **Corps de la réponse (Body) en cas de code 204 :**
L'opération demandée a réussi et il n'y a pas de corps de réponse.

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

- Method:** DELETE
- URL:** `/pushsmails/Push20171013135153`
- Headers:**
 - Authorization: Basic Y2xldmVydGV...
 - Content-Type: application/json
- Body:** Request Body (empty)
- Response:**
 - Status Code: 204 No Content
 - Cache-Control: no-cache
 - Connection: Keep-Alive
 - Content-Type: text/html; charset=UTF-8
 - Date: Fri, 13 Oct 2017 11:57:02 GMT
 - Keep-Alive: timeout=5, max=100
 - Server: Apache/2.4.23 (Win64) OpenSSL/1.0.2j PHP/5.6.25
 - X-Powered-By: PHP/5.6.25

3.19. Obtenir les informations d'une campagne de messages (Push)

- **URL**: `http://webservicesmultimedias.clever-is.fr/api/pushs/{referencePush}`
- **Commande** : GET
- **Paramètres** : Vide (Paramètre d'URL `referencePush`)

- **Retour** :

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
405	Method Not Allowed	L'opération demandée n'est pas permise.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple** :

- **Corps de requête** : Objet datas JSON
L'opération demandée de corps de requête.
- **Corps de la réponse (Body) en cas de code 200** :

```
{
  "push": {
    "reference": "Push20171012135655",
    "nb_destination": 1,
    "number_list": "+336000000001",
    "status": 2,
    "short_code": "",
    "encodage": 3,
    "text": "test envoi du message",
    "hd_send": "2017-10-12 13:56:55",
    "type_push": "tts",
    "unit": 9,
    "nb_sms": 1,
    "options": "F1"
  }
}
```

push : objet retourné.

reference : référence (Id) de la campagne (push) de messages.

nb_destination : nombre de destinataires dans la campagne.

number_list : liste des destinataires de la campagne.

status : statut de la campagne (Statut 0 : non traité , Statut 1 : en cours , Statut 2 : push traité).

short_code : émetteur d'envoi des messages de la campagne

text : message envoyé dans la campagne.

encodage : encodage du texte de la campagne.

hd_send : date d'envoi de la campagne.

type_push : type de la campagne.

filename : nom du fichier envoyé dans la campagne (cas de fichier wave).

unit : coût de la campagne en unités.

nb_sms : nombre de SMS pour le message de la campagne.

options : F0 pour les messages Flash F1 pour le mode normal.

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

- Method:** GET
- URL:** http://webservicesmultimedias.cle
- Headers:** Authorization: Basic Y2xldmVydGV... , Content-Type: application/json
- Body:** Request Body
- Response:** JSON data

```
1. {
2.   "push":
3.   {
4.     "reference": "Push20171012135655",
5.     "nb_destination": 1,
6.     "number_list": "+33600000001",
7.     "status": 2,
8.     "short_code": "",
9.     "encodage": 3,
10.    "text": "test envoi du message",
11.    "hd_send": "2017-10-12T13:56:55+0200",
12.    "type_push": "tts",
13.    "unit": 9,
14.    "nb_sms": 1,
15.    "options": "F1"
16.  }
17. }
```

3.20. Obtenir les informations d'une campagne mails (Push)

- **URL**: `http://webservicesmultimedias.clever-is.fr/api/pushsmails/{referencePush}`
- **Commande** : GET
- **Paramètres** : Vide (Paramètre d'URL `referencePush`)

- **Retour** :

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
405	Method Not Allowed	L'opération demandée n'est pas permise.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple** :

- **Corps de requête** : Objet datas JSON
L'opération demandée de corps de requête.
- **Corps de la réponse (Body) en cas de code 200** :

```
{
  "push": {
    "reference": "Push20171012153053",
    "email_list": "cleverimap@gmail.com;cleverimap@gmail.com",
    "status": 2,
    "text": "Send Push Mail",
    "email_subject": "push mail",
    "hd_send": "2017-10-12 15:30:53",
    "type_push": "mail",
    "unit": 4,
    "nb_mail": 2,
    "open_tag": 0,
    "type_mail": "html"
  }
}
```

push : objet retourné.

reference : référence (Id) de la campagne (push) de messages.

email_list : liste des destinataires de la campagne.

status : statut de la campagne (Statut 0 : non traité , Statut 1 : en cours , Statut 2 : push traité).

text : message envoyé dans la campagne.

email_subject : sujet du mail de la campagne.

hd_send : date d'envoi de la campagne.

filename : nom du fichier envoyé dans la campagne (cas de pièce jointe).

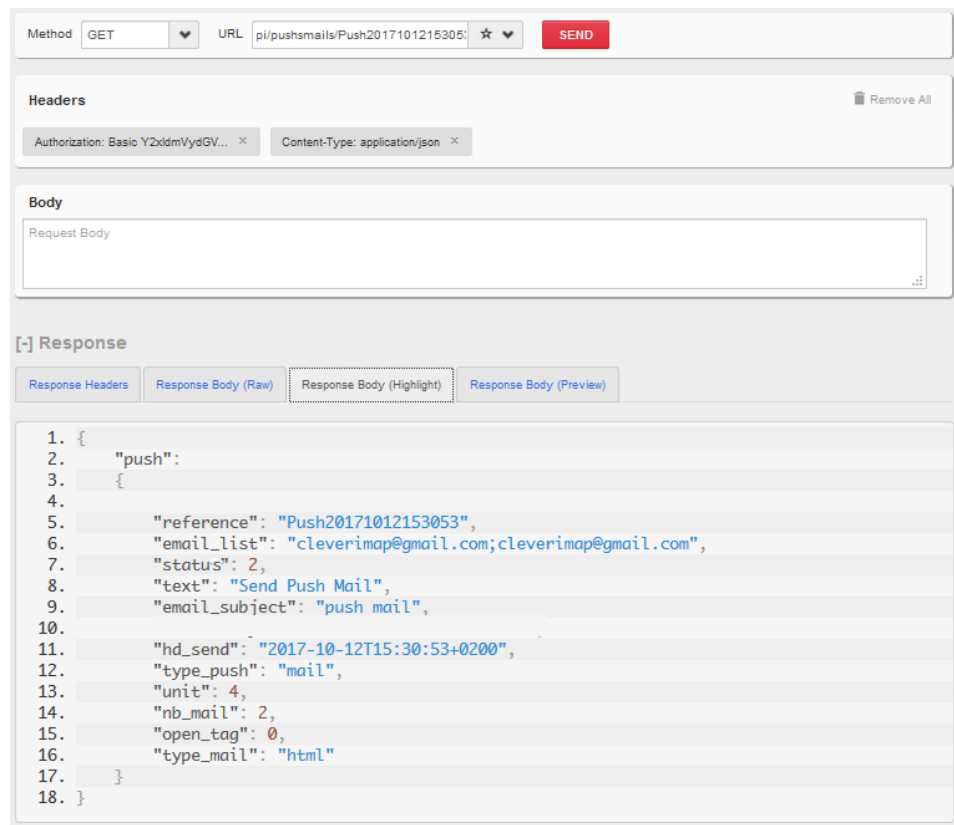
type_mail : type du mail de la campagne.

unit : coût de la campagne en unités.

nb_mail : nombre de destinataires dans la campagne.

open_tag : tag pour tracker l'ouverture des mails de la campagne.

duplicates : coût de la campagne en unités.



3.21. Obtenir les acquittements d'une campagne de messages (Push)

- **URL**: <http://webservicesmultimedias.clever-is.fr/api/pushs/{referencePush}/deliveries>
- **Commande** : GET
- **Paramètres** : Vide (Paramètre d'URL referencePush)
- **Retour** :

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
405	Method Not Allowed	L'opération demandée n'est pas permise.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple** :
 - **Corps de requête** : Objet datas JSON
L'opération demandée de corps de requête.

- **Corps de la réponse (Body) en cas de code 200 :**

```
{
  "deliveries": [
    {
      "message": "send push all contacts test",
      "typemessage": "sms",
      "date_send": "2017-10-13 16:24:51",
      "date_receive": "2017-10-13 16:24:52",
      "recipient": "+33651028452",
      "status": "3"
    },
    {
      "message": "send push all contacts test",
      "typemessage": "tts",
      "date_send": "2017-10-13 16:24:55",
      "date_receive": "2017-10-13 16:25:10",
      "recipient": "+336000000001",
      "status": "6"
    }
  ]
}
```

deliveries : objet retourné.

status : statut du message.

typemessage : type du message.

message : message envoyé.

date_send : date d'envoi du message.

date_receive : date de réception du message.

recipient : numéro de téléphone du destinataire du message.

Method: GET URL: http://webservicesmultimedias.clever-is.fr/api/pushmails/{referencePush}/deliveries

Headers: Authorization: Basic Y2xldmVydGV... Content-Type: application/json

Body: Request Body

[-] Response

```
1. {
2.   "deliveries":
3.     [
4.       {
5.         "message": "send push all contacts test",
6.         "typemessage": "sms",
7.         "date_send": "2017-10-13 16:24:51",
8.         "date_receive": "2017-10-13 16:24:52",
9.         "recipient": "+33600000001",
10.        "status": "3"
11.      },
12.      {
13.        "message": "send push all contacts test",
14.        "typemessage": "sms",
15.        "date_send": "2017-10-13 16:25:02",
16.        "recipient": "+33600000001",
17.        "status": "2"
18.      },
19.      {
20.        "message": "send push all contacts test".
```

3.22. Obtenir les acquittements d'une campagne mails (Push)

- **URL:** <http://webservicesmultimedias.clever-is.fr/api/pushmails/{referencePush}/deliveries>
- **Commande :** GET
- **Paramètres :** Vide (Paramètre d'URL referencePush)
- **Retour :**

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
405	Method Not Allowed	L'opération demandée n'est pas permise.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- Exemple :

- **Corps de requête :** Objet datas JSON
L'opération demandée de corps de requête.
- **Corps de la réponse (Body) en cas de code 200 :**

```
{
  "deliveries": [
    {
      "subject": "push mail",
      "body": "Send Push Mail test",
      "reply_to": "",
      "date_send": "2017-10-13 13:35:10",
      "date_receive": "2017-10-13 13:35:12",
      "recipient": "cleverimap@gmail.com",
      "status": "3"
    },
    {
      "subject": "push mail",
      "body": "Send Push Mail test",
      "reply_to": "",
      "date_send": "2017-10-13 13:35:10",
      "date_receive": "2017-10-13 13:35:12",
      "recipient": "cleverimap@gmail.com",
      "status": "3"
    }
  ]
}
```

Method: GET URL: <http://webservicesmultimedias.clever-is.fr/api/pushs/{referencePush}/mos> SEND

Headers Remove All

Authorization: Basic Y2xldmVydGV... Content-Type: application/json

Body

Request Body

[+] Response

Response Headers Response Body (Raw) Response Body (Highlight) Response Body (Preview)

```

1. {
2.   "deliveries":
3.   [
4.     {
5.       "subject": "push mail",
6.       "body": "Send Push Mail test",
7.       "reply_to": "",
8.       "date_send": "2017-10-13 13:35:10",
9.       "recipient": "cleverimap@gmail.com",
10.      "status": "2"
11.    },
12.    {
13.      "subject": "push mail",
14.      "body": "Send Push Mail test",
15.      "reply_to": "",
16.      "date_send": "2017-10-13 13:35:10",
17.      "recipient": "cleverimap@gmail.com",
18.      "status": "2"
19.    }
20.  ]
21. }
```

3.23. Obtenir les MO's d'une campagne de messages (Push)

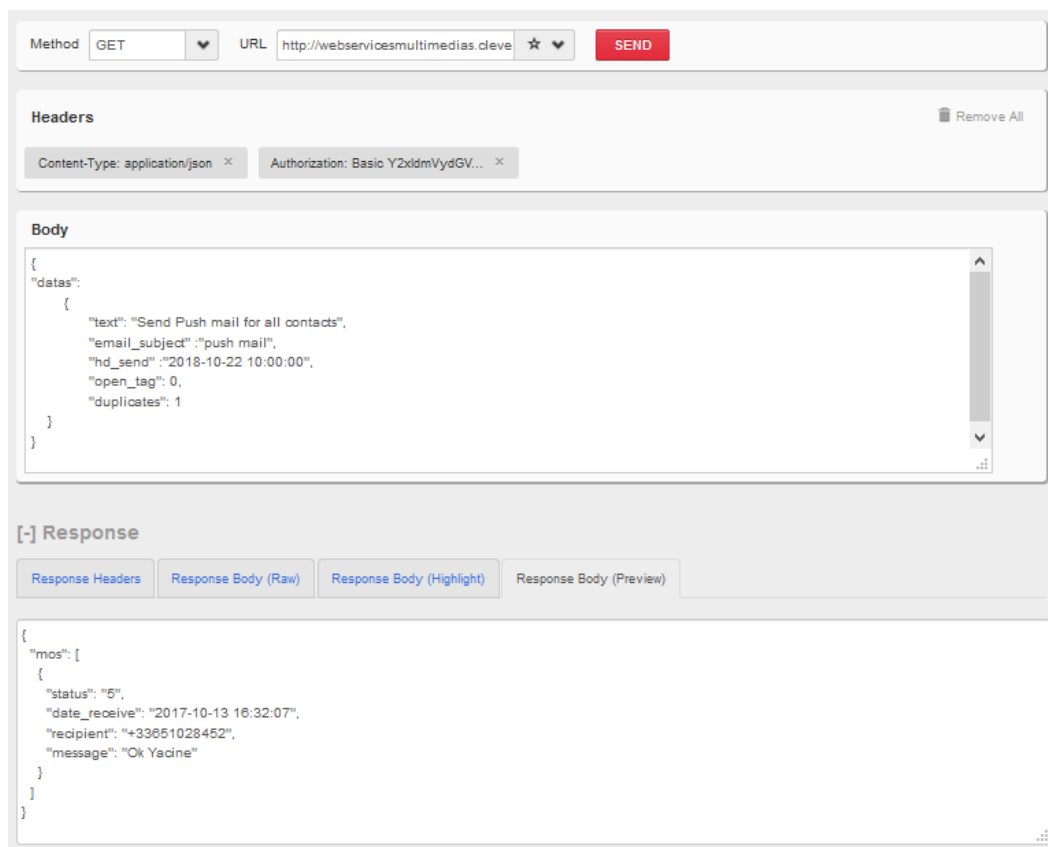
- **URL:** <http://webservicesmultimedias.clever-is.fr/api/pushs/{referencePush}/mos>
- **Commande :** GET
- **Paramètres :** Vide (Paramètre d'URL referencePush)
- **Retour :**

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
405	Method Not Allowed	L'opération demandée n'est pas permise.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

- **Exemple :**
 - **Corps de requête :** Objet datas JSON
L'opération demandée de corps de requête.

○ **Corps de la réponse (Body) en cas de code 200 :**

```
{
  "mos": [
    {
      "status": "5",
      "date_receive": "2017-10-13 16:32:07",
      "recipient": "+33651028452",
      "message": "Ok Yacine"
    }
  ]
}
```



3.24. Lister les champs additionnels (à venir)

- **URL:** <http://webservicesmultimedias.clever-is.fr/api/fieldlist>
- **Commande :** GET
- **Paramètres :** Vide
- **Retour :**

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.

3.25. Lister les messages prédéfinis (à venir)

- **URL**: <http://webservicesmultimedias.clever-is.fr/api/predefinedmessage>
- **Commande** : GET
- **Paramètres** : Vide

- **Retour** :

Code	Message	Signification
200	Ok	La requête s'est correctement exécutée.
400	Bad Request	Les paramètres de la requête sont incorrects.
401	Unauthorized	L'authentification est incorrecte.
430	Account expired YYYY-mm-JJ	Le compte est expiré.
500	Internal Server Error	Erreur du service lors du traitement de la requête.