**> Mode opératoire pour activer et configurer le module REST Drupal 7.34**
1.
Récupérer le core Drupal 7.34
<https://www.drupal.org/drupal-7.34-release-notes>

2.
Installer Drupal en configuration “standard”
Ici un lien concernant l’installation seule :

3.
Mettre en place le vhost pour redirection (facultatif mais démo friendly,
“drupal-example.com” dans mon cas)

4.
Activer le module REST :
4.1.
Télécharger les modules suivants :
4.1.1.
[CTools​](http://drupal.org/project/ctools)
(7.x-1.2)
4.1.2.
[Devel​](http://drupal.org/project/devel)
(7.x-1.3)
4.1.3.
[Libraries](https://www.drupal.org/project/libraries)​
 (7.x-2.2)
4.1.4.
[OAuth​](http://drupal.org/project/oauth)
(7.x-3.1)\*
4.1.5.
[Services​](http://drupal.org/project/services)
(7.x-3.3)
4.1.6.
[Services Views​](http://drupal.org/project/services_views)
(7.x-1.0-beta2+10-dev)
4.1.7.
[Views​](http://drupal.org/project/views)
(7.x-3.5)

4.2.
Les extraire dans sites/all/modules

4.3.
Dans l’administration, accéder à admin/modules, puis activer les modules
suivants :
4.3.1.
Chaos tools
4.3.2.
Libraries
4.3.3.
Services
4.3.4.
Services Views
4.3.5.
REST Server
4.3.6.
Views
4.3.7.
Views UI



5.
Ajouter un Service dans Services UI dans l’administration en accédant à
/Structure/Services/Add, renseigner les champs (cf screenshot) et sauvegarder.
note : dans ma compréhension l’api REST Drupal n’est pleinement accessible que pour un
utilisateur authentifié.

1er champ : jio
2nd champ : REST
3ème champ : api/jio
Champ Session Authentification coché

6.
Configurer les ressources accessibles en cliquant sur “edit ressources”, cocher “files” et
“user” puis sauvegarder


7.
Tester l’accès à l’endpoint
requête GET sur ​
http://​
<domain>​
/?q=​
<endpoint>​
/

8.
Ajouter le verbe “OPTIONS”
Afin de permettre les requêtes sur l’api depuis les navigateurs modernes, il faut ajouter
le verbe “OPTIONS” aux verbes exposés par l’API REST Drupal. En effet, les navigateurs
modernes effectuent une requête discrète de type “OPTIONS” dite “pre-flight” afin de
valider que la requête GET suivante est possible.

Dans le .htaccess, ajouter :
# Send a 200 HTTP response on OPTIONS request
RewriteCond %{REQUEST\_METHOD} OPTIONS
RewriteRule .\* / [R=200,L]

dans un bloc :
<IfModule mod\_headers.c></IfModule>

9.
Optionnel : activer les CORS
9.1.
Méthode .htaccess
Ajouter en fin de fichier :
<IfModule mod\_headers.c>
    SetEnvIf Origin "^http(s)?://(.+\.)?([NOMDEDOMAINE])$" DYN\_ORIGIN=$0
    Header set Access-Control-Allow-Origin %{DYN\_ORIGIN}e
    Header always set Access-Control-Allow-Methods "GET, PUT, POST, OPTIONS, DELETE"
    Header always set Access-Control-Allow-Credentials true
    Header always set Access-Control-Allow-Headers "Authorization, Origin, Content-Type,
X-CSRF-Token"
</IfModule>

9.2.
méthode module CORS
Télécharger le module[​](https://www.drupal.org/project/cors)
 [CORS​](https://www.drupal.org/project/cors)
(7.x-1.3)
Les extraire dans sites/all/modules
Activer dans admin/module
accéder à la configuration du module via configuration/CORS et inscrire dans la vue :
\*|\*|GET, PUT, POST, DELETE, OPTIONS|Authorization, Accept, Origin, Content-Type, X-CSRF-Token|true