
TaxHub Documentation

Version latest

sept. 19, 2018

Table des matières

1	SERVEUR	3
1.1	Prérequis	3
1.2	Installation et configuration du serveur	3
1.3	Installation et configuration de PosgreSQL	4
2	APPLICATION	7
2.1	Prérequis	7
2.2	Configuration initiale	7
2.3	Configuration Apache	8
2.4	Création de la base de données	8
2.5	Installation de l'application	8
2.6	Arrêter/Lancer l'application	9
2.7	Mise à jour de l'application	9
2.8	Développement	9
3	DEVELOPPEMENT	11
3.1	Routes Symfony	11
3.2	Bla bla bla	14
4	AUTEURS	15
4.1	Parc national des Ecrins	15
4.2	Parc national des Cevennes	15
4.3	Parc national de la Vanoise	15
4.4	Parc national des Calanques	16
4.5	Parc national du Mercantour	16
5	CHANGELOG	17
5.1	1.5.0 (2018-09-19)	17
5.2	1.4.1 (2018-08-20)	17
5.3	1.4.0 (2018-07-12)	17
5.4	1.3.2 (2017-12-15)	18
5.5	1.3.1 (2017-09-26)	18
5.6	1.3.0 (2017-09-20)	18
5.7	1.2.1 (2017-07-04)	19
5.8	1.2.0 (2017-06-21)	19
5.9	1.1.2 (2017-02-23)	20
5.10	1.1.1 (2016-12-14)	21

5.11	1.1.0 (2016-11-17)	21
5.12	1.0.0 (2016-09-06)	21
5.13	0.1.0 (2016-05-12)	22
5.14	0.0.1 (2015-04-01)	22

Contenu :

1.1 Prérequis

- Ressources minimum serveur :

Un serveur disposant d'au moins de 1 Go RAM et de 10 Go d'espace disque.

- disposer d'un utilisateur linux appartenant au groupe `www-data`. Cette documentation présente la procédure à suivre pour un utilisateur nommé `synthese`.

1.2 Installation et configuration du serveur

Installation pour Debian 7, 8, 9 et Ubuntu 14.04

notes Cette documentation concerne une installation sur Debian ou Ubuntu. Pour tout autre environnement les commandes sont à adapter.

notes Durant toute la procédure d'installation, travailler avec l'utilisateur `synthese`. Ne changer d'utilisateur que lorsque la documentation le spécifie.

```
su -
apt-get install apache2 libapache2-mod-proxy-html curl python-dev python-pip libpq-
↳dev libgeos-dev supervisor
pip install virtualenv
adduser --home /home/synthese synthese
usermod -g www-data synthese
usermod -a -G root synthese
adduser synthese sudo
exit
```

notes Sur Debian 8, il est nécessaire d'installer les paquets suivant pour faire fonctionner la librairie `opencv`

notes Sur Debian 8, il est nécessaire d'installer les paquets suivant pour faire fonctionner la librairie `opencv`

```
sudo apt-get install -y libsm6 libxrender1 libfontconfig1 2>/var/log/  
↳geonature/install_log.log  
sudo apt-get install -y python-qt4 2>/var/log/geonature/install_log.log
```

notes Sur Debian 9 libapache2-mod-proxy-html n'existe plus. L'application fonctionne sans ce paquet.

- Fermer la console et la réouvrir pour que les modifications soient prises en compte.
- Installer npm pour debian 7 et 8

```
su -  
sh -c 'echo "" >> /etc/apt/sources.list'  
sh -c 'echo "#Backports" >> /etc/apt/sources.list'  
sh -c 'echo "deb http://http.debian.net/debian wheezy-backports main" >> /etc/  
↳apt/sources.list'  
apt-get update  
aptitude -t wheezy-backports install nodejs  
update-alternatives --install /usr/bin/node nodejs /usr/bin/nodejs 100  
curl https://www.npmjs.com/install.sh | sh  
exit
```

- Installer npm pour debian 9

```
curl -sL https://deb.nodesource.com/setup_6.x | sudo bash -  
sudo apt install nodejs
```

- Activer le mod_rewrite et proxy_http et redémarrer Apache

```
sudo a2enmod rewrite  
sudo a2enmod proxy  
sudo a2enmod proxy_http  
sudo apache2ctl restart
```

1.3 Installation et configuration de PosgreSQL

- Sur Debian 8, Postgres est livré en version 9.4 et postgis 2.1, vous pouvez sauter l'étape suivante. Sur Debian 7, il faut revoir la configuration des dépôts pour avoir une version compatible de PostgreSQL (9.3) et PostGIS (2.1). Voir <http://foretribe.blogspot.fr/2013/12/the-postgresql-and-postgis-install-on.html>.

```
sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ wheezy-pgdg main"  
↳>> /etc/apt/sources.list'  
sudo wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo  
↳apt-key add -  
sudo apt-get update
```

- Installation de PostreSQL/PostGIS pour Debian 8

```
sudo apt-get update  
sudo apt-get install postgresql postgresql-client  
sudo apt-get install postgresql-9.4-postgis-2.1  
sudo adduser postgres sudo
```

- Installation de PostreSQL/PostGIS pour Debian 7

```
sudo apt-get install postgresql-9.3 postgresql-client-9.3  
sudo apt-get install postgresql-9.3-postgis-2.1  
sudo adduser postgres sudo
```

- Configuration de PostgreSQL pour Debian 8 - permettre l'écoute de toutes les IP


```
sed -e "s/#listen_addresses = 'localhost'/listen_addresses = '*' /g" -i /etc/
↳postgresql/9.4/main/postgresql.conf
sudo sed -e "s/# IPv4 local connections: /# IPv4 local_
↳connections:\nhost\tall\tall\t0.0.0.0\0\t md5/g" -i /etc/postgresql/9.4/main/
↳pg_hba.conf
/etc/init.d/postgresql restart
```

— Configuration de PostgreSQL pour Debian 7 - permettre l'écoute de toutes les IP

```
sed -e "s/#listen_addresses = 'localhost'/listen_addresses = '*' /g" -i /etc/
↳postgresql/9.3/main/postgresql.conf
sudo sed -e "s/# IPv4 local connections: /# IPv4 local_
↳connections:\nhost\tall\tall\t0.0.0.0\0\t md5/g" -i /etc/postgresql/9.3/main/
↳pg_hba.conf
sudo /etc/init.d/postgresql restart
```

— Création de 2 utilisateurs PostgreSQL

```
sudo su postgres
psql
CREATE ROLE geonatuser WITH LOGIN PASSWORD 'monpassachanger';
CREATE ROLE geonatadmin WITH SUPERUSER LOGIN PASSWORD 'monpassachanger';
\q
```

L'utilisateur `geonatuser` sera le propriétaire de la base de données `taxhubdb` et sera utilisé par l'application pour se connecter à celle-ci.

L'utilisateur `geonatadmin` est super utilisateur de PostgreSQL.

L'application fonctionne avec le mot de passe `monpassachanger` par défaut mais il est conseillé de le modifier !

Ce mot de passe, ainsi que les utilisateurs PostgreSQL créés ci-dessus (`geonatuser` et `geonatadmin`) sont des valeurs par défaut utilisées à plusieurs reprises dans l'application. Ils peuvent cependant être changés. S'ils doivent être changés, ils doivent l'être dans plusieurs fichiers de l'application `settings.ini` et `config.py`.

2.1 Prérequis

- Environnement serveur :

Voir le guide d'installation du serveur dans <https://github.com/PnX-SI/TaxHub/blob/master/docs/serveur.rst>

- Cette documentation présente la procédure avec un utilisateur linux nommé `synthese`. Dans ce guide, le répertoire de cet utilisateur est dans `/home/synthese`. Adapter les chemins selon votre serveur.
- Se connecter sur le serveur avec l'utilisateur `synthese` ou tout autre utilisateur linux faisant partie du groupe `www-data`.
- Récupérer le zip de l'application sur le Github du projet (`X.Y.Z` à remplacer par le numéro de version souhaitée), dézippez le dans le répertoire `/home/synthese` :

```
cd /home/synthese
wget https://github.com/PnX-SI/TaxHub/archive/X.Y.Z.zip
unzip X.Y.Z.zip
mv TaxHub-X.Y.Z/ taxhub/
```

2.2 Configuration initiale

Si Python 3 n'est pas déjà installé sur le serveur :

```
sudo apt-get install python3
```

- Créer et mettre à jour le fichier `settings.ini`

```
cd taxhub
cp settings.ini.sample settings.ini
nano settings.ini
```

Renseigner les informations nécessaires à la connexion à la base de données PostgreSQL. Il est possible mais non conseillé de laisser les valeurs proposées par défaut.

ATTENTION : Les valeurs renseignées dans ce fichier sont utilisées par le script d'installation de la base de données `install_db.sh` et par le script `install_app.sh`.

Les utilisateurs PostgreSQL doivent être en concordance avec ceux créés lors de la dernière étape de l'installation serveur Création de 2 utilisateurs PostgreSQL.

2.3 Configuration Apache

- Voici une des manières de configurer Apache via le fichier `/etc/apache2/sites-available/000-default.conf`. Vous pouvez aussi créer un virtualhost dédié à l'application.

Editer le fichier de configuration Apache ou en créer un nouveau :

```
#Nom du fichier indiqué à titre d'exemple
sudo nano /etc/apache2/sites-available/000-default.conf
```

Rajouter les informations suivantes entre les balises `VirtualHost`

```
# Configuration TaxHub
RewriteEngine on
RewriteRule "taxhub$" "taxhub/" [R]
<Location /taxhub>
    ProxyPass http://127.0.0.1:5000/
    ProxyPassReverse http://127.0.0.1:5000/
</Location>
#FIN Configuration TaxHub
```

- Activer les modules et redémarrer Apache

```
sudo a2enmod proxy
sudo a2enmod proxy_http
sudo a2enmod rewrite
sudo apache2ctl restart
```

2.4 Création de la base de données

- Lancer le fichier d'installation et de préparation de la base de données

```
cd /home/synthese/taxhub
sudo ./install_db.sh
```

notes En cas d'erreur : `could not change directory to "/home/synthese/taxhub": Permission non accordée, assurez vous que les répertoires taxhub et data/inpn aient bien des droits d'exécution pour les utilisateurs "autres"`

2.5 Installation de l'application

- Lancer le fichier d'installation et de configuration de l'application

```
./install_app.sh
```

- Tester l'accès à l'application : <http://mondomaine.fr/taxhub>

2.6 Arrêter/Lancer l'application

- Pour arrêter TaxHub

```
sudo supervisorctl stop taxhub
```

- Pour démarrer TaxHub

```
sudo supervisorctl start taxhub
```

2.7 Mise à jour de l'application

Les différentes versions de TaxHub sont disponibles sur le Github du projet (<https://github.com/PnX-SI/TaxHub/releases>)

- Lire attentivement les notes de chaque version si il y a des spécificités (<https://github.com/PnX-SI/TaxHub/releases>). Suivre ces instructions avant de continuer la mise à jour.
- Télécharger et extraire la version souhaitée dans un répertoire séparé (où X.Y.Z est à remplacer par le numéro de la version que vous installez) :

```
cd /home/synthese/
wget https://github.com/PnX-SI/TaxHub/archive/X.Y.Z.zip
unzip X.Y.Z.zip
mv taxhub taxhub_old
mv TaxHub-X.Y.Z/ taxhub
rm X.Y.Z.zip
```

- Récupérer les anciens fichiers de configuration :

```
cp taxhub_old/settings.ini taxhub/settings.ini
cp taxhub_old/config.py taxhub/config.py
cp taxhub_old/static/app/constants.js taxhub/static/app/constants.js
```

- Récupérer les médias uploadés dans la précédente version de TaxHub :

```
cp -aR taxhub_old/static/medias/ taxhub/static/
```

- Lancer l'installation de l'application et de ses dépendances :

```
cd taxhub
./install_app.sh
```

- Une fois que l'installation est terminée et fonctionnelle, vous pouvez supprimer la version précédente de TaxHub (répertoire `taxhub_old`).

2.8 Développement

Pour lancer l'application en mode debug

```
cd ~/taxhub
make develop
```


Cette rubrique est destinée aux développeurs qui souhaiteraient...

ATTENTION, depuis la version 1.0.0, TaxHub a été migré de PHP Symfony à Python Flask. De plus des changements en BDD ont eu lieu (notamment le renommage de bib_taxons en bib_noms). Les routes et la doc ci-dessous est donc en partie caduque.

3.1 Routes Symfony

Pour avoir toutes les routes à jour, il suffit dans symfony de lancer la commande

```
php app/console router:debug
```

Aujourd'hui les différentes routes générées par symfony sont

3.1.1 Taxref

- `/taxref/[?][limit=nb]&[page=nb]&[is_ref=boolean]&[is_inbibtaxons=boolean]&[nomColonne=ValeurFiltre]*&[ilike=deb`
- Retourne les données de la table taxonomie.taxref ainsi que le id_taxon pour les taxons présents dans bib_taxons
- Méthode autorisée : GET
- **Paramètres autorisés :**
 - limit (default = 50) : nombre d'élément à retourner
 - page (default = 0) : page à retourner
 - is_ref (default = false) : ne retourne que les nom valides (cd_nom = cd_ref)
 - bibtaxononly (default = false) : ne retourne que les taxons présents dans bib_taxref (cd_nom = cd_ref)
 - [nomColonne=ValeurFiltre]* = Permet de filtrer les données sur un ou plusieurs critères. Le nom du paramètre (nom_colonne) doit correspondre a un nom de champs de la table taxref au format camel case.

- [ilike=debutChaine] = Ne revoie les données de la colonne lbNom qui commence par debutChaine
- **/taxref/{id}**
 - Retourne un enregistrement de la table taxonomie.taxref
 - Méthode autorisée : GET
 - Paramètre : l'id de l'enregistrement correspond au cd_nom du taxref
- **/taxref/distinct/{field}[?[nomColonne=ValeurFiltre]*&[ilike=debutChaine]]**
 - Retourne un distinct de la table taxonomie.taxref sur un champ spécifié
 - Méthode autorisée : GET
 - Paramètre obligatoire : le champ du distinct (n'importe quel champ de la table taxref)
 - **Paramètres facultatifs :**
 - [nomColonne=ValeurFiltre]* = Permet de filtrer les données sur un ou plusieurs critères. Le nom du paramètre (nom_colonne) doit correspondre a un nom de champs de la table taxref au format camel case.
 - [ilike=debutChaine] = Ne revoie les données de la colonne recherchée qui commence par debutChaine
 - **Exemples**
 - /taxref/distinct/phylum : retourne tous les phylum de la table
 - /taxref/distinct/famille ?regne=Plantae&ordre=Rosales : retourne les familles du regne Plantae et de l'ordre Rosales
- **/taxref/bibtaxons/[?[limit=nb]&[page=nb]&[is_ref=boolean]&[nomColonne=ValeurFiltre]*&[ilike=debutChaine]]**
 - Retourne toutes les données de la table taxonomie.taxref uniquement pour les taxons présents dans bib_taxons
 - Méthode autorisée : GET
 - **Paramètres autorisés :**
 - limit (default = 50) : nombre d "élément à retourner
 - page (default = 0) : page à retourner
 - is_ref (default = false) : ne retourne que les nom valides (cd_nom = cd_ref)
 - [nomColonne=ValeurFiltre]* = Permet de filtrer les données sur un ou plusieurs critères. Le nom du paramètre (nom_colonne) doit correspondre a un nom de champs de la table taxref au format camel case.
 - [ilike=debutChaine] = Ne revoie les données de la colonne lbNom qui commence par debutChaine
- **/taxref/hierarchie/{rang}[?[limit=nb]&[nomColonne=ValeurFiltre]*&[ilike=debutChaine]]**
 - Selection des niveaux hiérarchiques de taxref avec le nombre de taxons associés aux différents rangs
 - Méthode autorisée : GET
 - Paramètre obligatoire : le nom du rang désiré
 - **Paramètres facultatifs :**
 - limit (default = 10) : nombre d'élément à retourner
 - [nomColonne=ValeurFiltre]* = Permet de filtrer les données sur un ou plusieurs critères. Le nom du paramètre (nom_colonne) doit correspondre a un nom de champs de la table taxref au format camel case.
 - [ilike=debutChaine] = Ne revoie les taxons du rang recherché qui commence par debutChaine
 - **Exemples**
 - /hierarchie/FM ?ordre=Chiroptera&limit=1000®ne=Animalia&ilike=m : retourne la liste des familles des chiroptères qui commencent par un m
- **/taxref/hierarchiebibtaxons/{rang}[?[limit=nb]&[nomColonne=ValeurFiltre]*&[ilike=debutChaine]]**
 - Selection des niveaux hiérarchiques de taxref pour les taxons présents dans bib_taxons avec le nombre de taxons associés aux différents rangs
 - Méthode autorisée : GET
 - Paramètre obligatoire : le nom du rang désiré
 - **Paramètres facultatifs :**
 - limit (default = 10) : nombre d'élément à retourner

- [nomColonne=ValeurFiltre]* = Permet de filtrer les données sur un ou plusieurs critères. Le nom du paramètre (nom_colonne) doit correspondre a un nom de champs de la table taxref au format camel case.
- [ilike=debutChaine] = Ne revoie les taxons du rang recherché qui commence par debutChaine
- **Exemples**
 - /hierarchie/FM ?ordre=Chiroptera&limit=1000®ne=Animalia&ilike=m : retourne la liste des familles des chiroptères qui commencent par un m

3.1.2 Bibtaxons

- **/bibtaxons/[?[limit=nb]&[page=nb]&[nomColonne=ValeurFiltre]&[ilikefr=debutChaine]&[ilikelatin=debutChaine]]**
 - Retourne les données de la table taxonomie.bib_taxons
 - Méthode autorisée : GET
 - **Paramètres autorisés :**
 - limit (defaut = 50) : nombre d'élément à retourner
 - page (defaut = 0) : page à retourner
 - [nomColonne=ValeurFiltre]* = Permet de filtrer les données sur un ou plusieurs critères. Le nom du paramètre (nom_colonne) doit correspondre a un nom de champs de la table bib_taxons ou de la table taxref au format camel case.
 - [ilikelfr=debutChaine] = Ne revoie les données de la colonne nomFrançais qui commence par debutChaine
 - [ilikelatin=debutChaine] = Ne revoie les données de la colonne nomLatin qui commence par debutChaine
- **/bibtaxons/taxonomie**
 - Retourne cd_nom, cd_taxsup, lb_nom et id_rang pour les familles, ordre, classe, phylum et regne des enregistrements de la table taxonomie.bibtaxons
 - Méthode autorisée : GET
- **/bibtaxons/{id}**
 - Retourne un enregistrement de la table taxonomie.bib_taxons
 - Méthode autorisée : GET
 - Paramètre : l'id de l'enregistrement
- **/bibtaxons/{id}**
 - Création ou mise à jour d'un enregistrement dans la table taxonomie.bib_taxons
 - Méthode autorisée : POSTIPUT
 - Paramètre : l'id de l'enregistrement (si update) ou rien (si create)
- **/bibtaxons/{id}**
 - SUppression d'un enregistrement dans la table taxonomie.bib_taxons
 - Méthode autorisée : DELETE
 - Paramètre : l'id de l'enregistrement à supprimer

3.1.3 Biblistes

- **/biblistes/{id}**
 - Selection des données relatives à la ou aux listes avec les taxons associés
 - Méthode autorisée : GET
 - **Paramètres facultatifs :**
 - id : identifiant de la liste
- **/biblistes/simpleliste**
 - Selection des données contenues uniquement dans la table biblistes
 - Méthode autorisée : GET
- **/biblistes/taxonliste/{id}**

- Selection des taxons associés à la liste demandée
- Méthode autorisée : GET
- **Paramètre obligatoire :**
 - id : identifiant de la liste

3.1.4 Bibattributs

- **/bibattributs/**
 - Retourne toutes les données de la table taxonomie.bib_attributs
 - Méthode autorisée : GET
- **/bibattributs/{id}**
 - Retourne un enregistrement de la table taxonomie.bib_attributs
 - Méthode autorisée : GET
 - **Paramètre :** id : id de l'enregistrement
- **/taxonsattribut/{id}/{value}**
 - Retourne tous les taxons ayant l'attribut passé en paramètre ainsi que le nom et la valeur de l'attribut.
 - il est possible de filtrer sur la valeur de l'attribut.
 - Méthode autorisée : GET
 - **Paramètre :** id : id de l'attribut, obligatoire value : valeur de l'attribut, facultatif
- **/taxonsattribut/{regne}/{group2inpn}**
 - Retourne les attributs correspondant au(x) filtre(s) taxonomique(s) passé(s) en paramètre.
 - En base, si un attribut n'a pas de regne renseigné, c'est qu'il concerne tous les règnes. L'attribut est toujours retourné quelques soient les paramètres transmis.
 - En base, si un attribut n'a pas de group2inpn renseigné mais un regne renseigné, c'est qu'il concerne tous les group2inpn ; il est donc retourné. Soit uniquement pour le regne transmis en paramètre soit pour tous les règnes si aucun regne valide n'est transmis.
 - En base, si un attribut n'a pas de regne mais group2inpn renseigné, c'est une erreur (un group2inpn correspond forcément à un regne). L'attribut est donc toujours retourné quelque soit les paramètres transmis.
 - Méthode autorisée : GET
 - **Paramètre :** regne : facultatif group2inpn : facultatif. Ne peut être utilisé si regne n'est pas fourni.

3.2 Bla bla bla

The most minimal components required to run an instance are :

- PostGIS 2 server
- GDAL, GEOS, libproj
- gettext
- libfreetype
- libxml2, libxslt
- Usual Python dev stuff

A voir : [the list of minimal packages on Debian/Ubuntu](#).

Note : En lançant `env_dev` et `update` is recommended after a pull of new source code, but is not mandatory : `make serve` is enough most of the time.

4.1 Parc national des Ecrins

- Gil Deluermoz
- Camille Monchicourt
- Quang Pham



4.2 Parc national des Cévennes

- Amandine Sahl



4.3 Parc national de la Vanoise

- Claire Lagaye
- Christophe Chillet

4.4 Parc national des Calanques

— Damien Frazzoni

4.5 Parc national du Mercantour

— Samuel Priou
— Thomas Lebard

5.1 1.5.0 (2018-09-19)

Nouveautés

- Ajout de la possibilité de filtrer les attributs par `id_theme` ou `id_attribut` au niveau de la route `taxoninfo`
- Ajout de routes pour récupérer `bib_taxref_habitats` et `bib_taxref_categories_lr` (listes rouges nationales)
- Installation : Ajout de paramètres permettant de mieux définir les données à intégrer et séparation des scripts SQL, notamment pour ne pas imposer d'intégrer toutes les données nécessaires à GeoNature V1 (attributs et listes)
- Mise à jour de Flask (0.11.1 à 1.0.2), Jinja, psycpg2 et Werkzeug

5.2 1.4.1 (2018-08-20)

Corrections

- Correction de l'enregistrement lors du peuplement d'une liste

5.3 1.4.0 (2018-07-12)

Nouveautés

- Migration de Taxref 9 à 11 et scripts de migration (#155 et #156)
- Ajout d'un champ `comments` à la table `bib_noms` et dans le formulaire de saisie
- Passage du champ `bib_noms.nom_francais` en `varchar(1000)`, du champ `taxref.nom_vern` en `varchar(1000)` et du champ `taxref.lb_auteur` en `varchar(250)`
- Amélioration des logs et mise en place d'une rotation des logs
- Création d'une fonction pour créer les répertoires système (`create_sys_dir()`)
- Amélioration de la vue permettant de rechercher un taxon (<https://github.com/PnX-SI/GeoNature/issues/334>)

Note de version

- Ajouter le mode d'authentification dans `config.py` (<https://github.com/PnX-SI/TaxHub/blob/87fbb11d360488e97eef3a0bb68f566744c54aa6/config.py.sample#L25>)
- Exécutez les scripts de migration de Taxref 9 à 11 (`data/scripts/update_taxref_v11/`) en suivant les indications de <https://github.com/PnX-SI/TaxHub/issues/156>
- Exécutez le script SQL de mise à jour de la BDD `data/update1.3.2to1.4.0.sql`
- Suivez la procédure générique de mise à jour de l'application

5.4 1.3.2 (2017-12-15)

Nouveautés

- Optimisation du chargement des noms dans les listes
- Optimisation des requêtes
- Affichage du rang sur les fiches des taxons/noms
- Ajout d'un champ `source` et `licence` pour les médias (sans interface de saisie pour le moment). Voir #151, #126
- Script de récupération de médias depuis mediawiki-commons (expérimental). Voir #150
- Ajout d'un service de redimensionnement à la volée des images (http://URL_TAXHUB/api/tmedias/thumbnail/2241?h=400&w=600 où 2241 est l'id du média). Il est aussi possible de ne spécifier qu'une largeur ou une hauteur pour que l'image garde ses proportions sans ajouter de bandes noires. Voir #108
- Correction et compléments documentation (compatibilité Debian 9 notamment)
- Compatibilité avec Python 2

Corrections

- Ajout d'une liste vide impossible #148
- Enregistrement d'un attribut de type `select` (bug de la version 1.3.1, ce n'était pas la valeur qui était enregistrée mais l'index)

Note de version

- Vous pouvez directement passer de la version 1.1.2 à la 1.3.2 mais en suivant les différentes notes de version.
- Exécutez le script SQL de mise à jour de la BDD `data/update1.3.1to1.3.2.sql`
- Suivez la procédure générique de mise à jour de l'application

5.5 1.3.1 (2017-09-26)

Corrections

- Optimisation des performances pour le rafraîchissement d'une vue matérialisée qui est devenue une table contrôlée (`vm_taxref_list_forautocomplete`) par trigger (`trg_refresh_mv_taxref_list_forautocomplete`). Voir #134
- Utilisation du nom français de la table `bib_noms` pour la table `vm_taxref_list_forautocomplete`. Cette table permet de stocker les noms sous la forme `nom_vern|lb_nom = nom_valide` pour les formulaires de recherche d'un taxon.
- Dans la liste `taxref`, tous les noms étaient considérés comme nouveaux (plus de possibilité de modification)

Note de version

- Vous pouvez directement passer de la version 1.1.2 à la 1.3.1 mais en suivant les différentes notes de version.
- Exécutez le script SQL de mise à jour de la BDD `data/update1.3.0to1.3.1.sql`

5.6 1.3.0 (2017-09-20)

Nouveautés

- Ajout d'un trigger assurant l'unicité de la photo principale pour chaque `cd_ref` dans la table `taxonomie.t_medias`. Si on ajoute une photo principale à un taxon qui en a déjà une, alors la précédente bascule en photo
- Performances dans les modules TaxRef et Taxons : au lieu de charger toutes les données côté client, on ne charge que les données présentes à l'écran et on lance une requête AJAX à chaque changement de page ou recherche
- Valeurs des listes déroulantes des attributs par ordre alphabétique
- Formulaire BIB_NOMS : Les champs `nom latin`, `auteur` et `cd_nom` ne sont plus modifiables car ce sont des infos venant de TaxRef.
- Performances de la BDD : création d'index sur la table Taxref
- Suppression de Taxref du dépôt pour le télécharger sur <http://geonature.fr/data/inpn/> lors de l'installation automatique de la BDD
- Ajout de nombreuses fonctions et vues matérialisées dans la BDD : <https://github.com/PnX-SI/TaxHub/blob/develop/data/update1.2.0to1.3.0.sql>
- Nettoyage et amélioration des routes de l'API

Note de version

- Exécutez le script SQL de mise à jour de la BDD `data/update1.2.0to1.3.0.sql`
- Installer Python3 : `sudo apt-get install python3`
- Installer Supervisor : `sudo apt-get install supervisor`
- Compléter le fichier `settings.ini` avec les nouveaux paramètres sur la base de la version par défaut (<https://github.com/PnX-SI/TaxHub/blob/master/settings.ini.sample>)
- Supprimer le paramètre `nb_results_limit` du fichier `static/app/constants.js` (voir <https://github.com/PnX-SI/TaxHub/blob/master/static/app/constants.js.sample>)
- Arrêter le serveur HTTP Gunicorn : `make prod-stop`
- Lancer le script d'installation : `./install_app.sh`
- Vous pouvez directement passer de la version 1.1.2 à la 1.3.0 mais en suivant les notes de version de la 1.2.0.

5.7 1.2.1 (2017-07-04)

Nouveautés

- Correction de la conf Apache pour un accès à l'application sans le slash final dans l'URL
- Retrait du « v » dans le tag de la release

Note de version

- Vous pouvez directement passer de la version 1.1.2 à la 1.2.1 mais en suivant les notes de version de la 1.2.0.

5.8 1.2.0 (2017-06-21)

Nouveautés

- Ajout de toutes les fonctionnalités de gestion des listes ainsi que des noms de taxons qu'elles peuvent contenir.
- Possibilité d'exporter le contenu d'une liste de noms en CSV.
- Correction du fonctionnement de la pagination.
- Permettre la validation du formulaire d'authentification avec la touche `Entrer`.
- `Bib_noms` : ajout de la possibilité de gérer le multiselect des attributs par checkboxx.
- Utilisation de gunicorn comme serveur http et mise en place d'un makefile.
- Suppression du sous-module d'authentification en tant que sous module git et intégration de ce dernier en tant que module python.
- Mise à jour de la lib `psycpg2`.
- Installation : passage des requirements en https pour les firewall.

Note de version

- Exécutez le script SQL de mise à jour de la BDD `data/update1.1.2to1.2.0.sql`.

- Exécutez le script `install_app.sh` qui permet l'installation de `gunicorn` et la mise à jour des dépendances `python` et `javascript`.

Attention TaxHub n'utilise plus `wsgi` mais un serveur HTTP `python` nommé `Gunicorn`. Il est nécessaire de revoir la configuration `Apache` et de lancer le serveur `http Gunicorn`

- Activer le mode proxy de `Apache`

```
sudo a2enmod proxy
sudo a2enmod proxy_http
sudo apache2ctl restart
```

- Supprimer la totalité de la configuration `Apache` concernant TaxHub et remplacez-la par celle-ci :

```
# Configuration TaxHub
    <Location /taxhub>
        ProxyPass http://127.0.0.1:8000/
        ProxyPassReverse http://127.0.0.1:8000/
    </Location>
# FIN Configuration TaxHub
```

- Redémarrer `Apache` :

```
sudo service apache2 restart
```

- Lancer le serveur HTTP `Gunicorn` :

```
make prod
```

- Si vous voulez arrêter le serveur HTTP `Gunicorn` :

```
make prod-stop
```

L'application doit être disponible à l'adresse <http://monserver.ext/taxhub>

5.9 1.1.2 (2017-02-23)

Nouveautés

- Correction du code pour compatibilité avec `Angular 1.6.1`.
- Passage à `npm` pour la gestion des dépendances (librairies).
- Mise à jour du sous-module d'authentification.
- Ajout de la liste des gymnospermes oubliés.
- Création d'une liste `Saisie possible`, remplaçant l'attribut `Saisie`. Cela permet de choisir les synonymes que l'on peut saisir ou non dans `GeoNature` en se basant sur les `cd_nom` (`bib_listes` et `cor_nom_liste`) et non plus sur les `cd_ref` (`bib_attributs` et `cor_taxon_attribut`).
- Création d'une documentation standard de mise à jour de l'application.
- Bugfix (cf <https://github.com/PnX-SI/TaxHub/issues/100>).

Note de version

- Exécutez la procédure standard de mise à jour de l'application (<http://taxhub.readthedocs.io/fr/latest/installation.html#mise-a-jour-de-l-application>)
- Si vous n'avez pas déjà fait ces modifications du schéma `taxonomie` depuis `GeoNature` (https://github.com/PnEcrins/GeoNature/blob/master/data/update_1.8.2to1.8.3.sql#L209-L225), exécutez le script SQL de mise à jour de la BDD `data/updates/1.1.1to1.1.2.sql`.

- Si vous ne l'avez pas fait côté GeoNature, vous pouvez supprimer l'attribut `Saisie` après avoir récupéré les informations dans la nouvelle liste avec ces lignes de SQL : https://github.com/PnEcrins/GeoNature/blob/master/data/update_1.8.2to1.8.3.sql#L307-L314
- Rajoutez le paramètre `COOKIE_AUTORENEW = True` dans le fichier `config.py`.

5.10 1.1.1 (2016-12-14)

Nouveautés

- Fixation et livraison des librairies suite à l'arrivée d'AngularJS1.6 (suppression du gestionnaire de dépendances `bower`)
- Mise à disposition des listes rouges (non encore utilisé dans l'application)

Note de version

- Exécutez la procédure standard de mise à jour de l'application (<http://taxhub.readthedocs.io/fr/latest/installation.html#mise-a-jour-de-l-application>)
- Mettre à jour la base de données
- Exécuter la commande suivante depuis la racine du projet TaxHub

```
unzip data/inpn/LR_FRANCE.zip -d /tmp
```
- Exécuter le fichier `data/update1.1.0to1.1.1.sql`

5.11 1.1.0 (2016-11-17)

Nouveautés

- Bugfix
- Ajout d'un titre à l'application
- Gestion des valeurs `null` et des chaînes vides
- Correction de l'installation
- Correction de l'effacement du type de média dans le tableau après enregistrement
- Ajout d'une clé étrangère manquante à la création de la base de données
- Ajout des listes rouges INPN (en BDD uniquement pour le moment)
- Compléments sur les attributs des taxons exemples
- Ajout d'une confirmation avant la suppression d'un média
- Champ `auteur` affiché au lieu du champ `description` dans le tableau des médias
- Modification du type de données pour l'attribut `milieu`
- Possibilité de choisir pour l'installation du schéma `utilisateurs` - en local ou en Foreign Data Wrapper
- Meilleure articulation et cohérence avec UsersHub, GeoNature et GeoNature-atlas
- Amélioration en vue d'une installation simplifiée

5.12 1.0.0 (2016-09-06)

Première version fonctionnelle et déployable de Taxhub (Python Flask)

Fonctionnalités

- Visualisation de `taxref`
- Gestion du catalogue de noms d'une structure
- Association de données attributaires aux taxons d'une structure
- Association de médias aux taxons d'une structure

5.13 0.1.0 (2016-05-12)

Première version de TaxHub, développée avec le framework PHP Symfony

Permet de lister le contenu de TaxRef, le contenu de `taxonomie.bib_taxons`, de faire des recherches, d'ajouter un taxon à `taxonomie.bib_taxons` depuis TaxRef et d'y renseigner ses propres attributs.

L'ajout d'un taxon dans des listes n'est pas encore développé.

Le MCD a été revu pour se baser sur `taxonomie.bib_attributs` et non plus sur les filtres de `bib_taxons` mais il reste encore à revoir le MCD pour ne pas pouvoir renseigner différemment les attributs d'un même taxon de référence - <https://github.com/PnX-SI/TaxHub/issues/71>

A suivre : Remplacement du framework Symfony (PHP) par Flask (Python) - <https://github.com/PnX-SI/TaxHub/issues/70>

5.14 0.0.1 (2015-04-01)

— Création du projet et de la documentation