

# Manuel d'installation de e-colle

e-colle est un gestionnaire de colles destiné aux enseignants de CPGE, à leurs élèves, leurs colleurs ainsi qu'à l'administration pour le décompte des heures effectuées. Son installation ne nécessite que des logiciels libres.

Toute l'installation et la configuration est décrite ici sur un système d'exploitation de type microsoft Windows(testé précisément sur Windows 10).

On donne à la fin les liens vers les documentations des différents logiciels utilisés.

## Table des matières

1.Prérequis logiciels.....	1
1.Python version >= 3.4.....	2
2.Apache2.....	3
3.Le SGBD (Système de Gestion de Bases de Données).....	5
2.Configuration de e-colle.....	6
1.Copie des fichiers et configuration de settings.py.....	6
2.Création de la base de données.....	7
3.Configuration de la base de données.....	8
4.Configuration du site dans Apache2.....	8
3.Personnalisation de e-colle.....	11
4.Exécution en local (développement).....	12
5.Sauvegardes de la base de données.....	12
6.Quelques liens utiles.....	12

## 1. Installation d'un serveur Apache/python3/django

Depuis quelques années, il est devenu très facile d'installer un serveur python3/django/apache sous windows. Bitnami propose un tel serveur clé en main (et gratuit) avec possibilité d'utiliser MySQL ou postgresQL comme gestionnaire de base de données.Il semble qu'il est régulièrement mis à jour avec les version stables les plus récentes des différents logiciels. On renvoie donc à la page :

<https://bitnami.com/stack/django/installer>

et à leur documentation pour la configuration du site.

## 2. Configuration de e-colle

Les étape décrites par la suite sont à faire dans l'ordre où elle sont présentées.

### 1. Copie des fichiers et configuration de settings.py

Décompressez l'archive e-colle dans le répertoire C:\Apache24\htdocs (et

renommer e-colle le répertoire)

Dans la suite on notera CheminVersEColle le chemin absolu vers l'intérieur du répertoire e-colle.

Il faut configurer le fichier settings.py qui se trouve à CheminVersEColle\ecolle\settings.py

Éditez le fichier avec votre éditeur de texte favori. Il faut alors préciser les points suivants :

- DEFAULT\_ADMIN\_PASSWD (le mot de passe par défaut à la création de l'utilisateur Admin).
- DEFAULT\_SECRETARIAT\_PASSWD (le mot de passe par défaut à la création de l'utilisateur Secrétaire).
- EMAIL\_ADMIN (l'email de l'admin. On peut laisser vide)
- EMAIL\_SECRETARIAT (l'email du secrétaire. On peut laisser vide)
- IP\_FILTRE\_ADMIN. Mettre True si on veut que la partie admin ne soit accessible que depuis certaines adresses IP pour des raisons de sécurité, False sinon.
- IP\_FILTRE\_ADRESSES. Dans le cas où IP\_FILTRE\_ADMIN vaut True, il faut préciser dans un tuple les adresses IP autorisées sous forme d'expressions régulières. Par exemple '^127\.\0\.\0\.\1\$' pour l'adresse 127.0.0.1, ou encore '^192\.\168\.\d{1-3}\.\d{1-3}\$' pour toutes les adresses IP de la forme 192.168.xxx.xxx
- IMAGEMAGICK. False (fonctionnalité non compatible avec Windows)
- ADMINS. Un tuple de couples (nom,email) des administrateurs du site.
- ALLOWED\_HOSTS. La liste des noms de domaine autorisés pour e-colle. Démarrer par un point pour autoriser les sous-domaines. Par exemple '.e-colle.fr' autorisera tous les noms de domaine du type 'xxxxxxxxxx.e-colle.fr'
- SECRET\_KEY. Une clé secrète de 50 caractères ASCII. À modifier.
- DATABASES. Il faut y préciser plusieurs champs
  - ENGINE : le logiciel de SGBD utilisé ('django.db.backends.mysql' pour MySQL, 'django.db.backends.postgresql' pour PostgreSQL, 'django.db.backends.sqlite3' pour SQLite)
  - NAME : le nom de la base de données ('e-colle' est mis par défaut)
  - USER : l'utilisateur ayant les droits de la base de données précisée par

NAME. ('e-colle' est mis par défaut)

- PASSWORD : le mot de passe de l'utilisateur précisé juste avant.
- HOST : l'adresse IP du serveur de BDD. 'localhost' si c'est en local.

Enfin, si les crédits ECTS sont activés et que l'on veut incruster dans les pdf directement le tampon/signature du proviseur/proviseur adjoint, il suffit de mettre une image (carrée) proviseur.jpg ou proviseur.png et/ou proviseuradjoint.jpg ou proviseuradjoint.png dans le répertoire resources.

## **2. Création de la base de données**

- MySQL

Si le SGBD que vous avez choisi est MySQL, le plus simple est de créer l'utilisateur e-colle et la base de donnée e-colle avec l'interface graphique fournie par PHPMyAdmin. Dans votre navigateur internet, allez à localhost/phpmyadmin/, connectez-vous avec le login root et le mot de passe défini lors de l'installation.

Cliquez sur l'onglet User, puis en bas cliquez sur add User.

Entrez e-colle comme nom d'utilisateur et cliquez sur le générateur de mot de passe pour avoir un mot de passe robuste. Copiez ce mot de passe dans le champs PASSWORD de DATABASES dans settings.py.

Cochez la case « Create database with same name and grant all privileges » puis validez (Go tout en bas à droite)

Cliquez ensuite sur la base de donnée e-colle apparue dans la liste de gauche, puis onglet Operations. En bas à droite, partie Collation, sélectionnez dans la liste déroulante utf8\_general\_ci (Il s'agit de l'encodage utilisé pour ordonner / comparer les chaînes de caractères, on choisit donc utf8, insensible à la casse), puis validez.

Votre base de données est prête.

- PostgreSQL

Si le SGBD que vous avez choisi est PostgreSQL, ouvrez PgAdmin3, créer une base de données e-colle (encodage UTF8), puis créer un utilisateur e-colle auquel vous donnez tous les droits sur la base e-colle. Le mot de passe attribué à l'utilisateur e-colle doit être copié dans le champs PASSWORD de DATABASES dans settings.py. Votre base de données est alors prête.

Il est important de noter que dans le cas de PostgreSQL, il peut être nécessaire d'effectuer régulièrement des VACUUM (une sorte de défragmentation) sur certaines tables, surtout celles qui subissent beaucoup d'effaçages/modifications.

- SQLite

Si le SGBD que vous avez choisi est SQLite, il n'y a rien à faire, puisqu'il n'y a aucune authentification nécessaire pour accéder à la base de données. (La sécurité est donc moins bonne qu'avec MySQL ou PostgreSQL). Il faut d'ailleurs laisser les champs USER, PASSWORD et HOST vides dans DATABASES du fichier settings.py, et mettre le chemin vers le fichier d'enregistrement dans NAME.

### 3. Configuration de la base de données

Dans un terminal, allez dans le répertoire e-colle, puis lancez la commande :

```
python3 manage.py migrate
```

Si jamais vous faites une mise à niveau de e-colle, il faudra d'abord lancer la commande

```
python3 manage.py makemigrations
```

pour mettre à jour la structure de la base de données.

Une fois cette étape finie, toutes les tables de la base de données ont été créées.

### 4. Configuration du site dans Apache2

Il reste à configurer le site dans Apache2.

- Gestion du protocole SSL (conseillé)

Si vous voulez crypter l'envoi et la réception des données (si vous ne le faites pas, les mots de passe vont passer en clair sur internet), il faut utiliser le protocole SSL (https), qui combine un cryptage + une certification du site. Il existe désormais un moyen d'obtenir gratuitement un tel certificat grâce à l'initiative « Let's encrypt ». On renvoie à la documentation disponible sur

<https://letsencrypt.org/>

À noter que l'utilisation des applications mobiles e-colle (iOS et android) est impossible avec des certificats non valides.

- Configuration du virtualhost.

Éditer le fichier e-collewin.conf (ou e-collewinssl.conf si vous utilisez le protocole SSL) présent à la racine du répertoire e-colle, et modifier les directives ServerName et ServerAdmin en conséquence. Modifiez si besoin les différents chemins qui y sont spécifiés (les chemins par défaut supposent que Apache est installé dans le répertoire C:\Apache24). Sauvegardez. Si on accède au serveur via son adresse IP, mettez son adresse IP en ServerName. Sinon mettez un nom de domaine qui pointe vers l'adresse IP de votre routeur/box (voir point suivant :

## Obtention d'un nom de domaine)

Dans tout ce qui suit, il faut probablement adapter les chemins « C:\apache24\conf\.. » aux vrais répertoires utilisés par django-stack de bitnami.

Couper ensuite le fichier e-collewin.conf (ou e-collewinssl.conf), puis le coller dans le répertoire C:\apache24\conf\extra.

Éditer le fichier httpd.conf. Décommenter les lignes :

```
LoadModule authz_host_module modules/mod_authz_host.so
```

```
LoadModule log_config_module modules/mod_log_config.so
```

```
LoadModule socache_shmcb_module modules/mod_socache_shmcb.so
```

```
Include conf/extra/httpd-vhosts.conf
```

Ouvrir ensuite le fichier httpd-vhosts.conf (qui se trouve au chemin indiqué juste avant). Tout effacer et remplacer par

```
<VirtualHost *:80>
ServerAdmin webmaster@localhost
DocumentRoot "C:/Apache24/htdocs"
LogLevel warn
ErrorLog "logs/error.log"
CustomLog "logs/access.log" common
</VirtualHost>
```

- Si vous n'utilisez pas le protocole SSL, écrire en dessous :

```
Include conf/extra/e-collewin.conf
```

pour y inclure le virtualhost du site e-colle.conf

- Si vous utilisez le protocole SSL, il faut éditer le fichier httpd.conf, et décommenter les lignes :

```
LoadModule ssl_module modules/mod_ssl.so
```

```
Include conf/extra/httpd-ssl.conf
```

Ouvrez ensuite le fichier httpd-ssl.conf (au chemin indiqué au-dessus). Effacer tout le contenu du fichier à partir de la ligne (incluse)

```
<VirtualHost _default_:443>
```

et écrire à la fin du fichier

```
Include conf/extra/e-collewinssl.conf
```

pour y inclure le virtualhost de e-colle.

Redémarrer enfin apache pour prendre en compte les changements.

**ATTENTION : Si vous utilisez Apache 2.2, la syntaxe « require all granted » n'est pas valable. Il faut la remplacer par la syntaxe en vigueur pour apache 2.2 (voir la documentation d'apache).**

- Obtention d'un nom de domaine

Si vous disposez d'une adresse IP fixe (établissement scolaire par exemple) vous pouvez vous dispenser d'avoir un nom de domaine et accéder directement à e-colle via l'adresse IP du serveur. Si vous l'installez depuis une adresse IP flottante (type box) un nom de domaine est obligatoire pour pouvoir suivre votre adresse IP. De nombreux sites proposent des noms de domaines (ou sous-domaines) gratuitement, avec des fonctionnalités limitées et un nombre de noms de domaines limité (~3). Par exemple no-ip, dyndns, etc.... Rendez-vous donc sur votre moteur de recherche préféré pour l'obtention d'un nom de domaine.

- Configuration de votre box/routeur

Pour rediriger les requêtes entrantes sur votre box/routeur vers l'ordinateur qui fait serveur, il faut aller dans les paramètres de votre routeur, et dans la partie NAT/PAT, il faut rediriger les requêtes entrantes http (port 80) et https (port 443) vers votre serveur (via son adresse IP ou son nom). Il est également conseillé de configurer le pare-feu pour qu'il ne laisse passer que les requêtes entrantes sur les ports 80 et 443 (sauf si vous en utilisez d'autres)

### 3. Personnalisation de e-colle

Vous pouvez bien entendu personnaliser e-colle bien au-delà des quelques paramètres du fichier settings.py. Par exemple tout le CSS est contenu dans un seul fichier (public\css\style.css), donc vous pouvez facilement modifier les couleurs, les images, les polices, etc .... Vous pouvez aussi, pour les pythoniciens, modifier les vues, rajouter vos propres vues dans les fichiers views.py adéquates pour rajouter des fonctionnalités si le cœur vous en dit (pensez, si vous ajoutez une vue, à ajouter le fichier de template qui va avec et l'url qui va avec, dans le fichier urls.py du répertoire concerné).

### 4. Exécution en local (développement)

À des fins de test, vous pouvez exécuter en local e-colle, en passant la variable DEBUG à True dans settings.py, et en lançant la commande

```
python manage.py runserver
```

dans le répertoire e-colle. E-colle sera alors accessible à l'adresse localhost:8000. Attention ce serveur ne doit jamais être utilisé autrement qu'en local, de même que la variable DEBUG ne doit valoir True qu'en local.

## 5. Sauvegardes de la base de données

Il est important de faire des sauvegardes régulières de la base de donnée.

Depuis la dernière version des commandes à exécuter dans les console ont été ajoutées :

- « backup ». Si vous exécutez depuis le répertoire e-colle la commande

```
python3 manage.py backup
```

cela crée une sauvegarde de la base de données dans le répertoire /backup, dont le nom correspond à la date de la sauvegarde.

Si on ajoute l'option -m ou -media :

```
python3 manage.py backup -m
```

on sauvegarde en plus les fichiers media (photos/programmes de colle,etc..).

À utiliser régulièrement pour avoir des points de sauvegarde des données en cas de problème.

- « restore ». Si vous exécutez depuis le répertoire e-colle la commande

```
python3 manage.py restore
```

On vous propose de réinitialiser la base de données/les fichiers media à une sauvegarde antérieure effectuée avec la commande backup.

## 6. Commande console

Voici les commandes que l'on peut exécuter depuis la console dans le répertoire e-colle :

- ```
python3 manage.py setdebugtrue
```

pour passer la variable DEBUG à True

- ```
python3 manage.py setdebugfalse
```

pour passer la variable DEBUG à False

- ```
python3 manage.py backup
```

pour faire une sauvegarde de la base de données

- `python3 manage.py backup -m`

pour faire une sauvegarde de la base de données et des fichiers media

- `python3 manage.py restore`

pour réinitialiser le base de données/les fichiers media à une sauvegarde antérieure effectuée avec la commande backup.

- `python3 manage.py initdata`

pour initialiser les données de e-colle

- `python3 manage.py nouvelle_annee`

Pour « nettoyer » la base de données entre 2 années, vous pouvez utiliser la commande. Cela vide certaines tables de la base de données ainsi que les fichiers media programme et image. Les données concernant les colleurs, élèves, classes, matières, sont bien entendu conservées.

## 7. Quelques liens utiles

- [Lien vers la documentation de Django](#)
- [lien vers la documentation de MySQL](#)
- [lien vers la documentation de PostgreSQL](#)
- [lien vers la documentation de SQLite](#)
- [lien vers la documentation de Apache 2](#)