

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.....	9
3.Création de l'admin et du secrétariat.....	11
4.Personnalisation de e-colle.....	12
5.Exécution en local (développement).....	12
6.Sauvegardes de la base de données.....	12
7.Quelques liens utiles.....	12

1. Prérequis logiciels

On dénombre ici la liste des logiciels nécessaires à l'installation de e-colle.

Il semble, d'après la documentation de mod_wsgi pour windows, que la version d'apache et de mod_wsgi doivent être compilées avec la même version de visual C/C++, sans quoi rien ne garanti que cela fonctionne.

Il faut quoi qu'il arrive commencer par installer Visual C/C++. Si vous êtes sur un OS 64 bits, par sécurité télécharger les 2 version (32 et 64 bits). À noter que si votre OS est trop ancien (windows XP ou windows server 2003 ou plus ancien), la dernière version de Visual C/C++ disponible est la version 10.

- Pour la version 14 (2015) :

<https://www.microsoft.com/fr-fr/download/details.aspx?id=48145>

- Pour la version 10 (2010) :

<https://www.microsoft.com/fr-fr/download/details.aspx?id=14632>

<https://www.microsoft.com/fr-fr/download/details.aspx?id=5555>

1. Python version ≥ 3.4

Télécharger Python 3.5 (ou 3.4 si votre OS est trop ancien) à l'adresse suivante. :

<https://www.python.org/downloads/release/python-352/>

Windows x86-64 executable installer sur un OS 64 bits ou

Windows x86 executable installer sur un OS 32 bits

Une fois téléchargé, lancer le fichier et choisir « custom installation ». Vous devez alors sélectionner (en plus de ce qui l'est déjà) :

- for all users (sinon Apache n'y aura pas accès et mod_wsgi ne fonctionnera pas)
- add Python to path (pour que l'on puisse exécuter Python depuis n'importe quel répertoire)

On peut maintenant installer les bibliothèques python qu'il nous faut. Pour cela, on ouvre un terminal (touche windows puis taper cmd). Dans le terminal, taper

```
pip install django reportlab pillow unicodecode
```

si vous voulez installer une version particulière de django (par exemple la 1.9), taper `django==1.9` au lieu de `django` dans la ligne précédente. La dernière version stable de django est la 1.10 et e-colle fonctionne avec django 1.9 et 1.10.

Cela va donc installer les bibliothèques suivantes :

- django version ≥ 1.9 (pour faire des applications web dynamiques)
- reportlab (pour créer des fichiers pdf)
- pillow (pour gérer les images)
- unicodecode (pour transformer en ASCII des caractères unicode)

Si le logiciel de SGBD que vous choisissez est PostgreSQL, il faut installer psycopg2 (qui fait le lien entre Python et PostgreSQL) :

```
pip install psycopg2
```

Il faut installer d'autres bibliothèques, mais que l'on va devoir télécharger. Aller à l'adresse : <http://www.lfd.uci.edu/~gohlke/pythonlibs/> puis télécharger (à ajuster selon que votre OS est 64 ou 32 bits, ici on suppose 64 bits)

- mod_wsgi-4.4.23+ap24vc14-cp35-cp35m-win_amd64.whl (pour qu'apache

comprene le python)

ATTENTION : ici mod_wsgi est compilé avec Visual C/C++ version 14 (2015), c'est indiqué par le « vc14 » dans le nom du fichier. Il faudra donc installer une version d'apache compilée avec Visual C/C++ version 14 (apache 2.4).

Si vous avez une version d'apache compilée avec visual C/C++ version 10 et que vous ne pouvez pas la changer (par exemple parce que votre OS est trop ancien ou que certains sites tournent déjà sur apache), il faudra prendre la version de mod_wsgi compilée avec visual C/C++ version 10, à savoir

mod_wsgi-4.4.23+ap24vc10-cp34-cp34m-win_amd64.whl

mais cette version est prévue pour python 3.4, donc il faudra bien sûr installer Python 3.4 au lieu de Python 3.5 (python 3.4 est aussi compatible avec e-coll et django ≥ 1.9). Dans ce cas, il faut aussi ajuster le téléchargement des fichiers qui suivent, puisqu'il faudra prendre les versions avec cp34 au lieu de cp35.

- lxml-3.6.4-cp35-cp35m-win_amd64.whl (pour parser le xml)

Si vous choisissez MySQL comme logiciel de SGBD, il faut télécharger

- mysqlclient-1.3.7-cp35-cp35m-win_amd64.whl

pour faire le lien entre Python et MySQL.

Une fois ces fichiers téléchargés, toujours à partir du terminal, rendez-vous dans le répertoire où ils ont été téléchargés, puis lancer la commande

```
pip install fichier1 fichier2 ...
```

où fichier1 fichier2 et ... sont à remplacer par les différents fichier whl téléchargés.

2. Apache2

Apache2 est un logiciel permettant à l'ordinateur de jouer le rôle de serveur. Il est tout-à-fait possible d'utiliser un autre logiciel, mais on décrit ici uniquement le cas du logiciel apache2, qui est le plus utilisé de tous, et qui est libre.

Allez à l'adresse suivante : <http://www.apachelounge.com/download/>

puis télécharger httpd-2.4.23-win64-VC14.zip (ou toute autre version en fonction de la version d'apache et de Visual C/C++ que vous voulez).

Suivez les instructions du fichier ReadMe.txt pour installer apache (avec la version indiquée, il suffit de dézipper dans le répertoire [C:\Apache24](#)).

Depuis un terminal, dans le répertoire C:\Apache24\bin taper

- httpd

pour lancer simplement apache, ou bien

- httpd -k install

pour le lancer en tant que service (et donc qu'il se lance au démarrage)

Vous pouvez aussi lancer apachemonitor qui affiche dans la barre des tâches le statut d'apache ainsi que la possibilité de l'arrêter et de le -re-démarrer. Il suffit de taper ApacheMonitor toujours depuis le répertoire C:\Apache24\bin (ou de double-cliquer dessus).

Si vous voulez profiter d'une interface type phpMyadmin (pour MySQL) ou Phppgadmin (pour PostgreSQL), il faut aussi installer et configurer php. Ce n'est pas indispensable mais cela facilite la gestion de la base de données.

Pour installer PHP, aller à l'adresse <http://windows.php.net/download/> puis télécharger la version « thread safe » adaptée à votre OS (x64 pour 64 bits et x86 pour 32 bits) et tout dézipper dans le répertoire C:\php.

Pour configurer PHP, il faut commencer par enregistrer le fichier php.ini-production sous le nom php.ini. (Il se trouve à la racine du répertoire C:\php).

Il faut ensuite éditer le fichier httpd.conf de apache, qui se trouve dans le répertoire C:\Apache24\conf, et ajouter les lignes :

```
LoadModule php7_module "c:/php/php7apache2_4.dll"
```

```
AddHandler application/x-httpd-php .php
```

```
PHPIniDir "C:/php"
```

à la suite de toutes les lignes commençant par « LoadModule ». Cela permet de charger le module php, et fait en sorte que tout fichier d'extension .php soit considéré comme un fichier contenant du code php.

Cherchez également dans le fichier httpd.conf la ligne contenant

```
DirectoryIndex index.html
```

et remplacez-la par

```
DirectoryIndex index.php index.html
```

pour que apache cherche le fichier index.php ou index.html comme fichier à lire par défaut dans un répertoire.

Enfin, il faut activer le module wsgi pour pouvoir interpréter le python. Pour cela, chercher le fichier mod_wsgi.so qui a été créé lorsqu'on a installé mod_wsgi. Il doit se trouver dans un répertoire du type C:\Program Files\Python35 (si vous ne l'y trouvez pas lancez une recherche sur toute votre partition).

Il faut ensuite copier ce fichier et le coller dans le répertoire où se trouvent les modules d'apache : C:\Apache24\modules.

Enfin, il faut éditer une nouvelle fois le fichier httpd.conf et à la fin de la liste des « LoadModule » ajouter la ligne :

```
LoadModule wsgi_module modules/mod_wsgi.so
```

3. Le SGBD (Système de Gestion de Bases de Données)

Les SGBD qui s'utilisent naturellement (et se configurent simplement) avec django sont SQLite (libre), MySQL (libre), PostgreSQL(libre), Oracle (payant, avec une version gratuite allégée et limitée à 1 BDD). On peut en utiliser d'autres mais la compatibilité avec django n'est pas garantie.

- SQLite est le plus léger (1 base de données = 1 fichier), mais aussi le moins performant et le moins sécurisé. Pour l'installer, suivre les instruction sur la page <http://www.sqlite.org/download.html>
- MySQL est un SGBD libre très utilisé pour les site web dynamiques relativement simples. Pour l'installer, il faut le télécharger sur la page officielle : <https://dev.mysql.com/downloads/windows/installer/5.7.html> (L'installateur est 32 bits, mais il va installer la version 64 bits si votre OS est un 64 bits), puis de lancer l'installation.

Pendant l'installation il vous faudra préciser le mot de passe de l'utilisateur root, qui vous permettra par la suite de vous connecter à MySQL.

Vous pouvez également installer PHPMyAdmin si vous voulez une interface graphique pour gérer vos bases de données MySQL. Pour cela, il faut le télécharger à la page <https://www.phpmyadmin.net/downloads/> puis le dézipper dans le répertoire c:\Apache24\htdocs (qui est le répertoire où l'on stocke les fichiers des sites web). On pourra renommer le répertoire en un nom plus sobre, comme phpmyadmin (plutôt que phpMyAdmin-4.6.4-all-languages).

Vous pouvez alors y accéder en allant à l'adresse «localhost/phpmyadmin» .

Il est alors assez simple et intuitif de créer et gérer des utilisateurs/bases de données.

- PostgreSQL est le SGBD le plus performant parmi les SGBD libres, même si un peu plus compliqué à gérer/configurer que MySQL. C'est probablement

le meilleur choix, parmi tous ceux possibles.

Pour l'installer, télécharger l'installateur adapté à votre OS (64 ou 32 bits) d'une version récente (au moins 9.5) disponible à cette adresse : <http://www.enterprisedb.com/products-services-training/pgdownload#windows>

À l'installation sera créé un super-utilisateur «postgres», pour lequel vous devrez spécifier un mot de passe (qu'il faudra retenir).

Par la suite, pour plus de commodité de gestion, vous pouvez utiliser PGAdmin3 qui s'est installé avec PostgreSQL (aller dans applications > PostgreSQL> PgAdmin3)

- Si vous voulez utiliser un autre SGBD (Oracle ou autre), toutes les infos sont disponibles sur le site de django :

<https://docs.djangoproject.com/en/1.9/ref/databases/#using-a-3rd-party-database-backend>

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 secrétariat. 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 adresse 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 cliquer 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 :

```
python manage.py makemigrations accueil
```

Tapez ensuite :

```
python manage.py migrate
```

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. On va auto-certifier le site (sinon il faut payer un organisme extérieur pour le certifier), donc les navigateurs internet afficheront une erreur de certificat en arrivant dessus.

On commence par activer le mod ssl de apache2. Pour cela éditer le fichier httpd.conf et décommenter la ligne

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

Il faut ensuite définir la variable d'environnement **OPENSSL_CONF** qui, par défaut est configurée pour un environnement de type GNU/Linux. Pour cela, dans un terminal, taper :

```
set OPENSSL_CONF=C:\Apache24\conf\openssl.cnf
```

(ou tout autre chemin correspondant à la location de votre fichier openssl.conf)

Il faut ensuite se rendre dans le répertoire C:\Apache24\bin puis lancer dans le terminal la commande

```
openssl req -new -out server.csr
```

Vous devez alors répondre à plusieurs questions (Des exemples parlants sont précisés à titre indicatif, à vous d'adapter) :

- PEM pass phrase (un mot de passe à retenir)
- Country Name : FR
- State : France
- Locality Name : Cayenne
- Organisation Name : Ma Prepa
- Organisation Unit Name : Ma Prepa
- Common Name : e-colle.mondomaine.org (nom de domaine)
- Email Address : monadresse@monsite.com (email de l'Admin)

taper ensuite, toujours depuis le répertoire C:\Apache24\bin

```
openssl rsa -in privkey.pem -out server.key
```

et enfin

```
openssl x509 -in server.csr -out server.cert -req -signkey  
server.key -days 365
```

où -days 365 indique que le certificat est valable un an.

Il faut enfin couper et coller les fichiers server.cert et server.key (qui se trouvent dans le répertoire C:\Apache24\bin) dans le répertoire C:\Apache24\conf

- 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](#))

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 (include)

```
<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. Création de l'admin et du secrétariat

Allez sur le site via son nom de domaine, rajoutez /init dans la barre d'adresse à la fin du nom de domaine puis validez. Les utilisateurs Admin et Secrétariat sont alors créés avec les valeurs par défaut du fichier settings.py.

Il est recommandé (par soucis de sécurité) d'aller désactiver ensuite cette adresse en éditant le fichier `ecolle\urls.py` et commenter la ligne 7 puis sauvegarder :

```
url(r'^init$', 'accueil.views.init',name="init")
```

4. 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).

5. 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.

6. Sauvegardes de la base de données

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

Django permet de faire cela de manière transparente vis-à-vis du SGBD, à l'aide de la commande `dumpdata`. La commande

```
python manage.py dumpdata > ecolle.json
```

exécutée dans le répertoire e-colle crée le fichier `ecolle.json`, qui contient toutes les données de la base de données. Pour les recharger dans la base, il suffit de lancer la commande

```
python manage.py loaddata ecolle.json
```

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](#)
- [lien vers la documentation de Mathjax](#)