














# Documentation technique

---

## Structure du projet

---

 .idea	03/05/2022 19:46	Dossier de fichiers	
 bin	09/05/2022 14:27	Dossier de fichiers	
 configuration	16/05/2022 12:31	Dossier de fichiers	
 data	31/05/2022 14:41	Dossier de fichiers	
 out	18/05/2022 15:39	Dossier de fichiers	
 projet	03/05/2022 18:58	Dossier de fichiers	
 python_classes	30/05/2022 12:39	Dossier de fichiers	
 static	03/05/2022 18:58	Dossier de fichiers	
 templates	03/05/2022 18:58	Dossier de fichiers	
 db.sqlite3	03/05/2022 18:58	Fichier SQLITE3	0 Ko
 manage.py	03/05/2022 18:58	Python File	1 Ko
 ModeEmploi.pdf	30/05/2022 15:43	Chrome HTML Do...	253 Ko
 requirements.txt	30/05/2022 14:02	Document texte	3 Ko

bin : Dossier qui contient l'ensemble des dépendances qui sont nécessaires à graphviz

configuration : Contient tous les fichiers python qui permettent de faire fonctionner l'application (routes, définition des templates, etc)

data : Dossier qui contient les fichiers d'entrée (cf. mode d'emploi)

out : Dossier qui contient les fichiers JSON qui permettent de faire fonctionner l'API

projet : Dossier qui contient tous les fichiers python qui permettent de paramétrer le projet django

python\_classes : Dossier qui contient l'ensemble des classes python utiles pour la logique de l'application

static : Dossier qui contient les éléments auxquels le template fait appel (images, js, css)

templates : Dossier qui contient les éléments HTML de l'application (templates)

manage.py : Fichier python qui permet d'exécuter toutes les commandes relatives à django, c'est grâce à ce fichier que l'on peut lancer le serveur HTTP (cf. mode d'emploi)

Les autres dossiers et documents n'interviennent pas dans l'application

## Configuration/urls.py

---

```
from django.urls import *
from . import views

urlpatterns = [
    path('', views.home),
    path('api/resumes_circulaires/<str:id_variable>/<str:unite>', views.get_resumes),
    path('api/moyenne_unite/<str:id_variable>/<str:unite>', views.get_moyennes_unite)
]
```

urls.py contient la définition des urls de l'application. La fonction `path()` prend en paramètres le chemin sous forme de string et la fonction qu'il faut exécuter quand le chemin est saisi par l'utilisateur.

Par exemple, quand l'utilisateur est sur la racine du site, c'est la fonction `home` située dans le fichiers `views` qui sera exécutée.

C'est dans ce fichier que sont définies les routes relatives à l'API.

## Configuration/utls.py

---

utls.py contient toutes les fonctions « utilitaires ». Ces dernières sont toutes documentées dans le code.

## Configuration/views.py

---

views.py est le fichier qui permet de faire le lien entre la logique du site et l'interface. Toutes les fonctions sont documentées dans le code.

Les autres fichiers du dossier Configuration sont générés automatiquement par Django, ils sont actuellement dans leur état d'origine car je n'ai pas eu besoin de les modifier.

## projet/settings.py

---

settings.py contient les paramètres généraux de Django, je ne détaillerai ici que les variables que j'ai modifié.

```
58  ▼ TEMPLATES = [  
59  ▼      {  
60      'BACKEND': 'django.template.backends.django.DjangoTemplates',  
61      'DIRS': [os.path.join(BASE_DIR, 'templates')],  
62      'APP_DIRS': True,  
63  ▼      'OPTIONS': {  
64  ▼          'context_processors': [  
65              'django.template.context_processors.debug',  
66              'django.template.context_processors.request',  
67              'django.contrib.auth.context_processors.auth',  
68              'django.contrib.messages.context_processors.messages',  
69          ],  
70      },  
71  },  
72  ]
```

La clé contient le chemin vers le dossier des templates, la variable `BASE_DIR` vaut « / » `os.path.join(BASE_DIR, 'templates')` vaut donc « /templates »

```
119  # Static files (CSS, JavaScript, Images)  
120  # https://docs.djangoproject.com/en/4.0/howto/static-files/  
121  
122  STATIC_URL = 'static/'  
123  STATICFILES_DIRS = [os.path.join(BASE_DIR, 'static')]  
124  MEDIA_URL = '/images/'
```

Paramétrage du lien vers le dossier static qui a été détaillé dans la structure du projet.

**POUR TOUTE AUTRE QUESTION ME CONTACTER PAR MAIL OU PAR  
TÉLÉPHONE :**

[ph.favrelle@gmail.com](mailto:ph.favrelle@gmail.com)

06.24.88.85.91