

---

# PLD SMART 4IF

## Document d'installation

---

Hexanôme 4213

Lucas POISSE, Cyril POTTIEZ, Julien EMMANUEL, Manuel AMOUROUX,

Pascal CHIU, Danh Lap NGUYEN, Timothée DURAND



SMART AFFLUENCE  
H4213

# Table des matières

<b>Table des matières</b>	<b>2</b>
<b>Introduction</b>	<b>2</b>
<b>Mise en place des serveurs</b>	<b>2</b>
Mise en place du site Web (hébergement)	2
Mise en place des interpréteurs (Workers)	3
Mise en place des analyseurs (prédiction)	4
<b>Mise en place de la base de données</b>	<b>4</b>

## Introduction

Ce document a vocation à détailler le processus d'installation du logiciel AFFLUATIF, produit dans le cadre du PLD SMART de 4ème année. Les différentes étapes de mise en oeuvre au niveau technique et applicatif y sont détaillées.

## Mise en place des serveurs

Cette partie a pour but de détailler la mise en oeuvre des nombreux serveurs utilisés dans le cadre de la mise en oeuvre du service affluatif.

### Mise en place du serveur rabbitMQ

Le serveur RabbitMq peut être configuré simplement en suivant les instructions sur cette page web: <https://tecadmin.net/install-rabbitmq-server-on-ubuntu/>

Nous avons également utilisé un fichier de configuration pour changer le port et activer le *garbage collect*. Ce fichier est disponible sur le git dans le dossier `/devops/rabbitmq` du dossier `/analyseur`.

### Mise en place du site Web (hébergement)

La mise en place du site web nécessite un serveur web :

La configuration recommandée est un serveur LAMP (linux, apache, mysql, php), dont la procédure d'installation est disponible ici :

<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-16-04>

Le site nécessite impérativement PHP de version  $\geq 7$ , une base de données MySQL, Composer (installé globalement ou téléchargé localement) et RabbitMQ.

L'intégralité des fichiers nécessaires pour déployer le site web est sur le dépôt git suivant :

<https://github.com/Webcretaire/AffluatifH4213/tree/master/SiteWeb>

Pour installer en développement, il suffit de lancer sur le serveur les commandes suivantes :

```
git clone <lien_du_git>

composer install
```

Pour déployer en production :

```
git clone <lien_du_git>

composer install --no-dev --optimize-autoloader

rm -rf Resources_Do_Not_Deploy
```

Les fichiers de configuration Apache (virtualhost) se trouvent sur le dépôt git, dans le dossier Resources\_Do\_Not\_Deploy/Apache.

[https://github.com/Webcretaire/AffluatifH4213/tree/master/SiteWeb/Resources\\_Do\\_Not\\_Deploy/Apache](https://github.com/Webcretaire/AffluatifH4213/tree/master/SiteWeb/Resources_Do_Not_Deploy/Apache)

Ces fichiers doivent être déployés dans le dossier /etc/apache2/sites-available et activés à l'aide des commandes :

```
a2ensite rabbitmq.conf

a2ensite smart.conf

a2ensite smart-le-ssl.conf

service apache2 restart
```

Une fois le virtualhost correctement configuré, utiliser certbot pour obtenir un certificat SSL.

## Mise en place des interpréteurs (Workers)

Le logiciel d'interprétation des flux vidéo est responsable du comptage des affluences et des notifications mail en cas d'alerte programmée par l'utilisateur. Ce logiciel est écrit en python, et peut tourner sur n'importe quelle machine disposant des technologies suivantes :

- Python 3.6, environnement Anaconda conseillé (à télécharger sur <https://docs.anaconda.com/anaconda/install/windows>). En cas d'utilisation d'anaconda, créer un environnement virtuel ainsi : `conda create`

`--no-default-packages -n <name> pip python=3.6` et l'activer via la commande `activate <name>`.

- Gestionnaire de paquets PIP, version 10 minimum
- API Google TensorFlow (pour CPU ou GPU). Pour la version GPU, il faut disposer d'une carte graphique NVIDIA possédant une compute capability  $\geq 3.0$ , de l'API CUDA 9.0 de NVIDIA et des outils CUDNN mis à disposition par NVIDIA (version 7.0 pour CUDA 9.0)

Il faut également suivre les instructions d'installation de l'API MS COCO utilisée pour la reconnaissance de formes :

[https://github.com/matterport/Mask\\_RCNN#user-content-installation](https://github.com/matterport/Mask_RCNN#user-content-installation)

Une fois tous les outils installés, l'interpréteur peut à présent être lancé en exécutant le script `python receiveStream.py` localisé dans le répertoire `rabbitmq\interpreter_queue` du projet.

Le projet dispose également d'un fichier .INI de configuration permettant de spécifier diverses options (nombre maximal de flux différents à traiter pour l'interpréteur, fréquence de mise à jour des modèles statistiques utilisés par les analyseurs pour chaque flux).

## Mise en place des analyseurs (prédiction)

Il faut disposer d'un serveur (free tier AWS par exemple) et s'y connecter en ssh:

1. installer apache2 avec mod\_wsgi: *sudo apt-get install libapache2-mod-wsgi-py3*
2. cloner le dépôt et se déplacer dans le dossier *analyseur*
3. installer anaconda, créer un environnement : *conda env create -f environment.yml*
4. installer R et les librairies *forecast*, *anytime*, *lubridate* et *plotly*
5. Pour la partie *fitting* du modèle
  - a. lancer une session screen
  - b. se placer dans le dossier */Python*
  - c. activer l'environnement: *source activate analyseur*
  - d. lancer le consommateur RabbitMQ : *python RMQ\_receiver.py*
6. Pour la partie prédiction
  - a. copier les fichiers du dossier */devops/apache2* dans */etc/apache2/sites-available/*
  - b. lancer un script : *bash fix\_perm.sh*
  - c. relancer apache : *sudo service apache2 restart*

## Mise en place de la base de données

Comme vu dans la partie I.A, la base de données il est conseillé, si possible, de déployer la base de données sur le même serveur que le serveur Web utilisé pour l'hébergement du site pour des raisons de simplicité de configuration.

Cette base doit être une base de données utilisant la dernière version de MySQL, nommée “affluatif” et liée à un utilisateur pouvant la consulter et la modifier, bénéficiant de tous les droits (SELECT/INSERT/UPDATE/DELETE + instructions de LDD, DROP, CREATE).

Le script de génération du schéma de la base et des tables se trouve sur le dépôt GIT du site dans le dossier Resources\_Do\_Not\_Deploy :

[https://github.com/Webcretaire/AffluatifH4213/blob/master/SiteWeb/Resources\\_Do\\_Not\\_Deploy/DB\\_Creation\\_Script\\_Affluatif.sql](https://github.com/Webcretaire/AffluatifH4213/blob/master/SiteWeb/Resources_Do_Not_Deploy/DB_Creation_Script_Affluatif.sql)