ThermiScan

# Cahier des charges

## Description du projet

Nous devons réaliser une application en Web qui doit nous permettre d’afficher une vidéo venant du smartphone CAT S60, qui possède une caméra thermique. L’application Web doit permettre de gérer plusieurs vidéos venant de plusieurs caméras, et permettre d’afficher une courbe de température.

## Travail à réaliser

* Réaliser une application en Web qui s'approche un maximum d'un point de vue fonctionnel de l'application FLIR Tools Mobile.
  + Affichage d’un graphique montrant la température maximal, minimal et la moyenne.
  + Affichage de la vidéo.
  + Gestion de plusieurs vidéos venant d’une seule caméra.

## Outils utilisés

* Visual Studio Code
* EasyPHP V 14.1
* MySQL
* Python 3
* Balsamiq Mockups 3

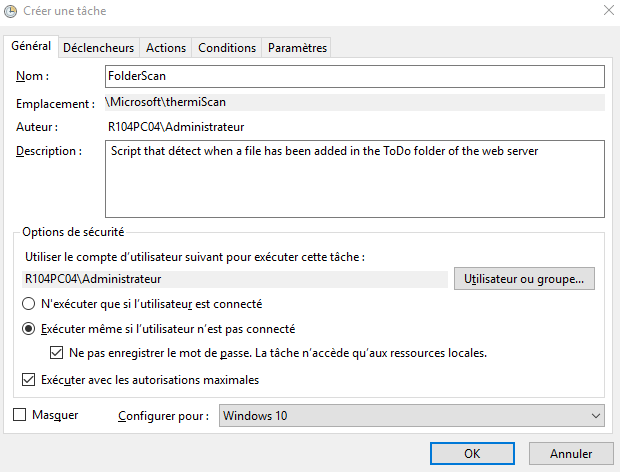
## Comment avons-nous procédé ?

* Réalisation d’une maquette du front-end
* Création d’un exécutable à partir d’un programme python
* Création d’une base de données pour les utilisateurs, les caméras et les vidéos
* Réalisation du front-end avec easyPHP
* Ajout d’un script JS permettant d’afficher un graphique
* Mise au propre du code

### Exécution d’un script Python en background

Pour Pouvoir exécuter le script python en arrière-plan j’ai utilisé le « planificateur de tâches » de Windows

1. Dans le planificateur de tâches, sous Actions j’ai cliqué sur « créer une Tâche »
2. Je l’ai appelé « FolderScan », puis sous Options de sécurité j’ai coché les options comme si dessous



1. Dans l’onglet Actions j’ai ajouté une nouvelle action « Démarrer un programme »
2. Ajouter le lien vers le script « fileDetector.py » sous Programme/script : et cliquez sur OK
3. Pour finir clic droit sur la nouvelle tâche, puis Exécuter

# Réalisations

## Exécution d’un script Python en background

## Création du site web

### Description du code

#### Classe User.php

La classe User.php

#### Classe Pdo.php

La classe Pdo.php permet d’exécuter des requêtes mySQL à partir du PHP, ses

//Crée une connexion à la BDD

    function CreateConnexion() {

        $db = new PDO("mysql:host=" . $this->\_host . ";dbname=" . $this->\_dbname, $this->\_user, $this->\_password);

        $db->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);

        return $db;

    }

### 

# Conclusion

## Bilan

## Problèmes rencontrés

### Téléphone ne permettant pas la diffusion en direct

Le problème majeur rencontré dès le début du projet est l’indisponibilité de live streaming du Cat S60. Selon les pages qu’on parcourait, il était mentionné que la Cat S60 possédait une option de live streaming, ce qui n’est au final pas le cas, contrairement à la nouvelle génération du Cat, le S61. Étant donné que le cahier des charges était de faire un live directement via une application web, nous nous sommes retrouvé à devoir modifier le cahier des charges, qui est passé d’un live sur une application web à juste afficher une vidéo.

### Affichage d’un graphique

Plusieurs problèmes se sont joint lors de la création du graphique. Premièrement, il nous fallait trouver un script JS permettant d’afficher un graphique à partir de valeur venant d’un fichier CSV. Lors de nos premiers tests, le problème principal était l’importation de nos valeurs dans les tableaux, les graphiques ne possédant pas de tableau souple au niveaux des données entrées. Nous avons essayé avec un graphique venant de amcharts.com et un autre venant de highcharts.com. En fin de compte, Nous nous sommes occupé d’en faire un, en reprenant des parties de code provenant des anciens tests.