

## TD5 : Architecture du projet

Compétences

- Création de packages Python
- Définir l'IHM du projet VideoTracker
- Mettre en place le pattern MVC

### IHM — VideoTracker

1. Dans le cahier des charges du projet VideoTracker identifier tous les besoins en rapport avec l'IHM et les transformer en exigences.
2. Dessiner sur une feuille de papier l'IHM de votre application VideoTracker.
3. Téléchargez l'archive `src_student.zip`
4. Créer un répertoire `videoTracker` et décompresser l'archive dans ce répertoire.
5. Puis créer l'arborescence suivante qui permettra d'organiser les fichiers pour le développement du logiciel *VideoTracker*

```
VideoTracker
|
|--- src
|   |--- controllers
|   |--- models
|   |--- views
|--- resources
|   |--- images
|   |--- videos
|--- tests
```

6. Quelles sont les instructions qui permettent de lancer l'application?
7. Étudier la section 6.4 de ce lien <https://docs.python.org/fr/3/tutorial/modules.html> puis ajouter les imports nécessaires pour lancer l'application
8. Que contient la `vue` de notre application?
9. Dans le concept MVC quels sont le(s) rôle(s) des classes `FileRepo` et `Point`. Les placer dans le bon répertoire.
10. Terminer la partie IHM lié au besoin de la sauvegarde des données.

### Lecture d'une vidéo

Pour lire une vidéo nous allons utiliser le package OpenCV.

#### Reading Video using OpenCV

Reading and writing videos in OpenCV is very similar to reading and writing images. A video is nothing but a series of images that are often referred to as frames. So, all you need to do is loop over all the frames in a video sequence, and then process one frame at a time. In this post, we will demonstrate how to read, display and write videos from a file, an image sequence and a webcam. We will also look into some of the errors that might occur in the process, and help understand how to resolve them. Learn OpenCV : <https://learnopencv.com/reading-and-writing-videos-using-opencv/>

1. Télécharger le fichier `play_video.py`. Placer une vidéo de votre dans le même dossier renseigner `path_video` dans le code et lancer la vidéo.
2. Implémenter la (les) tâche(s) affectées à la lecture complète de la vidéo