

Rapport du Projet 1

9 mars 2025

1 Membres de l'équipe et leurs rôles

- **Tarek Radwan** : Développeur principal, responsable de l'implémentation du script Python, gestion des ressources extraites.
- **Liwaa Zebian** : Développeur principal, responsable de la mise en place du processus de téléchargement des ressources (images, vidéos), optimisation de l'extraction et de la gestion des erreurs, création de la page HTML de présentation des ressources.

2 URLs avec images et/ou vidéos

Voici trois pages web contenant des images et/ou des vidéos extraites via notre script, avec les liens vers leurs versions HTML générées :

- **URL 1** : zone.votresite.ca/photo-video-audio/photos-et-videos
Page HTML générée : dirmit.example.com/page1.html
- **URL 2** : www-labs.iro.umontreal.ca
Page HTML générée : dirmit.example.com/page2.html
- **URL 3** : www.example.com/photos
Page HTML générée : dirmit.example.com/page3.html

Les liens ci-dessus sont cliquables et permettent de visualiser les ressources extraites.

3 Langages et bibliothèques utilisés

- Pour coder ce projet, les langages et bibliothèques suivants ont été utilisés :
- **Python** : Langage principal pour l'extraction et la génération de la page HTML.
 - **BeautifulSoup (bs4)** : Bibliothèque pour l'analyse des pages HTML et l'extraction des ressources.

- **urllib** : Gestion des requêtes HTTP et téléchargement des ressources.
- **re** : Expressions régulières pour filtrer les ressources (images/vidéos).
- **argparse** : Gestion des arguments de ligne de commande.
- **os** : Gestion des fichiers et des répertoires.

4 Traitements particuliers réalisés

- **Gestion des fichiers SVG** : Les images SVG ont été extraites sans conversion.
- **Téléchargement et stockage** : Sauvegarde des ressources dans un répertoire défini par l'utilisateur.
- **Génération de la page HTML** : Création automatique d'une page listant les images et vidéos avec affichage en carrousel ou galerie.

5 Instructions pour lancer les scripts

5.1 Lancer le script d'extraction

Utilisez la commande suivante pour extraire les ressources d'une page web et les sauvegarder dans un répertoire local :

```
python script.py extract [URL] -r [regex] -i [no-images] -v [no-videos]
-p [path]
```

5.2 Générer la page HTML

Après l'extraction, utilisez la commande suivante pour générer la page HTML :

```
python script.py genere > tapage.html
```

5.3 Installation des dépendances

Si nécessaire, installez les librairies Python avec :

```
pip install beautifulsoup4 urllib3
```

6 Conclusion

Ce projet a permis d'extraire des ressources multimédia de pages web et de les afficher dans un fichier HTML interactif. L'utilisation de Python et de ses librairies nous a permis d'automatiser l'analyse de pages HTML et la gestion des fichiers téléchargés. Grâce à ce projet, nous avons pu démontrer la capacité de manipuler des ressources web et de générer une page de visualisation facile à consulter.