




Édition 2025

PRÉSENTATION DU PROJET



| | |
|--|--|
| Nom de votre projet | Revers'Art |
| Membre de l'équipe n°1 (prénom/nom) | FAGES Loan |
| Membre de l'équipe n°2 (prénom/nom) | IVANOV Matheo |
| Membre de l'équipe n°3 (prénom/nom) | GACON-LAGAYE Arthur |
| Membre de l'équipe n°4 (prénom/nom) | CHABOT Luka |
| Membre de l'équipe n°5 (prénom/nom) | |
| Niveau d'étude (première ou terminale) | Terminale |
| Établissement scolaire | Lycée Joseph Saverne (L'Isle Jourdain) |
| Responsable du dépôt (professeur de NSI) | BOURDET Nathalie |

1 / PRÉSENTATION GÉNÉRALE

Le projet Revers'Art est une application web qui permet aux utilisateurs d'envoyer une image d'une œuvre d'art . Ensuite, l'application utilise le hash de l'image pour l'identifier et renvoyer des informations sur l'œuvre. Notre projet permet d'obtenir des informations sur une œuvre à partir d'un fichier image (png, jpg, etc...). On a également une version Tkinter téléchargeable sur le site web.

2 / ORGANISATION DU TRAVAIL

Luka - Crawler

Arthur - Application sous la forme Tkinter

Matheo - Site Web + Responsive

Loan - Scraping + Réorganisation du code

On a utilisé une telle répartition en fonction des préférences et qualités de chacun, on avait une idée de chaque étape à faire puis chacun s'est choisi une tâche.

Le projet nous a pris 2 semaines puis on a tous regroupés sur un repository github pour que chacun de nous puisse gérer sa branche.

3 / ÉTAPES DU PROJET

Etape 1 - Trouver un projet et avoir un plan de travail

Etape 2 - Répartir les tâches pour chaque personnes

Etape 3 - Chacun travaille de son côté

Etape 4 - Réunification de chaque code via github

Etape 5 - Création de la base de donnée

Etape 6 - Tester pour trouver des erreurs

Etape 7 - Dernière mise à jour (responsive, esthétique...)

Etape 8 - Photo et présentation du projet

4 / FONCTIONNEMENT ET OPÉRATIONNALITÉ



État d'avancement du projet



Application fonctionnelle



En cours de réalisation

- Recherche par lien
- Seuil sur tkinter



A faire

- Améliorer l'affichage sur tkinter
- Ajouter une page à propos / crédit



Gestion des bugs

Si un développeur trouve un bug, il le signale et le développeur chargé de cette fonctionnalité le résout dans les plus brefs délais.


Comment ça fonctionne ?

Une fois le serveur web lancé, l'utilisateur peut importer une image et définir un seuil. Il permet de régler la tolérance de la recherche d'œuvres similaires (valeurs recommandées : 0 à 20). Un seuil faible recherche des copies très proches, un seuil élevé élargit la recherche.

L'API recherche donc des œuvres d'art avec un hash similaire dans une base de données où elles sont stockées au format json grâce à un crawler recherchant les images sur wikiArt. De plus, pour trouver davantage d'informations un scraper recherche des informations supplémentaires affichées par la suite sur notre site web.

La version Tkinter fonctionne exactement pareil mais n'a pas l'option du seuil qui est prédéfini à 5.

5 / OUVERTURE

 Fonctionnalités et améliorations à venir

 Base de donnée

- Avoir une grande base de donnée
- Ajouter les oeuvres connues

 Language

- Ajuster le rendu des informations entièrement en français
- Créer un bouton pour changer le langage

 Fonctionnalités utilisateur

- Drag and Drop des Images
- Recherche d'oeuvres par lien
- Ajouter une gestion d'erreur plus fine (upload invalide, mauvais format)
- Packager le projet avec setup.py ou en .exe via pyinstaller pour Tkinter

✨ Ajouter une page "À propos" ou crédits

 Analyse critique du projet







✓ Ce qui a bien fonctionné

- Application simple d'utilisation
- Le hash des images pour trouver la bonne oeuvre
- Le scraping des infos de WikiArt

🧐 Si c'était à refaire ?

C'était une très bonne expérience pour une première approche dans un travail de groupe. Poser dès le début les bases et les étapes du projet serait plus évident ainsi que apprendre l'utilisation de GIT pour mettre en commun plus facilement le projet.

 Compétences acquise grâce au trophée NSI

-  Utilisation de Flask pour créer un serveur WEB
-  Scraping d'une page WEB avec BeautifulSoup et Request
-  Utilisation de Tkinter pour les interfaces
-  Plus généralement les langages WEB
-  Travail d'équipe
-  Expérience utilisateur fine et efficace