



Ce document est l'un des livrables à fournir lors du dépôt de votre projet : 4 pages maximum (hors documentation).

Pour accéder à la liste complète des éléments à fournir, consultez la page [Préparer votre participation](#).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ? Contactez-nous à [info@trophees-nsi.fr](mailto:info@trophees-nsi.fr).

---

**NOM DU PROJET : EKOROJI**

## > PRÉSENTATION GÉNÉRALE :

L'origine de ce projet découle du constat que de nombreuses personnes ne savent pas comment trier leurs déchets correctement, ce qui a un impact négatif sur l'environnement. Nous avons donc voulu créer une solution qui permettrait à chacun de devenir un écocitoyen responsable et de contribuer à la protection de notre planète.

Ekoroji est une application web conçue pour simplifier la recherche et la gestion des déchets. Elle est construite en utilisant le framework Flask et l'API OpenAI pour fournir des informations précises et fiables sur les différents types de déchets. Les utilisateurs peuvent rechercher des déchets spécifiques à l'aide d'un formulaire de recherche convivial et trouver des informations détaillées sur leur gestion, leur recyclage, leur réutilisation et leur élimination.

Ekoroji offre également aux utilisateurs la possibilité de contribuer à la base de données en ajoutant des informations sur les déchets. Ils peuvent ajouter, modifier ou supprimer les déchets existants pour maintenir la base de données à jour et pertinente.

En outre, Ekoroji propose des fonctionnalités de filtrage des résultats de recherche pour permettre aux utilisateurs de trouver des informations spécifiques en fonction du nom, de la description, etc. L'application fournit également des informations utiles sur les différentes options de gestion des déchets, telles que le tri et le recyclage.

En somme, Ekoroji est une application web pratique et complète pour la recherche et la gestion des déchets. Elle fournit des informations fiables et précises sur les différents types de déchets, tout en permettant aux utilisateurs de contribuer à la base de données et de trouver des informations spécifiques selon leurs besoins. Ekoroji encourage les utilisateurs à adopter une approche responsable dans la gestion de leurs déchets et à contribuer à la protection de l'environnement.

## > ORGANISATION DU TRAVAIL :

Notre équipe est composée d'une fille et de trois garçons.

**Ethan Facca** a géré l'équipe et s'est occupé de l'automatisation en post-production. Il a également développé en Python et en HTML, tout en automatisant la base de données grâce à l'API OpenAI pour informer automatiquement les utilisateurs sur la manière de recycler les déchets.

**Ambre Florette** a pris en charge l'aspect visuel du site ainsi que le développement en HTML et CSS.

**Benoît Urlacher** a conçu la base de données et l'a intégrée dans le projet. De plus, il a apporté ses connaissances en développement durable en tant qu'éco-délégué.

**Kylian Zhen-Claverie** a aidé au développement en Python, vérifié les erreurs potentielles dans le code et a également contribué à la partie HTML du projet.

Nous avons pu gérer notre temps grâce à nos heures de NSI (6 heures par semaine), qui nous ont permis de répartir le travail en petits groupes voire de travailler individuellement sur certains sujets clés du projet.

Nous avons utilisé plusieurs outils pour mener à bien le projet. GitHub a été utilisé pour gérer le code, Chat GPT et DALL-E pour l'utilisation des API, Figma pour le design de l'application et VS code comme éditeur de code. En ce qui concerne la communication, nous avons utilisé Notion et Instagram.

## > LES ÉTAPES DU PROJET :

Tout d'abord, nous avons réfléchi à un concept unique et créatif axé sur l'environnement en étudiant les besoins et les demandes des utilisateurs. Pour leur offrir la meilleure expérience possible, nous avons recherché une charte graphique et un design élégant en adéquation avec le thème du concept. Nous avons ensuite créé une maquette de l'application en utilisant des outils tels que Figma et Adobe Color.

Après avoir conçu la maquette, nous avons examiné les différents outils qui nous étaient proposés et choisi le langage Python, HTML et SQL pour la mise en œuvre. Cependant, nous avons rencontré des difficultés lors de la conception et avons finalement opté pour un site web plutôt qu'une application car Python n'est pas un langage compilé. Pour faciliter l'utilisation de l'application, nous voulions inclure un système de scan de code-barres, mais cela s'est avéré trop complexe compte tenu du temps imparti. Nous avons donc opté pour une simple recherche textuelle. Pour automatiser la base de données, nous avons envisagé d'utiliser des API et des IA telles que ChatGPT, cela signifiait que l'utilisateur n'avait qu'à saisir le nom du déchet. Cependant, nous avons dû faire confiance aux informations de ChatGPT même si certaines d'entre elles peuvent être fausses. En outre, certaines réponses peuvent être erronées en raison de l'éthique de l'IA.

Nous avons réparti les tâches en fonction des capacités de chacun. L'équipe de développement a travaillé dur pour coder le back-end et le front-end le plus rapidement possible. Au fil du temps, nous avons créé plusieurs maquettes distinctes pour parvenir à la meilleure version possible. Une fois l'application terminée, nous avons réalisé de nombreux tests pour détecter le moindre problème. Ensuite, nous avons réalisé une vidéo de présentation et d'explication du projet avec les différents membres de l'équipe. Enfin, nous avons présenté le site à de nombreux camarades pour qu'ils puissent donner leur avis, qui s'est avéré être globalement positif.

## > FONCTIONNEMENT ET OPÉRATIONNALITÉ :

En ce qui concerne l'avancement du projet, nous avons terminé l'interface utilisateur et nous avons actuellement deux points sur lesquels nous travaillons : la sécurité du site et la mise en place d'une gestion administrative et utilisateur. Nous sommes conscients de l'importance de ces aspects pour garantir la fiabilité et la sécurité de notre application.

Pour ce qui est des tests de vérification de l'absence de bugs et de la facilité d'utilisation, nous avons mené des tests en interne avec l'aide de nos camarades de classe, de notre professeur et d'autres personnes extérieures à l'établissement. Nous avons ainsi pu identifier et corriger plusieurs problèmes potentiels.

Cependant, nous avons également rencontré certaines difficultés techniques au cours du développement du projet. Par exemple, nous avons initialement envisagé de développer une application mobile, mais avons finalement opté pour un site web avec Flask en raison de contraintes techniques. Nous avons également dû réfléchir à la manière de renseigner les informations nécessaires à chaque requête, ce qui nous a amenés à utiliser ChatGPT. Enfin, l'insertion de données dans la base de données a été un défi que nous avons surmonté grâce à des recherches et des tests approfondis.

## > OUVERTURE :

Le développement d'Ekoroji n'étant pas optimal, nous avons réfléchi à certaines améliorations. Tout d'abord, nous aimerions le rendre accessible sur un serveur tel que Raspberry Pi pour qu'il soit disponible de n'importe où. Nous pourrions rendre cela natif en développant une application hors de Python. Cependant, le gain de rapidité peut être freiné par la présence de l'API. Nous devons également résoudre le problème de la gestion des comptes administrateurs et modérateurs, car actuellement, ils ne sont pas distincts, ce qui pose problème puisque tout le monde peut modifier le site à sa guise.

En prenant exemple sur la stratégie de modération de Wikipédia, nous avons pensé à améliorer la fiabilité des textes générés par ChatGPT car nous avons remarqué, lors de nos tests, certaines incohérences dans les explications. Pour cela, nous mettrions en place un système de modération humaine pour empêcher les utilisateurs malveillants de faire des recherches malsaines, telles que celle de "cadavre", mais aussi des erreurs de l'IA dues à une incompréhension de la demande.

Malgré notre réflexion sur le scan, nous n'abandonnons pas cette idée, car elle est pertinente et permettrait une collaboration avec des marques. De plus, c'est également un moyen plus rapide d'accéder à l'information.

Notre stratégie marketing pour attirer un large public est de créer un site web intuitif, minimaliste et ergonomique avec un design attrayant et réfléchi. De plus, cela est éducatif, car les utilisateurs participent à la création du site en ajoutant des éléments à la base de données.

# DOCUMENTATION

Si vous préférez consulter la documentation sur un support autre que GitHub, vous pouvez poursuivre la lecture de ce document. Toutefois, si vous souhaitez avoir accès à l'ensemble de la documentation et aux dernières mises à jour, nous vous recommandons de vous rendre sur notre [dépôt GitHub](#).

Maintenant, vous pouvez continuer à lire la suite de la documentation si vous le souhaitez.

## Prérequis

*Les instructions suivantes ont été testées sur un système d'exploitation Windows. Cependant elles devraient fonctionner sur n'importe quel système d'exploitation tel que Linux ou Mac OS.*

1. Assurez-vous d'avoir Python 3.11 installé sur votre système avant de commencer. Si ce n'est pas le cas, téléchargez-le depuis le site officiel de [Python](#) et installez-le.
2. Cloner le projet Ekoroji depuis GitHub en utilisant la commande suivante :  
`git clone https://github.com/Les-4-fantastiques/ekoroji.git`
3. Naviguez vers le répertoire du projet avec la commande :  
`cd ekoroji`
4. Installez les bibliothèques requises avec la commande suivante :  
`pip install -r requirements.txt`

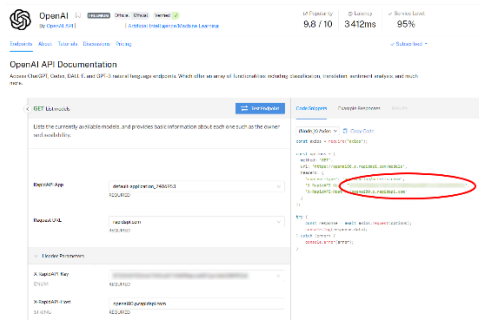
## Configuration de la clé d'API OpenAI

Avant de pouvoir exécuter Ekoroji, il est important de noter que nous utilisons l'API OpenAI. Pour cela vous devez configurer votre clé d'API OpenAI. Voici les étapes à suivre :

*L'utilisation de l'API OpenAI n'est pas obligatoire pour ce projet, cependant elle est nécessaire pour utiliser les fonctionnalités de génération de texte et d'image. Si vous ne souhaitez pas utiliser l'API OpenAI, vous pouvez passer à l'étape suivante.*

*Si vous ne possédez pas de clé d'API OpenAI, vous pouvez en obtenir une gratuitement en suivant les instructions ci-dessous.*

1. Créez un compte sur la plateforme [RapidAPI](#).
2. Accédez à la page de l'[API OpenAI](#) sur RapidAPI.
3. Abonnez-vous à l'API pour obtenir votre clé d'API.



4. Ouvrez le fichier `sources/app/openai/key_to_encryption.py`.
5. Lancez le code, rentrez votre clé d'API quand il vous la demande.
6. Copiez la clé d'API chiffrée retournée.
7. Ouvrez le fichier `sources/app/openai/key_api.txt`.
8. Remplacez la valeur déjà présente par la clé d'API chiffrée que vous avez copiée.

Maintenant, vous êtes prêt à exécuter Ekoroji en utilisant votre clé d'API OpenAI configurée.

Pour accéder à votre quota de demande à l'API, nous vous redirigeons vers [cette page](#).

## Exécution de l'application

1. Vous pouvez maintenant exécuter l'application avec la commande suivante :  
`python sources/run.py`  
*Assurez-vous d'être dans le répertoire du projet avant d'exécuter cette commande.*
2. Ouvrez votre navigateur et accédez à l'adresse <http://localhost:5000> pour voir l'application Ekoroji en action.
3. Pour profiter de l'expérience utilisateur sur notre site, vous devrez soit :
  - Utiliser un écran vertical pour bénéficier de la version portrait.  
*Les fonctionnalités de l'application ne sont pas optimisées pour les écrans horizontaux.*
  - Utiliser le mode développeur **F12** ou **inspecter** de votre navigateur en passant en mode téléphone pour bénéficier de la version portrait.  
 Nous avons conçu le site avec un format 1080x2400 mais il est possible de l'utiliser sur d'autres formats tels que les formats de téléphone proposés par votre navigateur.



# GALERIE

## Les pages principales



## Les déchets



## Les articles

