

Modustry



Aengus Sinanides

-

Térence Chardès

Lycée de l'Élorn

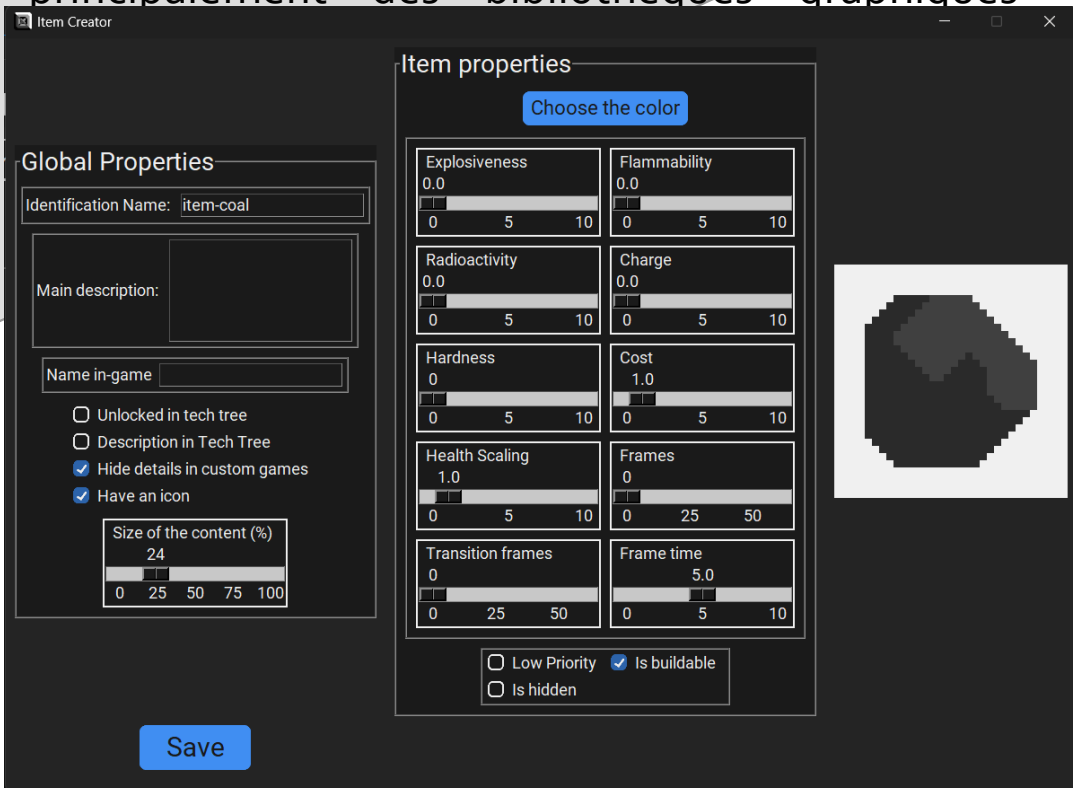
Présentation

Modustry est un logiciel de création de mods pour le jeu [Mindustry](#).

Il permet aux utilisateurs de créer, gérer et exporter leurs mods personnalisés de façon intuitive pour enrichir leur expérience de jeu.

Grâce à une interface graphique intuitive (développée en anglais pour des soucis de public large à atteindre à la fin du développement), les utilisateurs peuvent incorporer de nouveaux éléments à leur mod tel que des ressources et des liquides (les autres éléments du jeu étant toujours en développement mais promettant une immense possibilité de customisation).

Le projet entier est codé en python, utilisant principalement des bibliothèques graphiques



Organisation

La Modustry Team est composée de deux membres :

- **Aengus :**

principal développeur du logiciel. L'idée vient de lui. Il s'occupe de coder les prototypes de fenêtres, et les différentes variables/classes nécessaires au bon fonctionnement et à la gestion du logiciel.

```
from ..data.mindustry_class.Item import Item
from ..data.mindustry_class.Liquid import Liquid
from ..data.variables import item_list, liquid_list, id_list

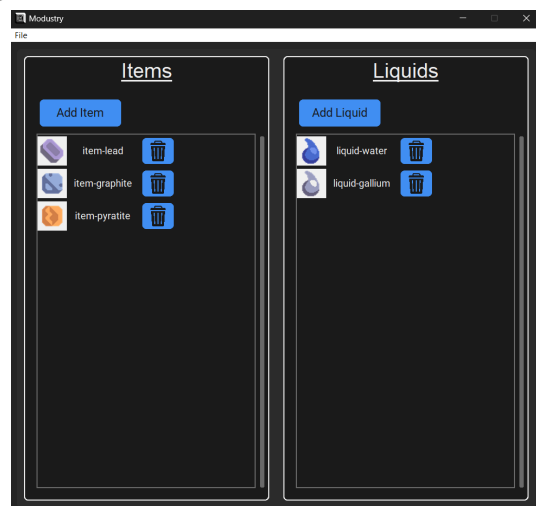
class Model:
    """Gestionnaire des objets"""
    def __init__(self) -> None:
        global item_list

    ##### Item gestion #####
    def add_item(self, item: Item, image_path: str) -> None:
        """Ajoute un nouvel item"""
        item_list.append(item)
        id_list.append({'name': item.name, 'item_id': item_list.index(item), 'image_path': image_path})

    def remove_item(self, index) -> None:
        """Supprime un item"""
        if 0 <= index < len(item_list):
            del item_list[index]
            #get item ids from id_list
            item_id_list = {}
```

- **Térence :**

La lisibilité du logiciel est de sa main : Il s'occupe de nettoyer et fixer les bugs, et de rendre lisible et intuitif l'interface du logiciel. Et il a accessoirement codé la sauvegarde de fichiers



Étapes Importantes du Projet

- 6/12/2024 ► Mise en place de l'environnement de travail
- 17/12/2024 ► Mise en place du système MVC (Model-View-Controller)
- 21/12/2024 ► Rework complet de l'interface graphique
- Du 22/12/2024 au 18/01/2025 ► Réglages de beaucoup de bugs
- 29/01/2025 ► Début de l'organisation du projet pour le rendre.
- Depuis le 29/01/2025 ► Débogage du logiciel entier, travail sur la vidéo et le compte-rendu, et ajout d'un exécutable ainsi que d'un installateur.

Operationalité et fonctionnement :

Malgré l'opérationnalité complète et fonctionnelle de notre logiciel, celui ci ne possède qu'une petite fraction des outils de création qu'il est censé posséder une fois finit. (approximation du travail fini ces 4 derniers mois : 10% des fonctionnalités totales)

Résolution des bugs (voir les commits sur [Github](#))

Notions du programme de NSI :

- Listes
- Dictionnaires
- Classes
- Gestion de fichiers
- Débogage

Difficultés Rencontrés

Widget 'listbox' de Tkinter ne prenant pas en charge l'affichage d'images, il a fallu ruser ► Utilisation intelligente de canvas couplés à des zones de texte pour aligner les noms et leurs icônes ensembles. (Solution apportée par TERENCE)

La solution au problème précédent entraîna un autre problème : les objets ne pouvaient plus être supprimés grâce au bouton 'remove item' ► La solution a été de mettre un bouton en face de chaque lignes d'objets et de les lier pour individuellement pouvoir les supprimer.

L'icône des fenêtres refusait de s'afficher ► s'ensuivit un long temps de tests et d'acharnement de la part de Aengus pour régler ce problème. La solution a été de créer une condition sous format 'try : except :' ainsi que la création d'un exécutable/installateur permettant le bon affichage de l'image ayant un emplacement dédiée sur l'ordinateur.

L'exportation des mods dans le format compatible avec Mindustry a été un challenge : valeurs inexistantes, valeurs du mauvais type, fichier non créé ► une relecture et une réécriture du code permet le bon fonctionnement de celle ci.

L'implémentation des fichiers de sauvegarde de projets était une source de bugs : variables non mises à jour à la suite du chargement, problèmes de duplications/écrasement des données, bugs d'affichages ► une étude approfondie de la structure des listes et de la structure de fichiers cryptés a permis la résolution de ce problème.

Conclusion

Compétences acquises :

- L'adaptabilité face à des problèmes complexes
- La gestion, modification et exportation de fichiers customisés
- Une maîtrise de la création d'interfaces graphiques dynamiques
- Une connaissance plus approfondie du langage python
- Une connaissance plus approfondie de la structure d'un projet python

Idée d'amélioration du projet :

- Ajouter un menu permettant de relocaliser les images si elles ne sont plus présentes.
- Permettre la création de tous les objets du jeu, permettant une personnalisation de l'expérience de jeu bien plus vaste.

Mots de la Modustry

Team

« Étant un grand joueur de Mindustry, l'aventure qu'à été ce projet m'a permis de prendre conscience de l'étendu de mes capacités en programmation mais aussi l'étendue du travail demandé pour la création d'un logiciel, aussi basique soit-il. Je compte continuer ce projet jusqu'au maximum des capacités de la machine, pour rendre plus accessible le monde du modding et montrer au monde entier que malgré les difficultés, la charge mentale, et avec de l'acharnement on peut arriver a tout et même peut-être à avoir son nom gravé dans l'histoire de la communauté que l'on sert » - Aengus

« N'ayant jamais entendu parler de Mindustry, j'ai apprécié participer au développement de ce projet car j'ai pu développer mes capacités à ergonomiser des interfaces et à structuré un projet complet. C'était aussi une expérience où j'ai pu travailler en équipe, une compétence essentielle pour un développeur. - TERENCE