

Date : 03/07/2024
Edition : 1.0

**NOTE DE SYNTHÈSE
DEVELOPPEMENT EN PYTHON
OUTILS ACCESSIBLES AUX UTILISATEURS
MALVOYANTS ET AVEUGLES**

Table des matières

1. PREAMBULE	3
2. LICENCE.....	3
3. LES DIFFERENTS OUTILS.....	5
3.1 ANACONDA SPYDER	5
3.2 CONSOLE ANACONDA PROMPT ET EDITEUR NOTEPAD++.....	5
3.3 ECLIPSE ET LE MODULE PYDEV	6
3.4 VS CODE	6
3.5 PYCHARM COMMUNITY	6
3.6 THONNY	7
3.7 MU EDITOR	7
4. CONCLUSION.....	7

1. Préambule

Ce document présente une petite synthèse suite à l'étude de l'accessibilité de différents outils pour le développement en langage python pour des utilisateurs malvoyants ou aveugles.

Il est impossible de mener une étude exhaustive sur l'ensemble des outils disponibles. Nous faisons donc le choix de nous limiter ici volontairement à des outils gratuits faciles à récupérer sur le web.

On privilégie les outils les plus largement répandus. Ainsi on évite ceux qui sont développées de manière spécifique à destination d'un petit groupe d'utilisateurs. Ce type de solution ne nous paraît en effet pas opportun en termes de pérennité de maintenance.

Dans la suite du document, chaque outil a été testé sous windows 11 avec la loupe windows en mode « contraste élevé » et avec le lecteur d'écran NVDA version 2024 au minimum.

Les outils sont sélectionnés et listés dans un ordre totalement arbitraire.

2. Licence

Ce document est publié par l'association My Human Kit (Rennes) sous licence Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0).

Vous êtes libre de :

- Partager : copier, distribuer et communiquer le matériel par tous moyens et sous tous formats.
- Adapter : remixer, transformer et créer à partir du matériel.

Sous les conditions suivantes :

- Attribution : Vous devez attribuer l'œuvre de la manière indiquée par l'auteur de l'œuvre ou le titulaire des droits (mais pas d'une manière qui suggérerait qu'ils vous soutiennent ou approuvent votre utilisation de l'œuvre).
- Partage dans les Mêmes Conditions : Si vous modifiez, transformez ou créez à partir du matériel, vous devez distribuer votre contribution sous la même licence que l'original.

Pas de restrictions supplémentaires :

Vous n'êtes pas autorisé à appliquer des conditions légales ou des mesures techniques qui restreindraient légalement autrui à utiliser l'œuvre dans les conditions décrites par la licence.

Pour plus d'informations, consultez le texte complet de la licence à l'adresse suivante :

<https://creativecommons.org/licenses/by-nc-sa/4.0>

©My Human Kit, 2024

3. Les différents outils

Nous présentons ci-après la liste des outils qui ont été analysés avec une petite synthèse des principales observations.

3.1 Anaconda spyder

Anaconda Spyder est un environnement de développement intégré (IDE) très complet. Son interface est plutôt intuitive et accessible avec le lecteur d'écran NVDA et avec la loupe windows 11.

Il est facile de régler les préférences de thème de couleur et de coloration syntaxique du code python.

L'intégration d'une fenêtre de console lpython facilite grandement le développement en python.

De nombreux raccourcis sont nativement disponibles. En particulier l'un permet de basculer le focus dans la sous-fenêtre d'édition du code et un second dans la sous-fenêtre lpython.

Cet outil était donc parfaitement accessible depuis des années, jusqu'au moment où une mise à jour à eu pour conséquence de dégrader l'accessibilité.

Depuis le passage de spyder en version 5.5.1, on observe deux problèmes dans la fenêtre d'édition du code :

- La loupe windows 11 ne peut plus suivre le curseur de l'éditeur de texte.
- D'autre part, NVDA répète le texte de la ligne précédente lorsque le curseur est placé sur une ligne vide.

Un ticket a donc été ouvert pour signaler ces deux anomalies d'accessibilité sur le dépôt github. A ce jour, la demande de correction n'a malheureusement pas été prise en compte.

3.2 Console anaconda prompt et éditeur Notepad++

Ici on utilise une simple console "anaconda prompt" en ouvrant à coté l'éditeur Notepad++.

Cette solution, est plus rustique mais efficace. On ne dispose pas ici d'un outil intégré comme dans un IDE classique.

Il faut noter qu'il existe des modules complémentaires pour Notepad ++ qui permettent de coder en Python et d'intégrer une console python en bas de la fenêtre principale de Notepad++.

Par ailleurs, il existe aussi un add-on NVDA dédié à Notepad ++ pour élargir les fonctionnalités.

Il serait sans doute intéressant de reprendre ces différents modules complémentaires déjà existants pour améliorer l'accessibilité et l'expérience utilisateur des développeurs malvoyants et aveugles.

On pourrait ainsi ajouter, par exemple, des raccourcis clavier

- La navigation de fonction en fonction
- Afficher la liste des fonctions
- La navigation de classe en classe
- Afficher la liste des classes
- Aller directement à la définition d'une classe, fonction ou variable
- Exécuter le code
- ...

3.3 Eclipse et le module pydev

L'IDE Eclipse avec le module Pydev présente une interface très accessible par navigation avec un grand nombre de raccourcis clavier. Mais en terme d'expérience utilisateur c'est très complexe surtout pour un informaticien amateur.

3.4 VS code

VS code est un IDE très complet. Mais l'interface est très complexe. La navigation n'est pas si simple malgré le très grand nombre de raccourcis. L'utilisation de cet outil est loin d'être intuitive. Sa prise en main reste donc difficile pour un débutant non accompagné.

Microsoft a publié un tuto sur l'accessibilité de cet outil sous forme de vidéo en Anglais.

3.5 PyCharm Community

Cette application semble plutôt accessible, bien que là encore l'IHM soit un peu complexe. La prise en main par un débutant ne paraît pas aisée sans accompagnement.

3.6 Thonny

Cette application n'est hélas absolument pas accessible par un utilisateur malvoyant qui utilise un lecteur d'écran.

Une issue a été ouverte sur le dépôt github pour signaler le manque d'accessibilité. La réponse indiquait que ce n'était pas possible de le rendre accessible dans la mesure où l'interface avait été développée sous TK (voir anomalie #1178).

Cela est bien dommage, car Thonny est un outil plutôt convivial et intuitif.

3.7 MU editor

L'interface de cet outil est simple à prendre en main. Le contraste des différents éléments est bien adapté aux déficients visuels.

NVDA permet d'accéder aux différents éléments de l'interface. Mais l'analyse doit être poussée plus en détail. Il se pourrait que le développement d'un module complémentaire soit nécessaire pour améliorer son utilisation.

En outre, MU editor permet aussi de développer du code en micro-python et de communiquer avec une carte microcontrôleur du type ESP 32.

Cette fonction est à la fois rare et intéressante, car il existe très peu de solution accessible et simple pour coder en micro-python.

4. Conclusion

Ce petit tour d'horizon de différents outils gratuit pour le développement en python est loin d'être exhaustif.

Il montre qu'à ce jour, aucune solution est véritablement à la fois simple à utiliser et totalement accessible.

Il serait donc intéressant à l'avenir de mettre en place des projets pour développer :

- Un module complémentaire pour Notepad++ à partir de celui qui existe déjà pour augmenter le nombre de raccourcis pour aider au développement en python (voir la section relative à Notepad++). Il serait aussi intéressant d'élargir son usage au module « nppftp » qui permet d'accéder à distance à une machine via SFTP. Enfin, il serait très utile de développer un module pour Notepad++ (en C/C++) pour NVDA (en

python) pour ajouter des fonctions pour coder en micropython et interagir avec un microcontrôleur (type ESP32).

- Un module complémentaire pour spyder afin de corriger le bug de manque du suivi du curseur texte avec la loupe windows. Ce module pourrait aussi aider à corriger le bug d'annonce du contenu de la ligne précédente lorsque l'on est sur une ligne vide.
- Un module complémentaire pour faciliter l'utilisation de MU editor et permettre le codage en micro-python sur microcontrôleur.

Ces activités pourraient être menées dans différents cadres. On pourrait imaginer plusieurs options telles que :

- Un programme de mécénat de compétence dans le cadre d'un accord de partenariat mis en place avec une entreprise dans un but social et solidaire,
- contribuer au programme d'un hackathon de 3 jours.
- Faire l'objet d'un sujet de projet étudiant dans le cadre de partenariat avec une université ou une école d'ingénieur

Toutes ces pistes restent à explorer et à consolider...