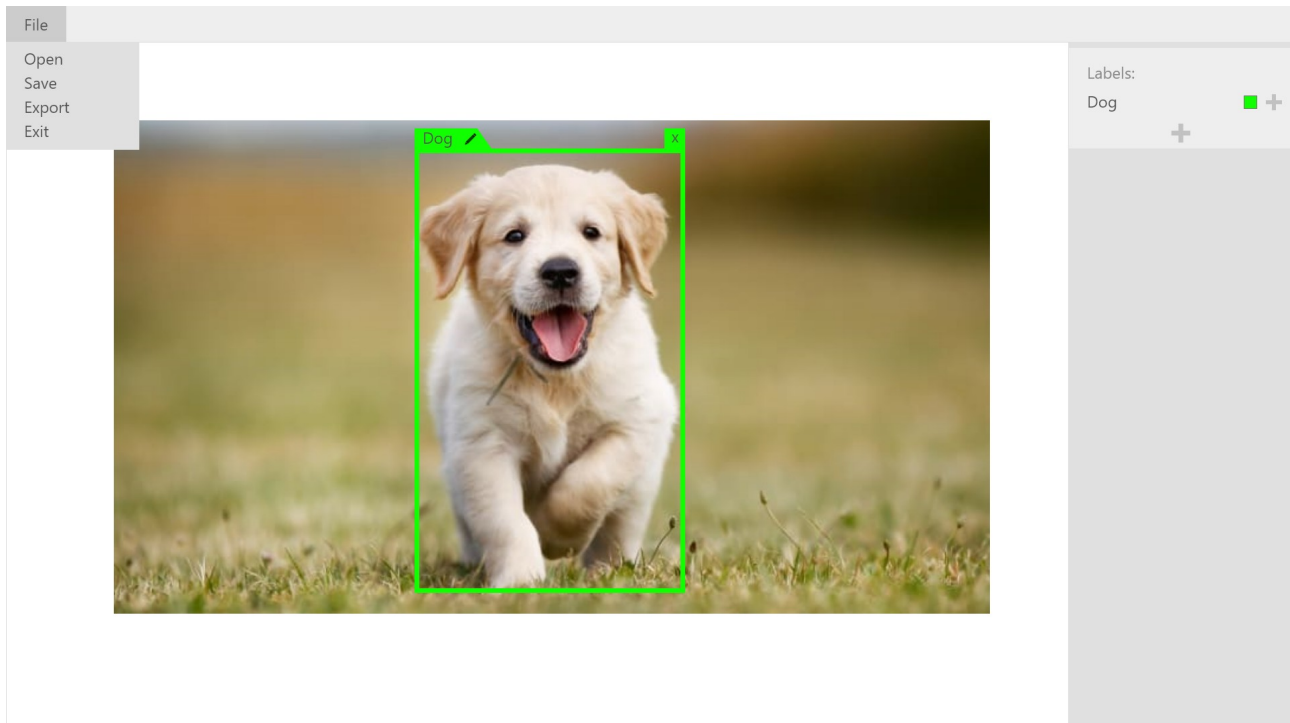


Rendu Laboratoire 1 IHM : Image Annotation Tool

Sketch de notre GUI:

Lors du précédent rendu nous avons imaginé une interface utilisateur épurée et facile d'accès pour l'annotation d'image.



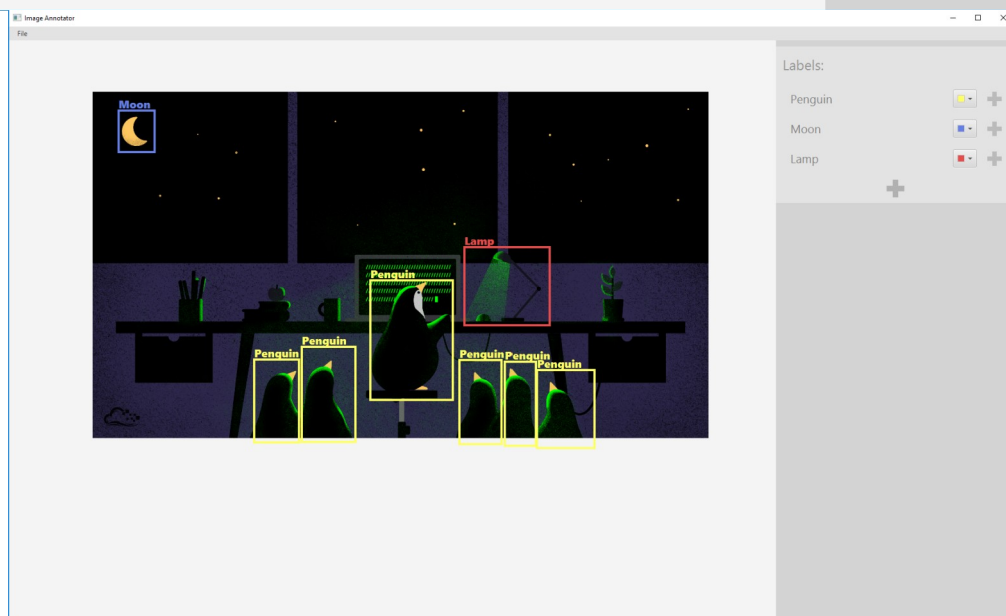
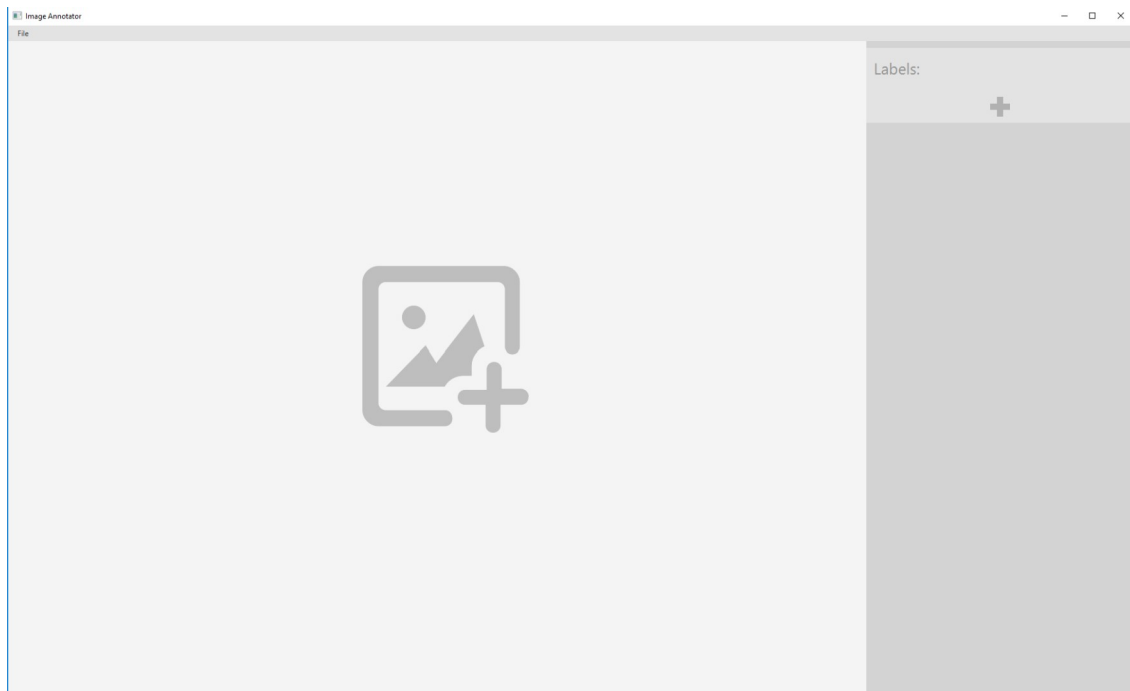
Implementation :

L'implémentation de notre GUI s'est effectuée sur Javafx du fait de sa facilité de prise en main et de sa documentation fournie. Cependant du fait de notre inexpérience sur cette API plusieurs modifications ont dû être apportées au Sketch durant la programmation de cette interface car certaines idées qui paraissaient faciles sur le papier se sont révélées très difficiles à implémenter. Cela nous a néanmoins permis de constater que certaines fonctionnalités que nous avions imaginées n'étaient pas utiles ou alors peu compréhensibles.

Notre application est basée sur un modèle MVC (Model-View-Controller). La view se compose d'un fichier FXML permettant de créer facilement une interface graphique, d'un contrôleur contenant toutes les fonctions gérant les événements de la fenêtre et d'un modèle composé d'une classe d'annotation stockant la position en X et en Y du rectangle de sélection ainsi que sa hauteur et sa largeur et également une classe label contenant le nom de l'annotation et la couleur de sélection de l'annotation.

Fonctionnement de L'application :

- L'utilisateur peut ouvrir une image grâce au bouton Open dans le menu File, celui-ci ouvre le gestionnaire de fichier et permet de sélectionner une image en .png , .jpg et .gif.
- Une fois l'image ouverte l'utilisateur peut créer un nouveau label et l'utiliser pour annoter l'image. Si aucun label n'est créé il n'est pas possible d'annoter l'image.
- On crée un label en appuyant sur le bouton + dans le panel de droite
- Une fois le label créé on peut modifier son nom et sa couleur.
- On peut choisir la couleur de l'annotation grâce au sélecteur de couleur.
- Pour créer une annotation sur l'image on doit appuyer sur le + a coté du nom de label avant de la cadrer dans l'image
- On peut changer le label et la couleur des annotations après les avoir créés sur l'image.
- L'utilisateur peut appuyer sur la touche Z pour supprimer la dernière annotation effectuée sur l'image ou bien sur la touche D(elete) pour supprimer toutes les annotations de l'image.
- Si l'utilisateur appuie sur le bouton Export du menu file il obtient un fichier .csv contenant toutes les annotations.



Difficultés d'implémentation :

La partie la plus dure de notre implémentation c'est trouve dans le fait de créer des nouveaux labels car cela requiert de rajouter des boutons a notre interface durant l'utilisation et donc de ne pas avoir une interface statique. Une autre partie compliquée a été de pouvoir modifier la couleur et le label d'une annotation a l'écran après que celle ci ai été crée, en effet dans notre application une fois que l'utilisateur a sélectionné une zone , il peut modifier la couleur du rectangle par rapport a la couleur initiale ou bien son label et ses modifications se feront directement sur l'interface.

Limitations :

Une limitation importante par rapport a notre Sketch est que le bouton Save n'est pas utilisable, en effet celui-ci devait servir a sauvegarder l'état actuel de notre travail mais cela impliquait de pouvoir modifier directement le fichier image ou bien de trouver un moyen de sauvegarde de l'url du fichier ainsi que des modifications apportées.

Conclusion

Durant implémentation de cette interface nous avons essayé de rester au plus proche de notre sketch original et de garder une facilité d'utilisation maximale, cependant la limitation principale est venu de notre manque de pratique sur JavaFx , ce qui a du nous contraindre a simplifier ou supprimer certains aspects de notre GUI, cependant avec plus de pratique nous pensons qu'il serait possible d'améliorer notre application pour la faire se rapprocher de plus en plus de notre idee originale.