

RAPPORT SAE32_2025

BUT Informatique - 2ème année

SAE3.2-Application "Primitive Image Format"

Youness BOULALAM Algassimou DIALLO Ayoub ANHDIRE

07 janvier 2026

Développement d'une application de conversion d'une image au format PIF (1ère image) + Affichage dans une fenetre d'une image contenue dans un fichier PIF (2ème image).

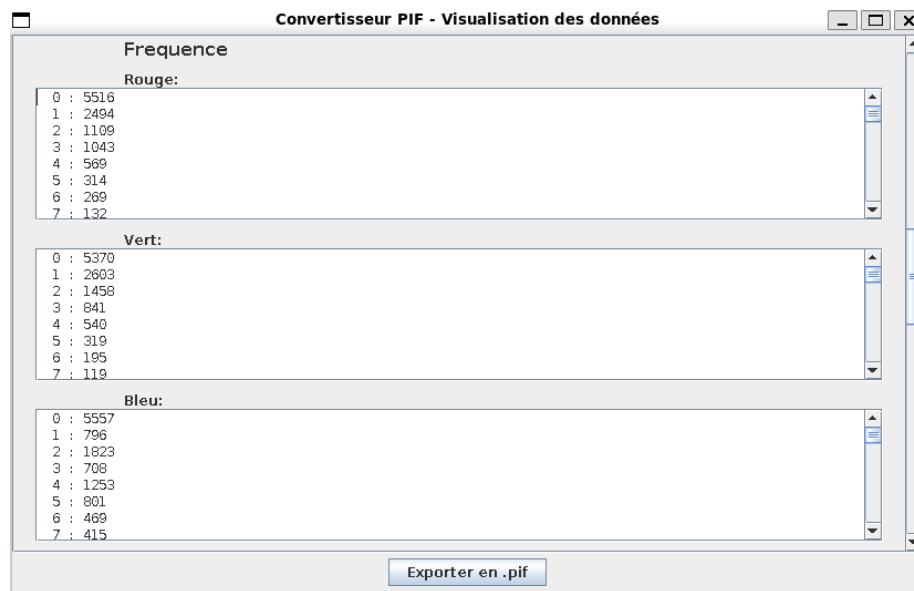


Figure 1: Interface principale du convertisseur , notamment avec les tables de fréquences

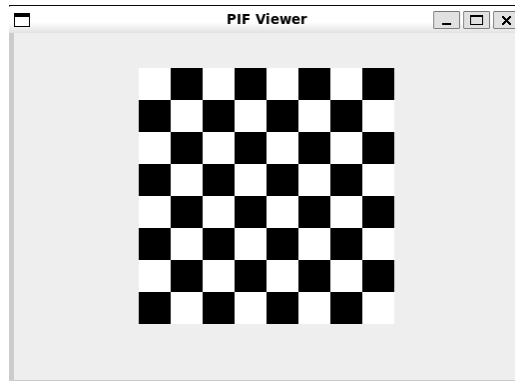


Figure 2: Interface principale du visualisateur ou l'utilisateur peut déplacer l'image avec la souris lorsqu'elle est trop grande

Comptes GIT utilisés pour ce projet

- Youness BOULALAM (Groupe 4) : youness
- Algassimou DIALLO (Groupe 4) : Diallo VM fbleau
- Ayoub ANHDIRE (Groupe 4) : Ayoub ANHDIRE Anhdire Ayoub ANHDIRE

Réalisé en JAVA : Architecture MVC

Dépôt Gitea : https://grond.iut-fbleau.fr/dialloa/SAE32_2025

Professeur : M.Luc Hernandez

Date de rendu : 11 janvier 2026

Sommaire

1	Introduction	3
2	Répartition des taches	3
3	Fontionnalités principales	4
3.1	Conversion au format PIF	4
3.2	Visualisateur au format PIF	4
4	MakeFile du Projet	4

5	Conclusion	4
5.1	Youness BOULALAM	5
5.2	Algassimou DIALLO	5
5.3	Ayoub ANHDIRE	5

1 Introduction

Pour cette deuxième SAE du semestre 3, il nous a fallu réaliser **deux** programmes : un convertisseur d'une image au format png ou du moins supportable par la méthode *read* de la classe **ImageIO**. L'image sera donnée en argument ou sinon elle sera sélectionnée par un **JFileChooser** (à part s'il y a un deuxième argument et qu'il représente un chemin et un nom acceptable). Pour ce programme, on aura une interface simple avec la table des fréquences, la table des codes initiaux (Huffman) et la table des codes canoniques. À la suite de cela, l'utilisateur a la possibilité ou non de pouvoir sauvegarder son image au format PIF.

Le deuxième programme est un visualisateur ou celui-ci est censé afficher dans une fenêtre une image contenue dans un fichier au format PIF. Celui-ci est donnée en argument ou la sélection du fichier sera réalisée à l'aide d'un **JFileChooser**. La taille de la fenêtre sera la taille de l'image à afficher avec une taille qui ne dépasse pas l'écran : si l'image est plus petite que la fenêtre, elle sera centrée et si elle est plus grande que la fenêtre, elle sera partiellement visible et pourra être déplacée à la souris en maintenant le bouton gauche enfoncé.

Ce projet a été réalisé en trinôme, et nous avons développé l'application entièrement en Java, en respectant les bonnes pratiques vues en cours : architecture MVC, création de makefile, et une interface graphique codée sans bibliothèque externe. Un soin particulier a été apporté à l'ergonomie : l'interface est pensée pour être simple, claire, et agréable à utiliser.

2 Répartition des tâches

Nom	Role	Taches effectués
Algassimou DIALLO	Développeur	Tache 1 Tache 2 Tache 3
Youness BOULALAM	Développeur	Tache 1 Tache 2 Tache 3 Tache 4
Ayoub ANHDIRE	Développeur	-JAVADOC(Equipe) -Génération des codes Huffman -Génération des codes canoniques -Interface simple d’affichage des fréquences -Interface simple d’affichage des codes canoniques -Test de BitOutputStream avec des flux réels -Finalisation des affichages des fréquences et codes(Huffman et canoniques) pour le convertisseur -Lecture de l’en-tête -Lecture des longueurs des codes canoniques -Diagramme de classe du convertisseur

3 Fontionnalités principales

3.1 Conversion au format PIF

3.2 Visualisateur au format PIF

4 MakeFile du Projet

5 Conclusion

Nous avons pensé que ce projet ... (à compléter)

5.1 Youness BOULALAM

5.2 Algassimou DIALLO

5.3 Ayoub ANHDIRE