

Rapport de soutenance finale

# **File Explorer**

Mardi, 4 Juin 2024

Arthur Garraud

Tristan Faure

Iliane Formet

**GroupAléa2**

# Sommaire

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Présentation du groupe</b>	<b>4</b>
2.1	Arthur Garraud . . . . .	4
2.2	Iliane Formet . . . . .	4
2.3	Tristan Faure . . . . .	5
<b>3</b>	<b>Projet</b>	<b>6</b>
3.1	Etat de l’art . . . . .	6
3.2	Aspect technique . . . . .	6
<b>4</b>	<b>Tâches</b>	<b>8</b>
4.1	Répartition par membre . . . . .	8
4.2	Description des tâches . . . . .	8
<b>5</b>	<b>Présentation du développement des parties</b>	<b>9</b>
5.1	Commandes . . . . .	9
5.2	Algorithmes . . . . .	12
5.3	UI . . . . .	15
5.4	Site internet . . . . .	20
<b>6</b>	<b>Ressenti personnel</b>	<b>24</b>
6.1	De Iliane . . . . .	24
6.2	De Arthur . . . . .	26
6.3	De Tristan . . . . .	28
<b>7</b>	<b>Conclusion</b>	<b>33</b>
7.1	Conclusion du groupe . . . . .	33
7.2	Sources annexes . . . . .	34

# 1 Introduction

Il existe une multitude d'explorateurs de fichiers, notamment ceux du système d'exploitation de votre ordinateur comme F'inder pour MacOS ou File Explorer pour Windows (pour ne citer qu'eux).

Cependant, nous avons remarqué que l'algorithme de recherche de ceux-ci peut s'avérer être extrêmement lent et que le design pouvait être amélioré pour une meilleure expérience utilisateur.

Nous avons donc décidé de créer notre propre explorateur de fichiers, avec une interface plus agréable et de meilleurs algorithmes.

Cet explorateur de fichiers utilisera une interface graphique.

Il nous permettra d'apprendre à maîtriser Rust, à comprendre et implémenter la manipulation de fichiers et dossiers et de savoir concevoir une interface utilisateur simple et efficace.

Notre objectif est de réussir à créer un explorateur de fichiers qui sera non seulement plus esthétique (design simple sans superflu), mais surtout plus performant.

Cet explorateur sera réalisé en Rust et disponible uniquement sur Linux.

## **2 Présentation du groupe**

### **2.1 Arthur Garraud**

Passionné par l'informatique depuis des années, j'ai toujours été curieux et autodidacte dans de nombreux domaines. J'ai réalisé de nombreux projets qui m'ont permis d'apprendre à maîtriser des outils puissants, que ce soit des langages de programmation ou bien des applications de 3D ou de graphisme.

J'apprécie également le cinéma et j'ai l'habitude de me passionner pour un sujet précis pendant quelques semaines.

Maintenant j'ai décidé de finir ingénieur en informatique afin de pouvoir avoir la possibilité de travailler sur des projets et des missions très différentes de l'une à l'autre.

### **2.2 Iliane Formet**

Depuis mon plus jeune âge, j'ai toujours été passionné par la technologie, la logique et les casse-tête. Ingénieur était donc ce à quoi je me destinais.

Lors d'un projet de labyrinthe pour robot en terminale, j'ai découvert les algorithmes et la programmation. Je me suis alors intéressé et ai commencé à toucher un peu à tout ce qui touche à ce domaine (jeux vidéo, sites internet, algorithmes de calcul). J'ai alors décidé de me tourner vers EPITA pour mes études. Je souhaite dans le futur utiliser mes compétences acquises en école à travers les cours, les conférences et les projets pour entreprendre.

Mes autres loisirs sont le sport, les jeux vidéo, la musique, le cinéma, la nourriture et les sorties entre amis.

Avec ce projet, j'espère perfectionner mes fondamentaux en Rust, mais aussi mieux comprendre les relations entre interface et algorithmes.

## **2.3 Tristan Faure**

Plongé dans l'informatique depuis mes 10 ans par mon oncle. Toujours passionné par de nouveaux langages informatiques, j'ai longtemps cherché le domaine qui me passionne le plus en informatique.

J'ai premièrement commencé en autodidacte à apprendre le codage web, commençant par des sites avec HTML, CSS et JavaScript puis je me suis vite tourné vers la programmation de jeux vidéo en faisant de nombreux jeux sur navigateur, certains plutôt complet, comportait plusieurs mondes à la manière d'un Mario.

Je suis également passé par la programmation en python à l'école et seul afin de créer des bots discords et en LUA lorsque j'ai créé mon serveur GTA 5 RP en début d'année 2020. Toutes ces expériences et les longues heures de conférence et de tuto que j'ai regardé m'ont finalement orienté vers la data science et l'Intelligence Artificiel. J'ai donc décidé de rejoindre Epita pour atteindre mes objectifs. Je pense que ce choix m'aidera à progresser rapidement et efficacement.

## **3 Projet**

### **3.1 Etat de l'art**

De nombreux explorateurs de fichiers existent de nos jours. La majorité intègre une fonction de recherche avancée, un outil de visualisation en mode ligne ou grille, ainsi que de temps en temps des outils de prévisualisation de fichiers. De plus, la fonctionnalité d'héberger des fichiers en ligne sur le Cloud est devenue quasiment essentielle pour explorateurs de fichiers récents. L'interface utilisateur est très souvent simpliste et la majorité des fonctionnalités compliquées disponibles sont masquées par défaut. Cependant, au sein de l'équipe nous n'avons jamais trouvé un explorateur de fichiers qui remplissait parfaitement chaque tâche qui lui était demandée, ils ont tous des points forts mais également des faiblesses. C'est pourquoi nous avons décidé de palier à ce manque.

### **3.2 Aspect technique**

Effectuer ce projet nous a permis de maîtriser les points suivants :

- Implémenter des fonctions basiques de création et gestion de fichiers
- Mettre en place le système permettant de parcourir les différents dossiers et de revenir en arrière au gré des envies de l'utilisateur.
- Rechercher et utiliser la documentation de différentes bibliothèques utiles au projet.
- Créer une interface, qui est pleinement fonctionnelle, mais également esthétique et pratique pour l'utilisateur.
- Interpréter les entrées clavier et souris d'un utilisateur.

- Mettre en place une testsuite qui permette d'accélérer la phase de debug.
- Développer un algorithme optimisé et rapide d'indexage et de recherche des fichiers.
- Concevoir et créer une interface utilisateur simpliste mais avec toutes les fonctionnalités essentielles pour l'utilisateur.
- Développer et mettre en place un site web servant de vitrine à notre application, et permettant son téléchargement.
- Mettre au point des algorithmes de compression rapides et adaptés au type de fichier à compresser.

## 4 Tâches

### 4.1 Répartition par membre

*Resp = Responsable et Supp = Suppléant*

Tâche / Membre	Arthur	Iliane	Tristan
Gestion du projet	-	Resp	-
UI	-	Supp	Resp
Commandes bash	Supp	Resp	-
Algorithmes	Resp	Supp	-
Site internet	Supp	-	Resp

### 4.2 Description des tâches

UI : Réaliser une interface graphique afin que l'utilisateur puisse naviguer dans ses fichiers et dossiers.

Site internet : Faire un site internet permettant de présenter le projet et les membres de l'équipe, ainsi que de permettre le téléchargement.

Commandes bash : Implémenter différentes opérations sur les fichiers sélectionnés par l'utilisateur.

Navigation : Permettre à l'utilisateur de naviguer en utilisant les touches de son clavier ou sa souris.

Fonctionnalités basiques : Implémenter le fonctionnement basique de l'application.

Algorithmes : Implémenter les différents algorithmes de recherche de fichier/contenu de fichier et les algorithmes de chiffrement de fichiers.



## 5 Présentation du développement des parties

### 5.1 Commandes

Pour les commandes, l'objectif était de pouvoir rendre fonctionnel tous les éléments présents dans l'ui et faire d'une liste de données un véritable explorateur de fichiers.

Pour commencer, il a fallu déterminer quelles seraient les fonctionnalités à implémenter afin de nous organiser sur le calendrier d'avancement des tâches. Nous en sommes venus au planning suivant :

#### Première soutenance

- Directory Listing : Afficher en permanence le chemin du dossier dans lequel on travaille
- Navigation : Navigation entre les dossiers depuis l'interface graphique
- Directory creation : Créer un dossier
- Directory deletion : Supprimer un dossier
- File deletion : Supprimer un fichier

#### Seconde soutenance

- File Copy : Copie d'un fichier
- Directory Copy : Copie d'un répertoire
- File cut : Coupe d'un fichier
- Directory cut : Coupe d'un répertoire

#### Rendu final

- Ajout de nouvelles fonctionnalités

Pour le déroulé du développement :

Pour la première soutenance, nous n'avions qu'à implémenter les fonctions faisant les fonctionnalités de base de l'explorateur de fichiers. Il s'agissait de prendre en mains les outils qui allaient être utilisés tout au long de ce projet.

Pour la seconde soutenance, nous devions tester nos fonctions bien que l'interface soit encore en développement. Pour répondre à ce problème, nous avons décidé de réaliser une interface simple afin de pouvoir continuer d'avancer. Ensuite, nous avons implémenter les nouvelles fonctions. Enfin, nous avons connecté tout ce que nous avons fait depuis le début à l'interface. Ici, le plus gros du travail fut la mise en place de procédés afin d'éviter d'avoir des bugs ainsi que la gestion de ceux-ci.

Pour la dernière soutenance, il s'agissait d'ajouter des fonctionnalités étant donné que tout le travail qui devait être fait a été fait. Nous avons donc décidé d'implémenter la possibilité d'ouvrir un terminal depuis un dossier ainsi que la possibilité de cacher les fichiers dont le nom commence par un point. Il a également fallut corriger certains bugs qui avaient été trouvés.

Pour la partie technique de ce chapitre :

Pour implémenter les fonctions, nous avons utilisé deux modules de la bibliothèque standard : `std : :fs` et `std : :env`. En Rust, `std : :fs` permet d'effectuer des opérations sur le système de fichiers. Il fournit des fonctions et des types permettant d'interagir avec le système de fichiers, comme la lecture et l'écriture de fichiers, la création et la suppression de répertoires et la manipulation de métadonnées de fichiers.

`std : :env`, lui, fournit des fonctions et des types pour interagir avec l'environnement, y compris l'accès aux arguments de la ligne de commande, aux variables d'environnement et au répertoire de travail actuel.

Le problème était que nous devions faire attention aux erreurs. Pour cela, il a fallu implémenter plusieurs tests en amont de l'utilisation de chaque fonction pour éviter tout problème lors de l'utilisation du `file explorer`.

De plus, étant donné que certaines boucles tournent en continu, on se retrouve avec énormément de valeurs empruntées après changement. Pour ce faire, il a fallu définir chaque paramètre global en utilisant la librairie `RefCell`. Cette librairie permet d'avoir plusieurs emprunts de données mutables ou immutables à n'importe quel moment. Par exemple, le chemin du répertoire de travail actuel doit être accessible et modifiable à n'importe quel moment et dans n'importe quelle boucle, il a donc fallu le définir en utilisant `RefCell`. De plus, à chaque utilisation, il faut cloner chaque paramètre pour éviter tout conflit.

## 5.2 Algorithmes

Pour ce projet l’une des parties essentielles à développer et implémenter dans l’application était les différents algorithmes utilisés. Ils sont nombreux et de toutes sortes, allant de l’indexation de fichiers à partir de leur chemin absolu jusqu’à un algorithme de compression de fichiers vidéo. Il était donc important que ces algorithmes soient optimisés et efficaces. A l’aide de plusieurs bibliothèques, de beaucoup de documentations et de recherches, l’on a finalement réussi à obtenir un résultat plus que correct, parfaitement fonctionnel et suffisamment optimisé pour notre usage.

L’algorithme d’indexation était le tout premier algorithme sur lequel on a travaillé puisqu’il est nécessaire pour faire une opération de recherche optimisée par la suite. De nombreuses options ont été étudiées (utilisation d’un B-Tree, un fichier json conservant l’indexation précédemment exécutée, et d’autres) mais nous avons finalement conservé une méthode simple mais efficace, la création d’un vecteur contenant un couple de données, le nom du fichier et son chemin absolu. Bien que assez coûteux en mémoire, nous avons décidé que les machines actuelles disposent de suffisamment de mémoire pour ne pas avoir à se soucier de ce degré d’optimisation. De plus, ce vecteur est initialisé à chaque lancement de l’application, afin d’éviter que de nouveaux fichiers sur le système ne soit pas accessibles depuis la recherche car non indexés.

Il a ensuite fallu concevoir l'algorithme permettant d'effectuer la recherche. Celle-ci est une recherche dichotomique qui s'effectue donc sur le vecteur des fichiers indexés. La vitesse de recherche est amplement suffisante, 1 seconde à peine pour chercher parmi plus de 10 millions de fichiers. Le problème de cette méthode est qu'elle ne retourne qu'un seul fichier, celui dont le nom correspondait parfaitement à l'entrée utilisateur. Cependant, l'on voulait que la recherche fonctionne sur un préfix, c'est-à-dire que si l'utilisateur cherche "to" il doit pouvoir avoir accès à tous les fichiers de son système qui commencent par "to" comme "tomate.py" ou "tomber.txt". Il a donc fallu adapter la fonction de recherche pour trouver un élément dont le préfix correspond à l'aide de la recherche dichotomique, puis vérifier si chaque élément à gauche et à droite de l'élément trouvé possèdent également un préfix qui correspond. Si c'est bien le cas, ils sont ajoutés au résultat de la recherche, et l'on vérifie si l'élément suivant peut également être ajouté au résultat. Cela permet d'obtenir un algorithme de recherche efficace, optimisé et qui inclut l'ensemble des fichiers dont le nom commence par la recherche utilisateur.

L'algorithme de compression est également très important dans un explorateur de fichiers, puisqu'il permet à l'utilisateur de compresser ses dossiers/fichiers afin de pouvoir les envoyer facilement si besoin. Il était nécessaire qu'il soit différent entre la compression de dossiers ou bien de fichiers vidéos. Nous n'avons pas intégré la compression de fichiers de type différent puisque cela ne nous a pas semblé utile vu la capacité de

compression possible pour un fichier texte ou une image par rapport à l'intérêt d'utiliser une telle fonction sur ces fichiers.

Pour la compression de dossiers, nous avons utilisé la bibliothèque zip qui a l'avantage d'être simple d'utilisation et très efficace. L'idée était que l'utilisateur choisisse le dossier qu'il souhaite compression, et que à l'aide d'un simple clic de souris il obtient un fichier .zip de ce dossier, fichier qu'il pourra ensuite envoyer. L'utilisation de la bibliothèque ayant beaucoup simplifié les choses, il restait à gérer toutes les erreurs possibles et à bien réagir en fonction de chacune.

Pour la compression de fichiers vidéo, nous avons utilisé la bibliothèque ffmpeg qui est un véritable couteau-suisse du traitement de flux vidéo ou audio. Après avoir lu beaucoup de documentation pour comprendre comment utiliser la bibliothèque, il a été relativement simple de compression le flux vidéo du fichier vidéo cible afin d'obtenir un taux de compression atteignant les 88% de la vidéo à traiter. C'est un résultat bien plus que convenable à nos yeux.

Ainsi, la conception et implémentation des différents algorithmes n'a pas été particulièrement ardue, mais il a été nécessaire de prendre le temps de bien réfléchir et étudier le sujet au début du projet afin de développer dans la bonne direction, évitant ainsi de concevoir un algorithme fonctionnel mais pas assez optimisé par rapport à nos besoins.

## 5.3 UI

Rappel détaillé de la partie UI aux trois soutenances

### 1. Première Soutenance

La première soutenance a été marquée par la prise en main de l'outil de création d'interface, utilisant la bibliothèque GTK4 via `gtk4-rs` pour développer l'interface utilisateur. L'objectif principal était de se familiariser avec les outils et les bibliothèques nécessaires pour créer une interface utilisateur moderne en utilisant Rust et GTK4. Cela impliquait de comprendre la documentation de GTK4, de configurer correctement l'environnement de développement, et de commencer à expérimenter avec les composants de base de l'interface. Dans cette phase, plusieurs composants clés ont été développés, notamment une barre de recherche dynamique utilisant `gtk : :SearchEntry`, permettant une saisie en temps réel et affichant les résultats instantanément avec des animations fluides grâce à `gtk : :Revealer`. La principale difficulté était de s'assurer que la barre de recherche était réactive et rapide, nécessitant une optimisation en temps réel pour éviter les ralentissements. Comprendre et appliquer les mécanismes d'animation dans GTK4 a également été un défi. Toutefois, voir la barre de recherche fonctionner de manière fluide et réactive a été extrêmement gratifiant, ajoutant une touche professionnelle à l'application. En parallèle, l'affichage de l'heure et de la date a été intégré dans le header de l'application, en utilisant la bibliothèque `'chrono'` pour obtenir et actualiser automatiquement l'heure et la date actuelles. La gestion de l'actualisation en temps réel sans surcharger l'interface a présenté des défis, mais voir l'heure et la date s'afficher correctement et s'actualiser

en temps réel a été un grand moment de réussite. De plus, un ‘Column View Datagrid’ a été développé utilisant ‘gtk : :TreeView’ pour afficher un grand nombre de données de manière structurée avec deux colonnes de 10 000 éléments. Gérer un grand nombre d’éléments sans ralentir l’application était une tâche ardue, mais voir le datagrid fonctionner de manière fluide a été très gratifiant. Enfin, un bouton interactif pour un jeu de pile ou face utilisant ‘gtk : :Button’ et un générateur de nombres aléatoires a été mis en œuvre, ajoutant une touche ludique à l’application. Intégrer ce générateur et s’assurer que le bouton réagissait de manière fiable à chaque clic a été un défi, mais ajouter cette fonctionnalité a montré la flexibilité des outils utilisés et a été un moment agréable.

## 2. Deuxième Soutenance

La deuxième soutenance s’est concentrée sur la création de l’interface finale en continuant à enrichir l’interface avec de nouvelles fonctionnalités interactives, basées sur les fondations établies lors de la première soutenance. Parmi les nouveaux composants de l’UI, un bouton ‘gtk : :Button’ permettant d’ouvrir une nouvelle fenêtre avec les crédits de l’application et le lien vers le site internet a été ajouté, utilisant ‘gtk : :AboutDialog’ pour afficher les informations sur l’application. La mise en place d’une nouvelle fenêtre sans interrompre le flux principal de l’application était un défi, mais voir une fenêtre de crédits bien conçue et fonctionnelle a été très satisfaisant. Un autre ajout a été un bouton ouvrant une fenêtre contenant un menu déroulant (‘gtk : :ComboBoxText’) et des zones de texte (‘gtk : :TextView’), permettant des sélections précises et la saisie d’informations. La gestion des entrées utilisateur et des sélections dans



un menu déroulant, ainsi que l'interaction avec les zones de texte de manière fluide, a présenté des défis, mais créer une interface plus interactive et observer les utilisateurs profiter de ces nouvelles fonctionnalités a été très gratifiant. Enfin, l'intégration de ces nouveaux composants a été planifiée pour une interface finale cohérente et esthétique, ajoutant de nombreuses options supplémentaires utiles à l'utilisateur. S'assurer que tous les composants fonctionnaient harmonieusement ensemble sans créer de conflits ou de ralentissements a été un défi, mais voir l'interface évoluer vers un produit final cohérent et riche en fonctionnalités a offert une meilleure expérience utilisateur.

### 3. Troisième et dernière Soutenance

La troisième soutenance a été une étape cruciale, marquée par une refonte significative du header avec des animations et l'ajout de fonctionnalités essentielles pour l'interaction avec différents types de fichiers. L'un des principaux objectifs était d'intégrer des animations fluides dans le header de l'application. Pour cela, nous avons utilisé 'gtk : :Revealer' afin d'ajouter des transitions élégantes à la barre de recherche et aux nouveaux dossiers. Cette tâche a nécessité de nombreux ajustements pour assurer que les animations soient à la fois fluides et non intrusives, apportant ainsi une dimension visuelle attrayante à l'application. Observer ces animations fonctionner parfaitement a été une source de grande satisfaction, enrichissant l'expérience utilisateur globale. L'une des innovations majeures de cette phase a été l'ajout de la fonctionnalité d'ouverture de fichiers texte, image, vidéo et audio directement depuis l'application. Cette fonctionnalité a été conçue pour offrir une expérience utilisateur

intuitive et fluide, permettant aux utilisateurs de cliquer sur différents fichiers et de les ouvrir selon leur type. Pour ce faire, nous avons utilisé un système de ‘match’ en Rust pour gérer les différents types de fichiers et ouvrir une fenêtre appropriée pour chaque type. Pour les fichiers texte, nous avons implémenté une fonction qui lit le contenu du fichier et l’affiche dans une nouvelle fenêtre GTK. Cette fonctionnalité a permis de gérer les fichiers texte de manière efficace, offrant une interface propre et facile à lire. Pour les images, nous avons utilisé une approche similaire, affichant les fichiers image directement dans une nouvelle fenêtre GTK. Cela a permis une intégration fluide des images, enrichissant visuellement l’application. Pour les fichiers vidéo et audio, la gestion a été plus complexe. Nous avons opté pour l’utilisation des applications par défaut du système pour lire ces fichiers, assurant ainsi une compatibilité maximale. Cette approche a permis de contourner les limitations rencontrées lors de la création d’un lecteur vidéo via GTK4, qui ne fonctionnait que sous Windows. Utiliser des applications externes a garanti une meilleure compatibilité multi-plateforme, offrant une solution robuste et efficace pour la lecture de médias. La gestion des différents types de fichiers et l’intégration des méthodes appropriées pour les ouvrir ont été un défi significatif, nécessitant une attention particulière pour assurer que les applications par défaut fonctionnaient correctement sur différents systèmes. Voir la fonctionnalité d’interaction avec les fichiers fonctionner parfaitement et offrir une expérience utilisateur fluide a été un moment de grande satisfaction, démontrant la capacité de l’application à gérer divers types de fichiers de manière efficace et élégante. Cette phase a apporté de nom-

breux moments de satisfaction et de fierté, soulignant l'évolution continue et l'amélioration de l'application.

#### Bilan des Avancements de l'UI

Première soutenance : Mise en place des bases de l'interface avec des éléments fonctionnels clés, démontrant l'efficacité et la réactivité de l'application.

Deuxième soutenance : Ajout de fonctionnalités interactives et planification de l'intégration de ces nouvelles fonctionnalités, orientant vers une interface finale cohérente, esthétique et riche en fonctionnalités.

Troisième soutenance : Refonte du header avec des animations pour une meilleure expérience utilisateur et mise en place de fonctionnalités pour interagir avec différents types de fichiers, améliorant ainsi la praticité et l'efficacité de l'application.

Ces avancements montrent une progression structurée et significative dans le développement de l'interface utilisateur, avec un accent sur l'amélioration de l'expérience utilisateur à chaque étape. Les défis techniques ont été relevés avec succès, apportant de nombreux moments de satisfaction et de fierté tout au long du processus.

## 5.4 Site internet

### La Réalisation du Site Internet

La création du site internet a été une étape cruciale de notre projet. Ce site devait non seulement refléter l'identité et les objectifs de notre explorateur de fichiers, mais aussi fournir une plateforme accessible pour les utilisateurs et les développeurs souhaitant en savoir plus sur notre travail. Voici un aperçu détaillé de la conception et de la réalisation de ce site internet.

### Objectifs du Site Internet

L'objectif principal du site internet était de servir de vitrine pour notre projet, offrant des informations complètes et à jour sur notre explorateur de fichiers. Il devait contenir plusieurs sections clés :

Présentation du Projet : Décrire l'objectif de notre explorateur de fichiers, ses fonctionnalités principales et les technologies utilisées (Rust et GTK4). Équipe de Développement : Présenter les membres de notre équipe, leurs rôles et contributions au projet.

Documentation : Fournir une documentation détaillée pour les développeurs intéressés à contribuer ou à utiliser notre code.

Téléchargements : Offrir des liens pour télécharger l'explorateur de fichiers. Contact : Permettre aux utilisateurs de nous contacter pour des questions, des retours ou des contributions.

### Développement du Site Internet

#### Implémentation Frontend

L'implémentation frontend a commencé par la structuration de l'application React. Nous avons divisé le site en composants réutilisables :

- Header : Contenant le logo du projet, la navigation principale et un accès rapide aux sections importantes.
- Home Page : Une introduction au projet avec des illustrations et des liens vers les sections détaillées.
- Project Details : Une page dédiée à la description technique de l'explorateur de fichiers, ses fonctionnalités, et les technologies utilisées.
- A propos : Présentation des membres de l'équipe avec des photos et des biographies courtes.
- Documentation : Un guide détaillé pour les utilisateurs et les développeurs, avec des instructions pour l'installation et l'utilisation.
- Download : Une page pour télécharger l'application.
- Contact : Une Newsletter de contact permettant aux utilisateurs recevoir nos nouveautés directement.
- Footer : un footer avec la Newsletter.

### Tests et Feedback

Une fois le site fonctionnel, nous avons effectué des tests utilisateurs pour recueillir des retours et identifier les points à améliorer. Ces tests ont impliqué des membres de notre communauté, qui ont exploré le site et donné leur avis sur l'ergonomie, la navigation et la clarté des informations. Les retours ont été très positifs, avec quelques suggestions d'améliorations que nous avons rapidement intégrées.

### Mises à Jour Futures

Bien que le site soit maintenant en ligne et pleinement fonctionnel, nous prévoyons de continuer à l'améliorer et à l'enrichir. Parmi les mises à jour futures, nous envisageons d'ajouter :

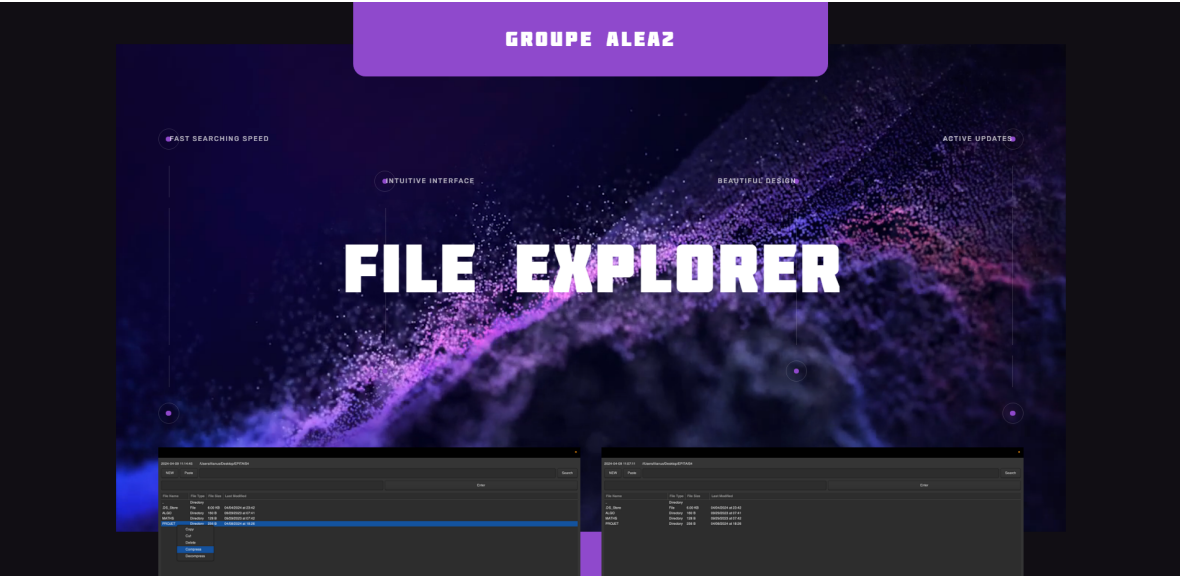
- Un Blog : Pour partager les mises à jour du projet, des tutoriels et des articles sur les technologies utilisées.

- Un Forum : Pour permettre aux utilisateurs et aux développeurs de poser des questions, partager des idées et discuter des améliorations possibles.

- Une Galerie : Présenter des captures d'écran et des démonstrations vidéo de notre explorateur de fichiers en action.

## Conclusion

La réalisation du site internet a été une expérience extrêmement enrichissante, nous permettant de mettre en pratique nos compétences en développement web et en gestion de projet. Le site est devenu une extension essentielle de notre projet d'explorateur de fichiers, offrant une plateforme accessible et informative pour tous ceux qui souhaitent en savoir plus ou contribuer. Chaque étape de la création, de l'implémentation frontend et la gestion des retours utilisateurs, a été marquée par des défis et des réussites, reflétant notre engagement à fournir un produit de qualité. Nous sommes fiers du résultat final et impatients de voir comment il évoluera à l'avenir.



ABOUT OUR FILE EXPLORER

**INTRODUCING OUR RUST FILE EXPLORER, A STREAMLINED SOLUTION FOR EFFICIENT FILE MANAGEMENT AND ORGANIZATION.**

- Lightweight and Fast: Built with Rust, our explorer offers swift performance without compromising on functionality.
- Simplified Interface: Enjoy a clean and intuitive interface for effortless navigation and file management.
- Secure and Reliable: With Rust's memory safety and robust security features, your files are always protected.
- Customizable Options: Tailor your experience with customizable themes and shortcut configurations.

[DOWNLOAD](#)

**ILIANE FORMET**  
Passionné par la technologie  
et les casse-tête

**TRISTAN FAURE**  
Passionné de technologie et de logique

**ARTHUR GARRAUD**  
Passionné par l'informatique

## 6 Ressenti personnel

### 6.1 De Iliane

Lorsque nous avons rendu notre cahier des charges, je pensais que mes collègues étaient complètement fous, car le projet me semblait irréalisable. J'avais tort.

Ce projet a été extrêmement enrichissant à plusieurs égards.

J'ai découvert le langage de programmation Rust, qui s'est révélé être à la fois puissant et performant. L'apprentissage de Rust m'a permis de comprendre de nouvelles approches de gestion de la mémoire et de la sécurité dans le code.

J'ai également appris à utiliser des outils spécifiques à Rust, comme Cargo pour la gestion des dépendances et la compilation, ainsi que des outils de création d'installateurs pour macOS, comme pkgbuild et productbuild. Créer un installateur pour notre application a été un défi technique car je n'avais rien fait de comparable auparavant.

La rédaction de rapports m'a aidé à développer mes compétences en communication technique, me permettant d'expliquer clairement et précisément les fonctionnalités et les aspects techniques de notre application tout en expliquant les difficultés surmontées.

J'ai aussi appris à coder en LaTeX, une compétence précieuse pour la rédaction de documents techniques et académiques.

Cette expérience m'a également permis de renforcer mes compétences en gestion de projet et de groupe. Coordonner les efforts de chacun, gérer



les délais et assurer la qualité du travail a été une expérience formatrice. En plus des aspects techniques, j'ai appris à résoudre des problèmes imprévus, comme les erreurs rencontrées. La capacité à surmonter les obstacles techniques s'est avérée cruciale dans les moments importants pour mener à bien le projet.

Pour conclure, ce projet m'a permis de développer de nombreuses compétences techniques et de gestion de projet. Il a été une expérience profondément formatrice et enrichissante, me préparant mieux pour les défis futurs.

## 6.2 De Arthur

Travailler sur notre explorateur de fichiers en Rust a été une expérience très enrichissante et marquée par plusieurs phases distinctes. Dès le début du projet, j'étais vraiment excité à l'idée de créer quelque chose de concret en appliquant des concepts théoriques. J'étais particulièrement motivé par les défis techniques que représentaient les algorithmes d'indexation pour la recherche dichotomique et la compression de données. La phase de conception du projet a été un mélange d'enthousiasme et de défis intellectuels. Définir les objectifs techniques et élaborer l'architecture de l'application étaient des étapes cruciales. J'ai pris beaucoup de plaisir à collaborer avec mes coéquipiers, à partager des idées et à construire une vision commune du projet.

Lorsque nous avons commencé le développement, j'ai rapidement compris les défis posés par le langage Rust. La courbe d'apprentissage était raide, surtout avec le système de gestion de la mémoire et le concept de propriété. Chaque difficulté surmontée me permettait de mieux comprendre le langage, ce qui était très gratifiant.

Le développement des fonctions d'indexation et de compression a été particulièrement intense. Travailler sur l'algorithme de recherche dichotomique demandait une rigueur particulière. Pour la compression, il m'a fallu approfondir mes connaissances en théorie de l'information et en traitement de données. Ces étapes ont souvent été accompagnées de moments de doute et de frustration, surtout face à des bugs persistants.

Mais la satisfaction de voir les fonctionnalités fonctionner correctement était immense.

En fin de compte, ce projet m'a permis de développer des compétences techniques avancées en programmation et en algorithmes, ainsi que des compétences en gestion de projet et en travail d'équipe. Les défis rencontrés et les apprentissages faits ont rendu cette expérience extrêmement précieuse pour ma formation en ingénierie informatique.

## 6.3 De Tristan

### Mon Ressenti sur le Projet UI

Au début de ce projet, je me sentais à la fois excité et un peu intimidé par l'ampleur de la tâche qui m'attendait. Plonger dans l'univers de Rust et GTK4 était un défi que je savais passionnant, mais aussi complexe. L'idée de créer une interface utilisateur moderne et réactive m'enthousiasmait, mais je savais que cela demanderait beaucoup de travail et d'apprentissage.

La première phase du projet, celle de la première soutenance, a été marquée par une prise en main des outils et des bibliothèques nécessaires pour notre développement. La courbe d'apprentissage était raide, mais chaque petite victoire me donnait confiance. Je me souviens particulièrement de la première fois où j'ai réussi à implémenter la barre de recherche dynamique. La satisfaction de voir cette fonctionnalité prendre vie, réagir instantanément aux entrées et afficher des résultats en temps réel, était immense. Les difficultés techniques étaient nombreuses, comme l'optimisation de la recherche en temps réel pour éviter les ralentissements, ou encore la compréhension et l'application des mécanismes d'animation dans GTK4. Chaque défi relevé était une nouvelle preuve de mes progrès et de ma capacité à surmonter les obstacles.

L'ajout de l'affichage de l'heure et de la date dans le header de l'application a été une autre étape gratifiante. La gestion de l'actualisation en temps réel sans surcharger l'interface a nécessité un équilibre délicat entre fréquence des mises à jour et performance. Voir l'heure et la date s'afficher correctement et s'actualiser en temps réel a été un grand mo-

ment de réussite. Cela m'a montré que je pouvais non seulement créer des fonctionnalités, mais aussi les optimiser pour une meilleure expérience utilisateur.

Le développement du 'Column View Datagrid' a été particulièrement exigeant. Gérer un grand nombre d'éléments sans que l'application ne devienne lente a nécessité une optimisation rigoureuse de l'affichage et une gestion efficace de la mémoire. Mais voir le datagrid fonctionner de manière fluide avec un grand nombre d'éléments a été très gratifiant. Cela a prouvé que les bases que nous posions étaient solides et capables de supporter des fonctionnalités complexes.

L'ajout d'un bouton interactif pour un jeu de pile ou face a apporté une touche ludique à l'application. Intégrer un générateur de nombres aléatoires et s'assurer que le bouton réagissait de manière fiable à chaque clic a été un défi technique, mais aussi une source de plaisir. Cela montrait la flexibilité des outils que nous utilisions et offrait une pause amusante dans le développement de fonctionnalités plus sérieuses.

La deuxième soutenance a marqué une étape importante dans l'enrichissement de l'interface. Basé sur les fondations solides établies lors de la première phase, j'ai pu ajouter de nouvelles fonctionnalités interactives. Le bouton permettant d'ouvrir une nouvelle fenêtre avec les crédits de l'application et le lien vers le site internet a été une addition notable. Voir une fenêtre de crédits bien conçue et fonctionnelle, offrant une transparence et des informations supplémentaires aux utilisateurs, m'a rempli de fierté. Cela montrait que l'application ne se contentait pas d'être fonctionnelle, mais qu'elle offrait aussi une expérience utilisateur riche et

complète.

Le bouton ouvrant une fenêtre contenant un menu déroulant et des zones de texte a été une autre avancée significative. La gestion des entrées utilisateur et des sélections dans un menu déroulant, ainsi que l'interaction avec les zones de texte de manière fluide, a présenté des défis techniques. Mais créer une interface plus interactive et observer les utilisateurs profiter de ces nouvelles fonctionnalités a été très gratifiant. Cela prouvait que nous étions sur la bonne voie pour offrir une application riche en fonctionnalités et agréable à utiliser.

L'intégration de ces nouveaux composants a nécessité une planification minutieuse pour assurer une interface finale cohérente et esthétique. S'assurer que tous les composants fonctionnaient harmonieusement ensemble sans créer de conflits ou de ralentissements a été un défi constant. Mais voir l'interface évoluer vers un produit final cohérent et riche en fonctionnalités a offert une meilleure expérience utilisateur et a été une source de grande satisfaction.

La troisième et dernière soutenance a représenté l'apogée de notre travail, avec une refonte significative du header et l'ajout de fonctionnalités essentielles pour l'interaction avec différents types de fichiers. L'utilisation de 'gtk : :Revealer' pour ajouter des animations fluides dans le header a enrichi visuellement l'application. Observer ces animations fonctionner parfaitement a été une source de grande satisfaction, ajoutant une dimension visuelle attrayante à l'application.

L'une des innovations majeures de cette phase a été l'intégration de la fonctionnalité d'ouverture de fichiers texte, image, vidéo et audio. Cette

fonctionnalité a été conçue pour offrir une expérience utilisateur intuitive et fluide, permettant aux utilisateurs de cliquer sur différents fichiers et de les ouvrir selon leur type. La gestion des différents types de fichiers et l'intégration des méthodes appropriées pour les ouvrir ont été un défi significatif. S'assurer que les applications par défaut fonctionnaient correctement sur différents systèmes a nécessité une attention particulière. Voir cette fonctionnalité fonctionner parfaitement et offrir une expérience utilisateur fluide a été un moment de grande satisfaction.

Le développement de cette fonctionnalité a été particulièrement gratifiant car elle a démontré la capacité de l'application à gérer divers types de fichiers de manière efficace et élégante. Cela montrait que nous avions réussi à créer une application non seulement fonctionnelle, mais aussi pratique et agréable à utiliser. Cette phase a apporté de nombreux moments de satisfaction et de fierté, soulignant l'évolution continue et l'amélioration de l'application.

En rétrospective, ce projet a été une aventure incroyablement enrichissante. Chaque phase a apporté son lot de défis et de récompenses. Les moments de difficulté, où il semblait que les obstacles étaient insurmontables, ont été compensés par la satisfaction de voir les fonctionnalités prendre vie et fonctionner parfaitement. Le chemin parcouru depuis la prise en main initiale des outils jusqu'à la création d'une application riche en fonctionnalités a été jalonné de progrès constants et de réussites.

Ce projet m'a permis de développer non seulement mes compétences techniques, mais aussi ma capacité à résoudre des problèmes et à surmonter les défis. Il m'a montré l'importance de la persévérance, de la

patience et de l'attention aux détails. Chaque petit succès a été un pas en avant vers la réalisation de notre vision d'une application moderne et réactive.

Au final, je me sens extrêmement fier de ce que nous avons accompli. Ce projet m'a apporté une compréhension profonde de la création d'interfaces utilisateur et une appréciation de la complexité et de la beauté du développement logiciel. Les défis techniques relevés, les moments de satisfaction et les succès obtenus ont fait de cette expérience un chapitre inoubliable de mon parcours.



## 7 Conclusion

### 7.1 Conclusion du groupe

En conclusion, le développement de notre explorateur de fichiers en Rust a été un projet ambitieux et extrêmement formateur. Nous avons commencé avec l'objectif de créer une application plus performante et esthétiquement agréable que les explorateurs de fichiers existants comme Finder ou File Explorer. En cours de route, nous avons non seulement réussi à atteindre cet objectif, mais nous avons également acquis de nombreuses compétences essentielles. Le projet nous a permis de maîtriser la création et la gestion de fichiers, de parcourir des dossiers de manière intuitive, et de mettre en œuvre une interface utilisateur à la fois fonctionnelle et esthétique. Nous avons également appris à rechercher et utiliser efficacement la documentation des bibliothèques Rust, ce qui s'est avéré crucial pour le succès de notre projet.

Nous avons développé des algorithmes optimisés pour l'indexation et la recherche de fichiers, ainsi que des algorithmes de compression adaptés à différents types de fichiers. Ces compétences techniques ont été complétées par la création d'une testsuite pour accélérer la phase de debug, et par le développement d'un site web pour présenter notre application et permettre son téléchargement. Le projet nous a également permis de nous familiariser avec la conception et la création d'interfaces utilisateur simplistes mais complètes, ainsi qu'avec l'interprétation des entrées clavier et souris des utilisateurs. L'implémentation des com-

mandes bash pour les opérations sur les fichiers sélectionnés a été un autre aspect important de notre travail.

En travaillant sur ce projet, nous avons non seulement amélioré nos compétences techniques, mais nous avons également appris à collaborer efficacement en équipe, à gérer un projet de bout en bout, et à surmonter les défis posés par un nouveau langage de programmation. Cette expérience a été extrêmement enrichissante et constitue un pilier central de notre formation en ingénierie informatique. Notre explorateur de fichiers en Rust, disponible uniquement sur Linux, est le fruit de ce travail acharné et de notre passion pour l’informatique.

## **7.2 Sources annexes**