



rustique

February 24, 2025

Contents

1	Introduction	3
1.1	Presentation du projet	3
2	Chronologie	5
2.1	Répartition des responsabilités	5
2.2	Collaboration	6
2.3	Première soutenance	7
3	Technologies retenues :	8

1 Introduction

1.1 Présentation du projet

Peinture numérique : rustique

Introduction :

Dans le cadre de notre projet S4, nous proposons de développer une application de peinture numérique en Rust, que nous appellerons "rustique". Ce projet vise à créer un outil de dessin fonctionnel tout en nous permettant d'explorer les concepts fondamentaux de la programmation en Rust et du développement d'interfaces graphiques.

Une application de peinture numérique est un logiciel qui permet aux utilisateurs de créer des œuvres d'art numériques en utilisant divers outils de dessin et de peinture. Ces applications sont largement utilisées dans des domaines tels que l'art numérique, le design graphique et l'illustration.

L'objectif principal de "rustique" est de fournir une interface intuitive pour la création artistique numérique tout en servant de plateforme d'apprentissage pour notre équipe. En implémentant des fonctionnalités de base comme des outils de dessin, un système de calques et des filtres, nous pourrions construire une application éducative et utile.

Objectifs du Projet :

1. Créer une application de peinture numérique fonctionnelle et extensible
2. Apprendre à utiliser Rust dans un contexte de développement d'applications graphiques
3. Implémenter des fonctionnalités de base de dessin, de manipulation d'images et de gestion de fichiers
4. Développer des compétences en conception d'interfaces utilisateur et en traitement d'images

Description du Projet :

"rustique" offrira aux utilisateurs un ensemble d'outils de dessin de base dans une interface graphique conviviale. L'application permettra de créer, éditer et sauvegarder des images numériques. Elle inclura également des fonctionnalités plus avancées comme un système de calques et des filtres personnalisés.

Fonctionnalités Principales :

1. Outils de Dessin : Implémentation de pinceaux, crayons, gommes et formes de base.
2. Système de Calques : Création et gestion de calques pour une édition plus flexible.
3. Palette de Couleurs : Sélection et gestion des couleurs.
4. Gestion de Fichiers : Sauvegarde et chargement d'images dans des formats courants.

Défis d'Apprentissage :

- Maîtriser les bases de la programmation en Rust - Apprendre à utiliser des bibliothèques graphiques en Rust - Implémenter des algorithmes de traitement d'image - Concevoir une interface utilisateur intuitive

Extensions Possibles (si le temps le permet) :

- Ajout d'un mode de dessin collaboratif en réseau local - Implémentation d'un mode console pour des croquis rapides - Création de filtres et d'effets inspirés du thème de la rouille

Conclusion : Une application de peinture numérique est un excellent projet pour explorer des concepts de programmation avancés tout en créant un outil pratique et créatif. Avec Rust, nous pourrions construire une application performante et sûre, tout en apprenant un langage de programmation moderne et puissant. Ce projet nous permettra de développer des compétences précieuses en développement logiciel, en traitement d'images et en conception d'interfaces utilisateur.

2 Chronologie

2.1 Répartition des responsabilités

Vincent : Responsable des outils de dessin et de la palette de couleurs

Tâches :

Implémenter les outils de dessin de base (pinceaux, crayons, gommages, formes).

Gérer la sensibilité des outils (taille, opacité).

Développer une palette de couleurs permettant de sélectionner, enregistrer et réutiliser des couleurs.

Bibliothèque : piet ou wgpu pour le rendu graphique.

Leonardo : Responsable du système de calques

Tâches :

Implémenter un système de calques (création, suppression, organisation, visibilité).

Ajouter la fonctionnalité permettant de fusionner ou dupliquer des calques.

Gérer les interactions entre calques (transparence, positionnement).

Bibliothèque : egui pour les interfaces graphiques.

Nabil : Responsable de la gestion des fichiers et des formats d'image

Tâches :

Implémenter la sauvegarde et le chargement des fichiers dans des formats courants (PNG, JPEG).

Développer une structure de projet permettant de stocker plusieurs calques dans un format personnalisé.

Ajouter la gestion des raccourcis pour des opérations courantes (Ctrl+S, Ctrl+O, etc.).

Bibliothèque : image pour la manipulation d'images.

Aurélien : Responsable de l'interface utilisateur et de l'expérience utilisateur (UX)

Tâches :

Concevoir une interface utilisateur intuitive (barre d'outils, fenêtres de calques, palette de couleurs).

Gérer les interactions utilisateur (menus, clics, raccourcis).

Tester l'interface pour garantir une expérience utilisateur fluide.

Bibliothèque : egui pour une interface graphique moderne et légère.

2.2 Collaboration

Bien qu'on ait donnée à chacun des tâches spécifiques à gérer, voici comment on a décidé de collaborer :

EC = En Charge, A = Aide

	Vincent	Leonardo	Nabil	Aurélien
Outils				
Outils de bases	EC	A		
Sensibilité	EC		A	
Couleurs	EC			A
Système de Calques				
Creation et Modification		EC	A	
Fusion et Duplication	A	EC		
Interactions entre Calques		EC		
Gestion de fichiers				
Sauvegarde et Chargement			EC	A
Sauvegarde de Calques		A	EC	
Interface Utilisateur				
Barre d'outils et fenêtres				EC
Interactions utilisateur	A	A	A	EC

Table 1: Table de division des rôles

Ce tableau représente la foundation de notre travail, c'est à dire ce que chacun prends en charge mais aussi l'aide que chacun peut apporter, même si il se peut que l'on s'entraide sur différentes parties autre que ce que le tableau indique au fur et à mesure de l'avancement du projet.

La partie "Interactions utilisateur" dans L'interface est spéciale car elle doit être modifiée par toutes les autres parties, donc bien qu'Aurélien y est en charge elle sera travaillé par tous.

2.3 Première soutenance

Pour la première soutenance, voici un tableau qui représente les tâches que l'on souhaite faire et aussi un tableau qui représente l'avancement voulu au cours de ces semaines.

Prénom	Tâches
Vincent	<ul style="list-style-type: none">• Une partie des outils de base (Pinceau, gomme, sceau, pipette)
Leonardo	<ul style="list-style-type: none">• Création et implémentation des calques
Nabil	<ul style="list-style-type: none">• Sauvegarde et chargement d'images
Aurélien	<ul style="list-style-type: none">• Interface simple

Évidemment, si le temps le permet, on essaiera d'avancer sur d'autres tâches, mais pour cette première soutenance on se concentrera sur des fonctionnalités assez simples.

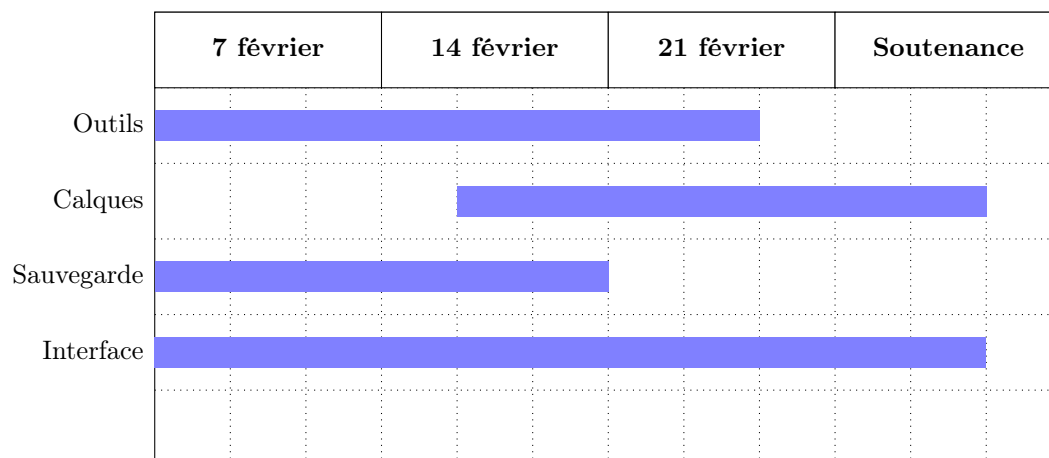


Figure 1: Avancement des taches

3 Technologies retenues :

"wgpu" : "wgpu" est une bibliothèque graphique basée sur WebGPU qui offre un rendu GPU performant. Elle permet d'afficher et de dessiner rapidement des images en exploitant la carte graphique. Contrairement à d'autres bibliothèques, elle est sécurisée et compatible avec Vulkan, DirectX et Metal. On l'utilisera pour dessiner sur la toile (canvas) en temps réel.

"egui" : "egui" est une bibliothèque ergonomique et facile à utiliser pour créer une interface graphique. Contrairement à druid ou gtk-rs, elle ne nécessite aucun fichier de configuration. Elle est réactive et mise à jour en temps réel, ce qui est essentiel pour un logiciel de dessin. On l'utilisera pour créer les menus, la barre d'outils et les boutons de sélection d'outils.

"image" : "image" est la bibliothèque standard pour charger, enregistrer et manipuler des images. Elle prend en charge de nombreux formats (PNG, JPEG, BMP, etc.). Permet l'application de filtres et la modification des pixels. On l'utilisera pour enregistrer les dessins, charger des images et appliquer des filtres.

"serde" : "serde" permet de sérialiser et désérialiser des données en JSON, TOML ou d'autres formats. Il est très utile pour enregistrer les préférences utilisateur et les paramètres de l'application. Il facilite aussi la gestion d'un format de fichier propre à "rustique" (exemple : .rustiq). On l'utilisera pour enregistrer les calques et les paramètres dans un fichier.