

EPIPAINTOR

par Fast Thinking

Rapport de soutenance : Soutenance 2.



MAHRAOUI Adam - HAMDOUN Zaky - VARLIETTE Robin - AZIKA Christian

Table des matières

1 Projet et groupe	5
1.1 Le projet : Epipaintor	5
1.2 Le groupe : Fast-thinking	5
1.3 Présentation des membres	5
1.3.1 Adam Mahraoui	5
1.3.2 Christian Aziaka	5
1.3.3 Zaky Hamdoun	6
1.3.4 Robin Varliette	6
1.4 Organisation du rapport de soutenance	6
2 Rappel sur le projet	7
2.1 Idée	7
2.1.1 Nom	7
3 Fonctionnalités	8
3.1 Les outils	8
3.1.1 Le pinceau	8
3.1.2 Le seau de remplissage	11
3.1.3 Palette de couleur	12
3.2 Les filtres	13
3.2.1 Filtre Nuances de Gris : "Grayscale"	14
3.2.2 Filtre Sépia	14
3.2.3 Filtre Gaussien	15
3.2.4 Filtre Rouge	16
3.2.5 Filtre Binaire et Otsu (noir et blanc)	17
3.2.6 Filtre Binaire (colorisé)	18
3.2.7 Filtre Négatif ou Inversion des couleurs	18
3.2.8 Filtre Gamma	19
3.3 Les formes géométriques	20
3.3.1 La ligne	21
3.3.2 Le rectangle (et le carré !)	22
3.3.3 Le triangle	22

3.3.4	Le cercle	23
3.3.5	Prévisualisation des formes	24
3.4	L'interface	25
3.4.1	Vue d'ensemble	25
3.5	Le retour en arrière / avant	30
3.6	Site Web	31
3.6.1	HTML : La base du site	31
3.6.2	Le CSS : application du style	32
3.6.3	Le Javascript : ajout de détail	32
3.7	Répartitions des tâches	33
3.8	Planification	34
3.9	Ressenti personnel	35
3.9.1	AZIAKA Christian	35
3.9.2	HAMDOUN Zaky	35
3.9.3	MAHRAOUI Adam	35
3.9.4	VARLIETTE Robin	35
3.10	Conclusion	36

1 Projet et groupe

1.1 Le projet : Epipaintor

Notre projet est simple : un photoshop intelligent, permettant d'aider les personnes étudiant ou travaillant à EPITA !

Derrière le doux nom d'**Epipaintor** résidera un utilitaire, permettant de dessiner ou modifier une image.

L'idée est d'implémenter tout un éventail de fonctionnalités graphique, utilisant chacun leur algorithmes propres, sur le modèle du logiciel Photoshop.

1.2 Le groupe : Fast-thinking

Notre groupe se nomme **Fast-Thinking** :

Il tire son nom de la vivacité d'esprit de ses membres, et de sa volonté à développer des projets ingénieux. Fast-Thinking est un groupe formé de 4 étudiants en deuxième année de l'école d'ingénieur EPITA, aux origines diverses (Italien, Algérien, Marocain, Français). La diversité culturelle est une des grandes forces de **Fast-Thinking** lui permettant d'avoir une combinaisons de raisonnements différents les uns des autres.

1.3 Présentation des membres

1.3.1 Adam Mahraoui

Étudiant à EPITA, je suis passionné par le monde de la programmation et du numérique en général. La curiosité est un trait de personnalité qui me caractérise bien et c'est pour cela que cette idée projet m'a plu. J'ai vu de plus en plus apparaître de nouvelles fonctionnalités incroyables sur les logiciels de montage photo en me demandant toujours comment cela fonctionnait mais je n'ai jamais vraiment penché sur le sujet. Grâce à ce projet je vais pouvoir enfin répondre à mes questions en renforçant mes compétences en C sur un projet concret.

1.3.2 Christian Aziaka

Bonjour, je suis Christian Aziaka et je suis passionné d'IoT, programmation, montage vidéo, d'UX, montage photo, de design et de production musicale.

1.3.3 Zaky Hamdoun

Passionné par l'informatique et la géopolitique, j'ai rejoint l'EPITA avec plus tard l'envie de devenir ingénieur en informatique spécialisé en cybersécurité, ou en big data et intelligence artificielle. Actuellement Trader pour Bright Future Investment Management, je m'occupe du développement d'algorithme de trading automatique. Cependant, mon penchant pour la géopolitique et les métiers de la défense font également que je suis intéressé par l'Open Source Intelligence (OSINT) et les renseignements, qui se résument très souvent à du traitement d'image. J'aime également les mathématiques et la philosophie.

1.3.4 Robin Varliette

Bonjour, et enchanté, à quiconque lisant ces lignes !

Je suis Robin Varliette et je suis le leader du groupe Fast-Thinking.

J'ai 20 ans, et je suis un passionné depuis toujours d'algorithmique, d'information, de mathématiques, ainsi que d'Intelligence Artificielle.

Amateur de sport, j'affectionne le volley, le handball, ou encore le badminton.

Ouvert à tous les horizons, j'adore découvrir de nouveaux univers par le voyage, ou bien la musique.

Enfin, j'ai pour hobby de partager mes passions en proposant mon aide en algorithmique ou en programmation aux élèves d'EPITA qui en ont besoin.

1.4 Organisation du rapport de soutenance

Le document est présenté d'une manière bien précise. Les premières parties constituent des rappels. Ensuite, les parties suivantes et notamment celles dans la section numéro 3 "Fonctionnalités" sont pour certaines reprises du rapport de soutenance précédent, et pour d'autres, nouvelles. Cette organisation permet d'avoir une vision globale de l'évolution du début du projet, et entre chacune des soutenances. Les parties nécessitant une mise en exergue de leur évolution temporelle seront précisées au fur et à mesure. Ainsi, si un changement a été effectué sur une fonctionnalité qui fut déjà présente précédemment, alors ceci sera précisé.

2 Rappel sur le projet

2.1 Idée

L'idée du projet était à l'origine de faire une intelligence artificielle : Un Chat-bot capable de converser avec un être humain. Elle était née du fait que les membres du groupe sont particulièrement attirés par les intelligences artificielles, eux-mêmes ayant pu en développer une dans le cadre leur projet effectué au semestre passé.

L'idée du Chat-Bot comme Intelligence Artificielle prends ses sources dans deux projets existants :

- ChatGPT, le chat bot aux performances fulgurantes développée par Open-AI, qui a émergée très récemment en 2022.

- Tay.ai, le chat bot développé par Microsoft et utilisable sur Twitter, qui était une expérience de "Conversational Understanding" (en français : Apprentissage en conversation).

Cependant, pour des problèmes de complexité, et de barème, le projet n'a pas été accepté et à du être radicalement changé.

Le projet de faire un utilitaire sur la base de Photoshop nous a alors été proposé par notre professeur en charge des projets de 4ème semestre, et nous avons décidé de suivre ses recommandations afin de pouvoir proposer un projet qualitatif.

2.1.1 Nom

Pour commencer, il est important de rappeler qu'en tant qu'élèves d'EPITA, nous avons pour **devoir** d'honorer notre merveilleuse école. Qui serions nous sans elle, pas vrai ?

Il existe une règle implicite dans la nomenclature des choses à EPITA : Chaque nom doit avoir pour préfixe '**Epi**' que cela soit les associations, les sites internets, les organisations ou projets reliés à l'école ont presque tous comme préfixe '**Epi**'. Ainsi, nous suivons donc cette convention car nous portons les traditions de l'EPITA dans notre cœur.

Au départ, il était question d'appeler notre ChatBot '**EpiHelp**'.

Cependant, nous avons grandi avec le jeu '**Akinator**', un ChatBot inventé par l'ingénieur informatique français Arnaud Megret, capable de deviner un personnage en posant des questions à un utilisateur.

Cela étant dit, nous avons décidé de rendre honneur à cette prouesse technologique de l'époque en ajoutant '**nator**' à notre préfixe '**Epi**', formant ainsi le nom '**Epinator**'.

Le projet ayant changé en étant un utilitaire graphique, nous avons légèrement changé le nom du projet en remplaçant le **nator** en **paintor**, donnant **Epipaintor**.

3 Fonctionnalités

3.1 Les outils

3.1.1 Le pinceau

Le pinceau est l'un des outils les plus importants dans les logiciels de retouche d'image, tels que Adobe Photoshop, GIMP, Corel Painter et bien d'autres. Cet outil permet de dessiner, de peindre et d'appliquer des effets spéciaux sur des images numériques. Il est souvent utilisé pour créer des illustrations, des peintures numériques, des graphismes, des logos, des bannières publicitaires, des photographies retouchées et bien plus encore.

Le pinceau est un outil très flexible qui peut être utilisé pour appliquer des couleurs, des textures et des motifs sur une image. Les artistes et les designers utilisent souvent des pinceaux personnalisés pour ajouter des effets spéciaux et des textures uniques à leurs œuvres. Ces pinceaux personnalisés peuvent être créés à partir de photos, de dessins ou d'autres images, puis importés dans le logiciel de retouche d'image.

Les retoucheurs d'images peuvent utiliser des pinceaux pour ajouter ou supprimer des détails, tels que des cicatrices, des imperfections ou des taches sur la peau.

Les logiciels de retouche d'image offrent souvent une grande variété de formes, de tailles et de duretés de pinceaux, ainsi que des options pour régler l'opacité et la pression du pinceau. Les utilisateurs peuvent également choisir parmi une variété de modes de fusion de pinceaux, qui leur permettent de superposer des couleurs, des textures et des effets pour créer des effets visuels complexes et personnalisés.

En somme, l'outil pinceau est un élément essentiel de la boîte à outils d'un artiste ou d'un designer numérique. Il permet une grande liberté d'expression et de créativité, tout en permettant aux utilisateurs de manipuler les images de manière précise et détaillée. Que ce soit pour créer des illustrations, retoucher des photographies ou créer des designs graphiques, le pinceau est un outil indispensable pour tout professionnel de la création numérique.

Pour gérer cette fonctionnalité, l'algorithme de Bresenham a été mis en place. L'algorithme de Bresenham est un algorithme de traçage de lignes largement utilisé en informatique graphique pour dessiner des lignes sur un écran ou une image. Il a été développé par Jack E. Bresenham en 1962 alors qu'il travaillait chez IBM.

L'algorithme de Bresenham fonctionne en tracant la ligne pixel par pixel, en déterminant si chaque pixel doit être allumé ou non pour que la ligne apparaisse continue. L'algorithme utilise un calcul incrémental pour trouver les pixels les plus proches de la ligne, ce qui permet de dessiner la ligne de manière rapide et efficace.

Pour commencer, l'algorithme reçoit les coordonnées des deux points A(x₁,y₁) et B(x₂,y₂) qui définissent la ligne à dessiner. L'algorithme détermine ensuite si la ligne est plus inclinée horizontalement ou verticalement, en calculant la différence entre les coordonnées x et y de A et B.

Si la différence entre les coordonnées x est plus grande que celle entre les coordonnées y, cela signifie que la ligne est plus inclinée horizontalement. L'algorithme initialise une variable appelée "erreur" à la moitié de la différence entre les coordonnées x. Il trace ensuite le premier pixel à la position (x1,y1) et examine le pixel suivant sur la ligne. Si l'erreur est positive, il déplace le tracé d'un pixel vers le haut ou le bas, selon la direction de la ligne. Sinon, il continue dans la même direction.

Si la différence entre les coordonnées y est plus grande que celle entre les coordonnées x, cela signifie que la ligne est plus inclinée verticalement. L'algorithme initialise la variable "erreur" à la moitié de la différence entre les coordonnées y et effectue une opération similaire au cas précédent, mais cette fois-ci en déplaçant le tracé vers la gauche ou la droite, selon la direction de la ligne.

L'algorithme continue de cette manière jusqu'à ce que le dernier pixel soit atteint, à la position (x2,y2). À chaque étape, il calcule l'erreur en fonction de la différence entre les coordonnées x et y et déplace le tracé en conséquence. L'avantage de l'algorithme de Bresenham est qu'il ne nécessite pas de calculs de virgule flottante, ce qui le rend rapide et efficace.

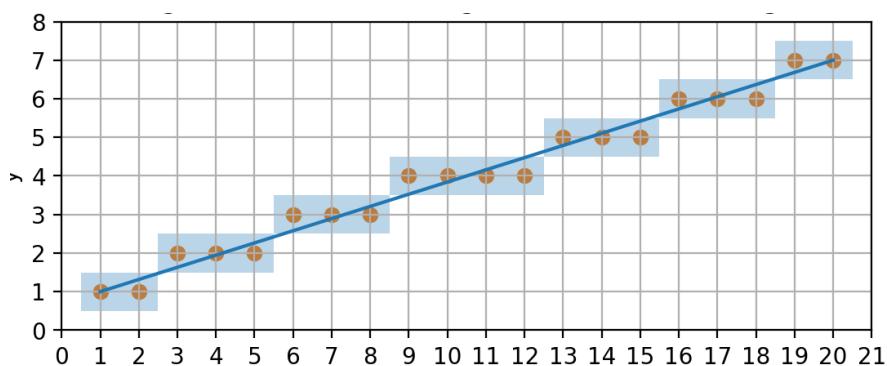


FIGURE 1 - Tracé d'un trait en utilisant l'algorithme de Bresenham.

En résumé, l'algorithme de Bresenham est un algorithme de traçage de lignes qui utilise un calcul incrémental pour dessiner des lignes de manière rapide et efficace en déterminant si chaque pixel doit être allumé ou non pour que la ligne apparaisse continue. Il est largement utilisé en informatique graphique pour dessiner des lignes sur un écran ou une image.



FIGURE 2 - Résultat de l'utilisation du pinceau.

3.1.2 Le seau de remplissage

L'outil Pot de Peinture, parfois également dénommé "seau", permet de modifier la couleur d'une zone en une autre en un seul clic. Ce dernier peut également permettre de remplir une zone à l'aide d'une texture. En l'occurrence, à l'état actuel, il ne permet que de remplir une zone avec une couleur. L'outil possède aussi un paramètre tolérance, permettant de spécifier dans quelles mesures les pixels doivent être sélectionnés. Pour implémenter le Pot de Peinture, on utilise l'algorithme "Flood Fill".

L'algorithme Flood Fill est un algorithme de remplissage utilisé en informatique graphique pour remplir une région close dans une image avec une couleur ou une texture. Il fonctionne en commençant par un point de départ (généralement un pixel) et en explorant les pixels adjacents jusqu'à ce que tous les pixels de la région soient remplis. L'algorithme examine chaque pixel adjacent à ceux qui ont déjà été remplis et décide s'il doit être rempli en fonction de certaines conditions. Cela peut être fait récursivement ou itérativement, selon l'implémentation choisie.

L'implémentation récursive de l'algorithme Flood Fill utilise la récursion pour remplir la région. À chaque étape, le pixel actuel est rempli et les pixels adjacents sont examinés pour voir s'ils doivent être remplis à leur tour. Cela se fait récursivement jusqu'à ce que tous les pixels de la région soient remplis. Cette implémentation est souvent plus simple à comprendre et à écrire, mais peut échouer si la profondeur de la récursion dépasse la limite maximale autorisée.

C'est la raison pour laquelle a été utilisée l'implémentation par file. Dans ce cas, le processus de remplissage fonctionne de manière similaire, mais les pixels sont ajoutés à la fin de la file et retirés du début de la file, ce qui crée un algorithme de largeur d'abord. À chaque étape, le pixel en début de file est retiré et rempli. Les pixels adjacents sont alors examinés pour voir s'ils doivent être ajoutés à la file. Si un pixel doit être rempli, il est ajouté à la fin de la file. Ce processus se poursuit jusqu'à ce que la file soit vide.

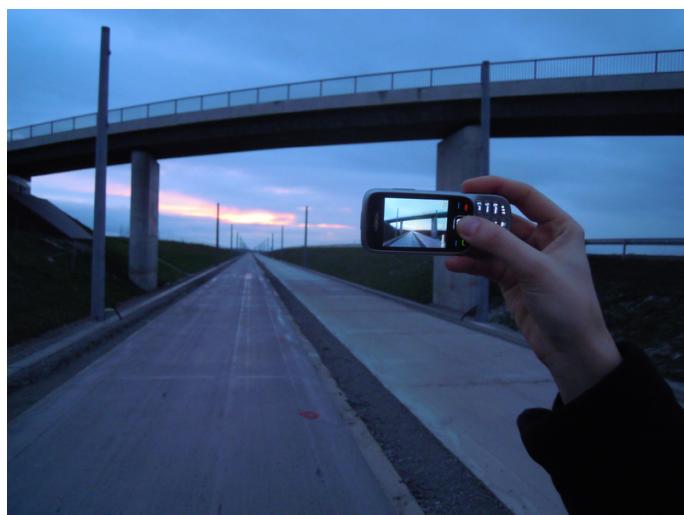


FIGURE 3 - Image de référence.



FIGURE 4 - Image de référence après utilisation du seau.

3.1.3 Palette de couleur

Les outils ne sont pas les seuls éléments indispensables à notre logiciel. En effet, il faut ajouter à cela une palette de couleur pour permettre à l'utilisateur de choisir les teintes qu'il veut utiliser sur la zone dessin.

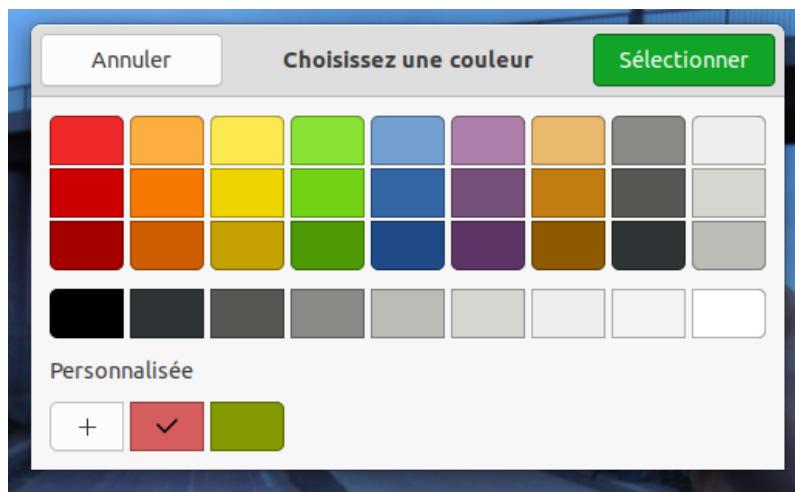


FIGURE 5 - Interface de la palette de couleur

En plus des couleurs prédéfinies, nous avons laissé le choix à l'utilisateur de pouvoir lui-même trouver la couleur qui lui conviendra parfaitement. Pour cela il peut utiliser un outil qui lui permet de choisir sur une teinte (faisant varier les valeurs R G B) puis de faire varier le niveau de noir et blanc.

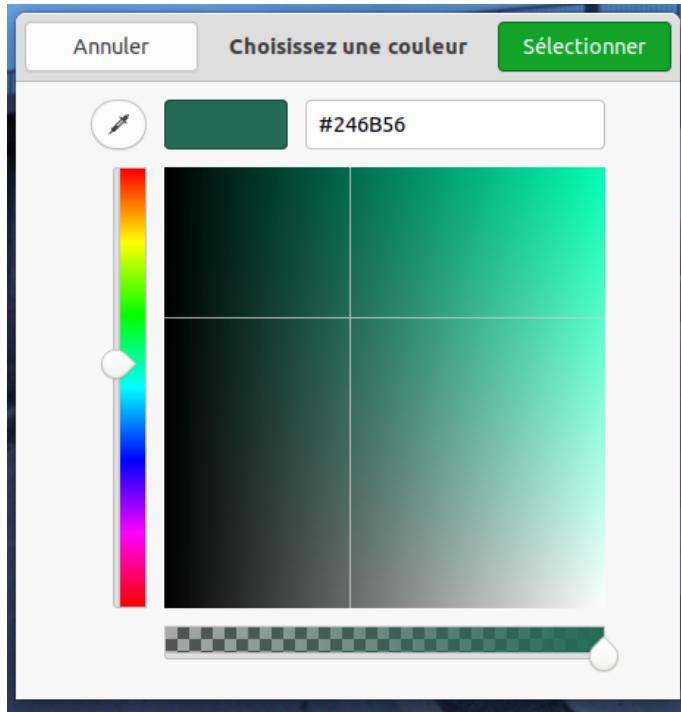


FIGURE 6 - Fenêtre de sélection d'une couleur personnalisée

Il peut aussi modifier l'indice de transparence de sa couleur mais aussi entrer directement la valeur de sa couleur en hexadécimal. En effet, chaque couleur peut-être symbolisé par un code de 6 caractères. Chaque doublet correspond respectivement au niveau de rouge, vert et bleu présent dans la couleur mais mis au format hexadécimal. Ce format permet alors de simplifié la représentation des couleurs de le rendre ainsi universel.

3.2 Les filtres

Évidemment, tout logiciel de traitement d'image ne serait pas complet s'il ne possédait pas une multitude de filtres permettant de modifier les images de manière prédéfinies. Un filtre de modification d'image est un algorithme utilisé en traitement d'image pour modifier les caractéristiques visuelles d'une image, telles que la couleur, la luminosité, le contraste, la netteté, etc. Les filtres de modification d'image sont largement utilisés en photographie, en design graphique et dans d'autres domaines pour améliorer la qualité d'une image ou en changer l'apparence.

Les filtres de modification d'image peuvent être implémentés de différentes manières, mais le principe général est de manipuler les valeurs des pixels d'une image en utilisant une formule mathématique spécifique. Par exemple, un filtre de luminosité ajustera la valeur des pixels d'une image pour augmenter ou diminuer la luminosité globale de l'image. Un filtre de flou généralement utilisera une moyenne pondérée des pixels autour d'un pixel donné pour lisser les bords et les transitions dans l'image.

3.2.1 Filtre Nuances de Gris : "Grayscale"

Pour convertir une image infographique couleur en niveau de gris il faut remplacer, pour chaque pixel les trois valeurs représentant les niveaux de rouge, de vert et de bleu, par une seule valeur représentant la luminosité. Pour cela, l'utilisation d'une formule mathématique calculant la moyenne des composantes rouges, vertes et bleues de chaque pixel est nécessaire. Celle-ci fut définie de la manière suivante :

$$gray = 0.4 * red + 0.35 * green + 0.25 * blue$$

FIGURE 7 : formule de calcul du niveau de gris.

Ces coefficients spécifiques sont utilisés afin de quand même rendre l'image visible vis-à-vis de l'utilisateur.



FIGURE 8 - Application du filtre nuance de gris

3.2.2 Filtre Sépia

Le filtre sépia est un filtre de modification d'image qui donne à une image une apparence d'antan en utilisant des couleurs brunes et dorées. Cette technique a été utilisée à l'origine pour simuler l'apparence des images en noir et blanc sur du papier photo jauni avec le temps.

Pour appliquer un filtre sépia à une image, les valeurs de couleur de chaque pixel sont modifiées pour donner une apparence d'image ancienne. Cela peut impliquer la modification des valeurs de rouge, de vert et de bleu pour produire des couleurs brunes et dorées. Par exemple, pour chaque pixel, la valeur de rouge peut être augmentée pour donner une apparence dorée, tandis que la valeur de bleu peut être diminuée pour donner une apparence brune.

Le filtre sépia est souvent utilisé pour donner une apparence vintage à des photos récentes. Il peut également être utilisé pour simuler l'apparence des photos anciennes

ou pour donner une apparence distincte à une image. Il est disponible dans de nombreux logiciels de traitement d'image, y compris les applications en ligne, les applications mobiles et les logiciels de bureau.

La fonction de conversion en pixel sépia est alors définie telle que :

$$\text{newRed} = \min(255, 0.393 * \text{red} + 0.769 * \text{green} + 0.189 * \text{blue})$$

$$\text{newGreen} = \min(255, 0.349 * \text{red} + 0.686 * \text{green} + 0.168 * \text{blue})$$

$$\text{newBlue} = \min(255, 0.272 * \text{red} + 0.534 * \text{green} + 0.131 * \text{blue})$$

FIGURE 9 : formule de calcul du niveau de gris.



FIGURE 10 - Application du filtre Sepia

3.2.3 Filtre Gaussien

Le filtre flou gaussien est un filtre de modification d'image qui est utilisé pour adoucir les détails d'une image et lisser les transitions entre les couleurs et les textures. Il est particulièrement utile pour supprimer les bruits indésirables dans une image, tels que le bruit numérique ou le grain filmique.

Le filtre flou gaussien est basé sur la théorie statistique et utilise une fonction de densité gaussienne pour déterminer la pondération des pixels voisins d'un pixel donné. En d'autres termes, pour chaque pixel, la valeur moyenne des pixels environnants est calculée en utilisant des coefficients pondérés pour déterminer la valeur finale du pixel. Les coefficients pondérés sont déterminés en utilisant une fonction gaussienne pour donner plus de poids aux pixels les plus proches du pixel en question et moins de poids aux pixels plus éloignés.

Le filtre flou gaussien est un filtre de flou linéaire qui peut être utilisé pour flouter une image entière ou des parties sélectionnées de l'image. Il peut également être utilisé en combinaison avec d'autres filtres pour produire des effets plus complexes.

Il est disponible dans de nombreux logiciels de traitement d'image, y compris les applications en ligne, les applications mobiles et les logiciels de bureau.

$$(R, G, B) = \left(\frac{\sum_{n=0}^n r}{n}, \frac{\sum_{n=0}^n g}{n}, \frac{\sum_{n=0}^n b}{n} \right)$$

FIGURE 11 : formule de calcul du flou gaussien.

Avec n correspondant au nombre de pixel choisi dans les paramètres de la fonction, le rayon du carré autour du pixel concerné.

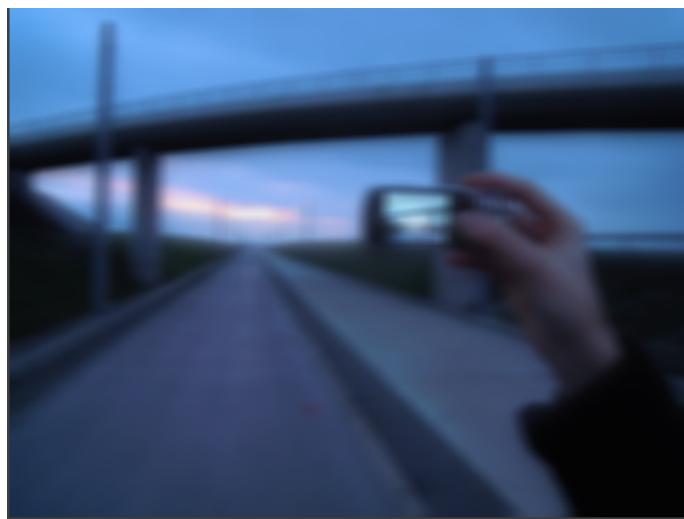


FIGURE 12 - Application du flou Gaussien

3.2.4 Filtre Rouge

Le filtre de lumière rouge est un filtre de modification d'image qui accentue la composante rouge de chaque pixel dans une image. Cela signifie que les parties de l'image qui étaient déjà rouges deviennent plus rouges, tandis que les autres parties sont modifiées pour paraître plus rouges.

L'application du filtre de lumière rouge est réalisée en augmentant la valeur de la composante rouge de chaque pixel, tout en laissant les autres composantes inchangées ou en les modifiant moins. Cela peut être réalisé en multipliant la composante rouge par un facteur d'amplification, tout en maintenant les autres composantes inchangées ou en les multipliant par un facteur plus faible.

Le filtre de lumière rouge peut être utilisé pour une variété de raisons, telles que la correction de couleurs pour les images en sous-exposition ou la visualisation de structures en relief dans les images médicales. Il peut également être utilisé pour des raisons esthétiques, pour ajouter une touche de couleur rouge à une image.



FIGURE 13 - Application du filtre rouge

3.2.5 Filtre Binaire et Otsu (noir et blanc)

Le thresholding d'image est une technique de traitement d'image qui consiste à convertir une image en noir et blanc en se basant sur une valeur de seuil. Cette valeur de seuil détermine les pixels de l'image qui seront considérés comme blancs et ceux qui seront considérés comme noirs.

Lors du thresholding, la luminosité de chaque pixel est comparée à la valeur de seuil. Si la luminosité d'un pixel est supérieure à la valeur de seuil, ce pixel est considéré comme blanc. Sinon, il est considéré comme noir. En fin de compte, cela permet de segmenter l'image en deux parties distinctes : les zones blanches et les zones noires.

Le thresholding d'image est utilisé pour un certain nombre d'applications, telles que la reconnaissance de formes, la numérisation de documents, la segmentation de l'image et la détection de bords. Il est également souvent utilisé en combinaison avec d'autres techniques de traitement d'image pour résoudre des problèmes plus complexes.

Il existe plusieurs méthodes pour déterminer la valeur de seuil, telles que le thresholding global, le thresholding local et le thresholding adaptatif. Chacun de ces méthodes a ses propres avantages et inconvénients en fonction de la qualité de l'image d'entrée et de la complexité du problème à résoudre. Le choix de la méthode dépend donc des besoins spécifiques de chaque application.

En l'occurrence, ont été implémentées une version manuelle, mais également une version automatique du thresholding basé sur les formules d'Otsu.

La valeur de seuil n'est pas fixe. Elle dépend de l'image traitée. Le calcul consiste à prendre les valeurs des pixels voisins à notre pixel traités afin de calculer la valeur de t tel que la variance reste identique le plus souvent, puis nous renvoyons cette valeur comme étant le seuil. Deux méthodes nous permettent de calculer cela :

$$\sigma^2(t) = \omega_{bg}(t)\sigma_{bg}^2(t) + \omega_{fg}(t)\sigma_{fg}^2(t)$$

FIGURE 14 : formule probabiliste de la variance.

$$\sigma^2(t) = \frac{\sum(x_i - \bar{x})^2}{N - 1}$$

FIGURE 15 : formule par somme de la variance.

Il est ensuite possible de calculer la variance pour différentes valeurs de t. La variance sera alors seuillée étant donnée que pour un certain $t > \text{threshold}$, celle-ci ne changera plus. Le t deviendra alors le seuil à utiliser dans le cadre de la binarisation. Cependant, il est important de noter que la binarisation d’Otsu ne fonctionne pas avec toutes les images.

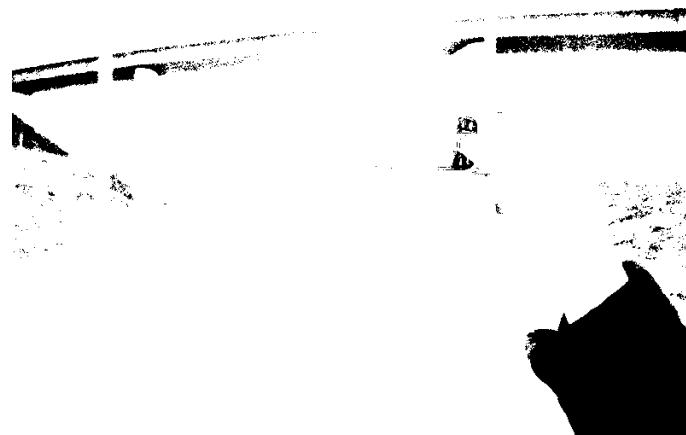


FIGURE 16 - Application du filtre binaire(noir ou blanc)

3.2.6 Filtre Binaire (colorisé)

Fonctionnant exactement de la même manière que pour le filtre Binaire classique, ce dernier permet de spécifier deux autres couleurs, au lieu du noir et du blanc.

3.2.7 Filtre Négatif ou Inversion des couleurs

Le filtre d’inversion des couleurs est un filtre de modification d’image qui inverse les couleurs d’une image. Cela signifie que les couleurs claires deviennent sombres, et vice versa.

L'inversion des couleurs est réalisée en inversant les valeurs des composantes de couleur de chaque pixel. Par exemple, si un pixel était originalement de couleur rouge, il deviendrait vert après l'inversion des couleurs. Les valeurs de chaque composante de couleur peuvent être inversées en les soustrayant de leur valeur maximale (par exemple, 255 pour les couleurs codées sur 8 bits).

Le filtre d'inversion des couleurs est souvent utilisé pour des raisons esthétiques ou pour des applications spécifiques, telles que la correction de couleurs inversées dans des images ou la visualisation de structures en relief dans des images médicales.

$$newRed = 255 - red$$

$$newGreen = 255 - green$$

$$newBlue = 255 - blue$$

FIGURE 17 : formule de calcul des valeurs R, G, B inversées.



FIGURE 18 - Application du filtre négatif

3.2.8 Filtre Gamma

Le filtre d'augmentation du gamma est un filtre de modification d'image qui permet de contrôler la luminosité d'une image. Il est souvent utilisé pour corriger l'exposition d'une image, pour la rendre plus sombre ou plus claire selon les besoins.

Le gamma est une valeur qui détermine comment les valeurs de luminosité d'une image sont interprétées et affichées. Une augmentation du gamma signifie qu'il y aura une amplification des valeurs de luminosité plus faibles et une atténuation des valeurs de luminosité plus élevées. Inversement, une diminution du gamma aura pour effet d'atténuer les valeurs de luminosité plus faibles et d'amplifier les valeurs de luminosité plus élevées.

L'augmentation du gamma peut être réglée en utilisant un curseur ou en entrée une valeur numérique spécifique. Il est important de noter que des valeurs de gamma extrêmement élevées ou extrêmement faibles peuvent rendre l'image peu naturelle ou même illisible. Par conséquent, il est souvent nécessaire de trouver un équilibre entre l'amélioration de la luminosité de l'image et la préservation de son aspect naturel.

L'augmentation du gamma se fait par l'intermédiaire d'une fonction $g(x)$ définie en l'occurrence par :

$$g(x) = 255 * \left(\frac{x}{255}\right)^n$$

FIGURE 19 : formule de l'augmentation du gamma.

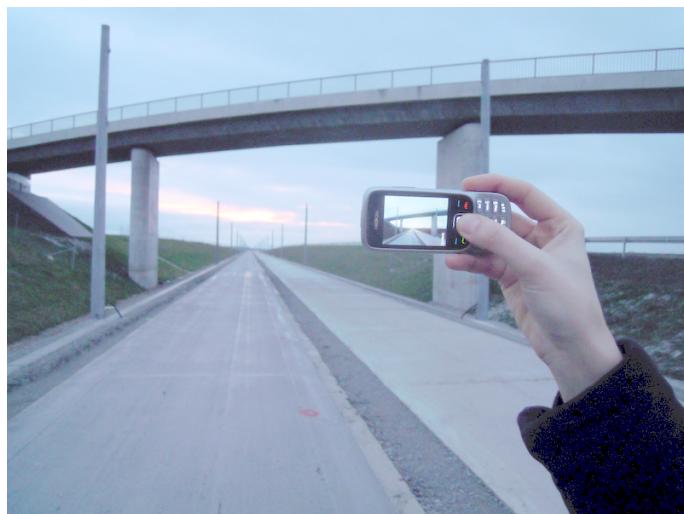


FIGURE 20 - Application du filtre gamma

3.3 Les formes géométriques

L'implémentation des formes géométriques dans les logiciels d'édition d'image est extrêmement importante car elle permet de dessiner des formes précises et régulières qui seraient difficiles à créer à la main. Les formes géométriques telles que les cercles, les carrés, les rectangles, les triangles, les ellipses, les pentagones, etc. sont des éléments fondamentaux dans la création d'images et de graphiques.

En utilisant des formes géométriques, les artistes et les graphistes peuvent créer des images qui sont parfaitement symétriques et proportionnées. Les formes géométriques sont également utilisées pour créer des graphiques, des diagrammes et des illustrations techniques qui nécessitent des formes précises pour transmettre des informations.

Les logiciels d'édition d'image permettent aux utilisateurs de dessiner des formes géométriques en utilisant des outils tels que le rectangle, le cercle, le polygone,

l'ellipse, etc. Les formes géométriques sont souvent utilisées comme base pour créer des illustrations plus complexes en ajoutant des détails et des textures.

De plus, l'utilisation de formes géométriques dans les logiciels d'édition d'image permet d'effectuer des opérations telles que le redimensionnement, le rognage, le déplacement et la rotation des formes en toute facilité. Cela permet aux utilisateurs de modifier facilement et rapidement la forme et la taille de leurs dessins, ce qui est particulièrement important dans les projets qui nécessitent une grande précision.

Enfin, l'utilisation de formes géométriques dans les logiciels d'édition d'image est également importante pour la création de logos, de bannières et d'autres éléments visuels destinés à la publicité et à la promotion. Les formes géométriques peuvent être utilisées pour créer des designs simples mais impactants qui peuvent être facilement reconnaissables et mémorisables.

En résumé, l'implémentation des formes géométriques dans les logiciels d'édition d'image est essentielle pour les artistes, les graphistes et les designers car elle leur permet de créer des images précises et régulières, de modifier facilement la forme et la taille de leurs dessins et de créer des designs simples mais impactants pour la publicité et la promotion.

3.3.1 La ligne

Une ligne est un élément de base de la géométrie qui est défini comme une succession finie de points qui se suivent dans une même direction. En l'occurrence, elle est caractérisée par son épaisseur, mais également par sa longueur et son orientation. Les lignes sont utilisées pour représenter des formes et des objets dans les dessins techniques, les graphiques et les schémas. Elles peuvent également être utilisées pour délimiter des zones et créer des motifs dans les designs artistiques. Comme indiqué dans les premières parties de ce document, c'est encore une fois l'algorithme de Bresenham qui est utilisé afin de tracer une ligne.

Pour tracer une ligne en utilisant l'interface, il faut tout d'abord sélectionner l'outil ligne, puis ensuite choisir les deux points correspondants au début et à la fin de la ligne.

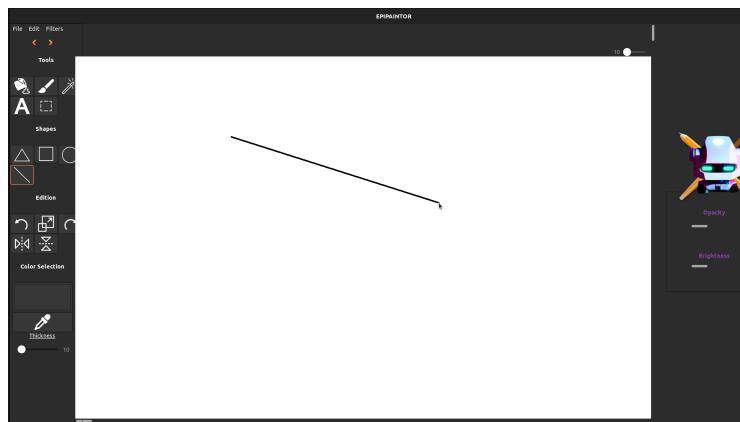


FIGURE 21 - Utilisation de l'outil traçage de ligne

3.3.2 Le rectangle (et le carré !)

Le rectangle est une figure géométrique qui est définie par quatre angles droits et quatre côtés. Les côtés opposés d'un rectangle ont la même longueur, tandis que les côtés adjacents ont des longueurs différentes. Les rectangles sont souvent utilisés pour représenter des objets dans les dessins techniques et les plans architecturaux, ainsi que dans les designs artistiques.

Le carré est une forme géométrique spéciale qui est définie par quatre côtés de même longueur et quatre angles droits. Le carré est donc un type particulier de rectangle où tous les côtés ont la même longueur. Les carrés sont également largement utilisés dans les dessins techniques et architecturaux, ainsi que dans les designs artistiques pour leur simplicité et leur équilibre visuel.

En l'occurrence, un rectangle peut être dessiné en sélectionnant l'outil rectangle, puis en cliquant sur deux points distincts. Les calculs seront automatiquement fait par le programme afin de relier les points de manière rectangulaire. Il est également intéressant de noter que cette même technique permet l'affichage du rectangle de sélection qui sera implémenté dans la prochaine mise à jour.

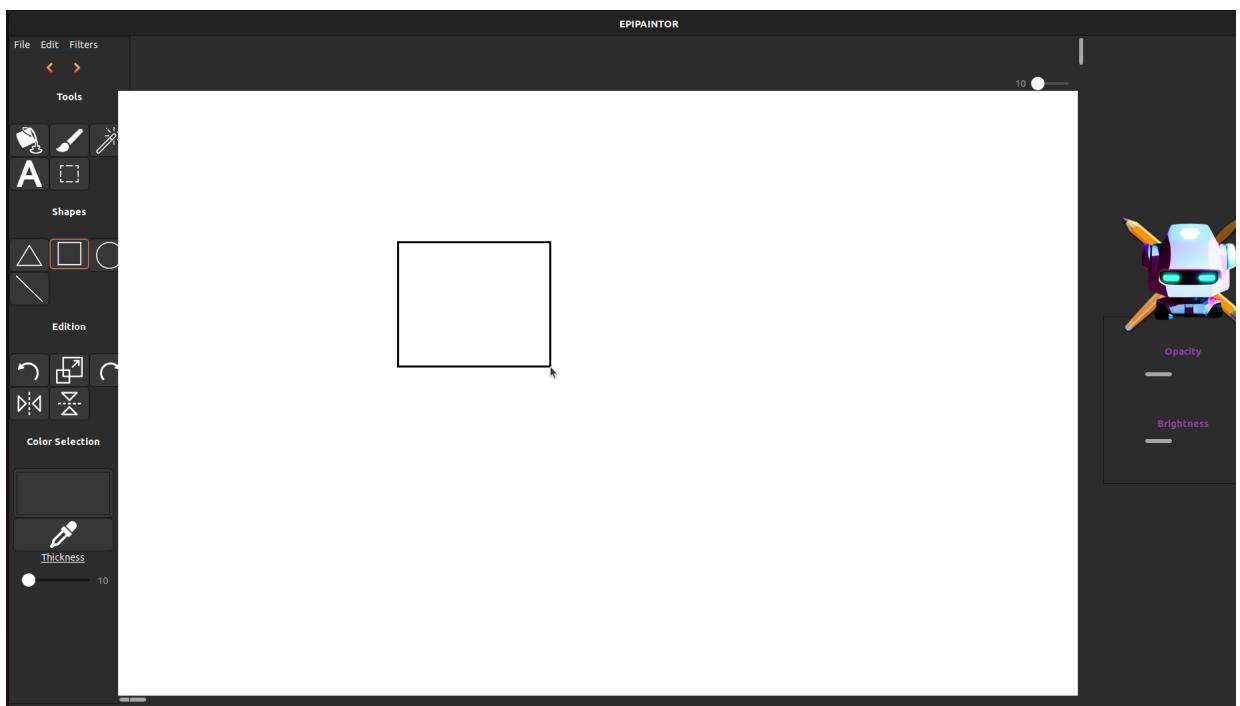


FIGURE 22 - Utilisation de l'outil traçage de rectangle

3.3.3 Le triangle

Le triangle est une figure géométrique composée de trois côtés et de trois angles. Il est souvent utilisé dans les dessins techniques, les plans architecturaux, les graphiques et les schémas pour représenter des objets et des formes. Les triangles

peuvent avoir différentes formes et tailles, et sont caractérisés par des propriétés telles que leur base, leur hauteur, leurs angles intérieurs et extérieurs, ainsi que leur aire et leur périmètre. Les triangles sont également utilisés dans les designs artistiques pour créer des formes géométriques intéressantes et des motifs abstraits.

Pour le moment, seul le dessin de triangle isocèle est possible. Pour cela, il faut choisir deux points et le triangle sera automatiquement dessiné. Plus tard, l'ajout du tracé de triangle quelconque sera évidemment ajouté.

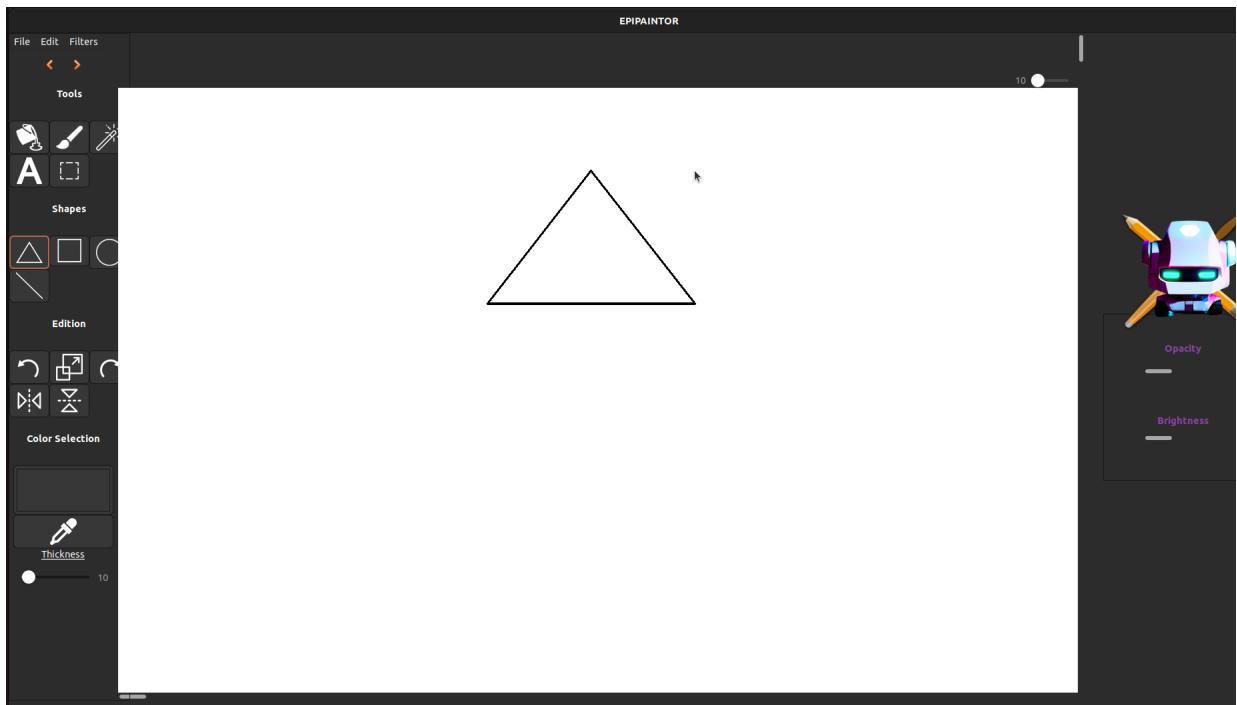


FIGURE 23 - Utilisation de l'outil traçage de triangle

3.3.4 Le cercle

Le cercle est une figure géométrique qui représente l'ensemble des points situés à une même distance d'un point central appelé le centre. Le cercle est caractérisé par son rayon (la distance entre le centre et n'importe quel point du cercle) et son diamètre (la distance entre deux points du cercle qui passent par le centre). Les cercles sont couramment utilisés dans les dessins techniques, les plans d'architecture, les graphiques et les schémas pour représenter des objets ronds ou pour définir des zones de distance égale autour d'un point. Les cercles sont également utilisés dans les designs artistiques pour créer des formes abstraites et pour ajouter des éléments de texture et de mouvement à une composition.

Il fallait, là aussi, utiliser une variante de l'algorithme de Bresenham, permettant de tracer des cercles et plus généralement des figures circulaires. L'algorithme de tracé de cercle de Bresenham est une méthode couramment utilisée pour dessiner des cercles à l'aide d'un ordinateur. Cette méthode permet de calculer rapidement les coordonnées des points d'un cercle en utilisant des opérations arithmétiques simples.

L'algorithme de Bresenham commence par définir le centre du cercle et le rayon du cercle. Ensuite, il utilise une boucle pour calculer les coordonnées des points du cercle en fonction de l'angle de rotation.

À chaque itération de la boucle, l'algorithme calcule les coordonnées x et y d'un point sur le cercle en utilisant l'équation d'un cercle et en arrondissant les résultats aux entiers les plus proches. Il utilise ensuite ces coordonnées pour dessiner le cercle en reliant les points à l'aide de lignes droites.

L'algorithme de Bresenham est très efficace car il utilise des opérations entières simples au lieu de fonctions trigonométriques plus coûteuses en termes de temps de traitement. Il est également précis et peut être adapté pour dessiner des cercles de différentes tailles et formes en modifiant les paramètres de l'algorithme.

Afin de pouvoir tracer un cercle, il suffit de sélectionner l'outil traçage de cercle, puis de cliquer sur le point défini comme centre et par la suite un point du cercle (c'est-à-dire à la fin du rayon).

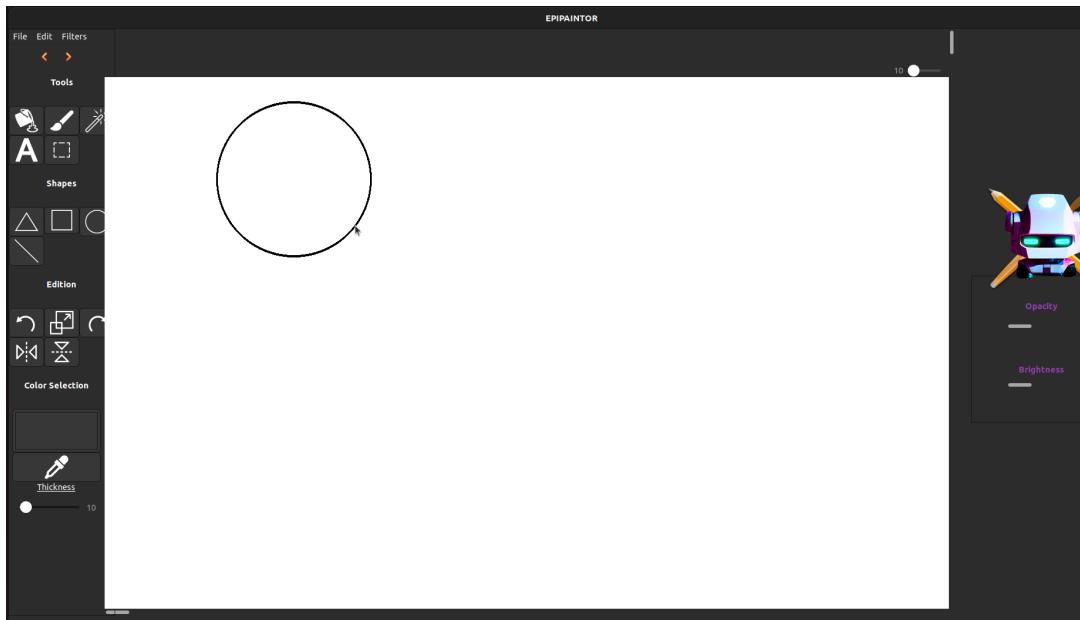


FIGURE 24 - Utilisation de l'outil traçage de cercle

3.3.5 Prévisualisation des formes

La prévisualisation des formes géométriques est une fonctionnalité importante dans les outils d'édition d'images car elle permet à l'utilisateur de voir l'effet d'une forme ou d'une transformation avant de l'appliquer définitivement sur l'image. Cette prévisualisation en temps réel permet d'affiner les ajustements et de les rendre plus précis, ce qui peut faire gagner du temps et éviter des erreurs coûteuses.

Lorsque l'utilisateur sélectionne une forme géométrique, comme un cercle ou un rectangle, il peut prévisualiser sa taille, sa position et son orientation sur l'image en temps réel avant de la dessiner. De même, lorsque l'utilisateur applique une transformation sur une forme géométrique, comme une rotation ou un redimensionnement,

il peut voir l'effet de cette transformation en temps réel et ajuster les paramètres jusqu'à ce que le résultat souhaité soit atteint.

En offrant une prévisualisation en temps réel, les outils d'édition d'images permettent à l'utilisateur de travailler de manière plus intuitive et créative. Cela peut également aider à réduire les erreurs et les frustrations, car l'utilisateur peut voir l'effet de chaque ajustement avant de le valider. En fin de compte, la prévisualisation des formes géométriques est un élément clé pour rendre les outils d'édition d'images plus conviviaux et plus efficaces pour les utilisateurs de tous niveaux de compétence.

Contrairement à ce qui pourrait être pensé, la prévisualisation des formes est en réalité un concept assez compliqué à mettre en place. En effet, cette dernière nécessite le calcul continu de nouvelles valeurs tant que la souris est utilisé et que cette dernière bouge.

En l'occurrence, la prévisualisation des formes a été implémentées pour absolument chacune des formes, comme en témoignent les images ci-dessus dans lesquelles on peut voir la souris toujours sur la forme. Pour cela, on instancie le signal de détection de mouvement de la souris, puis on sauvegarde une copie de l'image à chacun des mouvements, qu'on replace par une nouvelle version de la forme juste après.

3.4 L'interface

Pour que notre projet soit réellement utilisable et qu'il paraisse moins austère pour les utilisateurs lambda nous avons choisi de réaliser une interface avec la librairie GTK3 en l'associant à l'outil Glade.

La librairie GTK est composée d'un grand nombre de widget qui sont utilisables pour pouvoir modéliser notre interface. Pour pouvoir rendre l'utilisation de GTK plus facile et surtout beaucoup plus visuel le logiciel Glade nous permet de désigner en temps réel le visuel de notre interface.

En effet, Glade permet de générer des fichiers d'interface en .glade qui sont utilisables par GTK. Dans notre code principal il faut penser à bien définir chaque widget avec un nom, qui devra alors être indiqué sur GTK pour ensuite pouvoir les utiliser pour lancer des fonctions par exemple.

Pour pouvoir lancer des fonctions, nous utilisons certains signaux (cliques simples par exemple ou alors clique maintenue) qui peuvent être détectés et utiliser pour des actions précises.

Ces deux outils utilisés ensemble permettant de réaliser des interfaces épurées et simples idéales pour ce genre de projet où le côté pratique est plus important que le visuel.

3.4.1 Vue d'ensemble

Tout d'abord, voici une vue d'ensemble de l'interface :

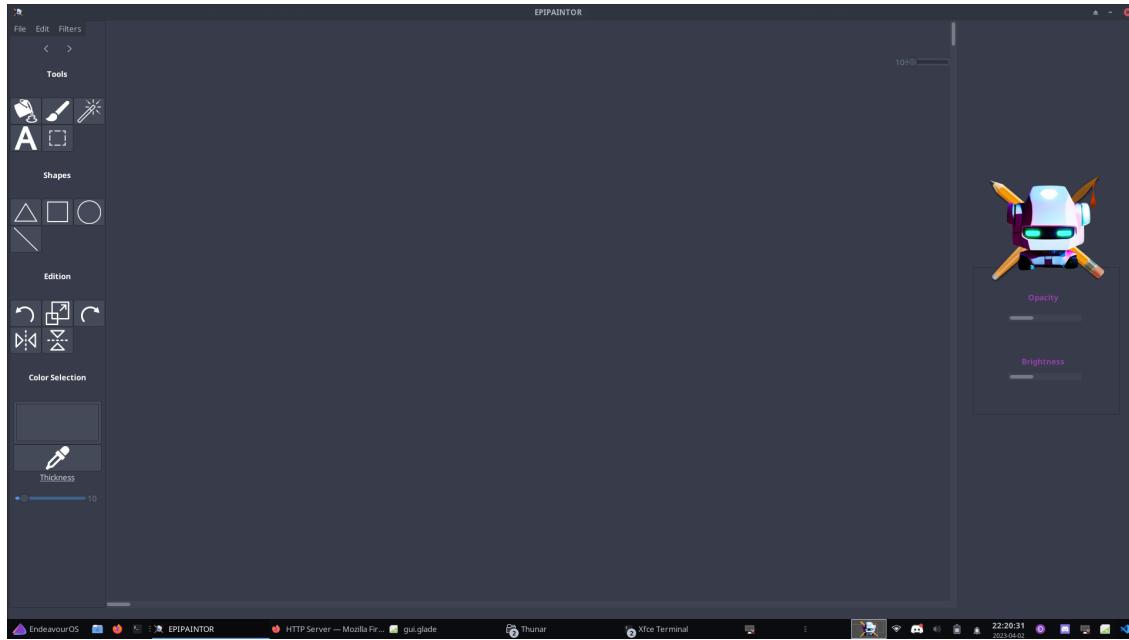


FIGURE 25 - Interface

Notre logiciel possède une interface intuitive et donc simple à comprendre pour n'importe quel utilisateur. Elle se compose :

D'une barre latéral gauche qui contient tous les outils disponibles ainsi que le choix de la couleur.



FIGURE 26 - Barre latérale gauche (outils)

Sur le coin supérieur gauche on retrouve 3 sous-menus. Ces derniers permettent

d'afficher plus d'options lorsque l'on passe la souris dessus ce qui permet de gagner l'espace sur la zone de dessin.

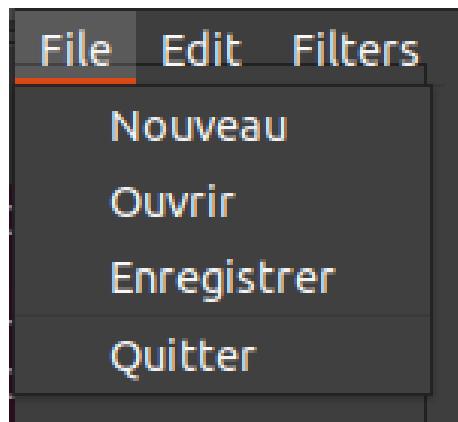


FIGURE 27 - Menu "File"

File : ce menu permet d'ouvrir un image présente dans les fichiers de l'utilisateur, enregistrer dans le fichier "save" la production faite et quitter l'application. Il permet aussi grâce au bouton "Nouveau" d'ouvrir une nouvelle zone de dessin

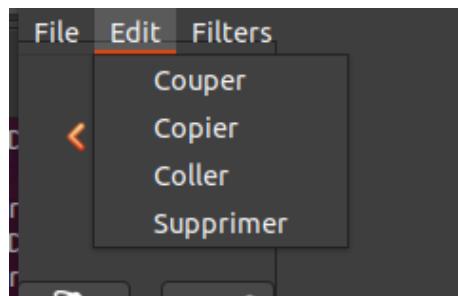


FIGURE 28 - Menu "Edit"

Edit : il permettra de couper, copier, coller ou supprimer la zone sélectionnée par l'outil de selection ou baguette magique.

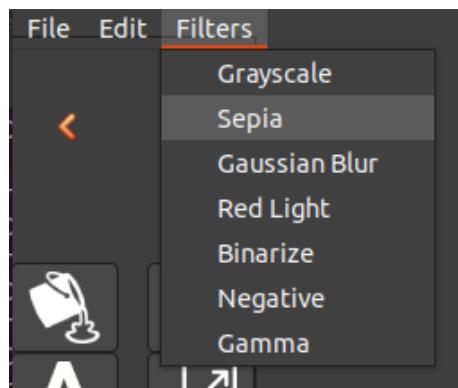


FIGURE 29 - Menu "Filters"

Filters : Il permet d'appliquer à toute la zone de dessin les différents filtres proposés. Pour chacun des filtres proposés, l'utilisateur a l'option de changer les inputs correspondants aux paramètres des filtres.

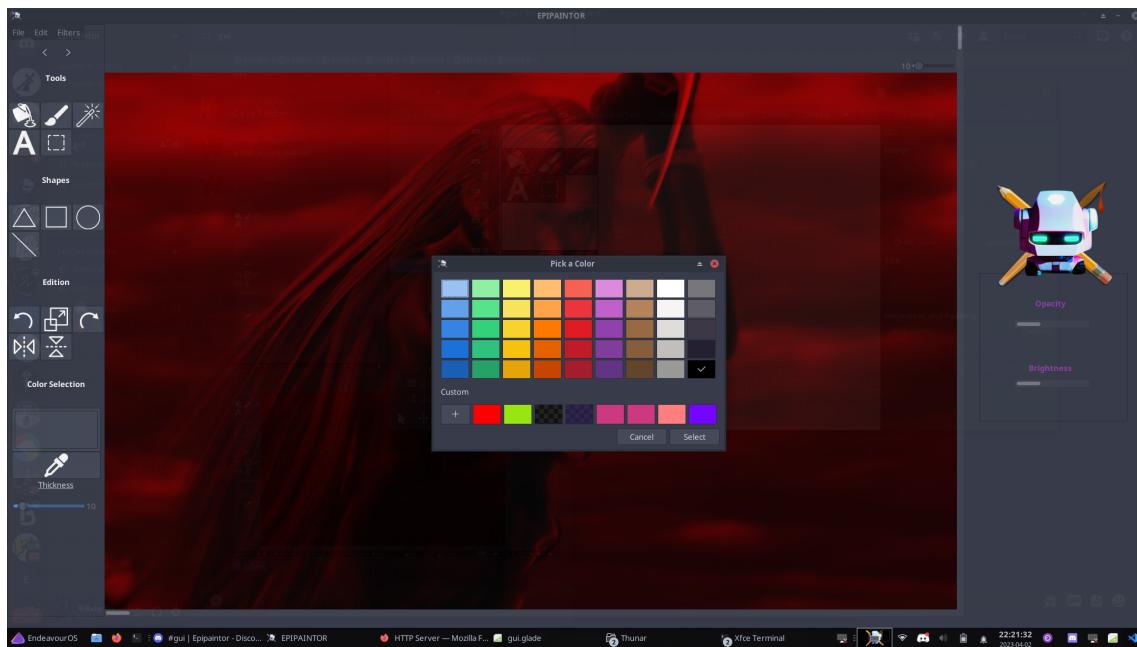


FIGURE 30 - Sélection de couleurs

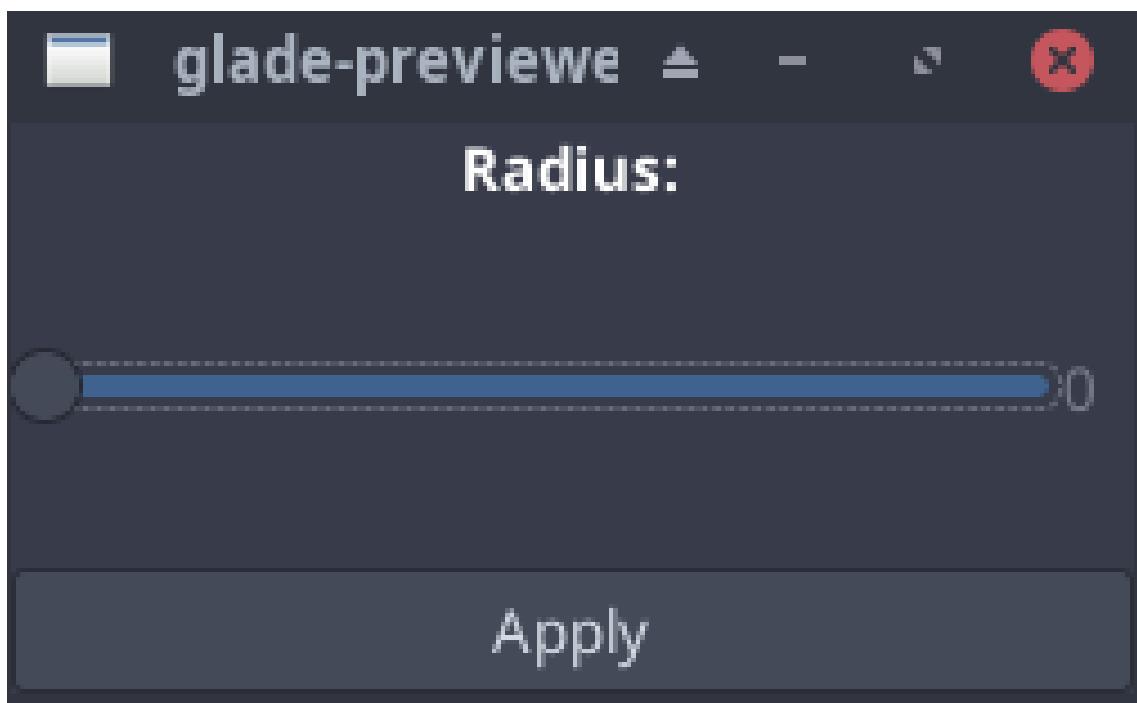


FIGURE 31 - Choix du rayon dans le cas du filtre de gauss.

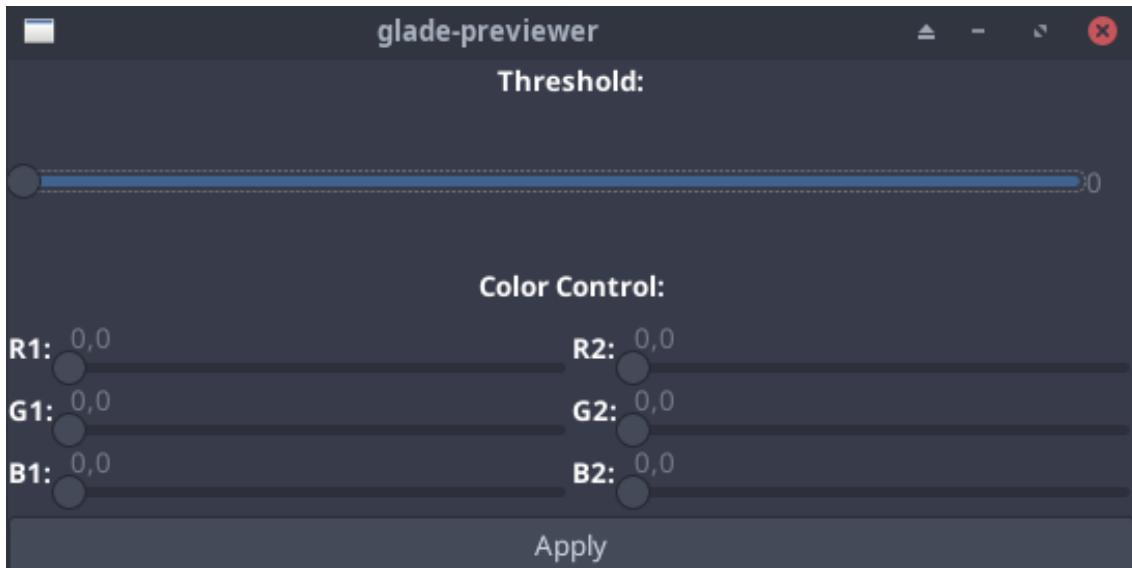


FIGURE 32 - Choix des options pour le filtre binaire colorisé.

Enfin, ci-dessous un exemple avant-après de l'utilisation du filtre binaire colorisé sur une image.



FIGURE 33 - Interface avec image avant utilisation du filtre.

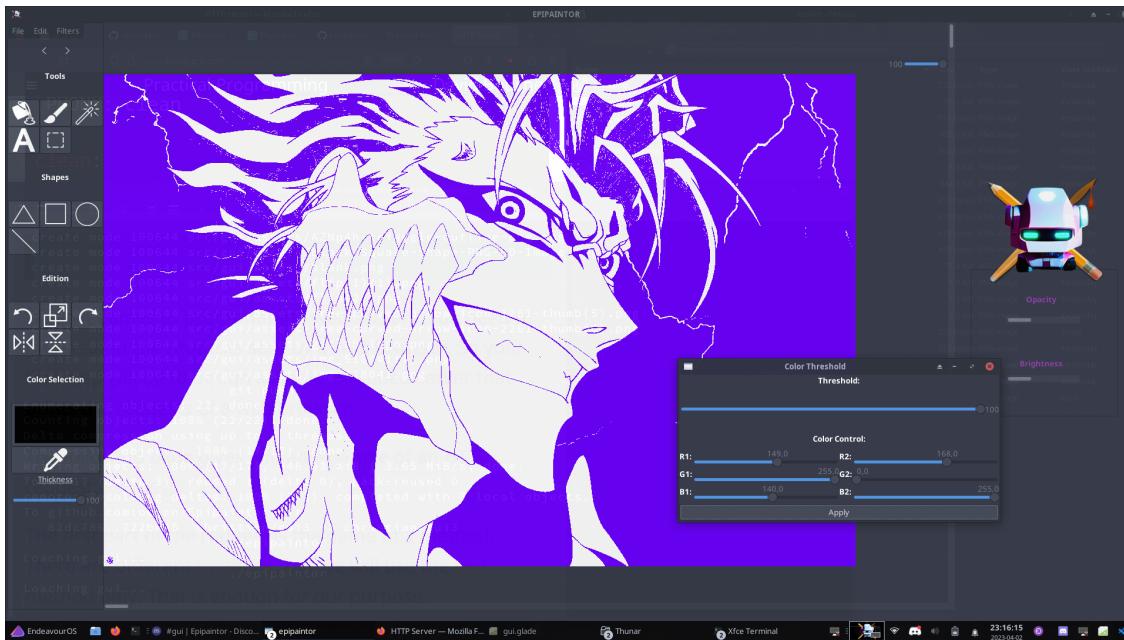


FIGURE 34 - Interface avec image après utilisation du filtre.

3.5 Le retour en arrière / avant

Le retour en arrière et en avant est un outil essentiel dans les logiciels d'édition d'images. Cela permet à l'utilisateur de revenir en arrière sur les modifications apportées à une image afin de corriger une erreur ou d'essayer une autre approche, puis de revenir en avant pour revenir aux modifications précédentes.

Le retour en arrière et en avant fonctionne en enregistrant chaque action effectuée sur l'image dans l'historique. Chaque fois que l'utilisateur effectue une modification, celle-ci est ajoutée à la fin de l'historique. En cliquant sur le bouton "retour en arrière", l'utilisateur peut annuler la dernière modification effectuée et revenir à l'état précédent de l'image. En cliquant sur le bouton "retour en avant", l'utilisateur peut revenir à l'état précédent de l'image, après avoir utilisé le bouton "retour en arrière".

Cet outil est particulièrement utile dans les situations où l'utilisateur souhaite expérimenter différentes approches d'édition sans avoir à enregistrer chaque étape comme une image séparée. Il permet également de revenir facilement en arrière en cas d'erreur ou de modification indésirable.



FIGURE 35 - Boutons de retour en arrière / avant

Afin d'implémenter cette fonctionnalité, il a fallu mettre en place encore une nouvelle structure de données : la pile (en anglais : stack). En effet, le retour en arrière ou en avant ne correspond en réalité qu'à l'empilement ou le dépilement de l'état précédent de la zone de dessin, sauvegardée à chaque nouvelle action.

3.6 Site Web

Le site web a été réalisé avec un total de 3 langages : l'html, le CSS et le javascript. Cette combinaison des 3 langages permet la réalisation et la confection d'un site web très complet, moderne et vivant. En effet, chaque langage permet de faire quelque chose de précis et lorsque l'on met bout à bout les différents fichiers qui composent notre site, on obtient un meilleur résultat qu'en utilisant uniquement de l'HTML.

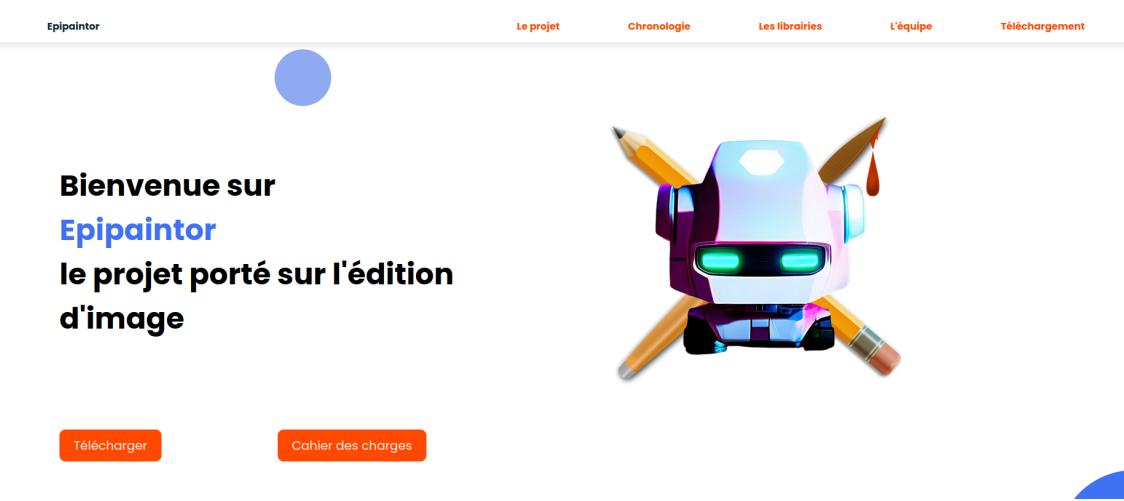


FIGURE 36 - En-tête du site

3.6.1 HTML : La base du site

HTML signifie « HyperText Markup Langage » cela se traduit en français par « langage de balises pour l'hypertexte ». Il est utilisé afin de représenter et créer le contenu d'une page web et sa structure. Le langage HTML utilise un système de « balises » pour annoter du texte, des images et d'autres contenus afin de les afficher dans un navigateur web. Le balisage HTML comprend des « éléments » spéciaux tels que <head>, <title>, <body>, <header>, <footer> et des dizaines d'autres.

Ces balises permettant de spécifier à quelle partie de la page un élément appartient. Par exemple, la balise « header » permet de spécifier que tous les éléments compris dans cet endroit du code appartiennent à l'en-tête de la page. Il est très important de bien « baliser » son code afin de pouvoir plus facilement appliquer du style à ça page via le CSS.

Une page HTML peut être vue comme une disposition de boîtes qui contiennent des éléments utiles à construction de la page. En effet, pour pouvoir délimiter chaque « boîte » de la page nous utilisons la balise <div> qui peut être couplée avec l'attribut « class ». Cet attribut est très pratique car il permet de spécifier l'utilité de chaque boîte. Bien évidemment il n'est possible et même courant que plusieurs boîtes soient les unes dans les autres.

3.6.2 Le CSS : application du style

Tout d'abord CSS signifie Cascading Style Sheets (feuilles de style en cascade). Comme nous avons dit précédemment, une page web se compose de plusieurs boîtes ou parties avec des utilités bien particulières. Le CSS permet donc d'appliquer et de changer divers paramètres de manière spécifique sur chaque partie de la page.

En effet, il est possible de spécifier pour chaque classe définie dans notre page HTML ou pour chaque balise des caractéristiques telles que pour les plus utilisées : la taille, la hauteur et largeur des marges, la couleur de la police, la couleur du fond... C'est pour cela qu'il est très important que chaque balise classe de notre page ait un sens sémantique.

Une balise ou classe sémantique est une balise HTML qui apporte une indication sur le contenu d'une page web. Par exemple, la balise « article » doit être utilisée sur les articles de la page, pour que par la suite dans notre fichier CSS nous pourrons instaurer un style commun à tous les articles et ainsi obtenir une page plus homogène visuellement.

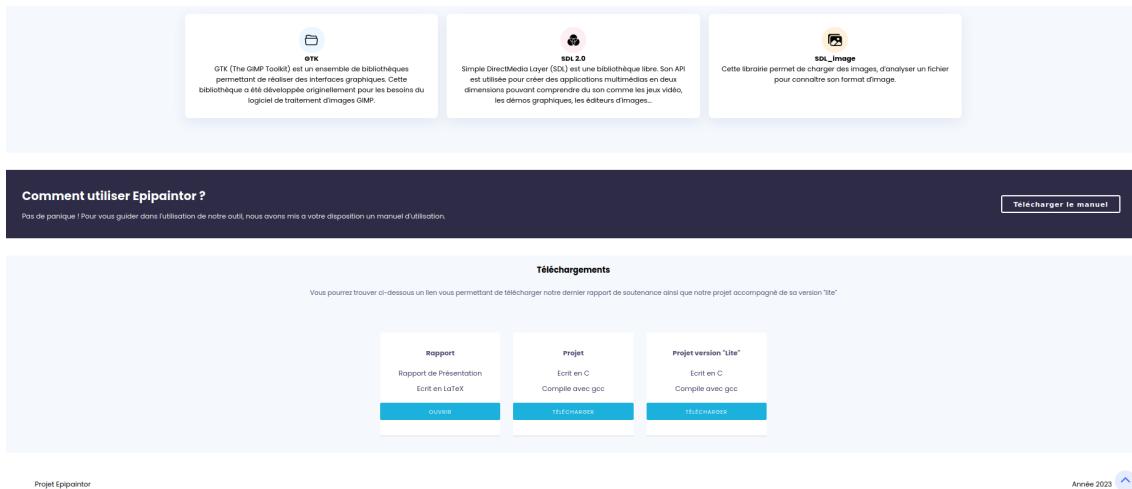


FIGURE 37 - Partie du site

3.6.3 Le Javascript : ajout de détail

Le Java est un langage de programmation. Il sert à créer des programmes, sur le web il permet l'intégration de contenu dynamique : il donne des instructions à l'ordinateur ou au serveur afin qu'il fasse certaines actions ; il "parle" donc avec les machines.

Sur notre page web il va nous permettre d'ajouter des animations à la page pour la rendre plus vivante. Dans un navigateur, JavaScript ne fait rien « tout seul ». Il a besoin d'être lancé depuis les pages web HTML. Pour appeler du code JavaScript depuis votre document HTML, vous aurez besoin de l'élément <script>. Il y a deux méthodes pour utiliser script : une qui sert lorsqu'on souhaite utiliser un script contenu dans un fichier tiers et une qui sert lorsqu'on intègre directement le code du script dans la page web.

C'est avec l'union de ces trois langages qu'il est donc possible de rendre une page web immersive et beaucoup plus ergonomique pour les utilisateurs ce qui est important lors de la présentation d'un projet dans notre cas.

3.7 Répartitions des tâches

La répartition des tâches n'a à ce stade pas changé.

Tâches	Adam	Zaky	Christian	Robin
SDL				
Gestion des images			R	S
Gestion des couleurs		S	R	
Gestion de la structure des images	S	R		
Filtres			S	R
GTK - Interface				
Menu utilisateur	S			R
Développement des outils		S		R
Ouverture / Sauvegarde de fichiers			R	S
Zone de dessin	R	S		
Pop-ups			R	S
Structure				
Design de l'application		S	R	
Architecture des fichiers sources	S			R
Raccord des parties (Compilation, makefile)	R	S		
Présentation				
Site internet			R	
Documents latex	S	R		

TABLE 1 – Répartition des Tâches - S :Suppléant - R : Responsable

3.8 Planification

La planification n'a également pas changé à ce stade.

Tâches	Soutenance 1	Soutenance 2	Soutenance 3
GTK - Interface			
Menu utilisateur	100%	100%	100%
Développement des outils	50%	70%	100%
Ouverture / Sauvegarde de fichiers	100%	100%	100%
Zone de dessin	50%	80%	100%
Pop-ups	45%	100%	100%
SDL			
Gestion des images	50%	85%	100%
Gestion des couleurs	70%	95%	100%
Gestion de la structure des images	70%	100%	100%
Filtres	50%	100%	100%
Interface graphique			
Design de l'application	40%	70%	100%
Architecture des fichiers sources	70%	100%	100%
Raccord des parties	30%	60%	100%

TABLE 2 – Planification de l'avancement des tâches

3.9 Ressenti personnel

3.9.1 AZIKA Christian

Tout au long de cette seconde étape de notre projet, j'ai grandement amélioré mes capacités à développer des interfaces, et plus généralement à travailler et à gérer les problèmes de mémoires liées à C. Par ailleurs, le code devenant de plus en plus grand, j'ai pu aiguisé mes capacités d'organisation et tout ce qui tourne autour des conventions de notations, afin de pouvoir produire du code propre et surtout compréhensible. Je me suis occupé de relier une grande partie des outils présents dans notre logiciel à l'interface, et je pense avoir réussi ma tâche avec succès. Je pense que nous avons surmonté nos difficultés, bien que j'ai eu du mal au début à m'organiser au niveau du timing.

3.9.2 HAMDOUN Zaky

Contrairement à la première soutenance, durant laquelle je pensais que l'aspect algorithmique de notre projet était faible, j'ai l'impression lors de cette deuxième étape que nous avons renforcé cet aspect algorithmique là par l'implémentation de différents outils et structures qui nécessitent une nouvelle pensée du stockage des informations visuelles des images. Je suis plutôt satisfait de l'avancement du projet, bien que je n'ai pas su m'organiser correctement au niveau du timing. Je pense qu'il s'agit là d'une bonne alternative entre paint et photoshop, permettant assez de modifications d'images, tout en restant simple.

3.9.3 MAHRAOUI Adam

Je me suis énormément amélioré quant à la création de sites web. Là où avant j'aurais du utilisé un éditeur de site internet en ligne, j'ai pas tout faire de A à Z sans trop de difficultés, et je suis extrêmement fier du rendu du site internet, qui je suis sûr est beaucoup meilleur que le site précédent.

3.9.4 VARLIETTE Robin

Durant cette deuxième soutenance, j'ai du faire face à l'implémentation d'une structure de donnée à part entière afin de faire de rendre possible la fonctionnalité de retour avant/ retour arrière. Cette structure de donnée étant les piles, nous l'avions déjà abordé durant notre deuxième semestre à EPITA, cela a donc été plus facile que prévu.

Cependant, j'ai du me remettre en question quant à l'implémentation de cette structure de donnée, car je pensais que il aurait été suffisant de simplement utiliser un tableau de surfaces SDL dynamiquement alloué. Cette approche bien que plus simple était néanmoins moins optimisé en terme de temps, et de toute manière pauvre vis à vis de l'aspect algorithmique.

3.10 Conclusion

En conclusion, il est clair que notre projet a connu une avancée impressionnante entre la première et la deuxième soutenance. Nous sommes fiers de constater que, grâce à notre travail acharné et notre détermination, nous avons tous progressé dans nos compétences et notre compréhension du projet. Nous avons réussi à relever les défis et à surmonter les obstacles pour arriver à une version améliorée du projet.

Nous sommes reconnaissants envers les membres de l'équipe pour leur engagement et leur collaboration tout au long du processus, ainsi que pour leur contribution au projet. Nous avons travaillé ensemble de manière cohérente et avons réussi à tirer le meilleur parti de nos compétences individuelles pour atteindre nos objectifs communs.

Même si nous avons un peu de retard, nous sommes totalement confiants du fait que nous allons pouvoir atteindre tous nos objectifs d'ici la soutenance finale.

En fin de compte, notre projet est un exemple réussi de ce que nous pouvons accomplir en travaillant ensemble avec détermination et persévérance. Nous sommes impatients de continuer à travailler sur ce projet et de voir où cela nous mènera à l'avenir.