

Projet - Editeur de textures Animées

Introduction

Le but du projet est de réaliser un éditeur de textures animées, c'est à dire des éléments graphiques dessinés ou d'images importées par l'utilisateur qui évoluent en fonction du temps.

Le projet est découpé en deux parties séparées. Une première partie individuelle de conception et de début de programmation. Une seconde partie en binômes de programmation avancée.

La programmation sera effectuée en Java avec la librairie Swing.

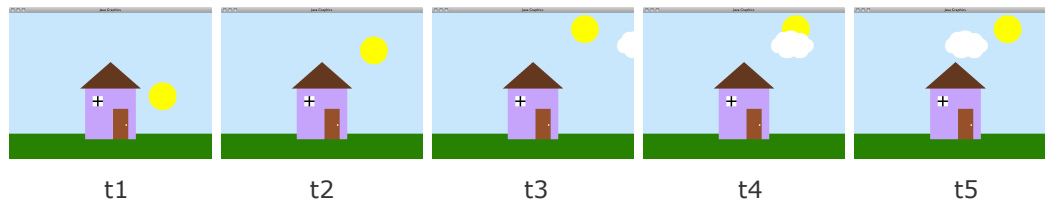
Vous pouvez éventuellement utiliser d'autres librairies ou langages de programmation mais vous devez contacter l'enseignant pour obtenir son accord. Vous pouvez également proposer des variations du sujet mais vous devez aussi avoir l'accord de votre enseignant d'abord.

Sujet

Le but du projet est de réaliser un éditeur graphique permettant le dessin et la création de textures animées, c'est à dire qui évoluent en fonction du temps. L'idée est de créer une application de dessin simple sur un canvas qui permet d'animer les éléments contenus ou dessinés dans le canvas.

L'application doit pouvoir s'utiliser sur ordinateur avec une souris. Votre objectif sera de concevoir des techniques d'interaction pertinentes pour la création et la modification d'animations sur les éléments du canvas par les utilisateurs.

Exemple de textures animées que votre système devrait permettre de créer :



Voici une liste de fonctionnalités minimale souhaitées pour votre système :

- Fonctions de dessin
 - Dessin de formes géométriques simples (lignes, rectangle et ellipses) et dessin libre à la souris sur le canvas.
 - Ajout d'images existantes dans le canvas.
 - Modifications des caractéristiques des éléments (taille, position, couleur, taille du trait) présent.
 - Possibilité de supprimer un élément du canvas.
- Fonctions d'animations
 - Animations simples (clignotement, changement de taille, mouvement de chute libre qui s'arrête en bas du canvas, oscillations de haut en bas ou droite à gauche...)
 - Permettre l'animation d'un élément selon une trajectoire tracée par l'utilisateur à la souris.
 - Réglage de la vitesse d'animation (fonctions pour augmenter ou réduire la vitesse globale des animations).

Vous pouvez ajouter d'autres fonctionnalités ou vous concentrer sur certains aspects en particulier si vous le souhaitez. Attention à ne pas faire quelque chose d'impossible à réaliser par la suite car la seconde partie du projet se concentrera sur la programmation de techniques d'interactions sur la base de vos propositions.

Dans tous les cas, il faut se concentrer sur l'interaction et l'utilisation de techniques avancées. En particulier, l'utilisation de *manipulation directe* et/ou de *drag and drop* et/ou de *reconnaissance de gestes* et/ou la proposition de nouvelles techniques sera très appréciée. Restez toutefois vigilant à la cohérence de votre application.

Partie 2 (à rendre le 10 avril au plus tard) :

Cette deuxième partie se concentre sur l'implémentation avancée de votre éditeur de textures animées. Vous devez désormais travailler en binômes et vous devez envoyer la liste des binômes à votre chargé de TP ainsi qu'à votre enseignant (jgarcia@ircam.fr) avant le 10 avril.

Travail de conception/programmation :

Vous devez programmer une version améliorée de l'application avec l'ensemble des fonctionnalités de dessin et d'animation. Votre version doit comprendre au moins deux techniques avancées d'interaction (drag and drop, reconnaissance de geste, crossing, guides magnétiques, etc.) et faciliter l'utilisation avec des mécanismes d'aide et de feedback.

Vous trouverez à la fin de ce document plusieurs exemples de widgets, techniques d'interactions et mécanismes de feedback dont vous pouvez vous inspirer.

Rendu

Vous devez envoyer les éléments suivants à votre chargé de TP ainsi qu'à votre enseignant (jgarcia@ircam.fr) dans un fichier compressé nommé PIAA15_nom1_nom2_projet2.zip :

1. Un rapport au format **pdf** comprenant
 1. Une petite introduction et un visuel de votre application (but et concept)
 2. Une liste des fonctionnalités implémentées et comment y accéder. Vous pouvez les décrire en texte, avec des storyboards. Détailler surtout les techniques d'interaction avancées.
 3. Décisions : listez les décisions que vous avez faites. Si vous avez eu plusieurs idées d'interactions ou façons de présenter vos fonctionnalités, décrivez vos alternatives et pourquoi vous avez gardé celles présentes dans l'application. Vous pouvez joindre des storyboards de techniques qui ne sont plus utilisées mais que vous aviez conçues.
 4. Principes de conception et de perception : Expliquer avec des exemples de votre interfaces quel sont les principes de conception et les principes de perception utilisés dans votre interface.
 5. Réflexion/Conclusion : Si on vous demandait maintenant de refaire le projet, comment est ce que vous procéderiez ? Quels sont les choix/partis de votre processus qui ont bien marché et que vous feriez différemment ?

Le rapport ne doit pas dépasser **8 pages** mais peut contenir des illustrations ou des storyboards en annexe si vous en avez plusieurs.

2. Les sources et la version exécutable (runnable jar), de votre projet. Ajoutez les explications nécessaires pour le lancement du programme.

Conseils :

Essayez de définir l'ampleur de votre projet avant de commencer. Développer plus de fonctionnalités ne se traduit pas toujours à une meilleure note. Ce qui est le plus important pour ce devoir, c'est la réalisation de techniques d'interaction ainsi que de la qualité de votre application finale.

Votre implémentation doit être *utilisable et fonctionnelles*. Elle doit gérer si possible les actions non prévues de l'utilisateur et fournir des mécanismes de feedback (retour utilisateur) qui aident l'utilisateur à éviter les erreurs.

Essayez de bien structurer et commenter votre code.

Soyez créatifs !

Exemples de widgets et techniques d'interaction :

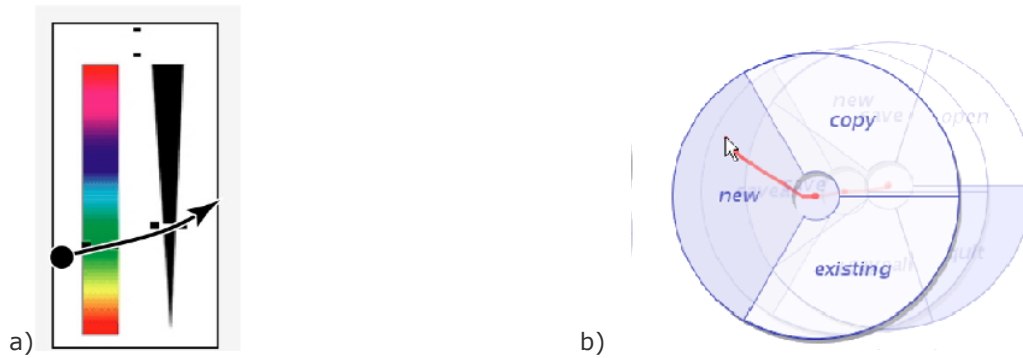


Figure 1: (a) Sélection de la couleur et de la taille du trait par franchissement (crossing), (b) menu gestuel.

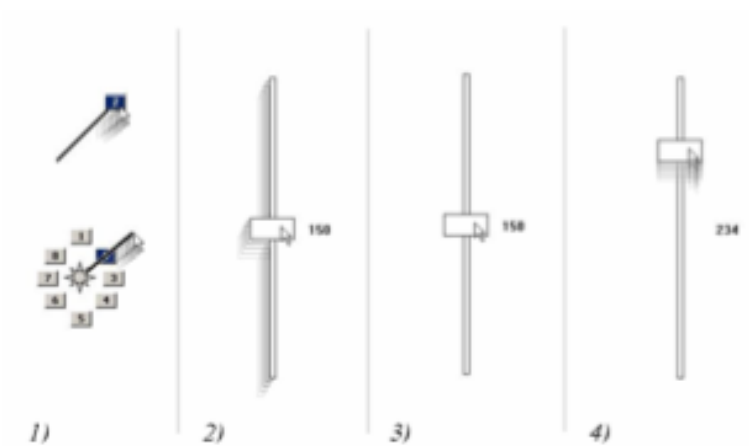


Figure 2 : (1) L'utilisateur invoque le menu (ex avec un geste) et avec un mouvement à droite il sélectionne l'option 2, qui fait apparaître le slider (2). L'utilisateur continue son mouvement (3,4) qui manipule le slider. Cette interaction est faite en un seul drag. L'idée de FaST Sliders viens d'un article de Michael McGuffin , Nicolas Burtnyk , et Gordon Kurtenbach (publiée à Graphics Interface 2002.)

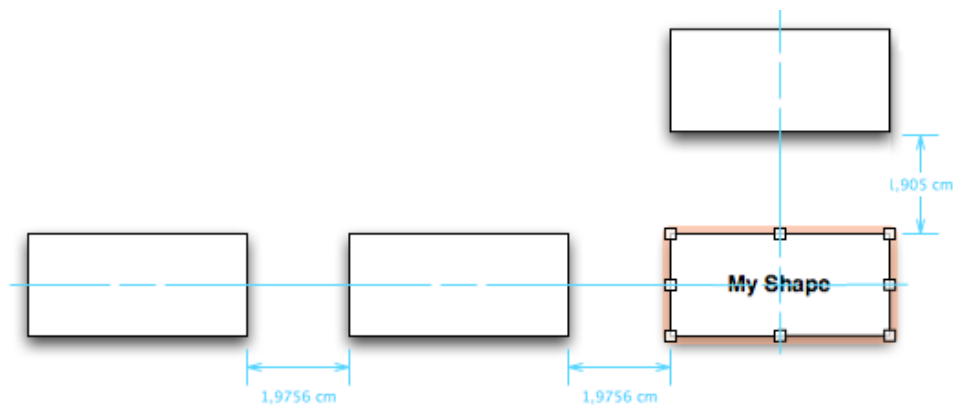


Figure 3 : Dans le logiciel Omnigraffle, lorsque l'utilisateur déplace (drag) une forme, des guides magnétiques sont créés par les objets du canvas afin de les aligner ensemble sur leurs bords ou leur centre. Des indications de distances apparaissent également pour indiquer lorsque l'espacement est le même entre plusieurs objets.