

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.

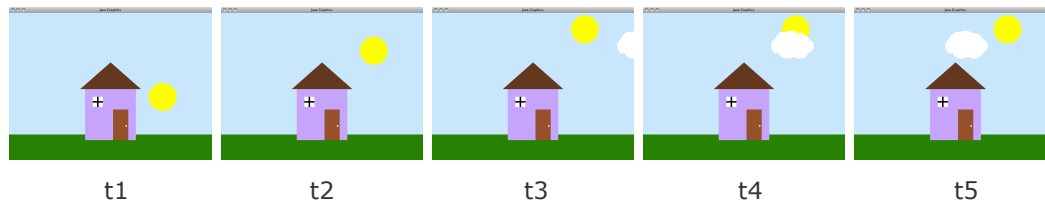
Vous pouvez éventuellement utiliser d'autres librairies ou langages de programmation mais vous devez contacter l'enseignant pour obtenir son accord. Vous pouvez également proposer des variations du sujet mais vous devez aussi avoir l'accord de votre enseignant d'abord.

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 1 (à rendre le 3 mars au plus tard) :

Comme il est difficile de construire un système complet, nous vous proposons tout d'abord une étape de conception et de programmation simplifiée. Vous devrez réfléchir à l'aspect visuel et aux interactions de l'utilisateur puis vous programmerez une petite partie du système.

Travail de conception :

Nous attendons de la partie de conception que vous proposiez l'aspect visuel de l'application et les techniques d'interactions qui permettront à un utilisateur d'utiliser les fonctions de dessin et d'animation. En particulier vous devez vous concentrer sur les aspects suivants : facilité d'utilisation et de compréhension de la technique, choix de métaphore et de style d'interaction, aide proposée à l'utilisateur, feedback et cohérence entre les différentes techniques.

Travail de programmation :

Vous devez programmer une première version de l'application avec l'ensemble des fonctionnalités de dessin décrites précédemment et la possibilité de faire du drag and drop pour ajouter une l'animation de clignotement, c'est à dire de disparition puis de réapparition à la même position avec les mêmes attributs graphique, à un élément du canvas.

Cette technique d'interaction consiste à avoir une représentation de la transformation de clignotement (une icône ou du texte par exemple) dans une barre d'outils de votre application et de pouvoir l'appliquer sur un élément du canvas avec du drag and drop. Attention, le drag and drop doit se faire entre deux composants séparés de l'application, pas au sein du canvas directement.

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_nom_prenom_projet1.zip :

1. Une description courte de votre projet dans un fichier pdf. Comprenant un visuel de votre application (dessin à la main ou image) et des explications sur les choix vous avez fait au niveau de l'interface graphique, du modèle et des techniques d'interactions. Pensez à justifier vos choix à partir des éléments vus en cours (Conception graphique, modèles conceptuels, styles d'interaction ...). 4 pages maximum.
2. Des storyboards (en tant qu'image) d'au moins trois techniques que vous envisagez (fait en papier ou transparents **avant** l'implémentation). Un storyboard par technique d'interaction montrant un état initial, la réalisation de la technique (si besoin en plusieurs étapes) et le résultat final.
3. Les sources et la version exécutable (runnable jar), de votre. 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 conception et la réalisation de techniques d'interaction ainsi que de la qualité de vos storyboards et du code.

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 !