

Saé 2.01 – Développement d'une application

Lecteur de diaporamas – Dossier d'Analyse et conception

Lien Github: <https://github.com/CRUSSIERS/SAE2.1/tree/main>
CRUSSIERS Lucas - GRATCHEV Grigori TD3/TP5

1. Compléments de spécifications externes.	2
2. Scénarios	2
3. Diagramme de classe (UML)	3
Version v0 – Version console seule	6
4. Implémentation et tests	6
4.1 Implémentation	6
4.2 Test	6

1. Compléments de spécifications externes.

Pas de spécifications externes en plus

2. Scénarios

Description du scénario nominal et de un / deux scénarios alternatifs afin de mettre en évidence les interactions entre le système et l'utilisateur

Lecture du diaporama Pré-conditions: <ul style="list-style-type: none">- l'utilisateur a lancé l'application- Le diaporama est chargé	
1. L'utilisateur clique sur le bouton "suivant"	
	2. Le lecteur de diaporama affiche l'image suivante dans le diaporama
3. L'utilisateur clique sur le bouton "précédent"	
	4. Le lecteur de diaporama affiche l'image précédente dans le diaporama
5. L'utilisateur clique sur le bouton "Démarrer le mode auto"	
	6. Le lecteur défile toutes les images

	automatiquement
7. L'utilisateur clique sur le bouton "Arrêter le mode auto"	
	7. Le lecteur arrête de faire défiler les images automatiquement

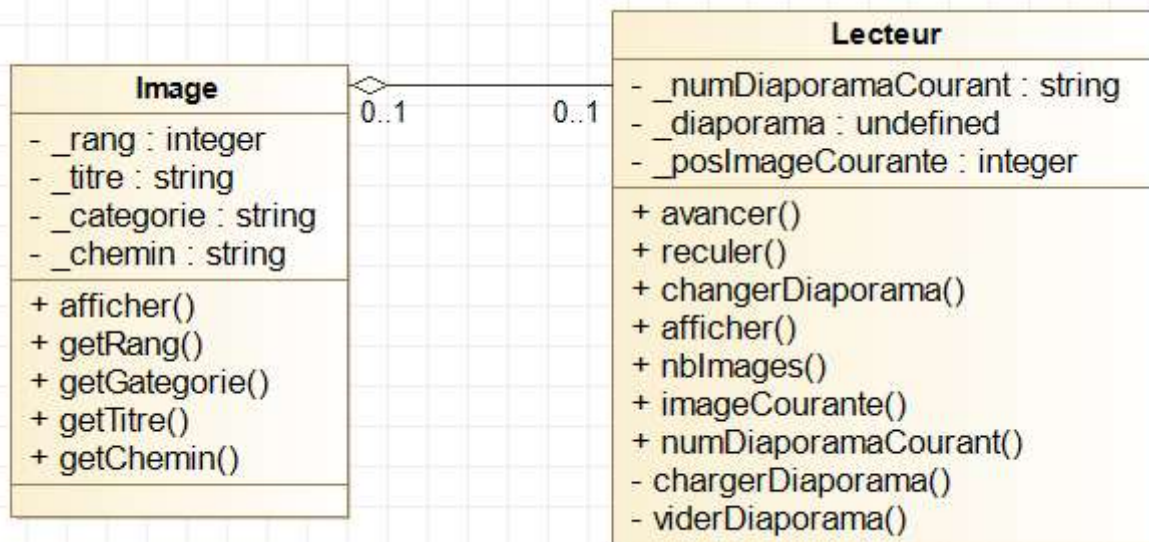
Scénarios alternatifs:

Lecture du diaporama Pré-conditions: - l'utilisateur a lancé l'application - Le diaporama est démarré	
1. L'utilisateur clique sur le bouton suivant	
	2. Le diaporama est vide et affiche "Impossible d'avance: fin du diaporama atteinte"

Lecture du diaporama Pré-conditions: - l'utilisateur a lancé l'application - Le diaporama est sur la dernière diapositive	
1. L'utilisateur clique sur le bouton suivant	
	2. Le diaporama recommence à la diapositive 1

3. Diagramme de classe (UML)

- (a) Le diagramme de classes UML se focalise sur les classes **métier**, cad celles décrivant les éléments structurants de l'application, indépendamment des éléments d'interface.



Nous avons mis le type undefined pour “_diaporama” car on ne peut pas créer de type sur modelio, le type est normalement Diaporama.

(b) Dictionnaire des éléments pour chaque classe

Classe Image			
Nom attribut	Signification	Type	Exemple
rang	rang de l'image au sein du diaporama auquel l'image est associée	int ege r	1
Titre	Intitulé de l'image	str ing	Donald
Categorie	Catégorie de l'image	str ing	personne, animal, objet
Chemin	Chemin complet vers le dossier où se trouve l'image	str ing	\\haya\dossiersetud_BaieSsd\ggratchev\Downloads\cartesDisney\Disney_2.gif

Tableau : Dictionnaire des éléments - Classe Image

Classe Lecteur			
Nom attribut	Signification	Type	Exemple
numDiaporamaCourant	Numéro du diaporama courant, par défaut 0	integer	1
Diaporama	Pointeurs vers les images du diaporama	vector	image(rang:2, titre:Cendrillon, categorie:personne, chemin:\\haya\dossiersetud_BaieSsd\ggratchev\Downloads\cartesDisney\Disney_2.gif)
posImageCourante	position, dans le diaporama, l'image courante. Indéfini quand diaporama vide. Démarre à 0 quand diaporama non vide	integer	0

Tableau : Dictionnaire des éléments - Classe Lecteur

(c) Dictionnaire des méthodes : vous pouvez fournir directement le fichier entête de chaque classe.

Exemple (classe lecteur de la version Console) :

```
#ifndef LECTEUR_H
#define LECTEUR_H
#include "image.h"
#include <iostream>
#include <vector>

typedef vector<Image*> Diaporama; // Structure de données contenant les infos sur les images

class Lecteur
{
public:
    Lecteur();
    void avancer(); // incrémente _posImageCourante, modulo nbImages()
    void reculer(); // décrémente _posImageCourante, modulo nbImages()
    void changerDiaporama(unsigned int pNumDiaporama); // permet de choisir un diaporama, 0 si
    // aucun diaporama souhaité
    void afficher(); // affiche les informations sur lecteur-diaporama et image courante
    unsigned int nbImages(); // affiche la taille de _diaporama
    Image* imageCourante(); // retourne le pointeur vers l'image courante
    unsigned int numDiaporamaCourant();

private:
    unsigned _numDiaporamaCourant; // numéro du diaporama courant, par défaut 0
    Diaporama _diaporama; // pointeurs vers les images du diaporama
    unsigned int _posImageCourante; // position, dans le diaporama,
    // de l'image courante.
    // Indéfini quand diaporama vide.
    // Démarre à 0 quand diaporama non vide */
    void chargerDiaporama(); // charge dans _diaporama les images du _numDiaporamaCourant
    void viderDiaporama(); // vide _diaporama de tous ses objets image et les delete
};

#endif // LECTEUR_H
```

Figure : Schéma de classes = Classe Lecteur

```
#ifndef IMAGE_H
#define IMAGE_H
#include <iostream>
using namespace std;

class Image
{
public:
    Image(unsigned int pRang=0,
           string pCategorie="", string pTitre="", string pChemin = "");
    unsigned int getRang();
    string getCategorie();
    string getTitre();
    string getChemin();
    void afficher(); // affiche tous les champs de l'image

private:
    unsigned int _rang; // rang de l'image au sein du diaporama
    // auquel l'image est associée */
    string _titre; // intitulé de l'image
    string _categorie; // catégorie de l'image (personne, animal, objet)
    string _chemin; // chemin complet vers le dossier où se trouve l'image
};

#endif // IMAGE_H
```

Figure : Schéma de classes = Classe Image

(d) Remarques concernant le schéma de classes

1. On ne s'intéresse qu'aux attributs et méthodes métier. Notamment, on ne met pas, pour l'instant, ce qui relève de l'affichage car ce sont d'autres objets du programme (widgets) qui se chargeront de l'affichage. Par contre, on n'oublie pas les méthodes `getXXX()`, qui permettront aux objets métier de communiquer leur valeur aux objets graphiques pour que ceux-ci s'affichent.
2. On n'a mis ni le constructeur ni le destructeur, pour alléger le schéma.
3. D'autres attributs et méthodes pourront venir ultérieurement compléter cette première vision ANALYTIQUE de l'application. Il s'agira des attributs et méthodes dits DE CONCEPTION nécessaires au développement de l'application.

Version v0 – Version console seule

4. Implémentation et tests

4.1 Implémentation

Liste et rôle des fichiers de cette version :

lecteur.h	Spécification de la classe Lecteur
lecteur.cpp	Corps de la classe Lecteur
image.h	Spécification de la classe Image
image.cpp	Corps de la classe Image
main.cpp	Teste les méthodes de la classe Lecteur

4.2 Test

Test avec le programme fournit main.cpp

Valeurs fournies / attendues... comme montré dans la ressource R2.03 (partie tests)

a) Résultat attendu

```
C:\Qt\Tools\QtCreator\bin\qtcreator_process_stub.exe
Lecteur vide
Diaporama num. 1 selectionne.
4 images chargees dans le diaporama
Diaporama num. 1
image courante : image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif)

Test avancer() : 4 fois
avancer() :
Diaporama num. 1
image courante : image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif)
avancer() :
Diaporama num. 1
image courante : image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif)
avancer() :
Diaporama num. 1
image courante : image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif)
avancer() :
Diaporama num. 1
image courante : image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif)

Test reculer() : 5 fois
reculer() :
Diaporama num. 1
image courante : image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif)
reculer() :
Diaporama num. 1
image courante : image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif)
reculer() :
Diaporama num. 1
image courante : image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif)
reculer() :
Diaporama num. 1
image courante : image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif)
reculer() :
Diaporama num. 1
image courante : image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif)

Enlever le diaporama courant = Choisir diaporama 0
0 images restantes dans le diaporama.
Lecteur vide
Press <RETURN> to close this window...
```

b) Résultat obtenu

The image displays a Qt IDE environment with two main panels. The left panel shows the project structure for 'lecteurDiaporama', including headers ('image.h', 'lecteur.h') and sources ('image.cpp', 'lecteur.cpp', 'main.cpp'). The right panel shows the source code of 'lecteur.cpp' and the application's output window.

Source Code (lecteur.cpp):

```
#include "lecteur.h"

Lecteur::Lecteur()
{
    _numDiaporamaCourant = 0; // = le lecteur est vide
}

void Lecteur::avancer()
{
    if (numDiaporamaCourant() > 0 && _posImageCourante < nbImages() - 1)
    {
        _posImageCourante++;
    }
}
```

Output Window:

```
Sortie de l'application
lecteurDiaporama x sae_v2 x
Lecteur vide.
16:14:13: F:\TP\R202DevAplisAvecIHM\build-lecteurDiaporama-Desktop_Qt_6_3_1_MinGW_64_bit-Debug\debug\lecteurDiaporama.exe
16:13:08: Starting F:\TP\R202DevAplisAvecIHM\build-lecteurDiaporama-Desktop_Qt_6_3_1_MinGW_64_bit-Debug\debug\lecteurDiaporama.exe
Lecteur vide.
Diaporama num. 1 selectionne.
4 images chargees dans le diaporama
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif)

Test avancer() : 4 fois
Image suivante
image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif)
avancer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif)

Image suivante
image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif)
avancer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif)

Image suivante
image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif)
avancer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif)

Impossible d'avancer : fin du diaporama atteinte
avancer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif)

Test reculer() : 5 fois
Image 2 affichee.
reculer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif)

Image 1 affichee.
reculer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:2, titre:Cendrillon, categorie:personne, chemin:C:\cartesDisney\carteDisney4.gif)

Image 0 affichee.
reculer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:1, titre:Grincheux, categorie:personne, chemin:C:\cartesDisney\carteDisney1.gif)

Image 3 affichee.
reculer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:4, titre:Mickey, categorie:animal, chemin:C:\cartesDisney\carteDisney1.gif)

Image 2 affichee.
reculer() :
Diaporama num. 1 est en cours de lecture.
Image courante : image( rang:3, titre:Blanche Neige, categorie:personne, chemin:C:\cartesDisney\carteDisney2.gif)

Enlever le diaporama courant = Choisir diaporama 0
0 images restantes dans le diaporama.
non chargee
Lecteur vide.
16:13:08: F:\TP\R202DevAplisAvecIHM\build-lecteurDiaporama-Desktop_Qt_6_3_1_MinGW_64_bit-Debug\debug\lecteurDiaporama.exe exited with code 0
```